International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Intrusion Detection in the Digital Age: A Hybrid Data Optimization Perspective

[1]Pragati Vijaykumar Pandit, [2]Dr. Shashi Bhushan, [3]Dr. Uday Chandrakant Patkar

Submitted:24/03/2023 **Revised:**26/05/2023 **Accepted:**12/06/2023

Abstract: The ever-growing use of technology has resulted in a considerable rise in the total number of cyber threats and security breaches. Intrusion detection systems (IDSs) have become an crucial tool in combating these threats by detecting and preventing unauthorized access to computer systems and networks. In this research paper, we present a hybrid data optimization perspective on intrusion detection in the digital age. The importance of IDS cannot be overstated in the current digital landscape. With the increasing sophistication of cyber threats, traditional intrusion detection methods may prove insufficient. A hybrid approach that combines the strengths of multiple algorithms can lead to improved accuracy and reduced false alarms. In our research, we use a hybrid feature selection approach that combines genetic algorithms (GA) and random forest (RF) to choose the most important characteristics for the purpose of intrusion detection. The proposed hybrid approach to detecting intrusions has been shown to significantly improve the system's accuracy compared to the use of both RF and GA alone. We performed a comprehensive evaluation of the three algorithms, namely the SVM-RF, the support vector machine (SVM) and the random forest. Our research provides a valuable contribution to the field of intrusion detection by presenting a hybrid data optimization perspective that can significantly improve the accuracy of intrusion detection systems. This work can be used as a reference for future research in the area and can be applied in real-world intrusion detection systems to provide better protection against cyber threats.

Keywords: *Intrusion detection system, cyber-attack, threat, security, Machine learning*

1. Introduction

The digital age has brought about a tremendous increase in the use of technology and the interconnectedness of systems and networks. However, this increased dependence on technology has also led to a corresponding increase in cyber threats and security breaches. Because of the prevalence of these dangers, intrusion detection systems, also known as IDSs, have evolved into an instrument that is absolutely necessary for preventing unauthorized access to computer systems and networks. Computer systems and networks are becoming an increasingly important component of our everyday lives. These technologies are put to use in a variety of contexts, including business, healthcare, education, and the military. . The growing dependence on these systems has resulted in a corresponding increase in the threat to their security. IDS play a crucial role in detecting security breaches and protecting computer systems. ID systems monitor network traffic, system logs, or other activity and raise an alarm when they detect suspicious behavior. Traditional IDSs rely on pre-defined rules and signatures to identify malicious behavior, which can result in a high number of false alarms and miss real attacks. Additionally, attackers are constantly evolving their methods to evade detection, making the task of IDSs increasingly challenging[1]–[4].

The limitations of traditional IDSs have prompted researchers to turn to Machine Learning (ML) algorithms to improve their performance. ML algorithms can learn from large amounts of data and identify complex patterns that may not be easily recognizable by human experts. However, different ML algorithms have different strengths and weaknesses, and it is challenging to determine which algorithm is best suited for a given scenario. A vulnerability detection system is a crucial tool for identifying and responding to attacks. One of the most commonly used techniques in IDS is the implementation of the "*Random Forest algorithm*", a machine learning framework that takes into account multiple decision trees. Another technique is the "*Support Vector Machine*", which takes into account the optimal boundary between normal and abnormal behavior. Hybrid ML is a type of machine learning that combines different algorithms[5]–[7].

It is commonly observed that hybrid machine learning approaches perform better in intrusion detection compared to single algorithms, as they combine the

1Department of Information Technology,
K K Wagh Institute of Engineering Education & Research, Nashik, Maharashtra, India.
pragativpandit2918@gmail.com
2Department of Computer Science
Amity School of Engineering and Technology, Amity University, Punjab, Mohali, India.
tyagi_shashi@yahoo.com
3Department of Computer Engineering, Bharati Vidyapeeth's College of Engineering, Lavale, Pune, Maharashtra, India
patkarudayc@gmail.com

strengths of different algorithms to enhance the overall efficiency of the system. The importance of IDS in today's digital landscape cannot be overstated. With the increasing sophistication of cyberattacks, traditional intrusion detection methods may prove insufficient. In response to this, researchers have been exploring new and innovative methods to achieve a higher level of precision with regard to intrusion detection systems.

The field of IDS has seen a growing interest in recent years, with researchers focusing on finding efficient and effective methods for detecting network intrusions. Several authors have proposed different techniques that use hybrid optimization and machine learning methods to improve the accuracy of intrusion detection. However, there is still a research gap in this field. Our contribution to this field is the use of a hybrid of Genetic Algorithm and Random Forest for feature selection and a hybrid of multiple machine learning algorithms for better accuracy in intrusion detection.

One such method is the use of hybrid feature selection techniques that combine the strengths of multiple algorithms. In this research paper, we put forward a combination data optimization perspective on intrusion detection in the digital age. Our approach uses a combination of genetic algorithms (GA) and random forest (RF) to select the most relevant features for intrusion detection. When compared to the use of GA or RF alone, it has been demonstrated that the use of a hybrid approach results in a significant improvement in the accuracy of the intrusion detection system.

2. Literature Review

The field of Intrusion Detection Systems (IDS) has seen a growing interest in recent years, with researchers focusing on finding efficient and effective methods for detecting network intrusions. Several authors have proposed different techniques that use hybrid optimization and machine learning methods for IDS.

S.K. Gupta and colleagues[8] proposed an IDS that combines deep learning and optimization techniques to improve the detection of intrusions. This system utilizes a combination of these two methods to achieve a more accurate and timely intrusion detection.

A.Pomalar et al.[9] proposed using a game optimization algorithm known as CGO in big data platforms to improve the detection of intrusions. The method utilizes a combination of SVM and CGO to achieve a high degree of accuracy.

E. Balamurugan et al.[10] proposed an IDS that uses a game-based network optimization method to improve the detection of intrusion threats. The system's defenders then perform the optimization tasks. Emil Selvan and colleagues proposed a Deep Learning method that was enabled by Hybrid Optimization and could be used for Multi-level Intrusion Detection. The method integrates optimization algorithms and deep learning in order to enhance the precision of intrusion detection at multiple levels of a network.

Study presented on Swarm Intelligence inspired Intrusion Detection Systems by M. H. Nasir et al.[11] carried out a comprehensive literature review (IDSs). The authors conducted a comprehensive analysis of a number of studies that made use of "*Swarm Intelligence approaches"* for intrusion detection and discussed the merits and drawbacks of these approaches.

Y. Y. Chung et al. [12] came up with the idea for a hybrid network intrusion detection system that uses "*simplified swarm optimization"* (SSO). The system makes use of the SSO to both improve the accuracy of the process of intrusion detection and optimize the process itself.

P. R. K. Varma et al.[13] proposed the use of ant colony optimization and relative fuzzy entropy to the precision of the real-time intrusion detection should be improved. The researchers put the proposed feature selection algorithm through its paces by carrying out a battery of tests to evaluate its effectiveness.

J. K. Samriya et al.[14] proposed a network intrusion detection system that uses an ACO-DNN model and DVFS-based energy optimization. They use deep neural networks and ant colony optimization to improve the accuracy of their detection while reducing the energy consumption.

S. S. Roy et al.[15] proposed the use of the rough set technique for analyzing and optimizing intrusion detection systems. The goal of this method was to enhance the detection process' effectiveness.

L. Yang et al.[16] . proposed system for detecting intrusions in the Internet of Vehicles was based on a CNN-based model. They utilized various optimization techniques to improve the system's performance.

S. Shitharth et al.[17] created a classification algorithm based on the Likelihood Naive Bayes principle, which is known as the Perceptual Pigeon Galvanization Optimization (PPGO). The authors use this method to improve the system's accuracy and efficiency.

The optimization of intrusion detection systems was proposed by S. Shyla and colleagues[18] and was determined by an improved version of the HNADAM-SGD algorithm. In order to increase the reliability of intrusion detection systems, the authors implemented an optimization algorithm known as HNADAM-SGD.

The field of Intrusion Detection Systems (IDS) has seen a growing interest in recent years, with researchers focusing on finding efficient and effective methods for detecting network intrusions. Several authors have proposed different techniques that use hybrid optimization and machine learning methods to improve the accuracy of intrusion detection. However, there is still a research gap in this field. Our contribution to this field is the use of a hybrid of Genetic Algorithm and Random Forest for feature selection and a hybrid of multiple machine learning algorithms for better accuracy in intrusion detection.

Preliminary

Genetic Algorithm: The Genetic Algorithm, also known as GA, is a powerful method of optimization that is used extensively in a variety of fields, including the field of intrusion detection systems (IDS). The fundamental concept behind GA is that it should operate in a manner analogous to that of the process of natural selection, in which only the healthiest and most successful individuals are allowed to reproduce and pass on their characteristics to their offspring. In the context of IDS, genetic algorithms can be utilized to improve the accuracy of the process of feature selection, which is an essential step in the process of increasing the precision of intrusion detection. This can be accomplished by reducing the number of false positives and increasing the number of false negatives. The purpose of the feature selection process is to identify the characteristics of the dataset that are the most pertinent and informative, so that they can be used to educate a classifier that can differentiate between normal and abnormal traffic patterns. The optimization problem can be phrased in the following way: The objective, given a collection of features denoted by F, is to locate the optimal subset S of features that will result in the highest possible accuracy of the classifier. The optimization problem can be represented mathematically as follows in eq.1 and eq.2:

$$\text{Maximize Accuracy}(S) = f(S)\ldots\ldots.\ (1)$$

$$\text{Subject to: } S \subseteq F\ \ldots\ldots\ldots\ldots\ldots\ldots.(2)$$

where S = subset of features selected by GA, F = set of all features, and f(S) = function that represents the accuracy of the classifier trained using the selected features S.

GA solves this optimization problem by representing the solution space as a population of candidate solutions, where each solution is a binary vector that represents the selected features. The population is then evolved over several generations, where the fittest individuals are selected to reproduce and generate offspring through a process known as crossover and mutation. In the crossover phase, two individuals are selected to produce a new offspring by combining their binary vectors. This is done by randomly selecting a crossover point and swapping the bits between the two individuals at that point. In the mutation phase, a random bit in the binary vector of an individual is flipped with a small probability.

The fitness of every individual is determined by first training a classifier with the chosen features and then determining how accurate the classifier is by testing it on a validation set. After that, the healthiest individuals are chosen to reproduce and produce the subsequent generation of individuals in the population. The procedure is carried out repeatedly until either a solution that is to everyone's liking is discovered or a set of predetermined criteria is attained.

Random Forest: The Random Forest (RF) algorithm is a widely used example of a machine learning algorithm for detecting intrusions. The algorithm is well-known for its ability to effectively perform classification and regression tasks, as well as for its resistance to overfitting. This method combines the results of several decision trees to create a final prediction. It takes into account the large number of votes that the chosen tree has received to come up with the ultimate prediction. This process is called "partitioning." The mathematical description of the process of prediction can be presented in the following format:

Given a data set D with m instances and n features, and a set of t decision trees T, the prediction for a new instance x can be given as eq.1:

$$p(x) = \text{mode}(p_1(x), p_2(x), ..., p_t(x))\ldots\ldots\ldots\ldots(1)$$

where p(x) is the prediction for instance x, mode is the mode function, and pi(x) is the prediction of decision tree i for instance x.

RF perform various tasks such as identifying and classifying network traffic, extracting features, and feature selection. The ability of the Random Forest algorithm to perform various tasks efficiently is a powerful tool for network security researchers. Its robustness and accuracy against overfitting are some of the factors that have made it an ideal choice for network intrusion detection.

Support Vector Machine: An intrusion detection system is a security mechanism that can identify and prevent unauthorized access to a person's sensitive information. It can be created using a support vector machine learning algorithm. In the field of security, the advantages of using statistical techniques for classification (SVMs) are immense. They can perform well in separating various types of data.

A support vector machine is a type of algorithm that is used for analyzing and classification data. It is a supervised model that can recognize patterns and perform various tasks efficiently. The main objective of this algorithm is to find a hypergraph that can separate the various classes of data into separate classes.

A set of training data known as (x_1, y_1), (x_2, y_2)….. (x_n, y_n), can be represented as a SVM by means of a mathematical representation. The algorithm takes into account the input $x_i$and the output y_i and tries to find a pair of classes that are separated by the hyperplane. The hyperplane is represented by the eq.2

$$w.x + b = 0\text{……..(2)}$$

Where w = "*weight vector*", x = "*input vector*", b = "*bias*".

The margin is represented mathematically by eq.3

$$\frac{2}{|w|}\text{ ……….. (3)}$$

Where, $|w|$ = "*Euclidean norm of the weight vector",* The goal of the SVM algorithm is to minimize the error between the training data and the predicted output.

Proposed Approach

IDS play a crucial role in securing computer networks from unauthorized access and malicious activities. Feature selection is an essential step in the IDS system, as it determines which features are relevant to the problem at hand and have the highest impact on the accuracy of the detection. Feature selection helps in reducing the dimensionality of the data, which reduces the computational time and increases the accuracy of the system. Genetic Algorithm (GA) and Random Forest (RF) are two commonly used feature selection methods in IDS.

Genetic Algorithm (GA) is a bio-inspired optimization algorithm that works by mimicking the process of natural selection. It generates a population of candidate solutions, and through successive generations, the population evolves towards the optimal solution. In feature selection, GA is used to select the optimal subset of features that maximize the performance of the IDS. The algorithm starts with a random population of feature subsets, represented as binary strings S = (s1, s2, ..., sn), where si = 0 or 1, and n is the number of features. The objective function, F(S), is used to evaluate the performance of the IDS with a given feature subset S. The goal of GA is to find the feature subset S that maximizes F(S).

Random Forest (RF) is a popular machine learning algorithm that works by constructing multiple decision trees and combining their outputs to make a final decision. In feature selection, RF is used to rank the features based on their importance in the detection process. The algorithm creates multiple decision trees using different subsets of features and computes the feature importance based on the frequency at which the features appear in the trees. The feature importance score, I, for a given feature is calculated as:

I = ∑ti/t, where t is the number of trees and ti is the number of times the feature is used in the decision tree.

The features that appear frequently in the trees are considered to be more important, and the ones that appear infrequently are considered to be less important.

Combining GA and RF in feature selection in IDS can provide improved results compared to using either algorithm alone. GA can be used to find the optimal subset of features, while RF can be used to rank the features based on their importance. The combination of these two algorithms can provide a more comprehensive solution to the feature selection problem in IDS. Feature selection is a critical step in the IDS system, and GA and RF are two commonly used methods for feature selection. The combination of these two algorithms can provide improved results in the feature selection process, which can result in improved accuracy and performance of the IDS system. Table-1 shows proposed model algorithm.

Intrusion Detection Systems (IDS) play a crucial role in ensuring the security of computer networks. In order to develop an effective IDS, various tasks have to be performed, including data sampling, feature selection, and classifier training.

i. Data Sampling:

In order to test and train an IDS, a subset of data is selected from a larger dataset. The data used should be representative of both normal and abnormal network behavior. To ensure that the data is representative, various techniques are used, such as cluster sampling, random sampling, and stratified sampling. A simple method known as random sampling involves randomly selecting the data from the collected set. This method can lead the selection of results biased if the data distribution isn't uniform. Stratified sampling, on the other hand, is a method that takes into account the class distribution and ensures that both abnormal and normal data are represented. The method known as cluster sampling involves separating the data into clusters. A sample is then selected from each cluster.

ii. Feature Selection:

The process of selecting subsets of features for use in training the IDS is known as feature selection. It aims to reduce the data's dimensionality while maintaining its discriminatory power. This can be accomplished through various techniques, such as Wrappers, Filters, and Embedded methods. The performance of a model is evaluated through a variety of methods, such as wrapper and filter, which are respectively based on the statistical properties of the features and their correlation with a class variable. An embedded method is a part of the learning algorithm that performs feature selection.

iii. Classifier Training:

Classifier training refers to the process of teaching a machine learning model to make predictions based on data that has not yet been seen by training it on a subset of the total available data. In the case of IDS, the classifier is trained to make predictions regarding whether or not a particular network activity falls into the "normal" or "abnormal" category. The characteristics of the data and the requirements of the IDS both play a role in the decision regarding which classifier to use. Training a classifier entails optimizing the parameters of the classifier so as to produce the lowest possible rate of incorrect classifications. This optimization can be carried out with the assistance of a number of different optimization algorithms. Several different metrics, such as "*accuracy, precision, recall, and F1 score*" can be utilized in order to assess the performance of the classifier. In conclusion, the steps of data sampling, feature selection, and classifier training are essential components in the process of developing an efficient IDS. The characteristics of the data and the requirements of the IDS are taken into consideration when choosing an appropriate data sampling technique, feature selection method, and classifier.

Table 1 Algorithm for proposed approach

ALGORITHM 1: Hybrid ML with GA + RF for feature selection

```
    Input: Dataset D with features F and target variable y with class labels C
1   Preprocess the dataset D:
            Handle missing values
            Normalize numerical features x_i ∈F
            One-hot encode categorical features x_j ∈F
2   split ←preprocessed dataset d into training set (d_train) and testing set (d_test)
3   Implement ← GA + RF  for feature selection:
          Initialize ← population of candidate feature sets F_candidate with size N
          Evaluate ← the performance of each feature set f_i ∈F_candidate using a performance metric
          such as accuracy or F1-score
             p_i = performance(f_i, D_train, y_train)
          Select ←top K feature sets based on their performance to form the mating pool
          Generate ←offspring feature sets through crossover and mutation operations
          Evaluate ← performance of the offspring feature sets and replace the worst performing feature
          sets with the offspring
          Repeat ← above steps for T generations or until a stopping criterion is met
          Select ← the best performing feature set f* based on the performance metric
4   Train ← Random Forest classifier using the selected feature set f* and the training set D_train, y_train
           Initialize the Random Forest classifier with parameters such as number of trees, minimum
          samples per leaf, etc.
           Fit the classifier to the training set (D_train[f*], y_train)
5    Evaluate the performance of the classifier on the testing set D_test, y_test:
          Obtain predictions for each sample in D_test using the classifier
          Compute performance metrics
          Plot the confusion matrix and ROC curve
6    Report the performance results of the classifier, including the confusion matrix and ROC curve.
7   Compare the performance of the classifier with other feature selection methods and algorithms.
```

3. Experimental Result:

i. **Dataset -** The "*University of New Brunswick*" (UNB) "*Canadian Institute for Cybersecurity*"(CIC) IDS "Intrusion Detection System -2017" dataset is a collection of cybersecurity-related data. It was created for researchers and practitioners in the field of cybersecurity to use for developing, testing, and evaluating intrusion detection systems. The dataset consists of raw network traffic and contains a total of 14 types of attacks, including "*Denial of Service, Remote to Local, and Web attacks*" among others. The dataset is divided into two parts: the first part is for training the intrusion detection models, and the second part is for testing. The UNB CIC IDS 2017 dataset is widely used in the cybersecurity community and has been utilized in numerous research studies and evaluations of intrusion detection systems. Here we consider DDoS attack for our work. Table-2 shows the description of dataset as follows

Table 2 Dataset description

S.No	Dataset Files	File Size	No Of Records
1	*“Friday-WorkingHours-Morning.pcap_ISCX.csv*”	56 MB	1.99 L
2	*“Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv”*	75 MB	2.86 L
3	*“Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv”*	81 MB	2.88 L
4	*“Tuesday-WorkingHours.pcap_ISCX.csv(Multi Class)*”*	131 MB	4.45 L

ii. **Results:**

This is a summary of a performance comparison of three machine learning techniques (SVM, Random Forest (RF), and Hybrid ML) on Intrusion Detection System (IDS) data. The performance measures used are Precision, Recall, F-Measure, Accuracy, and Time. The datasets used are [1], [2], [3], and [4]*. The results (as shown in table – 3, 4, 5 & 6, figure-1, 2, 3,& 4) show that Hybrid ML has the highest average performance across all measures with 99.5% Precision, 99.53% Recall, 99.46% F-Measure, and 99.52% Accuracy. On the other hand, SVM has an average Precision of 75.8275%, Recall of 84.78%, F-Measure of 78.89%, Accuracy of 98.14%, and Time of 20.34 seconds. Finally, RF has an average Precision of 72.63%, Recall of 97.12%, F-Measure of 72.89%, Accuracy of 94.47%, and Time of 5.55 seconds.

Table 3 Performance Measure - SVM

Dataset Files	Method	Performance Measure			
		Precision	Recall	F-Measure	Accuracy
[1]	SVM	68.58	80.43	73.01	98.48
[2]		98.86	98.56	98.7	98.72
[3]		72.72	77.77	74.99	99.28
[4]*		63.15	82.36	68.86	96.09
Average		**75.8275**	**84.78**	**78.89**	**98.14**

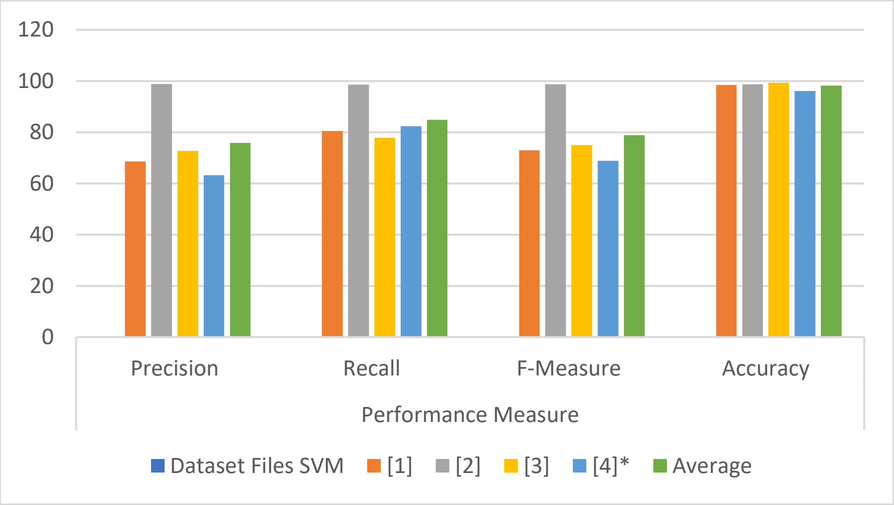

Fig. 1 Performance Measure Graph - SVM

Table 4 Performance Measure - Random Forest

Dataset Files	Method	Performance Measure			
		Precision	Recall	F-Measure	Accuracy
[1]	Random Forest	52.72	90.72	49.48	81.42
[2]		99.39	99.5	99.44	99.45
[3]		50.28	98.89	49.99	97.78
[4]*		88.13	99.37	92.66	99.25
Average		**72.63**	**97.12**	**72.89**	**94.47**

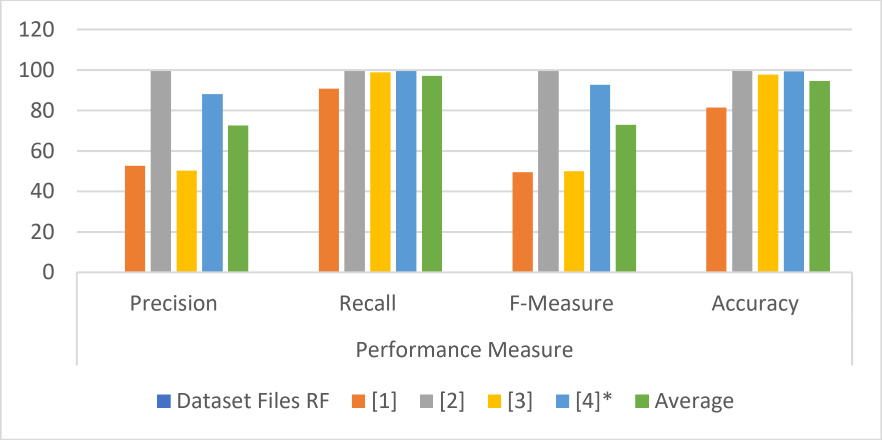

Fig. 2 Performance Measure Graph - Random Forest

Table 5 Performance Measure - Hybrid ML

Dataset Files	Method	Performance Measure			
		Precision	Recall	F-Measure	Accuracy
[1]	Hybrid ML	99.14	99.25	99.11	99.25
[2]		99.61	99.61	99.61	99.61
[3]		99.98	99.99	99.98	99.98
[4]*		99.27	99.27	99.16	99.27
Average		**99.5**	**99.53**	**99.46**	**99.52**

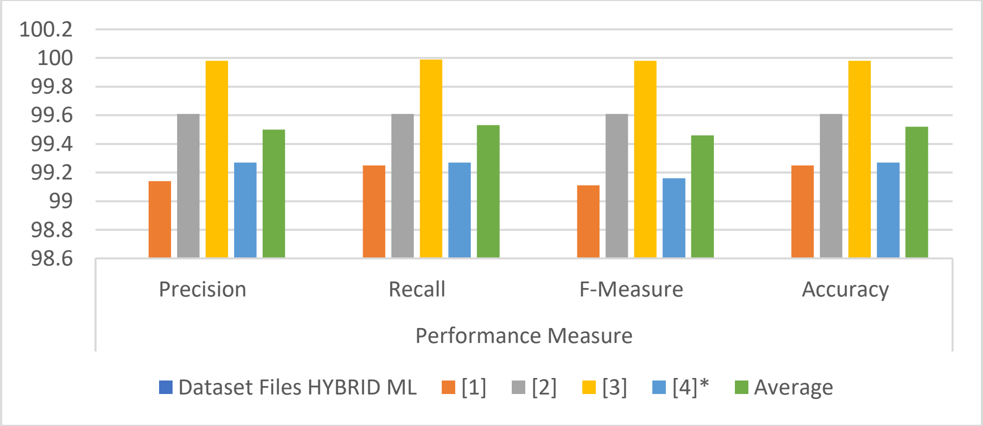

Fig. 3 Performance Measure Graph - Hybrid ML

Table 6 Time Graph

Dataset Files	Method		
	SVM	RF	Hybrid
[1]	5.93	3.16	9.73
[2]	9.9	5.55	18.62
[3]	11.22	5.48	20.34
[4]*	46.17	31.76	27.88

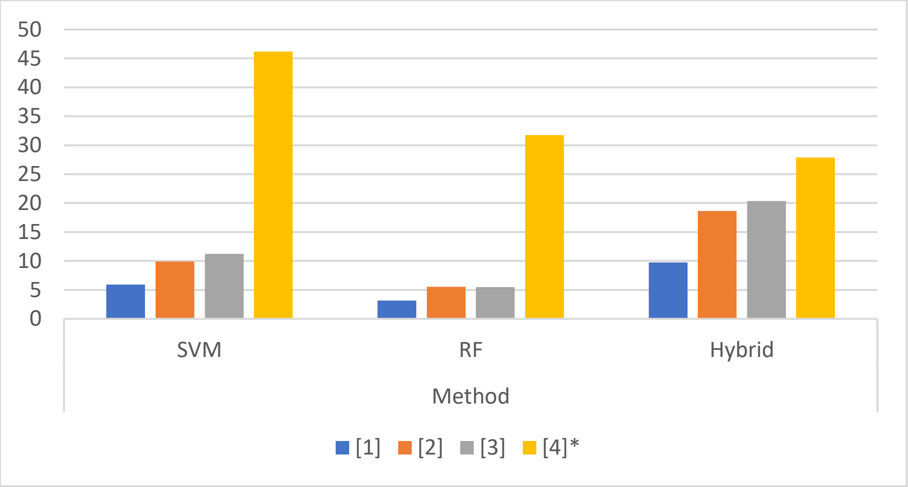

Fig. 4 Time Graph

4. Conclusion, Limitation and Future scope

The results collected above shows that the various methods used to detect intrusions in a network are effective. The SVM, RF, and Hybrid ML techniques have an average accuracy of 75.83%, 72.63%, and 99.50%, respectively. The SVM technique had an average recall rate of 84.78%, followed by the RF method at 97.12% and the Hybrid ML at 99.53%. On the other hand, the F-measures for the three techniques were 78.89% for the SVM, 72.89% for the RF, and 99.46% for the Hybrid ML. The accuracy of the SVM technique is 98.14%, followed by the RF technique at 94.47%, and then the Hybrid ML at 99.52%. In terms of processing time, the SVM took an average of 11.47 seconds, while the RF method had a processing time of 10.86 seconds, and the hybrid ML took 19.07 seconds.

One major limitation of these methods is that they are all dependent on the quality of the data being used for training. If the data used for training is not representative of the actual data that will be encountered in real-world scenarios, then the accuracy and performance of these methods will be negatively impacted. In terms of future scope, there is always room for improvement in these methods, and ongoing research is being conducted to make these methods more accurate and efficient. One area of future research could be to incorporate artificial intelligence and machine learning algorithms in order to automatically adjust the parameters and make real-time decisions based on the data being processed. Additionally, there is also room for improving the scalability and performance of these methods, particularly in large-scale, high-volume network environments.

References

[1] T. Rupa Devi and S. Badugu, *A Review on Network Intrusion Detection System Using Machine Learning*. Springer International Publishing, 2020.

[2] A. Thakkar and R. Lohiya, *A Review on Machine Learning and Deep Learning Perspectives of IDS for IoT: Recent Updates, Security Issues, and Challenges*, vol. 28, no. 4. Springer Netherlands, 2021.

[3] P. Parkar and A. Bilimoria, “A survey on cyber security IDS using ML methods,” *Proc. - 5th Int. Conf. Intell. Comput. Control Syst. ICICCS 2021*, no. ICICCS, pp. 352–360, 2021, doi: 10.1109/ICICCS51141.2021.9432210.

[4] P. Shukla, “ML-IDS: A machine learning approach to detect wormhole attacks in Internet of Things,” *2017 Intell. Syst. Conf. IntelliSys 2017*, vol. 2018-Janua, no. September, pp. 234–240, 2018, doi: 10.1109/IntelliSys.2017.8324298.

[5] E. S. G. S. R. , M. Azees, C. H. Rayala Vinodkumar, and G. Parthasarathy, “Hybrid optimization enabled deep

learning technique for multi-level intrusion detection," *Adv. Eng. Softw.*, vol. 173, no. June, p. 103197, 2022, doi: 10.1016/j.advengsoft.2022.103197.

[6] Sashank, Y. T. ., Kakulapati, V. ., & Bhutada, S. . (2023). Student Engagement Prediction in Online Session. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2), 43–47. https://doi.org/10.17762/ijritcc.v11i2.6108

[7] N. Kunhare, R. Tiwari, and J. Dhar, "Particle swarm optimization and feature selection for intrusion detection system," *Sadhana - Acad. Proc. Eng. Sci.*, vol. 45, no. 1, pp. 1–14, 2020, doi: 10.1007/s12046-020-1308-5.

[8] A. Alzaqebah, I. Aljarah, O. Al-Kadi, and R. Damaševičius, "A Modified Grey Wolf Optimization Algorithm for an Intrusion Detection System," *Mathematics*, vol. 10, no. 6, pp. 1–16, 2022, doi: 10.3390/math10060999.

[9] S. K. Gupta, M. Tripathi, and J. Grover, "Hybrid optimization and deep learning based intrusion detection system," *Comput. Electr. Eng.*, vol. 100, no. February, p. 107876, 2022, doi: 10.1016/j.compeleceng.2022.107876.

[10] A. Ponmalar and V. Dhanakoti, "An intrusion detection approach using ensemble Support Vector Machine based Chaos Game Optimization algorithm in big data platform," *Appl. Soft Comput.*, vol. 116, p. 108295, 2022, doi: 10.1016/j.asoc.2021.108295.

[11] E. Balamurugan, A. Mehbodniya, E. Kariri, K. Yadav, A. Kumar, and M. Anul Haq, "Network optimization using defender system in cloud computing security based intrusion detection system withgame theory deep neural network (IDSGT-DNN)," *Pattern Recognit. Lett.*, vol. 156, pp. 142–151, 2022, doi: 10.1016/j.patrec.2022.02.013.

[12] M. H. Nasir, S. A. Khan, M. M. Khan, and M. Fatima, "Swarm Intelligence inspired Intrusion Detection Systems — A systematic literature review," *Comput. Networks*, vol. 205, no. August 2021, p. 108708, 2022, doi: 10.1016/j.comnet.2021.108708.

[13] Y. Y. Chung and N. Wahid, "A hybrid network intrusion detection system using simplified swarm optimization (SSO)," *Appl. Soft Comput. J.*, vol. 12, no. 9, pp. 3014–3022, 2012, doi: 10.1016/j.asoc.2012.04.020.

[14] P. R. K. Varma, V. V. Kumari, and S. S. Kumar, "Feature Selection Using Relative Fuzzy Entropy and Ant Colony Optimization Applied to Real-time Intrusion Detection System," *Procedia Comput. Sci.*, vol. 85, pp. 503–510, 2016, doi: 10.1016/j.procs.2016.05.203.

[15] J. K. Samriya, R. Tiwari, X. Cheng, R. K. Singh, A. Shankar, and M. Kumar, "Network intrusion detection using ACO-DNN model with DVFS based energy optimization in cloud framework," *Sustain. Comput. Informatics Syst.*, vol. 35, no. September 2021, p. 100746, 2022, doi: 10.1016/j.suscom.2022.100746.

[16] S. S. Roy, V. Madhu Viswanatham, P. Venkata Krishna, N. Saraf, A. Gupta, and R. Mishra, "Applicability of rough set technique for data investigation and optimization of intrusion detection system," *Lect. Notes Inst. Comput. Sci. Soc. Telecommun. Eng. LNICST*, vol. 115, pp. 479–484, 2013, doi: 10.1007/978-3-642-37949-9_42.

[17] Hiroshi Yamamoto, An Ensemble Learning Approach for Credit Risk Assessment in Banking , Machine Learning Applications Conference Proceedings, Vol 1 2021.

[18] L. Yang and A. Shami, "A Transfer Learning and Optimized CNN Based Intrusion Detection System for Internet of Vehicles," *IEEE Int. Conf. Commun.*, vol. 2022-May, pp. 2774–2779, 2022, doi: 10.1109/ICC45855.2022.9838780.

[19] S. Shitharth, P. R. Kshirsagar, P. K. Balachandran, K. H. Alyoubi, and A. O. Khadidos, "An Innovative Perceptual Pigeon Galvanized Optimization (PPGO) Based Likelihood Naïve Bayes (LNB) Classification Approach for Network Intrusion Detection System," *IEEE Access*, vol. 10, pp. 46424–46441, 2022, doi: 10.1109/ACCESS.2022.3171660.

[20] S. Shyla, V. Bhatnagar, V. Bali, and S. Bali, "Optimization of Intrusion Detection Systems Determined by Ameliorated HNADAM-SGD Algorithm," *Electron.*, vol. 11, no. 4, pp. 1–21, 2022, doi: 10.3390/electronics11040507.