BAB 2
TINJAUAN PUSTAKA

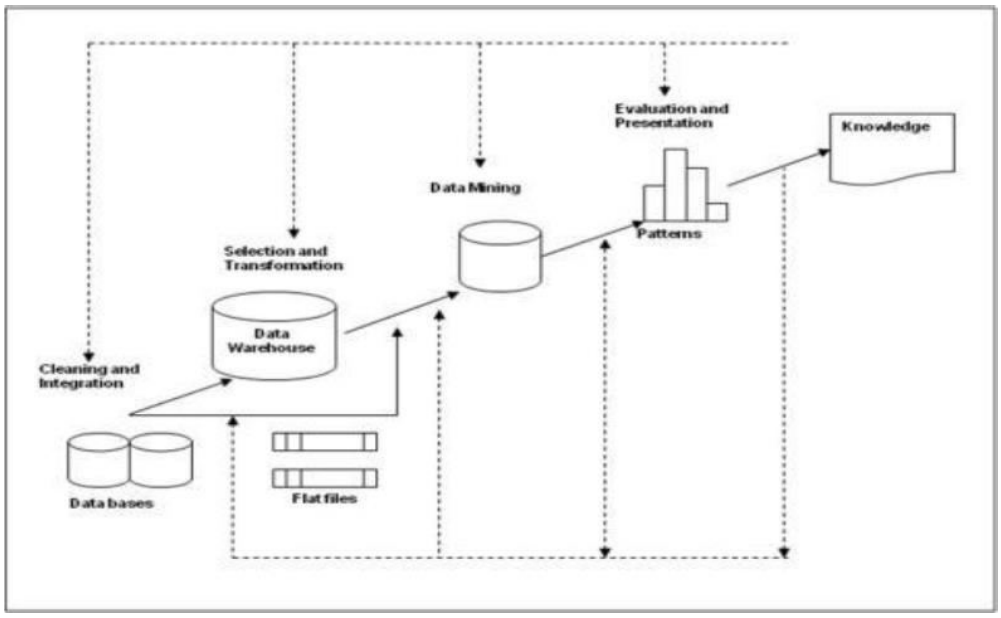
2.1 *Data mining*

Data mining merupakan proses untuk menggali nilai tambah, berupa informasi yang tidak diketahui dari suatu basis data (Ye, 2003). Proses yang dilakukan pada *data mining* adalah dengan melakukan penggalian pola-pola tertentu dari suatu basis data, yang bertujuan untuk memanipulasi data menjadi sebuah informasi yang lebih berharga, yang diperoleh dengan cara mengekstraksi dan mengenali pola yang penting atau menarik dari data yang terdapat dalam basis data.

Data mining juga dikenal dengan *Knowledge Discovery in Database* (KDD). Secara umum, definisi *data mining* dapat diartikan sebagai berikut (J. Han, 2006) :

1. Merupakan proses untuk menemukan pola yang menarik dari data yang tersimpan pada basis data dalam jumlah besar.
2. Ekstraksi menjadi suatu informasi yang berguna atau menarik pola yang sebelumnya tidak/ belum diketahui potensial kegunaannya atau pengetahan dari data yang di simpan pada basis data dalam jumlah besar.
3. Ekplorasi dari analisa secara otomatis atau semiotomatis terhadap data dalam jumlah besar untuk mencari suatu pola dan aturan yang memiliki suatu arti tertentu.

Data mining sangat diperlukan, terutama untuk mengelola data yang sangat besar yang dapat memudahkan aktifitas *recording* suatu transaksi dan untuk proses *data warehousing* agar dapat memberikan informasi yang akurat bagi pengguna. Alasan utama *data mining* dibutuhkan dalam industri informasi adalah karena tersedianya data dalam jumlah besar dan semakin besarnya kebutuhan untuk mengubah data tersebut menjadi informasi dan pengetahuan yang berguna. Berikut merupakan langkah-langkah dalam melakukan *data mining*



Gambar 2.1 Konsep *Data Mining*

Penjelasan dari konsep *data mining* yang dijelaskan pada Gambar 2.1 adalah sebagai berikut (J. Han, 2006):

a. *Data cleaning,* yaitu proses untuk menghilangkan *noise* data yang tidak koniten.
b. *Data integration,* yaitu proses penggabungan beberapa *file* atau *database.*
c. *Data selection,* yaitu data yang sudah relevan untuk tugas analisis akan dikembalikan ke *database* untuk proses *data mining*.
d. *Data transformation,* yaitu data berubah satu persatu menjadi bentuk yang tepat untuk menambang dengan ringkasan performa atau operasi agresi.
e. *Data mining,* yaitu metode yang intelijen yang digunakan untuk mengekstrak pola pada data.
f. *Knowledge discovery,* yaitu metode yang intelijen untuk mengekstrak pola pada data untuk mendapatkan pengetahuan.

g. *Pattern evolution,* yaitu untuk mengidentifikasi pola yang benar-benar menarik, yang mewakili pengetahuan berdasarkan atas beberapa tindakan yang menarik.
h. *Knowledge presentation,* yaitu gambaran teknik visualisasi dan pengetahuan yang digunakan untuk memberikan pengetahuan yang telah ditambahkan kepada pengguna.

2.1.1 Tahapan *Data mining*

Tahapan dalam melakukan *data mining,* salah satunya adalah melakukan *preprocessing data.* Tahapan ini diperlukan apabila data yang akan digunakan belum dalam keadaan baik, yang disebabkan oleh beberapa faktor, diantaranya (R.J.A Little, 2002):

1. *Incomplete,* yang terdiri dari kurang lengkapnya nilai suatu atribut, tidak lengkapnya atribut yang penting, atau hanya data yang merupakan rekapitulasi.
2. *Noisy,* yaitu *error* yang dikarenakan nilai yang tidak wajar.
3. *Inconsistent,* yaitu data yang mengandung nilai saling bertentangan.

Masalah ini muncul karena data yang didapatkan berasal dari berbagai sumber yang memungkinkan perbedaan persepsi pengolahan data. Selain itu, perbedaan ini muncul karena pelanggaran terhadap *functional dependency,* misalnya melakukan perubahan pada data yang terhubung dengan data yang lain.

Data yang dapat diterima agar dapat diproses menjadi sebuah informasi adalah data yang mempunyai kualitas yang akurat, lengkap, konsisten, relevan, dapat dipercaya, mempunyai nilai tambah, dan kemudahan untuk dapat dimengerti.

Jika data yang ada tidak dalam kualitas yang baik, maka kualitas analisis data menjadi kurang, sehingga menghasilkan informasi yang kurang bermakna. Hal tersebut harus dihindari, karena hasil yang diperoleh dapat berujung pada solusi yang salah. Untuk itu, perlu dilakukan *preprocessing data* yang bertujuan

untuk membuat data menjadi lebih berkualitas. Adapun tahapannya adalah sebagai berikut :

1. *Data cleaning,* mengisi/mengganti nilai-nilai yang hilang, menghaluskan data yang *noisy,* mengidentifikasi dan menghilangkan data yang tidak wajar, dan menyelesaikan masalah *inconsistent data.*
2. *Data integration,* menggabungkan beberapa *database* dan atau *file* menjadi satu sehingga didapatkan sumber data yang besar.
3. *Data transformation,* melakukan normalisasi dan agregasi data.
4. *Data reduction,* mengurangi volume data, namun tetap mempertahankan arti dalam hal hasil analisis data.
5. *Data discretization,* merupkan bagian dari data *reduction* dengan memperhitungkan data yang signifikan, khususnya pada data numerik.

2.1.2 *Assosiation Rule*

Association Rule merupakan salah satu metode yang digunakan untuk menemukan hubungan diantara data atau bagaimana suatu kelompok data mempengaruhi suatu keberadaan data lain. Aturan asosiasi merupakan salah satu metode yang sering digunakan untuk mencari hubungan antara berbagai *item*. Sebagai contoh, dari sebuah himpunan data transaksi mungkin akan ditemukan kejadian berikut, dimana seseorang yang membeli susu dalam transaksi yang sama akan membeli gula, atau seseorang yang membeli sikat gigi dalam yang transaksi yang sama juga akan membeli pasta gigi. Metode asosiasi dikenal sebagai salah satu metode *data mining* yang menjadi dasar dari berbagai metode *data mining* lainnya. Salah satu tahap dari analisis asosiasi yang disebut analisis pola frekuensi tinggi (*frequent pattern mining*) menarik perhatian banyak peneliti untuk menghasilkan algoritma yang efisien. Penting tidaknya suatu aturan assosiatif dapat diketahui dengan dua parameter, *support* (nilai penunjang) yaitu prosentase kombinasi *item* tersebut dalam *database* dan *confidence* (nilai kepastian) yaitu kuatnya hubungan antar *item* dalam aturan assosiatif. Analisis asosiasi didefinisikan suatu proses untuk menemukan semua aturan assosiatif yang

memenuhi syarat *minimum* untuk *support* (*minimum support*) dan syarat *minimum* untuk *confidence* (*minimum confidence*) (Ye, 2003).

Ada beberapa algoritma yang sudah dikembangkan mengenai aturan asosiasi, metode ini terbagi atas beberapa algoritma yaitu algoritma apriori, algoritma FP-Growth dan algoritma CT-Pro. Apriori merupakan algoritma klasik yang sering dipakai. Ide dasar dari algoritma ini adalah dengan mengembangkan *frequent itemset*, yaitu dengan cara menggunakan satu *item* dan kemudian secara rekursif mengembangkan *frequent itemset* tersebut dengan dua *item*, tiga *item* dan seterusnya hingga *frequent itemset* dengan semua ukuran. Untuk mengembangkan *frequent set* dengan dua *item*, dapat menggunakan frequent *set item*. Alasannya adalah bila *set* satu *item* tidak melebihi support minimum, maka sembarang ukuran *itemset* yang lebih besar tidak akan melebihi *support minimum* tersebut. Secara umum, mengembangkan *set* dengan *fc-item* menggunakan *frequent set* dengan k – 1 *item* yang dikembangkan dalam langkah sebelumnya. Setiap langkah memerlukan sekali pemeriksaan ke seluruh isi *database*.

2.1.3 Algoritma Apriori

Algoritma apriori merupakan salah satu algoritma yang digunakan untuk melakukan pencarian *frequent itemset* dengan menggunakan teknik *association rule*. Algoritma ini menggunaan pengetahuan *frekuensi* atribut yang telah diketahui sebelumnya, untuk melakukan penggalian informasi pada proses selanjutnya. Pada algoritma *apriori,* menentukan kandidat yang mungkin muncul dengan cara memperhatikan minimum *support* dan minimum *confidence.* Nilai *support* adalah nilai pengunjung atau persentase kombinasi *item* dalam *database* (Ye, 2003). Untuk mendapatkan nilai *support* untuk sebuah *item,* digunakan rumus sebagai berikut :

$$Support\ (A) = \left(\frac{Jumlah\ transaksi\ mengandung\ A}{Total\ Transaksi}\right) x\ 100\% \dots\dots\dots\dots (2.1)$$

Sedangkan untuk mencari nilai *support* untuk dua *item,* digunakan rumus sebagai berikut :

$$Support\ (A,B) = \left(\frac{Jumlah\ transaksi\ mengandung\ A\ dan\ B}{Total\ Transaksi}\right) x\ 100\% \ldots (2.2)$$

Sedangkan nilai *confidence* adalah nilai kepastian, yaitu kuatnya hubungan antar *item.* Nilai *confidence* dapat dicari setelah pola frekuensi munculnya sebuah *item* ditemukan. Rumus yang digunakan untuk mencari nilai *confidence* adalah sebagai berikut :

$$Confidence = \frac{Jumlah\ transaksi\ yang\ mengandung\ A\ dan\ B}{Jumlah\ transaksi\ yang\ mengandung\ A} \ldots\ldots\ldots (2.3)$$

Proses utama yang dilakukan dalam algoritma *apriori* untuk mendapatkan *frequent itemset* yaitu :

1. *Join* (Penggabungan)
   Proses ini dilakukan dengan cara mengkombinasikan suatu *item* dengan *item* yang lainnya hingga tidak dapat terbentuk kombinasi lagi.
2. *Prune* (Pemangkasan)
   Hasil dari *item* yang telah dikombinasikan, kemudian dipangkas dengan menggunakan *minimum support* yang telah ditentukan.

2.2 Sistem Rekomendasi

Sistem rekomendasi adalah *tools* perangkat lunak dan teknik yang memberikan saran untuk suatu *item* yang akan berguna bagi pengguna (Ricci, Rokach, Shapira, & Kantor, 2010). Saran tersebut berkaitan dengan berbagai proses pengambilan keputusan, seperti *item* apa yang harus dibeli, musik apa yang harus didengarkan, atau berita yang dibaca (Ricci, Rokach, Shapira, & Kantor, 2010). Kata "*Item*" adalah istilah umum yang digunakan untuk menunjukkan apa yang direkomendasikan sistem kepada pengguna. Sistem rekomendasi biasanya berfokus pada jenis *item* tertentu (contoh buku, film, atau berita) dan karena

disainnya, *user interface*, dan teknik rekomendasi utama yang digunakan untuk membuat rekomendasi (Ricci, Rokach, Shapira, & Kantor, 2010).

Sistem Rekomendasi memiliki beberapa pendekatan dasar yaitu (Ricci, Rokach, Shapira, & Kantor, 2010):

1. *Content-based*

   Pendekatan ini merekomendasikan *item* yang mirip dengan yang disukai pengguna di masa lalu. Kemiripan *item* dihitung berdasarkan fitur yang terkait dengan *item* yang dibandingkan. Misalnya, jika pengguna memberi penilaian positif pada film yang termasuk dalam genre komedi, maka sistem dapat belajar untuk merekomendasikan film lain dari genre yang sama (Ricci, Rokach, Shapira, & Kantor, 2010).

2. *Collaborative Filtering*

   Pendekatan ini merekomendasikan kepada pengguna aktif *item* yang disukai pengguna lain dengan selera serupa di masa lalu (Schafer, Frankowski, Herlocker, & Sen, 2007). Kesamaan selera dari dua pengguna dihitung berdasarkan kesamaan dalam *rating* pengguna terhadap *item* tersebut. Inilah alasan mengapa pendekatan ini mengacu pada "*user-to-user*" . *Collaborative Filtering* dianggap sebagai teknik yang paling populer dan diterapkan secara luas pada sistem rekomendasi.

3. *Hybrid-based*

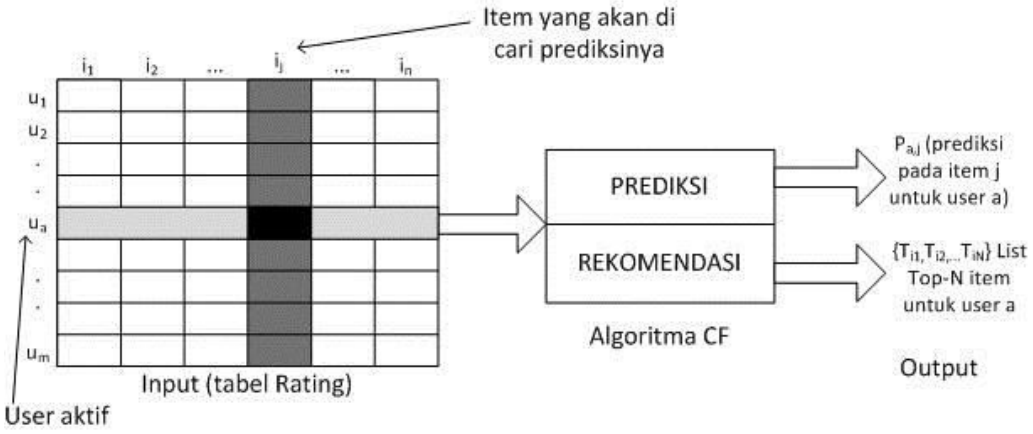
   Pendekatan ini didasarkan pada kombinasi teknik yang disebutkan pada poin 1 dan 2. Sistem *hybrid* yang menggabungkan teknik A dan B mencoba menggunakan keuntungan A untuk memperbaiki kelemahan B. Misalnya, metode *Collaborative Filtering* mengalami masalah *item* baru, yaitu, mereka tidak dapat merekomendasikan *item* yang tidak memiliki peringkat. Hal ini tidak membatasi, karena dapat digabungkan dengan pendekatan *content-based* karena prediksi *item* baru didasarkan pada deskripsi (fitur) mereka yang biasanya mudah tersedia. Dengan dua (atau

lebih) teknik dasar sistem rekomendasi, beberapa cara telah diusulkan untuk menggabungkannya untuk menciptakan sistem *hybrid*.

2.2.1 *Collaborative-Filtering*

Collaborative Filtering merupakan salah satu cara yang diterapkan sistem rekomendasi untuk memberikan prediksi otomatis terhadap keinginan pelanggan dengan cara mengoleksi informasi dari banyak pelanggan (Ampazis, 2008). Pada *Collaborative Filtering*, rekomendasi yang diberikan berdasarkan keterkaitan (preferensi) antara *item* satu dengan *item* lainnya dan keterkaitan yang didapat diantara *user* yang telah me-*rating* atau berinteraksi dalam sistem.

Collaborative Filtering memberikan rekomendasi berdasarkan kumpulan dari pendapat, minat dan ketertarikan beberapa *user* yang biasanya diberikan dalam bentuk *rating* yang diberikan *user* kepada suatu *item*. Pada metode *Collaborative Filtering*, terdapat kumpulan m *user* U = {u1,u2,u3,...,um} dan kumpulan n *item* I = {i1,i2,i3,...in} dimana setiap *user* (ui) mempunyai daftar *item* I ui yang merupakan ekpresi dari pendapatnya..



Gambar 2.2 Proses *Collaborative-filtering*

Pada Gambar 2.2 direpresentasikan sejumlah m *user* x n *item* sebagai sebuah matriks rating dimana berisi nilai *rating* dari *user* untuk setiap *item*. *User* aktif (Ua) pada skema tersebut adalah *user* yang akan diberikan rekomendasi *item*

yang mungkin disukainya. *Item* tersebut direpresentasikan dalam dua bentuk (Ampazis, 2008) yaitu :

1. Prediksi : merupakan nilai numerik, dimana Pa,j adalah nilai prediksi rating *item* yang mungkin disukai oleh Ua.
2. Rekomendasi : merupakan daftar N *item* yang mungkin disukai oleh Ua. Dengan catatan bahwa *item* yang direkomendasikan belum pernah dibeli atau di-*rating* oleh Ua. Hal ini sering juga disebut *top-N recommendation*.

Untuk memperoleh data *rating* dari *user* yang digunakan dalam sistem rekomendasi, dibedakan menjadi dua cara (Wibowo, Handojo, & Taliwang, 2010) yaitu:

1. Secara Eksplisit, yaitu proses pengumpulan data dimana *user* memberikan data secara sengaja contohnya : pemberian *rating*, pengisian *form* tentang data *user*, komentar *user* pada suatu *item*.
2. Secara Implisit, yaitu proses pengumpulan data dimana *user* tidak menyadari bahwa ia telah memberikan masukan terhadap sistem. Contohnya : catatan *item* yang dibeli atau dipinjam pada perpustakaan, *item* yang diklik, *item* yang dicari *user* dan lain lain.

Secara umum proses pemberian rekomendasi pada *Collaborative Filtering* terdiri atas 2 langkah (Sarwar, 2001) yaitu:

1. Penemuan nilai kemiripan,

   Nilai kemiripan atau *similarity* adalah menilai kemiripan barang dalam transaksi sehingga setelah didapatkannya daftar-daftar rekomendasi dari pendekatan yang akan digunakan maka nilai kemiripan ini berperan sebagai nilai urutan dari daftar rekomendasi tersebut. Adapun persamaannya sebagai berikut :

$$Adjust\ Cosine = S(i,j) = \frac{\sum_{u \in U} \left(Ru,i - \hat{R}u\right)(Ru,j - \hat{R}u)}{\sqrt{\sum_{u \in U} (Ru,i - \hat{R}u)^2} \sqrt{\sum_{u \in U} (Ru,j - \hat{R}u)^2}} \quad \ldots\ldots (2.5)$$

Keterangan :

S (i,j) = Nilai kemiripan antara *item* i dan *item* j

u ε U = Himpunan *user* yang melakukan *rating* baik *item* i dan *item* j

Ru,i = Rating *user* u pada *item* i

Ru,j = Rating *user* u pada *item* j

$\hat{R}u$ = Nilai rating rata-rata *user* u

$\hat{R}i$ = Nilai rating rata-rata *item* i

$\hat{R}j$ = Nilai rating rata-rata *item* j

2. Menghitung Prediksi *Rating*

Setelah mendapatkan daftar rekomendasi dan melakukan pengurutan yang bantu oleh nilai kemiripan maka tahapan selanjutnya yaitu menghitung prediksi *rating* yang didapat dari persamaan berikut :

$$P(a, j) = \frac{\sum_{i \epsilon I} (Ra,i \cdot Si,j)}{\sum_{i \epsilon I} |Si,j|} \quad \text{.....(2.6)}$$

Keterangan :

P(a,j) = Prediksi *rating item* j oleh *user* a

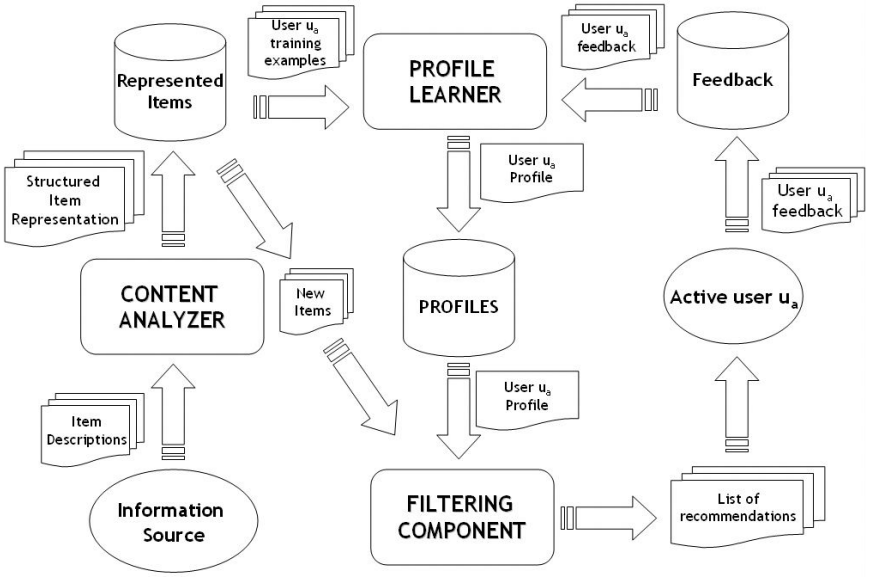
i ε I = Himpunan *item* yang mirip dengan *item* j

Ra,i = *Rating user* a pada *item* I

Si,j = Nilai *similarity* antara *item* i dan *item* j

2.2.2 *Content-based*

Content-based merupakan sistem rekomendasi yang cukup umum beberapa tahun terakhir dan sudah banyak diterapkan pada beberapa aplikasi terutama pada sistem penjualan *online*. Yang paling populer pendekatan ini diterapkan pada *film*, musik, berita, buku, artikel penelitian, *social tags*, dan produk lain pada secara umum. Beberapa penelitian pernah menerapkan pendekatan ini untuk melakukan pengangkatan angkatan bersenjata (Burke).



Gambar 2.3 Arsitektur *Content-based Recomender* Tingkat Tinggi

Sebagai studi kasus yaitu pada sistem rekomendasi yang terdapat pada rekomendasi *film* dengan beberapa atribut seperti *genre*, artis, penulis dan sutradara. Sehingga barang atau jasa yang di rekomendasikan berdasarkan yang sudah populer sebelumnya. Seperti apabila *genre* komedi sedang populer maka apabila ada *film* baru yang ber-genre komedi maka sistem rekomendasi akan menempatkan film tersebut pada daftar film yang direkomendasikan, walaupun *film* tersebut termasuk baru dalam data pendukung sistem rekomendasi.

Kekurangan dari pendekatan *Content-based* ini adalah (Pribadi, 2015) :

1. Hanya dapat digunakan untuk fitur yang sesuai, seperti gambar, film dan musik.
2. Barang / jasa yang di spesialisasi berlebihan, contoh pada *content-based* jarang sekali ditemukan rekomendasi yang kegunaannya berbeda padahal beberapa orang mungkin memiliki kepentingan yang berbeda. Misalkan seseorang yang baru saja membeli mobil sepertinya kurang pas apabila di berikan rekomendsasi barang yang serupa, namun akan cocok apabila barang yang direkomendasikan merupakan aksesoris dari mobil tersebut seperti (ban, lampu mobil, GPS dan akeseoris lain pelengkap mobil tersebut).
3. Tidak mampu menentukan profil dari *user* baru.

2.2.3 *Hybrid-Based*

Hybrid Recommender System yaitu metode yang menggabungkan dua atau lebih teknik untuk mendapatkan performa yang lebih baik dengan sedikit kelemahan dari setiap individu. Seringkali metode ini terdiri dari *collaborative-filtering* yang dikombinasikan dengan metode lain untuk menghindari *rump-up problem* (Burke). Berikut merupakan beberapa kombinasi yang pernah dilakukan sebelumnya.

1. *Weighted*

   Metode penggabungan antara dua atau lebih metode yang menghasilkan nilai prediksi pada masing-masing barang/ jasa yang direkomendasikan. Penelitian terkait yang pernah di lakukan oleh Huizhi Liang, Yue Xu, Yuefeng Li, Richi Nayak dan Gavin Shaw dengan judul *A Hybrid Recommender Systems based on Weighted Tags*. Pada penelitian tersebut mereka menggabungkan *Collaborative Filtering* dan *social tags* (Liang, Xu, Li, Nayak, & Shaw, 2011).

2. *Switching*

   Metode penggabungan antara dua atau lebih metode yang penggunaannya tergantung dari keadaan parameter. Metode ini dapat

dengan cerdas melakukan perpindahan antara pendekatan satu dengan pendekatan lain dengan berbagai criteria. Kelebihan dari pendekatan ini adalah merupakan metode yang *efficient* dalam penggunaan kekuatan dan kelemahan dari sistem rekomendasi konstitusional (Ghazanfar & Prugel Bennet, 2002).

3. *Mixed*

   Penggabungan metode *content-based* dan *collaborative-filtering* yang lebih menitik beratkan pada metode *content-based* dibanding dengan *collaborative-filtering*. Dimana *content-based* berperan sebagai pendekatan yang memberikan keunggulan akan suatu barang dan *collaborative-filtering* menambahkan apabila terdapat penilaian dari pengguna lain.

4. *Feature Combination.*

   Memungkinkan sistem rekomendasi mendapatkan hasil dengan keterkaitan yang tidak terlalu ekslusif sehingga beberapa barang / jasa yang berkaitan dapat saling melengkapi dan tidak esklusif, sebaliknya pendekatan *hybrid* dengan cara ini dapat menyebabkan buramnya *collaborative-filtering* yang digunakan.

2.3 Evaluasi Hasil Rekomendasi

Tahapan untuk menghitung atau mengukur akurasi antara nilai yang diprediksi dengan nilai yang sebenarnya, pada tahapan ini persamaan yang digunakan adalah *Mean Absolute Error (MAE)* detail persamaannya adalah sebagai berikut (Sarwar, 2001).

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=0}^{n} |\, \mathrm{pi} - \mathrm{fi} \,| \qquad \ldots\ldots\ldots\ldots\ldots\ldots (2.7)$$

Keterangan :

MAE = Nilai Rata-rata kesalahan hitung

n = Jumlah data yang dihitung

p i = Nilai Prediksi pada data ke-i

f I = Nilai sebenarnya pada data ke-i

2.4 Waikato Environment For Knowledge Analysis (WEKA)

WEKA dikembangkan oleh University of Waikato, New Zealand dibangun menggunakan Java, lisensi aplikasi ini yaitu Free software under The GNU General Public License. WEKA merupakan perangkat lunak *data mining* yang memiliki sekumpulan algoritma standar *data mining* WEKA dapat dijalankan berbasis GUI dan secara langsung melalui *Command line* (advanced users) WEKA dapat digunakan untuk melakukan *preprocessing*, klasifikasi, *clustering* (pengelompokan), regresi, *association rule mining* (ARM) dan visualisasi (www.cs.waikato.ac.nz, 2017).

2.5 Penelitian Terkait

Penelitian terkait yang pernah di lakukan oleh Huizhi Liang, Yue Xu, Yuefeng Li, Richi Nayak dan Gavin Shaw dengan judul *A Hybrid Recommender Systems based on Weighted Tags*. Pada penelitian tersebut mereka menggabungkan *Collaborative Filtering* dan *social tags** (Liang, Xu, Li, Nayak, & Shaw, 2011).

Penelitian dari Mustansar Ali Ghazanfar danAdam Prugel-Bennett pada tahun 2010. Penelitian dengan judul "*An Improved Switching Hybrid Recommender System Using Naive Bayes Classifier and Collaborative Filtering"*. Metode ini merupakan penggabungan antara dua atau lebih metode yang penggunaannya tergantung dari keadaan parameter. Metode ini dapat dengan cerdas melakukan perpindahan antara pendekatan satu dengan pendekatan lain dengan berbagai criteria. Kelebihan dari pendekatan ini adalah merupakan metode yang *efficient* dalam penggunaan kekuatan dan kelemahan dari sistem rekomendasi konstitusional (Ghazanfar & Prugel Bennet, 2002).