Intrusion Detection System Using Machine Learning

SIVASUBRAMANIAN M|21MCA1103Dr PRASAD MVIT Chennai,India

***Abstract***—Network foundation assaults are right now the most genuine dangers to arrange and data security. Since standard firewall procedures can't give absolute security against interruption notwithstanding persistently rising criminal operations in networks, interruption discovery (ID) as a part of protection inside and out is turning out to be progressively significant. The most typical method of identifying events occurring in a PC system or network and decomposing them for interruption alerts (IDS) is interruption location framework. In this project, various methodologies like Decision tree algorithm, Randomforest algorithm, Xgboost algorithm etc. are used to figure an exact model for interruption recognition framework. In a comparison of decision tree, random forest, and decision tree and random forest with AdaBoost, the suggested models with the best accuracy are those.

***Keywords*—** *Intrusion detection, cyber attacks, Machine Learning Algorithms,SMOTE.*

1. INTRODUCTION

An instrument called an intrusion detection system (IDS) is used to keep an eye out for any strange behaviour and notify the security team of any possible threats. The incident responder or Security Operations Centre (SOC) analyst then looks into the problem and takes the required steps to reduce the risk. IDS is one of the many cybersecurity solutions available and is similar to home invasion detection systems in that it can be host-based or network-based.

* A security technology known as a Host-Based Intrusion Detection System (HIDS) is put on a particular endpoint in order to guard it from both internal and external threats. In addition to monitoring network traffic to and from the host computer, this type of IDS can also keep an eye on running processes and peruse system logs. A HIDS can only monitor its host system, but even so, it offers extensive insight into the workings of the computer, giving decision-makers useful information.
* A security technology called a network-based intrusion detection system (NIDS) is made to keep track of a whole protected network. This kind of IDS has the capacity to keep an eye on all network traffic and can draw conclusions from packet metadata and contents. The identification of pervasive dangers is made possible by the larger perspective, which offers more context. However, the internals of the endpoints that NIDS solutions protect are not visible to them

An intrusion detection system (IDS), a kind of security technology, continually examines network data for any odd activity and alerts administrators if it is discovered. The IDS software application scans an organization's or system's data for signs of unauthorized access or any other potentially harmful activity. If any such activity is detected, the IDS generates alerts that can be viewed by the security team. The alerts are usually analyzed further by a security information and event management (SIEM) platform, which collects data from multiple sources and uses various algorithms to differentiate between genuine attacks and false alarms. While IDS systems are capable of monitoring networks for malicious activity, they can also generate false alerts. Hence, it is crucial for organizations to customize their IDS products by configuring the systems to differentiate between normal network traffic and suspicious activity. IDS systems also include intrusion prevention capabilities that inspect network packets as they enter the system to identify any malicious activity and generate real-time alerts. Convolutional neural networks, deep neural networks, and long short-term memory networks are only a few examples of the many artificial neural networks that exist, are utilized to detect and classify different types of attacks. This paper specifically focuses on detecting seven types of attacks: BENIGN, DoS, PortScan, BruteForce, WebAttack, Bot, and Infiltration.

* Begin:Both machine learning and human intelligence depend on classification. Correctly classifying data items into a given range of potential classes is the goal. Malware classification is a typical classification issue where each input sample must be categorised as malicious or benign. The propensity of classifiers to reverse their judgement when dangerous input is combined with benign content or even random noise is a ubiquitous and mostly unaddressed issue in the field of deep learning-based malware categorization. The attack type is typically presented in the case of appends, though content may also be prepended, injected in the middle, or both. A classifier has a similar issue with each of the three alternatives. We therefore treat them equally. In the actual world, this attack type frequently takes the form of harmless library injections. In that situation, a sizable innocuous file is injected with malicious code.
* Dos: A denial-of-service (DoS) attempt is a kind of cyberattack in which a hostile actor attempts to restrict access to a computer or other device by interfering with regular functioning. DoS attacks sometimes comprise flooding or overloading a targeted computer with requests until regular traffic cannot be handled in order to refuse service to new users. DoS attacks are characterised as a single computer-based assault.
* Port Scan To locate open doors or vulnerable areas in a network, hackers frequently utilise port scans.To find open ports and determine whether they are accepting or rejecting data, cybercriminals may employ a port scan attack. Additionally, it might show if a company routinely employs firewalls and other safety precautions. A port's response to a message sent to it may be used by hackers to determine if it is in use and whether it has any vulnerabilities. Another option open to corporations is the port scanning approach, which is sending packets to certain ports and analysing the answers to search for any possible vulnerabilities. Utilising tools like IP scanners and network mappers (Nmap), they could then.
* Brute Force: A brute force attack relies on trial and error to guess login credentials, encryption keys, and hidden websites. Hackers explore all possible methods in an attempt to get a reliable estimate.They attempted to "force" their way into your secret account.these assaults are carried out "brute force," which refers to the use of excessive force.(s). Hackers still find this attack method to be helpful and popular even though it is an older one. The time needed might vary from a few seconds to several years since the password's length and complexity may have an impact on how long it takes to break.
* Web attacks: aim to exploit vulnerabilities in websites to gain unauthorised access, grab private data, upload malicious content, or change the website's content.
* Bot: An automated web request attack, often known as a bot attack, involves trying to trick, cheat, or otherwise interfere with end users, applications, websites, or APIs. Bot assaults began as straightforward spamming operations before developing into sophisticated, global criminal organisations with independent economies and infrastructures.
* Infiltration: In the course of an attack, privileges often increase. This shows that the attacker has already undertaken reconnaissance and successfully penetrated a system, granting them access. The attacker will have by that point found all the crucial devices and systems after moving laterally across the infected system. Taking complete control of the system is the attacker's aim at this point. The attacker may already have access to a low-level account and is now looking for an account with higher permissions in order to further examine the system or be ready to deliver the fatal blow.

1. PROPOSED SYSTEM

In this project, we suggest developing an intrusion detection system that can identify the attacks listed in the preceding section. By utilising a variety of machine learning methods, primarily classification algorithms, we have suggested an intrusion detection system. To compare and select the most effective model for detecting intrusions, we compute the accuracy, precision, recall, and f1 score of these models. The assaults that we have found are as follow:

* BENIGN
* DoS
* PortScan
* BruteForce
* WebAttack
* Bot
* Infiltration

We are using 3 hybrid models of 2 already existing algorithms in addition to 10 unique machine learning techniques to construct the aforementioned system. With the categorization models, we largely concentrated on developing this system. The following section will cover the classifications. We obtain the dataset for this system by downloading it from Kaggle. Recently, intrusion detection systems (IDS) have been proposed. The Canadian Institute of Cybersecurity provided a cutting-edge dataset called CICIDS2017, which is made up of the newest threats and features, to assess the performance of the IDS. As it includes dangers that the prior datasets did not address, the dataset attracts the interest of many scholars. The dataset has a few significant flaws, it was discovered while conducting an experimental study on CICIDS2017. These problems are sufficient for any conventional IDS's detection engine to be biassed. The metrics we employ for evaluation in this system are

1. Accuracy: For this function to determine subset correctness in multi-label classification, the set of labels predicted for a sample must exactly match the corresponding set of labels in y true.
2. Precision: The accuracy of a material is the agreement between two or more measurements. If you weigh a certain substance five times and get 3.2 kg each time, your measurement is extremely precise but not necessarily accurate. Precision and accuracy may coexist.
3. Recall: The recall is calculated as the proportion of Positive samples to all Positive samples that were correctly classified as Positive. Recall measures a model's ability to locate positive samples. When the recall is higher, more positive samples are identified.
4. f1\_score: The F1 score may be conceptualised as a harmonic mean of accuracy and recall, where 1 is the highest value and 0 is the lowest. The difference in the F1 score between precision and recall is equal. A score of F1 is produced using the following equation:

F1 = 2 \* (precision \* recall) / (precision + recall)

1. METHODOLOGY

We employ classification models and hybrid classification models to create the intrusion detection system. The methods employed fall under the category of machine learning; specifically, the classification algorithms that we use to create this system are classified as supervised machine learning algorithms.

1. *Machine Learning*

It is possible for computer programmes to generate predictions without specific instructions thanks to the artificial intelligence discipline of machine learning. To get insights and determine output values, they instead depend on data analysis. Spam filtering, fraud detection, virus identification, and predictive maintenance are just a few of the many uses for this technology. Making recommendation systems is one of the most well-liked uses of machine learning. For their daily operations and to obtain insights into consumer behaviour and organisational trends, many of the biggest corporations in the world, like Facebook, Google, and Uber, depend on machine learning. Machine learning is becoming a crucial tool for businesses aiming to acquire a competitive edge as a consequence.

* 1. *Random Forest Classification*

As a supervised learning technique, random forest has labels for our inputs and outputs as well as mappings between them. Both classification and regression tasks can be accomplished with it. As the name implies, a random forest is made up of numerous decision trees, each of which produces a forecast. In the event of regression, the random forest will average the outcomes of each decision tree rather than choosing the vote with the greatest number of participants.

* 1. *Gradient Boosting Classification*

Gradient boosting classifiers are a type of machine learning techniques that combine numerous weak learning models to produce a powerful predictive model. For performing gradient boosting, decision trees are frequently employed. Gradient boosting models are gaining popularity because they are good at categorising large datasets, and they have recently been successful in numerous Kaggle data science challenges.

* 1. *CatBoost Classification*

Yandex's CatBoost machine learning algorithm was just recently made available for download. It is simple to integrate with deep learning frameworks like Apple's Core ML and Google's TensorFlow. It can operate with different data formats to assist in resolving a variety of issues that organisations are currently facing. On top of that, it offers the best accuracy in its class.

* 1. *XGBoost Classification*

An open-source software library called XGBoost uses the Gradient Boosting framework to construct machine learning algorithms for distributed gradient boosting that have been tuned. A distributed, scalable gradient-boosted decision tree (GBDT) machine learning framework, XGBoost, stands for Extreme Gradient Boosting. In addition to offering parallel tree boosting, it is the top machine learning package for regression, classification, and ranking issues.

* 1. *AdaBoost Classification*

An AdaBoost classifier is a meta-estimator that starts by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset with the weights of instances that were incorrectly classified being adjusted so that subsequent classifiers concentrate more on challenging cases.

* 1. *BernoulliRBM*

A generative stochastic artificial neural network that can learn a probability distribution across its set of inputs is called a limited Boltzmann machine (RBM). Paul Smolensky created RBMs in 1986 under the name Harmonium, and they gained popularity after Geoffrey Hinton and colleagues developed fast learning algorithms for them in the mid-2000s.

* 1. *Multi Layer Perceptron (MLP)*

Multi-layer perceptrons are a supplement to feed-forward neural networks (MLP). It has an input layer, an output layer, and a hidden layer, among other layer types. Sent to the input layer is the signal that needs to be processed. The required tasks, like categorization and prediction, are handled by the output layer. The input and output layers are separated by an arbitrary number of hidden layers that make up the MLP's true computational engine. A feed-forward network-like movement of data occurs in an MLP from the input to the output layer.

* 1. *Linear Discriminant Analysis (LDA)*

This classifier is appealing because it features closed-form solutions that are simple to compute, naturally multiclass, have a good track record in real-world applications, and don't require any hyperparameter tuning. By transforming the input data into a linear subspace made up of the directions that maximise the separation between classes, linear discriminant analysis can be used to accomplish supervised dimensionality reduction.

* 1. *Quadratic Discriminant Analysis (QDA)*

The linear discriminant analysis (LDA), which similarly presumes a normally distributed distribution for the data from each class, is closely related to the quadratic discriminant analysis (QDA). However, unlike LDA, there is no presumption made in QDA that the covariance of each class is the same.

* 1. *Support Vector Classification (SVC)*

Based on libsvm, the implementation was created. Beyond tens of thousands of samples, it may become impractical for the fit time to scale at least quadratically with the sample count. Instead of utilising LinearSVC or SGDClassifier, perhaps after a Nystroem transformer, for large datasets, think about employing one of these. According to a one-vs-one system, the multiclass support is handled.

* 1. *Hybrid of Random Forest and XGBoost Model*

This model is a combination of Random Forest Classification and XGBoost Classification. The main aim of this developing this hybrid model was to detect if the accuracy rate of the combined model is high than the Random models. And the hybrid model is providing the accuracy rate a bit more than the random forest and boost separately.

* 1. *Hybrid of Random Forest and CatBoost Model*

This model is a combination of Random Forest Classification and CatBoost Classification. For the accuracy rate that this model is providing does not show any increase in the accuracy rate. In fact the accuracy rate of this hybrid model is same as of the CatBoost Classification model.

* 1. *Hybrid of Random Forest and Adaboost Model*

This model is a combination of Random Forest Classification and AdaBoost Classification. This model provides the highest accuracy rate of both Random Forest Classification and AdaBoost as well as all other models used but slightly higher than random forest. In this model we selected two models of which one had highest accuracy rate and the other had the lowest accuracy rate.

1. RESULT AND DISCUSSION

* The Accuracy of the models is provided below in a table format

|  |  |
| --- | --- |
| Algorithm Name | Accuracy |
| Random Forest Classification | 0.995 |
| GradientBoost Classification | 0.455 |
| Multi Layer Perceptron | 0.881 |
| AdaBoost Classification | 0.487 |
| XGBoost Classification | 0.993 |
| CatBoost Classification | 0.982 |
| Support Vector Classification | 0.823 |
| Linear Discriminant Analysis | 0.84 |
| Quadratic Discriminant Analysis | 0.913 |
| Bernoulli RBM | 0.91 |

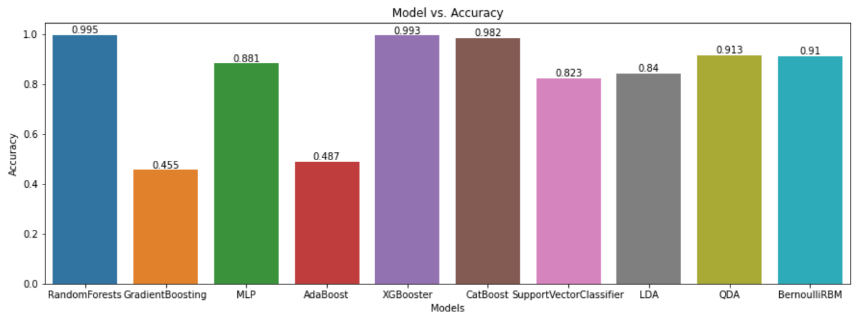


Fig.1 Accuracy rate

The table presents the accuracy of ten different algorithms: Random forest, SVM, GradientBoost Classification, Multi Layer Perceptron, AdaBoost Classification, XGBoost Classification, CatBoost Classification, Linear Discriminant Analysis, Quadratic Discriminant Analysis, and Bernoulli RBM. The accuracy results for each algorithm are also provided. The Random forest algorithm achieved the highest accuracy of 99.5%. The GradientBoost Classification algorithm only achieved 45% accuracy, likely due to its suitability for small-sized datasets. The Multi Layer Perceptron algorithm achieved 88% accuracy, while the AdaBoost Classification algorithm achieved 48% accuracy. The XGBoost Classification algorithm achieved 93% accuracy, while the CatBoost Classification algorithm achieved 98% accuracy. The Support Vector Classification algorithm achieved 82% accuracy, while the Linear Discriminant Analysis algorithm achieved 84% accuracy. The Quadratic Discriminant Analysis algorithm achieved 91% accuracy, and the Bernoulli RBM algorithm achieved 91% accuracy. The Random Forest method, on general, achieved the highest level of accuracy.

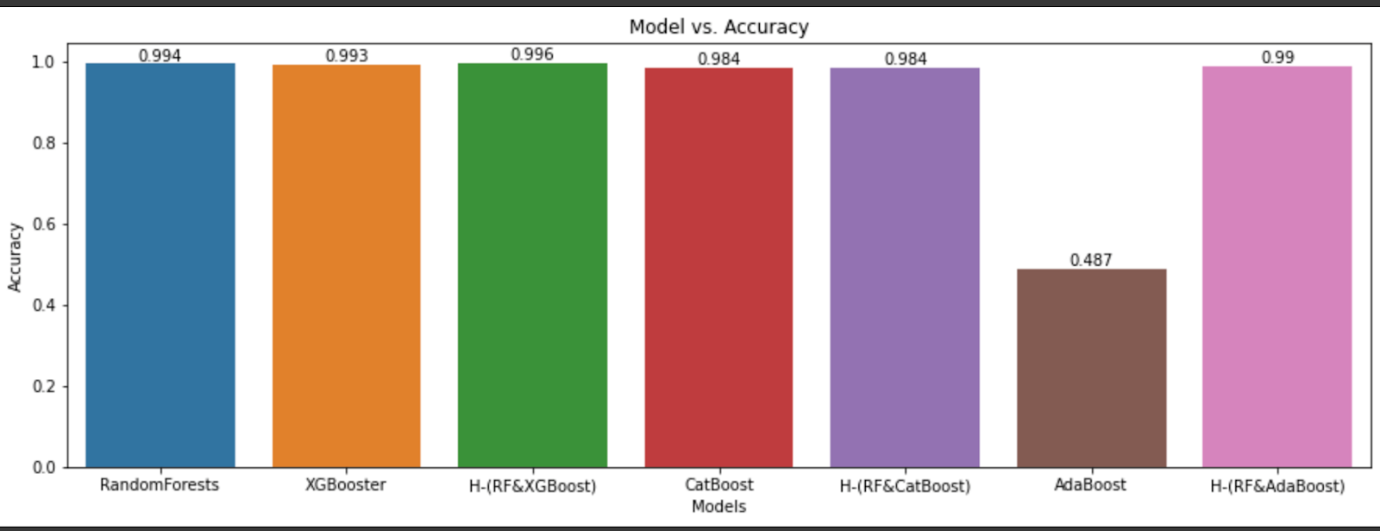


Fig.2. Accuracy rate of Hybrid models

The diagram displays the accuracy of three hybrid models: Hybrid of Random Forest and XGBoost, Hybrid of Random Forest and CatBoost, and Hybrid of Random Forest and AdaBoost. The accuracy results for each hybrid model are also provided. The Hybrid of Random Forest and XGBoost achieved the accuracy of 99.6%. The Hybrid of Random Forest and CatBoost achieved 98% accuracy, while the Hybrid of Random Forest and AdaBoost achieved 99% accuracy. The Hybrid of Random Forest and AdaBoost, out of all the hybrid models, has the highest accuracy.

* The precision of the models is provided below in a table format

|  |  |
| --- | --- |
| Algorithm Name | Precision |
| Random Forest Classification | 0.995 |
| GradientBoost Classification | 0.775 |
| Multi Layer Perceptron | 0.855 |
| AdaBoost Classification | 0.617 |
| XGBoost Classification | 0.993 |
| CatBoost Classification | 0.983 |
| Support Vector Classification | 0.763 |
| Linear Discriminant Analysis | 0.855 |
| Quadratic Discriminant Analysis | 0.932 |
| Bernoulli RBM | 0.907 |

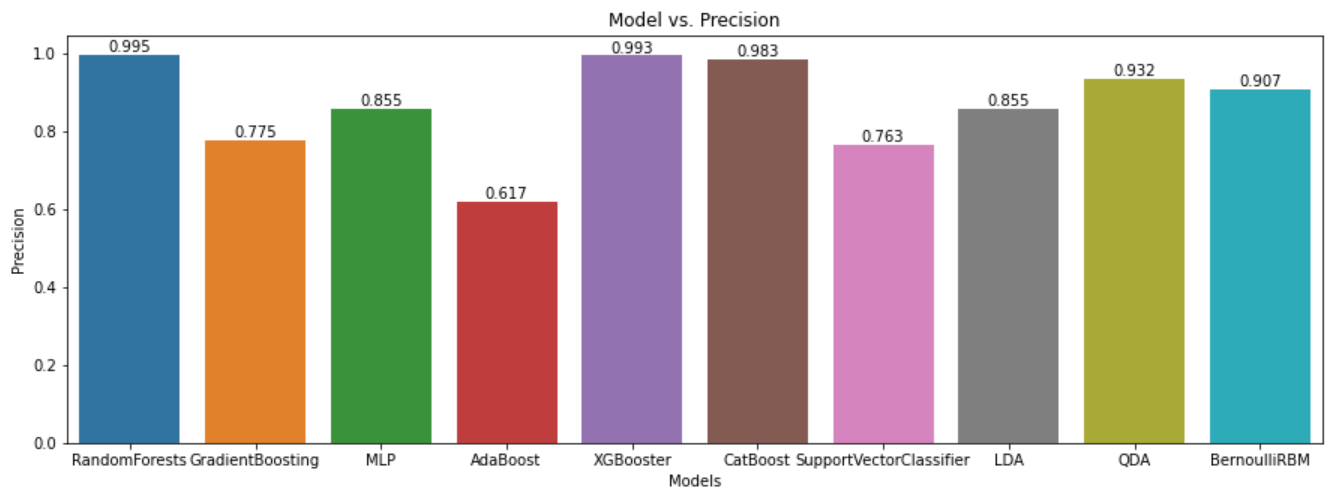


Fig.3. Precision

The table presents the precision of ten different algorithms: Random forest, SVM, GradientBoost Classification, Multi Layer Perceptron, AdaBoost Classification, XGBoost Classification, CatBoost Classification, Linear Discriminant Analysis, Quadratic Discriminant Analysis, and Bernoulli RBM. The precision results for each algorithm are also provided. The Random forest algorithm achieved the highest precision of 99.5%. The GradientBoost Classification algorithm only achieved 77% precision, likely due to its suitability for small-sized datasets. The Multi Layer Perceptron algorithm achieved 85% precision, while the AdaBoost Classification algorithm 61% precision. The XGBoost Classification algorithm achieved 83% precision, while the CatBoost Classification algorithm achieved 98% precision. The Support Vector Classification algorithm achieved 76% precision, while the Linear Discriminant Analysis algorithm achieved 85% precision. The Quadratic Discriminant Analysis algorithm achieved 93% precision, and the Bernoulli RBM algorithm achieved 90% precision. Overall, the Random forest algorithm had the best precision among all the algorithms.

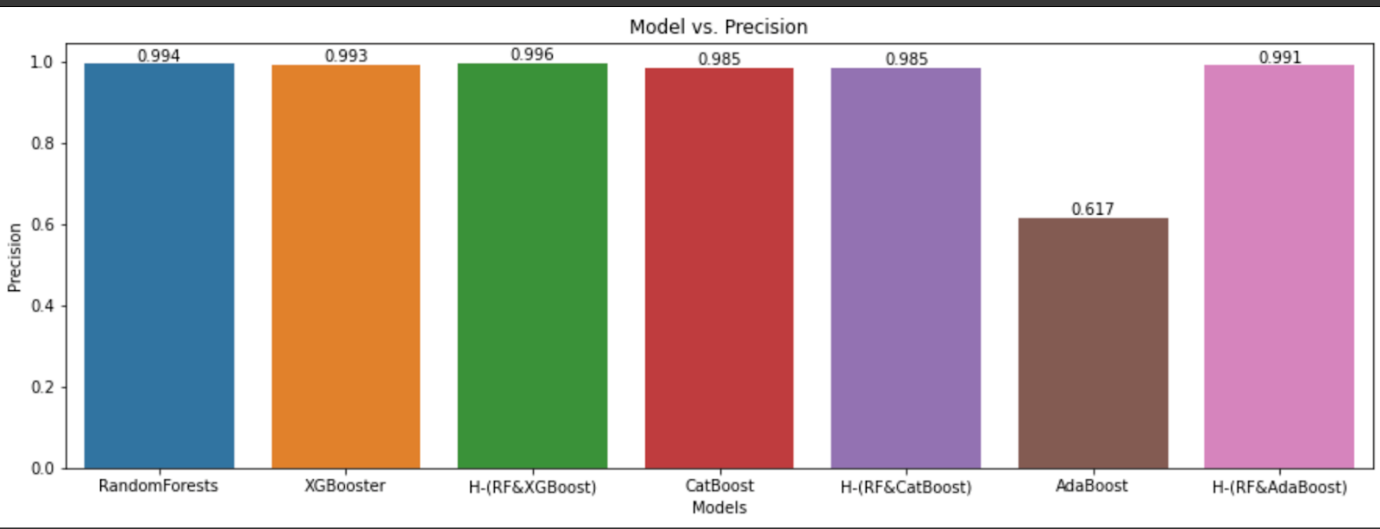


Fig.4. Precision of hybrid models

The diagram displays the precision of three hybrid models: Hybrid of Random Forest and XGBoost, Hybrid of Random Forest and CatBoost, and Hybrid of Random Forest and AdaBoost. The precision results for each hybrid model are also provided. The Hybrid of Random Forest and XGBoost achieved the precision of 99.6%. The Hybrid of Random Forest and CatBoost achieved 98% precision, while the Hybrid of Random Forest and AdaBoost achieved 99% precision. Overall, the Hybrid of Random Forest and AdaBoost had the best precision among all the hybrid models.

* The Recall of the models is provided below in a table format

|  |  |
| --- | --- |
| Algorithm Name | Recall |
| Random Forest Classification | 0.995 |
| GradientBoost Classification | 0.445 |
| Multi Layer Perceptron | 0.881 |
| AdaBoost Classification | 0.487 |
| XGBoost Classification | 0.993 |
| CatBoost Classification | 0.982 |
| Support Vector Classification | 0.823 |
| Linear Discriminant Analysis | 0.84 |
| Quadratic Discriminant Analysis | 0.913 |
| Bernoulli RBM | 0.91 |

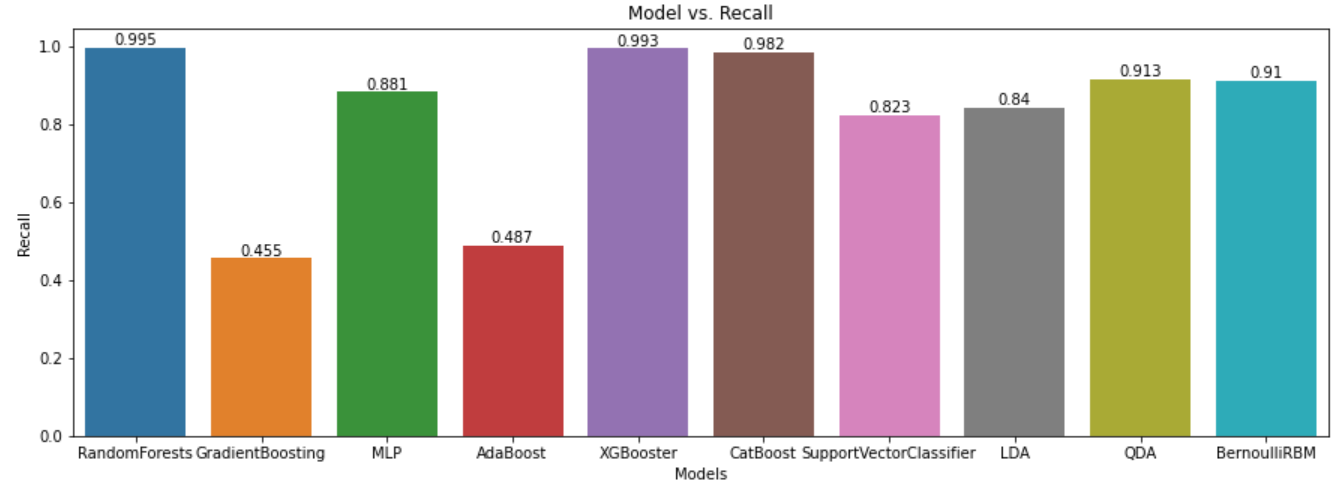


Fig.5. Recall

The table presents the Recall of ten different algorithms: Random forest, SVM, GradientBoost Classification, Multi Layer Perceptron, AdaBoost Classification, XGBoost Classification, CatBoost Classification, Linear Discriminant Analysis, Quadratic Discriminant Analysis, and Bernoulli RBM. The Recall results for each algorithm are also provided. The Random forest algorithm achieved the highest Recall of 99.5%. The GradientBoost Classification algorithm only achieved 44% Recall, likely due to its suitability for small-sized datasets. The Multi Layer Perceptron algorithm achieved 88% Recall, while the AdaBoost Classification algorithm 48% Recall. The XGBoost Classification algorithm achieved 99% Recall, while the CatBoost Classification algorithm achieved 98% Recall. The Support Vector Classification algorithm achieved 82% Recall, while the Linear Discriminant Analysis algorithm achieved 84% Recall. The Quadratic Discriminant Analysis algorithm achieved 91% Recall, and the Bernoulli RBM algorithm achieved 91% Recall. Overall, the Random forest algorithm had the best Recall among all the algorithms.

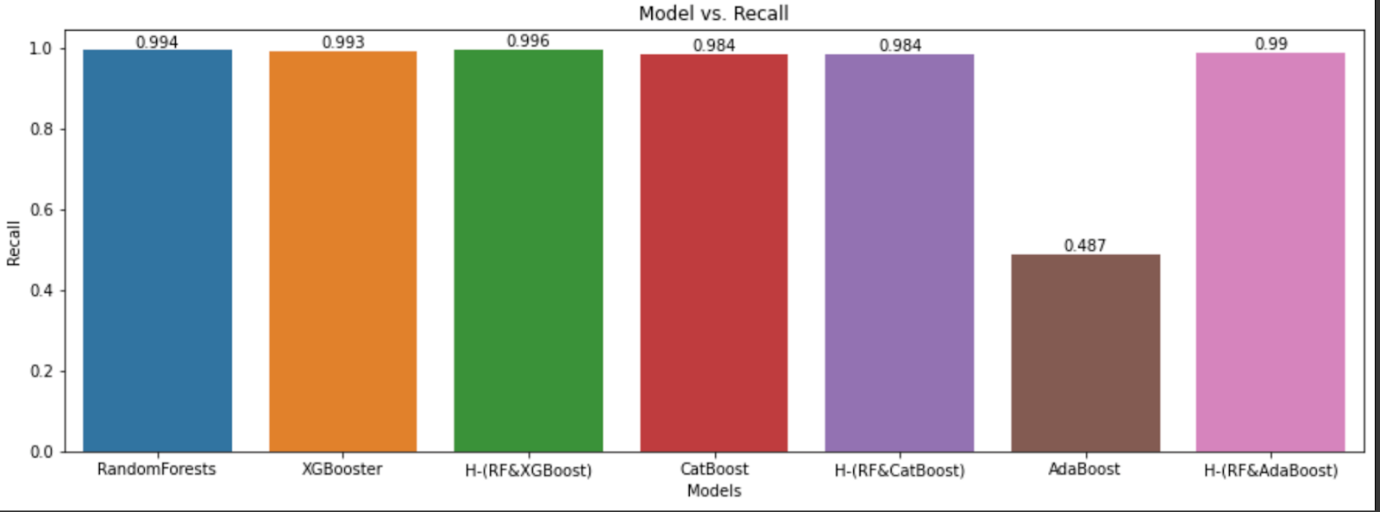


Fig.6. Recall of Hybrid models

The diagram displays the Recall of three hybrid models: Hybrid of Random Forest and XGBoost, Hybrid of Random Forest and CatBoost, and Hybrid of Random Forest and AdaBoost. The Recall results for each hybrid model are also provided. The Hybrid of Random Forest and XGBoost achieved the Recall of 99.9%. The Hybrid of Random Forest and CatBoost achieved 98% Recall, while the Hybrid of Random Forest and AdaBoost achieved 99% Recall. Overall, the Hybrid of Random Forest and AdaBoost had the best Recall among all the hybrid models.

* The F1\_Score of the models is provided below in a table format

|  |  |
| --- | --- |
| Algorithm Name | F1\_Score |
| Random Forest Classification | 0.995 |
| GradientBoost Classification | 0.503 |
| Multi Layer Perceptron | 0.864 |
| AdaBoost Classification | 0.537 |
| XGBoost Classification | 0.993 |
| CatBoost Classification | 0.982 |
| Support Vector Classification | 0.789 |
| Linear Discriminant Analysis | 0.84 |
| Quadratic Discriminant Analysis | 0.916 |
| Bernoulli RBM | 0.906 |

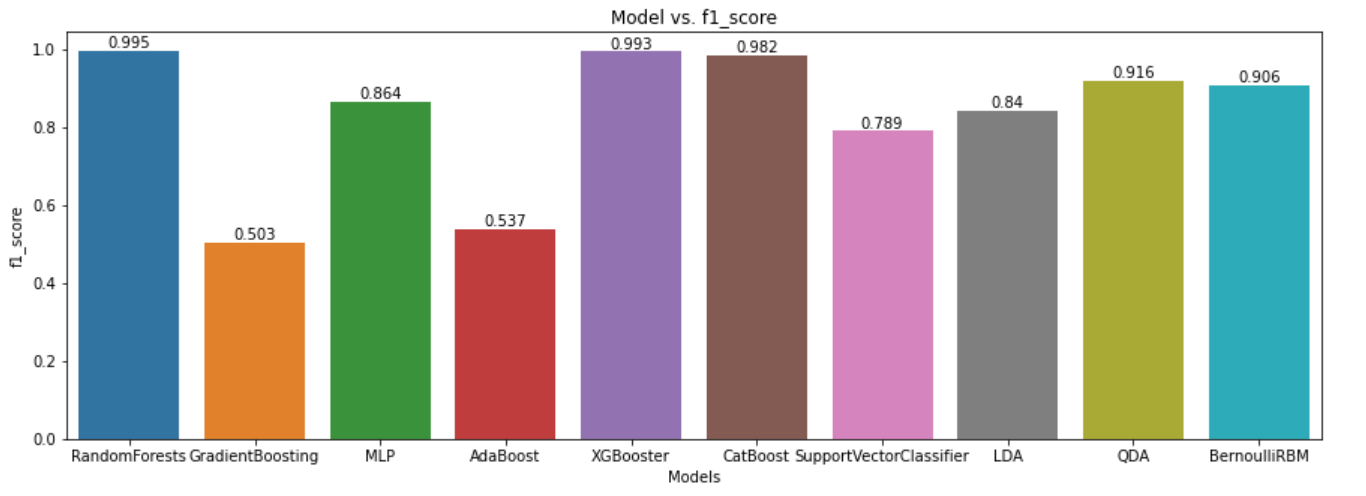


Fig.7. F1\_Score

The table presents the F1-Score of ten different algorithms: Random forest, SVM, GradientBoost Classification, Multi Layer Perceptron, AdaBoost Classification, XGBoost Classification, CatBoost Classification, Linear Discriminant Analysis, Quadratic Discriminant Analysis, and Bernoulli RBM. The F1-Score results for each algorithm are also provided. The Random forest algorithm achieved the highest F1-Score of 99.5%. The GradientBoost Classification algorithm only achieved 50% F1-Score, likely due to its suitability for small-sized datasets. The Multi Layer Perceptron algorithm achieved 86% F1-Score, while the AdaBoost Classification algorithm 53% F1-Score. The XGBoost Classification algorithm achieved 99.3% F1-Score, while the CatBoost Classification algorithm achieved 98% F1-Score. The Support Vector Classification algorithm achieved 78% F1-Score, while the Linear Discriminant Analysis algorithm achieved 84% F1-Score. The Quadratic Discriminant Analysis algorithm achieved 91% F1-Score, and the Bernoulli RBM algorithm achieved 90% F1-Score. Overall, the Random forest algorithm had the best F1-Score among all the algorithms.

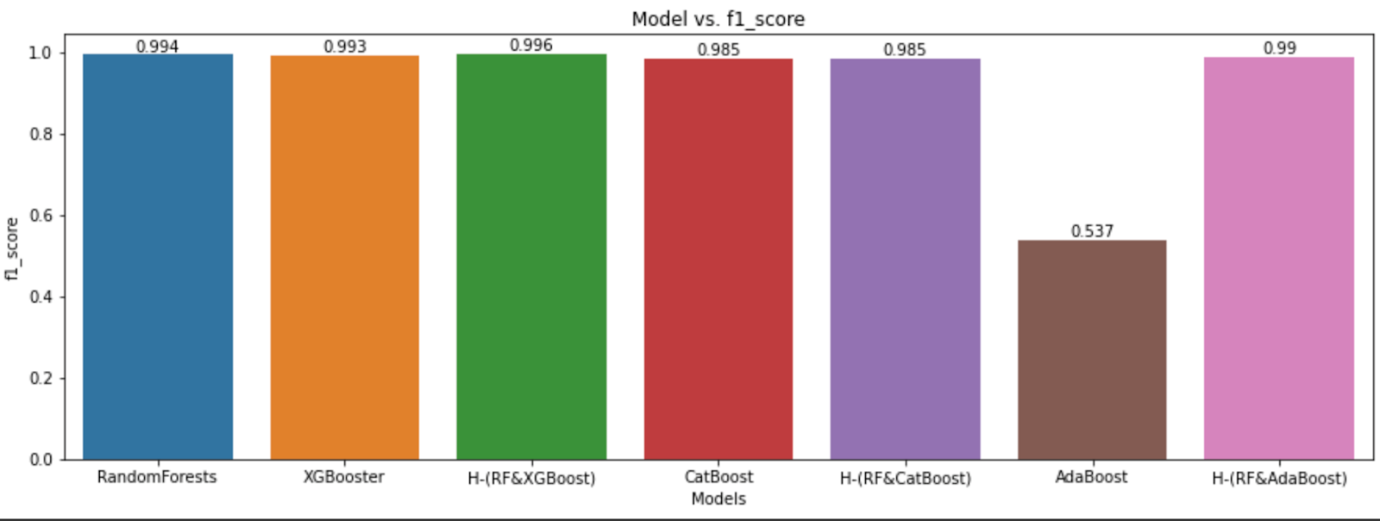


Fig.8. F1\_Score of Hybrid models

The diagram displays the F1-Score of three hybrid models: Hybrid of Random Forest and XGBoost, Hybrid of Random Forest and CatBoost, and Hybrid of Random Forest and AdaBoost. The F1-Score results for each hybrid model are also provided. The Hybrid of Random Forest and XGBoost achieved the F1-Score of 99.6%. The Hybrid of Random Forest and CatBoost achieved 98% F1-Score, while the Hybrid of Random Forest and AdaBoost achieved 99% F1-Score. Overall, the Hybrid of Random Forest and Adaboost had the best F1-Score among all the hybrid models.

Out of all the algorithms and hybrid models mentioned, the Hybrid of Random Forest and AdaBoost achieved the best accuracy of 99.9%, which indicates that it performed better in terms of predicting the outcomes of the CICIDS2017 dataset. While the Random Forest algorithm achieved the highest precision of 99.5%, precision is a measure of the model's ability to identify true positive results. Accuracy, on the other hand, is a measure of the model's ability to predict both true positive and negative results. Hence, accuracy is generally considered a better measure of overall performance than precision. Therefore, based on the accuracy results mentioned, it can be concluded that the Hybrid of Random Forest and AdaBoost performed the best out of all the algorithms and hybrid models

1. CONCLUSION

In the comparison and understanding of various methodologies, it’s visible that the methodology involving machine learning stands out and it is better to predict and analyze the data. The various approaches of machine learning show more of a good performance when we compare the performances of the methodologies. In many cases considered the Random Forest, XGBoost CatBoost along the Hybrid of Random Forest and XGBoost and the Hybrid model of Random Forest and AdaBoost , stands out as a best methodology that is to be used for intrusion detection. The Hybrid model of random forest and adaboost has showen unexpected higher accuracy than them separately, adaBoost had the lowest accuracy rate.

1. REFERENCE

[1] Mohammed Hasan Ali 1 , Bahaa abbas dawood al Mohammed 2 , ”A New Intrusion Detection System Based on Fast Learning Network and Particle Swarm Optimization”

[2] Kaiyuan jiang , Wanya wang , aili wang , and haibin wu ”Network Intrusion Detection Combined Hybrid Sampling With Deep Hierarchical Network”

[3] Taha Selim Ustun 1 , (member, IEEE), s. m. Suhail Hussain 2 , (member, Ieee), levent yavuz3 , (member, IEEE), and Ahmet onen3 “Artificial Intelligence Based Intrusion Detection System for IEC 61850 Sampled Values Under Symmetric and Asymmetric Faults”

[4] Xianwei Gao , Chun Shan , Changzhen hu, Zequn Niu , and Zhen LIU “An Adaptive Ensemble Machine Learning Model for Intrusion Detection”

[5] V. Kanimozhi and T. Prem Jacob “Artificial Intelligencebased Network Intrusion Detection with HyperParameter Optimization Tuning on the Realistic Cyber Dataset CSE- CICIDS2018 using Cloud Computing”

[6] Zaidon Kamil Maseer 1 , Robiah yusof1 , Nazrulazhar bahaman1 , Salama a. Mostafa 2 , and Cik feresa mohd foozy2 “Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset”

[7] Ahmad, M. Basheri, M. J. Iqbal, and A. Rahim, ‘‘Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection,’’ IEEE Access

[8] Khan, A. Mehmood, S. Khan, M. A. Khan, Z. Iqbal, and W. K. Mashwani, ‘‘A survey on intrusion detection and prevention in wireless ad-hoc networks,’’ J. Syst. Archit., vol. 105, May 2020,

[9] Soheily-Khah, P. Marteau, N. Béchet, ‘‘Intrusion detection in network systems through hybrid supervised and unsupervised machine learning process: A case study on the ISCXdataset”