IMPLEMENTASI EUCLIDEAN DAN CHEBYSHEV DISTANCE PADA K-MEDOIDS CLUSTERING

Gea Putri Inka Rani, Abdul Aziz, Muhammad Priyono Tri Sulistyono
Program Studi Teknik Informatika, Fakultas Sains dan Teknologi
Universitas PGRI Kanjuruhan Malang, Jalan S. Supriyadi No. 48 Malang, Indonesia
geaputriinkarani247@gmail.com

ABSTRAK

Clustering merupakan salah satu proses dalam data mining untuk melakukan pengelompokan sekumpulan data yang memiliki kemiripan karakteristik yang sama akan berada dalam satu *cluster* yang sama dan yang memiliki kemiripan karakteristik yang lain akan berada dalam *cluster* yang lain. Metode *clustering* atau pengelompokan ada banyak salah satu diantaranya adalah *K-Medoids* atau *PAM (Partitioning Around Medoids)* yang merupakan algoritma yang mirip dengan k-means karena memecah dataset menjadi kelompok-kelompok. Algoritma *k-medoids* digunakan untuk mengatasi kelemahan *k-means* yang sensitif terhadap *noise* dan *outlier*. Tingkat kemiripan karakteristik dalam clustering ditentukan dengan mengukur jarak (*distance measure*) antar data. Pada penelitian kali ini menggunakan *Euclidean Distance* dan *Chebyshev Distance* guna mengetahui hasil *cluster* yang optimal dari masing-masing *distance measure.* Penelitian ini bertujuan untuk mengetahui perbandingan hasil *implementasi euclidean* dan *chebyshev distance* pada *k-medoids clustering*. Pada penelitian ini hasil *clustering* yang diperoleh dengan menggunakan *euclidean* dan *chebyshev distance* pada *k-medoids* menunjukkan bahwa penggunaan euclidean distance menghasilkan cluster yang lebih optimal.

Kata kunci: *Clustering, K-Medoids, Euclidean Distance, Chebyshev Distance*

1. PENDAHULUAN

Clustering adalah teknik data mining yang digunakan untuk mengelompokkan sekumpulan data dengan karakteristik yang sama ke dalam cluster yang sama dan data dengan karakteristik yang sama ke dalam cluster yang berbeda. Terdapat banyak metode clustering, salah satunya adalah K-Medoids. *Partitioning Around medoids* atau yang sering juga disebut sebagai *k-medoids* merupakan suatu algoritma dalam data mining yang mirip dengan k-means dimana k-medoids membagi dataset menjadi beberapa kelompok. Tingkat kemiripan karakteristik dalam clustering ditentukan dengan mengukur jarak (*distance measure*) antar data [1]. Jarak yang diperoleh dalam proses clustering akan menentukan anggota setiap cluster yang terbentuk dimana anggota dalam setiap cluster memiliki kemiripan karakteristik berdasarkan kedekatan jarak antar data. Dalam proses data mining khususnya clustering terdapat banyak distance measure yang dapat digunakan sesuai dengan kebutuhan penelitian. Pemilihan *distance measure* yang tepat dapat meningkatkan performa algoritma dalam melakukan *clustering* [2]. Sebuah penelitian tahun 2016 oleh Mario Anggara melakukan penelitian yang membandingkan ukuran jarak antara jarak Euclidean, jarak Manhattan, dan jarak Chebyshev pada K-Means, menyimpulkan bahwa penggunaan jarak Chebyshev memberikan mampu menghasilkan cluster terbaik [3]. Penelitian tahun 2019 lainnya oleh Nishom membandingkan ukuran jarak antara Euclidean, Minkowski, dan Manhattan menggunakan algoritma K-Means dan menemukan hasil yangh berbeda dari penelitian sebelumnya yakni pada penilitian ini menyimpulkan bahwa jarak Euclidean memiliki akurasi tertinggi dari cluster yang dihasilkan dibandingkan dengan dua pengukuran jarak lainnya [4].

Pada penelitian kali ini akan menggunakan *euclidean* dan *chebyshev distance* dengan membandingkan hasil cluster dari *distance measure* yang berbeda berdasarkan nilai validasi *silhouette coefficient index* yang diperoleh. Jarak Euclidean adalah metode penghitungan jarak antara dua titik dalam ruang *euclidean* dimana jarak *euclidean* diperoleh dari hasil perhitungan akar kuadrat dari jumlah selisih kuadrat antar objek persegi. *Chebyshev distance* merupakan metode penghitungan jarak yang berdasarkan nilai mutlak atau absolut dari selisih antara koordinat titik-titik. Jika ada dua buah vektor yang berbeda, jarak chebyshev didasarkan pada nilai absolut dari perbedaan antara elemen dalam vektor, dan jumlah data secara otomatis harus sama.

Data yang digunakan merupakan data penjualan kendaraan dari gaikindo periode tahun 2021. Kendaraan merupakan alat transportasi yang sangat dibutuhkan dalam kehidupan sehari-hari khususnya mobil. Menurut pendapat dari Yohanes Nangoi selaku ketua umum Gaikindo dalam wawancaranya dengan media berita CNN Indonesia menyatakan bahwa penjualan mobil sepanjang tahun 2021 terbilang bagus karena adanya dukungan dari pemerintah melalui diskon pajak penjualan atas barang mewah. Berdasarkan data dari Gaikindo (Gabungan Industri Kendaraan Bermotor Indonesia) penjualan mobil di Indonesia mencatat angka yang tinggi pada tahun 2021 yakni sebesar 887.200unit terjual. Hal ini disebabkan oleh semakin meningkatnya peminat mobil di kalangan masyarakat Indonesia. Oleh sebab itu akan dilakukan pengelompokan agar konsumen

mengetahui merek mobil yang tergolong laris dan kurang laris dari segi kategori yang paling diminati. Data wholesales gaikindo yang digunakan dalam penelitian ini nantinya akan di pecah menjadi beberapa data berdasarkan spesifikasi mobil sebelum dilakukan proses *clustering* dengan mengguna *k-medoids*. Sehingga diperoleh hasil *cluster* mobil laris dan kurang laris berdasarkan dengan spesifikasi dari mobil.

2. TINJAUAN PUSTAKA

2.1 K-Medoids atau Partitioning Around Medoid (PAM)

K-Medoids atau Partitioning Around Medoid (PAM) ditemukan oleh Kaufman dan Rousseeuw pada tahun 1990. Clustering melibatkan pengelompokkan, pemisahan, atau pembagian objek data menjadi k cluster, dimana jumlah k lebih kecil atau sama dengan data cluster. Medoid adalah objek yang diasumsikan mewakili cluster dan pusat cluster [5]. Adapun proses clustering menggunakan K-Medoids adalah sebagai berikut:

1. Inisialisasi pusat cluster sebanyak jumlah cluster (k).
2. Alokasikan setiap objek dengan cluster terdekat menggunakan perhitungan jarak yang diberikan yaitu *euclidean distance* dengan rumus:

   $$d(x,y) = \sqrt{\sum_{i=0}^{n}(x_i - y_i)^2} \quad (2.1)$$

   Dimana:
   x = data medoid ke-*i*
   y = data *y* ke-*i*
   n = banyak data atau objek
   dan *chebyshev distance* dengan rumus:

   $$d(x,y) = max_{i=1}^{n}|x_i - y_i| \quad (2.2)$$

   Dimana:
   x = data medoid ke-*i*
   y = data *y* ke-*i*
   n = banyak data atau objek
3. Pilih secara acak objek di setiap cluster sebagai kandidat medoid baru.
4. Hitung jarak setiap objek disetiap cluster menggunakan calon medoid yang baru dengan persamaan (2.1) dan (2.2).
5. Hitung total simpangan yang dihasilkan dengan mengurangkan total jarak lama dari jarak baru. Jika nilai S kurang dari 0, maka objek ditukar dengan data cluster dan kumpulan objek baru terbentuk sebagai medoid.
6. Ulangi tahap ke-3 sampai ke-5 jika terjadi perubahan medoid, jika tidak ada perubahan, cluster dan anggota dari setiap cluster pada iterasi sebelumnya.

2.2 Euclidean distance

Euclidean distance merupakan metode perhitungan jarak yang digunakan untuk mengukur jarak antara dua titik dalam ruang euclidean [4]. Jarak euclidean adalah akar dari jumlah selisih kuadrat antara objek kuadrat [1].

2.3 Chebyshev Distance

Chebyshev Distance adalah metode yang pengukuran jarak berdasarkan nilai absolut atau nilai mutlak dari selisih sepasang titik koordinat. Jika memiliki dua vektor dengan nilai berbeda untuk setiap elemen, jarak yang diukur oleh chebyshev adalah berdasarkan pada nilai mutlak dari perbedaan antara elemen-elemen pada vektor tersebut dan jumlah data secara otomatis harus sama.

2.4 Silhouette Coefficient

Silhouette Coefficient merupakan salah satu metode validasi yang digunakan untuk melihat kualitas dari sebuah cluster berbasis kriteria internal. Silhouette index melakukan evaluasi penempatan setiap objek dalam setiap klaster dengan membandingkan jarak rata-rata objek dalam satu klaster dengan jarak rata-rata objek dalam klaster yang berbeda [1]. Nilai silhouette yang dihitung dapat memberikan hasil yang bervariasi antara -1 hingga 1. Si=1 berarti objek atau data berada dalam cluster yang tepat, jika si=0 maka objek atau data berada diantara dua cluster, maka objek tersebut tidak jelas harus dimasukkan ke dalam cluster yang mana dan jika si=-1 berarti objek atau data mengalami overlapping sehingga objek tersebut lebih tepat dimasukan ke dalam cluster yang terpisah[6]. Adapun tahapan perhitungan dengan Silhouette Coefficient adalah sebagai berikut:

1. Hitung rata-rata jarak objek atau data dalam satu cluster dengan persamaan:

   $$a(i) = \frac{1}{[A]-1}\sum j \in_{A,j\neq i} d(i,j) \quad (2.3)$$

2. Hitung rata-rata jarak objek dengan semua objek pada cluster lain dengan persamaan:

   $$d(i,C) = \frac{1}{[A]}\sum j \in C\ d(i,j) \quad (2.4)$$

   Dimana *d (i, j)* merupakan jarak rata-rata objek *i* dengan semua objek pada cluster lain C dimana A ≠ C.

   $$b(i) = \min_{C\neq A} d\ (i,C) \quad (2.5)$$

3. Hitung nilai silhouette coefficient dengan persamaan:

   $$s(i) = \frac{b(i)-a(i)}{max(a(i),b(i))} \quad (2.6)$$

Nilai s(i) berada antara -1 dan 1 dimana semakin dekat nilai s(i) ke 1 maka semakin baik pengelompokan dalam satu cluster. Namun apabila diperoleh nilai s(i) yang semakin dekat ke -1 maka semakin buruk pengelompokan dalam satu cluster.

2.5 Python

Python merupakan salah satu bahasa pemrograman yang paling mudah dipahami dimana metode beroriantasi objek dapat digunakan untuk mengeksekusi urutan pernyataan secara langsung dan semantik dinamis dapat digunakan untuk menyediakan beberapa tingkat keterbacaan sintaks

2.6 Numpy

Numpy adalah perpustakaan untuk bahasa pemrograman python yang mendukung set multidimensi besar atau array dan matriks dengan banyak koleksi fungsi matematika tingkat lanjut untuk memanipulasi array.

2.7 Scikit-learn

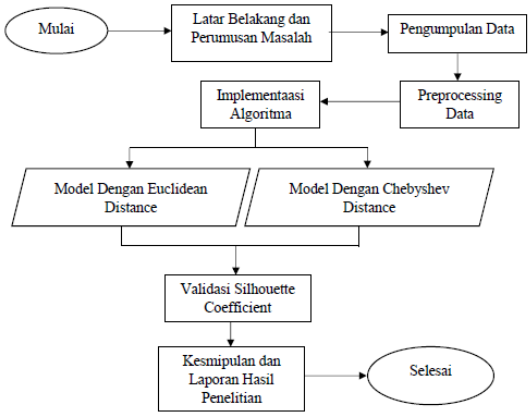
Scikit-learn adalah pustaka pembelajaran machine learning untuk bahasa pemrograman Python yang menampilkan berbagai metode klasifikasi, regresi dan pengelompokan termasuk Support Vector Machine, Random Forest, Gradient Boosting, K-Means dan DBSCAN dan dirancang untuk beroperasi dengan pustaka numerik dan ilmiah.

2.8 Jupyter Notebook

Jupyter notebook adalah aplikasi website sumber terbuka yang memungkin user untuk membuat dan membagikan dokumen interaktif dengan kode langsung, persamaan, visualisasi, dan teks naratif. Jupyter notebook digunakan untuk membersihkan data dan mengubah informasi data, simulasi numerik, pemodelan statistik, visualisasi informasi, pembelajaran mesin dan masih banyak tujuan lainnya.

3. METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini adalah metode penelitian kuantitatif. Penelitian kuantitatif adalah upaya untuk mempelajari masalah, pertanyaannya adalah bagaimana peneliti dapat mengumpulkan data, menentukan variabel, mengukurnya secara numerik, dan melakukan analisis menurut prosedur statistik yang berlaku. Penelitian kuantitatif bertujuan untuk membantu menarik kesimpulan atau menggeneralisasi teori prediksi yang sesuai [7]. Bagan atau alur kerja berikut dilakukan oleh peneliti dalam penelitian ini.



Gambar 1. Diagram Alur Penelitian

3.1. Pengumpulan data

Pengumpulan data untuk penelitian ini dilakukan dengan mengunjungi situs resmi Gaikindo dimana data yang diambil merupakan jenis data sekunder. Data yang diambil untuk penelitian kali ini merupakan data spesifikasi produk terjual selama 1 tahun terakhir yakni selama tahun 2021.

3.2. Pre-processing data

Preprocessing data adalah proses pembersihan data, transformasi data, dan normalisasi data sebelum akhirnya dapat diproses menggunakan algoritma. Pembersihan data dilakukan dengan menghilangkan atau menghapus data yang tidak bernilai, memeriksa konsistensi data dan mengoreksi data yang salah. Karena data yang digunakan dalam clustering harus berupa data numerik maka perlu dilakukan transformasi data dengan mengubah tipe data kategorik ke dalam bentuk numerik sehingga data dapat dimasukan dalam proses perhitungan algoritma k-medoids. Data yang ditransformasi diurutkan berdasarkan output data mulai dari yang terbesar ke terkecil. Normalisasi data merupakan teknik yang dilakukan pada data untuk memetakan data dalam rentang tertentu. Tujuan dari normalisasi data adalah untuk mempermudah dalam proses penerapan algoritma dimana peneliti perlu untuk mengubah nilai pada kolom numerik dalam dataset untuk menggunakan skala umum sehingga data dapat diolah dalam proses clustering. Terdapat beberapa metode normalisasi data yang dapat digunakan untuk pengolahan data. Pada penelitian ini menggunakan normalisasi *min max* yang mengubah atribut data diubah ke menjadi angka dengan skala kecil antara -1 sampai 1 atau 0 sampai 1[8]. Adapun persamaan yang digunakan untuk normalisasi data pada penelitian ini adalah sebagai berikut:

$$x_n = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Dimana:
Xn = Nilai data yang dinormalisasi
Xi = Nilai objek data yang akan dinormalisasi
Xmin = Nilai minimum data
Xmax = Nilai maksimum data

3.3. Implementasi Algoritma

Pada tahap ini, data yang telah melewati tahap preprocessing data telah siap untuk diolah menggunakan algoritma *K-Medoids*. *K-Medoids* atau *Partitioning Around Medoids (PAM)* ditemukan oleh Kaufman dan Rousseeuw pada tahun 1990. Saat membelah cluster, objek data dikelompokkan, dipisahkan, atau dipecah dalam K cluster, di mana K adalah jumlah data yang lebih sedikit atau dikelompokkan. Medoid adalah objek yang mewakili kedua cluster dan pusat cluster[9]-[5]. Adapun proses clustering menggunakan K-Medoids sebagai berikut:

1. Inisialisasi pusat cluster sebanyak jumlah k cluster.
2. Alokasikan setiap objek ke cluster terdekat dengan perhitungan jarak yang telah ditentukan yakni *euclidean distance* dengan rumus (2.1) dan *chebyshev distance* dengan rumus (2.2)
3. Pilih objek cluster secara acak pada masing-masing sebagai calon medoid baru.

4. Gunakan persamaan (1) dan (2) untuk menghitung jarak setiap objek di setiap cluster menggunakan medoid kandidat baru.
5. Hitung simpangan total yang diperoleh dengan mengurangkan jumlah jarak lama dari jumlah jarak baru. Untuk nilai S < 0, objek ditukar untuk membentuk satu set cluster dengan objek baru sebagai mediad.

Ulangi tahap ke-3 sampai ke-5 jika terjadi perubahan medoid. Jika tidak, maka diperoleh anggota cluster dari setiap cluster yang terbentuk pada iterasi sebelumnya.

3.4. Validasi Silhouette Coefficient Index

Setelah diperoleh hasil *clustering* dari dua perhitungan jarak yang berbeda, maka dilakukan validasi dengan menggunakan metode *Silhouette Coefficient* untuk menentukan nilai *silhouette*. Nilai silhouette inilah yang nantinya akan dibandingkan untuk mengetahui hasil cluster yang terbaik dari perhitungan jarak yang berbeda. *Silhouette Coefficient* merupakan metode validasi untuk memeriksa kualitas *cluster* yang terbentuk dengan menggunakan kriteria internal. *Silhouette coefficient index* melakukan evaluasi penempatan setiap objek dalam setiap *cluster* dengan membandingkan jarak rata-rata objek dalam satu *cluster* dengan jarak rata-rata objek dalam *cluster* yang berbeda [1]. Hasil perhitungan nilai *SI* berubah dari -1 menjadi 1. S*i*=1 berarti objek atau data tersebut berada pada *cluster* yang benar, untuk *si*=0 jelas bahwa objek atau data tersebut berada di antara dua *cluster* maka objek tersebut tidak ada. Cluster tersebut harus disertakan, dan *si*=-1 artinya objek atau data mengalami *overlapping* sehingga objek tersebut lebih baik diikutsertakan ke dalam *cluster* lain [6].

4. HASIL DAN PEMBAHASAN

4.1. Data

Data yang digunakan dalam penelitian ini adalah data *Wholesales Gaikindo 2021* yang merupakan data spesifikasi produk yang terjual dari gaikindo. Terdapat 36 atribut dalam data spesifikasi produk terjual. Namun tidak semua atribut data digunakan dalam penelitian ini, dan dari 36 atribut yang termasuk dalam dataset yang digunakan, dipilih menjadi 5 atribut yang sangat relevan untuk mendukung proses penelitian. Lima atribut yang akan digunakan adalah *Category, Brand, CC, Trans dan Total 2021 (Output).*

4.2. Implementasi K-Medoids Clustering

Pada tahap ini akan dilakukan implementasi algoritma *k-medoids* untuk memperoleh hasil *cluster* dengan perhitungan jarak *euclidean* ((ED) dan *chebyshev distance (CD)*. Dilakukan dua percobaan untuk membandingkan dua perhitungan jarak yang berbeda. Percobaan pertama dilakukan *clustering* dengan menggunakan perhitungan jarak *euclidean,* kemudian pada percobaan kedua *clustering* akan diproses dengan perhitungan jarak c*hebyshev.* Proses clustering menggunakan perhitungan *euclidean* (ED) dan *chebyshev distance* (CD) dilakukan dengan menggunakan program python sebagai alat bantu. Berikut merupakan hasil clustering berdasarkan masing-masing data:

Tabel 1. Hasil *Clustering* Data *Category*

No.	Category	Total 2021	Cluster	
			ED	CD
1.	Sedan Type	5.647	1	0
2.	4x2 Type	503.520	0	1
3.	4x4 Type	4.119	1	0
4.	Double Cabin 4x2/4x4 Type	13.476	1	0
5.	Affordable Energy Saving Cars 4x2 Type	146.520	0	1

Pada tabel diatas dapat dilihat bahwa diperoleh hasil *cluster* yang sama dengan label yang berbeda dari penggunaan *euclidean* (ED) dan *chebyshev distance (CD)* pada *k-medoids* yakni cluster 0 pada *euclidean distance (ED)* terdiri dari kendaraan dengan penjualan tinggi atau laris kemudian untuk cluster 1 terdiri dari 3 kategori kendaraan dengan penjualan rendah atau kurang laris. Sedangkan pada *chebyshev distance* (CD) diperoleh hasil cluster dengan anggota yang sama namun label yang berbeda yakni cluster 0 terdiri dari kendaraan dengan penjualan rendah atau kurang laris dan cluster 1 terdiri kendaraan dengan penjualan tinggi atau laris.

Tabel 2. Hasil *Clustering* Data *Brand*

No.	Brand	Total 2021	Cluster	
			ED	CD
1.	Honda	91112	0	1
2.	Hyundai HMID	3005	0	1
3.	Mercedes Benz PC	2096	1	0
4.	Toyota	293249	0	1
5.	Audi	38	1	0
6.	BMW	2389	0	1
7.	Lexus	972	1	0
8.	Mazda	3392	0	1
9.	Daihatsu	123541	0	1
10.	DFSK	665	1	0
11.	Hyundai HIM	148	1	0
12.	KIA	2767	0	1
13.	Morris Garage	1075	1	0
4.	Mini	657	1	0
15.	Mitsubishi Motors	79491	0	1
16.	Nissan	3177	0	1
17.	Renault	46	1	0
18.	Suzuki	38939	0	1
19.	Volkswagen	380	1	0
20.	Wuling	25564	0	1
21.	Isuzu	304	1	0
22.	Peugeot	265	1	0

Diperoleh hasil *cluster* yang sama dengan label yang berbeda dari penggunaan *euclidean (ED)* dan *chebyshev distance* (CD) pada *k-medoids*. Masing-

masing cluster pada kedua perhitungan jarak tersebut terdiri dari 11 brand kendaraan. *Cluster* 0 pada *euclidean distance* (ED) terdiri dari kendaraan dengan penjualan tinggi atau laris *cluster* terdiri dari kendaraan dengan penjualan rendah atau kurang laris. Sedangkan pada *chebyshev distance* (CD) di peroleh anggota *cluster* yang sama namun dengan label yang berbeda dimana *cluster* 0 terdiri dari kendaraan dengan penjualan rendah atau kurang laris dan *cluster* 1 terdiri dari kendaraan dengan penjualan tinggi atau laris.

Tabel 3. Hasil *Clustering* Data *Cc*

No.	CC	Total 2021	Cluster	
			ED	CD
1.	≤ 1200	19.272	1	1
2.	1201-1500	345.673	0	0
3.	1501-2000	27.192	1	1
4.	2001-3000	100.794	0	1
5.	≥ 3001	351	1	1

Pada tabel diatas dapat dilihat bahwa hasil cluster yang diperoleh dengan menggunakan *euclidean (ED)* dan *chebyshev distance (CD)* pada k-medoids adalah berbeda. *Euclidean distance (ED)* membentuk cluster 0 dengan penjualan yang tinggi atau laris dan *cluster* 1 dengan penjualan rendah atau kurang laris. Sedangkan pada *chebyshev distance (CD)* diperoleh cluster 0 dengan penjualan tinggi dan cluster 1 dengan penjualan rendah atau kurang laris.

Tabel 4. Hasil *Clustering* Data *Trans*

Trans	Total 2021	Cluster	
		ED	CD
AT	298.189	0	0
MT	313.408	1	1
CVT	61.685	0	0

Pada tabel diatas dapat dilihat bahwa *clustering* pada *k-medoids* dengan menggunakan *euclidean (ED)* dan *chebyshev distance (CD)* diperoleh hasil yang sama. *Cluster* 0 terdiri dari kendaraan dengan penjualan kurang laris sedangkan cluster 1 terdiri dari kendaraan dengan penjualan tinggi.

4.3. Validasi Silhouette Coefficient

Berdasarkan dari hasil *clustering* dari beberapa data yang berbeda dengan jumlah data yang bervariasi, diketahui bahwa jarak nilai data pada medoid mempengaruhi hasil cluster. Pada hasil yang telah di peroleh *medoid* yang digunakan pada masing-masing data memiliki jarak nilai data yang cukup jauh sehingga diperoleh hasil *cluster* yang lebih akurat.

Tabel 5. Hasil Validasi *Silhouette Coefficient*

Atribut Data	SCI	
	ED	CD
Category	0.6308	0.6308
Brand	0.7670	0.7670
CC	0.6266	0.5541
Trans	0.1450	0.1450

Berdasarkan hasil evaluasi jarak dengan *Silhouette Coefficient Index (SCI)* dapat dilihat bahwa hasil cluster dengan menggunakan *euclidean* maupun *chebyshev distance* memiliki nilai *SCI* yang sama baik dimana hasil validasi memiliki nilai yang mendekati 1. Kemudian untuk data *Trans* diperoleh nilai SI yang lebih rendah yakni 0.1450 untuk penggunaan *euclidean* dan *chebyshev distance* hal ini di sebab oleh jumlah data yang terlalu sedikit.

5. KESIMPULAN DAN SARAN

Berdasarkan dari hasil penelitian terkait implementasi *euclidean* dan *chebyshev distance* pada algoritma *k-medoids clustering* dengan data wholesales gaikindo 2021 dapat ditarik kesimpulan bahwa masing-masing dari *distance measure* yang digunakan pada *k-medoids clustering* dengan data yang berbeda mampu menghasilkan *cluster* dengan hasil validasi *SI* mendakati nilai 1. Hasil perbandingan validasi yang diperoleh dengan menggunakan *silhouette coefficient* berdasarkan hasil *cluster* dari beberapa data mengguanakan *euclidean* dan *chebyshev distance* pada *k-medoids clustering* menunjukan bahwa penggunaan *euclidean distance* menghasilkan *cluster* yang lebih optimal. Untuk penelitian selanjutnya disarankan *clustering* dengan menggunakan metode perhitungan jarak yang lain untuk mengetahui hasil *cluster* yang terbaik dari *distance measure* yang lain serta lebih teliti dan hati-hati dalam melakukan pemrosesan data jumlah besar.

DAFTAR PUSTAKA

[1] M. A. Nahdliyah, T. Widiharih, and A. Prahutama, "METODE k-MEDOIDS CLUSTERING DENGAN VALIDASI SILHOUETTE INDEX DAN C-INDEX (Studi Kasus Jumlah Kriminalitas Kabupaten/Kota di Jawa Tengah Tahun 2018)," *J. Gaussian*, vol. 8, no. 2, pp. 161–170, 2019.

[2] Y. Miftahuddin, S. Umaroh, and F. R. Karim, "Perbandingan Metode Perhitungan Jarak Euclidean, Haversine, Dan Manhattan Dalam Penentuan Posisi Karyawan," *J. Tekno Insentif*, vol. 14, no. 2, pp. 69–77, 2020.

[3] M. Anggara, H. Sujiani, and N. Helfi, "Pemilihan Distance Measure Pada K-Means Clustering Untuk Pengelompokkan Member Di Alvaro Fitness," *J. Sist. dan Teknol. Inf.*, vol. 1, no. 1, pp. 1–6, 2016.

[4] M. Nishom, "Perbandingan Akurasi Euclidean Distance, Minkowski Distance, dan Manhattan Distance pada Algoritma K-Means Clustering berbasis Chi-Square," *J. Inform. J. Pengemb. IT*, vol. 4, no. 1, pp. 20–24, 2019.

[5] J. Han and M. Kamber, *Data Mining Concepts and Techniques - Second Edition*. 1967.

[6] D. Marlina, N. Lina, A. Fernando, and A. Ramadhan, "Implementasi Algoritma K-Medoids dan K-Means untuk Pengelompokkan Wilayah Sebaran Cacat pada Anak," *J. CoreIT J.*

Has. Penelit. Ilmu Komput. dan Teknol. Inf., vol. 4, no. 2, p. 64, 2018.

[7] Creswell J.W, *Research Design Qualitative, Quantitative and Mixed Methods Approaches*, vol. 71, no. 4. 2014.

[8] W. E. Nurjanah, R. S. Perdana, and M. A. Fauzi, "Analisis Sentimen Terhadap Tayangan Televisi Berdasarkan Opini Masyarakat pada Media Sosial Twitter menggunakan Metode K-Nearest Neighbor dan Pembobotan Jumlah Retweet," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 1, no. 12, pp. 1750–1757, 2017.

[9] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques - Third Edition*. 2012.