A Comparative Evaluation of Machine Learning Algorithms for IDS in IoT network

Seyed Amir Mousavi
Computer Engineering
Ferdowsi University
Mashhad, Iran
amirmousavi@mail.um.ac.ir

Mostafa Sadeghi
Department of Computer
Islamic Azad University
Zavareh, Iran.
mostafa13h@gmail.com

Mohammad Sadegh Sirjani
Department of Computer
Ferdowsi University of Mashhad
Mashhad, Iran
mohammadsadegh.sirjani@mail.um.ac.ir

2023 14th International Conference on Information and Knowledge Technology (IKT) | 979-8-3503-4941-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/IKT62039.2023.10433047

Abstract— **With the increasing Internet use, network security has become essential due to the rise in cyber-attacks on network services. To detect these attacks, a robust Intrusion Detection System (IDS) is required. Traditional IDS face challenges like high false alert rates and slow real-time attack detection. Machine learning (ML) can improve this situation, providing a low False Alarm Rate and high detection rates. This research used five ML methods (Logistic Regression, Random Forest, k-Nearest Neighbors, Support Vector Machine, and XGBoost) to classify the UNSW-NB15 dataset. The goal is to evaluate the performance of various machine learning classifiers in detecting attacks for Internet of Things (IoT) network intrusion detection. The study highlighted the importance of further research to reduce false positives and negatives. To evaluate these classifiers, precision, accuracy, recall, and F1 score were used. The results show that XGBoost achieved the highest accuracy and recall. However, only some algorithms performed perfectly in all aspects, suggesting the need for diverse detection strategies. Future research should focus on developing comprehensive systems and ensemble approaches to minimize false alerts and missed detections.**

Keywords—Network Security; Intrusion Detection System; Artificial Intelligence; Machine Learning

I. INTRODUCTION

A cyber-attack targets a network and its resources with the intent of causing damage, disabling, modifying, or gaining unauthorized access. The rise in cyber-attacks poses new challenges for cybersecurity, especially with the emergence of technologies like IoT, cloud computing, and big data, making organizations more vulnerable to such threats. Thus, organizations must take necessary measures to protect their data. The primary goal of network security is to safeguard the network from harmful codes that can alter data and harm network resources [1]. Intrusion Detection Systems (IDS) serve as a second line of defense, scrutinizing all network and computer traffic. They continuously monitor incoming and outgoing traffic to detect hidden anomalies and raise security alerts if unusual activity is observed. IDS inspects network traffic (inbound and outbound) and takes appropriate actions when identifying malicious traffic. IDS can be categorized as anomaly-based or misuse-based. In the misuse-based approach, attacks are detected based on known attack signatures in the network's activity.

In contrast, the anomaly detection approach identifies abnormal system states by comparing them to normal conditions [2]. Figure 1 shows the activity diagram of the system, which includes activities including intrusion detection, observing current behavior, and comparing it with normal behavior. If any deviation is detected, an alarm is triggered.

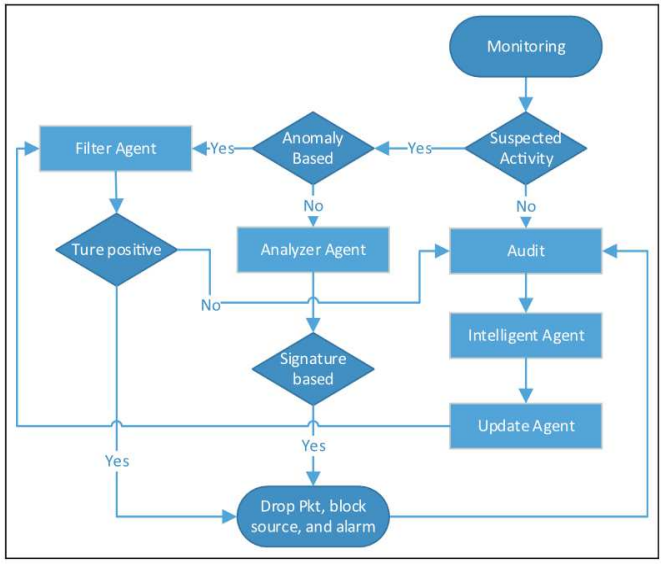

Figure 1: Activity diagram of the IDS [3]

Machine learning algorithms have successfully been applied in various domains, including image processing, natural language processing, and computer vision [4]. These algorithms utilize complex transformation functions and rely heavily on labeled and unlabeled training data to uncover hidden patterns. They employ two critical learning approaches: supervised learning uses training data with labeled examples, while unsupervised learning relies on unlabeled data, with the model identifying inherent patterns [5]. As intruders continually adapt their techniques, the research community must develop dynamic approaches to detect and prevent these intrusions effectively. Developing an efficient IDS capable of identifying new attacks poses significant challenges. The continuous progress in machine learning techniques has enhanced the predictive capabilities and computational power of machines, making them suitable for constructing robust IDS.

Machine learning algorithms can be categorized based on learning techniques, functional similarity, or learning depth, as depicted in Figure 2. Regarding IDS, classic machine learning algorithms might be more suitable than deep learning models due to their lower computational complexity and better

979-8-3503-4941-2/23/$31.00 ©2023 IEEE

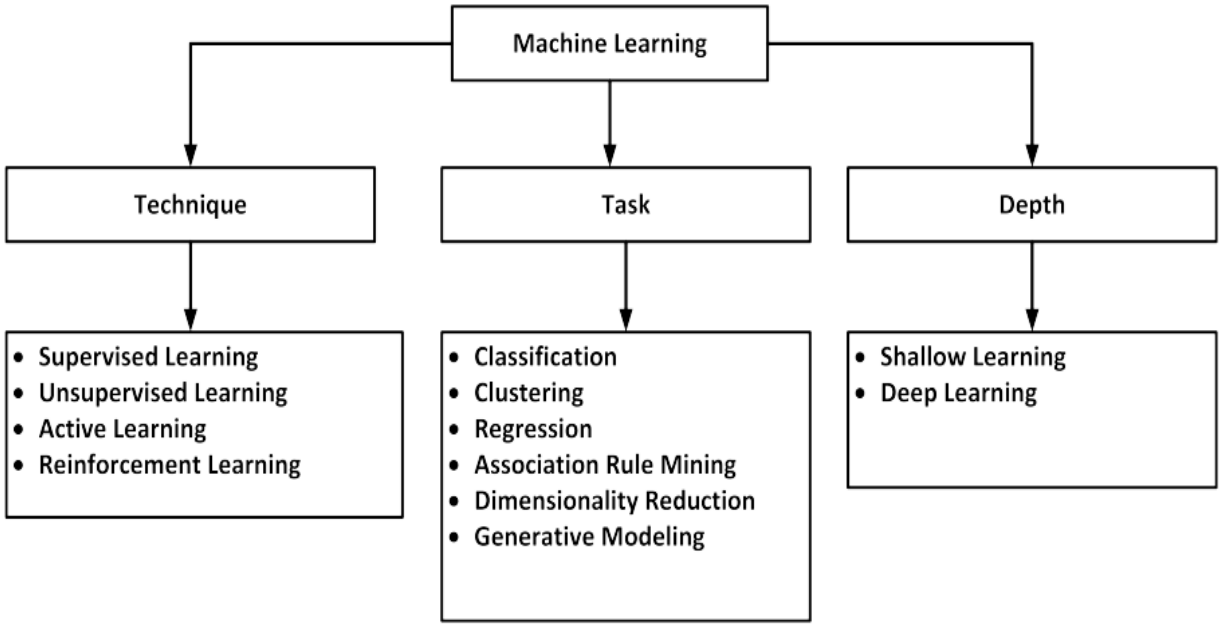

Figure 2: Three-dimensional classification of machine learning

interpretability [6][7]. Studies have demonstrated that traditional machine learning techniques can perform well even with limited training data, while deep learning requires larger datasets for optimal performance. Hence, when dealing with small datasets, using conventional machine-learning approaches is preferable [8]. Moreover, machine learning models are faster to train than deep learning models because they are more straightforward in structure [9].

Intrusion Detection Systems (IDS) encounter significant challenges, including heightened frequencies of erroneous alerts and sluggish real-time identification of attacks. To surmount these constraints and augment the efficacy of IDS, Machine Learning (ML) methods have been utilized, showcasing the capability to deliver reduced rates of false alarms and increased detection rates. This research study investigated the performance of five distinct ML methods, namely Logistic Regression, Random Forest, k-Nearest Neighbors, Support Vector Machine, and XGBoost, in classifying the UNSW-NB15 dataset. The primary objective is to evaluate the capabilities of these classifiers in effectively identifying and distinguishing attacks within the IoT network environment. Several established evaluation metrics, such as precision, accuracy, recall, and F1 score, were employed to evaluate the performance of these classifiers. These metrics provide a comprehensive view of the classifiers' effectiveness in correctly identifying and classifying network activities. The results of this evaluation, with detailed explanations, are presented in the subsequent sections.

II. Review of Related Works

The selection of following papers for review was based on their relevance to the research topic, recent publication date, utilization of similar machine learning methods and datasets, and their impact in the field. Examining these closely related works allows for a better understanding of the current study's context and provides a baseline for comparison. Furthermore, it helps identify research gaps that this study aims to fill.

Fatima et al. introduced a novel machine learning algorithm that combines a genetic algorithm, logistic regression, and artificial neural network (ANN) for Intrusion Detection Systems [10]. In the first stage, logistic regression and the genetic algorithm were utilized to extract a subset of relevant features from the dataset. In the second stage, the artificial neural network was trained using the PSO-GA algorithm to detect intrusions. The model's performance was evaluated using two datasets, NSL-KDD, and KDD cup'99. While the proposed model exhibited lower accuracy compared to other ANN-based methods, it demonstrated faster detection of attack patterns.

Hamamoto et al. [11] developed a system that combines a genetic algorithm and fuzzy logic to enhance IDS performance. The genetic algorithm generates a digital signature to predict network traffic behavior using the concept of Digital Signature Network Section Flow (DSNSF). The use of fuzzy logic aims to address the common issue of high false-positive rates in IDS. By employing the fuzzy method, the system can minimize false positives without compromising its ability to detect anomalies effectively. The authors justify the application of fuzzy logic for Network Anomaly Detection Systems (NADS) due to two main reasons. Firstly, intrusion detection involves numerous numerical features that are statistically collected and measured, which can lead to high detection errors. Secondly, in computer security, there is no precise boundary separating normal and abnormal behavior. The system proposed in this research outperforms other approaches in all evaluated metrics, exhibiting higher accuracy and lower misclassification and false-positive rates.

Verkerken et al. [12] conducted an assessment of unsupervised and self-supervised techniques, which primarily focus on recognizing normal behavior. Any significant deviation

from the normal state is considered potentially malicious. Unsupervised techniques offer the advantage of detecting zero-day attacks by identifying activities that differ from the norm. The study evaluated four unsupervised algorithms, namely Isolation Forest, One-class SVM, and Autoencoder, using the CIC-IDS-2017 dataset. The proposed models were analyzed based on their computational complexity and classification performance.

Zaman et al. [13], discussed IoT threats and presented machine learning based IoT security models. They introduced a layered IoT model but found that all countermeasures primarily involve analyzing network activity of devices at various network levels, specifically through dynamic analysis. However, there was no direct analysis of IoT as a software system. The ML techniques employed were limited to classification, clustering, and regression, without further analysis of the IoT itself.

Alaa Abd and colleagues [14], tackled the challenge of time in Intrusion Detection Systems when dealing with vast amounts of data. They addressed this issue by applying preprocessing to a subset of relevant features to form their model. The Nsl-kdd dataset was utilized for this research. The authors employed the Random Forest algorithm for classifying network data and enhanced its accuracy using the information gain method. During the first stage of preprocessing, the information gain method was utilized to select appropriate features from the dataset. Thirteen features were extracted out of the original forty-one at this stage. In the subsequent stage, the Random Forest algorithm was applied for classification. This algorithm combines multiple trees to create a more robust training model. The binary classification time achieved using this method was 16.84 seconds, with an impressive accuracy of 99.33 percent.

Despite the valuable contributions of these research in utilizing machine learning for network intrusion detection, several limitations still need to be addressed. These studies have highlighted challenges such as high false positive and false negative rates in the schemes [10] and [11], as well as difficulties in real-time attack detection with large datasets [14]. Moreover, many studies rely on older datasets like KDD instead of more up-to-date data [12] and have not specifically focused on IoT networks [13]. Furthermore, most evaluations only consider one or two algorithms on a subset of metrics, lacking a comprehensive benchmark of multiple algorithms. This paper aims to address these gaps by evaluating five machine learning algorithms on the recent UNSW-NB15 IoT dataset. A comprehensive set of evaluation metrics focused on false positives/negatives is used, and the study concludes that an ensemble approach is necessary since no single algorithm excels across all evaluation metrics.

III. Proposed Method

In this section, we present the proposed work, which involves the utilization of five classifiers to categorize packets as either normal or malicious using existing data. In this paper, the NB15-UNSW dataset is used first, and adaptive windowing is performed on each part, after which the appropriate features are selected on each window. As seen in Figure 3, suitable and effective features are selected; after that, we use classification algorithms to identify the network modes, and we use the K-fold method to validate the algorithm. Finally, if the accuracy of the classification is suitable, the algorithm ends, and if the accuracy is not suitable, the feature selection and classification process is repeated. This process continues until we reach the desired accuracy. The model's output has been evaluated using the NB15-UNSW dataset.

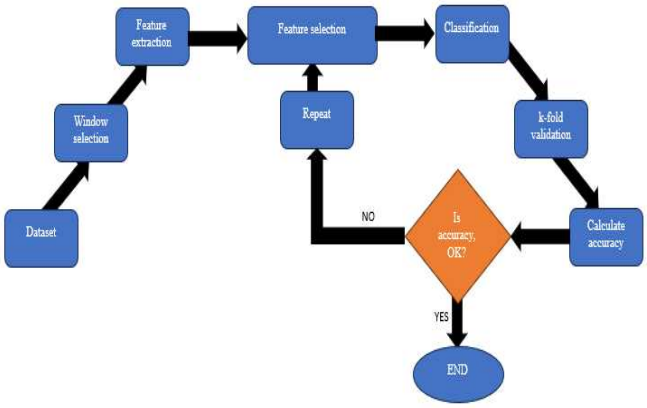

Figure 3. Proposed method for classification

A. Proposed Algorithms

As previously mentioned, ML techniques can address the challenges faced by IDS, including high false alarm rates and limited real-time response. By employing ML techniques, IDS can be enhanced to identify known and unknown attacks, reducing false positives and improving overall performance swiftly and accurately. In the following, we provide a brief description of the five classification algorithms used in this paper:

Logistic Regression (LR) is a supervised machine learning algorithm for classification tasks with discrete classes. It can handle multiple input features that describe network connections, such as protocol, data bytes, and flags and learns weights for each feature. LR performs predictive analysis based on probability concepts. The logistic function, also known as the sigmoid function, is used to map the predicted values to probabilities ranging from 0 to 1 and defined as follows:

$$sigmoid\ (x) = \frac{1}{1+ e^{x}} \quad (1)$$

Where Sigmoid(x), produces an output between 0 and 1. The input to the function is represented by 'x' and 'e' stands for the base of the natural logarithm. This function plays a crucial role in LR by transforming the output into a probability score.

Random forest (RF) offers built-in feature importance metrics, allowing for better model interpretability. Understanding the relevance of features is crucial for IDSs. RF is a complex non-linear supervised algorithm used for classification and regression. It generates multiple decision trees during the model training process and combines their predictions to produce an overall result, making it an ensemble technique. RF classifiers work in a way that increasing the number of trees in the model improves accuracy without risking overfitting. Previous studies have demonstrated that RF is often effective in achieving high accuracy for network intrusion detection tasks [15].

K-Nearest Neighbors (KNN) offers transparency in the classification process through distance and neighbor analysis, unlike black-box models [16]. The principle of KNN classification involves classifying a data point based on its proximity to the k nearest neighbors. The decision is made by considering the majority of neighbor votes. In the first step of the classification process, each data point is placed as a node in an n-dimensional space, where n represents the number of features in the dataset. In the second step, KNN calculates the Euclidean distance, as shown in formula (2), between the input data and each existing node.

In the third step, the data is sorted in ascending order, and most labels among the k nearest neighbors are calculated. The sorting process determines the computational complexity of the algorithm. KNN can outperform neural networks, especially when dealing with small or imbalanced datasets, which are common in IDS.

$$d(x.y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (2)$$

Support Vector Machines (SVMs) are proficient at modeling intricate non-linear decision boundaries to distinguish between normal and attack traffic, even when the data is not linearly separable. They excel in generalizing to identify new and evolving attacks, outperforming linear models such as logistic regression [17]. SVMs exhibit strong generalization capabilities and are well-suited for binary classification tasks like anomaly detection [18]. As one of the most widely used supervised machine learning algorithms, SVM can be trained with labeled data and applied to both classification and regression problems.

XGBoost demonstrates superior detection accuracy for cyber intrusions when compared to RF, SVM, and neural networks. XGBoost is a recently popular algorithm in the machine learning field, known for its high speed and performance. It is an implementation of decision tree gradient boosting, and its features include model features, system features, and algorithm features. XGBoost is notably faster compared to other gradient boosting implementations and decision tree methods.

B. Dataset

This study exclusively uses the UNSW-NB15 dataset to evaluate machine learning algorithms for IDS. UNSW-NB15 is selected as it serves as a modern benchmark dataset specifically designed for assessing IDS performance. Unlike older datasets such as KDD and NSL-KDD, which include synthetic and outdated samples, UNSW-NB15 contains real and contemporary normal traffic along with up-to-date attack scenarios generated by the IXIA PerfectStorm tool. This dataset presents a hybrid of current normal activities and synthetic attack behaviors, leading to a more realistic and balanced distribution compared to other datasets. UNSW-NB15 comprises 49 features extracted using 12 algorithms, providing meaningful and distinguishing attributes for analysis. As a recent IoT-focused dataset with modern attack traffic and relevant features, UNSW-NB15 proves to be an ideal resource for evaluating intrusion detection performance on contemporary networks. The use of a single standardized dataset ensures consistent comparison between the machine learning models. The total number of records in the dataset is 2,540,044, distributed across four files: UNSW-NB15, UNSW-NB15, UNSW-NB15, and UNSW-NB15. The dataset is further partitioned into training and test sets named UNSW-NB15_training-set.csv and UNSW-NB15_testing-set.csv, respectively. The dataset contains 2,218,761 normal records, accounting for approximately 81.5% of the total, and 499,458 attack/unusual records, representing about 18.5% of the total records. This leads to a significant class imbalance in the dataset.

C. Preprocessing

Data imbalance is a well-known issue in ML, occurring when the distribution of different classes is uneven. This can range from slight to severe imbalances in an imbalanced dataset. Training a learning model on a severely imbalanced dataset can lead to poor predictive performance, particularly for the minority classes. The UNSW-NB15 datasets are a prime example of imbalanced data, with 81.5 percent of the data representing usual instances. To address the influence of extensive features, normalization is crucial. This process ensures that the features in the datasets are scaled and plotted uniformly within the range [0,1].

$$x = \frac{X - MIN}{MAX - MIN} \quad (3)$$

IV. RESULTS

The experiments were performed in Google Colab to assess the performance of all classifiers in classifying the UNSW-NB15 dataset. The proposed solution was evaluated from multiple perspectives. This section presents the criteria and results of the evaluation.

A. Evaluation Metrics

The proposed solution must be evaluated from various perspectives. In data classification, each instance or individual belongs to either the positive or negative class. When using an algorithm for classification, each instance will be assigned to one of these two classes. Thus, four situations may occur for each data instance:

- True Positive: The instance belongs to the positive class and is correctly detected as a member of the same class.
- False Negative: The instance belongs to the positive class but is incorrectly detected as a member of the negative class.
- True Negative: The instance belongs to the negative class and is correctly detected as a member of the same class.
- False Positive: The instance belongs to the negative class but is incorrectly detected as a member of the positive class.

Accuracy is the most used criterion for evaluating classification methods. It indicates the percentage of data that has been correctly classified. The formula for calculating accuracy is as follows:

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

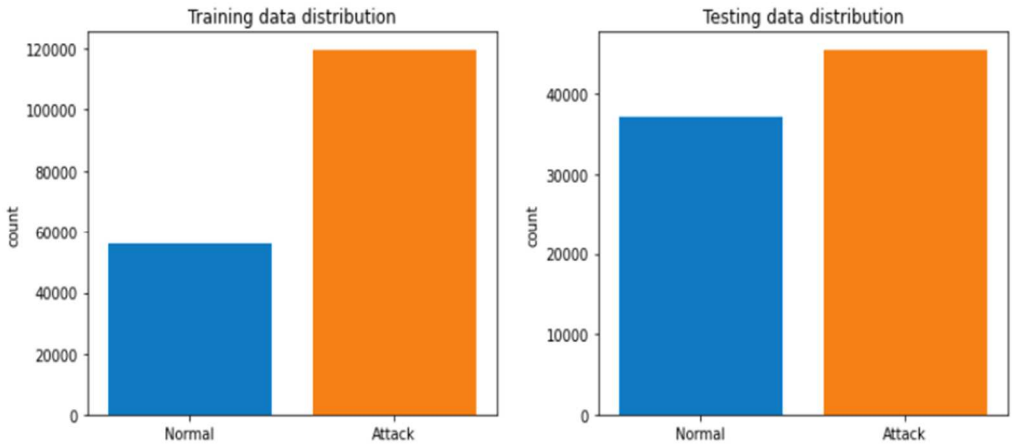

Figure 4. Training and Testing Dataset

Precision is a criterion that measures the percentage of data detected as members of a particular class that truly belong to that class. In the context of our problem, precision indicates the percentage of streams detected as attacks that are actually genuine attacks. The formula for calculating precision is as follows:

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

Recall is a criterion that measures the percentage of data belonging to a specific class that has been correctly detected. In our problem, recall is crucial as our objective is to identify all attacks. The formula for calculating recall is as follows:

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

F1 Score is a criterion that represents the harmonic mean of Recall and Precision. It is a balanced measure that takes into account both the ability to correctly identify positive instances (Recall) and the accuracy of positive predictions (Precision). The formula for calculating the F1 Score is as follows:

$$F1 = 2\frac{Pre * Recall}{Pre + Recall} \quad (7)$$

B. *Results Analysis*

Figure 4 illustrates the dataset divided into normal and attack categories. Table 1 shows the confusion matrix obtained from applying the RL algorithm. Figure 5 displays the confusion matrix resulting from the implementation of the RF algorithm. Figure 6 demonstrates the confusion matrix obtained from the k-Nearest Neighbor algorithm. Table 2 presents the confusion matrix resulting from the implementation of the XGBoost algorithm. Table 3 shows the confusion matrix resulting from the SVM algorithm applied to the UNSW-NB15 dataset. Additionally, Figure 7 provides a comparison of all five algorithms.

Table 1. Confusion Matrix for Logistic Regression

Predicted label True label	0 (normal)	1 (attack)
0 (normal)	17782	19218
1 (attack)	475	44857

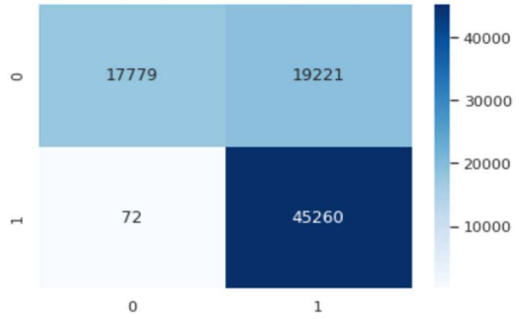

Figure 5. RF Confusion Matrix

Table 2. Confusion Matrix for XGBoost

Predicted label True label	0 (normal)	1 (attack)
0 (normal)	**27558**	9442
1 (attack)	1113	**44219**

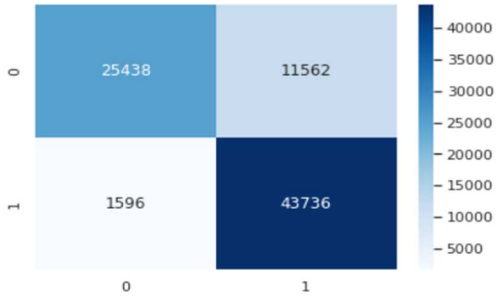

Figure 6. Confusion Matrix for KNN

Table 3. Support Vector Machine Confusion Matrix

Predicted label True label	0	1
0	**26971**	10029
1	8560	**36775**

Among the evaluated algorithms, XGBoost achieved the highest accuracy at 87%, with KNN and SVM tied closely at 84%. Logistic regression had the best precision score at 90%, indicating a low false positive rate. KNN, XGBoost, and SVM demonstrated good precision at 79-82%. XGBoost also obtained the highest recall score at 90%, indicating its ability to correctly identify positive instances. SVM and KNN showed strong recall at 85-86%. KNN achieved the top F1 score at 88%, indicating a good balance between precision and recall. XGBoost and SVM scored 87% and 85% F1 respectively. No single algorithm excelled across all metrics, emphasizing the need for an ensemble approach rather than relying solely on one machine learning model. Overall, the top three models in terms of their performance on accuracy, precision, recall, and F1 score are XGBoost, KNN, and SVM.

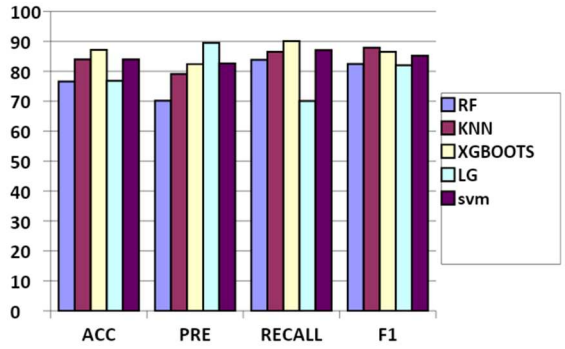

Figure 7. Comparison of Five Machine Learning Algorithms

V. Conclusion

This research examined the intricacies of constructing Intrusion Detection Systems for IoT models by assessing various machine learning algorithms. The experimental findings confirm prior studies, which demonstrated that no single ML approach can optimally detect all attack types. The performance of each tested algorithm varies across accuracy, precision, recall, and F1 metrics, with certain algorithms excelling in specific evaluation criteria. For example, XGBoost demonstrated the highest accuracy and recall, while logistic regression had the best precision score. However, relying solely on any single algorithm is inadequate, given the inconsistency in their metric scores. These findings align with existing research, which has demonstrated that an ensemble approach combining multiple models produces superior results compared to relying on any single model.

To mitigate the shortcomings of any individual technique, an ensemble of diverse classifiers is critical for improving robustness. The complex nature of contemporary attacks necessitates combining the strengths of different machine-learning approaches within a hybrid framework. In the future, research should focus on constructing integrated ensemble systems to effectively minimize false positives and negatives. The study's conclusions align with existing literature, emphasizing the importance of adopting an integrated approach rather than relying solely on one machine learning model for dependable intrusion detection.

VI. References

[1] S. S. Swarna Sugi and S. R. Ratna, "Investigation of Machine Learning Techniques in Intrusion Detection System for IoT Network," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS). IEEE, Dec. 03, 2020. doi: 10.1109/iciss49785.2020.9315900.

[2] N. Alghanmi, R. Alotaibi, and S. M. Buhari, "Machine Learning Approaches for Anomaly Detection in IoT: An Overview and Future Research Directions," Wireless Personal Communications, vol. 122, no. 3. Springer Science and Business Media LLC, pp. 2309–2324, Aug. 27, 2021. doi: 10.1007/s11277-021-08994-z.

[3] S. T. Bakhsh, S. Alghamdi, R. A. Alsemmeari, and S. R. Hassan, "An adaptive intrusion detection and prevention system for Internet of Things," International Journal of Distributed Sensor Networks, vol. 15, no. 11. SAGE Publications, p. 155014771988810, Nov. 2019. doi: 10.1177/1550147719888109.

[4] C. A. de Souza, C. B. Westphall, R. B. Machado, J. B. M. Sobral, and G. dos S. Vieira, "Hybrid approach to intrusion detection in fog-based IoT environments," Computer Networks, vol. 180. Elsevier BV, p. 107417, Oct. 2020. doi: 10.1016/j.comnet.2020.107417.

[5] Prabhat Kumar, G. P. Gupta, and R. Tripathi, "Design of Anomaly-Based Intrusion Detection System Using Fog Computing for IoT Network," Automatic Control and Computer Sciences, vol. 55, no. 2. Allerton Press, pp. 137–147, Mar. 2021. doi: 10.3103/s0146411621020085.

[6] J. Lansky et al., "Deep Learning-Based Intrusion Detection Systems: A Systematic Review," IEEE Access, vol. 9. Institute of Electrical and Electronics Engineers (IEEE), pp. 101574–101599, 2021. doi: 10.1109/access.2021.3097247.

[7] Dehnavi, M. S., Dehnavi, V. S., & Shafiee, M. (2021, December). Classification of mental states of human concentration based on EEG signal. In 2021 12th International Conference on Information and Knowledge Technology (IKT) (pp. 78-82). IEEE.

[8] S. Naseer et al., "Enhanced Network Anomaly Detection Based on Deep Neural Networks," IEEE Access, vol. 6. Institute of Electrical and Electronics Engineers (IEEE), pp. 48231–48246, 2018. doi: 10.1109/access.2018.2863036.

[9] T. E. Ali, Y.-W. Chong, and S. Manickam, "Comparison of ML/DL Approaches for Detecting DDoS Attacks in SDN," Applied Sciences, vol. 13, no. 5. MDPI AG, p. 3033, Feb. 27, 2023. doi: 10.3390/app13053033.

[10] Z. Fatima and A. Ali, "Effective Metaheuristic Based Classifiers for Multiclass Intrusion Detection." arXiv, 2022. doi: 10.48550/ARXIV.2210.02678.

[11] A. H. Hamamoto, L. F. Carvalho, L. D. H. Sampaio, T. Abrão, and M. L. Proença Jr., "Network Anomaly Detection System using Genetic Algorithm and Fuzzy Logic," Expert Systems with Applications, vol. 92. Elsevier BV, pp. 390–402, Feb. 2018. doi: 10.1016/j.eswa.2017.09.013.

[12] M. Verkerken, L. D'hooge, T. Wauters, B. Volckaert, and F. De Turck, "Unsupervised Machine Learning Techniques for Network Intrusion Detection on Modern Data," 2020 4th Cyber Security in Networking Conference (CSNet). IEEE, Oct. 21, 2020. doi: 10.1109/csnet50428.2020.9265461.

[13] S. Zaman et al., "Security Threats and Artificial Intelligence Based Countermeasures for Internet of Things Networks: A Comprehensive Survey," IEEE Access, vol. 9. Institute of Electrical and Electronics Engineers (IEEE), pp. 94668–94690, 2021. doi: 10.1109/access.2021.3089681.

[14] A. Alhowaide, I. Alsmadi, and J. Tang, "Ensemble Detection Model for IoT IDS," Internet of Things, vol. 16. Elsevier BV, p. 100435, Dec. 2021. doi: 10.1016/j.iot.2021.100435.

[15] P. A. A. Resende and A. C. Drummond, "A Survey of Random Forest Based Methods for Intrusion Detection Systems," ACM Computing Surveys, vol. 51, no. 3. Association for Computing Machinery (ACM), pp. 1–36, May 23, 2018. doi: 10.1145/3178582.

[16] N. Thockchom, M. M. Singh, and U. Nandi, "A novel ensemble learning-based model for network intrusion detection," Complex & Intelligent Systems, vol. 9, no. 5. Springer Science and Business Media LLC, pp. 5693–5714, Apr. 03, 2023. doi: 10.1007/s40747-023-01013-7.

[17] R. Doshi, N. Apthorpe, and N. Feamster, “Machine Learning DDoS Detection for Consumer Internet of Things Devices,” 2018 IEEE Security and Privacy Workshops (SPW). IEEE, May 2018. doi: 10.1109/spw.2018.00013.
[18] M. Myint Oo, S. Kamolphiwong, T. Kamolphiwong, and S. Vasupongayya, “Advanced Support Vector Machine- (ASVM-) Based Detection for Distributed Denial of Service (DDoS) Attack on Software Defined Networking (SDN),” Journal of Computer Networks and Communications, vol. 2019. Hindawi Limited, pp. 1–12, Mar. 04, 2019. doi: 10.1155/2019/8012568.