International Core Journal of Engineering Volume 8 Issue 4, 2022
ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

# Botnet Detection based on Manifold Learning

Kai Zhang[1], Peng Zhou[2], Yujing Fang[3]

[1] China Financial Certification Authority, Beijing 100054, China

[2] CFETS Information Technology, Shanghai 201203, China

[3] Chengfang Financial Information Technology Services Co. LTD, Shanghai 201201, China

## Abstract

**Botnet refers to the use of one or more means of transmission, will be a large number of hosts infected with bot program (zombie program) virus, so as to form a one-to-many control network between the controller and the infected host. Manifold learning is a basic method in pattern recognition, which is divided into linear manifold learning algorithm and nonlinear manifold learning algorithm. In this paper, dimension reduction method based on manifold learning is used, combined with a variety of machine learning algorithms, to compare and identify botnets.**

## Keywords

**Machine Learning; Cybersecurity; Botnet.**

## 1. Introduction

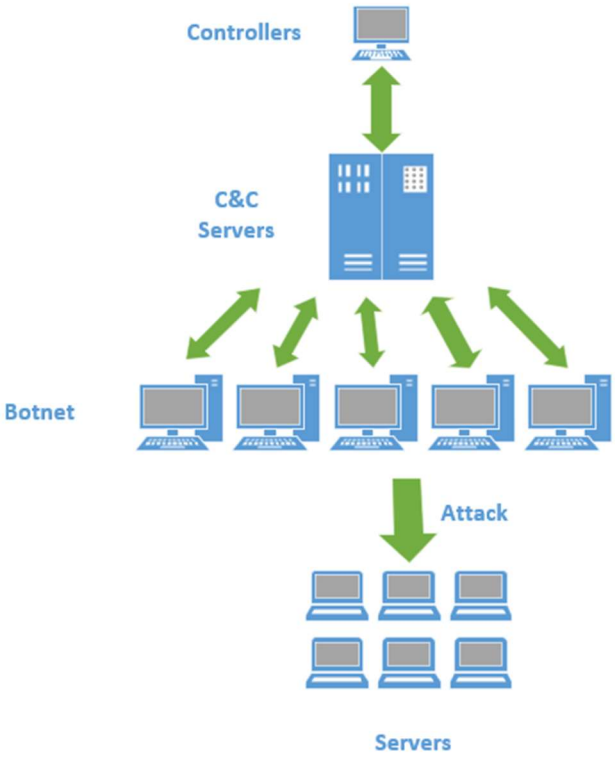


**Figure 1.** Schematic diagram of centralized structure

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

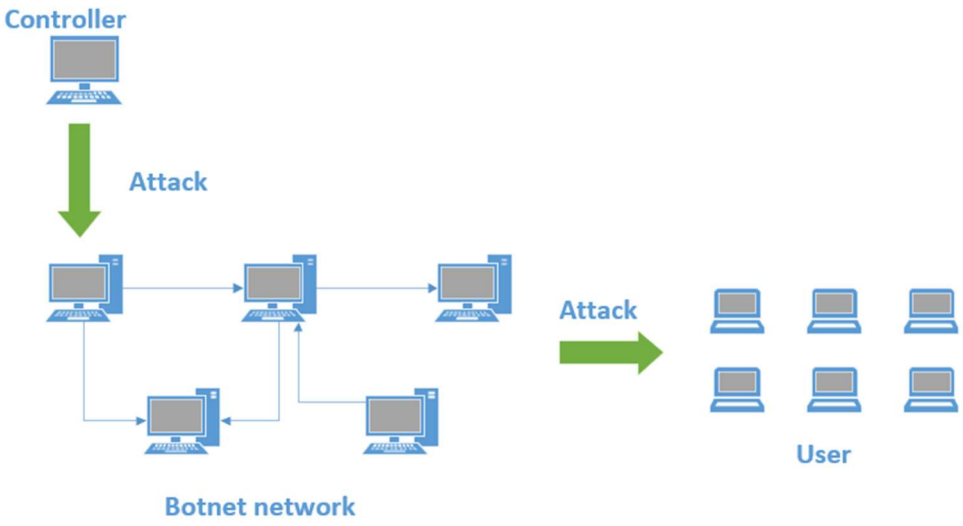


**Figure 2.** SCHEMATIC diagram of P2P structure

With the development of network security, the incidence of cyber crime is also increasing. Sophisticated types of cyberattacks have come to be considered the norm as their attacks become more frequent and varied. This constant evolution of the status quo also requires network security researchers in the field of network security defense needs continuous innovation. Botnet is one of the most serious threats to network security, and its attack means are increasingly hidden, causing great harm to the society. The controlled host of a botnet can be under the control of a command and control (C&C) server to implement a variety of cybercrimes, such as mass DDos attacks, spam, phishing links, and more.

Botnets can generally be divided into five architectures: centralized structure, P2P structure, hybrid structure, Equity structure and super botnet structure [1]. The most common are centralized and P2P structures. For centralized botnets, the most popular are IRC protocol, HTTP or POP3, as shown in Figure 1. For P2P structure, zombie computers control each other across each other's firewalls, as shown in Figure 2.

In recent years, the academic circle has also proposed a variety of methods for botnet detection. For example, the honeypot was used to obtain samples of malicious code and then analyzed at the host level to screen out botnet programs [2]. The high-entropy detector based on BotHunter proposed by Han Zhang [3] can detect the encrypted botnet. Huabo Li [4] et al. introduced the features of Irregular phase Similarity and could effectively identify P2P puppet machines, commonly known as broilers. Although these methods could effectively detect botnets, However, potential botnets are screened through rules base similar to black-and-white lists. However, new botnets cannot be effectively detected. In recent years, with the development of machine learning and artificial intelligence technology, many scholars in the field of network security are trying to introduce machine learning algorithms into botnet detection. Compared with traditional statistical methods, machine learning methods are more efficient and intelligent, because machine learning algorithms can learn potential patterns from past data, thus entering a virtuous cycle and improving detection accuracy [5].

At present, the detection results of most botnets in the field of machine learning are not very good, and the important reason is that some researchers [6] still choose to use their previous experience and understanding in the field to extract features due to the influence of high-dimensional features in the data processing botnet traffic. Such methods require a strong understanding of the field and are not easy to control in terms of time costs.

With the development of machine learning field, many scholars research on characteristics of the project and data dimension reduction with further research, including manifold learning (manifold learning) as a new type of supervised learning method, in the face of the characteristics of nonlinear,

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

high dimension data has important significance, will be better able to find the essence of such data dimension, It has better significance for later data analysis [7]. This paper studies how to use manifold learning technology to effectively reduce the dimension of botnet traffic data, mining the essential dimension of its data, speeding up the speed of data mining stage, but also can detect botnet more effectively.

## 2. Relevant Methods

### 2.1 Mainfold Learning

Manifold learning is a concept based on Riemannian geometry and differential manifold. The concept of Manifold comes from topology, which is that in a topological space, A local place is Euclid, also known as locally Euclid. In the local Euclidean, $R^{\mathrm{m}}$ represents the M-dimension Euclidean space. In the Euclidean space, any point has the concept of neighborhood, and the topology of neighborhood is the same as that of the open unit circle in the $R^{\mathrm{m}}$ Euclidean space, so that all local coordinates can be realized [7]. Its core idea is to extract its essential dimension from high-dimensional data and simplify complexity. With the development of manifold learning, many effective manifold learning algorithms have emerged in recent years. For example, ISOMAP, Locally linear embedding, T-distributed stochastic neighbor embedding (T-SNE) and so on. In this paper, ISOMAP, LLE and T-SNE will also be used to feature the botnet data.

### 2.2 Locally Linear Embedding

Local linear embedding algorithm is one of the most representative algorithms in manifold learning, and its core principle is to maintain the local order relation between data in essential space and data in embedded space [8]. In higher dimensions $R^n$ defines datasets $X=\{x_1,x_2,x_3\ldots,x_n\}$, In the space $R^m$ finding the corresponding data after dimensionality reduction (low dimension) $Y=\{y_1,y_2,y_3\ldots,y_n\}$. In higher dimensions $R^n$,

$$X_i=\varphi_{ij}x_j+\varphi_{ik}x_k+\varphi_{il}x_l \tag{1}$$

x represents higher dimensional space, $\varphi$ represents the distance between data points and data points in high-dimensional space, the core algorithm of LLE also hopes that Formula (1) can be maintained in low-dimensional space after dimensionality reduction. The specific steps of the algorithm are as follows:

Step 1: Matrix the Loss Function, $W_i$ represents k collection of neighborhood samples.

$$L(\varphi)=\sum_{i=1}^{m}\left\|x_i-\sum_{j\in W(i)}\varphi_{ij}x_j\right\|_2^2 \tag{2}$$

$$=\sum_{i=1}^{m}\left\|\sum_{j\in W(i)}\varphi_{ij}(x_i-x_j)\right\|_2^2 \tag{3}$$

$$=\sum_{i=1}^{m}\phi_i^T(x_i-x_j)(x_i-x_j)^T\phi_i \tag{4}$$

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

And, $\phi_i = (\varphi_{i1}, \varphi_{i2}, \varphi_{i3}, \ldots, \varphi_{ik})^T$ .Set $P = (x_i - x_j)(x_i - x_j)^T$ ,The matrix equation (3) can be obtained $\sum_{j \in W(i)} \varphi_{ij} = 1$.

Step 2: Use Lagrangian day multiplication to synthesize as an optimization target:

$$L(\phi) = \sum_{i=1}^{k} \phi_i^T P_i \phi_i + \lambda(\phi_i^T 1_k - 1) \tag{5}$$

The final weight coefficient is:

$$\phi_i = \frac{P_i^{-1} 1_k}{1_k^T P_i^{-1} 1_k} \tag{6}$$

Step 3: After obtaining the weight coefficient of the higher dimension, the core purpose is to maintain the same weight coefficient of the lower dimension. Even in low-dimensional data sets, the loss function is minimized:

$$Q(k) = \sum_{i=1}^{m} \left\| y_i - \sum_{j \in W(i)} \varphi_{ij} y_j \right\|_2^2 \tag{7}$$

Finally get the data in low dimensional space after dimensionality reduction, $Y = \{y_1, y_2, y_3 \ldots, y_n\}$.

## 2.3 (ISOMAP)

ISOMAP is a variation of multidimensional scale conversion MDS (Scaling) dimension reduction method. The core idea of the algorithm is to find high dimensional data $X = \{x_1, x_2, x_3 \ldots, x_n\}$ Neighborhood K in each sample keeps the distance between neighborhood and sample, and the orbit between samples outside the neighborhood and the sample is not connected. Then, the distance matrix between any two samples is reconstructed, and finally the distance between samples is used as the input of the standard MDS, so as to obtain the low-dimensional mapping of data $Y = \{y_1, y_2, y_3 \ldots, y_n\}$ [7].

## 2.4 (T-distributed Stochastic Neighbor Embedding)

T-distributed neighbor embedding (T-SNE) is a derivative of SNE algorithm, which is also a manifold learning algorithm. The biggest difference between T-SNE and ISOMAP and LLE algorithms is that they are not based on the core idea of distance invariant. In low - dimensional mapping, data cluster is too dense.

Its algorithm is described in detail in literature [10] [11] and will be briefly described in this paper. First, we get a dimension reduction $X = \{x_1, x_2, x_3 \ldots, x_n\}$, This algorithm converts the Euclidean distance between high dimensional data into conditional probability $p_{j|i}$, The probability density distribution is a Gaussian function centered on $x_i$, and the variance is $\sigma_i$, and the conditional probability can be expressed as:

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

$$p_{j|i} = \frac{\exp(-\left\|x_i - x_j\right\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\left\|x_i - x_k\right\|^2 / 2\sigma_i^2)} \tag{8}$$

The similarity of data points $x_i$ and $x_j$ is calculated and expressed by defining joint probability:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N} \tag{9}$$

In low-dimensional mapping space, conditional probability in Equation (8) is also used:

$$q_{j|i} = \frac{\exp(-\left\|y_i - y_j\right\|^2)}{\sum_{k \neq i} \exp(-\left\|y_i - y_k\right\|^2)} \tag{10}$$

KL divergence is used to express the difference between high-dimensional space and low-dimensional space data, where Q is the high-dimensional space structure, W is the low-dimensional space structure, and C is the cost function, expressed as follows:

$$C = KL(Q \| W) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{11}$$

The remaining parameter to choose is the bandwidth variance of the Gaussian distribution, because in most cases the density of higher-dimensional data is irregular, so it is unlikely to use a single value for all data points. The parameter "Perplexity" proposed in the reference [11] is a good expression for the above problems, and the expression is:

$$Perp(P_i) = 2^{-\sum_j p_{ij} \log_2 p_{ij}} \tag{12}$$

For each$x_i$ point in the calculation process, the Perplexity parameter is optimized until the Perp parameter reaches the user-set threshold. In practical application, for extremely high dimensional data, a large Perp threshold will enable t-SNE algorithm to obtain a very good result [10].

## 3. Model Design

As shown in Figure 3, the specific structure of botnet detection model is that after network traffic data is obtained, a preliminary data pretreatment is carried out first, then three manifold learning algorithms are used to perform feature engineering on the pre-processed data, and then three low-dimensional data are obtained respectively and then input into the machine learning algorithm for classification. The machine learning algorithms used in this paper are decision tree model [12], KNN [13], logistic regression [14] and Naive Bayes [15].

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

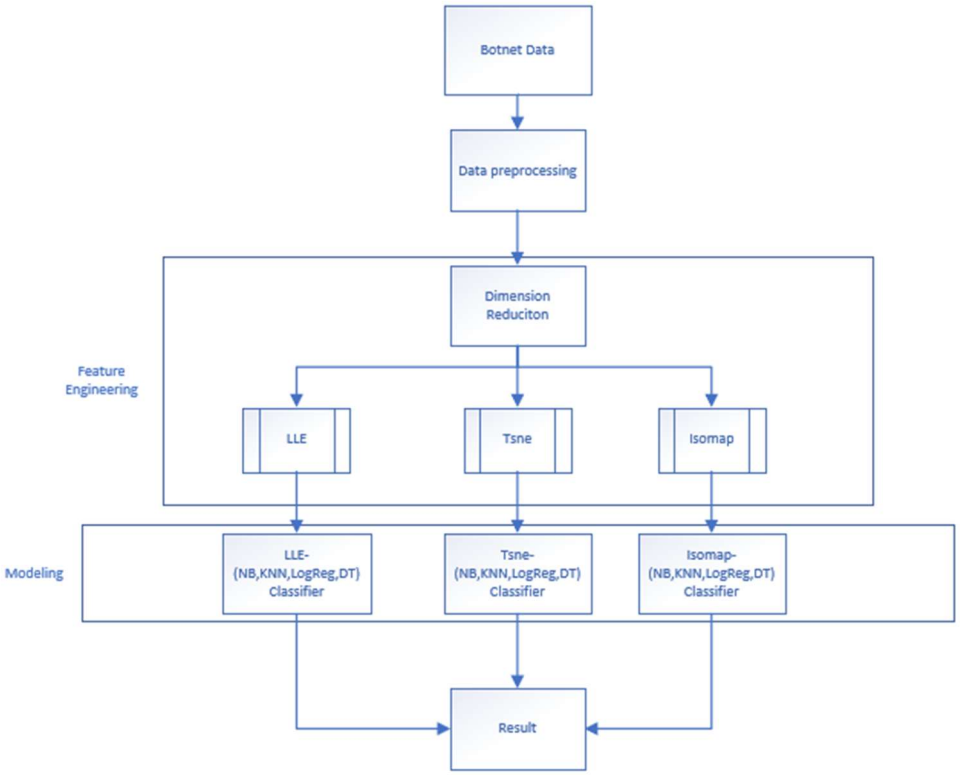


**Figure 3.** Botnet detection model design diagram

## 4. Data Set

In the process of combining machine learning and information security fields, it is very challenging to find an appropriate data set. The data we used in this experiment is named CTU-13 [16], which is a botnet data set captured by CTU University of Czech Republic in 2011. This data set contains a large amount of normal traffic, botnet traffic. The CTU-13 dataset is called CTU-13 because it contains botnet traffic sets for 13 different scenarios. Ctu-13-9, one of the scenarios with the largest amount of zombie traffic in CTU-13, was used in this experiment. Ctu-13-9 contains 383,215 zombie traffic and 362,594 normal traffic, each of which has the following characteristics: Start time, end time, duration, protocol name, source IP address, source port, destination IP address, destination port, number of bytes, number of packets, direction, service type, label, and status. Protocol, Status, and Direction are discrete features. In one-hot-encoding of data preprocessing, the above three characteristics are increased to 12, 166, 6. Therefore, in the final data set, the dimension of each network traffic is as high as 192 after eliminating useless features such as source address, IP address and port number. It is very inefficient to manually filter features in the face of such high dimensional data, so dimension reduction by manifold learning will achieve good performance.

## 5. Contrast Experiment

### 5.1 Visual Comparison of Dimension Reduction Effects

After data preprocessing, the data set was subjected to three different dimensionality reduction processes, the effect of which is shown in Figure 4. Because too large data will reduce the visualization effect, the author uses 400 traffic data after randomly shuffling and adjusting the balance as a visualization demonstration of dimensionality reduction, where [0] represents normal traffic and [1] represents zombie traffic.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

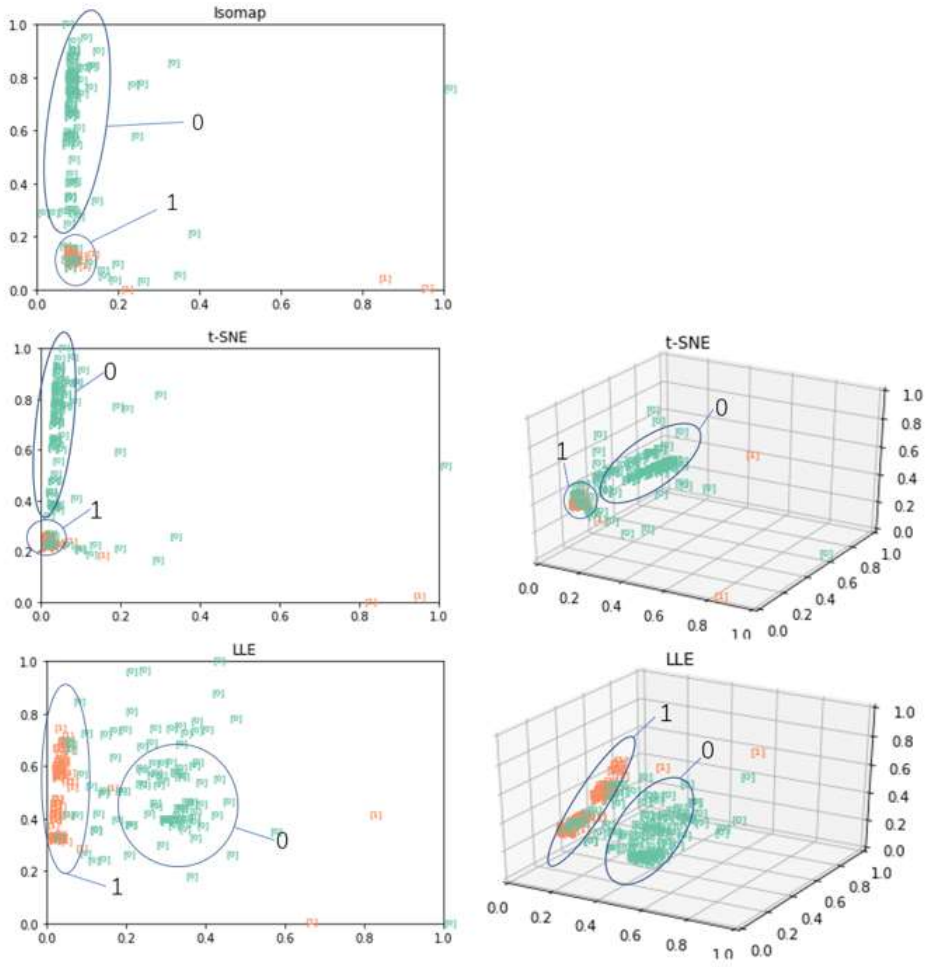


**Figure 4.** Visual effect of manifold learning algorithm for dimensionality reduction

It can be clearly observed from Figure 4 that the performance of LLE algorithm is significantly stronger than that of T-SNE and ISOMAP. There is little difference between ISOMAP and T-SNE in two-dimensional vision, but t-SNE is significantly higher than LLE and ISOMAP in computational complexity and cost.

## 5.2 Measure Standard

Confusion matrix, as the most commonly used method to detect the performance of classifiers, uses three indicators as standards in this paper:

Precision refers to the percentage of correctly predicted events out of the total number of predicted events:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

Recall refers to the percentage of the actual total number of events that were correctly predicted:

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

Accuracy refers to the percentage of correctly predicted results compared to the total number of actual events:

$$Precision = \frac{TP}{TP + FP} \tag{15}$$

True Positive (TP) indicates that the prediction is Positive and the actual is Positive. TN (True Negative) indicates that the predicted value is Negative and the actual value is Negative. FP(False Positive) indicates that the predicted value is Positive but the actual value is negative. FN (False Negative) indicates that the predicted value is Negative but the actual value is positive.

## 6. Experiment Result

The overall experimental results are shown in figure 5,6,7,8 and table 1. In combination with FIG. 5, FIG. 6 and FIG. 7, it can be seen that the performance of naive Bayes classifier is significantly inferior to the other three classifier algorithms in the three dimensionality reduction methods. The reason is that although naive Bayes is very efficient in operation, its core idea is to assume that all variables are independent of each other. The other three classifier models performed well under three different dimensionality reduction methods, among which logistic regression showed the best overall performance. In terms of dimension reduction algorithm, ISOMAP algorithm is slightly inferior to TSNE and LLE in overall effect, and LLE is the best overall. However, TSNE is significantly slower than ISOMAP and LLE in terms of computing speed. It can be seen from Table 1 that in terms of accuracy, t-SNE dimension reduction combined with logistic regression algorithm can reach 96.18%, which is the highest in this experiment. In Figure 8, the classifier with the best performance among the three dimensionality reduction algorithms is listed separately. It can be seen that there is no significant difference between LLE and TSNE algorithms combined with logistic regression classifier, and both algorithms are significantly stronger than ISOMAP algorithm.

**Table 1.** Experimental results

| DM- classifier | ACC | RECALL | PREC |
|---|---|---|---|
| ISOMAP-NB | 0.7789 | 0.96 | 0.76 |
| ISOMAP-DT | 0.9068 | 0.94 | 0.9 |
| ISOMAP-KNN | 0.8376 | 0.92 | 0.81 |
| ISOMAP-LR | 0.8944 | 0.91 | 0.87 |
| LLE-NB | 0.6468 | 0.98 | 0.62 |
| LLE-DT | 0.8896 | 0.97 | 0.82 |
| LLE-KNN | 0.9344 | 0.95 | 0.92 |
| LLE-LR | 0.9618 | 0.96 | 0.92 |
| TSNE-NB | 0.6769 | 0.95 | 0.67 |
| TSNE-DT | 0.8316 | 0.91 | 0.82 |
| TSNE-KNN | 0.9274 | 0.97 | 0.92 |
| TSNE-LR | 0.9582 | 0.96 | 0.95 |

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

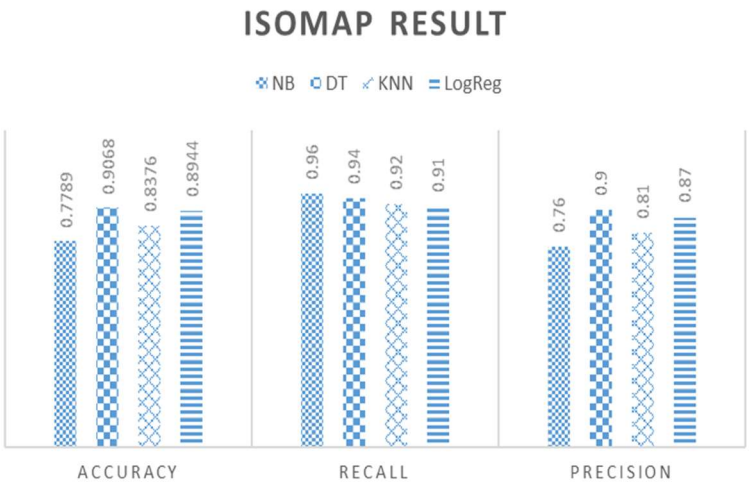


**Figure 5.** ISOMAP algorithm results

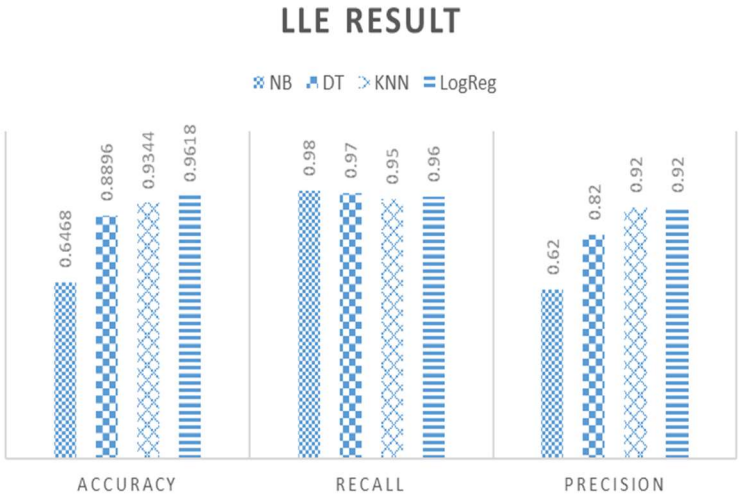


**Figure 6.** LLE algorithm results

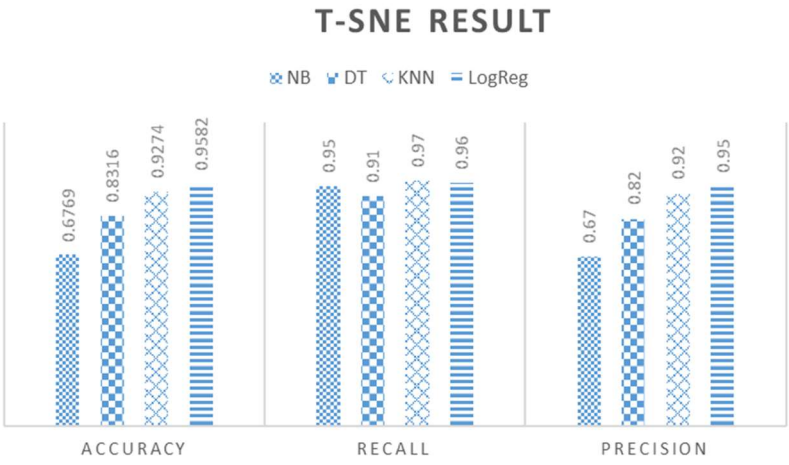


**Figure 7.** t-SNE algorithm results

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0071

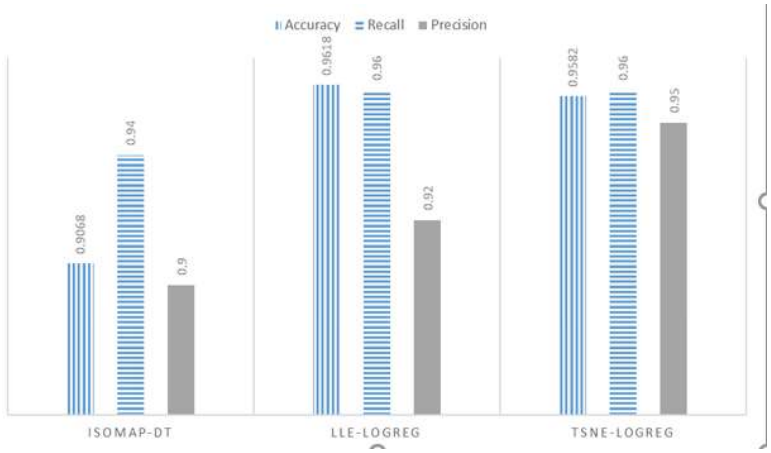


**Figure 8.** Comparison of the best classifiers

## References

[1] Miao Yuwei. Research on structural characteristics and robustness of botnet [J]. Science and Technology Information,2010(29):518-519.

[2] Fang Binxing, CUI Xiang, Wang Wei. Journal of computer research and development, 2011, 48(08): 1315-1331H. Zhang, C. Papadopoulos and D. Massey, "Detecting encrypted botnet traffic," 2013 Proceedings IEEE INFOCOM, Turin, 2013, pp. 3453-1358.

[3] H. Li, G. Hu, J. Yuan and H. Lai, "P2P Botnet Detection Based on Irregular Phased Similarity," 2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control, Harbin, 2012, pp. 79-82.

[4] O. Yavanoglu and M. Aydos, "A review on cyber security datasets for machine learning algorithms," 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, 2017, pp. 2186-2193.

[5] Guo Shangzan. Botnet Detection Based on Network Traffic Similarity Clustering [C]. China Communication Society, Liaoning Communication Administration. Proceedings of the 10th Annual Conference of China Communication Society. China Communication Society, Liaoning Communication Administration: Youth Working Committee of China Communication Society,2014:257-263.

[6] Xu Rong, Jiang Feng, Yao Hongxun. An overview of Manifold Learning [J]. Journal of Intelligent Systems,2006(01):44-51.

[7] Li PAN. BVM network intrusion detection method based on LLE feature extraction [D]. North China Electric Power University (Beijing),2011.

[8] Zhan Dechuan, Zhou Zhihua. Visualization of Manifold Learning Based on Integration [J]. Journal of Computer Research and Development,2005(09):1533-1537.

[9] Maaten, L. v. d.; Hinton, G., Visualizing data using t-SNE. J. Mach. Learn. Res. 2008, 9 (Nov), 2579-2605.

[10] Zhou H, Wang F, Tao P. t-Distributed Stochastic Neighbor Embedding Method with the Least Information Loss for Macromolecular Simulations. J Chem Theory Comput. 2018;14(11):5499–5510. doi:10.1021/acs.jctc.8b00652.

[11] Luan Lihua, Ji Genlin. Research on DT Classification Technology [J]. Computer Engineering, 2004 (09):94-96+105.

[12] Zhang Z. Introduction to machine learning: k-nearest neighbors. Ann Transl Med. 2016;4(11):218. doi: 10. 21037/atm.2016.03.37.

[13] Sperandei S. Understanding logistic regression analysis. Biochem Med (Zagreb). 2014;24(1):12–18. Published 2014 Feb 15. doi:10.11613/BM.2014.003.

[14] Xu, S. (2018). Bayesian Naïve Bayes classifiers to text classification. Journal of Information Science, 44 (1), 48–59.

[15] "An empirical comparison of botnet detection methods" Sebastian Garcia, Martin Grill, Jan Stiborek and Alejandro Zunino. Computers and Security Journal, Elsevier. 2014. Vol 45, pp 100-123. http://dx. doi.org/ 10.1016/j.cose.2014.05.011.