BAB II
LANDASAN TEORI

2.1 Karakteristik Sistem

Jogianto (2005: 3) mengemukakan sistem mempunyai karekteristik atau sifat-sifat tertentu, yakni :

1. Komponen

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu mempengaruhi proses sistem secara keseluruhan.

2. Batasan sistem

   Batasan sistem (boundary) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan suatu sistem menunjukan ruang lingkup dari sistem tersebut.

3. Lingkungan Luar Sistem.

   Lingkungan luar (*evinronment*) dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung Sistem

   Penghubung (interfance) merupakan media penghubung antara satu subsistem dengan subsistem yang lainya. melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainya membentuk satu kesatuan.

2.2 Pengertian *Data Mining*

Secara sederhana *data mining* adalah penambangan atau penemuan informasi baru dengan mencari pola atau aturan tertentu dari sejumlah data yang sangat besar (Davies, 2004). *Data mining* juga disebut sebagai serangkaian proses untuk menggali nilai tambah berupa pengetahuan yang selama ini tidak diketahui secara manual dari suatu kumpulan data (Pramudiono, 2007). *Data mining*, sering juga disebut sebagai *knowledge discovery in database* (KDD). KDD adalah kegiatan yang meliputi pengumpulan, pemakaian data, historis untuk menemukan keteraturan, pola atau hubungan dalam set data berukuran besar (Santoso, 2007).

Data mining adalah kegiatan menemukan pola yang menarik dari data dalam jumlah besar, data dapat disimpan dalam *database*, *data warehouse*, atau penyimpanan informasi lainnya. *Data mining* berkaitan dengan bidang ilmu – ilmu lain, seperti *database system*, *data warehousing*, statistik, *machine learning*, *information retrieval*, dan komputasi tingkat tinggi. Selain itu, *data mining* didukung oleh ilmu lain seperti *neural netwo*rk, pengenalan pola, *spatial data analysis*, *image database*, *signal processing* (Han, 2006). *Data mining* didefinisikan sebagai proses menemukan pola-pola dalam data. Proses ini otomatis atau seringnya semiotomatis. Pola yang ditemukan harus penuh arti dan pola tersebut memberikan keuntungan, biasanya keuntungan secara ekonomi. Data yang dibutuhkan dalam jumlah besar (Witten, 2005).

Karakteristik *data mining* sebagai berikut :

- *Data mining* berhubungan dengan penemuan sesuatu yang tersembunyi dan pola data tertentu yang tidak diketahui sebelumnya.
- *Data mining* biasa menggunakan data yang sangat besar. Biasanya data yang besar digunakan untuk membuat hasil lebih dipercaya.
- *Data mining* berguna untuk membuat keputusan yang kritis, terutama dalam strategi (Davies, 2004).

Berdasarkan beberapa pengertian tersebut dapat ditarik kesimpulan bahwa *data mining* adalah suatu teknik menggali informasi berharga yang terpendam

atau tersembunyi pada suatu koleksi data (*database*) yang sangat besar sehingga ditemukan suatu pola yang menarik yang sebelumnya tidak diketahui. Kata mining sendiri berarti usaha untuk mendapatkan sedikit barang berharga dari sejumlah besar material dasar. Karena itu *data mining* sebenarnya memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (*artificial intelligent*), *machine learning*, statistik dan *database*. Beberapa metode yang sering disebut-sebut dalam literatur *data mining* antara lain *clustering, classification, association rules mining, neural network, genetic algorithm* dan lain-lain (Pramudiono, 2007).

2.3 Tahap-Tahap *Data mining*

Tahap-tahap *data mining* ada 7 yaitu :

1. Pembersihan data (*data cleaning*)

   Pembersihan data merupakan proses menghilangkan noise dan data yang tidak konsisten atau data tidak relevan. Pada umumnya data yang diperoleh, baik dari *database* suatu perusahaan maupun hasil eksperimen, memiliki isian-isian yang tidak sempurna seperti data yang hilang, data yang tidak valid atau juga hanya sekedar salah ketik. Selain itu, ada juga atribut-atribut data yang tidak relevan dengan hipotesa *data mining* yang dimiliki. Data-data yang tidak relevan itu juga lebih baik dibuang. Pembersihan data juga akan mempengaruhi performasi dari teknik *data mining* karena data yang ditangani akan berkurang jumlah dan kompleksitasnya.

2. Integrasi data (*data integration*)

   Integrasi data merupakan penggabungan data dari berbagai *database* ke dalam satu *database* baru. Tidak jarang data yang diperlukan untuk *data mining* tidak hanya berasal dari satu *database* tetapi juga berasal dari beberapa *database* atau file teks. Integrasi data dilakukan pada atribut-aribut yang mengidentifikasikan entitas-entitas yang unik seperti atribut nama, jenis produk, nomor pelanggan dan lainnya. Integrasi data perlu dilakukan secara cermat karena kesalahan pada integrasi data bisa menghasilkan hasil yang menyimpang dan bahkan menyesatkan

pengambilan aksi nantinya. Sebagai contoh bila integrasi data berdasarkan jenis produk ternyata menggabungkan produk dari kategori yang berbeda maka akan didapatkan korelasi antar produk yang sebenarnya tidak ada.

3. Seleksi Data (*Data Selection*)

Data yang ada pada *database* sering kali tidak semuanya dipakai, oleh karena itu hanya data yang sesuai untuk dianalisis yang akan diambil dari *database*. Sebagai contoh, sebuah kasus yang meneliti faktor kecenderungan orang membeli dalam kasus *market basket analysis*, tidak perlu mengambil nama pelanggan, cukup dengan id pelanggan saja.

4. Transformasi data (*Data Transformation*)

Data diubah atau digabung ke dalam format yang sesuai untuk diproses dalam *data mining*. Beberapa metode *data mining* membutuhkan format data yang khusus sebelum bisa diaplikasikan. Sebagai contoh beberapa metode standar seperti analisis asosiasi dan *clustering* hanya bisa menerima input data kategorikal. Karenanya data berupa angka numerik yang berlanjut perlu dibagi-bagi menjadi beberapa interval. Proses ini sering disebut transformasi data.

5. Proses *mining*

Merupakan suatu proses utama saat metode diterapkan untuk menemukan pengetahuan berharga dan tersembunyi dari data.

6. Evaluasi pola (*pattern evaluation*)

Untuk mengidentifikasi pola-pola menarik kedalam *knowledge based* yang ditemukan. Dalam tahap ini hasil dari teknik *data mining* berupa pola-pola yang khas maupun model prediksi dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai. Bila ternyata hasil yang diperoleh tidak sesuai hipotesa ada beberapa alternatif yang dapat diambil seperti menjadikannya umpan balik untuk memperbaiki proses *data mining*, mencoba metode *data mining* lain yang lebih sesuai, atau menerima hasil ini sebagai suatu hasil yang di luar dugaan yang mungkin bermanfaat.

7. Presentasi pengetahuan (*knowledge presentation*),

Merupakan visualisasi dan penyajian pengetahuan mengenai metode yang digunakan untuk memperoleh pengetahuan yang diperoleh pengguna. Tahap terakhir dari proses *data mining* adalah bagaimana memformulasikan keputusan atau aksi dari hasil analisis yang didapat. Ada kalanya hal ini harus melibatkan orang-orang yang tidak memahami *data mining*. Karenanya presentasi hasil *data mining* dalam bentuk pengetahuan yang bisa dipahami semua orang adalah satu tahapan yang diperlukan dalam proses *data mining*. Dalam presentasi ini, visualisasi juga bisa membantu mengkomunikasikan hasil *data mining* (Han, 2006).

2.4 Fungsi *Data Mining*

Fungsi *data mining* dan macam-macam pola yang dapat ditemukan menurut Han dan Kamber (2006), yaitu:

1. *Concept/Class Description*: *Characterization and Discrimination*

   Data characterization adalah ringkasan dari semua karakteristik atau fitur dari data yang telah diperoleh dari target kelas. Data yang sesuai dengan kelas yang telah ditentukan oleh pengguna biasanya dikumpulkan di dalam *database*. Misalnya, untuk mempelajari karakteristik produk perangkat lunak dimana pada tahun lalu seluruh penjualan telah meningkat sebesar 10%, data yang terkait dengan produk-produk tersebut dapat dikumpulkan dengan menjalankan sebuah *query SQL*.

   Data discrimination adalah perbandingan antara fitur umum objek data target kelas dengan fitur umum objek dari satu atau satu set kelas lainnya. target diambil melalui *query database*. Misalnya, pengguna mungkin ingin membandingkan fitur umum dari produk perangkat lunak yang pada tahun lalu penjualannya meningkat sebesar 10% tetapi selama periode yang sama seluruh penjualan juga menurun setidaknya 30%.

2. *Classification and Prediction*

   Klasifikasi adalah proses untuk menemukan model atau fungsi yang menggambarkan dan membedakan kelas data atau konsep dengan tujuan memprediksikan kelas untuk data yang tidak diketahui kelasnya.

Model yang diturunkan didasarkan pada analisis dari training data (yaitu objek data yang memiliki label kelas yang diketahui). Model yang diturunkan dapat direpresentasikan dalam berbagai bentuk seperti *If-then* klasifikasi, *decision tree*, naïve bayes, dan sebagainya. Teknik *classification* bekerja dengan mengelompokkan data berdasarkan *data training* dan nilai atribut klasifikasi. Aturan pengelompokan tersebut akan digunakan untuk klasifikasi data baru ke dalam kelompok yang ada. Dalam banyak kasus, pengguna ingin memprediksikan nilai-nilai data yang tidak tersedia atau hilang (bukan label dari kelas). Dalam kasus ini nilai data yang akan diprediksi merupakan data *numeric*. Disamping itu, prediksi lebih menekankan pada identifikasi *trend* dari distribusi berdasarkan data yang tersedia.

3. *Cluster Analysis*

   Cluster adalah kumpulan objek data yang mirip satu sama lain dalam kelompok yang sama dan berbeda dengan objek data di kelompok lain. Sedangkan, *Clustering* atau Analisis *Custer* adalah proses pengelompokkan satu set benda-benda fisik atau abstrak kedalam kelas objek yang sama. Tujuannya adalah untuk menghasilkan pengelompokan objek yang mirip satu sama lain dalam kelompok-kelompok. Semakin besar kemiripan objek dalam suatu *cluster* dan semakin besar perbedaan tiap *cluster* maka kualitas analisis *cluster* semakin baik.

4. *Outlier analysis*

   Outlier merupakan objek data yang tidak mengikuti perilaku umum dari data. *Outlier* dianggap sebagai noise atau pengecualian. Analisis *data outlier* dapat dianggap sebagai *noise* atau pengecualian. Analisis *data outlier* dinamakan *Outlier Mining*. Teknik ini berguna dalam *fraud detection* dan *rare events analysis*.

5. *Evolution Analysis*

   Analisis evolusi data menjelaskan dan memodelkan *trend* dari objek yang memiliki perilaku yang berubah setiap waktu. Teknik ini dapat meliputi karakterisasi, diskriminasi, asosiasi, klasifikasi, atau *clustering* dari data yang berkaitan dengan waktu.

6. *Association rules*

   Association rules (aturan asosiasi) atau *affinity analysis* (analisis afinitas) berkenaan dengan studi tentang "apa bersama apa". Sebagai contoh dapat berupa berupa studi transaksi di supermarket, misalnya seseorang yang membeli susu bayi juga membeli sabun mandi. Pada kasus ini berarti susu bayi bersama dengan sabun mandi. Karena awalnya berasal dari studi tentang *database* transaksi pelanggan untuk menentukan kebiasaan suatu produk dibeli bersama produk apa, maka aturan asosiasi juga sering dinamakan *market basket analysis.*

   Aturan asosiasi ingin memberikan informasi tersebut dalam bentuk hubungan "if-then" atau "jika-maka". Aturan ini dihitung dari data yang sifatnya probabilistik (Santoso, 2007).

   Analisis asosiasi dikenal juga sebagai salah satu metode *data mining* yang menjadi dasar dari berbagai metode *data mining* lainnya. Khususnya 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*) (Pramudiono, 2007).

2.5 Analisis Asosiasi Apriori

Association rules (aturan asosiasi) atau *affinity analysis* (analisis afinitas) berkenaan dengan studi tentang "apa bersama apa". Sebagai contoh dapat berupa berupa studi transaksi di supermarket, misalnya seseorang yang membeli susu bayi juga membeli sabun mandi. Pada kasus ini berarti susu bayi bersama dengan sabun mandi. Karena awalnya berasal dari studi tentang *database* transaksi pelanggan untuk menentukan kebiasaan suatu produk dibeli bersama produk apa, maka aturan asosiasi juga sering dinamakan *market basket analysis.*

Aturan asosiasi ingin memberikan informasi tersebut dalam bentuk hubungan "if-then" atau "jika-maka". Aturan ini dihitung dari data yang sifatnya probabilistik (Santoso, 2007).

Analisis asosiasi dikenal juga sebagai salah satu metode *data mining* yang menjadi dasar dari berbagai metode *data mining* lainnya. Khususnya 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*) (Pramudiono, 2007).

Ada beberapa algoritma yang sudah dikembangkan mengenai aturan asosiasi, namun ada satu algoritma klasik yang sering dipakai yaitu algoritma *apriori*. Ide dasar dari algoritma ini adalah dengan mengembangkan *frequent itemset*. Dengan menggunakan satu item dan secara rekursif mengembangkan *frequent itemset* 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*.

Dalam asosiasi terdapat istilah *antecedent* dan *consequent, antecedent* untuk mewakili bagian "jika" dan *consequent* untuk mewakili bagian "maka". Dalam analisis ini, *antecedent* dan *consequent* adalah sekelompok item yang tidak punya hubungan secara bersama (Santoso, 2007).

Dari jumlah besar aturan yang mungkin dikembangkan, perlu memiliki aturan-aturan yang cukup kuat tingkat ketergantungan antar item. Untuk mengukur kekuatan aturan asosiasi ini, digunakan ukuran *support* dan *confidence*.

$$C = \frac{\text{Sup (XUY)}}{\text{Sup(X)}} \times 100\% \quad \text{..... (2.1)}$$

Keterangan :

C = Confidence

(Sup (XUY)) = Jumlah gabungan nilai support X dan Y

Sup(X) = Jumlah nilai support X

Langkah pertama algoritma *apriori* adalah, *support* dari setiap item dihitung dengan men-scan *database*. Setelah *support* dari setiap item didapat, item yang memiliki *support* lebih besar dari *minimum support* dipilih sebagai pola frekuensi tinggi dengan panjang 1 atau sering disingkat 1-itemset. Singkatan k-itemset berarti satu set yang terdiri dari k item.

Iterasi kedua menghasilkan 2-itemset yang tiap set-nya memiliki dua item. Pertama dibuat kandidat 2-itemset dari kombinasi semua 1-itemset. Lalu untuk tiap kandidat 2-itemset ini dihitung *support*-nya dengan men-scan *database*. *Support* artinya jumlah transaksi dalam *database* yang mengandung kedua item dalam kandidat 2-itemset. Setelah *support* dari semua kandidat 2-itemset didapatkan, kandidat 2-itemset yang memenuhi syarat *minimum support* dapat ditetapkan sebagai 2-itemset yang juga merupakan pola frekuensi tinggi dengan panjang 2. (Pramudiono, 2007)

Untuk selanjutnya iterasi iterasi ke-k dapat dibagi lagi menjadi beberapa bagian :

1. **Pembentukan kandidat itemset**

Kandidat k-itemset dibentuk dari kombinasi (k-1)-itemset yang didapat dari iterasi sebelumnya. Satu ciri dari algoritma *apriori* adalah adanya pemangkasan kandidat k-itemset yang subset-nya yang berisi k-1 item tidak termasuk dalam pola frekuensi tinggi dengan panjang k-1.

2. **Penghitungan support dari tiap kandidat k-itemset**

Support dari tiap kandidat k-itemset didapat dengan men-scan *database* untuk menghitung jumlah transaksi yang memuat semua item di dalam kandidat k-itemset tersebut. Ini adalah juga ciri dari algoritma *apriori* yaitu diperlukan penghitungan dengan scan seluruh *database* sebanyak k-itemset terpanjang.

3. **Tetapkan pola frekuensi tinggi**

Pola frekuensi tinggi yang memuat k item atau k-itemset ditetapkan dari kandidat k-itemset yang support-nya lebih besar dari *minimum support*. Kemudian dihitung *confidence* masing-masing kombinasi item. Iterasi berhenti ketika semua item telah dihitung sampai tidak ada kombinasi item lagi. (Pramudiono, 2007)

Selain algoritma *apriori*, terdapat juga algoritma lain seperti *FP-Grwoth*. Perbedaan algoritma *apriori* dengan *FP-Growth* pada banyaknya *scan database.* Algoritma *apriori* melakukan *scan database* setiap kali iterasi sedangkan algoritma *FP-Growth* hanya melakukan sekali di awal (Bramer, 2007).

Algoritma Apriori adalah algoritma paling terkenal untuk menemukan pola frekuensi tinggi. Pola frekuensi tinggi adalah pola – pola item di dalam suatu database yang memiliki frekuensi atau support di atas ambang batas tertentu yang disebut dengan istilah minimum support. Pola frekuensi tinggi ini digunakan untuk menyusun aturan assosiatif dan juga beberapa teknik data mining lainnya. Beberapa riset yang telah dilakukan berkaitan dengan kasus asosiasi yang menggunakan metode apriori , antara lain :

Penelitian yang berjudul "*Aplikasi Data Mining Untuk Menampilkan Informasi Tingkat Kelulusan Mahasiswa (Studi Kasus di Fakultas MIPA Universitas Diponegoro)*" oleh Nuqson Masykur Huda. Adapun data yang

diambil dalam penelitian ini adalah data induk mahasiswa dan data kelulusan mahasiswa, dapat menghasilkan informasi tentang tingkat kelulusan dengan data induk mahasiswa melalui teknik *data mining*. Kategori tingkat kelulusan di ukur dari lama studi dan IPK. Algoritma yang digunakan adalah algoritma *apriori*, informasi yang ditampilkan berupa nilai *support* dan *confidence* dari masing-masing kategori tingkat kelulusan. Kesimpulan yang dapat diambil adalah untuk menampilkan informasi tingkat kelulusan. Informasi yang ditampilkan berupa nilai *support* dan *confidence* hubungan antara tingkat kelulusan dengan data induk mahasiswa. Semakin tinggi nilai *confidence* dan *support* maka semakin kuat nilai hubungan antar atribut. Data induk mahasiswa yang diproses *mining* meliputi data proses masuk, data asal sekolah, data kota mahasiswa, dan data program studi. Hasil dari proses *data mining* ini dapat digunakan sebagai pertimbangan dalam mengambil keputusan lebih lanjut tentang faktor yang mempengaruhi tingkat kelulusan khususnya faktor dalam data induk mahasiswa.

Selain itu, penelitian lain dilakukan oleh Nurul Wardani., Tujuan penelitian ini adalah mendeskripsikan prosedur pelaksanaan pemberian kredit usaha rakyat (KUR) pada Bank Rakyat Indonesia Unit Kuwarasan Cabang Gombong dan permasalahan hukum yang timbul dalam pelaksanaan pemberian kredit usaha rakyat ini serta tindakan dari Bank Rakyat Indonesia Unit Kuwarasan Cabang Gombong dalam mengatasinya. Penelitian ini merupakan penelitian empiris bersifat deskriptif. Jenis data yang digunakan adalah data primer dan data sekunder. Teknik pengumpulan data yang dipergunakan yaitu melalui wawancara, dan studi kepustakaan. Teknik analisis data secara kualitatif dengan analisis model interaktif. Hasil penelitian menunjukkan bahwa pelaksanaan pemberian kredit usaha rakyat pada BRI Unit Kuwarasan Cabang Gombong melalui beberapa tahapan yaitu tahap permohonan, tahap pemeriksaan atau analisis kredit, pemberian putusan, dan tahap akad kredit/ pencairan kredit. Permasalahan hukum yang timbul atas pemberian kredit usaha rakyat adalah adanya kredit bermasalah serta ketidakseimbangan hak dan kewajiban antara pihak debitur dengan kreditur. Upaya atau tindakan yang dilakukan BRI Unit Kuwarasan Cabang Gombong adalah penagihan secara terus menerus kepada debitur serta memperketat analisis

kredit. Dalam hal kredit macet maka upaya yang dilakukan BRI Unit Kuwarasan Cabang Gombong adalah pengajuan klaim ke Askrindo sesuai dengan nota kesepahaman yang telah disepakati oleh Pemerintah, Perusahaan Penjamin, serta bank pelaksana karena kredit usaha rakyat ini merupakan program Pemerintah sebagai alternatif sumber pembiayaan UMKM untuk mengurangi tingkat kemiskinan di Indonesia.

Penelitian lain dilakukan oleh Muchammad Iljas mengenai *"Rancang Bangun Perangkat Lunak Analisa Keranjang pasar Dengan Metode Apriori"*. Dalam penelitiannya, Analisa keranjang pasar dengan metode Apriori sebagai salah satu teknik data mining dapat digunakan untuk menggali pola kecenderungan kemunculan barang secara bersamaan yang dilakukan oleh para pembeli. Dengan data transaksi yang besar maka bahasa SQL (Structured Query Language) dapat digunakan sebagai solusi yang tepat untuk meningkatkan kecepatan proses pembentukan frequent itemset, pembentukan 3 itemset dari 2 itemset, perhitungan nilai support dan confidence serta dapat melakukan pembentukan kaidah asosiasi dari 2-itemset dan 3-itemset. Berdasarkan hasil grafik analisa data transaksi order detail pada database northwide SQL Server yang telah diubah ke database MySQL, nilai support tertinggi yang dapat digunakan untuk membentuk 2-itemset sebesar 1% (satu persen) dengan kemunculan sebanyak 8 transaksi berbanding terhadap seluruh transaksi yang berjumlah 830 transaksi.

2.6 Contoh Algoritma Apriori untuk Pencarian Association Rule

Misalkan :

TID	Itemset
1	A.html, C.html, D.html
2	B.html, C.html, E.html
3	A.html, B.html, C.html, E.html
4	B.html, E.html

Misalkan diinginkan minsup : 50% (2 dari 4 transaksi)

Langkah 1:

L1={large 1-itemset}

Itemset	Support
A	50%
B	75%
C	75%
D	25%
E	75%

Langkah 2: Mencari kandidat itemset untuk L2:

2.1 : Gabungkan itemset pada L1 (algoritma apriori-gen)

{ A B, A C, A D, A E, B C, B D, B E, C D, C E, D E}

2.2 : Hapus yang tidak ada dalam itemset
Itemset { B D, DE} dihapus karena tidak ada dalam itemset

Langkah 3 :
Hitung support dari setiap kandidat itemset

Itemset	Support
A B	25 %
A C	50 %
A D	25 %
A E	25%
B C	50%
B E	75%
C D	25%
C E	50%

Langkah 4:
L2 { large 2-itemset

Itemset	Support
A C	50 %
B C	50%
B E	75%
C E	50%

Langkah 5 : Ulangi langkah 2-4

5.1 : Gabungkan itemset pada L2 & L2:

Itemset	Hasil Gabungan (3 itemset)
A C + B C	A C B
A C + B E	A C B, A C E, A B E
A C + C E	A C E
B C + B E	B C E
B C + C E	B C E
B E + C E	B C E

5. 2 : Hapus yang tidak ada dalam itemset : { A C E }

Langkah 6 : Hitung support dari setiap kandidat itemset L3

Itemset	Support
A B C	25 %
A B E	25 %
B C E	50 %

Langkah 7 : L3 { large 3-itemset } { B C E}

Langkah 8 : STOP karena sudah tidak ada lagi kandidat untuk 4-itemset.

Dari hasil – hasil diatas hasil akhir sebagai berikut:

L1

A	50%
B	75%
C	75%
D	25%
E	75%

L2

A C	50%
B C	50%
B E	75%
C E	50%

L3

B C E	50%

Untuk mencari aturan asosiasi diperlukan juga minconf

Misal minconf : 75 %, aturan asosiasi yang mungkin terbentuk:

Aturan (X → Y)	Sup(X ∪ Y)	Sup(X)	Confidence
B C → E	**50%**	**50%**	**100%**
B E → C	50%	75%	66.67%
C E → B	**50%**	**50%**	**100%**
A → C	**50%**	**50%**	**100 %**
C → A	50%	75%	66.67%
B → C	50%	75%	66.67%
C → B	50%	75%	66.67%
B → E	**75%**	**75%**	**100%**
E → B	**75%**	**75%**	**100%**
C → E	50%	75%	66.67%
E → C	50%	75%	66.67%