VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

BCI 3002: Disaster Recovery and Business Continuity Management (DRBCM)

Slot: A1+TA1

Research Paper

TITLE: PREVENTION OF ATTACKS USING ONE CLASS CLASSIFICATION AND AUTOENCODERS

29th November 2021

Team Leader:
Rohan Allen 18BCI0247

Team Members:
Rakshith Sachdev 18BCI0109
Harshita Pundir 18BCI0192

ABSTRACT

In this paper, we represent how one-class classifiers are prepared to utilize generous information to recognize ordinary and dangerous traffic redirected to an endpoint device. In this venture, the framework is prepared to utilize unsupervised / one-class-based demonstrating approaches by which the framework would comprehend the issues that we would confront day by day, and the preparation will be useful for what's to come. After the preparation of the framework, the framework can be utilized in reality to confront ongoing, new difficulties and by gaining from the past experiences it can develop as indicated by the users' issues, weaknesses, dangers, and conditions

KEYWORDS

Autoencoders, feature generation, machine Learning, malware detection, malware profiling, network protection, one-class classification,

INTRODUCTION

With today's day and age rapidly becoming digital, the network and endpoint devices become a target for attacks and exploitation; thus, the systems have long been associated with issues related to security. Therefore, making systems secure and safe is of extreme importance. As per the 2020 Unit 42 Threat Report, practically all traffic is decoded, implying that the majority of classified and indiIntrusion detection using deep sparse auto-encoder and self-taught learning, Qureshi by A. S., visual user information in the network is highly powerless against cyber attacks. Network security is utilized to delay unintentional harm which can be done to the network's private information, its users, or its devices. The main aim of network security is to secure the network running and for every single authentic client.

The objective of this project is to use machine learning to teach our system to distinguish between malicious network traffic, which could contain a virus or malware, and regular network data. The model has also been taught to detect and avoid fake data and the deployment of any specific software while there is a threat of an attack, which results in dramatically reducing the financial stress on an organization and prevents tarnishing their reputation. The major goal is to offer the most accurate findings possible by utilizing methods such as unsupervised learning/one-class-based modeling, thereby lowering processing time substantially.

In this model, we are using one-class classifications and autoencoders so that the system can detect bad traffic more accurately. With the help of this model, we can predict false data, and therefore, we can prevent the installation of software at the time of any risks to decrease the cost.

LITERATURE SURVEY

TITLE, AUTHOR, AND JOURNAL	OBJECTIVE	METHODOLOGY USED	LIMITATION
Khan, A., Shamim, N., & Durad, M. H. (2019) Neural Computing and Applications (2019). [1]	This paper gives the idea of self-education figuring out how to prepare the deep neural network for network intrusion detection.	In the proposed strategy, a Self Taught based Transfer Learning(DST-TL) is used to remove the highlights from the NSL-KDD dataset a relapse-related pre-prepared network is utilized.	This system cannot classify the different types of attacks as deep neural networks are not used.
A Clustering-based Shrink AutoEncoder for Detecting Anomalies in Intrusion Detection Systems by Bui, T. C., Hoang, M., & Nguyen, Q. U. (2019, October) 2019 11th International Conference on Knowledge and Systems Engineering (KSE). IEEE, 2019. [2]	This paper provides the hybrid model of the K-Means clustering algorithm and Shrink AutoEncoder(SAE) to lessen the limitations in handling the datasets.	In the proposed method, the hybrid model of the K-Means clustering algorithm and Shrink AutoEncoder is used to detect the anomalies occurring in the network. With the help of this method, datasets can also be handled properly.	It failed to extend these works by using better clustering algorithms and other metrics to find a suitable number of clusters in the data.
A Modular Multiple Classifier System for the Detection of Intrusions in Computer Networks Giorgio Giacinto, Fabio Roli, Luca Didaci Department of Electrical and Electronic Engineering, University of Cagliari, Italy. [3]	This paper provides a strategy to use Intrusion Detection Systems (IDS) and pattern recognition to increase the level of security on computer networks. The authors later discuss and evaluate the effectiveness of the IDS on the security of the network.	Utilizing IDS and example acknowledgment ways to deal with network intrusion detection dependent on the combination of numerous classifiers. Specifically, the author centers around Modular Multiple Classifier engineering plans where every	This study presents an extensive report on how IDS and pattern recognition can be used to provide higher levels of network security. The paper includes a descriptive report on how the authors' used methods like machine learning in their implementation and

		module in the design can identify intrusions against the administrations offered by the secured network.	design and also compared different training sets on the data to conclude which model is most effective. Although this paper is detailed, our methodology of achieving a system that can differentiate between good and bad networks containing viruses and malware includes methods and techniques using One-Class Classification and Auto Encoders.
A Clustering-based Shrink AutoEncoder for Detecting Anomalies in Intrusion Detection Systems Bui, T. C., Hoang, M., & Nguyen, Q. U. (2019, October) In 2019 11th International Conference on Knowledge and Systems Engineering (KSE) (pp. 1-5). IEEE. [4]	A detailed examination and investigation of different AI procedures have been completed to discover the reason for issues related to different AI strategies in identifying intrusive exercises.	AI methods have been examined and looked at as far as their recognition ability for distinguishing the different classes of assaults. Limits related to every classification of them are additionally talked about. Different data mining apparatuses for AI have additionally been remembered for the paper.	Existing literature is described which is based on similar techniques with most of the popular datasets as on date to generalize our observations. All the techniques have not been implemented to evaluate the performance to ensure that results are reproducible.

A novel statistical analysis and autoencoder-driven intelligent intrusion detection approach. Ieracitano, C., Adeel, A., Morabito, F. C., & Hussain, A. (2020). Neurocomputing, 387, 51-62. [5]	Statistical examination and autoencoder (AE) driven insightful intrusion detection (IDS) framework is acquainted with recognizing and alleviating the dangers of programmers growing much more complex and risky malware assaults that make intrusion detection a troublesome assignment.	Initially, the NSLKDD dataset is cleaned from anomalies and the min-max standardization procedure is utilized to scale data inside the range 0 and 1. Subsequently, the one-hot-encoding is applied to change over symbolic (or categorized) features into numeric quantities. At that point, the 38 numeric attributes are investigated statistically to choose the most associated features. At last, shallow (MLP, L-SVM, Q-SVM, LDA, QDA) and deep (AE, LSTM) networks are created to quantify the detection execution both in parallel and multi-classification situations.	The advancement of more precise deep architectures that can oversee continuous real-time data streams like NSL-KDD examples to recognize malicious attacks progressively can build its proficiency. Also, to misuse long-term learning, quicker choice models along with decreased computational complexity for constant needs to execute in this system.

J. K. Chahal, V. Gandhi, P. Kaushal, K. R. Ramkumar, A. Kaur and S. Mittal, "KAS-IDS: A Machine Learning based Intrusion Detection System," *2021 6th International Conference on Signal Processing, Computing and Control (ISPCC)*, 2021.[6]	Regrettably, the majority of commercial IDSs are based on abuse and are only designed to detect known threats. These require frequent signature updates and have a limited capacity for detecting new assaults. As a result, this study proposes an anomaly-based IDS as a viable solution to this challenge.	KAS-IDS, or K-Means and Adaptive SVM-based Intrusion Detection System, is the approach proposed in this study. K-Means were used in the first stage to creating data clusters, and adaptive SVM was used in the second step to classify the data.	As part of the current technique, the data is split into two categories: normal and abnormal data and correct findings are obtained using the NSL-KDD dataset, which can also be used for real-time traffic analysis. Apart from that, the intelligent agents of clustering and classification algorithms can improve their performance in real-time traffic analysis.
G. Yedukondalu, G. H. Bindu, J. Pavan, G. Venkatesh and A. SaiTeja, "Intrusion Detection System Framework Using Machine Learning," 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA), 2021[7].	The main purpose of this project is to compare and assess the effectiveness of neural network models on a data set.	To identify intrusion rates, the suggested application uses the SVM (Support Vector Machine) and ANN (Artificial Neural Networks) algorithms. Each algorithm is used to determine if the data being requested is allowed or includes any irregularities. These techniques employed feature selection algorithms based on correlation and Chi-Squared to decrease the dataset by removing unnecessary data.	Another dataset with a larger number of characteristics might be used to improve this research. Because ANN provides greater accuracy but slower calculations, we can apply alternative efficient algorithms in the future that may provide greater accuracy while also computing faster, allowing it to be employed in real-time applications.

D. Xuan, H. Hu, B. Wang and B. Liu, "Intrusion Detection System Based on RF-SVM Model Optimized with Feature Selection," 2021 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), 2021.[8]	The goal of this research is to improve network intrusion detection using machine learning. The two-stage IDS suggested in this research is based on machine learning algorithms RF and SVM that are tuned with the Feature Ranking CFS.	In this research, they present a two-stage IDS based on machine learning models RF and SVM tuned with the CFS method, and they tested it on NSL-KDD benchmark datasets, contrasting it to the RF and SVM modeling.	We plan to develop the IDS in the future with the goal of increasing the detection precision of low-frequency assaults, which is a common difficulty in IDS. Furthermore, the capacity of an IDS to identify unknown forms of threats is taken into account. We'd want to increase the accuracy of detecting unknown forms of assaults.
A. Aljohani and A. Bushnag, "An Intrusion Detection System Model in a Local Area Network using Different Machine Learning Classifiers," *2021 11th International Conference on Advanced Computer Information Technologies (ACIT)*, 2021.[9]	The article presents a security control for an IDS that is used to identify known and unknown attacks in order to avoid security concerns in LANs.	The suggested system uses Neural Network and Support Vector Machine (SVM) models for intrusion detection to avoid security risks in a Local Area Network (LAN). The KDD99 dataset is used to test the suggested method. The KDD99 is an anomaly-based detection benchmark. This method detects assaults quickly and effectively.	In future research, the KDD99 dataset may be used to classify threat categories to see which machine learning classifier performs better.

H. Elbahadır and E. Erdem, "Modeling Intrusion Detection System Using Machine Learning Algorithms in Wireless Sensor Networks," 2021 6th International Conference on Computer Science and Engineering (UBMK), 2021.[10]	An intrusion detection system (IDS) is modeled in this work to assure WSN security. Because signature, misuse, and anomaly-based intrusion detection approaches are insufficient to offer security on their own, a hybrid model is presented in which these methods are combined.	Anomaly criteria were created for attack detection, and the BayesNet, J48, and Random Forest machine learning algorithms were employed to categorize normal and anomalous traffic in the hybrid model.	Application of alternative efficient algorithms in the future that may provide greater accuracy while also computing faster, allowing it to be employed in real-time applications.
S. T. Zargar, J. Joshi, and D. Tipper, “A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks,” IEEECommun. Surveys Tuts., vol. 15, no. 4, pp. 2046–2069, Mar. 2013. [11]	This paper describes the developing creative, effective, efficient, and comprehensive prevention, detection, and response mechanisms that address the DDoS flooding problem before, during, and after an actual attack.	The paper provides the knowledge required about the attacks as this project works on the prevention of attacks. It stimulates our research into understanding how the attacks occur which in turn helps with solutions to prevent them.	Existing literature is described which is based on similar techniques with most of the popular datasets as on date to generalize our observations. All the techniques have not been implemented to evaluate the performance to ensure that results are reproducible.
V. Balamurugan and R. Saravanan, “Enhanced intrusion detection and prevention system on cloud environment using hybrid classification andOTS generation,” Cluster Comput., vol. 22, pp. 1–13, Nov. 2017. [12]	The proposed system in this paper is directed on intrusion detection systems and it uses cloudlet controller, trust authority, and virtual machine management in cloud environments.	The proposed classifier in this paper effectively detects the attackers which are experimentally proved by comparing with existing classification models which provide a very good insight.	Application of alternative efficient algorithms in the future that may provide greater accuracy while also computing faster, allowing it to be employed in real-time applications.

J. Yan, D. Jin, C. W. Lee, and P. Liu, ‘‘A comparative study of off-line deep learning based network intrusion detection,’’ in Proc. 10th Int. Conf.Ubiquitous Future Netw. (ICUFN), Jul. 2018, pp. 299–304. [13]	It intends to answer some of the research questions specified in and random forest classifiers, which are shallow. The authors indicated that their approach could reduce the SVM’s training and testing times in both binary and multiclass classifications and improve the prediction accuracy of the SVM.	They used ISCX 2012 dataset and achieved ten features from it by using an auto-encoder and feeding it to the SVM for its training. The authors indicated the benefits of their method regarding metrics like Kappa statistics, detection rate, accuracy, and FPR.	It is only able to conduct the binary classification and cannot handle the multiclass attack traffics.
M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, ‘‘Deep learning approach combining sparse autoencoder with SVM for network intrusion detection,’’ IEEE Access, vol. 6, pp. 52843–52856, 2018. [14]	This paper proposes a scheme that uses the self-taught learning framework for the learning of features and reduction of dimension. This IDS model benefits from a sparse auto-encoder for unsupervised reconstructing a new feature representation.	The authors indicated that their approach could reduce the SVM’s training and testing times in both binary and multiclass classificationsand improves the prediction accuracy of the SVM.	The efficiency of this approach in binary and multiclass classifications is evaluated against the naive random forest classifiers, which are shallow.
B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, ‘‘Deep learning for classification of malware system call sequences,’’ in Proc. Australas.Joint Conf. Artif. Intell., 2016, pp. 137–149. [15]	The authors used recurrent and convolutional network layers to construct an ANN model and by using one recurrent layer and two convolutional layers, they detect various malware.	This model achieves good accuracy for multiclass intrusion detection with both datasets. Also, FAR and the execution time of this scheme are low.	This scheme should be further evaluated on the other imbalanced datasets,in which some of their attack classes have much fewer data records than others, to verify the detection rate of the minority class security attacks.

Khan et. al.[1] in this paper gives the idea of self-education figuring out how to prepare the profound neural network for network intrusion detection. In the proposed strategy, a Self Taught based Transfer Learning(DST-TL) is used to remove the highlights from the NSL-KDD dataset a relapse-related pre-prepared network is utilized. The limit of this system cannot classify the different types of attacks as deep neural networks are not used. Bui et. al.[2] in this paper provides the hybrid model of the K-Means clustering algorithm and Shrink AutoEncoder(SAE) to lessen the limitations in handling the datasets. In the proposed method, the hybrid model of the K-Means clustering algorithm and Shrink AutoEncoder is used to detect the anomalies occurring in the network. With the help of this method, datasets can also be handled properly. The limit of the system is that it failed to extend these works by using better clustering algorithms and other metrics to find a suitable number of clusters in the data.

Giacinto et. al. [3] in this paper provide a strategy to use Intrusion Detection Systems (IDS) and pattern recognition to increase the level of security on computer networks. The authors later discuss and evaluate the effectiveness of the IDS on the security of the network. Utilizing IDS and example acknowledgment ways to deal with network intrusion detection dependent on the combination of numerous classifiers. Specifically, the author centers around Modular Multiple Classifier engineering plans where every module in the design can identify intrusions against the administrations offered by the secured network. The limit of this study presents an extensive report on how IDS and pattern recognition can be used to provide higher levels of network security. The paper includes a descriptive report on how the authors used methods like machine learning in their implementation and design and also compared different training sets on the data to conclude which model is most effective. Although this paper is detailed, our methodology of achieving a system that can differentiate between good and bad networks containing viruses and malware includes methods and techniques using One-Class Classification and Auto Encoders. Bui et. al. [4]do a detailed examination and investigation of different AI procedures have been completed to discover the reason for issues related to different AI strategies in identifying intrusive exercises.AI methods have been examined and looked at as far as their recognition ability for distinguishing the different classes of assaults. Limits related to every classification of them are additionally talked about. Different data mining apparatuses for AI have additionally been remembered for the paper. The limit of the existing literature is described which is based on similar techniques with most of the popular datasets as on date to generalize our observations. All the techniques have not been implemented to evaluate the performance to ensure that results are reproducible

Ieracitano et. al. [5] statistical examination and autoencoder (AE) driven insightful intrusion detection (IDS) framework is acquainted with recognizing and alleviating the dangers of programmers growing much more complex and risky malware assaults that make intrusion detection a troublesome assignment. Initially, the NSLKDD dataset is cleaned from anomalies and the min-max standardization procedure is utilized to scale data inside the range 0 and 1. Subsequently, the one-hot-encoding is applied to change over symbolic (or categorized) features into numeric quantities. At that point, the 38 numeric attributes are investigated statistically to choose the most associated features. At last, shallow (MLP, L-SVM, Q-SVM, LDA, QDA) and deep (AE, LSTM) networks are created to quantify the detection execution both in parallel and multi-classification situations. The limit of the advancement of more precise deep architectures that can oversee continuous real-time data streams like NSL-KDD examples to recognize malicious attacks progressively can build its proficiency. Also, to misuse long-term learning, quicker choice models along with decreased computational complexity for constant needs to execute in this system. Chahal et.al. [6] talk about machine learning-based IDS for anomaly detection. Intrusion detection systems (IDS) serve an important role in detecting intrusions. This paper introduces a hybrid strategy that combines K-Means and Adaptive SVM, concluding that the combined results are superior to the individual results of K-Means and Adaptive SVM. Furthermore, when compared to other methodologies, this algorithm is quite accurate.The hybrid technology correctly distinguishes between normal and attack data, and this approach is 99.54 percent more accurate than solo techniques. As a result, employing this approach in real-time results in an extremely high detection rate of assaults. Furthermore, this method is easy and effective, especially when it comes to lowering the false-positive ratio and increasing the false negative ratio.

Yedukondalu et.al. [7] To identify intrusion rates, the suggested application uses the SVM (Support Vector Machine) and ANN (Artificial Neural Networks) algorithms. Each algorithm is used to determine if the data being requested is allowed or includes any irregularities. While the IDS examines the requested data, if it detects any malicious material, the request is dropped. These techniques employed feature selection algorithms based on correlation and Chi-Squared to decrease the dataset by removing unnecessary data. The preprocessed dataset is trained and evaluated with the models to generate notable findings, which improves prediction accuracy. The experiment was conducted using the NSL KDD dataset. Finally, the SVM algorithm obtained a 48 percent accuracy, while the ANN method achieved a 97 percent accuracy. On this dataset, the ANN model performs better than the SVM. Xuan et.al. [8] IDS is a good way to cope with the ever-changing nature of network attacks. In this research, we present a two-stage IDS based on machine learning models RF and SVM optimized with the CFS method, and we tested it on NSL-KDD benchmark datasets, comparing it to the RF and SVM models. The following is a summary of the key findings: (1) Our two-stage IDS outperformed RF and SVM, increasing Precision by 4.31 percent, Recall by 3.39 percent, and F1-measure by 5.56 percent to 11.08 percent; (2) the feature selection algorithm CFS, which we used in this paper, improved accuracy by 1.50 percent while reducing the time by 8.07 percent; and (3) our approach reduced Test Set prediction time by 93.84 percent compared to SVM.

Aljohani et.al. [9] The suggested system uses Neural Network and Support Vector Machine (SVM) models for intrusion detection to avoid security risks in a Local Area Network (LAN). The KDD99 dataset is used to test the suggested method. The KDD99 is an anomaly-based detection benchmark. This method detects assaults quickly and effectively. A comparison of the SVM and Neural Network models' performance is carried out. In terms of classification accuracy, the findings demonstrate that the Neural Network outperformed all SVM kernel models. The SVM linear kernel outperforms the SVM Gaussian kernel by a small margin, and the SVM polynomial kernel by a large margin. Elbahadır et.al. [10] An intrusion detection system (IDS) is modeled in this work to assure WSN security. Because signature, misuse, and anomaly-based intrusion detection approaches are insufficient to offer security on their own, a hybrid model is presented in which these methods are combined. Anomaly criteria were created for attack detection, and the BayesNet, J48, and Random Forest machine learning algorithms were employed to categorize normal and anomalous traffic in the hybrid model. The findings revealed that the generated model has a high level of accuracy and a low percentage of false alarms.

Zargar et.al. [11] This paper describes the developing creative, effective, efficient, and comprehensive prevention, detection, and response mechanisms that address the DDoS flooding problem before, during, and after an actual attack. The paper provides the knowledge required about the attacks as this project works on the prevention of attacks. It stimulates our research into understanding how the attacks occur which in turn helps with solutions to prevent them. Balamurugan et.al. [12] The proposed system in this paper is directed on intrusion detection systems and it uses cloudlet controller, trust authority, and virtual machine management in cloud environments. The proposed classifier in this paper effectively detects the attackers which are experimentally proved by comparing with existing classification models which provide a very good insight.

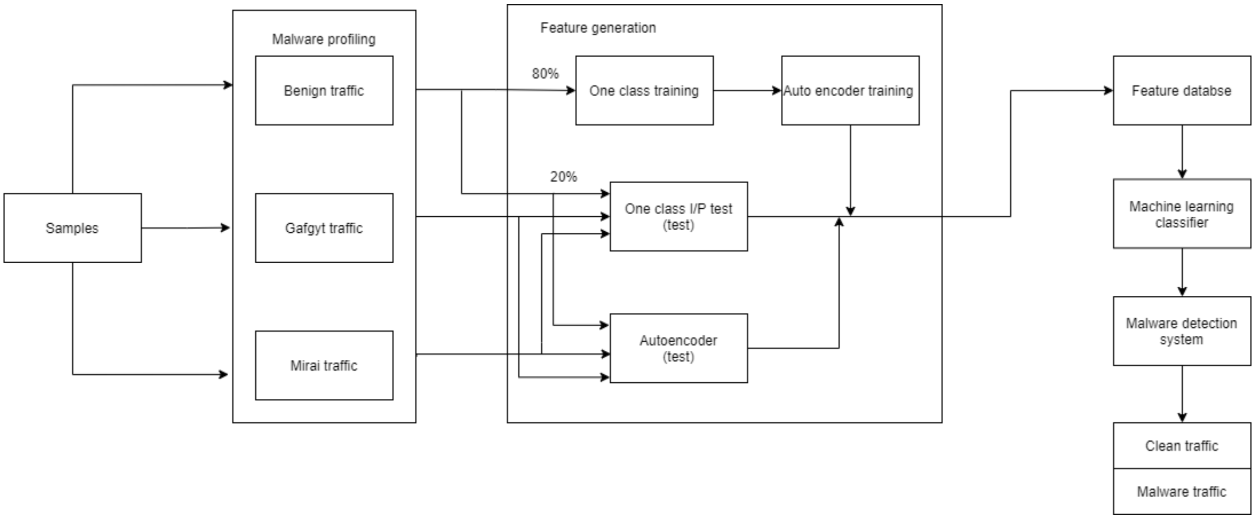
Yan et.al. [13] intends to answer some of the research questions specified in and random forest classifiers, which are shallow. The authors indicated that their approach could reduce the SVM's training and testing times in both binary and multiclass classifications and improve the prediction accuracy of the SVM. They used ISCX 2012 dataset and achieved ten features from it by using an auto-encoder and feeding it to the SVM for its training. The authors indicated the benefits of their method regarding metrics like Kappa statistics, detection rate, accuracy, and FPR. Al-Qatf et.al. [14] This paper proposes a scheme that uses the self-taught learning framework for the learning of features and reduction of dimension. This IDS model benefits from a sparse auto-encoder for unsupervised reconstructing a new feature representation. The authors indicated that their approach could reduce the SVM's training and testing times in both binary and multiclass classificationsand improve the prediction accuracy of the SVM.

Kolosnjaji et.al. [15] The authors used recurrent and convolutional network layers to construct an ANN model and by using one recurrent layer and two convolutional layers, they detect various malware. This model achieves good accuracy for multiclass intrusion detection with both datasets. Also, FAR and the execution time of this scheme are low.

METHODOLOGY

ONE CLASS SVM: One-class SVM is an unsupervised algorithm that learns a decision work for curiosity identification: ordering new information as comparative or diverse to the preparation set. One-class classification algorithms are often used for binary classification tasks with a severely skewed class distribution. These techniques are used to fit on the input examples from the huge class within the training dataset, then evaluated on the remaining test dataset. Albeit not designed for these types of problems, one-class classification methods are frequently successful for unbalanced classification datasets with no or few instances of the minority class, or datasets with no cohesive structure to distinguish the classes that a supervised algorithm would learn. The SVM algorithm, which was originally designed for classification tasks, is frequently employed for one-class classification.

AUTOENCODERS: An autoencoder neural network is a type of unsupervised machine learning technique that uses backpropagation to adjust the target values to the inputs. Autoencoders have a habit of condensing the dimensions of our inputs into a more compact representation. If the first data is required, the condensed data will be used to recreate it. The purpose of an autoencoder is to train the model to ignore signal noise in order to discover a symbol for a set of knowledge, generally to reduce dimensions. They work by condensing the input into a latent-space description and then reconstructing the outcome from there. We'll visualize our findings and results as part of the analysis once both of these models have been implemented and trained, and we'll try and compare the trained models to see which one gives an exact result.



There are three different types of datasets available as input: innocuous traffic (40,395 records), Mirai traffic (652,100 records), and Gafgyt traffic (316,650 records). Benign traffic is clean data flow, whereas Gafgyt and Mirai traffic both contain harmful data, resulting in malware traffic. Each record in the dataset has 115 attributes that were generated by the dataset's publishers using the raw characteristics of network traffic. Because both Gafgyt and Mirai traffic is generated from attack activity, they are combined to create the malicious data (968,750 records).

For training the model, 80 percent of the benign traffic data is fed into the one-class classifier. This means that the model was trained using 32,316 data, and the model's performance was evaluated with (40,395 – 32,316) + (652,100 + 316,650) = 976,829 records.

This is also further given as input for the training of the autoencoder model. The rest 20% of the benign traffic data is sent to the testing phase of both the one-class classifier and autoencoder model. This is also used as an input for the autoencoder model's training. The remaining 20% of the benign traffic data is supplied to the one-class classifier and autoencoder models for testing.

The Gafgyt and Mirai traffic data, which contains harmful data, are combined with 20% of the benign traffic data to see if the one-class classifier and autoencoder models can help sort traffic into clean and malware traffic.

Data is pre-processed in order to increase the quality of the datasets involved. Because there are various datasets to consider, functions are built to ensure that they are loaded and entered into the needed model.

Getting examples of cases with good traffic which displays benign behavior is simpler and easier to get in comparison with malware traffic that displays malicious behavior. This can be due to the fact that obtaining malware traffic can come at a cost or in some cases, impractical like the days with no attacks thus containing only good traffic. Other explanations regarding this could be on the basis of privacy, law, and ethics. But, despite all this, it can be easily convinced that corrupted traffic data can be used first to build models like the two-class classifier. But it cannot be assured that all scenarios involving bad traffic data can be replicated. This issue can be addressed using the unsupervised one-class classifier methodology approach here.

So, in this project, the training model is created only using benign instances, and this trained model is then implemented to detect any unknown/new cases of traffic using machine-learning and other statistical methods. If the data targeted shows considerable diversion according to predetermined calculations, it will be labeled as out-of-class. Thus, one-class classifiers that may belong to different families are examined here under the criterion of performance. One-Class Support Vector Machine from the conventional ML family and Autoencoder from the deep learning family are the two one-class classifiers and their related families illustrated here. In this case, we're investigating how benign occurrences vary from corrupted occurrences in terms of structure. The Autoencoder model is built on the basis of this base composition.

Through training sets comprising only examples of that class, one-class classifiers may distinguish instances of a given class from all other instances. This is a benign class that is being explored. There are various types of one-class classifiers that examine opposing examples to refine the categorization limit.

Other purposes that a one-class classifier can fulfill are in cases of binary and imbalanced classification datasets. In the scenarios where binary classification is to be done but the majority of the dataset overpowers the minority, the training set is modeled based on the majority category of data, before being jointly evaluated in the testing phase. Because one-class classifiers are not built for cases involving unbalanced categorization datasets where the minority may be non-existent or not evident enough, supervised machine learning approaches will need to be used.

Even though it's designed for binary classification, the support vector machine, or SVM, the technique may be used for one-class classification. Before using one-class classifiers, scaled and conventional support vector machines can be applied to the dataset in circumstances of unbalanced classification. For one-class classification, the approach aids in sizing up the density of the majority class and classifies outliers on either side of the probability density. A one-class support vector machine is a variant of the SVM classifier.

Autoencoder neural network is an unsupervised ML algorithm that incorporates backpropagation by keeping the required values the same as the inputted values. This helps decrease the size of the input into reduced representation. The original data can be reassembled from the compressed data. The primary goal of autoencoders is to learn the representation of a dataset, mainly for the simplification of dimensionality, by training it to overlook noise data. The input is compressed and put into a latent-space representation and the autoencoders then reconstruct the output out of this.

The autoencoder comprises an encoder, code, and decoder. The encoder helps compress the input into the latent space. The input image is encoded here as the compressed representation with a simplified dimensionality. Thus, this compressed version will be distorted in comparison to the original. The code part shows the input that was compressed which is loaded into the decoder. Finally, the decoder helps decode the compressed image back into the original version with the original dimension, resulting in a lossy reconstruction of the original from the latent space.

The features extracted through the above algorithms are then registered and stored in a database called the feature database. The machine learning classifiers included in the project are explained above. Thus, malware detection can be directly implemented resulting in the allowance of the benign traffic to be sent and received through the device and removal of any transmission back and forth of malicious data found due to the presence of Gafgyt and Mirai datasets

RESULTS AND DISCUSSION

Dataset

1. The dataset comprises three types of web traffic data, benign traffic containing 40, 395 records, Mirai traffic containing 652,100 records, and Gafgyt traffic containing 316,650 records.
2. Each record contains 115 features that were created by the distributors of the dataset utilizing crude credits of network traffic. As both Gafgyt and Mirai traffic delivered by the attack movement, the two information sources were joined to develop the general arrangement of noxious information (968,750 records) for this activity.
3. It tends to contend that a definitive objective of a model in that setting is to permit benign traffic to pass to and from the device, dispose of transmission, and gathering of malevolent data, a one-class classifier prepared utilizing benign data would satisfactorily suit the reason.
4. We utilized 80% of the benign records to assemble our model. This implies that 32, 316 records were utilized to prepare the model and (40,395 – 32,316) + (652,100 + 316,650) = 976,829 records were utilized to assess the presentation of the model.

Test Beds

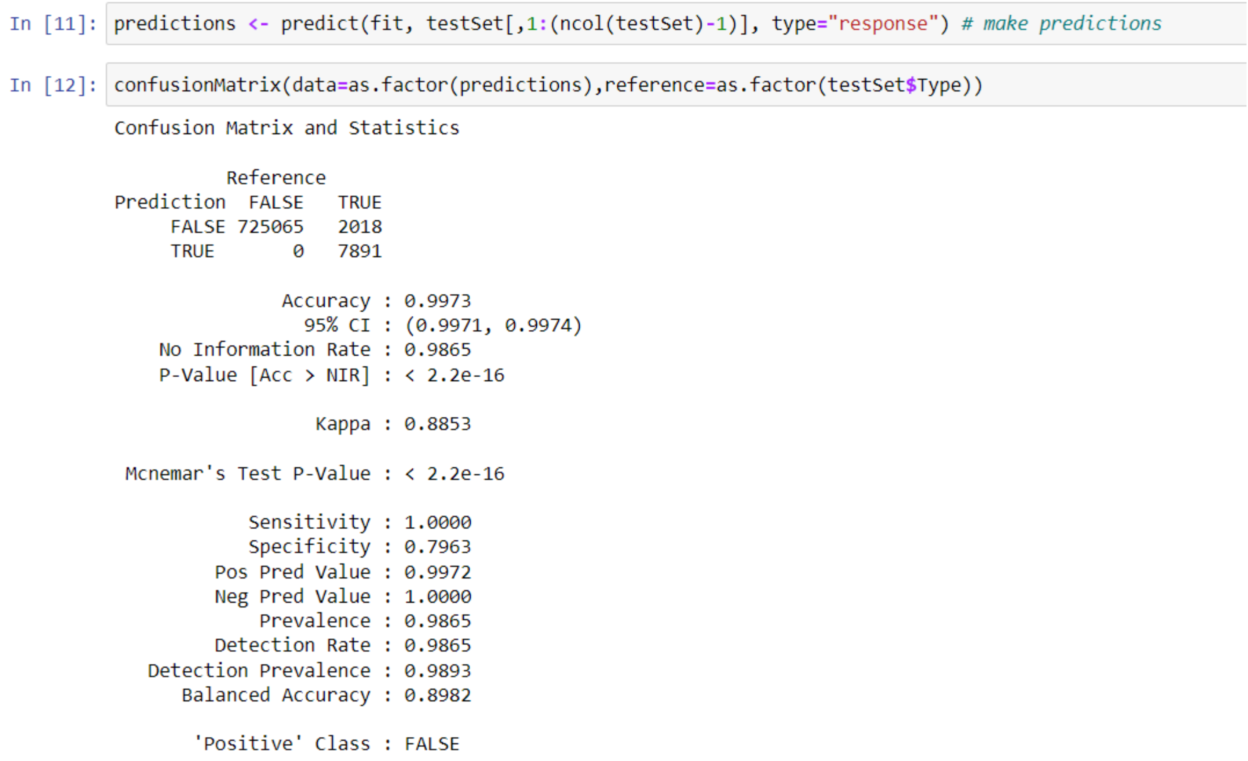
In this project, we have used Jupyter Notebook for the compilation of the work. The testing and training part is written in R studio and the visualization is done on Anaconda Navigator using Python. Jupyter notebook is used to compile the Python as well as the R code to get the desired graph and outputs. Not only that, in the code part, many libraries were used like NumPy, Pandas, matplotlib, Keras, scikit-learn, and TensorFlow.

Expected Result

The result which is expected is 100% accuracy in detecting a bad traffic network with some virus, malware to the system, or any other type and normal network using machine learning and deep learning. The output is determined using the confusion matrix. The model also predicts false data which helps in the prevention of installation of software during risks to minimize the cost.

ONE CLASS SVM MODEL EVALUATION

Model Evaluation

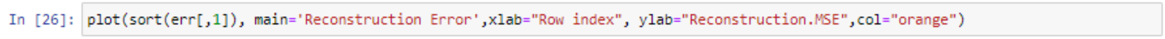


```
In [11]: predictions <- predict(fit, testSet[,1:(ncol(testSet)-1)], type="response") # make predictions
```

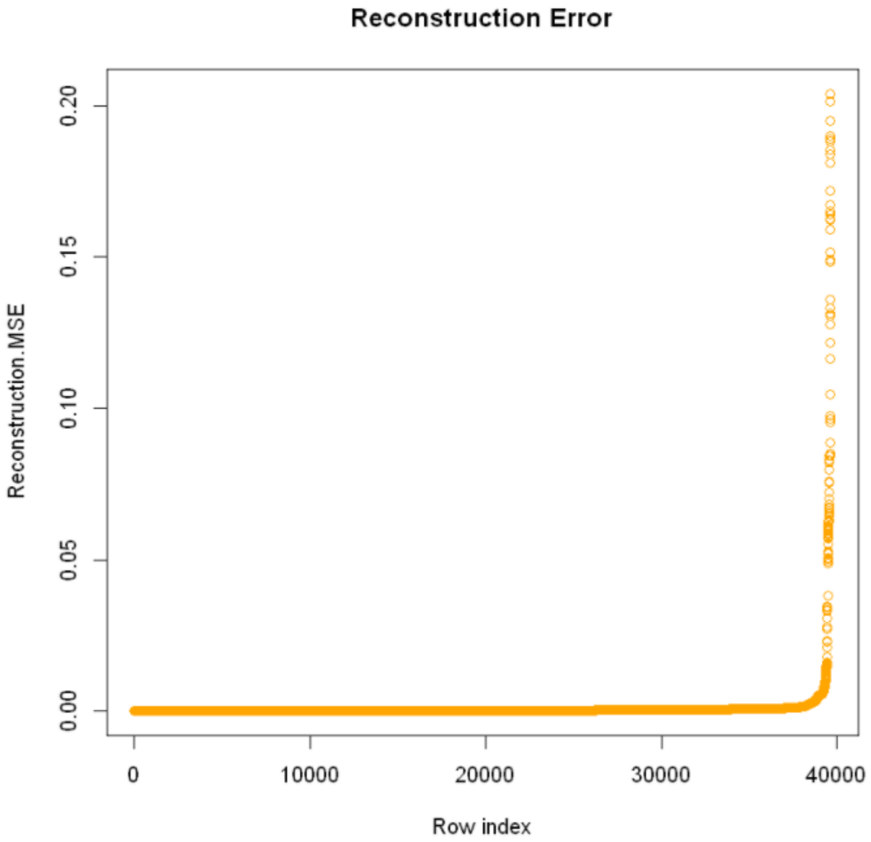
```
In [12]: confusionMatrix(data=as.factor(predictions),reference=as.factor(testSet$Type))
```

```
Confusion Matrix and Statistics

          Reference
Prediction  FALSE   TRUE
     FALSE 725065   2018
     TRUE       0   7891

               Accuracy : 0.9973
                 95% CI : (0.9971, 0.9974)
    No Information Rate : 0.9865
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8853

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 1.0000
            Specificity : 0.7963
         Pos Pred Value : 0.9972
         Neg Pred Value : 1.0000
             Prevalence : 0.9865
         Detection Rate : 0.9865
   Detection Prevalence : 0.9893
      Balanced Accuracy : 0.8982

       'Positive' Class : FALSE
```

AUTOENCODER IMPLEMENTATION



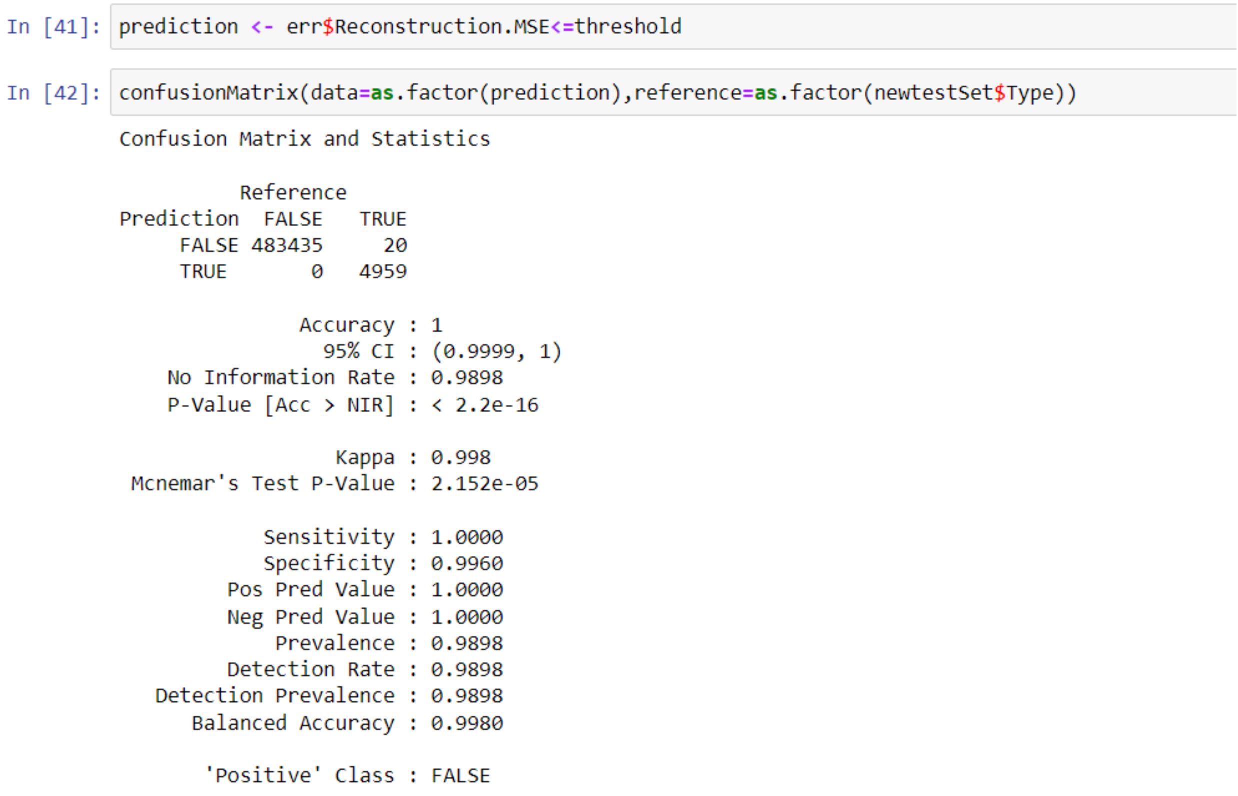
```
In [26]: plot(sort(err[,1]), main='Reconstruction Error',xlab="Row index", ylab="Reconstruction.MSE",col="orange")
```



reconstruction.MSE plot helps us to see where the model reconstructs the original records. Model was previously not able to learn patterns for those anomaly points. Reconstruction.MSE infer that at approximately 0.02 MSE, the data starts to become sparse, and therefore, including data above a 0.02 MSE will not be useful. Thus, we introduce a threshold value of 0.02 in the AutoEncoder model.

Defining the threshold based on Reconstruction. We randomly take 50% records of test instances due to the memory constraints and summarise (average) the results to approximate the accuracy if our computer has low computation resources. Then, we converted data to h20 compatible and calculated MSE across observations.

AUTOENCODER MODEL EVALUATION



```
In [41]: prediction <- err$Reconstruction.MSE<=threshold
```

```
In [42]: confusionMatrix(data=as.factor(prediction),reference=as.factor(newtestSet$Type))
```

```
Confusion Matrix and Statistics

          Reference
Prediction  FALSE   TRUE
     FALSE 483435     20
     TRUE       0   4959

               Accuracy : 1
                 95% CI : (0.9999, 1)
    No Information Rate : 0.9898
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.998
 Mcnemar's Test P-Value : 2.152e-05

            Sensitivity : 1.0000
            Specificity : 0.9960
         Pos Pred Value : 1.0000
         Neg Pred Value : 1.0000
             Prevalence : 0.9898
         Detection Rate : 0.9898
   Detection Prevalence : 0.9898
      Balanced Accuracy : 0.9980

       'Positive' Class : FALSE
```

Once the data has been fit into the model, the testset and the MSE threshold can be used to predict and show the performance of the AutoEncoder model. We can analyze the confusion matrix and infer the accuracy is now 100% with zero false negatives. This is a great improvement from the One-Class SVM model implemented above.

The 37 attack types mentioned in the dataset can be clustered into four general attack types as listed below:

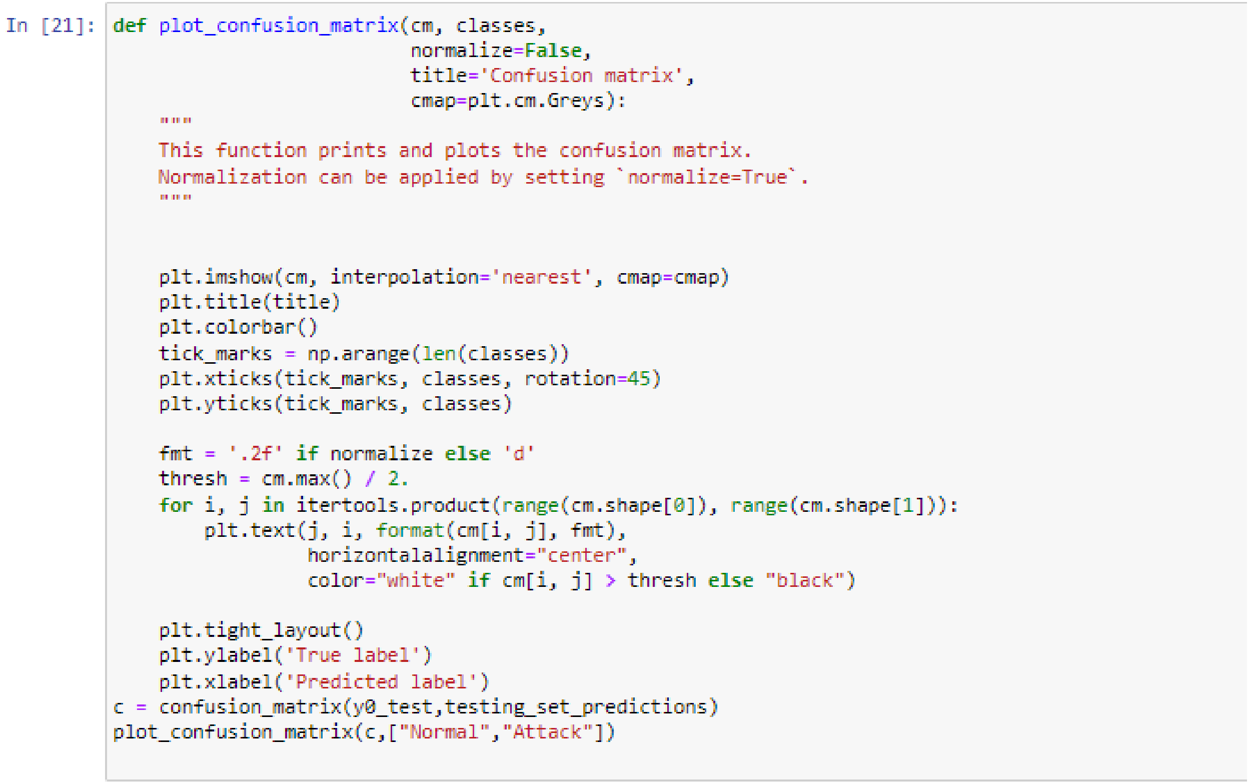
- Denial of service attacks
- Remote to Local attacks
- User to Root
- Probe attacks

Our model will perform binary classification of the data to two classes indicating whether the traffic is normal or is a malicious attack, however, we will use the four attack types to analyze the results and calculate performance metrics for each general attack type. The next section replaces the current outcome field with a Class field that has one of the following values:

- Normal
- Dos
- R2L
- U2R
- Probe

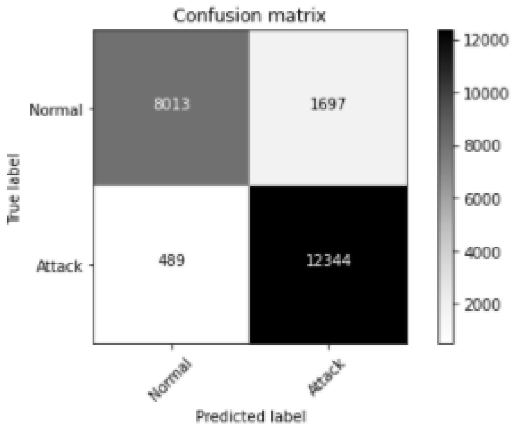
PLOTTING CONFUSION MATRIX AND VIOLIN PLOTS

Plotting confusion matrix

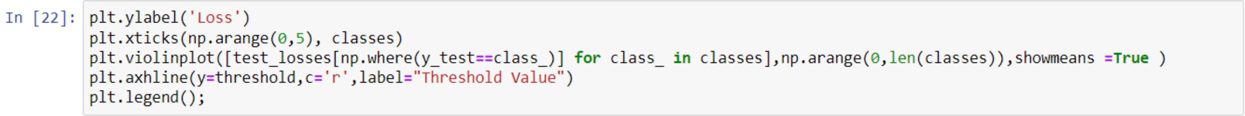


In [21]:

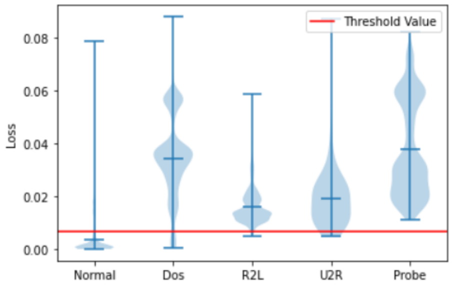
```
def plot_confusion_matrix(cm, classes,
                          normalize=False,
                          title='Confusion matrix',
                          cmap=plt.cm.Greys):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """


    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')
c = confusion_matrix(y0_test,testing_set_predictions)
plot_confusion_matrix(c,["Normal","Attack"])
```



Violin plot



```
In [22]: plt.ylabel('Loss')
         plt.xticks(np.arange(0,5), classes)
         plt.violinplot([test_losses[np.where(y_test==class_)] for class_ in classes],np.arange(0,len(classes)),showmeans =True )
         plt.axhline(y=threshold,c='r',label="Threshold Value")
         plt.legend();
```



The violin plot shows the distribution of reconstruction loss values for the testing dataset values and clearly infer that the loss values of attacks are mostly higher than the threshold value, the opposite is true for the normal dataset.

CONCLUSION

The one class classifiers we used in the project were used for both training and testing set with which it will segregate malicious and good traffic which comes through the network. To increase the efficiency of the one-class classifiers we implemented a model using autoencoders which use deep learning neural networks. By the use of both algorithms we increased the efficiency of the project and we also attempted to overcome the problems that exist in the datasets, namely the class imbalance issue and the data being unrealistic, by avoiding the attacks data during training, the model was trained only using normal traffic, so it was not affected by the class imbalance of the dataset. Another strength of this approach is its simplicity, it consists of only a single hidden layer of 8 neurons making it very easy to train and especially suitable for online learning. During the evaluation, we avoided human manipulation of the threshold in order to achieve reproducible results without human interference.

REFERENCES

1. Intrusion detection using deep sparse auto-encoder and self-taught learning, Qureshi by A. S., Khan, A., Shamim, N., & Durad, M. H. (2019) Neural Computing and Applications (2019)
2. A Clustering-based Shrink AutoEncoder for Detecting Anomalies in Intrusion Detection Systems by Bui, T. C., Hoang, M., & Nguyen, Q. U. (2019, October) 2019 11th International Conference on Knowledge and Systems Engineering (KSE). IEEE, 2019.
3. A Modular Multiple Classifier System for the Detection of Intrusions in Computer Networks Giorgio Giacinto, Fabio Roli, Luca Didaci Department of Electrical and Electronic Engineering, University of Cagliari, Italy

4. A Clustering-based Shrink AutoEncoder for Detecting Anomalies in Intrusion Detection Systems Bui, T. C., Hoang, M., & Nguyen, Q. U. (2019, October) In 2019 11th International Conference on Knowledge and Systems Engineering (KSE) (pp. 1-5). IEEE.
5. A novel statistical analysis and autoencoder-driven intelligent intrusion detection approach Ieracitano, C., Adeel, A., Morabito, F. C., & Hussain, A. (2020). Neurocomputing, 387, 51-62.
6. J. K. Chahal, V. Gandhi, P. Kaushal, K. R. Ramkumar, A. Kaur and S. Mittal, "KAS-IDS: A Machine Learning based Intrusion Detection System," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), 2021, pp. 90-95, doi: 10.1109/ISPCC53510.2021.9609402.
7. G. Yedukondalu, G. H. Bindu, J. Pavan, G. Venkatesh and A. SaiTeja, "Intrusion Detection System Framework Using Machine Learning," 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA), 2021, pp. 1224-1230, doi: 10.1109/ICIRCA51532.2021.9544717.
8. D. Xuan, H. Hu, B. Wang and B. Liu, "Intrusion Detection System Based on RF-SVM Model Optimized with Feature Selection," 2021 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), 2021, pp. 1-5, doi: 10.1109/CCCI52664.2021.9583206.
9. A. Aljohani and A. Bushnag, "An Intrusion Detection System Model in a Local Area Network using Different Machine Learning Classifiers," *2021 11th International Conference on Advanced Computer Information Technologies (ACIT)*, 2021, pp. 483-488, doi: 10.1109/ACIT52158.2021.9548421.
10. H. Elbahadır and E. Erdem, "Modeling Intrusion Detection System Using Machine Learning Algorithms in Wireless Sensor Networks," 2021 6th International Conference on Computer Science and Engineering (UBMK), 2021, pp. 401-406, doi: 10.1109/UBMK52708.2021.9558928.
11. S. T. Zargar, J. Joshi, and D. Tipper, ‘‘A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks,’’ IEEECommun. Surveys Tuts., vol. 15, no. 4, pp. 2046–2069, Mar. 2013.
12. V. Balamurugan and R. Saravanan, ‘‘Enhanced intrusion detection and prevention system on cloud environment using hybrid classification andOTS generation,’’ Cluster Comput., vol. 22, pp. 1–13, Nov. 2017.
13. J. Yan, D. Jin, C. W. Lee, and P. Liu, ‘‘A comparative study of off-line deep learning based network intrusion detection,’’ in Proc. 10th Int. Conf.Ubiquitous Future Netw. (ICUFN), Jul. 2018, pp. 299–304
14. M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, ‘‘Deep learning approach combining sparse autoencoder with SVM for network intrusion detection,’’ IEEE Access, vol. 6, pp. 52843–52856, 2018.
15. B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, ‘‘Deep learning for classification of malware system call sequences,’’ in Proc. Australas.Joint Conf. Artif. Intell., 2016, pp. 137–149.

Plagiarism Checker X Originality Report

Similarity Found: 11%

Date: Monday, November 29, 2021
Statistics: 693 words Plagiarized / 6142 Total words
Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

BCI 3002: Disaster Recovery and Business Continuity Management (DRBCM) Slot: A1+TA1 Research Paper TITLE: PREVENTION OF ATTACKS USING ONE CLASS CLASSIFICATION AND AUTOENCODERS 29th November 2021 Team Leader: Rohan Allen 18BCI0247 Team Members: Rakshith Sachdev 18BCI0109 Harshita Pundir 18BCI0192 ABSTRACT In this paper, we represent how one-class classifiers are prepared to utilize generous information to recognize ordinary and dangerous traffic redirected to an endpoint device.

In this venture, the framework is prepared to utilize unsupervised / one-class-based demonstrating approaches by which the framework would comprehend the issues that we would confront day by day, and the preparation will be useful for what's to come. After the preparation of the framework, the framework can be utilized in reality to confront ongoing, new difficulties and by gaining from the past experiences it can develop as indicated by the users' issues, weaknesses, dangers, and conditions KEYWORDS Autoencoders, feature generation, machine Learning, malware detection, malware profiling, network protection, one-class classification, INTRODUCTION With today’s day and age rapidly becoming digital, the network and endpoint devices become a target for attacks and exploitation; thus, the systems have long been associated with issues related to security.

Therefore, making systems secure and safe is of extreme importance. As per the 2020 Unit 42 Threat Report, practically all traffic is decoded, implying that the majority of classified and indiIntrusion detection using deep sparse auto-encoder and self-taught learning, Qureshi by A. S., visual user information in the network is highly powerless against cyber attacks.

Network security is utilized to delay unintentional harm which can be done to the network's private information, its users, or its devices. The main aim of network security is to secure the network running and for every single authentic client. The objective of this project is to use machine learning to teach our system to distinguish between malicious network traffic, which could contain a virus or malware, and regular network data.

The model has also been taught to detect and avoid fake data and the deployment of any specific software while there is a threat of an attack, which results in dramatically reducing the financial stress on an organization and prevents tarnishing their reputation. The major goal is to offer the most accurate findings possible by utilizing methods such as unsupervised learning/one-class-based modeling, thereby lowering processing time substantially. In this model, we are using one-class classifications and autoencoders so that the system can detect bad traffic more accurately.

With the help of this model, we can predict false data, and therefore, we can prevent the installation of software at the time of any risks to decrease the cost. LITERATURE SURVEY TITLE, AUTHOR, AND JOURNAL OBJECTIVE METHODOLOGY USED LIMITATION Khan, A., Shamim, N., & Durad, M. H. (2019) Neural Computing and Applications (2019).

[1] This paper gives the idea of self-education figuring out how to prepare the deep neural network for network intrusion detection. In the proposed strategy, a Self Taught based Transfer Learning(DST-TL) is used to remove the highlights from the NSL-KDD dataset a relapse related pre-prepared network is utilized This system cannot classify the different types of attacks as deep neural networks are not used.

A Clustering-based Shrink AutoEncoder for Detecting Anomalies in Intrusion Detection Systems by Bui, T. C., Hoang, M., & Nguyen, Q. U. (2019, October) 2019 11th International Conference on Knowledge and Systems Engineering (KSE). IEEE, 2019. [2] This paper provides the hybrid model of the K-Means clustering algorithm and Shrink AutoEncoder(SAE) to lessen the limitations in handling the datasets.

In the proposed method, the hybrid model of the K-Means clustering algorithm and Shrink AutoEncoder is used to detect the anomalies occurring in the network. With the help of this method, datasets can also be handled properly It failed to extend these works by using better clustering algorithms and other metrics to find a suitable number of clusters in the data A Modular Multiple Classifier System for the Detection of Intrusions in Computer Networks Giorgio Giacinto, Fabio Roli, Luca Didaci Department of Electrical and Electronic Engineering, University of Cagliari, Italy.

[3] This paper provides a strategy to use Intrusion Detection Systems (IDS) and pattern recognition to increase the level of security on computer networks. The authors later discuss and evaluate the effectiveness of the IDS on the security of the network. Utilizing IDS and example acknowledgment ways to deal with network intrusion detection dependent on the combination of numerous classifiers.

Specifically, the author centers around Modular Multiple Classifier engineering plans where every module in the design can identify intrusions against the administrations offered by the secured network. This study presents an extensive report on how IDS and pattern recognition can be used to provide higher levels of network security. The paper includes a descriptive report on how the authors used methods like machine learning in their implementation and design and also compared different training sets on the data to conclude which model is most effective.

Although this paper is detailed, our methodology of achieving a system that can differentiate between good and bad networks containing viruses and malware includes methods and techniques using One-Class Classification and Auto Encoders. A Clustering-based Shrink AutoEncoder for Detecting Anomalies in Intrusion Detection Systems Bui, T. C., Hoang, M., & Nguyen, Q. U.

(2019, October) In 2019 11th International Conference on Knowledge and Systems Engineering (KSE) (pp. 1-5). IEEE. [4] A detailed examination and investigation of different AI procedures have been completed to discover the reason for issues related to different AI strategies in identifying intrusive exercises. AI methods have been examined and looked at as far as their recognition ability for distinguishing the different classes of assaults.

Limits related to every classification of them are additionally talked about. Different data mining apparatuses for AI have additionally been remembered for the paper Existing literature is described which is based on similar techniques with most of the popular datasets as on date to generalize our observations.

All the techniques have not been implemented to evaluate the performance to ensure that results are reproducible A novel statistical analysis and autoencoder-driven intelligent intrusion detection approach. Ieracitano, C., Adeel, A., Morabito, F. C., & Hussain, A. (2020). Neurocomputing, 387, 51-62. [5] Statistical examination and autoencoder (AE) driven insightful intrusion detection (IDS) framework is acquainted with recognizing and alleviate the dangers of programmers growing much more complex and risky malware assaults that make intrusion detection a troublesome assignment Initially, the NSLKDD dataset is cleaned from anomalies and the min-max standardization procedure is utilized to scale data inside the range 0 and 1.

Subsequently, the one-hot-encoding is applied to change over symbolic (or categorized) features into numeric quantities. At that point, the 38 numeric attributes are investigated statistically to choose the most associated features. At last, shallow (MLP, L-SVM, Q-SVM, LDA, QDA) and deep (AE, LSTM) networks are created to quantify the detection execution both in parallel and multi-classification situations.

The advancement of more precise deep architectures that can oversee continuous real-time data streams like NSL-KDD examples to recognize malicious attacks progressively can build its proficiency. Also, to misuse long-term learning, quicker choice models along with decreased computational complexity for constant needs to execute in this system. J. K.

Chahal, V. Gandhi, P. Kaushal, K. R. Ramkumar, A. Kaur and S. Mittal, "KAS-IDS: A Machine Learning based Intrusion Detection System," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), 2021.[6] Regrettably, the majority of commercial IDSs are based on abuse and are only designed to detect known threats.

These require frequent signature updates and have a limited capacity for detecting new assaults. As a result, this study proposes an anomaly-based IDS as a viable solution to this challenge. KAS-IDS, or K-Means and Adaptive SVM based Intrusion Detection System, is the approach proposed in this study.

K-Means were used in the first stage to create data clusters, and adaptive SVM was used in the second step to classify the data. As part of the current technique, the data is split into two categories: normal and abnormal data, and correct findings are obtained using the NSL-KDD dataset, which can also be used for real-time traffic analysis.

Apart from that, the intelligent agents of clustering and classification algorithms can improve its performance in real-time traffic analysis. G. Yedukondalu, G. H. Bindu, J. Pavan, G. Venkatesh and A. SaiTeja, "Intrusion Detection System Framework Using Machine Learning," 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA), 2021[7]. The main purpose of this project is to compare and assess the effectiveness of neural network models on a data set.

To identify intrusion rates, the suggested application uses the SVM (Support Vector Machine) and ANN (Artificial Neural Networks) algorithms. Each algorithm is used to determine if the data being requested is allowed or includes any irregularities. These techniques employed feature selection algorithms based on correlation and

Chi-Squared to decrease the dataset by removing unnecessary data.

Another dataset with a larger number of characteristics might be used to improve this research. Because ANN provides greater accuracy but slower calculations, we can apply alternative efficient algorithms in the future that may provide greater accuracy while also computing faster, allowing it to be employed in real-time applications. D. Xuan, H. Hu, B. Wang and B.

Liu, "Intrusion Detection System Based on RF-SVM Model Optimized with Feature Selection," 2021 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), 2021.[8] The goal of this research is to improve network intrusion detection using machine learning. The two-stage IDS suggested in this research is based on machine learning algorithms RF and SVM that are tuned with the Feature Ranking CFS.

In this research, they present a two-stage IDS based on machine learning models RF and SVM tuned with the CFS method, and they tested it on NSL-KDD benchmark datasets, contrasting it to the RF and SVM modeling. We plan to develop the IDS in the future with the goal of increasing the detection precision of low-frequency assaults, which is a common difficulty in IDS.

Furthermore, the capacity of an IDS to identify unknown forms of threats is taken into account. We'd want to increase the accuracy of detecting unknown forms of assaults. A. Aljohani and A. Bushnag, "An Intrusion Detection System Model in a Local Area Network using Different Machine Learning Classifiers," 2021 11th International Conference on Advanced Computer Information Technologies (ACIT), 2021.[9] The article presents a security control for an IDS that is used to identify known and unknown attacks in order to avoid security concerns in LANs.

The suggested system uses Neural Network and Support Vector Machine (SVM) models for intrusion detection to avoid security risks in a Local Area Network (LAN). The KDD99 dataset is used to test the suggested method. The KDD99 is an anomaly-based detection benchmark. This method detects assaults quickly and effectively. In future research, the KDD99 dataset may be used to classify threat categories to see which machine learning classifier performs better. H. Elbahadir and E.

Erdem, "Modeling Intrusion Detection System Using Machine Learning Algorithms in Wireless Sensor Networks," 2021 6th International Conference on Computer Science and Engineering (UBMK), 2021.[10] An intrusion detection system (IDS) is modeled in this work to assure WSN security. Because signature, misuse, and anomaly-based intrusion detection approaches are insufficient to offer security on their own, a hybrid model is presented in which these methods are combined.

Anomaly criteria were created for attack detection, and the BayesNet, J48, and Random Forest machine learning algorithms were employed to categorize normal and anomalous traffic in the hybrid model. Application of alternative efficient algorithms in the future that may provide greater accuracy while also computing faster, allowing it to be employed in real-time applications. S. T. Zargar, J. Joshi, and D. Tipper, ''A survey of defense mechanisms against distributed denial of service (DDoS) ?ooding attacks,'' IEEECommun.

Surveys Tuts., vol. 15, no. 4, pp. 2046–2069, Mar. 2013. [11] This paper describes the developing creative, effective, efficient, and comprehensive prevention, detection, and response mechanisms that address the DDoS flooding problem before, during and after an actual attack. The paper provides the knowledge required about the attacks as this project works on prevention of attacks.

It stimulates our research into understanding how the attacks occur which in turn help with solutions to prevent them. Existing literature is described which is based on similar techniques with most of the popular datasets as on date to generalize our observations. All the techniques have not been implemented to evaluate the performance to ensure that results are reproducible. V. Balamurugan and R.

Saravanan, ''Enhanced intrusion detection and prevention system on cloud environment using hybrid classi?cation andOTS generation,'' Cluster Comput., vol. 22, pp. 1–13, Nov. 2017. [12] The proposed system in this paper is directed on intrusion detection systems and it uses cloudlet controller, trust authority and virtual machine management in cloud environments.

The proposed classifier in this paper effectively detects the attackers which are experimentally proved by comparing with existing classification models which provides a very good in sight. Application of alternative efficient algorithms in the future that may provide greater accuracy while also computing faster, allowing it to be employed in real-time applications. J. Yan, D. Jin, C. W. Lee, and P. Liu, ''A comparative study of off-line deep learning based network intrusion detection,'' in Proc. 10th Int. Conf.Ubiquitous Future Netw. (ICUFN), Jul. 2018, pp. 299–304. [13] It intends to answer some of the research questions speci?ed in and random forest classi?ers, which are shallow.

The authors indicated that their approach could reduce the SVM's training and testing times in both binary and multiclass classi?cations and improves the prediction accuracy of the SVM. They used ISCX 2012 dataset and achieved ten features from it by using an auto-encoder and fed it to the SVM for its training. Theauthors indicated the bene?ts of their method regarding met-rics like Kappa statistics, detection rate, accuracy, and FPR.

It is only able to conduct the binary classi?cation and cannot handle the multiclass attack traf?cs M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, ''Deep learning approach combining sparse autoencoder with SVM for network intrusion detection,'' IEEE Access, vol. 6, pp. 52843–52856, 2018. [14] This paper proposes a scheme that uses the self-taught learning framework for the learning of features and reduction of dimension.

This IDS model bene?ts from a sparse auto-encoder for unsupervised reconstructing a new feature representation. The authors indicated that their approach could reduce the SVM's training and testing times in both binary and multiclass classi?cationsand improves the prediction accuracy of the SVM. The ef?ciency of this approach in binary and multiclass classi?cations is evaluated against the naive random forest classi?ers, which are shallow.

B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, ''Deep learning for classi?cation of malware system call sequences,'' in Proc. Australas.Joint Conf. Artif. Intell., 2016, pp. 137–149. [15] The authors used recurrent and convolutional network layers to construct an ANN model and by using one recurrent layer and two convolutional layers, they detect various malware. This model achieves good accuracy for multiclass intrusion detection with both datasets. Also, FAR and the execution time of this scheme are low.

This scheme should be further evaluated on the other imbalanced datasets,in which some of their attack classes have much fewer data records than others, to verify the detection rate of the minority class security attacks. Khan et. al.[1] in this paper gives the idea of self-education figuring out how to prepare the profound neural network for network intrusion detection.

In the proposed strategy, a Self Taught based Transfer Learning(DST-TL) is used to remove the highlights from the NSL-KDD dataset a relapse-related pre-prepared network is utilized. The limit of this system cannot classify the different types of attacks as deep neural networks are not used. Bui et. al.[2] in this paper provides the hybrid model of the K-Means clustering algorithm and Shrink AutoEncoder(SAE) to lessen the limitations in handling the datasets.

In the proposed method, the hybrid model of the K-Means clustering algorithm and

Shrink AutoEncoder is used to detect the anomalies occurring in the network. With the help of this method, datasets can also be handled properly. The limit of the system is that it failed to extend these works by using better clustering algorithms and other metrics to find a suitable number of clusters in the data. Giacinto et. al.

[3] in this paper provide a strategy to use Intrusion Detection Systems (IDS) and pattern recognition to increase the level of security on computer networks. The authors later discuss and evaluate the effectiveness of the IDS on the security of the network. Utilizing IDS and example acknowledgment ways to deal with network intrusion detection dependent on the combination of numerous classifiers.

Specifically, the author centers around Modular Multiple Classifier engineering plans where every module in the design can identify intrusions against the administrations offered by the secured network. The limit of this study presents an extensive report on how IDS and pattern recognition can be used to provide higher levels of network security.

The paper includes a descriptive report on how the authors used methods like machine learning in their implementation and design and also compared different training sets on the data to conclude which model is most effective. Although this paper is detailed, our methodology of achieving a system that can differentiate between good and bad networks containing viruses and malware includes methods and techniques using One-Class Classification and Auto Encoders. Bui et. al.

[4]do a detailed examination and investigation of different AI procedures have been completed to discover the reason for issues related to different AI strategies in identifying intrusive exercises.AI methods have been examined and looked at as far as their recognition ability for distinguishing the different classes of assaults. Limits related to every classification of them are additionally talked about.

Different data mining apparatuses for AI have additionally been remembered for the paper. The limit of the existing literature is described which is based on similar techniques with most of the popular datasets as on date to generalize our observations. All the techniques have not been implemented to evaluate the performance to ensure that results are reproducible Ieracitano et. al.

[5] statistical examination and autoencoder (AE) driven insightful intrusion detection (IDS) framework is acquainted with recognizing and alleviating the dangers of programmers growing much more complex and risky malware assaults that make intrusion detection a troublesome assignment. Initially, the NSLKDD dataset is cleaned

from anomalies and the min-max standardization procedure is utilized to scale data inside the range 0 and 1.

Subsequently, the one-hot-encoding is applied to change over symbolic (or categorized) features into numeric quantities. At that point, the 38 numeric attributes are investigated statistically to choose the most associated features. At last, shallow (MLP, L-SVM, Q-SVM, LDA, QDA) and deep (AE, LSTM) networks are created to quantify the detection execution both in parallel and multi-classification situations.

The limit of the advancement of more precise deep architectures that can oversee continuous real-time data streams like NSL-KDD examples to recognize malicious attacks progressively can build its proficiency. Also, to misuse long-term learning, quicker choice models along with decreased computational complexity for constant needs to execute in this system. Chahal et.al.

[6] talk about machine learning-based IDS for anomaly detection. Intrusion detection systems (IDS) serve an important role in detecting intrusions. This paper introduces a hybrid strategy that combines K-Means and Adaptive SVM, concluding that the combined results are superior to the individual results of K-Means and Adaptive SVM. Furthermore, when compared to other methodologies, this algorithm is quite accurate.The hybrid technology correctly distinguishes between normal and attack data, and this approach is 99.54 percent more accurate than solo techniques.

As a result, employing this approach in real-time results in an extremely high detection rate of assaults. Furthermore, this method is easy and effective, especially when it comes to lowering the false-positive ratio and increasing the false negative ratio. Yedukondalu et.al. [7] To identify intrusion rates, the suggested application uses the SVM (Support Vector Machine) and ANN (Artificial Neural Networks) algorithms.

Each algorithm is used to determine if the data being requested is allowed or includes any irregularities. While the IDS examines the requested data, if it detects any malicious material, the request is dropped. These techniques employed feature selection algorithms based on correlation and Chi-Squared to decrease the dataset by removing unnecessary data.

The preprocessed dataset is trained and evaluated with the models to generate notable findings, which improves prediction accuracy. The experiment was conducted using the NSL KDD dataset. Finally, the SVM algorithm obtained a 48 percent accuracy, while the ANN method achieved a 97 percent accuracy. On this dataset, the ANN model performs better than the SVM. Xuan et.al. [8] IDS is a good way to cope with the ever-changing nature of network attacks.

In this research, we present a two-stage IDS based on machine learning models RF and SVM optimized with the CFS method, and we tested it on NSL-KDD benchmark datasets, comparing it to the RF and SVM models. The following is a summary of the key findings: (1) Our two-stage IDS outperformed RF and SVM, increasing Precision by 4.31 percent, Recall by 3.39 percent, and F1-measure by 5.56 percent to 11.08 percent; (2) the feature selection algorithm CFS, which we used in this paper, improved accuracy by 1.50 percent while reducing the time by 8.07 percent; and (3) our approach reduced Test Set prediction time by 93.84 percent compared to SVM. Aljohani et.al.

[9] The suggested system uses Neural Network and Support Vector Machine (SVM) models for intrusion detection to avoid security risks in a Local Area Network (LAN). The KDD99 dataset is used to test the suggested method. The KDD99 is an anomaly-based detection benchmark. This method detects assaults quickly and effectively. A comparison of the SVM and Neural Network models' performance is carried out.

In terms of classification accuracy, the findings demonstrate that the Neural Network outperformed all SVM kernel models. The SVM linear kernel outperforms the SVM Gaussian kernel by a small margin, and the SVM polynomial kernel by a large margin. Elbahadir et.al. [10] An intrusion detection system (IDS) is modeled in this work to assure WSN security.

Because signature, misuse, and anomaly-based intrusion detection approaches are insufficient to offer security on their own, a hybrid model is presented in which these methods are combined. Anomaly criteria were created for attack detection, and the BayesNet, J48, and Random Forest machine learning algorithms were employed to categorize normal and anomalous traffic in the hybrid model. The findings revealed that the generated model has a high level of accuracy and a low percentage of false alarms. Zargar et.al.

[11] This paper describes the developing creative, effective, efficient, and comprehensive prevention, detection, and response mechanisms that address the DDoS flooding problem before, during and after an actual attack. The paper provides the knowledge required about the attacks as this project works on prevention of attacks. It stimulates our research into understanding how the attacks occur which in turn help with solutions to prevent them. Balamurugan et.al.

[12] The proposed system in this paper is directed on intrusion detection systems and it uses cloudlet controller, trust authority and virtual machine management in cloud

environments. The proposed classifier in this paper effectively detects the attackers which are experimentally proved by comparing with existing classification models which provides a very good in sight. Yan et.al. [13] It intends to answer some of the research questions speci?ed in and random forest classi?ers, which are shallow.

The authors indicated that their approach could reduce the SVM’s training and testing times in both binary and multiclass classi?cations and improves the prediction accuracy of the SVM. They used ISCX 2012 dataset and achieved ten features from it by using an auto-encoder and fed it to the SVM for its training. Theauthors indicated the bene?ts of their method regarding met-rics like Kappa statistics, detection rate, accuracy, and FPR. Al-Qatf et.al.

[14] This paper proposes a scheme that uses the self-taught learning framework for the learning of features and reduction of dimension. This IDS model bene?ts from a sparse auto-encoder for unsupervised reconstructing a new feature representation. The authors indicated that their approach could reduce the SVM’s training and testing times in both binary and multiclass classi?cationsand improves the prediction accuracy of the SVM. Kolosnjaji et.al.

[15] The authors used recurrent and convolutional network layers to construct an ANN model and by using one recurrent layer and two convolutional layers, they detect various malware. This model achieves good accuracy for multiclass intrusion detection with both datasets. Also, FAR and the execution time of this scheme are low. METHODOLOGY ONE CLASS SVM: One-class SVM is an unsupervised algorithm that learns a decision work for curiosity identification: ordering new information as comparative or diverse to the preparation set. One-class classification algorithms are often used for binary classification tasks with a severely skewed class distribution.

These techniques are used to fit on the input examples from the huge class within the training dataset, then evaluated on the remaining test dataset. Albeit not designed for these types of problems, one-class classification methods are frequently successful for unbalanced classification datasets with no or few instances of the minority class, or datasets with no cohesive structure to distinguish the classes that a supervised algorithm would learn.

The SVM algorithm, which was originally designed for classification tasks, is frequently employed for one-class classification. AUTOENCODERS: An autoencoder neural network is a type of unsupervised machine learning technique that uses backpropagation to adjust the target values to the inputs. Autoencoders have a habit of condensing the dimensions of our inputs into a more compact representation.

If the first data is required, the condensed data will be used to recreate it. The purpose of an autoencoder is to train the model to ignore signal noise in order to discover a symbol for a set of knowledge, generally to reduce dimensions. They work by condensing the input into a latent-space description and then reconstructing the outcome from there.

We'll visualize our findings and results as part of the analysis once both of these models have been implemented and trained, and we'll try and compare the trained models to see which one gives an exact result. There are three different types of datasets available as input: innocuous traffic (40,395 records), Mirai traffic (652,100 records), and Gafgyt traffic (316,650 records). Benign traffic is clean data flow, whereas Gafgyt and Mirai traffic both contain harmful data, resulting in malware traffic.

Each record in the dataset has 115 attributes that were generated by the dataset's publishers using the raw characteristics of network traffic. Because both Gafgyt and Mirai traffic are generated from attack activity, they are combined to create the malicious data (968,750 records). For training the model, 80 percent of the benign traffic data is fed into the one-class classifier.

This means that the model was trained using 32,316 data, and the model's performance was evaluated with (40,395 – 32,316) + (652,100 + 316,650) = 976,829 records. This is also further given as input for the training of the autoencoder model. The rest 20% of the benign traffic data is sent to the testing phase of both the one-class classifier and autoencoder model.

This is also used as an input for the autoencoder model's training. The remaining 20% of the benign traffic data is supplied to the one-class classifier and autoencoder models for testing. The Gafgyt and Mirai traffic data, which contains harmful data, are combined with 20% of the benign traffic data to see if the one-class classifier and autoencoder models can help sort traffic into clean and malware traffic. Data is pre-processed in order to increase the quality of the datasets involved.

Because there are various datasets to consider, functions are built to ensure that they are loaded and entered into the needed model. Getting examples of cases with good traffic which displays benign behavior is simpler and easier to get in comparison with malware traffic that displays malicious behavior.

This can be due to the fact that obtaining malware traffic can come at a cost or in some cases, impractical like the days with no attacks thus containing only good traffic. Other explanations regarding this could be on the basis of privacy, law, and ethics. But, despite all this, it can be easily convinced that corrupted traffic data can be used first to build models like the two-class classifier.

But it cannot be assured that all scenarios involving bad traffic data can be replicated. This issue can be addressed using the unsupervised one-class classifier methodology approach here. So, in this project, the training model is created only using benign instances, and this trained model is then implemented to detect any unknown/new cases of traffic using machine-learning and other statistical methods.

If the data targeted shows considerable diversion according to predetermined calculations, it will be labeled as out-of-class. Thus, one-class classifiers that may belong to different families are examined here under the criterion of performance. One-Class Support Vector Machine from the conventional ML family and Autoencoder from the deep learning family are the two one-class classifiers and their related families illustrated here. In this case, we're investigating how benign occurrences vary from corrupted occurrences in terms of structure.

The Autoencoder model is built on the basis of this base composition. Through training sets comprising only examples of that class, one-class classifiers may distinguish instances of a given class from all other instances. This is a benign class that is being explored. There are various types of one-class classifiers that examine opposing examples to refine the categorization limit.

Other purposes that a one-class classifier can fulfill are in cases of binary and imbalanced classification datasets. In the scenarios where binary classification is to be done but the majority of the dataset overpowers the minority, the training set is modeled based on the majority category of data, before being jointly evaluated in the testing phase.

Because one-class classifiers are not built for cases involving unbalanced categorization datasets where the minority may be non-existent or not evident enough, supervised machine learning approaches will need to be used. Even though it's designed for binary classification, the support vector machine, or SVM, the technique may be used for one-class classification.

Before using one-class classifiers, scaled and conventional support vector machines can be applied to the dataset in circumstances of unbalanced classification. For one-class classification, the approach aids in sizing up the density of the majority class and classifies outliers on either side of the probability density. A one-class support vector machine is a variant of the SVM classifier.

Autoencoder neural network is an unsupervised ML algorithm that incorporates backpropagation by keeping the required values the same as the inputted values. This helps decrease the size of the input into reduced representation. The original data can be reassembled from the compressed data. The primary goal of autoencoders is to learn the representation of a dataset, mainly for the simplification of dimensionality, by training it to overlook noise data.

The input is compressed and put into a latent-space representation and the autoencoders then reconstruct the output out of this. The autoencoder comprises an encoder, code, and decoder. The encoder helps compress the input into the latent space. The input image is encoded here as the compressed representation with a simplified dimensionality.

Thus, this compressed version will be distorted in comparison to the original. The code part shows the input that was compressed which is loaded into the decoder. Finally, the decoder helps decode the compressed image back into the original version with the original dimension, resulting in a lossy reconstruction of the original from the latent space. The features extracted through the above algorithms are then registered and stored in a database called the feature database.

The machine learning classifiers included in the project are explained above. Thus, malware detection can be directly implemented resulting in the allowance of the benign traffic to be sent and received through the device and removal of any transmission back and forth of malicious data found due to the presence of Gafgyt and Mirai datasets RESULTS AND DISCUSSION Dataset 1.

The dataset comprises three types of web traffic data, benign traffic containing 40, 395 records, Mirai traffic containing 652,100 records, and Gafgyt traffic containing 316,650 records. 2. Each record contains 115 features that were created by the distributors of the dataset utilizing crude credits of network traffic. As both Gafgyt and Mirai traffic delivered by the attack movement, the two information sources were joined to develop the general arrangement of noxious information (968,750 records) for this activity. 3.

It tends to contend that a definitive objective of a model in that setting is to permit benign traffic to pass to and from the device, dispose of transmission, and gathering of malevolent data, a one-class classifier prepared utilizing benign data would satisfactorily suit the reason. 4. We utilized 80% of the benign records to assemble our model.

This implies that 32, 316 records were utilized to prepare the model and (40,395 – 32,316) + (652,100 + 316,650) = 976,829 records were utilized to assess the presentation of the model. Test Beds In this project, we have used Jupyter Notebook for the compilation of the work. The testing and training part is written in R studio and the visualization is done on Anaconda Navigator using Python.

Jupyter notebook is used to compile the Python as well as the R code to get the desired graph and outputs. Not only that, in the code part, many libraries were used like NumPy, Pandas, matplotlib, Keras, scikit-learn, and TensorFlow. Expected Result The result which is expected is 100% accuracy in detecting a bad traffic network with some virus, malware to the system, or any other type and normal network using machine learning and deep learning. The output is determined using the confusion matrix.

The model also predicts false data which helps in the prevention of installation of software during risks to minimize the cost. ONE CLASS SVM MODEL EVALUATION Model Evaluation AUTOENCODER IMPLEMENTATION reconstruction.MSE plot helps us to see where the model reconstructs the original records. Model was previously not able to learn patterns for those anomaly points. Reconstruction.MSE infer that at approximately 0.02 MSE, the data starts to become sparse, and therefore, including data above a 0.02 MSE will not be useful. Thus, we introduce a threshold value of 0.02 in the AutoEncoder model. Defining the threshold based on Reconstruction.

We randomly take 50% records of test instances due to the memory constraints and summarise (average) the results to approximate the accuracy if our computer has low computation resources. Then, we converted data to h20 compatible and calculated MSE across observations. AUTOENCODER MODEL EVALUATION Once the data has been fit into the model, the testset and the MSE threshold can be used to predict and show the performance of the AutoEncoder model. We can analyze the confusion matrix and infer the accuracy is now 100% with zero false negatives.

This is a great improvement from the One-Class SVM model implemented above. The 37 attack types mentioned in the dataset can be clustered into four general attack types as listed below: ? Denial of service attacks ? Remote to Local attacks ? User to Root ? Probe attacks Our model will perform binary classification of the data to two classes indicating whether the traffic is normal or is a malicious attack, however, we will use the four attack types to analyze the results and calculate performance metrics for each general attack type.

The next section replaces the current outcome field with a Class field that has one of the following values: ? Normal ? Dos ? R2L ? U2R ? Probe PLOTTING CONFUSION MATRIX

AND VIOLIN PLOTS Plotting confusion matrix Violin plot The violin plot shows the distribution of reconstruction loss values for the testing dataset values and clearly infer that the loss values of attacks are mostly higher than the threshold value, the opposite is true for the normal dataset.

CONCLUSION The one class classifiers we used in the project were used for both training and testing set with which it will segregate malicious and good traffic which comes through the network. To increase the efficiency of the one-class classifiers we implemented a model using autoencoders which use deep learning neural networks. By the use of both algorithms we increased the efficiency of the project and we also attempted to overcome the problems that exist in the datasets, namely the class imbalance issue and the data being unrealistic, by avoiding the attacks data during training, the model was trained only using normal traffic, so it was not affected by the class imbalance of the dataset.

Another strength of this approach is its simplicity, it consists of only a single hidden layer of 8 neurons making it very easy to train and especially suitable for online learning. During the evaluation, we avoided human manipulation of the threshold in order to achieve reproducible results without human interference.

INTERNET SOURCES:

<1% - takecareinternational.org › fundraising › green-india
<1% - machinelearningmastery.com › process-for-working
<1% - www.researchgate.net › publication › 340329101
1% - www.researchgate.net › publication › 347064242
<1% - kmeducationhub.de › international-conference-on
<1% - shabbirhasan.com › files › papers
<1% - user.engineering.uiowa.edu › ~ie_155 › Lecture
<1% - www.researchgate.net › publication › 221093988_A
<1% - scholar.google.com › citations
<1% - cybersecurity.springeropen.com › articles › 10
1% - www.sciencedirect.com › science › article
<1% - ieeexplore.ieee.org › xpl › conhome
<1% - cmilab.org › publications
<1% - hackernoon.com › what-is-one-hot-encoding-why-and
<1% - www.fmsreliability.com › event › ispcc-2021-6th
<1% - people.eecs.ku.edu › ~hossein › 710
<1% - www.briefmenow.org › comptia › which-security
<1% - www.mirlabs.org › ijcisim › regular_papers_2020

<1% - www.researchgate.net › publication › 273213434_K-SVM

<1% - digitalcommons.latech.edu › cgi › viewcontent

<1% - www.sciencedirect.com › traffic-analysis

<1% - toc.proceedings.com › 60511webtoc

<1% - socialsciences.cornell.edu › research-incubation

<1% - www.ijettjournal.org › volume-3 › issue-4

<1% - www.researchgate.net › publication › 318279418_An

<1% - jis-eurasipjournals.springeropen.com › articles › 10

<1% - www.researchgate.net › publication › 355026200_An

<1% - www.wunu.edu.ua › en › 10216-11th-international

<1% - www.ai.rug.nl › ~mwiering › GROUP

<1% - ijana.in › papers › V6I4-10

<1% - www.sciencedirect.com › intrusion-detection-system

<1% - www.ifsecglobal.com › uncategorized › signature

<1% - aijsh.com › wp-content › uploads

<1% - dl.acm.org › doi › 10

<1% - www.ijert.org › analysis-of-denial-of-services-dos

<1% - www.researchgate.net › figure › KEY-TERMS-APPEARED

<1% - www.researchgate.net › publication › 328766118_HyINT

<1% - www.hindawi.com › journals › scn

<1% - www.researchgate.net › publication › 221533942

<1% - www.researchgate.net › publication › 349531131_A

<1% - www.scribd.com › document › 430024136

<1% - www.ncbi.nlm.nih.gov › pmc › articles

<1% - downloads.hindawi.com › journals › mpe

<1% - www.checkpoint.com › cyber-hub › network-security

<1% - www.researchgate.net › publication › 12413257_New

<1% - serokell.io › blog › classification-algorithms

<1% - machinelearningmastery.com › one-class

<1% - blockgeni.com › classification-algorithms-for

<1% - cse.iitkgp.ac.in › ~sudeshna › courses

<1% - iq.opengenus.org › types-of-autoencoder

1% - github.com › Putting-data-into-action › IoT-Security

1% - www.irjet.net › archives › V8

<1% - www.mdpi.com › 2079/9292/10-21 › 2696

<1% - www.irjet.net › archives › V5

<1% - link.springer.com › chapter › 10

<1% - course.ccs.neu.edu › cs5100f11 › resources

<1% - ufldl.stanford.edu › tutorial › unsupervised

<1% - towardsdatascience.com › deep-inside-autoencoders

<1% - deepai.org › publication › autoencoders
<1% - www.researchgate.net › publication › 337811525
<1% - machinelearningmastery.com › lstm-autoencoders
<1% - www.academia.edu › 30688608 › K_Means_Cluster_based
<1% - pt.scribd.com › document › 464972254