

REKOMENDASI SISTEM PEMILIHAN MOBIL MENGUNAKAN K-NEAREST NEIGHBOR (KNN) COLLABORATIVE FILTERING

Ilham Gumantung Gusti¹, Muhammad Nasrun², Ratna Astuti Nugrahaeni³

^{1,2,3}Prodi S1 Teknik Komputer, Fakultas Teknik Elektro, Universitas Telkom,
Bandung, Indonesia

¹iggusti@student.telkomuniversity.ac.id, ²nasrun@telkomuniversity.ac.id,
³ratnaan@telkomuniversity.ac.id

Abstrak

Mobil merupakan kendaraan yang sangat dibutuhkan pada masa ini. Banyak dari pengguna ketika ingin memilih mobil hanya mengetahui sebagian dari informasi mobil yang disukainya tanpa mengetahui informasi mobil lain yang sejenis. Rekomendasi sistem pemilihan mobil merupakan sistem yang dapat digunakan oleh pengguna dalam memilih mobil. Dengan diterapkannya rekomendasi sistem pemilihan mobil, pengguna akan mendapatkan informasi lebih mengenai mobil yang ingin dipilih, dan mobil lain yang mungkin mobil tersebut sama sekali belum diketahui oleh pengguna. Dalam rekomendasi sistem pemilihan mobil, penulis menerapkan metode K-Nearest Neighbor (KNN) *Collaborative Filtering* yang dilakukan berdasarkan jarak kedekatan Data *Testing* dengan Data *Training*. Kedekatan data (kemiripan data) tersebut digunakan untuk merekomendasikan mobil ke pengguna. Hasil yang diperoleh dalam penelitian ini adalah jika ingin mendapatkan 10 mobil terbaik maka jarak maksimal yang digunakan adalah 5%, dan akurasi terbaik didapatkan ketika $K = 10$ yaitu sebesar 95,15%.

Kata Kunci: mobil, K-Nearest Neighbor (KNN), collaborative filtering, sistem perekomendasi.

Abstract

A car is vehicle that is very much needed at this time. Many of the users when they want to choose a car only know some of the information they like without knowing the information of other similar cars. Car selection system recommendation is a system that can be used by users in choosing a car. By applying car selection recommendations, users will get more information about the car that they want to choose, and other cars that may not be known by the user at all. In the recommendation of the car selection system, the authors apply the K-Nearest Neighbor (KNN) Collaborative Filtering method based on web applications where recommendations are made based on the proximity of testing data to training data. The proximity of the data (data similarity) is used to recommend the car to the user. The results obtained in this study is if user want to get the 10 best cars then the maximal distance is 5%, and the best accuracy is obtained when $K = 10$ is equal to 95.15%.

Key Words: car, K-Nearest Neighbor (KNN), collaborative filtering, recommendation system.

1. Pendahuluan

Perkembangan industri otomotif di Indonesia khususnya pada industri mobil sudah sedemikian pesatnya sehingga membuat tingkat persaingan semakin ketat. Dengan semakin beraneka ragamnya merek dan tipe mobil di Indonesia, merupakan dampak dari para produsen mobil yang terus melakukan inovasi terhadap produknya. Akibatnya pengguna harus semakin selektif dalam memilih mobil yang akan dibeli. Menurut [1] faktor yang menjadi pertimbangan bagi pengguna dalam memilih mobil yaitu faktor nilai dan manfaat yang akan diperoleh pengguna dari suatu produk. Banyak aspek yang harus dipertimbangkan dalam membeli sebuah mobil, maka orang sering kali bingung untuk

memilih karena dihadapkan dengan banyaknya merek dan tipe mobil yang ada dipasar. Orang yang telah memilih mobil pun terkadang merasa kecewa karena kurang sesuai dengan yang diharapkan.

Dewasa ini, internet mengalami kemajuan yang sangat cepat, terutama pada aspek mesin pencariannya. Penelitian [2] menyimpulkan mesin pencari itu sendiri merupakan algoritma komputer yang dirancang sedemikian rupa untuk membantu pengguna menemukan informasi yang ingin dicari, misalnya dalam komputer sendiri atau di dalam website. Mesin pencari tidak hanya dapat memungkinkan kita untuk meminta data informasi dengan cara mencari dari hal khusus ke umum namun juga dapat mencari dari hal umum ke khusus.

Menurut [3] mesin pencari merupakan salah satu elemen penting dalam *e-Commerce* yang harus ada, guna mempercepat pencarian data yang pengguna inginkan. Jika pengguna melakukan pencarian di mesin pencari, mungkin akan mendapatkan jutaan hasil relevan dari data yang pengguna masukan. Melihat peranan yang begitu penting dari mesin pencari, maka dibuatlah beberapa metode untuk mempersempit pencarian agar didapatkan hasil yang optimal.

Dalam konteks pencarian, biasanya pengguna ingin mencari informasi yang dibutuhkannya hanya dengan kata kunci yang pengguna pahami untuk mendapatkan data, tetapi data tersebut tidak sesuai dengan database sistem yang ada. Kelemahan tersebut sebenarnya dapat diantisipasi apabila pengguna paham dengan informasi yang akan dicari. Terkait masalah tersebut maka penulis menyarankan agar menggunakan salah satu metode *Collaborative Filtering*, yaitu dengan metode *K-Nearest Neighbor* (KNN). Metode KNN dapat mengantisipasi jika pengguna kurang paham dengan apa yang ingin dicari karena metode ini menerapkan prinsip pencarian menggunakan jarak kedekatan (kemiripan data) sampel dengan database sistem yang ada. Metode KNN ini juga digunakan untuk melakukan klasifikasi terhadap sampel dalam pemilihan mobil pada kategori yang tepat.

2. Dasar Teori

2.1 Metode K-Nearest Neighbor (KNN)

Dasar metode KNN, antara lain:

- temukan *database* sistem yang paling dekat dengan data sampel yang tidak diketahui kelasnya,
- ambil data dari *database* sistem yang paling sering muncul untuk klasifikasi *rating* nilai K.

Menurut [4] metode kombinasi klasifikasi berbasis *voting* adalah metode sederhana dan umum untuk digunakan dalam semua jenis masalah klasifikasi. *Voting* juga secara komputasi merupakan cara yang efektif untuk menggabungkan hasil klasifikasi yang terpisah, karena hasil klasifikasi akhir diputuskan hanya berdasarkan suara yang diberikan oleh penggolong dasar dan tidak diperlukan pelatihan klasifikasi dalam klasifikasi akhir.

Merujuk pada hasil penelitian [5-10], maka persamaan Metode KNN yang penulis terapkan adalah:

$$D(x,y) = \sqrt{\sum_{i=0}^n (x_i - y_i)^2} \quad (1)$$

dengan $D(x,y)$ adalah Jarak Euclidean (Euclidean Distance), (x_i) adalah nilai Data *Testing* ke- i , dan (y_i) adalah nilai Data *Training* ke- i .

Peneliti [6] menyatakan kelebihan dari metode KNN, antara lain:

- keakuratan hasil yang diperoleh lebih dijamin,
- untuk Data *Training* yang besar, hasilnya akan lebih efektif.

Akan tetapi, kekurangan dari metode KNN, antara lain:

- berdasarkan perhitungan nilai jarak (*Distance Based Learning*), tidak jelas atribut mana yang memberikan hasil yang baik dan perhitungan jarak mana yang sebaiknya digunakan,
- peneliti perlu menghitung nilai baru ke semua data yang ada pada Data *Training* dan menghitung jarak karena nilai komputasinya tinggi,
- parameter K perlu ditunjukkan (jumlah tetangga terdekat).

2.2 Collaborative Filtering

Merujuk pada jurnal [11] dalam teknik penyaringan kolaboratif, berbagai macam informasi tentang perilaku pengguna, rutinitas atau bahkan pilihan biasanya diakumulasikan, dipelajari, dan merekomendasikan dengan tepat apa yang pelanggan butuhkan berdasarkan kemiripan mereka dengan pelanggan lain. Ketika sistem memberikan timbal balik kepada pengguna dengan mengolah data-data tersebut, sebagai gambaran sistem mengharuskan pemberian skala yang mengindikasikan penilaian yang paling tidak disukai hingga paling disukai menurut sudut pandang pengguna, data ini memungkinkan untuk dilakukannya perhitungan statistik yang hasilnya menunjukkan produk mana yang paling disukai oleh pengguna.

2.3 Text to Vector Space Model

Berdasarkan jurnal [12-15] yang melakukan penelitian mengenai strategi baru dimana dokumen dikodekan ke dalam vektor dan modifikasi KNN agar dapat beradaptasi dengan vektor *string* untuk kategorisasi teks. Secara tradisional, ketika KNN digunakan untuk klasifikasi pola, data mentah harus dikodekan ke dalam vektor numerik. Pengkodean ini mungkin sulit, tergantung pada area aplikasi klasifikasi pola yang diberikan. Sebagai contoh dalam kategorisasi teks, penyandian teks lengkap yang diberikan sebagai data mentah ke dalam vektor numerik mengarah ke dua masalah utama, yaitu dimensi besar dan distribusi jarang. Merujuk pada jurnal [16], penulis menyandikan teks lengkap menjadi korelasi antara dua vektor yang sesuai menggunakan Persamaan Cosinus Similarity (2).

$$\text{SIM}(A,B) = \frac{\vec{A} \cdot \vec{B}}{|\vec{A}| \times |\vec{B}|} \quad (2)$$

dengan $SIM(A,B)$ adalah korelasi (*similarity*), \vec{A} adalah vektor *testing*, dan \vec{B} adalah vektor *training*.

2.4 Parameter

Berdasarkan jurnal [7], untuk klasifikasi mobil terbaik tidak mudah, karena kekurangan dan kelebihan dimiliki pilihan yang lain dan masing-masing membahas keputusan untuk memilih alternatif mobil terbaik. Menurut [17,18] selama ini pengambil keputusan cenderung bias karena mempertimbangkan keterbatasan kognitif yang dimiliki manusia dapat memungkinkan besarnya pilihan yang lebih banyak ditentukan oleh karakteristik pengguna itu sendiri dan seberapa besar pihak penjual dapat menarik hati pengguna.

3. Analisis dan Perancangan Sistem

Pada sistem yang akan dibuat, penulis mengidentifikasi bahwa masalah inti yang dibahas dalam penelitian ini adalah mencari tetangga terdekat dari parameter masukan yang diberikan pengguna terhadap Data *Training* yang dimiliki sistem. Pada penelitian sebelumnya, yaitu pada penelitian yang dilakukan oleh [19] parameter yang digunakan pada metode K-Nearest Neighbor (KNN) ada 7 macam, diantaranya:

1. tujuan pembelian,
2. harga,
3. tahun pembuatan,
4. kapasitas penumpang,
5. warna,
6. kapasitas mesin, dan
7. jenis transmisi.

Untuk menetapkan parameter masukan yang akan diberikan pengguna, penulis mendapatkan datanya dari survei di lapangan, karena dikhawatirkan parameter yang lama sudah tidak relevan lagi dengan keadaan saat ini. Sedangkan untuk Data *Training* yang penulis pakai, penulis mencarinya pada internet dan menanyakannya langsung pada *showrooms* mobil terkait. Tabel 1 berikut merupakan hasil survei pada *showrooms* untuk mencari parameter membeli mobil yang relevan di Indonesia:

Tabel 1a. Parameter yang relevan di Indonesia

Showroom	Narasumber	Parameter yang paling mempengaruhi
Daihatsu Astra Beiz Center	Angga	Merk, harga, jenis transmisi, warna, cc, seater, harga purna jual, dan jenis bahan bakar minyak.
Honda Ahmad Yani	Reza	Merk, harga, cc, jenis transmisi, warna, harga purna jual, dan seater.

Tabel 1b. Parameter yang relevan di Indonesia

Showroom	Narasumber	Parameter yang paling mempengaruhi
Hyundai Bandung Center	Joko	Merk, harga, jenis transmisi, warna, cc, dan seater.
Mazda Bandung	Handoko	Jenis transmisi, harga, seater, warna, cc, dan merk.
Suzuki Bandung	Ridwan	Merk, harga, jenis transmisi, seater, warna, cc, dan jenis bahan bakar minyak.
Toyota	George	Merk, jenis transmisi, harga, warna, seater, cc, dan jenis transmisi.
Wuling Bandung	Aji	Merk, harga, seater, transmisi, warna, dan cc.

Dari banyaknya parameter keputusan membeli dan parameter mobil yang didapat dari narasumber seperti yang ditunjukkan pada Tabel 1, penulis mengerucutkan menjadi beberapa parameter yang digunakan pada sistem, diantaranya:

1. nama merek mobil,
2. nama model mobil,
3. nama varian mobil,
4. warna mobil,
5. harga mobil,
6. CC,
7. jenis transmisi,
8. jumlah tempat duduk,
9. harga purna jual, dan
10. BBM.

Metode KNN merupakan metode yang sesuai untuk pemecahan masalah yang ada, karena parameternya saling *independent* dan Data *Training* yang dimiliki penulis tidak memiliki *noise*, dikarenakan setiap data merupakan data yang penting. Dimensi yang digunakan pada sistem pun tidak perlu ditambah lagi. Hal-hal tersebut merupakan *point* penting yang penulis analisis dimana penulis memutuskan menggunakan metode KNN daripada metode pengklasifikasian yang lain, seperti: Support Vector Machine, Decision Trees, Naïve Bayes, dan Logistic Regression.

Langkah pemrosesan metode KNN sebagai berikut. Mula-mula pengguna memasukkan data mobil atau parameter mobil yang diketahui yang ingin direkomendasikan. Langkah selanjutnya adalah mengelompokkan Data *Testing* dengan Data *Training* berdasarkan parameternya. Kemudian program akan melakukan pengecekan parameter tersebut, apakah berbentuk teks atau bukan. Jika parameter tersebut berbentuk teks, maka tahap selanjutnya adalah mengubahnya menjadi *uppercase* atau huruf kapital semua. Lalu tahap selanjutnya mengubahnya menjadi *Vector Space Model*. Gambar

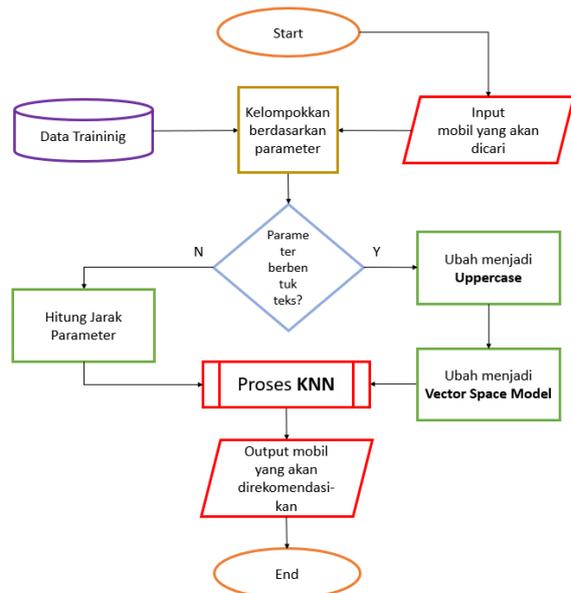
1 berikut merupakan Pseudocode Algoritma dari cara merubah teks menjadi *Vector Space Model*.

```

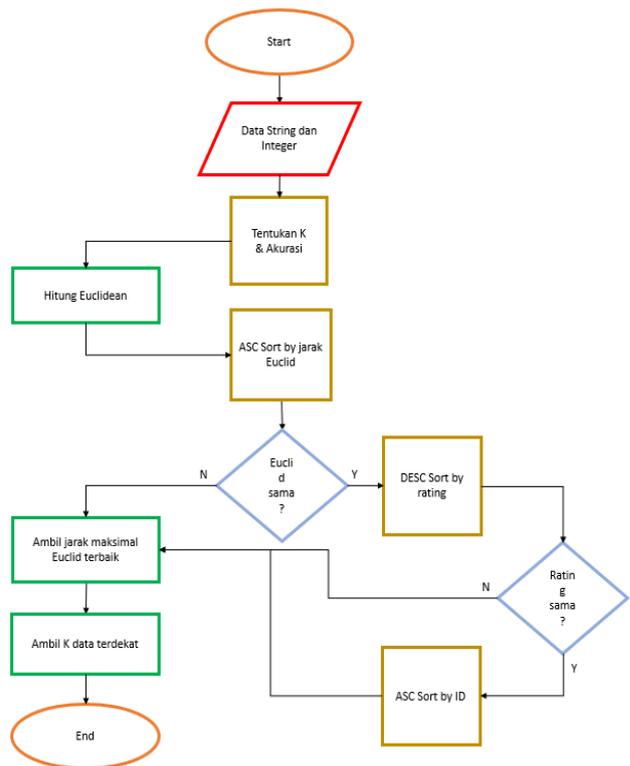
Algoritma Text to Vector Space Model
Deklarasi :
    atas, bawah1, bawah2, bawah,
    vektor : float
    i, j, a : integer
    asciitra, asciites : integer
    datatraining, datatesting :
string
Deskripsi :
    begin
        atas <- 0
        bawah1 <- 0, bawah2 <- 0
        for i in datatraining do
            for j in i do
                asciidtra[a] <-
                ascii(datatraining[i][j])
                a <- a+1
            endfor
        endfor
        for i in datatesting do
            for j in i do
                asciidtes[b] <-
                ascii(datatesting[i][j])
                b <- b+1
            endfor
        endfor
        for i in
        range(length(asciidtes))
        atas <-
        atas+(asciidtes[i]*asciidtra[i])
        endfor
        for i in asciidtes do
            bawah1 <- bawah1+i^2
        endfor
        for i in asciidtra do
            bawah2 <- bawah2+i^2
        endfor
        bawah <-
        (bawah1^0.5) * (bawah2^0.5)
        vektor <- divide(atas,bawah)
    end
    
```

Gambar 1. Pseudocode Algoritma teks menjadi *Vector Space Model*

Langkah selanjutnya adalah jika parameter tersebut bukan parameter teks, maka parameter tersebut akan dihitung jaraknya. Kemudian mengolah data yang ada menggunakan metode K-Nearest Neighbor (KNN). Langkah terakhir adalah mengeluarkan *output* yang dihasilkan proses KNN. Jurnal ini memang sedikit berbeda dari jurnal-jurnal rujukan yang hanya meneliti KNN menggunakan Persamaan Cosinus Similarity saja atau hanya menggunakan KNN biasa untuk menghitung hasil penelitiannya. Gambar 2 di bawah merupakan diagram gambaran langkah pemrosesan metode KNN.



Gambar 2. Diagram alir alur pemrosesan metode KNN



Gambar 3. Diagram alir alur inti pemrosesan metode KNN

Gambar 3 di atas merupakan diagram alir alur inti pemrosesan metode KNN. Mula-mula siapkan *Data Testing* dan *Data Training*. Langkah selanjutnya adalah penentuan banyaknya K dan Jarak Maksimal yang akan digunakan. Kemudian adalah penghitungan nilai Jarak Euclidean antara *Data Training* dengan *Data Testing*. Setelah itu dilakukan pengurutan secara *ascending* terhadap hasil Jarak Euclidean. Langkah selanjutnya jika ada

Jarak Euclidean-nya yang sama maka data akan diurutkan secara *descending* terhadap *rating*. Kemudian langkah selanjutnya jika ada *rating* yang Jarak Euclidean-nya sama, ternyata sama juga. Maka dilakukan langkah selanjutnya, yaitu pengurutan secara *ascending* berdasarkan ID mobilnya. Kemudian jika semuanya telah terurut, langkah berikutnya adalah menyeleksi Jarak Euclidean-nya berdasarkan Jarak Maksimal yang sudah ditetapkan. Langkah terakhir adalah menyeleksi data tersebut menggunakan K yang sudah ditetapkan juga. Gambar 4 berikut ini merupakan Pseudocode Algoritma dari inti proses metode K-Nearest Neighbor (KNN).

```

Algoritma Inti Proses Metode KNN
Deklarasi :
    i,j,a : integer
    datatraining : string
    vektor, normalparameter, dalam, KNN : float
Deskripsi :
    begin
        for i in datatraining do
            for j in vektor&normalparameter do
                dalam <- dalam+(j^2)
            endfor
            KNN[a] <- dalam^0.5
            a <- a+1
        endfor
        sortbyjarak(KNN)
        sortbyrating(KNN)
        sortbyID(KNN)
    end
    
```

Gambar 4. Pseudocode Algoritma inti pemrosesan metode KNN

Dalam jurnal [11] pada teknik *Collaborative Filtering*, berbagai macam informasi tentang perilaku pengguna, rutinitas atau bahkan pilihan biasanya diakumulasikan, dipelajari, dan merekomendasikan dengan tepat apa yang pengguna butuhkan berdasarkan kemiripan mereka dengan pengguna lain. Salah satu cara mempelajari kegemaran pengguna adalah dengan memaksanya memberikan *rating* terhadap suatu mobil. Aktivitas memberikan *rating* mobil merupakan elemen terpenting dari sistem yang penulis gunakan ini, *rating* diperoleh dari pengguna dimana pengguna secara *explicit* memberikan penilaiannya minimal terhadap 3 mobil yang berbeda pada awal registrasi aplikasi. Dan pada akhirnya pengguna bebas memberikan *rating* kepada mobil mana pun sesuka hati.

Kesimpulannya ialah sistem memberikan timbal balik kepada pengguna dengan mengolah data-data masukan dan *rating* tersebut. Sebagai gambaran *rating* yang digunakan adalah dari skala nol sampai lima yang mengindikasikan penilaian mobil yang paling tidak disukai hingga mobil paling disukai menurut sudut pandang pengguna. Data ini memungkinkan untuk dilakukannya perhitungan statistik yang hasilnya menunjukkan mobil mana yang diberikan *rating* tinggi oleh kebanyakan pengguna.

4. Pengujian Sistem

Pada Pengujian Sistem yang dibuat, penulis menggunakan 20 Data *Testing* dengan 1088 Data *Training*. Berikut merupakan 3 *sample* dari 20 Data *Testing* yang penulis gunakan, antara lain:

- Data *Testing* ke-1 : Honda / Brio / Manual / - / - / - / Silver / - / - / - ,
- Data *Testing* ke-2 : Mitsubishi / E-class / 1.0 g m/t / - / - / - / - / - / - / - ,
- Data *Testing* ke-3 : Daihatsu / Xenia / - / - / - / - / - / Black / 7 / - / Gasoline.

4.1 Pengujian Pengaruh Nilai Persentase Jarak Euclidean terhadap Jumlah Anggota

Tujuan pengujian ini adalah dengan adanya nilai persentasi Jarak Euclidean yang berbeda, kita dapat mengetahui jumlah mobil terbaik yang akan direkomendasikan kepada pengguna. Pada pengujian ini digunakan nilai persentasi Jarak Euclidean yang berbeda yaitu Jarak Maksimal = 5%, Jarak Maksimal = 10%, Jarak Maksimal = 20%, Jarak Maksimal = 40%, Jarak Maksimal = 80% dan tiga Data *Testing* yang berbeda. Jumlah Data *Training* yang digunakan adalah sebanyak 1088 mobil.

Tabel 2. Skenario Pengujian Pengaruh Nilai Persentasi Jarak Euclidean terhadap Jumlah Anggota

Data <i>Testing</i> ke-	Jarak Maksimal Pengujian				
	1	2	3	4	5
1	5%	10%	20%	40%	80%
2	5%	10%	20%	40%	80%
3	5%	10%	20%	40%	80%
...
20	5%	10%	20%	40%	80%

Pada Data *Testing* pertama dengan pengujian 5% dilakukan perhitungan Jarak Euclid terdekat senilai 0.0 dan Jarak Euclid terjauh senilai 0,62, maka perhitungan Jarak Maksimalnya, yaitu:

$$\text{Jarak Maks} = ((x - y) \times \text{target}) + y \quad (3)$$

dengan x adalah Jarak Euclidean terjauh, y adalah Jarak Euclidean terdekat, dan target adalah target Jarak Maksimal.

Berikut merupakan perhitungan Jarak Maksimal menggunakan Persamaan Mencari Jarak Maksimal (3), yaitu:

$$J. \text{ Maksimal } 5\% \text{ Data } \textit{Testing} \ 1 = ((0,62 - 0.0) \times 0,05) + 0,0 = 0,03$$

Dengan menghitung mobil yang mempunyai Jarak Euclidean di bawah Jarak Maksimal, maka didapatkan sebanyak 8 mobil. Dengan melakukan

perlakuan yang sama kepada seluruh Jarak Maksimal Pengujian, didapatkan hasil sebagai berikut:

Tabel 3. Hasil Pengujian Data *Testing* 1 Pengaruh Nilai Persentase Jarak Euclidean terhadap Jumlah Anggota

Jarak Maksimal Pengujian	Data <i>Testing</i> ke-1			
	E(0)	E(-1)	Jarak Maksimal	Jumlah anggota
5%	0,0	0,62	0,03	8
10%			0,06	12
20%			0,12	42
40%			0,25	314
80%			0,50	955

Tabel 4. Hasil Pengujian Data *Testing* 2 Pengaruh Nilai Persentase Jarak Euclidean terhadap Jumlah Anggota

Jarak Maksimal Pengujian	Data <i>Testing</i> ke-2			
	E(0)	E(-1)	Jarak Maksimal	Jumlah anggota
5%	0.16	0.95	0,20	12
10%			0,24	68
20%			0,31	180
40%			0,47	560
80%			0,79	1056

Tabel 5. Hasil Pengujian Data *Testing* 3 Pengaruh Nilai Persentase Jarak Euclidean terhadap Jumlah Anggota

Jarak Maksimal Pengujian	Data <i>Testing</i> ke-3			
	E(0)	E(-1)	Jarak Maksimal	Jumlah anggota
5%	0.0	1.47	0,07	12
10%			0,15	41
20%			0,30	271
40%			0,59	792
80%			1,18	1076

Tabel 6. Kesimpulan Pengujian Pengaruh Nilai Persentase Jarak Euclid terhadap Jumlah Anggota

Jarak Maksimal Pengujian	Rata-rata anggota
5%	13
10%	43
20%	197
40%	691
80%	1061

Dengan mengamati tabel 6, kita dapat mengetahui Jumlah Rata-rata Anggota yang dimiliki pada setiap Jarak Maksimal Pengujian. Dimana jika ingin mendapatkan 10 mobil (K = 10) terbaik, minimal Jarak Maksimal Pengujian yang digunakan adalah 5%. Maksudnya adakah ketika sistem di atur menggunakan Jarak Maksimal Pengujian di bawah

5%, maka mobil yang akan di rekomendasikan kepada pengguna pasti di bawah 10 mobil.

4.2 Pengujian Pengaruh Nilai K terhadap Akurasi

Tujuan pengujian ini adalah untuk mengetahui pengaruh banyaknya tetangga terhadap akurasi dan untuk mengetahui nilai K yang paling optimal pada penentuan aplikasi ini. Pada pengujian ini digunakan nilai K yang berbeda yaitu K yang berbeda yaitu K = 10, K = 50, K = 100, K = 500, dan K = 1000. Jumlah data latih yang digunakan adalah sebanyak 1088. Pada Data *Testing* pertama dengan pengujian 5% dilakukan perhitungan Jarak Euclid terdekat senilai 0.0 dan Jarak Euclidean terjauh senilai 0,62 dengan mengacu pada Persamaan Mencari Jarak Maksimal (3), maka turunan Persamaan Akurasi (4) yang penulis gunakan yaitu:

$$Akurasi = \left(1 - \frac{K-x}{y-x}\right) \times 100 \quad (4)$$

dengan K adalah Jarak Euclidean ke K, x adalah Jarak Euclidean Awal, dan y adalah Jarak Euclidean Akhir.

Berikut merupakan contoh perhitungan akurasi Data *Testing* 1 berdasarkan K menggunakan Persamaan Akurasi (4):

- K (10) = 0,06:
 $Akurasi = \left(1 - \frac{0,06 - 0,0}{0,62 - 0,0}\right) \times 100 = 90,37\%$

Dengan melakukan perlakuan yang sama kepada seluruh Data *Testing*, didapatkan hasil sebagai berikut:

Tabel 7. Hasil Pengujian Data *Testing* 1 sampai 3 Pengaruh Nilai K terhadap Akurasi

Nilai K	Data <i>Testing</i> ke-1 (%)	Data <i>Testing</i> ke-2 (%)	Data <i>Testing</i> ke-3 (%)
K = 10	90,37	95,13	99,25
K = 50	76,99	91,00	88,44
K = 100	70,62	86,53	86,53
K = 500	52,49	61,90	69,06
K = 1000	13,39	29,62	26,88

Tabel 8. Kesimpulan Pengujian Pengaruh Nilai Nilai K terhadap Akurasi

Nilai K	Rata-rata Data <i>Testing</i> (%)
K = 10	95,15
K = 50	87,84
K = 100	84,04
K = 500	66,41
K = 1000	34,55

Pada hasil pengujian pengaruh nilai K terhadap akurasi yang ditampilkan pada tabel 8, dapat diketahui bahwa akurasi terbesar dihasilkan pada jumlah K = 10 yaitu sebesar 95,15%. Sementara itu, nilai K yang berbeda menghasilkan akurasi yang berbeda pula sehingga dapat disimpulkan bahwa nilai K berpengaruh terhadap akurasi yang dihasilkan meskipun akurasi yang berubah tidak terlalu signifikan. Berdasarkan pada tabel 8 juga dapat ditarik kesimpulan bahwa semakin besar nilai K, maka akurasi yang dihasilkan akan menurun. Hal ini dikarenakan semakin besar nilai K, semakin banyak data yang diambil dalam menghasilkan klasifikasi sehingga semakin banyak pula data yang tidak relevan dan mempengaruhi kesalahan dalam prediksi. Dari hasil simulasi, penelitian menunjukkan bahwa K = 10 pada sistem bukan hanya atas dasar akurasi yang besar, tetapi juga berdasarkan waktu pencarian yang akan dilakukan oleh sistem.

5. Kesimpulan

Pada penelitian ini, *rating* yang diberikan oleh pengguna merupakan suatu data *explicit* yang dapat dipelajari dan dihitung menggunakan metode K-Nearest Neighbor (KNN). Kemudian pada sistem rekomendasi metode KNN jika ingin mendapatkan 10 mobil (K = 10) terbaik, Jarak Maksimal Pengujian yang digunakan adalah 5%. Selain itu, pada sistem rekomendasi metode KNN diketahui bahwa akurasi terbesar dihasilkan pada jumlah K = 10 yaitu sebesar 95,15%.

Daftar Pustaka

- [1] W. He, Z. Li, and G. Xie, "Research and Development Trends of Car Networking," *Commun. Comput. Inf. Sci.*, vol. 134, no. PART 1, pp. 296–301, 2011.
- [2] A. Jain, "The Role and Importance of Search Engine and Search Engine Optimization," vol. 2, no. 3, pp. 99–102, 2013.
- [3] T. Klatt, "E-Commerce Business Models and Search Engine Dependency," vol. 2013, no. September, pp. 205–208, 2013.
- [4] L. Lepisto, I. Kunttu, and A. Visa, "Rock Image Classification Based on K-Nearest Neighbour Voting," *Chinese J. Electron.*, vol. 14, no. 1, pp. 115–118, 2005.
- [5] W. P. Sari, E. Cahyaningsih, D. I. Sensuse, and H. Noprisson, "The Welfare Classification of Indonesian National Civil Servant using TOPSIS and K-Nearest Neighbour (KNN)," *Proc. - 14th IEEE Student Conf. Res. Dev. Adv. Technol. Humanit. SCORED 2016*, 2017.
- [6] A. Brandrud, *Nearest Neighbor Frame Classification for Articulatory Speech Recognition*, no. January. 2015.
- [7] A. Nouvel, "Klasifikasi Kendaraan Roda Empat Berbasis KNN," *Bianglala Inform.*, vol. 3, no. 2, pp. 66–69, 2015.
- [8] F. Hermawan and H. Agung, "Implementasi Metode K-Nearest Neighbor pada Aplikasi Data Penjualan PT. Multitek Mitra Sejati," pp. 103–109.
- [9] N. A. Putra, A. T. Putri, D. A. Prabowo, L. Surtiningsih, R. Arniantya, and I. Cholissodin, "Klasifikasi Sepeda Motor Berdasarkan Karakteristik Konsumen dengan Metode K-Nearest Neighbour Pada Big Data Menggunakan Hadoop Single Node Cluster," vol. 4, no. 2, pp. 81–86, 2017.
- [10] A. Sukma, D. Ramadhan, B. P. Santoso, T. R. Sari, N. M. A. K. Wiraswari, and B. Zaman, "K – Nearest Neighbor Information Retrieval (Sistem Temu Kembali Informasi)," 2014.
- [11] A. Thomas and S. A. K., "Comparative Study of Recommender Systems," 2016.
- [12] R. Al-Shalabi, G. Kanaan, and M. H. Gharaibeh, "Arabic Text Categorization Using kNN Algorithm," no. February, 2016.
- [13] T. Jo, "K Nearest Neighbor for Text Summarization using Feature Similarity," pp. 0–4, 2017.
- [14] T. Jo, "Inverted Index based Modified Version of KNN for Text Categorization," vol. 4, no. 1, pp. 17–26, 2008.
- [15] T. Jo, "Modifying KNN for Representing Documents into String Vectors for Text Categorization."
- [16] C. S. D. Prasetya, "Sistem Rekomendasi pada E-Commerce Menggunakan K-Nearest Neighbor," vol. 4, no. 3, pp. 194–200, 2017.
- [17] Y. Suhari, "E-Commerce : Model Perilaku Konsumen," *J. Din. Inform.*, vol. 2, no. 2, 2010.
- [18] I. Widiyanto and S. L. Prasilowati, "Perilaku Pembelian Melalui Internet," vol. 17, no. 2, pp. 109–112, 2015.
- [19] N. L. G. P. Suwirmayanti, "Penerapan Metode K-Nearest Neighbor untuk Sistem Rekomendasi Pemilihan Mobil," vol. 16, no. 2, pp. 120–131, 2017.