

Analisis dan Implementasi Singular Value Decomposition (SVD) untuk Sistem Rekomendasi Anime Berbasis Collaborative Filtering

Muhammad Daffa Ginting*, Muhammad Ikhsan

Fakultas Sains dan Teknologi, Ilmu Komputer, Universitas Islam Negeri Sumatera Utara, Medan, Indonesia

Email: ¹*daffaginting@gmail.com, ²mhdikhsan@uinsu.ac.id

Email Penulis Korespondensi: daffaginting8@gmail.com

Submitted 02-01-2025; Accepted 02-02-2026; Published 25-02-2026

Abstrak

Perkembangan platform digital menyebabkan jumlah anime yang tersedia semakin bertambah, sehingga pengguna sering mengalami kesulitan dalam memilih tontonan yang sesuai dengan preferensi mereka. Di sisi lain, variasi genre serta karakteristik konten yang dapat diakses oleh berbagai kelompok usia menunjukkan perlunya sistem yang mampu memberikan rekomendasi secara lebih tepat dan efisien. Penelitian ini bertujuan untuk membangun sistem rekomendasi anime berbasis collaborative filtering dengan menerapkan metode truncated Singular Value Decomposition (SVD) guna mengatasi permasalahan data sparsity pada matriks pengguna-item. Data diperoleh dari platform AniList yang mencakup 30 pengguna, 100 judul anime, dan 1.230 rating eksplisit. Evaluasi dilakukan menggunakan skema hold-out dengan pembagian 80% data latih (992 rating) dan 20% data uji (248 rating). Kinerja prediksi diukur menggunakan RMSE sebesar 1.6464 dan MAE sebesar 1.2370, sedangkan kualitas rekomendasi Top-N dianalisis melalui Precision@10. Hasil pengujian menunjukkan konfigurasi terbaik berada pada $k = 4$, yang menghasilkan kesalahan prediksi paling rendah pada data uji serta mampu memberikan rekomendasi yang relevan. Nilai rata-rata Precision@10 sebesar 0,1897, yang berarti sekitar 18,97% rekomendasi Top-10 yang diberikan sistem tergolong relevan.

Kata Kunci: Sistem Rekomendasi; Anime; Singular Value Decomposition; Truncated SVD; Precision@N

Abstract

The rapid growth of digital platforms has significantly increased the number of available anime titles, making it difficult for users to choose content that matches their preferences. In addition, the diversity of genres and content characteristics accessible to different age groups highlights the need for a system that can provide recommendations more accurately and efficiently. This study aims to develop an anime recommendation system based on collaborative filtering by applying the truncated Singular Value Decomposition (SVD) method to address the data sparsity problem in the user-item matrix. The dataset was collected from the AniList platform and consists of 30 users, 100 anime titles, and 1,230 explicit rating records. The evaluation was conducted using a hold-out scheme with an 80% training set (992 ratings) and a 20% testing set (248 ratings). Prediction performance was measured using an RMSE of 1.6464 and an MAE of 1.2370, while the quality of Top-N recommendations was assessed using Precision@10. The experimental results indicate that the best configuration was achieved at $k = 4$, which produced the lowest prediction error on the test set and generated relevant recommendations. The average Precision@10 obtained was 0.1897, meaning that approximately 18.97% of the Top-10 recommendations provided by the system were considered relevant.

Keywords: Recommendation System; Anime; Singular Value Decomposition; Truncated SVD; Precision@N

2. PENDAHULUAN

Perkembangan era digital telah mendorong kemudahan akses terhadap berbagai informasi melalui beragam *platform*, termasuk pada sektor industri hiburan. Salah satu bidang yang menunjukkan pertumbuhan pesat adalah anime, yaitu tayangan hiburan asal Jepang yang telah berkembang menjadi fenomena global dengan jangkauan pasar yang luas serta didukung oleh keberagaman *genre* yang dapat dinikmati oleh berbagai kelompok usia [1]. Meskipun memberikan nilai hiburan yang tinggi, pesatnya perkembangan industri anime dan kemudahan akses digital juga berpotensi menimbulkan permasalahan sosial, khususnya bagi anak-anak dan remaja. Sejumlah penelitian mengungkapkan bahwa paparan konten anime yang tidak sesuai dengan usia penonton dapat berdampak pada perilaku, kondisi emosional, serta pola interaksi sosial, seperti meningkatnya kecenderungan perilaku agresif, menurunnya konsentrasi belajar, dan perubahan sikap emosional yang kurang positif [2]. Di sisi lain, setiap tahun jumlah judul anime yang dirilis terus meningkat dengan variasi *genre*, gaya visual, dan kompleksitas alur cerita yang beragam, sehingga memperluas pilihan tontonan bagi pengguna. Kondisi tersebut menegaskan pentingnya penerapan sistem rekomendasi yang mampu menyajikan informasi secara lebih terarah, sehingga konten anime yang ditampilkan menjadi lebih relevan dan sesuai dengan kebutuhan pengguna, khususnya anak-anak dan remaja, berdasarkan minat, preferensi, serta karakteristik usia mereka [3].

Sistem rekomendasi merupakan suatu sistem yang dirancang untuk menyajikan informasi yang relevan atau memprediksi tindakan pengguna dalam mencapai tujuan tertentu. Salah satu metode yang banyak digunakan dalam pengembangan sistem rekomendasi adalah *collaborative filtering*. Metode ini menghasilkan rekomendasi dengan memanfaatkan pola preferensi, minat, dan ketertarikan sejumlah pengguna, yang umumnya direpresentasikan dalam bentuk *rating* yang diberikan pengguna terhadap suatu *item* [4]. Meskipun demikian, pendekatan *collaborative filtering* juga menghadapi sejumlah tantangan teknis, antara lain data *sparsity*, *cold-start*, dan *scalability*. Di antara ketiga tantangan tersebut, data *sparsity* merupakan permasalahan yang paling dominan, khususnya pada *platform* dengan jumlah *item* yang besar dan tingkat partisipasi pengguna yang tidak merata, di mana sebagian besar entri pada matriks pengguna-item tidak terisi akibat pengguna hanya memberikan *rating* pada sebagian kecil dari keseluruhan item yang tersedia [5]. Untuk mengatasi data *sparsity* pada matriks *rating*, penelitian ini menerapkan metode faktorisasi matriks berupa *Singular Value Decomposition* (SVD). *Singular Value Decomposition* (SVD) merupakan metode dalam aljabar linear yang banyak

digunakan dalam pengembangan sistem rekomendasi untuk mereduksi dimensi data. Dalam konteks sistem rekomendasi, SVD digunakan karena kemampuannya dalam mengekstraksi faktor laten dari matriks pengguna-*item*, sehingga dapat mengurangi dampak *data sparsity* dan meningkatkan akurasi prediksi [6].

Penelitian serupa telah dilakukan oleh Shalannanda et al (2022), yang menerapkan algoritma *Singular Value Decomposition* (SVD) pada sistem rekomendasi di domain *e-commerce* untuk menghasilkan rekomendasi produk berdasarkan kemiripan preferensi pengguna. Dataset yang digunakan berasal dari ModCloth Amazon, dengan evaluasi kinerja model menggunakan metrik *Root Mean Square Error* (RMSE). Hasil pengujian menunjukkan bahwa metode SVD memiliki tingkat akurasi terbaik dibandingkan algoritma lain berbasis faktorisasi matriks dengan nilai RMSE sebesar 1,055586. Selain itu, setelah dilakukan optimasi parameter, performa model SVD meningkat dengan nilai RMSE mencapai 1,041784 [7].

Selanjutnya penelitian yang dilakukan oleh Ramadhan & Wibowo (2023), menerapkan algoritma *Singular Value Decomposition* (SVD) pada sistem rekomendasi musik berbasis jumlah pemutaran sebagai pengembangan dari metode *Matrix Factorization*. Sistem rekomendasi tersebut memanfaatkan data jumlah pemutaran musik dari *Million Song Dataset* (MSD) sebagai umpan balik implisit untuk memprediksi preferensi pengguna. Evaluasi kinerja model dilakukan menggunakan metode *k-fold cross-validation* dengan metrik *Root Mean Square Error* (RMSE) dan *Normalized Discounted Cumulative Gain* (NDCG). Hasil evaluasi menunjukkan bahwa konfigurasi parameter optimal dengan nilai $n_epochs = 20$ dan $n_factors = 10$ pada $k = 5$ menghasilkan rata-rata RMSE sebesar 0,4423, serta nilai NDCG@5 dan NDCG@10 masing-masing sebesar 0,8232 dan 0,8231 [8].

Kemudian penelitian yang dilakukan oleh Attalariq & Baizal (2023), mengembangkan sistem rekomendasi buku berbasis *chatbot* dengan menerapkan algoritma *Singular Value Decomposition* (SVD). Sistem rekomendasi ini dirancang untuk berinteraksi dengan pengguna melalui dialog berbasis *natural language* menggunakan *Dialogflow* yang terintegrasi pada platform Telegram. Rekomendasi buku dihasilkan berdasarkan preferensi pengguna yang diproses melalui pendekatan *model-based collaborative filtering*, dengan penghitungan kemiripan menggunakan metode *Cosine Similarity*. Evaluasi akurasi prediksi dilakukan menggunakan metrik *Root Mean Squared Error* (RMSE) dan *Mean Absolute Error* (MAE), yang masing-masing menghasilkan nilai terbaik sebesar 0,8287 dan 0,6481 pada konfigurasi faktor laten optimal. Selain itu, tingkat kepuasan pengguna diukur melalui kuesioner, yang menunjukkan tingkat akurasi rekomendasi sebesar 83,2% serta tingkat kepuasan pengguna secara keseluruhan mencapai 87,9% [9].

Dan Penelitian yang dilakukan oleh Colace et al (2022), mengembangkan sistem rekomendasi berbasis konten dengan menerapkan metode *Singular Value Decomposition* (SVD). Sistem rekomendasi ini dirancang untuk memprediksi *rating item* berdasarkan karakteristik profil pengguna dan *item* tanpa memanfaatkan data *rating* awal. Pengujian sistem dilakukan menggunakan dataset MovieLens100K, dengan hasil evaluasi menunjukkan nilai *Mean Absolute Error* (MAE) sebesar 0,7664 dan *Root Mean Squared Error* (RMSE) sebesar 0,9512 [10].

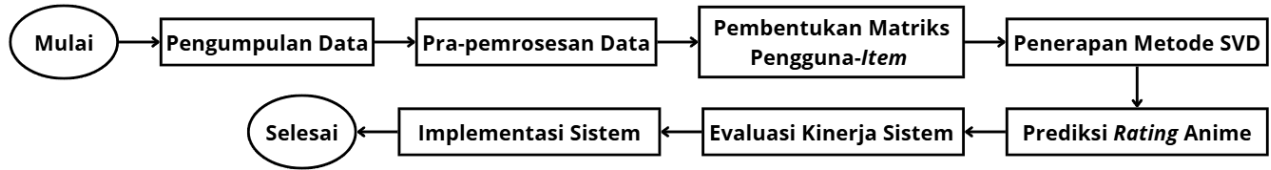
Terakhir penelitian yang dilakukan oleh Siringoringo & Lumbantoruan (2021), mengembangkan sistem rekomendasi dengan mengombinasikan metode *Singular Value Decomposition* (SVD) dan teknik *Pearson Similarity* untuk mengatasi permasalahan sparsitas matriks pada sistem rekomendasi konvensional. Dengan menggunakan dataset *Book-Crossing*, sistem rekomendasi menerapkan proses dekomposisi matriks guna menghasilkan rekomendasi buku yang akurat. Tahapan implementasi meliputi pembentukan matriks utilitas, dekomposisi matriks menjadi matriks fitur pengguna, matriks nilai singular, serta matriks fitur item, yang kemudian diikuti dengan pengukuran tingkat kemiripan antar *item* menggunakan koefisien korelasi Pearson. Hasil penelitian menunjukkan bahwa kombinasi metode SVD dan teknik *Pearson Similarity* mampu menghasilkan rekomendasi yang relevan dan akurat, bahkan ketika dibandingkan dengan sistem rekomendasi pada platform *e-commerce* seperti Amazon [11].

Berdasarkan penelitian terdahulu, metode *Singular Value Decomposition* (SVD) telah banyak digunakan dalam pengembangan sistem rekomendasi pada berbagai domain, seperti *e-commerce*, musik, dan buku, guna meningkatkan akurasi prediksi serta kualitas rekomendasi. Namun, implementasi SVD pada sistem rekomendasi anime masih relatif terbatas, dimana domain anime memiliki karakteristik preferensi pengguna yang cenderung lebih subjektif serta tingkat *data sparsity* yang tinggi pada data *rating* eksplisit. Selain itu, sebagian besar penelitian sebelumnya belum secara spesifik menganalisis pengaruh sparsitas data terhadap penentuan jumlah faktor laten (k) serta implikasinya terhadap performa prediksi. Oleh karena itu, penelitian ini menghadirkan kebaruan melalui penerapan metode *Singular Value Decomposition* (SVD) pada sistem rekomendasi anime berbasis *collaborative filtering* menggunakan dataset AniList, dengan fokus pada penanganan *data sparsity* serta evaluasi kinerja model menggunakan metrik RMSE, MAE, dan Precision@10 untuk menghasilkan rekomendasi anime yang lebih relevan sesuai preferensi pengguna.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan fokus pada pengolahan dan analisis data numerik berupa *rating* yang diberikan pengguna terhadap anime. Tahapan penelitian meliputi pengumpulan data, pra-pemrosesan data, pembentukan matriks pengguna-*item*, penerapan metode *Singular Value Decomposition* (SVD), prediksi *rating* anime, pengukuran akurasi prediksi, serta implementasi sistem. Seluruh tahapan tersebut dilakukan secara berurutan untuk memastikan bahwa setiap proses mendukung pencapaian tujuan penelitian, yaitu menghasilkan sistem rekomendasi anime yang akurat dan relevan, sebagaimana ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

2.2 Pengumpulan Data

Data penelitian diperoleh dari *platform* AniList dengan mengombinasikan teknik *web scraping* dan GraphQL API. Dataset yang digunakan melibatkan 30 pengguna dan 100 judul anime, dengan total 1.230 interaksi pengguna dalam bentuk *rating*. Setiap entri data memuat sebelas atribut utama, yakni *User_id* dan *User_Name* sebagai identitas pengguna, *Anime* sebagai identitas *item*, serta data pelengkap yaitu *Type*, *Genre*, *Year*, *Content_Rating*, *status*, *Total_episode*, *Rating_user* dan *Rating_total*.

2.3 Pra-pemrosesan Data

Pra-pemrosesan merupakan tahap transformasi data mentah yang bertujuan untuk menghasilkan data terstruktur dan layak dianalisis sebelum digunakan pada proses selanjutnya [12]. Data yang diperoleh dari platform AniList melalui GraphQL API umumnya masih dalam bentuk mentah, sehingga perlu dievaluasi untuk mengidentifikasi nilai yang hilang, duplikasi, serta ketidakkonsistenan data. Pada tahap ini dilakukan proses *data cleaning* untuk menangani data yang tidak valid atau tidak relevan, memperbaiki kesalahan struktural, serta memastikan konsistensi data. Data yang telah dibersihkan kemudian ditransformasikan ke dalam bentuk matriks interaksi pengguna-*item* sebagai masukan dalam penerapan algoritma *Singular Value Decomposition* (SVD) [13].

2.4 Pembentukan Matriks Pengguna-Item

Matriks pengguna-*item* dibentuk dari data *rating*, di mana baris merepresentasikan pengguna (*m*) dan kolom merepresentasikan *item* atau *anime* (*n*). Setiap elemen matriks menunjukkan nilai *rating* pengguna terhadap *item*, yang selanjutnya digunakan sebagai *input* pada metode SVD [14]

2.5 Penerapan Metode SVD

a. Normalisasi Matriks

Normalisasi matriks dilakukan sebelum penerapan *Singular Value Decomposition* (SVD) untuk mengatasi perbedaan skala dan ketidakseimbangan data antar pengguna atau *item*. Dengan normalisasi, tingkat *noise* pada setiap baris matriks dapat ditekan sehingga proses ekstraksi faktor laten menjadi lebih optimal [15]. Normalisasi matriks dilakukan menggunakan persamaan berikut:

$$M_j = \frac{1}{|u_j|} \sum_{i \in U_j} A_{ij} \quad (1)$$

Dimana:

- m_j : Rata-rata *rating* untuk *item* ke-*j* (anime-*j*).
- $|u_j|$: jumlah pengguna yang memberikan *rating* pada *item* *j*.
- U_j : Himpunan pengguna yang memberi *rating* pada *item* *j*.
- A_{ij} : *Rating* yang diberikan oleh pengguna *i* terhadap *item* *j*.

b. Singular Value Decomposition (SVD)

Berbagai algoritma dapat digunakan dalam pengembangan sistem rekomendasi, salah satunya adalah *Singular Value Decomposition* (SVD). *Singular Value Decomposition* adalah algoritma yang digunakan untuk menganalisis matriks riil dan kompleks, serta membantu dalam memahami hubungan dan pola yang terdapat dalam data [16]. SVD merupakan metode berbasis aljabar linier yang banyak dimanfaatkan dalam sistem rekomendasi karena kemampuannya dalam mereduksi dimensi data dan mengekstraksi pola laten. Dalam konteks *collaborative filtering*, SVD digunakan sebagai teknik faktorisasi matriks, di mana data *rating* direpresentasikan dalam bentuk matriks. Baris pada matriks merepresentasikan pengguna, sedangkan kolom merepresentasikan *item*, dan setiap elemen matriks menunjukkan nilai *rating* yang diberikan pengguna terhadap *item* tertentu. Proses faktorisasi matriks ini bertujuan untuk memetakan hubungan antara pengguna dan *item* ke dalam ruang dimensi yang lebih rendah sehingga sistem dapat menghasilkan rekomendasi secara lebih efektif [17]. Pada metode *Singular Value Decomposition* (SVD), matriks pengguna-*item* direpresentasikan sebagai hasil dekomposisi menjadi tiga matriks utama, yaitu matriks pengguna, matriks nilai singular, dan matriks *item* [18]. Proses dekomposisi tersebut secara matematis dinyatakan dalam bentuk $A = U \Sigma V^T$. Persamaannya sebagai berikut:

$$A = \sum_{i=1}^n \sigma_i U_i V_i^t \quad (2)$$

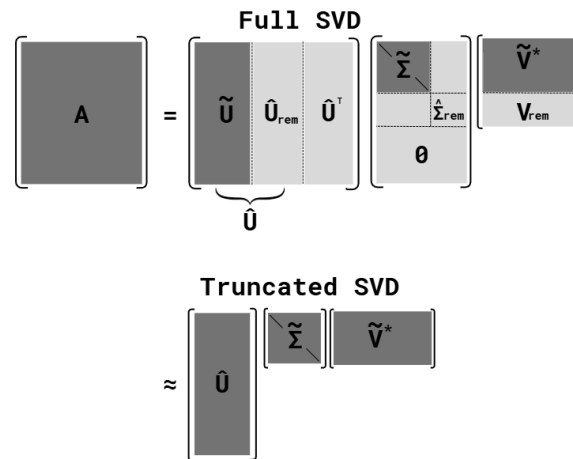
Dimana:

U_i : Matriks ortogonal berukuran $m \times m$ yang kolom-kolomnya adalah vektor singular kiri.
 V_i^t : Matriks ortogonal berukuran $n \times n$ yang kolom-kolomnya adalah vektor singular kanan.
 Σ : Matriks diagonal berukuran $m \times n$ yang berisi nilai-nilai singular berupa bilangan riil tak negatif, yang dinyatakan sebagai $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$ dengan urutan $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$.

Namun, penggunaan full SVD dengan seluruh komponen singular kurang efisien dan berpotensi memasukkan *noise* pada data, sehingga pada penelitian ini digunakan pendekatan *truncated Singular Value Decomposition*.

c. Truncated Singular Value Decomposition

Truncated Singular Value Decomposition (SVD) adalah metode aproksimasi matriks ber-rank rendah yang diperoleh dengan mempertahankan r nilai singular terbesar beserta vektor singular yang bersesuaian dan membuang sisanya, sehingga menghasilkan aproksimasi matriks dengan galat rekonstruksi minimum dalam norma Frobenius [19]. Perbedaan antara *full* SVD dan *truncated* SVD tersebut dapat dilihat pada gambar 2.



Gambar 2. Perbandingan *Full* SVD dengan *Truncated* SVD

Gambar 2 memperlihatkan perbedaan antara *full* Singular Value Decomposition (SVD) dan *truncated* SVD. Pada *full* SVD, matriks A didekomposisi menjadi seluruh komponen vektor singular kiri $\tilde{U}$, nilai singular $\tilde{\Sigma}$, dan vektor singular kanan $\tilde{V}^*$, termasuk komponen sisa (*remainder*). Sebaliknya, pada *truncated* SVD hanya k nilai singular terbesar beserta vektor singular yang bersesuaian yang dipertahankan, sementara komponen sisa dibuang, sehingga diperoleh aproksimasi matriks ber-rank rendah dari A . Persamaan *Truncated* SVD sebagai berikut:

$$A \approx A_k = U_k \Sigma_k V_k^T \quad (3)$$

Dimana:

A : Matriks data asli.
 A_r : Aproksimasi ber-rank r dari matriks A .
 U_r : Berisi k vektor singular kiri utama.
 Σ_r : Matriks diagonal yang memuat r nilai singular.
 V_r^T : Berisi k vektor singular kanan.

d. Penentuan Jumlah Faktor Laten (k)

Pada metode *truncated* SVD, jumlah faktor laten (k) merupakan parameter penting yang menentukan jumlah nilai singular yang dipertahankan dalam proses dekomposisi matriks. Penentuan nilai k dilakukan menggunakan pendekatan *cumulative energy* untuk menjaga sebagian besar informasi matriks. Nilai k dapat dipilih ketika persentase variansi kumulatif telah melampaui ambang batas tertentu, umumnya berada pada rentang 70% hingga 90% [20]. Persamaanya sebagai berikut:

$$\frac{\sum_{i=1}^k \sigma_i^2}{\sum_{i=1}^r \sigma_i^2} > t^* \quad (4)$$

Dimana:

σ_i : Nilai singular ke- i yang diperoleh dari hasil dekomposisi SVD.
 σ_i^2 : Energi atau kontribusi variansi dari nilai singular ke- i .
 k : Jumlah faktor laten atau banyaknya nilai singular terbesar yang dipertahankan pada proses *truncated* SVD.
 r : Rank maksimum dari matriks pengguna-item.
 $\sum_{i=1}^k \sigma_i^2$: Total energi parsial yang diwakili oleh nilai k singular terbesar.
 $\sum_{i=1}^r \sigma_i^2$: Total energi keseluruhan matriks.
 t^* : *Threshold* energi kumulatif yang ditetapkan, dengan nilai umumnya berada pada rentang 70% - 90%.

2.6 Evaluasi Kinerja Sistem

Evaluasi kinerja sistem rekomendasi bertujuan untuk menilai kemampuan sistem dalam menghasilkan prediksi *rating* serta rekomendasi *Top-N* anime yang sesuai dengan preferensi pengguna. Pada penelitian ini, proses evaluasi dilakukan menggunakan skema *hold-out* dengan membagi data *rating* ke dalam data latih dan data uji. Dari total 1.240 data *rating* yang terdapat pada matriks pengguna-*item*, sebanyak 992 *rating* (sekitar 80%) digunakan sebagai data latih untuk membangun model, sedangkan 248 *rating* (sekitar 20%) digunakan sebagai data uji. Pengukuran kinerja dilakukan menggunakan data uji, di mana performa sistem direpresentasikan oleh kinerja model SVD sebagai komponen utama dalam menghasilkan rekomendasi. Untuk mengukur akurasi prediksi *rating*, digunakan metrik *Mean Absolute Error* (MAE) dan *Root Mean Squared Error* (RMSE). MAE digunakan untuk mengukur rata-rata kesalahan absolut antara nilai prediksi dan nilai aktual, Nilai MAE yang lebih kecil menunjukkan bahwa model memiliki tingkat akurasi yang lebih baik [21]. Perhitungan MAE dinyatakan menggunakan persamaan sebagai berikut:

$$MAE = \frac{1}{|\hat{R}|} \sum_{r_{ui} \in \hat{R}} |r_{ui} - \hat{r}_{ui}| \quad (5)$$

Dimana:

- r_{ui} : Nilai aktual yang diberikan oleh pengguna u terhadap *item* i .
- $\hat{r}_{ui}$: Nilai prediksi yang diberikan oleh pengguna u terhadap *item* i .
- $\hat{R}$: Merepresentasikan total data *point* yang diprediksi oleh model.

RMSE digunakan untuk mengukur besarnya kesalahan prediksi, di mana nilai yang lebih kecil menunjukkan performa model yang lebih baik [22]. Perhitungan RMSE dinyatakan menggunakan persamaan sebagai berikut:

$$RMSE = \sqrt{\frac{1}{|\hat{R}|} \sum_{r_{ui} \in \hat{R}} (r_{ui} - \hat{r}_{ui})^2} \quad (6)$$

Dimana:

- r_{ui} : Nilai aktual yang diberikan oleh pengguna u terhadap *item* i .
- $\hat{r}_{ui}$: Nilai prediksi yang diberikan oleh pengguna u terhadap *item* i .
- $\hat{R}$: Merepresentasikan total data *point* yang diprediksi oleh model.

Selain itu, kualitas rekomendasi *Top-N* anime dievaluasi menggunakan metrik *Precision@N*. *Precision@N* digunakan untuk mengukur proporsi *item* relevan pada daftar rekomendasi *Top-N* yang dihasilkan sistem [23]. Perhitungan *Precision@N* dinyatakan menggunakan persamaan sebagai berikut:

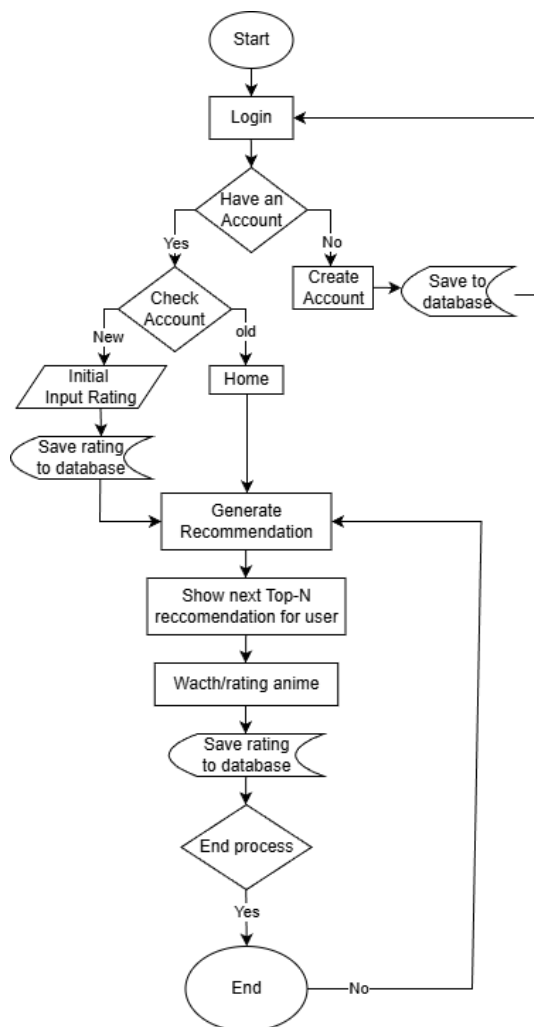
$$Precision@N = \frac{r}{n} \quad (7)$$

Dimana:

- r : Jumlah *item* relevan yang terdapat dalam N rekomendasi teratas.
- n : Jumlah keseluruhan *item* yang direkomendasikan

2.7 Implementasi Sistem

Implementasi sistem dilakukan dengan menerapkan model sistem rekomendasi anime berbasis (SVD) ke dalam sebuah aplikasi berbasis web. Aplikasi ini dirancang untuk menampilkan hasil rekomendasi anime kepada pengguna berdasarkan nilai prediksi *rating* yang dihasilkan oleh model. Sistem berperan sebagai antarmuka yang menghubungkan proses perhitungan model dengan penyajian rekomendasi kepada pengguna. Alur kerja sistem secara keseluruhan ditunjukkan pada gambar 3.



Gambar 3. Alur Flowchart Sistem

3. HASIL DAN PEMBAHASAN

3.1 Representase Data

Pada penelitian ini, Dataset yang digunakan terdiri dari 30 pengguna dan 100 judul anime, dengan total 1.230 interaksi berupa data *rating* sedangkan sisanya merupakan data kosong (*missing value*) dengan keseluruhan total sebanyak 3000 data. Setiap data *rating* memiliki tujuh atribut, yaitu *User_id*, *User_Name*, *Anime*, *Type*, *Genre*, *Year*, dan *Rating*. Data tersebut menjadi data awal sebelum dilakukan tahap pra-pemrosesan data. Struktur dan karakteristik dataset dapat dilihat pada Tabel 1.

Tabel 1. Dataset *Rating*

No	User_id	User_name	Anime	Type	Genre	...	Rating User	Rating Total
1	5448630	Bruhgreenbean	Kimetsu no Yaiba	TV	Action, Adventure, Drama, Fantasy, Supernatural	...	8.4	8.42
2	5448630	Bruhgreenbean	Jujutsu Kaisen	TV	Action, Drama, Supernatural	...	8.0	8.52
3	5448630	Bruhgreenbean	Shingeki no Kyojin	TV	Action, Drama, Fantasy, Mystery	...	7.8	8.57
4	5448630	Bruhgreenbean	Chainsaw Man	TV	Action, Drama, Horror, Supernatural	...	8.7	8.43
5	5448630	Bruhgreenbean	Vinland Saga	TV	Action, Adventure, Drama	...	-	-
...						...		
2999	6418684	aktivn	Ao no Exorcist	TV	Action, Fantasy, Supernatural	...	-	7.47
3000	6418684	aktivn	Haikyuu	TV	Comedy, Drama, Sports	...	-	8.43

Pada tabel 1, menampilkan data historis interaksi pengguna terhadap anime berukuran 30×100 dengan total 3.000 kemungkinan entri. Sebelum digunakan dalam proses pemodelan sistem rekomendasi, dilakukan tahap pra-pemrosesan data termasuk penghapusan atribut yang tidak berkontribusi langsung terhadap pemodelan seperti *User_Name*, *Type*, *Genre*, *Year*, *Content_Rating*, *status*, *Total_episode*, dan *Rating_total*.

3.2 Pra-pemrosesan Data

Pada tahap pra-pemrosesan, dilakukan proses pembersihan data dengan menghilangkan atribut pendukung yang tidak berpengaruh langsung terhadap pemodelan, serta menangani duplikasi *rating* yang terdapat pada pasangan pengguna-*item* yang sama. Selanjutnya, data yang telah dibersihkan diseleksi dan disusun kembali untuk memastikan konsistensi serta kesiapan data sebelum dilakukan normalisasi. data hasil pra-pemrosesan dapat dilihat pada tabel 2.

Tabel 2. Dataset Setelah Pra-pemrosesan

No	User_id	86: eighty six	akame ga kill!	akudama drive	...	yofukashi no uta	youkoso jitsuryoku shijou shugi no kyoushitsu e
1	30949	-	1	-	...	-	-
2	97248	-	5	-	...	8	-
3	155475	3	3	-	...	5	-
4	186132	8.6	9.5	7.5	...	8.5	8.7
5	214861	-	-	-	...	8	-
...	...	...	...	...	...	...	...
29	6698347	6	8	-	...	7	5
30	7424780	7	7	-	...	10	5

Pada tabel 2, menampilkan matriks interaksi pengguna-*item* yang diperoleh setelah tahap pra-pemrosesan data. Setiap baris merepresentasikan pengguna dan setiap kolom merepresentasikan item, sedangkan nilai pada setiap sel menunjukkan *rating* yang diberikan pengguna terhadap *item* tertentu. Sel yang tidak terisi menunjukkan *rating* yang tidak terobservasi dan direpresentasikan sebagai nilai hilang (*missing values*). Data tersebut selanjutnya ditransformasikan ke dalam bentuk matriks interaksi pengguna-*item* *A*.

3.3 Pembentukan Matriks Pengguna-Item

Setelah data melalui seluruh tahapan pra-pemrosesan, langkah selanjutnya adalah membentuk matriks interaksi pengguna-*item* yang merepresentasikan hubungan antara pengguna dan *item* berdasarkan nilai *rating* yang diberikan. Persamaannya sebagai berikut:

$$A = \begin{bmatrix} - & 1 & - & - & - & \dots & - & - \\ - & 5 & - & - & - & \dots & 8 & - \\ 3 & 3 & - & 3 & 2 & \dots & 5 & - \\ 8.6 & 9.5 & 7.5 & 8 & 8 & \dots & 8.5 & 8.7 \\ - & - & - & - & - & \dots & 8 & - \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 6 & 8 & - & 5 & 6 & \dots & 7 & 5 \\ 7 & 7 & - & 10 & 5 & \dots & 10 & 5 \end{bmatrix}$$

Matriks diatas disusun dengan baris merepresentasikan pengguna dan kolom merepresentasikan item, sedangkan setiap elemen matriks menunjukkan nilai penilaian pengguna terhadap item tertentu.

3.4 Penerapan SVD

a. Normalisasi Matriks

Setelah data dibersihkan, selanjutnya dilakukan proses normalisasi matriks untuk menangani *missing value*, perbedaan skala penilaian, serta ketidakseimbangan data antar pengguna maupun antar *item*. Hasil yang diperoleh setelah proses normalisasi pada persamaan berikut:

$$A = \begin{bmatrix} 2.5500 & -4.0000 & 3.0570 & 2.6710 & 1.2560 & \dots & 2.5330 & 1.7750 \\ -0.4190 & -2.9690 & 0.0880 & -0.2970 & -1.7130 & \dots & 0.0310 & -1.1940 \\ -1.4930 & -1.4930 & 3.5640 & -1.4930 & -2.4930 & \dots & 0.5070 & 2.2820 \\ 0.8180 & 1.7180 & -0.2820 & 0.2180 & 0.2180 & \dots & 0.7180 & 0.9180 \\ -0.4680 & -1.8620 & 0.0390 & -0.3460 & -1.7620 & \dots & -0.0180 & -1.2430 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ -1.2800 & 0.7200 & 0.7770 & -2.2800 & -1.2800 & \dots & -0.2800 & -2.2800 \\ 0.2920 & 0.2920 & 3.2920 & 3.2920 & -1.7080 & \dots & 3.2920 & -1.7080 \end{bmatrix}$$

Pada matriks diatas, Nilai pada setiap baris menunjukkan *rating* yang telah dinormalisasi untuk masing-masing pengguna terhadap *item* tertentu. Proses normalisasi ini bertujuan untuk menghilangkan perbedaan skala penilaian antar

pengguna serta mengurangi bias penilaian, sehingga data yang dihasilkan memiliki rata-rata yang sebanding dan dapat digunakan secara efektif pada tahap pemodelan selanjutnya.

b. Singular Value Decomposition (SVD)

Pada tahap SVD matriks A yang telah dinormalisasikan akan didekomposisi menjadi tiga matriks, yaitu matriks ortogonal U , matriks diagonal Σ , dan matriks ortogonal V^T . Untuk memperoleh matriks U , dengan menentukan vektor eigen dari matriks AA^T , nilai eigen dapat diperoleh dengan menyelesaikan persamaan karakteristik $\det(\lambda I - AA^T) = 0$. Dikarenakan ukuran matriks A 30 x 30 maka, nilai eigen yang diperoleh sebanyak 30. Himpunan nilai eigen adalah $\lambda = \{2459.8830, 1688.7682, 1688.7682, 447.9510, 401.2640 \dots, 5.6953, 2.2650\}$. Selanjutnya, matriks U berukuran $m \times m$ dibentuk dari vektor-vektor eigen yang bersesuaian dengan nilai eigen, sebagaimana ditunjukkan pada persamaan berikut:

$$U = \begin{bmatrix} -0.0746 & -0.5301 & -0.0675 & 0.1439 & -0.2211 & \dots & 0.0506 & 0.0419 \\ 0.1761 & -0.0516 & 0.0347 & -0.0091 & -0.0896 & \dots & 0.0844 & 0.0164 \\ 0.0618 & -0.3909 & 0.6797 & -0.4988 & 0.1716 & \dots & 0.0103 & 0.0403 \\ 0.1463 & -0.0414 & -0.1478 & -0.0603 & -0.5522 & \dots & 0.0348 & 0.0331 \\ 0.2011 & -0.0147 & 0.0154 & -0.0384 & 0.0163 & \dots & 0.0781 & 0.0716 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0.0730 & -0.0232 & 0.0536 & -0.0455 & -0.3302 & \dots & 0.0220 & 0.0486 \\ 0.1809 & -0.1598 & 0.3371 & 0.8159 & 0.1660 & \dots & -0.0010 & 0.0209 \end{bmatrix}$$

Selanjutnya, nilai singular pada matriks Σ diperoleh dari akar kuadrat nilai eigen yang dihasilkan dari matriks AA^T . Nilai singular tersebut disusun secara menurun pada diagonal utama matriks Σ berukuran 30 x 30, sehingga dapat dilihat pada persamaan berikut:

$$\Sigma = \begin{bmatrix} 49.5972 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 41.0946 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 21.1649 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 20.0316 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & 0 & \dots & 2.3867 & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1.5050 \end{bmatrix}$$

Setelah matriks U dan Σ diperoleh, matriks vektor singular kanan V dihitung menggunakan hubungan: $V = A^T U \Sigma^{-1}$. Kemudian matriks V ditranposekan sehingga menjadi matriks V^T dengan berukuran 30 x 100, sehingga dapat dilihat pada persamaan berikut:

$$V^T = \begin{bmatrix} -0.0497 & -0.1716 & -0.0017 & -0.0235 & -0.1840 & \dots & -0.0380 & -0.1359 \\ -0.0924 & 0.0571 & -0.1383 & -0.0624 & -0.0256 & \dots & -0.0471 & -0.0852 \\ -0.1587 & -0.0404 & 0.0699 & -0.0440 & -0.1267 & \dots & 0.1453 & 0.0034 \\ 0.0676 & 0.0480 & 0.0698 & 0.2089 & 0.0416 & \dots & 0.1658 & -0.0858 \\ -0.0013 & -0.0045 & 0.0844 & -0.0165 & -0.0682 & \dots & -0.1576 & 0.0271 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ -0.0233 & 0.0753 & 0.0205 & -0.0226 & 0.1344 & \dots & 0.0780 & 0.0668 \\ -0.1171 & 0.0612 & -0.0244 & -0.0324 & 0.0205 & \dots & -0.0412 & 0.1122 \end{bmatrix}$$

Maka matriks A dapat direpresentasikan kembali menggunakan persamaan berikut $A = U \Sigma V^T$, Sehingga membentuk persamaan:

$$A = \begin{bmatrix} -0.0746 & -0.5301 & -0.0675 & 0.1439 & -0.2211 & \dots & 0.0506 & 0.0419 \\ 0.1761 & -0.0516 & 0.0347 & -0.0091 & -0.0896 & \dots & 0.0844 & 0.0164 \\ 0.0618 & -0.3909 & 0.6797 & -0.4988 & 0.1716 & \dots & 0.0103 & 0.0403 \\ 0.1463 & -0.0414 & -0.1478 & -0.0603 & -0.5522 & \dots & 0.0348 & 0.0331 \\ 0.2011 & -0.0147 & 0.0154 & -0.0384 & 0.0163 & \dots & 0.0781 & 0.0716 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0.0730 & -0.0232 & 0.0536 & -0.0455 & -0.3302 & \dots & 0.0220 & 0.0486 \\ 0.1809 & -0.1598 & 0.3371 & 0.8159 & 0.1660 & \dots & -0.0010 & 0.0209 \end{bmatrix}$$

$$x \begin{bmatrix} 49.5972 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 41.0946 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 21.1649 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 20.0316 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & 0 & \dots & 2.3867 & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1.5050 \end{bmatrix}$$

$$x \begin{bmatrix} -0.0497 & -0.1716 & -0.0017 & -0.0235 & -0.1840 & \dots & -0.0380 & -0.1359 \\ -0.0924 & 0.0571 & -0.1383 & -0.0624 & -0.0256 & \dots & -0.0471 & -0.0852 \\ -0.1587 & -0.0404 & 0.0699 & -0.0440 & -0.1267 & \dots & 0.1453 & 0.0034 \\ 0.0676 & 0.0480 & 0.0698 & 0.2089 & 0.0416 & \dots & 0.1658 & -0.0858 \\ -0.0013 & -0.0045 & 0.0844 & -0.0165 & -0.0682 & \dots & -0.1576 & 0.0271 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ -0.0233 & 0.0753 & 0.0205 & -0.0226 & 0.1344 & \dots & 0.0780 & 0.0668 \\ -0.1171 & 0.0612 & -0.0244 & -0.0324 & 0.0205 & \dots & -0.0412 & 0.1122 \end{bmatrix}$$

Matriks di atas merepresentasikan hasil dekomposisi *Singular Value Decomposition* (SVD) dari matriks pengguna-item A, yang dinyatakan sebagai perkalian antara matriks vektor singular kiri U, matriks diagonal nilai singular Σ , dan matriks vektor singular kanan V^T . Matriks U memuat vektor singular kiri yang merepresentasikan pola laten pengguna, matriks Σ berisi nilai singular yang menunjukkan tingkat kepentingan masing-masing komponen, sedangkan matriks V^T merepresentasikan pola laten item.

c. *Truncated Singular Value Decomposition*

Pada tahap ini, *truncated* SVD digunakan untuk membentuk model sistem rekomendasi dengan mempertahankan sejumlah nilai singular terbesar (k). Proses ini menghasilkan representasi laten berdimensi rendah yang menggambarkan preferensi pengguna dan karakteristik item. Berbeda dengan *full* SVD yang menggunakan seluruh komponen, *truncated* SVD hanya mempertahankan komponen utama (k) yang merepresentasikan pola interaksi pengguna-item. Penentuan nilai k dilakukan menggunakan pendekatan *cumulative energy* untuk memastikan sebagian besar informasi pada matriks tetap dipertahankan. Hasil perhitungan *cumulative energy* dapat dilihat pada tabel 3.

Tabel 3. Hasil Perhitungan Energi Kumulatif (k)

No	k	σ^2 (Energi parsial)	Energi Kumulatif	Presentase (%)
1	1	2460.86	2460.86	36.70
2	2	1688.77	4149.63	61.89
3	3	447.95	4597.58	68.57
4	4	401.26	4998.84	74.56
5	5	252.00	5250.84	78.32
6	6	212.32	5463.16	81.49
7	7	151.31	5614.47	83.74
8	8	130.66	5745.13	85.69
9	9	121.53	5866.66	87.50
10	10	108.00	5974.66	89.12
11	11	88.46	6063.12	90.44
12	12	84.37	6147.49	91.69
13	13	79.08	6226.57	92.87
14	14	67.41	6293.98	93.88
15	15	56.47	6350.45	94.72
16	16	53.78	6404.23	95.52
17	17	48.79	6453.02	96.25
18	18	41.72	6494.74	96.87
19	19	32.02	6526.76	97.35
20	20	29.21	6555.97	97.79
21	21	26.94	6582.91	98.19
22	22	22.40	6605.31	98.52
23	23	21.91	6627.22	98.85
24	24	18.69	6645.91	99.13
25	25	17.07	6662.98	99.38
26	26	12.59	6675.57	99.57
27	27	11.29	6686.86	99.74
28	28	9.54	6696.40	99.88
29	29	5.69	6702.09	99.97

30	30	2.27	6704.36	100.00
----	----	------	---------	--------

Tabel 3 menyajikan nilai energi parsial yang diperoleh dari kuadrat nilai singular (σ^2), beserta energi kumulatif dan persentase variansi yang dijelaskan oleh masing-masing komponen. Berdasarkan tabel tersebut, dapat dilihat bahwa energi kumulatif meningkat seiring dengan bertambahnya jumlah komponen k . Pada penelitian ini, ambang batas variansi kumulatif ditetapkan sebesar 85%, yang dipilih sebagai nilai kompromi dalam rentang 70–90% yang umum digunakan. Pada nilai $k = 8$, persentase variansi kumulatif telah mencapai 85,69%, yang menunjukkan bahwa sebagian besar informasi utama pada matriks telah berhasil dipertahankan. Nilai $k = 8$ digunakan sebagai kandidat awal berdasarkan analisis energi kumulatif untuk menggambarkan penerapan metode *truncated SVD*.

Setelah k ditentukan, selanjutnya dilakukan proses *truncation* pada hasil dekomposisi SVD dengan mempertahankan k nilai singular terbesar beserta vektor singular yang bersesuaian. Pada tahap ini, Untuk matriks U , menggunakan delapan komponen utama berupa vektor singular kiri yang bersesuaian dengan delapan nilai singular terbesar, sebagaimana ditunjukkan pada persamaan berikut:

$$U_8 = \begin{bmatrix} -0.0746 & -0.5301 & -0.0675 & 0.1439 & -0.2211 & 0.3161 & -0.1435 & 0.5714 \\ 0.1761 & -0.0516 & 0.0347 & -0.0091 & -0.0896 & 0.1957 & -0.1388 & -0.0976 \\ 0.0618 & -0.3909 & 0.6797 & -0.4988 & 0.1716 & -0.2085 & 0.0384 & 0.0426 \\ 0.1463 & -0.0414 & -0.1478 & -0.0603 & -0.5522 & -0.4649 & -0.0688 & -0.1638 \\ 0.2011 & -0.0147 & 0.0154 & -0.0384 & 0.0163 & -0.0347 & -0.1461 & 0.0529 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0.0730 & -0.0232 & 0.0536 & -0.0455 & -0.3302 & -0.2852 & -0.1479 & 0.2933 \\ 0.1809 & -0.1598 & 0.3371 & 0.8159 & 0.1660 & -0.2389 & 0.0956 & -0.0023 \end{bmatrix}$$

Selanjutnya untuk matriks Σ , delapan nilai singular terbesar yang disusun pada diagonal utama matriks, persamaanya sebagai berikut:

$$\Sigma_8 = \begin{bmatrix} 49.5972 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 41.0946 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 21.1649 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 20.0316 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 15.8736 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Kemudian matriks V^T , delapan komponen utama berupa vektor singular kanan yang bersesuaian dengan delapan nilai singular terbesar, sehingga persamaannya sebagai berikut:

$$V_8^T = \begin{bmatrix} -0.0497 & -0.1716 & -0.0017 & -0.0235 & -0.1840 & -0.2244 & -0.1205 & -0.0477 \\ -0.0924 & 0.0571 & -0.1383 & -0.0624 & -0.0256 & -0.0123 & -0.0336 & -0.1282 \\ -0.1587 & -0.0404 & 0.0699 & -0.0440 & -0.1267 & 0.1654 & -0.1603 & 0.0133 \\ 0.0676 & 0.0480 & 0.0698 & 0.2089 & 0.0416 & 0.0098 & -0.0148 & -0.0287 \\ -0.0013 & -0.0045 & 0.0844 & -0.0165 & -0.0682 & -0.0552 & -0.0760 & 0.0282 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ -0.0233 & 0.0753 & 0.0205 & -0.0226 & 0.1344 & 0.0038 & 0.0636 & 0.1382 \\ -0.1171 & 0.0612 & -0.0244 & -0.0324 & 0.0205 & 0.0659 & -0.0189 & 0.1129 \end{bmatrix}$$

Matriks U_8 , Σ_8 , dan V_8^T yang diperoleh selanjutnya digunakan untuk merekonstruksi matriks pengguna-item melalui persamaan *truncated SVD*. Maka persamaannya sebagai berikut:

$$A_k = \begin{bmatrix} 2.3726 & -3.5649 & 3.1000 & 2.6958 & 1.1084 & \dots & 2.6815 & 1.3476 \\ -0.4428 & -2.4815 & -0.2499 & 0.0296 & -1.6308 & \dots & 0.3593 & -1.0707 \\ -1.5362 & -1.4518 & 3.2134 & -2.0473 & -2.5424 & \dots & 0.3939 & 2.0438 \\ 0.4753 & 1.8217 & -0.0685 & -0.6443 & -0.1877 & \dots & -0.0113 & -0.5094 \\ -0.5353 & -1.6928 & 0.2997 & -0.4380 & -2.0787 & \dots & -0.3795 & -1.2266 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ -0.2380 & 0.1853 & 0.7966 & -0.4421 & -0.7969 & \dots & 0.6483 & -0.4787 \\ 0.2460 & -0.0959 & 3.1740 & 3.0089 & -1.7572 & \dots & 2.9523 & -1.8055 \end{bmatrix}$$

Setelah matriks direkonstruksi, langkah selanjutnya adalah mengembalikan hasil aproksimasi ke skala *rating* asli dengan menambahkan kembali nilai rata-rata *rating* masing-masing pengguna. Proses ini bertujuan agar nilai prediksi yang dihasilkan dapat diinterpretasikan sesuai dengan skala penilaian awal. Persamaannya sebagai berikut:

$$A = \begin{bmatrix} 7.3726 & 1.4351 & 8.1000 & 7.6958 & 6.1084 & \dots & 7.6815 & 6.3476 \\ 7.5260 & 5.4873 & 7.7188 & 7.9984 & 6.3379 & \dots & 8.3280 & 6.8980 \\ 2.9566 & 3.0409 & 7.7062 & 2.4454 & 1.9504 & \dots & 4.8867 & 6.5365 \\ 8.2569 & 9.6034 & 7.7131 & 7.1373 & 7.5939 & \dots & 7.7703 & 7.2722 \\ 7.4826 & 6.3251 & 7.7131 & 7.5799 & 5.9392 & \dots & 7.6383 & 6.7913 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 7.0425 & 7.4657 & 8.0771 & 6.8384 & 6.4836 & \dots & 7.9288 & 6.8018 \\ 6.9539 & 6.6120 & 9.8819 & 9.7168 & 4.9507 & \dots & 9.6601 & 4.9024 \end{bmatrix}$$

Proses pengembalian hasil aproksimasi ke skala *rating* asli dilakukan dengan menambahkan kembali nilai rata-rata *rating* setiap pengguna, sehingga diperoleh matriks prediksi yang merepresentasikan estimasi penilaian pengguna terhadap *item* yang belum terobservasi.

3.5 Evaluasi Kinerja Sistem

a. Akurasi Prediksi Model *Truncated SVD*

Dalam evaluasi akurasi prediksi *rating*, nilai k pada model *truncated Singular Value Decomposition* merepresentasikan jumlah komponen laten yang dipertahankan dalam proses rekonstruksi matriks prediksi. Oleh karena itu, kinerja model dievaluasi menggunakan metrik *Root Mean Squared Error* (RMSE) dan *Mean Absolute Error* (MAE) untuk mengukur tingkat kesalahan prediksi yang dihasilkan pada nilai k yang telah ditentukan. Setelah diperoleh nilai $k = 8$ berdasarkan analisis energi kumulatif, tahap selanjutnya adalah melakukan uji validasi untuk menilai kemampuan model *truncated SVD* dalam melakukan generalisasi terhadap data yang tidak terlihat secara langsung (*unseen data*). Hasil evaluasi akurasi prediksi dapat dilihat pada tabel 4.

Tabel 4. Hasil Akurasi Prediksi Model *Truncated SVD* ($k = 8$)

Data	Jumlah <i>Rating</i>	RMSE	MAE
Data Latih	992	0.7916	0.5662
Data Uji	248	1.7452	1.3062

Berdasarkan Tabel 4, diperoleh nilai RMSE sebesar 0,7916 pada data latih dan 1,7452 pada data uji, sedangkan nilai MAE sebesar 0,5662 pada data latih dan 1,3062 pada data uji. Nilai RMSE dan MAE yang lebih rendah pada data latih menunjukkan bahwa model mampu mempelajari pola interaksi pengguna-*item* dengan baik.

Peningkatan nilai RMSE dan MAE pada data uji dibandingkan dengan data latih menunjukkan bahwa kesalahan prediksi cenderung meningkat ketika model diterapkan pada data yang tidak digunakan selama proses pelatihan. Untuk memastikan bahwa kondisi tersebut tidak disebabkan oleh pemilihan parameter yang kurang optimal, dilakukan evaluasi terhadap beberapa nilai k yang berada dalam rentang energi kumulatif yang direkomendasikan. Berdasarkan hasil evaluasi tersebut, pengujian lanjutan dilakukan dengan mencoba beberapa nilai k di sekitar kandidat awal, seperti $k = 4, 5, 6, 7$ dan 9. Hasil dari evaluasi beberapa k dapat dilihat pada tabel 5.

Tabel 5. Hasil Evaluasi Model *Truncated SVD* untuk Berbagai Nilai k

k	RMSE Latih	MAE Latih	RMSE Uji	MAE Uji
4	1.0996	0.8069	1.6464	1.2370
5	1.0188	0.7300	1.6918	1.2361
6	0.9225	0.6553	1.7037	1.2623
7	0.8514	0.6102	1.7472	1.3044
9	0.7343	0.5258	1.7495	1.3142

Tabel 5 menyajikan hasil evaluasi model *truncated SVD* pada berbagai nilai k . Berdasarkan hasil tersebut, diperoleh bahwa nilai $k = 4$ menghasilkan nilai RMSE dan MAE terendah pada data uji. Hal ini menunjukkan bahwa penggunaan empat komponen laten memberikan keseimbangan yang paling optimal antara kemampuan model dalam mempelajari pola data dan kemampuan generalisasi terhadap data yang tidak digunakan selama proses pelatihan. Oleh karena itu, nilai $k = 4$ ditetapkan sebagai parameter akhir dalam pembangunan model *truncated SVD* pada penelitian ini.

b. Evaluasi Kualitas Rekomendasi *Top-N*

Selain evaluasi akurasi prediksi *rating*, kualitas rekomendasi yang dihasilkan oleh sistem juga perlu dievaluasi. Evaluasi ini dilakukan menggunakan metrik *Precision@N*, yang mengukur proporsi *item* relevan dalam daftar rekomendasi *Top-N* yang diberikan kepada pengguna. Pada penelitian ini digunakan *Precision@10*, karena rekomendasi pada peringkat teratas memiliki peran paling penting dalam mendukung pengambilan keputusan pengguna.

Precision@10 untuk satu pengguna dihitung sebagai perbandingan antara jumlah *item* relevan dalam sepuluh rekomendasi teratas terhadap jumlah rekomendasi yang diberikan, yaitu 10. Sebagai contoh, apabila dari sepuluh rekomendasi terdapat dua *item* yang relevan, maka nilai *Precision@10* adalah:

$$Precision@10 = \frac{2}{10} = 0,2$$

Nilai *Precision@10* selanjutnya dihitung untuk setiap pengguna, kemudian dirata-ratakan untuk memperoleh nilai *Precision@10* rata-rata sebagai berikut:

$$Precision@10_{avg} = \frac{1}{29} \times \sum_{i=1}^{29} Precision@10_i$$

$$Precision@10_{avg} = \frac{5.4}{29} = 0.1897 \times 290 = 55$$

Pada penelitian ini, jumlah pengguna yang dievaluasi adalah 29 pengguna, sehingga total rekomendasi yang dianalisis adalah sebanyak $29 \times 10 = 290$ item. Berdasarkan hasil perhitungan, diperoleh nilai *Precision@10* rata-rata sebesar 0,1897. Nilai tersebut menunjukkan bahwa sekitar 18,97% dari rekomendasi yang diberikan bersifat relevan. Dengan demikian, dari total 290 rekomendasi yang dihasilkan, terdapat sekitar 55 item relevan.

Berdasarkan pengujian, nilai RMSE dan MAE terendah diperoleh pada $k = 4$, evaluasi performa model rekomendasi menggunakan metrik RMSE dan MAE menunjukkan bahwa metode *truncated Singular Value Decomposition* (SVD) mampu memberikan prediksi *rating* yang cukup baik pada data *rating* yang bersifat *sparse*. Hasil ini sejalan dengan beberapa penelitian sebelumnya yang menyatakan bahwa SVD efektif digunakan untuk meningkatkan akurasi prediksi pada sistem rekomendasi berbasis *collaborative filtering*. Shalannanda et al. (2022) misalnya, menunjukkan bahwa penerapan SVD pada domain *e-commerce* menghasilkan nilai RMSE terbaik dibandingkan metode faktorisasi lainnya, bahkan setelah dilakukan optimasi parameter performa model semakin meningkat. Temuan tersebut mendukung hasil penelitian ini bahwa pemilihan parameter laten yang tepat berperan penting dalam menekan kesalahan prediksi serta meningkatkan kualitas rekomendasi.

Selain itu, Ramadhan & Wibowo (2023) yang menerapkan SVD pada sistem rekomendasi musik juga memperoleh performa prediksi yang stabil melalui pengaturan parameter $n_factors$ dan n_epochs , serta menunjukkan peningkatan kualitas rekomendasi melalui pengujian NDCG. Walaupun domain penelitian berbeda, hasil tersebut memperkuat bahwa pendekatan SVD dapat digunakan secara luas untuk berbagai jenis item karena mampu mengekstraksi pola laten preferensi pengguna. Berbeda dari penelitian-penelitian tersebut, penelitian ini berfokus pada domain anime yang cenderung memiliki preferensi yang lebih subjektif serta pola *rating* yang tidak merata. Dengan demikian, penerapan SVD dalam konteks rekomendasi anime menjadi kontribusi tambahan karena memperlihatkan bahwa metode ini tetap mampu bekerja secara efektif pada karakteristik data yang lebih bervariasi.

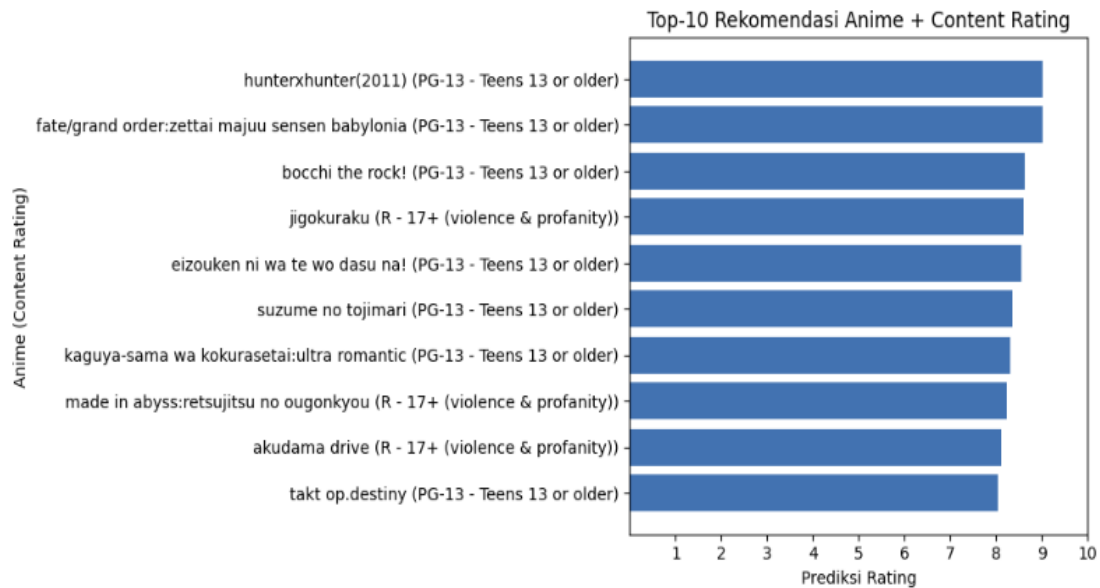
3.6 Hasil Prediksi *Rating* dan Analisis Rekomendasi Anime

Berdasarkan nilai k optimal yang diperoleh dari hasil evaluasi kinerja sistem, yaitu $k = 4$, dilakukan rekonstruksi ulang matriks pengguna–item menggunakan model *truncated* SVD. Hasil rekonstruksi ini digunakan untuk menghasilkan prediksi *rating* pengguna terhadap item yang belum terobservasi. Data hasil prediksi bisa dilihat pada tabel 6.

Tabel 6. Hasil Prediksi *Rating*

No	User_id	86: eighty six	akame ga kill!	akudama drive	...	yofukashi no uta	youkoso jitsuryoku shijou shugi no kyoushitsu e
1	30949	7.6	4.6	8.1	...	6.4	6.4
2	97248	7.6	6.3	8.3	...	7.8	7
3	155475	2.9	2	7	...	5.6	6.8
4	186132	8	6.5	7.7	...	6.9	8.1
5	214861	7.5	6.2	8.1	...	7.6	7.1
...	...	...	...	...	...	...	...
29	6698347	6.9	6.5	7.4	...	7.2	7
30	7424780	6.8	5.3	9.2	...	10	4.7

Tabel 6 menampilkan contoh hasil rekonstruksi matriks pengguna–item setelah penerapan *truncated* SVD dan pengembalian ke skala *rating* asli. Nilai pada tabel merepresentasikan prediksi *rating* pengguna terhadap item. Berdasarkan matriks prediksi *rating* yang dihasilkan, rekomendasi *Top-N* dibentuk berdasarkan nilai prediksi *rating* yang dihasilkan oleh model *truncated* SVD dengan parameter akhir $k = 4$. Item yang belum pernah dirating oleh pengguna diurutkan berdasarkan nilai prediksi tertinggi untuk menghasilkan daftar rekomendasi. Contoh hasil rekomendasi *Top-10* untuk salah satu pengguna dengan *user_id* 330949 ditunjukkan pada gambar 3.



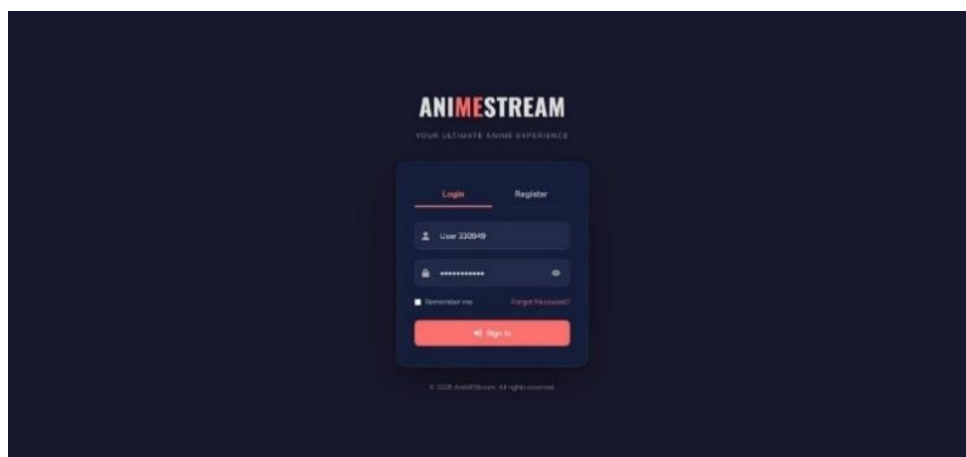
Gambar 3. Top-10 Rekomendasi Pada User 330949

Berdasarkan Gambar 3, dapat diketahui bahwa sistem rekomendasi menghasilkan daftar *Top-10* anime untuk pengguna 30949 dengan mengurutkan *item* berdasarkan nilai prediksi *rating* tertinggi. Anime yang menempati posisi teratas memiliki nilai prediksi yang lebih besar dibandingkan anime lainnya, sehingga diprioritaskan sebagai rekomendasi utama bagi pengguna tersebut.

Hasil analisis menunjukkan bahwa model mampu memberikan rekomendasi *Top-10* dengan nilai prediksi *rating* yang relatif tinggi, yaitu berada pada rentang 8–10. Hal ini mengindikasikan bahwa rekomendasi yang dihasilkan memiliki tingkat kesesuaian yang baik terhadap preferensi pengguna. Jika dilihat dari sisi *content rating*, sebagian besar anime yang direkomendasikan termasuk dalam kategori PG-13 (*Teens 13 or older*), yang berarti konten tersebut umumnya sesuai untuk penonton berusia 13 tahun ke atas atau kelompok remaja. Meskipun demikian, terdapat beberapa rekomendasi yang masuk dalam kategori R-17+ (*violence & profanity*), sehingga menunjukkan adanya konten dengan unsur yang lebih sensitif dan ditujukan bagi penonton usia 17 tahun ke atas. Temuan ini menegaskan bahwa sistem rekomendasi berbasis SVD tidak hanya efektif dalam memprediksi ketertarikan pengguna, tetapi juga perlu mempertimbangkan aspek kesesuaian usia agar rekomendasi yang diberikan tetap relevan sekaligus aman, khususnya bagi pengguna remaja. Oleh karena itu, sistem rekomendasi perlu mempertimbangkan aspek kesesuaian usia agar rekomendasi tidak hanya relevan, tetapi juga aman bagi pengguna.

3.7 Implementasi Sistem

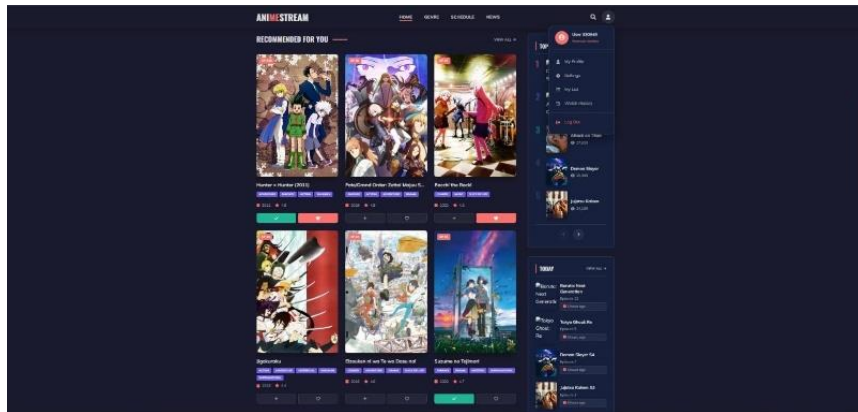
Pada tahap implementasi sistem, halaman *login* digunakan sebagai akses awal bagi pengguna untuk memasuki sistem rekomendasi anime. Tampilan antarmuka halaman *login* yang telah dirancang ditunjukkan pada gambar 4.



Gambar 4. Halaman *Login* Pengguna

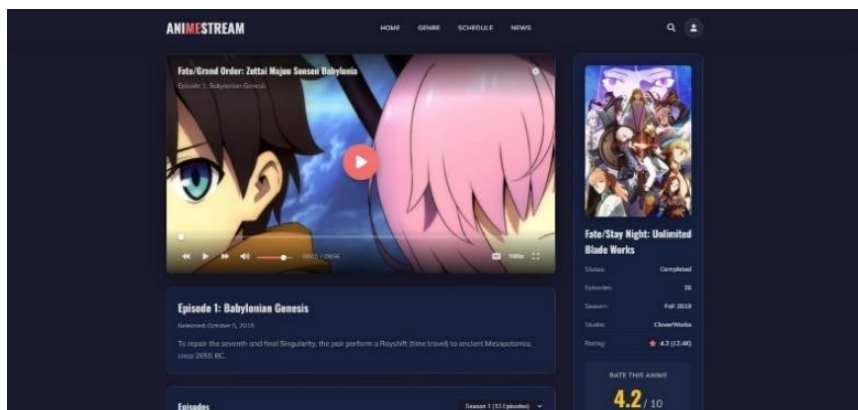
Gambar 4 merupakan antarmuka login yang digunakan oleh pengguna sebagai pintu masuk ke dalam web sistem rekomendasi, di mana pengguna diminta untuk memasukkan informasi akun seperti nama pengguna dan kata sandi sebelum dapat mengakses fitur-fitur yang tersedia di dalam sistem.

Setelah pengguna *login* dan masuk ke halaman utama sistem, sistem secara otomatis menampilkan daftar rekomendasi anime yang disesuaikan dengan preferensi pengguna. Tampilan halaman utama dapat dilihat pada gambar 5.



Gambar 5. Tampilan Rekomendasi Kepada Pengguna

Gambar 5 menunjukkan tampilan antarmuka rekomendasi anime pada sistem, di mana pengguna dapat melihat daftar anime yang direkomendasikan secara personal. Rekomendasi yang ditampilkan merupakan hasil perhitungan model *truncated Singular Value Decomposition* (SVD) berdasarkan nilai prediksi *rating* tertinggi untuk setiap pengguna. Setiap anime yang ditampilkan merupakan *item-item* yang belum pernah *dirating* oleh pengguna dan memiliki nilai prediksi tertinggi. Pada tampilan ini, anime disusun dalam bentuk kartu (*card*) yang menampilkan informasi visual seperti poster anime, judul, serta elemen pendukung lainnya untuk memudahkan pengguna dalam menelusuri rekomendasi. Dengan tampilan visual yang informatif dan terstruktur, sistem diharapkan dapat membantu pengguna dalam menemukan anime yang sesuai dengan minat mereka secara lebih efektif dan efisien. Tampilan antarmuka rekomendasi dapat dilihat pada gambar 6.



Gambar 6. Halaman Detail Anime dan Pemberian *Rating* oleh Pengguna

Gambar 6 menampilkan halaman detail anime pada sistem rekomendasi anime berbasis web. Pada halaman ini, pengguna dapat melihat informasi lengkap mengenai anime yang dipilih, seperti judul anime, cuplikan episode, deskripsi singkat, jumlah episode, serta nilai *rating* yang tersedia. Selain menampilkan informasi konten, halaman ini juga menyediakan fitur bagi pengguna untuk memberikan penilaian (*rating*) terhadap anime yang telah ditonton.

Rating yang diberikan oleh pengguna akan disimpan ke dalam basis data dan selanjutnya digunakan sebagai data masukan dalam pembentukan matriks pengguna-*item*. Data *rating* tersebut berperan penting dalam memperbarui model rekomendasi, karena semakin banyak interaksi pengguna yang tercatat, semakin akurat sistem dalam mempelajari preferensi pengguna. Dengan adanya fitur ini, sistem tidak hanya menampilkan rekomendasi, tetapi juga memungkinkan pengguna untuk berkontribusi secara langsung dalam meningkatkan kualitas rekomendasi yang dihasilkan.

4. KESIMPULAN

Berdasarkan hasil penelitian, penerapan metode *truncated Singular Value Decomposition* (SVD) pada sistem rekomendasi anime berbasis *collaborative filtering* terbukti mampu mengurangi dampak *data sparsity* pada matriks pengguna-*item*, sehingga sistem tetap dapat menangkap pola preferensi pengguna meskipun data *rating* yang tersedia tidak lengkap. Evaluasi kinerja model menunjukkan bahwa konfigurasi terbaik diperoleh pada $k = 4$, yang menghasilkan nilai kesalahan prediksi terendah pada data uji berdasarkan metrik *Root Mean Squared Error* (RMSE) dan *Mean Absolute Error* (MAE), sehingga mengindikasikan kemampuan generalisasi model yang baik terhadap data yang belum

terobservasi. Selain itu, pengukuran kualitas rekomendasi menggunakan *Precision@10* menghasilkan nilai rata-rata sebesar 0,1897, yang berarti sekitar 18,97% dari rekomendasi *Top-10* yang diberikan sistem tergolong relevan. Temuan ini menunjukkan bahwa pendekatan berbasis faktorisasi matriks mampu menghasilkan rekomendasi yang lebih akurat dan membantu pengguna dalam menemukan anime sesuai preferensi secara lebih efektif, khususnya di tengah meningkatnya jumlah konten anime pada platform digital. Dengan demikian, sistem rekomendasi yang diusulkan dapat menjadi solusi dalam mendukung proses pengambilan keputusan pengguna, meningkatkan efisiensi pencarian tontonan, serta memberikan pengalaman menonton yang lebih personal.

REFERENCES

- [1] D. Kurniawan Salamoon, "Anime Sebagai Media Edukasi Digital Mengenai Fungsi Sel Darah Merah (Analisis Visualisasi Karakter AE 3803 Pada Anime Hataraku Saibou) Anime as a Digital Educational Media Regarding the Function of Red Blood Cells (AE 3803 Character Visualization Analysis in Anime Hataraku Saibou)," *Jurnal Seni Budaya*, vol. 36, no. 2, pp. 197–203, 2021.
- [2] A. Nurmana and Florencia Irene Yuliani, "anime Pengaruh Anime (kartun Jepang) Terhadap Perilaku Kekerasan Pada Remaja," *Pixel :Jurnal Ilmiah Komputer Grafis*, vol. 17, no. 2, pp. 67–76, Dec. 2024, doi: 10.51903/pixel.v17i2.2107.
- [3] W. Ferbiansyah, A. Muhammad Irwan, B. Santoso, and S. Kacung, "IMPLEMENTASI METODE CONTENT BASED FILTERING MENGGUNAKAN SYNOPSIS SIMILARITY UNTUK PEMILIHAN ANIME," *Jurnal Informatika Teknologi dan Sains*, vol. 7, no. 2, p. 955, 2025.
- [4] Rosita, "REKOMENDASI BUKU PERPUSTAKAAN KAMPUS DENGAN METODE ITEM-BASED COLLABORATIVE FILTERING," *Sebatik*, vol. 26, no. 1, pp. 340–346, Jun. 2022, doi: 10.46984/sebatik.v26i1.1551.
- [5] N. Winnie, "ALiterature Review of Collaborative Filtering Recommendation System using Matrix Factorization algorithms," in *Proceedings of ACM Conference (Conference'17)*, IEEE Computer Society, 2022, pp. 36–47. doi: 10.1145/nnnnnnnn.nnnnnnnn.
- [6] I. K. G. Aryadi Pramarta and Z. K. A. Baizal, "HYBRID RECOMMENDER SYSTEM USING SINGULAR VALUE DECOMPOSITION AND SUPPORT VECTOR MACHINE IN BALITOURISM," *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 07, no. 02, 2022.
- [7] W. Shalannanda, R. F. Mulia, A. I. Muttaqien, N. R. Hibatullah, and A. Firdaus, "Singular value decomposition model application for e-commerce recommendation system," *JITEL (Jurnal Ilmiah Telekomunikasi, Elektronika, dan Listrik Tenaga)*, vol. 2, no. 2, pp. 103–110, Sep. 2022, doi: 10.35313/jitel.v2.i2.2022.103-110.
- [8] M. E. Ramadhan and A. T. Wibowo, "Music Recommender System Based on Play Count Using Singular Value Decomposition++," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 7, no. 3, p. 1210, Jul. 2023, doi: 10.30865/mib.v7i3.6424.
- [9] M. Attalariq and Z. K. A. Baizal, "Chatbot-Based Book Recommender System Using Singular Value Decomposition," *Journal of Information System Research (JOSH)*, vol. 4, no. 4, pp. 1293–1301, Jul. 2023, doi: 10.47065/josh.v4i4.3817.
- [10] F. Colace et al., "Recommender systems: a novel approach based on singular value decomposition," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 6, pp. 6513–6521, Dec. 2022, doi: 10.11591/ijece.v12i6.pp6513-6521.
- [11] R. Siringoringo and G. Lumbantoruan, "SISTEM PEREKOMENDASI DENGAN SINGULAR VALUE DECOMPOSITION DAN TEKNIK SIMILARITAS PEARSON CORRELATION," *Jurnal METHODIKA*, vol. 7, no. 1, 2021, [Online]. Available: <http://www2.informatik.uni-freiburg.de/~ciegler/BX/>.
- [12] W. Andriyani et al., *PERANGKAT LUNAK DATA MINING*. Penerbit Widina, 2024. [Online]. Available: <https://books.google.co.id/books?id=bdouEQAAQBAJ>
- [13] I. Zuhdiansyah and A. Luthfiarta, "Sistem Rekomendasi Pembelian Smartphone berbasis Algoritma K-Means dan Singular Value Decomposition," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 10, no. 1, pp. 45–53, May 2024, doi: 10.25077/teknosi.v10i1.2024.45-53.
- [14] F. Nissa, A. H. Primandari, and A. K. Thalib, "COLLABORATIVE FILTERING APPROACH: SKINCARE PRODUCT RECOMMENDATION USING SINGULAR VALUE DECOMPOSITION (SVD)," *MEDIA STATISTIKA*, vol. 15, no. 2, pp. 139–150, Apr. 2023, doi: 10.14710/medstat.15.2.139-150.
- [15] Z. T. Ke and M. Wang, "Using SVD for Topic Modeling," *J Am Stat Assoc*, vol. 119, no. 545, pp. 434–449, 2024, doi: 10.1080/01621459.2022.2123813.
- [16] R. Alabduljabbar, "Matrix Factorization Collaborative-Based Recommender System for Riyadh Restaurants: Leveraging Machine Learning to Enhance Consumer Choice," *Applied Sciences (Switzerland)*, vol. 13, no. 17, Sep. 2023, doi: 10.3390/app13179574.
- [17] K. K. Jena et al., "E-Learning Course Recommender System Using Collaborative Filtering Models," *Electronics (Switzerland)*, vol. 12, no. 1, Jan. 2023, doi: 10.3390/electronics12010157.
- [18] A. abbas, A. S. T. J. S. Manish Jaiswal, "Hybrid Recommender System based on Singular value decomposition using content features," *NeuroQuantology*, vol. 20, no. 9, Jun. 2024, doi: 10.48047/nq.2022.20.9.nq44485.
- [19] S. L. Brunton and J. N. Kutz, *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, 2022. [Online]. Available: <https://books.google.co.id/books?id=rxNkEAAAQBAJ>
- [20] A. Falini, "A review on the selection criteria for the truncated SVD in Data Science applications," *Journal of Computational Mathematics and Data Science*, vol. 5, Dec. 2022, doi: 10.1016/j.jcmds.2022.100064.
- [21] C. Wibisono, L. S. Haryadi, J. E. Widyaya, and S. L. Liliawati, "Sistem Rekomendasi Suku Cadang Berdasarkan Item Based Filtering," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 7, no. 1, Apr. 2021, doi: 10.28932/jutisi.v7i1.3036.
- [22] T. A. W. Tyas, Z. K. A. Baizal, and R. Dharayani, "Tourist Places Recommender System Using Cosine Similarity and Singular Value Decomposition Methods," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 5, no. 4, p. 1201, Oct. 2021, doi: 10.30865/mib.v5i4.3151.
- [23] R. Buana Perdana, D. Hartanti, and H. Hasanah, "Sistem Rekomendasi Peminjaman Buku Menggunakan Metode Vector Space Model Berbasis Pembobotan TF-IDF dan FastText (Studi Kasus Perpustakaan SMP Negeri 1 Kartasura)," *Infotek: Jurnal Informatika dan Teknologi*, vol. 8, no. 2, pp. 391–401, Jul. 2025, doi: 10.29408/jit.v8i2.30540.