iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** V **Month of publication:** May 2020

DOI: http://doi.org/10.22214/ijraset.2020.5022

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 8 Issue V May 2020- Available at www.ijraset.com

Intrusion Detection using Deep Learning Approach with Different Optimization

Ashwini V. Solanke[1], Prof. Dr. Girish K. Patnaik[2]
[1, 2]*Department of Computer Science and Engineering, SSBT's College of Engineering and Technology, Kaviyatri Bahinabai Chaudhari N.M.U, Jalgaon[M.S], India*

Abstract: Intrusion detection plays an important role in security. Various deep learning approaches are used for intrusion detection, but they suffer from certain level of problems such as, high error rate and also number of iterations to be increased for processing desired output because of that accuracy of classification system gets low. The proposed system uses convolutional neural network as a deep learning approach along with different gradient optimization methods to minimize error rate in the training process to make easier. Threshold-based feature selection is used for reducing redundant or unwanted data as preprocessing step to improve performance. The comparative analysis of different optimization methods demonstrates that, the proposed system is achieved high performance accuracy to detect intrusion in traffic. Adagrad, Adadelta, RMSProp and Adam optimization algorithms are evaluated through experiment. As per experiment point of view, an Adam gives much better results in terms of precision, recall and f-measure.

Keywords: Intrusion Detection, Optimization Methods

I. INTRODUCTION

Intrusion is the term that can violate security of computer system or network and another is intrusion detection is the process to identify intrusion. Intrusion detection plays a precious role in taking security against malicious events and also accurately identifies various attacks in the network. Intrusion Detection (ID) system put in place to monitor computer networks. With the rapid development of internet technology, network security problems have become serious with each passing day so, the intrusion detection system protects network system from malware and attack. An intrusion detection system analyses the patterns of capture data from network to detect treats. The studies of existing techniques, a deep learning approach is effective for intrusion detection.

Deep learning is a subpart of machine learning in Artificial Intelligence (AI). Deep learning has networks which is on the basis of unsupervised learning i.e. data is present in unstructured or unlabeled form. Deep learning is attracted towards real-world applications such as, computer vision, graphical modeling, pattern recognition, speech, audio, image, video, natural language and signal processing etc. Intrusion Detection is an act of detecting actions that can lose the confidentiality, integrity or availability of a resource. The goal of the intrusion detection systems is to identify events that can violate security system. Intrusion detection system plays a passive role to identify, gather, log and alert to the system [1].

A. Convolutional Neural Network

Convolutional neural network plays an important role in the history of deep learning. CNN is specialized kind of neural network for processing data in grid form and it is proposed by LeCun in 1989. Convolutional neural network are regularized versions of multilayer perceprton. Multilayer perceptron consist of fully connected layer network, means one layer is connected to all neurons in the next layer. Convolutional networks are successful in practical applications. A Convolutional Neural Network architecture consist of mainly three different types of layers i.e., convolution layer, pooling layer and classification or fully connected layer. CNN indicates that the network with the use of mathematical operation called as convolution. Convolution or filter is main part of CNN and it consists of weights and biases. The main goal of convolution layer is to learn different character representation of input. After the convolution layer, pooling layer is use for reducing computational burden between first layers. The result of convolution goes through non-linearity activation function optimizers at the time of training phase. Every layer is made up of many convolutions filter that produces different feature map with the help of pooling layer. That feature map is divided into sub-sampling layer or in down-sampling by applying respective function such as max pooling or average pooling. The computational burden or complexity is reducing by using pooling. In max pooling, it decreases the dimensionality of data by taking only value from batch of data. In average pooling, computation is carried out with the use of average values of each region. The choice of pooling operation is based on data available for processing over non-overlapping subsets of feature map. Layer-by-layer information is fed for generating desired outcome. These layers are put in between convolution layer and then forward it to the fully connected layer. In fully connected layer each neuron in first layer is connected to another layer. The last fully connected layers output is fed to output layer. The generated attended matrix is carried through fully connected layer to classify the target output.

B. Optimization Methods

Optimization algorithm is use for neural networks to produce better and faster results with the help of models parameters such as weight and bias values. An optimization algorithm is use for minimizing objective function or error rate in the model and also plays a major role in the training process of the neural networks to produce accurate and effective results. Following are the gradient optimization methods

1) *Adaptive Gradient Moment (Adagrad):* Adagrad is an adaptive learning rate method used for minimizing error rate in the weight update. Adagrad method is developed by J. Duchi in 2011. Adagrad method adaptively covered the learning rate for each dimension. Learning rate decay is performing on weight by weight basis and for each and every weight, an adagrad takes gradient that are used in update step. Only the one thing is happened in this method is, learning rate is computed uniquely for each weight on the basis of history values. Adagrad adapts learning rate to the parameter and uses different for every parameter. An adagrad completely removes necessary to manually added learning rate. This method is particularly effective for sparse features means for rarely occurring features because, it decreases learning rate faster to frequent parameters and slow for infrequent parameter, where the learning rate is necessary to scattered more slowly for rarely occurring terms.
2) *Adaptive Delta Moment (Adadelta):* Adadelta is an improved version of adagrad. Adadelta is presented by Matthew D. Zeiler in 2012. Adadelta method is dynamically adapts learning rate for computation of gradient for each dimension and tried to reduce learning rate in decrease form. In this process, instead of gathering or collecting of all past squared gradients, adadelta takes history of gradient but the recent gradients are more important. Adadelta combines two terms i.e. first one is to scaling of learning rate base on past gradient only for limited bunch of set or for recent time window instead of using whole history like, adagrad and second is use the component that serves to speed up terms, that gradually acquiring past gradient like momentum. An Adadelta is to adapts or acquire learning rate based on moving average of gradients updates instead of taking all past gradients. This method is used for overcome the problem of gradient is decrease to zero and also avoiding the continue decay of learning rate to whole process of training. In the denominator the squared gradient is gathering gradually for each iteration from the starting of training. Each and every term is positive so, the acquired gradients sum is continually increase throughout training process. After so many iteration of computation of gradient, the learning rate is become too small. Instead of gathering sum of all gradients, the adadelta is restricting to some limit for batch of set.
3) *Root Mean Squared Propagation:* RMSProp is adaptive learning rate method proposed by Tieleman and Geoffrey Hinton in 2012. RMSProp means Root Mean Squared Propagation method and is similar to the gradient descent with momentum. RMSProp is mini-batch version of Rprop. In RMSProp method, the learning rate is adjusted automatically. An RMSProp algorithm chooses different learning rate for each parameter. RMSProp algorithm carries moving average of squared gradients for each weight and then divides the gradient to mean square by square root. The learning rate is adapts by dividing the root of square of gradient. RMSProp uses exponentially weighted average of squared gradient and then divides learning rate by this average to speed up convergence. This method tries to make slightly wet the oscillations, different way than the momentum. The gradient is roughly calculated on current mini-batch. This optimizers is utilizes the measures of recent gradient to normalize the gradient itself.
4) *Adaptive Moment Estimation:* Adam is name suggest that, adaptive moment estimation. This optimizer is alternative to the classical stochastic gradient descent procedure use for updating the weight of network in iterative fashion based on training dataset. Adam is proposed by Diederik Kingma from university of Toronto in 2015. The idea of Adam combines from RMSProp and momentum. This method computes adaptive learning rate separately for each and every different parameter from the previous estimates such as first and second moments of gradients to adapt learning rate. An Adam algorithm first updates exponential weighted average of gradient and also squared gradients. These are use for estimation of first and second moment and also used for making completely less learning rate of adagrad. An Adam uses squared gradients to change the learning rate and also takes advantage of momentum by using moving average of gradients. Adam is computationally efficient and also it has little memory required for process with current and past gradients. Adam is useful for noisy or sparse i.e. rarely occurring gradients and also well suited for large data or for parameter.

The contribution in proposed system is to reduce unwanted data for improving classification accuracy by using threshold based feature section method. In which convolutional neural network approach of deep learning used for intrusion detection by adding various gradient descent optimization methods for minimizing error rate in the training process.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

II. OBJECTIVES

To develop a system for intrusion detection using convolutional neural network approach with optimization algorithm. To improve the performance of intrusion detection system in terms of precision, recall and f-measure.

III. METHODOLOGY

The proposed system is implementing in two steps such as, for attribute selection threshold- based ranking is used for removing unwanted data in preprocessing. Another is convolutional neural network approach of deep learning along with analysis of different types of optimizers such as Adagrad, Adadelta, RMSProp and Adam.

A. Proposed System Architecture

Fig. 1 illustrates the architecture of proposed system. The proposed system gives the detailed description for comparative analysis of different optimizers to minimize error rate with the use of convolutional neural network as deep learning approach for intrusion detection. The architecture of convolutional neural network consist of different layers i.e., convolution layer, pooling layer, fully connected and final output layer. The more important extracted features go from all these layers for acquiring desired or target output. The input is goes to the preprocessing steps. According to the preprocessing steps, it reduces unwanted data from available training records.

1) *Data Cleaning***:** The term data cleaning can express the redundancies in the datasets. Because of data cleaning, we can reduce the noisy, unnecessary or unwanted data from existing dataset. Only the relevant data is carried out for detection of intruder.
2) *Threshold-based Feature Selection:* For attribute selection, here threshold-based feature selection approach is use. With the use of threshold, we can differentiate coming attack traffic from legitimate one. Feature selection method is widely used for reducing the number of features that are not required for overall process. In the threshold-based feature selection method, each and every attribute is evaluate oppose to the class and free from all the other features in dataset. After normalizing to those attribute or feature set the range in between 0 and 4. This means that, decision of detection can be made with certain level. With the use of threshold-based feature selection network traffic is check against normal problem. If the normal problem records are greater than threshold, it will be considered it as an attack or intruder. After the preprocessing stage, the main propose convolutional neural network with optimizers is process.

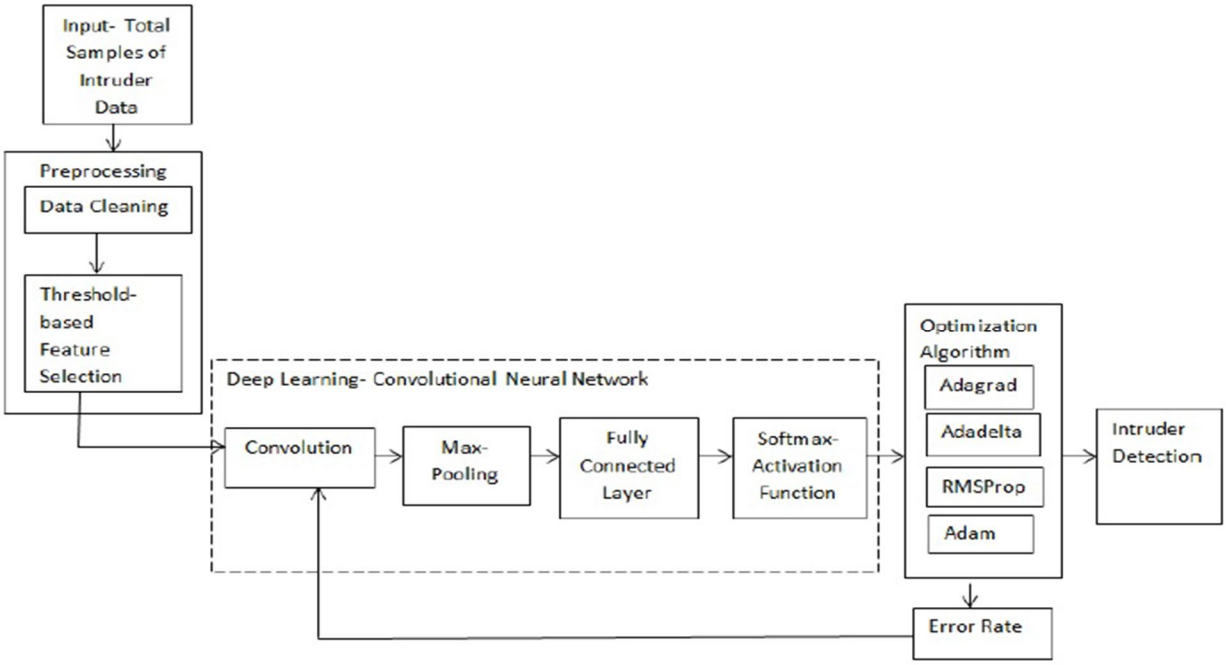


Fig. 1 Proposed System Architecture of Convolutional Neural Network with Optimization

After preprocessing step, in the proposed modified CNN algorithm, the selected features from preprocessing is carried forward to CNN classification model through different layers with different filters in the process. The CNN structure varying in number of layers i.e. convolution and pooling. Convolution or filter is main part of CNN and it consists of weights that are applied to the input.

In this process, each and every filters are initialize with size of two-dimensional matrix p1*p2 such as number of columns and rows with initialization of weight and bias i.e. W= w1, w2,…,wn and B= b1, b2,…,bn at each layer of CNN for obtaining new characteristics features and it is convolve with input data. The convolution is placed between data and each kernel for producing the new feature map. After the convolution layer, pooling layer generally samples the feature map with different rules of sampling to obtain optimal feature. Pooling layer is use for reducing computational load between first layers. The generated features map is divided into subsampling layer by applying respective function such as max-pooling. The max-pooling mainly decreases, the dimension of features by taking values only from batch of data because of that, redundant features are reduce. The extracted more abstract features are fed to the fully connected layer for achieving desired classification based on intrusion detection. In fully connected layer, all the optimal features that are learn from different convolution kernels are combining to provide classification. The neurons or weights are activated during this layer based on convolution and pooling features are present in the input and produces different activation patterns. Those patterns are compact representation of input to easily classify with softmax activation function. In CNN structure, softmax activation function is used is used for calculating probabilistic distribution of normal or anomalous records and it is normalized exponential function. A softmax activation function is to introduces nonlinearity to CNN and used for finding outputs as normal or anomalous intruder in the network.

Finally, compile CNN classification with different gradient-based optimization methods such as Adagrad, Adadelta, RMSProp, Adam to train the network. After one or multiple convolution and pooling layers, the values of bias and weight are train and tune with different optimization algorithm. There are different hyper parameters such as learning rate, decay rate, initial parameters are used for improving the performance of system and automatically optimized parameters during training process for converging. In this process, the weights and filter values are automatically adjust through training process called as backpropagation. In backpropagation, loss function is defined for minimizing loss or error rate in training. Each time an optimizer finds or calculate gradient and update values of hyperparameters to achieve optimum solution for faster training model. Once the weight is update according to the accumulated results, so the changes should occurs in opposite direction of gradient.

Algorithm 1 shows that, the above process is repeat for fixed number of iteration for training sample of records. The proposed system, suggest the different gradient based optimization algorithm for finding optimum solution and minimize error rate. The comparative analysis is done with Adagrad, Adadelta, RMSProp and Adam algorithms on mini-batch sample size of records.

B. Algorithm 1

1) *Input:* NSL-KDD dataset
2) *Output:* Intruder detection, Error rate
3) *Assumptions:* f: Selected features, t: Threshold, P1: Number of rows in matrix, p2: Number of columns in matrix, m: Matrix of size p1*p2, w (i, j): Weight, bi: Bias, f': Generated features from convolution, e: Error rate.
4) Data cleaning performed on dataset to remove duplicate records.
5) Select features f based on threshold t.
6) while (Process is continue until max iteration criteria)
7) Apply forward-propagation
8) Apply features f to convolution layer.
9) Pretrain the filter, Initialize filter size m: p1*p2.
10) Process training dataset into matrix as filter size.
11) Initialize weight w (i, j) and bias bi
12) To obtain first layer convolve feature matrixX(1) use two-dimensional convolution operation
13) matrixx(1) as input give to pooling layer use MAX-pooling operation to training and Weight update
14) f' features are generated from convolution and max-pooling fed to fully-connected layer
15) Apply softmax activation to generate probabilistic classification of intrusion based on train data
16) Use optimizer functions to derive learning rate for update weight wi and bias bi
17) Error rate is generate from optimizer functions then,
18) Apply back-propagation! Generate second convolution. Repeat step 12-21 to derive feature matrixX(2).
19) End while
20) Merge feature matrixX(3) into column as input of neuron to fully connected layer multiply with weight and bias
21) Then, use softmax activation function to generate probabilistic classification of Intrusion Detection result with achieving minimum error rate.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

IV. EXPERIMENTAL RESULTS

The proposed system is implemented with the use of variant of JAVA module development kit with version 7 and MySQL database. The proposed experiment is performed on NSL-KDD dataset.

Generally, performance of intrusion detection is evaluate with the help of following metrics such as, accuracy, precision, recall and f-score. Performance metric used in intrusion detection using deep learning approach with different gradient descent optimizers for faster training of model and identifying whether the traffic is normal or anomalous. The proposed system performance is depends on two parameters such as precision and recall. A Precision and Recall are defined in terms of a set of retrieved samples of anomalies and a set of relevant samples of anomalies. For retrieved and relevant samples of anomalies all the following rates is calculated.

F1-Score or F- measure: F1-Score is a measure of a test's accuracy. It considers both the precision and recall of the test to compute the score: p is the number of relevant retrieved records divided by the total number of all retrieved records and r is the number of relevant retrieved records divided by the total number of relevant records that should have been returned. The F-score is weighted average of the precision and recall, where an F-score reaches its worst value at 0 and best value at 1. The F-measure or balanced F-score (F1-score) is the harmonic mean of precision and recall:

$$F1 - Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall}$$

Table 1 shows experimental results carried for different optimization algorithms such as, Adagrad, Adadelta, RMSProp and Adam for comparative analysis with same sample records of NSL/KDD dataset. The proposed experiment is performed on set of 5 numbers of random total sample records for finding relevant and retrieved records from that. Comparative analysis is done for gradient descent optimization such as Adagrad, Adadelta, RMSProp and Adam at the time of training process for minimizing error rate in the function and update specified parameter with different learning rate at each and every position.

TABLE I
F1-Score Using Adagrad, Adadelta, Rmsprop And Adam Optimization

Total Number of Samples	F1-Score of Adagrad Algorithm	F1-Score of Adadelta Algorithm	F1-Score of RMSProp Algorithm	F1-Score of Adam Algorithm
1024	0.8716	0.8836	0.9371	0.9594
1200	0.8713	0.8905	0.9387	0.9496
31439	0.8666	0.8883	0.9400	0.9551
46940	0.8706	0.8927	0.9350	0.9494
73589	0.8727	0.8873	0.9367	0.9546

Fig. 2 shows the F1-score values of random sample of 5 anomalies records from NSL/KDD dataset of intrusion detection. In Adagrad optimization algorithm, the highest F1-score value is 0.8727 and lowest is 0.8666 as compared to all. In Adadelta algorithm, the highest and lowest F-measure value is 0.8927 and 0.8836. In RMSProp algorithm, the highest and lowest precision is 0.94 and 0.9350. And last Adam algorithm's F1-score value of highest and lowest is 0.9594 and 0.9494. From the overall comparison of four optimization algorithms is, an Adam gets better F-score than other three algorithms.

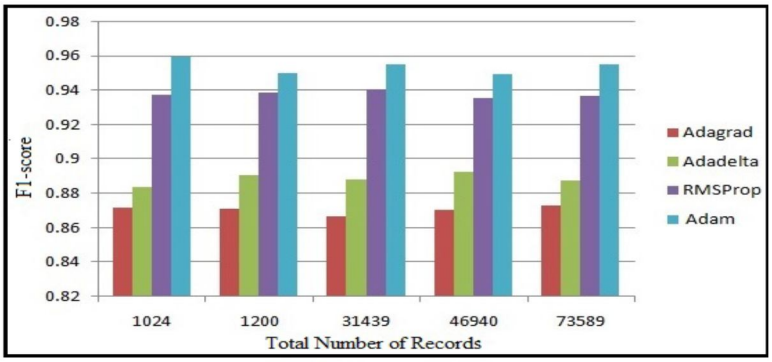


Fig. 2 F1-Score using Adagrad, Adadelta, RMSProp and Adam Optimization Algorithm

The performance of above mention four algorithms is varied according to their tuning hyperparameter. The result is compared on the basis of performance metrics such as precision, recall and F-measure. In that, F1-score values are change according to the precision and recall values are change. Average precision values of Adagrad, Adadelta, RMSProp and Adam is 0.9193, 0.9381, 0.9856, and 0.9968 respectively. Average recall values of same algorithms is 0.8320, 0.8917, 0.9008 and 0.9248 respectively. And then, average f1-score values for same algorithms are 0.8727, 0.8927, 0.9387 and 0.9594 respectively. Result point of view, precision, recall and f1-score values of Adam algorithm is increases as compared to other three algorithms and it gets better performance accuracy in terms of minimizing error rate in the training phase for parameter update. Other three algorithms are suffered from some drawbacks such as it doesn't solve issues with previous gradient descent algorithm which requires manual selection of global learning rate. It is also suffer from one problem i.e. it is continually decaying learning rate throughout training process means that learning rate will become extremely small after so many iterations. the Adam gives much better results than other three algorithms in terms of fast convergence and train the model.

The overall comparative analysis of optimization illustrate, Adam optimizer is best choice for fast training of model and it also deals with large dataset but it takes more time to acquire best results. Adam adds finally bias-correction and momentum to RMSProp towards the end of optimization as gradients become sparser. If our input is sparse or scatter, for achieving best result from the data the adaptive learning rate method is use.

V. CONCLUSION

Some challenges are arises while implementing an effective intrusion detection with unknown attacks is that selection of all features from dataset is difficult for detection. Feature select for one class of attacks may or may not work well for one class of attack because of simultaneously changing behavior of network. Without optimization, deep learning approaches suffers from some issue i.e., high error rate, taking of maximum iteration while detection of intrusion. So, the performance accuracy may gets varies. To avoid such problem, proposed system includes convolutional neural network approach with different gradient-based optimization methods such as, Adagrad, Adadelta, RMSProp and Adam. The experimental results shows, the average of Adagrad optimization algorithm is 0.91, Adadelta is 0.93, RMSProp is 0.98 and Adam is 0.99. The proposed system comparatively evaluate the results in between different optimization algorithm and finally, conclusion is get from that Adam gives much better result than other three because of fast update of parameter with different learning rate for each parameter. Precision, recall and f-measures values are improved using Adam optimization algorithm.

As part of future work, instead of only detecting intruder in system and detection system will identifies disreputable event and allow system's administrator to start exploration with different AI algorithm to learn modern attacks and their behavior. The same work is extended to prevent or to avoid the intruders.

VI. ACKNOWLEDGEMENT

I present my sincere thanks to Prof. Dr. Kishor S. Wani, Principal for moral support and providing excellent infrastructure in carrying out the project work. I am very thankful to Prof. Dr. Girish K. Patnaik, guide and HOD of Computer Engineering for his cooperation and valuable guidance during the work.

I would like to thank all teaching and nonteaching staff in computer department who helped me in my work. I would like to thank my classmates who helped me for project work, also thanks all those people who helped me in anyway what so ever at some point in time. Last but not least, I would like to thank my parents without whose co-operation this won't be possible.

REFERENCES

[1] R. R. Chaudhari and S. P. Patil, “Intrusion detection system: Classification, techniques and datasets to implement”, International Research Journal of Engineering and Technology (IRJET), Volume: 04 Issue: 02, pp. 186066, Feb -2017.

[2] S. Prasad, M. Srinath and M. Basha, “Intrusion detection systems, tools and techniques- an overview”, Indian Journal of Science and Technology, Vol 8(35), pp. 1-7, December 2015.

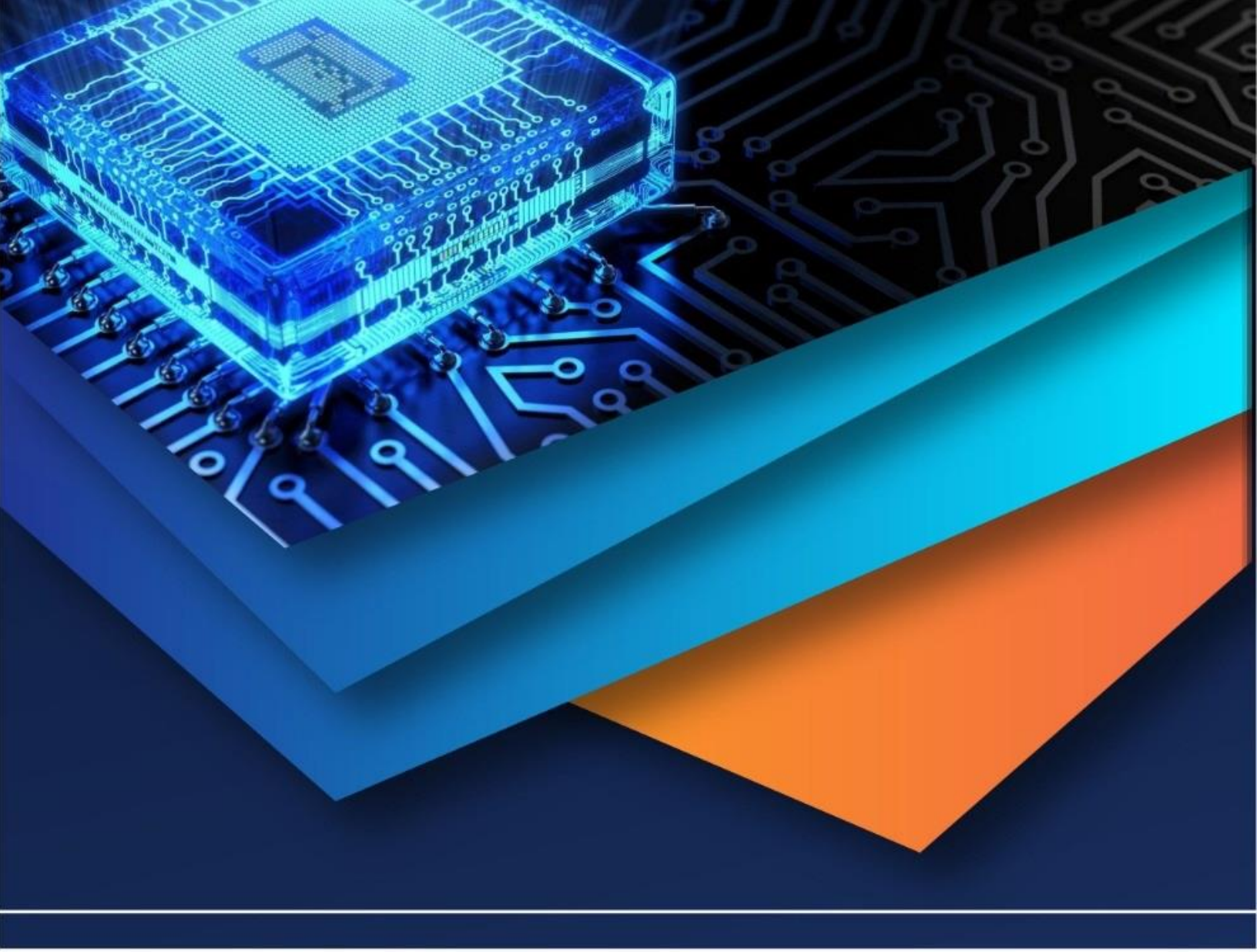
[3] M. E. Aminanto and K. Kim, Deep learning-based feature selection for intrusion detection system in transport layer," Research of Information and Communication Technology and National Research foundation of Korea, Vol. 26, No. 1, pp. 535-538, 2015.

[4] N. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, “Deep learning approach to network intrusion detection”, IEEE Transactions on Emerging Topics in Computational Intelligence, Volume: 2 , Issue: 1, pp. 41-50, Feb. 2018.

[5] Q. Niyaz, W. Sun, A. Y. Javaid and M. Alam, “Deep learning approach for network intrusion detection system”, Bio-inspired Information and communication Technologies, The University of Toledo, 1872, pp. 21-26, Dec. 03-05, 2015.

[6] Y. Yu, J. Long and Z. Cai, “Session-based network intrusion detection using a deep learning architecture”, International Conference on modeling Decisions for Artificial Intelligence, Volume 10571, pp. 144-155, Sept 13, 2017.

[7] A. Jayaswal and R. Nahar, "Detecting network intrusion through a deep learning approach", International Journal of Computer Applications (0975 8887) Volume 180, No.14, pp. 15-19, January 2018.
[8] U. Fiore, F. Palmieri, A. Castiglione and A. D. Santis, "Network anomaly detection with the restricted boltzmann machine", Neurocomputing, vol. 122, pp. 13- 23, Dec. 2013.
[9] T. Aldwairi, D. Pereraa and M. A. Novotnya, "An evaluation of the performance of restricted boltzmann machines as a model for anomaly network intrusion detection", Computer Networks 144, pp. 111-119, 2018.
[10] S. Zhai, Y. Cheng, W. Lu and Z. M. Zheng, "Deep structured energy based models for anomaly detection", Proceedings of the 33 rd International Conference on Machine Learning, New York, NY, USA, pp. 1-10, 2016.
[11] M. Salama, H. Eid, R. Ramadan and A. Darwish, "Hybrid intelligent intrusion detection scheme", Soft Computing in Industrial Applications, Springer, Berlin Heidelberg, 2011, volume 96, pp. 1-11, 2011.
[12] S. Potluri and C. Diedrich, "Deep feature extraction for multi-class intrusion detection in industrial control systems", International Journal of Computer Theory and Engineering, Vol. 9, No. 5, pp. 374-379, October 2017.
[13] BaoyiWang, S. Sun and S. Zhang, "Research on feature selection method of intrusion detection based on deep belief network", 3rd International Conference on Machinery, Materials and Information Technology Applications (ICMMITA 2015), pp. 556-561, 2015.
[14] M. J. Kang and J. W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security", Published in PloS one 2016 DOI:10.1371/journal.pone.0155781, pp. 1-17, June 7, 2016.
[15] C. Yin, Y. Zhu, J. Fei and X. He, "A deep learning approach for intrusion detection using recurrent neural networks", IEEE Access, Volume 5, pp. 21 954-21961, 12 October 2017.
[16] J. Kim and H. Kim, "Applying recurrent neural network to intrusion detection with hessian free optimization", International workshop on Information security applications2015, pp. 357-369, 2015.
[17] M.Ponkarthika and Dr.V.R.Saraswathy, "Network intrusion detection using deep neural networks", Asian Journal of Applied Science and Technology (AJAST) (Open Access Quarterly International Journal) Volume 2, Issue 2, pp. 665-673, April-June 2018.
[18] J. Kim, J. Kim, H. L. T. Thu and H. Kim, "Long short term memory recurrent neural network classifier for intrusion detection", International Conference on Platform Technology and Service, 2015, pp. 411-420, 2015.
[19] Y. LeCun, Y. Bengio and G. Hinton, Deep learning, Nature, vol. 521, no. 7553, pp. 436-444, May 2015.
[20] Y. Liu, S. Liu and X. Zhao, "Intrusion detection algorithm based on convolutional neural network", 4th International Conference on Engineering Technology and Application (ICETA 2017), pp. 9-13, 2017.
[21] L. Mohammadpour, T. C. Ling, C. S. Liew, and C. Y. Chong, "A convolutional neural network for network intrusion detection system", Proceedings of the APAN Research Workshop 2018, pp. 50-55, 2018.
[22] L.Dhanabal and D. S. Shantharajah, "A study on nsl-kdd dataset for intrusion detection system based on classi_cation algorithms", International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 6, pp. 446-452, June 2015.
[23] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set", Proceedings of the 2009 IEEE Symposysum Computational Intelligence in Security and Defense Application, pp. 1-6, Jul. 2009.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY