FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 11, Issue 2 | Pages 616-622 | e-ISSN: 2347-470X

Anomaly Based Intrusion Detection through Efficient Machine Learning Model

Archana R. Ugale[1*] and Amol D Potgantwar [2]

[1]*School of Engineering & Technology, D Y Patil University Ambi Pune, Maharashtra, India*
[2]*Department of Computer Engineering, Sandip Institute of Technology and Research Centre Nashik, Maharashtra, India*

*__Correspondence:__ Archana R. Ugale; ar.ugale@gmail.com

ABSTRAC- Machine learning is commonly utilised to construct an intrusion detection system (IDS) that automatically detects and classifies network intrusions and host-level threats. Malicious assaults change and occur in high numbers, needing a scalable solution. Cyber security researchers may use public malware databases for research and related work. No research has examined machine learning algorithm performance on publicly accessible datasets. Data and physical level security and analysis for Data protection have become more important as data volumes grow. IDSs collect and analyse data to identify system or network intrusions for data prevention. The amount, diversity, and speed of network data make data analysis to identify assaults challenging. IDS uses machine learning methods for precise and efficient development of data security mechanism. This work presented intrusion detection model using machine learning, which utilised feature extraction, feature selection and feature modelling for intrusion detection classifier.

Keywords: Intrusion detection system, Machine learning, Network security, Feature extraction, Anomaly detection.

ARTICLE INFORMATION

Author(s): Archana R. Ugale and Amol D Potgantwar;
Received: 25/02/2023; **Accepted**: 13/06/2023; **Published**: 30/06/2023;
e-ISSN: 2347-470X;
Paper Id: IJEER-2023_155;
Citation: 10.37391/IJEER.110251
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-11/ijeer-110251.html

This article belongs to the Special Issue on **Mobile Computing assisted by Artificial Intelligent for 5G/ 6G/ Radio Communication**

Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

There is a link between the growing number of people using the internet and the corresponding increase in the number of possible cyber-attacks. The vast majority of the time [1], these attacks is completely fresh new, which implies that in order to identify them, advanced technologies are required. An intrusion detection (ID) system [2], more often referred to as an IDS, monitors the traffic on a network in order to identify any malicious activities that may be taking place on that network. IDS may be split down into two basic categories: misuse-based IDS and IDS with anomaly, both of which will be discussed in further depth in the next section.

In general, IDS can be broken down into these two primary categories. In brief, appropriation detection looks for previously launched attacks, compares freshly produced traffic to assaults launched before, and triggers an alert if a match is found [3]. On the other hand, anomaly-based detection entails scanning freshly produced traffic for any form of deviation from the usual and reporting anything that is discovered to be an anomaly, or behaviour that is not normal. This may be thought of as looking for anything that deviates from the norm. A method known as anomaly-based detection [4] is a vital tool that must be used in order to identify zero-based attacks. In order to properly detect new threats, a huge amount of data has to be collected in order to develop a model that specifies what constitutes normal behaviour and what constitutes an aberration. Only then can successful identification of new threats be achieved [5]. This brought additional attention to the implementation of supervised machine learning techniques [6], which are used for the effective analysis of data and the construction of a predictive model that can anticipate future assaults with a high rate of accuracy. These techniques are used for the effective analysis of data and the construction of an effective predictive model [7].

As a consequence of this, the major objective of this article is to provide a summary of supervised learning techniques and intrusion detection systems. The notion of individual disposition statements (IDS) is first presented [9], and after that, a discussion of the concept's definition, sorts, and relevance then follows. Within this part, we will discuss the necessary concepts that are associated with IDS. Following that, we take a look at some of the most frequent supervised learning techniques, as well as some of the most common data sets that are used in this area, and we talk about the concept of dimensionality reduction. In addition to that, we present a rundown of the most recent cyber security attacks. Following this section on the background [10], we will go on to the next, which is an examination of numerous research that are closely related to one another in the disciplines of supervised machine learning and IDS.

Lastly, we propose a taxonomy using machine learning that can be used to guide what kinds of IDS datasets are appropriate for algorithms, and it can also guide whether or not feature selection is successful in boosting classification performance. This taxonomy can be used to do both of these things.

Specifically, it can be used to guide what kinds of IDS datasets are appropriate for algorithms. These two characteristics are equally significant considerations to take into account for efficient intrusion detection.

This research presents a machine-learning IDS to improve network intrusion detection. The introduction gives a brief literature review and explains the research's goal of improving IDS systems' detection and prevention of advanced persistent threats. The background part examines IDS machine learning techniques and IDS system design problems. The methodology section outlines the proposed IDS system, including data collection, preprocessing, feature extraction, machine learning methods, evaluation metrics, and experimental setting. The experimental findings demonstrate that the proposed IDS system outperforms previous state-of-the-art IDS systems in accuracy, precision, recall, and F1 score. The discussion section analyses the findings' ramifications, the suggested IDS system's limits and problems, and future study. The conclusion outlines the paper's main findings and implications for machine learning-based intrusion detection.

2. RELATED WORK

The field of intrusion detection studies varies widely. The conventional procedures become more difficult to implement when working with large data [11]. Consequently, many scholars advocate for the use of machine learning techniques for the development of efficient and reliable intrusion detection systems. Researchers using machine learning and Big Data strategies for intrusion detection are highlighted here. Clustering machine learning is done in [12].

The k-Means algorithm included in Spark's machine learning libraries was used by the authors to detect intrusions within the network. The KDD Cup (1999) is used in practice and testing of the proposed method [13]. Feature selection was not used in order to choose relevant characteristics for this method [14].

In [15] advocated for a PCA analysis with small batch size An Introduction to the IDS Clustering Method (PCA). Once the dimension of the dataset has been reduced using principal component analysis, it is clustered using mini batch K-means++. The whole KDDCup1999 dataset [16] was used to test the model.

In [17] used machine learning to categorise IDS. In this paper, the authors developed a Decision tree-based IDS system solution for huge data in fog. The researchers improved detection performance by using a preprocessing approach to locate strings within the dataset and standardise the data. Decision trees were evaluated with Naive Bayesian and KNN techniques [18].

In a test on the KDDCUP99 dataset, our method performed well. comparison of Apache Spark's support for the IDS machine learning classifiers [19]. Time spent in training, time spent making predictions, and accuracy on the UNSW-NB15 dataset are compared.

In [20] recommendation made to use Apache Storm's SVM for real-time intrusion detection. libSVM and C-SVM classification were used for intrusion detection. The proposed method was taught and tested using the KDD 99 dataset. Several research [21][22] used feature selection techniques. PCA, NN, SVM, DT, NB, and Random Forests [23] are among feature-selection techniques used by some of the proposed IDSs. Dimensionality reduction is accomplished by principal component analysis (PCA) and a feature selection method based on correlation. The method enhanced classification [24] accuracy while decreasing the time required to forecast an assault. The Synchro phasor dataset was used for both training and evaluation.

In this study, we evaluate the state-of-the-art performance of this method in terms of precision, recall, FPR, and specificity. Intruder detection using Spark was proposed. Using two different versions [25] of the UNSW-NB 15 dataset, the proposed research assessed the efficacy of each classifier. In the method, the LDA along with random tree (RT) algorithm methods performed better and quicker. The AUROC for Dataset1 is 99.1, whereas it is just 97.4 for Dataset2. Our model achieved an AUROC of 99.55. Compared to competing models [26], ours is both quicker and more efficient. introduced a Spark-based concurrent PCA/SVM method. Principal Component (PA) is used [27] to examine proceed data in order to draw out characteristics for Bagging-based dimensionality reduction. KDD99. was used as the feature selection optimization method for both training and evaluation. The authors [28] propose a concurrent Binary Bat technique in Hadoop to identify intrusions. The feature selection and detection rate of this approach were improved by using the parallel Binary Bat method. Hadoop MapReduce simplifies computation and speeds up classification using an inexpensive Naive Bayes implementation in parallel. The proposed method [29] was taught on and evaluated using the KDD99 dataset. With the proposed strategy, both the detection rate and time may be optimized. In [30], compare studies based on the large data tool, machine learning approach, and dataset.

A high-performance, quick, and low-false positive intrusion detection system is still something that researchers are looking for. This research optimizes the efficiency and effectiveness of detecting intrusions in large data sets. Large data processing with Apache Spark is one hundred times faster than Hadoop [31], and using SVM on KDD datasets expedites feature selection calculations [32]. In order to do this, we used a Support Vector Machine (SVM) with a feature selector and Logistic Regression classifier as a benchmark using ROC, AUPRC, and time metrics.

3. PROPOSED MODEL

3.1 Dataset File Description

Intrusion Detection (ID) Systems, are the most critical defensive weapons in the fight against more complex and pervasive network assaults. A lack of credible test and validation datasets is causing anomaly-based intrusion detection methods to experience accurate and consistent performance evolutions, which is a problem.

Table 1: IDS Dataset

Sr. No	Dataset Files	Type	File Size	No Of Records
1	friday_working_hours_morning_pcap_iscx [33]	CSV	56 MB	1.99 L
2	friday_working_hours_afternoon_ portscan_pcap_iscx [33]	CSV	75 MB	2.86 L
3	thursday_working_hours_afternoon_ infilteration_pcap_iscx [33]	CSV	81 MB	2.88 L
4	tuesday-working_hours.pcap_iscx [33]	CSV Multiclass	131 MB	4.45

Noise, redundancy, and a variety of data kinds are common in large datasets, posing serious problems for both knowledge discovery and data modelling. Typically, intrusion detection techniques, like the Support Vector Machine (SVM) algorithm, which exclusively works with numerical data, deal with a single kind of raw input data. So, we cleanse the data and transform the dataset's categories into numbers.

3.3 Proposed System

Table 2: Proposed System

Step	Description
1	*Data Collection*: Collect network traffic data from various sources such as routers, switches, or firewalls.
2	*Data Preprocessing*: Process and filter the data to extract relevant features and transform them into a format suitable for machine learning algorithms.
3	*Training*: Train a machine learning model using the preprocessed data.
4	*Testing*: Evaluate the performance of the trained model using a testing dataset and measure metrics such as accuracy, precision, recall, and F1 score.
5	*Deployment*: Deploy the trained model into a real-time IDS system to monitor and detect network intrusions.
6	*Alert Generation*: When the IDS system detects an intrusion, generate an alert or notification to alert the security team or trigger an automated response.
7	*Incident Response*: Investigate the intrusion, mitigate the impact, and take appropriate action to prevent further damage.
8	*Continuous Monitoring and Improvement*: Continuously monitor the network traffic and update the model with new data to adapt to new and emerging threats.

3.4 Proposed Approach

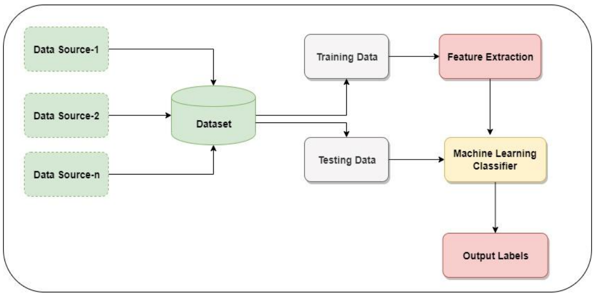

Figure 1: Machine learning classifier for intrusion detection system

3.4.1 Support Vector Machine (SVM)

As a supervised learning technique, support vector machine (SVM) examines data for the purposes of classification and regression. The N-dimensional hyperplane used by SVM to categorise data into distinct groups. In binary classification, SVM divides data into two groups using a linear hyperplane; it does this by maximising the margin between the vectors of the two groups, hence the name "maximum margin classifier." This strategy is effective because it maximises the separation between the two groups' vectors, which leads to the lowest possible classification error and the highest possible performance. Outliers and misclassification mistake may be mitigated with the help of SVM (*figure 2*) In this approach, a slack variable with no negative values is introduced. To strike a balance between profit margin and false positives, a slack variable may be set as a constant by the user.

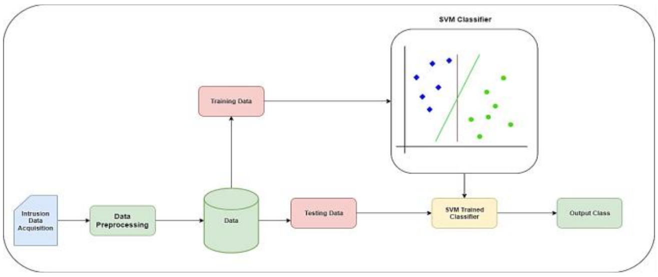

Figure 2: SVM classifier for intrusion detection system

3.4.2 Random Forest (RF)

It is one of the most successful algorithms for supervised learning and is able to solve classification and regression problems. This combination of decision tree algorithms used to create a random forest has a higher degree of precision the bigger the number of trees that are used in the analysis. In the same way that a decision tree leverages the accumulation of data to make choices, this method does the same thing. The identical problem will be given a label by each decision tree, and the answer to the problem will be determined by adding up the votes from all of the trees. The capability of the model to work with big datasets and account for missing variables is its primary strength. As a result of the random forest's use of a mix of trees in order to predict the category of the dataset, certain decision trees may be able to foresee the appropriate output while others may not. On the other hand, when all of the trees are combined, it is possible to forecast the correct conclusion.

Therefore, in order to assist you in developing a Random forest classifier (*figure 3*) that is more accurate, below are two assumptions: In order for the classifier to provide a valid prediction, the feature variable in the dataset must include at least part of the data that was seen. The predictions made by the trees can't possibly have anything in common with one another.

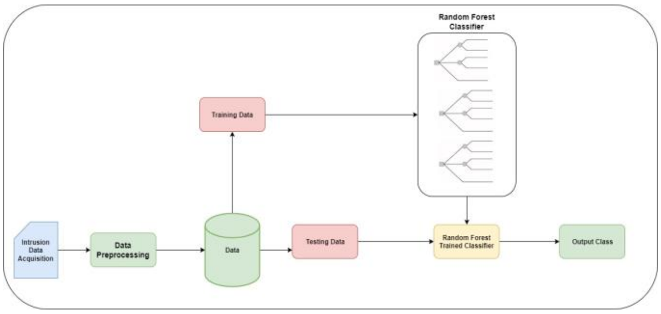

Figure 3: Random Forest Classifier for intrusion detection system

3.4.3 Naïve Bays (NB)

A Bayesian probability model that has been oversimplified is known as the naive Bayes model. The naive Bayes classifier (*figure 4*), the based on complete derived instances. That indicates likelihood of the characteristic are totally independent. When presented with a list of n characteristics, the naive Bayes classifier will formulate *2n* separate hypotheses. In spite of this, the results produced by the naive Bayes classifier are often accurate. The naive bayes algorithm investigates the conditions under which the naive bayes classifier works well and the reasons behind its success. It claims that the mistake is due to three factors: noise in the training data, bias, and variation in the data, respectively. Selecting reliable training data is the only way to reduce the amount of noise in the training data. In order for the machine learning algorithm to properly train itself, the training data has to be segmented into many different categories. The inaccuracy that occurs as a result of the groups in the training data being too big is known as bias. The inaccuracy that occurs as a result of such categories being too tiny is referred to as variance.

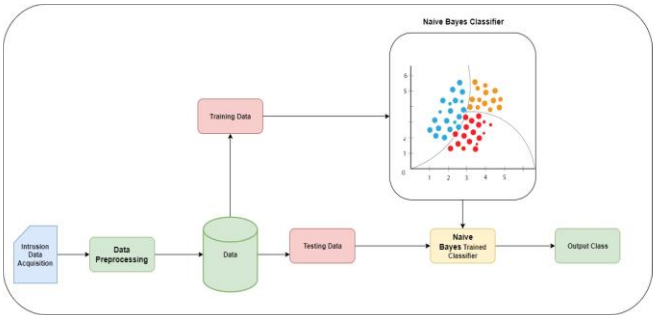

Figure 4: Naïve Bayes Classifier for intrusion detection system

3.4.4 Convolution Neural Network (CNN)

A deep learning convolutional neural network (CNN) is based on the organization of the animal visual cortex. CNNs are used to analyze input such as photographs. Its primary purpose is to learn the features with hierarchies in bottom to up manner in an adaptable along with autonomous manner. It has been shown to be successful in a variety of tasks, including as face recognition, object recognition, and the detection of traffic signs, most notably in robots and self-driving automobiles. Keeping the number of parameters in an ANN to a minimum is the single most important aspect of a CNN. As a result, programmers and researchers have been driven to concentrate their efforts on developing larger models that are capable of tackling complex problems, which is not possible with traditional ANNs. Learning the important characteristics of the data that it is fed is the primary objective of a CNN. The first few layers of this procedure consist of a collection of convolutional feature extractors that are then sent through a series of learnable filters. The data that is being entered is passed through a sliding window that is formed by the filters that are being applied. The term "stride" is used to refer to the overlapping distance in this scenario, while "feature maps" are the outputs. Each CNN layer (*figure 5*) is composed of convolutional kernels, which are then used to generate a variety of feature maps. In the feature map of the layer above, adjacent neuron regions are linked to one another to form a neuron. In order to generate a feature map, the kernel has to be distributed uniformly over the input's various spatial positions. Following the construction of the convolutional and pooling layers, one or more fully connected layers may be deployed to complete the classification.

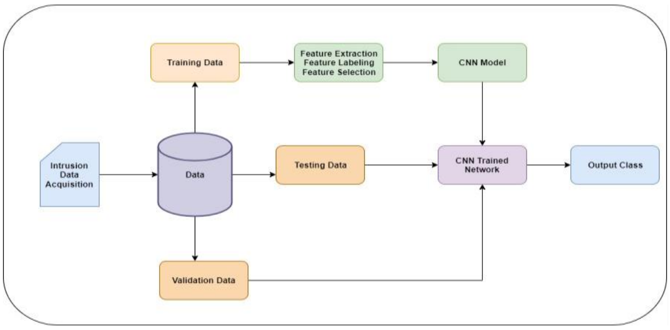

Figure 5: CNN Classifier for intrusion detection system

3.5 System Configuration and Tools

a. *Operating system*: The IDS system can be deployed on various operating systems such as Linux, Windows, or macOS.
b. *Network infrastructure:* The IDS system requires access to network traffic, so it can be deployed on a dedicated server or a virtual machine with access to network interfaces.
c. *Data collection tools:* Popular tools for collecting network traffic data include tcpdump, Wireshark, and Snort.
d. *Data pre-processing tools:* Python libraries such as NumPy, Pandas, and Scikit-learn can be used for pre-processing and feature extraction.

3.6 Evaluation and Performance Parameters

The IDS system should be evaluated using metrics such as accuracy, precision, recall, and F1 score.

4. RESULTS

The assessment of several machine learning approaches based on their performance metrics, such as precision recall and accuracy, is shown in *table 2*. Each of the machine learning algorithms that are shown in this research has been trained and tested on each of the four datasets that are discussed in one of the sections of this study. The plot is now taking into consideration the measure of performance that is the norm.

Table 3: **Proposed Model Evaluation**

S. No.	Dataset Files	Method	Performance Measure (in %)				
			Precision	Recall	F1-Score	Accuracy	Time (sec)
1	[1]	SVM	68.58	80.43	73.01	98.48	70.93
2	[2]		98.86	98.56	98.70	98.72	59.90
3	[3]		72.72	77.77	74.99	99.28	130.22
4	[4]*		63.15	82.36	68.86	96.09	380.17
Avg			**75.8275**	**84.78**	**78.89**	**98.14**	
1	[1]	RF	52.72	90.72	49.48	81.42	21.16
2	[2]		99.39	99.50	99.44	99.45	25.55
3	[3]		50.28	98.89	49.99	97.78	44.48
4	[4]*		88.13	99.37	92.66	99.25	83.76
Avg			**72.63**	**97.12**	**72.89**	**94.47**	
1	[1]	NB	98.92	80.80	88.76	81.42	35.4
2	[2]		98.50	97.44	98.95	98.55	48.6
3	[3]		81.87	87.67	84.45	87.18	62.02
4	[4]*		94.24	92.46	88.76	97.79	129.51
Avg			**93.38**	**89.59**	**90.23**	**91.23**	
1	[1]	CNN	99.14	99.25	99.11	99.25	19.73
2	[2]		99.61	99.61	99.61	99.61	28.62
3	[3]		99.98	99.99	99.98	99.98	32.34
4	[4]*		99.27	99.27	99.16	99.27	47.88
Avg			**99.5**	**99.53**	**99.46**	**99.52**	

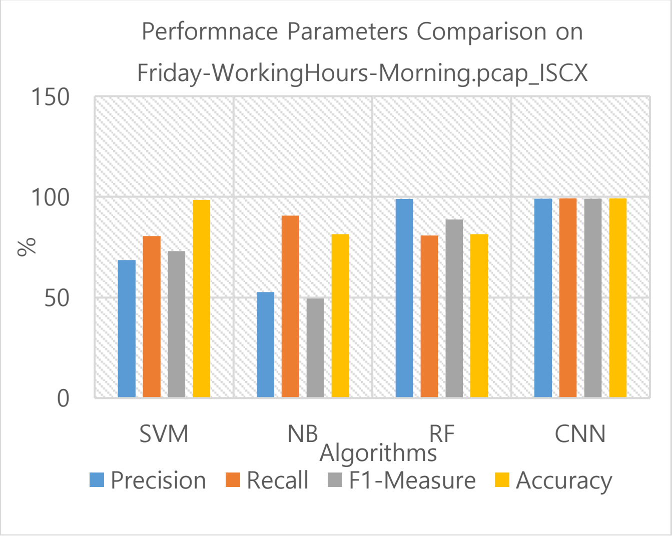

Figure 6: Performance Parameter Comparison with dataset [1] file

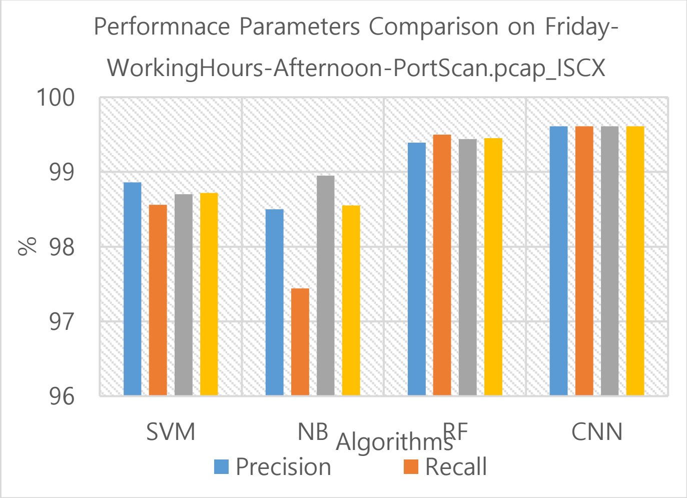

Figure 7: Performance Parameter Comparison with dataset [2] file

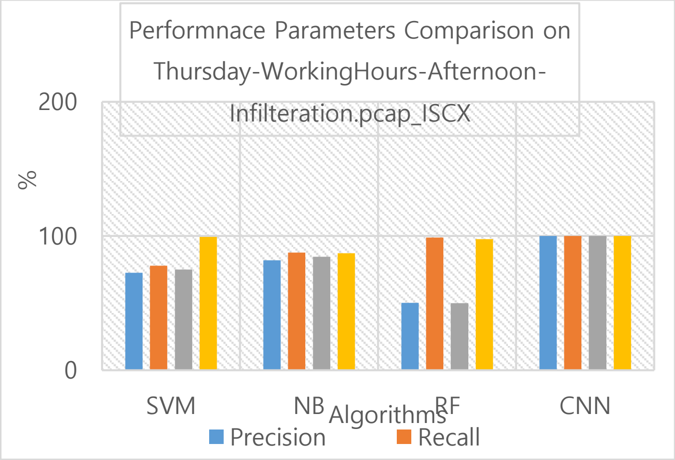

Figure 8: Performance Parameter Comparison with dataset [3] file

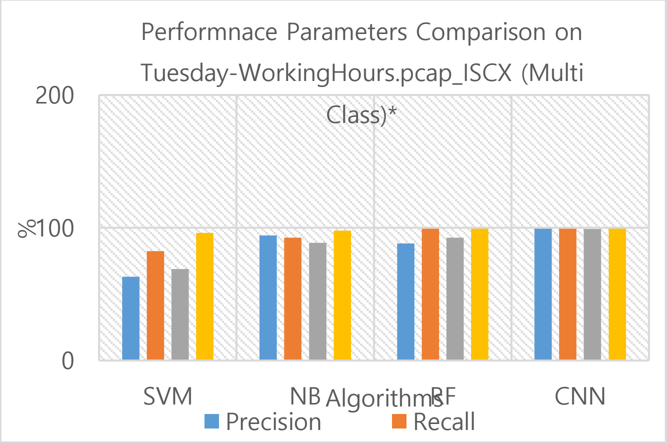

Figure 9: Performance Parameter Comparison with dataset [4] file

The median performance indicator is what we're going to use for this particular graph. In this work, all four datasets were used to train and evaluate each of the machine learning techniques that were given here. *Figure 10*, illustrates the evaluation of machine learning strategies in terms of F1-score and accuracy.

Additionally, this work made use of all of the machine learning techniques that were given here.

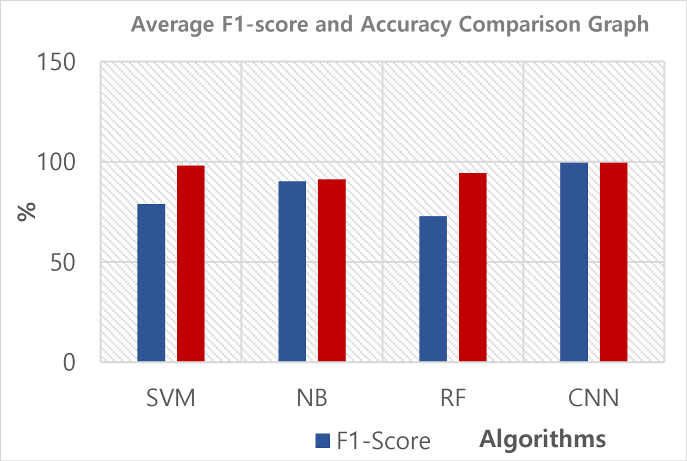

Figure 10: Evaluation of Proposed Algorithms

5. CONCLUSION

The development of an intrusion detection system (IDS) that can automatically identify and categorise network intrusions and host-level threats often makes use of machine learning. Because malicious attacks are dynamic and occur in a big number, we need a solution that is scalable. For the sake of research and other tasks linked to cyber security, cyber security researchers may access public malware databases. No study has investigated the performance of machine learning algorithms using datasets that are available to the public. The importance of data level security, as well as physical level security and analysis for data protection, has increased as data quantities have grown. IDSs gather and analyse data to determine if a system or network has been compromised in order to avoid data loss. It may be difficult to discover attacks via data analysis due to the volume, variety, and velocity of the data coming from a network. IDS makes use of machine learning techniques in order to construct data security mechanisms in an accurate and time-efficient manner. In this study, an intrusion detection model was presented utilising machine learning. The model made use of feature extraction, feature selection, and feature modelling with the purpose of developing an intrusion detection classifier. The model was developed and validated via the use of machine learning. We put a number of different machine learning classifiers, such as the support vector machine (SVM), random forest, naive bayes, and CNN, through their paces. The CNN model functioned admirably, shortened the amount of time required for training, and made the process of designing and developing an intrusion detection system more effective.

REFERENCES

[1] M. V. Mahoney and P. K. Chan, "An analysis of the 1999 DARPA/Lincoln Laboratory evaluation data for network anomaly detection" in Recent Advances in Intrusion Detection, Berlin, Germany:Springer, vol. 2820, pp. 220-237, 2003.

[2] M. Sabhnani and G. Serpen, "Why machine learning algorithms fail in misuse detection on KDD intrusion detection data set", Intell. Data Anal., vol. 8, no. 4, pp. 403-415, 2004.

[3] Y. Bouzida and F. Cuppens, "Neural networks vs. decision trees for intrusion detection", Proc. IEEE/IST Workshop Monitoring Attack Detection Mitigation (MonAM), pp. 1-29, Sep. 2006.

[4] Ajani, S., Amdani, S.Y. (2022). Obstacle Collision Prediction Model for Path Planning Using Obstacle Trajectory Clustering. In: Sharma, S., Peng, SL., Agrawal, J., Shukla, R.K., Le, DN. (eds) Data, Engineering and Applications. Lecture Notes in Electrical Engineering, vol 907. Springer, Singapore. https://doi.org/10.1007/978-981-19-4687-5_8

[5] X. Glorot, A. Bordes and Y. Bengio, "Deep sparse rectifier neural networks", Proc. 14th Int. Conf. Artif. Intell. Statist., pp. 315-323, 2011.

[6] A. Alazab, M. Hobbs, J. Abawajy and M. Alazab, "Using feature selection for intrusion detection system", Proc. Int. Symp. Commun. Inf. Technol. (ISCIT), pp. 296-301, Oct. 2012.

[7] S. N. Ajani and S. Y. Amdani, "Probabilistic path planning using current obstacle position in static environment," 2nd International Conference on Data, Engineering and Applications (IDEA), 2020, pp. 1-6, doi: 10.1109/IDEA49133.2020.9170727.

[8] G. Creech and J. Hu, "A semantic approach to host-based intrusion detection systems using contiguousand discontiguous system call patterns", IEEE Trans. Comput., vol. 63, no. 4, pp. 807-819, Apr. 2014.

[9] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", Proc. Int. Conf. Mach. Learn., pp. 448-456, 2015.

[10] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)", Proc. IEEE Mil. Commun. Inf. Syst. Conf. (MilCIS), pp. 1-6, Nov. 2015.

[11] Manzoor MA, Morgan Y. Real-time support vector machine based network intrusion detection system using Apache Storm. In: IEEE 7th annual information technology, electronics and mobile communication conference (IEMCON), 2016. Piscataway: IEEE. 2016; p. 1–5.

[12] Wang H, Xiao Y, Long Y. Research of intrusion detection algorithm based on parallel SVM on Spark. In: 7th IEEE International conference on electronics information and emergency communication (ICEIEC), 2017. Piscataway: IEEE; 2017. p. 153–156.

[13] Vimalkumar K, Radhika N. A big data framework for intrusion detection in smart grids using Apache Spark. In: International conference on advances in computing, communications and informatics (ICACCI), 2017. Piscataway: IEEE; 2017. p. 198–204.

[14] W. Wang *et al.*, "HAST-IDS: Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection", IEEE Access, vol. 6, pp. 1792-1806, 2018.

[15] Ajani, S.N., Amdani, S.Y. (2021). Agent-Based Path Prediction Strategy (ABPP) for Navigation Over Dynamic Environment. In: Muthu Kumar, P., Sarkar, D.K., De, D., De, C.K. (eds) Innovations in Sustainable Energy and Technology. Advances in Sustainability Science and Technology. Springer, Singapore.

[16] M. N. Kurt, Y. Yılmaz and X. Wang, "Real-time detection of hybrid and stealthy cyber-attacks in smart grid", IEEE Trans. Inf. Forensics Security, vol. 14, no. 2, pp. 498-513, Feb. 2019.

[17] Ali Sadiqui, "Putting in Place an Intrusion Prevention System (IPS)," in Computer Network Security, Wiley, 2020, pp.101-124, doi: 10.1002/9781119706762.ch6.

[18] Ariani and M. Salman, "Modeling Study of Priority Intrusion Response Selected on Intrusion Detection System Alert," 2020 6th International Conference on Science and Technology (ICST), Yogyakarta, Indonesia, 2020, pp. 1-6, doi: 10.1109/ICST50505.2020.9732867.

[19] S. Bhadauria and T. Mohanty, "Hybrid Intrusion Detection System using an Unsupervised method for Anomaly-based Detection," 2021 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), Hyderabad, India, 2021, pp. 1-6, doi: 10.1109/ANTS52808.2021.9936919.

[20] Prasanthi Rathnala, M.S. Pradeep Kumar Patnaik, Srinivasa Rao Sura, Bolla Prasad, N Siva Mallikarjuna Rao and Delione N Rayan (2022), Design of an Efficient Face Recognition system using Deep Learning Technique. IJEER 10(3), 689-693. DOI: 10.37391/IJEER.100345.

[21] K. Sood, M. R. Nosouhi, D. D. N. Nguyen, F. Jiang, M. Chowdhury and R. Doss, "Intrusion Detection Scheme With Dimensionality Reduction in Next Generation Networks," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 965-979, 2023, doi: 10.1109/TIFS.2022.3233777.

[22] Shengjie Xu; Yi Qian; Rose Qingyang Hu, "Edge Intelligence for Intrusion Detection," in Cybersecurity in Intelligent Networking Systems , IEEE, 2023, pp.45-54, doi: 10.1002/9781119784135.ch4.

[23] M. L. Han, B. I. Kwak and H. K. Kim, "TOW-IDS: Intrusion Detection System Based on Three Overlapped Wavelets for Automotive Ethernet," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 411-422, 2023, doi: 10.1109/TIFS.2022.3221893.

[24] https://www.unb.ca/cic/datasets/ids-2017.html