SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal
of Cryptocurrency Research

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Optimization Model for Intrusion Detection System in IoT Applications

Abeer. Y.A. Salawi[1] and Mohammed. I. Alghamdi[2*]

[1]College of Computer Science and Information Technology, Department of Engineering and Computer Sciences, Al-Baha University,, Al-Baha City, Kingdom of Saudi Arabia E-mail: 442020222@stu.bu.edu.sa

[2]College of Computer Science and Information Technology, Department of Engineering and Computer Sciences, Al-Baha University, Al-Baha City, Kingdom of Saudi Arabia E-mail: mialmushilah@bu.edu.sa

Article Info

Volume 2, Issue 1, June 2022
Received : 19 April 2022
Accepted : 22 May 2022
Published : 05 June 2022
doi: 10.51483/IJCCR.2.1.2022.41-51

Abstract

In the era of the Internet of Things (IoT), connected objects produce an enormous amount of data traffic that feed big data analytics, which could be used in discovering unseen patterns and identifying anomalous traffic. In this paper, we identify five key design principles that should be considered when developing a deep learning-based Intrusion Detection System (IDS) for the IoT. Based on these principles, we design and implement Temporal Convolution Neural Network (TCNN), a deep learning framework for intrusion detection systems in IoT, which combines Convolution Neural Network (CNN) with causal convolution. TCNN is combined with Synthetic Minority Oversampling Technique-Nominal Continuous (SMOTE-NC) to handle unbalanced dataset. It is also combined with efficient feature engineering techniques, which consist of feature space reduction and feature transformation. TCNN is evaluated on Bot-IoT dataset and compared with two common machine learning algorithms, i.e., Logistic Regression (LR) and CNN. Experimental results show that TCNN achieves a good trade-off between effectiveness and efficiency. It outperforms the state-of-the-art deep learning IDSs that are tested on Bot-IoT dataset and records an accuracy of 99.9986% for multiclass traffic detection and shows a very close performance to CNN with respect to the training time.

Keywords: *Cascade Forward Neural Network, Internet of things, Intrusion Detection System, Metaheuristics, Political Optimizer, Neural Network*

1. Introduction

The Internet of Things (IoT) implies that everything and everybody is connected billions of individuals connected to billions of devices creating a relentless flow of world data, in real time, at massive volumes. IoT provides secure, bi-directional communication between Internet-connected things (such as sensors, actuators, embedded devices, or smart appliances) and also the cloud. We discover our custom IoT-Data endpoint to speak with, configure rules for processing and integration with other services, organize resources related to each thing (Thing Registry), configure logging, make and manage policies and credentials to authenticate things. Application Programming Interfaces (API) is to store and retrieve data from things using the HTTP protocol over the net or via a Neighborhood Area Network (LAN). It enables location tracking applications, a social network of things with status updates and therefore the creation of sensor logging applications. API is that the logical connectors that allow applications to speak with each manufacturer's IoT

* *Corresponding author: Mohammed. I. Alghamdi, College of Computer Science and Information Technology, Department of Engineering and Computer Sciences, Al-Baha University, Al-Baha City, Kingdom of Saudi Arabia. E-mail: mialmushilah@bu.edu.sa*

devices. APIs expose data that allows those devices to transmit data to the applications, acting as a knowledge interface, or they will allow the appliance to manage the device and function a function interface. With IoT technology, we are able to monitor the mushroom house wherever we are using our smartphone. Unpredictable weather change gave problem to mushroom farmers. When different atmospheric phenomenon occur, the standard of mushrooms produced may vary cause instability in quality of the mushroom being produced. With using sensor, the water sprinkle are function when the humidity of the mushroom house was too low. The IoT revolves around expanded machine to machine correspondence, its supported cloud computing and systems of knowledge event sensors, its portable, virtual, and prompt connection and that they say its visiting make everything in our lives from streetlights to seaports. Gradual improvement of IoT has cause a revolution within the wireless communication. With a growing technology of IoT, a giant range of smart and tiny wireless sensing devices are deployed for type of application environments. A misconception is that the IoT can be a monolithic thing, but it's more than that after all is more sort of a phenomena there is no simple explanation for all the technologies involved within the movement. What we all understand are the real-life applications: Smart City, Connected Cars, Lightning, Fleet Management and every one the employment cases in Industrial IoT. "Initially, leaders viewed the IoT as a solution, a technology which will solve the myriad IT and business problems that their organizations faced. Very quickly, though, they recognized that without the right framing of the issues, the IoT was essentially an answer searching for a controversy.

Now the framing is more clear, we have several used cases where IoT is a good fit. As in the telecommunication industry, there are lot of technologies and solutions going on behind the scene that compose that framework, such as IoT Platforms, Security, IoT Devices, Edge Gateways, etc. and of course, the connectivity network. So in this world, what is the role of the CSP in this Digital IoT transformation, and a better question, there will be any profit in it for them? One of many beliefs is that operators will be the leaders in IoT Platform solutions, but in fact the ones that are leading the change in that regard are the major vendors of consumer electronics or industrial assets, which are the one that will benefit the most from the device Philips with Healthsuite ecosystem and Philips Lightning, and Bosh with Bosh IoT Platform are clear examples of this. What is the most likely outlook for the customer market: A customer buys an appliance to their favorite retailer, no matter which one, will bring the device home, the device self-connects to internet (through a mobile network) and the customer register the device in the vendor's mobile app and everything is working. In that scenario what I foresee within the upcoming years may be a booming of IoT platforms within the cloud. The large vendors for appliances will have their own (with management portals and SDKs), there will be other players which will act as integrators with only platform so middle size companies can ride that and in fact some CSPs which will venture into that, will have their own which will be a good suited small business. These platforms will see one another in any case where the knowledge from one device have synergies with others, and a bit like that we have the communication era for the machines.

Security is one in every of the key concerns of the IoT. As billions of objects are going to be connected to that within the future. There has been an ever increasing must address the safety issues associated with the IoT. The data transmission across the networks have to be monitored and guarded from thefts and unauthorized malicious attacks. Different kinds of protocols and security measures are available within the existing internet scenario but most of them have limited applicability within the domain of IoT.

2. Background

With the variety of cases of recent security attacks, the importance of security in IoT applications and the consequences of these attacks increases. IDS will need to improve to take into account new features and attacks in IoT applications. In this section we put some basic information, starting with the IoT and its attacks and then IDS, then how PO-CFNN can contribute in improving IDS to detect threats and intrusion on IoT applications.

3. Internet of Things

For the purpose of this study, the IoT is defined from a technical perspective as a network of physical objects that send and receive data through an object in which there are many sensors, and then the data is received from the sensors and shared on the network.

4. IoT Attacks

The debate: the most prominent cyberattacks on the Internet of Things, and these attacks were labeled into two categories, the first type is passive security attacks that threaten network security through data traffic that does not cause effects on the network, and this category is difficult to detect early (such as spyware, Passive Token Injection, Flood, and Sybil) the second type are active security attacks that cause a direct impact on data and information security (such as spoofing, MITM, DoS, Active Token Injection).

5. Architecture of IDS in IoT

The review of IDS in IoT, IoT hacking is defined as activity that is harmful to the IoT ecosystem, and any attack that cripples the movement or availability of a system or service is an intrusion, IDS is defined as a system or device that detects activities or any suspicious network traffic to maintain the security of computer systems, IDS is categorized in IoT into three categories, IDS rating according to Placement strategy, Detection methods, and Validation methods, Figure 1 indicates classes IDS in IoT community in detail.

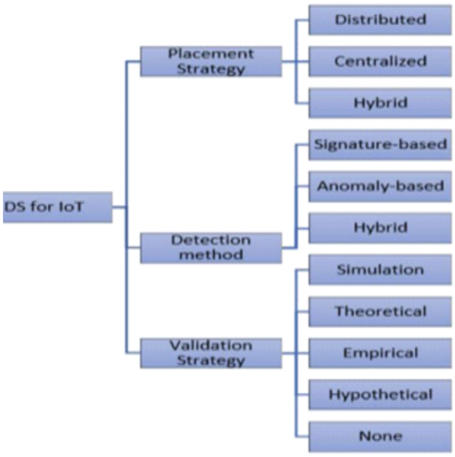

Figure 1: Classes IDS in IoT

5.1. PO-CFNN

5.1.1. CFNN

ANN modeling has evolved greatly in recent years with the developments in the computer field, it is defined as a network of interconnected neurons that process the input data with the aim of classifying the information and recognizing patterns through learning, in which there are many inputs and one output in which Exchange and send signals to analyze and infer results directly from the input layer to the output layer and this relationship is called linear relationship, NN produces greater and better accuracy in prediction operations. The idea that Distinguishing CFNN's work is that it has a direct data analytic relationship between the input layer and the output layer, as well as an indirect data analytic relationship via the hidden layer. This method explores the nonlinear relationships between two variables between the inputs and the outputs. Figure 2 shows how it CFNN works.

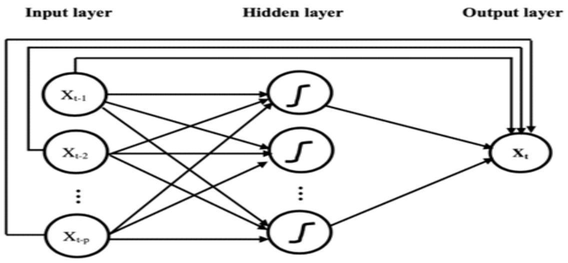

Figure 2: Architecture of CFNN

5.1.2. PO

It is an algorithm that takes the same approach as human behavior in political systems, consisting of five stages to filter the optimal solution: (1) Party formation and Constituency Allocation; (2) Election campaign; (3) Party switching; (4) Election; and (5) Parliamentary affairs. It is expected that this algorithm will be used to train FNN weights with the aim of minimizing the error between the outputs. The process of the PO algorithm is summarized in Figure 3 shown below: (a) the first stage is preparing the population; (b) polling the opinions of political party members; (c) Determine the heads of the parties and election winners; (d) Reupdate party members' positions based on winners in electoral districts; (e) Re-updating the party members' positions by the heads of the parties; (f) summing up positions according to constituency winners and party members; (g) and (h) exchanging parties; (i) reappointing party leaders and winners in constituencies; (j) switching parties by updating Member of Parliament, if appropriate, after calling a random Member of Parliament; and (k) the winner's final position in the constituency.

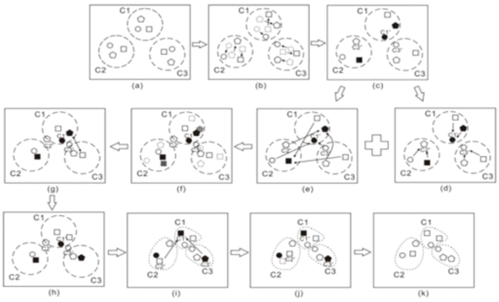

Figure 3: PO Process

6. Challenges of IoT Security

As IoT runs in parallel with the internet, it also inherits the security issues present in the internet as explained below.

- End to end security is the process of protecting the data transfer between the two ends of the communication without being eavesdropped, intercepted, or modified. The heterogeneous nature of the IoT and its devices involve the sharing of information between things across many networks. When they operate using different technologies, (e.g., 802.11 vs. 802.15.4) establishing a secure communication session becomes a complex task to achieve. Each device in the IoT may have limited computation capabilities.
- Data security is the process of protecting data from unauthorized access. Security breaches in the IoT can seriously threaten human safety. For example, security breaches in a forest fire detection system could lead to catastrophic results.
- Identity and access management: In the IoT, the processes of identification, representation and accesses to things are yet to be known. For example, a masquerade attack can occur where one device could use a fake identity to gain illegal access to services of another IoT device.
- Compliance: Things in IoT need to comply to government laws and regulations in order to preserve security of their systems as most of these systems are built as autonomous things with their own privacy features.
- Access control is the process of granting and restricting the access to a particular resource (for e.g., role-based access control). The low cost of things is an important factor which drives the support for large scale deployment of IoT devices. However, this requirement makes the system more resource constrained. Conventional Internet security solutions cannot provide complete security for the IoT architecture. In IoT, users might require access to data anytime or from anywhere with various types of devices, and hence the IoT poses a new challenge to access control.
- Physical and DoS Security Risks Network equipment is vulnerable to physical attacks like Denial of Service (DoS) and Distributed Denial of Service (DDoS). As many IoT devices have limited processing and memory capabilities, such attacks could easily exhaust the system resources. The presence of embedded devices in public locations can create opportunities for malicious attackers to exploit and use the user's private information. For example, a home GPS system can provide attackers with the user's location by tracking the vehicle.

7. Attacks on IoT Ecosystem

As IoT technology involves many devices like sensors, processors and plenty of other technologies, the aim of sharing the information and connecting to other networks has been served successfully, because it involves many devices connected, the info shared might not be secure and therefore the security concern raises. IoT Security refers to guard the knowledge shared among different networks through IoT devices using IoT technology. These devices are connected to others using the web which allows vulnerabilities to require place by allowing the hacker to hack the information. Data without the safety will cause many concerns and brings huge loss for several industries and even to the individuals ending with the loss of the info from their systems. As I mentioned before, IoT grabbed the eye of the people and also the organizations from many sectors onto it, by providing extreme benefits to them together with its tremendous growth, some security issues have risen by which IoT attacks have taken place by preventing people to use many of its upcoming applications. IoT devices is accessed from anywhere within a trusted network. So, there are chances of many malicious attacks within the IoT network. Hence, security, privacy, and confidentiality issues must be appropriately addressed within the IoT to safeguard it from malicious attacks. For instance, the attacking of traffic lights and driverless vehicles not only reasons chaos and rises contamination but can also initiate harm and severe collisions resulting in wounded.

Figure 4 shows the IoT system architecture with layers where attacks can occur. An IoT system can comprise three fundamental layers which are the perception layer, network layer, and application layer. The perception layer is the lowest layer of the conventional architecture of IoT. This layer consists of devices, sensors, and controllers. This layer's fundamental task is to gather valuable information from IoT sensors systems.

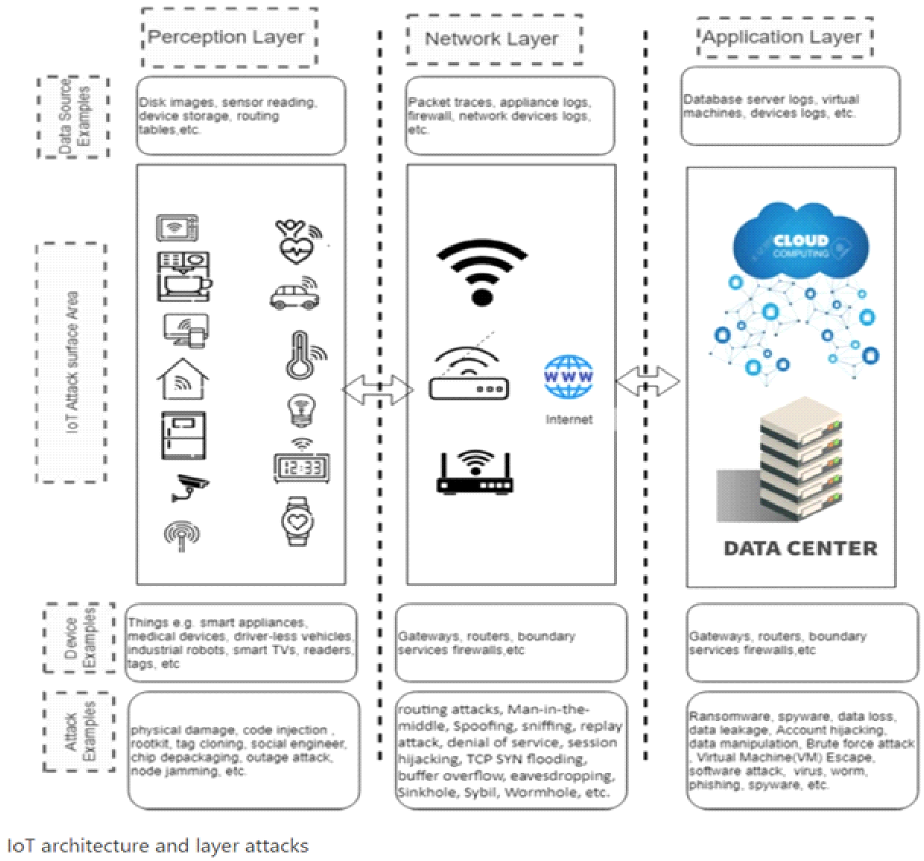
Perception Layer
Network Layer
Application Layer
Data Source Examples
Disk images, sensor reading, device storage, routing tables,etc.
Packet traces, appliance logs, firewall, network devices logs, etc.
Database server logs, virtual machines, devices logs, etc.
IoT Attack surface Area
12:33
Internet
CLOUD
COMPUTING
DATA CENTER
Device Examples
Things e.g. smart appliances, medical devices, driver-less vehicles, industrial robots, smart TVs, readers, tags, etc
Gateways, routers, boundary services firewalls,etc
Gateways, routers, boundary services firewalls,etc
Attack Examples
physical damage, code injection , rootkit, tag cloning, social engineer, chip depackaging, outage attack, node jamming, etc.
routing attacks, Man-in-the-middle, Spoofing, sniffing, replay attack, denial of service, session hijacking, TCP SYN flooding, buffer overflow, eavesdropping, Sinkhole, Sybil, Wormhole, etc.
Ransomware, spyware, data loss, data leakage, Account hijacking, data manipulation, Brute force attack , Virtual Machine(VM) Escape, software attack, virus, worm, phishing, spyware, etc.
IoT architecture and layer attacks

Figure 4: IoT Architecture and Layer Attacks

In the network layer, IoT involves a variety of diverse networks, such as, WSNs, wireless mesh networks, WLAN, etc. These networks help sensors in IoT exchange information. A gateway can simplify the communication of several sensors over the network. Thus, a gateway could be beneficial to handle many complex aspects involved in communication on the network. The network layer ensures the successful transmission of data while the application layer is the highest layer that processes the data for visualization.

In the application layer, the data source can be obtained from Internet Service Provider (ISP) and mobile network providers' web-based services, virtual online identities, edge network, devices logs, Radio-Frequency Identification (RFID) tags, and readers, etc.

Most of the attackers' target IoT devices and equipment rather than a single PC. IoT has an interconnection of various devices and equipment along with some embedded devices as well. The major causes of IoT as a malware target can be summarized below:

- All the devices and equipment in an IoT need to be always on and it is easy for attackers to assess that equipment where the power mode is on at any point in time.
- Devices and equipment interconnected in an IoT are always connected and the attackers may access the interconnected devices from a single device.
- In most cases, proper security measures and knowledge to defend and tackle attack in a whole set of interconnected devices is difficult than in a single PC.
- Lack of proper encryption features in the interconnecting devices and weak passwords is another cause of malware target in IoT.
- The level of sophistication for the exploitation of the IoT is much lower and easy as compared to a single device.
- Twenty four hours of internet exposure of the IoT devices and equipment is another cause of IoT as a malware target. Due to the unlimited internet connection, the devices will accept the incoming traffic signals and become vulnerable to attacks.

In this paper, we propose five design principles to be considered when developing an effective and efficient deep learning IDS for IoT, and we use these principles to propose TCNN, a variant of CNN that uses causal convolutions. TCNN is combined with data balancing and efficient feature engineering. More specifically, the main contributions of the paper are the following:

(i) We identify five key design principles for the development of deep learning-based IDS for IoT, including handling overfitting, balancing dataset, feature engineering, model optimization, and testing on IoT dataset.

(ii) Based on the identified key design principles, we compare the state-of-the-art methods, identify their gaps, and analyze the main differences with respect to our work.

(iii) We design and implement Temporal Convolution Neural Network (TCNN), a deep learning framework for intrusion detection systems in IoT. TCNN combines Convolution Neural Network (CNN) with causal convolution.

(iv) To handle the issue of imbalanced dataset, we integrate TCNN with Synthetic Minority Oversampling Technique-Nominal Continuous (SMOTE-NC).

(v) We employ efficient feature engineering, which consists of the following:

  (1) Feature Space Reduction: It helps in reducing memory consumption.

  (2) Feature Transformation: It is applied on continuous numerical features using log transformation and standard scaler, which transforms skewed data to Gaussian-like distribution. It is also applied on categorical features using label-encoding, which replaces a categorical column with a unique integer value.

(vi) We evaluate the effectiveness and efficiency of the proposed TCNN on Bot-IoT dataset, and compare it with CNN, LSTM, logistic regression, random forest, and other state-of-the-art methods. The results show the superiority of TCNN in scoring an accuracy of 99.9986% for multiclass traffic detection.

8. Classes of IDS

IDS is classified in two main categories as follows:

Host-based IDS detects intrusion behavior by scanning log and audit records. This kind of IDS is usually used on important hosts to protect the host security from all directions. The advantage of the host-based IDS is that it provides more detailed information, lower false alarm rates, and has less complexity than network-based IDS. However, it reduces the efficiency of the application relies excessively on the log data and monitoring capability of the host.

Because of the characteristics of the IoT and because many IoT devices can be connected to the network, network-based IDSs need attention. Network-based IDS can detect the abnormal behavior and data flow in the network, to find potential intrusions. It does not change the host configuration and does not affect the performance of the business system. Even if the network IDS fail, it will not affect normal business operation. A problem is that network-based IDS only checks its direct connection to the network segment without looking at other network segments. It is also difficult to process encrypted sessions with network-based IDS.

Detection Methods: Based on the nature of various intrusion attacks, intrusion detection methods are classified into four major categories: Signature-based methods, Specification-based methods, Anomaly-based methods, and Hybrid methods (Figure 5).

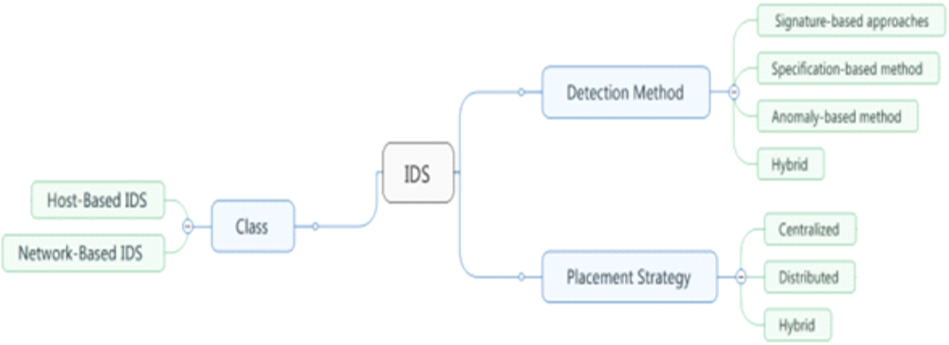

Figure 5: Classification of IDS

Signature-based methods first scan the data in the network and compare it with a feature database. If the scanned data is found to match the features in the signature database, the data will be treated as an intrusion. The advantage is that it can accurately determine the type of attack. It is relatively convenient to use, and the demand for resources is comparatively small. Specification-based methods require the system administrators to set rules and thresholds in advance. IDS detect the status of the current system and network according to the rules and thresholds set by administrators. If the threshold is exceeded or the rules are violated, the IDS will detect an abnormal situation and act accordingly.

Anomaly-based methods depend on identifying abnormal patterns and by comparing traffic patterns. The advantage of using this method is that it enables the detection of new and unknown intrusions. However, the primary limitation is that the method tends to result in high false positive rates. Research is now focusing on applying machine learning algorithms in anomaly-based intrusion detection methods to improve the robustness of this kind of method. By employing machine learning algorithms, anomaly-based intrusion detection methods can monitor the ongoing intrusion footprints and compare them with existing datasets to be alert to potential future attacks.

Hybrid methods refer to the use of any combination of the above-mentioned detection methods in the same IDS. This approach can help to overcome the shortcomings of a single method thereby enhancing the reliability of the entire IoT system. However, the obvious drawback is that the entire IDS will become very large and complex. This will make the whole system more difficult to operate and will require more resources. Especially when there are many protocols involved in the IoT system, the intrusion detection process will have large resources and time demands.

9. Key Design Principle for Deep Learning IDS in IoT

The objective of deep learning-based IDS solutions for IoT is to generate models that perform well in terms of effectiveness and efficiency. However, each model adopts some design choices that might limit its ability in achieving this objective. For example, some deep learning IDSs in IoT do not consider the overfitting problem, or apply their model on an unbalanced dataset, or neglect employing feature engineering, which negatively affects their performance in terms of accuracy, memory consumption, and computational time. Also, some IDSs do not try to optimize their learning model, and some are evaluated on outdated or irrelevant datasets, which do not reflect the real world IoT network traffic.

Motivated by the above observations, the deep learning-based IDS solution for IoT should advocate the following key design principle:

(i) Handling Overfitting: Overfitting happens when the model achieves a good fit on the training data, but it does not generalize well on unseen data. In deep learning, overfitting could be avoided by the following methods:

(1) Applying regularization, which adds a cost to the loss function of the model for large weights.

(2) Using dropout layers, which randomly remove certain features by setting them to 0.

(ii) Balancing Dataset: Data imbalance refers to a disproportion distribution of classes within a dataset. If a model is trained under an imbalanced dataset, it will become biased, i.e., it will favor the majority classes and fail to detect the minority classes. By balancing the dataset, the effectiveness of the model will be improved.

(iii) Feature Engineering: It allows reducing the cost of the deep learning workflow in terms of memory consumption and time. It also allows improving the accuracy of the model by discarding irrelevant features and applying feature transformation to improve the accuracy of the learning model.

(iv) Model Optimization: The objective of model optimization is to minimize a loss function, which computes the difference between the predicted output and the actual output. This is achieved by iteratively adjusting the weights of the model. By applying an optimization algorithm such as SGD and Adam, the effectiveness of the model will be improved.

(v) Testing on IoT Dataset: A deep learning-based IDS for IoT should be tested under an IoT dataset to get results that reflect real-world IoT traffic.

10. Proposed Framework

Deep learning is a concatenation of different layers. The first layer is called the input layer, and the last layer is called output layer. In addition, hidden layer are inserted between the input and output layers. Each layer is composed of a set of units, called neurons. The size of input layer depends on the dimension of the input data, whereas the output layer is composed of *C* units, which corresponds to the *C* classes of a classification task (Figure 6).

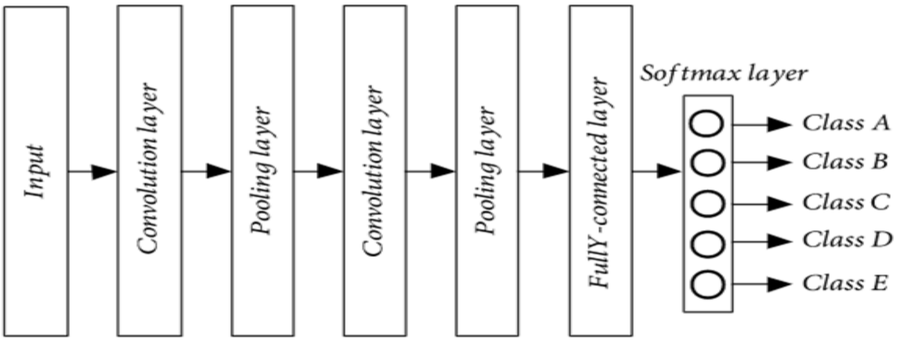

Figure 6: Framework of Different Layers

Convolutional Neural Network (CNN), as shown in Figure 1, is a deep neural network that is composed of multiple layers. The three main types of layers are the following:

(i) Convolutional Layer: It applies a set of filters, also known as convolutional kernels, on the input data. Each filter slides over the input data to produce a feature map. By stacking all the produced feature maps together, we get the final output of the convolution layer.

(ii) Pooling Layer: It operates over the feature maps to perform subsampling, which reduces the dimensionality of the feature maps. Average pooling and max pooling are the most common pooling methods.

(iii) Fully Connected Layer: It takes the output of the previous layers, and turns them into a single vector that can be an input for the next layer.

The proposed architecture is composed of the following phases:

(i) Dataset Balancing: As mentioned above, an imbalanced dataset can produce misleading results. To handle this problem, we use in this phase the SMOTE-NC method, which creates synthetic samples of minority classes and is capable of handling mixed dataset of categorical and continuous features.

(ii) First Feature Engineering (Feature Space Reduction): In this phase, we clean the dataset, i.e., reduce the feature space by removing unnecessary features, and converting the memory-consumption features into lower-size datatype.

(iii) Dataset Splitting: In this phase, the dataset is split into: training, validation, and testing subsets in order to counter overfitting.

(iv) Second Feature Engineering (Feature Transformation): In this phase, we apply feature transformation on the training subset. Log transformation and standard scaler are applied on the continuous numerical features. In addition, label encoding is applied to categorical features, which simply replaces each categorical column with a specific number. This transformation process is later applied on the validation and the testing subsets.

(iv) Training and Optimization: In this phase, the TCNN model is built, as described in Section. It is trained using the training subset, and its parameters are optimized using Adam optimizer and the validation subset.

(v) Classification: The generated TCNN model is applied on the testing subset to attribute each testing record to its actual class: normal or a specific category of attack.

11. Training and Optimization of TCNN Framework

The training and optimization phase of the proposed TCNN is composed of two 1D causal convolution layers, two dense layers, and a softmax layer, which applies softmax functions for multiclass classification task. To overcome overfitting, we use global maximum pooling, batch normalization, and dropout layers. We choose Adam optimizer to update weights and optimize cross-entropy loss function. Adam optimizer combines the advantages of two stochastic gradient descent algorithms, namely Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp).

Specifically, the training and optimization phase of the proposed TCNN architecture is composed of the following layers:

(i) First 1D Causal Convolution Layer: It convolves across the input vectors with 64 filters and filter size of 3.

(ii) Second 1D Causal Convolution Layer: It uses 128 filters and a filter size of 3. This second layer before pooling allows the model to learn more complex features.

(iii) 1D Global Maximum Pooling Layer: It replaces data, which is covered by the filter, with its maximum value. It prevents overfitting of the learned features by taking the maximum value.

(iv) Batch Normalization Layer: It normalizes the data coming from the previous layer before going to the next layer.

(v) Fully Connected Dense Layer: It employs 128 hidden units and a dropout ratio of 30%.

(vi) Fully Connected Dense Layer with Softmax Activation Function: It produces five units that correspond to the five categories of traffic for multiclass classification.

12. Implementation

To implement the detection learning models, we use Intel Quad-core i7-8550U processor with 8 GB RAM and 256 GB Hard drive. As for software, we use Python 3.6 programming language, and TensorFlow to build deep learning models. Moreover, different libraries are used including Scikit-learn, Keras API, Panda, and Inmblearn. We implement the framework in Figure 3 on Bot-IoT Dataset.

13. Bot-IoT Dataset

We use Bot-IoT and IoT dataset that was released in 2018 by the Cyber Center in the University of New South Wales. By virtualizing the setup of various smart home appliances including weather stations, smart fridges, motion-activated lights, remotely activated garage doors, and smart thermostats, legitimate and malicious traffic is generated. The dataset consists of more than 73,000,000 records, which are represented by 42 features. Each record is labeled either as normal or attack. In addition, the attack dataset is divided into four categories: DoS, DDoS, reconnaissance, and information theft, and each category is further divided into subcategories.

14. Dataset Balancing

In the dataset, there are 9,543 normal and 73,360,900 attack samples. The subset of the dataset is composed of 477 normal samples and 3,668,045 attack samples. We can notice that more than 97% of the samples belong to DoS and DDoS categories. In this way, the learning model will predict the majority classes and fail to spot the minority classes, which means the model is biased.

To deal with this problem, different resampling methods have been proposed like (1) random oversampling, which randomly replicates the exact samples of the minority classes; and (2) oversampling by creating synthetic samples of minority classes using techniques such as Synthetic Minority Oversampling Technique (SMOTE), Synthetic Minority Oversampling Technique for Nominal and Continuous (SMOTE-NC), and Adaptive Synthetic (ADASYN). In this work, we use the SMOTE-NC technique as it is capable of handling mixed dataset of categorical and continuous features. The minority classes such as normal and theft are increased to 100,0000 samples in the training subset.

15. Feature Space Reduction

One of the main objective of this work is to develop a lightweight IDS for IoT environment. Therefore, it is important to improve the efficiency of the detection models by reducing the feature space and noise in the dataset, as well as reducing the memory usage and computation complexity. By using the full set of features, 2.9 GB of memory is used. Feature space reduction decreases the processing complexity and speeds up the training and detection processes. The following steps are applied to the dataset, which successfully decrease the memory consumption to 668 MB, i.e., 77% reduction.

(i) Conversion of object data type into categorical data type there are nine memory-consuming features that are encoded as objects, which are “flgs,” “proto,” “saddr,” “sport,” “daddr,” “dport,” “state,” “category,” and “subcategory.” As category datatype is more efficient, object features are converted into category datatype. .

(ii) Conversion of Int64 Data Type into Int32 Data Type: By default, the 22 integer features in the dataset are stored as Int64 (8-bytes) type. After checking these features, we find out that they do not exceed the capacity of Int32 (4-bytes) type. Therefore, all the values of Int64 type are encoded into Int32 type, which incurs half of the memory consumption that is incurred by the Int64 type.

(iii) Removing Unnecessary Features: In the dataset, we exclude some useless features such as the following: (1) “pkSeqID”: It has the same role as the automatically generated index; (2) “stime” and “ltime”: they are captured in the “dur” feature, which computes the duration between “stime” and “ltime”.

16. Dataset Splitting

Conventional splitting and cross-validation are the main approaches used to split datasets. Cross-validation is mainly used in legacy machine learning to overcome the overfitting problem. When a large dataset is used with deep learning, cross-validation increases the training cost. In this work, the dataset is split using the conventional three-way split into: training, validation, and testing subsets. In addition, regularization is applied to deal with the overfitting if it appears also, a stratified split is used to ensure that there is a portion of each class in each split.

TCNN has been evaluated on Bot-IoT dataset and compared with logistic regression, random forest, LSTM, and CNN. Evaluation results show that TCNN achieves a good trade-off between effectiveness and efficiency. It outperforms the state-of-the-art deep learning IDS methods, which were tested under Bot-IoT dataset, by recording an accuracy of 99.9986% for multiclass traffic detection. Also, it shows a very close performance to CNN with respect to training time. As part of future work, it would be interesting to consider another design principle, i.e., testing the resiliency of IDS against adversarial attacks, which can confuse the deep learning model to produce wrong predictions. Interesting to consider another design principle, i.e., testing the resiliency of IDS against adversarial attacks, which can confuse the deep learning model to produce wrong predictions.

References

Alexander, D. (2020). *Neural Networks: History and Applications*, 1st Edition, pp. 6-7, Nova Science Publishers Inc.

Alur, R., Berger, E., Drobnis, A.W., Fix, L., Fu, K., Hager, G.D., Lopresti, D., Nahrstedt, K., Mynatt, E., Patel, S., Rexford, J., Stankovic, J.A. and Zorn, B. (2015). Systems Computing Challenges in the Internet of Things, *Computing Community Consortium*, published.

Derhab, A., Belaoued, M., Guerroumi, M. and Khan, F.A. (2020). Two-Factor Mutual Authentication Offloading for Mobile Cloud Computing. *IEEE Access*, 8, 28956-28969.

Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*, MIT Press.

Khraisat, Ansam. and Alazab, Ammar. (2021). A Critical Review Of Intrusion Detection Systems in the Internet of Things: Techniques, Deployment Strategy, Validation Strategy, Attacks, Public Datasets and Challenges. *Cybersecurity*, 4, 18. DOI: https://www.doi.org/10.1186/s42400-021-00077-7

Leading the IoT - Gartner Insights on How to Lead in a Connected World - eBook - Mark Hung, Gartner Research Vice President – 2017.

Lopez-Martin, M., Carro, B., Sanchez-Esguevillas, A. and Lloret, J. (2017). Network Traffic Classifier with Convolutional and Recurrent Neural Networks for Internet of Things. *IEEE Access*, 5, 18042-18050.

Ni, S., Qian, Q. and Zhang, R. (2018). Malware Identification Using Visualization Images and Deep Learning. *Computers & Security*, 77, 871-885.

Rehman, E. *et al*. (2022). Intrusion Detection Based on Machine Learning in the Internet of Things, Attacks and Counter Measures. *The Journal of Supercomputing*. Available: 10.1007/s11227-021-04188-3.

Thamilarasu, G. and Chawla, S. (2019). Towards Deep-Learning-Driven Intrusion Detection for the Internet of Things. *Sensors*, 19(9), article 1977.

Wang, X., Ning, Z., Guo, S. and Wang, L. (2020). Imitation Learning Enabled Task Scheduling for Online Vehicular Edge Computing. *IEEE Transactions on Mobile Computing*.

Warsito, B., Santoso, R., Suparti and Yasin, H. (2018). Cascade Forward Neural Network for Time Series Prediction. *Journal of Physics: Conference Series*, 1025, 012097. Available: 10.1088/1742-6596/1025/1/012097.

White paper on "Cisco IoT System Security: Mitigate Risk, Simplify Compliance, and Build Trust" Available at http://www.cisco.com/c/en/us/solutions/internet-of-things/resources/case-studies.html

Zanella, A., Bui, N., Castellani, A., Vangelista, L. and Zorzi, M. (2014). Internet of Things for Smart Cities. *IEEE Internet Things*.

Zhu, A., Gu, Z., Hu, C., Niu, J., Xu, C. and Li, Z. (2021). Political Optimizer With Interpolation Strategy for Global Optimization. PLOS ONE, 16(5), e0251204. Available: 10.1371/journal.pone.0251204

Cite this article as: Abeer, Y.A. Salawi. and Mohammed, I. Alghamdi. (2022). Optimization Model for Intrusion Detection System in IoT Applications. *International Journal of Cryptocurrency Research*, 2(1), 41-51. doi: 10.51483/IJCCR.2.1.2022.41-51.