Presentation of a New Method for Intrusion Detection by using Deep Learning in Network

Hui MA*
Modern Educational and Technological Center, Henan Quality Institute Pingdingshan, Henan, 467000, China

***Abstract*—Intrusion detection in cyberspace is an important field for today's research on the scope of the security of computer networks. The purpose of designing and implementing the systems of intrusion detection is to accurately categorize the virtual users, the hackers and the network intruders based on their normal or abnormal behavior. Due to the significant increase in the volume of the exchanged data in cyberspace, the identification and the reduction of inappropriate data characteristics will play a significant role in the increment of accuracy and speed of intrusion detection systems. The most advanced systems for intrusion detection are designed for the detection of an attack with the inspection of the full data of an attack. It means that a system of detection will be able to recognize the attack only after the execution of the attack on the attacked computer. In this paper, a system for end-to-end early intrusion detection is presented for the prevention of attacks on the network before these attacks cause further detriment to the system. The proposed method uses a classifier based on the network of the deep neural for the detection of an attack. The proposed network on a supervised method is trained for the exploitation of the related features by the raw data of the traffic of the network. Experimentally, the proposed approach has been evaluated on the dataset of NSL-KDD. The extensive experiments show that the presented approach performs better than the advanced approaches based on the accuracy, the rate of detection and the rate of the false positive, and also, the proposed system betters the rate of detection for the classes of the minority.**

Keywords*—Attack; security on cyberspace; classification; intrusion detection; deep learning

I. Introduction

The ever-increasing expansion of the Internet, in terms of infrastructure and in terms of software, has caused an increment in the number of network users' number and their applications [1]. Today, many public sector services and many private sector services are virtually done on the Internet. The development of this virtual space has caused the detection of intrusion to become the most important subject in the scope of the security of computer networks. The systems for intrusion detection try to classify the activity of the made connections by the users into two categories: the normal and the abnormal [2]. In more advanced systems, sometimes, the type of abnormal behavior, which is also called an attack, is specified. Each connection in the network is described based on the collection of features, which these features can be used for the determine of the normal connection or the abnormal connection [3], [4].

The accuracy of the detection for the systems of intrusion detection is the most important indicator of the efficiency of these systems. The increment of the accuracy in the systems of the detection of the intrusion prevents the result of more attacks in the system. The attack neutralization will play a decisive role in the reduction of the costs caused by the attacks on valuable network resources [5]. The attacks of the Denial of Service (DoS) are the most popular kind of attacks that are constantly sent to servers by different IP addresses and prevent the serving of the servers or shut down of the servers [6]. Usually, the attacker varies the address of the IP to carry out an attack on the victim's computer. Although the network firewall that first responds to the attack performs the process of filtering the attack, unfortunately, today, this amount of security is not enough. Then, the attacks that can elapse the firewall are sent through an intermediate router into the deep learning-based approach to the detection of the intrusion. Here, the multitude of variant classes of new attacks can be identified. If an unusual position is observed, it is immediately related to the center of the management of the system. Usually, this center informs the system officers about this situation via SMS or email, and then it automatically blocks this attack [7], [8].

It is essential to provide a more precise method of the detection of attack for the detection of novel attacks in the servers and the networks. The detection methods do not use the programming of the unequivocal according to problem complexity [9]. Usually, they use the methods of machine learning, which can be remarkably prosperous on the problems of decision-making as long as the features of the sufficient are presented on problem [9], [10]. In deep learning, incumbent features are also exploited by training certain layers like Recurrent Neural Networks (RNNs) or Convolutional neural networks (CNNs). The achievement of an extended network of deep learning strongly appertains the kind of layers of feature extraction. Also, it depends on sufficiently large training dataset. In addition, the preprocessing and organization of input data may have an important result in achievement [11]. Thus, hunt for superior methods in different fields is a topic that is studied by researchers. In current paper, a system for end-to-end early intrusion detection is presented for prevention of attacks on network before these attacks cause further detriment to the system. The proposed method uses a classifier based on the network of the deep neural for the detection of an attack. The proposed network on a supervised method is trained for the exploitation of the related features by the raw data of the traffic of the network.

In summary, the contributions and the reasons for choosing the methods of this paper are as follows: 1) A basic network intrusion detection method is presented. If more data is available, the proposed approach can make a more informed decision about the class label of a given network traffic data,

but it delays the decision. Therefore, in this work, the focus is on optimizing the attack detection accuracy with minimum delay. 2) The proposed approach extracts relevant features from raw network traffic data end-to-end instead of relying on manual feature engineering process. Therefore, the proposed approach is domain-independent and does not require domain-specific data preprocessing steps. 3) A new metric is introduced to evaluate how early the proposed approach can detect attacks.

The continuation of this article is as follows: Section II provides a review of the research literature. Section III the proposed method, which uses the model of deep learning, is presented. In Section IV, the dataset used in the designed experiments details and the outcomes of these experiments are presented. In Section V, the general conclusions and the prospects for future research are presented.

II. Review of Literature

The system of intrusion detection in the network is applied for the monitoring of the network traffic to protect the system from network threats. This system tries to find the destructive acting like the attacks of DoS, the attacks that monitor the network traffic, and the port scanning attacks [12]. In general, two methods are used to detect the intrusion on the network:

A. Detection of Anomaly

It is when the observed behavior of the user does not follow an expected behavior [12]–[14]. In network anomaly detection, the normal system activities such as the network bandwidth, the ports, the rules and the device communication are examined. The detection of the anomaly of the intrusion is a hard problem with several proofs. First, the use of the system and user behavior is constantly evolving. Therefore, the intrusion detector must also evolve. Without the permission for these changes in the behavior, the network administrator will soon be inundated with false alarms, and it will rapidly affect the trust in the system. A second important factor in the detection of the anomaly is that an alert of the behavior of the abnormal may not furnish any specific beneficial data for the administrator of the network. The ambiguous alert about which the system may be beneath the attack makes it hard to catch the firm's measured action. Also, sometimes, the known attack may not be recognized by the system of the detection of the anomaly. This is the fundamental problem [12]– [14].

B. Signature-based Detection

It is when the observed behavior indicates an intention for the misuse of the network computing resources. The benefits of the detection based on the signature consist of simpleness and effectiveness, and it has a great ability for detecting the attacks of the known. The other important profit is that the alert that is published is a certain alert because it reconciles the signature of the pattern of a certain attack. By a special alert, the administrator of the network can rapidly assess whether the attack of the doubtful is a wrong alert or real. I.e., if it is detected as real, the network administrator can adopt an appropriate response. Another advantage of this approach is the ability to produce accurate results and reduce false alarms. However, a disadvantage of the system of detection based on signature is that the signature file must be updated. Also, with the increment of the number of signatures, the efficiency of the system is decreased. On the other hand, the system will be able to recognize the attacks of the known. The smallest change in the known attacks causes the possible loss of the attack detection by the system [15]–[17].

In the literature, there are different researches for IDS with the use of the methods of machine learning. Here, the latest articles based on the deep learning are presented. In [18], the authors have presented a hybrid method based on the networks of the deep belief and the networks of the probabilistic neural. The authors deal with the imbalance of the class in the NSL-KDD dataset with the use of SMOTE about the increment of the classes of the minority and with the use of NCL about the classes of the majority of the under-sampling. In [19], the authors have presented a technique for the extraction of the feature with the use of the sparse auto-encoder. Their presented system is contrasted to the prior methods of the extraction of the feature. The process of the classification is speeded up, and a superior process for the learning is acceded. An impressive practical model has been developed for use in the systems of the detection of intrusion.

In study [20], the authors have proposed a revision of the literature on machine learning and the applications of deep learning in the detection of intrusion on the network. Also, the authors have appraised the different databases, the different approaches and the different problems with the detection of the intrusion. In research [21], the authors have provided the models based on the optimization of the particle swarm for the selection of the feature and for the meta-parameter selection. They have evaluated the different models of deep learning, like the Long Short-Term Memory (LSTM) and the Deep belief network (DBN). In study [22], the researchers have provided a review of research about deep learning for the IDS. They have also reviewed the light research and the directions for the future. The authors provide an approach for the architectures of deep learning for the kinds of IDS and for the databases. In research [23], the authors have proposed the categorization of the models of deep learning for the systems of the detection of intrusion. Also, they have presented the literature review in their research. The training and the test have been performed on the various databases with the use of $four$ methods of deep learning. Their test outcomes are compared by articles in the literature. Also, the assessments for the latter research about the systems of the detection of the intrusion, which are based on deep learning, are provided.

The authors have provided an approach to the detection of the attack based on deep learning about the attacks of Distrubed Denial of Service (DDOS). Their presented approach has been evaluated in the dataset of CICIDS. Also, it has been simulated the traffic of the DDOS. It is expressed that the presented approach outperforms the several prior approaches for the detection of the attacks of the DDOS [24]. In [25], the authors have provided an asymmetric deep auto-encoder classification method for the unsupervised learning of the features. Their proposed method is evaluated on GPU with the use of the dataset of KDD-CUP99 and the dataset of NSL-KDD. The obtained outcomes have been contrasted by the research on literature. In study [26], the researchers have extended a model of the deep network, which consists of the

RNNs with the units of the recurrent of the gated, the Multi Layer Perceprton (MLP) and the modules of the softmax for an increment of the efficiency of the systems of the detection of the intrusion. Their experiments on the presented model have been performed in the dataset of KDD-99 and the dataset of NSL-KDD. The outcomes of the experiments have displayed that the GRU has superior outcomes over LSTM in the systems of the detection of intrusion. In study [27], the authors have extended an IDS basis on self-learning according to the framework of STL to propose the superior security of the network over the standard technologies for the defense of the network. The experiments in the dataset of NSL-KDD incur accuracy of the classification of binary and the accuracy of the five-class classification. Also, their experiments reduce the time of the training and the time of the test.

In research [28], the authors have provided an approach based on the cloud for the detection of the intrusion in the network in real-time with the use of the binomial deep learning models and the models of the 5-class deep learning along with the framework of H2O. In case of an attack, the models can dispatch a notification to the mobile to the model authorities with the help of a web page on the architecture basis on the cloud, which the authors are planning. In study [29], the authors have proposed a model of IDS based on the improved DBN by the algorithms of the genetic for IoTs. The structure of the network of the optimal with GA is characterized by the use of numerous iterations. In study [30], the researchers have investigated a model of DNN by the different layers. Also, authors have used the NSL-KDD dataset, the UNSW-NB15 dataset, the Kyoto dataset, the WSN-DS dataset and the CICIDS 2017 dataset for NIDS and HIDS. Due to the done tests, this approach can be scouted on the time of the real, and it has superior outcomes over the traditional algorithms of machine learning. In research [31], the researchers have extended an approach based on DBN by a layer of the classification with four-layer and Support Vector Machine (SVM). These authors have done the experiments using different kernels like Radial Basis Function (RBF), the linear, the sigmoid and the polynomial. Due to the empirical outcomes in the dataset of NSL-KDD, authors have obtained the foremost outcomes with the use of the kernel of RBF.

III. The Presented Method

In the current section, the proposed system for the detection of intrusion and the detection of network attacks is presented. The primary purpose of the presented method is the monitoring of the traffic of the network on the time of the real, the extraction of the automatic features from the raw data of the traffic of the network, the prevention of the time-consuming task of the feature extraction using the traditional methods and the accurate detection of the attacks in the network. The presented method can be intersected into two general stages. The overview of the presented method is displayed in Fig. 1. In the first step, the proposed flow classifier is trained and then evaluated using a dataset that has the labeled flows of the network and the related packets of the network into these flows. In the second step, the trained classifier of the flow is used for the prediction of whether a given flow from the network is destructive or the usual. The flow of the network is the two-way trail of the packets, which is swapped among two endpoints in a specified interval of time by several joint features of the flow [9], such as the addresses of the IP of the source, the addresses of the IP of the destination, the numbers of the port of the source, the numbers of the port of the destination and the protocol kind. In the proposed method, a flow of the network is defined as a trail from the regular packets T. In it, T displays the longitude of a full flow. The given flow is shown as follows:

$$F_T = \{P_1.P_2.\cdots.P_T\}\ \forall P_i \in \mathbb{R}^d \wedge 1 \leq i \leq T \quad (1)$$

d is the packet length. Two main steps of the proposed method use a flow processing approach, which this approach includes three modules: the filtering of the packet, the identification of the flow and the preprocessing of the packet. The module of the filtering of the packet takes the packets of the network and then sends them into the latter modules if these packets meet certain criteria. The next modules convert the packets, and then these modules categorize them in the flows of the network. When a flow of the network is updated by a novel packet, then the classifier of the flow is used for the updation of the related prediction to the flow.

A. Approach of the Processing of the Flow

In the current section, the modules of the approach of the designed flow processing are introduced. As mentioned in the previous section, in this approach, three modules are used, which are as follows:

1) Filtering of packet: The traffic of the raw of the network among the unreliable network and the attacked system is monitored. Just the packets from the network are selected that are relevant to the kind of attacks that have to recognize them. For instance, if the goal is the detection of web attacks [10], then we only capture the packets of HTTP.

2) Identification of flow: With the reception of a novel packet, the model checks the packet features, like the addresses of the IP of the source and the addresses of the IP of the destination, for the identification of a flow of the appropriate activity for it. The flow of the active displays a session of communication between two endpoints of the network. Also, if no active flow is found that matches the features of the packet, then a novel flow is created. A flow of the network is presumed to be inactive or terminated. After that, the connection is lost or the time at which the flow has not embraced a novel packet on a specified time period. The value of the timeout of the flow can be set due to the kind of protocol of the traffic of the network, which is adapted for the detection of attacks.

3) Preprocessing of packet: When the suitable flow of the novel packets is identified, then every packet is sent via the below preprocessing stages to truncate the unfavorable data and to convert it to the byte vector with a uniform size: the truncation step and the transformation step. The primary goal of the mentioned stages is the confidence that the classifier must rely on the related features to classify the flow. The details of these steps are as follows:

a) Truncation step: The packets of the raw received consist of the header of Ethernet. This header contains the data

about the link of the physical. However, this information is worthless for the detection of an attack insomuch it can be faked as easily. Therefore, the header of Ethernet is deleted from the packet. Also, the header of the IP on the packets contains data like the perfect longitude of the packet, the version of the protocol, the addresses of the IP of the source and the addresses of the IP of the destination. The mentioned data is essential for the routing of the packets on the network. This data is considered disjointed for the classifier because it is possible that the classifier starts to emphasize the IP information for the detection of the flows of the attack. Thus, it is deleted from the packets. It permits the classifier to operate consistently if the node's address on the network has varied. Also, it allows the classifier to popularize the learned information from an environment of the network to another environment.

b) Transformation step: At the time of the usage of the neural network for the classification operation, an input with a fixed size is required. In order to uniform the header longitude of the layer of the transport and in order to uniform the payload of the packets, these packets with the zeros to a fixed length are cutted. It should be noted that if the packet's longitude on a flow is limited, then, the flow longitude is not limited, unlike the other proposed methods like the presented approaches [11].

B. Flow Classification

An essential aspect of proper forecasting is the prediction of the attack with enough time to implement the true reaction to the attack as rapidly as possible before it causes further damage to the attacked system. Therefore, to minimize the time of the prediction, a DNN is used as a classifier of the flow insomuch with an increment of the model size, the prediction time is increased because more computations are required. The key problem is that the decrement of the size of the model usually makes the finite power and the little accuracy [32], [33]. To alleviate the mentioned problem, instead of the training of a model of the large complex for the classification of all kinds of attacks, a set from the naive models is trained which in it, every model is trained for the classification with just a subset of the classes of the attack. Several ensemble strategies, like the voting of the majority and the ranking, have been proposed in the literature for the employment of multiple models and the combination of their predictions [34]. However, the discussion of the ensemble strategies is beyond the domain of the current article.

In the current article, the networks of the neural convolutional as one-dimensional [35] are used for the extraction of a good representation of the internal flows of the network and the presentation of it as the input of a network of the fully-connected or the network of the dense. A layer of the softmax [36] is used as the layer of the final of the network for the calculation of the distribution of the probability for the classes of the target. CNNs are used for the extraction of the related features by the data of the input of the network, like the images and the trails. These networks are able to model the dependencies of the spatial and the dependencies of the temporal on the data with the learning of the corresponding filters of the convolution. A convolution layer consists of several convolution filters, and each filter is used for the extraction of a feature of the specific from the data of the input. Therefore, the output of the layer of the convolution is named the map of the feature.

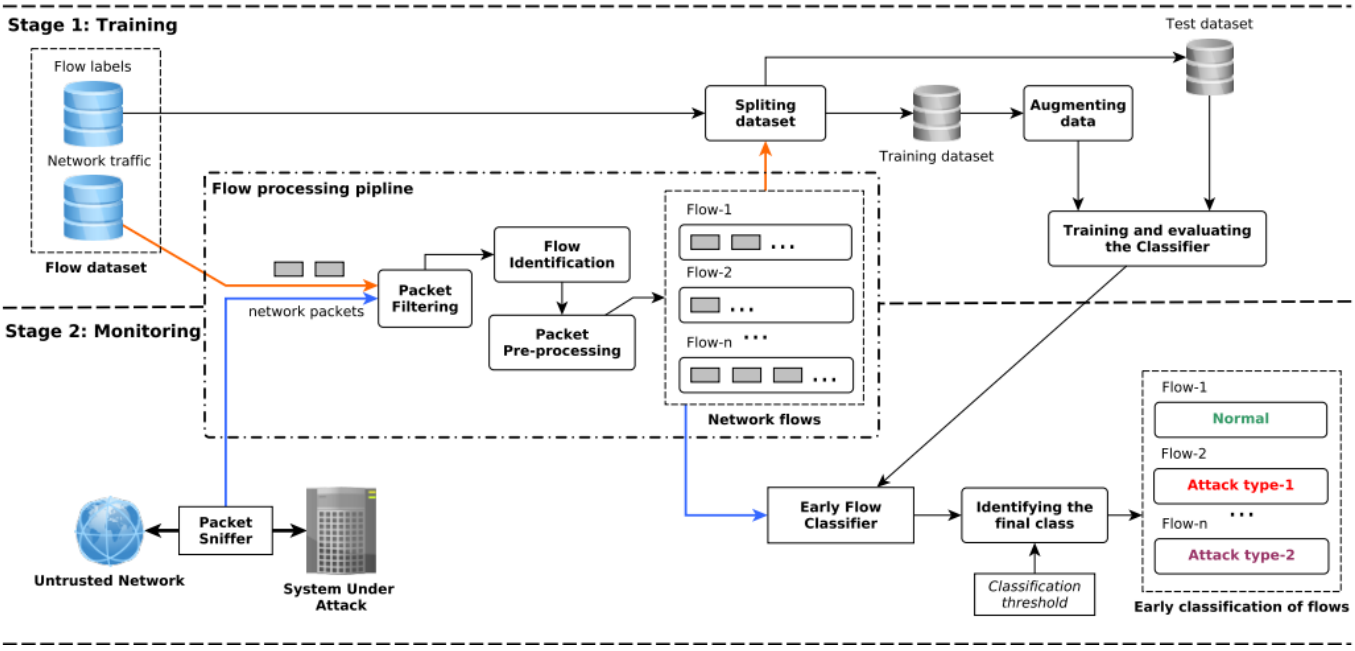

Fig. 1. An overview of the steps of the presented model for network intrusion detection.

The data of the input of ID-CNN has two dimensions. 1-th dimension determines the trail of the circumstances. In contrast, 2 -th dimension is related to the features of the individual from a circumstance. ReLU [37] is used as a function of the activation of the nonlinear in each neuron on the layer of the convolution. Usually, every layer of the convolution is coupled with a layer of the pooling [36] to get the translation-invariance from the returned output with the layer of the convolution. The mentioned layer decreases the size of the time of the output by replacing every partition with the static size by a statistic of the summary from the adjacent elements. CNNs have fewer trainable parameters than the other kinds of ANNs [35]. Therefore, they have fewer possibilities for the overfitting of the data of the training versus the networks of the fully connected. After the convolution operations and the operations of the pooling, a flow of the network with the changeable longitude is displayed with a series of maps of the feature with the changeable longitude. A global layer of the pooling [38] is used for the transformation of the series to a vector with the changeable longitude, and this vector is presented as the input of the layers of the fully connected to obtain the vector of the feature. Finally, a layer of the softmax is applied to the vector of the feature to achieve a distribution of the probability for every class. Based on the distribution of the probability and the basis of the threshold of the classification, the predictions of the final are made.

C. Training

The classifier is trained before its use for the detection of early intrusion online. The system's purpose is to learn automatically the features of the spatial-temporal from the flows of the raw of the network, and then, this system uses these learned features for the reliable identification of the attack flows as quickly as possible. a dataset of the flow labeled is needed for the training of the supervised, which dataset includes the normal flows and the flows of the attack. Moreover, the flows of the labeled and the dataset must contain the corresponding packets of the network with flows. Most of the datasets that are used to train IDSs suffer from the imbalance of the class [2]. That is, the sample number between the various classes in the dataset is not the same. The trained classifier on an unbalanced dataset usually shows the bad efficiency basis the accuracy of the prediction. Thus, in the current article, to correct the effect of the imbalance of the class, the used classifier is trained by the weighting of the sample, which this weighting performs as a factor for the value of the computed loss in every sample among the process of the training. The weight of every sample is based on the related class. Inversely, it is computed with the frequencies of the class in the data of the training of the proportional. The goal is that the classifier must pay further attention to the examples which belong to the classes of the down-represented.

The dataset of the training is prepared with the processing of each packet into flows with the use of the mentioned method above. A dataset of the flow is displayed as $= \{(F_T^{(i)}. y_j)\}$ $1 \le j \le N$. In it, N is the flow number F_T and y shows their labels. As regards the goal, it is the training of a classifier that can reliably detect an attack flow after the view of the first packets from a certain flow. Thus, the dataset is extended with the creation of a stack of the short fragments of a flow with the variable longitudes. The expanding process of the dataset of the training with the generation of more data than the existing data is named the augmentation of the data [39]. The process start with the creation of the smallest fragment from a certain flow that contains just the first packet from the flow. Subsequently, by cumulatively adding more packets, more fragments per flow are created, according to the predefined fragmentation rate s_r such that $0 < s_r < 1$. The fragmentation rate s_r is a meta-parameter which is used for the calculation of the fragment size $s_z = [s_r * T]$ in a certain flow, which in it, T is the flow longitude. The value of this parameter reins the number of the produced fragments in every flow. For example, with the decrement of the value of s_r, the more fragments are generated in each flow.

Presume that a flow is given as $F_T^{(j)} = \{P_1. P_2. \cdots . P_T\}$, the fragments set of this flow is as below:

$$\left\{F_{t=k*s_z}^{(j)} \middle| \, k = 1.2. \cdots . \left\lfloor \frac{T-1}{s_z} \right\rfloor \right. \quad (2)$$

All fragments have the label of a similar y_j which the main flow is needed. For instance, suppose 3 flows by the various longitudes: $F_6^{(1)}$, $F_{15}^{(2)}$, and $F_{70}^{(3)}$. The fragmentation rate s_r is set with 0.25. The fragment sizes s_z for $F_6^{(1)}$, $F_{15}^{(2)}$ and $F_{70}^{(3)}$ are determined 2, 4 and 18. Table 1 catalogs the fragments from the generated flows with the process of the augmentation of the data.

TABLE I. THE FRAGMENTS FROM THE GENERATED FLOWS BY THE DATA AUGMENTATION PROCESS FOR THE MENTIONED EXAMPLE

No.	Flows	Flow Segments
1	$F_6^{(1)} = \{P_1. P_2. \cdots . P_6\}$	$\{P_1. P_2\}$
2		$\{P_1. P_2. \cdots . P_4\}$
3	$F_{15}^{(2)} = \{P_1. P_2. \cdots . P_{15}\}$	$\{P_1. P_2. \cdots . P_4\}$
4		$\{P_1. P_2. \cdots . P_8\}$
5		$\{P_1. P_2. \cdots . P_{12}\}$
6	$F_{70}^{(3)} = \{P_1. P_2. \cdots . P_{70}\}$	$\{P_1. P_2. \cdots . P_{18}\}$
7		$\{P_1. P_2. \cdots . P_{36}\}$
8		$\{P_1. P_2. \cdots . P_{54}\}$

The data augmentation is applied only to the flows of the dataset of the training. This dataset is expanded with the inclusion of the created flow fragments. Table I shows the fragments from the generated flows by the data augmentation process. The proposed classifier is trained for the learning of the map function $H: F_t^{(j)} \rightarrow y_j$ (where $t \le T$). Namely, the

classifier must be able to predict the label of the class y_j from a certain flow $F_t^{(j)}$ by just the first packets t. In the proposed method, the function of the loss of the cross-entropy and the optimizer of Adam [40] have been used for the training of the proposed classifier.

D. Monitoring

Real-time monitoring on high-speed networks is challenging work because of the great rates of the packet. In the proposed method, the mentioned point is the main reason that only process the packets of the network that are relevant to the attack's kind, which want to recognize. The module of the sniffer of the packet is responsible for the monitoring of the traffic of the network on the time of the real. As shown in Fig. 1, it captures and then forwards the incoming network packets and the outgoing packets of the network to the approach of the processing of the flow. The mentioned module, with the use of the library of libpcap is implemented, which furnishes an interface of the programming for the capturing of the packets which are passed via the interfaces of the network. Also, this library asserts the filters that can be configured to take just the specific packets. These filters, which usually are supported by the kernel of the system of the operating, cure the efficiency with the reduction of the overhead of the process of the filtering of the packet.

Also, a roster of the flows of the active and the prognostications about these flows are holded, which are done by the classifier of the flow. As shown in Fig. 2, while the flow of the network is updated by a novel packet, the classifier of the flow is used for the obtention of a prediction. The class of the final of a flow is the class that has the greater probability over the other classes, and classification threshold $\in [0.1)$. Whenever none of the probabilities of the class is greater than the certain threshold, then the proposed method returns Unknown as the class of the final. If the threshold of the classification is increased, then the rate of false positives is reduced, which improves the classification accuracy. The thresholds are presented by the administrator of the network, who sees the traffic of the network, and this person is responsible for the reaction to the attacks.

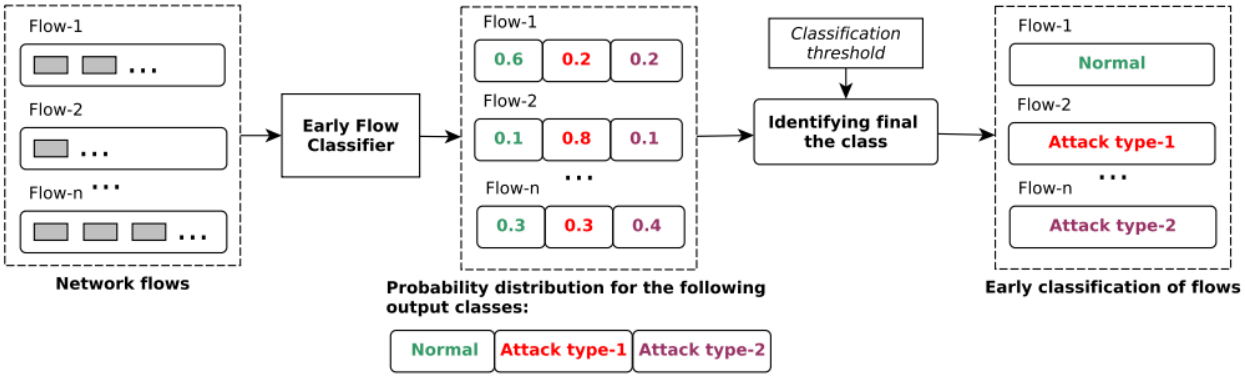

Fig. 2. The flow classification process.

IV. Experiments and Evaluation of Results

In the current part, the used dataset, the performed experiments and the obtained results are presented. The presented method is implemented on the computer by Core (TM) i7 CPU 3.0 GHz Intel(R) and 8G RAM. The efficiency of the presented approach is tested in the dataset of NSL-KDD. The data distribution of this dataset is shown in Fig. 3. It should be noted that the distribution of the data is unbalanced. The efficiency is displayed with the performing of the comparisons by the other deep learning models for the verification of the performance of the various components.

A. The Evaluation Criteria and the Parameter Setting

In the designed experiments, the strategy of K -Fold Stratified cross-validation is used for the test and for the training to ensure a good ratio of the training test. The optimizer of Adam [40] is applied as an optimizer for the optimization of weights for the training. The proposed model is trained in 150 iterations on the architecture of the basic. The rate of learning is adjusted to 0.001. Also, the exponential rate of the decay for the estimation of the first moment is adjusted to 0.9, and the estimation of the second moment is adjusted to 0.999. The tests are done with the use of Keras. Then, the proposed approach is appraised, due to three criteria: FPR, ACC and DR. ACC evaluates the ability of the model to predict the traffic of the normal and the traffic of the attack, while DR measures its ability for the detect of the attack traffic. A higher DR indicates that this approach is susceptible to the attacks of the network and that it aids in the persons to act the precautions on time. FPR is applied for evaluation of the misclassification of the traffic of the normal. DR and FPR should be presumed as common since a great DR can be overshadowed by a great FPR. The above criteria definitions are as follows:

$$ACC = \frac{TN + TP}{FP + FN + TP + TN} \tag{3}$$

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

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

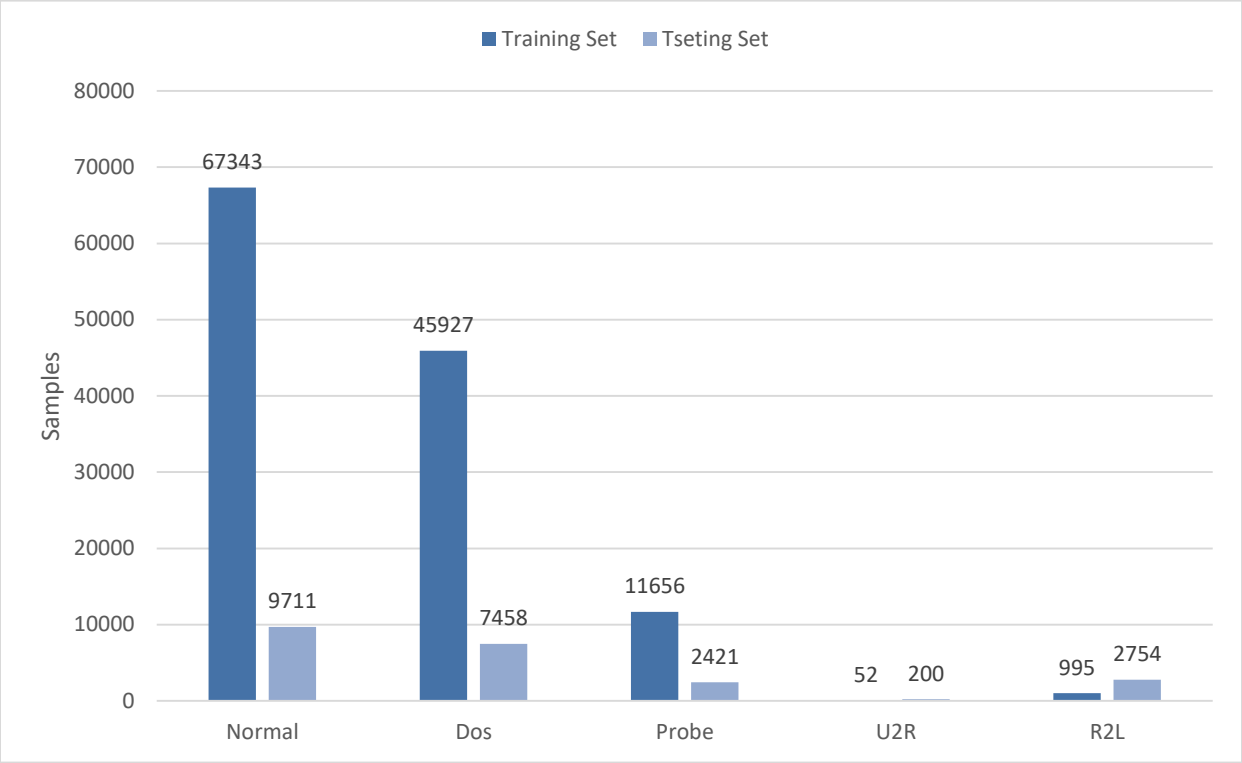

Fig. 3. The data distribution in the dataset of NSL-KDD.

TN and TP display the number of the traffic of the normal and the number of attacks that are properly classified, respectively. FP display the number of the records of the real normal that are misclassified as attacks. Also, FN display the number of attacks that are misclassified as the traffic of the normal.

B. The Acquired Results

To display the performance of the presented approach, two scenarios are considered: 1) the binary classification where the proposed method judges whether an instance is an attack or the traffic of the normal. 2) the classification of the multi-class where the proposed approach forecasts whether an instance is a class from the given attacks on the dataset or the normal.

1) The classification of the Binary: Table II displays the outcomes of the classification of the binary with the K-Fold Stratified cross-validation, where k is in the range of 2-10. For the dataset of NSL-KDD, the mean ACC is equal to 99.71%, the mean DR is equal to 99.75%, and the mean FPR is equal to 0.28%. In Fig. 4, it is clear that the presented method has the great DR and the great ACC. Also, it has a little FPR. With analysis of Fig. 4, it can be seen that the foremost outcomes become visible at K equal to 10. Because with an increment of the folds number, there will be more examples from every class of the attack/normal that the approach can be trained with them. Thus, the approach can classify them as superior. The outcomes show that the proposed method has the powerful ability to discern between the traffic of the normal and the traffic of the attack.

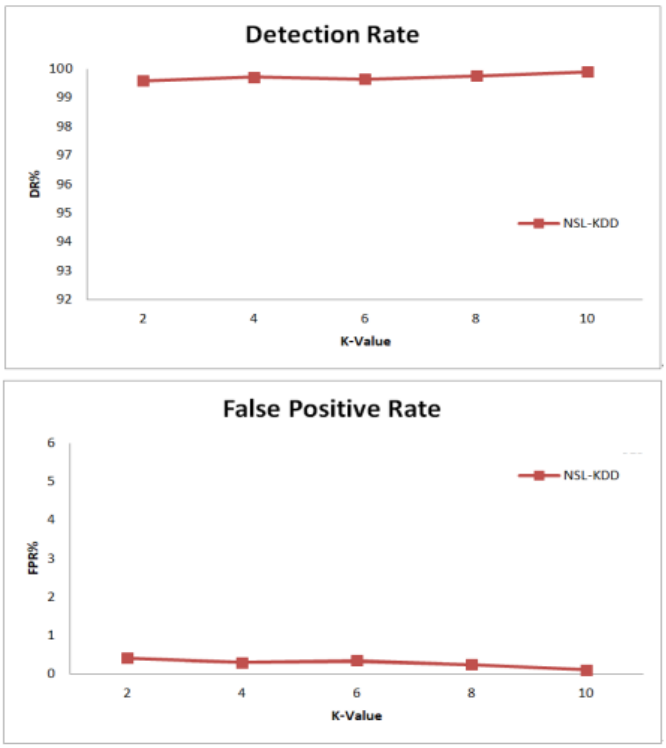

Fig. 4. The comparison of DR and FPR in the binary classification mode

TABLE II. THE OUTCOMES OF THE PRESENTED APPROACH ON THE BINARY CLASSIFICATION MODE

K	ACC	DR	FPR
2	99.62	99.63	0.40
4	99.66	99.73	0.27
6	99.62	99.67	0.36
8	99.81	99.79	0.25
10	99.83	99.93	0.11
Average	99.71	99.75	0.28

2) The classification of the multi-class: The outcomes of the classification of the multi-class are presented in Table III. For the dataset of NSL-KDD, the mean *ACC* is equal to 99.71%, the mean *DR* is equal to 99.73%, and the mean *FPR* is equal to 0.32%. Fig. 5 displays *DR* of the classes for the five-classes classification in the dataset of NSL-KDD. As it is clear from Fig. 5, the proposed method shows excellent performance in the detection of malicious attacks. To display the outcomes of the detection of the approach directly, a matrix of the confusion is used for the representation of the outcomes of the test. Fig. 6 displays the matrix of the confusion in the dataset of NSL-KDD for the classification of the multi-class. According to Fig. 6, it is shown which most instances are focused on the matrix diameter, and this point shows that the total efficiency of the detection is great.

TABLE III. THE OUTCOMES OF THE PRESENTED APPROACH ON THE MULTI-CLASS CLASSIFICATION MODE

K	ACC	DR	FPR
2	99.58	99.46	0.33
4	99.72	99.85	0.40
6	99.69	99.77	0.37
8	99.81	99.78	0.20
10	99.74	99.80	0.31
Average	99.71	99.73	0.32

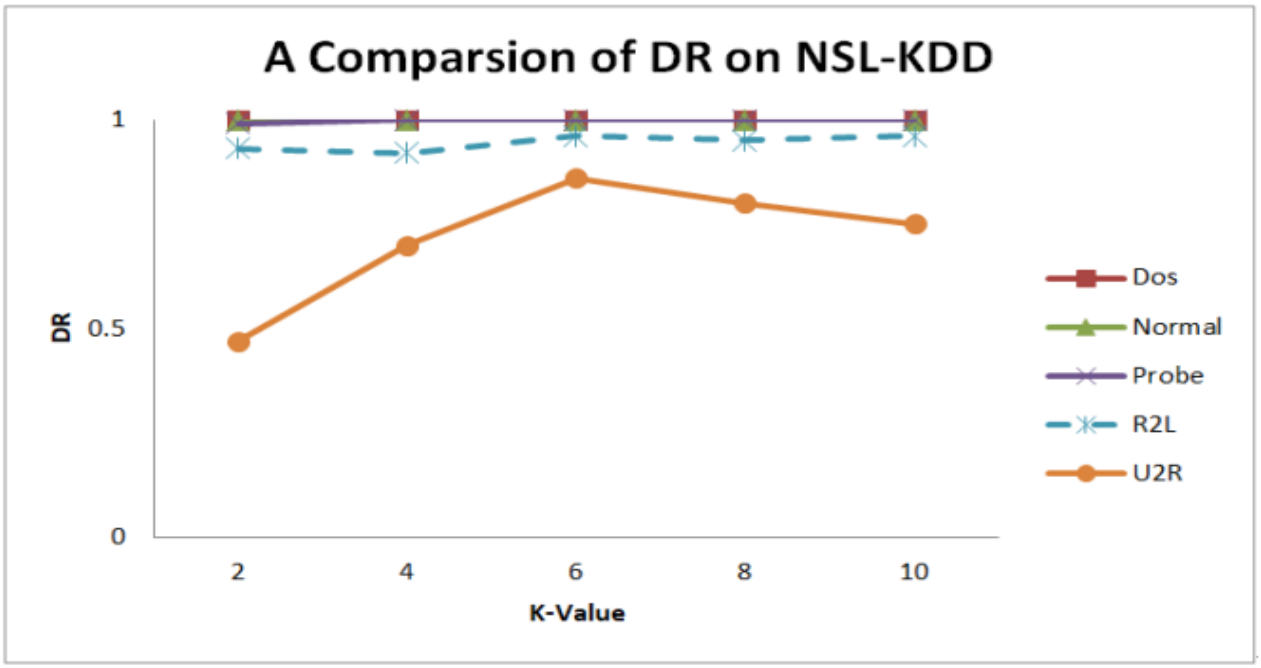

Fig. 5. The *DR* results of all classes in the dataset of NSL-KDD.

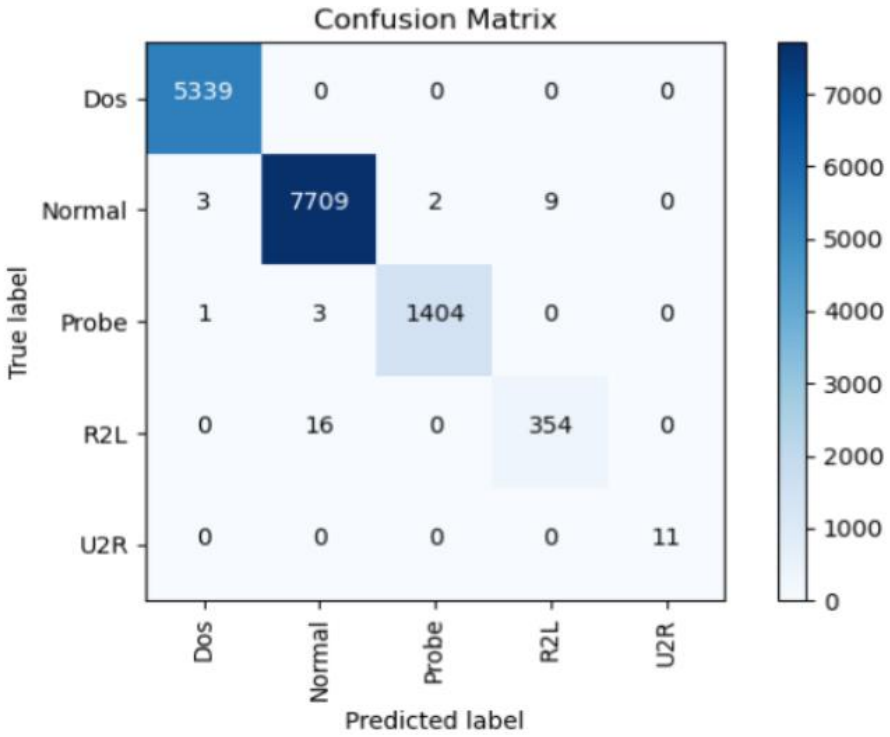

Fig. 6. Confusion matrix for the dataset of NSL-KDD.

TABLE IV. THE COMPARISON OF OUR PRESENTED APPROACH WITH SIMILAR METHODS

Model	ACC	DR	FPR
SVM [41]	69.52	70.23	1.03
RF [42]	69.84	68.79	1.16
AdaBoost [43]	71.98	72.36	0.99
HAST-IDS [44]	93.27	95.85	0.52
CNN-BiLSTM [45]	99.22	98.88	0.43
LuNet [46]	99.14	99.02	0.61
Pelican[48]	99.21	99.13	0.65
Proposed Method	99.81	99.80	0.20

To prove the strong efficiency of the presented approach, the proposed approach of this paper is compared with seven advanced models. These methods are: SVM in [41], RF in [42], AdaBoost in [43], HAST-IDS in [44], CNN-BiLSTM in [45], LuNet in [46] and Pelican in [47]. These methods are trained and tested with a similar data partition and, with the strategy of the cross-validation with k equal to 10. The outcomes of the proposed approach are provided in the best case. Table IV shows the outcomes of the multi-class comparison of the proposed method with seven advanced models. It is clear that the proposed model outperforms the other models on all evaluation criteria. The outcomes display that the presented model betters the detection rate. Also, it maintains a low false positive rate. This point shows the greater effectiveness of the model for network intrusion detection.

C. Discussion

Despite the good performance of the proposed method, there are limitations to the presented work, which are discussed in this section. The main threat to the validity of the proposed method is that only one dataset was used in the evaluation. Therefore, the test results may be different for other datasets that have different types of attack classes. To the best of the authors' knowledge, most publicly available datasets, with the exception of NSL-KDD, are outdated and/or lack raw network traffic data. The proposed approach can extract relevant features from raw network traffic data instead of relying on manual feature selection process. Therefore, it can be easily applied to other datasets. Future work could be to conduct additional experiments by using different datasets such as CSE-CIC-IDS2018 to mitigate this threat.

Another threat to the validity of the proposed method is that the evaluation may appear subjective. Another threat to the proposed work is that only three attack classes are considered. As mentioned, a simple DNN model (ie, a model with a relatively small number of trainable parameters) is trained to detect only certain types of attacks to achieve good accuracy and reasonable prediction time.

V. CONCLUSION

In the current article, a system of the early detection of the intrusion, which is based on the end-to-end, has been presented for the prevention of real-time network attacks before they cause further detriment to the system of the attack. A classifier basis on CNN is used, to detect the attack. The model is trained as a method of the supervised extraction of the related features by the data of the raw traffic of the network rather than relying on the process of the manual selection of the feature, which is applied to the most relevant methods. The designed experiments have been evaluated in the dataset of NSL-KDD, and the obtained results display that the presented approach improves the overall results (especially the rate of detection and the rate of false positives). The presented approach in the current paper is competitive. It obtains the lower overhead of the computational, which is essential for practical network intrusion detection.

Further analysis has shown that the proposed method has gaps for the betterment of the discerning among the groups by similar features, and the future work will be reviewed here. Furthermore, in the latter task, the researchers can aim to evaluate the proposed approach in the different datasets. Also, the various architectures of the network of the neural can be investigated to check the comparative efficiency in the early detection of the attack.

REFERENCES

[1] F. Jiang et al., "Deep Learning Based Multi-Channel Intelligent Attack Detection for Data Security," IEEE Trans. Sustain. Comput., vol. 5, no. 2, pp. 204–212, Apr. 2020.

[2] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," Transactions on Emerging Telecommunications Technologies, vol. 32, no. 1, p. e4150, 2021.

[3] Z. Wang, "Deep Learning-Based Intrusion Detection With Adversaries," IEEE Access, vol. 6, pp. 38367–38384, 2018.

[4] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung, "Intrusion detection system: A comprehensive review," Journal of Network and Computer Applications, vol. 36, no. 1, pp. 16–24, 2013.

[5] J. Li, Y. Qu, F. Chao, H. P. H. Shum, E. S. L. Ho, and L. Yang, "Machine learning algorithms for network intrusion detection," AI in Cybersecurity, pp. 151–179, 2019.

[6] A. A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for Internet of Things," Futur. Gener. Comput. Syst., vol. 82, pp. 761–768, May 2018.

[7] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," in Proceedings of ICML workshop on unsupervised and transfer learning, JMLR Workshop and Conference Proceedings, 2012, pp. 17–36.

[8] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization.," ICISSp, vol. 1, pp. 108–116, 2018.

[9] B. Claise, B. Trammell, and P. Aitken, "Specification of the IP flow information export (IPFIX) protocol for the exchange of flow information," 2013.

[10] A. D. Khairkar, D. D. Kshirsagar, and S. Kumar, "Ontology for detection of web attacks," in 2013 International Conference on Communication Systems and Network Technologies, IEEE, 2013, pp. 612–615.

[11] Y. Zhang, X. Chen, L. Jin, X. Wang, and D. Guo, "Network intrusion detection: Based on deep hierarchical network and original flow data," IEEE Access, vol. 7, pp. 37004–37016, 2019.

[12] O. Joldzic, Z. Djuric, and P. Vuletic, "A transparent and scalable anomaly-based DoS detection method," Computer Networks, vol. 104, pp. 27–42, 2016.

[13] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," Journal of Network and Computer Applications, vol. 60, pp. 19–31, 2016.

[14] T. F. Ghanem, W. S. Elkilani, and H. M. Abdul-Kader, "A hybrid approach for efficient anomaly detection using metaheuristic methods," J Adv Res, vol. 6, no. 4, pp. 609–619, 2015.

[15] C. N. Modi, D. R. Patel, A. Patel, and M. Rajarajan, "Integrating signature apriori based network intrusion detection system (NIDS) in cloud computing," Procedia Technology, vol. 6, pp. 905–912, 2012.

[16] K. Shafi and H. A. Abbass, "An adaptive genetic-based signature learning system for intrusion detection," Expert Syst Appl, vol. 36, no. 10, pp. 12036–12043, 2009.

[17] Y. Li, J. Xia, S. Zhang, J. Yan, X. Ai, and K. Dai, "An efficient intrusion detection system based on support vector machines and gradually feature removal method," Expert Syst Appl, vol. 39, no. 1, pp. 424–430, 2012.

[18] Y. Zhang, H. Zhang, X. Zhang, and D. Qi, "Deep learning intrusion detection model based on optimized imbalanced network data," in 2018 IEEE 18th International Conference on Communication Technology (ICCT), IEEE, 2018, pp. 1128–1132.

[19] B. Yan and G. Han, "Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system," IEEE Access, vol. 6, pp. 41238–41248, 2018.

[20] Y. Xin et al., "Machine learning and deep learning methods for cybersecurity," Ieee access, vol. 6, pp. 35365–35381, 2018.

[21] W. Elmasry, A. Akbulut, and A. H. Zaim, "Evolving deep learning architectures for network intrusion detection using a double PSO metaheuristic," Computer Networks, vol. 168, p. 107042, 2020.

[22] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," Knowl Based Syst, vol. 189, p. 105124, 2020.

[23] A. Salari, H. Shakibi, M. Alimohammadi, A. Naghdbishi, and S. Goodarzi, "A machine learning approach to optimize the performance of a combined solar chimney-photovoltaic thermal power plant," Renew Energy, vol. 212, pp. 717–737, 2023.

[24] Ö. Kasim, "An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks," Computer Networks, vol. 180, p. 107390, 2020.

[25] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," IEEE Trans Emerg Top Comput Intell, vol. 2, no. 1, pp. 41–50, 2018.

[26] S. Afzal, B. M. Ziapour, A. Shokri, H. Shakibi, and B. Sobhani, "Building energy consumption prediction using multilayer perceptron neural network-assisted models; comparison of different optimization algorithms," Energy, vol. 282, p. 128446, 2023.

[27] M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, "Deep learning approach combining sparse autoencoder with SVM for network intrusion detection," Ieee Access, vol. 6, pp. 52843–52856, 2018.

[28] S. Parampottupadam and A.-N. Moldovann, "Cloud-based real-time network intrusion detection using deep learning," in 2018 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), IEEE, 2018, pp. 1–8.

[29] Y. Zhang, P. Li, and X. Wang, "Intrusion detection for IoT based on improved genetic algorithm and deep belief network," IEEE Access, vol. 7, pp. 31711–31722, 2019.

[30] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," Ieee Access, vol. 7, pp. 41525–41550, 2019.

[31] H. Yang, G. Qin, and L. Ye, "Combined wireless network intrusion detection model based on deep learning," IEEE Access, vol. 7, pp. 82624–82632, 2019.

[32] M. Elsayed and M. Erol-Kantarci, "Deep reinforcement learning for reducing latency in mission critical services," in 2018 IEEE Global Communications Conference (GLOBECOM), IEEE, 2018, pp. 1–6.

[33] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang, "The expressive power of neural networks: A view from the width," Adv Neural Inf Process Syst, vol. 30, 2017.

[34] S. Yang and A. Browne, "Neural network ensembles: combining multiple models for enhanced performance using a multistage approach," Expert Syst, vol. 21, no. 5, pp. 279–288, 2004.

[35] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[36] I. Goodfellow, Y. Bengio, and A. Courville, Deep learning. MIT press, 2016.

[37] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 807–814.

[38] M. Lin, Q. Chen, and S. Yan, "Network in network. arXiv 2013," arXiv preprint arXiv:1312.4400, 2013.

[39] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," J Big Data, vol. 6, no. 1, pp. 1–48, 2019.

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[41] I. Ahmad, M. Basheri, M. J. Iqbal, and A. Rahim, "Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection," IEEE access, vol. 6, pp. 33789–33795, 2018.

[42] J. Zhang, M. Zulkernine, and A. Haque, "Random-forests-based network intrusion detection systems," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 38, no. 5, pp. 649–659, 2008.

[43] W. Hu, J. Gao, Y. Wang, O. Wu, and S. Maybank, "Online adaboost-based parameterized methods for dynamic distributed network intrusion detection," IEEE Trans Cybern, vol. 44, no. 1, pp. 66–82, 2013.

[44] W. Wang et al., "HAST-IDS: Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection," IEEE access, vol. 6, pp. 1792–1806, 2017.

[45] J. Sinha and M. Manollas, "Efficient deep CNN-BiLSTM model for network intrusion detection," in Proceedings of the 2020 3rd International Conference on Artificial Intelligence and Pattern Recognition, 2020, pp. 223–231.

[46] Q. Gao, "Recommended System Optimization in Social Networks based on Cooperative Filter with Deep MVR Algorithm," 2022.

[47] P. Wu, H. Guo, and N. Moustafa, "Pelican: A deep residual network for network intrusion detection," in 2020 50th annual IEEE/IFIP international conference on dependable systems and networks workshops (DSN-W), IEEE, 2020, pp. 55–62.

[48] P. Wu, H. Guo, and N. Moustafa, "Pelican: A deep residual network for network intrusion detection," in 2020 50th annual IEEE/IFIP international conference on dependable systems and networks workshops (DSN-W), IEEE, 2020, pp. 55–62.