A Novel Approach to Detect Intrusion using Machine Learning

Syed Mezbah Ul Hasan[1],Bupasha A Noor Rahman[1], Obantika Roy Anti[1], Tahia Tazin[1]andMohammad Monirujjaman Khan[1*]

[1]Department of Electrical and Computer Engineering, North South University, Bashundhara, Dhaka-1229, Bangladesh

[*]Corresponding Author: Mohammad Monirujjaman Khan. Email: monirujjaman.khan@northsouth.edu

Received: XX Month 202X; Accepted: XX Month 202X

Abstract: An Intrusion Detection System (IDS) is a software application that uses machine learning algorithms to detect network intrusions. Selective logging, privacy protection, reputation-based protection, protection against numerous threats, and dynamic threat response are just a few of the challenges addressed by intrusion detection. Mobile phones, wearable devices, and self-driving cars are all instances of distributed networks that routinely generate large amounts of data. Intrusion detection services are critical for the device's security and privacy. Machine learning (ML) is a subset of artificial intelligence (AI) that enables software programs to improve their predictive accuracy without requiring direct programming. Machine learning and deep learning with intrusion detection systems have gained great attention due to their high categorization accuracy. As a result of technological advancements, cyber-attacks are expected to significantly rise in frequency by 2021. Every day, 30,000 websites globally are hacked. As a result, cybersecurity has become an integral part of our daily digital activities. Data and money are stolen from a diverse range of businesses, financial institutions, and individuals. Our objective is to resolve this issue. The purpose of this study is to analyze an intrusion detection system that makes use of machine learning to boost its efficiency and accuracy. Due to the plethora of research on this subject, the generated model makes use of both a transfer learning technique and a bespoke model to attempt to enhance accuracy. By analyzing the combinations of the most popular feature selection techniques and classifiers, such as K-Nearest Neighbors (KNN) Classification, Decision Tree (DT) Classification, Logistic Regression (LR), and Random Forest (RF) Classification, this article introduces an IDS for networks based on machine learning that has a good union of feature selection techniques and classifiers. The decision tree classifier had a 99.40 percent accuracy rate, while K-Nearest Neighbors (KNN) had a 97.79 percent accuracy rate, Logistic Regression had a 95 percent accuracy rate, and Random Forest had a 99.67 percent accuracy rate.The Random Forest classifier, on the other hand, is the most accurate of them all. The accuracy rate of the models used in this investigation is significantly higher than that of earlier studies, indicating that the models used in this study are more reliable. This study will aid in the reduction of cyber dangers and will benefit businesses, servers, banks, and other organizations seeking to secure their systems from hacking.

Keywords: Intrusion; Detection;AI, Algorithm; Machine Learning; Security;

1Introduction

Intrusion detection systems are designed to identify assaults on computer systems and networks, or on information systems in general. Indeed, it is challenging to establish provably secure information systems and to keep them safe throughout their existence and use. Occasionally, historical or operational restrictions preclude the creation of a completely secure information system. At the moment, the usage of digital

.

technology in all spheres of life, particularly business, is accelerating. Along with the internet's good influence and expansion on worldwide communication, its evil side cannot be ignored. Cyber assaults have also risen in number as a result of the proliferation of linked gadgets. The more attacks, the more powerful and difficult it is to fight against them. Between researchers and hackers, there is an ongoing war for cyber security. To counter increasingly sophisticated attacks, we need specialized hardware or a sophisticated software solution. Thus, intrusion–detection systems are responsible for monitoring the use of such systems in order to identify the emergence of unsafe conditions [1]. Existing intrusion detection and prevention systems must be regularly upgraded to address the most recent threats. Because the internet is advancing at a breakneck pace, attackers' strategies are always altering, making traditional tools increasingly vulnerable. Security professionals are overburdened with maintaining and updating these antiquated systems, which is a lot of work for them. Machine learning (ML) technologies may now be used to identify these intrusions [2].To demonstrate the necessity of ML, several forms of IDS are explored, as well as relevant machine learning algorithms. The research provides a comprehensive review of the use of machine learning in many areas of intrusion detection. Additionally, the methodologies used to implement ML are highlighted, which offers insight into the spectrum of future research areas.

Each day, approximately 30,000 websites are attacked by hackers. Each day, about 25,000 distinct malicious programs are identified. Cyber-crime affects roughly 90,000 individuals every month in the United States of America. Additionally, the United States has the greatest rate of cybercrime, at 23.6 percent [3]. 70% of all data breaches in the world are motivated by monetary gain. Year after year, this number rises. Cybercrime generates more than $1.5 trillion in revenue each year. The increasing number of cyber-intrusions has resulted in the evolution of cyber security as a key part of our everyday online activities. Data security and the privacy of individuals are at high risk [4].Without having to explicitly design them, machine learning enables software programs to increase their accuracy at predicting outcomes. When used to address business problems, machine learning is called predictive analytics [5].

In today's technology - driven world, cyber security has become an integral aspect of our everyday online activity. Strikes are also growing more powerful and tougher to mitigate as the number of attacks increases. For the sake of cyber security, academics are always at odds with hackers. The increasing sophistication of assaults requires either high-end hardware or a well-developed software-based solution. Existing intrusion detection and prevention solutions need ongoing updates to keep pace with new threats. As the online world advances, attackers' techniques evolve as well, leaving old technologies exposed. These legacy systems need regular maintenance and upgrades in order to identify increasingly sophisticated cyber-attacks, which may be rather taxing for security specialists. As a result, a technique based on machine learning has been developed to identify these invasions. Several algorithms based on standard machine learning techniques, such as SVM [6], K-Nearest Neighbor (KNN) [7], ANN [8], Random Forest (RF) [9] and others have been presented and successfully used for an intrusion detection system.

SVM is one of the most successful approaches mentioned before. The primary reason for this is because the distribution of attack types is skewed, such that the learning sample size for low-frequency assaults is insufficient in comparison to high-frequency attacks. The SVM is a margin-based classifier that is extensively used in real-world classification applications. It is based on small-scale learning and has high generalization capabilities [6].The study [10]focuses on network-based anomaly detection. The authors applied our technique to a set of industry-standard benchmark data from MIT Lincoln Lab, namely the 1998 DARPA data. To accelerate the SVM training process, the authors applied clustering analysis to estimate support vectors. They proposed a strategy termed "clustering trees based on SVM" for decreasing the training set and estimating support vectors (CTSVM). Their techniques, however, are less than 70% accurate.

Bhupendra Ingre and Anamika Yadav used ANN to examine the performance of the NSL-KDD dataset in their work. Both binary and five-class categorizations produced the same classification result (type of attack). The findings are evaluated against a range of performance indicators, and an accuracy of around 80% is found[8].

R. Thornton [11] presented a model for detecting unknown threats that combines machine learning (ML)

and deep learning (DL) techniques. The authors reviewed several machine learning and deep learning strategies that fall under the categories of supervised and unsupervised learning. To attain great accuracy, however, a suitable standardized data collection must be provided. According to the various peer-reviewed papers for the research, 2017 had the highest number of publications in journals and conferences devoted to DL and ANN. This trend is continuing, with a rising number of articles including machine learning or deep learning approaches. According to the survey, the most frequently utilized techniques for NIDS are DL and ANN. While KDD199 [12] and NSL KDD [13] are the most often used datasets for NIDS assessment.

The authors of the studies [14-15] developed a paradigm in which various data sources such as logs, packet flow, and sessions are aggregated and provided to machine learning algorithms. They have also argued that the world is evolving toward artificial intelligence (AI), which will emphasize the use of AI in conjunction with automated technologies such as NIDS. Intrusion detection systems based on signatures are ineffective against contemporary threats. As a result, since 2017, the use of machine learning and ANN-based techniques for NIDS has been rising in studies. The authors describe a method for intrusion detection in the study[16]that makes use of the discriminating features of genetic algorithms. Experiments and numerical findings using the KDD CUP 99 and UNSW-NBI5 datasets demonstrated the efficiency. Using the UNSW-NBI5 dataset, they discovered results that were 97.44 percent of normal.

Yi et al. discussed the detection challenge associated with flooding assaults [17-18]. To saturate the connection capacity, an intruder delivers massive packets of attack data as a route request. The author suggested neighbor suppression, a general protection mechanism for MANETs [19] against flooding threats. DoS attacks on wireless networks [20-22]may be quite damaging. He evaluated its efficacy. Yi et al. provide a cross-layer adaptive technique [23]for detecting black and gray hole assaults. Additionally, finite state machine detection [24], distributed intrusion detection [25], wireless intrusion prevention [26], multiagent cooperative intrusion response [27], and green firewall [28] are also intrusion detection technologies.Overall, recent research studies have addressed different aspects of the machine learning process, which is an active research area, such as communication costs, privacy, security, and resource allocation.

In 2021, the number of cyber-attacks has been increased considerably due to advances in technology. Different researchers are relentlessly working on different approaches to ML in order to reduce the number of intrusion attacks occurring every day. As a result, they produced some outputs with these approaches, but their percentage of accuracy is not high enough with respect to the rate of intrusion. The goal of our effort is to find a solution to this problem. The major target of this project is to attain the maximum achievable prediction accuracy for the input data. This ensures that the result is credible.

Four machine learning algorithms were used in our study, and all of them were accurate to greater than 94 percent. With the Random Forest classifier, we achieved the greatest accuracy of 99.67 percent. As a result, the accuracy gap in our research was closed. The advantage of using these four models is that they provide comparative analysis. These comparisons enable us to determine which model provides the best level of accuracy. Any business may simply adopt our model to detect intrusions by applying it. It will help any company save a lot of confidential information, privacy and money.One of the most important contributions of this study is the use of publically accessible datasets to test various machine learning models. Several important models were previously used by the vast majority of the researchers. As a consequence, we tested four alternative models and also compared our findings to earlier research. The next section provides a succinct summary of all the findings and comparisons. The following is the remainder of the article: Experiment technique and methods are described in Section 2, while findings are analyzed in Section 3 and the conclusion discussed in Section 4.

2Procedure and Experimental Methodology

This section contains a thorough explanation of the dataset as well as a diagram of the proposed system and also diagrams of numerous algorithms and evaluation matrices, in addition to information on the study's procedure and methodology.

2.1 Proposed System

In order to construct a model, data needs to be available after completing the preprocessing part. To develop the model, a preprocessed dataset and machine learning algorithms are required. The models implemented include Random Forest (RF), Decision Tree (DT), K-Nearest Neighbor (KNN), and Logistic Regression classifier. The accuracy measurements obtained from these models, namely the accuracy score, will be utilized for comparison analysis. In Figure 1,the proposed system's block diagram isshown.

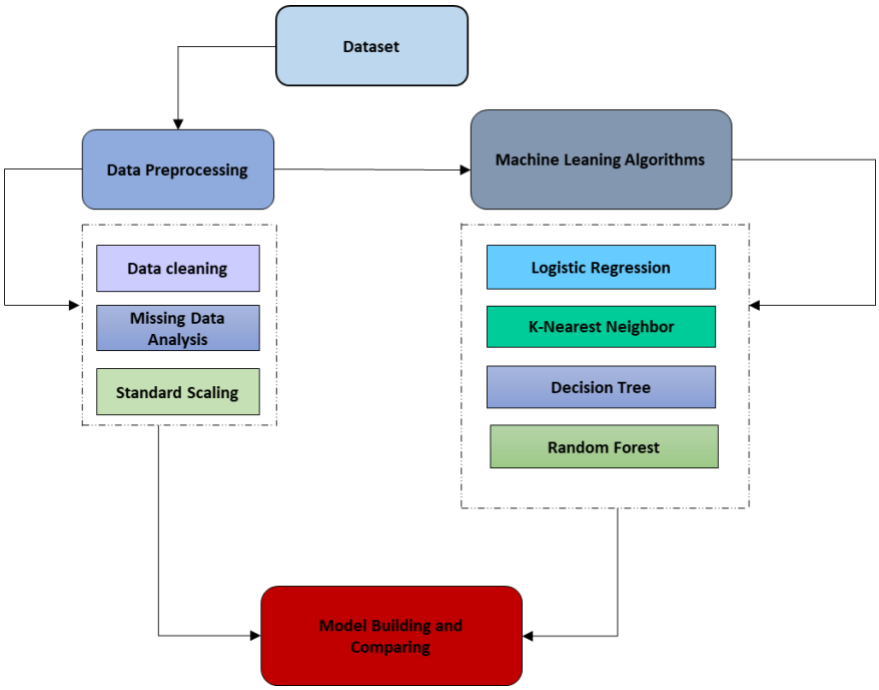

Figure 1: Block Diagram of the Proposed System

2.2 Dataset

The research was based on data from an intrusion detection system dataset [29]. This dataset has 25192 rows and 41 columns. The output column class has a value of either normal (0) or anomaly (1). The value 0 indicates that no attack has been detected, whereas 1 indicates that an assault has been detected.

2.3 Data Preprocessing

Data preprocessing is the process of preparing raw data in such a manner that it may be utilized directly by a machine learning model. It is the first and most critical step in constructing a machine learning model. It is fairly uncommon for real-world data to have noise, missing values, null values, and be in an inappropriate format for machine learning models. The accuracy and efficiency of a machine learning model may be increased by doing data preparation operations such as cleaning and preparing the data for use in the model. Our machine learning model may suffer significantly if our dataset contains missing data. As a consequence, it is critical to address the missing values as well as fill in the null values in the dataset. The dataset is examined for null and missing values. Because our dataset has no missing or null values, it will remain unchanged. The Label Encoding function turns the string values of the dataset to computer-readable integer values. That is why it must be changed in order to train models. Three string values have

been converted to integers. Due to the balanced nature of the dataset, more precision may be obtained.To increase the accuracy and efficiency of this work, the data is separated into train and test segments, with an 80/20 ratio of training to assessment. Following the split, the classifier is constructed using a number of classifiers to improve its accuracy. Random Forest (RF), Decision Tree (DT), K-Nearest Neighbor (KNN), and Logistic Regression are the classification techniques used in this research.

2.4 Proposed Algorithms

The core function of network security is intrusion detection systems (IDS). IDSs continuously monitor a system's events inside a network and evaluate its activity in addition to identifying intrusions. The research evaluated four machine learning approaches for predicting intrusions using the publicly available intrusion detection system dataset. They are as follows:

- Random Forest
- Decision Tree
- K-Nearest Neighbor
- Logistic Regression

2.4.1 Random Forest

The Random Forest Classifier [38] was utilized as the first algorithm. It comprises several independent trees referred to as decision trees in RF that are finally trained using training sample data. Following that, the outcomes from all these trees are subjected to a voting procedure in order to generate estimates. The ultimate outcome is determined by the majority of votes using an RF classifier. Figure 2,illustrates a block diagramof RFclassifiers.

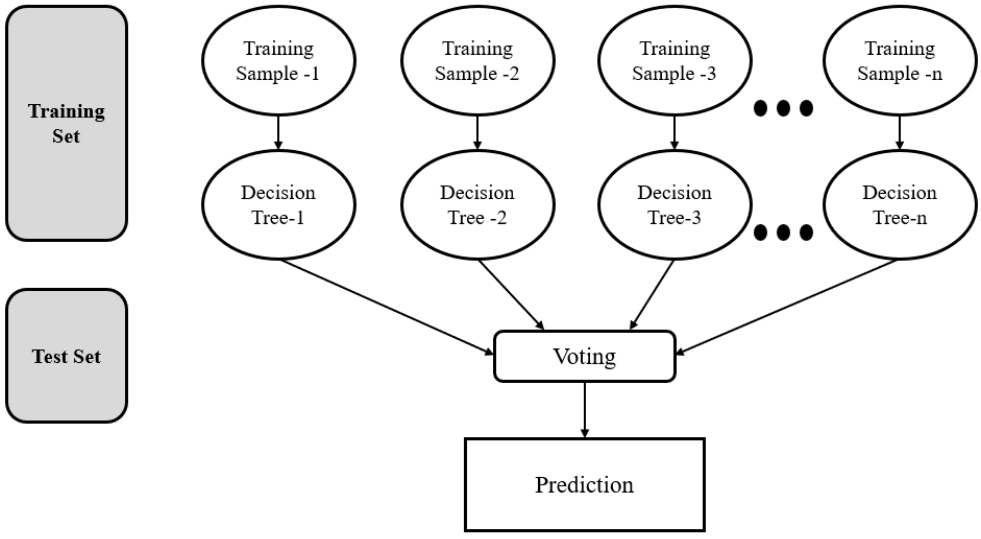

Figure2: Diagram of Random Forest Classifier[30]

Due to the fact that RF requires less training time than other algorithms, it will help us get better outcomes. The size of the dataset has little effect on the accuracy, since it produces great accuracy on larger data. The absence of large volumes of data has no influence on accuracy[31]. It also works well since the

standard hyperparameters it uses generally convey reasonable expectations. Acknowledging the hyperparameters is crucial since there are so few[32]. The primary issue that commonly arises in machine learning is overfitting. However, it happens less often in the Random Forest. Sufficient trees assist the classifier in avoiding overfitting.

2.4.2 Decision Tree

Decision tree algorithms are heavily used in machine learning to handle classification and regression issues[33]. A root node generates two types of nodes: the internal node and the leaf node. Internal nodes are characterized as decision makers due to their various branches, while leaf nodes provide output due to their lack of further branches. As a result, it configures the structure of a tree. The fundamental design of the Decision Tree classifier is representedin Figure 3.

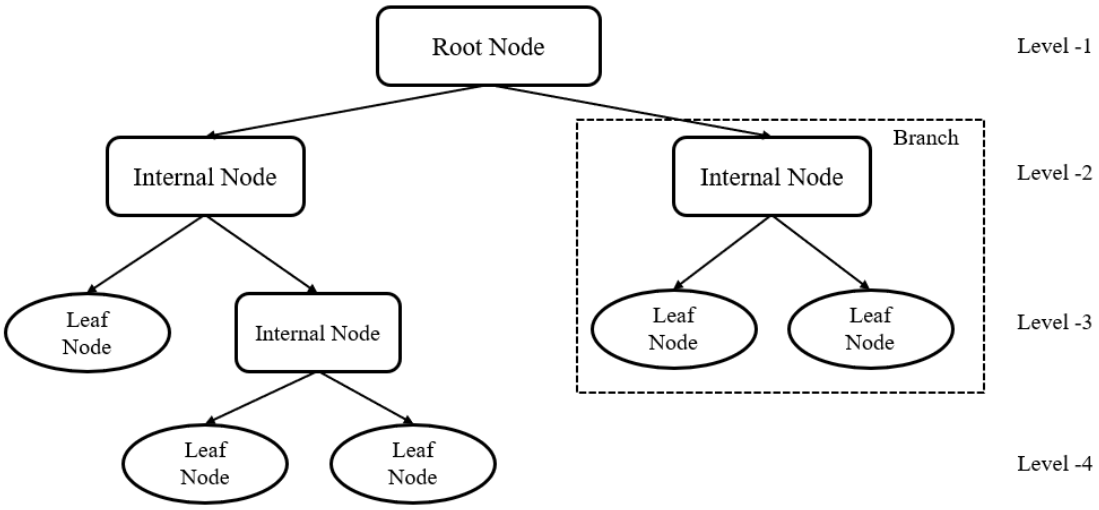

Figure3: Diagram of Decision Tree Classifier

Data cleaning is the least of our concerns when it comes to DT. It is easily comprehensible owing to its tree structure. The DT is simple to explain since it matches the stages that an individualexperience while making a real-world choice[40].

2.4.3 K-Nearest Neighbor

Numerous machine learning methods are supervised learning-based. K-Nearest Neighbor[34] is one of the most uncomplicated. The K-NN method saves all available data and assigns a new data point a classification based on its similarity to an existing data point. This implies that when fresh data is generated, it can be simply categorized using the K-NN algorithm into a suitable category[35]. In Figure 5, the diagram of K-NN is shown.

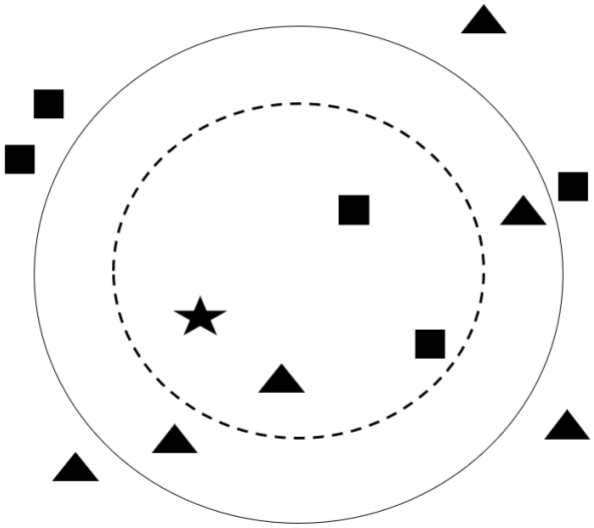

Figure 4: Diagram of K-Nearest Neighbor Classifier[36]

The rationale for selecting KNN is the ease with which it can be implemented. It is more effective when dealing with huge amounts of training data. It is a robust framework for dealing with noisy training data.

2.4.4 Logistic Regression Classifier

Binary outcomes are modelled using the well-known statistical technique of logistic regression. In statistics research, logistic regression is carried out using a variety of different learning methods. The Logistic Regression algorithm was developed using a variation of the Neural Network approach. In many aspects, this approach is similar to neural networks, but it is easier to set up and utilize. The block diagram of logistic regression is shown in Figure 5.

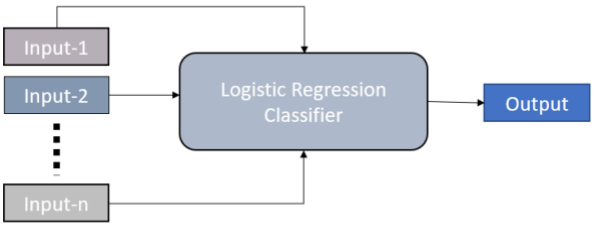

Figure5: Diagram of Logistic Regression Classifier

2.5 Evaluation Matrix

The evaluation matrix is an indicator that shows the execution efficiency of ML algorithms in terms of the confusion matrix. The list of all models will be evaluated using the confusion matrix. The confusion matrix shows how frequently our models make accurate and erroneous guesses. False positives and negatives will be assigned to values that are poorly protected, while real positives and negatives are assigned to correctly predicted values as shown in Figure 6. To evaluate the effectiveness of the algorithm after combining all estimated values in the matrix, the accuracy, precision-recall trade-off, and AUC were all measured and calculated.

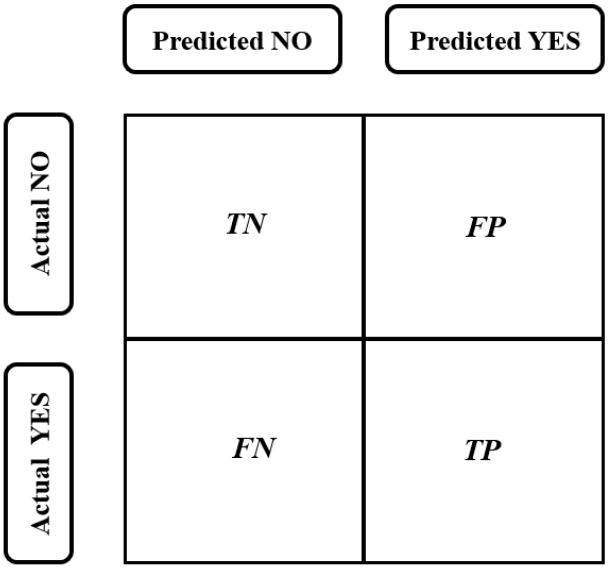

Figure6: Block Diagram of Evaluation Matrix

3 Result Analysis

3.1 Model Accuracy

The accuracy of four distinct models is shown in Table 1.

Table 1: Accuracy list of different Machine Learning Model

Model Name	Accuracy (%)
Random Forest	99.38
Decision Tree	99.22
Logistic Regression	97.88
K-Nearest Neighbor	94.83

In this case, based on the data, the random forest classifier has the maximum accuracy of 99.67 percent. The Decision Tree Classifier achieves the second greatest accuracy, 99.40 percent, which is extremely close to the Random Forest Classifier. The lowest level of precision is 95 percent, which comes from Logistic Regression classifier.

3.2 Evaluation Matrix of the Model

3.2.1 Random Forest

The prediction made by the random forest model is seen in Figure 7.

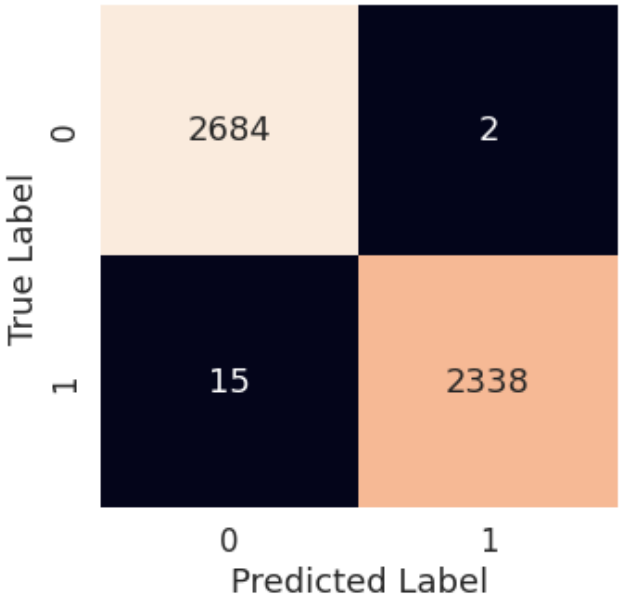

Figure 7: Confusion matrix of random forest classifier

The confusion matrix depicts the projected result and the model's computed performance. There were 5022 correct predictions and 17 incorrect ones.

3.2.2 Decision Tree

In Figure 8shows the prediction made by decision tree model.

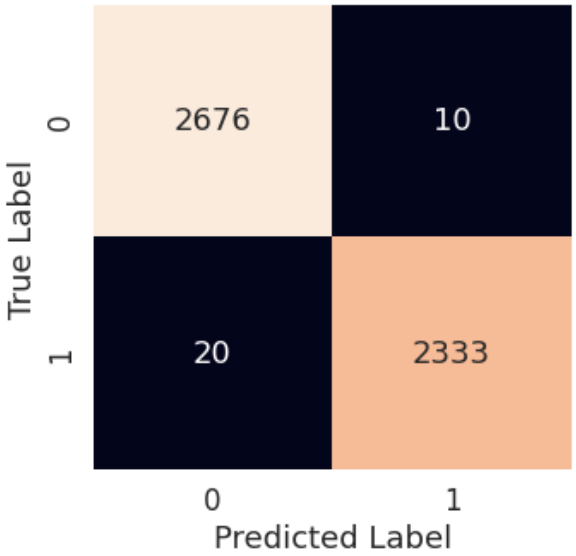

Figure 8: Confusion matrix of decision tree classifier

There were 5009 correct guesses and 30 wrong guesses.

3.2.3Logistic Regression

Figure 9 depicts the Logistic Regression's prediction. The total number of correct predictions is 5954, while the total number of incorrect predictions is 344.

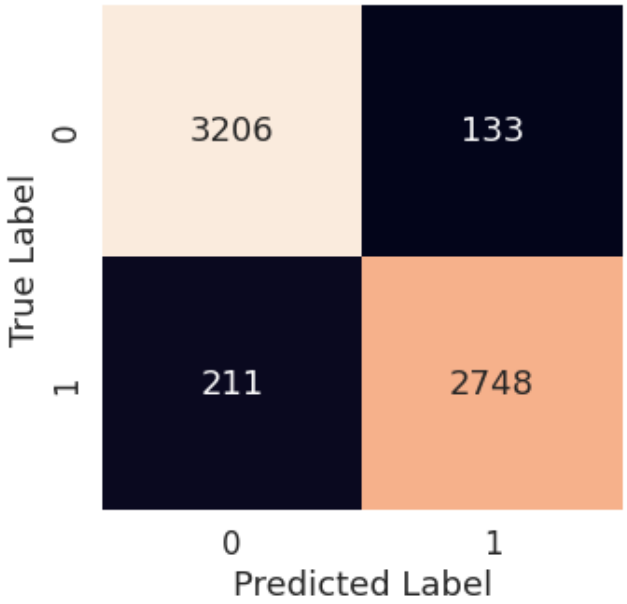

Figure 9: Confusion matrix of Logistic Regression classifier

3.2.4 K-Nearest Neighbor

Figure 10 depicts the K-Nearest Neighbor's prediction. There have been 5009 correct predictions, compared to 30 incorrect ones.

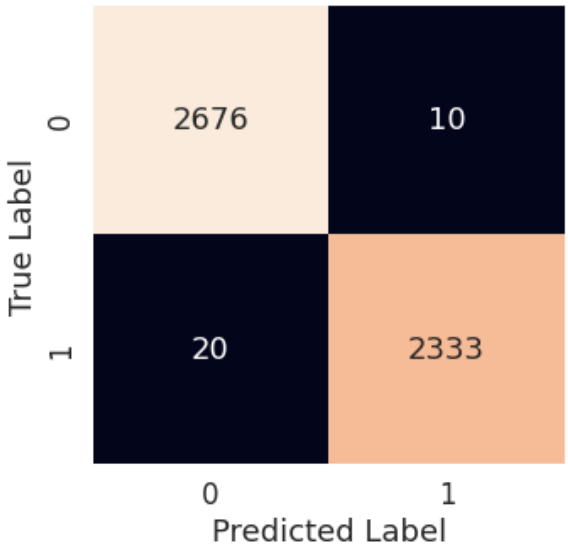

Figure 10: Confusion matrix of k-nearest neighborclassifier

3.3 Model Comparison

As shown in Table 2, the models are compared to those previously studied. It is evident from the table that the Random Forest model outperforms all others in the framework.

Table 2: Performance Comparison

This Paper (Model Name)	Accuracy (%)	Reference Paper (Model Name)	Accuracy (%)
Random Forest	**99.67**	Ref [16]SVM	97.45
Decision Tree	99.40	Ref [30]Neural Fuzzy	95.3
Logistic Regression	95.0	Ref [33]MLP	94.5
K-Nearest Neighbor	97.79	Ref [34]K-Nearest Neighbor	78

Although all of the algorithms in Table 2 have an acceptable degree of accuracy, it is obvious that only the random forest approach offers a significant advantage in terms of precision. The Random Forest method was used in this study to achieve 99.67 percent accuracy, however the authors of [16] achieved just 97.45 percent accuracy using SVM method.In addition, this paper's decision tree approach yielded a precision rate of 99.40 percent, whereas using Neural Fuzzy, the authors in [30] only managed a precision rate of 95.3 percent.Finally, this work attained 97.79 percent accuracy utilizing the K-Nearest Neighbor technique, compared to 78 percent accuracy achieved by the authors in [34].While K-Nearest Neighbor isn't as good as the other algorithms in this paper, other algorithms are better.

4 Conclusion

Security is a major concern for all companies' and institutions' networks at the moment. Intrusions are attempting to gain successful access to the data networks of these companies and Web services, despite the development of multiple methods to ensure the infiltration of intrusions to the network infrastructure[37]. The goal of intrusion detection systems (IDS) is to facilitate computer systems in dealing with attacks. The development of a machine learning model could facilitate the early detection of intrusion and subsequent mitigation of its severe consequences. This study focuses on the efficiency of several ML algorithms in appropriately anticipating intrusions based on a number of numerical variables. With a classification accuracy of 99.67 percent, random forest classification outperforms the other methods tested. The framework models may be boosted in the future using a larger dataset and machine learning models such as ExtraTreesClassifier and Voting Classifier. This will improve the framework's constancy as well as its demonstration. The future of intrusion detection systems will be to interpret and apply these techniques for detecting new and emerging threats. In exchange for just offering some basic info, the machine learning architecture may assist the general public in estimating the probability of a network intrusion occurring. In an ideal world, it would assist businesses in obtaining early detection of intrusions, allowing them to protect highly classified information and financial resources.

Data Availability Statement: The data used to support the findings of this study are freely available at https://www.kaggle.com/munaalhawawreh/xiiotid-iiot-intrusion-dataset

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors would like to confirm that there are no conflicts of interest regarding the study.

Reference

[1] H. Debar, "An Introduction to Intrusion-Detection Systems," 06 01 2009. [Online]. Available: https://www.researchgate.net/publication/228589845_An_Introduction_to_Intrusion-Detection_Systems. [Accessed 20 11 2021].

[2] S. Agrawal, S. Sagnik, A. Ons, Y. Gokul, P. Kandaraj*et al*, "Federated Learning for Intrusion Detection System:" *Federated Learning for Intrusion Detection System:* 2021.

[3] Rapid7, "The Pros & Cons of Intrusion Detection Systems | Rapid7 Blog," Rapid7, 11 01 2017. [Online]. Available: https://www.rapid7.com/blog/post/2017/01/11/the-pros-cons-of-intrusion-detection-systems/.

[4] W. Goddard, "Cyber Security Statistics 2020," ITChronicles, 28 05 2021. [Online]. Available: https://itchronicles.com/information-security/cyber-security-statistics-2020/. [Accessed 22 11 2021].

[5] S. Bag, "Federated Learning for Beginners | What is Federated Learning," Analytics Vidhya, 15 05 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/federated-learning-a-beginners-guide/. [Accessed 20 11 2021].

[6] F. Kuang, W. Xu and S. Zhang, "A novel hybrid KPCA and SVM with GA model for intrusion detection," *Appl. Soft Comput.,* vol. 18, p. 178–184, 2014.

[7] W. Li, P. Yi, Y. Wu, L. Pan and J. Li, "A new intrusion detection system based on KNN classification algorithm in wireless sensor network," *J. Elect. Comput. Eng.,* vol. 2014.

[8] B. Ingre and A. Yadav, "Performance analysis of NSL-KDD dataset using ANN," *Proc. Int. Conf. Signal Process. Commun. Eng. Syst.,* p. 92–96, 2015.

[9] N. Farnaaz and M. A. Jabbar, "Random forest modeling for network intrusion detection system," *Procedia Comput. Sci.,* vol. 89, p. 213–217, 2016.

[10] L. Khan, M. Awad and B. Thuraisingham, "A new intrusion detection system using support vector machines and hierarchical clustering," 31 08 2006. [Online]. Available: https://link.springer.com/article/10.1007/s00778-006-0002-5. [Accessed 22 11 2021].

[11] Thornton, "AT&T Business and Cybersecurity," 20 07 2020. [Online]. Available: https://cybersecurity.att.com/solutions/intrusion-detection-system/ids-explained. [Accessed 20 11 2021].

[12] C. I. f. Cybersecurity, "NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB," University of New Brunswick, [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html.

[13] B. Stephen D., F. K. Dennis, P. Michael J. and S. Padhraic, "UCI Machine Learning Repository: KDD Cup 1999 Data Data Set," archive.ics.uci.edu, 01 01 1999. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/kdd+cup+1999+data.

[14] K. G. A. H. S. G. L. Mehra, "An effectual & secure approach for the detection and efficient searching of Network Intrusion Detection System (NIDS)," in *2015 International Conference on Computer, Communication and Control (IC4)*, 2015.

[15] B. L. H, "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey.," *MDPI, Applied Sciences,* vol. 9, no. 20, p. 4396, 2019.

[16] G. Hossein and H. Hamid, "A new feature selection IDS based on genetic algorithm and SVM," 20 03 2017. [Online]. Available: https://ieeexplore.ieee.org/document/7881798. [Accessed 15 11 2021].

[17] Y. Ping, H. Yafei, Z. Yiping, Z. Shiyong and D. Zhoulin, "Flooding attack and defence in ad hoc networks," *Journal of Systems Engineering and Electronics,* vol. 17, no. 2, pp. 410-416, 2006.

[18] P. Yi, T. Zhu, Q. Zhang, Y. Wu and J. Li, "A denial of service attack in advanced metering infrastructure network," *2014 IEEE International Conference on Communications (ICC),* 2014

[19] P. Yi, Z. Dai, S. Zhang and Y. Zhong, "A New Routing Attack in Mobile Ad Hoc Networks," *International Journal of Information Technology,* vol. 11, no. 2, 2017.

[20] P. Yi, F. Zou, V. Zou and Z. Wang, "Performance analysis of mobile ad hoc networks under flooding attacks," *Journal of Systems Engineering and Electronics,* vol. 22, no. 2, pp. 334-339, 2011.

[21] P. Yi, Y.-k. Zhou, Y. Wu and N. Liu, "Effects of denial of service attack in mobile ad hoc networks," Journal of Shanghai Jiaotong University (Science), vol. 14, no. 5, p. 580, 2009.

[22] P. Yi, J. Cai, Y. Wu and Y. Li, "Impact of two kinds of DoS attacks on mobile ad hoc networks," 2009.

[23] P. Yi, T. Zhu, N. Liu, Y. Wu and J. Li, "Cross-layer Detection for Black Hole Attack in Wireless Network," *Journal of Computational Information Systems,* vol. 8, no. 10, pp. 4101-4109, 2012.

[24] P. Yi, Z. Wang and Y. Wu, "Intrusion Detection for Wireless Mesh Networks using Finite State Machine," China Communication, 2010.

[25] Y. Ping, J. Xinghao, W. Yue and L. Ning, "Distributed intrusion detection for mobile ad hoc networks," *Journal of Systems Engineering and Electronics,* vol. 19, no. 4, p. 851–859, 2008.

[26] P. Shanghai, J. Tong, T. Zhu, Y. Wu, P. Yi*et al*, "An Intrusion Prevention Mechanism in Mobile Ad Hoc Networks An Intrusion Prevention Mechanism in Mobile ad hoc Networks," *Ad Hoc & Sensor Wireless Networks,* vol. 17, pp. 269-292.

[27] Y. Ping, Z. Futai, J. Xinghao and L. Jianhua, "Multi-agent cooperative intrusion response in mobile adhoc networks," *Journal of Systems Engineering and Electronics,* vol. 18, no. 4, p. 785–794, 2007.

[28] P. Yi, T. Zhu, Q. Zhang, Y. Wu and J. Li, "Green firewall: An energy-efficient intrusion prevention mechanism in wireless sensor network," *IEEE Xplore,* p. 3037–3042, 2012.

[29] Network Intrusion Detection. Available on: https://www.kaggle.com/sampadab17/network-intrusion-detection

[30] Scikit-learn, "3.2.4.3.1. sklearn.ensemble.RandomForestClassifier — scikit-learn 0.20.3 documentation," Scikit-learn.org, 2018. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html. [Accessed 5 1 2022].

[31] "Machine Learning Random Forest Algorithm - Javatpoint," www.javatpoint.com, [Online]. Available: https://www.javatpoint.com/machine-learning-random-forest-algorithm. [Accessed 5 1 2022].

[32] T. Tazin, M. N. Alam, N. N. Dola, M. S. Bari, S. Bourouis*et al*, "Stroke Disease Detection and Prediction Using Robust Learning Approaches," *Journal of Healthcare Engineering,* p. e7633381, 2021.

[33] "sklearn.tree.DecisionTreeClassifier — scikit-learn 0.22.1 documentation," Scikit-learn.org, 2019. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html. [Accessed 6 1 2022].

[34] "sklearn.neighbors.NearestNeighbors," scikit-learn, [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.NearestNeighbors.html?highlight=k%20nearest#sklearn.neighbors.NearestNeighbors.kneighbors. [Accessed 6 1 2022].

[35] "K-Nearest Neighbor(KNN) Algorithm for Machine Learning - Javatpoint," www.javatpoint.com, [Online]. Available: https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning. [Accessed 6 1 2022].

[36] Y. Che, Y. Ju, P. Xuan, R. Long, F. Xing *et al*, "Identification of Multi-Functional Enzyme with Multi-Label Classifier," *PLOS ONE,* vol. 11, no. 4, p. e0153503, 2016.

[37] "sklearn.ensemble.AdaBoostClassifier — scikit-learn 0.22.1 documentation," scikit-learn.org, [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html. [Accessed 6 1 2022].