# Penerapan Metode Clustering K-Means Untuk Menentukan Nilai Burst Header Packet Flooding Attack Pada Optical Burst Switching

Ikhsan Nur Rizkiana*, Alam Rahmatulloh, Rohmat Gunawan
Program Studi S1 Teknik Informatika, Fakultas Teknik, Universitas Siliwangi
Email: ikhsanfullbuster@gmail.com

**Info Artikel**

**Kata Kunci :**
Keamanan Komputer, Data Mining, flooding BHP, Clustering k-means, Jaringan OBS

**Keywords :**
Computer Security, Data Mining , Flooding BHP , Clustering K-means ,OBS Network.

**Tanggal Artikel**
Dikirim : 19 Februari 2020
Direvisi : 4 Meret 2020
Diterima : 7 Mei 2020

**Abstrak**

Optical Burst Switching merupakan solusi yang menjanjikan dalam teknologi switching saat ini. Salah satu tantangan keamanan utama yang dihadapi kinerja yang mempengaruhi OBS ialah serangan flood terhadap burst header packet. Kondisi tersebut menyebabkan jaringan melambat atau dalam beberapa kasus besarnya ialah denial of service. Dalam hal ini dicoba untuk menerapkan metode clustering dengan algoritma k-means untuk mengetahui nilai data dari Class OBS yang disebabkan oleh flood pada BHP antara lain NB-No Block, Block, No Block, dan NB-Wait. Clustering merupakan metode pengelompokan data menggunakan algoritma k-means yang banyak digunakan dalam berbagai penerapan salah satunya untuk keamanan. Hasil Penelitian menunjukan jumlah nilai data flooding BHP lebih besar terdapat pada class NB-No Block dan NB-Wait.

**Abstarct**

*Optical Burst Switching is a promising solution in current switching technology. One of the main security challenges facing performance that affects OBS is flood attacks against packet header bursts. This condition causes the network to slow down or in some cases the amount is denial of service. In this case try to apply clustering method with k-means algorithm to find out the value of data from Class OBS caused by flooding at BHP, among others, NB-No Block, Block, No Block, and NB-Wait. Clustering is a method of grouping data that is widely used in various applications, one of which is security. The results showed that the amount of BHP flooding data values was greater in the NB-No Block and NB-Wait classes.*

## 1. PENDAHULUAN

Optical Burst Switching (OBS) merupakan sebuah jaringan switching optic yang telah menjadi teknologi switching untuk membangun infrastruktur backbone (koneksi berkecepatan tinggi yang menjadi lintasan utama dalam sebuah jaringan) internet generasi berikutnya [1]. Kondisi OBS perlu diprediksi secara akurat karena OBS sangat rentan terhadap serangan flood (salah satu dari Denial of service yang dapat membanjiri data) pada burst header packet (BHP)merupakan control yang mengendalikan aliran paket data dalam optical burst switching [2]. Namun, dalam kasus jaringan lainnya OBS yang dibanjiri(flooding) dengan BHP berbahaya ditransmisikan oleh penyerang(attacker) tanpa mengirimkan data burst(data masuk) terkait BHP, sehingga menghabiskan sumber daya kinerja jaringan OBS dan dapat menimbulkan masalah denial of service(sebuah serangan terhadap sebuah computer atau server didalam jaringan internet dengan cara menghabiskan sumber (resource) yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar) [3].

Terlepas dari masalah tersebut beberapa percobaan pun dilakukan untuk menangani masalah pada jaringan OBS yang sudah dilakukan pada penelitian sebelumnya diantaranya, Hasan [4] dengan menggunakan Deep Convolution Neural Network untuk mendeteksi data performance dan nilai confusion matrix serangan flood pada

BHP OBS hasilnya berupa tingkat keakurasian performance, Alsboul [3] menerapkan model pendekatan data mining dengan metode naïve bayes untuk memaksimalkan nilai precison rate dan nilai recall dengan hasil klasifikasi terhadap Class, sedangkan Rajab [1] menggunakan metode decision tree untuk pengklasifikasian terhadap behaving dan misbehaving node. DCNN memiliki nilai yang tinggi dalam penentuan akurasi data [4], sedangkan naïve bayes dalam keakuratan lebih rendah dibanding DCNN , dalam menghasilkan nilai recall dan precision naïve bayes dapat mencapai 90,4% [3]. Penelitian Rajab [1] menggunakan decision tree untuk pengklasifikasian yang hasilnya berupa pohon keputusan. Dari ke-3 penelitian tersebut masih terdapat hal yang perlu diperhatikan yaitu berapa jumlah nilai data flooding dari setiap kelas OBS tersebut. Class OBS dibagi menjadi 4 class yaitu NB no Block, Block, No Block, NB Wait, dari setiap class OBS memiliki jalur traffic tersendiri [2].

Maka dari itu perlu adanya penelitian bagaimana menentukan nilai dari data OBS Class diantaranya ialah NB No Block, Block, No Block, NB Wait dengan metode clustering dan K-means. K-means merupakan salah satu algoritma clustering yang tujuannya membagi nilai dari data menjadi beberapa kelompok [5]. Sehingga dapat diketahui nilai dari data BHP flood OBS dari setiap Class.

## 2. METODE PENELITIAN

### 2.1 *Pengumpulan Data*

Data yang digunakan adalah dataset dari Burst Header Packet (BHP) flooding attack on Optical Burst Switching (OBS) yang didalamnya terdapat 22 atribut.

**Tabel 1. Daftar Attribute data OBS**

| Index | Nomor | Index | Attribute |
|---|---|---|---|
| 1 | *@attribute Node numeric* | 12 | *@attribute Packet_Transmitted numeric* |
| 2 | *@attribute 'Utilised Bandwith Rate' numeric* | 13 | *@attribute Packet_Received numeric* |
| 3 | *@attribute 'Packet Drop Rate' numeric* | 14 | *@attribute Packet_lost numeric* |
| 4 | *@attribute Full_Bandwidth numeric* | 15 | *@attribute Transmitted_Byte numeric* |
| 5 | *@attributeAverage_Delay_Time_Per_Sec numeric* | 16 | *@attribute Received_Byte numeric* |
| 6 | *@attributePercentage_Of_Lost_Pcaket_Rate numeric* | 17 | *@attribute 10-Run-AVG-Drop-Rate numeric* |
| 7 | *@attributePercentage_Of_Lost_Byte_Rate numeric* | 18 | *@attribute 10-Run-AVG-Bandwith-Use numeric* |
| 8 | *@attribute 'Packet Received Rate' numeric* | 19 | *@attribute 10-Run-Delay numeric* |
| 9 | *@attribute 'of Used_Bandwidth' numeric* | 20 | *@attribute 'Node Status' {B,NB,'P NB'}* |
| 10 | *@attribute Lost_Bandwidth numeric* | 21 | *@attribute 'Flood Status' numeric* |
| 11 | *@attribute 'Packet Size_Byte' numeric* | 22 | *@attribute 'Class' {'NB-No Block',Block,'No Block',NB-Wait}* |

https://archive.ics.uci.edu/ml/datasets/Burst+Header+Packet+%28BHP%29+flooding+attack+on+Optical+Burst+Switching+%28OBS%29+Network#

### 2.2 *Preproses Data*

Pada preproses menggunakan langkah data mining dan tools machine learning yang digunakan untuk mendeteksi jumlah keseluruhan data data ialah menggunakan Weka.

#### 2.2.1 *Machine Learning*

Suatu bidang ilmu komputer yang memberikan kemampuan pembelajaran kepada komputer untuk mengetahui sesuatu tanpa pemrograman yang jelas. Machine learning dapat didefinisikan sebagai metode komputasi berdasarkan pengalaman untuk meningkatkan performa atau membuat prediksi yang akurat [6][7][8].

### 2.2.2 *Data Mining*

Data mining merupakan proses yang mempekerjakan satu atau lebih teknik pembelajaran komputer (machine learning) untuk menganalisis dan mengekstraksi pengetahuan (knowledge) secara otomatis. Tujuan dilakukannya data mining dapat dikelompokkan menjadi 2, yaitu untuk dapat memahami lebih jauh mengenai perilaku data yang diamati, atau sering disebut sebagai deskripsi, dan untuk dapat memperkirakan kondisi yang akan terjadi di masa mendatang atau disebut Prediksi [9][10][11].

### 2.3 *Pemrosesan Data*

Setelah dataset didefinisikan menggunakan weka proses selanjutnya menggunakan metode *Clustering* dan proses *cluster* sendiri menggunakan algoritma *K-means* pada pemrosesan output data.
Pada penelitian digunakan 3 percobaan diantaranya pengujian 1 dliakukan tanpa menggunakan *percentage split,* untuk percobaan 2 dan 3 menggunakan split sekitar 60% dan 70% [12].

### 2.3.1 *Clustering*

Metode clustering juga harus dapat mengukur kemampuannya sendiri dalam usaha untuk menemukan suatu pola tersembunyi pada data yang sedang diteliti [13]. Terdapat berbagai metode yang dapat digunakan untuk mengukur nilai kesamaan antar objek-objek yang dibandingkan. Salah satunya ialah dengan *weighted Euclidean Distance*. *Euclidean distance* menghitung jarak dua buah point dengan mengetahui nilai dari masing-masing atribut pada kedua poin tersebut [14]. Berikut formula pada persamaan 1 yang digunakan untuk menghitung jarak dengan *Euclidean distance* :

$$distance(p,q) = (\sum_{k}^{n} \mu k|Pk - qk|r\,)\frac{1}{r} \qquad (1)$$

Keterangan:
N = Jumlah record data
K= Urutan field data
r= 2
μk= Bobot field yang diberikan user

### 2.3.2 *K-means*

K-means clustering merupakan adalah suatu metode penganalisaan data atau metode Data Mining yang melakukan proses pemodelan tanpa supervisi *(unsupervised)* dan merupakan salah satu metode yang melakukan pengelompokan data dengan sistem partisi [15]. Metode K-means merupakan algoritma sederhana dan cepat yang berupaya untuk meningkatkan arbitrary dalam k-means clustering [16]. Berikut persamaan 2 dari *k-means*:

$$\emptyset = \sum_{X_i \in S_k} Min_{c \in C} \left|\left|X_i - C\right|\right|^2 \qquad (2)$$

## 3. HASIL DAN PEMBAHASAN

### 3.1 Pengujian 1

Pengujian 1 dilakukan tanpa menggunakan proses *percentage split,* data diolah pada menu cluster pada tab cluster mode dengan atribut yang digunakan sebagai penentu cluster adalah atribut Class. Hasil dari pengujian

tersebut tertera pada tab clusterer output yang digambarkan pada tabel 2 hasil cluster output model, yakni clusterer 0 dan clusterer 1 clustered instanace sekitar 48% dan 52%.

**Tabel 2. Model Summary Clusterer Output Pengujian I**

| jumlah | *Clusterer instance* |
|---|---|
| *Cluster 0* | 51% (520) |
| *Clusterer 1* | 49% (555) |

Kemudian didapat model classes to cluster dari pengujian pertama tergambar dalam tabel 3 yang berisi niliai clusterer 0 dan 1 class *NB-no Block, Block, No Block, NB-Wait.*

**Tabel 3. Model Classes to Cluster 0 Pengujian I**

| Clusterer 0 | Class |
|---|---|
| *165* | NB-no Block |
| *120* | Block |
| *0* | No Block |
| *235* | NB-Wait |

Jumlah clusterer 1 berbeda dengan 0 dikarenakan hasil instance lebih kecil dari clusterer 1 tergambar dalam tabel 4 yang berisi kategori class beserta jumlahnya.

**Tabel 4. Model Classes to Cluster 1 Pengujian I**

| Clusterer 1 | *Class* |
|---|---|
| *335* | NB-no Block |
| *0* | Block |
| *155* | No Block |
| *65* | NB-Wait |

### 3.2 Pengujian 2

Data terlebih dahulu di split sekitar 60% melalui opsi *percentage split* pada tab *cluster* mode dengan atribut yang digunakan sebagai penentu cluster adalah atribut Class. Hasil dari pengujian tersebut tertera pada tab clusterer output yang digambarkan pada tabel 5 hasil *cluster output* model, yakni clusterer 0 dan *clusterer* 1 *clustered instance* sekitar 73% dan 27%.

**Tabel 5. Model Summary Clusterer Output Pengujian II**

| jumlah | *Clusterer instance* |
|---|---|
| *Cluster 0* | 73% (790) |
| *Clusterer 1* | 27% (285) |

Kemudian pada didapat Model *Classes to Cluster* dari pengujian pertama tergambar dalam tabel 6 yang berisi nilai *clusterer* 0 dan 1 class *NB-no Block, Block, No Block, NB-Wait.*

**Tabel 6. Model Classes to Cluster 0 Pengujian II**

| *Clusterer 0* | Class |
|---|---|
| 440 | NB-no Block |
| 0 | Block |
| 155 | No Block |
| 195 | NB-Wait |

Jumlah clusterer 1 berbeda dengan 0 dikarenakan hasil instance lebih kecil dari clusterer 0 tergambar dalam tabel 7 yang berisi kategori class beserta jumlahnya.

**Tabel 7. Model Classes to Cluster 1 Pengujian II**

| *Clusterer 1* | Class |
|---|---|
| 205 | NB-no Block |
| 70 | Block |
| 70 | No Block |
| 185 | NB-Wait |

### 3.3 Pengujian 3

Data terlebih dahulu di split sekitar 70% melalui opsi *percentage split* pada *tab cluster mode* dengan atribut yang digunakan sebagai penentu *cluster* adalah atribut Class. Hasil dari pengujian tersebut tertera pada tab clusterer output yang digambarkan pada tabel 8 hasil cluster output model, yakni *clusterer* 0 dan *clusterer* 1 *clustered instance* sekitar 89% dan 11%.

**Tabel 8. Model Summary Clusterer Output Pengujian III**

| jumlah | *Clusterer instance* |
|---|---|
| *Cluster 0* | 89% (960) |
| *Clusterer 1* | 11% (115) |

Kemudian pada didapat Model *Classes to Cluster* dari pengujian pertama tergambar dalam tabel 9 yang berisi nilai *clusterer* 0 dan 1 *class NB-no Block, Block, No Block, NB-Wait.*

**Tabel 9. Model Classes to Cluster 0 Pengujian III**

| *Clusterer 0* | Class |
|---|---|
| 430 | NB-no Block |
| 100 | Block |
| 155 | No Block |
| 275 | NB-Wait |

Jumlah *clusterer* 1 berbeda dengan 0 dikarenakan hasil instance lebih kecil dari *clusterer* 0 tergambar dalam tabel 10 yang berisi kategori class beserta jumlahnya.

**Tabel 10. Model Classes to Cluster 1 Pengujian III**

| *Clusterer 1* | Class |
|---|---|
| 70 | NB-no Block |
| 20 | Block |
| 0 | No Block |
| 25 | NB-Wait |

Berikut merupakan hasil dari pengukuran digambarkan oleh tabel 11 dan 12

**Tabel 11. data cluster 0**

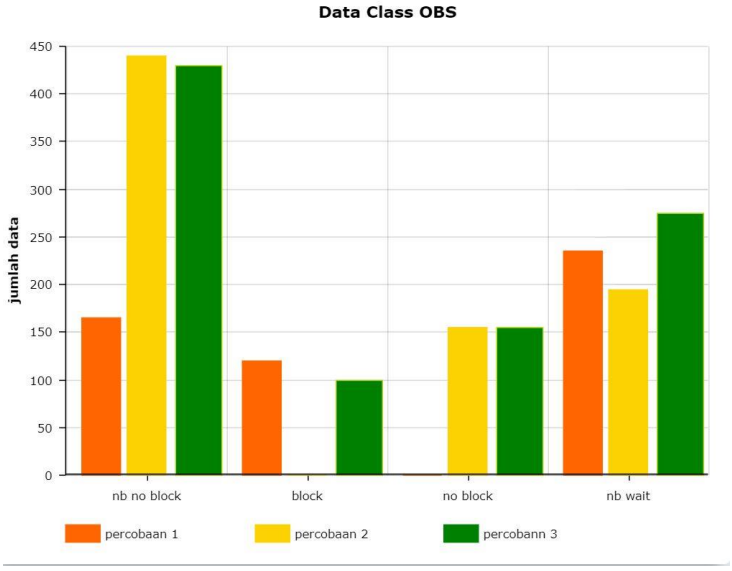
| **UJI COBA** | **Dataset *cluster* 0** | | | | | ***Clustered instances*** |
|---|---|---|---|---|---|---|
| | NB no blok | block | No block | NB Wait | Jumlah | |
| **I** | 165 | 120 | 0 | 235 | **520** | 48% |
| **II** | 440 | 0 | 155 | 195 | **790** | 73% |
| **III** | 430 | 100 | 155 | 275 | **960** | 89% |
| | | | | Rata-rata | **558,3** | 70% |

Pada tabel 11 merupakan keseluruhan hasil yang didapat pada cluster 0. Berikut merupakan rincian lengkapnya yaitu pada percobaan pertama tanpa menggunakan split didapat Nb-no Block, Block, No Block, NB-Wait adalah 165, 120, 0, dan 235 dengan total data dihasilkan 520. Pada percobaan kedua dengan proses split 60% didapat Nb-no Block, Block, No Block, NB-Wait adalah 440, 0, 155, dan 195 dengan total data dihasilkan 790. Pada percobaan ketiga dengan proses split 70% didapat Nb-no Block, Block, No Block, NB-Wait adalah 430, 100, 155, dan 275 dengan total yang dihasilkan 960

**Tabel 12 data cluster 1**

| UJI COBA | Dataset cluster 1 | | | | | Clustered Instances |
|---|---|---|---|---|---|---|
| | NB no blok | block | No block | NB Wait | Jumlah | |
| **I** | 335 | 0 | 155 | 65 | **555** | 52% |
| **II** | 60 | 120 | 0 | 105 | **285** | 27% |
| **III** | 70 | 20 | 0 | 25 | **115** | 45% |
| | | | | Rata-rata | **318,3** | 41,3% |

Pada tabel 12 merupakan keseluruhan hasil yang didapat pada cluster 0. Berikut merupakan rincian lengkapnya yaitu pada percobaan pertama tanpa menggunakan split didapat Nb-no Block, Block, No Block, NB-Wait adalah 335, 0, 15, dan 65 dengan total yang dihasilkan 555. Pada percobaan kedua dengan proses split 60% didapat Nb-no Block, Block, No Block, NB-Wait adalah 60, 120, 0, dan 105 dengan total yang dihasilkan 285. Pada percobaan ketiga dengan proses split 70% didapat Nb-no Block, Block, No Block, NB-Wait adalah 70, 20, 0, dan 25 dengan total yang dihasilkan 115.




**Gambar 1. grafik hasil class OBS**

Pada gambar 1 menunjukan gambar grafik hasil keseluruhan dari data yang telah di *cluster* dimana class *nb-no block* dan *nb wait(Non Behave)* mendapat nilai lebih besar daripada class *block* dan *no block.*

## 4. KESIMPULAN

Dari ketiga hasil percobaan dengan menggunakan K-means hasil clustering yang didapat bahwa nilai dari 4 class dari OBS BHP Flooding merupakan nilai dalam bentuk data flood yang dimana NB(nonbehave) yaitu NB-no Block dan NB-Wait mendapat nilai terbesar dibandinkan dengan Block dan No Block, hal ini terbukti pada pengujian dengan melihat jumlah data, dimana data dengan nilai tertinggi di setiap clusternya terdapat pada NB-no Block dan NB-Wait. Dalam penelitian ini pengujian pertama tanpa menggunakan split mode memiliki nilai jarak atau selisih yang sama terdapat pada class NB-no Block dan NB-Wait yaitu dengan kisaran nilai 70 dari setiap clusternya.

Untuk masalah pada Flooding sendiri ada baiknya mengecek keaadaan server dan mempersiapkan penangannya sebelum terjadi serangan yang lebih serius seperti memperkuat keamanan firewall atau Menghubungi Spesialis DDoS, jika terkena terkena serangan tersebut.

**DAFTAR PUSTAKA**

[1] A. Rajab, C. T. Huang, and M. Al-Shargabi, "Decision tree rule learning approach to counter burst header packet flooding attack in Optical Burst Switching network," *Opt. Switch. Netw.*, vol. 29, pp. 15–26, 2018.
[2] C. S. R. Murthy, *An analytical approach to optical Burst switched networks*. Springer New York Dordrecht Heidelberg London.
[3] R. Alshboul, "Flood Attacks Control in Optical Burst Networks by Inducing Rules using Data Mining," vol. 18, no. 2, pp. 160–167, 2018.
[4] M. Zahid Hasan, K. M. Zubair Hasan, and A. Sattar, "Burst header packet flood detection in optical burst switching network using deep learning model," *Procedia Comput. Sci.*, vol. 143, pp. 970–977, 2018.
[5] K. LaRose and K. J. Elwood, "Performance of headed shear stud clusters for precast concrete bridge deck panels," *TAC/ATC 2006 - 2006 Annu. Conf. Exhib. Transp. Assoc. Canada Transp. Without Boundaries*, no. December, 2006.
[6] C. Eckart, "Some Studies," *Phys. Rev.*, vol. 47, pp. 552–558, 1935.
[7] H. Widayu, S. Darma, N. Silalahi, and Mesran, "Data Mining Untuk Memprediksi Jenis Transaksi Nasabah Pada Koperasi Simpan Pinjam Dengan Algoritma C4.5," *Issn 2548-8368*, vol. Vol 1, No, no. June, p. 7, 2017.
[8] A. Ahmad, "Mengenal Artificial Intelligence, Machine Learning, Neural Network, dan Deep Learning," *J. Teknol. Indones.*, no. October, p. 3, 2017.
[9] D. Tantangannya, D. Masa, D. Sri, and A. Thamrin, "Penggunaan Data Mining Saat Ini," *Stat. Komputasi*, vol. 3, no. 1, 2006.
[10] Y. G. Sucahyo, "D a t a M i n i n g," *Database*, vol. 35, no. 3, pp. 1–3, 2003.
[11] S. Shadroo and A. M. Rahmani, "Systematic survey of big data and data mining in internet of things," *Comput. Networks*, vol. 139, pp. 19–47, 2018.
[12] R. Patil, S. Deshmukh, and K. Rajeswari, "Analysis of SimpleKMeans with Multiple Dimensions using WEKA," *Int. J. Comput. Appl.*, vol. 110, no. 1, pp. 14–17, 2015.
[13] C. Song, F. Liu, Y. Huang, L. Wang, and T. Tan, "Auto-encoder based data clustering," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8258 LNCS, no. PART 1, pp. 117–124, 2013.
[14] R. Parhizkar, J. Ranieri, and M. Vetterli, "Euclidean Distance Matrices [," no. november, pp. 12–30, 2015.
[15] Agus Nur Khormarudin, "Teknik Data Mining : Algoritma K-Means Clustering," pp. 1–12, 2016.
[16] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," *Proc. Annu. ACM-SIAM Symp. Discret. Algorithms*, vol. 07-09-January-2007, pp. 1027–1035, 2007.