

Sistem Rekomendasi Film Dengan *Item-Based Collaborative Filtering* Menggunakan Flask Framework

I Made Nusa Yudiskara^{a1}, I Gede Santi Astawa^{a2}, Luh Gede Astuti^{a3}, Made Agung Raharja^{a4}, I Made Widiartha^{a5}, I Wayan Supriana^{a6}

^aProgram Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Badung, Bali, Indonesia

¹yudiskara@gmail.com

²santi.astawa@cs.unud.ac.id

³lg.astuti@unud.ac.id

⁴made.agung@unud.ac.id

⁵madewidiartha@unud.ac.id

⁶wayan.supriana@unud.ac.id

Abstrak

Jumlah informasi yang tersedia seiring dengan berkembangnya teknologi akan menyulitkan pengguna internet untuk mencari dan menyaring informasi di internet. Sistem rekomendasi dirancang untuk membantu pengguna menemukan data dengan cepat yang mungkin sesuai dengan data pribadi mereka dan mengurangi informasi yang berlebihan. Item atau film dalam hal ini dapat disarankan berdasarkan rating film yang diberikan oleh pengguna atau berdasarkan pengguna lain dengan kebiasaan serupa. Dalam penelitian ini penulis mengimplementasikan item-based collaborative filtering dengan membuat sistem berupa aplikasi web menggunakan python dan flask framework dimana aplikasi ini dapat memberikan user rekomendasi film berdasarkan rating yang diberikan sebelumnya oleh user. Dataset pada penelitian ini diperoleh dari website grouplens.org dengan total jumlah data rating 100.836 dari 610 user berbeda terhadap 9742 judul film. Hasil pengujian MAE dan RMSE, dengan 20 top-N similarity untuk menghitung prediksi rating, mendapatkan skor yang terbaik dengan nilai MAE 0,513536 dan nilai RMSE 0,659217674. Dengan melihat nilai RMSE dan MAE, variance error dari implementasi sistem tidak terlalu besar. Untuk pengujian pemberian ranking menggunakan Spearman Rank Correlation, IBCF dengan 20 top-N similarity dan 15 top-N similarity mendapatkan skor yang paling tinggi yaitu 0,91515 yang berarti peringkat yang diberikan sistem untuk rating prediksi sudah hampir mirip dengan peringkat rating aktual.

Kata Kunci: Rekomendasi, Recommendation System, Collaborative Filtering, Item-based CF, Aplikasi Web

1. Pendahuluan

Sistem rekomendasi dirancang untuk membantu pengguna menemukan data dengan cepat yang mungkin sesuai dengan data pribadi mereka dan mengurangi informasi yang berlebihan. Sistem rekomendasi yang efektif tidak hanya dapat memfasilitasi proses pencarian informasi pengguna [1], tetapi juga dapat menciptakan loyalitas pelanggan dan meningkatkan keuntungan bagi perusahaan. Dengan pentingnya peran sistem rekomendasi dalam sistem informasi online, sistem rekomendasi telah menjadi topik penelitian yang aktif dan semakin menarik perhatian dalam bidang sistem temu kembali informasi dan komunitas data mining [2].

Amazon pertama kali menciptakan dan menggunakan Item-based collaborative filtering pada tahun 1998. Daripada mencocokkan pengguna dengan pelanggan serupa, Item collaborative based filtering mencocokkan setiap produk yang dibeli dan dirating oleh pelanggan dengan item serupa, kemudian mengumpulkan item-item serupa, yang nantinya akan dihitung prediksi nilai rating dimana item dengan prediksi rating tertinggi akan dimasukkan ke dalam daftar item rekomendasi [3]. Penelitian dengan topik item based collaborative filtering sudah banyak dilakukan. Contohnya penelitian oleh Lisniati Dzumiroh dan Ristu Saptono (2016). dengan judul "Penerapan Metode Collaborative Filtering Menggunakan Rating Implisit pada Sistem Perekomendasi Pemilihan Film di Rental VCD", berdasarkan hasil penelitian, item-based collaborative filtering menghasilkan kualitas rekomendasi

yang lebih baik dibandingkan user-based collaborative filtering [4]. Nilai akurasi yang diperoleh dari item-based collaborative filtering yang diukur dengan menggunakan metrik akurasi MAE mendapatkan nilai yang lebih rendah dibandingkan user-based collaborative filtering. Peneliti juga mengkombinasikan item-based collaborative filtering dengan fitur konten, dimana peningkatan nilai akurasi tidak signifikan, dan hasil rekomendasi tidak memiliki karakteristik rekomendasi oleh collaborative filtering.

2. Metode Penelitian

2.1. Pengumpulan Data

Data pada penelitian ini menggunakan data sekunder dimana data-data tersebut berupa data film, dan rating yang di berikan pada film dari tanggal 29 Maret 1996 sampai 24 September, 2018. User dipilih secara acak, semua user setidaknya merating 20 film, tidak ada informasi demografis yang disertakan. Setiap user diwakili oleh "Id" yang diberikan, tidak ada informasi lain yang disediakan.

movieId	title	genres
1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy
2	Jumanji (1995)	Adventure Children Fantasy
3	Grumpier Old Men (1995)	Comedy Romance
4	Waiting to Exhale (1995)	Comedy Drama Romance
5	Father of the Bride Part II (1995)	Comedy
6	Heat (1995)	Action Crime Thriller
7	Sabrina (1995)	Comedy Romance
8	Tom and Huck (1995)	Adventure Children
9	Sudden Death (1995)	Action
10	GoldenEye (1995)	Action Adventure Thriller
11	American President, The (1995)	Comedy Drama Romance
12	Dracula: Dead and Loving It (1995)	Comedy Horror
13	Balto (1995)	Adventure Animation Children
14	Nixon (1995)	Drama

Gambar 1. Dataset Movies

Gambar berikut merupakan data movies, yang digunakan dalam penelitian ini, movieId merupakan Id film, title merupakan judul film, dan genre yang merupakan genre film. Total data pada dataset ini adalah 9.742 data

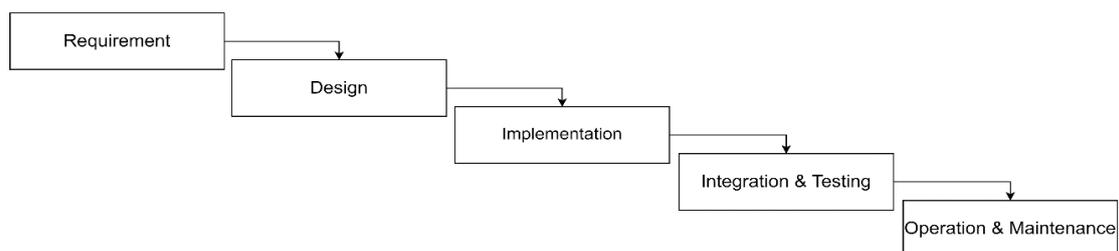
userId	movieId	rating	timestamp
1	31	2.5	1260759144
1	1029	3	1260759179
1	1061	3	1260759182
1	1129	2	1260759185
1	1172	4	1260759205
1	1263	2	1260759151
1	1287	2	1260759187
1	1293	2	1260759148
1	1339	3.5	1260759125
1	1343	2	1260759131
1	1371	2.5	1260759135
1	1405	1	1260759203
1	1953	4	1260759191

Gambar 2. Dataset Ratings

Gambar berikut merupakan data ratings, yang digunakan dalam penelitian ini, userId merupakan Id user, movieId merupakan Id film, timestamp merupakan waktu pemberian rating oleh user format yang digunakan adalah timestamp with milliseconds, data rating memperlihatkan nilai yang diberikan oleh user kepada film yang telah mereka tonton. Jumlah user pada data ini adalah 610, dan total data pada dataset ini adalah 100.836 data

2.2. Metode Pengembangan Sistem

Pada proses pengembangan sistem digunakan model *Waterfall*. Dalam metode *Waterfall* proses pengembangan sistem dibagi kedalam beberapa tahap, setiap tahap harus diselesaikan sebelum tahap selanjutnya dan tidak boleh ada tahap yang dilakukan secara bersamaan. Berikut merupakan ilustrasi model *Waterfall* [5].



Gambar 3. Tahapan model waterfall

Berikut merupakan tahapan dalam model waterfall :

1. Requirements

Semua kebutuhan yang mungkin diperlukan dalam sistem yang akan dikembangkan dicatat dan didokumentasikan. Pada tahapan ini pengembang akan mendapatkan informasi kebutuhan sistem baik itu kebutuhan fungsional ataupun non-fungsional

2. Design

Sesuai dengan kebutuhan sistem yang didapatkan pada tahap pertama. Tahap selanjutnya adalah membuat desain sistem, mencatat kebutuhan hardware dan software, serta mendokumentasikan hasil desain sistem

3. Implementation

Berdasarkan inputan yang didapatkan dari tahap *system design*, sistem pertama dikembangkan kedalam kumpulan program yang disebut *units*, yang mana akan diintegrasikan pada tahap selanjutnya. Setiap unit akan dikembangkan dan diuji kegunaannya

4. Integration and Testing

Semua unit yang dikembangkan pada tahap implementation akan diintegrasikan pada sistem setelah dilakukan pengujian pada setiap unit. Setelah dilakukan integrasi, sistem akan diuji untuk mencari kesalahan yang mungkin terjadi dan melakukan perbaikan atas kesalahan tersebut.

5. Operation and Maintenance.

Setelah sistem diuji baik itu fungsional maupun non-fungsional, Sistem dapat dijalankan atau dioperasikan oleh pengguna. Jika terdapat masalah yang terjadi pada saat penggunaan akan dilakukan maintenance untuk memperbaiki masalah tersebut.

2.3. Item-based Collaborative Filtering

Amazon pertama kali menciptakan dan menggunakan Item-based collaborative filtering pada tahun 1998. Daripada mencocokkan pengguna dengan pelanggan serupa, Item collaborative based filtering mencocokkan setiap produk yang dibeli dan dirating oleh pelanggan dengan item serupa, kemudian mengumpulkan item-item serupa, yang nantinya akan dihitung prediksi nilai rating dimana item dengan prediksi rating tertinggi akan dimasukkan ke dalam daftar item rekomendasi [6]. Tahap pertama dalam IBCF adalah mencari similarity antar item, kemudian berdasarkan similarity yang didapatkan item yang telah dirating akan dicari dan user akan direkomendasikan berdasarkan item yang telah dirating [7].

2.4. Pearson Correlation Coefficient

Kalkulasi similarity atau nilai kemiripan antar item merupakan langkah penting dalam *Item-Based Collaborative Filtering* [8]. Setiap pasangan item akan dihitung tingkat kemiripan similarity antar item menggunakan persamaan Pearson Correlation Coefficient, dimana nilai yang dihasilkan oleh persamaan *Pearson Correlation Coefficient* memiliki rentang nilai antara +1,0 dan -1,0 [9]. Berikut merupakan keterangan dari nilai yang dihasilkan:

- Nilai similarity 0: pasangan item tidak berkorelasi (independen)
- Nilai similarity mendekati +1,0: pasangan item memiliki korelasi tinggi
- Nilai similarity mendekati -1,0: pasangan item bertolak belakang.

Persamaan Pearson Correlation Coefficient adalah sebagai berikut:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \quad (1)$$

Keterangan:

r = Nilai kemiripan antara x dan y

x_i = Nilai sebuah item x

y_i = Nilai sebuah item y

\bar{x} = Nilai rata-rata semua item x

\bar{y} = Nilai rata-rata semua item y

2.5. Weighted Sum

Setelah menghitung nilai similarity dari semua pasangan item, langkah selanjutnya adalah menghitung prediksi rating yang akan diberikan pada item yang belum dirating oleh pengguna. Hasil perhitungan prediksi kemudian akan diurutkan sesuai dari nilai yang paling tinggi, kemudian sistem akan merekomendasikan item-item yang diprediksi akan dirating dengan nilai tinggi oleh user. Persamaan yang digunakan untuk menghitung nilai prediksi adalah weighted sum [10]. Untuk menghitung prediksi nilai rating yang akan diberikan user, untuk semua item yang telah dirating user,

Jumlah nilai similarity dari N-item dengan nilai similarity teratas (*Top-N Similarity*) terhadap item prediksi dikalikan dengan nilai rating yang telah dirating user pada setiap item tersebut, dibagi dengan jumlah dari semua N nilai similarity teratas.

$$P(u,i) = \frac{\sum \text{semua item similar}(S_{i,N} * R_{u,i})}{\sum \text{semua item similar}(|S_{i,N}|)} \quad (2)$$

Keterangan ;

$P(u,i)$ = Nilai prediksi rating untuk user u pada item i.

$R(u,i)$ = Rating yang diberikan user u pada item i.

$S(i,N)$ = Nilai similarity untuk item i dan Top-N item similar

2.6. Pengujian Akurasi

Uji akurasi akan dilakukan menggunakan 3 metrik, yaitu MAE, RMSE, dan Spearman Correlation Coefficient. RMSE dan MAE dapat digunakan bersama untuk mengetahui variance error pada kumpulan nilai prediksi, nilai RMSE akan selalu lebih besar atau sama dengan MAE, semakin besar perbedaan antara keduanya maka semakin besar pula variance error untuk setiap individu pada sampel. Spearman Rank Correlation digunakan untuk menghitung seberapa mirip peringkat item aktual yang diberikan user dengan peringkat prediksi hasil rekomendasi sistem.

Berikut merupakan skenario pengujian akurasi sistem:

1. Persiapkan data yang berisi user serta rating yang telah diberikan pada berbagai judul film.
2. Dari data awal, sejumlah rating yang telah diberikan oleh beberapa user, akan sengaja dihilangkan.
3. Sistem akan memprediksi nilai rating yang telah dihilangkan dan dibandingkan dengan nilai rating sebenarnya yang diberikan user untuk memperoleh nilai akurasi.
4. Penulis menghitung akurasi menggunakan 3 metrik yaitu MAE, RMSE, dan Spearman Correlation Coefficient
5. Setelah dilakukan perhitungan akurasi dilakukan, laporan hasil pengujian akan dilampirkan dalam bentuk tabel.

2.6.1 MAE

Dalam statistika, MAE (Mean Absolute Error) adalah cara untuk mengukur tingkat kesalahan antara dua buah nilai untuk kasus yang serupa. Contohnya pada perbandingan X dan Y dimana X adalah nilai prediksi dan Y adalah nilai aktual. Semakin rendah nilai MAE maka semakin baik hasil prediksi, begitu pula sebaliknya. Secara formal didefinisikan sebagai berikut:

$$MAE = \frac{\sum_{i=1}^N |x_i - \hat{x}_i|}{N} \quad (3)$$

i = variable i .

N = Jumlah data.

x_i = Nilai data aktual.

\hat{x}_i = Nilai hasil prediksi.

Pengujian dilakukan pada 5 user dimana 20 rating yang sudah diberikan akan sengaja dihilangkan untuk digunakan sebagai data uji. Data tersebut kemudian dibandingkan dengan nilai sebenarnya untuk mendapatkan skor MAE

2.6.2 RMSE

RMSE (Root Mean Square Error) adalah cara yang sering digunakan untuk mengukur perbedaan nilai prediksi dan nilai aktual pada sebuah hasil prediksi, nilai RMSE selalu bernilai positif atau 0 (Hampir tidak pernah terjadi dalam praktik) yang akan menandakan nilai prediksi yang dihasilkan sempurna dan tidak memiliki kesalahan. Pada umumnya semakin kecil nilai RMSE maka semakin baik hasil prediksi. Secara formal didefinisikan sebagai berikut:

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (x_i - \hat{x}_i)^2}{N}} \quad (4)$$

N = Jumlah data.

x_i = Nilai data aktual.

\hat{x}_i = Nilai hasil prediksi

2.6.3 Spearman Correlation Coefficient

Uji Spearman merupakan salah satu uji statistik non paramateris yang digunakan untuk mengukur korelasi peringkat. Pengujian spearman akan menghitung skor seberapa baik sistem memberikan ranking dibandingkan dengan bagaimana urutan ranking yang seharusnya diberikan. Secara formal didefinisikan sebagai berikut:

$$r_s = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \quad (5)$$

Keterangan:

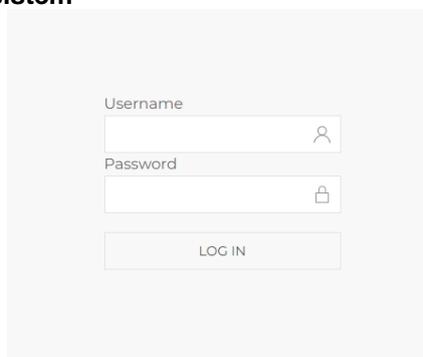
$d_i = R(X_i) - R(Y_i)$ adalah perbedaan peringkat prediksi dan peringkat sebenarnya.

n = jumlah data

Spearman rank correlation memiliki rentang nilai antara 1 dan -1. Jika nilai mendekati 1 maka urutan ranking item semakin sesuai, jika nilai mendekati -1 maka urutan ranking item berbanding terbalik dengan urutan peringkat sebenarnya, dan nilai mendekati 0 maka peringkat prediksi dan peringkat sebenarnya tidak memiliki korelasi.

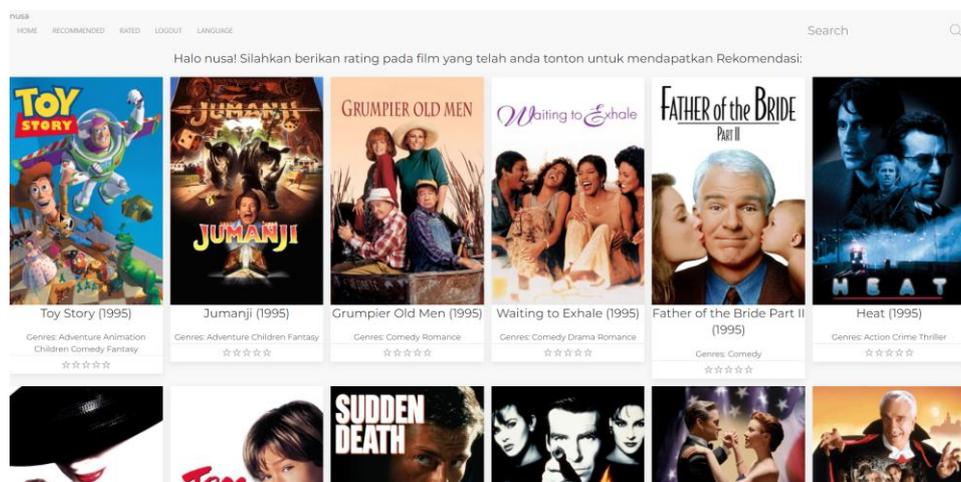
3. Hasil dan Pembahasan

3.1. Implementasi antarmuka sistem



Gambar 4. Tampilan Halaman Login dan Register

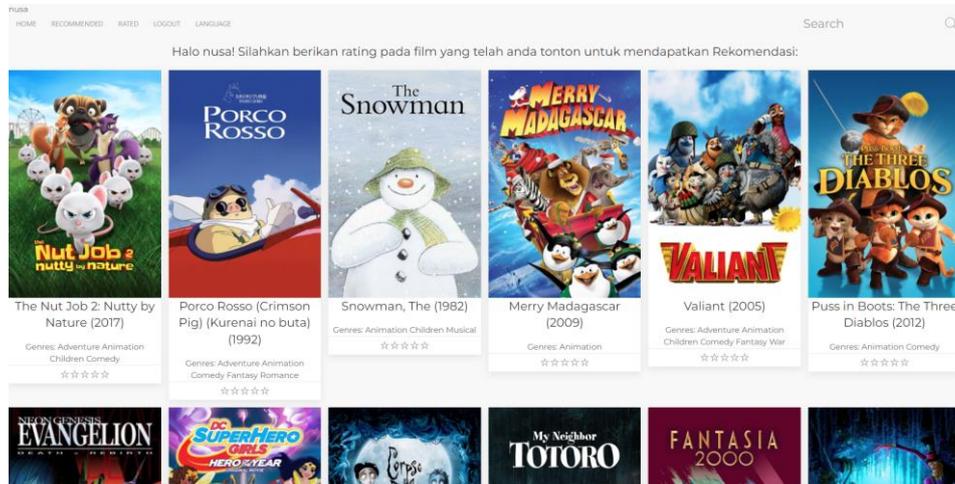
Gambar 4 merupakan tampilan halaman login dan register. Halaman ini digunakan user untuk masuk ke dalam sistem tujuan dari fitur ini adalah agar user dapat melakukan peratingan dan mendapatkan rekomendasi berdasarkan rating yang diberikan oleh film-film yang sebelumnya telah ditonton oleh user. Untuk melakukan login pada sistem user harus menginput username dan password yang sebelumnya sudah didaftarkan. Apabila user belum mendaftarkan username dan password, maka user harus menuju ke halaman register untuk mendaftar



Gambar 5. Tampilan Halaman Home

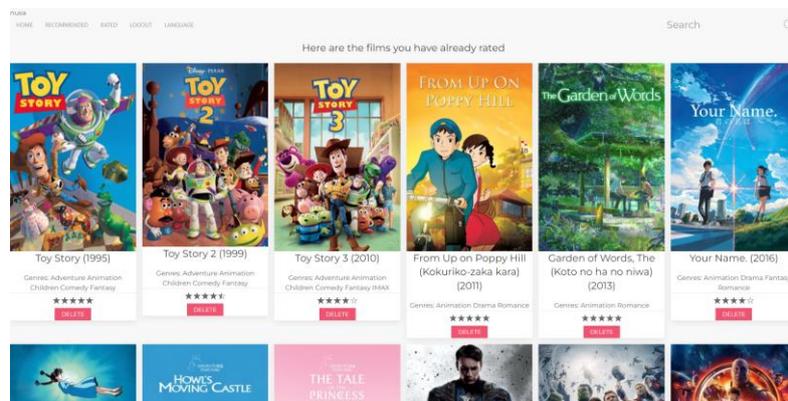
Gambar 5 merupakan tampilan halaman home. Halaman ini merupakan halaman utama pada sistem, user dapat mengakses halaman ini sebelum login ataupun register username dan password,

namun user tidak dapat melakukan peratingan dan tidak akan bisa mengakses halaman rekomendasi, karena belum memiliki akun. Pada halaman ini user dapat menjelajahi film-film yang sudah terdaftar pada database. Dalam satu halaman akan ditampilkan 18 film yang dapat dirating oleh user setelah melakukan login, untuk menampilkan halaman selanjutnya pada bagian paling bawah terdapat tombol Next dan Previous untuk menuju halaman sebelumnya. Selain itu user dapat melakukan pencarian dengan menginputkan judul atau memfilter genre film yang ingin ditampilkan



Gambar 6. Tampilan Halaman Recommendation

Gambar 6 merupakan tampilan halaman recommendation. Pada halaman ini user akan ditampilkan dengan film-film yang belum dirating oleh user. setiap item pada halaman ini diurutkan berdasarkan prediksi rating paling besar yang akan diberikan user setelah menonton film tersebut



Gambar 7. Tampilan Halaman Rating

Gambar 7 merupakan tampilan halaman rating. Pada halaman ini user dapat mengubah ataupun menghapus rating yang telah diberikan pada sebuah film dengan menghapus tombol delete. Mengubah ataupun menghapus item pada halaman ini akan mempengaruhi film-film yang akan ditampilkan pada halaman Rekomendasi

3.2. Pengujian Sistem

Tujuan dari pengujian adalah untuk memastikan semua fitur yang diimplementasikan pada sistem berjalan sesuai dengan ekspektasi dan seberapa akurat prediksi rating pada item yang belum dirating oleh pengguna, yang mana nantinya akan digunakan untuk memberikan rekomendasi film kepada pengguna. Pada penelitian ini dilakukan dua jenis pengujian yaitu Black Box Test, untuk menguji fitur yang ada pada aplikasi, dan pengujian akurasi untuk menghitung seberapa akurat rating prediksi, pengujian akurasi pada penelitian ini menggunakan 3 metrik yaitu MAE, RMSE, dan Spearman Correlation Coefficient.

3.3. Pengujian Black Box

Tabel 1. Black Box Test Skenario Pendaftaran

Ekspektasi dan Hasil Observasi (Data Benar)			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Username dan Password	Setelah menekan tombol register user akan diarahkan ke halaman login	User diarahkan ke halaman login	VALID
Ekspektasi dan Hasil Observasi (Data Salah)			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Menginput username terdaftar	Setelah menekan tombol register akan ditampilkan pesan error	User ditampilkan pesan error	VALID
Tidak menginput username dan/atau password	Setelah menekan tombol register akan ditampilkan pesan error	User ditampilkan pesan error	VALID

Tabel 2. Black Box Test Skenario Login

Ekspektasi dan Hasil Observasi (Data Benar)			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Username dan Password	Setelah menekan tombol login akan diarahkan ke halaman home	User diarahkan ke halaman home	VALID
Ekspektasi dan Hasil Observasi (Data Salah)			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Menginput username terdaftar	Setelah menekan tombol login akan ditampilkan pesan error	User ditampilkan pesan error	VALID
Tidak menginput username dan/atau password	Setelah menekan tombol login akan ditampilkan pesan error	User ditampilkan pesan error	VALID

Tabel 3. Black Box Test Skenario Logout

Ekspektasi dan Hasil Observasi			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Klik logout pada menu navigasi	Setelah menekan tombol logout sesi user akan selesai dan akan dikembalikan ke halaman index	Sesi user selesai dan dikembalikan ke halaman index	VALID

Tabel 4. Black Box Test Skenario Pencarian

Ekspektasi dan Hasil Observasi			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Kata kunci judul film	Setelah menginputkan kata kunci dan menekan enter akan ditampilkan film yang terkait dengan kata kunci	User ditampilkan film yang terkait dengan kata kunci	VALID

Tabel 5. Black Box Test Skenario Filter

Ekspektasi dan Hasil Observasi			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Kategori film	Setelah menginputkan kategori film dengan mencentang salah satu atau beberapa kategori film pada menu drop down pencarian kemudian tekan "Enter" pada keyboard, user ditampilkan film dengan kategori inputan kategori	User ditampilkan film dengan kategori sesuai inputan kategori	VALID

Tabel 6. Black Box Test Skenario Pencarian Dengan Filter

Ekspektasi dan Hasil Observasi			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Kategori film	Setelah menginputkan kata kunci dan kategori film dengan mencentang salah satu atau beberapa kategori film pada menu drop down pencarian, user ditampilkan film dengan inputan kata kunci dan kategori.	User ditampilkan film dengan kategori sesuai inputan kata kunci dan kategori	VALID

Tabel 7. Black Box Test Pemberian Rating

Ekspektasi dan Hasil Observasi			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
Rating Film	Setelah menginputkan nilai rating dengan cara memilih jumlah bintang yang akan diberikan kepada film yang telah ditonton, data rating akan tersimpan dan ditampilkan pada halaman rated	Data rating tersimpan dan ditampilkan pada halaman rated	VALID

Tabel 8. Black Box Test Pemberian Rekomendasi

Ekspektasi dan Hasil Observasi (Data Benar)			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
≥15 inputan rating film	Setelah memilih recommendation pada menu navigasi user akan ditampilkan sejumlah film rekomendasi dari sistem	User ditampilkan sejumlah film rekomendasi dari sistem	VALID
Ekspektasi dan Hasil Observasi (Data Salah)			
Input	Ekspektasi	Hasil Observasi	Kesimpulan
≤15 inputan rating film	Setelah memilih recommendation pada menu navigasi user akan ditampilkan pesan error dan diarahkan ke halaman home untuk merating paling tidak 15 film	User ditampilkan pesan error dan diarahkan ke halaman home untuk merating paling tidak 15 film	VALID

3.4. Hasil Pengujian Akurasi

3.4.1 MAE

Dari hasil pengujian akurasi sistem dengan metrik MAE, mengikuti skenario pengujian yang dijelaskan sebelumnya pada 2.6. dilakukan pengujian untuk top-N similarity 5, 10, 15, 20. Dengan meningkatkan jumlah nilai similarity yang digunakan untuk menghitung prediksi rating, nilai akurasi prediksi rating juga semakin baik. Top-20 Similarity mendapatkan akurasi terbaik dengan nilai MAE 0,513536 nilai tersebut 4,48% lebih baik dari Top-5 Similarity

Tabel 9. Hasil pengujian akurasi sistem dengan metrik MAE

Top-N Similarity	MAE
5	0,537652
10	0,531548
15	0,523733
20	0,513536

3.4.2 RMSE

Dari hasil pengujian akurasi sistem dengan metrik RMSE, mengikuti skenario pengujian yang dijelaskan sebelumnya pada 2.6. dilakukan pengujian untuk top-N similarity 5, 10, 15, 20. Dengan meningkatkan jumlah nilai similarity yang digunakan untuk menghitung prediksi rating, nilai akurasi prediksi rating juga semakin baik. Top-20 Similarity mendapatkan akurasi terbaik dengan nilai RMSE 0,659217674 nilai tersebut 6,79% lebih baik dari Top-5 Similarity

Tabel 10. Hasil pengujian akurasi sistem dengan metrik RMSE

Top-N Similarity	RMSE
5	0,707284891
10	0,675281266
15	0,671004491
20	0,659217674

3.4.3 Spearman Rank Correlation

Dari hasil pengujian akurasi sistem dengan menggunakan Spearman Rank Correlation, mengikuti skenario pengujian yang dijelaskan sebelumnya pada 2.6. dilakukan pengujian untuk top-N similarity 5, 10, 15, 20. Dengan meningkatkan jumlah nilai similarity yang digunakan untuk menghitung prediksi

rating, korelasi urutan peringkat tabel rekomendasi dan urutan peringkat nilai rating yang diberikan oleh pengguna semakin baik. Top-20 Similarity dan Top-15 Similarity mendapatkan nilai terbaik yaitu 0,91515 yang berarti urutan peringkat dari tabel rekomendasi pengguna memiliki korelasi tinggi dengan urutan peringkat nilai rating yang diberikan oleh pengguna memiliki korelasi tinggi.

Tabel 11. Hasil pengujian akurasi sistem dengan Spearman Rank Correlation

Top-N Similarity	Spearman Rank Correlation
5	0,00606
10	0,491
15	0,91515
20	0,91515

4. Kesimpulan

Sistem Rekomendasi Film Dengan Algoritma Item-Based Collaborative Filtering yang telah dibangun berupa aplikasi web, telah berhasil menerapkan Algoritma Item-Based Collaborative Filtering dengan menggunakan bahasa python, web-framework flask, dan menggunakan sqlite sebagai database. Dari hasil pengujian MAE dan RMSE, yang menggunakan 20 nilai similarity tertinggi untuk menghitung prediksi rating, mendapatkan skor yang cukup baik yaitu nilai MAE sebesar 0,513536 dan RMSE sebesar 0,659217674. Selain itu RMSE dan MAE dapat digunakan bersama untuk mengetahui variance pada kumpulan prediksi, nilai RMSE akan selalu lebih besar atau sama dengan MAE, semakin besar perbedaan antara keduanya maka semakin besar pula variance error untuk setiap individu pada sampel, dari hasil RMSE dan MAE variance error dari implementasi sistem tidak terlalu besar. Untuk pengujian pemberian ranking menggunakan Spearman Rank Correlation pada top-n similarity 15 dan 20 mendapatkan skor sebesar 0,91515 yang berarti peringkat yang diberikan sistem untuk rating prediksi sudah hampir mirip dengan peringkat rating Aktual.

Daftar Pustaka

- [1] I. Dwicahya, P. H. Rosa, and R. Nugroho, "Movie Recommender System Comparison of User-based and Item-based Collaborative Filtering Systems," 2019.
- [2] J. P. Verma, B. Patel, and A. Patel, "Big data analysis: Recommendation system with hadoop framework," 2015.
- [3] M. K. Kharita, A. Kumar, and P. Singh, "Item-Based Collaborative Filtering in Movie Recommendation in Real time," 2018.
- [4] L. Dzumiroh and R. Saptono, "Penerapan Metode Collaborative Filtering Menggunakan Rating Implisit pada Sistem Perekomendasi Pemilihan Film di Rental VCD," *J. Teknol. Inf. ITSmart*, vol. 1, no. 2, 2016.
- [5] P. W. W. Andrew Hans Ritdrix, "Sistem rekomendasi buku menggunakan metode item-based collaborative filtering skripsi," *J. Masy. Inform.*, vol. 9, 2018.
- [6] S. Sari and D. Tri Hendra, "Aplikasi Rekomendasi Film menggunakan Pendekatan Collaborative Filtering dan Euclidean Distance sebagai Ukuran kemiripan Rating," *Semin. Nas. Teknol. Inf. dan Komun. Terap.*, vol. 1, no. 1, 2015.
- [7] V. L. Jaja, B. Susanto, and L. R. Sasongko, "Penerapan Metode Item-Based Collaborative Filtering Untuk Sistem Rekomendasi Data MovieLens," *d'CARTESIAN*, vol. 9, no. 2, 2020.
- [8] M. M. Dewi and I. Artikel, "Optimasi Pearson Correlation untuk Sistem Rekomendasi menggunakan Algoritma Firefly," *J. Inform.*, vol. 9, no. 1, pp. 1–5, 2022,
- [9] B. Prasetyo, H. Haryanto, S. Astuti, E. Z. Astuti, and Y. Rahayu, "Implementasi Metode Item-Based Collaborative Filtering dalam Pemberian Rekomendasi Calon Pembeli Aksesoris Smartphone," *Eksplora Inform.*, vol. 9, no. 1, 2019.
- [10] A. Rosita, N. Puspitasari, and V. Z. Kamila, "REKOMENDASI BUKU PERPUSTAKAAN KAMPUS DENGAN METODE ITEM-BASED COLLABORATIVE FILTERING," *Sebatik*, vol. 26, no. 1, 2022.