



Sistem Rekomendasi Pemilihan Laptop Menggunakan Metode *Content Based Filtering* Dan *K-Nearest Neighbor*

Ilzam Rojabi*, Ilham Saifudin, Guruh Wijaya

Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Jember

Email: ilzamrojabi8@gmail.com*, ilham.saifudin@unmuhjember.ac.id, guruh wijaya@unmuhjember.ac.id

ABSTRAK

Sistem rekomendasi memiliki peran penting dalam membantu pengguna menemukan produk yang tepat di antara banyaknya pilihan. Penelitian ini bertujuan mengembangkan sistem rekomendasi laptop menggunakan *Metode Content-Based Filtering* dan *K-Nearest Neighbors* (KNN). Sistem ini dirancang untuk memberikan saran laptop berdasarkan spesifikasi dan harga. Dataset yang digunakan mencakup atribut penting seperti RAM, SSD, HDD, sistem operasi, dan prosesor. Penelitian ini menggunakan TF-IDF (*Term Frequency-Inverse Document Frequency*) untuk mengukur bobot atribut setiap laptop dan *cosine similarity* untuk menilai kesamaan antar laptop. Metode KNN digunakan untuk menemukan laptop yang paling mirip berdasarkan atribut harga yang dipilih pengguna. Dataset diambil dari Kaggle dan diproses menggunakan berbagai pustaka Python seperti pandas, numpy, dan scikit-learn. Hasil penelitian menunjukkan bahwa metode *content-based filtering* dan KNN efektif dalam memberikan rekomendasi laptop yang relevan dan sesuai dengan kebutuhan pengguna. Pengujian sistem menunjukkan akurasi yang tinggi dalam merekomendasikan laptop yang sesuai dengan preferensi spesifik pengguna, sehingga membantu mereka membuat keputusan pembelian yang lebih baik dan efisien.

Kata Kunci: Sistem rekomendasi, Laptop, *Content Based Filtering*, *K-Nearest Neighbor*

ABSTRACT

Recommendation systems play an important role in helping users find products that suit their needs amidst a multitude of choices. This research aims to develop a laptop selection recommendation system using Content-Based Filtering and K-Nearest Neighbors (KNN) methods. This system is designed to provide laptop suggestions based on specifications and price. The dataset used in this research includes important attributes such as RAM, SSD, HDD, operating system, and processor. The research method involves using TF-IDF (Term Frequency-Inverse Document Frequency) to measure the attribute weights of each laptop and cosine similarity to assess the similarity between laptops. The KNN approach is used to find the most similar laptops based on the price attribute selected by the user. The dataset is taken from Kaggle and processed using various Python libraries such as pandas, numpy, and scikit-learn. The research results show that the content-based filtering and KNN methods are effective in providing laptop recommendations that are relevant and in accordance with user needs. Testing of the system shows high accuracy in recommending laptops that match users' specific preferences, helping them make better and more efficient purchasing decisions.

Keywords: Recommendation system, Laptop, *Content Based Filtering*, *K-Nearest Neighbor*

1. PENDAHULUAN

Komputer telah menjadi sebuah inovasi yang memberikan dampak besar bagi kehidupan manusia. Seiring dengan kemajuan teknologi, komputer telah berkembang dari bentuk yang besar dan berat menjadi lebih kecil dan portabel, seperti laptop. Sejak tahun 1980, minat masyarakat terhadap komputer portabel, atau laptop, telah meningkat secara signifikan (Bejarano dkk., 2023; Jalui dkk., 2023). Namun, penting untuk diingat bahwa setiap individu memiliki preferensi yang berbeda dalam memilih laptop (Adamczak dkk., 2021). Laptop telah menjadi salah satu perangkat teknologi yang paling penting dalam kehidupan sehari-hari. Dalam beberapa tahun terakhir, permintaan untuk pembelian laptop telah meningkat secara signifikan. Namun, dengan banyaknya merek dan model yang tersedia di pasaran, memilih laptop yang tepat dapat menjadi tugas yang menantang (Polatidis & Georgiadis, 2016). Oleh karena itu, sistem rekomendasi pemilihan laptop dapat membantu pengguna dalam memilih laptop yang sesuai dengan kebutuhan mereka, seperti spesifikasi pada laptop (Anwar & Uma, 2021).

Perkembangan teknologi yang sangat cepat pada laptop membuat pemahaman konsumen menjadi kurang, baik dari segi kriteria maupun spesifikasi, sehingga konsumen merasa kesulitan dalam memilih laptop. Dengan demikian, saat sudah memilih laptop, konsumen tidak akan merasa kecewa dengan pilihannya. Penelitian sistem rekomendasai ini menawarkan solusi pemilihan merk laptop sesuai dengan keinginan konsumen dengan menggunakan metode *content-based filtering* (Saifudin & Widiyaningtyas, 2024).

Pada tahun 2023, terdapat penelitian sebelumnya implementasi sistem rekomendasi laptop menggunakan metode *content-based filtering* dan *k-means* berbasis mobile dengan hasil penelitian yang baik. Dalam penelitian ini, dengan penggunaan metode *content-based filtering* diketahui hasil perhitungan yang dilakukan oleh *content-based filtering* dengan nilai 14,4231 pada dokumen ke 1. Maka yang direkomendasikan kepada pengguna yaitu dokumen ke 1, dan metode *K-Means* untuk meng-cluster-kan harga yang terdapat pada dataset laptop mendapatkan hasil bahwa dari data tersebut dibuat cluster sebanyak 3 cluster (Ikhsani dkk., 2023).

Oleh karena itu, penelitian ini akan membantu konsumen dengan memberikan rekomendasi dalam memilih laptop dengan metode *Content Based Filtering* dan *K-Nearest Neighbor*. Penelitian ini menggunakan *Content Based Filtering* dan *K-Nearest Neighbor* karena merupakan metode yang simpel dan efektif digunakan untuk data besar.

2. KAJIAN PUSTAKA

A. Sistem Rekomendasi

Sistem rekomendasi adalah suatu perangkat yang dapat membantu pengguna dalam menemukan hal-hal yang mungkin sesuai dengan minatnya. Tujuannya adalah untuk memprediksi produk yang diminati dan memberikan rekomendasi berdasarkan faktor-faktor tertentu. Untuk melakukan hal ini, sistem memproses data besar mengenai preferensi dan minat pengguna. Informasi tersebut digunakan untuk menemukan kesamaan antara pengguna dan item, sehingga dapat memberikan rekomendasi yang sesuai. Sistem rekomendasi memiliki berbagai aplikasi, seperti membantu pengguna menemukan barang yang mereka sukai, membantu penyedia barang mengirimkan produk kepada pelanggan yang tepat, dan lain sebagainya. Klasifikasi sistem rekomendasi dapat dikelompokkan ke dalam beberapa tipe, antara lain: berbasis konten (*content-based*), berbasis kolaboratif (*collaborative-based*), dan berbasis hibrida (*hybrid-based*) (Widiyaningtyas dkk., 2021).

B. *Content Based Filtering*

Metode *Content Based Filtering* adalah sistem yang merekomendasikan berdasarkan konten kepada pengguna berdasarkan profil preferensi pengguna dan hubungan antara deskripsi item, yang relevan dengan item lain yang dibandingkan. Metode ini dapat mengekstrak informasi dari item dan membandingkannya dengan informasi yang ada pada pengguna, seperti item yang pernah dilihat atau disukai, untuk menghasilkan rekomendasi yang sesuai untuk pengguna tersebut. Kelebihan dari metode *Content Based Filtering* termasuk kemampuannya untuk menjelaskan bagaimana rekomendasi dihasilkan dan merekomendasikan item yang mungkin tidak mendapatkan rating dari pengguna lain. *Content Based Filtering* dipilih karena mampu mencocokkan spesifikasi laptop yang dicari oleh pengguna dengan spesifikasi laptop yang tersedia dalam database, kemudian merekomendasikan laptop yang sesuai dengan spesifikasi tersebut (Widiyaningtyas dkk., 2022).

Rumus perhitungan *Content Based Filtering* (Yang dkk., 2020) yang dipergunakan untuk menghitung bobot spesifikasi pada laptop sebagai berikut:

$$TF = \text{Jumlah kemunculan kata } t \text{ pada dokumen } d \quad (1)$$

Keterangan:

TF = *Term Frekuensi*

Mengindikasikan seberapa sering suatu kata atau atribut muncul dalam sebuah dokumen.

$$IDF = \log \frac{\text{Total jumlah dokumen dalam } d}{\text{jumlah dokumen di mana kata } t \text{ muncul}} \quad (2)$$

Keterangan:

- D = jumlah semua dokumen
- DF = jumlah dokumen yang mengandung *term*
- IDF = *Inverse* dokumen *frequency*

Inverse Document Frequency (IDF) (Choudhury dkk., 2021; Isinkaye dkk., 2015) adalah matrik yang digunakan untuk menentukan pentingnya suatu kata (*term*) dalam keseluruhan kumpulan dokumen. IDF digunakan bersama *Term Frequency* (TF) dalam pendekatan TF-IDF untuk mengukur relevansi sebuah kata dalam satu dokumen dibandingkan dengan semua dokumen dalam kumpulan tersebut.

$$W = TF \times (IDF + 1) \quad (3)$$

Keterangan:

- W = bobot atribut
- TF = *Term Frekuensi*
- IDF = *Inverse document frequency*

Menggabungkan nilai *Term Frequency* (TF) (Bichel-Findlay dkk., 2023; Zhou & Han, 2019) dengan *Inverse Document Frequency* (IDF) untuk menghitung seberapa penting suatu atribut pada suatu laptop.

$$\text{Cosine Similarity} = \frac{\text{dot product}(v_1, v_2)}{\|v_1\| \times \|v_2\|} \quad (4)$$

Keterangan:

- v_1 dan v_2 = vektor *TF-IDF* dari dua spesifikasi yang ingin dibandingkan
- $\text{dot product}(v_1, v_2)$ = hasil perkalian dot antara kedua vektor
- $\|v_1\|$ dan $\|v_2\|$ = panjang dari masing-masing vektor

Bobot TF-IDF adalah hasil kali dari *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF).

C. *K-Nearest Neighbor*

Larose Metode *K-Nearest Neighbor* (K-NN) (Liu & Ren, 2023; Zhang dkk., 2023) merupakan sebuah metode yang simpel namun efektif dan efisien dalam mengenali pola, mengklasifikasikan teks, memproses objek, dan bidang lainnya. Keunggulan utamanya terletak pada kemampuan pemrosesan yang sederhana serta kemampuannya untuk menangani data pelatihan yang besar. K-NN menggunakan algoritma pembelajaran terawasi di mana kasus baru diklasifikasikan berdasarkan mayoritas kelas dari K tetangga terdekatnya (Saifudin & Widiyaningtyas, 2024).

Rumus perhitungan *K-Nearest Neighbor* yang dipergunakan untuk mendefinisikan jarak antar data prediksi dan data pada dataset sebagai berikut:

$$\text{Euclidean} = |x - y| \quad (5)$$

Keterangan:

- Euclidean* = jarak terdekat
- x_1 = harga yang ingin dicari
- x_2 = harga yang ada dalam dataset

3. METODE PENELITIAN

A. Identifikasi Masalah

Pada tahapan awal penelitian dilakukan identifikasi masalah, penulis menganalisis permasalahan yang terjadi ketika akan membeli laptop. Permasalahan yang muncul adalah merk laptop yang begitu banyak ada 13 merek laptop dan juga ada spesifikasi laptop yang begitu banyak seperti SSD, HDD, processor, Ram dan lain-lain. Dari masalah yang sering terjadi, disarankan untuk membuat suatu sistem rekomendasi untuk laptop. Hal ini dapat membantu untuk mencari laptop yang sesuai dengan kebutuhan dan kemampuannya dengan lebih mudah.

B. Pengambilan Dataset

Tahapan selanjutnya yaitu pengambilan dataset, data yang digunakan merupakan data sekunder, yang artinya data tersebut sudah tersedia di publik. Pengumpulan data sekunder diambil dari situs kaggle.com Laptop Price (kaggle.com) tahun 2024 dengan jumlah data yaitu 1303 data mengenai laptop yang dapat dilihat pada Tabel 1 berikut ini.

Tabel 1. Pengumpulan dataset

| No | Atribut | Deskripsi | Tipe Data |
|------------|------------------|--|---------------|
| Atribut 1 | laptop_ID | Nomor pada dataset | Int |
| Atribut 2 | Company | Merupakan nama perusahaan atau produsen yang membuat laptop | String |
| Atribut 3 | Product | Merek yang memproduksi dan menjual perangkat tersebut | String |
| Atribut 4 | TypeName | Kategori atau jenis laptop | String |
| Atribut 5 | Inches | Satuan dari besaran panjang pada sistem satuan imperial | Int |
| Atribut 6 | ScreenResolution | Jumlah pixel dalam satu layar, dinyatakan dengan perbandingan antara lebar dan tinggi layar dalam satuan pixel | Int, String |
| Atribut 7 | Cpu | Perangkat keras komputer yang akan mengambil dan mengeksekusi instruksi | String |
| Atribut 8 | Ram | Menunjukkan jumlah RAM (Random Access Memory) yang dimiliki laptop, diukur dalam gigabyte (GB) | Int |
| Atribut 9 | Memory | Penyimpanan pada laptop untuk menyimpan data | Int, String |
| Atribut 10 | Gpu | Merek atau produsen dari kartu grafis (GPU) laptop | String |
| Atribut 11 | OpSys | Sistem operasi yang diinstal pada laptop seperti Windows, MacOS, Linux | String |
| Atribut 12 | Weight | Ukuran berat pada laptop | Float, String |
| Atribut 13 | Price_euro | Harga laptop | Float |

C. Preprocessing

Tahap *preprocessing* merupakan proses di mana data diperiksa dan diperbaiki sebelum diproses oleh algoritma. Pada tahap ini, kolom yang tidak sesuai akan dirubah.

D. Metode Content Based Filtering

Pada tahap proses *Content Based Filtering*, menghitung bobot (W) untuk setiap atribut pada setiap laptop metode yang digunakan *Content-Based Filtering*. Prosesnya melibatkan perhitungan *Term Frequency* (TF), *Inverse Document Frequency* (IDF), dan melakukan perhitungan *cosine similarity* untuk mencari kemiripan setiap atribut, agar hasil rekomendasi lebih mirip dengan *user* input.

E. Metode K-Nearest Neighbor

Tahap proses pada *K-Nearest Neighbor* untuk menemukan k tetangga terdekat dari harga laptop berdasarkan jarak *Euclidean* antara laptop tersebut dan laptop lainnya.

4. HASIL DAN PEMBAHASAN

Data memiliki beberapa kolom yang perlu diubah, masih memiliki format yang belum sesuai dengan kebutuhan sistem seperti kolom *Product*, *TypeName*, yang harus digabungkan dengan kolom *Company* dan kolom *Price Euro* dikonversi menjadi *Price Rupiah*.

A. Proses Metode *Content Based Filtering* dan *K-Nearest Neighbor*

Data yang telah disiapkan terlebih dahulu diproses menggunakan bahasa pemrograman python yang dijalankan pada jupyter notebook untuk diterapkan rekomendasi menggunakan metode *Content Based Filtering* dengan mencari pendekatan dari bobot pada atribut laptop dan *K-Nearest Neighbor* dengan skenario mencari jarak terdekat berdasarkan harga. Hasil rekomendasi pada program python menggunakan metode *Content Based Filtering* yaitu spesifikasi laptop berdasarkan pendekatan bobot dari *user input*, dan hasil rekomendasi pada program python menggunakan metode *K-Nearest Neighbor* yaitu harga laptop berdasarkan jarak *euclidean* dari *user input*.

B. Hasil Rekomendasi *Content Based Filtering*

Hasil rekomendasi dari program python *Content Based Filtering* menggunakan TF-IDF yaitu pendekatan bobot dari *user input* dengan dataset, dan *cosine similarity* untuk mengurutkan hasil rekomendasi berdasarkan kemiripan dari *user input* yang digambarkan pada Gambar 1 berikut.

| Laptop_ID | Company | Model Laptop | Intake | ScreenResolution | Cpu | Ram | Memory | Gpu | Os | Weight | Price_rupiah |
|-----------|---------|--------------|-------------------------|------------------|-------------------|---------------|-----------|-----------------------|-----------------------|--------|--------------|
| 229 | 215 | Lenovo | LenovoThinkPad E15 3340 | 15.6 | 7thGen i7 | 8GB | 256GB SSD | Intel HD Graphics 620 | Windows | 2.2kg | 15.225.264 |
| 87 | 95 | HP | HP250 G5Notebook | 15.6 | Full HD 1920x1080 | Intel Core i3 | 4GB | 256GB SSD | Intel HD Graphics 620 | 1.98kg | 15.302.878 |
| 227 | 222 | Acer | AcerAspire5 A515-51-73W | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Nvidia GeForce G330M | 2kg | 17.312.828 |
| 223 | 228 | Acer | AcerAspire5 A515-51-73W | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Intel HD Graphics 620 | 1.94kg | 14.271.828 |
| 221 | 337 | Lenovo | LenovoThinkPad E15 3340 | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Intel HD Graphics 620 | 2.2kg | 16.225.216 |
| 887 | 202 | Acer | AcerAspire5 A515-51-73W | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Nvidia GeForce G330M | 1.7kg | 16.212.828 |
| 218 | 82 | HP | HP250 G5Notebook | 15.6 | Full HD 1920x1080 | Intel Core i3 | 4GB | 256GB SSD | Intel HD Graphics 620 | 1.95kg | 15.982.760 |
| 219 | 212 | Lenovo | LenovoThinkPad E15 3340 | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Intel HD Graphics 620 | 1.7kg | 17.284.808 |
| 479 | 174 | Lenovo | LenovoThinkPad E15 3340 | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Intel HD Graphics 620 | 1.65kg | 23.255.808 |
| 112 | 115 | Lenovo | LenovoThinkPad E15 3340 | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Intel HD Graphics 620 | 1.57kg | 16.212.828 |

Gambar 1. Hasil rekomendasi menggunakan *Content Based Filtering*

C. Hasil Rekomendasi *K-Nearest Neighbor*

Hasil rekomendasi dari program python *K-Nearest Neighbor* menggunakan jarak *euclidean* yaitu mencari jarak terdekat dari *user input* dengan dataset berdasarkan harga laptop yang digambarkan pada Gambar 2 berikut.

| Laptop_ID | Company | Model Laptop | Intake | ScreenResolution | Cpu | Ram | Memory | Gpu | Os | Weight | Price_rupiah |
|-----------|---------|--------------|-------------------------|------------------|-------------------|---------------|--------|-----------|---------|--------|--------------|
| 227 | 242 | Acer | AcerAspire5 A515-51-73W | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Windows | 2.2kg | 3.954.420 |
| 2 | 3 | HP | HP250 G5Notebook | 15.6 | Full HD 1920x1080 | Intel Core i3 | 4GB | 256GB SSD | Windows | 1.98kg | 3.954.420 |
| 88 | 100 | Acer | AcerAspire5 A515-51-73W | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Windows | 2kg | 3.889.024 |
| 152 | 761 | Acer | AcerAspire5 A515-51-73W | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Windows | 2.45kg | 15.223.648 |
| 62 | 641 | Lenovo | LenovoThinkPad E15 3340 | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Windows | 2.2kg | 15.223.648 |
| 495 | 500 | Acer | AcerAspire5 A515-51-73W | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Windows | 2.2kg | 15.223.648 |
| 1288 | 1228 | HP | HP250 G5Notebook | 15.6 | Full HD 1920x1080 | Intel Core i3 | 4GB | 256GB SSD | Windows | 1.91kg | 15.223.648 |
| 283 | 288 | Lenovo | LenovoThinkPad E15 3340 | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Windows | 2.2kg | 15.223.648 |
| 978 | 900 | HP | HP250 G5Notebook | 15.6 | Full HD 1920x1080 | Intel Core i3 | 4GB | 256GB SSD | Windows | 2.04kg | 15.223.648 |
| 179 | 188 | Acer | AcerAspire5 A515-51-73W | 15.6 | Full HD 1920x1080 | Intel Core i5 | 8GB | 256GB SSD | Windows | 2kg | 15.223.648 |

Gambar 2. Hasil rekomendasi menggunakan *K-Nearest Neighbor*

D. Implementasi Website

Website sistem rekomendasi ini dibuat menggunakan *html*, *css*, *javascript*, dan *bootstrap* untuk tampilan *website*. Selain itu menggunakan *framework* dari python yaitu *Flask*, untuk membuat server dari *website* sistem rekomendasi. Dengan *Flask*, membuat aplikasi web yang interaktif dan responsif dengan mudah, serta menyediakan rekomendasi kepada pengguna berdasarkan *user input* yang telah disediakan.

1) Tampilan Beranda

Menyediakan menu navigasi untuk memudahkan pengguna menemukan fitur sesuai kebutuhan, seperti mencari laptop berdasarkan spesifikasi laptop, fungsi laptop, dan harga laptop.



Gambar 3. Tampilan halaman website

2) Tampilan Website Spesifikasi

Menampilkan formulir pencarian yang memungkinkan konsumen untuk memasukkan kriteria atau kata kunci untuk mencari rekomendasi.



Gambar 4. Tampilan halaman spesifikasi

3) Tampilan Hasil Rekomendasi

Menampilkan hasil pencarian atau rekomendasi berdasarkan kriteria yang dimasukkan pengguna.



Gambar 5. Tampilan hasil rekomendasi berdasarkan spesifikasi

- 4) Tampilan Fungsi Laptop
Menampilkan halaman untuk memilih laptop berdasarkan fungsi laptop.



Gambar 6. Tampilan fungsi Laptop

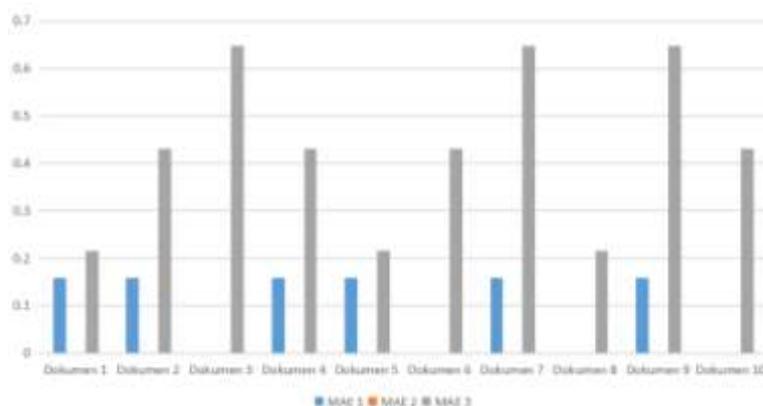
- 5) Tampilan Harga Laptop
Menampilkan halaman untuk memilih laptop berdasarkan harga laptop.



Gambar 7. Tampilan harga laptop

E. Pengujian Hasil Rekomendasi

Pengujian hasil rekomendasi menggunakan MAE (*Mean Absolut Error*) untuk mengukur rata-rata selisih absolut antara nilai-nilai yang diprediksi dan nilai-nilai aktual dalam suatu dataset. Semakin rendah nilai MAE, maka semakin baik model yang digunakan. Program sistem rekomendasi dicoba dengan memasukkan *user input* 3 kali dengan spesifikasi yang berbeda dan menghitung MAE untuk mengetahui seberapa baik model sistem rekomendasi. Pada Gambar 8 diagram di bawah ini hasil MAE pada *user input* 1 dan 2 mendapatkan hasil yang rendah. Hal ini membuktikan bahwa model sistem rekomendasi sudah baik.



Gambar 8. Pengujian hasil rekomendasi

F. Pengujian Menggunakan Akurasi

Nilai akurasi adalah metrik penting dalam evaluasi model pembelajaran mesin, karena menunjukkan seberapa sering model membuat prediksi yang benar dibandingkan dengan keseluruhan prediksi yang dilakukan. Hasil pengujian menggunakan akurasi ditunjukkan pada Tabel 2 berikut ini.

Tabel 2. Pengujian menggunakan akurasi

| Nilai K | Akurasi |
|---------|---------|
| K = 3 | 60% |
| K = 5 | 60% |
| K = 7 | 70% |

5. KESIMPULAN

Berdasarkan penelitian dapat disimpulkan bahwa sistem rekomendasi pemilihan merk laptop menggunakan *Content Based Filtering* dan *K-Nearest Neighbors* yang telah dibangun dapat memberikan rekomendasi kepada konsumen yang akan membeli laptop berdasarkan spesifikasi yang diinputkan dan berdasarkan fungsi yang diinputkan dan juga berdasarkan harga yang diinputkan. Adapun saran yang diberikan untuk penelitian selanjutnya yaitu penelitian ini dapat dijadikan bahan referensi atau dilakukan pengembangan dengan meningkatkan jumlah data atau mengganti data dan mengganti metodenya dengan tujuan mungkin akan mendapat hasil MAE yang lebih tinggi dan mendapat hasil akurasi yang lebih tinggi atau akurat.

6. DAFTAR PUSTAKA

- Adamczak, J., Deldjoo, Y., Moghaddam, F. B., Knees, P., Leyson, G.-P., & Monreal, P. (2021). Session-based Hotel Recommendations Dataset. *ACM Transactions on Intelligent Systems and Technology*, 12(1), 1–20. <https://doi.org/10.1145/3412379>.
- Anwar, T., & Uma, V. (2021). Comparative study of recommender system approaches and movie recommendation using collaborative filtering. *International Journal of System Assurance Engineering and Management*, 12(3), 426–436. <https://doi.org/10.1007/s13198-021-01087-x>.
- Bejarano, A. C., Adams, J. E., McDowell, J., Parkerton, T. F., & Hanson, M. L. (2023). Recommendations for improving the reporting and communication of aquatic toxicity studies for oil spill planning, response, and environmental assessment. In *Aquatic Toxicology* (Vol. 255). Elsevier B.V. <https://doi.org/10.1016/j.aquatox.2022.106391>.
- Bichel-Findlay, J., Koch, S., Mantas, J., Abdul, S. S., Al-Shorbaji, N., Ammenwerth, E., Baum, A., Borycki, E. M., Demiris, G., Hasman, A., Hersh, W., Hovenga, E., Huebner, U. H., Huesing, E. S., Kushniruk, A., Hwa Lee, K., Lehmann, C. U., Lillehaug, S. I., Marin, H. F., ... Wright, G. (2023). Recommendations of the International Medical Informatics Association (IMIA) on Education in Biomedical and Health Informatics: Second Revision. *International Journal of Medical Informatics*, 170. <https://doi.org/10.1016/j.ijmedinf.2022.104908>.
- Choudhury, S. S., Mohanty, S. N., & Jagadev, A. K. (2021). Multimodal trust based recommender system with machine learning approaches for movie recommendation. *International Journal of Information Technology (Singapore)*, 13(2), 475–482. <https://doi.org/10.1007/s41870-020-00553-2>.
- Ikhsani, N., Futri, S., Nuraini, R., & Fathonah, S. (2023). Implementasi Sistem Rekomendasi Laptop Menggunakan Metode Content Based Filtering Dan K-Means Berbasis Mobile. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 7, Issue 2).

- Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16(3), 261–273. <https://doi.org/10.1016/j.eij.2015.06.005>.
- Jalui, M., Kadam, T., Karkera, K., Wani, R., & Pathak, M. (2023). FlixTime: Group Movie Recommendation System. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.4428451>.
- Liu, Z., & Ren, F. (2023). Algorithm Improvement of Movie Recommendation System based on Hybrid Recommendation Algorithm. *Frontiers in Computing and Intelligent Systems*, 3(3), 113–117. <https://doi.org/10.54097/fcis.v3i3.8581>.
- Polatidis, N., & Georgiadis, C. K. (2016). A multi-level collaborative filtering method that improves recommendations. *Expert Systems with Applications*, 48, 100–110. <https://doi.org/https://doi.org/10.1016/j.eswa.2015.11.023>.
- Saifudin, I., & Widiyaningtyas, T. (2024). Systematic Literature Review on Recommender System: Approach, Problem, Evaluation Techniques, Datasets. *IEEE Access*, 12, 19827–19847. <https://doi.org/10.1109/ACCESS.2024.3359274>.
- Widiyaningtyas, T., Ardiansyah, M. I., & Adji, T. B. (2022). Recommendation Algorithm Using SVD and Weight Point Rank (SVD-WPR). *Big Data and Cognitive Computing*, 6(4). <https://doi.org/10.3390/bdcc6040121>.
- Widiyaningtyas, T., Hidayah, I., & Adji, T. B. (2021). Recommendation algorithm using clustering-based upcsim (Cb-upcsim). *Computers*, 10(10), 1–17. <https://doi.org/10.3390/computers10100123>.
- Yang, Y., Jang, H.-J., & Kim, B. (2020). A Hybrid Recommender System for Sequential Recommendation: Combining Similarity Models With Markov Chains. *IEEE Access*, 8, 190136–190146. <https://doi.org/10.1109/ACCESS.2020.3027380>.
- Zhang, Z., Patra, B. G., Yaseen, A., Zhu, J., Sabharwal, R., Roberts, K., Cao, T., & Wu, H. (2023). Scholarly recommendation systems: a literature survey. *Knowledge and Information Systems*, 65(11), 4433–4478. <https://doi.org/10.1007/s10115-023-01901-x>.
- Zhou, W., & Han, W. (2019). Personalized recommendation via user preference matching. *Information Processing and Management*, 56(3), 955–968. <https://doi.org/10.1016/j.ipm.2019.02.002>.