SISTEM REKOMENDASI FILM MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBORS

Maximillian Huang[1)], Sany Noor Fauzianty[2)], Naufal Nuryanto[3)], Saila Julia[4)], Fransiskus Octavianus Mado Hurint[5)], Ivana Lucia Kharisma[6)]

[1, 2, 3, 4, 5)]Program Studi Teknik Informatika Fakultas Teknik, Komputer, dan Desain Universitas Nusa Putra
Jl. Raya Cibolang Cisaat - Sukabhumi No.21, Cibolang Kaler, Kec. Cisaat, Kabupaten Sukabumi, Jawa Barat 43152
e-mail: maximillian.huang_ti22@nusaputra.ac.id[1)], sany.noor_ti22@nusaputra.ac.id[2)], naufal.nuryanto_ti22@nusaputra.ac.id[3),] saila.julia_ti22@nusaputra.ac.id[4)], fransiskus.octavianus_ti22@nusaputra.ac.id[5)]. ivana.lucia@nusaputra.ac.id[6)]

ABSTRAK

Pemilihan film yang relevan dengan preferensi pengguna menjadi tantangan seiring meningkatnya jumlah pilihan film di berbagai platform. Penelitian ini bertujuan untuk mengembangkan sistem rekomendasi film berbasis algoritma K-Nearest Neighbors (KNN) guna memberikan rekomendasi yang lebih personal. Dataset yang digunakan dalam penelitian ini adalah movies_metadata.csv dan ratings_small.csv dari sumber publik. Model diuji dengan parameter k=5, 8, dan 10 menggunakan metrik kesamaan Cosine Similarity dan Euclidean Distance. Hasil pengujian menunjukkan bahwa konfigurasi k=10 dengan metrik Cosine Similarity memberikan hasil terbaik, dengan nilai Root Mean Square Error (RMSE) sebesar 1.0168. Sistem rekomendasi yang dikembangkan mampu memberikan rekomendasi film yang sesuai dengan preferensi pengguna berdasarkan data rating yang tersedia. Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem rekomendasi untuk meningkatkan pengalaman pengguna dalam memilih film.

***Kata Kunci**: K-Nearest Neighbors,* Sistem Rekomendasi, Film.

ABSTRACT

The selection of films relevant to user preferences becomes increasingly challenging as the number of film options on various platforms continues to grow. This study aims to develop a movie recommendation system based on the K-Nearest Neighbors (KNN) algorithm to provide more personalized recommendations. The datasets used in this research are movies_metadata.csv and ratings_small.csv from public sources. The model was tested with k values of 5, 8, and 10 using Cosine Similarity and Euclidean Distance as similarity metrics. The test results indicate that the configuration with k=10 and Cosine Similarity metric yielded the best performance, with a Root Mean Square Error (RMSE) value of 1.0168. The developed recommendation system effectively provides movie recommendations aligned with user preferences based on the available rating data. This research is expected to contribute to the development of recommendation systems to enhance user experience in selecting films.

***Keywords**: K-Nearest Neighbors, Recommendation System, Movie.*

I. PENDAHULUAN

Perkembangan teknologi digital telah mengubah cara masyarakat mengakses dan menikmati berbagai konten hiburan, termasuk film. Penggemar film yang semakin ramai membutuhkan informasi tentang film agar mereka tertarik dan ingin menontonnya. Saat ini, jumlah film yang tersedia di berbagai platform streaming terus meningkat secara signifikan, sehingga memunculkan tantangan baru bagi pengguna dalam memilih film yang sesuai dengan preferensi mereka. Masalah ini dikenal sebagai *information overload*, di mana banyaknya pilihan justru membuat pengguna kesulitan menentukan film yang akan ditonton[1].

Sistem rekomendasi (*recommender system*) menjadi salah satu solusi yang dapat membantu mengatasi masalah tersebut. Dengan memanfaatkan data pengguna, sistem ini mampu memberikan

rekomendasi yang relevan dan personal, sehingga dapat meningkatkan pengalaman pengguna. Berbagai metode telah dikembangkan untuk membangun sistem rekomendasi, mulai dari pendekatan berbasis konten (*content-based filtering*), kolaboratif (*collaborative filtering*), hingga metode berbasis algoritma pembelajaran mesin seperti *K-Nearest Neighbors* (KNN)[2].

Algoritma KNN merupakan salah satu algoritma sederhana namun efektif dalam membangun sistem rekomendasi. Algoritma ini bekerja dengan mengidentifikasi kesamaan antara pengguna atau item berdasarkan metrik tertentu, seperti *Cosine Similarity* atau *Euclidean Distance*. Parameter k, yang menentukan jumlah tetangga terdekat, menjadi faktor penting dalam menentukan performa sistem rekomendasi.

Penelitian ini bertujuan untuk mengembangkan sistem rekomendasi film berbasis algoritma KNN dengan menggunakan dataset publik "*movies_metadata.csv*" dan "*ratings_small.csv*". Penelitian ini juga mengevaluasi performa sistem dengan berbagai konfigurasi nilai k dan metrik kesamaan, guna menemukan konfigurasi yang memberikan hasil rekomendasi terbaik..

II. TINJAUAN PUSTAKA

A. K-Nearest Neighbor

Algoritma K-NN adalah metode yang dapat digunakan untuk prediksi atau klasifikasi data, tergantung pada jenis data yang terdapat dalam kumpulan data tersebut. Dalam proses klasifikasi, K-NN mengklasifikasikan data berdasarkan nilai k yang telah ditentukan sebelumnya. Untuk klasifikasi, nilai k pada K-NN sebaiknya berupa angka ganjil, sementara untuk prediksi, nilai k bisa berupa angka ganjil maupun genap. Perhitungan jarak yang digunakan dalam K-NN adalah *Euclidean Distance*, yang dihitung dengan menggunakan persamaan berikut[3].

$$d(X_i, X_j) = \sqrt{\sum_{r}^{n} = 1 \left(a_r(x_i) - a_r(x_j)\right)^2}$$

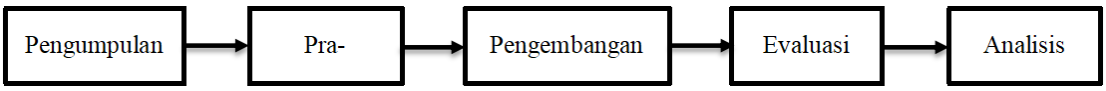
Keterangan:
d(Xi,Xj): Jarak Euclidean
(xi): record ke-i (baris)
(xj): record ke-j (kolom)
(ar): data ke-r
i,j: 1,2,3...n

B. Sistem Rekomendasi

Sistem rekomendasi adalah aplikasi atau sistem yang dirancang untuk menyediakan dan memberikan rekomendasi terkait suatu item guna membantu pengguna dalam membuat keputusan yang diinginkan. Selain itu, sistem rekomendasi juga dapat diartikan sebagai alat dan teknik dalam perangkat lunak yang berfungsi memberikan saran kepada pengguna mengenai item yang berpotensi bermanfaat, sehingga memudahkan pengguna dalam menentukan pilihan[4].

III. METODE PENELITIAN

Penelitian ini bertujuan untuk mengembangkan sistem rekomendasi film berbasis algoritma *K-Nearest Neighbors* (KNN) dengan memanfaatkan dataset film dan rating. Dalam upaya mencapai tujuan tersebut, metode penelitian ini mencakup beberapa tahap utama yang dirancang secara sistematis. Tahapan-tahapan tersebut meliputi pengumpulan data, praproses data, pengembangan model, evaluasi model, dan analisis hasil. Setiap tahap dirancang untuk memastikan bahwa data diolah dengan benar dan model yang dihasilkan memiliki kinerja yang optimal.



Gambar 1. Tahapan

A. Pengumpulan Data

Tahap pengumpulan data dilakukan dengan mengambil dua dataset utama, yaitu "*movies_metadata.csv*" dan "*ratings_small.csv*", yang tersedia dari Kaggle. Dataset "*movies_metadata.csv*" berisi informasi mendetail tentang berbagai film, termasuk judul, genre, tahun rilis, deskripsi, dan metadata lainnya yang relevan untuk sistem rekomendasi. Sedangkan "*ratings_small.csv*" mencakup data rating yang diberikan oleh pengguna kepada berbagai film. Rating ini berkisar dari 0,5 hingga 5,0 dengan interval 0,5, yang mencerminkan tingkat kesukaan pengguna terhadap film tersebut.

Dataset "*movies_metadata.csv*" terdiri dari ribuan baris data yang memuat atribut-atribut penting untuk menggambarkan karakteristik film. Informasi ini digunakan untuk memahami konteks dan preferensi pengguna terhadap film tertentu. Di sisi lain, "*ratings_small.csv*" mencatat interaksi antara pengguna dan film dalam bentuk rating, yang menjadi dasar dalam membangun model rekomendasi berbasis kolaborasi.

Setelah dataset diunduh, dilakukan proses validasi awal untuk memastikan kualitas data. Hal ini mencakup pemeriksaan kelengkapan data, deteksi nilai yang hilang atau null, dan konsistensi antar kolom. Data yang tidak valid, seperti entri yang tidak lengkap atau duplikat, dihapus atau diperbaiki untuk memastikan keakuratan dan keandalan data yang akan digunakan dalam proses selanjutnya.

B. Praproses Data

Tahap ini mencakup serangkaian langkah untuk mempersiapkan data sebelum digunakan dalam pengembangan model. Langkah pertama adalah penggabungan dataset "*movies_metadata.csv*" dengan "*ratings_small.csv*" untuk menyelaraskan data film dengan data rating. Proses ini dilakukan dengan memanfaatkan atribut kunci seperti "movieId" yang ada pada kedua dataset untuk memastikan setiap film yang memiliki data rating dapat dihubungkan dengan metadata yang relevan.

Selanjutnya, dilakukan deteksi dan penanganan nilai yang hilang. Nilai-nilai kosong pada atribut penting, seperti "*rating*" atau "*title*", dihapus untuk memastikan data yang digunakan lengkap dan relevan. Selain itu, *outlier* dalam data, seperti *rating* yang berada di luar rentang 0,5 hingga 5,0, diidentifikasi dan dihapus untuk menghindari bias dalam model.

Data yang telah bersih kemudian diubah ke dalam format yang sesuai untuk analisis. Rating film disusun dalam bentuk matriks pengguna-film, di mana baris merepresentasikan pengguna dan kolom merepresentasikan film, dengan nilai berupa rating yang diberikan. Matriks ini menjadi dasar dalam algoritma KNN untuk menghitung kesamaan antara pengguna atau item.

Setelah pembentukan matriks, dataset dibagi menjadi dua bagian: data pelatihan (70%) dan data pengujian (30%). Pembagian ini dilakukan secara acak namun tetap memperhatikan distribusi data untuk memastikan bahwa pola dalam data pelatihan dapat mencerminkan pola pada data pengujian. Data pelatihan digunakan untuk membangun model, sedangkan data pengujian digunakan untuk mengevaluasi kinerja model.

C. Pengembangan Model

Model sistem rekomendasi dibangun dengan algoritma *K-Nearest Neighbors* (KNN), yang dirancang untuk memanfaatkan kesamaan antar pengguna atau item untuk memberikan rekomendasi. Tiga nilai k (k=5, k=8, k=10) digunakan untuk mengeksplorasi pengaruh jumlah tetangga terdekat terhadap kinerja model. Selain itu, berbagai metrik kesamaan, seperti *cosine*, *Mean Squared Difference* (MSD), dan *Pearson*, diterapkan untuk mengukur jarak atau kesamaan antara entitas yang dibandingkan.

Algoritma KNN diimplementasikan menggunakan pustaka *Python* "*Surprise*". Setiap kombinasi parameter k dan metrik kesamaan diuji untuk mengidentifikasi pengaturan yang menghasilkan performa terbaik. Proses ini melibatkan pembangunan matriks kesamaan, yang menjadi dasar dalam menentukan tetangga terdekat untuk setiap pengguna atau item dalam dataset.

D. Evaluasi Model

Evaluasi model dilakukan dengan menggunakan tiga metrik utama, yaitu RMSE (*Root Mean Square Error*), *Precision*, dan *Recall*. RMSE digunakan untuk mengukur sejauh mana prediksi model mendekati rating sebenarnya, dengan nilai lebih rendah menunjukkan performa yang lebih baik. *Precision* dan *Recall* digunakan untuk mengevaluasi kemampuan model dalam memberikan rekomendasi yang relevan.

Precision mengukur proporsi rekomendasi yang benar-benar relevan, sedangkan *Recall* mengukur proporsi item relevan yang berhasil direkomendasikan oleh model. Selain itu, beberapa nilai ambang batas (*threshold*), seperti 3.0, 3.5, dan 4.0, diterapkan untuk menentukan apakah sebuah rating dianggap positif (layak direkomendasikan). Dengan variasi nilai threshold ini, dapat dianalisis bagaimana sensitivitas dan spesifisitas model berubah seiring dengan perubahan ambang batas.

E. Analisis Hasil

Analisis hasil dilakukan dengan membandingkan kinerja model berdasarkan kombinasi parameter k, metrik kesamaan, dan nilai *threshold*. Hasil evaluasi, seperti nilai RMSE, *Precision*, dan *Recall*, dianalisis untuk mengidentifikasi pola atau tren tertentu. Misalnya, hubungan antara peningkatan nilai k dengan perubahan akurasi model, atau dampak penggunaan metrik kesamaan tertentu terhadap performa rekomendasi. Analisis juga mencakup interpretasi hasil untuk memberikan wawasan mendalam tentang kekuatan dan kelemahan dari masing-masing kombinasi parameter. Hal ini bertujuan untuk memberikan rekomendasi tentang pengaturan optimal yang dapat digunakan dalam implementasi nyata dari sistem rekomendasi.

IV. HASIL DAN PEMBAHASAN

Analisis dilakukan berdasarkan metrik evaluasi, yaitu RMSE, *Precision*, dan *Recall*, serta mempertimbangkan variasi nilai k (5, 8, dan 10) dan threshold (3.0, 3.5, dan 4.0).

A. Hasil Eksperimen

Sistem diuji dengan nilai k=5, k=8, dan k=10 menggunakan tiga metrik kesamaan: *cosine*, MSD, dan *Pearson*. Hasil menunjukkan bahwa peningkatan nilai k secara umum menghasilkan RMSE yang lebih rendah, yang menunjukkan prediksi yang lebih akurat. *Precision* dan *Recall* juga meningkat pada nilai k yang lebih tinggi, terutama untuk threshold rendah (3.0), tetapi cenderung menurun untuk *threshold* tinggi (4.0).

K	Threshold	RMSE	Precision (%)	Recall (%)
5	3.0	1.0496	86.98	89.20
5	3.5	1.0496	68.19	73.98
5	4.0	1.0496	64.95	43.87
8	3.0	1.0235	87.09	90.50
8	3.5	1.0235	69.44	75.42
8	4.0	1.0235	67.57	41.43
10	3.0	1.0168	87.24	91.02
10	3.5	1.0168	69.89	75.85
10	4.0	1.0168	67.62	40.38

Table 1. Hasil Evaluasi dengan Metrik *Cosine Similarity*

Hasil evaluasi dengan metrik *Cosine Similarity* menunjukkan bahwa kinerja model terbaik dicapai pada nilai k=10 dengan *threshold* 3.0, menghasilkan *Precision* sebesar 87.24% dan *Recall* sebesar 91.02%. Namun, ketika *threshold* dinaikkan menjadi 4.0, *Recall* menurun drastis karena model cenderung lebih selektif, hanya merekomendasikan item dengan kesamaan tinggi. RMSE menunjukkan konsistensi di sekitar 1.01-1.05, mengindikasikan prediksi model cukup akurat.

K	Threshold	RMSE	Precision (%)	Recall (%)
5	3.0	0.9973	87.32	89.02
5	3.5	0.9973	69.81	73.48
5	4.0	0.9973	66.79	43.44
8	3.0	0.9794	87.57	90.05
8	3.5	0.9794	70.37	74.36
8	4.0	0.9794	67.65	40.82
10	3.0	0.9757	87.58	90.56
10	3.5	0.9757	70.49	74.79
10	4.0	0.9757	68.74	39.64

Table 2. Hasil Evaluasi dengan Metrik *Mean Squared Difference*

Metrik MSD memberikan hasil RMSE yang lebih rendah dibandingkan metrik lainnya, dengan nilai terbaik sebesar 0.9757 pada k=10 dan *threshold* 3.0. *Precision* dan *Recall* juga konsisten tinggi pada *threshold* yang lebih rendah, menunjukkan kemampuan model dalam memberikan rekomendasi yang relevan. Ketika *threshold* dinaikkan, Precision tetap cukup baik, tetapi *Recall* menurun akibat selektivitas model yang lebih tinggi.

K	Threshold	RMSE	Precision (%)	Recall (%)
5	3.0	1.0496	87.49	85.97
5	3.5	1.0496	70.60	69.84
5	4.0	1.0496	66.85	38.34
8	3.0	1.0268	87.69	86.77
8	3.5	1.0268	71.30	71.18
8	4.0	1.0268	67.62	36.05
10	3.0	1.0208	87.58	86.72
10	3.5	1.0208	71.48	71.54
10	4.0	1.0208	68.40	35.62

Table 3. Hasil Evaluasi dengan Metrik *Pearson Correlation*

Pada metrik *Pearson Correlation*, performa model menunjukkan pola yang serupa dengan metrik *Cosine Similarity*, namun dengan RMSE yang sedikit lebih tinggi. Pada k=10 dan *threshold* 3.5, model mencapai keseimbangan terbaik antara *Precision* (71.48%) dan *Recall* (71.54%). Ketika threshold dinaikkan menjadi 4.0, Recall menurun drastis, menunjukkan bahwa model hanya memberikan rekomendasi pada item dengan kesamaan yang sangat tinggi.

B. Pembahasan Hasil Evaluasi

Hasil evaluasi menunjukkan bahwa kombinasi k=10 dengan *threshold* 3.0 memberikan hasil yang konsisten baik untuk ketiga metrik kesamaan. Metrik MSD memberikan RMSE terendah, mengindikasikan bahwa metrik ini lebih baik dalam memprediksi rating sebenarnya dibandingkan *Cosine Similarity* dan *Pearson Correlation*. Namun, *Precision* dan *Recall* tetap menjadi indikator utama untuk mengevaluasi relevansi rekomendasi, dengan metrik *Cosine Similarity* unggul dalam menangkap item relevan pada *threshold* rendah. Analisis ini memberikan wawasan yang dapat digunakan untuk mengoptimalkan sistem rekomendasi lebih lanjut.

V. KESIMPULAN

Penelitian ini telah berhasil mengembangkan dan mengevaluasi sistem rekomendasi film berbasis algoritma *K-Nearest Neighbors* (KNN) dengan menggunakan dataset film dan rating. Sistem ini dirancang melalui tahapan pengumpulan data, praproses data, pengembangan model, evaluasi model, dan analisis hasil. Dalam proses evaluasi, berbagai metrik kesamaan seperti *Cosine Similarity, Mean Squared Difference* (MSD), dan *Pearson Correlation* diuji untuk memahami performa algoritma KNN dalam berbagai kondisi parameter K dan *threshold*.

Hasil evaluasi menunjukkan bahwa metrik MSD menghasilkan RMSE terendah, menandakan prediksi rating yang lebih akurat, sementara metrik *Cosine Similarity* menunjukkan performa terbaik pada *Precision* dan *Recall*, khususnya pada *threshold* rendah. Parameter k=10, k = 10, k=10 dengan threshold 3.0 ditemukan memberikan keseimbangan optimal antara RMSE, *Precision*, dan *Recall* untuk ketiga metrik kesamaan. Selain itu, penelitian ini juga mengungkap adanya *trade-off* antara *Precision* dan *Recall*, di mana peningkatan *Precision* sering diiringi penurunan *Recall*, yang memberikan wawasan penting dalam desain sistem rekomendasi. Secara keseluruhan, penelitian ini tidak hanya membuktikan keefektifan algoritma KNN dalam menghasilkan rekomendasi yang relevan, tetapi juga menawarkan peluang untuk eksplorasi lebih lanjut, seperti penggunaan data yang lebih besar atau pendekatan algoritma yang lebih kompleks untuk peningkatan performa dan efisiensi di masa depan.

VI. DAFTAR PUSTAKA

[1] R. Firmansyah, “Rancang Bangun Jaringan Komputer Dengan Kabel Listrik Sebagai Media Transmisi Untuk Komunikasi Data,” *J. Inform.*, vol. 1, no. 2, pp. 104–110, 2016.

[2] E. Salim, J. Pragantha, and D. L. Manatap, “Perancangan Sistem Rekomendasi Film menggunakan metode Content- based Filtering,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 6, pp. 2188–2199, 2021.

[3] A. Khairi, A. F. Ghozali, and A. D. N. Hidayah, “Implementasi K-Nearest Neighbor (KNN) untuk Mengklasifikasi Masyarakat Pra-Sejahtera Desa Sapikerep Kecamatan Sukapura,” *TRILOGI J. Ilmu Teknol. Kesehatan, dan Hum.*, vol. 2, no. 3, pp. 319–323, 2021.

[4] H. Februariyanti, A. Dwi Laksono, J. Sasongko Wibowo, and M. Siswo Utomo, “Implementasi Metode Collaborative Filtering Untuk Sistem Rekomendasi Penjualan Pada Toko Mebel,” *Khatulistiwa Inform.*, vol. 9, no. 1, pp. 43–45, 2021.