

PERBANDINGAN KINERJA ALGORITMA APRIORI DAN EQUIVALENCE CLASS TRANSFORMATION (ECLAT) DALAM MENEMUKAN POLA PEMBELIAN PADA DATA TRANSAKSI MINIMARKET

PERFORMANCE COMPARISON OF APRIORI AND EQUIVALENCE CLASS TRANSFORMATION (ECLAT) ALGORITHMS IN FINDING PURCHASE PATTERNS IN MINIMARKET TRANSACTION DATA

I Putu Susila Handika¹⁾, I Kadek Susila Satwika²⁾

¹⁾ Jurusan Teknik Informatika, Institut Bisnis dan Teknologi Indonesia

²⁾ Jurusan Rekayasa Sistem Komputer, Institut Bisnis dan Teknologi Indonesia

Jl. Tukad Pakerisan No.97, Panjer, Denpasar Selatan, Kota Denpasar, Bali

Email : susila.handika@instiki.ac.id¹⁾, susila.satwika@instiki.ac.id²⁾

Abstrak

Studi ini membandingkan kinerja algoritma Apriori dan ECLAT dalam menganalisis data transaksi penjualan dari sebuah minimarket. Penelitian ini berfokus pada efisiensi kedua algoritma dalam hal waktu eksekusi dan penggunaan memori saat mengidentifikasi itemset yang sering muncul dan menghasilkan aturan asosiasi. Mengingat terbatasnya variasi produk yang dijual di minimarket, nilai minimum support (0,001) dan minimum confidence (0,005) yang lebih rendah diterapkan untuk memastikan hasil yang bermakna, karena ambang batas yang lebih tinggi tidak menghasilkan temuan signifikan. Pengujian pertama mengevaluasi waktu yang dibutuhkan untuk menemukan itemset yang sering muncul, menunjukkan bahwa ECLAT secara konsisten mengungguli Apriori dengan rata-rata waktu eksekusi 0,71634 detik dibandingkan 4,88256 detik pada Apriori. Pengujian kedua menilai waktu yang dibutuhkan untuk menghasilkan aturan asosiasi, di mana ECLAT kembali menunjukkan kinerja yang sedikit lebih baik, dengan rata-rata 0,01352 detik dibandingkan dengan 0,01618 detik pada Apriori. Pengujian penggunaan memori menunjukkan bahwa ECLAT lebih efisien, menggunakan rata-rata 0,12436 MB untuk menemukan itemset yang sering dan 0,01052 MB untuk menghasilkan aturan asosiasi, dibandingkan dengan 0,1385 MB dan 0,01136 MB pada Apriori. Hasil ini menunjukkan bahwa algoritma ECLAT umumnya lebih efektif untuk menganalisis transaksi penjualan di lingkungan minimarket, terutama saat menangani dataset besar dan efisiensi komputasi sangat penting. Temuan ini memberikan wawasan berharga dalam memilih algoritma yang tepat untuk mengoptimalkan strategi pemasaran dan manajemen inventaris di lingkungan ritel

Kata kunci: Analisa Keranjang Belanja, Apriori, Aturan Asosiasi, ECLAT

Abstract

This study compares the performance of the Apriori and ECLAT algorithms in analyzing sales transaction data from a minimarket. The research focuses on examining both algorithms' efficiency in terms of execution time and memory usage when identifying frequent itemsets and generating association rules. Given the limited variety of products sold in a minimarket, a lower minimum support (0.001) and minimum confidence (0.005) were applied to ensure meaningful results, as higher thresholds resulted in no significant findings. The first test evaluated the time required to find frequent itemsets, revealing that ECLAT consistently outperformed Apriori with an average execution time of 0.71634 seconds compared to Apriori's 4.88256 seconds. The second test assessed the time taken to generate association rules, where ECLAT again showed slightly better performance, averaging 0.01352 seconds versus Apriori's 0.01618 seconds. Memory usage tests showed that ECLAT was more efficient, using an average of 0.12436 MB to find frequent itemsets and 0.01052 MB to generate association rules, compared to Apriori's 0.1385 MB and 0.01136 MB, respectively. The results indicate that the ECLAT algorithm is generally more effective for analyzing sales transactions in a minimarket environment, particularly when handling large datasets and when computational efficiency is critical. The findings provide valuable insights for selecting the appropriate algorithm to optimize marketing strategies and inventory management in retail settings.

Keywords: Market Basket Analysis, Apriori, Association Rule, ECLAT

1. PENDAHULUAN

Analisis Keranjang Belanja (*Market Basket Analysis* atau MBA) adalah salah satu alat analitik yang penting dalam manajemen hubungan pelanggan (CRM) untuk memahami pola perilaku pembelian konsumen dan mengoptimalkan strategi pemasaran. Dalam memahami perilaku pembelian konsumen, hubungan antara aturan asosiasi dan MBA memainkan peran yang sangat penting di lingkungan ritel. MBA adalah aplikasi khusus dari association rule mining yang berfokus pada penemuan hubungan dan pola menarik antara produk-produk yang sering dibeli bersama dalam data transaksi. Tujuan utama dari MBA adalah untuk mengidentifikasi keterkaitan produk yang dapat digunakan untuk mengembangkan strategi pemasaran, penempatan produk, dan manajemen persediaan yang lebih efektif [1].

Aturan asosiasi adalah representasi formal dari hubungan yang ditemukan melalui MBA dan biasanya berbentuk pernyataan “jika-maka”. Dalam konteks ini, bagian “jika” (*antecedent*) menunjukkan satu atau lebih produk yang dibeli, sementara bagian “maka” (*consequent*) mengindikasikan kemungkinan pembelian produk lain. Misalnya, aturan asosiasi dapat mengungkapkan bahwa “jika seorang pelanggan membeli roti dan mentega, maka mereka juga cenderung membeli selai.” Aturan ini dihasilkan dari analisis data transaksi historis untuk menentukan kekuatan hubungan antarproduk[2]. Kekuatan dari aturan ini diukur dengan menggunakan metrik seperti *support*, *confidence*, dan *lift*, yang membantu peritel menilai signifikansi dan keandalan pola yang ditemukan[3][4]. Dengan mengidentifikasi produk mana yang sering dibeli bersama, peritel dapat merancang strategi penjualan silang yang lebih efektif, mengoptimalkan tata letak produk, dan menciptakan promosi yang lebih tepat sasaran sesuai preferensi pelanggan. Sebagai contoh, jika analisis menunjukkan bahwa pelanggan yang membeli popok juga sering membeli tisu basah, peritel dapat menempatkan kedua produk ini berdekatan atau menawarkan diskon pada tisu basah saat popok dibeli, untuk meningkatkan penjualan tambahan[5].

Data mining juga dapat diartikan sebagai proses menganalisis dan menemukan informasi yang berkaitan dengan database dalam cakupan yang luas[6]. Salah satu algoritma *data mining* yaitu Apriori, yang merupakan metode dasar dalam pencarian aturan asosiasi, telah banyak digunakan untuk menemukan pola dalam *dataset* berukuran besar. Namun, algoritma ini memiliki sejumlah kelemahan yang dapat secara signifikan mempengaruhi efisiensi dan efektivitasnya dalam aplikasi praktis. Salah satu kelemahan utama dari algoritma Apriori adalah kebutuhan untuk melakukan pemindaian berulang terhadap database. Algoritma ini memerlukan pemindaian menyeluruh terhadap *dataset* berkali-kali untuk mengidentifikasi frequent *itemsets*, yang dapat menyebabkan beban komputasi yang besar, terutama pada *dataset* yang sangat besar [7][8][9]. Kelemahan penting lainnya dari algoritma Apriori adalah penggunaan memorinya yang tinggi. Selama eksekusi, algoritma ini menghasilkan sejumlah besar *itemset* kandidat, yang dapat menyebabkan penggunaan memori yang berlebihan. Masalah ini menjadi sangat menonjol ketika berhadapan dengan *dataset* yang memiliki banyak item unik, karena jumlah *itemset* kandidat dapat berkembang secara eksponensial [7][10]. Akibatnya, algoritma ini mungkin kurang efektif untuk *dataset* yang sangat besar atau dalam situasi di mana sumber daya memori terbatas. Ketidakefisienan penggunaan memori ini juga dapat memperlambat waktu pemrosesan, yang dapat mempersulit analisis data berskala besar [11][12].

Algoritma *Equivalence Class Transformation* (ECLAT) adalah teknik yang populer untuk mencari *itemset* yang sering muncul dari basis data transaksi. Algoritma ini dikembangkan oleh Zaki dkk dan menggunakan format data vertikal, di mana setiap item dikaitkan dengan daftar ID transaksi (TID) yang menunjukkan transaksi-transaksi di mana item tersebut muncul [13]. Dengan representasi vertikal ini, perhitungan dukungan *itemset* dapat dilakukan dengan lebih efisien melalui operasi perpotongan pada daftar TID, sehingga mengurangi jumlah pemindaian basis data dibandingkan algoritma tradisional seperti Apriori[14]. ECLAT menerapkan strategi pencarian berbasis kedalaman (*depth-first search*), yang memungkinkannya untuk menjelajahi *itemset* secara mendalam dan cepat, sehingga sangat efektif untuk digunakan pada *dataset* yang besar dan padat [13]. Salah satu kelebihan utama dari ECLAT adalah kemampuannya untuk

mengolah *dataset* besar dengan penggunaan memori yang lebih sedikit dan kecepatan pemrosesan yang lebih tinggi, karena mengurangi kebutuhan untuk melakukan pemindaian basis data berulang kali [14].

Penelitian ini berfokus pada perbandingan algoritma Apriori dan ECLAT dalam konteks data penjualan minimarket, yang memiliki karakteristik berbeda dari data penjualan supermarket yang digunakan oleh Wahyuningsih dkk [15]. Meskipun penelitian yang dilakukan oleh Wahyuningsih dkk telah mengeksplorasi perbandingan kedua algoritma menggunakan data penjualan supermarket, penelitian tersebut lebih terfokus pada skala besar dengan jumlah transaksi dan item yang lebih tinggi serta variasi produk yang lebih luas. Penelitian ini menawarkan kebaruan dengan menguji kinerja Apriori dan ECLAT pada *dataset* minimarket, yang cenderung memiliki volume transaksi lebih kecil namun frekwensi kunjungan yang lebih tinggi dan jumlah item yang terbatas. Kondisi ini memungkinkan pengujian efektivitas dan efisiensi kedua algoritma dalam skala yang lebih kecil, yang lebih relevan untuk usaha ritel lokal dengan keterbatasan sumber daya komputasi. Dengan demikian, penelitian ini akan memberikan wawasan baru tentang bagaimana kedua algoritma ini berperforma dalam konteks bisnis minimarket, yang belum banyak dibahas dalam literatur sebelumnya, serta memberikan panduan praktis bagi pelaku bisnis ritel skala kecil dalam memilih algoritma yang tepat untuk analisis pola pembelian konsumen.

2. DASAR TEORI

2.1. Aturan Asosiasi

Aturan asosiasi pada *data mining* adalah teknik yang digunakan untuk menemukan hubungan, pola, atau korelasi menarik di antara kumpulan item dalam *dataset*, terutama dalam basis data transaksi. Metode ini banyak diterapkan di berbagai bidang, termasuk analisis keranjang belanja, di mana teknik tersebut membantu perusahaan retail memahami perilaku pembelian konsumen dengan mengidentifikasi item yang sering dibeli bersama [16]. Proses pencarian aturan asosiasi melibatkan dua langkah utama: pertama, mengidentifikasi *itemset* yang sering muncul dan memenuhi ambang batas *minimum support*, dan kedua, menghasilkan aturan asosiasi dari *itemset* tersebut yang memenuhi tingkat *minimum confidence*. *Support* mengukur seberapa sering sebuah *itemset* muncul dalam *dataset*, sedangkan *confidence* menunjukkan kemungkinan bahwa *consequent* dari aturan tersebut ada ketika *antecedent* juga ada. Kombinasi dari kedua metrik ini memungkinkan bisnis untuk menyaring aturan yang kurang signifikan dan fokus pada aturan yang memberikan wawasan yang dapat ditindaklanjuti [17], [18]. Rumus *support* dan *confidence* ditunjukkan pada Persamaan 1 dan Persamaan 2.

$$Support(A, B) = \frac{\sigma(A \cup B)}{N} \tag{1}$$

$$Confidence(A, B) = \frac{\sigma(A \cup B)}{A} \tag{2}$$

Keterangan:

- A : Antecedent
- B : Consequent
- N : Total Transaksi

2.2. Apriori

Algoritma Apriori adalah metode yang terkenal untuk mencari *itemset* yang sering muncul dan menghasilkan aturan asosiasi dalam basis data transaksi. Algoritma ini bekerja melalui proses sistematis yang terdiri dari beberapa tahap penting. Berikut adalah penjelasan rinci mengenai tahapan-tahapan tersebut [19], [20]

A. Menghasilkan Item Kandidat

Algoritma dimulai dengan mengidentifikasi semua item individu dalam *dataset* dan menghitung kemunculannya untuk menentukan item mana yang memenuhi ambang batas *support* minimum. Tahap ini menghasilkan set pertama dari *itemset* kandidat, yang biasanya terdiri dari item tunggal.

B. Menghitung *Support*

Untuk setiap *itemset* kandidat, algoritma menghitung *support*-nya, yaitu proporsi transaksi yang mengandung *itemset* tersebut. Ini dilakukan dengan memindai seluruh *dataset* dan menghitung berapa banyak transaksi yang mencakup setiap *itemset* kandidat. *Itemset* yang memenuhi atau melampaui ambang batas *support* minimum akan disimpan sebagai *itemset* yang sering muncul.

C. Proses Iteratif

Algoritma secara iteratif menghasilkan *itemset* kandidat yang lebih besar (C_k) dari *itemset* yang sering ditemukan pada iterasi sebelumnya (L_{k-1}). Ini dilakukan dengan menggabungkan *itemset* yang sering muncul untuk membentuk kandidat baru, kemudian menghitung *support*-nya kembali. Proses ini berlanjut hingga tidak ada lagi *itemset* yang sering dapat ditemukan

D. Menghasilkan Aturan Asosiasi

Setelah semua *itemset* yang sering diidentifikasi, algoritma akan menghasilkan aturan asosiasi dari *itemset* tersebut. Untuk setiap *itemset* yang sering muncul, algoritma menghitung tingkat *confidence* dari aturan tersebut, yang merupakan kemungkinan bahwa *itemset* konsekuen muncul dalam transaksi yang mengandung *itemset* antecedent. Aturan yang memenuhi ambang batas *confidence* minimum akan disimpan untuk analisis lebih lanjut.

2.3. Equivalence Class Transformation (ECLAT)

Algoritma ECLAT adalah metode populer untuk menemukan *itemset* yang sering muncul dalam basis data transaksi, algoritma ECLAT dikenal karena efisiensinya dan efektivitasnya dalam menangani *dataset* yang besar. Berbeda dengan algoritma Apriori yang menggunakan pendekatan pencarian berbasis lebar (*breadth-first search*), ECLAT menerapkan strategi pencarian berbasis kedalaman (*depth-first search*) dan bekerja dengan format data vertikal. Berikut adalah tahapan utama dari algoritma ECLAT, yang merinci proses dan fungsinya [21], [22]

A. Penelusuran *Depth-First Search* (DFS)

Algoritma ECLAT menggunakan pendekatan pencarian *depth-first* untuk mengeksplorasi *itemset*. Dimulai dengan item individu, algoritma secara rekursif menggabungkan item-item tersebut untuk membentuk *itemset* yang lebih besar. Pada setiap level penelusuran ini, algoritma menghitung *support* dari *itemset* yang sedang diproses dengan menginterseksi daftar TID (Transaction ID) dari item yang terlibat. Operasi interseksi ini merupakan keunggulan signifikan dari ECLAT, karena memungkinkan perhitungan yang efisien tanpa perlu memindai basis data berkali-kali, yang merupakan keterbatasan pada algoritma Apriori.

B. Menghitung *Support*

Selama penelusuran DFS, algoritma menghitung *support* untuk setiap *itemset* kandidat. *Support* didefinisikan sebagai proporsi transaksi yang mengandung *itemset* tertentu. Jika *support* dari suatu *itemset* memenuhi ambang batas *support* minimum yang telah ditentukan, *itemset* tersebut dianggap sebagai *itemset* yang sering muncul dan disimpan untuk analisis lebih lanjut.

C. Pemangkasan

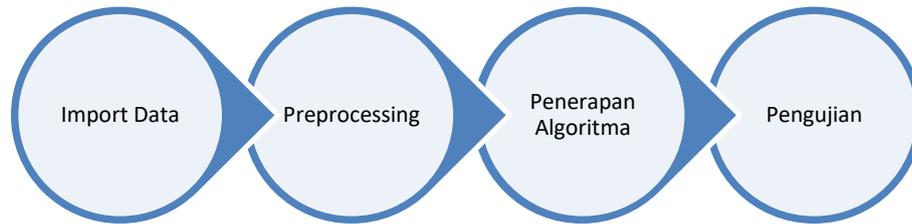
Setelah menghitung *support*, algoritma memangkas *itemset* kandidat yang tidak memenuhi persyaratan *support* minimum. Langkah ini mengurangi jumlah *itemset* yang perlu dipertimbangkan dalam iterasi berikutnya, sehingga meningkatkan efisiensi algoritma.

D. Menghasilkan Aturan Asosiasi

Setelah semua *itemset* yang sering muncul diidentifikasi, algoritma dapat menghasilkan aturan asosiasi dari *itemset* tersebut. Untuk setiap *itemset* yang sering muncul, algoritma menghitung tingkat *confidence* dari aturan, yang mengukur kemungkinan *itemset* konsekuen muncul dalam transaksi yang mengandung *itemset* antecedent. Aturan yang memenuhi ambang batas *confidence* minimum akan disimpan untuk analisis lebih lanjut.

3. METODOLOGI PENELITIAN

Metodologi penelitian yang dilakukan pada penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Metodologi Penelitian

3.1. Data Penelitian

Data yang digunakan dalam penelitian ini terdiri dari 8.453 baris data transaksi penjualan dengan total 24 Kolom yang ditunjukkan pada Tabel 1.

Tabel 1. Kolom Pada Tabel Transaksi Penjualan.

NAMA KOLOM	KETERANGAN
TRFLAG	Flag Transaksi
STORE	Kode Toko
BTYPENUM	Tipe Bisnis
TRXDATE	Tanggal Transaksi
REGNUM	Nomer Register
CASHNUM	ID Kasir
TRXNUM	Nomer Transaksi
TRXSEQNO	Urutan Transaksi
MDSEID	Kode Barcode
SKU	ID Produk
DESCRIPTOR	Nama Produk
DEPTNUM	ID Departement Produk
ITMQTY	QTY Produk
ITMPRICE	Harga Produk
PLUPRICE	Harga Barcode
DISCREASON	Alasan Diskon
DISCTYPE	Tipe Diskon
PROMOAMT	Nominal Promo
PROMOQTY	Qty Promo
GIFTFLAG	Flag Gift Produk
SALESPID	ID Promo
MEMBERPRICE	Harga Member
EMPLOYEEEDISC	Diskon Karyawan
AUTHID	ID Otorisasi

Penelitian ini, nilai *support* minimum sebesar 0,001 dan nilai *confidence* minimum sebesar 0,005 digunakan untuk membandingkan kinerja algoritma Apriori dan ECLAT. Pemilihan nilai parameter yang lebih rendah ini didasarkan pada karakteristik unik dari *dataset*, khususnya data penjualan minimarket. Dalam konteks minimarket, variasi produk yang dibeli

oleh pelanggan cenderung lebih sedikit dibandingkan dengan data penjualan dari supermarket atau pusat perbelanjaan besar. Hal ini mengakibatkan frekuensi kemunculan kombinasi produk tertentu dalam transaksi relatif lebih rendah [23]. Karena variasi produk yang terbatas, penggunaan nilai *support* dan *confidence* minimum yang lebih tinggi dapat menyebabkan tidak ada *itemset* atau aturan asosiasi yang memenuhi ambang batas yang ditetapkan. Misalnya, jika nilai *support* minimum terlalu tinggi, *itemset* yang terdiri dari produk yang sering dibeli bersama mungkin tidak muncul cukup sering dalam total transaksi untuk dianggap signifikan. Hal ini dapat menyebabkan keluaran algoritma kosong atau tidak relevan karena tidak ada pola yang terdeteksi [24].

3.2. Preprocessing Data

Tahap preprocessing data, kolom-kolom pada Tabel 1 yang tidak diperlukan untuk proses perhitungan pada algoritma dihilangkan. Kolom tersebut tidak digunakan karena pada proses perhitungan asosiasi hanya kolom *trxnum* dan *descriptor* yang diperlukan. Kolom *trxnum* digunakan untuk mengelompokkan barang dalam satu transaksi, sedangkan *descriptor* digunakan untuk mengidentifikasi barang-barang pada kelompok tersebut. Pada Gambar 2 dapat dilihat bahwa kolom yang tampil adalah *trxnum* dan *descriptor*. Data yang dimuat mencakup total 8.453 entri transaksi penjualan. Data ini merupakan kumpulan transaksi yang digunakan sebagai dasar untuk menganalisis pola pembelian.

	TRXNUM	DESCRIPTOR
0	202407152455	CHARM N WNG 35CM 6S
1	202407152457	CHIT AYAM BUMB 68 GR
2	202407152457	GOODAY AVOCADO PET
3	202407152457	GD ORI CAPPCNNO 250M
4	202407152458	SELADA KERITING 200
...
8449	20240731280	ULTRA UHT CKLT 250ML
8450	20240731280	ULTRA M STRW 250
8451	20240731281	FORTUNE POUCH 1LT
8452	20240731282	PEPAYA CL PTG MDM
8453	20240731282	VIXAL PORC BLUE 470M

8454 rows x 2 columns

Gambar 2. Data Hasil Preprocessing.

3.3. Penerapan Algoritma

Penelitian ini, digunakan nilai *min_support* sebesar 0.001 dan minimum *confidence* sebesar 0.005 untuk membandingkan kinerja algoritma Apriori dan ECLAT. Menggunakan *support* minimum sebesar 0,001 berarti suatu *itemset* akan dianggap sering muncul hanya jika muncul dalam setidaknya 0,1% dari total transaksi. Sementara itu, *confidence* minimum sebesar 0,005 berarti aturan asosiasi dianggap valid jika kemungkinan satu produk dibeli setelah produk lain setidaknya 0,5%. Hasil pencarian aturan asosiasi menggunakan algoritma Apriori dan Eclat ditunjukkan pada Gambar 3 dan Gambar 4.

```

Aturan Asosiasi Algoritma Apriori (diurutkan berdasarkan confidence terbesar):
{'GG MILD SHIVER 16'} -> frozenset({'TEREA BLACK GREEN'}) (Confidence: 100.00%)
{'RELX BBLMON BUZZ RED'} -> frozenset({'RELX BBLMON ALOE GRP'}) (Confidence: 100.00%)
{'RELX BBLMON ALOE GRP'} -> frozenset({'RELX BBLMON BUZZ RED'}) (Confidence: 100.00%)
{'MAESTRO WJN SNGRAI S'} -> frozenset({'MIX SLD MN DRESS 150'}) (Confidence: 83.33%)
{'TEREA EMERALD'} -> frozenset({'TEREA BLACK GREEN'}) (Confidence: 75.00%)
{'TEREA BLACK GREEN'} -> frozenset({'TEREA EMERALD'}) (Confidence: 54.55%)
{'TEREA BLACK GREEN'} -> frozenset({'GG MILD SHIVER 16'}) (Confidence: 45.45%)
{'ULTRA M STRW 250'} -> frozenset({'ULTRA UHT CKLT 250ML'}) (Confidence: 28.12%)
{'SAUDARA TL NGR 10S'} -> frozenset({'PSG CVNDH 500'}) (Confidence: 17.65%)
{'SEMANGKA MRH PTG MDM'} -> frozenset({'MIX SLD MN DRESS 150'}) (Confidence: 15.00%)
{'ULTRA UHT CKLT 250ML'} -> frozenset({'ULTRA M STRW 250'}) (Confidence: 11.11%)
{'PSG CVNDH 500'} -> frozenset({'SAUDARA TL NGR 10S'}) (Confidence: 10.00%)
{'MIX SLD MN DRESS 150'} -> frozenset({'SEMANGKA MRH PTG MDM'}) (Confidence: 9.38%)
{'PSG CVNDH 500'} -> frozenset({'MIX SLD MN DRESS 150'}) (Confidence: 8.33%)
{'MIX SLD MN DRESS 150'} -> frozenset({'MAESTRO WJN SNGRAI S'}) (Confidence: 7.81%)
{'MIX SLD MN DRESS 150'} -> frozenset({'PSG CVNDH 500'}) (Confidence: 7.81%)
    
```

Gambar 3. Hasil Pencarian Aturan Asosiasi Algoritma Apriori.

```

Aturan Asosiasi Algoritma ECLAT (diurutkan berdasarkan confidence terbesar):
{'RELX BBLMON BUZZ RED'} -> {'RELX BBLMON ALOE GRP'} (Confidence: 100.00%)
{'RELX BBLMON ALOE GRP'} -> {'RELX BBLMON BUZZ RED'} (Confidence: 100.00%)
{'GG MILD SHIVER 16'} -> {'TEREA BLACK GREEN'} (Confidence: 100.00%)
{'MAESTRO WJN SNGRAI S'} -> {'MIX SLD MN DRESS 150'} (Confidence: 83.33%)
{'TEREA EMERALD'} -> {'TEREA BLACK GREEN'} (Confidence: 75.00%)
{'TEREA BLACK GREEN'} -> {'TEREA EMERALD'} (Confidence: 54.55%)
{'TEREA BLACK GREEN'} -> {'GG MILD SHIVER 16'} (Confidence: 45.45%)
{'ULTRA M STRW 250'} -> {'ULTRA UHT CKLT 250ML'} (Confidence: 28.12%)
{'SAUDARA TL NGR 10S'} -> {'PSG CVNDH 500'} (Confidence: 17.65%)
{'SEMANGKA MRH PTG MDM'} -> {'MIX SLD MN DRESS 150'} (Confidence: 15.00%)
{'ULTRA UHT CKLT 250ML'} -> {'ULTRA M STRW 250'} (Confidence: 11.11%)
{'PSG CVNDH 500'} -> {'SAUDARA TL NGR 10S'} (Confidence: 10.00%)
{'MIX SLD MN DRESS 150'} -> {'SEMANGKA MRH PTG MDM'} (Confidence: 9.38%)
{'PSG CVNDH 500'} -> {'MIX SLD MN DRESS 150'} (Confidence: 8.33%)
{'MIX SLD MN DRESS 150'} -> {'PSG CVNDH 500'} (Confidence: 7.81%)
{'MIX SLD MN DRESS 150'} -> {'MAESTRO WJN SNGRAI S'} (Confidence: 7.81%)
    
```

Gambar 4. Hasil Pencarian Aturan Asosiasi Algoritma ECLAT.

4. PENGUJIAN DAN PEMBAHASAN

Penelitian ini, pengujian dilakukan untuk menentukan waktu pemrosesan dan penggunaan memori dari algoritma Apriori dan ECLAT dalam menemukan *itemset* dan aturan asosiasi pada transaksi penjualan minimarket. Pengujian ini penting karena kedua algoritma menggunakan pendekatan yang berbeda dalam analisis data, yang dapat mempengaruhi efisiensi dan kinerja dalam konteks data transaksi minimarket yang besar dan bervariasi. Dengan melakukan pengujian ini, memungkinkan pemilihan algoritma yang paling optimal untuk meningkatkan kecepatan analisis dan penggunaan sumber daya, sehingga mendukung keputusan bisnis yang lebih baik.

Pengujian pertama bertujuan untuk mengevaluasi waktu eksekusi algoritma Apriori dan ECLAT dalam menemukan *itemset* pada transaksi penjualan minimarket. Hasil pengujian waktu eksekusi untuk menemukan *itemset* menggunakan algoritma Apriori dan ECLAT ditunjukkan pada Tabel 2, dan Gambar 5 menggambarkan grafik perbandingan waktu eksekusi kedua algoritma tersebut.

Tabel 2. Waktu Proses Untuk Menemukan *Itemset*

Pengujian ke-	Apriori (s)	ECLAT (s)
1	4.015	0.7458
2	4.7296	0.7133
3	6.8376	0.702
4	4.6775	0.7168
5	4.1531	0.7038
Rata-rata	4.88256	0.71634



Gambar 5. Perbandingan Waktu Eksekusi Pencarian *Itemset*

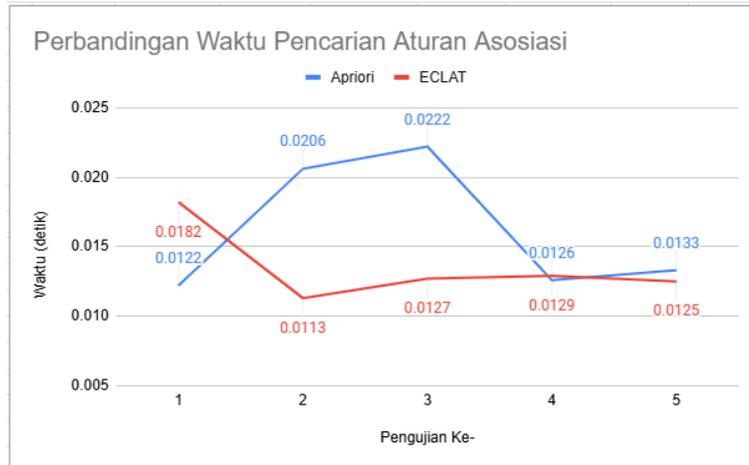
Tabel 2 menunjukkan hasil pengujian waktu eksekusi untuk algoritma Apriori dan ECLAT dalam menemukan *itemset* yang sering muncul dari *dataset* transaksi penjualan minimarket. Pengujian dilakukan sebanyak lima kali untuk masing-masing algoritma, dan waktu eksekusi diukur dalam satuan detik.

Pengujian pertama, algoritma Apriori membutuhkan waktu 4.015 detik, sedangkan algoritma ECLAT hanya memerlukan 0.7458 detik untuk menyelesaikan tugas yang sama. Dapat dilihat jika perbedaan waktu komputasi Algoritma ECLAT jauh lebih cepat dibandingkan dengan algoritma Apriori karena ECLAT menggunakan metode *dept-first search* dalam proses pencarian *itemset*. Begitu juga ada pengujian kedua, Apriori membutuhkan 4.7296 detik, sementara ECLAT hanya memerlukan 0.7133 detik. Pada pengujian ketiga, Apriori mencatat waktu eksekusi terlama, yaitu 6.8376 detik, sementara ECLAT tetap konsisten dengan waktu yang lebih singkat, yaitu 0.702 detik. Pengujian keempat dan kelima menunjukkan pola yang serupa, di mana Apriori membutuhkan 4.6775 dan 4.1531 detik, sementara ECLAT hanya memerlukan 0.7168 dan 0.7038 detik. Rata-rata waktu eksekusi untuk Apriori adalah 4.88256 detik, sedangkan rata-rata waktu eksekusi untuk ECLAT adalah 0.71634 detik. Hasil ini dengan jelas menunjukkan bahwa algoritma ECLAT secara konsisten lebih cepat daripada Apriori dalam menemukan *itemset* yang sering muncul dalam *dataset* transaksi minimarket.

Pengujian kedua mengevaluasi waktu eksekusi algoritma Apriori dan ECLAT dalam menemukan aturan asosiasi pada transaksi penjualan minimarket. Hasil pengujian kedua untuk kedua algoritma ditunjukkan pada Tabel 3, dan Gambar 6 menggambarkan grafik perbandingan waktu eksekusi kedua algoritma tersebut.

Tabel 3. Waktu Eksekusi Pencarian Aturan Asosiasi.

Pengujian ke-	Apriori (s)	ECLAT (s)
1	0.0122	0.0182
2	0.0206	0.0113
3	0.0222	0.0127
4	0.0126	0.0129
5	0.0133	0.0125
Rata-rata	0.01618	0.01352



Gambar 6. Perbandingan Waktu Eksekusi Pencarian Aturan Asosiasi.

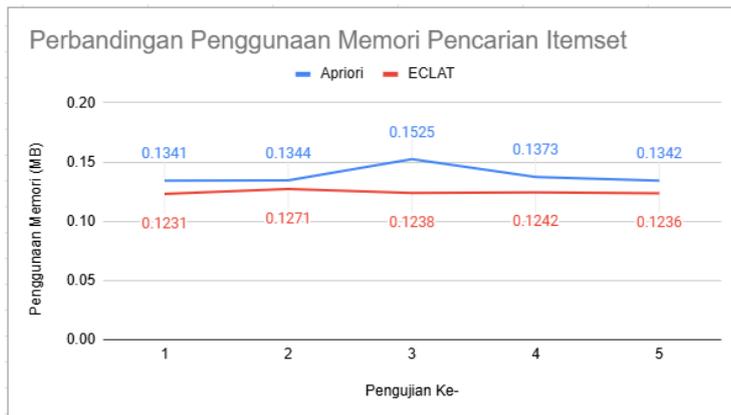
Tabel 3 menunjukkan hasil pengujian waktu eksekusi untuk algoritma Apriori dan ECLAT dalam menemukan aturan asosiasi dari *dataset* transaksi penjualan minimarket. Pengujian ini dilakukan sebanyak lima kali untuk masing-masing algoritma, dan waktu eksekusi diukur dalam satuan detik (s).

Pengujian pertama, algoritma Apriori membutuhkan waktu 0,0122 detik, sementara algoritma ECLAT memerlukan sedikit lebih lama, yaitu 0,0182 detik. Namun, pada pengujian kedua, ECLAT lebih cepat dengan waktu 0,0113 detik dibandingkan Apriori yang membutuhkan 0,0206 detik. Pola serupa terlihat pada pengujian ketiga, di mana Apriori memerlukan waktu 0,0222 detik, sedangkan ECLAT membutuhkan 0,0127 detik. Pada pengujian keempat, waktu eksekusi Apriori adalah 0,0126 detik, hampir sama dengan ECLAT yang mencatat 0,0129 detik. Pada pengujian kelima, Apriori mencatat waktu 0,0133 detik, sementara ECLAT sedikit lebih cepat dengan waktu 0,0125 detik. Secara rata-rata, algoritma Apriori membutuhkan waktu 0,01618 detik untuk menemukan aturan asosiasi, sedangkan ECLAT membutuhkan rata-rata 0,01352 detik. Meskipun terdapat beberapa variasi dalam hasil pengujian individu, secara keseluruhan, algoritma ECLAT cenderung sedikit lebih cepat daripada Apriori dalam menemukan aturan asosiasi. Namun, perbedaan rata-rata waktu eksekusi antara kedua algoritma ini relatif kecil, menunjukkan bahwa keduanya cukup efisien dalam proses ini.

Pengujian ketiga dalam penelitian ini melibatkan evaluasi penggunaan memori dalam menemukan *itemset* pada transaksi penjualan minimarket. Hasilnya dapat dilihat pada Tabel 4 dan Gambar 7.

Tabel 4. Penggunaan Memori Pencarian *Itemset*.

Pengujian ke-	Apriori (MB)	ECLAT (MB)
1	0.1341	0.1231
2	0.1344	0.1271
3	0.1525	0.1238
4	0.1373	0.1242
5	0.1342	0.1236
Rata-rata	0.1385	0.12436



Gambar 7. Perbandingan Penggunaan Memori Pencarian *Itemset*.

Tabel 4 menunjukkan hasil pengujian yang dilakukan sebanyak lima kali untuk masing-masing algoritma, dengan penggunaan memori yang diukur dalam megabyte (MB). Pada pengujian pertama, algoritma Apriori menggunakan 0,1341 MB memori, sementara ECLAT menggunakan memori lebih sedikit, yaitu 0,1231 MB. Pada pengujian kedua, penggunaan memori Apriori sedikit meningkat menjadi 0,1344 MB, sedangkan ECLAT mencatat penggunaan memori sebesar 0,1271 MB. Pengujian ketiga menunjukkan peningkatan penggunaan memori yang lebih besar untuk Apriori, yaitu 0,1525 MB, sementara ECLAT tetap lebih rendah di 0,1238 MB. Pada pengujian keempat dan kelima, penggunaan memori Apriori masing-masing adalah 0,1373 MB dan 0,1342 MB, sedangkan ECLAT menggunakan 0,1242 MB dan 0,1236 MB. Secara rata-rata, algoritma Apriori menggunakan 0,1385 MB memori, sementara ECLAT rata-rata menggunakan 0,12436 MB. Hasil ini menunjukkan bahwa algoritma ECLAT umumnya lebih efisien dalam penggunaan memori dibandingkan Apriori, yang mengindikasikan bahwa ECLAT membutuhkan sumber daya memori yang lebih sedikit untuk menyelesaikan tugas yang sama.

Tabel 5. Penggunaan Memori Pencarian Aturan Asosiasi.

Pengujian ke	Apriori (MB)	ECLAT (MB)
1	0.0113	0.0099
2	0.011	0.0099
3	0.0111	0.0107
4	0.0125	0.0106
5	0.0109	0.0115
Rata-rata	0.01136	0.01052



Gambar 8. Perbandingan Penggunaan Memori Pencarian Aturan Asosiasi.

Seperti pengujian ketiga, pengujian keempat yang ditunjukkan pada Tabel 5 dan Gambar 8 dilakukan sebanyak lima kali, dan penggunaan memori diukur dalam megabyte (MB). Pada pengujian pertama, algoritma Apriori menggunakan 0,0113 MB memori, sementara ECLAT

hanya menggunakan 0,0099 MB. Pada pengujian kedua, Apriori menggunakan 0,011 MB, masih lebih tinggi dibandingkan ECLAT yang menggunakan 0,0099 MB. Pada pengujian ketiga, penggunaan memori Apriori sedikit lebih tinggi, yaitu 0,0111 MB, sementara ECLAT menggunakan 0,0107 MB. Pada pengujian keempat, Apriori menggunakan lebih banyak memori, yaitu 0,0125 MB, sedangkan ECLAT membutuhkan 0,0106 MB. Pada pengujian kelima, penggunaan memori Apriori sebesar 0,0109 MB, sedikit lebih rendah daripada ECLAT yang mencatat 0,0115 MB. Hasil dari pengujian keempat konsisten dengan pengujian ketiga, menunjukkan bahwa algoritma ECLAT umumnya menggunakan lebih sedikit memori dibandingkan Apriori saat menemukan aturan asosiasi karena kemampuan algoritma ECLAT yang melakukan pencarian secara vertikal, meskipun terdapat beberapa variasi pada setiap pengujian. Secara keseluruhan, ECLAT menunjukkan efisiensi yang lebih baik dalam penggunaan memori.

5. KESIMPULAN

Hasil penelitian yang membandingkan algoritma Apriori dan ECLAT dalam analisis transaksi penjualan minimarket, ditemukan bahwa algoritma ECLAT secara konsisten menunjukkan performa yang lebih baik dibandingkan Apriori. ECLAT memiliki waktu eksekusi yang lebih cepat dalam menemukan *itemset* yang sering muncul dan menghasilkan aturan asosiasi, serta menggunakan lebih sedikit memori dalam kedua tugas tersebut. Hal ini sebagian besar disebabkan oleh pendekatan pencarian berbasis kedalaman (*depth-first search*) dan representasi data vertikal yang digunakan oleh ECLAT, yang memungkinkannya untuk melakukan operasi dengan lebih efisien dan mengurangi kebutuhan pemindaian *dataset* yang berulang. Sementara itu, algoritma Apriori, yang menggunakan strategi pencarian berbasis lebar (*breadth-first search*), membutuhkan lebih banyak waktu dan memori, terutama ketika jumlah *itemset* bertambah besar. ECLAT dapat membantu dalam analisis data transaksi yang lebih cepat dan hemat sumber daya, yang sangat berguna dalam lingkungan bisnis ritel dengan keterbatasan kapasitas komputasi atau ketika waktu respons yang cepat diperlukan untuk mendukung keputusan bisnis.

Penelitian di masa depan diharapkan dapat menerapkan konsep parallel programming pada algoritma ECLAT ataupun Apriori untuk mempercepat proses analisis data. Dengan memanfaatkan parallel programming, beberapa bagian dari algoritma dapat dijalankan secara bersamaan, yang berpotensi mengurangi waktu eksekusi secara signifikan, terutama ketika menangani *dataset* yang sangat besar. Penerapan teknik ini pada algoritma Apriori atau ECLAT diharapkan dapat membantu mengoptimalkan pemrosesan data yang kompleks dengan memanfaatkan banyak inti prosesor atau lingkungan komputasi terdistribusi. Hal ini diharapkan akan menghasilkan algoritma lebih skalabel dan efisien dalam situasi dunia nyata di mana kecepatan analisis data sangat penting untuk pengambilan keputusan yang cepat dan tepat.

DAFTAR PUSTAKA

- [1] M. Szymkowiak, T. Klimanek, dan T. Józefowski, "Applying *Market Basket Analysis* to Official Statistical Data," 2018. doi: 10.15611/eada.2018.1.03.
- [2] M.-B. Belaid, C. Bessière, dan N. Lazaar, "Constraint Programming for Association Rules," 2019. doi: 10.1137/1.9781611975673.15.
- [3] T. Arreeras, M. Arimura, T. Asada, dan S. Arreeras, "Association Rule Mining Tourist-Attractive Destinations for the Sustainable Development of a Large Tourism Area in Hokkaido Using Wi-Fi Tracking Data," 2019. doi: 10.3390/su11143967.
- [4] M. Shaikh, P. D. McNicholas, M.-L. Antonie, dan B. Murphy, "Standardizing Interestingness Measures for Association Rules," 2018. doi: 10.1002/sam.11394.
- [5] R. Yanti, J. N. Elquthb, I. P. Rachmadewi, dan Q. Qurtubi, "Bibliometric Study of Association Rule-*Market Basket Analysis*," 2024. doi: 10.11591/ijaas.v13.i2.pp282-290.
- [6] Ach. N. A. Wahid dan D. Avianto, "Penerapan Association Rule Terhadap Diagnosa Penyakit Menggunakan Algoritma Frequent Pattern Growth." Diakses: 12 November 2024. [Daring]. Tersedia pada: <https://journal.trunojoyo.ac.id/nero/article/view/22566>

-
- [7] Moch. Syahrir dan L. Z. A. Mardedi, "Determination of the Best Rule-Based Analysis Results From the Comparison of the Fp-Growth, Apriori, and TPQ-Apriori Algorithms for Recommendation Systems," 2023. doi: 10.31940/matrix.v13i2.52-67.
- [8] K. Chain, "Applications of Data Mining Algorithms for Network Security," 2019. doi: 10.20944/preprints201906.0144.v1.
- [9] A. Farouk, F. F. M. Ghaleb, M. Abdel-Rahman, dan W. Zakaria, "A New Algorithm for Mining Correct Sequences of a Specific Behaviour for Smart Monitoring Daily Life Activities," 2022. doi: 10.21608/ejaps.2022.168315.1045.
- [10] Y. Chen, R. Wang, B. Zeng, dan W. S. A. Griffith, "Temporal Association Rules Discovery Algorithm Based on Improved Index Tree," 2021. doi: 10.2478/amns.2021.1.00016.
- [11] B. Bouaita, A. Beghriche, A. Kout, dan A. Moussaoui, "A New Approach for Optimizing the Extraction of Association Rules," 2023. doi: 10.48084/etasr.5722.
- [12] A. L. S. Saabith, E. Sundararajan, dan A. A. Bakar, "A Parallel Apriori-Transaction Reduction Algorithm Using Hadoop-Mapreduce in Cloud," 2018. doi: 10.9734/ajrcos/2018/v1i124719.
- [13] J. A. Jusoh, M. Man, dan W. A. W. A. Bakar, "Performance of IF-Postdiffset and R-Eclat Variants in Large *Dataset*," 2018. doi: 10.14419/ijet.v7i4.1.28241.
- [14] P. Singh, S. Singh, P. Mishra, dan R. Garg, "RDD-Eclat: Approaches to Parallelize Eclat Algorithm on Spark RDD Framework (Extended Version)," 2021. doi: 10.21203/rs.3.rs-1079576/v1.
- [15] R. Wahyuningsih, A. Suharsono, dan N. Iriawan, "Comparison of *Market Basket Analysis* Method Using Apriori Algorithm, Frequent Pattern Growth (Fp- Growth) and Equivalence Class Transformation (Eclat) (Case Study: Supermarket 'X' Transaction Data for 2021)," 2023. doi: 10.33086/bfj.v8i2.5226.
- [16] S. Bhaskar, "Association Rule Development for Market Basket *Dataset*," 2018. doi: 10.5120/ijca2018917310.
- [17] L. Hamdad dan K. Benatchba, "Association Rules Mining," 2021. doi: 10.1007/s42979-021-00819-x.
- [18] S. Bagui dan P. C. Dhar, "Positive and Negative Association Rule Mining in Hadoop's MapReduce Environment," 2019. doi: 10.1186/s40537-019-0238-8.
- [19] K. D. Fernanda, A. P. Widodo, dan J. Lemantara, "Analysis and Implementation of the Apriori Algorithm for Strategies to Increase Sales at Sakinah Mart," 2023. doi: 10.30595/juita.v11i2.17341.
- [20] M. A. Zidan, N. R. Ismayanti, dan N. W. W. Sari, "Application of Web-Based Apriori Algorithm for Drug Inventory at Khairi Farma Pharmacy," 2022. doi: 10.47002/mst.v2i2.365.
- [21] F. Ren, Z. Pei, dan K. Wu, "Selection of Satisfied Association Rules via Aggregation of Linguistic Satisfied Degrees," 2019. doi: 10.1109/access.2019.2926735.
- [22] M. Man, J. A. Jusoh, S. I. A. Saany, W. A. W. A. Bakar, dan M. H. Ibrahim, "Analysis Study on R-Eclat Algorithm in Infrequent *Itemsets* Mining," 2019. doi: 10.11591/ijece.v9i6.pp5446-5453.
- [23] W. Lestari, H. Hasanah, dan R. Susanto, "Implementation of Association Rules Using Apriori Algorithm for Angkringan," 2023. doi: 10.47701/icohetech.v4i1.3425.
- [24] J. P. Ortiz, L. Rodríguez-Mazahua, J. Mejía, I. Machorro-Cano, G. Alor-Hernández, dan U. Juárez-Martínez, "Towards Association Rule-Based Item Selection Strategy in Computerized Adaptive Testing," 2020. doi: 10.16967/23898186.666.