



## **Penerapan Metode *Content Based Filtering* pada Sistem Rekomendasi Drama Korea berbasis Algoritma *K-Nearest Neighbors (KNN)***

**Rangga Nur Wahyu<sup>1</sup>, Yulianti<sup>2</sup>**

<sup>1,2</sup>Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang, Tangerang Selatan, Indonesia

Email: <sup>1</sup>[ranganurwahyu22@gmail.com](mailto:ranganurwahyu22@gmail.com), <sup>2</sup>[dosen00838@unpam.ac.id](mailto:dosen00838@unpam.ac.id)

**Abstrak**—Era digital membawa peningkatan signifikan dalam kebutuhan hiburan, terutama terhadap tayangan drama Korea yang digemari oleh berbagai kalangan. Banyaknya judul yang terus bertambah sering kali menyulitkan pengguna dalam menentukan tontonan yang sesuai dengan minat mereka. Sistem rekomendasi dirancang untuk membantu pengguna menemukan tayangan yang relevan dengan preferensi individu, menggunakan metode *Content-Based Filtering* yang dipadukan dengan algoritma *K-Nearest Neighbors (K-NN)*. Data diambil dari sumber terbuka seperti Kaggle dan Drakor.id, lalu diolah dalam aplikasi berbasis web. Sistem menganalisis atribut tayangan seperti genre, aktor, sutradara, dan rating untuk menghitung tingkat kemiripan antar drama. Hasil implementasi menunjukkan bahwa sistem mampu menampilkan rekomendasi akurat berdasarkan input pengguna, meskipun tanpa riwayat tontonan sebelumnya. Antarmuka yang sederhana juga memberikan kemudahan dalam penggunaan serta meningkatkan kepuasan dalam memilih tayangan yang sesuai. Metode *Content-Based Filtering* yang dikombinasikan dengan algoritma *K-Nearest Neighbors* terbukti efektif dalam menghasilkan rekomendasi yang personal dan mendukung pengalaman menonton yang lebih efisien dan menyenangkan di tengah banyaknya pilihan konten.

**Kata kunci:** Sistem Rekomendasi, *Content-Based Filtering*, *K-Nearest Neighbors (KNN)*, Drama Korea.

**Abstract**—The digital era has driven a significant rise in entertainment demands, with Korean dramas becoming increasingly popular across various audiences. The rapid growth in the number of available titles often creates difficulties for users in selecting content that aligns with their interests. A recommendation system was designed to help users discover content that matches individual preferences using the *Content-Based Filtering* method combined with the *K-Nearest Neighbors (K-NN)* algorithm. Data was collected from open sources such as Kaggle and Drakor.id, and processed through a web-based application. The system analyzes content attributes such as genre, cast, director, and rating to calculate similarity between dramas. Implementation results indicate that the system successfully provides accurate recommendations based on user input, even without prior viewing history. The simple user interface improves usability and enhances the overall experience in selecting relevant content. The combination of *Content-Based Filtering* and the *K-NN* algorithm proves effective in delivering personalized recommendations and optimizing the viewing experience amidst a wide array of digital content choices.

**Keywords:** Recommendation System, *Content-Based Filtering*, *K-Nearest Neighbors (KNN)*, Korean Drama.

### **1. PENDAHULUAN**

Perkembangan industri hiburan pada era digital menunjukkan peningkatan yang pesat. Masyarakat menjadikan drama Korea sebagai salah satu bentuk hiburan yang paling diminati. Banyaknya judul yang terus bertambah membuat pengguna mengalami kesulitan dalam menentukan tontonan yang sesuai dengan minat mereka. Sistem rekomendasi dikembangkan untuk membantu pengguna memilih tayangan berdasarkan preferensi individu. Sistem ini berfungsi sebagai alat bantu yang menyaring berbagai pilihan tontonan, sehingga pengguna dapat menemukan drama yang relevan dengan cepat dan memperoleh pengalaman menonton yang lebih memuaskan.

Berbagai platform streaming telah menerapkan sistem rekomendasi dengan tujuan mempermudah pengguna dalam menemukan tayangan yang sesuai dengan minat mereka. Sistem tersebut umumnya menggunakan beberapa metode, seperti *Collaborative Filtering*, *Hybrid Filtering*, dan *Content-Based Filtering*. Di antara ketiganya, metode *Content-Based Filtering* dianggap paling efektif karena menganalisis karakteristik konten, seperti genre, pemeran, serta alur cerita, untuk



menemukan tingkat kemiripan antar drama. Melalui pendekatan ini, sistem dapat memberikan rekomendasi yang lebih personal, relevan, dan efisien dalam membantu pengguna menentukan tontonan yang diinginkan.

Penelitian ini menggabungkan metode *Content-Based Filtering* dengan algoritma K-Nearest Neighbors (KNN) untuk meningkatkan ketepatan hasil rekomendasi. Sistem menggunakan atribut seperti genre, sutradara, aktor, dan tema dalam menghitung tingkat kesamaan antar drama. Setelah pengguna memilih tayangan yang disukai, sistem akan menghasilkan daftar rekomendasi dengan karakteristik yang serupa. Penerapan kombinasi metode ini memungkinkan sistem memberikan hasil yang akurat, personal, serta membantu pengguna menemukan tontonan sesuai minatnya secara lebih cepat dan tepat.

## 2. METODE

### 2.1. Penerapan

Penerapan merupakan aktivitas yang dilakukan untuk mengimplementasikan sistem, metode, atau teknologi tertentu dalam suatu lingkungan guna mencapai sasaran yang telah direncanakan sebelumnya. Dalam konteks organisasi maupun bisnis, penerapan biasanya dimaknai sebagai proses memperkenalkan serta mulai menggunakan sistem informasi, perangkat lunak, atau prosedur baru demi meningkatkan efisiensi kerja dan efektivitas operasional. Kegiatan ini umumnya melalui beberapa tahap, seperti perencanaan awal, proses pengembangan, hingga evaluasi terhadap pelaksanaannya. Keberhasilan penerapan akan sangat bergantung pada dukungan dari manajemen, kesiapan pengguna melalui pelatihan, serta kemampuan organisasi dalam beradaptasi terhadap perubahan yang muncul. (Alvianto, Adam, & Sodik, 2021).

### 2.2. Metode

Metode dapat didefinisikan sebagai seperangkat langkah atau prosedur yang disusun secara sistematis dan logis guna mencapai tujuan tertentu dalam suatu kegiatan, baik itu penelitian maupun proyek. Dalam konteks penelitian ilmiah, metode berfungsi sebagai pedoman dalam melakukan pengumpulan, pengolahan, serta penafsiran data secara terstruktur. Peneliti memiliki kebebasan untuk memilih pendekatan yang sesuai, seperti metode kualitatif yang fokus pada pemahaman mendalam terhadap fenomena sosial, atau metode kuantitatif yang menitikberatkan pada data numerik dan analisis statistik. Pemilihan metode yang tepat akan berdampak langsung terhadap keakuratan, konsistensi, dan keandalan hasil penelitian yang diperoleh. (Pramudito & Sari, 2023).

### 2.3. Sistem Rekomendasi

Sistem rekomendasi merupakan perangkat lunak yang dirancang untuk membantu pengguna dalam menemukan item yang relevan dengan preferensi mereka. Sistem ini bekerja dengan cara menganalisis berbagai jenis data yang berkaitan dengan pengguna, seperti riwayat pencarian, pembelian, serta penilaian terhadap item tertentu. Melalui pendekatan ini, sistem mampu memberikan saran atau rekomendasi atas item yang kemungkinan besar akan diminati pengguna. Dengan demikian, sistem ini tidak hanya mempermudah proses pencarian, tetapi juga meningkatkan kualitas pengalaman pengguna dalam menggunakan layanan digital, seperti platform streaming, e-commerce, maupun aplikasi berita. (Wahono, Saputra, & Rahman, 2024).

### 2.4. *Content-Based Filtering*

Sistem rekomendasi *Content-Based Filtering* merupakan pendekatan yang memberikan saran kepada pengguna berdasarkan data aktivitas pengguna itu sendiri, bukan dari pengguna lain. Sistem ini mengandalkan informasi seperti item yang pernah dikunjungi, disukai, atau diberi penilaian oleh pengguna untuk menentukan preferensi individu. Berdasarkan preferensi tersebut, algoritma akan mengidentifikasi dan merekomendasikan item yang memiliki karakteristik serupa. Keunggulan utama dari metode ini terletak pada kemampuannya memberikan hasil yang tetap relevan meskipun data



eksplisit pengguna terbatas. Namun, metode ini juga memiliki keterbatasan, khususnya dalam menjangkau variasi minat yang belum terlihat dalam profil pengguna. Contohnya, jika pengguna hanya pernah menunjukkan minat terhadap genre tertentu, sistem cenderung hanya merekomendasikan genre tersebut tanpa mencoba mengenalkan genre lain yang mungkin juga disukai. (Wahono, Saputra, & Rahman, 2024).

## 2.5. K-Nearest Neighbors

Algoritma K-Nearest Neighbors (KNN) merupakan teknik pembelajaran mesin yang digunakan untuk menentukan kelas objek dengan mengandalkan kedekatannya dengan data yang sudah memiliki kategori. Dalam metode ini, kelas objek yang akan diklasifikasikan ditentukan berdasarkan mayoritas kelas dari K tetangga terdekat. Nilai K, yang ditetapkan oleh pengguna, menunjukkan jumlah tetangga yang dipertimbangkan untuk klasifikasi. Proses pengklasifikasian ini mengukur kedekatan objek menggunakan berbagai ukuran jarak, seperti Euclidean, Manhattan, atau Minkowski, untuk menentukan seberapa dekat objek tersebut dengan data lainnya.

Algoritma K-Nearest Neighbors memiliki beberapa jenis klasifikasi yang sering digunakan, yaitu (Khairia dkk, 2022):

1. *Fine KNN*: Yaitu, Pengklasifikasi tetangga terdekat yang membuat perbedaan antara kelas dengan jumlah tetangga diatur ke 1.
2. *Medium KNN*: Yaitu, kelas tetangga terdekat yang membuat perbedaan lebih sedikit dari pada *Fine KNN* dengan jumlah tetangga di atur ke 10.
3. *Course KNN*: Yaitu, Pengklasifikasi tetangga terdekat yang membuat perbedaan kasar antar kelas, dengan jumlah tetangga diatur ke 100.
4. *Cosine KNN*: Yaitu, pengklasifikasi tetangga terdekat yang menggunakan jarak cosinus metrik.
5. *Cubic KNN*: Yaitu, kelas tetangga terdekat yang menggunakan jarak kubik metrik
6. *Weighted KNN*: Yaitu, sebuah kelas tetangga terdekat yang menggunakan pembobotan jarak.

Cosine KNN merupakan metode klasifikasi berbasis tetangga terdekat yang memanfaatkan kesamaan arah antar vektor sebagai acuan utama. Sistem ini menilai kemiripan dua data dengan membandingkan sudut antar vektor, bukan dengan mengukur jarak fisik seperti pada metode Euclidean. Jika dua vektor memiliki sudut yang kecil, maka keduanya dianggap memiliki tingkat kemiripan yang tinggi karena berada pada arah yang hampir sama. (Subroto, Mulyono, Firmansyah, Qomaruddin, & Susila 2022).

## 2.6. Drama Korea

Drama Korea merupakan salah satu bentuk hiburan asal Korea Selatan yang telah meraih popularitas secara global, termasuk di Indonesia. Sejak dikenalkan pada tahun 2002, genre ini berhasil menarik perhatian penonton melalui alur cerita yang memikat, karakter yang kuat, serta kualitas produksi yang tinggi. Tema-tema yang diangkat dalam drama Korea sering kali mencerminkan kehidupan sehari-hari, kisah percintaan, hingga persoalan sosial yang relevan dengan realitas masyarakat, sehingga mudah diterima oleh berbagai kalangan. Selain itu, penyampaian cerita yang penuh emosi juga menjadi daya tarik tersendiri yang mampu membangun ikatan emosional antara penonton dan cerita yang ditampilkan. (Wahyuni, Missriani, & Fitriani, 2022).

# 3. HASIL DAN PEMBAHASAN

## 3.1. Spesifikasi Perangkat Lunak

Perangkat lunak yang digunakan untuk membangun penerapan metode *Content-Based Filtering* pada sistem rekomendasi drama korea menggunakan algoritma *K-Nearest Neighbors (K-NN)* ini adalah sebagai berikut:

1. Microsoft Windows 10
2. Visual Studio Code
3. XAMPP



4. Google Chrome
5. Python
6. PHP (*Hypertext Preprocessor*)
7. Microsoft Excel

### 3.2. Spesifikasi Perangkat Keras

Perangkat keras yang digunakan dalam membangun penerapan metode *Content-Based Filtering* pada sistem rekomendasi drama korea menggunakan algoritma *K-Nearest Neighbors (K-NN)* ini adalah sebagai berikut:

1. Perangkat (*Device*): Lenovo IdeaPad Slim 3
2. Prosesor (*Processor*): AMD Athlon Gold 3150U With Radeon Graphics, 2.40Mhz, 2 Core(s), 4 Logical Processor(s)
3. Ram/Penyimpanan (*RAM/Storage*): 8GB/SSD 512 GB

### 3.3. Perhitungan

Proses perhitungan pada penelitian ini menggunakan bahasa Python. Diawali dengan menginisialisasi library dan data yang akan diproses.

```
import pandas as pd
import sys
import json
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
import os
```

**Gambar 1.** Inisialisasi Data

Program menggunakan library **Pandas**, **Sys**, **JSON**, **Scikit-learn**, dan **OS**. Pandas digunakan untuk membaca data CSV, Sys untuk mengambil input dari PHP, JSON untuk mengirim hasil ke PHP, Scikit-learn untuk membuat model TF-IDF dan menghitung kemiripan antar drama, sedangkan OS untuk mengatur jalur file dataset.

```
# 1. Baca data CSV
base_dir = os.path.dirname(os.path.abspath(__file__))
csv_path = os.path.join(base_dir, '../database/kdrama.csv')
df = pd.read_csv(csv_path, encoding='latin1')

# Bersihkan spasi kolom dan isi kosong
df.columns = df.columns.str.strip()
df['Genre'] = df['Genre'].fillna('')
df['Synopsis'] = df['Synopsis'].fillna('')
df['Rating'] = df['Rating'].fillna(0)
```

**Gambar 2.** Baca Data CSV

Tahap ini membaca file `kdrama.csv` dari folder `../database/`, kemudian disimpan ke dalam *DataFrame* bernama `df`. Data dibersihkan dengan menghapus spasi pada nama kolom dan mengisi nilai kosong pada kolom *Genre*, *Synopsis*, dan *Rating* agar siap digunakan untuk proses selanjutnya.

```
# 2. Gabungkan fitur untuk TF-IDF
df['fitur_konten'] = df['Genre'] + ' ' + df['Synopsis'] + ' Rating:' + df['Rating'].astype(str)
```

**Gambar 3.** Gabungkan Fitur TF – IDF

Kolom *Genre*, *Synopsis*, dan *Rating* digabung menjadi satu kolom baru bernama *fitur\_konten*. Tujuannya agar setiap drama memiliki deskripsi konten yang lengkap untuk proses analisis kemiripan.

```
# 3. Buat TF-IDF matrix
vectorizer = TfidfVectorizer(stop_words='english')
tfidf_matrix = vectorizer.fit_transform(df['fitur_konten'])

# Lowercase untuk pencocokan judul
df['Judul_lower'] = df['Judul'].str.lower()
```

**Gambar 4.** Buat TF - IDF Matrix

Tahap ini mengubah teks dalam kolom *fitur\_konten* menjadi bentuk numerik menggunakan **TF-IDF Vectorizer**, yang mengukur tingkat kepentingan kata dalam setiap drama. Kolom baru *Judul\_lower* dibuat agar pencarian judul tidak sensitif terhadap huruf besar atau kecil.

```
# 4. Fungsi rekomendasi
def rekomendasi_drama(input_judul, min_rating=0, filter_genre="", jumlah_rekomendasi=3):
    input_judul_lower = input_judul.lower()

    if input_judul_lower not in df['Judul_lower'].values:
        return {"error": "Judul tidak ditemukan dalam dataset."}

    index_input = df[df['Judul_lower'] == input_judul_lower].index[0]
    cosine_sim = cosine_similarity(tfidf_matrix[index_input], tfidf_matrix).flatten()

    similar_indices = cosine_sim.argsort()[::-1]
    similar_indices = [i for i in similar_indices if i != index_input]

    rekomendasi = []
    for i in similar_indices:
        drama_genre = df['Genre'][i].lower()
        if float(df['Rating'][i]) >= min_rating:
            if filter_genre == "" or filter_genre in drama_genre:
                rekomendasi.append({
                    'judul': df['Judul'][i],
                    'genre': df['Genre'][i],
                    'rating': df['Rating'][i]
                })
            if len(rekomendasi) == jumlah_rekomendasi:
                break

    return {
        "input_judul": df['Judul'][index_input],
        "genre": df['Genre'][index_input],
        "rekomendasi": rekomendasi
    }
```

**Gambar 5.** Fungsi Rekomendasi

Fungsi `rekomendasi_drama()` adalah inti program. Pertama, program memeriksa apakah judul input ada di dataset. Jika ada, sistem menghitung **cosine similarity** antara drama yang dipilih dengan seluruh drama lain berdasarkan TF-IDF. Kemudian hasil diurutkan dari yang paling mirip, difilter



berdasarkan *rating* dan *genre*, lalu disimpan ke daftar rekomendasi sesuai jumlah yang diminta. Fungsi ini mengembalikan hasil berupa judul input, genre-nya, dan daftar drama rekomendasi.

```
# 5. Main: ambil argumen dari PHP
if name == 'main':
    if len(sys.argv) < 4:
        print(json.dumps({"error": "Argumen judul, rating, dan genre dibutuhkan"}, ensure_ascii=False))
        sys.exit()

    input_user = sys.argv[1]
    try:
        input_rating = float(sys.argv[2])
    except ValueError:
        input_rating = 0.0

    input_genre = sys.argv[3].lower()

    hasil = rekomendasi_drama(input_user, min_rating=input_rating, filter_genre=input_genre)
    print(json.dumps(hasil, ensure_ascii=False))
```

**Gambar 6.** Ambil Argumen Dari PHP

Tahap terakhir menjalankan program utama. Bagian ini menerima tiga argumen dari PHP yaitu **judul**, **rating**, dan **genre**. Jika input tidak lengkap, muncul pesan error. Jika lengkap, fungsi `rekomendasi_drama()` dijalankan dan hasilnya dicetak dalam format **JSON** agar dapat dikirim kembali ke halaman web.

```
Administrator@AppmanSE-PC MINGW64 /c:/xampp/htdocs/sistem_rekomendasi/database (master)
$ python rekomendasi.py "Study Group" 8 "Horor"
{"input_judul": "Study Group", "genre": "Aksi, Komedi, Thriller Remaja", "rekomendasi": [{"judul": "All of Us Are Dead", "genre": "Thriller, Horor, Drama, Zombie", "rating": 8.0}, {"judul": "Revenant", "genre": "Horor, Misteri, Thriller, Supranatural", "rating": 8.3}, {"judul": "Sweet Home", "genre": "Horor, Thriller, Drama, Fantasi, Apokaliptik", "rating": 8.7}]}
```

**Gambar 7.** Uji Coba Rekomendasi

Setelah program dijalankan melalui terminal dengan perintah `python rekomendasi.py "Study Group" 8 "Horor"`, hasil yang ditampilkan sesuai dengan rekomendasi yang diberikan oleh sistem. Artinya, program berhasil menampilkan daftar drama yang relevan berdasarkan judul input, genre, dan batas minimal rating yang telah ditentukan.

## 4. KESIMPULAN

Penelitian ini membahas penerapan metode Content-Based Filtering pada sistem rekomendasi Drama Korea dengan menggunakan algoritma K-Nearest Neighbors (K-NN). Hasil analisis dan implementasi menunjukkan bahwa sistem rekomendasi yang dikembangkan mampu menyajikan tayangan yang relevan dengan menghitung kemiripan antar fitur dari berbagai tayangan berdasarkan preferensi pengguna, sehingga dapat menghasilkan rekomendasi yang sesuai tanpa memerlukan data dari pengguna lain. Pengujian dalam bentuk aplikasi berbasis web juga memperlihatkan bahwa input berupa judul, genre, dan batas rating minimum dapat dimanfaatkan dengan baik oleh pengguna. Dengan antarmuka yang sederhana, sistem berhasil menampilkan daftar rekomendasi yang akurat, sehingga dapat disimpulkan bahwa algoritma K-NN dapat diimplementasikan secara efektif dalam pengembangan sistem rekomendasi hiburan berbasis konten.



**JRIIN : Jurnal Riset Informatika dan Inovasi**  
**Volume 3, No. 7, Desember Tahun 2025**  
**ISSN 3025-0919 (media online)**  
**Hal 2035-2041**

## **REFERENCES**

- Agustian, E. R., & Nugroho, E. P. (2020). Sistem rekomendasi film menggunakan metode collaborative filtering dan k-nearest neighbors. *JATIKOM: Jurnal Aplikasi dan Teori Ilmu Komputer*, 3(1), 18-21.
- Arfisko, H. H., & Wibowo, A. T. (2022). Sistem Rekomendasi Film Menggunakan Metode Hybrid Collaborative Filtering Dan Content-based Filtering. *eProceedings of Engineering*, 9(3),2149-2159.
- Fajriansyah, M., Adikara, P. P., & Widodo, A. W. (2021). Sistem Rekomendasi Film Menggunakan Content Based Filtering. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 5(6), 2188-2199.
- Fitriani, A., Rohmani, A., & Widjarto. (2020). Sistem Rekomendasi Film Berbasis Website dengan Metode Prototype Menggunakan Algoritma K-Nearest Neighbors (KNN). *Jurnal Teknologi Informasi*, 12(3), 45–60.
- Wahono, A. R., Saputra, B. A., & Rahman, F. F. (2024). Sistem Rekomendasi Film Menggunakan Metode Content-Based Filtering dan Algoritma K-Nearest Neighbors (KNN). In *Prosiding Seminar Nasional Teknologi Informasi dan Bisnis* (pp. 1-6).