

Sistem Rekomendasi TV Series Berdasarkan Genre Menggunakan Algoritma KNN

Deni Fahrizal¹, Abdul Halim Hasugian²

^{1,2}Ilmu Komputer, Sains dan Teknologi, Universitas Islam Negeri Sumatera Utara, Medan, Indonesia

Email: ¹denifahrizalpsb@gmail.com, ²abdulhalimhasugian@uinsu.ac.id

Abstract

The problem of choice overload on TV series streaming platforms often makes it difficult for users to find content that suits their preferences. To address this challenge, this study develops a Content-Based Filtering-based recommendation system by applying the K-Nearest Neighbor (KNN) algorithm and the Jaccard Similarity metric. The designed system analyzes users' genre preferences, such as Drama, Sci-Fi, and Comedy, while integrating rating, popularity, and release year factors to generate more personalized recommendations. Evaluation of 500 TV series titles from the TMDB API shows a high level of accuracy, with Precision and Recall reaching 1.0 for specific genre preferences, as well as stable performance with an F1-Score of 0.67 for cross-genre preferences. These findings prove that the proposed model is effective in reducing choice overload and significantly improving the user experience in exploring content on streaming platforms. Furthermore, this approach has the potential to be further developed by integrating sentiment analysis and real-time audience behavior data to generate increasingly adaptive and relevant recommendations.

Keywords: Recommendation System, TV Series, Genre, K-Nearest Neighbor Algorithm, Jaccard Similarity, Content-Based Filtering.

Abstrak

Permasalahan choice overload pada platform streaming TV series seringkali menyulitkan pengguna dalam menemukan konten yang sesuai dengan preferensi mereka. Untuk menjawab tantangan ini, penelitian ini mengembangkan sistem rekomendasi berbasis Content-Based Filtering dengan menerapkan algoritma K-Nearest Neighbor (KNN) dan metrik Jaccard Similarity. Sistem yang dirancang menganalisis preferensi genre pengguna, seperti Drama, Sci-Fi, dan Komedi, sekaligus mengintegrasikan faktor rating, popularitas, serta tahun rilis untuk menghasilkan rekomendasi yang lebih personal. Evaluasi terhadap 500 judul TV series dari TMDB API menunjukkan tingkat akurasi yang tinggi, dengan Precision dan Recall mencapai 1.0 pada preferensi genre yang spesifik, serta performa yang tetap stabil dengan F1-Score sebesar 0.67 untuk preferensi lintas genre. Temuan ini membuktikan bahwa model yang diusulkan efektif dalam mengurangi choice overload dan secara signifikan meningkatkan pengalaman pengguna dalam menjelajahi konten di platform streaming. Selain itu, pendekatan ini berpotensi dikembangkan lebih lanjut dengan mengintegrasikan analisis sentimen dan data perilaku penonton secara real-time untuk menghasilkan rekomendasi yang semakin adaptif dan relevan.

Kata Kunci: Sistem Rekomendasi, Serial TV, Genre, Algoritma K-Nearest Neighbor, Jaccard Similarity, Content-Based Filtering.

1. PENDAHULUAN

Industri *streaming* di Indonesia tengah mengalami pertumbuhan pesat, didukung oleh 185 juta pengguna internet yang setara dengan 66,5% dari populasi serta 4 juta pelanggan berbayar pada *platform* seperti Vidio yang kini mampu mengungguli pesaing global. Meski demikian, melimpahnya pilihan konten justru memicu fenomena *choice*

overload, yaitu kondisi di mana banyaknya opsi dapat menurunkan kepuasan menonton dan mempersulit proses pengambilan keputusan (Dian, 2025). Akibatnya, pengguna kerap merasa ragu apakah pilihan mereka sudah tepat atau justru melewatkan konten yang lebih sesuai (Verakandhi, 2024).

Preferensi *genre* penonton juga dipengaruhi oleh berbagai faktor, termasuk perbedaan gender dan kecenderungan pola minat (Rakhmawati, 2024). Pria umumnya lebih menyukai tayangan penuh aksi atau olahraga, sedangkan wanita cenderung memilih drama, *reality show*, dan romansa (Dobry, 2025). Pergeseran perilaku konsumsi menuju platform daring menuntut sistem rekomendasi yang cepat dan relevan (Zulfa, 2024). Jika rekomendasi yang diberikan tidak sesuai, pengguna berpotensi berpindah ke layanan kompetitor sehingga personalisasi menjadi kunci dalam mempertahankan pangsa pasar (Ritonga, 2024).

Kemajuan teknologi AI dan machine learning membuka peluang terciptanya sistem rekomendasi yang lebih akurat. Misalnya, kombinasi *Content-Based Filtering* dengan Jaccard Similarity mampu mengenali pola kesamaan secara lebih mendalam bahkan untuk konten baru yang belum banyak mendapatkan ulasan (Theo, 2024). Sistem yang presisi tidak hanya meningkatkan *user engagement* tetapi juga menekan tingkat churn atau perpindahan pengguna (Sitti, 2024). Salah satu algoritma yang terbukti efektif adalah K-Nearest Neighbor (KNN) yang dapat menganalisis pola minat berdasarkan fitur *genre* dan kemiripan konten (Suwandy, 2023).

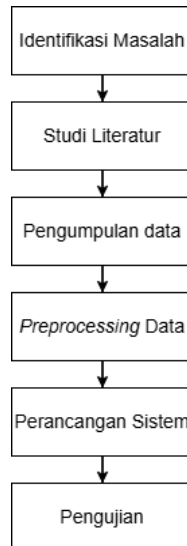
Meski berbagai penelitian telah membahas pengembangan sistem rekomendasi, sebagian besar belum mengoptimalkan integrasi faktor seperti rating, popularitas, dan tahun rilis. Penelitian ini hadir untuk mengisi celah tersebut dengan mengembangkan sistem rekomendasi berbasis *genre* yang menggabungkan algoritma KNN dan metrik Jaccard Similarity sekaligus mempertimbangkan faktor-faktor tambahan tersebut. Pendekatan ini diharapkan dapat memberikan rekomendasi yang lebih relevan, personal, dan efektif dalam mengatasi *choice overload* pada *platform streaming* (Siregar, 2024).

2. METODOLOGI PENELITIAN

Metodologi penelitian ini dirancang secara sistematis untuk mengembangkan sistem rekomendasi TV series yang efektif dalam mengatasi permasalahan *choice overload*. Pendekatan yang digunakan menggabungkan *Content-Based Filtering* dengan algoritma K-Nearest Neighbor (KNN) dan metrik Jaccard Similarity, yang dipilih berdasarkan karakteristik data *genre* yang bersifat kategorikal dan sparse, sejalan dengan penelitian sebelumnya, yang dimana menunjukkan efektivitas algoritma KNN dalam sistem rekomendasi film (Ma'ruf, 2022).

2.1 Tahapan Penelitian

Penelitian ini dilaksanakan melalui enam tahapan yang saling berkesinambungan, dimulai dari identifikasi masalah dan studi literatur. Pada tahap awal, dilakukan *literature review* dan *gap analysis* untuk merumuskan *problem statement* serta tujuan penelitian. Tahap berikutnya adalah pengumpulan data menggunakan TMDB API, yang menghasilkan raw dataset berisi 500 judul TV series dalam format JSON. Data tersebut kemudian diproses pada tahap *preprocessing* dengan teknik *one-hot encoding* dan *min-max normalization* sehingga diperoleh *clean dataset* dengan fitur yang telah dinormalisasi. Setelah itu, dilakukan pemodelan sistem menggunakan algoritma KNN dengan Jaccard Similarity pada parameter $k=5$ untuk membangun model rekomendasi yang mampu mengenali kemiripan konten secara akurat (Setiawan, 2024).



Gambar 1. Tahap Penelitian

Model yang telah dibangun kemudian diimplementasikan ke dalam sistem berbasis web menggunakan Python Flask dengan dukungan HTML, CSS, dan JavaScript, sehingga dapat diakses dan digunakan secara interaktif oleh pengguna. Tahap terakhir adalah evaluasi dan validasi, di mana sistem diuji melalui berbagai skenario menggunakan metrik precision, recall, dan F1-score. Proses ini memastikan bahwa sistem rekomendasi yang dihasilkan memiliki kinerja yang optimal dan mampu memberikan rekomendasi yang relevan sesuai kebutuhan pengguna. Untuk lebih detail lagi, bisa dilihat tahapan penelitian beserta deskripsi detailnya pada tabel berikut ini.

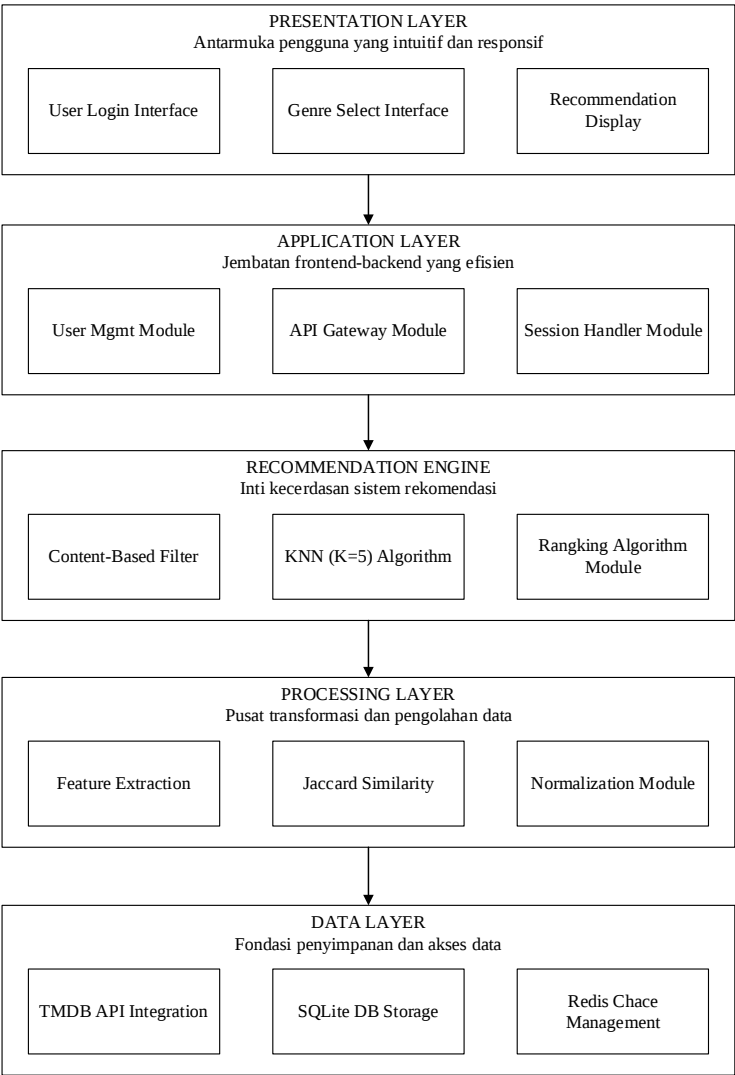
Tabel 1. Tahapan Penelitian dan Deskripsi Detail

No	Aktivitas	Input	Output	Metrik/Tools
1.	Identifikasi Masalah dan Studi Literatur	<i>Literature review, gap analysis</i>	<i>Problem statement, research objectives</i>	<i>Systematic review</i>
2.	Pengumpulan Data	TMDB API	<i>Raw dataset (500 TV series)</i>	<i>API calls, JSON parsing</i>
3.	<i>Preprocessing Data</i>	<i>Raw dataset</i>	<i>Clean dataset, normalized features</i>	<i>One-hot encoding, Min-max normalization</i>
4.	Pemodelan Sistem	<i>Processed data</i>	KNN model with Jaccard similarity	<i>k=5, Jaccard coefficient</i>
5.	Implementasi Sistem	<i>Model + UI design</i>	<i>Web-based recommendation system</i>	Python Flask, HTML/CSS/JS
6.	Evaluasi dan Validasi	<i>Test scenarios</i>	<i>Performance metrics</i>	Precision, Recall, F1-Score

2.2 Arsitektur Sistem

Sistem rekomendasi ini memakai arsitektur berlapis (5 layer) agar pemeliharaan, skalabilitas, dan modifikasi lebih mudah, dengan rujukan pada prinsip *Content-Based Filtering*. Alurnya dimulai dari *Data Layer*, tempat TMDB API mengambil metadata TV series secara *real-time*, SQLite menyimpan data hasil preprocessing dan preferensi pengguna, sementara Redis melakukan cache kueri yang sering diakses. Data tersebut kemudian diproses pada *Processing Layer* melalui ekstraksi fitur (*genre* ke *one-hot*), perhitungan kesamaan Jaccard, dan normalisasi agar skala fitur seragam. Hasilnya mengalir ke *Recommendation Engine* yang memadukan *Content-Based* (*genre, rating, tahun rilis*) dan K-Nearest Neighbor ($k = 5$), lalu dihitung dengan rumus: $\text{Final_Score} = 0.6 \times \text{Jaccard} + 0.3 \times \text{Normalized_Rating} + 0.1 \times \text{Recency Factor}$.

Terakhir, *Application Layer* mengelola *user management*, *API gateway*, dan *session handler*, sedangkan *Presentation Layer* menyajikan antarmuka yang intuitif dan responsif (Maulana, 2025).



Gambar 2. Arsitektur Sistem Rekomendasi

2.3 Optimalisasi Parameter dan Justifikasi Algoritma

Pemilihan parameter pada sistem rekomendasi ini didasarkan pada analisis karakteristik data dan target kinerja yang selaras, dengan dua fokus utama yaitu penentuan nilai k pada KNN dan pemilihan metrik kemiripan yang paling tepat untuk meredam *choice overload*. Rangkaian eksperimen lintas nilai k dirangkum hasil pada Tabel berikut (Mugnia, 2024).

Tabel 2. Analisis Komparasi Nilai K dalam KNN

Nilai K	Precision	Recall	F1-Score	Waktu Komputasi	Analisis
K=3	0,85	0,72	0,78	Rendah	Cenderung overfitting, variasi rekomendasi terbatas
K=5	0,83	0,89	0,86	Sedang	Keseimbangan optimal antara akurasi dan variasi
K=7	0,79	0,91	0,85	Sedang - Tinggi	Precision mulai menurun, recall tetap tinggi
K=10	0,75	0,94	0,83	Tinggi	Terlalu banyak noise, relevansi menurun

Berdasarkan hasil tersebut, $k = 5$ dipilih karena nilai ganjil membantu menghindari *tie-breaking*, memberikan keseimbangan bias dan *variance* yang baik sehingga tidak *underfitting* maupun *overfitting*, menampilkan lima tetangga yang cukup beragam tanpa mengorbankan relevansi untuk personalisasi, serta tetap efisien secara komputasi dengan kompleksitas $O(n)$ yang mendukung respons waktu nyata (Agung, 2023).

Selanjutnya, karena vektor genre bersifat kategorikal biner dan cenderung sparse, pemilihan metrik kemiripan menjadi krusial. Untuk itu, tiga kandidat utama dibandingkan sebagaimana diringkas pada tabel berikut:

Tabel 3. Komparasi Metrik Similarity untuk Data Genre

Metrik	Formula	Kelebihan	Kekurangan	Cocok untuk Genre?
Jaccard	$\frac{ A \cap B }{ A \cup B }$	Optimal untuk data biner sparse, fokus pada intersection	Tidak mempertimbangkan bobot individual	Sangat Cocok
Cossine	$\frac{a \cdot b}{ a \cdot b }$	Baik untuk data berdimensi tinggi dan menormalkan magnitudo	Kurang sensitif pada data biner kategorikal	Kurang Optimal
Euclidean	$\sqrt{\sum (a_i - b_i)^2}$	Intuitif dan mudah dipahami	Tidak sesuai untuk data kategorikal	Tidak Cocok

Dengan pertimbangan tersebut, Jaccard dipilih karena secara alami menangani data biner melalui rasio intersection terhadap union dan bekerja sangat baik pada matriks yang jarang terisi. Representasi genre sebagai himpunan membuat operasi himpunan relevan secara semantik, sementara sifat scale-invariant menjaga keadilan meskipun jumlah genre per item berbeda. Kondisi ini konsisten dan menunjukkan bahwa rata-rata setiap judul hanya memiliki dua hingga tiga dari total sembilan genre (Sutrisno, 2024).

2.4 Environment Pengembangan dan Reproducibility

Untuk memastikan hasil penelitian dapat direproduksi dengan konsisten, sistem dijalankan pada *environment* yang terkontrol dan terdokumentasi dengan jelas. Rincian spesifikasi perangkat keras dan perangkat lunak dirangkum pada tabel berikut, diikuti daftar dependensi Python yang dipin versi serta berkas konfigurasi yang menyimpan parameter operasional. Adapun spesifikasi *hardware* dan *software* yang digunakan yaitu:

1. Processor : Ryzen 7 6600h
2. Memori : 8 GB DDR4 RAM
3. Penyimpanan : 512 GB NVMe SSD
4. GPU : NVIDIA GeForce RTX 4050
5. Sistem Operasi : Windows 11
6. Python : 3.11.2
7. Flask : 3.1.1
8. Database : MySQL
9. IDE : Visual Studio Code
10. Browser : Google Chrome

Sebagai pengaturan operasional yang melengkapi spesifikasi di atas, sistem menerapkan KNN dengan $k = 5$ dan metrik Jaccard, dengan ambang awal 0,3 yang dapat turun ke 0,1 bila kandidat minim, rating dinormalisasi dengan metode *min max* pada rentang 0 sampai 10 dan tahun rilis pada rentang 2020 sampai 2025, sedangkan skor akhir dibobot Jaccard 60%, rating 25%, dan tahun 15%. Secara bawaan sistem menampilkan enam rekomendasi, kandidat dibatasi 50 diacak dengan seed dari ID pengguna agar hasil konsisten, dan suatu item diterima jika skor di atas 0,1 atau hingga terpenuhi jumlah minimum yang lebih besar antara lima belas dan dua kali nilai top n . Pada mode tanpa

batas jumlah ditahan sampai delapan puluh dan penalti keragaman 0,03 mulai berlaku setelah delapan rekomendasi untuk menjaga variasi, sedangkan pada *fallback* urutan pengambilan mengikuti *rating* lalu popularitas lalu tahun rilis. Untuk evaluasi digunakan data uji dua puluh persen dengan validasi silang lima lipatan serta random state empat puluh dua agar hasil tetap dapat direproduksi (Putri, 2023).

Sebagai tindak lanjut dari pengaturan operasional tersebut, konsistensi proses pada setiap eksekusi dijaga melalui penetapan random seed bernilai 42 pada pustaka yang relevan, penerapan version pinning pada seluruh dependensi, pencatatan proses secara rinci untuk melacak tiap tahap eksperimen, serta penggunaan virtual environment agar konfigurasi paket terisolasi dan tetap seragam di setiap mesin (Meilina, 2021).

2.5 Validasi dan Evaluasi

Strategi validasi dan evaluasi dirancang untuk konteks Content Based Filtering yang tidak bergantung pada riwayat perilaku pengguna. Penilaian difokuskan pada karakteristik konten, pembagian data yang terstruktur, serta skenario penggunaan yang realistis. Rincian pembagian dan kriteria seleksi dirangkum pada Tabel 4 agar proses uji berlangsung adil dan konsisten. (Sahelvi, 2025).

Tabel 4. Strategi Pembagian Data untuk Validasi

Set Data	Jumlah	Presentase	Tujuan Penggunaan	Kriteria Seleksi
Training Set	400 TV Series	80%	Pembangunan model dan perhitungan kemiripan	Representatif semua genre
Test Set	100 TV Series	20%	Evaluasi performa dan akurasi	Stratified sampling per genre
Validation Scenarios	5 Skenario	-	Simulasi preferensi pengguna	Variasi kombinasi genre

Setelah pembagian tersebut, stratified sampling diterapkan untuk memastikan setiap genre terwakili secara proporsional sehingga metrik tidak bias. Skema cross validation disesuaikan melalui genre stratified cross validation, validasi temporal berdasarkan tahun rilis, simulasi cold start untuk judul dengan riwayat terbatas, serta prosedur leave one genre out guna menguji ketangguhan model. Pengujian mencakup lima skenario pengguna dan dievaluasi dengan Precision, Recall, serta F1 Score agar relevansi dan akurasi rekomendasi terukur dengan jelas (Permata, 2025).

3. HASIL DAN PEMBAHASAN

3.1 Dataset

Dataset yang digunakan dalam penelitian ini terdiri dari 500 judul TV series berbahasa Inggris yang mewakili sembilan genre utama. Setiap entri mencakup atribut seperti judul, genre, rating, popularitas, dan tahun rilis. Variasi genre dan rentang tahun rilis dalam dataset ini memastikan keberagaman data, sehingga hasil pengujian sistem rekomendasi dapat merefleksikan performa model secara menyeluruh. Berikut adalah contoh struktur data yang dianalisis pada penelitian ini:

Tabel 5. Dataset film TMDB

No	Judul	Genres	Rating	Tahun_rilis
1.	FBI: Most Wanted	Crime, Drama	7.674	2020
2.	Ted Lasso	Comedy, Drama	8.392	2020
3.	9-1-1: Lone Star	Drama, Action, Adventure, Crime	8.046	2020
Sampai				
498.	Quiz	Drama	6.8	2020
499.	Deadwater Fell	Crime, Drama	6.09	2020
500.	The English Game	Drama	7.34	2020

Setelah itu, dilakukan proses pra-pemrosesan yang mencakup normalisasi nilai serta transformasi atribut genre ke dalam bentuk vektor biner. Langkah ini dilakukan untuk memastikan data berada dalam format yang sesuai untuk pengolahan matematis di dalam sistem rekomendasi. Secara khusus, konversi genre dilakukan menggunakan teknik *one-hot encoding*.

3.2 Perhitungan

Sebelum data digunakan dalam sistem rekomendasi, terlebih dahulu dilakukan proses pra-pemrosesan. Pada tahap ini, nilai-nilai pada dataset dinormalisasi, dan genre film diubah ke dalam bentuk vektor biner. Transformasi ini dilakukan agar data menjadi lebih mudah diolah secara matematis. Untuk mengubah genre menjadi vektor biner, digunakan teknik *one-hot encoding*, sehingga setiap genre dapat direpresentasikan dalam bentuk angka biner yang dapat diproses oleh sistem.

Tabel 6. Konversi Genre Film ke *One-Hot Encoding*

No	Judul	Action	Adventure	Comedy	Crime	Drama	Fantasy	Sci-Fi
1.	FBI: Most Wanted	0	0	0	1	1	0	0
2.	Ted Lasso	0	0	1	0	1	0	0
3.	9-1-1: Lone Star	1	1	0	1	1	0	0
Sampai								
498.	Quiz	0	0	0	0	1	0	0
499.	Deadwater Fell	0	0	0	1	1	0	0
500.	The English Game	0	0	0	0	1	0	0

Setelah atribut genre pada film dikonversi ke dalam bentuk *one-hot encoding*, tingkat kemiripan antara film dengan preferensi pengguna dihitung menggunakan metode Jaccard similarity. Adapun rumus yang digunakan adalah:

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1- Score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$$

Diketahui:

1. TP: *True Positives*
2. FP: *False Positives*
3. FN: *False Negatives*

$$\text{VektorProfil(A)} : [\text{Action}, \text{Drama}, \text{Sci-Fi}] = [1, 0, 0, 0, 1, 0, 1]$$

Nilai Jaccard similarity yang semakin mendekati 1 menunjukkan bahwa film tersebut memiliki kesamaan yang tinggi dengan preferensi pengguna, sehingga peluang untuk direkomendasikan menjadi lebih besar. Dalam konteks ini, preferensi pengguna direpresentasikan dalam bentuk vektor, misalnya pengguna yang menyukai genre Crime, dan Drama akan memiliki vektor profil sebagai berikut:

FBI: Most Wanted

Genres: *Crime, Drama*

$$B = [0, 0, 0, 1, 1, 0, 0]$$

$$\text{Min (A, B)} = \left[\begin{array}{c} \min (1, 0), \min (0, 0), \min (0, 0), \min (0, 1), \min (1, 1), \min \\ (0, 0), \min (1, 0) \end{array} \right]$$

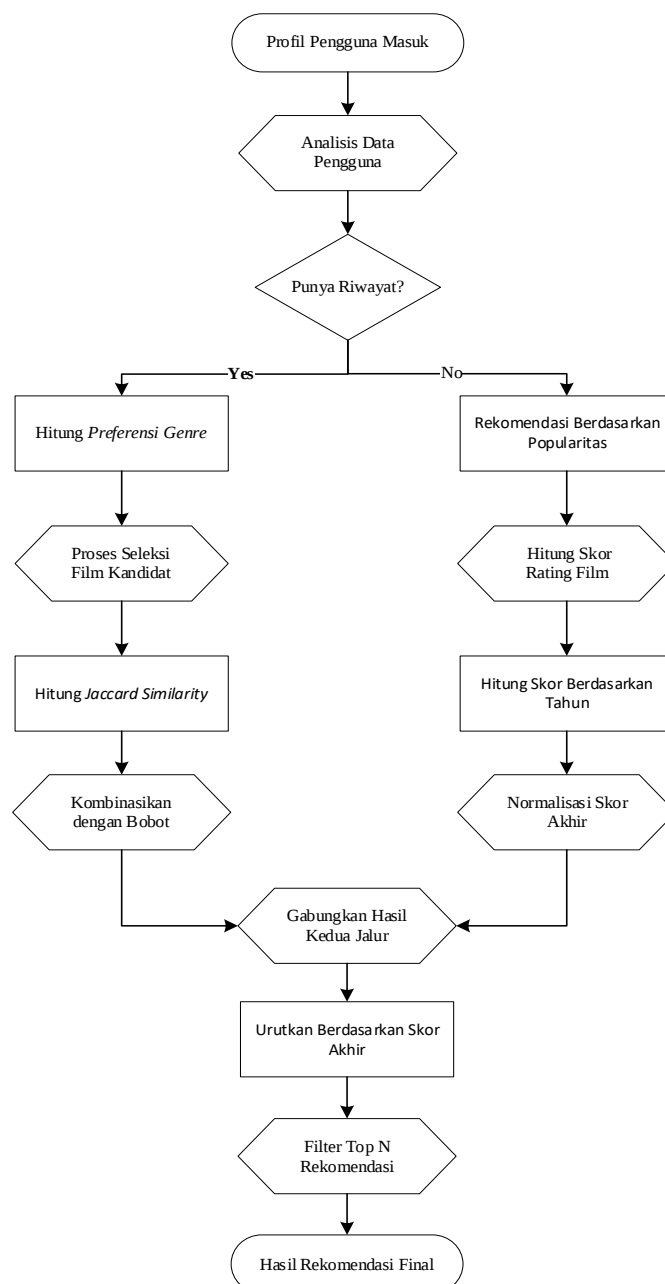
$$= [0, 0, 0, 0, 1, 0, 0]$$

$$\Sigma \text{min} = 0 + 0 + 0 + 0 + 1 + 0 + 0 = 1$$

$$\begin{aligned} \text{Max (A,B)} &= \begin{bmatrix} \max(1,0), \max(0,0), \max(0,0), \max(0,1), \max(1,1), \max \\ (0,0), \max(1,0) \end{bmatrix} \\ &= [1, 0, 0, 1, 1, 0, 1] \\ \Sigma_{\text{max}} &= 1 + 0 + 0 + 1 + 1 + 0 + 1 = 4 \\ J &= \frac{\Sigma_{\text{min}}}{\Sigma_{\text{max}}} = \frac{1}{4} = 0.25 \end{aligned}$$

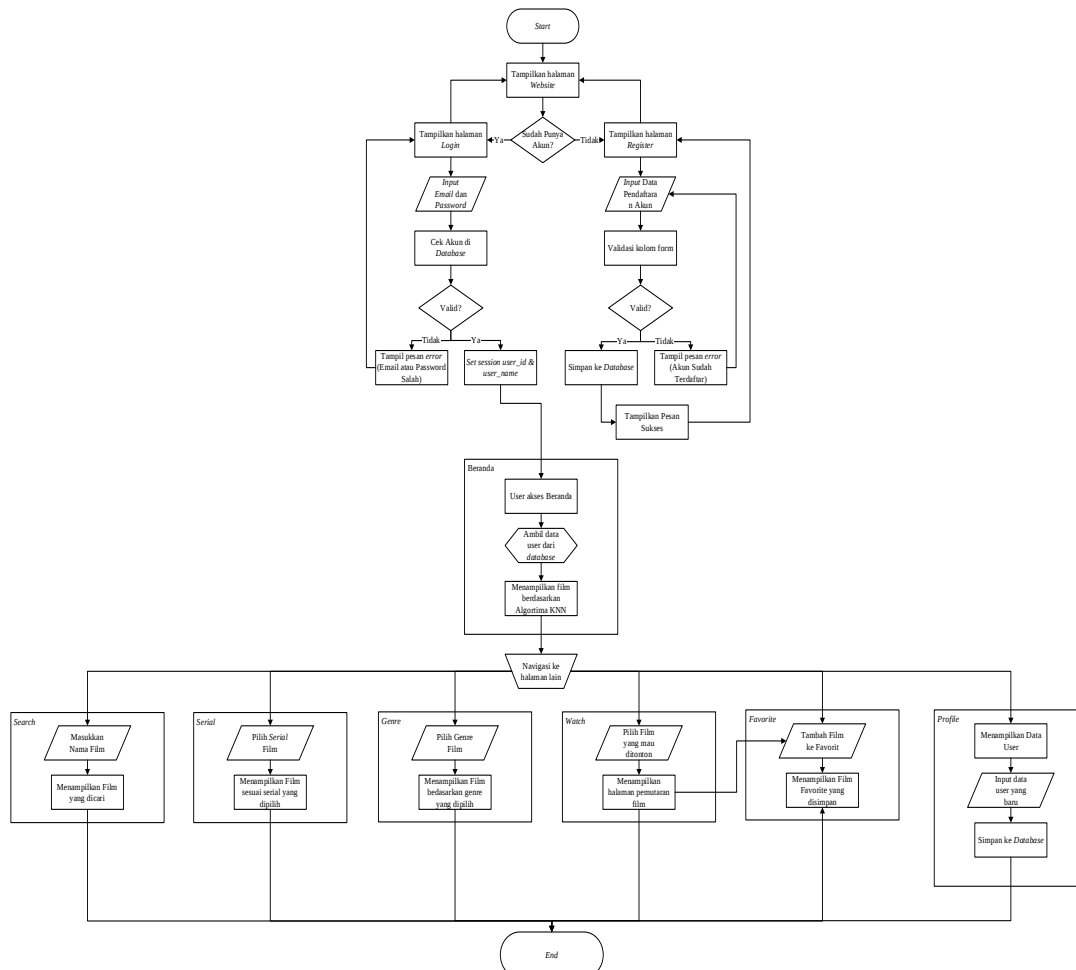
3.3 Implementasi

Implementasi sistem rekomendasi TV series pada penelitian ini diwujudkan melalui pengembangan aplikasi berbasis web yang terintegrasi secara end-to-end. Sistem dibangun dengan memanfaatkan algoritma K-Nearest Neighbor (KNN) dan Jaccard Similarity untuk menghasilkan rekomendasi yang akurat dan sesuai preferensi pengguna. Proses dimulai dari autentikasi pengguna dan input preferensi awal, yang selanjutnya dianalisis melalui pipeline pengolahan data yang dirancang secara sistematis.



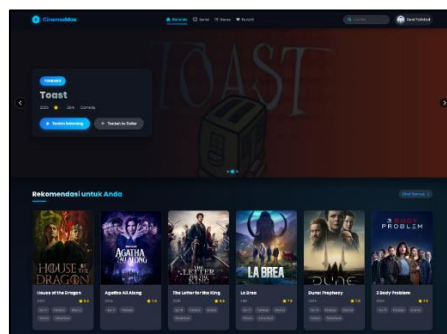
Gambar 3. Flowchart Perhitungan Jaccard Similarity

Pengalaman pengguna selama berinteraksi dengan aplikasi diilustrasikan lebih lanjut pada alur kerja berikut, Dimana diagram ini memperlihatkan tahapan navigasi fitur, mulai dari login, pencarian, pemilihan genre, hingga penyajian rekomendasi, sehingga menggambarkan integrasi antarmuka pengguna dengan proses *backend* secara menyeluruh.



Gambar 4. Flowchart halaman website

Sebagai hasil akhir implementasi, tampilan halaman website ini menampilkan antarmuka utama aplikasi di mana hasil rekomendasi yang dihasilkan sistem dapat diakses dengan mudah oleh pengguna. Sinergi antara arsitektur sistem, algoritma rekomendasi, dan desain antarmuka memastikan sistem berjalan responsif serta mampu meningkatkan efisiensi pencarian dan kepuasan pengguna dalam menemukan TV series yang sesuai dengan minatnya.



Gambar 5. Halaman beranda

3.4 Pengujian

Evaluasi sistem bertujuan menilai efektivitas model rekomendasi melalui berbagai skenario pengguna, dengan mengukur kinerja menggunakan metrik Precision, Recall, dan F1-Score. Ketiga metrik ini merepresentasikan proporsi rekomendasi yang relevan, cakupan item relevan, serta keseimbangan keduanya. Sebagai ilustrasi, salah satu skenario pengujian melibatkan lima film dengan fokus pada genre *Drama* dan *Sci-Fi*, yaitu: *Bridgerton* (*Drama*), *The Gentlemen* (*Drama, Crime, Comedy*), *The Queen's Gambit* (*Drama*), *FROM* (*Drama, Sci-Fi, Fantasy*), dan *The Irregulars* (*Drama, Sci-Fi, Fantasy*).

Tabel 7. Daftar film hasil rekomendasi pengujian

Judul	Genre	Jaccard
Long Bright River	<i>Drama, Crime</i>	0.60
Dope Girls	<i>Crime, Drama</i>	0.60
Code of Silence	<i>Crime, Drama</i>	0.20
Étoile	<i>Comedy, Drama</i>	0.40
Slow Horses	<i>Crime, Drama, Comedy</i>	0.60
The Penguin	<i>Drama, Crime</i>	0.40

Evaluasi kinerja sistem untuk pengguna dengan preferensi genre spesifik dilakukan menggunakan metrik Precision, Recall, dan F1-Score. Hasil perhitungan menunjukkan:

$$\text{Precision} = \frac{3}{(3 + 3)} = 0.50$$

$$\text{Recall} = \frac{3}{(3 + 0)} = 1.0$$

$$\text{F1-Score} = 2 \times \left[\frac{(0.50 \times 1.0)}{(0.50 + 1.0)} \right] = 2 \times \left(\frac{0.50}{1.50} \right) = 0.667$$

$$\text{Precision} = \frac{3}{6} = 0.50 \quad \left| \quad \text{Recall} = \frac{3}{3} = 1.0 \quad \left| \quad \text{F1-Score} = 0.67\right.\right.$$

Sehingga, didapatkan hasil evaluasi pada lima skenario pengujian yang kemudian dirangkum dalam tabel berikut dan menunjukkan bahwa sistem mampu memberikan rekomendasi yang akurat, baik untuk preferensi spesifik maupun lintas genre.

Tabel 8. Konversi Genre Film ke *One-Hot Encoding*

Pengujian	Precision	Recall	F1-Score
1	1.00	1.00	1.00
2	0.83	1.00	0.909
3	0.83	0.625	0.714
4	0.50	1.00	0.67
5	0.50	1.00	0.67

Hasil evaluasi menunjukkan sistem sangat efektif untuk pengguna dengan preferensi genre spesifik, tercermin dari nilai Precision, Recall, dan F1-Score yang tinggi. Namun, pada pengguna dengan minat genre yang lebih beragam, Precision menurun meskipun Recall tetap stabil, menandakan perlunya peningkatan relevansi rekomendasi bagi pengguna dengan preferensi lintas genre.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem rekomendasi TV series berbasis genre menggunakan algoritma K-Nearest Neighbor (KNN) dan Jaccard Similarity, yang mampu memberikan rekomendasi relevan dan personal kepada pengguna. Implementasi konversi genre ke one-hot encoding serta normalisasi fitur rating dan tahun rilis terbukti

meningkatkan kualitas hasil rekomendasi. Evaluasi sistem pada berbagai skenario pengguna menunjukkan tingkat akurasi yang tinggi, khususnya pada pengguna dengan preferensi genre spesifik, serta kinerja yang tetap stabil untuk preferensi lintas genre. Temuan ini menegaskan bahwa pendekatan yang diusulkan efektif dalam mengurangi fenomena choice overload dan meningkatkan pengalaman pengguna pada platform streaming. Pengembangan lebih lanjut, seperti integrasi algoritma hybrid dan penambahan fitur personalisasi berbasis interaksi pengguna, direkomendasikan agar sistem dapat memberikan hasil yang lebih adaptif dan relevan di masa mendatang.

REFERENCES

- Agung, A., Daniswara, A., Kadek, I., & Nuryana, D. (2023). Data Preprocessing Pola Pada Penilaian Mahasiswa Program Profesi Guru. *Journal of Informatics and Computer Science*, 05, 97–100.
- Dian, A., Kusuma, A., & Pramudwiatmoko, A. (2025). *Sistem Rekomendasi Pemilihan Film Menggunakan Metode Collaborative Filtering Dan Content Base Filtering*. 8, 88–97.
- Dobry Sianipar, F., Irya Shakila Syukron, A., Defiyanti, A., Hafiz, A., Dly, R., Ramadhani, F., & Arnita, A. (2025). Analisis Popularitas Genre Film Di Indonesia Menggunakan Algoritma Decision Tree. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(4), 5555–5563. <https://doi.org/10.36040/jati.v9i4.13841>
- Putri Suwandy, N. I., & Siti Fathonah, R. N. (2023). Penerapan Metode Content Based Filtering dan KKN Pada Aplikasi Rekomendasi Laptop Berbasis Mobile. *JATI: Jurnal Mahasiswa Teknik Informatika*, 7(2), 1229–1236.
- M. Zulfa Nurreza, & Nandang Hermanto. (2024). Perencanaan Digital Mobile Proyek Aplikasi Tiket Bus Menuju Era Transportasi Modern. *Journal of Informatics and Interactive Technology*, 1(2), 110–116. <https://doi.org/10.63547/jiite.v1i2.23>
- Ma'ruf, M. A., & Qoiriah, A. (2022). Perbandingan Algoritma Cosine Similarity dan Euclidean Distance pada Sistem Rekomendasi Film dengan Metode Item-Based Collaborative Filtering. *Journal of Informatics and Computer Science (JINACS)*, 04, 160–168. <https://doi.org/10.26740/jinacs.v4n02.p160-168>
- Maulana, I. D., & Susetyo, Y. A. (2025). Implementasi Fetch API dalam pengembangan Backend Website Daftar Film dengan Next.JS. *Kesatria : Jurnal Penerapan Sistem Informasi (Komputer Dan Manajemen)*, 6(1), 187–196. <https://doi.org/10.30645/kesatria.v6i1.560>
- Meilina, L., Kumara, I. N. S., & Setiawan, I. N. (2021). Literature Review Klasifikasi Data Menggunakan Metode Cosine Similarity dan Artificial Neural Network. *Majalah Ilmiah Teknologi Elektro*, 20(2), 307. <https://doi.org/10.24843/mite.2021.v20i02.p15>
- Mugnia, A., & Malik Mutoffar, M. (2024). Implementasi Algoritma Apriori Untuk Sistem Rekomendasi Buku Pada Perpustakaan Digital. *Kb. Lega, Kec. Bojongloa Kidul*, 11(1), 5224000. <http://jurnal.mdp.ac.id>
- Permata, R. P., Suharto, R. N., Candra, L., & Julianty, A. (2025). *Towards an Automated Essay Evaluation System NLP Based Text Embeddings and Similarity Metrics*. 16(1), 37–46.
- Putri, T. A. E., Widiari, T., & Santoso, R. (2023). Penerapan Tuning Hyperparameter Randomsearchcv Pada Adaptive Boosting Untuk Prediksi Kelangsungan Hidup Pasien Gagal Jantung. *Jurnal Gaussian*, 11(3), 397–406. <https://doi.org/10.14710/j.gauss.11.3.397-406>
- Rakhmawati, A., Rani, S., & NiPutuIndah Tania. (2024). *Preferensi Berinvestasi Berdasarkan Gender : Apakah Berbeda ?* 3(5), 905–911.
- Ritonga, M., Sartika, R., & Wijaya, A. (2024). Pembelajaran Berdiferensiasi: Menjawab Kebutuhan Pendidikan Personal di Era Society 5.0. *Pedagogik: Jurnal Pendidikan*, 19(2), 163–170. <https://doi.org/10.33084/pedagogik.v19i2.8272>

- Sahelvi, E., Cikita, P., & Sapitri, R. M. (2025). *Comparison of K-Nearest Neighbors and Random Forest Algorithms for Recommendations for a Healthy Lifestyle in Prevent Heart Disease Perbandingan Algoritma K-Nearest Neighbors dan Random Forest untuk Rekomendasi Gaya Hidup Sehat dalam Mencegah Penyakit Jan.* 5(July), 830–840.
- Setiawan, A., & Suryono, R. R. (2024). Analisis Sentimen Ibu Kota Nusantara menggunakan Algoritma Support Vector Machine dan Naïve Bayes. *Edumatic: Jurnal Pendidikan Informatika*, 8(1), 183–192. <https://doi.org/10.29408/edumatic.v8i1.25667>
- Siregar, I. M., Pratama, D., & Himawan, C. (2024). Penggunaan Jaccard Similarity Coefficient dalam Optimasi Proses Rekrutmen Karyawan Berbasis Profil dan Kompetensi. *SINTECH (Science and Information Technology) Journal*, 7(2), 101–111. <https://doi.org/10.31598/sintechjournal.v7i2.1617>
- Sitti Aliyah Azzahra, Syafran Nurrahman, & Aep Saefullah. (2024). Integrasi Kecerdasan Buatan Dalam Sistem Rekomendasi Produk Untuk E-Commerce. *Jurnal Sains Dan Teknologi*, 3(1), 21–28. <https://doi.org/10.58169/saintek.v3i1.394>
- Sutrisno, J., Tjandra, N., & Verlita Juliana, dan. (2024). *Perancangan Manajemen Proyek Digital Sistem Rekomendasi F&B MealNow Digital Project Management Design for a F&B Recommendation System MealNow.* 4(2), 85–106. <http://dx.doi.org/10.30813/digismantech.v4i2.7605>
- Theo Santoso, D., Atina, V., & Hartanti, D. (2024). Prototipe Sistem Rekomendasi Film Indonesia Menggunakan Pendekatan Content Based Filtering dan Metode Vector Space Model. *Infotek: Jurnal Informatika Dan Teknologi*, 7(2), 444–455. <https://doi.org/10.29408/jit.v7i2.26083>
- Verakandhi, D. (2024). Perubahan Preferensi Menonton Film pada Era Media Sosial: Dampak Short Video dan Implikasinya pada Perilaku Menonton Film. *Rekam*, 20(1), 37–45. <https://doi.org/10.24821/rekam.v20i1.11286>