

BAB III

Landasan Teori

3.1 Sistem Rekomendasi

Sistem rekomendasi merupakan suatu aplikasi untuk menyediakan dan merekomendasikan suatu item dalam membuat suatu keputusan yang diinginkan oleh pengguna (Ungkawa, et al., 2013). Penerapan rekomendasi didalam sebuah sistem biasanya melakukan prediksi suatu item, seperti rekomendasi film, musik, buku, berita dan lain sebagainya yang menarik *user*. Sistem ini berjalan dengan mengumpulkan data dari *user* 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 ranking 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 *e-commerce*. 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*.

Dalam pembangunan sistem rekomendasi, ada beberapa macam metode untuk menyelesaikan permasalahan, antara lain *user-based collaborative filtering*, *content-based filtering*, dan *hybrid*. Namun beberapa peneliti menambahkan metode baru, yaitu *knowledge based recommendation*. Secara umum, Metode *user-based collaborative filtering* menggunakan *feedback*, ulasan dan rating untuk mendapatkan hasil rekomendasi. Metode *content-based* memberikan rekomendasi dengan membangun *profile* pengguna. Metode *hybrid-based* menggabungkan dua atau lebih metode. Penggabungan dilakukan dengan tujuan saling melengkapi kekurangan dari metode yang digunakan. Sedangkan metode *knowledge-based* menggunakan pola pengetahuan untuk memberikan hasil rekomendasi.

3.2 Content-Based Filtering

Ada berbagai macam metode pendekatan yang digunakan untuk menyelesaikan permasalahan pada sistem rekomendasi, antara lain *user-based collaborative filtering*, *content-based filtering* dan *hybrid*. Dalam kasus pemilihan pekerjaan untuk mahasiswa program sudi Teknik Informatika UAJY, metode yang digunakan adalah metode *content-based filtering*. *Content-based filtering* memberikan suatu rekomendasi berdasarkan hasil analisa kemiripan item yang telah dinilai oleh penggunanya (Adi, 2010) *Content-Based*

Filtering membentuk profil penggunaannya berdasarkan atribut pembentuk suatu item. *Cosine Similarity* yang tertinggi yang akan menjadi hasil rekomendasi.

Content-Based Filtering membentuk profil penggunaannya berdasarkan atribut pembentuk suatu item. Sebagai contoh untuk suatu dokumen, atribut pembentuknya adalah kata-kata yang terdapat pada dokumen tersebut. Parameter pembentuk profil pengguna ini juga diberi nilai bobot berdasarkan kriteria tertentu. Adapun langkah - langkah algoritmanya :

1. Suatu item barang dibagi-bagi berdasarkan suatu vektor komponen pembentuknya.
2. Sistem akan membuat profil pengguna berdasarkan bobot vektor komponen pembentuk suatu item. Pembuatan profil pengguna dapat menggunakan algoritma TF-IDF (*term frequency-invers document frequency*). TF adalah jumlah term dalam suatu dokumen. Sedangkan nilai IDF dapat dihitung menggunakan rumus:

$$idf_i = \log \left(\frac{n}{df_i} \right) \dots\dots\dots (3.1)$$

n merupakan jumlah semua dokumen sedangkan df adalah jumlah dokumen yang memiliki term i.

3. Berdasarkan profil pengguna tersebut, sistem akan memperkirakan penilaian suka atau tidak suka suatu item berdasarkan analisis kemiripan profil pengguna dengan vektor komponen pembentuk item. Jika sistem memperkirakan bahwa item tersebut akan disukai oleh

pengguna maka item tersebut akan direkomendasikan ke pengguna.

3.3 Cosine Similarity

Menurut Manning, Raghavan, dan Schutze (2009), *cosine similarity* digunakan untuk mengukur kedekatan antara dua vektor. Selain menggunakan cosine, untuk mengukur kedekatan antar vektor dengan menggunakan fungsi similaritas. Beberapa fungsi similaritas yang sering dijumpai adalah *Jaccard*, *Overlap*, *Assimetric*, *Minowski Distance*, *Pearson Corrlation*, dan *Cosine*. Untuk tujuan klastering yang baik adalah *Cosine Similarity*. Berikut adalah persamaan dari metode *Cosine Similarity* :

$$\text{Similarity}(x,y) = \frac{\sum_{i=1}^t x_i y_i}{\sqrt{\sum_{i=1}^t x_i^2 \cdot \sum_{i=1}^t y_i^2}} \dots\dots\dots (3.2)$$

dimana :

x dan y adalah vector yang berbeda.

x_i = term i yang ada pada vector x

y_i = term I yang ada pada vector y

Pada hasil perhitungan cosine similarity, hasil pehitungan tertinggi yang menjadi vektor yang terdekat dan vektor yang ingin dibandingkan. Pada kasus SIREP, nilai tertinggi adalah rekomendasi pekerjaan yang akan berikan kepada user.

3.4 Sum Square Error (SSE)

Sum Square Error adalah proses perhitungan yang bertujuan untuk menghitung seberapa besar error kemungkinan yang bisa terjadi pada sistem. Berikut adalah rumus perhitungan SSE :

$$SSE = \sum_{i=1}^n (x_i - \bar{X})^2 \dots\dots\dots (3.3)$$

dimana:

x_i = nilai yang akan dihitung seberapa besar errornya

\bar{X} = nilai rata-rata dalam satu cluster dimana x_i berada

Pada hasil perhitungan SSE semakin kecil nilai SSE-nya akan semakin baik. Sebaliknya, semakin besar nilai SSE-nya, maka semakin buruk.

3.5 Dunn Index

Dunn index adalah salah satu pengukuran cluster validity yang diajukan oleh J.C.Dunn. Cluster validity ini berlandaskan pada fakta bahwa klaster yang terpisah itu biasanya memiliki jarak antar klaster yang besar dan diameter intra klaster yang kecil (Gita & Saikhu, n.d.). Hasil perhitungan dunn index, semakin besar semakin baik.

$$D(C) = \frac{\min_{C_k, C_l \in C, C_k \neq C_l} \left(\min_{i \in C_k, j \in C_l} \text{dist}(i, j) \right)}{\max_{C_m \in C} \text{diam}(C_m)} \dots\dots\dots (3.4)$$

3.6 Silhouette

Silhouette merupakan validasi yang digunakan untuk menentukan baik atau buruknya klaster yang dibentuk. Dalam silhouette memiliki rentang -1 sampai 1. Dikatakan klaster itu baik, apabila hasil perhitungan silhouette semakin mendekati 1. Sebaliknya, klaster dikatakan buruk, apabila silhouette mendekati -1 (Gita & Saikhu, n.d.).

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \dots\dots\dots (3.5)$$

Nilai a adalah rata-rata dari jarak objek i ke objek-objek lain yang berada pada satu klaster. Sedangkan nilai b adalah minimal dari rata-rata 6 jarak objek i ke objek-objek lain yang berada pada klaster yang berbeda.

3.7 Structured Query Language

SQL (*Structured Query Language*) adalah sebuah bahasa yang dipergunakan untuk mengakses data dalam basis data relasional. Bahasa ini secara *de facto* merupakan bahasa standar yang digunakan dalam manajemen basis data relasional. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya (Adelia & Setiawan, 2011).