Anomaly Detection for Cyber-Security Based on Convolution Neural Network : A survey

1st Montdher Alabadi
Computer Engineering Department
Karabuk University
Karabuk,Turkey
montdher10@gmail.com

2nd Yuksel Celik
Computer Engineering Department
Karabuk University
Karabuk,Turkey
yukselcelik@karabuk.edu.tr

***Abstract*—The expanding growth of computer and communication technologies results in a vast amount of security concerns. Various types of cyber-security enabled mechanisms have been developed to limit these concerns. Anomaly detection is among these mechanisms. Anomaly detection means using multiple techniques and methods to detect different patterns that do not conform to defined features of whole data. Recently, deep learning techniques adopted as a satisfactory solution because of its ability to extract data features from data itself. Convolution neural network (CNN) is mainly utilized because of its ability to process input with multi-dimensions. In this paper, a comprehensive survey about using CNN as a key solution for anomaly detection is provided. Most of the existing solutions in the literature have been gathered and classified according to the input data source; furthermore, this paper suggests a unified cross framework that simulates end-to-end anomaly detection mechanisms that exist in the previous studies. A unified cross framework enriches this paper with in-depth analysis to clarify how the solution in the literature uses CNN in anomaly detection. Finally, this paper suggests several future research directions that can support the audience in their future works in this context.**

***Index Terms*—Anomaly Detection ,CNN ,Deep Learning , Security**

I. Introduction

Anomalies are some data points that have different patterns that do not conform to defined features of whole data. Anomalies' existence is related to many reasons, such as fraud, cyber-security attack, and malicious activity [1]. Anomaly detection means using various techniques and methods to detect these anomalies; these techniques range from artificial intelligence to the statistical methods. Anomaly Detection is very challenging because it requires analyzing many aspects. The first challenge is defining the boundary between normal and abnormal behaviors. The second one is the continuous evolution of malicious activity. Another problem is the heterogeneity of applications; for example, the health care anomaly detection solution might not fit into transportation applications. The last challenge is the need for the labeled data that can be used to train the model or system. In the literature, most studies classify anomaly into three categories [1] :

1) Point Anomalies: It is the basic form of the anomaly and is the objective of most works on the identification of anomalies. Point Anomalies can be described as an individual data point that is considered abnormal in proportion to the rest of the data.
2) Contextual Anomalies: If a data point is abnormal in a particular context, or otherwise not, then a contextual anomaly is characterized
3) Collective Anomalies: When a series of similar data points are anomalous across the whole data set, it is considered a collective anomaly. In a collective anomaly, the single data instances may not be anomalies by themselves, but their presence as a set together would be abnormal.

The solutions that target anomaly detection can take several forms, such as fraud detection, intrusion detection, and specific domain solutions. Choosing the appropriate solution for anomaly detection depends on many factors; the most important one is types of input data; another is whether the detection needs to process in real-time or not. In modern studies, researchers focus on the machine and deep learning techniques to provide accurate anomaly detection schemes—the reason behind using these techniques, especially deep learning, reduces the effort for input pre-processing. Deep learning can extract features of data from input data itself, which may fit for schemes that need real-time processing.
In this study, the use of Convolutional Neural Network CNN [2] to provide anomaly detection schemes has been conducted. Convolutional Neural Network is the primary implementation of deep learning when the input has multiple dimensions, which is hard to be handled by the traditional neural network. In this survey, an organized methodology has been applied to collect the existing articles in the literature related to anomaly detection by using CNN. All of the gathered studies were analyzed in terms of core solutions, mechanisms of work,

978-1-7281-9352-6/20/$31.00 ©2020 IEEE

datasets used, performance metrics, novelty comparisons, and future works. This survey analysis has been supported by a suggested framework called a unified cross framework, which highlights the techniques that act in the background of each solution to provide anomaly detection mechanisms. from all the above, the contributions of this paper can be summarized in the following points :

1) According to our knowledge, this survey is first, which discusses specific CNN solutions for anomaly detection.
2) This survey provides a novel classification for the existing solution based on the type of input data.
3) This survey investigates each solution in detail regarding theoretical and experimental aspects to enrich the readers with respect amount of knowledge; this has been done by suggesting a framework called a unified cross framework. Most of the existing solutions have been crossed checked to the unified cross framework, which shows the mechanism of anomaly detection and techniques involved.
4) Future research directions have been highlighted to support researchers in their future works.

A. Methodology and Orgnization

The methodology that has been adopted for preparing this survey is shown in the figure 1 . This paper methodology is conducted by applying the following steps :

1) Several databases have been used as a source for preparing this survey, such as IEEE, Springer, and other repositories.
2) The search starts with three keywords, CNN, anomaly detection, and machine learning; besides, the publishing date adjusted to those published after 2016.
3) The result was finding 14 reviews and 55 proposed solutions that match search entries.
4) The 55 proposed solutions have been refined to those focused on CNN and anomaly detection, and the result was 21 study.
5) The 21 studies analyzed in terms of core solution, techniques used, experimental aspects, and results comparisons.
6) The 14 reviews analyzed and presented as a related work section of this survey.

This survey is organized as follows; in II, related works are addressed and analyzed. In section III, theoretical background and machine learning and convolution neural network. In section IV, the taxonomy of solutions based on the input source is presented. Section V provides an in-depth analysis of proposed solutions. Future work is addressed in section VI, while section VII contains the survey conclusion.

II. Related Works

Several reviews on machine learning techniques and algorithms for anomaly detection were presented recently, each of which covers a different aspect of this integration. In [3], authors study the use of ML in numerous IoT contexts, the attempts to bring the headlines to the forefront, the research analyzes ML applications across both IoT data processing and management functions, study directions, and difficulties. In [4], the work offers a scientific classification of existing IoT security dangers and provides a guide for novel and energizing research difficulties in applying ML and SDN ideas to address IoT security concerns. In [5], Authors review several detailed works on safety problems and defensive tactics from a data-driven viewpoint when learning and evaluating or inferring machine learning. They emphasize the drifting of data distribution induced by adversarial samples and sensitive information leakage issues in predictive machine learning algorithms. In [6], authors provide Survey a few cybersecurity aspects in which machine-learning can be used. There has also been a debate on the dangers of adversary attacks, which can exploit the training and testing data for classifiers. They particularly highlight knowledge of cybersecurity machine learning techniques. In [7], the author's Survey of Internet / Communication Security provided by machine learning and data mining techniques. The study highlighted papers trying to define the use of multiple machine learning and data mining methods for anomaly-based detection on the Internet domain. In [8], studies on machine learning and deep learning methods of network analysis intrusion detection and gives a brief walkthrough summary of each ML and DL technique. [9] survey overviews several works that used machine learning techniques and deep learning techniques in several research areas, including networking, communications, and lossy environment. The primary objective of this survey study is to distinguish potential topics and challenging tasks for using different deep learning and machine learning algorithms.
In [10], research studies different machine learning and deep learning algorithms used to develop network intrusion detection and strategies used to define cybersecurity and intrusion detection accuracy in addition to existing blockchain technology applications. In [11] author's Survey categorizes the IoT anomaly detection methods developed into machine learning, statistical, and deep learning methods. Authors were attempting to provide up-to-date information and research gaps. In [12] researchers, Survey use of deep learning and machine-learning anomaly detection techniques in IoT. In [13], Offer an overview of the state-of-the-art, deep learning algorithms and network traffic management algorithms. They also address the deep learning enabling factors for network systems and the deep learning-based smart routing. In [14] authors Provide an anomaly detection system categorization for DDoS attacks. Highlight then the statistical methods used to detect anomalies. In [15] authors, provide an overview of graph-based anomaly detection approaches, highlight relevant research problems related to applications, and identifying potential directions for expanding fraud detection studies. They rely on research using graph-based strategies with data that contains interactions among actors in the network to attain these targets. They also establish a classification system to modulate existing works. In [16], authors Analyzing the amount of research for anomaly detection based on real-time big data processing. They suggest categorizing published articles into several categories like big

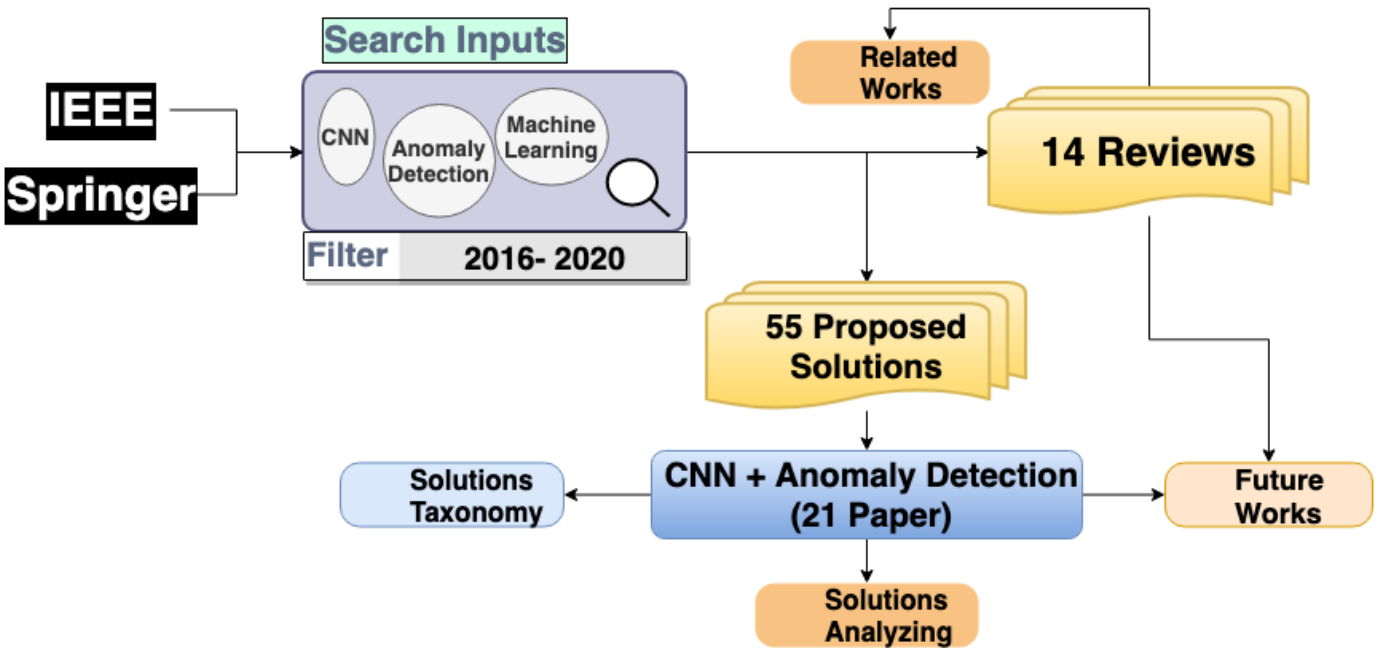

Fig. 1. Survey Methodology

data analytics, identification of anomalies, machine learning algorithms, mechanisms, information, and implementation. Finally, the research challenges and recommendations for the future researchers are highlighted.

Table I provide a summarization of all the above explained related works.

III. Machine Learning and Deep Learning

The term learning is simply the ability to improve task execution over time; that was the motivation behind the existence of machine learning. Machine learning is the term that builds around many types of algorithms; the goal behind these algorithms is to develop task execution and improve this execution like what humans can do. During past years, a massive amount of work has been done to improve the performance of these algorithms; these improvements allow establishing these algorithms in many human life aspects and real-life problems. Machine learning algorithms are classified according to the type of input besides how the output is produced. Supervised algorithms depend on data, which is labeled with the correct output. The unsupervised algorithm uses an array of unlabeled knowledge to see the patterns underlying it. In reinforcement learning, the correct output is not available from the initial. In any case in a trial-and-error style, the predicted output can be evaluated with a positive or negative reward showing how good or bad the output. [3].

Machine learning has some constraints, and the most challenging one is handling new problems that need a vast amount of labeled data. The solution to the previous issue is to identify the feature of data input before applying the machine learning algorithm. The mentioned solution could lead to a high level of a bottleneck because there is a limit of what the human can do in terms of identifying the data features. Deep learning overcomes this bottleneck because of its ability to identify the features and representation from raw data itself. The word deep refers to the hierarchy of distinguishing features and the ability to learn from the raw data itself. Deep learning algorithms are composed of multiple layers called hidden layers; these layers can provide the model with feature extraction and learning methodology [17].

Deep learning comes with several implementations. These implementations differ in the purpose, which requires modifying or rearranging the hidden layers. These implementations can have only forwarding processing or both forwarding and backward processing. Feedforward neural network is a base for all deep learning implementation; this network consists of an input layer, hidden layers, and output layer. Each layer of feed-forward consists of a specific number of perceptrons that receive input data, weights, and a bias value. According to the function called activation function, that output will be produced. In the feed-forward neural network, there is error calculated using values provided in the output layer. The goal is to reduce these errors by tuning the parameters (bias and weights) [17]. Recurrent neural networks (RNN) use the same concept of the feed-forward neural network; the difference is that RNN uses the idea of ”data memory. “Data memory means to store the value of outputs of different layers and pass it to previous layers cross sequence steps cite [18]. There is also an extension for RNN called Long Short-term Memory (LSTM) [19]. LSTM uses a gating approach. In LSTM, there is a hidden state generated from paste outputs. The gate takes the input and the hidden state as a parameter to the activation function to create a new input.LSTM is fit to override the problem of the vanishing gradient [20].

TABLE I
RELATED WORKS SUMMARY

Reference	Domain	Solutions Scope	Survey Summary
[3]	IoT Security	Machine Learning	Discuss the Security of Data Processing and Mangement
[4]	IoT Security	Machine Learning + SDN	Classifying IoT Security Threats and the Ideas to Handle Them
[5]	ICT Security	Machin Learning	Discuss Predictive Machine Learning Algorithms for Cyper-Security
[6]	ICT Security	Machin Learning	Survey Cybersecurity Aspects in Which Machine- Learning Can Be Used
[7]	Internet Security	Machine Learning + Data Mining	Discuss the Use of Machine Learning / Data Mining for Anomaly Detection
[8]	Network Security	Machine Learning + Deep Learning	Discuss Machine Learning and Deep Learning for Intrusion Detection
[9]	Network Security	Machine Learning + Deep Learning	Discuss Potential Topics and Challenging Tasks for Using Different Deep Learning and MachineLearning in Network Security
[10]	Network Security	Machine Learning + Deep Learning	Discuss Network Intrusion Detection Based on Deep and Machine Learning
[11]	IoT Security	Machine Learning +Statistical Method + Deep Learning	Classify Anomaly Detection Schemes Based on Machine Learning , Statistical Method and Deep Learning
[12]	IoT Security	Machine Learning + Deep Learning	Classify Anomaly Detection Schemes Based on Machine Learning and Deep Learning
[13]	Network Security	Deep Learning	Discuss Deep Learning Algorithms Securing Network Traffic
[14]	Network Security	Statistical Methods	Analyze Anomaly Detection System Categorization for DDoS Attacks
[15]	Network Security	Graph-Based Strategies	Address a Graph-Based Anomaly Detection Approaches
[16]	Network Security	Big Data	Discuss Anomaly Detection Solutions Based on Big Data Processing

Convolutional Neural Network (CNN) is the primary implementation of deep learning when the input has multi-dimensions, which is hard to be handled by the traditional neural network [2]. In the next section, CNN will be discussed with more details because it is selected as a research scope in this paper.

A. CNN

Convolutional Neural Network is the most appropriate implementation of deep learning when the input has multi-dimensions such as images, which is hard to be handled by the traditional neural network. CNN has a different layer regarding the architecture. Traditional CNN network consists of the following layer :

1) Input Layer: this layer takes the input, which could have one or more dimensions and feed it to the next layer.
2) Convolution Layer: this layer has filters with specific sizes; these filters responsible for apply convolution operation to data come from the input layer. The convolution operation is applied where each filter scans the input data, and this scanning operation is applied to all data according to the filters and stride sizes [2].
3) Pooling Layer: this layer applies its functions after the convolution layer by summarizing the data snapped by the filters, this summarizing is usually done by applying the max pooling operation, which chooses a max value in the given screen [2].
4)
5) Fully Connected Layer: typically exist at the end of CNN, this layer flattened input so each input will connect to all perceptrons. The existence of this layer, along with perceptron's, can optimize classification accuracy [2].

The output of each convolution layer on CNN with conjunction with pooling operation usually called the feature map; this feature map depends highly on the size of the input and filter. The feature map can be obtained by using a specific formula. For example, if we have input has a two-dimension$I(m,n)$ and the filter size are $F(a,b)$, the output can be calculated according to the following function :

$$O = \sum_{a} \sum_{b} I(m+a, n+b).F(a,b) \tag{1}$$

The output from the convolution layer will feed to the pooling layer to provide a more abstract view of the feature map, max-pooling usually used as a candidate method to apply pooling processes. In max-pooling another filter with specific size will be applied,along with the filter size there is something called stride which govern the distance that filter will move to scan the input.

After the convolution and pooling complete, output can now feed to the fully connected layer to provide a flat view of the feature map. The flat view usually connects to another perceptron's network, which used to optimize the accuracy of the model as an existing feed-forward neural network. Besides the input and filter size, some parameters need to be addressed to make CNN work efficiently. An example of these parameters is something called padding [21]. Padding is the process of adding zeros to the boundaries of the input; padding operation can help filter scanning, so every unit in the input will be scanned once and will be no reputation due to the filter size.

CNN has much application starting form object detection in images until anomaly detection in the cyber-security. Using CNN with object detection requires fewer efforts than other algorithms, this because there is no need to establish complex pre-processing to the input data, but the classification with CNN is challenging when the input needs pre-processing such as network traffic. CNN has many advantages over other deep learning types; the most important is the number of the parameters involved and needs to be tuned. Another benefit regarding CNN is that it does very well when the problem is related to object detection and classification. CNN, like most of the other machine and deep learning techniques, has

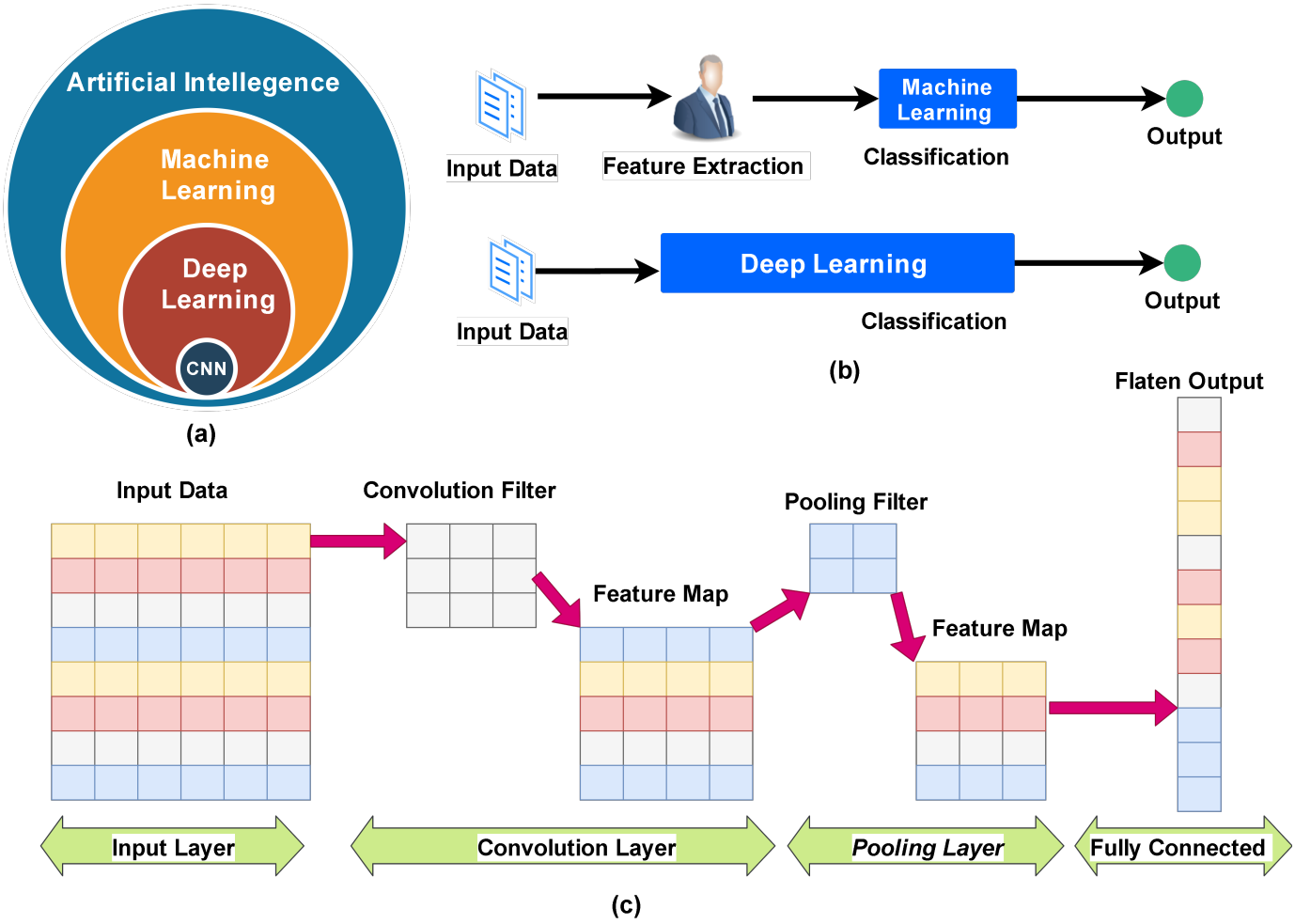

Fig. 2. (a) CNN with Respect to AI , (b) Machine Learning vs Deep Learning , (c) CNN Archiicture

many performance metrics; these metrics are used to evaluate how the model is good [22]. Accuracy is the most used to investigate the model performance; Accuracy can be given as the percentage of correct prediction to the overall input samples :

$$Accuracy = \frac{CorrectPredictions}{TotalPredictions} \tag{2}$$

the second metric called the confusion matrix; in this metric, the output is provided as a matrix to show the overall performance of the model. This metrics has sub-terms that can offer a more precise view about the performance .these sub-terms are :

1) True Positives (TP): The cases in which the model prediction is positive, and the actual output was also positive.
2) True Negatives (TN): The cases in which the model prediction is negative, and the actual output was also negative.
3) False Positives (FP): The cases in which the model prediction is positive, and the actual output was negative.
4) False Negatives (FN): The cases in which the model prediction is negative, and the actual output was positive.

Confusion matrix accuracy can then calculate with the following formula :

$$Accuracy = \frac{TP + TN}{TotalSample} \tag{3}$$

according to the sub-terms, we can derive other performance metrics such as true positive rate (TPR) which represents the ratio of data classified or predicted positively to the all positive data as the shown in the current formula :

$$TNR = \frac{TP}{FN + TP} \tag{4}$$

in the same way, we can derive true negative rate (TNR) representing the ratio of data classified or predicted negatively to all negative data as shown in the following formula :

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

false positive rate (FPR) realize the ratio of negative that is mistakenly considered as positive to all negative data as shown in the following formula :

$$FPR = \frac{FP}{TN + FP} \tag{6}$$

Another essential metric is called F1; this metric can be used to explore how the given model is excellent and robust in terms of instance, classifying. In order to calculate F1, we need to calculate the precision and recall, as shown in the following formulas :

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F1 = 2 * \frac{1}{\frac{1}{Precision} + \frac{1}{Recall}} \tag{9}$$

Figure 2 provide graphical explanation about CNN mechanism as well as showing the differences between machine learning and deep learning .

IV. Solutions Taxonomy

In the literature, there are many classifications for anomaly detection solutions, such as these classifications based on the techniques and methods involved. This study suggests classifying anomaly detection solutions according to the source of input data. As a result of this classification, there are two types of data sources. The first one is network traffic data, and the second one is system log files.

A. Traffic – Based Solutions

In [23], the authors suggest an anomaly identification method to HTTP messages. This paper is utilizing a binary image transformation based on convolutional autoencoder (CAE)with the character level. The encoder 's structure is symmetrical to that of the decoder. In the case of normal messages, the suggested CAE generates outputs for anomalous messages that are supposed to have different values. They use binary cross var-entropy (BCV) as a decision variable to represent this property. BCE used to decide whether such a given message is normal or abnormal. About 155000 HTTP messages collected over 600 websites in 2018 as a dataset. As metrics, they show (TPR) and (FPR). Finally, authors Compare their system performance to that of a one-class service Vectoring machine (SVM) and forest isolation (IF).
In [24], the study addresses a Strategy that increases the quality of identifying anomalies using a CNN mechanism. Data encoding is an integral component in this system for converting one-dimensional connection records to an image-like to use CNN for the detection of network anomalies. This Encoding strategy enhances the efficiency of network anomalies detection using a CNN architecture. For the evaluation ,three datasets used UNSW-NB15, IDS2017, and NSL-KDD . Authors investigate F1-score to analyze the encoding efficiency. They Compare the output to the encoding on a grayscale approach.
in [25] , paper purpose Vehicle Intrusion Detection System (IDS). The system starts with the translation of in-vehicle network data into vehicle functions; accordingly, IDS focused on maps data from the controller area network (CAN) to the 2D images. The underlying physical structure can then dynamically organize features into a system that allows the study of the coevolutionary neural network (CNN) More Realistic and find the anomalies. Dataset was collected by generating images from CAN bus traffic. For evaluation, this paper defined custom metrics called blind test accuracy.
In [26] , authors develop an in-depth convolutional approach to anomaly-identification using Convolutional Neural networks (CNN). The system supposes to be applied to the data gathered from various sensors in a time series. They produce 'images' from continuous real-time raw sensor stream and then help anomaly identification by integrating CNN with well-established Kalman filtering anomaly detection. In the evaluation, the authors use SPMD dataset. They also investigate the system in terms of detection performance of constant anomaly and detection performance of instant anomaly. the system results were compared with the performance of CNN alone and performance Kalman filtering alone also.
In [27], the authors' proposed framework for anomaly detection .the processing start when Gray Wolf Optimization (GWO) is used to extract the feature. On the other hand, the classification of anomalies is done using modified CNN. With modified drop-out functionality, CNN helps prevent overfitting and raises the weights of the network's most necessary features. Framework evaluation has been done using Benchmark Dataset-DARPA'98 and Dataset-KDD'99, they also measure feature set against error rate, recall, FPR, precision, accuracy, and F1-score as a performance metrics. The compression to proof framework performance was Performed against standard GWO.
In the [28], the work addresses the electricity grid's vulnerability to false injection of data Attacks (FDI).To catch the complex behavior of the power system, a recurrent neural network with LSTM cells is deployed, and a coevolutionary neural network is adopted to match two data sources. An attack is detected when the differential between the measurements observed, and the estimated measurements exceed a given threshold. Performance evaluation was done using previously collected dataset through Ethernet. The accuracy of detecting FDI was measured as a performance metric.
In [29], the work Addresses deep convolutional Neural Network Using an ensemble approach for detection and identification of DDoS in SDNs. Two single CNN implementations form an ensemble approach with an equal number of layers, activation function, and rounds. This solution aims to identify and detect DDoS attacks inside normal traffic. The DDoS is classified as abnormal data by using this system . Model implementation utilizes the CICIDS2017 dataset. Simultaneously, the performance matrices were FPR, accuracy, precision, recall, FDR, FNR, false omission rate (FOR), and CPU usage.

In [30], authors Develop a CNN and RNN-based model that can extract the features from the original traffic and then identify the malicious activity flow in SDN framework. Firstly, the source data is processed using the preprocessing technique, followed by the CNN and RNN dependent algorithm. Finally, the classification was a linear based classification. A self-generated dataset has been generated to implement this model. Performance metrics in the evaluation part were accuracy, recall, and F1-score. Finally, they Compare malicious flow detection with another deep learning-based scheme.

In [31] , paper suggest Using a convolution neural network (CNN) classifies network events that occur in the LAN. They introduce a solution for creating a particular duration image from the network packet dump. By placing this image via CNN, the network features can be learned. First, they inserted the traffic data protocol information into the LAN By using Hilbert Curve to generate feature maps representing different types of events. After that, the inference model can be used, which is a method that continuously monitors and protects the network, according to which the model can simultaneously detect and classify attack types. Dataset was collected as a Traffic captured with an interval of one day. They evaluate a Recall as a performance metric.

In [32], authors evaluate the theory of using Generative adversarial networks (GANs) to construct specific network traffic packets that follow network standards for real network transmission. The implementation framework is a CNN GAN traffic generator, named PAC-GAN. Network packets are processed using a traffic flow encoding scheme for converting and mapping data over the network traffic into the image-based used in CNN frameworks to detect the anomalies. During implementation, these papers work on to generate three different types of traffic which they are ICMP Ping, DNS queries, and HTTP Get requests to test the model. As a performance metric, the adopt success rate, which is the ratio between packets that are sent effectively versus the total number of Packages created through the GAN / Byte error. Finally, the model has been compared with an alternative to existing traffic generators.

In [33], the paper suggests a model that involves CNN- and LSTM to detect Trojan in the HTTP-based traffic. At the packet level, after encoding the feature of input data, the model uses CNN to derive spatial and character features directly from the raw data and outputs as a matrix. At the flow level, the model uses LSTM to further extract temporal features between packets. Then, the model synthesizes everything Feature information is provided across hidden layers and outputs to demonstrate the probability of anomaly activity. In the implementation, the BTHT dataset has been used. Precision and recall have been measured as a performance metrics .finally authors compare the system result with classical machine learning algorithms such as Bayes, SVM and Decision Tree.

In [34] , the study develops a face spoofing detection solution through the fusion of various anomaly experts. A pool of 63 learners was created through the creation of different combinations from the set of three anomaly detectors, seven facial regions, and three CNN architectures. They have a standardization approach to improve this process to enable a multiple fusion of spoofing detectors using normal data only. Datasets involved in the implementation were Replay-Attack, Replay-Mobile, and Rose-Youtu. The solution has been compared to the system with a single classifier to prove its efficiency in anomaly detection.

In [35], the authors Implement a CNN-based IoT deep learning method for auto-learning traffic characteristics and traffic classification directly from raw traffic for only a few first packets per stream. Subsequently, the auto-learning approach will substantially save efforts to build traffic patterns for a complex network. Datasets used in the evaluation were USTC-TFC2016 and Mirai-RGU, while the performance indicators were the precision, recall, and F1-measure.

In [36], authors Implement a deep learning-based anomaly detector aimed to examine much more of the traffic 's entire payload. For this reason, they consider the unidirectional sequence of packets called message as a standard processing unit rather than each packet. They start with generate message data by aggregating payloads of concurrent packets and regard it as a standard processing unit, then obtain feature vectors in each of the message data using the suggested CNN. Eventually, the RNN is used to decide if the flow is abnormal or not. UNSW-NB15 dataset was used in the evaluation. Performance metrics were accuracy, precision, recall, and F1 score. Finally, they compare their detector with deep learning-based detectors called HAST-IDS and PL-RNN.

In [37], work suggests a deep learning methodology for DDoS attack detection within SDNs. This proposed model utilizes the CNN model ensembles for Flow-based data detection. A system architecture for the deep CNN ensemble has been placed in the SDN controller. This framework consists of four architectures based on the DL (Ensemble RNN, LSTM, CNN, and Hybrid RL). System output would either be 1 or 0,1 for DDoS attack and 0 for normal traffic. Dataset used during implementation was CICIDS-2017. Model performance was measured in terms of recall precision, receiver operator characteristics curve (ROC) graph, and f1-measure. Finally, authors Compare their model with other ensemble models.

In [38] ,authors Implement a web attack detection framework . The framework is equipped with known normal and abnormal HTTP requests on a labeled dataset. The system took web requests as an input of character sequences, CNN will report those local features in the requests for HTTP. Then To identify the attack, these local features are given to the LSTM network. Dataset used was CSIC 2010 HTTP. Evaluation indicators were precision, recall rate, F1-score, and accuracy. Finally, the authors compare their framework with specially designed CNN on the CSIC 2010 dataset.

B. Log-Based Solutions

In [39], the work suggests Industrial Control Systems (ICS) intrusion detection algorithm based on CNN and state transfer algorithm. The CNN model is used for the traditional identification of anomalies and the extraction of features from

TABLE II
SUMMARY OF THE SOLUTIONS

Ref	Domain	Specific Issue	Type of CNN	Role of CNN	Solution Summary	Supporting Techniques	Dataset Used	Performance Metrics	Solution Compared With
[23]	Traffic Based	HTTP Anomaly	CAE	Primary	Identify HTTP messages	Character-level binary image transformation	Prepare their own Dataset	Accuracy / FPR /TPR	One-class service Vectoring machine/ Forest isolation
[24]	Traffic Based	Network Anomaly	CNN	Primary	Enhance network anomalies detection	NO	UNSW-NB15/ IDS2017 / NSL-KDD	F1 score	Encoding on a gray scale
[25]	Traffic Based	Vehicle Network Anomaly	CNN	Primary	Intrusion Detection System supported with CNN	No	Prepare their own Dataset	Accuracy	No
[26]	Traffic Based	Vehicle Network Anomaly	CNN	Primary	Identify Anomaly in the sensor data using CNN	Kalman filtering	SPMD	Detection Perforamnce	Traditional CNN / Kalman filtering alone
[39]	Log Based	Application Anomaly	CNN	Secondary	Detect anomalies in enterprise applications	State transfer algorithm	Prepare their own Dataset	Recall /Accuracy / Precision /F1-score	Bayesian Network / Random Forest / Support Vector Machin / Hidden Markov Model
[27]	Traffic Based	IoT Anomaly	CNN	Primary	Classification of anomalies	Grey wolf optimization GWO	DARPA'98 / KDD'99	Recall / False Positive Rate / Precision /Accuracy/ F-score	Standard GWO
[28]	Traffic Based	Smart Grid Anomaly	CNN	Primary	False Data Injection Using CNN	LSTM	Prepare their own Dataset	Accuracy	No
[29]	Traffic Based	SDN Anomaly	CAE	Primary	Detecting DDoS in SDN Using CNN	No	CICIDS2017	FPR /Accuracy /Precision / Recall /FDR / FNR / FOR/ CPU usage	No
[30]	Traffic Based	SDN Anomaly	CNN	Primary	Identify the malicious activity flow in SDN	RNN	Prepare their own Dataset	Accuracy / Recall /F1-score	Deep learning based scheme
[31]	Traffic Based	LAN Anoamly	CNN	Primary	Detect and classify attack types in LAN	Hilbert Curve	Prepare their own Dataset	Recall	No
[32]	Traffic Based	Generative Adversarial Networks Anomaly	CNN	Secondary	Anomlies Detection using traffic flow encoding scheme	No	Prepare their own Dataset	Success rate	Another traffic generators
[40]	Log Based	System Anomaly	CNN	Primary	Anomaly detection in log files	No	Not Mentioned	Precision / Recall	Supervised CNN Classification / Convolutional Autoencoder(CAE)
[33]	Traffic Based	HTTP Anomaly	CNN	Primary	HTTP trojan detection using CNN	LSTM	BTHT	Precision / Recall	Bayes / SVM / Decision Tree
[34]	Traffic Based	Network Spoofing	CNN	Primary	Spoofing Attack Detection using CNN	No	Replay-Attack / Replay-Mobile/Rose-Youtu	No Clear Deatils	Single classifier
[35]	Traffic Based	IoT Anomaly	CNN	Primary	Anomaly Detection in IoT raw data using CNN	No	USTC-TFC2016 / Mirai-RGU /	Precision /Recall/ F1-mesaure	No
[41]	Log Based	Computer Anomaly	CNN	Primary	Recognize Anomalies based on computer-generated log analysis	LSTM	HDFS	Precision /Recall / F-Measure,FPR	Multi-Layer Perceptron Autoencoder / K-Means
[36]	Traffic Based	Network Anomaly	CNN	Secondary	Intrusion detection sytem using CNN	RNN	UNSW-NB15	Accuracy/ Precision /Recall F1 score	Deep learning based detectors called HAST-IDS and PL-RNN
[37]	Traffic Based	SDN Anomaly	CNN	Primary	DDoS detection in SDN using CNN	RNN / LSTM	CICIDS-2017	Precision / Recall /F1-measure Receiver operator characteristics curve (ROC) graph	Another esenmble model
[42]	Log Based	Anomaly in user activity	GCN	Primary	Threats and fraud detection system	No	CMU CERT v4.2	Accuracy /Precision / Recall	Random Forest / Logistic Regression / SVM / CNN
[43]	Log Based	Anomaly in user activity	CNN	Primary	System for detect abnormal actions of network users	No	NSL-KDD	Stability	No
[38]	Traffic Based	HTTP Anomaly	CNN	Secondary	Web attack detection framework	LSTM	CSIC 2010 HTTP	Precision / Recall / F1-score / Accuracy	Specially Designed CNN

log files. The feature vector is used to create a process state transformation model that can detect anomalies in enterprise applications and detect unknown threats or zero-day threats. Dataset used the implementation was gathered from the gas pipeline network system. As for performance metrics, F1-score recall, precision, and accuracy have been investigated. The system results also compared with the results by using Support Vector Machine (SVM), Random Forest (RF), Hidden Markov Model(HMM) models, and Bayesian Network (BN).

In [40], the paper Implement a convolution hybrid framework that models dispersion of discrete series of events obtained through preprocessing the log files, and the probability is directly generated of data which belong to the standard data. The framework consists of two stages, first train a convolution autoencoder for learning Image representation of the encoded one-hot image and then train variational autoencoder, which produces the probability for the given image contain anomaly pattern. During the evaluation precision and recall has been measured in term of performance metrics. Framework implementation result also compared with those related to supervised CNN classification and convolution autoencoder(CAE).

In [41], the authors propose an anomaly detection system that can recognize anomalies based on computer-generated log analysis. They consider normal log sequences as a language modeling problem to learn the structure of a language by constructing a model that can be used in previous tokens and predicting how the next token will be based upon these previous tokens. During the detection process, they continuously apply the model to predict the next log key to be predicted. If the forthcoming next log key tends to agree with model prediction, it will be labeled as normal, or it will be regarded as an anomaly. In the evaluation section, authors use HDFS as a dataset while measuring precision, recall, F-Measure, and FPR as performance metrics. Finally, the authors compare the system results with results by using a multi-layer perceptron autoencoder and K-Means.

In [42], authors develop an anomaly detection scheme based on the GCN (graphic convolution networks) that can detect insider danger and prevent fraud. GCN is a CNN extension in the graph domain. They classify the activities of users and their relation to a graph and then train anomaly detection models for insider threats and fraud detection using the GCN algorithms. They conceive a weighted function that utilizes the associations between the consumers and the similarity of the actions in quantifying systemic knowledge Networking. To implement the proposed system, CMU CERT v4.2 dataset has been used while accuracy, precision, recall were calculated as performance metrics. The work also provides a result comparison with SVM random forest, CNN, and logistic regression.

In [43], the work includes implementing a supervised deep-learning classifier with a traditional neural network architecture (CNN). CNN is adopted to distinguish normal and abnormal actions of network users for the detection of attacks. In this paper, the authors state that synaptic weights and, consequently, deep learning classification efficiency are highly dependent on the optimization algorithm. NSL-KDD dataset was utilized in the implementation part, and a new metric named "Stability" has been measured as performance indicators.

V. Solutions Analysis

In this section, the survey provides in depth-analysis to the 21 article papers discussed in the previous section. Based on the two aspects, an in-depth analysis has been done. The first aspect starts by analyzing keywords in each article. Keywords analysis highlights the significant aspects of each solution in terms of domains, techniques, and goals. Keywords analysis starts by collecting all keywords and then apply some modification and normalization.The result is shown in figure 3. The second aspect, perform mechanism analysis on previous solutions, shows that these mechanisms consist mainly of 4 processing stages: input, input preprocessing, classification, and output. To combine the results of the two analyses mentioned above, this survey suggests a framework called a unified cross framework. The unified cross framework allows an in-depth analysis of each stage of each model presented in the previous solutions. The unified cross model consists of 4 phases, input phase, input preprocessing phase, classification phase, and output phase, as shown in figure 4. Each phase of framework phases has specific tasks; executing these tasks usually require to adopt one or more techniques. Table III presents the techniques used in the previous solutions by mapping them with the suggested unified cross framework. Furthermore, the explanation of each phase is provided in the next subsections.

A. Input Phase

The solutions that have been analyzed show that these solutions developed to handle various types of data inputs that supposed to contain one anomaly type of anomalies or more. Usually, the input data are formulated as traffic data, and these records contain information about networks such as the protocols used. Another type of input data is the system log records; these records monitor system status such as user activity. Vehicle sensor data has been conducted in [26] as another type of inputs. Some solutions that have conducted to work in smart grids domains depend on voltage measurement utilized as input data in [28]. Regarding the implementation, these input data usually need to be stored as a suitable dataset to enhance the learning process in the following phases. Some papers depend on ready datasets to investigate the efficiency of their models while others collect and construct their datasets. Most of the input data have a form or structure that cannot be processed directly by CNN; these raw data need to be cleaned and prepared to be used as a suitable input for CNN in order to detect anomalies within these inputs.

B. Input preprocessing Phase

This phase's main aim is to feed the next phase with suitable input data, usually as a matrix with 2D dimensions. Machine learning and deep learning models depend on a specific

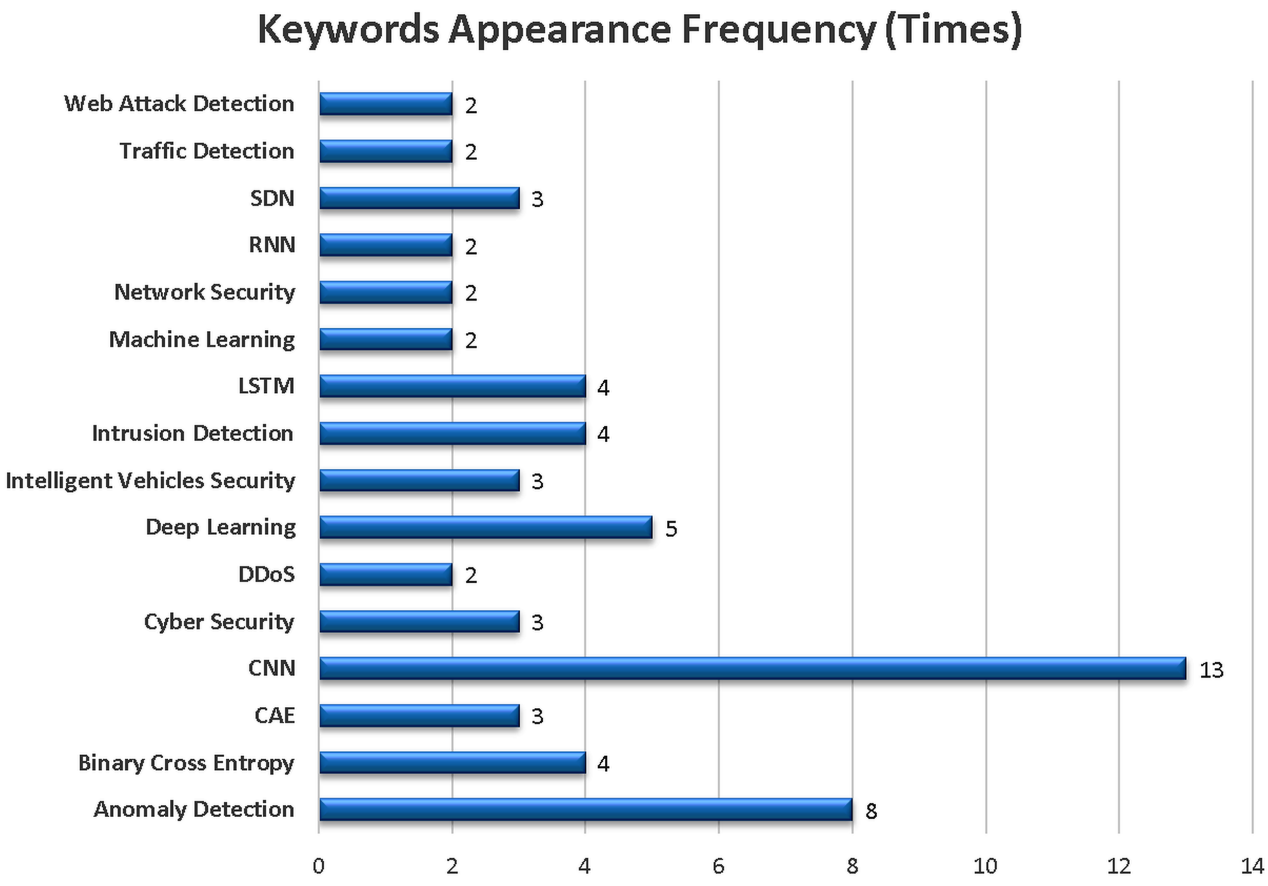

Fig. 3. Frequency of Keywords in the Solutions

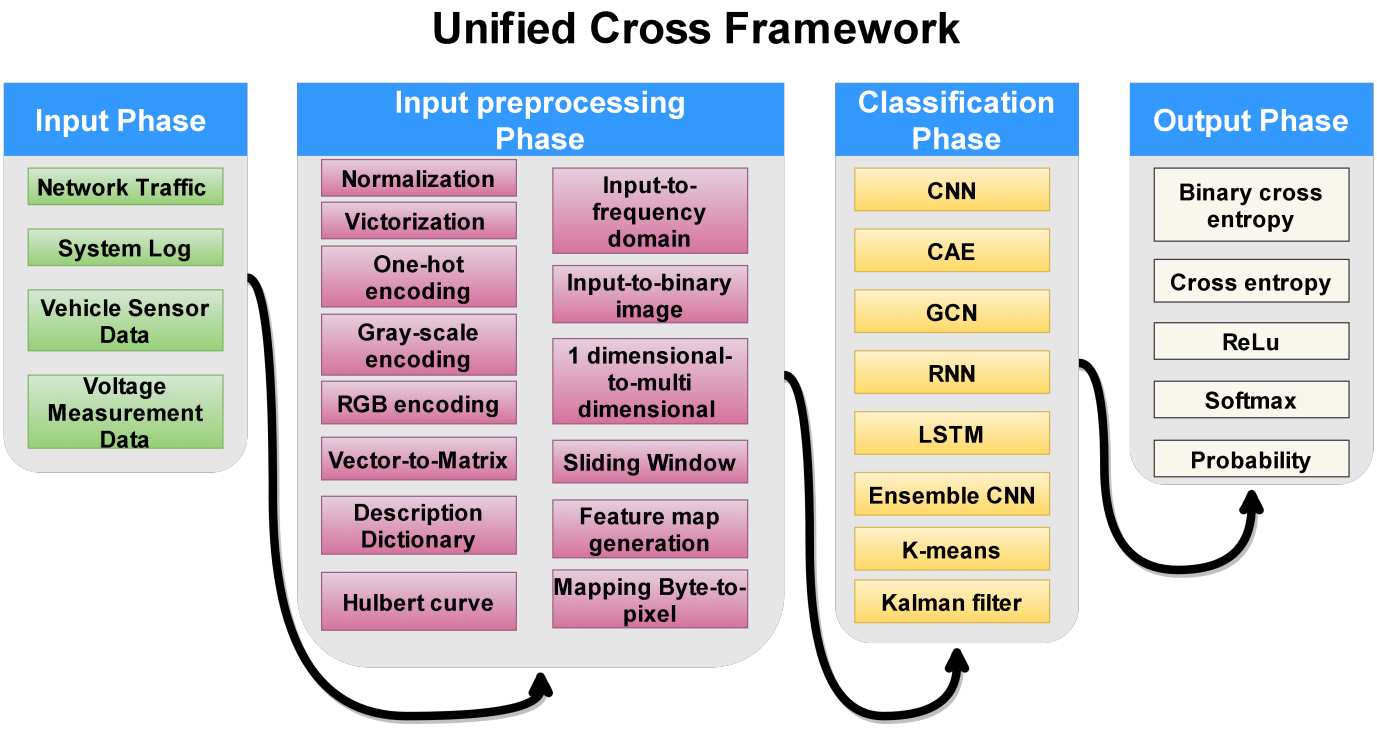

Fig. 4. Unified Cross Framework

TABLE III
SOLUTIONS TECHNIQUES ACCORDING TO THE UNIFIED CROSS FRAMEWORK

Phase & Techneques		Reference	Phase & Techneques		Reference
Phase 1 **(Input Phase)**	Network Traffic	[23] [24] [25] [27] [29] [30] [31] [32] [33] [35] [36] [37] [38]	**Phase 3** **(Classification)**	CNN	[24] [25] [26] [39] [27] [28] [30] [31] [32] [33] [34] [35] [41] [36] [43] [38]
	Log Records	[44] [40] [41] [42] [43]		Ensemble CNN	[29] [37]
	Vehicle Sensor Data	[26]		LSTM	[28] [30] [33] [36] [38]
	Power Traffic Data	[34]		CAE	[23] [40]
Phase 2 **(Input Preprocessing)**	Normalization	[24] [29] [43]		GCN	[42]
	One-Hot Encoding	[24]		K-Means	[25]
	Binary-to-Image	[23]		Kalman Filter	[26]
	Gray-Scale Encoding	[24]	**Phase 4** **(Output)**	Cross Entropy	[24] [30] [31] [38] [42]
	RGB Encoding	[24] [27]		Binary Cross Entropy	[23] [40] [26]
	Naive Assumption	[25]		ReLu	[26] [33] [36] [37]
	Input-to-Frequency Domain	[25]		SoftMax	[39] [41] [42]
	1D-to-2D Projection	[25]			
	Sliding Window	[26]			
	Victorization	[39] [40] [35] [37] [38] [28] [33] [41] [36] [29] [43]		Probability	[27]
	Vector-to-Matrix	[39] [40] [35] [37] [38] [29] [43]			
	Discription Dictionary	[30]			
	Feature Map Generation	[31]			
	Hulbert Curve	[31]			
	Byte-to-Pixel	[32]			

type of data input. This concern becomes more challenging when CNN is used, because CNN usually works with image classification that needs the input to be a matrix of pixel image or array of data. To overcome this problem, researchers depend on many techniques to process the input data and convert them into an appropriate format suitable for CNN. Most of the techniques used are listed in Table III. .Normalization of input data is among the techniques that highly used by the author as a first step; normalization is the process wherein data within an input is cleaned and reorganized in such a way that the later algorithm can adequately utilize them. Vector-to-Matrix is another essential preprocessing techniques which used to convert that the input data to form like a matrix of data which ease the task of CNN.

Choosing the preprocessing techniques depends on two things, type of input in the first phase and organization of the classification network in the later phase, furthermore some models require establishing more than one technique.

C. Classification Phase

This phase is the core of anomaly detection models that have analyzed. At this phase, CNN utilized to extract features and detect anomaly points in the input data. However, some solution adds some hierarchy organization to their proposed solution by combing CNN other algorithms or techniques. RNN or LSTM is the most common technique that can enhance anomaly detection efficiency; these techniques have been used in many models such as those in [18] and [19]. The models that invest RNN or LSTM suggest a specific mechanism; this mechanism starts by feeding the input data to the CNN to extract the feature; next, the output of CNN will be input to the RNN or LSTM to learn the feature and detect anomalies data point. Ensemble organizations also adopted in some models where there are more than CNN in architecture that works in parallel or sequence [29] [37]. Graph convolution network (GCN), which is the graph version of CNN [45], is adopted in [42]. In contrast, convolution auto-encoder (CAE) [46] has been used to provide extraction of input features in [23] [40] .

The proposed model can be supported with different techniques beside rather than deep learning methods, in [25], CNN was supported with K-means [47] while in [26] Kalman filter [48] has been used along with CNN to provide a more accurate anomaly detection scheme.

D. Output Phase

Most of the deep learning algorithms use specific techniques and activation functions that enable optimizing the outputs to provide the most accurate one. In terms of error reduction or loss minimizing, gradient descent techniques have been used widely in conjunction with the multiple types of activation functions.CNN models that have been analyzed in this survey adopt several activation functions. Cross entropy (CE) has been used in most models, and some consider the binary version of cross-entropy (BCE) [49]. Activation functions like The rectified linear activation function (ReLu) and SoftMax [50] have been utilized in some solutions in the output phase. A combination approach that used more than one function in the output phase attracts many authors to use it in their suggested solution [39] [42] [26]. Also, few works establish the custom probability method of anomaly existence as a measure in the output phase, such as the solution provided in [27].

The output phase is usually implemented by specific neural network layers that locate after a fully-connected layer of CNN; according to that, the output of CNN can be considered as new input to another deep learning network. The final step in this phase the result of anomaly existence in the input data. In the evaluation section of the analyzed models, the results are usually measured using performance metrics explained previously, such as recall, F1, and precision.

VI. Future Works

CNN has a good impact on the area of developing anomaly detection schemes, as shown in this survey; however, there is a need to fulfill some existing gaps. Solutions and Articles analyzed in this survey recognize some aspects that need to be highlighted in future works, as shown in Table IV.

However, this survey suggests specific perspective regarding future works can be, and these future works can be summarized in the following directions :

1) Models Accuracy still needs further improvements, especially when there are different types of attacks.
2) There is a crucial need to develop a detection model that can act in real-time without the need for offline training.
3) Develop a flexible scheme that can be adapted well when necessary to modify the network status.
4) Most of the current models ignore many parameters in the input preprocessing, which could negatively affect anomaly detection efficiency.
5) Input preprocessing requires more effort and conduct other techniques and especially data fusion methods.
6) CNN anomaly detection based on prediction is one of the approaches that can increase the robustness of detection mechanisms.

VII. Conclusion

This survey presents a comprehensive analysis of the use of a convolution neural network (CNN) as a core for anomaly detection solutions. Many of the established studies in the literature were collected and categorized according to the input data source. In order to provide convenient analysis, this paper proposes a framework called a unified cross framework. All the collected solutions have been analyzed under this framework. The analysis section in this survey applies the cross-checking process between the collected solutions and unified cross framework allows for understanding how CNN enforced in these solutions in order to provide anomaly detection. The use of the mentioned framework highlights most of the techniques and ecosystems that work together to execute anomaly detection tasks. Finally, this paper proposes some potential trends for research, which, in this sense, will assist the audience in their future works, such as the need for real-time models and high accurate systems.

References

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, 2009.

[2] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.

[3] F. Samie, L. Bauer, and J. Henkel, "From cloud down to things: An overview of machine learning in internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4921–4934, 2019.

[4] F. Restuccia, S. D'Oro, and T. Melodia, "Securing the Internet of Things in the Age of Machine Learning and Software-Defined Networking," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4829–4842, 2018.

[5] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. Leung, "A survey on security threats and defensive techniques of machine learning: A data driven view," *IEEE Access*, vol. 6, pp. 12 103–12 117, 2018.

[6] A. Handa, A. Sharma, and S. K. Shukla, "Machine learning in cybersecurity: A review," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 9, no. 4, 2019.

[7] B. Ahmad, W. Jian, and Z. Anwar Ali, "Role of Machine Learning and Data Mining in Internet Security: Standing State with Future Directions," *Journal of Computer Networks and Communications*, vol. 2018, 2018.

[8] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine Learning and Deep Learning Methods for Cybersecurity," *IEEE Access*, vol. 6, pp. 35 365–35 381, 2018.

[9] P. Sarao, "Machine learning and deep learning techniques on wireless networks," *International Journal of Engineering Research and Technology*, vol. 12, no. 3, pp. 311–320, 2019.

[10] S. Chandran and K. Senthil Kumar, "A survey of intrusion detection techniques," *International Journal of Engineering & Technology*, vol. 7, no. 2.4, p. 187, 2018.

[11] M. Fahim and A. Sillitti, "Anomaly Detection, Analysis and Prediction Techniques in IoT Environment: A Systematic Literature Review," *IEEE Access*, vol. 7, pp. 81 664–81 681, 2019.

[12] B. Sharma, L. Sharma, and C. Lal, "Anomaly Detection Techniques using Deep Learning in IoT: A Survey," *Proceedings of 2019 International Conference on Computational Intelligence and Knowledge Economy, ICCIKE 2019*, pp. 146–149, 2019.

[13] Z. M. Fadlullah, F. Tang, B. Mao, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 4, pp. 2432–2455, 2017.

[14] M. Nooribakhsh and M. Mollamotalebi, "A review on statistical approaches for anomaly detection in DDoS attacks," *Information Security Journal*, vol. 29, no. 3, pp. 118–133, 2020.

[15] T. Pourhabibi, K. L. Ong, B. H. Kam, and Y. L. Boo, "Fraud detection: A systematic literature review of graph-based anomaly detection approaches," *Decision Support Systems*, vol. 133, 2020.

TABLE IV
FUTURE WORKS SUMMARY

Reference	Future Work
[24]	Result need optimization in terms of accuracy
[25]	Study of non-square CNN method filters to be more effective and take advantage of the time-based 2D image functions
[26]	Add some enhancement to enforce identifying the type of anomaly which will allow for other real-time innovations actions intended to mitigate the effects of cyber-attacks
[39]	Investigate the usability of extraction of the deep Learning Features in the detection process of a state transition model
[27]	Expand malware and anomaly detection efforts, especially for cloud environments
[29]	Focus on the use of deep learning techniques with ensemble to provide robust anomaly detection schemes
[30]	Model assessments using real world network traffic
[31]	Develop a model which can be modifiable in response to the network situation
[32]	Create multi-serial network traffic using recurrent neural network from a wider variety of network traffic types
[40]	Develop a more sophisticated algorithm which would take into account certain variable system parameters and continuous events
[33]	Trying to add more malicious traffic which allow increasing model generalization
[35]	Taking advantage of deep learning to develop successful online anomaly detection systems without substantial delay in detection.
[41]	Use of proposed model to extended log-data types and module incorporation to help users identify any known abnormalities more easily
[36]	Improve performance detection with enhanced dataset as well as actual network traffic
[42]	Develop Graph CNN models in real-world applications
[38]	Framework need to be applied in a practical web service scenario
[11]	Fuse the perceptual data streams which will help analyzing suspicious activities
[12]	Instant data streaming in IoT requires real-time prediction of anomalies
[14]	Develop DDoS detection for the cloud computing
[6]	Provide more intelligent defense systems for protecting machine learning model itself from adversarial attacks

[16] R. A. Ariyaluran Habeeb, F. Nasaruddin, A. Gani, I. A. Targio Hashem, E. Ahmed, and M. Imran, "Real-time big data processing for anomaly detection: A Survey," *International Journal of Information Management*, vol. 45, pp. 289–307, 2019.

[17] J. Schmidhuber, "Deep Learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[18] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[19] S. Hochreiter and J. Urgen Schmidhuber, "Long Shortterm Memory," *Neural Computation*, vol. 9, no. 8, p. 17351780, 1997.

[20] J. F. Kolen and S. C. Kremer, "Gradient Flow in Recurrent Nets: The Difficulty of Learning LongTerm Dependencies," *A Field Guide to Dynamical Recurrent Networks*, 2010.

[21] A. D. Nguyen, S. Choi, W. Kim, S. Ahn, J. Kim, and S. Lee, "Distribution Padding in Convolutional Neural Networks," *Proceedings - International Conference on Image Processing, ICIP*, vol. 2019-Septe, pp. 4275–4279, 2019.

[22] T. Sapatinas, "The Elements of Statistical Learning," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, vol. 167, no. 1, pp. 192–192, 2004.

[23] S. Park, M. Kim, and S. Lee, "Anomaly detection for HTTP using convolutional autoencoders," *IEEE Access*, vol. 6, pp. 70 884–70 901, 2018.

[24] T. Kim, S. C. Suh, H. Kim, J. Kim, and J. Kim, "An Encoding Technique for CNN-based Network Anomaly Detection," *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, pp. 2960–2965, 2019.

[25] M. R. Moore and J. M. Vann, "Anomaly Detection of Cyber Physical Network Data Using 2D Images," *2019 IEEE International Conference on Consumer Electronics, ICCE 2019*, 2019.

[26] F. Van Wyk, Y. Wang, A. Khojandi, and N. Masoud, "Real-time sensor anomaly detection and identification in automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1264–1276, 2020.

[27] N. K. Sahil Gar , Kuljeet Kaur and A. R. R. Georges Kaddoum, "A Hybrid Deep Learning-Based Model for Anomaly Detection in Cloud Datacenter Networks," *924 IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT*, vol. 16, no. 3, 2019.

[28] X. Niu, J. Li, J. Sun, and K. Tomsovic, "Dynamic Detection of False Data Injection Attack in Smart Grid using Deep Learning," *2019 IEEE Power and Energy Society Innovative Smart Grid Technologies Conference, ISGT 2019*, 2019.

[29] S. Haider, A. Akhunzada, G. Ahmed, and M. Raza, "Deep Learning based Ensemble Convolutional Neural Network Solution for Distributed Denial of Service Detection in SDNs," *2019 UK/China Emerging Technologies, UCET 2019*, 2019.

[30] Y. Qin, J. Wei, and W. Yang, "Deep Learning Based Anomaly Detection Scheme in Software-Defined Networking," *2019 20th Asia-Pacific Network Operations and Management Symposium: Management in a Cyber-Physical World, APNOMS 2019*, 2019.

[31] Y. Sun, H. Esaki, and H. Ochiai, "Detection and Classification of Network Events in LAN Using CNN," *Proceedings of 2019 4th International Conference on Information Technology: Encompassing Intelligent Technology and Innovation Towards the New Era of Human Life, InCIT 2019*, pp. 203–207, 2019.

[32] A. Cheng, "PAC-GAN: Packet Generation of Network Traffic using Generative Adversarial Networks," *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON 2019*, pp. 728–734, 2019.

[33] J. Xie, S. Li, Y. Zhang, X. Yun, and J. Li, "A Method Based on Hierarchical Spatiotemporal Features for Trojan Traffic Detection," *2019 IEEE 38th International Performance Computing and Communications Conference, IPCCC 2019*, 2019.

[34] S. Fatemifar, M. Awais, S. R. Arashloo, and J. Kittler, "Combining Multiple one-class Classifiers for Anomaly based Face Spoofing Attack Detection," *2019 International Conference on Biometrics, ICB 2019*, 2019.

[35] R. H. Hwang, M. C. Peng, C. W. Huang, P. C. Lin, and V. L. Nguyen, "An Unsupervised Deep Learning Model for Early Network Traffic Anomaly Detection," *IEEE Access*, vol. 8, pp. 30 387–30 399, 2020.

[36] S. J. Ryu, W. Go, D. Lee, and H. J. Yoon, "Hierarchical neural networks for detecting anomalous traffic flows," *2019 IEEE Global Communications Conference, GLOBECOM 2019 - Proceedings*, 2019.

[37] S. Haider, A. Akhunzada, I. Mustafa, T. B. Patel, A. Fernandez, K. K. R. Choo, and J. Iqbal, "A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks," *IEEE Access*, vol. 8, pp. 53 972–53 983, 2020.

[38] X. Gong, J. Lu, Y. Wang, H. Qiu, R. He, and M. Qiu, "CECoR-Net: A Character-Level Neural Network Model for Web Attack Detection," pp. 98–103, 2020.

[39] J. Liu, L. Yin, Y. Hu, S. Lv, and L. Sun, "A Novel Intrusion Detection Algorithm for Industrial Control Systems Based on CNN and Process State Transition," *2018 IEEE 37th International Performance Computing and Communications Conference, IPCCC 2018*, 2018.

[40] A. Wadekar, T. Gupta, R. Vijan, and F. Kazi, "Hybrid CAE-VAE for Unsupervised Anomaly Detection in Log File Systems," *2019 10th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2019*, 2019.

[41] S. Yen, M. Moh, and T. S. Moh, "CausalConvLSTM: Semi-supervised

log anomaly detection through sequence modeling," *Proceedings - 18th IEEE International Conference on Machine Learning and Applications, ICMLA 2019*, pp. 1334–1341, 2019.

[42] J. Jiang, J. Chen, T. Gu, K. K. R. Choo, C. Liu, M. Yu, W. Huang, and P. Mohapatra, "Anomaly Detection with Graph Convolutional Networks for Insider Threat and Fraud Detection," *Proceedings - IEEE Military Communications Conference MILCOM*, vol. 2019-Novem, 2019.

[43] K. Bennaceur, Z. Sahraoui, A. Labed, and M. Ahmed-Nacer, "Training Function Stability in Anomaly Intrusion Detection based Deep Learning," *2019 International Conference on Control, Automation and Diagnosis, ICCAD 2019 - Proceedings*, 2019.

[44] Y. F. A. Gaus, N. Bhowmik, S. Akcay, P. M. Guillen-Garcia, J. W. Barker, and T. P. Breckon, "Evaluation of a Dual Convolutional Neural Network Architecture for Object-wise Anomaly Detection in Cluttered X-ray Security Imagery," *Proceedings of the International Joint Conference on Neural Networks*, vol. 2019-July, 2019.

[45] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, 2019.

[46] Z. Zhang, D. Chen, Z. Wang, H. Li, L. Bai, and E. R. Hancock, "Depth-based subgraph convolutional auto-encoder for network representation learning," *Pattern Recognition*, vol. 90, pp. 363–376, 2019.

[47] J. A. Hartigan, "Clustering Algorithms John Wiley & Sons," *Inc., New York, NY*, 1975.

[48] P. Zarchan and H. Musoff, "Fundamentals of Kalman Filtering: A Practical Approach," *Virginia, Published by the American Institute of* ..., p. 852, 2009.

[49] P. T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of Operations Research*, vol. 134, no. 1, pp. 19–67, 2005.

[50] "Searching for activation functions," *6th International Conference on Learning Representations, ICLR 2018 - Workshop Track Proceedings*, 2018.