Advances in Nonlinear Variational Inequalities
ISSN: 1092-910X
Vol 28 No. 3s (2025)

An Ensemble Model for Cyber Attack and Threat Detection in Applications Network Using Random Forest, Lightgbm and Xgboost

Ms. Nidhi[1], Dr. Sachin Kadam[2], Dr Milind Gayakwad[3*]
Gurpreet Kaur[4], Ramesh Patnaik[5], Anand Moharikar[6], Asha Pandit Ghodake[7]

[1]Assistant Professor, Bharati Vidyapeeth Institute of Management and Information Technology, Navi Mumbai. nidhi.poonia@bharatividyapeeth.edu

[2]Professor, Institute of Management and Entrepreneurship Development, Bharati Vidyapeeth (Deemed to be University), Pune (India). sachin.a.kadam@bharatividyapeeth.edu

[3]Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune-411043, India. mdgayakwad@bvucoep.edu.in

[4]Assistant Professor, Hierank business school, CCSU Meerut, India. mtech.gurpreetkaur@gmail.com

[5]Program Manager Merck Group Life Science, India. Ramesh.pattnaik@merckgroup.com

[6]Technical Project Manager, Persistent Systems, anand_moharikar@persistent.com

[7]Dept of Electronics and Telecommunications Engineering, Bharati Vidyapeeth College Of Engineering, Navi Mumbai asha.p.ghodake@gmail.com

Article History:

Received: *10-10-2024*

Revised: *27-11-2024*

Accepted: *09-12-2024*

Abstract:

Introduction: In the modern digital age, the increasing sophistication of Cyber-attacks and threats jeopardize the integrity and security of networks, systems, and sensitive data. Traditional methods of cyber threat detection, primarily based on predefined signatures, struggle to identify novel or evolving attacks, making organizations vulnerable to breaches. This research proposes a machine learning-based approach to enhance cyber-attack detection by leveraging network traffic analysis. The system utilizes Random Forest, XGBoost, and LightGBM algorithms to categorize network behaviors as either benign or harmful by analyzing traffic patterns. By analyzing correlations between multiple network variables, the proposed solution aims to detect cyber threats on web applications in real-time, thus improving the accuracy and efficiency of cybersecurity measures. To address the challenge of unavailable network standards in datasets this paper explores the UNSW-NB15 dataset which integrates real-world normal network traffic with simulated contemporary attack activities. In the paper, evaluation of the different machine learning algorithms LightGBM, Random Forest, and XGBoost for detecting cyber-attacks and threats in applications from real network traffic datasets and the result performance of all three algorithms are compared which will help improve didgital security.

Objectives: To analyze the effectiveness of Random Forest, XGBoost and LightGBM algorithms in identifying various types of attacks and enhance the security of digital networks by identifying correlations between network features and attack patterns

Methods: *T*he modal analyses the network and checks whether the traffic is harmful to the user by using XGBoost, Random Forest, and LightGBM algorithms for categorization.

Results: The conclusion of the research shows that the accuracy of Random Forest, XGBoost, and LightGBM algorithms are 0.98096, 0.98045, and 0.98023 respectively.

Conclusions: The Random Forest algorithm outperformed the other models. The modal

classification of cyber-attacks like DOS and Fuzzers are giving the best results, this possible with the help of Machine Learning Algorithms and their combinations.

Keywords: Network threats, UNSW-NB15, Random Forest, XGBoost, LightGBM, Attack Detection, network cyber -attacks.

1. Introduction

The fast development of the digital era also leads to increased cyber security threats and it has a significant effect on the truthfulness of networks, systems, and the information and data it contains. Safety measures for digital data from critical attacks are required and detecting them before it happen is very crucial[1][2]. To fight these threats on the network it is decisive to develop advanced methodologies for early detection and prevention of cyber-attacks. This paper focuses on enhancing cybersecurity in web applications and their networks by applying machine learning algorithms to detect various types of cyber threats [3][4]. For analyzing the network traffic data and identifying possible attacks in real-time in the research we used Random Forest, XGBoost, and LightGBM models which have proven activeness in identifying attack patterns and different types of attacks that are possible, leading to quicker responses and improved security measures [5][6]. Different types of attack can apply on networks like Distributed Denial of Service (DDoS), malware infections, fuzzers, and worms and we require security for our digital data, in this case, Machine learning plays a vital role in detecting patterns which can lead to attacks and recognize them [7] [8]. Old conventional methods fail to detect the attacks as per advancing technology whereas in ML approach detects the behavior quickly and more accurately [9] [10]. The rising number of attacks and threats in the network requires new techniques and methods to detect them early so that mitigation techniques can be applied to detect vulnerabilities on time [11][12][13]. Present methods struggle to keep up with the growing threats and lead to negative impacts on organizations, and users so early detection is a requirement [14][15].

Literature Survey

Zaman and team applied seven different machine learning algorithms on the Kyoto 2006+ dataset, recognized the intrusion in the network, and assessed the methods based on different machine learning parameters [16]. In the paper the researcher applied different ML algorithms like K-Nearest Neighbors (KNN), Fuzzy C-Means, Support Vector Machine, Radial Basis Function, and an ensemble method for detecting threats in the network [17][18]. This paper's findings provided a comparative analysis of different techniques for addressing security challenges in network environments and intrusion detection systems [19][20]. In the paper researcher used decision trees as a deterministic alternative to the stochastic nature of genetic algorithms in detection [21][22]. In the paper researcher apply a combination of different algorithms to enhance the system's consistency and decision-making capabilities and in the finding decision Tree algorithm works well to find the main root feature or attribute of the data set [23][24]. Nagar implemented an ensembled intrusion detection system (IDS) to improve the detection accuracy of single ML classifiers, particularly KNN, which struggles to effectively identify minority classes in the CICIDS dataset [25][26][27]. The proposed ensemble model integrates three base classifiers—KNN and Random Forest—into a unified framework designed for enhanced multi-class attack classification [28]. This approach addresses a critical limitation of

single ML classifiers, namely their weak performance in detecting diverse types of intrusions. Jing highlighted the advantages of the UNSW-NB15 dataset over the older KDDCUP99 and NSL-KDD datasets [29][30]. The UNSW-NB15 dataset incorporates more contemporary abnormal behaviors and normal activities observed over time, along with detailed network traffic features [31][32]. This complexity ensures a more reliable evaluation of network attack detection, making it a preferred choice for modern intrusion detection research. Moustafa emphasized the strengths of the UNSW-NB15 dataset, which represents nine major attack categories generated using the PerfectStorm tool [33][34]. The dataset includes 49 features extracted by Bro-IDS tools, along with twelve algorithms that comprehensively characterize network packets [35]. Compared to older standard datasets such as KDD98, KDDCUP99, and NSL-KDD, UNSW-NB15 addresses their limitations by offering a broader variety of attacks and more detailed, up-to-date packet information. Meftah pointed out certain limitations of the UNSW-NB15 dataset, noting that some attack categories—such as Backdoor, Analysis, Shellcode, and Worms—cannot be effectively trained or tested due to insufficient records or the lack of necessary features in the dataset [36][37]. These gaps highlight the need for enhancements to ensure comprehensive coverage of all advertised attack types [39]. Farnaaz highlighted Random Forest (RF) as an ensemble-based classification method that improves accuracy by combining the predictions of multiple decision trees [40]. Compared to traditional classifiers, RF achieves lower classification errors. Its performance is influenced by several key parameters, including the number of trees in the forest and the minimum size of nodes. Sommer emphasized that machine learning should be viewed as a tool for deriving insights rather than as the ultimate solution in intrusion detection [8]. Its primary value lies in analyzing the significance of various features distinguishing benign from malicious activities [41]. These insights can then form the foundation for developing more effective, non-machine-learning-based detection systems [42].

Methods

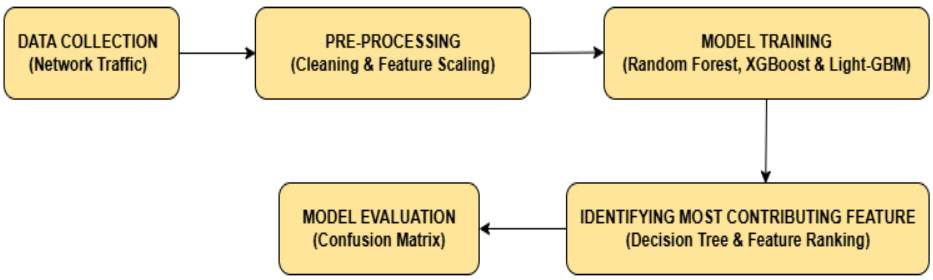

Fig. No. 1: Model

The overall detailed process or methodology of this study is described below in Figure 1:

Data Collection Process

Environment: Data was collected in a real-world testbed environment using the IXIA PerfectStorm tool.

Traffic Generation: Both normal and malicious traffic were generated.

Normal Traffic: Simulated typical activities like web browsing, email communication, file transfer, and database access.

Malicious Traffic: Composed of 9 attack types, including modern and legacy attack techniques.

Packet Capture: Network traffic was captured and processed using tools to extract 49 key features from the raw data.

The dataset captures attack scenarios that mimic contemporary threat vectors, ensuring that machine learning models trained on this data are relevant to current cybersecurity challenges and Table 1 represents the attack categories.

Table 1: Attack Categories

Attack Category	Description
Fuzzers	Sending malformed inputs to test vulnerabilities in systems.
Analysis	Techniques like packet analysis, port scanning, and IP sweeps.
Backdoor	Exploiting systems to gain unauthorized remote access.
DoS	Overloading systems with traffic to disrupt services.
Exploits	Taking advantage of software vulnerabilities to execute malicious code.
Generic	Cryptographic attacks and brute-force attempts on secure protocols.
Reconnaissance	Gaining information about systems through scanning and other means.
Shellcode	Injecting and executing malicious payloads.
Worms	Self-replicating malware spreading through networks.

Data Volume and Composition

The dataset contains more than 2.5 million records. It includes both normal and attack traffic, making it well-suited for binary and multiclass classification tasks.

Table 2: Traffic Types

Traffic Type	Count	Percentage
Normal	22,19,905	87.37%
Attack	3,20,239	12.63%
Total	25,40,144	100%

Feature Categories

The dataset includes 49 features, which can be categorized as shown in Table 3.

Table 3: Features table

Feature Name	Type	Description	Feature Name	Type	Description
srcip	Nominal	IP address of the source packet.	**sttl**	Continuous	Time-to-live value of packets from the network the source to the destination.
sport	Nominal	Source port number.	**dttl**	Continuous	Time-to-live value of packets from the network the destination to the source.

dstip	Nominal	Destination IP address of the packet.	**sloss**	Continuous	Number of packets lost from source of network to destination.
dsport	Nominal	Destination port number.	**dloss**	Continuous	Number of packets lost from destination of network to source.
proto	Nominal	Protocol used	**service**	Nominal	Service flow
state	Nominal	Connection state (e.g., ESTABLISHED, FIN_WAIT).	**Sload**	Continuous	Source network-to-destination bits per second.
dur	Continuous	Duration of the flow (in seconds).	**Dload**	Continuous	Destination network-to-source bits per second.
sbytes	Continuous	Bytes sent from network source to destination.	**Spkts**	Continuous	Number of packets sent from source data to destination network .
dbytes	Continuous	Bytes sent from network destination to source.			

Table 4 describes the types of attack and Training and Testing split for all the categories

Table 4: Category-wise training, testing split

Attack Category	Attack Training Set	Attack Testing Set
Backdoor Attack	1,746	583
Reconnaissance Attack	10,491	3,496
Normal Attack	56,000	37,000
DOS Attack	12,264	4,089
Exploits Attack	33,393	11,132
Worms Attack	130	44
Generic Attack	40,000	18,871
Analysis Attack	2,000	677
Shellcode Attack	1,133	378
Fuzzing Attack	18,184	6,062
Total Records Attack	1,75,341	82,332

Visualization of the dataset after some data cleaning procedure is shown in Figure 2.

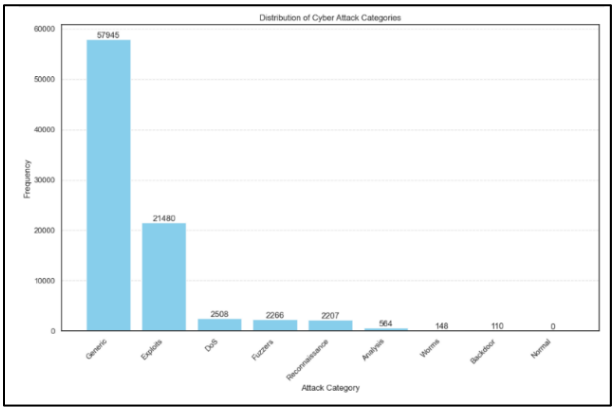

Fig. 2. Visualization of Attack Categories

The given visualization represents Normal category as zero as during this study it was required to remove all the records with 0 duration and unknown state, as a part of data preprocessing.

Pre-processing (Cleaning, Feature Scaling) -

The collected data is cleaned to remove noise and irrelevant features. It involves:

removing records with 0 duration and unknown state

Feature Scaling.

Basic Rule Visualization: This visualization depicts the decision-making process of a classification model, likely a decision tree.

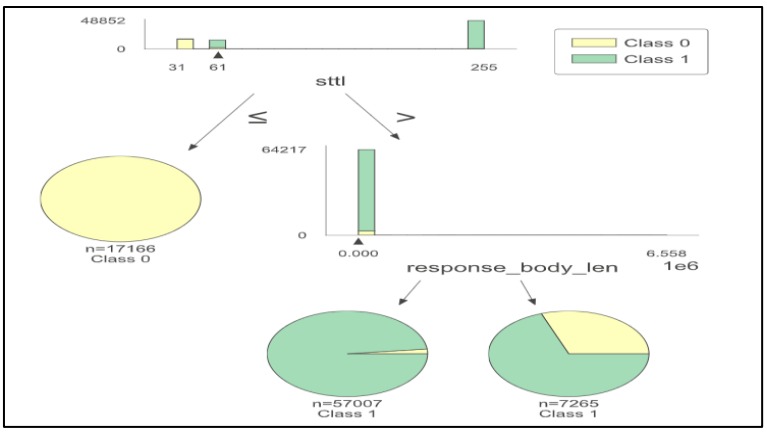

Fig. 3. Data Split & Rule Visualization

Figure 3. shows how the data is split across different features to separate two classes, Class 0 and Class 1. The root node starts by splitting the data based on the sttl feature, using the threshold ≤ 64217 to create two branches. The left branch leads to a node dominated by Class 0, while the right branch splits further based on the response_body_len feature. Each split results in new nodes, with pie charts indicating the distribution of Class 0 (yellow) and Class 1 (green) samples. The size of each segment corresponds to the number of samples in that category, as labeled (e.g., n=17166). The visualization effectively demonstrates how features and thresholds contribute to the classification process, highlighting the flow of data and the rules applied to separate the two classes.

Experiments and processes on data using all the models can be visualized as shown in Figure 4.

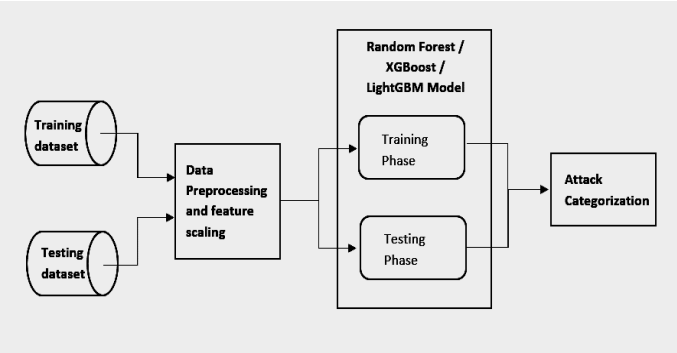

Fig. 4. Visualization of working of the models

Random Forest Model for Categorization: **Random Forest** is an ensemble learning method that builds multiple decision trees during training and aggregates their outputs to make predictions. Its primary advantage lies in its robustness against overfitting, particularly when handling noisy or imbalanced datasets. For the attack categorization random forest works with 98% accuracy, the model evaluation is represented in Figure 5.

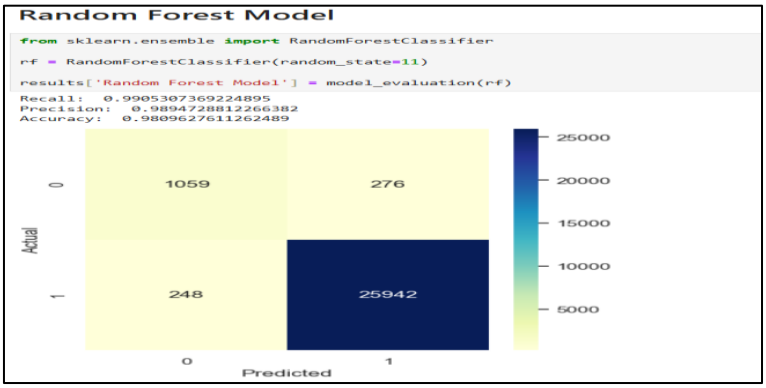

Fig. 5. Random Forest Model Evaluation

Feature Ranking from Random Forest:

Table 5: Feature Importance Table

Name	Importance
Sttl	0.210799
ct_state_ttl	0.138904
Dttl	0.081721
Dload	0.059493
Dbytes	0.048407
Dmean	0.040825
ackdat	0.038555
dpkt	0.037226
tcprtt	0.034609
dur	0.034318

Table 5 shows the relative feature importance of different attributes, ranked by their contribution to the model's predictive ability. Here's a breakdown of the key aspects:

Most Important Features:

sttl (importance: 0.210799): This feature has the highest impact on the model's performance.

Moderately Important Features:

Features like dttl (0.081721), dload (0.059493), and dbytes (0.048407) have a moderate influence.

Less Important Features:

dur (0.034318) and tcprtt (0.034609) contribute the least.

XGBoost Model for Categorization: **XGBoost (Extreme Gradient Boosting)** builds trees sequentially, optimizing a loss function while incorporating regularization to prevent overfitting. It is highly efficient, leveraging advanced techniques such as parallelization and sparsity awareness, which make it well-suited for large datasets. For the attack categorization, XGBoost also works with 98% accuracy, the model evaluation is as represented in the given in Figure 6.

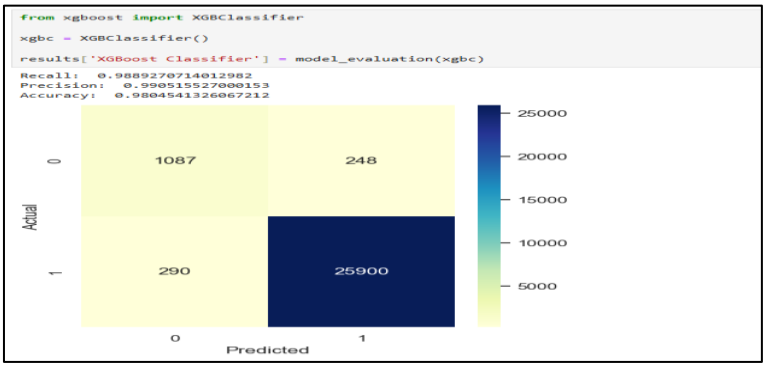

Fig. 6. XGBoost Model Evaluation

LightGBM Model for Categorization: **LightGBM (Light Gradient Boosting Machine)** stands out for its exceptional speed and memory efficiency. It employs histogram-based algorithms and leaf-wise tree growth, allowing it to process large-scale datasets quickly. For the attack categorization, LightGBM also works with 98% accuracy, the model evaluation is as represented in the given in Figure 7.

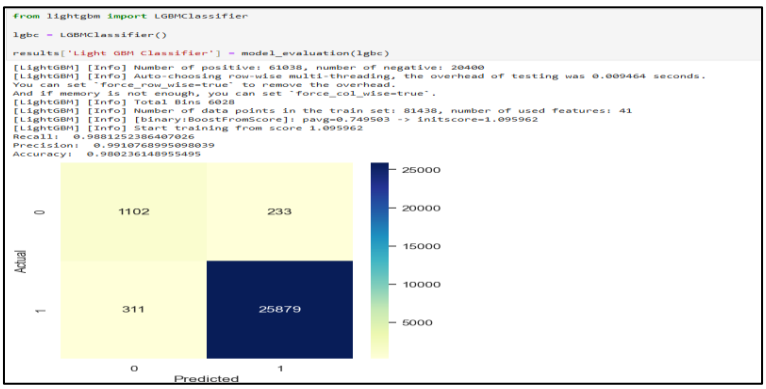

Fig. 7. LightGBM Model Evaluation

Method validation

To evaluate the performance of our model, we utilize the confusion matrix, which allows us to identify true positives, true negatives, false positives, and false negatives.

Below is the representation of the confusion matrix of some of the attack categories of our study.

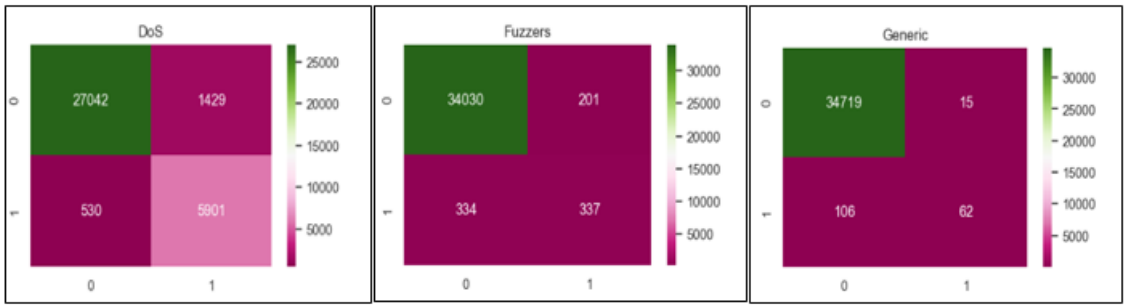

Fig. 8. Confusion Matrix for DOS Attack, Fuzzer Attack and Generic type

From Figure 8 analysis is drawn that our model performs reasonably well for the 'DOS' class, 'Fuzzers' class and Generic type

The comparative evaluation of Random Forest, XGBoost, and LightGBM for attack categorization using the UNSW-NB15 dataset is given in Table 6, it highlights the strengths of each model across key metrics: Recall, Precision, and Accuracy.

Table 6 Comparative Evaluation of Models

Algorithm	Recall	Precision	Accuracy
Random Forest	0.99053	0.98947	0.98096
XGBoost	0.98892	0.99051	0.98045
LightGBM	0.98812	0.99107	0.98023

Random Forest exhibited the highest Recall, demonstrating its ability to identify the majority of attack instances effectively, making it particularly suitable for scenarios where minimizing false negatives is critical, such as intrusion detection systems aimed at capturing as many attacks as possible.

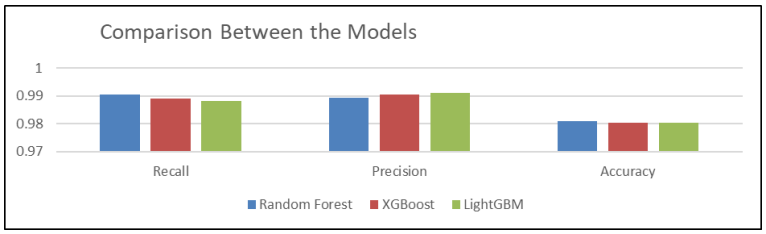

Fig. 9. Recall, Precision, and Accuracy score metrics of Models

Figure 9. shows LightGBM, on the other hand, achieved the highest Precision, reflecting its effectiveness in minimizing false positives. This makes it an ideal choice for environments where reducing false alarms is paramount, such as automated response systems that could be disrupted by frequent false positives. In the result from the analysis of different ML algorithms Random Forest gives best result in terms of accuracy for detecting intrusion in the network.

In terms of Accuracy, all three models performed similarly, with Random Forest and XGBoost showing slightly better results than LightGBM. Precision-wise for categorizing the attacks in the network LightGBM gives the best results. According to (High Recall) Random Forest gives better results for attack and threat detection. From the result, we can analyze that different modal or algoritmns can be used according to requirements.

CONCLUSION

The results show similar accuracy and recall on our Dataset for attack detection and modal excel in detecting attack types like **DOS** and **Fuzzers**, they may need further refinement to handle other attack types such as **Generic** class, more effectively. Future improvements could involve augmenting the training dataset with a wider variety of attack types or fine-tuning the model to better generalize across different kinds of attacks. Additionally, investigating methods to reduce false positives in more challenging attack classes would contribute to enhancing the model's overall performance and usefulness in real-world problems. In conclusion, our model shows promising results for some of the attack categories, but there is room for improvement, particularly in handling other remaining attack types, to accomplish more robust and reliable intrusion/ attack detection.

References

[1] M. Zaman and C. -H. Lung, "Evaluation of machine learning techniques for network intrusion detection," NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 2018, pp. 1-5.

[2] C. Sinclair, L. Pierce and S. Matzner, "An application of machine learning to network intrusion detection," Proceedings 15th Annual Computer Security Applications Conference (ACSAC'99), Phoenix, AZ, USA, 1999, pp. 371-377.

[3] Nagar, U. (2021). A study on feature analysis and ensemble-based intrusion detection scheme using CICIDS-2017 dataset (Doctoral dissertation, University of Technology Sydney (Australia)).

[4] Jing, D., & Chen, H. B. (2019, October). SVM based network intrusion detection for the UNSW-NB15 dataset. In 2019 IEEE 13th international conference on ASIC (ASICON) (pp. 1-4). IEEE.

[5] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, ACT, Australia, 2015, pp. 1-6.

[6] Meftah, S., Rachidi, T., & Assem, N. (2019). Network based intrusion detection using the UNSW-NB15 dataset. International Journal of Computing and Digital Systems, 8(5), 478-487.

[7] Farnaaz, N., & Jabbar, M. A. (2016). Random forest modeling for network intrusion detection system. Procedia Computer Science, 89, 213-217.

[8] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," 2010 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 2010, pp. 305-316.

[9] https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15/data

[10] Dr. Milind Gayakwad, Prof. Shrikala Deshmukh,Dr. Nisha Auti, Prof.Renuka Amit Mane , Dr. Priyanka Paygude, Dr Rahul Joshi, Dr. Kalyani Kadam(2024) The Analysis of The Daily Return Percentage as An Alternative To The Closing Price Of The Stock Using The Ensemble Model.Library Progress International, 44(3), 16789-16799.

[11] Gajanan V. Bhole, Prakash Devale, Ashwini Khairkar , Nisha Auti , Shrikala Deshmukh, Milind Gayakwad, Rahul Joshi (2024). Automated Web Service Discovery And Computing Approaches And Methods , Library Progress International, 44(3), 11590-11602.

[12] Paygude, Priyanka, Sandip Thite, Ajay Kumar, Amol Bhosle, Rajendra Pawar, Renuka Mane, Rahul Joshi, Manisha Kasar, Prashant Chavan, and Milind Gayakwad. "A Dataset Revolutionizing Indian Bay Leaf Analysis." Data in Brief (2024): 111024.

[13] S. Deshumkh, M. Gayakwad, N. S. More, R. Jadhav, K. Kadam and H. Magar, "Smart Traffic Management System Using RFID System," 2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon), Pune, India, 2024, pp. 1-4, doi: 10.1109/MITADTSoCiCon60330.2024.10575670.
[14] S. Deshmukh, S. Chaudhary, M. Gayakwad, K. Kadam, N. S. More and A. Bhosale, "Advances in Facial Emotion Recognition: Deep Learning Approaches and Future Prospects," 2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon), Pune, India, 2024, pp. 1-3, doi: 10.1109/MITADTSoCiCon60330.2024.10574908.
[15] Khatik, I., Kadam, S., Gayakwad, M., Joshi, R., & Kotecha, K. (2024). Automatic Diagnosis of Fracture using Deep Learning and External Validation: A Systematic Review and Meta-Analysis. International Journal of Intelligent Systems and Applications in Engineering, 12, 41-48.
[16] Kadam, A., B. Garg, M. Gayakwad, K. Kotecha, and R. Joshi. "Novel DSIDS-Deep Sniffer Intrusion Detection System." International Journal of Intelligent Systems and Applications in Engineering 12 (2024): 400-407.
[17] Beldar, Miss Menka K., M. D. Gayakwad, Miss Kavita K. Beldar, and M. K. Beldar. 2018. “Survey on Classification of Online Reviews Based on Social Networking.” IJFRCSCE 4 (3): 55.
[18] Boukhari, Mahamat Adam, Prof Milnid Gayakwad, and Prof Dr Suhas Patil. 2019. “Survey on Inappropriate Content Detection in Online Social Media.” International Journal of Innovative Research in Science, Engineering and Technology 8 (9): 9297–9302.
[19] Gayakwad, M. D., and B. D. Phulpagar. 2013. “Research Article Review on Various Searching Methodologies and Comparative Analysis for Re-Ranking the Searched Results.” International Journal of Recent Scientific Research 4: 1817–20.
[20] Gayakwad, Milind. 2011. “VLAN Implementation Using IP over ATM.” Journal of Engineering Research and Studies 2 (4): 186–92.
[21] Gayakwad, Milind, and Suhas Patil. 2020. “Content Modelling for Unbiased Information Analysis.” Libr. Philos. Pract, 1–17.K. Boyat and B. K. Joshi, “A Review Paper: Noise Models in Digital Image Processing,” arXiv:1505.03489 [cs], May 2015.
[22] Omarov, Batyrkhan Sultanovich, et al., "Exploring Image Processing and Image Restoration Techniques," International Journal of Fuzzy Logic and Intelligent Systems, vol. 15, no. 3, pp. 172-179, June 2015.
[23] Gayakwad, Milind, Suhas Patil, Rahul Joshi, Sudhanshu Gonge, and Sandeep Dwarkanath Pande. “Credibility Evaluation of User-Generated Content Using Novel Multinomial Classification Technique.” International Journal on Recent and Innovation Trends in Computing and Communication 10 (2s): 151–57.
[24] Rajendra Pawar et al.,“ Farmer Buddy-Plant Leaf Disease Detection on Android Phone” In International Journal of Research and Analytical Reviews. Vol 6 (2), 874-879
[25] Gayakwad, Milind, Suhas Patil, Amol Kadam, Shashank Joshi, Ketan Kotecha, Rahul Joshi, Sharnil Pandya, et al. 2022. “Credibility Analysis of User-Designed Content Using Machine Learning Techniques.” Applied System Innovation 5 (2): 43.
[26] Harane, Swati T., Gajanan Bhole, and Milind Gayakwad. 2017. “SECURE SEARCH OVER ENCRYPTED DATA TECHNIQUES: SURVEY.” International Journal of Advanced Research in Computer Science 8 (7).
[27] Kavita Shevale, Gajanan Bhole, Milind Gayakwad. 2017. “Literature Review on Probabilistic Threshold Query on Uncertain Data.” International Journal of Current Research and Review 9 (6): 52482–84
[28] Mahamat Adam Boukhari, Milind Gayakwad. 2019. “An Experimental Technique on Fake News Detection in Online Social Media.” International Journal of Innovative Technology and Exploring Engineering (IJITEE) 8 (8S3): 526–30.
[29] Maurya, Maruti, and Milind Gayakwad. 2020. “People, Technologies, and Organizations Interactions in a Social Commerce Era.” In Proceeding of the International Conference on Computer Networks, Big Data and IoT (ICCBI-2018), 836–49. Springer International Publishing.
[30] Milind Gayakwad, B. D. Phulpagar. 2013. “Requirement Specific Search.” IJARCSSE 3 (11): 121.
[31] Panicker, Aishwarya, Milind Gayakwad, Sandeep Vanjale, Pramod Jadhav, Prakash Devale, and Suhas Patil. n.d. “Fake News Detection Using Machine Learning Framework.”
[32] Gonge, S. et al. (2023). A Comparative Study of DWT and DCT Along with AES Techniques for Safety Transmission of Digital Bank Cheque Image. In: Chaubey, N., Thampi, S.M., Jhanjhi, N.Z., Parikh, S., Amin, K. (eds) Computing

Science, Communication and Security. COMS2 2023. Communications in Computer and Information Science, vol 1861. Springer, Cham. https://doi.org/10.1007/978-3-031-40564-8_6

[33] Self-Driving Electrical Car Simulation using Mutation and DNN Paygude, P. Idate, S. Gayakwad, M. Kadam, K. Shinde, A. SSRG International Journal of Electronics and Communication Engineering, 2023, 10(6), pp. 27–34

[34] Probing to Reduce Operational Losses in NRW by using IoT Hingmire, S. Paygude, P. Gayakwad, M. Devale, P. SSRG International Journal of Electronics and Communication Engineering, 2023, 10(6), pp. 23–32

[35] Paygude, P.., Singh, A.., Tripathi, E.., Priya, S.., Gayakwad, M.., Chavan, P.., Chaudhary, S.., Joshi, R.., & Kotecha, K.. (2023).

[36] A Parameter-Based Comparative Study of Deep Learning Algorithms for Stock Price Prediction. International Journal on Recent and Innovation Trends in Computing and Communication, 11(7s), 138–146. https://doi.org/10.17762/ijritcc.v11i7s.6985

[37] Dixit, B., Pawar, R. G., Gayakwad, M., Joshi, R., & Mahajan, A. (2023). Challenges and a Novel Approach for Image Captioning Using Neural Network and Searching Techniques. International Journal of Intelligent Systems and Applications in Engineering, 11(3), 712-720.

[38] Godse, D. ., Mulla, N. ., Jadhav, R. ., Gayakwad, M. ., Joshi, R. ., Kadam, K. ., & Jadhav, J. . (2023). Automated Video and Audio-based Stress Detection using Deep Learning Techniques. International Journal on Recent and Innovation Trends in Computing and Communication, 11(11s), 487–492. https://doi.org/10.17762/ijritcc.v11i11s.8178

[39] Paygude, P. ., Chavan, P. ., Gayakwad, M. ., Gupta, K. ., Joshi, S. ., Gopika, G., Joshi, R., Gonge, S., & Kotecha, K. . (2023). Optimizing Hyperparameters for Enhanced LSTM-Based Prediction System Performance. International Journal on Recent and Innovation Trends in Computing and Communication, 11(10s), 203–213. https://doi.org/10.17762/ijritcc.v11i10s.7620

[40] Bhole, G. V., et al. "Implementation of Virtual Mouse Control System Using Hand Gestures for Web Service Discovery." International Journal of Intelligent Systems and Applications in Engineering 12.13s (2024): 663-672.

[41] Bytyqi, V., & Rexha, B. (2024, March). Machine Learning Boosted Trees Algorithms in Cybersecurity: A Comprehensive Review. In Future of Information and Communication Conference (pp. 158-173). Cham: Springer Nature Switzerland.

[42] Alenazi, M., & Mishra, S. (2024). Cyberatttack Detection and Classification in IIoT systems using XGBoost and Gaussian Naïve Bayes: A Comparative Study. Engineering, Technology & Applied Science Research, 14(4), 15074-15082.Gayakwad, Milind (2024), “White Fragrant Flowers Western Region of India”, Mendeley Data, V1, doi: 10.17632/yr23h7x8dv.1.