

Enhance density peak clustering algorithm for anomaly intrusion detection system

Salam Saad Alkafagi¹, Rafah M.Almuttairi²

^{1,2}College of Information Technology, University of Babylon, Babylon, 51002, Iraq

ABSTRACT

In this paper proposed new model of Density Peak Clustering algorithm to enhance clustering of intrusion attacks. The Anomaly Intrusion Detection System (AIDS) by using original density peak clustering algorithm shows the stable in result to be applied to data-mining module of the intrusion detection system. The proposed system depends on two objectives; the first objective is to analyzing the disadvantage of DPC; however, we propose a novel improvement of DPC algorithm by modifying the calculation of local density method based on cosine similarity instead of the cut off distance parameter to improve the operation of selecting the peak points. The second objective is using the Gaussian kernel measure as a distance metric instead of Euclidean distance to improve clustering of high-dimensional complex nonlinear inseparable network traffic data and reduce the noise. The experimentations evaluated with NSL-KDD dataset.

Keywords: Data Mining, Anomaly Intrusion Detection System, Density Peak Cluster algorithm

Corresponding Author:

Salam Saad Alkafagi

College of Information Technology

University of Babylon

Babylon, Iraq

salam.saad@student.uobabylon.edu.iq

1. Introduction

The techniques of Intrusion detection grouped into two major types are signature-based and anomaly-based intrusion detection. Signature-based detection work to discover threats based on manner obtained from known threats [1][2]. Anomaly-based detection identifies threats based on the most important perversion from the normal activates [3][4].

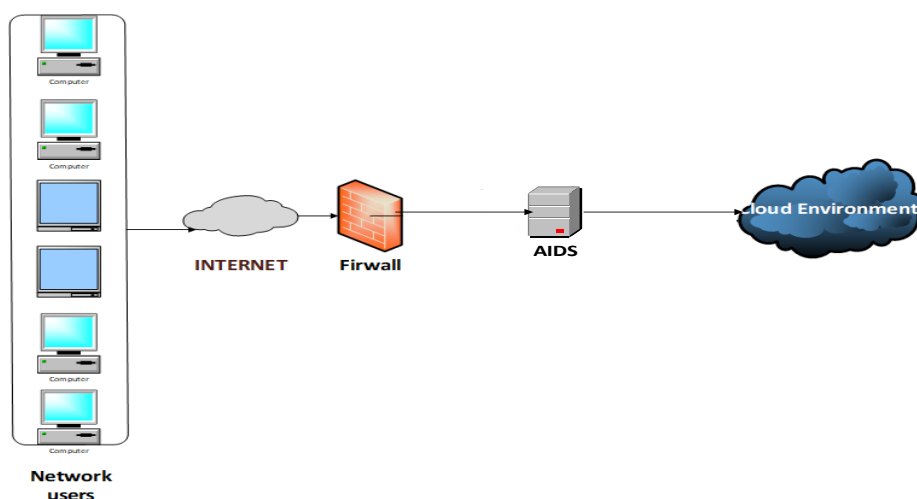


Figure 1. The Anomaly Intrusion Detection System (AIDS)

The Supervised Anomaly Intrusion Detection is work on training the normal activities of data from historical normal behavior patterns, which mostly use machine-learning methods. The problem with this approach is the

new activities are not discovered, therefore, is usually lacks capability for detecting new intrusion. To handle these limitations of supervised anomaly intrusion detection approaches by using unsupervised learning. Unsupervised anomaly detection approaches do not require labelled training data. The clustering algorithms are one of unsupervised learning has been a focus recently. Clustering methods are grouping the data points based on distance or local density to the nearest centroid of the cluster [5]. Clustering algorithms divide data set into groups of subsets (cluster). Each subset has the highest similarity within data in the same subset, and the data between any subset is most dissimilar to that of other subset. Clustering identifies natural structures in data and cluster appear in various shapes, sizes, sparseness, and degrees of separation. Cluster technique can classify to five types as Model-based, Grid-based, Hierarchical based, Partition based and Density-based techniques [6]. The Partitioning and hierarchical based models perform well in the case of finding spherically shaped clusters, but they has limitation on arbitrary clusters[7]. However, the Density-based model can use to handle this limitation; however, the methods work based on two major parts are dense area and boundary area[8]. In 2014, Rodriguez and Laio present novel Density-based clustering algorithm called density peaks clustering (DPC). The core idea of DPC is based on compute the local density and separation distance of data [9] [10].

1.1. Problem statement

Until this moment, the IDS still has many limitations such as below:

1. The real network environment has imbalanced network traffic, which means that threats records show less frequently occurring than normal records. The classification algorithms are biased towards the more frequently occurring records in the dataset. The imbalanced network traffic strongly reduce the detection performance of most traditional classification algorithms, and this case will impact the small attacker records by strongly decrease the accuracy of these small attacks (such as u2r and r2l attacks).
2. Data mining model must detect the non-spherical shape for network IDS classes. The conventional Data mining such as k-means is not able to detect these classes. Thus, this problem plays a major role to choose the right a proactive model to classify these attacks.
3. The core process of clustering of original DPC based on calculate the distance for all data samples using Euclidean distance metric, however the Euclidean distance is cause misclassifications when the dataset is complex and has high dimensional features.
4. In the real word network, the structure and operating environment are changing continually which show unknown attacks, which are, not appear in the training dataset. Moreover, for this reason most of supervised IDS algorithms usually perform poorly.

1.2. Contribution

The major contributions of this proposal are:

- We enhance the density peak clustering algorithm (DPC) by replacing the Euclidean distance calculation method with the Gaussian kernel function to project the data attributes into the high-dimensional kernel space for improving the process of cluster the complex nonlinear inseparable network traffic data.
- Modified the calculation method of local density based on the average of cosine similarity for all the data points to work as threshold instead of cutoff distance.

2. Enhanced density peak clustering (EDPC)

The Enhanced Density Peak Clustering (EDPC) is an improved algorithm based on original density peak clustering algorithm. It is one of the Density-based methods, which work to discover clusters based on high local density of data points [8], [9]. The DPC deal with continuous regions. The core idea of DPC to calculate cluster centers depend on measure both the local Density and the separation distance [12]. Those methods depend on computing the distance between all samples in the given dataset. The original DPC uses the Euclidean distance measure to calculate the distance of all samples in given dataset. The Euclidean distance is cause misclassifications when the dataset is complex and has high dimensional features. Therefore, we use the Gaussian kernel function to measure the kernel function between two data points; here the Gaussian kernel compute as equation below:

$$K(\vec{x}_i, \vec{x}_j) = \exp\left(-\frac{\|\vec{x}_i - \vec{x}_j\|^2}{2(\sigma^2)}\right) \text{ where } \sigma > 0 \quad (1)$$

Where σ is the scale parameter [13]. Then the kernel distance between two samples, \vec{x}_i and \vec{x}_j , is calculate as equation below:

$$d_{i,j} = \sqrt{2(1 - k(\vec{x}_i, \vec{x}_j))} \quad (2)$$

Then the algorithm starts to calculate the local density for each data point in the given dataset. The given dataset S that formed as real multi-dimensional vectors. Local density will be P_i for the data point $i \in S$, which is a number of points, can be a neighbor of i . We will use the exponential kernel method to calculate the local density of data point that equation is [14]:

$$P_i = \sum_{i,j \in S} \exp\left(-\frac{d_{i,j}^2}{cd^2}\right) \quad (3)$$

Where the Cd refers to the cutoff distance, which is the initial specified parameter that works as the threshold to control the weight degradation rate. The determination of Cd is the assignment of a neighbor of i in S dataset who have less distance than cutoff distance [6]. In [15-18] the authors suggest to determine the neighborhood radius Cd to include 1% to 2% of the data in the neighborhood and this will influence on the clustering results. In real application, there is a hard issue to determine the best value of CD parameter initially before starting with clustering process. In the figure (2) shows the distances from point 1 to other points except 2, 3, 5 and 6 are used to be less distance than cutoff distance. In addition, similarities between point 1 and 8, 9, 11, 12, 13, 14 are used to be less distance than the cutoff distance also.

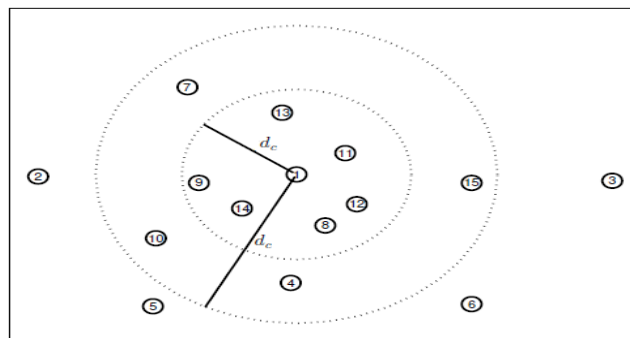


Figure 2. Cutoff distances from point to others

For that reason, we modified the calculation method of local density based on the average of cosine similarity for all the data points to work as threshold instead of cutoff distance. The modification in the equations below:

$$C_s = \frac{\sum_{i=1}^n X_i * X_j}{\sqrt{\sum_{i=1}^n (x_i)^2} * \sqrt{\sum_{i=1}^n (x_j)^2}} \quad (4)$$

Where C_s means the cosine similarity for two points.

$$A_s = \frac{1}{N} \sum_{i=1}^N C_s(i) \quad (5)$$

Where A_s is the average of all cosine similarity points, which will be the threshold to assign of neighbors of i in the dataset whose have less distance than A_s .

$$P_i = \sum_{i,j \in S} \exp\left(-\frac{d_{i,j}^2}{A_s}\right) \quad (6)$$

After calculating local density, we have to compute the separation distance by select minimum distance from x_i to x_j if the local density of x_i is bigger than local density of x_j . The other way by selecting maximum distance from x_i to any other point if that point has no local density. The first step to calculate separation distance is to represent the local density set index in descending order index set then calculate the equation below:

$$\varphi(x_i) = \begin{cases} \min(d_{ij}), & \text{if } i > j \\ \text{or} \\ \max(d_{ij}), \end{cases} \quad (7)$$

The peak points determined manually based on sorting from highest to lowest for both local density and high separation distance and select these highest points to be the clusters center, however this process done with help of decision graph by drawing the vector of local density in horizontal axes and the vector of separation distance in vertical axes.

$$\gamma_i = P_i * \delta_i \quad (8)$$

The result of γ_i will be sorted for all data points to select data points with higher γ_i be the density peaks. After select density peaks, they will be the center of each cluster that means the peak number equals to cluster number. The rest of the non-peak data point will be assign to nearest distance cluster center (peak point).

Algorithm1: Enhanced Density Peak Clustering

- 1: Compute distance $d_{i,j}$ between data point's x_i and x_j using Gaussian kernel distance
 - 2: compute cosine similarity C_s between data point's x_i and x_j then calculate average of cosine similarity for all the data point
 - 3: Compute the modified local density P_i of each data point x_i using exponential kernel method based average of cosine similarity as cutoff threshold.
 - 4: Compute the separation distance ϕ_i of each data point x_i .
 - 5: Compute γ_i , γ_i is production parameter.
 - 6: sort γ_i in descending order index set to select highest γ_i to be the peak (centroid)point
 - 7: finally, compute the remaining (nonpeak) points and assign each point of them to cluster as its nearest neighbour, which has nearest distance.
 - 8: return the groups of subsets which is each subset called cluster
-

3. Preparatory work

3.1. Dataset description

The NSL-KDD dataset is not the newest dataset, but is a new version that removes the disadvantage of the KDDCup 1999 dataset. This data set proposed in 2015 as modified KDD-CUP99 dataset. The advantage of this dataset is the training dataset doesn't has redundant records and the testing set doesn't has duplicate records, this will help the classifiers to escape frequent records and not be biased [19-21]. Until now, Most of research in the techniques of Intrusion Detection, the researcher use this NSL-KDD dataset as benchmark dataset[22]. In this paper, KDDTrain+_20Percent data sets that used as 70% training sets and 30% testing sets of our system. The KDDTrain+_20Percent dataset contains 25,192 instances that each record of this dataset represents a connection of network is composed by the 41-dimensional feature vector[11] [23]. The samples of dataset labelled as normal or threats . the threats are subdivided into four main classes: 'probe', 'dos', 'u2r', and 'r2l', a total of 22 types of attacks. The figure (3) below describes the KDDTrain+_20Percent data set attributes with class labels.

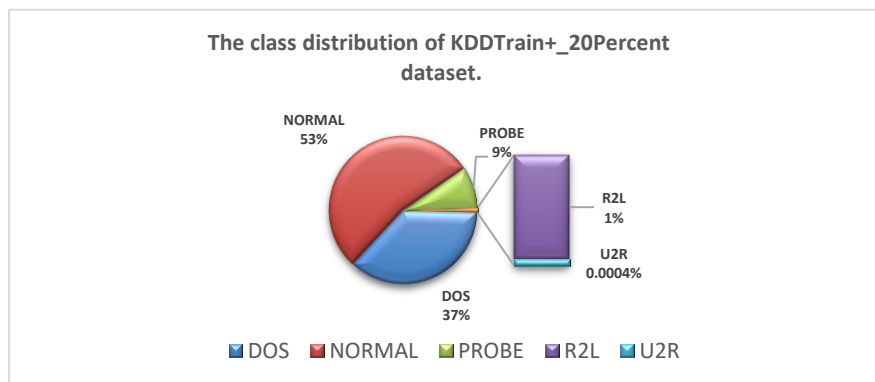


Figure 3. Class distribution of KDDTrain+_20percent

4. The proposed system

The proposed system has two main phases: preprocess and clustering phases. The figure (4) shows the working steps of our proposed system.

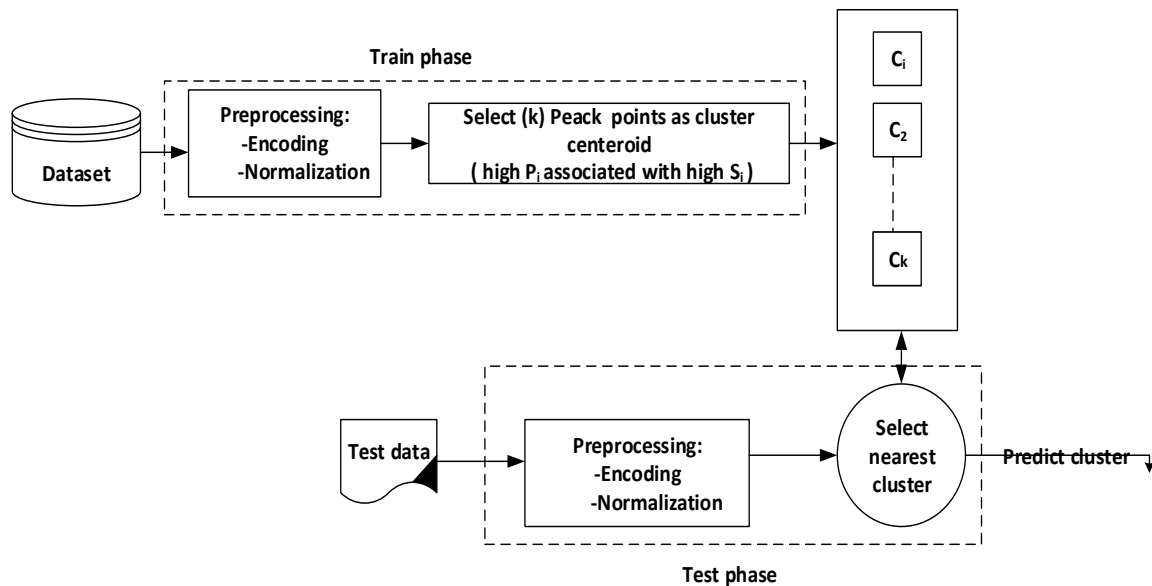


Figure 4. The proposed system

4.1. Pre-processing phases

The main operations in this phase are:

4.1.1. Encoding dataset (transformation)

In the most of intrusion detection data set, some non-numeric attributes. The non-numeric data not detectable by most of machine learning algorithms, however, the one-hot encoding method is the most well encoding schema to convert the non-numeric attributes to numeric data. In short, this method produces a vector with length equal to the number of categories in the data set [24-26]., there are three numeric data should be encode in the NSL-kdd dataset as described in figure below.

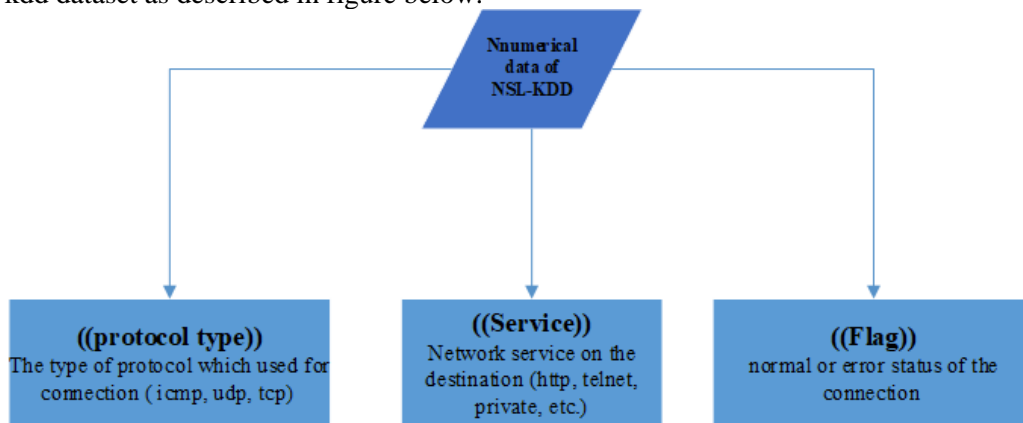


Figure 5. The numerical data of NSL-kdd dataset

4.1.2. Normalization

Most of mining algorithms involving clustering techniques produce a good accuracy result if data normalized in the preprocessing phase [27]. It standardizes the data by transforming within a given range. Min-max scaler

scales each of the features individually between (0 ,1). This is good scaling options when you want to preserve the zero's in a sparse dataset. Equation (9) calculates the normalization features (Min_Max):

$$Min_Max(xi') = \frac{xi - min(xi)}{max(xi) - min(xi)} \tag{9}$$

Where: xi is the current value of the variable in the data, $min(xi)$ is the smaller value in the given data record, $max(xi)$ is bigger value in the same given data record.

4.2. Clustering phase

In summary, the steps of EDP algorithm process is declare as follows:

1. The first step Calculate the kernel function based on scaler a parameter (σ) that specify initially, then calculate Gaussian function as the distance measure based on kernel functions to find the distance between any two points of the training data.
2. Compute the average of cosine similarity for all the data points to work as threshold instead of a cutoff distance parameter.
3. Compute the modified local density based on the average of cosine similarity then calculate the separation distance to find the peak points with the help of decision graph.
4. Calculate Y_i , Y_i is the production of highest local density and separation distance respectively to produce peak points. Then compute the remaining (nonpeak) points and assign each point of them to cluster as its nearest neighbor.

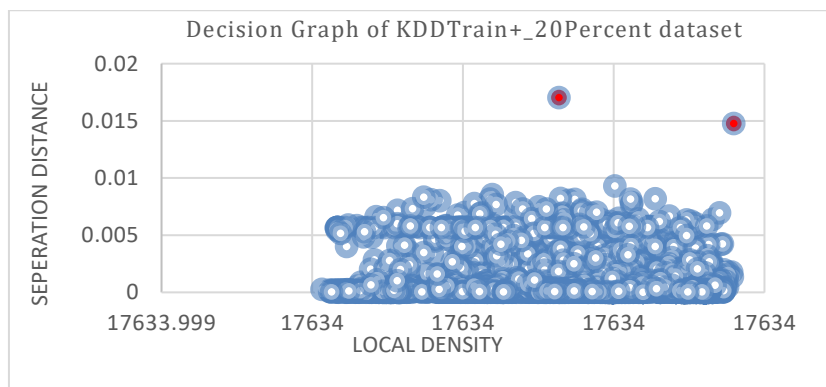


Figure 6. Decision Graph of NSL-kddTrain+_20Percent dataset

5. Experimentation

The model in this paper build in pycharm-2019.3.5(community edition), open JDK 64-bit server VM. The programing language is python on 2.20 GHz Core i7 CPU, RAM16 GB.

5.1. Metrics of evaluation

5.1.1. Silhouette index

It's the main criteria used to evaluate the performance of cluster analysis. In a very simple way, it used to show if the object is in the right cluster or if it takes place in another cluster [28]. Its worth to mention that silhouette index has three different values: -1 show value that object in the wrong cluster. 0 show the object in another cluster. +1 show objects in the right cluster Silhouette Index is calculated in the below equation as [29-33]:

$$Silhouette\ index = \frac{x(i) - y(i)}{max\{x(i), y(i)\}} \tag{10}$$

$x(i)$ is a lower distance among cluster and object and $y(i)$ is the higher distance between object and clusters.

5.1.2. Cohesion measurement

Cohesion measurement is a new measurement used to evaluate cluster analysis. The goals of the cohesion measurement are to determine the distance between cluster groups. In another word, it measures the distance between objects within the same cluster, the high value of cohesion measurement refers to the cluster high quality. It simply applied as a measurement metric of diversity of a cluster as shown in below formula [34].

$$Cohesion = \sum_{i=1}^K m_i d(c_i, c)^2 \tag{11}$$

When c_i is mean of i th cluster, c is total clusters, m_i is the total samples in the cluster c_i , k is number of all clusters.

5.1.3. Rand index

Rand Index It simply applied as a measurement metric used to evaluate cluster analysis and used for cluster validation. The Rand Index metric which calculated using the formula below [35]:

$$RI = \frac{a + d}{a + d + c + b} \tag{12}$$

Where:

- a - objects in a pair that put up in the same subset in cluster (1) and in the same subset in cluster (2).
- b - Objects in a pair that put up in the same subset in cluster (1) and in different subsets in cluster (2).
- c - Objects in a pair that put up in the same subset in cluster (2) and in different subsets in cluster (1).
- d - Objects in a pair that put up in different subsets in cluster (1) and in different subsets in cluster (2).

5.2. Discussion and evaluation of results

The DPC is a very good clustering method, but it has some limitations in the structure are: the first issue is regarding to computing the local density of all data points depending on cutoff distance (Cd) parameter which is working to determine the neighborhood radius. The Cd parameter influence of this percentage of the clustering results.

We investigate Cd with 2% and with less than 2% of the data in the neighborhood, in case of original DPC the Cd parameter has an influence on accuracy results but not sensitive to peak numbers. With an assist to choose the peak points, plot all data samples based on both high local density and high separation distance to select as the peak points, which called a cluster center. However, the results show only one peak point and this case will lead to a dramatic bad clustering quality. This declare by the table (1) DPC results of peak numbers.

Table 1. DPC results of peak numbers

method	Train set	Distance metric	Cutoff distance	No. of peaks
Original DPC	17634	Euclidean	%2	Only one peak
Original DPC	17634	Euclidean	%1	Only one peak
Original DPC	17634	Euclidean	%1.5	Only one peak

The figure below describe plotting samples of data with both high local density and high separation distance using decision graph(DG) and sorting graph(SG) for the original DPC algorithm.

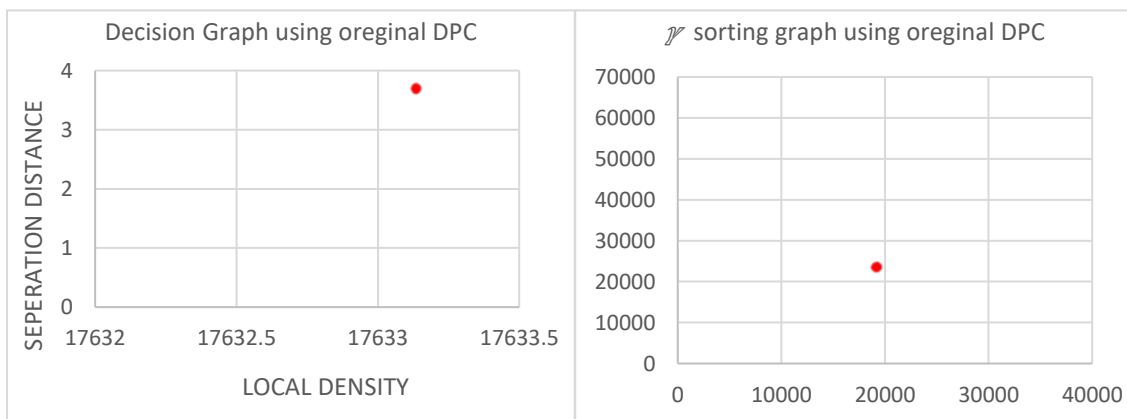


Figure 7. the DG and SG for original DPC using NSL-kdd dataset

However, we modified the local density function based on the average of cosine similarity to be a dynamic value parameter to determine the neighborhood radius instead of a static cutoff distance parameter. This modification gives excellent enhancement of the quality of clustering. The experiment shows that EDPC algorithm based on the modified local density function gives two peak points. The figure (8) shows the EDPC decision graphs.

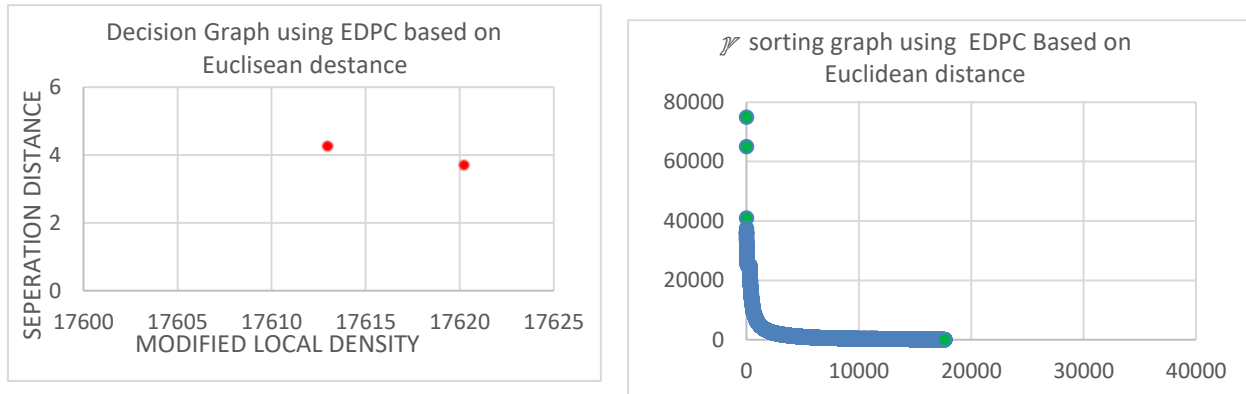


Figure 8. EDPC decision graph and sorting graph based on Euclidean distance

The environment of training data clusters shows grouping two subsets. The figure (9) describes the size and class size of each subset.

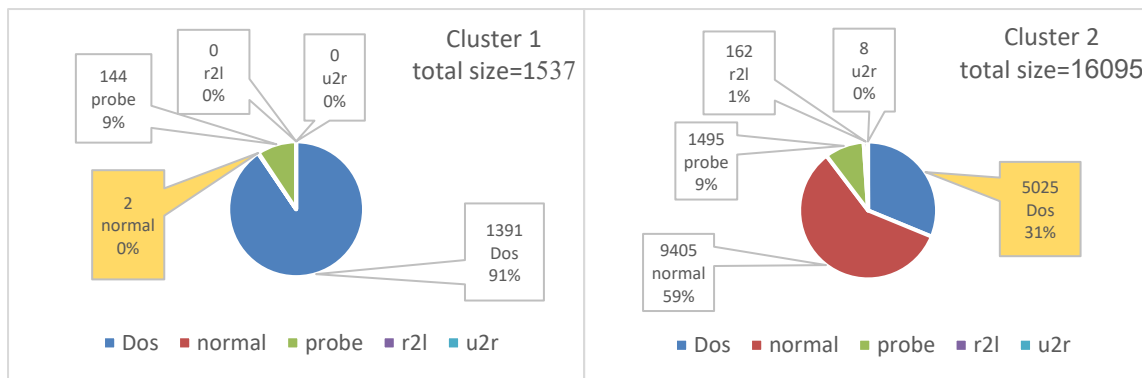


Figure 9. description of the size and class size of each subset (Euclidean based)

The second issue regarding with data dimensionality, DPC shows good performance only when the dimensionality of data is very low in this reason we used the Gaussian function instead of Euclidean function as the distance measure to find the distance between any two points of the train dataset to solve this limitation. The Gaussian function gives improved with the clustering result by reducing the imbalance of network data and enhance the accuracy of small attacker's classes. The experiment shows that EDPC algorithm based on modified local density function and Gaussian kernel distance in Figure (10).

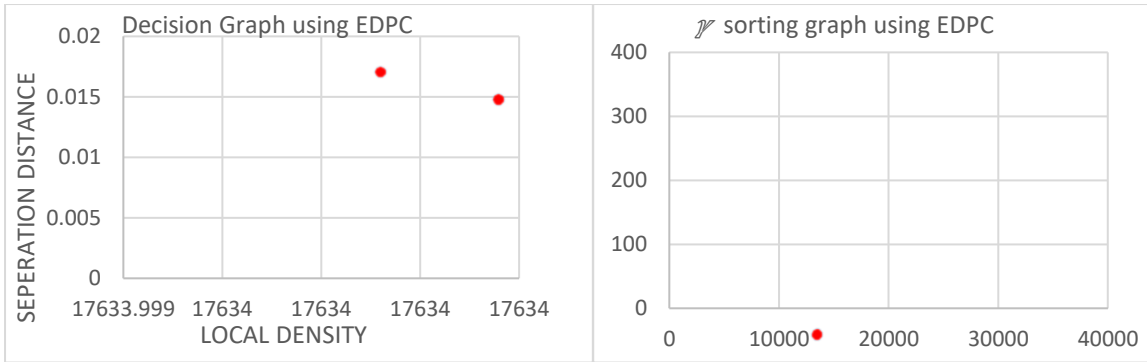


Figure 10. EDPC decision graph and sorting graph based on Gaussian distance

The description of the size and class size of each training subset shows that minority classes like R2l and U2r attacks grouped with one subset, and divide the high occurs classes in two clusters (normal, dos, probe). The Figure (11) shows the size and class size of each subset.

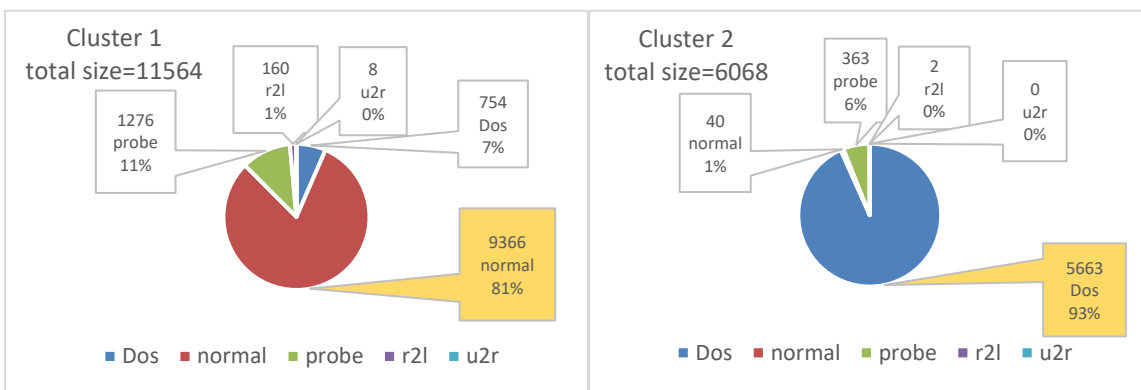


Figure 11. Description of the size and class size of each subset (Gaussian based)

Our model based on novel enhance density peak clustering based on modified local density and Gaussian kernel show excellent evaluation on Rand Index metric with 0.86, Cohesion 92005 and Silhouette Index 0.39 compared to an original DPC algorithm.

Table 2. Comparison of result

Metho d	Cluster No.	Cd	Pi	Distance measure	Evaluation metrics		
					Silhouette Index	Cohesion	Rand Index
DPC	1	%2	original	Euclidean	0.25	89985	0.56
EDPC	2	A_s	modified	Euclidean	0.13	82124	0.61
EDPC	2	A_s	modified	Gaussian	0.39	92005	0.86

6. Conclusion

Anomaly-based Intrusion detection system is a significantly important technique for the security of the network, But the problem with this approach is the new activities that not discoverer, therefore, is usually lakes capability for detecting new intrusion. To handle these limitations of supervised anomaly intrusion detection approaches by using unsupervised learning. Unsupervised anomaly detection approaches do not require labelled training data. The clustering algorithms are one of unsupervised learning has been a focus recently. In this paper, we proposed a new model with novel modification of the clustering algorithm called EDPCA. This model shows a high-performance in intrusion detection. The result of our Experimental demonstrates that the model is operative and give higher accuracy than the existing general clustering algorithm.

References

- [1] Z. Chiba, N. Abghour, K. Moussaid, A. El, and M. Rida, "New Anomaly Network Intrusion Detection System in Cloud Environment Based on Optimized Back Propagation Neural Network Using Improved Genetic Algorithm," vol. 11, no. 1, pp. 61–84, 2019.
- [2] Y. Y. Aung and M. M. Min, "An analysis of K-means algorithm based network intrusion detection system," *Adv. Sci. Technol. Eng. Syst.*, vol. 3, no. 1, pp. 496–501, 2018.
- [3] N. Shahadat, I. Hossain, A. Rohman, and N. Matin, "Experimental analysis of data mining application for intrusion detection with feature reduction," *ECCE 2017 - Int. Conf. Electr. Comput. Commun. Eng.*, pp. 209–216, 2017.
- [4] C. Series, "PAPER • OPEN ACCESS A Proactive Model for Optimizing Swarm Search Algorithms for Intrusion Detection System, Salam Saad Alkafagi and Rafah M.Almuttairi University of Babylon, Iraq".
- [5] X. Ni, D. H. B, S. Chan, and F. Ahmad, "Using Unsupervised Feature Selection and Density Peak Clustering," vol. 1, pp. 212–227.
- [6] M. Parmar, D. Wang, A. Tan, C. Miao, J. Jiang, and Y. Zhou, "A Novel Density Peak Clustering Algorithm based on Squared Residual Error."
- [7] C. Meng, Y. Lv, L. You, and Y. Yue, "Intrusion Detection Method Based on Improved K-Means Algorithm," *J. Phys. Conf. Ser.*, vol. 1302, no. 3, 2019.
- [8] Z. Jiang, X. Liu, and M. Sun, "A Density Peak Clustering Algorithm Based on the K-Nearest Shannon Entropy and Tissue-Like P System," vol. 2019, 2019.
- [9] L. Carlos, C. Heredia, and A. Rodrigo, "discharge sources," vol. 107, no. November 2018, pp. 224–230, 2019.
- [10] Y. Shi and H. Shen, "Anomaly Detection for Network Flow Using Immune Network and Density Peak," vol. 2019, pp. 1–10, 2019.
- [11] S. M. Sangve and R. C. Thool, "ANIDS : Anomaly Network Intrusion Detection System Using Hierarchical Clustering Technique," pp. 121–129.
- [12] J. Lin, "Accelerating Density Peak Clustering Algorithm," pp. 1–18, 2019.
- [13] Z. Zhang, X. Liu, and L. Wang, "Spectral Clustering Algorithm Based on Improved Gaussian Kernel Function and Beetle Antennae Search with Damping Factor," vol. 2020, 2020.
- [14] N. Science, C. Phenomena, L. Li, H. Zhang, H. Peng, and Y. Yang, "Chaos , Solitons and Fractals Nearest neighbors based density peaks approach to intrusion detection," vol. 110, pp. 33–40, 2018.
- [15] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks Alex Rodriguez and Alessandro Laio," vol. 1492, 2014.
- [16] Y. S. Mezaal, L. N. Yousif, Z. J. Abdulkareem, H. A. Hussein, S. K. Khaleel, "Review about effects of IOT and Nano-technology techniques in the development of IONT in wireless systems," *International Journal of Engineering and Technology (UAE)*, vol. 7, no. 4, 2018.
- [17] A. Sallomi, Seham Ahmed Hashem, Y. S.h Mezaal, "A novel theoretical model for cellular base station radiation prediction," *International Journal of Simulation: Systems, Science and Technology*, Vol.19, No.6, 2018.
- [18] I. A. Aljazaery, H. Alhasan, F. N. Al Hachami, and H. T. S. Alrikabi, "Simulation Study to Calculate the Vibration Energy of Two Molecules of Hydrogen Chloride and Carbon Oxide," *Journal of Green Engineering*, vol. 10, no. 9, pp. 5989-6010, 2020.
- [19] S. Revathi and A. Malathi, "A Detailed Analysis on NSL-KDD Dataset Using Various Machine Learning Techniques for Intrusion Detection," vol. 2, no. 12, pp. 1848–1853, 2013.
- [20] A. Ghazi, S. Aljunid, A. Fareed, S. Z. S. Idrus, C. M. Rashidi, A. Al-dawoodi, and A. M. Fakhrudeen, "Performance Analysis of ZCC-Optical-CDMA over SMF for Fiber-To-The-Home Access Network," in *Journal of Physics: Conference Series*, 2020, vol. 1529, no. 2, p. 022013: IOP Publishing.
- [21] T. Masunda, A. Amphawan, S. Alshwani, and A. Aldawoodi, "Modal properties of a varied high indexed large core 4-mode photonic crystal fiber," in *2018 IEEE 7th International Conference on Photonics (ICP)*, 2018, pp. 1-3: IEEE.
- [22] P. Aggarwal and S. K. Sharma, "Analysis of KDD Dataset Attributes - Class wise for Intrusion Detection," *Procedia Comput. Sci.*, vol. 57, pp. 842–851, 2015.
- [23] S. Aljawarneh, M. Aldwairi, and M. Bani, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *J. Comput. Sci.*, vol. 25, pp. 152–160, 2018.

- [24] K. Potdar, "A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers," vol. 175, no. 4, pp. 7–9, 2017.
- [25] A. Ghazi, S. Aljunid, S. Z. S. Idrus, C. Rashidi, A. Al-dawoodi, B. A. Mahmood, A. Fareed, M. U. Zaenal, N. H. Qasim, and R. M. Rafeeq, "A Systematic review of Multi-Mode Fiber based on Dimensional Code in Optical-CDMA," in *Journal of Physics: Conference Series*, 2021, vol. 1860, no. 1, p. 012016: IOP Publishing.
- [26] M. Al-dabag, H. ALRikabi, and R. Al-Nima, "Anticipating Atrial Fibrillation Signal Using Efficient Algorithm," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 17, no. 2, pp. 106-120, 2021.
- [27] C. Science and S. Publications, "Data Mining : A Preprocessing Engine Luai Al Shalabi , Zyad Shaaban and Basel Kasasbeh Applied Science University , Amman , Jordan," vol. 2, no. 9, pp. 735–739, 2006.
- [28] L. Dalton et al., "Clustering Algorithms : On Learning , Validation , Performance , and Applications to Genomics," pp. 430–445, 2009.
- [29] T. Thinsungnoen, N. Kaoungku, and P. Durongdumronchai, "The Clustering Validity with Silhouette and Sum of Squared Errors," no. January 2015, 2016.
- [30] H. A. Naman, N. A. Hussein, M. L. Al-dabag, H. Salim, "Encryption System for Hiding Information Based on Internet of Things," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 15, no. 2, 2021
- [31] S. Miqdad Mohamed, H. Al Rikabi, and S. M. Ali, "Finding the discriminative frequencies of motor electroencephalography signal using genetic algorithm," *Telkomnika*, vol. 19, no. 1, pp. 285-292, 2021.
- [32] A. Fareed, A. Ghazi, A. Al-Dawoodi, S. Aljunid, S. Z. S. Idrus, C. Rashidi, A. Amphawan, A. M. Fakhrudeen, and I. E. I. Fadhel, "Comparison of Laguerre-Gaussian, Hermite-Gaussian and linearly polarized modes in SDM over FMF with electrical nonlinear equalizer," in *AIP Conference Proceedings*, 2020, vol. 2203, no. 1, p. 020045: AIP Publishing LLC.
- [33] B. K. Mohammed, M. B. Mortatha, A. S. Abdalrada, H. T. Salim, "A comprehensive system for detection of flammable and toxic gases using IoT," *Periodicals of Engineering and Natural Sciences (PEN)*, vol. 9, no. 2, pp. 702-711, 2021.
- [34] F. Berzal, "Evaluation Metrics for Unsupervised Learning Algorithms," arXiv preprint arXiv:1905.05667 ,2019.
- [35] J. M. Santos and M. Embrechts, "On the Use of the Adjusted Rand Index as a Metric for Evaluating Supervised Classification On the Use of the Adjusted Rand Index as a Metric for Evaluating Supervised Classification," no. September, 2009.