

# Perbandingan Kinerja Algoritma Apriori dan Frequent Pattern Growth dalam Menemukan Pola Asosiasi

Fajrawati<sup>1,\*</sup>, Wawan Firgiawan<sup>2</sup>, Nahya Nur<sup>3</sup>

<sup>1</sup> Fakultas Teknik, Program Studi Informatika, Universitas Sulawesi Barat, Majene, Indonesia  
Email: <sup>1,\*</sup>fajrawati03@email.com, <sup>2</sup>wawanfirgiawan@unsulbar.com, <sup>3</sup>nahya.nur@unsulbar.ac.id

**Abstrak**– Dengan menggunakan dataset yang terdiri dari 500 transaksi dan 18 item produk, penelitian ini bertujuan untuk membandingkan kedua algoritma dalam penemuan pola asosiasi, Algoritma Apriori dan Frequent Pattern Growth (FP-Growth). Fokus penelitian adalah untuk menganalisis perbedaan algoritma dalam menghasilkan aturan asosiasi yang dapat digunakan secara terarah dan efisien dalam strategi pemasaran dan tata letak barang. Dalam kinerjanya Algoritma Apriori yang menggunakan metode *candidate generation*, memerlukan waktu yang lebih lama yaitu selama 0.9942 detik, sebab dalam pembentukan kandidat itemset dilakukan secara berulang dan memerlukan banyak proses *scanning* terhadap data transaksi. Proses iterasi yang tinggi ini membuat Apriori kurang efisien untuk dataset yang besar, namun dapat mengidentifikasi berbagai pola tersembunyi yang lebih rinci. Sementara itu memanfaatkan struktur *FP-Tree*, algoritma FP-Growth menghasilkan kumpulan itemset yang sering tanpa perlu membuat banyak kandidat. Ini mengurangi jumlah iterasi dan mempercepat proses penentuan aturan asosiasi, sehingga waktu yang dibutuhkan hanya 0.0184 detik jauh lebih efisien dibandingkan dengan Algoritma Apriori. Hasil pola asosiasi menunjukkan bahwa Algoritma Apriori lebih mampu menghasilkan lebih banyak aturan asosiasi, yaitu sebanyak 5 aturan, dibandingkan dengan Algoritma FP-Growth yang hanya menghasilkan satu aturan. Selain itu, Algoritma Apriori memiliki nilai *average confident* yang lebih tinggi 58.34% dibandingkan FP-Growth dengan nilai 52%. Hal ini mengindikasikan bahwa aturan yang dihasilkan oleh Algoritma Apriori memiliki tingkat kepercayaan yang lebih baik dari segi kekuatan hubungan antar item. Algoritma FP-Growth menghasilkan nilai *average Lift Rasio* yang lebih tinggi yaitu 3.32 dibandingkan Algoritma Apriori 0.023. Nilai lift rasio yang lebih besar dari satu menunjukkan bahwa aturan FP-Growth yang dihasilkan memiliki hubungan yang lebih kuat antara produk yang dianalisis. Secara keseluruhan, penelitian ini memberikan gambaran bahwa pemilihan algoritma yang tepat sangat bergantung pada tujuan analisis. Dengan memanfaatkan Algoritma Apriori, studi ini secara sistematis mengungkapkan pola hubungan antar item dapat menghasilkan tingkat kepercayaan yang tinggi, sedangkan FP-Growth mendapatkan hasil analisis yang lebih efisien dalam penggunaan waktu.

**Kata Kunci:** Algoritma Apriori, Algoritma FP-Growth, *Association Rule*

**Abstract**– Using a dataset consisting of 500 transactions and 18 product items, this study aims to compare the two algorithms in discovering association patterns, the Apriori Algorithm and Frequent Pattern Growth (FP-Growth). The focus of the study is to analyze the differences in algorithms in generating association rules that can be used in a targeted and efficient manner in marketing strategies and product layout. In its performance, the Apriori Algorithm using the candidate generation method takes a longer time, which is 0.9942 seconds, because the formation of candidate itemsets is done repeatedly and requires a lot of scanning processes for transaction data. This high iteration process makes Apriori less efficient for large datasets, but it can identify various more detailed hidden patterns. Meanwhile, utilizing the *FP-Tree* structure, the FP-Growth algorithm produces a collection of frequent itemsets without the need to create many candidates. This reduces the number of iterations and speeds up the process of determining association rules, so that the time required is only 0.0184 seconds, much more efficient than the Apriori Algorithm. The association pattern results show that the Apriori Algorithm is more capable of producing more association rules, which is 5 rules, compared to the FP-Growth Algorithm which only produces one rule. In addition, the Apriori Algorithm has a higher average confidence value of 58.34% compared to FP-Growth with a value of 52%. This indicates that the rules generated by the Apriori Algorithm have a better level of confidence in terms of the strength of the relationship between items. The FP-Growth Algorithm produces a higher average Lift Ratio value of 3.32 compared to the Apriori Algorithm 0.023. A lift ratio value greater than one indicates that the resulting FP-Growth rules have a stronger relationship between the products analyzed. Overall, this study provides an illustration that the selection of the right algorithm is highly dependent on the purpose of the analysis. By utilizing the Apriori Algorithm, this study systematically reveals that the relationship pattern between items can produce a high level of confidence, while FP-Growth obtains more efficient analysis results in terms of time usage.

**Keywords:** Apriori Algorithm, FP-Growth Algorithm, *Association Rule*

## 1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi di era saat ini semakin memberikan dampak perubahan yang besar dalam bidang bisnis baik itu dalam bidang perdagangan. Dalam perdagangan, persaingan bisnis sangat ketat sehingga diperlukan strategi pengelolaan bisnis yang matang. Dalam era persaingan pasar bebas dan kemajuan teknologi informasi yang pesat, teknologi memainkan peran penting dalam meningkatkan persaingan bisnis yang semakin ketat dan transparan, serta memungkinkan perusahaan untuk memenuhi kebutuhan dan tuntutan yang semakin meningkat dari pelanggan mereka[1]. Persaingan toko ritel atau swalayan dapat dipengaruhi dengan tata letak barang dalam toko tersebut. Dalam jangka panjang, efisiensi operasi ditentukan oleh tata letak yang mencakup desain bagian pusat kerja dan peralatan. Dalam hal ini, tata letak memiliki beberapa implikasi strategis karena dapat digunakan untuk menentukan prioritas persaingan bisnis seperti kapasitas, proses, fleksibilitas, dan biaya[2].

Penjualan langsung atau toko ritel modern seperti alfamart yang tersedia berbagai kebutuhan sehari-hari dan barang kebutuhan hidup, sekarang semakin mengacu pada data yang dihasilkan dari transaksi harian untuk meningkatkan efektivitas dan pelayanan. Seluruh transaksi yang dilakukan oleh konsumen memberikan jejak data yang sangat membantu untuk menemukan pola pembelian dalam perancangan stok barang, promosi produk dan penyusunan taktik penjualan yang lebih efektif. Data penjualan tidak hanya berfungsi sebagai arsip tapi juga sebagai laporan bagi perusahaan serta dapat dimanfaatkan menjadi informasi yang berguna untuk merancang strategi pemasaran, Namun demikian pengolahan data secara sederhana tidak akan mudah mendapatkan informasi yang diharapkan bila pengolahan data dalam jumlah yang besar.

Teknik data mining dapat digunakan untuk mengubah data transaksi penjualan menjadi informasi baru [3]. Untuk mempermudah transaksi dan memberikan informasi yang akurat kepada pengguna data, data mining diperlukan untuk mengelola jumlah data yang besar atau dari berbagai sumber yang ada. Ini adalah bagian dari proses pengelolaan data yang dikenal sebagai *warehousing* [4]. Salah satu metode untuk mendapatkan informasi tentang pola transaksi penjualan adalah pengolahan aturan asosiasi metode ini sangat berguna untuk menemukan pola asosiasi antara produk atau barang dominan yang dibeli oleh konsumen pada saat yang sama. Untuk memberikan informasi mengenai kemunculan item yang sering muncul secara bersamaan dalam setiap transaksi, toko ritel seperti alfamart dapat membuat keputusan yang efektif yaitu dengan menempatkan barang atau produk yang dominan dibeli oleh konsumen secara bersamaan. Salah satu algoritma data mining yang paling efektif untuk mengidentifikasi pola asosiasi dalam data transaksi adalah algoritma Apriori. Algoritma ini bekerja dengan mengidentifikasi item-item yang sering dibeli bersama dalam transaksi pembelian, sehingga membantu menetapkan aturan asosiasi antara item-item tersebut.[5]. Demikian pula, algoritma alternatif yang disebut Frequent Pattern Growth (FP-Growth) digunakan untuk menentukan kumpulan data yang paling sering muncul dalam set data[6].

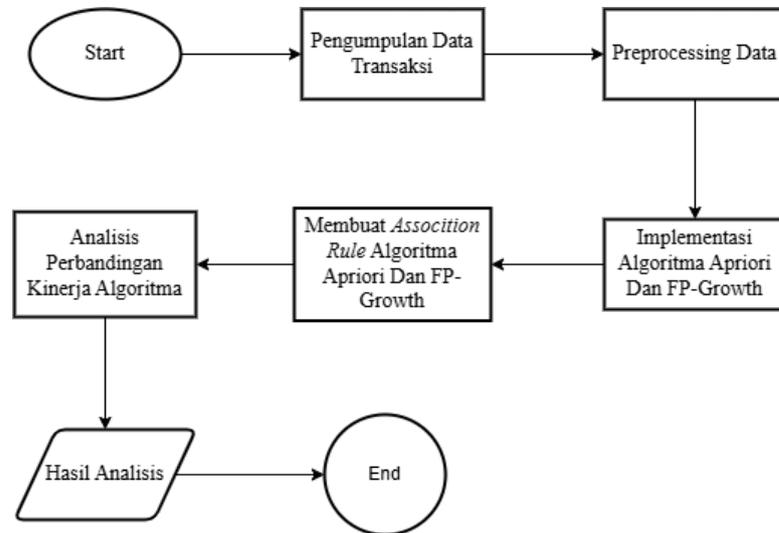
Pada Penelitian sebelumnya oleh ifa Musdalifah et al. [7] “Analisis Perbandingan Algoritma Apriori dan FP-Growth dalam Pembentukan Pola Asosiasi Keranjang Belanja Pelanggan” membahas pengujian dan perbandingan kinerja Algoritma Apriori dan FP-Growth dalam membedakan strategi penjualan yang lebih efektif dengan data transaksi penjualan. Penelitian lainnya oleh Gunawan Bayu Atmaja et al. [8] “Perbandingan Algoritma Apriori dan FP Growth pada Analisis Perilaku Konsumen terhadap Pembelian Data Elektronik” menjelaskan bahwa mengembangkan strategi pemasaran yang efektif memerlukan pemahaman tentang perilaku konsumen. Selain itu, penelitian ini bertujuan untuk membandingkan kinerja Algoritma FP-Growth dan Algoritma Apriori untuk mengidentifikasi pola hubungan dalam data penjualan di cabang Polewali Mandar dari Alfamart. Penelitian sebelumnya telah menunjukkan bahwa kedua algoritma ini lebih efisien. Kebaruan dari penelitian ini terletak pada penggunaan data transaksi yang diambil langsung di Alfamart dan diolah sendiri oleh peneliti, serta untuk melakukan evaluasi kinerja menggunakan beberapa parameter perbandingan, yaitu jumlah aturan, nilai *confidence*, *lift Ratio*, dan efisiensi waktu proses. Penelitian ini juga menyajikan pendekatan analisis berbasis merek produk, bukan kategori umum, yang memberikan sudut pandang berbeda dalam strategi pemasaran ritel.

Alfamart cabang Polewali Mandar yang terletak di jalan Ahmad Yani menghasilkan ratusan transaksi dalam setiap bulannya. Dari data transaksi tersebut memiliki potensi untuk diteliti agar mendapatkan informasi pola pembelian konsumen. Untuk itu perlu adanya perbandingan untuk mengetahui algoritma mana yang lebih efektif dalam mencari informasi dari pola transaksi penjualan. Oleh karena itu, penelitian ini berfokus pada perbandingan kinerja Algoritma Apriori dan FP-Growth dalam mendapatkan rekomendasi algoritma yang lebih efektif digunakan pada penanganan data transaksi yang besar dalam studi kasus pada Alfamart cabang Polewali Mandar.

## 2. METODE PENELITIAN

### 2.1 Tahapan Penelitian

Dalam tahapan penelitian, beberapa prosedur digunakan untuk mengumpulkan informasi baru tentang barang-barang yang sering dibeli konsumen. Untuk mencapai tujuan ini, algoritma Apriori dan FP-Growth digunakan untuk membandingkan pola asosiasi pada Alfamart. Gambar 1 menunjukkan tahapan penelitian secara keseluruhan.



Gambar 1. Tahapan Penelitian

Penelitian ini diawali dengan tahapan pengumpulan data transaksi, dimana data dikumpulkan secara primer. Selanjutnya *Preprocessing* data, yang berarti membersihkan dan mempersiapkan data untuk digunakan lebih lanjut. Setelah itu, Algoritma Apriori dan FP-Growth digunakan untuk menemukan pola asosiasi. Dari hasil implementasi dilakukan pembuatan *association rule* untuk mendapatkan pola pembelian barang serta menghasilkan pola keterkaitan antar produk. Tahap berikutnya adalah analisis perbandingan kinerja algoritma, dari hasil Algoritma Apriori dan FP-Growth dibandingkan berdasarkan parameter tertentu. Hasil dari analisis kemudian disusun, yang akan digunakan untuk menyimpulkan keunggulan dan kelemahan masing-masing algoritma. Setelah semua proses selesai, penelitian pun diakhiri.

### 2.2 Pengumpulan Data

Pengumpulan data adalah proses mencari data di lapangan yang akan dimanfaatkan untuk memberikan solusi terhadap permasalahan penelitian[9]. Untuk menyiapkan data awal yang diperlukan untuk penelitian, langkah ini merupakan persiapan. Data yang digunakan dalam penelitian ini berasal dari transaksi penjualan Alfamart, data primer ini berisi informasi tentang produk yang dibeli konsumen. Data diolah tanpa melibatkan identitas pelanggan untuk menjaga kerahasiaan informasi, serta dimana data tersebut diperoleh melalui kerjasama dengan pihak kasir dan bagian pencatatan transaksi penjualan. Total data yang digunakan dalam penelitian ini sebanyak 500 transaksi dengan jumlah *item* yang digunakan adalah 18 *item*. Jika produk terdapat dalam list transaksi, maka akan disimbolkan dengan angka 1, akan tetapi jika tidak ada maka akan disimbolkan dengan angka 0.

Tabel 1. Tabel Transaksi

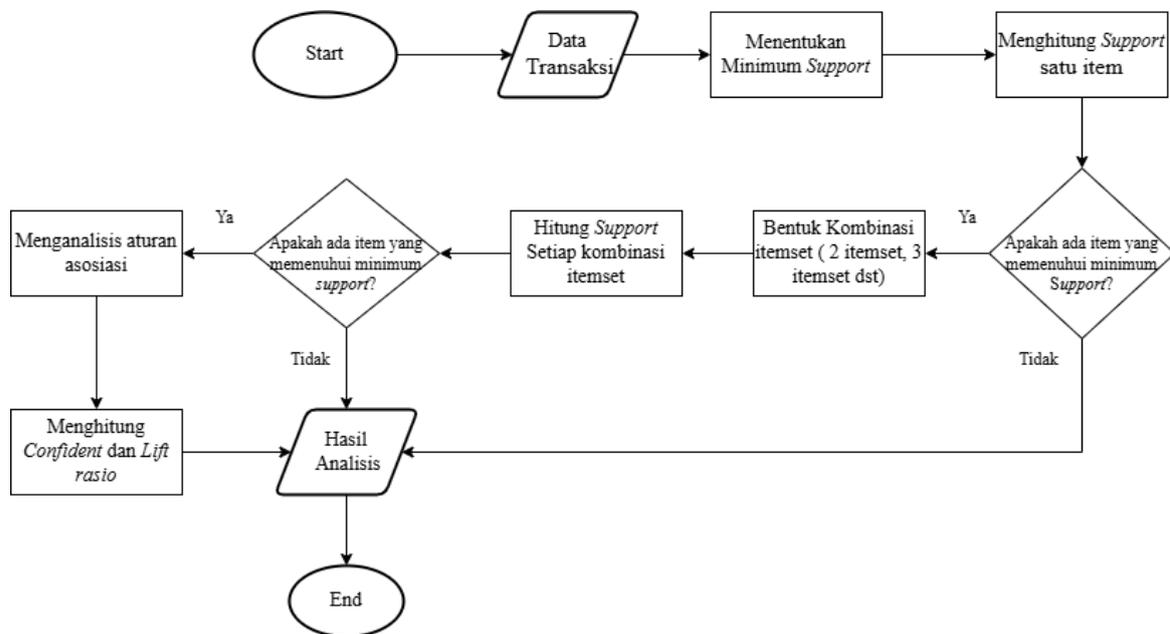
| Transaksi | Pucuk | Kanzler | Ultra | Pepsodent | Cimory | Aqua | Ichitan | Sempoerna | Abc | Sedap | Sunlight | Happy Tos | Sgrn | Ghico | Camel | Shinzui | Sunco | Indomie |
|-----------|-------|---------|-------|-----------|--------|------|---------|-----------|-----|-------|----------|-----------|------|-------|-------|---------|-------|---------|
| 1         | 0     | 1       | 1     | 0         | 0      | 0    | 0       | 0         | 0   | 0     | 0        | 0         | 0    | 0     | 0     | 0       | 0     | 1       |
| 2         | 0     | 1       | 1     | 0         | 0      | 0    | 0       | 0         | 0   | 0     | 0        | 0         | 0    | 0     | 0     | 0       | 0     | 1       |
| 3         | 0     | 0       | 1     | 0         | 0      | 0    | 0       | 0         | 0   | 0     | 0        | 0         | 0    | 0     | 0     | 0       | 0     | 1       |
| 4         | 0     | 0       | 1     | 0         | 0      | 1    | 0       | 0         | 0   | 0     | 1        | 0         | 0    | 0     | 0     | 0       | 0     | 0       |
| 5         | 0     | 0       | 1     | 0         | 0      | 0    | 0       | 0         | 0   | 0     | 0        | 0         | 0    | 0     | 1     | 0       | 0     | 0       |
| 6         | 0     | 0       | 1     | 0         | 0      | 0    | 0       | 0         | 0   | 0     | 0        | 0         | 0    | 1     | 0     | 0       | 0     | 1       |
| 7         | 0     | 0       | 1     | 0         | 0      | 0    | 0       | 0         | 0   | 0     | 0        | 0         | 0    | 0     | 0     | 0       | 0     | 1       |
| 8         | 0     | 1       | 0     | 0         | 1      | 0    | 0       | 0         | 0   | 0     | 0        | 0         | 0    | 1     | 0     | 0       | 0     | 1       |
| 9         | 0     | 0       | 0     | 1         | 0      | 0    | 0       | 0         | 0   | 0     | 1        | 0         | 1    | 1     | 0     | 0       | 0     | 0       |
| 10        | 0     | 0       | 0     | 0         | 1      | 0    | 0       | 0         | 0   | 0     | 0        | 0         | 0    | 0     | 0     | 0       | 0     | 0       |

|     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| :   | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : |
| :   | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : |
| 499 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 500 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

Tabel 1 menampilkan data transaksi sebanyak 500 data transaksi yang terdiri dari 18 produk yaitu Kanzler, Ultra, Pepsodent, Cimory, Aqua, Ichitan, Sampoerna, Abc, Sedap, Sunlight, Happy Tos, Sgm, Glico, Shinzui, Sunco dan Indomie. Penelitian ini memilih 18 item produk berdasarkan seberapa sering item tersebut muncul dalam data transaksi penjualan. Produk yang digunakan karena produk tersebut paling sering muncul selama periode pengambilan data, sehingga dapat menunjukkan pola belanja pelanggan di Alfamart Cabang Polewali Mandar. Selain itu, item dalam dataset ditampilkan sebagai merek produk daripada kategori umum, karena pelanggan di Alfamart cenderung membeli barang dengan merek tertentu. Misalnya, pelanggan membeli produk Pepsodent secara spesifik, bukan sekedar Pasta Gigi. Dengan cara ini, hasil analisis bisa memberikan pemahaman yang lebih praktis untuk menyusun strategi pemasaran. Meskipun jumlah total produk di Alfamart jauh lebih banyak, 18 item yang digunakan telah dipilih berdasarkan frekuensi tertinggi dari keseluruhan data transaksi.

### 2.3 Algoritma Apriori

Salah satu algoritma data mining yang paling umum digunakan untuk membuat sejumlah aturan asosiasi (*association rules*) adalah algoritma apriori [10]. Algoritma Apriori banyak digunakan dalam pencarian pola asosiasi untuk mengatur tata letak barang, sehingga banyak yang menggunakannya di bidang seperti ritel, *e-commerce*, dan logistik. Algoritma Apriori ini digunakan untuk menemukan pola hubungan antara satu objek dan objek lainnya. Data industri ritel seperti swalayan adalah jenis data transaksi yang paling sering digunakan oleh algoritma ini. Pengelola bisnis dapat menganalisis pembelian konsumen terhadap kombinasi produk yang dibeli secara bersamaan. Sebagai contoh, ilustrasi jika seorang konsumen membeli produk A dan B, ada kemungkinan mereka juga akan membeli produk C. Dengan demikian algoritma ini dapat membantu menemukan pola yang berpotensi untuk dimanfaatkan dalam strategi penjualan dan tata letak produk. Gambar 2 merupakan alur dari Algoritma Apriori.



Gambar 2. Diagram Flowchart Algoritma Apriori

Gambar 2 menggambarkan alur Algoritma Apriori dalam menganalisis data. Proses dimulai dengan memasukkan data transaksi dan menentukan nilai minimum *support*. Selanjutnya *support* satu itemset dihitung, jika terdapat item yang memenuhi minimum *support*, maka akan membentuk kombinasi itemset (2 itemset, 3 itemset, dst) dan menghitung *support* disetiap kombinasi. Kemudian menghitung nilai *confidence* dan *Lift Rasio* untuk setiap kombinasi yang memenuhi minimum *support*. Tahap tersebut dilakukan untuk mengevaluasi kekuatan antar item di setiap kombinasi.

Analisis asosiasi, juga dikenal sebagai *association rule mining* adalah metode data mining yang banyak digunakan oleh peneliti untuk mengembangkan algoritma yang efektif [11]. Dalam algoritma apriori pembentukan aturan asosiasi melibatkan dua tahapan atau proses utama yaitu *support* dan *confidence*. *Support*, atau nilai penunjang, menunjukkan persentase kombinasi item dalam database dan *confidence* atau nilai kepastian, menunjukkan kekuatan hubungan antar item dalam aturan asosiasi[12]. Pencarian pola atau acuan frekuensi yang tinggi adalah bagian dari analisis asosiasi, menghitung nilai *support* dan *confidence* memungkinkan untuk menentukan aturan asosiasi[13]. Berikut adalah langkah-langkah dalam penentuan nilai *support* satu item :

$$support = \frac{jumlah\ transaksi\ mengandung\ A}{Total\ transaksi} \times 100\% \quad (1)$$

Nilai *support* 2 item dapat dihitung menggunakan persamaan (2):

$$support = \frac{jumlah\ transaksi\ mengandung\ A\ dan\ B}{Total\ transaksi} \times 100\% \quad (2)$$

Namun, nilai *confidence* dapat dihitung dengan menggunakan persamaan (3):

$$Confidence = \frac{\sum\ transaksi\ A\ dan\ B}{\sum\ transaksi\ A} \times 100\% \quad (3)$$

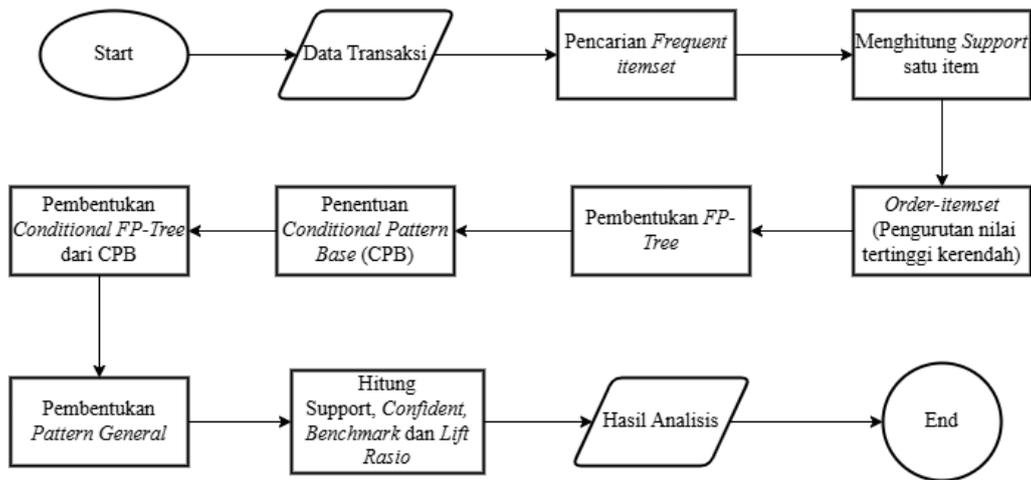
Pada tahap selanjutnya, aturan asosiasi diuji terhadap aturan yang telah dibuat sebelumnya. Ini dilakukan dengan melakukan perhitungan kembali menggunakan nilai *lift ratio* untuk menentukan kekuatan aturan[14]. Untuk menentukan *Lift Rasio* dapat menggunakan persamaan (4) sebagai berikut:

$$Lift\ Ratio = \frac{Support\ (A+B)}{support\ A * Support\ B} \quad (4)$$

## 2.4 Algoritma FP-Growth

Dengan tujuan yang sama seperti mining aturan asosiasi, FP-Growth adalah teknik data mining yang digunakan untuk menemukan pola item yang sering muncul dalam kumpulan transaksi. Ini adalah untuk menemukan hubungan atau keterkaitan antara item-item yang terdapat dalam transaksi [15]. Salah satu fungsi algoritma FP-Growth adalah mencari himpunan data yang sering muncul dalam dataset, juga dikenal sebagai *frequent itemset*. Salah satu ciri khas algoritma ini adalah bahwa itu dapat mendapatkan kumpulan itemset yang sering tanpa perlu mendapatkan kandidat yang dilakukan oleh algoritma Apriori. Ini dilakukan dengan menggunakan konsep pembuatan pohon *FP-Tree*. Algoritma FP-Growth dibagi menjadi empat tahapan utama, yaitu sebagai berikut :

1. Tahap Pembentukan *FP-Tree*  
Tahap pembentukan *FP-Tree* yaitu proses membentuk struktur pohon. Pembentukan *FP-Tree* melibatkan penyiapan dataset, pencarian frequent itemset, dan pengurutan itemset berdasarkan frekuensi tertinggi. Item dalam transaksi diurutkan dari nilai tertinggi ke rendah sebelum struktur pohon *FP-Tree* dibentuk untuk analisis lebih lanjut.
2. Tahap Penentuan *Conditional Pattern Base*  
Tahap penentuan *Conditional Pattern Base* (CPB) merupakan langkah yang dilakukan setelah langkah pembentukan *FP-Tree* dilakukan. Tahap ini menghasilkan hasil dari kumpulan jalur dari root ke node yang menunjukkan setiap item.
3. Tahap Pembentukan *Conditional FP-Tree*  
Tahap Pembentukan *Conditional FP-Tree* dibentuk dengan menggunakan kumpulan dari *Conditional Pattern Base* yang telah dibentuk. Item item yang tidak memenuhi minimum *support* akan dihapus. Proses setiap item dalam *FP-Tree* akan diulang sehingga terbentuk *Conditional FP-Tree* disetiap item
4. Tahap Pembentukan Pattern General  
Tahap Pembentukan Pattern General merupakan tahapan akhir dari penentuan *frequent itemset* (Pattern General) dari *Conditional FP-Tree* yang telah dibentuk sebelumnya, dan setiap kombinasi item yang telah memenuhi minimum *support* akan dihasilkan sebagai *frequent itemset*. Kemudian akan dilakukan pengujian untuk mendapatkan nilai *Lift Ratio*. Gambar 3 menunjukkan alur dari Algoritma FP- Growth.



Gambar 3. Flowchart Algoritma Fp-Growth

Gambar 3 menggambarkan alur FP-Growth Algoritma untuk mengidentifikasi pola yang sering terjadi dalam dataset. Proses dimulai dengan memasukkan data transaksi, kemudian mencari *frequent itemset* yang memenuhi minimum *support*. Untuk menghasilkan pola pembelian pelanggan, itemset diurutkan berdasarkan frekuensi kemunculannya sebelum membangun FP-Tree untuk menyimpan data transaksi. Tahap selanjutnya pembentukan *Conditional Pattern Base (CPB)* dan *Conditional FP-Tree* untuk mendapatkan informasi pola yang lebih spesifik. Selanjutnya pembentukan Pattern Geratel dari hasil yang dibentuk CPB dan FP-Tree. Kemudian menghitung *support*, *confidence*, *benchmark* dan *Lift Rasio* untuk mengidentifikasi hubungan yang sering terjadi dalam transaksi.

Salah satu metode untuk menemukan relasi atau korelasi antar himpunan item adalah *Association Rule*, di mana satu item memiliki hubungan dengan item lainnya. Fungsi aturan hubungan sering disebut sebagai *Market Basket Analysis*, dan tujuannya adalah untuk menemukan relasi atau korelasi antar himpunan item [16]. Untuk itu berikut adalah rumus Asosiasi yang digunakan Algoritma FP-Growth untuk mencari hubungan *item* di dataset :

Untuk mencari nilai *support* 1 item sama dengan penentuan nilai *support* pada Algoritma Apriori dapat dilihat pada persamaan (1), Pada tahap ini, agar mengetahui nilai *support* 2 itemset menggunakan persamaan (3). Begitupun dengan mencari *confident* dapat dilihat pada persamaan (4).

Berbeda dengan Algoritma Apriori, Algoritma FP-Growth memiliki rumus tambahan yaitu *Benchmark*. *Benchmark confidence* adalah perbandingan antara jumlah total transaksi dan jumlah semua item yang menjadi konsekuen. Persamaan benchmark menggunakan perhitungan yang sama dengan perhitungan nilai *support*, jadi dapat dikatakan bahwa benchmark adalah *support* untuk produk konsekuen[17]. Untuk menghitung nilai *benchmark confidence* digunakan rumus berikut:

$$Benchmark = \frac{Jumlah\ transaksi\ mengandung\ B}{Total\ Transaksi} \quad (8)$$

Berikut adalah rumus menghitung nilai *lift Ratio* :

$$Lift\ Ratio = \frac{Confident\ (A+B)}{Benchmark} \quad (9)$$

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pengujian Algoritma Apriori

##### a. Pengujian satu itemset

Proses pengujian dimulai dengan pembentukan satu itemset dengan mencari item yang sering muncul dalam transaksi serta mencari nilai *support* setiap item. Penentuan nilai minimum *Support* sebesar 10 pada penelitian ini dilakukan dengan mempertimbangkan jumlah total transaksi sebanyak 500 data. Nilai tersebut setara dengan 2% dari 500 total transaksi. Nilai ini dianggap optimal karena tidak terlalu tinggi sehingga tetap memungkinkan item dengan frekuensi sedang untuk dianalisis, namun juga tidak terlalu rendah agar dapat menghindari terbentuknya pola bersifat acak. Berikut adalah pengujian satu itemset.

**Tabel 2.** Tabel Satu itemset

| No | Item      | Jumlah | Support | Keterangan  |
|----|-----------|--------|---------|-------------|
| 1  | Pucuk     | 152    | 30.4    | Lulus       |
| 2  | Kanzler   | 178    | 35.6    | Lulus       |
| 3  | Ultra     | 126    | 25.2    | Lulus       |
| 4  | Pepsodent | 79     | 15.8    | Lulus       |
| 5  | Cimory    | 123    | 24.6    | Lulus       |
| 6  | Aqua      | 81     | 16.2    | Lulus       |
| 7  | Ichitan   | 81     | 16.2    | Lulus       |
| 8  | Sampoerna | 43     | 8.6     | Tidak Lulus |
| 9  | Abc       | 52     | 10.4    | Lulus       |
| 10 | Sedap     | 76     | 15.2    | Lulus       |
| 11 | Sunlight  | 73     | 14.6    | Lulus       |
| 12 | Happy Tos | 31     | 6.2     | Tidak Lulus |
| 13 | Sgm       | 105    | 21.0    | Lulus       |
| 14 | Glico     | 10     | 2.0     | Tidak Lulus |
| 15 | Camel     | 26     | 5.2     | Tidak Lulus |
| 16 | Shinzui   | 14     | 2.8     | Tidak Lulus |
| 17 | Sunco     | 19     | 3.8     | Tidak Lulus |
| 18 | Indomie   | 127    | 25.4    | Lulus       |

Pada Tabel 2 ada kemungkinan bahwa setiap item yang memenuhi nilai minimum *support* akan diberi label LULUS, sedangkan item yang memiliki nilai *support* kurang dari 10 akan diberi label TIDAK LULUS. Dari 18 item yang diuji, sebanyak 13 item yang dinyatakan lulus sementara 5 item lainnya dinyatakan tidak lulus atau dieliminasi dari pembentukan itemset selanjutnya.

- b. Pembentukan Lagrange Hasil Pengujian dari satu Itemset  
Setelah melakukan seleksi pada satu itemset, langkah selanjutnya adalah membentuk Tabel Lagrange, yang berisi item-item yang lolos seleksi minimum *support*.

**Tabel 3.** Lagrange Satu itemset

| No | Item      | Jumlah | Support |
|----|-----------|--------|---------|
| 1  | Pucuk     | 152    | 30.4    |
| 2  | Kanzler   | 178    | 35.6    |
| 3  | Ultra     | 126    | 25.2    |
| 4  | Pepsodent | 79     | 15.8    |
| 5  | Cimory    | 123    | 24.6    |
| 6  | Aqua      | 81     | 16.2    |
| 7  | Ichitan   | 81     | 16.2    |
| 9  | Abc       | 52     | 10.4    |
| 10 | Sedap     | 76     | 15.2    |
| 11 | Sunlight  | 73     | 14.6    |
| 13 | Sgm       | 105    | 21.0    |
| 18 | Indomie   | 127    | 25.4    |

Tabel 3 menampilkan data yang memenuhi nilai minimum *support* 10 maka akan diproses lanjut untuk pembentukan 2 itemset.

- c. Pembentukan Lagrange Hasil Pengujian dari 2 Itemset  
Langkah selanjutnya membentuk 2 itemset dari item-item yang lolos seleksi seperti tabel 3. 2 Itemset merupakan kumpulan dua item yang muncul bersamaan dalam transaksi atau biasa disebut dengan metode *candidate generation*. Dalam penelitian ini, proses pengujian tidak ditampilkan secara detail,

melainkan langsung menampilkan hasil akhir berupa Tabel Lagrange 2 itemset yang berisi kombinasi item yang lulus seleksi. Tabel 4 merupakan Lagrange 2 itemset yang dihasilkan.

**Tabel 4.** Lagrange 2 Itemset

| No | Item            | Jumlah | Support |
|----|-----------------|--------|---------|
| 1  | Pucuk, Kanzler  | 63     | 12.6    |
| 2  | Pucuk, Ichitan  | 51     | 10.2    |
| 3  | Kanzler, Ultra  | 71     | 14.2    |
| 4  | Kanzler, Cimory | 62     | 12.4    |
| 5  | Ultra, Indomie  | 51     | 10.2    |
| 6  | Pepsodent, Sgm  | 55     | 11.0    |

Pada tabel 4 menampilkan hasil pengujian 2 itemset, dinyatakan 6 itemset yang lolos maka akan dilanjutkan pada proses pembentukan 3 itemset.

d. Pengujian dengan 3 itemset

Pada langkah selanjutnya 3 itemset dibuat untuk menggabungkan itemset yang memenuhi syarat minimum *support* dari tahap pengujian 2 itemset.

**Tabel 5.** Lagrange 3 Itemset

| No | Item                      | Jumlah | Support | Keterangan  |
|----|---------------------------|--------|---------|-------------|
| 1  | Pucuk, Kanzler, ichitan   | 15     | 3.0     | Tidak Lulus |
| 2  | Pucuk, Kanzler, Ultra     | 14     | 2.8     | Tidak Lulus |
| 3  | Pucuk, Kanzler, Cimory    | 19     | 3.8     | Tidak Lulus |
| 4  | Pucuk, Kanzler, Indomie   | 15     | 3.0     | Tidak Lulus |
| 5  | Pucuk, Kanzler, Pepsodent | 2      | 0.4     | Tidak Lulus |
| 6  | Pucuk, Kanzler, Sgm       | 6      | 1.2     | Tidak Lulus |
| 7  | Pucuk, Ichitan, Ultra     | 6      | 1.2     | Tidak Lulus |
| 8  | Pucuk, Ichitan, Cimory    | 6      | 1.2     | Tidak Lulus |
| 9  | Pucuk, Ichitan, Indomie   | 10     | 2.0     | Tidak Lulus |
| 10 | Pucuk, Ichitan, Pepsodent | 7      | 1.4     | Tidak Lulus |
| :  | :                         | :      | :       | :           |
| :  | :                         | :      | :       | :           |
| 19 | Ultra, Indomie, Pepsodent | 2      | 0.4     | Tidak Lulus |
| 20 | Ultra, Indomie, Sgm       | 1      | 0.2     | Tidak Lulus |

Tabel 5 menunjukan bahwa sebanyak 20 data dalam membentuk 3 itemset tidak memenuhi nilai minimum *support* 10, dari data tersebut dinyatakan Tidak Lulus, karena tidak ditemukan kombinasi tambahan, proses pembentukan itemset dihentikan.

e. Aturan Asosiasi 2 itemset

Setelah melakukan pembentukan dan seleksi 2 itemset. Langkah selanjutnya adalah menghasilkan aturan asosiasi dari kombinasi item yang telah lulus. Aturan asosiasi merupakan gambaran hubungan antara dua itemset. Tabel 6 menunjukan hasil aturan asosiasi dari 2 itemset.

**Tabel 6.** Aturan Asosiasi 2 Itemset

| No | Item                                     | Jumlah Transaksi 2 itemset | Jumlah A | Confident |
|----|--|----------------------------|----------|-----------|
| 1  | Jika membeli Pucuk maka membeli Kanzler  | 63                         | 152      | 41.45     |
| 2  | Jika membeli Kanzler maka membeli Pucuk  | 63                         | 178      | 35.39     |
| 3  | Jika membeli Pucuk maka membeli Ichitan  | 51                         | 152      | 33.55     |
| 4  | Jika membeli Ichitan maka membeli Pucuk  | 51                         | 81       | 62.96     |
| 5  | Jika membeli Kanzler maka membeli Ultra  | 71                         | 178      | 39.89     |
| 6  | Jika membeli Ultra maka membeli Kanzler  | 71                         | 126      | 56.35     |
| 7  | Jika membeli Kanzler maka membeli Cimory | 62                         | 178      | 34.83     |
| 8  | Jika membeli Cimory maka membeli Kanzler | 62                         | 123      | 50.41     |
| 9  | Jika membeli Ultra maka membeli Indomie  | 51                         | 126      | 40.48     |
| 10 | Jika membeli Indomie maka membeli Ultra  | 51                         | 127      | 40.16     |
| 11 | Jika membeli Pepsodent maka membeli Sgm  | 55                         | 79       | 69.62     |
| 12 | Jika membeli Sgm maka membeli Pepsodent  | 55                         | 105      | 52.38     |

Tabel 6 merupakan hasil analisis semua frekuensi tinggi yang diperoleh dari hasil kombinasi 2 itemset dapat dilihat pada tabel 4, sehingga mendapat 12 pola aturan asosiasi yang memenuhi nilai minimum *confident* sebesar 50%. Aturan asosiasi A → B dihitung dengan memisahkan itemset menjadi dua bagian, yaitu *antecedent* yang disebut sebagai pendahulu merupakan penyebab suatu kejadian dan *consequent* disebut sebagai konsekuensi yang merupakan hasil dari peristiwa tersebut. Parameter Jumlah A dalam tabel merujuk pada total frekuensi pembelian item A dibeli sendiri. Hal ini dilakukan untuk mendapatkan semua kemungkinan asosiasi yang dapat terbentuk. Sebagai contoh, pada itemset {Pucuk, Kanzler}, konsumen yang membeli Pucuk memiliki kemungkinan untuk juga membeli Kanzler. Dalam hal ini, Pucuk berperan sebagai *antecedent*, sedangkan Kanzler berperan sebagai *consequent*. Dalam aturan asosiasi, arah hubungan digunakan untuk menentukan *antecedent* dan *consequent*, yang menunjukkan hubungan antara item yang sering muncul bersama dalam data transaksi.

f. Hasil Uji *Lift Ratio*

Tahap berikut adalah melakukan uji *Lift Ratio* untuk menilai kekuatan hubungan antara itemset. berikut adalah hasil uji lift dari aturan asosiasi yang telah bentuk.

**Tabel 7.** Hasil Uji *Lift Ratio*

| No | Jika                   | Maka              | Support | Confident | Lift Ratio |
|----|------------------------|-------------------|---------|-----------|------------|
| 1  | Jika membeli Ichitan   | Membeli Pucuk     | 10.2    | 62.96     | 0.02071    |
| 2  | Jika membeli Ultra     | Membeli Kanzler   | 14.2    | 56.35     | 0.01583    |
| 3  | Jika membeli Cimory    | Membeli Kanzler   | 12.4    | 50.41     | 0.01416    |
| 4  | Jika membeli Sgm       | Membeli Pepsodent | 11.0    | 69.62     | 0.03315    |
| 5  | Jika membeli Pepsodent | Membeli Sgm       | 11.0    | 52.38     | 0.03315    |

Berdasarkan Tabel 7 didapatkan hasil perhitungan pembentukan aturan asosiasi dan didapatkan 5 *rule asosiasi* yang memenuhi minimum *confidence* 50% yaitu jika membeli Ichitan maka membeli Pucuk *confident* 62.96%, Jika membeli Ultra maka membeli Kanzler 56.35%, Jika membeli Cimory maka

membeli Kanzler *confident* 50.41%, Jika membeli SGM maka membeli Pepsodent *confident* 69.62%, dan Jika membeli Pepsodent maka membeli SGM 52.38%.

Dalam aturan asosiasi, *lift ratio* adalah alat ukur penting untuk mengukur ketepatan dan keakuratan suatu alat ukur (*support* dan *confidence*) agar dapat diandalkan sepenuhnya. Dalam penelitian ini, *lift ratio* digunakan untuk menentukan apakah barang A dan B dibeli bersamaan [18]. Nilai Lift Ratio yang lebih besar dari satu menunjukkan bahwa aturan tersebut memiliki manfaat yang signifikan [19]. Jadi hasil yang didapatkan adalah *rule* jika membeli Ichitan maka membeli Pucuk *Lift Ratio* 0.02071, Jika membeli Ultra maka membeli Kanzler *Lift Ratio* 0.01583, Jika membeli Cimory maka membeli Kanzler *Lift Ratio* 0.01416, Jika membeli SGM maka membeli Pepsodent *Lift Ratio* 0.03315 dan membeli Pepsodent maka membeli SGM *Lift Ratio* 0.03315.

### 3.2 Pengujian Algoritma Fp-Growth

Untuk menghasilkan Association Rule, Algoritma FP-Growth melewati beberapa tahapan. Prosesnya sebagai berikut:

#### a. Pencarian *Frequent Item*

Proses diawali dengan menyiapkan dataset 500 transaksi. Dilanjutkan dengan mencari *Frequent Item* yang mencapai atau melebihi 10 minimum *support* . Dataset ini nantinya akan menjadi dasar untuk membangun *Frequent Pattern Tree (FP-Tree)* dan menentukan pola-pola yang sering muncul.

**Tabel 8.** Pencarian *Frequent Itemset*

| Item (Awal) | <i>Support Count</i> | Item      | Urutan <i>Frequent Item</i> Berdasarkan Prioritas |
|-------------|----------------------|-----------|---|
| Pucuk       | 152                  | Kanzler   | 178   |
| Kanzler     | 178                  | Pucuk     | 152   |
| Ultra       | 126                  | Indomie   | 127   |
| Pepsodent   | 79                   | Ultra     | 126   |
| Cimory      | 123                  | Cimory    | 123   |
| Aqua        | 82                   | Sgm       | 105   |
| Ichitan     | 81                   | Aqua      | 82  |
| Sampoerna   | 43                   | Ichitan   | 81  |
| Abc         | 52                   | Pepsodent | 79  |
| Sedap       | 76                   | Sedap     | 76  |
| Sunlight    | 73                   | Sunlight  | 73  |
| Happy Tos   | 31                   | Abc       | 52  |
| Sgm         | 105                  | Sampoerna | 43  |
| Glico       | 10                   | Happy Tos | 31  |
| Camel       | 26                   | Camel     | 26  |
| Shinzui     | 14                   | Sunco     | 19  |
| Sunco       | 19                   | Shinzui   | 14  |
| Indomie     | 127                  | Glico     | 10  |

Tabel 8 menunjukkan bahwa Kanzler memiliki *support count* tertinggi 178, disusul Pucuk dan Indomie. Item seperti Glico 10 dan Shinzui 14 muncul lebih sedikit namun tetap memenuhi minimum *support*.

#### b. Dataset Berdasarkan Prioritas

Setelah *Frequent Item* terbentuk, langkah berikutnya mengurutkan itemset berdasarkan prioritas, di mana urutan ditentukan dari nilai *support count* tertinggi ke terendah.

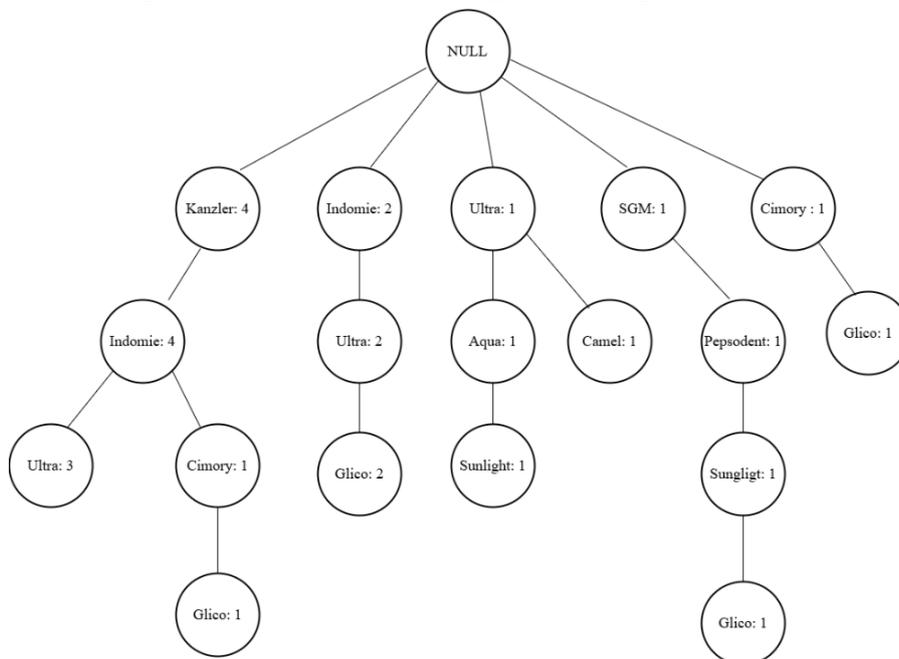
Tabel 9. Dataset Berdasarkan Prioritas

| Transaksi | Dataset Awal                 | Dataset Berdasarkan Prioritas |
|-----------|------------------------------|-------------------------------|
| 1         | Kanzler,Ultra,Indomie        | Kanzler,Indomie,Ultra         |
| 2         | Kanzler, Ultra,Indomie       | Kanzler, Indomie, Ultra       |
| 3         | Ultra, Indomie               | Indomie, Ultra                |
| 4         | Ultra,Aqua,Sunlight          | Ultra,Aqua,Sunlight           |
| 5         | Ultra,Camel                  | Ultra,Camel                   |
| 6         | Ultra,Glico, Indomie         | Indomie,Ultra,Glico           |
| 7         | Kanzler,Ultra,Indomie        | Kanzler,Indomie,Ultra         |
| 8         | Kanzler,Cimory,Glico,Indomie | Kanzler,Indomie,Cimory,Glico  |
| 9         | Pepsodent,Sunlight,Sgm,Glico | Sgm,Pepsodent,Sunlight,Glico  |
| 10        | Cimory,Glico                 | Cimory,Glico                  |
| ⋮         | ⋮                            | ⋮                             |
| ⋮         | ⋮                            | ⋮                             |
| 499       | Kanzler,Cimory               | Kanzler,Cimory                |
| 500       | Pucuk,Kanzler,Ultra,Glico    | Kanzler,Pucuk,Ultra,Glico     |

Pada Tabel 9 Dataset Awal diperoleh dari Tabel 1. Tabel Transaksi, dimana pada transaksi pertama terdapat pembelian yaitu Kanzler, Ultra, Indomie dst. Pembentukan Dataset Berdasarkan Prioritas Dapat dilihat pada Tabel 8 item Kanzler memiliki *support count* tertinggi yaitu 178, sedangkan Glico berada diurutan terbawah dengan *support count* 10. Pada Tabel 9 menunjukkan perbandingan antara dataset awal dan dataset yang telah diatur, tetapi pada transaksi pertama urutan item tidak berubah sebab Kanzler memiliki *support count* tertinggi dilanjutkan Indomie dan Ultra.

c. *Frequent Pattern Tree (FP-Tree)*

*FP-Tree* dibentuk setelah *Frequent Item* diurutkan berdasarkan *support count* tertinggi. Struktur pohon ini dimulai dari simpul akar (Null) dan disusun mengikuti dataset prioritas.



Gambar 4. *FP-Tree*

Pada Gambar 4, *FP-Tree* dibangun menggunakan sampel data sebanyak 10 data. Pembentukan cabang dengan melihat Tabel 8 pada transaksi pertama yaitu dengan *support count* tertinggi (Kanzler) akan

menjadi cabang pertama, maka pembentukan Node Anak yaitu item berikutnya dalam urutan prioritas (Indomie, Ultra dan lain lain) akan membentuk cabang dibawahnya. Proses tersebut dilakukan sampai Tree terbentuk menjadi 500 dataset. Misalnya pada Gambar 4 *FP-Tree* Kanzler memiliki jumlah kemunculan 4, sehingga menjadi Root (sampul utama) dengan cabang menuju Indomie 4, yang kemudian bercabang ke Ultra 3, Cimory 1. Dari Cimory terdapat cabang yaitu Glico. Begitupun dengan item yang lain, Root akan dibuat jika terdapat item yang belum ada Root yang dibuat sebelumnya dengan catatan memiliki *support count* tertinggi.

d. Pembangkitan *Conditional Pattern Base*

*Conditional Pattern Base* adalah tahap untuk mencari pola kemunculan item-item tertentu dalam itemset yang lain. Pada tahap ini, setiap item dalam *FP-Tree* dianalisis untuk menemukan pola kondisional yang membentuk dasar dari *Conditional FP-Tree*.

**Tabel 10.** *Conditional Pattern Base*

| Itemset   | <i>Conditional Pattern Base</i>                          |
|-----------|--|
| Kanzler   | {-}  |
| Pucuk     | {Kanzler: 63}  |
| Indomie   | {Pucuk : 15},{Pucuk, Indomie: 6},...{Kanzler: 36}        |
| Ultra     | {Indomie: 22 }, {Kanzler: 36}, ... {Pucuk, Kanzler: 12}  |
| Cimory    | {Pucuk: 10}, {Kanzler: 26 }, ... {Pucuk, Kanzler: 15 }   |
| Sgm       | {Pucuk: 13}, {Cimory,Kanzler: 5}, ... {Cimory: 9}        |
| Aqua      | {Pucuk: 8}, {Sgm: 9}, ... {Indomie: 7}                   |
| Ichitan   | {Aqua: 4},{Kanzler, Pucuk: 4}, ... {Pucuk: 17}           |
| Pepsodent | {Sgm: 34}, {Indomie, Sgm: 5}, ... {Indomie: 2}           |
| Sedap     | {Pucuk, Ichitan: 3}, ... {Cimory:9}, {Sgm:8}             |
| :         | ⋮  |
| :         | ⋮  |
| Shinzui   | { Ichitan:1}, {Abc: 3}, ... {Kanzler, Pucuk:1}           |
| Glico     | {Shinzui:1}, {Cimory:2 },...{Kanzler, Indomie, Cimory:1} |

Pada Tabel 10 menunjukkan bahwa setiap item seperti Pucuk dan Indomie memiliki *Conditional Pattern Base* berisi kombinasi item yang sering muncul bersamaan. Misalnya, Pucuk muncul bersama Kanzler sebanyak 63 kali, dan Indomie bersama Pucuk dalam beberapa kombinasi.

e. Pembangkitan *Conditional FP-Tree*

Dalam penelitian ini menggunakan tabel karena kondisi *FP-Tree* adalah kondisi yang dibuat dari *conditional pattern base* Bentuk yang dibangun dalam *FP-Tree* seharusnya dibangun menggunakan struktur pohon kecil dapat dilihat pada tabel 10.

**Tabel 11.** *Conditional FP-Tree*

| Itemset   | <i>Conditional FP-Tree</i>              |
|-----------|---|
| Kanzler   | -                                       |
| Pucuk     | < Kanzler: 63 >                         |
| Indomie   | < Pucuk: 15 > ... < Kanzler : 36>       |
| Ultra     | < Indomie: 22 > ... <Pucuk,Kanzler: 12> |
| Cimory    | < Pucuk: 10 > ... < Pucuk, Kanzler >    |
| Sgm       | < Pucuk: 13> ... < Cimory :9 >          |
| Aqua      | < Pucuk:9 > ... < Indomie: 7 >          |
| Ichitan   | < Aqua: 4 > ... < Pucuk: 17 >           |
| Pepsodent | < Sgm: 34 > ... <Indomie: 2>            |

|         |   |
|---------|---|
| ⋮       | ⋮   |
| ⋮       | ⋮   |
| Shinzui | <Ichitan: 1> ... <Kanzler, Pucuk:             |
| Glico   | <Shinzui: 1> ... <Kanzler, Indomie, Cimory:1> |

Tabel 11 Setiap item memiliki *Conditional FP-Tree* yang merepresentasikan pola kemunculan item-item lain secara bersamaan. Misalnya, untuk item Pucuk, *Conditional FP-Tree*-nya adalah <Kanzler: 63 >, yang berarti Pucuk muncul bersama Kanzler sebanyak 63 kali. Begitupun dengan itemset yang lain.

f. *Conditional Pattern Generated*

*Conditional Pattern Generated* adalah tahap dimana pola yang dihasilkan dari *Conditional FP-Tree* ditemukan dan disusun. Pada tahap ini kombinasi item yang sering muncul bersama dan memenuhi minimum *support* 10. Hasilnya akan digunakan untuk membentuk *Association Rule*.

**Tabel 12.** *Conditional Pattern Generated*

| Itemset   | <i>Conditional Pattern Generated</i> |
|-----------|--------------------------------------|
| Pucuk     | [ Kanzler ]                          |
| Indomie   | [Kanzler]                            |
| Ultra     | [Kanzler]                            |
| Pepsodent | [Sgm]                                |
| Cimory    | [Kanzler]                            |
| Aqua      | [Kanzler]                            |
| Ichitan   | [Pucuk]                              |
| Sampoerna | [Kanzler]                            |
| Abc       | [Pucuk]                              |
| Sedap     | [Cimory]                             |
| Sunlight  | [Pepsodent]                          |
| ⋮         | ⋮                                    |
| ⋮         | ⋮                                    |
| Sgm       | [Pucuk]                              |
| Ichitan   | [Pucuk]                              |

Pada Tabel 12 merupakan hasil dari kombinasi item-item yang sering muncul bersamaan dengan itemset utama dan memenuhi kriteria minimum *support*. Untuk item Pucuk, *Conditional Pattern Generated*-nya adalah Kanzler yang berarti Pucuk sering muncul bersama dengan Kanzler. Item Indomie, *Conditional Pattern Generated*-nya adalah Kanzler yang berarti Indomie sering muncul bersama dengan Kanzler, begitupun dengan itemset lainnya.

g. Aturan Asosiasi

Analisis aturan asosiasi adalah cara untuk mengidentifikasi keterkaitan antara dua atau lebih produk berdasarkan data transaksi. Metode ini menghasilkan metrik penting yaitu *Support*, *Confidence*, *Benchmark* dan *Lift Rasio* yang membantu dalam memahami kekuatan dan hubungan antar produk. Berikut adalah Aturan Asosiasi yang dihasilkan.

Tabel 13 Aturan Asosiasi

| No | Jika             | Maka              | Support | Confident | Benchmark | Lift Ratio |
|----|------------------|-------------------|---------|-----------|-----------|------------|
| 1  | Jika membeli Sgm | Membeli Pepsodent | 11      | 52        | 16        | 3.32       |

Pada Tabel 13 mengeluarkan aturan bahwa orang yang membeli Sgm juga akan membeli Pepsodent. Nilai *support* sebesar 11 menunjukkan bahwa kombinasi Sgm dengan Pepsodent terjadi 11% dari total seluruh transaksi. Nilai *confident* sebesar 52 menunjukkan 52% dari transaksi yang membeli Sgm melibatkan pembelian Pepsodent. Benchmark yang dihasilkan adalah 16% kemungkinan konsumen membeli Pepsodent secara acak. *Lift Rasio* sebesar 3.32 menunjukkan hubungan yang kuat antara pembelian kedua produk tersebut.

### 3.3 Perbandingan Kinerja Algoritma Apriori dan FP-Growth

Analisis perbandingan kinerja antara Algoritma Apriori dan FP-Growth dilakukan untuk mengukur efisiensi dan efektifitas kedua algoritma dalam menghasilkan aturan asosiasi. Beberapa parameter yang digunakan yaitu waktu eksekusi, jumlah aturan asosiasi yang dihasilkan, *average confident* dan *average Lift Rasio*.

Tabel 14. Perbandingan Kinerja

| Parameter Perbandingan         | Algoritma Apriori | Algoritma FP-Growth |
|--------------------------------|-------------------|---------------------|
| Waktu Eksekusi (Detik)         | 0.9942 detik      | 0.0184 detik        |
| Jumlah <i>Association Rule</i> | 5                 | 1                   |
| <i>Average Confident</i>       | 58.34             | 52                  |
| <i>Average Lift Ratio</i>      | 0.023             | 3.32                |

Pada tabel 14 terdapat perbandingan dari kedua Algoritma tersebut, Berdasarkan Tabel 13 terlihat bahwa Algoritma FP-Growth lebih cepat dalam waktu eksekusi, hanya memerlukan waktu 0,0184 detik, jauh lebih efisien dibanding dengan Algoritma Apriori yang memakan waktu 0,9942 detik. Namun pada jumlah aturan asosiasi yang dihasilkan Algoritma Apriori lebih unggul dengan menghasilkan 5 *rule*, sedangkan FP-Growth hanya mendapatkan satu *rule* saja. Hal ini membuktikan bahwa Algoritma Apriori lebih mampu menemukan lebih banyak pola hubungan dibanding dengan Algoritma FP-Growth. Dalam hal *Average Confidence*, Algoritma Apriori memiliki *Average Confident* lebih tinggi yaitu 58.34% dibandingkan dengan FP-Growth yang memiliki nilai 52 %. Ini menunjukkan bahwa *rule* yang dihasilkan Apriori memiliki tingkat kepastian yang lebih baik. Tapi dalam hal *Average Lift Ratio* FP-Growth unggul dengan nilai 3,32 dari pada Algoritma Apriori yang memiliki nilai 0.023. Hal membuktikan bahwa aturan yang dihasilkan FP-Growth memegang hubungan yang lebih kuat antara produk-produk yang dianalisis.

## 4. KESIMPULAN

Hasil penelitian menunjukkan bahwa kedua algoritma memiliki keunggulan dan kelemahan masing-masing, tergantung pada parameter yang digunakan untuk menilai kinerjanya. Algoritma Apriori terbukti lebih efektif dalam menghasilkan jumlah aturan asosiasi yang lebih banyak, yaitu sebanyak 5 aturan, dibandingkan dengan Algoritma FP-Growth yang hanya menghasilkan satu aturan. Hal ini menunjukkan bahwa Algoritma Apriori lebih mampu mengidentifikasi lebih banyak pola hubungan antar item. Selain itu, Algoritma Apriori juga memiliki nilai *average confident* yang lebih tinggi 58.34% dibandingkan FP-Growth 52%, yang mengindikasikan bahwa aturan yang dihasilkan oleh Algoritma Apriori memiliki tingkat kepercayaan yang lebih baik. Namun, dari segi waktu eksekusi, Algoritma FP-Growth jauh lebih efisien dengan waktu eksekusi hanya 0.0184 detik sedangkan Algoritma Apriori membutuhkan waktu 0.9942 detik. Ini menunjukkan bahwa FP-Growth lebih cepat dalam memproses data, terutama pada dataset yang besar. Dari segi hubungan antar item Algoritma FP-Growth mendapatkan nilai *average Lift Rasio* yang lebih tinggi 3.32 dibandingkan Algoritma Apriori 0.023. Nilai *lift Ratio* yang lebih besar dari satu menunjukkan bahwa aturan yang dihasilkan FP-Growth memiliki hubungan yang lebih kuat antara produk-produk yang dianalisis. Perbedaan mendasar dari kedua algoritma ini terletak pada pendekatan proses pencarian pola frekuensi. FP-Growth membuat struktur data pohon (*FP-Tree*) yang memungkinkan pencarian pola dilakukan lebih cepat dan efisien tanpa perlu membangkitkan kandidat itemset,

meskipun metode ini memakan waktu lebih lama dan menghasilkan hasil yang lebih besar. Apriori menggunakan pendekatan penciptaan kandidat itemset berulang-ulang, yang membuat proses lebih lama. Oleh karena itu, pemilihan algoritma yang tepat sangat bergantung pada kebutuhan analisis.

## REFERENCES

- [1] A. Anggrawan, M. Mayadi, and C. Satria, "Menentukan Akurasi Tata Letak Barang dengan Menggunakan Algoritma Apriori dan Algoritma FP-Growth," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 1, pp. 125–138, Nov. 2021, doi: 10.30812/matrik.v21i1.1260.
- [2] E. Munanda and S. Monalisa, "penerapan algoritma fp-growth pada data transaksi penjualan untuk penentuan tataletak barang 1," *Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi*, vol. 7, no. 2, pp. 173–184, 2021.
- [3] R. Takdirillah, "Edumatic: Jurnal Pendidikan Informatika Penerapan Data Mining Menggunakan Algoritma Apriori Terhadap Data Transaksi Sebagai Pendukung Informasi Strategi Penjualan," vol. 4, no. 1, pp. 37–46, 2020, doi: 10.29408/edumatic.v4i1.2081.
- [4] B. S. 2, S. Vita Rahayu1, "Analisis Algoritma Apriori dan FP-Growth Dalam Menemukan Pola *Frequent Item* Data Association Rule Pada Supermarket," pp. 20–20, 2021.
- [5] A. T. Gatot Soepriyono1, "Implementasi Data Mining dengan Algoritma Apriori dalam Menentukan Pola Pembelian Aksesoris Laptop," *MEDIA INFORMATIKA BUDIDARMA*, pp. 2088–2088, 2023.
- [6] Y. S. Putra, R. Kurniawan, and Y. A. Wijaya, "penerapan data mining menggunakan algoritma fp-growth pada data penjualan sembako," 2024.
- [7] A. J. Ifa Musdalifah1\*, "Analisis Perbandingan Algoritma Apriori Dan FP-Growth Dalam Pembentukan Pola Asosiasi Keranjang Belanja Pelanggan," p. 1, 2022.
- [8] G. Bayu Atmaja and R. Rachman, "perbandingan algoritma apriori dan fp-growth pada analisis perilaku konsumen terhadap pembelian data elektronik," *Jurnal Informatika Teknologi dan Sains*, vol. 7, no. 1, p. 307, 2025.
- [9] F. Rodzy *et al.*, "Implementasi Algoritma Apriori terhadap Kombinasi Pola Penjualan Kuliner Sidamanik Square," 2021.
- [10] S. Listanto and Y. Meisella Kristania, "implementasi data mining terhadap data penjualan dengan algoritma apriori pada pt. duta kencana swaguna," 2022. [Online]. Available: <https://ejurnal.teknokrat.ac.id/index.php/teknoinfo/index>
- [11] V. Syafira Rifania and E. Penulis Korespondensi, "Penerapan Algoritma Apriori Dalam Mencari Pola Pembelian Konsumen," vol. 2, no. 2, 2023, [Online]. Available: <https://ojs.trigunadharma.ac.id/index.php/jsi>
- [12] T. A. Lestari, R. Ramadhanti, and A. P. Windarto, "Implementasi Algoritma Apriori Terhadap Data Penjualan di Toko Roti Ganda Pematangsiantar," 2021. [Online]. Available: <https://hostjournals.com/bulletincsr>
- [13] C. Falisha Suardi and H. S. Y, "Implementasi Algoritma Apriori Untuk Analisis Data Transaksi Penjualan Pada Toko Berbasis Desktop," vol. 17, no. 1.
- [14] R. Takdirillah, "Penerapan Data Mining Menggunakan Algoritma Apriori Terhadap Data Transaksi Sebagai Pendukung Informasi Strategi Penjualan," *Edumatic : Jurnal Pendidikan Informatika*, vol. 4, no. 1, pp. 37–46, Jun. 2020, doi: 10.29408/edumatic.v4i1.2081.
- [15] Y. S. Putra, R. Kurniawan, and Y. A. Wijaya, "penerapan data mining menggunakan algoritma fp-growth pada data penjualan sembako," 2024.
- [16] D. Winarti *et al.*, "Penerapan Data Mining untuk Analisa Tingkat Kriminalitas Dengan Algoritma Association Rule Metode FP-Growth," *Jurnal SIMTIKA*, vol. 4, no. 3, 2021.
- [17] F. Achmad, O. Nurdiawan, and Y. A. Wijaya, "analisa pola transaksi pembelian konsumen pada toko ritel kesehatan menggunakan algoritma fp-growth," 2023. [Online]. Available: [www.researchgate.net](http://www.researchgate.net)
- [18] I. Made *et al.*, "Aplikasi Data Mining Asosiasi Barang Menggunakan Algoritma Apriori-TID," 2022.
- [19] A. N. Rahmi and A. Mikola, "implementasi algoritma apriori untuk menentukan pola pembelian pada customer (studi kasus : toko bakoel sembako)."