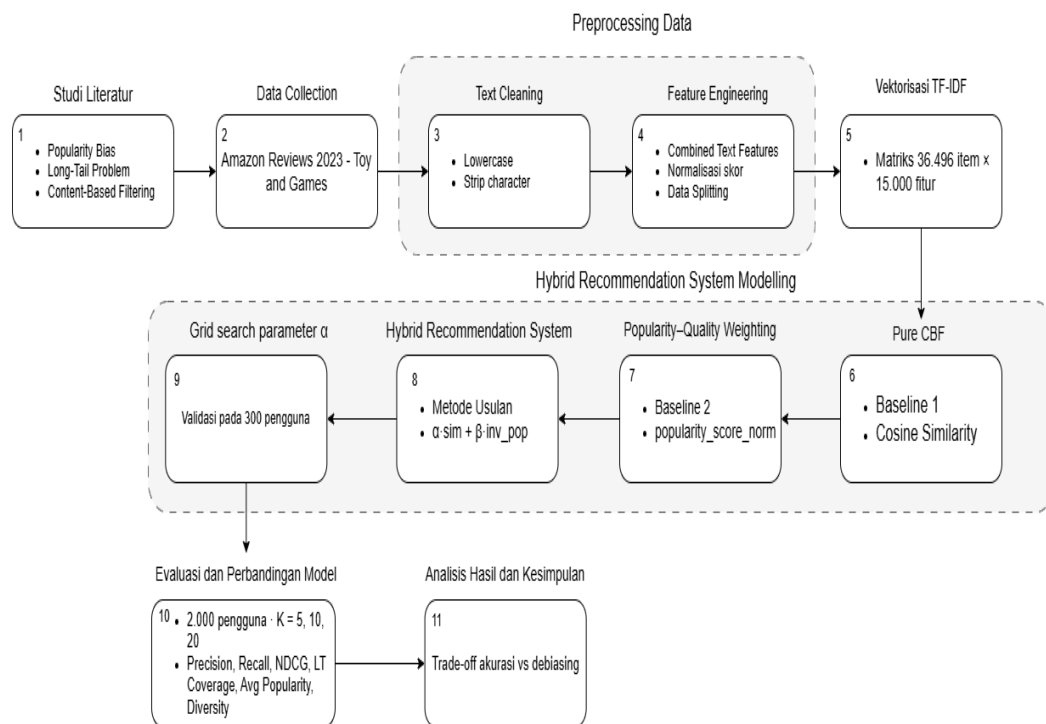


BAB III

METODOLOGI PENELITIAN

3.1 Tahapan Penelitian



Gambar 3. 1 Tahapan Penelitian

Gambar 3.1 menggambarkan keseluruhan dari tahapan penelitian yang dilaksanakan secara berurutan, dimulai dari studi literatur hingga penarikan kesimpulan. Setiap tahapan dirancang untuk menjawab rumusan masalah yang telah ditetapkan secara sistematis dan terukur.

3.1.1 Studi Literatur dan Identifikasi Masalah

Tahap awal penelitian dilakukan melalui studi literatur untuk membangun landasan teoritis yang memadai. Kajian dilakukan terhadap tiga topik utama:

fenomena *information overload* pada platform e-commerce, pendekatan *content-based filtering* dalam sistem rekomendasi, serta permasalahan *popularity bias* dan *long-tail problem*.

Berdasarkan studi literatur yang dilakukan, ditemukan bahwa sebagian besar penelitian mitigasi *popularity bias* berfokus pada pendekatan *collaborative filtering*, sementara integrasi mekanisme pengendalian popularitas pada sistem rekomendasi berbasis konten masih relatif terbatas. Kesenjangan inilah yang menjadi landasan bagi penelitian ini untuk mengusulkan pendekatan *hybrid content-based filtering* dengan mekanisme *Inverse Popularity Weighting*.

3.1.2 Data Collection

Penelitian ini menggunakan dataset *Amazon Reviews 2023* kategori *Toys and Games* yang dikembangkan oleh *McAuley-Lab*, University of California San Diego (Hou *et al.*, 2023). Pemilihan dataset ini didasarkan pada dua pertimbangan utama: kelengkapan atribut konten produk yang mendukung pendekatan *content-based filtering*, serta ketersediaan data interaksi pengguna yang terstruktur untuk keperluan evaluasi menggunakan protokol standar.

Dataset terdiri atas dua komponen utama, yaitu data interaksi pengguna dan data metadata produk. Atribut yang digunakan secara eksplisit dalam pemodelan disajikan pada Tabel 3.1.

Tabel 3. 1 Atribut Dataset yang Digunakan

Atribut	Deskripsi
<i>title</i>	Representasi konten utama produk

<i>categories</i>	Representasi konten kategori produk
<i>features</i>	Representasi konten fitur produk
<i>store</i>	Representasi nama toko
<i>rating_number</i>	Indikator popularitas item
<i>rating</i>	Data preferensi pengguna
<i>user_id</i>	Identitas pengguna
<i>parent_asin</i>	Identitas item

Data interaksi akan dibagi menggunakan protokol *last-out*, di mana item terakhir setiap pengguna berdasarkan urutan *timestamp* ditetapkan sebagai data uji, satu item sebelumnya sebagai data validasi, dan sisanya sebagai data latih.

3.1.3 *Preprocessing Data*

Tahap *preprocessing* dirancang untuk mempersiapkan data mentah menjadi representasi yang siap digunakan oleh model. Terdapat dua prosedur utama yang dirancang pada tahap ini .

1. *Text Cleaning dan Pembangunan Combined Features*

Tahap *text cleaning* dirancang untuk menghasilkan representasi teks yang konsisten sebelum dilakukan proses vektorisasi. Proses ini meliputi konversi seluruh teks menjadi huruf kecil (*lowercase*), penghapusan karakter non-alfanumerik, penguraian atribut berbentuk list, serta penanganan nilai kosong.

Selanjutnya, atribut *title*, *categories*, *features*, dan *store* digabungkan menjadi satu representasi konten produk (*combined_features*). Untuk memberikan bobot representasi yang lebih besar terhadap informasi utama produk, atribut *title* direplikasi sebanyak tiga kali dan *categories* sebanyak dua kali, sedangkan atribut *features* dan *store* digunakan masing-masing satu kali. Representasi *combined_features* dirumuskan sebagai berikut:

$$\text{combined_features} = (\text{title} \times 3) + (\text{categories} \times 2) + \text{features} + \text{store} \quad (\text{Persamaan 3.1})$$

Representasi *combined_features* selanjutnya digunakan sebagai masukan pada proses vektorisasi TF-IDF untuk membentuk representasi numerik setiap produk.

2. Normalisasi Skor Popularitas

Skor popularitas akan diturunkan dari atribut *rating_number* menggunakan dua tahap normalisasi. Pertama, transformasi logaritmik untuk mereduksi efek *skewness* pada distribusi popularitas. Kedua, *min-max scaling* untuk menghasilkan nilai dalam rentang [0,1] yang dapat dikombinasikan langsung dengan komponen lain dalam formula model :

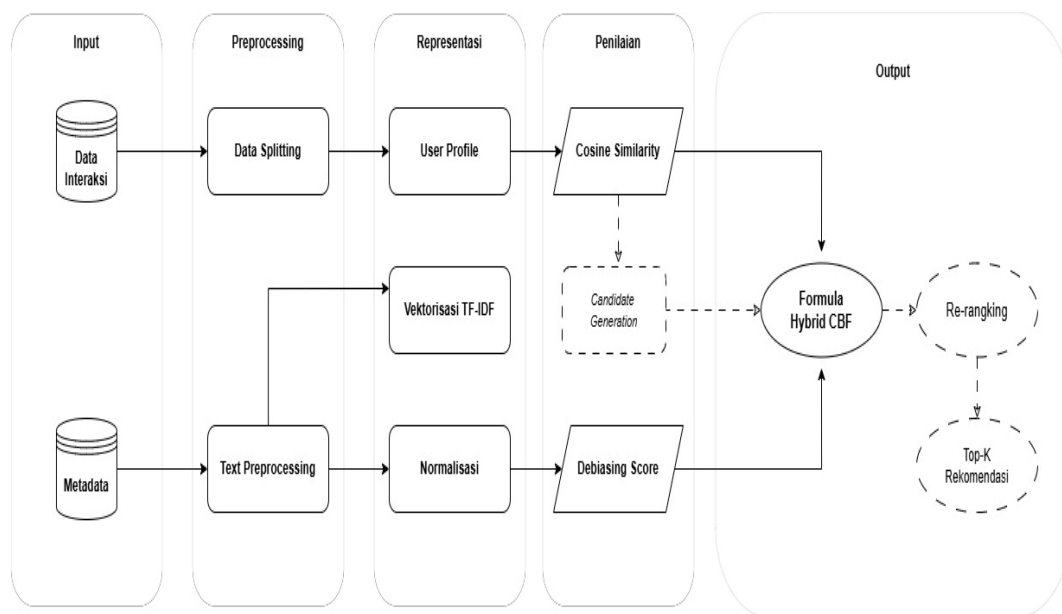
$$\text{popularity_score_norm} = \frac{\log(1 + \text{rating_number}) - \min}{\max - \min} \quad (\text{Persamaan 3.2})$$

Dari skor popularitas tersebut, selanjutnya diturunkan skor kebalikan popularitas sebagai komponen *debiasing*:

$$\text{inv_popularity_norm} = 1 - \text{popularity_score_norm} \quad (\text{Persamaan 3.3})$$

Dengan formula ini, item *long-tail* akan memperoleh nilai *inv_popularity_norm* yang tinggi sehingga mendapat dorongan skor pada tahap *re-scoring* model *hybrid*.

3.1.4 Pemodelan Sistem Rekomendasi



Gambar 3. 2 Arsitektur Sistem Rekomendasi *Hybrid*

Arsitektur sistem rekomendasi yang dirancang dalam penelitian ini ditampilkan pada Gambar 3.2. Sistem dibangun melalui lima lapisan utama, yaitu input, *preprocessing*, representasi, penilaian, dan output.

Untuk keperluan studi komparatif, penelitian ini membangun tiga model sistem rekomendasi yang terdiri atas *Pure Content-Based Filtering*, *Popularity-Based*, dan *Hybrid Content-Based Filtering*.

1. *Pure Content-Based Filtering* (Baseline 1)

Model Pure Content-Based Filtering menghasilkan rekomendasi berdasarkan tingkat kemiripan konten antara profil pengguna dan produk. Tahapan perhitungan terdiri atas pembentukan profil pengguna, perhitungan *Cosine Similarity*, dan pemeringkatan berdasarkan nilai kemiripan.

Profil pengguna dibentuk dari rata-rata berbobot vektor TF-IDF produk yang memperoleh $rating \geq 4,0$ pada data latih.

$$P_U = \frac{\sum_{i \in I_U} r_i v_i}{\sum_{i \in I_U} r_i} \quad (\text{Persamaan 3.4})$$

Penjelasan :

I_U = himpunan produk yang pernah diberi rating oleh pengguna u

r_i = rating produk ke- i

v_i = vektor TF-IDF ke- i

Selanjutnya, tingkat kemiripan antara profil pengguna dan setiap produk dihitung menggunakan *Cosine Similarity*.

$$\text{similarity}(u, i) = \frac{\vec{p}_u \cdot \vec{v}_i}{\|\vec{p}_u\| \cdot \|\vec{v}_i\|} \quad (\text{Persamaan 3.5})$$

Nilai *similarity* kemudian diurutkan secara menurun sehingga diperoleh daftar rekomendasi Top-K. Model ini digunakan sebagai *baseline* utama untuk mengevaluasi kualitas relevansi sekaligus mengidentifikasi tingkat *popularity bias* pada pendekatan *Content-Based Filtering*.

2. *Popularity-Based* (Baseline 2)

Model *Popularity-Based* menghasilkan rekomendasi berdasarkan tingkat popularitas produk tanpa mempertimbangkan preferensi pengguna. Nilai popularitas diperoleh dari atribut *rating_number* yang telah dinormalisasi melalui transformasi logaritmik dan *min-max scaling* pada tahap *preprocessing*. Skor setiap produk dinyatakan sebagai:

$$Score_i = PopularityScore_i \quad (Persamaan 3.6)$$

dengan

$$PopularityScore_i = popularity_score_norm_i \quad (Persamaan 3.7)$$

Seluruh produk kemudian diurutkan berdasarkan nilai *PopularityScore* secara menurun sehingga diperoleh daftar rekomendasi Top-K. Model ini digunakan sebagai baseline yang merepresentasikan kondisi *popularity bias* yang tinggi karena proses pemeringkatan hanya ditentukan oleh tingkat popularitas produk.

3. *Hybrid Recommendation System*

Model yang diusulkan mengombinasikan relevansi konten dengan mekanisme *Inverse Popularity Weighting*. Perhitungan dilakukan dalam dua tahap, yaitu *candidate generation* dan *re-scoring*.

Tahap pertama menghasilkan kandidat rekomendasi berdasarkan nilai *Cosine Similarity* tertinggi.

$$Candidate_i = Similarity(u, i) \quad (Persamaan 3.8)$$

Sebanyak 5.000 kandidat dengan nilai *similarity* tertinggi dipilih untuk diproses pada tahap berikutnya.

Tahap kedua menghitung skor akhir menggunakan kombinasi antara relevansi konten dan komponen *inverse popularity*.

$$FinalScore_i = \alpha \times Similarity(u,i) + \beta \times InvPopularity \quad (Persamaan 3.9)$$

dengan

$$InvPopularity_i = 1 - PopularityScore_i \quad (Persamaan 3.10)$$

serta

$$\alpha + \beta = 1, \quad \alpha, \beta \in [0,1] \quad (Persamaan 3.11)$$

Parameter α merepresentasikan bobot relevansi konten, sedangkan β merepresentasikan bobot komponen *inverse popularity* yang digunakan untuk meningkatkan peluang eksposur produk *long-tail*.

Nilai parameter ditentukan melalui proses *Grid Search* pada data validasi menggunakan metrik *Precision*, *Recall*, *NDCG*, *Long-tail Coverage*, dan *Average Popularity*. Kombinasi parameter yang memenuhi batas minimum *Recall* terhadap *baseline Pure Content-Based Filtering* dipilih sebagai kandidat, kemudian dipilih parameter dengan nilai *Long-tail Coverage* tertinggi. Selanjutnya seluruh kandidat diurutkan berdasarkan nilai *FinalScore* untuk menghasilkan daftar rekomendasi akhir.

3.1.5 Evaluasi Sistem Rekomendasi

Evaluasi dilakukan secara *offline* menggunakan protokol *last-out*, di mana item terakhir setiap pengguna pada data uji dijadikan sebagai *ground truth*. Ketiga model dibandingkan menggunakan nilai $K = 5, 10, \text{ dan } 20$. Sementara Kinerja

ketiga model dievaluasi menggunakan dua kelompok metrik, yaitu *accuracy metrics* dan *beyond-accuracy metrics*, sebagaimana yang disajikan pada Tabel 3.2.

Nilai $K = 5, 10, \text{ dan } 20$ dipilih sebagai *cut-off* evaluasi yang umum digunakan pada penelitian sistem rekomendasi untuk mengukur kinerja model pada panjang daftar rekomendasi yang berbeda. Pembahasan utama difokuskan pada $K=10$ karena memberikan keseimbangan antara jumlah item yang direkomendasikan dan kemudahan interpretasi hasil.

Tabel 3. 2 Rancangan Metrik Evaluasi Penelitian

Kategori	Metrik	Tujuan
<i>Accuracy</i>	<i>Precision@K</i>	Mengukur relevansi rekomendasi
	<i>Recall@K</i>	Mengukur cakupan item relevan
	<i>NDCG@K</i>	Mengukur kualitas <i>ranking</i> rekomendasi
<i>Beyond-Accuracy</i>	<i>Catalog</i>	Mengukur proporsi katalog yang
	<i>Coverage@K</i>	tercakup
	<i>Diversity@K</i>	Mengukur keberagaman daftar rekomendasi
	<i>LT</i>	Mengukur eksposur item <i>long-tail</i>
	<i>Coverage@K</i>	
	<i>Avg</i>	Mengukur tingkat dominasi item populer
	<i>Popularity@K</i>	

Perbandingan dilakukan terhadap tiga metode sebagaimana disajikan pada

Tabel 3.3

Tabel 3. 3 Rancangan Studi Komparatif

Metode	Deskripsi	Tujuan Perbandingan
<i>Pure Content-Based</i>	TF-IDF + <i>cosine similarity</i> tanpa debiasing	<i>Baseline</i> relevansi konten
<i>Pure Popularity-Based</i>	Ranking berdasarkan <i>popularity_score_norm</i>	<i>Baseline</i> dominasi popularitas
<i>Hybrid CBF</i> (Metode yang Diusulkan)	$\alpha \times \text{similarity} + \beta \times \text{inv_popularity_norm}$	Metode Usulan Penelitian

3.1.6 Simulasi Implementasi Sistem Rekomendasi

Sebagai pelengkap dari evaluasi offline, penelitian ini mengimplementasikan model *Pure Content-Based Filtering*, *Popularity-Based*, dan *Hybrid CBF* dalam bentuk simulasi sistem rekomendasi berbasis web. Simulasi ini bertujuan untuk menunjukkan bahwa model yang diusulkan tidak hanya dapat dievaluasi secara kuantitatif, tetapi juga dapat diterapkan pada sebuah aplikasi yang menghasilkan rekomendasi produk secara langsung.

Simulasi ini dikembangkan menggunakan arsitektur *client-server*, dengan *React* sebagai antarmuka pengguna, *FastAPI* sebagai *backend* layanan rekomendasi, dan *PostgreSQL* sebagai basis data metadata produk serta profil pengguna. Model hasil penelitian diekspor dari notebook dalam bentuk artefak model, kemudian digunakan oleh *backend* untuk menghasilkan rekomendasi berdasarkan permintaan pengguna.

Untuk keperluan simulasi digunakan subset sebanyak 4.000 produk yang terdiri atas 2.000 *head items* dan 2.000 *long-tail items*, serta 10 profil pengguna yang dipilih dari data latih. Simulasi ini hanya digunakan untuk mendemonstrasikan implementasi model dan tidak digunakan sebagai dasar evaluasi performa, karena pengukuran kinerja tetap dilakukan menggunakan keseluruhan data pada tahap evaluasi *offline*.

3.1.7 Analisis Hasil dan Kesimpulan

Tahap akhir penelitian adalah analisis komparatif kinerja ketiga metode berdasarkan seluruh metrik evaluasi pada $K = 5, 10$, dan 20 . Analisis difokuskan pada dua pertanyaan utama: seberapa efektif *Hybrid CBF* dalam mengurangi *popularity bias*, dan seberapa besar *trade-off* yang terjadi terhadap metrik akurasi. Kesimpulan akan mencakup penilaian efektivitas, keterbatasan penelitian, serta rekomendasi pengembangan lanjutan.