ISSN 2581 – 3781

AJCSE
ASIAN JOURNAL OF COMPUTER SCIENCE ENGINEERING

Available Online at www.ajcse.info
Asian Journal of Computer Science Engineering 2023;8(4):1-9

REVIEW ARTICLE

Balancenet: Addressing Class Imbalance in Ai-Powered Intrusion Detection Through Adaptive Sampling

Gaurav Sarraf*

Independent Researcher

Received on: 15-10-2023; Revised on: 12-11-2023; Accepted on: 20-12-2023

ABSTRACT

The constantly increasing cases of computer attacks in the modern digitally connected world have led to the necessity of the most efficient intrusion detection systems (IDSs). Since innocuous traffic flow greatly outweighs the occurrence of attacks, one of the most crucial difficulties in IDSs is investigating the class imbalance of data flow from networks. Since this is the case, it impacts the accuracy with which machine learning algorithms detect dangers to minority classes. The research study introduces an IDSs that uses adaptive sampling techniques to tackle the issue of network traffic class imbalance. It uses the UNSW-NB15 dataset, eXtreme Gradient Boosting (XGBoost), and oversampling based on ADASYN, and it promises to improve the capacity to detect intrusions that impact minority classes. The model's 99.59% accuracy, 99.8% precision, 99.5% recall, and 99.6% F1-score indicate that it is very good at detecting harmful activity with few false alarms. In comparison to LR, NB, and LSTM, XGBoost performs better across the board when it comes to critical metrics. The combination of adaptive data balancing with a robust ensemble classifier provides a scalable and robust solution to real-time network anomaly detection in complex and unbalanced network settings, which can be used to further develop intelligent cybersecurity systems.

Key words: Cyberattack, internet of things, intrusion detection system, machine learning, network traffic, UNSW-NB15

INTRODUCTION

The increasing number of sensor-based data streams in the era of the Internet of Things (IoT) has brought about new possibilities and threats in the field of cybersecurity.[1,2] New studies have shown that there is a growing number of cybersecurity risks to sensor-based systems, including autonomous systems and IoT networks.[3,4] One example is the IoT infrastructure, which exposes autonomous systems to the risk of distributed denial of service and data manipulation attacks because of its weak processing capacity and absence of security measures. Network resources must be kept available, private, and secure at all times; intrusion detection systems (IDSs) help with this by setting up protections for when danger strikes. They fall into two main categories: Signature-based detection, which looks for previously identified patterns of threats, and anomaly-based detection, which uses a normalized pattern of network activity to identify potentially dangerous ones. Nonetheless, a major problem experienced when applying IDS is that of uneven training data.[5,6]

Machine learning (ML) is becoming a promising solution to the limitations of traditional IDS, as it has attracted the attention of the cybersecurity community.[7] ML based IDS utilizes the behavior analysis to identify anomalies and threats and provides the possibility of much greater accuracy and shorter detection times.[8,9] This is a paradigm change in the field of IDS which promises to not only enhance security but also transform the privacy scene.[10] The effectiveness of ML algorithms is that they can detect threats, but this usually requires sensitive information.[11,12] ML in cybersecurity can be used as an effective tool to enhance the capacity of systems to interpret various patterns as well as predict possible data threats.

Address for correspondence:
Gaurav Sarraf
E-mail: sarrafgsarraf@gmail.com

Motivation and Contribution

Cyberattacks on vital network infrastructures are becoming more sophisticated and common, necessitating the development of reliable IDS. Conventional detection techniques are generally ineffective with high-dimensional data, class imbalance, and changing attack patterns, resulting in decreased accuracy and slower threat response. This project aims to provide a robust framework to support real-time network security monitors, enhance detection rates, decrease false alarms, and apply state-of-the-art ML models for efficient data preparation, feature selection, and class balancing. This study has a number of important contributions as follows:

- Created a full pipeline of pre-processing, consisting of cleaning, encoding, normalization, and class balancing with ADASYN on the UNSW-NB15 data.
- Applied Chi-square statistical techniques to choose the most pertinent features, which minimizes the complexity of the computation and maximization of the performance of the model.
- Enhanced attack traffic categorization using XGBoost, a hybrid of adaptive sampling and feature selection.
- The model's performance was evaluated using receiver operating characteristic (ROC) curve analysis, F1, REC, ACC, and PRE, among other tools.

The proposed model also deals with an important problem of IDS, which is the issue of class imbalance, by combining adaptive sampling with an ensemble classifier with high performance. This guarantees enhancement in detecting minority-class attacks, which are usually ignored by the traditional models. It is novel in the sense that it integrates ADASYN with XGBoost to achieve the best learning based on thin threat patterns and high accuracy, and low false alarms. The solution not only enhances detection reliability, but also adds to the modern cybersecurity systems with a scalable and data-sensitive solution.

Organization of the Paper

The structure of the paper is as follows: Study on IDS methods that is relevant to this topic is reviewed in Section II. Section III details the method that is being suggested. In Section IV, shows the experimental findings and compare how well the models performed. Conclusions and suggestions for further research are provided in Section V, which also summarizes the study's main findings.

LITERATURE REVIEW

The construction of this study was guided and strengthened by a comprehensive assessment and analysis of significant research works on IDS.

Kabir *et al*. developed an IDSs and intrusion prevention system model for an entire network. Using the ET Classifier and Mutual Information Gain feature selection techniques, this work presents two independent stacking ML models to increase the NIDS's ACC. One of the suggested models outperforms all other competing models in terms of ACC (96.24%), according to the comparison data.[13]

Gupta and Saxena (2022) despite advancements, the majority of commercial IDS that are currently available rely on signatures to identify intruders. Recently, anomaly detection has seen a rise in the use of ML-based classification algorithms. Results, recall, and ACC for the majority of ML methods on this dataset were 90% or higher. On the other hand, radial basis function is the best of the seven algorithms when looking at the area under the ROC.[14]

Umamaheshwari *et al*. (2021) employs a WSN-DS dataset that is open to the public to assess the system's efficiency. All of the suggested feature selection methods are tested with important performance indicators. Train duration, ACC, sensitivity, and specificity are 15.12 s, 98.58%, 92.81%, and 98.46%, respectively, while using MRMR feature selection. Thereby, a solid IDS in a WSN might be predicated on this research.[15]

Das *et al*. offer a non-linear learning PIDS that integrates ML and NLP ensembles. A number of supervised and ensemble-based ML models are trained using the language-based vectors converted by the proposed NLPIDS from HTTP requests. With a lower number of false alarms (0.007) and a higher F1-score (0.999), the NLPIDS clearly outperforms competing methods. The NLPIDS is independent of attack vectors and tactics.[16]

Srivastava *et al*. (2019) helped identify suspicious activity in the data pertaining to the traffic on the

network. Much study has focused on the use of ML algorithms for anomaly identification in network data. The public repositories now accommodate additional datasets. Using innovative feature reduction-based ML algorithms, the authors of this paper were able to spot suspicious patterns in the newly supplied dataset. A level of 86.15% ACC has been maintained.[17]

Singh and Mathai (2019) used the NSLKDD dataset for ML classification and compared the SPELM approach to its DBN counterpart. Computer time (90.8 vs. 102 s), accuracy (93.20 vs. 52.8%), and precision (69.492 vs. 66.836) are three areas where SPELM excels beyond the DBN method.[18]

Table 1 provides an overview of current studies on adaptive sampling for IDS, including the models suggested, datasets used, important results, and problems encountered. There are still a number of unanswered questions about IDS, even though these technologies have made great strides in recent years. Most studies depend on popular datasets such as UNSW-NB15, NSL-KDD, and Kyoto 2006+, which may not reflect the dynamic nature of 0-day threats and complex multi-stage invasions. This is a problem in the current state of cyberattack research. The ACC of detection has been enhanced by ensemble methods and feature selection strategies; nonetheless, numerous systems continue to face challenges when dealing with high-dimensional data, processing in real-time, and minimizing false positives. In addition, limited research has addressed adaptive or hybrid models that can dynamically adjust to new attack patterns without frequent retraining. There is also a lack of comprehensive studies integrating anomaly-based and signature-based detection to balance detection speed, ACC, and robustness across heterogeneous network environments. Due to these shortcomings, IDS requires to be more flexible, scalable, and proven in the real world.

RESEARCH METHODOLOGY

This study employs the UNSW-NB15 dataset, applying pre-processing steps such as cleaning, encoding, normalization, and ADASYN-based class balancing. Utilizing Chi-square feature selection allows for the preservation of critical attributes while simultaneously improving the model's performance. Following this, the cleaned-up dataset was split in half: half to be used for model training and half for model testing. For classification, the XGBoost model is employed, and its performance is evaluated using Accuracy, Precision, Recall, F1-score, and ROC curve analysis. The overall architecture of the proposed IDS is presented in Figure 1.

The whole steps of implementation are explained in next section.

Table 1: Recent studies on intrusion detection systems using machine and deep learning techniques

Author	Proposed work	Results	Key findings	Limitations and future work
Kabir *et al*. (2022)	ML NIDS algorithms utilizing ET classifiers and mutual information gain	Testing ACC of stacking models: 96.24%	Stacking models outperform individual models; enhanced detection ACC on UNSW-NB15 dataset	Further optimization could improve performance on emerging attack types
Gupta and Saxena (2022)	Applied seven ML techniques for anomaly detection on Kyoto 2006 + dataset using information entropy	Most ML models achieved ~ 90% ACC, with performing best (AUC)	ML-based approaches are more effective than signature-based methods for anomaly detection	Extend to real-time detection and newer datasets
Umamaheshwari *et al*. (2021)	Built IDS for WSN using ML; feature selection through correlation score, Fisher score, KW test, MRMR, and relief	ACC 98.58%, Sensitivity 92.81%, Specificity 98.46%, PRE 93.86%, Training time 15.12s	Feature selection reduces detection time and improves IDS performance	Apply to larger WSN datasets and real-time deployment
Das *et al*. (2020)	The proposed NLPIDS uses ensemble ML and natural language processing to identify HTTP requests.	Using the CSIC 2010 dataset, the results demonstrate an F1-score of 0.999 and a false alarm rate of 0.007.	NLPIDS is attack-independent and achieves high detection performance	Explore the application to other protocols and network types
Srivastava *et al*. (2019)	Used feature reduction-based ML algorithms to detect anomalies in network traffic	ACC 86.15%	Novel feature reduction techniques improve detection on recent datasets	Improve ACC and handle evolving attack types
Singh and Mathai (2019)	Proposed SPELM algorithm and compared with DBN using NSL-KDD dataset	SPELM: 93.20% versus 52.8% for ACC; PRE: 69.49% versus 66.736% for DBN.	SPELM outperforms DBN in accuracy and efficiency	Explore application to larger, more complex datasets and hybrid ML models

Data Gathering and Analysis

The UNSW-NB15 dataset, a new dataset, is referenced in this study. There are a total of 49 attributes in this dataset, with a class label and 25,40,044 tagged occurrences that are categorized as either normal or attack. Data visualizations such as bar plots and heatmaps were used to examine attack distribution, feature correlations, etc., and were given below:

Figure 2 provides a comprehensive visual overview of inter-feature relationships, highlighting both positive and negative associations among variables such as Time, Dist_To_CH, ADV_S, JOIN_R, Expanded Energy, and Attack type. Each cell encodes the correlation coefficient using a color gradient from blue (strong negative) to red (strong positive), with white indicating near-zero correlation. The circular markers within cells further emphasize the magnitude of these relationships, aiding in intuitive pattern recognition. This matrix is instrumental for feature selection and model refinement, revealing potential redundancies and dependencies critical to cybersecurity analytics.

The UNSW-NB15 dataset includes a wide range of damaging attacks and traffic types, as illustrated in Figure 3. Normal traffic is the dominant type of data, containing more than 50,000 records and the next most prevalent data is the generic traffic, which has a total of more than 30,000 records and the final and the most prevalent data are the Exploits, with the total of more than 30, 000 records. Fuzzing exhibits a significantly smaller, although still significant, number of 18,000 records and DOS and Reconnaissance attacks take their places, with counts ranging between 10,000 and 12,000. All other attack types Analysis, Backdoor, Shellcode, and Worms occupy a relatively small percentage of the dataset with only fewer than 2,000 records each, which suggests a very skewed

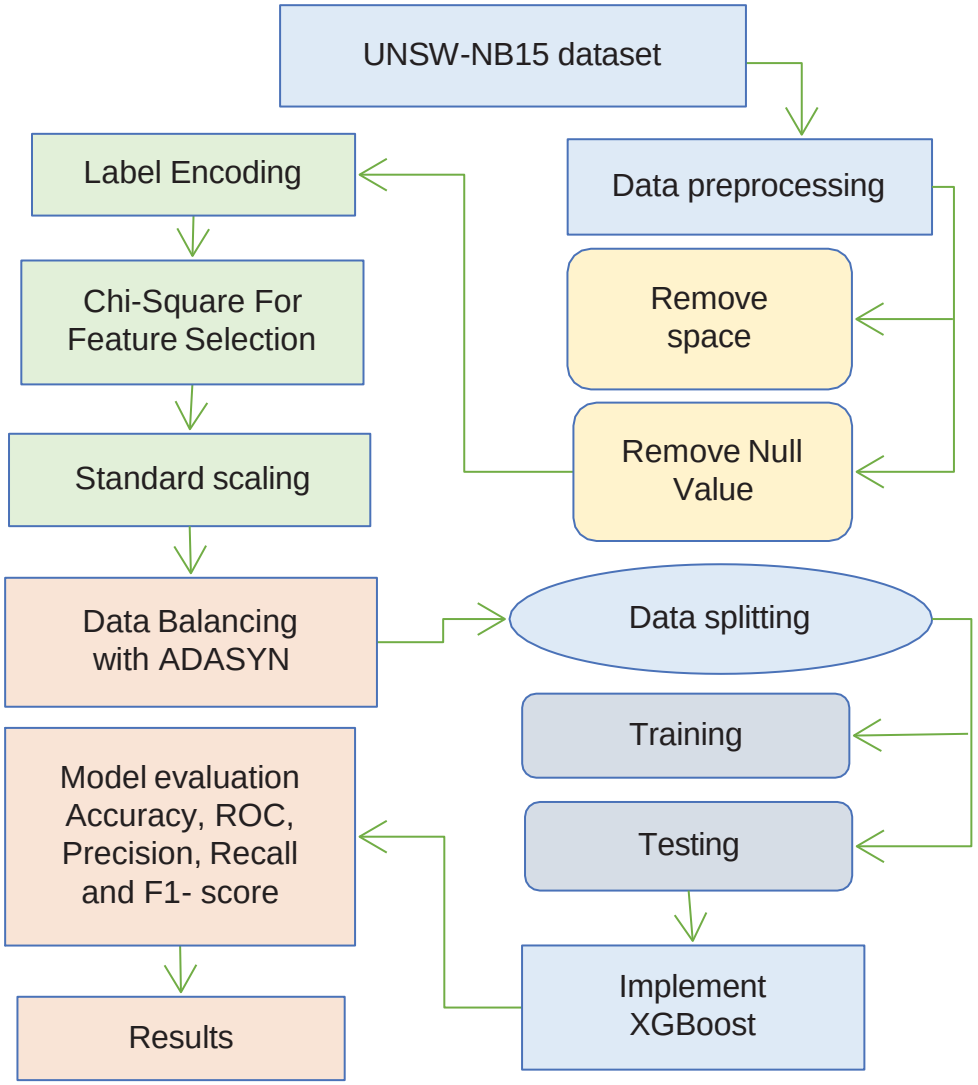

Figure 1: Proposed flowchart for intrusion detection system

distribution centered around normal traffic, generic detection and attempts to exploit.

Data Pre-processing

Data preparation used the UNSW-NB15 dataset and entailed concatenation, cleaning, and feature engineering. Its pre-processing steps involved handling of missing values, duplication, noise removal, encoding, feature selection, normalization, and balancing. The most important steps of pre-processing are as follows:

- Remove space: Remove spaces from column names for simpler manipulation, and keep only the first row, and remove all others to eliminate duplicate rows from the dataset.
- Remove Null values: To improve the study's ACC, the wrong values of the attributes ct_flw_http_mthd, is_ftp_login, and attack_cat are removed.

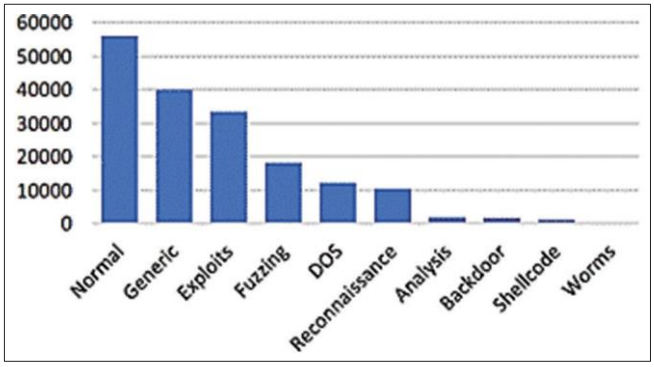

Figure 3: Number of records that represent normal traffic and malicious types of attacks in the UNSW-NB15 dataset

Label Encoding For Data Encoding

Label encoding converts categorical data into numbers, allowing ML algorithms to handle the categorical data. Each distinct category is given an integer in the range 0 to (n −1), n being the number of distinct classes. As an example, using 11 categories the number from 0 to 10 is used.

Feature Selection Using Chi-square

The term "feature selection" describes the steps used to determine which dataset characteristics are most relevant for building and training an ML model. To make AI models more compact and easier to work with, features are included. To find out which attributes are most essential to the target group, and compare the actual and expected frequencies of the categorical data using a statistical filter like Chi-square. Features with high Chi-square scores or low *P*-values are retained for improved model ACC.

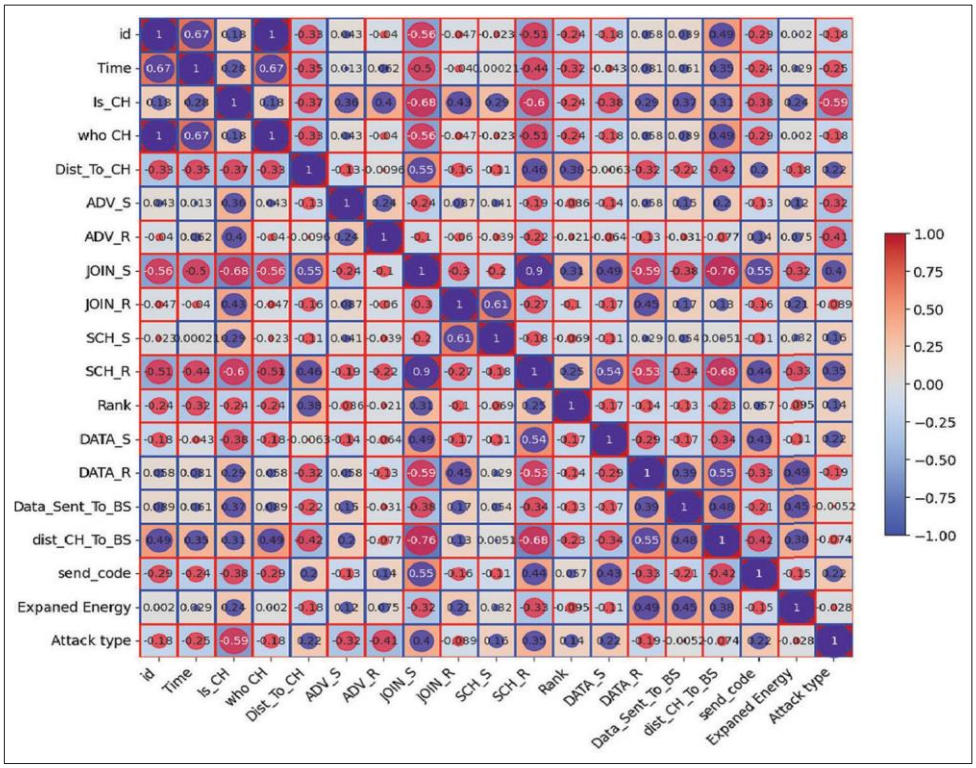

Figure 2: Sample of correlation matrix on UNSW-NB15

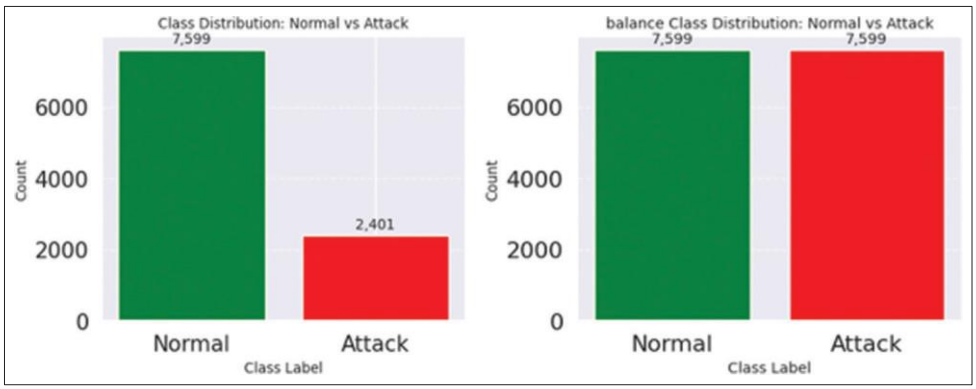

Figure 4: Before and after applying adasyn for class blanacing

Standard Scalar for Normalization

A normal distribution with a mean of 0 and a standard deviation of 1 was generated by standardizing the dataset using the StandardAero function. Here, observe that the standard deviation is divided by the mean of each observation and then subtracted once to achieve this transformation Equation (1).

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

The translated feature value (z), original descriptor values (x), mean (μ), and standard deviation (σ) are some of the variables found in this dataset.

Data Balancing using ADASYN

Data balancing strategies fix the problem of unequal class distributions and stop the model from happening. One adaptable oversampling approach that uses samples from minority classes is adaptive synthetic sampling, or ADASYN. To enhance classifier focus and decision boundaries, ADASYN generates synthetic data around harder-to-learn examples, prioritizes samples from minority classes in low-density regions, and estimates the density of those classes.

Figure 4 illustrates the impact of ADASYN on class balancing by comparing the original and resampled distributions of "Normal" and "Attack" instances. The dataset is initially unbalanced, which could lead to biased model performance. With 7,599 samples in each class, the "Attack" minority class is synthetically extended to have the same size as the majority class after ADASYN is applied. Anomaly detection tasks in particular benefit from this tweak, since it increases the model's robustness for classification and its capacity to learn from patterns that are under-represented.

Data Splitting

The efficacy of the dataset was assessed by dividing it into training and testing subsets. 80% of the dataset was allocated for model development and parameter refining, while the remaining 20% was reserved for performance evaluation and testing.

Proposed eXtreme Gradient Boosting (XGBoost)

XGBoost uses DT to generate predictions; it is an ensemble-based learning method. Regression issues can be tackled in a few different ways: One is by minimizing a loss function that measures the difference between actual and forecasted values. Two possible representations of the XGBoost regression model exist in mathematics Equation (2):

$$y = f(x) \quad (2)$$

Where y represents the predicted price of the property, x represents the input feature (i.e., square footage, the number of bedrooms, etc.), and $f(x)$ represents the XGBoost model that predicts y as a result of x. XGBoost creates a sequence of decision trees to compute the $f(x)$ by training them to reach a minimum MSE loss function. The model uses the combined predictions of several DT to arrive at a final forecast. A simplified version of the XGBoost regression model is Equation (3):

$$y = \sum (k = 1 \text{ to } K)\ fk(x)) \quad (3)$$

$fk(x)$ is the forecast of the kth decision tree, and K is the number of DT in the ensemble. Each tree is predicted as a weighted sum of the leaf values

of the tree, which are trained during the training process. The XGBoost model prediction of the input x is calculated by adding the prediction of all decision trees of the ensemble.

Evaluation Metrics

The suggested design was tested using several metrics to measure its performance. To summarize the results of the classification, a confusion matrix was created. The total number of correct and wrong predictions for each class is displayed in this matrix. Extracting useful metrics from this matrix included TP, FP, TN, and FN. Following the formulation in (4) to (7), these values were utilized to calculate crucial performance indicators, such as ACC, PRE, REC, and F1:

$$Accuracy = \frac{TP + TN}{TP + Fp + TN + FN} \quad (4)$$

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

$$F1-score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (7)$$

A model's ACC can be defined as the percentage of cases for which it made a correct prediction relative to all instances in the dataset. PRE is the proportion of positive events that the model accurately anticipated as a percentage of all positive occurrences forecasted. The REC ratio is the number of positive events predicted out of all the possible positive instances. The F1 aids in remembering information and accurately recalling it since it is a harmonic mean of the two. With the help of the ROC curve, show how the percentage of FP and the percentage of TP for various decision criteria relate to one another schematically.

RESULTS AND DISCUSSION

This section offers the performance of the suggested model and describes the experimental setup. The experiments were conducted on a robust PC with an Intel Core (TM) i3–1005G1 CPU clocking in at 1.20 GHz, 4 GB of RAM, with Python installed.

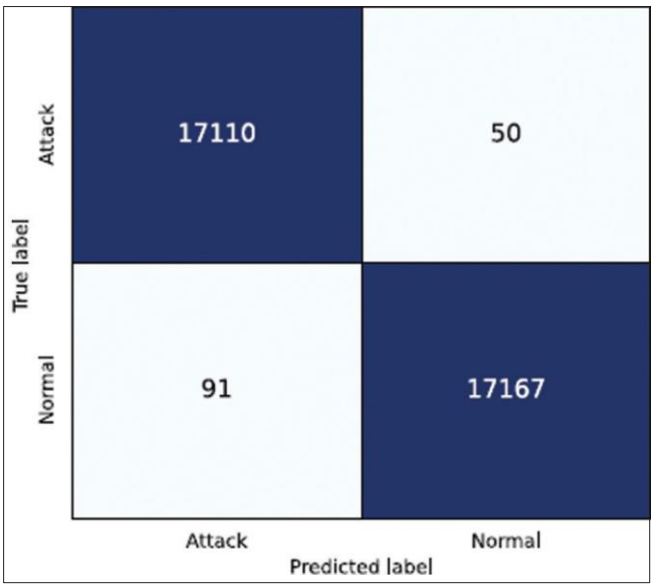

Figure 5: Confusion matrix for the XGBoost model

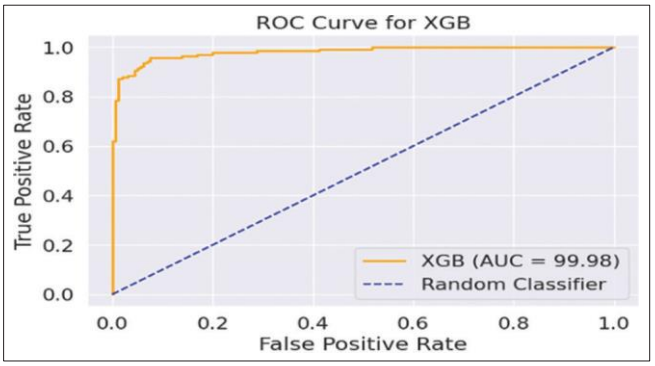

Figure 6: Receiver operating characteristic curve for XGBoost model

Table 2: Results of the proposed model for intrusion detection

Performance matrix	eXtreme gradient boosting (XGBoost)
Accuracy	99.59
Precision	99.8
Recall	99.5
F1-score	99.6

Table 3: Comparison of different mL and nl models for intrusion detection on UNSW-NB15 dataset

Model	Accuracy	Precision	Recall	F1-score
LR[19]	70.5	65.9	96.1	78.2
NB[20]	76.5	99	69	82
LSTM[21]	91.2	87.3	80.6	83.8
XGBoost	99.59	99.8	99.5	99.6

With 64 GB of RAM, the system can handle applications that require a lot of memory, and it comes with a substantial 40 GB of disc space for data storage. Table 2 shows the proposed model's performance summarized. With a PRE of 99.59%, the suggested XGBoost model successfully

categorized almost all network activities. The ACC of 99.5% in detecting real incursions and the PRE of 99.8% in minimizing false positives demonstrate the model's usefulness. An F1 of 99.6% shows that the model is very reliable and robust for effective IDS in complicated network environments, since it strikes a great balance between REC and PRE.

A confusion matrix showing the results of a classification model is shown in Figure 5. The results of a model that classifies incoming data as "Attack" or "Normal" are shown in this array. Here, the rows show the actual labels and the columns show the expected ones. Matrix data show that the model properly classified 17,110 occurrences as "Attack" and 17,167 as "Normal." False negatives totalling 50 and false positives totalling 91 occurred when it incorrectly classified 50 "Attack" instances as "Normal" and 91 "Normal" instances as "Attack," respectively. The model seems to be very accurate in general, with few misclassifications in comparison to the overall number of occurrences that were correctly detected.

Figure 6 shows how the TPR and the FPR intersect. Here can see the model's performance illustrated by the orange curve. The fact that the curve remains near the diagonal indicates that the model outperforms random guessing by a little margin. Despite this, the reported AUC of 99.98 seems at odds with the curve's visual trend; after all, a top-notch classifier would have a ROC curve that is much higher than the diagonal. This discrepancy may indicate either a plotting or evaluation error in the results.

Comparative Analysis

Table 3 provides a comparison of the proposed XGBoost model's accuracy with that of other current models to evaluate its usefulness. Among the traditional ML models, LR achieved moderate performance with an accuracy of 70.5%, and NB performed better in terms of PRE at 99%. The DL model, LSTM, showed a significant improvement with an accuracy of 91.2%, balanced PRE and recall, respectively. XGBoost demonstrated its exceptional capacity to accurately and reliably detect intrusions while minimizing false positives by reaching virtually flawless metrics, outperforming all other models by a considerable margin.

The proposed IDS model has several interesting strengths that make it more effective in cybersecurity. Utilizing adaptive sampling techniques, it can address the problem of class imbalance by reducing bias in favor of majority classes and improving the detection of unusual attack patterns. XGBoost is appropriate in complex and dynamic network environments in which the predictive accuracy, robustness, and scalability are required to be high. Its high performance in the major metrics proves that it has good classification with few false positives and negatives. These capabilities make them a more balanced and smart IDS that can assist in real-time monitoring of threats and decision-making in current digital infrastructures.

CONCLUSION AND FUTURE STUDY

IDS are an important part of safeguarding digital infrastructure against more advanced cyber-attacks. This paper presented a new AI-based platform that would increase the IDS through the reduction of class imbalance. With the UNSW-NB15 data set and eXtreme Gradient Boosting (XGBoost), the highest ACC of 99.59%, a PRE of 99.8%, a REC of 99.5% and an F1 of 99.6% were obtained, which proves the effectiveness of the method in detecting the frequent and rare attack patterns. Conventional methods such as Logistic Regression (70.5%) and the Naive Bayes (76.5%) demonstrated weak results, whereas DL based LSTM had a significant accuracy of 91.2%. XGBoost performs well in IDS, but the evaluation on one dataset restricts its usefulness in a generalized setting in most network environments. ROC curve inconsistencies suggest potential issues in metric interpretation, and the computational cost of ADASYN and XGBoost may challenge deployment on low-resource systems. Future work will explore multi-dataset validation, real-time and edge optimization, and integration of explainable AI to enhance scalability, transparency, and practical applicability in dynamic cybersecurity settings.

REFERENCES

1. Thomas J, Vedi KV, Gupta S. The effect and challenges of the internet of things (IOT) on the management of supply chains. Int J Res Anal Rev 2021;8:874-8.
2. Patel N. Sustainable smart cities: Leveraging Iot and data analytics for energy efficiency and urban development. J Emerg Technol Innov Res 2021;8: pp 313-319.
3. Abdur M, Habib S, Ali M, Ullah S. Security issues in the internet of things (IoT): A comprehensive study. Int J Adv Comput Sci Appl 2017;8:383 .
4. Nerella VM. Architecting secure, automated multi-cloud database platforms strategies for scalable compliance. Int J Intell Syst Appl Eng 2021;9:128-38.
5. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic minority over-sampling technique. J Artif Intell Res 2002;16:321-57.
6. Jamalipour A, Murali S. A taxonomy of machine-learning-based intrusion detection systems for the internet of things: A survey. IEEE Internet Things J 2022;9:9444-66.
7. Thangavel S, Sunkara KC, Srinivasan S. Software-defined networking (SDN) in cloud data centers: Optimizing traffic management for hyper-scale infrastructure. Int J Emerg Trends Comput Sci Inf Technol 2022;3:29-42.
8. Preuveneers D, Joosen W. Sharing machine learning models as indicators of compromise for cyber threat intelligence. J Cybersecur Priv 2021;1:140-63.
9. Sohal AS, Sandhu R, Sood SK, Chang V. A cybersecurity framework to identify malicious edge device in fog computing and cloud-of-things environments. Comput Secur 2018;74:340-54.
10. Venkata Naga SB, Sunkara KC, Thangavel S, Sundaram R. Secure and scalable data replication strategies in distributed storage networks. Int J AI BigData Comput Manag Stud 2021;2:18-27.
11. Mohammadi S, Mirvaziri H, Ghazizadeh-Ahsaee M, Karimipour H. Cyber intrusion detection by combined feature selection algorithm. J Inf Secur Appl 2019;44:80-8.
12. Sarker IH, Kayes AS, Badsha S, Alqahtani H, Watters P, Ng A. Cybersecurity data science: An overview from machine learning perspective. J Big Data 2020;7:41.
13. Kabir MH, Rajib MS, Rahman AS, Rahman MM, Dey SK. Network Intrusion Detection using UNSW-NB15 Dataset: Stacking Machine Learning Based Approach. In: 2022 International Conference on Advancement in Electrical and Electronic Engineering, ICAEEE 2022; 2022.
14. Gupta D, Saxena AK. Using Machine Learning for Network Intrusion Detection. In: 2022 Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing and Communication Engineering (ICATIECE); 2022. p. 1-5.
15. Umamaheshwari S, Kumar SA, Sasikala S. Towards Building Robust Intrusion Detection System in Wireless Sensor Networks using Machine Learning and Feature Selection. In: 2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation, ICAECA 2021; 2021.
16. Das S, Ashrafuzzaman M, Sheldon FT, Shiva S. Network Intrusion Detection using Natural Language Processing and Ensemble Machine Learning. In: 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020; 2020.
17. Srivastava A, Agarwal A, Kaur G. Novel Machine Learning Technique for Intrusion Detection in Recent Network-based Attacks. In: 2019 4th International Conference on Information Systems and Computer Networks (ISCON), IEEE; 2019. p. 524-8.
18. Singh K, Mathai KJ. Performance Comparison of Intrusion Detection System Between Deep Belief Network (DBN)Algorithm and State Preserving Extreme Learning Machine (SPELM) Algorithm. In: Proceedings of 2019 3rd IEEE International Conference on Electrical, Computer and Communication Technologies, ICECCT 2019; 2019.
19. Shushlevska M, Efnusheva D, Jakimovski G, Todorov Z. Anomaly Detection with Various Machine Learning Classification Techniques over UNSW-NB15 Dataset. In: Proceedings of International Conference on Applied Innovation in IT; 2022.
20. Kocher G, Kumar G. Analysis of machine learning algorithms with feature selection for intrusion detection using UNSW-NB15 dataset. Int J Netw Secur Its Appl 2021;13:21-31.
21. Van Duong L. Network anomaly detection technique based on LSTM network and UNSW-NB15 dataset. Int J Adv Trends Comput Sci Eng 2020;9:6527-32.