Contents lists available at ScienceDirect

Computer Communications

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

Interpretable intrusion detection for next generation of Internet of Things

Youcef Djenouri [a], Asma Belhadi [b], Gautam Srivastava [c,d,e], Jerry Chun-Wei Lin [f,*], Anis Yazidi [g]

[a] *NORCE, Norwegian Research Center, Oslo, Norway*
[b] *Kristiania University College, Oslo, Norway*
[c] *Brandon University, Brandon, Canada*
[d] *China Medical University, Taichung, Taiwan*
[e] *Lebanese American University, Beirut, Lebanon*
[f] *Western Norway University of Applied Sciences, Bergen, Norway*
[g] *OsloMet, Oslo, Norway*

ARTICLE INFO

Keywords:
Intrusion detection
XAI
NG-IoT
Deep learning

ABSTRACT

This paper presents a new framework for intrusion detection in the next-generation Internet of Things. MinMax normalization strategy is used to collect and preprocess data. The Marine Predator algorithm is then used to select relevant features to be used in the learning process. The selected features are then trained with an advanced and state-of-the-art recurrent neural network that includes an attention mechanism. Finally, Shapely values are calculated to determine how much each feature contributes to the final output. The dataset NSL-KDD was used for intensive simulations. The results show the advantages of the proposed system as well as its superiority over state-of-the-art methods. In fact, the proposed solution achieved a rate of more than 94% for both true negative and true position, while the rates of the existing solutions are below 90% for the challenging NSL-KDD datasets.

1. Introduction

With the advent of the next-generation Internet of Things (NG-IoT), new research problems and goals have emerged [1–3]. Artificial intelligence (AI) has the potential to address the highlighted priorities of this new technology, which requires a high degree of autonomy and adaptability. In addition, NG-IoT technology has been actively deployed in intelligent transportation to meet the new market demands while achieving traditional business objectives [4–6]. The NG-IoT paves the way for better understanding of manufacturing processes and enables effective and sustainable production [7–9]. IoT devices produce a huge amount of data that requires the use of smart data analytics, not only to process the data, but also to secure the various communications between NG-IoT nodes. Securing NG-IoT and big data systems is a challenging task and has become an active research topic in the last two years [10–12]. NG-IoT devices are often distributed over a large area and have limited storage, processing, and energy resources. These characteristics make the networks and systems that contain such devices particularly vulnerable and attractive targets for cyberattacks by hackers. These systems have a large number of communication channels, storage, devices, and intrusion risks. These characteristics increase the likelihood that intrusions will occur in different patterns at different points in the system, in addition to the limitations mentioned above. The Mirai incident, in which a large number of NG-IoT devices were compromised and used for distributed denial-of-service (DDoS) attacks that crashed numerous servers, including Etsy, Github, Netflix, Shopify, Soundcloud, Spotify, Twitter, and many others, is a typical example of a NG-IoT attack. This highlights the need for efficient and proactive intrusion detection systems that can not only detect and alert intruders, but also provide the user with a better understanding of the detection process.

1.1. Motivation

Many strategies have been taken into account when devising IoT intrusion detection systems [13–16]. While signature-based methods can identify the attack, they are only effective against known threats and cannot detect new patterns. Anomaly-based solutions, usually based on monitoring the traffic flow of individual devices and comparing it to typical patterns, can be used to detect them. The traffic patterns of individual devices are often treated separately to find anomalies. Both solutions are limited and so far are only suitable for use in NG-IoT environments where different types of attacks may be present in real time. Advanced deep learning [17–20] has been widely explored in many areas of IoT. They have achieved great success in many applications. In addition, eXplanaible AI (XAI) [21–23] will be included to achieve a better understanding of the black box deep learning models used during the detection process.

[*] Corresponding author.
E-mail address: jerrylin@ieee.org (J.C.-W. Lin).

https://doi.org/10.1016/j.comcom.2023.03.005
Received 9 November 2022; Received in revised form 20 January 2023; Accepted 7 March 2023
Available online 9 March 2023

1.2. Contributions

Motivated by the success of advanced deep learning solutions and explainable AI in addressing IoT challenges, we propose a new framework called Interpretable Recurrent Neural Network (IRNN) for intrusion detection in NG-IoT. The main contributions are listed below:

1. We provide normalization and feature selection steps based on the MinMax and marine predator algorithm to select the most relevant features to be used in the learning phase.
2. We present an effective deep learning method for extracting intrusion information from NG-IoT data. The model uses an attention mechanism to compute the relevant features with the recurrent neural network to avoid the vanishing gradient problem.
3. We develop a strategy based on shape values to calculate the contribution of each feature in the detection process. This allows the user to better understand the developed black-box deep learning model.
4. Using various criteria and an extensive intrusion detection dataset, the proposed methodology is evaluated. The results clearly show the superiority of the proposed framework over the standard methods.

The rest of the paper is organized as follows. Modern techniques for intrusion detection are discussed in Section 2. The proposed framework and its essential elements are described in Section 3. The design and results of the experiment are summarized in Section 4. Section 5 presents discussions and future directions of this paper. The paper is then concluded in Section 6.

2. Related work

To create adversarial hostile traffic records intended to evade detection by intrusion detection systems, a generative hostile network framework known as IDSGAN was proposed in [24]. Since attackers do not know the basic structure and parameters of the detection system, adversarial attack examples use black box attacks against the system. IDSGAN uses a generator to convert the initial malicious traffic records into adversarial malicious records. The real-time blackbox detection system is dynamically learned by a discriminator that also categorizes traffic instances. Moreover, for the generation of the malicious data, the constrained modification technique has been developed to preserve the original attack capabilities of the hostile traffic data. To provide justifications for important Deep Learning (DL)-based decisions for IoT-related IDS, Zakaria et al. [25] created a new framework based on XAI. To find IoT-related intrusions, they used a unique IDS for IoT networks, which they had also created using deep neural networks. To the DL-based model, they added three primary XAI techniques, e.g., RuleFit, Local Interpretable Model-Agnostic Explanations (LIME), and SHapley Additive exPlanations (SHAP). To improve the interpretation of DL-based judgments, they provide both local and global explanations. While global explanations focus on identifying the key factors that led to each decision made, local explanations focus on a single/specific DL outcome (e.g., intrusion detection). To identify security risks in IoT contexts, a brand-new deep learning-based intrusion detection system (DL-IDS) was presented [26].

Although several IDSs are described in the literature, they all have deficiencies in learning and dataset management that significantly impact the accuracy of attack detection. To achieve optimal detection, the authors coupled the Spider Monkey Optimization (SMO) algorithm and the Stacked-Deep Polynomial Network (SDPN). SMO selects the best features from the datasets, while SDPN categorizes the data as normal or anomalous. Denial of Service (DoS), User-to-Root (U2R), probing, and Remote-to-Local (R2L) attacks are among the anomalies that DL-IDS can identify. Tanzila et al. [27] presented a CNN-based strategy for anomaly-based intrusion detection systems that leverages the potential of IoT and provides capabilities to effectively probe all traffic in IoT. The proposed model has shown the ability to detect potential intrusions and unusual traffic patterns. Using deep learning-based recurrent models, Ravi et al. [28] provided a complete network attack detection and classification model. The proposed model collects hidden layer features from recurrent models and then selects the best features using kernel-based principal component analysis (KPCA) for feature selection. An ensemble meta-classifier is used to classify the data after combining the best features from recurrent models. Based on cost-sensitive deep learning and ensemble algorithms, CSE-IDS, a three-layer NIDS, was proposed [29]. Layer 1 of the proposed CSE-IDS uses a cost-sensitive Deep Neural Network to distinguish between valid and malicious network traffic. These dubious samples are then forwarded to Layer 2, where they are classified into normal, multiple majority attack classes, and a single class encompassing all minority attack classes using the eXtreme Gradient Boosting algorithm. Finally, layer 3 uses Random Forest to assign appropriate classes to the minority attacks found in layer 2. Zhang et al. [30] presented TIKI-TAKA, a comprehensive framework for evaluating the resilience of cutting-edge deep learning-based NIDS against hostile tampering. They also incorporated defense mechanisms to strengthen resistance against attacks using evasion strategies. In particular, they developed five new types of attacks to subvert three widely used neural network-based malicious traffic detectors. Alladi et al. [31]presented an AI-based intrusion detection architecture consisting of deep-learning models to identify and classify vehicle traffic networks into potential types of cyberattacks. Due to the mobility of vehicles and the real-time requirements of traffic network channels, these deep-learning models are run on edge servers rather than in a remote cloud. Thakkar et al. [32] have developed a fusion-based solution that aims to improve the effectiveness of deep learning systems for intrusion detection by presenting a new algorithm that selects features based on the standard error variance of historical observations. Features are matched based on their rank, which is derived from statistical significance fusion. Moreover, statistical significance fusion aims to produce relevant features with high distinctiveness and variance, which contributes to better learning. However, the strategy used is not powerful for NG-IoT data where the data is collected in different learning spaces.

Based on this relatively brief literature review, we conclude that intrusion detection methods are efficient in identifying outliers. However, they are still far from being deployed in NG-IoT environments where there may be different types of attacks in real time. Effective preprocessing and feature extraction are required before the learning process. Moreover, in most of the cases described, there is a lack of explanations, so the network manager has difficulty in understanding the features that contribute positively to learning and the features that contribute negatively to learning. Indeed, an efficient interpretation strategy is also needed. To overcome the problems of existing solutions, and motivated by the success of advanced deep learning and XAI, we introduce in the following section a suitable platform, called IRNN, for network attack detection and understanding. Unlike existing solutions, IRNN enables better extraction of relevant features from network traffic data, accurate learning using advanced recurrent neural network architecture, and better understanding of the learning process using XAI.

3. IRNN design

In this section, we present a novel framework for intrusion detection from the next generation Internet of Things. The proposed framework, which we call IRNN (Interpretable Recurrent Neural Network), is shown in Fig. 1. IRNN combines advanced deep learning architectures with an interpretive process to identify intrusions in the next-generation IoT environment. The process begins with preprocessing and extraction of relevant features through normalization and feature selection. The intruders are then detected using the Attention Segmental Recurrent Neural Network (ASRNN) algorithm [33]; the interpretation of the derived intrusions is finally examined using the Shapely value. Each phase is described in detail below:

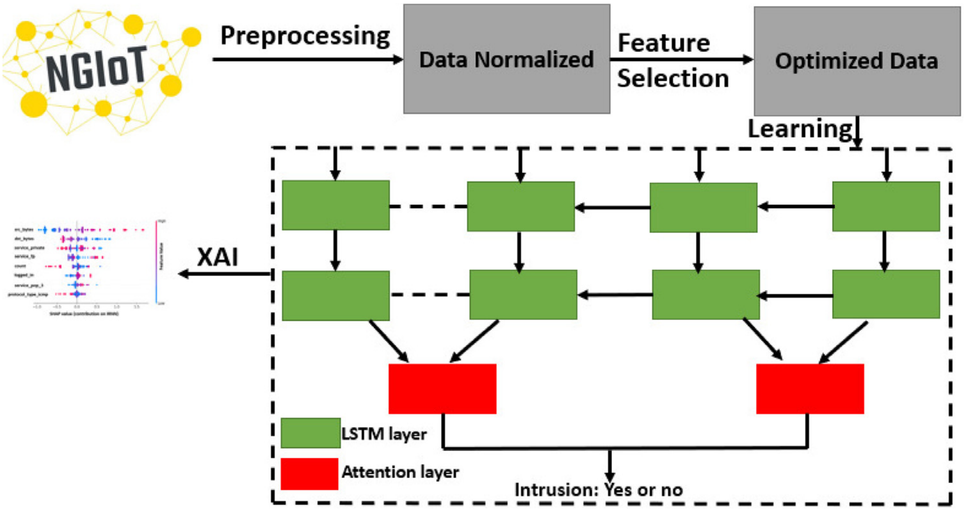

Fig. 1. IRNN Framework: Preprocessing is done first using normalization, and then the pertinent features are extracted using feature selection. The ASRNN technique is then used to detect the intrusions, and the Shapely value is then used to interpret the resulting intrusions.

3.1. Preprocessing

Data preprocessing is the process of modifying raw data that is duplicated, erroneous, irrelevant, redundant, incomplete, or incorrectly formulated. Data preprocessing is the process of removing information. The main goal was to remove data from the datasets to standardize data analysis and make it easier to find relevant data for the study. Since some of the data were already missing or unclear, it was important to modify the existing data to improve quality by omitting inaccurate information. The MinMax normalization method [34] is essential for integrating and normalizing data. Where "one" is assigned as the lowest feature value and "zero" as the highest value. The binary equivalents of each value of 0 and 1 are calculated. For each sample x, the $MinMax$ formula is determined as follows:

$$MinMax(x) = \frac{x - min(x)}{max(x) - min(x)}, \tag{1}$$

in which $min(x)$, and $max(x)$ are the minimum and the maximum values of all samples, respectively. Due to corrupted traffic data, even after complete normalization for unstructured data, the data is still suspect. The collection of these attributes from many complex systems enables the investigation of intrusion detection.

3.2. Feature selection

The feature values are automatically added to the feature selection when using the data from the preprocessing phase, which helps to increase accuracy. Feature values that are not needed, redundant, or irrelevant are disregarded and no longer help classify attacks. Therefore, to assess the accuracy of the search domain, feature selection techniques are used to select essential features. The classifier eliminates the irrelevant components and selects the top ten features based on their relevance. Combining optimization strategies with exploration algorithms strengthens the exploration capabilities. We used the Marine Predators Algorithm (MPA) [35] to extract the relevant features for prediction. It is a brand new meta-heuristic algorithm inspired by nature. Similar to other meta-heuristic algorithms, the MPA algorithm is used to solve practical optimization problems. The broad-scale foraging of marine predators and the encounters or interactions between predators and prey serve as inspiration for MPA. Here, a predator strategically controls encounter rates to increase its chances of survival in the wild. Using L'evy flight and Brownian motion, MPA performs a search using two basic random walk methods. The first type of random walk is commonly used in meta-heuristic algorithms and is probably most successful in preventing solution stagnation by performing an advantageous search in local areas [36]. The latter, on the other hand, is a well-known stochastic tool for global search. To maximize the balance between exploration and exploitation, the MPA inventors merged the search efficiency of the two random walk algorithms. Similar to a number of other population-based metaheuristic algorithms, MPA begins the search process by randomly distributing N search agents over the search area using Eq. (2):

$$\overrightarrow{X_i} = \overrightarrow{lb_i} + r \times (\overrightarrow{ub_i} - \overrightarrow{lb_i}); i \in \{1, 2, \dots, N\} \tag{2}$$

where $\overrightarrow{lb_i}$ and $\overrightarrow{ub_i}$ are two vectors that indicate the lower and higher bounds for the search to be conducted within, and r denotes a random variable between [0, 1]. Another $N \times D$ matrix made up of search agents with the best fitness values is formed during initialization along with the primary population matrix, where N and D stand for population size and problem dimensions, respectively. MPA refers to it as Elite, which is composed by the set of vectors with top fitness. Prey is a different matrix of the same dimension as Elite, and the predators adjust their places in accordance with it. The initialization creates the initial Prey in a single term from which the strongest individual (predator) creates the Elite. These two matrices play a key role in the optimization process. After initialization, the main iterative search process begins. This process is divided into three phases that simulate various predator–prey scenarios while coming up with various search tactics. These phases are based on iterations $t \in \{1, 2, 3 \dots t_{max}\}$ where t_{max} is maximum iterations. Note that MPA updates candidate solutions dimension-wise during these phases.

3.3. Learning

The traffic network is used to detect intrusions with the ASRNN algorithm [33]. LSTM (Long Short Term Memory) performs better than the currently used RNN-based systems. Namely, the LSTM model ensures correlation between different elements in a sequence where a long dependency is checked. This allows to mitigate the vanishing gradient problem of the RNN-based systems. Therefore, the LSTM model is used in the modified ASRNN model proposed in this study. Two different LSTM models are used to learn from the traffic network. While the second model uses the attention mechanism to determine the local features of each element in the context vector, the first model is based on determining the context vector on the flow time. At each timestamp, a Bi-LSTM is used to retrieve the context vector, and a second Bi-LSTM is used in a recursive manner to dynamically generate the segmental representation for each segment using an attention mechanism. Dynamic recursion is used in the computation of the segment. Then, each segmental representation is subjected to label categorization using a fully connected layer. The score computed by a fully linked layer is directly used to compute the neural feature scores. Since the sum of the neural feature scores can be greater than one, the softmax operation with the fully connected layer is not required for label classification. The semi-CRF model [37] is then trained together with the computed neural feature values, which are transferred to the model along with the traditional semi-CRF features. For the network structure, the hidden dimension in both the lower and upper Bi-LSTM networks was set to 50, resulting in a character-level representation vector with 100 dimensions. The hidden dimension of the lower Bi-LSTM network and the upper network was set to 100 for the word-level encoder, resulting in a segmental representation with 200 dimensions. The output size of the fully linked layer, which had a hidden size of 256 and a number of labels for each task, was equal to the number of labels. For optimization, we used a mini-batch stochastic gradient descent with 10 batches and a momentum of 0.9. The initial learning rate was set to 0.01 and the decay rate to 0.1. To avoid "breaking out the gradient", the gradient clipping was set to 5.

3.4. Shapely value

Shapley values are an idea from cooperative game theory. Shapley values were introduced to fairly allocate a player's contribution to the final outcome of a game. Suppose we have a cooperative game where a group of players work together to create value. If we can calculate the total payoff of the game, the Shapley values capture each player's marginal contribution to the final outcome. More formally, suppose we have a game with n players, with players $1, 2, \dots, n$ and a value function v that accepts a small proportion of players and returns the real value of the game if only those players participated. We also have S as a coalition or subset of players. Formally, then, the contribution Θ of the ith player is defined as:

$$\Theta(v)_i = \sum_{S\subset\{N-\{i\}\}} \frac{|S|!(N-|S|-1)!}{N!}(v(S\ cup\{i\}) - v(S)) \tag{3}$$

The Shapley score is a metric that can used in explainable machine learning to quantify the contributions of input features (players) to the output of an instance-level machine learning model. The goal is to break down the model prediction into its components and assign Shapley values to each instance feature given a single data point. The only requirement for a cooperative game for the interpretation proposed in this research is that the model can produce a scalar-valued output, such as the probability of assigning a class label to an instance. Since the principle of efficiency applies, determining the Shapley value in such a game leads to a complete deconstruction of the process of intrusion detection. Missing values of input features are replaced by a reference value, e.g., the mean value determined from numerous examples, and the Shapley values of the feature values are explanatory assignments to the input features. We suppose that the Shapley values are approximated over linear time. Our goal is to model the explainability of neurons using a game in which neurons are actors and neural attributions are rewards. These games will be solved and the attributions will be computed with respect to the neurons and filters. The output of the neural network obtained by hiding certain neurons is what is known in practice as "payoffs". Individual neurons can be evaluated using the Shapley values obtained in these games. The proposed strategy is exclusive to deep learning and shares goals and designs with the games mentioned in universal explainability.

4. Performance evaluation

Extensive simulations were performed to validate the performance of the proposed IRNN system. The evaluation uses the True Positive Rate (TPR) and True Negative Rate (TNR), which are commonly used to evaluate intrusion detection systems. They are specified as follows:

$$TPR = \frac{TP}{TP+FP}, \tag{4}$$

and,

$$TNR = \frac{TN}{TN+FP}, \tag{5}$$

where the TP, FP, and TN variables stand for the number of true positives, false positives, and true negatives, respectively.

In this experiment, the performance of the IRNN algorithm with the following baseline solutions: TIKI-TAKA [30], CSE-IDS [29], and DL-IDS [26]. For the training data, we use a batch size of 512 samples by default and reduce the batch size if the model does not fit in memory. We find that we achieve the same speed with batch sizes of 64, 128, 256, or 512. The stack size of the training phase is used to determine the number of training epochs and the learning rate. The final layer of our framework and the baseline models is treated with a dropout rate of 0.7.

4.1. Data description

The NSL-KDD dataset [38] represents an improved version of the KDD'99 dataset that DARPA had previously published, adding a wide range of actual attacks on a transportation network. Like the KDD'99 dataset, the NSL-KDD dataset contains 41 network-related features collected from TCP/IP dumps, as well as examples of 23 different attacks in the training set and 17 new attacks in the test set. The NSL-KDD dataset improves upon the KDD'99 dataset in several ways, including deleting duplicate data streams and using a proportional inclusion approach to reduce class imbalances caused by unusual attack types. These improvements are expected to improve consistency and fairness when comparing different NIDS. In this paper, we train and test our approach using the datasets "KDDTrain+" and "DDTest+". Both datasets are annotated where ground truth is available. This facilitates both the training process for building supervised learners and the testing process for evaluating the developed system before deployment. Although the NSL-KDD dataset has certain limitations in capturing examples of more recent attack methods, it is one of the few publicly available datasets that can be used to evaluate the performance of a NIDS when the training and testing distributions differ.

4.2. Inference runtime and accuracy

Extensive testing was performed to evaluate the inference time of the IRNN. The experimental data was divided into buckets, each containing a certain amount of network data. The first bucket contains 20% of the test data, the second 50%, the third 80%, and the last 100% (all test data). The models were trained first and then the inference was performed with the test data to calculate the inference time. Fig. 2 shows how the inference times for IRNN and baseline approaches differ. It can be seen that as the percentage of data features and test data increases, the inference time for the four methods also increases. The results also show the superiority of the proposed strategy, with a time difference of more than 500 ms for the data from NSL-KDD. These results are justified by combining a state-of-the-art feature selection technique with a powerful recurrent neural network architecture and an effective attention mechanism. While IRNN explores a novel MPA-based method, the baseline methods use traditional feature selection techniques such as PCA. Extensive experiments have been conducted to evaluate the quality of traffic network performance in intrusion detection. The TPR and TNR values are obtained for each iteration of the experiments. The results are highlighted in the Tables 1, 2. Compared to the baseline approaches, the numerical results show that IRNN can identify the correct interventions. The feature selection used in this study and a deep learning model that analyzes and learns from the numerous correlations between the input data to determine the interventions were both carefully employed to achieve these results.

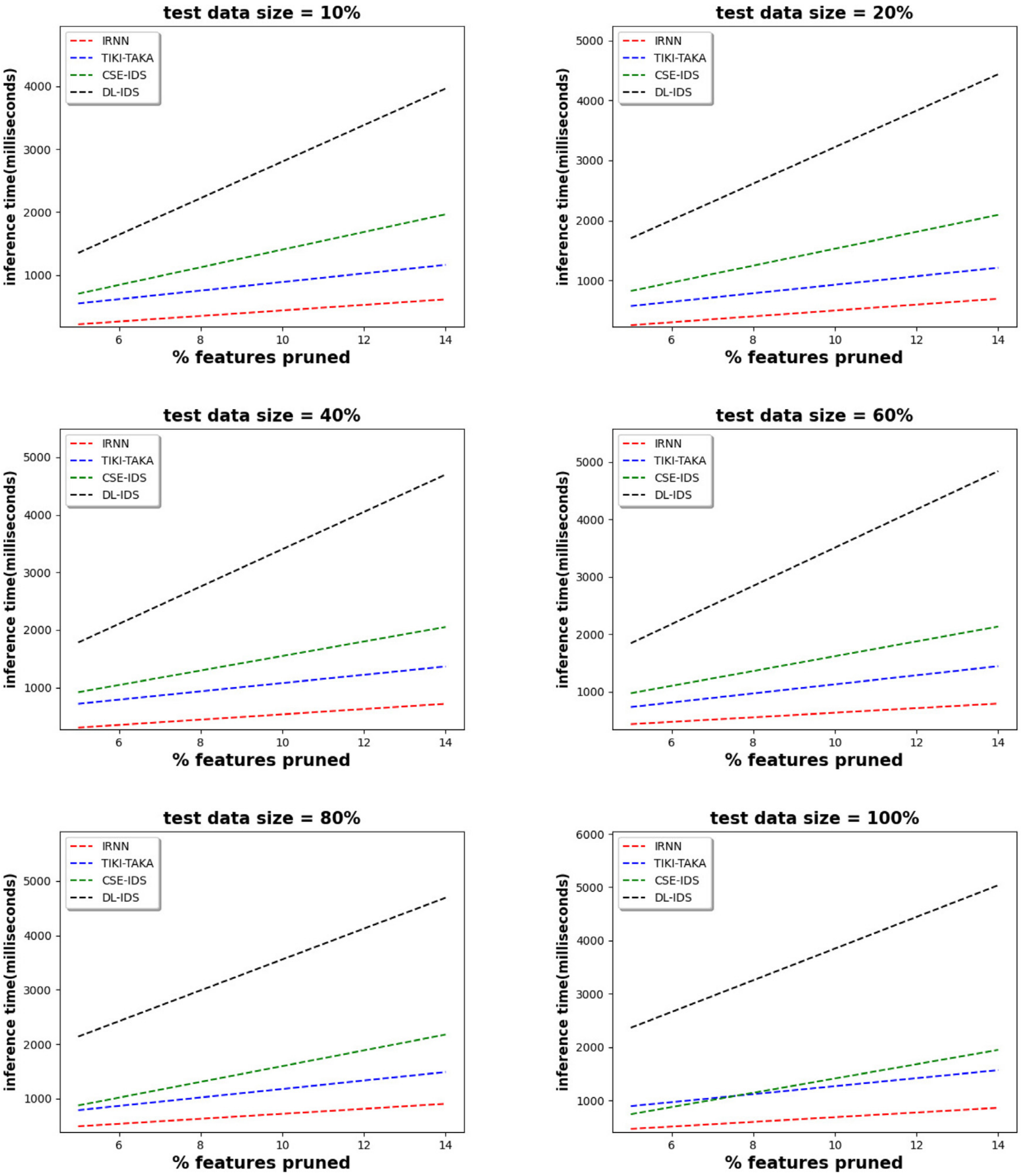

Fig. 2. Inference time in milliseconds of IRNN vs. state-of-the art NG-IoT intrusion detection algorithms.

Table 1
TNR of IRNN vs. state-of-the art NG-IoT intrusion detection algorithms.

Number of Epochs	IRNN	TIKI-TAKA	CSE-IDS	DL-IDS
100	25	20	8	7
200	41	27	15	13
500	78	63	38	34
800	95	89	61	54

Table 2
TPR of IRNN vs. state-of-the art NG-IoT intrusion detection algorithms.

Number of Epochs	IRNN	TIKI-TAKA	CSE-IDS	DL-IDS
100	38	18	4	3
200	66	41	29	26
500	87	53	41	38
800	94	75	58	56

4.3. Interpretation

To understand the behavior of the IRNN model, an experiment was conducted to visualize the output of the Shapely value. Fig. 3 shows the SHAP value for the eight most important variables in the learning process. It is worth noting that when the SHAP value is less than 0, the variable has a negative impact on learning, while when the SHAP value is greater than 0, the variable has a positive impact on learning. From Fig. 3, we can conclude that there are many variables that contribute positively to learning, but that more robust feature selection methods are needed to weed out the variables that contribute negatively to learning. This will improve intrusion detection performance. Even though MPA performs very well in selecting the most relevant features, more efforts need to be made. For example, combining traditional methods such as PCA with MPA could be a good direction for future work.

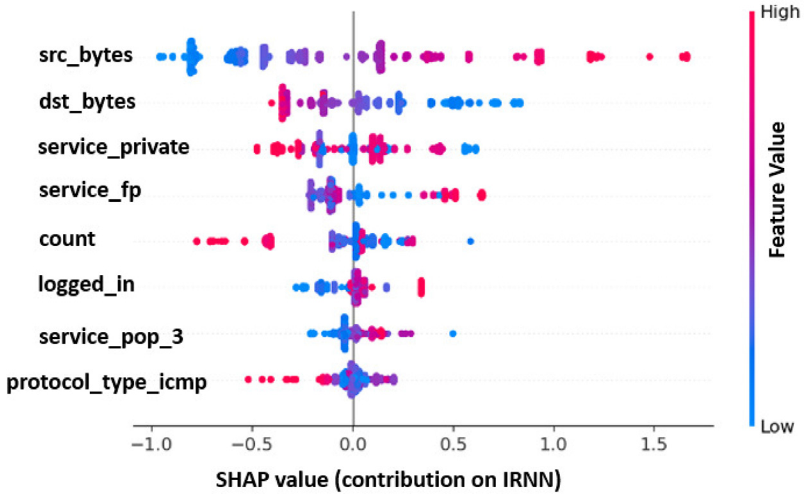

Fig. 3. Contribution of the most relevant features for the IRNN model output.

5. Discussions and future directions

The first challenge is to build a comprehensive dictionary of attack signatures in complex NG-IoT systems. This makes detecting a zero-day attack difficult. Since IRNN lacks attack data, modern data augmentation techniques such as generative adversarial network and variable auto-encoder can be useful to generate relevant training data with different attack types. Although not all attacks generated by data augmentation models are actual attacks, the comprehensive attack dictionary created appears to be an effective strategy for defending against zero-day attacks. Heterogeneous data in IoT, especially in NG-IoT, is considered more vulnerable to a variety of threats than their wired counterparts. This is because of the complex network topology and high connectivity between traffic variables. Adversaries face a large attack surface that includes multiple entry points. Moreover, the attackers can change their behavior, rendering the initial learning of the IRNN ineffective. The second challenge is to develop an IRNN model that is capable of detecting new attacks in a traffic network using a lifetime learning mechanism. Federated learning (FL) has recently attracted the interest of academia and industry as an alternative to the traditional centralized ML approaches. FL has a significant privacy advantage, as training nodes can build a global model without transmitting their data. The learning process occurs over a set number of training rounds, in which each node continuously monitors the parameters of a modeling framework by training with its local data. In each training round, these parameters are then accumulated by a central entity to compute an updated copy of the global model, which is in turn communicated to the nodes. The third challenge of this work is to take advantage of FL in the context of IRNN to create distributed models that are shared among different entities in the network without these entities having access to their own data.

6. Conclusion

This study presents a revolutionary paradigm for the next generation Internet of Things dedicated to intrusion detection. The MinMax normalization approach is used to collect and preprocess the dataset. The Marine Predator algorithm is then used to select features. The selected feature is then trained with an advanced recurrent neural network that includes an attention mechanism. The introduction of the Shapely value is the final step to determine how each feature affects the final output. The dataset NSL-KDD was subjected to extensive simulation. The results illustrate the benefits of the framework provided and how it outperforms state-of-the-art techniques. In the future, we plan to explore other Deep Learning architectures for intrusion detection, such as those based on convolutional neural networks [39]. We also plan to consider group detection [40] as a future strategy in next-generation IoT. Exploring genetic algorithm, pattern mining, and decomposition in learning [41–43] is also on our agenda.

CRediT authorship contribution statement

Youcef Djenouri: Conceptualization, Writing – original draft. **Asma Belhadi:** Methodology. **Gautam Srivastava:** Writing – review & editing. **Jerry Chun-Wei Lin:** Formal analysis, Writing – review & editing. **Anis Yazidi:** Validation.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgment

This work is supported by the project ULEARN “Unsupervised Lifelong Learning” and co-funded under the grant number 316080 of the Research Council of Norway.

References

[1] M. Maiti, U. Ghosh, Next generation Internet of Things in fintech ecosystem, IEEE Internet Things J. (2021).
[2] A. Asheralieva, D. Niyato, Optimizing age of information and security of the next-generation internet of everything systems, IEEE Internet Things J. (2022).
[3] A. Rejeb, K. Rejeb, S. Simske, H. Treiblmaier, S. Zailani, The big picture on the Internet of Things and the smart city: A review of what we know and what we need to know, Internet of Things 19 (2022) 100565.
[4] A.B. Adam, M.S.A. Muthanna, A. Muthanna, T.N. Nguyen, A.A. Abd El-Latif, Toward smart traffic management with 3D placement optimization in UAV-assisted NOMA IIoT networks, IEEE Trans. Intell. Transp. Syst. (2022).
[5] N. Zhang, T. Han, M. Dianati, N. Lu, S. Wang, Guest editorial special issue on space-air-ground integrated networks for intelligent transportation systems, IEEE Trans. Intell. Transp. Syst. 23 (3) (2022) 2701–2704.
[6] Y. Yao, H. Zhang, L. Lin, G. Lin, R. Shibasaki, X. Song, K. Yu, Internet of Things positioning technology based intelligent delivery system, IEEE Trans. Intell. Transp. Syst. (2022).
[7] V. Delpla, J.-P. Kenné, L.A. Hof, Circular manufacturing 4.0: Towards Internet of Things embedded closed-loop supply chains, Int. J. Adv. Manuf. Technol. 118 (9) (2022) 3241–3264.

[8] C. Liu, Z. Su, X. Xu, Y. Lu, Service-oriented industrial Internet of Things gateway for cloud manufacturing, Robot. Comput.-Integr. Manuf. 73 (2022) 102217.

[9] J. Sousa, J.P. Mendonça, J. Machado, A generic interface and a framework designed for industrial metrology integration for the Internet of Things, Comput. Ind. 138 (2022) 103632.

[10] A. El Kamel, H. Eltaief, H. Youssef, On-the-fly (D) DoS attack mitigation in SDN using deep neural network-based rate limiting, Comput. Commun. 182 (2022) 153–169.

[11] W. Xue, Y. Shen, C. Luo, W. Xu, W. Hu, A. Seneviratne, A differential privacy-based classification system for edge computing in IoT, Comput. Commun. 182 (2022) 117–128.

[12] M. Rana, A. Shafiq, I. Altaf, M. Alazab, K. Mahmood, S.A. Chaudhry, Y.B. Zikria, A secure and lightweight authentication scheme for next generation IoT infrastructure, Comput. Commun. 165 (2021) 85–96.

[13] Y. Djenouri, A. Belhadi, G. Srivastava, U. Ghosh, P. Chatterjee, J.C.-W. Lin, Fast and accurate deep learning framework for secure fault diagnosis in the industrial Internet of Things, IEEE Internet Things J. (2021).

[14] F. Hussain, R. Hussain, S.A. Hassan, E. Hossain, Machine learning in IoT security: Current solutions and future challenges, IEEE Commun. Surv. Tutor. 22 (3) (2020) 1686–1721.

[15] J.C.-W. Lin, G. Srivastava, Y. Zhang, Y. Djenouri, M. Aloqaily, Privacy-preserving multiobjective sanitization model in 6G IoT environments, IEEE Internet Things J. 8 (7) (2020) 5340–5349.

[16] P. Sharma, S. Jain, S. Gupta, V. Chamola, Role of machine learning and deep learning in securing 5G-driven industrial IoT applications, Ad Hoc Netw. 123 (2021) 102685.

[17] S. Singh, R. Sulthana, T. Shewale, V. Chamola, A. Benslimane, B. Sikdar, Machine-learning-assisted security and privacy provisioning for edge computing: A survey, IEEE Internet Things J. 9 (1) (2021) 236–260.

[18] S. Hui, H. Wang, Z. Wang, X. Yang, Z. Liu, D. Jin, Y. Li, Knowledge enhanced GAN for IoT traffic generation, in: Proceedings of the ACM Web Conference 2022, 2022, pp. 3336–3346.

[19] R. She, P. Fan, From MIM-based GAN to anomaly detection: Event probability influence on generative adversarial networks, IEEE Internet Things J. (2022).

[20] X. Cao, G. Sun, H. Yu, M. Guizani, PerFED-GAN: Personalized federated learning via generative adversarial networks, IEEE Internet Things J. (2022).

[21] S.K. Jagatheesaperumal, Q.-V. Pham, R. Ruby, Z. Yang, C. Xu, Z. Zhang, Explainable AI over the Internet of Things (IoT): Overview, state-of-the-art and future directions, IEEE Open J. Commun. Soc. (2022).

[22] H. Elayan, M. Aloqaily, F. Karray, M. Guizani, Internet of behavior (IoB) and explainable ai systems for influencing iot behavior, IEEE Netw. (2022).

[23] L.M. Alkwai, An explainable artificial-intelligence-based CNN model for knowledge extraction from the social Internet of Things: Proposing a new model, IEEE Syst. Man Cybern. Mag. 8 (4) (2022) 48–51.

[24] Z. Lin, Y. Shi, Z. Xue, Idsgan: Generative adversarial networks for attack generation against intrusion detection, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2022, pp. 79–91.

[25] Z. Abou El Houda, B. Brik, L. Khoukhi, ''Why should I trust your IDS?'': An explainable deep learning framework for intrusion detection systems in Internet of Things networks, IEEE Open J. Commun. Soc. 3 (2022) 1164–1176.

[26] Y. Otoum, D. Liu, A. Nayak, DL-IDS: A deep learning–based intrusion detection framework for securing IoT, Trans. Emerg. Telecommun. Technol. 33 (3) (2022) e3803.

[27] T. Saba, A. Rehman, T. Sadad, H. Kolivand, S.A. Bahaj, Anomaly-based intrusion detection system for IoT networks through deep learning model, Comput. Electr. Eng. 99 (2022) 107810.

[28] V. Ravi, R. Chaganti, M. Alazab, Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system, Comput. Electr. Eng. 102 (2022) 108156.

[29] N. Gupta, V. Jindal, P. Bedi, CSE-IDS: Using cost-sensitive deep learning and ensemble algorithms to handle class imbalance in network-based intrusion detection systems, Comput. Secur. 112 (2022) 102499.

[30] C. Zhang, X. Costa-Pérez, P. Patras, Adversarial attacks against deep learning-based network intrusion detection systems and defense mechanisms, IEEE/ACM Trans. Netw. (2022).

[31] T. Alladi, V. Kohli, V. Chamola, F.R. Yu, M. Guizani, Artificial intelligence (AI)-empowered intrusion detection architecture for the Internet of Vehicles, IEEE Wirel. Commun. 28 (3) (2021) 144–149.

[32] A. Thakkar, R. Lohiya, Fusion of statistical importance for feature selection in deep neural network-based intrusion detection system, Inf. Fusion 90 (2023) 353–363.

[33] J.C.-W. Lin, Y. Shao, Y. Djenouri, U. Yun, ASRNN: A recurrent neural network with an attention model for sequence labeling, Knowl.-Based Syst. 212 (2021) 106548.

[34] S. Patro, K.K. Sahu, Normalization: A preprocessing stage, 2015, arXiv preprint arXiv:1503.06462.

[35] A. Faramarzi, M. Heidarinejad, S. Mirjalili, A.H. Gandomi, Marine predators algorithm: A nature-inspired metaheuristic, Expert Syst. Appl. (2020) 113377.

[36] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, A. Cosar, A survey on new generation metaheuristic algorithms, Comput. Ind. Eng. 137 (2019) 106040.

[37] N. Qun, H. Yan, X.-P. Qiu, X.-J. Huang, Chinese word segmentation via BiLSTM+ semi-CRF with relay node, J. Comput. Sci. Tech. 35 (5) (2020) 1115–1126.

[38] M. Tavallaee, E. Bagheri, W. Lu, 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, Ieee, 2009, pp. 1–6.

[39] Y. Djenouri, G. Srivastava, J.C.-W. Lin, Fast and accurate convolution neural network for detecting manufacturing data, IEEE Trans. Ind. Inform. 17 (4) (2020) 2947–2955.

[40] Y. Djenouri, A. Belhadi, J.C.-W. Lin, Recurrent neural network with density-based clustering for group pattern detection in energy systems, Sustain. Energy Technol. Assess. 52 (2022) 102308.

[41] Y. Djenouri, M. Comuzzi, Combining apriori heuristic and bio-inspired algorithms for solving the frequent itemsets mining problem, Inform. Sci. 420 (2017) 1–15.

[42] Y. Djenouri, D. Djenouri, J.C.-W. Lin, A. Belhadi, Frequent itemset mining in big data with effective single scan algorithms, Ieee Access 6 (2018) 68013–68026.

[43] Y. Djenouri, A. Belhadi, H.-C. Chen, J.C.-W. Lin, Intelligent deep fusion network for urban traffic flow anomaly identification, Comput. Commun. 189 (2022) 175–181.