

BAB 2

LANDASAN TEORI

2.1 Machine Learning

Machine Learning merupakan sebuah bidang studi ilmiah tentang algoritma dan model statistik yang digunakan pada sebuah sistem komputer untuk melakukan tugas/perintah tertentu tanpa diprogram secara eksplisit. Algoritma *Machine Learning* sering digunakan dalam aplikasi sehari-hari, dan keuntungan dari *Machine Learning* adalah algoritma dapat melakukan tugas/perintahnya secara otomatis setelah memproses data yang dipelajari [18].

2.2 Sistem Rekomendasi

Sistem rekomendasi merupakan sistem yang dapat memberikan saran kepada pengguna berdasarkan preferensi dan kebutuhan pengguna. Sistem rekomendasi dapat diterapkan pada berbagai bidang dan dapat meningkatkan kepuasan pengguna terhadap layanan yang disediakan. Sistem rekomendasi membutuhkan model rekomendasi yang tepat agar hasil rekomendasi sesuai dengan preferensi pengguna yang akan mempermudah pengguna mengambil keputusan yang tepat [19]. Sistem rekomendasi memanfaatkan algoritma *filtering* untuk memprediksi preferensi pengguna seperti metode *content based recommendation* dan *collaborative filtering* [20].

2.3 Collaborative Filtering

Collaborative Filtering (CF) adalah teknik yang populer dan banyak digunakan oleh sistem rekomendasi, yang memanfaatkan kemiripan antara pengguna atau *item* untuk menghasilkan rekomendasi. *Collaborative Filtering* bertujuan untuk memprediksi penilaian atau preferensi yang akan diberikan pengguna terhadap *item* yang belum diketahui atau belum dinilai [21, 22].

Terdapat dua pendekatan utama dalam CF, yaitu *user-based* dan *item-based* [23]. *User-based* CF mengidentifikasi pengguna lain yang memiliki pola preferensi serupa dengan pengguna target, kemudian merekomendasikan *item* yang disukai oleh pengguna lain tersebut. Sementara *item-based* CF mengidentifikasi *item-item* serupa dengan *item* yang disukai pengguna target berdasarkan penilaian pengguna

lain, lalu merekomendasikan *item-item* tersebut.

Keunggulan CF terletak pada tidak diperlukannya informasi detail tentang *item* atau pengguna. Namun, kelemahan utamanya adalah masalah *data sparsity* dan *cold-start problem* [24], di mana sistem sulit memberikan rekomendasi akurat untuk pengguna atau *item* baru yang tidak memiliki data historis.

Pada *item-based collaborative filtering*, sekumpulan *item* yang dinilai oleh pengguna digunakan untuk mengidentifikasi *item* yang paling mirip penilaiannya. Lalu, metode *mean square differences* (MSD) digunakan untuk mengukur tingkat kemiripan antara *item a* dan *b* dengan mengevaluasi keakuratan prediksi *item a* sebagai pemberi rekomendasi tunggal untuk *item b*. Metode prediksi Resnick digunakan untuk menghitung prediksi penilaian dari sebuah *item* yang hanya menggunakan satu *item* tetangga:

$$P_{u,a} = \bar{r}_a + (r_{u,a} - \bar{r}_b) \quad (2.1)$$

$r_{u,a}$ merupakan penilaian *item a* oleh pengguna u . r_a dan r_b masing-masing merupakan penilaian rata-rata dari *item a* dan *b*. MSD digunakan untuk mengukur tingkat kemiripan berdasarkan perbedaan antara skor penilaian aktual dari pengguna umum yang telah menilai keduanya. Skor dari penilaian yang diprediksi $P_{u,a}$ dan penilaian aktual $r_{u,a}$ akan dinormalisasi menggunakan *max-min normalization* untuk memastikan bahwa nilai $MSD_{a,b} \in [0, 1]$.

$$iSim_{a,b}^{MSD} = 1 - \left(\frac{\sum_{u=1}^{|U_{a \cap b}|} (P_{u,a} - r_{u,a})^2}{|U_{a \cap b}|} \right) \quad (2.2)$$

Pengukuran kemiripan penilaian menggunakan *Ochiai similarity measure* dapat mempertimbangkan persentase jumlah pengguna yang telah menilai $U_{a \cap b}$ terhadap jumlah pengguna yang telah menilai setiap *item* U_a dan U_b , akan dimasukkan ke dalam MSD.

$$iSim_{a,b}^{Ochiai} = \frac{|U_{a \cap b}|}{\sqrt{|U_a| \times |U_b|}} \quad (2.3)$$

$$iSim_{a,b}^{CF} = iSim_{a,b}^{MSD} \times iSim_{a,b}^{Ochiai} \quad (2.4)$$

2.4 Content-Based Filtering

Content-based filtering adalah algoritma yang digunakan dalam sistem rekomendasi dengan cara merekomendasikan konten yang mirip dengan interaksi pengguna. Algoritma ini menganalisis karakteristik konten yang berinteraksi dengan pengguna seperti kata kunci, topik, atau genre, dan memprediksi preferensi pengguna terhadap konten yang mirip di masa mendatang. Pendekatan ini membantu pengguna menemukan konten serupa di masa mendatang [25].

Kelebihan CBF adalah kemampuannya memberikan rekomendasi untuk pengguna atau *item* baru tanpa memerlukan data historis dari pengguna lain. Namun, kelemahan utamanya adalah *overspecialization*, di mana rekomendasi terbatas hanya pada *item-item* yang sangat mirip dengan preferensi pengguna di masa lalu [26].

Dalam *item-based content similarity*, setiap *item* memiliki kategorinya masing-masing. Ketika terdapat dua *item* yang memiliki kategori yang sama, maka kedua *item* tersebut dianggap berhubungan/mirip satu sama lain. Semua *item* akan direpresentasikan dengan vektor dengan nilai $[0, 1]$

$$\vec{V}_a = (v_{a,1}, v_{a,2}, \dots, v_{a,c}), \text{ dimana}$$
$$v_{a,c} = \begin{cases} 1 & \text{jika item a termasuk kategori C} \\ 0 & \text{jika item a tidak termasuk kategori C} \end{cases} \quad (2.5)$$

Kemudian, *vector-based cosine similarity* digunakan untuk menghitung kemiripan konten berbasis *item* pada pasangan *item*.

$$i \text{ Sim}_{a,b}^{\text{Content}} = \frac{\sum_{u=1}^n \vec{V}_a \times \vec{V}_b}{\sqrt{\sum_{u=1}^n (\vec{V}_a)^2} \times \sqrt{\sum_{u=1}^n (\vec{V}_b)^2}} \quad (2.6)$$

2.5 Hybrid Content-Based dan Collaborative Filtering

Sistem rekomendasi *hybrid* yang menggabungkan *collaborative* dan *content-based filtering* dapat menghasilkan akurasi rekomendasi yang lebih tinggi dibandingkan dengan sistem *collaborative* atau *content-based*. Dengan memanfaatkan pengetahuan umum dari *collaborative filtering* dan potensi dari *content-based filtering* untuk memperluas cakupan rekomendasi, sistem rekomendasi akan mempertimbangkan lebih banyak faktor, seperti preferensi

pengguna dan karakteristik *item*, yang akan meningkatkan kualitas rekomendasi secara keseluruhan [14].

Sistem rekomendasi *hybrid* dapat dilakukan dengan berbagai cara, seperti mengimplementasi *collaborative* dan *content-based* secara individual dan menggabungkan prediksinya, mengintegrasikan karakteristik dari satu metode ke metode lainnya, atau membangun model yang menggabungkan karakteristik keduanya [27].

Beberapa teknik dapat digunakan dalam sistem rekomendasi *hybrid* seperti:

- *Weighted Method*, dimana skor dari metode rekomendasi digabungkan untuk menghasilkan satu rekomendasi
- *Switching*, pemilihan metode rekomendasi terbaik berdasarkan situasi
- *Mixed*, penggabungan rekomendasi dari berbagai metode ke dalam satu list
- *Feature combination*, penggabungan fitur dari berbagai metode
- *Feature augmentation*, yaitu penambahan fitur baru untuk meningkatkan rekomendasi
- *Cascade*, yang dapat membantu memutuskan ketika rekomendasi bertentangan
- *Meta-level*, dimana hasil dari satu metode digunakan sebagai input untuk metode lainnya

Dalam penelitian ini, metode hybrid *Feature Combination* akan digunakan untuk sistem rekomendasi makanan. Metode ini dipilih karena dapat menggabungkan kelebihan dari CF dan CBF secara optimal.

CF menggunakan data perilaku dan preferensi pengguna sebelumnya untuk memberikan rekomendasi, sementara CBF menggunakan informasi dan karakteristik *item*/makanan untuk membuat rekomendasi. Dengan menggabungkan fitur-fitur dari kedua metode tersebut, sistem rekomendasi dapat menghasilkan rekomendasi yang lebih akurat dan relevan bagi setiap pengguna.

Metode *Feature Combination* telah terbukti efektif dalam meningkatkan akurasi rekomendasi pada beberapa penelitian sebelumnya. Dalam penelitian [28], *feature combination* digunakan dalam sistem rekomendasi dengan menggabungkan pendekatan CF berbasis pengguna dengan informasi demografis. Hasil penelitian menunjukkan bahwa sistem dapat meningkatkan kualitas rekomendasi secara

keseluruhan dan dapat mengatasi tantangan umum dalam sistem rekomendasi, seperti data yang jarang dan akurasi yang lebih baik, dengan membandingkan berbagai klasifikasi menggunakan dataset yang besar.

Dalam penelitian [29], *feature combination* digunakan dalam sistem rekomendasi *playlist* musik. Dengan menggabungkan informasi kolaboratif dari *playlist* musik dengan vektor fitur lagu dari berbagai sumber, sistem akan lebih akurat memprediksi kelanjutan *playlist* yang sesuai, terutama untuk lagu-lagu yang jarang muncul dalam *playlist*. Dengan mengintegrasikan fitur lagu ke dalam model kolaboratif, sistem dapat mengatasi *data sparsity* dan meningkatkan representasi lagu, sehingga meningkatkan rekomendasi dalam skenario *cold-start* dan membantu dalam merekomendasikan lagu yang kurang populer.

Salah satu metode *hybrid* yang dapat digunakan dalam sistem rekomendasi memiliki tiga tahap, yaitu *item-based collaborative filtering similarity*, *item-based content similarity*, dan *hybrid prediction*.

Pada langkah *hybrid prediction*, prediksi penilaian dari *item a* yang tidak diketahui pengguna *u* dibagi menjadi dua langkah. *Weighted sum* menghitung jumlah penilaian yang diberikan oleh pengguna *u* terhadap *item* yang paling mirip $b \in I$ yang mirip dengan *item a*, digunakan dalam metode *item-based CF* dan *item-based content* untuk melakukan prediksi penilaian.

$$P_{z,a}^{CF} = \frac{\sum_{b \in I} (r_{u,b} \times \text{Sim}_{a,b}^{CF})}{\sum_{b \in I} |\text{Sim}_{a,b}^{CF}|} \quad (2.7)$$

$$P_{\approx,a}^{\text{Content}} = \frac{\sum_{b \in I} (r_{u,b} \times \text{Sim}_{a,b}^{\text{Content}})}{\sum_{b \in I} |\text{Sim}_{a,b}^{\text{Content}}|} \quad (2.8)$$

Kemudian, metode linear weighted hybridization digunakan untuk menggabungkan penilaian yang diprediksi dari metode *item-based CF* dan *item-based content* untuk menghasilkan prediksi akhir sebagai berikut:

$$P_{x,\omega}^{\text{Hybrid}} = \lambda \cdot P_{\alpha,\omega}^{CF} + (1 - \lambda) \cdot P_{\psi,\mu}^{\text{Content}} \quad (2.9)$$

λ dan $1 - \lambda \in [0, 1]$ masing-masing menunjukkan signifikansi relatif dari metode *item-based CF* dan *item-based content* pada prediksi akhir [30].

2.6 Vector Space Model

Vector Space Model (VSM) merupakan sebuah model yang digunakan dalam pencarian informasi untuk merepresentasikan dokumen dengan cara yang memudahkan perbandingan dan pencarian. Setiap dokumen direpresentasikan dengan sebuah titik di dalam sebuah ruang, di mana setiap dimensi berhubungan dengan sebuah kata unik dari dokumen tersebut. Keberadaan dan pentingnya sebuah kata dalam sebuah dokumen direpresentasikan dengan bobot di sepanjang dimensi tersebut. Ketika permintaan pencarian dibuat, permintaan tersebut juga berubah menjadi sebuah titik dalam ruang tersebut. Kedekatan antara dua titik (dokumen dan kueri, atau dua dokumen) menunjukkan kemiripan antara keduanya. VSM bertujuan agar dokumen-dokumen yang memiliki topik yang mirip saling berdekatan, sehingga dokumen yang relevan mudah untuk ditemukan ketika pencarian dibuat. Model ini sangat berguna karena memungkinkan penghitungan kemiripan menggunakan rumus matematika, membuat proses pencarian menjadi efisien dan efektif [31].

2.7 Singular Value Decomposition

Singular Value Decomposition (SVD) adalah sebuah teknik matematika yang digunakan untuk memecah matriks persegi panjang menjadi tiga matriks yang lebih sederhana, yaitu dua matriks ortogonal dan satu matriks diagonal. SVD dapat membantu menemukan pola dalam data dengan mengubahnya ke dalam format yang lebih mudah dianalisis. Dua matriks ortogonal merepresentasikan baris dan kolom dari matriks asli, namun dengan cara yang menyoroti hubungan antara baris dan kolom tersebut. Sementara itu, matriks diagonal berisi nilai tunggal yang merepresentasikan kekuatan dari hubungan tersebut [32].

2.8 End-User Computing Satisfaction

End-User Computing Satisfaction (EUCS) merupakan sebuah metode untuk mengukur tingkat kepuasan pengguna terhadap suatu sistem informasi. EUCS dapat digunakan untuk mengetahui seberapa baik sistem memenuhi kebutuhan pengguna dengan membandingkan harapan dan kenyataan dari sistem [33].

Model EUCS terdiri dari lima aspek utama yang mempengaruhi kepuasan pengguna yaitu:

- *Content*, yang meliputi kualitas dan kelengkapan informasi yang disediakan oleh sistem.
- *Accuracy*, yang meliputi ketepatan dan kebenaran informasi yang dihasilkan oleh sistem
- *Format*, yang meliputi kemudahan membaca dan memahami informasi yang ditampilkan oleh sistem
- *Ease of Use*, yang meliputi kemudahan pengguna dalam mempelajari dan menggunakan sistem
- *Timeliness*, yaitu ketepatan waktu sistem dalam menyediakan informasi yang dibutuhkan pengguna

