iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** XI **Month of publication:** November 2023

DOI: https://doi.org/10.22214/ijraset.2023.56541

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

Enhanced Network Intrusion Detection System Using Machine Learning

Vinay Singh[1], Raju Singh[2]
[1, 2]*Assistant Professor, Department of Computer Science and Engineering, MPEC Kanpur , India*

Abstract: ***In order to forecast anomalies more correctly, a moderately excellent network detection system for intrusions requires a high rate of detection and a relatively low false alarm rate. Because older datasets cannot capture the design of a set of recent attacks, modeling on the basis of these datasets lacks generalizability. We discuss numerous models before concluding with the one that performs best utilizing various types of evaluation measures. Along with modeling, a detailed data analysis on the properties of the set of data itself is performed for a more complete picture employing our comprehension of a correlation variance, and similar aspects. Furthermore, hypothetical considerations for potential network intrusion detection systems are presented, including advice on prospective modeling and dataset production.***
Keywords: Deep learning; anomaly detection; intrusion detection system; network security.

I. INTRODUCTION

With the sheer quantity of assets dispersed and devices in use, cyberspace has gotten immensely complex. It is extremely difficult to precisely define the variables required to forecast whether an attack has happened within a system context. Historically, intrusion detection systems for networks (NIDS) were employed as a type of network forensics, detecting suspicious events and alerting the appropriate authorities. These examples indicate that even little breaches can be extremely destructive, necessitating the requirement for a sophisticated detection system in such a grave situation. Because of the constantly rising connection of assets and gadgets in general, we may see an even stronger growth in the future months. Corporations and households demand network forensics in order to comprehend flows that are causing breaches and, as a result, to prevent further attacks. Many older datasets used in the development of NIDS have proven inadequate for real-time scenarios. As a result, the cyber security research group at the Australian Centre for Cyber Security (ACCS) and other researchers in this domain around the world took on the challenge of producing a good enough dataset that would work effectively in real-time environments.. Using this dataset, we hope to develop a model for an NIDS that accurately predicts anomalous events, i.e., attacks, with a low misclassification error. Section 3 will delve deeper into the Dataset's construction, features, labels, and evaluated metrics. Before modeling, preprocessing is performed in a separate section. In Section 4, we will look into the models that have been used and their particular technical definitions in relation to the dataset as described in Section 3. Section 5 will go over the findings and the various types of metrics used for cross validation. In Section 6, we will propose alternative modeling methodologies and hypothetically create additional data sets NIDS/NIPS construction. Finally, we conclude the conversation with the findings from Section 7.

II. LITERATURE REVIEW

The authors of [3] executed an approach in which they used network forensic analysis to sniff packets at the edge of the network and then archived said transmissions in large volumes in the database created with SQL. The data set for packet-feature analysis, but their model is significantly weaker. The model's selected features for network that forensic analysis are insufficient for accurately predicting system attacks. As a result, compared to our simulation, it may leave gaps in the NIDS the system due to information loss. Furthermore, accessing large amounts of data from the MySQL database is extremely slow in comparison to our regionally recorded packets. Nour Moustafa and colleagues developed an internet-based intelligence about threats system for Industry 4.0. Their method primarily employs Bro-IDS for collecting packets and is focused on product-oriented IoT devices and machinery used by industries and consumers. As the name implies, "Industry 4.0" refers to new sectors and startup companies in the IT sector that aim to bring about revolutionary changes in technology. As a result, they intended to migrate their system to the cloud. They attempted to build an interconnected architecture using multiple edge devices and internet bridges. These are, by definition, IoT devices. With their system's carefully planned architecture, they use MHMM (mixed hidden Markov models) and other complicated models for threat detection.

©IJRASET: All Rights are Reserved

The main disadvantage of their system is the large number of public-end network equipment and a lack of expertise in specifying the device-specific firewalls and which could give rise to complicated MITM (man in the middle) attempts by experienced hackers/crackers. Debugging such complex models takes a long time because the cloud system includes a large amount of middleware. Any data breach or hardware damage will be difficult to resolve on time in comparison to our end-to-end architecture, which is far less complicated and has a lot far more flexibility, with the further advantage of fewer displayed accessible devices in the framework itself [4]. The authors of [5] developed a multiple-layer feed forward artificial neural network (ANN). The model has been assessed using different tests on the modeling architecture, with impressive results. Nevertheless, their concluded model is computationally complex, so it takes a significant amount of energy and time to train and then run it, which is inefficient for detecting attacks as soon as possible. The authors of proposed a BMM (beta mixture model) centered ADS (anomaly detection system), which is a likely representative model of a subset of the entire dataset, in [6]. In this case, the probability distribution can adapt to arbitrary variables with varying ranges. Despite being a flexible model, the pattern of distribution depends on some subset uniformity. As a result, it is unable to achieve greater rate of detection and false-positive rates. Preeti et al. used a software detection system for intrusions involving different hardware components consisting of software that is connected to an internet connection and the cloud to implement the concept of "cloud hypervisor" for the security-based architecture in [7]. They deemed cloud security to be extremely important. As a result, they used the PSI-NetVisor a hypervisor layer to deploy the virtual machine contemplation (VMI) functionality for network monitoring and the associated process execution tracing. A behavior-based detection system for intrusions (BIDM) is also used to detect attacks. The model, on the other hand, is resistant to linking devices at the end to the cloud. Despite the fact that based on the cloud security continues to operate and the same set of data is used as in our case, many devices that run programs have connections to an independently CSP and can be compromised if malware is introduced into any of the of the VMs, as opposed to our case in which the model runs independently of the hypervisor layer. They did not place a high value on modeling for attack prediction and instead concentrated their efforts on system architecture, failing to make the whole thing widely implementable. The authors of [8] focused on the issue of huge network of things volumes, unbalanced data sets, and other obsolete datasets for various NIDSs, resulting in a less effective system for anticipating attacks and network security loopholes. As a result, they built a robust model to deal with data imbalances and used a sparse classification with multiple classes approach known as Ramp-KSVCR (ramp K-support vector classification regression). For better generalizability, this support vector classification was primarily used to eliminate noise and other outliers from the dataset. However, our model has an advantage because, rather than eliminating anomalies we work with these individuals and avoid information loss due to the fact that these very anomalies may be extremely descriptive of event anomalies instead. The authors of proposed a collection model based on flow-level analysis theory in [9]. Their model selects observations for IDS (intrusion detection system) using basic random sampling and ARM (association rule mining). In the case of an antecedent event, the ARM method is based on the association of features pertaining to the rate of a preceding event. They primarily focus on modeling based on associated measures because they believe it is the most effective way to model an IDS. However, they place too much emphasis on the relevance of flows rather than their predictive ability. They use a very small subset of features to reduce processing time, but lose predictive performance and generalizability as a result. The authors of [10] addressed the issues of hacking and captures within a network of interconnected devices. To reduce such risks, they proposed an innovative GAA (geometric area analysis).

The method focuses on calculating the appropriate portions of the features determined from the beta the mixture model parameters, and the network samples are assumed to be (that is, independent and identically distributed). The authors of [11] focused primarily on recognizing zero-day attacks on systems.

They attempted to address this by focusing on lowering the false-positive rate with ADS (anomaly detection system).They used finite Dirichlet mixture models while maintaining a normal dataset to avoid under fitting in the model. This is difficult to achieve on a large scale because cloud platforms have the nodes that cannot be classified as centralized or distributed. As a result, high-speed data transfer among nodes may have an impact on ADS performance in regard to false-positive and detection rates. The authors of [12] used algorithms based on machine learning such as decision trees, association rule mining models, neural networks with artificial intelligence, and naive Bayes classifiers to address a bot network operation during the research they conducted on IoT (internet of things). They got the best results for the binary classification problem by using a decision tree, and they also ran their algorithms . They attempted to combine these models with IoT network forensics. They worked on these models primarily for removing networks of bots in IoT, which is right now a major issue in this age of technological interconnectivity. The authors of [13] used a combination of machine learning along with a deep learning strategy to solve intrusion detection with a heterogeneous dataset. Spark MLlib-based robust classical regression Anomaly detection using ML classifiers and a misuse attack using a convolution-auto encoder.

Furthermore, the authors of [14] worked on rapid identification of an invasion for those authors. Swift IDS was proposed, which used a light gradient enhancement device to cope with the massive. Data on traffic. Another study [15] focused on intrusion detection using decision trees. By classifying networks, tree algorithms and rule-based concepts can be used to secure the internet of things. Traffic can be classified as malicious or benign.

III. PROPOSED METHODOLOGY

We use eight different models in this study: neural networks, naive Bayes classifiers, SVM, decision trees, extreme gradient boosted trees, random forest classifiers, AdaBoost classifiers, and logistical regression models. When running cross-validation on each model, random contends were used to avoid overfitting.

A. Evaluation Metrics

Accuracy, DR (detection rate), FAR (false alarm rate), ROC-AUC score, precision, recall, and F1 score are the metrics used to evaluate model performance for both the training and testing splits, and their definitions are as follows. Accuracy is defined as the ratio of correctly classified information to all data [18-22].

Accuracy = TP + TN/ TP + TN + FP + FN.

False alarm rate (FAR) is the percentage of incorrect predictions made of anomalous events out of all predictions made.

FAR = 100 × FP/ (FP + TN) (2)

where FP represents the number of false positives and TN represents the number of true negatives. ROC-AUC score is the area under the receiver operating characteristic curve that calculates true and false positives at different classification thresholds [20]. Precision is the percentage of relevant predictions made out of all predictions.

'Relevant' here means the positively labelled entries.

Precision = TP/ (TP + FP) (3)

The Recall is the sensitivity of the model of outputting relevant predictions

Recall = TP/ (TP + TN) (4)

F1 score is the collative score generated via the results from precision and recall and is understood as a single-valued metric for the performance over specific class labels [20].

F1 = 2 × Precision × Recall /Precision + Recall.

IV. MODEL DEFINITIONS

A. Artificial Neural Network

The neural network has five hidden layers with the output as a softmax unit for two classes. The unit is mapped for each neuron mathematically defined in Equation (6).

softmax(xi) = e (xi)/ ∑j e (xi) (6)

Where x is a single-valued output from a list of multi-classed predictions.

B. The Decision Tree

We used a decision tree with an optimal depth of nine levels. The Gini method was used to split nodes, with a minimum of two samples split. We also make the classification weights proportional in reverse to the class With higher frequencies in the information being provided to make incorrectly identifying the least frequently occurring classes more difficult.

C. Random Forest Classifier

A random forest classifier with no maximum depth (all leaves expanded with no pruning) was implemented with 300 estimators with other hyperparameters similar to the earlier mentioned decision tree classifier.

D. AdaBoost Classifier

An AdaBoost classifier was used with its base estimator as a random forest of three estimators. This entire classifier had a total of 200 estimators. An algorithm called "SAMME. R" was used, which is a regularized version of the algorithm mentioned in [26] and is used since our base estimator produces continuous values in terms of label probabilities. It also converges faster than other algorithms.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

E. Logistic Regression

A logistic regression model was fitted using the non-linear conjugate gradient method as illustrated in [27] over maximum iterations of 300 with an L2 penalty on weights to prevent overfitting. We experimented with other kinds of penalties, but the L2 norm helps us penalize misclassification of anomalies.

V. METHODOLOGIES

For cross-validation, the hypotheses were trained across five stratified folds. The splits were generated at random using distinct seeds, and the characteristic importance's were calculated for the best executing model using the scores for F1 with respect to the separate weights. Early stopping was implemented based on the observed reduction in losses for ten values, and the best collections of weights for the neural network that was created were restored. All eight models were evaluated experimentally based on the bias-variance trade-off for training and testing accuracies, as well as accuracy, recollection, and F1 scores for each class in our label set.

VI. RESULTS

Due to the larger amount of data on normal events, we can see that all models perform very similarly in terms of classifying events as normal events. This is why an ideal model is critical for capturing attacks more frequently and reporting falsities less frequently.

VII. FUTURE SCOPE

This dataset does not account for the human component that makes up malicious strategies. As a result, we believe that for a better NIDS, an ensemble of models should focus on packet flows, modifications to asset inventory, data downloaded and/or uploaded, and other comparable events that can happen in corporate organizations. This hypothetical ensemble would effectively compensate for the lack of knowledge on human agency.

REFERENCES

[1] Alnaghes, M.S.; Gebali, F. A Survey on SomeCurrently Existing Intrusion Detection Systems for Mobile Ad Hoc Networks. In Proceedings of the 2nd International Conference on Electrical and Electronics Engineering, Clean Energy and Green Computing (EEECEGC2015), Konya, Turkey, 26–28 May 2015; pp. 12–18.

[2] Irwin, L. List of Data Breaches & Cyber Attacks in May 2020. In IT Governance UK Blog; 21 September 2020. Available online: https://www.itgovernance.co.uk/blog/list-of-data-breaches-cyber-attacks-may-2020 (accessed on 2 November 2020).

[3] Moustafa, N.; Slay, J. A network forensic scheme using correntropy-variation for attack detection. IFIP Adv. Inf. Commun. Technol. 2018, 532, 225–239. [CrossRef]

[4] Moustafa, N.; Adi, E.; Turnbull, B.; Hu, J. A New Threat Intelligence Scheme for Safeguarding Industry 4.0 Systems. IEEE Access 2018, 6, 32910–32924. [CrossRef]

[5] Al-Zewairi, M.; Almajali, S.; Awajan, A. Experimental evaluation of a multi-layer feedforward artificial neural network classifier for network intrusion detection system. In Proceedings of the 2017 International Conference on New Trends in Computing Sciences (ICTCS), Amman, Jordan, 11–13 October 2017; pp. 167–172.

[6] Moustafa, N.; Creech, G.; Slay, J. Anomaly detection system using beta mixture models and outlier detection. Adv. Intell. Syst. Comput. 2018, 710, 125–135. [CrossRef]

[7] Mishra, P.; Pilli, E.S.; Varadharajan, V.; Tupakula, U. PSI-NetVisor: Program semantic aware intrusion detection at network and hypervisor layer in cloud. J. Intell. Fuzzy Syst. 2017, 32, 2909–2921. [CrossRef]

[8] Hosseini Bamakan, S.M.; Wang, H.; Shi, Y. Ramp loss k-support vector classification-regression; a robust and sparse multiclass approach to the intrusion detection problem. Knowl.-Based Syst. 2017, 126, 113–126. [CrossRef]

[9] Moustafa, N.; Creech, G.; Slay, J. Flow aggregator module for analyzing network traffic. Adv. Intell. Syst. Comput. 2018, 710, 19–29. [CrossRef]

[10] Moustafa, N.; Slay, J.; Creech, G. Novel Geometric Area Analysis Technique for Anomaly Detection Using Trapezoidal Area Estimation on Large-Scale Networks. IEEE Trans. Big Data 2017, 5, 481–494. [CrossRef]

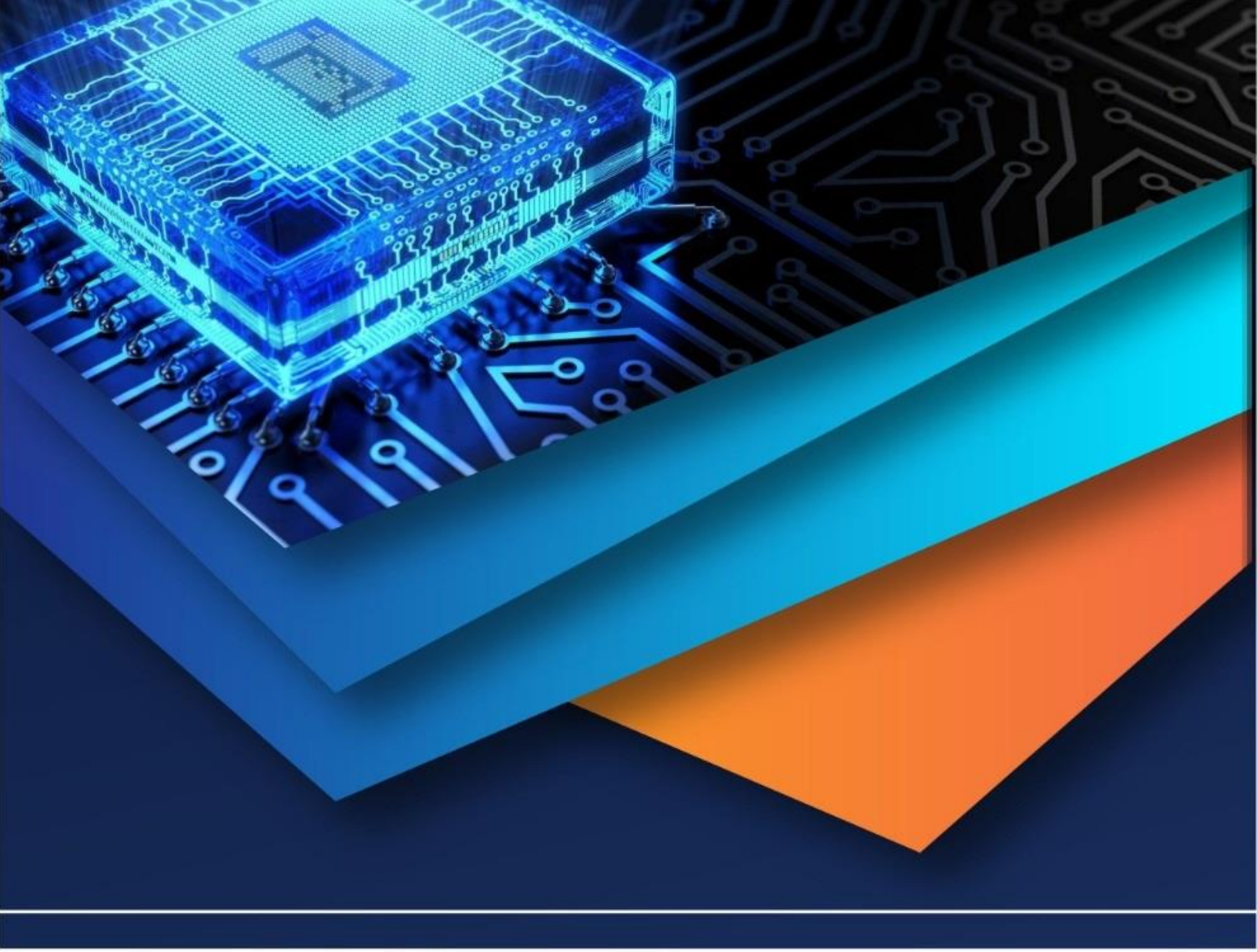
[11] Moustafa, N.; Creech, G.; Slay, J. Big Data Analytics for Intrusion Detection System: Statistical Decision- Making Using Finite Dirichlet Mixture Models. In Data Analytics and Decision Support for Cybersecurity; Springer: Cham, Switzerland, 2017; pp. 127–156

[12] Koroniotis, N.; Moustafa, N.; Sitnikova, E.; Slay, J. Towards developing network forensic mechanism for botnet activities in the IoT based on machine learning techniques. In Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering; Springer: Cham, Switzerland, 2018; Volume 235, pp. 30–44. [CrossRef]

[13] Khan, M.A.; Kim, J. Toward developing efficient Conv-AE-based intrusion detection system using heterogeneous dataset. Electronics 2020, 9, 1771. [CrossRef]

[14] Jin, D.; Lu, Y.; Qin, J.; Cheng, Z.; Mao, Z. SwiftIDS: Real-time intrusion detection system based on LightGBM and parallel intrusion detection mechanism. Comput. Secur. 2020, 97, 101984. [CrossRef]

[15] Ferrag, M.A.; Maglaras, L.; Ahmim, A.; Derdour, M.; Janicke, H. Rdtids: Rules and decision tree-based intrusion detection system for internet-of-things networks. Future Internet 2020, 12, 44. [CrossRef]



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)