JURASIK (Jurnal Riset Sistem Informasi & Teknik Informatika) ISSN 2527-5771
Volume 1, Nomor 1, Juli 2016

OPTIMASI PEMROGRAMAN QUERY UNTUK ALGORITMA APRIORI BERBASIS ASOSIASI DATA MINING

Femi Dwi Astuti[1], Widyastuti Andriyani[2]

[1,2] Teknik Informatika, STMIK AKAKOM Yogyakarta
E-mail : femi@akakom.ac.id, widya@akakom.ac.id

Abstrak

Secara umum, setiap pengukuran diasosiasikan dengan sebuah nilai threshold yang nilainya dapat ditentukan sendiri oleh pengguna data mining. Aturan asosiasi yang tidak menggunakan threshold cenderung tidak menarik karena tidak merepresentasikan pengetahuan kepada pengguna data mining. Salah satu faktor yang memberikan kontribusi untuk menentukan apakah suatu pola menarik atau tidak adalah kesederhanaannya dalam pemahaman manusia. Semakin kompleks struktur sebuah aturan, maka semakin sulit untuk diinterpretasikan sehingga pola yang dibentuk semakin tidak menarik.

Kata kunci : *Data mining; aturan asosiasi; algoritma apriori*

Pendahuluan

Aturan asosiasi dalam data mining bertujuan untuk menemukan pola frekuensi, asosiasi, korelasi dan struktur hubungan antar item atau obyek dari sekumpulan item yang ada dalam transaksi database, database relasional maupun informasi dari tempat penyimpanan yang lain.Setiap item atau obyek adalah variabel bernilai Boolean yang merepresentasikan ada tidaknya item tersebut dalam transaksi. Setiap transaksi dapat direpresentasikan oleh nilai vektor Boolean untuk setiap variabelnya. Vektor Boolean dapat dianalisis untuk menemukan pola dari item-item yang cenderung muncul bersamaan dengan nilai frekuensi.

Sebagai contoh aturan asosiasi adalah informasi tentang pelanggan yang membeli komputer juga membeli perangkat lunak keuangan, dapat ditulis dalam aturan asosiasi sebagai berikut:
Komputer = perangkat lunak keuangan (support=2,confidence=60)

Nilai support dan confidence pada aturan tersebut di atas adalah dua pengukuran untuk pola kecenderungan dalam transaksi tersebut. Setiap pengukuran yang obyektif untuk pola yang menarik didasarkan pada struktur dari pola itu sendiri dan nilai statistik yang mendasarinya. Secara umum, setiap pengukuran diasosiasikan dengan sebuah nilai threshold yang nilainya dapat ditentukan sendiri oleh pengguna data mining. Aturan asosiasi yang tidak menggunakan threshold cenderung tidak menarik karena tidak merepresentasikan pengetahuan kepada pengguna data mining.

Salah satu faktor yang memberikan konstribusi untuk menentukan apakah suatu pola menarik atau tidak adalah kesederhanaannya dalam pemahaman manusia. Semakin kompleks struktur sebuah aturan, maka semakin sulit untuk diinterpretasikan sehingga pola yang dibentuk semakin

tidak menarik. Setiap pola yang disajikan harus mempunyai sebuah nilai pengukuran yang memberikan kepastian bahwa pola tersebut layak dipercaya sebagai pola yang terbaik atau tidak.

Pengukuran untuk kepastian dalam aturan asosiasi dirumuskan secara singkat dalam bentuk "A&B", dimana A dan B adalah transaksi yang akan dianalisis. Rumus aturan asosiasi dengan tingkat kepastian disebut sebagai confidence. Confidence dari transaksi "A&B" dari data semua transaksi yang ada dalam database dapat didefinisikan sebagai berikut:

$$\text{confidence(AB)} = \frac{\text{jumlah transaksi dari A dan B}}{\text{jumlah transaksi dari A}}$$

Aturan asosiasi untuk pembelian komputer dan perangkat lunak keuangan dengan confidence sebesar 60% menunjukkan bahwa 60% dari pelanggan yang membeli komputer juga membeli perangkat lunak keuangan. Salah satu kegunaan dari sebuah pola adalah faktor yang menentukan pola tersebut menarik atau tidak. Hal tersebut dapat diperkirakan dengan menggunakan fungsi tertentu yang disebut sebagai support. Support dari pola asosiasi menunjukkan besarnya persentase data yang ada dalam transaksi yang dianalisis, sehingga persentase tersebut dapat menunjukkan bahwa pola tersebut benar. Support dari aturan asosiasi transaksi "AÞB" dari data semua transaksi yang ada dalam database dapat didefinisikan sebagai berikut:

$$\text{support(A=B)} = \frac{\text{Jumlah transaksi dari A dan B}}{\text{jumlah transaksi keseluruhan}}$$

Aturan asosiasi untuk pembelian komputer dan perangkat lunak keuangan dengan support sebesar 2% menunjukkan bahwa 2% dari keseluruhan transaksi adalah transaksi pembelian komputer dan perangkat lunak keuangan yang selalu dibeli secara bersamaan. Banyak teknik pemrograman yang digunakan dalam penyelesaian aturan asosiasi. Umumnya penyelesaian menggunakan teknik pemrograman dengan pendekatan iteratif yaitu dengan memanfaatkan perulangan atau looping. Teknik iteratif memiliki berbagai kekurangan dari sisi kinerja perangkat keras dan perangkat lunak yang terbebani dengan jumlah data berukuran besar.

Dengan adanya berbagai perkembangan teknologi di bidang database server, maka penelitian ini ingin menggali lebih lanjut teknik selain iteratif yang dapat diimplementasikan dalam penyelesaian masalah asosiasi data. Tentunya hasil akhir yang ingin dicapai dari penelitian ini adalah efektifitas teknik pemrograman dan juga efisiensi waktu penyelesaian analisis.

Kajian Teori

1. Rakesh Agrawal dkk (Agrawal, R., Manilla, H.,Srikant, R., Toivonen, H., & Verkamo, I., 1996) yang membahas tentang cara kerja aturan asosiasi menggunakan algoritma Apriori untuk menemukan itemset terbesar dengan menggunakan data sintetik.
2. Fayyad, U.M., Piatetsky – Shapiro, G., & Smyth, P., 1996) menjelaskan tentang Knowledge Discovery in Database (KDD) yang disebut data mining, bagaimana proses KDD, metode data mining, apa saja komponen – komponen data mining, algoritma, dan penerapannya dalam industri dan ilmu analisa data.
3. Yeong – Chyi Lee dkk (Yeong – Chyi Lee, Tzung – Pei Hong, & Wen – Yang Lin, 2005) yang membahas teknik data mining menggunakan maximum constraint.

4. N. Badal, dan Shruti Tripathi (N. Badal, & Shruti Tripathi, 2010) dalam penelitianya menjelaskan tentang proses data mining menggunakan algoritma VS_Apriori sebagai pengembangan algoritma Apriori. Sedangkan penelitian yang dilakukan oleh Pratima Gautam (Pratima Gautam dan K. R., Pardasani, 2011) membahas tentang usulan dalam proses data mining menggunakan aturan asosiasi bermacam level dalam waktu yang bersamaan.
5. Sanjeev Rao (Sanjeev Rao, dan Priyanka Gupta, 2012) tentang pengembangan algoritma Apriori pada aturan asosiasi yaitu FP-Growth Algorithm karena membutuhkan iterasi dan waktu yang lebih singkat.
6. Bala Sundar V., T Devi dan N Saravanan (Bala Sundar V, T Devi, & N Saravanan, 2012) tentang bagaimana menerapkan algoritma K-Means pada pengklusteran data mining analisa tentang Predicting Heart Disease dan mendesain metode apa yang lebih efektif.
7. Othman Yahya (Othman Yahya, Osman Hegazy, dan Ehab Ezat, 2012) yaitu implementasi dari algoritma Apriori berbasiskan HADOOP-MAPREDUCE MODEL.
8. Jogi Suresh, dan T. Ramanjaneyulu (Jogi Suresh, T. Ramanjaneyulu, 2013) tentang pencarian pola frekuensi itemset pada data mining menggunakan algoritma Apriori.
9. Shruti Aggarwal (Shruti Aggarwal, dan Ranveer Kaur, 2013) tentang studi perbandingan antara algoritma Apriori dengan algoritma versi lain seperti AIS Algorithm, DHP, dan Partition Algorithm.
10. Paresh Tanna (Paresh Tanna dan Yogesh Ghodasara, 2013) tentang dasar-dasar implementasi pola frekuensi tinggi pada algoritma data mining.
11. Abdullah Saad Almalaise Alghamdi (Abdullah Saad Almalaise Alghamdi, 2011) membahas bagaimana implementasi algoritma FP Growth pada data mining data medis.
12. Rajneech Kumar Singh et al (Rajneech Kumar Singh, Manoj Kumar Pandey, dan Jawed Ahmed, 2013) yang menjelaskan pemakaian pendekatan Disconnected Approach pada algoritma Apriori.

Metode Penelitian

Metode penelitian yang digunakan pada penelitian ini adalah:

1. Melakukan studi kepustakaan yang berkaitan dengan permasalahan ini
   Metode ini digunakan untuk mencari literatur atau sumber pustaka yang berkaitan dengan perangkat lunak yang dibuat dan membantu mempertegas teori-teori yang ada serta memperoleh data yang sesungguhnya.
2. Mengumpulkan data-data yang dibutuhkan
   Metode ini digunakan untuk mengumpulkan data dan informasi yang diperlukan dari sumber-sumber yang sebagian besar adalah materi sejenis dokumen yang berkenaan dengan masalah yang diteliti.
3. Metode Pembangunan Perangkat Lunak
   Metode ini terdiri dari :
   a) Analisis Kebutuhan Perangkat Lunak
      Dilakukan dengan menganalisis data dan informasi yang diperoleh sehingga dapat dijadikan bahan pengembangan perangkat lunak.
   b) Perancangan Perangkat Lunak
      Dilakukan untuk mendapatkan deskripsi arsitektural perangkat lunak, deskripsi antarmuka,

deskripsi data, dan deskripsi prosedural.

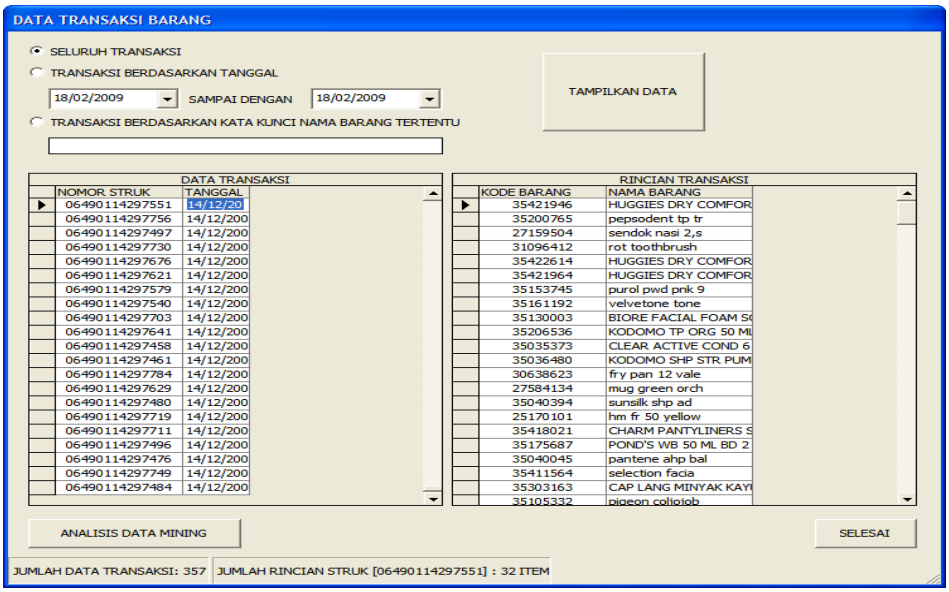
c) Implementasi Perangkat Lunak
Implementasi dilakukan dengan menerjemahkan deskripsi perancangan ke dalam bahasa pemrograman.

d) Pengujian Perangkat Lunak
Pengujian dilakukan untuk menguji fungsionalitas perangkat lunak.

Hasil dan Pembahasan

Implementasi dalam bentuk program terbagi menjadi 2 bagian yang terumuskan dalam bentuk form yaitu form untuk menampilkan informasi transaksi yang tercatat dalam database dan form untuk analisis untuk mencari pengelompokan barang. Form untuk mengimplementasikan sistem informasi pendataan dilengkapi dengan 3 pilihan untuk pengguna sistem yaitu pilihan menampilkan keseluruhan data transaksi dari rekaman data, pilihan untuk menampilkan data transaksi berdasarkan pilihan periode waktu dan pilihan berdasarkan kata kunci nama barang yang akan digunakan.

Oleh karena form ini menyediakan informasi tentang transaksi barang, maka perlu dilengkapi dengan nomor struk transaksi barang dan rincian dari setiap struk yang dipilih. Pada bagian lain dari form disediakan informasi yang cukup tentang jumlah keseluruhan transaksi dan rincian barang dari setiap transaksi.



Gambar.1 Form informasi transaksi

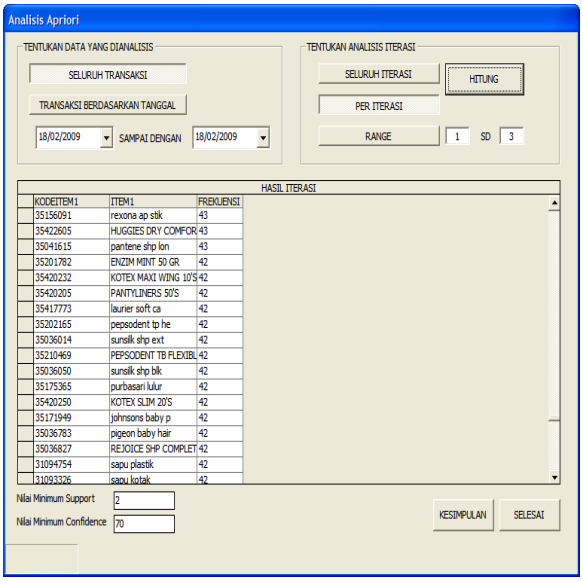
Form data transaksi barang dilengkapi dengan 3 pilihan dalam bentuk Option Button yaitu pilihan seluruh transaksi, transaksi berdasarkan tanggal dimana pengguna dapat memilih periode transaksi berdasarkan tanggal tertentu dan pilihan hanya untuk transaksi barang tertentu saja.

Setelah pengguna memilih pilihan, maka tombol atau button Tampilkan Data digunakan untuk melakukan pemrosesan terhadap pilihan yang diberikan. Data akan ditampilan secara terurut descending berdasarkan tanggal Di bawah ini adalah kode program dalam bentuk query untuk menampilkan data pada tombol Tampilkan Data.

Form untuk analisis data yang berkelompok dilengkapi dengan 2

pilihan atau option yaitu data keseluruhan yang dianalisis dan data berdasarkan periode tanggal tertentu saja yang dianalisis dengan pilihan tanggal yang dinamis sesuai dengan yang ditentukan oleh pengguna sistem.

Hasil analisis dapat dipilih oleh pengguna oleh karena sistem menyediakan 3 pilihan bentuk analisis yaitu analisis seluruh iterasi, analisis per iterasi dan analisis berdasarkan range iterasi yang diberikan. Secara mendasar, ketiga pilihan mempunyai kode program yang sama, yang membedakan adalah proses menampilkan hasil analisis. Pengguna dapat menampilkan secara kontinyu seluruh proses iterasi algoritma apriori atau proses ditampilkan secara per iterasi maupun hanya iterasi-iterasi tertentu saja. Hasil analisis ditampilkan dalam bentuk tabel hasil analisis, dimana tabel tersebut menampilkan data kombinasi setiap item barang dan jumlah atau frekuensi transaksi yang memuat kombinasi tersebut.



Gambar .2 Form analisis apriori

Berikut query untuk menghitung proses apriori untuk **iterasi pertama** ;

```
SELECT A.NoItem AS
KODEITEM1,
A.ItemBarang AS ITEM1
INTO Iterasi1
FROM tblBarang AS A
```

Sedangkan untuk perhitungan iterasi kedua, ketiga dan seterusnya menggunakan sistem join item barang. Kode program untuk proses join sebagaimana dipaparkan dalam bentuk listing program di bawah ini:

```
SELECT A.NoItem AS
KODEITEM1, A.ItemBarang AS
ITEM1,
B.NoItem AS KODEITEM2,
B.ItemBarang AS ITEM2
INTO Iterasi2
FROM tblBarang AS A, tblBarang
AS B
WHERE A.NoItem < B.NoItem
```

Setiap iterasi hasil proses join yaitu selalu meenghitung nilai frequent set atau frekuensi jumlah barang berkelompok. Di bawah ini adalah kode program untuk menghitung frequent set.

```
SELECT COUNT(JUMLAH) AS SC
FROM
(SELECT COUNT(NoStruk) AS
Jumlah
From tblTransaksi
WHERE NoItem IN
(@NoItemBarang)
GROUP BY NoStruk HAVING
COUNT(NoStruk) >=
@MinimumSupport)
```

Hasil perhitungan selalu dibandingkan dengan nilai minimum support yang diberikan oleh pengguna program atau melalui proses pruning. Proses pruning di awali dengan

perhitungan nilai Support Count (SC) untuk setiap kombinasi item barang sebagaimana yang dilakukan pada kode program di atas, nilai yang tidak memenuhi dengan nilai Minimum Support Count akan dihapus dari hitungan iterasi. Kode programnya adalah sebagai berikut:

```
DELETE FROM Iterasi WHERE SC
<= @MinimumSC
```

Untuk membandingkan hasil analisis menggunakan query dan pemrograman biasa (loop), maka setiap proses iterasi akan dilengkapi dengan pemrograman looping. Tujuannya adala untuk mebandingkan optimasi query bila dibandingkan dengan pemrograman konvensional tersebut dari sisi hasil dan waktu analisis. Berikut adalah kode looping yang diimplementasikan ke dalam pemrograman.

```
Ck: Candidate itemset of size k
Lk : frequent itemset of size k
L1 = {frequent items};
for (k = 1; Lk !=n; k++) do begin
Ck+1 = candidates generated from
Lk;
for each transaction t in database do
increment the count of all candidates
in
Ck+1 that are contained in t
Lk+1 = candidates in Ck+1 with
min_support
end
return k Lk;
```

```
SELECT NoStruk AS 'NOMORSTRUK', Tanggal AS 'TANGGAL'
FROM tblStruk
WHERE Tanggal BETWEEN
DATEVALUE('?') AND DATEVALUE('?')
ORDER BY Tanggal
```

Rincian ditampilkan dalam bentuk 2 lajur, lajur yang pertama berisi informasi tentang data transaksi barang dan waktu transaksi dan lajur yang lain digunakan untuk menampilkan rincian transaksi yang dipilih berdasarkan nomor transaksi dari lajur sebelumnya. Informasi pelengkap juga diberikan dalam bentuk Status Bar pada bagian bawah dari form yang berisi informasi tentang jumlah transaksi yang ditampilkan dan jumlah rincian dari setiap transaksi. Berikut adalah kode programnya:

```
SELECT tblBarang.NoItem AS
[KODE BARANG],
tblBarang.ItemBarang AS [NAMA
BARANG]
FROM tblStruk, tblTransaksi,
tblBarang
WHERE tblStruk.NoStruk =
tblTransaksi.NoStruk
AND tblTransaksi.NoItem =
tblBarang.NoItem
AND tblStruk.NoStruk =
@nomorstruk
```

Sistem

Hasil perhitungan dalam bentuk program diujicobakan dalam bentuk pengujian hasil untuk melihat hasil perhitungan algoritma apriori secara menyeluruh. Data yang dianalisis adalah data dengan sample 20 item barang dan 50 item barang yang diambil secara acak atau random berdasasarkan pertimbangan bahwa data-data tersebut dapat mewakili jumlah transaksi keseluruhan. Rekomendasi sistem minimal dalam menjalankan sistem ini, khususnya prosesor dan memori ditetapkan sesuai dengan standar

minimal Visual Basic sebagai perangkat dasar yang digunakan dalam mengembangkan sistem ini. Rekomendasi sistem minimal dari Visual Basic adalah Prosesor: 133 Mhertz dan jumlah Memori: 128 Mbyte. Sistem operasi yang digunakan adalah sistem operasi windows.

Perhitungan yang dilakukan melalui proses sistem, akan berbeda karena dipengaruhi oleh spesifikasi komputer yang digunakan, data yang dianalisis, dan banyaknya sampel yang digunakan dalam proses analisis data mining. Perhitungan di bawah ini menggunakan spesifikasi komputer berkecepatan 1 Gigahertz dengan besar ukuran memori 1 Gigabyte .

Analisis 20 item barang

Dari hasil analisis data menggunakan 20 item barang, waktu yang dibutuhkan untuk analisis iterasi 1, iterasi 2 dan iterasi 3 menggunakan algoritma apriori adalah masing-masing 45 detik, 65 detik dan 95 detik. Perhitungan tersebut menggunakan nilai minimum support count 2. Sedangkan perhitungan menggunakan minimum support count 8, menghasilkan masing-masing 45 detik, 52 detik dan 78 detik.

Jika perhitungan yang sama menggunakan proses looping, maka akan membutuhkan waktu analisis selama 52 detik, 95 detik dan 148 detik untuk minimum support count 2. Sedangkan menggunakan minimum support count 8, maka proses looping membutuhkan waktu 50 detik, 68 detik dan 95 detik.

Analisis 50 item barang

Sedangkan hasil analisis data menggunakan 50 item barang, waktu yang dibutuhkan untuk analisis iterasi 1, iterasi 2 dan iterasi 3 adalah masing-masing 62 detik, 256 detik dan 456 detik. Perhitungan menggunakan nilai minimum support count 6, Sedangkan perhitungan menggunakan minimum support count 8, menghasilkan masing-masing 59 detik, 213 detik dan 387 detik. Bila dibandingkan dengan analisis menggunakan proses looping, maka membutuhkan waktu analisis yang lebih lama. Masing-masing iterasi 1, 2 dan 3 membutuhkan waktu 98 detik, 432 detik dan 789 detik dengan nilai minimum support count 6. Dengan teknik yang sama untuk minimum support count 8, menghasilkan waktu 79 detik, 364 detik dan 578 detik untuk iterasi 1, 2 dan 3.

Analisis 20 Item Barang Menggunakan Nilai 2 Untuk Minimum Support Count
Iterasi 1:

Pada iterasi 1, program menampilkan semua item barang sejumlah 20 item dan jumlah frekuensi transaksi yang menggunakan nilai minimum support count sebesar 2 atau minimal berada di 2 transaksi. Waktu yang dibutuhkan adalah 36 detik untuk query dan 107 detik untuk proses looping.

Tabel 1. Frekuensi transaksi

KODEITEM1	ITEM1	FREKUENSI
25170094	hm fr 50 merah	55
27555684	basi pegangan 7	53
27481316	table top	52
22329491	clasio t kecil	50
27555639	sendok nasi	50
…… dst	…….. dst	……. dst
27555648	piring makan 9	45
27555693	basi tutup 10	45

27555764	mug phoenix 4.2	41
25443905	hm 60x120 anima	39
27228653	Fork	38
27555620	sendok soup 2.7	38

Iterasi 2:
Pada iterasi 2, program melakukan join antar 2 item barang dan proses perhitungan menghasilkan 190 kombinasi item barang yang memenuhi minimum support count sebesar 2. Waktu yang dibutuhkan untuk query yaitu 89 detik dan looping sebesar 579 detik. Terdapat perbedaan optimasi yang sangat besar.

Tabel 2. Join antar 2 item barang

KODEITEM1	ITEM1	KODEITEM2	ITEM2	FREK
25170094	hm fr 50 merah	25427343	hm 60x120 275 e	16
22329491	clasio t kecil	27159504	sendok nasi 2,s	16
22329491	clasio t kecil	25068866	mat towel mix	15
25068866	mat towel mix	25170094	hm fr 50 merah	15
25068866	mat towel mix	25170101	hm fr 50 yellow	15
	….			
	….			
25443905	hm 60x120 anima	27159504	sendok nasi 2,s	7

Iterasi 3:
Pada iterasi 3, program melakukan join antar 3 item barang dan proses perhitungan menghasilkan 784 kombinasi item barang yang memenuhi minimum support count sebesar 2. Waktu yang dibutuhkan oleh query sebesar 798 detik sedangkan menggunakan looping membutuhkan 8976 detik. Proses iterasi 4 dan seterusnya masih memungkinkan proses query untuk melakukan analisis, sedangkan proses looping menghabiskan proses memori yang sangat besar dan tidak bisa dilanjutkan.

Tabel 3. Join antar 3 item barang

KODE ITEM1	ITEM1	KODE ITEM2	ITEM2	KODE ITEM3	ITEM3	FREK
25068866	mat towel mix	25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	8
25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	27159504	sendok nasi 2,s	7
22329491	clasio t kecil	25170094	hm fr 50 merah	25427343	hm 60x120 275 e	6
25170094	hm fr 50 merah	25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	6
25068866	mat towel mix	25427343	hm 60x120 275 e	27159504	sendok nasi 2,s	6
25170094	hm fr 50 merah	25427343	hm 60x120 275 e	27159504	sendok nasi 2,s	6
22329491	clasio t kecil	25068866	mat towel mix	25170101	hm fr 50 yellow	5
25068866	mat towel mix	25170094	hm fr 50 merah	25427343	hm 60x120 275 e	5
22329491	clasio t kecil	25427343	hm 60x120 275 e	27159504	sendok nasi 2,s	5
22329491	clasio t kecil	25170094	hm fr 50 merah	25170101	hm fr 50 yellow	4
22329491	clasio t kecil	25068866	mat towel mix	25427343	hm 60x120 275 e	4
22329491	clasio t kecil	25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	4
25068866	mat towel mix	25170101	hm fr 50 yellow	25443905	hm 60x120 anima	4
25068866	mat towel mix	25427343	hm 60x120 275 e	25443905	hm 60x120 anima	4

25068866	mat towel mix	25170101	hm fr 50 yellow	27159504	sendok nasi 2,s	4
	…					

Analisis 50 Item Barang Menggunakan Nilai 6 Untuk Minimum Support Count Iterasi 1:

Pada iterasi 1, program menampilkan semua item barang sejumlah 50 item dan jumlah frekuensi transaksi yang menggunakan nilai minimum support count sebesar 6 atau minimal berada di 6 transaksi.

Tabel 4. Analisis 50 item barang

KODEITEM1	ITEM1	FREK
31603631	pail fluted lid	63
30638623	fry pan 12 vale	59
34031977	boneka st.bb	59
34708497	jaya plu amplop	57
25170094	hm fr 50 merah	55
27555835	piring oval 12	54
27555684	basi pegangan 7	53
27584134	mug green orch	53
27481316	table top	52
31096412	rot toothbrush	51
22329491	clasio t kecil	50
27555639	sendok nasi	50
34708406	dunia baru box	50
27598085	bowl with cover	49
27621281	pitcher aldio	49
31632369	sealware	49
25068866	mat towel mix	48
25427343	hm 60x120 275 e	48
27555755	mangkok sayur 8	48
27621423	gelas set 10 al	48
34019660	philips essenti	48
25170101	hm fr 50 yellow	47
27196750	cake fork bianc	47
27228617	spoon	46
27492322	glass helsinki	46
34000456	osram classic c	46
34695582	nagata ty	46
34706640	bantex ordner 7	46
27159504	sendok nasi 2,s	45
27492162	glass defice un	45
27555648	piring makan 9	45
27555693	basi tutup 10	45
34324606	ls boto air telp	45
35002345	gillette goal m	45
27621227	gelas set aldio	44
30552895	sp 16 orchid	44
31081900	saringan air	44
34441300	hw sikat wc	44
27620737	mangkok 8.5	43
31093326	sapu kotak	42

31094754	sapu plastik	42
27555764	mug phoenix 4.2	41
27597273	piring makan 9	40
35002603	good shp lemon	40
25443905	hm 60x120 anima	39
27555808	mangkok soup 7	39
34024636	md 40 11.2 fl 0	39
27228653	Fork	38
27555620	sendok soup 2.7	38
31095379	vitto round slw	37

Iterasi 2:
Pada iterasi 2, program melakukan join antar 2 item barang dan proses perhitungan menghasilkan 9857 kombinasi item barang yang memenuhi minimum support count sebesar 6.

Tabel 5. Join antar 2 item barang dan proses perhitungan menghasilkan 9857 kombinasi

KODE ITEM1	ITEM1	KODE ITEM2	ITEM2	FREK
25170094	hm fr 50 merah	25427343	hm 60x120 275 e	16
22329491	clasio t kecil	27159504	sendok nasi 2,s	16
34708497	jaya plu amplop	35002603	good shp lemon	16
22329491	clasio t kecil	25068866	mat towel mix	15
25068866	mat towel mix	25170094	hm fr 50 merah	15
25068866	mat towel mix	25170101	hm fr 50 yellow	15
25170094	hm fr 50 merah	25170101	hm fr 50 yellow	15
25068866	mat towel mix	25427343	hm 60x120 275 e	15
22329491	clasio t kecil	25427343	hm 60x120 275 e	14
25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	14
25170094	hm fr 50 merah	25443905	hm 60x120 anima	14
25427343	hm 60x120 275 e	25443905	hm 60x120 anima	14
25170094	hm fr 50 merah	27159504	sendok nasi 2,s	14
25068866	mat towel mix	25443905	hm 60x120 anima	13
22329491	clasio t kecil	25170094	hm fr 50 merah	12
25068866	mat towel mix	27159504	sendok nasi 2,s	12
34708406	dunia baru box	35002603	good shp lemon	12
22329491	clasio t kecil	25443905	hm 60x120 anima	11
25427343	hm 60x120 275 e	27159504	sendok nasi 2,s	11
	……..			
	……..			

Iterasi 3:
Pada iterasi 3, program melakukan join antar 3 item barang dan proses perhitungan menghasilkan 19.918 kombinasi item barang yang memenuhi minimum support count sebesar 6.

Tabel 6. Join antar 3 item barang dan proses perhitungan dihasilkan 19.918 kombinasi

KODE ITEM1	ITEM1	KODE ITEM2	ITEM2	KODE ITEM3	ITEM3	FREK
34708497	jaya plu amplop	35002345	gillette goal m	35002603	good shp lemon	9
22329491	clasio t kecil	25170094	hm fr 50 merah	25170101	hm fr 50 yellow	8
25068866	mat towel mix	25170094	hm fr 50 merah	25427343	hm 60x120 275 e	8
22329491	clasio t kecil	25427343	hm 60x120 275 e	25443905	hm 60x120 anima	8
25068866	mat towel mix	25427343	hm 60x120 275 e	25443905	hm 60x120 anima	8
25068866	mat towel mix	25170101	hm fr 50 yellow	27196750	cake fork bianc	8
25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	27196750	cake fork bianc	8
34708406	dunia baru box	35002345	gillette goal m	35002603	good shp lemon	8
22329491	clasio t kecil	25068866	mat towel mix	25170101	hm fr 50 yellow	7
25068866	mat towel mix	25170094	hm fr 50 merah	25170101	hm fr 50 yellow	7
22329491	clasio t kecil	25170094	hm fr 50 merah	25427343	hm 60x120 275 e	7
25068866	mat towel mix	25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	7
25068866	mat towel mix	25170101	hm fr 50 yellow	25443905	hm 60x120 anima	7
25170094	hm fr 50 merah	25170101	hm fr 50 yellow	25443905	hm 60x120 anima	7
25170094	hm fr 50 merah	25427343	hm 60x120 275 e	25443905	hm 60x120 anima	7
22329491	clasio t kecil	25068866	mat towel mix	27159504	sendok nasi 2,s	7
22329491	clasio t kecil	25170094	hm fr 50 merah	27159504	sendok nasi 2,s	7
25068866	mat towel mix	25170094	hm fr 50 merah	27159504	sendok nasi 2,s	7
22329491	clasio t kecil	25170101	hm fr 50 yellow	27159504	sendok nasi 2,s	7
25170094	hm fr 50 merah	25170101	hm fr 50 yellow	27159504	sendok nasi 2,s	7
25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	27159504	sendok nasi 2,s	7
25170094	hm fr 50 merah	25170101	hm fr 50 yellow	27196750	cake fork bianc	7
22329491	clasio t kecil	25427343	hm 60x120 275 e	27196750	cake fork bianc	7
25068866	mat towel mix	25427343	hm 60x120 275 e	27196750	cake fork bianc	7
25170101	hm fr 50 yellow	25443905	hm 60x120 anima	27196750	cake fork bianc	7
22329491	clasio t kecil	27159504	sendok nasi 2,s	27196750	cake fork bianc	7
25068866	mat towel mix	27159504	sendok nasi 2,s	27196750	cake fork bianc	7
25170101	hm fr 50 yellow	27159504	sendok nasi 2,s	27196750	cake fork bianc	7

25427343	hm 60x120 275 e	27159504	sendok nasi 2,s	27196750	cake fork bianc	7
22329491	clasio t kecil	25170101	hm fr 50 yellow	27228617	spoon	7
25068866	mat towel mix	25170101	hm fr 50 yellow	27228617	spoon	7
25170094	hm fr 50 merah	25170101	hm fr 50 yellow	27228617	spoon	7
25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	27228617	spoon	7
25170094	hm fr 50 merah	25443905	hm 60x120 anima	27228617	spoon	7
34706640	bantex ordner 7	35002345	gillette goal m	35002603	good shp lemon	7
22329491	clasio t kecil	25068866	mat towel mix	25170094	hm fr 50 merah	6
22329491	clasio t kecil	25068866	mat towel mix	25427343	hm 60x120 275 e	6
22329491	clasio t kecil	25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	6
25170094	hm fr 50 merah	25170101	hm fr 50 yellow	25427343	hm 60x120 275 e	6
22329491	clasio t kecil	25068866	mat towel mix	25443905	hm 60x120 anima	6
	….					
	….					

Dari hasil perhitungan yang disampaikan pada pembahasan di atas, maka algoritma apriori dapat menyelesaikan semua perhitungan sesuai dengan sample data yang diberikan. Proses perhitungan membutuhkan penggunaan sumber daya komputer yang cukup besar, baik kecepatan prosesor maupun jumlah memori yang lebih banyak.

Hasil perhitungan bervariasi, namun secara umum dapat dilihat bahwa:

1. Semakin banyak item barang yang dianalisis, maka waktu pemrosesan akan semakin lama baik oleh proses query maupun proses looping.
2. Semakin kecil nilai Minimum Support Count, maka proses perhitungan membutuhkan waktu yang lebih lama
3. Semakin kecil nilai Minimum Support Count dan semakin sedikit item barang yang dianalisis, akan berpengaruh pada proses perhitungan algoritma yang lebih cepat.

Proses query menggunakan optimasi yang sangat baik bila dibandingkan dengan proses looping. Proses looping cenderung hanya berjalan sampai proses iterasi 3 dengan 3 kombinasi item sedangkan proses iterasi berikut sudah tidak memungkinkan. Dengan proses query, maka optimasi setiap iterasi masih dimungkinkan walaupun dengan waktu yang cenderung lebih lama.

Dari hasil perhitungan algoritma juga dapat dianalisis bahwa proses yang membutuhkan sumber daya yang sangat besar dan waktu analisis yang lama adalah proses join antar item barang. Dengan proses kombinasi item barang yang semakin banyak menyebabkan kombinasi antar item juga semakin banyak. Hasil akhir menunjukkan perhitungan 50 item barang membutuhkan waktu yang lebih banyak dibandingkan perhitungan 20 item barang.

Kesimpulan

Setelah melalui tahap perancangan sistem dan implementasi diperoleh hasil optimasi pemrograman query, waktu yang digunakan menjadi lebih singkat karena semua data disimpan dalam bentuk tabel kemudian diolah dengan cara mengoptimalkan proses asosiasi antar item dan tingkat keakurasian hasil asosiasi menjadi lebih tinggi

Daftar Pustaka

[1] Berry, Michael J.A and Gordons S. Linnoff, 2004, Data Mining Techniques , Second Edition, Wiley Publishing, Inc., Indianapolis, Indiana
[2] Cross Industry Standard Process for Data Mining, http://www.crisp-dm.org/
[3] Han, Jiawei and Micheline Kamber, 2001, Data Mining : Concepts and Techniques, Morgan Kaufmann Publisher, London
[4] Inmon, W.H., 2002, Building the Data Warehouse, 3rd Edition, John Wiley & Sons, Inc., Canada
[5] Kantardzic, Mehmed, 2003, Data Mining-Concepts, Models, Methods, and Algorithms, New John Wiley & Sons, Inc., New Jersey
[6] Kelly, Sean, 1997, Data Warehouse in Action, John Wiley & Sons, Inc., New Jersey
[7] Kimball, Ralph and Margy Ross, 2002, The Data Warehouse Toolkit-the Complete Guide to Dimensional Modeling, Second Edition, John Wiley & Sons, Inc., Canada
[8] Mallach, Efrem G, 2000, Decision Support and Data Warehouse Systems, McGraw-Hill, International Edition
[9] Marakas, George M., 2003, Modern Data Warehousing, Mining, and Visualization - Core Concepts, Prentice Hall, Inc., New Jersey
[10] Poe, Vidette and Patricia Klauer., 1997, Building a Data Warehouse for Decision Support, Second Edition, Prentice Hall, Inc., New Jersey