Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7765504

MACHINE LEARNING BASED CYBER ATTACK DETECTION USING NSL-KDD DATASET

Tata Sowjanya kumari
PG Research Scholar, Department of Computer Science Engineering, Raghu Engineering College, Dakamarri, Visakhapatnam – 531162, Andhra Pradesh, India.
Krithikganesh13@gmail.com

Anuradha Tutika
Assistance Professor, Department of Computer Science Engineering, Raghu Engineering College, Dakamarri, Visakhapatnam – 531162, Andhra Pradesh, India.
anuradha.tutika@raghuenggcollege.in

Jagannadhasastry. V
Assistant professor, Department of physics, M.R College (autonomous), Vizianagaram – 535002, Andhra Pradesh, India. sastry.physics@gmail.com

Abstract— Cybercrime is spreading rapidly as hackers find new ways to take advantage of flaws in the global information infrastructure. Ethical hackers place a greater emphasis on identifying security flaws and proposing solutions. In the area of cyber security, there is a significant need for the creation of efficient methods. The complex and unpredictable nature of cyber-attacks against communications networks indicates that most techniques employed in today's Intrusion Detection System (IDS) are inadequate. Machine learning (ML) in cyber security has lately gained prominence due to its efficiency in cyber security challenges. Intrusion, malware, spam, and phishing categorization and identification are just some of the primary challenges that ML approaches were applied to in the field of cyber security. While ML can't be used to automate a whole cyber security system, it does aid in the identification of cyber security threats more effectively than other software-oriented techniques, which in turn relieves pressure on security analysts. Data from NSL-KDD is collected and analysed in this paper. Correlation-based Feature Selection (CFS) and Information Gain (IG) are employed to pick out the most relevant aspects of the processed data. Random Forest (RF), K-Nearest Neighbour (KNN), and Naive Bayes (NB) are the three ML models created. In the end, the measurements are used to determine which method of ML model and feature selection is the most effective. Good accuracy rates decreased false rates, and manageable computational and communication expenses can all result from combining ML with an effective feature selection strategy.

Keywords— Cyber-attack, One Hot Encoding, Machine Learning, Feature Selection, Internet, Accuracy.

Introduction

Now more than ever, both governments and corporations use sophisticated forms of cyber-warfare to damage, disrupt, or limit access to data stored in online systems [1]. While designing network protocols, it is important to ensure that they are robust against intrusions by powerful attackers who can influence even a small number of network participants. Subjects under control might engage in both passive (such as eavesdropping and avoiding confrontation) and active (e.g., jamming, message dropping, corruption, and forging) forms of aggression. The term "infiltration detection" is used to describe the process of keeping a close eye on a computer system or network, analysing the collected data in search of any indications of an incursion, and shutting it down if necessary. In many cases, this is accomplished by the use of automated data collection from a variety of nodes in the target system or network, followed by a thorough examination for security flaws. Firewalls, access control systems, and encryption are all examples of classic IDS/IPS technology, but they can't stop every possible threat, like a denial of service. Most systems built using these approaches also have high rates of false positive and false negative detection and are unable to continuously adjust to changing hostile behaviour. Over the past decade, many ML methods have been applied to the issue of intrusion detection to raise detection rates and adaptability [2]. These methods are used frequently to keep attack databases up-to-date and complete. How to ensure cyber safety and protection against growing cyber threats has become an important concern in recent years. As the Internet continues to expand and become more widely used, many users of networks reap benefits from a variety of resources. Furthermore, as more people come to rely on the internet, the relevance of network security grows in parallel with it. Network security [3] is the practice of protecting a network's computers, networks, programs, data, and other assets from being hacked or otherwise modified by unauthorized users. Network assaults can cause significant losses in the banking sector, online retail, and the military as more and more systems connect to the internet. Large-scale networks are more vulnerable to cyber-attacks, which can result in both financial and physical losses. Unfortunately, the security measures already in place, such as firewalls (both hardware and software), user authentication, and data encryption, are not enough to keep a computer network safe from cybercriminals. Due to the rapid and rigorous development of infiltration technologies, the traditional security architecture is no longer an adequate safeguard. A firewall's sole function is to control the flow of data between networks; hence it can only limit their accessibility. The warning does not trigger, however, if the attack comes from within the system itself. Hence, it is obvious that developing accurate defensive measures, such as an intrusion detection system (IDS) based on ML, is necessary for the security of the system [4]. An IDS is a tool used to keep an eye out for suspicious behavior on a network or for policy violations. Denial-of-service (Dos) attacks and other threats to network security can be uncovered with the help of an IDS, which looks for anomalies in the typical functioning of a network or system. Unauthorized access, modification, or destruction of a system can be detected, decided upon, and managed with the help of an IDS.

To keep networks, secure, it is essential to provide effective methods for monitoring and countering attacks. Furthermore, various assaults call for various responses. Therefore, the most pressing problem in network security is identifying and blocking novel threats. Researchers have started using ML methods to build IDS that can adapt to the various forms of cyber threats. Artificial intelligence methods could be able to differentiate between "normal"

and "abnormal" data with a high degree of accuracy. In the paper [5], the authors suggest three different feature selection algorithms for IDS, which are then proceeded by ML and Deep Learning (DL). They gathered two datasets and used the ANOVA F-value, mutual information, and impurity-based feature selection to choose the important features. Next, they used Neural Networks, K-NN, DT, and Logistic Regression on two datasets, with resulting good accuracy. The study [6] describes a method for detecting DDoS assaults using ML. Because independent component analysis is multidimensional, it can reduce the number of attributes commonly utilized in detection. Data could be converted into higher dimensions through independent component analysis, allowing for the discovery of feature subsets. During training and testing, the false rate and detection accuracy of the updated SVM for categorization can be tracked. When given fewer data points to deal with, a modified SVM outperforms existing algorithms for pattern categorization. To enhance the accuracy of the classifications made, they use a modified SVM that has been optimized with the help of the BAT and the Cuckoo Search technique. When compared to the other classifiers, the research shows substantial improvement. The research [7] presents an artificial intelligence technique for monitoring IoT networks for intrusion. The suggested framework for monitoring the Internet of Things (IoT) and detecting any suspicious or destructive behaviour is an IDS that leverages an ML algorithm. In this study, they used supervised ML to improve detection accuracy while decreasing data processing time. The suggested model is evaluated using four classification algorithms. The trials revealed that KNN has the highest accuracy and the quickest training time. Finally, they suggested deploying the designed IDS in a production IoT environment. To strengthen IIoT-enabled networks, a novel approach is proposed in the article [8]. Here, they offer a strategy for reliable and precise cyberattack identification in SCADA networks. The proposed technique integrates the SCADA-based IIoT system with DL-based pyramidal recurrent units (PRU) and DT. They use an ensemble-learning technique to further enhance the security of SCADA-based IIoT networks against attacks. The proposed approach is tested using 15 data collected from SCADA-based networks. The experimental outcomes demonstrate the superiority of the suggested strategy over conventional and ML-based detection strategies. With the suggested method, IIoT-enabled networks are safer and more trustworthy.

The methods outlined in [9] can be used to steal electricity surreptitiously by impersonating regular usage while also hacking into surrounding residences' meters. Because the manipulated data differs so little from the genuine usage records, present tools are practically incapable of detecting such an assault. To address this risk, they differentiate between two types of consumption deviations: individual and social. Then, develop a feature extraction method capable of catching the link between attackers and legal users. The research results in a revolutionary DL-based identification system. The research using real-world datasets shows the suggested attack can avoid detection by existing mainstream detectors while still offering large financial rewards. The proposed countermeasure also outperforms the best available detection methods. The goal of the survey [10] is to provide a comprehensive overview of ML techniques for allowing stealthy and extremely successful attacks. As a result, the purpose of this research is to determine the benefits of ML attack complexity and to investigate various responses that have previously been proposed in the literature. To begin, they list the most significant threats and potential attacks on IoT networks. Then, look at previous cyberattacks

that used ML techniques and make recommendations for future research, focusing on jamming, side channels, false data injection, and adversarial ML. The author [11] proposes applying an ML approach to protect autonomous vehicles against cyberattacks. They are interested in identifying instances of malicious data insertion into a vehicle's data bus. To solve this problem, they would use extreme gradient boosting, a powerful ML method. Details of the research procedure, including data collection, cleaning, purposely introducing fake data, and data classification. The results indicate that the suggested method may detect anomalous action on the vehicle's data bus with good accuracy. The author [12] proposes a unique collaborative learning model based on Transfer Learning (TL). With the novel collaborative learning approach, a target network with little to no labelled data may rapidly and effectively "learn" from a source network with a significant number of labelled data. The contributing network datasets in cutting-edge studies must all share the same properties, limiting IDS' effectiveness, adaptability, and scalability. Despite this, the suggested method could address these concerns of DL models, regardless of the features available in their separate datasets. Extensive research on recently released real-world cybersecurity data shows that the suggested framework can outperform conventional DL networks.

The structure of the research paper is given below:

Section I: Effects of cyber-attack, its detection importance and recent research in identifying cyber-attack in the network using advanced technologies like ML and DL

Section II: The methodology of the suggested framework with Flow chart. The different stages involved in suggested framework

Section III: Outcome Comparison of the feature selection approaches with three different ML models using bar graph.

Section IV: Concludes the research framework by suggesting best features selection and ML model combination for cyber-attack detection.

Methodology

The NSL-KDD dataset is used by this proposed framework. There are essentially three phases to this process. Data pre-processing, including encoding and normalization, is the initial phase. In multi-dimensional data, inappropriate or redundant features may lower the effectiveness of attack identification. Feature selection like Correlation-based and IG is employed as a second phase to overcome this issue by removing irrelevant and noisy features from a high-dimensional dataset. The third phase is to train various ML classifiers using the extracted features to identify the types of attacks with extreme accuracy. Then, the model is tested using performance metrics. Figure 1 depicts the proposed technique representing the overall framework.

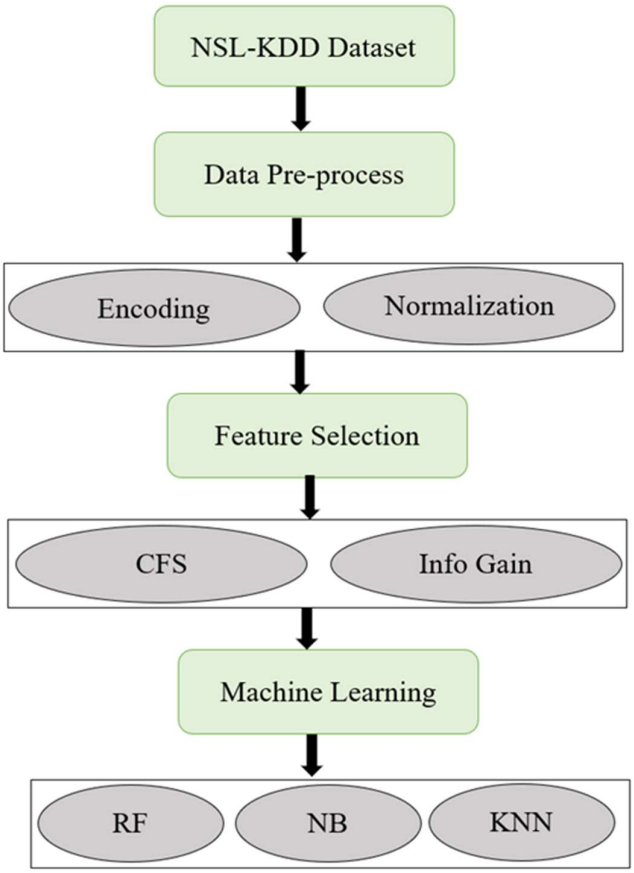


Fig. 1. Proposed Methodology

Data and its processing

The study used the NSL-KDD dataset [13] for cyber-attack detection. There are a total of 148,486 samples: 125,943 for training and 22,543 for testing. There are 42 features in all, 41 of which are truly independent and 1 of which is dependent on another feature. There are a total of 37 attacks: Probe, user-to-root (U2R), DoS, and reverse-to-local (R2L) attacks are the four main types of attacks.

Data must be pre-processed before IDS can function. It includes data normalization, and one-hot encoding.

The Data Normalization Process The fact that different dimensions operate on dramatically different time and size scales is a critical obstacle in applying DL. The NSL-KDD dataset contains 41 dimensions with exceptionally wide-ranging values. We utilize the min-max normalization technique [14] in this work to merge data from several scales into a single one. The original data is converted linearly so that it fits inside the range [0, 1]. The min-max values are transformed using the following equation.

$$\tilde{x}_{fj} = \frac{x_{fj} - \text{lo}\ \ (x_f)}{\text{high}\ (x_f) - \text{low}\ (x_f)} \quad [1]$$

Where,

$\min\ (x_f)$ and $\max(x_f)$ → lowest and highest values of the f[th] numeric feature

x_f, and $\tilde{x}_{fj}$ → Normalized feature [0,1].

One-hot-encoding is the standard practice for handling categorical data because it is efficient and easy to implement. Every qualitative feature can be transformed into a binary vector, where one component is assigned a score of 1 and the others are assigned 0. If the element has a value of 1, then the criterion can have several different values. NSL-KDD employs three distinct sorts of classifications, or "features," namely the protocol type, the service, and the flag (x_2, x_3, x_4). For example, the protocol_ type attribute can take the values tcp, udp, or icmp. With one-hot encoding, TCP is represented by (1,0,0), UDP by (0,1,0), and ICMP by (0,0,1). In the same way, the service and the flag are transformed using one-hot encoding. The three nominal attributes (service, flag, and protocol_ type) are assigned to a total of 84-dimensional binary values.

Feature Selection

The amount of memory needed can be decreased, processing time can be sped up, and classification accuracy can be improved by using feature selection methods [15]. The focus of this study is on comparing the effectiveness of using different feature selection methods with ML classifiers to arrive at a final recommendation. Data mining has employed a wide range of methods for deciding upon feature subsets.

Correlation-based feature selection (CFS): CFS is a very simple and effective feature selection technique, evaluating feature subsets with a correlation-based evaluation mechanism under the presumption that attributes are conditionally independent. For a feature subset to be useful, its features must be highly correlated with the class while still being uncorrelated with one another. When compared to wrapper selection methods [16], CFS is much quicker because it does not necessitate the execution of the learning algorithms.

$$\rho(X,Y) = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{[\sum_i (x_i - \bar{x})^2 (y_i - \bar{y})^2]^{\frac{1}{2}}} \quad [2]$$

The Pearson correlation coefficient assumes that all variables have been standardized and shows that the strength of the relationship between a dependent and an independent variable is determined by the number of variables that comprise.

Information Gain (IG): IG is a metric for measuring the value of a characteristic in terms of the entropy concept. A rise in entropy means that there is more data. The entropy of a system can be thought of as a measure of its degree of uncertainty. Within a particular group, the features with the highest mutual information are chosen for further investigation. Based on the study result, it is obvious that such well-chosen features can result in increased classification performance [17]. Prioritization of features based on information.

Input: Train data→ $T = D(F, C)$, Total features →f_i
Output: Selected features S

1. Relative parameters are initialized
2. For all feature$f_i \in$ F
   Compute information gain $IG(f_i)$;
   Put f_i into S based on $IG(f_i)$, in descending;
3. Remove all features except 1st feature in S
4. Return S Selected features

Machine Learning

Similar to regression, classification is another popular application of supervisory ML. ML-based classifiers are successfully used for cyber threat detection. RF, KNN, and NB Classification are the ML algorithms used for classification.

RF: An RF classifier is composed of many trees that grow in random order. The posterior distribution estimates for the various image kinds are labelled on the leaves of each tree. Each internal node is outfitted with a test that splits the training data area optimally. Data is classified by distributing it to the leaves of each tree and averaging the distributions at each level. Randomness could be introduced into the training phase by subsampling the training sample so that each tree is built with various subsets by randomly picking the node test. A bigger quantity of trees is required for optimal performance. A comparison of forest forecasts with subgroup forest estimates is the most effective method for determining how many trees are required. We know we have enough trees when different areas of the forest operate as well as the overall. Before agreeing on the appropriate m_try value, Breiman [18] suggests cycling through the half and twice the default. When there are many variables but just a tiny fraction are regarded "Important," Increasing m_try can help improve efficiency. A vast number of trees are required to provide meaningful estimations of the relative significance and closeness of variables. Because the procedure is “embarrassingly parallel,” executing many RF on various machines and then aggregating the votes from each will provide the same outcome.

The RF classifier adds an extra level of ambiguity to the bagging operation. RFs, alter the creation of classification trees, in addition to the use of various bootstrap sampling of the data for all trees. RF uses the optimal split over a subset of predictors selected at random on every node, rather than the optimal split among all variables at every node. RF is also easy to employ because it simply requires two parameters (total variables in a random subset at every node and total trees in the forest) and isn’t oversensitive.

NB: NB Classifiers are simple to use and extremely accurate at their intended goal. The foundation of this method is Bayesian Networks, a probabilistic graphical framework for expressing a group of random variables and its conditional dependence [19]. A wide variety of effective reasoning and training methods exist inside Bayesian Networks. The only condition is that the data must include independent features. Although there seem to be a few connections among the features of the dataset, they are weak and likely the consequence of random chance. So, the NB method is employed for categorization under the assumption that features in the input are irrelevant. At first, this technique is applied to the sequence database to produce a probability score for every sequence. Each sequence’s probability is calculated after input data has been given. The probabilities are used to categorize sequences. The NB model uses the Bayes theorem with the strong independence condition to create a straightforward probabilistic classification. The probability of cyber-attack features X with the type C_j is:

$$P\left(\frac{C_j}{X}\right) = \frac{P\left(\frac{X}{C_j}\right).P(C_j)}{P(X)} \quad [3]$$

A sequence's classification in input data is decided by the attack type C_j with the maximum probability score.

KNN: The KNN algorithm is a pattern recognition approach that uses instance-based learning to categorize the attack in the feature space using examples that are most similar to them. It is a fair method to place an item in the group that has the most support from its closest neighbours, with a k-positive integer [20]. Using the labels of its closest neighbours, the KNN method assigns new test features to one of several possible classes. T If want to find the closest neighbour, the KNN method will employ Euclidean distance measurements. To determine the Euclidean distance among two points x and y, the following equation is used.

$$d(x,y) = \sum_{i=1}^{N} \sqrt{x^2 - y^2} \quad [4]$$

KNN performs effectively with a large number of examples and is resistant to noisy training data. However, before using this strategy, the parameters k and d must be set. Calculating the distance among each sample in all training data could take a lengthy time, and this slowdown becomes more noticeable as the number of samples grows. However, it isn't necessary to construct a model, experiment with a variety of parameters, or establish any assumptions. The KNN method of supervised ML is simple, flexible, and straightforward to apply. It's useful for solving searching, and categorization issues. The calculation assumes that there are multiple occurrences of the same object nearby. The related objects prefer to cluster together. The KNN method's efficiency is based on this assumption being correct. Since KNN slows down as the amount of input grows, and not a good option for quickly making predictions. There are speedier approaches that really can generate more accurate classification outcomes. But if there are enough computational capabilities to analyse the data quickly and generate forecasting, KNN could still be beneficial in handling issues whose answers depend on locating related objects.

Results and Discussion

The NSL-KDD data was gathered in this research to design the ML model for identifying cyber-attack. The collected data is not in a proper format. To convert this raw data into a useful format, the processing approaches like normalization and encoding are employed. The processed data holds too many features which leads to reduce the effectiveness of the model. To improve the effectiveness, the feature selection is done. Finally, the ML model is created, trained, and tested using the selected features. The ML model outcome with IG and CFS is evaluated by the below-mentioned metrics.

$$Accuracy = \frac{TP+T}{TP+TN+FP+FN} * 100 \quad [5]$$

$$TPR = \frac{TP}{FN+TP} * 100 \quad [6]$$

$$TNR = \frac{TN}{TN+FP} * 100 \quad [7]$$

$$FPR = \frac{FP}{TN+F} * 100 \quad [8]$$

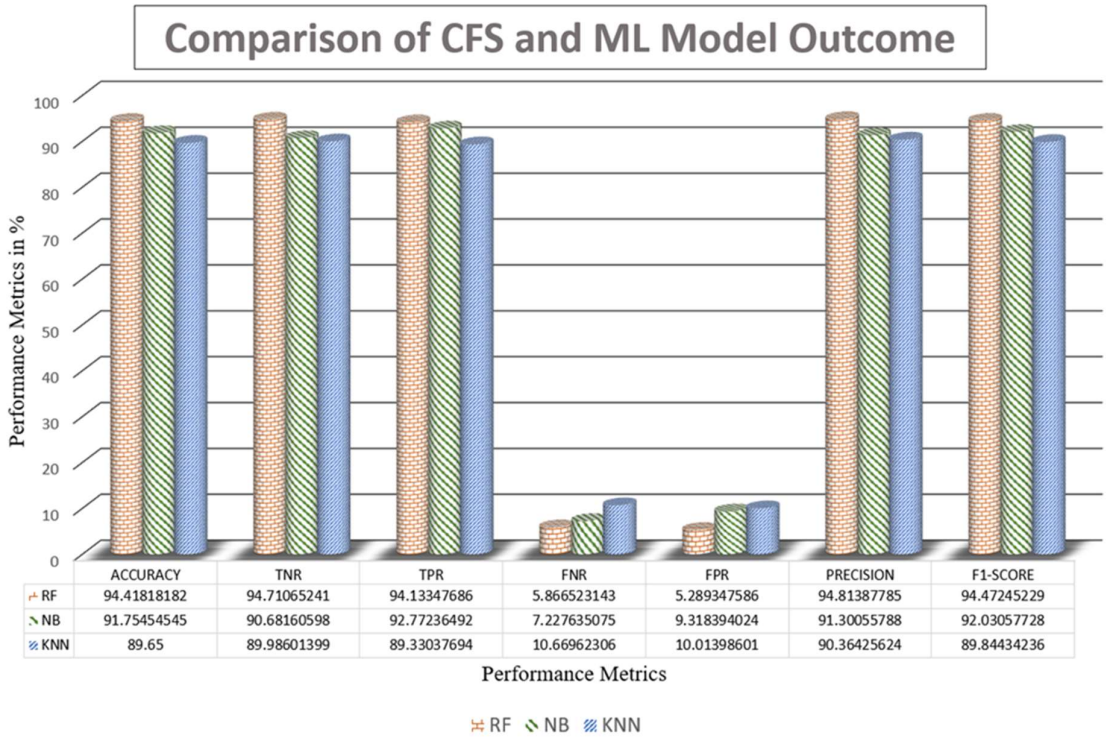
$$FNR = \frac{FN}{FN+T} * 100 \quad [9]$$

$$Precision = \frac{TP}{FP+T} * 100 \quad [10]$$

$$F1 = \frac{TP}{TP\ \frac{1}{2}(FP+FN)} * 100 \quad [11]$$

In the above equations, metrics are calculated using the confusion matrix. The TP is used to represent the count of True Positive, TN for True Negative, FP for False Positive, and FN for False Negative.

The metrics score attained by the ML model using the features of the CFS approach is depicted in figure 2. The positive metrics value will be maximum for RF and minimum for KNN. All positive metrics scores of RF will fall in and around 94%, NB scores in the range of 90-92%, and KNN scores are made of 89-90%. In case of negative metrics, the RF score is minimum and KNN scores will be maximum. The RF and KNN values are in the range of 5% and 10%.



	ACCURACY	TNR	TPR	FNR	FPR	PRECISION	F1-SCORE
RF	94.41818182	94.71065241	94.13347686	5.866523143	5.289347586	94.81387785	94.47245229
NB	91.75454545	90.68160598	92.77236492	7.227635075	9.318394024	91.30055788	92.03057728
KNN	89.65	89.98601399	89.33037694	10.66962306	10.01398601	90.36425624	89.84434236

Fig. 2. Comparison bar graph of ML model using CFS outcome

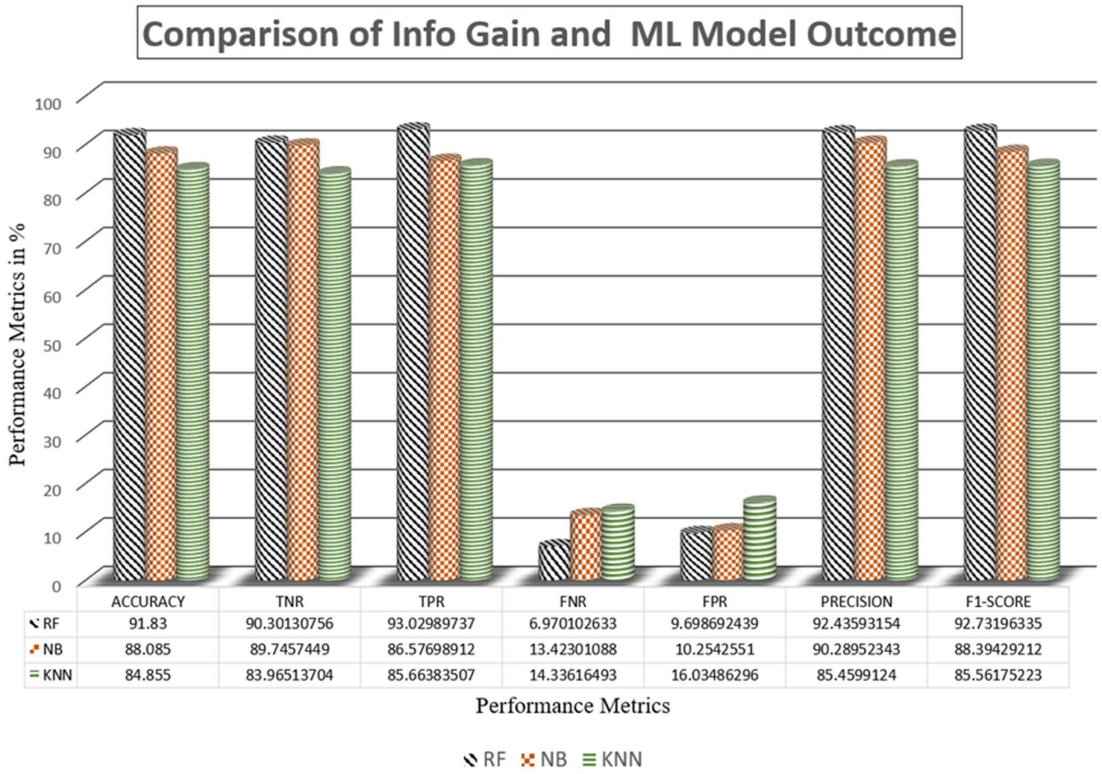


Fig. 3. Comparison bar graph of ML model using info-gain outcome

Figure 3 shows the metric value achieved by the ML algorithm while utilizing the IG features. It follows that RF will have the highest positive metrics number and KNN will have the lowest. The average RF value would be approximately 90% to 93%, the overall NB score would be 86% to 90%, and finally, the KNN value would be 83% to 85%. When the value of a metric is negative, the KNN result will be higher than the RF score.

The above figures 2 and 3 help to identify the best ML model for cyber-attack detection using feature selection outcomes. The figure clearly shows that the CFS and RF model is the better combination of feature selection and ML model when compared to all other five combinations like CFS+KNN, CFS+NB, IG+RF, IG+KNN, and IG+NB.

Conclusion

Because of the combination of physical systems, communication networks, and computation power, cyber-physical systems had made a tremendous change in several real-time applications. But, cyberattacks pose a significant challenge to such infrastructures. Cyber-attacks, in contrast to cyber-physical system flaws caused by accidents, are intentional and hidden. Some attacks involve introducing incorrect information, tampering with existing data, or providing misleading input. Without prior knowledge of these threats, the system will be unable to prevent or mitigate their effects, which could include a decrease in functionality or even complete inoperability. For this reason, it is crucial to modify existing techniques to detect such attacks in these infrastructures. It is vital to utilize the ML model to assist the analysis and assessment of data to find the important features because the data collected in these platforms is generated in such huge numbers, with so much variation, and at high speed. Before doing

the analysis, feature selection was performed in this study. The ML model is developed to alert the network to the existence of an attack. Based on our experimental results, we conclude that a combination of RF and CFS can improve the effectiveness of attack detection and make cyber security safer.

REFERENCES

- Aftergood, Steven. "Cybersecurity: The cold war online." (2017): 30-31.
- Majhi, Santosh Kumar, Ganesh Patra, and Sunil Kumar Dhal. "Cyber physical systems & public utility in India: State of art." Procedia Computer Science 78 (2016): 777-781.
- Jang-Jaccard, Julian, and Surya Nepal. "A survey of emerging threats in cybersecurity." Journal of Computer and System Sciences 80, no. 5 (2014): 973-993.
- Ahmad, Bilal, Wang Jian, and Zain Anwar Ali. "Role of machine learning and data mining in internet security: standing state with future directions." Journal of Computer Networks and Communications 2018 (2018).
- Umarani, S., R. Aruna, and V. Kavitha. "Predicting Distributed Denial of Service Attacks in Machine Learning Field." In 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), pp. 594-597. IEEE, 2022.
- Karmous, Nader, Mohamed Ould-Elhassen Aoueileyine, Manel Abdelkader, and Neji Youssef. "IoT Real-Time Attacks Classification Framework Using Machine Learning." In 2022 IEEE Ninth International Conference on Communications and Networking (ComNet), pp. 1-5. IEEE, 2022.
- Khan, Fazlullah, Ryan Alturki, Md Arafatur Rahman, Spyridon Mastorakis, Imran Razzak, and Syed Tauhidullah Shah. "Trustworthy and Reliable Deep-Learning-Based Cyberattack Detection in Industrial IoT." IEEE Transactions on Industrial Informatics 19, no. 1 (2022): 1030-1038.
- Cui, Lei, Lei Guo, Longxiang Gao, Borui Cai, Youyang Qu, Yipeng Zhou, and Shui Yu. "A Covert Electricity-Theft Cyberattack Against Machine Learning-Based Detection Models." IEEE Transactions on Industrial Informatics 18, no. 11 (2021): 7824-7833.
- Bout, Emilie, Valeria Loscri, and Antoine Gallais. "How Machine Learning changes the nature of cyberattacks on IoT networks: A survey." IEEE Communications Surveys & Tutorials 24, no. 1 (2021): 248-279.
- Berry, Hunter, Mai A. Abdel-Malek, and Ahmed S. Ibrahim. "A Machine Learning Approach for Combating Cyber Attacks in Self-Driving Vehicles." In SoutheastCon 2021, pp. 1-3. IEEE, 2021.
- Khoa, Tran Viet, Dinh Thai Hoang, Nguyen Linh Trung, Cong T. Nguyen, Tran Thi Thuy Quynh, Diep N. Nguyen, Nguyen Viet Ha, and Eryk Dutkiewicz. "Deep transfer learning: A novel collaborative learning model for cyberattack detection systems in IoT networks." IEEE Internet of Things Journal (2022).
- Lakshmanarao, A., A. Srisaila, and T. Srinivasa Ravi Kiran. "Machine Learning and Deep Learning framework with Feature Selection for Intrusion Detection." In 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), pp. 1-5. IEEE, 2022.

- "NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB," Unb.ca, 2020. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html. [Accessed: 21 October 2020].
- Khare, Neelu, Preethi Devan, Chiranji Lal Chowdhary, Sweta Bhattacharya, Geeta Singh, Saurabh Singh, and Byungun Yoon. "Smo-dnn: Spider monkey optimization and deep neural network hybrid classifier model for intrusion detection." Electronics 9, no. 4 (2020): 692.
- Dewa, Zibusiso, and Leandros A. Maglaras. "Data mining and intrusion detection systems." International Journal of Advanced Computer Science and Applications 7, no. 1 (2016).
- Shahbaz, Mahsa Bataghva, Xianbin Wang, Aydin Behnad, and Jagath Samarabandu. "On efficiency enhancement of the correlation-based feature selection for intrusion detection systems." In 2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), pp. 1-7. IEEE, 2016.
- Barot, Virendra, Sameer Singh Chauhan, and Bhavesh Patel. "Feature selection for modeling intrusion detection." International Journal of Computer Network and Information Security (IJCNIS) 6, no. 7 (2014): 56-62.
- Breiman, Leo. "Bagging predictors." Machine learning 24 (1996): 123-140.
- Helman, Paul, Robert Veroff, Susan R. Atlas, and Cheryl Willman. "A Bayesian network classification methodology for gene expression data." Journal of computational biology 11, no. 4 (2004): 581-615.
- Hmeidi, Ismail, Bilal Hawashin, and Eyas El-Qawasmeh. "Performance of KNN and SVM classifiers on full word Arabic articles." Advanced Engineering Informatics 22, no. 1 (2008): 106-111.