International Journal of Communication Networks and Information Security
ISSN: 2073-607X, 2076-0930
Volume 14 Issue 1s Year 2022 Page 01:08

A DDoS Attack Detection using PCA Dimensionality Reduction and Support Vector Machine

***Bhargavi Goparaju[1], Dr. Bandla Srinivasa Rao[2]**

[1]Research Scholar, Department of Computer Science and Engineering, Acharya Nagarjuna University, Guntur, Andhra Pradesh, India.
gbhargavi5007@gmail.com
[2]Research Guide, Department of Computer Science and Engineering, Acharya Nagarjuna University, Guntur, Andhra Pradesh, India.
sreenibandla@gmail.com

Article History

Received: 12 July 2022
Revised: 22 September 2022
Accepted: 14 October 2022

Abstract

Distributed denial-of-service attack (DDoS) is one of the most frequently occurring network attacks. Because of rapid growth in the communication and computer technology, the DDoS attacks became severe. So, it is essential to research the detection of a DDoS attack. There are different modes of DDoS attacks because of which a single method cannot provide good security. To overcome this, a DDoS attack detection technique is presented in this paper using machine learning algorithm. The proposed method has two phases, dimensionality reduction and model training for attack detection. The first phase identifies important components from the large proportion of the internet data. These extracted components are used as machine learning's input features in the phase of model detection. Support Vector Machine (SVM) algorithm is used to train the features and learn the model. The experimental results shows that the proposed method detects DDoS attacks with good accuracy.

Keywords: DDoS attack, Feature extraction, Model detection, Machine learning.

1. Introduction

For integrating multiple computes as an attack platform, Denial-of-service (DDoS) attack utilizes for launching attacks several targets for increasing attack power based on the client or server technology [1]. The conventional peer-to-peer attack mode has changed by the distributed denial-of-service attack, and the statistical rule is not there for attack behaviour. In the attack, standard protocols and services utilize additionally. The attack's differentiation is a challenging task or normal behaviour only based on the kinds of protocols and services. It's a tough task to detect the distributed denial-of-service attack [2]. Using the network intrusion detection technique, the research has made on defence technology against a DDoS attack. For demonstrating the attack characteristics, three different features [3-5] involving the flow density, the number of output ports, and the number of sources IP addresses utilize based on the features of many-to-one attack in the process of DDoS. The attack flows' rationality can differentiate by these techniques, but less information about message uses that only includes the source IP address and the output port. The specific type of attack cannot determine,

so the higher detection rate achieves. Machine learning is a crucial role in predicting the attacks. The detection progression of DDoS attack could involve using machine learning that includes specifically support vector machine, hidden Markov model, and naïve Bayesian algorithm [6]. For modelling the network data stream, Tama's team used the anomaly detection technique [7] based on the header attribute. In previous literature, the detection accuracy could improve up to a certain extent using the methods, but not using the data stream context completely [8].

A detection method of DDoS attack detection proposes using the machine learning technique is presented in this paper. Three common DDoS attack patterns could be identified in the feature extraction phase. The attack flow characteristics obtain by using the standard flow data analysis. In model detection, the attack traffic's characteristics retrieve that trained in the training model using the algorithm of random forest. Section II contains the proposed framework, followed by experimental results and conclusion.

2. Proposed framework

The proposed framework consists of two parts. Principle Component Analysis (PCA) based dimensionality reduction and SVM based classification. This section explains in detail, the two phases in the proposed framework.

2.1 Principle Component Analysis (PCA)

PCA has targeted to the data set's dimensionality that contains different variables correlated each other. In the dataset, the available variation retains up to the maximum extent. Same task performs based on the variables' transformation into a set called as the principal components or PCs. The nature of variables is orthogonal and they order in such a way that the variation's retention reduces in the original variables as moving down in the order. The maximum variation retains in the 1st principal component and it was presented in the original components. Here, the PCs refer to the covariance matrix's eigenvectors and are orthogonal. The figure1 shows the High redundancy and Low redundancy.

PCA includes the supplying of user using a lower-dimensional picture and it is a projection or "shadow" of the object if the most informative viewpoint considers.

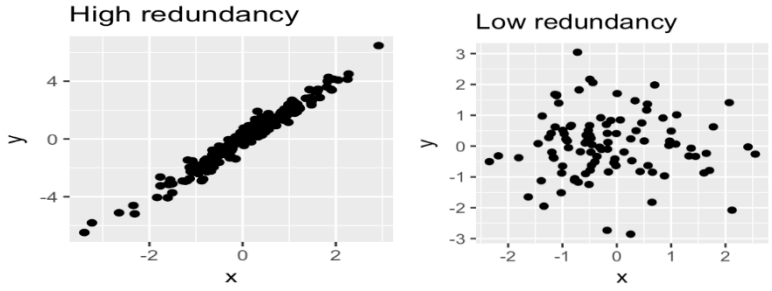

Figure .1. High Redundancy And Low Redundancy

- Dimensionality: It indicates the number of columns in dataset or the number of features simply.
- Correlation: It demonstrates the strength of relating two variables each other and ranging for -1 to +1. The value greater than zero indicates the increment of one with the increased value of other. The negative refers to the decrement of another variable on increasing the previous variable. The relation strength shows by the modulus value.
- Orthogonal: The corelation between the variable in zero.
- Eigenvectors: Eigenvectors refer to a big domain, let us consider the restriction on the knowledge of the same which we would need and a non-zero vector as $\boldsymbol{v}$. If $\boldsymbol{Av}$ is a scalar multiple of **v**, it is a square matrix ***A's*** eigen vector. Or simply:

$$\boldsymbol{Av} = \lambda \boldsymbol{v} \quad \textbf{Eq (1)}$$

Where, λ indicates the associated eigenvalue, and $\boldsymbol{v}$ is the eigenvector.

- Covariance Matrix: The covariances between the variables' pair contain in this matrix. The covariance between i^{th} and j^{th} variable shows by the (i, j)th element.

2.2 Properties of Principal Component

The optimally-weighted observed variables with a linear combination is the definition of a principal component technically. These principal components are the output of PCA. The number of PCs is lower or equivalent to the quantity of the original variables. Some useful properties include in the PCs such as:

1. In the PCs, the original variables with linear combinations are included specifically. In this combination, the weights vector is found as the eigenvector that satisfies the principle of least squares.
2. The PCs are orthogonal.
3. As moving from the 1st PC to the last one, the variation in the PCs reduce.

Sometimes, the PCs are useful least importantly in outlier detection, regression, etc.

2.3 Implementation of PCA

Step 1: Normalize the data

The data normalization is the first step for proper operation of PCA through the subtraction of respective means from the respective column's numbers. All X become ***x***-for a dimension X and all Y become ***y***-for a dimension Y.A dataset produces that whose mean is zero.

Step 2: Covariance matrix

As we have considered the dataset as two-dimensional, it will result in a Covariance matrix with the size of 2×2.

$$Matrix\ (Covariance) = \begin{bmatrix} Var[X_1] & Cov[X_1, X_2] \\ Cov[X_2, X_1] & Var[X_2] \end{bmatrix} \quad \text{Eq (2)}$$

$$Var[X_1] = Cov[X_1, X_1]\ and\ Var[X_2] = Cov[X_2, X_2] \quad \text{Eq (3)}$$

Step 3: Eigenvalues and eigenvectors

In the next step, eigenvectors and eigenvalues are calculated for the covariance matrix. As it is a square matrix, the determination of values is possible. For a matrix ***A***, $\boldsymbol{\lambda}$ is an eigenvalue if a solution derives for the characteristic equation:

$$\boldsymbol{det(\lambda I - A) = 0} \quad \textbf{Eq (4)}$$

Where, ***I***indicates the same dimension's identity matrix as ***A*** which requires a condition for the subtraction of matrix and the matrix determinant is indicated as '***det'.*** By solving below equation, a corresponding eigen-vector ***v*** determinate for each eigenvalue $\boldsymbol{\lambda}$:

$$\boldsymbol{(\lambda I - A)v = 0} \quad \textbf{Eq (5)}$$

Step 4: Reduced dimensionality vector

From the order of largest to smallest, the eigenvalues are arranged that helps to provide the components in significance. The dimensionality with a reduction part is considered. The corresponding n eigenvalues and eigenvectors when a dataset with n variables. It results the principal component of a dataset is the eigenvector relates to the highest eigenvalue and it is required to select the number of eigenvalues to make an analysis on principal components. The first eigenvalues of *p* choose for reducing the dimensions and the test ignores. When small eigenvalues are resulted, we do not lose much data, but some information lose in the process.

A feature vector forms as a matrix of vectors, the eigenvectors which we want to consider only. We can select the one related to the greater eigenvalue or take both as 2 dimensions have included in the running example.

Feature Vector = (eig_1, eig_2)

Step 5: Forming PCs:

The final step is forming of principal components based on the math operations. The feature vector's transpose considers and left-multiply with the original dataset's transpose of scaled version.

$$NewData = FeatureVector^T x\ Scaledata^T \quad \text{Eq (6)}$$

Where, *NewData* indicates the matrix that consists the principal components, *FeatureVector* refers to the formed matrix based on the chosen eigenvectors, and *ScaledData* is the original dataset's scaled version.
T is the superscript which refers to the matrix transpose formed through the interchanging of rows to columns and vice versa. A transpose of size 3x2 for a 2x3 matrix.
The eigenvectors provide the data about patterns. When the eigenvectors plot on the scatterplot of data in the running example of 2-D set, the principal eigenvector is found out that it fits well with the provided information (relates to the largest eigenvalue). Much information does not carry out by the other one, being perpendicular to it. If deprecating it, much loss does not happen and the dimension reduces.

2.4 Support Vector Machine (SVM)

SVM is an essential versatile machine learning algorithm that has an ability to make nonlinear and linear classification, outlier detection, and regression. For both regression and classification problems, machine learning people use the support vector machines or SVM widely but especially used for classification tasks. Due to the features of less computation and notable accuracy, the most preference will give to SVM over other classification algorithms. Although less data is there, reliable results provide by SVM. If the labelled data set provides to the algorithm in the training set, the generalization between two different classes could be possible with SVM. SVM has an important function of verifying whether the hyperplane can differentiate the two classes. The similar type of task can perform by many hyperplanes but the main objective of hyperplane is determining it with the highest margin, i.e. the maximum distances between two classes. Test features the easier classification could be possible in future.

2.5 SVM Working

By considering the example of two classes as shown in the below Figure 2 which includes a class A: Circle & class B: Triangle, the working of SVM could understood. The best hyperplane determines that categorizes the both classes to implement the SVM algorithm.

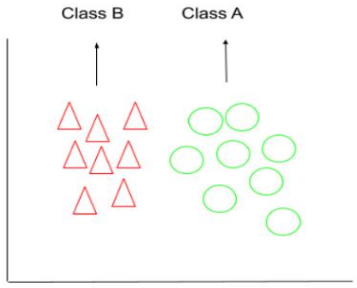

Figure 2. Class A and B

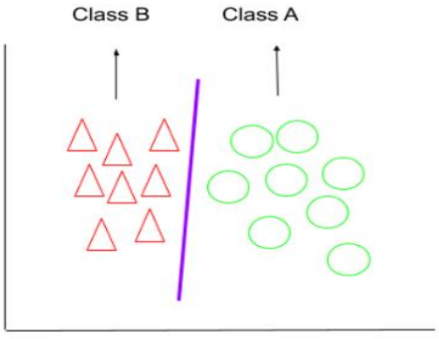

Figure 3. Labelled Data

SVM provides a line called as ***'Hyperplane*** 'based on the points which categorizes both classes. Here, the line known as ***'Decision boundary'***. In the circle class, anything that resembles a circle will be categorized.

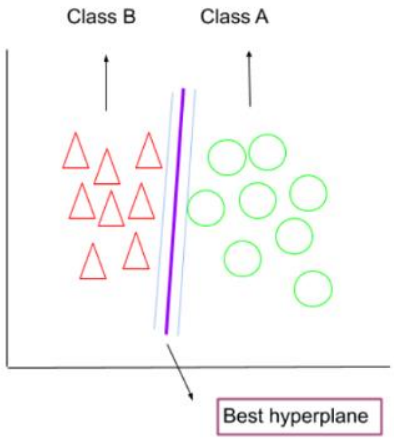

Figure 4. All Hyperplanes Are Not Good at Classification

Many hyperplanes can be there but the best hyperplane categorizes the data accurately with minimum distance from the data points in the dataset. Such type of best hyperplanes is the main objective of SVM.
By depending on the features, different dimensions are there. If the features are more than 3, it's difficult to visualize.

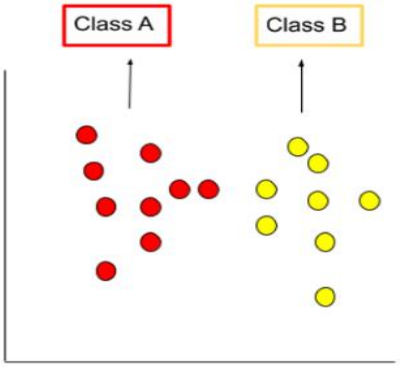

Figure 5. Class A- Red & Class- B Yellow

Two classes are considered. The best hyperplane between them is required to determine and it categorizes the two classes.

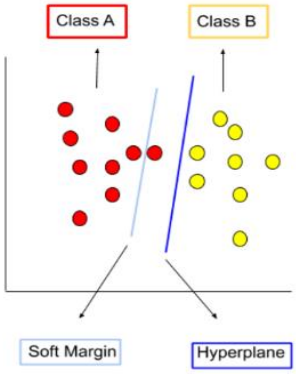

Figure 6. Soft margin and hyperplane

Some of the a forementioned data points allow in the soft margin to get misclassified. To maximize the margin and make less misclassifications.

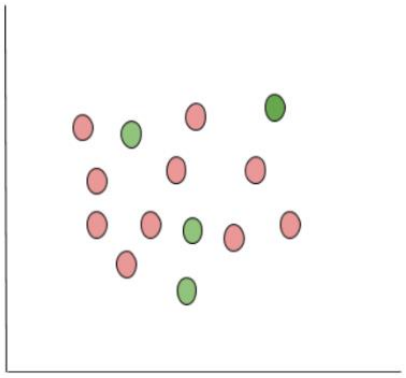

Figure 7. Non-Linear Dataset

As shown in the above figure7, kernel tricks use by the SVM to make the data to linearly separable if it is non-linearly separable. **Cover's theorem** defines as the transformation of information of non-linearly separable into linearly separable - "***given a set of training data that is not linearly separable, with high probability it can be transformed into a linearly separable training set by projecting it into a higher-dimensional space via some non-linear transformation***".
The data points project to the space with higher dimensionality using kernel tricks and they become more easily separable.

3. Results

The total dataset is classified in DDoS attacks and normal data. The total dataset has 113270 samples and it is divided into training and testing. Table I has Training and testing sample datasets. In training process, 45927 samples and 67343 samples include in DDoS and Normal respectively. In testing process, DDoS has 7458 samples and Normal has 9711 samples. PCA uses for dimensionality reduction and SVM classification methods utilize for classification.

Table 1. Training and Testing Samples

	DDOS	**Normal**
Training	45927	67343
Testing	7458	9711

Table 2 represents the SVM Training classification confusion matrix. The accuracy results of DDoS and Normal include 97.57% and 96.77% respectively. The total accuracy is 97.17%.

Table 2. Training Confusion Matrix

	DDOS	**Normal**	**Accuracy (%)**	**Error (%)**
DDOS	44814	1113	97.57	2.43
Normal	4194	63149	96.77	6.23
			Over all accuracy	97.17

Table 3 represents the SVM Testing classification confusion matrix. The Normal accuracy (no attack) is 92.80% and the DDoS accuracy is 94.87%. The total percentage of accuracy is 93.83%.

Table 3. Testing Confusion Matrix

	DDOS	Normal	Accuracy (%)	Error (%)
DDOS	7076	382	94.87	5.13
Normal	699	9012	92.80	7.2
			Over all accuracy	93.83

The confusion matrices are shown in tables 2 and 3. The training accuracy is 97.17 and the testing accuracy is 93.83.

4. Conclusion

A new technique of DDoS attack detection proposes that includes SVM and PCA model using machine learning. For extracting the traffic characteristics of DDoS attack with a large proportion, format conversion and feature extraction are carried out based on the extraction of protocol attack packets of the tool. By using the technique of PCA dimensionality reduction, the large proportion data reduces. The features utilize like the inputs of machine learning. For training purpose and obtain the detection model of DDoS attack, SVM algorithm utilizes. For model test, the data about normal traffic mix with attack data. A good detection rate results by the proposed detection method using machine learning for the current DDoS attacks based on the analyzation of experimental outcomes.

References

[1] Zargar S T，Joshi J，Tipper D．A survey of defence mechanisms against distributed denial of service (DDoS)flooding attacks[J]．IEEE Corn— communications Surveys＆Tutorials。2013，15(4)：2046—2069．

[2] Wang Bing，Zheng Yao，Lou Wenjing，et a1．DDoS attack protection in the era of cloud computing and software—defined networking[J]．Computer Networks，2015，81(4)：308—319．

[3] Yu Shui，Tian Yonghong，Cuo Song，et a1．Can we beat DDoS attacks in clouds? IEEE Trans on Parallel and Distributed Systems，2014，25(9)：2245—2254．

[4] Kotenko I ，Ulanov A ．Agent—based simulation of DDOS attacks and de— fense mechanisms[J]．International Joumal of Computing，2014，4(2)：113—123.

[5] Gupta B B，Joshi R C，Misra M．ANN based scheme to predict number of zombies in a DDoS attack f J]．Intimation JoumaI of Network Security，2012，14(2)：61-70.

[6] Yu Penchen, Qi Yong, Li Qianmu. DDoS attack detection method based on random forest classification model [J]. Application Research of Computers, 2017, 34(10):3068-3072(in Chinese).

[7] Tama B A，Rhee K H．Data mining techniques in DoS／DDoS attack detection：a literature review Proc of the 3rd International Conference on Computer Applications and Information Processing Techno— logy．2015：23-26.

[8] Hoda, S. A., & Mondal, A. C. (2022). A study of data security on E-governance using steganographic optimization algorithms. International Journal on Recent and Innovation Trends in Computing and Communication, 10(5), 13-21. doi:10.17762/ijritcc.v10i5.5548

[9] Tan Miao. Research and Implementation of DDoS Attack Detection Based on Machine Learning in Distributed Environment [D],2018(in Chinese).

[10] Goar, D. V. . (2021). Biometric Image Analysis in Enhancing Security Based on Cloud IOT Module in Classification Using Deep Learning- Techniques. Research Journal of Computer Systems and Engineering, 2(1), 01:05.

[11] Zellar, P. I. . (2021). Business Security Design Improvement Using Digitization. International Journal of New Practices in Management and Engineering, 10(01), 19–21. https://doi.org/10.17762/ijnpme.v10i01.98

[12] Dhabliya, D. (2021). Feature Selection Intrusion Detection System for The Attack Classification with Data Summarization. Machine Learning Applications in Engineering Education and Management, 1(1), 20–25.

[13] Prasad, A., Prasad, S., Arockiasamy, K., Karthika, P., & Yuan, X. (2022). Detection of DDoS attack in software-defined networking environment and its protocol-wise analysis using machine learning. International Journal of Intelligent Systems and Applications in Engineering, 10(3), 147-153.