

Metode *Market Basket Analysis* menggunakan Algoritma *Pincer Search* untuk Sistem Pembantu Pengambilan Keputusan

Gregorius S. Budhi, Leo W. Santoso, Edward Susanto

Jurusan Teknik Informatika, Fakultas Teknologi Industri Universitas Kristen Petra
Jl. Siwalankerto 121 - 131, Surabaya 60236
E-mail: greg@petra.ac.id, leow@petra.ac.id

Abstrak

*Dewasa ini muncul tuntutan bagi seorang pengambil keputusan untuk melihat peluang yang dapat meningkatkan keuntungan di perusahaannya, berdasarkan informasi yang relevan. Pengambil keputusan perlu memakai sistem yang dapat memberi pengetahuan (knowledge) untuk selanjutnya dipakai mengambil keputusan secara cepat dan tepat. Salah satu cara untuk mendapatkan knowledge adalah dengan metode *Market Basket Analysis*. Sistem ini akan menggunakan algoritma *Pincer Search* untuk melakukan analisa market basket pada data penjualan sebuah apotik pada periode tertentu. Data dipersiapkan melalui proses *cleaning* dan di transformasi ke bentuk yang dapat diolah sistem. Selanjutnya data diolah menggunakan algoritma *Pincer Search* sehingga menghasilkan *frequent itemset* dan pada akhirnya menghasilkan *association rules*. Rules ditampilkan dalam bentuk tabel dan grafik. Sistem dibangun menggunakan *Delphi 7* dan *Access 2000*. Dengan menggunakan output dari sistem ini, yang berupa tabel dan grafik *association rules*, pengambil keputusan dapat mengetahui barang-barang apa saja yang sering dibeli bersamaan di apotek tersebut. Informasi ini akan memberikan pertimbangan tambahan bagi pengambil keputusan saat pengambilan keputusan.*

Kata Kunci: Market Basket Analysis, Pincer Search, Association Rules

1. Pendahuluan

Dewasa ini perusahaan banyak yang sudah menggunakan sistem komputerisasi. Dengan sistem yang telah terkomputerisasi, perusahaan dapat mengumpulkan data transaksi dengan cepat, serta didukung dengan kemampuan menyimpan data yang besar juga. Namun sayang data yang telah terkumpul banyak itu seringkali hanya disimpan dan jarang digunakan untuk analisa, padahal bisa saja ada informasi-informasi tidak terduga sebelumnya yang bisa digali. Untuk itu perlu dibangun sebuah sistem pembantu pengambilan keputusan yang memudahkan user dalam menggali informasi dari data transaksi. Pada paper ini peneliti akan mengetengahkan penggunaan metode *Market Basket Analysis* menggunakan Algoritma *Pincer Search* untuk menggali informasi.

Pada sebuah apotek dapat digali berbagai macam informasi dari data transaksi pembelian obat - obatan, misal: asosiasi antar jenis obat yang dibeli konsumen, *packing* obat yang disukai konsumen, kapan obat tertentu banyak dibeli, dan lain lain. Informasi ini nantinya dapat digunakan pihak apotik dalam mengatur peletakan obat di etalase dan rak obat, sehingga dapat membantu pelayan apotik dan konsumen dalam mencari dan mengingatkan kembali bila lupa. Selain itu pihak apotik juga dapat menentukan kapan perlu mendatangkan lebih banyak jenis obat-obat tertentu di waktu-waktu tertentu. Keputusan - keputusan ini pada akhirnya dapat membantu apotik untuk lebih mengpotimalkan kinerjanya.

2. Metodologi

a. Survei dan Pengumpulan Data

Pada tahap ini peneliti melakukan survei dan pengumpulan sampel data pada sebuah apotik di surabaya. Sampel data ini berupa data transaksi penjualan obat mulai bulan mei 2004 sampai dengan september 2004.

b. Studi Literatur

Disini peneliti melakukan studi literatur tentang teori dan konsep *Data Mining*, *Market Basket Analysis*, *Association Rule*, Algoritma *Apriori* dan Algoritma *Pincer Search*.

c. Perencanaan dan Pembuatan Sistem

Perencanaan dan pembuatan Sistem Pembantu Pengambilan Keputusan ini dibagi menjadi 3 tahap yaitu:

- Perencanaan dan pembuatan modul *Pre-Processing Data* untuk merubah format data transaksi yang ada menjadi bentuk format data transaksi yang dapat diproses menggunakan Algoritma *Pincer Search*.



- Perencanaan dan pembuatan modul untuk memproses data menjadi informasi dalam bentuk *Association Rules* menggunakan Algoritma *Pincer Search*.
- Perencanaan dan pembuatan modul *Post-Processing* untuk menampilkan *Association Rules* yang dihasilkan dalam bentuk teks / tabel dan grafik.

d. Pengujian Sistem

Pengujian sistem dilakukan sebanyak tiga tahap, yaitu:

- Pengujian antar-muka dan kebenaran hasil proses dari sistem.
- Pengujian kecepatan proses sistem.
- Pengujian oleh calon pemakai, yaitu pemilik merangkap apoteker dari apotik yang digunakan datanya

e. Pengambilan Kesimpulan

Pengambilan kesimpulan berdasarkan hasil pengujian.

3. Tinjauan Pustaka

3.1. Market Basket Analysis

Market basket analysis adalah salah satu cara yang digunakan untuk menganalisis data penjualan dari suatu perusahaan. Proses ini menganalisis *buying habits* konsumen dengan menemukan asosiasi antar *item-item* yang berbeda yang diletakkan konsumen dalam *shopping basket*. Hasil yang telah didapatkan ini nantinya dapat dimanfaatkan oleh perusahaan retail seperti toko atau swalayan untuk mengembangkan strategi pemasaran dengan melihat *item-item* mana saja yang sering dibeli secara bersamaan oleh konsumen [3].

3.2. Association Rule

Association rule mining adalah suatu prosedur untuk mencari hubungan antar *item* dalam suatu *data set* yang ditentukan [3].

Association rule meliputi dua tahap [8]:

1. Mencari kombinasi yang paling sering terjadi dari suatu *itemset*.
2. Mendefinisikan *Condition* dan *Result* (untuk *conditional association rule*).

Umumnya ada dua ukuran kepercayaan (*interestingness measure*) yang digunakan dalam menentukan suatu *association rule*, yaitu [3]:

- **Support:** suatu ukuran yang menunjukkan seberapa besar tingkat dominasi suatu *item* / *itemset* dari keseluruhan transaksi. Ukuran ini menentukan apakah suatu *item* / *itemset* layak untuk dicari *confidence factor*-nya.
- **Confidence:** suatu ukuran yang menunjukkan hubungan antar 2 *item* secara *conditional* (misal, seberapa sering *item* B dibeli jika orang membeli *item* A).

Kedua ukuran ini nantinya berguna dalam menentukan *interesting association rules*, yaitu untuk dibandingkan dengan batasan (*threshold*) yang ditentukan oleh *user*. Batasan tersebut umumnya terdiri dari *min_support* dan *min_confidence*. Bila memenuhi kedua batasan maka sebuah *rule* dapat disebut *interesting rule*.

3.3. Algoritma Pincer-Search

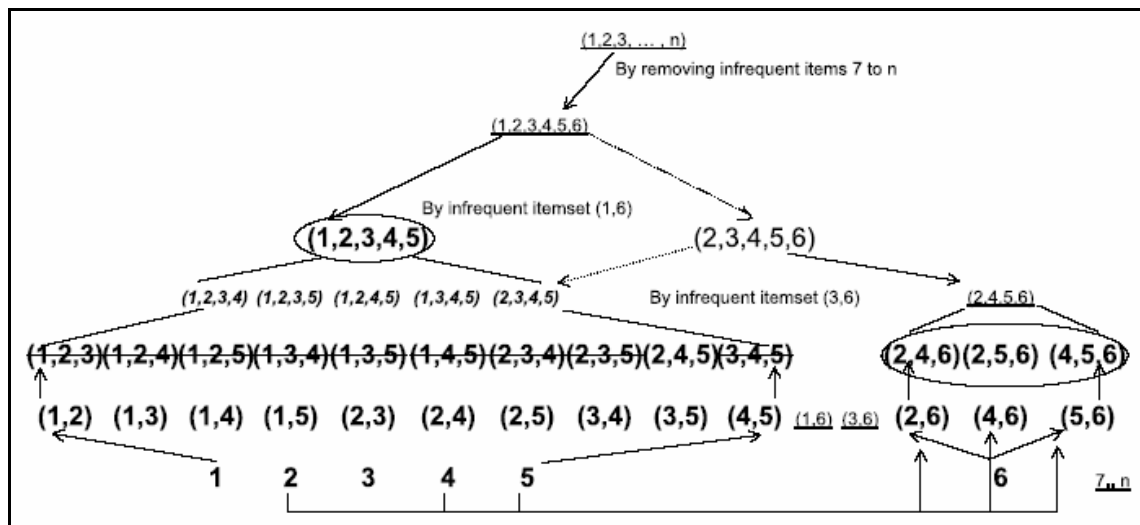
Pincer-Search dikembangkan oleh Dao-I Lin dan Zvi M. Kedem pada tahun 1997 [6, 7]. Algoritma ini dinamakan *Two-Way Search* karena memakai 2 cara pendekatan, yaitu *Top Down* serta *Bottom Up*. Dalam prosesnya arah pencarian utama dari *Pincer search* adalah *Bottom Up*, yaitu menggunakan Algoritma *Apriori* untuk menemukan *Frequent Itemset*, bedanya pada *Pincer search* secara *concurrent* dilakukan pencarian secara *Top Down*, yaitu *Maximum Frequent Candidat Set* (MFCS) yang menghasilkan output berupa *Maximum Frequent Set* (MFS). *Maximum Frequent Set* adalah sekumpulan *maximal itemset* - *itemset* yang tergolong *frequent*. Guna dari *Maximum Frequent Set* adalah untuk mengurangi (*prunning*) jumlah kandidat *Frequent Itemset* yang perlu diperiksa secara *Bottom Up*.

Selain menggabungkan 2 pendekatan *Bottom Up* dan *Top Down*, *Pincer Search* juga menggunakan dua properti khusus, yakni *Downward Closure Property* dan *Upward Closure Property*. Kedua properti ini digunakan untuk melakukan *pruning candidate set* sehingga dapat mengurangi waktu penghitungan. Pendekatan untuk melakukan *prune* dengan cara seperti ini adalah sebagai berikut [6]:

- Jika semua *Maximal Frequent Itemset* ditemukan pada arah *Top Down*, maka *itemset* tersebut dapat digunakan untuk mengeliminasi banyak kandidat dari arah *Bottom Up*. Semua *subset* dari *itemset* ini pastilah *frequent*.
- Jika semua *infrequent itemset* ditemukan di *Bottom Up*, hasilnya dapat digunakan untuk mengeliminasi kandidat yang ditemukan pada arah *Top Down* yang telah ditemukan sejauh ini.

Berikut ini adalah gambaran cara kerja dari Algoritma *Pincer Search*:



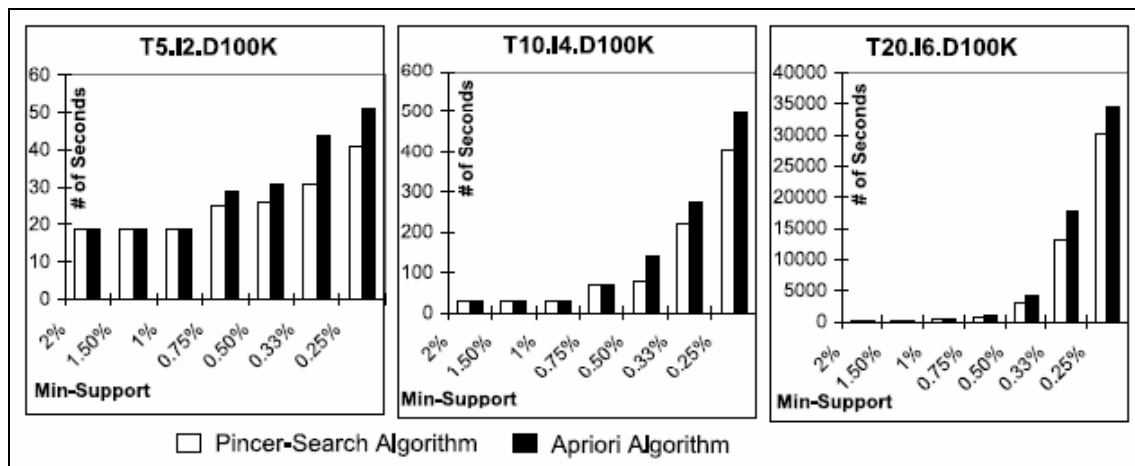


Gambar 1: Cara kerja Algoritma *Pincer-Search* [6, 7]

Keterangan gambar:

- Tulisan bercetak tebal adalah kandidat yang *frequent*
- Tulisan bergaris bawah adalah kandidat yang *infrequent*
- Tulisan bercetak miring adalah kandidat yang dihilangkan (*prune*) dengan MFCS
- Tulisan dengan coretan ditengahnya mewakili kandidat yang *frequent* tapi tidak perlu digunakan untuk membuat (*generate*) kandidat *itemset* yang baru.
- *Itemset* dengan garis yang melingkarinya mewakili *Maximum Frequent Itemset*

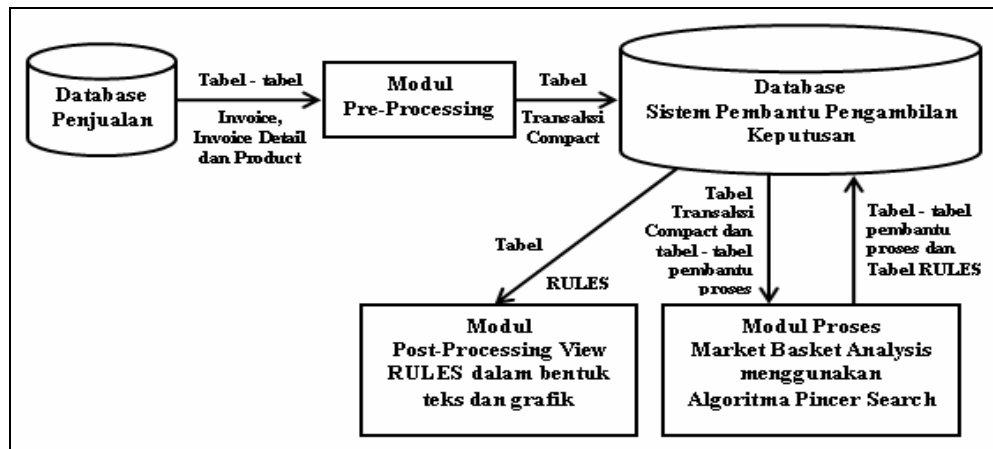
Dari hasil studi literatur terhadap pengujian yang membandingkan performace dari Algoritma *Pincer Search* dan Algoritma *Apriori* yang dilakukan oleh Dao-I Lin pada dissertasinya [7], peneliti memutuskan untuk menggunakan Algoritma *Pincer Search* untuk *Market Basket Analysis* pada sistem ini.



Gambar 2: Hasil Perbandingan kecepatan proses antara Algoritma *Pincer Search* dan Algoritma *Apriori* untuk data berjenis *scattered distribution* [6]



4. Desain Sistem

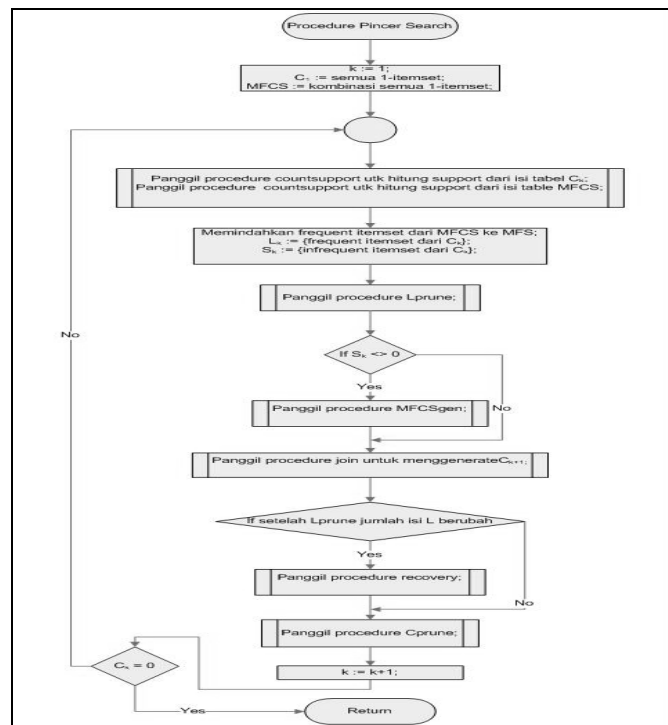


Gambar 3: Blok Diagram Sistem

Sistem Pembantu Pengambilan keputusan ini didisain dalam tiga modul dan satu database penyimpanan data - data sementara dan *Rules* yang dihasilkan, yaitu:

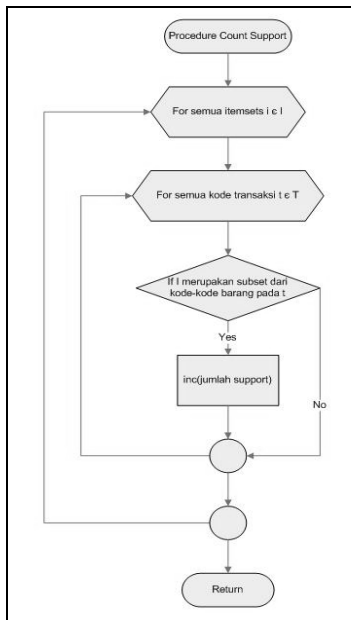
1. **Modul Pre-Processing Data**, digunakan untuk merubah data transaksi penjualan obat pada tabel *Invoice*, *Invoice Detail* dan *Product*, sesuai dengan periode tertentu yg dipilih, menjadi sebuah tabel *Transaksi Compact* yang memiliki attribut - attribut sebagai berikut: Date, CInvoiceNum, CPCode, Name.
2. **Modul Proses** adalah modul utama dari sistem ini, digunakan untuk merubah data - data pada tabel *Transaksi Compact* menjadi Tabel *Rules* menggunakan Algoritma *Pincer Search*. Tabel *Rules* berisi attribut - attribut sebagai berikut: Id, Combination, Rules, SupportPercent, ConfidencePercent.
3. **Modul Post-Processing**, digunakan untuk menampilkan rule - rule asosiasi yang dihasilkan pada tabel *Rules* kedalam bentuk teks maupun grafik.

Berikut ini adalah *flowchart* - *flowchat* proses *Pincer Search* dalam **Modul Proses**:

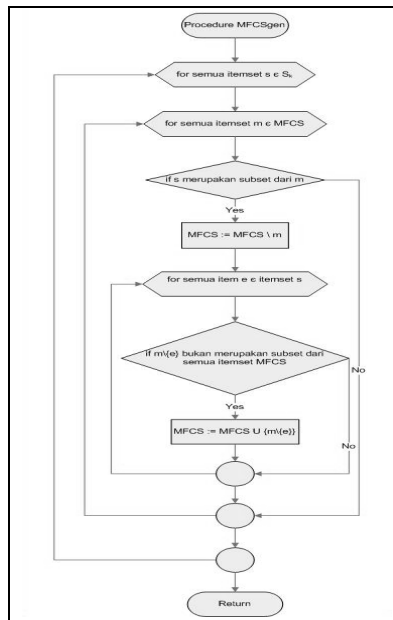


Gambar 4: Flowchart Prosedur *Pincer Search*

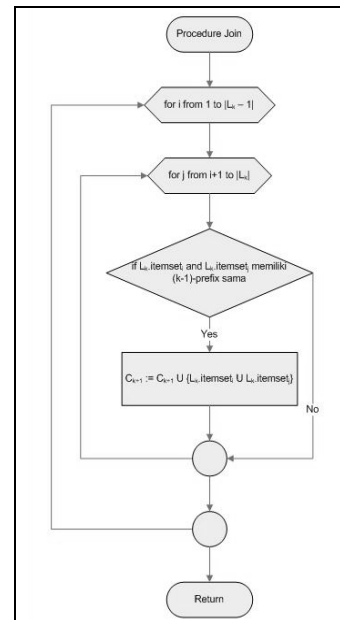




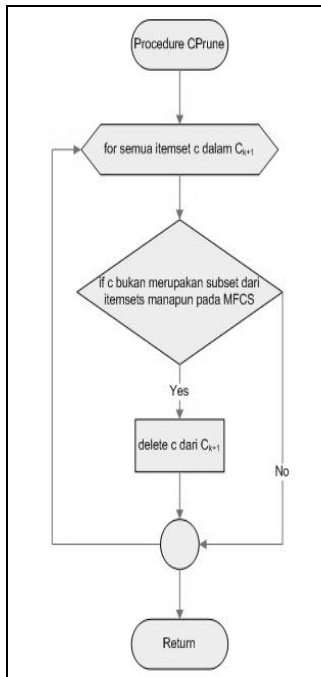
Gambar 5: *Flowchart* Prosedur CountSupport



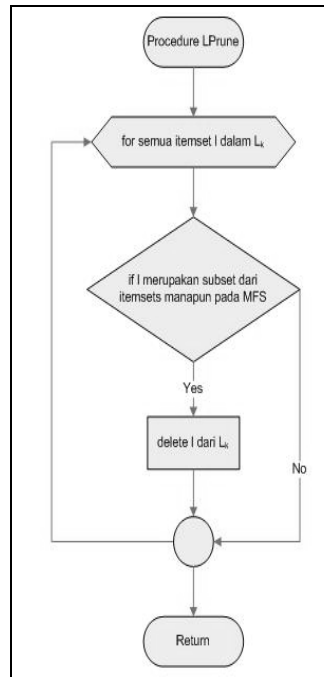
Gambar 6: *Flowchart* Prosedur MFCSgen



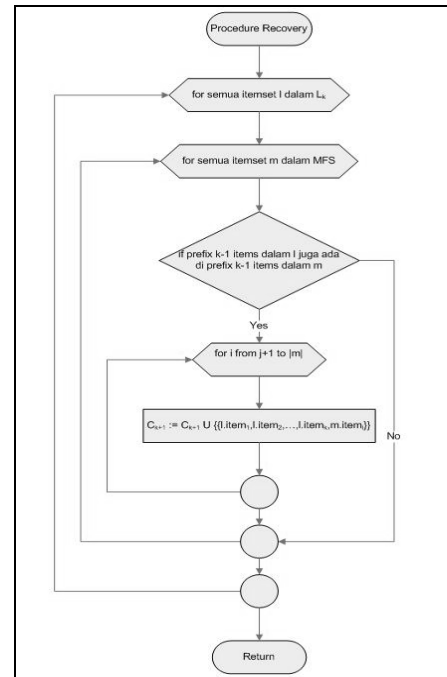
Gambar 7: *Flowchart* Prosedur Join



Gambar 8: *Flowchart* Prosedur CPrune



Gambar 9: *Flowchart* Prosedur LPrune



Gambar 10: *Flowchart* Prosedur Recovery



5. Pengujian

5.1. Hardware Yang Digunakan Untuk Pengujian

Pengujian program ini dilakukan pada komputer dengan spesifikasi sebagai berikut:

- Processor Pentium IV 1600 MHz
- Memory 512 Mbyte
- HardDisk 40 Gigabyte
- Sistem Operasi: Windows XP Professional
- Database: Microsoft SQL Server 2000

5.2. Pengujian Antar-Muka dan Kebenaran Hasil Proses

Untuk menguji kebenaran hasil proses dari sistem yang dibuat digunakan data sampel yang ada pada paper *Fast Algorithms for Mining Association Rules* [1], buku *Data Mining: Concepts and Techniques* [3], paper *Pincer-Search: A New Algorithm for Discovering the Maximum Frequent Sets* [6] dan disertasi *Fast Algorithms for Discovering the Maximum Frequent Set* [7]. Hasil pengujian perbandingan ini menunjukkan bahwa proses *generate Frequent Itemset* telah benar. Hal ini berarti bahwa Algoritma *Pincer Search* telah diprogram dengan baik dan benar. Salah satu contohnya dapat dilihat pada gambar - gambar berikut ini:

comb	no_seq	itemsets	support
1	1	1	4
1	2	2	3
1	3	3	3
1	4	4	2
*			

Tabel L1

comb	no_seq	itemsets	support
2	1	1,2	3
2	2	1,3	3
2	3	1,4	2
2	4	2,3	2
2	5	2,4	2
2	6	3,4	2
*			

Tabel L2

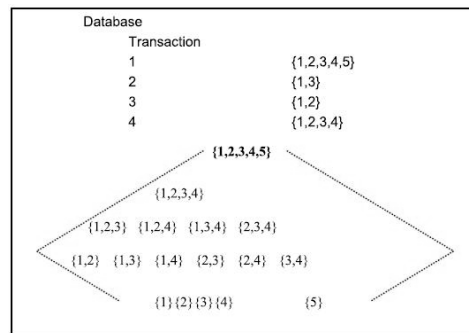
comb	no_seq	itemsets	support
3	1	2,3,4	2
3	2	1,3,4	2
3	3	1,2,4	2
3	4	1,2,3	2
*			

Tabel L3

comb	no_seq	itemsets	support
4	1	1,2,3,4	2
*			

Tabel L4

Gambar 11: Isi tabel pembantu proses L1, L2, L3 dan L4 setelah proses *generate Frequent Itemset* dengan *minimum support = 2*. Sampel data dari disertasi berjudul *Fast Algorithms for Discovering the Maximum Frequent Set* [7]



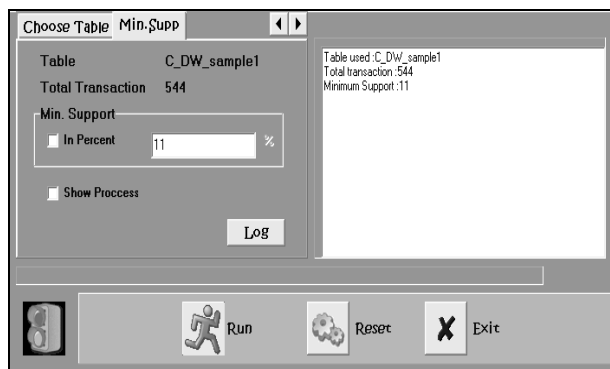
Gambar 12: Database sederhana dan proses *generate Frequent Itemset* dari disertasi berjudul *Fast Algorithms for Discovering the Maximum Frequent Set* [7]

Sementara itu hasil pengujian Antar-Muka dapat dilihat pada gambar - gambar berikut ini:

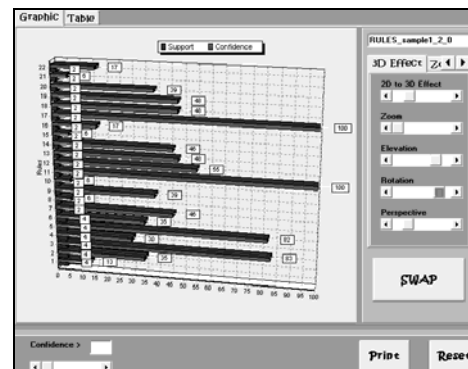
Gambar 13: Antar-Muka untuk menyusun tabel *custom* yang akan dianalisa, dari tabel *Invoice*, *Invoice Detail* dan *Product*

Gambar 14: Antar-Muka untuk memilih tabel *custom* dan merubahnya menjadi Tabel *Transaksi Compact*





Gambar 15: Antar-Muka untuk memilih *Minimum Support* dan melakukan proses *Pincer Search*



Gambar 16: Antar-Muka tampilan *Rules* secara grafis

comb	rules	conf	supp
2	POCARI SWEAT -> COMBANTRIN 250 MG TAB	40	3
2	COMBANTRIN 250 MG TAB -> POCARI SWEAT	100	3
2	DECOLGEN TABLET -> COMBANTRIN 250 MG TAB	40	3
2	COMBANTRIN 250 MG TAB -> DECOLGEN TABLET	100	3
2	DECOLGEN TABLET -> POCARI SWEAT	40	3
2	POCARI SWEAT -> DECOLGEN TABLET	40	3
2	NEOZEP FORTE -> VITACIMIN STRIP	100	3
2	VITACIMIN STRIP -> NEOZEP FORTE	33	3
2	CEREBROFORT PLUS DHA SYR -> VITACIMIN STRIP	75	4
2	VITACIMIN STRIP -> CEREBROFORT PLUS DHA SYR	50	4
2	FATIGON -> VITACIMIN STRIP	100	3
2	VITACIMIN STRIP -> FATIGON	33	3
2	PROMAG PER STRIP -> DECOLGEN TABLET	40	3
2	DECOLGEN TABLET -> PROMAG PER STRIP	40	3
2	LARUTAN PENYEGAR K3 200 ML -> DECOLGEN TABLET	67	3
2	DECOLGEN TABLET -> LARUTAN PENYEGAR K3 200 ML	40	3
2	SALONPAS SUPER 10 X 10 STRIP -> STOP COLD DRAGEE	50	3
2	STOP COLD DRAGEE -> SALONPAS SUPER 10 X 10 STRIP	100	3
3	BIAYA PUYER & BIAYA NON-PUYER -> PEHACORT TAB	40	3

Gambar 17: Antar-Muka tampilan *Rules* dalam teks (*Minimum Confidence* = 25 %)

comb	rules	conf	supp
3	BIAYA NON-PUYER & CAPSUL KOSONG 2 -> BIAYA PUYER	100	3
3	BIAYA PUYER & CAPSUL KOSONG 2 -> BIAYA NON-PUYER	100	3
3	BIAYA PUYER & BIAYA NON-PUYER -> CAPSUL KOSONG 2	40	3
3	CAPSUL KOSONG 2 -> BIAYA PUYER & BIAYA NON-PUYER	100	3
3	POCARI SWEAT & DECOLGEN TABLET -> COMBANTRIN 250 MG TAB	100	3
3	COMBANTRIN 250 MG TAB & DECOLGEN TABLET -> POCARI SWEAT	100	3
3	COMBANTRIN 250 MG TAB & POCARI SWEAT -> DECOLGEN TABLET	100	3
3	DECOLGEN TABLET -> COMBANTRIN 250 MG TAB & POCARI SWEAT	40	3
3	POCARI SWEAT -> COMBANTRIN 250 MG TAB & DECOLGEN TABLET	40	3
3	COMBANTRIN 250 MG TAB -> POCARI SWEAT & DECOLGEN TABLET	100	3
4	BIAYA PUYER & BIAYA NON-PUYER & ROMILAR TAB 15 MG -> PEHACORT TAB	67	3
4	PEHACORT TAB & BIAYA NON-PUYER & ROMILAR TAB 15 MG -> BIAYA PUYER	100	3
4	PEHACORT TAB & BIAYA PUYER & ROMILAR TAB 15 MG -> BIAYA NON-PUYER	100	3
4	PEHACORT TAB & BIAYA PUYER & BIAYA NON-PUYER -> ROMILAR TAB	100	3
4	BIAYA NON-PUYER & ROMILAR TAB 15 MG -> PEHACORT TAB & BIAYA PUYER	67	3
4	BIAYA PUYER & ROMILAR TAB 15 MG -> PEHACORT TAB & BIAYA NON-PUYER	67	3
4	BIAYA PUYER & BIAYA NON-PUYER -> PEHACORT TAB & ROMILAR TAB	40	3
4	PEHACORT TAB & ROMILAR TAB 15 MG -> BIAYA PUYER & BIAYA NON-PUYER	100	3
4	PEHACORT TAB & BIAYA NON-PUYER -> BIAYA PUYER & ROMILAR TAB	100	3

Gambar 18: Antar-Muka tampilan *Rules* dalam teks (*Minimum Confidence* = 25 %) (lanjutan)

5.3. Pengujian Sistem oleh Calon Pemakai

Sistem ini diuji oleh pemilik Apotik sekaligus Apoteker dari Apotik yang digunakan data transaksi penjualannya. Hasil pengujian dapat dilihat pada tabel berikut ini:

Tabel I: Hasil Penilaian dari Calon Pemakai

No	Kriteria	Nilai (*)
1	Keakuratan hasil <i>Association Rules</i>	50
2	Desain Antar - Muka	40
3	Kemudahan Penggunaan Sistem	40
4	Tingkat Kegunaan dari Informasi yang dihasilkan	50
Hasil Rata - Rata:		45

(*) Nilai: 10 → Sangat Buruk s/d 50 → Sangat Baik

5.4. Pengujian Kecepatan Proses dari Sistem

Berikut ini ditampilkan tabel hasil pengujian kecepatan proses dari sistem

Tabel II: Hasil Pengujian Kecepatan Proses untuk data 2 bulan

Jumlah Data	<i>Generate Frequent Item Set</i>			<i>Generate Association Rules</i>	
	<i>Minimum Support</i>	Jumlah Frequent Itemsets (kombinasi 2 ke atas)	Lama Proses	Jumlah Rules	Lama Proses
Transaksi 306	9	kombinasi 2 = 2	13 menit 14 detik	4	1 detik
Barang 569	6	kombinasi 2 = 5 kombinasi 3 = 1	1 jam 14 menit 48 detik	16	1 detik



Tabel III: Hasil Pengujian Kecepatan Proses untuk data 4 bulan

Jumlah Data	Generate Frequent Item Set			Generate Association Rules	
	Minimum Support	Jumlah Frequent Itemsets (kombinasi 2 ke atas)	Lama Proses	Jumlah Rules	Lama Proses
Transaksi 544 Barang 776	22	kombinasi 2 = 2	5 menit 13 detik	4	1 detik
	16	kombinasi 2 = 3	15 menit 40 detik	6	1 detik
	11	kombinasi 2 = 5 kombinasi 3 = 2	1 jam 27 menit 38 detik	22	2 detik
	5	kombinasi 2 = 29 kombinasi 3 = 10 kombinasi 4 = 1	15 jam 30 menit 48 detik	132	12 detik

6. Kesimpulan

- Kemampuan sistem melakukan customisasi terhadap data transaksi yang akan diproses memberikan tingkat fleksibilitas lebih tinggi kepada user dalam menggali informasi yang diinginkan.
- Dari hasil perbandingan hasil *Frequent Itemset* yang dibuat oleh sistem dan contoh sampel database yang ada pada literatur, dapat disimpulkan bahwa sistem pembantu pengambilan keputusan ini telah diprogram dengan baik dan benar.
- Hasil pengujian membuktikan bahwa sistem dapat memproses data 'real world' cukup banyak dan tidak mengalami 'error' pada saat proses, hal ini berarti bahwa sistem telah layak untuk digunakan.
- Kecepatan proses yang lambat dapat menjadi masukan bagi peneliti untuk lebih mengefisienkan teknik pemrograman sistem ini dengan menggunakan teknik lain yang lebih cepat, misal: pemrosesan data tanpa melibatkan *database* sebagai penyimpan data sementara.
- Dari hasil pengujian oleh calon pemakai dapat disimpulkan bahwa sistem cukup mudah dimengerti dan mudah digunakan sebagai sistem pembantu pengambilan keputusan.
- Dengan menganalisa *rules* yang dihasilkan oleh sistem, ditambah dengan pendapat dari calon pemakai, dapat disimpulkan bahwa informasi yang dihasilkan oleh sistem cukup menarik (*interesting*) dan berguna sebagai pendukung pengambilan keputusan bagi pihak apotik sebagai pemakai.
- Setelah menganalisa tampilan grafik dari *rules*, dapat disimpulkan bahwa tampilan grafik tersebut cukup membingungkan, terutama bila rule yang dihasilkan cukup banyak, dan kurang berguna bagi proses pengambilan keputusan. Hal ini dapat menjadi masukan bagi peneliti untuk mencari dan mengimplementasikan bentuk - bentuk tampilan grafis lain untuk menampilkan *rules*, misal: dalam bentuk *tree* atau *connected graph*.

Referensi

- [1] Agrawal, Rakesh, Ramakrishnan Srikant, 1994, Fast Algorithms for Mining Association Rules, *Proceeding Of The 1994 International Conference Very Large Data Bases*, Santiago, Chile, September 1994.
- [2] Dilly, Ruth, 2004, *Data Mining: An Introduction*, Parallel Computer Centre, Queens University Belfast.
- [3] Han, Jiawei, Micheline Kamber, 2001, *Data Mining: Concepts and Techniques*, Morgan Kaufmann
- [4] Han, Jiawei, Yongjian Fu, 1999, Discovery Of Multiple-Level Association Rules From Large Databases, *IEEE Transactions On Knowledge And Data Engineering*, Vol. 11, No. 5.
- [5] Hand, David, Heikki Mannila, Padhraic Smyth, 2001, *Principles Of Data Mining*, The MIT Press
- [6] Lin, Dao-I, Zvi M. Kedem, 1997, Pincer-Search: A New Algorithm for Discovering the Maximum Frequent Sets, *Proceeding Of The 6th International Conference On Extending Database Technology (EDBT)*.
- [7] Lin, Dao-I, 1998, *Fast Algorithms for Discovering the Maximum Frequent Set*, Dissertation, Department of Computer Science, New York University.
- [8] Ulmer, David, 2002, Mining an Online Auctions Data Warehouse, The Mid-Atlantic Student Workshop on Programming Languages and Systems, Pace University, <http://csis.pace.edu/cs/masplas/p8.pdf>
- [9] Witten, Ian H., Eibe Frank, 2000, *Data Mining Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann Publishers.

