[1]Dr. Satish S. Banait

[2]Dr. Ashwini B. Gavali

[3]Dr. Shrinivas T. Shirkande

[4]Aditi Lule

[5]Dr. Anup Bhange

[6]Kanchan Wagh

An Approach for Detecting Security Attacks using Machine Learning in IoT Environment

Abstract: - The strong security measures are becoming even more necessary to protect these networked systems as a result of the proliferation of Internet of Things (IoT) devices. The dynamic and varied nature of IoT networks frequently makes it impossible for traditional security solutions to be effective. The method suggested in this research uses machine learning to identify security attacks in IoT contexts. The suggested method makes use of the capabilities of machine learning algorithms to examine the enormous amounts of data produced by IoT devices. The system can develop the ability to recognise possible security threats and take immediate action by training models on labelled datasets that include both normal and attack patterns. In this paper multiple ML models for security threat detection in IoT environments in this study. Our evaluation uses both binary and multiclass classification models in an effort to accurately reflect the variety of assaults that can be found in IoT environments. The proposed method provide a new specialised IoT dataset that was created especially to reflect the characteristics of actual IoT environments in order to assure the validity of our assessment. By completing this extensive analysis, we hope to shed light on how well AI-based methods for security attack detection in IoT contexts work. The results of this study can help researchers and professionals decide which ML models and feature engineering techniques are best for IoT security. At the end of the day, we want to help with the creation of reliable security systems that safeguard IoT devices and shield user data from harmful attacks.

Keywords: Machine Learning, Network Security, Internet of Things, Security Attacks

I. INTRODUCTION

IoT is based on the idea that by connecting billions of devices, the world can collect and share enormous amounts of data. This connectivity, along with cutting-edge communication technologies, makes it possible to analyse data and make decisions quickly. The number of IoT strategies worldwide has already topped 16.4 billion, and by 2020, Statistica [2] predicts that this number will surpass 30 billion. Attackers now have ways to take advantage of IoT devices and compromise sensitive data thanks to these flaws. The variety of IoT devices and their frequent connections with external networks create opportunities for malicious actors to enter the system. The compromise of integrity can result in the manipulation of data, which can lead to inaccurate conclusions and incorrect decision-making, while the breach of confidentiality can result in unauthorised access to personal information. Understanding IoT vulnerabilities in depth and creating strong security measures are necessary to address these

[1]Assistant Professor, Department of Computer Engineering, K.K. Wagh Institute of Engineering Education and Research, Nashik. Maharashtra.(SPPU-Pune)

[2]Assistant Professor, S. B. Patil College of Engineering Indapur, Pune, Maharashtra, India

[3]Principal, S.B.Patil College of Engineering Indapur, Pune, Maharashtra, India

[4]Assistant Professor, Symbiosis School of Planning Architecture and Design, Nagpur Campus, Symbiosis International (Deemed University), Pune, India

[5]Assistant Professor, Department of Computer Science and Engineering, KDK College of Engineering Nagpur, Maharashtra, India

[6]Assistant professor, Department of Electronics and Telecommunication, Cummins College of Engineering for Women, Nagpur, Maharashtra, India

ssbanait@kkwagh.edu.in[1], dnyane.ash@gmail.com[2], shri.shirkande8@gmail.com[3], aditi.lule@sspad.edu.in[4], anupbhange@gmail.com[5], kanchanwagh5@gmail.com[6]

security concerns. IoT security has been improved by the use of methods like encryption, access control, and secure communication protocols. In addition, continued integrity and safety of IoT systems depend on the detection and monitoring of security assaults.

Several fundamental difficulties, including as privacy, access control, authentication, secrecy, and trust, must be overcome when designing a secure IoT ecosystem. Due to the sensitive nature of the user data processed by the majority of IoT devices, these difficulties are essential. The existence of malware botnets like Mirai, as emphasised by Kolias et al. [3], shows the potential for rapid propagation and control of IoT devices, exploiting their weaknesses. This underlines the fact that vulnerable IoT devices directly endanger not only themselves but also every other connected item in their network. Attackers frequently aim for IoT devices in an effort to illegally get user data, which could lead to monetary damages and breaches of privacy [4],[5]. Network assaults like phishing, theft of information. Data breaches and ransomware assaults are only two examples of the serious consequences that can result from these attacks, which can cause organisations significant financial and operational harm.

Phishing attacks take advantage of flaws in IoT devices to trick users into disclosing private information, jeopardising their security and privacy. Unauthorised access to private or sensitive information kept on Internet of Things devices might result in data misuse or disclosure. Spoofing attacks impersonate legitimate IoT networks or devices in order to obtain sensitive data or take control of targets. Attacks known as denial of service (DoS) stop IoT networks or devices from operating normally and make them unavailable or unresponsive. These cyber security vulnerabilities can have serious repercussions that will cost a lot of money and time to repair. Sensitive information may be exposed via data breaches, harming an organization's reputation and having legal repercussions. Critical data can be encrypted by ransomware attacks, which demand payments in exchange for its disclosure, resulting in operational hiccups and monetary losses. Implementing strong authentication mechanisms, enforcing access control regulations, encrypting data communications, and routinely updating IoT device firmware to fix vulnerabilities are just a few of the many different strategies needed to address these security concerns. Real-time detection of possible assaults can be aided by continuous monitoring and threat detection systems.

II. BACKGROUND

In order to better protect IoT networks from assaults, several academics have concentrated on creating machine learning (ML) models. The UNSW-NB15 dataset is frequently utilised in literature reviews [13]. A model based on machine learning for intrusion detection was developed in a different study by Khatib et al. [17] to increase the resilience of IoT networks to hostile attacks. The [13] researchers used the 2,540,044 samples and 49 features in UNSW-NB15 dataset. They applied the several machine learning classifiers to deal with the problem of network security in IoT. The SVM achieved the highest multiclass classification accuracy of 93%. These papers show how ML approaches may be used to rise the precision of intrusion recognition in IoT networks. The UNSW-NB15 dataset and multiple ML classifiers have been used to get important insights into how well various models perform at identifying cyberattacks. These results support ongoing efforts to strengthen IoT system security and defend against malicious actions.

Table 1: Summary of Different ML Method used in IoT Environment

Paper	Algorithm Used	Dataset	Attributes	Findings
[14]	ERT, AB, XGB, CART, GBM, MLP, RF	CIDDS-001, UNSW-NB15, NSL-KDD	-	Method shows 96.47% Accoracy
[18]	LR, SVM, RF, DT, KNN, ANN, bagging, boosting, and stacking ensemble	UNSW-NB15 (175,341 samples), CIC-IDS2017 (190,774 samples)	25	With use of ensemble method 99.12& Accuracy

[17]	RF, DT, AdaBoost, LR, LDA, SVM, Nystrom-SVM	UNSW-NB15 (2,540,044 samples)	49	95% of Accuracy using Binary classification and 93% accuracy using SV Multiclass classification
[19]	RF	UNSW-NB15 (699,934 samples)	12	It represent the 99.34% accuracy
[20]	LR, NB, DT, SVM, KNN, RF, AB, XGB	ToN-IoT	20	98% of Accuracy using Binary classification and 97% accuracy using Multiclass classification
[22]	RF, GBM	CSE-CIC-IDS2018-V2	-	98.27% accuracy
[24]	RF, SVM, DT, ANN, KNN, NB	SCADA attacks dataset	7	99.84% accuracy using RF
[25]	GRU	Modbus-based network dataset	-	90.25% Accuracy

III. DATASET

The dataset Wheelus and Zhu [27] offered for public use is the one we used in our investigation. This dataset was gathered over a nine-month period and went through numerous preprocessing stages. The raw data was divided into sessions during preprocessing based on factors including source and destination Internet Protocol (IP) addresses, ports, and temporal features. The features contained in the dataset are shown in Table 2. To categorise samples into both the attack or regular categories in the binary classification case. The dataset tries to group samples in the multiclass classification scenario into four groups: normal, querying cache (QC), zone transference (ZT), and no shared secrets (NSS).

Table 2: Description of Dataset

Sr. No.	Attribute Name	Details of Attribute
1	in_rep	Repetition of the session—packet count of those with the most typical packet size
2	out_rep	Repetition of the session—packet count of those with the most typical packet size
3	in_prdcty	Measure of a session's periodicity provided by the variance of timestamp discrepancies between packets
4	out_prdcty	Measure of a session's periodicity provided by the variance of timestamp discrepancies between packets
5	in_conv	Examining the variation in packet sizes reveals session convergence, which is the self-similarity of the packets in the session.

6	out_conv	Examining the variation in packet sizes reveals session convergence, which is the self-similarity of the packets in the session.
7	invel_pps	Traffic speed is determined in packets per second, or packets per second.
8	outvel_pps	Traffic speed is determined in packets per second, or packets per second.
9	invel_bps	Bits per second (bps) refers to the speed of data transfer.
10	outvel_bps	Bits per second (bps) refers to the speed of data transfer.
11	invel_bpp	Bytes per packet—traffic speed expressed in bytes each packet
12	outvel_bpp	Bytes per packet—traffic speed expressed in bytes each packet
13	riotp	RIOT packages (inbound and outbound traffic combined) are the ratio of inbound to outgoing traffic measured in packets.
14	riotb	RIOT packages (inbound and outbound traffic combined) are the ratio of inbound to outgoing traffic measured in packets.
15	duration	Duration is the sum of the session's inbound and outbound times.
16	orig_bytes	Sessions traffic size in bytes, or byte count
17	resp_bytes	Sessions traffic size in bytes, or byte count
18	orig_packets	Session traffic amount in packets and packet count
19	resp_packets	Session traffic amount in packets and packet count

The binary dataset has a total of 212,834 samples, 178,576 of which are considered to be normal, and 34,258 of which are considered to be assaults. There are four classes in the multiclass dataset: normal, NSS, QC, and ZT. There are 178,576 examples in the standard class, 23,022 in the NSS class, 6,901 in the QC class, and 4,335 in the ZT class.

IV. RESEARCH METHODOLOGY

ML algorithms to identify and categorise security breaches in IoT networks. For this task, the bagging, KNN, J48, RF, LR, and MLP ML models have been chosen. We used a publically accessible dataset from Wheelus and Zhu [27] that was created specifically for identifying and classifying IoT network assaults to train these algorithms. We performed many preparation steps to make sure the dataset was in the best format before starting the model training process.

Following that, the efficacy of the ML models was evaluated using a range of metrics, including classification accuracy, F-score, recall, precision, and receiver operating characteristic (ROC). The models' robustness and generalizability were confirmed by using 10-fold cross-validation during their creation. We carried out two binary classification experiments to discriminate between normal and attack periods. Additionally, we conducted two classification tests utilising multiclass classification to categorise both typical sessions and three distinct attack types, including zone transfer (ZT), query cache, and no shared secret (NSS). To further emphasise the importance of feature selection in preserving or enhancing the models' performance, we only used a portion of the features

during the tests. Figure 1 clearly represents the stages of the research approach and the general flow of our investigation.

a) Preprocessing:

Preprocessing operations were carried out to get the dataset ready for model testing and training. These procedures included loading, purging, treating, and converting the data into a format that was appropriate for the tasks at hand. 188,576 of the 218,834 cases in the original dataset, which represented typical traffic, totalled 218,834. As the normal class accounted for 83.9% of the sample, while the other classes were noticeably lower in contrast, this suggests a large class imbalance. Different strategies were used to handle the missing values based on the characteristics of each class, taking into account the uneven nature of the dataset and the occurrence of missing values across all classes. The normal class was chosen as the method for missing value handling since it was substantially larger than the other classes.

The average value for the binary categorization experiments was used to impute the missing values for the second class, which represents attacks. Because the characteristics with missing values related to variance values, this method was reasonable, and employing the median as an imputation method was suitable in this situation.

The missing data in the NSS class were eliminated in the multiclass experiments. This strategy was chosen such that even when the missing data were removed, there would still be more NSS attack instances than the lowest class. These preprocessing methods ensured a more solid basis for the upcoming studies by treating the dataset suitably to address class imbalance and missing values.

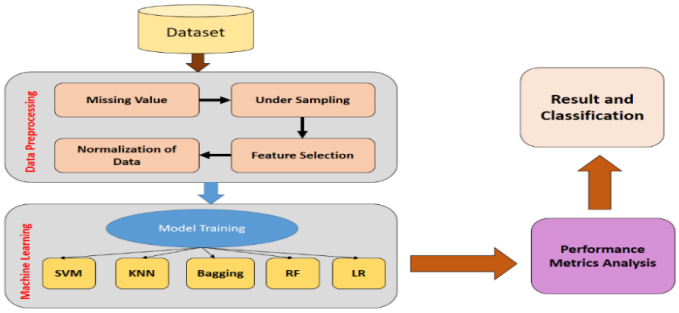

Figure 1: Proposed method Step wise representation

b) Feature Selection:

The dataset used in our analysis included 20 attributes that provide pertinent data on traffic sessions. To acquire the best results in our analysis, we ran four different trials. The class attribute for the binary classification studies had two possible values: normal or attack. The attack values were then divided into three classes for the multiclass classification experiments: NSS, QC, and ZT.

Table 3: Feature selection and Relation with Information Gain Vs Correlation

Attribute	Correlation in %	Information Gain in %
orig_packets	67.09	64.8
riotp	64.91	86.0
outvel_bpp	66.39	85.4
orig_bytes	64.77	82.0
resp_packets	51.77	66.3
resp_bytes	54.23	81.1
duration	29.33	68.9

We determined the correlation between the data gain of each feature in the dataset before doing feature selection. Based on these evaluations, we chose the top 30% of characteristics. Seven features were ultimately picked since they had the highest correlations. The characteristic "duration" was also added because of its great information gain value. The "riotb" and "riotp" qualities have a 100% association, we noticed after computing the correlation between the chosen features. We decided to preserve just one of these functionalities to prevent duplication. We kept the "riotb" feature because of its higher information gain value. So, for all four studies, we had a total of seven attributes.

C) Performance Metrics:

The accuracy (ACC) is calculated as the percentage of correctly classified instances, whether they are normal or attacks, and is determined by the following formula:

$$ACC = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

The formula for calculating precision (P), which is the proportion of pertinent instances among the identified instances:

$$P = \frac{TP}{(TP + FP)}$$

Recall (R) is calculated as the ratio of the number of relevant instances over the total number of relevant instances discovered:

$$\mathrm{R} = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FN})}$$

The F1-Score is a metric that combines recall and precision into one number. It can be calculated using the formula below as the weighted average of recall and precision:

$$\mathrm{F1Score} = \frac{(2 * \mathrm{P} * \mathrm{R})}{(\mathrm{P} + \mathrm{R})}$$

In particular, when α = 1, the formula for the F1-Score simplifies. Overall, these formulas allow us to calculate accuracy, precision, recall, and the F1-Score, which are commonly used metrics for evaluating classification performance.

V. RESULT AND DISCUSSION

For the tests in our study, we used a range of machine learning (ML) models, comprising RF, LR, KNN, SVM, bagging ensemble, and MLP. These techniques were chosen in light of earlier research [14, 18, 20, 24] that showed their efficacy in building classification models across a range of domains. The collection initially had 218,834 occurrences. But because of the problem of class imbalance, we had to solve it by under sampling the dataset. To provide a balanced representation of the classes, each set of experiments contained a varied number of cases. Additionally, we chose features depending on how well they correlated with the target class. Seven features were chosen for the following experiments after examining the feature correlations.

Table 4: Performance evaluation ML model with Precision and Recall

Model	Precision	Recall
KNN	0.93	0.94
LR	0.98	0.98
SVM	0.99	0.99
RF	0.99	0.99
Bagging	0.98	0.99

The outcomes of our tests demonstrated how well different machine learning models performed in terms of recall and precision in table 4. Precision and recall for KNN were 0.93 and 0.94, respectively. This indicates that KNN did a good job accurately detecting positive cases (high accuracy) and accurately capturing a large percentage of the real positive instances (high recall). Precision and recall for LR were both 0.98. A significant number of the real positive occurrences were successfully recorded by LR, which showed a high level of accuracy in recognising positive instances. SVM achieved 0.99 precision and 0.99 recall. Because of its excellent precision and recall levels, SVM performed remarkably well at correctly recognising positive cases.

Additionally, RF attained 0.99 for both recall and precision. RF showed a strong aptitude for accurately and vividly recalling examples of positive behaviour. Precision and recall for bagging were 0.98 and 0.99, respectively. In terms of reliably recognising positive instances, bagging fared well, keeping a high recall while having a little lower precision, figure 2 shows the representation precision and recall.

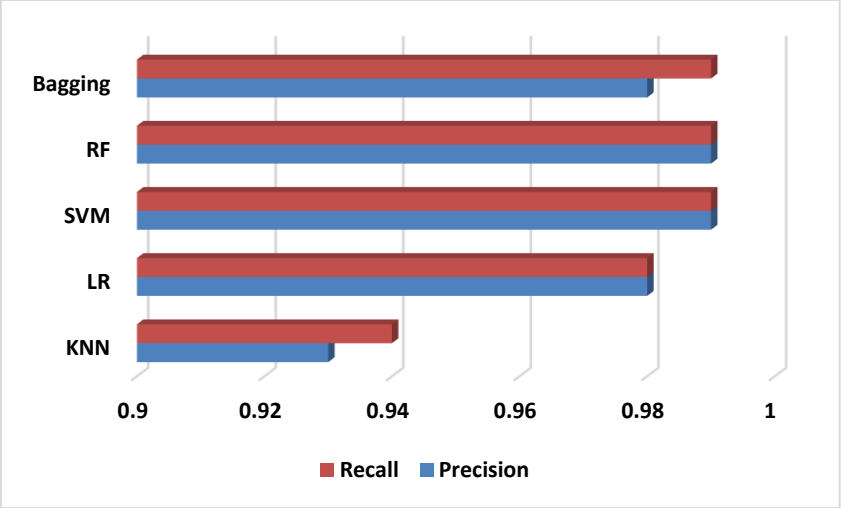

Figure 2: Representation of Performance evaluation ML model with Precision and Recall

Overall, the findings show that good precision and recall values were attained by the models, which included KNN, LR, SVM, RF, and Bagging. These models exhibit promise for accurately categorising events and identifying the crucially important positive instances. To have a thorough knowledge of their performance in the particular setting of the study, it is crucial to take into account other evaluation measures and do additional analysis.

Table 5: Performance evaluation ML model with F1 Score and ROC

Model	F1-Score	ROC
KNN	0.92	0.94
LR	0.95	0.92
SVM	0.94	0.96
RF	0.96	0.99
Bagging	0.95	0.99

We are able to evaluate the effectiveness of various machine learning models based on the F1-Score and ROC values we received from our trials. KNN attained a ROC value of 0.94 and an F1-Score of 0.92. This shows that KNN displayed a strong balance between recall and precision, collecting a large percentage of positive cases while retaining a high level of overall accuracy. According to the ROC value, the model did a good job of differentiating between positive and negative events. The F1-Score and ROC values for LR were 0.95 and 0.92,

respectively. High F1-Score for LR demonstrated great precision and recall skills. The ROC value indicates that the model successfully maintained a low false positive rate while achieving a high true positive rate as mentioned in table 5.

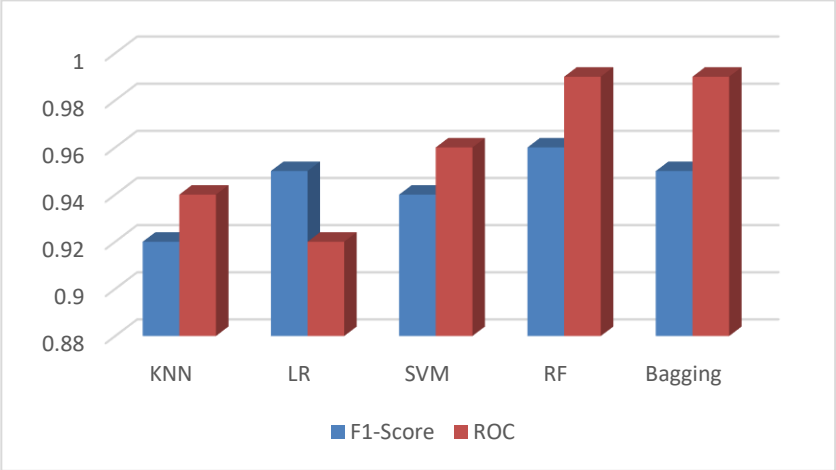

Figure 3: Representation of Performance evaluation ML model with F1 Score and ROC

Using SVM, an F1-Score of 0.94 and a ROC value of 0.96 were obtained. SVM displayed a balanced efficiency with a ROC value and F1-Score that were both rather high. This figure 3 shows that the model successfully identified positive occurrences while reducing false positives. F1-Score of 0.96 and ROC value of 0.99 were attained with RF. As seen by its high F1-Score, RF demonstrated great performance with regard to of precision, recall, and overall accuracy. An excellent capacity to discriminate between both positive and negative instances is indicated by the high ROC value. An F1-Score of 0.95 and a ROC value of 0.99 were attained by bagging. According to its F1-Score, bagging displayed good precision and recall. The high ROC value indicates that the model did a remarkable job of distinguishing between positive and negative events.

Table 6: Performance evaluation ML model with Accuracy

Model	Accuracy
KNN	94.40%
LR	97.38%
SVM	99.77%
RF	99.49%
Bagging	98.46%

KNN had a 94.40% accuracy rate. This shows that KNN did a good job classifying cases accurately, resulting in a pretty high overall accuracy. LR achieved an accuracy rate of 97.38%. LR demonstrated a higher level of accuracy in comparison to KNN, demonstrating that it could categorise events with greater precision. SVM achieved accuracy of 99.77%. With a high accuracy rate for correctly identifying occurrences, SVM performed remarkably well. 99.49% of the time, RF was accurate. The exceptionally high level of accuracy that RF demonstrated its capacity to accurately and precisely identify circumstances. Bagging has an accuracy percentage of 98.46%. Bagging performed well in accurately identifying cases, although being much less precise than certain other models.

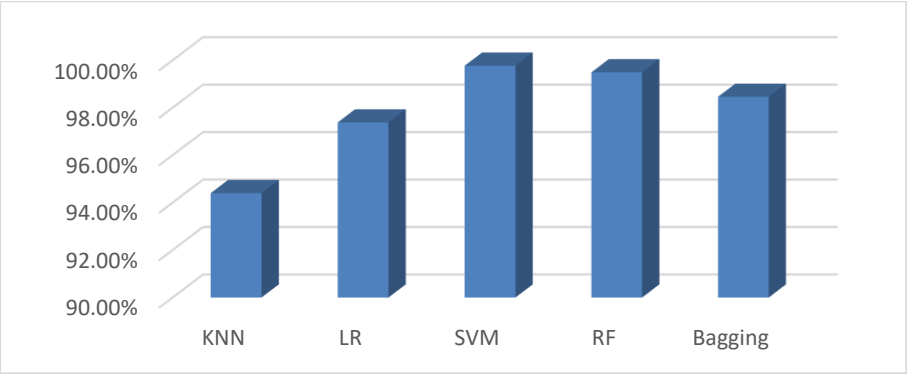

Figure 4: Accuracy of Different ML Methods

Table 7: Comparative analysis of Method

Model	Precision	Recall	F1-Score	Accuracy	ROC
KNN	0.93	0.94	0.92	94.40%	0.94
LR	0.98	0.98	0.95	97.38%	0.92
SVM	0.99	0.99	0.94	99.77%	0.96
RF	0.99	0.99	0.96	99.49%	0.99
Bagging	0.98	0.99	0.95	98.46%	0.99

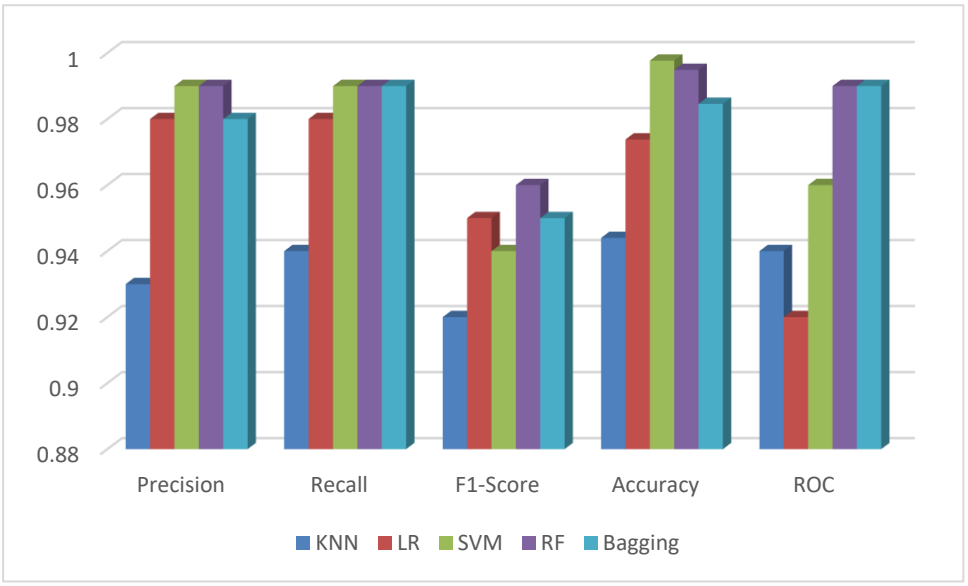

Figure 8: Comparative analysis of different ML Algorithms

The models, which combined KNN, LR, SVM, RF, and bagging, acquired high accuracy values, proving their effectiveness in correctly classifying instances in the given context. It is essential to consider various evaluation criteria and do extra research to fully comprehend their performance and usefulness for the specific problem domain. Performance metrics for the KNN model were precision of 0.93, recall of 0.94, F1-Score of 0.92, accuracy of 94.40%, and ROC of 0.94. Precision, recall, F1-Score, accuracy, and ROC values for LR were respectively 0.98, 0.95, and 0.92. SVM achieved a ROC of 0.96, an F1-Score of 0.94, accuracy of 99.77%,

precision of 0.99, recall of 0.99, and. RF attained ROC values of 0.99, 0.96, 99.49%, F1-Score, precision, recall, and accuracy. Through bagging, a precision of 0.98, recall of 0.99, F1-Score of 0.95, accuracy of 98.46%, and ROC of 0.99 were achieved.

VI. CONCLUSION

The necessity for adequate cybersecurity measures to thwart the growing number of cyber attacks in IoT networks has grown critical with the exponential proliferation of IoT devices. In order to combat these dangers, researchers are increasingly using AI techniques because of their reliability and effectiveness. In this study, we focused on developing machine learning (ML) models with a fresh dataset and engineering features. To adjust to the changing nature of cyber threats, it is essential to investigate new cybersecurity datasets. We performed binary classification studies (normal vs. attack) and multiclass experiments (normal, QC, ZT, NSS). The dataset underwent randomised undersampling to guarantee an equal number of instances across classes in order to alleviate class imbalance. Then, feature selection and normalisation were carried out utilising correlation and information gain. Two sets of tests were run for each classification case: one with all features and the other with the best features chosen. Bagging, KNN, J48, RF, LR, and MLP models were used. Accuracy, F-score, recall, precision, and ROC measures were all included in the performance evaluation. Results showed that RF outperformed all other experiment sets, obtaining an astounding ROC of 99.9%. Furthermore, trials on binary classification performed better than those on multiclass classification. The choice of features has no effect on except for KNN, where it significantly enhanced classifier performance.

To avoid the loss or destruction of sensitive data, future work should concentrate on developing real-time models capable of recognising and categorising attacks in IoT devices in real-time. To further assess the effectiveness of ML models, this dataset can be used as a model for creating customised datasets that cover a greater variety of attack types.

REFERENCES:

[1] Aljabri, M.; Zagrouba, R.; Shaahid, A.; Alnasser, F.; Saleh, A.; Alomari, D.M. Machine learning-based social media bot detection: A comprehensive literature review. Soc. Netw. Anal. Min. 2023, 13, 20.

[2] Global IoT and Non-IoT Connections 2010–2025|Statista. Available online: https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/ (accessed on 21 February 2022).

[3] Kolias, C.; Kambourakis, G.; Stavrou, A.; Voas, J. DDoS in the IoT: Mirai and other botnets. Computer Long. Beach. Calif. 2017, 50, 80–84.

[4] Aljabri, M.; Alhaidari, F.; Mohammad, R.M.A.; Mirza, U.S.; Alhamed, D.H.; Altamimi, H.S.; Chrouf, S.M.B. An Assessment of Lexical, Network, and Content-Based Features for Detecting Malicious URLs Using Machine Learning and Deep Learning Models. Comput. Intell. Neurosci. 2022, 2022, 1–14.

[5] Aljabri, M.; Aldossary, M.; Al-Homeed, N.; Alhetelah, B.; Althubiany, M.; Alotaibi, O.; Alsaqer, S. Testing and Exploiting Tools to Improve OWASP Top Ten Security Vulnerabilities Detection. In Proceedings of the 2022 14th International Conference on Computational Intelligence and Communication Networks (CICN), Al-Khobar, Saudi Arabia, 4–6 December 2022; pp. 797–803.

[6] Das, R.; Tuna, A.; Demirel, S.; Yurdakul, M.K. A Survey on the Internet of Things Solutions for the Elderly and Disabled: Applications, Prospects, and Challenges. Int. J. Comput. Netw. Appl. 2017, 4, 84–92.

[7] Aljabri, M.; Alahmadi, A.A.; Mohammad, R.M.A.; Aboulnour, M.; Alomari, D.M.; Almotiri, S.H. Classification of Firewall Log Data Using Multiclass Machine Learning Models. Electron 2022, 11, 1851.

[8] Aljabri, M.; Mirza, S. Phishing Attacks Detection using Machine Learning and Deep Learning Models. In Proceedings of the 2022 7th International Conference on Data Science and Machine Learning Applications (CDMA), Riyadh, Saudi Arabia, 1–3 March 2022; pp. 175–180.

[9] MORE Alarming Cybersecurity Stats For 2021 ! Available online: https://www.forbes.com/sites/chuckbrooks/2021/10/24/more-alarming-cybersecurity-stats-for-2021-/?sh=4a9c31b24a36 (accessed on 21 February 2022).

[10] Aljabri, M.; Aljameel, S.S.; Mohammad, R.M.A.; Almotiri, S.H.; Mirza, S.; Anis, F.M.; Aboulnour, M.; Alomari, D.M.; Alhamed, D.H.; Altamimi, H.S. Intelligent Techniques for Detecting Network Attacks: Review and Research Directions. Sensors 2021, 21, 7070. [

[11] Aljabri, M.; Altamimi, H.S.; Albelali, S.A.; Al-Harbi, M.; Alhuraib, H.T.; Alotaibi, N.K.; Alahmadi, A.A.; Alhaidari, F.; Mohammad, R.M.A.; Salah, K. Detecting Malicious URLs Using Machine Learning Techniques: Review and Research Directions. IEEE Access 2022, 10, 121395–121417.

[12] Alzahrani, R.A.; Aljabri, M. AI-Based Techniques for Ad Click Fraud Detection and Prevention: Review and Research Directions. J. Sens. Actuator Netw. 2023, 12, 4.

[13] The UNSW-NB15 Dataset|UNSW Research. Available online: https://research.unsw.edu.au/projects/unsw-nb15-dataset (accessed on 19 February 2022).

[14] Verma, A.; Ranga, V. Machine Learning Based Intrusion Detection Systems for IoT Applications. Wirel. Pers. Commun. 2020, 111, 2287–2310.

[15] CIDDS—Coburg Intrusion Detection Data Sets: Hochschule Coburg. Available online: https://www.hs-coburg.de/forschung/forschungsprojekte-oeffentlich/informationstechnologie/cidds-coburg-intrusion-detection-data-sets.html (accessed on 19 February 2022).

[16] Datasets|Research|Canadian Institute for Cybersecurity|UNB. Available online: https://www.unb.ca/cic/datasets/index.html (accessed on 19 February 2022).

[17] Khatib, A.; Hamlich, M.; Hamad, D. Machine Learning based Intrusion Detection for Cyber-Security in IoT Networks. E3S Web Conf. 2021, 297, 01057.

[18] Rashid, M.M.; Kamruzzaman, J.; Hassan, M.M.; Imam, T.; Gordon, S. Cyberattacks detection in iot-based smart city applications using machine learning techniques. Int. J. Environ. Res. Public Health 2020, 17, 9347.

[19] Alrashdi, I.; Alqazzaz, A.; Aloufi, E.; Alharthi, R.; Zohdy, M.; Ming, H. AD-IoT: Anomaly detection of IoT cyberattacks in smart city using machine learning. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC, Las Vegas, NV, USA, 7–9 January 2019; pp. 305–310.

[20] R. Patil Rashmi, Y. Gandhi, V. Sarmalkar, P. Pund and V. Khetani, "RDPC: Secure Cloud Storage with Deduplication Technique," 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 2020, pp. 1280-1283, doi: 10.1109/I-SMAC49090.2020.9243442.

[21] Khetani, V. ., Gandhi, Y. ., Bhattacharya, S. ., Ajani, S. N. ., & Limkar, S. . (2023). Cross-Domain Analysis of ML and DL: Evaluating their Impact in Diverse Domains. International Journal of Intelligent Systems and Applications in Engineering, 11(7s), 253–262.

[22] Verma, P.; Dumka, A.; Singh, R.; Ashok, A.; Gehlot, A.; Malik, P.K.; Gaba, G.S.; Hedabou, M. A Novel Intrusion Detection Approach Using Machine Learning Ensemble for IoT Environments. Appl. Sci. 2021, 11, 10268.

[23] IDS 2018|Datasets|Research|Canadian Institute for Cybersecurity|UNB. Available online: https://www.unb.ca/cic/datasets/ids-2018.html (accessed on 21 February 2022).

[24] Arora, P.; Kaur, B.; Teixeira, M.A. Evaluation of Machine Learning Algorithms Used on Attacks Detection in Industrial Control Systems. J. Inst. Eng. India Ser. B 2021, 102, 605–616.

[25] Mothukuri, V.; Khare, P.; Parizi, R.M.; Pouriyeh, S.; Dehghantanha, A.; Srivastava, G. Federated Learning-based Anomaly Detection for IoT Security Attacks. IEEE Internet Things J. 2022, 9, 2545–2554.

[26] Frazão, I.; Abreu, P.H.; Cruz, T.; Araújo, H.; Simões, P. Denial of Service Attacks: Detecting the Frailties of Machine Learning Algorithms in the Classification Process. Lect. Notes Comput. Sci. 2018, 11260, 230–235.

[27] Wheelus, C.; Zhu, X. IoT Network Security: Threats, Risks, and a Data-Driven Defense Framework. IoT 2020, 1, 259–285.