



Dampak pembobotan pada metode *hybrid user-based* dan *item-based* untuk sistem rekomendasi film

Fadetul Fitriyeh¹, Nuskhatul Haqqi², Layla Mufah Choiriyah³, Noor Ifada^{*4}

Email: ¹fadetultitriyeh.id@gmail.com, ²nuskhatulhaqqi27@gmail.com, ³layla.mfh25@gmail.com,

⁴noor.ifada@trunojoyo.ac.id

¹Program Studi Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo Madura

²Program Studi Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo Madura

³Program Studi Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo Madura

⁴Program Studi Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo Madura

Diterima: 09 Agustus 2024 | Direvisi: - | Disetujui: 10 November 2024

©2020 Program Studi Teknik Informatika Fakultas Ilmu Komputer,
Universitas Muhammadiyah Riau, Indonesia

Abstrak

Pesatnya perkembangan teknologi membuat informasi semakin banyak dan beragam, termasuk dalam industri film. Jumlah film setiap tahunnya bisa mencapai puluhan ribu film dengan berbagai genre, menyebabkan calon penonton kesulitan untuk memilih film yang sesuai dengan minatnya. Salah satu solusi dari permasalahan tersebut adalah dengan adanya sistem rekomendasi yang dapat memberikan rekomendasi film berdasarkan informasi film yang pernah ditonton sebelumnya. Collaborative Filtering adalah pendekatan yang banyak digunakan dalam sistem rekomendasi. Collaborative Filtering memberikan rekomendasi berdasarkan kesamaan antar user untuk metode User-based dan kesamaan antar item untuk metode Item-based. Namun untuk data dengan ketersebaran yang tinggi, nilai kemiripannya bisa tinggi walaupun hanya ada satu kesamaan item yang di rating. Metode Hybrid bisa menjadi solusi untuk mengatasi hal tersebut, yaitu dengan menggabungkan metode User-based dan Item-based dan menambahkan informasi genre dari item. Hasil prediksi akhir diperoleh dari hasil prediksi seluruh metode yang digabungkan menggunakan kombinasi linear. Penggabungan dilakukan dengan memberikan bobot pada masing-masing metode kemudian dijumlahkan. Tujuan dari penelitian ini adalah untuk mengetahui dampak variasi pembobotan pada metode Hybrid User-based dan Item-based Collaborative Filtering untuk sistem rekomendasi film. Hasil yang diperoleh dari penelitian ini menunjukkan bahwa pembobotan Lebih Dominan User-based (LDUB) dan Sangat Dominan User-based (SDUB) lebih unggul dari pada pembobotan lainnya karena menunjukkan kinerja yang baik untuk daftar rekomendasi yang lebih sedikit.

Kata kunci: *sistem rekomendasi film, user-based, item-based, pembobotan, hybrid*

Weighting impact on hybrid user-based and item-based method for movie recommendation system

Abstract

The rapid development of technology means that information is becoming more abundant and diverse, including in the movie industry. The number of movies each year can reach tens of thousands with various genres, making it difficult for potential viewers to choose a movie that suits their interests. One solution to the problem is the existence of a recommendation system that can provide movie recommendations based on information on movies that have been watched previously. Collaborative Filtering is a widely used approach in recommendation systems. Collaborative Filtering offers recommendations based on the similarity between users for User-based methods and the similarity between items for Item-based methods. However, the similarity value can be high for data with high dispersion even though only one item is in common. The Hybrid method can be a solution to overcome this by combining User-based and Item-based methods and adding genre information from the items. The final prediction result is obtained from the prediction results of all methods combined using linear combination. The combination is done by giving weight to each method and then summarising it. This study aims to determine the impact of weighting variations

on Hybrid User-based and Item-based Collaborative Filtering methods. The results obtained from this study show that More Dominant User-based and Very Dominant User-based weightings are superior to other weightings because they show good performance for a smaller list of recommendations.

Keywords: movie recommendation system, user-based, item-based, weighting, hybrid

1. PENDAHULUAN

Peningkatan penggunaan internet telah memberikan dampak pada jumlah informasi yang meningkat secara eksponensial. Akibatnya, pencarian informasi yang penting semakin sulit dan memerlukan banyak waktu [1]. Hal ini terjadi hampir pada semua aspek kehidupan, salah satunya adalah industri film. Semakin banyak film yang dirilis setiap tahunnya, membuat para penggemar film kesulitan dalam menentukan film mana yang ingin mereka tonton. Oleh karena itu, dibutuhkan sistem rekomendasi yang bisa memberikan rekomendasi film yang akan disukai oleh *user* berdasarkan minat dan preferensinya [1].

Sistem rekomendasi film menggunakan suatu algoritma untuk bisa memprediksi film yang mungkin akan disukai oleh *user* berdasarkan aktivitas dari *user* sebelumnya. Agar dapat menghasilkan prediksi film, sistem rekomendasi membutuhkan data dari film tersebut. Beberapa data yang umum digunakan adalah data rating dan genre. Data tersebut akan diproses berdasarkan ketentuan metode yang digunakan agar bisa menghasilkan daftar rekomendasi film [2].

Algoritma yang dapat digunakan untuk menghasilkan rekomendasi dibagi menjadi tiga kategori, yaitu *Collaborative Filtering* (CF), *Content Based* (CB), dan *Hybrid*. CF adalah pendekatan yang paling umum digunakan, dimana rekomendasi dibuat berdasarkan pemahaman pola rating dari *user* atau *item* dalam menghasilkan daftar rekomendasi [3]. Namun, CF memiliki beberapa kekurangan, diantaranya yaitu, tidak dapat memberikan rekomendasi yang baik jika data rating tidak cukup dan mengalami masalah *cold-start* pada *user* atau *item* baru [4]. CB menggunakan informasi dari profil *user* atau profil *item* untuk memberikan rekomendasi. Profil *user target* (*user* yang ingin diberikan rekomendasi) atau *item target* akan dianalisis dan *item* yang cocok dengan profil *user* atau *item* akan dipilih untuk direkomendasikan [5, 6]. Sehingga metode CB dapat mengatasi permasalahan *cold-start* karena tidak memerlukan informasi dari tetangga *user* atau *item*. Metode *Hybrid* menggabungkan dua atau lebih metode yang berbeda dengan tujuan mengambil kelebihan dari masing-masing metode. Sistem rekomendasi yang menggunakan metode *Hybrid* memiliki keunggulan kinerja karena mengkombinasikan metode dan meningkatkan akurasi dengan menghindari keterbatasan metode tertentu [7].

Pada penelitian yang dilakukan oleh Pirasteh, et al. [8] diusulkan metode *Hybrid* yang menggabungkan metode *User-based* dan *Item-based Collaborative Filtering*. Selain itu, ditambahkan juga penggunaan informasi genre film yang mengasumsikan bahwa film dengan genre serupa memiliki kesamaan fitur dan lebih mirip satu sama lain. Pendekatan tersebut bertujuan untuk mengatasi masalah *cold-start* dan meningkatkan kualitas dari sistem rekomendasi. Hasil penelitian yang telah dilakukan menunjukkan bahwa dengan memasukkan informasi genre film dalam perhitungan rekomendasi, membuat kinerja model menjadi lebih baik. Dengan kata lain, sistem rekomendasi yang mempertimbangkan genre film dapat memberikan rekomendasi yang lebih sesuai dengan preferensi *user*. Namun, meskipun hasil performanya bagus, setiap metodenya digabungkan menggunakan pembobotan yang setara. Sedangkan penelitian lain [9] menunjukkan bahwa performa metode *User-based* dan *Item-based* berbeda dalam perhitungan individu. Dengan demikian, penggabungan metode *User-based* dan *Item-based* seharusnya menggunakan pembobotan yang tidak setara.

Penelitian ini bertujuan untuk meningkatkan nilai akurasi dari hasil sistem rekomendasi dengan menggunakan metode *Hybrid User-based* dan *Item-based Collaborative Filtering*. Informasi *genre* juga digunakan dalam pembentukan matriks sebagai data masukan dan menggunakan pembobotan yang tidak setara dalam menghasilkan prediksi akhir.

2. METODE PENELITIAN

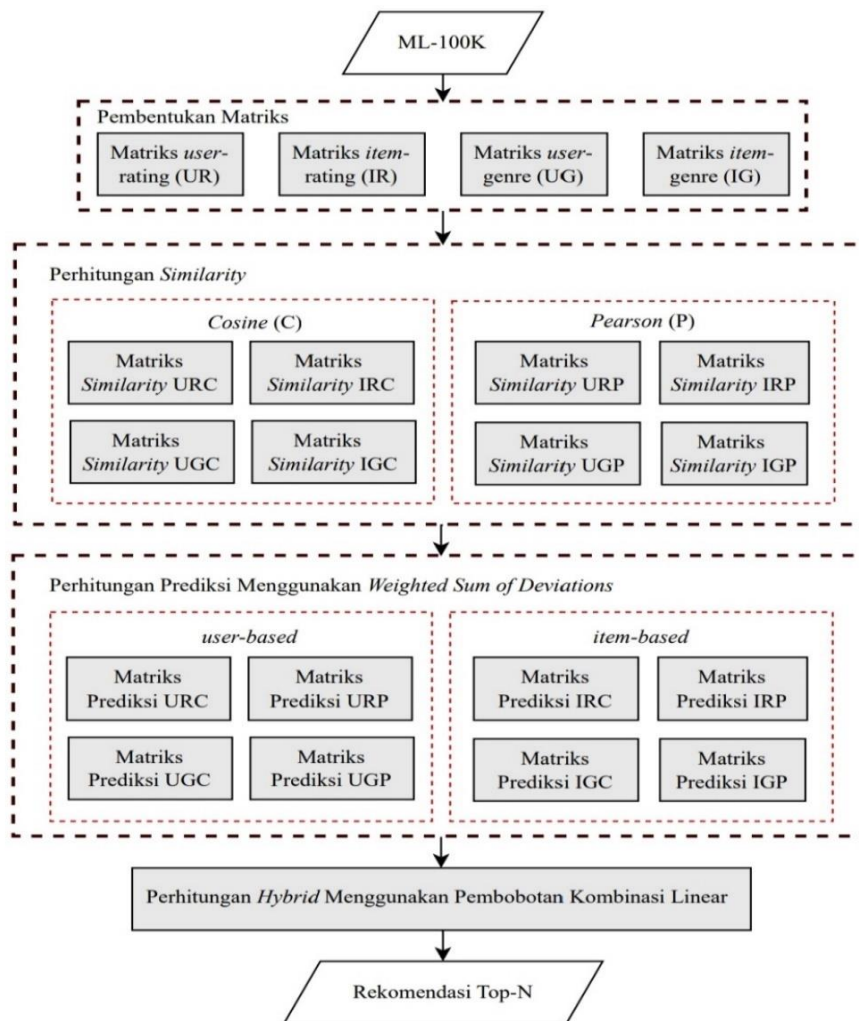
Pada penelitian ini terdapat beberapa tahapan dan proses yang dilakukan, yaitu menyiapkan data, rancangan sistem, dan evaluasi rekomendasi *Top-N*.

2.1. Dataset

Dataset yang digunakan adalah *dataset* Movielens 100k [10] yang memiliki 100.000 rating dari 1682 film, yang dikategorikan dalam 19 *genre*, dan diberi *rating* oleh 943 *user*. Informasi dari *dataset* yang akan digunakan, yaitu *id_user*, *id_item*, *rating*, dan *genre*.

2.2. Rancangan Sistem

Pada penelitian ini terdapat empat tahapan yang dilakukan secara terurut agar dapat menghasilkan rekomendasi film. Gambar 1 menunjukkan diagram alur penelitian yang meliputi pembentukan empat matriks sebagai *input*, Perhitungan *similarity* untuk semua matriks menggunakan *Cosine* dan *Pearson*, dilanjutkan dengan perhitungan prediksi rating menggunakan *Weighted Sum of Deviation*. Perbandingan dilakukan pada variasi nilai pembobotan metode *Hybrid User-based* dan *Item-based* yang digabungkan menggunakan Kombinasi Linear.



Gambar 1. Diagram alur penelitian yang meliputi pembentukan matriks, perhitungan similarity, perhitungan prediksi rating, dan perhitungan hybrid untuk menghasilkan prediksi akhir

2.2.1. Pembentukan Matriks

Berdasarkan diagram alur penelitian, tahap awal dari penelitian ini yaitu pembentukan matriks. Akan dibentuk sebanyak empat matriks sebagai data masukan yaitu matriks *user-rating*, *rating-user*, *user-genre*, dan *item-genre* yang diilustrasikan dalam Gambar 2. Matriks tersebut diperoleh dari file *u.data*, file *u.item*, dan file *u.genre* pada *dataset*. Proses pembentukan matriks adalah sebagai berikut:

- 1) Matriks *user-rating* : matriks yang merepresentasikan baris sebagai *user* dan kolom sebagai *item* yang nilainya didapat dari rating tiap *user* terhadap *item*.
- 2) Matriks *item-rating* : matriks yang diperoleh dengan melakukan *transpose* terhadap matriks *user-rating*.
- 3) Matriks *user-genre*: matriks yang merepresentasikan baris sebagai *user* dan kolom sebagai genre yang nilainya didapat dari informasi genre dari *item* yang telah di rating oleh *user*. Bernilai 0 jika *user* belum merating *item* dalam hal ini film dengan genre tersebut. Bernilai (1-n) jika *user* telah merating film dengan genre tersebut. Misal *user* u1 telah merating film avatar yang bergenre *action* dan *adventure*, maka genre *action* dan *adventure* akan bernilai 1 untuk *user* u1. Nilai n akan terus meningkat sesuai dengan jumlah film bergenre sama yang di rating oleh *user*.
- 4) Matriks *item-genre*: yang merepresentasikan baris sebagai *item* dan kolom sebagai genre yang nilainya didapat dari informasi genre tiap *item*. Nilai genre adalah 0 jika *item* tidak termasuk dalam kategori genre tersebut dan akan bernilai 1 jika sebaliknya.

		item			
		i1	i2	i3	i4
user	u1	1	?	?	3
	u2	?	2	?	?
	u3	?	?	5	3
	u4	5	2	?	?
	u5	?	2	4	2

(a)

		genre					
		g1	g2	g3	g4	g5	g6
user	u1	2	1	1	0	0	2
	u2	1	0	0	1	1	1
	u3	1	1	1	0	1	1
	u4	2	0	1	1	1	2
	u5	2	1	1	1	2	2

(c)

		user				
		u1	u2	u3	u4	u5
item	i1	1	?	?	5	?
	i2	?	2	?	2	2
	i3	?	?	5	?	4
	i4	3	?	3	?	2

(b)

		genre					
		g1	g2	g3	g4	g5	g6
item	i1	1	0	1	0	0	1
	i2	1	0	0	1	1	1
	i3	0	0	1	0	1	0
	i4	1	1	0	0	0	1

(d)

Gambar 2. Matriks yang akan digunakan sebagai data masukan, yang meliputi a) matriks user-rating, b) matriks item-rating, c) matriks user-genre, dan d) matriks item-genre

2.2.2. Perhitungan Similarity

Metode Collaborative Filtering memerlukan perhitungan similarity pada masing-masing matriks yang akan dihitung menggunakan dua metode, yaitu Cosine dan Pearson. Dari perhitungan tersebut akan dihasilkan similarity dengan user atau item tetangga.

a) Perhitungan Cosine Similarity

Perhitungan similarity dengan menggunakan Cosine bertujuan untuk mengukur sejauh mana kesamaan antara dua vektor (film) [11]. Similarity antar user pada metode User-based menggunakan fungsi Cosine dihitung menggunakan Persamaan (1).

$$Cosine_{(u,v)} = \frac{\sum_{k \in I_u \cap I_v} r_{uk} \cdot r_{vk}}{\sqrt{\sum_{k \in I_u \cap I_v} r_{uk}^2} \cdot \sqrt{\sum_{k \in I_u \cap I_v} r_{vk}^2}} \quad (1)$$

Dimana r_{uk} adalah rating yang diberikan oleh user u terhadap item k dan r_{vk} adalah rating yang diberikan oleh user v terhadap item k . I_u dan I_v adalah himpunan item yang telah di rating oleh user u dan v . Perhitungan similarity Cosine untuk mencari kemiripan antar item pada metode Item-based dihitung menggunakan Persamaan (2).

$$Cosine_{(i,j)} = \frac{\sum_{l \in U_i \cap U_j} r_{il} \cdot r_{jl}}{\sqrt{\sum_{l \in U_i \cap U_j} r_{il}^2} \cdot \sqrt{\sum_{l \in U_i \cap U_j} r_{jl}^2}} \quad (2)$$

Dimana r_{il} adalah rating yang diberikan oleh user l terhadap item i dan r_{jl} adalah rating yang diberikan oleh user l terhadap item j . U_i dan U_j adalah himpunan user yang telah merating item i dan j .

b) Perhitungan Pearson Similarity

Perhitungan similarity menggunakan Pearson menghasilkan nilai antara -1 sampai 1. Tujuan dari perhitungan similarity dengan Pearson adalah untuk menghitung kemiripan antar objek berdasarkan perhitungan korelasi linear antar himpunan yang berbeda [12].

$$Pearson(u, v) = \frac{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u) \cdot (r_{vk} - \mu_v)}{\sqrt{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u)^2} \cdot \sqrt{\sum_{k \in I_u \cap I_v} (r_{vk} - \mu_v)^2}} \quad (3)$$

Pearson untuk metode User-based dihitung dengan menggunakan Persamaan (3). Dimana u adalah user target dan v adalah tetangga dari user u . μ_u adalah nilai rata-rata dari user u dan μ_v adalah nilai rata-rata dari user v . similarity Pearson metode Item-based dihitung menggunakan Persamaan (4).

$$Pearson(i, j) = \frac{\sum_{l \in U_i \cap U_j} (r_{il} - \mu_i) \cdot (r_{jl} - \mu_j)}{\sqrt{\sum_{l \in U_i \cap U_j} (r_{il} - \mu_i)^2} \cdot \sqrt{\sum_{l \in U_i \cap U_j} (r_{jl} - \mu_j)^2}} \quad (4)$$

Dimana i adalah item target dan j adalah tetangga dari item i . μ_i adalah nilai rata-rata dari item i dan μ_j adalah nilai rata-rata dari item j .

2.2.3. Perhitungan Prediksi Menggunakan *Weighted Sum of Deviations*

Setelah didapatkan hasil *similarity* dengan menggunakan *Pearson* dan *Cosine*, maka selanjutnya menghitung nilai prediksi rating menggunakan *Weighted Sum of Deviations*. Persamaan (5) adalah perhitungan prediksi *User-based* menggunakan *Weighted Sum of Deviations*.

$$P_{ai} = \bar{r}_a + \frac{\sum_{\{u \in N_a, i \in S_u\}} \text{sim}(a,u) * (r_{ui} - \bar{r}_u)}{\sum_{\{u \in N_a, i \in S_u\}} |\text{sim}(a,u)|} \tag{5}$$

Dimana \bar{r}_u merupakan representasi dari rata-rata nilai rating *user* u . N_a adalah himpunan tetangga *user* a , jumlah tetangga terdekat *user* $K \leq N_a$. S_u adalah himpunan *item* yang telah di rating *user* u . $\text{sim}(a, u)$ adalah nilai *similarity* *user* a terhadap *user* u . Sedangkan untuk *Item-based* dihitung menggunakan Persamaan (6).

$$P_{ia} = \bar{r}_i + \frac{\sum_{\{j \in S_a \cap N_i\}} \text{sim}(i,j) * (r_{aj} - \bar{r}_j)}{\sum_{\{j \in S_a \cap N_i\}} |\text{sim}(i,j)|} \tag{6}$$

Dimana \bar{r}_i merupakan representasi dari rata-rata rating *item* i , $\text{sim}(i, j)$ adalah nilai *similarity* *item* i terhadap *item* j , dan jumlah tetangga terdekat *item* $K \leq N_i$.

2.3.1. Perhitungan Hybrid Menggunakan Pembobotan Kombinasi Linear

Prediksi akhir dihitung dengan menggabungkan hasil delapan matriks prediksi menggunakan Kombinasi Linear yang ditentukan pada Persamaan (7).

$$\hat{r}_{ui} = \omega_1 \times r_{ui}^{(URC)} + \omega_2 \times r_{ui}^{(URP)} + \omega_3 \times r_{ui}^{(IRC)} + \omega_4 \times r_{ui}^{(IRP)} + \omega_5 \times r_{ui}^{(UGC)} + \omega_6 \times r_{ui}^{(UGP)} + \omega_7 \times r_{ui}^{(IGC)} + \omega_8 \times r_{ui}^{(IGP)} \tag{7}$$

Dimana $r_{ui}^{(URC)}$ adalah prediksi rating yang diperoleh berdasarkan matriks *user-rating* dan menggunakan *similarity Cosine*, $r_{ui}^{(URP)}$ adalah prediksi rating yang diperoleh berdasarkan matriks *user-rating* dan menggunakan *similarity Pearson*, $r_{ui}^{(IRC)}$ adalah prediksi rating yang diperoleh berdasarkan matriks *item-rating* dan menggunakan *similarity Cosine*, $r_{ui}^{(IRP)}$ adalah prediksi rating yang diperoleh berdasarkan matriks *item-rating* dan menggunakan *similarity Pearson*, $r_{ui}^{(UGC)}$ adalah prediksi rating yang diperoleh berdasarkan matriks *user-genre* dan menggunakan *similarity Cosine*, $r_{ui}^{(UGP)}$ adalah prediksi rating yang diperoleh berdasarkan matriks *user-genre* dan menggunakan *similarity Pearson*, $r_{ui}^{(IGC)}$ adalah prediksi rating yang diperoleh berdasarkan matriks *item-genre* dan menggunakan *similarity Cosine*, dan $r_{ui}^{(IGP)}$ adalah prediksi rating yang diperoleh berdasarkan matriks *item-genre* dan menggunakan *similarity Pearson*.

Pada penelitian [9] menunjukkan bahwa performa model yang menggunakan metode *User-based* dan *Item-based* itu berbeda. Sehingga perlu dilakukan pembobotan yang tidak setara antara matriks prediksi yang menggunakan metode *User-based* dan *Item-based*, agar dapat diketahui tingkat kepentingan setiap metode dan menghasilkan akurasi yang lebih baik. Setiap matriks hasil prediksi akan diberikan nilai bobot (ω) sebelum digabungkan. ω adalah parameter yang mengontrol penggabungan hasil prediksi rating. Penelitian ini dilakukan percobaan dengan nilai ω berbeda yang diestimasi dalam Tabel 1.

Tabel 1. Variasi Nilai ω

Matriks	$r_{ui}^{(URC)}$	$r_{ui}^{(URP)}$	$r_{ui}^{(IRC)}$	$r_{ui}^{(IRP)}$	$r_{ui}^{(UGC)}$	$r_{ui}^{(UGP)}$	$r_{ui}^{(IGC)}$	$r_{ui}^{(IGP)}$
Variasi Nilai ω	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8
Setara [8]	0,125	0,125	0,125	0,125	0,125	0,125	0,125	0,125
Lebih dominan <i>User-based</i> (LDUB)	0,15	0,15	0,1	0,1	0,15	0,15	0,1	0,1
Sangat dominan <i>User-based</i> (SDUB)	0,2	0,2	0,05	0,05	0,2	0,2	0,05	0,05
Lebih dominan <i>Item-based</i> (LDIB)	0,1	0,1	0,15	0,15	0,1	0,1	0,15	0,15
Sangat dominan <i>Item-based</i> (SDIB)	0,05	0,05	0,2	0,2	0,05	0,05	0,2	0,2

2.2.5. Rekomendasi *Top-N*

User target mendapatkan daftar rekomendasi *Top-N item* dari hasil prediksi rating yang diurutkan secara *descending* ditentukan menggunakan Persamaan (8).

$$Top_u(N) = \underset{i \in \hat{I}_u}{argmax} \hat{r}_{ui} \tag{8}$$

Dimana N adalah jumlah *item* yang ingin direkomendasikan, \hat{I}_u adalah himpunan *item* yang belum di rating oleh *user* u dengan syarat $I_u \cap \hat{I}_u = \emptyset$.

2.3. Evaluasi Rekomendasi Top-N

Penelitian ini menggunakan *k-fold cross validation* sebagai metode evaluasi untuk menguji hasil penelitian. Cara kerjanya adalah dengan membagi *dataset* secara acak sebanyak *k* dengan ukuran yang sama, satu *k* digunakan sebagai data *test* dan sisanya sebagai data *training*. Proses ini akan diulang sebanyak *k* kali, dengan syarat data yang digunakan sebagai data *test* pada setiap perulangan berbeda. Kemudian hasil dari proses perulangan akan digabungkan dengan menghitung nilai rata-ratanya sehingga menghasilkan nilai tunggal. Nilai *k* yang akan digunakan adalah 5 dengan artian data *training* sebanyak 80% dan data *test* sebanyak 20%.

Metrik evaluasi yang digunakan untuk menghitung kinerja sistem rekomendasi adalah *Normalized Discounted Cumulative Gain* (NDCG). *Discounted Cumulative Gain* (DCG) digunakan untuk mengukur kualitas peringkat *item* yang direkomendasikan dan mengevaluasi kegunaan atau keuntungan suatu *item* berdasarkan posisinya dalam daftar hasil. Nilai DCG tinggi ketika *item* yang relevan ditempatkan lebih tinggi dalam daftar. NDCG adalah normalisasi DCG dihitung dengan pembagian DCG oleh Ideal DCG (IDCG) [13]. DCG dihitung menggunakan Persamaan (9), IDCG dihitung menggunakan Persamaan (10), NDCG dihitung menggunakan Persamaan (11).

$$DCG_U(N) = \sum_{n=1}^N \frac{1}{\log_2(1+n)} \cdot \mathbb{1}(Top_u(n) \in GT_U) \tag{9}$$

$$IDCG(N) = \sum_{n=1}^N \frac{1}{\log_2(1+n)} \tag{10}$$

$$NDCG_U(N) = \frac{DCG_U(N)}{IDCG(N)} \tag{11}$$

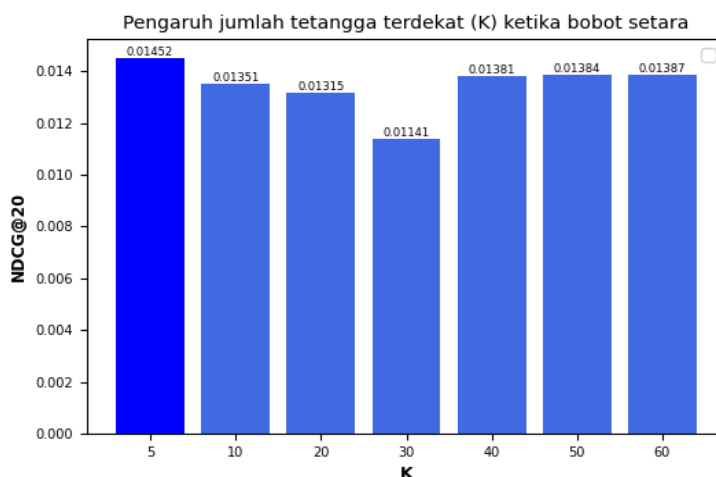
Dimana *N* adalah banyaknya film yang akan direkomendasikan terhadap *user*, $\mathbb{1}(Top_u(n) \in GT_U)$ bernilai 1 jika *item* yang direkomendasikan terdapat dalam data test dan bernilai 0 jika tidak.

3. HASIL DAN PEMBAHASAN

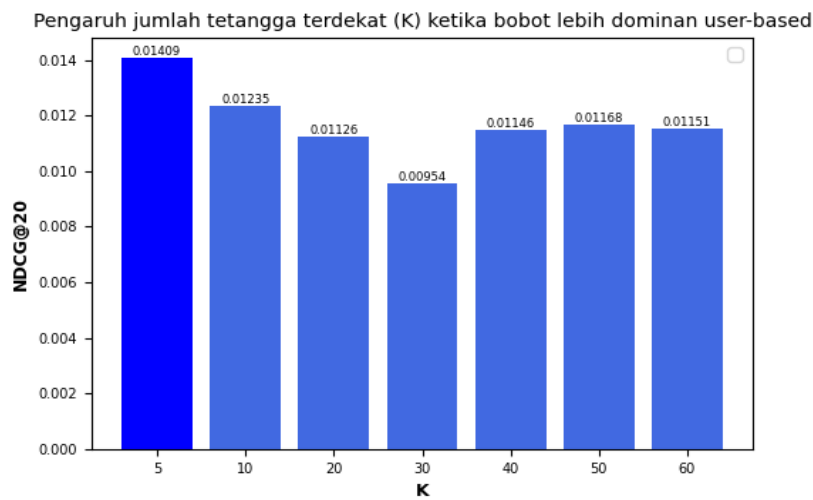
Penelitian ini melakukan eksperimen untuk menemukan jumlah tetangga terdekat terbaik yang digunakan dalam metode *User-based* dan *Item-based*. Setelah nilai *K* pada masing-masing model diketahui selanjutnya adalah mencari nilai variasi pembobotan terbaik pada metode *Hybrid*. Variasi pembobotan yang digunakan yaitu pembobotan Setara [8], Lebih Dominan *User-based* (LDUB), Sangat Dominan *User-based* (SDUB), Lebih Dominan *Item-based* (LDIB), dan Sangat Dominan *Item-based* (SDIB).

3.1. Pengaruh Nilai Tetangga Terdekat untuk Semua Pembobotan

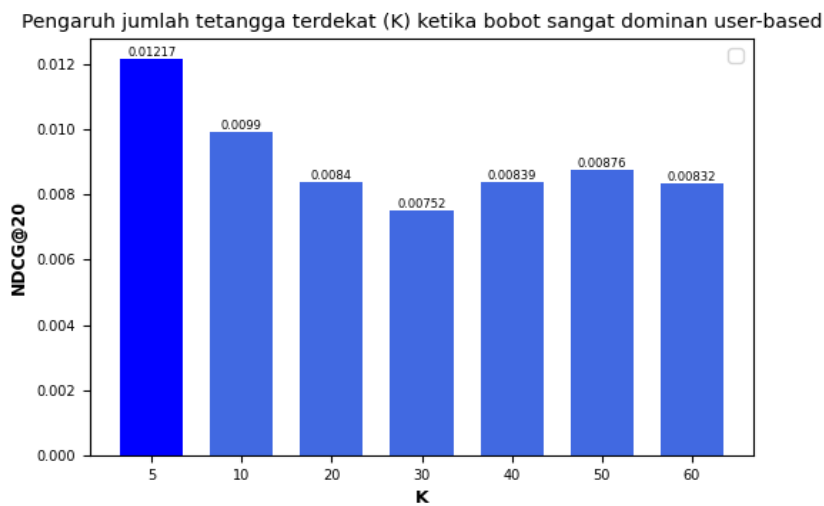
Jumlah tetangga terdekat yang digunakan untuk eksperimen adalah $K = \{5, 10, 20, 30, 40, 50, 60\}$. Nilai *K* diimplementasikan ketika menghitung prediksi pada delapan matriks dari hasil *similarity Pearson* dan *Cosine*. Sehingga pada masing-masing eksperimen nilai *K* menghasilkan 8 matriks prediksi yang kemudian digabungkan menggunakan lima variasi pembobotan. Perbandingan dilakukan pada semua hasil nilai *K* untuk masing-masing pembobotan. Gambar 3 merupakan hasil dari pembobotan Setara, Gambar 4 merupakan hasil ketika pembobotan LDUB, Gambar 5 merupakan hasil ketika pembobotan SDUB, Gambar 6 merupakan hasil ketika pembobotan LDIB, dan Gambar 7 merupakan hasil ketika pembobotan SDIB.



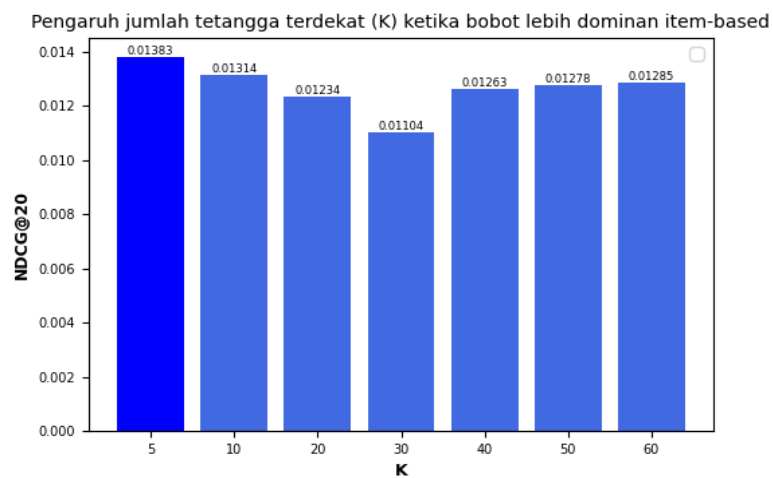
Gambar 3. Diagram perbandingan nilai *K* dengan pembobotan Setara pada NDCG@20. Diagram tersebut menunjukkan bahwa nilai $K = 5$ menjadi nilai tetangga terbaik



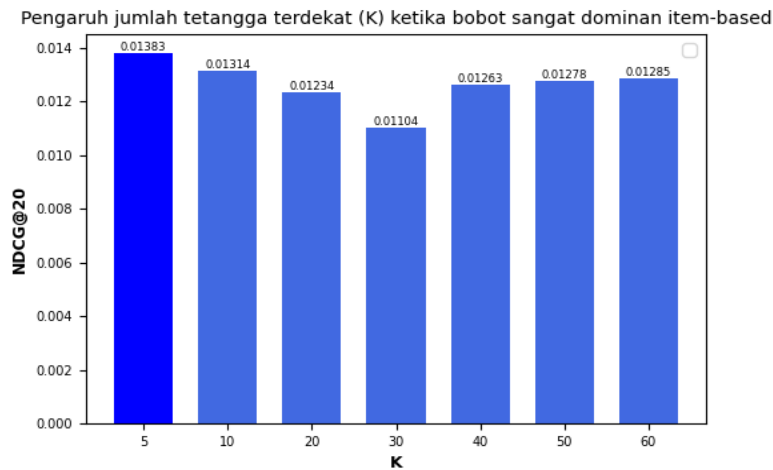
Gambar 4. Diagram perbandingan nilai K dengan pembobotan Lebih Dominan *User-based* (LDUB) pada NDCG@20. Diagram tersebut menunjukkan bahwa nilai $K = 5$ menjadi nilai tetangga terbaik



Gambar 5. Diagram perbandingan nilai K dengan pembobotan Sangat Dominan *User-based* (SDUB) pada NDCG@20. Diagram tersebut menunjukkan bahwa nilai $K = 5$ menjadi nilai tetangga terbaik



Gambar 6. Diagram perbandingan nilai K dengan pembobotan Lebih Dominan *Item-based* (LDIB) pada NDCG@20. Diagram tersebut menunjukkan bahwa nilai $K = 5$ menjadi nilai tetangga terbaik



Gambar 7. Diagram perbandingan nilai K dengan pembobotan Sangat Dominan *Item-based* (SDIB) pada NDCG@20. Diagram tersebut menunjukkan bahwa nilai $K = 5$ menjadi nilai tetangga terbaik

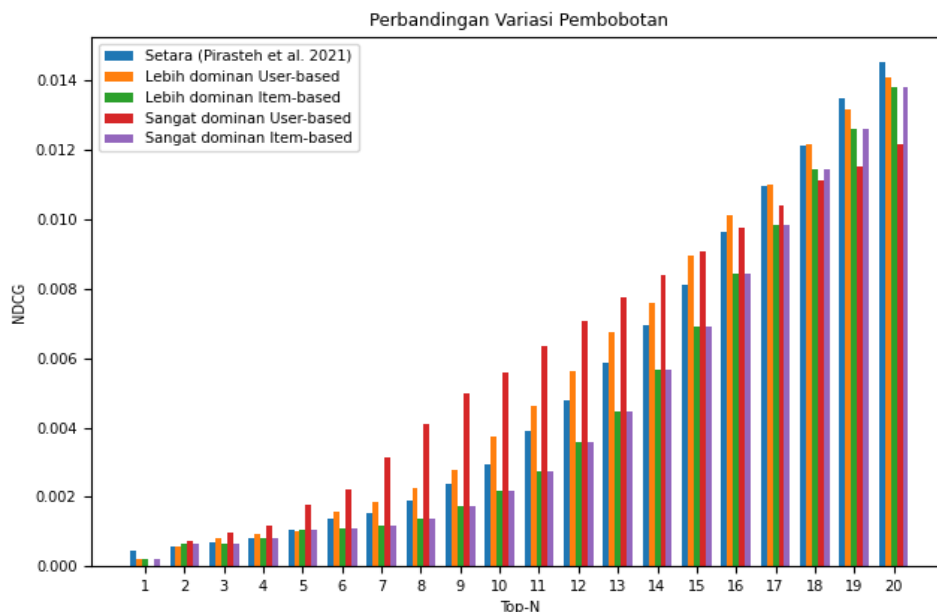
Dari hasil eksperimen yang telah dilakukan diperoleh nilai tetangga terbaik pada NDCG@20 untuk semua pembobotan adalah $K = 5$. Hal ini membuktikan bahwa kemiripan tetangga sangat mempengaruhi hasil prediksi. Semakin kecil nilai K yang digunakan, maka hasil sistem rekomendasi menjadi semakin baik.

3.2. Dampak Pembobotan Pada Metode *Hybrid User-based* dan *Item-based*

Berdasarkan hasil dari sub bab sebelumnya didapatkan tetangga terbaik dari semua pembobotan adalah 5, oleh karena itu setiap model menggunakan nilai $K = 5$. Kemudian perbandingan dilakukan pada nilai $Top-N = \{1,2,3,4, \dots, 20\}$.

Gambar 8 menunjukkan perbandingan hasil rekomendasi dari semua variasi pembobotan dengan nilai $Top-1$ sampai $Top-20$. Dapat diketahui bahwa pembobotan Lebih Dominan *User-based* (LDUB) dan Sangat Dominan *User-based* (SDUB) memiliki kecenderungan untuk lebih unggul dibanding yang lain. Meskipun pembobotan Setara juga mampu menjadi lebih unggul dibanding yang lain, namun hal ini hanya terjadi ketika $N > 18$. Karena tujuan sistem rekomendasi adalah mencari *item* yang dibutuhkan oleh *user* dari banyaknya *item* yang ada, maka *user* sebenarnya memiliki kecenderungan untuk lebih tertarik kepada daftar rekomendasi *item* yang sedikit dibandingkan dengan daftar rekomendasi *item* yang banyak [14].

Sedangkan pembobotan Lebih Dominan *Item-based* (LDIB) dan Sangat Dominan *Item-based* (SDIB) menunjukkan kinerja yang kurang bagus pada semua $Top-N$. Dengan kata lain, hasil penelitian ini membuktikan bahwa pembobotan yang didominasi oleh *Item-based* tidak disarankan untuk diterapkan pada metode *Hybrid*.



Gambar 8. Diagram perbandingan dari nilai ω . Kita dapat melihat bahwa variasi nilai ω sangat dominan User-based unggul mulai dari $Top-2$ sampai dengan $Top-15$. Ketika $Top-N = \{16,17,18\}$, nilai ω Lebih Dominan User-based (LDUB) lebih unggul. Selebihnya ketika nilai ω dengan nilai setara

3.3. Perbandingan Hasil Sistem Rekomendasi Metode Hybrid

Sistem rekomendasi dibangun menggunakan metode *Hybrid* dengan beberapa variasi pembobotan. Hasil rekomendasi diperoleh berdasarkan target *user* dan jumlah *Top-N*. Seperti yang ditunjukkan Gambar 9, rekomendasi yang dihasilkan pada target *user* 100 dan *Top-20* berbeda pada masing-masing pembobotan. Untuk pembobotan Setara, film “As Good As It Gets (1997)” berada pada urutan ketiga belas untuk direkomendasikan, ketika LDUB berada di urutan kedelapan dan urutan ketiga ketika SDUB. Sedangkan ketika LDIB dan SDIB film tersebut tidak direkomendasikan. Begitu pula dengan film “Titanic (1997)” yang berada pada urutan kesembilan belas jika menggunakan pembobotan LDUB dan urutan keempat belas ketika SDUB, sedangkan pada pembobotan lain film “Titanic (1997)” tidak masuk dalam daftar rekomendasi. Sedangkan film “Confidential (1997)” hanya direkomendasikan pada saat SDUB. Dapat disimpulkan bahwa metode SDUB lebih baik dari pembobotan lainnya dalam memberikan rekomendasi.

<p>Metode : <i>Hybrid</i> Pembobotan Setara Target User : 100 Top-N: 20 Nilai NDCG : 0.037</p> <p>Hasil Rekomendasi</p> <ol style="list-style-type: none"> Someone Else's America (1995) Star Kid (1997) Prefontaine (1997) Entertaining Angels: The Dorothy Day Story (1996) Some Mother's Son (1996) Santa with Muscles (1996) Schindler's List (1993) Marlene Dietrich: Shadow and Light (1996) Faust (1994) They Made Me a Criminal (1939) Saint of Fort Washington, The (1993) Pather Panchali (1955) As Good As It Gets (1997) Close Shave, A (1995) Casablanca (1942) Sunset Blvd. (1950) Shawshank Redemption, The (1994) Everest (1998) Nightwatch (1997) Rear Window (1954) <p>(a)</p>	<p>Metode : <i>Hybrid</i> Pembobotan LDUB Target User : 100 Top-N: 20 Nilai NDCG : 0.078</p> <p>Hasil Rekomendasi</p> <ol style="list-style-type: none"> Someone Else's America (1995) Star Kid (1997) Prefontaine (1997) Some Mother's Son (1996) Entertaining Angels: The Dorothy Day Story (1996) Schindler's List (1993) Santa with Muscles (1996) As Good As It Gets (1997) Faust (1994) Shawshank Redemption, The (1994) Casablanca (1942) Marlene Dietrich: Shadow and Light (1996) Pather Panchali (1955) Nightwatch (1997) They Made Me a Criminal (1939) Sunset Blvd. (1950) Close Shave, A (1995) Rear Window (1954) Titanic (1997) Saint of Fort Washington, The (1993) <p>(b)</p>	<p>Metode : <i>Hybrid</i> Pembobotan SDUB Target User : 100 Top-N: 20 Nilai NDCG : 0.14</p> <p>Hasil Rekomendasi</p> <ol style="list-style-type: none"> Someone Else's America (1995) Some Mother's Son (1996) As Good As It Gets (1997) Star Kid (1997) Schindler's List (1993) Shawshank Redemption, The (1994) Prefontaine (1997) Nightwatch (1997) Rear Window (1954) Casablanca (1942) Sound of Music, The (1965) Faust (1994) When We Were Kings (1996) Titanic (1997) Sunset Blvd. (1950) Entertaining Angels: The Dorothy Day Story (1996) Pather Panchali (1955) Star Wars (1977) L.A. Confidential (1997) Boat, Das (1981) <p>(c)</p>
<p>Metode : <i>Hybrid</i> Pembobotan LDIB Target User : 100 Top-N: 20 Nilai NDCG : 0</p> <p>Hasil Rekomendasi</p> <ol style="list-style-type: none"> Someone Else's America (1995) Prefontaine (1997) Saint of Fort Washington, The (1993) Star Kid (1997) Entertaining Angels: The Dorothy Day Story (1996) Marlene Dietrich: Shadow and Light (1996) Santa with Muscles (1996) They Made Me a Criminal (1939) Faust (1994) Everest (1998) Pather Panchali (1955) Schindler's List (1993) Close Shave, A (1995) Perfect Candidate, A (1996) Sunset Blvd. (1950) Some Mother's Son (1996) Casablanca (1942) Affair to Remember, An (1957) Third Man, The (1949) Usual Suspects, The (1995) <p>(d)</p>	<p>Metode : <i>Hybrid</i> Pembobotan SDIB Target User : 100 Top-N: 20 Nilai NDCG : 0</p> <p>Hasil Rekomendasi</p> <ol style="list-style-type: none"> Someone Else's America (1995) Prefontaine (1997) Saint of Fort Washington, The (1993) Star Kid (1997) Entertaining Angels: The Dorothy Day Story (1996) Marlene Dietrich: Shadow and Light (1996) Santa with Muscles (1996) They Made Me a Criminal (1939) Faust (1994) Everest (1998) Pather Panchali (1955) Schindler's List (1993) Close Shave, A (1995) Perfect Candidate, A (1996) Sunset Blvd. (1950) Some Mother's Son (1996) Casablanca (1942) Affair to Remember, An (1957) Third Man, The (1949) Usual Suspects, The (1995) <p>(e)</p>	

Gambar 9. Perbandingan hasil rekomendasi metode *Hybrid* dengan variasi nilai pembobotan. Yang meliputi (a) Pembobotan Setara, (b) Pembobotan Lebih Dominan *User-based* (LDUB), (c) Pembobotan Sangat Dominan *User-based* (SDUB), (d) Pembobotan Lebih Dominan *Item-based* (LDIB), (e) Pembobotan Sangat Dominan *Item-based* (SDIB)

4. KESIMPULAN

Penelitian ini melakukan perbandingan antara nilai pembobotan setara, Lebih Dominan *User-based* (LDUB), Sangat Dominan *User-based* (SDUB), Lebih Dominan *Item-based* (LDIB), dan Sangat Dominan *Item-based* (SDIB). Hasil uji coba menunjukkan bahwa pembobotan LDUB dan SDUB cenderung unggul pada nilai *Top-N* yang kecil. Sehingga dapat disimpulkan bahwa pembobotan LDUB dan SDUB lebih baik daripada pembobotan setara yang unggul ketika jumlah *item* yang direkomendasikan lebih banyak. Hal ini dikarenakan *user* cenderung lebih memilih *item* dari daftar rekomendasi teratas yang lebih sesuai dengan minat *user*. Sedangkan pembobotan LDIB dan SDIB tidak bagus untuk diimplementasikan pada metode *Hybrid* karena memiliki kinerja yang kurang bagus pada semua *Top-N*.

5. DAFTAR PUSTAKA

- [1] Anwar, T. and Uma, V., 2021, Comparative study of recommender system approaches and movie recommendation using collaborative filtering. *International Journal of System Assurance Engineering and Management*, 12(2021), pp. 426-436, doi: <https://doi.org/10.1007/s13198-021-01087-x>.
- [2] Thakker, U., Ruhi Patel, and Shah, M., 2021, A comprehensive analysis on movie recommendation system employing collaborative filtering. *Multimedia Tools and Applications* 80, pp. 28647-28672, doi: <https://doi.org/10.1007/s11042-021-10965-2>.
- [3] Ifada, N., Rahman, T. F., and Sophan, M. K., 2020, Comparing Collaborative Filtering and Hybrid based Approaches for Movie Recommendation. *The 6th Information Technology International Seminar (ITIS)*, Surabaya, pp. 219-223, doi: <https://doi.org/10.1109/ITIS50118.2020.9321014>.
- [4] Dietz, A. E., *Personalizing Talent Development Utilizing A Network Framework*. Thesis. Monterey, CA: Naval Postgraduate School, 2021.
- [5] Barman, S. D., Hasan, M., and Roy, F., 2019, A genre-based item-item collaborative filtering: facing the cold-start problem. *The 2019 8th International Conference on Software and Computer Applications*, Penang, Malaysia, pp. 258-262, doi: <https://doi.org/10.1145/3316615.3316732>.
- [6] Ardiansyah, R., Saputra, B. D., and Bianto, M. A., 2023, Sistem Rekomendasi Buku Perpustakaan Sekolah menggunakan Metode Content-Based Filtering. *Jurnal Computer Science and Information Technology (CoSciTech)*, 4(2), pp. 510-517, doi: <https://doi.org/10.37859/coscitech.v4i2.5131>.
- [7] Kanmani, R. S. A., Surendiran, B., and Ibrahim, S. P. S., 2021, Recency augmented hybrid collaborative movie recommendation system. *International Journal of Information Technology and Management*, 13(5), pp. 1829-1836, doi: <https://doi.org/10.1007/s41870-021-00769-w>.
- [8] Pirasteh, P., Bouguelia, M.-R., and Santosh, K. C., 2021, Personalized recommendation: an enhanced hybrid collaborative filtering. *Advances in Computational Intelligence* 1, p. 1, doi: <https://doi.org/10.1007/s43674-021-00001-z>.
- [9] Nudrat, S., Khan, H. U., Iqbal, S., Talha, M. M., Alarfaj, F. K., and Almusallam, N., 2022, Users' Rating Predictions Using Collaborating Filtering Based on Users and Items Similarity Measures. *Computational Intelligence and Neuroscience*, 2022(1), p. 2347641, doi: <https://doi.org/10.1155/2022/2347641>.
- [10] Harper, F. and Konstan, J., 2015, The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4), pp. 1-19, doi: <https://doi.org/10.1145/2827872>.
- [11] Fajriansyah, M., Adikara, P. P., and Widodo, A. W., 2021, Sistem Rekomendasi Film Menggunakan Content Based Filtering. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 5(6), pp. 2188-2199.
- [12] Lubis, Y. I., Napitupulu, D. J., and Dharma, A. S., 2020, Implementasi Metode Hybrid Filtering (Collaborative dan Content-based) untuk Sistem Rekomendasi Pariwisata. *The 12th Conference on Information Technology and Electrical Engineering*, Yogyakarta, Indonesia, pp. 28-35.
- [13] Tong, Y., Luo, Y., Zhang, Z., Sadiq, S., and Cui, P., 2019, Collaborative generative adversarial network for recommendation systems. *The 2019 IEEE 35th international conference on data engineering workshops (ICDEW)*, Macao, China, pp. 161-168, doi: <https://doi.org/10.1109/ICDEW.2019.00-16>.
- [14] Ifada, N., Putri, N. F. D., and Sophan, M. K., 2020, Normalization based Multi-Criteria Collaborative Filtering Approach for Recommendation System. *Rekayasa*, 13(3), pp. 234-239, doi: <https://doi.org/10.21107/rekayasa.v13i3.8545>.