Jurnal Mantik, 6 (3) (2022), ISSN 2685-4236 (Online)

Published by:Institute of Computer Science (IOCS)

Jurnal Mantik

Journal homepage: www.iocscience.org/ejournal/index.php/mantik

Implementation of K-Means Clustering Method for Network Traffic Anomaly Detection

Haeni Budiati[1*], Antonius Bima Murti Wijaya[2], Barita Suci Vernando Zebua[3], Jatmika[4], Yo'el Pieter Sumihar[5]

[1,2*,3,4,5]Universitas Kristen Immanuel, Indonesia

A R T I C L E I N F O

Article history:

Received Oct 6, 2022
Revised Oct 25, 2022
Accepted Nov 15, 2022

Keywords:

Anomaly
K-Means Clustering
Prediction

ABSTRACT

Anomalies may degrade network performance for specific network traffic. Because of its nature, it causes abnormal network traffic. Using the K-means clustering method, this study addresses the formulation of the problem of detecting network bandwidth usage anomalies. The objective of this study is to identify potential network traffic anomalies. This study uses the K-Means Method to predict the value of the network traffic anomalies that will appear. K-Means operates by repeatedly iterating based on the initial cluster entered, until the same cluster results are discovered. The results of the study indicate that predicting the occurrence of anomalies with K-Means will help suppress activities that impede network traffic.

Corresponding Author:

Haeni Budiati,
Universitas Kristen Immanuel,
Ukrim No.KM. 11, Kadirojo I Street, Purwomartani, Kalasan, Sleman District, Special Region of Yogyakarta, 55571, Indonesia
Email: heni@ukrimuniversity.ac.id

1. INTRODUCTION

The evolution of internet technology continues to advance. The Internet network is now one of the basic human requirements for personal, professional, and commercial activities. Diverse requirements and network development are accelerating rapidly. Obtaining information that is rapidly and widely disseminated is one of the gaps that can be exploited by some individuals with the sophistication of Internet technology, whose use of the Internet is becoming increasingly widespread across multiple social strata. Alongside the development of an internet-connected network, network security and management are essential components of a system (Ananto et al., 2017)(Chakraborty, 2013). The emergence of anomalies in an internet traffic network disadvantages certain internet users. Internet service users can see the anomaly with the naked eye. Moreover, further investigation reveals that many people are unaware of this fact. The majority of internet users are unaware of these threats (Chandel, 2017)(Theodoridis & Koutroumbas, 2006).

In this instance, network traffic analysis was used to find these actions and activities. It is challenging and time-consuming to distinguish between normal and abnormal network traffic activity. To determine the order of anomalies (odd) on network connections, network analysts must examine all large, small, and wide data (Fink et al., 2002), not just the

possibility for network analysts, namely using Paramiko with the K-means Clustering method (Aini et al., 2018).

Paramiko is a Python library for Network Automation and Remote Secure Shell (SSH) that supports multiple network device vendors (Gopi, 2007). Numerous devices, including Cisco, Mikrotik, Aruba, Juniper, Palo Alto, and many others, are already supported by Paramiko. The results of this study can be implemented as a network traffic anomaly detection system utilizing paramiko and the K-Means Clustering method, as well as provide network administrators with decision-making aids for various network system disruptions.

2. RESEARCH METHOD

At this stage, the researcher describes the K-means Clustering performance design research flowchart in flowchart form. Utilizing a system flowchart, the researcher creates the flowchart. The following is a description of the flowchart depicted in Figure 1:

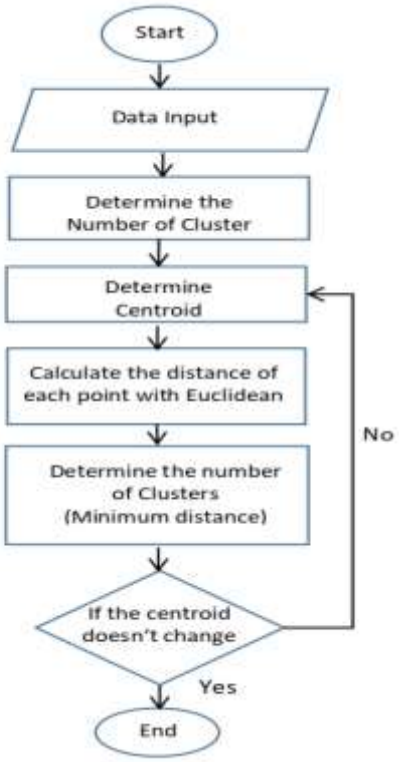


Figure 1. Research Method Flowchart

According to Figure 1, the first stage consists of data input in the form of client bandwidth data. Then, remove unneeded data from client bandwidth data.

a. Determine the number of clusters randomly determined by the user (Wulandari, 2014).
b. Calculate the distance of each point using Euclidean.
c. Determine the minimum distance after calculating the distance at each centroid point.
d. The system will then determine if the centroid is no longer shifting.

The K-means method is a data analysis or data mining technique that utilizes a partitioning system to perform data grouping. The K-Means method attempts to group the existing data into multiple groups, where the data in one group share similar

characteristics while the data in other groups have different characteristics. The K-Means method attempts to minimize differences between data within a cluster and maximize differences between clusters (Harjono & Wicaksono, 2013).

In this study, K-Means is employed because it gathers a large amount of testing data and produces cluster data after each iteration. The centroid and the number of clusters that were used to determine the data must first be determined(Lubis, 2016). Each iteration is used to help with the next iteration's process for other data clusters. The predicted occurrence of network traffic is then determined using the cluster data from each iteration.

The following is a list of the steps that need to be taken in order to implement k-means clustering(Hermanto, n.d.) in the system:

a. Select a keyword as the initial centroid.
b. In this system, the centroid or center point in this system is a keyword with two attributes: upload and download.
c. Calculate the distance of the input data keyword with the centroid
d. In this system, the distance between the input keyword and the centroid, namely upload and download, is used to calculate the input keyword. The result of the shortest distance concludes that the closest distance belongs to the selected keyword(Putra et al., 2015).
e. Update the value of the centroid point
f. In this system, the centroid point can be updated so that if a new major is added to the agency, the centroid can be updated.
g. Repeat the calculation of the centroid of each group

In this system, the calculation of the centroid point can be repeated, if the centroid has additional data(Ridho & Kusuma, 2018).
Calculation of K-means clustering in this system will yield results when calculating the distance between keywords inputted with the centroid and determining the cluster based on the shortest distance(Zulfadhilah et al., 2016).

The topology that was utilized in the research, which is depicted in Figure 2 below(Rosa, 2018):

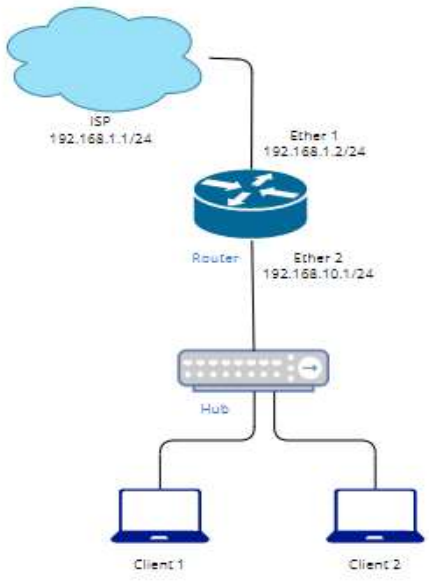


Figure 2. Network Topology

Case studies will be conducted on the local network topology design. As a test of data retrieval, the topology design employs a single ISP as the Internet source, a Mikrotik and a Switch, and two clients.

3. RESULTS AND DISCUSSIONS

The data to be tested is tx and rx on each client, The data to be tested is 128.912.

3.1 Client Detection Test

Using the paramiko method, data were collected on each client at this stage. By invoking Ether 2 on the primary router with IP 192.168.10.0/24 for 12 hours (43200s). Export data from the terminal to.csv because the client displayed on the terminal cannot be executed immediately after data retrieval.

3.2 Test result

Based on the initial test results, the distance of each data centroid has been determined, with only the top 5 and bottom 5 centroids displayed.

	tx	rx	Distance from Centroid 0	Distance from Centroid 1	Distance from Centroid 2
0	11.6	21.0	208.805077	231.880055	274.026933
1	8.7	8.8	211.301514	242.986275	281.130806
2	15.8	167.2	258.925240	152.572212	258.524815
3	15.8	176.0	264.427003	151.274717	260.444313
4	8.3	8.3	211.700213	243.626312	281.690220
...	...	...	...	...	...
128906	3.3	9.0	216.702307	246.366171	286.063787
128907	11.6	18.9	208.684858	233.451001	274.754381
128908	8.3	9.9	211.708526	242.408952	281.088064
128909	3.3	11.1	216.722172	244.793178	285.292306
128910	11.6	21.0	208.805077	231.880055	274.026933

[128911 rows x 5 columns]

[[11.517444076993327, 18.06976921508303], [34.65713354624259, 441.98452062829955], [814.4803594351732, 86.27150192554556]]

Figure 3. Test Results Calculating Centroid Distance

a. Centroid Test Result

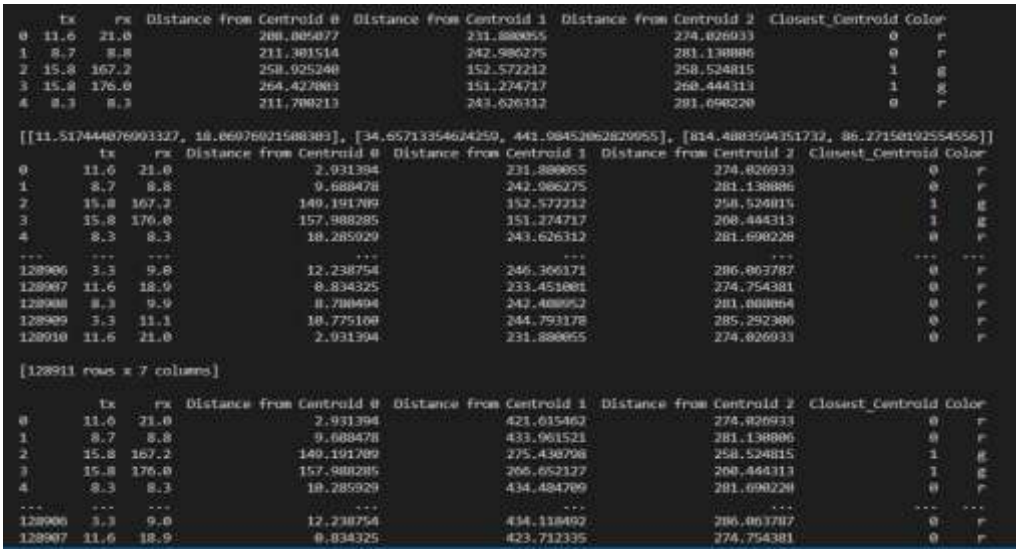
In addition, the results of the tests conducted to examine changes in each centroid point show that there is still a change in the centroid point in each data set, even though the data displayed only includes the top 5 and bottom 5.

	tx	rx	Distance from Centroid 0	Distance from Centroid 1	Distance from Centroid 2	Closest_Centroid	Color
0	11.6	21.0	2.931394	231.880055	274.026933	0	r
1	8.7	8.8	9.688478	242.986275	281.130806	0	r
2	15.8	167.2	149.191709	152.572212	258.524815	1	g
3	15.8	176.0	157.988285	151.274717	260.444313	1	g
4	8.3	8.3	10.285929	243.626312	281.690220	0	r
...	...	...	...	...	...	...	...
128906	3.3	9.0	12.238754	246.366171	286.063787	0	r
128907	11.6	18.9	0.834325	233.451001	274.754381	0	r
128908	8.3	9.9	8.780494	242.408952	281.088064	0	r
128909	3.3	11.1	10.775160	244.793178	285.292306	0	r
128910	11.6	21.0	2.931394	231.880055	274.026933	0	r

Figure 4. Centoroid Check Test Results

b. Centroid Check loop test

At this point, after iterative checks on each data, there is no change in the centroid point of each data, where only the top 5 and bottom 5 data are displayed; therefore, the K-Means method test was successfully applied.



	tx	rx	Distance from Centroid 0	Distance from Centroid 1	Distance from Centroid 2	Closest_Centroid	Color
0	11.6	21.0	200.805077	231.880055	274.026933	0	r
1	8.7	8.8	211.301514	242.986275	281.130806	0	r
2	15.8	167.2	258.925240	152.572212	258.524815	1	g
3	15.8	176.0	264.427003	151.274717	260.444313	1	g
4	8.3	8.3	211.700213	243.626312	281.690220	0	r

[[11.517444076993327, 10.869760215883803], [34.65713354624259, 441.98452062829655], [814.4883594351732, 86.27150192554556]]

	tx	rx	Distance from Centroid 0	Distance from Centroid 1	Distance from Centroid 2	Closest_Centroid	Color
0	11.6	21.0	2.931394	231.880055	274.026933	0	r
1	8.7	8.8	9.688478	242.986275	281.130806	0	r
2	15.8	167.2	149.191709	152.572212	258.524815	1	g
3	15.8	176.0	157.988285	151.274717	260.444313	1	g
4	8.3	8.3	10.285929	243.626312	281.690220	0	r
...	...	...	...	...	...	...	...
128906	3.3	9.0	12.238754	246.366171	286.003787	0	r
128907	11.6	18.9	8.834325	233.451001	274.754381	0	r
128908	8.3	9.9	8.780494	242.480952	281.008864	0	r
128909	3.3	11.1	10.775100	244.793178	285.292306	0	r
128910	11.6	21.0	2.931394	231.880055	274.026933	0	r

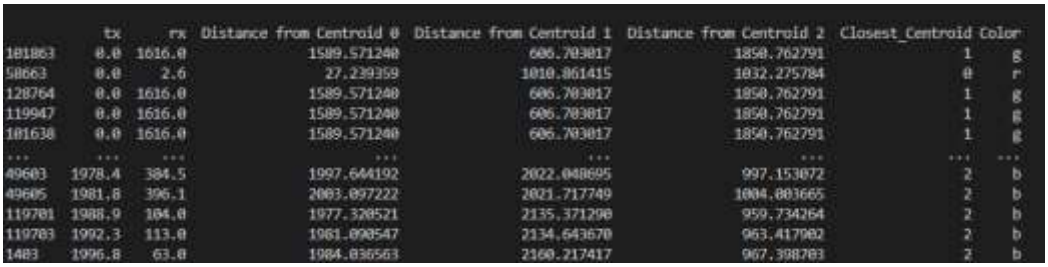
[128911 rows x 7 columns]

	tx	rx	Distance from Centroid 0	Distance from Centroid 1	Distance from Centroid 2	Closest_Centroid	Color
0	11.6	21.0	2.931394	421.615462	274.026933	0	r
1	8.7	8.8	9.688478	433.961521	281.130806	0	r
2	15.8	167.2	149.191709	275.438708	258.524815	1	g
3	15.8	176.0	157.988285	266.652127	260.444313	1	g
4	8.3	8.3	10.285929	434.484709	281.690220	0	r
...	...	...	...	...	...	...	...
128906	3.3	9.0	12.238754	434.118492	286.003787	0	r
128907	11.6	18.9	8.834325	423.712335	274.754381	0	r

Figure 5. Centroid Check Loop Test Results

c. Maximum Value Test

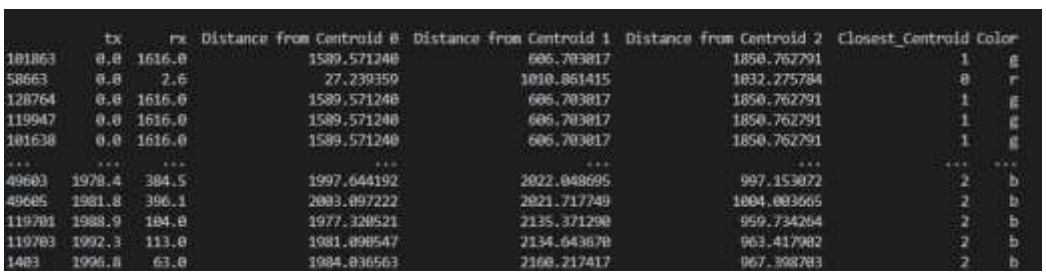
The following test looks for the maximum value in each data to determine the anomaly in each data, where the user determines the anomaly by looking at the data in the image below, where only the top 5 and bottom 5 are displayed.



	tx	rx	Distance from Centroid 0	Distance from Centroid 1	Distance from Centroid 2	Closest_Centroid	Color
101863	0.0	1616.0	1589.571240	606.703017	1850.762791	1	g
58663	0.0	2.6	27.239359	1010.861415	1032.275784	0	r
128764	0.0	1616.0	1589.571240	606.703017	1850.762791	1	g
119947	0.0	1616.0	1589.571240	606.703017	1850.762791	1	g
101638	0.0	1616.0	1589.571240	606.703017	1850.762791	1	g
...	...	...	...	...	...	...	...
49603	1978.4	384.5	1997.644192	2022.048695	997.153072	2	b
49605	1981.8	396.1	2003.097222	2021.717749	1004.083665	2	b
119701	1988.9	104.0	1977.320521	2135.371290	959.734264	2	b
119703	1992.3	113.0	1981.090547	2134.643670	963.417002	2	b
1403	1996.8	63.0	1984.036563	2160.217417	967.398703	2	b

Figure 6. Test Results Finding the Maximum Value

Based on the test results data, an anomaly detection analysis is performed manually by the user to find out anomalies in network traffic at each specified distance. Can be seen in Figure 7 below:



	tx	rx	Distance from Centroid 0	Distance from Centroid 1	Distance from Centroid 2	Closest_Centroid	Color
101863	0.0	1616.0	1589.571240	606.703017	1850.762791	1	g
58663	0.0	2.6	27.239359	1010.861415	1032.275784	0	r
128764	0.0	1616.0	1589.571240	606.703017	1850.762791	1	g
119947	0.0	1616.0	1589.571240	606.703017	1850.762791	1	g
101638	0.0	1616.0	1589.571240	606.703017	1850.762791	1	g
...	...	...	...	...	...	...	...
49603	1978.4	384.5	1997.644192	2022.048695	997.153072	2	b
49605	1981.8	396.1	2003.097222	2021.717749	1004.083665	2	b
119701	1988.9	104.0	1977.320521	2135.371290	959.734264	2	b
119703	1992.3	113.0	1981.090547	2134.643670	963.417002	2	b
1403	1996.8	63.0	1984.036563	2160.217417	967.398703	2	b

Figure 7. Anomaly test results on network traffic

Based on the results of the tests conducted to detect anomalies in network traffic by determining the maximum value of each data, the top 5 and bottom 5 data are displayed during testing. Based on the image above, the top 5 data can be grouped into 2 clusters with the code closest centroid 0 and 1 or color codes g and r, while the bottom 5 data can only be grouped into 1 cluster with the code closest centroid 2 and color code b.

4. CONCLUSION

The following conclusions can be drawn from the analysis and discussion of Anomaly Detection in Network Traffic: (1) The findings of this study indicate that the K-Means Clustering algorithm can be used to detect anomalies in network traffic. And demonstrates that the k-means clustering algorithm can group bandwidth data into a predetermined number of clusters by calculating the distance between each data point to determine the data's proximity. (2) This research can detect anomalies in network traffic.

REFERENCES

Aini, F. D., Riadi, I., & Umar, R. (2018). Perancangan Deteksi Anomali Traffic Untuk Investigasi Log Menggunakan Metode K-Means Clusters. *Prosiding SNST Fakultas Teknik*, *1*(1).

Ananto, R. P., Purwanto, Y., & Novianty, A. (2017). Deteksi Jenis Serangan Pada Distributed Denial Of Service Berbasis Clustering dan Classification Menggunakan Algoritma Minkowski Weighted K-Means dan Decision Tree. *EProceedings of Engineering*, *4*(1).

Chakraborty, N. (2013). Intrusion detection system and intrusion prevention system: A comparative study. *International Journal of Computing and Business Research (IJCBR)*, *4*(2), 1–8.

Chandel, S. K. (2017). Intrusion Detection System using K-Means Data Mining and Outlier Detection Approach. *Bangalore: Faculty of Informatics, Masaryk University*.

Fink, G. A., Chappell, B. L., Turner, T. G., & O'Donoghue, K. F. (2002). A metrics-based approach to intrusion detection system evaluation for distributed real-time systems. *Proceedings 16th International Parallel and Distributed Processing Symposium*, 8-pp.

Gopi, E. S. (2007). *Algorithm collections for digital signal processing applications using Matlab*. Springer Science & Business Media.

Harjono, H., & Wicaksono, A. P. (2013). Honeyd untuk Mendeteksi Serangan Jaringan di Universitas Muhammadiyah Purwokerto. *JUITA: Jurnal Informatika*, *2*(4).

Hermanto, T. I. (n.d.). Implementasi Algoritma Association Rule Dan K-Means Sebagai Sistem Rekomendasi Produk Pada Website Penjualan Online. *Stt-Wastukancana. Ac. Id*, 70–73.

Lubis, A. H. (2016). Model segmentasi pelanggan dengan kernel k-means clustering berbasis customer relationship management. *Sinkron: Jurnal Dan Penelitian Teknik Informatika*, *1*(1).

Putra, I. W. O. K., Purwanto, Y., & Suratman, F. Y. (2015). Perancangan dan Analisis Deteksi Anomaly Berbasis Clustering Menggunakan Algoritma Modified K-Means dengan Timestamp Initialization pada Sliding Window. *EProceedings of Engineering*, *2*(2).

Ridho, F., & Kusuma, A. A. (2018). Deteksi Intrusi Jaringan dengan K-Means Clustering pada Akses Log dengan Teknik Pengolahan Big Data. *Jurnal Aplikasi Statistika & Komputasi Statistik*, *10*(1), 53–66.

Rosa, S. L. (2018). Pendeteksian Anomali Penggunaan Internet di LAN Universitas Islam Riau Indonesia. *IT Journal Research and Development*, *3*(1), 72–83.

Theodoridis, S., & Koutroumbas, K. (2006). *Pattern recognition*. Elsevier.

Wulandari, G. F. (2014). Segmantasi Pelanggan Menggunakan Algoritma K-Means Untuk Customer Relationship Management (CRM) Pada Hijab Miulan. *Ind. Mark. Manag*, *1*, 7.

Zulfadhilah, M., Riadi, I., & Prayudi, Y. (2016). Log classification using K-means clustering for identify Internet user behaviors. *International Journal Of Computer Applications*, *154*(3).