



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## BAB II

### LANDASAN TEORI

#### 2.1 Sistem Rekomendasi

Sistem rekomendasi merupakan suatu aplikasi untuk menyediakan dan merekomendasikan suatu *item* dalam membuat suatu keputusan yang diinginkan oleh pengguna (Ungkawa, 2013). Penerapan rekomendasi didalam sebuah sistem biasanya melakukan prediksi suatu *item*, seperti rekomendasi film, musik, buku, berita dan lain sebagainya yang menarik pengguna. Sistem ini berjalan dengan mengumpulkan data dari pengguna secara langsung maupun tidak (Fadlil & Mahmudy, 2010).

Pengumpulan data secara langsung dapat dilakukan sebagai berikut :

1. Meminta *user* untuk melakukan *rating* pada sebuah *item*.
2. Meminta *user* untuk melakukan *rating* pada *item* favorit setidaknya memilih satu *item* favorit.
3. Memberikan beberapa pilihan *item* pada *user* dan memintanya memilih yang terbaik.
4. Meminta *user* untuk mendaftar *item* yang paling disukai atau *item* yang tidak disukainya.

Pengumpulan data dengan tidak langsung berhubungan dengan seorang *user*, dilakukan dengan cara mengamati *item* yang dilihat oleh seorang *user* pada sebuah *web ecommerce*. Dari data hasil yang dikumpulkan tersebut, kemudian diolah dengan menggunakan algoritma tertentu. Setelah itu, hasilnya tersebut dikembalikan lagi kepada *user* sebagai sebuah rekomendasi *item* dengan parameter

dari *user* tersebut. Sistem rekomendasi juga merupakan salah satu alternatif sebagai mesin pencari suatu *item* yang dicari oleh *user*

## 2.2 Collaborative Filtering

*Collaborative filtering* (CF) merupakan proses penyaringan atau pengevaluasian *item* dengan menggunakan opini dari orang lain. Ide utamanya adalah untuk mengeksploitasi informasi mengenai perilaku di masa lampau maupun opini dari suatu komunitas pengguna yang kemudian digunakan untuk memprediksi *item* mana yang akan disukai atau menarik bagi seorang pengguna. CF murni menggunakan matriks yang berisi *user-item rating* sebagai satu-satunya input, sedangkan output yang dihasilkan ada dua jenis :

- a. prediksi (numerik) yang mengindikasikan seberapa besar tingkat kesukaan seorang pengguna terhadap sebuah *item*, dan
- b. sebuah daftar berisi *item* yang direkomendasikan
- c. Istilah pengguna (*user*) dalam CF mengacu kepada mereka yang memberi penilaian terhadap *item* di dalam sistem, sekaligus nantinya menerima rekomendasi dari sistem.

Tabel 2. 1 Representasi *matrix user-item rating*

| <i>User</i> | <i>Item 1</i> | <i>Item 2</i> | <i>Item 3</i> |
|-------------|---------------|---------------|---------------|
| Gaby        | 1             | 1             | 0             |
| User 1      | 1             | 1             | 1             |
| User 2      | 1             | 0             | 0             |
| User 3      | 0             | 1             | 1             |

*Rating* dapat dikumpulkan dengan cara eksplisit, implisit, atau keduanya bersamaan. *Rating* eksplisit adalah ketika seorang pengguna secara langsung diminta untuk memberikan opini terhadap suatu *item*. *Rating* implisit mengandung

arti bahwa sistem secara otomatis mendapatkan preferensi pengguna secara pasif dengan memonitor aksi pengguna. Penilaian hanya didasarkan pada perilaku pengguna, misalnya ketika seorang member di perpustakaan memutuskan untuk meminjam suatu *item* buku maka member tersebut dianggap tertarik atau menyukai *item* tersebut, dan sebaliknya dianggap tidak tertarik atau tidak menyukai apabila tidak meminjamnya. Dengan cara ini, user profile dibentuk tanpa melibatkan *effort* tambahan dari *user*. Kekurangan dari cara ini tentu saja bahwa dugaan feedback bisa jadi tidak tepat. *Rating* dalam *collaborative filtering* dapat berbentuk:

- a. Model *rating* skalar yang terdiri atas *rating* numerik seperti 1 sampai 5.
- b. Model *rating* biner dengan memilih antara setuju atau tidak setuju, atau dapat pula baik atau buruk.
- c. *Rating* unary dapat mengindikasikan bahwa pengguna telah mengobservasi atau membeli *item* atau merating *item* dengan positif.

Tidak tersedianya *rating* mengindikasikan tidak terdapat informasi yang menghubungkan pengguna dengan *item*. *Rating* dapat dikumpulkan secara eksplisit, implisit, ataupun gabungan antara eksplisit dan implisit. *Rating* eksplisit yaitu *rating* yang didapatkan pada saat pelanggan/pengguna diminta menyediakan opini terhadap *item* tertentu. *Rating* implisit yaitu *rating* yang didapatkan melalui aksi yang dilakukan pelanggan.

Schafer membagi algoritma *collaborative filtering* ke dalam dua kelas yang berbeda menurut teori dan kepraktisannya, yaitu algoritma probabilistik dan non-probabilistik. Suatu algoritma dianggap probabilistik bila algoritma tersebut berdasarkan model *probabilistic*. Algoritma tersebut mewakili distribusi probabilitas saat menghitung prediksi *rating* atau daftar *ranking* rekomendasi.

Algoritma non-probabilistik yang terkenal yaitu *nearest neighbours algorithm*. Algoritma ini dibagi menjadi dua kelas yaitu *user-based* dan *item-based* (Schafer dkk, 2007).

*User-Based Collaborative Filtering* menggunakan teknik statistika untuk menemukan sekumpulan pengguna, dikenal sebagai tetangga (*neighbour*), yang memiliki sejarah setuju dengan pengguna yang menjadi sasaran. Setelah sekumpulan tetangga terbentuk, sistem menggunakan algoritma yang berbeda untuk menggabungkan kesukaan *neighbours* untuk menghasilkan prediksi atau rekomendasi N-teratas untuk *active user* (Sarwar dkk, 2001).

Untuk menghitung nilai kemiripan antar *user* dapat digunakan persamaan *Euclidean Distance Score* seperti yang terdapat pada rumus 2.1. Berikut adalah rumus menghitung nilai kemiripan antar *user* (Sarwar, 2011):

$$d(p1, p2) = \sqrt{\sum_{i \in item} (S_{p1} - S_{p2})^2}$$

...(2.1)

### 2.3 Prediksi Rating

Langkah selanjutnya adalah menghitung prediksi *rating* dari *item* tersebut berdasarkan informasi yang diberikan oleh *user-user* yang memiliki nilai kemiripan dengan *user* tersebut. Perhitungan *rating* dilakukan dengan cara membandingkan *rating* yang pernah diberikan pengguna pada suatu *item* dengan kemiripan antara *item* tersebut dengan *item* lainnya. Metode yang dipakai adalah metode *weighted sum*. Berikut adalah rumus prediksi Weighted Sum (Sarwar, 2011).

$$P_{(u,t)} = \frac{\sum \text{all similiar items, } N(S_i, N * R_u, N)}{\sum \text{all similiar items, } N(|S_i, N|)} \dots(2.2)$$

### 2.3 Skala Likert

Skala Likert adalah metode pengukuran yang digunakan untuk mengukur sikap, pendapat dan persepsi seseorang atau sekelompok orang tentang fenomena sosial (Supriyatna, 2015). Pilihan terhadap masing-masing jawaban untuk tanggapan responden atas dimensi kualitas kepuasan mulai dari sangat setuju, setuju, ragu-ragu, tidak setuju, dan sangat tidak setuju masing-masing akan diberi skor yang berbeda-beda.

Tabel 2. 2 Skala Likert (Supriyatna, 2015)

| No | Pilihan Jawaban           | Skor |
|----|---------------------------|------|
| 1  | Sangat Setuju (SS)        | 5    |
| 2  | Setuju (S)                | 4    |
| 3  | Ragu-ragu (RR)            | 3    |
| 4  | Tidak Setuju (TS)         | 2    |
| 5  | Sangat Tidak Setuju (STS) | 1    |

Dengan adanya tabel skala jawaban pada Skala Likert, maka dapat dilakukan perhitungan untuk mendapatkan nilai skor akhir dari pengukuran yang dilakukan. Digunakan rumus sebagai berikut.

Setelah mendapatkan hasil perhitungan diatas, maka skor masing-masing setiap perhitungan dijumlahkan untuk mendapatkan total hasilnya. Kemudian dilakukan perhitungan persentase untuk mengetahui jumlah jawaban dari para responden dengan rumus sebagai berikut.

Kemudian, presentase nilai skor pada suatu kuesioner dapat dihitung menggunakan rumus yang dijelaskan oleh Sugiyono (2012).

$$\text{Persentase Skor} = (((\text{Sangat Setuju} * 5) + (\text{Setuju} * 4) + (\text{Netral} * 3) + (\text{Tidak Setuju} * 2) + (\text{Sangat Tidak Setuju} * 1)) / (5 * \text{Jumlah Responden})) * 100\% \quad \dots(2.3)$$

Setelah mendapatkan hasil perhitungan semuanya maka hasil akhir dari persentase diubah ke dalam nilai interval seperti pada Tabel 2.3.

Tabel 2. 3 Skor dan Interval Skala Linkert Lima Tingkat

| <b>Pernyataan</b>                               | <b>Skor Positif</b> | <b>Interval</b>   |
|---|---------------------|-------------------|
| Sangat Setuju/Selalu                            | 5                   | Skor >= 80%       |
| Setuju / Sering                                 | 4                   | 80% > Skor >= 60% |
| Ragu – Ragu / Kadang<br>–<br>Kadang/ Biasa Saja | 3                   | 60% > Skor >= 40% |
| Tidak Setuju                                    | 2                   | 40% > Skor >= 20% |
| Sangat Tidak Setuju                             | 1                   | Skor < 20%        |

Dalam penelitian ini, Skala Likert akan digunakan untuk memberikan penilaian terhadap pertanyaan yang terdapat pada kuisisioner.

## 2.4 Kepuasan Pengguna

Pengujian aplikasi dilakukan dengan penyebaran kuesioner kepada para pengguna yang telah menggunakan aplikasi. Terdapat beberapa cara untuk mengukur tingkat kepuasan pengguna, salah satunya dengan menggunakan metode *End-user Computing Satisfaction* (EUCS). Menurut Dewa (2016), instrumen *EUCS* mencakup 5 komponen, yaitu isi, keakuratan, bentuk, kemudahan penggunaan dan ketepatan waktu. Berikut adalah penjelasan dari tiap indikator *EUCS*.

1. Isi (*Content*), mengukur kepuasan pengguna ditinjau dari isi suatu sistem. Isi sistem biasanya berupa fungsi dan modul yang dapat digunakan oleh pengguna sistem dan juga informasi yang dihasilkan oleh sistem.
2. Akurasi (*Accuracy*), mengukur kepuasan pengguna dari sisi keakuratan data ketika sistem menerima input kemudian mengolahnya menjadi informasi.
3. Bentuk (*Format*), mengukur kepuasan pengguna dari sisi tampilan dan estetika antarmuka sistem.
4. Kemudahan pengguna (*Ease of Use*), mengukur kepuasan pengguna dari sisi kemudahan pengguna atau *user friendly* dalam menggunakan sistem, seperti proses memasukkan data, mengolah data, dan mencari informasi yang dibutuhkan.
5. Ketepatan waktu (*timeliness*), mengukur kepuasan pengguna dari sisi ketepatan waktu sistem dalam menyajikan atau menyediakan data dan informasi yang dibutuhkan oleh pengguna.

Dengan adanya pengukuran kepuasan pengguna, maka dapat dibentuk beberapa pertanyaan yang digunakan pada kuesioner untuk mengetahui tingkat kepuasan pengguna yang menggunakan aplikasi rekomendasi ini. Skala Likert digunakan untuk memberikan penilaian terhadap pertanyaan yang ada.

## **2.5 Sistem Rekomendasi**

Provokatif *coffee and space* pertamakali dijalankan pada bulan september 2017. Pemilik dari Provokatif *coffee and space* adalah seorang peminat kopi, dengan alasan itulah beliau memulai usahanya dengan mengajak beberapa temannya untuk bekerjasama menjalankan usaha *coffee shop* tersebut (Ibnu, 2019).

Nama dari *coffee shop* itu sendiri memiliki arti yaitu Provokatif *coffee and space* yang dimaksud *space* disini adalah diberikan ruang atau sarana kepada *costumer* jika membutuhkan *space* untuk melakukan kegiatan dan aktivitas, sembari mencoba minuman kopi yang di sediakan, nama provokatif sendiri didapatkan ketika pemilik provokatif melakukan konsultasi bersama dengan rekannya dan mengambil nama provokatif dari istilah memprovokasi maka jadilah nama provokatif *coffe and space* (Ibnu, 2019).