

Performance analysis of intrusion detection for deep learning model based on CSE-CIC-IDS2018 dataset

Baraa Ismael Farhan, Ammar D. Jasim

Department of Information and Communication Engineering, College of Information Engineering, Al-Nahrain University, Baghdad, Iraq

Article Info

Article history:

Received Jan 22, 2021

Revised Mar 10, 2022

Accepted Mar 24, 2022

Keywords:

CSE-CIC-IDS2018

Deep learning

Internet of thing

Intrusion detection

Long short-term memory

ABSTRACT

The evolution of the internet of things as a promising and modern technology has facilitated daily life. Its emergence was accompanied by challenges represented by its frequent exposure to attacks and its being a target for intruders who exploit the gaps in this technology in terms of the nature of its heterogeneous data and its large quantity. This made the study of cyber security an urgent necessity to monitor infrastructures. It has network flaw detection and intrusion detection that helps protect the network by detecting attacks early and preventing them. As a result of advances in machine learning techniques, especially deep learning and its ability to self-learning and feature extraction with high accuracy, the research exploits deep learning to analyze the real data set of CSE-CIC-IDS2018 network traffic, which includes normal behavior and attacks, and evaluate our deep model long short-term memory (LSTM), That achieves accuracy of detection up to 99%.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Baraa Ismael Farhan

Department of Information and Communication Engineering, College of Information Engineering

Al-Nahrain University

Jadriya District, Baghdad, Iraq

Email: bfarhan@uowasit.edu.iq

1. INTRODUCTION

The technological revolution that resulted from the emergence of the Internet of things and the wide growth in its applications and uses, and the difficulties and challenges that accompanied this technology of heterogeneity, privacy and control of data security [1]–[3]. These challenges led to the necessity of network monitoring and intrusion detection, network intrusion detection system (NIDS) plays a key part in detecting attacks and the challenges in security. It monitors and distinguishes suspicious activities and analyzes traffic into normal and malicious. It also detects security violations, intrusion and anomalies [4]–[6]. This NIDS system aims to propose an infrastructure capable of detecting vulnerabilities and warning them in a smart, secure and reliable manner, unlike the firewall, which acts as protection only by allowing only authenticated networks to pass through [7].

Methods for detecting attacks in networks include three ways: "signature-based detection", That matches the signature of the known attack with the current traffic, "anomaly-based detection", Depends on visualizing a normal or legitimate profile obtained under normal network conditions without attacks, and comparing the network's actions with it for identify anomalies and "specification-based detection", This type depends on matching the predetermined and memorized specification with the criteria or specification to detect a certain programmer's operation and notify any violation of such criteria [8]–[10].

The protection and prevent unauthorized access for the systems can achieve by using Firewalls and authentication methods. But these methods lost the ability to monitor in NIDS the network traffic. While, the

intrusion detection system in the network monitors the incoming and outgoing flows to it [11], [12]. Most of the previous work on the topic of NIDS used old simulation-based data sets for experiments such as KDDcup99 or NSL-KDD, which does not represent real data nor reflective of real network traffic scenarios [13].

The choice of the type of database used to extract the information is of great importance as it supports the work of the model used in the detection. As it is important to design a model that adopts an effective algorithm for extracting features, so we use deep learning algorithms, which are better than machine learning methods because they extract features automatically and not manually, which gives high accuracy and detection speed, especially in the field of big data [14], [15]. Because of the importance of using real datasets that reflect network traffic and associated intrusions to ensure accurate evaluation of models better than old data sets. In this paper, the real data set from the Amazon web services (AWS) platform was used, where CSE-CIC-IDS2018 represents dynamic, modifiable, repeatable, scalable data, and a deep learning model was designed using long short-term memory (LSTM), which is one of the best deep learning models in dealing with prediction and obtaining an accuracy rate High in detection up to 99%.

2. RELATED WORKS

Due to the importance of the topic of network traffic analysis and verification, many recently published studies have addressed this topic. In this section the most important works that use deep learning for intrusion detection in network NIDS are presented. Algorithms show different performance in terms of detecting different attacks, and they work well with some attacks, while being poor with others. Ferrag *et al.* [16] presented compared different deep learning (DL) techniques including: "deep neural network", "recurrent neural network", "constrained Boltzmann machine", "deep belief networks", "convolutional neural networks", "deep Boltzmann machine", and "deep automatic coding". It was applied to two real datasets (CIC-IDS 2018, BoT-IoT) covering the latest attacks and showed that recurrent neural network (RNN) scored the highest detection rate for seven of the attacks which are "Brute force cross-site scripting (XSS)", "Brute force-web", "A denial-of-service (DoS) attack Hulk", "DoS slow hypertext transfer protocol (HTTP) test", "DoS attack slowloris attack", "DoS attack Goldeneye". While the network recorded convolutional neural the High detection rates for the remaining four types of attacks, "namely DDoS HOIC attack", "DDoS LOIC-UDP attack" and "Botnet". But it used a small percentage of the large data volume.

Karatas *et al.* [17] presented the analysis for six intrusion detection systems using machine learning: "Adaboost", "Decision tree", "Gradient boosting", "Random forest", "K nearest neighbor", and "Linear Discriminant" Analysis algorithm. With the use of a modern dataset instead of the old data, and an attempt to reduce the imbalance rate in the data, which increases the detection rate. Using model synthetic minority oversampling technique (SMOTE).

Lin *et al.* [13] viewed a proposed system to detect anomalies using LSTM long-term memory and Attention Mechanism (AM) to increase network training performance. The CIC-IDS 2018 data set has been used to train the proposed form and the results analysis has been mentioned the accuracy as 96.22% and detection rate 15% and recall rate 96%. Kanimozhi and Jacob [15] presented a proposed system classifying a bot attack in banking transactions. And it was applied to the CIC-IDS 2018 data set. It was proposed to use several methods combined with each other and with artificial intelligence, and reliability charts were used to verify the expected possibilities of the items.

Zhou and Pazaros [18] presented six methods of deep learning were applied to the CIC-AWS-2018 dataset to detect attacks and classify Zero-Day attacks, as this data contains eight types of attacks and fourteen types of breaches. Recorded an intrusion detection rate of 100%, a zero-day intrusion accuracy rate of 96%, and a 5% false-positive rate. Basnet *et al.* [11] presented the capabilities of detecting breaches in deep learning algorithms were presented by comparing a set of deep learning implementation methods to detect attacks and breaches and categorizing them as PyTorch, Keras, TensorFlow, and Theano and fast.ai. Kim *et al.* [19] have suggested A convolutional neural network (CNN) model, which converts data into images and executes them on the CIC-2018 dataset. The image is classified so that the size of each image is 13×6 where this set contains 78 features.

3. METHOD

This part introduces the design of a model for network intrusion detection that using deep learning technique, which overcomes the high dimensions of network traffic content. To increase the effectiveness of detection, use a system based on anomaly detection and classification of various attacks, and the system was implemented on a real data set that includes all attacks, which is the CSE-CIC-IDS2018, where the data is initialized by implementing standard procedures on it, including (pre-

processing of data, selecting features, training the model, and evaluating model performance). Pre-processing involves collecting and arranging the data, deleting all unnecessary features, then we perform a normalization of all data in [-1,1] followed by implementing intrusion detection system using deep learning model LSTM, as shown in Figure 1.

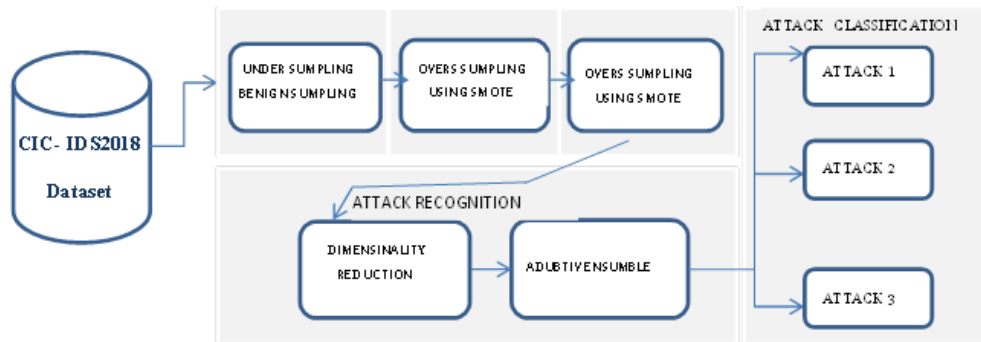


Figure 1. The preprocessing for intrusion detection

3.1. Real dataset (CSE-CIC-IDS2018)

In this part, the type of data used to implement intrusion detection is explained. Which represents real and dynamic data taken from Amazon platform AWS by Communications Security Corporation (CSE) and Canadian Cybersecurity Institute (CIC) and represents real-time network traffic [20]. It is considered one of the most reliable data for evaluating intrusion detection models based on network anomalies [21]. This data contains the latest attacks and includes ten classes of attacks as shown in Figure 2 as columns and arranged according to the percentage of detection in the data: Benign, Bot, FTP-BruteForce, SSH-Bruteforce, DDOS attack-HOIC, DDOS attack-LOIC-UDP, DoS attacks-GoldenEye, DoS attacks-Slow HTTP test, intrusion, and web attacks [22]. Table 1 also shows the number of each attack class and its percentage of the original data volume. The attack infrastructure also includes 50 devices, and the victim organization contains 30 servers, 420 terminals, and 5 sections [23]. This data contains 80 features extracted using the CICFlowMeter-V3 tool [24]. And in Table 2, a set of features extracted from the traffic are shown.

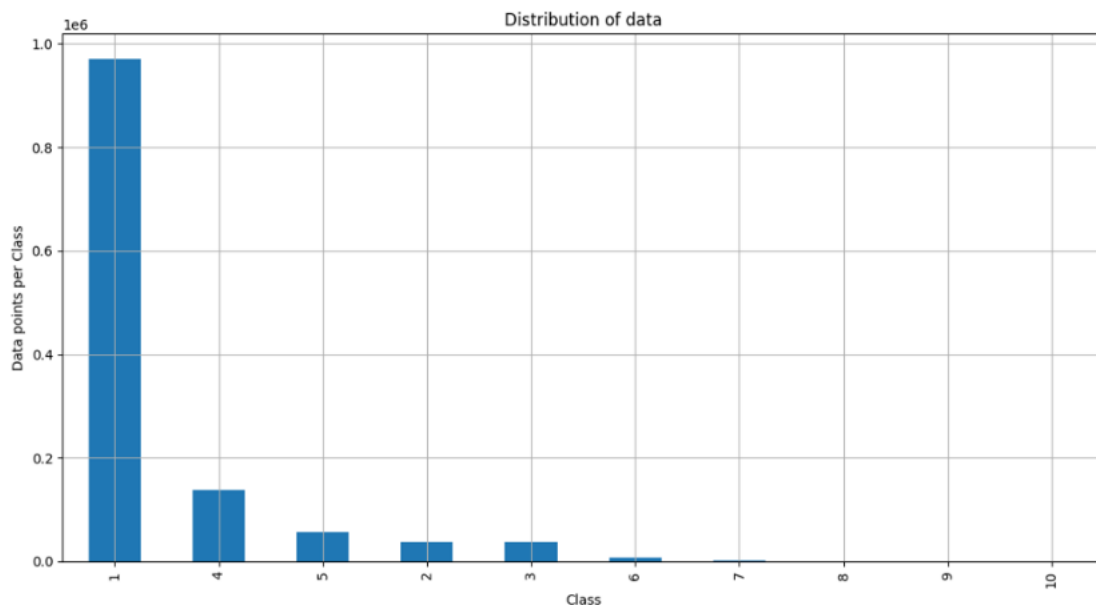


Figure 2. Distribution of the attack classes in CIC-IDS 2018 dataset

Table 2. Volume of data points in attack class and ratio of it

| Class number | Attack Class | Volume of data points in class | Ratio from the original data (1252835 row) |
|--------------|--------------------------|--------------------------------|--|
| 1 | Benign | 971016 | 77.505 % |
| 2 | Infiltration | 38703 | 3.089 % |
| 3 | DoS attacks-Hulk | 37323 | 2.979 % |
| 4 | Bot | 137185 | 10.95 % |
| 5 | DDOS attack-HOIC | 57507 | 4.59 % |
| 6 | DDOS attack-LOIC-UDP | 8377 | 0.669 % |
| 7 | FTP-BruteForce | 2234 | 0.178 % |
| 8 | DoS attacks-GoldenEye | 332 | 0.026 % |
| 9 | DoS attacks-SlowHTTPTest | 103 | 0.008 % |
| 10 | SSH-Bruteforce | 55 | 0.004 % |

Table 2. Sample from CIC-IDS 2018 dataset features

| Feature name | Description of feature |
|---------------|--|
| down_up_ratio | Download and upload ratio |
| Fl/dur | Flow duration |
| fw/pkt/avg | Average size of packet in forward direction |
| fw/act/pkt | Number of packets have transmission control protocol (TCP) data payload at least 1 byte in forward |
| fw/pkt/std | Standard deviation size of the packet in forward |
| tot/bw/pk | Total packets in the backward direction |
| tot/fw/pk | Total packets in the forward direction |
| Pkt/ len/var | Mini inter-arrival time of packet |
| bw/pkt/max | Max size of packet/backward |
| bw/pkt/min | Min size of packet/backward |
| fw/win/byt | The Number of bytes that send in initial window/forward |
| bw/win/byt | The Number of bytes that send in initial window/backward |
| bw/hdr/len | The total bytes that use in headers/ backward |
| Fw/hdr/len | The total bytes that use in headers/forward |

3.2. Pre-processing on dataset

The original dataset contained 80 features. And there are some features that have little effect on interpreting the behavior of data and traffic, whether it is normal or not. Therefore, these features such as the timestamp feature and internet protocol (IP) addresses that do not help in training the neuron to detect errors and intrusions are deleted, so we use 78 features from the original number of features. Then, we divide the data set into a training set that includes 70%, and a test set that includes 30% of the original data.

3.3. Long short-term memory (LSTM)

Deep learning is a powerful method for making accurate detection and prediction of large and complex data such as videos, images, and texts [25]. Therefore, we use the capabilities of deep learning, represented by the use of multiple processing layers that teach data in multiple hidden layers [26]. Which contribute to increasing the accuracy and reducing the cost in the detection of attacks and malware [27]. The survey on techniques used in cyber security in intrusion detection [28], which described DL types and their way of working, showed their superiority over traditional machine learning methods, being able to extract features automatically instead of the method of engineering features in machine learning (ML).

LSTM is one of the most important types of deep learning used with sequential data [29], as it is able to know the current traffic and the previous traffic of the network. Because the attackers carry out the attack as a series of continuous processes, it is important to know the current and past traffic. And it helps to resolve the issue of long-term reliance [30] and is considered a development on the RNN, by adding the forget gateway, input gateway and output gateway on the RNN model. Here, the LSTM model was used with the network traffic data as it is generally considered sequential and to take advantage of the capabilities of the LSTM in dealing with the sequential data well in practice.

3.4. Experimental environment

Our LSTM model implemented by Visual code program 2019 contain python version 3.9. That use Tensorflow [31] which includes libraries (Panda, Scikit-learn, Numpy) and Keras [32] in Windows 10 environment. Using hardware includes CPU core i7, 4 GB memory and hard disk capacity 512 GB.

4. RESULTS AND DISCUSSION

In this part, the experimental results of the LSTM model are presented. Then we present the evaluation of these results with the main measures, then a comparison is made between the LSTM model and other models.

4.1. Results

In this section we review the hyperparameters that must be set to avoid overfitting, as shown in the Table 3. The LSTM model contains three layers, the first layer includes 78 neurons, while the second layer contains 64 neurons, and both layers use the same ReLu activation function. The third layer contains 8 neurons and uses the Softmax activation function. These two functions represent non-linear activation functions, which are faster and more accurate than linear activation functions. The loss function, which represents the difference between the actual and expected output values, was calculated. And to reduce the loss function we use optimizer Adam by calculating the loss gradients to update the values and improve the model results.

Table 3. Hyperparameter of proposed LSTM model

| Parameters Name | Value |
|----------------------|--------------------------|
| Hidden nodes in LSTM | 150 |
| Batch size | 200 |
| Epoch | 30 |
| The length of flow | 10 |
| Learning rate | 0.001 |
| Loss function | categorical_crossentropy |
| Activation function | Relu, Soft max |
| Optimizer | Adam |

4.2. Used metrics

To evaluate the performance of the model, in this paper used three scales, namely, the accuracy scale and the loss scale. Accuracy is a representation of the ability to classify samples correctly as (1).

$$Accuracy = \frac{true\ positive + true\ negative}{true\ positive + false\ negative + true\ negative + false\ negative} \tag{1}$$

Accuracy represents the proportion of samples that are properly classified. Accuracy is inversely proportional to the false alarm rate (FAR). The higher the accuracy, the lower the false alarm rate, Figure 3 shows the accuracy measurement in the training and testing phases. The loss function is the variation between the expected and actual output, Figure 4 shows the loss measurement in the training and testing phases. The confusion matrix is a graphical representation that summarizes the performance and accuracy of the classification process, illustrating true and false positive values, and gives an idea of the errors in which the model occurs and an idea of how to correct them. Predict natural and attacking packets in network traffic, Table 4 shows confusion matrix.

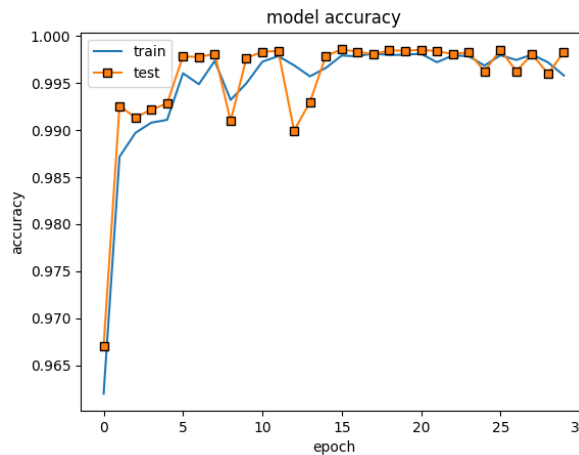


Figure 3. The accuracy of train stage and test stage

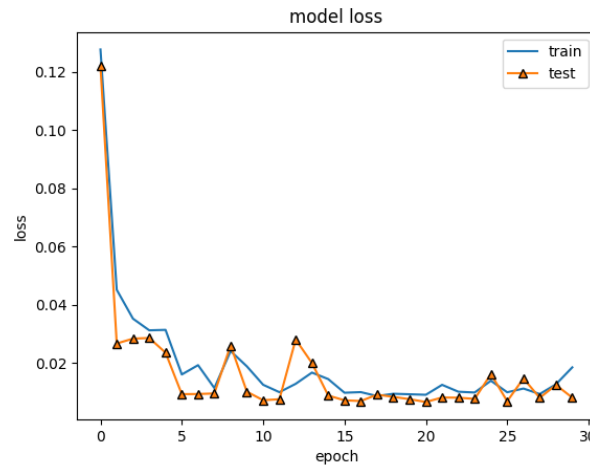


Figure 4. Loss of train stage and test stage

Table 4. Confusion matrix

| | Benign | Info | DOS-Hulk | Bot | DDOS - HOIC | DDOS-LOIC | FTP-Bruteforce | Dos-GoldenEye | Dos-Slow HTTP Test | SSH-Bruteforce |
|--------------------|--------|------|----------|-------|-------------|-----------|----------------|---------------|--------------------|----------------|
| Benign | 339502 | 1 | 1 | 4 | 124 | 190 | 22 | 12 | 0 | 0 |
| Info | 6 | 11 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 |
| DOS-Hulk | 0 | 0 | 13546 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bot | 7 | 0 | 4 | 13052 | 0 | 0 | 0 | 0 | 0 | 0 |
| DDOS - HOIC | 23 | 0 | 0 | 0 | 47992 | 0 | 0 | 0 | 0 | 0 |
| DDOS - LOIC | 96 | 0 | 0 | 0 | 0 | 20032 | 0 | 0 | 0 | 0 |
| FTP-Bruteforce | 13 | 0 | 0 | 0 | 0 | 0 | 2919 | 0 | 0 | 0 |
| Dos-GoldenEye | 22 | 0 | 0 | 0 | 0 | 0 | 2 | 758 | 0 | 0 |
| Dos-Slow HTTP Test | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 116 | 0 |
| SSH-Bruteforce | 24 | 9 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 0 |

4.3. Comparative analysis

In this part, the difference between our LSTM model and other models is presented. As shown in the Table 5. The proposed model in [13] detects and classifies bot attacks that pose a threat to banking transactions and use the dataset CSE -CIC-IDS 2018. Seth *et al.* [7] suggested identifying different types of attacks by ranking the detection ability of the classifiers and building an ensemble. Rios *et al.* [33] suggested the use of a broad learning system (BLS) that achieves good performance in less training time to detect cyber-denial of service attacks in telecommunication networks.

Table 5. The comparison among LSTM model and another method

| Research | DL and ML | Data set | Accuracy |
|-------------------------|------------------|------------------|----------|
| Lin <i>et al.</i> [13] | LSTM +AM | CSE-CIC-IDS 2018 | 96.2% |
| Seth <i>et al.</i> [7] | Light GBM + HBGB | CSE-CIC-IDS 2018 | 97.5% |
| Rios <i>et al.</i> [33] | CFBLS and BLS | CSE-CIC-IDS 2018 | 97.46% |
| The proposed LSTM | LSTM | CSE-CIC-IDS 2018 | 99% |

5. CONCLUSION

In this paper, a system for detecting intrusion in the network is proposed using deep learning technology. Where LSTM method was used to build the neural network that applied to CSE-CIC-IDS2018 real data set to detect intrusion during data flow. The accuracy of the detection of the model was equal to 99%, which is a good accuracy, but there are problems and challenges represented by the imbalance in the

CSE-CIC-IDS2018 dataset and its large size which may cause an fault of accuracy computing. As well as difficulties of designing the LSTM model, In terms of increasing the nodes and linking between the multiple layers. Looking forward, we plan to increase accuracy, reduce error, and speed up the training process by using methods to identify the most relevant features that support detection method.

ACKNOWLEDGEMENTS

Special thanks to my professors at Al-Nahrain University and thanks to Wasit University. This work is not supported by any party.




REFERENCES

- [1] G. A. H. Alawadi, H. N. A. Ali, and B. I. Farhan, "Usage of iot and cot based environment for telemedicine in health care," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 11, no. 2, pp. 1446–1453, 2019.
- [2] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, "A survey of machine and deep learning methods for internet of things (IoT) security," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1646–1685, 2020, doi: 10.1109/COMST.2020.2988293.
- [3] S. Aldhaheri, D. Alghazzawi, L. Cheng, B. Alzahrani, and A. Al-Barakati, "DeepDCA: novel network-based detection of IoT attacks using artificial immune system," *Applied Sciences*, vol. 10, no. 6, p. 1909, Mar. 2020, doi: 10.3390/app10061909.
- [4] M. Verkerken, L. D'hooge, T. Wauters, B. Volckaert, and F. De Turck, "Towards model generalization for intrusion detection: unsupervised machine learning techniques," *Journal of Network and Systems Management*, vol. 30, no. 1, p. 12, Jan. 2022, doi: 10.1007/s10922-021-09615-7.
- [5] A. Singh, J. Nagar, S. Sharma, and V. Kotiyal, "A Gaussian process regression approach to predict the k-barrier coverage probability for intrusion detection in wireless sensor networks," *Expert Systems with Applications*, vol. 172, p. 114603, Jun. 2021, doi: 10.1016/j.eswa.2021.114603.
- [6] A. Singh, J. Amutha, J. Nagar, S. Sharma, and C.-C. Lee, "LT-FS-ID: log-transformed feature learning and feature-scaling-based machine learning algorithms to predict the k-barriers for intrusion detection using wireless sensor network," *Sensors*, vol. 22, no. 3, p. 1070, Jan. 2022, doi: 10.3390/s22031070.
- [7] S. Seth, K. K. Chahal, and G. Singh, "A novel ensemble framework for an intelligent intrusion detection system," *IEEE Access*, vol. 9, pp. 138451–138467, 2021, doi: 10.1109/ACCESS.2021.3116219.
- [8] Z. Li, A. L. G. Rios, and L. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 7, pp. 2254–2264, Jul. 2021, doi: 10.1109/JSAC.2021.3078497.
- [9] B. I. Farhan and A. D. Jasim, "A survey of intrusion detection using deep learning in internet of things," *Iraqi Journal for Computer Science and Mathematics*, vol. 3, no. 1, pp. 83–93, Jan. 2022, doi: 10.52866/ijcsm.2022.01.01.009.
- [10] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection," *ACM Computing Surveys*, vol. 54, no. 2, pp. 1–38, Mar. 2022, doi: 10.1145/3439950.
- [11] R. B. Basnet, R. Shash, C. Johnson, L. Walgren, and T. Doleck, "Towards detecting and classifying network intrusion traffic using deep learning frameworks," *Journal of Internet Services and Information Security*, vol. 9, no. 4, pp. 1–17, 2019, doi: 10.22667/JISIS.2019.11.30.001.
- [12] S. Zavrak and M. Iskefiyeli, "Anomaly-based intrusion detection from network flow features using variational autoencoder," *IEEE Access*, vol. 8, pp. 108346–108358, 2020, doi: 10.1109/ACCESS.2020.3001350.
- [13] P. Lin, K. Ye, and C.-Z. Xu, "Dynamic network anomaly detection system by using deep learning techniques," in *Cloud Computing – CLOUD 2019*, vol. 11513 LNCS, 2019, pp. 161–176.
- [14] J. L. Leevy and T. M. Khoshgoftaar, "A survey and analysis of intrusion detection models based on CSE-CIC-IDS2018 Big Data," *Journal of Big Data*, vol. 7, no. 1, p. 104, Dec. 2020, doi: 10.1186/s40537-020-00382-x.
- [15] V. Kanimozhi and T. P. Jacob, "Artificial Intelligence outflanks all other machine learning classifiers in network intrusion detection system on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing," *ICT Express*, vol. 7, no. 3, pp. 366–370, Sep. 2021, doi: 10.1016/j.ict.2020.12.004.
- [16] M. A. Ferrag, L. Maglaras, S. Moschogiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *Journal of Information Security and Applications*, vol. 50, p. 102419, Feb. 2020, doi: 10.1016/j.jisa.2019.102419.
- [17] G. Karatas, O. Demir, and O. K. Sahingoz, "Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset," *IEEE Access*, vol. 8, pp. 32150–32162, 2020, doi: 10.1109/ACCESS.2020.2973219.
- [18] Q. Zhou and D. Pezaros, "Evaluation of machine learning classifiers for zero-day intrusion detection - an analysis on CIC-AWS-2018 dataset," May 2019, *arXiv: 1905.03685*.
- [19] J. Kim, Y. Shin, and E. Choi, "An intrusion detection model based on a convolutional neural network," *Journal of Multimedia Information System*, vol. 6, no. 4, pp. 165–172, Dec. 2019, doi: 10.33851/JMIS.2019.6.4.165.
- [20] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, p. 107247, Jun. 2020, doi: 10.1016/j.comnet.2020.107247.
- [21] R. I. Farhan, A. T. Malood, and N. F. Hassan, "Optimized deep learning with binary PSO for intrusion detection on CSE-CIC-IDS2018 dataset," *Journal of Al-Qadisiyah for Computer Science and Mathematics*, vol. 12, no. 3, pp. 16–27, 2020, doi: <https://doi.org/10.29304/jqcm.2020.12.3.706>.
- [22] "Registry of open data on AWS." <https://registry.opendata.aws/cse-cic-ids2021/> (accessed May 30, 2020).
- [23] Canadian Institute for Cybersecurity (CIC), "CSE-CIC-IDS2018 on AWS." <https://www.unb.ca/cic/datasets/ids-2018.html> (accessed May 30, 2020).
- [24] M. K. Ibraheem, I. M. A. Al-Khafaji, and S. A. Dheyab, "Network intrusion detection using deep learning based on dimensionality reduction," *REVISTA AUS*, vol. 26, no. 2, pp. 168–174, 2019.
- [25] S. Lu *et al.*, "New era of deeplearning-based malware intrusion detection: The malware detection and prediction based on deep learning," Jul. 2019, *arXiv: 1907.08356*.
- [26] H. S. Abdulkarem and A. D. Alethawy, "DDOS attack detection and mitigation at sdn enviroment," *Iraqi Journal of Information and Communications Technology*, vol. 4, no. 1, pp. 1–9, 2021.




- [27] S. Mahdaviifar and A. A. Ghorbani, "Application of deep learning to cybersecurity: A survey," *Neurocomputing*, vol. 347, pp. 149–176, Jun. 2019, doi: 10.1016/j.neucom.2019.02.056.
- [28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
- [29] M. Zhu, K. Ye, Y. Wang, and C.-Z. Xu, "A deep learning approach for network anomaly detection based on AMF-LSTM," in *Network and Parallel Computin*, vol. 11276 LNCS, 2018, pp. 137–141.
- [30] N. Chockwanich and V. Visoottiviset, "Intrusion detection by deep learning with TensorFlow," in *2019 21st International Conference on Advanced Communication Technology (ICACT)*, Feb. 2019, pp. 654–659, doi: 10.23919/ICACT.2019.8701969.
- [31] Keras Documentation, 2019, [Online]. Available at: <https://keras.io/>
- [32] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334.
- [33] A. L. G. Rios, Z. Li, K. Bekshentayeva, and L. Trajkovic, "Detection of denial of service attacks in communication networks," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, Oct. 2020, pp. 1–5, doi: 10.1109/ISCAS45731.2020.9180445.

BIOGRAPHIES OF AUTHORS



Baraa Ismael Farhan    is lecture at college of Computer Science & Information Technology, Wasit University, Iraq. She Holds a M.Tech degree in Computer Engineering with specialization in CSE from Osmania University, Heyderabad, India in 2014. Her 10 researchs areas are Networks, Security, and Artificial Intelligence. She is Ph.D. student in the Information and Communication Engineering Department, University of Al-Nahrain University, Baghdad, Iraq since Septemper 2019. She can be contacted at email: bfarhan@uowasit.edu.iq.



Assist. Prof. Dr. Ammar D. Jasim    is Assistant Professor at college of Information Engineering, Al-Nahrain University, Iraq. He Holds a PhD degree in Information Engineering with specialization in networks. His research areas are Networks, Communications, Security, and Artificial Intelligence. He is head of information system department in Al-Nahrain University, Iraq. He is supervisor for a number of students in Ph.D. He can be contacted at email: ammar@coie.nahrain.edu.iq.