

PENERAPAN METODE HYBRID-BASED RECOMMENDATION PADA SISTEM REKOMENDASI LAPTOP

Nova Noor Kamala Sari ^{a,1,*}, Efrans Christian ^{b,2}, Rizky Alparez Rati ^{c,3}

^{a,b,c} Jurusan Teknik Informatika, Fakultas Teknik, Universitas Palangka Raya, Kampus UPR Tunjung Nyaho Jalan Yos Sudarso, Palangka Raya 73112

¹ novanoorks@it.upr.ac.id*; ²efrans@it.upr.ac.id ; ³rizkyarez@mhs.upr.ac.id

* corresponding author

ARTICLE INFO

Keywords

Hybrid-Based
Recommendation, Content-
Based Filtering, Collaborative
Filtering, TF-IDF, cosine
similarity, Singular Value
Decomposition.

ABSTRACT

Recommendation systems play a crucial role in helping users choose complex and diverse products, such as laptops, which have numerous and varied technical attributes. This research aims to implement a Hybrid-Based Recommendation method that combines Content-Based Filtering (CBF) and Collaborative Filtering (CF). CBF is implemented using TF-IDF Vectorization and cosine similarity to recommend laptops based on technical attribute similarity. Concurrently, CF uses Singular Value Decomposition (SVD) to predict user preferences based on rating history. A Cascade Hybrid strategy is applied by filtering initial candidates from CBF and then re-ranking them using rating predictions from CF. The dataset comprises laptop data and user ratings obtained from Kaggle. Evaluation is performed using the NDCG metric to measure the relevance order of recommendations and MAPE to assess prediction accuracy. The research results indicate that this hybrid system is capable of generating relevant and personalized recommendations, with an NDCG value of 0.9838 and a MAPE value of 27.94%. The study concludes that the integration of CBF and CF through a hybrid approach effectively produces relevant and effective recommendations. For future development, exploring other hybrid methods, parameter optimization, and direct user testing are suggested.

1. Pendahuluan

Dalam era digital saat ini, dengan pertumbuhan besar dalam volume dan ragam produk, sistem rekomendasi menjadi krusial dalam membantu pengguna dalam pengambilan keputusan. Laptop adalah produk yang memiliki sejumlah atribut kompleks seperti spesifikasi teknis, merek, harga, dan preferensi pengguna, sehingga proses pemilihannya dapat rumit. Sistem rekomendasi yang andal dapat mempermudah pengguna menavigasi beragam opsi dan membuat keputusan yang tepat. Pendekatan umum sistem rekomendasi adalah Content-Based Filtering, Collaborative Filtering, dan Hybrid Filtering.

Penelitian terdahulu oleh Detri Amelia Chandra tahun 2021 menggunakan metode Item Based Collaborative Filtering. Teknik Collaborative Filtering dipilih karena teknik ini mampu mengeksplorasi asosiasi implisit, yaitu asosiasi yang dihasilkan dari riwayat pilihan pengguna. Item based Collaborative Filtering merupakan metode rekomendasi yang didasari atas adanya kesamaan antara pemberian rating terhadap suatu produk dengan produk yang dibeli [4]. Metode rekomendasi tunggal, Content-Based Filtering atau Collaborative Filtering, memiliki keterbatasan dalam memberikan rekomendasi yang optimal. Content-Based Filtering tidak dapat menghasilkan rekomendasi yang beragam karena bergantung pada atribut produk yang telah ditentukan. Di sisi lain, Collaborative Filtering, cenderung menghasilkan rekomendasi yang kurang akurat untuk produk yang baru atau untuk pengguna yang memiliki preferensi unik

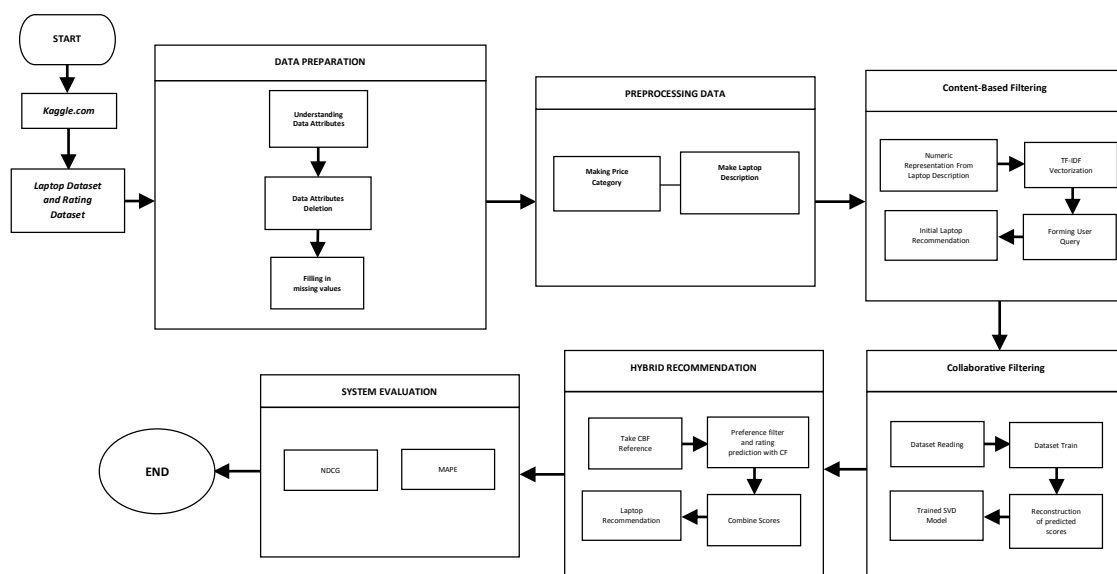
Metode *Hybrid-Based Recommendation* menggabungkan kekuatan berbagai teknik rekomendasi untuk mengatasi keterbatasan ini, sehingga dapat meningkatkan akurasi dan relevansi rekomendasi [12]. Sistem rekomendasi *hybrid* ini sedang marak dikembangkan untuk menggabungkan kekuatan model dan algoritma yang berbeda. Penerapan metode *Hybrid-Based Recommendation* pada sistem rekomendasi laptop diharapkan dapat memberikan rekomendasi yang lebih akurat dan sesuai dengan kebutuhan pengguna.

Penelitian ini berfokus pada penerapan metode *Hybrid-Based Recommendation* dengan penggabungan *Content-Based Filtering* dan *Collaborative Filtering* menggunakan bahasa pemrograman Python, dengan hasil akhir berupa pemodelan sistem. Teknik yang digunakan adalah *Cascade Hybrid Filtering*, yang akan memberikan rekomendasi laptop teratas yang relevan berdasarkan merk, spesifikasi prosesor (CPU dan VGA), dan harga laptop.

Tujuan penelitian ini adalah membuat model sistem rekomendasi laptop yang dapat memberikan rekomendasi produk berdasarkan kriteria relevan dan mengimplementasikan metode *Hybrid-Based Recommendation* dalam sistem rekomendasi laptop. Manfaat penelitian adalah menguji dan mendapatkan hasil penerapan metode *Hybrid-Based Recommendation* serta menyediakan sistem rekomendasi laptop dengan relevansi yang baik melalui gabungan dua metode.

2. Metodologi Penelitian

Proses dan tahapan dari penelitian ini meliputi beberapa tahap, mulai dari tahap awal yaitu melakukan pengumpulan data yang akan digunakan, lalu tahap selanjutnya data yang telah dikumpulkan akan dilakukan penyiapan data atau *data preparation* yaitu tahap pembersihan dataset yang tidak digunakan dalam penelitian. Tahap selanjutnya adalah preprocessing, dimana data akan di proses sebelum nantinya data tersebut digunakan ke dua metode pemodelan, yaitu metode *Content-Based Filtering* dan *Collaborative Filtering*. Tahap selanjutnya akan dilakukan penggabungan prediksi dari kedua metode sebelumnya menjadi *Hybrid-Based Recommendation* menggunakan *Cascade Hybrid*. Tahap setelahnya akan dilakukan analisis dan evaluasi metrik terkait hasil dari pemodelan dan penerapan metode.



Gambar 1. Alur Penelitian

2.1. Metode *Content Based Filtering*

Content-Based Filtering (CBF) merupakan salah satu teknik rekomendasi yang populer digunakan untuk merekomendasikan *item* kepada pengguna berdasarkan kesamaan konten antara *item* yang direkomendasikan dan *item* yang disukai pengguna sebelumnya [2]. Teknik ini telah banyak diterapkan dalam berbagai aplikasi, seperti rekomendasi film, musik, buku, dan berbagai produk.

Content-Based Filtering (CBF) bekerja dengan menganalisis fitur-fitur dari *item* dan pengguna. Fitur-fitur ini dapat berupa kata kunci, genre, *rating*, dan lain sebagainya. Algoritma *Content-Based Filtering* kemudian akan menggunakan fitur-fitur ini untuk menghitung tingkat kesamaan antara *item* dan pengguna [2]. *Item* dengan tingkat kesamaan yang tinggi dengan *item* yang disukai pengguna sebelumnya akan direkomendasikan kepada pengguna [2].

Perhitungan menggunakan algoritma *TF-IDF* dan *cosine similarity*. *TF-IDF* merupakan metode statistik yang digunakan untuk mengekstrak fitur teks dengan menimbang pentingnya suatu kata dalam dokumen relatif terhadap korpus [5]. *Cosine Similarity* adalah metode yang digunakan dalam sistem rekomendasi untuk mengukur kemiripan antara dua *item* atau pengguna.

Rumus *TF-IDF* untuk *term t* dalam dokumen *d*:

$$TF-IDF(t, d) = TF(t, d) \times IDF(t) \quad (1)$$

TF (*Term Frequency*):

$$TF(t, d) = \frac{\text{Jumlah kemunculan } t \text{ di } d}{\text{Total term di } d} \quad (2)$$

IDF (*Inverse Document Frequency*):

$$IDF(t) = \log \left(\frac{N}{df(t)} \right) \quad (3)$$

Keterangan:

(*N* = total dokumen, *df(t)* = Jumlah dokumen yang mengandung *term t*)

$$\text{cosine_similarity}(A, B) = \frac{A \cdot B}{\|A\| \|B\|} \quad (4)$$

Keterangan:

A adalah vektor *TF-IDF* untuk preferensi pengguna

B adalah vektor *TF-IDF* untuk deskripsi laptop

2.2. Metode *Collaborative Filtering*

Collaborative Filtering (CF) merupakan salah satu metode yang paling umum digunakan dalam sistem rekomendasi. CF bekerja dengan menganalisis perilaku pengguna sebelumnya untuk merekomendasikan *item* yang kemungkinan akan disukai oleh pengguna tersebut [4]. *Collaborative Filtering* (CF) bekerja dengan membangun model yang mewakili preferensi pengguna dan *item*. Model ini kemudian digunakan untuk memprediksi preferensi pengguna terhadap *item* yang belum pernah mereka lihat sebelumnya [4]. Prediksi ini kemudian digunakan untuk merekomendasikan *item* kepada pengguna [4]. Perhitungan menggunakan algoritma SVD (*Singular Value Decomposition*).

Singular Value Decomposition, atau disingkat SVD, merupakan salah satu metode dalam aljabar linier yang memiliki banyak manfaat dalam bidang pengolahan data dan kecerdasan buatan. Teknik ini memungkinkan sebuah matriks diuraikan menjadi tiga bagian utama yang saling berhubungan dan memuat informasi penting mengenai struktur awal matriks tersebut [3].

$$R \approx U \Sigma V^T \quad (5)$$

Keterangan

U adalah matriks ortonormal berukuran $m \times m$

Σ adalah matriks diagonal berukuran $m \times n$ yang memuat nilai-nilai singular
 V^T merupakan transpose dari matriks ortonormal V berukuran $n \times n$

2.3. Evaluasi Sistem

2.3.1. NDCG

Normalized Discounted Cumulative Gain adalah pengukur kualitas pemeringkatan yang menangkap kinerja algoritma pemeringkatan dalam sistem rekomendasi. Metrik ini mengevaluasi seberapa baik peringkat *item* yang diprediksi sesuai dengan peringkat ideal, dengan mempertimbangkan relevansi setiap item [3].

$$DCG_p = \sum_{i=1}^p \frac{2^{rel_i-1}}{\log_2(i+1)} \quad (6)$$

$$IDCG_p = \sum_{i=1}^p \frac{2^{rel_i-1}}{\log_2(i+1)} \quad (7)$$

Keterangan:

rel_i adalah relevansi (rating) untuk item ke- i .

i adalah urutan item dalam daftar hasil rekomendasi

2.3.2. MAPE

Mean Absolute Percentage Error merupakan ukuran statistik yang digunakan untuk menilai akurasi model peramalan. *MAPE* merupakan rata-rata dari persentase kesalahan absolut untuk setiap titik data, dengan mengabaikan arah kesalahan [3].

$$MAPE = \left(\frac{100\%}{n}\right) \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (8)$$

Keterangan:

n merupakan jumlah observasi

y_i merupakan nilai actual

$\hat{y}_i$ merupakan nilai ramalan.

3. Hasil dan Pembahasan

3.1. Data Preparation

Dataset yang digunakan adalah data laptop yang telah dikumpulkan sebanyak 1303 jenis produk laptop berbeda dengan informasi spesifikasi yang juga berbeda-beda. Tahap data preparation ini termasuk membersihkan data dari *noise* atau *missing values*, melakukan normalisasi jika diperlukan, dan memilih fitur-fitur yang relevan untuk digunakan dalam model rekomendasi.

Tabel 1. Data laptop hasil Preparation

laptop_id	merk	nama_produk	merk_cpu	nama_cpu	merk_vga	nama_vga	harga
1	Asus	Vivobook 15	Intel	Core i5	Intel	UHD 620	8000000
2	Lenovo	ThinkPad X1	Intel	Core i7	NVIDIA	GTX 1650	15000000
3	Asus	ROG Zephyrus	AMD	Ryzen 5	NVIDIA	GTX 1660	18000000
4	Acer	Aspire 5	Intel	Core i5	AMD	Radeon RX	9500000
5	HP	Pavilion 15	Intel	Core i3	NVIDIA	MX250	7000000

Dataset lain yang digunakan adalah dataset rating, yang berisikan informasi nilai atau rating yang diberikan oleh pengguna terhadap produk. Jumlah dari rating yang diberikan oleh user-user ini berjumlah 100.836 secara total, dalam kasus ini produk yang diberi rating adalah laptop. Untuk rentang rating yang diberikan oleh para user ini berskala dari 0,5 – 5,0.

3.2. Data Preprocessing

Tahap ini melakukan berisi dua tahapan dalam pemrosesan dataset, yaitu membuat kategori harga berdasarkan rentang harganya menjadi 3 kategori: murah, sedang, dan mahal. Kategori murah: mulai dari 0 – 4.999.999, kategori sedang: mulai dari 5.000.000 – 14.999.999, lalu kategori mahal: mulai dari 15.000.000 ke atas. Kemudian tahap kedua adalah membuat kolom deskripsi laptop. Kolom deskripsi dihasilkan dengan menggabungkan beberapa kolom teks, seperti merk, nama produk, merk CPU, nama CPU, merk VGA, dan nama VGA, menjadi satu string. Tujuan dari penggabungan ini adalah untuk memudahkan representasi konten saat melakukan Content-Based Filtering.

3.3. Implementasi Metode Content-Based Filtering (CBF)

Pada tahap ini TF-IDF digunakan untuk mengubah teks pada kolom description yang sebelumnya telah dibuat menjadi representasi numerik (matriks TF-IDF) yang mencerminkan pentingnya kata-kata dalam dokumen relatif terhadap korpus dokumen. Kata-kata yang umum akan diabaikan (stop_words='english') contohnya seperti ('the', 'a', 'is'). Output dari fungsi ini akan mengembalikan tfidf_vectorizer yang sudah dilatih dan tfidf_matrix yang berisi representasi numerik dari semua deskripsi laptop.

```
# --- 3. MODEL CONTENT-BASED FILTERING (TF-IDF) ---
def initialize_tfidf_vectorizer(df_laptop):
    """
    Menginisialisasi dan melatih TfidfVectorizer.
    """
    tfidf_vectorizer = TfidfVectorizer(stop_words='english')
    tfidf_matrix = tfidf_vectorizer.fit_transform(df_laptop['description'])
    print("TfidfVectorizer diinisialisasi dan dilatih.")
    return tfidf_vectorizer, tfidf_matrix
```

Gambar 2. Kode TF-IDF Vectorization

Penerapan metode CBF dilakukan dengan cara memasukkan input dari pengguna secara manual. Pengguna diminta untuk memilih ID, merek laptop yang diinginkan, jenis CPU dan VGA, serta kategori harga yang sesuai dengan kebutuhan mereka.

Berikut ini adalah uji coba program menggunakan kriteria inputan:

Preferensi Pertama

1. User ID : 1
2. Merk Laptop : Asus
3. Merk CPU : Intel
4. Merk VGA : Intel
5. Kategori Harga : Sedang

Tabel 2 Matriks TF-IDF

Kata	TF (Dokumen 1)	IDF Koleksi	TF-IDF (Dokumen 1)
Asus	0.14	0.3010	0.0421
Intel	0.29	0.2218	0.0643
Core	0.14	0.4771	0.0669
i5	0.14	0.4771	0.0669
UHD 620	0.14	0.69897	0.0978

Setelah mendapatkan TF-IDF untuk semua kata dalam dokumen, model dapat melanjutkan menghitung Cosine Similarity antara preferensi pengguna dan deskripsi laptop

Dalam kasus ini, model sudah memiliki vektor TF-IDF dari proses sebelumnya untuk preferensi pengguna dan Laptop 1, yaitu:

1. Preferensi Pengguna: [0.1, 0.3, 0.2, 0.4, 0.5]
2. Laptop 1 (Asus Vivobook 15): [0.14, 0.29, 0.14, 0.14, 0.14]

Sebagai langkah awal dalam menghitung cosine similarity, model akan memperhitungkan dot product terlebih dahulu.

$$A \cdot B = (0.1 \times 0.14) + (0.3 \times 0.29) + (0.2 \times 0.14) + (0.4 \times 0.14) + (0.5 \times 0.14)$$

$$A \cdot B = 0.014 + 0.087 + 0.028 + 0.056 + 0.07 = 0.255$$

Langkah selanjutnya, model akan melakukan perhitungan Norma (magnitude) dari vektor A dan B:

$$\|A\| = (0.1^2 + 0.3^2 + 0.2^2 + 0.4^2 + 0.5^2) = 0.01 + 0.09 + 0.04 + 0.16 + 0.25 = 0.55 \approx 0.742$$

$$\|B\| = (0.14^2 + 0.29^2 + 0.14^2 + 0.14^2 + 0.14^2) = 0.0196 + 0.0841 + 0.0196 + 0.0196 + 0.0196 = 0.1625 \approx 0.403$$

Setelah kedua tahap sebelumnya dilakukan, model akan mulai menghitung cosine similarity antara preferensi dari pengguna dengan Laptop 1. Berikut adalah perhitungannya:

$$\text{cosine_similarity}(A, B) = \frac{0.255}{0.742 \times 0.403} = \frac{0.255}{0.298} \approx 0.855$$

3.4. Collaborative Filtering (CF)

Dataset rating akan dilakukan pelatihan terlebih dahulu menggunakan matrix factorization dengan SVD (Singular Value Decomposition). Model akan menggunakan SVD untuk memprediksi rating pengguna terhadap laptop yang belum diberikan rating. Pelatihan data akan dilakukan dengan membagi data menjadi trainset dan testset. 80% data akan digunakan untuk pelatihan dan 20% data akan digunakan untuk pengujian. Output dari tahap ini akan menghasilkan model SVD yang telah dilatih seperti yang terlihat pada tabel 3.

```
# --- 4. MODEL COLLABORATIVE FILTERING (SVD) ---
def train_svd_model(df_rating):
    """
    Melatih model SVD untuk collaborative filtering.
    Mengembalikan model SVD, trainset, dan testset.
    """
    reader = Reader(rating_scale=(1, 5))
    data_surprise = Dataset.load_from_df(df_rating[['user_id', 'laptop_id', 'rating']], reader)

    # Gunakan train_test_split langsung pada objek Dataset
    trainset_surprise, testset_surprise = train_test_split(data_surprise, test_size=0.2, random_state=42)

    model_svd = SVD(n_factors=50, n_epochs=20, random_state=42)
    model_svd.fit(trainset_surprise)
    print("Model SVD berhasil dilatih.")

    # Mengembalikan model, trainset, dan testset
    return model_svd, trainset_surprise, testset_surprise

print("Model SVD berhasil dilatih pada training set.")
return model_svd, trainset_surprise, testset_surprise # Mengembalikan trainset dan testset juga
```

Gambar 3. Kode Pelatihan Data dengan SVD

Tabel 3. Tabel Matriks Rating Pengguna

User ID	Laptop ID	Rating
1	1	5
1	2	3
2	3	4
2	4	5
3	2	4
3	5	2

3.5. Cascade Hybrid

Setelah mendapatkan hasil dari Content-Based Filtering dan Collaborative Filtering, model akan menggabungkan hasil-hasil tersebut dalam langkah cascade hybrid.

Tabel 4. Data Hasil Gabungan

Laptop ID	Merk	Nama Produk	Cosine Similarity	Prediksi Rating (SVD)
-----------	------	-------------	-------------------	-----------------------

1	Asus	Vivobook 15	0.96	4.8
2	Lenovo	ThinkPad X1	0.65	4.2
3	Asus	ROG Zephyrus	0.50	4.0

Laptop yang memiliki Cosine Similarity tinggi dan prediksi rating tinggi akan ditempatkan lebih tinggi dalam daftar rekomendasi. Dari contoh di atas, Laptop 1 (Asus Vivobook 15) akan berada di urutan pertama, diikuti oleh Laptop 2 (Lenovo ThinkPad X1) dan Laptop 3 (Asus ROG Zephyrus).

3.6. Evaluasi Model

Berdasarkan hasil output dari preferensi sebelumnya, didapatkan 10 buah laptop urutan teratas yang direkomendasikan dan yang paling relevan dengan preferensi pengguna. Preferensi yang digunakan tersebut merekomendasikan laptop yang memiliki relevansi paling mendekati inputan dan difilter menurut laptop-laptop yang mungkin juga disukai oleh user dengan ID 1

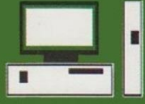
Sistem yang dikembangkan mampu menghasilkan daftar 10 rekomendasi laptop teratas berdasarkan input preferensi pengguna. Contoh hasil rekomendasi untuk seorang pengguna dengan kriteria spesifik disajikan pada Gambar 4. Hasil ini menunjukkan perpaduan antara relevansi spesifikasi (dari CBF) dan preferensi personal (dari CF).

Gambar 4. Uji Coba dan Output Rekomendasi

Evaluasi performa peringkat rekomendasi diukur menggunakan NDCG@10, yang berfokus pada seberapa baik item yang paling relevan ditempatkan di posisi teratas daftar rekomendasi. Hasil yang diperoleh menunjukkan nilai NDCG@10 sebesar 0,968 seperti yang terlihat pada tabel 5. Nilai ini mendekati 1, mengindikasikan bahwa sistem rekomendasi memiliki kemampuan yang sangat baik dalam menyajikan laptop yang relevan bagi pengguna di peringkat teratas dari daftar rekomendasi.

Tabel 5. Pengujian *Normalized Discounted Cumulative Gain*

Posisi	Rating	Discount Factor $\log_2(i + 1)$	DCG (untuk item i)	Rating Ideal	Discount Factor $\log_2(i + 1)$	Ideal DCG (untuk item i)
1	5	1.0000	$\frac{31}{1} = 31$	5	1.0000	31
2	5	1.3010	$\frac{31}{1.3010} \approx 23.85$	5	1.3010	23.85
3	4	1.5849	$\frac{15}{1.5849} \approx 9.47$	4	1.5849	9.47
4	3	2.0000	$\frac{7}{2.0000} = 3.5$	4	2.0000	3.5



5	4	2.3219	$\frac{15}{2.3219} \approx 6.46$	3	2.3219	6.46
6	2	2.5849	$\frac{15}{2.5849} \approx 5.81$	3	2.5849	5.81
7	3	2.8074	$\frac{7}{2.8074} \approx 2.49$	3	2.8074	2.49
8	2	3.0000	$\frac{3}{3.0000} = 1.0$	2	3.0000	1.0
9	4	3.1699	$\frac{3}{3.1699} \approx 0.94$	2	3.1699	0.94
10	2	3.3219	$\frac{1}{3.3219} \approx 0.30$	1	3.3219	0.30

Total DCG:

$$DCG = 31 + 23.85 + 9.47 + 3.5 + 6.46 + 5.81 + 2.49 + 1.0 + 0.94 + 0.30 = \mathbf{76.61}$$

IDCG (Ideal DCG):

$$IDCG = 31 + 23.85 + 9.47 + 7.5 + 3.02 + 3.02 + 2.47 + 1.0 + 0.94 + 0.30 = \mathbf{77.87}$$

Menghitung NDCG

$$NDCG = \frac{DCG}{IDCG} = \frac{76.61}{77.87} \approx \mathbf{0.9838}$$

Nilai NDCG berkisar antara 0.0 hingga 1.0, di mana nilai 1.0 menunjukkan rekomendasi yang optimal.

Akurasi prediksi rating model SVD dievaluasi menggunakan MAPE, baik pada level pengguna individu maupun secara keseluruhan.

1. MAPE untuk Pengguna Spesifik (User 1): Untuk pengguna dengan user_id 1, nilai MAPE yang tercatat adalah 15.96% seperti yang terlihat pada tabel 6. Angka ini menunjukkan bahwa rata-rata kesalahan persentase antara rating yang diprediksi oleh model SVD dan rating aktual yang diberikan oleh pengguna 1 adalah sekitar 15.96%. Tingkat kesalahan ini relatif rendah, menunjukkan bahwa model cukup akurat dalam memprediksi preferensi rating untuk pengguna spesifik ini.
2. MAPE Keseluruhan Model SVD: Secara agregat, nilai MAPE keseluruhan untuk model SVD adalah 27.94% seperti yang terlihat pada tabel 7. Nilai ini merepresentasikan rata-rata kesalahan persentase prediksi rating di seluruh dataset. Meskipun sedikit lebih tinggi dibandingkan dengan MAPE untuk pengguna spesifik (User 1), nilai ini masih berada dalam rentang yang dapat diterima untuk model rekomendasi di banyak domain, mengindikasikan kemampuan model untuk membuat prediksi rating yang cukup akurat secara umum. Perbedaan antara MAPE pengguna spesifik dan MAPE keseluruhan dapat menunjukkan variasi dalam pola rating atau jumlah data yang tersedia antar pengguna.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa sistem rekomendasi *hybrid cascade* ini efektif dalam memberikan rekomendasi yang relevan (ditunjukkan oleh NDCG yang tinggi) dan memiliki akurasi prediksi rating yang memadai (ditunjukkan oleh nilai MAPE).

Tabel 6. Pengujian Mean Absolute Percentage Error User-1

No	user_id	laptop_id	rating	predicted_rating	$\left \frac{y_i - \hat{y}_i}{y_i} \right \times 100\%$
1	1	1	4	4.81057254179275	20.264313544818748
2	1	3	4	4.147083947561766	3.677098689044156
3	1	6	4	4.260768264187	6.519206604674999
4	1	47	5	4.587133230514441	8.25733538971118
5	1	356	4	4.793841920928016	19.846048023200403

6	1	70	3	3.846299757555217	28.20999191850723
7	1	101	5	4.101923685595185	17.961526288096294
8	1	110	4	4.908213406940082	22.70533517350204
9	1	151	5	4.416667284012923	11.666654319741543
10	1	157	5	4.262997835751799	14.740043284964024
11	1	163	5	4.557005626160272	8.859887476794555
12	1	216	5	4.210628174738849	15.78743650522302
13	1	223	3	4.393321992175965	46.44406640586549
14	1	231	5	3.776499006302609	24.47001987394782
15	1	235	4	4.371687267370635	9.292181684265866
16	1	260	5	4.9557502944857585	0.8849941102848291
17	1	296	3	4.738572805029736	57.95242683432452
18	1	316	3	4.098547804037554	36.61826013458513
19	1	333	5	4.544629317335404	9.107413653291925
20	1	34	4	4.246901143817978	6.172528595449456
...	...	...	...	...	...
MAPE				15.95681	

Tabel 7. Pengujian *Mean Absolute Percentage Error* keseluruhan

No	user_id	laptop_id	rating	predicted_rating	$\left \frac{y_i - \hat{y}_i}{y_i} \right \times 100\%$
1	1	1	4	4.804182303848426	20.10455759621066
2	1	3	4	4.331314109117252	8.2828527279313
3	1	6	4	4.642358593236194	16.058964830904856
4	1	50	5	4.739075118992552	5.218497620148952
5	1	50	5	4.739075118992552	5.218497620148952
...	...	...	...	...	...
100832	610	1053	4	3.31238490124176	17.190377468955997
100833	610	161	5	4.0831195650826375	18.33760869834725
100834	610	163	5	3.797857342473155	24.042853150536896
100835	610	165	5	3.956425320714284	20.871493585714322
100836	610	182	3	3.5824093886318242	19.413646287727474
MAPE				27,93789	

4. Kesimpulan

Penelitian ini mengembangkan dan mengimplementasikan sistem rekomendasi laptop menggunakan pendekatan Hybrid-Based Recommendation dengan strategi Cascade Hybrid. Sistem ini mengintegrasikan dua metode utama: Content-Based Filtering (CBF) yang memanfaatkan algoritma TF-IDF dan Cosine Similarity, serta Collaborative Filtering (CF) yang diimplementasikan dengan metode Singular Value Decomposition (SVD).

Proses pembangunan sistem dimulai dengan pengumpulan data informasi laptop dan data rating pengguna. Tahap preprocessing meliputi pengkategorian harga laptop, membentuk kolom deskripsi yang merupakan gabungan dari beberapa kolom menjadi 1 string.

Dalam proses rekomendasi, Content-Based Filtering berperan sebagai saringan awal untuk mendapatkan laptop kandidat berdasarkan kriteria preferensi pengguna seperti merk, CPU, VGA, dan kategori harga. Kemudian, dari daftar kandidat ini, *Collaborative Filtering* dengan SVD memprediksi

rating untuk laptop yang belum diulas pengguna dan melakukan perankingan ulang, menghasilkan rekomendasi akhir yang disesuaikan dengan preferensi pengguna.

Berdasarkan hasil pengujian dan evaluasi, sistem rekomendasi ini menunjukkan kinerja yang efektif. Nilai NDCG@10 sebesar 0.968 menunjukkan bahwa sistem sangat baik dalam menyajikan rekomendasi yang relevan di posisi teratas daftar. Sementara itu, MAPE untuk pengguna spesifik (*User 1*) sebesar 15.96% dan MAPE keseluruhan model SVD sebesar 27.94% mengindikasikan bahwa model memiliki akurasi yang memadai dalam memprediksi rating, dengan rata-rata kesalahan persentase yang dapat diterima. Dengan demikian, penerapan metode *Hybrid-Based Recommendation* pada sistem rekomendasi laptop ini sudah berhasil memberikan rekomendasi yang lebih akurat dan sesuai dengan kebutuhan pengguna, mengatasi beberapa keterbatasan metode tunggal

alam kesimpulan tidak boleh ada referensi. Kesimpulan berisi fakta yang didapatkan. Nyatakan kemungkinan aplikasi, implikasi dan spekulasi yang sesuai. Jika diperlukan, berikan saran untuk penelitian selanjutnya.

Daftar Pustaka

- [1] A. Wijaya and D. Alfian, "Sistem Rekomendasi Laptop menggunakan Collaborative Filtering dan Content-Based Filtering," *Jurnal Computech & Bisnis*, vol. 12, no. 1, pp. 11–27, 2018.J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
- [2] G. Parthasarathy and S. Sathiya Devi, "Hybrid Recommendation System Based on Collaborative and Content-Based Filtering," *Cybernetics and Systems*, vol. 54, no. 4, pp. 432–453, 2023.K. Elissa, "Title of paper if known," unpublished.
- [3] Jadon, A., Patil, A. (2025). A Comprehensive Survey of Evaluation Techniques for Recommendation Systems. In: Bairwa, A.K., Tiwari, V., Vishwakarma, S.K., Tuba, M., Ganokratanaa, T. (eds) *Computation of Artificial Intelligence and Machine Learning. ICCAIML 2024. Communications in Computer and Information Science*, vol 2185. Springer, Cham. https://doi.org/10.1007/978-3-031-71484-9_2
- [4] Chandra, D. A., Santosa, F., & Wahyudi, S. (2021). Penerapan Metode Item Base Collaborative Filtering Berbasis Web Pada Recommender System Laptop. *Engineering and Technology International Journal (EATIJ)*, 3(02), 70.
- [5] Daniel Jurafsky and James H. Martin. 2025. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*, 3rd edition. Online manuscript released January 12, 2025.<https://web.stanford.edu/~jurafsky/slp>
- [6] F. Putra Utama, T. Mardiansyah, R. Faurina, and A. Vatesia, "SCIENTIFIC ARTICLES RECOMMENDATION SYSTEM BASED ON USER'S RELATEDNESS USING ITEM-BASED COLLABORATIVE FILTERING METHOD", *J. Tek. Inform. (JUTIF)*, vol. 4, no. 3, pp. 467-475, Jun. 2023.
- [7] Johari, M., & Laksito, A. (2021). The Hybrid Recommender System of the Indonesian Online Market Products using IMDb weight rating and *TF-IDF*. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(5), 977 - 983.
- [8] Lianhuan Li, Zheng Zhang, Shaoda Zhang, "Hybrid Algorithm Based on Content and Collaborative Filtering in Recommendation System Optimization and Simulation", *Scientific Programming*, vol. 2021, Article ID 7427409, 11 pages, 2021.
- [9] Loukili, M., & Messaoudi, F. (2024). Collaborative Singular Value Decomposition with user-item interaction expansion for first-time user and item recommendations. *International Journal of Informatics and Communication Technology*, 14(1), 111. <https://doi.org/10.11591/ijict.v14i1.pp111-121>
- [10] Ni, Jianjun, Yu Cai, Guangyi Tang, and Yingjuan Xie. 2021. "Collaborative Filtering Recommendation Algorithm Based on *TF-IDF* and User Characteristics" *Applied Sciences* 11, no. 20: 9554. <https://doi.org/10.3390/app11209554>

- [11] V. Atina and D. Hartanti, "KNOWLEDGE BASED RECOMMENDATION MODELING FOR CLOTHING PRODUCT SELECTION RECOMMENDATION SYSTEM", *J. Tek. Inform. (JUTIF)*, vol. 3, no. 5, pp. 1407-1413, Oct. 2022.
- [12] Yassine Afoudi, Mohamed Lazaar, Mohammed Al Achhab, *Hybrid* recommendation system combined Content-Based Filtering and collaborative prediction using artificial neural network, *Simulation Modelling Practice and Theory*, Volume 113, 2021, 102375, ISSN 1569-190X
- [13] Santoso, F., Wahyudi, S., & Chandra, D. A. (2025). Sistem Rekomendasi Laptop Menggunakan Metode Collaborative Filtering Dan Weighted Product Pada Toko Online Indojaya Computer. *Seminar Nasional Teknologi & Sains*, 4(1), 594-604.
- [14] Suwandy, N. I. F., Fathonah, R. N. S., & Prianto, C. (2023). *IMPLEMENTASI SISTEM REKOMENDASI LAPTOP MENGGUNAKAN METODE CONTENT BASED FILTERING DAN K-MEANS BERBASIS MOBILE*.
- [15] Ardiansyah, R., Bianto, M. A., & Saputra, B. D. (2023). Sistem Rekomendasi Buku Perpustakaan Sekolah menggunakan Metode Content-Based Filtering. *Jurnal CoSciTech (Computer Science and Information Technology)*, 4(2), 510-518.