2022 10th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud) | 978-1-6654-9845-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/MobileCloud55333.2022.00011

An Explainable Deep Neural Framework for Trustworthy Network Intrusion Detection

Souradip Roy
Department of Computer Science
North Dakota State University
Fargo, USA
souradip.roy@ndsu.edu

Juan Li
Department of Computer Science
North Dakota State University
Fargo, USA
j.li@ndsu.edu

Vikram Pandey
Department of Computer Science
North Dakota State University
Fargo, USA
vikram.pandey@ndsu.edu

Yan Bai
School of Engineering and Technology
University of Washington Tacoma
Tacoma, USA
yanb@uw.edu

***Abstract*— In recent years, there has been an increase in cyber-attacks in mobile cloud environment. Intrusion Detection Systems (IDS) have played an important role in protecting mobile cloud security. Many techniques have been utilized to implement IDS, among them, machine learning-based techniques have generated promising results. Especially, complex deep neural networks show a higher detection rate than traditional machine learning models. However, the interpretation of the decision made by a neural network becomes harder to understand as its architectural complexity increases. This challenge makes it difficult for the human experts to fine-tune their detection systems, trust the detection system's results, and make decisions accordingly when IDS systems are deployed. To address this issue, we propose an explainable intrusion detection framework that employs deep learning mechanisms to identify cyber-attacks and utilizes knowledge graph as the knowledge foundation to add human understanding of machine learning and explain the machine learning results. The use case study demonstrates that the proposed framework can not only successfully identify network intrusions but also effectively reveal important information about its internal working mechanisms of the mysterious deep learning Blackbox.**

Keywords—Explainable AI, Intrusion Detection, security, deep learning, knowledge graph

I. INTRODUCTION

The enormous growth of the Internet has increased the use of various internet-dependent devices. The high demand of network usage has elevated concerns about the security of these devices. These devices transfer a vast range of personal information such as web camera footage, health information, financial information, via Internet that are vulnerable to various kinds of malicious activities [1]. It is becoming even worse with the popularity of mobile and cloud computing technologies. Internet security vulnerabilities makes intrusion detection systems (IDS) an important and essential technology that provides a more secure environment. IDS can identify malicious activities from legitimate ones using different techniques such as signature based logical operations, statistical analysis, machine learning and data mining techniques [2].

Various machine learning techniques such as Decision Trees [3], [4], [5], Support Vector Machines [6], [7], [8], Random Forest [9], [10], Ensemble Learning [11], [12] to identify intrusions in the network. More recently, researchers have adopted deep learning-based intrusion detection models to implement IDS. These deep learning-based systems have proved to be more effective in identifying cyber-attacks by producing higher detection rate and lower false positive rate [21]. However, the rational and clarity behind the deep learning models' decision making are missing. The decision of intrusion detection system is used by human administrators to take appropriate actions against abnormal behavior in the network. Therefore, it is important for administrators to understand a model including what dataset was used for training, what machine learning approach was adopted and how the training was performed.

To address the aforementioned problem, in this paper, we suggest a knowledge graph-assisted deep neural network framework to identify network intrusions with clear explanation of how the detection mechanism works, so that human users can understand and trust the results generated by the proposed IDS system. A deep-learning based approach has been applied to classify network traffic. Knowledge graphs and ontologies. are employed to provide background knowledge of the explanation in the framework. The proposed framework explains dataset features, machine learning models, and, most importantly, prediction results. The framework based on knowledge graphs improves understanding of the machine learning models and prediction results for cybersecurity experts. We have implemented and tested the proposed model using a public datasets CICIDS2017 [22]. Our experiments have shown that the deep neural model generates high precision results in detecting different cyber-threats with little to negligible false positive values. Moreover, the explainable results aid cybersecurity experts interpret the achieved system outputs.

The rest of the paper is organized as follows. Section II surveys related research in intrusion detection systems and explainable AI in various domains. Section III provides details of the proposed framework. Section IV explains the experimental results. Section V concludes the paper and points out our future work.

This work was supported in part by the National Science Foundation (NSF) with award number: 1722913 and 1921576

978-1-6654-9845-6/22/$31.00 ©2022 IEEE
DOI 10.1109/MobileCloud55333.2022.00011

II. Related Work

Over the last decade, many machine learning, especially deep learning-based intrusion detection systems such as [23], [24], [25] have been developed to identify cyber-attacks. Popular shallow learning algorithms used for intrusion detection include Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Artificial Neural Network (ANN), and Ensemble-based approaches. Moon et al. [26] proposed an IDS based on the decision tree to classify network attacks. The decision tree uses nodes' behavior information to detect advanced persistent threats attacks that can change after an intrusion occurs. Similar IDS systems [3], [4], [5] also use decision trees to classify malicious behavior and prevent attacks. KNN utilizes feature similarity to classify the class of an intrusion. In their work, Shapoorifard and Shamsinejad improved intrusion detection tasks by combining K-means clustering and KNN classification [27]. Ádám et al. proposed an anomaly-based Network Intrusion Detection System that uses artificial neural network [28]. The proposed system is able to successfully recognize learned malicious activities such as the SYN flood attack, UDP flood attack, nMap scanning attack, and non-malicious communication in a network environment. A similar work by Dias et al. also used ANN to classify network traffics and achieved good performance [29].

Compared with the aforementioned shallow learning approaches, recent research in applying deep learning algorithms in intrusion detection achieves better detection performance. Many deep learning-based intrusion detection models [13], [14], [15], such as Convolutional Neural Network (CNN), Recurrent Neural Networks (RNN), variational autoencoder [20], have been proposed and evaluated. Zhang et al. proposed an intrusion detection model based on CNN [30]. Before CNN training, they applied Synthetic Minority Oversampling Technique and Edited Nearest Neighbors (SMOTE-ENN) algorithm to balance the imbalanced data. Their evaluation results have demonstrated that SMOTE-ENN-based CNN IDS model achieves good accuracy and better detection rates of User to Root (U2R) and Remote to Local (R2L) attacks. Similarly, IDSs proposed in [16] [17] [18] also used CNN for classification. Studies presented in [15] and [19] proposed deep learning approaches for intrusion detection using RNN. They applied RNN for both binary classification and multiclass classification and found that RNN IDS can achieve good accuracy. Narayana et al. found out that detection rate in IDS can be improved by using variational autoencoder and DNN [31].

These deep learning-based intrusion detection models have gained noticeable detection rate for abnormal network activities; however, they lack the capability to explain the machine learning models and the decisions made by the models. To overcome this problem, researchers have been working on proposing systems that can generate information to explain the machine learning's prediction results. For example Bach and Binder introduced the so-called "layer-wise relevance propagation" to visualize the contribution of single pixels as heatmaps and help human experts to verify the validity of the classification decision. The layer-wise relevance propagation method was used in another research work done by Marino et al. [32]. However, they provided explanations only for misclassifications performed by the IDS. They utilized adversarial approach to generate these explanations. Amarasinghe et al. [33] proposed a DNN-based anomaly detection system that also provides explanation of the detected anomalies. The explanation feedback was generated feedback based on the feature contribution in the classification process. "Local Explanation Method using Nonlinear Approach" (LEMNA) was utilized by Li et al. [34] to generate the explanations for anomaly-based IDS.

To the best of our knowledge, existing research focused on providing explanation for certain types of machine learning results. A compressive and complete explanation model is still missing. This paper tries to address the problems of existing systems and proposed a compressive detection and explanation framework to not only detect intrusions but also explain how results are predicted, and why they are predicted in that way.

III. Methodology

We propose an intrusion detection with multi-modal explanation framework based on network intrusion detection-related domain ontologies. Besides identifying attacks, this framework is able to provide explanation and insights from different perspectives of a detection model.

Fig. 1 illustrates the architecture of the proposed framework. Input network traffic metadata is passed to the IDS system, which is a DNN trained by specific dataset. Prediction in benign or malicious traffic will be made by the DNN. The entire prediction process will be explained using an explanation model. The Knowledge Base provides machine-interpretable representations about the entire prediction process. It consists of ontologies about machine learning and network intrusion. The explanation module explains dataset features, machine learning models and most importantly the prediction results. The framework improves understanding of machine learning models and prediction results for cybersecurity experts.

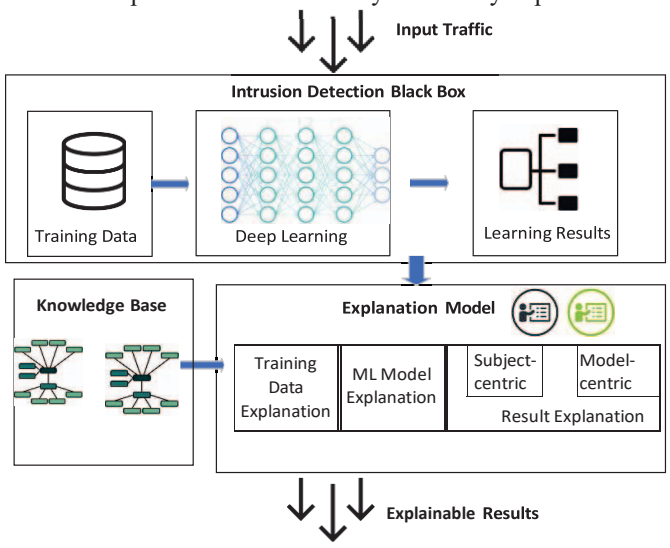

Fig. 1. System architecture

The remainder of this section contains and explains details about the framework.

A. Knowledge Base

The knowledge foundation of our explainable model includes two ontologies: one defines important concepts and their relations in network security; another defines important concepts and relations in machine learning. Together, they provide background information about the proposed intrusion detection model. There are attempts to define taxonomy of network attacks [35]. However, none of them can sufficiently describe current network attacks and related traffic characteristics. Hence, we have created a comprehensive ontology to model the upper-level concepts of network traffic and security by incorporating taxonomy and concepts presented in [35] [36]. The ontology defines major concepts such as types of attacks, tools used by attackers, network traffic features and so on.

Fig. 2. Part of the network security ontology's classes

Fig. 2 shows high-level ontology about network security. As shown in Fig. 2 general concepts are extended to incorporate subclasses. For instance, types of attacks are extended to include botnet attacks, DoS attacks and DDoS attacks. Tools for attacks are extended to include tools such as Hulk, Golden eye, etc. These ontology concepts help in identifying the type of an attack along with determining network vulnerabilities exploited in the attack.

Similarly, high-level ontology about machine learning models have also been developed which reorganizes and reuses some existing machine learning ontologies [38]. Due to space limit, we can only show a small part of the ontology. A complete version can be found in our GitHub project site [37].

B. Intrusion Detection Model Using Deep Neural Network

The detection model aims at constructing an efficient classifier model to detect various attacks as accurately as possible. For this purpose, we build a Feed Forward multiclass classifier. The input layer has 78 neurons whereas each hidden layer contains 1024,768,768,512 neurons, respectively; the output layer has 7 neurons. The presence of such high neurons in the hidden layers understanding of data processing becomes overly complicated. To understand the data processing by the neural network we need to rely on the explanation model to add transparency to the decision making of the model. The Feed Forward neural network is chosen because of its simplistic design structure although other complex neural network architectures, such as CNN and RNN, can also be used as the classifier model.

The classifier model is trained using stochastic gradient descent learning method [39]. This learning method minimizes the loss value based on a loss function's mean squared error, MSE, which is defined in Equation 1.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y')^2 \qquad (1)$$

In Equation 1, Y_i is the original class label of the data sample, and Y' is the predicted class label for the same sample. The input data is transferred from one hidden layer to another in forward direction. The model tries to minimize the difference between the original value and the predicted value as much as possible by updating the weights of different layers via a back-propagation method at each iteration.

Our neural network contains dropout layers. Dropout is a regularization method that helps prevent the neural network from overfitting. Overfitting can be a problem when a neural network is very dense, i.e. the neural network contains huge number of neurons in its hidden layers. Using the dropout regularization method, the neural network randomly chooses certain number of neurons in its layers and decides to discard the output value produced by that neuron and sends the remaining values to the next layer. Based on multiple experiments, we finalized 10% of the neurons to be discarded at each layer. The use of dropout layer also helps the neural network to generalize its decision while performing classification of the input. Each hidden layer has a ReLu activation function [40] for computation purpose. This activation function has been finalized after neurons experiments which proves that ReLu produces the best output. Equation 2 shows the ReLu activation function *g(x)*.

$$g(x) = \max\{0, x\} \qquad (2)$$

The intrusion detection deep learning model performs a multiclass classification. Therefore, the last layer contains SoftMax activation [41] as shown in Equation 3. The input sample then gets classified as a class label with the highest probability.

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \qquad (3)$$

C. Explanation Model

The explanation model explains the intrusion detection from three aspects: (1) the machine learning model used to classify network intrusions, (2) the data used to train and test the machine learning model, and (3) the results generated by the machine learning model. When data scientists build a machine learning model, instances will be generated based on the model using the ontology defined in Section A. All the metadata about the machine learning model, such as the machine learning algorithm and parameters, are stored in the knowledge in the format of ontology. Query and reasoning can be performed on the knowledgebase for explanation purposes. To understand the data ɔd for training and testing of the machine learning model, data ıtures are mapped to the network security ontology.

The mapping is through a semantic k-nearest neighbor search process we proposed in a previous work [42].

We explain results generated by the deep neural network from two perspectives: model-centric and subject-centric explanation. Model-centric explanations explain the whole model, while subject-centric explanation explain the model decisions for individual predictions. Both models' explanations are based on the explanatory of features with the greatest influence on the prediction model. We adopted Shapely Additive Explanations (SHAP) [43] for the detection model explanation. SHAP is very suitable for explaining the decision of complex models such as DNN. SHAP is derived from Shapley Values in game theory. It interprets and describes the impact of each feature for a specific data sample. This process of interpretation is executed for all the data samples, which results in the global explanation of the feature importance of the dataset. We adopted the Local Interpretable Model-Agnostic Explanations (LIME) [44] for subject-centric explanation. LIME tries to find a simple model that locally approximates the complex machine learning model *f()* in the vicinity of a certain test instance *x*.

IV. Evaluation

We evaluate the proposed explainable IDS from two aspects: the performance of the DNN detection model and the explainability of the IDS system.

A. Detection Model Evaluation

1) Dataset and Data Preprocessing

We use the CICIDS2017 dataset [22] for training and testing of our intrusion detection models. It includes 78 independent features, and a total of 14 types of attack traffic and normal traffic. The dataset is highly imbalanced considering the number of samples in each type of attack.

Therefore, we grouped similar attack types into a single category. The grouping of class variables reduces the number of classes into 6 types of malicious traffic activities and 1 normal traffic activity. We used a series of sampling techniques including Random Under Sampling with replacement [45] and SMOTE [46] oversampling technique to increase the number of samples in the minority class(es), and Tomek Links [47] to remove noise after applying SMOTE. All these techniques help us gain a well-balanced and organized subset of the CICIDS2017 dataset that is then used in the training and evaluation of our framework. We used Quantile Transformer [48] as the scaling technique to scale the dataset because it can produce scaled output that follows the same distribution (Uniform) as the original dataset. Quantile Transformer scaling method is also very robust towards outliers that helps scale the data correctly and doesn't remove the outliers from the scaled dataset.

2) Evaluation Results

We evaluated our intrusion detection deep learning-based neural network based on Accuracy, Precision-Recall, AUC (Area Under the Curve). These metrices are developed for binary classification systems. Since our model performs a multiclass classification, we converted our problem to binary classification using the One-vs-Rest strategy [49].

In Fig. 4 we plot the ROC Curve for test data. This plot helps us to understand how well our deep learning-based IDS performs for each class. We compared the performance of our

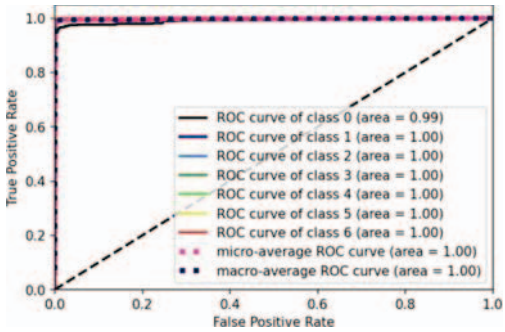

Fig. 4: ROC Curve of Intrusion Detection Model

intrusion detection model with other state-of-the-art intrusion detection models including RKM [23], KNN-SVM-RBF [23], Decision Jungle [24], MNN [24], MDF [24], MLR[24], and DNN 5 layers [25] in Table I. Metrices which have been utilized for comparing are accuracy, precision, recall, and FI.

Table I: Performance Comparison of our Intrusion Detection Model with state-of-the-art IDS Models

Model	*Accuracy*	*Precision*	*Recall*	*F1-*
RKM	98.04	99.86	93.29	95.99
KNN-SVM-RBF	98.97	99.90	94.42	97.08
Decision Jungle	96.56	93.92	86.19	-
MNN	91.55	90.34	83.98	-
MDF	92.78	91.99	85.80	-
MLR	90.60	91.76	84.90	-
DNN 5 Layers	95.6	96.2	95.6	-
Proposed DNN	**99.2**	**99.2**	**99.6**	**99.0**

B. Detection Result Explanation

First set of experimental cases explain how individual sample is predicted. Fig. 6, show the explanation of why a particular individual sample *x* is classified as a certain class. The explanation was produced with the help of LIME. It identifies the probability of each feature that contributes to the classification of *x* by the proposed DNN model. Due to space limitation, we presented the interpretation of 1 class (i.e., classification of benign or malicious only. We have used LIME to produce the local interpretation of all different features for all the 7 classes. Fig. 7 shows that *x* was eventually classified as a DDoS attack.

Fig 6: Contribution of each feature in detecting if x is benign or malicious traffic

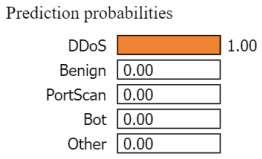

Fig 7: Final prediction of sample *x*

The second experimental cases explain how the DNN model make the classification. Fig. 8 shows the global explanation using the 20 features. It can be noticed in Fig. 6 that the global 20 features are the only feature which contributes to the probability of local interpretation. The result was created by SHAP. It explains the impact or importance of each feature towards the classification results.

Fig. 8: Feature Impact on DNN model decision Global Explanation

C. *Detection Model Explanation*

The ontology-based knowledgebase can help human experts to understand how diabetes risk prediction is made. The proposed system supports the model explanation through answering human's natural language's query and visualizing a dataset's features. Fig. 11 shows a visualization of the CICIDS2017 dataset that was used to train and test the prediction model. It visualizes features using a radial tree. The tree corresponds data features in the dataset with concepts in the knowledge graph. The hierarchical structure helps users to gain a general-to-specific understanding of each feature. For example, from Fig. 9, a user can understand that the feature "BwdURGFlags" belongs to "Flag Features", which, in turn, is a traffic feature. When the user clicks on a particular feature, for example, the "BwdURGFlags" feature, its detailed explanation retrieved from the ontological knowledgebase will be provided as shown in the yellow text box in Fig. 9.

The system also provides a query interface for human experts to ask questions regarding the prediction model. User's questions will be converted to SPARQL queries to retrieve results from the ontology knowledgebase [50].

The following query use cases demonstrate how this function explains the prediction model.

- User's question: *"What kind of algorithm was employed on the particular dataset for classification?"*

SELECT ?algorithm WHERE {*?algorithm a ml:Algorithms .?algorithm ml:isAppliedOnData ?data .?data ml:isDatasetType ?datasetType . FILTER (?datasetType =ml: IDSDataset*)}

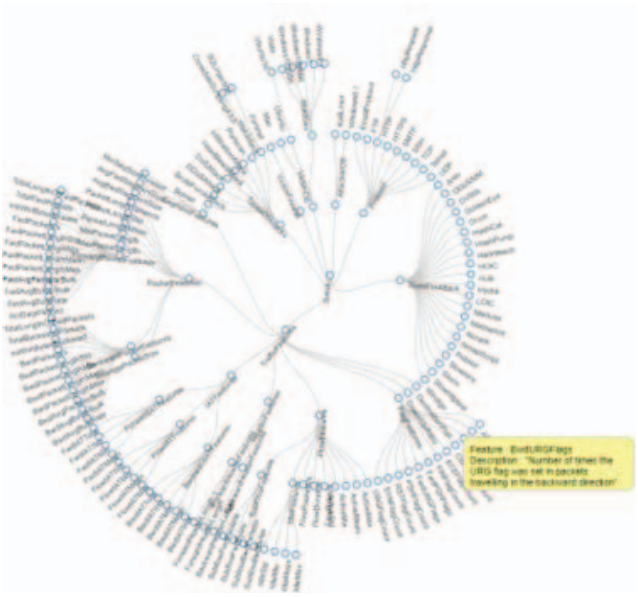

Fig. 9: Visualization-based explanation of data feature

V. CONCLUSIONS

This paper presents an explainable machine learning model that can not only detect network intrusions, but also explain the prediction process and results. It helps human experts to understand why machine learning detection model makes a particular prediction, and how the decision is made. The proposed model has been implemented and evaluated. The experimental results show the effectiveness of the proposed system. The explainable results also enable cybersecurity experts to interpret the detection result of the system. This will increase human expert's trust towards IDS.

ACKNOWLEDGEMENT

This work was supported in part by the National Science Foundation (NSF) with award numbers: 1722913 and 1921576.

REFERENCES

[1] M. Conti, A. Dehghantanha, K. Franke, and S. Watson, "Internet of Things security and forensics: Challenges and opportunities," *Future Generation Computer Systems*. 2018, doi: 10.1016/j.future.2017.07.060.

[2] A. Khraisat and A. Alazab, "A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges," *Cybersecurity*, 2021, doi: 10.1186/s42400-021-00077-7.

[3] B. Mahbooba, M. Timilsina, R. Sahal, and M. Serrano, "Explainable Artificial Intelligence (XAI) to Enhance Trust Management in Intrusion Detection Systems Using Decision Tree Model," *Complexity*, 2021, doi: 10.1155/2021/6634811.

[4] T. Baraas, A. Juliansyah, and A. A. Rizal, "Klasifikasi Data Log Intrusion Detection Sistem (Ids) Dengan Decision Tree C4.5," *J. Bumigora Inf. Technol.*, 2019, doi: 10.30812/bite.v1i2.609.

[5] P. Shukla, "ML-IDS: A machine learning approach to detect wormhole attacks in Internet of Things," 2018, doi: 10.1109/IntelliSys.2017.8324298.

[6] M. A. Ferrag, M. Babaghayou, and M. A. Yazici, "Cyber security for fog-based smart grid SCADA systems: Solutions and challenges," *J. Inf. Secur. Appl.*, 2020, doi: 10.1016/j.jisa.2020.102500.

[7] M. Safaldin, M. Otair, and L. Abualigah, "Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks," *J. Ambient Intell. Humaniz. Comput.*, 2021, doi: 10.1007/s12652-020-02228-z.

[8] E. Kabir, J. Hu, H. Wang, and G. Zhuo, "A novel statistical technique for intrusion detection systems," *Futur. Gener. Comput. Syst.*, 2018, doi: 10.1016/j.future.2017.01.029.

[9] M. A. M. Hasan, M. Nasser, B. Pal, and S. Ahmad, "Support Vector Machine and Random Forest Modeling for Intrusion Detection System (IDS)," *J. Intell. Learn. Syst. Appl.*, 2014, doi: 10.4236/jilsa.2014.61005.

[10] A. Y. Hussein, P. Falcarin, and A. T. Sadiq, "Enhancement performance of random forest algorithm via one hot encoding for IoT IDS," *Period. Eng. Nat. Sci.*, 2021, doi: 10.21533/pen.v9i3.2204.

[11] D. Stiawan *et al.*, "An Approach for Optimizing Ensemble Intrusion Detection Systems," *IEEE Access*, 2021, doi: 10.1109/ACCESS.2020.3046246.

[12] S. Roy, J. Li, B. J. Choi, and Y. Bai, "A lightweight supervised intrusion detection mechanism for IoT networks," *Futur. Gener. Comput. Syst.*, 2022, doi: 10.1016/j.future.2021.09.027.

[13] Z. Chiba, N. Abghour, K. Moussaid, A. El omri, and M. Rida, "Intelligent approach to build a Deep Neural Network based IDS for cloud environment using combination of machine learning algorithms," *Comput. Secur.*, 2019, doi: 10.1016/j.cose.2019.06.013.

[14] M. J. Kang and J. W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security," *PLoS One*, 2016, doi: 10.1371/journal.pone.0155781.

[15] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," *IEEE Access*, 2017, doi: 10.1109/ACCESS.2017.2762418.

[16] A. J. Kapoor, H. Fan, and M. S. Sardar, "Intelligent Detection Using Convolutional Neural Network (ID-CNN)," 2019, doi: 10.1088/1755-1315/234/1/012061.

[17] Y. Xiao, C. Xing, T. Zhang, and Z. Zhao, "An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks," *IEEE Access*, 2019, doi: 10.1109/ACCESS.2019.2904620.

[18] Q. Wang, W. Zhao, and J. Ren, "Intrusion detection algorithm based on image enhanced convolutional neural network," *J. Intell. Fuzzy Syst.*, 2021, doi: 10.3233/JIFS-210863.

[19] A. Dushimimana, T. Tao, R. Kindong, and A. Nishyirimbere, "Bi-directional Recurrent Neural network for Intrusion Detection System (IDS) in the internet of things (IoT)," *Int. J. Adv. Eng. Res. Sci.*, 2020, doi: 10.22161/ijaers.73.68.

[20] J. Kim, A. Sim, J. Kim, K. Wu, and J. Hahm, "Transfer Learning Approach for Botnet Detection Based on Recurrent Variational Autoencoder," 2020, doi: 10.1145/3391812.3396273.

[21] A. Nisioti, A. Mylonas, P. D. Yoo, and V. Katos, "From intrusion detection to attacker attribution: A comprehensive survey of unsupervised methods," *IEEE Commun. Surv. Tutorials*, 2018, doi: 10.1109/COMST.2018.2854724.

[22] A. Boukhamla and J. C. Gaviro, "CICIDS2017 Dataset: Performance Improvements and Validation as a Robust Intrusion Detection System Testbed," *Int. J. Inf. Comput. Secur.*, 2018.

[23] M. Yousefnezhad, J. Hamidzadeh, and M. Aliannejadi, "Ensemble classification for intrusion detection via feature extraction based on deep Learning," *Soft Comput.*, 2021, doi: 10.1007/s00500-021-06067-8.

[24] S. Rajagopal, P. P. Kundapur, and K. S. Hareesha, "Towards Effective Network Intrusion Detection: From Concept to Creation on Azure Cloud," *IEEE Access*, 2021, doi: 10.1109/ACCESS.2021.3054688.

[25] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," *IEEE Access*, 2019, doi: 10.1109/ACCESS.2019.2895334.

[26] D. Moon, H. Im, I. Kim, and J. H. Park, "DTB-IDS: an intrusion detection system based on decision tree using behavior analysis for preventing APT attacks," *J. Supercomput.*, 2017, doi: 10.1007/s11227-015-1604-8.

[27] H. Shapoorifard and P. Shamsinejad, "Intrusion Detection using a Novel Hybrid Method Incorporating an Improved KNN," *Int. J.Comput. Appl.*, 2017, doi: 10.5120/ijca2017914340.

[28] N. Ádám, B. Madoš, A. Baláž, and T. Pavlik, "Artificial neural network based IDS," 2017, doi: 10.1109/SAMI.2017.7880294.

[29] L. P. Dias, J. J. F. Cerqueira, K. D. R. Assis, and R. C. Almeida, "Using artificial neural network in intrusion detection systems to computer networks," 2017, doi: 10.1109/CEEC.2017.8101615.

[30] X. Zhang, J. Ran, and J. Mi, "An Intrusion Detection System Based on Convolutional Neural Network for Imbalanced Network Traffic," 2019, doi: 10.1109/ICCSNT47585.2019.8962490.

[31] K. Narayana Rao, K. Venkata Rao, and P. R. Prasad, "A hybrid Intrusion Detection System based on Sparse autoencoder and Deep Neural Network," *Comput. Commun.*, 2021, doi: 10.1016/j.comcom.2021.08.026.

[32] D. L. Marino, C. S. Wickramasinghe, and M. Manic, "An adversarial approach for explainable AI in intrusion detection systems," 2018, doi: 10.1109/IECON.2018.8591457.

[33] K. Amarasinghe, K. Kenney, and M. Manic, "Toward explainable deep neural network based anomaly detection," 2018, doi: 10.1109/HSI.2018.8430788.

[34] H. Li, F. Wei, and H. Hu, "Enabling dynamic network access control with anomaly-based IDS and SDN," 2019, doi: 10.1145/3309194.3309199.

[35] S. Hansman and R. Hunt, "A taxonomy of network and computer attacks," *Comput. Secur.*, 2005, doi: 10.1016/j.cose.2004.06.011.

[36] A. Georgiadou, S. Mouzakitis, and D. Askounis, "Assessing mitre att&ck risk using a cyber-security culture framework," *Sensors*, 2021, doi: 10.3390/s21093267.

[37] "No Title." .

[38] J. Braga, F. Regateiro, J. L. R. Dias, and I. Stiubiener, "A machine learning domain ontology to populate knowledge base to support intelligent agents working in autonomous systems domains of the Internet infrastructure," 2021, doi: 10.5753/wpietf.2021.15778.

[39] M. Kobayashi, "Gradient descent learning for quaternionic Hopfield neural networks," *Neurocomputing*, 2017, doi: 10.1016/j.neucom.2017.04.025.

[40] K. Eckle and J. Schmidt-Hieber, "A comparison of deep networks with ReLU activation function and linear spline-type methods," *Neural Networks*, 2019, doi: 10.1016/j.neunet.2018.11.005.

[41] Q. Zhu, Z. He, T. Zhang, and W. Cui, "Improving classification performance of softmax loss function based on scalable batch-normalization," *Appl. Sci.*, 2020, doi: 10.3390/APP10082950.

[42] R. Hendawi, S. Alian, and J. Li, "A Smart Mobile App to Simplify Medical Documents and Improve Health Literacy: System Design and Feasibility Validation.," *JMIR Form. Res.*, vol. 6, no. 4, p. e35069, Apr. 2022, doi: 10.2196/35069.

[43] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," 2017.

[44] M. T. Ribeiro, S. Singh, and C. Guestrin, "Anchors: High-precision model-agnostic explanations," 2018.

[45] S. Bagui and K. Li, "Resampling imbalanced data for network intrusion detection datasets," *J. Big Data*, 2021, doi: 10.1186/s40537-020-00390-x.

[46] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, 2002, doi: 10.1613/jair.953.

[47] T. Sasada, Z. Liu, T. Baba, K. Hatano, and Y. Kimura, "A resampling method for imbalanced datasets considering noise and overlap," 2020, doi: 10.1016/j.procs.2020.08.043.

[48] L. Rüschendorf, "Copulas, Sklar's Theorem, and Distributional Transform," in *Springer Series in Operations Research and Financial Engineering*, 2013.

[49] "Pattern Recognition and Machine Learning," *J. Electron. Imaging*, 2007, doi: 10.1117/1.2819119.

[50] V. Pandey, J. Li, and S. Alian, "Evaluation and Evolution of NAOnto - An Ontology for Personalized Diabetes Management for Native Americans," *2021 7th Int. Conf. Comput. Commun. ICCC 2021*, pp. 1635–1641, 2021, doi: 10.1109/ICCC54389.2021.9674339.