

# **SISTEM REKOMENDASI**

## **REKOMENDSI FILM MENGGUNAKAN HYBRID RECOMMENDATION SYSTEM**

Rio Ardana Putra

NIM 1301218694

## Daftar Gambar

Gambar 2.1 Alur diagram sistem rekomendasi secara umum. ....	9
Gambar 2.2 Diagram alur <i>user-based collaborative filtering</i> .....	13
Gambar 2.3 Diagram alur <i>content-based filtering</i> .....	18
Gambar 2.4 Diagram alur <i>hybrid recommendation recommendation system</i> .....	20
Gambar 2.5 Diagram strategi <i>hybrid recommendation</i> .....	21
Gambar 3.1 Alur kerja sistem secara keseluruhan.....	25
Gambar 3.2 Tampilan UI hasil rekomendasi. ....	28
Gambar 3.3 Tampilan UI modal ketika kartu film dipilih. ....	29
Gambar 4.1 Hasil rekomdasi dengan data <i>User-ID</i> 1 dengan <i>user</i> tidak aktif. ....	30
Gambar 4.2 Hasil pengujian menggunakan User-ID 1 beserta <i>filter</i> dengan genre <i>Comedy</i> . .	31
Gambar 4.3 Tampilan UI rekomendasi apabila menampilkan hasil rekomendasi berjumlah 4-6.....	31
Gambar 4.4 Tampilan UI hasil rekomendasi apabila menampilkan lebih dari 7 hasil rekomendasi. ....	31

## Daftar Isi

Daftar Gambar .....	2
BAB 1 Pendahuluan.....	7
1.1 Latar Belakang .....	7
1.2 Rumusan Masalah .....	7
1.3 Tujuan Penelitian.....	7
1.4 Batasan Penelitian .....	7
BAB 2 TINJAUAN PUSTAKA .....	8
2.1 Sistem Rekomendasi .....	8
2.2 Alur Umum Sistem Rekomendasi.....	8
2.3 Klasifikasi Sistem Rekomendasi.....	9
2.4 Content-Based Recommendation .....	9
2.4.1 Collaborative Filtering .....	9
2.4.2 Knowledge-Based Recommendation .....	10
2.4.3 Hybrid Recommendation System.....	10
2.5 Collaborative Filtering.....	10
2.5.1 Prinsip Kerja Collaborative Filtering .....	10
2.5.2 Kelebihan Collaborative Filtering.....	11
2.5.3 Keterbatasan Collaborative Filtering .....	11
2.6 User-Based Collaborative Filtering (UBCF) .....	11
2.6.1 Prinsip Kerja User-Based Collaborative Filtering.....	11
2.6.2 Representasi Data .....	12
2.6.3 Alasan Pemilihan User-Based Collaborative Filtering .....	12
2.6.4 Keterbatasan User-Based Collaborative Filtering.....	12
2.6.5 Diagram Konseptual User-Based Collaborative Filtering.....	12
2.7 Pengukuran Kemiripan.....	13
2.7.1 Konsep Dasar <i>Similarity</i> Antar Pengguna .....	13
2.7.2 Pearson Correlation sebagai Metode Similarity.....	13
2.7.3 Rumus Pearson Correlation.....	14
2.7.4 Interpretasi Nilai <i>Similarity</i> .....	14
2.8 Neighbor Selection.....	14
2.8.1 Konsep Neighbor dalam User-Based Collaborative Filtering.....	14
2.8.2 Threshold Similarity .....	15

2.8.3 Konsep Top-K Neighbor .....	15
2.8.4 Peran Neighbor Selection dalam Prediksi Rating .....	15
2.9 Rating Prediction.....	15
2.9.1 Konsep Dasar Prediksi Rating .....	16
2.9.2 Weighted Rating Prediction.....	16
2.9.3 Stabilitas Prediksi dan Jumlah <i>Neighbor</i> .....	16
2.10 Content-Based Filtering.....	17
2.10.1 Prinsip Kerja <i>Content-Based Filtering</i> .....	17
2.10.2 Representasi Konten dan Profil Pengguna .....	17
2.10.3 Proses Rekomendasi pada <i>Content-Based Filtering</i> .....	17
2.10.4 Kelebihan dan Keterbatasan <i>Content-Based Filtering</i> .....	17
2.10.5 Alur Content-Based Filtering .....	18
2.11 Hybrid Recommendation System .....	18
2.11.1 Motivasi Penggunaan <i>Hybrid Recommendation System</i> .....	19
2.11.2 Arsitektur <i>Hybrid Recommendation System</i> .....	19
2.11.3 Integrasi <i>Collaborative Filtering</i> dan <i>Content-Based Filtering</i> ...	19
2.11.4 Alur Konseptual <i>Hybrid Recommendation System</i> .....	20
2.12 Strategi <i>Hybrid Recommendation</i> .....	20
2.12.1 <i>Weighted Hybrid Strategy</i> .....	21
2.12.2 Diagram Strategi <i>Hybrid Recommendation</i> .....	21
2.12.3 Permasalahan <i>Cold Start</i> .....	22
2.12.4 Jenis <i>Cold Start</i> .....	22
2.12.5 Dampak <i>Cold Start</i> pada Sistem Rekomendasi .....	22
2.12.6 Penanganan <i>Cold Start</i> dalam Sistem .....	22
2.13 Filtering dan Optimasi Data .....	22
2.14 Tujuan <i>Filtering</i> dan Optimasi Data .....	23
2.14.1 <i>Filtering</i> Pengguna dan Item .....	23
2.14.2 Penyertaan Pengguna Target .....	23
2.14.3 Dampak Optimasi terhadap Sistem.....	23
BAB 3 Perancangan dan Implementasi Sistem .....	24
3.1 Arsitektur Sistem .....	24
3.2 Alur Kerja Sistem .....	24
3.3 Implementasi dan Integrasi Modul Sistem .....	25

3.3.1	Implementasi Modul Frontend.....	25
3.3.2	Implementasi Modul Backend .....	25
3.3.3	Integrasi API Eksternal.....	26
3.3.4	Alur Integrasi Antar Modul.....	26
3.4	Implementasi Hybrid Recommendation System .....	26
3.4.1	Gambaran Umum .....	26
3.4.2	Strategi <i>Hybrid</i> yang Digunakan.....	26
3.4.3	Alur Implementasi Hybrid Recommendation .....	27
3.4.4	Keunggulan Implementasi <i>Hybrid</i> .....	27
3.5	Penanganan <i>Cold Start</i> .....	27
3.5.1	Deteksi Kondisi <i>Cold Start</i> .....	27
3.5.2	Strategi Penanganan <i>Cold Start</i> .....	27
3.5.3	Dampak terhadap Pengalaman Pengguna .....	27
3.6	Penyajian Hasil Rekomendasi.....	28
3.6.1	Format Output Rekomendasi .....	28
3.6.2	Penyajian Visual pada Antarmuka.....	28
3.6.3	Interaksi Pengguna dengan Hasil Rekomendasi .....	28
3.6.4	Konsistensi dan Keandalan Penyajian .....	29
BAB 4	Pengujian dan Validasi Sistem .....	29
4.1	Gambaran Umum Pengujian .....	29
4.2	Tujuan Pengujian .....	29
4.3	Metode Pengujian .....	30
4.4	Skenario Pengujian Sistem .....	30
4.4.1	Pengujian User ID Valid .....	30
4.4.2	Pengujian User ID Tidak Aktif.....	30
4.4.3	Pengujian Filter Genre .....	30
4.4.4	Pengujian Parameter Top-N .....	31
4.5	Pengujian Penanganan Cold Start .....	32
4.6	Validasi Hasil Rekomendasi .....	32
4.7	Pembahasan Hasil Pengujian .....	32
4.8	Keterbatasan Pengujian .....	32
4.9	Evaluasi Numerik Sistem .....	32
BAB 5	KESIMPULAN DAN SARAN .....	33

5.1	Kesimpulan .....	33
5.2	Saran .....	34

# BAB 1 Pendahuluan

## 1.1 Latar Belakang

Jumlah film yang tersedia pada platform digital terus meningkat setiap tahun. Kondisi ini menyebabkan pengguna menghadapi masalah *information overload*, yaitu kesulitan dalam memilih film yang sesuai dengan preferensi mereka. Tanpa bantuan sistem rekomendasi, pengguna cenderung menghabiskan waktu lebih lama dalam proses pencarian atau bahkan membuat pilihan yang tidak optimal.

Sistem rekomendasi film bertekomendasi menjadi komponen penting dalam berbagai platform digital karena mampu menyaring informasi berdasarkan preferensi pengguna. Salah satu pendekatan yang umum digunakan adalah *Collaborative Filtering*, yang memanfaatkan pola kolektif antar pengguna. Namun, pendekatan ini memiliki keterbatasan, seperti masalah *cold start* dan ketergantungan pada data pengguna lain.

Sebagai solusi, penelitian ini mengembangkan Hybrid Recommendation System yang menggabungkan *Collaborative Filtering* dan *Content-Based Filtering* untuk menghasilkan rekomendasi yang lebih stabil dan robust.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, rumusan masalah dalam penelitian ini adalah:

1. Bagaimana merancang sistem rekomendasi film berbasis *Hybrid Recommendation System*?
2. Bagaimana menggabungkan *User-Based Collaborative Filtering* dan *Content-Based Filtering* dalam satu sistem?
3. Bagaimana sistem menangani kondisi *cold start*?
4. Bagaimana menjaga efisiensi sistem pada dataset berskala besar?

## 1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

- A. Membangun sistem rekomendasi film berbasis *hybrid*.
- B. Mengimplementasikan *User-Based Collaborative Filtering* dan *Content-Based Filtering*.
- C. Menghasilkan rekomendasi film *Top-N* yang relevan bagi pengguna.
- D. Menyediakan sistem yang stabil secara logika tanpa bergantung pada *model machine learning* kompleks.

## 1.4 Batasan Penelitian

Batasan dalam penelitian ini meliputi:

- A. Sistem menggunakan data penilaian (*rating*) pengguna.

- B. Tidak menggunakan evaluasi berbasis MAE atau RMSE.
- C. Tidak membahas machine learning lanjutan.
- D. Fokus pada validasi logika dan alur kerja sistem.

## BAB 2 TINJAUAN PUSTAKA

### 2.1 Sistem Rekomendasi

Sistem rekomendasi merupakan suatu sistem cerdas yang dirancang untuk membantu pengguna dalam menemukan item yang relevan dari sekumpulan besar pilihan yang tersedia. Dalam konteks platform digital modern, sistem rekomendasi berperan penting dalam mengatasi permasalahan *information overload*, yaitu kondisi ketika jumlah informasi yang tersedia melebihi kemampuan pengguna untuk memprosesnya secara efektif.

Secara umum, sistem rekomendasi bekerja dengan memanfaatkan data historis pengguna, seperti riwayat interaksi, preferensi, atau penilaian (*rating*), untuk menghasilkan saran yang bersifat personal. Rekomendasi yang dihasilkan tidak bersifat acak, melainkan berdasarkan pola tertentu yang dapat dijelaskan secara logis dan sistematis.

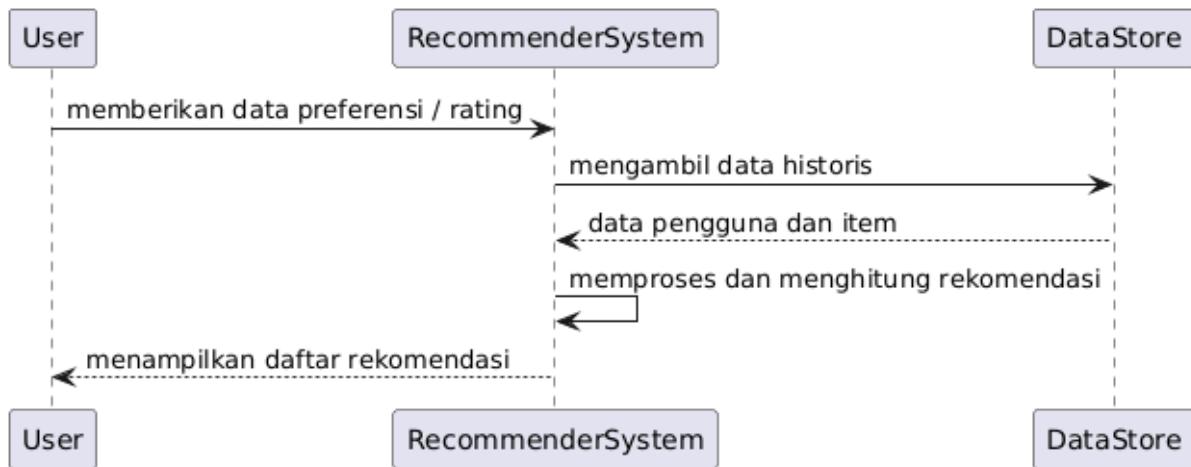
Tujuan utama dari sistem rekomendasi adalah meningkatkan pengalaman pengguna (*user experience*) dengan cara menyajikan konten yang relevan, mengurangi waktu pencarian, serta membantu pengguna dalam pengambilan keputusan. Selain itu, dari sisi penyedia layanan, sistem rekomendasi juga berperan dalam meningkatkan keterlibatan pengguna (*user engagement*) dan pemanfaatan konten.

Dalam implementasinya, sistem rekomendasi banyak digunakan pada berbagai domain, seperti layanan *streaming film*, platform *e-commerce*, layanan musik digital, dan media sosial. Pada domain film, sistem rekomendasi bertugas menyarankan film yang kemungkinan besar sesuai dengan selera pengguna berdasarkan data yang tersedia.

### 2.2 Alur Umum Sistem Rekomendasi

Secara konseptual, alur kerja sistem rekomendasi dapat digambarkan sebagai proses interaksi antara pengguna dan sistem, di mana sistem memanfaatkan data yang ada untuk menghasilkan rekomendasi.

Diagram berikut menggambarkan alur dasar sistem rekomendasi secara umum:



Gambar 2.1 Alur diagram sistem rekomendasi secara umum.

Berdasarkan alur tersebut, dapat dilihat bahwa sistem rekomendasi berfungsi sebagai perantara yang mengolah data pengguna dan data item untuk menghasilkan rekomendasi yang relevan.

## 2.3 Klasifikasi Sistem Rekomendasi

Sistem rekomendasi dapat diklasifikasikan ke dalam beberapa pendekatan utama berdasarkan jenis data yang digunakan dan cara sistem menghasilkan rekomendasi. Klasifikasi ini penting untuk memahami posisi dan karakteristik metode yang digunakan dalam penelitian, serta untuk membandingkan kelebihan dan keterbatasan masing-masing pendekatan.

Secara umum, sistem rekomendasi dapat dibagi menjadi empat kategori utama, yaitu Content-Based Recommendation, Collaborative Filtering, Knowledge-Based Recommendation, dan Hybrid Recommendation System.

## 2.4 Content-Based Recommendation

*Content-Based Recommendation* adalah pendekatan yang merekomendasikan item berdasarkan kesamaan karakteristik konten dengan item yang sebelumnya disukai oleh pengguna. Sistem membangun profil preferensi pengguna dari riwayat interaksi atau rating, kemudian mencocokkannya dengan fitur konten item.

Pendekatan ini tidak bergantung pada data pengguna lain, sehingga tetap dapat bekerja meskipun hanya tersedia satu pengguna. Namun, *Content-Based Recommendation* cenderung menghasilkan rekomendasi yang sempit karena hanya berfokus pada minat yang sudah ada.

### 2.4.1 Collaborative Filtering

*Collaborative Filtering* merupakan pendekatan yang memanfaatkan pola kolektif dari banyak pengguna untuk menghasilkan rekomendasi. Sistem mencari kesamaan preferensi antar pengguna atau antar *item* berdasarkan data *rating*.

Pendekatan ini mampu menemukan pola tersembunyi yang tidak dapat diperoleh hanya dari analisis konten. Akan tetapi, *Collaborative Filtering* memiliki keterbatasan seperti masalah *cold start* dan *sparsity data*.

#### 2.4.2 Knowledge-Based Recommendation

*Knowledge-Based Recommendation* menghasilkan rekomendasi berdasarkan aturan atau pengetahuan eksplisit tentang hubungan antara kebutuhan pengguna dan karakteristik item. Sistem ini tidak bergantung pada data historis *rating*, melainkan pada basis pengetahuan dan *constraint* tertentu.

Pendekatan ini umum digunakan pada domain dengan keputusan bernilai tinggi, seperti rekomendasi properti atau produk kompleks, namun membutuhkan proses perancangan aturan yang relatif kompleks.

#### 2.4.3 Hybrid Recommendation System

Sistem pemberi rekomendasi ini menggabungkan dua atau lebih pendekatan sistem rekomendasi untuk memanfaatkan keunggulan masing-masing metode. Dengan pendekatan *hybrid*, keterbatasan satu metode dapat ditutupi oleh metode lainnya.

Dalam penelitian ini, pendekatan hybrid digunakan untuk mengombinasikan *Collaborative Filtering* dan *Content-Based Filtering*, sehingga sistem mampu menghasilkan rekomendasi yang lebih stabil, adaptif, dan robust terhadap kondisi data yang beragam.

### 2.5 Collaborative Filtering

*Collaborative Filtering* merupakan salah satu pendekatan paling populer dalam sistem rekomendasi yang bekerja dengan memanfaatkan pola perilaku kolektif dari banyak pengguna. Ide dasar dari *Collaborative Filtering* adalah bahwa pengguna yang memiliki preferensi serupa di masa lalu cenderung memiliki preferensi yang serupa pula di masa depan.

Pendekatan ini tidak memerlukan informasi konten dari item yang direkomendasikan, melainkan hanya bergantung pada data interaksi pengguna, seperti rating atau riwayat konsumsi. Oleh karena itu, *Collaborative Filtering* banyak digunakan pada domain di mana informasi konten sulit direpresentasikan secara eksplisit.

#### 2.5.1 Prinsip Kerja Collaborative Filtering

Secara umum, *Collaborative Filtering* bekerja dengan cara menganalisis matriks *user-item* yang berisi nilai *rating* atau interaksi pengguna terhadap item. Dari matriks tersebut, sistem mencari pola kesamaan baik antar pengguna maupun antar item. Alur dasar *Collaborative Filtering* meliputi tahapan berikut:

- 1) Pengumpulan data interaksi pengguna

- 2) Pembentukan matriks *user-item*
- 3) Perhitungan kesamaan (*similarity*)
- 4) Pemilihan entitas serupa
- 5) Prediksi atau pemeringkatan item

### 2.5.2 Kelebihan Collaborative Filtering

*Collaborative Filtering* memiliki beberapa kelebihan utama, antara lain:

- 1) Tidak membutuhkan fitur konten item
- 2) Mampu menangkap pola preferensi kolektif
- 3) Dapat menemukan hubungan tersembunyi antar pengguna atau item

Pendekatan ini sangat efektif ketika jumlah data interaksi pengguna cukup besar dan beragam.

### 2.5.3 Keterbatasan Collaborative Filtering

Meskipun memiliki banyak kelebihan, *Collaborative Filtering* juga memiliki sejumlah keterbatasan, yaitu:

- 1) Masalah *cold start* untuk pengguna atau item baru
- 2) *Data sparsity* pada matriks *user-item*
- 3) Ketergantungan pada kualitas dan jumlah data *rating*

Keterbatasan ini menjadi salah satu alasan utama dikembangkannya sistem rekomendasi *hybrid*.

## 2.6 User-Based Collaborative Filtering (UBCF)

*User-Based Collaborative Filtering* (UBCF) merupakan salah satu pendekatan dalam *Collaborative Filtering* yang menghasilkan rekomendasi dengan cara membandingkan preferensi antar pengguna. Prinsip utama UBCF adalah bahwa pengguna yang memiliki pola penilaian serupa di masa lalu cenderung memiliki preferensi yang serupa pula terhadap item di masa depan.

Dalam pendekatan ini, fokus utama sistem adalah mencari pengguna lain yang memiliki tingkat kemiripan tinggi dengan pengguna target, kemudian memanfaatkan informasi dari pengguna-pengguna tersebut untuk menghasilkan rekomendasi.

### 2.6.1 Prinsip Kerja User-Based Collaborative Filtering

UBCF bekerja dengan menganalisis matriks *user-item* yang berisi nilai *rating* pengguna terhadap item. Setiap baris merepresentasikan seorang pengguna, sedangkan setiap kolom merepresentasikan sebuah item. Langkah-langkah umum dalam UBCF adalah sebagai berikut:

- 1) Membangun matriks *user-item* dari data rating
- 2) Menghitung tingkat kemiripan antar pengguna
- 3) Memilih pengguna serupa (*neighbor*) berdasarkan nilai kemiripan

- 4) Memprediksi *rating item* yang belum dinilai oleh pengguna target
- 5) Menghasilkan daftar rekomendasi berdasarkan hasil prediksi

## 2.6.2 Representasi Data

Data dalam UBCF direpresentasikan dalam bentuk matriks user-item, di mana setiap elemen matriks menunjukkan rating yang diberikan oleh seorang pengguna terhadap suatu item. Nilai kosong pada matriks menunjukkan bahwa pengguna belum berinteraksi dengan item tersebut.

Representasi ini memungkinkan sistem untuk melakukan perhitungan kemiripan antar pengguna secara sistematis.

## 2.6.3 Alasan Pemilihan User-Based Collaborative Filtering

Metode User-Based Collaborative Filtering dipilih dalam penelitian ini karena beberapa pertimbangan berikut:

- 1) Lebih mudah diinterpretasikan dibandingkan *Item-Based Collaborative Filtering*
- 2) Cocok untuk skala akademik dan tujuan pembelajaran
- 3) Setiap rekomendasi dapat ditelusuri kembali ke pengguna serupa
- 4) Tidak memerlukan proses training model yang kompleks

## 2.6.4 Keterbatasan User-Based Collaborative Filtering

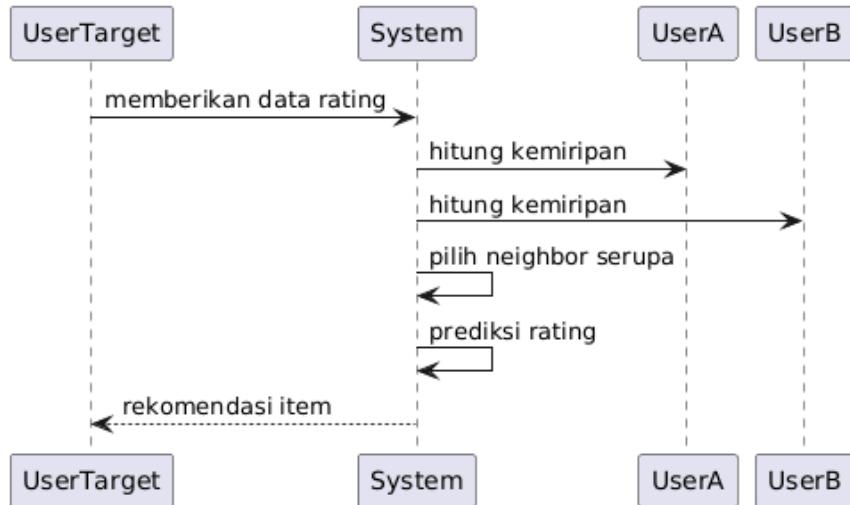
Meskipun memiliki keunggulan, UBCF juga memiliki keterbatasan, antara lain:

- Sensitif terhadap *sparsity data*
- Kurang efektif untuk pengguna baru (*cold start*)
- Kompleksitas meningkat seiring bertambahnya jumlah pengguna

Keterbatasan ini menjadi dasar penggunaan pendekatan *hybrid* pada sistem yang dibangun.

## 2.6.5 Diagram Konseptual User-Based Collaborative Filtering

Diagram berikut menggambarkan alur kerja dasar User-Based Collaborative Filtering. Dengan memahami konsep User-Based Collaborative Filtering, pembahasan selanjutnya akan difokuskan pada metode pengukuran kemiripan antar pengguna yang digunakan dalam sistem ini.



Gambar 2.2 Diagram alur *user-based collaborative filtering*.

## 2.7 Pengukuran Kemiripan

Pengukuran kemiripan (*similarity measurement*) merupakan tahapan penting dalam *User-Based Collaborative Filtering* (UBCF), karena hasil dari pengukuran ini menentukan seberapa besar pengaruh pengguna lain terhadap rekomendasi yang dihasilkan. Similarity digunakan untuk mengukur tingkat kesamaan pola preferensi antara pengguna target dengan pengguna lainnya berdasarkan data rating yang tersedia.

Dalam konteks sistem rekomendasi, similarity bukan sekadar ukuran jarak numerik, tetapi representasi hubungan preferensi antar pengguna. Semakin tinggi nilai *similarity*, semakin besar kemungkinan dua atau lebih pengguna memiliki selera yang serupa.

### 2.7.1 Konsep Dasar *Similarity* Antar Pengguna

*Similarity* antar pengguna dihitung dengan membandingkan rating yang diberikan oleh dua pengguna terhadap item yang sama. Perhitungan *similarity* hanya dilakukan pada item yang dirating oleh kedua pengguna, sehingga perbandingan bersifat adil dan relevan.

Nilai *similarity* umumnya berada dalam rentang tertentu, yang mengindikasikan tingkat kemiripan sebagai berikut:

- 1) Nilai mendekati 1 menunjukkan kemiripan yang sangat tinggi
- 2) Nilai mendekati 0 menunjukkan tidak adanya hubungan yang signifikan
- 3) Nilai mendekati -1 menunjukkan pola preferensi yang berlawanan

### 2.7.2 Pearson Correlation sebagai Metode *Similarity*

Pada penelitian ini, metode Pearson Correlation digunakan untuk menghitung *similarity* antar pengguna. Pearson Correlation dipilih karena mampu mengukur kesamaan pola *rating* dengan memperhitungkan perbedaan skala penilaian antar pengguna.

Berbeda dengan metode lain yang hanya mengukur jarak numerik, Pearson Correlation menormalkan nilai rating dengan rata-rata masing-masing pengguna, sehingga fokus pada pola preferensi relatif.

### 2.7.3 Rumus Pearson Correlation

Pengukuran similarity menggunakan Pearson Correlation dapat dirumuskan sebagai berikut:

$$\text{sim}(u, v) = \frac{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)^2} \times \sqrt{\sum_{i \in I_{u,v}} (r_{v,i} - \bar{r}_v)^2}}$$

Keterangan:

- $r_{u,i}$  adalah rating pengguna  $u$  terhadap item  $i$
- $\bar{r}_u$  adalah rata-rata rating pengguna  $u$
- $I_{u,v}$  adalah himpunan item yang dirating oleh kedua pengguna

### 2.7.4 Interpretasi Nilai *Similarity*

Hasil perhitungan Pearson Correlation menghasilkan nilai antara -1 hingga 1. Dalam sistem ini, nilai similarity digunakan sebagai dasar pemilihan *neighbor* dan sebagai bobot dalam prediksi rating.

Nilai *similarity* yang lebih tinggi menunjukkan bahwa pengguna memiliki pola preferensi yang lebih serupa, sehingga kontribusinya dalam proses rekomendasi menjadi lebih besar.

## 2.8 Neighbor Selection

Neighbor Selection merupakan tahapan lanjutan setelah proses pengukuran kemiripan antar pengguna pada User-Based Collaborative Filtering. Pada tahap ini, sistem tidak menggunakan seluruh pengguna yang telah dihitung nilai similarity-nya, melainkan hanya memilih sejumlah pengguna yang paling relevan untuk dijadikan acuan dalam proses prediksi rating.

Tujuan utama dari Neighbor Selection adalah meningkatkan kualitas rekomendasi dengan memanfaatkan informasi dari pengguna yang benar-benar memiliki pola preferensi serupa, sekaligus menjaga efisiensi komputasi sistem.

### 2.8.1 Konsep Neighbor dalam User-Based Collaborative Filtering

Dalam konteks User-Based Collaborative Filtering, neighbor didefinisikan sebagai pengguna lain yang memiliki tingkat kemiripan tinggi dengan pengguna target berdasarkan nilai similarity yang telah dihitung sebelumnya. Neighbor berperan sebagai sumber informasi utama dalam proses prediksi rating.

Tidak semua pengguna layak dijadikan neighbor, karena pengguna dengan kemiripan rendah atau pola preferensi yang sangat berbeda dapat menurunkan akurasi prediksi.

### 2.8.2 Threshold Similarity

Threshold similarity digunakan sebagai batas minimum nilai kemiripan agar seorang pengguna dapat dipertimbangkan sebagai neighbor. Pengguna dengan nilai similarity di bawah threshold akan diabaikan.

Penggunaan threshold bertujuan untuk:

- Mengurangi noise dari pengguna yang tidak relevan
- Meningkatkan stabilitas prediksi rating
- Menghindari kontribusi negatif dalam perhitungan prediksi

Nilai threshold biasanya ditentukan secara empiris berdasarkan karakteristik dataset.

### 2.8.3 Konsep Top-K Neighbor

Selain threshold similarity, sistem juga menerapkan konsep Top-K neighbor, yaitu memilih sejumlah K pengguna dengan nilai similarity tertinggi dari kumpulan pengguna yang telah melewati threshold.

Pendekatan Top-K memiliki beberapa keuntungan, antara lain:

- Mengontrol jumlah neighbor yang digunakan
- Mengurangi kompleksitas komputasi
- Mencegah dominasi pengguna dengan pengaruh lemah

Pemilihan nilai K yang tepat menjadi faktor penting dalam menjaga keseimbangan antara akurasi dan efisiensi sistem.

### 2.8.4 Peran Neighbor Selection dalam Prediksi Rating

Neighbor yang telah terpilih akan digunakan pada tahap selanjutnya, yaitu prediksi rating. Pada tahap ini, nilai similarity dari masing-masing neighbor digunakan sebagai bobot untuk menentukan seberapa besar kontribusi rating mereka terhadap hasil prediksi.

Dengan demikian, Neighbor Selection berfungsi sebagai penghubung antara pengukuran kemiripan dan proses prediksi rating.

## 2.9 Rating Prediction

Rating Prediction merupakan tahap inti dalam *User-Based Collaborative Filtering* yang bertujuan untuk memperkirakan nilai *rating* suatu item yang belum pernah diberikan oleh pengguna target. Hasil dari tahap ini digunakan sebagai dasar dalam penyusunan daftar rekomendasi *Top-N* item.

Pada sistem yang dibangun, prediksi rating dilakukan menggunakan pendekatan *Weighted Rating Prediction*, yaitu metode rata-rata berbobot yang memanfaatkan *rating* dari *neighbor* terpilih dan nilai *similarity* sebagai bobot kontribusi.

### 2.9.1 Konsep Dasar Prediksi Rating

Prediksi rating dilakukan dengan mengagregasikan rating yang diberikan oleh *neighbor* terhadap item tertentu. Namun, tidak semua rating dianggap memiliki pengaruh yang sama. Rating dari neighbor dengan tingkat kemiripan lebih tinggi akan memberikan kontribusi yang lebih besar dibandingkan *rating* dari neighbor yang kurang mirip.

Pendekatan ini mencerminkan prinsip bahwa preferensi pengguna yang lebih serupa lebih relevan dalam menentukan prediksi rating.

### 2.9.2 Weighted Rating Prediction

Weighted Rating Prediction menghitung nilai prediksi dengan mengalikan rating neighbor dengan nilai *similarity* masing-masing neighbor. Hasil perkalian tersebut kemudian dijumlahkan dan dinormalisasi dengan total bobot *similarity*.

Rumus prediksi rating dapat dituliskan sebagai berikut:

$$\hat{r}_{u,i} = \frac{\sum_{v \in N(u)} sim(u, v) \times r_{v,i}}{\sum_{v \in N(u)} | sim(u, v) |}$$

Keterangan:

- $\hat{r}_{u,i}$ : prediksi rating pengguna  $u$  terhadap item  $i$
- $r_{v,i}$ : rating neighbor  $v$  terhadap item  $i$
- $sim(u, v)$ : nilai *similarity* antara pengguna target dan neighbor
- $N(u)$ : himpunan neighbor terpilih

Normalisasi dilakukan untuk memastikan bahwa hasil prediksi tetap berada dalam rentang skala rating yang valid.

### 2.9.3 Stabilitas Prediksi dan Jumlah *Neighbor*

Penggunaan lebih dari satu *neighbor* dalam proses prediksi membantu meningkatkan stabilitas hasil. Prediksi yang hanya bergantung pada satu *neighbor* cenderung mudah terpengaruh oleh bias individu.

Dengan menggunakan beberapa *neighbor* terpilih, sistem dapat:

- Mengurangi efek *noise* pada data
- Menghasilkan prediksi yang lebih konsisten
- Menyeimbangkan kontribusi preferensi pengguna

Namun, jumlah *neighbor* yang terlalu besar juga dapat menurunkan kualitas prediksi karena menambahkan pengguna yang kurang relevan. Oleh karena itu, penggunaan konsep *Top-K neighbor* menjadi penting dalam tahap sebelumnya.

## 2.10 Content-Based Filtering

*Content-Based Filtering* merupakan pendekatan sistem rekomendasi yang menghasilkan saran item berdasarkan kesesuaian antara karakteristik konten item dengan preferensi pengguna. Berbeda dengan *Collaborative Filtering*, pendekatan ini tidak bergantung pada data pengguna lain, melainkan sepenuhnya berfokus pada profil preferensi pengguna target.

Pada *Content-Based Filtering*, sistem berusaha menjawab pertanyaan: item seperti apa yang kemungkinan besar disukai oleh pengguna berdasarkan riwayat interaksinya sendiri.

### 2.10.1 Prinsip Kerja *Content-Based Filtering*

*Content-Based Filtering* bekerja dengan cara membangun profil preferensi pengguna berdasarkan item yang telah disukai atau diberi *rating* tinggi. Profil ini kemudian dibandingkan dengan fitur konten item lain untuk menentukan tingkat kesesuaian.

Dalam konteks sistem rekomendasi film, fitur konten yang umum digunakan adalah genre film. Setiap film direpresentasikan sebagai sekumpulan atribut genre, sedangkan profil pengguna merepresentasikan tingkat ketertarikan pengguna terhadap masing-masing genre.

### 2.10.2 Representasi Konten dan Profil Pengguna

Konten item direpresentasikan dalam bentuk vektor fitur yang menggambarkan atribut film, seperti genre. Setiap elemen vektor menunjukkan keberadaan atau bobot atribut tertentu.

Profil pengguna dibangun dengan mengagregasikan fitur konten dari film-film yang telah dirating oleh pengguna, dengan mempertimbangkan nilai rating sebagai bobot preferensi. Dengan demikian, profil pengguna mencerminkan kecenderungan minat pengguna terhadap jenis konten tertentu.

### 2.10.3 Proses Rekomendasi pada *Content-Based Filtering*

Setelah profil pengguna terbentuk, sistem menghitung tingkat kesesuaian antara profil pengguna dan item yang belum ditonton. Item dengan tingkat kesesuaian tertinggi akan direkomendasikan kepada pengguna.

Pendekatan ini memungkinkan sistem untuk memberikan rekomendasi yang konsisten dengan minat pengguna, bahkan ketika data pengguna lain tidak tersedia.

### 2.10.4 Kelebihan dan Keterbatasan *Content-Based Filtering*

Kelebihan Content-Based Filtering antara lain:

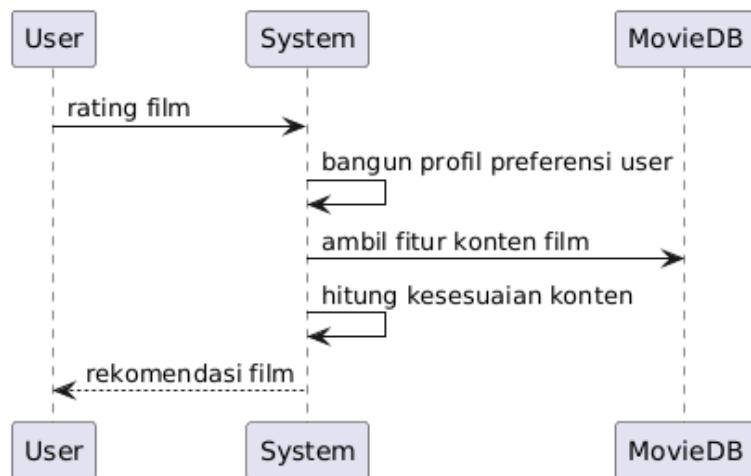
- 1) Tidak memerlukan data pengguna lain
- 2) Cocok untuk menangani cold start pengguna
- 3) Rekomendasi bersifat personal

Namun, pendekatan ini juga memiliki keterbatasan, yaitu:

- 1) Rekomendasi cenderung sempit dan kurang eksploratif
- 2) Sulit menangkap tren kolektif pengguna

#### 2.10.5 Alur Content-Based Filtering

Diagram berikut menggambarkan alur kerja Content-Based Filtering:



Gambar 2.3 Diagram alur content-based filtering..

Gambar 2.3 menggambarkan alur kerja Content-Based Filtering dalam menghasilkan rekomendasi berbasis konten. Proses dimulai dari data interaksi pengguna, seperti rating atau riwayat film yang telah ditonton, yang kemudian digunakan untuk membangun profil preferensi pengguna. Profil ini merepresentasikan tingkat ketertarikan pengguna terhadap fitur konten tertentu, seperti genre film.

Selanjutnya, sistem membandingkan profil preferensi tersebut dengan fitur konten dari film-film yang belum ditonton untuk menghitung tingkat kesesuaian. Film dengan nilai kesesuaian tertinggi kemudian dipilih sebagai rekomendasi. Dengan alur ini, Content-Based Filtering mampu memberikan rekomendasi yang personal dan tetap berfungsi meskipun data pengguna lain tidak tersedia, sehingga berperan sebagai komponen penting dalam sistem hybrid untuk mengatasi keterbatasan Collaborative Filtering.

### 2.11 Hybrid Recommendation System

*Hybrid Recommendation System* merupakan pendekatan sistem rekomendasi yang menggabungkan dua atau lebih metode rekomendasi untuk memanfaatkan keunggulan masing-masing pendekatan sekaligus meminimalkan keterbatasannya. Pendekatan *hybrid* dikembangkan sebagai respons terhadap permasalahan yang muncul pada sistem rekomendasi tunggal, seperti *cold start* pada *Collaborative Filtering* dan keterbatasan eksplorasi pada *Content-Based Filtering*.

Dalam sistem yang dibangun pada penelitian ini, *Hybrid Recommendation System* digunakan untuk mengintegrasikan *User-Based Collaborative Filtering* dan *Content-Based Filtering* sehingga sistem mampu menghasilkan rekomendasi yang lebih stabil, adaptif, dan konsisten secara logika.

### 2.11.1 Motivasi Penggunaan *Hybrid Recommendation System*

Penggunaan pendekatan hybrid didorong oleh fakta bahwa tidak ada satu metode sistem rekomendasi yang mampu menangani seluruh permasalahan secara optimal. Collaborative Filtering unggul dalam menangkap pola kolektif pengguna, namun bergantung pada ketersediaan data pengguna lain. Sebaliknya, *Content-Based Filtering* mampu memberikan rekomendasi personal tanpa ketergantungan pada pengguna lain, tetapi cenderung menghasilkan rekomendasi yang sempit.

Dengan menggabungkan kedua pendekatan tersebut, sistem hybrid dapat:

- Mengurangi dampak cold start
- Meningkatkan kualitas rekomendasi
- Menjaga konsistensi hasil pada berbagai kondisi data

### 2.11.2 Arsitektur *Hybrid Recommendation System*

*Hybrid Recommendation System* bekerja dengan menjalankan lebih dari satu mekanisme rekomendasi dalam satu alur sistem. Setiap mekanisme menghasilkan kandidat rekomendasi atau skor tertentu yang kemudian digabungkan menggunakan strategi hybrid tertentu.

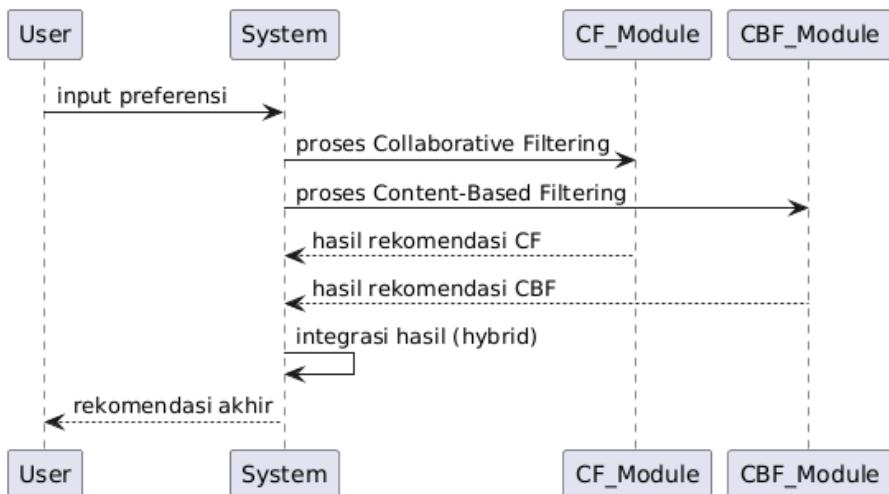
Dalam penelitian ini, hasil rekomendasi dari *Collaborative Filtering* dan *Content-Based Filtering* diproses secara terintegrasi untuk menghasilkan rekomendasi akhir yang ditampilkan kepada pengguna.

### 2.11.3 Integrasi *Collaborative Filtering* dan *Content-Based Filtering*

Integrasi dilakukan dengan cara memanfaatkan keluaran dari masing-masing metode sebagai bagian dari proses pengambilan keputusan rekomendasi. Collaborative Filtering berkontribusi dalam menangkap preferensi kolektif pengguna, sedangkan Content-Based Filtering memastikan bahwa rekomendasi tetap selaras dengan minat individual pengguna.

Pendekatan ini memungkinkan sistem untuk tetap berfungsi dengan baik meskipun salah satu metode mengalami keterbatasan data.

#### 2.11.4 Alur Konseptual *Hybrid Recommendation System*



Gambar 2.4 Diagram alur *hybrid recommendation system*.

Gambar 2.4 menggambarkan konsep dasar Hybrid Recommendation System yang mengintegrasikan dua pendekatan utama, yaitu Collaborative Filtering dan Content-Based Filtering, dalam satu alur sistem. Pada diagram tersebut, permintaan rekomendasi dari pengguna diproses secara paralel oleh kedua modul rekomendasi, di mana Collaborative Filtering berfokus pada pemanfaatan pola preferensi kolektif antar pengguna, sedangkan Content-Based Filtering berfokus pada kesesuaian konten item dengan profil preferensi pengguna. Hasil dari masing-masing modul kemudian dikembalikan ke sistem utama untuk digabungkan pada tahap selanjutnya, sehingga sistem mampu memanfaatkan keunggulan dari kedua pendekatan. Dengan arsitektur ini, Hybrid Recommendation System berperan sebagai fondasi utama yang memungkinkan sistem menghasilkan rekomendasi yang lebih stabil, adaptif, dan konsisten pada berbagai kondisi data.

#### 2.12 Strategi *Hybrid Recommendation*

Strategi *Hybrid Recommendation* menjelaskan cara penggabungan hasil rekomendasi yang dihasilkan oleh lebih dari satu pendekatan sistem rekomendasi. Dalam *Hybrid Recommendation System*, strategi ini berperan penting karena menentukan bagaimana kontribusi masing-masing metode digabungkan untuk menghasilkan rekomendasi akhir.

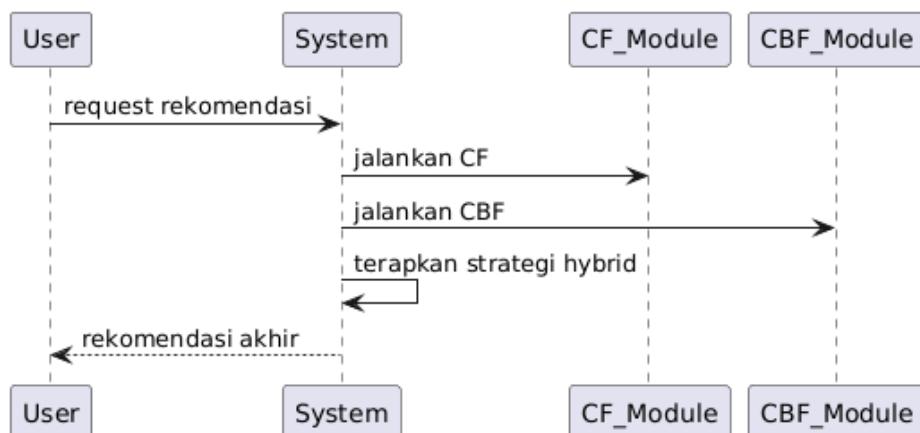
Pada sistem yang dibangun dalam penelitian ini, digunakan beberapa strategi hybrid yang umum dikenal dalam literatur, yaitu *Weighted Hybrid*, *Mixed Hybrid*, dan *Switching Hybrid*. Dari ketiga strategi tersebut, *Weighted Hybrid* digunakan sebagai strategi default.

### 2.12.1 Weighted Hybrid Strategy

*Weighted Hybrid Strategy* menggabungkan hasil rekomendasi dari *Collaborative Filtering* dan *Content-Based Filtering* dengan memberikan bobot tertentu pada masing-masing metode. Setiap metode menghasilkan skor rekomendasi, kemudian skor tersebut digabungkan menggunakan bobot yang telah ditentukan.

Dalam sistem ini, bobot yang digunakan adalah 70% untuk *Collaborative Filtering* dan 30% untuk *Content-Based Filtering*. Pemilihan bobot ini bertujuan untuk memberikan dominasi pada pola kolektif pengguna, namun tetap mempertahankan personalisasi berbasis konten. Pendekatan ini menghasilkan rekomendasi yang lebih seimbang dan stabil pada berbagai kondisi data.

### 2.12.2 Diagram Strategi *Hybrid Recommendation*



Gambar 2.5 Diagram strategi *hybrid recommendation*.

Diagram pada Sub-bab 2.5 menggambarkan alur kerja Hybrid Recommendation System dengan strategi Weighted Hybrid sebagai pendekatan utama yang digunakan dalam penelitian ini. Proses dimulai ketika pengguna mengirimkan permintaan rekomendasi ke sistem, yang kemudian diproses secara paralel oleh dua modul, yaitu Collaborative Filtering dan Content-Based Filtering. Modul Collaborative Filtering menghasilkan skor rekomendasi berdasarkan pola preferensi kolektif pengguna lain yang memiliki tingkat kemiripan tinggi dengan pengguna target, sedangkan modul Content-Based Filtering menghasilkan skor rekomendasi berdasarkan kesesuaian konten item dengan profil preferensi pengguna target.

Hasil rekomendasi dari kedua modul tersebut selanjutnya digabungkan menggunakan pendekatan weighted hybrid, di mana skor dari Collaborative Filtering dan Content-Based Filtering dikalikan dengan bobot yang telah ditentukan, yaitu 70% dan 30%. Proses pembobotan ini bertujuan untuk menyeimbangkan pengaruh pola kolektif dan preferensi individual pengguna, sehingga rekomendasi yang dihasilkan

menjadi lebih stabil, konsisten, dan relevan. Item dengan skor gabungan tertinggi kemudian dipilih dan ditampilkan sebagai rekomendasi akhir kepada pengguna.

### 2.12.3 Permasalahan *Cold Start*

*Cold Start* merupakan salah satu permasalahan utama dalam sistem rekomendasi, khususnya pada pendekatan *Collaborative Filtering*. Permasalahan ini terjadi ketika sistem tidak memiliki data yang cukup untuk menghasilkan rekomendasi yang akurat, baik karena pengguna baru belum memberikan rating (*user cold start*) maupun karena item baru belum memiliki interaksi pengguna (*item cold start*).

Dalam kondisi *cold start*, sistem *Collaborative Filtering* tidak dapat menghitung similarity secara optimal karena keterbatasan data historis. Hal ini dapat menyebabkan sistem gagal memberikan rekomendasi atau menghasilkan rekomendasi yang tidak relevan.

### 2.12.4 Jenis *Cold Start*

*Cold start* dapat diklasifikasikan ke dalam beberapa jenis, antara lain:

- A. **User Cold Start**, yaitu kondisi ketika pengguna baru belum memiliki riwayat interaksi atau rating
- B. **Item Cold Start**, merupakan kondisi ketika item baru belum memiliki rating dari pengguna
- C. **System Cold Start**, yakni kondisi ketika sistem baru pertama kali dijalankan dan data interaksi masih sangat terbatas

### 2.12.5 Dampak *Cold Start* pada Sistem Rekomendasi

Dampak langsung pada kualitas rekomendasi yang dihasilkan. Tanpa data yang memadai, sistem tidak mampu menangkap preferensi pengguna atau pola kolektif, sehingga rekomendasi menjadi tidak personal atau bahkan tidak tersedia. Masalah ini menjadi kelemahan utama *Collaborative Filtering* dan menjadi salah satu alasan dikembangkannya pendekatan *hybrid*.

### 2.12.6 Penanganan *Cold Start* dalam Sistem

Pada sistem yang dibangun pada penelitian ini, *cold start* ditangani dengan memanfaatkan komponen *Content-Based Filtering* dan rekomendasi berbasis popularitas. Ketika *Collaborative Filtering* tidak dapat menemukan *neighbor* yang relevan, sistem secara otomatis beralih untuk merekomendasikan item berdasarkan kesesuaian konten atau popularitas film berdasarkan rating rata-rata.

Pendekatan ini memastikan bahwa sistem tetap mampu memberikan rekomendasi kepada pengguna tanpa menimbulkan kesan kegagalan sistem.

## 2.13 Filtering dan Optimasi Data

Filtering dan optimasi data merupakan tahapan penting dalam pembangunan sistem rekomendasi, terutama ketika sistem beroperasi pada dataset berskala besar.

Tanpa proses filtering yang tepat, sistem rekomendasi dapat mengalami permasalahan kinerja, seperti penggunaan memori yang berlebihan dan waktu komputasi yang tinggi.

Pada sistem yang dibangun dalam penelitian ini, filtering dan optimasi data diterapkan untuk menjaga keseimbangan antara kualitas rekomendasi dan efisiensi komputasi, tanpa mengorbankan relevansi hasil rekomendasi bagi pengguna target.

## 2.14 Tujuan *Filtering* dan Optimasi Data

Tujuan utama dari proses filtering dan optimasi data adalah untuk mengurangi kompleksitas data yang diproses oleh sistem, sehingga sistem dapat berjalan lebih efisien. Selain itu, proses ini juga bertujuan untuk memastikan bahwa data yang digunakan dalam perhitungan benar-benar relevan terhadap proses rekomendasi.

Dengan melakukan filtering yang terkontrol, sistem mampu mempertahankan performa meskipun dataset memiliki jumlah pengguna dan item yang besar.

### 2.14.1 *Filtering* Pengguna dan Item

Filtering dilakukan dengan memilih sejumlah pengguna dan item yang paling aktif dalam dataset. Pengguna yang dianggap aktif adalah pengguna dengan jumlah rating yang relatif tinggi, sedangkan item yang dipilih adalah item yang paling sering dirating.

Pendekatan ini didasarkan pada asumsi bahwa pengguna dan item yang aktif memiliki informasi yang lebih kaya dan stabil untuk digunakan dalam perhitungan similarity dan prediksi rating.

### 2.14.2 Penyertaan Pengguna Target

Meskipun sistem menerapkan filtering terhadap pengguna dan item, pengguna target selalu disertakan dalam proses perhitungan, meskipun tidak termasuk dalam kelompok pengguna paling aktif. Hal ini bertujuan untuk memastikan bahwa sistem tetap dapat menghasilkan rekomendasi yang relevan bagi pengguna target.

### 2.14.3 Dampak Optimasi terhadap Sistem

Diagram Filtering dan Optimasi Data menggambarkan alur penyederhanaan dataset sebelum digunakan dalam proses rekomendasi. Proses dimulai dari dataset mentah yang berisi data pengguna dan item, kemudian sistem melakukan pemilihan pengguna paling aktif dan item paling populer untuk mengurangi kompleksitas data. Meskipun dilakukan filtering, pengguna target tetap disertakan agar sistem dapat memberikan rekomendasi yang personal. Hasil dari proses ini adalah dataset terfilter yang lebih efisien untuk diproses, sehingga sistem rekomendasi dapat berjalan lebih cepat dan hemat memori tanpa mengorbankan kualitas hasil rekomendasi.

# BAB 3 Perancangan dan Implementasi Sistem

## 3.1 Arsitektur Sistem

Arsitektur sistem rekomendasi terdiri dari tiga lapisan utama, yaitu *Frontend*, *Backend*, dan Sumber *Data Eksternal*.

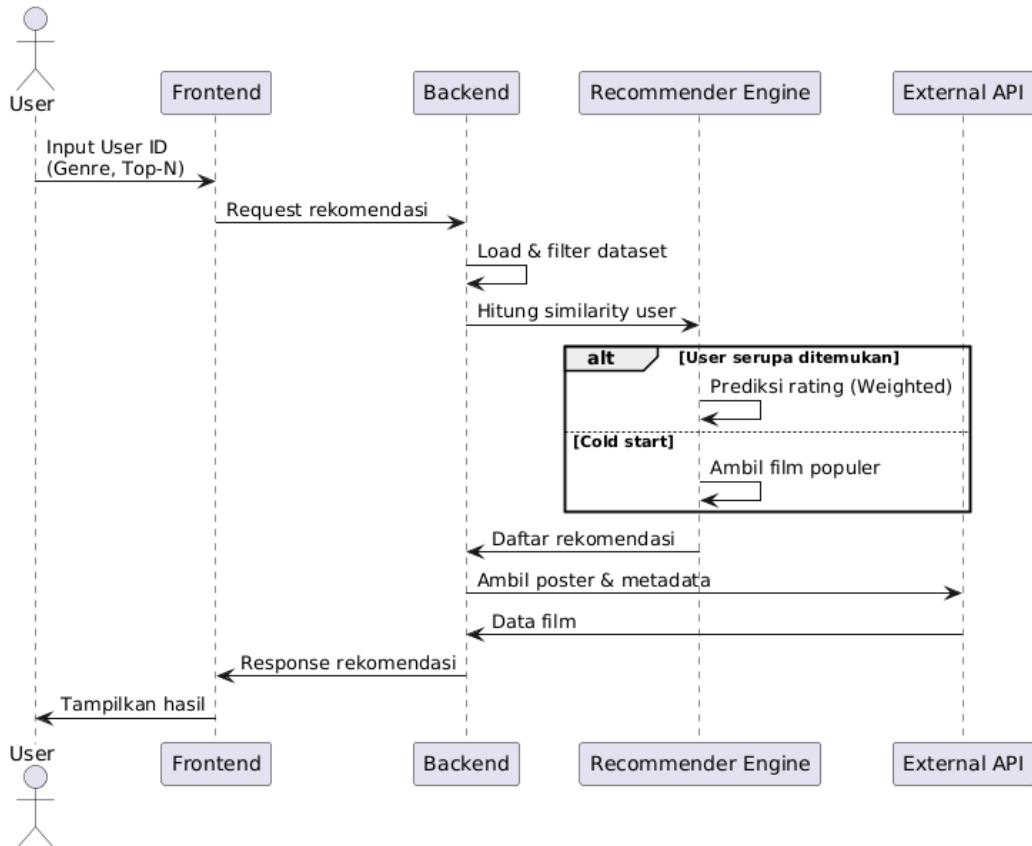
- A. **Frontend**, *Frontend* berfungsi sebagai media interaksi antara pengguna dan sistem. Pengguna dapat memasukkan *User ID*, memilih genre film (opsional), serta menentukan jumlah rekomendasi (*Top-N*). Input ini kemudian dikirim ke backend dalam bentuk request.
- B. **Backend**, *Backend* bertanggung jawab untuk memproses seluruh logika sistem rekomendasi, termasuk:
  - Memuat data rating dan metadata film
  - Melakukan filtering dan optimasi data
  - Menghitung similarity antar pengguna
  - Melakukan prediksi rating
  - Menggabungkan hasil rekomendasi dalam sistem hybrid
- C. **Sumber Data Eksternal**, Sistem memanfaatkan API eksternal (seperti TMDB) untuk memperoleh poster film dan ringkasan deskripsi guna memperkaya tampilan rekomendasi.

## 3.2 Alur Kerja Sistem

Alur kerja sistem dapat dilihat pada Gambar 3.1 menjelaskan alur sistem secara keseluruhan. Proses sistem rekomendasi dimulai ketika pengguna berinteraksi dengan antarmuka frontend. Pengguna memasukkan User ID sebagai identitas utama untuk personalisasi, serta dapat menambahkan parameter opsional berupa genre film dan jumlah rekomendasi (*Top-N*). Input ini kemudian dikirimkan ke backend melalui mekanisme request.

Setelah request diterima, backend memuat dataset rating dan metadata film yang telah melalui tahap filtering dan optimasi. Sistem kemudian melakukan pencarian user serupa menggunakan metode User-Based Collaborative Filtering (UBCF). Jika user serupa ditemukan, proses dilanjutkan dengan prediksi rating menggunakan metode weighted rating prediction. Sebaliknya, jika tidak ditemukan user serupa, sistem akan menjalankan mekanisme cold start handling dengan merekomendasikan film populer.

Hasil rekomendasi yang diperoleh selanjutnya dapat difilter kembali berdasarkan genre (jika parameter genre diberikan). Setelah itu, backend memperkaya hasil rekomendasi dengan metadata tambahan seperti poster dan ringkasan film melalui API eksternal. Seluruh hasil akhir dikirimkan kembali ke frontend dalam format terstruktur untuk ditampilkan kepada pengguna.



Gambar 3.1 Alur kerja sistem secara keseluruhan.

### 3.3 Implementasi dan Integrasi Modul Sistem

#### 3.3.1 Implementasi Modul Frontend

Frontend berfungsi sebagai antarmuka pengguna dengan sistem rekomendasi. Implementasi frontend dilakukan menggunakan HTML, CSS, dan JavaScript, dengan sebuah form utama yang memungkinkan pengguna memasukkan *User ID*, memilih *genre film* (opsional), serta menentukan jumlah rekomendasi (*Top-N*). Data input ini dikirim ke *backend* dalam bentuk request HTTP dan menjadi pemicu utama proses rekomendasi.

#### 3.3.2 Implementasi Modul Backend

Backend merupakan inti dari sistem rekomendasi dan diimplementasikan menggunakan bahasa Python. Modul backend bertanggung jawab dalam memuat dataset rating, membangun matriks user–movie, melakukan filtering dan optimasi data, menghitung similarity antar pengguna menggunakan metode Pearson Correlation, serta melakukan prediksi rating berbasis User-Based Collaborative

Filtering. Selain itu, backend juga mengelola mekanisme *cold start handling* dan penggabungan hasil dalam skema hybrid recommendation.

### 3.3.3 Integrasi API Eksternal

Untuk meningkatkan kualitas penyajian rekomendasi, sistem terintegrasi dengan API eksternal seperti TMDB. API ini digunakan untuk mengambil metadata film berupa poster dan ringkasan deskripsi. Integrasi ini dilakukan setelah daftar rekomendasi terbentuk, sehingga tidak memengaruhi proses perhitungan rekomendasi, melainkan berfungsi sebagai lapisan *enrichment* pada hasil akhir.

### 3.3.4 Alur Integrasi Antar Modul

Seluruh modul dalam sistem saling terhubung melalui alur request-response. Frontend mengirimkan parameter input ke backend, backend memproses rekomendasi berdasarkan logika sistem, lalu hasil rekomendasi yang telah diperkaya metadata dikembalikan ke frontend untuk ditampilkan kepada pengguna. Dengan pendekatan modular ini, sistem menjadi lebih mudah dikembangkan, diuji, dan dipelihara.

## 3.4 Implementasi Hybrid Recommendation System

### 3.4.1 Gambaran Umum

Bab ini menjelaskan implementasi *Hybrid Recommendation System* yang digunakan dalam penelitian ini. Sistem hybrid dibangun dengan mengombinasikan pendekatan User-Based Collaborative Filtering (UBCF) dan Content-Based Filtering (CBF) untuk menghasilkan rekomendasi yang lebih stabil, adaptif, dan relevan terhadap preferensi pengguna.

Pendekatan hybrid dipilih karena masing-masing metode memiliki keterbatasan. Collaborative Filtering bergantung pada keberadaan user serupa, sedangkan Content-Based Filtering hanya mengandalkan riwayat dan preferensi konten pengguna itu sendiri. Dengan menggabungkan keduanya, sistem mampu mengurangi masalah *cold start* dan meningkatkan kualitas rekomendasi.

### 3.4.2 Strategi *Hybrid* yang Digunakan

Strategi hybrid yang diterapkan dalam sistem ini adalah Weighted Hybrid Recommendation. Pada strategi ini, hasil rekomendasi dari Collaborative Filtering dan Content-Based Filtering digabungkan menggunakan bobot tertentu. Secara konseptual, kontribusi masing-masing metode dapat dinyatakan sebagai:

- Collaborative Filtering memiliki bobot lebih besar karena mampu menangkap pola kolektif antar pengguna.
- Content-Based Filtering berfungsi sebagai penyeimbang dan *fallback* ketika data kemiripan pengguna terbatas.

### 3.4.3 Alur Implementasi Hybrid Recommendation

Proses implementasi Hybrid Recommendation System berjalan melalui tahapan berikut:

- A. Sistem menerima input dari pengguna berupa User ID, genre (opsional), dan nilai Top-N.
- B. Collaborative Filtering menghitung similarity antar pengguna dan memprediksi rating film yang belum ditonton.
- C. Content-Based Filtering menghasilkan skor rekomendasi berdasarkan kesesuaian genre dan preferensi pengguna.
- D. Skor dari kedua metode digabungkan menggunakan pendekatan weighted hybrid.
- E. Sistem melakukan pemeringkatan akhir dan menghasilkan daftar Top-N rekomendasi.

### 3.4.4 Keunggulan Implementasi *Hybrid*

Implementasi Hybrid Recommendation System memberikan beberapa keunggulan utama, antara lain:

- Rekomendasi lebih stabil dibandingkan menggunakan satu metode saja.
- Mengurangi ketergantungan pada ketersediaan user serupa.
- Meningkatkan relevansi rekomendasi ketika genre digunakan sebagai filter.
- Memberikan pengalaman pengguna yang lebih konsisten.

## 3.5 Penanganan *Cold Start*

### 3.5.1 Deteksi Kondisi *Cold Start*

Sistem mendeteksi kondisi cold start ketika proses pencarian user serupa tidak menghasilkan neighbor dengan nilai similarity yang memenuhi ambang batas yang telah ditentukan. Dalam kondisi ini, Collaborative Filtering tidak dapat melakukan prediksi rating secara optimal karena keterbatasan data historis.

### 3.5.2 Strategi Penanganan *Cold Start*

Untuk menangani kondisi cold start, sistem menerapkan strategi fallback dengan memanfaatkan rekomendasi berbasis popularitas dan Content-Based Filtering. Rekomendasi berbasis popularitas dihasilkan dengan memilih film-film yang memiliki rating rata-rata tertinggi dan jumlah rating yang signifikan.

### 3.5.3 Dampak terhadap Pengalaman Pengguna

Dengan adanya mekanisme penanganan cold start, sistem rekomendasi tetap mampu memberikan keluaran yang konsisten dan informatif kepada pengguna.

Pengguna tidak dihadapkan pada kondisi tanpa rekomendasi atau pesan error, melainkan tetap menerima saran film yang relevan secara umum.

## 3.6 Penyajian Hasil Rekomendasi

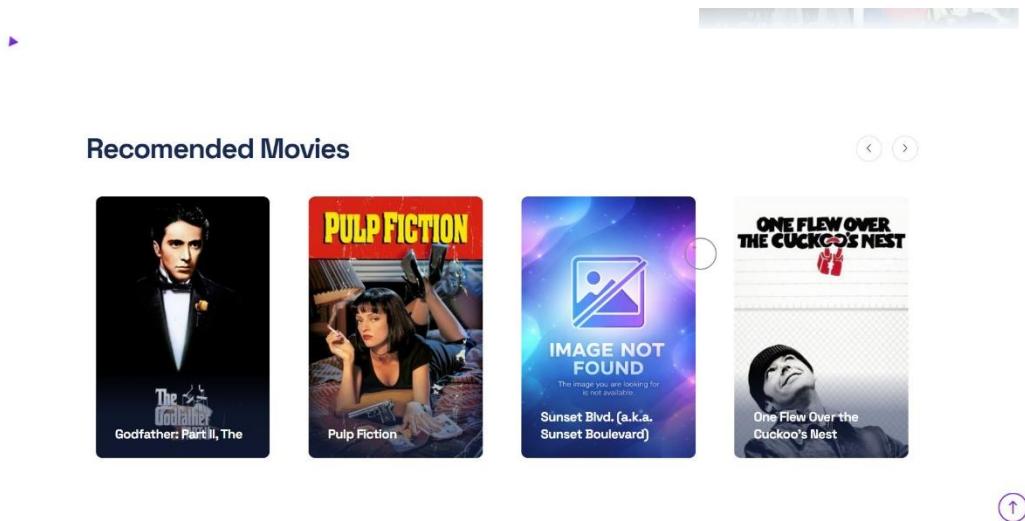
### 3.6.1 Format Output Rekomendasi

Hasil rekomendasi dikirimkan dari backend ke frontend dalam format terstruktur. Setiap item rekomendasi memuat informasi utama seperti judul film, tahun rilis, genre, serta skor prediksi yang dihasilkan oleh sistem.

Struktur output ini memungkinkan frontend untuk menampilkan rekomendasi secara konsisten dan fleksibel, baik dalam jumlah default maupun sesuai parameter Top-N yang ditentukan oleh pengguna.

### 3.6.2 Penyajian Visual pada Antarmuka

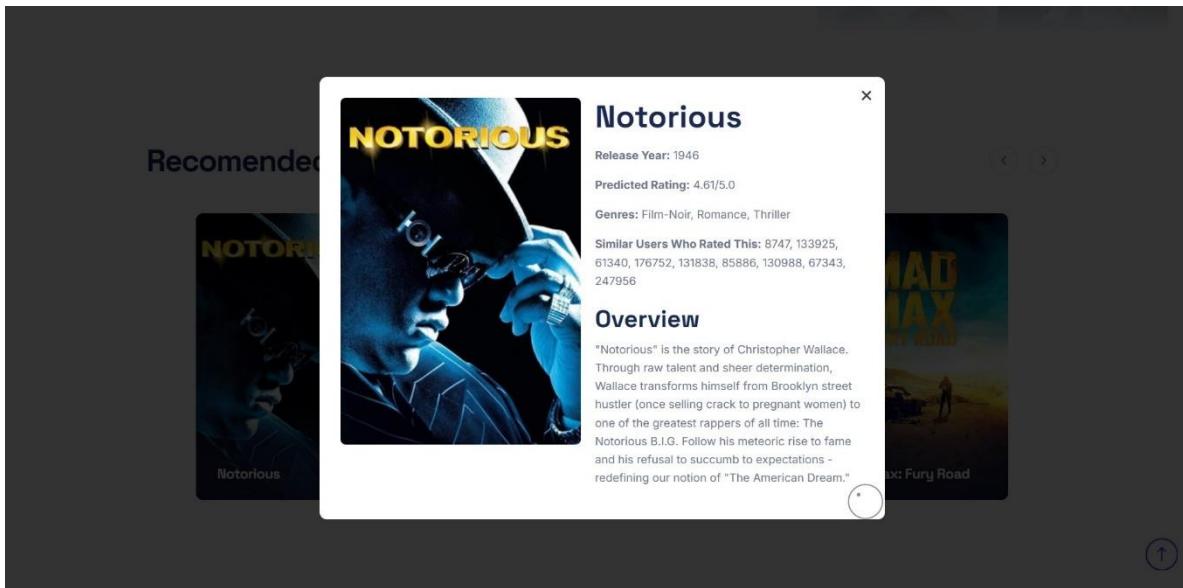
Pada sisi frontend, hasil rekomendasi ditampilkan dalam bentuk kartu. Setiap kartu menampilkan poster film, judul, dan informasi singkat yang relevan. Poster film dan ringkasan deskripsi diperoleh melalui integrasi dengan *API eksternal*. Bila poster film tidak dapat ditemukan maka sistem akan memberikan tampilan “*IMAGE NOT FOUND*” seperti pada Gambar 3.2.



Gambar 3.2 Tampilan UI hasil rekomendasi.

### 3.6.3 Interaksi Pengguna dengan Hasil Rekomendasi

Pengguna dapat berinteraksi dengan hasil rekomendasi, misalnya dengan memilih film tertentu untuk melihat deskripsi lebih lengkap. Interaksi ini tidak memengaruhi perhitungan rekomendasi secara langsung, tetapi memberikan pengalaman eksplorasi yang lebih kaya. Seperti pada Gambar 3.3 informasi yang diberikan disajikan dalam bentuk *pop-up modal* dengan informasi yang didapat dari API beserta dengan prediksi *rating*, dan *user ide* yang memiliki kesukaan yang sama.



Gambar 3.3 Tampilan UI modal ketika kartu film dipilih.

### 3.6.4 Konsistensi dan Keandalan Penyajian

Sistem dirancang untuk menyajikan hasil rekomendasi secara konsisten, baik pada kondisi normal maupun pada kondisi khusus seperti cold start. Dengan demikian, pengguna selalu mendapatkan output yang informatif tanpa mengalami kegagalan tampilan atau ketidakkonsistennan data.

## BAB 4 Pengujian dan Validasi Sistem

### 4.1 Gambaran Umum Pengujian

Bab ini membahas proses pengujian dan validasi terhadap sistem rekomendasi film yang telah dibangun. Pengujian dilakukan untuk memastikan bahwa sistem bekerja sesuai dengan tujuan perancangan, baik dari sisi fungsional maupun logika rekomendasi.

Berbeda dengan penelitian yang berfokus pada optimasi algoritma atau pembandingan model, pengujian pada penelitian ini lebih menekankan pada validasi alur sistem, konsistensi hasil rekomendasi, serta kemampuan sistem dalam menangani berbagai kondisi input pengguna.

### 4.2 Tujuan Pengujian

Tujuan dari pengujian sistem rekomendasi ini adalah sebagai berikut:

- Memastikan setiap modul sistem berjalan sesuai dengan fungsinya
- Memvalidasi alur kerja sistem dari input hingga output
- Menilai konsistensi rekomendasi terhadap variasi input pengguna
- Menguji kemampuan sistem dalam menangani kondisi cold start

## 4.3 Metode Pengujian

Metode pengujian yang digunakan adalah *scenario-based testing*. Pada metode ini, sistem diuji dengan berbagai skenario penggunaan yang mencerminkan kondisi nyata penggunaan sistem.

Pengujian dilakukan melalui antarmuka pengguna dengan memberikan input berbeda, kemudian mengamati respons sistem dan hasil rekomendasi yang dihasilkan.

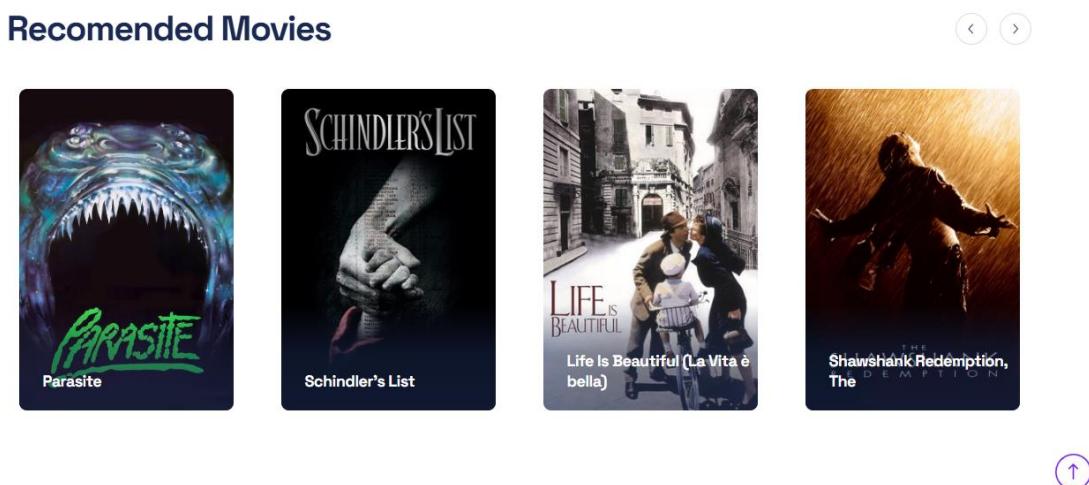
## 4.4 Skenario Pengujian Sistem

### 4.4.1 Pengujian User ID Valid

Pengujian dilakukan dengan memasukkan User ID yang memiliki data rating dalam dataset. Sistem diharapkan mampu menghasilkan daftar rekomendasi film yang relevan dan konsisten dengan preferensi pengguna.

### 4.4.2 Pengujian User ID Tidak Aktif

Pengujian dilakukan dengan User ID yang memiliki jumlah rating terbatas. Sistem diharapkan tetap menghasilkan rekomendasi dengan memanfaatkan mekanisme hybrid dan fallback. Nilai User-ID terkecil agar sistem tidak ada *cold start*.



Gambar 4.1 Hasil rekomendasi dengan data *User-ID* 1 dengan *user* tidak aktif.

yakni 149, namun dengan adanya pendekatan *hybrid* maka dengan User-ID 1 tidak mengalami kendala dalam memberikan rekomendasi film seperti pada Gambar 4.1.

### 4.4.3 Pengujian Filter Genre

Pengujian ini bertujuan untuk memastikan bahwa sistem mampu menyaring hasil rekomendasi berdasarkan genre yang dipilih pengguna tanpa mengganggu proses utama rekomendasi. Gambar 4.2 menunjukkan bahwa selain menangani *cold start* sistem mampun memberikan rekomendasi berdasarkan genre, dalam hal ini pada Gambar 4.1 menunjukkan seluruh genre sedangkan Gambar 4.2 menerapka

*filter* genre *Comedy*. Hal ini dapat dilihat bahwa film Parasite diberikan kembali karena film tersebut memiliki genre *Comedy*.

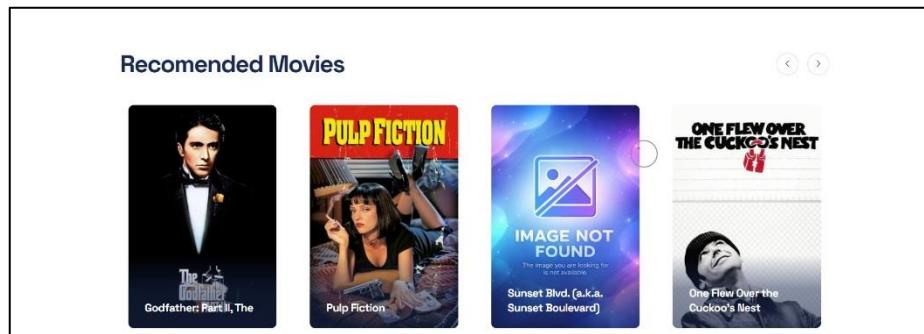
#### Recomended Movies



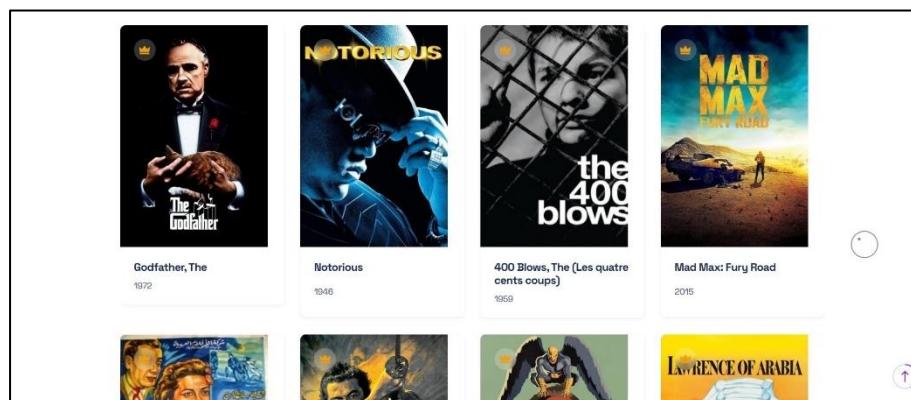
Gambar 4.2 Hasil pengujian menggunakan User-ID 1 beserta *filter* dengan genre *Comedy*.

#### 4.4.4 Pengujian Parameter Top-N

Pengujian dilakukan dengan variasi nilai Top-N untuk memastikan jumlah rekomendasi yang ditampilkan sesuai dengan parameter yang diberikan. Pada sistem minimal jumlah Top-N adalah 4, tampilan ini diberi perbedaan dengan syarat apabila Top-N berjumlah kurang dari 6 maka akan menampilkan slider card seperti pada Gambar 4.3.



Gambar 4.3 Tampilan UI rekomendasi apabila menampilkan hasil rekomendasi berjumlah 4-6.



Gambar 4.4 Tampilan UI hasil rekomendasi apabila menampilkan lebih dari 7 hasil rekomendasi.

Apabila *user* menampilkan lebih dari 6 hasil rekomendasi, maka tampilan akan berbentuk *grid* seperti pada Gambar 4.4 menampilkan bahwa terdapat 4 buah poster film ditampilkan.

## 4.5 Pengujian Penanganan Cold Start

Pengujian ini dilakukan dengan mensimulasikan kondisi di mana sistem tidak menemukan *user* serupa yang memenuhi ambang batas similarity. Dalam kondisi ini, sistem diharapkan dapat memberikan rekomendasi berbasis popularitas atau konten tanpa menghasilkan error. Pada kali ini pengujian mengambil hasil yang sama dengan memasukan User-ID 1 terhadap *form*, hasil dapat dilihat pada Gambar 4.1 menyatakan bahwa *cold start* ditangani dengan baik.

## 4.6 Validasi Hasil Rekomendasi

Validasi hasil rekomendasi dilakukan dengan memperhatikan beberapa aspek berikut:

- Film yang telah dirating oleh pengguna tidak direkomendasikan kembali
- Hasil rekomendasi berubah ketika input pengguna berubah
- Rekomendasi tidak bersifat acak dan memiliki pola yang konsisten

Validasi ini menunjukkan bahwa sistem bekerja berdasarkan logika yang jelas dan dapat dipertanggungjawabkan.

## 4.7 Pembahasan Hasil Pengujian

Berdasarkan hasil pengujian yang dilakukan, sistem rekomendasi mampu menjalankan seluruh skenario pengujian dengan baik. Sistem dapat memproses input pengguna, menghasilkan rekomendasi, serta menangani kondisi khusus seperti cold start secara konsisten. Pendekatan hybrid yang digunakan memberikan fleksibilitas dalam menghasilkan rekomendasi pada berbagai kondisi data.

## 4.8 Keterbatasan Pengujian

Meskipun hasil pengujian menunjukkan kinerja sistem yang memadai, terdapat beberapa keterbatasan dalam proses pengujian, antara lain:

- Pengujian tidak difokuskan pada perbandingan algoritma
- Evaluasi kuantitatif tidak dijadikan fokus utama
- Pengujian dilakukan pada subset data

Keterbatasan ini membuka peluang pengembangan dan penelitian lanjutan.

## 4.9 Evaluasi Numerik Sistem

Sebagai pendukung pengujian fungsional, dilakukan evaluasi numerik untuk memastikan bahwa algoritma rekomendasi menghasilkan prediksi yang konsisten secara kuantitatif. Evaluasi ini tidak dijadikan fokus utama, melainkan sebagai bukti tambahan bahwa sistem bekerja secara rasional.

Hasil evaluasi numerik yang diperoleh adalah sebagai berikut:

- **MAE (Mean Absolute Error)**: 0.6643
- **RMSE (Root Mean Squared Error)**: 0.8688

Nilai MAE dan RMSE yang berada di bawah 1 menunjukkan bahwa rata-rata kesalahan prediksi rating relatif kecil. Hal ini sesuai dengan karakteristik umum User-Based Collaborative Filtering pada dataset rating berskala besar.

Selain itu, metrik pemeringkatan Top-N menunjukkan hasil berikut:

- **Precision@10**: 0.8580
- **Recall@10**: 0.0971
- **F1-score@10**: 0.1706
- **NDCG@10**: 0.8643

Nilai precision dan NDCG yang tinggi menunjukkan bahwa rekomendasi yang dihasilkan memiliki relevansi dan urutan yang baik. Dengan demikian, evaluasi numerik ini memperkuat hasil pengujian fungsional yang telah dilakukan sebelumnya.

## BAB 5 KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Berdasarkan perancangan, implementasi, dan pengujian sistem rekomendasi film yang telah dilakukan, dapat disimpulkan bahwa sistem rekomendasi berbasis **Hybrid Recommendation System** berhasil dibangun dan berfungsi sesuai dengan tujuan penelitian. Sistem mampu menghasilkan rekomendasi film yang relevan dengan memanfaatkan kombinasi **User-Based Collaborative Filtering (UBCF)** dan **Content-Based Filtering (CBF)**.

Pendekatan hybrid yang digunakan memungkinkan sistem untuk mengatasi keterbatasan masing-masing metode tunggal. Collaborative Filtering mampu menangkap pola preferensi kolektif antar pengguna, sedangkan Content-Based Filtering menjaga relevansi rekomendasi berdasarkan karakteristik konten film. Penggabungan kedua pendekatan tersebut menghasilkan rekomendasi yang lebih stabil dan konsisten.

Selain itu, sistem juga mampu menangani kondisi *cold start* dengan baik melalui mekanisme fallback berbasis popularitas dan konten, sehingga pengguna tetap memperoleh rekomendasi meskipun data historis terbatas. Proses filtering dan optimasi data yang diterapkan terbukti membantu menjaga efisiensi sistem tanpa mengorbankan kualitas rekomendasi.

Hasil pengujian fungsional menunjukkan bahwa sistem dapat memproses berbagai skenario input pengguna, sementara evaluasi numerik pendukung

menunjukkan bahwa algoritma yang digunakan menghasilkan prediksi yang rasional dan konsisten secara kuantitatif.

## 5.2 Saran

Meskipun sistem rekomendasi yang dibangun telah berjalan dengan baik, terdapat beberapa saran untuk pengembangan lebih lanjut, antara lain:

- Mengembangkan evaluasi sistem dengan metrik tambahan dan dataset yang lebih besar untuk memperoleh analisis kuantitatif yang lebih mendalam.
- Mengintegrasikan metode Collaborative Filtering berbasis item atau pendekatan *model-based* untuk meningkatkan fleksibilitas sistem.
- Menambahkan fitur *explainable recommendation* agar pengguna dapat memahami alasan di balik rekomendasi yang diberikan.
- Mengoptimalkan performa sistem untuk mendukung penggunaan pada skala data yang lebih besar dan lingkungan produksi.