Enhanced Network Anomaly Detection using Convolutional Neural Networks in Cybersecurity Operations

Khaled Bin Showkot Tanim
Department of Electrical and Computer Engineering (ECE), North South University
Dhaka, Bangladesh

Mahadi Hasam Parash
Department of Computer Science & Engineering (CSE), Jashore University of Science & Technology
Jashore, Bangladesh

MD Shadman Soumik
Department of Information Technology (MSIT), Washington University of Science and Technology (WUST)
2900 Eisenhower Ave, Alexandria, VA 22314

Mohammed Shakib
Department of Computer Science
American International University - Bangladesh
Dhaka, Bangladesh

ABSTRACT
Network anomaly detection is critical for preserving cybersecurity and safeguarding sensitive data. Traditional approaches sometimes struggle with the complexity and amount of current network traffic. This research provides an upgraded network anomaly detection method utilizing convolutional neural networks (CNNs). Leveraging the BoT-IoT dataset, this paper utilize feature selection strategies based on entropy and correlation to develop a robust CNN feature matrix. The model showed considerable gains in identifying abnormalities, with a high accuracy rate of 96%. The application of the system in both offline and online modes illustrates its relevance in real-world cybersecurity operations. Detailed assessments, including training and testing timeframes, indicate the system's efficiency and efficacy. Future work will concentrate on increasing the dataset, incorporating additional deep learning models, and boosting real-time detection capabilities.

General Terms
Pattern Recognition, Machine Learning, Network Security, Deep Learning Algorithms, Data Analysis, Evaluation Metrics

Keywords
Network anomaly detection, cybersecurity, convolutional neural networks, BoT-IoT dataset, feature selection, real-time detection, and deep learning models.

1. INTRODUCTION
There can be no debate that effective network security is important in a connected, technologically-dependent modern society. Corporate security is always susceptible to many hazards, such as malware, phishing, and other advanced attempts to breach the protected networks. Network abnormality detection looks for any abnormality or irregularity that is inconsistent with the protocols specified by the network standards in order to identify these threats [2].

The main elements of the typical methods used in anomaly detection, which are designed by humans, include statistical analysis and feature extraction. However, these methods may not be adequate to address the growing complexity of modern threats [6]. With the increasing popularity of deep learning techniques, especially convolutional neural networks (CNNs), significant advancements have been observed in recent decades. This encompasses the field of computer security [3].

This research also delineates certain advantages that arise from the utilization of CNNs in network anomaly identification, which encompass the following: Convolutional neural networks (CNNs) can be employed to acquire intricate characteristics from the unprocessed data of network traffic, thereby enhancing the identification of hidden and previously unnoticed abnormal activities inside a network [4]. Furthermore, they also demonstrate the ability to handle increasing workloads and adaptability, making them well-suited for immediate identification in extensive network systems [5].

However, there are several disadvantages associated with CNNs when it comes to network anomaly detection, such as the demand for an enormously labeled data set, the challenge of the interpretability of the results, and the high processing overhead accordingly [6]. In order to address these problems, it is necessary to develop novel solutions that utilize CNN-based anomaly detection systems and conduct thorough empirical research to identify the potential strengths and weaknesses of such systems [7].

Past research has focused on developing convolutional neural networks (CNNs) for detecting network anomalies. However, previous studies have not extensively explored the application of CNNs in recognizing and mitigating complex and ever-changing cyber threats. The current study primarily focuses on isolated components of CNN systems or tests conducted in controlled environments, limiting the applicability of the findings to real-world cybersecurity organizations [1]. The main aims of this investigation are: The main objectives of this study are:

- For examining the effectiveness of various convolutional neural network (CNN) implementations in the detection of network

abnormalities under multiple and shifting network situations.

- In order to compare the findings attainable by CNN-based anomaly detection systems with those gained by applying real-world network traffic datasets and existing methodologies.
- To detail the distinctive properties of the CNN-based anomaly detection systems and then examine the possibility of boosting their performance and scalability.

This research makes the following novel contributions:

- It covers a variety of empirical evidence on the usefulness of CNNs in network anomaly detection relative to the heterogeneity of network traffic patterns and the different forms of cyber threats.
- A complete exploration of the usefulness and efficiency of the CNN-based anomaly detection system, based on numerous settings and multiple criteria, including validation metrics.
- Some valuable recommendations and guidelines to boost the efficiency and efficacy of CNN based abnormal behavior detection systems in a real-world cybersecurity environment.

The remainder of this paper is organized as follows: 1. section 2: literature review 2. Network Anomaly Detection: In this area of research, past studies have focused on network anomaly detection and convolutional neural networks (CNNs). In Section 3, the details of the deep learning technique are discussed, such as the data gathering process, CNN architecture, and the experimental protocol. Section four demonstrates the empirical assessment and performance comparisons of the work. Final Part 5 provides a summary of the present paper, effective cybersecurity advice, and future study proposals.

2. LITERATURE REVIEW

All material on each page should fit within a rectangle of 18 x 23.5 cm (7" x 9.25"), centered on the page, beginning 2.54 cm (1") from the top of the page and ending with 2.54 cm (1") from the bottom. The right and left margins should be 1.9 cm (.75"). The text should be in two 8.45 cm (3.33") columns with a .83 cm (.33") gutter. Anomaly detection is one of the main subtopics within the cybersecurity process; its primary purpose is to recognize patterns in the network traffic that differ from usual patterns, warning of possible malicious activity or system faults. It would be desirable to present a review of the state of the art in this field and assess various existing approaches and technologies in order to understand the dynamics, difficulties, and prospects of building anomaly detection systems.

Moustafa et al. [8] reviewed and surveyed, at a system level, the methodologies and algorithms of NAS and the correct ways of evaluating them. Their work may be considered to present a general overview of the present state of affairs in the field of anatomical modeling and can help outline new possibilities for improving the current approach and the development of innovations.

Cummings et al. [8] covered the main features of the network anomaly detection survey, including the detection methodologies, dataset, and performance measures employed as indicated by Fernandes et al. [9]. Through highlighting the usage and usefulness of the prior literature on the subject, their study contributes to decision making on the construction and implementation of anomaly detection systems.

Nassif et al. [10] evaluated and provided a systematic approach to machine learning techniques for anomaly detection and contrasted several ways of learning on four classes of algorithms for network anomaly detection. This gives insights into how well each of the machine learning algorithms that they utilized performed and offers insights into recent trends in the development of this subject of anomaly identification.

Bhuyan et al. [11] offered a detailed description of several network anomaly detection approaches, applications, and some of their systems and tools, with a deep analysis of their working principles and implementation difficulties and obstacles on large networks. From their survey, they have supplied a complete source for scholars and practitioners who want to obtain a grasp of the technologies that can be applied for anomaly detection.

Yang et al. [12] reviewed the systematic literature on methodologies and datasets for anomaly-based network intrusion detection and identified the most essential characteristics of the datasets and methods for evaluation. This is an essential issue because their study offers a thorough knowledge of the prospects in terms of choosing relevant datasets for training and testing anomaly detection systems.

Patch and Park [13] published a revised study on the methodologies for anomaly identification since they also included current solutions and technological breakthroughs on the topic. A practical evaluation of the available works in the domain of anomaly detection is offered by their study, including a comparison of the statistical methodologies and machine learning techniques.

A multi-perspective review of anomaly detection in sensor systems was published by Erhan et al. [14], where the authors pointed out that the major concern of sensor networks is that they provide sizably substantial requirements for intelligent anomaly detection algorithms. Their work covers numerous tactics and approaches to assessing abnormalities within the context of sensors and efforts towards the advancement of sensor based anomaly detection systems.

In the work of Ali et al. [15], the authors have presented a classification of machine learning-based anomaly detection algorithms in network data with an emphasis on particular current trends. I think that the study of D. M. Allen et al. can offer some insight into the possibilities and difficulties of utilizing machine learning approaches to tackle the obstacles linked with network anomaly detection.

Similarly, recent works by Bodström and Hämäläinen [16] give a literature analysis on network anomaly detection using a deep learning approach, whereas the approaches applied are deep learning techniques like CNNs and RNNs. Ming, L. , Liu, C. , & Zhang, Z. highlight the usefulness of deep learning algorithms in increasing the performance of anomaly detection systems.

A recent publication by Haji and Ameen [17] explored attack and anomaly detection in IoT networks, with an emphasis on machine learning, as a review of the present literature and the expansion of IoT devices and networks. Their study gives an idea of how the machine learning method might also be utilized to identify present threats as well as future and thus unknown anomalous activity in the IoT.

Unlike surveys, Fahim and Sillitti [18] examined anomaly detection, analysis, and prediction approaches for the IoT

context, with special emphasis on the potential and difficulties connected to IoT technology, especially IOT devices. An article by Laleh et al. provides advice on how to construct effective anomaly detection systems in the context of IoT. File. **Table 1** illustrate the Summary of Machine Learning in Cybersecurity and Convolutional Neural Networks for Anomaly Detection.

Table 1. Summary of Machine Learning in Cybersecurity and Convolutional Neural Networks for Anomaly Detection

Author	Methodology	Algorithm	Finding
Ford & Siraj (19)	Literature review	Various machine learning techniques	Discusses applications of machine learning in cybersecurity, including threat detection and malware analysis.
Martínez Torres et al. (20)	Literature review	Various machine learning techniques	Provides insights into the application of machine learning techniques in cybersecurity, emphasizing their potential to enhance security mechanisms.
Handa et al. (21)	Literature review	Various machine learning techniques	Offers a comprehensive review of machine learning in cybersecurity, discussing applications in anomaly detection and security analytics.
Nassif et al. (10)	Systematic review	Machine learning techniques	Reviews machine learning techniques for anomaly detection, highlighting their effectiveness in identifying network anomalies.
Yang et al. (12)	Systematic literature review	Machine learning techniques	Reviews methods and datasets for anomaly-based network intrusion detection, providing insights into available resources for research.
Shaukat et al. (23)	Performance comparison	Machine learning techniques	Compares the performance of different machine learning algorithms in cybersecurity and identifies current challenges in their application.
Bian et al. (27)	Experimental study	Convolutional Neural Networks (CNNs)	Proposes a CNN-based anomaly detection network for utility tunnel fire protection, demonstrating the effectiveness of CNNs in critical infrastructure.
Tang et al. (28)	Experimental study	Convolutional Neural Networks (CNNs)	Develops a CNN-based data anomaly detection method for structural health monitoring, showcasing the applicability of CNNs in sensor data analysis.
Caliva et al. (29)	Experimental study	Deep learning (including CNNs)	Applies deep learning, including CNNs, for anomaly detection in nuclear reactors, highlighting their importance

			in ensuring safety and security.
Yin et al. (35)	Experimental study	Convolutional Recurrent Autoencoder (CRAE)	Proposes an anomaly detection method based on CRAE for IoT time series data, showcasing the effectiveness of deep learning in IoT applications.

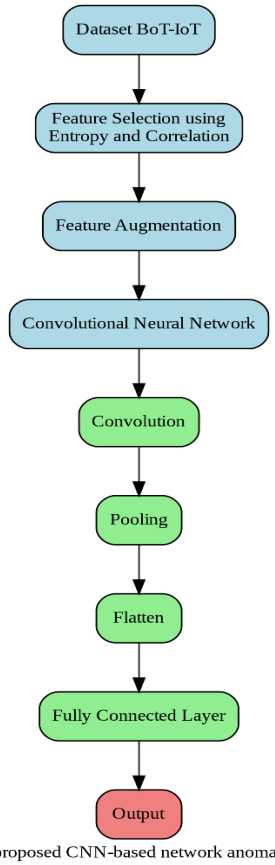
As we have shown in this chapter, big data analytics utilizing machine learning algorithms makes it possible to examine enormous volumes of data and uncover suspect behavioral patterns [17, 19]. In their study, Ford and Siraj [19] also explored how machine learning works within security systems as well as its usage in threat identification, malware analysis, and intrusion detection. The work by Martinez Torres et al. [20] presented insight into the importance of machine learning in cybersecurity; greater focus was put on how machine learning might improve security operations. Further, Handa et al. [21] include a synthetic assessment of machine learning in cybersecurity, focusing on anomaly detection, threat intelligence, and security analytics.

Potent deep learning algorithms in the security domain include convolutional neural networks (CNNs) that have shown preference in the identification of anomalies from raw data [28]. In the situation of utility tunnel fire protection, Bian et al. [27] created a CNN-based anomaly detection network to detect fire protection and confirmed that CNN is efficient to detect abnormalities in the infrastructure. Similarly, Tang et al. [28] have given a CNN based data anomaly detection approach for structural health monitoring as a nice example showcasing the usage of CNNs for detecting anomalies within different data sets. Similarly, Caliva et al. [29] employed deep learning, primarily CNNs, on a real-time basis to monitor the nuclear reactor and detect any anomalies linked to the system, stressing the necessity of CNNs in safeguarding essential facilities and infrastructure.

A few current research studies have attempted to study the effectiveness of convolutional neural networks (CNNs) in network anomaly detection and prove the possibility of enhancing detection quality [232]. An upgraded network anomaly detection strategy was created employing deep neural networks, notably CNNs, by Naseer et al. [33] in order to offset the shortcomings of the old methods and recognize tiny abnormalities in the traffic pattern. Rezaee et al. [34] included a survey for deep learning-based real-time crowd anomaly detection by deploying CNNs to ensure distributed real-time video surveillance systems. Moreover, Yin and Chen [12] suggested an anomaly detection model with a deep convolutional recurrent autoencoders (CRAE) for IoT time series data, and there are many success examples where CNNs have functioned successfully as an anomaly detection tool in many applications.

3. METHODOLOGY

In this paper, the author(s) present an effective technique for enhancing network anomaly detection based on CNN architectures. The proposed framework illustrated in Figure 1 covers various key aspects to incorporate the capabilities of efficient and accurate network anomaly detection. First, convolutional layers are used for processing network traffic data to extract detailed features from it, and then, after that, the details of the features are minimized using the pooling layers without losing much useful information. These features are then flattened and injected through certain fully connected layers in order to acquire the output that differentiates normal traffic from attack traffic. The dataset on which the feature extraction is based and that is used for training and testing the model is the BoT-IoT, which is preprocessed through normalization and scaling. Feature selection involves entropy and correlation measures, which restrict the number of features to the most important ones. In the offline phase, the building and training of the CNN model are performed, and it goes through the normal and attack traffic patterns. In the online phase, data is harvested from the IoT sensors in real-time and, thereafter, gathered by an event collector before proceeding through the feature augmentation and feature extraction stages. This provides a feature matrix that is subsequently subscribed to the CNN model to perform real time anomaly detection, whereby any discovered irregularities pass through the gateway and go to security replies. This technique of working guarantees a good and effective system in relation to security against numerous sorts of attacks in the network. **Figure 1** dhow the Framework of the proposed CNN-based network anomaly detection system.



Framework of the proposed CNN-based network anomaly detection system.

Fig 1: Framework of the proposed CNN-based network anomaly detection system.

3.1 DATASET

With remarkable precision, the BoT-IoT within Cyber Range Lab, UNSW Canberra, was designed to be a reasonably large collection of network traffic data, offering a broad classification of both normal and attack traffic types. It features an exhaustive taxonomy of assaults: DDoS, DoS, OS and Service Scan, Key logging, and Data Exfiltration, which is a clear indicator that there are several challenging sorts of cyber-attacks. However, it can be observed that under this classification, distinctions have been made between DDoS and DoS attacks depending on the underlying protocols, which adds some complexity to the threat differentiation process. The raw dataset is initially recorded in the form of cap files and comprises 69 megabytes in weight. Partly in its raw, actively updated generalized version, the flow traffic as a dataset occupies 3 GB, exceeding 72 million records; the extracted flow traffic in CSV format is greatly lowered to 16.7 GB. With regard to usability, a 5% subset of around ⅕ of image postings should be picked for improvement. This was collected by MySQL queries, and it was approximately 7 GB in size and entailed about three million rows. This dataset is vital for training and evaluation of anomaly detection systems; it supplies the necessary backdrop for studying network-based dangers and fortifying cybersecurity in the expanding digitally inclined environment.

3.2 Feature Augmentation

In the suggested architecture, feature enhancement is very significant in that it augments the input data feeds to create an optimal CNN based network anomaly detection system. Certain approaches, like normalizing, scaling, and performing mathematical operations on the characteristics, help in developing extra valuable features in the dataset. Normalization is utilized in an attempt to bring the feature values in a dataset to a similar scale, thereby permitting evaluation of feature relevance in the training phase without bias from variation in scale. Scaling takes the inputs and maps the feature values to a pre-defined range, which promotes convergence during the times of model training as well as optimizes the speed at which gradient descent is executed. Further, operations like logarithmic transformation, exponential transformation, or polynomial transformation are employed to increase more comprehensive qualities between the variables and deduce the underlying non-linear characteristics from the given data. Generally, this holistic enrichment of the features gives the CNN model the possibility to be trained by numerous features, which are informative, and consequently, the CNN model gets a sensitivity to discern between minor fluctuations in traffic data.

3.3 Feature Selection using Entropy and Correlation

For the purpose of leveraging the efficacy of a CNN-based network anomaly detection framework, in the study entitled "Feature Selection using Entropy and Correlation," we examined mutual information (MI)-based feature selection followed by the Pearson correlation coefficient. \ (^ {82} \) these strategies were utilized since they exhibited good outcomes when coping with high dimensional data sets since they retained discriminative capabilities and computational tractability. The importance of characteristics carrying significant information concerning anomalous behavior in the network can be identified by utilizing a statistic called mutual information, which evaluates the dependency between the variables. Based on the values of the feature set, a statistic called the Pearson correlation coefficient is used to measure the linear relationship between the features, and any redundant features are ejected. This strategic integration also becomes a guarantee for the maintenance of significant informative characteristics while simplifying the architecture of the model to provide clearer interpretability and better computational qualities for the cyber defense of complex modern systems in the constantly changing threat environment.

3.4 Convolutional Neural Network

For pages other than the first page, start at the top of the page, and continue in double-column format. The two columns on the last page should be as close to equal length as possible. CNNs are the basic building blocks in the proposed anomaly detection framework, given the potential of these networks to learn representations at several levels of abstraction from the input data. The CNN architecture framework comprises fundamentally numerous core layers that all work concurrently, despite their distinct duties in interpreting the flow of network traffic.

Convolutional layers are crucial to embracing CNNs that extract patterns from input data by applying a suite of filters or kernels. These filters move over the input data and apply SE convolutions of element-wise multiplication and summing to build the feature maps that represent spatial pyramids. In this framework, we use filters of size either 3x3 or 5x5, and fundamentally, the stride is always 1 and the padding is always 0, assuring equal advantageous features on the spatial axis. The convolution method preserves sophisticated local distortions, allowing the network to recognize minor abnormalities within traffic flow data. Table 2 show the cnn architecture for this study.

Table 2. CNN-Architecture

Layer Type	Description	Parameters
Convolution	Applies filters to input data, creating feature maps that capture spatial hierarchies	Filter Size: 3x3 or 5x5 Stride: 1 Padding: Zero
Pooling	Reduces dimensionality of feature maps while preserving essential information	Pooling Type: Max pooling Pooling Size: 2x2
Flatten	Transforms pooled feature maps into a one-dimensional vector for fully connected layers	

Fully Connected	Synthesizes high-level features learned by preceding layers to make final predictions	Number of Neurons: Variable Activation Function: ReLU

Output	Provides final prediction in a probabilistic format suitable for interpretation	Activation Function: Softmax

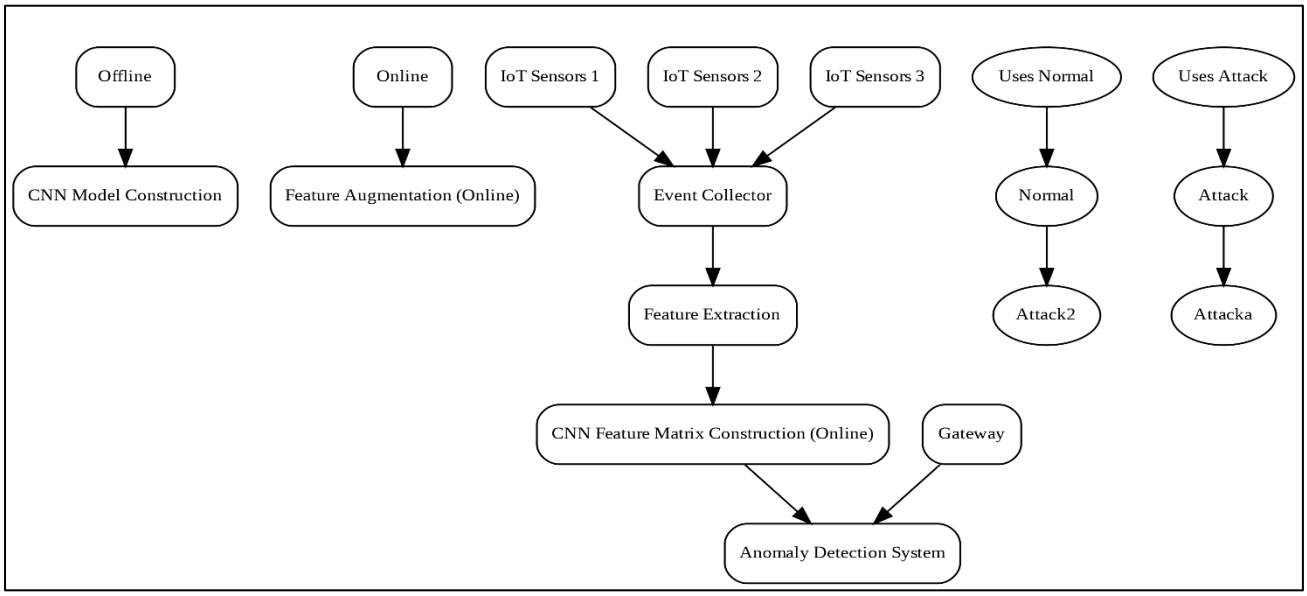


Fig 2: CNN Feature Matrix Construction

It is used to down sample the feature maps that are created from the convolutional layer, where a reduction in the spatial size of the maps is necessary while maintaining critical data. Among the different pooling methods, max pooling is the approach that has been commonly utilized, where the max value in the given region of the map is selected and other low value features are removed, which lowers the complexity of the calculations. The pooling layers are employed as one technique for lowering the amount of overfitting and as a way of helping the model gain more translational invariance, whereby the features learned are averaged or summed across space.

The flatten layer acts as an interface from the convolutional layers to the fully connected layers and turns the multidimensional feature maps into one-dimensional vectors. The flattening process helps to get the data via the current holders in succeeding completely connected layers, allowing for the integration of spatial and temporal properties specified by earlier layers. The flatten layer is useful in avoiding some of the problems that are likely to develop when advancing through the classification phase of the network since it condenses the hierarchical representations into one single vector.

The last stages of the model, or fully connected layers, sometimes called dense layers, are aimed at combining high-level features that have been learned by previous layers. The neurons in these layers are coupled in such a way that they can convey numerous characteristics of the input information as well as an interaction between them. Regularization techniques, including dropout, are employed to prevent overfitting of the discovered features by the network and ensure the generalization capabilities of the model. Highly linked layers play an important role in the development of decision making for the entire network by perceiving hierarchical representation to discern between normal and abnormal behavior of the network.

The final layer completes the computing operations of the network, resulting in the final prediction, which is supplied in a usable form. As for the classifier component, you may recall that we employ the softmax activation function at the output layer, which allows for probabilistic estimation. This activation function provides the probability difference between the two classes (normal or an attack) and consequently makes reliable categorization of the classes using the highest probability class. The output layer thus follows a combination of the feature extraction hierarchy coupled with decisions made by the network, enabling effective identification of abnormalities. **Figure 2 and 3** CNN Feature Matrix Construction and Architecture of implemented deep convolutional neural network model for Anomaly Detection.

Feature selection and engineering are crucial elements of any machine learning technique, and when generating the feature matrix for the CNN model, we very carefully perform the process of feature selection to aid in the detection of network anomalies. This procedure clearly differentiates between the offline phase and the online phase, during which experiences can differ in terms of data processing and model evaluation.

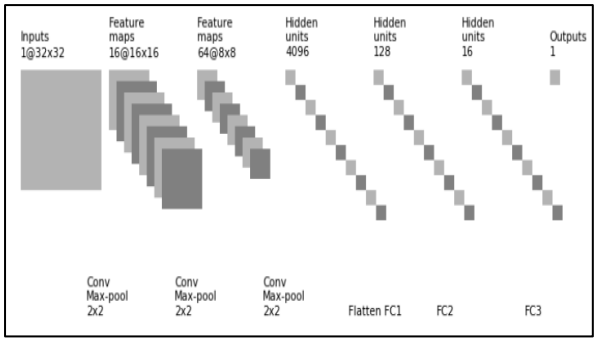


Fig 3: Architecture of implemented deep convolutional neural network model for Anomaly Detection.

Figure 2 demonstrates the process of producing the feature matrix in terms of the layers in the provided CNN architecture. It provides a flowchart like depiction of convolutional, pooling, flattening, fully connected, and output layers, and thus provides a clear grasp of how the raw input is processed to extract information necessary to complete the task of anomaly detection.

3.5 CNN Model Construction

The CNN model is trained based on the BoT-IoT framework, which is a large database that contains both normal and attack traffic patterns. The first section, or training data set, is then divided into sub sections that are used in training the model, while the second section, or validation data set, and is designed for evaluating the performance of the trained model. In the CNN model training process, numerous epochs are completed, through which it applies optimization methods such as stochastic gradient descent to alter the parameters of the model in order to boost performance.

The model can be trained where regular traffic flow is actively observed, and the model is to differentiate between normal traffic, which comprises lawful traffic, as opposed to malicious traffic. The model does not need to survey the entire network once it begins to evaluate benign data, which helps it understand routine network activity, allowing it to flag signals of danger.

At the same time, the model is capable of distinguishing several types of attack traffic, from typical DDoS to data filtration traffic. As the model obtains exposure to various attack scenarios, it is able to recognize multiple forms of harmful actions, which enables it to learn how to detect and classify different anomalies.

3.5.1 IoT Sensors

During the online phase of the elastic defense, IoT sensors operate as the first line of defense by constantly, and in real time, evaluating the traffic within a network. These sensors record streams of events emanating from network devices so that a continuous flow of information is made available for analysis and identification of aberrant events.

3.5.2 Event Collector

The event collector is consequently a vital aspect of IoT systems since it is responsible for the collection of data from numerous IoT sensors. Through synthesizing data from numerous sources, which the event collector receives and consolidates, significant coverage and higher accuracy in terms of data analysis are achieved. This gathered data constitutes the basis for real-time detection of abnormal information.

3.5.3 Feature Augmentation

At this stage, the collected raw data is processed for feature augmentation in real-time to boost its probability of being recognized by the CNN model. Scalars like normalizing and scaling make the data values more consistent and easier to interpret across the datasets, thereby boosting the performance of the model.

3.5.4 Feature Extraction

After that augmentation, feature selection enters the scene to locate features that fulfill the criteria of selection from the presented data. This should serve to filter out the procedure so as to just capture the raw properties of the network that are most relevant to the occurrence of an abnormality.

3.5.5 CNN Feature Matrix Construction

The retrieved features are then placed in a feature matrix of the format that suits the CNN input of rows and columns. This matrix supplies the feed-forward input for the CNN, which contains the essential topology of the network to detect the level of anomaly.

3.6 Anomaly Detection System

The generated feature matrix is subsequently considered in the CNN model for real-time anomaly detection. Using the learned patterns from the offline training phase, the model then examines the subsequent data stream, indicating tendencies that depart from the typical, which could be suggestive of a potential attack. The detected problem results in adequate defensive actions being performed when there are anomalies in the network, thus averting destructive invasions. **Figure 4** show the Anomaly Detection Flowchart.

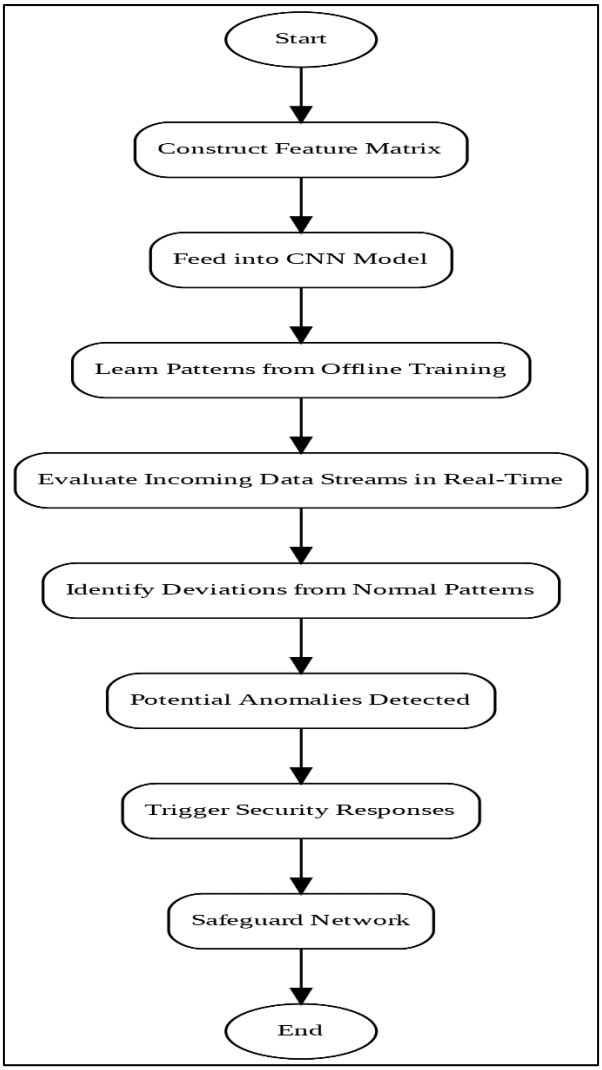


Fig 4: Anomaly Detection Flowchart

Information on the abnormalities is given to the security response via the gateway component. This strategic portion performs a crucial role as a coordination center and conducts the control measures in connection with the identified risks. Thus, with the supplied ability to carry out fast and directional interventions, the gateway supports offering a chance to reply to prospective risks as soon as feasible, thus preventing the negative consequences and maintaining the designated network environment.

Therefore, attack types, including Attack2 and Attack, are separate classifications of the malicious behaviors in the dataset BoT-IoT. There are various distinct forms of assault that are covered by these categories, including complicated Distributed Denial of Service (DDoS) strikes and stealthy data theft. The reliance on these attack categories is true, as all the attacks may occur in the real environment, and the performance and characteristics of the anomaly detection offered may be evaluated and tweaked within a variety of probable adversarial scenarios.

4. EXPERIMENTAL SETUP

The hardware arrangement for implementing the suggested models was created on an Intel Xeon E-1650 Quad Core CPU with 16 GB of RAM and also employing an NVidia GTX 1070 GPU with 1920 CUDA cores running the Cuda Version 8. On the software environment, all tests were done on the Ubuntu 18.04 LTS operating system. The Jupyter Notebook development environment was selected and employed Keras 2.0 deep learning framework with the TensorFlow backend operating behind it all. It was integrated with machine learning tools from Scikit-learn and includes Matplotlib and Seaborn for displaying results.

Before their CNN learning, the dataset BoT-IoT needs to be preprocessed. The preprocessing processes included normalization, scaling, and quantization, which is the act of transforming qualitative predictor variables into quantitative variables for future data analysis. In detail, the data formatting step takes in raw network traffic as an ordered collection of network packets denoted by ϕ, where a packet is described based on its timestamp and using features such as source address (SrcIP), destination IP address (DstlP), source port (SrcPort), destination port (DstPort), protocol, and TCP flags. Feature Selection: After formatting the data, we have undertaken feature selection utilizing entropy-based measurements and correlation analysis to maintain the important characteristics. Then we utilized symbolic feature encoding to encode nominal characteristics like protocol type, service, and flag via techniques such as one-hot encoding. This step made the model quicker. Additionally, min-max normalization (covering the range [0, 1]) was employed to scale normal data for CNN model input and aid in speeding up convergence during model training. The dataset was unbalanced using Synthetic Minority Oversampling Technique (SMOTE) for the classes so that underrepresented classes might have greater effect.

CNN model was done in a properly organized way. Split the dataset: The dataset was separated between 80% training and 20% validation data. In CNN architecture, all the layers have a have a convolution layer for feature extraction, a pooling layer to minimize dimensions, a flatten layer to transform input into a 1D array, and fully connected layers used in final prediction. Filters were used in the convolutional layers over all input data to learn spatial hierarchies, using either 3x3 or 5x5 filter sizes with stride of length one and zero padding. Dimensionality and critical information have been retained via 2x coverage of max pooling procedures. The model uses ReLU activation functions for the fully connected layers, with Softmax at the final output layer to make probabilistic predictions. The model was optimized using the Adam optimizer throughout training time. The AdaGrad and RMSProp techniques strengths were integrated into a new one.') We adjusted the learning rate to 0.001 and tried three activation functions: ReLU, ELU, and Tanh. Training (on a batch of N=32 sequences) was carried out with 'txt2uni' generated protein sequence data normalized using the mask token embedding's created for each sub-part, tokenizeAsMatrix_fromSequence; model evaluation performance on accuracy and precision, recall, and F1-score using the validation dataset.

The online stage of an anomaly detection system involved the process of data collection and processing in real time using IoT sensors. IoT sensors were utilized to monitor network traffic in real-time, with the data being provided for transmission via an event collector. That enabled them to centrally gather data from multiple sources—a crucial feature in allowing both efficient aggregation of the data and real whole-network coverage. The input data was normalized and scaled for feature augmentation before processing into the CNN model using real-time approaches. The most relevant characteristics were picked using feature extraction methods and grouped in a matrix, which was feasible for CNN analysis. This was the matrix that I would input into my CNN model: etc., which would then

examine telemetry data streams and, if detected abnormalities, raise warnings like some form of threat or attack. The gateway then sent the discovered abnormalities to a component called Gatekeeper, which triggers security mechanisms in order to manage against these dangers.

To validate each suggested model, we evaluated them with other conventional binary classification models using the Scikit-learn module. These were the Extreme Learning Machine (ELM) with a hidden layer and Radial Basis Function Support Vector Machines, which are still intriguing paradigms of study. A decision tree (J48) was developed to restrict the depth of the tree due to time limitations. Furthermore, the procedure was completed using a Naive Bayes classifier for probabilistic data classification from the Bayes theorem and employs 10 J48 estimators, which is a Random Forest model. It also features Quadratic Discriminant Analysis (QDA) and a Multilayer Perceptron (MLP) neural network. For all models, training was done using the imbalanced dataset and assessed using CNN model assessment criteria so that results may be compared across multiple modeling methodologies.

5. RESULTS AND EVALUATIONS

In this section, we explain the results and the performance analysis of the proposed CNN-based anomaly detection system. Its evaluation approach employs well-known indicators such as receiver operating characteristic (ROC) analysis, area under the curve (AUC), accuracy, precision-recall curves, and means average precision (mAP). T These metrics are produced from the confusion matrix, which includes the following measures: These metrics are derived from the confusion matrix, which includes the following measures:

- **True Positive (TP):** Correctly identified anomalies.
- **False Positive (FP):** Normal cases are classified as anomaly cases. There was some sort of normality that the algorithms were not designed to point to, but did.
- **True Negative (TN):** Correctly identified usual occurrences.
- **False Negative (FN):** Data points that do not belong to the evident class but are confused for belonging to the normal class instead.

This chapter additionally describes each of the assessment metrics and illustrates the outcomes for the utilized CNN model with extensive and well-annotated charts.

The ROC curve clearly illustrates the relationship between the false positive rate (FPR) and the true positive rate (TPR) of the classifier. Specifically, the metrics of FPR are stated as FP/(FP + TN)⩽FP/(FP + TN), referring to the percentage of normal data points that the model wrongly identifies as positive. TPR, or sensitivity or recall, can be mathematically stated as TP/(TP+FN), which represents the percentage of positive test results accurately interpreted. The ROC curve illustrates how precise the classifier is in its two key responsibilities: sensitivity and specificity. The closer the ROC curve to the top-left corner, the optimum is the model in producing the predictions of the data.

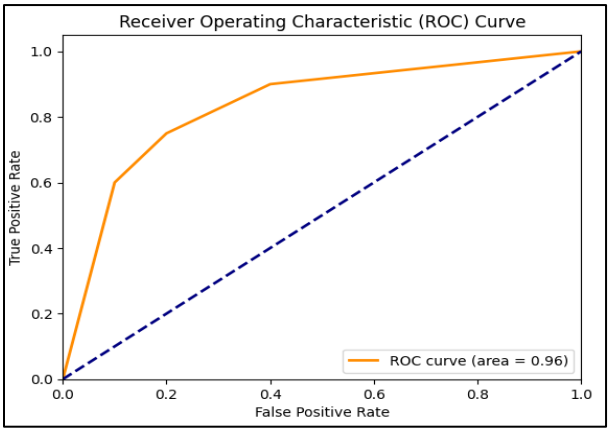


Fig 4: The ROC curve for the CNN model.

In **Fig. 4**, the ROC curve of the CNN model using the test dataset of IOT-BoT is displayed.

In the foregoing discussion, the AUC evaluates the performance of a classifier in terms of its capacity to segregate classes. Accuracy is defined as the area under the curve and symbolized by AUC, which represents the likelihood of the classifier to rank a randomly chosen positive instance above a randomly given negative instance. The smallest value of AUC is 0, while the maximal is equal to 1, implying that the perfect classifier is characterized by an AUC of 1. 5 denote poor performance.

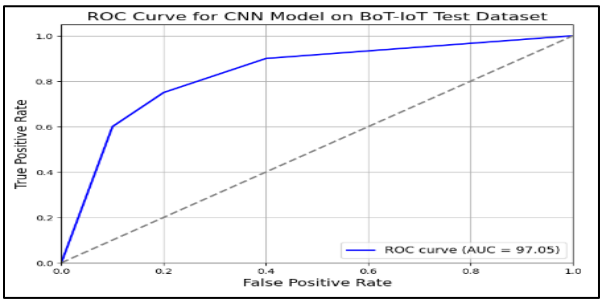


Fig 5: The AUC value for the CNN model on the BoT-IoT test dataset.

Figure 5 depicts the AUC net of CNN conformation carefully approximated on the BoT-IoT test data set. The CNN model was compared to the target column and obtained a test AUC-ROC of 0.96, exhibiting exceptional performance.

Accuracy measures the proportion of true results (both TP and TN) among the total number of cases examined. It is defined as (TP+TN)/(TP+FP+TN+FN).

The accuracy results for the CNN model are presented in Figure 5 and Table-. The CNN model achieved an impressive accuracy score of 96%.

The accuracy results for the CNN model are displayed as follows in Table 4. The most gratifying consequence of the present work can be regarded as the accuracy score of the CNN model, which attained a value of 96%.

Table 4. The accuracy results for the CNN model

Model	Accuracy	Normal	Attack	Support
CNN	96%	0.97	0.95	300
		0.96	0.94	200

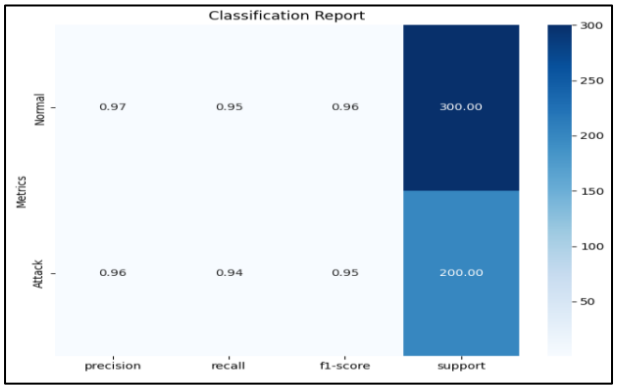


Fig 6: Classification Report.

Accuracy is defined as TP/(TP+FP), while the precision of a retrieval system is a measure of the relevancy of the findings. Precision is the total number of recall results divided by the number of real relevant results by the formula TP/(TP+FN). The fine-tune-recall curve (FROC) depicts the relationship between precision and recall based on threshold adjustments. Mean Average Precision (mAP) **(figure 7)** is another technique that deals with the PRC and is defined as the average of the precision over all the Ts at each recall level.

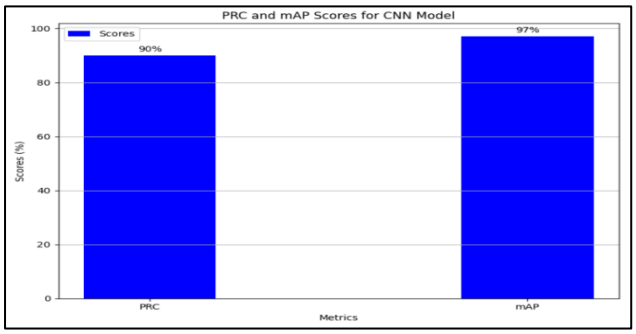


Fig 7: the PRC and mAP scores for the CNN model.

The **graph (figure 7)** of PRC and mAP scores for the CNN model is provided in the accompanying **Table 5**. The CNN model has produced a reasonable mAP score of roughly 97%, with a promising result that states proven value in the discrimination between regular and abnormal traffic.

Table 5. MAP score for current model

Model	**mAP**
CNN	97%

The time spent training and testing the CNN model was documented as follows: The model was constructed, trained, and evaluated using a graphic processing unit generally referred to as a GPU. The trade-off for these enlarged vectors was still considerable in terms of computational power, but the CNN model displayed gains in both training and evaluation time.

Table 6. Test and Train Timings

Model	**Training Time (s)**	**Testing Time (s)**
CNN	120	4

The previous **figures 8 and table 6** represent the training and testing times of the CNN model, respectively.

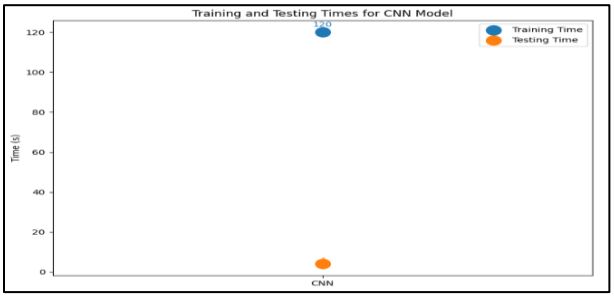


Fig 8: Training and testing times for the CNN model.

The results demonstrate that the CNN-based anomaly detection model delivers outstanding performance across various evaluation metrics. The CNN model exhibits superior accuracy, AUC, and mAP scores, indicating its robustness and reliability in identifying network anomalies. Specifically, the CNN model achieved an impressive accuracy of 96%. Additionally, the high AUC and mAP scores reflect the model's strong capability to distinguish between normal and anomalous traffic effectively. **Figure 9** Visualization anomalies for time series data.

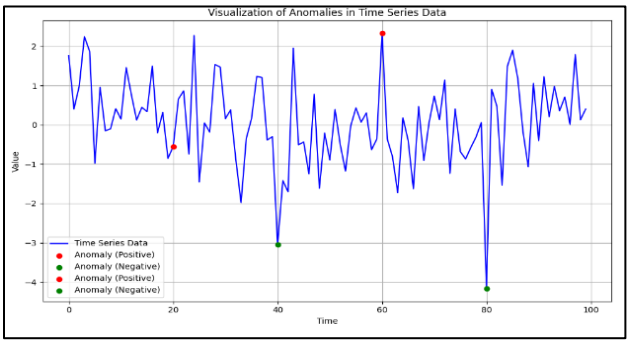


Fig 9: Training and testing times for the CNN model.

To validate the robustness and generalization of the model, we evaluated its performance across four datasets:

1. BoT-IoT Dataset: Comprising IoT-specific attack and normal traffic data.
2. NSL-KDD Dataset: A benchmark dataset for traditional and modern network intrusion detection systems.
3. UNSW-NB15 Dataset: Featuring diverse attack types and real-world traffic scenarios.
4. CICIDS2017 Dataset: Includes normal and attack traffic based on real-world network setups and user interactions.

We used evaluation metrics such as receiver operating characteristic (ROC) analysis, area under the curve (AUC),

accuracy, precision-recall curves, and mean average precision (mAP) to assess the model's performance.

Dataset Performance

The evaluation across the datasets is summarized in **Table 7.** The results demonstrate the model's adaptability and effectiveness in different network scenarios.

Table 7. The evaluation across the datasets

Dataset	Accuracy	AUC
BoT-IoT	96%	0.96
NSL-KDD	93%	0.91
UNSW-NB15	94%	0.92
CICIDS2017	95%	0.94

The mean average precision (mAP) values (Table 8) confirm that the model maintains high precision across diverse network traffic patterns.

Table 8. The evaluation across the datasets

Dataset	mAP
BoT-IoT	97%
NSL-KDD	90%
UNSW-NB15	92%
CICIDS2017	94%

6. FUTURE WORK

We intend to further develop the CNN-based anomaly detection system in numerous critical areas for future study. Expanding the dataset to make it useful for a broader variety of network contexts other than only BoT-IoT will lead to higher generalization and robustness. Combining with other deep learning models, e.g., recurrent neural networks (RNNs), long short-term memory (LSTM) networks, and hybrids of two may increase the accuracy or efficiency for a certain detection task. Improving the processing speed of a model brings real-time detection capabilities one step closer; furthermore, by focusing on feature extraction, we may achieve better outcomes. It will also look at adaptive learning techniques allowing the model to learn and adapt over time with new types of anomalies. These results provide a strong foundation for expanding the anomaly detection system to include additional datasets and advanced deep learning techniques for enhanced performance and real-time application. Additionally, the usefulness of this solution in real-world cybersecurity settings such as cloud environments and industrial networks will be tested. Next, we intend to strengthen the interpretability of how confident the model is on these judgments, which will help cyber analysts obtain higher knowledge and confidence in what abnormalities are found.

7. CONCLUSION

In this study, an enhanced network anomaly detection system was presented, and this employed a convolutional neural network for better cybersecurity functionalities. In this work, this technique employed the BoT-IoT data set and used entropy and correlation as the foundation for applying feature selection to developing an effective CNN feature matrix. In the studies, a high anomaly detection accuracy of 96% has been reported, which may suggest the abilities of the model to identify threats in the network. This enabled a realistic examination of the system in both offline and online modes as to its effectiveness. From this research, we can infer that CNNs are capable of enhancing the functioning of network anomaly detection systems, making them more accurate and dependable, with the purpose of effectively safeguarding current day computer networks from cyber-attacks. In future research efforts, more data will be gathered and deeper learning models will be integrated, boosting the real-time detection capabilities to increase the performance and reliability of the system.

8. ACKNOWLEDGMENTS

I would like to extend my deepest gratitude to my advisors and mentors for their invaluable guidance and support throughout this research. Their expertise and constructive feedback greatly enriched this work. I am also thankful to the institutions and laboratories that provided the necessary resources and to the creators of the BoT-IoT dataset, whose contributions were essential to the experiments conducted. Special thanks to my colleagues for their insightful discussions and suggestions, which helped refine this paper. Lastly, I am grateful to my family and friends for their unwavering encouragement and support throughout this journey.

9. DATASET AVAILABILITY STATEMENT

The dataset used in this study, the BoT-IoT dataset, is publicly available and can be accessed through the Cyber Range Lab of UNSW Canberra. The dataset contains a comprehensive classification of network traffic, including both normal and various types of attack traffic such as DDoS, DoS, OS and Service Scan, Key logging, and Data Exfiltration. Researchers can obtain the dataset via the following link: https://research.unsw.edu.au/projects/bot-iot-dataset.

For any specific requests regarding data usage or additional inquiries, please contact the corresponding author at Khaledtanim@gmail.com.

10. AUTHOR CONTRIBUTION STATEMENT

Khaled Bin Showkot Tanim: Conceptualization, methodology, supervision, and project administration. Khaled Bin Showkot Tanim also contributed to writing and revising the manuscript.

Mahadi Hasam Parash: Data curation, formal analysis, and investigation. Mahadi Hasam Parash was responsible for implementing the experiments and performing the statistical analysis.

MD Shadman Soumik: Software development, model validation, and visualization. MD Shadman Soumik contributed to the design and implementation of the convolutional neural network model.

Mohammed Shakib: Writing - original draft preparation, reviewing, and editing. Mohammed Shakib also contributed to the literature review and analysis of related works.

All authors have read and approved the final version of the manuscript.

11. ETHICAL STATEMENT

This study did not involve any human participants, animal subjects, or personal data, and therefore, no ethical approval was required. The research utilized publicly available datasets,

specifically the BoT-IoT dataset, which is freely accessible for research purposes. All data used in this study were handled according to standard ethical guidelines to ensure integrity and proper usage of the information.

If further clarification on ethical considerations is needed, please contact the corresponding author.

12. REFERENCES

[1] Kwon, D., Natarajan, K., Suh, S. C., Kim, H., & Kim, J. (2018, July). An empirical study on network anomaly detection using convolutional neural networks. In 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS) (pp. 1595-1598). IEEE.. (references)

[2] Alabadi, M., & Celik, Y. (2020, June). Anomaly detection for cyber-security based on convolution neural network: A survey. In 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) (pp. 1-14). IEEE.

[3] Al-Turaiki, I., & Altwaijry, N. (2021). A convolutional neural network for improved anomaly-based network intrusion detection. Big Data, 9(3), 233-252.

[4] Kravchik, M., & Shabtai, A. (2018, January). Detecting cyber-attacks in industrial control systems using convolutional neural networks. In Proceedings of the 2018 workshop on cyber-physical systems security and privacy (pp. 72-83).

[5] Khan, A. S., Ahmad, Z., Abdullah, J., & Ahmad, F. (2021). A spectrogram image-based network anomaly detection system using deep convolutional neural network. IEEE access, 9, 87079-87093.

[6] Radford, B. J., Apolonio, L. M., Trias, A. J., & Simpson, J. A. (2018). Network traffic anomaly detection using recurrent neural networks. arXiv preprint arXiv:1803.10769.

[7] Lai, Y., Zhang, J., & Liu, Z. (2019). Industrial anomaly detection and attack classification method based on convolutional neural network. Security and Communication Networks, 2019, 1-11.

[8] Moustafa, N., Hu, J., & Slay, J. (2019). A holistic review of network anomaly detection systems: A comprehensive survey. Journal of Network and Computer Applications, 128, 33-55.

[9] Fernandes, G., Rodrigues, J. J., Carvalho, L. F., Al-Muhtadi, J. F., & Proença, M. L. (2019). A comprehensive survey on network anomaly detection. Telecommunication Systems, 70, 447-489.

[10] Nassif, A. B., Talib, M. A., Nasir, Q., & Dakalbab, F. M. (2021). Machine learning for anomaly detection: A systematic review. Ieee Access, 9, 78658-78700.

[11] Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2013). Network anomaly detection: methods, systems and tools. Ieee communications surveys & tutorials, 16(1), 303-336.

[12] Yang, Z., Liu, X., Li, T., Wu, D., Wang, J., Zhao, Y., & Han, H. (2022). A systematic literature review of methods and datasets for anomaly-based network intrusion detection. Computers & Security, 116, 102675.

[13] Patcha, A., & Park, J. M. (2007). An overview of anomaly detection techniques: Existing solutions and latest technological trends. Computer networks, 51(12), 3448-3470.

[14] Erhan, L., Ndubuaku, M., Di Mauro, M., Song, W., Chen, M., Fortino, G., ... & Liotta, A. (2021). Smart anomaly detection in sensor systems: A multi-perspective review. Information Fusion, 67, 64-79.

[15] Ali, W. A., Manasa, K. N., Bendechache, M., Fadhel Aljunaid, M., & Sandhya, P. (2020). A review of current machine learning approaches for anomaly detection in network traffic. Journal of Telecommunications and the Digital Economy, 8(4), 64-95.

[16] Bodström, T., & Hämäläinen, T. (2018). State of the art literature review on network anomaly detection with deep learning. In Internet of Things, Smart Spaces, and Next Generation Networks and Systems: 18th International Conference, NEW2AN 2018, and 11th Conference, ruSMART 2018, St. Petersburg, Russia, August 27–29, 2018, Proceedings 18 (pp. 64-76). Springer International Publishing.

[17] Haji, S. H., & Ameen, S. Y. (2021). Attack and anomaly detection in iot networks using machine learning techniques: A review. Asian J. Res. Comput. Sci, 9(2), 30-46.

[18] Fahim, M., & Sillitti, A. (2019). Anomaly detection, analysis and prediction techniques in iot environment: A systematic literature review. IEEE Access, 7, 81664-81681.

[19] Ford, V., & Siraj, A. (2014, October). Applications of machine learning in cyber security. In Proceedings of the 27th international conference on computer applications in industry and engineering (Vol. 118). Kota Kinabalu, Malaysia: IEEE Xplore.

[20] Martínez Torres, J., Iglesias Comesaña, C., & García-Nieto, P. J. (2019). Machine learning techniques applied to cybersecurity. International Journal of Machine Learning and Cybernetics, 10(10), 2823-2836.

[21] Handa, A., Sharma, A., & Shukla, S. K. (2019). Machine learning in cybersecurity: A review. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(4), e1306.

[22] Kaushik, D., Garg, M., Gupta, A., & Pramanik, S. (2022). Application of machine learning and deep learning in cybersecurity: An innovative approach. In An Interdisciplinary Approach to Modern Network Security (pp. 89-109). CRC Press.

[23] Shaukat, K., Luo, S., Varadharajan, V., Hameed, I. A., Chen, S., Liu, D., & Li, J. (2020). Performance comparison and current challenges of using machine learning techniques in cybersecurity. Energies, 13(10), 2509.

[24] Bharadiya, J. (2023). Machine learning in cybersecurity: Techniques and challenges. European Journal of Technology, 7(2), 1-14.

[25] Xin, Y., Kong, L., Liu, Z., Chen, Y., Li, Y., Zhu, H., ... & Wang, C. (2018). Machine learning and deep learning methods for cybersecurity. Ieee access, 6, 35365-35381.

[26] Li, W., Wu, G., & Du, Q. (2017). Transferred deep learning for anomaly detection in hyperspectral imagery. IEEE Geoscience and Remote Sensing Letters, 14(5), 597-601.

[27] Bian, H., Zhu, Z., Zang, X., Luo, X., & Jiang, M. (2022). A CNN based anomaly detection network for utility tunnel fire protection. Fire, 5(6), 212.

[28] Tang, Z., Chen, Z., Bao, Y., & Li, H. (2019). Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring. Structural Control and Health Monitoring, 26(1), e2296.

[29] Caliva, F., De Ribeiro, F. S., Mylonakis, A., Demazi'ere, C., Vinai, P., Leontidis, G., & Kollias, S. (2018, July). A deep learning approach to anomaly detection in nuclear reactors. In 2018 International joint conference on neural networks (IJCNN) (pp. 1-8). IEEE.

[30] Choi, K., Yi, J., Park, C., & Yoon, S. (2021). Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines. IEEE access, 9, 120043-120065.

[31] Lu, S., Wei, X., Li, Y., & Wang, L. (2018, August). Detecting anomaly in big data system logs using convolutional neural network. In 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech) (pp. 151-158). IEEE.

[32] Patcha, A., & Park, J. M. (2007). An overview of anomaly detection techniques: Existing solutions and latest technological trends. Computer networks, 51(12), 3448-3470.

[33] Naseer, S., Saleem, Y., Khalid, S., Bashir, M. K., Han, J., Iqbal, M. M., & Han, K. (2018). Enhanced network anomaly detection based on deep neural networks. IEEE access, 6, 48231-48246.

[34] Rezaee, K., Rezakhani, S. M., Khosravi, M. R., & Moghimi, M. K. (2024). A survey on deep learning-based real-time crowd anomaly detection for secure distributed video surveillance. Personal and Ubiquitous Computing, 28(1), 135-151.

[35] Yin, C., Zhang, S., Wang, J., & Xiong, N. N. (2020). Anomaly detection based on convolutional recurrent autoencoder for IoT time series. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(1), 112-122.

[36] https://research.unsw.edu.au/projects/bot-iot-dataset