Hindawi
Security and Communication Networks
Volume 2023, Article ID 9840753, 1 page
https://doi.org/10.1155/2023/9840753

Retraction

Retracted: Detection of DDoS Attack within Industrial IoT Devices Based on Clustering and Graph Structure Features

Security and Communication Networks

Received 5 December 2023; Accepted 5 December 2023; Published 6 December 2023

This article has been retracted by Hindawi, as publisher, following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of systematic manipulation of the publication and peer-review process. We cannot, therefore, vouch for the reliability or integrity of this article.

Please note that this notice is intended solely to alert readers that the peer-review process of this article has been compromised.

Wiley and Hindawi regret that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] H. Jing and J. Wang, "Detection of DDoS Attack within Industrial IoT Devices Based on Clustering and Graph Structure Features," *Security and Communication Networks*, vol. 2022, Article ID 1401683, 9 pages, 2022.

Hindawi
Security and Communication Networks
Volume 2022, Article ID 1401683, 9 pages
https://doi.org/10.1155/2022/1401683

Research Article

Detection of DDoS Attack within Industrial IoT Devices Based on Clustering and Graph Structure Features

Hengchang Jing and Jian Wang

College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

Correspondence should be addressed to Hengchang Jing; jhc@nuaa.edu.cn

Received 16 December 2021; Accepted 5 February 2022; Published 7 March 2022

Academic Editor: Chin-Ling Chen

Network available and accessible is of great importance to the Internet of things (IoT) devices. In this study, a novel machine learning method is presented to predict the occurrence of distributed denial-of-service (DDoS) attacks. Firstly, a structure of edges and vertices within graph theory is created to simultaneously extract traffic data characteristics. Eight characteristics of traffic data are selected as input variables. Secondly, the principal component analysis (PCA) model is adopted to extract DDoS and normal communication features further. Then, DDoSs are detected by fuzzy C-means (FCM) clustering with these features. In the case study, 2000 traffic data in dataset CICIDS-2017 are used to verify the practicability of this method. The results of recall, false positive, true positive, true negative, and false negative are 100.00%, 1.05%, 68.95%, 0.00%, and 30.00%. Compared with other methods, the results demonstrate that the detecting reliability is improved, and the method has a good effect on the detection of DDoS attacks.

1. Introduction

Network security problems have become increasingly outstanding with the development of the Internet of things (IoT) technology. There are a lot of malicious attacks on the network. Maintaining the stability and reliability of IoT devices is a complex task due to the highly distributed and multiple connected characteristics. Distributed denial-of-service (DDoS) attacks are the most common way to destroy the accessibility of a network. DDoS attacks have the characteristics of low launching cost and high attack intensity, which can cause significant harm to the victims quickly. The DDoS attack is different from a penetration attack, which does not invade the target servers by a Trojan or root program. DDoS attacks have two types, and one is a network protocol attack to damage servers by the network system vulnerability maliciously. The other is directly run out of resources by infinitely sending useless packages to the object [1, 2], which will lead the target system service to block, and the IoT equipment cannot provide a normal service or access to clients. The first type of attack can be effectively defended by system patching, but the second one must accurately distinguish legitimate traffic data from network flows. Thus, this dataset mining problem has drawn attention to many researchers in network security.

Service resources for the victims of DDoS attacks include network bandwidth, file system space capacity, open processes, or allowed connections [3]. These attacks will lead to the decrease in memory capacity resources, and bandwidth speed will inevitably decrease. According to the popularization of information technology, especially the IoT, more and more host types of botnets that is a host infected with a malicious program and under the control of an attacker appear [4]. Verizon revealed a DDoS attack on a US university, the campus network speed has slowed down significantly, and the domain name server (DNS) was flooded with abnormal queries from the school's approximately 5,000 IoT devices, including streetlights, vending machines, and other botnet devices [5].

It is usually hard for network security officers to identify them because many network devices such as routers, switches, and servers produce a vast amount of system log data. An effective way to track network status is to deploy monitoring agents in the network and collect log information corresponding to a change in system status [6]. Researchers have developed different models to address this problem, such as signature-based intrusion detection, entropy variation method, machine learning detection, and artificial intelligence-based method. Analyzing the correlation model is used to detect anomalous network activities through the temporal and process information [7]. A causal inference algorithm is developed to detect a nationwide research and education network in Japan by 15 months long system log messages collected [8]. A DDoS defense scheme for the IoT using dynamic population and point process theory is presented to predict and detect DDoS attacks by analyzing traffic data. A generalized entropy-based metric is proposed to detect the low rate DDoS attacks to the control layer [9]. RBF neural network is used as an anomaly based approach, and the detection ratio of 96% is shown in the UCLA dataset [10]. The clustering models, such as the K-means model [11] and Gaussian mixture density model [12], are unsupervised methods that classify datasets into multiple clusters only with varying distances of membership, which can divide each traffic data into different partitions for distincting DDoS and normal flows. The label will be not required in unsupervised methods. When the traffic data of network communication are divided into different partitions with clustering, the DDoS attacks will be easily found. However, the fuzzy C-means (FCM) cluster model in data mining is seldom used to perform DDoS detection.

For acquiring an effective detection method of DDoS attacks, this study proposes a novel detected method. The traffic dataset of network communication is first analyzed using graph theory. Then, the principal component analysis (PCA) is used to filter the characterization factors of DDoS attacks. The FCM clustering model divides the network flows of traffic data into different partitions. In the case study, the dataset of CICIDS-2017 was selected to verify the practicability of the method, and the results were presented. The novelty of this model is as follows: (1) the traffic data can be unsupervised for training, so labels are not needed; (2) using graph theory not only considers the topological structure relationship between IP and ports but also considers flows; and (3) many factors of traffic data can be automatically selected to reduce the overload of calculation and improve the accuracy of clustering.

2. Graph Structure Features

Graph theory [13–15] is used to build a topological structure of traffic data. The traffic data can be abstracted as a directed graph (DG) in communication networks. The communication relationship, frequency, flow duration, and other valuable information between vertices could be regarded as the edges (links) $E = \{e1, e2, \ldots, em\}$ and the IP addresses and ports are vertices (nodes) $V = \{v1, v2, \ldots, vn\}$, where m is the total number of edges and n is the total number of vertices.

For instance, six vertices exist in a graph structure containing $v1 = 172.16.0.1 : 43201$, $v2 = 192.168.10.50 : 80$, $v3 = 101.69.185.208 : 443$, $v4 = 192.168.10.16 : 51784$, $v5 = 103.43.91.16 : 443$, and $v6 = 192.168.10.9 : 9901$, to which three edges connect ($e1 = 172.16.0.1 : 43201 \longrightarrow 192.168.10.50 : 80$, $e2 = 101.69.185.208 : 443 \longrightarrow 192.168.10.16 : 51784$, and $e3 = 103.43.91.16 : 443 \longrightarrow 192.168.10.9 : 9901$).

The weight of the edge contains various information that can be expressed as an array. The connectivity of traffic data can be considered an adjacent matrix A to show the relationship between these IP addresses and ports clearly. The matrix A is as follows:

$$A = \begin{bmatrix} \mathbf{w}_{11} & \cdots & \mathbf{w}_{1j} \\ \vdots & \ddots & \vdots \\ \mathbf{w}_{i1} & \cdots & \mathbf{w}_{ij} \end{bmatrix}, \quad (1)$$

where the vector wij represented the array of weights between nodes i and j. If the nodes i and j are connected, the weights are nonzero. Otherwise, it is zero.

The weights are the traffic data features. Different features can reveal various communication relationship characteristics in the topological structure. The DDoS attack contains directed attack and reflected attack [16, 17]. In the reflected DDoS attack, attackers indirectly attack the target IP service and send specialized packet data to an opening server for disguising IP address, and the opening server will reply to the request packet data sent to the attacked server many times. It is difficult to judge a DDoS attack only by its IP address and ports. However, a DDoS attack is from one source address to a terminal address to break down servers and have diverse characteristics. Thus, the features of edges can recognize attacks effectively. The DDoS attack should also be distinguished from the flash crowd that is a normal access behavior of the clients. Flash crowd appears when a huge number of clients access a server simultaneously due to top search results, popular products, and so on. Users want to get interested in information from the server as soon as possible. The server is slow or even shut down, which is unexpected, and most do not want to see it in advance. Overall consideration, we analyzed graph-based and flow-based features under the DDoS attack environment to select features to detect DDoS attacks. Eight features are selected, as follows.

2.1. Total Forward Packet. The forward packet means a request sends from a source node to the target node. In traffic data, the total forward packet represents the number of received data packets of the target node from an adjacent source node in the network. The total forward packet can be regarded as an indicator of the activity of a source node. Useless information and command send to slaves from masters in the DDoS attacks.

2.2. Total Backward Packet. The backward packet is the reply information sent to the source node after the target node receives a request. The total backward packet represents the

number of data packets sent from a target node to an adjacent source node in the network. In detecting DDoS attacks, it can represent the slaves' activity of the network.

2.3. Standard Deviation of Backward Packet Length. The standard deviation of backward packet length represents the fluctuation of packets replying from a target node to a source node. The standard deviation of the backward packet length of the DDoS attack is smaller than normal traffic. In DDoS attacks, the length of packets between two particular nodes is all the same, and the interval time tends to be stable. The standard deviation is almost zero or the same small size. Thus, the length of packets is the same when the message of the victim sever returns to the attack node. In normal traffic, the length of packets fluctuates significantly due to different requests. Thus, the standard deviations are large and variable to the different connection nodes.

2.4. Total Visit View. The total visit view is the number of accesses to a destination IP and port from a source node continuously. In the DDoS attacks, the source node will continue sending packets to disrupt normal traffic on the target server until managers detect it.

2.5. Average Packet Length. The average packet length is a statistical value of a packet in a duration of time. In DDoS attacks, the average packet length is small because the duplicate packets only contain header files without any data fields or less content. Each data packet has the same header but different contents in normal flows. The average length of the packet is large and various.

2.6. Flow Duration. The flow duration is the total communication time between two nodes from connection to disconnection. The flow duration of packets sent by the same attacker tends to be stable in the DDoS attacks, while the duration time frequently fluctuates in normal communication.

2.7. Standard Deviation of Flow Interval Time. The flow interval represents the interval between sending each packet during a flow. When the DDoS attackers send packets, the interval time of flow tends to be equal. However, the interval time of normal flows depends on the reply time of the target server. The destination vertices receive different packets, and the processing time is also different. Furthermore, the interval time of normal traffic is affected by noise, network bandwidth, receiving window size, sending window size, etc., which shows a significant difference from DDoS attacks [5]. In addition, the interval time of normal traffic is limited by network bandwidth, noise, size of sending window, and other factors, which is significantly different from DDoS attacks.

2.8. Mean Active Time of Flow. The meaning of active mean is different from the traffic duration mentioned above. It represents the survival time of each packet sent within the communication time of two vertices. The definition is the total interval between sending the connection request packet and the last disconnect request packet. DDoS attackers make attacks many times in a short period, and the sending packet is generally the same and small. On the contrary, the normal flow survival time depends on the communication time. Otherwise, the value is zero.

The label of each traffic data is normal communication or DDoS attack. For detecting convenience, the labels are digitized (zero for normal communication and one for DDoS attacks). In the dataset, each flow has been labeled based on its weights.

For the example above, *e1* is assumed as a DDoS attack. *e2* and *e3* are normal communication.

Then, the nodes *v1-v6* are connected by *e1-e3*, and the weight array is as follows:

$$A = \begin{bmatrix} 0 & \mathbf{w}_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{w}_{34} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{w}_{56} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{w_{ij}} = \left[w_{ij}^1, w_{ij}^2, w_{ij}^3, w_{ij}^4, w_{ij}^5, w_{ij}^6, \ldots, w_{ij}^k \right]^T, \tag{2}$$

where the w_{ij}^k is the kth weight between nodes i and j.

The eight features ($k = 8$) can be inputted into the weight array. The values of features are assumed as known. Then, the $\mathbf{w_{ij}}$ can be written in Table 1, where the k is in keeping with the above orders.

The label array can be expressed as follows:

$$Y = [y_1, y_2, y_3]^T, \tag{3}$$

where Y is the array of label between nodes i and j; y_i is the label of ith edge.

With respect to the assumption, the label array can be written as $[1, 0, 0]^T$.

3. Dimensionality Reduction in the Weight Matrix

The PCA is commonly applied for dimensionality reduction, which projects data onto only the first few principal components to obtain lower-dimensional data [18]. Thus, PCA can be solved by lossy compression of a dataset to express characteristics by less dimensional data.

A group of new orthogonal bases should be found in the PCA algorithm where the projection's data have a maximum variance value. In other words, the distance of data is the largest in the projection of orthogonal basis. When the m weights: $\{\mathbf{w}_1, \mathbf{w}_2, \ldots \mathbf{w_m}\}$ exist, and each weight has n dimensions: $\mathbf{w_i} = [w_i^1, w_i^2 \ldots, w_i^n]^T$, the variance of all data projected onto that basis can be expressed as [19]follows:

$$J_j = \frac{1}{m} \sum_{i=1}^{m} \left(\mathbf{w_i^T u_j} - \overline{w}_\mathbf{i} \mathbf{u_j} \right)^2, \tag{4}$$

Table 1: Weight array of three edges.

Edge	w^1	w^2	w^3	w^4	w^5	w^6	w^7	w^8
e1	3	4	5795.50	7	1661.86	77116	30796.08	1000
e2	2	3	0	1	13.60	655938	327914.33	0
e3	2	0	0	1	9	377	0	0

where m is the number of weight samples; $\mathbf{w_i}$ is the ith weight after the zero-mean initialization; $\overline{w}_\mathbf{i}$ is the average weight; $\mathbf{u_j}$ is the jth orthogonal basis; and J_j is the variance when the dataset projects onto the orthogonal basis j.

Then, the zero mean is processed for each element of X by (5). The X columns are centered on having an average value zero and scaled to have a standard deviation one.

$$\mathbf{w_i} = \frac{\mathbf{w_i} - \overline{w}_\mathbf{i}}{\sigma}, \tag{5}$$

where σ is the standard deviation of weight array $\mathbf{w_i}$.

$\overline{w}_\mathbf{i}$ is zero when zero-mean initialization is processed. Then, (4) can be written as [19]follows:

$$\begin{aligned} J_j &= \frac{1}{m}\sum_{i=1}^{m}\left(\mathbf{w_i^T u_j}\right)^2 \\ &= \mathbf{u_j^T}\frac{1}{m}\sum_{i=1}^{m}\left(\mathbf{w_i w_i^T}\right)^2\mathbf{u_j}. \end{aligned} \tag{6}$$

The matrix form can be expressed as follows:

$$\begin{aligned} J_j &= \frac{1}{m}\mathbf{u_j^T}XX^T\mathbf{u_j} \\ &= \frac{1}{m}\mathbf{u_j^T}S\mathbf{u_j}, \end{aligned} \tag{7}$$

where X is the matrix of weights, and the equation is shown in (8); S is the value of matrix multiplication between X and X^T, which is also called the covariance matrix.

$$\begin{aligned} X &= \begin{bmatrix} \mathbf{w_1} \\ \cdots \\ \mathbf{w_m} \end{bmatrix} \\ &= \begin{bmatrix} w_1^1 & \cdots & w_1^n \\ \vdots & \ddots & \vdots \\ w_m^1 & \cdots & w_m^n \end{bmatrix}. \end{aligned} \tag{8}$$

The orthogonal basis can be deviated by the Lagrangian operator [20]. For obtaining an orthogonal basis, the maximum variance of the data projected onto the basis is equal to the eigenvalue of the covariance matrix of X. It can be written as follows:

$$\max J_j = \lambda_j \tag{9}$$

When the dimensionality reduction is processed, the eigenvalues are first arranged in descending order. The weight matrix of reduced dimension can be solved by the eigenvectors corresponding to the first k maximum eigenvalues of the covariance matrix if the dimension reduces to k. In order words, the orthogonal basis is equal to the eigenvectors of the covariance matrix of X.

With respect to the definition of covariance, the covariance of matrix X can be expressed as [21]follows:

$$\begin{aligned} \mathrm{cov}(X) &= \frac{1}{m}XX^T \\ &= S, \end{aligned} \tag{10}$$

where cov represents the covariance matrix.

Then, the covariance matrix S is diagonalized, and the eigenvectors and eigenvalues can be obtained. Thus, the matrix of dimensionality reduction can be calculated by (9).

$$X_{\mathrm{new}} = \begin{bmatrix} \mathbf{u}_1 \\ \cdots \\ \mathbf{u_k} \end{bmatrix}^T X, \tag{11}$$

where X_{new} is the matrix of $k * m$.

The above example is further used to instruct, and the first weight arrays are selected for simplifying the weight matrix X to express the process clearly. It can be written as follows:

$$\begin{aligned} X &= \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{w}_3 \end{bmatrix} \\ &= \begin{bmatrix} 3 & 4 \\ 2 & 3 \\ 2 & 0 \end{bmatrix}. \end{aligned} \tag{12}$$

After the zero-mean initialization, the weights of three edges values are shown in Table 2.

Covariance matrix S is solved by $1/mXX^T$, shown in Table 3. Then, the eigenvectors and eigenvalues of S can be obtained, shown in Table 4.

The weight matrix should be reduced from n dimensions to k dimensions. Thus, an appropriate k value ought to be determined. The general selection criterion is the proportion of variance before and after projection. The higher proportion will have a higher correlation, so they are used as the selection criterion of the k value.

With respect to the relationship between covariance and eigenvalue, it can be expressed as [22]follows:

$$\begin{aligned} q &= \frac{J(X_{\mathrm{new}})}{J(X)} \\ &= \frac{\sum_{j=1}^{j}\lambda_j}{\sum_{j=1}^{n}\lambda_j}, \end{aligned} \tag{13}$$

where q is the expectation value.

However, not just only one array is selected. A higher proportion q will have a higher correlation. Thus, a series of arrays with large expectation values are used. The sum of the expectation value q is larger than 90% with respect to the analysis of some references [5], [23-25].

Table 2: Zero-mean initialization weights of three edges.

Edge	w^1	w^2
e1	0.67	1.67
e2	-0.33	0.67
e3	-0.33	-2.33

Table 3: Covariance matrix of three edges.

	1	2
1	0.22	0.56
2	0.56	2.89

Table 4: Eigenvalue and eigenvector of three edges.

Eigenvector	1	2
1	0.22	0.56
2	0.56	2.89
Eigenvalue	0.11	3.00

In this example, the expectation values are 96.42% and 3.57%, respectively. Thus, the second weight array can be ignored. The dimension can be reduced to one. Finally, a new weight matrix ($3 * 1$) can be solved by (9). It can be written as follows: $[1.08, 0.30, -1.38]^{\mathrm{T}}$.

4. Fuzzy C-Means (FCM) Clustering

After the PCA dimensionality reduction, cluster analysis can be processed with respect to the new weight matrix. A fuzzy C-means (FCM) is an unsupervised learning model presented in 1973 [26, 27], which does not require manual creation of categories for dataset labels. The FCM algorithm is an effective cluster model based on a fuzzy clustering algorithm to minimize an objective function, dividing data into different classes by the degree of membership. It is widely applied in different areas, such as news classifying, user buying patterns (cross-selling), image segmentation, and genetic technology. However, it is seldom used to classify nodes to normal access and DDoS attack in the network security area. Therefore, this study applies the FCM to judge the DDoS attack.

The weight analysis matrix (11) is used as a sample observation matrix to divide each edge into different partitions. The number of partitions c is determined manually, and a membership matrix $\mathbf{M}$ is generated randomly, where the number of matrix rows is the same as the number of partitions (total of c classes) and the columns are equal to the index of edges, a total of m (such as three in the example of above). When the number of dimensions of sensitivity is assumed as n, the membership matrix $\mathbf{M}$ can be expressed as follows [28]:

$$\mathbf{M} = \begin{bmatrix} \Delta\mathbf{M}_1 \\ \vdots \\ \Delta\mathbf{M}_m \end{bmatrix} = \begin{bmatrix} M_{11} & \cdots & M_{1c} \\ \vdots & \ddots & \vdots \\ M_{m1} & \cdots & M_{mc} \end{bmatrix}^T, \tag{14}$$

where M_{ij} is the membership of edge i at the partition j, and the membership values in the membership matrix are all ranged from 0 to 1. The membership represents the degree of reliability of an edge in a partition.

Then, the center of partitions $\mathbf{C}_j$ $[\mathbf{C}_j = (C_{1j}, C_{2j}, \ldots, C_{Mj})]$ in each class is determined as follows:

$$\mathbf{C}_j = \frac{\sum_{i=1}^{N} M_{ij}^m \boldsymbol{\omega}_j}{\sum_{i=1}^{N} M_{ij}^m}, \tag{15}$$

where m is a power exponent m $(m > 1)$.

With respect to the center of clustering, the membership matrix can be revised via solving the Euler distance [27]:

$$\mathbf{M}_{\mathbf{ij}} = \frac{1}{\sum_{k=1}^{c} \left(d_{ij}/d_{ik}\right) 2/m - 1}, \tag{16}$$

where d_{ij} is the Euler distance of edge i at the partition j; represents the distance solving equation that can be expressed as follows:

$$\boldsymbol{\omega}_j - \mathbf{C}_j = \sqrt{\sum_{i=1}^{N} \left(\boldsymbol{\omega}_\mathbf{j} - \mathbf{C}_{ij}\right)^2}. \tag{18}$$

Then, an objective function is employed to solve the weights that are the sum of squares for the distance sensitivity values to their cluster centers, expressed in (19). The objective function should be minimized and the partition of the minimum value isselected as their divided clusters [29].

$$F(\mathbf{M}, \mathbf{C}) = \sum_{i=1}^{N} \sum_{j=1}^{c} M_{ij} d_{ij}^2, \tag{19}$$

where F is the objective function that should be optimized; C is the matrix of the center of partition that can be expressed as $\mathbf{C} = [\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_c]^T$.

It is not easy to decrease the convergence value to zero in the numerical calculation. Thus, a convergence condition ξ $(\xi > 0)$ can be set to judge to stop the looping. Meanwhile, a maximum iteration time is also set to prevent an endless loop. The convergence condition can be expressed as follows: $|F^l(\mathbf{M}, \mathbf{C}) - F^{l-1}(\mathbf{M}, \mathbf{C})| < \xi$, where l is the lth iteration time. (15)–(19) are repeated until the result is up to the convergence condition or maximum iteration time minimizes the objective function. Finally, the objective function up to the minimum and the final membership matrix is obtained. The edges of IP and port connections are all classified.

The above example can be classified into different partitions using the FCM algorithm. When c assumes two, the three edges will be divided into two partitions. The first partition only contains one edge, *e1*. The other contains two: *e2* and *e3*. The label of *e1* is DDoS. *e2* and *e3* are normal communications. Thus, a similar dataset can be separated into different partitions.

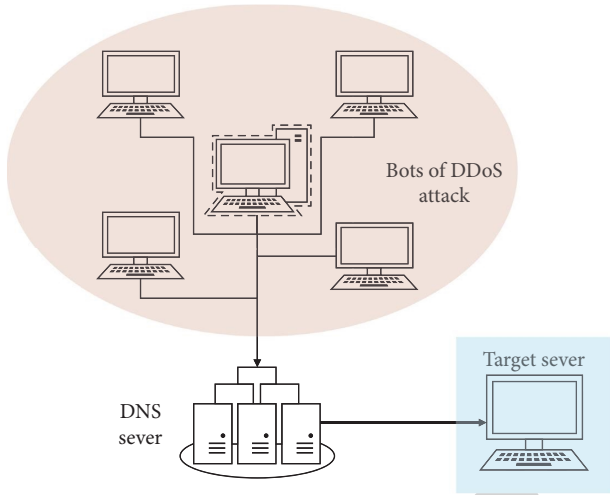

Figure 1: DDoS attack process.

5. Case Study

CICIDS-2017 dataset is employed to verify the practicability of this method [30, 31]. The dataset contains benign and the most up-to-date common attacks, which resembles the actual real-world data (PCAPs). Cases of high frequently used network flows are employed, including the traffic data of benign and DDoS attacks. It also includes the results of the network traffic analysis using CICFlowMeter with labeled flows based on the time stamp, source, and destination IPs, source and destination ports, protocols, and attack [31]. The DDoS attacks were implemented on Friday afternoon and captured in the dataset, which has a total of 225,747 flows, of which more than 40,000 DDoS attack flows. According to the official label, there are three bots, and their IP addresses are 205.174.165.69, 205.174.165.70, and 205.174.165.71. The network firewall IP addresses are 205.174.165.80 and 172.16.0.1, and the victim host IP address is 192.168.10.50. The flowchart of DDoS attacks process is shown in Figure 1.

These traffic data are set to a CSV file. Every flow has 83 properties in the CSV files, such as the timestamp, source, and destination IPs, source and destination ports, and flow duration, while one label exists, which can represent the DDoS attack or normal communication. Besides, the computer devices are as follows: CPU is i7-9700K; RAM is DDR4-96G; ROM is Intel SSD 1T; the operating system is Windows 10; and MATLAB 2018b is used.

Then, the combined PCA and FCM algorithm is used to create a DDoS detection model. The process of clustering is illustrated in Figure 2.

At first, the direct graph (DG) model is created within graph theory to reveal traffic data characteristics for both the victims and bots to generate the relationship between source and destination IP port structure. In the DG model, the vertices are presented by the combination of IP and ports. Two vertices directly point to the destination IP port from the source, called the edges. Some properties of edges can be selected as the input variables for detecting DDoS attacks. The most obvious characteristics are total forward packet, total backward packet, the standard deviation of backward packet length, total visit view, average packet length, flow duration, the standard deviation of flow interval time, and the mean active time of flow, which are extracted as the input variables to generate a matrix A by (1).

However, only edges can be found in the CSV file. The preprocessing is measured to extract the information of vertices by MATLAB. The second and third columns are the source IP and port, respectively. The content of two columns is extracted and combined as a node of DG. Meanwhile, the fourth and fifth columns are the destination IP and port. These two columns are also combined as a node pointed by the source IP port. In this study, a total of 2037 nodes are selected as graph-based features. These nodes can formulate 2000 edges, including 600 DDoS attack edges and 1400 normal edges (3 : 7).

Then, the eight characteristics are converted into edge features in a weight matrix X. The matrix X of dimensionality reduction in eight features should be solved in the PCA processing. To further reduce X's dimension, an appropriate k value should be determined. The proportion of the selection criterion of k value is solved by (9), and the results are shown in Table 5.

When the weights are three, the total proportion is 98.48%. These preceding three weights can be regarded as the essential factors for predicting DDoS attacks. Thus, only total forward packet, total backward packet, and standard deviation of backward packet length are retained. A new three-dimensional weight matrix can be solved by (11).

At last, the FCM clustering algorithm is employed to predict the DDoS flows. In this study, FCM clustering was performed within different partition values c. The power

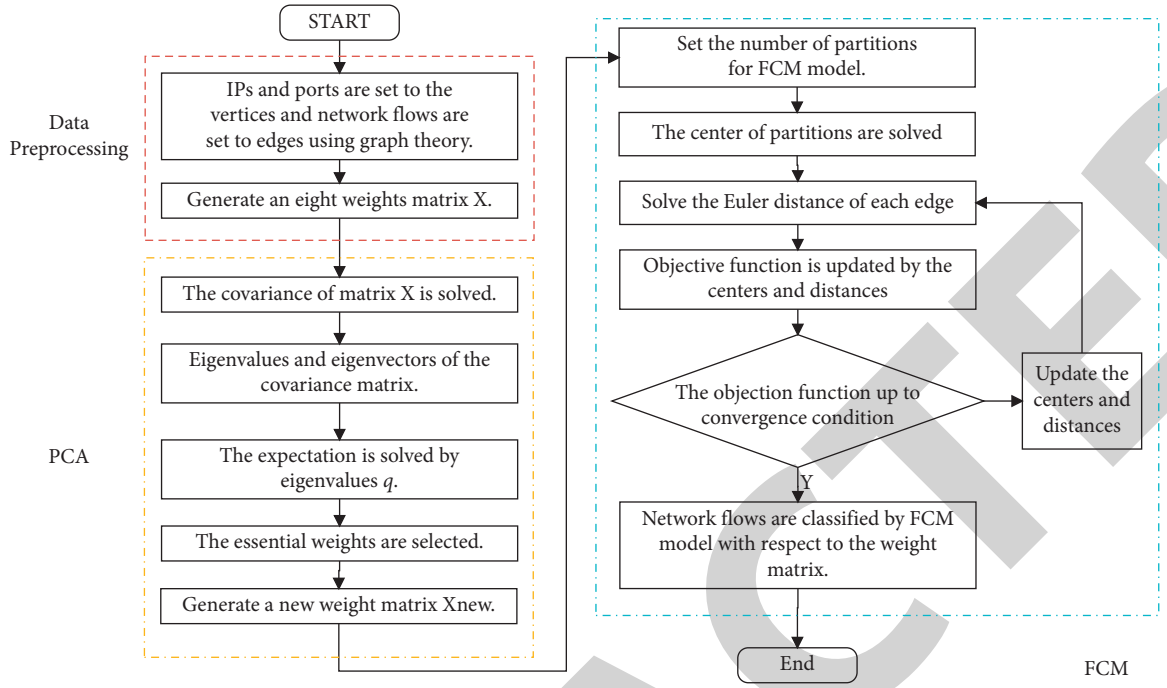

Figure 2: Flowchart of the clustering.

Table 5: Expectation values of the weight matrix.

Weight	w^1	w^2	w^3	w^4	w^5	w^6	w^7	w^8
Proportion (%)	93.83	3.13	1.51	0.76	0.58	0.14	0.05	0.00
Total proportion (%)	100.00	6.17	3.04	1.52	0.77	0.19	0.05	0.00

Table 6: Number of detection of DDoS attacks in the PCA-FCM.

Partition (c)	Normal communication	DDoS attack
2	1202	798
3	1377	623
4	1379	621
5	1367	633

Table 7: Detection efficiency with different c values in the PCA-FCM.

Partition (c)	Number of attack node	Recall (%)	False-positive rates (%)	True-positive rates (%)	True-negative rates (%)	False-negative rates (%)
2	600	100	9.90	60.10	0.00	30.00
3	600	100	1.05	68.95	0.00	30.00
4	600	100	1.75	68.25	0.00	30.00
5	600	100	1.65	68.35	0.00	30.00

index is set to 2.0, the tolerance is 10^{-5}, and the maximum iteration time is 100. In the clustering of different c values tested, all c values detect the attack edges, and the lowest false alarm rate is when $c = 3$. The center of three partitions is (0.45, 0.01, 0.01), (0.08, 0.02, -0.01), and (0.78, 0.02, -0.01). The results of detection efficiency are shown in Table 6, and

Table 8: Predicted results for the CICIDS-2017 dataset.

Method	Partition (c)	Dimension	Recall (%)	False positive (%)	True positive (%)	True negative (%)	False negative (%)
FCM	3	8	100	1.20	68.80	0.00	30.00
PCA-FCM	3	4	100	1.05	68.95	0.00	30.00
K-means	2	8	100	10.25	59.75	0.00	30.00
K-means	3	8	100	1.20	68.80	0.00	30.00
K-means	4	8	100	1.65	68.35	0.00	30.00
K-means	5	8	100	1.20	68.80	0.00	30.00
PCA-K-means	2	3	100	10.35	59.65	0.00	30.00
PCA-K-means	3	3	100	10.20	59.80	0.00	30.00
PCA-K-means	4	3	100	1.65	68.35	0.00	30.00
PCA-K-means	5	3	100	1.70	68.30	0.00	30.00
NMF-K-means [5]	2	4	100	4.80	65.20	0.00	30.00
NMF-K-means [5]	3	4	100	6.20	63.80	0.00	30.00
NMF-K-means [5]	4	4	100	2.03	67.97	0.00	30.00
NMF-K-means [5]	5	4	100	2.34	67.66	0.00	30.00

the recall, true-positive, true-negative, false-negative, and false-positive rates are shown in Table 7. No one DDoS attack is missed.

6. Comparisons

For verifying the effectiveness of this method for clustering the traffic data to normal and DDoS flows, the K-means clustering algorithm, and nonnegative matrix factorization (NMF), dimensional reduction model can be employed in the above case [32]. NMF was proposed in 1999, which makes all components after decomposition nonnegative, and at the same time realizes nonlinear dimension reduction. It corresponds to the intuitive understanding that the whole is made up of the parts, so it captures in a sense the nature of intelligent data description. Meanwhile, the pure clustering methods that the dimension is not reduced are also used to compare. The results of the two methods after optimization are shown in Table 8.

The PCA-FCM model in this study greatly affects DDoS attacks in network communications, compared with other methods. The recall rate, true negative rate, and false negative rate are 100%, 0%, and 30%, respectively, indicating that all DDoS attacks are detected. The false positives have decreased to 1.05%, while true positives have increased to 68.85%, when partition and dimension are 4 and 3, respectively.

7. Conclusion

This study presents a novel PCA-FCM model to detect DDoS attacks where the topological structure is taken into account between IP ports of source and destination. Then, characteristics, including total forward packet, total backward packet, the standard deviation of backward packet length, total visit view, average packet length, flow duration, the standard deviation of flow interval time, and mean active time of flow, are considered input variables for clustering. The PCA model is employed to reduce the dimensions of features further. Then, the bots are detected by FCM clustering with these features. The CICIDS-2017 dataset is employed to verify this method in the case study. The results demonstrate that the method has a high detecting reliability. The PCA-FCM method is suitable for DDoS detection. The recall, true negative, and false negative are 100.00%, 0.00%, and 30.00% that means no one DDoS attack is missed. The false positive and true positive are 1.05% and 68.95% compared with FCM, which has a considerable improvement.

With respect to the results, the PCA-FCM model has three advantages. Firstly, FCM uses unsupervised training and does not require labels; secondly, the topological structure relationship between IP and ports is connected by a DG structure. Thirdly, input variables can be automatically selected by PCA within many factors of dataset to reduce the overload of calculation. Therefore, this method provides a new horizon to network security.

However, some disadvantages can be discovered. Firstly, the vertex property is not considered an input variable for clustering. Secondly, the number of partitions should be calculated automatically in the clustering algorithm. Thirdly, a supervised model can be applied further to recognize new data after clustering by the edges label. Therefore, further research and improvement of this method should be conducted in the future to accurately and quickly detect DDoS attacks.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

[1] A. Bhardwaj, G. V. B. Subrahmanyam, V. Avasthi, H. Sastry, and S. Goundar, "DDoS attacks, new DDoS taxonomy and mitigation solutions - a survey," in *Proceedings of the 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pp. 793–798, Paralakhemundi, India, October 2016.

[2] X. Chen, Z. Jiang, H. Li, J. Ma, and P. S. Yu, "Community hiding by link perturbation in social networks," *IEEE Transactions on Computational Social Systems*, vol. 8, no. 3, pp. 704–715, 2021.

[3] M. E. Ahmed, S. Ullah, and H. Kim, "Statistical application fingerprinting for DDoS attack mitigation," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1471–1484, 2019.

[4] W. Pan and W. Li, "A hybrid neural network approach to the classification of novel attacks for intrusion detection," *Parallel and Distributed Processing and Applications*, pp. 564–575, Berlin, Germany, November 2005.

[5] H. Jing and J. Wang, "DDoS detection based on graph structure features and non-negative matrix factorization," *Concurrency and Computation: Practice and Experience*, vol. 71, 2020.

[6] S. Kobayashi, K. Otomo, K. Fukuda, and H. Esaki, "Mining causality of network events in log data," *IEEE Transactions on Network and Service Management*, vol. 15, no. 1, pp. 53–67, 2018.

[7] W. Cui, R. H. Katz, and W. Tan, "BINDER: an extrusion-based break-in detector for personal computers," in *Proceedings of the Annual Conference on USENIX Annual Technical Conference*, p. 18, Anaheim, CA, USA, April 2005.

[8] S. Urushidani, M. Aoki, K. Fukuda et al., "Highly available network design and resource management of SINET4," *Telecommunication Systems*, vol. 56, no. 1, pp. 33–47, 2014.

[9] K. S. Sahoo, D. Puthal, M. Tiwary, J. J. P. C. Rodrigues, B. Sahoo, and R. Dash, "An early detection of low rate DDoS attack to SDN based data center networks using information distance metrics," *Future Generation Computer Systems*, vol. 89, pp. 685–697, 2018.

[10] R. Karimazad and A. Faraahi, "An anomaly-based method for ddos attacks detection using rbf neural networks," in *Proceedings of the 2011 International Conference on Network and Electronics Engineering*, Singapore, September 2011.

[11] "Information theory, inference, and learning algorithms," *Kybernetes*, vol. 33, no. 7, pp. 1217-1218, 2004.

[12] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[13] E. Kay, "Graph theory with applications," *Journal of the Operational Research Society*, vol. 28, no. 1, pp. 237-238, 1977.

[14] N. R. Malik, "Graph theory with applications to engineering and computer science," *Proceedings of the IEEE*, IEEE, vol. 63, no. 10, pp. 1533-1534, 1975.

[15] S. Chowdhury, M. Khanzadeh, R. Akula et al., "Botnet detection using graph-based feature clustering," *Journal of Big Data*, vol. 4, no. 1, p. 14, 2017.

[16] L. Nie, Y. Wu, X. Wang et al., "Intrusion detection for secure social internet of things based on collaborative edge computing: a generative adversarial network-based approach," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 1, pp. 134–145, 2022.

[17] J. Chen, Y. Chen, L. Chen, M. Zhao, and Q. Xuan, "Multiscale evolutionary perturbation attack on community detection," *IEEE Transactions on Computational Social Systems*, vol. 8, no. 1, pp. 62–75, 2021.

[18] I. Jolliffe, "Principal component analysis," in *Encyclopedia of Statistics in Behavioral Science*John Wiley & Sons, Chichester, UK, 2005.

[19] B. Zhang, Z. Liu, Y. Jia, J. Ren, and X. Zhao, "Network intrusion detection method based on PCA and bayes algorithm," *Security and Communication Networks*, vol. 2018, Article ID 1914980, 11 pages, 2018.

[20] P. Favaro, R. Vidal, and A. Ravichandran, "A closed form solution to robust subspace estimation and clustering," in *Proceedings of the 2011*, pp. 1801–1807, Colorado Springs, CO, June 2011.

[21] Z. Shen, Y. Zhang, and W. Chen, "A bayesian classification intrusion detection method based on the fusion of PCA and LDA," *Security and Communication Networks*, vol. 2019, Article ID 6346708, 11 pages, 2019.

[22] Y. Xu, G. Yang, and S. Bai, "Laplace input and output perturbation for differentially private principal components analysis," *Security and Communication Networks*, vol. 2019, Article ID 9169802, 10 pages, 2019.

[23] L. M. Elshenawy, S. Yin, A. S. Naik, and S. X. Ding, "Efficient recursive principal component analysis algorithms for process monitoring," *Industrial & Engineering Chemistry Research*, vol. 49, no. 1, pp. 252–259, 2010.

[24] R. Wang and Y. Zhou, "Flower pollination algorithm with dimension by dimension improvement," *Mathematical Problems in Engineering*, vol. 2014, Article ID 481791, 9 pages, 2014.

[25] Z. Moghaddasi, H. A. Jalab, R. Md Noor, and S. Aghabozorgi, "Improving RLRN image splicing detection with the use of PCA and kernel PCA," *The Scientific World Journal*, vol. 2014, Article ID 606570, 10 pages, 2014.

[26] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Springer US, Boston, MA, 1981.

[27] T. Mu, M. Huang, H. Tan, G. Chen, and R. Zhang, "Pressure and water quality integrated sensor placement considering leakage and contamination intrusion within water distribution systems," *ACS ES&T Water*, vol. 1, no. 11, pp. 2348–2358, 2021.

[28] M. Gong, Y. Liang, J. Shi, W. Ma, and J. Ma, "Fuzzy C-means clustering with local information and kernel metric for image segmentation," *IEEE Transactions on Image Processing*, vol. 22, no. 2, pp. 573–584, 2013.

[29] Q. Tang, Y. Zhao, Y. Wei, and L. Jiang, "Research on the mental health of college students based on fuzzy clustering algorithm," *Security and Communication Networks*, vol. 2021, Article ID 3960559, 8 pages, 2021.

[30] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, pp. 108–116, Funchal, Madeira, Portugal, January 2018.

[31] A. Boukhamla and J. Coronel, "CICIDS2017 Dataset: Performance Improvements and Validation as a Robust Intrusion Detection System Testbed." *International Journal of Information and Computer Security*, vol. 16, no. 1-2, 2019.

[32] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.