

BAB II

LANDASAN TEORI

2.1 Pengertian Perpustakaan

Defenisi Perpustakaan Menurut Undang-Undang Nomor 43 Tahun 2007 tentang perpustakaan, perpustakaan adalah institusi pengelola karya tulis, karya cetak, dan/atau karya rekam secara professional dengan sistem yang baku guna memenuhi kebutuhan pendidikan, penelitian, pelestarian, informasi, rekreasi para pemustaka.

Adapun Defenisi Perpustakaan menurut para ahli adalah suatu unit kerja dari satu badan atau lembaga tertentu yang mengelola bahan-bahan pustaka, baik berupa buku-buku maupun bukan berupa buku (non book material) yang diatur secara sistematis menurut aturan tertentu sehingga dapat digunakan sebagai sumber informasi oleh setiap pemakainya.

2.2 Pengertian Sistem Rekomendasi

Sistem rekomendasi adalah suatu sistem yang dirancang untuk memprediksi suatu item yang sesuai dengan minat user, yang mana item tersebut akan direkomendasikan pada user (Sanjung,2011). Prediksi informasi minat pengguna dapat diperoleh berdasarkan pola aksi perilaku pengguna atau sering dikatakan sebagai profil pengguna. Salah satu cara yang dilakukan untuk membentuk suatu profil pengguna adalah, misanya ketika pengguna melakukan transaksi peminjaman maka data peminjaman ini akan disimpan sebagai profil pengguna. Berdasarkan profil pengguna yang telah terbentuk akan diketahui minat buku yang disukai oleh pengguna.

Pada sistem rekomendasi, terdapat beberapa item yang akan disaring untuk direkomendasikan kepada pengguna berdasarkan profil pengguna, skala rating, daln lain-lain sehingga menghasilkan beberapa item yang direkomendasikan kepada pengguna. Saat ini telah sistem yang mengadopsi sistem rekomendasi pada penerapannya. Seperti halnya pada sosial media (twitter, facebook), youtube.com, amazon.com dan lain-lain.

2.3 Metode Dalam Sistem Rekomendasi

Dalam sistem rekomendasi terdapat tiga bentuk metode yang digunakan dalam pembangkitan rekomendasi. Metode tersebut adalah Content-based, Collaboratif Filtering, dan Hybrid Recommender system.

a. *Content Based Recommender System.*

Metode *content based* membuat rekomendasi dengan menganalisa deskripsi setiap *item* untuk mengidentifikasi *item* mana yang mempunyai hubungan ketertarikan khusus dari seorang *user*. Deskripsi ketertarikan *user* diperoleh dari profil *user* yang didasarkan atas penilaian menarik atau tidaknya suatu *item* yang ada pada *recommender system* ini (Pazzani dan Billsus, 2007).

b. *Collaborative Filtering Recommender System.*

Pendekatan *collaborative filtering* merekomendasikan *item* kepada *user* berdasarkan korelasi antara *rating user* yang berbeda terhadap *item* atau transaksi *user* di dalam sistem. Sementara *user* merating *item*, sistem akan menggunakan data *rating user* tertentu untuk melakukan perhitungan prediksi dan memberikan rekomendasi *item* yang belum pernah dirating oleh *user* tersebut (Leben, 2010).

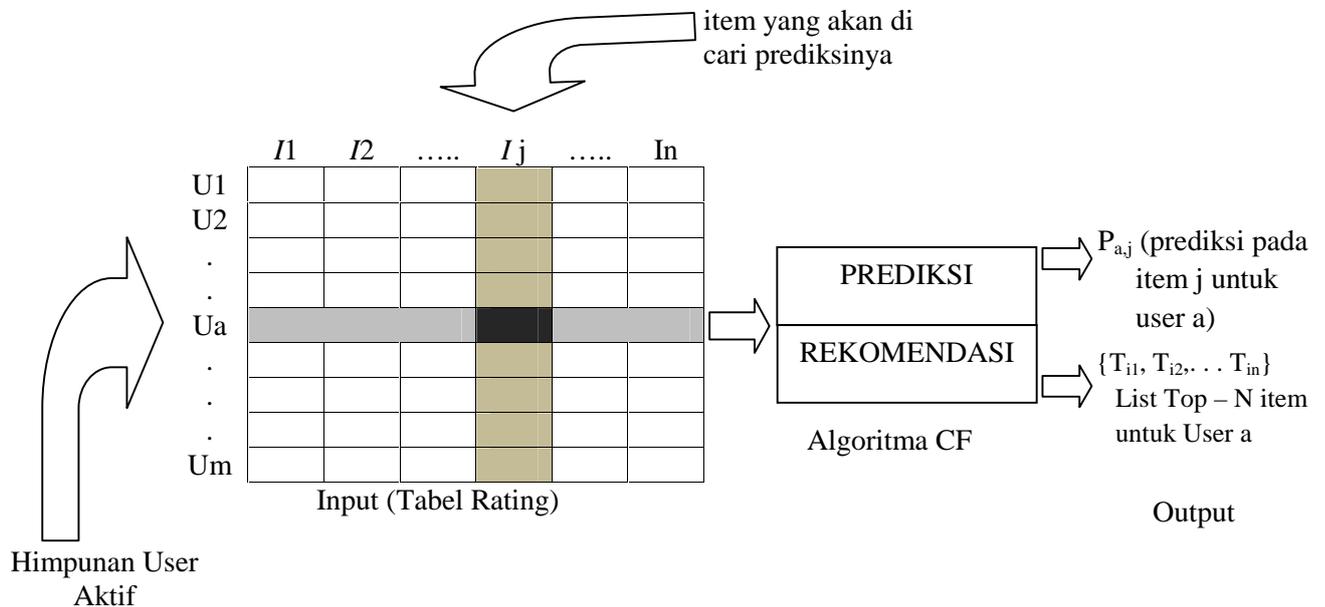
c. *Hybrid Recommender System.*

Secara umum pendekatan *hybrid recommendations* adalah dengan menggabungkan lebih dari satu metode yang ada pada *recommender system*, kombinasi yang ada pada teknik ini misalnya dengan menggabungkan metode *content based* dengan *collaborative filtering* (Leimstoll dan Stormer, 2007).

2.4 Collaborative Filtering

Collaborative filtering merupakan proses penyaringan atau pengevaluasian item menggunakan opini orang lain (Schafer dkk, 2007). Pada prosesnya metode Collaborative filtering melakukan penyaringan data berdasarkan tingkah laku karakteristik pengguna sehingga dapat memberikan informasi yang baru kepada pengguna lainnya karna system memberikan informasi berdasarkan pola satu kelompok pengguna yang hampir sama.

Berikut merupakan skema dari Collaboratif Filtering dalam pemberian rekomendasi kepada user aktif.



Gambar . 2.1 Skema Collaborative Filtering

Dalam skenario CF terdapat daftar pengguna m user $U = \{u_1, u_2, \dots, u_m\}$ dan daftar item $I = \{p_1, p_2, \dots, p_n\}$. Setiap u_i user mengekspresikan pendapatnya tentang daftar item miliknya. Kumpulan set dari pendapat itu disebut dengan rating dari user u_i dan dilambangkan dengan I_{u_i} . Setelah sistem ini menentukan ketetanggaan terdekat, maka sistem akan merepresentasikan item yang mungkin disukai user dalam dua bentuk (Sarwar dkk, 2001), yaitu:

1. Prediksi, merupakan nilai numerik dimana $P_{a,j}$ adalah nilai prediksi rating item j yang mungkin disukai oleh active user (U_a). Nilai prediksi ini digunakan dengan skala yang sama dengan nilai yang disediakan (misalnya, dari skala 1 sampai 5).
2. Rekomendasi adalah daftar N item yang mungkin akan disukai oleh user U_a . Daftar yang direkomendasikan biasanya terdiri dari item yang belum pernah dibeli atau dirating oleh active user. Output dari algoritma CF ini juga dikenal sebagai *Top-N Recommendation*.

Gambar 2.1 menunjukkan diagram skema dari proses *collaborative filtering*. Algoritma CF merepresentasikan seluruh $m \times n$ *user-item* sebagai matriks *rating* dimana setiap entri merupakan nilai *rating* dari *user* untuk setiap *item*. *Active user* (U_a) pada skema ini merupakan *user* yang akan dicari *item* yang mungkin disukainya dengan menggunakan algoritma CF (Sarwar dkk, 2001).

2.5 Algoritma Collaborative Filtering

Menurut teori dan penggunaannya Schafer membagi algoritma collaborative filtering ke dalam dua kelas yang berbeda yaitu dengan menggunakan algoritma probabilistik dan non probabilistik. Suatu algoritma dianggap probabilistik bila algoritma tersebut berdasarkan model probabilistik. Algoritma tersebut menggunakan opini orang lain (Schafer dkk, 2007).

Model peratingan dalam sistem rekomendasi berbasis Collaboratif Filtering ini memiliki dua bentuk yaitu :

a. *User-based* Collaboratif Filtering

Model *User-based* collaborative filtering mengasumsikan bahwa cara yang baik untuk menemukan *item* yang menarik bagi *user* tertentu adalah dengan mencari *user* lain yang memiliki minat yang sama. Jadi, pada awalnya *user-based* CF mencoba untuk menemukan *user neighbor* berdasarkan pada *user similarity* dan kemudian setiap nilai *rating* dari *user neighbor* akan dijadikan bahan rekomendasi bagi *user* aktif.

b. *Item-based* Filtering

Pada model *Item-based* collaborative filtering memiliki skema yang hampir sama dengan *user-based*, jika sebelumnya *user-based* yang dicari adalah korelasi antara *user*, maka pada *item-based* collaborative filtering korelasi yang dicari adalah antar *item* yang disukai oleh *user* kemudian *item* yang berkorelasi tersebut direkomendasikan terhadap sejumlah *user* lainnya (Gong, 2010).

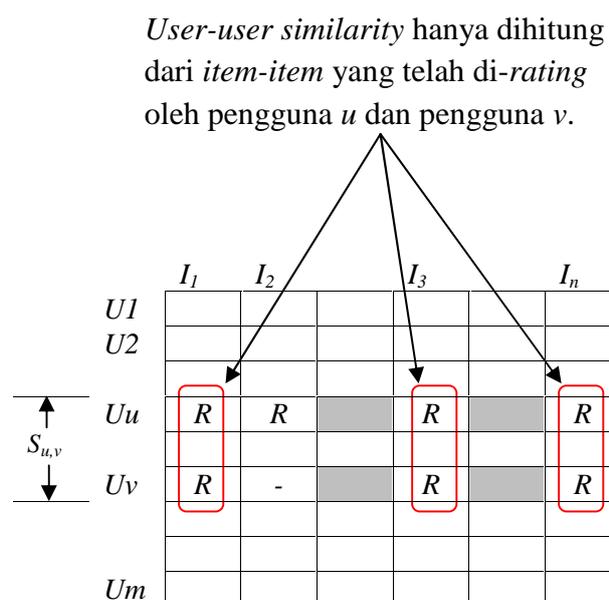
2.6 *User-based* Collaborative Filtering

User-based nearest neighbour algorithm 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 user aktif. (Sarwar dkk, 2001).

Pada pendekatan user based collaborative filtering sistem memberikan rekomendasi kepada user item-item yang disukai atau dirating oleh user-user lain yang memiliki banyak kemiripan dengannya. Misalnya, user a menyukai atau merating item 1, 2 dan 3, kemudian user b menyukai item 1, 2 dan 4 maka sistem akan merekomendasikan item3 kepada user b dan item 4 kepada user a.

Kelebihan dari pendekatan user based collaborative filtering adalah dapat menghasilkan rekomendasi yang berkualitas baik. Berikut merupakan skema user based filtering.



Gambar. 2.2 Pola User Based Filtering

2.7 Penemuan *Similar User*

Tahap ini merupakan tahap untuk mencari *user* yang memiliki kesamaan minat. Setiap *user* yang pernah merating *item* akan dibandingkan dengan *user* lain yang belum pernah merating *item*. Hasil dari tahapan ini adalah nilai *similarity* (kemiripan) yang menggambarkan seberapa mirip suatu *user* dengan

user lain. Terdapat beberapa Algoritma untuk menemukan *similar user*, yaitu: Cosine-based Similarity, Adjusted-cossine similarity, dan Correlation-based Similarity (Sarwar,2001).

a. Algoritma Cosine-based Similarity

Pada kasus ini dua item dianggap sebagai 2 vektor. Kesamaan antara 2 item ini diukur dengan menghitung kosinus dari sudut antara 2 vektor item. Item dibandingkan misalnya u dan v , dianggap sebagai sebuah vektor baris dengan anggotanya adalah nilai rating yang diberikan terhadap kedua item tersebut. Dua vektor dikatakan sama jika membentuk sudut 0° atau nilai kosinusnya 1. Dengan kata lain dua item dikatakan mirip jika nilai kosinus dari perhitungan mendekati 1 (Wiranto, 2010).

Persamaan cosine-based similarity:

$$sim(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\|_2 \cdot \|\vec{j}\|_2} \quad (2.1)$$

Dimana \vec{i} dan \vec{j} merupakan vektor vektor baris dengan anggota nilai *rating* pada *item* i dan *item* j . $\cos(\vec{i}, \vec{j})$ merupakan nilai cosinus sudut yang dibentuk vektor baris *ratingitem* i dan j .

b. Algoritma Adjusted-cossine similarity

Persamaan adjusted cosine similarity dapat digunakan untuk menghitung nilai kemiripan antar user. Perhitungan kemiripan ini merupakan modifikasi dari perhitungan kemiripan berbasis vektor dimana dengan melihat fakta bahwa setiap item di rating dengan skema rating yang berbeda-beda. Terkadang user memberi rating yang tinggi terhadap item a disisi lain user memberi rating yang sangat rendah pada item b . Maka dari itu untuk setiap rating dikurangi dengan rata-rata rating yang diberikan user.

Persamaan untuk adjusted cosine similarity adalah sebagai berikut ;

$$sim(a, b) = \frac{\sum_{u \in U} (R_{u,a} - \overline{R_u}) (R_{u,b} - \overline{R_u})}{\sqrt{\sum_{u \in U} (R_{u,a} - \overline{R_u})^2} \sqrt{\sum_{u \in U} (R_{u,b} - \overline{R_u})^2}} \quad (2.2)$$

Keterangan :

$Sim(a,b)$ = Nilai kemiripan antara *user* a dan *user* b

- $u \in U$ = Himpunan *item* yang di-*rating* oleh *user* a dan b
 $R_{u,i}$ = *Rating item* u oleh *user* a
 $R_{u,j}$ = *Rating item* u oleh *user* b
 \bar{R}_i = Nilai rata-rata *rating* pada *item*

Untuk menghitung nilai kemiripan (*similarity*) antar 2 *user*, diperlukan himpunan *item* yang di-*rating* oleh *user* tersebut. Nilai yang dihasilkan pada persamaan *adjusted-cosine similarity* adalah berkisar antara +1.0 dengan -1.0. *User* dianggap saling berkolerasi jika nilai *similarity* antara kedua *user* tersebut mendekati +1, begitu juga sebaliknya *user* dianggap tidak berkolerasi apabila nilai *similarity*-nya mendekati -1.

c. Algoritma Pearson correlation Similarity

Pada algoritma ini kemiripan antara dua *user* a dan b diukur dengan menghitung korelasi Pearson-r correlation. Agar perhitungan korelasi yang diperoleh akurat, terlebih dahulu dilakukan pemisahan terhadap co-rated user (*user-user* yang kedua *user* a dan b merating suatu *item* yang sama).

Persamaan person corelation similarity :

$$sim(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_i)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_j)^2}} \quad (2.3)$$

Keterangan :

- $Sim(a,b)$ = Nilai kemiripan antara *user* a dan *user* b
 $u \in U$ = Himpunan *item* yang di-*rating* oleh *user* a dan b
 $R_{u,a}$ = *Rating item* u oleh *user* a
 $R_{u,b}$ = *Rating item* u oleh *user* b
 \bar{R}_u = Nilai rata-rata *rating user* pada *item* u

Pada penelitian kali ini, algoritma pearson correlation similarity dipilih untuk digunakan dalam menentukan nilai similarity antar user. Berdasarkan nilai hasil perhitungan similarity, dapat dilakukan perhitungan pembangkitan rekomendasi yang didasari oleh similarity antar *user*.

2.8 Pembangkitan Rekomendasi

Berdasarkan nilai similarity antar user, maka langkah selanjutnya adalah proses perhitungan prediksi. Proses prediksi yang dilakukan adalah dengan memperkirakan nilai *rating* dari *item* terhadap seorang *user* yang belum pernah *me-rating item* tersebut. Algoritma yang dapat digunakan untuk mendapatkan prediksi dari suatu *user* yaitu algoritma *weighted sum* dan Algoritma *Regression* (Sarwar, 2001). Berikut merupakan penjelasan kedua algoritma tersebut.

a. Algoritma *weighted sum*

Algoritma ini mendapatkan nilai prediksi dengan menghitung total *rating* pada *item* yang diberikan oleh *user* yang mirip dengan *user* yang ingin diprediksi. Teknik ini memprediksi *user* *a* untuk *item* *i* dengan menghitung jumlah *rating* *item* yang diberikan oleh *user* yang berkorelasi dengan *user* *a*. Setiap *rating* yang diberikan *item* pada *user* yang berkorelasi dengan *user* *i* akan dikalikan dengan nilai kemiripannya. Kemudian dibagi dengan jumlah nilai absolut kemiripan seluruh *user* yang berkorelasi. Persamaan *weighted sum* :

$$P(i, a) = \frac{\sum_{i \in I} (R_{i,a} * S_{a,b})}{\sum_{i \in I} |S_{a,b}|} \quad (2.4)$$

Keterangan :

$P(i, a)$ = Prediksi untuk *item* *i* terhadap *user* *a*

$i \in I$ = Himpunan *user* yang mirip dengan *user* *a*

$R_{i,a}$ = *Rating* *item* *i* oleh *user* *a*

$S_{a,b}$ = Nilai kemiripan antar *user* *a* dan *user* *b*

Prediksi dengan persamaan *weighted sum* diperoleh dengan menghitung total keseluruhan dari perkalian antara *rating* *item* terhadap suatu *user* dan nilai *similarity* *user* tersebut ($\sum_{i \in I} (R_{i,a} * S_{a,b})$). Selanjutnya menghitung total keseluruhan dari nilai *similarity* *user* yang mirip dengan *user* yang akan prediksi ($\sum_{i \in I} |S_{a,b}|$). Langkah terakhir yaitu membagi langkah pertama dengan langkah kedua, yang mana akan menghasilkan nilai prediksi untuk *user* terhadap *item* *i* ($P(i, a)$).

b. Algoritma *Regression*

Pendekatan ini mirip dengan metode *weighted sum* tapi metode ini tidak langsung menggunakan *rating* dari *item* yang mirip melainkan menggunakan penaksiran *rating* berdasarkan model regresi. Model ini memperkirakan nilai prediksi *rating* dari R'_N berdasarkan *linear regression model*.

Persamaan *regression*:

$$\bar{R}'_N = \alpha \bar{R}_i + \beta + \epsilon \quad \dots(2.5)$$

Parameter regresi α dan β ditentukan berdasarkan kedua *ratingvector* dan merupakan *error* dari model regresi.

2.9 Skala Penilaian

Skala penilaian sering digunakan peneliti untuk mengetahui pendapat responden terhadap bahan penelitian yang dilakukan. Peneliti meminta responden untuk memberikan penilaian terhadap sekumpulan *item*. Hal ini dilakukan untuk mengetahui seberapa besar kesukaan responden terhadap *item* tersebut.

Ada beberapa jenis skala penilaian yang dapat digunakan pada saat penelitian, yaitu: skala *Likert*, skala Guttman dan skala *Rating*.

2.9.1 Skala *Likert*

Skala *Likert* adalah skala yang dapat dipergunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang mengenai suatu gejala atau fenomena pendidikan. Dalam skala *Likert* terdapat dua bentuk pernyataan yaitu pernyataan positif yang berfungsi untuk mengukur sikap positif, dan pernyataan *negative* yang berfungsi untuk mengukur sikap *negative* objek sikap (Babbie, 2012).

Skala yang digunakan pada skala *likert* yaitu antara 1 sampai 5 dimana tiap nilai diartikan sebagai berikut:

1. Sangat tidak setuju
2. Tidak setuju
3. Netral
4. Setuju
5. Sangat setuju

2.9.2 Skala Guttman

Skala Guttman merupakan skala yang menginginkan tipe jawaban tegas, seperti jawaban benar-salah, ya-tidak, pernah-tidak pernah, positif-*negative*, tinggi-rendah, baik-buruk, dan seterusnya. Pada skala Guttman, hanya ada dua interval, yaitu setuju dan tidak setuju (Babbie,2012).

Skala Guttman dapat dibuat dalam bentuk pilihan ganda maupun daftar checklist. Untuk jawaban positif seperti benar, ya, tinggi, baik, dan sebagainya diberi skor 1; sedangkan untuk jawaban *negative* seperti salah, tidak, rendah, buruk, dan sebagainya diberi skor 0.

2.9.3 Rating Scale

Rating scale merupakan skala penilaian yang lebih fleksibel, skala penilaian ini tidak hanya untuk mengukur sikap tetapi dapat juga digunakan untuk mengukur persepsi responden terhadap fenomena lingkungan, seperti skala untuk mengukur status sosial, ekonomi, pengetahuan, kemampuan, dan lain-lain.

Pada skala penilaian *rating scale* menurut (Wimmer dkk,2006) keputusan peneliti yang menentukan skala mana yang digunakan untuk penilaian yang digunakan, apakah 1 sampai 3, 1 sampai 5, 1 sampai 7, 1 sampai 10 atau 1 sampai 100. Memilih jenis skala *rating* sebagian besar merupakan masalah preferensi pribadi. Namun ada beberapa hal yang harus dipertimbangkan, yaitu:

1. Skala dengan nilai yang lebih tinggi akan menggambarkan perbedaan penilaian *rating* yang lebih detail pada *item* jikadibandingkan dengan skala dengan penilain rendah. Sebagai contoh ketika kita ingin menilai pentingnya suatu program dalam acara *weekday radio show*, jika responden mengatakan “semakin tinggi penilai maka semakin penting program tersebut untuk mereka”. Apakah skala 1-3 atau 1-10 yang digunakan. Tentu saja 1-10 karena skala ini menggambarkan perbedaan paling lebar (Wimmer dkk,2006).
2. Penelitian yang dilakukan (Wimmer dkk,2006) menyatakan bahwa laki-laki, perempuan, disemua tingkatan usia, semua suku dan bangsa menyukai penilaian 1-10. Hal ini dikarenakan skala 1-10 digunakan secara

universal, terutama pada acara olahraga seperti pertandingan olimpiade. Hampir setiap orang mengerti skala 1-10 dimana 10 merupakan penilaian sempurna atau yang terbaik sedangkan 1 merupakan penilaian yang paling buruk. Penelitian (Wimmer dkk,2006) juga menunjukkan bahwa peneliti seharusnya tidak menggunakan skala *rating* 0-9 atau 1-9 karena responden umumnya tidak mengenali kalau 9 merupakan nilai tertinggi.

2.10 Sparsity

Sparsity merupakan suatu kondisi yang terjadi akibat terdapat banyak kekosongan data *rating* pada *item*, sehingga menyebabkan nilai *rating* pengguna terhadap *item* cenderung renggang. Hal ini disebabkan karena pengguna cenderung hanya memberikan nilai *rating*nya terhadap beberapa *item* saja. Masalah *sparsity* merupakan kendala yang ada pada sistem rekomendasi dengan metode *collaborative filtering* karena pada metode *collaborative filtering* merupakan hal yang paling penting dalam pemberian rekomendasi. Semakin banyak *rating* yang terkumpul maka semakin baik hasil rekomendasi yang diberikan. Semakin tinggi *sparsity* data *rating* maka akan memberikan *poor accuracy* (akurasi yang rendah) terhadap hasil rekomendasi (Xue,2009) dalam (Sanjung, 2011).

Jumlah *user* dan jumlah *item* yang besar akan sangat membantu proses perekomendasi, namun tidak mungkin semua *user* akan aktif menggunakan sistem dan memberikan penilaian pada setiap *item*. Terkadang *user* merasa malas untuk member *rating* pada *item* setelah memberi *rating* terhadap beberapa *item* saja. Pada umumnya *user* hanya melakukan *rating* terhadap *item-item* dalam jumlah yang lebih kecil dari jumlah *item* yang tersedia di *database*, sehingga menyebabkan matriks *user-item* mengalami kekosongan data yang sangat tinggi (Budianto,2012).

2.11 Pengujian Sistem Rekomendasi

Pengujian sistem digunakan untuk mengetahui seberapa bagus hasil rekomendasi yang diberikan sistem kepada pengguna. Beberapa cara yang dapat

dilakukan untuk mengevaluasi sistem rekomendasi yaitu dengan menghitung tingkat akurasi pemberian rekomendasi pada hasil rekomendasi dan juga menghitung nilai presisi dari hasil UAT (*User Acceptance Test*) yang diberikan *user*.

2.11.1 Presisi (*Precision*)

Pada kinerjanya, sistem rekomendasi akan memberikan sekumpulan item yang akan di rekomendasikan kepada pengguna. Namun dari kumpulan item tersebut ada beberapa item yang tidak sesuai dengan kebutuhan pengguna. Salah satu cara yang dapat digunakan untuk menghitung kualitas hasil rekomendasi adalah dengan menghitung nilai presisi (*precision*). Dengan rumus ini, dapat dibandingkan antara item yang sesuai dengan keinginan pengguna dan total keseluruhan hasil rekomendasi. Sehingga dapat diketahui seberapa besar tingkat kesesuaian hasil rekomendasi sistem dengan kebutuhan pengguna.

Persamaan *precision* (Laoli,2009):

$$Precision = \frac{\#(relevant\ items\ retrieved)}{\#(retrieved\ items)} = P(relevant|retrieved) \quad (2,6)$$

Relevant items retrieved merupakan total jumlah item yang sesuai (relevan) yang direkomendasikan. Sedangkan *retrived item* merupakan total keseluruhan item yang direkomendasikan oleh sistem rekomendasi.

2.12 Kajian Penelitian Terkait

Penelitian tentang sistem rekomendasi dengan metode *collaboratif filtering* telah banyak dikembangkan. Seperti penelitian tentang pembuatan website movie dengan *Sistem Recommender Berbasis Collaborative Filtering*. (Devi Dwi Purwanto : 2011). Pada penelitian ini, sistem rekomendasi dikembangkan dengan model *adjusted cosine similarity*. Dengan model ini, kesamaan karakteristik pengguna akan dihitung berdasarkan rating yang diberikan pengguna terhadap judul film. Berdasarkan penelitian ini, sistem rekomendasi mampu memberikan rekomendasi film kepada pengguna lainnya yang memiliki kesamaan hobi genre pengguna.

Sistem rekomendasi juga pernah digunakan dalam penelitian pada toko buku online untuk Rekomendasi Pembelian Buku secara Online, (Shofwatul, Uyun, dkk :2009). Metode yang digunakan dalam sistem rekomendasi ini adalah *item-based collaboratif filtering*. Proses pemberian rekomendasi kepada pengguna dilakukan dengan menghitung kedekatan antar satu item dengan item lainnya. Berdasarkan kesimpulan dari penelitian ini, Sistem rekomendasi pada toko buku online berhasil menampilkan rekomendasi buku kepada pelanggan sesuai dengan rating yang diberikan oleh pelanggan terhadap buku yang diminati pelanggan tersebut.

Penelitian terkait lainnya tentang sistem rekomendasi yang telah dilakukan adalah Perbandingan *Semantic Classification* dan *Cluster-based Smoothed* pada *Recommender System* berbasis *Collaborative filtering* (sanjung, aryani : 2011). *Semantic classification based-CF (SCF Algorithm)* merupakan algoritma untuk *CF system* yang dapat menangani masalah *sparsity* dan *scalability*. Jumlah user dan jumlah item sangatlah besar, sedangkan umumnya user hanya merating item-item dalam jumlah yang kecil sehingga menyebabkan user-item matrix memiliki tingkat *sparsity* yang tinggi. Sistem rekomendasi yang dibangun dengan algoritma *Cluster-based Smoothed* bisa memberikan rekomendasi yang maksimal. Kesimpulan dari penelitian ini adalah *Cluster-based Smoothed* tidak selalu memberikan nilai MAE lebih rendah dibandingkan *Semantic Classification* untuk seorang user active, hal ini dipengaruhi oleh kepadatan rating kelompok yang sesuai dengan item yang akan diprediksi.