AN INTELLIGENT NETWORK INTRUSION DETECTION SYSTEM USING MACHINE LEARNING TOWARDS ENHANCING CYBERSECURITY

Ayesha Siddiqua, Khair Unnisa Naaz, Sabiha Mahveen, K. Mohammadi Jabeen, Assistant Professor, Dept of CS&AI, MJCET, O.U, Hyderabad, T.S, India**.** ayesha.siddiqua@mjcollege.ac.in
Zeba Ruqshanh, Assistant Professor, Dept of CSE, CMR Engineering College, JNTUH, T.S, India.

ABSTRACT

The security of information systems and networks is paramount due to the increasing number of security attacks. Adversaries use novel techniques to break into systems, making it crucial to enhance security to safeguard data and communications. Artificial intelligence (AI) agents now enable learning-based approaches to provide an extra layer of protection to information systems and networks, with machine learning models being beneficial for detecting intrusions as they occur. However, there is a need to explore the efficiency of machine learning models to enhance cybersecurity. This paper proposes a machine learning-based framework for realizing an intrusion detection system. We used multiple machine learning models as part of the framework and evaluated them for intrusion detection. We introduce an algorithm called Learning-Based Intrusion Detection (LBID), efficiently performing multi-class classification. An empirical study with a benchmark dataset known as CICIDS 2019 revealed that our proposed framework is efficient in intrusion detection, with model Voting exhibiting the highest performance at 96.61% accuracy. Therefore, our framework can be incorporated into real-time applications to safeguard against various intrusions.

Keywords - Intrusion Detection System, Artificial Intelligence, Machine Learning, Cyber Security, Learning Based Intrusion Detection

1. INTRODUCTION

The widespread use of the Internet and various networks has made cyberspace a prime target for adversaries seeking to exploit its benefits. With the increasing use of applications, networks, and emerging technologies like the Internet of Things, there has been a rise in security threats to information systems and networks [1]. While it may seem implausible, artificial intelligence can address real-world problems, and there have been efforts to use machine learning models for intrusion detection systems to secure cyberspace [2]. These intrusion detection systems help protect information systems, but security is not a one-time solution and must be continually improved to detect new intrusions [3], [4].

Areview of the literature indicates that artificial intelligence has made an impact on solving security problems. Various machine learning and deep learning techniques are being used to safeguard information systems and communications. Despite these efforts, there has been an increase in multiple types of cyber-attacks globally. This underscores the need for further improvements in cybersecurity. To address this, our paper proposes a machine learning-based framework for an intrusion detection system. We used multiple machine learning models as part of the framework and evaluated them for intrusion detection. We introduced an algorithm called Learning-Based Intrusion Detection (LBID), efficiently performing multi-class classification. An empirical study using a benchmark dataset called CICIDS 2019 showed that our framework is efficient in intrusion detection, with the Voting model exhibiting the highest performance at 96.61% accuracy. This suggests that our framework can be incorporated into real-time applications to safeguard against various intrusions. The remainder of the paper is structured as follows: Section 2 reviews literature on various existing methods based on machine learning techniques. Section 3 presents the proposed intrusion detection system that exploits a learning-based approach to detect intrusions efficiently. Section 4 presents experimental results with a benchmark dataset, showing that the proposed framework can detect various intrusions with multi-class classification. Section 5 concludes our research work and provides directions for the future scope of the research.

2. RELATED WORK

This section reviews the literature on various existing methods used for intrusion detection. Shaukat *et al.* [1] discussed how the widespread use of applications and the internet has enormously increasedcyber threats. The author argued that machine learning should be used in cybersecurity despite its reliability and evasion problems. Sarker [2] examined the potential of machine learning for automated and intelligent data processing in the context of cybersecurity. Despite all the possible challenges, proactive cyber protection will require further research to address problems like algorithm performance and data quality problems. Alqahtani *et al.* [3] compared the effectiveness of many classifiers on different attack datasets to explore machine learning techniques for developing an intrusion detection system (IDS) in cybersecurity. Future studies will try to automate security services, increase dataset variety, and strike a balance between accuracy challenges. Anthi*et al.* [4] explored how intrusion detection systems (IDSs) based on machine learning are used in industrial control systems and identified the shortcomings of adversarial machine learning (AML). It recommends using adversarial training to strengthen resistance by putting classifiers like Random Forest and J48 to the test against adversarial attacks. Research on further offensive and defensive tactics will be conducted. Zhang *et al.* [5] introduced a novel brute-force attack method called BFAM to evaluate how resilient machine learning classifiers are to hostile cybersecurity situations. Compared to GAN-based methods, it operates more effectively and efficiently. More studies aim to increase the applicability of BFAM to multiple contexts and classifiers.

Novo *et al.* [6] evaluate machine learning techniques in cybersecurity, focusing on neural networks for intrusion detection using the UNSW-NB15 dataset. Subsequent investigations will concentrate on enhancing performance metrics beyond precision and expanding to larger datasets. Sarker [7] offered "CyberLearning," a machine learning approach to cybersecurity that uses binary and multi-class classification algorithms. It evaluates ten popular classifiers and neural networks using datasets like UNSW-NB15 and NSL-KDD. Research on IoT data will be conducted in the future, to enhance system security through the application of sophisticated learning algorithms. Shaukat *et al.* [8] evaluated the effectiveness of deep belief networks, decision trees, and support vector machines as machine learning techniques for cyber threat identification. Future research aims to increase detection accuracy by utilizing a range of datasets despite present challenges with dataset diversity and model restrictions. Jiang and Atif [9] presented a cognitive cybersecurity model that assesses vulnerabilities by integrating a variety of inputs. Future work in machine learning-based cybersecurity will concentrate on refining ensemble techniques like stacking, expanding the range of available data sources, and addressing problems with resource allocation and class disparity. Panda *et al.* [10] recommended a feature-engineering approach that uses machine learning and deep learning to identify IoT botnet attacks with high accuracy and little processing expense. To detect cyberattacks in real-time, future research will examine a range of datasets and integrate blockchain, edge computing, cloud computing, machine learning, and deep learning. Elsisi*et al.* [11] proposed an Internet of Things architecture that monitors gas-insulated switchgear (GIS) in real-time and uses machine learning to detect vulnerabilities and threats. Future studies will concentrate on expanding the scope of application and fortifying cybersecurity protections in power systems. Bland *et al.* [12] used Petri nets with players, strategies, and pricing to simulate attacks like spear phishing and XSS. Further work will employ these models in real-world computing environments and use accurate system data to optimize parameters. Verkerken *et al.* [13] assessment of unsupervised intrusion detection techniques on a modern dataset demonstrates strong generalization and performance. Subsequent studies aim to verify the applicability of these findings in different real-world scenarios. Alabadi and Celik [14] outlined convolutional neural networks (CNNs) in anomaly identification. It presents a unified, all-inclusive framework for analysis and categorizing earlier research according to the incoming data source. Future development will primarily concentrate on improving high-accuracy, real-time models. Musa *et al.* [15] investigated machine learning classifiers for use in single, hybrid, and ensemble intrusion detection systems (IDS), evaluated on seven distinct datasets.

The range of datasets and computational complexity present challenges for future research aiming to increase the accuracy and efficiency of these systems.

Carley [16] examined and introduced the idea of social cybersecurity, which targets cybercrimes that exploit digital and social interactions. Owing to data limitations and social interactions' dynamic character, it emphasizes the need for socially aware computational social science and cautions against depending too much on AI solutions. Future research should prioritize the use of human-centric techniques. Verkerken *et al*. [17] evaluated anomaly-based unsupervised machine learning approaches for network intrusion detection systems. It highlights the challenges associated with cross-dataset generalization by showcasing significant drops in accuracy and AUROC scores between comparable datasets. Future research should focus on improving model generality to enhance practical application. Tran *et al*. [18] recommended an Internet of Things architecture that uses machine learning to detect and prevent cyberattacks and monitor issues with induction motors. CONTACT Elements demonstrates a high flaw detection accuracy of 99.03% with Random Forest, which enhances industrial efficiency and decision-making. Future research should extend this approach to be used in various machine applications. Dwivedi *et al*. [19] employed machine learning (ML) techniques such as RF, SVMs, Keras DL, and XGBoost along with modern datasets (UNSW-NB15, Bot-IoT, CSE-CIC-IDS2018) to evaluate IDS performance. Further research will center on IDS evasion tactics and enhance datasets for more realistic implementation. Bagaa*et al*. [20] described an ML-based solution that combines SDN and NFV to address IoT security issues. Future efforts will focus on standardizing framework interfaces, adapting to evolving IoT threats, optimizing dynamically to respond to ML algorithms, and balancing the consequences of security and performance.

Hossain and Islam [21] demonstrate the limitations of traditional intrusion detection systems (IDS) in their capacity to detect unexpected threats and offer a novel intrusion detection method based on ensemble-based machine learning. The recommended Random Forest method outperforms other methods on a range of datasets, suggesting that network security may be enhanced with high accuracy and efficacy. Stevens [22] explored how AI is utilized to combat cyber threats and how cybersecurity evolves across different global stakeholders. Though it raises concerns about accountability and ethical ramifications, it highlights AI's contribution to increasing operational efficiency. Future negotiations over AI's impact on cybersecurity for the military and intelligence community will need to address significant issues around the proper usage and regulation of the technology. Masser *et al*. [23] discussed arbitrarily selected algorithms and outdated datasets in the corpus of research on anomaly-based intrusion detection systems (AIDS). After evaluating 31 ML-AIDS models using a range of KPIs and datasets, it concludes that the k-NN, DT, and NB models perform the best. However, that also draws attention to problems with multi-classification and the identification of novel attacks. More studies should look into feature selection and advance CNN-AIDS development. Mbona and Eloff [24] examined attempts at zero-day network intrusions and emphasized Benford's law to determine crucial network properties. One-class support vector machines (SVM) (MCC 74%, F1 score 85%) are a practical semi-supervised machine learning approach. Future research should integrate various feature selection methods with machine learning classifiers to get better detection performance. The literature showed a need to improve intrusion detection systems with efficient models and empirical evolution.

3. PROPOSED SYSTEM

The following section outlines the methodology for developing an intrusion detection system using machine learning models. The framework employs a supervised learning approach with labeled data to train the models, allowing them to gather knowledge from newly available training samples continuously. Over time, the machine learning models acquire the knowledge to detect various intrusions. As it is a supervised learning approach, where models are trained using labeled data to gainan understanding of intrusion detection, it is crucial to retrain the models when new data becomes available. Consequently, the proposed framework is designed to create an intrusion detection system that continually enhances knowledge and effectively detects intrusions.

The framework for the intrusion detection system is detailed in Figure 1. The provided dataset is preprocessedto handle null and categorical values efficiently, ensuring proper support for the training of machine learning models. The framework also includes a feature selection approach that utilizes entropy and gain metrics to choose features contributing to class label selection. The measures used for computing feature importance assist in filtering features and improving training quality through a filter approach. Feature selection is crucial for ensuring high-quality inputs for the training process, as the performance of machine learning models can deteriorate without quality inputs. As a result, we employed a feature selection process to enhance the training process for the proposed intrusion detection system.

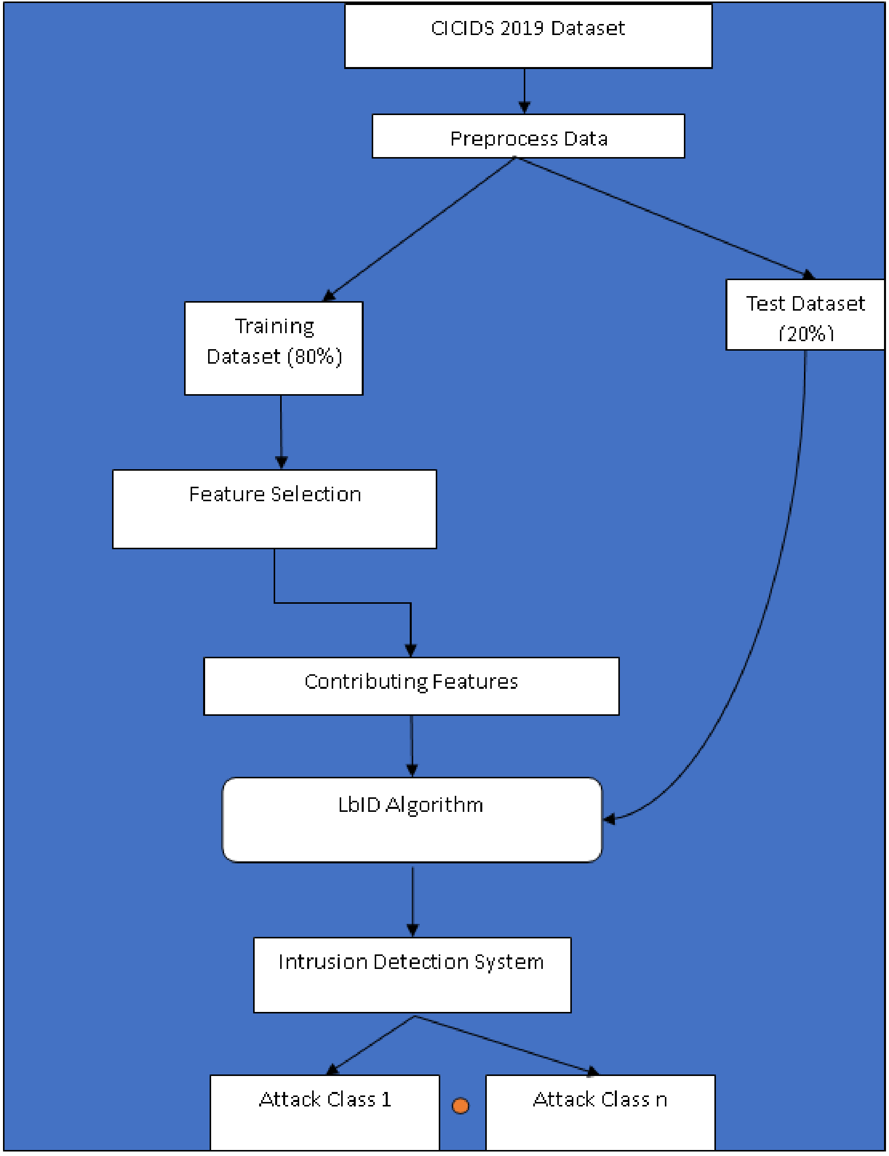


Figure 1: The proposed intrusion detection system based on machine learning

Figure 1 outlines using the CICIDS 2019 Dataset to develop an intrusion detection system. The process begins with the data preprocessing, after which the dataset is split into a training dataset (80%) and a test dataset (20%). Feature selection is then performed on the training dataset to identify the contributing features. These features are used in the LBID algorithm, a core component of the intrusion detection system. The system then classifies the data into attack classes, ranging from Attack Class 1 to Attack Class N.

Attack Category	**Class Label**	**Description**
Reconnaissance	0	This attack involves gathering information about a target system to identify potential vulnerabilities. Techniques include scanning networks and probing for open ports. The goal is to collect intelligence that will assist in planning a more impactful attack.
Generic	1	This category covers attacks that do not fit neatly into specific classifications. It may include a variety of uncommon or less well-defined attack methods that can be used in different contexts.
Shellcode	2	In the context of exploits, Shellcode is a small piece of code that enables an attacker to take control of the compromised system by running arbitrary commands. It is commonly utilized in buffer overflow attacks.
Fuzzers	3	Fuzzing, a technique for uncovering vulnerabilities, involves sending random or malformed inputs to a program. The goal is to trigger unexpected behaviors or crashes that indicate potential security flaws.
Analysis	4	Analysis: This includes attacks focused on studying and understanding the system's weaknesses, often through detailed examination of vulnerabilities and potential exploits. It is used to refine attack strategies and exploit potential weaknesses more effectively.
Backdoor	5	A backdoor attack involves installing software that allows unauthorized access to a system by bypassing standard authentication methods. This type of attack enables attackers to maintain control even after initial vulnerabilities are patched.
Normal	6	This category refers to legitimate, non-malicious activities that do not pose any threat. It includes standard operations and regular traffic in a system or network.
Worms	7	Worms are self-replicating malware that spread across networks without user interaction, exploiting vulnerabilities to infect other systems, often causing widespread damage and consuming network resources.
DoS	8	DoS (Denial of Service) attacks aim to overwhelm a system or network with traffic, rendering it unavailable to legitimate users. The attack typically floods the target with excessive requests, leading to performance degradation or complete service disruption.
Exploits	9	Exploits are methods or tools used to exploit vulnerabilities in software or hardware. By leveraging these weaknesses, attackers can gain unauthorized access, escalate privileges, or perform malicious actions on a system.

Table 1: Different attack categories corresponding class labels and description of attacks

As shown in Table 1, different kinds of attacks are possible. For this reason, the deep learning models used in the proposed intrusion detection system are trained with various types of training and normal samples. The proposed entrance detection system supports multi-class classification to detect all these attacks.

Algorithm: Learning-Based Intrusion Detection (LBID)
Input: CICIDS 2019 dataset D, ML models M (Decision Tree, Random Forest, ExtraTree, and Voting)
Output: Intrusion detection results R, performance statistics P

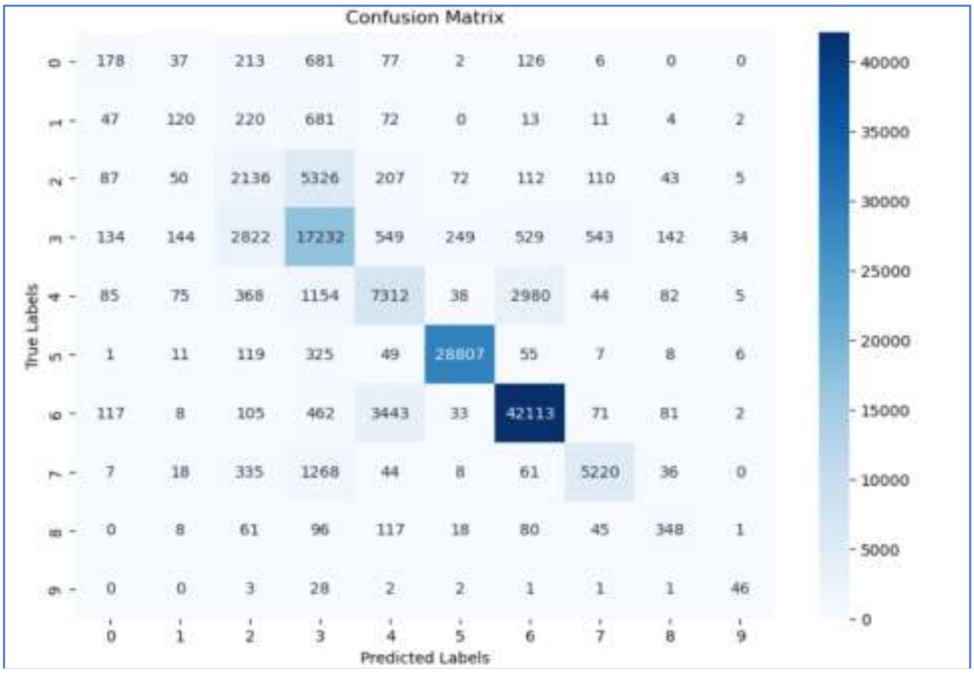
```
Begin
D'←Preprocess(D)
(T1, T2)←SplitData(D')
F1←FeatureSelection(T1)
F2←FeatureSelection(T2)
For each model m in M
m'←TrainModel(M, F1, T1)
Persist model m'
End For
For each model m'in M
Load m'
R←DetectIntrusions(m', T2, F2)
  P←EvlautePerformance(R, ground truth)
  Print R
  Print P
End For
End
```

Algorithm 1: Learning-Based Intrusion Detection (LBID)

Algorithm 1 is designed to detect intrusions and assess model performance. The dataset D undergoes preprocessing to create D', which is then divided into two parts, T1 and T2. The feature selection process is carried out on both parts, resulting in F1 and F2. Each model m in the set M, which includes Decision Tree, Random Forest, ExtraTree, and Voting models, is trained using the preprocessed and feature-selected data T1 and F1. The trained models are then saved for future use. In the subsequent phase, each saved model m' is loaded and utilized for intrusion detection on the second set of data, T2, along with its corresponding feature set, F2. The intrusion detection results R are then assessed against the ground truth to generate performance statistics P. Both R and P are output for analysis. This algorithm follows a structured approach to training and evaluating machine learning models for intrusion detection, ensuring that the models are trained on one dataset and tested on another for validation. Using multiple models allows for performance comparison, and the final output includes intrusion detection results and performance statistics for each model.

4. EXPERIMENTAL RESULTS

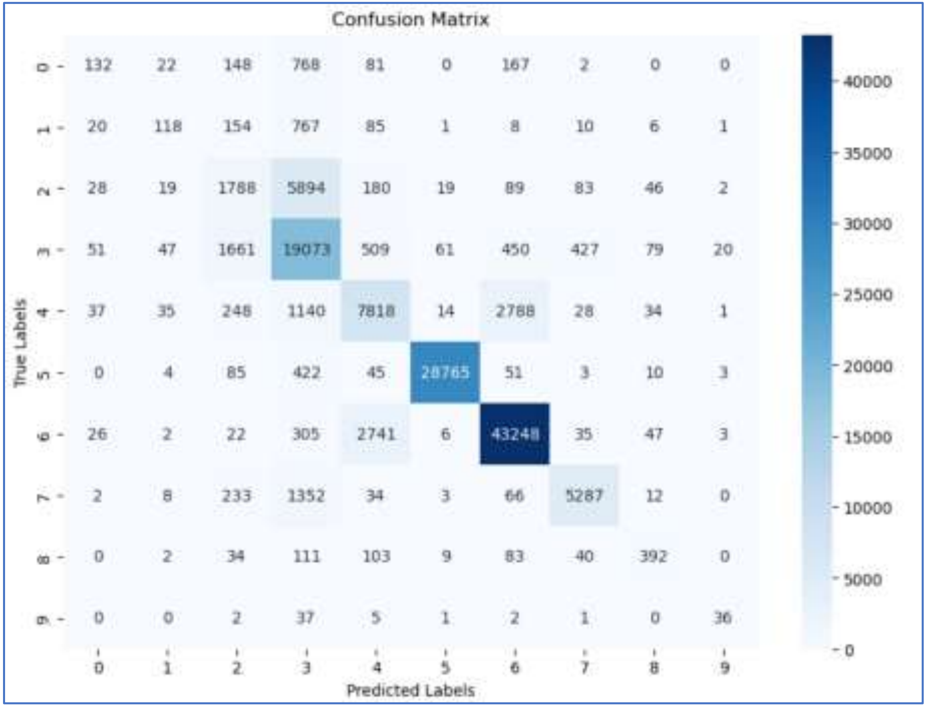
Below are the experimental results of the intrusion detection system, which uses multiple machine learning models. All the models in the proposed system are assessed to determine their performance in the intrusion detection process. Each model is tested with the samples, and the trained model effectively handles various intrusions. In summary, each model is evaluated using the test data for multiclass classification to detect different types of intrusions.



	0	1	2	3	4	5	6	7	8	9
0	178	37	213	681	77	2	126	6	0	0
1	47	120	220	681	72	0	13	11	4	2
2	87	50	2136	5326	207	72	112	110	43	5
3	134	144	2822	17232	549	249	529	543	142	34
4	85	75	368	1154	7312	38	2980	44	82	5
5	1	11	119	325	49	28807	55	7	8	6
6	117	8	105	462	3443	33	42113	71	81	2
7	7	18	335	1268	44	8	61	5220	36	0
8	0	8	61	96	117	18	80	45	348	1
9	0	0	3	28	2	2	1	1	1	46

Figure 2: Confusion matrix of DT model

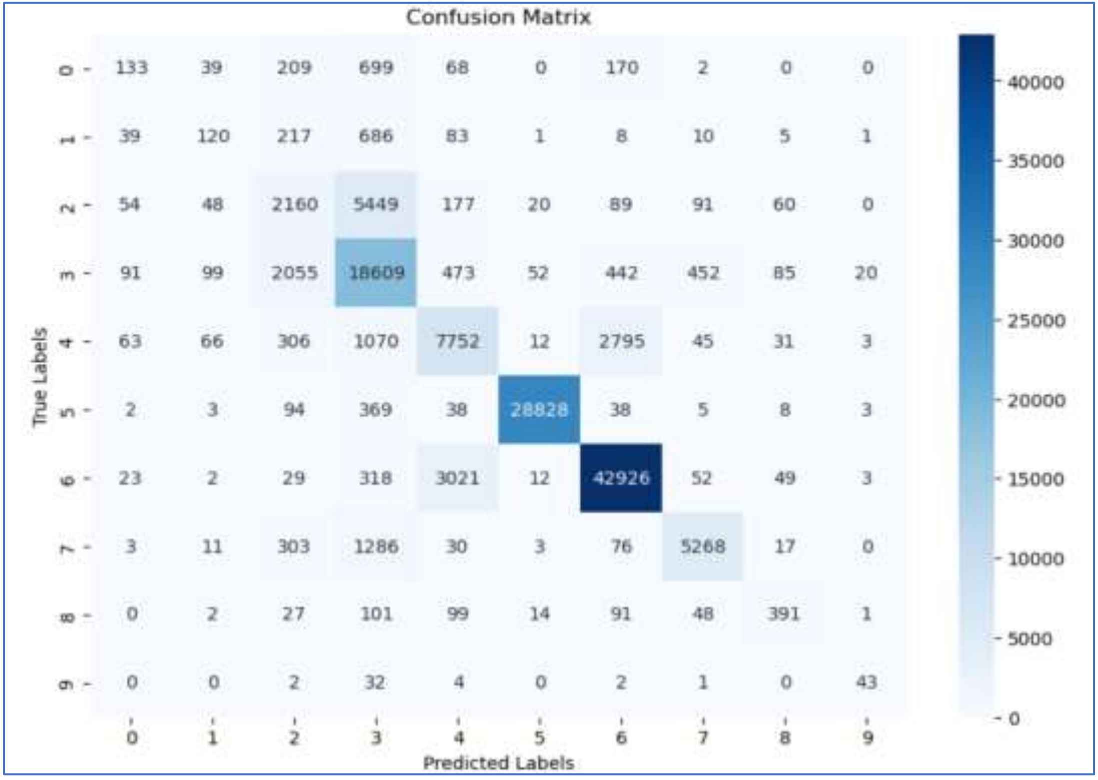
Figure 2 shows a confusion matrix for a multi-class classification problem, where the rows represent the actual labels and the columns represent the predicted labels. Each cell in the matrix indicates the number of instances where the corresponding accurate label (on the y-axis) was expected as the label on the x-axis.The diagonal cells (from top-left to bottom-right) represent the number of correctly predicted instances for each class, while the off-diagonal cells represent misclassifications. The color intensity in the matrix indicates the frequency of predictions, with darker shades indicating higher counts.For example, the matrix shows that the actual label '6' has the highest correct predictions with 42,113 instances, indicating that the model performs well in this particular class. However, there are also significant misclassifications, as seen by the non-zero values off the diagonal. This confusion matrix helps understand the classification model's performance, showing where it excels and struggles.



	0	1	2	3	4	5	6	7	8	9
0	132	22	148	768	81	0	167	2	0	0
1	20	118	154	767	85	1	8	10	6	1
2	28	19	1788	5894	180	19	89	83	46	2
3	51	47	1661	19073	509	61	450	427	79	20
4	37	35	248	1140	7818	14	2788	28	34	1
5	0	4	85	422	45	28765	51	3	10	3
6	26	2	22	305	2741	6	43248	35	47	3
7	2	8	233	1352	34	3	66	5287	12	0
8	0	2	34	111	103	9	83	40	392	0
9	0	0	2	37	5	1	2	1	0	36

Figure 3: Confusion matrix of the RF model

Figure 3 illustrates a confusion matrix for a Random Forest (RF) model applied to a multi-class classification task. In this matrix, the rows correspond to the actual labels, and the columns correspond to the predicted labels. The diagonal elements of the matrix indicate the number of instances where the predicted labels match the actual labels, representing correct classifications. The darker shades along this diagonal, particularly for the class labeled '6' with 43,248 correct predictions, show that the model accurately identifies this class. However, the matrix also reveals instances of misclassification, as indicated by the off-diagonal elements where the accurate labels differ from the predicted labels. These misclassifications are spread across various classes, with some courses like '3', having a notable number of incorrect predictions. The color gradient in the matrix visually emphasizes the distribution of correct and incorrect predictions, helping to assess the model's performance. Overall, this confusion matrix provides insight into the Random Forest model's strengths and areas for improvement in predicting different classes.



	0	1	2	3	4	5	6	7	8	9
0	133	39	209	699	68	0	170	2	0	0
1	39	120	217	686	83	1	8	10	5	1
2	54	48	2160	5449	177	20	89	91	60	0
3	91	99	2055	18609	473	52	442	452	85	20
4	63	66	306	1070	7752	12	2795	45	31	3
5	2	3	94	369	38	28828	38	5	8	3
6	23	2	29	318	3021	12	42926	52	49	3
7	3	11	303	1286	30	3	76	5268	17	0
8	0	2	27	101	99	14	91	48	391	1
9	0	0	2	32	4	0	2	1	0	43

Figure 4: Confusion matrix of Extra Trees model

Figure 4 shows a confusion matrix for an Extra Trees model applied to a multi-class classification task. The rows in the matrix represent the actual labels, while the columns represent the predicted labels. The diagonal elements indicate the number of correctly classified instances for each class, with the darkest shade on the diagonal corresponding to class '6', which has 42,926 correct predictions, suggesting strong performance in this category. The off-diagonal elements represent misclassifications, where the predicted label does not match the actual label. These misclassifications are spread across various classes, indicating areas where the model struggles to predict correctly. The color intensity helps visualize the distribution of correct and incorrect predictions, with darker shades indicating higher counts. Overall, this confusion matrix is a valuable tool for evaluating the performance of the Extra Trees model, showing its accuracy in certain classes while highlighting areas for potential improvement in others.

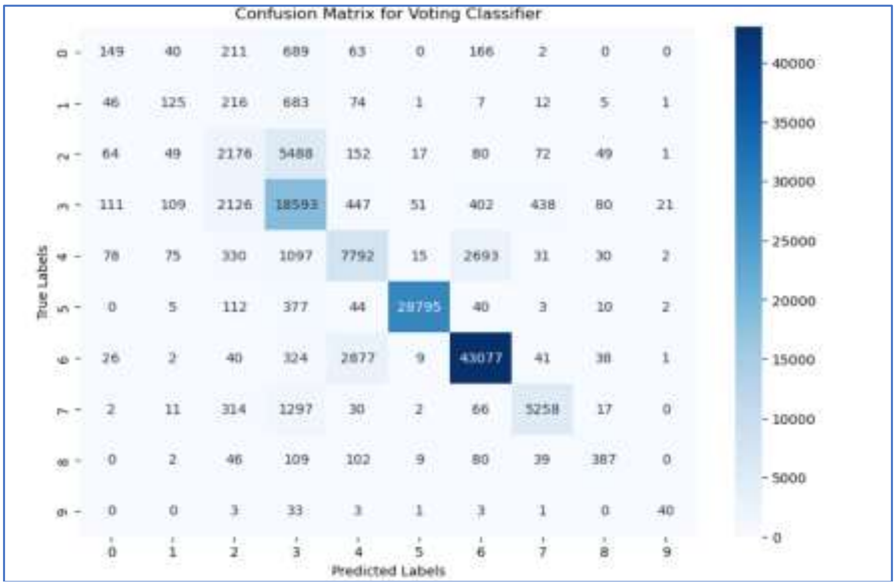


Figure 5: Confusion matrix of Voting model

Figure 5 demonstrates the performance of a voting classifier across multiple classes. Each cell represents the count of accurate labels (rows) against the predicted labels (columns), with diagonal elements indicating correctly predicted instances and off-diagonal elements representing misclassifications. Class 6 has the highest number of accurate predictions with 43,077 instances, followed by Class 5 with 28,795 correct predictions. Misclassifications are dispersed across various classes, with Classes 3 and 4 showing significant misclassification counts. The color intensity of the cells, with darker shades indicating higher values, serves as a visual aid to quickly identify the classifier's strengths and weaknesses. For example, the darker diagonal cells demonstrate strong performance in accurately identifying Classes 5 and 6, while the lighter shades of the diagonal suggest areas where the classifier's performance could be improved.

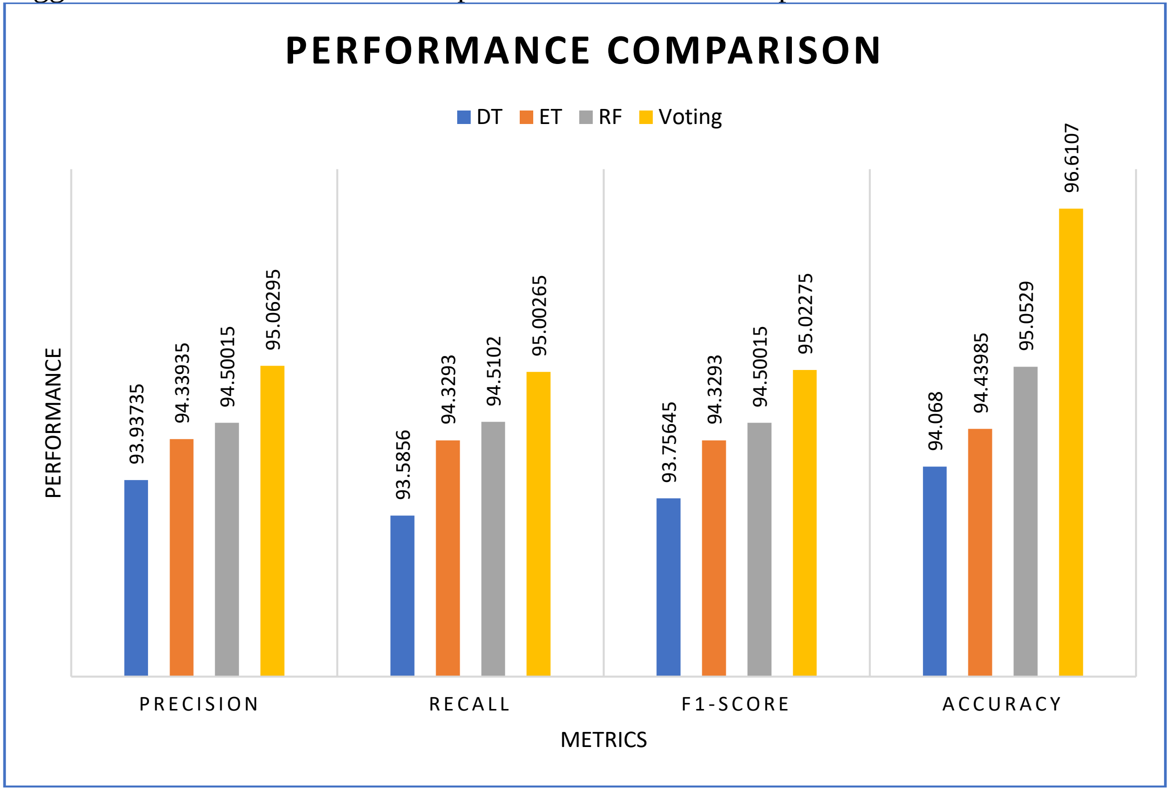


Figure 6: Performance comparison among intrusion detection models

Figure 6 provides a comparative analysis of different classifiers—Decision Tree (DT), Extra Trees (ET), Random Forest (RF), and a Voting Classifier—across various performance metrics: precision, recall, F1-score, and accuracy. In terms of precision, the Voting Classifier outperforms the others, achieving a precision of 95.66295, followed by the Random Forest with 94.50015, Extra Trees with 94.3935, and Decision Tree with 93.9375. For recall, the Voting Classifier again leads with 95.00265, with Random Forest (94.50015), Extra Trees (94.2393), and Decision Tree (93.5856) following. The F1-score, which balances precision and recall, shows the Voting Classifier at the top with 95.02275, trailed by Random Forest (94.50015), Extra Trees (94.3293), and Decision Tree (93.75645). Lastly, in terms of accuracy, the Voting Classifier achieves the highest at 96.6107, followed by Random Forest (95.0529), Extra Trees (94.43985), and Decision Tree (94.068).

5. CONCLUSION AND FUTURE WORK

Our paperproposed a framework based on machine learning for building an intrusion detection system. We used multiple machine-learning models within this framework and assessed their effectiveness in detecting intrusions. The proposed framework harnesses various machine learning models and employs feature selection to ensure that the models only learn from the most relevant features for class label selection. Additionally, the framework includes mechanisms for dealing with the given dataset in a supervised learning approach, ensuring that machine learning models are adequately trained while avoiding overfitting issues. We introduce an algorithm called Learning-Based Intrusion Detection (LBID), which proficiently performs multi-class classification. Through an empirical study using the CICIDS 2019 dataset, we demonstrated that our proposed framework is highly efficient in detecting intrusions. In our research, the Voting model exhibited the highest performance with 96.61% accuracy. As a result, our framework can be implemented in real-time applications to protect against various intrusions. We have identified several future directions for improving intrusion detection systems, including enhancing the proposed framework through hybrid feature engineering methodologies and hyperparameter tuning to improve detection performance. Furthermore, we aim to incorporate deep neural networks to enhance the quality of training, ultimately leading to more efficient performance in intrusion detection. Another future direction for our research is to develop an intrusion detection system that operates without requiring labeled data for training purposes.

REFERENCES

[1] Shaukat, K., Luo, S., Varadharajan, V., Hameed, I. A., & Xu, M. (2020). A Survey on Machine Learning Techniques for Cyber Security in the Last Decade. IEEE Access, 8, pp.222310–222354. doi:10.1109/access.2020.3041951

[2] Sarker, I.H., (2023). Machine learning for intelligent data analysis and automation in cybersecurity: current and future prospects. Annals of Data Science, 10(6), pp.1473-1498. https://doi.org/10.1007/s40745-022-00444-2

[3] Alqahtani, H., Sarker, I.H., Kalim, A., Minhaz Hossain, S.M., Ikhlaq, S. and Hossain, S., (2020). Cyber intrusion detection using machine learning classification techniques. In Computing Science, Communication and Security: First International Conference, COMS2 2020, Gujarat, India, March 26–27, 2020, Revised Selected Papers 1 pp. 121-131. Springer Singapore.

[4]Anthi, E., Williams, L., Rhode, M., Burnap, P., & Wedgbury, A. (2021). Adversarial attacks on machine learning cybersecurity defences in Industrial Control Systems. Journal of Information Security and Applications, 58, pp.1-9. doi:10.1016/j.jisa.2020.102717.

[5] Zhang, S., Xie, X., & Xu, Y. (2020). A Brute-force Black-box Method to Attack Machine Learning-Based Systems in Cybersecurity. IEEE Access, pp.1–14. doi:10.1109/access.2020.3008433

[6] Larriva-Novo, X. A., Vega-Barbas, M., Villagra, V. A., & Sanz Rodrigo, M. (2020). Evaluation of Cybersecurity Data Set Characteristics for Their Applicability to Neural Networks Algorithms Detecting Cybersecurity Anomalies. IEEE Access, 8, pp.9005–9014. doi:10.1109/access.2019.2963407

[7] Sarker, I. H. (2021). CyberLearning: Effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks. Internet of Things, 14, pp.1-18. doi:10.1016/j.iot.2021.100393

[8] Shaukat, K., Luo, S., Chen, S., & Liu, D. (2020). Cyber Threat Detection Using Machine Learning Techniques: A Performance Evaluation Perspective. 2020 International Conference on Cyber Warfare and Security (ICCWS). pp.106. doi:10.1109/iccws48432.2020.9292388

[9] Jiang, Y., & Atif, Y. (2021). A selective ensemble model for cognitive cybersecurity analysis. Journal of Network and Computer Applications, 193, pp.1-16. doi:10.1016/j.jnca.2021.103210

[10] Panda, M., Mousa, A. A. A., &Hassanien, A. E. (2021). Developing an Efficient Feature Engineering and Machine Learning Model for Detecting IoT-Botnet Cyber Attacks. IEEE Access, 9, pp.91038–91052. doi:10.1109/access.2021.3092054

[11]Elsisi, M., Tran, M.-Q., Mahmoud, K., Mansour, D.-E. A., Lehtonen, M., & Darwish, M. M. F. (2021). Towards Secured Online Monitoring for Digitalized GIS Against Cyber-Attacks Based on IoT and Machine Learning. IEEE Access, 9. pp.78415–78427. doi:10.1109/access.2021.3083499

[12] Bland, J. A., Petty, M. D., Whitaker, T. S., Maxwell, K. P., & Cantrell, W. A. (2020). Machine Learning Cyberattack and Defense Strategies. Computers & Security, 92, pp.1-23. doi:10.1016/j.cose.2020.101738

[13] Verkerken, M., D'hooge, L., Wauters, T., Volckaert, B., & De Turck, F. (2020). Unsupervised Machine Learning Techniques for Network Intrusion Detection on Modern Data. 2020 4th Cyber Security in Networking Conference (CSNet). pp.1-8. doi:10.1109/csnet50428.2020.9265461

[14]Alabadi, M., & Celik, Y. (2020). Anomaly Detection for Cyber-Security Based on Convolution Neural Network : A survey. 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA). pp.1-14. doi:10.1109/hora49412.2020.9152899

[15] Musa, U. S., Chhabra, M., Ali, A., & Kaur, M. (2020). Intrusion Detection System using Machine Learning Techniques: A Review. 2020 International Conference on Smart Electronics and Communication (ICOSEC). pp.1-7. doi:10.1109/icosec49089.2020.9215333

[16] Carley, K. M. (2020). Social cybersecurity: an emerging science. Computational and Mathematical Organization Theory, 26(4), pp.365–381. doi:10.1007/s10588-020-09322-9

[17] Verkerken, M., D'hooge, L., Wauters, T., Volckaert, B. and De Turck, F., (2022). Towards model generalization for intrusion detection: Unsupervised machine learning techniques. Journal of Network and Systems Management, 30, pp.1-25. https://doi.org/10.1007/s10922-021-09615-7

[18] Tran, M.Q., Elsisi, M., Mahmoud, K., Liu, M.K., Lehtonen, M. and Darwish, M.M., (20210. Experimental setup for online fault diagnosis of induction machines via promising IoT and machine learning: Towards industry 4.0 empowerment. IEEE access, 9, pp.115429-115441. Digital Object Identifier 10.1109/ACCESS.2021.3105297

[19]Dwibedi, S., Pujari, M., & Sun, W. (2020). A Comparative Study on Contemporary Intrusion Detection Datasets for Machine Learning Research. 2020 IEEE International Conference on Intelligence and Security Informatics (ISI). pp.1-6. doi:10.1109/isi49825.2020.9280519

[20]Bagaa, M., Taleb, T., Bernabe, J. B., &Skarmeta, A. (2020). A Machine Learning Security Framework for Iot Systems. IEEE Access, 8, pp.114066–114077. doi:10.1109/access.2020.2996214

[21] Hossain, M.A. and Islam, M.S., (2023). Ensuring network security with a robust intrusion detection system using ensemble-based machine learning. Array, 19, pp.-14. https://doi.org/10.1016/j.array.2023.100306.

[22] Stevens, T. (2020). Knowledge in the grey zone: AI and cybersecurity. Digital War, 1(1-3), pp.164–170. doi:10.1057/s42984-020-00007-w

[23]Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., &Foozy, C. F. M. (2021). Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset. IEEE Access, 9, pp.22351–22370. doi:10.1109/access.2021.3056614

[24]Mbona, I. and Eloff, J.H., (2022). Detecting zero-day intrusion attacks using semi-supervised machine learning approaches. IEEE Access, 10, pp.69822-69838. Digital Object Identifier 10.1109/ACCESS.2022.3187116

[25] CICIDS 2019 Dataset. Retrieved from https://www.kaggle.com/datasets/tarundhamor/cicids-2019-dataset