iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** X **Month of publication:** October 2021

DOI: https://doi.org/10.22214/ijraset.2021.38503

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: 7.429
Volume 9 Issue X Oct 2021- Available at www.ijraset.com

NIDM: Novel Intrusion Detection Model Using CNN-LSTM for Big Data Environment

Rohan Benhal[1], Tanmayee Tushar Parbat[2], Honey Jain[3], Balaso Jagdale[4]
[1, 2, 3]*BBA IT, Pune*
[4]*Guide, Dr. Vishwanath Karad MIT World Peace University, Pune, Maharashtra, India*

Abstract: ***Machine learning-based (IDS) have become a critical component of safeguarding our economic and national security because of the massive quantities of data produced each day and the growing interconnection of the world's Internet infrastructures. The existing machine Learning Model technique may have difficulty comprehending the ever-increasingly complex distribution of data invasion patterns. With a small number of data points, a single deep learning algorithm may be ineffective at capturing different patterns for intrusive attacks. We presented CNN-LSTM Novel Intrusion Detection Model for Big Data to improve the efficiency of IDS-based CNN-LSTM even further (NIDM). NIDM uses behavioural traits and content functions to understand the characteristics when compared to earlier single learning model tactics, this strategy can improve the rate of intrusive attack detection.***
Keywords: IDS, Machine Learning, LSTM, CNN.

I. INTRODUCTION

ICT systems and networks manage different sensitive user data that are prone to numerous assaults from both internal and external attackers [1]. These assaults can be human and engine-generated, varied and progressive, leading to undiscovered data violations. For example, the violation of Yahoo data led to the loss of $350M and the breach of Bitcoin led to a preliminary estimate of $70M damage [2]. Such cyber attacks are continuously growing in complex algorithms, particularly in recent advances on the Internet of Things (Iot)[4], with the improvement of hardware, software and network topologies. Malicious cyber attacks offer severe safety problems which require a new, adaptable and trustworthy intrusion detection system (IDS). An IDS is a proactive intrusion detection technology used to automatically identify and categorise intrusions, attack or security breaches on network and host infrastructure. A network-based IDS system is termed NIDS. Networking behaviour through network equipment is gathered and evaluated using network equipment mirrors such as switches, routers and networking taps to discover attacks and probable dangers in network traffic. An IDS system, which employs system activity to identify assaults in the form of different log files on the host local computer, is termed HIDS. Local sensors record the log files. While NIDS inspects each packet content in network traffic, HIDS depends on log file information including logs, system logs, software logs, file systems, disc resources, user accounting information and other system information. Many companies employ both NIDS and HIDS hybrids. Network traffic flows are analysed utilising abuse detection, anomaly detection and a comprehensive protocol analysis. In order to detect assaults, misuse detection utilises preset signatures and filters. It relies on human inputs to update the signature database continually. This approach is correct for identifying known assaults, yet in the event of unknown attacks it is entirely worthless. Anomaly detection utilises heuristic processes to locate unknown harmful actions . In the majority of situations, anomaly detection has a substantial false positive rate[5]. Most companies utilise a mix of abuse and anomaly detection to address this challenge in their business solutions. As state-of-the-art analysis works on the network layer, application layer and transport layer, the three detection approaches are most potent. This technique identifies deviations by leveraging known supplier specification parameters in appropriate protocols and applications. While deep learning approaches are being researched to enhance intrusion detection intelligence, a study has yet to be carried out comparing such machine learning algorithms with public data sets. The main challenges with present solutions based on machine learning are: Firstly, models create a high false positive rate [3]; secondly, models cannot be generalised because most current research have only utilised one dataset to describe the performance of the machine learning model. This is the motivating factor behind this investigation, which attempts to evaluate the performance of several conventional classification systems and deep neural networks (CNN-LSTM) in NIDS and HIDS applications.to organize the paper in the following manner, Second section represent the related work , thirst section represent the proposed methodology and proposed algorithm , forth section to represent the conclusion and future work

.

II. RELATED WORK

Kotenko, I. et al[1]This study proposes a novel technique to detecting such cyber assaults. The method is built on sharing weighted ensembles of several classifiers as well as the Big Data processing architecture.

W. Zhong, et al[2 When compared to prior single learning model techniques, solutions can enhance the detection rate of intrusive attacks.

E. Unal, et al[3]. This study investigates how effectively these SDN assaults can be predicted with this technique. The preliminary findings acquired from four models of machine learning by using conventional machine learning algorithms on SDN simulated routing data, if assaults occur.

E. Unal et al[4] The coupling of rapid and large network measurement data with the adaptive training paradigm presents significant problems for data processing speed, which we handle using big data platforms for parallel stream processing.

P. T. Dinh et al.[5] This study offers a big data framework for overcoming traditional data processing constraints and for the efficient exploitation of distributed resources for most computational activities. Our actual tests illustrate the resilience, scalability and efficiency of our system.

P. Casas, et al[6] utilises Big DAMA to benchmarking several monitor ML models to detect various kinds of network assaults and abnormalities.

G. Clark, et al.[7] This study primarily contributes to the early evaluation of the effective manipulation of controlling real-time traffic via an indirect attack. Research has shown that this attack is effective and relevant.

III. PROPOSED METHODOLOGY

In this work, NIDM aims to study the particular distribution of data of certain intrusive attacks pertaining to various families. This approach is very successful in capturing subtle data patterns for a limited number of invasive attacks. NIDM also takes both behavioural and content characteristics.

Taking both behavioural characteristics and content functions together, NIDM allows the analysis of invasive attack samples using both network traffic and payload content characteristics. This method can enhance IDS performance because prior systems never integrate both sorts of characteristics. Our work also indicates that large-scale data technology and parallel methodologies for selecting, grouping and training features may considerably cut model development time. This helps academics to iterate quicker for their computer issues to seek for the optimum model parameters. This study uses a basic decision fusion method to integrate the output of the several deep learning models in the cluster. This approach may not be the perfect answer; Thus, improved CNN-LSTM -fusion algorithms that integrate outputs from various profound tree learning models can be tested soon. To define merger models, we try to use deep neural networks instead of human experts to combine decisions made from various attacking patterns. Since selection and adoption of features have major consequences for the performance of deep learning models, we intend to add new sets of behavioural features into deep learning models so as to improve performance. The rapid approaches for producing multi-level cluster trees must also be investigated to further decrease model creation. CNN-LSTM requires far more computing resources to produce performance benefits compared to the unique deep learning technique. How to minimise the minimal computing resources necessary to obtain the same performance benefits will be explored soon..

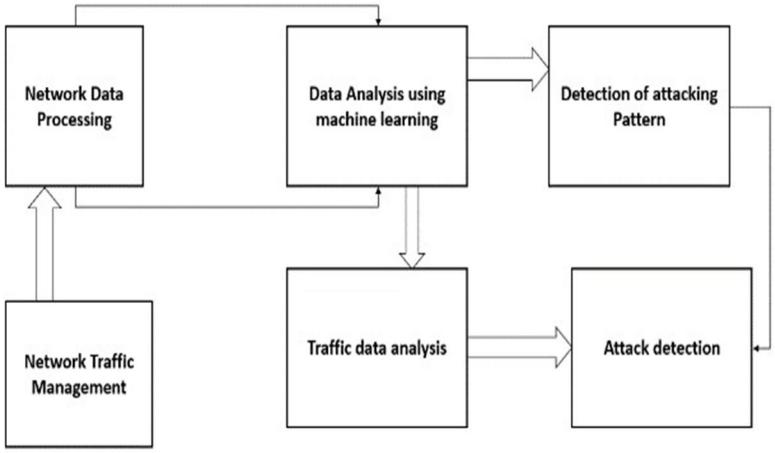


Figure 1: Attacker data analysis

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue X Oct 2021- Available at www.ijraset.com

LSTM LSTM is an RNN variant designed to deal with gradient difficulties such as disappearing and exploding gradients. LSTM can be used to learn long dependencies. The LSTM cell includes three gates to govern and protect cell states: forget, input, and output. The input, forget, output, and state of a memory cell are computed as follows: it, ft, ot, and ct are the input, forget, and output gates, respectively, and ct is the cell state. The sigmoid function, where xt represents the input layer value at time t, ht represents the hidden cell state at time t, and W represents the weight values. The Stochastic Gradient Descent approach uses the learning rate to optimise neural network weights and attain the lowest loss function. In response to the SGD update of (L) /modification, the current gradient weight w oscillates. The authors of [9] employed LSTM to identify and categorise Android malware, with the Android real-world malware test dataset attaining the greatest accuracy.

To detect attacks, we use LSTM in our proposal.

Performance of the LSTM mode

$$i_t = \sigma(W_i h_{t-1} + W_i x_i + b_i)$$
$$f_t = \sigma(W_f h_{t-1} + W_f x_t + b_f)$$
$$o_t = \sigma(W_o X_t + W_o h_{t-1} + b_o)$$
$$h_t = o_t \tanh(c_t)$$
$$c_t = f_t c_{t-1} + i_t \tanh(W_c x_t + W_c h_{t-1})$$

$$w_{t+1} = w_t - \alpha \frac{\partial(L)}{\partial(w_t)}$$

CNN-LSTM was created to predict visual times as well as the time of textual images from sequences such as activity detection and attacking pattern description. CNN-LSTM The CNN-LSTM architecture combines CNN layers and LSTM layers to extract features for time sequence prediction from inputs. DNN's language recognition has improved thanks to the use of CNN-LSTM. It is employed in the fields of visual recognition and description.

IV. RESULTS ANALYSIS

To run the simulation, I utilised the Anaconda tool with a Python application and a variety of python deep learning libraries. We use a variety of datasets to train and evaluate CNN-LSTM models in binary and multi-class classifications. The is being used to detect traditional attacks as well as other IoT-based attacks (UNSW-NB-2015, CICIDS2017 and NSL-KDD). The data sets were chosen based on the attacks' freshness and diversity. We choose to prove that the suggested system can detect classical assaults in addition to IoT threats. We also need to prove that utilising deep learning for IoT threat detection is a good idea and technology. All of these models are capable of extracting deep properties from raw data. During the detection procedure, the DL models' attributes are compared to the test characteristics. The response time in CNN-LSTM detection is estimated to demonstrate the effectiveness of the proposed deep learning CNN-LSTM attack detection framework in terms of reaction times. To evaluate DL models, DR and FAR are calculated as measurement metrics. For measuring and comparing performance, accuracy, recall, F1 measurement, and detection time were also used. The fraction of the overall proper categorization number is referred to as DR. The term "FAR" refers to an erroneously classified ratio of regular events as dangerous. The fraction of samples properly classified over the total sample number is known as precision. The number of positive samples that are identified as positive is determined by recall. The weighted average accuracy and remainder is represented by F1. The detection time is the time it takes for a packet to be classified as normal or malicious. Although training time is not crucial, the DL model can be trained offline utilising GPU to speed up training. The average detection time and the average time are calculated in our experiment. The mathematical representation of the assessment measurements is shown in the equations below.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN},$$
$$FAR = \frac{FP}{FP + TN}, DR = \frac{TP}{TP + FN}$$
$$Precision = \frac{TP}{TP + FP}, Recall = \frac{TP}{TP + FN},$$
$$F1 - Measure = 2 * \frac{Precision * Recall}{Precision + Recall}$$

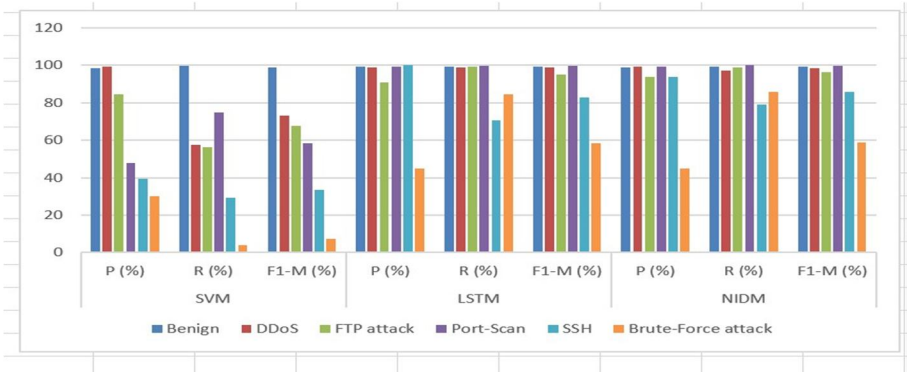
To acquire the best-trained models in binary classification, we employ learning rates of 0.1, 0.01 and 0.001 for all DL models, batch sizes of 32, 64, and 128 for batch size, and epoch number of 100 for epoch number. The best results were obtained with 0.01 as the rate, 64 as the batch size, and Adam as the optimization algorithm. The CNN-LSTM is made up of a 1024 cell input layer, five 512 cell hidden layers with ReLU activation each, and one cell output layer with sigmoid activation. The LSTM is made up of a sigmoid-activated 128-cell input layer, three 256-cell hidden layers, and a single-cell output layer. The CNN-LSTM is made up of a 128-cell input layer, three hidden layers of 128 cells each, and a single cell output layer with sigmoid activation. GRU features a 64-cell input function, three hidden levels of 64 cells each, and a sigmoid-activated single-cell output layer. The CNN is made up of three convolutional layers, each having 64 ReLU activating filters, three pooling layers, and one output layer with one cell triggered by the sigmoid activation function.

DL models for multi-class classification have the same architecture as binary classification, but the output layer is equal to the number of classes, and the softmax activation function is used instead of the sigmoid. Dropout is a technique for avoiding overfitting. Each DL model is trained and tested using five data sets for binary and multi-class classifications. Each dataset provides a variety of assaults with different patterns to test the deeper models' ability to recognise different raw data patterns.

The best deep learning model across all datasets is picked. All DL models' assessment metrics, as well as their binary classification performance, are listed in Table 2. The maximum accuracy is achieved by LSTM (Table 1). The five datasets in binary classification were used to evaluate CNN-LSTM models.

Dataset Name	DL Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Measure (%)	Time (msec)
Attacking Dataset	Proposed	99.32	99.79	99.87	99.83	0.120
	LSTM	95.96	95.96	99.97	99.98	0.345
	Decision Tree	93.52	92.32	99.83	99.83	0.8245
	RF	90.32	89.45	99.77	99.78	0.453
	MCNN	97.32	93.43	99.78	99.82	0.203
	CNN	95.52	95.43	98.97	99.46	0.342

Table 2: Performance of CNN-LSTM models with attacking dataset in multi-class classification.



V. CONCLUSION AND FUTURE WORK

To propose the attack detection system, the CNN-LSTM model for assaulting data traffic classification is proposed in this study. Edge layer traffic is gathered and transmitted to the cloud layer to train the LSTM model in the proposed architecture. The trained model is then utilised as a detection engine on the fog layer nodes to recognise attacks. A cloud service is used to track and update the CNN-LSTM model's performance. The trials revealed that the deep learning models used in cyber security were capable of detecting a wide range of threats with high detection and accuracy rates. Standard DL models are also shown to be capable of detecting and identifying cyber hazards in a range of datasets. Because it has forgotten about the possibilities of preserving prior state information, the CNN-LSTM model beats the other supervised DL models utilised in the trial. It is inconvenient, however, to label data received on the edge layer for cloud training of the LSTM model. In the future, the suggested attack detection method will be tested in distributed computing settings such as Apache Spark and other data sets, leveraging unsupervised deep learning models and enhanced learning.

REFERENCES

[1] Kotenko, I. Saenko and A. Branitskiy, "Detection of Distributed Cyber Attacks Based on Weighted Ensembles of Classifiers and Big Data Processing Architecture," IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2019, pp. 1-6.

[2] W. Zhong, N. Yu and C. Ai, "Applying big data based deep learning systems to intrusion detection," in Big Data Mining and Analytics, vol. 3, no. 3, pp. 181-195, Sept. 2020, doi: 10.26599/BDMA.2020.9020003.

[3] P. Mulinka, P. Casas and J. Vanerio, "Continuous and Adaptive Learning over Big Streaming Data for Network Security," 2019 IEEE 8th International Conference on Cloud Networking (CloudNet), 2019, pp. 1-4, doi: 10.1109/CloudNet47604.2019.9064134.

[4] E. Unal, S. Sen-Baidya and R. Hewett, "Towards Prediction of Security Attacks on Software Defined Networks: A Big Data Analytic Approach," 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 4582-4588, doi: 10.1109/BigData.2018.8622524.

[5] P. T. Dinh and M. Park, "BDF-SDN: A Big Data Framework for DDoS Attack Detection in Large-Scale SDN-Based Cloud," 2021 IEEE Conference on Dependable and Secure Computing (DSC), 2021, pp. 1-8, doi: 10.1109/DSC49826.2021.9346269.

[6] P. Casas, F. Soro, J. Vanerio, G. Settanni and A. D'Alconzo, "Network security and anomaly detection with Big-DAMA, a big data analytics framework," 2017 IEEE 6th International Conference on Cloud Networking (CloudNet), 2017, pp. 1-7, doi: 10.1109/CloudNet.2017.8071525.

[7] G. Clark, M. Doran and W. Glisson, "A Malicious Attack on the Machine Learning Policy of a Robotic System," 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), 2018, pp. 516-521, doi: 10.1109/TrustCom/BigDataSE.2018.00079.

[8] K. Veeramachaneni, I. Arnaldo, V. Korrapati, C. Bassias and K. Li, "AI^2: Training a Big Data Machine to Defend," 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016, pp. 49-54, doi: 10.1109/BigDataSecurity-HPSC-IDS.201 6.79.

[9] I. Kotenko, I. Saenko, A. Kushnerevich and A. Branitskiy, "Attack Detection in IoT Critical Infrastructures: A Machine Learning and Big Data Processing Approach," 2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), 2019, pp. 340-347, doi: 10.1109/EMPDP.2019.8671571.

[10] A. Boukhalfa, A. Abdellaoui, N. Hmina and H. Chaoui, "Network Traffic Analysis using Big Data and Deep Learning Techniques," 2020 IEEE 6th International Conference on Optimization and Applications (ICOA), 2020, pp. 1-4, doi: 10.1109/ICOA49421.2020.9094455.

[11] J. Ali, B. -h. Roh, B. Lee, J. Oh and M. Adil, "A Machine Learning Framework for Prevention of Software-Defined Networking controller from DDoS Attacks and dimensionality reduction of big data," 2020 International Conference on Information and Communication Technology Convergence (ICTC), 2020, pp. 515-519, doi: 10.1109/ICTC49870.2020.9289504.

[12] X. Chen, J. Ji, C. Luo, W. Liao and P. Li, "When Machine Learning Meets Blockchain: A Decentralized, Privacy-preserving and Secure Design," 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 1178-1187, doi: 10.1109/BigData.2018.8622598.

[13] Rajawat A.S., Rawat R., Barhanpurkar K., Shaw R.N., Ghosh A. (2021) Vulnerability Analysis at Industrial Internet of Things Platform on Dark Web Network Using Computational Intelligence. In: Bansal J.C., Paprzycki M., Bianchini M., Das S. (eds) Computationally Intelligent Systems and their Applications. Studies in Computational Intelligence, vol 950. Springer, Singapore. https://doi.org/10.1007/978-981-16-04072_4.

[14] Gautam S., Om H., Dixit K. (2021) A Novel Multilevel Classifier Hybrid Model for Intrusion Detection Using Machine Learning. In: Das S.K., Dao TP., Perumal T. (eds) Nature-Inspired Computing for Smart Application Design. Springer Tracts in Nature-Inspired Computing. Springer, Singapore. https://doi.org/10.1007/978-981-33-6195-9_12

[15] Keerthi Priya L., Perumal V. (2021) A Novel Intrusion Detection System for Wireless Networks Using Deep Learning. In: Uddin M.S., Bansal J.C. (eds) Proceedings of International Joint Conference on Advances in Computational Intelligence. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-16-0586-4_39

[16] Rajawat A.S., Rawat R., Barhanpurkar K., Shaw R.N., Ghosh A. (2021) Blockchain-Based Model for Expanding IoT Device Data Security. In: Bansal J.C., Fung L.C.C., Simic M., Ghosh A. (eds) Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing, vol 1319. Springer, Singapore. https://doi.org/10.1007/978-981-33-6919-1_5

[17] Satpute K., Agrawal S., Agrawal J., Sharma S. (2013) A Survey on Anomaly Detection in Network Intrusion Detection System Using Particle Swarm Optimization Based Machine Learning Techniques. In: Satapathy S., Udgata S., Biswal B. (eds) Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA). Advances in Intelligent Systems and Computing, vol 199. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-35314-7_50 [18]. https://doi.org/10.1007/978-3-030-38040-3_76

[18] Gulghane S., Shingate V., Bondgulwar S., Awari G., Sagar P. (2020) A Survey on Intrusion Detection System Using Machine Learning Algorithms. In: Raj J., Bashar A., Ramson S. (eds) Innovative Data Communication Technologies and Application. ICIDCA2019. Lecture Notes on Data Engineering and Communications Technologies, vol46. Springer, Cham

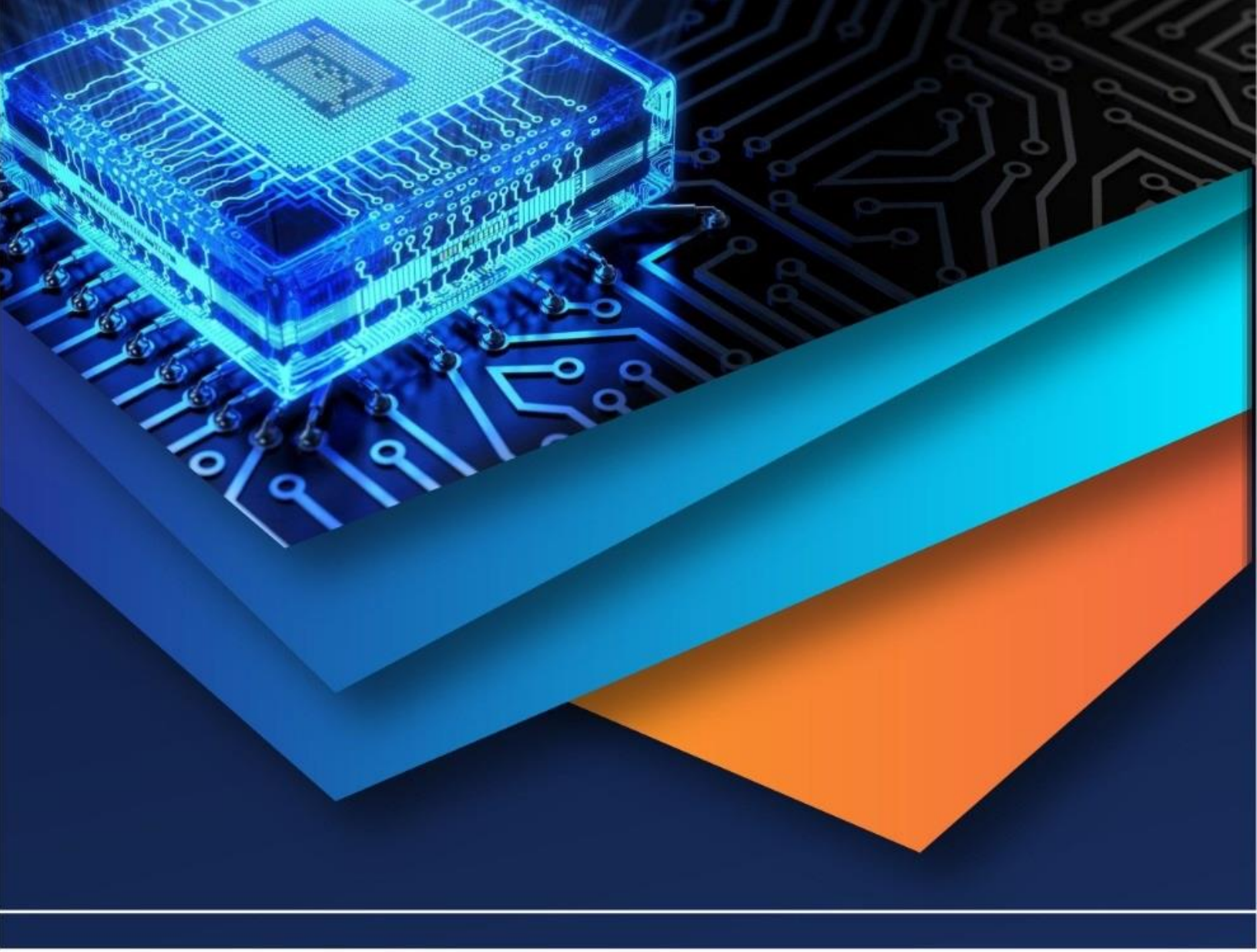
[19] Rajawat A.S., Rawat R., Shaw R.N., Ghosh A. (2021) Cyber Physical System Fraud Analysis by Mobile Robot. In: Bianchini M., Simic M., Ghosh A., Shaw R.N. (eds) Machine Learning for Robotics Applications. Studies in Computational Intelligence, vol 960. Springer, Singapore. https://doi.org/10.1007/978-981-16-0598-7_4

[20] Dutt I., Borah S., Maitra I.K., Bhowmik K., Maity A., Das S. (2018) Real-Time Hybrid Intrusion Detection System Using Machine Learning Techniques. In: Bera R., Sarkar S., Chakraborty S. (eds) Advances in Communication, Devices and Networking. Lecture Notes in Electrical Engineering, vol 462. Springer, Singapore. https://doi.org/10.1007/978-981-10-7901-6_95

[21] Ahmadi R., Macredie R.D., Tucker A. (2018) Intrusion Detection Using Transfer Learning in Machine Learning Classifiers Between Non-cloud and Cloud Datasets. In: Yin H., Camacho D., Novais P., Tallón-Ballesteros A. (eds) Intelligent Data Engineering and Automated Learning – IDEAL 2018. IDEAL 2018. Lecture Notes in Computer Science, vol 11314. Springer, Cham. https://doi.org/10.1007/978-3-030-03493-1_58

[22] Rachidi T., Koucham O., Assem N. (2016) Combined Data and Execution Flow Host Intrusion Detection Using Machine Learning. In: Bi Y., Kapoor S., Bhatia R. (eds) Intelligent Systems and Applications. Studies in Computational Intelligence, vol 650. Springer, Cham. https://doi.org/10.1007/978-3-319-33386-1_21

[23] Mridha M.F., Abdul Hamid M., Asaduzzaman M. (2020) Issues of Internet of Things (IoT) and an Intrusion Detection System for IoT Using Machine Learning Paradigm. In: Uddin M., Bansal J. (eds) Proceedings of International Joint Conference on Computational Intelligence. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-13-75644_34



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY