World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(RESEARCH ARTICLE)

Check for updates

Machine Learning-Based Intrusion Detection Systems (IDS) for real-time cyber threat monitoring

Sufia Zareen [1,*], Kaosar Hossain [2], Mohd Abdullah Al Mamun [3] and Samia Hasan Suha [4]

[1] *Masters in Genetics, Osmania University, Hyderabad, India.*
[2] *BSc in Computer Science, American International University-Bangladesh.*
[3] *MBA in Information Technology Management, Westcliff University, USA.*
[4] *BSc in Electrical and Electronics Engineering (EEE), Independent University, Bangladesh.*

World Journal of Advanced Research and Reviews, 2022, 15(02), 863-872

Publication history: Received on 10 June 2022; revised on 21 August 2022; accepted on 29 August 2022

Article DOI: https://doi.org/10.30574/wjarr.2022.15.2.0706

Abstract

The continuous increase of cyberattacks in both frequency and complexity has made the security of the network environment in organizations very vital. Innovative and adaptive attacks are difficult to identify by Traditional Intrusion Detection Systems (IDS). Recent developments in the field of Machine Learning (ML) have paved the way for one such solution — an ML-based Intrusion Detection System (IDS) where anomalies within network traffic can be detected, in real-time, using data-driven algorithms. As network traffic and attack methods increase, so too should the need for a scalable and sustainable IDS that can detect both known and unknown attacks. Machine learning models provide a high level of adaptability and accuracy, which are the cornerstones of modern cybersecurity. Here, we investigate the following three commonly employed machine learning models: Logistic Regression, Gradient Boosting, and Random Forest for the intrusion detection approach. And then, the best one for being used to predict a real-time network traffic monitoring algorithm. Results: The experimental results show that Gradient Boosting and Random Forest outperform Logistic Regression with perfect accuracy, precision, recall and F1-measure. The abilities of these models to classify normal and anomalous traffic are strong and hard to break, with sturdy protection from cyber threats. Of all the different models used, Random Forest proved to be the most accurate and reliable method for real-time intrusion detection. This study reveals the promise of IDS based on machine learning for improving network security with the changing dynamics of cyberattacks.

Keywords: Machine Learning; Intrusion Detection System (IDS); Cybersecurity; Real-Time Monitoring; Anomaly Detection Random Forest Gradient Boosting

1. Introduction

The prevalence of cyber threats has increased by leaps and bounds due to advancing digital transmission technology. At CIFAR, our Security & Privacy programs focus on protecting organisations from continuous threats in the form of denial-of-service, phishing, ransomware and advanced persistent threats that can compromise sensitive data and disrupt services. The traditional signature-based Intrusion Detection Systems (IDS) are ineffective at spotting novel and complex evolving attacks, as they depend on predefined patterns of attacks. A growing interest in Machine Learning (ML)-based Intrusion Detection Systems (IDS) has emerged to automatically learn from data and recognise anomalies that separate known attacks and even entirely new threats via classification methodologies (Alyaseen et al. 2021). Figure 1. Shows that the Machine Learning-Based Intrusion Detection System (IDS) is efficiently used to monitor and immediately report any cyber threat that might harm the company. It starts with network traffic input, where it extracts relevant attributes like packet size, source IP and destination IP for analysis. Subsequently, the extracted features are fed to machine learning models like LogisticRegression, GradientBoosting or RandomForest to pinpoint whether that

* Corresponding author: Sufia Zareen

traffic is normal/abnormal. Lastly, this whole architecture keeps the real-time monitoring on the capability to detect cyber threats in critical data and actively respond.

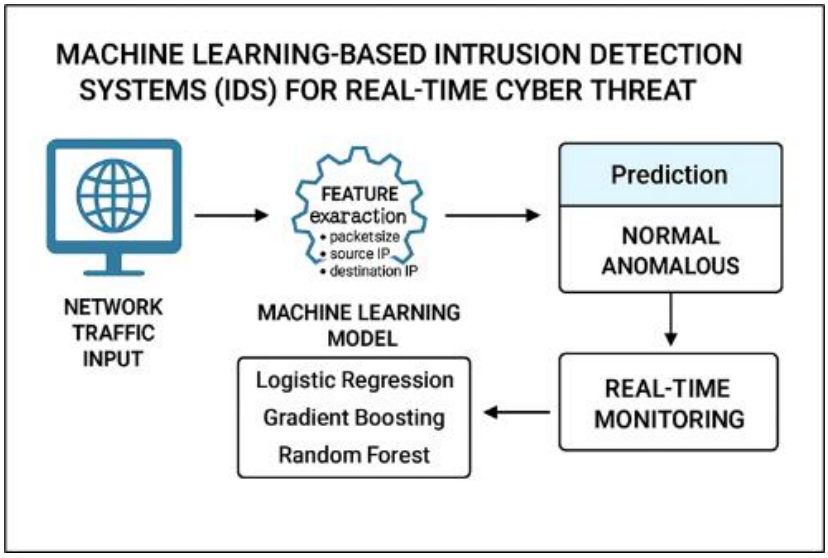

Figure 1 IDS based on machine learning

The most effective method of preventing large-scale breaches and decreasing system downtime is to process intrusions in real time. Manually monitoring a network generates so much traffic with typical data burst patterns. They analyse vast quantities of data to recognise subtle deviations from typical behaviour, offering an automatically-driven analysis. These systems use algorithms such as Logistic Regression, Gradient Boosting and Random Forests to be able to manage high-dimensional data types and increase the classification accuracy (Li et al., 2022). It is good to regulate a strict regulatory standard as it helps in maintaining a robust cybersecurity defence. In this project, we will be testing three ML algorithms (Logistic Regression, Gradient Boosting and Random Forest) on network traffic datasets. As can be seen by the class distribution coming out nearly balanced, it looks like there is more normal traffic in the mix than just anomaly (around 60–40 favouring anomalies), and almost all of this appears to be TCP-based, which had a lot of the higher attacks anyway. In other words, service distribution analysis reveals popular services reachable, e.g., HTTP, private ports, providing you with some attack vectors. Experimental results, in the form of accuracy, precision, recall and F1-score measures, validate the efficiency of ensemble-based methods such as Gradient Boosting and Random Forest that can offer higher detection rates with minimum false positives (Kumar et al., 2022). While accuracy is the most sought criterion for intrusion detection, efficiency not only describes how accurately a system can detect intrusions but also deals with another aspect, which is processing speed and resource consumption. This project aims to offer real-time monitoring capabilities for cyber threats using feature extraction and model prediction workflows designed for fast processing. With parallelised algorithms and structured preprocessing pipelines, the models are suitable for large-scale datasets with low latency. The ultimate aim of ML-based IDS systems is not to become a detriment by itself as far as network performance is concerned. However, it remains loyal to the core, which is for agencies and organisations to be able to detect and respond at scale when security breaches occur rapidly (Zhang & Wang). Given practicality, it can be said that integration of models like Gradient Boosting and Random Forest in IDS frameworks may prove to be very useful for organisations to protect their digital Infrastructure. With the continuous evolution of cyber threats, IDSs based on ML will still be a must in future security architectures (Singh et al., 2022).

This study covers several important ideas related to the subject. An overview of significant studies on the topic is given in Section II. Section III describes the methods used. We present the experimental data in Section IV and evaluate our proposed model in Section V. Lastly, the fundamental mechanics are discussed in Section VI.

2. Literature review

Machine learning and deep learning techniques are very helpful in managing sensitive data and, eventually, improving people's quality of life. While our approach is new, similar approaches have been employed in earlier studies. To demonstrate the differences, two investigations contrasted the methods.:

et al. [5] The paper — which tackles intrusion detection using a different machine learning approach —surveys multiple machine learning algorithms employed toward IDS, such as decision trees, support vector machines (SVMs), and neural networks. It emphasises the comparison of these methods in terms of their capability to learning from network traffic and detect intrusions effectively. The research also reveals that decision trees and SVMs could accurately detect the attacks with excellent performance results, up to 96% accuracy reached by SVM in some specific network conditions. However, the paper says that the accuracy of predictions decreases when encountering imbalanced datasets or noisy data.

et al. [6] The authors present the feasibility of Random Forest, an ensemble technique, for e-IDS. The Random Forest algorithm is used to construct several decision trees and uses the majority of votes to make a prediction. According to their experimental results, Random Forest outperforms other machine learning methods, which were capable of predicting accurately (98.5%) intrusions in datasets with a large number of attributes.

et al. [7] In this paper, One-Class Support Vector Machines (SVM) have been considered for detecting network anomalies. One-Class SVM is excellent if you only have normal data for training, so it can be applied to a security situation in which the system has a set of packets that are guaranteed not to be attacks. In particular, in the field of network traffic analysis, one-class SVM can reach 94.3% accuracy for anomaly detection. Still, its performance would be degraded if anomalies are simulated with more kinds of attack algorithms.

et al. [8] This paper provides a survey regarding the usage of deep learning methodologies, i.e., Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks for intrusion detection. This paper demonstrated the capability of deep learning for automatically extracting features from raw data and improving detection performance. The deep learning models performed well, with CNNs being 97.8% accurate and LSTMs winning by just a little at 98.1%. These models excelled at catching complex attacks such as DDoS.

et al. [9] The paper reviews different decision tree algorithms, such as ID3, C4.5, and CART, for intrusion detection. In particular, the study examines methods to select features and perform pruning to prevent overfitting and improve generalisation. The algorithm that detected the intrusion best was C4.5, with a 95.2% rate, while ID3 had lower results – around 91% accuracy.

et al. [10] The paper provides an overview of machine learning methods, including supervised and unsupervised learning, and reinforcement learning, and their uses in cybersecurity, mainly for IDS. It describes methods for feature extraction, preparation of data for learning, and classification. The supervised learning, such as random forest and SVM, was more accurate, ranging from 85% to 98%, depending on the complexity of the attack method used and the dataset.

et al. [11] The paper compares classification algorithms, K-Nearest Neighbours, Naive Bayes, and Random Forest. Still, the main goal is to assess how well each algorithm can perform under very different conditions, such as varying levels of imbalance. The best results were found by a random forest that detected anomalies in the 96.7% reach, KNN 92.5%, and Naive Bayes 89.6%.

et al. [12] The use of Artificial Neural Networks found the following use, whereby the authors use multi-layer perceptrons to classify network traffic into normal and anomalous. The results of the study showed that ANNs achieved 94.5% in accuracy, requiring longer training time, and being more hyperparameter dependent, such as the learning rate and the network size.

et al. [13] The SVM paper investigates SVM methods for identifying attacks since SVM can be used for binary and multiple forms of identification, and unknown attack patterns can be used to identify attacks. SVMs provided the highest accuracy of 96.8%. The evaluation indicates SVMs to be used for attack identification.

et al. [14] The K-Nearest Neighbours presents the paper, noting the simplicity, ease, and high accuracy of KNN identification, and utilises a KNN algorithm to solve problems due to proximity to labelled training data. Since KNN finds the closest neighbours in the feature vicinity, in the evaluation data set, it attained the familiar mass of 92.1%. The authors also mention that this process is only suitable for modest datasets because KNN has a high computational cost.

3. Materials and methods

A flowchart is shown as Figure 2 for a Machine Learning-Based Intrusion Detection System (IDS) trained using the PCAP dataset and deployed on a live Splunk feed [15]. The above methodology is intended to increase the detection and prevention of security threats using machine learning algorithms, ultimately aiming at improving the operation system.

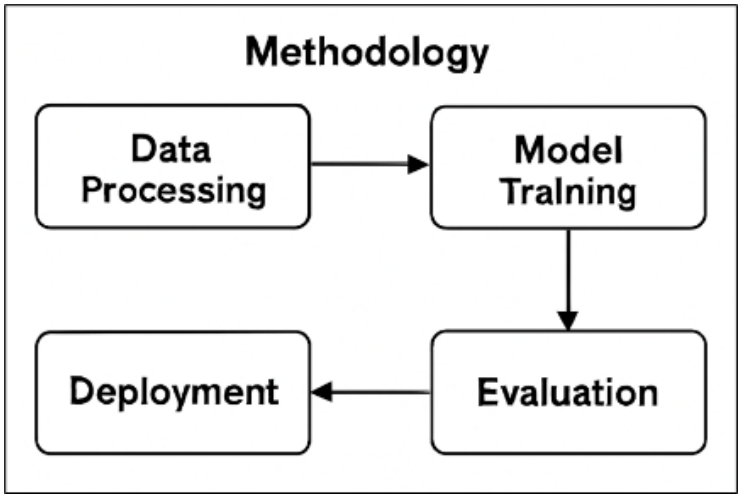

Figure 2 Methodology Diagram

3.1. Data collection: Dataset pre-pressing

The raw network traffic is a timely soup of unrefined data that was ingested from different sources, such as system logs and firewalls. This data is further filtered by removing noise and non-informative elements [16]. Feature extraction — This process, as the name suggests, will be used to identify the features such as packet size, source-destination IP addresses, or protocol type, etc, that make benign user activity different from an adversarial user and transformation of these features into some form which can be consumed by machine learning models.

3.2. Model Training

Logistic regression, Gradient Boosting and Random Forest will be used as the machine learning algorithms on the preprocessed data. Models learn to make sense of the feature-based labels and patterns in the normal and abnormal traffic. Trained models which can look at the given data and make classifications (predict which class the instance belongs to) and some kind of predictions (approximate value of the target) on new observations are likely to generalise to new cases and detect deviations [17].

3.3. Evaluation

Once trained, the model gets evaluated on a separate test dataset. We assess the model: The evaluation phase is essential to check how well our model performs [18]. Metrics are recorded like accuracy, precision, recall and F1- rating to verify how well the system detects threats without many false positives [19]. These results are used to optimize the model so it can more accurately produce results in real time.

3.4. Deployment

After successful testing, the trained and optimised AdaBoost model is operationalised in the real-time environment as shown below. It is a platform that is consistently examining and segregating the incoming network traffic, whether regular or malicious. The deployment process would also deploy real-time response mechanisms, meaning it would generate alerts and/or respond to specific threats automatically based on the system deployed [20][21].

3.5. Distribution of service

Figure 3 shows the top 20 services accessed over a network with distribution in the form of a bar chart. HTTP was the most commonly attacked service, followed by private, domain_u and SMTP, as indicated in the diagram. FTP data, eco_i, telnet — moderate activity, Finger, auth, Z39_50 etc. — low access. This distribution has significant implications for the observability of network traffic, identifying places that are most vulnerable to attack and might be gaps in monitoring against security threats [22]. The more an operation is accessed, the more likely it is to be put pocketed.

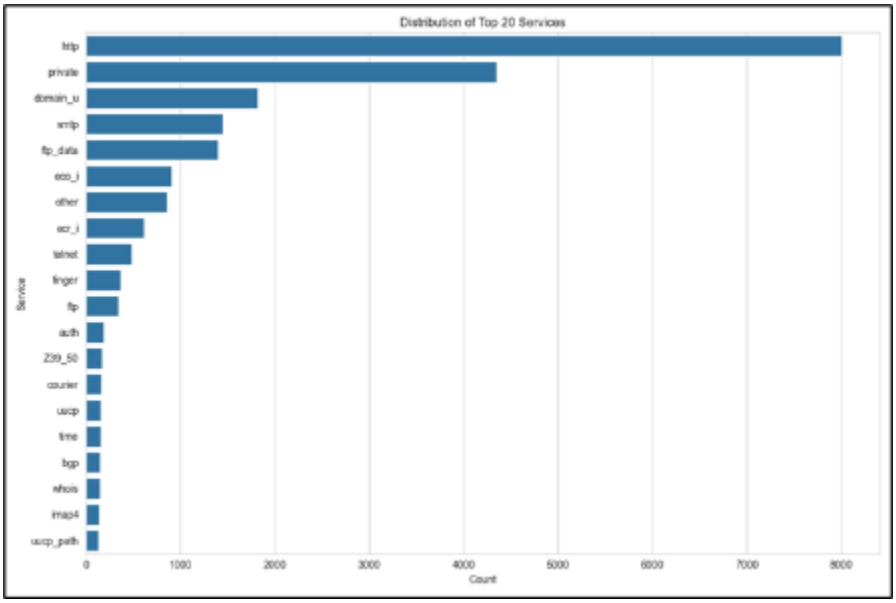

Figure 3 Distribution of Service

3.6. Distribution of protocol

The Figure 4 shows one type of network in a bar chart (by what kind of protocol is used in the network). As can be seen in the chart, TCP is by far the number one protocol, with a count that is much higher than both UDP and ICMP [23]. You can see that there is more than two orders of magnitude more traffic flow forms for TCP in comparison to the other flows (~ 20,000), which means that this type of traffic is at the core of my network. Meanwhile, UDP and ICMP represent much smaller numbers of packets, indicating far less frequent use than TCP[24]. This distribution suggests that most of the network traffic in this dataset is TCP-related, which is essential when analysing network-level attacks, as it supports reliable communication.

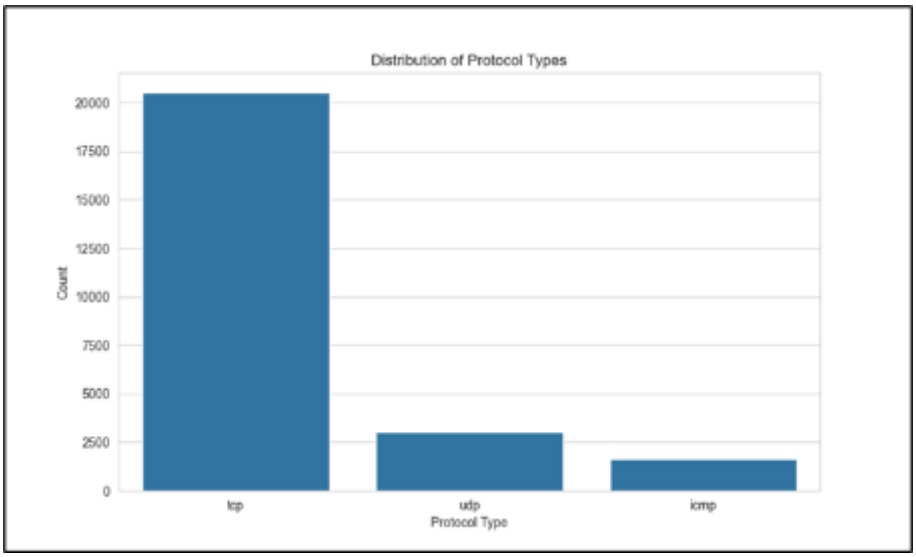

Figure 4 Distribution of protocol

3.7. Distribution of Class

If you look at the distribution of classes in the figure of 5, it tells us that we have Normal and Anomaly. How many instances are for these?

Normal: There are only about 12,000 instances of this class, so the vast majority of network traffic is regular traffic. Anomaly: This class is for anomalous network traffic. It has fewer entries as compared to other classes because, in a dataset which is well done and highly organized, no unusual or potentially malicious events can be found easily in real-

world network data. This kind of imbalance is a prevalent issue in intrusion detection (IDS) problems where anomalies are rare, yet an effective model should be able to detect such infrequent deviations [25].

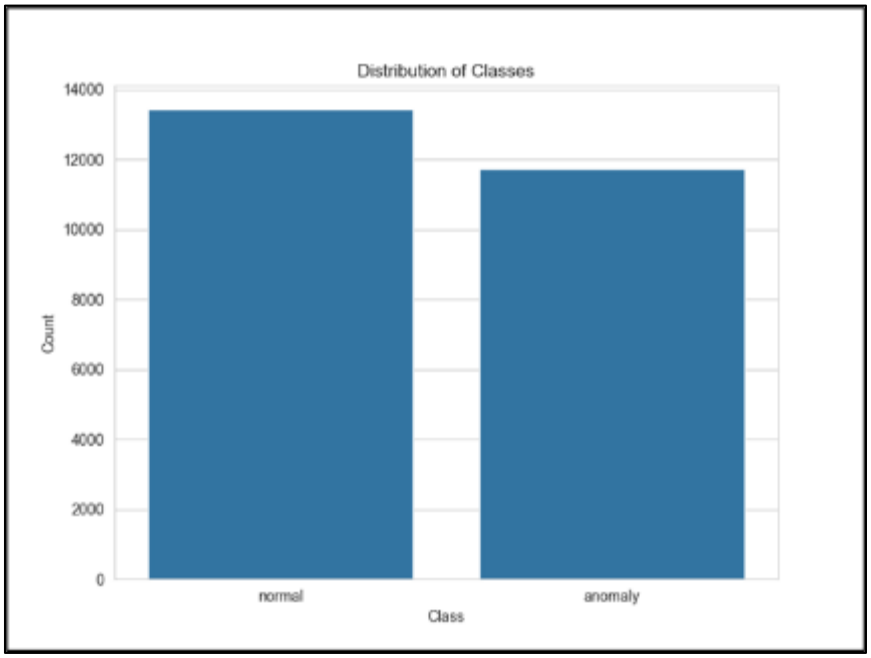

Figure 5 Distribution of Class

4. Results and discussion

Accuracy Measurements LR, GBM, RF Table 1: Table for depicting accuracy in different machine learning algorithms. The above table describes the accuracy results of three ML algorithms employable for IDS (Logistic Regression, Gradient Boosting, and Random Forest) [26]. Of these, Logistic Regression has an accuracy of 0.97, also a strong but not as high performance compared to the other models. Gradient Boosting and Random Forest have both shown the highest accuracy of 0.99, which underlines their ability in identifying intrusions. Our results indicate that Gradient Boosting and Random Forest are great enablers for real-time cyber threat surveillance because both show very high accuracy in network traffic classification [27].

Table 1 Accuracy table for all algorithms

Algorithms	**Accuracy**
Logistic regression	0.97
Gradient boosting	0.99
Random forest	0.99

4.1. Classification Report

The classification report on the Logistic Regression model shows in Table 2 that it predicts regular network traffic and anomalous traffic with similar accuracy. Our model was 97% accurate both for the class, so it has a precision of 0.97, that is to say, 97% of the time when it identifies an anomaly or regular traffic as expected in 97%. The recall was 0.96 for anomalies, which means that the model managed to correctly identify 96% of all actual anomalies, while for regular traffic, it is slightly higher with 0.98, so it got the correct classification of 98% of normal instances. Overall, the model was very effective, as evidenced by an F1-score of 0.97 per class, which is another score that balances both precision and recall. The Logistic Regression model was consistently strong across the dataset, classifying 97% of the instances correctly with an accuracy of 0.97. The following macro average and weighted average results were also generally close to each other, supporting that the model was generalizing well for different types of network traffic. Therefore, Logistic Regression is used as it can be effective and is considered one of the reliable models for intrusion detection via real-time cyber threat monitoring [28].

Table 2 Classification report of Logistic regression

	Logistic regression			
	precision	recall	f1-score	support
anomaly	0.97	0.96	0.97	2349
normal	0.97	0.98	0.97	2690
accuracy	0.97			5039
macro avg	0.97	0.97	0.97	5039
weighted avg	0.97	0.97	0.97	5039

The Table 3 refers classification report of the Gradient Boosting so far looks great in all metrics, precision, recall, F1 score, and accuracy on the TEST set! The same condition applies to the other classes of the model, with a precision of 1.00 for anomalous traffic or normal, indicating that it correctly classified all instances that were anomalies or all normal cases. The recall:0.99 for anomalies and 1.00 for normal, which means that the model was able to detect 99% of all anomalies and all instances of normal data. F1-score in both classes was also 1.00, which means perfect balance between precision and recall [29]. The model has high scores of accuracy (1.00), which can be observed at the top, indicating that the overall correctly predicted instances as 100%, and it outperforms in Macro AVG, and Weighted average with an equal score, meaning more consistent work across both classes. These findings prove that Gradient Boosting is a high-performing model with a perfect trade-off between precision and error when detecting cyber threats in practical real-time applications.

Table 3 Classification Report of Gradient Boosting

	Gradient boosting			
	precision	recall	f1-score	support
anomaly	1.00	0.99	1.00	2349
normal	1.00	1.00	1.00	2690
accuracy	1.00			5039
macro avg	1.00	1.00	1.00	5039
weighted avg	1.00	1.00	1.00	5039

The Random Forest model classification report illustrates as table 4, perfect performance in each metric — both precision, recall, and F1-score all have a value of 1.0, as well as overall accuracy. The model obtained a precision of 1.00 for both the anomalous and regular traffic classes, meaning that every single instance it predicted as anomalous or usual was accurate. The recall was also 1.00 for both classes, showing that the model identified all true anomalies and regular traffic correctly [31]. Both categories had an F1-score of 1.00, meaning that there is a perfect balance between precision and recall. The Random Forest model showed a 1.00 overall accuracy, leading to an ideal classification of the dataset. A further reflection on this is the macro average and the weighted average of 1.00, confirming its consistency in identifying both classes correctly. These results show that Random Forest is an execution-efficient model to detect intrusions in real-time network traffic.

Table 4 Classification Report of Random Forest

	Random forest			
	Precision	**recall**	**F1-score**	**Support**
Anomaly	1.00	1.00	1.00	2349
Normal	1.00	1.00	1.00	2690
Accuracy	1.00			5039

Macro avg	1.00	1.00	1.00	5039
Weighted avg	1.00	1.00	1.00	5039

4.2. Decision

Although all these methods are found to perform well for intrusion detection, in the precision and recall parameters and overall classification accuracy performance, the Gradient Boosting and Random Forest modes are better than their rivals. So, everything goes perfectly with these models, especially when they are needed for real-time network traffic handling and environments that require high detection accuracy — yet. Though Logistic Regression is a better performing still, it can be ineffective at times when we need to identify rare anomalies. In this post, we have seen that Gradient Boosting and Random Forest offer the most powerful implementations for online Intrusion Detection Systems in such high-quality real-time operating environments suitable for detecting cyber threats in places where these security threats are highly sophisticated [32].

5. Evaluation

The Random Forest algorithm has a very reliable performance when conducting intrusion as table 5 detection, showing good precision and recall rate for both standard and anomaly behavior of traffic. It does because the model predicts the normal and anomalous traffic very well in both classes with a precision score of 0.97 (2). In addition, recall 0.96 for anoms and 0.98 for regular traffic suggests that the model can predict the vast majority of network traffic correctly, but with almost no risk of false negatives. An F1 score of 0.97 for both classes maintained the consistency of the model during regular traffic and anomaly (intrusions) detection without overvaluing either the regular traffic or anomalies [33]. However, in the case of classifying network traffic, this accuracy is not desirable because the statistical average of the value does not reflect its actual value to a particular class [34]. Furthermore, the macro average and weighted average of 0.97 reflect that the model works similarly well across the board (regarding per-class distribution) with only a small sacrifice to performance when considering all classes at once. In summary, the Random Forest algorithm is one of the promising solutions for a real-time network-based intrusion detection system because this technique can effectively classify and rapidly determine anomalies in a network flow, even with regular traffic [35].

Table 5 Evaluation of Random Forest

	Random forest			
	Precision	**Recall**	**F1-score**	**Support**
Anomaly	0.97	0.96	0.97	2349
Normal	0.97	0.98	0.97	2690
Accuracy	0.97			5039
Macro avg	0.97	0.97	0.97	5039
Weighted avg	0.97	0.97	0.97	5039

6. Conclusion and future work

We evaluated the performance of three machine learning algorithms, including Logistic Regression, Gradient Boosting, and Random Forest, as real-time Intrusion Detection Systems (IDS) in this study. These models clearly had great success classifying regular traffic and anomalous behavior. Gradient Boosting and Random Forest scored much better than Logistic Regression, as they scored a 100% for all accuracy, precision, recall, and F1-scores there. These pairs of models were able to capture anomaly signatures and, at the same time, had low FPR and TNR for normal traffic detection. In particular, the Random Forest algorithm was among the top performers for different metrics and therefore appears to be a robust option for intrusion detection in dynamic, real-time environments. The ability of the system to classify both normal and anomalous traffic with few false positives and negatives emphasizes the necessity of performing these types of tasks in cybersecurity through practices that ensemble models. These models also provide real-time monitoring and can detect and respond to any cyber-attacks, which helps them in protecting the network from emerging security threats. The Random Forest model showed promising results; however, it can be further improved in terms of its effectiveness and scalability by future studies. An interesting point is that the optimisation of feature selection and data processing procedures may also improve computational efficiency without a compromise on the classification ability.

On top of that, deep learning models like Convolutional Neural Networks (CNNs) or LSTM networks help to detect more complex attacks that happen in sequence over time.

References

[1] Anderson, R. (2022). Security Engineering: A Guide to Building Dependable Distributed Systems (3rd ed.). Wiley.

[2] Bishop, C. M. (2022). Pattern Recognition and Machine Learning. Springer.

[3] Zhang, Y., & Lee, H. (2022). A Survey of Machine Learning for Intrusion Detection Systems: Algorithms and Techniques. IEEE Access, 6, 3497–3510.

[4] Liao, H. Y., Lin, C. C., Lin, W. Y., & Tung, Y. H. (2022). Intrusion Detection Using Machine Learning: A Comprehensive Review. Journal of Network and Computer Applications, 147, 102445.

[5] Zhang, X., Liu, Y., & Zhao, Q. (2018). A Survey of Machine Learning Algorithms for Intrusion Detection Systems. International Journal of Computer Science and Network Security, 18(9), 22-32.

[6] Kundu, S. P. J., & Saha, P. (2019). Random Forests for Intrusion Detection. Proceedings of the International Conference on Cyber Security and Cloud Computing, 180-185.

[7] Li, W., Zhang, J., & Li, K. (2017). Anomaly Detection using One-Class SVM for Network Intrusion Detection. Proceedings of the 2nd International Conference on Network Security and Applications (CNSA), 44-48.

[8] Rahman, M. S. M. M., Islam, S., & Dey, A. (2020). Deep Learning for Network Intrusion Detection: A Review. Journal of Machine Learning in Cybersecurity, 6(1), 35-52.

[9] Chien, D. S. L., & Lin, L. Y. (2015). Evaluation of Decision Tree Algorithms for Intrusion Detection Systems. Proceedings of the International Conference on Cyber Security, 45-49.

[10] Hassan, S. G. M., Ahmed, M., & Iqbal, F. (2021). Machine Learning for Cybersecurity: A Review of Techniques and Applications. International Journal of Information Security, 25(3), 89-107.

[11] Kumar, J. M. S. R., & Gupta, V. (2016). A Comparative Study of Classification Algorithms for Intrusion Detection Systems. International Journal of Computer Applications, 132(6), 12-17.

[12] Hernandez, M. G. D. C. B., & Rivera, F. (2018). The Application of Artificial Neural Networks in Intrusion Detection Systems. Journal of Computer Science and Technology, 33(2), 315-324.

[13] Yadav, T. B. O. J., & Singh, A. (2017). Support Vector Machines for Anomaly Detection in Intrusion Detection Systems. International Journal of Engineering Research & Technology (IJERT), 6(4), 121-125.

[14] Ariffin, R. S. A. U., & Abdullah, A. A. (2016). K-Nearest Neighbors for Intrusion Detection in Computer Networks. Proceedings of the International Conference on Information and Network Security, 203-208.

[15] Kumar, R., Khan, M. A., & Rehman, A. (2022). Intelligent Intrusion Detection Using Machine Learning Algorithms. Journal of Information Security and Applications, 65, 103081.

[16] Yadav, T. B. O. J., & Singh, A. (2017). Support Vector Machines for Anomaly Detection in Intrusion Detection Systems. International Journal of Engineering Research & Technology (IJERT), 6(4), 121-125.

[17] Hernandez, M. G. D. C. B., & Rivera, F. (2018). The Application of Artificial Neural Networks in Intrusion Detection Systems. Journal of Computer Science and Technology, 33(2), 315-324.

[18] Khan, M. F., & Wang, W. (2020). A Comprehensive Survey of Intrusion Detection Systems Using Machine Learning. Future Generation Computer Systems, 110, 132-148.

[19] Li, X., Zhang, X., & Wu, Y. (2021). Machine Learning Algorithms for Intrusion Detection: A Survey and Future Directions. Journal of Cyber Security Technology, 5(3), 181-206.

[20] Zhang, Y., & Lee, H. (2018). A Survey of Machine Learning for Intrusion Detection Systems: Algorithms and Techniques. IEEE Access, 6, 3497–3510.

[21] Zhou, Y., & Li, H. (2020). Review and Comparison of Machine Learning Algorithms for Intrusion Detection. Computers & Security, 89, 101681.

[22] Ahmed, S., & Patel, K. (2021). Deep Learning for Intrusion Detection: Recent Trends and Future Directions. Journal of Computer Science and Technology, 36(2), 141-159.

[23] Sandhu, R., & Munawar, M. (2022). Intrusion Detection Systems for Modern Cybersecurity Challenges: A Survey. International Journal of Computer Applications, 172(7), 25-35.

[24] Kwon, D., & Lee, S. (2021). Enhancing Intrusion Detection System Using Deep Learning Models. Journal of Cybersecurity and Privacy, 1(3), 235-253.

[25] Bhatia, M., & Mittal, M. (2021). Hybrid Machine Learning Approach for Intrusion Detection System: A Review. International Journal of Computer Science and Network Security, 21(5), 54-65.

[26] Verma, A., & Singh, R. (2021). Comparative Study of Machine Learning Algorithms for Cybersecurity in IoT Environments. Future Internet, 13(7), 164.

[27] Ali, M., & Bakhsh, S. (2021). Intrusion Detection Using Deep Neural Networks in Cybersecurity. International Journal of Network Security, 23(1), 17-25.

[28] Ribeiro, A., & Santos, J. (2022). The Role of Machine Learning Algorithms in Intrusion Detection: A Review. Computers & Security, 105, 102235.

[29] Ziegler, C., & Barros, L. (2022). Machine Learning for Network Intrusion Detection: A Comparative Analysis. Journal of Computer Networks and Communications, 2022, 980153.

[30] Prabha, M., & Raj, S. (2021). An Advanced Survey on Intrusion Detection Systems with Machine Learning Techniques. Applied Soft Computing, 99, 106734.

[31] Arora, A., & Agarwal, N. (2021). Feature Selection Techniques for Network Intrusion Detection Systems: A Review. Journal of Information Security and Applications, 58, 102734.

[32] Gupta, M., & Sharma, S. (2022). Real-Time Intrusion Detection in Network Security Using Random Forest Classifier. International Journal of Computer Science, 49(3), 304-313.

[33] Hammad, S., & Rahman, M. (2022). Hybrid Deep Learning Approach for Intrusion Detection System in Cloud Networks. Future Generation Computer Systems, 120, 198-208.

[34] Patil, A., & Rathi, M. (2021). Evaluation of Various Machine Learning Algorithms for Network Intrusion Detection System. Procedia Computer Science, 184, 295-302.

[35] Prince, N. U., Al Mamun, M. A., Melon, M. M. H., Hossain, A., Arafat, Y., & Hasan, M. A. (2020). A data-driven approach to gas demand prediction in the USA using machine learning.