A Hybrid Method for Intrusion Detection in the IOT

Hossein Faghih Aliabadi*

MSC.Computer Networks, Faculty of Electricity, Computer and Advanced Technologies of Urmia University, Iran;
Hosseinfaghih1995@gmail.com

ABSTRACT

In computer networks, introducing an intrusion detection system with high precision and accuracy is considered vital. In this article, a proposed model using a deep learning algorithm is presented and its results are analyzed. To evaluate the performance of this algorithm, NSL-KDD, CIC-IDS 2018, UNSW-NB15 and MQTT datasets have been used. The evaluation criteria include precision, accuracy, F1 score, and, readability. The new approach uses a hybrid algorithm that includes a convolutional neural network (CNN) to extract general features and long-short-term memory (LSTM) to extract periodic features that are in the form of a layer. are cross-connected, it is introduced to detect penetration. This algorithm showed the highest known accuracy of 99% on the NSL-KDD dataset. It has reached 97% in all criteria in UNSW-NB15, 96% in all criteria in CIC-IDS 2018, and also, in MQTT for three abstraction levels of features, i.e. packet-based flow features, unidirectional flow, and The two-way flow has reached above 97%, which shows the superiority of this algorithm.

Keywords: Internet of Things, Intrusion Detection System, Hybrid System, Deep Learning Introduction.

1. Introduction

Attacks and intrusions on computer networks have increased significantly along with the growth of these networks. To deal with intruders to networks and computer systems, several methods were developed, which are called intrusion detection methods. The purpose of the intrusion detection process is to identify unauthorized uses, abuses, and possible damages to computer systems and networks. In general, intrusion detection methods are divided into two main categories: detection of abuse (signature) and detection of abnormal behavior (anomaly) [1]. The signature detection method works based on the pattern of known intrusions. In this method, the intrusion detection problem is transformed into a classification problem and detects intrusion. One of the problems of intrusion detection systems based on signature is recognizing a normal behavior as a wrong behavior. The meaning of false alarm is a warning that is announced by the intrusion detection system when the attack does not occur [2]. The existence of a false alarm, even at a low level, if the normal traffic load of the network is high, causes numerous and boring alarms, for this reason, the false alarm generation rate by an intrusion detection system should be as low as possible. Of course, it is important to mention that keeping the rate low reduces the system's ability to detect attacks. In other words, a balance should be established between high detection accuracy and low false alarm rate [3]. Abnormality detection should identify normal behaviors and implement specific patterns and rules for them. In this method, the identification of unknown attacks is based on the user's profile, and the drawback of this method is the high rate of false alarms [4]. IDS tools are divided into two groups based on the host and based on the network from the perspective of the monitoring environment. A host-based system is a method of detecting malicious activity on a computer that can identify unauthorized objects that cannot be detected by other categories. In the network-based method, everyone monitors the network and controls network traffic to detect intrusion [5]. This research has tried to identify redundant and less effective features in the detection of each attack while examining and analyzing the datasets statistically. Therefore, this study has been done to detect intrusion in the network and to achieve this goal, a deep learning method has been used. In the second part, we will review the past works, in the third part, we will review the materials and methods, in the fourth part, we will review the results of the implementation, and finally, the results of this research.

2. Related Work

The first study presented about the necessity of automatic system security inspection dates back to 1980 [6] [From 1984 to 1986, Neumann Peter and Denning Dorothy conducted research in the field of computer system security, and the system The result was named IDES[7]. The idea proposed in this project has been used as the basis of many penetration systems. Khan et al[8] developed a convolutional neural network, which used the ISCX-UNB dataset. The authors achieved a precision of 97.29 and a false alarm of 71.0. Ammar[9] used LSTM on UNB-ISCX with 97% precision, a 22% drop, and a 1.47% false alarm rate. In Ganavan et al.'s model [10], data training is done in a new way, in this method, algorithms are defined to analyze the increase in training time. In the article[11], they used the combination of genetic algorithm and fuzzy logic to detect the abnormality of the network. A genetic algorithm is used to create a digital signature using network traversal and fuzzy logic is used to identify whether a sample is abnormal or not. Yin et al[12] improved RNN for intrusion detection on NSL-KDD data and showed 97% precision and 1765 time. Kim et al[6] used CNN to detect DoS attacks, achieving 99% precision for KDD99 data and 91.5% precision for IDS2018 data. Nguyen and Kim

[13] experimented with CNN with a genetic combination of NSL-KDD data. In this method, unauthorized activities with 98.2% precision, FPR value equal to 0.52%, and TFR value equal to 95.44% were obtained from the implementation of the algorithm. Altobiti et al[7] achieved 85% precision from the LSTM algorithm with the CI-DDS dataset. Vinayakumar Ravi [14] proposed a model that extracts recurrent features and uses KPCA to discover useful features. Specifically, the method in this paper has demonstrated an accuracy of 94% using the SDN-IoT dataset. Tanzila[15] presents a CNN-based approach that takes advantage of the power of the Internet of Things and provides qualities to efficiently survey the entire traffic across the Internet of Things. The model is tested using NID and BoT-IoT datasets. It has reached 96.51% accuracy for NID data and 92.85% accuracy for BoT-IoT data.

3. Methodology

Our proposed model in this paper is CNN+LSTM, which uses convolution features for intrusion detection and LSTM to extract data from network flow. The purpose of the proposed model is to combine the features using CNN+LSTM so that we can extract the data that have a lot of attachment to the previous data so that we can increase the evaluation parameters such as accuracy with this method.

3.1. Dataset

Almost all IDS algorithms work well only for specific data sets while they do not work well for other data sets. Therefore, in this work, four public datasets will be used to test the proposed CNN+LSTM algorithm.

NSL-KDD is a dataset proposed to solve some of the problems inherent in the KDD'99 dataset. Since KDD99 has duplicate entries, researchers in 2009 improved NSL-KDD as an alternative to KDD99. NSL-KDD includes 10 basic features, 12 content, and 19 traffic features [16].

CIC-IDS 2018 is a popular dataset in the field of intrusion detection. This dataset is developed using the AWS platform. In this dataset, several attacks are displayed that can be used in the field of security and applied in a general approach to network topologies and protocols. This dataset has been enhanced according to IDS2017 standards. The CSE-CIC IDS2018 dataset is available to all researchers over the Internet in both CSV and PCAP formats. This data with approximately 5 million records has 1048575 samples and 80 features, which has binary and multi-class mode [17].

UNSW–NB15 dataset generated normal and attack behaviors of network traffic using the Perfect Storm tool. This data has used two servers in the IXIA traffic generation tool, where one server creates normal activities and the other creates malicious activities in the network. The UNSW-NB15 dataset contains 42 features with class labels and 9 different attacks named [18].

MQTT dataset: In this data, five scenarios are recorded, which include normal operations and four attack scenarios. Five scenarios are recorded in this data, which include normal operations and four attack scenarios. The data is obtained using the TCP-dump function. The attacker performs four attacks Scan A, Scan sU, Sparta, and brute force, each of which is analyzed separately. It evaluates three abstract levels of features, namely, packet-based features, unidirectional-based features, and bidirectional-based features. become [19].

3.2. Pre-processing

Preprocessing includes data adjustment and normalization. In the original data set, the values are of different types. In deep learning, the data must be numeric for preparation, some features are strings that cannot be processed. Likewise, some values may not fall within the specified range due to different representations. Some features are too different and are not suitable for the final classification of intrusion detection, normalization, and feature processing [20], in this section, as shown in Figure 1, with the methods of Cleaning, Feature Filtering, OHE, and We normalizing and trying to reduce these problems. For cleaning, we used Python libraries and functions to examine the dataset. Boolean returns two values, true or false. True means data loss and false means data are clean. There were records in the dataset that needed to be purged first. For this purpose, we removed all unclear and empty samples. If a data set has features that cannot affect the model, we should remove those features. Because it creates too much fit and lack of fit, which prolongs the time of the model and reduces the performance of the system. Feature selection is a method of removing useless features from a dataset. The main purpose of feature selection has been to avoid overfitting and lack of fit, improve efficiency, reduce learning time, and increase model response time. We apply the OHE function to convert symbolic features into numeric vectors to represent integers. The dataset contains numeric attributes and some non-numeric attributes such as protocol type. Non-numeric features such as service and flag must be converted to numeric features, so that the model input is in the form of a numeric matrix. The class attribute is also labeled as a numeric type, which we consider to be zero for Normal and one for Attack. After one-hot, numerical properties are enhanced if values are applied. To reduce the effect of large features, the normalization method is used in such a way that the features in the data set should be in the [0,1] range. The normalization method is stated in Equ.(1).
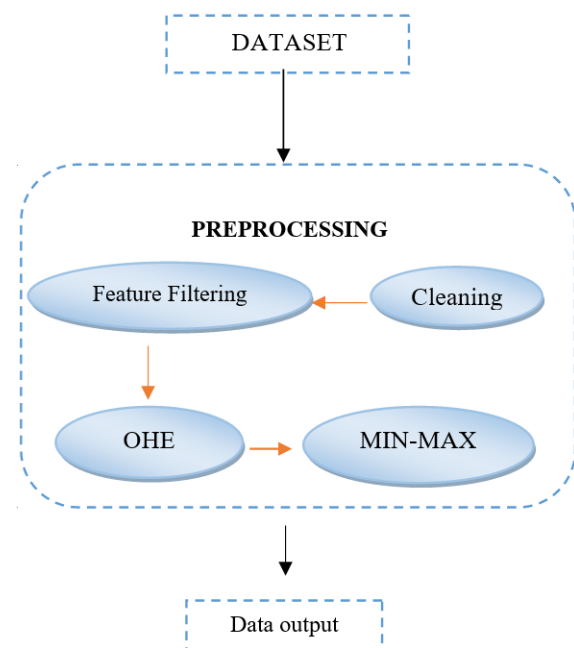

Figure. 1. Show the preprocessing of the dataset

$$x = \frac{X - MIN}{MAX - MIN} \qquad (1)$$

3.3. Deep Learning

Deep learning is a type of machine learning and artificial intelligence that actually mimics the way the human mind learns a certain subject. This type of learning is one of the important elements in data science, which includes statistics and predictive modeling. for by example, Deep learning is a key technology behind driverless cars, enabling them to recognize a stop sign or to distinguish a pedestrian from a lamppost[21]. Deep learning consists of three categories. The first type is supervised learning, which is used to extract features, which are presented to machine learning methods to perform tasks such as classification and recognition. The second type is unsupervised feature learning, which only depends on extracting useful features from the whole model [22-23]. The third is a compound to enhance the training of neural networks [24]. The main purpose of deep learning algorithms is feature learning and classification. Table 1 shows the Benefits and challenges of deep learning.

A convolutional neural network (CNN or ConvNet) is a network architecture for deep learning that learns directly from data. CNN's are particularly useful for finding patterns in images to recognize objects, classes, and categories. They can also be quite effective for classifying audio, time series, and signal data. Different from traditional neural networks, convolutional neural networks share the weights in the convolution layer, which greatly reduces the weights and improves the performance. A typical convolutional neural network mainly consists of the input layer, convolution layer, pooling layer, and fully connected layer [25].

LSTM stands for long short-term memory networks, used in the field of Deep Learning. It is a variety of recurrent neural networks (RNNs) that are capable of learning long-term dependencies, especially in sequence prediction problems. LSTM is a special type of RNN network that solves the long-term memory problem of RNN. This network has internal mechanisms called gates that control information flows and determine which data in the sequence are important and should be preserved and which data should be deleted. In this way, the network passes important information along the sequence chain to get the desired output [26].

3.4. Suggested Method

The proposed method consists of three main phases., in the first phase, pre-processing is the normalization of the data set to prepare and standardize the data. In the next step, the processed data are given as input to the deep learning network in the proposed method so that feature extraction and classification can be done on them. The proposed method is an algorithm based on the integration of the mutual features of CNN and LSTM layers. This model consists of CNN components with 32 dimensions and LSTM with 100 neurons that are connected. The CNN component extracts global features, while the LSTM component extracts periodic features. After the extracted features are mixed together, abnormal information is detected. This algorithm mixes the features of CNN and LSTM, so it shows good performance in detecting attacks. The third phase is classification, based on a training set, the system learns to divide the data into correct groups with the least error. The training set contains data whose categories are known, each pattern or category has a

label, and data with the same target label are placed in a group. The classification process has two phases, training and testing. In this work, we select about 70% of the data in the dataset as training data and 30% of the remaining data for testing and validation. Figure 2 shows the proposed hybrid architecture and Table 2 shows the proposed network parameters. Also, in Figure 3, the proposed method is shown as an example for the UNSW–NB15 dataset with ten classes.
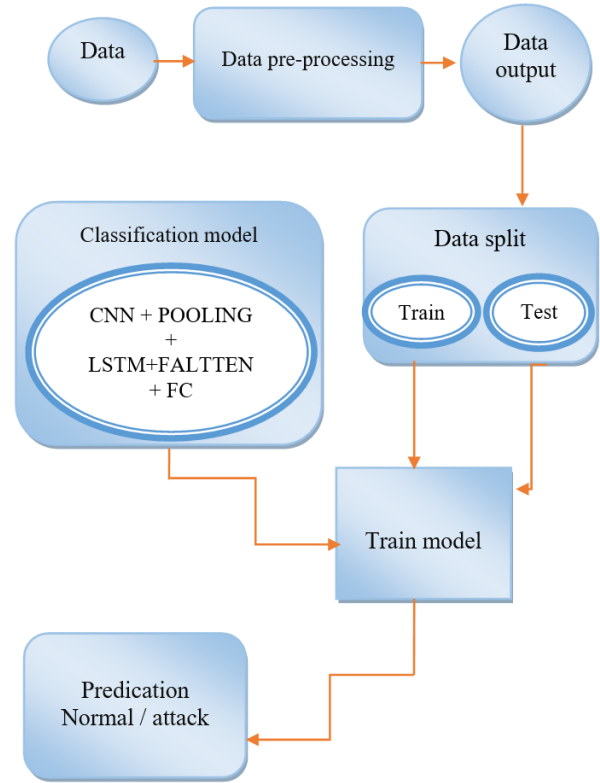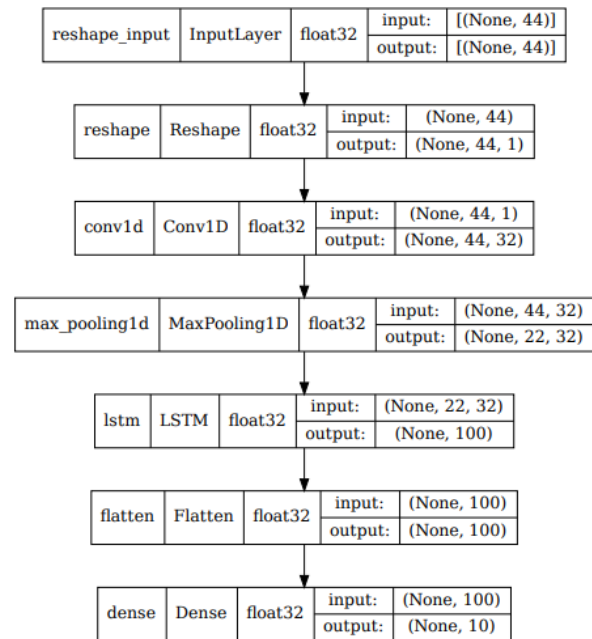

Figure. 2. Proposed hybrid architecture

Figure. 3. the proposed method for the UNSW–NB15

4. Experimental Results

In this section, we present the tests performed and the simulation results obtained from the implementation of the CNN+LSTM model by applying different types of running parameters. Regarding the datasets, NSL-KDD, CSECIC-IDS 2018, UNSW-NB15, and MQTT datasets were selected as our training and testing benchmark datasets. In evaluating the performance of deep learning algorithms based on various criteria, most researchers focused on accuracy as the primary criterion, while the improvement of the intrusion detection system depends on accuracy, detection, and classification.

4.1. Performance Metrics

In this layered model, the confusion matrix, which is a method to evaluate the classification results, is used, which includes information about the actual output. The following evaluation criteria are used in this article.

True positive (TP) Attack data that is correctly classified as an attack.

True negative (TN) normal data that is correctly classified as normal.

False positive (FP) Normal data that is wrongly classified as an attack.

False negative (FN) Attack data that is wrongly classified as normal[30].

If the attack is classified as a normal record, the attackers manage to bypass the ids, then there are problems with the confidentiality and availability of network resources. The arrangement of the confusion matrix is shown in Table 3. [31].

4.2. Evaluation of the IDS System

In this section, we will show the results of the CNN+LSTM architecture simulation in the TensorFlow platform with 30 test periods and a learning rate of 0.005. For this purpose, evaluation criteria including precision, accuracy, F1 score, and readability were used, which will be displayed on the confusion matrix. Table 4 shows the clutter matrix for the NSL-KDD dataset, which achieved 99% in all criteria. Figure 4 shows the distribution of the CSE-CIC 2018 data set and Table 5 shows the confusion matrix of this data set for three classes: Benign, FTP, and SSH. Tables 6, 7 and 8 are the confusion matrix related to the MQTT dataset and for three features and 5 classes of this dataset, we reached acceptable and high results of 97%. The results of this simulation are shown in Tables 9 and 10. Figure 5 shows the confusion matrix of the UNSW-NB15 dataset for 10 attack classes, which reached 97% in all parameters. Also, we implemented some other AI algorithms on the UNSW-NB15 dataset and found out our proposed model showed the best results in all parameters. We showed the result of this implementation in Figure 6.

5. Conclusions and Outlook

The purpose of this work was to investigate the challenges and requirements for building an intrusion detection system for the Internet of Things models, also this article shows that deep learning algorithms are very effective for anomaly detection and intrusion prevention. Because machine learning methods cannot detect modern attacks with good accuracy. Instead,
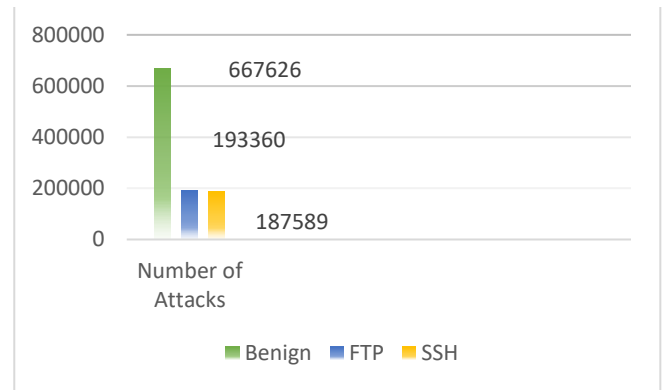

Figure. 4. Distribution of the CIC-IDS 2018 dataset

Predicted lable / True lable	normal	Exploits	Reconnaissance	DoS	Generic	Shellcode	Fuzzers	Worms	Backdoors	Analysis
normal	۸۴۷۸۸۲	۰	۰	۰	۰	۰	۰	۰	۰	۰
Exploits	۰	۲۱۲۴۸	۰	۰	۰	۰	۰	۰	۰	۰
Reconnaissance	۰	۰	۶۲۷۷	۰	۰	۰	۰	۰	۰	۰
DoS	۰	۰	۰	۸۰۷۰	۰	۰	۰	۰	۰	۰
Generic	۰	۰	۰	۰	۸۴۰۴۴	۰	۰	۰	۰	۰
Shellcode	۰	۰	۰	۰	۰	۶۹۸	۰	۰	۰	۰
Fuzzers	۰	۰	۰	۰	۰	۰	۱۰۵۳۲	۸	۰	۰
Worms	۰	۰	۰	۰	۰	۰	۰	۷۷	۰	۰
Backdoors	۰	۰	۰	۰	۰	۰	۰	۰	۹۵۶	۰
Analysis	۰	۰	۰	۰	۰	۰	۰	۰	۰	۱۲۳۵

Figure. 5. Confusion matrix of UNSW-NB15
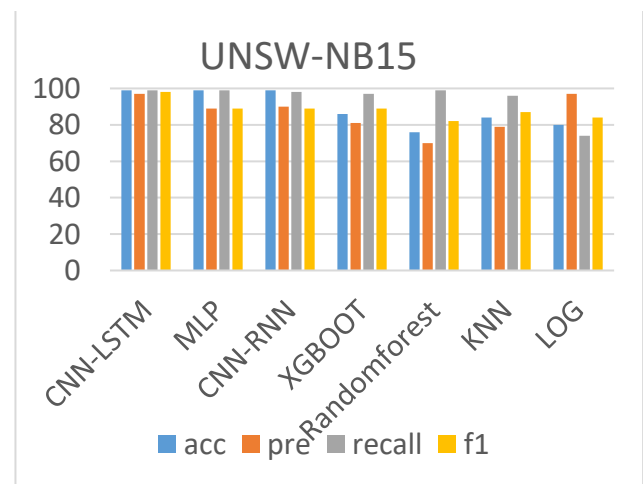
Figure. 6. Implementation of the algorithm on UNSW-NB15 database

deep learning methods have the ability to find better ways to create better models. Deep learning methods work very well in identifying and discovering different types of attackers. We have used NSL-KDD, UNSW-NB15, and MQTT datasets for our evaluations. In NSL-KDD, we reached unprecedented results of 99% in all criteria. In Table 11, We compared the method we proposed with the methods proposed by other authors and showed that this algorithm achieved better accuracy. In CSE-CIC2018, we reached 96% accuracy for three classes Benign, FTP, and SSH. In UNSW-NB15, we reached 97 in all parameters, And in Table 12, we checked the

methods of other authors with our proposed method and showed the superiority of the method presented in this article. In MQTT, three feature levels were extracted from the dataset of raw pcap files, i.e. closed, one-way, and two-way features. Each feature level is used independently in the experiments and we reached results above 97%. The results showed that even if the data, tools, and techniques are available, many points such as improvement, development, and analysis are necessary to increase the efficiency of the model. We proved that the combination of CNN and LSTM is an excellent method for network intrusion detection.

Table 1. Benefits and challenges of deep learning

Benefits	Deep architecture is very suitable for feature extraction [27].
	It has high computing power [28].
	Deep learning can solve the challenges of working with huge data to a great extent [28].
Challenges	The time complexity of processes in deep learning is a challenge [29].
	Incremental learning in non-stationary data is a challenge [28].
	Educational data is considered a challenge [27].

Table 2. The best parameters of the CNN+LSTM network

Layer	Output shape
Reshape	(None, 44, 1)
Conv1D	(None, 44, 32)
Max_pooling1d	(None, 22, 32)
LSTM	(None, 100)
Flatten	(None, 100)
Dense	(None,)

Table 3. Confusion matrix for the IDS scheme

Actual Label	Predicted Label	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Table 4. Confusion matrix for NSL-KDD

Predicted labele / True label	0	1
0	41302	63
1	17	34202

Table 5. CIC-IDS2018 confusion matrix

	Benign	FTP	SSH
Benign	2043	13	0
FTP	8	1984	8
SSH	25	13	1915

Table 6. Bidirectional-based features confusion matrix

Predicted label / True label	NORMAL	SCAN-A	SCAN-SU	SPARTA	M-BF
NORMAL	25990	0	0	2	0
SCAN-A	0	606	0	0	4
SCAN-SU	14	4	643	0	0
SPARTA	0	0	0	4162	0
M-BF	141	0	0	0	4175

Table 7. Unidirectional-based features confusion matrix

Predicted label / True label	NORMAL	SCAN-A	SCAN-SU	SPARTA	M-BF
NORMAL	5100	4	0	0	2
SCAN-A	0	1204	0	20	0
SCAN-SU	0	5	670	0	0
SPARTA	7	0	0	8491	0
M-BF	195	0	0	0	8537

Table 8. Confusion matrix of packet-based features

Predicted label / True label	NORMAL	SCAN-A	SCAN-SU	SPARTA	M-BF
NORMAL	58002	0	2	0	10
SCAN-A	0	4993	0	7	0
SCAN-SU	0	9	5160	0	0
SPARTA	0	0	0	9E+04	0
M-BF	20	0	0	0	65200

Table 9. The results of MQTT

	CNN+LSTM					
	ACC			PRE		
	PACKET	UNI	BI	PACKET	UNI	BI
NORMAL	100	98	100	100	99	99
SCAN-A	99	99	99	100	100	99
SCAN-SU	100	98	97	99	99	98
SPARTA	98	99	99	98	97	96
MQTT-BF	98	97	96.9	99	97	98

Table 10. The results of MQTT

	CNN+LSTM					
	RECALL			F1		
	PACKET	UNI	BI	PACKET	UNI	BI
NORMAL	99	99	98	97	99	100
SCAN-A	9	98	98	97	96	97
SCAN-SU	99	98	98	99	98	99
SPARTA	100	99	100	99	99	100
MQTT-BF	99	98	97	99	98	97

Table 11. Comparison of methods in NSL-KDD

	ACC	PRE	RECAL	F1
BiDLSTM[32]	91.36	92.81	91.36	91.67
XGBoost[33]	97	97	96.8	96.8
AE + ANN[34]	76.88	80.22	75.74	72.75
RNN+LSTM[35]	81.60	81.24	89.74	89.54
Proposed model	99	99	99	99

Table 12. Comparison of methods in UNSW-NB15

	ACC	*PRE*	*RECAL*	*F1*
1D-CNN 1 layer[36]	90.91	85.84	96.07	90.94
ANN[37]	94.49	81.54	98.06	89.04
OCNN–HMLST[38]	96.334	100	95.87	98.132
Fully connected DNN[39]	89	85	97	91
Proposed model	**97.8**	**97.5**	**96.9**	**97**

Declarations

Authors' contributions

HF: Study design, acquisition of data, interpretation of the results, statistical analysis, drafting the manuscript

Conflict of interest

The authors declare that there is no conflict of interest

References

[1] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model", *Journal of Computational Science*, vol. 25, pp.152-160, 2018.

[2] B. Susilo and R. F. Sari, "Intrusion detection in IoT networks using a deep learning algorithm. *Information*, vol. 11, no. 5, p. 279, 2020.

[3] M. Grill, T. Pevný, M. and Rehak, "Reducing false positives of network anomaly detection by local adaptive multivariate smoothing", *Journal of Computer and System Sciences*, vol. 83, pp.43-57, 2017.

[4] S. Smys, A. Basar, and H. Wang, "Hybrid intrusion detection system for internet of things (IoT)", *Journal of ISMAC.*, vol. 2, no. 04, pp.190-9, 2020.

[5] W. Meng, E. W. Tischhauser, Q. Wang, Y. Wang, and J. Han, "When intrusion detection meets blockchain technology: A review", *Ieee Access*, vol. 6, pp.10179-10188, 2018.

[6] J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, "CNN-based network intrusion detection against denial-of-service attacks", *Electronics*, vol. 9, no. 6, p. 916, 2020.

[7] S. A. Althubiti, E. M. Jones, and K. Roy, "LSTM for anomaly-based network intrusion detection", In *2018 28th International Telecommunication Networks and Applications Conference (ITNAC)*, IEEE, 2018, pp. 1-3.

[8] M. A. Khan, M. Karim, and Y. Kim, "A scalable and hybrid intrusion detection system based on the convolutional-LSTM network", Symmetry, vol. 11, no. 4, p. 583, 2019.

[9] M. Amar, and B. E. Ouahidi, "Weighted LSTM for intrusion detection and data mining to prevent attacks", *Int. J. Data Mining, Modell. Manage.*, vol. 12, no. 3, pp. 308–329, 2020.

[10] M. Uddin, A. A. Rahman, N. Uddin, J. Memon, R. A. Alsaqour, and S. Kazi, "Signature-based MultiLayer Distributed Intrusion Detection System using Mobile Agents", *Int. J. Network Security*, vol. 15, pp.97-105, 2013.

[11] A. H. Hamamoto, L. F. Carvalho, L. D. H. Sampaio, T. Abrão, and M. L. Proença Jr, "Network anomaly detection system using genetic algorithm and fuzzy logic", *Expert Systems with Applications*, vol. 92, pp. 390-402, 2018.

[12] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks", *Ieee Access*, vol. 5 pp. 21954–21961, 2017

[13] M. T. Nguyen, and K. Kim, "Genetic convolutional neural network for intrusion detection systems", *Future Gener. Comput. Syst.*, vol. 113, 418–427, 2020M. T. Nguyen, and K. Kim, "Genetic convolutional neural network for intrusion detection systems", *Future Gener. Comput. Syst.*, vol. 113, 418–427, 2020.

[14] V. Ravi, R. Chaganti, and M. Alazab, "Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system". *Computers and Electrical Engineering*, vol. 102, p. 108156, 2022

[15] T. Saba, A. Rehman, T. Sadad, H. Kolivand, and S. A. Bahaj, "Anomaly-based intrusion detection system for IoT networks through deep learning model", *Computers and Electrical Engineering*, vol. 99, p. 107810, 2022

[16] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," In *Computational Intelligence for Security and Defense Applications*, IEEE, 2009, pp. 1–6.

[17] https://www.unb.ca/cic/datasets/ids-2018.html

[18] N. Moustafa, and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)", In *Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, IEEE, November 2015, pp. 1–6

[19] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, A. Hotho, "A Toolset for Intrusion and Insider Threat Detection, In *Data Analytics and Decision Support for Cybersecurity*, Cham, Springer, 2017, pp. 3–31. https://doi.org/10.1007/978-3-319-59439-2-1.

[20] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[21] L. Deng, "Deep Learning: Methods and Applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3-4, pp. 197–387, 2014

[22] S. Choudhury and A. Bhowal, "Comparative analysis of machine learning algorithms along with classifiers for network intrusion detection," In *2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, IEEE, May 2015, pp. 89–95.

[23] M. Anbar, R. Abdullah, I. H. Hasbullah, Y. W. Chong, and O. E. Elejla, "Comparative performance analysis of classification algorithms for intrusion detection system," In *2016 14th Annual Conference on Privacy, Security and Trust (PST)*, IEEE, Dec 2016, pp. 282– 288

[24] Y. Y. Aung and M. M. Min, "An analysis of random forest algorithm based network intrusion detection system", In *2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, IEEE, June 2017, pp. 127–132.

[25] C. Garcia Cordero, S. Hauke, M. Muhlhauser, and M. Fischer, "Analyzing flow-based anomaly intrusion detection using Replicator Neural Networks," In *2016 14th Annual Conference on Privacy, Security and Trust (PST)*. Auckland, New Zeland: IEEE, Dec 2016, pp. 317–324.

[26] J. Wang, J. Zhang, and X. Wang, "Bilateral LSTM: A two-dimensional long shortterm memory model with multiply memory units for short-term cycle time forecasting in re-entrant manufacturing systems", *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, pp. 748-758, 2017.

[27] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review", *Neurocomputing*, vol. 187, pp. 27-48, 2016.

[28] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagc, "Deep Learning Techniques in Big Data Analytics", In *Big Data Technologies and Applications*, Cham, Springer, 2016, pp. 133-156.

[29] H. Malallah S. R. Zeebaree R. R. Zebari M. A. Sadeeq Z. S. Ageed, I. M. Ibrahim,... and K. J. Merceedi, "A Comprehensive Study of Kernel (Issues and Concepts) in Different Operating Systems", *Asian Journal of Research in Computer Science*, vol. 8, no. 3, pp. 16-31, 2021.

[30] M. Almiani, A. AbuGhazleh, A. Al-Rahayfeh, S. Atiewi, A. Razaque, "Deep recurrent neural network for IoT intrusion detection system", *Simulation Modelling Practice and Theory*, vol. 101, p. 102031, 2020.

[31] G. Kumar, "Evaluation metrics for intrusion detection systems-a study", *Evaluation*, vol. 2, no. 11, pp.11-17, 2014.

[32] Y. Imrana Y. Xiang L. Ali, and Z. Abdul-Rauf, "A bidirectional LSTM deep learning approach for intrusion detection", *Expert Systems with Applications*, vol. 185, p. 115524, 2021.

[33] J. Liu, B. Kantarci, and C. Adams "Machine learning-driven intrusion detection for Contiki-NG-based IoT networks exposed to NSL-KDD dataset", In *Proceedings of the 2nd ACM workshop on wireless security and machine learning*, 2020 Jul 13, pp. 25-30.

[34] S. Gamage, J. Samarabandu, "Deep learning methods in network intrusion detection: A survey and an objective comparison", *Journal of Network and Computer Applications*, vol. 169, p.102767, 2020.

[35] S. Al-Emadi A. Al-Mohannadi, and F. Al-Senaid, "Using deep learning techniques for network intrusion detection", In *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT),* IEEE, 2020 Feb 2, pp. 171-176

[36] M. Azizjon A. Jumabek, and W. Kim, "1D CNN-based network intrusion detection with normalization on imbalanced data", In *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, IEEE, 2020 Feb 19, pp. 218-224.

[37] S. M. Kasongo, and Y. Sun, "Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset", *Journal of Big Data*, vol. 7, no. 1, pp. 1-20, 2020.

[38] P. R. Kanna and P. Santhi, "Unified deep learning approach for efficient intrusion detection system using integrated spatial–temporal features", *Knowledge-Based Systems*, vol. 226, p.107132, 2021.

[39] H. G. Gülmez and P. Angın, P. "A study on the efficacy of deep reinforcement learning for intrusion detection", *Sakarya University Journal of Computer and Information Sciences*, vol. 4, no. 1, pp. 11-25, 2021.

Hossein Faghih Aliabadi was born in 1995 in Mazandaran and received his M.Sc. in Computer Engineering from the University of Urmia, Iran in 2022 respectively. He has served as a reviewer for several journals and his research interests include IoT, Cloud Computing, and Deep learning.