Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

Accurate Deep Neural Network Technique Based Network Intrusions Detection System

Batool Jameel Zaidan

Foundation of Martyrs, Najaf Martyrs Directorate, Department of IT, Najaf/Iraq, Email: batoolkapy@gmail.com

ARTICLE INFO

Article history:
Received: 2 /9/2024
Rrevised form: 20 /10/2024
Accepted : 02 /12/2024
Available online: 30 /12/2024

Keywords:

DNN,

intrusion detection,

ML, F1-score,

Accuracy,

Precision,

Recall

ABSTRACT

Because of the fast growing in network system, many categories of intrusion has been discovered that differs from current one and convention firewall and definite rules set and strategies are unable of recognizing this intrusion in real-time. Hence, this demand is requirements of real-times intrusions detection systems (RTs-IDS). The vital aim of this paper is to build an RT-IDSs proficient of classifying intrusion by analyzing the outbound and incoming networks information in real-times. The suggested method contains of deep neural networks (DNNs) trained by use 28 types of the NSL-KDDs datasets. Furthermore, it comprises the machine learning (MLs) pipelines with successive modules for category of data encode and features scaling, that is use before transmit the real-times information to the train DNNs models to create prediction. Composed of the train DNNs models, the MLs pipelines are introduced in the servers that can be access through representation state transfer applications program interface (RESTs API). The DNNs has displayed outstand test performance result realizing around 70% to 96% for f1-score, accuracy, precisions, and recalls. These works comprise a complete practical clarification regarding the implementations and functional of the whole systems. The suggested system usability and efficiency have been increased by its comfort of implementations and remotely accessing. In addition, the proposed model is extremely beneficial for rapidly detects the intrusion by analyze incoming and outbound networks traffics.

MSC..

https://doi.org/ 10.29304/jqcsm.2024.16.41781

1. Introduction

The industry revolutions were considered the growth of internet and computing technologies along with the sufficiently potential related to advanced human activities [1-3]. Therefore, many processing in environment is presently in electronics or wholly digital and physically. As an example, the payment for good and service could currently be carry out with minimum or zero physical existence of the buyers and sellers, since online vendor is existing and make it probable for both buyers and sellers to manage online

∗Corresponding author

Email addresses:

Communicated by ‘sub etitor’

[4]. Moreover, the services of bank do not essentially follow currently in a physical manor because of digitally knowledge; this service can be access by basically push button or swipe over the interface of computer device which is network to other device. According to [5-7], the computers networks are an interconnection of computer device that not only interchange information and data but could participate the resource as well. Though, despite the several assistances that computers and internet network have carried to mankind the illegal predilection of this technology advancement continue to increase. The criminals' element is flooding the internet and other intra-network every day in current searches. This hateful actor tends to preys upon victim for diverse details like vengeance, spying, payment, race; ego boosts [8-10]. Several threat and attack have been employing to settlement the smoothed function of computers network for criminal gain [11]. This network threat and attack like denials of services (DoSs), distribute denial of services (DDoSs), worm, virus, injection attack [12-14] attempts to compromised the privacy, integrities and accessibility of computers network and internet service [15-20]. The rising in the internet user numbers and the appearance of internet of thing (IoTs) have make it progressively dominant to offer intervention that could detects threat and attack that can compromised data network [21]. The widespread interference that is realized in the cyber security spaces are IDS. The intrusions detections systems are definite basically as arrangement of hardware and software system for sensing intrusion or attack in the computers system [22]. Also, the IDS scan networks traffics to classify and then reported intrusion depends on preconfigure detections flag [23]. Usual IDS could variety from one computer systems big networks infrastructures [24]. The classification of IDSs depends on scopes of habitations such as HIDSs and NIDSs. Besides, the methods to detection by IDSs also bring about another arrangement of IDS such as Signature-based IDS (SIDSs) and Anomaly-based IDS. Signature-based IDSs commonly detect intrusion by observing a specific pattern in the data traffic such as byte sequence or instruction sequence. It thereafter matches them against a database of already known attack signatures.

Currently, the major advance in an automotive system has been made with integrating a number of computing device called Electronic Control Units. This computing device is used for controlling and monitoring a subsystem of network efficiency enhancement, and noise reduction. These systems replace conventional mechanical controlling parts. As technologies develop, new and complex cyber-attacks are being deployed to break through system in order to exploit vulnerabilities and other malicious activity. Networks infrastructure is one of the main system which experiences a dense quantity of different types of cyber-attacks such as Denial of Service (DoS) or distributed denial-of-service attack (DDoS), TCP SYN Flood attacks, Ping of death attacks, Teardrop attacks, Scan attack, etc. Hence, it has been observed that a great amount of effort was put in finding and implementing different methods and techniques to block these attacks and ensure that the network is safe and secure while maintaining high availability to the legit users of the network. As well-known method of securing the network is through implementing an intrusion detection system (IDS). This was originally implemented in 1980 by the academic staff. The main aim of their work was to introduce a mechanism which differentiates between benign activities from malicious ones. Additional research was carried out to optimizing this methodology to aid monitoring the network traffic in case of attack; this system is now known as Network Intrusion. Traditional machine learning based anomaly detection systems mainly classify and detect network traffic by analyzing the manually extracted feature of network traffic [25]. Such approach still presents a high false positive rate, which significantly limit the in-time detection efficiency, incurs large manual scrutiny workloads, and cannot detect any unknown and new (0-day) attack. On the other hand, Deep Learning base system can not only analyze the manually extracted feature but also automatically extracts the feature from the original traffic and have been proven to detect new feature and attacks pattern automatically to discover new attack in this constantly evolving landscape [25].

The main type of intrusion detection system include network intrusion detection system (NIDS) may consist of both hardware (sensor) and software (console) to control and monitor network traffic packet at multiple locations for a potential intrusion or anomaly. Host intrusion detection system (HIDS) reside

on a particular computers or servers, identified as the host, and monitors activity only on these systems. Although tapered to only one system, it offers higher capabilities than NIDS, as it can access encrypted information traversing the network, including system configuration database, registries, and file attribute. A Cloud intrusion detection system is a combination of cloud, networks, and host layer. The cloud layer provides a secure authentication into the demand-based access to a shared group or application programming interface (API). Similarly, it will create a bridge between existing IDS and hypervisors [26].

The paper structure consists of three sections. In section II, the related works is present. Section II includes the methodology of the techniques used in this study. Section IV presents the results and discussion; finaly in section V the conclusion is explained.

2. RELATED WORKS

Collections of current approaches and experimentations on RT-IDS are present in this section. Though, the common of these suggestions on IDSs have been prepared the accomplishment benchmark on different machine learning algorithm by use many dataset. Hence, restricted numbers of study existing on RT-IDS is summarized in briefly. The real-time intrusions and anomalies detection systems depend on self-organize maps (SOPs) have been proposed by [25]. They classify many attack or normal and once attacks are recognized, it categorizes rendering to the applicable attacks types. Furthermore, they use 2-subset of KDD99 datasets for testing and training purpose. Though, their works lack evocative practical explanation on real-time dataset capture. Experimentally, the demonstration of decision trees (DTs) method outperform Ripper Rules, back-propagations neural networks (BPNNs), Bayesian networks (BNs), Naïve Bayes (NBs), and radial basis functional neural networks (RBF-NNs) have been introduced by [26]. Besides, their DTs algorithms depend on RT-IDS could categorize arriving information as normal or attacks with a detections rates greater than 97%. Their DTs algorithms were trained by use the RLD09 datasets. Additionally, they have extracts 12 types and the data gains technique was use for features selections. The post-process techniques for lower the false alarms rates is used and have displayed that the RT-IDS is effective in finding rates and recall utilizations and could categorized the income networks information in 2 seconds. The authors in [27] have advanced an RT-IDS talented of detects intrusion for networks traffics with a high precisions. They comprise 4-module include networks data acquisitions, data pre-process, convolution neural networks (CNNs), and intrusions detections. Their CNNs was train by use NSL-KDD datasets. Within data pre-process, one-hot encode and features control was use for definite information encode and features scale. A novelty of frameworks design that contains of five components include pre-process, auto encoders, database, classifications, and feedbacks have been suggested by [28]. This framework suggestion was assessed by use CICIDS2017 datasets with sparse auto encoders were usage to handle a dimensional via removing unimportant feature. The RF is use as core supervises classifications algorithms in their frameworks. The authors achieved capable result in their assessment, with accuracy of 0.9992 for binary classifications and 0.9990 for multi-class classifications. The authors in [29] propose detection of intrusion method depend on deeps SAEs. They conclude learning of unsupervised and deeps SAEs method were use to extracts feature efficiently. Their DSAE has been trained on regression-relate task was used to extracts feature from the NSL-KDDs datasets. The sources duty not same dataset distributions as the objective domains and all fields are associated through times natures of inputs feature and irregular performance. An networks intrusion detections frameworks depend on a Bayesian networks by use a wrappers method have been proposed by [30]. Their suggested framework removes irrelevant feature by use genetic algorithms features collection methods, and a

Bayesian classifiers are engaged as the base classifiers to classify the attacks type. Their method behavior was assessed by use NSL-KDDs datasets and has realized an accuracy of 98.2653%, outstripping algorithm like KNNs, Boost DTs, Hidden NBs, and Markov Chains. Many other approaches were introduced in this filed such as [30-42]. The presence of many categories of threat has required systems accomplished of caring system and network. Though, the firewalls can agree, disagree, or reject incoming data packet depend on the rules sets, it is unable of classifying intrusion. Nevertheless, the widespread study has been done to examine the behavior of many MLs algorithm depend on the present dataset, a feasible RT-IDS that could classify intrusion is not yet advanced. The qualified analyze we achieved among six MLs algorithm to classify the optimal MLs algorithms for an IDS through previous approaches. The shortage of study interrelated to RTIDS advanced by use DL approach to create prediction by analyzing networks traffics and inaccessibility of full feature RT-IDSs could implements in several systems or use by way of software-as-services to simulate an inspiration for those work. In our paper, we propose advance of DNNs base RT-IDSs developing a NSLs-KDDs datasets to overcome the above challenges.

3. MATERIALS AND METHODS

The proposed model illustrated in Figure 1 shows the alarm systems that has multiple attacks of intrusion detection sensor. Hence, when the system detects an intrusion attack, it gives a short time to disable the alarms. Besides, when the system not disables the alarms within the selected time, the system calls the security of network. The system show how to use the local event program to manage the parallel state, inputs and output events to simulate periodic trigger of the systems. The model contains of four parallel situations include one for every types of anti-intrusion sensors, and a fourth situation that control the alarms. The input to the system contains signals that control whether the alarms are enabled and, for every sensors, an on/off controls and an intrusion signals. The outputs signal to thorough an alerts and to call the security of the networks.

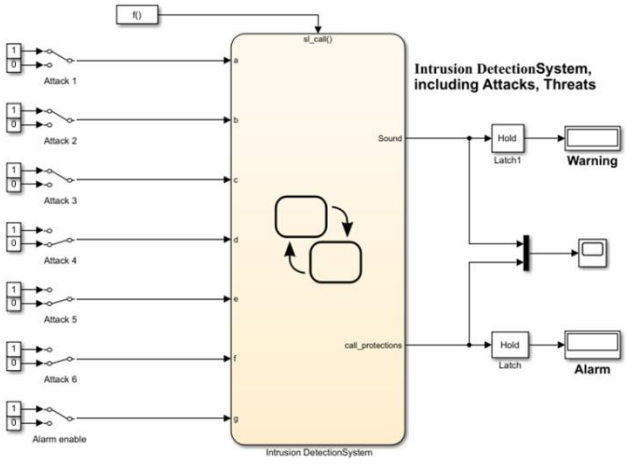

Figure 1: Intrusions attack detection system

9-b: The proposed model is taken from "Design and realization of motion detector system for house security"

Dear Reviewers: The proposed system is totally different…please inspect the input and output and this is just state flow

A. Datasets

The datasets obtainability for the detection of intrusions is infrequent due to utmost dataset not be share because of many privacy and security anxieties. The NSL-KDDs datasets offers open contact to the full datasets and was advanced to astound the inherent problematic of the KDD99 datasets that established depend on the information taken in [30]. The numbers of record in every datasets and the numbers of record related to every attacks kind. Furthermore, the present data in this paper was divided into 125.973 for training and 22.544 for testing process. Nowadays, the DNNs are widely used in the field of intrusions detection and the researchers focusing, and it is an efficient technique. Deep neural network is capable to form or abstract representation and simulated extremely difficult model. This technique has massive probable for realizing active data representations to form suitable solution. The above-mention fact and the relative examination carry out among six MLs algorithms, categorized under supervise, semi-supervise, and unsupervised learning led to employ DNNs for the suggested technique. The DNNs creates output depend on the weight apply to the connection and the associated activations function of the neuron and it is make up of many process layer. The suggested method train the DNNs by use NSLKDD datasets, resultant in high classification accuracy. ML pipeline manual data transformations earlier to training MLs algorithms are unsuccessful and unpractical for real-time commerce-levels application. The MLs work flow of data transformations and comparing the data with the models could be automatic usage the MLs pipeline. The effectiveness and the simply build MLs model will be improved by developing MLs pipeline since the redundant task related to the work flow will be removed. Since data pass through the networks in the form of packet, packets sniffing tool could quickly captured the data packet. Packets sniff application is identified as packets sniffer, and they could read packet that passes over the networks layers of TCP/IP layers. The packets sniffer application is divided into two classes depend on their projected uses. Marketable packets sniffer is used by networks administrator to monitoring and validated networks traffics, while underground packets sniffer is use by individual who sniff other people private and thoughtful data for private benefits. Packets sniffing tool is frequently used to monitor the networks traffics troubleshoot of communication issue, evaluating networks performances, extract username, and recognizing networks intruder. The proposed system modeling of this paper illustrated in Figure 2 which shows the whole flow of the complete system. The Linux environments are installing straight among the association networks and the gateways routers. The networks traffic rolling over the environments of Linux shall be sniff thru a packets sniff technique. Any features extractions formerly extract feature from information that sniff via packets sniff method. Formerly the Linux environments controllers arrange the extracted information as features arrays and drive it as a hypertexts transfer protocols (HTTPs) requested through internets to APIs endpoints of APIs back end. Any APIs back ends controllers extract information from HTTPs requests with feed it to MLs pipelines.

The MLs pipelines contain of two modules definite information encode and transform definite information to arithmetical value, with features scale, that scaled a complete datasets into standards scales. After information pre-process was complete, a pre-process information are send to train DNNs models. Subsequent, the APIs backend controllers return the predictions results as HTTPs responses for the correspond HTTPs requested. Lastly, the Linux environments controllers alert the networks administrators if the HTTPs responses contain anomalies. This work implementations was conducts under eight methods stages include data pre-process, DNNs implementations, MLs pipelines developments, APIs endpoints developments and documentations, MLs integrations, APIs deploy, networks formation and features extractions.

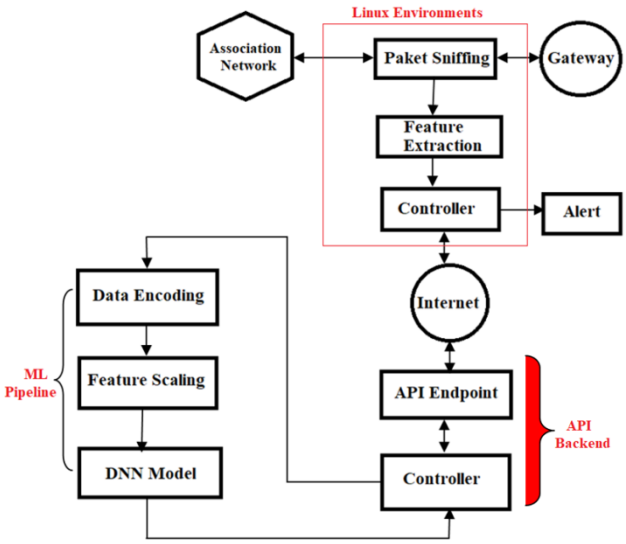

Figure 2: The suggested system modeling

B. Pre-process of Data

The pre-process of data is the first stage that must be achieved before feeds the data into the MLs models. This task is features collection; categorized data encoded and features scale.

Features collection: The NSL-KDDs dataset 41 attribute is categorized into three classes include their basics, contents, and traffic feature. Deprived of checking the payloads, the basic feature could be deriving from the packets header. The duration is used to compute traffic feature. Field skill is necessary, though, to measure the payloads of the packets to derives contented feature. Moreover, the NSL-KDDs datasets author has not obviously state how to derive the contented feature from the packet. Because of the complexity of coming feature from a payloads, the DNNs models is train by use the remains feature whereas exclude contents feature.

Encoded data categorization: The one-hot encoded have been assumed to achieve categorization the data encoding since MLs algorithm realize greatest performances in case of the arithmetical value is used. One-hot encoding was used instead of integer encode. The categorical amount before and after grouping is illustrated in Table 1. Afterward the categorical reductions stage is accomplished; one-hot encoding is assumed for category feature by use the 'One-Hot-Encoders' functions in the Scikit-Learns libraries.

Table 1: dataset before and after categorical reductions

Names	**Numbers of categorical before**	**Number of categorical after**
Protocols	3	3

types		
Services	70	25
Flags	11	11

Features scale: any features scale conclude information pre-process, and employ in convert a numeric value of whole datasets into standards scales. Besides, the Standardize technique is scale tool accomplished of rescale the attribute to zero means and the distributions with unity standards deviations.

C. Implementations of DNN

The DNNs was constructed by use Keras (open access datasets) from [1] that is open-sources software library involves 16 layers (exclude outputs layers) with diverse numbers of neuron in every hidden layer as show in Table 2.

Table 2: neurons amount in every layers

Layers Numbers	1	2	3	4	5	6	7	8	9-16
Neurons amount	64	160	352	320	448	384	192	224	32

Numerous hyper parameters are connected in DNN, which must be prearranged, that have impacts on the behavioral of last model, like the numbers of hidden layer, the numbers of neuron, activations functions, weight initializers, bias initializers, learning rates, regularization coefficients, and optimizers. In DNNs models, an inputs layers with wholly hid layer was activating by use ReLU functions. ReLUs activation is linear functions in case of the inputs are positives the outputs remain zeros. A node is activates by use this function is refer as rectify linear activation units. The outputs layer was activate by use the Sigmoid functions, which could maps any real values zero or one. These functions convert the outputs of the DNNs networks into probable scores. The initialize of the weight and the bias is critical since incorrect initialized might leads to slope explode or disappearing phenomena. Therefore, in case of initialize outsized, it leads to explode gradient, whereas in case of too slight initialize leads to disappearing gradient. Then, to overcome the above phenomenon, the activation must have zeros means, and the alteration must be fixed through every layers. Hence, the weight of the layer which activate by use the ReLU functions is initialize by use uniform initializers, and the outputs layers is initialize by use another uniform initializers. Furthermore, the Tensor Flows-bass Keras initializers function was used for weights initializations. The biases initializations of wholly layer were achieved by use the Zeros initializers. In addition, in the DNNs, the stochastic gradients descents (SGDs) are use as the optimizers within learn rates of 10^{-3}. As loss function, the cross-entropies is employed, and it is accomplished of estimation the losses of the models, with weight of the DNNs updating to decrease the losses on the following assessment. The model of DNNs is trained after perform it for more than 100 epoch, and the cross-validations method have been use to classify the behavioral of models for not train dataset. Additionally, it's probable to classify the models are in fit or under fit by analyze the train and cross validations accuracies curve. Hence, to overcome the over ^{fit}, the 'quick stop' technique is used. Hyper parameters

tune is require enhancing the behavioral of the MLs models. Besides, the Keras tune libraries is used to tune the hyper parameter, the hidden layer numbers, the neuron number in every hidden layers, regularizations coefficients, and learning rates. After train process is completed, DNNs-ML model kept in Java Script objects notations files, texts file in dataset store and transport. Furthermore, a weight of DNNs is kept as hierarchic information formats file, that saved data in hierarchic data formats. The machine learning model is kept to repossess back when prediction is being make, the prediction processing will be disturbed because of the dimension mismatched suffered whereas feeding the data into the machine learning models. Hence, the data must pre-process in similar format; the information pre-process is complete in train step.

D. ML Pipelines Improvement

The advanced MLs pipelines mostly contain two sequential component involve columns transformer and the trained DNNs model. Besides, the columns transformers is a collective of encoders with standards scale is employed to categorize the information encode with features scale consequently. Earlier starting the real-time predictions processing, the pre-train DNNs models and the save column transformers file are consumed into the MLs pipelines as sequential component. Initially, the real-times information extracts from the incoming traffic is fed to the MLs pipelines, and then the 'column transformers' perform one-hot encoder on three predetermine column. Succeeding that, features scale is perform by use the standard scale on wholly column with decimal value. After the achievement of information pre-process, will be feed to a train DNNs, the prediction shall make base on capability. The pipelines use to train the MLs models and to preprocessing the tested datasets and tested the train models. Furthermore, the pipelines could continue stream of networks traffics.

E. Developments of API

The API helps as the back end of the real-times predictions systems and was constructing with flasks. Besides, use flasks are not needed a specialization tool or library; therefore, this observed as micro frameworks. Any ends of communications channels are identified as APIs endpoints. APIs documentations are practical explanation that offers instruction and interacts with APIs, like procedure to call APIs, a formatting of return responses from APIs, diverse responses format dependents on errors kind. A Swaggers documentations frameworks have been use to create the APIs documentations, which is accessible through the (host/. URL)

F. Machine Learning Integration

Micro services are launch with Flasks, and wholly rout pathway was arranged. In order to running the machine learning pipelines, three files should add to the webservers. In the machine learning integrations unit, the JSONs file of machine learning models, the files contains the DNNs weight, with PICKLEs file of column transformers containing the category data encoders and features scalars was developed by use Flasks. Furthermore, the Pythons file including the machine learning pipelines codes is loaded. The train machine learning model is in reserve mode when the Flasks servers are activated, and in case of an APIs requested is made, instant responses are transmit to the clients through the APIs endpoints.

G. Deployment of API

The deployments stage of the Flasks applications requires a production levels servers at the ends of its development. Consequence the Pythons Web Servers Gateways Interfaces has been use. Additionally, for movability among platform, the Flask application was containerize. Though, to offer access through the developments Pipelines construction stage, the web servers is use, and it was made into single container. Subsequently, this container and the container comprising the Flask applications we're combining into a

single file. Therefore, these systems could be performed in any environments by use Dockers technologies. Lastly, the docked file comprising containers was implemented on Linux servers.

H. Features Extraction

The first phase of real-time implementation, the features extraction is the networks configurations. The Linux work stations with two networks interface is install among the gateways routers and LANs to data connections to sniffing incoming and out-bound information packet. Moreover, by use Linux machines configuration, the networks interface was connected practically, then, captured the outbound and inbound data packet flows through Linux work stations employing the packets sniff tool advanced in C++ program languages.

4. RESULT AND DISCUSSIONS

The performance of DNN model performance is assessed by use the loss, accuracy, f1-score, precision, recall, and confusion matrix with the curve. By using the Tensor Figure 3 and Figure 4 were achieved.

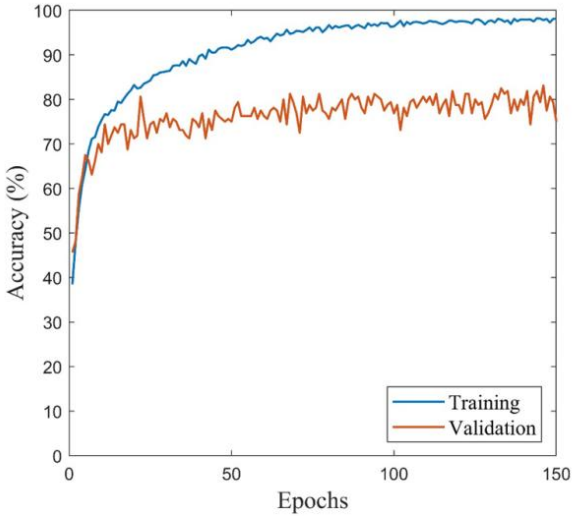

Figure 3: training and validation accuracy

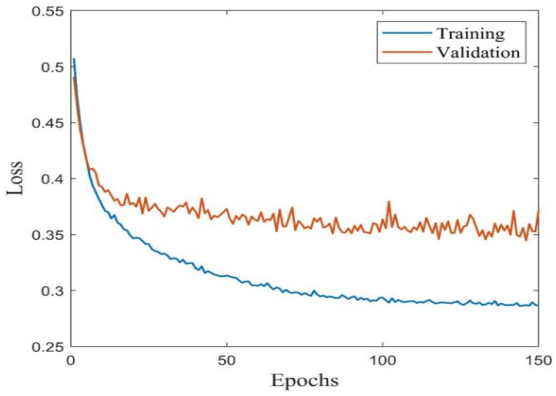

Figure 4: training and validation loss

The Tensors Flow toolkits are used to determine the performance of the loss, accuracy, recall and precision of the training and cross-validations set with the numbers of epoch depend on numbers of predicts trues positive (TPs), false positive (FPs), trues negative (TNs) with false negative (FNs). This behavioral indicate is mostly active when analyzing the performances if the class distributions is tilted. The formula used for all matric is as in [43]:

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad \text{..... (1)}$$

$$Precition = \frac{TP}{(TP+FP)} \quad \text{..... (2)}$$

$$Recall = \frac{TP}{(TP+FN)} \quad \text{..... (3)}$$

$$F1\ Scores = \frac{(2*\text{Precision}*\text{Recall})}{(\text{Precision}+\text{Recall})} \quad \text{..... (4)}$$

$$Specificity = \frac{TN}{(TN+FP)} \quad \text{..... (5)}$$

Table 3 show the performance comparison of the suggested DNNs model with some machine learning classifiers for binary classifications by trained 28 feature of the NSLKDDs datasets.

Table 3: comparison of suggested DNN performance algorithms

Algorithms	**Accuracy**	**Precisions**	**Recalls**	**F1-scores**
KNNs	0.7902	0.9559	0.6163	0.7392
SVMs	0.7388	0.9633	0.5562	0.7061
OSVMs	0.7962	0.9611	0.5433	0.6701
K-Means	0.79499	0.9544	0.5392	0.6915
Suggested DNNs	0.8211	0.9655	0.7120	0.8610

Precision determine the numbers of positive prediction that is accurately positive. Furthermore, recalls calculate the quantity of positives prediction produced using the positives instance in the datasets. The F1-score is consequential by compute the weight middling among precisions and recalls, and it might be uses to find a balance among precisions and recalls. The performances of the models are remarkable when the F1-scores are better. The training and testing result for accuracy, precisions, recalls, and f1-scores has been achieved by use the NSL-KDDs tests datasets. Rendering to the accuracy, losses, precisions, and recall graph in Figs. 3 and Figure 4, the train DNNs model not over fitting or under fitting since both curve have exposed nearly similar value without any significant difference. Additionally, the

normalize confusion matrix obtain by use the test datasets has exposed acceptable result by realizing high value for TP and TN.

5. CONCLUSIONS

This paper presents descriptive methodical data about RT-IDSs based on DNNs machine learning algorithms. Proposed technique can capture real-times networks traffics and classify critical intrusion and it is host in web servers to offer availability for private and commercial sectors network to occupation it to their network through a APIs. The real-time features extraction modules are containerize and it is easy to integrated into any systems. The suggested system usability and efficiency have been increased by its comfort of implementations and remotely accessing. The suggested systems is exceedingly beneficial for rapidly detects the intrusion by analyze incoming and outbound networks traffics. The model output expressive data concerning intrusions information packet. The structure of the DNNs, which is train by use the NSL-KDDs datasets are methodically, discuss with the practical implementations and the simulations result on training and testing aspect. Furthermore, the methods and procedure employ in real-time features extractions from the outbound and inbound networks traffics are obviously stated. Additionally, how the machine learning predictions pipelines are host in website servers was discuss in the study. An experiential result of DNNs train and test has exhibited excellent train result and acceptable result in test phase and precisions of 96%. Lastly, this work has contribution by presents full function RT-IDSs which could be basically implements as an further layers of the networks protections.

REFERENCES

[1] Tavallaee M, Bagheri E, Lu W, Ghorbani AA. A detailed analysis of the KDD CUP 99 data set. In: Proceedings of the 2009 IEEE symposium on computational intelligence in security and defense applications (CISDA), Ottawa, ON, Canada, July 2009. https:// doi.org/10.1109/CISDA.2009.5356528

[2] Edosa Osa, et al., Design and implementation of a deep neural network approach for intrusion detection systems, e-Prime - Advances in Electrical Engineering, Electronics and Energy 7 (2024) 100434

[3] Albara Awajan, A Novel Deep Learning-Based Intrusion Detection System for IoT Networks, Computers 2023, 12, 34

[4] E. Osa, Cyber security terminologies and concepts.' SMART-IEEE-ACity-ICTUCRACC-ICTU-foundations series book chapter on web of deceit - African multistakeholders' perspective on online safety and associated correlates using multi-throng theoretical, Rev. Empir. Des. Approaches (2022) 231–236,

[5] T. Mazhar, D.B. Talpur, T.A. Shloul, Y.Y. Ghadi, I. Haq, I. Ullah, K. Ouahada, H. Hamam, Analysis of IoT security challenges and its solutions using artificial intelligence, Brain Sci. 13 (2023) 683, https://doi.org/10.3390/brainsci13040683.

[6] R.R. Devi, M. Abualkibash, Intrusion detection system classification using different machine learning algorithms on KDD-99 and NSL-KDD datasets - a review paper, Int. J. Comput. Sci. Inf. Technol. 11 (3) (2019) (IJCSIT) VolJune.

[7] A. Abuh, P.E. Orukpe, Development of an integrated campus security alerting and access control system, in: Chapter 7 in Emerging Trends in Engineering Research and Technology, 9, Book Publisher International, United Kingdom, 2020, pp. 57–67.

[8] A. G´eron 'Hands-on machine learning with scikit-learn, Keras, and TensorFlow' O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472 (2019).

[9] S. Sapre, P. Ahmadi, K. Islam 'A robust comparison of the KDDCup99 and NSL-KDD IoT network intrusion detection datasets through various machine learning algorithms' arXiv:1912.13204v1 (2019) [cs.LG] 31 Dec 2019.

[10] P. Wei, Y. Li, Z. Zhang, T. Hu, Z. Li, D. Liu, An optimization method for intrusion detection classification model based on deep belief network, IEEE Access 7 (2019) 87593–8760 https://doi.org/10.1109/ACCESS.2019.2925828.

[11] R. Vinayakumar, M. Alazab, K.P. Soman, P. Poornachandran, A. Al-Nemrat, S. Venkatraman, Deep learning approach for intelligent intrusion detection system, IEEE Access 7 (2019) 41525–41550, https://doi.org/10.1109/ ACCESS.2019.2895334.

[12] E. Osa, O.E. Oghenevbaire, Comparative analysis of machine learning models in computer network intrusion detection, in: Proceedings of the IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON), 2022, pp. 1–5, https://doi.org/10.1109/ NIGERCON54645.2022.9803175.

[13] W. Wang, M. Zhu, X. Zeng, X. Ye, Malware traffic classification using convolutional neural network for representation learning, Int. Conf. Inf. Network. (2017) 712–717.

[14] A.K. Sahu, S. Sharma, M. Tanveer, R. Raja, Internet of things attack detection using hybrid deep learning model, Comput. Commun. 176 (2021) 146–154, 2021.

[15] R.C. Aygun, A.G. Yavuz, Network anomaly detection with stochastically improved autoencoder based models, in: Proceedings of the 4th International Conference on Cyber Security and Cloud Computing, 2017, pp. 193–198.

[16] M. Asad, M. Asim, T. Javed, M.O. Beg, H. Mujtaba, S. Abbas, Deep-detect: detection of distributed denial of service attacks using deep learning, Comput. J. 63 (2020) 983–994.

[17] F. Meng, Y. Fu, F. Lou, Z. Chen, An effective network attack detection method based on kernel PCA and LSTMRNN, in: Proceedings of the International Conference on Computer Systems, Electronics and Control (ICCSEC), 2017.

[18] I. Sharafaldin, A.H. Lashkari, A.A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization, in: Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP), Portugal, 2018. January

[19] H.A. Afolabi, A.A. Aburas, RTL-DL: a hybrid deep learning framework for DDOS attack detection in a big data environment, Int. J. Comput. Netw. Commun. Vol.14 (No.6) (2022) (IJCNC)November.

[20] Faruqui, N.; Yousuf, M.A.; Whaiduzzaman, M.; Azad, A.; Barros, A.; Moni, M.A. LungNet: A hybrid deep-CNN model for lung cancer diagnosis using CT and wearable sensor-based medical IoT data. Comput. Biol. Med. 2021, 139, 104961.

[21] Wójcicki, K.; Biega ´nska, M.; Paliwoda, B.; Górna, J. Internet of Things in Industry: Research Profiling, Application, Challenges and Opportunities—A Review. Energies 2022, 15, 1806.

[22] Hasan, M.K.; Akhtaruzzaman, M.; Kabir, S.R.; Gadekallu, T.R.; Islam, S.; Magalingam, P.; Hassan, R.; Alazab, M.; Alazab, M.A. Evolution of industry and blockchain era: Monitoring price hike and corruption using BIoT for smart government and industry 4.0. IEEE Trans. Ind. Inform. 2022, 18, 9153–9161.

[23] Zhao, Y.; Lian, Y. Event-driven Circuits and Systems: A Promising Low Power Technique for Intelligent Sensors in AIoT Era. IEEE Trans. Circuits Syst. II Express Briefs 2022, 69, 3122–3128.

[24] Soldatos, J.; Gusmeroli, S.; Malo, P.; Di Orio, G. Internet of things applications in future manufacturing. In Digitising the Industry Internet of Things Connecting the Physical, Digital and Virtual Worlds; River Publishers: Delft, The Netherlands, 2022; pp. 153–183.

[25] Kshitiz Sharma, et al., Anomaly Detection in Network Traffic using Deep Learning, 2023 International Conference on Recent Advances in Science and Engineering Technology (ICRASET), **DOI:** 10.1109/ICRASET59632.2023.10419951

[26] Rondon, L.P.; Babun, L.; Aris, A.; Akkaya, K.; Uluagac, A.S. Survey on enterprise Internet-of-Things systems (E-IoT): A security perspective. Ad Hoc Netw. 2022, 125, 102728.

[27] Nayak, J.; Meher, S.K.; Souri, A.; Naik, B.; Vimal, S. Extreme learning machine and bayesian optimization-driven intelligent framework for IoMT cyber-attack detection. J. Supercomput. 2022, 78, 14866–14891.

[28] Husnain, M.; Hayat, K.; Cambiaso, E.; Fayyaz, U.U.; Mongelli, M.; Akram, H.; Ghazanfar Abbas, S.; Shah, G.A. Preventing MQTT Vulnerabilities Using IoT-Enabled Intrusion Detection System. Sensors 2022, 22, 567.

[29] Zheng, Y.; Li, Z.; Xu, X.; Zhao, Q. Dynamic defenses in cyber security: Techniques, methods and challenges. Digit. Commun. Netw. 2022, 8, 422–435.

[30] Tavallaee M, Bagheri E, Lu W, Ghorbani AA. A detailed analysis of the KDD CUP 99 data set. In: Proceedings of the 2009 IEEE symposium on computational intelligence in security and defense applications (CISDA), Ottawa, ON, Canada, July 2009. https:// doi.org/10.1109/CISDA.2009.5356528.

[31] Nimbalkar, P.; Kshirsagar, D. Analysis of rule-based classifiers for IDS in IoT. In Data Science and Security; Springer: New York, NY, USA, 2021; pp. 461–467.

[32] Yılmaz, H.E.; Sirel, A.; Esen, M.F. The impact of internet of things self-security on daily business and business continuity. In Research Anthology on Business Continuity and Navigating Times of Crisis; IGI Global: Hershey, PA, USA, 2022; pp. 695–712. 19.

[33] Harada, R.; Shibata, N.; Kaneko, S.; Honda, K.; Terada, J.; Ishida, Y.; Akashi, K.; Miyachi, T. Quick Suppression of DDoS Attacks by Frame Priority Control in IoT Backhaul with Construction of Mirai-based Attacks. IEEE Access 2022, 10, 22392–22399.

[34] Alazab, A.; Khraisat, A.; Alazab, M.; Singh, S. Detection of Obfuscated Malicious JavaScript Code. Future Internet 2022, 14, 217.

[35] Alazab, M.; Abu Khurma, R.; Awajan, A.; Wedyan, M. Digital Forensics Classification Based on a Hybrid Neural Network and the Salp Swarm Algorithm. Electronics 2022, 11, 1903.

[36] Alazab, M.; Khurma, R.A.; Awajan, A.; Camacho, D. A new intrusion detection system based on moth–flame optimizer algorithm. Expert Syst. Appl. 2022, 210, 118439.

[37] Alzubi, O.A.; Alzubi, J.A.; Alazab, M.; Alrabea, A.; Awajan, A.; Qiqieh, I. Optimized Machine Learning-Based Intrusion Detection System for Fog and Edge Computing Environment. Electronics 2022, 11, 3007.

[38] Alani, M.M.; Damiani, E.; Ghosh, U. DeepIIoT: An Explainable Deep Learning Based Intrusion Detection System for Industrial IOT. In Proceedings of the 2022 IEEE 42nd International Conference on Distributed Computing Systems Workshops (ICDCSW), Hong Kong, China, 18–21 July 2022; IEEE: New York, NY, USA, 2022; pp. 169–174. 31.

[39] Ravi, V.; Chaganti, R.; Alazab, M. Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system. Comput. Electr. Eng. 2022, 102, 108156.

[40] Abdel-Basset, M.; Moustafa, N.; Hawash, H.; Ding, W. Deep Learning Techniques for IoT Security and Privacy; Springer: New York, NY, USA, 2022; Volume 997.

[41] An, G.H.; Cho, T.H. Improving Sinkhole Attack Detection Rate through Knowledge-Based Specification Rule for a Sinkhole Attack Intrusion Detection Technique of IoT. Int. J. Comput. Netw. Appl. (IJCNA) 2022, 9, 169–178.

[42] Hou, D.; Zhao, K.; Li, W.; Du, S. A Realistic, Flexible and Extendible Network Emulation Platform for Space Networks. Electronics 2022, 11, 1236.

[43] Abdullah Aljumah, IoT-based intrusion detection system using convolution neural networks, PeerJ Comput. Sci., DOI 10.7717/peerj-cs.721,2021