

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sistem Rekomendasi**

Sistem rekomendasi adalah salah satu elemen yang sering dimanfaatkan dalam perangkat lunak modern. Sistem rekomendasi merupakan alat yang efisien dalam menyaring informasi yang tersebar secara luas, adanya perubahan kebiasaan penggunaan komputer, tren personalisasi, dan munculnya akses internet. Sistem rekomendasi banyak mendapat perhatian akhir-akhir ini, karena mampu menyarankan berbagai hal kepada pengguna secara efektif. Individu atau perusahaan menggunakan sistem rekomendasi untuk memproyeksikan minat pengguna terhadap produk[8]. Perkembangan sistem rekomendasi dimulai dengan pengamatan yang sederhana, yaitu kesadaran bahwa hampir setiap orang tertarik dan mempertimbangkan saran dari orang lain dalam membuat keputusan sehari-hari. Hal ini terlihat dalam berbagai aktivitas seperti memilih film untuk ditonton, buku untuk dibaca, tempat makan yang patut dikunjungi, dan hal-hal lainnya[9]. Tujuan dari mengimplementasikan sistem rekomendasi diantaranya meningkatkan jumlah penjualan, merekomendasikan ke teman, meningkatkan kepuasan pengguna, meningkatkan pendapatan perusahaan, dan lainnya[2]. Sistem rekomendasi yang unggul mampu memberikan rekomendasi dengan tepat. Namun, sistem rekomendasi seringkali menghadapi masalah berupa *scalability*, *sparsity*, dan *cold start*[3].

#### **2.2 Collaborative Filtering**

*Collaborative filtering* adalah salah satu teknik rekomendasi yang menggunakan informasi peringkat dari pengguna lain untuk menghasilkan saran rekomendasi. Pendekatan ini mengasumsikan bahwa preferensi pengguna terhadap produk cenderung tetap seiring waktu, khususnya dengan pengguna yang memiliki preferensi serupa. Adapun penggunaan *collaborative filtering* telah terbukti sangat efektif[10]. Dalam algoritma ini, data preferensi dapat dikelompokkan berdasarkan kesamaan preferensi, sehingga memungkinkan pengambilan keputusan yang lebih baik. Keunggulan *collaborative filtering* mencakup kemampuannya untuk mengatasi konten yang sulit untuk dianalisis.

*Collaborative filtering* beroperasi dengan menghimpun dan menganalisis sejumlah besar data yang mencerminkan aktivitas, preferensi, dan penilaian pengguna terhadap suatu item. Kemudian, berdasarkan kesamaan item yang dipilih oleh pengguna lain, metode ini memproyeksikan rekomendasi untuk item yang berbeda [11]. Terdapat tiga masalah utama dalam *Collaborative Filtering*, yakni *cold start*, *sparsity* dan *scalability* [12]. *Cold start* merupakan

situasi ketika sistem menghadapi tantangan dalam memberikan rekomendasi untuk item atau pengguna baru yang belum memiliki riwayat atau data yang cukup dalam sistem. Kendala ini dapat muncul baik pada sisi item maupun pada sisi pengguna [13]. *Sparsity* menggambarkan situasi di mana suatu dataset belum terisi sepenuhnya atau mengalami kekurangan data. Salah satu penyebabnya adalah ketika suatu produk kurang diminati oleh pengguna atau kurangnya informasi sehingga pengguna belum dapat memberikan penilaian terhadap produk tersebut. Selain itu, *sparsity* dapat disebabkan oleh pengguna yang tidak menyukai produk atau memilih untuk tidak memberikan penilaian terhadap produk tersebut. Ketika data penilaian bersifat renggang, nilai kemiripan antar pengguna menjadi rendah, sehingga berpotensi memengaruhi kualitas rekomendasi[14]. Sedangkan *scalability* adalah keadaan di mana sistem rekomendasi perlu meningkatkan kapasitas komputasinya agar dapat memberikan rekomendasi yang akurat dan tepat waktu, terutama saat berhadapan dengan data dalam skala besar[14]. *Scalability* timbul ketika jumlah data yang perlu dicari kemiripannya cukup besar.

### 2.3 *K-Means Clustering*

Algoritma *clustering* adalah teknik yang digunakan untuk mengelompokkan data ke dalam sejumlah kelompok (klaster), di mana k harus lebih besar atau sama dengan 2. Dalam proses ini, objek-objek dengan tingkat kesamaan yang tinggi akan ditempatkan dalam satu klaster yang sama sekaligus meningkatkan jarak dari objek-objek yang terdapat dalam klaster lain [15]. Umumnya, metode klaster digunakan pada tahap awal untuk menemukan profil kelompok pengguna dengan tingkat kesamaan yang tinggi. Profil kelompok pengguna yang terbentuk dapat berfungsi sebagai preferensi dan memiliki potensi untuk mengurangi waktu yang dibutuhkan dalam proses pembuatan rekomendasi [16]. Di antara berbagai teknik klaster yang tersedia, algoritma *K-Means* adalah yang paling sering digunakan karena menggunakan rumus yang sederhana dan proses iterasinya berlangsung dengan cepat [17]. Perhitungan jarak menggunakan *Euclidean Distance* pada *K-Means* ditunjukkan pada Persamaan (1) berikut

$$D(x_i, c_j) = \sqrt{\sum_{i=1}^N (x_i - c_j)^2} \quad (1)$$

Keterangan :

D (i, j) : Jarak data ke i ke pusat cluster j

$x_i$  : Data ke i pada atribut data ke k

$c_j$  : Titik pusat ke k pada atribut ke k

#### 2.4 *K-Nearest Neighbor (KNN)*

Algoritma *KNN* adalah salah satu algoritma yang menggunakan rumus similarity yang mudah seperti *euclidean distance*, namun memiliki kemampuan untuk menyelesaikan permasalahan klasifikasi. Keunggulan dari *KNN* adalah kemampuannya untuk memberikan rekomendasi dengan cepat dan akurat dengan kualitas yang tinggi. *KNN* umumnya memanfaatkan dua rumus dasar yang dapat digunakan untuk mengukur kemiripan antara data latih dan data uji, yakni *euclidean distance* dan *cosine similarity* [18]. Beberapa peneliti telah mencoba mengatasi masalah *sparsity* data melalui serangkaian eksperimen [15], [16], [17], [19] salah satunya dengan menerapkan algoritma *KNN*. Perhitungan jarak pada algoritma *KNN* ditunjukkan pada persamaan (2)

$$Dist(x_1, x_2) = \sqrt{\sum_{i=1}^n (x_{1i} - x_{2i})^2} \quad (2)$$

Keterangan :

$Dist(x_1, x_2)$  : Jarak  
 $x_{1i}$  : Nilai data pertama  
 $x_{2i}$  : Nilai data kedua

## 2.5 Penelitian Terkait

Penelitian-penelitian terdahulu yang terkait dan mendasari penelitian ini dapat dilihat pada tabel berikut.

**Tabel 2.1 Penelitian Terkait**

No	Peneliti, Tahun	Judul Penelitian	Dataset	Metode	Hasil Penelitian
1	Chenrui Xiong, Youwu Liu, Zijiang Yang, Eric Liu (2020)	Improving Recommendation Performance with Clustering and Missing Value Prediction	Movielens Dataset 100k	K-Means Clustering dan Prediksi Missing Value berdasarkan Similarity	Hasil percobaan menunjukkan bahwa pendekatan yang diusulkan berhasil meningkatkan efisiensi rekomendasi dan juga meningkatkan kemampuan scalability serta kualitas rekomendasi.
2	Keyvan Vahidy Rodpysh, Seyed Javad Mirabedini, Touraj Banirostam (2021)	Resolving Cold Start and Sparse Data Challenge in Recommender Systems using Multi-Level Singular Value Decomposition	Private dataset	User Context Feature Matrix (UCFM), Item Context Feature Matrix (ICFM), Context Similarity Matrix (CSM), Context Feature Singular Value Decomposition (CFSVD)	Hasil dari penelitian ini menunjukkan bahwa metode yang diusulkan meningkatkan keakuratan entitas yang disarankan dalam sistem pemberi rekomendasi.
3	Bilal Hawashin, Shadi Alzubi, Ala Mughaid, Farshad Fotouhi, Ahmad Abusukhon (2020)	An Efficient Cold Start Solution for Recommender Systems Based on Machine Learning and User Interests	Private dataset	J48, Bayesian Classifier Chain, Random Forest, Deep Neural Network	Pada penelitian ini, peneliti mengusulkan pola yang mengaitkan informasi profil pengguna dan minat yang diekstraksi pengguna, dan didapati hasil bahwa pendekatan ini efisien dalam hal waktu pelatihan, waktu klasifikasi, dan akurasi. Secara detail, penggunaan Bayesian Classifier Chain classifier terbukti menjadi waktu pelatihan dan klasifikasi tercepat, sementara semua classifier terbukti efisien dalam hal akurasi.