ISSN : 2527-5836
e-ISSN : 2528-0074

Vol. 8 No. 1, Januari 2023

JISKa

Jurnal Informatika Sunan Kalijaga

Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
UIN Sunan Kalijaga Yogyakarta

Tim Pengelola JISKa Edisi Januari 2023

Mitra Bestari (*Reviewer*)

Reviewer Internal:

1. Mandahadi Kusuma, M.Eng. (UIN Sunan Kalijaga Yogyakarta, Indonesia)
2. Maria Ulfa Siregar, Ph.D. (UIN Sunan Kalijaga Yogyakarta, Indonesia)

Reviewer Eksternal (Mitra Bestari):

1. Ahmad Fathan Hidayatullah, M.Cs. (Universitas Islam Indonesia Yogyakarta, Indonesia)
2. Alam Rahmatulloh, M.T. (Universitas Siliwangi Tasikmalaya, Indonesia)
3. Alfan Farizki Wicaksono, Ph.D. (Universitas Indonesia, Indonesia)
4. Ardiansyah Musa Efendi, Ph.D. (Chonnam National University, Korea Selatan)
5. Dr. Aris Puji Widodo, M.T. (Universitas Diponegoro, Indonesia)
6. Dr. Cahyo Crysdian (UIN Maulana Malik Ibrahim Malang, Indonesia)
7. Dr. Enny Itje Sela (Universitas Teknologi Yogyakarta, Indonesia)
8. Dr.Eng. Ganjar Alfian (Universitas Gadjah Mada, Indonesia)
9. Muhammad Habibi, M.Cs. (Universitas Jenderal Achmad Yani Yogyakarta, Indonesia)
10. Muhammad Rifqi Maarif, M.Eng. (Universitas Jenderal Achmad Yani Yogyakarta, Indonesia)
11. Dr.Eng. M. Muhammad Syafrudin (Sejong University, Korea Selatan)
12. Dr.Eng. M. Alex Syaekhoni (Dongguk University Seoul, Korea Selatan)
13. Norma Latif Fitriyani, M.Sc. (Sejong University, Korea Selatan)
14. Nur Aini Rakhmawati, Ph.D. (Institut Teknologi Sepuluh November, Indonesia)
15. Prof. Dr. Hj. Okfalisa, S.T., M.Sc. (UIN Sultan Syarif Kasim Riau, Indonesia)
16. Oman Somantri, M.Kom. (Politeknik Negeri Cilacap, Indonesia)
17. Puji Winar Cahyo, M.Cs. (Universitas Jenderal Achmad Yani Yogyakarta, Indonesia)
18. Rischan Mafrur, M.Eng. (The University of Queensland Brisbane, Australia)
19. Dr.Eng. Sunu Wibirama, M.Eng. (Universitas Gadjah Mada, Indonesia)
20. Yudistira Dwi Wardhana Asnar, Ph.D. (Institut Teknologi Bandung, Indonesia)

ISSN : 2527-5836

e-ISSN: 2528-0074

JISKa

Vol. 8, No. 1, JANUARI 2023

DAFTAR ISI

Pengelompokan Obyek Wisata Potensial dengan *Self Organizing Maps* (SOM) dan *Sum Additive Weighting* (SAW) **1-9**

Indra Dharma Wijaya, Muhammad Afif Hendrawan, Nurcahya Nania Anabela

Penerapan *Data Mining* dengan Metode Regresi Linear untuk Memprediksi Data Nilai Hasil Ujian Menggunakan RapidMiner **10-21**

Muhammad Sholeh, Erna Kumalasari Nurnawati, Uning Lestari

Perbandingan Waktu Respon Aplikasi *Database* NoSQL Elasticsearch dan MongoDB pada Pengujian Operasi CRUD **22-35**

Theresia Liana Sinaga, Novrido Charibaldi, Nur Heri Cahyana

Penentuan Kelayakan Masyarakat Miskin Penerima Bantuan Menggunakan Metode Naïve Bayes (Studi Kasus: Kabupaten Penajam Paser Utara) **36-49**

Nur Madia, Anindita Septiarini, Heliza Rahmania Hatta, Hamdani Hamdani, Masna Wati

Analisis Perbandingan Metode Pendukung Keputusan Pemilihan Kos Mahasiswa di Pontianak **50-65**

Noerul Hanin, David Jordy Dhandio, Della Zaria

Penerapan Algoritma K-Means untuk Klasterisasi Penduduk Miskin pada Kota Pagar Alam **66-77**

Febriansyah Febriansyah, Siti Muntari

Analisa Deteksi dan Pengenalan Wajah pada Citra dengan Permasalahan Visual **78-89**

Verry Noval Kristanto, Imam Riadi, Yudi Prayudi

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023, Pp. 1 – 9
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Pengelompokan Obyek Wisata Potensial dengan *Self Organizing Maps* (SOM) dan *Sum Additive Weighting* (SAW)

Indra Dharma Wijaya [(1)], Muhammad Afif Hendrawan [(2)*], Nurcahya Nania Anabela [(3)]
Teknologi Informasi, Politeknik Negeri Malang, Malang
e-mail : {indra.dharma,afif.hendrawan}@polinema.ac.id, nania.anabela22@gmail.com.
* Penulis korespondensi.
Artikel ini diajukan 28 April 2022, direvisi 12 Agustus 2022, diterima 15 Agustus 2022, dan dipublikasikan 30 Januari 2023.

Abstract

Probolinggo Regency is an area in East Java that has tourism potential. The condition is seen from the many tourists visiting various attractions in Probolinggo Regency. To increase the number of tourist visits, it is necessary to develop tourism objects. However, not all attractions in Probolinggo Regency can be developed at the same time. This is due to budget limitations for tourism development. Therefore, it is necessary to have a grouping of attractions according to the priority level of development. In this study, researchers utilized Self Organizing Maps (SOM) and Sum Additive Weighing (SAW) methods to group attractions based on their development priority levels. SOM is used to determine groups of tourist objects based on the parameters of the number of domestic tourists, the number of foreign tourists, infrastructure, and the number of attractions. Furthermore, SAW is used to find out which group has the highest priority among other groups based on these parameters. To measure the quality of the resulting group, researchers used the value of the silhouette coefficient. Results from the grouping process resulted in three groups. Group C1 consists of 4 attractions, group C2 consists of 20 attractions, and group C3 consists of 10 attractions. The value of the silhouette coefficient also holds a good value, especially in group 1, which is 0.75006. Furthermore, based on the ranking of groups by the SAW method, the C1 group is the group of tourist attractions with the highest priority for development.

Keywords: Tourism, Data Mining, Clustering, SOM, SAW

Abstrak

Kabupaten Probolinggo merupakan daerah di Jawa Timur yang memiliki potensi pariwisata. Kondisi tersebut dilihat dari banyaknya wisatawan yang berkunjung ke berbagai objek wisata di Kabupaten Probolinggo. Untuk meningkatkan jumlah kunjungan wisatawan, maka perlu dilakukan pengembangan obyek pariwisata. Akan tetapi, tidak semua objek wisata di Kabupaten Probolinggo dapat dikembangkan dalam waktu yang bersamaan. Hal ini dikarenakan adanya keterbatasan anggaran untuk pengembangan pariwisata. Oleh karena itu, perlu adanya pengelompokan objek wisata sesuai dengan tingkat prioritas pengembangannya. Dalam penelitian ini, peneliti memanfaatkan metode *Self Organizing Maps* (SOM) dan *Sum Additive Weighting* (SAW) untuk mengelompokkan objek wisata berdasarkan tingkat prioritas pengembangannya. SOM digunakan untuk mengetahui kelompok-kelompok obyek wisata berdasarkan parameter jumlah wisatawan domestik, jumlah wisatawan mancanegara, sarana prasarana, dan jumlah daya tarik. Selanjutnya, SAW digunakan untuk mengetahui kelompok mana yang memiliki prioritas tertinggi di antara kelompok yang lain. Untuk mengukur kualitas kelompok yang dihasilkan, peneliti menggunakan nilai koefisien *silhouette*. Hasil dari proses pengelompokan, menghasilkan tiga kelompok. Kelompok C1 terdiri dari 4 objek wisata, kelompok C2 terdiri dari 20 objek wisata, dan kelompok C3 terdiri dari 10 objek wisata. Nilai koefisien *silhouette* juga menjukkan nilai yang baik, khususnya pada kelompok 1, yaitu sebesar 0,75006. Selanjutnya, berdasarkan pemeringkatan kelompok dengan metode SAW, didapatkan kelompok C1 merupakan kelompok obyek wisata dengan prioritas paling tinggi untuk dilakukan pengembangan.

Kata Kunci: Pariwisata, Penambangan Data, *Clustering*, SOM, SAW

1. PENDAHULUAN

Pariwisata merupakan salah satu sumber devisa negara yang berpotensi dan berperan meningkatkan pertumbuhan ekonomi suatu negara serta mengenalkan alam budaya di berbagai daerahnya (Simamora & Sinaga, 2016). Salah satu daerah yang memiliki potensi pariwisata di Indonesia adalah Kabupaten Probolinggo. Kabupaten Probolinggo merupakan daerah transit, menghubungkan berbagai kota di bagian barat dan timur provinsi Jawa Timur sehingga banyak dilewati pengunjung. Pariwisata yang ada berkembang baik dilihat dari beberapa objek wisata yang dikenal hingga mancanegara. Kondisi ini menjadikan Kabupaten Probolinggo sebagai kota tujuan wisata. Berdasarkan data statistik, tahun 2017 jumlah kunjungan wisata meningkat 43,91%, tahun 2018 meningkat 30%, dan di tahun 2019 meningkat sekitar 13,85% dari tahun sebelumnya (Badan Pusat Statistik, 2020). Hal tersebut membuktikan kunjungan wisata di objek wisata Kabupaten Probolinggo terus meningkat setiap tahunnya. Berbagai upaya dilakukan untuk menjaga dan meningkatkan minat kunjungan wisatawan. Upaya tersebut meliputi pengembangan sarana dan prasarana serta daya tarik tempat wisata. Upaya tersebut dilakukan karena fasilitas disetiap objek wisata perlu ditingkatkan untuk memenuhi kebutuhan wisatawan (Arifin, 2020).

Dalam upaya mengembangkan potensi wisata, Dinas Pemuda, Olahraga, Pariwisata dan Kebudayaan (DISPORAPARBUD) Pemerintah Kabupaten Probolinggo mengalami berbagai kendala. Kendala tersebut salah satunya adalah keterbatasan anggaran. Hal ini disebabkan oleh tugas DISPORAPARBUD yang terbagi atas bidang pemuda, olahraga, pariwisata dan budaya. Tidak semua anggaran dapat dialokasikan pada bidang pariwisata. Oleh karena itu, perlu ditentukan prioritas pengembangan terhadap objek wisata yang akan dikembangkan terlebih dahulu. Langkah ini bertujuan agar objek wisata yang menjadi skala prioritas dapat dimaksimalkan untuk menjadi tolak ukur objek wisata yang lain. Penentuan objek wisata yang menjadi skala prioritas pengembangan tidak mudah dilakukan, mengingat semua objek wisata yang ada di Kabupaten Probolinggo berpotensi untuk berkembang. Sebelumnya DISPORAPARBUD menggunakan cara manual untuk menentukan prioritas pengembangan obyek wisata. Salah satu cara yang dilakukan adalah melihat jumlah pengunjung. Cara tersebut kurang efektif karena penentuannya dinilai kurang merata. Selain itu, tidak ada faktor-faktor penting lain yang menjadi pertimbangan, seperti fasilitas, kondisi infrastruktur jalan, ataupun jenis wisata yang ditawarkan.

Permasalahan tersebut dapat dipecahkan dengan memanfaatkan teknik pengelompokan data (*clustering*). Pengelompokan data adalah proses membagi data ke dalam kelompok-kelompok yang memiliki kemiripan dan dapat dimanfaatkan untuk memecahkan kasus tertentu (Hale, 1981). Cara ini banyak dimanfaatkan dalam bidang, tidak terkecuali pembelajaran mesin (*machine learning*) ataupun pengenalan pola (*pattern recognition*) (Al-Otaibi et al., 2016; Bi et al., 2016; Song et al., 2018). Hal ini dikarenakan, pengelompokan data dapat mengetahui karakteristik kelompok data. Pengetahuan terhadap karakteristik penting untuk menentukan proses selanjutnya dalam mengolah data atau dalam hal pengambilan keputusan.

Saat ini, terdapat banyak penelitian yang memanfaatkan metode pengelompokan data untuk memecahkan permasalahan yang dihadapi. Pada penelitian yang dilakukan Maulida (2018) menggunakan metode K-Means untuk mengelompokkan obyek wisata berdasarkan jumlah kunjungan wisatawan. Hasil penelitian tersebut menunjukkan bahwa, parameter jumlah kunjungan wisatawan dapat menjadi parameter untuk menghasilkan kelompok yang baik dalam konteks permasalahan pariwisata. Selain itu, kelompok yang dihasilkan dalam penelitian tersebut digunakan untuk memberikan saran perbaikan sarana dan prasarana obyek wisata dalam rangka meningkatkan jumlah kunjungan wisatawan. Penelitian lainnya dilakukan oleh Umar, dkk., memanfaatkan pengelompokan data dengan metode *self organizing maps* (SOM) untuk mengelompokkan siswa SMK berdasarkan keterampilan, minat, dan bakat ke dalam kelompok-kelompok penjurusan (Umar et al., 2018). Kelompok yang dihasilkan oleh penelitian tersebut cukup baik akan tetapi tidak ada pengujian keakurasian pengelompokan. Sehingga tidak dapat diketahui tingkat keakuratan kelompok yang dihasilkan. Sedangkan, tingkat keakuratan pengelompokan data juga didasarkan kepada parameter pengelompokan yang digunakan.

Kondisi ini tentunya akan berakibat pada kurang tepatnya sebuah data dikelompokkan pada kelompok tertentu. Pada penelitian yang dilakukan oleh Hartatik & Cahya 2020), SOM terbukti dapat menghasilkan anggota kelompok yang stabil, dilihat dari nilai titik dengan kelompok (*centroid*) yang tidak berubah setiap kali pengujian. Pada pendekatan lainnya, (Muin et al., 2018) membuktikan bahwa SOM memiliki tingkat akurasi cukup baik pada kasus klasifikasi penduduk untuk menentukan prioritas pembangunan berdasarkan daerah, tempat, geografis, serta wilayah.

Pada penelitian ini dilakukan pengelompokan objek wisata dengan menerapkan metode SOM. SOM dipilih karena proses pengelompokkan yang mengedepankan intuisi peletakkan titik tengah yang adaptif sehingga kelompok yang dihasilkan lebih natural dengan kondisi data (Asan & Ercan, 2012; Miljkovic, 2017). Untuk mengetahui tingkat keakurasian pengelompokan data, diperlukan metode validasi yang dapat menilai derajat keakuratan pengelompokan berdasarkan data-data di dalam kelompok tersebut dan antar kelompok yang lain. Salah satu metode yang dapat digunakan adalah koefisien *silhouette* (Wira et al., 2019). Pada penelitian yang dilakukan oleh Wira, dkk., nilai koefisien *silhouette* dapat memvalidasi kekuatan kelompok yang dihasilkan (Wira et al., 2019). Selanjutnya, metode SAW digunakan untuk mengetahui tingkat kepentingan setiap kelompok yang dihasilkan. Metode SAW dapat memberikan rekomendasi berdasarkan parameter-parameter yang telah ditentukan. Sehingga, berdasarkan hasil rekomendasi SAW, dapat diketahui kelompok obyek wisata yang memiliki prioritas lebih tinggi dibandingkan dengan yang lainnya. Hasil pengelompokan dan rekomendasi kelompok obyek wisata diharapkan dapat memberikan gambaran prioritas pengembangan objek wisata, sehingga dapat membantu DISPORAPARBUD dalam menentukan alokasi anggaran dengan sasaran yang tepat.

2. METODE PENELITIAN

Pada penelitian ini proses pengelompokan prioritas objek wisata dibagi menjadi empat tahapan utama yaitu pengumpulan data, pra pengolahan data, pengelompokan data, dan validasi kelompok data.

2.1 Pengumpulan Data

Pengumpulan data dilakukan untuk mengetahui daftar obyek wisata potensial yang ada di Kabupaten Probolinggo dan menggali informasi terkait dengan parameter yang dapat digunakan untuk pengelompokan. Berdasarkan hasil wawancara dengan DISPORAPARBUD didapatkan 34 obyek wisata potensial yang akan dikelompokkan. Selanjutnya, parameter yang akan digunakan pada proses pengelompokan merujuk kepada laporan akhir Review Rencana Induk Pembangunan Kepariwisataan Kabupaten Probolinggo (RIPPARDA) tahun 2019-2034. Berdasarkan RIPPARDA, terdapat empat parameter yang dapat digunakan untuk menentukan kelompok obyek wisata potensial, yaitu, jumlah wisatawan domestik (P1), jumlah wisatawan mancanegara (P2), sarana prasarana obyek wisata (P3), dan daya tarik obyek wisata (P4).

2.2 Pra Pengolahan Data

Tahapan pra pengolahan data terdiri dari dua sub tahapan. Pertama adalah proses transformasi data dan kedua adalah proses normalisasi. Proses transformasi dilakukan untuk mengubah nilai kategorial menjadi nilai numerik agar dapat dilakukan proses pengelompokan. Pada penelitian ini, parameter daya tarik wisata adalah data kategorial sehingga perlu diubah menjadi nilai numerik. Nilai numerik parameter daya tarik didapatkan dari jumlah daya tarik setiap obyek wisata. Tabel 1 merupakan contoh nilai parameter daya tarik obyek wisata. Selanjutnya, dilakukan penyekalaan terdapat nilai jumlah daya tarik. Penyekalaan dilakukan untuk menghindari nilai ekstrem antar obyek wisata pada parameter daya tarik. Tabel 2 digunakan sebagai acuan penyekalaan nilai parameter daya tarik.

Tabel 1 Nilai Daya Tarik

Nama Obyek Wisata	Daya Tarik	Jumlah Daya Tarik
Gunung Agropuro	Trek pendakian terpanjang di Pulau Jawa Banyak Pemandangan Indah Terdapat Peninggalan Sejarah	3

Tabel 2 Pembobotan Nilai Daya Tarik

Jumlah Daya Tarik	Nilai Bobot
1 – 2	1
3 – 4	2
5 – 7	3
8 – 10	4

Proses pra pengolahan data dilanjutkan dengan proses normalisasi. Proses normalisasi dilakukan untuk menormalkan rentang nilai setiap parameter dengan nilai minimal adalah 0 dan maksimal adalah 1. Selain itu, proses normalisasi bertujuan agar setiap parameter miliki skala yang sama sehingga memiliki nilai yang seimbang. Pers. (1) digunakan pada proses normalisasi untuk setiap parameter. Persamaan $norm_i$ merupakan nilai ternomalisasi untuk data ke-i di mana x_i merupakan nilai numerik parameter dan $\max(x)$ merupakan nilai maksimal dari paramater. Contoh hasil normalisasi dapat dilihat pada Tabel 3.

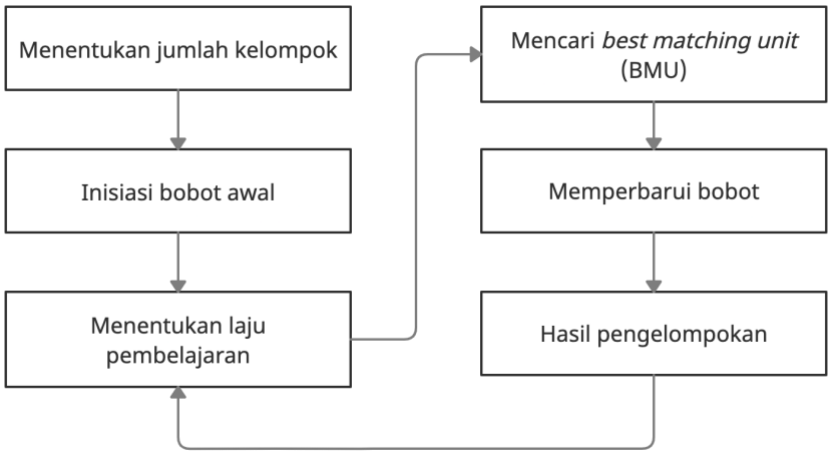
$$norm_i = \frac{x_i}{\max(x)} \qquad (1)$$

Tabel 3 Contoh Hasil Normalisasi Setiap Parameter

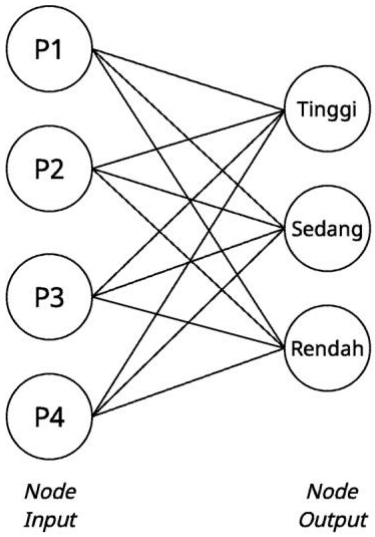
No.	Nama Obyek Wisata	P1	P2	P3	P4
1	Pantai Bentar	0,61741	0,05504	0,45385	1,00000
2	Gunung Bromo	1,00000	1,00000	0,65385	1,00000
3	Tirta Ronggojalu	0,02307	0,00000	0,60000	0,66667
4	Air Terjun Madakaripura	0,16557	0,18447	0,49231	0,66667
5	Ranu Segaran	0,01905	0,00192	0,50769	0,66667
6	Ranu Agung	0,01941	0,00000	0,64615	0,33333
7	Miniatur Ka'bah	0,13878	0,00005	0,73846	0,33333
8	Rafting Sungai Pekalen	0,15496	0,10984	0,71538	0,66667
9	Candi Jabung	0,30272	0,02814	0,63846	0,33333
10	Candi Kedaton	0,02200	0,00526	0,62308	0,33333

2.3 Pengelompokan Data

Tahap selanjutnya dilanjutkan dengan mengelompokkan data. Data pariwisata yang telah dinormalisasi akan dikelompokkan berdasarkan nilai-nilai parameternya menggunakan metode *Self Organizing Maps* (SOM). Tahapan pengelompokan data menggunakan SOM dapat dilihat pada Gambar 1.



Gambar 1 Tahapan SOM



Gambar 2 Arsitektur SOM Penelitian

Tabel 4 Inisiasi Bobot Awal

Kelompok	P1	P2	P3	P4
C1	0,66	0,34	0,58	0,80
C2	0,13	0,01	0,56	0,33
C3	0,07	0,02	0,64	0,56

Pertama ditentukan jumlah kelompok yang akan digunakan. Pada penelitian ini digunakan tiga buah kelompok, yaitu C1, C2, dan C3. Ketiga kelompok ini akan digunakan untuk mewakili prioritas pengembangan obyek wisata, yaitu prioritas tinggi, sedang, dan rendah. Dikarenakan jumlah kelompok adalah tiga buah, maka jumlah *node* keluaran SOM adalah tiga. Gambar 2 merupakan arsitektur SOM yang digunakan pada penelitian ini. Selanjutnya adalah tahap inisiasi bobot awal. Bobot yang digunakan dalam penelitian ini adalah bobot dengan rentang nilai 0 sampai dengan 1 yang dipilih secara acak. Tabel 4 merupakan bobot awal yang digunakan pada setiap *node*.

Tahapan selanjutnya adalah menentukan laju pembelajaran (α) atau *learning rate*. Pada penelitian ini, nilai laju pembelajaran awal yang digunakan adalah 0,5. Untuk setiap iterasi yang dilakukan, nilai α akan diperbarui dengan menggunakan Pers. (2) (Satoto et al., 2015). Setelah nilai bobot awal dan nilai α ditentukan, langkah selanjutnya adalah mencari *node* dengan nilai bobot terkecil atau disebut sebagai *best matching unit* (BMU). Jarak Euclidean digunakan untuk menentukan nilai BMU. Pers. (3) merupakan perhitungan jarak dengan menggunakan jarak Euclidean. Di mana x_i merupakan nilai parameter ke-i, sedangkan w_i merupakan bobot pada parameter ke-i. Nilai BMU menunjukkan tingkatan kedekatan *node* dengan data. Sehingga data dengan *node* yang sama berada pada satu kelompok yang sama. Pada tahapan ini, pengelompokkan data mulai terbentuk. Akan tetapi, anggota kelompok pada setiap kelompok masih belum stabil. Sehingga perlu dilakukan tahapan pembaharuan bobot pada iterasi selanjutnya. Seluruh bobot pada setiap *node* akan diperbaharui dengan menggunakan Pers. (4).

$$\alpha_{baru} = \frac{1}{2} * \alpha_{lama} \quad (2)$$

$$Jarak = \sqrt{\sum_{i=0}^{n}(x_i - w_i)^2} \quad (3)$$

$$w_{baru} = w_{lama} + \alpha(x - w_{lama}) \quad (4)$$

Tahapan mulai dari mencari nilai BMU hingga pembaharuan bobot akan terus diulang hingga angkota kelompok pada sebuah kelompok tidak berubah atau mencapai batasan iterasi.

2.4 Validasi Kelompok

Setelah kelompok data didapatkan, peneliti melakukan validasi kelompok untuk mengetahui kualitas pengelompokan yang dihasilkan oleh SOM. Koefisien *silhouette* digunakan untuk mengetahui kualitas pengelompokan tersebut. Pers. (5) digunakan untuk mencari nilai koefisien *silhouette* (s). Di mana a adalah nilai jarak intra-kelompok sedangkan b adalah nilai antar antar-kelompok.

$$s = \frac{b-a}{\max(a,b)} \quad (5)$$

Jika nilai s adalah 1, maka kelompok terpisah dengan baik dengan kelompok yang lainnya. Jika nilai s adalah 0, maka kelompok tidak terpisah dengan baik. Masih terdapat yang salah dikelompokkan pada nilai ini. Namun, jika nilai s adalah -1, maka metode pengelompokkan data tidak tepat dalam membuat kelompok.

2.5 Menentukan Kelompok Prioritas

Tahapan terakhir dalam penelitian ini adalah menentukan kelompok yang masuk ke dalam tingkatkan prioritas. Untuk menentukan kondisi tersebut, peneliti menggunakan metode pemeringkatan SAW. Nilai yang digunakan pada proses pemeringkatan adalah nilai rata-rata tiap parameter pada setiap kelompok. Pers. (6) digunakan untuk mengetahui nilai setiap kelompok berdasarkan SAW.

$$A_i = \sum_{j=1}^{N} r_{ij} W_j, untuk\ i = 1,2,3, \dots m \quad (6)$$

Di mana A_i adalah nilai SAW untuk kelompok ke-i. Nilai r_{ij} adalah nilai ternormalisasi untuk kelompok ke-i parameter ke-j, sedangkan W_j adalah bobot paramater ke-j. Pada penelitian ini, setiap nilai parameter akan dinormalisasi. Tujuan dari normalisasi adalah untuk menyamakan unit yang digunakan oleh paramater dan membedakan tingkat kepentingan antara parameter yang dianggap sebagai keuntungan (*benefit*) dan juga biaya (*cost*). Parameter keuntungan adalah parameter yang semakin besar nilainya, maka semakin mempengaruhi tingkatan prioritas yang diberikan. Sebaliknya, parameter biaya merupakan parameter yang semakin besar nilainya,

maka akan semakin menurunkan tingkat prioritas kelompok obyek wisata. Pers. (7) dan Pers. (8) digunakan untuk proses normalisasi. X_{ij} merupakan nilai asli kelompok ke-i paramater ke-j.

Pada penelitian ini parameter keuntungan adalah P1, P2, dan P4. Sedangkan parameter biaya adalah P3. Parameter P3 menjadi parameter biaya dikarenakan nilai yang digunakan kepada RIPPARDA berdasarkan nilai *importance performance analysis* (IPA). Semakin besar nilai IPA, maka sarana prasarana pada obyek wisata semakin baik, sehingga semakin turun prioritas pengembangan yang akan diberikan.

$$r_{ij} = \frac{X_{ij}}{\max(X_{ij})} \quad (7)$$

$$r_{ij} = \frac{\min(X_{ij})}{\mathrm{X_{ij}}} \quad (8)$$

Selanjutnya, ditentukan bobot untuk setiap parameter. Bobot pada konteks SAW berbeda dengan bobot yang digunakan pada proses SOM. Bobot pada SAW merujuk kepada derajat kepentingan sebuah paramater terhadap parameter yang lainnya, sehingga total bobot pada SAW adalah satu. Penentuan bobot parameter ditentukan secara subyektif berdasarkan tingkat kepentingan parameter dalam hal pengembangan obyek pariwisata. Tabel 5 merupakan bobot yang digunakan dalam mencari nilai SAW untuk setiap kelompok pada penelitian ini. Selanjutnya, kelompok obyek pariwisata dengan prioritas pengembangan tinggi ditunjukkan dengan nilai SAW terbesar.

Tabel 5 Bobot SAW

Parameter	Bobot
P1	0,50
P2	0,30
P3	0,05
P4	0,15

3. HASIL DAN PEMBAHASAN

Hasil percobaan yang dilakukan, menghasilkan jumlah anggota pada C1 sebanyak empat objek wisata, C2 sebanyak 20 objek wisata dan C3 sebanyak 10 objek wisata. Kelompok C1 mendapatkan nilai koefisien *silhouette* sebesar 0,75006, Kelompok C2 mendapatkan nilai koefisien *silhouette* sebesar -0,18109, sedangkan kelompok C3 mendapatkan nilai *silhouette* sebesar 0,37651. Berdasarkan hasil validasi kelompok, kelompok C1 memiliki nilai koefisien *silhouette* yang mendekati angka 1. Hal ini berarti bahwa kelompok C1 terpisah dengan baik dengan kelompok data yang lain. Pada kelompok C1, nilai a relatif kecil, sehingga menunjukkan data-data di dalam kelompok C1 cenderung memusat. Sedangkan pada C2 mendapatkan nilai a relatif besar, sehingga terdapat kecenderungan data-data pada kelompok C2 berada pada lokasi yang menyebar. Hal ini berdampak kepada nilai koefisien *silhouette* sebesar -0,18109. Nilai koefisien *silhouette* pada kelompok C2 yang mendekati nilai 0 menunjukkan bahwa kelompok C2 tidak terpisah dengan baik dengan kelompok yang lainnya. Pada kelompok C3, nilai a relatif kecil. Akan tetapi, terdapat data pada kelompok C3 yang relatif lebih dekat ke kelompok yang lain. Fenomena ini menujukan bahwa terdapat beberapa data di dalam kelompok C3 yang seharusnya tidak masuk ke dalam C3.

Selanjutnya, setelah kelompok-kelompok obyek wisata telah terbentuk, dilakukan pencarian terhadap kelompok yang akan menjadi prioritas pengembangan pariwisata. Parameter yang digunakan adalah nilai rata-rata parameter seitap kelompok. Nilai rata-rata parameter tiap kelompok ditunjukkan oleh Tabel 6. Nilai rata-rata yang digunakan adalah nilai parameter asli yang belum dinormalisasi, sedangkan Tabel 7 merupakan hasil normalisasi nilai pada Tabel 6. Nilai pada Tabel 7 kemudian digunakan untuk menghitung nilai SAW untuk setiap kelompok berdasarkan Pers. (6). Hasil perhitungan SAW ditunjukkan pada Tabel 8.

Tabel 6 Nilai Rata-rata Parameter Tiap Kelompok

Kelompok	P1	P2	P3	P4
C1	118280	7966	0,733	2,500
C2	12152	630	0,857	1,769
C3	19836	174	0,729	1,000

Tabel 7 Hasil Normalisasi SAW

Kelompok	P1	P2	P3	P4
C1	1,0000	1,0000	0,9950	1,0000
C2	0,1027	0,0791	0,8505	0,7077
C3	0,1677	0,0218	1,0000	0,4000

Tabel 8 Nilai SAW Tiap Kelompok

Kelompok	Nilai SAW
C1	0,999749
C2	0,223771
C3	0,200388

Berdasarkan perhitungan SAW, kelompok C1 menduduki peringkat pertama dengan nilai 0,999749. Selanjutnya, kelompok C2 dengan menduduki peringkat kedua dengan nilai 0,223771, dan lelompok C3 menduduki peringkat ketiga dengan nilai 0,200388. Berdasarkan hasil tersebut, Kelompok C1 merupakan kelompok obyek pariwisata yang direkomendasikan menjadi kelompok dengan prioritas tertinggi dalam proses pengembangan pariwisata, diikuti dengan kelompok C2 dan C3 yang merupakan kelompok obyek wisata dengan prioritas sedang dan rendah.

4. KESIMPULAN

Pada penelitian ini telah dilakukan pengelompokan data obyek pariwisata di Kabupaten Probolinggo dengan menerapkan metode SOM dan SAW untuk menentukan tingkat prioritas pengembangan obyek pariwisata. Hasil penelitian menunjukkan bahwa metode SOM dapat digunakan untuk mengelompokkan data obyek pariwisata dengan cukup baik. Selain itu, hasil validasi kelompok dengan menggunakan koefisien *silhouette* mendapatkan nilai cukup baik. Selanjutnya, dalam hal menentukan rekomendasi prioritas pengembangan, metode SAW mendapatkan hasil yang cukup baik ditinjau dari hasil pemeringkatan dan nilai setiap data pada kelompok obyek pariwisata. Metode SAW menghasilkan hasil yang baik dengan nilai parameter yang linier seperti tinggi, sedang, rendah seperti yang diterapkan pada penelitian ini.

Hal yang dapat diperbaiki untuk penelitian selanjutnya adalah menambahkan lebih banyak jumlah data pariwisata baru dan menambahkan parameter baru seperti parameter jarak, sehingga nantinya tingkat kualitas pengelompokan obyek wisata yang dihasilkan bisa menghasilkan kelompok yang lebih baik.

DAFTAR PUSTAKA

Al-Otaibi, R., Jin, N., Wilcox, T., & Flach, P. (2016). Feature Construction and Calibration for Clustering Daily Load Curves from Smart-Meter Data. *IEEE Transactions on Industrial Informatics*, *12*(2), 645–654. https://doi.org/10.1109/TII.2016.2528819

Arifin, J. (2020, October). *SDM Pelaku Wisata di Probolinggo Perlu Ditingkatkan*. Radar Bromo. https://radarbromo.jawapos.com/daerah/kraksaan/27/10/2020/sdm-pelaku-wisata-di-probolinggo-perlu-ditingkatkan/

Asan, U., & Ercan, S. (2012). An Introduction to Self-Organizing Maps. In *Computational Intelligence Systems in Industrial Engineering* (pp. 295–315). https://doi.org/10.2991/978-94-91216-77-0_14

Badan Pusat Statistik. (2020). *BPS Kabupaten Probolinggo*. Badan Pusat Statistik. https://probolinggokab.bps.go.id/

Bi, W., Cai, M., Liu, M., & Li, G. (2016). A Big Data Clustering Algorithm for Mitigating the Risk of Customer Churn. *IEEE Transactions on Industrial Informatics*, *12*(3), 1270–1281. https://doi.org/10.1109/TII.2016.2547584

Hale, R. L. (1981). Cluster analysis in school psychology: An example. *Journal of School Psychology*, *19*(1), 51–56. https://doi.org/10.1016/0022-4405(81)90007-8

Hartatik, H., & Cahya, A. S. D. (2020). Clusterisasi Kerusakan Gempa Bumi di Pulau Jawa Menggunakan SOM. *Jurnal Ilmiah Intech : Information Technology Journal of UMUS*, *2*(02). https://doi.org/10.46772/intech.v2i02.286

Maulida, L. (2018). Penerapan Data Mining dalam Mengelompokkan Kunjungan Wisatawan ke Objek Wisata Unggulan di Prov. DKI Jakarta dengan K-Means. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *2*(3), 167. https://doi.org/10.14421/jiska.2018.23-06

Miljkovic, D. (2017). Brief review of self-organizing maps. *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 1061–1066. https://doi.org/10.23919/MIPRO.2017.7973581

Muin, A. A., Informasi, S., Sain, F., Teknik, D., & Alauddin, U. (2018). Implementasi Self Organizing Maps (SOM) Klasifikasi Penduduk untuk Menentukan Keputusan Pembangunan Daerah Prioritas Miskin (Studi Kasus Kota Makassar). *Jurnal INSYPRO (Information System and Processing)*, *3*(2). https://doi.org/10.24252/INSYPRO.V3I2.5888

Satoto, B. D., Muhammad, A., & Khotimah, B. K. (2015). Pengelompokan Tingkat Kesehatan Masyarakat Menggunakan Shelf Organizing Maps Dengan Cluster Validation. *Seminar Nasional Aplikasi Teknologi Informasi (SNATI) 2015*.

Simamora, R. K., & Sinaga, R. S. (2016). Peran Pemerintah Daerah dalam Pengembangan Pariwisata Alam dan Budaya di Kabupaten Tapanuli Utara. *JPPUMA: Jurnal Ilmu Pemerintahan Dan Sosial Politik UMA (Journal of Governance and Political Social UMA)*, *4*(1), 79–96. https://doi.org/10.31289/jppuma.v4i1.895

Song, X., Li, W., Ma, D., Wu, Y., & Ji, D. (2018). An Enhanced Clustering-Based Method for Determining Time-of-Day Breakpoints Through Process Optimization. *IEEE Access*, *6*, 29241–29253. https://doi.org/10.1109/ACCESS.2018.2843564

Umar, R., Fadlil, A., & Az-Zahra, R. R. (2018). Self Organizing Maps (SOM) untuk Pengelompokkan Jurusan di SMK. *Khazanah Informatika : Jurnal Ilmu Komputer Dan Informatika*, *4*(2), 131–137. https://doi.org/10.23917/KHIF.V4I2.7044

Wira, B., Budianto, A. E., & Wiguna, A. S. (2019). Implementasi Metode K-Medoids Clustering untuk Mengetahui Pola Pemilihan Program Studi Mahasiswa Baru Tahun 2018 di Universitas Kanjuruhan Malang. *RAINSTEK : Jurnal Terapan Sains & Teknologi*, *1*(3), 53–68. https://doi.org/10.21067/jtst.v1i3.3046

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023, Pp. 10 – 21
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Penerapan *Data Mining* dengan Metode Regresi Linear untuk Memprediksi Data Nilai Hasil Ujian Menggunakan RapidMiner

Muhammad Sholeh [(1)*], Erna Kumalasari Nurnawati [(2)], Uning Lestari [(3)]
Informatika, Fakultas Teknologi Informasi dan Bisnis, Institut Sains & Teknologi AKPRIND, Yogyakarta
e-mail : {muhash,ernakumala,uning}@akprind.ac.id.
* Penulis korespondensi.
Artikel ini diajukan 7 Juni 2022, direvisi 1 September 2022, diterima 3 September 2022, dan dipublikasikan 30 Januari 2023.

Abstract

Prediction is one of the methods in data mining. One of the models that can be used in prediction is using linear regression. Linear regression is used to make predictions on the data that has been provided. In this study, a linear regression model was made with a datasheet containing data that affected student achievement in achieving final exam scores. The linear regression model developed can be used to predict student test scores. The linear regression model developed can be used to predict student test scores. The datasheet used in the test uses a public datasheet, namely student_performance.csv. The datasheet consists of 395 records and 33 attributes. The attributes used are selected that influence the label. The selection of attributes is based on the results of the weighting in the process of checking the correlation matrix. Based on the weighting, the attributes used are seven attributes and one attribute becomes a label. The research method uses CRISP DM which consists of business understanding, data understanding, data preparation, model making, evaluation, and deploying. The data mining process uses the Rapid Miner application. The results of the study resulted in a linear regression model $y = 0.729 - (0.024 \times Medu) - (0.020 \times Fedu) + (0.053 \times failures) - (0.077 \times goout) - (0.012 \times absences) + (0.126 \times G1) + (0.862 \times G2)$*. The result of evaluating the performance of the RMSE value was 0.675. Based on these results, it can be concluded that the resulting model can be recommended for use in predicting student test scores.*

Keywords: Model, Data Mining, Linear Regression, RapidMiner, Datasheet

Abstrak

Salah satu metode dalam *data mining* adalah prediksi. Salah satu model yang dapat digunakan dalam prediksi adalah menggunakan regresi linear. Regresi linear digunakan untuk melakukan prediksi pada data yang sudah disediakan. Pada penelitian ini dilakukan pembuatan model regresi linear dengan *datasheet* berisi data-data yang mempengaruhi prestasi siswa dalam meraih nilai ujian akhir. Model regresi linear yang dikembangkan dapat digunakan untuk melakukan prediksi hasil nilai ujian siswa. *Datasheet* yang digunakan dalam pengujian menggunakan *datasheet* publik yaitu student_performance.csv. *Datasheet* terdiri dari 395 data dan 33 atribut. Atribut yang digunakan dipilih yang mempunyai pengaruh pada label. Pemilihan atribut berdasar hasil pembobotan pada proses pengecekan korelasi matriks. Berdasarkan pada pembobotan, atribut yang digunakan adalah tujuh atribut dan satu atribut menjadi label. Metode penelitian menggunakan CRISP-DM yang terdiri dari *business understanding, data understanding, data preparation*, pembuatan model, evaluasi dan *deploying*. Proses *data mining* menggunakan aplikasi RapidMiner. Hasil penelitian menghasilkan model regresi linear $y = 0{,}729 - (0{,}024 \times Medu) - (0{,}020 \times Fedu) + (0{,}053 \times failures) - (0{,}077 \times goout) - (0{,}012 \times absences) + (0{,}126 \times G1) + (0{,}862 \times G2)$. Hasil evaluasi performance nilai RMSE adalah 0,675. Berdasar hasil tersebut dapat disimpulkan bahwa model yang dihasilkan dapat direkomendasikan untuk digunakan dalam memprediksi nilai ujian siswa.

Kata Kunci: Model, *Data Mining*, Regresi Linear, RapidMiner, *Datasheet*

1. PENDAHULUAN

Perkembangan *data mining* tumbuh dengan sangat pesat. Hal ini seiring dengan tumbuhnya data yang semakin banyak dan digunakan dalam proses pengambilan kebijakan. Kebijakan yang diambil dengan menggunakan data dilakukan dengan pembuatan model *data mining*. Saat ini, data menjadi unsur penting dalam suatu perusahaan. Data menjadi aset yang dapat digunakan untuk mencari pola yang dapat digunakan dalam pengambilan kebijakan. Informasi dari model yang diolah dapat digunakan dalam memproyeksikan strategi atau kebijakan yang dilakukan untuk proses pengembangan bisnis. Proses penelusuran data yang besar harus dilakukan secara cermat. Semakin besar data semakin besar proses untuk melakukan pemilahan data yang sesuai dengan keperluan.

Data mining merupakan proses menemukan informasi dari suatu data yang tersimpan dalam suatu *database* atau *datasheet*. Pembuatan model dilakukan dengan proses menggunakan algoritma atau rumus tertentu. Proses *data mining* menggunakan berbagai teknik seperti teknik dalam proses statistik, matematika, dan *machine learning* yang digunakan dalam melakukan identifikasi dan mengolah berbagai data menjadi informasi yang bermanfaat (Arhami & Nasir, 2020; Jollyta et al., 2020).

Salah satu model prediksi yang digunakan pada *data mining* adalah regresi linear. Analisis regresi dapat digunakan untuk melihat pengaruh antara variabel bebas (*independen*) dan variabel tidak bebas (*dependen*). Regresi linear dibedakan menjadi regresi sederhana dan regresi linear berganda. Regresi linear sederhana hanya terdapat satu variabel bebas dan satu variabel yang menjadi variabel tidak bebas dan regresi linear berganda apabila terdapat lebih dari satu variabel bebas. Kegunaan dari analisis regresi linear adalah untuk mengetahui arah dan seberapa besar pengaruh variabel independen terhadap variabel dependen. Variabel yang dapat mempengaruhi sering disebut variabel dependen atau tidak bebas dan variabel yang mempengaruhi variabel lain disebut variabel independen atau variabel bebas.. Model persamaannya matematika ditampilkan pada Pers. (1).

$$y = a_0 + \ A_1x_1 + \ A_2x_2 \ + \ \ldots\ldots \ + A_nx_n \qquad (1)$$

Di mana y adalah variabel *dependen* dan $x_1, x_2, \ldots, x_n$ merupakan variabel *independen*, a merupakan nilai konstanta, dan b adalah nilai koefisien regresi (Kurniawan, 2016).

Model regresi linear dalam *data mining* menjadi salah satu model yang banyak digunakan. Penggunaan regresi linear dalam *data mining* menggunakan berbagai *datasheet* baik *datasheet* milik sendiri ataupun *datasheet* publik. Salah satu *datasheet* publik yang sering dugunakan dalam pembuatan model *data mining* adalah *datasheet* student_rerformance.csv yang diambil dari https://archive.ics.uci.edu/ml/datasheets.php. *Datasheet* ini berisi data yang diolah dari data siswa dan data keluarga pada sekolah menengah di Portugal. *Datasheet* tersebut menjadi salah satu *datasheet* yang digunakan dalam penelitian pembuatan model *data mining*. Setiyorini & Asmono (2020) menggunakan *Student Performance dataset* untuk pembuatan model klasifikasi terutama dengan menggunakan K-Nearest Neighbour, Ünal (2021) membuat model klasifikasi dengan menggunakan Decision Tree, Random Forest dan Naive Bayes, Deepika & Sathyanarayana (2018) menggunakan Decision Tree, dan Oyedeji et al. (2020) yang melakukan analisis kinerja akademik siswa untuk mengetahui cara-cara meningkatkan kinerja individu siswa.

Model *data mining* untuk memprediksi keberhasilan siswa dengan *datasheet* yang selain *student performance* dan menggunakan *datasheet* yang bersifat privat dilakukan oleh Ofori et al. (2020) yang melakukan identifikasi model *machine learning* untuk melakukan prediksi kinerja siswa dan model *machine learning* yang tepat dalam meningkatkan pembelajaran bagi siswa. Bahri et al. (2022), membuat model yang diharapkan dapat membantu siswa dalam menentukan jurusan yang akan diambil dalam menempuh jenjang pendidikan tinggi. Hasil pengujian, faktor yang paling mempengaruhi kesalahan dalam mengambil jurusan di perguruan tinggi adalah variabel informasi jurusan dan pengujian yang dilakukan dengan menggunakan tiga algoritma, algoritma

Decision Tree merupakan algoritma yang menghasilkan nilai akurasi tertinggi dengan tingkat akurasi tinggi 75,38%. Penelitian lain yang masih terkait *data mining* yang menggunakan data siswa dilakukan oleh Hendrian (2018), Putro et al. (2021), dan Ramadhani & Hendriyani (2021).

Penelitian *data mining* yang menggunakan berbagai *datasheet* dan menggunakan model regresi linear sudah dilakukan dengan menggunakan berbagai aplikasi. Ariesanto & Ekka (2020) melakukan penelitian dengan menerapkan *data mining* dengan regresi linear. Regresi linear digunakan untuk melakukan prediksi harga suatu saham pada perusahaan pelayaran. Evaluasi dilakukan dengan nilai *Root Mean Square Error*. Nilai RMSE menunjukkan angka plus 7,522 dari data aktual harga penutupan saham. Gaol et al. (2019) menggunakan regresi linear berganda untuk melakukan prediksi data dalam ketersediaan buku. Penelitian serupa juga dilakukan oleh Rahayu et al. (2022), Sinaga et al. (2022), dan Siregar (2021).

Pembuatan model *data mining* dapat diimplementasikan dengan berbagai aplikasi baik yang berbasis dengan membuat program dan menggunakan aplikasi Visual Programming. Pengembangan model *data mining* yang menggunakan bahasa pemrograman sudah dilakukan oleh Sholeh et al. (2022). Penelitian dilakukan dengan menggunakan *datasheet* asuransi kesehatan untuk memprediksi biaya asuransi. Pembuatan model menggunakan pemrograman Python dengan memanfaatkan berbagai *library* yang mendukung proses pembuatan model seperti pandas dan sklearn. Penelitian sejenis yang menggunakan pemrograman Python untuk *data mining* dilakukan oleh Kurniatullah & Pramudi (2017), N. et al. (2019), Nishadi (2019), dan Prabha et al. (2020).

Selain menggunakan bahasa pemrograman, pembuatan model *data mining* dapat menggunakan Visual Programming seperti RapidMiner. RapidMiner dapat digunakan untuk melakukan proses analisis pada *data mining*, *text mining*, dan analisis prediksi. RapidMiner menggunakan berbagai cara dan teknik deskriptif serta prediksi dalam pembuatan model yang dapat digunakan dalam pengambilan keputusan (Chisholm, 2013). Penggunaan RapidMiner dalam pembuatan model tidak memerlukan program dan semua yang digunakan dalam pembuatan model sudah tersedia dalam bentuk operator. Pembuatan model menggunakan berbagai operator yang sesuai dan saling dikaitkan dalam membentuk suatu model. Proses pembuatan *data mining* dengan RapidMiner sudah dilakukan beberapa peneliti dengan berbagai *datasheet* dan model serta algoritma yang digunakan. Penelitian tersebut di antaranya dilakukan oleh Chisholm (2013), Prasetyo et al. (2021), dan Sudarsono et al. (2021).

Berdasar latar belakang, tinjauan pustaka, dan studi literatur, proses *data mining* menjadi salah satu cara dalam memberikan informasi dalam bentuk model dalam pengambilan keputusan. Salah satu model yang dapat digunakan dalam pembuatan *data mining* adalah regresi linear. Batasan dalam penelitian ini adalah membuat model regresi linear dengan *datasheet student performance* dan tidak semua atribut digunakan. *Datasheet student performance* merupakan *datasheet* publik, yang terdiri dari 395 data dan 33 atribut. *Datasheet* tersebut dapat diunduh pada laman www.archive.ics.uci.edu/ml/datasheets/student+performance. Pertimbangan penggunaan *datasheet* ini karena memiliki jumlah data dan atribut cukup banyak sehingga dapat dilakukan berbagai pengujian sehingga dapat ditentukan atribut apa saja yang mempengaruhi dalam pembuatan model. Dengan demikian, model yang dihasilkan diharapkan dapat digunakan dalam memprediksi nilai siswa.

2. METODE PENELITIAN

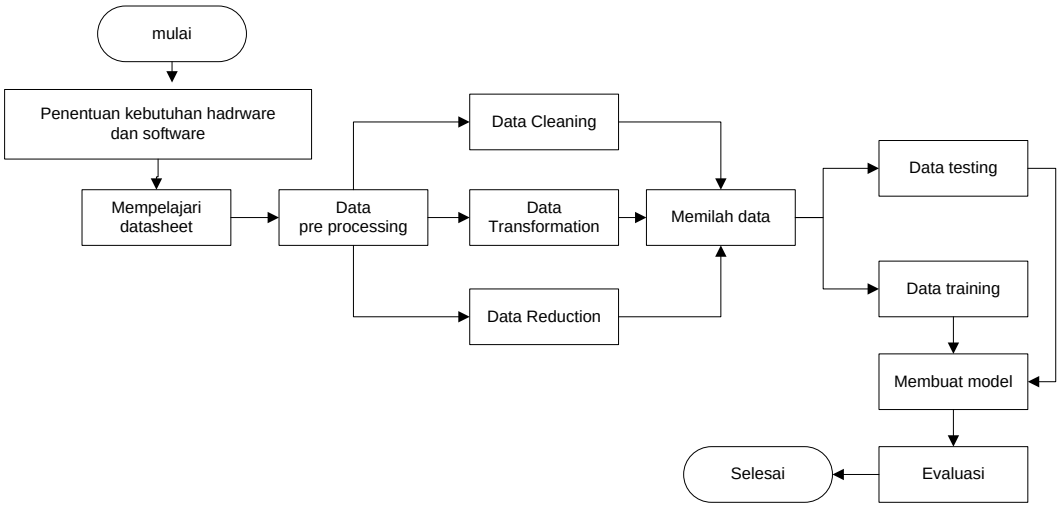
Metodologi penelitian dalam membuat model *data mining* regresi linear, menggunakan metodologi CRISP-DM. Pembuatan model dengan metodologi CRISP-DM terdapat enam tahapan, yaitu *business understanding*, *data understanding, data preparation, modeling, evaluation,* dan *deployment* (Hidayati et al., 2021).

2.1 *Datasheet*

Datasheet yang digunakan adalah StudentPerformance.csv. Sumber *datasheet* ada pada laman https://archive.ics.uci.edu/ml/datasheets/student%2Bperformance. *Datasheet* terdiri dari 395 baris dan 33 atribut.

2.2 Tahapan Penelitian

Tahapan penelitian dengan berdasar pada model CRISP-DM, langkah–langkah yang dilakukan di antaranya mempelajari *datasheet* dan membersihkan *datasheet* seperti memeriksa data kosong, memeriksa batasan nilai dan lainnya. Dalam pembuatan model, *datasheet* dibagi menjadi dua terdiri dari 80% *datasheet* digunakan untuk *data training* dan 20% digunakan untuk *data testing*. Gambar 1 merupakan tahapan dalam pembuatan model *data mining.*



Gambar 1 Tahapan Pembuatan *Data Mining*

3. HASIL DAN PEMBAHASAN

3.1 *Business Understanding*

Langkah awal dalam penelitian adalah mengidentifikasi manfaat dan kegunaan dari model yang dikembangkan. *Datashee*t dilakukan identifikasi keterkaitan antar atribut terutama dengan atribut yang menjadi label. Hasil dari model, model diharapkan dapat digunakan untuk melakukan identifikasi faktor-faktor yang berkontribusi terhadap kegagalan siswa dalam menempuh ujian dan dapat digunakan untuk melakukan prediksi nilai ujian akhir.

3.2 *Data Understanding*

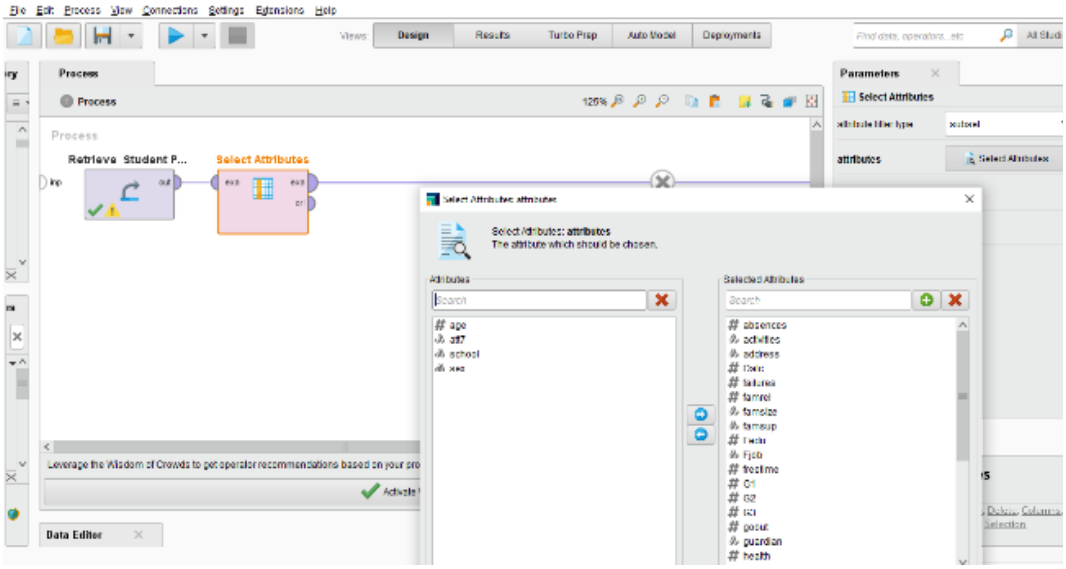
Proses ini dilakukan dengan mengidentifikasi atribut yang ada dalam *datasheet*. Atribut yang ada dilakukan proses pemilahan awal, di antaranya adalah menentukan atau memilih atribut yang tidak mempunyai keterkaitan dalam pembuatan model. Atribut yang tidak mempengaruhi dalam pembuatan model tidak akan digunakan.

3.3 *Data Preparation*

Data preparing sangat diperlukan dan bertujuan untuk mengolah data agar data tidak mengandung kesalahan. Pemeriksaan data yang dilakukan diantaranya memeriksa data kosong, data yang di luar ambang batas dan tipe data yang tidak sesuai. Proses *data preparing* yang dilakukan sebagai berikut.

1) Memilih atribut yang akan digunakan.

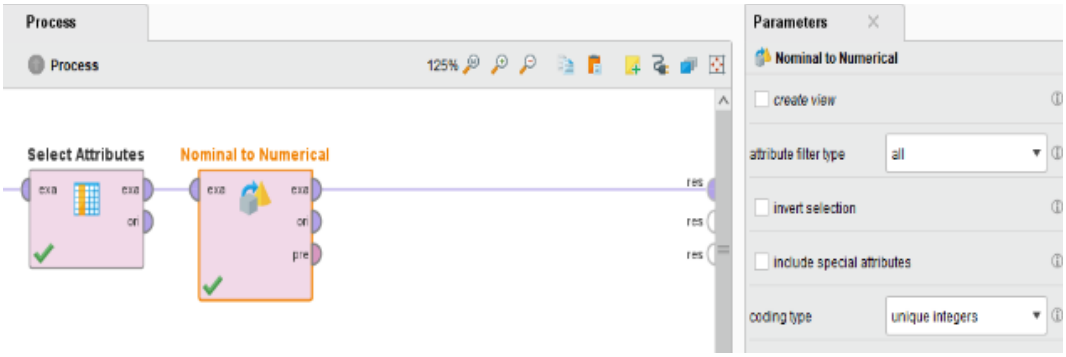
Tidak semua atribut digunakan, sehingga akan dipilih atribut yang mempengaruhi model. Dari 33 atribut yang ada, atribut att, sex, age, dan school tidak dipilih dalam proses pembuatan model. Atribut tersebut tidak saling mempengaruhi dalam pembuatan model. Gambar 2 menunjukkan penggunaan operator *select atribut* dalam proses pemilihan atribut.



Gambar 2 Penggunaan Operator *Select Atribut* dalam Proses Pemilihan Atribut

2) Mengubah tipe data nominal menjadi numerik.

Atribut yang berisi data selain numerik akan diubah menjadi atribut yang berisi data numerik. Atribut sex yang berisi nominal F dan M akan diubah menjadi 0 dan 1, atribut Mjob yang berisi other, services, at_home, teacher, dan health akan diubah menjadi numerik 0-4. Gambar 3 merupakan penggunaan operator *nominal to numerical* dalam proses mengubah tipe nominal menjadi numerik.

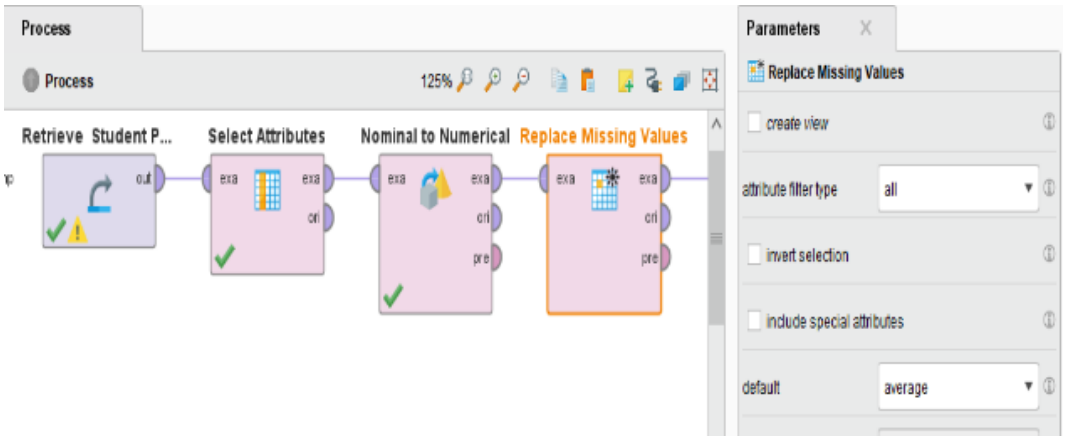


Gambar 3 Penggunaan Operator *Nominal to Numerical*

3) Mengganti data *missing value* dengan data tertentu.

Semua isi *datasheet* yang digunakan harus terisi data atau tidak boleh mengandung data kosong. Agar data tidak mengadung data kosong, data yang berisi nilai kosong akan diganti dengan data rata-rata pada masing-masing atribut. Proses untuk mengganti data kosong dengan data tertentu

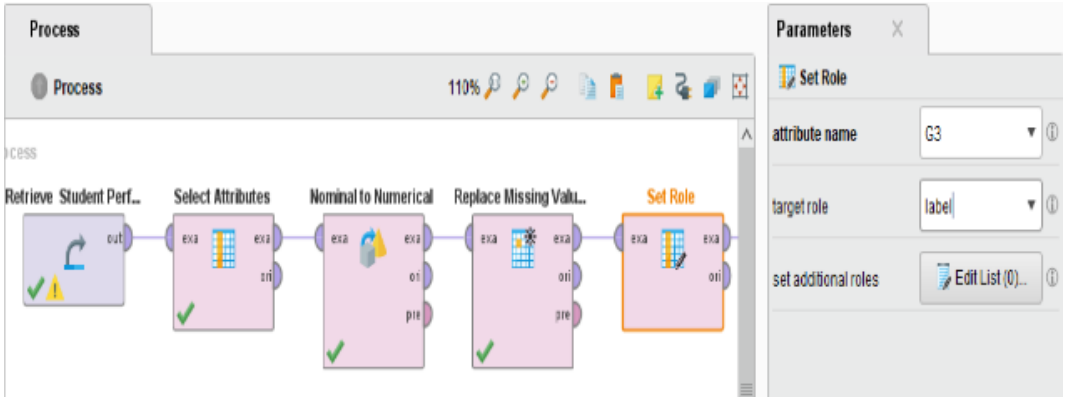
dapat menggunakan operator *replace missing value*. Gambar 4 memperlihatkan proses penggantian data kosong dengan nilai rata-rata.



Gambar 4 Proses Mengganti Data Kosong dengan Nilai Rata-Rata

4) Menentukan atribut yang menjadi label.

Label yang digunakan dalam pembuatan model adalah G3. Gambar 5 menunjukkan penggunaan operator *set role* untuk menentukan atribut G3 menjadi label.



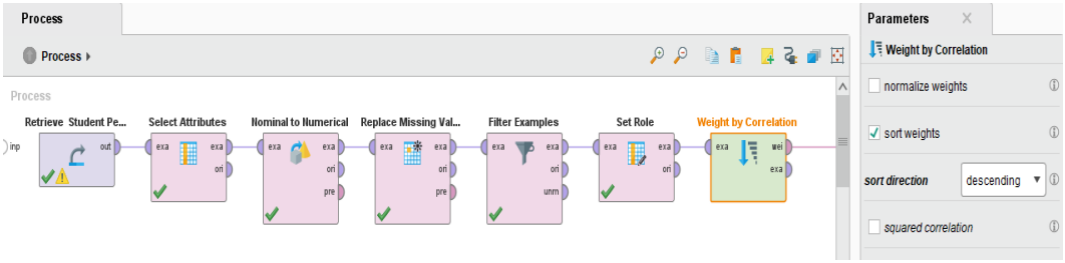
Gambar 5 Penggunaan Operator *Set Role* untuk Menentukan Label

3.4 Pembuatan model

Tahapan berikutnya adalah membuat model regresi linear dengan atribut yang sudah ditentukan dengan tahapan pemuatan model sebagai berikut.

1) Menampilkan bobot hasil matriks korelasi.

Matriks korelasi merupakan matriks yang memuat koefisien korelasi dari semua atribut yang digunakan. Hasil dari matriks dapat digunakan untuk memperoleh nilai kedekatan hubungan antar atribut. Gambar 6 menunjukkan penggunaan operator *weight by Correlation* untuk mengetahui keterkaitan antar atribut dan Gambar 7 memperlihatkan hasil pembobotan keterkaitan antar atribut terutama dengan atribut label yaitu G3.



Gambar 6 Penggunaan Operator *Weight by Correlation*

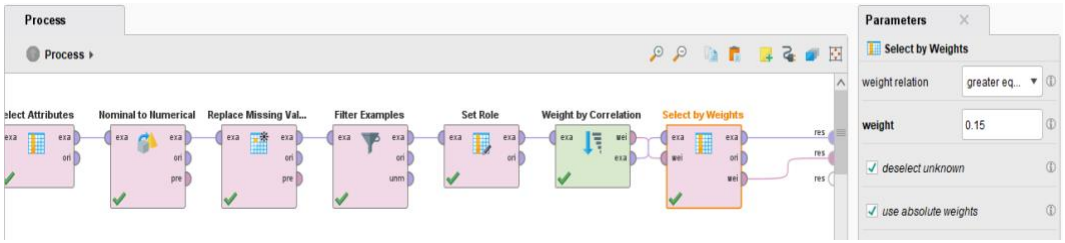
attribute	weight	attribute	weight
G2	0.966	traveltime	0.102
G1	0.892	Mjob	0.084
failures	0.294	health	0.082
schoolsup	0.238	famsup	0.067
absences	0.213	reason	0.061
Walc	0.190	activities	0.059
Medu	0.188	romantic	0.050
goout	0.177	famsize	0.040
Fedu	0.163	famrel	0.038
Dalc	0.141	guardian	0.035
address	0.130	Fjob	0.032
studytime	0.121	paid	0.029
higher	0.113	Pstatus	0.027
internet	0.112	nursery	0.027
		freetime	0.022

Gambar 7 Hasil Pembobotan Keterkaitan Atribut Terutama dengan Atribut Label G3

2) Menentukan nilai bobot yang digunakan dalam proses pembuatan model.

Hasil pembobotan pada Gambar 7, ditentukan batas nilai bobot yang digunakan. Dalam penelitian ini, batas pembobotan yang akan digunakan adalah pembobotan di atas 0,15. Batas ini dipilih dengan pertimbangan batas 0,15 masih mendekati batas korelasi cukup (>0,25 – 0,5) dan akan dilakukan pengujian, apakah atribut yang dibawah batas cukup mempengaruhi model. Atribut yang digunakan dalam membuat model adalah G2 (nilai ujian kedua), G1 (nilai ujian pertama), failures (berapa kali pernah tinggal kelas), schoolsup (tambahan pelajaran di luar sekolah) , absences (kehadiran di kelas), Walc (konsumsi alkohol), Medu (tingkat pendidikan orang tua (ibu)), goout (berapa banyak bermain/keluar rumah dengan teman sebaya), dan Fedu (tingkat pendidikan orang tua (ayah)).

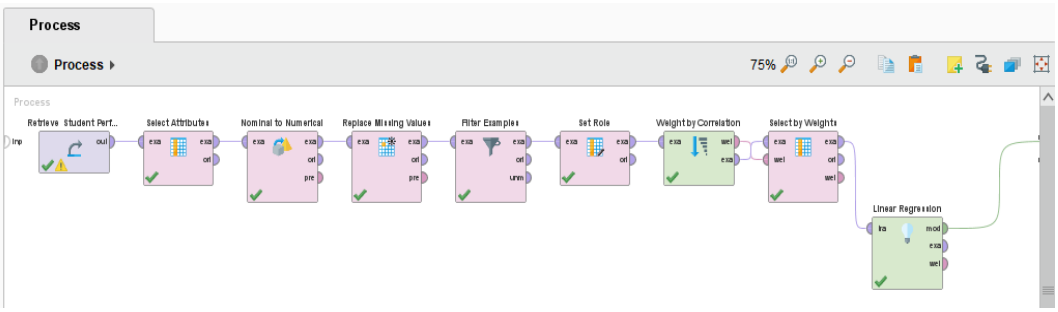
Proses untuk pemilihan bobot menggunakan operator *select by weight dengan* parameter pemilihan bobot di atas 0,15. Gambar 8 menunjukkan proses penggunaan operator *select by weight*.



Gambar 8 Proses Pemberian Bobot di Atas 0,15

3) Proses pembuatan model.

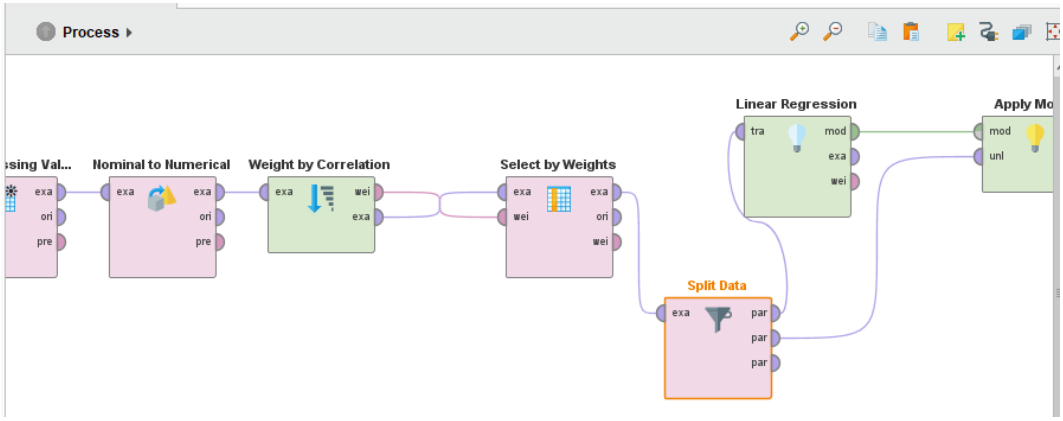
Berdasar pada pembobotan, pembuatan model *linear regresi* dapat dilakukan. Gambar 9 merupakan penggunaan operator *linear regresi* dalam proses pembuatan model.



Gambar 9 Penggunaan Operator *Linear Regresi*

4) Pembuatan model dengan *data training* dan pengujian dengan *data testing.*

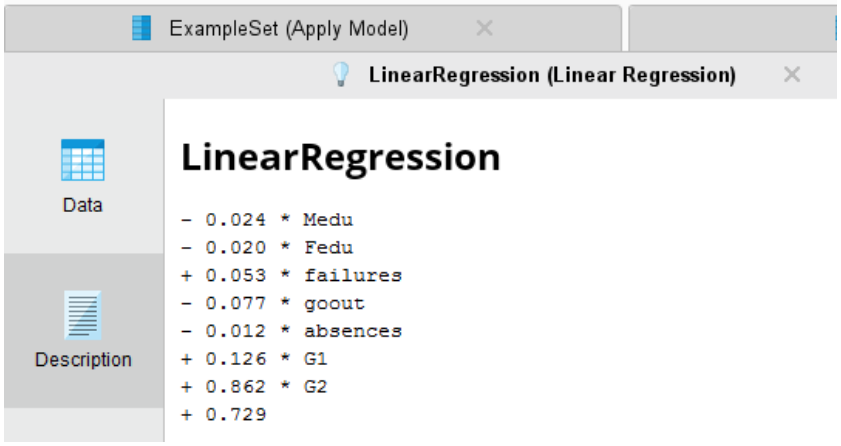
Model yang dihasilkan pada Gambar 9, menggunakan semua data yang ada di *datasheet* dan belum melakukan pengujian. Proses pembuatan model akan menggunakan *data training* dan proses evaluasi menggunakan *data testing*. Pembagian data dilakukan dengan membagi *data training* sebanyak 80% dan *data testing* sebanyak 20%. Proses pembagian data menggunakan operator *split data* dan proses untuk melakukan pengujian menggunakan operator *apply data*. Gambar 10 merupakan proses pembuatan data dengan *data training* dan pengujian dengan *data testing*.



Gambar 10 Proses Pembagian Data untuk *Data Training* dan *Data Testing*

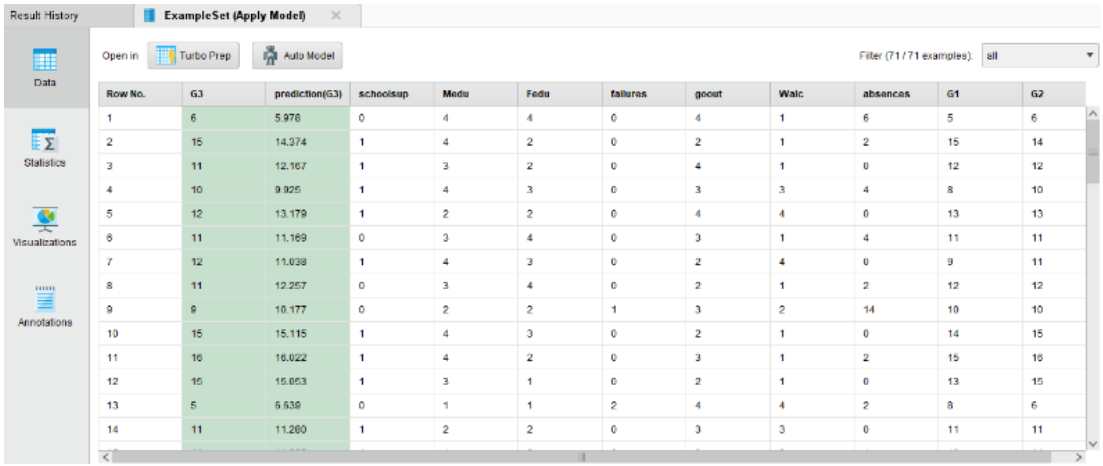
Hasil dari proses pembuatan model menghasilkan model linear regresi yang hasilnya ditunjukkan pada Pers. (2) dan Gambar 11.

$$y = 0{,}729 - (0{,}024 \times Medu) - (0{,}020 \times Fedu) + (0{,}053 \times failures) - (0{,}077 \times goout) - (0{,}012 \times absences) + (0{,}126 \times G1) + (0{,}862 \times G2) \quad (2)$$



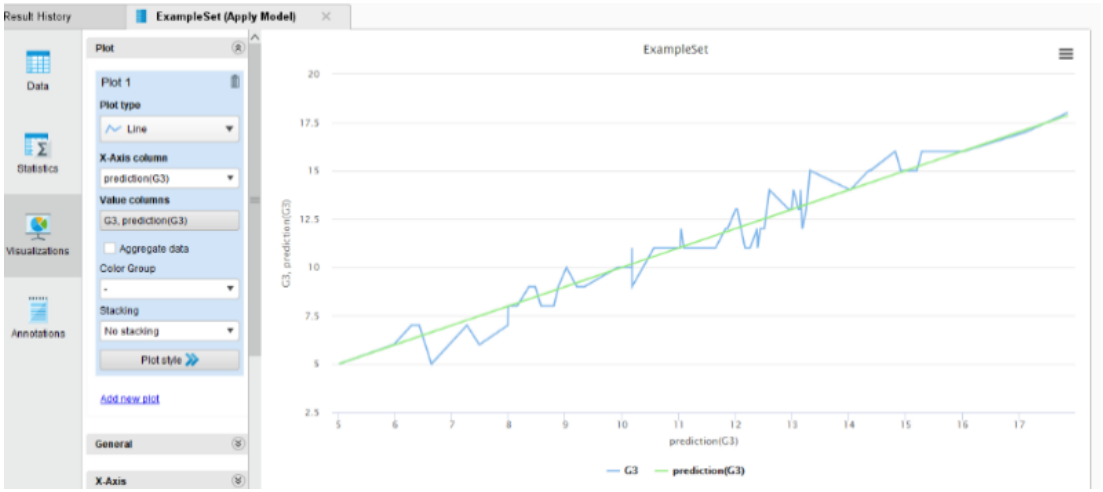
Gambar 11 Hasil Model Linear Regresi

Model yang dihasilkan pada Gambar 11, dilakukan pengujian dengan menggunakan *data training*. Hasil pengujian model dapat dilihat dengan melakukan perbandingan antara nilai G3 dengan nilai prediksi. Gambar 12 memperlihatkan hasil pengujian model dengan melihat perbandingan antara nilai G3 dengan nilai hasil prediksi. Sesuai hasil pada Gambar 12, pada baris 1, nilai G3 asli sebesar 6 dan hasil nilai prediksi G3 sebesar 5,978. Hasil ini menunjukkan ada selisih, hasil prediksi masih di bawah nilai G3 asli. Nilai prediksi baris 14, nilai G3 asli sebesar 11 dan hasil nilai prediksi G3 sebesar 11,280. Hasil ini menunjukkan ada selisih, hasil prediksi di atas dari nilai G3 asli. Hasil nilai prediksi dengan nilai G3 (nilai asli) dapat disajikan dalam bentuk grafik. Nilai prediksi ditunjukkan dengan garis lurus dan nilai asi ada di sekitar garis lurus hasil prediksi. Gambar 13 merupakan penyajian perbandingan antara nilai G3 dengan nilai hasil prediksi dalam bentuk grafik.



Row No.	G3	prediction(G3)	schoolsup	Medu	Fedu	failures	goout	Walc	absences	G1	G2
1	6	5.978	0	4	4	0	4	1	6	5	6
2	15	14.374	1	4	2	0	2	1	2	15	14
3	11	12.167	1	3	2	0	4	1	0	12	12
4	10	9.925	1	4	3	0	3	3	4	8	10
5	12	13.179	1	2	2	0	4	4	0	13	13
6	11	11.169	0	3	4	0	3	1	4	11	11
7	12	11.038	1	4	3	0	2	4	0	9	11
8	11	12.257	0	3	4	0	2	1	2	12	12
9	9	10.177	0	2	2	1	3	2	14	10	10
10	15	15.115	1	4	3	0	2	1	0	14	15
11	16	16.022	1	4	2	0	3	1	2	15	16
12	15	15.053	1	3	1	0	2	1	0	13	15
13	5	6.539	0	1	1	2	4	4	2	8	6
14	11	11.280	1	2	2	0	3	3	0	11	11

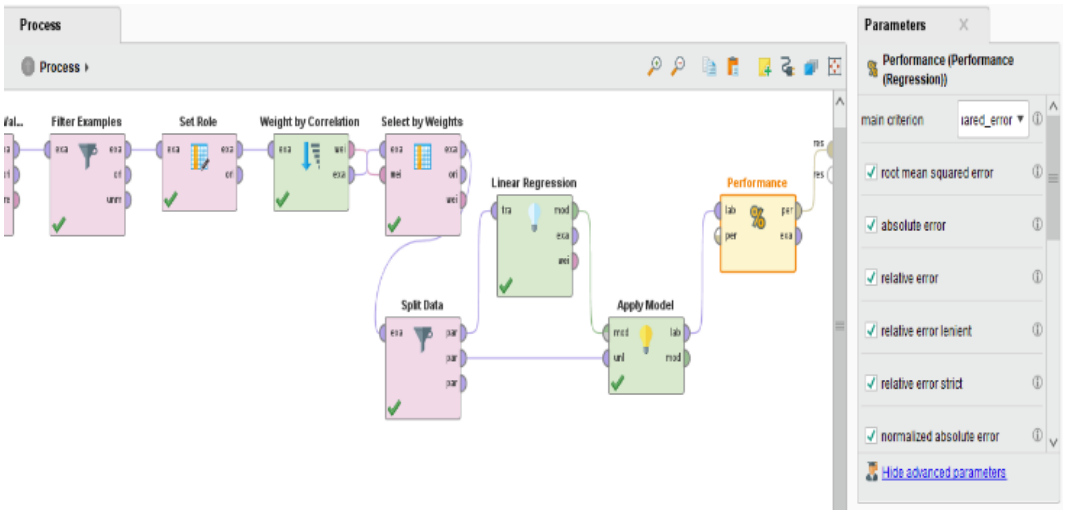
Gambar 12 Perbandingan Nilai G3 dengan Nilai Hasil Prediksi



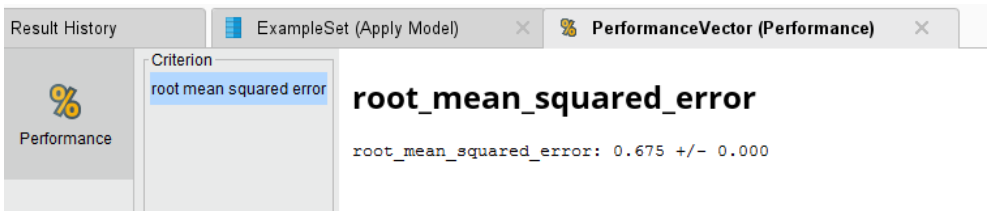
Gambar 13 Grafik Perbandingan Nilai G3 dengan Nilai Hasil Prediksi

3.5 Evaluasi Model

Model yang sudah selesai dibangun, dilakukan proses evaluasi. Proses evaluasi dilakukan untuk melihat *performance*. Proses evaluasi dilakukan dengan menggunakan *data testing* yang sudah ditentukan sebanyak 20% dari *datasheet*. Proses pengujian menggunakan operator *performance*. Gambar 14 merupakan proses pengujian dengan menggunakan operator *performance* dan Gambar 15 adalah hasil luaran dari *performance.*



Gambar 14 Proses Pengujian dengan Melihat *Performance*



Gambar 15 Hasil Luaran *Performance*

Berdasar pada Gambar 15, hasil luaran *performance* dengan menggunakan evaluasi *root mean squared error* menunjukkan nilai 0,675. Hasil RMSE 0,675 dapat disimpulkan hasil model mempunyai nilai kesalahan yang kecil.

4. KESIMPULAN

Data mining dapat digunakan dalam membantu proses pengambilan kebijakan. Salah satu model yang dapat digunakan adalah dengan melakukan prediksi dengan regresi linear. Hasil penelitian dari 33 atribut yang ada pada *datasheet* student_performance.csv, tidak semua atribut mempunyai berpengaruh yang signifikan pada atribut yang menjadi prediktor. Atribut yang sangat kuat mempengaruhi adalah G2 (nilai ujian ke 2) dan G1 (nilai ujian ke 1). Bobot dari G2 sebesar 0,966 dan G1 sebesar 0,892. Atribut lain dengan bobot di atas 0,2 adalah failures (berapa kali pernah tinggal kelas), schoolsup (tambahan pelajaran di luar sekolah) dan absences (kehadiran di kelas) sedangkan atribut yang lain di bawah 0,2. Bobot di bawah 0,2 ini tidak mempunyai pengaruh pada prediktor. Hasil model adalah $y = 0{,}729 - (0{,}024 \times Medu) - (0{,}020 \times Fedu) + (0{,}053 \times failures) - (0{,}077 \times goout) - (0{,}012 \times absences) + (0{,}126 \times G1) + (0{,}862 \times G2)$. Hasil pengujian nilai RMSE adalah 0,675. Semakin kecil nilai RMSE berarti nilai yang diprediksi dekat dengan nilai yang diamati atau observasi. Berdasar hasil RMSE tersebut, hasil model mempunyai nilai kesalahan yang kecil dan model yang dihasilkan dapat direkomendasikan untuk dapat digunakan dalam melakukan prediksi nilai siswa.

DAFTAR PUSTAKA

Arhami, M., & Nasir, M. (2020). *Data Mining - Algoritma dan Implementasi*. Penerbit Andi. https://books.google.co.id/books/about/Data_Mining_Algoritma_dan_Implementasi.html?id=AtcCEAAAQBAJ&redir_esc=y

Ariesanto, A., & Ekka, P. (2020). Data Mining Menggunakan Regresi Linear untuk Prediksi Harga Saham Perusahaan Pelayaran. *Jurnal Aplikasi Pelayaran Dan Kepelabuhanan*, *10*(2), 120. https://doi.org/10.30649/japk.v10i2.83

Bahri, S., Itb, A., & Dahlan, J. (2022). Implementasi Data Mining Untuk Menentukan Minat Siswa Dalam Menentukan Jurusan Pada Perguruan Tinggi. *Jurnal Sistem Informasi (JUSIN)*, *3*(1), 23–33. https://ojs.itb-ad.ac.id/index.php/JUSIN/article/view/1644

Chisholm, A. (2013). *Exploring Data with RapidMiner* (Vol. 1). Packt Publishing. https://www.perlego.com/book/390375/exploring-data-with-rapidminer-pdf

Deepika, K., & Sathyanarayana, N. (2018). Comparison Of Student Academic Performance On Different Educational Datasets Using Different Data Mining Techniques. *International Journal of Computational Engineering Research (IJCER)*, *8*(9), 28–38. http://www.ijceronline.com/papers/Vol8_issue9/Version-2/E0809022838.pdf

N., A. G., Singh, B. P., Sah, B., & Tiwari, D. (2019). Air Quality Index Prediction using Linear Regression. *International Journal of Recent Technology and Engineering (IJRTE)*, *8*(2), 4247–4252. https://doi.org/10.35940/ijrte.B2437.078219

Gaol, I. L. L., Sinurat, S., & Siagian, E. R. (2019). IMPLEMENTASI DATA MINING DENGAN METODE REGRESI LINEAR BERGANDA UNTUK MEMPREDIKSI DATA PERSEDIAAN BUKU PADA PT. YUDHISTIRA GHALIA INDONESIA AREA SUMATERA UTARA. *KOMIK (Konferensi Nasional Teknologi Informasi Dan Komputer)*, *3*(1). https://doi.org/10.30865/komik.v3i1.1579

Hendrian, S. (2018). Algoritma Klasifikasi Data Mining Untuk Memprediksi Siswa Dalam Memperoleh Bantuan Dana Pendidikan. *Faktor Exacta*, *11*(3). https://doi.org/10.30998/faktorexacta.v11i3.2777

Hidayati, N., Suntoro, J., & Setiaji, G. G. (2021). Perbandingan Algoritma Klasifikasi untuk Prediksi Cacat Software dengan Pendekatan CRISP-DM. *Jurnal Sains Dan Informatika*, *7*(2), 117–126. https://doi.org/10.34128/jsi.v7i2.313

Jollyta, D., Ramdhan, W., & Zarlis, M. (2020). Konsep Data Mining Dan Penerapan. In *Konsep Data Mining Dan Penerapan*. Deepublish. https://deepublishstore.com/shop/buku-konsep-data-mining-dan-penerapan/

Kurniatullah, B. D. F., & Pramudi, Y. T. C. (2017). Estimation of Students' Graduation Using Multiple Linear Regression Method. *Journal of Applied Intelligent System*, *2*(1), 29–36.

https://doi.org/10.33633/jais.v2i1.1415

Kurniawan, R. (2016). *Analisis Regresi. Dasar dan Penerapannya dengan R*. Prenada Media. https://prenadamedia.com/product/analisis-regresi-dasar-dan-penerapannya-dengan-r/

Nishadi, A. S. T. (2019). Predicting Heart Diseases In Logistic Regression Of Machine Learning Algorithms By Python Jupyterlab. *International Journal of Advanced Research and Publications*, *3*(8), 69–74. https://www.kaggle.com

Ofori, F., Maina, E., & Gitonga, R. (2020). Using Machine Learning Algorithms to Predict Students' Performance and Improve Learning Outcome: A Literature Based Review. *Journal of Information and Technology*, *4*(1), 2616–3573. https://stratfordjournals.org/journals/index.php/Journal-of-Information-and-Techn/article/view/480

Oyedeji, A. O., Salami, A. M., Folorunsho, O., & Abolade, O. R. (2020). Analysis and Prediction of Student Academic Performance Using Machine Learning. *JITCE (Journal of Information Technology and Computer Engineering)*, *4*(01), 10–15. https://doi.org/10.25077/jitce.4.01.10-15.2020

Prabha, D., Anindhitha, A., Archana, A., & Balaji, N. M. v. (2020). Predicting House Price Values Using Linear Regression with Ridge Regularization Approach. *International Journal of Advanced Science and Technology*, *29*(9s), 5489–5495. http://sersc.org/journals/index.php/IJAST/article/view/18069

Prasetyo, V. R., Lazuardi, H., Mulyono, A. A., & Lauw, C. (2021). Penerapan Aplikasi RapidMiner Untuk Prediksi Nilai Tukar Rupiah Terhadap US Dollar Dengan Metode Linear Regression. *Jurnal Nasional Teknologi Dan Sistem Informasi*, *7*(1), 8–17. https://doi.org/10.25077/TEKNOSI.v7i1.2021.8-17

Putro, M. F., Prayitno, E., Siregar, J., & Muharrom, M. (2021). PENERAPAN DATA MINING DENGAN NAÏVE BAYES UNTUK KLASIFIKASI SISWA SEKOLAH MENENGAH ATAS DALAM PENENTUAN PERGURUAN TINGGI. *Akrab Juara : Jurnal Ilmu-Ilmu Sosial*, *6*(2), 306–312. https://doi.org/10.58487/AKRABJUARA.V6I2.1473

Rahayu, E., Parlina, I., & Siregar, Z. A. (2022). Application of Multiple Linear Regression Algorithm for Motorcycle Sales Estimation. *JOMLAI: Journal of Machine Learning and Artificial Intelligence*, *1*(1), 1–10. https://doi.org/10.55123/jomlai.v1i1.142

Ramadhani, R., & Hendriyani, Y. (2021). Prediksi Prestasi Siswa Berbasis Data Mining Menggunakan Algoritma Decision Tree (Studi Kasus: SMKN 2 Padang). *Voteteknika (Vocational Teknik Elektronika Dan Informatika)*, *9*(3), 11. https://doi.org/10.24036/voteteknika.v9i3.112633

Setiyorini, T., & Asmono, R. T. (2020). IMPLEMENTATION OF GAIN RATIO AND K-NEAREST NEIGHBOR FOR CLASSIFICATION OF STUDENT PERFORMANCE. *Jurnal Pilar Nusa Mandiri*, *16*(1), 19–24. https://doi.org/10.33480/pilar.v16i1.813

Sholeh, M., Suraya, S., & Andayati, D. (2022). Machine Linear untuk Analisis Regresi Linier Biaya Asuransi Kesehatan dengan Menggunakan Python Jupyter Notebook. *JEPIN (Jurnal Edukasi Dan Penelitian Informatika)*, *8*(1), 20–27. https://doi.org/10.26418/JP.V8I1.48822

Sinaga, W. A. L., Sumarno, S., & Sari, I. P. (2022). The Application of Multiple Linear Regression Method for Population Estimation Gunung Malela District. *JOMLAI: Journal of Machine Learning and Artificial Intelligence*, *1*(1), 55–64. https://doi.org/10.55123/jomlai.v1i1.143

Siregar, A. Z. (2021). Implementasi Metode Regresi Linier Berganda Dalam Estimasi Tingkat Pendaftaran Mahasiswa Baru. *Kesatria : Jurnal Penerapan Sistem Informasi (Komputer Dan Manajemen)*, *2*(3), 133–137. https://doi.org/10.30645/KESATRIA.V2I3.73

Sudarsono, B. G., Leo, M. I., Santoso, A., & Hendrawan, F. (2021). ANALISIS DATA MINING DATA NETFLIX MENGGUNAKAN APLIKASI RAPID MINER. *JBASE - Journal of Business and Audit Information Systems*, *4*(1), 13–21. https://doi.org/10.30813/jbase.v4i1.2729

Ünal, F. (2021). Data Mining for Student Performance Prediction in Education. In *Data Mining - Methods, Applications and Systems*. IntechOpen. https://doi.org/10.5772/intechopen.91449

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023, Pp. 22 – 35
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Perbandingan Waktu Respon Aplikasi *Database* NoSQL Elasticsearch dan MongoDB pada Pengujian Operasi CRUD

Theresia Liana Sinaga [(1)], Novrido Charibaldi [(2)*], Nur Heri Cahyana [(3)]
Informatika, Fakultas Teknik Industri, Universtas Pembangunan Nasional "Veteran" Yogyakarta, Yogyakarta
e-mail : sinagatheresia48@gmail.com, {novrido,nur.hericahyana}@upnyk.ac.id.
* Penulis korespondensi.
Artikel ini diajukan 1 November 2022, direvisi 23 Januari 2023, diterima 23 Januari 2023, dan dipublikasikan 30 Januari 2023.

Abstract

Currently, humans live in an era of data oceans, where the amount of data production is increasing from time to time, which is followed by severe challenges in terms of processing, storing, and analyzing data, especially big data. The increase in the number of large data production can affect the speed of access to the database, effectiveness, and speed of response time in the data processing. Relational databases have been the leading model for data storage, analysis, processing, and retrieval for more than forty years. However, due to the increasing need for large-scale data storage, the scalability and performance of a data processing system, as well as the constant growth of the amount of data, another alternative to databases emerged, namely NoSQL technology. Based on previous studies regarding the comparison of response time and database performance, the average concludes that NoSQL performance is more effective and efficient than relational databases. Based on the implementation and testing, it can be concluded that the NoSQL database application MongoDB is proven to be superior in every command of CRUD tested compared to the Elasticsearch NoSQL database application, where in testing the create data command with a JSON file, the MongoDB database application is 42.5 times faster than the Elasticsearch database application. In testing the command to create data into a database containing different amounts of data, the MongoDB database application is 333.9 times faster than the average response time of the Elasticsearch database application. In testing the read command for data in a database containing different amounts of data, the MongoDB database application is 35.5 times faster than the Elasticsearch database application. In testing the update operation of data in a database containing different amounts of data, the MongoDB database application is 9.8 times faster than the Elasticsearch database application. in testing the delete operation of data in a database containing different amounts of data, the MongoDB database application is 58.9 times faster than the Elasticsearch database application.

Keywords: Database, NoSQL, Response Time, CRUD Operation Testing, Elasticsearch, MongoDB

Abstrak

Saat ini manusia hidup di era lautan data, di mana jumlah produksi data semakin bertambah dari waktu ke waktu yang diikuti tantangan berat dalam hal pemrosesan, penyimpanan, dan analisis data, terkhusus pada data besar. Peningkatan jumlah produksi data yang besar dapat mempengaruhi kecepatan akses pada *database*, efektivitas, dan kecepatan waktu respon dalam pemrosesan data. *Database* relasional telah menjadi model terdepan untuk penyimpanan, analisis, pemrosesan, dan pengambilan data selama lebih dari empat puluh tahun. Namun, dikarenakan meningkatnya kebutuhan akan penyimpanan data dengan skala besar, skalabilitas dan kinerja dari suatu sistem pengolahan data, serta pertumbuhan konstan dari jumlah data, maka muncul alternatif lain dari basis data, yaitu teknologi NoSQL. Berdasarkan penelitian-penelitian yang telah dilakukan sebelumnya mengenai perbandingan dari waktu respon dan performa *database*, rata-rata menyimpulkan bahwa performa NoSQL lebih efektif dan efisien dibanding *database* relasional. Berdasarkan implementasi dan pengujian dapat disimpulkan bahwa aplikasi *database* NoSQL MongoDB terbukti memiliki waktu respon yang lebih cepat dalam melakukan perintah operasi CRUD yang diujikan dibandingkan dengan aplikasi *database* NoSQL Elasticsearch, di mana pada pengujian perintah *create data* dengan *file* JSON, aplikasi

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023: 22 – 35

database MongoDB 42,5 kali lebih cepat dibanding aplikasi *database* Elasticsearch. Pada pengujian perintah *create* sebuah data ke dalam *database* yang berisi jumlah data yang berbeda-beda, aplikasi *database* MongoDB 333,9 kali lebih cepat dibanding nilai rata-rata waktu respon aplikasi *database* Elasticsearch. Pada pengujian perintah *read* sebuah data di dalam *database* yang berisi jumlah data yang berbeda-beda, aplikasi *database* MongoDB 35,5 kali lebih cepat dibandingkan aplikasi *database* Elasticsearch. Pada pengujian operasi *update* sebuah data di dalam *database* yang berisi jumlah data yang berbeda-beda, aplikasi *database* MongoDB 9,8 kali lebih cepat dibandingkan aplikasi *database* Elasticsearch. Pada pengujian operasi *delete* sebuah data di dalam *database* yang berisi jumlah data yang berbeda-beda, aplikasi *database* MongoDB 58,9 kali lebih cepat dibandingkan aplikasi *database* Elasticsearch.

Kata Kunci: Basis Data, NoSQL, Waktu Respon, Pengujian Operasi CRUD, Elasticsearch, MongoDB

1. PENDAHULUAN

Saat ini manusia hidup di era lautan data (Ahmed et al., 2018), di mana jumlah produksi data semakin bertambah dari waktu ke waktu yang diikuti tantangan berat dalam hal pemrosesan, penyimpanan, dan analisis data, terkhusus pada data besar (Modhiya, 2021). Peningkatan jumlah produksi data yang besar dapat mempengaruhi kecepatan akses pada *database*, efektivitas, dan kecepatan waktu respon dalam pemrosesan data (Tavares et al., 2020).

Basis data (*database*) merupakan sebuah solusi terhadap penyimpanan data yang memberi ruang untuk menyimpan dan memanipulasi data. *Database* relasional merupakan salah satu tipe *database* yang dapat melakukan penyimpanan, ekstraksi, dan manipulasi data dengan menggunakan bahasa Structured *Query* Languages (SQL) (Amandeep & Singh, 2016). *Database* relasional telah menjadi model terdepan untuk penyimpanan, analisis, pemrosesan, dan pengambilan data selama lebih dari empat puluh tahun. Namun, dikarenakan meningkatnya kebutuhan akan penyimpanan data dengan skala besar, skalabilitas dan kinerja dari suatu sistem pengolahan data, serta pertumbuhan konstan dari jumlah data, maka muncul alternatif lain dari basis data, yaitu teknologi NoSQL (Not Only SQL atau No SQL) (Lourenço et al., 2015).

NoSQL merupakan paradigma baru dalam teknologi *database*, yang merupakan tipe lain dari *database* relasional. NoSQL merupakan sistem manajemen data non-relasional yang tidak memerlukan skema tetap maupun *query* yang kompleks. NoSQL lahir disebabkan oleh pertumbuhan yang pesat pada internet dan laju perkembangan aplikasi web yang semakin kompleks dan memerlukan pengolahan data dalam skala yang besar (Wicaksana, 2017). Perbedaan utama dari *database* relasional dengan *database* NoSQL terdapat pada skema, di mana skema pada *database* relasional secara kaku menentukan bagaimana data harus dimasukkan ke dalam *database* dan menyimpan data dalam format terstruktur yang menggunakan tabel dengan baris dan kolom, sedangkan *database* NoSQL dapat menjadi skema agnostik, yang artinya *database* NoSQL mengakui adanya skema pada *database* lain, tetapi *database* NoSQL tidak menggunakan skema kaku (*schemeless*) pada penyimpanan datanya, sehingga memungkinkan untuk melakukan penyimpanan dan manipulasi data pada data tidak terstruktur dan data semi terstruktur. Terdapat beberapa tipe dari *database* NoSQL, di antaranya yaitu *key-value stores*, *wide-column stores*, *graph database*s, dan *document stores* (Bhaswara et al., 2017).

Ada beberapa tinjauan terhadap penelitian sebelumnya yang membahas perbandingan kecepatan waktu respon dari aplikasi *database*, baik perbandingan waktu respon antara aplikasi *database* relasional dengan aplikasi *database* NoSQL, maupun antar aplikasi *database* yang non relasional. Pertama, Analisis Pemanfaatan NoSQL *Database* Elasticsearch pada Mesin Pencari Tokopedia (Yafet, 2020), penelitian ini dilakukan untuk menganalisis keunggulan dari Elasticsearch yang digunakan sebagai mesin pencarian oleh Tokopedia dibandingkan dengan aplikasi *database* PostgreSQL, yang juga digunakan oleh Tokopedia. Penelitian tersebut menyimpulkan bahwa Elasticsearch terbukti memiliki *availability* dan *flexibility*, serta

Elasticsearch juga dapat melakukan operasi pencarian dan pembacaan data sekitar 92,96% lebih cepat dibandingkan PostgreSQL.

Penelitian kedua berjudul Analisis Perbandingan Performansi Waktu Respon *query* antara MySQL PHP 7.2.27 dan NoSQL MongoDB (Tavares et al., 2020), Kedua *database* tersebut diuji pada Sistem Layanan Aspirasi dan Informasi Kelurahan Oebufu (SELMA) dengan jumlah data 50, 100, 500, 1000, 5000, 10000, dan 100000, pengujian dilakukan dengan Data Manipulation Language (DML), dengan fungsi agregat, dengan fungsi operator, serta *import* dan *export* data. Penelitian itu menyimpulkan bahwa kedua aplikasi *database* tersebut memiliki waktu respon *query* yang berbeda pada setiap model pengujian. MongoDB terbukti lebih unggul di semua model pengujian *query* DML yang terdiri atas proses memasukkan data, membaca data, *update data*, dan hapus data; pengujian fungsi agregat, pengujian fungsi operator penghubung, serta pengujian *import* dan *export* data. Namun, MongoDB memiliki kelemahan pada proses *query* select untuk menampilkan data dengan selisih waktu respon 1,95 detik lebih lambat dari MySQL PHP 7.2.27.

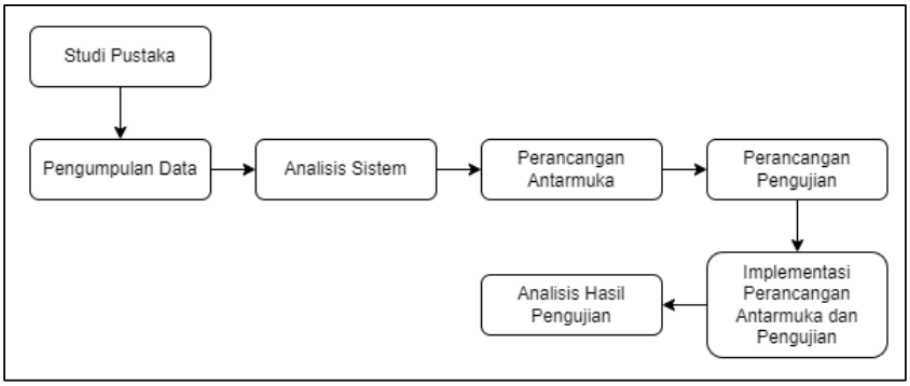
Penelitian ketiga membandingkan kecepatan waktu respon dari aplikasi *database* NoSQL terhadap SQL pada kasus ERP Retail (Bhaswara et al., 2017), Penelitian ini membandingkan *performance*, *flexibility*, dan *scalability* antara *database* relasional (SQL) dengan *database* NoSQL. Setelah didapatkan hasil dari perbandingan kedua *database* tersebut, maka *database* yang unggul diterapkan pada aplikasi *Enterprise Resource Planning* (ERP) *Retail* yang berorientasi *multitenancy*. Dengan harapan, aplikasi ERP *Retail* dapat memiliki kinerja yang optimal untuk menyimpan data. Penelitian ini mendapatkan kesimpulan bahwa aplikasi MongoDB terbukti unggul dalam proses transaksi pada pengujian *Create*, *Read*, *Update*, *Delete* (CRUD) dibandingkan dengan MySQL dengan jumlah selisih waktu eksekusi pada *create data* 0,167 detik, *read data* 0,009 detik, *update data* 0,44 detik, dan *delete data* 0,056 detik. Selain itu, MongoDB mampu memenuhi kebutuhan dari aplikasi ERP Retail yang mendukung perbedaan skema dari setiap *tenant*. Namun, MongoDB lemah pada proses transaksi agregasi dengan jumlah selisih waktu eksekusi lebih lambat 0,039 detik.

Penelitian keempat dengan judul *An Analysis on the Comparison of the Performance and Configuration Features of Big Data Tools Solr and Elasticsearch* (Aydoğan et al., 2016). Penelitian ini membandingkan kinerja dari aplikasi *database* NoSQL Solr dan aplikasi *database* NoSQL Elasticsearch. Variabel yang dibandingkan adalah waktu operasi *query*, kemudahan dan kesulitan dalam penggunaan *database*, bentuk konfigurasi, dan arsitektur. Kesimpulan yang didapatkan pada penelitian ini adalah Elasticsearch mendukung lebih banyak bahasa pemrograman daripada Solr, dalam hal pengindeksan, Elasticsearch memiliki kinerja yang lebih baik jika data lebih kecil, sedangkan Solr memiliki kinerja yang lebih baik untuk data yang besar. Waktu pemrosesan *query* bervariasi tergantung pada jenis data yang digunakan.

Berdasarkan tinjauan penelitian mengenai perbandingan dari waktu respon *database* (Gunawan, 2018; Halimi & Sudarmanto, 2021), ternyata kinerja NoSQL lebih baik dibanding *database* relasional. Namun, penelitian yang menbandingakan waktu respon antara aplikasi *database* NoSQL MongoDB dan aplikasi *database* NoSQL Elasticsearch masih minim dilakukan. Beberapa perusahaan besar yang menerapkan MongoDB dalam pengolahan datanya, seperti SEGA HARDlight yang bergerak di bidang *mobile game development*, Medtronic yang merupakan perusahaan penyedia alat-alat medis di Amerika Serikat, EA yang merupakan *developer game* yang bertema olahraga (pada *game* EA FIFA Online 3 dikembangkan menggunakan MongoDB). Sedangkan beberapa perusahaan di Indonesia yang menerapkan Elasticsearch yaitu Tokopedia dan BCA, serta perusahaan enterprise seperti Facebook, Airbus, dan IEEE (Renaldi et al., 2020). Penelitian ini membandingkan waktu respon pada operasi *Create*, *Read*, *Update*, *Delete* (CRUD) data dari aplikasi *database* NoSQL Elasticsearch dan MongoDB, sehingga dapat diketahui aplikasi *database* NoSQL manakah yang bekerja lebih cepat. Pengujian pada penelitian ini dilakukan sebanyak 3 kali untuk tiap operasi CRUD data dalam beragam *record data* (100, 1000, dan 8900), sedangkan waktu respon dari setiap operasi disajikan dalam bentuk tabel dan grafik.

2. METODE PENELITIAN

Metodologi pada penelitian ini menggunakan metodologi penelitian kuantitatif. Adapun yang dimaksud dengan penelitian kuantitatif adalah penelitian yang erat kaitannya dengan filsafat positivisme. Penelitian ini lebih berfokus kepada pemecahan masalah yang erat kaitannya dengan angka-angka dan statistik. Berikut tahapan penelitian ditunjukkan pada Gambar 1.



Gambar 1 Tahapan Penelitian

2.1 Studi Pustaka

Studi pustaka yang dilakukan dalam penelitian ini yaitu dengan membaca artikel, buku, jurnal, dan tesis yang berkaitan dengan topik aplikasi *database* NoSQL Elasticsearch dan MongoDB, serta penelitian yang berfokus pada perbandingan kecepatan waktu respon dari aplikasi *database*.

2.2 Pengumpulan Data

Data yang digunakan dalam penelitian ini berupa data *dummy* yang bersumber dari *website* data *world* berupa *countries*, *states*, dan *cities*. Data yang didapatkan berupa *file* JSON.

2.3 Analisis Sistem

Tahapan berikutnya merupakan analisis terhadap batasan lingkup sistem secara general dan pengujian yang dilakukan dengan mengukur kecepatan waktu respon dari aplikasi *database* NoSQL Elasticsearch dan MongoDB melibatkan perangkat lunak (*software*), perangkat keras (*hardware*), serta data digunakan untuk mengukur waktu respon saat melakukan operasi CRUD pada Elasticsearch dan MongoDB.

2.3.1 Kebutuhan Perangkat Keras dan Perangkat Lunak

Kebutuhan perangkat keras dan perangkat lunak merupakan kebutuhan terkait spesifikasi dalam penggunaan sistem yang akan dibangun. Spesifikasi yang dibutuhkan terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*).

Tabel 1 Kebutuhan Perangkat Keras

No.	Perangkat Keras	Keterangan
1	Laptop	LENOVO Ideapad Slim 3-14ada05 Business Black Series
2	*Processor*	AMD Athlon Silver 3050U *with* Radeon Graphics (2 CPUs)
3	RAM	8192 MB
4	*Storage*	256 GB SSD
5	Perangkat *input* dan *output*	*Keyboard, mouse, monitor, printer*
6	Koneksi internet	Wi-Fi

Tabel 2 Kebutuhan Perangkat Lunak

No.	Perangkat Lunak	Keterangan
1	Windows 10	Sistem operasi
2	Python versi 3,8 ke atas	Bahasa pemrograman
3	Flask	*Micro framework* Python untuk pengembangan aplikasi berbasis web
4	Elasticsearch versi 8 ke atas	*Database* NoSQL
5	MongoDB versi 5 ke atas	*Database* NoSQL
6	Google Chrome	*Browser*

2.3.2 Kebutuhan Penelitian

1) Aplikasi *database* NoSQL Elasticsearch

Elasticsearch adalah salah aplikasi *database* NoSQL yang dikembangkan menggunakan bahasa pemrograman Java menggunakan Lucene Library oleh Shay Banon pada tahun 2010. Elasticsearch merupakan *hybrid database*/*search tool open-source* yang berorientasi pada dokumen dan memiliki skalabititas yang tinggi. Elasticsearch dibangun oleh Apache Lucene sebagai search engine *database* yang memiliki *query low level* dan penyimpanan dokumen tanpa skema dalam format JSON, dan dapat lebih mudah untuk digunakan karena dapat diakses dari antarmuka layanan web RESTful API. Kecepatan dan skalabilitas yang dimiliki Elasticsearch, serta kemampuan untuk mengindeks banyak jenis konten dapat diaplikasi pada sejumlah kasus penggunaan, seperti aplikasi pencarian, *website* pencarian, analisis keamanan, analisis bisnis, analisis dan visualisasi data geospasial, pemantauan kinerja aplikasi, dan lain-lain (Aydoğan et al., 2016). Terdapat beberapa konsep dasar untuk memahami struktur Elasticsearch, di antaranya yaitu (Yafet, 2020):

a) *Node*
   Node merupakan sebuah instance Elasticsearch yang sedang bekerja. Banyaknya *node* tergantung pada kemampuan sumber daya fisiknya seperti Random Access Memory (RAM), memory, dan kemampuan untuk melakukan pemrosesan.
b) *Cluster*
   Cluster merupakan kumpulan dari beberapa *node* yang bekerja sama untuk membaca atau menulis ke indeks. Setiap *node* pada sebuah *cluster* berkontribusi untuk melakukan pengindeksan dan pencarian data.
c) *Document*
   Document (dokumen) merupakan kumpulan unit dasar informasi atau data dalam format JSON. Misalnya pengguna menyimpan data seorang siswa, maka pengguna menambahkan satu objek yang memiliki nama, umur, dan properti lainnya. Dokumen memiliki ID unik yang diberikan oleh pengguna saat menambahkannya ke *index*. Dokumen-dokumen tersebut disimpan pada *index*.
d) *Index*
   Index merupakan kumpulan dari dokumen yang memiliki sifat serupa, misalnya *index* untuk data pesanan, data produk, data pelanggan, dan lain-lain.
e) *Shard*
   Elasticsearch memiliki kemampuan untuk memotong *index* menjadi beberapa *shard*/partisi yang disebut dengan proses *shard*/*sharding*. Secara *default*, Elasticsearch melakukan *sharding* setelah terbuatnya sebuah *index*. Dengan adanya *sharding* pada *index*, maka dapat memudahkan pendistribusian *index* ke beberapa *node* dalam sebuah *cluster* di Elasticsearch.
f) *Replication*
   Elasticsearch memiliki fitur replikasi untuk menduplikasi *index* yang telah melewati proses *sharding*. Secara *default*, setelah *index* dibuat, Elasticsearch akan membuat replikasi dari setiap *shard index*. Hal ini berguna untuk mencegah kehilangan data dokumen *index* karena kegagalan jaringan yang tidak terduga. Replika *shard* tidak pernah ditempatkan pada *node* yang sama dengan *shard* primer, sehingga dapat meningkatkan *throughput* dan kinerja

pencarian. Saat membuat *index*, pengguna dapat memilih berapa jumlah *shard* dan replikanya.

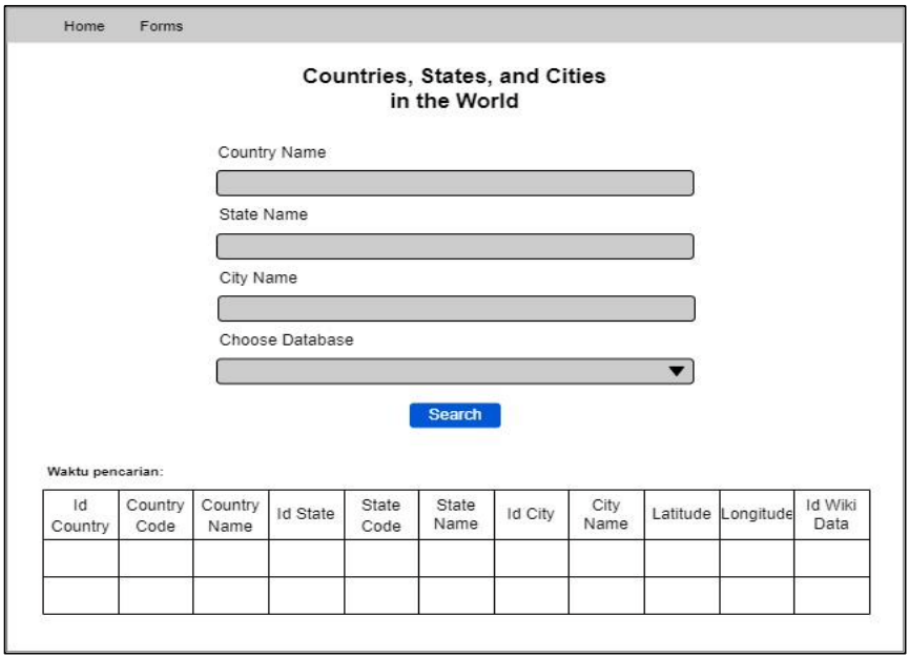
2) Aplikasi *database* NoSQL MongoDB

MongoDB merupakan aplikasi *database* NoSQL yang menyimpan datanya dalam dokumen dengan format JSON. MongoDB merupakan aplikasi *database* yang dikembangkan oleh Mongo, Inc. MongoDB tidak perlu mendefinisikan struktur seperti atribut dan tipe data terlebih dahulu untuk melakukan penyimpanan data. Karena MongoDB menggunakan skema dinamis, sehingga model aplikasi *database* seperti ini dapat membantu untuk menyimpan *array* dan struktur lain yang lebih kompleks dengan mudah (Damodaran et al., 2016). Ada tiga komponen penting pada aplikasi *database* NoSQL, yaitu (Chauhan, 2019):

a) *Database*, merupakan wadah yang memiliki struktur penyimpanan yang disebut *collection*.
b) *Collection*, merupakan kumpulan dokumen-dokumen (*collection* ini setara dengan tabel pada sistem *database* relasional).
c) *Document*, merupakan unit data satuan terkecil dalam MongoDB yang berisikan baris-baris data yang berupa struktur pasangan *key-value* yang berfungsi saling memasangkan informasi. *Document* dapat dianalogikan seperti *record* pada *database* relasional (Kriestanto & Arnado, 2017).

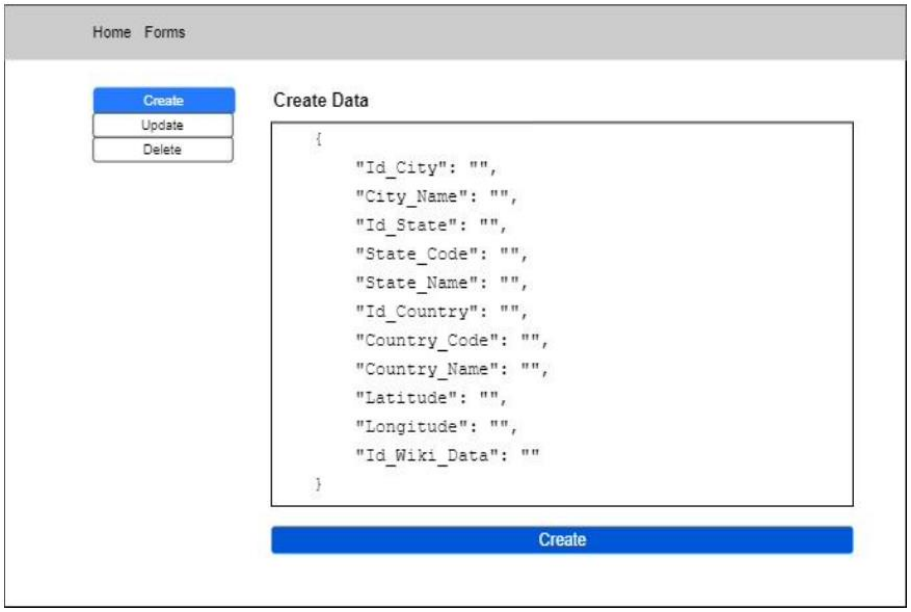
2.4 Perancangan Antarmuka

Rancangan antarmuka pengguna pada penelitian ini berupa rancangan halaman aplikasi web sederhana. Rancangan antarmuka terdiri atas halaman *home* dan halaman *forms*. Halaman *home* digunakan untuk melakukan pencarian data, menampilkan seluruh data dalam sebuah tabel, dan menampilan waktu dari pencarian data. Sedangkan halaman *forms*, merupakan halaman bagi pengguna untuk melakukan *create data*, *update data*, dan *delete data*, sedangkan waktu respon setiap perintah ditampilkan setelah instruksi *create*, *update*, dan *delete data*. Tampilan dari rancangan antarmuka halaman *home* dapat dilihat pada Gambar 2.



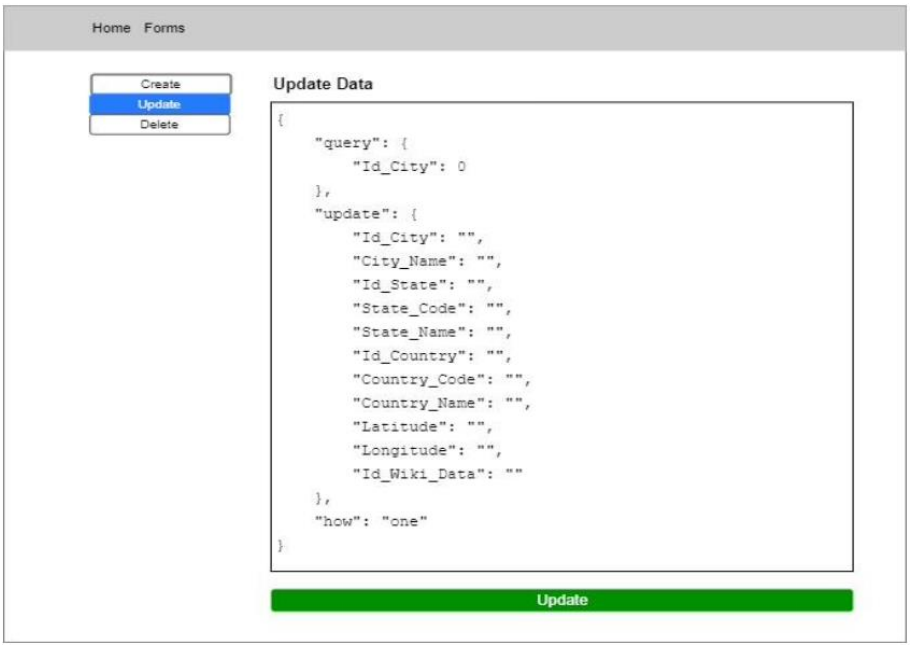
Gambar 2 Rancangan Tampilan Antarmuka Halaman *Home*

Gambar 3 merupakan rancangan tampilan antarmuka halaman *forms* yaitu *create data* yang digunakan untuk memasukkan data baru.



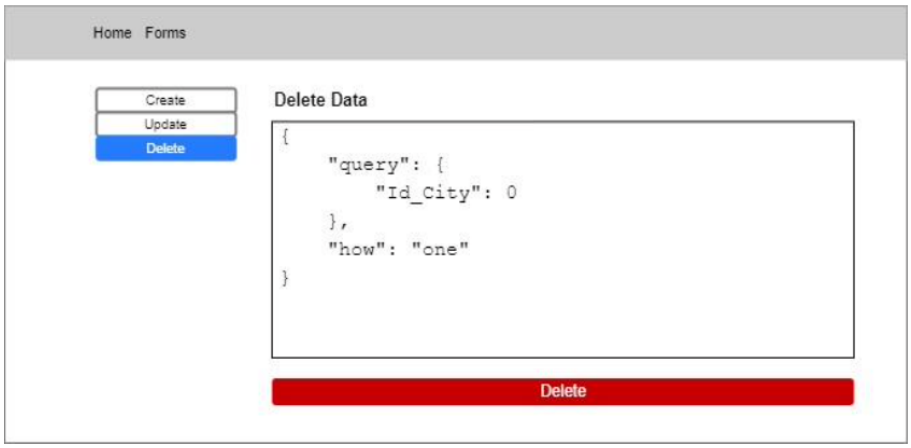
Gambar 3 Rancangan Tampilan Antarmuka Halaman *Forms Create Data*

Gambar 4 merupakan rancangan tampilan antarmuka halaman *forms* untuk proses *update data*.



Gambar 4 Rancangan Tampilan Antarmuka Halaman *Forms Update Data*

Gambar 5 merupakan rancangan tampilan antarmuka halaman *forms* untuk proses *delete data* yang dikehendaki.



Gambar 5 Rancangan Tampilan Antarmuka Halaman *Forms Delete Data*

2.5 Perancangan Pengujian

Pengujian dilakukan dengan mengukur waktu respon saat menjalankan operasi CRUD dari awal *query* berjalan sampai dengan mendapatkan hasil *query*. Pengujian dilakukan pada aplikasi web sederhana. Berikut merupakan skenario dari pengujian yang dilakukan.

2.5.1 Pengujian kecepatan waktu respon *create data* (menambahkan data)

Pada pengujian ini, masing-masing aplikasi *database* NoSQL Elasticsearch dan MongoDB diuji dengan melakukan *query* penambahan data baru. Terdapat dua cara dalam memasukkan data, yang pertama dengan memasukkan data melalui *file* JSON yang berisi lebih dari satu data di dalamnya. Sedangkan untuk cara kedua, yaitu memasukkan data melalui halaman *forms create data* pada aplikasi web sederhana. Setiap *field* dan tipe data pada data yang ditambahkan diharuskan sama, serta sesuai dengan skema masing-masing aplikasi *database*. Setelah proses *insert file*/*create data* selesai, maka waktu respon dari kedua aplikasi *database* ditampilkan.

2.5.2 Pengujian kecepatan waktu respon *read data* (mencari data)

Pada pengujian ini, masing-masing *database* NoSQL Elasticsearch dan MongoDB diuji dengan melakukan *query* pencarian data dan mengembalikan hasil dari pencarian data pada aplikasi web sederhana dan menapilkan waktu pencarian. Setelah proses *read data* selesai, maka waktu respon dari kedua aplikasi *database* ditampilkan.

2.5.3 Pengujian kecepatan waktu respon *update data* (mengubah data)

Pada pengujian ini, masing-masing aplikasi *database* NoSQL Elasticsearch dan MongoDB diuji dengan operasi *query* pencarian data dan mengembalikan hasil dari pencarian data ke aplikasi web sederhana dan menampilkan waktu pencarian. Setelah proses *read data* selesai, maka waktu respon dari kedua aplikasi *database* ditampilkan.

2.5.4 Pengujian waktu respon *delete data* (menghapus data)

Pada pengujian ini, masing-masing aplikasi *database* NoSQL Elasticsearch dan MongoDB diuji dengan melakukan *query* hapus data yang diinginkan pada masing-masing aplikasi *database*. Setelah proses *delete data* selesai, maka waktu respon dari kedua aplikasi *database* ditampilkan.

3. HASIL DAN PEMBAHASAN

Pengujian yang telah dilakukan untuk mengetahui perbandingan kecepatan waktu respon dari perintah *create*, *read*, *update*, dan *delete data* pada aplikasi *database* NoSQL MongoDB dan Elasticsearch dijabarkan dalam bentuk tabel dan grafik. Agar dapat diketahui setiap perintah yang

memiliki nilai waktu respon rendah pada masing-masing pengujian data dianggap lebih cepat dibandingkan pengujian yang memiliki nilai waktu respon lambat. Berikut penjabaran perbandingan waktu respon yang berupa tabel dan grafik untuk setiap perintah perintah *create*, *read*, *update*, dan *delete data* yang telah dilakukan.

3.1 Pengujian waktu respon perintah *create data* dengan *file* JSON

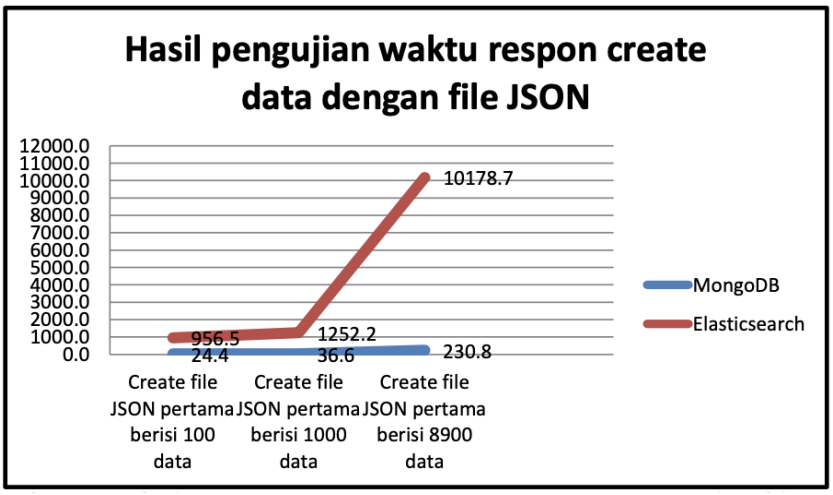
Pengujian kecepatan waktu respon saat memasukkan data dengan *file* JSON yang berisi lebih dari satu data *countries*, *states*, dan *cities* dilakukan sebanyak tiga kali proses *create data* dengan data *file* JSON. *File* pertama berisi 100 data *countries*, *states*, dan *cities*; *file* kedua berisi 1000 data *countries*, *states*, dan *cities*; dan *file* ketiga berisi 8900 data *countries*, *states*, dan *cities*; sehingga total data *countries*, *states*, dan *cities* yang dimasukkan yaitu berjumlah 10000 data.

Tabel 3 Tabel Perbandingan Waktu Respon Perintah *Create Data* dengan *File JSON*

Database / **Jumlah Data**	***Create file* JSON pertama berisi 100 data**	***Create file* JSON pertama berisi 1000 data**	***Create file* JSON pertama berisi 8900 data**
MongoDB	24,4 ms	36,6 ms	230,8 ms
Elasticsearch	956,5 ms	1252,2 ms	10178,7 ms

Saat melakukan *create data file* JSON pertama yang berisi 100 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 24,4 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 956,5 ms. Pada saat melakukan *create data file* JSON kedua yang berisi 1000 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 36,6 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 1252,2 ms. Pada saat melakukan *create data file* JSON ketiga yang berisi 8900 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 230,8 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 10178,7 ms.

Gambar 6 merupakan data hasil pengujian yang disajikan dalam bentuk grafik. Pada pengujian *create data* dengan *file* JSON dapat dibandingkan bahwa aplikasi *database* MongoDB memiliki waktu respon lebih cepat dibandingkan dengan Elasticsearch, nilai rata-rata waktu respon aplikasi *database* MongoDB yaitu 97,2 ms, sedangkan nilai rata-rata waktu respon aplikasi *database* Elasticsearch yaitu 4129,1 ms. Dengan kata lain, aplikasi *database* MongoDB 42,5 kali lebih cepat dibanding aplikasi *database* Elasticsearch.



Gambar 6 Grafik Hasil Pengujian Waktu Respon *Create Data* dengan *File* JSON

3.2 Pengujian waktu respon perintah *create data* melalui *forms create data*

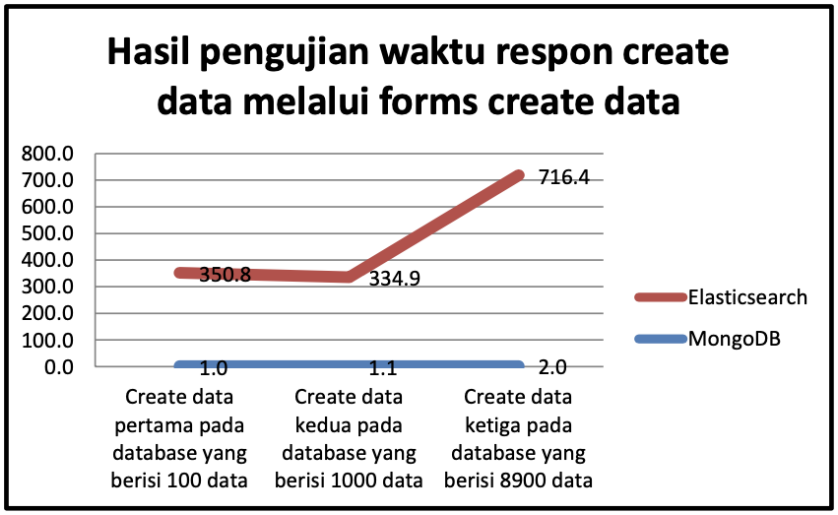
Pengujian kecepatan waktu respon *create data* dengan cara kedua yaitu memasukkan data baru melalui *forms create data* yang terdapat pada aplikasi web sederhana. Cara ini hanya dilakukan dengan memasukkan satu data baru/satu dokumen baru. Pengujian pertama, memasukkan satu data baru pada *database* yang telah menyimpan 100 data. Pengujian kedua, memasukkan satu data baru pada *database* yang telah menyimpan 1000 data. Pengujian ketiga, memasukkan satu data baru pada *database* yang menyimpan 8900 data.

Tabel 4 Tabel Perbandingan Waktu Respon Perintah *Create Data* Melalui *Forms Create Data*

Database / **Jumlah Data**	***Create file* JSON pertama berisi 100 data**	***Create file* JSON pertama berisi 1000 data**	***Create file* JSON pertama berisi 8900 data**
MongoDB	1,0 ms	1,1 ms	2,0 ms
Elasticsearch	334,9 ms	350,8 ms	716,4 ms

Saat melakukan *create data* baru pada *database* yang telah menyimpan 100 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 1,0 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 334,9 ms. *Create data* baru pada *database* yang telah menyimpan 1000 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 1,1 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 350,8 ms. *Create data* baru pada *database* yang telah menyimpan 8900 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 2,0 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 716,4 ms.

Gambar 7 merupakan hasil pengujian yang disajikan dalam bentuk grafik. Pengujian *create data* melalui *forms create data* dapat dinyatakan bahwa aplikasi *database* MongoDB masih memiliki waktu respon lebih cepat dibandingkan dengan Elasticsearch, nilai rata-rata waktu respon aplikasi *database* MongoDB yaitu 1,4 ms, sedangkan nilai rata-rata waktu respon aplikasi *database* Elasticsearch yaitu 467,4 ms. Dengan kata lain, nilai rata-rata waktu respon aplikasi *database* MongoDB 333,9 kali lebih cepat dibanding nilai rata-rata waktu respon aplikasi *database* Elasticsearch.



Gambar 7 Grafik Hasil Pengujian Waktu Respon *Create Data* Melalui *Forms Create Data*

3.3 Pengujian waktu respon perintah *read data*/pencarian data

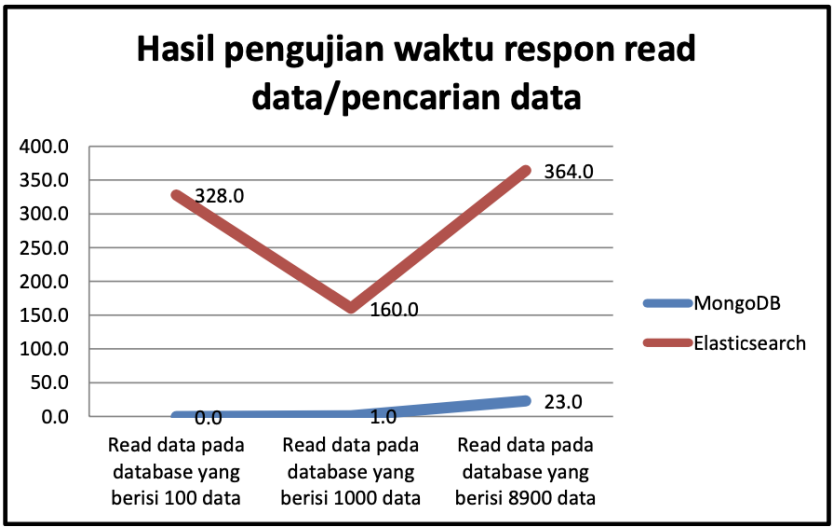
Pengujian waktu respon proses pencarian data dilakukan sebanyak tiga kali, pengujian pertama mencari suatu data yang tersimpan dalam *database* berisi 100 data, pengujian kedua mencari suatu data yang tersimpan dalam *database* berisi 1000 data, dan pengujian ketiga mencari suatu data yang tersimpan dalam *database* berisi 8900 data.

Tabel 5 Tabel Perbandingan Waktu Respon Perintah *Read Data*

Database / Jumlah Data	*Read data* pada *database* berisi 100 data	*Read data* pada *database* berisi 1000 data	*Read data* pada *database* berisi 8900 data
MongoDB	0,01 ms	1 ms	23 ms
Elasticsearch	328 ms	160 ms	364 ms

Operasi *read data*/pencarian suatu data dalam *database* yang berisi 100 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 0,01 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 328 ms. Operasi *read data*/pencarian suatu data pada *database* yang telah menyimpan 1000 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 1 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 160 ms. Pada saat melakukan *read data*/pencarian suatu data pada *database* yang telah menyimpan 8900 data, waktu respon yang dibutuhkan aplikasi *database* MongoDB adalah 23 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 364 ms.

Gambar 8 adalah data hasil pengujian yang disajilkan dalam bentuk grafik. Operasi pengujian *read data*/pencarian data dapat dinyatakan bahwa aplikasi *database* MongoDB masih memiliki waktu respon lebih cepat dibandingkan dengan Elasticsearch, nilai rata-rata waktu respon aplikasi *database* MongoDB yaitu 8,0 ms, sedangkan nilai rata-rata waktu respon aplikasi *database* Elasticsearch yaitu 284 ms. Dengan kata lain, waktu respon yang diperlukan aplikasi *database* MongoDB 35,5 kali lebih cepat dibandingkan aplikasi *database* Elasticsearch.



Gambar 8 Grafik Hasil Pengujian Waktu Respon *Read Data*/Pencarian Data

3.4 Pengujian waktu respon perintah *update data* melalui *forms update data*

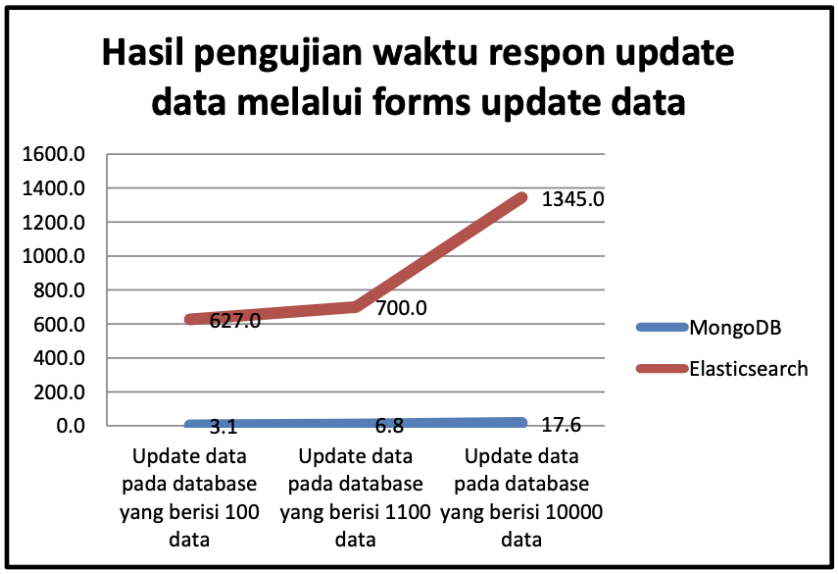
Pengujian waktu respon perintah *update data* melalui *forms update data* dilakukan sebanyak tiga kali, pengujian pertama adalah operasi *update* suatu data di dalam *database* berisi 100 data,

pengujian kedua adalah operasi *update* suatu data di dalam *database* berisi 1000 data, dan pengujian ketiga adalah operasi *update* suatu data di dalam *database* berisi 8900 data.

Tabel 6 Tabel Perbandingan Waktu Respon Perintah *Update Data* Melalui *Forms Update Data*

Database / **Jumlah Data**	***Update data* pada *database* berisi 100 data**	***Update data* pada *database* berisi 1000 data**	***Update data* pada *database* berisi 8900 data**
MongoDB	3,1 ms	6,8 ms	17,6 ms
Elasticsearch	627 ms	700 ms	1345 ms

Waktu respon pengujian pertama yang dibutuhkan aplikasi *database* MongoDB adalah 3,1 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 627 ms. Waktu respon pengujian kedua yang dibutuhkan aplikasi *database* MongoDB adalah 6,8 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 700 ms. Waktu respon pengujian ketiga yang dibutuhkan aplikasi *database* MongoDB adalah 17,6 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 1345 ms. Gambar 9 adalah data hasil pengujian operasi *update data* yang disajikan dalam bentuk grafik. Pengujian *update data* melalui *forms update data* dapat dinyatakan bahwa aplikasi *database* MongoDB masih memiliki waktu respon lebih cepat dibandingkan dengan Elasticsearch, nilai rata-rata waktu respon aplikasi *database* MongoDB yaitu 91,2 ms, sedangkan nilai rata-rata waktu respon aplikasi *database* Elasticsearch yaitu 890,6 ms. Dengan kata lain, waktu respon yang diperlukan aplikasi *database* MongoDB 9,8 kali lebih cepat dibandingkan aplikasi *database* Elasticsearch.



Gambar 9 Grafik Hasil Pengujian Waktu Respon *Update Data* Melalui *Forms Update Data*

3.5 Pengujian waktu respon perintah *delete data* melalui *forms delete data*

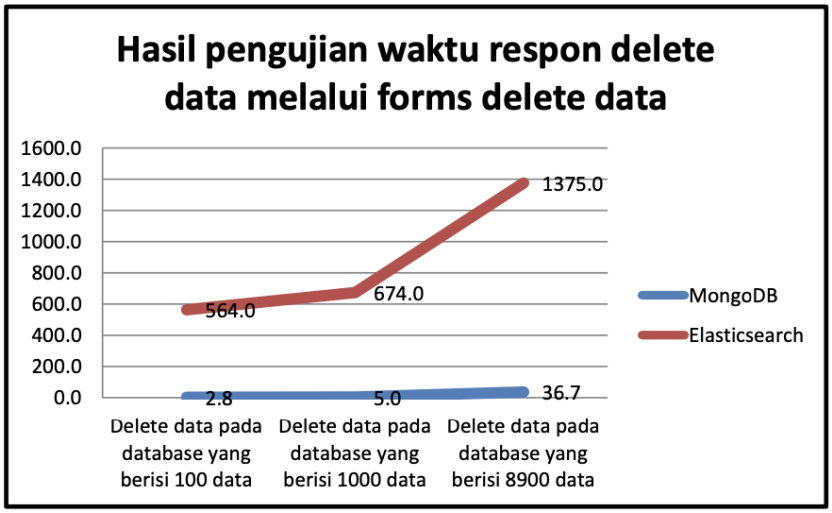
Pengujian waktu respon perintah *delete data* melalui halaman *forms delete data* dilakukan sebanyak tiga kali, pengujian pertama yaitu operasi *delete* suatu data di dalam *database* yang berisi 100 data, pengujian kedua yaitu operasi *delete* suatu data di dalam *database* yang berisi 1000 data, dan pengujian ketiga yaitu operasi *delete* suatu data di dalam *database* yang berisi 8900 data.

Tabel 7 Tabel Perbandingan Waktu Respon Perintah *Delete Data* Melalui *Forms Delete Data*

Database / Jumlah Data	*Delete data* pada *database* berisi 100 data	*Delete data* pada *database* berisi 1000 data	*Delete data* pada *database* berisi 8900 data
MongoDB	2,8 ms	5,0 ms	36,7 ms
Elasticsearch	564 ms	674 ms	1375 ms

Waktu respon pengujian pertama yang dibutuhkan aplikasi *database* MongoDB adalah 2,8 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 564 ms. Waktu respon pengujian kedua yang dibutuhkan aplikasi *database* MongoDB adalah 5,0 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 674 ms. Waktu respon pengujian ketiga yang dibutuhkan aplikasi *database* MongoDB adalah 36,7 ms, sedangkan waktu respon yang dibutuhkan aplikasi *database* Elasticsearch adalah 1375 ms.

Gambar 10 merupakan data hasil pengujian yang disajikan dalam bentuk grafik. Pengujian *delete data* melalui *forms delete data* dapat dinyatakan bahwa aplikasi *database* MongoDB masih memiliki waktu respon lebih cepat dibandingkan dengan Elasticsearch, nilai rata-rata waktu respon aplikasi *database* MongoDB yaitu 14,8 ms, sedangkan nilai rata-rata waktu respon aplikasi *database* Elasticsearch yaitu 871 ms. Dengan kata lain, waktu respon yang diperlukan aplikasi *database* MongoDB 58,9 kali lebih cepat dibandingkan aplikasi *database* Elasticsearch.



Gambar 10 Grafik Hasil Pengujian Waktu Respon *Delete Data* Melalui *Forms Delete Data*

4. KESIMPULAN

Berdasarkan tabel dan grafik hasil perbandingan pengujian waktu respon, dapat disimpulkan bahwa, waktu respon aplikasi *database* NoSQL MongoDB membutuhkan waktu yang lebih singkat di semua perintah *create*, *read*, *update*, dan *delete* (CRUD) data dibandingkan waktu respon aplikasi *database* NoSQL Elasticsearch.

Pada pengujian perintah *create data* dengan *file* JSON, aplikasi *database* MongoDB 42,5 kali lebih cepat dibanding aplikasi *database* Elasticsearch. Pada pengujian perintah *create* sebuah data ke dalam *database* yang berisi jumlah data yang berbeda-beda, aplikasi *database* MongoDB 333,9 kali lebih cepat dibanding nilai rata-rata waktu respon aplikasi *database* Elasticsearch. Pada pengujian perintah *read* sebuah data di dalam *database* yang berisi jumlah data yang berbeda-beda, aplikasi *database* MongoDB 35,5 kali lebih cepat dibandingkan aplikasi *database* Elasticsearch. Pada pengujian operasi *update* sebuah data di dalam *database* yang berisi jumlah data yang berbeda-beda, aplikasi *database* MongoDB 9,8 kali lebih cepat dibandingkan aplikasi

database Elasticsearch. pada pengujian operasi *delete* sebuah data di dalam *database* yang berisi jumlah data yang berbeda-beda, aplikasi *database* MongoDB 58,9 kali lebih cepat dibandingkan aplikasi *database* Elasticsearch.

Sehingga, pada penelitian ini dapat disimpulkan bahwa aplikasi *database* MongoDB memiliki waktu respon yang lebih cepat dibandingkan aplikasi *database* Elasticsearch berdasarkan waktu respon saat operasikan perintah *create*, *read*, *update*, dan *delete data* menggunakan aplikasi web sederhana.

DAFTAR PUSTAKA

Ahmed, M. R., Khatun, M. A., Ali, M. A., & Sundaraj, K. (2018). A literature review on NoSQL database for big data processing. *International Journal of Engineering & Technology*, *7*(2), 902–906. https://doi.org/10.14419/IJET.V7I2.12113

Aydoğan, T., İlkuçar, M., & AKCA, M. A. (2016). An Analysis on the Comparison of the Performance and Configuration Features of Big Data Tools Solr and Elasticsearch. *International Journal of Intelligent Systems and Applications in Engineering*, *4*(Special Issue-1), 8–12. https://doi.org/10.18201/ijisae.271328

Amandeep, K., & Singh, D. K. (2016). Performance Evaluation For Crud Operations In NoSQL Databases. *I-Manager's Journal on Cloud Computing*, *3*(2), 1. https://doi.org/10.26634/jcc.3.2.8164

Bhaswara, F. A., Sarno, R., & Sunaryono, D. (2017). Perbandingan Kemampuan Database NoSQL dan SQL dalam Kasus ERP Retail. *Jurnal Teknik ITS*, *6*(2), A511–A514. https://doi.org/10.12962/J23373539.V6I2.24031

Chauhan, A. (2019). A Review on Various Aspects of MongoDb Databases. *International Journal of Engineering Research & Technology*, *8*(5). https://doi.org/10.17577/IJERTV8IS050031

Damodaran, D., Salim, S., & Vargese, S. M. (2016). Performance Evaluation of MySQL and MongoDB Databases. *International Journal on Cybernetics & Informatics (IJCI)*, *5*(2). https://doi.org/10.5121/ijci.2016.5241

Gunawan, R. (2018). Pengukuran Query Respon Time pada NoSQL Database Berbasis Document Stored. *Jurnal Siliwangi Seri Sains Dan Teknologi*, *4*(2). https://jurnal.unsil.ac.id/index.php/jssainstek/article/view/609

Halimi, A., & Sudarmanto, A. (2021). Analisis Perbandingan Kinerja Waktu Respon MySQL 8.0 dan NoSQL MongoDB Mengunakan REST API NodeJS pada Studi Kasus Kelas Online. *Jurnal Informatika Wicida*, *10*(1), 26–33. https://doi.org/10.46984/INF-WCD.1185

Kriestanto, D., & Arnado, A. B. (2017). Implementasi Website Pencarian Kos dengan NoSQL. *JIKO (Jurnal Informatika Dan Komputer)*, *2*(2), 103–108. https://doi.org/10.26798/JIKO.V2I2.66

Lourenço, J. R., Cabral, B., Carreiro, P., Vieira, M., & Bernardino, J. (2015). Choosing the right NoSQL database for the job: a quality attribute evaluation. *Journal of Big Data*, *2*(1), 1–26. https://doi.org/10.1186/S40537-015-0025-0/TABLES/2

Modhiya, K. (2021). Introduction to DBMS, RDBMS, and NoSQL Database: NoSQL Database Challenges. *SSRN Electronic Journal*. https://doi.org/10.2139/SSRN.3798989

Renaldi, R., Santoso, B. C., Natasya, Y., willian, steven, & alfando, fladianand. (2020). Tinjauan Pustaka Sistematis terhadap Basis Data MongoDB. *Jurnal Inovasi Informatika*, *5*(2), 132–142. https://doi.org/10.51170/JII.V5I2.79

Tavares, O. M. I., Rangkoly, S. M., Desy Bawan, S. B., Utami, E., & Mustafa, M. S. (2020). Analisis Perbandingan Performansi Waktu Respons Kueri antara MySQL PHP 7.2.27 dan NoSQL MongoDB. *Jurnal Teknologi Informasi*, *4*(2), 303–313. https://doi.org/10.36294/jurti.v4i2.1695

Wicaksana, I. G. N. A. (2017). *Sinkronisasi Basis Data Sql dengan Basis Data Nosql Menggunakan Data Adapter dengan Pendekatan Query Direct Access*. Institut Teknologi Sepuluh Nopember.

Yafet, T. T. (2020). *Analisis Pemanfaatan NoSQL Database Elasticsearch pada Mesin Pencarian Tokopedia* [Universitas Atma Jaya]. http://e-journal.uajy.ac.id/23338/

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023, Pp. 36 – 49
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Penentuan Kelayakan Masyarakat Miskin Penerima Bantuan Menggunakan Metode Naïve Bayes (Studi Kasus: Kabupaten Penajam Paser Utara)

Nur Madia [(1)], Anindita Septiarini [(2)*], Heliza Rahmania Hatta [(3)], Hamdani Hamdani [(3)], Masna Wati [(3)]
Informatika, Fakultas Teknik, Universitas Mulawarman, Samarinda
e-mail : nmndia18@gmail.com, {anindita,heliza,hamdani}@unmul.ac.id, masnawati@fkti.unmul.ac.id.
* Penulis korespondensi.
Artikel ini diajukan 7 November 2022, direvisi 27 Desember 2022, diterima 2 Januari 2023, dan dipublikasikan 30 Januari 2023.

Abstract

Contents Poverty is the inability to meet the necessities of life, such as food, clothing, and shelter. The poor have an average monthly per capita expenditure below the poverty line. The case of poverty in Indonesia is still unresolved; the Government continues to try to give the best to the entire community so that the problem of poverty can at least continue to decrease. One form of government concern for the poor is the assistance program provided to the poor. This study will classify based on data from the North Penajam Paser (PPU) community obtained from the results of the National Socio-Economic Survey (Susenas) to know how the Naïve Bayes method is in determining the eligibility of the poor recipients of assistance. Based on the research that has been carried out, a system for determining the poor recipients of assistance is produced, where the test results get the highest accuracy in the third scenario, namely 60% or 328 training data and 40% or 218 test data, where the accuracy obtained is 77.98%.

Keywords: Classification, Poor Society, Laplace Correction, Data Mining, Naïve Bayes

Abstrak

Kemiskinan adalah ketidakmampuan dalam memenuhi kebutuhan hidup seperti makanan, pakaian, dan tempat tinggal. Penduduk miskin adalah penduduk yang memiliki rata-rata pengeluaran perkapita per-bulan di bawah garis kemiskinan. Kasus kemiskinan di Indonesia hingga saat ini masih belum terselesaikan, pemerintah terus berusaha memberikan yang terbaik kepada seluruh masyarakat agar permasalahan kemiskinan setidaknya bisa terus berkurang. Salah satu bentuk kepedulian pemerintah pada masyarakat miskin adalah program bantuan yang diberikan kepada masyarakat miskin. Penelitian ini melakukan klasifikasi berdasarkan data masyarakat Penajam Paser Utara (PPU) yang didapatkan dari hasil Survey Sosial Ekonomi Nasional (Susenas). Klasifikasi dilakukan untuk menentukan kelayakan masyarakat miskin dalam penerima bantuan yang diterapkan menggunakan metode Naïve Bayes. Sejumlah 547 data digunakan untuk evaluasi kinerja dari metode Naïve Bayes. Data tersebut dibedakan menjadi dua sebagai data latih dan data uji yaitu 60% (328 data latih) dan 40% (219 data uji). Berdasarkan evaluasi yang dilakukan, nilai akurasi yang dihasilkan mencapai 77,98%.

Kata Kunci: Klasifikasi, Masyarakat Miskin, *Laplace Correction*, *Data Mining*, Naïve Bayes

1. PENDAHULUAN

Kemiskinan merupakan salah satu permasalahan sosial yang belum terselesaikan di Indonesia. Berdasarkan data dari Badan Pusat Statistik (BPS) jumlah masyarakat miskin di Indonesia tahun 2021 mengalami kenaikan dari tahun sebelumnya. Salah satu daerah di Indonesia yang juga mengalami kenaikan tersebut adalah Kabupaten Penajam Paser Utara. Persentase tingkat kemiskinan pada tahun 2021 di Penajam mengalami kenaikan sebanyak 0,25% dari tahun sebelumnya (Badan Pusat Statistik, 2021). Kemiskinan merupakan masalah yang dipengaruhi oleh faktor-faktor yang saling berkaitan, seperti tingkat pendapatan masyarakat, pengangguran,

kesehatan, pendidikan, akses terhadap barang dan jasa, lokasi, geografis, gender, dan lokasi lingkungan (Novriansyah, 2018).

Pemerintah mendirikan program pengentasan dalam mengatasi permasalahan kemiskinan di Indonesia. Program utama yang dilakukan adalah kelompok program yang bertujuan untuk mengurangi beban hidup dalam memenuhi kebutuhan pangan, kesehatan, dan pendidikan. Contoh program seperti pendistribusian beras miskin (Raskin dan BPNT), pemberian Jaminan Kesehatan Masyarakat (Jamkesmas), pemberian bantuan keuangan (BLT dan PKH), bantuan pendidikan bagi siswa miskin (BSM dan PIP), dan Bantuan/Subsidi Pemerintah Daerah. Program tersebut secara langsung berkaitan dengan terwujudnya hak asasi manusia (Umami, 2013). Bantuan yang diberikan melewati suatu proses sehingga bantuan tersalurkan dengan tepat sasaran kepada masyarakat miskin. Proses yang dapat dilakukan dalam menentukan masyarakat miskin penerima bantuan yaitu dengan melakukan klasifikasi masyarakat miskin menggunakan metode *machine learning* seperti pada penelitian Aji (2019), Arifando et al. (2017), Fitriani (2020), Kurnia et al. (2019), Purnama et al. (2020), dan Riyanah & Fatmawati (2021).

Klasifikasi menggunakan metode Naïve Bayes telah banyak digunakan dalam penelitian terkait penentuan masyarakat miskin dan penerima bantuan. Naïve Bayes merupakan teknik prediksi probabilistik sederhana yang berdasar pada penerapan teorema Bayes dengan asumsi independensi yang kuat (Aji, 2019). Metode ini memiliki sifat yang efektif dan cepat dalam mengolah data berjumlah besar. Kemampuan tersebut membuat metode ini sering digunakan pada aplikasi seperti *spam filtering* dan deteksi anomali di jaringan komputer (Kurniawan, 2020). Metode Naïve Bayes juga memiliki kemampuan yang baik dari metode *data mining* lainnya seperti *Support Vector Machine* dalam melakukan klasifikasi (Maarif, 2016).

Penelitian sebelumnya terkait klasifikasi masyarakat miskin dan penerima bantuan telah dilakukan oleh Putri et al. (2021). Pada penelitian tersebut dilakukan klasifikasi untuk rumah tangga miskin di Provinsi Papua menggunakan metode Naïve Bayes. Variabel yang digunakan pada penelitian tersebut yaitu, jenis kelamin, pendidikan KRT, lapangan usaha KRT, jenis atap terluas, jenis dinding terluas, jenis lantai terluas, sumber air minum, sumber penerangan, dan bahan bakar untuk memasak. Hasil diperoleh nilai akurasi sebesar 80%. Kemudian pada penelitian lainnya yang dilakukan oleh Annur (2018) melakukan klasifikasi penduduk miskin di Kecamatan Tibawa. Variabel yang digunakan yaitu umur, pendidikan, pekerjaan, penghasilan, tanggungan, dan status (kawin/belum kawin). Hasilnya didapatkan nilai akurasi sebesar 73%. Selanjutnya penelitian yang telah dilakukan oleh Nurmayanti et al. (2021) melakukan klasifikasi masyarakat miskin menggunakan metode Naïve Bayes dan variabel yang digunakan yaitu jenis kelamin, peserta pkh, jumlah anggota rumah tangga, status kepemilikan rumah, fasilitas rumah, luas lantai rumah, dan fasilitas elektronik. Berdasarkan hasil penelitian yang telah dilakukan diperoleh nilai akurasi sebesar 96,63%. Kesimpulan yang diperoleh yaitu metode Naïve Bayes memiliki kemampuan yang baik untuk melakukan klasifikasi masyarakat miskin.

Pada penelitian ini metode Naïve Bayes diterapkan untuk melakukan klasifikasi kelayakan masyakat miskin penerima bantuan di wilayah Kabupaten Penajam Paser Utara. Perbedaan dengan penelitian sebelumnya terletak pada jenis variabel yang digunakan. Pada penelitian ini jumlah variabel yang digunakan lebih banyak dari penelitian sebelumnya, di mana variabel tersebut dapat menggambarkan karakteristik dari masyarakat miskin. Data yang digunakan pada penelitian ini merupakan data yang diperoleh dari Survey Sosial Ekonomi Nasional (Susenas) KOR RT bulan Maret Tahun 2021 pada wilayah Kabupaten Penajam Paser Utara. Data yang dikumpulkan terdiri dari 546 data dengan jumlah data masyarakat yang layak menerima bantuan sebanyak 172 data dan masyarakat yang tidak menerima bantuan sebanyak 374 data.

2. METODE PENELITIAN

2.1 Pengumpulan Data

Data yang dikumpulkan pada penelitian ini berupa data sekunder. Data yang dikumpulkan berupa data Survei Sosial Ekonomi Nasional KOR RT Wilayah Penajam Paser Utara Bulan Maret Tahun

2021. Data didapatkan dari situs resmi BPS Indonesia. Data yang dikumpulkan terdiri dari 41 parameter yang dapat dilihat pada Tabel 1. Adapun contoh dari data yang dikumpulkan pada penelitian ini dapat dilihat pada Tabel 2. Pada Tabel 2, nilai 1, 5, 8, dan 9 merupakan jawaban untuk “ya”, “tidak”, “tidak tahu”, dan “menolak menjawab”.

Tabel 1 Parameter Penelitian

Variabel	Keterangan
R1 = R1701	Selama setahun terakhir, apakah khawatir tidak akan memiliki cukup makanan?
R2 = R1702	Selama setahun terakhir, apakah ada saat di mana tidak dapat menyantap makanan sehat dan bergizi?
R3 = R1703	Selama setahun terakhir, apakah hanya menyantap sedikit jenis makanan?
R5 = R1705	Selama setahun terakhir, apakah makan lebih sedikit daripada seharusnya?
R6 = R1706	Selama setahun terakhir, apakah kehabisan makanan?
R7 = R1707	Selama setahun terakhir, apakah merasa lapar tapi tidak makan?
R8 = R1708	Selama setahun terakhir, apakah tidak makan seharian?
R9 = R1801	Berapa jumlah keluarga yang tinggal di dalam bangunan sensus/rumah ini?
R10 = R1802	Apakah status kepemilikan bangunan tempat tinggal yang ditempati?
R11 = R1803	Apa jenis bukti kepemilikan tanah bangunan tempat tinggal ini?
R12 = R1804	Berapa luas lantai rumah bangunan tempat tinggal?
R13 = R1805	Apakah KRT/pasangannya/anaknya memiliki rumah lain, selain rumah yang ditempati saat ini?
R14 = R1806	Apakah bahan bangunan utama atap rumah terluas?
R15 = R1807	Apakah bahan bangunan utama dinding rumah terluas?
R16 = R1808	Apakah bahan bangunan utama lantai rumah terluas?
R17 = R1809A	Apakah memiliki fasilitas tempat buang air besar?
R18 = R1809B	Apakah jenis kloset yang digunakan?
R19 = R1809C	Di manakah tempat pembuangan akhir tinja?
R20 = R1810A	Apa sumber air utama yang digunakan untuk minum?
R21 = R1812	Apakah pernah mengalami kekurangan air minum selama minimal 24 jam?
R22 = R1814A	Apakah sumber air utama untuk mandi/cuci/dll?
R23 = R1814B	Berapa jarak ke tempat penampungan limbah/kotoran/tinja terdekat?
R24 = R1816	Apakah sumber utama penerangan rumah tangga ini?
R25 = R1817	Apakah jenis bahan bakar utama yang digunakan untuk memasak?
R26 = R2001A	Apakah memiliki tabung gas 5,5 kg atau lebih?
R27 = R2001B	Apakah memiliki lemari es/kulkas?
R28 = R2001C	Apakah memiliki AC?
R29 = R2001D	Apakah memiliki pemanas air (water heater)?
R30 = R2001E	Apakah memiliki telepon rumah (PSTN)?
R31 = R2001F	Apakah memiliki komputer/laptop?
R32 = R2001G	Apakah memiliki emas/perhiasan (minimal 10 gram)?
R33 = R2001H	Apakah memiliki sepeda motor?
R34 = R2001I	Apakah memiliki perahu?
R35 = R2001J	Apakah memiliki perahu motor?
R36 = R2001K	Apakah memiliki mobil?
R37 = R2001L	Apakah memiliki televisi layar datar (minimal 30 inch)?
R38 = R2001M	Apakah memiliki tanah/lahan?
R39 = R2101A	Apakah sumber terbesar pembiayaan di rumah tangga ini?
R40 = R2101C	Dari manakah sumber utama kiriman uang/barang?
R41 = R301	Banyaknya ART

Tabel 2 Data Penelitian

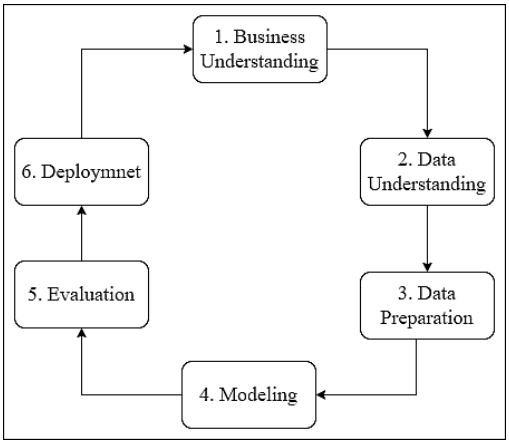
No	R1	R2	R3	R4	R5	R6	…	R38	R39	R40	R41	Label
1	5	5	5	5	5	5	…	1	1	2	3	1
2	5	5	5	5	5	5	…	1	1	2	4	0
3	5	5	5	5	5	5	…	1	1	2	3	0
4	5	5	5	5	5	5	…	5	1	2	4	0
5	5	5	5	5	5	5	…	1	1	2	5	0
6	5	5	5	5	5	5	…	5	2	2	2	0
7	5	5	5	5	5	5	…	1	1	2	1	0
8	5	5	5	5	5	5	…	1	1	2	4	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
539	5	5	5	5	5	5	…	5	1	2	4	1
540	5	5	1	1	1	1	…	5	1	2	1	0
541	1	5	1	5	5	5	…	5	1	2	6	1
542	5	5	5	5	5	5	…	1	1	2	8	1
543	5	5	5	5	5	5	…	1	1	2	3	1
544	5	5	5	5	5	5	…	1	2	2	1	1
545	5	5	5	5	5	5	…	1	2	2	1	0
546	5	5	5	5	5	5	…	1	1	2	5	0

2.2 Proses Pengolahan Data

Data mining adalah suatu proses menemukan hubungan yang berarti, pola, dan kecenderungan dengan memeriksa sekumpulan besar data yang tersimpan dalam penyimpanan dengan menggunakan teknik pengenalan pola seperti teknik statistik dan matematika (Nastuti & Harahap, 2019). *Cross Industry Standard Process for Data Mining* (CRISP-DM) merupakan salah satu metodologi dalam *data mining*. Terdapat 6 (enam) tahapan dalam metodologi CRISP-DM yang dapat dilihat pada Gambar 1.

Tahap dari metodologi CRISP-DM pada Gambar 1 dapat dijelaskan sebagai berikut (Suntoro, 2019).

1) *Business understanding* merupakan tahap pemahaman pada bisnis (penelitian) yang mana pada tahap tersebut menentukan tujuan bisnis (penelitian) serta kebutuhan atas tujuan yang akan dicapai.



Gambar 1 Tahapan CRISP-DM

2) *Data understanding* merupakan tahapan pemahaman pada data yang mana pada tahap ini peneliti mempersiapkan, mengevaluasi persiapan data, dan mengumpulkan data, selanjutnya data yang terkumpul akan dideskripsikan.

3) *Data preparation* disebut juga dengan data *pre-processing*, merupakan tahap membangun data ke dalam format yang diinginkan dengan mengidentifikasi, memilih, dan membersihkan data yang diperoleh.
4) *Modeling* adalah tahap pembuatan aplikasi dari algoritma untuk mencari, mengidentifikasi, dan menampilkan pola dari data yang akan diestimasi, prediksi, klasifikasi, clustering, atau melihat hubungan asosiatif.
5) *Evaluation* digunakan untuk membantu pengukuran evaluasi pada model. Pada penerapan klasifikasi, pengukuran evaluasi yang banyak digunakan adalah *accuracy, recall, precision, G-Mean, F-Measure*, dan lain sebagainya.
6) *Deployment*, tahapan *deployment* digunakan untuk melakukan otomatisasi model atau pengembangan aplikasi, terintegrasi dengan sistem informasi manajemen atau operasional yang ada.

2.3 Penggunaan Algoritma Naïve Bayes

Menurut Muljono et al. (2018) Naïve Bayes merupakan algoritma klasifikasi sederhana yang sering diimplementasikan untuk klasifikasi dokumen. Terdapat 2 tahapan dalam proses klasifikasi yaitu proses pelatihan dan pengujian. Proses pelatihan adalah pelatihan pada proses pembuatan model dengan menggunakan data latih yang sudah ditentukan label dari dokumen. Sedangkan pengujian merupakan proses untuk mengetahui keakuratan model dengan menggunakan data yang disebut dengan data uji. Adapun persamaan dari teorema Naïve Bayes ditunjukkan pada Pers. (1) (Natuzzuhriyyah et al., 2021).

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \qquad (1)$$

Di mana X merupakan data dengan kelas yang belum diketahui dan H adalah hipotesis data yang merupakan suatu kelas spesifik. Sementara itu, $P(H|X)$ dan $P(X|H)$ adalah probabilitas hipotesis H berdasarkan kondisi X (posteriori Prob.) dan probabilitas X berdasarkan kondisi tersebut. Variabel $P(H)$ dan $P(X)$ merupakan hipotesis H (prior Prob.) dan probabilitas dari X.

Persamaan $P(H|X)$ dapat disederhanakan menjadi Pers. (2) dengan asumsi independensi. Penyederhanaan ini dilakukan agar perhitungan tidak sulit dilakukan karena jika semakin banyak faktor-faktor kompleks yang mempengaruhi nilai probabilitas maka semakin mustahil untuk menghitung nilai tersebut satu persatu (Suntoro, 2019).

$$P(C|F_1, \dots, F_n) = P(C)\prod_{i=1}^{n} P(F_i|C) \qquad (2)$$

Dataset dengan tipe data berupa numerik dihitung dengan perhitungan gaussian (Suntoro, 2019) di mana perhitungan tersebut dapat dilihat dari Pers. (3).

$$(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi.\sigma}}\, exp^{\frac{-(x-\mu)^2}{2\sigma^2}} \qquad (3)$$

Perhitungan rata-rata dapat dilihat pada Pers. (4).

$$\mu = \frac{\sum_i^n x_i}{n} \qquad (4)$$

Perhitungan standar deviasi dapat dilihat pada Pers. (5).

$$\sigma = \sqrt{\frac{\sum_i^n (x_i - \mu)^2}{n-1}} \qquad (5)$$

Langkah dari algoritma Naïve Bayes diawali dengan menyiapkan *dataset*, kemudian menghitung jumlah kelas pada *data training*. Selanjutnya menghitung jumlah kasus yang sama dengan kelas yang sama, kemudian dilakukan terhadap semua hasil sesuai dengan *data testing* yang akan

dicari kelasnya. Terakhir adalah membandingkan hasil per kelas, nilai tertinggi ditetapkan sebagai kelas baru.

2.4 *Laplace Correction*

Hasil perhitungan menggunakan metode Naïve Bayes akan memungkinkan menghasilkan nilai perhitungan 0 karena saat dilakukan pengujian tidak ditemukan atribut pada data latih. *Laplace Correction* merupakan metode yang digunakan untuk mengatasi permasalahan tersebut. Persamaan metode *Laplace Correction* dapat dilihat pada Pers. (6) (Setiawan et al., 2021).

$$\text{Pi} = \frac{p(mi)+1}{n+k} \qquad (6)$$

Di mana Pi adalah probabilitas dari *mi* yang merupakan jumlah sampel dalam kelas dari anggota dengan n dan k jumlah sampel dan kelas dari atribut mi.

2.5 *Confusion Matrix*

Pengujian yang dilakukan untuk mengevaluasi kinerja dari penggunaan model klasifikasi yaitu menggunakan *confusion matrix*. Menurut Gunawan et al. (2018) terdapat empat istilah yang digunakan dalam metode tersebut yaitu:

1) True Positive : Jumlah data hasil output benar dan kelas output positif.
2) True Negative : Jumlah data hasil output benar dan kelas output negatif.
3) False Positive : Jumlah data hasil output salah dan kelas output positif.
4) False Negative : Jumlah data hasil output salah dan kelas output negatif.

Metode tersebut pada umumnya digunakan dalam menghitung tingkat *accuracy* suatu model, selain itu terdapat parameter lain yang dapat dihitung menggunakan metode tersebut yaitu, *recall* dan *precision*. Tabel dari *confusion matrix* dapat dilihat pada Tabel 3.

Tabel 3 *Confusion Matrix*

Kelas Target	**Kelas *Output***	
	+	**-**
+	*True Positives (TP)*	*False Negatives (FN)*
-	*False Positives (FP)*	*True Negatives (TN)*

Accuracy yaitu pengujian yang dilakukan untuk mendapatkan nilai dari kemampuan model klasifikasi secara keseluruhan. *Recall* adalah persentase keberhasilan sistem saat menentukan ulang informasi. *Precision* adalah persentase keakuratan antara informasi yang diminta oleh pengguna dan respon yang ditunjukkan oleh system. *F1-Score* merupakan alat ukur yang digunakan untuk mendapatkan nilai rata-rata harmonik dari *recall* dan *precision*. Nilai *accuracy, recall,* dan *f1-score* dapat dihitung dengan menggunakan rumus pada Pers. (7) sampai (10).

$$Accuracy \quad : \frac{(TP+TN)}{(TP+FN+FP+TN)} \times 100\% \qquad (7)$$

$$Recall \quad : \frac{TP}{(TP+FN)} \times 100\% \qquad (8)$$

$$Precision \quad : \frac{TP}{(TP+FP)} \times 100\% \qquad (9)$$

$$F1-Score \quad : \frac{(2\times Recall \times Precision)}{(Recall+Precision)} \times 100\% \qquad (10)$$

3. HASIL DAN PEMBAHASAN

3.1 *Data Understanding*

Pada tahap *data understanding* dijelaskan proses apa saja yang dilakukan oleh peneliti dalam memahami dan mengevaluasi persiapan data. Dalam proses memahami dan mengevaluasi persiapan data pada penelitian ini peneliti melakukan seleksi variabel, pelabelan kelas, dan eksplorasi data.

3.1.1 Seleksi Variabel

Data yang digunakan merupakan data Susenas Kor RT wilayah Kabupaten Penajam Paser Utara, data tersebut berisi data diri masyarakat wilayah penajam. Variabel penelitian ini menggunakan variabel yang ada pada data Susenas kemudian dilakukan seleksi berdasarkan kriteria dari masyarakat miskin. Penelitian ini menggunakan variabel dari hasi studi literatur terkait variabel dan kriteria masyarakat miskin. Karakteristik masyarakat miskin calon penerima bantuan yang digunakan pada data penelitian ini memiliki 5 kriteria yang diduga dapat menentukan kelayakan masyarakat penerima bantuan yaitu makanan, perumahan, kepemilikan barang, sumber pembiayaan, dan banyaknya ART. Adapun penjelasan mengenai kriteria tersebut adalah sebagai berikut:

1) Makanan (Wati & Hadi, 2016), berisi data mengenai akses KRT/Pasangan/ART terhadap makanan. Kriteria tersebut terdiri dari beberapa variabel dan atribut yang memiliki tipe data kategorik.
2) Perumahan (Arifando et al., 2017; Purnama et al., 2020), berisi data jumlah keluarga yang tinggal dalam rumah, kepemilikan rumah, dan kondisi rumah serta fasilitas rumah. Kriteria tersebut terdiri dari beberapa variabel dengan atribut bertipe data kategorik atau numerik.
3) Kepemilikan barang (Nurmayanti et al., 2021; Sugianto & Maulana, 2019), pada kriteria tersebut berisi data barang-barang yang dimiliki suatu rumah tangga seperti TV, Kulkas, AC, Pemanas Air, Telepon Rumah, Komputer/Laptop, Kendaraan, dan Aset (Tanah & Perhiasan). Kriteria tersebut terdiri dari tiga belas variabel dengan atribut yang bertipe data kategorik.
4) Sumber Pembiayaan (Badan Pusat Statistik, 2021), berisi data tentang sumber pembiayaan suatu rumah tangga. Kriteria tersebut terdiri dari 2 variabel dengan atribut yang bertipe data kategorial.
5) Banyaknya ART (Arifando et al., 2017; Kurnia et al., 2019; Sugianto & Maulana, 2019), berisi data jumlah anggota rumah tangga dalam suatu rumah tangga. Kriteria tersebut terdiri dari satu variabel dengan atribut yang bertipe data numerik.

3.1.2 Pelabelan Kelas

Proses pelabelan kelas pada data yang digunakan penelitian ini didapatkan dari data bantuan yang ada pada data Susenas. Adapun jenis bantuan yang digunakan dalam penelitian ini mencakup bantuan PKH, KKS, PIP SD, PIP SMP, PIP SM, PIP kuliah, dan BPNT. Data masyarakat yang menerima bantuan pada penjelasan sebelumnya diberikan label sebagai data masyarakat yang “Layak” menerima bantuan dan data masyarakat yang tidak menerima bantuan diatas diberi label sebagai masyarakat yang “Tidak Layak” menerima bantuan.

3.1.3 Eksplorasi Data

Pemahaman data dilakukan untuk mengetahui kecenderungan pusat data yang mana hal tersebut dapat berguna untuk membersihkan data yang kotor. Proses pemahaman data yang dilakukan yaitu mencari nilai kosong = 0, data bernilai tidak tahu = 8, data bernilai menolak jawab = 9, atribut yang tidak terpakai, dan modus dari setiap atribut yang ada pada masing-masing variable data yang digunakan. Hasil proses pemahaman data dalam mencari jumlah data yang kosong = 0, data bernilai tidak tahu = 8, dan data bernilai menolak jawab = 9. Atribut tidak terpakai dan nilai modus setiap variabel dapat dilihat pada Tabel 4.

Tabel 4 Hasil Eksplorasi Data

Var	Data Terisi	Jumlah 0	Jumlah 8	Jumlah 9	Atribut Tidak Ada	Modus
R1	546	0	3	1	0	5
R2	546	0	3	0	9	5
R3	546	0	4	0	8 – 9	5
R4	546	0	0	0	9	5
R5	546	0	2	0	9	5
R6	546	0	1	0	8 – 9	5
R7	546	0	0	0	8 – 9	5
R8	546	0	0	0	4 – 5 -6 -7	5
R9	546	0	0	0	5	1
R10	546	0	0	0	-	1
R11	455	91	0	0	-	1
R12	546	0	0	0	-	72
R13	546	0	0	0	-	5
R14	546	0	0	0	7 – 8	3
R15	546	0	0	0	2 – 4 – 5 – 6 – 7	3
R16	546	0	0	0	7 – 9	5
R17	546	0	0	0	3	1
R18	526	20	0	0	-	1
R19	526	20	0	0	2 – 6	1
R20	546	0	0	0	11	2
R21	546	0	0	0	8	5
R22	546	0	0	0	1 – 11 – 2	4
R23	370	176	17	0		2
R24	546	0	0	0	4	1
R25	546	0	0	0	0 – 1 – 6 – 8 – 9 - 11	4
R26	546	0	0	0	-	5
R27	546	0	0	0	-	1
R28	546	0	0	0	-	5
R29	546	0	0	0	-	5
R30	546	0	0	0	-	5
R31	546	0	0	0	-	5
R32	546	0	0	0	-	5
R33	546	0	0	0	-	1
R34	546	0	0	0	-	5
R35	546	0	0	0	-	5
R36	546	0	0	0	-	5
R37	546	0	0	0	-	5
R38	546	0	0	0	-	1
R39	546	0	0	0	-	1
R40	35	511	0	0	-	2
R41	546	0	0	0	-	4

3.2 *Data Preparation*

Data preparation atau biasa disebut juga dengan pra-pemrosesan merupakan proses yang dilakukan untuk meningkatkan kualitas data dan meningkatkan efisiensi dan kemudahan penambangan data. Pada penelitian ini pra-pemrosesan yang dilakukan adalah dengan melakukan proses pembersihan data dan reduksi data.

3.2.1 Pembersihan Data

Berdasarkan Tabel 4 dapat dilihat bahwa pada data yang digunakan masih ada data yang bernilai kosong atau tidak terdefinisi yaitu pada variabel R11, R18, R19, R23, dan R40, data bernilai 8 atau tidak tahu yaitu pada variabel R1, R2, R3, R5, R6, dan R23, serta data bernilai 9 atau

menolak menjawab pada variabel R23. Data kotor tersebut harus dibersihkan terlebih dahulu agar data dapat menghasilkan proses klasifikasi yang lebih baik. Proses yang dilakukan yaitu dengan mengubah data kosong tersebut dengan nilai modus dan atribut yang sesuai dengan kondisi pada data. Hasil pra-pemrosesan dapat dilihat pada Tabel 5.

Tabel 5 Hasil Pra-pemrosesan Pembersihan Data

Variabel	Total 0	Total 8	Total 9	Modus
R1	0	3	0	5
R2	0	3	0	5
R3	0	4	0	5
R5	0	2	0	5
R6	0	1	0	5
R11	91	0	0	1
R18	20	0	0	1
R19	20	0	0	1
R23	176	17	0	2
R40	511	0	0	2

Hasil pra-pemrosesan dengan teknik pembersihan data didapatkan hasil sebagai berikut:
1) Sebanyak 5 variabel yang memiliki data yang kosong yaitu, R11 dengan total data kosong sebanyak 91 data, R18 dengan total data kosong sebanyak 20, R19 dengan total data kosong sebanyak 20, R23 dengan total data kosong sebanyak 176, dan R40 dengan data kosong sebanyak 511.
2) Sebanyak 6 variabel yang memiliki data bernilai 8 dan 9 yaitu R1 dengan total data bernilai 8 sebanyak 3 data dan data bernilai 9 sebanyak 1 data, R2 dengan total data bernilai 8 sebanyak 3 data, R3 dengan total data bernilai 8 sebanyak 4 data, R5 dengan total data bernilai 8 sebanyak 2 data, R6 dengan total data bernilai 8 sebanyak 1 data, dan R23 dengan total data bernilai 8 sebanyak 17 data.

Data kotor tersebut diisi menggunakan nilai yang sering muncul atau modus masing-masing variabel dan atribut sesuai kondisi pada data.

3.2.2 Reduksi Data

Berdasarkan data pada Tabel 4 dapat dilihat bahwa pada data yang digunakan terdapat atribut yang tidak terpakai, yang mana atribut tersebut terdaftar pada meta data Susenas tetapi tidak ada pada data Susenas. Oleh karena itu, pada penelitian ini atribut yang tidak terpakai dihilangkan. Adapun atribut yang dihilangkan yaitu variabel R2, R3, R5, R6 dengan atribut 9, R4, R7, R8, dengan atribut 8 dan 9, R9 dengan atribut 4, 5, 6, dan 7, R10 dengan atribut 5, R14 dengan atribut 7 dan 9, R15 dengan atribut 2, 4, 5, 6, dan 7, R16 dengan atribut 7 dan 9, R17 dengan atribut 3, R19 dengan atribut 2 dan 6, R20 dengan atribut 11, R21 dengan atribut 8, R22 dengan atribut 1, 11, dan 2, R24 dengan atribut 4, dan R25 dengan atribut 0, 1, 6, 8, 9, dan 11.

3.3 *Modeling*

Pada tahap ini dilakukan proses pembuatan model dari metode yang digunakan yaitu metode Naïve Bayes. Pada proses pembuatan model klasifikasi menggunakan metode Naïve Bayes terlebih dahulu dilakukan proses *split* data untuk membagi data menjadi data latih dan data uji. Setelah itu dilakukan perhitungan probabilitas data latih kemudian dilanjutkan dengan melakukan perhitungan probabilitas terhadap data uji dan terakhir dilakukan pengujian terhadap data uji.

3.3.1 Membagi Data

Proses pembagian data dilakukan dengan teknik *percentage split.* Pembagian data pada *percentage split* menggunakan *3 scenario*, yang mana *scenario* 1 dengan jumlah perbandingan 60% (328 data latih)–40% (218 data uji), *scenario* 2 dengan jumlah perbandingan 70% (382 data

latih)–30% (164 data uji), dan *scenario* 3 dengan jumlah perbandingan 80% (437 data latih)–20% (109 data uji).

3.3.2 Perhitungan Algoritma Naïve Bayes

1) Perhitungan Probabilitas Kelas

Proses pertama dilakukan perhitungan probabilitas dari masing-masing kelas, yaitu probabilitas kelas layak dan probabilitas kelas tidak layak. Berikut adalah perhitungan probabilitas kelas:

$S = 437$

$N_{(layak)} = 144$

$N_{(tidak\ layak)} = 293$

$P_{(layak)} = \frac{144}{437} = 0{,}3295$

$P_{(tidak\ layak)} = \frac{293}{437} = 0{,}6705$

Di mana S merupakan jumlah seluruh data dengan N(Layak) dan N(Tidak Layak) adalah jumlah kelas “Layak” dan “Tidak Layak”. Sementara itu, P(Layak) dan P(Tidak Layak) merupakan jumlah probabilitas kelas “Layak” dan “Tidak Layak”.

2) Perhitungan Probabilitas Atribut Tipe Data Kategorial

Langkah selanjutnya adalah menghitung probabilitas data uji. Atribut pada *dataset* memiliki tipe data kategorial dan numerik. Pada tahap ini dilakukan perhitungan nilai probabilitas data uji yang bertipe data kategorial menggunakan Pers. (2). Salah satu perhitungan probabilitas atribut dengan tipe data kategorial ada pada variabel ‘’R26 - Apakah memiliki tabung gas 5,5 kg atau lebih?” yang memiliki atribut “1 = Ya”, dan “5=Tidak”. Contoh perhitungan probabilitas variabel tersebut adalah sebagai berikut:

$P_{(1 = Ya \mid 1 = Layak)} = \frac{109}{144} = 0{,}7569444444444444$

$P_{(1 = Ya \mid 0 = Tidak\ Layak)} = \frac{234}{293} = 0{,}7986348122866894$

$P_{(5 = Tidak \mid 1 = Layak)} = \frac{35}{144} = 0{,}2430555555555556$

$P_{(5 = Tidak \mid 0 = Tidak\ Layak)} = \frac{59}{293} = 0{,}2013651877133106$

3) Perhitungan Probabilitas Atribut Numerik

Selain menghitung probabilitas data uji bertipe data kategorial selanjutnya juga dilakukan perhitungan probabilitas pada data uji yang bertipe data numerik dengan menggunakan Pers. (3). Salah satu perhitungan probabilitas atribut dengan tipe data numerik ada pada variabel “R12 - Berapa luas lantai rumah bangunan tempat tinggal?”. Contoh perhitungan probabilitas variabel tersebut dengan atribut bernilai 38 adalah sebagai berikut:

$$\mu = \frac{\sum_i^n x_i}{n}$$

$$\mu_{kelas="1"} = \frac{(30 + 72 + 36 + 36 + 72 + 20 + \ldots + 96 + 120 + 66 + 30 + 60 + 48 + 76)}{144}$$

$$= \frac{9046}{144} = 62{,}81944$$

$$\mu_{kelas="0"} = \frac{(81 + 54 + 63 + 112 + 40 + 40 + \ldots + 90 + 50 + 80 + 32 + 60 + 72 + 60)}{293}$$

$$= \frac{21480}{293} = 73{,}31058$$

$$\sigma = \sqrt{\frac{\sum_i^n (x_i - \mu)^2}{n - 1}}$$

$$\sigma_{kelas="1"} = \sqrt{\frac{(30 - 62{,}81)^2 + (72 - 62{,}81)^2 + \ldots + (48 - 62{,}81)^2 + (76 - 62{,}81)^2}{144 - 1}}$$

$$= 27{,}47272$$

$$\sigma_{kelas="0"} = \sqrt{\frac{(91 - 73{,}31)^2 + (54 - 73{,}31)^2 + \ldots + (72 - 73{,}31)^2 + (60 - 73{,}31)^2}{293 - 1}}$$

$$= 40{,}29552$$

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi} \cdot \sigma} \, exp^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

$$g(8_1, \mu_1, \sigma_1) = \frac{1}{\sqrt{2 \times 3{,}14} \times 27{,}47272} \, exp^{\frac{-(32 - 62{,}81944)^2}{2(27{,}47272)^2}} = 0{,}0077$$

$$g(8_0, \mu_0, \sigma_0) = \frac{1}{\sqrt{2 \times 3{,}14} \times 40{,}29552} \, exp^{\frac{-(32 - 73{,}31058)^2}{2(40{,}29552)^2}} = 0{,}0058$$

3.4 Evaluasi Metode

Pengujian sistem penentuan masyarakat miskin penerima bantuan menggunakan metode *Confusion Matrix* untuk mendeteksi *accuracy, recall, precision*, dan *f1-score*. Proses pembagian data untuk mendapatkan hasil dari pengujian menggunakan *percentage split* dengan tiga *scenario*. Pengujian dilakukan berdasarkan keakuratan hasil klasifikasi data uji dengan data masyarakat yang layak dan tidak layak menerima bantuan sesuai dengan data Susenas yang telah didapatkan. Adapun *confusion matrix* hasil pengujian dari keseluruhan *scenario* pada *percentage split* dapat dilihat pada Tabel 6.

Tabel 6 Hasil *Confusion Matrix Percentage Split*

Keterangan	Persentase Data Latih dan Data Uji	TP	TN	FP	FN
Scenario 1	80%-20%	15	68	13	13
Scenario 2	70%-30%	24	99	19	22
Scenario 3	60%-40%	43	127	22	26

Hasil pengujian perolehan *accuracy, recall, precision,* dan *f1-score* seluruh data uji dapat dilihat pada Tabel 7.

Tabel 7 Hasil Pengujian *Percentage Split*

Keterangan	Persentase Data Latih dan Data Uji	*Accuracy*	*Recall*	*Precision*	*F1-Score*
Scenario 1	80%-20%	76,15%	53,57%	53,57%	53,57%
Scenario 2	70%-30%	75%	52,17%	55,80%	53,93%
Scenario 3	60%-40%	77,98%	62,32%	66,15%	64,18%

Berdasarkan Tabel 7, dapat disimpulkan bahwa sistem penentuan kelayakan masyarakat miskin penerima bantuan yang dibuat dapat melakukan prediksi dengan cukup baik pada *scenario* ketiga yaitu 60% data latih (328 data) dan 40% data uji (218 data). Hasil *accuracy* yang didapatkan adalah 77,98% dengan *recall* 62,32%, *precision* 66,15%, dan *F1-score* 64,18%.

Pada penelitian ini dilakukan perbandingan hasil yang diperoleh dari metode, variabel, dan jumlah data yang diusulkan dengan metode, variabel, dan jumlah data yang digunakan penelitian lain. Hal ini dilakukan untuk mengevaluasi model yang dibangun. Adapun perbandingan penelitian sebelumnya ditunjukkan pada Tabel 8.

Tabel 8 Perbandingan Kinerja Metode yang diusulkan dengan Metode Lain

No.	Sitasi	Parameter	Klasifikasi	Jumlah Data	Akurasi
1	Annur (2018)	Pendidikan, Pekerjaan, Penghasilan, Tanggungan, dan Status Perkawinan.	Naïve Bayes	190	73%
2	Aji (2019)	Luas Bangunan, Jenis Lantai, Jenis Dinding, Fasilitas MCK, Sumber Air Minum, Sumber Penerangan, Pekerjaan Kepala Keluarga, Penghasilan, Kondisi Rumah, Jumlah Tanggungan, Bahan Bakar Memasak, dan Kepemilikan Asset (status Kepemilikan Rumah).	Naïve Bayes	300	80%
3	Arifando et al. (2017)	Jumlah Anggota Keluarga, Pendapatan, Umur, Kondisi Rumah, Status Kepemilikan Rumah, Pengeluaran, dan Pendidikan Terakhir.	*Learning Vector Quantization* (LVQ)	N/A	98%
4	Fitriani (2020)	Lansia, Pendidikan Terakhir, Anak Sekolah, Pekerjaan, Ibu Hamil, Membeli Pakaian, Frekuensi Makanan, Berobat Ke Puskesmas, Membeli (Daging Ayam, Susu), Aset, Jenis Dinding, Jenis Lantai, Sumber Penerangan, Sumber Air Minum, Jenis Bahan Bakar Memasak, Fasilitas BAB, dan Luas Lantai.	C4.5 Naïve Bayes	1.109	91,25% 87,11%
5	Purnama et al. (2020)	Jumlah Tanggungan, Kondisi Rumah, Sumber Air Minum, Ketersediaan WC, dan Bahan Bakar.	Naïve Bayes	210	82,14%
6	Kurnia et al. (2019)	Kartu Keluarga, Jumlah Anggota Keluarga, Jenis Pekerjaan, dan Penghasilan Bulanan.	K- Nearest Neighbor	100	90%

Tabel 9 Perbandingan Kinerja Metode yang diusulkan dengan Metode Lain (Lanjutan)

No.	Sitasi	Parameter	Klasifikasi	Jumlah Data	Akurasi
7	Nurmayanti et al. (2021)	Jenis Kelamin, Peserta PKH, Jumlah Anggota Rumah Tangga, Status Kepemilikan Rumah, Fasilitas Rumah, Luas Lantai Rumah, dan Fasilitas Elektronik.	Naïve Bayes	168	96,63%
8	Ramadani et al. (2020)	Jumlah Tanggungan, Kepala Rumah Tangga, Status Tempat Tinggal, Jenis Lantai, Jenis Dinding, Pendapatan, dan Sumber Penerangan.	Naïve Bayes	20	-
9	Riyanah & Fatmawati (2021)	Pekerjaan, Penghasilan, Usia, Status, Kendaraan, Kepemilikan Rumah, dan Atap Bangunan.	Naïve Bayes	35	62,86%

Berdasarkan Tabel 8 dapat diketahui bahwa terdapat penelitian yang memiliki variabel dan jumlah data lebih sedikit dari penelitian ini. Hasil penelitian yang dilakukan oleh Annur (2018) dan Riyanah & Fatmawati (2021) memiliki tingkat akurasi yang lebih kecil dari penelitian yang dilakukan, sedangkan penelitian yang lainnya menghasilkan tingkat akurasi yang lebih besar dari penelitian yang dilakukan.

4. KESIMPULAN

Berdasarkan implementasi dan hasil pengujian pada penelitian penentuan kelayakan masyarakat miskin penerima bantuan dapat diambil beberapa kesimpulan yaitu metode Naïve Bayes dapat diterapkan pada sistem penentuan kelayakan masyarakat miskin penerima bantuan karena mendapatkan *accuracy* yang cukup baik, dan berdasarkan hasil pengujian menunjukkan bahwa metode Naïve Bayes dapat menghasilkan *accuracy* tertinggi pada scenario ketiga yaitu 60% atau 328 data latih dan 40% atau 218 data uji dengan nilai *accuracy* sebanyak 77,98%. Pada penelitian lebih lanjut dapat dikembangkan dengan menggunakan metode lain seperti *Artificial Neural Network*, *Support Vector Machine*, atau K-Nearest Neighbor untuk meningkatkan nilai akurasi.

DAFTAR PUSTAKA

Aji, A. (2019). *Penerapan Metode Naive Bayes untuk Mengklasifikasi Kelayakan Penerima Bantuan Beras Miskin (Studi Kasus: Kantor Kelurahan Desa Tegalyoso)* [Universitas Teknologi Yogyakarta]. http://eprints.uty.ac.id/2660/

Annur, H. (2018). Klasifikasi Masyarakat Miskin Menggunakan Metode Naive Bayes. *ILKOM Jurnal Ilmiah*, *10*(2), 160–165. https://doi.org/10.33096/ilkom.v10i2.303.160-165

Arifando, R., Hidayat, N., & Soebroto, A. A. (2017). Klasifikasi Calon Penerima Bantuan Keluarga Miskin Menggunakan Metode Learning Vector Quantization (LVQ) (Studi Kasus: Daerah Kecamatan Mlandingan, Situbondo). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *2*(6), 2173–2181. https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/1625

Badan Pusat Statistik. (2021). *Profil Kemiskinan di Penajam Paser Utara Tahun 2021*. Badan Pusat Statistik. https://ppukab.bps.go.id/pressrelease/2021/12/22/229/profil-kemiskinan-di-kabupaten-penajam-paser-utara-tahun-2021.html

Fitriani, E. (2020). Perbandingan Algoritma C4.5 dan Naïve Bayes untuk Menentukan Kelayakan Penerima Bantuan Program Keluarga Harapan. *SISTEMASI*, *9*(1), 103. https://doi.org/10.32520/stmsi.v9i1.596

Gunawan, B., Pratiwi, H. S., & Pratama, E. E. (2018). Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, *4*(2), 113. https://doi.org/10.26418/jp.v4i2.27526

Kurnia, F., Kurniawan, J., Fahmi, I., & Monalisa, S. (2019). Klasifikasi Keluarga Miskin Menggunakan Metode K-Nearest Neighbor Berbasis Euclidean Distance. *Seminar Nasional*

Teknologi Informasi Komunikasi Dan Industri, 230–239. https://ejournal.uin-suska.ac.id/index.php/SNTIKI/article/view/8089

Kurniawan, D. (2020). *Pengenalan Machine Learning dengan Python*. PT Elex Media Komputindo.

Maarif, M. R. (2016). Perbandingan Naïve Bayes Classifier dan Support Vector Machine untuk Klasifikasi Judul Artikel. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *1*(2), 90–93. https://doi.org/10.14421/jiska.2016.12-05

Muljono, Artanti, D. P., Syukur, A., Prihandono, A., & Setiadi, D. R. I. M. (2018). *Analisa Sentimen Untuk Penilaian Pelayanan Situs Belanja Online Menggunakan Algoritma Naïve Bayes*. 8–9.

Nastuti, A., & Harahap, S. Z. (2019). Teknik Data Mining untuk Penentuan Paket Hemat Sembako dan Kebutuhan Harian Dengan Menggunakan Algoritma FP-Growth (Studi Kasus di Ulfamart Lubuk Alung). *Jurnal Informatika*, *7*(3), 111–119. https://doi.org/10.36987/informatika.v7i3.1381

Natuzzuhriyyah, A., Nafisah, N., & Mayasari, R. (2021). Klasifikasi Tingkat Kepuasan Mahasiswa Terhadap Pembelajaran Secara Daring Menggunakan Algoritma Naïve Bayes. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *6*(3), 161–170. https://doi.org/10.14421/jiska.2021.6.3.161-170

Novriansyah, M. A. (2018). Pengaruh Pengangguran dan Kemiskinan Terhadap Pertumbuhan Ekonomi di Provinsi Gorontalo. *Gorontalo Development Review*, *1*(1), 59–73. https://doi.org/10.32662/GOLDER.V1I1.115

Nurmayanti, W. P., Saky, D. A. L., Malthuf, M., Gazali, M., & Hirzi, R. H. (2021). Penerapan Naive Bayes dalam Mengklasifikasikan Masyarakat Miskin di Desa Lepak. *Geodika: Jurnal Kajian Ilmu Dan Pendidikan Geografi*, *5*(1), 123–132. https://doi.org/10.29408/geodika.v5i1.3430

Purnama, A. I., Aziz, A., & Sartika Wiguna, A. (2020). Penerapan Data Mining untuk Mengklasifikasi Penerima Bantuan PKH Desa Wae Jare Menggunakan Metode Naïve Bayes. *KURAWAL Jurnal Teknologi, Informasi Dan Industri*, *3*(2), 173–180. https://jurnal.machung.ac.id/index.php/kurawal

Putri, A. C., Hariyanto, F. E., Andini, N. L. E., & Zulkarnaen, Z. C. S. (2021). Klasifikasi Rumah Tangga Miskin di Provinsi Papua Tahun 2017 Menggunakan Metode Naive Bayes. *Jurnal Sains Matematika Dan Statistika*, *7*(1), 89. https://doi.org/10.24014/jsms.v7i1.11924

Ramadani, S., Zannah, N., Ayu, S., Nurhayati, N., Azzahra, F., & Windarto, A. P. (2020). Analisis Data Mining Naive Bayes Klasifikasi Pada Kelayakan Penerima PKH. *KOMIK (Konferensi Nasional Teknologi Informasi Dan Komputer)*, *4*(1). https://doi.org/10.30865/KOMIK.V4I1.2726

Riyanah, N., & Fatmawati, F. (2021). Penerapan Algoritma Naive Bayes Untuk Klasifikasi Penerima Bantuan Surat Keterangan Tidak Mampu. *JTIM : Jurnal Teknologi Informasi Dan Multimedia*, *2*(4), 206–213. https://doi.org/10.35746/jtim.v2i4.117

Setiawan, D. A., Halilintar, R., & Wahyuniar, L. S. (2021). Penerapan Metode Naive Bayes Untuk Klasifikasi Penentuan Penerima Bantuan PKH. *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, *5*(2), 249–254. https://proceeding.unpkediri.ac.id/index.php/inotek/article/view/1137

Sugianto, C. A., & Maulana, F. R. (2019). Algoritma Naïve Bayes Untuk Klasifikasi Penerima Bantuan Pangan Non Tunai (Studi Kasus Kelurahan Utama). *Techno.Com*, *18*(4), 321–331. https://doi.org/10.33633/tc.v18i4.2587

Suntoro, J. (2019). *Data Mining Algoritma dan Implementasi dengan Pemrograman PHP*. PT Elex Media Komputindo. https://elexmedia.id/produk/detail/elexmedia2018-data-mining-algoritma-dan-implementasi-dengan-pemrograman-php/9786020498812

Umami, U. (2013). Cara Pandang dan Upaya Pemerintah dalam Mengurangi Kemiskinan. *Jurnal Pembangunan Wilayah & Kota*, *9*(4), 343. https://doi.org/10.14710/pwk.v9i4.6673

Wati, M., & Hadi, A. (2016). Implementasi Algoritma Naive Bayesian Dalam Penentuan Penerima Program Bantuan Pemerintah. *JTRISTE*, *3*(1), 22–26.

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023, Pp. 50 – 65
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Analisis Perbandingan Metode Pendukung Keputusan Pemilihan Kos Mahasiswa di Pontianak

Noerul Hanin [(1)*], David Jordy Dhandio [(2)], Della Zaria [(3)]
Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Tanjungpura, Pontianak
e-mail : h1091211018@student.untan.ac.id, {davidjordy2014,dellazariaa}@gmail.com.
* Penulis korespondensi.
Artikel ini diajukan 24 November 2022, direvisi 4 Januari 2023, diterima 7 Januari 2023, dan dipublikasikan 30 Januari 2023.

Abstract

The existence of boarding houses in public spaces is highly expected by the community, especially migrants such as students who need a temporary house in oversea areas. In Pontianak, especially around Tanjungpura University, there are many boarding houses that offer various facilities with various rental prices. Thus, decision support analysis is needed to choose a good boarding house for students around Tanjungpura University. In this study, two decision support system methods were selected, those are SAW and TOPSIS. These two methods were chosen because they have uncomplicated calculations, but are capable to produce good decisions. A comparison of the two methods was carried out to find out differences in results and calculation concepts to choose boarding houses for students in Pontianak. Data that was used for the trial were 10 alternative boarding houses located around the university. Based on trial results, the best boarding house obtained using SAW and TOPSIS methods is Yoga Kost.

Keywords: SAW, TOPSIS, Decision Support, Boarding House, Student

Abstrak

Keberadaan kos di ruang publik sangat diharapkan oleh masyarakat, khususnya perantau seperti mahasiswa yang membutuhkan tempat tinggal sementara di daerah rantauannya. Di Pontianak, khususnya di sekitar Universitas Tanjungpura, terdapat banyak kos yang menawarkan berbagai fasilitas dengan harga sewa yang beragam. Dengan demikian, diperlukan suatu analisis pendukung keputusan dalam pemilihan kos yang baik bagi mahasiswa di sekitar Universitas Tanjungpura. Pada penelitian ini, dipilih dua metode sistem pendukung keputusan, yaitu SAW dan TOPSIS. Kedua metode ini dipilih karena memiliki perhitungan yang tidak rumit, tetapi mampu menghasilkan keputusan yang baik. Perbandingan kedua metode dilakukan untuk mengetahui perbedaan hasil dan konsep perhitungan dalam memilih kos bagi mahasiswa di Pontianak. Data yang digunakan untuk uji coba sebanyak 10 data alternatif kos yang berada di sekitar universitas. Berdasarkan hasil uji coba, diperoleh hasil berupa keputusan kos terbaik dengan metode SAW dan TOPSIS ialah Yoga Kost.

Kata Kunci: SAW, TOPSIS, Pendukung Keputusan, Kos, Mahasiswa

1. PENDAHULUAN

Perkembangan teknologi dari waktu ke waktu menjadikan kehidupan manusia di muka bumi menjadi semakin mudah. Bagaimana tidak, kini jutaan informasi dapat mengalir dengan mudahnya di setiap lapisan masyarakat dalam waktu yang sesingkat-singkatnya (Sudi, 2019). Sayangnya, jutaan informasi yang tersalur ini juga dapat memberikan efek kesulitan bagi *decision maker* dalam mengambil keputusan karena hendak mencapai suatu keoptimalan melalui keputusan yang ditarik. Hal ini menjadikan timbulnya urgensi diperlukannya suatu sistem yang dapat mendukung *decision maker* mengambil keputusan optimal, yakni *decision support system*. Sistem ini dirancang sebagai pendukung setiap tahap pengambilan keputusan di antaranya tahap pengidentifikasian masalah, pemilihan data yang sesuai, penentuan pendekatan yang akan digunakan dalam pengambilan keputusan, dan terakhir pada tahap pengevaluasian pemilihan alternatif (Zulkifli & Sarifuddin, 2017). *Decision support* system telah diaplikasikan oleh banyak pihak dengan berdasar pada usaha pengoptimalan keputusan yang diambil, seperti sistem

pendukung keputusan penentuan karyawan terbaik oleh bos, kepala sekolah dalam menentukan pihak penerima beasiswa, dan pihak-pihak lainnya. Sistem pendukung keputusan ini juga telah diaplikasikan dalam menentukan kesesuaian lahan tanaman padi guna mempermudah petani dalam memilih lahan penanaman yang mumpuni (Wulandari et al., 2022).

Namun, jika dilihat lebih dalam, pihak utama yang sebenarnya memerlukan sistem pendukung keputusan pemilihan ini tak lain dan tak bukan adalah mahasiswa (Kolatlena & Riry, 2022). Tak perlu diragukan lagi, mahasiswa seringkali mengalami kesulitan dalam menentukan segala hal yang terkait dengan kehidupan perkuliahannya, salah satunya tempat tinggal. Padahal, hal ini merupakan faktor yang esensial bagi mahasiswa terutama dalam menunjang pendidikan yang ditempuh (Wardhani & Nur, 2017). Tempat tinggal sementara atau yang sering disebut kos merupakan kebutuhan vital bagi mahasiswa yang berasal dari luar maupun dalam kota dalam masa menempuh pendidikan. Sayangnya, kesulitan mendapatkan rumah kos yang sesuai dengan keinginan mahasiswa terkait adalah permasalahan yang seringkali dihadapi mahasiswa, terutama mahasiswa pendatang pada nyatanya (Pattriskak et al., 2020).

Problema mahasiswa kesulitan mencari kos yang tepat ini terjadi di banyak wilayah di Indonesia, termasuk Pontianak. Ketersediaan kos yang cukup menjamur di Pontianak, khususnya di sekitar Universitas Tanjungpura, memberikan kegelisahan mahasiswa selaku *decision maker* dalam menentukan tempat tinggal yang sesuai. Ketersediaan fasilitas yang memadai menjadi satu di antara banyak faktor pemilihan kos oleh mahasiswa (Wijoyo & Maimunah, 2019). Selain itu, didukung dengan faktor-faktor lainnya, seperti harga sewa, lokasi, keamanan, kebersihan, dan faktor lainnya (R. N. Sari & Hayati, 2019). Dengan tawaran berbagai macam jenis dan bentuk fasilitas tunjangan yang ditawarkan pihak pemilik kos inilah memberikan rasa bingung bagi mahasiswa dalam menentukan pilihan kosnya (Ayyasy et al., 2019). Untuk itu, diperlukan suatu sistem yang dapat membantu mendukung keputusan mahasiswa dalam memilih kos di Pontianak, khususnya Universitas Tanjungpura. Guna mendapatkan hasil analisis yang paling optimal, dilakukan perbandingan antara 2 metode, yakni metode *Simple Additive Weighting* (SAW) dan *Technique For Others Reference by Similarity to Ideal Solution* (TOPSIS). Melalui perbandingan ini diharapkan dapat mendukung keputusan pemilihan mahasiswa dalam menentukan kos di Pontianak khususnya di sekitar Universitas Tanjungpura dengan optimal.

Perbandingan antara kedua metode inilah, yakni SAW dan TOPSIS dalam mendukung pengambilan keputusan yang menjadi eksklusivitas yang dihadirkan pada penelitian ini, meskipun penelitian terkait sistem pendukung pengambilan keputusan tak dapat dipungkiri telah banyak diteliti. Selain itu, penitikberatan wilayah pada penelitian ini yang terarah pada kota Pontianak, khususnya sekitar Universitas Tanjungpura yang berurgensi untuk segera diselesaikan terkait problema kesulitan mahasiswa mencari kos yang sesuai melalui sistem pendukung pengambilan keputusan ini.

2. METODE PENELITIAN

Penelitian dilakukan dengan mengumpulkan data primer dan data sekunder yang terkait dengan topik penelitian. Data primer dikolektif secara langsung oleh peneliti melalui kuesioner yang disebarkan kepada para responden dan data sekunder yang bersumber dari kumpulan sumber data literatur tercakup jurnal dan artikel yang dapat dipertanggungjawabkan. Data-data ini dianalisis sedemikian rupa sehingga dapat menghasilkan output berupa suatu jawaban atas hal yang diteliti pada topik ini. Hasil analisis akan diimplementasikan pada rancangan sistem pendukung keputusan pemilihan kos mahasiswa di Pontianak. Adapun metode analisis yang hendak diuji pengaplikasiannya adalah metode *Simple Additive Weighting* (SAW) dan *Technique for Others Reference by Similarity to Ideal Solution* (TOPSIS). Hasil analisis kedua metode pendukung keputusan tersebut kemudian dibandingkan satu sama lain untuk mendapatkan metode pendukung keputusan yang lebih optimal di antara keduanya. Perbandingan ini dilakukan guna mendapatkan hasil akhir yang paling optimal dalam menimbang keputusan pemilihan kos mahasiswa di Pontianak, tepatnya di sekitar Universitas Tanjungpura.

2.1 *Simple Additive Weighting* (SAW)

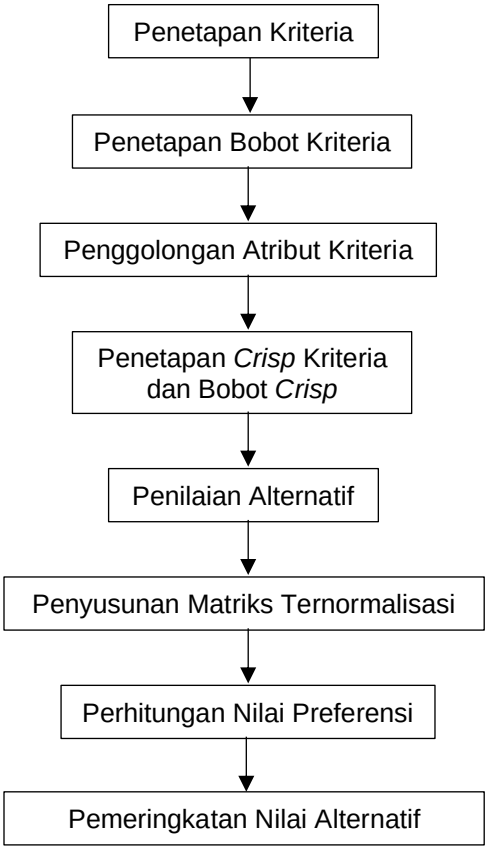
SAW merupakan metode yang memerlukan proses normalisasi di mana metode ini memperoleh hasil total perubahan nilai lebih banyak dibanding metode lainnya sehingga metode ini dapat dikatakan sangat relevan dalam penyelesaian problema pengambilan keputusan. Adapun rumus yang digunakan pada tahap normalisasi dan nilai preferensi berturut-turut sebagai berikut (Febriyati et al., 2016).

$$R_{ij} = \begin{cases} \frac{X_{ij}}{Max\ X_{ij}}, jika\ j\ adalah\ atribut\ benefit \\ \frac{Min\ X_{ij}}{X_{ij}}, jika\ j\ adalah\ atribut\ cost \end{cases} \quad (1)$$

$$V_i = \sum_{j=1}^{n} W_i r_{ij} \quad (2)$$

Pers. (1) adalah rumus untuk menghitung nilai *rating* kinerja ternormalisasi dari alternatif pada setiap kriteria. Hasil nilai ternormalisasi disimbolkan dengan R_{ij}. Sementara itu, Pers. (2) adalah rumus untuk menghitung nilai preferensi setiap alternatif yang disimbolkan dengan V_i. Adapun X_{ij} merupakan nilai alternatif ke-i dari kriteria ke-j.

Langkah pengerjaan menggunakan metode SAW dapat dilihat pada Gambar 1.



Gambar 1 Alur Pengerjaan Metode SAW

2.2 *Technique for Order Performance of Similarity to Ideal Solution* (TOPSIS)

TOPSIS merupakan metode yang berdasar pada alternatif mempunyai jarak terdekat dengan solusi ideal positif dan jarak terjauh dengan solusi ideal negatif dalam pengambilan keputusan

(Putra et al., 2020). Adapun rumus yang digunakan pada tahap-tahap pengaplikasian metode TOPSIS sebagai berikut (Doni et al., 2019).

1) Tahap pembuatan matriks keputusan ternormalisasi seperti pada Pers. (3).

$$R_{ij} = \frac{[X_{ij} - Min(X_{ij})]}{[Max(X_j) - Min(X_j]} \quad (3)$$

2) Tahap pembuatan matriks keputusan ternormalisasi terbobot seperti pada Pers. (4).

$$V_{ij} = r_{ij} w_{ij} = 1,2,3, \dots n; i = 1,2,3, \dots m \quad (4)$$

3) Tahap penentuan matriks solusi ideal positif dan negatif seperti pada Pers. (5) dan (6).

$$\{V_1^+, V_2^+, V_3^+, \dots V_n^+\}\{(Max_i V_{ij} | i = 1, \dots, m)\} \quad (5)$$

$$\{V_1^-, V_2^-, V_3^-, \dots V_n^-\}\{(Min_i V_{ij} | i = 1, \dots, m)\} \quad (6)$$

4) Tahap penentuan jarak setiap nilai alternatif melalui matriks solusi ideal positif dan negatif seperti pada Pers. (7) dan (8).

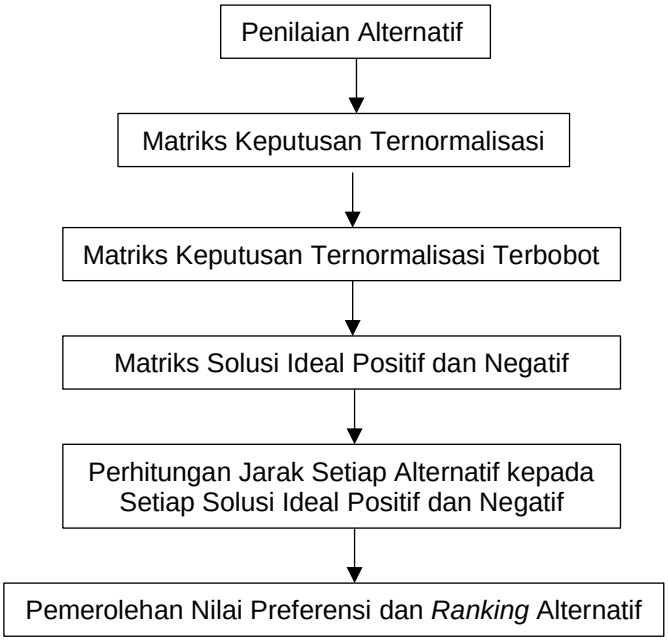
$$D_i^+ = (\sum_j^n (V_{ij} - V_j^+)^2)^{0.5} \quad (7)$$

$$D_i^- = (\sum_j^n (V_{ij} - V_j^-)^2)^{0.5} \quad (8)$$

5) Tahap penentuan nilai preferensi setiap alternatif seperti pada Pers. (9).

$$C_i = \frac{D_i^-}{D_i^- - D_i^+} ; i = 1,2,3, \dots m; 0 < C_i < 1 \quad (9)$$

Adapun langkah pengerjaan menggunakan metode TOPSIS dapat dilihat pada Gambar 2.



Gambar 2 Alur Pengerjaan Metode TOPSIS

3. HASIL DAN PEMBAHASAN

Setelah melakukan pengumpulan data, diperoleh nilai-nilai yang terkait dengan pendukung keputusan dalam pemilihan kos ideal di sekitar Universitas Tanjungpura, Pontianak. Nilai-nilai yang diperoleh selanjutnya dianalisis dengan metode SAW dan TOPSIS. Pada penelitian sebelumnya di tahun 2020, penggunaan metode TOPSIS terbukti mampu memberi keputusan yang lebih akurat dalam memilih pendaftar beasiswa terbaik yang layak untuk menerima beasiswa (W. E. Sari et al., 2021). Sementara itu, metode SAW yang diterapkan oleh Pramudhita (2107) dalam sistem pemilihan kos terbaik di Kota Malang terbukti mampu memberi keakuratan sebesar 100% (Pramudhita, 2107).

Berdasarkan penelitian sebelumnya yang telah dilakukan, penelitian ini menerapkan metode SAW dan TOPSIS untuk memilih kos terbaik dan mengetahui pemeringkatan sampel kos di sekitar Universitas Tanjungpura. Dari analisis yang dilakukan dengan metode SAW, dari 10 sampel kos yang diambil dengan pengkodean A1 hingga A10, diperoleh hasil berupa kos terbaik adalah alternatif A8 atau Yoga Kost. Dengan metode TOPSIS, diperoleh kos terbaik di antara alternatif A1 sampai dengan A10 ialah alternatif A1, yakni Kost Nadia Fikri. Perbedaan hasil di antara kedua metode dapat disebabkan oleh perbedaan metode perhitungan (Mahendra & Suprapto, 2020).

3.1 Tahapan Pengumpulan Data

Dari pengumpulan data yang dilakukan terhadap mahasiswa-mahasiswa dan beberapa kos di sekitar Universitas Tanjungpura, diperoleh data seperti pada Tabel 1.

Tabel 1 Data yang Digunakan

Kode Alternatif	Nama Kos	C1	C2	C3	C4
A1	Kost Nadia Fikri	1 km	Tempat tidur, lemari, kamar mandi luar	4,5 juta	Asri, bersih, tidak bising
A2	Muslimah	2 km	Tempat tidur, lemari, kamar mandi luar	5 juta	Tidak bersih, tidak bising
A3	Azhumy	1,6 km	Tempat tidur, lemari, kamar mandi luar	5,4 juta	Bersih, tidak bising
A4	Koster RS	1,7 km	Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	9 juta	Asri, bersih, tidak bising
A5	Kost Rawasari	6 km	Tempat tidur, lemari, kamar mandi dalam, wifi	7,2 juta	Bersih, tidak bising
A6	Kost 46	3,3 km	Tempat tidur, lemari, kamar mandi luar	6 juta	Asri, bersih, tidak bising
A7	Almadika	2 km	Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	12 juta	Asri, bersih, tidak bising
A8	Yoga Kost	4,1 km	Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	7,8 juta	Asri, bersih, tidak bising
A9	Kost Bu Nuraini	1,2 km	Tempat tidur, lemari, kamar mandi luar	4,98 juta	Tidak bersih, tidak bising
A10	Karya Baru Kost 09	1,5 km	Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	10,8 juta	Asri, bersih, tidak bising

3.2 Tahapan Penerapan Metode

3.2.1 Metode *Simple Additive Weighting* (SAW)

Metode SAW (*Simple Additive Weighting*) atau yang sering juga disebut sebagai metode penjumlahan terbobot merupakan metode pengambilan keputusan dengan mencari penjumlahan dari setiap atribut alternatif sesuai dengan bobot yang ditetapkan (Syahrudin & Yunita, 2021). Langkah-langkah dalam menerapkan metode SAW ialah sebagai berikut (Dhiki et al., 2022).

1) Penetapan Kriteria

Langkah pertama dalam tahap pengaplikasian metode adalah dengan menetapkan kriteria yang digunakan untuk menganalisis keputusan pemilihan kos terbaik. Kriteria (Cj) yang digunakan pada penelitian ini dapat dilihat pada Tabel 2.

Tabel 2 Kriteria pada Analisis SAW

Kode	Kriteria
C1	Jarak ke Kampus
C2	Harga
C3	Fasilitas
C4	Lingkungan

2) Penetapan Bobot Kriteria

Setelah kriteria ditetapkan, masing-masing kriteria selanjutnya diberikan bobot (Wj) sesuai dengan tingkat kepentingan setiap kriteria. Apabila kriteria memiliki kepentingan yang sama, maka setiap kriteria diberikan bobot dengan nilai yang sama pula. Bobot kriteria yang dimaksud dapat dilihat pada Tabel 3.

Tabel 3 Bobot Kriteria pada Analisis SAW

Cj	Wj	Bobot
C1	W1	5
C2	W2	5
C3	W3	5
C4	W4	5

3) Penggolongan Atribut Kriteria
Tahap selanjutnya ialah menggolongkan kriteria ke dalam jenis atribut benefit atau cost, yang mana penggolongannya dapat dilihat pada Tabel 4.

Tabel 4 Atribut Kriteria Metode SAW

Cj	Atribut
C1	Benefit
C2	Benefit
C3	Benefit
C4	Benefit

4) Penetapan *Crisp* Kriteria dan Bobot *Crisp*

Crisp adalah nilai rentang dari setiap kriteria. Pada tahap ini, setiap kriteria dibagi ke dalam *crisp* tertentu dan masing-masing *crisp* diberikan bobot sesuai tingkat kepentingannya. *Crisp* kriteria pada penelitian ini dapat dilihat pada Tabel 5 sampai 8.

Tabel 5 *Crisp* Jarak ke Kampus

C1 = Jarak ke Kampus	
Nilai	**Bobot**
0-1 km	1
1-2 km	0,8
2-3 km	0,6
3-4 km	0,4
> 4 km	0,2

Tabel 6 *Crisp* Harga Sewa

C2 = Harga Kos	
Nilai	**Bobot**
< 5 juta	1
5 - 7 juta	0,8
7 - 9 juta	0,6
9 - 11 juta	0,4
> 11 juta	0,2

Tabel 7 *Crisp* Fasilitas

C3 = Fasilitas Kos	
Nilai	**Bobot**
Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	1
Tempat tidur, lemari, kamar mandi dalam, wifi	0,8
Tempat tidur, lemari, kamar mandi dalam, kipas angin	0,6
Tempat tidur, lemari, kamar mandi dalam	0,4
Tempat tidur, lemari, kamar mandi luar	0,2

Tabel 8 *Crisp* Lingkungan

C4 = Lingkungan Kos	
Nilai	**Bobot**
Asri, bersih, tidak bising	1
Bersih, tidak bising	0,8
Bersih, bising	0,6
Tidak bersih, tidak bising	0,4
Tidak bersih, bising	0,2

5) Penilaian Alternatif

Dengan menggunakan data yang telah terkumpul sebagaimana terlampir pada Tabel 1, setiap alternatif diberikan penilaian sesuai dengan *crisp* yang telah ditetapkan. Adapun penilaian kriteria pada setiap alternatif dapat dilihat pada Tabel 9.

Setelah setiap alternatif diberi penilaian sesuai jenis dan *crisp* kriteria, alternatif tersebut dibobotkan dengan nilai pembobotan yang telah ditetapkan sebelumnya. Pembobotan alternatif tersebut dapat dilihat pada Tabel 10.

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023: 50 – 65

Tabel 9 Penilaian Alternatif

Nama Kos	C1	C2	C3	C4
Kost Nadia Fikri	0-1 km	Tempat tidur, lemari, kamar mandi luar	≤ 5 juta	Asri, bersih, tidak bising
Muslimah	1-2 km	Tempat tidur, lemari, kamar mandi luar	≤ 5 juta	Tidak bersih, tidak bising
Azhumy	1-2 km	Tempat tidur, lemari, kamar mandi luar	5-7 juta	Bersih, tidak bising
Koster RS	1-2 km	Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	7-9 juta	Asri, bersih, tidak bising
Kost Rawasari	> 4 km	Tempat tidur, lemari, kamar mandi dalam, wifi	7-9 juta	Bersih, tidak bising
Kost 46	3-4 km	Tempat tidur, lemari, kamar mandi luar	5-7 juta	Asri, bersih, tidak bising
Almadika	1-2 km	Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	> 11 juta	Asri, bersih, tidak bising
Yoga Kost	> 4 km	Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	7-9 juta	Asri, bersih, tidak bising
Kost Bu Nuraini	1-2 km	Tempat tidur, lemari, kamar mandi luar	≤ 5 juta	Tidak bersih, tidak bising
Karya Baru Kost 09	1-2 km	Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin	9-11 juta	Asri, bersih, tidak bising

Tabel 10 Pembobotan Alternatif pada Metode SAW

Kode Alternatif	C1	C2	C3	C4
A1	1	0,2	1	1
A2	0,8	0,2	1	0,5
A3	0,8	0,2	0,8	0,8
A4	0,8	1	0,6	1
A5	0,2	0,7	0,6	0,8
A6	0,4	0,2	0,8	1
A7	0,8	1	0,2	1
A8	0,2	1	0,6	1
A9	0,8	0,2	1	0,5
A10	0,8	1	0,4	1

6) Penyusunan Matriks Ternormalisasi

Alternatif yang telah mengalami pembobotan pada setiap kriteria selanjutnya dinormalisasi untuk mendapatkan rentang nilai pembobotan yang sama antar kriteria. Langkah normalisasi dilakukan dengan membagi nilai masing-masing bobot dengan nilai bobot maksimum pada kriteria Cj sehingga didapatkan matriks normalisasi seperti pada Tabel 11.

Tabel 11 Matriks Ternormalisasi Metode SAW

Kode Alternatif	C1	C2	C3	C4
A1	0,2	0,2	0,2	1
A2	0,25	0,2	0,2	0,5
A3	0,25	0,2	0,25	0,8
A4	0,25	1	0,33	1
A5	1	0,7	0,33	0,8
A6	0,5	0,2	0,25	1
A7	0,25	1	1	1
A8	1	1	0,33	1
A9	0,25	0,2	0,2	0,5
A10	0,25	1	0,5	1

7) Perhitungan Nilai Preferensi

Nilai preferensi (Vj) dihitung dengan mengalikan hasil normalisasi pada setiap kriteria dengan bobot kriteria pada Tabel 3 kemudian menjumlahkan hasil perkalian dalam satu alternatif yang sama. Perhitungan nilai preferensi pada penelitian ini ialah sebagai berikut.

Preferensi A1 = V1 = $(5 \times 0{,}2) + (5 \times 0{,}2) + (5 \times 0{,}2) + (5 \times 1) = 8$
Preferensi A2 = V2 = $(5 \times 0{,}25) + (5 \times 0{,}2) + (5 \times 0{,}2) + (5 \times 0{,}5) = 5{,}75$
Preferensi A3 = V3 = $(5 \times 0{,}25) + (5 \times 0{,}2) + (5 \times 0{,}25) + (5 \times 0{,}8) = 7{,}5$
Preferensi A4 = V4 = $(5 \times 0{,}25) + (5 \times 1) + (5 \times 0{,}33) + (5 \times 1) = 12{,}92$
Preferensi A5 = V5 = $(5 \times 1) + (5 \times 0{,}7) + (5 \times 0{,}33) + (5 \times 0{,}8) = 14{,}17$
Preferensi A6 = V6 = $(5 \times 0{,}5) + (5 \times 0{,}2) + (5 \times 0{,}25) + (5 \times 1) = 9{,}75$
Preferensi A7 = V7 = $(5 \times 0{,}25) + (5 \times 1) + (5 \times 1) + (5 \times 1) = 16{,}25$
Preferensi A8 = V8 = $(5 \times 1) + (5 \times 1) + (5 \times 0{,}33) + (5 \times 1) = 16{,}67$
Preferensi A9 = V9 = $(5 \times 0{,}25) + (5 \times 0{,}2) + (5 \times 0{,}2) + (5 \times 0{,}5) = 5{,}75$
Preferensi A10 = V10 = $(5 \times 0{,}25) + (5 \times 1) + (5 \times 0{,}5) + (5 \times 1) = 13{,}75$

Nilai preferensi dari setiap alternatif dirangkum pada Tabel 12.

Tabel 12 Nilai Preferensi Alternatif dengan Metode SAW

Vj	Nilai V
V1	8
V2	5,75
V3	7,5
V4	12,92
V5	14,17
V6	9,75
V7	16,25
V8	16,67
V9	5,75
V10	13,75

8) Pemeringkatan Alternatif

Berdasarkan nilai preferensi yang dihitung sebelumnya, diperoleh peringkat atau *ranking* dari setiap alternatif, di mana alternatif dengan nilai preferensi paling tinggi menempati peringkat pertama (paling tinggi), dan seterusnya. Pemeringkatan dengan metode SAW dapat dilihat pada Tabel 13.

Tabel 13 eringkat Alternatif dengan Metode SAW

Ranking	Kode Alternatif	Nama Kos
1	A8	Yoga Kost
2	A7	Almadika
3	A5	Kost Rawasari
4	A10	Karya Baru Kost 09
5	A4	Koster RS
6	A6	Kost 46
7	A1	Kost Nadia Fikri
8	A3	Azhumy
9,5	A2	Muslimah
9,5	A9	Kost Bu Nuraini

Dari pemeringkatan di atas, diperoleh hasil keputusan berupa kos terbaik di sekitar Universitas Tanjungpura dengan metode SAW ialah Yoga Kost (A8), disusul Almadika (A7), dan Kost Rawasari (A5).

3.2.2 Metode *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS)

Metode TOPSIS (*Technique for Order Preference by Similarity to Ideal Solution*) adalah metode pengambilan keputusan dengan pengukuran kinerja relatif dari alternatif keputusan terbaik memiliki jarak terdekat dengan solusi ideal positif dan jarak terjauh dari solusi ideal negative (Mutmainah & Yunita, 2021). Langkah-langkah dalam penerapan metode TOPSIS pada penelitian ini yaitu sebagai berikut (Wijaya, 2022).

1) Menentukan Kriteria Beserta Atribut dan Bobotnya

Langkah pertama dalam penelitian ini adalah menentukan kriteria beserta identitasnya (atribut dan bobot). Kriteria dalam penelitian ini diperlukan sebagai dasar dalam pengambilan keputusan pemilihan kos mahasiswa terbaik. Penentuan kriteria beserta identitasnya dapat dilihat pada Tabel 14.

Tabel 14 Data Kriteria Beserta Atribut dan Bobotnya

Nama Kriteria	Sub Kriteria	Atribut	Bobot
Lokasi/Jarak Kos ke Kampus (C1)	• 0-1 km (1) • 1-2 km (0,8) • 2-3 km (0,6) • 3-4 km (0,4) • > 4 km (0,2)	Benefit	5
Fasilitas (C2)	• Tempat tidur, lemari, kamar mandi dalam, wifi, AC/kipas angin (1) • Tempat tidur, lemari, kamar mandi dalam, wifi (0,7) • Tempat tidur, lemari, kamar mandi dalam, kipas angin (0,7) • Tempat tidur, lemari, kamar mandi dalam (0,4) • Tempat tidur, lemari, kamar mandi luar (0,2)	Benefit	5
Harga (C3)	• ≤ 5 juta (1) • 5-7 juta (0,8) • 7-9 juta (0,6) • 9-11 juta (0,4) • > 11 juta (0,2)	Benefit	5
Lingkungan (C4)	• Asri, bersih, tidak bising (1) • Bersih, tidak bising (0,8) • Bersih, bising (0,5) • Tidak bersih, tidak bising (0,5) • Tidak bersih, bising (0,2)	Benefit	5

2) Menetapkan Alternatif Beserta Nilai Alternatifnya

Setelah menentukan kriteria beserta atribut dan bobotnya, langkah selanjutnya adalah menentukan nilai alternatifnya beserta dengan data yang digunakan. Data nilai alternatif tersebut dapat dilihat pada Tabel 15.

Tabel 15 Data Alternatif Beserta Nilai Alternatifnya

Alternatif	C1	C2	C3	C4
A1	1	0,2	1	1
A2	0,8	0,2	1	0,5
A3	0,8	0,2	0,8	0,8
A4	0,8	1	0,6	1
A5	0,2	0,7	0,6	0,8
A6	0,4	0,2	0,8	1
A7	0,8	1	0,2	1
A8	0,2	1	0,6	1
A9	0,8	0,2	1	0,5
A10	0,8	1	0,4	1

3) Membuat Matriks Keputusan yang Ternormalisasi

Tahapan selanjutnya adalah membuat matriks keputusan yang ternormalisasi. Tahapan tersebut dilakukan dengan mengkuadratkan setiap elemen matriks. Hasil keseluruhan kuadrat setiap elemen matriks dapat dilihat pada Tabel 16.

Tabel 16 Data Keseluruhan Kuadrat Setiap Elemen Matriks

Alternatif	C1	C2	C3	C4
A1	1	0,04	1	1
A2	0,64	0,04	1	0,25
A3	0,64	0,04	0,64	0,64
A4	0,64	1	0,36	1
A5	0,04	0,49	0,36	0,64
A6	0,16	0,04	0,64	1
A7	0,64	1	0,04	1
A8	0,04	1	0,36	1
A9	0,64	0,04	1	0,25
A10	0,64	1	0,16	1
Total	**5,080**	**4,690**	**5,560**	**7,780**
Pembagi	**2,254**	**2,166**	**2,358**	**2,789**

Berikutnya adalah membuat matriks keputusan yang ternormalisasi dengan membagi setiap kuadrat elemen matriks dengan nilai pembagi. Nilai pembagi didapat dari mengkuadratkan nilai total setiap kolom kriteria. Hasil matriks keputusan yang ternormalisasi tersaji pada Tabel 17.

Tabel 17 Matriks Keputusan yang Ternormalisasi

Alternatif	C1	C2	C3	C4
A1	0,444	0,092	0,424	0,359
A2	0,355	0,092	0,424	0,179
A3	0,355	0,092	0,339	0,287
A4	0,355	0,462	0,254	0,359
A5	0,089	0,323	0,254	0,287
A6	0,177	0,092	0,339	0,359
A7	0,355	0,462	0,085	0,359
A8	0,089	0,462	0,254	0,359
A9	0,355	0,092	0,424	0,179
A10	0,355	0,462	0,170	0,359

4) Membuat Matriks Keputusan yang Ternormalisasi Terbobot

Matriks keputusan yang ternormalisasi terbobot ditentukan dengan mengalikan nilai tiap elemen matriks keputusan yang ternormalisasi dengan bobot masing-masing kriteria yang terdapat pada Tabel 1. Hasil matriks keputusan yang ternormalisasi terbobot dapat dilihat pada Tabel 18.

Tabel 18 Matriks Keputusan yang Ternormalisasi Terbobot

Alternatif	C1	C2	C3	C4
A1	2,218	0,462	2,120	1,793
A2	1,755	0,462	2,120	0,896
A3	1,755	0,462	1,696	1,434
A4	1,775	2,309	1,272	1,793
A5	0,444	1,616	1,272	1,434
A6	0,887	0,462	1,696	1,793
A7	1,775	2,309	0,424	1,793
A8	0,444	2,309	1,272	1,793
A9	1,775	0,462	2,120	0,896
A10	1,775	2,309	0,848	1,793

5) Menentukan Matriks Solusi Ideal

Matriks solusi ideal ditentukan dari atribut kriteria dan matriks keputusan ternormalisasi terbobot yang telah dibuat pada tahapan sebelumnya. Solusi ideal yang ditentukan adalah solusi ideal positif dan negatif. Solusi ideal positif adalah nilai maksimum dari tiap kolom kriteria pada matriks keputusan ternormalisasi terbobot jika atribut kriteria benefit, dan nilai minimum jika atribut kriteria cost. Sedangkan solusi ideal negatif adalah nilai minimum dari tiap kolom kriteria pada matriks keputusan ternormalisasi terbobot jika atribut kriteria benefit, dan nilai maksimum jika atribut kriteria cost. Pada penelitian ini, kriteria yang memiliki atribut benefit adalah fasilitas (C2) dan lingkungan (C4), sehingga solusi ideal positifnya adalah nilai maksimum dan negatifnya adalah nilai minimum. Dan pada penelitian ini juga, kriteria yang memiliki atribut cost adalah lokasi/jarak (C1) dan harga (C3), sehingga solusi ideal positifnya adalah nilai minimum dan negatifnya adalah nilai maksimum. Hasil dari solusi ideal dapat dilihat pada Tabel 19.

Tabel 19 Matriks Solusi Ideal

Solusi Ideal	C1	C2	C3	C4
Positif (D_i^+)	0,444	2,309	0,424	1,793
Negatif (D_i^-)	2,218	0,462	2,120	0,896

6) Menentukan Jarak Alternatif dengan Solusi Ideal

Tabel 20 Jarak Alternatif dengan Solusi Ideal

Alternatif	Positif	Negatif
A1	3,072	0,896
A2	2,977	0,444
A3	2,633	0,816
A4	1,578	2,265
A5	1,152	2,343
A6	2,286	1,660
A7	1,331	2,700
A8	0,848	2,843
A9	2,977	0,444
A10	1,397	2,456

Jarak alternatif dengan solusi ideal ditentukan dengan menjumlahkan tiap alternatif dari kuadrat selisih antara tiap elemen kriteria pada matriks ternormalisasi terbobot dengan masing-masing matriks solusi ideal. Hasilnya dapat dilihat pada Tabel 20.

7) Menentukan Nilai Preferensi Setiap Alternatif

Langkah terakhir dalam metode TOPSIS pada penelitian ini adalah menentukan nilai preferensi setiap alternatif serta melakukan pemeringkatan. Nilai preferensi didapatkan dari membagi solusi ideal negatif dengan penjumlahan solusi ideal negatif dan solusi ideal positif. Peringkat diurutkan dari nilai preferensi tertinggi hingga terendah. Hasilnya dapat dilihat pada Tabel 21.

Tabel 21 Nilai Preferensi Alternatif dan Peringkatnya

Alternatif	Positif	Negatif	Preferensi	Peringkat
A1	3,072	0,896	0,2258	8
A2	2,977	0,444	0,1297	9,5
A3	2,633	0,816	0,2366	7
A4	1,578	2,265	0,5894	5
A5	1,152	2,343	0,6704	2
A6	2,286	1,660	0,4206	6
A7	1,331	2,700	0,6698	3
A8	0,848	2,843	0,7702	1
A9	2,977	0,444	0,1297	9,5
A10	1,397	2,456	0,6374	4

Pemilihan kos terbaik berdasarkan pada alternatif yang memiliki *ranking* tertinggi pada nilai preferensi alternatif. Berdasarkan nilai tersebut, jika diambil 3 *ranking* tertinggi, maka penentuan kos terbaik adalah Yoga Kost (A8), Kost Rawasari (A5), dan Almadika (A7).

3.3 Perbandingan Metode SAW dan TOPSIS

Berikut adalah perbandingan hasil akhir pengujian dengan metode SAW dan TOPSIS yang dapat dilihat pada Tabel 22.

Tabel 22 Perbandingan Nilai Preferensi Metode SAW dan TOPSIS

Peringkat	SAW		TOPSIS	
	Alternatif	Preferensi	Alternatif	Preferensi
1	A8	16,6667	A8	0,7702
2	A7	16,2500	A5	0,6704
3	A5	14,1667	A7	0,6698
4	A10	13,7500	A10	0,6374
5	A4	12,9167	A4	0,5894
6	A6	9,7500	A6	0,4206
7	A1	8,0000	A3	0,2366
8	A3	7,5000	A1	0,2258
9,5	A2	5,7500	A2	0,1297
9,5	A9	5,7500	A9	0,1297

Berdasarkan hasil perhitungan dengan metode SAW dan TOPSIS, didapatkan hasil bahwa kedua metode tersebut memberikan peringkat alternatif yang hampir serupa. Perbedaan *ranking* hanya terdapat antara alternatif A5 dan A7 serta A1 dan A3. Pada metode TOPSIS, A5 mendapat peringkat 2 dan A7 mendapat peringkat 3, sedangkan pada metode SAW berlaku sebaliknya, di mana A5 mendapat peringkat 3 dan A7 mendapat peringkat 2. Kemudian, perbedaan juga terdapat antara A1 dan A3, di mana pada metode TOPSIS keduanya berturut-turut mendapat peringkat 8 dan 7, sementara pada metode SAW mendapat peringkat 7 dan 8.

Berdasarkan definisi dari metode TOPSIS yang menetapkan bahwa alternatif terbaik ialah altermatif yang memiliki jarak terpendek dari solusi ideal positif dan jarak terjauh dari solusi ideal negatif (Saputra & Pakereng, 2020). Tabel 3 dan Tabel 14 telah menunjukkan bahwa kedua metode tersebut diberikan bobot yang sama untuk setiap kriterianya. Oleh karena itu, berdasarkan konsep TOPSIS, dapat dibandingkan jarak setiap nilai ternormalisasi dan terbobot untuk alternatif yang memiliki perbedaan *ranking* pada metode SAW dan TOPSIS. Perbandingan nilai ternormalisasi dan terbobot serta nilai solusi ideal positif untuk A5 dan A7 dapat dilihat pada Tabel 23.

Tabel 23 Perbandingan Nilai untuk A5 dan A7

	C1	C2	C3	C4
A5	0,444	1,616	1,272	1,434
A7	1,775	2,309	0,424	1,793
D_i^+	0,444	2,309	0,424	1,793

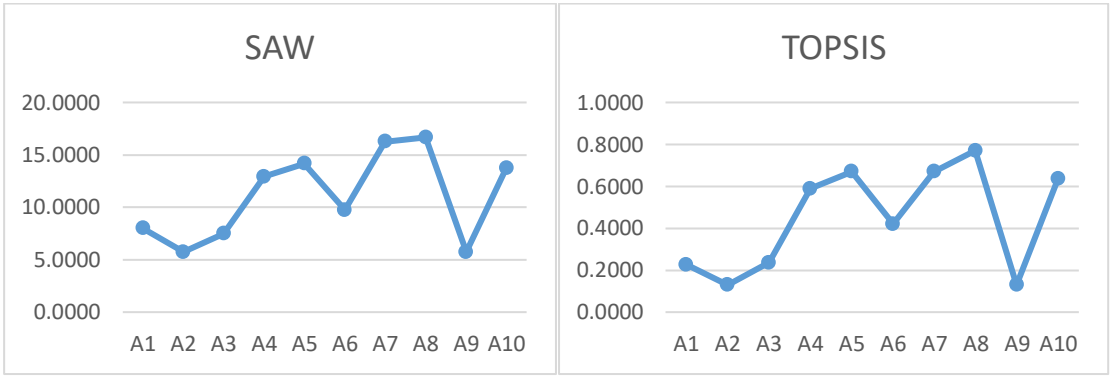
Sementara itu, perbandingan nilai ternormalisasi dan terbobot serta nilai solusi ideal positif untuk A1 dan A3 dapat dilihat pada Tabel 24.

Tabel 24 Perbandingan Nilai untuk A1 dan A3

	C1	C2	C3	C4
A1	2,218	0,462	2,120	1,793
A3	1,775	0,462	1,696	1,434
D_i^+	0,444	2,309	0,424	1,793

Berdasarkan perbandingan pada Tabel 23 dan Tabel 24, dapat dilihat bahwa terdapat berbedaan alternatif yang memiliki jarak terdekat per kriteria. Berdasarkan perbandingan pada Tabel 23, alternatif A5 merupakan solusi dengan jarak terdekat untuk kriteria C1, sementara alternatif A7 merupakan solusi dengan jarak terdekat untuk kriteria C2, C3, dan C4. Kemudian, dari Tabel 24 juga dapat dilihat perbedaan, di mana alternatif A1 merupakan solusi dengan jarak terdekat untuk kriteria C4, sementara alternatif A3 merupakan solusi dengan jarak terdekat untuk kriteria C1 dan C3 (untuk kriteria C2 nilai kedua alternatif sama sehingga tidak dibandingkan). Karena adanya perbedaan kedekatan per kriteria inilah kedua alternatif, yakni A1 dan A3, serta A5 dan A7 memiliki yang berbeda pada metode SAW dan TOPSIS.

Selain itu, perbandingan antara kedua metode juga dapat dilihat dari keragaman atau persebaran nilai preferensinya. Grafik sebaran nilai preferensi alternatif dengan menngunakan metode SAW dan TOPSIS dapat dilihat pada Gambar 3.



Gambar 3 Grafik Persebaran Nilai Preferensi Alternatif

Dari Gambar 3 terlihat bahwa nilai preferensi alternatif dengan metode SAW dan TOPSIS memiliki sebaran yang hampir serupa. Namun, jika dihitung nilai koefisen variansinya, maka

metode TOPSIS menghasilkan nilai koefisien variansi yang lebih besar, yakni 55,49%. Sementara itu, metode SAW memiliki nilai koefisien variansi sebesar 38,02%. Semakin besar koefisien variansi, maka semakin besar keragaman nilai datanya (Yusniyanti & Kurniati, 2017). Oleh karena itu, berdasarkan kriteria tersebut dan pada kasus data pada penelitian ini, metode TOPSIS lebih baik dibandingkan metode SAW.

4. KESIMPULAN

Pemanfaatan metode pendukung keputusan dapat mempermudah para pencari kos, khususnya mahasiswa untuk memilih kos yang sesuai dengan kebutuhan. Metode yang digunakan dalam penelitian ini adalah metode SAW dan TOPSIS. Proses penyelesaian metode dimulai dari menentukan kriteria, melakukan pembobotan kriteria, mencari data alternatif, perhitungan, serta pemeringkatan.

Sistem pendukung keputusan memberikan alternatif pemilihan kos terbaik berdasarkan dengan kriteria-kriteria yang telah ditentukan. Kriteria yang digunakan berdasarkan pada hasil survei terbanyak dan dapat dipertanggungjawabkan. Dengan kriteria yang ditetapkan, diperoleh hasil kos terbaik berdasarkan metode SAW dan TOPSIS adalah Yoga Kost. Berdasarkan keragaman nilai preferensi, disimpulkan bahwa metode TOPSIS lebih baik dibandingkan metode SAW dalam mendukung keputusan pemilihan kos mahasiswa di Pontianak pada penelitian ini. Saran untuk penelitian selanjutnya yaitu dengan menambah indikator kriteria pemilihan kos dan melibatkan variabel-variabel keputusan yang lebih beragam.

DAFTAR PUSTAKA

Ayyasy, M. F., Sari, P. R. K., & Maradita, F. (2019). Faktor- Faktor yang Mempengaruhi Keputusan Mahasiswa untuk Mengontrak Tempat Tinggal (Studi Kasus Mahasiswa UTS Nusantara 2016). *Jurnal Manajemen Dan Bisnis*, *2*(2), 80–89. https://doi.org/10.37673/JMB.V2I2.527

Dhiki, T. E., Londa, M. A., & Radja, M. (2022). Sistem Pendukung Keputusan Pemilihan Kost Di Sekitaran Kampus Universitas Flores Menggunakan Metode Simple Additive Weighting (Saw). *JUPITER (Jurnal Penelitian Ilmu Dan Teknik Komputer)*, *14*(2-b), 413–422. https://doi.org/10.5281./5148/5.jupiter.2022.10

Doni, R., Amir, F., & Juliawan, D. (2019). Sistem Pendukung Keputusan Kenaikan Jabatan Menggunakan Metode Technique for Order Preference by Similarity to Ideal Solution (TOPSIS). *Prosiding Seminar Nasional Riset Information Science (SENARIS)*, *1*(0), 69–75. https://doi.org/10.30645/SENARIS.V1I0.9

Febriyati, M. N., Sophan, Moch. K., & Yunitarini, R. (2016). Perbandingan SAW dan TOPSIS untuk Open Recruitment Warga Laboratorium Teknik Informatika di Universitas Trunojoyo Madura. *Jurnal Simantec*, *5*(3). https://doi.org/10.21107/SIMANTEC.V5I3.2348

Kolatlena, R. S., & Riry, W. A. (2022). Sistem Penunjang Keputusan Pemilihan Mahasiswa Berprestasi Menggunakan Metode Profile Matching. *SANISA: Jurnal Kreativitas Mahasiswa Hukum*, *2*(1), 24–31. https://fhukum.unpatti.ac.id/jurnal/sanisa/article/view/995

Mahendra, I., & Suprapto, A. (2020). Penerapan Metode TOPSIS & SAW Dalam Pemilihan Destinasi Wisata Di Jawa Timur. *INFORMAL: Informatics Journal*, *5*(1), 18–25. https://doi.org/10.19184/ISJ.V5I1.15311

Mutmainah, I., & Yunita, Y. (2021). Penerapan Metode Topsis Dalam Pemilihan Jasa Ekspedisi. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, *10*(1), 86–92. https://doi.org/10.32736/SISFOKOM.V10I1.1028

Pattriskak, B. E. G., Santosa, G. R., & Chrismanto, A. R. (2020). Implementasi Algoritma Dijkstra untuk Mencari Rumah Kost Terdekat di Kodya Yogyakarta Berbasis Android. *Jurnal Terapan Teknologi Informasi*, *4*(1), 45–54. https://doi.org/10.21460/JUTEI.2020.41.193

Pramudhita, A. (2107). Sistem Pendukung Keputusan Pemilihan Rumah Kost Putra untuk Mahasiswa di Kota Malang dengan Menggunakan Metode SAW. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *1*(1), 906–912. https://doi.org/10.36040/JATI.V1I1.2085

Putra, D. W. T., Santi, S. N., Swara, G. Y., & Yulianti, E. (2020). Metode TOPSIS dalam Sistem Pendukung Keputusan Pemilihan Objek Wisata. *Jurnal Teknoif Teknik Informatika Institut*

Teknologi Padang, *8*(1), 1–6. https://doi.org/10.21063/jtif.2020.V8.1.1-6

Saputra, G. T., & Pakereng, M. A. I. (2020). Analisis Perbandingan Metode TOPSIS dan SAW pada Penilaian Karyawan (Studi Kasus : PT Pura Barutama Unit Paper Mill 5, 6, 9). *Jurnal Informatika*, *7*(2), 156–165. https://doi.org/10.31294/JI.V7I2.8612

Sari, R. N., & Hayati, R. S. (2019). Penerapan Metode Simple Additive Weighting Dalam Pemilihan Rumah Kost. *CogITo Smart Journal*, *5*(2), 215–226. https://doi.org/10.31154/cogito.v5i2.217.215-226

Sari, W. E., B, M., & Rani, S. (2021). Perbandingan Metode SAW dan Topsis pada Sistem Pendukung Keputusan Seleksi Penerima Beasiswa. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, *10*(1), 52–58. https://doi.org/10.32736/sisfokom.v10i1.1027

Sudi, M. (2019). Implikasi Perkembangan Teknologi Komunikasi Terhadap Peradaban dan Komunikasi Antar Manusia. *Gema Kampus IISIP YAPIS Biak*, *13*(2), 33–46. https://doi.org/10.52049/gemakampus.v13i2.68

Syahrudin, S., & Yunita, S. (2021). Sistem Pendukung Keputusan Pemilihan Tempat Kost Menggunakan Metode Simple Addtive Weighting (SAW) Kotawaringin Timur. *KLIK: Kajian Ilmiah Informatika Dan Komputer*, *2*(2), 84–87. https://djournals.com/klik/article/view/227

Wardhani, N., & Nur, M. A. (2017). Sistem Pendukung Keputusan Pemilihan Tempat Kos untuk Mahasiswa di Luwuk Banggai dengan Metode SAW (Simple Additive Weighting). *JTRISTE*, *4*(1), 9–14.

Wijaya, R. (2022). Sistem Pendukung Keputusan Pemilihan Indekos Terbaik Bagi Mahasiswa Menggunakan Metode Topsis. *Jurnal Ilmiah Core IT : Community Research Information Technology*, *10*(4), 1978–1520. https://ijcoreit.org/index.php/coreit/article/view/362

Wijoyo, S., & Maimunah, E. (2019). Faktor-faktor Pertimbangan Mahasiswa UNILA dalam Pemilihan Rumah Indekos dikelurahan Kampung Baru dan Gedung Meneng Bandar Lampung. *Jurnal Ekonomi Pembangunan*, *8*(1), 45–55. https://doi.org/10.23960/jep.v8i1.35

Wulandari, S. R., Hamdani, H., & Septiarini, A. (2022). Sistem Pendukung Keputusan Kesesuaian Lahan Tanaman Padi Menggunakan Metode AHP dan SAW. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *7*(3), 226–236. https://doi.org/10.14421/jiska.2022.7.3.226-236

Yusniyanti, E., & Kurniati, K. (2017). Analisa Puncak Banjir Dengan Metode MAF (Studi Kasus Sungai Krueng Keureuto). *EINSTEIN E-JOURNAL*, *5*(1). https://doi.org/10.24114/einstein.v5i1.7224

Zulkifli, Z., & Sarifuddin, S. (2017). Decision Support System Pemberian Bonus Tahunan pada Karyawan Berdasarkan Kinerja Karyawan Menggunakan Metode Simple Additive Weighting (Study Kasus : STIMIK Pringsewu). *Jurnal TAM (Technology Acceptance Model)*, *7*, 67–73. https://doi.org/10.56327/JURNALTAM.V7I0.74

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023, Pp. 66 – 77
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Penerapan Algoritma K-Means untuk Klasterisasi Penduduk Miskin pada Kota Pagar Alam

Febriansyah Febriansyah [(1)*], Siti Muntari [(2)]
Teknik Informatika, Institut Teknologi Pagar Alam, Pagar Alam
e-mail : {febriansyahh1213,muntariaza}@gmail.com.
* Penulis korespondensi.
Artikel ini diajukan 26 September 2022, direvisi 24 Januari 2023, diterima 25 Januari 2023, dan dipublikasikan 30 Januari 2023.

Abstract

The purpose of this study was to obtain a poverty data cluster in Pagar Alam City. The data collection of beneficiaries of the Program Keluarga Harapan (PKH) is not correct, the provision of assistance only pays attention to the criteria for poverty in general, so there are still many poor people who feel more deserving of PKH assistance. To overcome the problem of PKH recipients, it is necessary to cluster the community into various levels, so that the government can know the level of poverty of the community and can provide PKH assistance appropriately. The methods used in this study are CRISP-DM and the K-Means clustering algorithm. The attributes used are Identity Number, Name, Family Family Card Number, Poverty Rate, Pregnant Women, Early Childhood, Elementary School, Junior High School, Senior High School, Elderly, and Family Hope Program Recipient Group. This clustering process produced three clusters, namely cluster_0 as many as 156 people, cluster_1 as many as 82 people, and cluster_2 as many as 233 people. Furthermore, it was developed into a system with the Rapid Application Development (RAD) system development method. Thus producing a K-Means algorithm system to classify the poor in Pagar Alam City. The system test method uses black box testing with the alpha method and obtained database test results with a value of 4, interfaces with a value of 4, functionality of 4.42, and algorithms with a value of 4. In the testing process with UAT, in the system aspect got 87% of users agreed, in the user aspect 86% agreed, and in the interaction aspect 87% of users agreed. So it can be concluded that this system is worth using.

Keywords: Data Mining, Poverty, K-Means, Clustering, Black Box

Abstrak

Tujuan dari penelitian ini adalah untuk mendapatkan klaster data kemiskinan di Kota Pagar Alam. Pendataan penerima bantuan Program Keluarga Harapan (PKH) belum tepat, pemberian bantuan hanya memperhatikan kriteria kemiskinan secara umum, sehingga masih banyak masyarakat miskin yang merasa lebih pantas dapat bantuan PKH. Untuk mengatasi masalah penerima PKH tersebut perlu adanya pengklasteran untuk membagi masyarakat ke dalam berbagai tingkatan, sehingga pemerintah dapat mengetahui tingkat kemiskinan masyarakat dan dapat memberikan bantuan PKH secara tepat. Metode yang digunakan dalam penelitian ini ialah *CRISP-DM* dan algoritma *clustering* K-Means. Atribut yang digunakan adalah Nomor Induk Kependudukan, Nama, Nomor Kartu Keluarga Keluarga, Tingkat Kemiskinan, Ibu Hamil, Usia Dini, Sekolah Dasar, Sekolah Menengah Pertama, Sekolah Menengah Akhir, Lansia, dan Kelompok Penerima Program Keluarga Harapan. Proses *Clustering* ini menghasilkan tiga *cluster* yaitu *cluster*_0 sebanyak 156 orang, *cluster*_1 sebanyak 82 orang, dan *cluster*_2 sebanyak 233 orang. Selanjutnya dikembangkan menjadi sebuah sistem dengan metode pengembangan sistem Rapid Application Development (RAD). Sehingga menghasilkan sistem algoritma K-Means untuk mengklasifikasikan penduduk miskin di Kota Pagar Alam. Metode pengujian sistem menggunakan *black box testing* dengan metode alpha dan didapatkan hasil pengujian *database* dengan nilai 4, antarmuka dengan nilai 4, fungsionalitas 4,42, dan algoritma dengan nilai 4. Pada proses pengujian dengan UAT aspek sistem mendapatkan 87% pengguna menyatakan setuju, aspek pengguna bernilai 86% setuju, dan pada aspek interaksi sebesar 87% pengguna setuju. Sehingga dapat disimpulkan jika sistem ini layak digunakan.

Kata Kunci: *Data Mining*, Kemiskinan, K-Means, Klasterisasi, *Black Box*

1. PENDAHULUAN

Berdasarkan Undang-Undang No. 24 Tahun 2004, kemiskinan adalah kondisi sosial ekonomi seseorang atau sekelompok orang yang tidak terpenuhinya hak-hak dasarnya untuk mempertahankan dan mengembangkan kehidupan yang bermartabat. Kebutuhan dasar yang menjadi hak seseorang atau sekelompok orang meliputi kebutuhan pangan, kesehatan, pendidikan, pekerjaan, perumahan, air bersih, pertanahan, sumber daya alam, lingkungan hidup, rasa aman dari perlakuan atau ancaman tindak kekerasan, dan hak untuk berpartisipasi dalam penyelenggaraan kehidupan sosial dan politik.

Berdasarkan hasil jumlah dan persentase penduduk miskin di Kota Pagar Alam selama periode tahun 2018-2019. Jumlah penduduk miskin tahun 2018-2019 tidak mengalami banyak perubahan karena persentase penduduk miskin di Pagar Alam masih berada angka 8,77% dan 8,90%. Namun, jumlah penduduk miskin justru mengalami peningkatan pada tahun 2020 berjumlah 12,71 ribu jiwa dengan persentase penduduk miskin 9,07% pada Kota Pagar Alam hal ini menunjukan bahwa masalah kemiskinan masih menjadi masalah yang cukup serius di kota Pagar Alam terutama pada Kecamatan Dempo Selatan. Dari 455 jumlah penerima (keluarga penerima manfaat) bahwa yang tergolong miskin adalah sebanyak 4,55%, Kecamatan Dempo Tengah dari 471 jumlah penerima bahwa yang tergolong miskin adalah sebanyak 4,71%, Kecamatan Dempo Utara dari 450 jumlah penerima bahwa yang tergolong miskin adalah sebanyak 4,5%, Kecamatan Pagar Alam Selatan dari 441 jumlah penerima bahwa yang tergolong miskin adalah sebanyak 4,41%, dan Kecamatan Pagar Alam Utara dari 455 jumlah penerima bahwa yang tergolong miskin adalah sebanyak 4,55%.

Faktor-faktor pembagian kelompok penerima bantuan PKH dilihat dari keluarga miskin atau pra sejahtera, memiliki anggota keluarga dengan kriteria ibu hamil/menyusui, memiliki anak berusia 0 sampai dengan 6 tahun, memiliki anak dengan kategori pendidikan SD, SMP, atau SMA sederajat, memiliki keluarga lanjut usia minimal 60 tahun, dan penyandang distabilitas yang diutamakan peyandang distabilitas berat. Akan tetapi, dalam proses penyaluran bantuan keluarga miskin melalui Program Keluarga Harapan (PKH) ini masih menemui banyak permasalahan dalam penyalurannya. Di antaranya ialah pendataan dan penyaluran penerima bantuan PKH masih belum tepat, pemberian bantuan hanya memperhatikan kriteria kemiskinan secara umum, sehingga masih banyak masyarakat miskin yang merasa lebih pantas mendapatkan bantuan PKH.

Berdasarkan latar belakang tersebut dilakukanlah upaya penyelesaian yang salah satunya menggunakan proses *Data Mining. Data mining* merupakan proses analisa data untuk menemukan suatu pola dari kumpulan data*.* Tujuan dari *clustering* adalah mengelompokan item data ke dalam sejumlah kecil grup sedemikian rupa sehingga masing-masing grup mempunyai sesuatu persamaan yang esensial (Bahauddin et al., 2021). *Data mining* dapat mengatasi masalah tersebut dengan proses pengklasteran untuk mengetahui yang mana termasuk *cluster* miskin tinggi, *cluster* miskin rendah, dan *cluster* miskin sedang, sehingga pemerintah dapat mengetahui tingkat kemiskinan masyarakat dan dapat memberikan bantuan PKH secara tepat.

Berdasarkan penelitian terdahulu tentang klasterisasi kemiskinan penduduk di provinsi oleh Nasution et al. (2020) menghasilkan pengelompokkan kemiskinan sebanyak 8 provinsi dengan *cluster* tinggi dan 26 provinsi *cluster* rendah, hasil *clustering* selanjutnya dapat digunakan oleh pemerintah untuk memberikan perhatian lebih pada provinsi yang masih memiliki *cluster* kemiskinan tinggi. Metode yang pernah digunakan dalam penelitian sebelumnya yaitu metode algoritma K-Means sebagai metode *clustering* kemiskinan dan hasil evaluasi dapat divisualisasikan dalam bentuk peta sehingga mempermudah dalam melihat sebaran penduduk miskin (Astuti, 2017). Penelitian lainnya menggunakan metode K-Means *clustering* dilakukan oleh Paramitha et al. (2020) di mana teknik K-Means yaitu suatu metode penganalisaan data yang melakukan proses pemodelan tanpa supervis (*unsupervised*) dan merupakan salah satu metode yang melakukan pengelompokan data dengan sistem partisi. Metode *clustering* K-Means digunakan agar penelitian prioritas penduduk tidak mampu bisa lebih berkualitas dan efektif.

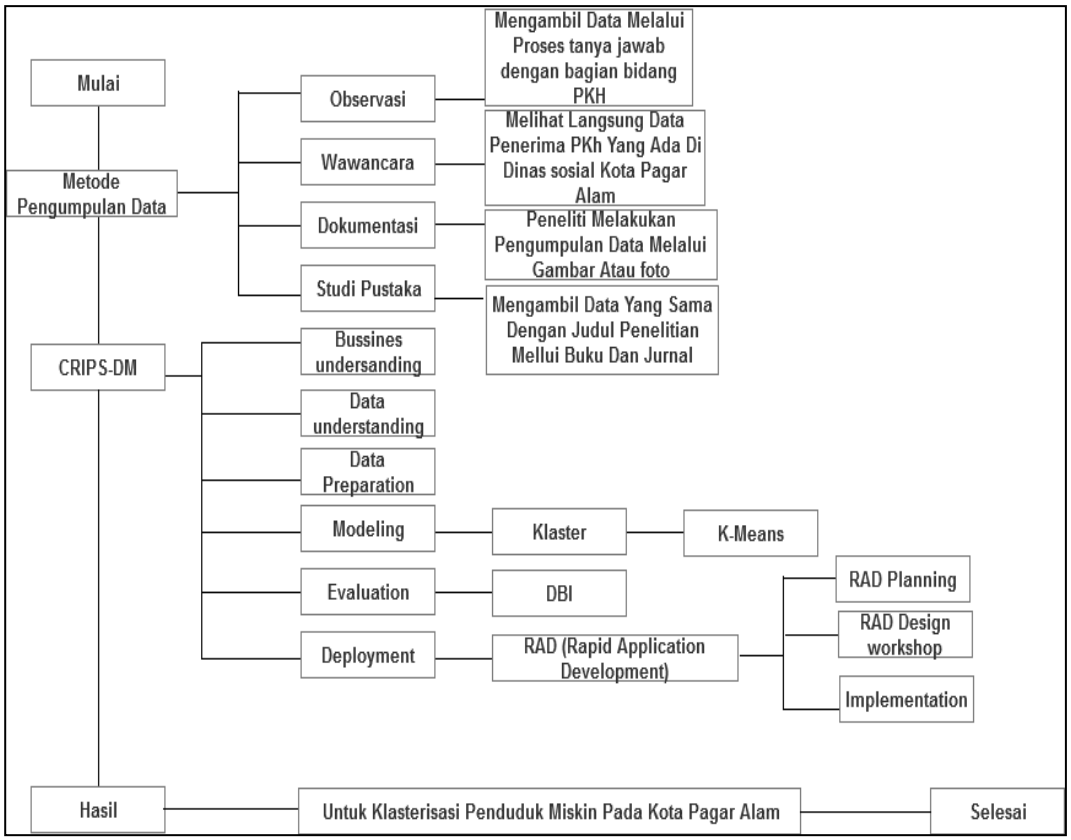
Beberapa penelitian lainnya mengenai penerapan *data mining* tehadap penanganan kemiskinan juga telah dilakukan oleh Jainuddin et al. (2018), Parjito & Permata (2021), dan Sudibyo et al. (2020). Sedangkan Fatmawati & Windarto (2018), Marzuki (2015), dan Sucipto (2019) melakukan penelitian mengenai implementasi algoritma K-Means.

Dalam penelitian ini akan mengelompokkan data kemiskinan kota Pagar Alam diharapkan dapat memberikan masukkan kepada pemerintah agar dapat menjadikan hasil *clustering* sebagai bahan pertimbangan dalam penentuan penerima PKH secara tepat. Dalam penelitian sebelumnya hasil dari *cluster* dapat dijadikan masukan bagi pemerintah agar provinsi yang masuk ke dalam *cluster* tinggi mendapat perhatian lebih (Rofiqo et al., 2018) selain itu juga pengklasteran untuk membantu Dinas Sosial dalam pengelompokan keluarga miskin sehingga bantuan dapat tersalurkan dengan tepat (Paramitha et al., 2020). Pengujian pada penelitian ini menggunakan *black box testing* dengan metode alpha, yaitu memastikan aplikasi dapat berjalan dengan lancar tanpa ganguan (Masripah & Ramayanti, 2020).

2. METODE PENELITIAN

2.1 Tahapan Penelitian

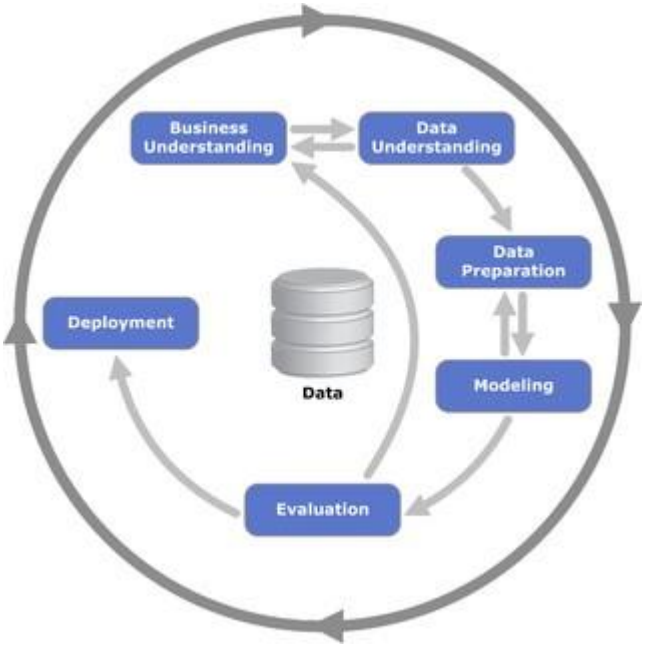
Pelaksanaan penelitian dilakukan dengan beberapa tahapan, dimulai dari proses pengumpulan data dengan metode observasi, wawancara, dokumentasi, dan studi pustaka. Berdasarkan data yang telah ada maka dilanjutkan proses klasterisasi dengan metode *CRISP-DM* dengan algoritma K-Means. Di mana pada tahapan ini terdapat proses *business understanding, data understanding, data preparation, modeling, evaluation,* dan *deployment.* Hasil dari proses klasterisasi selanjutnya akan diimplementasikan ke dalam sistem berbasis *website* yang dapat mengklaster penduduk miskin di Kota Pagar Alam. Adapun tahapan penelitian secara lengkap dapat dilihat pada Gambar 1.



Gambar 1 Tahapan Penelitian

2.2 *Cross Industry Standard Process* untuk *Data Mining* (CRISP-DM)

Cross industry standard process untuk *data mining* atau CRISP-DM dikembangkan tahun 1996 oleh analisis dari beberapa industri seperti Daimler Chrysler, SPSS, dan NCR. CRISP-DM merupakan standarisasi proses *data mining* sebagai strategi pemecahan masalah secara umum dari bisnis atau unit penelitian. Dalam CRISP-DM sebuah proyek *data mining* memiliki siklus hidup yang terbagi dan enam fase (Feblian, 2021). CRISP-DM bukan merupakan satu-satunya standar dalam *data mining* namun merupakan yang terpopuler saat ini. CRISP-DM merupakan metode yang menggunakan model proses pengembangan data yang banyak digunakan para ahli untuk memecahkan masalah terbukti 3 sampai 4 kali lebih banyak digunakan dibanding dengan standar lain yang digunakan. Gambar 2 merupakan gambaran secara umum mengenai siklus hidup dalam CRISP-DM.



Gambar 2 Siklus Hidup dalam CRISP-DM (Astuti, 2017)

2.3 Algoritma K-Means

K-Means merupakan salah satu algoritma *cluster*ing yang digunakan untuk mempartisi data ke dalam beberapa *cluster*. Di mana data yang memiliki tingat kemiripan yang tinggi dikelompokkan dalam satu *cluster* sedangkan data yang memiliki karakteristik yang berbeda dikelompokkan ke dalam *cluster* yang berbeda (Butarbutar et al., 2017; Rahayu et al., 2019).

Dalam penelitian ini peneliti menggunakan data *sample* sebanyak 471 data keluarga dengan jumlah dalam keluarga dengan tipe *integer* berjumlah 8.306 orang. Cara pengelompokkan menggunakan K-Means yaitu:

1) Menentukan banyaknya *cluster* yang dibentuk ada 3 *cluster* (k=3) hal ini didasarkan pada perhitungan *Euclidiean Distance* yang telah dilakukan di mana ketika perhitungan nilai *Euclidiean Distance* tidak lagi berubah pada bentukan 3 *cluster*. Penentuan *cluster* harus lebih kecil dari pada banyaknya data (k<n).
2) Menentukan nilai secara manual atau random untuk pusat *cluster* awal sebanyak *cluster* yang ditentukan.
3) Untuk menghitung jarak data dengan *centroid* menggunakan rumus *Euclidiean Distance.* Persamaan *Euclidian Distance* ditunjukkan pada Pers. (1).

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \qquad (1)$$

Di mana $d(x,y)$ adalah jarak antara data pusat x ke data pusat y, dengan x_i sebagai *data testing* ke i dan y_i sebagai *data training* ke i.

4) Mengecek setiap data berdasarkan kedekatannya dengan jarak terkecil.
5) *Centroid* baru dihitung dengan menghitung nilai rata-rata data pada setiap *cluster*.
6) Melakukan perulangan. Jika perhitugan iterasi baru berbeda dengan iterasi sebelumnya, maka proses dilanjutkan ke langkah perulangan selanjutnya. Namun jika iterasi yang baru dihitung sama dengan iterasi sebelumnya, maka proses *clustering* selesai. Dengan demikian, nilai pusat *cluster* (μj) pada iterasi terakhir akan digunakan sebagai parameter untuk menentukan klasifikasi data.

2.4 Klasterisasi

Klasterisasi adalah salah satu teknik yang digunakan dalam *data mining*. Pengertian klasterisasi dalam *data mining* menurut Sucipto (2019) adalah suatu teknik untuk mengelompokan data ke dalam suatu klaster tertentu yang memungkinkan data dalam klaster tersebut memiliki kesamaan dan memiliki perbedaan yang jelas dengan data pada klaster lainnya. Sedangkan Marzuki (2015) berpendapat klasterisasi adalah suatu kumpulan objek atau data yang memiliki kesamaan di antara mereka dan data yang tidak memiliki kesamaan dimasukkan ke dalam klaster lain, sedangkan klasterisasi proses pengelompokan objek atau data ke dalam grup yang anggotanya memiliki kesamaan tertentu.

Maka dapat disimpulkan bahwa klasterisasi adalah metode pengelompokan data ke dalam suatu kesamaan tertentu. Hal ini berbeda dengan klasifikasi yaitu proses pengelompokkan data baru berdasarkan kelompok atau klasifikasi yang sudah ada, *clustering* akan mengelompokan data baru berdasarkan atribut dengan karakteristik yang sama yang dalam hal ini lebih cocok digunakan pada data penelitian ini.

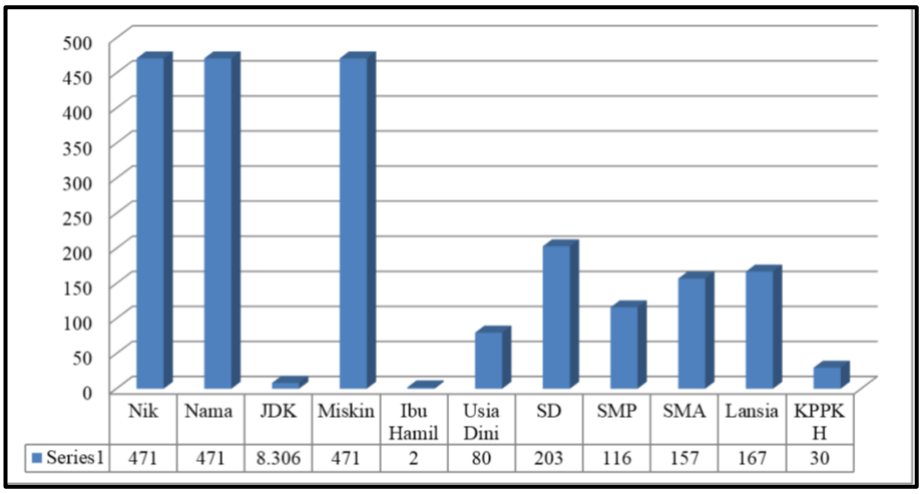
3. HASIL DAN PEMBAHASAN

3.1 Pemahaman Bisnis (*Business Understanding*)

Pada pemahaman bisnis dilakukan tahapan menentukan tujuan penelitian dan ruang lingkup penelitian. Di mana pada tujuan penelitian ini adalah untuk menghasilkan *cluster* yang tinggi dengan menggunakan data PKH, sehingga nantinya model yang menghasilkan nilai *cluster* yang tinggi dapat digunakan untuk melakukan klasterisasi pada penerima PKH. Adapun ruang lingkup dalam penelitian ini ialah menggunakan metode *clustering* dengan algoritma K-Means.

3.2 Pemahaman Data (*Data Understanding*)

Pada fase pemahaman data ini, data didapat dari Dinas Sosial Kota Pagar Alam. Data yang diambil yaitu data Kecamatan Dempo Tengah pada tahun 2020. Terdapat 471 keluarga dengan jumlah dalam keluarga berumlah 8.306 *record*. Atribut dalam data tersebut yaitu Jdk, Miskin, Ibu Hamil, Usia Dini, SD, SMP, SMA, Lansia dan KPPKH, Nik, Nama, dan Jumlah dalam Keluarga. Kategori data yang diterima dalam bentuk *excel* dan tidak terdapat *missing data* di dalamnya. Dalam penenlitian ini atribut yang digunakan dapat dilihat pada Gambar 3.



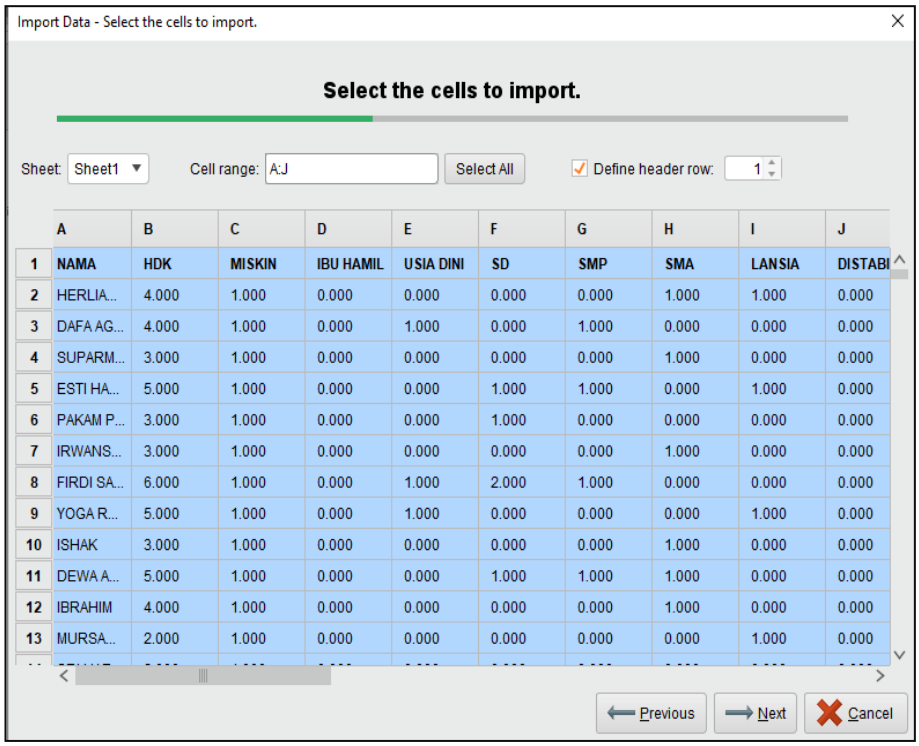
Gambar 3 Grafik Atribut

3.3 Pengolahan Data (*Data Preparation*)

Tahapan pengolahan data langsung diimplementasikan pada RapidMiner dengan tiga tahapan sebagai berikut.

3.3.1 *Data Selection*

Pada tahapan ini ada sepuluh atribut yang didapat dari Dinas Sosial Kota Pagar Alam dengan 471 data. Namun jika ingin menyeleksi data bisa menggunakan *cell range* yang telah disediakan pada *software* RapidMiner. Semua atribut pada data penelitian ini dapat digunakan sehingga peneliti tidak melakukan *selection data* seperti yang ditunjukkan pada Gambar 4.

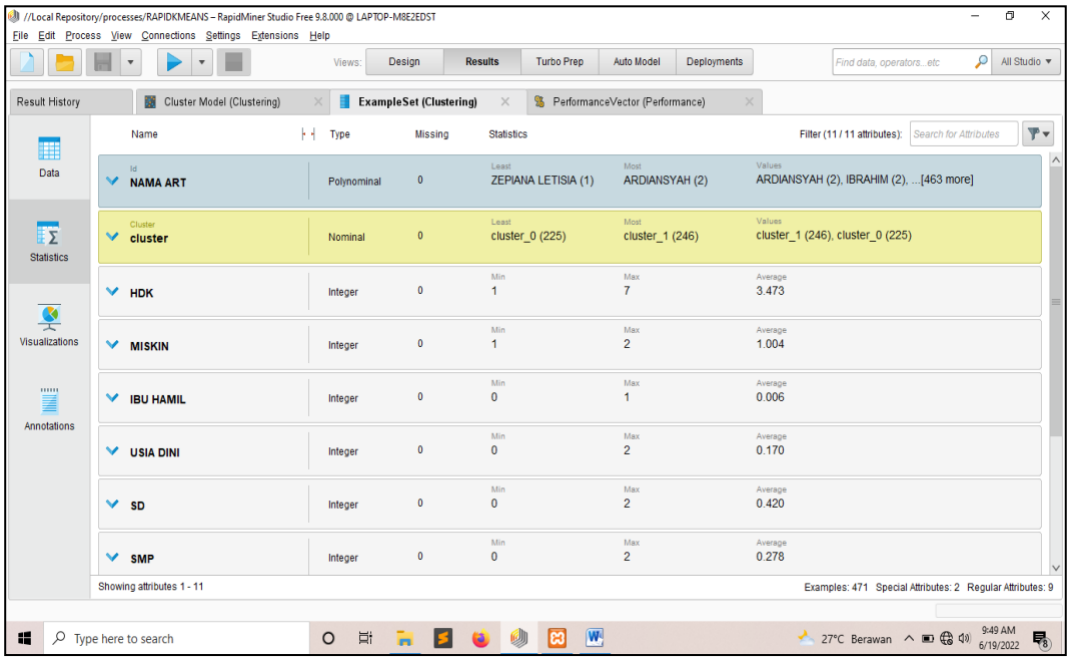


	A	B	C	D	E	F	G	H	I	J
1	NAMA	HDK	MISKIN	IBU HAMIL	USIA DINI	SD	SMP	SMA	LANSIA	DISTABI
2	HERLIA...	4.000	1.000	0.000	0.000	0.000	0.000	1.000	1.000	0.000
3	DAFA AG...	4.000	1.000	0.000	1.000	0.000	1.000	0.000	0.000	0.000
4	SUPARM...	3.000	1.000	0.000	0.000	0.000	0.000	1.000	0.000	0.000
5	ESTI HA...	5.000	1.000	0.000	0.000	1.000	1.000	0.000	1.000	0.000
6	PAKAM P...	3.000	1.000	0.000	0.000	1.000	0.000	0.000	0.000	0.000
7	IRWANS...	3.000	1.000	0.000	0.000	0.000	0.000	1.000	0.000	0.000
8	FIRDI SA...	6.000	1.000	0.000	1.000	2.000	1.000	0.000	0.000	0.000
9	YOGA R...	5.000	1.000	0.000	1.000	0.000	0.000	0.000	1.000	0.000
10	ISHAK	3.000	1.000	0.000	0.000	0.000	0.000	1.000	0.000	0.000
11	DEWA A...	5.000	1.000	0.000	0.000	1.000	1.000	1.000	0.000	0.000
12	IBRAHIM	4.000	1.000	0.000	0.000	0.000	0.000	1.000	0.000	0.000
13	MURSA...	2.000	1.000	0.000	0.000	0.000	0.000	0.000	1.000	0.000

Gambar 4 *Data Selection*

3.3.2 *Data Processing*

Pada tahap *data processing* memastikan bahwa tidak ada lagi *missing value* atau data yang kosong. Seperti yang terlihat pada Gambar 5 seluruh atribut yang digunakan tidak terdapat *missing value*.



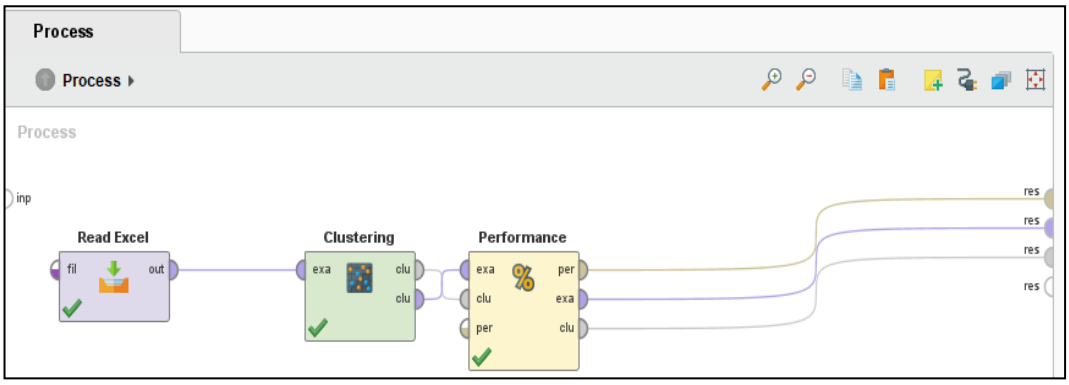
Gambar 5 *Data Processing*

3.3.3 *Data Transformation*

Pada tahap data *transformation* data yang sudah diproses akan menampilkan atribut yang dipilih dan disatukan dalam RapidMiner. Atribut yang pertama yaitu nama penerima dengan tipe *polinomonal* berjumlah 471 nama penerima. Atribut yang kedua yaitu jumlah dalam keluarga dengan tipe *integer* dengan jumlah 8.306 orang. Atribut yang ketiga yaitu miskin dengan tipe *integer* dengan jumlah 471 orang. Atribut yang keempat yaitu ibu hamil dengan tipe *integer* dengan jumlah 2 orang. Atribut yang kelima yaitu usia dini dengan tipe *integer* berjumlah 80 orang. Atribut yang keenam yaitu SD dengan tipe *integer* dengan jumlah 203 orang. Atribut yang ketujuh yaitu SMP dengan tipe *integer* dengan jumlah 116 orang. Atribut yang kedelapan yaitu SMA dengan tipe *integer* dengan jumlah 375 orang. Atribut yang kesembilan yaitu lansia dengan tipe *integer* dengan jumlah 167 orang. Atribut yang kesepuluh yaitu disabilitas dengan tipe *integer* dengan jumlah 30 orang.

3.4 Pemodelan (*Modeling*)

Tahapan pemodelan dilakukan menggunakan teknik klasterisasi dengan algoritma yang digunakan yaitu K-means. *Clustering* operator ini mengambil objek dari *port input* dan mengirimkan salinannya ke *port output*. Setiap *port* yang terhubung membuat salinan yang independen (tidak terikat). Jadi, ketika mengubah suatu salinan tidak berpengaruh pada salinan yang lainnya sehingga dapat dihubungkan dengan *performance* yaitu untuk mengetahui suatu model algoritma K-means. Hasil pemodelan yaitu berupa pola informasi yang dapat memudahkan pihak yang berkepentingan seperti yang terlihat pada Gambar 6.



Gambar 6 Model Algoritma K-Means

1) Tentukan jumlah *cluster*

Untuk menentukan jumlah *cluster* dilakukan percobaan jumlah *cluster* 3. Percobaan 3 *cluster* itu ada C0, C1 dan C2 dengan atribut Jdk, Miskin, Ibu hamil, Usia dini, SD, SMP,SMA, Lansia, dan KPPKH. Dengan pengukuran *performance vector* rata-rata dalam *centroid distance* yang didapat dengan nilai 0,154 kemudian rata-rata *cluster*_0 bernilai 0,187, rata-rata dalam *cluster*_1 bernilai 0,129, dan rata-rata dalam *centroid cluster*_2 bernilai 0,141 dengan nilai Davies Bouldin Index 0,160 seperti yang ditunjukkan pada Gambar 7.

Attribute	cluster_0	cluster_1	cluster_2
HDK	4.474	1.817	3.386
MISKIN	1.013	1	1
IBU HAMIL	0	0	0.013
USIA DINI	0.346	0	0.112
SD	1.051	0.037	0.133
SMP	0.327	0.073	0.318
SMA	0.308	0.061	0.425
LANSIA	0.154	1	0.236
DISTABILITAS	0.051	0.073	0.069

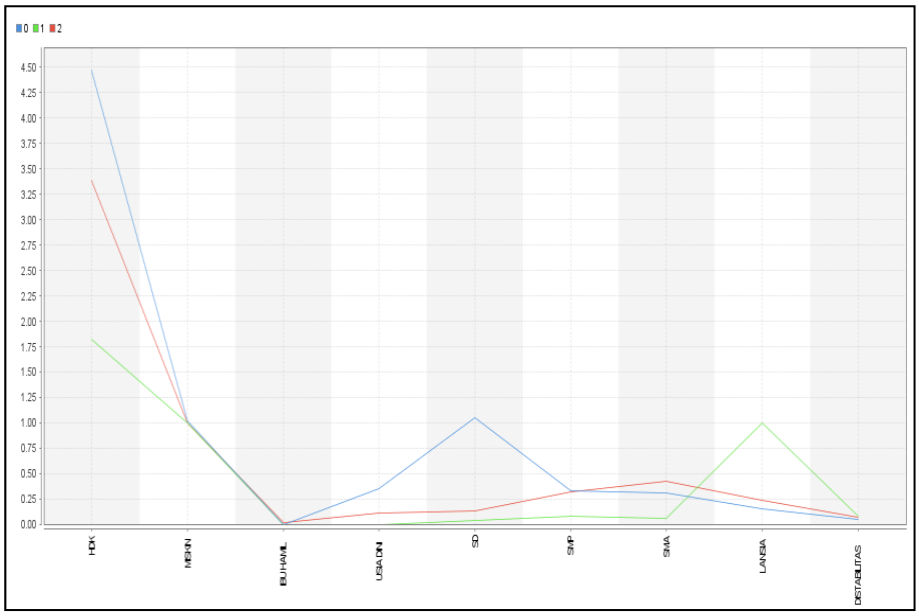
Gambar 7 Percobaan 3 *Cluster*

2) Menentukan titik *centroid* secara acak

Titik centeroid secara acak ditentukan berdasarkan beberapa percobaan *cluster* di RapidMiner. Dari 3 percobaan tersebut dipilih titik *centeroid* terkecil untuk mendapatkan hasil terbaik.

3) Hitung jarak data ke *centroid*

Setelah memasukkan *dataset* dan algoritma K-Means pada RapidMiner maka proses data dengan klik run untuk melihat jarak data ke *centeroid*. Grafik jarak data ke *centroid* dapat dilihat pada Gambar 8.



Gambar 8 Jarak Data ke *Centroid*

4) Perbarui nilai titik *centroid*

Setelah dilakukan beberapa percobaan, titik *centroid* pada 3 *cluster* tidak mengalami perubahan. Apabila *centeroid* berubah maka dilakukan perulangan iterasi namun jika tidak berubah maka alhasil pengulangan dihentikan dan telah didapatkan masing-masing kelompok.

Berdasarkan perhitungan RapidMiner maka diperoleh pola yang nantinya akan diimplementasikan pada sistem. Pola yang digunakan untuk mengelompokkan/*cluster* data berdasarkan hasil perhitungan jarak adalah:

a) Jika C0<C1 dan C0<C2 maka *cluster* 0 dengan keterangan sedang.
b) Jika C1<C0 dan C1<C2 maka *cluster* 1 dengan keterangan rendah.
c) Jika C2<C0 dan C2<C1 maka *cluster* 2 dengan keterangan tinggi.

Maka dari pola yang didapatkan dari RapidMiner yang digunakan pada sistem dengan metode *clustering* K-Means diperoleh bahwa *cluster_0* memiliki tingkat kesejahteraan sedang dengan jumlah 156, *cluster_1* memiliki tingkat kesejahteraan rendah dengan jumlah 82, dan *cluster*_2 memiliki tingkat kesejahteraan tinggi dengan jumlah 233. Setelah dilakukan proses *clustering* maka dapat diketahui *cluster* penerima berdasarkan tingkat kesejahteraannya dan yang lebih direkomendasikan untuk mendapatkan PKH adalah dengan tingkat kesejahteraan rendah yang berada pada *cluster_1.*

3.5 *Evaluation*

Pengujian dilakukan dengan menggunakan Davies Bouldin Index (DBI) terhadap 3 *cluster* dengan keseluruhan data 471 *record*. *Performance vector* rata-rata dalam *centroid* yang didapat bernilai 0,154, kemudian rata-rata *centroid*_0 bernilai 0,187, rata-rata dalam *cluster*_1 bernilai 0,129, dan rata-rata dalam *centroid cluster*_2 berniali 0,141, kemudian nilai pengukuran jarak antar titik dalam *cluster* didapat nilai DBI 0,160 seperti yang ditunjukkan pada Gambar 9.

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023: 66 – 77

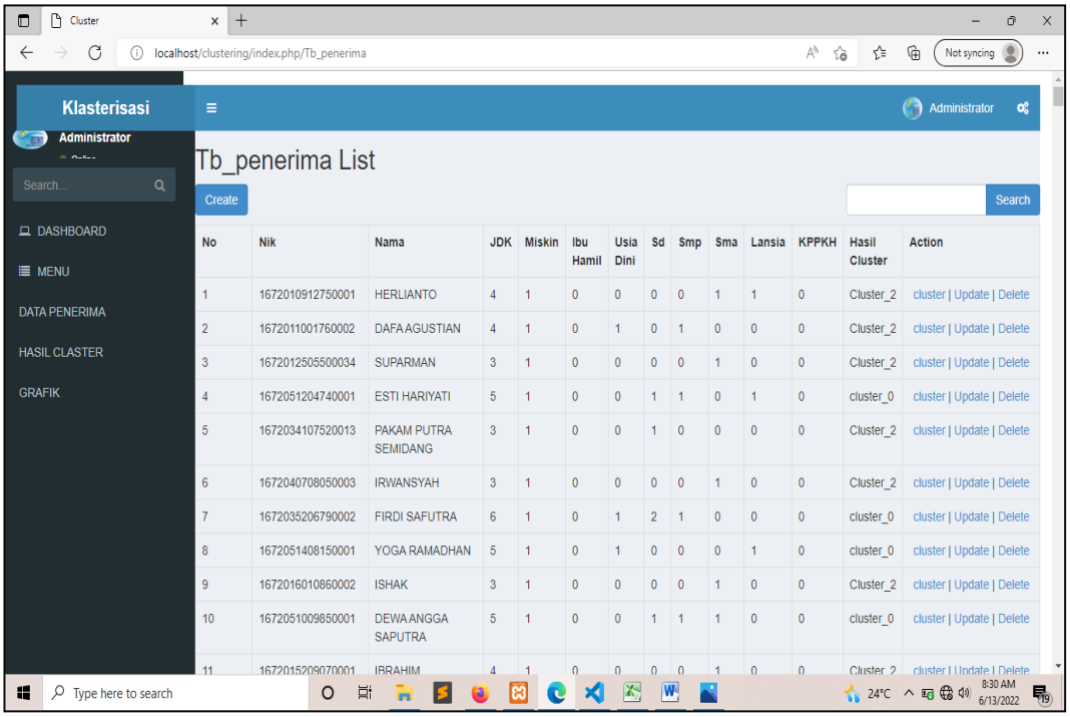
PerformanceVector

```
PerformanceVector:
Avg. within centroid distance: 0.154
Avg. within centroid distance_cluster_0: 0.187
Avg. within centroid distance_cluster_1: 0.129
Avg. within centroid distance_cluster_2: 0.141
Davies Bouldin: 0.160
```

Gambar 9 Nilai DBI

3.6 *Deployment*

Pada tahapan *deployment* ini ialah tahapan terakhir yaitu menerapkan model algoritma *clustering* yang dihasilkan pada bahasa pemograman *framework php* dengan metode pengembangan *Rapid Application Development* (RAD) untuk dapat mengelompokkan data penerima PKH. Gambar 10 merupakan halaman Data Penerima yang terdapat informasi berupa data penerima PKH.



Gambar 10 Halaman Data Penerima Admin

4. KESIMPULAN

Berdasarkan penelitian yang dilakukan menghasilkan klasterisasi penduduk miskin pada Kota Pagar Alam. Hasil dari proses *clustering* data dalam menentukan penerima PKH melalui aplikasi RapidMiner sama dengan hasil yang diterapkan di sistem yang telah dibangun dengan jumlah *cluster* yang terdiri dari tiga *cluster*. *Cluster* dimulai dari *cluster*_0, *cluster*_1, dan *cluster*_2. Data yang berada pada *cluster*_0 berjumlah 156 data, *cluster*_1 berjumlah 82 data, dan *cluster*_2 berjumlah 233 data. *Performance vector* rata-rata dalam *centroid* yang didapat bernilai 0,154, kemudian rata-rata *centroid*_0 bernilai 0,187, rata-rata dalam *cluster*_1 bernilai 0,129, dan rata-rata dalam *centroid cluster*_2 berniali 0,141, kemudian nilai pengukuran jarak antar titik dalam *cluster* didapat nilai DBI 0,160. Hasil pengujian *black box testing* berupa pengujian *alpha* menggunakan kuesioner yang diisi oleh pakar menghasilkan nilai pengujian *database* sebesar 4,

pengujian antarmuka bernilai 4, pengujian fungsionalitas bernilai 4,42, dan untuk algoritma bernilai 4. Dengan demikian, *alpha* memperoleh nilai kelayakan rata-rata nilai 4 sehingga menyimpulkan bahwa sistem layak digunakan. Pada proses pengujian dengan UAT aspek sistem 87% pengguna menyatakan setuju, aspek pengguna sebanyak 86% setuju, dan pada aspek interaksi sebesar 87% pengguna setuju. Berdasarkan hasil pengujian yang telah dilakukan tersebut maka sistem layak digunakan.

Penelitian ini menghasilkan *cluster* pada data penduduk miskin penerima bantuan PKH, dengan adanya hasil *cluster* ini pemerintah dapat lebih bijak dalam menentukan penerima bantuan ataupun membuat kebijakan. Sistem *clustering* yang dibuat akan mengelompokan data peneriman bantuan secara tepat dengan algoritma K-Means yang teruji berdasarkan data sebelumnya.

DAFTAR PUSTAKA

Astuti, F. D. (2017). Penerapan Data Mining Untuk Clustering Data Penduduk Miskin Menggunakan Algoritma Hard C-Means. *Data Manajemen Dan Teknologi Informasi (DASI)*, *18*(1), 64–69. https://ojs.amikom.ac.id/index.php/dasi/article/view/1836

Bahauddin, A., Fatmawati, A., & Sari, F. P. (2021). Analisis Clustering Provinsi di Indonesia Berdasarkan Tingkat Kemiskinan Menggunakan Algoritma K-Means. *Jurnal Manajemen Informatika Dan Sistem Informasi*, *4*(1), 1–8. https://doi.org/10.36595/misi.v4i1.216

Butarbutar, N., Windarto, A. P., Hartama, D., & Solikhun, S. (2017). Komparasi Kinerja Algoritma Fuzzy C-Means dan K-Means dalam Pengelompokan Data Siswa Berdasarkan Prestasi Nilai Akademik Siswa. *Jurasik (Jurnal Riset Sistem Informasi Dan Teknik Informatika)*, *1*(1), 46. https://doi.org/10.30645/jurasik.v1i1.8

Fatmawati, K., & Windarto, A. P. (2018). Data Mining: Penerapan Rapidminer dengan K-Means Cluster pada Daerah Terjangkit Demam Berdarah Dengue (DBD) Berdasarkan Provinsi. *Computer Engineering, Science and System Journal*, *3*(2), 173. https://doi.org/10.24114/cess.v3i2.9661

Feblian, D. (2021). Implementasi Model CRISP-DM untuk Menentukan Sales Pipeline pada PT. X [Universitas Trisakti]. In *THESIS-2016*. http://repository.trisakti.ac.id/usaktiana/index.php/home/detail/detail_koleksi/0/THE/judul/0000000000000105201/

Jainuddin, J., Agus, F., & Astuti, I. F. (2018). Sistem Informasi Data Kriteria Rakyat Miskin Desa Liang Ilir Kecamatan Kota Bangun. *Informatika Mulawarman : Jurnal Ilmiah Ilmu Komputer*, *13*(1), 39. https://doi.org/10.30872/jim.v13i1.1004

Marzuki, I. (2015). Temu Kembali Informasi Big Data Menggunakan K-Means Clustering. *SMATIKA JURNAL : STIKI Informatika Jurnal*, *5*(02), 01–07. https://doi.org/10.32664/SMATIKA.V5I02.75

Masripah, S., & Ramayanti, L. (2020). Penerapan Pengujian Alpha dan Beta pada Aplikasi Penerimaan Siswa Baru. *Swabumi (Suara Wawasan Sukabumi) : Ilmu Komputer, Manajemen, Dan Sosial*, *8*(1), 100–105. https://doi.org/10.31294/SWABUMI.V8I1.7448

Nasution, I., Windarto, A. P., & Fauzan, M. (2020). Penerapan Algoritma K-Means Dalam Pengelompokan Data Penduduk Miskin Menurut Provinsi. *Building of Informatics, Technology and Science (BITS)*, *2*(2), 76–83. https://doi.org/10.47065/bits.v2i2.492

Paramitha, I. A. S. D., Sasmita, G. M. A., & Raharja, I. M. S. (2020). Analisis Data Log IDS Snort dengan Algoritma Clustering Fuzzy C-Means. *Majalah Ilmiah Teknologi Elektro*, *19*(1), 95. https://doi.org/10.24843/MITE.2020.v19i01.P14

Parjito, P., & Permata, P. (2021). Penerapan Data Mining Untuk Clustering Data Penduduk Miskin Menggunakan Metode K-Means. *Ainet : Jurnal Informatika*, *3*(1), 31–37. https://doi.org/10.26618/AINET.V3I1.5878

Rahayu, A. E., Hikmah, K., Yustia, N., & Fauzan, Abd. C. (2019). Penerapan K-Means Clustering Untuk Penentuan Klasterisasi Beasiswa Bidikmisi Mahasiswa. *ILKOMNIKA: Journal of Computer Science and Applied Informatics*, *1*(2), 82–86. https://doi.org/10.28926/ilkomnika.v1i2.23

Rofiqo, N., Windarto, A. P., & Hartama, D. (2018). Penerapan Clustering pada Penduduk yang Mempunyai Keluhan Kesehatan dengan Datamining K-Means. *KOMIK (Konferensi Nasional Teknologi Informasi Dan Komputer)*, *2*(1). https://doi.org/10.30865/komik.v2i1.929

Sucipto, A. (2019). Klasterisasi Calon Mahasiswa Baru Menggunakan Algoritma K-Means. *Science Tech: Jurnal Ilmu Pengetahuan Dan Teknologi*, *5*(2), 50–56. https://doi.org/10.30738/jst.v5i2.5829

Sudibyo, N. A., Iswardani, A., Sari, K., & Suprihatiningsih, S. (2020). Penerapan Data Mining pada Jumlah Penduduk Miskin di Indonesia. *Jurnal Lebesgue : Jurnal Ilmiah Pendidikan Matematika, Matematika Dan Statistika*, *1*(3), 199–207. https://doi.org/10.46306/lb.v1i3.42

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023, Pp. 78 – 89
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Analisa Deteksi dan Pengenalan Wajah pada Citra dengan Permasalahan Visual

Verry Noval Kristanto [(1)*], Imam Riadi [(2)], Yudi Prayudi [(3)]
[1,3] Magister Teknik Informatika, Fakultas Teknik Industri, Universitas Islam Indonesia, Yogyakarta
[2] Sistem Informasi, Fakultas Sains dan Teknologi Terapan, Universitas Ahmad Dahlan, Yogyakarta
e-mail : verry.kristanto@students.uii.ac.id, imam.riadi@is.uad.ac.id, prayudi@uii.ac.id.
* Penulis korespondensi.
Artikel ini diajukan 11 Desember 2022, direvisi 30 Januari 2023, diterima 30 Januari 2023, dan dipublikasikan 30 Januari 2023.

Abstract

Facial recognition is a significant part of criminal investigations because it may be used to identify the offender when the criminal's face is consciously or accidentally recorded on camera or video. However, a majority of these digital photos have poor picture quality, which complicates and lengthens the process of identifying a face image. The purpose of this study is to discover and identify faces in these low-quality digital photographs using the Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) face identification method and the Viola-Jones face recognition method. The success percentage for the labeled face in the wild (LFW) dataset is 63.33%, whereas the success rate for face94 is 46.66%, while LDA is only a maximum of 20% on noise and brightness. One of the names and faces from the dataset is displayed by the facial recognition system. The brightness of the image, where the facial item is located, and any new objects that have entered the scene have an impact on the success rate.

Keywords: Face, Detection, Recognition, Digital Image, Visual Problems

Abstrak

Pengenalan wajah merupakan aspek penting dari investigasi kriminal karena digunakan untuk mengungkap identitas pelaku ketika wajah pelaku secara sengaja atau tidak sengaja terekam kamera atau video. Namun, banyak dari hasil gambar digital ini menunjukkan kualitas gambar yang buruk yang membuat proses pengenalan wajah menjadi lebih sulit dan memakan waktu lebih lama. Fokus dari penelitian ini adalah untuk mendeteksi dan mengenali wajah pada citra digital dengan permasalahan visual tersebut dengan metode deteksi wajah Viola-Jones dan metode pengenlan wajah dengan *Principal Component Analysis* (PCA) dan *Linier Discriminant Analysis (LDA)*. Program pengenalan wajah menampilkan salah satu nama dan wajah yang tersimpan dalam *dataset*, tingkat keberhasilan *dataset* yang digunakan sebesar 63,33% pada *dataset Labeled Face in the Wild* (LFW) pada 46,66% pada face94, sedangkan LDA hanya maksimal 20% pada noise dan kecerahan. Tingkat keberhasilan ditentukan oleh kecerahan citra dan posisi objek wajah pada citra atau objek tambahan yang sebelumnya tidak ada pada citra.

Kata Kunci: Wajah, Deteksi, Pengenalan, Citra Digital, Permasalahan Visual

1. PENDAHULUAN

Pengenalan wajah banyak digunakan untuk semua jenis tindakan keamanan seperti membuka kunci ponsel, memeriksa informasi seseorang di tempat umum, dan dalam penyelidikan seperti melacak orang, menemukan orang hilang, bahkan di bidang lain (Singh & Goel, 2020). Gambar wajah dapat diambil dengan kamera CCTV, *smartphone*, dan perangkat lain yang dapat merekam subjek sebagai sinyal digital. Namun citra digital terkadang memiliki masalah kualitas, ada banyak alasan yang menyebabkan kualitas gambar buruk seperti: kemampuan kamera untuk menangkap cahaya, posisi kamera, cahaya yang kurang atau berlebihan yang menyebabkan hasil citra digital memiliki visual yang kurang baik. Citra yang diperoleh dari hasil *screenshot* atau perekaman secara langsung terkadang menyajikan informasi yang kurang lengkap karena permasalahan kualitas citra itu sendiri (Tang et al., 2020; Xiong, 2020).

Noise (derau) disebabkan oleh banyak faktor seperti kurangnya pencahayaan saat pengambilan gambar, keterbatasan resolusi pixel dari lensa kamera yang digunakan, keterbatasan kamera dalam menangkap gambar bergerak, inferensi gelombang elektromagnetik, dan sebagainya. Dijelaskan oleh Moradmand et al. (2020) terdapat berbagai macam jenis *noise* di antaranya rician *noise*, *gaussian noise*, *salt-and-pepper nois*e. Namun yang digunakan dalam penelitian ini ada *gaussian noise*. *Blur* (citra kabur) dapat diakibatkan oleh berbagai hal seperti pergerakan selama pengambilan gambar oleh lensa kamera, penggunaan alat optik yang tidak fokus, penggunaan lensa dengan sudut yang lebar, gangguan atmostfer, tingkat pencahayaan yang singkat sehingga mengurangi jumlah foton yang ditangkap oleh lensa kamera. *Brightness* (kecerahan) dapat diperbaiki dengan menaikan atau mengurangi jumlah konstanta pada setiap *pixel* pada citra.

Permasalahan visual dari yang sudah disebutkan membuat proses esktraksi informasi menjadi lebih sulit dan membutuhkan waktu yang lama, oleh karena itu tujuan dari penilitian ini adalah untuk mengenali objek wajah pada citra yang memiliki permasalahan visual terutama *noise*, *blur,* dan permasalahan kecerahan maka diperlukan *preprocessing* citra. Menurut Furht et al. (2018), *image preprocessing* merupakan proses pengurangan *noise*, menyesuaikan tingkat kecerahan dan memperbaiki kualitas data sebelum masuk tahap ekstraksi objek pada citra, di mana teknik *image preprocessing* yang digunakan dalam penelitian ini adalah *noise filtering*, *deblurring*, dan *brightness adjustment*.

Deteksi wajah adalah langkah awal yang harus dilalui dalam pengenalan wajah dengan mengekstrak objek wajah, metode Viola-Jones memiliki kecepatan, dan akurasi yang cukup besar karena menggabungkan beberapa konsep (fitur Haar, Integral Image, Adabost, Cascade Classifier) menjadikan metode yang popular dan sering digunakan dalam penelitian deteksi wajah seseorang pada citra. Kemudian akan digabungkan dengan metode *Principal Component Analysis* (PCA) dan *Linier Discriminant Analysis* (LDA) untuk mendeteksi kecocokan wajah yang telah diekstrak. Metode PCA digunakan untuk mereduksi dimensi dan melakukan pengenalan wajah secara efisien, pada penelitian yang dilakukan Borade et al. (2016) menunjukan bahwa setiap wajah dapat direkrontruksi dengan menggunakan sejumlah kecil *eigenfaces* dan bobot yang sesuai serta mendapatkan tingkat keberhasilan pengenalan wajah sebesar 100%. Penelitian dari Al-Ghrairi et al. (2022) melakukan proses analisis dengan dua tahap yaitu tahap deteksi dan tahap pengenalan. Hasil analisis menunjukan efektifitas sebesar 96% tingkat pengenalan dan efisiensi yang baik dengan rata-rata waktu eksekusi perintah hanya selama 0,32 detik. Penelitian oleh Aggarwal et al. (2021) menunjukan pengenalan wajah menggunakan PCA *eigen faces* menghasilkan hasil yang kurang akurat pada kedua *dataset* karena menghasilkan tingkat kesalahan tertinggi masing-masing sekitar 10,61% pada *dataset* ORL dan 15,01% pada *dataset* yale dibandingkan dengan teknik lain yang diterapkan. Penelitian pengenalan wajah oleh Kosasih (2021) diperoleh tingkat akurasi tertinggi terjadi ketika data latih tiap orang sebanyak 7 dan data uji tiap orang sebanyak 3 dengan tingkat akurasi sebesar 96,67%. Penelitian oleh Anam (2020) dengan pengukuran jarak euclidean akan mendapatkan nilai minimum dan maksimum, sehingga mendapat hasil yaitu wajah yang dikenali dan tidak dikenali, dari hasil percobaan didapatkan hasil bahwa persentase keakuratan identifikasi wajah menggunakan eigenfaces menunjukkan hasil yang memuaskan.

Dari penelitian yang telah disebutkan proses pengenalan wajah tidak dilakukan pada citra digital terutama pada permasalahan *noise*, *blur,* dan kecerahan. Kemudian untuk memastikan terdapat objek wajah pada citra akan dilakukan deteksi wajah menggunakan metode Viola-Jones sampai akhirnya akan dilakukan proses pengenalan wajah dengan PCA dan LDA. Untuk mebuat analisis pengenalan wajah lebih akurat akan digunakan dua *dataset* penelitian yang digunakan adalah *Labeled Face in the Wild* (LFW) *dataset* dan face94 *dataset*, lalu membandingkan tingkat keakuratan dalam mendeteksi kecocokan pada setiap *dataset*.

2. METODE PENELITIAN

Pada dasarnya ada beberapa langkah yang dilakukan dalam pengenalan wajah yaitu *preprocessing* lalu deteksi objek wajah pada citra kemudian ekstraksi dan akhirnya akan dilakukan pengenalan wajah.

2.1 *Dataset*

Dataset yang digunakan adalah *dataset* yang sudah dilakukan uji coba. *Labeled face in the wild* adalah *database* foto wajah yang dirancang untuk mempelajari masalah pengenalan wajah tanpa batasan apapun, di mana *database* berisi lebih dari 13000 citra wajah yang dikumpulkan dari internet. LFW *Dataset* telah banyak digunakan sebagai *dataset* untuk penelitian pengenalan wajah dari berbagai aspek permasalahan seperti pada penelitian ElSayed et al. (2017), Knoche et al. (2021), dan Proenca et al. (2016). *Dataset* kedua yang digunakan adalah face94 yang terdiri lebih dari 3000 citra wajah dan telah digunakan untuk penelitian oleh Barnouti et al. (2018), Jalal et al. (2016), dan Matin et al. (2016). Contoh citra dari kedua *dataset* dapat dilihat pada Gambar 1.

Gambar 1 Sampel Data Citra LFW *Dataset* dan Face94 *Dataset*

2.2 *Preprocessing*

Pemrosesan signal terdiri dari penanganan data untuk mengekstrak informasi dianggap relevan, atau untuk memodifikasinya sehingga memberi properti yang. Menurut Gonzalez & Woods (2018) pengolahan citra digital dibagi menjadi beberapa proses pengolahan citra di antaranya:

1) Peningkatan kualitas (*enhancement*) adalah proses manipulasi citra agar hasil citra jelas.
2) Pemugaran (*restoration*) adalah proses untuk menghilangkan dan meminimalkan kecacatan pada citra.
3) Rekonstruksi (*reconstruction*) adalah jenis operasi untuk membentuk ulang objek dari beberapa bagian pada citra.
4) Segmentasi (*segmentation*) proses memecah suatu citra menjadi beberapa bagian dengan suatu kriteria tertentu.
5) Ekstraksi (*extraction*) merupakan proses pengolahan yang dilakukan setelah segmentasi pada citra.
6) Analisis citra (*image analysis*) adalah jenis operasi yang bertujuan untuk mengkalkulasi besaran kuantitatif dari citra untuk mencari deskripsinya.
7) Pengenalan objek pada citra (*image pattern classification*) adalah proses pemberian label ke sebuah objek jika objek tersebut memiliki kecocokan dengan objek lain di luar citra.

2.3 Deteksi Wajah Viola-Jones

Pada tahun 2001, Viola & Jones (2001) merancang sebuah metode yang cepat dan akurat dalam mendeteksi wajah dan objek lain pada citra digital, metode Viola-Jones memiliki empat fase dalam prosesnya.

1) Karakteristik Haar pada Viola-Jones yang terdiri dari deteksi *edge feature, linear feature, central feature,* dan *diagonal feature* (Lu & Yang, 2019). Nilai yang ada pada bagian hitam dan putih adalah *eigenvalue* yang akan dikalkulasi dengan Pers. (1).

$$v = \sum_{putih} - \sum_{hitam} \quad (1)$$

2) Integral digunakan mempercepat kalkulasi pada fitur haar yang sangat luas dengan membagi citra ke beberapa kelompok sel citra.
3) Adaboost memiliki peran untuk memastikan tingkat kecepatan pada deteksi wajah pada proses integral.
4) Klasifikasi *cascade* yang dilakukan secara bertingkat di mana setiap tingkatan memberikan hasil subcitra yang diyakini bukan wajah.

2.4 Pengenalan Wajah

Meskipun ada teknik lain untuk mengenali wajah, dalam penelitian ini menggunakan 2 metode pengenalan yaitu *Principal Component Analysis* (PCA) dan *Linier Discriminant Analysis* (LDA). Meskipun model pengenalannya sama pada semua teknik ini, persyaratan data pelatihan dan persyaratan perhitungan matriks bervariasi.

2.5 Validasi *Confusion Matrix*

Confusion Matrix adalah tabel dengan 4 kombinasi berbeda dari nilai prediksi dan nilai aktual (Ariza-Lopez et al., 2018). Ada empat istilah yang merupakan representasi hasil proses klasifikasi pada *confusion matrix* yaitu:

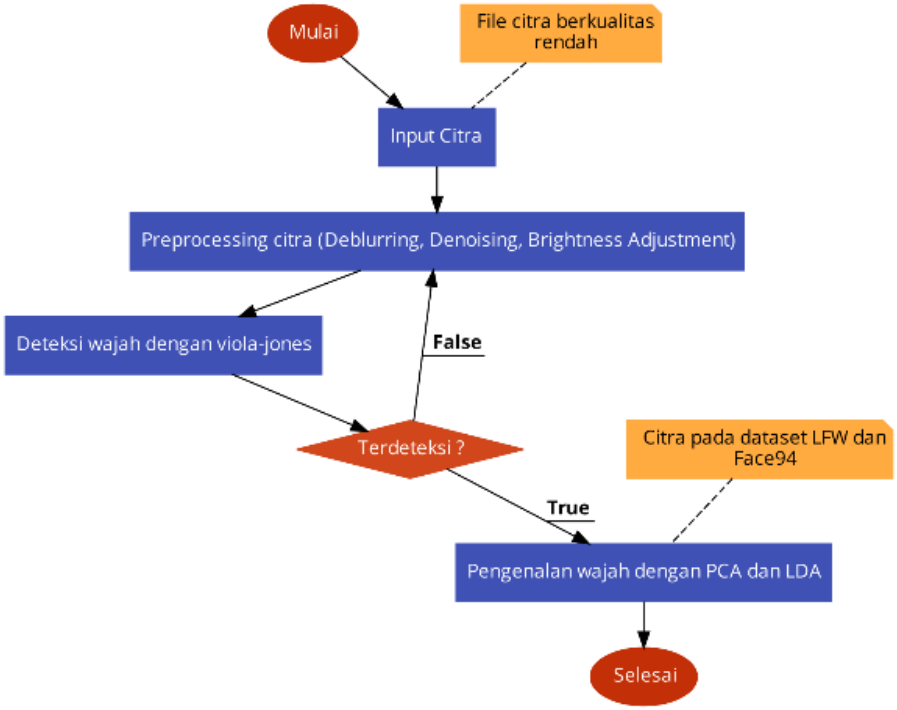
1) *True Positive* (TP) memprediksi positif dan itu benar. Sistem memprediksikan bahwa wajah sama dan wajah tersebut memang benar sama.
2) *True Negative* (TN) memprediksi negatif dan itu benar. Memprediksikan bahwa wajah tidak sama dan memang benar wajah tersebut tidak sama.
3) *False Positive* (FP) memprediksi positif dan itu salah. Sistem memprediksikan bahwa wajah sama tetapi sebenaranya wajah tersebut tidak sama.
4) *False Negative* (FN) memprediksi negatif dan itu salah. Sistem memperkirakan bahwa seorang wajah tidak sama tetapi sebenarnya wajah tersebut sama persis.

Dengan didapatkan 4 klasifikasi tersebut, selanjutnya akan dilakukan penghitungan akurasi, presisi, dan *recall* untuk menentukan tingkat keberhasilan sistem dalam mengenali objek wajah tersebut.

1) *Accuracy* mennggambarkan seberapa akurat model dalam mengklasifikasikan dengan benar.
2) *Precision* menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model.
3) *Recall* atau *sensitivity* menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi.

3. HASIL DAN PEMBAHASAN

Dalam *preprocessing* citra guna meningkatkan kualitas objek yang ada di dalamnya, kami membuat sebuah sistem dalam melakukan *preprocessing* tersebut.



Gambar 2 Diagram Alur Sistem

Dari alur sistem pada Gambar 2 citra dengan permasalahan visual akan melewati tahap *preprocessing* citra dikategorikan berdasarkan permasalahan dari citra tersebut untuk memperbaiki visual citra. Setelah melewati proses tersebut akan melalui tahap deteksi wajah dengan Viola-Jones dan akan kembali ke tahap *preprocessing* jika tidak terdeteksi wajah pada citra tersebut. Jika terdeteksi wajah akan dilakukan pemotongan untuk menghilangkan objek lain selain wajah sampai akhirnya ke tahap akhir di pengenalan wajah menggunakan PCA dan LDA.

3.1 *Preprocessing Deblurring* Citra

Algoritma *blind deconvolution* dapat digunakan secara efektif ketika tidak ada informasi tentang distorsi (kabur dan kebisingan), dengan menggunakan teknik *deblurring* seperti berikut dan contohnya ditunjukkan pada Gambar 3.

1) *Deblurring* PSF berukuran kecil menggunakan *array* berukuran kecil, UNDERPSF sebagai perkiraan awal PSF. Ukuran array UNDERPSF adalah 4 piksel lebih pendek di setiap dimensi daripada PSF yang sebenarnya.
2) *Deblurring* PSF berukuran besar menggunakan *array* lebih besar dari UNDERPSF, OVERPSF untuk PSF awal yang 4 piksel lebih panjang di setiap dimensi dari PSF yang sebenarnya.
3) INITPSF untuk PSF awal yang persis dengan ukuran yang sama dengan PSF yang sebenarnya.
4) Dering dalam gambar yang dipulihkan INITPSF, terjadi di sepanjang area kontras intensitas tajam dan di sepanjang batas gambar. Dengan cara mengurangi efek dering dengan menentukan fungsi pembobotan. Algoritma menimbang setiap piksel sesuai dengan *array* bobot dengan memulihkan gambar dan PSF.
5) *Deconvblind* dengan *array* bobot dan peningkatan jumlah iterasi sebesar 30.
6) *Deconvblind* dengan batasan tambahan pada PSF. Fungsi FUN mengembalikan *array* PSF yang dimodifikasi yang digunakan *deconvblind* untuk iterasi berikutnya.

Blur Underpsf Overpsf Initpsf Deconvblind Deconvblind2

Gambar 3 *Preprocessing Deblurring* Citra

Dari hasil penggunaan metode *blind deconvolution* terhadap rasio blur yang membuat informasi dari lebih terlihat dan berarti kesalahan kuadrat lebih rendah menunjukkan jumlah kesalahan *deblurring* yang lebih sedikit, hasil tersebut juga mendukung (Goilkar & Yadav, 2021).

3.2 *Preprocessing Denoising* Citra

Denoising dengan algoritma *median filter* (sejenis *filter* linier) ke gambar secara adaptif. *Filter* median menyesuaikan dirinya dengan varian citra. Dengan varian besar, algoritma median *filter* melakukan sedikit penghalusan yang merupakan varian kecil sehingga melakukan lebih banyak penghalusan.

Citra digital dengan noise Denoising citra

Gambar 4 *Preprocessing Denoising* Citra

Dari hasil percobaan seperti pada Gambar 4, median *filter* terbukti efektif menghilangkan *noise* dan menampilkan informasi lebih pada citra, karena hanya memproses piksel yang bermasalah dan tetap mempertahankan garis tepi citra. Hasil yang sama juga terpadat pada (George et al., 2018).

3.3 *Preprocessing Brightness Adjustment* Citra

Penyesuaian kecerahan tiga fungsi seperti: imadjust, histeq, dan adapthisteq. Dengan menggunakan pengaturan default, bandingkan efektivitas ketiga teknik tersebut seperti pada Gambar 5.

1) IMADJUST meningkatkan kontras gambar dengan memetakan nilai-nilai gambar intensitas input ke nilai-nilai 1%.
2) HISTEQ melakukan pemerataan histogram dengan meningkatkan kontras gambar dengan mengubah nilai dalam gambar intensitas sehingga histogram gambar output yang cocok dengan histogram tertentu.
3) ADAPTHISTEQ juga melakukan pemerataan histogram seperti histeq, namun ia beroperasi pada bagian pixel pada citra.

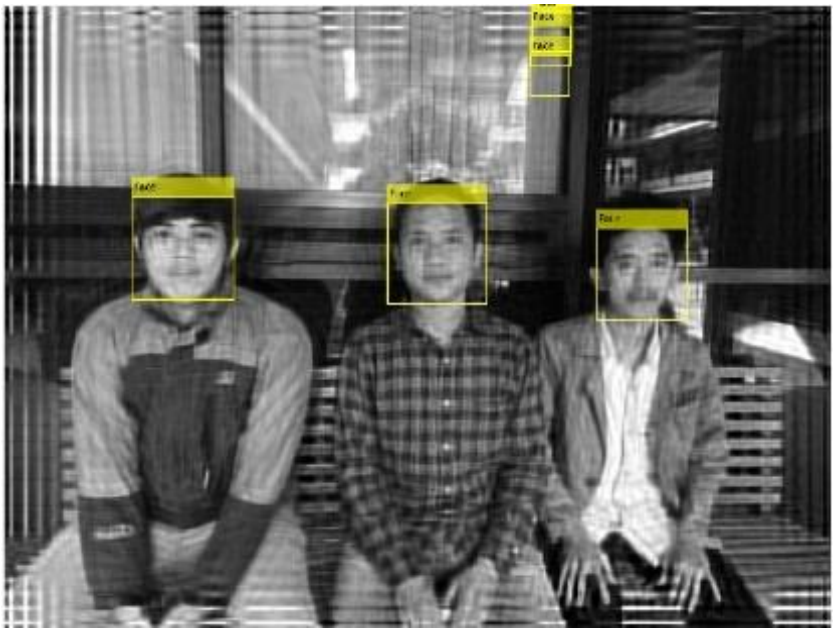
Citra kecerahan rendah Imadjust Histeq Adapthisteq

Gambar 5 *Preprocessing Brightness Adjustment* Citra

Fungsi yang digunakan melakukan penyesuaian gambar dengan metode pemerataan *histogram*, hasil yang diberikan dengan metode *equalization histogram* dapat meningkatkan kualitas gambar, sehingga informasi dalam gambar lebih jelas terlihat. Hasil penelitian yang sama juga ditemukan di (Oktavianto & Purboyo, 2018).

3.4 Deteksi Wajah dengan Viola-Jones

Algoritma Viola-Jones diimplementasikan pada fungsi *cascade object detecti*on, yaitu mendeteksi objek wajah pada citra dengan memuat kotak pada citra. Detektor kemudian menggunakan klasifikasi untuk memutuskan apakah kotak berisi objek wajah. Ukuran kotak bervariasi untuk mendeteksi objek wajah pada skala yang berbeda tetapi dalam rasio yang konsisten. Setelah wajah terdeteksi, sistem memotong setiap kotak agar tingkat deteksi lebih akurat karena tidak ada objek yang dapat dideteksi kecuali wajah.



Gambar 6 Deteksi Wajah Pada Citra

Fungsi algoritma Viola-Jones yang hasilnya ditunjukkan pada Gambar 6 berfungsi sangat baik dalam mendeteksi wajah, namun juga terdapat kesalahan di mana terdapat objek yang bukan wajah dideteksi sebagai wajah, namun itu tidak menjadi masalah dalam penelitian ini.

JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 8, No. 1, JANUARI, 2023: 78 – 89

3.5 Analisis pengenalan wajah

Sistem menerima sampel citra berupa objek wajah yang telah dikonversikan ke dalam matriks lalu dianalisis *eigenvalue* pada citra terbut dengan *eigenvalue* yang ada pada semua citra pada *dataset* wajah. Dengan nilai mean pada setiap dimensi citra (MxN), maka jumlah maksimal *eigenvalue* dengan nilai bukan nol pada matrik kovariannya adalah min[M-1,N-1], karena jumlah dimensi (piksel) dari setiap vektor citra sangat tinggi dibandingkan dimensi citra yang akan di analisa, sehingga jumlah nilai maksimal eigen tanpa nol adalah P-1 (P adalah jumlah citra pada *dataset*). Dengan menggunakan aturan Kaiser untuk menemukan jumlah *eigenvector* yaitu jika nilai eigenvalue lebih besar daripada 1 maka *eigenvector* akan dipilih untuk dibuat sebagai *eigenface*, kemudian akan dibuat tampilah nilai matrik dari *eigenface*, dari nilai tersebut akan dibuat *array* seperti pada Gambar 7.

```
Columns 1 through 12

  1.8066    1.5079    2.3891    0.8396    1.4483    3.6112

Columns 13 through 24

  1.3711    2.9219    3.8254    3.8081    0.7493    3.3261

Columns 25 through 28

  4.2953    3.1649    1.4309    0.7419
```

Gambar 7 Nilai Jarak Euclidean dalam Bentuk Array

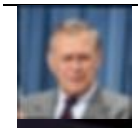
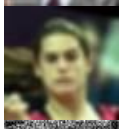
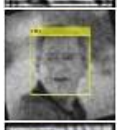
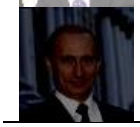
Setelah nilai dari *eigenface* dari citra sampel didapatkan, akan dilakukan proses penghitungan jarak euclidean dari masing matrik *eigenface* dan dicari nilai minimalnya karena dalam citra tersebut terdapat objek wajah yang menyerupai atau sama dengan objek yang citra sampel.

Citra sampel Citra wajah dikenali

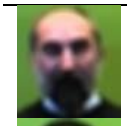
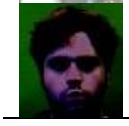
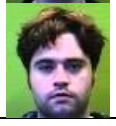
Gambar 8 Hasil Pengujian Pengenalan Wajah

Proses pengenalan wajah dikatakan selesai jika didapat citra wajah dengan nilai euclidean terkecil yang menandakan kedua objek wajah tersebut adalah sama, seperti pada Gambar 8 karena citra tersebut memiliki nilai euclidean terkecil dari citra lainnya pada *dataset*.

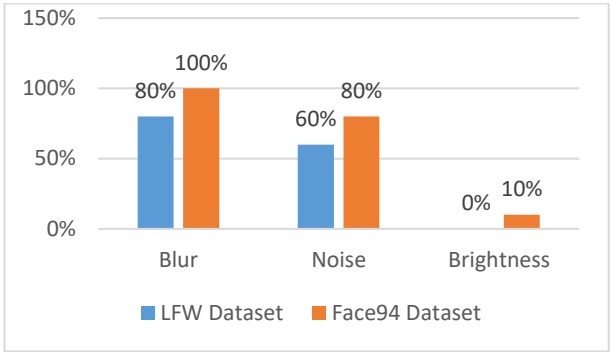
Tabel 1 Hasil Analisis Wajah pada *Dataset* LFW

Sampel	Masalah	*Preprocessing & Deteksi*	*Output*	Akurasi, Presisi, *Recall*	Hasil
	Blur			96%, 0%, 0%	False
	Blur			88%, 0%, 14%	True
	Noise			96%, 1%, 20%	True
	Noise			80%, 0%, 0%	False
	Kecerahan tinggi			99%, 0%, 0%	False
	Kecerahan rendah			74%, 0%, 0%	False

Tabel 2 Hasil Analisis Wajah pada *Dataset* Face94

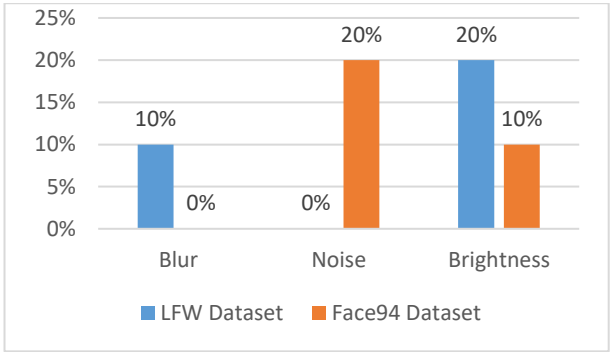
Sampel	Masalah	*Preprocessing & Deteksi*	*Output*	*Akurasi, Presisi, Recall*	Hasil
	Blur			99%, 100%, 100%	True
	Blur			99%, 100%, 65%	True
	Noise			99%, 0%, 0%	False
	Noise			99%, 48%, 52%	True
	Kecerahan tinggi			92%, 0%, 0%	False
	Kecerahan rendah			97%, 17%, 100%	True

Berdasarkan analisis yang dilakukan pada Tabel 1 dan 2, persentase keakuratan pengenalan wajah pada PCA dari *eigenfaces* dihitung dengan menggunakan *detection rate*. Pada setiap *dataset* terdapat 30 citra yang dianalisis dengan 10 citra untuk setiap permasalahan citra.



Gambar 9 Grafik tingkat keakuratan pengenalan wajah dengan PCA

Jumlah wajah yang berhasil dikenali pada LFW *Dataset* adalah 14 dari 30 wajah, jadi *detection rate*-nya adalah 46,66% yang terdiri dari 80% pada citra permasalahan *blur*, 60% pada citra permasalahan *noise,* dan 0% pada citra permasalahan kecerahan. Pada dan face94 *Dataset* adalah 19 dari 30 wajah jadi *detection rate*-nya sebanyak 63,33% terdiri dari 100% pada citra permasalahan *blur*, 80% pada citra permalahan *noise,* dan 10% pada citra permasalahan kecerahan. Seperti yang ditunjukkan pada grafik tingkat keakuratan pengenalan wajah dengan PCA pada Gambar 9. Penelitian ini sejalan dengan (Darmawan, 2019) yang menyimpulkan hasil dari pengenalan wajah berbagai pose yaitu tingkat keakuratan di atas 70%.



Gambar 10 Grafik tingkat keakuratan pengenalan wajah dengan LDA

Hasil analisis data pada LDA yang ditunjukkan pada Gambar 10 hampir semua tidak lebih dari 20%. Hal ini menandakan kalau algoritma tidak direkomendasikan untuk menganalisis wajah terutama pada citra yang memiliki permasalahan visual. Penelitian yang sama dengan penggunaan 2 metode pengenalan tersebut adalah yang dilakukan oleh (Cintisa et al., 2019) yang menyimpulkan jika LDA mempunyai tingkat akurasi yang tinggi namun hasil tersebut berbanding terbalik pada penelitian yang khusus meneliti citra digital dengan permasalahan visual ini, tetapi hasil untuk penggunaan PCA cukup selaras dalam menganalisis pengenalan wajah.dengan tingkat keakuratan 100%.

4. KESIMPULAN

Penelitian yang berfokus untuk mendeteksi dan mengenali wajah pada citra digital dengan permasalahan visual khususnya pada permasalahan *blur*, *noise,* dan permasalahan kecerahan menggunakan metode Viola-Jones dengan metode pengenalan PCA dan LDA. Hasil penelitian dan pembahasan yang telah dilakukan menunjukkan bahwa Viola-Jones dapat dilakukan untuk mendeteksi wajah pada citra dengan permasalahan visual dengan dilakukan *preprocessing* terlebih dahulu. Kemudian untuk melakukan pengenalan wajah pada citra digital dengan PCA dan LDA lebih sederhana dan cepat. Jumlah wajah yang berhasil dikenali pada LFW *Dataset* adalah 14 dari 30 wajah dengan *detection rate* sebesar 46,66% yang terdiri dari 80% pada citra permasalahan blur, 60% pada citra permasalahan *noise,* dan 0% pada citra permasalahan

kecerahan. Pada dan face94 *Dataset* adalah 19 dari 30 wajah jadi *detection rate*-nya sebanyak 63,33% terdiri dari 100% pada citra permasalahan *blur*, 80% pada citra permalahan *noise,* dan 10% pada citra permasalahan kecerahan, sedangkan pada LDA hampir semua analisis data tidak sampai 20%. Dari hasil ini dapat disimpulkan bahwa metode Viola-Jones dan PCA dapat dijadikan sebagai metode kombinasi untuk pengenalan pengenalan wajah manusia pada citra digital yang bermasalah secara efektif tapi tidak pada citra dengan permasalahan kecerahan. Hal ini dapat dilihat dari hasil ketepatan pengenalan wajah yang tinggi kecuali pada permasalahan kecerahan.

DAFTAR PUSTAKA

Aggarwal, A., Alshehri, M., Kumar, M., Sharma, P., Alfarraj, O., & Deep, V. (2021). Principal component analysis, hidden Markov model, and artificial neural network inspired techniques to recognize faces. *Concurrency and Computation: Practice and Experience*, *33*(9), e6157. https://doi.org/10.1002/cpe.6157

Al-Ghrairi, A. H. T., Mohammed, A. A., & Sameen, E. Z. (2022). Face detection and recognition with 180 degree rotation based on principal component analysis algorithm. *IAES International Journal of Artificial Intelligence (IJ-AI)*, *11*(2), 593. https://doi.org/10.11591/ijai.v11.i2.pp593-602

Anam, M. K. (2020). 82 Metode Eigenface/Principle Component Analysis (PCA) Untuk Identifikasi Wajah Manusia. *Jutis (Jurnal Teknik Informatika)*, *6*(2), 82–88. https://doi.org/https://doi.org/10.33592/jutis.Vol6.Iss2.133

Ariza-Lopez, F. J., Rodriguez-Avi, J., & Alba-Fernandez, M. V. (2018). Complete Control of an Observed Confusion Matrix. *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, *2018-July*, 1222–1225. https://doi.org/10.1109/IGARSS.2018.8517540

Barnouti, N. H., Al-Mayyahi, M. H. N., & Al-Dabbagh, S. S. M. (2018). Real-Time Face Tracking and Recognition System Using Kanade-Lucas-Tomasi and Two-Dimensional Principal Component Analysis. *2018 International Conference on Advanced Science and Engineering (ICOASE)*, 24–29. https://doi.org/10.1109/ICOASE.2018.8548818

Borade, S. N., Deshmukh, R. R., & Shrishrimal, P. (2016). Effect of distance measures on the performance of face recognition using principal component analysis. *Advances in Intelligent Systems and Computing*, *384*, 569–577. https://doi.org/10.1007/978-3-319-23036-8_50/COVER

Cintisa, N., Suhartono, E., & Aulia, S. (2019). Pengenalan Ekspresi Pada Raut Wajah Pada Keselamatan Berkendara Menggunakan Principal Component Analysis (pca) Dan Linear Discriminant Analysis (lda). *EProceedings of Engineering*, *6*(3). https://doi.org/10.34818/EOE.V6I3.11354

Darmawan, A. (2019). Aplikasi Mobile Pengenalan Wajah Secara Real-Time Berbasis Principal Component Analysis. *Ubiquitous: Computers and Its Applications Journal*, *2*(1), 57–66. https://doi.org/10.51804/ucaiaj.v2i1.57-66

ElSayed, A., Mahmood, A., & Sobh, T. (2017). Effect of Super Resolution on High Dimensional Features for Unsupervised Face Recognition in the Wild. *2017 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, *2017-October*, 1–5. https://doi.org/10.1109/AIPR.2017.8457967

Furht, B., Akar, E., & Andrews, W. A. (2018). *Digital Image Processing: Practical Approach*. Springer International Publishing. https://doi.org/10.1007/978-3-319-96634-2

George, G., Oommen, R. M., Shelly, S., Philipose, S. S., & Varghese, A. M. (2018). A Survey on Various Median Filtering Techniques For Removal of Impulse Noise From Digital Image. *2018 Conference on Emerging Devices and Smart Systems (ICEDSS)*, 235–238. https://doi.org/10.1109/ICEDSS.2018.8544273

Goilkar, S. S., & Yadav, D. M. (2021). Implementation of Blind and Non-blind Deconvolution for Restoration of Defocused Image. *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)*, 560–563. https://doi.org/10.1109/ESCI50559.2021.9397046

Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing* (4th ed). Pearson.

Jalal, A. S., Bhatnagar, C., Khan, Mohd. A., & Solanki, M. S. (2016). LBP based face recognition system for multi-view face using single sample per person. *2016 11th International Conference on Industrial and Information Systems (ICIIS)*, *2018-January*, 414–419. https://doi.org/10.1109/ICIINFS.2016.8262976

Knoche, M., Hormann, S., & Rigoll, G. (2021). Cross-Quality LFW: A Database for Analyzing Cross- Resolution Image Face Recognition in Unconstrained Environments. *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, 1–5. https://doi.org/10.1109/FG52635.2021.9666960

Kosasih, R. (2021). Pengenalan Wajah Menggunakan PCA dengan Memperhatikan Jumlah Data Latih dan Vektor Eigen. *Jurnal Informatika Universitas Pamulang*, *6*(1), 1. https://doi.org/10.32493/informatika.v6i1.7261

Lu, W., & Yang, M. (2019). Face Detection Based on Viola-Jones Algorithm Applying Composite Features. *2019 International Conference on Robots & Intelligent System (ICRIS)*, 82–85. https://doi.org/10.1109/ICRIS.2019.00029

Matin, A., Mahmud, F., & Shawkat, M. T. B. (2016). Recognition of an individual using the unique features of human face. *2016 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, 57–60. https://doi.org/10.1109/WIECON-ECE.2016.8009087

Moradmand, H., Aghamiri, S. M. R., & Ghaderi, R. (2020). Impact of image preprocessing methods on reproducibility of radiomic features in multimodal magnetic resonance imaging in glioblastoma. *Journal of Applied Clinical Medical Physics*, *21*(1), 179–190. https://doi.org/10.1002/acm2.12795

Oktavianto, B., & Purboyo, T. W. (2018). A Study of Histogram Equalization Techniques for Image Enhancement. *International Journal of Applied Engineering Research*, *13*(2), 1165–1170. http://www.ripublication.com

Proenca, H., Neves, J. C., Barra, S., Marques, T., & Moreno, J. C. (2016). Joint Head Pose/Soft Label Estimation for Human Recognition In-The-Wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(12), 2444–2456. https://doi.org/10.1109/TPAMI.2016.2522441

Singh, G., & Goel, A. K. (2020). Face Detection and Recognition System using Digital Image Processing. *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, 348–352. https://doi.org/10.1109/ICIMIA48430.2020.9074838

Tang, C., Cai, A., Zhang, W., Zheng, Z., Liang, N., Li, L., & Yan, B. (2020). Joint Regularized-based Image Reconstruction by Combining Super-Resolution Sinogram for Computed Tomography Imaging. *2020 5th International Conference on Communication, Image and Signal Processing (CCISP)*, 188–193. https://doi.org/10.1109/CCISP51026.2020.9273488

Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, *1*, I-511-I–518. https://doi.org/10.1109/CVPR.2001.990517

Xiong, W. (2020). Research on Fire Detection and Image Information Processing System Based on Image Processing. *2020 International Conference on Advance in Ambient Computing and Intelligence (ICAACI)*, 106–109. https://doi.org/10.1109/ICAACI50733.2020.00027

9 772527 583007