e-Prime - Advances in Electrical Engineering, Electronics and Energy 9 (2024) 100673

Contents lists available at ScienceDirect

e-Prime - Advances in Electrical Engineering, Electronics and Energy

journal homepage: www.elsevier.com/locate/prime

Two-step data clustering for improved intrusion detection system using CICIoT2023 dataset

Hadeel Q. Gheni [a,*], Wathiq L. Al-Yaseen [b]

[a] Department of Software, Information Technology College, University of Babylon, Babylon, Iraq
[b] Karbala Technical Institute, Al-Furat Al-Awsat Technical University, 56001 Karbala, Iraq

ARTICLE INFO

Keywords:
Intrusion detection
GSK
CICIoT2023

ABSTRACT

The issue of network security is an important and delicate issue when it comes to the privacy of organizations and individuals, especially when important and sensitive information is transmitted across these networks. The importance of intrusion detection systems, which is a very important component of protecting information and reducing the damage resulting from attacks and penetrations of network systems, has increased due to the adoption of the most recent regulations on advanced web services, whether government services, banking services, e-mail, or e-marketing. The goal of this paper is to construct an intrusion detection system using deep learning algorithms based on a new dataset named the CICIoT2023. The proposed intrusion detection model addresses challenges associated with intrusion detection datasets in terms of high dimensionality by adopting new methods to reduce their size and improve efficiency. A new clustering technique for intrusion detection datasets based on a new method combination between an optimization algorithm and static tools was proposed. The proposed model was evaluated to determine its efficiency using several evaluation measures. The results show that in comparison to earlier research conducted on the same datasets, the suggested model performs better in attack detection. As a result, the proposed model offers a high level of network security trust.

1. Introduction

As there has been a rise in data breaches and leaks in recent years, there could be significant challenges and controversy if this data ends up in the wrong hands because personal data can be used and controlled without authorization [1]. However, by utilizing security flaws, they have also become more accessible for attackers to access and utilize for illicit reasons. People are now more interested in safeguarding and securing these systems as a result.

An intrusion is accessing and entering the network or a particular computer to spy on information, steal data, change the system, or obtain security holes in the operating system to sabotage and destroy it [2]. Because of these things, using a system to find intrusions and lower network and computer security risks was essential [3]. Due to the unstable behavior of the intrusion, the system is unable to predict the network traffic correctly, which exposes the security requirements, which are availability, integrity, and confidentiality [4], therefore, the importance of developing an advanced intrusion detection system (IDS) has increased.

The goal of this paper is to make network security systems better in general by developing an accurate and efficient intrusion detection system. First, the selected dataset was preprocessed, and then several metrics were applied to reduce the number of features involved in building the neural network classifier. Second, reduce the dimensionality of the intrusion detection dataset without losing the important data by employing an optimization algorithm called the Gaining–sharing knowledge (GSK) algorithm.

The rest of this paper is as follows: Section 2 relates to the previous studies. Section 3 relates to the materials and methods. Section 4 relates to the proposed model and the performance assessments. Section 5 relates to the conclusion and the future works.

2. Related studies

The field of network security, including intrusion detection systems, has always been a source of interest for researchers around the world. In this paper, the latest intrusion detection dataset named CICIoT2023 was used. A small number of researches have been conducted on this dataset and will be mentioned in this section.

In [5], Deep neural networks and bidirectional long short-term

* Corresponding author.
E-mail address: wsci.hadeel.qasem@uobabylon.edu.iq (H.Q. Gheni).

https://doi.org/10.1016/j.prime.2024.100673
Received 11 March 2024; Accepted 1 July 2024
Available online 5 July 2024

Table 1
Details of related studies.

Ref	Dataset	Year	Technique
5	CIC-IoT-2023	2023	BiLSTM
6	CIC-IoT-2023	2023	Random Forest, Catboost gradient boosting, MLP-Prod
7	CIC-IoT-2023	2023	DNN, CNN, RNN
8	CIC-IoT-2023	2023	KNN, Weighted KNN, RF, MLP, DT

Table 2
Attack types in CICIoT2023 Dataset [12].

Class	Sub-Class
DoS	DoS-UDP_Flood, DoS-SYN_Flood, DoS-TCP_Flood, DoS-HTTP_Flood
DDoS	DDoS-HTTP_Flood, DDoS-SlowLoris, DDoS-ICMP_Fragmentation, DDoS-UDP_Fragmentation, DDoS-ACK_Fragmentation, DDoS-SynonymousIP_Flood, DDoS-ICMP_Flood, DDoS-TCP_Flood, DDoS-UDP_Flood, DDoS-SYN_Flood, DDoS-PSHACK_Flood, DDoS-RSTFINFlood.
Web-based	CommandInjection, SqlInjection, Uploading_Attack, XSS, Backdoor_Malware, BrowserHijacking,
Recon	Recon-HostDiscovery, VulnerabilityScan, Recon-PortScan, Recon-OSScan, Recon-PingSweep
Spoofing	MITM-ArpSpoofing, DNS_Spoofing,
Brute force	DictionaryBruteForce,
Mirai	Mirai-greip_flood, Mirai-udpplain, Mirai-greeth_flood

Table 3
Primary classes and their representation for the CICIoT2023 dataset.

Classes	No. Patterns	Per.%
Normal	5400	2.318 %
DOS	40,492	**17.387** %
DDoS	169,276	**72.686** %
Web-based	**125**	0.053 %
Recon	1742	0.748 %
Spoofing	2473	1.061 %
Brute force	**55**	0.023 %
Mirai	13,322	**5.720** %
Total	**232,885**	**100 %**

memory networks are used by the DL-BiLSTM lightweight IoT intrusion detection model to extract complicated network information, improve detection performance, and handle resource restrictions in the CICIoT2023 dataset. For feature selection, it makes use of the incremental principal component analysis algorithm.

In [6], the authors use the CIC-IoT-2023 dataset to identify unusual behavior in IoT networks and investigate the impact of adversarial attacks based on data leaks on machine learning models implemented in cloud applications.

In [7], the authors suggested several deep learning model variations to identify cyberattacks in practical Internet of Things scenarios using different network stream packet samples by employing the robust scalar approach for data preprocessing.

In [8], the authors adopted a "Less is More" approach to increase detection rates. Through the use of Random Forest feature selection, the interpacket arrival time (IAT) emerged as the primary factor. By focusing on this one feature, the dimensions of the data were reduced. Table 1. illustrates the details of the related studies mentioned above.

3. Materials and methods

3.1. The CICIoT2023 dataset description

The “Canadian Institute for Cybersecurity” (CIC) [9] created the innovative and practical CICIoT2023 dataset in 2023, which is a collection of IoT attacks [5]. All attacks recorded in this dataset were carried out by malicious IoT devices whose goal was to attack other IoT devices [10]. The CICIoT2023 dataset involves 232,885 connections with 47 features, it includes 33 sub-attacks and falls within seven different attack classifications [11] which are: DoS, DDoS, Web-based, Recon, Spoofing, Brute force, and Mirai [5], as mentioned below. Table 2 illustrates the types of subattacks found in the CICIoT2023 datasets and their classifications, while Table 3 shows the percentage of each class.

3.2. Gaining-Sharing knowledge optimization algorithm

Gaining-Sharing Knowledge (GSK) is a revolutionary optimization algorithm based on human knowledge that has been proposed by Mohamed et al., in 2019 [13]. The GSK algorithm is built on how people learn and impart knowledge throughout their lives [14]. Junior and Senior are the two major stages of GSK, where the Junior stage represents gaining knowledge and the Senior stage represents knowledge sharing [15]. This algorithm consists of two stages. The first is related to gaining and sharing knowledge through interaction in small environments. The second is concerned with gaining and sharing knowledge through interaction in large environments [16]. In this algorithm, every person learns from others and engages in social interaction to reap the rewards of learning and impart their expertise where it is more practical

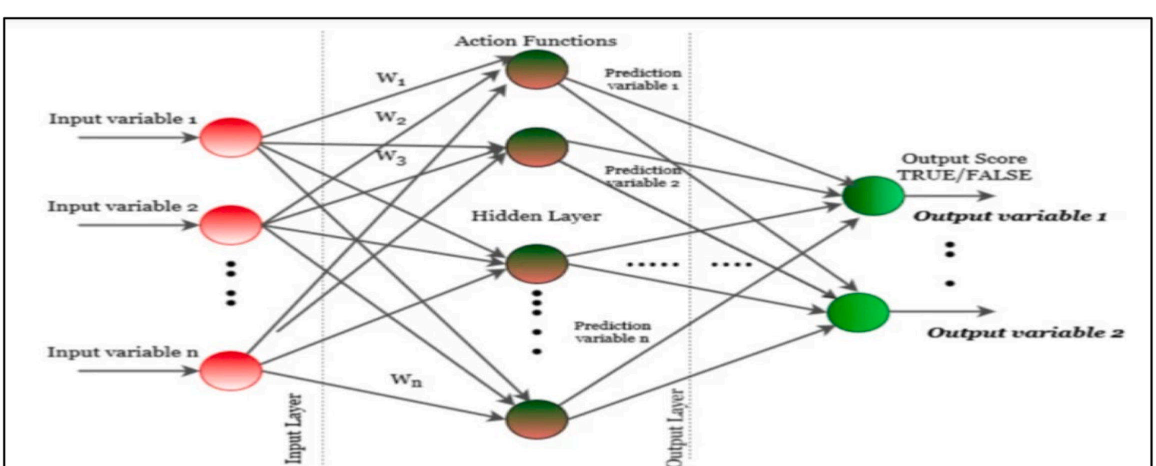


Fig. 1. MLP Architecture [26].

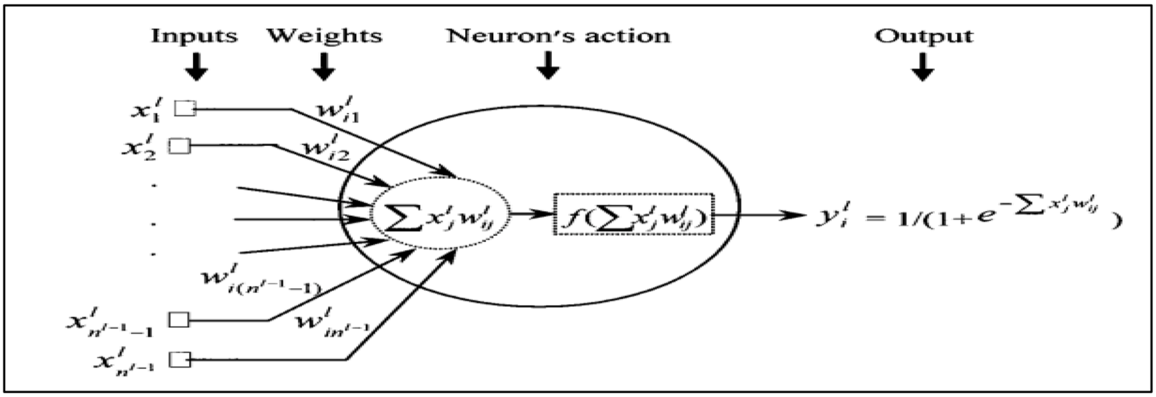


Fig. 2. Single neuron's structure in MLP [27].

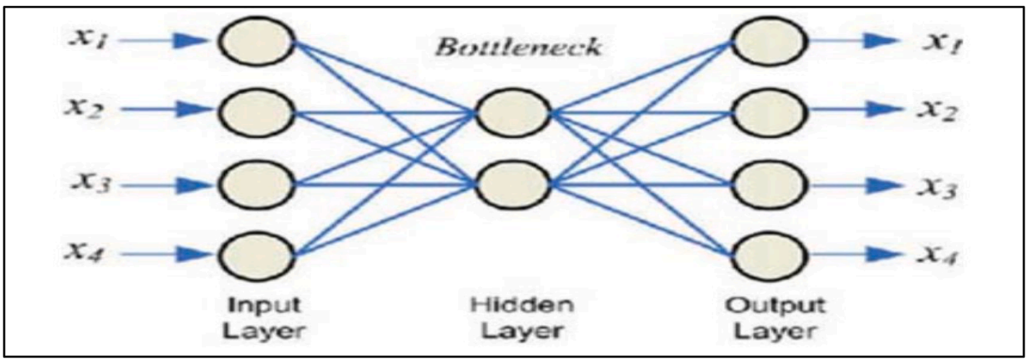


Fig. 3. Simple architecture of AutoEncoder [30].

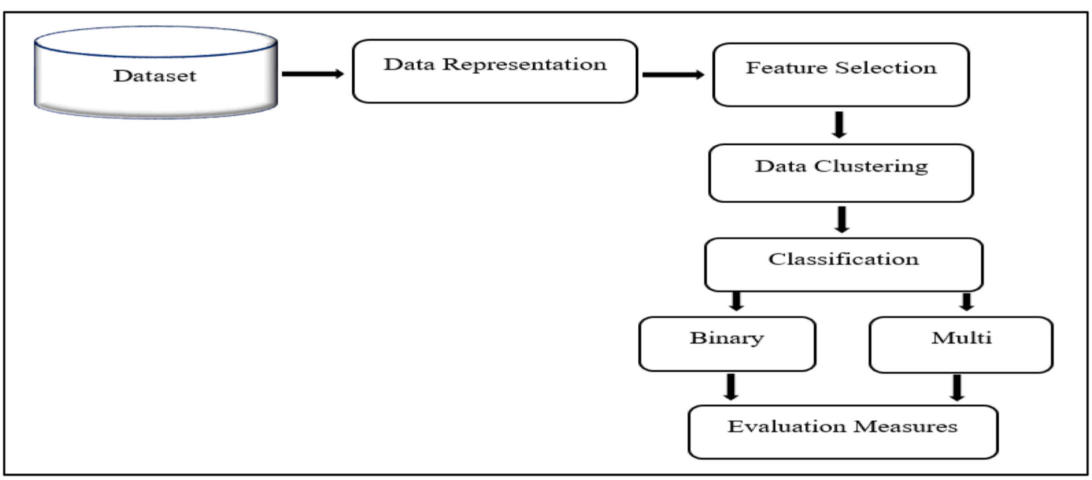


Fig. 4. General proposed model flowchart.

to obtain knowledge from their little network, then learn new things and impart them to the most suited people so they can improve their skills [17]. The junior dimension is computed as follows Equation [19]:

$$Dj = D \times (1 - (G/Gmax))^k \tag{1}$$

Where: Dj is the dimension of the junior stage, D is the dimension of population, Gmax is the max number of generations, G is the current generation, and k is the knowledge rate that determines the experience rate [18].

While the senior dimension is computed as follows [19]:

$$D_s = D - Dj \tag{2}$$

3.3. Multilayer perceptron algorithm

Concerning network architectures, the Multilayer Perceptron (MLP) is the most widely utilized type [19]. MLP is a feed-forward neural network that transfers data from the input layer to the output layer in a forward direction [20]. The three layers of MLP constitute the basic structure of the Artificial Neural Network (ANN) [21].

The input layer is responsible for receiving the data to be processed, then sending it to the hidden layer that performs the actual computations, and then to the output layer that gives the result of the required task such as classification or prediction [22]. Neurons in each layer are

Table 4
The Numerical Coding for CICIoT2023 dataset.

Class	Subclass	Code	Class	Subclass	Code
Normal		0	Web-based	BrowserHijacking	3
Dos	DoS-UDP_Flood	1		Backdoor_Malware	3
	DoS-TCP_Flood	1		XSS	3
	DoS-HTTP_Flood	1		Uploading_Attack	3
	DoS-SYN_Flood	1		SqlInjection	3
DDoS	DDoS-RSTFINFlood	2		CommandInjection	3
	DDoS-PSHACK_Flood	2	Recon	Recon-PingSweep	4
	DDoS-SYN_Flood’	2		Recon-OSScan	4
	DDoS-UDP_Flood	2		Recon-PortScan	4
	DDoS-TCP_Flood	2		Recon-VulnerabilityScan	4
	DDoS-ICMP_Flood	2		Recon-HostDiscovery	4
	DDoS-SynonymousIP_Flood	2	Spoofing	DNS_Spoofing	5
	DDoS-ACK_Fragmentation	2		MITM-ArpSpoofing	5
	DDoS-UDP_Fragmentation	2	BruteForce	DictionaryBruteForce	6
	DDoS-ICMP_Fragmentation	2	Mirai	Mirai-greeth_flood	7
	DDoS-SlowLoris	2		Mirai-greip_flood	7
	DDoS-HTTP_Flood	2		Mirai-udpplain	7

Table 5
Information gain calculation.

Feature	Information Gain	Feature	Information Gain
flow_duration	0.83682	DNS	0.0004
Header_Length	0.72855	SSH	0.00037
Protocol Type	0.44467	IRC	0
Duration	0.1986	TCP	0.07822
Rate	0.28142	UDP	0.04144
Srate	0.28142	DHCP	0
Drate	3.00E-05	ARP	0.00017
fin_flag_number	0.0416	ICMP	0.08196
syn_flag_number	0.03284	IPv	0.00055
rst_flag_number	0.0346	LLC	0.00055
psh_flag_number	0.03803	Tot sum	0.75153
ack_flag_number	0.09481	Min	0.60805
ece_flag_number	0	Max	0.70235
cwr_flag_number	0	AVG	0.81199
ack_count	0.07194	Std	0.57249
syn_count	0.14442	Tot size	0.70531
fin_count	0.02834	IAT	1.23428
urg_count	0.22203	Number	0.24548
rst_count	0.24409	Magnitude	0.80844
HTTP	0.00423	Radius	0.56314
HTTPS	0.07566	Covariance	0.55991
Telnet	0	Variance	0.25806
SMTp	0	Weight	0.24553

connected to their neighbors using weights, and a bias is used to provide a threshold to activate neurons [23]. MLP attempts to accurately predict the class labels of the input data by iteratively changing synaptic weights and biases until the network achieves a predetermined degree of accuracy and the amount of learned information is adequate [24]. Problems that cannot be solved linearly can be solved using MLPs, which are designed to approximate any continuous function [25]. Fig. 1 shows a simple architecture of MLP, while Fig. 2 shows a neuron structure inside a particular layer (Figs. 1-7).

3.4. AutoEncoder algorithm

The AutoEncoder (AE) algorithm is an unsupervised feed-forward neural network that is utilized to rebuild its input and looks for the best subspace in which the normal and anomalous data look extremely different [28]. The encoder and decoder are the two components that make up an AutoEncoder where the goal of the encoder is to reduce the dimensions of the input data and the decoder reconstructs the input data using the encoder’s low-dimension representation [29]. by using the idea of the artificial neural network, AE aims to reduce the dimensionality through the stack on the hidden layer by reducing the reconstruction layer [30]. The AE gets trained in an unsupervised manner and can extract important features from data that has not been labelled [31]. Fig. 3 Explain the main layers of AE.

4. The intrusion detection model

The architecture and details of the intrusion detection model are discussed in the following sections based on the general architecture illustrated in Fig. 4.

4.1. Data representation stage

Initially, the first stage was the preparatory processing of the data. Any decision-making system that deals with a large amount of data requires effective preparatory processing of the data.

First, as long as there are features with symbolic values available in the dataset, these values are converted into numeric ones to make the dataset easy to handle by the algorithm.

For the CICIoT2023 dataset, the only feature that contains symbolic values is the target feature, which is the last column of each connection record that contains the type of connection record, whether normal or attack. The attack types are converted into sequential numerical values. Table 4 shows the numerical coding for the types of attacks and Normal events.

Second, The Z-Score normalization method explained in Eq. (3), is implemented to rescale the values to the same range, taking into account the mean value and standard deviation of the data and performing the scaling.

$$\mathrm{Xnew} = (\mathrm{X} - \mathrm{M})/\mathrm{SD} \tag{3}$$

Where: Xnew is the new value after scaling, X is the current value, M is the mean value, and SD is the standard deviation.

4.2. Feature selection stage

Feature selection is performed to minimize the number of features in the dataset and select only the features of interest in intrusion detection. Information Gain was employed to select the important features from the CICIoT2023 dataset as shown in Eq. (4) by computing the information gain for each feature and then taking the mean of them. Only the features that have information gain higher than the mean will pass to the next stage as shown in Table 5.

$$IG(C,X) = H(C) - H(C|X) \tag{4}$$

Where C represents the class, X is the attribute, and H is the entropy function.

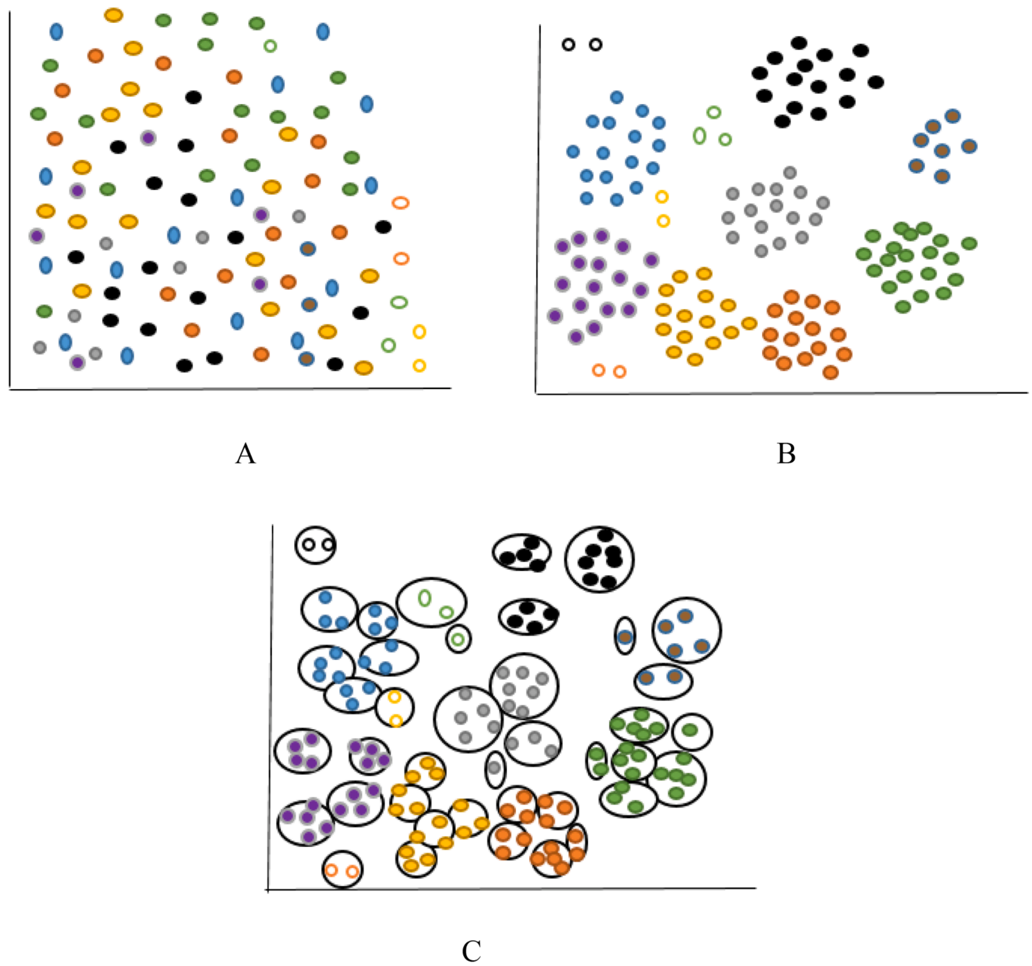


Fig. 5. Fitness value clustering (A: before clustering, B: after one-step clustering, C: after two-step clustering).

Table 6
The new size of the CICIoT2023 dataset and the reduction rate.

Dataset	Old Size	New Size	Reduction Rate
CICIoT2023	232,885 * 47	87,438* 16	62.45 %

Table 7
Binary classification of the CICIoT2023 dataset.

Classifier	Accuracy%	Precision%	Recall%	F-score%	Test Time / s
MLP	99.26	99	99	99	0.02
AE	98.76	99	99	99	1.15

Table 8
Multi classification of the CICIoT2023 dataset.

Classifier	Accuracy%	Precision%	Recall%	F-score%	Test Time / s
MLP	97.46	97	97	97	0.03
AE	83.81	83	84	80	1.03

$$H(X) = -\sum_{i=1}^{k} P_i log\ P_i \tag{5}$$

Where: H(X) is the Entropy value of the specific feature, calculated by summing the probabilities of individual points multiplied by logarithm probability.

The mean of all the information gain calculations is 0.26248, so the features that pass the mean are 15 features (16 features with the target feature).

4.3. Data clustering stage

The clustering strategy does not lessen the total amount of data, since the overall data will remain the same if all the data from each cluster are summed up, but the usage of clustering can be established to serve as the foundation for data reduction in addition to data clustering. To construct an accurate and effective intrusion detection model, a clustering technique for the data was proposed to handle the unbalanced data problem and thus increase the accuracy in detecting malicious activities and reduce the time required. The clustering technique employed the GSK optimization algorithm because it contains an important characteristic, it first finds the most useful information present in each node, and then shares the most important of it. This idea was exploited similarly; it identifies the node that generates the most percentage of intrusion information.

The GSK optimization algorithm works to find the proportion of important information present in the data and thus updates the fitness value of the data based on this proportion as explained in Fig. 5-A. Then the data was clustered according to similar fitness values, this step resulted in several groups, each containing data with similar fitness values as explained in Fig. 5-B. In the second step, the data in each group was clustered according to the majority of targets it contained, as described in Fig. 5-C, one data that represents the majority of targets in each group will then be taken, thus reducing the dimensionality of data

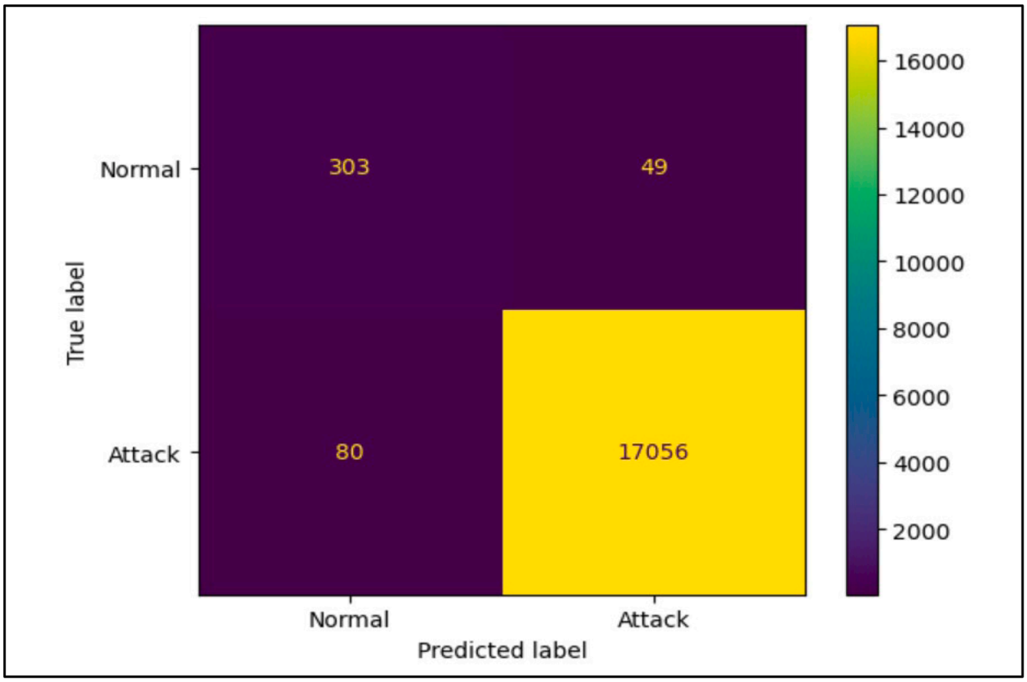


Fig. 6. Binary confusion matrix for MLP algorithm.

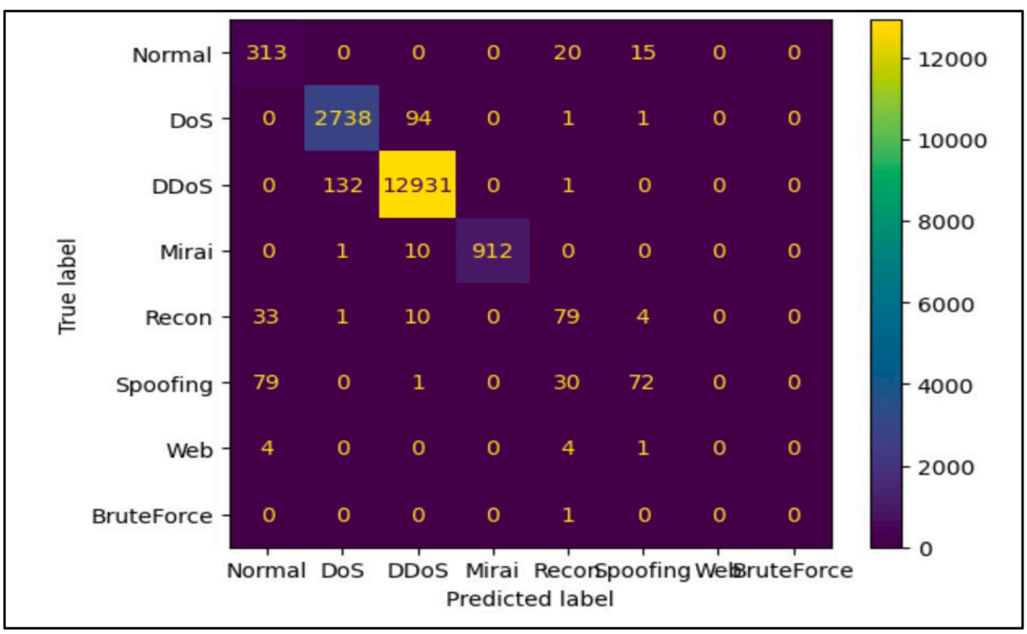


Fig. 7. Multi confusion matrix for MLP algorithm.

Table 9
Comparison between previous works and proposed model for CICIoT2023 dataset.

Reference	Accuracy%	Precision%	Recall%	F-score%	Test Time / s	Classification
Ref [5]	93.13	91.80	93.13	91.94	6.4	Multi
Ref [6]	95.25	99.36	–	99.20	–	Binary
Ref [7]	96.52	96.25	96.52	95.73	–	Multi
Ref [8]	96.32	64.26	56.23	60.00	5.68e-5	Multi
Proposed Model	**99.26**	**99**	**99**	**99**	**0.02**	**Binary**
	97.46	**97**	**97**	**97**	**0.03**	**Multi**

that will be subjected to training. The new size of the CICIoT2023 dataset will be changed as shown in Table 6.

It is clear from the figure above that similar fitness values formed groups of different sizes in the first phase of the clustering, and then the fitness values in each group were grouped according to the type of attack in the second phase of the clustering.

4.4. Classification stage

To construct an intrusion detection model with high accuracy and shorter testing time, two deep learning algorithms used are MLP and AE. The dataset was first trained with the MLP algorithm, then with AE. The suggested model is validated using four primary assessment metrics: accuracy, recall, precision, and F1 score, in addition to the testing time.

The data is divided into 80 % for training and 20 % for testing. The structure of the MLP algorithm consists of three hidden layers, the first one involves 100 neurons, the second one involves 50 neurons, and the third one involves 10 neurons. The activation function is ReLu, the optimizer is Adam, the learning rate is 0.01, the number of epochs is 200, and the batch size is 300. The structure of AutoEncoder consists of 5 layers, the first one with 128 neurons, the second with 64 neurons, the third with 32 neurons, the fourth with 64 neurons, and the last with 128. The number of epochs is 20, and the batch size is 32.

Tables 7 and Table 8 illustrate the evaluation results for the CICIoT2023 dataset in terms of binary classification and multi-classification, respectively.

From the Tables above, the MLP algorithm has the best outcomes in terms of binary and multi-classification. The confusion matrix of the binary and multi-classification of the MLP is shown in Fig. 6 and Fig. 7, respectively.

The comparison between the result of the proposed model based on the CICIoT2023 dataset and the previous research is illustrated in Table 9.

5. Conclusion

The choice of the Gaining-Sharing Knowledge (GSK) optimization algorithm was successful and valuable in building the intrusion detection system, as this algorithm proved effective in identifying the intrusion information contained in the data. An important feature that the GSK algorithm possesses, it first finds the useful information present in each node, and then shares the most important of it. This idea was exploited in this paper, by identifying the nodes that generate the most percentage of intrusion information. In fact, after its implementation, this algorithm collected the data that goes back to the normal target together at the beginning, which indicates that normal and intrusion-free data is data of little importance and that the data that goes back to the attack target is of higher importance. The two-step data clustering method has proven effective in reducing the dimensions of the data and thus obtaining better results in intrusion detection in terms of higher accuracy and shorter implementation time. The reduction percentage in the CICIoT2023 dataset size was 62.45 %. Reducing the dataset's dimensions helps preserve quick execution times while obtaining high detection performance and accuracy levels. The significance of applying these strategies to improve the efficacy of intrusion detection systems is underscored by these findings. In future works, we suggest applying the proposed model by suggesting other deep learning algorithms and then comparing the results with the currently used algorithms, Expand the scope of the ongoing work to encompass defense against various attacks.

CRediT authorship contribution statement

Hadeel Q. Gheni: Data curation, Formal analysis, Funding acquisition, Investigation, Project administration, Resources, Software, Visualization, Writing – original draft, Writing – review & editing. **Wathiq L. Al-Yaseen:** Conceptualization, Methodology, Supervision, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] M. Abdul Quadir, J. Dibyanshu, D. Jay, H. Sabireen, I. Celestine, K.J. Sanjiv, Efficient dynamic phishing safeguard system using neural boost phishing protection, Electronics (Basel) 11 (2022) 3133, https://doi.org/10.3390/electronics11193133.

[2] M. Souhail, R. Tajjeeddine, A. Nasser, Network based intrusion detection using the UNSW-NB15 dataset, Int. J. Comput. Digit. Syst. 8 (2019) 478–487, https://doi.org/10.12785/ijcds/080505.

[3] H. Shuokang, L. Kai, IGAN-IDS: an imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks, Ad Hoc Netw. 105 (2020) 102177, https://doi.org/10.1016/j.adhoc.2020.102177.

[4] K. Nilesh, T. Ritu. D. Joydip, Particle swarm optimization and feature selection for intrusion detection system, Sādhanā 45 (2022) 1–14, https://doi.org/10.1007/s12046-020-1308-5.

[5] W. Zhendong, C. Hui, Y. Shuxin, L. Xiao, L. Dahai, W. Junling, A lightweight intrusion detection method for IoT based on deep learning and dynamic quantization, PeerJ Comput. Sci. 9 (2023) e1569, https://doi.org/10.7717/peerj-cs.1569.

[6] P. Denis, G. Lubov, Z. Artur, P. Anton, Investigation of the impact effectiveness of adversarial data leakage attacks on the machine learning models, In ITM Web of Conferences. 59 (2024) 04011. https://doi.org/10.1051/itmconf/20245904011.

[7] A. Sidra, B. Imen, O. Stephen, Al H. Abdullah, A.S. Gabriel, A. Ahmad, G. Michal, Evaluating deep learning variants for cyber-attacks detection and multi-class classification in IoT networks, PeerJ Comput. Sci. 10 (2024) e1793, https://doi.org/10.7717/peerj-cs.1793.

[8] S. Onur, U. Suleyman, Advancing intrusion detection efficiency: a'less is more'approach via feature selection, (2023). https://doi.org/10.21203/rs.3.rs-3398752/v1.

[9] A.B. Shahid, A.K. Muhammad, A. Fawad, S.A. Mohammed, A. Hisham, A. Jawad, Enhancing IoT network security through deep learning-powered Intrusion detection system, Internet of Things 24 (2023) 100936, https://doi.org/10.1016/j.iot.2023.100936.

[10] P. Adrian, P. Yuda, I.S. Redho, A.W. Eko, Improvement attack detection on internet of thinks using principal component analysis and random forest, Media J. General Comput. Sci. 1 (2024) 14–19, https://doi.org/10.62205/mjgcs.v1i1.8.

[11] M.S. Mousa'B, K.H. Mohammad, S. Rossilawati, I. Shayla, R.K. Atta Ur, An explainable ensemble deep learning approach for intrusion detection in industrial internet of things, IEEE Access 11 (2023) 115047–115061, https://doi.org/10.1109/ACCESS.2023.3323573.

[12] C.P.N. Euclides, D. Sajjad, F. Raphael, Z. Alireza, L. Rongxing, A.G. Ali, CICIoT2023: a real-time dataset and benchmark for large-scale attacks in IoT environment, Sensors 23 (2023) 5941, https://doi.org/10.3390/s23135941.

[13] W.M. Ali, A.H. Anas, K.M. Ali, Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm, Int. J. Mach. Learn. Cybern. 11 (2020) 1501–1529, https://doi.org/10.1007/s13042-019-01053-x.

[14] A. Prachi, G. Talari, W.M. Ali, A novel binary gaining–sharing knowledge-based optimization algorithm for feature selection, Neural Comput. Appl. 33 (2021) 5989–6008, https://doi.org/10.1007/s00521-020-05375-8.

[15] Al.J. Samaher, M. Hadeer, M. Saif, One step to enhancement the performance of XGBoost through GSK for prediction ethanol, ethylene, ammonia, acetaldehyde, acetone, and toluene. data science for genomics, Academic Press (2023) 179–203, https://doi.org/10.1016/B978-0-323-98352-5.00011-2.

[16] A. Prachi, G. Talari, W.M. Ali, Chaotic gaining sharing knowledge-based optimization algorithm: an improved metaheuristic algorithm for feature selection, Soft comput. 25 (2021) 9505–9528, https://doi.org/10.1007/s00500-021-05874-3.

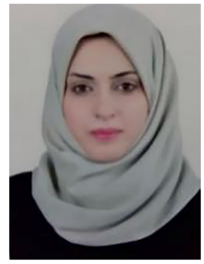
[17] W.M. Ali, F.A. Hattan, A.H. Anas, K.M. ALi, Gaining-sharing knowledge based algorithm with adaptive parameters for engineering optimization, IEEE Access 9 (2021) 65934–65946, https://doi.org/10.1109/ACCESS-2021-3076091.

[18] A.H. Said, M.A. Yousra, A. Khalid, A. Prachi, W.M. Ali, Stochastic travelling advisor problem simulation with a case study: a novel binary gaining-sharing knowledge-based optimization algorithm, Complexity (2020) 1–15, https://doi.org/10.1155/2020/6692978.

[19] D. Rita, F. Alessandra, M. Augusto, Geometrical synthesis of MLP neural networks, Neurocomputing 71 (2008) 919–930, https://doi.org/10.1016/j.neucom.2007.02.006.

[20] R. Pethuro, E. Preetha, The Digital Twin Paradigm For Smarter Systems and environments: The industry Use Cases, Academic Press, 2020.

[21] N. Jaebum, B. Trevon, L. Chihun, Y. Jooyeong, S. Sunae, R. Junsuk, Inverse design meets nanophotonics: from computational optimization to artificial neural network, Intelligent Nanotechnology 3 (2023), https://doi.org/10.1016/B978-0-323-85796-3.00001-9.
[22] S. Abinaya, M.K.D. Kavitha, Enhancing crop productivity through autoencoder-based disease detection and context-aware remedy recommendation system, Application of Machine Learning in Agriculture (2022) 239–262, https://doi.org/10.1016/B978-0-323-90550-3.00014-X.
[23] R. Ravi, B. Divya, Neural network model for biological waste management systems, Current Trends and Advances in Computer-Aided Intelligent Environmental Data Engineering (2022) 393–415, https://doi.org/10.1016/B978-0-323-85597-6.00010-0.
[24] B. Priti, L. Rishabh, J. Vaibhav, J. Tanmay, S. Sanchit, K. Sumit, K.S. Pradeep, K. Baseem, GGA-MLP: a greedy genetic algorithm to optimize weights and biases in multilayer perceptron, Contrast Media Mol. Imaging (2022) 2022, https://doi.org/10.1155/2022/4036035.
[25] T. Menzies, E. Kocaguneli, L.M. Leandro, P. Fayola, T. Burak, Using goals in model-based reasoning, Sharing Data and Models in Software Engineering 1 (2015) 321–353, 1016/B978012/417295-100024-2.
[26] J. Naskath, G. Sivakamasundari, & A.A.S. Begum, A study on different deep learning algorithms used in deep neural nets: MLP SOM and DBN. Wireless personal communications, 128 (2023) 2913–2936. https://doi.org/10.1007/s11277-022-10079-4.
[27] Z. Xiaoqin, D.S. Yeung, Sensitivity analysis of multilayer perceptron to input and weight perturbations, IEEE Trans. Neural Networks 12 (2001) 1358–1366, https://doi.org/10.1109/72.963772.
[28] S. Amardeep, J.J. Julian, Autoencoder-based unsupervised intrusion detection using multi-scale convolutional recurrent networks. 2022. https://doi.org/10.48550/arXiv.2204.03779.
[29] F. Fahimeh, H. Jukka, A deep auto-encoder based approach for intrusion detection system, in: 2018 20th International Conference on Advanced Communication Technology (ICACT), IEEE, 2018, pp. 178–183, https://doi.org/10.23919/ICACT.2018.8323688.
[30] N.K. Yesi, N. Siti, S. Deris, Z. Ahmad, Jasmir Firdaus, Automatic features extraction using autoencoder in intrusion detection system, in: 2018 International Conference on Electrical Engineering and Computer Science (ICECOS), IEEE, 2018, pp. 219–224, https://doi.org/10.1109/ICECOS.2018.8605181.
[31] I. Cosimo, A. Ahsan, C.M. Francesco, H. Amir, A novel statistical analysis and autoencoder driven intelligent intrusion detection approach, Neurocomputing 387 (2020) 51–62, https://doi.org/10.1016/j.neucom.2019.11.016.



Hadeel Qasem Gheni was born in Iraq / Babylon in 1984, received a bachelor's degree in computer science from the university of Babylon, faculty of science for women, computer department, Iraq, in 2006, and a master's degree in artificial intelligence from the university of Babylon, faculty of information technology, software department, Iraq, in 2016. She is currently a lecturer at the university of Babylon, faculty of science for women, computer department, Iraq. Her current research interests include artificial intelligence, machine learning, and data mining.

Wathiq Laftah Al-Yaseen: received the bachelor's degree in computer science from the University of Basra, Iraq, in 2000, and the master's degree in computer science from the University of Babylon, Iraq, in 2003, and PhD degree in computer science/artificial intelligence from UKM, Malaysia, in 2017. He is currently an Assistant Professor in Technical Institute of Karbala, Al-Furat Al-Awsat Technical University, Iraq. His current research interests include artificial intelligence, network security, bioinformatics, machine learning, multi agent system, and data mining. He can be contacted at email: wathiq@atu.edu.iq