The International Journal of Advanced Manufacturing Technology (2022) 123:1973–1983
https://doi.org/10.1007/s00170-022-10329-6

ORIGINAL ARTICLE

A deep hybrid learning model for detection of cyber attacks in industrial IoT devices

Mohammad Shahin[1] · F. Frank Chen[1] · Ali Hosseinzadeh[1] · Hamed Bouzary[1] · Rasoul Rashidifar[1]

Received: 12 October 2022 / Accepted: 13 October 2022 / Published online: 26 October 2022
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

Abstract
With the rapid advancement of wireless technology, the problem of cybersecurity monitoring and detection of cyber-attacks has been receiving widespread attention from industry and academia. The consequences of an undetected cyber-attack in a manufacturing system are not limited to intellectual property theft and cost. It may include destroying equipment, changing product plans, or altering processes. This paper proposes a deep hybrid learning model to improve network intrusion detection systems. To this end, initially, the data set is normalized and preprocessed. Afterward, deep hybrid learning models integrating Attention-based Long Short Term Memory (ALSTM) and Fully Convolutional Neural Network (FCN) with Gradient Boosting, such as Extreme Gradient Boosting (XGBoost) and Adaptive Boost (AdaBoost) are constructed to detect anomalies in traffic data of industrial internet of things (IoT) devices, successfully. The proposed model managed to detect cybersecurity threats in seven different Industrial Internet of Things (IIoT) devices with high-performance measures. The results reveal that deep hybrid learning can ideally detect cyber-security attacks and be versatile in detecting different types of attacks.

Keywords Cybersecurity · Industry 4.0 · Big Data · Machine learning · Deep learning

1 Introduction

The continuous integration of sensor technology into traditional manufacturing systems [1] has increased the risk of exposure to cyber-attacks [2–4]. As a result, cyber-attacks are becoming more frequent and capable of inflicting damage [5–7]. Moreover, these attacks do not just stop at the theft of Intellectual Property (IPs) but go beyond that to include destroying manufacturing equipment, changing product design plans and schematics, and altering manufacturing processes [8]. For example, in January 2022, more than eleven cyber-attacks were reported by the Center for Strategic and International Studies (CSIS) [9]. These attacks ranged from breaching state-owned servers across the globe, as in the case of Belarusian Railway, to hacking privately-owned companies, as in the case of attacking a Minecraft tournament [9]. This shows that cyber-attacks are capable of reaching various sectors of the economy of any country on earth. However, among all these affected sectors, manufacturing was the most targeted sector [7]. Usually, these attacks with different attributions (taxonomy) in terms of goals, targeted systems, and tools [10] have targeted the organizations through their data traffic via cyberspace, intending to break up, debilitate, damage, or take over a server's environment and infrastructure [11]. To this end, various scholars have proposed different models to detect such attacks to reduce the damage caused by them, such as network intrusion detection systems (NIDS) [12]. In this paper, we propose the use of a fusion approach, one that combines machine learning (ML) with deep learning (DL) to build a state-of-the-art deep hybrid learning (DHL) model to detect cyber security threats and improve the deterrence level of malicious attacks in industrial IoT devices.

2 Background

ML is an artificial intelligence (AI) branch that uses algorithms to analyze data, extract information from data, and use that information to make informed decisions. DL, on the other hand, is a subfield of ML that assembles algorithms in layers to create an Artificial Neural Network (ANN) that can learn and make intelligent decisions on its own. The

✉ F. Frank Chen
FF.Chen@utsa.edu

[1] Mechanical Engineering Department, The University of Texas at San Antonio, San Antonio, TX, USA

simplest ANN structure consists of nodes network (neurons or perceptron) arranged in three layers: input, hidden, and output. The word "Learning" indicates that both ML and DL models get progressively better without prior knowledge. The main difference is that ML models require intervention when their AI algorithm returns an inaccurate prediction. This intervention can be referred to as feature engineering, which improves the final accuracy and performance of the AI algorithm. On the other hand, with DL models, the AI algorithm can determine whether a prediction is accurate or not through its neural network requiring more computational power. In addition, DL is susceptible to an over-fitting problem when its fed with insufficient data.

A DHL uses a neural network of DL algorithm to extract features and standard ML algorithm for classification resulting in high accuracy without over-fitting and without using too many computational resources [13–15]. By the end of 2021, over 96 billion dollars have been spent on implementing ML and DL models in cyber security [16]. Cybersecurity experts observe streams of network traffic by applying data analytics in real-time to detect intrusions [17–19]. ML has been used in manufacturing to detect cyber-physical attacks, surface defects, weld defects, and machine failures in preventative maintenance [20–22]. DL has been used in manufacturing to detect hidden hardware Trojans in electronics manufacturing assembly lines and supply chains [23]. Malware analysis using ML and DL models effectively detects cybersecurity threats [24] that could affect manufacturing systems. Figure 1 shows the relationship between DL, ML, DHL, and AI. Different strategies and frameworks [25] were deployed to enhance a manufacturing enterprise's information security front [26].

Our proposed DHL algorithms combine the high accuracy of gradient boosting algorithms from ML and the efficient feature extraction of neural networks to end with a model of low misclassification rates and a fast detection speed. An example of gradient boosting algorithms is Extreme Gradient Boosting (XGBoost) has been used to detect network intrusions based on traffic data anomalies [27–29]. XGBoost is considered a form of gradient boosting decision tree that is regarded as practical along with its high speed and performance and low dependency on computational resources [30–32]. Another example of gradient boosting algorithm is Adaptive Boost (AdaBoost). AdaBoost has been used in network intrusion detection systems to detect phishing attempts [33] and Denial of Service (DoS) attacks [34]. Like XGBoost, it utilizes a low amount of computational resources to achieve a high-performance detection level by keeping on iterating until reaching the lowest error value possible [35, 36].

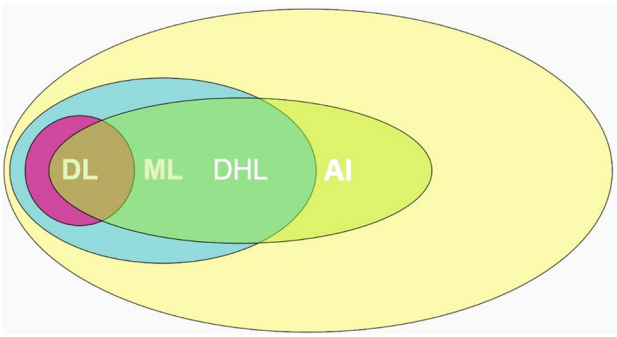

Fig. 1 Relationship between DL, ML, DHL, and AI

ANN has been used in various aspects of scientific applications in both supervised learning methods and unsupervised learning methods [37]. ANN allows AI applications to program themselves to solve data-based problems. Multilayer Perceptron (MLP) is a feedforward (FF) ANN type. MLP can use linear and non-linear transformation to draw relationships between input data and output data through tuning the random weights of its neuron layers [38–40]. Allowing ANN to propagate data (neuron signals) in backward but cyclic flow from later processing to earlier stages and including it in an ongoing process results in a Recurrent Neural Network (RNN). RNN has been widely used in numerous applications, including detecting malware in network data streams with high performance [41–43]. An RNN capable of remembering much previous information and learning order dependence in sequence prediction to improve the vanishing information problem is referred to as Long Short Term Memory (LSTM) [44]. LSTM has wide use in applications such as speech recognition, text prediction, machine translation, and more [41]. Combining the memory function of LSTM with an attention mechanism to enhance the powerful series data learning ability to pay attention to crucial attributes in longer sequence predictions produces what is known as Attention-based Long Short Term Memory (ALSTM) [45, 46]. Thus, it can achieve higher performance in cybersecurity threat detection [47].

MLP with dense neuron layers that are fully connected (all output neurons are connected to all input neurons) is referred to as Convolutional Neural Network (CNN) [41]. CNN has been used in intrusion detection systems to detect DoS attacks in different datasets with good performance [48] and to detect malware as well [49–51]. On the other hand, CNN with no fully connected layers is referred to as Fully Convolutional Neural Network (FCN) [52]. This difference allows FCN to perform better in learning tasks where there is a vast space of possibilities and will enable FCN to have fewer running times and less consumption of computational resources [53]. All these advantages allowed FCN to be the best choice in detecting fake fingerprints [53] in numerous smartphones models, for example. In our paper, we proposed the use of two new DHL models to detect different

Table 1 List of cybersecurity attacks in the TON_IoT Dataset

Attack type	Brief details
Distributed Denial of Service (DDoS)	A sequence of malicious attacks is used to flood target devices with several botnets, overwhelming the device resources to disrupt future access to their services [62, 63]
Ransomware	A malware attack encrypts a system or service to prevent user access to it to give them back that access for a certain amount of money [64, 65]
Backdoor	It is a passive attack that allows intruders to gain unauthorized access to the compromised device or service [66]
Injection	An attack aims to inject malicious information into a device's source code to gain control over its operations [67]
Cross-Site Scripting (XSS)	An attack compromises the traffic between IIoT devices and their remote Web server by injecting malicious Web scripts [67]
Password cracking	An attack that guesses the password of an IIoT device to compromise them [66, 67]
Scanning	An attack intercepts the data traffic stream between two devices to redirect or manipulate their information [68]

cybersecurity attacks in several IoT applications. The first model consists of an Attention-based Long Short Term Memory Fully Convolutional Neural Network with Extreme Gradient Boosting (ALSTM-FCN with XGBoost), and the second model is an Attention-based Long Short Term Memory Fully Convolutional Neural Network with Adaptive Boost (ALSTM-FCN with AdaBoost).

3 Dataset

TON_IoT dataset is one of the latest generated datasets in the Industry 4.0 world and was released in the year 2020 [54–56]. The dataset was constructed by the School of Engineering and Information technology and the Cyber Range and IoT Labs of the University of New South Wales (UNSW) in Canberra, Australia, in cooperation with the Australian Defence Force Academy (ADFA) [57–61]. The dataset was created for the Industry 4.0 network that includes IoT and IIoT networks. The primary purpose of making this type of dataset was to evaluate the fidelity and efficiency of various cybersecurity applications based on ML, DL, and AI. In our paper, we will assess such applications using DHL. The dataset consists of seven IoT devices: a Fridge sensor, Garage Door sensor, GPS tracker, Motion Light sensor, Modbus communication device, Thermostat sensor, and Weather sensor. Each one of those devices was subjected to the following attacks listed in Table 1, while Fig. 2 shows the percentages of each attack instance for all the seven IoT devices from a total of 407,932 records. Note that Bening indicates normal traffic where no attack has occurred.

4 Methodology

The initial step for any network intrusion detection system is data collection through simulation or from various online resources, such as in our TON_IoT dataset. After

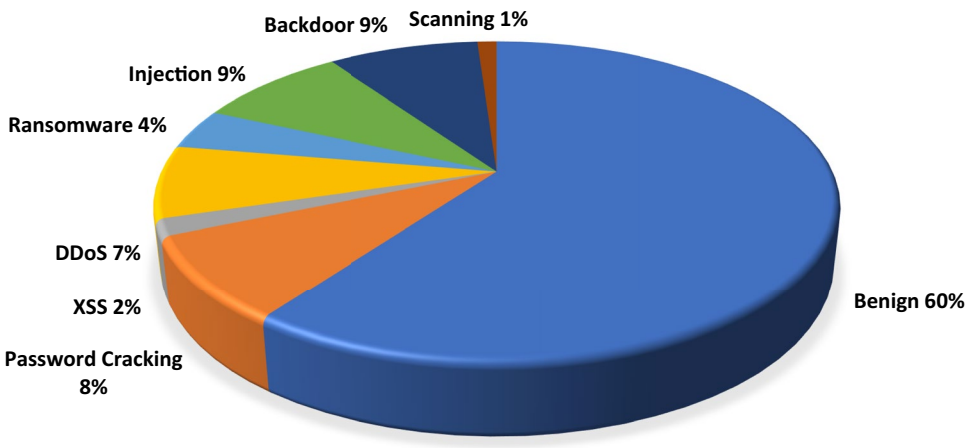

Fig. 2 Attack percentages from the total number of records

that comes the next step: the dataset was preprocessed before feeding into the algorithms. A mix of downsampling of the majority types and oversampling for the minority types were executed. One-Hot Encoder was used to encode categorical features [69], and SKlearn Robust Scaler was used to scale the features and make them robust to outliers [70]. A split of 90/10 was used for training and testing, respectively. Then the DHL model is applied to the training and testing datasets to extract features by its DL algorithm and perform classification by its ML algorithm. Figure 3 shows a summary of the utilized framework in our paper.

Two different models were proposed and applied to our dataset. The first model consists of an Attention-based Long Short Term Memory Fully Convolutional Neural Network with Extreme Gradient Boosting (ALSTM-FCN with XGBoost). The second model is an Attention-based Long Short Term Memory Fully Convolutional Neural Network with Adaptive Boost (ALSTM-FCN with AdaBoost). Both consist of two 1D convolutional hidden layers that operate over a 1D sequence with 32 kernel collections (filters) each of size 2. These kernels were used to store values learned during the training process. It is worth noting that each hidden convolutional layer was accompanied by batch normalization to normalize its input by applying a transformation that maintains the mean output close to 0 and the output standard deviation close to 1. In addition, these hidden convolutional layers were used for feature extraction.

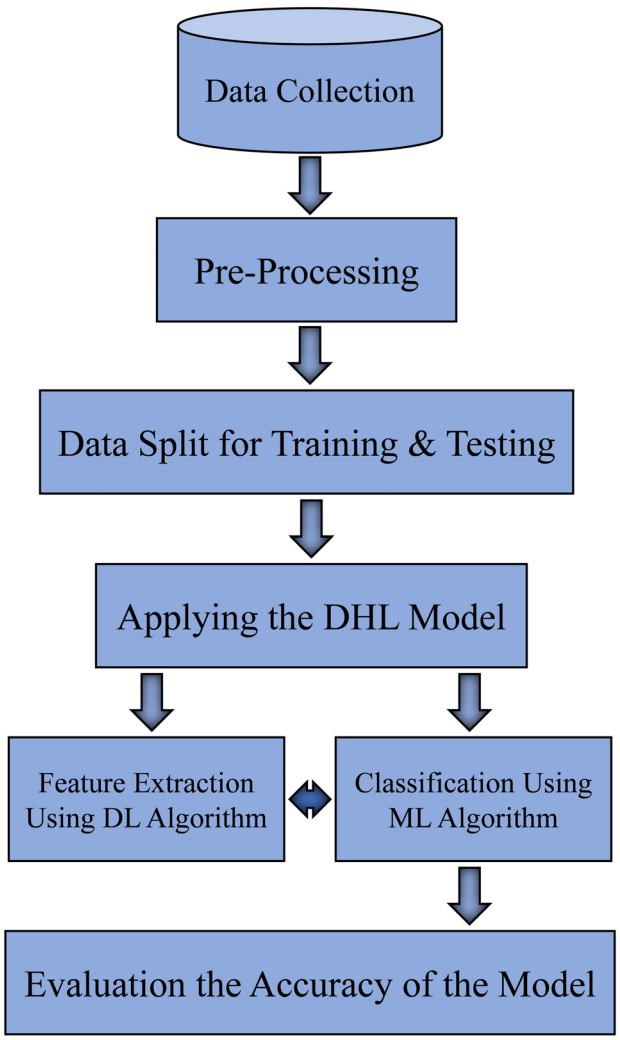

Fig. 3 NIDS framework using the DHL approach

The Rectified Linear Activation (ReLU) was used to allow the model to learn more complex functions to enhance the training process results. A GlobalAveragePooling1D layer follows the ReLU to retain much information about the "less important" outputs. Dropout layers were introduced to reduce the chances of overfitting. The oriented dropout layer had a probability value of 0.2 and 0.3 at which results of the layer are dropped out. The optimal number of LSTM cells was found to be 100. The default weight initializer that was used in both models was the GlorotUniform or Xavier Uniform. An attention mechanism was used to help the algorithm's learning process. Both models have used the Adam Optimization Algorithm with a steady learning rate of 0.03. The dropout layers are followed by a concatenation layer, a single vector-valued column from multiple columns, which is an arrangement of all the previous outputs along a specified dimension. The newly generated results are fed into the ML algorithm (XGBoost or AdaBoost in our case) that uses a RandomizedSearchCV to allow the models to select the combinations randomly (tuning the parameters) to increase the model generalizability. Figures 4 and 5 illustrate our proposed ALSTM-FCN with XGBoost and ALSTM-FCN with AdaBoost, respectively.

5 Results and discussion

Performance metrics values were calculated using values from the confusion matrix obtained for each device using the two models. These values include accuracy, precision, recall, and F1 score. Table 2 below summarizes precision, recall, and F1 scores for all the seven IIoT devices using the two proposed models.

In general, the ALSTM-FCN with AdaBoost model performed slightly better than the ALSTM-FCN with XGBoost showing slightly higher performance values for precision, recall, and F1 scores for all devices except the Weather sensor device. The two models performed in an excellent way when detecting attacks on The Garage

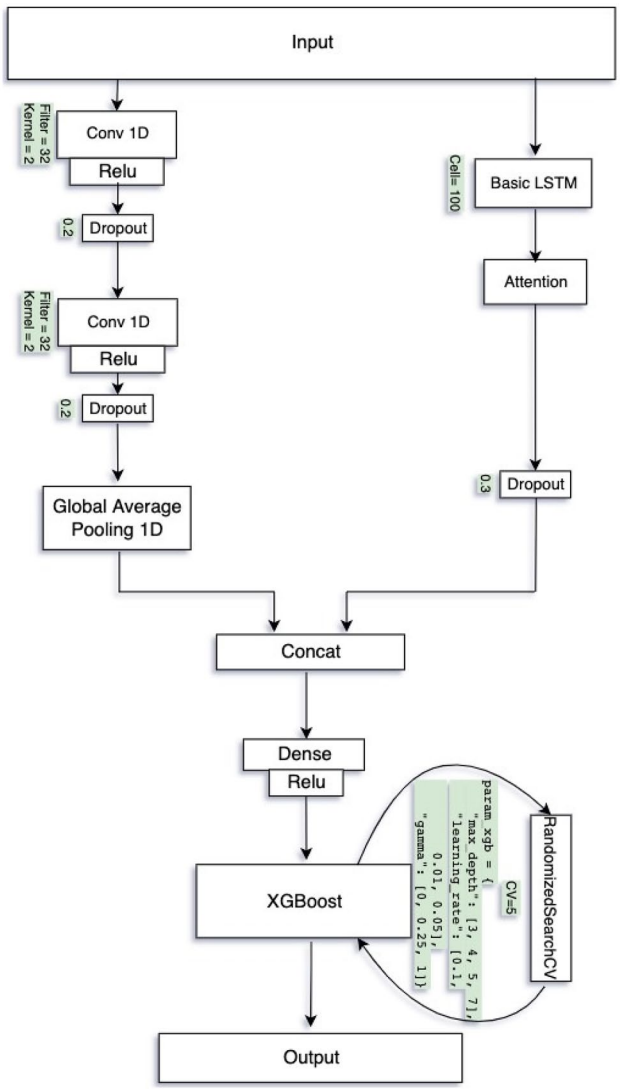

Fig. 4 ALSTM-FCN with XGBoost model

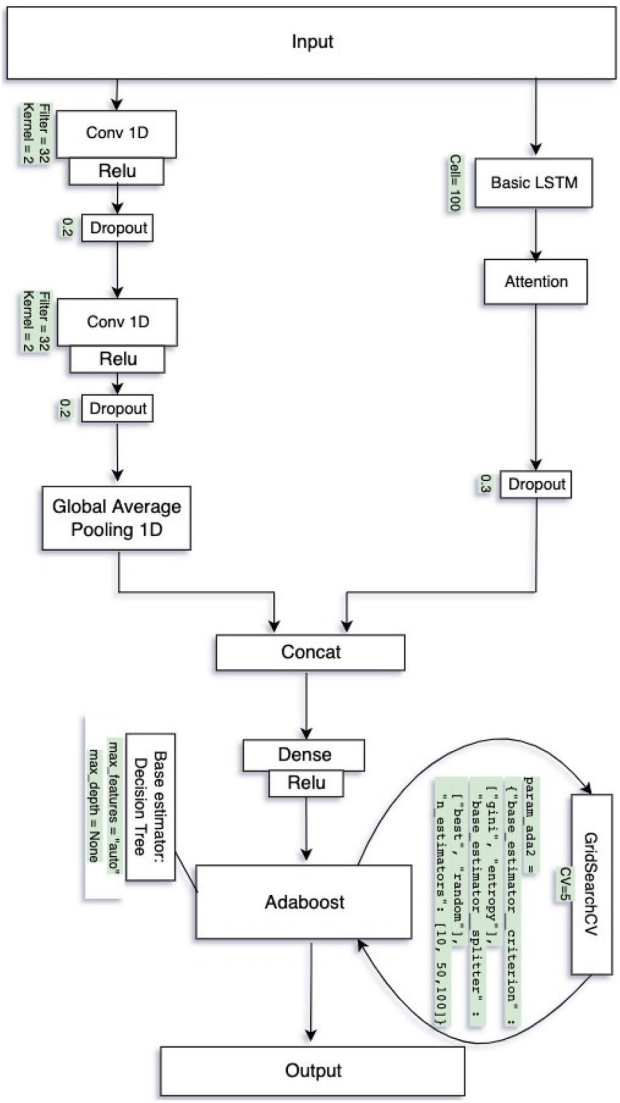

Fig. 5 ALSTM-FCN with AdaBoost model

Door sensor showing a perfect score of 100% for precision, recall, and F1 score values. At the same time, both models performed poorly when detecting attacks on the Modbus communication device.

Precision values describe how many detected instances were True Positive out of the Total Predicted Positive. Precision is used when a high risk is associated with a False Positive.

$$\text{Precision} = {}^{TP}/_{TP+FP} \tag{1}$$

For instance, in DDoS attack detection, a False Positive means that normal traffic that is Benign (True Negative) has been identified as a DDoS attack (True Positive). Therefore, the manufacturing enterprise might lose important traffic data if the precision is not high for the DDoS detection model. Our proposed models show high precision values above 92% for all devices except the Modbus device and average precision values for all instruments of 94.54% and 95.97% for both models, respectively.

Recall values describe how many detected instances were True Positive out of the Total Actual Positive. The recall is used when a high risk is associated with a False Negative.

$$\text{Recall} = {}^{TP}/_{TP+FP} \tag{2}$$

For instance, in ransomware attack detection, if a ransomware attack (True Positive) was detected as Benign (False

Table 2 Summary of precision, recall, and F1 score values

IIoT devices	Model	ALSTM-FCN with XGBoost			ALSTM-FCN with AdaBoost		
	Metric	Precision	Recall	F1	Precision	Recall	F1
Fridge sensor		94.22%	93.93%	93.94%	97.48%	97.47%	97.47%
Garage Door sensor		100%	100%	100%	100%	100%	100%
GPS tracker		99.85%	99.85%	99.85%	99.9%	99.9%	99.9%
Motion Light sensor		92.33%	92.4%	92.37%	96.77%	92.95%	93.08%
Modbus device		76.49%	75.63%	75.8%	87.67%	85.69%	86.08%
Thermostat sensor		98.98%	98.94%	98.95%	99.2%	99.2%	99.2%
Weather sensor		95.5%	95.38%	95.4%	95.22%	95.11%	95.15%
Average		94.54%	93.81%	93.86%	95.97%	95.68%	95.74%

Negative). Then, the financial consequence can be dreadful for the manufacturing enterprise. Our proposed models show high recall values above 92% for all devices except for the Modbus device and average recall values for all instruments of 93.81% and 95.68% for both models, respectively.

F1 score is a function used to balance between precision and recall values.

$$F1 = 2 \times \left({}^{\text{Precision} \times \text{Recall}} / {}_{\text{Precision+Recall}}\right) \tag{3}$$

The F1 score is used when there is a high risk associated with False Negatives and False Positives simultaneously. This makes the F1 score a suitable metric to use in NIDS since, usually, a normal traffic stream carries out information that, if it was lost (or labeled as an anomaly), might cause delays or incur costs, while at the same time, an abnormal traffic stream can cause damage if it went undetected. Our proposed models show high F1 score values above 92% for all devices except for the Modbus device and average F1 score values for all instruments of 93.86% and 95.74% for both models, respectively.

Usually, an F1 score is used when there is an uneven class distribution between False Positives and False Negatives. However, if the risk of False Positives and False Negatives are very different, then it is recommended to look at both precision and recall values.

Accuracy works best if False Positives and False Negatives have similar risk weights. Then, it is simply a ratio of the correctly detected instances (True Positive + True Negative) to the total number of cases.

$$\text{Accuracy} = {}^{TP+TN} / {}_{TP+FN+TN+FP} \tag{4}$$

Accuracy is best used as a performance measure when the number of False Positives and False Negatives are almost the same. Figure 6 displays accuracies for all devices for the two proposed models.

Figure 6 demonstrates that ALSTM-FCN with AdaBoost model had slightly higher attack detection accuracies for all devices except for the Motion Light sensor and Weather sensor than ALSTM-FCN with XGBoost model. However, the differences were less than 3% in all instruments except for the Modbus device. Figure 7 shows the accuracy values for each attack type for all seven devices using the two proposed models.

Figure 7 can be summarized in Table 3 below shows which of the two proposed models can better detect a specific type of attack (DDoS, ransomware, backdoor, injection, XSS, password, and scanning) on a particular device.

From Table 3, we can conclude that

- ALSTM-FCN with AdaBoost had similar or better accuracy detecting DDoS attacks in all devices except for the Weather sensor.
- ALSTM-FCN with AdaBoost had similar or better accuracy detecting backdoor attacks in all devices except for the GPS tracker.
- ALSTM-FCN with AdaBoost had similar or better accuracy in detecting injection attacks in all devices except for the GPS tracker.
- ALSTM-FCN with AdaBoost had similar or better accuracy in detecting XSS attacks in all devices except for the Weather sensor.
- ALSTM-FCN with AdaBoost had similar or better accuracy in detecting password cracking attacks in all devices except for the GPS tracker.
- ALSTM-FCN with XGBoost had similar or better accuracy in detecting scanning attacks in all devices except the Modbus communication device.
- ALSTM-FCN with XGBoost had similar or better accuracy in detecting ransomware attacks in all devices except the GPS tracker and the Fridge sensor, where ALSTM-FCN with AdaBoost performed better.

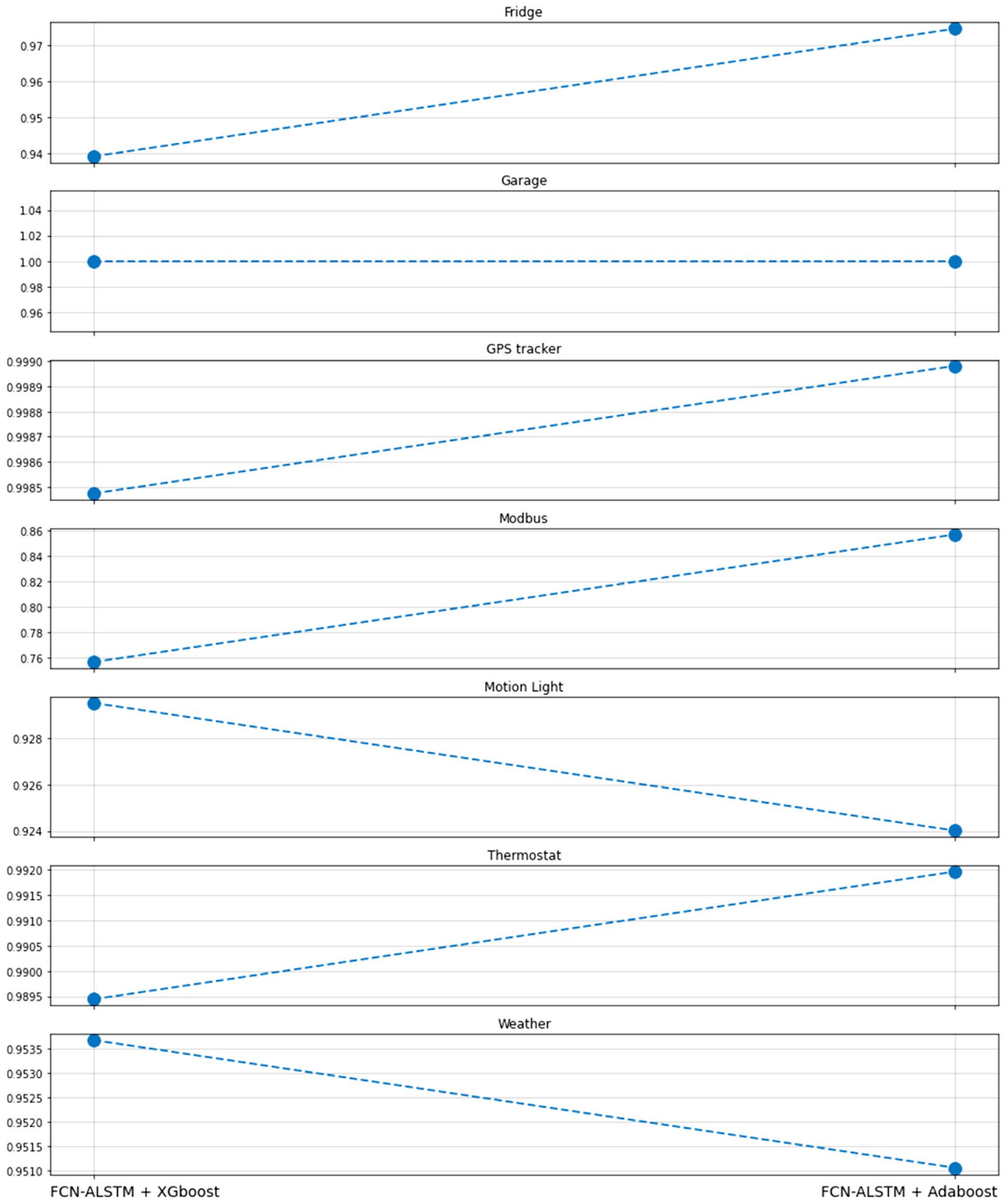

Fig. 6 Accuracies for all attacks on all devices for both ALSTM-FCN with AdaBoost and ALSTM-FCN with XGBoost models

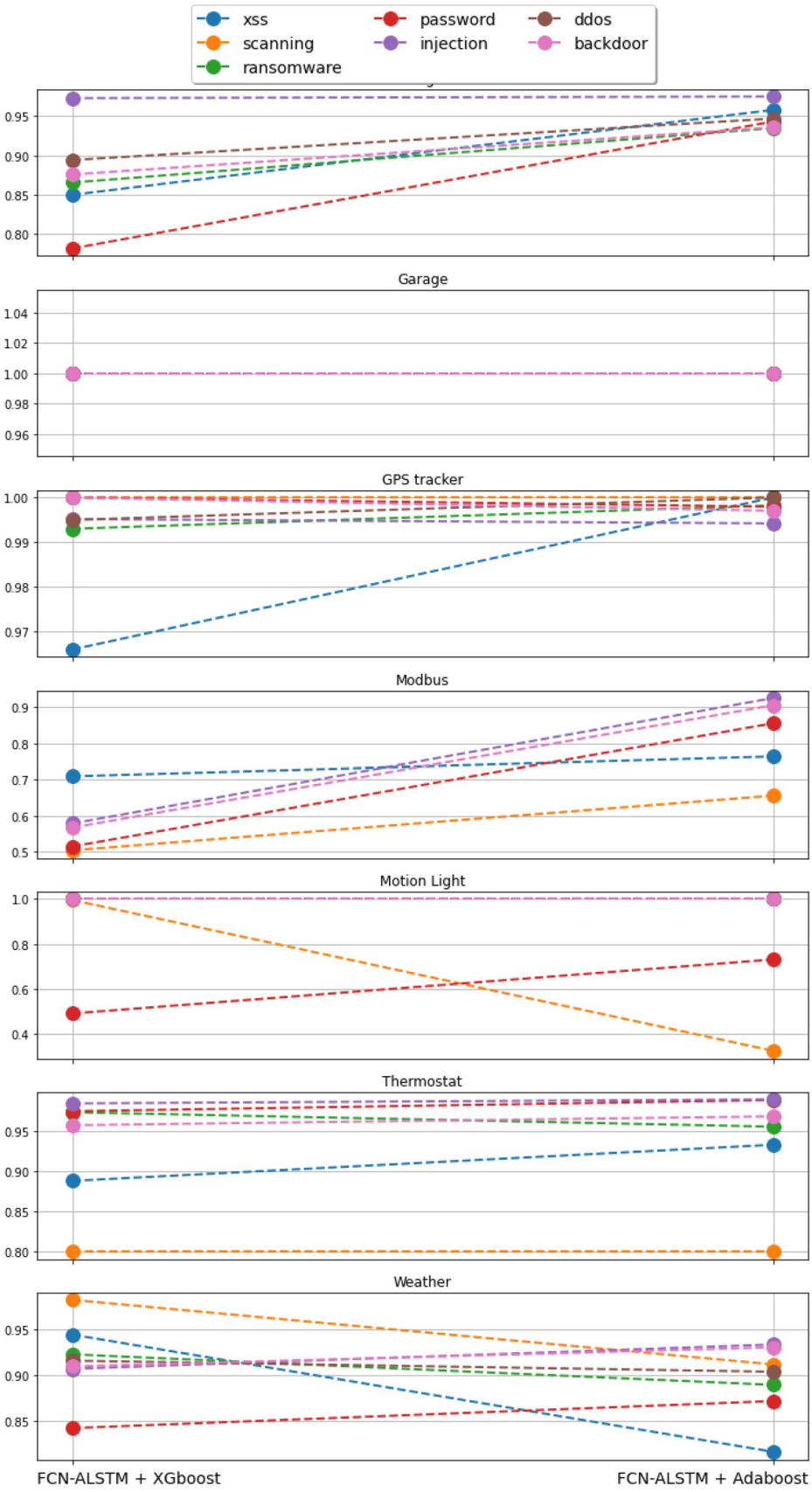

Fig. 7 Accuracy values for each attack type for all the seven devices for both ALSTM-FCN with AdaBoost and ALSTM-FCN with XGBoost models

Table 3 Accuracy comparison between ALSTM-FCN with AdaBoost and ALSTM-FCN with XGBoost models in detecting attacks on all the seven devices

Devices	DDoS	Ransomware	Backdoor	Injection	XSS	Password	Scanning
Fridge sensor	A	A	A	A	A	A	NA
Garage Door sensor	B	B	B	B	B	B	B
GPS tracker	A	A	X	X	A	X	B
Modbus device	NA	NA	A	A	A	A	A
Motion Light sensor	B	B	B	B	B	A	X
Thermostat sensor	NA	X	A	A	A	A	B
Weather sensor	X	X	A	A	X	A	X
X: ALSTM-FCN with XGBoost performed better in detecting a specific attack on a specific sensor				A: ALSTM-FCN with AdaBoost performed better in detecting a specific attack on a specific sensor			
B: Both models performed equally				NA: The attack was not carried out			

6 Conclusion

By leveraging deep hybrid learning, this paper proposed using NIDS based on ALSTM-FCN with XGBoost and ALSTM-FCN with AdaBoost. These proposed models managed to detect cybersecurity threats in seven different IIoT devices with high-performance measures. The results showed that the type of cybersecurity threat and the device it attacks could determine which proposed model is better for detecting specific threats in certain devices. Future work can focus on comparing the performance measures of other DL and ML models to our two proposed DHL models. In the future, we intend to apply the same models to different datasets to validate their efficiency.

Further attention can be given to studying the impact of device features (traffic stream variables used as input) on the performance measures of the proposed model. Future work can also focus on automatically determining the optimal hyperparameters of a deep learning-based NIDS using automated machine learning [71]. Future work can also explore the possibility of replacing XGBoost and AdaBoost with Random Forest (RF) or Support Vector Machine (SVM). Also, further deployment of the two proposed models can be done on real-time datasets.

Author contribution All authors contributed to this paper's conception and design. Material preparation, data collection, and analysis were performed by Mohammad Shahin, Hamed Bouzary, and Ali Hosseinzadeh. In addition, Mohammad Shahin wrote the first draft of the manuscript, and Rasoul Rashidifar commented on previous versions. Finally, all authors read and approved the final manuscript.

Funding The reported research work received partial financial support from Office of Naval Research MEEP Program (Award Number: N00014-19–1-2728) as well as from the Lutcher Brown Distinguished Chair Professorship fund of the University of Texas at San Antonio.

Data availability All the data have been presented in the manuscript.

Declarations

Ethics approval The paper follows the guidelines of the Committee on Publication Ethics (COPE).

Consent to participate The authors declare that they all consent to participate this research.

Consent for publication The authors declare that they all consent to publish the manuscript.

Competing interests The authors declare no competing interests.

References

1. Ralph BJ, Sorger M, Hartl K, Schwarz-Gsaxner A, Messner F, Stockinger M (2022) Transformation of a rolling mill aggregate to a cyber physical production system: from sensor retrofitting to machine learning. J Intell Manuf 33(2):493–518. https://doi.org/10.1007/s10845-021-01856-2
2. Shahin M, Chen FF, Bouzary H, Krishnaiyer K (2020) Integration of Lean practices and Industry 4.0 technologies: smart manufacturing for next-generation enterprises. Int J Adv Manuf Technol 107(5):2927–2936. https://doi.org/10.1007/s00170-020-05124-0
3. Dafflon B, Moalla N, Ouzrout Y (2021) The challenges, approaches, and used techniques of CPS for manufacturing in Industry 4.0: a literature review. Int J Adv Manuf Technol 113(7):2395–2412. https://doi.org/10.1007/s00170-020-06572-4
4. Shafae MS, Wells LJ, Purdy GT (2019) Defending against product-oriented cyber-physical attacks on machining systems. Int J Adv Manuf Technol 105(9):3829–3850. https://doi.org/10.1007/s00170-019-03805-z
5. Yuan C, Li G, Kamarthi S, Jin X, Moghaddam M (2022) Trends in intelligent manufacturing research: a keyword co-occurrence network based review. J Intell Manuf 33(2):425–439. https://doi.org/10.1007/s10845-021-01885-x
6. Oztemel E, Gursev S (2020) Literature review of Industry 4.0 and related technologies. J Intell Manuf 31(1):127–182. https://doi.org/10.1007/s10845-018-1433-8
7. Elhabashy AE, Wells LJ, Camelio JA (2019) Cyber-physical security research efforts in manufacturing - a literature review.

Procedia Manuf 34:921–931. https://doi.org/10.1016/j.promfg.2019.06.115
8. Giannetti C, Essien A (2022) Towards scalable and reusable predictive models for cyber twins in manufacturing systems. J Intell Manuf 33(2):441–455. https://doi.org/10.1007/s10845-021-01804-0
9. Significant Cyber Incidents | Center for Strategic and International Studies. https://www.csis.org/programs/strategic-technologies-program/significant-cyber-incidents (Accessed 25 Feb 2022)
10. Elhabashy AE, Wells LJ, Camelio JA, Woodall WH (2019) A cyber-physical attack taxonomy for production systems: a quality control perspective. J Intell Manuf 30(6):2489–2504. https://doi.org/10.1007/s10845-018-1408-9
11. O'Reilly P, Rigopoulos K, Feldman L, Witte G (2021) 2020 cybersecurity and privacy annual report. Natl Inst Stand Technol. https://doi.org/10.6028/NIST.SP.800-214
12. Koroniotis N, Moustafa N, Sitnikova E, Turnbull B (2019) Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. Futur Gener Comput Syst 100:779–796. https://doi.org/10.1016/j.future.2019.05.041
13. Bhattacharya A (2021) Deep Hybrid Learning — a fusion of conventional ML with state of the art DL. Medium. https://towardsdatascience.com/deep-hybrid-learning-a-fusion-of-conventional-ml-with-state-of-the-art-dl-cb43887fe14 (Accessed 25 Feb 2022)
14. Adib QAR, Mehedi MdHK, Sakib MdS, Patwary KK, Hossain MS, Rasel AA (2021) A deep hybrid learning approach to detect bangla fake news. Int Symp Multidiscip Stud Innov Technol (ISMSIT) 442–447. https://doi.org/10.1109/ISMSIT52890.2021.9604712
15. Shinde K, Thakare A (2021) Deep hybrid learning method for classification of fetal brain abnormalities. Int Conf Artif Intell Mach Vis (AIMV) 1–6. https://doi.org/10.1109/AIMV53313.2021.9670994
16. Machine learning in cybersecurity to boost Big Data, Intelligence, and Analytics spending to $96 billion by 2021. https://www.abiresearch.com/press/machine-learning-cybersecurity-boost-big-data-inte/ (Accessed 25 Feb 2022)
17. Mahmood T, Afzal U (2013) Security Analytics: Big Data Analytics for cybersecurity: a review of trends, techniques and tools. Natl Conf Inf Assurance (NCIA) 129–134. https://doi.org/10.1109/NCIA.2013.6725337
18. Terzi DS, Terzi R, Sagiroglu S (2017) Big data analytics for network anomaly detection from netflow data. Int Conf Comput Sci Eng (UBMK) 592–597. https://doi.org/10.1109/UBMK.2017.8093473
19. Gaggero GB, Rossi M, Girdinio P, Marchese M (2019) Neural network architecture to detect system faults/cyberattacks anomalies within a photovoltaic system connected to the grid. Int Symp Adv Electr Commun Technol (ISAECT) 1–4. https://doi.org/10.1109/ISAECT47714.2019.9069683
20. Wu M, Song Z, Moon YB (2019) Detecting cyber-physical attacks in CyberManufacturing systems with machine learning methods. J Intell Manuf 30(3):1111–1123. https://doi.org/10.1007/s10845-017-1315-5
21. Wu X, Goepp V, Siadat A (2020) Concept and engineering development of cyber physical production systems: a systematic literature review. Int J Adv Manuf Technol 111(1):243–261. https://doi.org/10.1007/s00170-020-06110-2
22. Cruz Salazar LA, Ryashentseva D, Lüder A, Vogel-Heuser B (2019) Cyber-physical production systems architecture based on multi-agent's design pattern—comparison of selected approaches mapping four agent patterns. Int J Adv Manuf Technol 105(9):4005–4034. https://doi.org/10.1007/s00170-019-03800-4
23. Kulkarni A, Xu C (2021) A deep learning approach in optical inspection to detect hidden hardware Trojans and secure cybersecurity in electronics manufacturing supply chains. Front Mech Eng 7. Accessed: 25 Feb 2022. [Online]. Available: https://www.frontiersin.org/article/10.3389/fmech.2021.709924
24. Bruce PC, Shmueli G, Patel NR (2016) Data mining for business analytics: concepts, techniques, and applications in Microsoft Office Excel with XLMiner. Wiley-Blackwell
25. Shahin M, Chen FF, Bouzary H, Zarreh A (2020) Frameworks proposed to address the threat of cyber-physical attacks to lean 4.0 systems. Procedia Manuf 51:1184–1191. https://doi.org/10.1016/j.promfg.2020.10.166
26. Ahmad A, Maynard S, Park S (2014) Information security strategies: towards an organizational multi-strategy perspective. J Intell Manuf 25(2):357–370. https://doi.org/10.1007/s10845-012-0683-0
27. Dhaliwal SS, Nahid A-A, Abbas R (2018) Effective intrusion detection system using XGBoost. Information 9(7). https://doi.org/10.3390/info9070149
28. Gouveia A, Correia M (2020) Network intrusion detection with XGBoost. Recent Advances in Security, Privacy, and Trust for Internet of Things (IoT) and Cyber-Physical Systems (CPS). Chapman and Hall/CRC. 137–166. https://doi.org/10.1201/9780429270567-6
29. Attia A, Faezipour M, Abuzneid A (2020) Network intrusion detection with XGBoost and deep learning algorithms: an evaluation study. In 2020 international conference on computational science and computational intelligence (CSCI) (pp 138–143). IEEE. https://doi.org/10.1109/CSCI51800.2020.00031
30. Friedman J, Hastie T, Tibshirani R (2000) Additive logistic regression: a statistical view of boosting. Ann Stat 28:337–407. https://doi.org/10.1214/aos/1016218223
31. Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29(5):1189–1232. https://doi.org/10.1214/aos/1013203451
32. Chen T, Guestrin C (2016) XGBoost: a scalable tree boosting system. Proc ACM SIGKDD Int Conf Knowledge Discov Data Min 785–794. https://doi.org/10.1145/2939672.2939785
33. Subasi A, Kremic E (2020) Comparison of Adaboost with Multiboosting for phishing website detection. Procedia Comput Sci 168:272–278. https://doi.org/10.1016/j.procs.2020.02.251
34. Tang D, Tang L, Dai R, Chen J, Li X, Rodrigues JJPC (2020) MF-Adaboost: LDoS attack detection based on multi-features and improved Adaboost. Futur Gener Comput Syst 106:347–359. https://doi.org/10.1016/j.future.2019.12.034
35. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 55(1):119–139. https://doi.org/10.1006/jcss.1997.1504
36. Freund Y, Schapire RE (1999) A short introduction to boosting. Proc Int Joint Conf Artif Intell 1401–1406
37. Yang X, Guo C (2018) Prediction of catalytic hydro conversion of normal heptane over catalysts using multi-layer perceptron artificial neural network (ANN-MLP). Pet Sci Technol 36(22):1875–1882. https://doi.org/10.1080/10916466.2018.1517164
38. Rumelhart DE, McClelland JL (Eds) (1986) Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations. Cambridge, MA, USA: MIT Press
39. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning internal representations by error propagation. Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations, Cambridge, MA, USA: MIT Press, pp. 318–362
40. Svozil D, Kvasnicka V, Pospíchal J (1997) Introduction to multilayer feed-forward neural networks. Chemom Intell Lab Syst 39:43–62. https://doi.org/10.1016/S0169-7439(97)00061-0
41. Ciaburro G (2017) Neural networks with R. Packt Publishing. Accessed: 18 Oct 2021. [Online]. Available: https://libproxy.txstate.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=cat00022a&AN=txi.b5582708&site=eds-live&scope=site
42. Pascanu R, Stokes JW, Sanossian H, Marinescu M, Thomas A (2015) Malware classification with recurrent networks. IEEE Int

Conf Acoust Speech Signal Process (ICASSP) 1916–1920. https://doi.org/10.1109/ICASSP.2015.7178304
43. Shibahara T, Yagi T, Akiyama M, Chiba D, Yada T (2016) Efficient dynamic malware analysis based on network behavior using deep learning. IEEE Glob Commun Conf (GLOBECOM) 1–7. https://doi.org/10.1109/GLOCOM.2016.7841778
44. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
45. Bahdanau D, Cho K, Bengio Y (2015) Neural machine translation by jointly learning to align and translate. Presented at the 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings. Accessed: 21 Oct 2021. [Online]. Available: https://libproxy.txstate.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=edselc&AN=edselc.2-52.0-85083953689&site=eds-live&scope=site
46. Yang S, Tan M, Xia S, Liu F (2020) A method of intrusion detection based on Attention-LSTM neural network. Proc Int Conf Mach Learn Technol. New York, NY, USA, pp. 46–50. https://doi.org/10.1145/3409073.3409096
47. Shi Z, Mamun AA, Kan C, Tian W, Liu C (2022) An LSTM-autoencoder based online side channel monitoring approach for cyber-physical attack detection in additive manufacturing. J Intell Manuf 1–17. https://doi.org/10.1007/s10845-021-01879-9
48. Kim J, Kim J, Kim H, Shim M, Choi E (2020) CNN-based network intrusion detection against denial-of-service attacks. Electronics 9(916):916. https://doi.org/10.3390/electronics9060916
49. McLaughlin N et al (2017) Deep android malware detection. Proc ACM Conf Data Appl Secur Privacy. Scottsdale, Arizona, USA, pp. 301–308. https://doi.org/10.1145/3029806.3029823
50. Gibert D, Mateu C, Planes J, Vicens R (2019) Using convolutional neural networks for classification of malware represented as images. J Comput Virol Hack Tech 15(1):15–28. https://doi.org/10.1007/s11416-018-0323-0
51. Wang W, Zhu M, Zeng X, Ye X, Sheng Y (2017) Malware traffic classification using convolutional neural network for representation learning. In 2017 International conference on information networking (ICOIN) (pp. 712-717). IEEE. https://doi.org/10.1109/ICOIN.2017.7899588
52. Karim F, Majumdar S, Darabi H (2019) Insights into LSTM fully convolutional networks for time series classification. IEEE Access 7:67718–67725. https://doi.org/10.1109/ACCESS.2019.2916828
53. Wang Z, Yan W, Oates T (2017) Time series classification from scratch with deep neural networks: a strong baseline. Int Joint Conf Neural Netw (IJCNN) 1578–1585. https://doi.org/10.1109/IJCNN.2017.7966039
54. Moustafa N (2021) A new distributed architecture for evaluating AI-based security systems at the edge: network TON_IoT datasets. Sustain Cities Soc 72:102994. https://doi.org/10.1016/j.scs.2021.102994
55. Booij TM, Chiscop I, Meeuwissen E, Moustafa N, den Hartog FTH (2021) ToN_IoT: the role of heterogeneity and the need for standardization of features and attack types in IoT network intrusion datasets. IEEE Internet Things J 1–1. https://doi.org/10.1109/JIOT.2021.3085194
56. Alsaedi A, Moustafa N, Tari Z, Mahmood A, Anwar A (2020) TON_IoT telemetry dataset: a new generation dataset of IoT and IIoT for data-driven intrusion detection systems. IEEE Access 8:165130–165150. https://doi.org/10.1109/ACCESS.2020.3022862
57. Moustafa N, Keshky M, Debiez E, Janicke H (2020) Federated TON_IoT windows datasets for evaluating AI-based security applications. IEEE Int Conf Trust Secur Privacy Comput Commun (TrustCom) 848–855. https://doi.org/10.1109/TrustCom50675.2020.00114
58. Moustafa N, Ahmed M, Ahmed S (2020) Data analytics-enabled intrusion detection: evaluations of ToN_IoT linux datasets. IEEE Int Conf Trust Secur Privacy Comput Commun (TrustCom) 727–735. https://doi.org/10.1109/TrustCom50675.2020.00100
59. Moustafa (2019) New generations of internet of things datasets for cybersecurity applications based machine learning: TON_IoT datasets. Research Data Australia. https://researchdata.edu.au/new-generations-internet-toniot-datasets/1425941 (Accessed 11 Dec 2021)
60. Moustafa N (2019) A systemic IoT-fog-cloud architecture for big-data analytics and cyber security systems: a review of fog computing. arXiv:1906.01055 *[cs]*, Accessed: 11 Dec 2021. [Online]. Available: http://arxiv.org/abs/1906.01055
61. Ashraf J et al (2021) IoTBoT-IDS: A novel statistical learning-enabled botnet detection framework for protecting networks of smart cities. Sustain Cities Soc 72:103041. https://doi.org/10.1016/j.scs.2021.103041
62. Zargar ST, Joshi J, Tipper D (2013) A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks. IEEE Communications Surveys Tutorials 15(4):2046–2069. https://doi.org/10.1109/SURV.2013.031413.00127
63. Krupp J, Backes M, Rossow C (2016) Identifying the scan and attack infrastructures behind amplification DDoS attacks. Proc ACM SIGSAC Conf Comput Commun Secur. New York, NY, USA, pp. 1426–1437. https://doi.org/10.1145/2976749.2978293
64. Al-rimy BAS, Maarof MA, Shaid SZM (2018) Ransomware threat success factors, taxonomy, and countermeasures: a survey and research directions. Comput Secur 74:144–166. https://doi.org/10.1016/j.cose.2018.01.001
65. Al-Hawawreh M, Hartog FD, Sitnikova E (2019) Targeted ransomware: a new cyber threat to edge system of brownfield industrial internet of things. IEEE Internet Things J. https://doi.org/10.1109/JIOT.2019.2914390
66. Kolias C, Kambourakis G, Stavrou A, Gritzalis S (2016) Intrusion detection in 802.11 networks: empirical evaluation of threats and a public dataset. IEEE Commun Surv Tutor. https://doi.org/10.1109/COMST.2015.2402161
67. Zolanvari M, Teixeira MA, Gupta L, Khan KM, Jain R (2019) Machine learning-based network vulnerability analysis of industrial internet of things. IEEE Internet Things J 6(4):6822–6834. https://doi.org/10.1109/JIOT.2019.2912022
68. Chaabouni N, Mosbah M, Zemmari A, Sauvignac C, Faruki P (2019) Network intrusion detection for IoT security based on learning techniques. IEEE Commun Surv Tutor 21(3):2671–2701. https://doi.org/10.1109/COMST.2019.2896380
69. Zheng A, Casari A (2018) Feature engineering for machine learning : principles and techniques for data scientists, First edition. O'Reilly Media. Accessed: 11 Dec 2021. [Online]. Available: https://libproxy.txstate.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=cat00022a&AN=txi.b5167004&site=eds-live&scope=site
70. Witten IH, Frank E, Hall MA, Pal CJ (2017) Data mining : practical machine learning tools and techniques, Fourth edition. Morgan Kaufmann. Accessed: 11 Dec 2021. [Online]. Available: https://libproxy.txstate.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=cat00022a&AN=txi.b5158398&site=eds-live&scope=site
71. Zhou X, Feng J, Li Y (2021) Non-intrusive load decomposition based on CNN–LSTM hybrid deep learning model. Energy Rep 7:5762–5771. https://doi.org/10.1016/j.egyr.2021.09.001

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.