

[Click here and write your Article Category](#)

Clustering Menggunakan Metode K-Medoids dengan pendekatan Manhattan Distance

Novriadi Antonius Siagian¹, Alex Rikki², Pandi Barita Nauli Simangunsong³

^{1,2,3}Fakultas Ilmu Komputer, Universitas Katolik Santo Thomas, Jalan Setia Budi No.479 F, Tanjung Sari Medan, Indonesia.

ARTICLE INFORMATION

Received: September 20, 2024

Revised: Oktober 10, 2024

Available online: Oktober 30, 2024

KEYWORDS

K-Medoids, Silhouette Score, Manhattan Distance

CORRESPONDENCE

Phone: +6285373711010

E-mail: novriadi.antonius95@gmail.com

ABSTRACT

Metode K-Medoids dengan pendekatan Manhattan Distance digunakan dalam proses clustering untuk menentukan jumlah kelompok yang optimal. Berdasarkan evaluasi menggunakan Silhouette Score, hasil menunjukkan bahwa pemilihan jumlah cluster $K=2$ dan $K=3$ memberikan performa yang lebih baik dibandingkan dengan jumlah cluster yang lebih besar. Nilai Silhouette Score tertinggi ditemukan pada $K=2$, yaitu 0.761148, yang mengindikasikan bahwa data dalam cluster memiliki tingkat kemiripan yang tinggi dan terpisah dengan jelas dari cluster lainnya. Sementara itu, pada $K=3$, nilai Silhouette Score mengalami sedikit penurunan menjadi 0.742137, tetapi masih menunjukkan kualitas pengelompokan yang cukup baik. Ketika jumlah cluster bertambah dari $K=4$ hingga $K=10$, nilai Silhouette Score cenderung menurun dan stabil dalam kisaran 0.673 hingga 0.682. Hal ini menunjukkan bahwa menambahkan lebih banyak cluster tidak selalu meningkatkan kualitas pengelompokan. Dari hasil analisis ini, dapat disimpulkan bahwa metode K-Medoids dengan Manhattan Distance bekerja optimal ketika jumlah cluster ditetapkan pada $K=2$ atau $K=3$. Dengan jumlah ini, cluster yang terbentuk lebih jelas dan terdefinisi dengan baik.

PENDAHULUAN

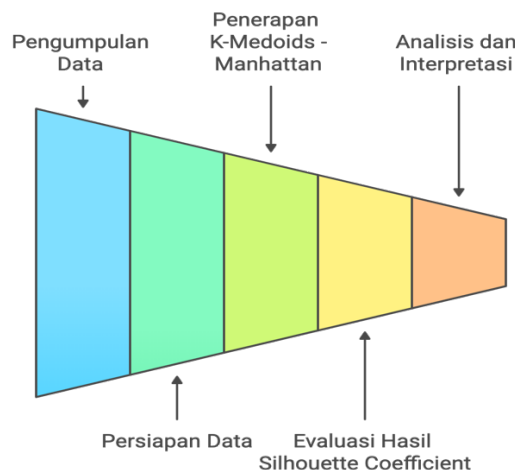
Perkembangan ilmu pengetahuan semakin meningkatkan kompleksitas pengambilan keputusan. Setiap individu menerima beragam informasi setiap hari, yang membentuk kognisi dan latar belakang pengetahuan mereka secara unik. Dalam mengekspresikan preferensi atau evaluasi, digunakan berbagai format informasi. Dalam konteks ini, dikembangkan tiga ukuran jarak baru: jarak minimal, jarak pusat, dan jarak optimal[1]. Dengan kemajuan pesat industri proses dan kompleksitas proses[2]. Memberikan gambaran bagaimana tingkat penjualan tahunan menjadi acuan yang penting di dalam perusahaan. Sehingga banyak penerapan yang dilakukan oleh perusahaan untuk memberikan pelayanan baik bagi pelanggan. Salah satu nya clustering yang dimana metode pembelajaran tanpa pengawasan, yang telah banyak dipelajari dan diterapkan dalam bidang statistika, pembelajaran mesin, pengenalan pola, dan pemrosesan gambar[3]. Metode statistik yang digunakan untuk mengelompokkan objek-objek yang serupa berdasarkan karakteristik atau fitur-fiturnya. *K-medoids* memilih satu titik data sebagai *medoid* dalam setiap cluster. *Medoid* adalah titik yang paling representatif atau paling mendekati semua titik lain dalam cluster tersebut. Ini berbeda dengan metode lain, seperti *K-means*, yang menggunakan rata-rata (centroid) sebagai pusat cluster[4].

Ukuran evaluasi kualitas clustering internal yang telah dikembangkan, seperti *Silhouette*, *Davies-Bouldin index*, dan *Dunn index*. Di antara ukuran tersebut, *Silhouette* dianggap salah satu yang paling robust[5]. Clustering menggunakan algoritma *K-Medoids*, didapatkan hasil evaluasi *Silhouette Coefficient* terbaik sebesar 0.39415227406014575 dengan jumlah cluster sebanyak 2. Dari total 389 data yang dianalisis, ditemukan 3 atribut yang memiliki rentang nilai yang berbeda antara cluster[6]. *K-medoids* dalam lingkungan berbasis probabilitas ini memperlihatkan kelebihan dalam mengatasi tantangan pengolahan informasi heterogen dan mendukung pengambilan keputusan yang lebih tepat[1]. Penggabungan algoritma

K-medoids dengan *crow search algorithm* dapat secara signifikan meningkatkan kinerja clustering, menjadikannya alat yang lebih efektif untuk pengelompokan data[7]. Pemetaan pola sebaran titik gempa bumi di Kepulauan Mentawai antara tahun 2009 hingga 2019 menggunakan metode *K-Medoids Clustering* terbagi menjadi tiga kluster berdasarkan kedalaman dan kekuatan gempa[8]. Metode *K-Medoids* klasik maupun *Quantum K-Medoids* mencapai akurasi pengelompokan yang sama yaitu 91% dikombinasikan dengan perhitungan jarak Manhattan dapat memberikan hasil yang lebih baik dalam mendeteksi diabetes, terutama dengan dataset yang lebih besar [9]. Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk melihat seberapa akurat Metode *K-Medoids* dengan pendekatan jarak Manhattan, dengan evaluasi yang digunakan *Silhouette Coefficient* guna melihat bagaimana cluster yang terbaik di bandingkan untuk banyaknya data yang digunakan dalam penelitian ini.

METODE PENELITIAN

Penelitian ini menggunakan informasi Dataset Penjualan Toyota Motors Stock (Kaggle). Memiliki 8 atribut yaitu Date, Adj Close, Close, High, Low, Open, Volume. Dari dataset penelitian ini menggunakan metode *K-Medoids* dengan pendekatan Jarak Manhattan yang dimana akan menggunakan *Silhouette Coefficient* sebagai evaluasi dari Price High dan Close Price yang nantinya bisa dilihat dari tingkat kluster. Berikut, tahapan penelitian yang dilakukan :



Gambar 1. Tahapan penelitian

1. Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data sekunder. Data didapatkan dari Kaggle. Data yang digunakan merupakan data Toyota Motors Stock. Berisi informasi data stok dan penjualan dari tahun 1980 – 2024.

2. Persiapan data

Pada tahap ini, data akan di lakukan pengamatan, dimana memahami pola dan tren dalam data Toyota Motors Stock. Dilakukan pembersihan data, kemudian dilakukan analisis untuk mengevaluasi hubungan antara berbagai variabel dan menentukan atribut yang akan di lakukan evaluasi.

3. Penerapan K-Medoids Dengan Jarak Manhattan

Penelitian ini menggunakan *K-Medoids* dengan pendekatan Manhattan Distance untuk clustering umumnya digunakan untuk mengelompokkan dataset yang memiliki karakteristik numerik atau data yang memiliki banyak dimensi.

a. Algoritma *K-Medoids*

K-Medoids adalah algoritma clustering yang mirip dengan *K-Means*, namun lebih kuat untuk dataset besar. Tujuannya adalah membagi data ke dalam K cluster dengan meminimalkan jarak antara titik data dan pusat cluster. Perbedaananya, *K-Means* menggunakan centroid (rata-rata koordinat titik), sementara *K-Medoids* menggunakan titik data aktual yang meminimalkan jarak, yang disebut medoid, sebagai pusat cluster[10]. Medoid dapat merepresentasikan pusat cluster yang tepat karena ketahanannya terhadap outlier dan noise[7]. Algoritma *K-Medoids* meminimalkan suatu fungsi

objektif yang dikenal sebagai fungsi kesalahan absolut. Fungsi ini dirumuskan dalam persamaan berikut.

$$\text{absolute error } (E) = \sum_{j=0}^k \sum_{p \in c_j}^n |p - ob_j| \quad [7]$$

Di mana E mewakili jumlah kesalahan absolut, p adalah titik data yang mewakili sebuah objek dalam cluster (C_j), dan ob_j adalah objek representatif dari C_j . Algoritma ini akan mengulang proses tersebut hingga objek representatif menjadi medoid, yaitu objek yang terletak di pusat cluster. Berdasarkan hal tersebut, algoritma *K-Medoids* dikembangkan untuk mengelompokkan n objek ke dalam k cluster.

b. *Manhattan Distance*

Jarak *Manhattan* dihitung dengan menjumlahkan selisih absolut antara dua vektor dalam ruang fitur bilangan bulat. Jarak ini lebih efektif untuk vektor yang menggambarkan objek pada kisi-kisi seragam, seperti papan catur atau blok kota, dan lebih tahan terhadap *outliers* dibandingkan jarak Euclidean. Selain itu, perhitungannya lebih cepat [11].

$$d(x, y) = \sum_{n=1}^k |x_n - y_n| \quad [12]$$

- $x=(x_1, x_2, \dots, x_k)$ dan $y=(y_1, y_2, \dots, y_k)$ adalah dua titik dalam ruang vektor berdimensi k .
- $|x_n - y_n|$ adalah nilai mutlak selisih antara komponen untuk setiap dimensi n
- Jumlahkan nilai mutlak ini untuk seluruh dimensi k untuk mendapatkan jarak total antara x dan y

4. Evaluasi *Silhouette Coefficient*

Silhouette adalah ukuran yang populer untuk mengevaluasi validitas pengelompokan, dan berkinerja sangat baik, memilih jumlah cluster secara performansi bahkan pada set data yang lebih besar [5]. Koefisien ini menghasilkan evaluasi kluster yang komprehensif [13].

$$\text{Silhouette Coefficient} = \sum_{i=1}^n S(x_i) \quad [12]$$

- n = jumlah total titik data dalam dataset.
- $S(x_i)$ = *Silhouette Score* untuk setiap titik data x_i

Koefisien *Silhouette* yang lebih tinggi mengindikasikan kesamaan yang lebih besar antar titik dalam kluster yang sama. Nilai rata-rata ini digunakan untuk menilai kualitas clustering, dengan nilai tinggi menunjukkan hasil clustering yang baik, sementara nilai rendah mengindikasikan potensi masalah pada jumlah kluster.

5. Analisis dan Interpretasi

Proses mengidentifikasi pola dan hubungan dalam data, seperti menilai kualitas clustering dan memeriksa distribusi data dan memberikan hasil analisis untuk memahami makna dan implikasi dari data, serta mengambil keputusan atau tindakan berdasarkan temuan tersebut. Hasil clustering menggunakan *K-Medoids* dengan *Manhattan Distance* menunjukkan bahwa metode ini efektif dalam mengelompokkan data dengan lebih stabil, terutama dalam menghadapi outlier. Evaluasi *Silhouette Score* yang diperoleh menunjukkan kluster yang cukup baik dari pendekatan jarak *Manhattan*.

HASIL DAN PEMBAHASAN

Dataset ini berisi data harga saham Toyota Motors dengan kolom yang mencakup Date, Adj Close, Close, High, Low, Open, dan Volume. Data mencakup periode dari 1980 hingga 2024.

Tabel 1. Dataset (Kaggle)

	Date	Adj Close	Close	High	Low	Open	Volume
0	1980-03-17	2.296798	3.291227	3.344743	3.291227	0.000000	41109
1	1980-03-18	2.306134	3.304606	3.358122	3.304606	0.000000	9343
2	1980-03-19	2.306134	3.304606	3.304606	3.304606	3.304606	0
3	1980-03-20	2.306134	3.304606	3.358122	3.304606	0.000000	10277
4	1980-03-21	2.362154	3.384880	3.438396	3.384880	0.000000	8409
...
11286	2024-12-20	178.169998	178.169998	179.919998	175.839996	175.839996	425700
11287	2024-12-23	180.449997	180.449997	180.619995	177.970001	179.119995	422700
11288	2024-12-24	181.429993	181.429993	181.720001	180.830002	181.000000	168600
11289	2024-12-26	197.360001	197.360001	198.000000	193.130005	195.970001	1281200
11290	2024-12-27	199.520004	199.520004	201.000000	198.179993	200.360001	779500

Dari dataset yang digunakan mencakup atribut-atribut penting dari saham Toyota Motors Stock untuk dilakukan clustering, yaitu Close, High, Low, Open, dan Volume. Close mewakili harga penutupan saham pada setiap hari perdagangan, sementara High dan Low menunjukkan harga tertinggi dan terendah yang tercatat selama sesi perdagangan. Open adalah harga pembukaan saham pada hari tersebut, dan Volume menggambarkan jumlah saham yang diperdagangkan.

Tabel 2. Inisialisasi Medoid Secara Acak

	Close	High	Low	Open	Volume
0	83.50	84.125000	83.500	83.500000	17500.0
1	49.52	49.950001	49.410	49.650002	75500.0
2	53.00	53.500000	52.625	52.625000	49400.0

Memulai proses clustering menggunakan algoritma K-Medoids, langkah pertama adalah menginisialisasi medoid secara acak. Dalam hal ini, medoid dipilih dari dataset yang ada dengan menggunakan fungsi *np.random.choice()*, yang memungkinkan pemilihan acak sejumlah medoid sesuai jumlah cluster yang diinginkan (dalam hal ini, 3 cluster). Pemilihan medoid yang tepat sangat penting untuk kualitas pengelompokan.

Selanjutnya dilakukan menghitung jarak Manhattan:

$$d(x, y) = \sum_{n=1}^k |x_n - y_n| \quad [12]$$

1. Jarak antara data pertama dan Medoid pertama

$$d = |3.2912-83.50| + |3.3447-84.125| + |3.2912-83.500| + |0.0000-83.500| + |41109-17500|$$

$$d = 80.2088 + 80.7803 + 80.2088 + 83.5000 + 23609 = 31863.6979$$

2. Jarak antara data pertama dan *Medoid* kedua

$$d = |3.2912-49.52| + |3.3447-49.950| + |3.2912-49.410| + |0.0000-49.650| + |41109-75500|$$

$$d = 46.2288 + 46.6053 + 46.1188 + 49.6500 + 34391 = 34089$$

3. Jarak antara data pertama dan *Medoid* ketiga

$$d = |3.2912-53.00| + |3.3447-53.500| + |3.2912-52.625| + |0.0000-52.625| + |41109-49400|$$

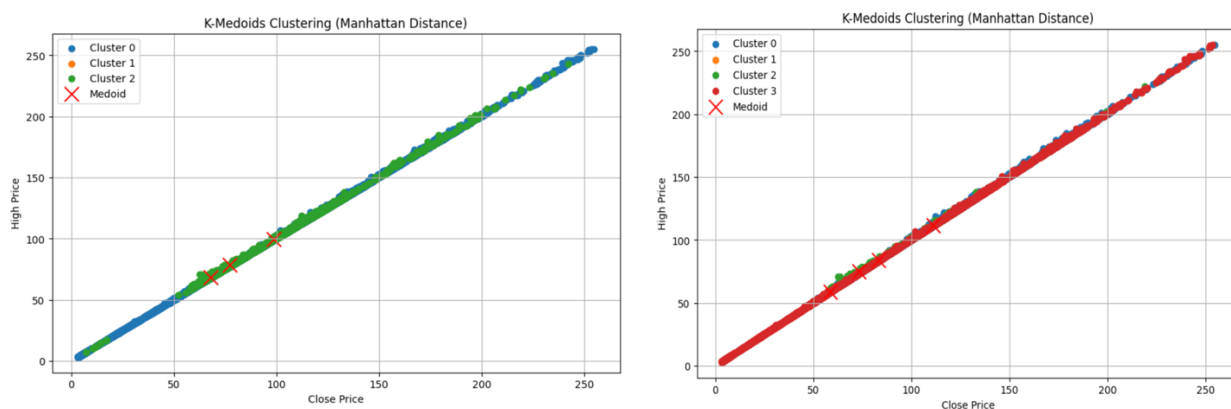
$$d = 49.7088 + 50.1553 + 49.3338 + 52.6250 + 8291 = 12493.823$$

Langkah perhitungan jarak Manhattan akan berlanjut melalui iterasi yang di capai, di mana setiap data dihitung jaraknya ke setiap medoid. Kemudian, berdasarkan jarak terkecil, data akan dikelompokkan ke medoid terdekat. Proses ini diulang hingga posisi medoid stabil, menciptakan pembagian kluster yang optimal.

Tabel 3. Hasil Iterasi dan Cluster

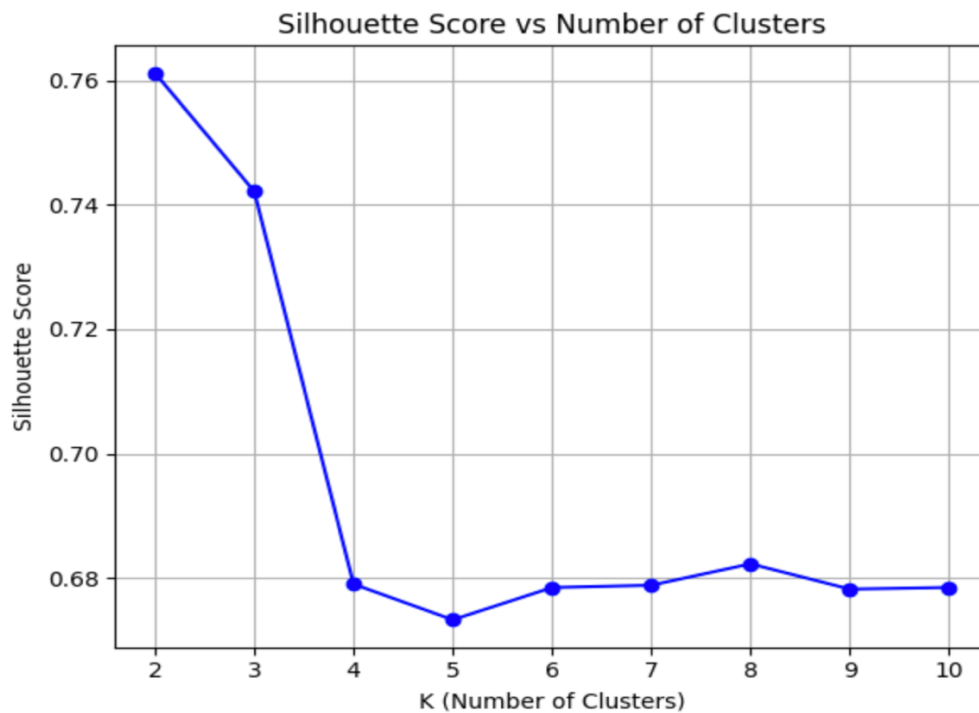
Index	Close	High	Low	Open	Volume	Iteration	Cluster
0	83.5	84.125	83.5	83.5	17500.0	1	0
1	49.52000046	49.95000076	49.40999985	49.65000153	75500.0	1	1
2	53.0	53.5	52.625	52.625	49400.0	1	2
3	31.34425502009605	31.536708679110546	31.234132601238453	23.270735413043706	11345.156750572083	2	0
4	106.093342905443	106.70479131164261	105.43756311836556	105.22910138978723	385065.5225923483	2	1
5	38.83304646433722	39.0732182706966	38.615123443427066	32.068030968249715	45642.2337222871	2	2
6	30.946839979529777	31.136911422903744	30.840957420684667	22.7883251274498	10133.466083150985	3	0
7	110.92217529577725	111.63860318810795	110.15356804397169	110.7498989416214	524607.5130795131	3	1
8	77.79604310617512	78.15986008676967	77.42172778941612	74.14861201678026	107489.44950256255	3	2
9	32.50929838632263	32.70929634822337	32.38250382559877	24.650390705622414	16460.28701374105	4	0
10	104.85776812849967	105.61175458892764	104.0409595519808	104.76536164591195	656248.3142746013	4	1
11	105.83717422332761	106.33894292713218	105.30537638226368	104.2961929509907	183356.69546861807	4	2
12	34.7429752957789	34.95035539660584	34.60080810722584	27.164153336759718	22609.461953682745	5	0
13	99.67818989272953	100.47984565684135	98.80132980614218	99.60012762456181	800403.2698781196	5	1
14	115.64180871298137	116.21238023269488	115.03596622023909	114.94638813394238	239160.8062745098	5	2
15	38.52138102150979	38.737671092029345	38.3627878494385	31.26628337114729	28683.82460136674	6	0
16	94.67130944244256	95.51387556532774	93.739477335189341	94.59945896633621	938076.2868852459	6	1
17	117.54215066468001	118.1529595179157	116.8967742579251	117.12376295145474	281015.8476433121	6	2
18	41.98984759849807	42.21852828198033	41.815024942067005	35.045668137863686	34758.84881515712	7	0
19	91.266063070308	92.1212411500989	90.26420379388239	91.13570949995606	1066810.7623762377	7	1
20	117.74946136134324	118.39026367286563	117.08133671865438	117.46165800411049	315770.0827763496	7	2
21	45.6324806602739	45.773137008934135	45.339790916560325	38.8763181839485	40978.06013493693	8	0
22	88.70914058721281	89.60946726280115	87.6490145588346	88.55383744703078	1181472.1540616245	8	1
23	117.00600348214313	117.66337104201928	116.32125888379535	116.78598292288366	346041.90635807393	8	2
24	48.67938094868068	48.93373632191196	48.47092648803441	42.29088048947326	47006.56715369486	9	0

Menggunakan 3 Cluster Medoids mendapatkan 50 iterasi, proses K-medoids mengelompokkan data menjadi tiga kluster berdasarkan jarak Manhattan ke medoid terdekat. Menggambarkan kesamaan pola antara harga penutupan (close price) dan harga tertinggi (high price) saham, dengan medoid yang mencerminkan pusat cluster. Proses ini memastikan bahwa data dikelompokkan secara optimal, dengan setiap iterasi menyempurnakan pembagian kluster.



Gambar 2. Hasil Visualisasi K-3 dan K-4

Kluster-kluster ini tidak didasarkan pada kesamaan harga saham secara keseluruhan, melainkan pada hubungan linier antara harga penutupan dan harga tertinggi, seperti yang terlihat dalam gambar di atas, yang menunjukkan pola harga yang konsisten di setiap kluster.

Gambar 3. Hasil Evaluasi *Silhouette Score*

Silhouette Score dan jumlah kluster (dari 2 hingga 10 kluster). Tampak bahwa nilai Silhouette Score tertinggi terjadi pada 2 kluster, dan seiring dengan meningkatnya jumlah kluster, skor tersebut cenderung menurun. Skor mulai stabil setelah 5 kluster, menunjukkan bahwa menambah jumlah kluster lebih dari 4 tidak memberikan perbaikan signifikan pada kualitas clustering.

Tabel 4. Hasil Nilai *Silhouette Score*

	Number of Clusters (K)	Silhouette Score
0	2	0.761148
1	3	0.742137
2	4	0.679049
3	5	0.673310
4	6	0.678496
5	7	0.678866
6	8	0.682301
7	9	0.678240
8	10	0.678503

Tabel di atas menunjukkan nilai Silhouette Score untuk berbagai jumlah kluster (dari 2 hingga 10). Skor tertinggi terjadi pada 2 kluster dengan nilai 0.7611, yang menurun secara bertahap saat jumlah kluster meningkat.

KESIMPULAN

Penelitian ini menggunakan metode K-Medoids dengan pendekatan Manhattan Distance untuk clustering data, yang dievaluasi dengan Silhouette Score. Berdasarkan tabel, hasil menunjukkan bahwa jumlah kluster optimal adalah 2, dengan Silhouette Score tertinggi 0.7611. Nilai ini menurun seiring dengan bertambahnya jumlah kluster, menunjukkan bahwa penambahan kluster lebih dari dua tidak meningkatkan kualitas clustering secara signifikan. Skor stabil pada kluster lebih dari lima, dengan nilai sekitar 0.68. Pendekatan ini terbukti efektif dalam menghasilkan kluster yang jelas, dengan pemisahan yang baik antar kluster, dan memberikan hasil yang memadai untuk clustering. Penelitian lanjutan dapat dilakukan dengan menguji metode K-Medoids pada dataset yang tidak seimbang untuk mengevaluasi robusta dan performa clustering dalam kondisi tersebut. Keterbatasan dari penelitian ini terletak pada asumsi bahwa data memiliki distribusi yang seimbang, padahal dalam kasus dunia nyata, dataset sering kali tidak seimbang, yang bisa memengaruhi kualitas klusterisasi.

REFERENSI

- [1] Y. He, Z. Xu, and N. Liu, "Research on K-medoids Algorithm with Probabilistic-based Expressions and Its Applications," *Applied Intelligence*, vol. 52, no. 10, pp. 12016–12033, Aug. 2022, doi: 10.1007/s10489-021-02937-8.
- [2] Q. X. Zhu, X. W. Wang, N. Zhang, Y. Xu, and Y. L. He, "Novel K-Medoids Based SMOTE Integrated With Locality Preserving Projections for Fault Diagnosis," *IEEE Trans Instrum Meas*, vol. 71, 2022, doi: 10.1109/TIM.2022.3218551.
- [3] D. Cheng, Q. Zhu, J. Huang, Q. Wu, and L. Yang, "Clustering with Local Density Peaks-Based Minimum Spanning Tree," *IEEE Trans Knowl Data Eng*, vol. 33, no. 2, pp. 374–387, Feb. 2021, doi: 10.1109/TKDE.2019.2930056.
- [4] D. Wang, C. Jin, B. Xiao, Z. Li, and X. He, "Proof-of-Activity Consensus Algorithm Based on K-Medoids Clustering," *Big Data Research*, vol. 26, Nov. 2021, doi: 10.1016/j.bdr.2021.100266.
- [5] L. Lenssen and E. Schubert, "Clustering by Direct Optimization of the Medoid Silhouette," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2022, pp. 190–204. doi: 10.1007/978-3-031-17849-8_15.
- [6] S. Bahri, D. Marisa Midyanti, and P. Korespondensi, "PENERAPAN METODE K-MEDOIDS UNTUK PENGELOMPOKAN MAHASISWA BERPOTENSI DROP OUT APPLICATION OF K-MEDOIDS METHOD FOR DROPOUT POTENTIAL STUDENT GROUPING," vol. 10, no. 1, pp. 165–172, 2023, doi: 10.25126/jtiik.2023106643.
- [7] N. Sureja, B. Chawda, and A. Vasant, "An improved K-medoids clustering approach based on the crow search algorithm," *Journal of Computational Mathematics and Data Science*, vol. 3, Jun. 2022, doi: 10.1016/j.jcmds.2022.100034.
- [8] Y. Setiawan, S. Aziz Suprianto, A. Wijanarko, D. Setyo Rini, and M. Yusa, "PEMETAAN KELOMPOK SEBARAN TITIK GEMPA BUMI MENTAWAI DENGAN METODE K-MEDOIDS CLUSTERING," 2022.
- [9] Solikhun, M. Rahmansyah Siregar, L. Pujiastuti, and M. Wahyudi, "Manhattan Distance-based K-Medoids Clustering Improvement for Diagnosing Diabetic Disease," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 6, pp. 710–718, Dec. 2024, doi: 10.29207/resti.v8i6.5894.
- [10] A. Sobrinho Campolina Martins, L. Ramos de Araujo, and D. Rosana Ribeiro Penido, "K-Medoids clustering applications for high-dimensionality multiphase probabilistic power flow," *International Journal of Electrical Power and Energy Systems*, vol. 157, Jun. 2024, doi: 10.1016/j.ijepes.2024.109861.
- [11] Z. Wu, T. Song, and Y. Zhang, "Quantum k-means algorithm based on Manhattan distance," *Quantum Inf Process*, vol. 21, no. 1, Jan. 2022, doi: 10.1007/s11128-021-03384-7.
- [12] M. Raeisi and A. B. Sesay, "A Distance Metric for Uneven Clusters of Unsupervised K-Means Clustering Algorithm," *IEEE Access*, vol. 10, pp. 86286–86297, 2022, doi: 10.1109/ACCESS.2022.3198992.
- [13] H. Lai, T. Huang, B. L. Lu, S. Zhang, and R. Xiaog, "Silhouette coefficient-based weighting k-means algorithm," *Neural Comput Appl*, 2024, doi: 10.1007/s00521-024-10706-0.