A Snapshot Ensemble Deep Neural Network Model for Attack Detection in Industrial Internet of Things

Hossein Mohammadi Rouzbahani, Amir Hossein Bahrami, and Hadis Karimipour

1 Introduction

In the past few years, as a result of developments in the field of electronics and improvements of wireless systems, the term Internet of Things (IoT) emerged. The opportunity to connect devices together and share information and data while performing their individual tasks without being bound to locations and physical equipment [1].

Industrial Internet of Things (IIoT) is a new application of the Internet of Things (IoT) in the industrial sector. The IIoT enables an enterprise to perform operations in an efficient way while maintaining quality and validation [2]. IIoT makes monitoring and maintenance tasks more convenient, which will be discussed under the category of smart manufacturing systems [3]. By integrating Cyber-Physical Systems CPS, a smart manufacturing execution system can be created such that, documents all data obtained from production and performs decision-making based on predictions on the data for better and optimized future steps [4]. IoT has been progressively used in different sectors of the industry and created a new revolution, IIoT or Industry 4.0 [5, 6], which improves the efficiency, security and productivity in the industry [7–10]. Based on the environment and the purpose of its application, IIoT can have different architectures, but generally, it can be described in a four-layered architecture, as can be observed in Fig. 1.

The physical layer, consists of all physical elements such as actuators, sensors, machines, etc. The network layer consists of communication networks and

H. M. Rouzbahani (✉) · H. Karimipour
Department of Electrical and Software Engineering, University of Calgary, Calgary, AB, Canada
e-mail: hmoham15@uoguelph.ca; hadis.karimipour@ucalgary.ca

A. H. Bahrami
Peter the Great St. Petersburg Polytechnic University, St. Petersburg, Russia

H. Karimipour, F. Derakhshan (eds.), *AI-Enabled Threat Detection and Security Analysis for Industrial IoT*, https://doi.org/10.1007/978-3-030-76613-9_10

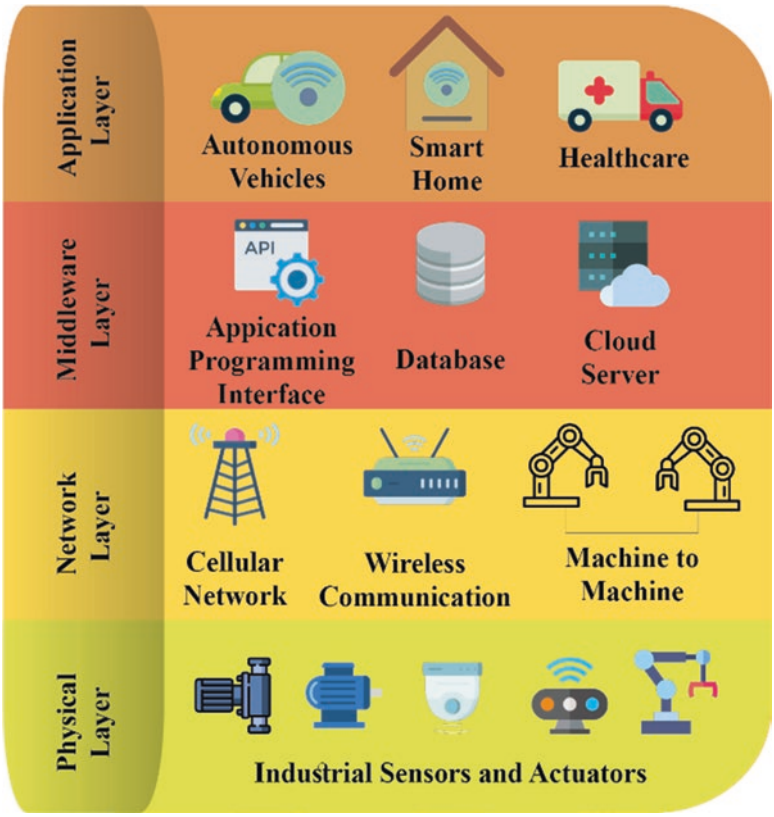

Fig. 1 Four layered architecture IIoT

protocols. The middleware layer makes the communication between the second layer and the first layer possible. It consists of an application programming interface (API), database, cloud server, etc. The fourth layer, the application layer, describes the application of the IIoT. Some instances of the applications are autonomous vehicles, smart home, healthcare, etc. [11–15]. for instance, Rouzbahani et al. proposed an Incentive-based Demand Response Optimization (IDRO) model in order to efficiently schedule household appliances for minimum usage during peak hours [16, 17], which demonstrates noticeable improvements in power factor and cost-saving during peak hours for individual households.

While IIoT is an excellent solution to facilitate industrial processes, it creates new challenges with its application. As the devices start to operate simultaneously, they generate valuable data for online monitoring and control of the system, which can also be used by attacker to manipulate the system performace [18, 19]. There are several attacks that can be performed in IIoT, one of such is cyber-attacks, and this type of attack has other variations itself, such as Denial of Service (DoS), Datatype Probing (DP), Scan, and etc.

In order to perform data processing and analysis, Machine Learning (ML) is preferred over the traditional methods due to the huge quantity of data that is being generated throughout the operations. ML is considered to be a useful paradigm for detecting security threats [20]. Apruzzese et al. [21] conducted an investigation on the effectiveness of ML for cyber threat detection to find and address the limitation of it in such tasks. Lee et al. [22] conducted the same topic of research, but the focus was on the reduction of error in the solution. These are a few instances to show the effectiveness of ML in the area of detection and classification of cyber-security threats.

In this paper, we proposed a Snapshot Ensemble Deep Neural Network (SEDNN) for cyber-attack detection. The model has high accuracy in the detection of cyber threats. It is worth noticing that the classification of the attacks was not considered in this paper. In sect. II, some previous work on the same area of study will be presented, section III will be devoted to the methodology, in sect. IV the results will be presented and in sect. V conclusion and future steps will be discussed.

2 Previous Works in IIoT Security

As the system becomes more complex and data quantity becomes enormous [23–25], the computation and control become more challenging, resulting in traditional methods not to perform as expected because of latency and long response time [26, 27]. ML algorithms improve industrial processes' security and reliability and are rapidly used to detect and address security threats in IIoT [28, 29]. Previous studies in the area of ML application in IIoT security show promising results in using ML algorithms for addressing cyber threats in IIoT.

Rouzbahani, Karimipour and Lei [30] proposed an Ensemble Deep Convolutional Neural Network (EDCNN) model for electricity theft detection in smart grids. In this study, they used a dataset consisting of the daily consumption of 42,372 users. They used an unbalanced dataset in which 8% of customers were attackers, and the rest were normal users. They compared the results with other models and concluded that EDCNN could detect electricity theft in smart grids with an accuracy of 0.981, which indicates that the model is precise.

Farahnakian and Heikkonen [31] approached intrusion detection by presenting a Deep Auto-Encoder (DAE) based system. They used the model on the KDD-CUP'99 dataset and achieved an accuracy of 94.71% for attack detection, which then they concluded that their approach obtained better results as opposed to other deep learning-based approaches. Moukhafi et al. [32] chose a novel hybrid genetic algorithm and support vector machine with the particle swarm optimization feature selection approach for detecting Denial of Service (DoS) attack detection, which they implemented on KDD 99 dataset and obtained an accuracy of 96.38%. Rouzbahani et al. [33] presented research on using ML algorithms for the classification of False Data Injection (FDI) attacks in CPS.

Vajayanand et al. [34] proposed a support vector machine (SVM)-based model, and by doing so, they improved the classification algorithm. They used the ADFA-LD dataset for the implementation of their model and obtained an accuracy of 94.51%. In the research of Khalvati et al. [35], they proposed the SVM and Bayesian model to successfully classify IoT attacks. They conducted research with their proposed model on KDD CUP 99 dataset and achieved an accuracy of 91.50%. Li et al. [36] proposed a bidirectional long and short-term memory network with a multi-feature layer (B-MLSTM) on the classical IIoT datasets: CTU-13 [37], Gas-Water [38], and AWID [39] in order to detect low-frequency and multi-stage attacks in IIoT. After the implementation of the model, an accuracy of 95.01% on the CTU-13 and 97.58% on AWID was obtained. Rouzbahani et al. [40] conducted research and performed cyber-attack detection in smart cyber-physical grids by using different ML algorithms, which resulted in a great performance for Random forest K-Nearest Neighbor (KNN).

Overall, investigations show that ML can efficiently and precisely detect security threats in IIoT. What is worth noticing is that the datasets in these studies are classical datasets that are available on the internet and are considered to be outdated. We are obligated to use new datasets because of the modern security requirements of IIoT. This paper proposes a modern ML model that will be implemented on newer datasets and will also address the compatibility of the model with resource-constrained devices.

3 Methodology

In this section, a brief description of the dataset has been presented. The section will then continue with a description of the preprocessing of the dataset, the proposed model, and evaluation parameters that were considered to evaluate the model's performance.

3.1 Dataset

The dataset used in this paper is an open-source dataset obtained from Kaggle [41]. It was provided by Pahl et al. [42]. This dataset contains communications between different IoT nodes, sensors and applications. In this dataset, multiple attacks were performed on the IIoT applications, for example, "spying", "wrong setup" and etc., which resulted in an anomaly in some of the 357,952 data samples [43, 44]. This paper tried to address the cyber-attack performed on the data. Classification of the attacks will be discussed in another paper.

3.2 Preprocessing of Data

In order to obtain acceptable results from ML models, a comprehensive dataset is the main requirement. Most of the time in data mining is devoted to data processing [45], and the most essential problem in data processing is missing values, which can be caused by various reasons such as power outage, sensor damage or cyber-attacks [46].

In this dataset, there are missing values. Deleting them can result in losing valuable data on other columns. Therefore, the missing values need to be replaced. Figure 2 shows a diagram of the algorithm for attack detection. The processing of replacing the missing values is as follows:

3.2.1 Features

First, we need to select the features that we want to create our model based on. Table 1 shows the features that were selected. It demonstrates which methods were considered in order to encode the features as well.

3.2.2 Replacing Missing/NaN Values

Backward Difference Encoding: this coding system is one of the coding systems of categorical encoding. When a regression is performed on a set of variables with K categories, these variables will enter the regression as a sequence of K-1 dummy variables. The regression coefficient of these K-1 variables corresponds to linear hypotheses on the cell means.

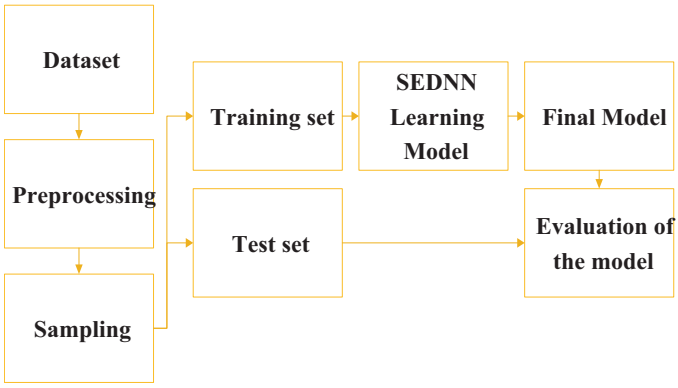

Fig. 2 Diagram of the attack detection algorithm

Table 1 Methods for feature encoding

Feature	Method
Source ID	Label encoding
Source type	Label encoding
Source location	Label encoding
Destination service type	Label encoding
Destination location	Label encoding
Accessed node type	Label encoding
Operation	Label encoding
Source address	Backward difference encoding
Destination service address	Backward difference encoding
Accessed node address	Backward difference encoding

Table 2 Replacing missing values

Variable	Assigned value
Blank	−2
False	−1
True	0.1
20	20
None	0

In this coding system, the mean of the dependent variable for one level of the categorical variable is compared to the mean of the dependent variable for the prior adjacent level.

Label Encoding: in this encoding, a number will be assigned to each variable. The model should be able to understand the difference between "blank," "False," and "None" variables. Therefore we cannot assign 0 to all of them. Table 2 demonstrates the values which were assigned to each variable.

3.3 Snapshot Ensemble Deep Neural Network

In this paper, a Snapshot Ensemble Deep Neural Network (SEDNN) was proposed in order to detect cyber-attacks on the dataset. The disadvantage of an ordinary Ensemble Deep Neural Network (EDNN) is a high computational cost, so that with ordinary hardware, the time of the training and testing will be high. In order to overcome this problem, this paper approached this problem with an SEDNN model [47]. The difference between and ordinary EDNN and SEDNN is that every time the SEDNN reaches a local minimum, it will save the model's weights and biases and continues to do so until the model finds the optimal minimum, resulting in a set of neural networks with low errors. After this process, the model will ensemble all models in this set and obtains the perfect model. The algorithm uses Gradient Descent in order to find the minimum in each step. Two types of activation functions

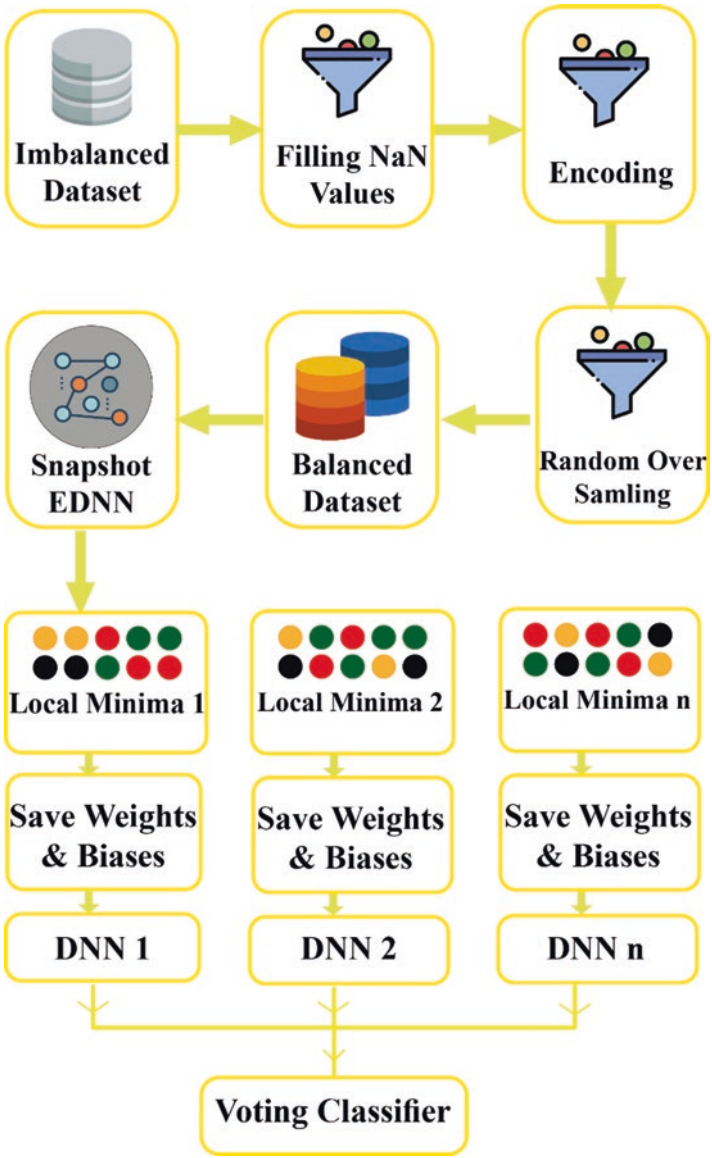

Fig. 3 Architecture of The Proposed Algorithm

were considered for the DNN layers, for the first three layers, a "Relu" activation function was assigned, and for the last layer, a "Sigmoid" function was considered to conduct a binary classification in this paper. As an output, each of the test set data will be given a label of 0 (Normal) or 1 (Attack). Figure 3 shows a visualization of the proposed algorithm, and the architecture of the DNNs can be observed in Fig. 4.

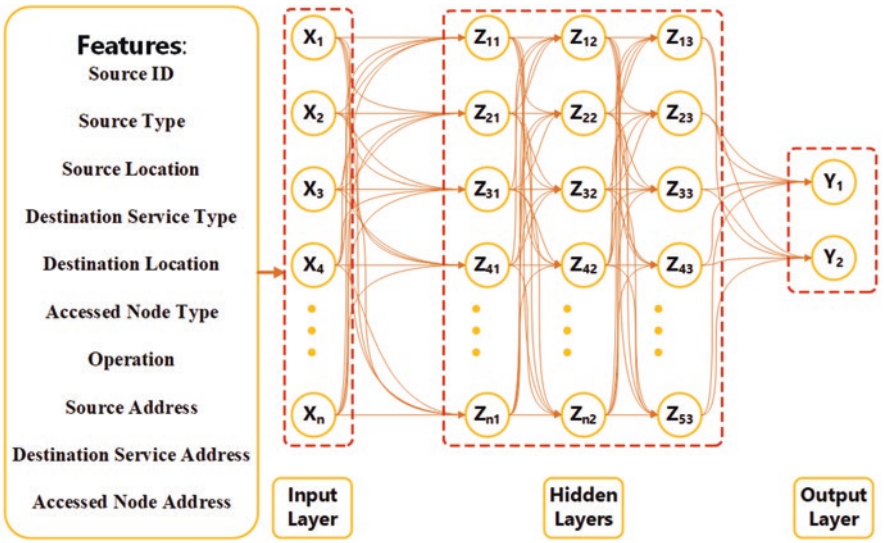

Fig. 4 Deep Neural Network Architecture

3.4 Evaluation Parameters

In order to evaluate a ML model, there are some parameters that can be used. In this section, these parameters will be briefly explained. There are some terms used in the calculation of the evaluation parameters that need to be defined.

True positive is the resulting term where the model correctly predicted the positive class. True negative is the resulting term where the model correctly predicted the negative class. False-positive is the resulting term where the model incorrectly predicted the positive class. False-negative is the resulting term where the model incorrectly predicted the negative class.

Accuracy is the most common measure for evaluating the ML model, and it is defined as the ratio of correctly predicted results to the total predicted results. It may be implied that the higher the accuracy, the more precise model. This is not true in all possible cases. This assumption is only correct when there are symmetric datasets where false positives and false negatives are almost the same. Therefore, we have to look for other parameters to evaluate our model more accurately. The mathematical formula for accuracy calculation is described in Eq. 1.

$$Accuracy = \frac{T_{Pos} + T_{Neg}}{T_{Pos} + T_{Neg} + F_{Pos} + F_{Neg}} \tag{1}$$

Precision is the ratio of true positives to all optimistic predictions. The formula for precision calculation is described in Eq. 2. High precision will result in low false-positive rate.

$$Precision = \frac{T_{Pos}}{T_{Pos} + F_{Pos}} \tag{2}$$

The recall is the ratio between true positive to all predictions (true positive and false negative) of the same class. The formula for recall calculation is described in Eq. 3.

$$Recall = \frac{T_{Pos}}{T_{Pos} + F_{Neg}} \tag{3}$$

F1-Score is the weighted average of Precision and Recall. Therefore, it takes false positives and false negatives into account. The formula for F1-score calculation is described in Eq. 4.

$$F1 - Score = \frac{2 \times \left(Precision + Recall \right)}{Precision + Recall} \tag{4}$$

4 Implementation and Results

In this section, hardware and software equipment will be discussed. The section will continue to present the results in detail.

4.1 Software and Hardware

The proposed model has been tested using Python 3.7.4 on a system with an Intel Core i7-97580H CPU, 16.0 GB of RAM, and the model's design is structured based on TensorFlow. In order to analyze the performance of the model, we need to obtain the confusion matrix, which will offer us true positive, false positive, true negative and false negative.

4.2 Results

The general form of a confusion matrix can be observed in Table 3.

In this research, different classifiers have been tested on the dataset in order to compare the results and accuracy percentage. In Table 4, the confusion matrix of the proposed model can be observed; moreover, Table 5 presents the proposed model's performance with evaluation parameters.

Table 3 Confusion matrix

Actual/Detected	Normal	Attacker
Normal	T_{Pos}	F_{Neg}
Attacker	F_{Pos}	T_{Neg}

Table 4 Confusion matrix of the proposed model

Actual/Detected	Normal	Attacker
Normal	49,478	7123
Attacker	3285	50,613

Table 5 Result comparison of different classifiers

Classifier	Accuracy	Precision	Recall	F1-Score
SEDNN	0.9058	0.8742	0.9377	0.9048

As it can be implied from Tables 4 and 5, the model presents promising results. Obtained accuracy of 90.58% and F1-Score of 90.48% show the great performance of SEDNN in detecting cyber-attacks in IIoT applications. Figures 5 and 6 show the accuracy and loss rate of the model.

In Fig. 5, the accuracy is not stable, and this is caused by changes of DNN between each time it reaches a local minimum, the algorithm uses a new DNN with new weights and biases. Overall, the test set's accuracy is higher than the train set, which shows the model's outstanding performance.

In Fig. 6, we can observe the loss diagram of the train and test set. The nose in the test diagram was caused by utilizing multiple DNN in between each local minimum, as was described before. It can be observed that overall, the loss of the test set is lower than the train set, which shows the model is performing great.

5 Conclusion and Future Work

In this paper, a SEDNN model was proposed for cyber-attack detection in industrial IoT systems. As the model searches for a global minimum, upon finding every local minimum, it will save the weights and biases of that particular DNN (Snapshots), and when it reaches the global minima, it generates the best possible model from the set of DNNs, instead of training and testing different models on the entire dataset. The proposed model has a high accuracy of 90.58%, demonstrating the model's excellent performance in cyber-attack detection. The model was tested on an open-source dataset, DS2OS, which showed promising results. The dataset consists of communication between different IoT nodes such as sensors and actuators. In the future steps, more real-time experiments and ìnvestigations can be conducted with the proposed model to test the model on real IIoT systems; furthermore classification of the attacks with the proposed model will be conducted in future researches.

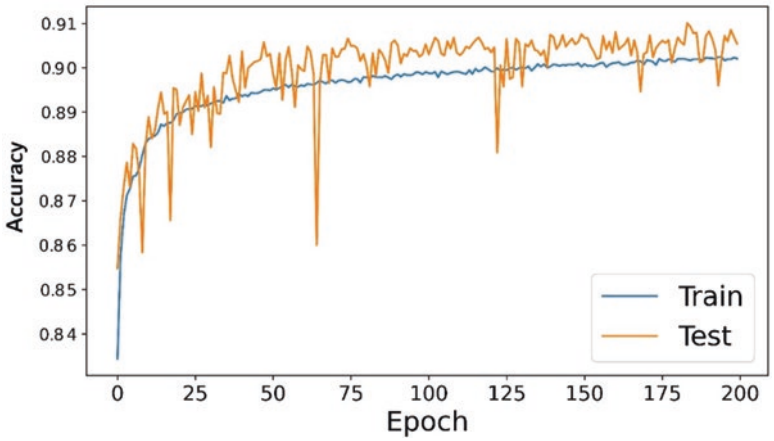

Fig. 5 SEDNN accuracy rate

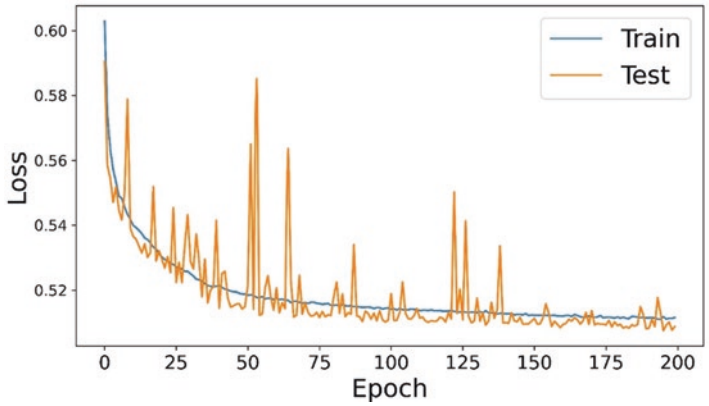

Fig. 6 SEDNN loss rate

References

1. E. Borgia, "The Internet of Things vision: Key features, applications and open issues," Comput. Commun., vol. 54, pp. 1–31, 2014, https://doi.org/10.1016/j.comcom.2014.09.008.
2. A. Alabasi, H. Karimipour, A. Dehghantanha, "An Ensemble Deep Learning-based Cyber-Attack Detection in Industrial Control System", IEEE Access, vol. 8, pp. 83965-83973, April. 2020. doi: https://doi.org/10.1109/ACCESS.2020.2992249.
3. J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications," IEEE Internet Things J., vol. 4, no. 5, pp. 1125–1142, 2017, doi: https://doi.org/10.1109/JIOT.2017.2683200.
4. V. V Potekhin, A. H. Bahrami, and B. Katalinič, "Developing manufacturing execution system with predictive analysis," IOP Conf. Ser. Mater. Sci. Eng., vol. 966, p. 12117, 2020, doi: https://doi.org/10.1088/1757-899x/966/1/012117.

5. J. Sakhnini, H. Karimipour, A. Dehghantanha, R. Parizi, G. Srivastava "Security Aspects of Internet of Things Aided Smart Grids: a Bibliometric Survey", Internet of Things Journal—Elsevier, pp. 1–15, Sept. 2019. https://doi.org/10.1016/j.iot.2019.100111
6. F. Anwaar, N. Iltaf, H. Afzal, and R. Nawaz, "HRS-CE: A hybrid framework to integrate content embeddings in recommender systems for cold start items," J. Comput. Sci., vol. 29, pp. 9–18, 2018, doi: https://doi.org/10.1016/j.jocs.2018.09.008.
7. S. M. Tahsien, H. Karimipour, P. Spachos, "Machine Learning Based Solutions for Security of Internet of Things (IoT): A Survey", Journal of Network and Computer Applications—Elsevier, vol. 161, pp. 1–18, April. 2020. https://doi.org/10.1016/j.jnca.2020.102630
8. A. Yazdinejad, R. M. Parizi, A. Dehghantanha, H. Karimipour, G. Srivastava, and M. Aledhari, "Enabling Drones in the Internet of Things with Decentralized Blockchain-based Security," IEEE Internet of Things Journal, 2020.
9. E. Nowroozi, A. Dehghantanha, R. M. Parizi, and K.-K. R. Choo, "A survey of machine learning techniques in adversarial image forensics," *Computers & Security,* vol. 100, p. 102092, 2021.
10. A. Yazdinejad, G. Srivastava, R. M. Parizi, A. Dehghantanha, H. Karimipour, and S. R. Karizno, "SLPoW: Secure and Low Latency Proof of Work Protocol for Blockchain in Green IoT Networks," in 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), 2020, pp. 1–5: IEEE.
11. M. Aledhari, R. M. Parizi, A. Dehghantanha and K. R. Choo, "A Hybrid RSA Algorithm in Support of IoT Greenhouse Applications," *2019 IEEE International Conference on Industrial Internet (ICII)*, Orlando, FL, USA, 2019, pp. 233–240, https://doi.org/10.1109/ICII.2019.00049
12. A. Yazdinejadna, R. M. Parizi, A. Dehghantanha, and M. S. Khan, "A kangaroo-based intrusion detection system on software-defined networks," *Computer Networks,* vol. 184, p. 107688, 2021.
13. A. Yazdinejad, R. M. Parizi, A. Dehghantanha, and K.-K. R. Choo, "Blockchain-enabled authentication handover with efficient privacy protection in SDN-based 5G networks," IEEE Transactions on Network Science and Engineering, 2019.
14. M. Aledhari, R. Razzak, R. M. Parizi and F. Saeed, "Federated Learning: A Survey on Enabling Technologies, Protocols, and Applications," in IEEE Access, vol. 8, pp. 140699–140725, 2020, doi: https://doi.org/10.1109/ACCESS.2020.3013541
15. A. Yazdinejad, G. Srivastava, R. M. Parizi, A. Dehghantanha, K.-K. R. Choo, and M. Aledhari, "Decentralized authentication of distributed patients in hospital networks using blockchain," IEEE journal of biomedical and health informatics, vol. 24, no. 8, pp. 2146–2156, 2020.
16. H. M. Ruzbahani and H. Karimipour, "Optimal incentive-based demand response management of smart households," in 2018 IEEE/IAS 54th Industrial and Commercial Power Systems Technical Conference (I&CPS), 2018, pp. 1–7, https://doi.org/10.1109/ICPS.2018.8369971.
17. H. M. Ruzbahani, A. Rahimnejad, and H. Karimipour, "Smart Households Demand Response Management with Micro Grid," in 2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), 2019, pp. 1–5, https://doi.org/10.1109/ISGT.2019.8791595.
18. E. Modiri, A. Azmoodeh, A. Dehghantanha, H. Karimipour, "Fuzzy Pattern Tree for Edge Malware Detection and Categorization in IoT", Journal of Systems Architecture, vol. 9, pp. 1–7, Jan. 2018. https://doi.org/10.1016/j.sysarc.2019.01.017
19. Z. Ling, J. Luo, Y. Xu, C. Gao, K. Wu, and X. Fu, "Security Vulnerabilities of Internet of Things: A Case Study of the Smart Plug System," IEEE Internet Things J., vol. 4, no. 6, pp. 1899–1909, 2017, doi: https://doi.org/10.1109/JIOT.2017.2707465.
20. M. Hazrati, H. Karimipour, A. N. Jahromi, A. Dehghantanha, "Ensemble sparse representation-based cyber threat hunting for security of smart cities", Computer & Electrical Engineering Journal- Elsevier (IF: 2.6), Dec. 2020. https://doi.org/10.1016/j.compeleceng.2020.106825
21. G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, "On the effectiveness of machine and deep learning for cyber security," in 2018 10th International Conference on Cyber Conflict (CyCon), 2018, pp. 371–390, https://doi.org/10.23919/CYCON.2018.8405026.

22. J. Lee, J. Kim, I. Kim, and K. Han, "Cyber Threat Detection Based on Artificial Neural Networks Using Event Profiles," IEEE Access, vol. 7, pp. 165607–165626, 2019, doi: https://doi.org/10.1109/ACCESS.2019.2953095.
23. A. Yazdinejad, H. HaddadPajouh, A. Dehghantanha, R. M. Parizi, G. Srivastava, and M.-Y. Chen, "Cryptocurrency malware hunting: A deep recurrent neural network approach," Applied Soft Computing, vol. 96, p. 106630, 2020.
24. A. Namavar Jahromi et al., "An improved two-hidden-layer extreme learning machine for malware hunting," Computers & Security, vol. 89, p. 101655, 2020.
25. A. Yazdinejad, R. M. Parizi, A. Dehghantanha, G. Srivastava, S. Mohan, and A. M. Rababah, "Cost optimization of secure routing with untrusted devices in software defined networking," Journal of Parallel and distributed Computing, vol. 143, pp. 36–46, 2020
26. J. Chen et al., "Collaborative Trust Blockchain Based Unbiased Control Transfer Mechanism for Industrial Automation," IEEE Trans. Ind. Appl., vol. 56, no. 4, pp. 4478–4488, 2020, doi: https://doi.org/10.1109/TIA.2019.2959550.
27. W. G. Hatcher and W. Yu, "A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends," IEEE Access, vol. 6, pp. 24411–24432, 2018, doi: https://doi.org/10.1109/ACCESS.2018.2830661.
28. M. Lavassani, S. Forsström, U. Jennehag, and T. Zhang, "Combining Fog Computing with Sensor Mote Machine Learning for Industrial IoT," Sensors, vol. 18, p. 1532, May 2018, doi: https://doi.org/10.3390/s18051532.
29. A. N. Jahromi, A. Dehghantanha, R. Choo, H. Karimipour, R. Parizi, "An Improved Two-Hidden-Layer Extreme Learning Machine for Malware Hunting", Computer and Security, vol. 89, pp. 1–11, Sept. 2019. https://doi.org/10.1016/j.cose.2019.101655
30. H. M. Rouzbahani, H. Karimipour, and L. Lei, "An Ensemble Deep Convolutional Neural Network Model for Electricity Theft Detection in Smart Grids," in 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2020, pp. 3637–3642, https://doi.org/10.1109/SMC42975.2020.9282837.
31. F. Farahnakian and J. Heikkonen, "A deep auto-encoder based approach for intrusion detection system," in 2018 20th International Conference on Advanced Communication Technology (ICACT), 2018, pp. 178–183, https://doi.org/10.23919/ICACT.2018.8323688.
32. Mehdi Moukhafi, Khalid El Yassini, and Seddik Bri, "A novel hybrid GA and SVM with PSO feature selection for intrusion detection system," I. J. Adv. Sci. Res. Eng. (ISSN 2454—8006), vol. 4, no. 5 SE-, pp. 129–134, May 2018, https://doi.org/10.31695/IJASRE.2018.32724.
33. H. Mohammadi Rouzbahani, H. Karimipour, A. Rahimnejad, A. Dehghantanha, and G. Srivastava, "Anomaly Detection in Cyber-Physical Systems Using Machine Learning BT—Handbook of Big Data Privacy," K.-K. R. Choo and A. Dehghantanha, Eds. Cham: Springer International Publishing, 2020, pp. 219–235.
34. R. Vijayanand, D. Devaraj, and B. Kannapiran, "A novel intrusion detection system for wireless mesh network with hybrid feature selection technique based on GA and MI," J. Intell. Fuzzy Syst., vol. 34, pp. 1243–1250, 2018, doi: https://doi.org/10.3233/JIFS-169421.
35. L. Khalvati, M. Keshtgary, and N. Rikhtegar, "Intrusion Detection based on a Novel Hybrid Learning Approach," J. AI Data Min., vol. 6, no. 1, pp. 157–162, 2018, doi: https://doi.org/10.22044/jadm.2017.979.
36. X. Li, M. Xu, P. Vijayakumar, N. Kumar, and X. Liu, "Detection of Low-Frequency and Multi-Stage Attacks in Industrial Internet of Things," IEEE Trans. Veh. Technol., vol. 69, no. 8, pp. 8820–8831, 2020, doi: https://doi.org/10.1109/TVT.2020.2995133.
37. P. Bereziński, B. Jasiul, and M. Szpyrka, "An Entropy-Based Network Anomaly Detection Method," Entropy, vol. 17, no. 4, pp. 2367–2408, Apr. 2015, doi: https://doi.org/10.3390/e17042367.
38. T. Morris and W. Gao, "Industrial Control System Traffic Data Sets for Intrusion Detection Research BT—Critical Infrastructure Protection VIII," 2014, pp. 65–78.

39. M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” in 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009, pp. 1–6, https://doi.org/10.1109/CISDA.2009.5356528.
40. H. M. Rouzbahani, Z. Faraji, M. Amiri-Zarandi, and H. Karimipour, “AI-Enabled Security Monitoring in Smart Cyber Physical Grids”, Security of Cyber-Physical Systems: Vulnerability and Impact, Eds. Cham: Springer International Publishing, 2020, pp. 145–167.
41. M.-O. P. and F.-X. Aubet., “Ds2Os Traffic Traces IoT Traffic Traces Gathered in a The Ds2Os IoT Environment. [Online].,” 2018. https://www.kaggle.com/francoisxa/ds2ostraffictraces.
42. M. Pahl and F. Aubet, “All Eyes on You: Distributed Multi-Dimensional IoT Microservice Anomaly Detection,” in 2018 14th International Conference on Network and Service Management (CNSM), 2018, pp. 72–80, [Online]. Available: http://doi.ieeecomputersociety.org/.
43. M. Hasan, M. M. Islam, M. I. I. Zarif, and M. M. A. Hashem, “Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches,” Internet of Things, vol. 7, p. 100059, 2019, doi: https://doi.org/10.1016/j.iot.2019.100059.
44. O. Brun, Y. Yin, and E. Gelenbe, “Deep Learning with Dense Random Neural Network for Detecting Attacks against IoT-connected Home Environments,” Procedia Comput. Sci., vol. 134, pp. 458–463, 2018, doi: https://doi.org/10.1016/j.procs.2018.07.183.
45. F. Honghai, C. Guoshun, Y. Cheng, Y. Bingru, and C. Yumei, “A SVM Regression Based Approach to Filling in Missing Values BT—Knowledge-Based Intelligent Information and Engineering Systems,” 2005, pp. 581–587.
46. Sh. Yousefi, F. Derakhshan, H. Karimipour, “Applications of Big Data Analytics and Machine Learning in the Internet of Things”, Handbook of Big Data and Privacy, Springer Books, pp. 1–32, Feb. 2020. https://doi.org/10.1007/978-3-030-38557-6_5
47. G. Huang, Y. Li, G. Pleiss, Z. Liu, J. Hopcroft, and K. Weinberger, “Snapshot Ensembles: Train 1, get M for free,” Mar. 2017.