

PROSES MENEMUKAN POLA DATA BISNIS RITAIL DENGAN PENGEMBANGAN ALGORITMA APRIORI MENGUNAKAN PENDEKATAN K-WAY JOIN

Joko Aryanto

Informatika, Fakultas Teknologi Informasi dan Elektro, Universitas Teknologi Yogyakarta
Jl.Siliwangi(Ring Road Utara), Jombor, Sleman, Yogyakarta
Joko.Aryanto@uty.ac.id

ABSTRAK

Persaingan sesama pelaku bisnis retail membuat para pemilik usaha ritel harus bersaing satu sama lain untuk memperoleh pelanggan guna meningkatkan pendapatan, dengan demikian pengelola dituntut untuk melakukan improvisasi dengan langkah yang efektif dan efisien. Algoritma Apriori merupakan metode dasar di dalam data mining. Prinsip yang digunakan algoritma Apriori adalah jika sebuah itemset sering muncul, maka seluruh subset dari itemset tersebut juga harus sering muncul. Hal ini mengakibatkan pengecekan yang berulang-ulang dan akan membutuhkan waktu yang tidak sebentar. Sebuah metode diusulkan untuk dapat memecahkan masalah ini, yaitu dengan melakukan pengembangan atau modifikasi pada algoritma Apriori tersebut, khususnya dalam proses penghitungan nilai support dan confidence. Metode yang digunakan adalah memanipulasi penggunaan bahasa query memanfaatkan teknik k-Way untuk optimasi susunan yang sesuai. Hasil yang didapat pada penelitian ini adalah waktu eksekusi relatif lebih cepat, dan hasil aturan asosiasi yang sama dengan yang dihasilkan oleh metode Apriori tanpa adanya pengembangan atau modifikasi.

Kata Kunci : Data Mining, Improvisasi Apriori, Algoritma *Association rule*, Pengembangan Apriori, k-way.

ABSTRACT

Competition among retail businesses makes retail business owners have to compete with each other to get customers to increase revenue, thus managers are required to improvise with effective and efficient steps. The Apriori algorithm is the basic method in data mining. The principle used by the Apriori algorithm is that if an itemset appears frequently, then all subset of itemset must also appear frequently. This results in repeated checks and will require a short time. A method is proposed to be able to solve this problem, namely by developing or modifying the Apriori algorithm, especially in the process of calculating the value of support and confidence. The method used is to manipulate the use of query languages using the k-Way technique for optimization of the appropriate arrangement. The results obtained in this study are relatively faster execution times, and the results of the association rules are the same as those produced by the Apriori method without any development or modification.

Keywords: Data Mining, Apriori Improvisation, Association rule Algorithm, Apriori Development, k-way.

PENDAHULUAN

Metode Market Basket Analysis adalah proses analisa pada level data transaksi yang dapat meningkatkan produktifitas bisnis. Gambarnya berupa kecendrungan kelompok item (terdiri dari 2 item atau lebih) yang dibeli oleh seorang konsumen dalam satu transaksi penjualan. Metode ini mampu

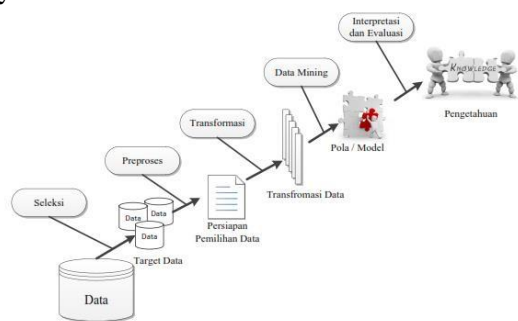
menemukan sebuah trend penjualan, pengelompokan produk umum atau mudah ditebak dan bahkan dapat menemukan trend yang tidak pernah terpikirkan sebelumnya. Market Basket Analysis didasarkan pada analisis dari kebiasaan konsumen dengan mencari asosiasi dan korelasi antar item berbeda

yang dilakukan pada satu transaksi yang sama oleh konsumen.

Data Mining

Data mining merupakan proses komputasi untuk menemukan pola dalam data set yang besar melibatkan interseksi metode dari kecerdasan buatan, pembelajaran mesin, statistik, dan sistem database. Tujuan keseluruhan dari proses data mining adalah untuk mengekstrak informasi dari kumpulan data dan mengubahnya menjadi struktur yang dapat dimengerti untuk digunakan lebih lanjut. Selain dari langkah analisis mentah, melibatkan database dan manajemen data aspek, data yang pra-pengolahan, model dan pertimbangan inferensi, metrik *interestingness*, pertimbangan kompleksitas, pasca-pengolahan struktur ditemukan, visualisasi, dan memperbaiki secara online.

Data mining merupakan bagian dari *Knowledge Discovery In Databases* (KDD). Sedangkan data mining memiliki tiga metode, yakni *clustering*, *classification* dan *Association Rule*. KDD adalah proses multi langkah yang mengubah data mentah menjadi pengetahuan yang berguna (Bagga and Singh, 2011). Tahapan-tahapan yang terjadi pada proses data mining atau *knowledge discovery* menurut Kenneth Collier (1998) dibagi menjadi 5 tahapan yaitu:



Gambar 1 Tahapan proses data mining

a. Seleksi Data

Tujuan dari fase ini adalah ekstraksi dari gudang data yang besar menjadi data yang relevan dengan analisis data mining. Proses ekstraksi data membantu untuk merampingkan dan mempercepat proses.

b. Data Preprocessing

Fase ini berkaitan dengan pembersihan data dan persiapan tugas yang diperlukan untuk memastikan hasil yang

benar. Menghilangkan *missing value* dalam data, memastikan bahwa nilai-nilai kode memiliki arti seragam dan memastikan bahwa tidak ada nilai data palsu adalah tindakan khas yang terjadi selama fase ini.

c. Transformasi Data

Tahap ini mengubah data ke dalam bentuk atau format yang sesuai untuk kebutuhan data mining. Proses normalisasi biasanya diperlukan dalam tahap data transformasi.

d. Data mining

Tujuan dari tahap data mining adalah untuk menganalisis database sesuai algoritma yang digunakan sehingga menemukan pola atau aturan yang bermakna serta menghasilkan model prediksi. Data mining adalah elemen inti dari siklus KDD.

e. Interpretasi dan Evaluasi

Sementara algoritma data mining memiliki potensi untuk menghasilkan jumlah yang tidak terbatas dari pola tersembunyi dalam data, banyak hasil dari proses tersebut mungkin tidak bermakna atau berguna. Tahap akhir ini bertujuan untuk memilih model-model yang valid dan berguna untuk membuat keputusan bisnis masa depan.

Aturan Asosiasi

Association rule mining merupakan salah satu teknik yang digunakan dalam data mining yang bertujuan untuk menemukan aturan- aturan asosiasi yang terdapat dalam sekumpulan data. Teknik *association rule mining* akan menemukan pola hubungan antar data yang saling berasosiasi. Teknik ini juga dikenal dengan nama market basket analysis yang mendefinisikan itemset dalam suatu transaksi secara bersamaan, Contohnya dalam proses transaksi penjualan barang. Prosesnya diawali dengan mencari sejumlah frequent itemset dengan dilanjutkan pembentukan aturan-aturan asosiasi yang diperoleh. Aturan yang menyatakan asosiasi antara beberapa atribut sering disebut affinity analysis atau market basket analysis. Metodologi dasar analisis asosiasi terbagi menjadi dua tahap:

a. Analisa pola frekuensi tinggi

Tahap ini mencari kombinasi item yang memenuhi syarat minimum dari nilai

support dalam database. Nilai support sebuah item diperoleh dengan rumus berikut:

$$\text{Support}(A) = \frac{\text{Jumlah Transaksi Mengandung } A}{\text{Total Transaksi}}$$

Sedangkan nilai support yang mengandung 2 itemset dapat dituliskan dalam rumus sebagai berikut :

$$\text{Support}(A \cap B) = \frac{\text{Jml Transaksi Mengandung } A \text{ dan } B}{\text{Total Transaksi}}$$

b. Pembentukan aturan asosiasi

Setelah semua pola frekuensi tinggi ditemukan, langkah berikutnya dilakukan pencarian aturan asosiatif yang memenuhi syarat minimum untuk confidence dengan menghitung confidence aturan asosiatif $A > B$. Nilai confidence dari aturan $A > B$ diperoleh dari rumus berikut:

$$\text{Support } P(B/A) = \frac{\text{Jumlah Transaksi Mengandung } A \text{ dan } B}{\text{Total Transaksi mengandung } A}$$

Analisis asosiasi didefinisikan suatu proses untuk menemukan semua aturan asosiasi yang memenuhi syarat minimum untuk support (minimum support) dan syarat minimum untuk confidence (minimum confidence).

Algoritma Apriori

Dari penelitian yang dilakukan D. M. Tank, dijelaskan bahwa algoritma Apriori adalah algoritma klasik untuk memperoleh aturan asosiasi dari data yang sering muncul. Ide dasar dari algoritma ini adalah dengan menggunakan pendekatan berulang lapisan demi lapis untuk menemukan yang frekuensi kemunculan yang sering muncul. Proses pertama algoritma akan mendapatkan k-itemset, dan kemudian menggunakan itemsets k- untuk mengeksplorasi (k + 1) itemset. Pertama, memperkenalkan pengetahuan apriori itemset yang sering muncul, setiap bagian dari frekuensi itemset juga itemset yang sering muncul. Algoritma Apriori menggunakan pengetahuan sebelumnya dari itemset yang sering muncul, pertama dengan menemukan koleksi keseringan 1-itemset, dinotasikan L_1 . Kemudian gunakan 2 itemset L_1 untuk mendapatkan L_2 , dan kemudian L_2 , dan seterusnya, sampai tidak dapat ditemukan frekuensi k-itemset. Algoritma Apriori terutama terdiri dari tiga langkah berikut :

1. Langkah menghubungkan : menghubungkan kfrekuensi itemset untuk

menghasilkan kandidat (k + 1), dinotasikan dengan C_{k+1} . Kondisi selanjutnya dari langkah menghubungkan adalah bahwa dua kitemset memiliki item pertama (k-1) yang sama dan item k-selanjutnya yang berbeda. Ditunjukkan $l_{1|j}$ merupakan item ke- j dari l_1 dapat ditunjukkan sebagai berikut :

$$\{l_1[1], l_1[2], l_1[3], \dots, l_1[k], l_2[k]\}$$

2. Langkah pemangkasan : Untuk memilih itemset yang sering muncul L_{k+1} berasal dari kandidat C_{k+1} , karena kandidat set C_{k+1} merupakan superset dari itemset L yang sering muncul. Menurut sifat Apriori: setiap himpunan bagian dari frekuensi set juga harus sering muncul, yaitu setiap (k-1) item himpunan bagian dari k-item juga harus sering muncul. Dengan properti ini kita dapat mengetahui apakah item k-subset dari C_{k+1} berada di L_{k+1} , jika tidak, maka hapus calon (k+1) - itemset dihapus dari C .
3. Langkah menghitung : scanning data dalam database, menumpuk jumlah calon k+1 muncul dalam database. Jika jumlah kemunculan dari calon kurang dari ambang batas minimum support yang diberikan, calon itemset akan dihapus.

Algoritma yang digunakan dalam pencarian pola antara lain Apriori, FP-Growth, dan CT-Pro. Penggunaan Algoritma Apriori dikenal dengan metode asosiasi, dengan ide dasar menghitung pola kemunculan item yang muncul dalam data transaksi dengan beberapa iterasi sehingga akan diperoleh pola dan aturan. Kendala mendasar dalam proses pencarian hubungan antar data yaitu dibutuhkannya waktu yang tidak sebentar.

Struktur Kombinasi

Struktur dari itemset disini adalah mengikuti suatu bentuk dari kombinasi. Pengertian kombinasi adalah menggabungkan beberapa objek dari suatu grup tanpa memperhatikan urutan. Di dalam kombinasi, urutan objek tidak diperhatikan sebagai contoh dimana {1,2,3} adalah sama dengan {2,3,1} dan {3,1,2}. Kombinasi dapat dibagi menjadi dua yaitu kombinasi dengan pengulangan dan kombinasi tanpa pengulangan. Kombinasi tanpa pengulangan ketika urutan tidak diperhatikan akan tetapi setiap objek yang ada hanya bisa dipilih

sekali maka jumlah kombinasi yang ada adalah:

$$\frac{n!}{r!(n-r)!} = \binom{n}{r}$$

Dimana n adalah jumlah objek yang bisa dipilih dan r adalah jumlah yang harus dipilih. Sebagai contoh, terdapat 5 pensil warna dengan warna yang berbeda yaitu merah, kuning, hijau, biru dan ungu. Pensil warna tersebut hanya boleh dipilih dua warna. Banyak cara untuk mengkombinasikan pensil warna yang ada dengan menggunakan rumus di atas adalah :

$$5!/(5-2)!(2)! = 10 \text{ kombinasi}$$

Kombinasi dengan pengulangan jika urutan tidak diperhatikan dan objek bisa dipilih lebih dari sekali, maka jumlah kombinasi yang ada adalah ditunjukkan pada rumus berikut :

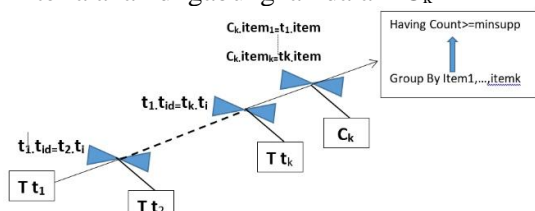
$$\frac{(n+r-1)!}{r!(n-1)!} = \binom{n+r-1}{r} = \binom{n+r-1}{n-1}$$

Di mana n adalah jumlah objek yang bisa dipilih dan r adalah jumlah yang harus dipilih. Sebagai contoh adalah terdapat 10 jenis kue donat berbeda pada suatu toko donat. Kombinasi yang dihasilkan jika ingin untuk membeli tiga buah donat adalah :

$$(10+3-1)!/3!(10-1)! = 220 \text{ kombinasi.}$$

Konsep K-Way

Pendekatan menggunakan metode K-Way merupakan salah satu metode untuk mengurangi beban kerja terutama dalam menghitung nilai support dan confidence dalam sekumpulan data yang sangat banyak. Tujuan optimasi dan analisis menggunakan pendekatan ini adalah untuk mengetahui dampak dari optimasi pada dataset yang memiliki karakteristik seperti banyaknya transaksi rata-rata, dukungan, kepercayaan, dan lainnya untuk mendapatkan heuristik yang dipengaruhi oleh input dataset. Untuk penghitungan support, setiap k yang sesuai kriteria akan di gabungkan dalam C_k



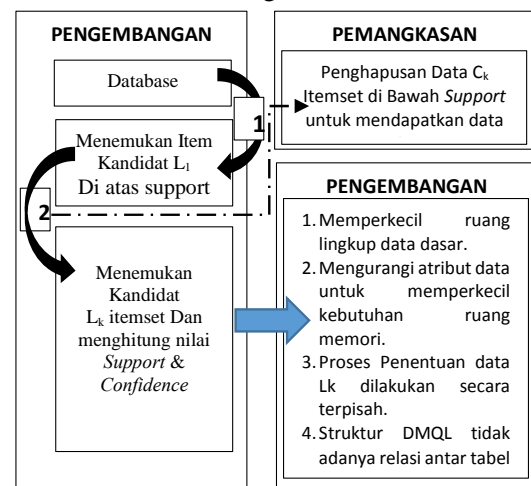
Gambar 2. Konsep K-Way

Total jumlah item yang dihasilkan dari penggabungan C_k dengan T sama dengan

jumlah data dalam C_k *rata-rata support dari item pertama dalam C_k . Dengan menggunakan notasi “join”, nilai join dapat di wakili sebagai gabungan (C_k, R, C_k*S_1). Dengan demikian, gabungan dari T dengan C akan menghasilkan sebuah tabel, yang berisi jumlah support untuk item pertama itemset di C_k . Dalam hal notasi join ini bisa diwakili sebagai join in (C_k*sm-1, R, C_k*sm). Nilai penggabungan terakhir, tidak dapat dihitung dari rumus di atas sebagai nilai s untuk itemset dengan panjang k tidak diketahui, karena itemset dari panjang k yang dihasilkan tidak sering muncul. Tetapi relasi yang diperoleh dari join terakhir akan memiliki catatan tupel sebagai jumlah dukungan untuk setiap itemset di set C_k . Menggunakan notasi join ini dapat direpresentasikan sebagai join ($C_k * sk-1, R, S(C_k)$). Oleh karena itu untuk pendekatan ini dapat dinotasikan sebagai:

METODE PENELITIAN

Dalam pengembangannya, untuk dapat diperoleh hasil yang diinginkan, sebagai acuan digunakan bagan kerja sebagai pemandu langkah- langkah yang akan dilakukan adalah sebagai berikut,



Gambar 3. Bagan kerja

Langkah pengembangan dijelaskan sebagai berikut,

1. Langkah pertama memperkecil ruang lingkup data dasar yang memiliki jumlah besar menjadi lebih sedikit sehingga proses pencarian data menjadi lebih mudah.
2. Data dengan ruang lingkup lebih kecil, karena sudah pasti bahwa data tersebut yang akan dilakukan pemrosesan,

dimungkinkan untuk mengurangi atributnya sehingga dapat mengurangi permasalahan filter data.

3. Penentuan kandidat item set dilakukan secara terpisah setelah data diletakkan dalam tabel sementara.
4. Proses DMQL (Data Mining Query Language) dalam satu buah tabel tanpa adanya relasi antar tabel akan mempengaruhi kinerja waktu proses.

Peneliti membagi menjadi beberapa tahapan untuk dapat mengurangi waktu proses yaitu :

1. Proses pemilihan kandidat item.

Proses ini dapat mengurangi waktu dikarenakan akan langsung mengambil data yang memiliki nilai support sama dengan atau lebih besar dari nilai yang sudah kita tentukan. Proses di atas akan mendapatkan hasil item, support atau jumlah kontribusi, dan nilai confidence dari item- item yang menjadi kandidat. Dalam proses selanjutnya tidak akan dilakukan seleksi ulang untuk dilakukan pemangkasan data. Berbeda dengan Apriori tradisional, dalam pencarian data kandidat item akan ditampilkan semua item tanpa memperhatikan item mana yang memiliki nilai support dan confidence dengan batasan yang kita tentukan. Syntax pencarian kandidat untuk algoritma tradisional adalah seperti yang terlihat di bawah ini :

```
//Seleksi Item
select * from (SELECT j.ITEM,count(J.trans)
as juml, (COUNT(J.trans)/@ttl)*100 as pross
FROM D_Item j,TRANS TJ where
Tj.trans=j.trans and tj.TGL >= @tg11 and
tj.TGL<= @tg12 GROUP BY ITEM) as
DataTabel where pross>=@suport
```

Pada syntax di atas terdapat dua proses yaitu menampilkan semua kandidat yang muncul, kemudian dilanjutkan dengan seleksi data yang masuk kriteria batasan support yang sudah ditentukan sebelumnya.

2. Dalam proses menentukan kandidat itemset.

Proses dalam menemukan pasangan kandidat itemset yang dengan menentukan jumlah *support* dan *confidence* merupakan proses yang membutuhkan waktu tidak sedikit. Dalam proses algoritma tradisional

dilakukan proses untuk menghasilkan data C_k yaitu menemukan semua pasangan kandidat item set yang selanjutnya dilakukan proses untuk menemukan data L_k dimana data dalam L_k merupakan data yang sudah dilakukan pemangkasan data dengan nilai *support* dan *confidence* dibawah nilai yang telah kita tentukan sebelumnya.

Berbeda dengan proses pengembangan algoritma yang peneliti lakukan yaitu secara langsung menemukan data dalam L_k tanpa mencari terlebih dahulu data C_k . Jadi proses yang dilakukan tanpa melewati proses scanning C_k terlebih dahulu.

syntax query di atas digunakan dalam pengembangannya untuk mencari kandidat itemset yang dengan menemukan L_k bukan untuk menemukan C_k terlebih dahulu. Sehingga tidak akan melewati proses pemangkasan data C_k untuk mendapatkan data L_k . Dengan demikian proses di sini akan mengurangi waktu dalam menemukan semua kandidat itemset, dan menghilangkan waktu proses pemangkasan. Berikut adalah bentuk *pseudocode*.

```
L1:={Kandidat itemset}
K:=2
While (Lk-1<>0) do
Begin
Forall transaksi Lk-1 do
Lk :=select count(*) from kadidat pasangan
yang bersal dari Lk-1
K:=K+1
End
Answer :=Uk Lk
```

```
Insert into Lk-itemset
Select * from (select p.item1,p.item2
,...,p.itemk-1 , p.nsup,p.nconf From L(k-1)-itemset p,
L(k-1)-itemset q where p.item1
=q.item1,..., p.itemk-2=q.itemk-2, p.itemk-1 <
q.itemk-1 ) as Tabel
where nsup>=sup and nconf>=conf
```

HASIL DAN PEMBAHASAN

Data yang digunakan berasal dari basisdata sistem Informasi Manajemen Retail SFA Toserba. Data yang digunakan adalah data transaksi dari bulan Oktober 2014 hingga November 2016 dengan jumlah record untuk tabel transaksi mencapai lebih dari 1328282 record. Terdapat tiga partisi penting yang mempengaruhi proses penelitian ini yaitu,

Tabel 1. Partisi proses penelitian

Atribut	Type Data	Keterangan
Tgl_jual	Datetime	Kelompok Trasnaksaksi
Kode_jual	Varchar	T _{id}
Kode_brg	Varchar	T _{item}

Pemasangan Kandidat Item Set k-way Join

Dalam pengembangan model yang dilakukan dengan memperkecil ruang lingkup pencarian dalam tabel "JUAL" tersebut. Langkahnya akan mengambil data transaksi yang terjadi pada batasan tanggal yang dimaksud dengan tujuan agar jumlah total data akan lebih sedikit dibandingkan dengan total data secara keseluruhan. Langkah- langkah yang dilakukan dari syntax tersebut adalah sebagai berikut :

1. Membuat tabel penampungan untuk menampung data pasangan itemset dalam hal ini adalah tabel @L₁.
2. Dengan menggunakan perintah perulangan "while" dilakukan memasangkan kandidat item dari tabel C₁.
3. Dalam proses memasangkan item, sekaligus dilakukan penghitungan nilai support dan confidence dari pasangan tersebut.
4. Jika nilai support dan confidence sesuai dengan batasan yang telah ditentukan, maka akan dimasukkan ke dalam tabel L₁.

Hitung Support dan Confidence

Dalam tahapan ini merupakan proses yang membutuhkan waktu paling lama, karena akan dilakukan scanning data secara keseluruhan dari item set yang terbentuk. Implementasi rumus untuk menghitung nilai support dan confidence dilakukan dengan pseudocode,

```
@tgl1=batas awal tanggal transaksi
@tgl2=batas akhir tanggal transaksi
@ttl= select count(kode_tran) as jml FROM
@C1 WHERE tanggal>= @tgl1 AND
tanggal<=@tgl2 AND jumlah item<=3;
@transak= select count(kode_tran) as jml
FROM @C1 WHERE tanggal>= @tgl1
AND tanggal<=@tgl2 A ND jumlah
item<=3 AND (item1,item2)
@support=(@transak/@ttl)*100
```

Setelah diperoleh nilai jumlah transaksi yang mengandung A dan B maka dapat di hitung nilai supportnya,

```
//memasukkan data ke tabel @L1 yang sesuai
kriteria
Jika (item1<>item2) and (item2<>item1) and
(confidance>=batas)
Insert @L1 @itm1,@itm2,@jml, (@support),
(@conf)
```

Perbandingan Proses Pengujian Data

Proses selengkapnya dilakukan pengujian metode yang dilakukan dengan pembagian 3 kelompok data berdasarkan rentang waktu transaksi.

1. Uji coba kelompok data 1

Menghasilkan data kandidat itemset berjumlah 5 item merupakan data transaksi dalam rentang waktu tanggal 1-10 bulan November 2016, jumlah transaksi yang terjadi sebanyak 11885 dengan data C₁=5 item, L₁=2 item, L₂=0 item menghasilkan waktu tempuh untuk syntax tradisional dalam hitungan detik adalah 69 detik sedangkan untuk pengembangan syntax menghasilkan waktu 65 detik dengan kandidat item,

Tabel 2. Kelompok data 1

Item	Jumlah Trans	Dalam %
100038	1310	11.02
100201	1940	16.32
100336	1220	10.26
100338	2030	17.08
753797	1280	10.76

Hasil pemasangan kandidat item yang terbentuk aturan asosiasi L₁ sebagai berikut :

- a. Jika membeli 100201 maka 100338 jumlah transaksi sebanyak 140 transaksi dan nilai confidence sebesar 72.16 %.
- b. Jika membeli 100338 maka 100201 jumlah transaksi sebanyak 140 transaksi dan nilai confidence sebesar 68.96 %.

Tabel 3. Nilai confidence data 1

Item 1	Item 2	Jumlah Trans	Nil Confidence (%)
100201	100338	140	72.16
100338	100201	140	68.96

2. Uji coba kelompok data 2

Merupakan data transaksi dalam rentang waktu tanggal 1-20 bulan November 2016, jumlah transaksi yang terjadi sebanyak

23631 dengan data C1=4 item, L1=2 item, L2=0 item menghasilkan waktu tempuh untuk syntax tradisional dalam hitungan detik adalah 85 detik sedangkan untuk pengembangan syntax menghasilkan waktu 77 detik.

Tabel 4. Kelompok data 2

Item	Jumlah Trans	Dalam %
100038	2650	11.21
100201	3760	15.91
100338	3900	16.50
753797	2440	10.32

Hasil pemasangan kandidat item yang terbentuk aturan asosiasi L1 sebagai berikut :

- a. Jika membeli 100201 maka 100338 jumlah transaksi sebanyak 270 transaksi dan nilai confidence sebesar 71.80 %.
- b. Jika membeli 100338 maka 100201 jumlah transaksi sebanyak 270 transaksi dan nilai confidence sebesar 69.23 %.

Tabel 5. Nilai confidence data 2

Item 1	Item 2	Jumlah Trans	Nil Confidence (%)
100201	100338	270	71.80
100338	100201	270	69.23

3. Uji coba kelompok data 3

Merupakan data transaksi dalam rentang waktu tanggal 1-30 bulan November 2016, jumlah transaksi yang terjadi sebanyak 34589 dengan data C1=4 item, L1=3 item, L2=0 item menghasilkan waktu tempuh untuk syntax tradisional dalam hitungan detik adalah 185 detik sedangkan untuk pengembangan syntax menghasilkan waktu 181 detik.

Tabel 6. Kelompok data 3

Item	Jumlah Trans	Dalam %
100038	3810	11.01
100201	5140	14.86
100336	3530	10.20
100338	5680	16.42

Hasil pemasangan kandidat item yang terbentuk aturan asosiasi L1 sebagai berikut :

- a. Jika membeli 100201 maka 100338 jumlah transaksi sebanyak 300 transaksi dan nilai confidence sebesar 58.36 %.
- b. Jika membeli 100338 maka 100201 jumlah transaksi sebanyak 300 transaksi dan nilai confidence sebesar 52.81 %.

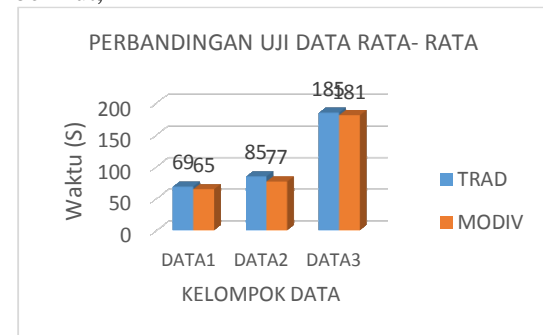
Tabel 7. Nilai confidence data 3

Item 1	Item 2	Jumlah Trans	Nil Confidence (%)
100201	100338	300	58.36
100338	100201	300	52.81

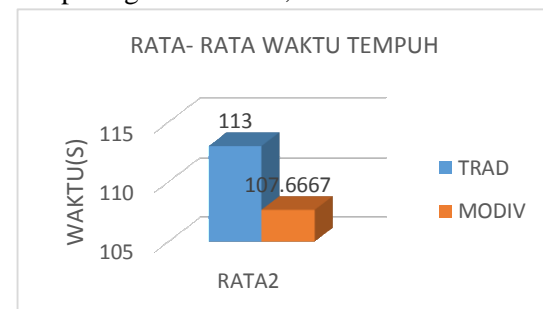
Perbandingan Uji Data

Dari ketiga kelompok data yang dilakukan pengujian dilakukan perhitungan nilai tempuh rata-rata diperoleh nilai sebagai berikut :

Dalam bentuk grafik dapat dilihat seperti berikut,



Gambar 4. Perbandingan uji data rata-rata Perbandingan nilai rata-rata kedua metode ada pada grafik berikut,



Gambar 5. Rata-rata waktu tempuh Dari ketiga kelompok data yang dilakukan uji algoritma, terjadi nilai penurunan waktu sebesar 4,95 %.

Tabel 8. Uji algoritma

JENIS	DATA 1	DATA 2	DATA 3	RATA-RATA	SELISIH (%)
TRAD	69	85	185	113	49,5
MODIV	65	77	181	107,66	

PENUTUP

Hasil uji algoritma dalam 2 metode yang dilakukan diperoleh kesimpulan untuk dapat dijadikan acuan penelitian berikutnya adalah,

1. Penghilangan proses penghapusan itemset yang tidak memenuhi kriteria dapat mengurangi waktu proses scanning.

2. Pendekatan menggunakan metode k-way join dalam penerapan algoritma apriori, dapat meningkatkan performa dalam iterasi yang terjadi. Pendekatan ini juga dapat mengurangi beban kerja terutama dalam menghitung nilai support dan confidence. Fokus dari proses kerja k-way join adalah DMQL atau Data Mining Query Language. Dengan menentukan formula tanda operator dalam penelitian ini dilakukan pengurangan klausa group by, between, dan relasi banyak partisi dengan model data yang dipilih menggunakan vertical data dapat mengurangi beban dengan hasil rata-rata waktu tempuh untuk apriori tradisional 71,66 detik sedangkan menggunakan pendekatan k-way join mencapai 61,11 detik, sehingga terjadi penurunan waktu tempuh selama 10,55 detik.
3. Pendekatan k-way join method mengurangi beban kerja yaitu proses yang terjadi dengan penurunan waktu tempuh selesai dilakukan secara langsung mempengaruhi konsumsi memory.

Sebagai bahan acuan untuk dilakukan penelitian selanjutnya, ada beberapa hal yang harus diperhatikan berdasarkan penelitian ini,

1. Isi atribut "kode_brg" yang digunakan menggunakan jumlah karakter seminimal mungkin. Dikarenakan akan mempengaruhi proses *searching* menjadi lebih lama. Meskipun hanya akan berpengaruh dalam jumlah record yang sangat besar.
2. Pemilihan tanda operator yang tepat dalam proses DMQL sangat berpengaruh. Perbedaan DBMS yang digunakan menyebabkan perbedaan hasil yang berbeda meskipun menggunakan tanda operator yang sama.

DAFTAR PUSTAKA

- V. Mohan and D. S. Rajpoot, "Matrix-OverApriori: An Improvement Over Apriori Using Matrix," vol. 5, no. 01, pp. 1–6, 2016.
- J. H. and M. Kamber, *Data Mining :Concept and Technique*, S. Edition, S.Edition. 500 Sansome Street, Suite 400, San

Francisco, CA 94111 This: Diane Cerra, 2006.

- D. Edwards, "Data Mining: Concepts, Models, Methods, and Algorithms," *J. Proteome Res.*, vol. 2, no. 3, pp. 334–334, 2003.
- M. B. Nichol, T. K. Knight, T. Dow, G. Wygant, G. Borok, O. Hauch, and R. O'Connor, "Fast Algorithms for Mining Association Rules," *Ann. Pharmacother.*, vol. 42, no. 1, pp. 62–70, 2008.
- C. D. Cleaning, D. D. Transformation, A. T. Analysis, E. D. Mining, and B. D. Selection, "Three phase iterative model of kdd," vol. 4, no. 2, pp. 695–697, 2011.
- S. A. Abaya, "Association Rule Mining based on Apriori Algorithm in Minimizing Candidate Generation," vol. 3, no. 7, pp. 1–4, 2012.
- J. Singh and H. Ram, "Improving Efficiency of Apriori Algorithm Using," vol. 3, no. 1, pp. 1–4, 2013.
- J. Yabing, "Research of an Improved Apriori Algorithm in Data Mining Association Rules," vol. 2, no. 1, pp. 25–27, 2013.
- D. M. Tank, "Improved Apriori Algorithm for Mining Association Rules," *Int. J. Inf. Technol. Comput. Sci.*, vol. 6, no. 7, pp. 1523, 2014.
- S. Chaudhari, M. Borkhatariya, A. Churi, and M. Bhonsle, "Implementation and Analysis of Improved Apriori Algorithm," pp. 70–78, 2008.
- J. Kaur, R. Singh, and R. K. Gurm, "Performance evaluation of Apriori algorithm using association rule mining technique," vol. 2, no. 5, 2016.
- A. Ansari, A. Parab, and S. Kadam, "Apriori A Big Data Analysis - A Review," pp. 35173520.
- Q. Liu and J. Xin, "An improved Apriori algorithm based on data stream classification," *J. Comput. Inf. Syst.*, vol. 10, no. 23, pp. 10259–10266, 2014.
- R. S. Dm, V. Saldanha, and S. Sebastian, "Apriori Algorithm and its Applications

- in The Retail Industry for Analyzing Customer Interests,” vol. 2, no. 3, pp. 46–51, 2015.
- B. S. Dhak and M. Sawarkar, “Apriori: a promising data warehouse tool for finding frequent itemset and to define association rules,” vol. 4, no. 1, pp. 60–65, 2016.
- N. Gutierrez, “Demystifying Market Basket Analysis,” <http://www.informationmanagement.com>, 2006. .
- ACM SIGKDD, Data Mining Curriculum. 2016.
- M. Kantardzic, Data Mining: Concepts, Models, Methods, and Algorithms, Vol. 2. No. Proteome Research, 2003.
- Pete Chapman., Step-by-step data mining guide. The CRISP-DM Consortium, 2000.
- Tan, P.N., Steinbach, M., Kumar, V., Introduction to Data Mining, AddisonWesley, Boston., 2006
- L. Notes, C. Science, and C. Growth, “Performance Evaluation and Analysis of KWay Join Variants for Association Rule Mining,” no. January, 2003.