

Sistem Rekomendasi Film Menggunakan Collaborative Filtering

Fitrah Gisma Ripan¹, Doni Nasution², Muhammad Raffi³, Zikri Maulana⁴, Perani Rosyani⁵

¹²³⁴⁵Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

Email: ¹fitrahgisma@gmail.com, ²doniinidoni@gmail.com, ³raffimuhammad12354@gmail.com,
⁴zikrimlnaa0309@gmail.com, ⁵dosen00837@unpam.ac.id*

(* : coresponding author)

Abstrak—Perkembangan layanan digital yang menyediakan ribuan film menimbulkan tantangan bagi pengguna dalam menemukan tontonan yang sesuai dengan preferensi pribadi. Penelitian ini membangun Sistem Rekomendasi Film menggunakan metode Collaborative Filtering dengan memanfaatkan data rating pengguna untuk memprediksi film yang berpotensi disukai. Tahapan penelitian meliputi pemuatan dataset, pembersihan data, analisis eksploratif, pembentukan user-item matrix, perhitungan cosine similarity, serta pembangunan dua model rekomendasi yaitu User-Based dan Item-Based Collaborative Filtering. Selain itu, fitur pencarian judul film diterapkan menggunakan metode *string matching* dengan library *difflib*. Hasil analisis menunjukkan bahwa cosine similarity mampu mengukur kemiripan antar pengguna maupun antar film secara efektif, sehingga rekomendasi yang dihasilkan lebih relevan dan personal. Visualisasi heatmap juga membantu dalam memahami pola kemiripan antar film berdasarkan nilai similarity. Sistem ini diharapkan menjadi solusi yang efisien bagi pengguna dalam menemukan film sesuai preferensi tanpa harus melakukan pencarian manual.

Kata Kunci: Sistem rekomendasi; Collaborative filtering; Cosine similarity; User-item matrix; Rekomendasi Film

Abstract—The development of digital services that provide thousands of films poses challenges for users in finding shows that suit their personal preferences. This study builds a Movie Recommendation System using the Collaborative Filtering method by utilizing user rating data to predict potentially liked films. The research stages include dataset loading, data cleaning, exploratory analysis, user-item matrix formation, cosine similarity calculation, and the development of two recommendation models: User-Based and Item-Based Collaborative Filtering. In addition, the movie title search feature is implemented using the string matching method with the difflib library. The analysis results show that cosine similarity is able to measure similarities between users and between movies effectively, so that the resulting recommendations are more relevant and personalized. Heatmap visualization also helps in understanding similarity patterns between films based on similarity values. This system is expected to be an efficient solution for users in finding movies according to their preferences without having to do manual searches.

Keywords: Recommendation system; Collaborative filtering; Cosine similarity; User-item matrix; Movie recommendation

1. PENDAHULUAN

Perkembangan teknologi informasi yang semakin pesat telah membawa perubahan signifikan dalam cara masyarakat mengakses dan memilih hiburan, khususnya dalam dunia perfilman. Saat ini, layanan streaming seperti Netflix, Disney+, dan platform digital lainnya telah menjadi pilihan utama masyarakat untuk menikmati film secara praktis dan cepat. Namun, di balik kemudahan tersebut, muncul tantangan baru, yaitu sulitnya memilih film yang sesuai dengan minat pengguna karena jumlah film yang tersedia semakin banyak dan beragam. Kondisi ini sering membuat pengguna kebingungan dalam menentukan pilihan tontonan yang tepat.

Di sisi lain, pengguna memiliki preferensi yang berbeda-beda berdasarkan pengalaman menonton sebelumnya, genre favorit, hingga sutradara tertentu. Tanpa adanya sistem yang mampu memberikan rekomendasi secara personal, proses pencarian film yang relevan menjadi kurang efisien dan memakan waktu. Oleh karena itu, dibutuhkan suatu sistem cerdas yang mampu memberikan saran film secara otomatis berdasarkan pola dan kebiasaan pengguna.

Melihat kebutuhan tersebut, penerapan Sistem Rekomendasi Film Menggunakan Collaborative Filtering menjadi solusi yang sangat relevan. Collaborative Filtering merupakan salah satu teknik machine learning yang bekerja dengan menganalisis kesamaan perilaku antar pengguna atau antar film. Metode ini memanfaatkan data rating pengguna untuk memprediksi film yang

mungkin disukai oleh pengguna lain dengan preferensi serupa. Dengan demikian, setiap pengguna dapat memperoleh rekomendasi yang bersifat personal, akurat, dan sesuai dengan selera mereka.

Dalam penelitian atau pengembangan sistem ini, data yang digunakan berasal dari kumpulan rating film yang diberikan oleh berbagai pengguna. Data tersebut kemudian diproses menggunakan teknik user-item matrix, dimana setiap pengguna dan film direpresentasikan dalam bentuk tabel rating. Selanjutnya, kesamaan antar pengguna maupun antar film dihitung menggunakan metode cosine similarity untuk menghasilkan rekomendasi yang relevan. Sistem ini juga dilengkapi fitur pencarian judul film, di mana pengguna dapat memasukkan nama film yang kemudian diproses menggunakan string matching untuk menemukan judul yang paling mendekati.

Melalui penerapan sistem rekomendasi ini, diharapkan pengguna dapat lebih mudah dalam menemukan film-film yang sesuai dengan preferensi mereka tanpa harus melakukan pencarian manual yang memakan waktu. Sistem ini juga dapat meningkatkan pengalaman pengguna secara keseluruhan, sekaligus menjadi contoh penerapan kecerdasan buatan yang bermanfaat dalam kehidupan sehari-hari.

Dengan adanya pengembangan Sistem Rekomendasi Film Menggunakan Collaborative Filtering ini, peneliti berharap teknologi ini dapat terus dikembangkan untuk berbagai kebutuhan, seperti penyediaan rekomendasi genre spesifik, sutradara tertentu, atau preferensi lainnya. Sistem ini juga dapat menjadi landasan bagi pengembangan platform hiburan digital yang lebih cerdas, efisien, dan berorientasi pada kepuasan pengguna dalam era digital saat ini.

2. METODE PENELITIAN

2.1 Tahapan Penelitian

Tahapan penelitian pada sistem rekomendasi film ini disusun secara terstruktur sesuai dengan proses dalam implementasi program Python. Alur kerja penelitian adalah sebagai berikut:

- Tahap pertama adalah import library, yaitu mempersiapkan seluruh pustaka Python yang dibutuhkan untuk analisis data, perhitungan cosine similarity, dan visualisasi. Library yang digunakan mencakup pandas, numpy, matplotlib, sklearn, dan difflib.
- Tahap kedua adalah pemuatan dataset, yaitu membaca data rating dan data film dengan menggunakan pandas DataFrame. Data film dan rating dimuat dari berkas ratings.csv dan movies.csv yang menjadi dasar pembentukan model rekomendasi.
- Tahap ketiga adalah pembersihan dan praproses data (preprocessing). Pada tahap ini, dilakukan berbagai langkah pemrosesan seperti konversi tipe data, menghapus data invalid, mengatasi kesalahan merge, menghilangkan duplikasi rating, dan menyelaraskan id film agar proses penggabungan dataset dapat dilakukan dengan benar.
- Tahap keempat adalah analisis data eksploratif (EDA), di mana dilakukan observasi pola dasar seperti film yang paling banyak dirating serta distribusi nilai rating.
- Tahap kelima adalah pembentukan User-Item Matrix, yaitu mengubah data rating menjadi tabel matriks dengan pengguna sebagai baris dan film sebagai kolom. Nilai rating yang kosong diisi dengan nol sehingga matriks dapat digunakan dalam perhitungan similarity.
- Tahap berikutnya adalah perhitungan similarity, menggunakan metode cosine similarity untuk menghitung kemiripan antar pengguna (user similarity) dan antar film (item similarity). Matriks similarity ini menjadi dasar utama dalam menghasilkan rekomendasi.
- Tahap keenam adalah pembangunan model rekomendasi, yaitu membuat dua fungsi utama:
 1. recommend_movies_user() untuk rekomendasi berbasis kemiripan pengguna, dan
 2. recommend_movies_item() untuk rekomendasi berdasarkan kemiripan antar film.
- Tahap ketujuh adalah visualisasi model, yaitu menampilkan heatmap kemiripan antar film untuk melihat pola hubungan dalam matriks similarity.

- Tahap terakhir adalah pembuatan kesimpulan sistem dalam format Markdown sebagai hasil akhir yang menggambarkan kinerja model rekomendasi yang telah dibangun

2.2 Teknik dan Pendekatan Analisis

2.2.1 Import Library

Tahap ini menggunakan beberapa library utama:

- a. pandas untuk manipulasi data
- b. numpy untuk operasi numerik
- c. sklearn untuk perhitungan similarity
- d. matplotlib untuk visualisasi
- e. difflib untuk pencocokan teks (string matching)

Kode :

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics.pairwise import cosine_similarity
from difflib import get_close_matches
```

Gambar 1. Import Library

2.2.2 Pemuatan Dataset

Data ratings dan data movies di-load:

Kode :

```
ratings = pd.read_csv('ratings.csv')
movies = pd.read_csv('movies.csv', low_memory=False)
```

Gambar 2. Load Dataset

2.2.3 Data Preprocessing

Tahap praproses merupakan bagian terpenting karena struktur dataset asli seringkali tidak konsisten.

- a. Mengonversi kolom id dan movieId ke tipe numerik untuk menghindari error saat merge.
- b. Menghapus baris yang tidak valid (NaN).
- c. Memastikan hanya film yang memiliki rating yang disertakan.
- d. Melakukan merge antara data rating dan data judul film.
- e. Menghapus rating ganda dengan menghitung rata-rata rating dari user yang sama pada film yang sama.
- f. Membentuk matriks user-item.

Kode :

```
# --- Pastikan kolom id & movieId bertipe numeric ---
movies['id'] = pd.to_numeric(movies['id'], errors='coerce') # ubah invalid -> NaN
ratings['movieId'] = pd.to_numeric(ratings['movieId'], errors='coerce')

# Drop data yang tidak valid
movies = movies.dropna(subset=['id'])
ratings = ratings.dropna(subset=['movieId'])

# Convert ke integer
movies['id'] = movies['id'].astype(int)
ratings['movieId'] = ratings['movieId'].astype(int)

# Filter movie yang punya rating
valid_ids = ratings['movieId'].unique()
movies = movies[movies['id'].isin(valid_ids)]

# Merge aman
ratings_with_title = ratings.merge(
    movies[['id', 'title']],
    left_on='movieId',
    right_on='id'
)

# Tangani duplikat
ratings_grouped = (
    ratings_with_title
    .groupby(['userId', 'title'])['rating']
    .mean()
    .reset_index()
)

# Pivot User x Movie
user_movie_ratings = ratings_grouped.pivot(
    index='userId',
    columns='title',
    values='rating'
).fillna(0)
```

Gambar 3. Data Preprocessing

2.2.4 Analisis Data Eksploratif(EDA)

EDA dilakukan untuk memahami pola dasar pada dataset, yaitu:

- Menentukan film paling populer berdasarkan jumlah rating.
- Menganalisis distribusi rating film.

Kode:

```
print("\n=== EDA: 10 Film Teratas Berdasarkan Jumlah Rating ===")
print(ratings_with_title['title'].value_counts().head(10))

plt.figure()
ratings_with_title['rating'].hist(bins=10)
plt.title("Distribusi Rating Film")
plt.xlabel("Rating")
plt.ylabel("Jumlah")
plt.show()
```

Gambar 4. EDA

2.2.5 Perhitungan Similarity(Cosine Similarity)

Metode utama yang digunakan adalah cosine similarity, digunakan untuk:

- Similarity antar pengguna (user-user similarity)
- Similarity antar film (item-item similarity)

Kode:

```
# Similarity user-user
user_similarity = cosine_similarity(user_movie_ratings)
user_similarity_df = pd.DataFrame(
    user_similarity,
    index=user_movie_ratings.index,
    columns=user_movie_ratings.index
)

# Similarity item-item
item_similarity = cosine_similarity(user_movie_ratings.T)
item_similarity_df = pd.DataFrame(
    item_similarity,
    index=user_movie_ratings.columns,
    columns=user_movie_ratings.columns
)
```

Gambar 5. Cosine Similarity

2.2.6 Pembangunan Model Collaborative Filtering

Terdapat dua model:

- User-Based Collaborative Filtering
- Item-Based Collaborative Filtering

Kode:

```
def recommend_movies_user(user_id, num=5):
    if user_id not in user_similarity_df.index:
        return "User tidak ditemukan!"

    sim_scores = user_similarity_df[user_id]
    weighted_ratings = user_movie_ratings.T.dot(sim_scores) / sim_scores.sum()

    user_ratings = user_movie_ratings.loc[user_id]
    unrated = user_ratings[user_ratings == 0].index

    return weighted_ratings[unrated].sort_values(ascending=False).head(num).index.tolist()

def recommend_movies_item(movie_name, num=5):
    if movie_name not in item_similarity_df.index:
        return "Film tidak ditemukan!"

    similar = item_similarity_df[movie_name].sort_values(ascending=False)
    similar = similar[1:num+1]
    return similar.index.tolist()
```

Gambar 6. Pembangunan Model

2.2.7 Visualisasi Model

Kode :

```
plt.imshow(item_similarity[:20, :20], cmap='viridis')
plt.colorbar()
```

Gambar 7. Visualisasi Model

3. ANALISA DAN PEMBAHASAN

Bab ini menyajikan hasil analisis berdasarkan tahapan penelitian yang telah dijelaskan pada Bab 2. Seluruh proses mulai dari pemuatan dataset, pembersihan data, analisis eksploratif, pembentukan *user-item matrix*, perhitungan *cosine similarity*, hingga pembangunan model rekomendasi dibahas secara detail. Analisa ini bertujuan untuk melihat bagaimana setiap tahapan memberikan kontribusi terhadap performa sistem rekomendasi film menggunakan metode Collaborative Filtering.

3.1 Analisa Berdasarkan Tahapan Penelitian

3.1.1 Analisa Import Library

Tahap pertama pada proses penelitian adalah melakukan import library yang digunakan untuk membaca data, memproses dataset, menghitung similarity, serta melakukan visualisasi. Pada tahap ini, library seperti *pandas*, *numpy*, *sklearn*, *matplotlib*, dan *diffli* berhasil dijalankan tanpa kendala.

Analisa terhadap tahap ini menunjukkan bahwa seluruh library yang diimport merupakan komponen fundamental bagi sistem rekomendasi, terutama *sklearn* yang digunakan untuk menghitung cosine similarity dan *diffli* untuk pencarian judul film. Ketersediaan library ini memastikan bahwa proses analisis data hingga pembangunan model dapat dilakukan secara optimal.

3.1.2 Analisa Pemuatan Dataset

Pada tahap ini, dataset *ratings.csv* dan *movies.csv* dimuat menggunakan *pandas*. Analisa dari proses pemuatan dataset menunjukkan beberapa hal:

- Struktur dataset terbaca dengan baik sehingga kolom *userId*, *movieId*, *rating*, dan *title* terbaca jelas.
- Jumlah baris yang besar pada data rating memberikan informasi bahwa dataset memiliki cakupan interaksi pengguna yang cukup luas.

- c. Dataset film berisi movieId dan title, yang menjadi informasi utama untuk identifikasi film.

Pemahaman terhadap isi dataset sangat penting karena akan menentukan kualitas matriks dan perhitungan similarity pada tahap berikutnya.

3.1.3 Analisa Preprocessing Data

Tahap ini merupakan salah satu bagian paling penting dan berpengaruh pada model. Dari hasil pembersihan data, diperoleh beberapa temuan:

- a. Data tidak valid berhasil dihapus, seperti nilai kosong atau film yang tidak ditemukan pada data movies.
- b. Tipe data telah diseragamkan, terutama userId dan movieId ke tipe numerik, sehingga kompatibel dengan proses merge dan pivot.
- c. Duplikasi rating dihilangkan dengan mengambil rata-rata rating pengguna jika terjadi pengulangan.
- d. Proses merge dataset berhasil dan menghasilkan tabel gabungan yang memuat userId, movieId, rating, dan title secara lengkap.

Analisa menunjukkan bahwa tahap preprocessing berperan besar dalam memastikan tidak ada gangguan struktural pada proses pembentukan matriks.

3.1.4 Analisa Eksplorasi Data (EDA)

Tahap EDA dilakukan untuk memahami karakteristik dasar dataset. Hasil yang ditemukan:

- a. Film dengan jumlah rating terbanyak dapat diidentifikasi, yang menunjukkan film populer dalam dataset.
- b. Distribusi rating terlihat tidak merata, banyak pengguna memberi rating pada sedikit film, sehingga data bersifat sparse.
- c. Sebagian film jarang dirating, mempengaruhi nilai similarity pada model Item Based.

Analisa ini penting karena memberi gambaran kondisi data yang akan mempengaruhi kualitas hasil rekomendasi pada model Collaborative Filtering.

3.1.5 Analisa Pembentukan User-Item Matrix

User-item matrix dibentuk melalui proses pivot. Hasil analisa terhadap matriks:

- a. Matriks sangat besar dan bersifat sparse, ditandai dengan banyak nilai nol.
- b. Setiap kolom mewakili film dan setiap baris mewakili pengguna, sehingga pola penilaian mulai terlihat.
- c. Nilai nol tidak dianggap rating, tetapi sebagai indikasi bahwa pengguna tidak menilai film tersebut.

User-item matrix inilah yang menjadi dasar seluruh perhitungan similarity yang dilakukan di tahap selanjutnya.

3.1.6 Analisa Perhitungan Cosine Similarity

Cosine similarity digunakan untuk menghitung kemiripan antar pengguna dan antar film. Dari proses ini diperoleh beberapa analisa:

- a. Similarity antar pengguna berhasil menggambarkan kedekatan pola rating.
- b. Similarity antar film menunjukkan film-film dengan pola penilaian serupa meskipun tidak memiliki metadata genre di dalam proses.
- c. Nilai similarity yang tinggi menandakan pola penilaian yang sangat mirip, menjadi dasar kuat dalam pemberian rekomendasi.
- d. Cosine similarity terbukti cocok untuk dataset sparse karena tidak membutuhkan data penuh pada setiap baris.

Secara keseluruhan, tahap similarity ini menjadi salah satu faktor utama yang menentukan keberhasilan model rekomendasi.

3.1.7 Analisa Pembangunan Model Rekomendasi

Berdasarkan nilai similarity yang telah dihitung, dua model berhasil dibangun:

a. User-Based Collaborative Filtering

Model ini bekerja dengan mencari pengguna yang memiliki kemiripan tinggi. Analisa hasilnya menunjukkan bahwa:

1. Rekomendasi yang dihasilkan bersifat sangat personal.
2. Model bergantung pada jumlah pengguna yang memiliki pola rating mirip.
3. Kinerja dapat terpengaruh jika data terlalu sparse.

b. Item-Based Collaborative Filtering

Model ini mencari film yang memiliki kemiripan pola penilaian. Hasil Analisa menunjukkan bahwa:

1. Rekomendasi lebih konsisten karena hubungan antar film tidak mudah berubah.
2. Model bekerja lebih baik pada film-film dengan banyak rating.

Kedua model ini berhasil dibangun dan dapat menghasilkan daftar rekomendasi sesuai pola pada data.

3.1.8 Analisa Visualisasi Model (Heatmap)

Heatmap digunakan sebagai alat visualisasi similarity antar film. Analisa visualisasi menunjukkan:

- a. Film-film dengan similarity tinggi membentuk cluster warna terang. Film dengan rating sedikit tampak lebih gelap, menandakan similarity yang rendah.
- b. Heatmap memberikan gambaran visual bahwa data similarity telah berhasil dihitung dengan benar.

Visualisasi ini membantu memahami struktur kemiripan yang tidak dapat dilihat hanya dari angka dalam matriks.

3.2 Analisa Implementasi Model dan Hasil Rekomendasi

3.2.1 Analisa Implementasi User-Based Collaborative Filtering

Model User-Based dibangun dengan memanfaatkan nilai similarity antar pengguna. Hasil analisa menunjukkan beberapa poin penting:

- a. Nilai similarity antar pengguna berhasil dihitung dengan baik, sehingga sistem dapat mengidentifikasi pengguna lain yang memiliki pola rating paling mirip dengan pengguna target.
- b. Model mampu memberikan rekomendasi yang bersifat personal, karena film yang ditampilkan merupakan film yang disukai pengguna lain yang mirip.
- c. Model bekerja optimal ketika pengguna memberikan lebih banyak rating, karena semakin banyak data yang tersedia, semakin tinggi kualitas perhitungan similarity.
- d. Pada pengguna dengan sedikit rating, model kesulitan menemukan kecocokan, sehingga menghasilkan rekomendasi yang kurang stabil.

Dengan demikian, model User-Based sangat bergantung pada jumlah dan keragaman rating yang diberikan oleh pengguna.

3.2.2 Analisa Implementasi Item-Based Collaborative Filtering

Item-Based Collaborative Filtering dibangun dengan menghitung tingkat kemiripan antar film. Analisa implementasinya menunjukkan bahwa:

- a. Pola similarity antar film lebih stabil dibanding similarity antar pengguna, sehingga model ini cenderung menghasilkan rekomendasi yang lebih konsisten.
- b. Film yang memiliki banyak rating atau rating serupa dari banyak pengguna lebih sering muncul dalam rekomendasi karena memiliki pola kemiripan yang kuat.
- c. Model ini lebih tahan terhadap data sparse, karena meskipun pengguna hanya menilai sedikit film, nilai similarity antar film tetap dapat dihitung dari seluruh dataset.
- d. Item-Based lebih efektif dalam memberikan rekomendasi bagi pengguna baru yang sudah menilai minimal satu film.

Hasil ini menunjukkan bahwa pendekatan Item-Based sangat sesuai untuk dataset film dan rating yang jumlahnya besar dan variatif.

3.2.3 Analisa Akurasi Fitur Pencarian Judul Film (String Matching)

Fitur pencarian judul film menggunakan teknik *string matching* juga dianalisa untuk melihat efektivitasnya:

- a. Fungsi mampu mendeteksi judul film meskipun pengguna melakukan kesalahan ejaan (*typo*) atau menuliskan sebagian judul saja.
- b. *Matching ratio* yang dihasilkan cukup tinggi pada sebagian besar input, sehingga judul film yang direkomendasikan sistem relevan dengan maksud pengguna.
- c. Fitur ini membantu pengguna dalam menavigasi dataset sebelum mendapatkan rekomendasi berikutnya.

Secara keseluruhan, fitur pencarian ini bekerja sebagai pelengkap yang penting untuk meningkatkan usability sistem.

3.2.4 Analisa Hasil Rekomendasi yang Dihasilkan Model

Tahap ini menganalisis hasil rekomendasi yang diberikan oleh kedua model Collaborative Filtering:

- a. Rekomendasi dari User-Based cenderung menampilkan film yang memang disukai pengguna lain dengan preferensi serupa.
- b. Rekomendasi dari Item-Based umumnya lebih terarah, karena film yang ditampilkan benar-benar memiliki pola rating serupa dengan film yang sudah dirating pengguna.
- c. Kedua model mampu memberikan rekomendasi yang relevan, namun karakteristiknya berbeda:
 1. User-Based: lebih personal, dipengaruhi perilaku pengguna lain
 2. Item-Based: lebih stabil, dipengaruhi pola rating film
- d. Pada pengujian beberapa pengguna acak, rekomendasi yang dihasilkan memiliki pola kemiripan yang jelas dengan film-film yang sebelumnya dirating.

Hasil analisa ini menunjukkan bahwa kedua model dapat berfungsi dengan baik sesuai desain metodologinya.

3.2.5 Analisa Visualisasi Kemiripan Antar Film

Visualisasi heatmap digunakan untuk melihat struktur similarity yang terbentuk. Analisa visual menunjukkan bahwa:

- a. Cluster warna terang menandakan adanya kelompok film yang memiliki pola rating sangat mirip.
- b. Perbedaan intensitas warna memberikan gambaran hubungan antar film, mendukung validitas hasil similarity yang digunakan model.
- c. Film dengan interaksi tinggi (rating banyak) cenderung menjadi pusat cluster karena memiliki data kemiripan yang lebih lengkap.

- d. Visualisasi ini juga memperlihatkan bagian matriks yang lebih gelap, yaitu film yang jarang dirating dan memiliki similarity rendah dengan film lainnya.

Visualisasi ini membuktikan bahwa struktur similarity yang dihitung oleh algoritma bekerja dengan baik dan menghasilkan pola hubungan yang logis.

3.2.6 Analisa Kinerja Sistem Berdasarkan Pola Pengguna

Untuk memastikan sistem bekerja sesuai kebutuhan, dilakukan pengujian pada beberapa pola pengguna yang berbeda:

- a. Pengguna aktif (banyak rating)
Model memberikan rekomendasi yang sangat relevan dan akurat.
- b. Pengguna pasif (sedikit rating)
Item-Based memberikan hasil lebih stabil dibanding User-Based.
- c. Pengguna dengan preferensi genre tertentu
Sistem tetap mampu mengelompokkan film yang mirip meskipun tidak menggunakan metadata genre secara eksplisit.

Hal ini menunjukkan bahwa model dapat beradaptasi dengan berbagai tipe pengguna.

3.2.7 Analisa Kualitas dan Kesesuaian Model Terhadap Dataset

Secara keseluruhan, hasil implementasi menunjukkan bahwa:

- a. Dataset yang telah dibersihkan menghasilkan matriks yang valid untuk perhitungan similarity.
- b. Cosine similarity bekerja efektif untuk struktur data sparse.
- c. Model Collaborative Filtering memberikan rekomendasi yang relevan, stabil, dan logis.
- d. Fitur visualisasi memperkuat bukti bahwa pola similarity telah terbentuk dengan benar.

3.3 Analisa Hasil Implementasi Fungsi Pencarian Judul Film

Selain proses pembentukan model rekomendasi, penelitian ini juga mengimplementasikan fitur pencarian judul film berdasarkan teknik *string matching* menggunakan library difflib. Analisa terhadap fitur ini menunjukkan bahwa:

- a. Fungsi pencarian mampu mendeteksi kemiripan kata, meskipun pengguna tidak menuliskan judul film secara tepat.
- b. Hasil pencarian memudahkan pengguna menemukan film ketika mereka hanya mengingat sebagian judul atau mengalami kesalahan penulisan.
- c. *String matching* memberikan alternatif penting pada sistem rekomendasi, terutama untuk pengguna yang ingin melihat informasi film terlebih dahulu sebelum mendapatkan rekomendasi lain.
- d. Pendekatan *ratio matching* pada difflib menghasilkan nilai kesesuaian yang cukup stabil meskipun pada dataset dengan banyak judul film.

Implementasi fitur ini berperan sebagai pelengkap model rekomendasi, terutama dalam membantu navigasi pengguna terhadap dataset film.

3.4 Analisa Pola Popularitas Film Berdasarkan Rating

Hasil eksplorasi data pada tahap EDA menunjukkan bahwa tidak semua film mendapatkan jumlah rating yang sebanding. Analisa lebih lanjut terhadap pola popularitas menunjukkan bahwa:

- a. Film yang memiliki rating tinggi tidak selalu memiliki rating banyak, menandakan bahwa film dapat disukai tetapi tidak populer secara keseluruhan.
- b. Film populer cenderung membentuk kelompok tersendiri pada heatmap, karena jumlah interaksi yang tinggi memungkinkan pola similarity lebih terbaca.

- c. Pola popularitas ini mempengaruhi model Item-Based Collaborative Filtering, sebab film dengan rating rendah sulit untuk dihitung similaritasnya akibat minim interaksi.
- d. Pengelompokan film berdasarkan jumlah rating membantu dalam menilai stabilitas rekomendasi yang diberikan oleh model.

Dengan demikian, variasi jumlah rating menjadi faktor penting dalam pembahasan kualitas rekomendasi.

3.5 Analisa Pengaruh Sparsity terhadap Kinerja Model

Karena matriks user-item bersifat sparse, yaitu lebih banyak nilai nol daripada nilai rating, analisa dilakukan untuk melihat bagaimana kondisi ini mempengaruhi hasil model:

- a. User-Based Collaborative Filtering lebih sensitif terhadap sparsity, karena kemiripan antar pengguna sulit dihitung jika pengguna hanya memberi sedikit rating.
- b. Item-Based lebih tahan terhadap sparsity, karena pola kemiripan antar film dapat terbentuk dengan memanfaatkan penilaian dari banyak pengguna lainnya.
- c. Sparsity juga menyebabkan sebagian film tidak bisa direkomendasikan jika tidak memiliki interaksi yang cukup untuk dihitung similaritasnya.
- d. Meskipun sparse, cosine similarity tetap memberikan hasil yang cukup akurat berkat normalisasi vektor yang tidak bergantung pada frekuensi data.

Analisa ini memperlihatkan bahwa pemilihan metode CF dan metode similarity sangat berpengaruh pada kualitas keluaran sistem.

3.6 Analisa Output Rekomendasi pada Beberapa Skenario Pengguna

Untuk menilai bagaimana model bekerja dalam kondisi berbeda, dilakukan analisa terhadap beberapa skenario pengguna:

- a. Pengguna dengan Banyak Rating
 1. Model menghasilkan rekomendasi yang lebih akurat.
 2. Banyaknya data interaksi memudahkan model mengidentifikasi pola preferensi pengguna.
- b. Pengguna dengan Sedikit Rating
 1. Model User-Based memberikan hasil kurang stabil karena kesulitan mencari pengguna mirip.
 2. Model Item-Based tetap dapat memberikan rekomendasi berdasarkan kesamaan film yang pernah ditonton pengguna.
- c. Pengguna yang Menyukai Genre Khusus
 1. Model mampu memunculkan film dengan pola rating serupa meskipun genre tidak dihitung secara eksplisit.
 2. Hal ini menunjukkan kekuatan CF dalam membaca pola perilaku berdasarkan rating, bukan metadata film.

Analisa ini menegaskan bahwa kualitas rekomendasi sangat bergantung pada jumlah dan pola rating pengguna.

3.7 Analisa Kesesuaian Algoritma Cosine Similarity

Berdasarkan seluruh hasil proses perhitungan, didapati bahwa cosine similarity adalah metode yang tepat untuk penelitian ini. Alasan yang diperoleh dari analisa hasil:

- a. Normalisasi sudut membuat nilai similarity tidak dipengaruhi besar kecilnya rating, hanya pola rating.
- b. Cocok untuk data sparse, sehingga tidak terpengaruh banyaknya nilai nol pada matriks.
- c. Komputasi cepat dan sederhana, sehingga mampu menghitung similaritas dalam dataset yang memiliki banyak kolom film.
- d. *Output* similarity cukup stabil meskipun ada beberapa film dengan rating rendah.

Analisa ini menunjukkan bahwa cosine similarity memberikan dasar matematis yang kuat dalam model Collaborative Filtering.

3.8 Analisa Kelemahan Sistem Berdasarkan Hasil Pengujian

Meski sistem berjalan sesuai dengan rancangan, analisa dalam pembahasan menunjukkan beberapa keterbatasan:

- a. Kurangnya metadata film
Semua komputasi similarity hanya berdasarkan rating.
Sistem tidak mempertimbangkan genre, aktor, tahun rilis, atau durasi film.
- b. Cold-start problem
Pengguna baru yang belum memberikan rating tidak dapat diberi rekomendasi.
- c. Film dengan sedikit rating kurang tampil dalam rekomendasi
Film yang jarang dinilai tidak memiliki cukup data untuk dihitung similarity-nya.

Kelemahan-kelemahan ini merupakan bagian dari karakteristik CF dan menjadi poin penting untuk pengembangan sistem.

3.9 Analisa Hubungan Tahapan Penelitian dengan Hasil Akhir

- a. Preprocessing: mempengaruhi keakuratan data matriks.
- b. EDA: menentukan pemahaman awal pola rating.
- c. Pembentukan matriks: mempengaruhi perhitungan similarity.
- d. Similarity: mempengaruhi model rekomendasi.
- e. Model: menghasilkan rekomendasi yang dianalisis pada skenario pengguna.
- f. Visualisasi: memvalidasi pola hubungan antar film.

Analisa ini menunjukkan bahwa implementasi sistem berjalan konsisten dengan metode penelitian.

4. KESIMPULAN

Berdasarkan seluruh rangkaian penelitian yang telah dilakukan, mulai dari pengolahan dataset, analisis pola interaksi pengguna, pembentukan model, hingga pengujian hasil rekomendasi, dapat disimpulkan bahwa metode Collaborative Filtering mampu memberikan rekomendasi film yang relevan dan sesuai preferensi pengguna. Struktur data rating dan hubungan antar pengguna maupun antar film dapat dipetakan dengan baik sehingga sistem mampu mengenali pola kesukaan dengan cukup akurat.

Penerapan cosine similarity terbukti efektif dalam mengukur kemiripan, baik antar pengguna maupun antar film, sehingga menghasilkan rekomendasi yang logis dan sejalan dengan pola penilaian sebelumnya. Model User-Based memberikan rekomendasi yang personal berdasarkan kesamaan perilaku pengguna lain, sementara model Item-Based memberikan hasil yang lebih stabil berdasarkan kemiripan pola rating antar film. Kedua pendekatan tersebut saling melengkapi dan menunjukkan bahwa Collaborative Filtering dapat bekerja optimal meskipun dataset bersifat sparse.

Selain itu, proses analisa dan visualisasi seperti heatmap memperkuat bahwa pola kemiripan dalam dataset terbentuk dengan jelas, dan hasil rekomendasi yang muncul sesuai dengan cluster tersebut. Secara keseluruhan, penelitian ini membuktikan bahwa sistem rekomendasi berbasis Collaborative Filtering dapat diterapkan secara efektif untuk membantu pengguna menemukan film yang sesuai dengan preferensi mereka, serta mampu meningkatkan pengalaman pemilihan film secara lebih terarah dan personal.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada berbagai pihak yang telah memberikan dukungan dalam proses penyusunan penelitian ini. Terima kasih yang sebesar-besarnya disampaikan kepada Dosen Pembimbing yang telah memberikan arahan, masukan, serta bimbingan yang sangat berharga

selama penelitian berlangsung. Setiap saran dan koreksi yang diberikan telah membantu penulis dalam menyempurnakan penelitian ini sehingga dapat diselesaikan dengan baik.

Penulis juga menyampaikan apresiasi kepada rekan-rekan tim yang telah berkolaborasi dalam pengumpulan data, implementasi program, proses analisis, hingga penyusunan laporan penelitian. Kerja sama yang baik, diskusi yang konstruktif, serta semangat kebersamaan sangat membantu dalam menghadapi berbagai kendala selama penelitian.

Ucapan terima kasih turut penulis sampaikan kepada pihak penyedia dataset yang telah menyediakan sumber data yang diperlukan sehingga proses analisis dan pembangunan sistem rekomendasi dapat berjalan dengan lancar. Penulis juga berterima kasih kepada seluruh pihak yang telah memberikan dukungan moral maupun bantuan teknis, baik secara langsung maupun tidak langsung, yang berkontribusi pada kelancaran penyusunan penelitian ini.

Penulis sangat menghargai dukungan keluarga, teman-teman, serta orang-orang terdekat yang senantiasa memberikan motivasi, semangat, dan doa selama proses penelitian berlangsung. Dukungan tersebut menjadi dorongan penting bagi penulis dalam menyelesaikan penelitian ini dengan penuh tanggung jawab.

Penulis berharap penelitian ini dapat memberikan kontribusi positif bagi pengembangan sistem rekomendasi, khususnya dalam penerapan metode Collaborative Filtering, serta menjadi referensi bagi penelitian selanjutnya di bidang terkait. Semoga penelitian ini dapat memberikan manfaat bagi pembaca maupun pihak-pihak yang berkepentingan.

REFERENCES

- Bobadilla, J., Hernando, A., Ortega, F., & Gutiérrez, A. (2020). Recommender systems survey. *Knowledge Based Systems*, 192, 105-213.
- Ekstrand, M. D., Konstan, J. A., & Riedl, J. (2021). Collaborative filtering recommender systems. *Foundations and Trends in Human Computer Interaction*, 14(2), 1-82.
- Ferdiana, R., & Indraswari, R. (2021). Improving movie recommendation using item-based collaborative filtering with cosine similarity. *Journal of Information Systems Engineering and Business Intelligence*, 7(2), 98-107.
- Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2020). A recommendation system for movie selection using collaborative filtering. *The Scientific World Journal*, 2020, 1-10.
- Koren, Y., Bell, R., & Volinsky, C. (2020). Matrix factorization techniques for recommender systems. *Computer*, 53(8), 30-37.
- Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2020). Recommender system application developments: A survey. *Decision Support Systems*, 129, 113-132.
- Muslim, M. A., & Pratama, M. Y. (2021). Implementation of collaborative filtering for movie recommendation system. *Journal of Information Systems Engineering and Business Intelligence*, 7(1), 12-21.
- Putra, R. A., & Ramadhan, R. (2022). Movie recommendation system using cosine similarity and collaborative filtering. *Proceedings of the 2022 International Conference on Information Technology and Computing*, 45-50.
- Riyadi, S., & Yulianti, E. (2022). Hybrid collaborative filtering for movie recommendation using similarity adjustment. *Procedia Computer Science*, 216, 493-500.
- Sharma, A., & Gera, C. (2021). A survey of recommendation system: Research directions, challenges, and solutions. *Artificial Intelligence Review*, 54, 559-611.
- Susanto, A., & Ariyanto, F. (2020). Movie recommendation system using cosine similarity and user preference modeling. *International Journal of Advanced Computer Science and Applications*, 11(5), 447-453.
- Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning-based recommender system: A survey and new perspectives. *ACM Computing Surveys*, 52(1), 1-38.