

**Sistem Rekomendasi Film Menggunakan Metode *Hybrid Collaborative Filtering* Dan *Content-Based Filtering***

**Tugas Akhir**

**diajukan untuk memenuhi salah satu syarat**

**memperoleh gelar sarjana**

**dari Program Studi S1 Informatika**

**Fakultas Informatika**

**Universitas Telkom**

**1301170142**

**HILMI HIDAYAT ARFISKO**



**Program Studi Sarjana Informatika**

**Fakultas Informatika**

**Universitas Telkom**

**Bandung**

**2021**

## LEMBAR PENGESAHAN

**Sistem Rekomendasi Film Menggunakan *Metode Hybrid Collaborative Filtering* Dan  
*Content-Based Filtering***

***Movie Recommendation System Using Hybrid Collaborative Filtering and Content-Based  
Filtering***

**NIM : 1301170142**

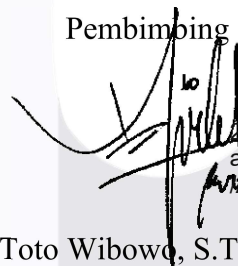
**Hilmi Hidayat Arfisko**

Tugas akhir ini telah diterima dan disahkan untuk memenuhi sebagai syarat memperoleh gelar  
pada Program Studi Sarjana Informatika  
Fakultas Informatika  
Universitas Telkom

Bandung, 26 Januari 2022

Menyetujui

Pembimbing 1



20200202 11.25  
approval laporan TA  
ATA PW

Agung Toto Wibowo, S.T., M.T., Ph.D.

NIP: 06810035

Ketua Program Studi  
Sarjana Informatika



Dr. Erwin Budi Setiawan, S.Si., M.T.

NIP: 00760045

## LEMBAR PERNYATAAN

Dengan ini saya, Hilmi Hidayat Arfisko, menyatakan sesungguhnya bahwa Tugas Akhir saya dengan judul Sistem Rekomendasi Film Menggunakan Metode Hybrid Collaborative Filtering Dan Content-Based Filtering beserta dengan seluruh isinya adalah merupakan hasil karya sendiri, dan saya tidak melakukan penjiplakan yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan. Saya siap menanggung resiko/sanksi yang diberikan jika di kemudian hari ditemukan pelanggaran terhadap etika keilmuan dalam Laporan TA atau jika ada klaim dari pihak lain terhadap keaslian karya,

Bandung, 26 Januari 2022

Yang menyatakan



Hilmi Hidayat Arfisko

## Sistem Rekomendasi Film Menggunakan Metode *Hybrid Collaborative Filtering Dan Content-Based Filtering*

Hilmi Hidayat Arfisko<sup>1</sup>, Agung Toto Wibowo<sup>2</sup>

<sup>12</sup>Fakultas Informatika, Universitas Telkom, Bandung

<sup>1</sup>hilmiha@students.telkomuniversity.ac.id, <sup>2</sup>agungtoto@telkomuniversity.ac.id

---

### Abstrak

Sistem rekomendasi pada dasarnya merupakan sistem yang berguna untuk menyaring dan mengidentifikasi item berupa produk, layanan atau informasi yang memiliki potensi besar untuk dipilih, dibeli ataupun digunakan oleh pengguna. Terdapat beberapa metode yang dapat digunakan dalam membangun sistem rekomendasi, seperti *collaborative filtering* yang merekomendasikan *item* berdasar kemiripan pengguna dalam hal memilih atau memberi nilai kepada *item* dan *content-based filtering* yang merekomendasikan *item* berdasarkan kemiripan *item* dalam hal isi atau konten *item* yang disukai oleh pengguna. Namun perlu diketahui, masing-masing metode ini memiliki kelemahan dan kelebihan. Untuk menutupi kelemahan masing-masing metode ini, pendekatan *hybrid* dapat dilakukan dimana kedua metode ini digabungkan dengan harapan dapat mengurangi kelemahan dari satu metode melalui kelebihan dari satu metode lainnya dan menghasilkan rekomendasi yang lebih baik. Oleh karena itu dalam penelitian ini dilakukan pembuatan sistem rekomendasi film menggunakan metode *hybrid collaborative filtering* dan *content-based filtering*. Dalam pengujiannya, hasil rekomendasi metode ini dibandingkan dengan hasil rekomendasi ketika hanya menggunakan metode *collaborative filtering* saja, metode *content-based filtering* saja dan metode *hybrid* dengan kedua metode tersebut dibalik. Dapat disimpulkan bahwa hasil pengujian yang dilakukan menggunakan metode *hybrid collaborative filtering* dan *content-based filtering* menghasilkan list rekomendasi item film yang lebih baik dibandingkan 3 metode lainnya yang diujicobakan terhadap keseluruhan pengguna dalam dataset pengujian.

**Kata kunci:** sistem rekomendasi film, pendekatan *hybrid*, *collaborative filtering*, *content-based filtering*

---

### Abstract

The recommendation system is basically a system that is useful for filtering and identifying items in the form of products, services or information that have great potential to be selected, purchased or used by users. There are several methods that can be used in building a recommendation system, such as collaborative filtering which recommends items based on user similarity in terms of selecting or assigning value to items and content-based filtering which recommends items based on item similarity in terms of content or item content liked by users. However, it should be noted that each of these methods has advantages and disadvantages. To cover the weaknesses of each of these methods, a hybrid approach can be used where the two methods are combined in the hope of reducing the weaknesses of one method through the advantages of one method and producing better recommendations. Therefore, in this study, a film recommendation system was developed using hybrid collaborative filtering and content-based filtering methods. In the test, the recommended results of this method are compared with the recommendations when using only the collaborative filtering method, the content-based filtering method and the hybrid method with the two reversed. It can be concluded that the results of the tests carried out using the hybrid collaborative filtering method and content-based filtering resulted in a list of recommended film items that was better than the other 3 methods that were tested on all users in the test dataset.

**Keywords:** movie recommendation system, hybrid approach, collaborative filtering, content-based filtering

---

## 1. Pendahuluan

### Latar Belakang

Budaya menonton acara TV dan film sekarang ini dipermudah dengan adanya internet. Platform-platform *streaming* seperti Netflix, Prime Video, HBO Max dan Disney+ memberikan lebih banyak fleksibilitas kepada pengguna untuk menonton acara TV dan film favorit mereka, kapan saja dan di perangkat apa pun. Jumlah acara TV atau film yang disediakan oleh masing-masing platform ini pun bisa terbilang cukup besar, dengan salah satu platform seperti Netflix memiliki katalog *item* berkisaran 6000 item acara TV dan film. Dengan banyaknya jumlah *item* yang disediakan oleh platform-platform ini, sistem rekomendasi menjadi fitur penting untuk dibangun dan memiliki peran besar dalam membantu pengguna menemukan sesuatu *item* relevan yang mungkin mereka sukai. Selain itu, dari sudut pandang bisnis, sistem rekomendasi dapat membantu meningkatkan waktu aktfitas atau

*engagement* oleh pengguna di dalam suatu platform dengan cara menampilkan *item-item* relevan kepada pengguna, yang mana nantinya seiring waktu dapat meningkatkan pendapatan untuk platform itu sendiri. Oleh karena itu, sistem rekomendasi merupakan salah satu fitur penting yang harus dimiliki oleh platform-platform seperti layanan *streaming* acara TV dan film.

Sistem rekomendasi pada dasarnya merupakan sistem yang berguna untuk menyaring dan mengidentifikasi *item* berupa produk, layanan atau informasi yang memiliki potensi besar untuk dipilih, dibeli ataupun digunakan oleh pengguna [1], [2]. Terdapat beberapa metode yang dapat digunakan dalam membangun sistem rekomendasi, seperti *collaborative filtering* (CF) yang merekomendasikan *item* berdasar kemiripan pengguna dalam hal memilih atau memberi nilai kepada *item* dan *content-based filtering* (CBF) yang merekomendasikan *item* berdasarkan kemiripan *item* dalam hal isi atau konten *item* yang disukai oleh pengguna [1], [3], [4]. Namun perlu diketahui, masing-masing metode ini memiliki kelemahan. Misalnya *cold-start problem* pada metode CF untuk pengguna baru dan *item* baru dalam sistem yang disebabkan oleh tidak adanya informasi yang memadai tentang interaksi pengguna atau *item* untuk membuat prediksi yang relevan dan *over-specialization problem* pada metode CBF dimana pengguna dibatasi untuk mendapatkan rekomendasi yang mirip dengan yang sudah dikenal atau ditentukan dalam profilnya saja [1], [3], [4]. Untuk menutupi kelemahan masing-masing metode ini, pendekatan *hybrid* dapat dilakukan dimana kedua metode ini digabungkan dengan harapan dapat mengurangi kelemahan dari satu metode melalui kelebihan dari satu metode lainnya dan menghasilkan sistem rekomendasi yang lebih baik [1], [3], [4].

Oleh karena itu, pada penelitian ini akan dicoba untuk membangun sistem rekomendasi menggunakan pendekatan *hybrid* metode *collaborative filtering* (CF) dan metode *content-based filtering* (CBF) terhadap *item* film. Adapun dataset yang digunakan dalam penelitian ini adalah dataset *movielens* yang disediakan oleh *grouplens.org*. Dataset ini berisi informasi *ratings* untuk film yang dibuat oleh pengguna serta informasi konten berupa genre film dan tag mengenai film oleh pengguna.

### Topik dan Batasannya

Dalam penelitian ini akan dibahas bagaimana mengimplementasikan metode *hybrid collaborative filtering* dan *content-based filtering* (Hybrid CF-CBF) dalam membangun sistem rekomendasi film. Dimana penggunaan metode *hybrid* dengan mengimplementasikan metode CBF terhadap hasil rekomendasi CF yang dipercaya dapat menghasilkan prediksi rekomendasi *item* film yang lebih baik kepada pengguna dibandingkan dengan hanya menggunakan metode CF secara tersendiri.

Adapun dataset yang digunakan adalah dataset *movielens* yang telah diproses sebelumnya. Dataset ini berupa dataset 100.836 rating *item* film dengan 610 pengguna yang minimal telah memberikan rating terhadap 20 *item* film dan 9.742 *item* film. Selain itu terdapat pula dataset konten *item* film yang berisikan 32.977 tag untuk semua *item* film. Batasan pembahasan dari pembuatan tugas akhir ini tidak membahas dan juga membandingkan pembangunan sistem rekomendasi dengan metode lain selain penggunaan metode CF dan CBF, serta hanya berfokus pada pembangunan sistem rekomendasi menggunakan metode *hybrid collaborative filtering* dan *content-based filtering* (Hybrid CF-CBF). Dikarenakan sifat dari kedua metode yang digunakan dalam pendekatan *hybrid* ini, sistem rekomendasi yang dibangun tidak dapat menangani *cold start problem* pada pengguna baru.

### Tujuan

Tujuan dari pembuatan tugas akhir ini adalah membangun sistem rekomendasi film menggunakan pendekatan metode *Hybrid CF-CBF* dan bagaimana dampak pendekatan *hybrid* terhadap keakuratan hasil rekomendasi.

### Organisasi Tulisan

Selanjutnya pada bab 2 akan dibahas mengenai dasar teori yang terkait dengan penelitian yang dilakukan. Bab 3 akan dibahas tentang perancangan sistem rekomendasi yang dibangun. Bab 4 akan membahas evaluasi dari sistem yang dibangun dan terakhir bab 5 akan membahas kesimpulan dari penelitian ini.

## 2. Dasar Teori

### 2.1. Sistem Rekomendasi

Sistem rekomendasi adalah perangkat lunak atau metode yang menghasilkan usulan atau saran berupa *item-item* spesifik yang mungkin menarik bagi suatu pengguna tertentu. Istilah "*item*" dalam sistem rekomendasi digunakan untuk menunjukkan apa yang direkomendasikan oleh sistem kepada pengguna. Dalam sistem rekomendasi, terdapat dua pendekatan dalam menghasilkan usulan atau saran kepada pengguna, yaitu sistem rekomendasi yang bersifat *personalized* dan sistem rekomendasi yang bersifat *non-personalized* [4].

Sistem rekomendasi yang bersifat *non-personalized* menghasilkan usulan atau saran dengan mengevaluasi keseluruhan *item* dalam sistem secara sekaligus tanpa mempertimbangkan preferensi atau minat setiap pengguna [3]. Di dalam sistem ini, preferensi atau minat setiap pengguna tidak dipertimbangkan, sehingga apapun rekomendasi yang diberikan mungkin atau tidak relevan untuk suatu pengguna. Contoh implementasi dalam

membentuk rekomendasi yang bersifat *non-personalized* seperti menampilkan *item-item* yang paling populer atau *item-item* yang baru saja rilis.

Sedangkan sistem rekomendasi yang bersifat *personalized* menghasilkan usulan atau saran dengan mempertimbangkan preferensi atau minat setiap pengguna, sehingga sistem lebih efektif merekomendasikan item tertentu kepada pengguna atau kelompok pengguna [4]. Dalam mengimplementasikan sistem rekomendasi yang bersifat *personalized*, terdapat 3 metode utama yang dapat dilakukan, yaitu *collaborative filtering* (CF), *content-based filtering* (CBF) dan pendekatan *hybrid* [3].

### 2.1.1. Collaborative Filtering

Sistem rekomendasi yang menggunakan metode *collaborative filtering* (CF) bekerja dengan cara mengumpulkan umpan balik pengguna dalam bentuk *rating* atau preferensi *item-item* dan kemudian memanfaatkan kesamaan perilaku antar pengguna dalam memprediksi *rating* untuk menentukan bagaimana suatu *item* direkomendasikan [1], [3], [4]. Salah satu pendekatan dalam membangun sistem rekomendasi CF adalah *neighborhood-based method* yang didasarkan pada fakta bahwa pengguna yang mirip cenderung menunjukkan pola perilaku penilaian yang serupa dan item yang mirip menerima peringkat yang sama [3]. Dalam pendekatan ini dilakukan prediksi *rating* terhadap target *item* dengan cara menghitung kemiripan (*similarity*) menggunakan koleksi *rating item-item* yang sebelumnya telah dinilai oleh pengguna. Pendekatan *neighborhood-based method* dalam CF dapat dibedakan menjadi dua jenis, yaitu user-based CF dan item-based CF [3].

*User-based* CF bekerja dengan mencari hubungan kemiripan (*similarity*) pengguna berdasarkan tabel *rating* untuk membentuk sebuah rekomendasi terhadap *item* kepada pengguna [3]. Metode ini mengasumsikan bahwa jika sekelompok pengguna memiliki preferensi *item* yang mirip dengan seorang target pengguna, maka target pengguna tersebut akan tertarik dengan *item* yang sama.

Sedangkan *item-based* CF bekerja dengan mencari hubungan kemiripan (*similarity*) *item* berdasarkan tabel *rating* untuk membentuk sebuah rekomendasi terhadap *item* kepada pengguna [3]. Metode ini mengasumsikan bahwa jika beberapa *item* diberi nilai yang mirip oleh sekelompok pengguna, maka seorang target pengguna di dalam kelompok akan memberikan penilaian terhadap *item-item* tersebut secara serupa dengan kebanyakan pengguna lain dalam kelompok.

Salah satu cara menghitung kemiripan (*similarity*) pengguna dalam *user-based* CF maupun *item* dalam *item-based* CF adalah dengan menggunakan perhitungan *pearson correlation* [1], [3], [5], [6]. *Pearson correlation* adalah ukuran korelasi linier antara dua objek yang menghasilkan nilai antara +1 dan -1, dimana jika nilai kemiripan bernilai 0, maka kedua objek tidak memiliki korelasi atau tidak memiliki relasi [6]. Jika nilai kemiripan diatas nilai 0 dapat diartikan kedua objek memiliki kenaikan nilai yang sejajar atau memiliki kecenderungan mirip dan jika nilai kemiripan dibawah 0 dapat diartikan kedua objek memiliki penurunan nilai yang sejajar atau memiliki kecenderungan tidak mirip. Nilai kemiripan antar pengguna menggunakan *pearson correlation* dapat dihitung menggunakan persamaan (1).

$$PC(u, v) = \frac{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)^2 \sum_{i \in I_{uv}} (r_{vi} - \bar{r}_v)^2}} \quad (1)$$

Di mana:

$PC(u, v)$	=	Nilai kemiripan pengguna $u$ dengan pengguna $v$
$I_{uv}$	=	Kumpulan item yang diberi rating oleh pengguna $u$ dan $v$
$r_{ui}$ dan $r_{vi}$	=	Nilai rating yang diberikan oleh pengguna $u$ dan pengguna $v$ terhadap item $i$
$\bar{r}_u$ dan $\bar{r}_v$	=	nilai rata-rata rating pengguna $u$ dan pengguna $v$

Sedangkan untuk nilai kemiripan antar item menggunakan *pearson correlation* dapat dihitung menggunakan persamaan (2).

$$PC(i, j) = \frac{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_i)(r_{uj} - \bar{r}_j)}{\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_i)^2 \sum_{u \in U_{ij}} (r_{uj} - \bar{r}_j)^2}} \quad (2)$$

Di mana:

$PC(i, j)$	=	Nilai kemiripan item $i$ dengan item $j$
$U_{ij}$	=	Kumpulan pengguna yang memberi rating terhadap item $i$ dan $j$
$r_{ui}$ dan $r_{uj}$	=	Nilai rating yang diberikan oleh pengguna $u$ dan terhadap item $i$ dan $j$
$\bar{r}_i$ dan $\bar{r}_j$	=	Nilai rata-rata rating item $i$ dan item $j$

Tahap selanjutnya setelah nilai kemiripan pengguna ataupun *item* didapatkan adalah memilih *neighbor* berupa *subset* pengguna/*item* berdasarkan nilai kemiripan yang dihitung pada langkah sebelumnya untuk menentukan nilai prediksi *rating* yang akan diberikan suatu pengguna terhadap suatu *item*. Salah satu metode perhitungan yang digunakan untuk memprediksi *rating* baik untuk *user-based CF* maupun *item-based CF* adalah *weighted average* [2], [3], [6]. Untuk menghitung prediksi menggunakan *weighted average* pada *user-based CF* digunakan persamaan (3).

$$P(a, i) = \bar{r}_a + \frac{\sum_{u \in U} (r_{ui} - \bar{r}_u) w_{au}}{\sum_{u \in U} |w_{au}|} \quad (3)$$

Di mana:

$P(a, i)$	=	Nilai prediksi pengguna $a$ terhadap item $i$
$U$	=	Kumpulan <i>neighbor</i> pengguna $a$
$r_{ui}$	=	Nilai <i>rating</i> item $i$ oleh pengguna $u$
$\bar{r}_a$ dan $\bar{r}_u$	=	Nilai rata-rata <i>rating</i> pengguna $a$ dan pengguna $u$

Sedangkan untuk menghitung prediksi menggunakan *weighted average* pada *item-based collaborative filtering* digunakan persamaan (4).

$$P(u, i) = \frac{\sum_{n \in N} r_{un} w_{in}}{\sum_{n \in N} |w_{in}|} \quad (4)$$

Di mana:

$P(u, i)$	=	Nilai prediksi pengguna $u$ terhadap item $i$
$N$	=	Kumpulan <i>neighbor</i> item $i$
$r_{un}$	=	Nilai <i>rating</i> item $n$ oleh pengguna $u$
$w_{in}$	=	Nilai <i>similarity</i> item $i$ dengan item $n$

Nantinya item-item yang memiliki nilai prediksi tertinggi dalam *neighbor* akan dijadikan rekomendasi.

Hal lain yang perlu diketahui mengenai sistem rekomendasi CF adalah sistem dengan metode ini memiliki beberapa kelebihan dan kekurangan yang dapat dilihat pada *Table 1*. Salah satu kelebihan dari sistem rekomendasi dengan metode ini adalah *serendipity*, dimana sistem dapat merekomendasikan item baru yang tidak hanya relevan dan baru bagi target pengguna, tetapi juga berbeda secara signifikan dari item yang telah dinilai pengguna sebelumnya [1]. Hal ini dikarenakan sistem membandingkan item-item yang telah dinilai target pengguna dengan pengguna lain.

Table 1. Kelebihan dan kekurangan metode *collaborative filtering*

Sistem Rekomendasi <i>Collaborative Filtering</i>	
Kelebihan	Kekurangan
<ul style="list-style-type: none"> <li>Hasil rekomendasi yang beragam dan bersifat <i>serendipitous</i></li> </ul>	<ul style="list-style-type: none"> <li><i>Cold start problem</i> pengguna baru dan <i>item</i> baru</li> <li><i>Sparse problem</i></li> </ul>

Sedangkan untuk kekurangan dari penggunaan metode ini salah satunya adalah *cold-start problem* pada pengguna baru dan *item* baru dalam sistem. *cold-start problem* adalah masalah dimana sistem rekomendasi tidak dapat memberikan rekomendasi dikarenakan belum adanya informasi mengenai preferensi pengguna atau informasi interaksi pengguna dengan item, yang mana masalah ini dapat terjadi pada pengguna baru atau item baru dalam sistem [1], [3], [4]. Masalah lain yang dapat mempengaruhi kinerja sistem rekomendasi dengan metode CF adalah *sparsity* data yang dikarenakan informasi *rating* pengguna terhadap *item* yang tersedia kurang memadai [1], [3], [6]. Secara umum, dalam sistem rekomendasi CF terdapat banyak *item* dan pengguna, sehingga sangat sulit bagi setiap pengguna untuk menilai setiap *item* atau setiap *item* dinilai oleh setiap pengguna. Hal ini mengakibatkan adanya informasi *rating* pengguna terhadap *item* yang kosong dalam matriks *rating*. Rekomendasi yang dihasilkan menggunakan matriks *rating* yang jarang akan mempengaruhi keakuratan algoritma [3].

### 2.1.2. Content-Based Filtering

Sistem rekomendasi yang menggunakan metode *content-based filtering* (CBF) bekerja dengan menggunakan informasi profil pengguna yang terkait dengan preferensi pengguna terhadap *item* sebelumnya untuk nantinya dicari *item* dengan jenis yang mirip sebagai hasil sistem rekomendasi [1], [3], [4]. Sistem rekomendasi



CBF sebagian besar dirancang untuk merekomendasikan *item* berbasis teks, sehingga dalam hal ini konten dapat berupa kata kunci seperti kategori *item*, tag, genre dan lain-lain [3].

Salah satu pendekatan dalam membangun sistem rekomendasi CBF adalah membangun profil pengguna dan profil *item* menggunakan konten yang telah diberi nilai [7]. Dalam hal ini profil pengguna memuat nilai bobot yang mewakili minat pengguna untuk setiap konten di dalam *item-item* yang pernah dinilai pengguna. Sedangkan profil *item* memuat nilai bobot yang mewakili seberapa relevan konten-konten yang ada terhadap *item*.

Untuk membangun matriks profil *item* dapat digunakan TF-IDF (term frequency-inverse document frequency) yang merupakan statistik numerik yang mencerminkan betapa pentingnya sebuah kata untuk dokumen di dalam koleksi [1], [3]. Nilai TF-IDF didapat dari perhitungan TF (*term frequency*) yang melambangkan frekuensi dari istilah subjek muncul dalam dokumen, dan IDF yang melambangkan frekuensi dokumen yang mengandung istilah subjek dalam koleksi. Adapun nilai TF untuk suatu istilah dalam dokumen dapat dihitung menggunakan persamaan (5).

$$tf(t, d) = 1 + \log(f_{td}) \quad (5)$$

Di mana:

$$\begin{aligned} tf(t, d) &= \text{Nilai TF untuk istilah } t \text{ dalam dokumen } d \\ f_{td} &= \text{Jumlah istilah } t \text{ dalam dokumen } d \end{aligned}$$

Dan adapun nilai IDF untuk suatu istilah di dalam koleksi dapat dihitung menggunakan persamaan (6).

$$idf(t) = \log\left(\frac{N}{df_t}\right) \quad (6)$$

Di mana:

$$\begin{aligned} idf(t) &= \text{Nilai IDF untuk istilah } t \text{ dalam koleksi} \\ df_t &= \text{Jumlah dokumen yang mengandung istilah } t \\ N &= \text{Total jumlah dokumen dalam koleksi} \end{aligned}$$

Setelah nilai TF dan IDF didapatkan, bobot TF-IDF untuk istilah *t* di dalam dokumen *d* dapat dihitung menggunakan persamaan (7).

$$tf-idf(t, d) = tf(t, d) \times idf(t) \quad (7)$$

Di mana:

$$\begin{aligned} tf-idf(t, d) &= \text{Nilai TF-IDF untuk istilah } t \text{ untuk dokumen } d \text{ dalam koleksi} \\ tf(t, d) &= \text{Nilai TF untuk istilah } t \text{ dalam dokumen } d \\ idf(t) &= \text{Nilai IDF untuk istilah } t \text{ dalam koleksi} \end{aligned}$$

Setelah matriks profil *item* dibuat, profil pengguna dapat dibangun. Dikarenakan profil pengguna menunjukkan tingkat preferensi pengguna terhadap *item*, profil pengguna dapat dibangun menggunakan jumlah total dari vektor fitur/konten untuk semua *item* yang dinilai positif atau relevan oleh pengguna [8].

Menggunakan vektor profil *item* dengan vektor profil pengguna yang telah dibangun, dapat dilakukan prediksi nilai ketertarikan pengguna terhadap isi konten suatu *item* apakah sesuai dengan preferensi pengguna untuk nantinya direkomendasikan [8], [9]. Salah satu cara menghitung prediksi antara dua variabel ini adalah dengan menggunakan *dot product* antar vektor profil *item* dengan vektor profil pengguna [8]. Adapun perhitungan *dot product* dapat dihitung menggunakan persamaan (8).

$$P(u, v) = \sum_{i=1}^n u_i v_i \quad (8)$$

Di mana:

$$\begin{aligned} P(u, v) &= \text{Nilai prediksi pengguna } u \text{ terhadap item } v \\ u \text{ dan } v &= \text{Vektor } u \text{ dan vektor } v \end{aligned}$$

Nantinya *item-item* yang memiliki nilai similarity tertinggi akan dijadikan rekomendasi.

Hal lain yang perlu diketahui mengenai sistem rekomendasi CBF adalah sistem dengan metode ini memiliki beberapa kelebihan dan kekurangan yang dapat dilihat pada *Table 2*. Salah satu kelebihan dari penggunaan metode ini adalah sistem tidak memerlukan data apa pun tentang pengguna lain atau *user-independent*, dikarenakan rekomendasi hanya berdasar pada preferensi target pengguna terhadap fitur/konten yang disukainya [1], [3]. Hal



ini memudahkan ketika sistem memiliki jumlah pengguna yang besar dan informasi rating item oleh pengguna yang *sparse*. Selain itu metode ini dapat menangkap minat khusus pengguna, dan dapat merekomendasikan item khusus yang sangat sedikit diminati pengguna lain atau dipilih oleh pengguna lain sebelumnya (tidak terjadi *cold start problem* untuk item baru) [1].

Table 2. Kelebihan dan kekurangan metode *content-based filtering*

Sistem Rekomendasi <i>Content-Based Filtering</i>	
Kelebihan	Kekurangan
<ul style="list-style-type: none"> <li>• <i>User-independent</i> sehingga tidak mengalami <i>sparse problem</i></li> <li>• Tidak mengalami <i>cold start problem</i> untuk item baru</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Cold start problem</i> untuk pengguna baru</li> <li>• <i>Over-specialization problem</i></li> </ul>

Sedangkan untuk kekurangan dari penggunaan metode ini salah satunya adalah *over-specialization problem* dimana pengguna dibatasi untuk mendapatkan rekomendasi yang mirip dengan yang sudah dikenal atau ditentukan dalam profilnya saja [3]. Hal ini mencegah pengguna menemukan item baru dan opsi lain yang tersedia. Selain itu sistem rekomendasi CBF juga mengalami *cold-start problem*. Berbeda dengan sistem rekomendasi CF, *cold-start problem* dalam sistem rekomendasi CBF hanya terjadi pada pengguna baru saja [3].

### 2.1.3. Pendekatan *Hybrid*

Pendekatan *hybrid* menggabungkan beberapa metode sistem rekomendasi yang berbeda untuk menghasilkan sistem yang lebih baik. Dengan menggabungkan beberapa sistem rekomendasi, diharapkan dapat mengurangi kelemahan dari satu metode melalui kelebihan dari satu metode lain [1], [3], [4]. Misalnya dengan penggunaan metode CBF untuk menangani *cold-start problem* item baru pada metode CF [4]. Atau penggunaan metode CF untuk menangani *over-specialization problem* pada metode CBF [4]. Dan atau mengatasi permasalahan sparsity data pada CF menggunakan CBF yang hanya memerlukan preferensi konten item yang dikonsumsi pengguna saja [4].

Pendekatan *hybrid* pada sistem rekomendasi dapat dibagi menjadi tiga kategori utama, yang masing-masing kategori memiliki beberapa metode pendekatan. Adapun kategori tersebut adalah *monolithic hybridization*, *parallel hybridization*, dan *pipeline hybridization* [10].

Pada *monolithic hybridization* beberapa sistem rekomendasi diimplementasikan dan digabungkan ke dalam satu algoritma untuk menghasilkan kumpulan rekomendasi akhir. Metode *feature combination* dan *feature augmentation* merupakan contoh metode dengan pendekatan *hybrid* kategori ini [10].

Dalam *parallel hybridization*, setiap sistem rekomendasi yang digunakan, berpartisipasi beroperasi secara independen dan masing-masing sistem memiliki hasil rekomendasi tersendiri. Hasil dari beberapa sistem yang ada ini kemudian digabungkan untuk menghasilkan sebuah set rekomendasi akhir. Adapun metode pada pendekatan *hybrid* kategori ini adalah metode *mixed*, *weighted*, dan *switching* [10].

Dalam *pipeline hybridization*, output dari sistem rekomendasi sebelumnya akan menjadi input dari sistem berikutnya dan sistem rekomendasi terakhir akan menghasilkan rekomendasi final untuk pengguna. Sehingga, keluaran dari sistem rekomendasi pertama mempengaruhi semua rantai sistem rekomendasi selanjutnya yang ada di dalam *pipeline*. Secara opsional, sistem rekomendasi berikutnya dapat menggunakan bagian dari data input asli sebagai komponen tambahan dalam menghasilkan outputnya. Adapun metode pada pendekatan *hybrid* kategori ini adalah metode *cascade* dan *meta-level* [10].

Pada penelitian ini, difokuskan pada pembuatan sistem rekomendasi dengan pendekatan *hybrid cascade* terhadap metode CF dan CBF untuk pendekatan *hybrid*nya. Pendekatan *hybrid cascade* adalah contoh proses rekomendasi yang bertahap dimana metode rekomendasi pertama diminta untuk menghasilkan list kasar kandidat item rekomendasi dan metode rekomendasi kedua dan seterusnya menyaring kembali daftar kandidat awal untuk dijadikan list hasil rekomendasi akhir [10], [11]. Dikarenakan sifatnya ini, semua metode kecuali metode pertama dalam pendekatan *hybrid* ini hanya dapat mengubah urutan dan menghilangkan item dalam daftar rekomendasi serta tidak dapat memperkenalkan item baru atau mengembalikan item yang telah dihilangkan dari metode sebelumnya ke dalam daftar rekomendasi [10]. Selain itu, pendekatan *hybrid* dengan metode *cascade* peka terhadap urutan item dalam list rekomendasi, dimana hasil rekomendasi dengan metode *hybrid* CF-CBF tentu akan berbeda dengan hasil rekomendasi *hybrid* CBF-CF.

## 2.2. Pengujian Sistem Rekomendasi

Untuk menguji sistem rekomendasi, terdapat beberapa metrik pengujian yang dapat digunakan. Dalam konteks sistem rekomendasi yang biasanya dibuat untuk menghasilkan *top-N item* teratas kepada pengguna untuk direkomendasikan, diperlukan metrik yang memberi sistem penghargaan untuk merekomendasikan item yang

tidak hanya benar dan relevan, namun berhasil menempatkan *item* paling relevan di posisi atas dan yang kurang relevan di posisi bawah dalam daftar rekomendasi. Oleh karena itu, dalam penelitian ini akan digunakan metrik pengujian *Hit Rate* dan *Mean Average Precision* (MAP).

### 2.2.1. Metrik Pengujian *Hit Rate*

*Hit Rate* (HR) dalam konteks rekomendasi merupakan metrik ya atau tidak, yang melihat apakah ada item relevan dalam list rekomendasi untuk pengguna tertentu. Jika ada item relevan dalam list hasil rekomendasi untuk pengguna tersebut, maka hal tersebut dihitung 1 hit [12]. Adapun perhitungan hit rate terhadap sekumpulan pengguna dalam satu set pengujian dapat dihitung menggunakan persamaan (9).

$$HR = \frac{|U_{hit}^k|}{|U_{all}|} \quad (9)$$

Di mana:

$HR$	=	Nilai hit rate
$U_{hit}^k$	=	Jumlah pengguna dalam pengujian dimana item relevan berada dalam top-k hasil rekomendasi
$U_{all}$	=	Jumlah pengguna dalam pengujian

### 2.2.2. Metrik Pengujian *Mean Average Precision*

*Mean Average Precision* (MAP) merupakan metrik untuk mengukur kinerja model yang melakukan tugas pencarian dokumen/informasi dan deteksi objek [12]. Penggunaan metrik MAP cocok untuk algoritma yang mengembalikan urutan peringkat *item* dalam *list*, dimana setiap item bisa *hit* atau *miss* (relevan atau tidak relevan) untuk suatu pengguna dan *item* yang rendah dalam *list* cenderung tidak digunakan [12]. Sebelum MAP dapat dihitung, diperlukan informasi mengenai *precision*, *precision at k* dan *average precision*.

Table 3. Confusion matrix

		Reality	
		Relevant	Not Relevant
Prediction	Recommended	True Positive (TP)	False Positive (FP)
	Not Recommended	False Negative (FN)	True Negative (TN)

*Precision* adalah rasio banyak item relevan yang direkomendasikan di dalam satu set item yang direkomendasikan [12]–[15]. Menggunakan confusion matrix yang dapat dilihat pada Table 3, *precision* dapat dihitung menggunakan persamaan (10).

$$P = \frac{TP}{TP + FP} \quad (10)$$

Di mana:

$P$	=	Nilai precision
$TP$	=	Jumlah item relevan yang direkomendasikan
$TP + FP$	=	Jumlah item yang direkomendasikan

*Precision at k* atau bisa disebut dengan  $P(k)$ , merupakan perhitungan *precision* dengan mempertimbangkan subset daftar rekomendasi dari peringkat satu hingga  $k$  [13], [14]. Adapun perhitungan *precision at k* dapat dicontohkan seperti pada Table 4.

Table 4. Contoh perhitungan *precision at k*

rank	prediction	result	rank	prediction	result
1	Item 1	True Positive	1	Item 1	True Positive
2	Item 6	False Positive	2	Item 6	False Positive
3	Item 4	False Positive	3	Item 4	False Positive
	Item 7	False Positive	4	Item 7	False Positive
	Item 3	True Positive	5	Item 3	True Positive
	Item 2	True Positive	6	Item 2	True Positive
	Item 5	False Positive		Item 5	False Positive

$$k=3$$

$$P(k) = \frac{1}{3}$$

$$k=6$$

$$P(k) = \frac{3}{6}$$

*Average precision* (AP) merupakan gambaran lebih mengenai kemampuan sistem untuk mengurutkan hasil rekomendasi di dalam daftar/list [13], [14]. Adapun nilai *average precision* untuk  $N$  jumlah item rekomendasi dapat dihitung menggunakan persamaan (11).

$$AP@N = \frac{1}{\min(m, N)} \sum_{k=1}^N P(k) \times rel(k) \quad \text{if } m \neq 0 \quad (11)$$

$$AP = 0 \quad \text{if } m = 0$$

Di mana:

$AP@N$	=	Nilai average precision untuk $N$ jumlah item yang direkomendasikan
$N$	=	Jumlah item yang direkomendasikan
$m$	=	Jumlah item relevan
$P(k)$	=	Nilai <i>Precision at k</i>
$rek(k)$	=	Indikator jika item ke- $k$ relevan ( $rek(k) = 1$ ) atau tidak relevan ( $rek(k) = 0$ )

*Mean Average Precision* (MAP) merupakan gambaran *average precision* untuk semua pengguna [13], [14]. Adapun nilai MAP untuk  $N$  jumlah item yang direkomendasikan dapat dihitung menggunakan persamaan (12).

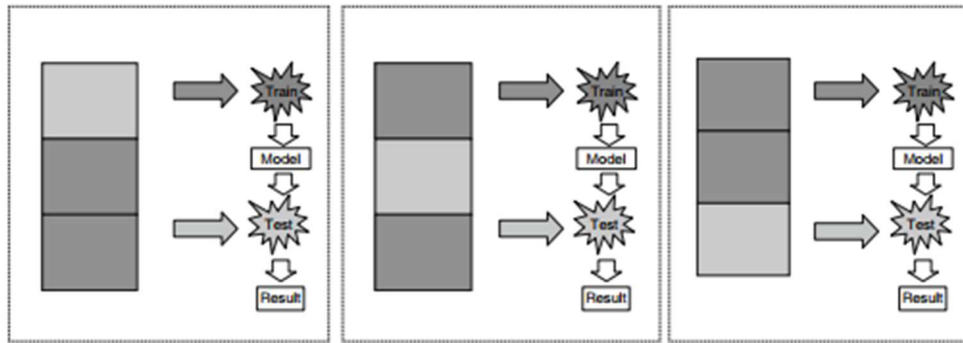
$$MAP@N = \frac{1}{|U|} \sum_{u \in U} (AP@N)_u \quad (12)$$

Di mana:

$MAP@N$	=	Nilai mean average precision untuk $N$ jumlah item yang direkomendasikan terhadap seluruh pengguna.
$N$	=	Jumlah item yang direkomendasikan
$U$	=	Kumpulan pengguna

### 2.2.3. K-Fold Cross Validation

*Cross validation* adalah metode statistik untuk mengevaluasi dan membandingkan algoritma pembelajaran dengan membagi dataset menjadi dua segmen, dimana satu segmen digunakan untuk melatih model, dan segmen lainnya digunakan untuk memvalidasi model [16]. Bentuk dasar dari *cross validation* adalah *k-fold cross validation* dimana dataset pertama-tama dipartisi menjadi  $k$  buah segmen yang berukuran sama atau hampir sama. Selanjutnya sebanyak  $k$  iterasi pelatihan dan validasi dilakukan sedemikian rupa sehingga dalam setiap iterasi, segmen dataset yang berbeda diadakan untuk validasi, sedangkan  $k-1$  yang tersisa digunakan untuk pelatihan model. Pada Gambar 1 dapat dilihat ilustrasi contoh  $k=3$ , dimana segmen data berwarna gelap digunakan sebagai dataset pelatihan model, dan segmen dataset yang terang digunakan untuk validasi.

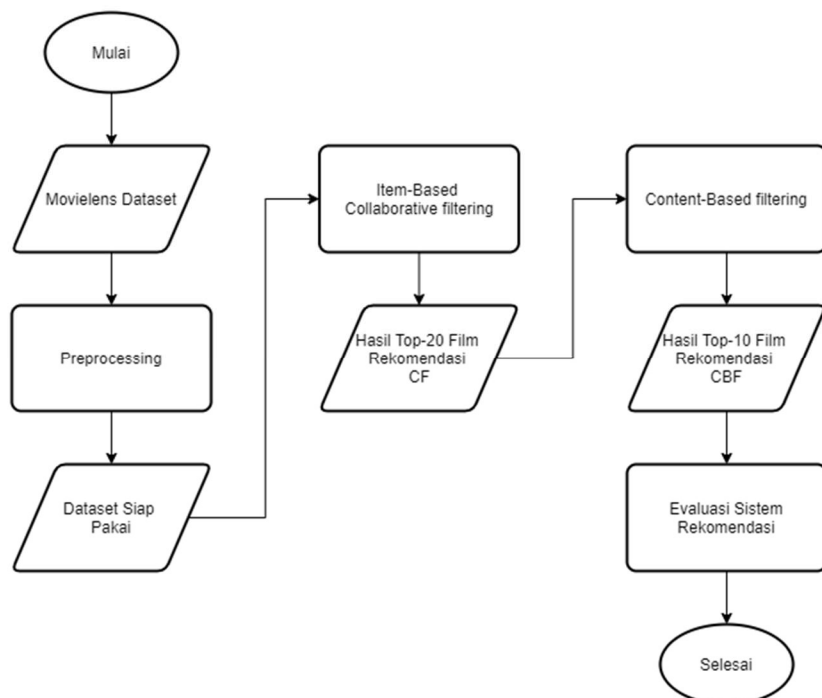


Gambar 1. Ilustrasi pembagian dataset dengan  $k=3$  dalam k-fold cross validation

Dalam mengevaluasi atau membandingkan algoritma pembelajaran menggunakan cross validation dilakukan dengan tahapan sebagai berikut: dalam setiap iterasi, satu atau lebih algoritma pembelajaran menggunakan  $k-1$  segmen dataset untuk mempelajari satu atau lebih model, dan selanjutnya model yang dipelajari diminta untuk membuat prediksi tentang data dalam segmen dataset validasi. Kinerja setiap algoritma pembelajaran pada setiap segmen dilacak menggunakan beberapa perhitungan akurasi seperti metrik evaluasi yang telah ditentukan. Setelah selesai,  $k$  sampel metrik evaluasi akan tersedia untuk setiap algoritma dan penggunaan metodologi seperti rata-rata dapat digunakan untuk mendapatkan ukuran agregat dari sampel metrik evaluasi untuk setiap algoritma [16].

### 3. Sistem yang Dibangun

Pada penelitian ini, metode yang digunakan adalah metode *Hybrid CF-CBF*, dimana jenis pendekatan hybrid yang dilakukan adalah pendekatan cascade. Pada pendekatan hybrid ini, sistem rekomendasi yang dibuat diminta untuk menghasilkan 20 item rekomendasi menggunakan metode *item-based CF*, kemudian dari 20 item tersebut dilakukan pengurutan ulang menggunakan metode CBF untuk nantinya diambil 10 item teratas sebagai hasil rekomendasi akhir. Adapun alur dari sistem rekomendasi yang dibangun dapat dilihat pada *Gambar 2*.



Gambar 2. Alur sistem rekomendasi hybrid CF-CBF yang dibangun

### 3.1. Preprocessing Data

Dalam penelitian ini, digunakan dataset movielens yang disediakan oleh grouplens.org. Dataset ini berisikan *ratings* untuk item film yang diberikan oleh pengguna serta informasi konten item berupa judul, genre dan tag mengenai item film. Dataset-dataset ini nantinya di proses menjadi dataset rating film-pengguna, dan dataset konten item film.

Dataset rating pengguna-film, yang berisi daftar rating-rating item film yang diberi oleh pengguna. Dalam proses pembuatan dataset ini, sebelumnya telah dilakukan pengecekan terhadap dataset movielens untuk menghindari adanya data yang kosong. Adapun contoh representasi dataset rating film-pengguna setelah diproses digambarkan pada *Table 5*.

*Table 5. Representasi dataset rating film-pengguna*

userId	movieId	rating
1	1	4.0
1	3	4.0
:	:	:
610	168252	5.0
610	170875	3.0

Sedangkan untuk dataset konten *item* film, berisi daftar *item* film dan kontennya yang berupa kata yang didapatkan dari judul, genre dan tag-tag mengenai *item* film. Dalam proses pembuatan dataset ini, konten-konten item yang ada telah dilakukan proses pemotongan yang berupa kalimat menjadi kata. Selain itu dilakukan pula penghilangan tanda baca, karakter non-ascii, nomer dan stopwords dari daftar konten item film. Adapun contoh representasi dataset ini digambarkan pada *Tabel 6*.

*Table 6. Representasi dataset konten item film*

movieId	konten
1	"toy"
1	"story"
1	"adventure"
:	:
176329	"comedy"
145994	"soviet"
145994	"classics"

### 3.2. Pembuatan List Rekomendasi Film Metode *Item-Based CF*

Dalam proses ini dilakukan pembuatan list rekomendasi item film menggunakan metode item-based CF. Metode ini digunakan untuk mencari hubungan kemiripan (*similarity*) antar *item* berdasarkan tabel *rating item* untuk membentuk rekomendasi kepada pengguna. Untuk mencari kemiripan ini, digunakan perhitungan *pearson correlation* sesuai dengan persamaan (2). Adapun contoh representasi hasil *pearson correlation* terhadap dua item dapat digambarkan pada *Table 7*.

Table 7. Representasi nilai kemiripan antar item hasil perhitungan pearson correlation

		movieId						
		1	2	3	4	5	...	193587
movieId	1	1.00	0.21	0.13	0.00	0.11	...	0.00
	2	0.21	1.00	0.12	0.01	0.14	...	0.00
	3	0.13	0.12	1.00	0.00	0.28	...	0.00
	4	0.00	0.01	0.00	1.00	0.01	...	0.00
	5	0.11	0.14	0.28	0.01	1.00		0.00
	:	:	:	:	:	:	:	:
	:	:	:	:	:	:	:	:
	193587	0.00	0.00	0.00	0.00	0.00	...	1.00

Tahap selanjutnya setelah nilai kemiripan *item* didapatkan adalah memilih *subset item* film atau *neighbor* berdasarkan nilai kemiripannya untuk digunakan untuk memprediksi rating yang akan diberikan suatu pengguna terhadap item film. Dalam penelitian ini, jumlah *neighbor* yang digunakan pada proses CF adalah 15 buah *item* film.

Nilai prediksi *rating* nantinya dihitung menggunakan *weighted sum* sesuai dengan persamaan (4) yang kemudian untuk setiap pengguna dipilih 20 *item* dengan nilai prediksi rating terbesar untuk dijadikan hasil rekomendasi tahap pertama. List 20 *item* rekomendasi tahap pertama ini nantinya akan dilanjutkan ke proses pengurutan ulang menggunakan metode CBF. Adapun contoh representasi hasil prediksi dari perhitungan *weighted sum* terhadap *item* film dan pengguna digambarkan pada Table 8.

Table 8. Representasi nilai prediksi rating item untuk pengguna hasil perhitungan weighted sum

		userId						
		1	2	3	4	5	...	610
movieId	1	0.00	0.00	0.00	0.83	0.00	...	0.00
	2	0.98	0.00	0.00	0.20	1.10	...	0.67
	3	0.00	0.29	0.00	0.00	0.00	...	0.33
	4	0.00	0.00	0.00	0.00	0.40	...	0.00
	5	0.24	0.00	0.00	0.00	0.00	...	0.00
	:	:	:	:	:	:	:	:
	:	:	:	:	:	:	:	:
	193587	0.00	0.24	0.00	0.00	0.00	...	1.74

### 3.3. Pembuatan Profil Item dan Profil Pengguna untuk Metode CBF

Dalam proses ini, dilakukan pembuatan profil *item* film dan profil pengguna. Profil *item* film memuat nilai bobot yang menggambarkan seberapa relevannya konten-konten yang ada, terhadap *item-item* film. Sedangkan profil pengguna memuat nilai bobot yang menggambarkan minat pengguna terhadap setiap konten di dalam *item-item* relevan bagi pengguna tersebut. Untuk membangun profil *item* film, dilakukan perhitungan TF-IDF terhadap konten untuk *item* film menggunakan persamaan (7). Adapun contoh representasi profil *item* film dapat digambarkan pada Table 9.

Table 9. Representasi profil konten item film

		konten					
		"animation"	"adventure"	"comedy"	"pixar"	...	"action"
MovieId	1	0.37	0.04	0.02	0.76	...	0.00
	2	0.00	0.01	0.00	0.00	...	0.00
	3	0.00	0,00	0.03	0.00	...	0.00
	4	0,00	0,00	0.03	0.00	...	0.00
	:	:	:	:	:	:	:
	:	:	:	:	:	:	:
	193609	0.00	0.00	0.04	0.00	...	0.00

Untuk pembuatan profil pengguna, dapat dilakukan dengan menjumlah total vektor fitur/konten dari semua item yang dinilai positif atau relevan oleh pengguna. Dalam hal ini item yang relevan adalah item film yang diberi rating  $\geq 3$  oleh pengguna. Adapun contoh representasi profil pengguna dapat digambarkan pada Table 10.

Table 10. Representasi profil konten pengguna

		konten					
		"animation"	"adventure"	"comedy"	"pixar"	...	"action"
UserId	1	0.37	0.04	0.02	0.76	...	0.00
	2	0.00	0.01	0.00	0.00	...	0.00
	3	0.00	0,00	0.03	0.00	...	0.00
	4	0,00	0,00	0.03	0.00	...	0.00
	:	:	:	:	:	:	:
	:	:	:	:	:	:	:
	610	0.00	0.00	0.04	0.00	...	0.00

### 3.4. Pembuatan List Rekomendasi Film Metode CBF Menggunakan Hasil Rekomendasi CF

Menggunakan profil pengguna dan profil *item* film, dalam tahap ini dilakukan proses pengurutan ulang 20 *item* dalam *list* hasil rekomendasi metode Cf menggunakan CBF, untuk nantinya diambil 10 item teratas sebagai hasil rekomendasi akhir. Pada tahap ini, vektor profil *item* hasil rekomendasi CF dan vektor profil pengguna digunakan untuk memprediksi konten *item-item* film apakah sesuai dengan preferensi pengguna dan cocok untuk diletakkan pada posisi 10 item teratas atau tidak. Untuk menghitung nilai prediksi ini, digunakan perhitungan *dot product* sesuai dengan persamaan (8). Adapun contoh representasi kesamaan suatu profil pengguna dengan *item-item* dalam *list* hasil rekomendasi tahap sebelumnya digambarkan pada Table 11.

Table 11. Representasi nilai prediksi item untuk suatu pengguna hasil perhitungan dot product

		MovieId						
		2080	1200	2087	2011	1722	...	608
UserId	1	0.00	0.00	0.00	0.83	0.00	...	0.00



*Item* kemudian disusun secara descending berdasarkan hasil perhitungan prediksinya dan diambil 10 *item* teratas dalam *list* untuk kemudian digunakan sebagai hasil rekomendasi akhir terhadap pengguna.

### 3.5. Melakukan Evaluasi Hasil Rekomendasi Akhir

Pengujian sistem yang dibangun dilakukan menggunakan *k-fold cross validation* untuk membagi dataset *rating* film-pengguna menjadi dataset *training* dan dataset *testing*. Digunakan nilai  $k=5$  untuk membagi dataset menjadi 5 bagian, dimana nantinya 4 dari 5 bagian dataset tersebut digunakan dalam proses *training* dan 1 sisanya digunakan dalam proses *testing* yang dilakukan secara preulangan sebanyak 5 kali.

Dalam proses *training*, sistem rekomendasi akan menghasilkan list 20 *item* film sebagai *list* rekomendasi untuk setiap pengguna yang ada dalam dataset *training*. Dari 20 *item* tersebut, hanya 10 *item* teratas didalam *list* yang dijadikan hasil rekomendasi akhir yang kemudian dibandingkan dengan dataset *testing* apakah relevan atau tidak untuk setiap pengguna menggunakan metrik evaluasi *Hit Rate* dan  $MAP@N$  dengan nilai  $N$  pada  $MAP=10$ . Selain pengujian dilakukan pada setiap pengguna dalam dataset *training*, dilakukan pula pengujian terhadap 10 pengguna paling aktif dalam memberikan rating (rata-rata memberi rating terhadap 1313,6 *item*) dan 10 pengguna paling tidak aktif dalam memberikan rating (rata-rata memberi rating terhadap 14,14 *item*). Hal ini bertujuan untuk mengetahui seberapa baik masing-masing metode dalam menghasilkan rekomendasi terhadap pengguna dengan dua kondisi tersebut.

Hasil rekomendasi yang diuji dan nantinya dibandingkan untuk dievaluasi dalam penelitian ini adalah hasil rekomendasi menggunakan metode CF saja tanpa proses *hybrid* (CF) dan CF dengan proses *hybrid* CBF (*Hybrid* CF-CBF). Selain kedua metode itu, dilakukan pula pengujian untuk hasil rekomendasi ketika kedua metode dibalik dalam proses pendekatan *hybrid*nya, sehingga dilakukan pengujian terhadap hasil rekomendasi CBF saja tanpa proses *hybrid* (CBF), dan CBF dengan proses *hybrid* CF (*Hybrid* CBF-CF).

## 4. Evaluasi

Untuk mengevaluasi sistem rekomendasi yang dibangun, digunakan metrik pengujian *Hit Rate* (HR) yang merupakan metrik untuk mengukur kinerja model dalam menghasilkan *item* yang *hit* (relevan) di dalam *list* hasil rekomendasi. Selain itu, digunakan pula metrik evaluasi *Mean Average Precision* (MAP) yang merupakan metrik untuk mengukur kinerja model dalam pengurutan peringkat *item* dalam *list* hasil rekomendasi, dimana setiap *item* di dalam *list* bisa dikategorikan *hit* atau *miss* (relevan atau tidak relevan) untuk suatu pengguna dan *item* yang rendah dalam *list* cenderung tidak digunakan. Adapun tujuan dilakukan pengujian terhadap sistem rekomendasi *hybrid collaborative filtering* dan *content-based filtering* yang dibangun adalah untuk melihat seberapa baik sistem menghasilkan hasil rekomendasi serta melihat perbandingan hasil rekomendasi sebelum adanya pendekatan *hybrid* dan hasil rekomendasi ketika kedua metode tersebut dibalik dalam proses pendekatan *hybrid*nya.

### 4.1. Hasil Pengujian

Pengujian pertama dilakukan untuk menghasilkan list rekomendasi untuk semua pengguna dari 4 metode, yaitu metode CF saja tanpa proses *hybrid* (CF), metode CF dengan proses *hybrid* CBF (*Hybrid* CF-CBF), metode CBF saja tanpa proses *hybrid* (CBF) dan metode CBF dengan proses *hybrid* CF (*Hybrid* CBF-CF). Berdasarkan 5 kali pengulangan *k-fold cross validation*, didapatkan nilai metrik evaluasi *hit rate* untuk keempat metode seperti yang ditampilkan pada Table 12.

Table 12. Hasil perhitungan hit rate percobaan pertama

Iterasi	Hit Rate K-Fold Cross Validation			
	CF	Hybrid CF-CBF	CBF	Hybrid CBF-CF
1	0.8081	0.8114	0.7213	0.7655
2	0.8	0.8032	0.6786	0.7245
3	0.7737	0.7852	0.6918	0.7344
4	0.7770	0.8196	0.7180	0.7409
5	0.79344	0.8016	0.6950	0.7344
Rata-rata	0.7904	<b>0.8042</b>	0.7009	0.74

Dapat dilihat dari hasil percobaan pertama pada Table 12 bahwa metode *Hybrid* CF-CBF memiliki nilai *hit rate* tertinggi diantara metode lain yang dilakukan pengujian, dengan nilai *hit rate* 0,8042. Sedangkan metode CBF

merupakan metode dengan nilai *hit rate* terendah dengan nilai *hit rate* 0,7009. Dari hasil pengujian ini dapat dilihat pula pola nilai *hit rate* yang meningkat untuk metode CF dan CBF setelah dilakukan proses pendekatan *hybrid*. Dari 5 kali perulangan *k-fold cross validation* didapatkan pula nilai metrik evaluasi MAP@10 untuk keempat metode seperti yang ditampilkan pada Table 13.

Table 13. Hasil perhitungan MAP@10 percobaan pertama

Iterasi	MAP@10 K-Fold Cross Validation			
	CF	Hybrid CF-CBF	CBF	Hybrid CBF-CF
1	0.3085	0.3243	0.2293	0.2684
2	0.3051	0.3181	0.2133	0.2522
3	0.3011	0.3077	0.2237	0.2570
4	0.3008	0.3186	0.2195	0.2543
5	0.3004	0.3106	0.2239	0.2560
Rata-rata	0.3054	<b>0.3159</b>	0.2219	0.2575

Dapat dilihat dari hasil percobaan pertama pada Table 13 bahwa metode *Hybrid* CF-CBF memiliki nilai MAP tertinggi diantara metode lain yang dilakukan pengujian dengan nilai MAP 0,3159. Sedangkan metode CBF merupakan metode dengan nilai MAP terendah dengan nilai MAP 0,2219. Dari hasil pengujian ini dapat dilihat pula pola nilai MAP yang meningkat untuk metode CF dan CBF setelah dilakukan proses pendekatan *hybrid*.

Selain pengujian dilakukan pada setiap pengguna dalam dataset, dilakukan pula pengujian terhadap 10 pengguna paling aktif dalam memberikan *rating* (rata-rata memberi *rating* terhadap 1313,6 *item*) dan 10 pengguna paling tidak aktif dalam memberikan *rating* (rata-rata memberi *rating* terhadap 14,14 *item*). Dari 5 kali perulangan *k-fold cross validation* didapatkan nilai rata-rata metrik evaluasi *hit rate* untuk keempat metode seperti yang ditampilkan pada Table 14.

Table 14. Hasil perhitungan hit rate percobaan kedua

Skenario percobaan	Rata-Rata Hit Rate K-Fold Cross Validation			
	CF	Hybrid CF-CBF	CBF	Hybrid CBF-CF
10 pengguna paling aktif	<b>1.0</b>	<b>1.0</b>	<b>1.0</b>	<b>1.0</b>
10 pengguna paling tidak aktif	0.38	<b>0.4399</b>	0.36	0.4

Dapat dilihat dari hasil percobaan kedua pada Table 14 bahwa dalam kasus pembuatan rekomendasi untuk 10 pengguna paling aktif, semua metode memiliki nilai *hit rate* sempurna dengan nilai *hit rate* 1,00.

Dari Table 14 dapat diketahui pula dalam kasus pembuatan rekomendasi untuk 10 pengguna paling tidak aktif, metode *Hybrid* CF-CBF memiliki nilai *hit rate* tertinggi dengan nilai *hit rate* 0,4399 dan metode CBF merupakan metode dengan nilai *hit rate* terendah dengan nilai *hit rate* 0,36. Dari 5 kali perulangan *k-fold cross validation* didapatkan pula nilai rata-rata metrik evaluasi MAP@10 untuk keempat metode seperti yang ditampilkan pada Table 15.

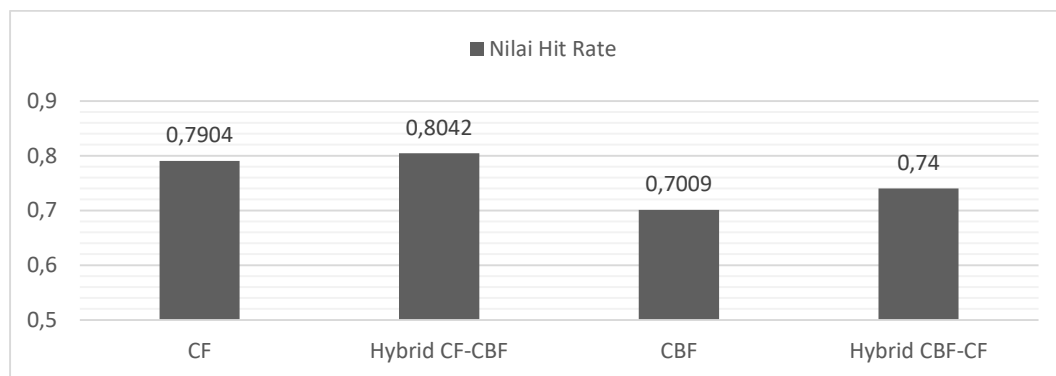
Table 15. Hasil perhitungan MAP@10 percobaan kedua

Skenario percobaan	Rata-Rata MAP@10 K-Fold Cross Validation			
	CF	Hybrid CF-CBF	CBF	Hybrid CBF-CF
10 pengguna paling aktif	0.7886	<b>0.7924</b>	0.7502	0.7818
10 pengguna paling tidak aktif	0.1307	<b>0.1392</b>	0.0911	0.0993

Dapat dilihat dari hasil percobaan kedua pada *Table 15* bahwa dalam kasus pembuatan rekomendasi untuk 10 pengguna paling aktif, metode Hybrid CF-CBF memiliki nilai MAP tertinggi dengan nilai MAP 0,7924. Sedangkan metode CBF merupakan metode dengan nilai hitrate terendah dengan nilai hitrate 0,7502. Dari table tersebut juga dapat diketahui pula dalam kasus pembuatan rekomendasi untuk 10 pengguna paling tidak aktif, metode Hybrid CF-CBF memiliki nilai MAP tertinggi dengan nilai MAP 0,1392 dan metode CBF merupakan metode dengan nilai MAP terendah dengan nilai MAP 0,0911.

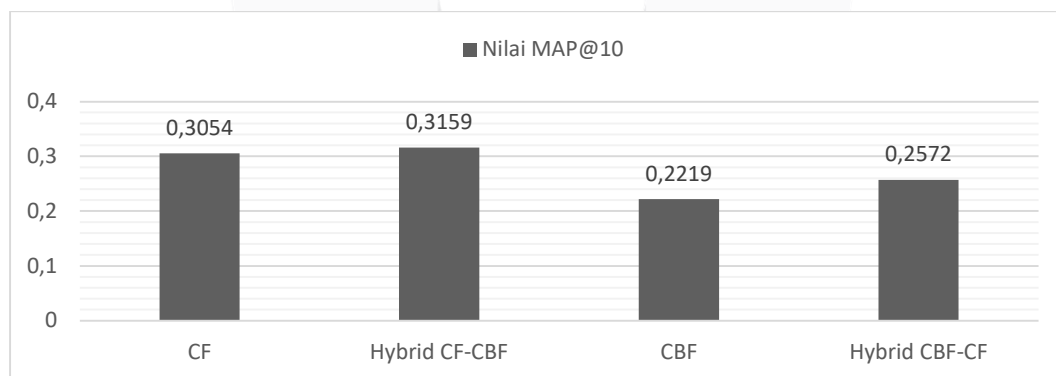
#### 4.2. Analisis Hasil Pengujian

Berdasarkan hasil pengujian yang dilakukan untuk menghasilkan *list* rekomendasi untuk semua pengguna pada *Table 12* dan *Gambar 4*, dari empat metode yang dijalankan, metode Hybrid CF-CBF memiliki nilai *hit rate* tertinggi diantara metode lain dengan nilai *hit rate* 0,8042. Hal ini menunjukkan metode tersebut berhasil menampilkan setidaknya satu item relevan dalam *list* hasil rekomendasi terhadap 80,4% dari keseluruhan pengguna dalam dataset training disetiap iterasi *k-fold cross validation*.



Gambar 4. Hasil perhitungan hit rate percobaan pertama

Selain itu dalam pengujian ini dapat dilihat pula nilai *hit rate* yang meningkat untuk hasil metode CF dan CBF setelah dilakukan proses pendekatan *hybrid*, dengan metode Hybrid CF-CBF memiliki nilai hitrate lebih besar dibandingkan Hybrid CBF-CF. Hal ini menunjukkan bahwa dengan adanya penyusunan ulang 20 *item* hasil rekomendasi CF atau CBF menggunakan pendekatan *hybrid*, dapat memunculkan *item-item* relevan yang mungkin tersembunyi di dalam *list* rekomendasi ke posisi lebih atas sehingga masuk ke dalam 10 *item* rekomendasi akhir. Hal ini pula yang menjadi alasan mengapa nilai MAP yang meningkat untuk metode CF dan CBF setelah dilakukan proses pendekatan *hybrid*, dengan metode Hybrid CF-CBF memiliki nilai MAP lebih besar dibandingkan Hybrid CBF-CF, sesuai dengan hasil pengujian pada *Table 13* dan *Gambar 5*.

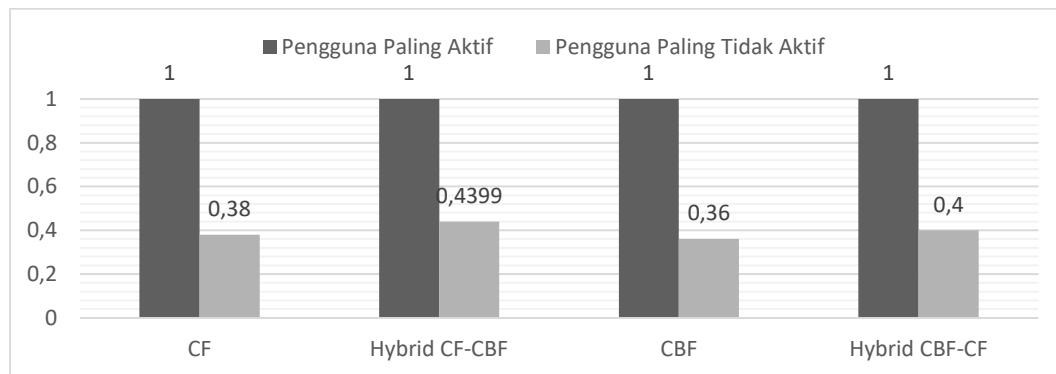


Gambar 5. Hasil perhitungan MAP@10 percobaan pertama

Pada *Table 13* dan *Gambar 5*, dapat dilihat pula bahwa metode Hybrid CF-CBF memiliki nilai MAP tertinggi diantara metode lain dengan nilai MAP 0,3159. Hal ini menunjukkan bahwa metode tersebut dapat menghasilkan urutan hasil rekomendasi yang lebih baik dibandingkan dengan 3 metode lainnya, dimana *item-item* yang mungkin relevan untuk pengguna berada di posisi teratas dan *item* yang mungkin kurang relevan berada di posisi bawah dalam *list* rekomendasi akhir.

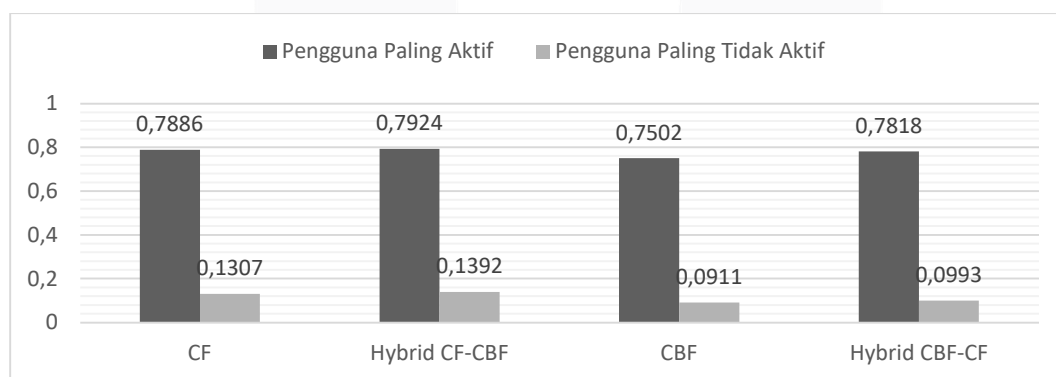
Dari kedua perhitungan evaluasi *hit rate* dan MAP yang telah dilakukan terhadap keempat metode dalam menghasilkan rekomendasi film untuk semua pengguna dalam dataset, didapatkan metode Hybrid CF-CBF dapat menghasilkan *list* rekomendasi *item* film yang paling baik dibandingkan dengan metode lainnya dengan *item* yang relevan berada pada posisi teratas bagi pengguna. Adapun penyebab mengapa metode Hybrid CF-CBF memiliki

nilai hitrate dan MAP yang lebih besar dibandingkan metode *Hybrid* CBF-CF dikarenakan sifat dari metode cascade dalam pendekatan hybrid yang dibangun, yang peka terhadap urutan *item* dalam *list* rekomendasi dimana *list* hasil rekomendasi metode pertama dalam pendekatan *hybrid* akan menjadi *list* rekomendasi kasar yang hanya akan diubah urutannya *itemnya* atau dihilangkan *itemnya* dari *list* oleh metode rekomendasi selanjutnya menjadi *list* hasil rekomendasi akhir. Hal ini menyebabkan kualitas hasil rekomendasi akhir dari pendekatan *hybrid* ini akan dipengaruhi oleh seberapa baik metode pertama dalam pendekatan *hybrid* dalam menghasilkan *list* rekomendasi kasarnya.



Gambar 6. Hasil perhitungan hit rate percobaan kedua

Berdasarkan hasil pengujian yang dilakukan untuk menghasilkan *list* rekomendasi untuk kasus 10 pengguna paling aktif pada Table 14 dan Gambar 6, metode CF, metode Hybrid CF-CBF, metode CBF dan metode Hybrid CBF-CF memiliki nilai *hit rate* sempurna dengan nilai *hit rate* 1,00. Hal ini menunjukkan bahwa keempat metode tersebut dapat menampilkan setidaknya satu *item* relevan dalam *list* hasil rekomendasi untuk seluruh 10 pengguna paling aktif disetiap iterasi *k-fold cross validation*. Namun hal ini berubah ketika keempat metode diminta untuk menghasilkan *list* rekomendasi untuk kasus 10 pengguna paling tidak aktif seperti yang dapat dilihat pada Table 14 dan Gambar 6, dimana nilai *hit rate* untuk setiap metode menurun drastis jika dibandingkan dengan hasil rekomendasi untuk kasus 10 pengguna paling aktif.



Gambar 7. Hasil perhitungan MAP@10 percobaan kedua

Pola yang sama dapat pula dilihat pada nilai MAP hasil rekomendasi 10 pengguna paling tidak aktif yang menurun jika dibandingkan dengan nilai MAP hasil rekomendasi 10 pengguna paling aktif, sesuai dengan hasil pengujian pada Table 15 dan Gambar 7. Hal ini menunjukkan bahwa jumlah *item* yang dirating oleh pengguna dapat mempengaruhi kualitas hasil rekomendasi keempat metode. Semakin banyak *item* yang telah dirating oleh pengguna, maka semakin baik pula kualitas hasil rekomendasi yang dihasilkan oleh masing-masing metode diuji.

Berdasarkan Table 14 dan Gambar 6 ketika dihadapkan dengan kasus 10 pengguna paling tidak aktif, dapat dilihat pola nilai *hit rate* yang meningkat untuk hasil metode CF dan CBF setelah dilakukan proses pendekatan *hybrid*, dengan metode Hybrid CF-CBF memiliki nilai hitrate lebih besar dibandingkan Hybrid CBF-CF. Hal ini menunjukkan meskipun dengan keadaan dimana pengguna baru memberikan *rating* terhadap sedikit *item* film, dengan adanya penambahan pendekatan *hybrid* terhadap hasil rekomendasi CF ataupun CBF dapat meningkatkan kualitas hasil rekomendasi jika dibandingkan dengan hanya menggunakan kedua metode tersebut secara tersendiri. Selain itu perlu dilihat pula dalam hal pengurutan *item* relevan dalam *list* rekomendasi akhir pada kasus ini, seperti yang dapat dilihat pada table Table 15 dan Gambar 7, terdapat pola yang sama dimana adanya peningkatan nilai MAP untuk hasil metode CF dan CBF setelah dilakukan proses pendekatan *hybrid*.

## 5. Kesimpulan

Penulisan tugas akhir ini berfokus pada bagaimana membangun sistem rekomendasi film menggunakan pendekatan *hybrid collaborative filtering* dan *content-based filtering* (Hybrid CF-CBF) serta bagaimana dampak pendekatan *hybrid* terhadap keakuratan hasil rekomendasi. Dalam penelitian ini, digunakan dataset movielens yang disediakan oleh grouplens.org, dimana dataset yang telah dilakukan *preprocessing* digunakan untuk menghasilkan list 20 item film sebagai hasil rekomendasi menggunakan metode *item-based collaborative filtering* dan kemudian hasil rekomendasi tersebut diproses kembali menggunakan metode *content-based filtering* sebagai pendekatan *hybrid* menjadi list 10 item film rekomendasi akhir. Selain dilakukan pengujian terhadap metode tersebut, dilakukan pula pengujian terhadap hasil rekomendasi metode *collaborative filtering* saja (CF), metode *content-based filtering* saja (CBF) serta metode ketika pendekatan *hybrid* kedua metode dibalikan (*Hybrid CBF-CF*). Berdasarkan hasil pengujian yang telah dilakukan terhadap keempat metode tersebut, didapatkan kesimpulan sebagai berikut:

1. Metode Hybrid CF-CBF yang dibangun menghasilkan list rekomendasi item film yang paling baik dibandingkan tiga metode lain yang diujikan berdasarkan perhitungan evaluasi metrik hit rate dan MAP.
2. Jumlah item yang dirating oleh pengguna dapat mempengaruhi hasil rekomendasi. Semakin banyak item yang telah dirating oleh pengguna, maka semakin baik masing-masing metode dalam menghasilkan hasil rekomendasi yang berkualitas.
3. Penggunaan pendekatan hybrid terhadap hasil rekomendasi CF ataupun CBF dapat membantu dalam menghasilkan hasil rekomendasi yang lebih baik ketiha dihadapkan dengan kasus pengguna yang baru baru memberikan *rating* terhadap sedikit *item* film.
4. Penambahan pendekatan *hybrid* terhadap hasil rekomendasi CF ataupun CBF dapat meningkatkan kualitas hasil rekomendasi jika dibandingkan dengan hanya menggunakan kedua metode tersebut secara tersendiri. Dengan adanya penyusunan ulang 20 item hasil rekomendasi CF atau CBF menggunakan pendekatan *hybrid*, dapat memunculkan *item-item* relevan yang mungkin tersembunyi di dalam list rekomendasi ke posisi lebih atas sehingga masuk ke dalam 10 *item* rekomendasi akhir. Hal ini menghasilkan list rekomendasi yang lebih baik dengan item-item yang mungkin relevan untuk pengguna berada pada posisi teratas.
5. Dikarenakan sifat dari metode cascade dalam pendekatan *hybrid* yang dibangun, kualitas hasil rekomendasi akhir dari pendekatan *hybrid* CF-CBF ataupun *hybrid* CBF-CF akan dipengaruhi oleh seberapa baik metode pertama dalam pendekatan *hybrid* dalam menghasilkan list rekomendasi kasarnya.

Saran yang dapat dipertimbangkan untuk penelitian selanjutnya adalah mecoba metode rekomendasi lain dan atau mencoba menggunakan pendekatan *hybrid* dengan tipe lain dalam menghasilkan list rekomendasi *item* film sebagai pembanding.

## Referensi

- [1] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261–273, 2015, doi: 10.1016/j.eij.2015.06.005.
- [2] G. Geetha, M. Safa, C. Fancy, and D. Saranya, "A Hybrid Approach using Collaborative filtering and Content based Filtering for Recommender System," *Journal of Physics: Conference Series*, vol. 1000, no. 1, 2018, doi: 10.1088/1742-6596/1000/1/012101.
- [3] P. Nagarnaik and A. Thomas, "Survey on recommendation system methods," *2nd International Conference on Electronics and Communication Systems, ICECS 2015*, vol. 137, no. 7, pp. 1603–1608, 2015, doi: 10.1109/ECS.2015.7124857.
- [4] D. Das, L. Sahoo, and S. Datta, "A Survey on Recommendation System," *International Journal of Computer Applications*, vol. 160, no. 7, pp. 6–10, 2017, doi: 10.5120/ijca2017913081.
- [5] C. S. M. Wu, D. Garg, and U. Bhandary, "Movie Recommendation System Using Collaborative Filtering," *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences, ICSESS*, vol. 2018-Novem, pp. 11–15, 2019, doi: 10.1109/ICSESS.2018.8663822.
- [6] K. Y. Jung, D. H. Park, and J. H. Lee, "Hybrid collaborative filtering and content-based filtering for improved recommender system," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3036, pp. 295–302, 2004, doi: 10.1007/978-3-540-24685-5\_37.
- [7] S. Reddy, S. Nalluri, S. Kuniseti, S. Ashok, and B. Venkatesh, *Content-based movie recommendation system using genre correlation*, vol. 105, no. September. Springer Singapore, 2019. doi: 10.1007/978-981-13-1927-3\_42.
- [8] J. Jeon, "Data Science Series: Content-based Recommender System using Azure Databricks," *Visualbi*, 2018. <https://visualbi.com/blogs/microsoft/azure/data-science-series-content-based-recommender-system-using-azure-databricks/>
- [9] K. Luk, "Introduction to TWO approaches of Content-based Recommendation System," *Towards Data Science Web page*, 2019. <https://towardsdatascience.com/introduction-to-two-approaches-of-content-based-recommendation-system-fc797460c18c>
- [10] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems*. Cambridge: Cambridge University Press, 2010. doi: 10.1017/CBO9780511763113.
- [11] E. Çano and M. Morisio, "Hybrid recommender systems: A systematic literature review," *Intelligent Data Analysis*, vol. 21, no. 6. IOS Press, pp. 1487–1524, 2017. doi: 10.3233/IDA-163209.
- [12] B. Wang, "Ranking Evaluation Metrics for Recommender Systems," *Towards Data Science*, 2021. <https://towardsdatascience.com/ranking-evaluation-metrics-for-recommender-systems-263d0a66ef54>
- [13] R. J. Tan, "Breaking Down Mean Average Precision (mAP)," *Towards Data Science*, 2019. <https://towardsdatascience.com/breaking-down-mean-average-precision-map-ae462f623a52#1a59>
- [14] S. Sawtelle, "Mean Average Precision (MAP) For Recommender Systems," 2016. [sdsawtelle.github.io/blog/output/mean-average-precision-MAP-for-recommender-systems.html](https://sdsawtelle.github.io/blog/output/mean-average-precision-MAP-for-recommender-systems.html)
- [15] G. Shani and A. Gunawardana, "Evaluating Recommendation Systems," *Recommender Systems Handbook*, pp. 257–297, 2011, doi: 10.1007/978-0-387-85820-3\_8.
- [16] P. Refaeilzadeh, L. Tang, and H. Liu, "Cross-Validation," in *Encyclopedia of Database Systems*, Springer New York, 2016, pp. 1–7. doi: 10.1007/978-1-4899-7993-3\_565-2.