

Analisis Keranjang Belanja pada Data Ritel *Non-Toko* menggunakan Algoritma *FP-Growth*

I Dewa Ayu Indah Saraswati¹, I Made Agus Oka Gunawan^{2*}, I Made Agus Widiana Putra³
Sistem Informasi, Universitas Tabanan, Tabanan, Indonesia
*e-mail *Corresponding Author*: agusokagunawan@gmail.com

Abstract

Sales transaction data analysis is necessary to uncover relationships between items in the shopping basket, providing strategic insights for business decision-making. This study analyzes associations from a non-store retail sales transaction dataset using the CRISP-DM framework, processed in Google Colab with Python programming language. The model was built using the FP-Growth algorithm with various minimum support values. The results show that the model with a 1% (0.01) minimum support achieved the best balance, forming up to 4-itemsets and producing 954 association rules. The evaluation of the rules revealed a maximum confidence of 93.91% and a minimum of 9.39%, with a lift value > 1, indicating strong relationships between items that are frequently purchased together. These findings can be utilized for product layout arrangement and item recommendations, implemented through a web-based association analysis application.

Keywords: *FP-Growth Algorithm; Market Basket Analysis; CRISP-DM; Retail Data; Data Mining*

Abstrak

Analisis data transaksi penjualan diperlukan untuk mengungkap hubungan antar *item* dalam keranjang belanja, sehingga memberikan wawasan strategis bagi pengambilan keputusan bisnis. Penelitian ini menganalisis asosiasi dari *dataset* transaksi penjualan ritel *non-toko* melalui kerangka kerja CRISP-DM, yang diproses ke dalam *Google Colab* dengan bahasa pemrograman *Python*. Model dibangun menggunakan algoritma *FP-Growth* dengan berbagai nilai *minimum support*. Hasilnya, model dengan *minimum support* 1% (0,01) mencapai keseimbangan terbaik, membentuk hingga 4-*itemset* dan menghasilkan 954 aturan asosiasi. Evaluasi aturan asosiasi mengungkap nilai *confidence* maksimum 93,91% dan minimum 9,39%, dengan nilai *lift* > 1, yang menunjukkan hubungan kuat antar *item* yang sering dibeli bersamaan. Temuan ini dapat dimanfaatkan untuk pengaturan tata letak dan rekomendasi *item* yang diimplementasikan melalui aplikasi analisis asosiasi berbasis web.

Kata kunci: *Algoritma FP-Growth; Analisis Keranjang Belanja; CRISP-DM; Data Ritel; Penambangan Data*

1. Pendahuluan

Dalam era digital saat ini, perkembangan teknologi informasi dan komunikasi telah mengubah cara untuk berinteraksi, berbisnis, dan mengambil keputusan. Dunia yang semakin digital ditandai dengan pertumbuhan pesat data, termasuk dalam sektor *e-commerce*, yang telah menjadi salah satu pendorong utama pertumbuhan ekonomi global [1]. Melalui jaringan global ini, bisnis dapat dengan mudah mempromosikan produk barang dan jasa yang ditawarkan melalui situs web dan menyediakan fasilitas transaksi *online* yang efisien, serta membuka peluang baru untuk menjangkau pasar yang lebih luas. Konsumen juga dimudahkan dengan kemampuan untuk bertransaksi melalui komputer atau *smartphone* tanpa harus bertemu langsung dengan penjual [2]. Pada tahun 2023, penjualan *e-commerce* ritel global mencapai sekitar 5,8 triliun dolar AS, dengan proyeksi pertumbuhan mencapai 39% dalam beberapa tahun ke depan, diperkirakan akan melampaui delapan triliun dolar AS pada tahun 2027 [3]. Pertumbuhan ini didorong oleh meningkatnya minat dan kebutuhan konsumen terhadap berbagai barang yang ditawarkan oleh perusahaan ritel [4] yang akhirnya menghasilkan kumpulan data besar, seperti transaksi penjualan [5]. Menurut penelitian Jia Xu,

et al [6] kemampuan untuk mengelola dan memanfaatkan data telah menjadi keunggulan bersaing masa kini, di mana menurut penelitian Ifa Musdalifah, *et al* [7] perusahaan harus merancang berbagai strategi, termasuk memanfaatkan data transaksi untuk meningkatkan penjualan. Untuk itu, penting bagi organisasi dalam mengelola data dengan cermat, karena data dianggap sebagai aset yang berharga [8].

Eric D. Brown mengatakan bahwa saat ini kita tenggelam dalam data dan kelaparan akan pengetahuan [9]. Penting dipahami bahwa pada prinsipnya data memang perlu disimpan. Akan tetapi, lebih penting lagi ketika data yang dikumpulkan dari interaksi pelanggan dapat diolah menjadi informasi yang berguna. Proses ini melibatkan serangkaian langkah, mulai dari pengumpulan data, pembersihan data, analisis data, hingga interpretasi hasil analisis [2]. Namun, banyak perusahaan belum mengelola data dengan baik. Tanpa adopsi teknologi analitik, mereka akan tertinggal dalam persaingan [10]. terutama ketika berhadapan dengan volume data yang besar, tanpa analisis yang tepat dapat menyebabkan kesalahan dalam pengambilan keputusan, seperti penempatan produk yang tidak sesuai [11].

Analisis *dataset* memerlukan perhatian pada variabel-variabel yang relevan [4]. Sebagai hasil evolusi alami teknologi informasi, munculah *data mining* [12], yaitu proses penggunaan metode dan algoritma untuk mengekstraksi informasi penting dari data besar [13], [14]. Dalam proses ini, data dirangkum agar lebih mudah dipahami dan bermanfaat bagi pemilikinya [15]. Pemilihan metode asosiasi beserta algoritma *FP-growth* digunakan untuk menganalisis keranjang belanja guna menemukan kombinasi barang yang sering dibeli bersamaan oleh pelanggan pada *dataset* transaksi penjualan [16]. Dan Kehadiran integrasi antara tiga komponen utama, yakni *hardware*, *software*, dan *brainware* [17] sangatlah penting untuk memastikan kelancaran analisis.

Berdasarkan uraian yang telah dijabarkan, maka tujuan penelitian ini adalah membentuk pola kombinasi *item* untuk menghasilkan aturan asosiasi yang diperoleh dari model *FP-Growth*. Aturan ini berbentuk 'jika A maka B', di mana A disebut *antecedent*, yaitu kondisi atau *item* yang harus terpenuhi agar aturan berlaku, dan terletak di sisi kiri aturan. *Antecedent* mewakili pemicu atau penyebab dalam hubungan aturan. Sedangkan B disebut *consequent*, yaitu hasil atau *item* yang muncul sebagai konsekuensi jika *antecedent* terpenuhi, dan terletak di sisi kanan aturan. Dengan kata lain, *consequent* adalah efek atau hasil yang diharapkan setelah *antecedent* terjadi [18]. Manfaat dari temuan ini dapat digunakan untuk pengaturan tata letak dan rekomendasi *item*.

2. Tinjauan Pustaka

Penelitian yang dilakukan oleh Yusuf Husainb, *et al* [19] dengan membandingkan algoritma *Apriori*, *FP-Growth*, dan *Eclat* untuk menemukan pola pembelian konsumen di Apotek Sasameh Sehat. Tujuan penelitian ini adalah mengidentifikasi algoritma terbaik berdasarkan kecepatan eksekusi dan aturan (*rules*) yang dihasilkan menggunakan metode *association rules mining*. Parameter yang digunakan meliputi *minimum support*, *minimum confidence*, dan nilai *lift*. Hasil penelitian menunjukkan bahwa dalam hal kecepatan, algoritma *FP-Growth* adalah yang tercepat, dengan waktu eksekusi 0,00 detik pada kelima percobaan dengan berbagai nilai *support*. Lalu, diikuti algoritma *Apriori* dan *Eclat* dengan kecepatan yang cukup berimbang memiliki waktu eksekusi yang bervariasi di setiap percobaan, dengan perbedaan waktu sekitar 1 detik antara keduanya. Untuk jumlah *rule* yang dihasilkan, ketiga algoritma menghasilkan jumlah *rule* yang sama pada setiap percobaan. Penelitian serupa dilakukan oleh Dhea Rachmawati, *et al* [20] dengan membandingkan algoritma *Apriori* dan *FP-Growth* dalam menentukan pola penjualan pupuk di PT Pupuk Kujang. Hasil evaluasi pada grafik menunjukkan waktu pemrosesan sekitar $X \pm 30$ milidetik dimiliki algoritma *FP-Growth* dan $X \pm 45$ milidetik dimiliki algoritma *apriori* dengan *minimum support threshold* sebesar $Y=0,050$. Algoritma *FP-Growth* terbukti memiliki kecepatan eksekusi saat membentuk aturan asosiasi, dibandingkan dengan algoritma *Apriori*.

Penelitian yang dilakukan oleh Suardi Yakub, *et al* [21] dengan membangun sebuah sistem penerapan *data mining* menggunakan algoritma *Apriori* untuk mengoptimalkan tata letak barang di Toko Berkah Swalayan, yang bergerak di bidang ritel atau eceran, dengan tujuan meningkatkan strategi penjualan. Konsep penelitian ini didasarkan pada penggunaan *data mining* dengan algoritma *Apriori* untuk menemukan pola pembelian barang yang sering dibeli bersamaan oleh konsumen. Dengan menganalisis data transaksi, algoritma ini menghasilkan

aturan asosiasi yang memandu tata letak barang, yang kemudian diimplementasikan di toko melalui sistem. Parameter yang digunakan meliputi *minimum support* dan *minimum confidence*.

Penelitian oleh Andi Akram Nur Risal, et al [22] melakukan analisis pada data penjualan *retail online* menggunakan algoritma *Apriori* dalam teknik *data mining* sebagai salah satu cara untuk meningkatkan penjualan. Dengan tujuan, untuk menemukan aturan asosiasi antara produk-produk yang paling sering dibeli bersamaan oleh pelanggan. Penelitian ini juga menggunakan perbandingan berbagai parameter *minimum support* dan *confidence*, serta nilai *lift*. Hasil penelitian disajikan dalam bentuk tabel yang memuat aturan asosiasi antara produk-produk tersebut.

Penelitian yang dilakukan oleh Firmansyah, et al [23] bertujuan untuk mengembangkan strategi promosi yang lebih efektif di toko ritel Gramedia Matraman, yang menjual buku, alat tulis, alat edukasi, dan perlengkapan sekolah. Fokus penelitian ini adalah menganalisis pola pembelian buku oleh pelanggan dengan menggunakan metode *Market Basket Analysis* (MBA). Dalam implementasi *data mining*, penelitian ini menggunakan *framework* CRISP-DM, di mana pada tahap *modelling* digunakan algoritma *FP-Growth*, yang dipilih karena lebih cepat dalam memproses data besar. Proses ini dilakukan menggunakan aplikasi *Rapid Miner* dengan parameter *minimum support* dan *minimum confidence*. Hasil penelitian ini disajikan dalam bentuk tabel yang memuat 7 aturan asosiasi (*rules*) yang direkomendasikan untuk mendukung strategi promosi produk. Penelitian lainnya didukung oleh Didi Supriyadi, et al juga menerapkan *association rule mining* berbasis algoritma *FP-Growth* untuk memberikan rekomendasi penjualan pada data ritel *online* sebagai strategi pemasaran. Penggunaan algoritma *FP-Growth* lebih menguntungkan, karena hanya memerlukan satu atau dua kali *scanning database*, berbeda dengan algoritma *Apriori* yang memerlukan *scanning* berulang kali. Parameter yang digunakan meliputi *minimum support*, *minimum confidence*, dan nilai *lift* [4].

Merangkum penelitian sebelumnya, penelitian ini juga menganalisis keranjang belanja dari *dataset* transaksi penjualan ritel *non-toko* menggunakan metode *association rule mining* melalui kerangka kerja CRISP-DM, yang diproses ke dalam *tools Google Colab*. Model asosiasi pada penelitian ini dibangun dengan algoritma *FP-Growth* dengan parameter *minimum support*, kemudian hasil aturan asosiasi dievaluasi dengan parameter *confidence* dan *lift*. Perbedaannya terletak pada implementasi hasil analisis dalam bentuk aplikasi berbasis web yang dibangun menggunakan *library Python Streamlit* versi 1.36.0 melalui *Visual Studio Code* dan di-*hosting* menggunakan *tools Streamlit Share yang terintegrasi dengan repository GitHub*. Menurut penelitian Suryawan [24] penerapan *deployment* berbasis web dapat meningkatkan aksesibilitas bagi pengguna.

3. Metodologi

3.1. Algoritma *FP-Growth*

FP-Growth adalah algoritma populer dalam penambangan pola asosiasi yang mengatasi kelemahan *Apriori*. Dengan menggunakan struktur *FP-Tree*, algoritma ini memungkinkan penambangan pola frekuensi tanpa perlu menghasilkan kandidat *itemset*, sehingga lebih efisien dan mempercepat waktu pemrosesan [25], [26]. Formula yang digunakan dalam mencari hasil analisis [27], sebagai berikut:

1) Nilai *minimum support* digunakan untuk mencari kombinasi *item* yang sering muncul dalam *dataset* transaksi penjualan, seperti yang ditunjukkan pada persamaan (1).

$$\text{Support (A)} = \frac{\text{Jumlah transaksi mengandung A}}{\text{Total transaksi}} \times 100\% \quad \dots\dots\dots (1)$$

2) Setelah terbentuk pola kombinasi *item*, maka aturan asosiasi diperoleh. Nilai *confidence* digunakan untuk mengukur kekuatan aturan asosiasi, yaitu seberapa besar kemungkinan *item* B muncul dalam transaksi yang sudah mengandung *item* A, seperti ditunjukkan pada persamaan (2).

$$\text{Confidence} = P(B|A) = \frac{\sum \text{Transaksi mengandung A dan B}}{\text{Transaksi mengandung A}} \times 100\% \quad \dots\dots\dots (2)$$

3) Setelah nilai *confidence* dihitung, nilai *lift* digunakan untuk mengukur kevalidan aturan, seperti yang ditunjukkan pada persamaan (3).

$$Lift\ rasio = \frac{Confidence(A,B)}{Benchmark\ confidence(A,B)} \dots\dots\dots (3)$$

Keterangan:

$$Benchmark\ confidence = \frac{Nc(\sum Transaksi\ mengandung\ consequent)}{N(\sum Total\ Transaksi)}$$

3.2. Data dan Parameter

Dataset yang digunakan bersifat *open sources* yang bersumber dari situs web resmi UCI (*University of California Irvine Machine Learning Repository*). *Dataset* ini berisi transaksi penjualan ritel *non-toko* di Inggris, yang berlangsung dari tanggal 1 Desember 2010 hingga 9 Desember 2011. Penjualan dilakukan melalui media selain toko fisik, dengan *items* yang dijual berupa dekorasi dan aksesoris unik [28]. Pada tahapan model, parameter input yang diproses meliputi berbagai nilai *minimum support*, yaitu 0,09% (0,0009), 0,5% (0,005), 1% (0,01), 1,5% (0,015), 2% (0,02), dan 2,5% (0,025), untuk membentuk pola kombinasi antar *item* guna memperoleh aturan asosiasi. Setiap aturan kemudian dievaluasi berdasarkan parameter *confidence* untuk mengukur kekuatan hubungan antar *item*, serta parameter *lift* untuk menilai validitas aturan.

No	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
1	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850	United Kingdom
2	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850	United Kingdom
3	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850	United Kingdom
4	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850	United Kingdom
5	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850	United Kingdom
.....
541905	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/9/2011 12:50	0.85	12680	France
541906	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	12/9/2011 12:50	2.1	12680	France
541907	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	12/9/2011 12:50	4.15	12680	France
541908	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	12/9/2011 12:50	4.15	12680	France
541909	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12/9/2011 12:50	4.95	12680	France

Gambar 1. Tampilan *Dataset* Transaksi Penjualan Ritel *Non-Toko*

Jumlah *dataset* yang akan diuji coba adalah 541.909 baris data transaksi × 8 kolom variabel. Jumlah ini masih termasuk transaksi duplikat yang belum dihitung secara unik berdasarkan variabel *InvoiceNo*. seperti yang terlihat pada Gambar 1.

3.3. Metode Pengembangan Analisis Data

Penelitian ini menggunakan *framework* CRISP-DM (*CRoss-Industry Standard Process for Data Mining*), sebagai standar umum dalam proyek penambangan data, analitik, dan ilmu data [29]. CRISP-DM terdiri dari enam tahapan pengembangan, antara lain:

- 1) Pemahaman bisnis (*business understanding*), tahap ini berfokus pada pemahaman tujuan dan kebutuhan bisnis yang ingin dicapai melalui analisis data.
- 2) Pemahaman data (*data understanding*), setelah memahami tujuan bisnis, langkah selanjutnya adalah mengumpulkan dan mempelajari data yang tersedia.
- 3) Persiapan data (*data preparation*), pada tahap ini, data yang telah dipahami disiapkan untuk analisis lebih lanjut.
- 4) Pemodelan (*modeling*), setelah data disiapkan, langkah berikutnya adalah membangun model analitik yang akan digunakan untuk menganalisis data.
- 5) Evaluasi (*evaluation*), setelah model dibangun, evaluasi akan dilakukan menggunakan parameter *confidence* dan *lift*.
- 6) Penyebaran (*deployment*), Tahap terakhir adalah penyebaran model yang telah dievaluasi ke dalam lingkungan bisnis.

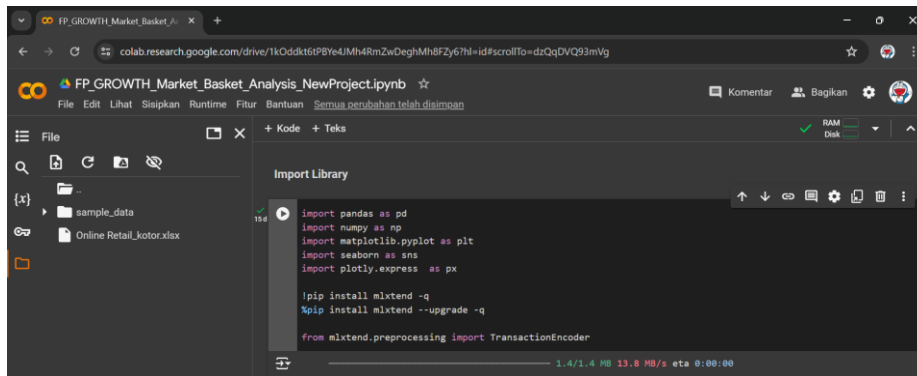
4. Hasil dan Pembahasan

4.1. Pemahaman Bisnis (*Business Understanding*)

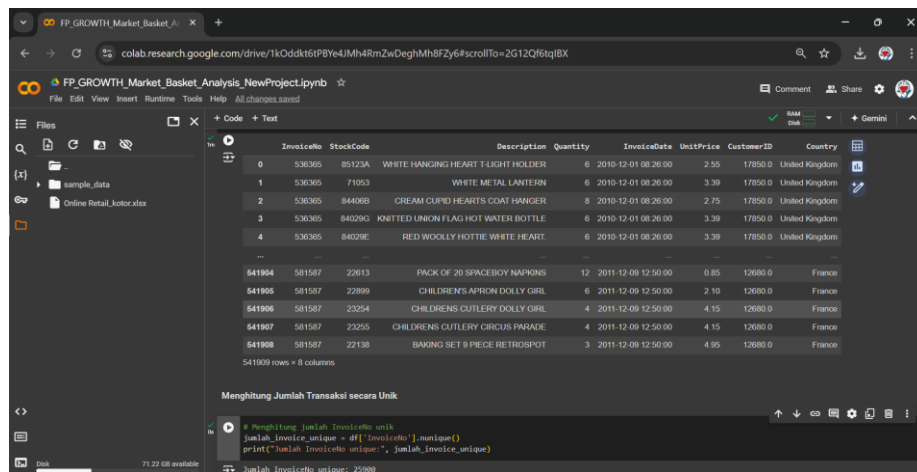
Tujuan bisnis adalah melakukan analisis keranjang belanja pada *dataset* transaksi penjualan ritel *non-toko* untuk menemukan aturan asosiasi melalui pemodelan *FP-Growth*, serta mengevaluasi aturan yang dihasilkan dari analisis tersebut.

4.2. Pemahaman Data (*Data Understanding*)

Memahami data dimulai dari mengimpor *dataset* transaksi penjualan ritel *non-toko* ke dalam *tools Google Colab* menggunakan bahasa pemrograman *Python* memungkinkan pemrosesan *dataset* secara menyeluruh, seperti yang terlihat pada Gambar 2. Selanjutnya, data transaksi dihitung secara unik berdasarkan satu kemunculan nomor transaksi (*InvoiceNo*), dimana terdiri dari 25.900 jumlah transaksi yang belum melalui proses *pre-processing*, seperti yang terlihat pada Gambar 3. Kemudian, dilihat dari 8 kolom variabel *dataset* tidak memiliki hubungan sebab-akibat untuk diklasifikasikan sebagai variabel terikat dan bebas, melainkan sebagai variabel yang mendeskripsikan fakta terkait faktur, seperti yang terlihat pada Tabel 1.



Gambar 2. Memuat *Dataset* ke *Google Colab*



Gambar 3. Menghitung Jumlah Transaksi secara Unik sebelum *Pre-processing*

Tabel 1. Variabel-variabel yang Termuat Dalam *Dataset*

No	Variabel (Dalam bahasa Inggris)	Variabel (Dalam bahasa Indonesia)	Deskripsi variabel	Tipe data
1	<i>InvoiceNo</i>	Nomor Faktur	Merupakan pengenal unik untuk setiap penjualan	<i>Object</i>
2	<i>StockCode</i>	Kode Stok	Merupakan kode untuk menemukan barang yang dijual	<i>Object</i>
3	<i>Description</i>	Deskripsi	Memberikan informasi tentang nama barang yang dijual	<i>Object</i>
4	<i>Quantity</i>	Kuantitas	Memberikan informasi jumlah barang yang dijual dalam satu faktur	<i>Int64</i>
5	<i>InvoiceDate</i>	Tanggal Faktur	Memberikan informasi kapan transaksi penjualan terjadi, memuat jam, tanggal, bulan, dan tahun	<i>Datetime</i>
6	<i>UnitPrice</i>	Harga Satuan	Merupakan harga per unit barang yang dijual	<i>Float64</i>
7	<i>CustomerID</i>	ID Pelanggan	Merupakan pengenal unik untuk pelanggan yang membeli barang	<i>Float64</i>
8	<i>Country</i>	Negara	Menunjukkan negara tujuan pengiriman barang kepada pelanggan	<i>Object</i>

4.3. Persiapan Data (Data Preparation)

Dalam tahap persiapan data, proses dimulai dengan pembersihan setiap variabel untuk menghapus *record* yang tidak konsisten, seperti yang terlihat pada Gambar 4, serta *record* yang hilang, seperti yang terlihat pada Gambar 5. Langkah berikutnya, setelah pembersihan, adalah menghitung kembali jumlah transaksi unik berdasarkan variabel *InvoiceNo*, di mana terdapat 18.405 transaksi. Jumlah ini akan digunakan untuk analisis lebih lanjut, seperti yang terlihat pada Gambar 6. Lalu, pada Gambar 7 menunjukkan pemilihan variabel nomor faktur (*invoice no*), nama barang (*description*), dan kuantitas (*quantity*) yang kemudian dikelompokkan dan diubah menjadi tabel pivot. Setiap baris dalam tabel pivot mewakili nomor faktur, sementara setiap kolom mewakili nama barang. Jumlah kemunculan barang dalam setiap transaksi dihitung menggunakan kode *.count()* berdasarkan *quantity*. Artinya, terlepas dari jumlah barang (*item*) yang dibeli, baik 1 atau 100, setiap kemunculan barang tetap dihitung sebagai satu kemunculan dengan nilai 1.0 per transaksi. Nilai 1.0 ini muncul, karena tipe data kolom adalah *float* dan dipengaruhi oleh penggunaan kode *.fillna(0)*, yang secara otomatis menggantikan nilai kosong (NaN) dengan 0.0, jika barang tidak ada dalam transaksi. Sedangkan, Gambar 8 menunjukkan proses *one-hot encoding* menggunakan kode *def encode_units(x): if x <= 0: return 0 if x >= 1: return 1*. Kode ini merepresentasikan setiap nilai barang sebagai vektor biner, di mana nilai 1.0 diganti menjadi 1 (barang dibeli) dan nilai 0.0 diganti menjadi 0 (barang tidak dibeli). Melalui proses ini, data kategorikal diubah menjadi format numerik yang lebih sederhana dan dapat dipahami oleh algoritma pembelajaran mesin, sehingga analisis dapat dilakukan dengan lebih baik pada tahap pemodelan.

```

# Menghilangkan spasi di awal dan akhir dari kolom 'Description'
data_select[Description] = data_select[Description].str.strip()
# Menghapus baris yang memiliki nilai NaN pada kolom 'InvoiceNo'
data_select.dropna(subset=['InvoiceNo'], inplace=True)
# Menghapus nilai 'InvoiceNo' menjadi tipe string
data_select['InvoiceNo'] = data_select['InvoiceNo'].astype(str)
# Memeriksa baris di mana 'InvoiceNo' tidak mengandung huruf 'I' (dengan asumsi 'InvoiceNo' adalah kolom string)
data_select = data_select[data_select['InvoiceNo'].astype(str).str.contains('I')]
# Memeriksa baris di mana 'InvoiceNo' tidak mengandung huruf '4' (dengan asumsi 'InvoiceNo' adalah kolom string)
data_select = data_select[data_select['InvoiceNo'].astype(str).str.contains('4')]
# Memeriksa baris di mana 'StockCode' tidak mengandung data 'gift' (dengan asumsi 'StockCode' adalah kolom string)
data_select = data_select[data_select['StockCode'].astype(str).str.contains('gift')]
# Mengubah semua teks di kolom 'Description' menjadi huruf kecil
data_select[Description] = data_select[Description].str.lower()
# Memeriksa baris di mana 'Description' tidak mengandung huruf 'w' (dengan asumsi 'Description' adalah kolom string)
data_select = data_select[data_select['Description'].str.contains('w')]
# Memeriksa baris di mana 'StockCode' tidak sama dengan '9999'
data_select = data_select[data_select['StockCode'] != '9999']
# Memeriksa baris di mana 'StockCode' tidak sama dengan '000'
data_select = data_select[data_select['StockCode'] != '000']
# Memeriksa baris di mana 'StockCode' tidak sama dengan '2000'
data_select = data_select[data_select['StockCode'] != '2000']
# Memeriksa baris di mana 'StockCode' tidak sama dengan '0'
data_select = data_select[data_select['StockCode'] != '0']
# Memeriksa baris di mana 'Description' tidak sama dengan 'check'
data_select = data_select[data_select['Description'] != 'check']
# Memeriksa baris di mana 'Description' tidak sama dengan 'return adjust'
data_select = data_select[data_select['Description'] != 'return adjust']
# Memeriksa baris di mana 'Description' tidak sama dengan 'return stock'
data_select = data_select[data_select['Description'] != 'return stock']
# Memeriksa baris di mana 'Description' tidak sama dengan 'check'
data_select = data_select[data_select['Description'] != 'check']
# Memeriksa baris di mana 'Description' tidak sama dengan 'return adjust'
data_select = data_select[data_select['Description'] != 'return adjust']
# Memeriksa baris di mana 'Description' tidak sama dengan 'return stock'
data_select = data_select[data_select['Description'] != 'return stock']
# Memeriksa baris di mana 'Description' tidak sama dengan 'check'
data_select = data_select[data_select['Description'] != 'check']

```

Gambar 4. Penghapusan Nilai Record yang Tidak Konsisten

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	CustomerID	Country	UnitPrice
0	0	584	0	0	132413	0	0

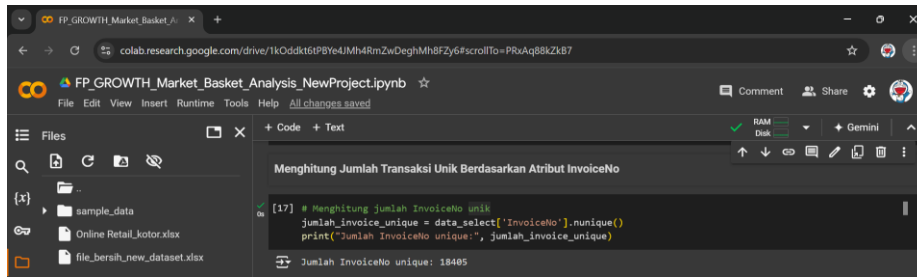
```

# Memeriksa jumlah nilai yang hilang di setiap kolom tertentu
data_select.isnull().sum()

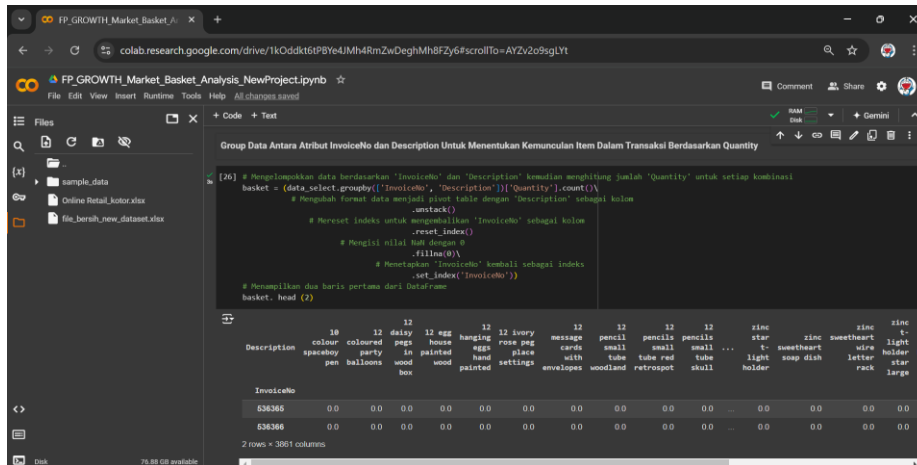
# Menghapus baris dengan nilai yang hilang di beberapa kolom tertentu
cols_to_check = ['Description', 'CustomerID']
data_select.dropna(subset=cols_to_check, inplace=True)

```

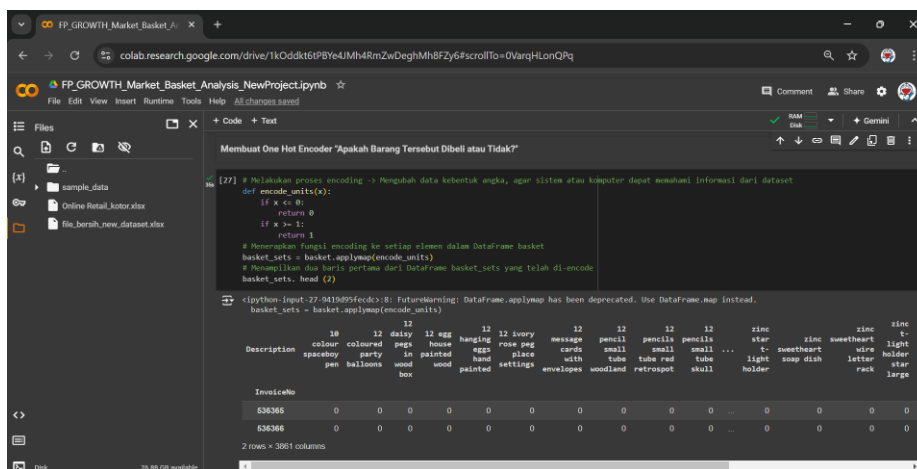
Gambar 5. Penghapusan Nilai Record yang Hilang



Gambar 6. Menghitung Jumlah Transaksi secara Unik setelah *Pre-processing*



Gambar 7. Pengelompokkan Variabel untuk Analisis Keranjang Belanja



Gambar 8. Proses *One-Hot Encoding*

4.4. Pemodelan (*Modeling*)

Pada tahap ini, model dibangun menggunakan algoritma *FP-Growth* dengan membandingkan berbagai nilai *minimum support*, yaitu 0,09% (0,0009), 0,5% (0,005), 1% (0,01), 1,5% (0,015), 2% (0,02), dan 2,5% (0,025). Interval nilai yang dipilih tidak terlalu jauh maupun terlalu dekat, sehingga analisis dapat dilakukan dengan lebih detail, seperti yang terlihat pada Tabel 2. Tahapan ini memiliki kode tertentu untuk penyelesaiannya, antara lain:

- 1) Kode untuk membangun model dengan berbagai nilai *minimum support*:

```

from mlxtend.frequent_patterns.fpgrowth import fpgrowth frequent_itemsets =
fpgrowth(basket_sets, min_support=#sesuaikan nilai minimum support, use_colnames = True)
frequent_itemsets
    
```

- 2) Kode untuk menghitung jumlah *frequent itemset* yang terbentuk berdasarkan panjang kombinasi:

```
combination_counts = frequent_itemsets['itemsets'].apply(lambda x:
len(x).value_counts().sort_index() print("Jumlah kombinasi itemset yang terbentuk:") for length,
count in combination_counts.items(): print(f"{length}-itemset: {count}")
```

Tabel 2. Hasil Perbandingan Model FP-Growth berdasarkan Minimum Support

No	Nilai minimum support	Jumlah frequent itemsets yang terbentuk	Jumlah frequent itemsets berdasarkan panjang kombinasi
1	2,5% (0,025)	149 itemset	1-itemset: 140 2-itemset: 9
2	2% (0,02)	242 itemset	1-itemset: 206 2-itemset: 35 3-itemset: 1
3	1,5% (0,015)	448 itemset	1-itemset: 359 2-itemset: 85 3-itemset: 4
4	1% (0,01)	980 itemset	1-itemset: 621 2-itemset: 304 3-itemset: 53 4-itemset: 2
5	0,5% (0,005)	4039 itemset	1-itemset: 1248 2-itemset: 1733 3-itemset: 757 4-itemset: 255 5-itemset: 45 6-itemset: 1
6	0,09% (0,0009)	397027 itemset	1-itemset: 2641 2-itemset: 77613 3-itemset: 88380 4-itemset: 72048 5-itemset: 66609 6-itemset: 49109 7-itemset: 27112 8-itemset: 10712 9-itemset: 2533 10-itemset: 261 11-itemset: 9

Pembentukan model FP-Growth ini menggunakan fungsi *fpgrowth* dari pustaka *mlxtend* dengan berbagai parameter *minimum support* yang diterapkan. Dalam fungsi *fpgrowth* ini, kerja dari algoritma FP-Growth dilakukan di belakang kode, di mana mencakup semua perhitungan dari awal pembentukan kombinasi sederhana (1-itemset) hingga lebih kompleks (2-itemset atau lebih).

4.5. Evaluasi (Evaluation)

Tahap evaluasi dilakukan setelah fase pemodelan untuk menilai kualitas aturan asosiasi yang dihasilkan menggunakan parameter *confidence* dan *lift*, seperti yang terlihat pada Tabel 3. Tahapan ini memiliki kode tertentu untuk penyelesaiannya, antara lain:

- 1) Kode untuk menghasilkan aturan asosiasi dari frequent itemset:

```
from mlxtend.frequent_patterns import association_rules data_asosiasi =
association_rules(frequent_itemsets, metric="lift") data_asosiasi=data_asosiasi[['antecedents',
'consequents', 'support', 'confidence', 'lift']]
```
- 2) Kode untuk menghitung jumlah aturan asosiasi yang dihasilkan:

```
jumlah_aturan_asosiasi = len(data_asosiasi) print(f"Jumlah aturan asosiasi yang
terbentuk: {jumlah_aturan_asosiasi}")
```
- 3) Kode untuk menampilkan maksimum dan minimum nilai confidence:

```
min_confidence = data_asosiasi['confidence'].min() * 100 max_confidence =
data_asosiasi['confidence'].max() * 100 print(f"Confidence minimum: {min_confidence}%")
print(f"Confidence maksimum: {max_confidence}%")
```
- 4) Kode untuk menampilkan nilai lift:

```
data = data_asosiasi['lift'] df = pd.DataFrame(data) def count_lift(df, criteria): return
df.query(criteria).shape[0] lift_1_count = count_lift(df, "lift == 1") lift_gte_1_count = count_lift(df,
"lift > 1") lift_lt_1_count = count_lift(df, "lift < 1") lift_min = df['lift'].min() lift_max = df['lift'].max()
```



```
print("Jumlah data dengan lift == 1:", lift_1_count) print("Jumlah data dengan lift > 1:",
lift_gte_1_count) print("Jumlah data dengan lift < 1:", lift_lt_1_count) print("Nilai lift minimum:",
lift_min) print("Nilai lift maksimum:", lift_max)
```

Tabel 3. Evaluasi Aturan Asosiasi

No	Nilai <i>minimum support</i>	Aturan asosiasi yang dihasilkan	Nilai <i>confidence</i>	Nilai <i>lift</i>
1	2,5% (0,025)	18 aturan	Minimum 34.13% Maksimum 78.29%	Minimum 6.75 Maksimum 18.40 Semua aturan asosiasi >1
2	2% (0,02)	76 aturan	Minimum 21.95% Maksimum 89.45%	Minimum 3.80 Maksimum 23.86 Semua aturan asosiasi >1
3	1,5% (0,015)	194 aturan	Minimum 14.16% Maksimum 89.45%	Minimum 1.89 Maksimum 29.29 Semua aturan asosiasi >1
4	1% (0,01)	954 aturan	Minimum 9.39% Maksimum 93.91%	Minimum 1.29 Maksimum 66.38 Semua aturan asosiasi >1
5	0,5% (0,005)	12990 aturan	Minimum 4.72% Maksimum 99.25%	Minimum 0.99 Maksimum 134.43 12986 aturan asosiasi >1 dan 4 aturan asosiasi <1
6	0,09% (0,0009)	-	-	-

4.6. Penyebaran (*Deployment*)

4.6.1. Pembahasan

Berdasarkan hasil penelitian yang telah dilakukan, maka untuk memilih salah satu model *FP-Growth* yang digunakan sebagai acuan untuk menampilkan hasil model aturan asosiasi yang ditemukan dari *dataset* penjualan ritel *non-toko* berdasarkan pertimbangan beberapa kriteria, sebagai berikut:

1) Jumlah *frequent itemset*:

Semakin banyak *frequent itemsets* yang dihasilkan semakin baik, karena banyaknya aturan asosiasi yang dihasilkan dapat memberikan wawasan yang lebih baik tentang perilaku konsumen [30]. Namun, perlu diperhatikan agar tidak terjadi ledakan kombinatorial yang dapat membuat analisis menjadi kurang bermakna [31].

2) Parameter *confidence*:

Nilai *confidence* yang lebih tinggi menunjukkan aturan asosiasi yang lebih kuat. Namun, nilai di bawah 100%, masih dianggap cukup penting, karena evaluasi aturan asosiasi juga memperhitungkan nilai *lift* [32].

3) Parameter *lift*:

Jika nilai *lift* lebih dari 1, aturan dianggap positif, karena *item-item* tersebut sering dibeli bersamaan dalam transaksi. Sebaliknya, jika nilai *lift* kurang dari 1, aturan dianggap negatif, karena *item-item* tersebut jarang dibeli bersamaan dalam transaksi. Dan nilai *lift* sama dengan 1 menunjukkan bahwa *item-item* tersebut hanya kebetulan dibeli bersamaan dalam transaksi dan tidak memiliki pengaruh khusus [32].

Dari berbagai pertimbangan, model *FP-Growth* ke-4 dengan nilai minimum support 1% (0,01) memberikan keseimbangan terbaik, karena menghasilkan lebih banyak *frequent itemsets* tanpa terjadi ledakan kombinatorial. Model ke-1 dan ke-2 dieliminasi karena menghasilkan terlalu sedikit *frequent itemsets*, sementara model ke-6 dieliminasi karena ledakan kombinatorial yang menyebabkan ketidakmampuan untuk menghasilkan aturan asosiasi. Model ke-5 juga dieliminasi, karena beberapa aturan yang dihasilkan memiliki nilai *lift* di bawah satu. Meskipun model ke-3 dan ke-4 sama-sama menghasilkan aturan dengan nilai *confidence* dan *lift* yang tinggi, model ke-3 dieliminasi, karena menghasilkan lebih sedikit *frequent itemsets* dibandingkan model ke-4.

Oleh karena itu, model ke-4, seperti yang terlihat pada Gambar 9, digunakan sebagai acuan dalam menganalisis asosiasi yang ditemukan dari *dataset* transaksi penjualan ritel *non-toko*. Setelah itu, dilakukan proses *pickling* pada model, di mana objek *Python* diubah menjadi *format byte stream* dalam sebuah file melalui fungsi `pickle.dump(frequent_itemsets, f)`.

No	antecedents	consequents	support	confidence	lift
1	frozenset({'hand warmer owl design'})	frozenset({'hand warmer union jack'})	0.01135561	0.365384615	15.24921507
2	frozenset({'hand warmer union jack'})	frozenset({'hand warmer owl design'})	0.01135561	0.473922902	15.24921507
3	frozenset({'hand warmer scotty dog design'})	frozenset({'hand warmer union jack'})	0.010323282	0.403397028	16.83565146
4	frozenset({'hand warmer union jack'})	frozenset({'hand warmer scotty dog design'})	0.010323282	0.430839002	16.83565146
5	frozenset({'assorted colour bird ornament'})	frozenset({'white hanging heart t-light holder'})	0.01472426	0.197090909	1.84041511
.....
950	frozenset({'lunch bag vintage doily'})	frozenset({'lunch bag apple design'})	0.010051616	0.343866171	7.308148819
951	frozenset({'hot water bottle keep calm'})	frozenset({'love hot water bottle'})	0.011844607	0.303198887	12.42845328
952	frozenset({'love hot water bottle'})	frozenset({'hot water bottle keep calm'})	0.011844607	0.485523385	12.42845328
953	frozenset({'hand warmer owl design'})	frozenset({'hand warmer red love heart'})	0.011029612	0.354895105	17.60604961
954	frozenset({'hand warmer red love heart'})	frozenset({'hand warmer owl design'})	0.011029612	0.547169811	17.60604961

Gambar 9. Tampilan Model Aturan dengan *Minimum Support* 1% (0,01)

4.6.2. Penyebaran

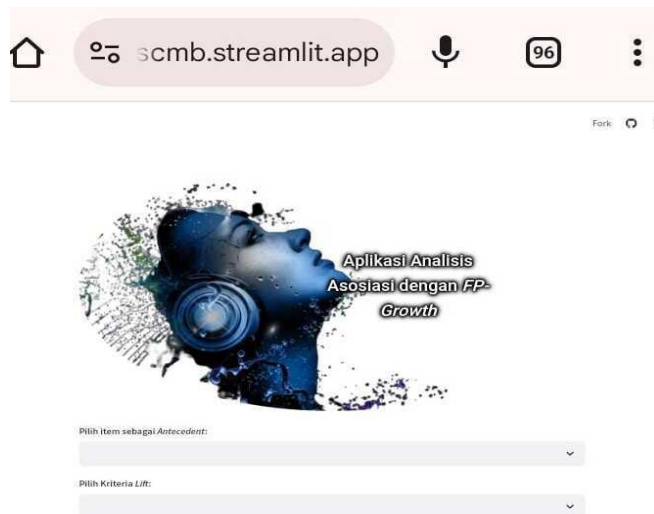
Pada tahap penyebaran, hasil analisis disampaikan melalui aplikasi berbasis web. Model aturan asosiasi ini akan diimplementasikan melalui aplikasi berbasis web dengan antarmuka yang sederhana, intuitif, dan interaktif, dibangun menggunakan *library Python Streamlit* versi 1.36.0 di *Visual Studio Code*. Seperti terlihat pada Gambar 10, kode ini memuat model ke-4 dan *dataset* penjualan ritel *non-toko*. Saat pengguna memilih barang (*item*) dari *dataset*, fungsi *def analyze_association*, yang terintegrasi langsung dengan model, akan dipanggil untuk melakukan analisis dan menampilkan aturan asosiasi.

```

54 try:
55     # Memuat model pickle
56     with open('frequent_itemsets_new_model_0.01.pkl', 'rb') as file:
57         frequent_itemsets = pickle.load(file)
58
59     # Memuat dataset atau objek data yang berisi nama produk
60     data = pd.read_excel('C:\\Users\\ASUS\\Desktop\\SKRIPSI\\asosiasi\\file_bersih_new_dataset.xlsx')
61
62     # Mengambil semua nama produk dari kolom 'Description'
63     products = data['Description'].unique().tolist()
64     # Menambahkan opsi kosong di awal daftar produk
65     products.insert(0, "")
66     # Membuat dropdown menu untuk memilih produk sebagai antecedent
67     selected_product = st.selectbox(
68         '**Pilih Produk sebagai *Antecedent**', products)
69
70     # Menambahkan dropdown menu untuk memilih kriteria lift
71     selected_lift = st.selectbox(

```

Gambar 10. Membangun Aplikasi Analisis Asosiasi Menggunakan *Streamlit* di *Visual Studio Code*



Gambar 11. Tampilan Aplikasi Analisis Asosiasi berbasis Web

Aplikasi ini berhasil di-hosting melalui *tools Streamlit Share* yang terintegrasi dengan *repository GitHub* dan dapat diakses melalui tautan berikut <https://s.id/AplikasiAnalisisAsosiasiFP-Growth>, ketika aplikasi digunakan, jika *item A* sebagai *antecedent*, yaitu penyebab yang mengarahkan pada pembelian *item consequent* memiliki aturan asosiasi, maka *item B* akan muncul sebagai *consequent*, yaitu *item* yang dibeli sebagai hasil dari pembelian *antecedent*. Akan tetapi, jika *item A* sebagai *antecedent* yang dipilih tidak memiliki aturan asosiasi, maka muncul pesan bahwa tidak ada aturan asosiasi yang ditemukan, seperti yang terlihat pada Gambar 11.

5. Simpulan

Analisis keranjang belanja pada *dataset* transaksi penjualan ritel *non-toko* dilakukan melalui tahapan CRISP-DM dan diproses secara menyeluruh dalam *tools google colab*. Dengan membandingkan berbagai nilai *minimum support*, ditemukan bahwa model dengan *minimum support* 1% (0,01) memberikan keseimbangan terbaik. Model ini membentuk 980 jumlah *frequent itemsets* dengan panjang kombinasi 1-*itemset* sebanyak 621, 2-*itemset* sebanyak 304, 3-*itemset* sebanyak 53, dan 4-*itemset* sebanyak 2, yang menghasilkan 954 aturan asosiasi. Evaluasi aturan asosiasi berdasarkan parameter nilai *confidence* menunjukkan nilai maksimum sebesar 93,91% dan minimum sebesar 9,39%. Lalu berdasarkan evaluasi parameter nilai *lift*, semua aturan asosiasi memiliki nilai > 1 , dengan maksimum sebesar 66,38 dan minimum sebesar 1,29. Hal ini menunjukkan bahwa semua aturan asosiasi kuat dan sering terjadi, di mana *item B* (*consequent*) sering dibeli bersamaan dengan *item A* (*antecedent*) dalam satu keranjang belanja oleh pelanggan. Pengetahuan yang diperoleh dari model aturan asosiasi ini memungkinkan untuk pengaturan tata letak *item* secara berdekatan dan rekomendasi *item* dalam konteks penjualan ritel *non-toko*. Dengan harapan mampu meningkatkan pengalaman belanja pelanggan secara keseluruhan sebagai strategi penjualan. Untuk memudahkan pemahaman dalam pengambilan keputusan oleh pemilik bisnis, hasil analisis asosiasi ini disajikan melalui aplikasi berbasis web yang dibangun menggunakan *library Python Streamlit* versi 1.36.0 dan telah berhasil di-hosting melalui *tools Streamlit Share* yang terintegrasi dengan *repository GitHub*, sehingga dapat diakses melalui tautan web. Pengembangan selanjutnya dapat menambahkan parameter nilai *leverage* dan *conviction* untuk mengevaluasi lebih lanjut kualitas aturan asosiasi yang dihasilkan. Selain itu, uji coba lapangan perlu dilakukan untuk melihat bagaimana analisis berfungsi dalam praktik.

Daftar Referensi

- [1] F. Provost and T. Fawcett, "Data Science and Its Relationship to Big Data and Data-Driven Decision Making," *Big Data*, vol. 1, no. 1, pp. 51–59, 2013, doi: 10.1089/big.2013.1508.
- [2] A. Baijuri et al., *Analisis Sistem Informasi*, I. Padang: CV. Gita Lentera, 2023.
- [3] S. Chevalier, "Global retail e-commerce sales 2014-2027", Statista, May 22, 2024. [Online]. Tersedia: <https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/> [Diakses: 5 Mei 2024].
- [4] D. Supriyadi, "Penerapan Association Rule Mining Berbasis Algoritma Frequent Pattern Growth untuk Rekomendasi Penjualan," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 7, no. 2, pp. 135–148, 2020, doi: 10.35957/jatisi.v7i2.339.
- [5] U. Sa'adah, M. Yessi Rochayani, D. Wahyu Lestari, and D. Ayu Lusua, *Kupas Tuntas Algoritma Data Mining dan Implementasinya menggunakan R*, I. Malang: UB Press, 2021.
- [6] J. Xu, H. Naseer, S. Maynard, and J. Filippou, "Leveraging Data and Analytics for Digital Business Transformation through DataOps: An Information Processing Perspective," in *Seminar Digital Business Transformation through DataOps, ACIS - Australasian Conference on Information Systems, Proceedings*, pp. 1–11, 2021.
- [7] I. Musdalifah and A. Jananto, "Analisis Perbandingan Algoritma Apriori Dan FP-Growth Dalam Pembentukan Pola Asosiasi Keranjang," *J. Ilm. Komput.*, vol. 18, no. 2, p. 10-21, 2022.
- [8] A. F. Ramadhan, N. I. Jaafar, and F. P. Tajudeen, "Data Governance and its Scientific Outlook in Indonesia: A Literature Review," *J. Manag. Inf. Decis. Sci.*, vol. 24, no. 3, pp. 1–10, 2021.
- [9] E. D. Brown, "Drowning in Data, Starved for Information, Big Data Featured," *October*

- 22, 2014. [Online]. Tersedia: <https://ericbrown.com/drowning-in-data-starved-for-information.htm> [Diakses: 12 Maret 2024]
- [10] E. S. A. Sahputra, "Application of Big Data and Analytics to Increase Competitive Advantage," *J. Indones. Sos. Teknol.*, vol. 5, no. 5, pp. 2383–2390, 2024, doi: 10.59141/jjst.v5i5.1084.
- [11] A. Kaur and S. Lodhia, "Stakeholder Engagement in Sustainability Accounting and Reporting," *Account. Audit. Account. J.*, vol. 31, no. 1, pp. 338–368, 2018, doi: 10.1108/aaaj-12-2014-1901.
- [12] M. Fitri, L. M. Anastasia, and M. A. Tubagus, *Data Mining Konsep Dan Penerapannya*, I. Yogyakarta: Deepublish, 2021.
- [13] A. T. Y. Al Abd Alazeez, "Data Stream Mining Between Classical and Modern Applications: A Review," *J. Educ. Sci.*, vol. 30, no. 5, pp. 30–43, 2021, doi: 10.33899/edusj.2021.130093.1158.
- [14] P. N. Harahap and S. Sulindawaty, "Implementasi Data Mining Dalam Memprediksi Transaksi Penjualan Menggunakan Algoritma Apriori (Studi Kasus PT.Arma Anugerah Abadi Cabang Sei Rampah)," *J. Sist. Inf. Kaputama*, vol. 4, no. 1, pp. 1-12, 2020.
- [15] L. Indah Prahartiwi, S. Informasi, S. Nusa Mandiri, J. Damai No, and W. Jati Barat Jakarta Selatan DKI Jakarta, "Pencarian Frequent Itemset pada Analisis Keranjang Belanja Menggunakan Algoritma FP-Growth," *Inf. Syst. Educ. Prof.*, vol. 2, no. 1, pp. 1–10, 2017.
- [16] S. W. Romi, "5 Peran Data Mining: Kuasai Konsep dengan Mudah dan Praktek Cepat untuk Data Scientist Galau - YouTube," 2022. [Online]. Tersedia: <https://www.youtube.com/watch?v=VstEO9nkVoM> [Diakses: 28 September 2024].
- [17] BAKTI, "3 Komponen Sistem Komputer : Hardware, Software dan Brainware", BAKTI KOMINFO, 2024. [Online]. Tersedia: <https://www.baktikominfo.id/id/detail-berita/3-komponen-sistem-komputer-hardware-software-dan-brainware> [Diakses: 25 Maret 2024].
- [18] I. G. A. Indrawan, M. Sudarma, and L. Jasa, "Implementasi Algoritma FP-Growth Dengan Closure Table Untuk Penemuan Frequent Itemset Pada Keranjang Belanja," *Maj. Ilm. Teknol. Elektro*, vol. 17, no. 2, pp. 199-208, 2018, doi: 10.24843/mite.2018.v17i02.p06.
- [19] Y. Husain, E.D. Oktaviyani, and S. Christina, "Analisis Perbandingan Algoritma Apriori, FP-Growth, Dan Eclat dalam Menemukan Pola Pembelian Konsumen," *KONSTELASI/ Konvergensi Teknol. dan Sist. Inf.*, vol. 3, no. 2, pp. 231–243, 2023, doi: 10.24002/konstelasi.v3i2.7007.
- [20] D. Rachmawati, Y. Cahyana, E. E. Awal, and S. Faisal, "Perbandingan Algoritma Apriori dan Algoritma FP-Growth dalam Menentukan Pola Penjualan Pupuk," vol. 3, no. 1, pp. 21–31, 2024.
- [21] S. Yakub, A. Fitri Boy, I. Mariami, W. Stmik, and T. Dharma, "J-SISKO TECH Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD Penerapan Data Mining Pengaturan Pola Tata Letak Barang Pada Berkah Swalayan Untuk Strategi Penjualan Menggunakan Algoritma Apriori," *J-SISKO TECH J. Teknol. Sist. Inf. dan Sist. Komput. TGD*, vol. 69, no. 1, pp. 69–75, 2019.
- [22] A. A. N. Risal, F. Adiba, and A. A. Nurfitri, "Aturan Asosiasi Berbasis Algoritma Apriori Pada Penjualan Retail Online," *J. Mediat.*, vol. 6, no. 2, pp. 1–4, 2024, doi: 10.59562/mediatik.v6i2.1394.
- [23] F. Firmansyah and A. Yulianto, "Market Basket Analysis for Books Sales Promotion using FP Growth Algorithm, Case Study : Gramedia Matraman Jakarta," *J. Informatics Telecommun. Eng.*, vol. 4, no. 2, pp. 383–392, 2021, doi: 10.31289/jite.v4i2.4539.
- [24] D. H. Suryawan, "Dializer: Aplikasi Simulasi Diagram Alir Berbasis Web," *Jati (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 5, pp. 3284–3490, 2024, doi: 10.36040/jati.v7i5.7685.
- [25] A. I. Idris *et al.*, "Comparison of Apriori, Apriori-Tid and FP-Growth Algorithms in Market Basket Analysis at Grocery Stores," *Ijics (International J. Informatics Comput. Sci.)*, vol. 6, no. 2, pp. 107-116, 2022, doi: 10.30865/ijics.v6i2.4535.
- [26] A. H. Nasyuha *et al.*, "Frequent Pattern Growth Algorithm for Maximizing Display Items," *Telkomnika (Telecommunication Comput. Electron. Control.)*, vol. 19, no. 2, pp. 390-401, 2020, doi: 10.12928/telkomnika.v19i2.16192.
- [27] R. Yogasuwara and F. Ferdiansyah, "Implementasi Algoritma Frequent Growth (FP-

- Growth) Menentukan Asosiasi Antar Produk,” *J. Sist. Komput. dan Inform.*, vol. 4, no. 1, pp. 165-175, 2022, doi: 10.30865/json.v4i1.4894.
- [28] D. Aha, “Dataset”, Home - UCI Machine Learning Repository, 2023. [Online]. Tersedia: <https://archive.ics.uci.edu/dataset/352/online+retail> [Diakses: 25 Maret 2024].
- [29] J. Saltz, I. Shamshurin, and C. Connors, “Predicting data science sociotechnical execution challenges by categorizing data science projects,” *J. Assoc. Inf. Sci. Technol.*, vol. 68, no. 12, pp. 2720–2728, Dec. 2017, doi: 10.1002/ASI.23873.
- [30] F. E. Gulo, “Penerapan Aturan Asosiasi Di Darvina Mart Untuk Menentukan Pola Pembelian Pelanggan,” *J. Ind. Innov. Saf. Eng.*, vol. 1, no. 2, pp. 92–100, 2023, doi: 10.35718/jinseng.v1i2.897.
- [31] J. Ha, M. Kambe, and J. Pe, *Data Mining: Concepts and Techniques*, III. Amerika: Morgan Kaufmann, 2011.
- [32] Lia Farokhah, “Evaluasi Algoritma Asosiasi Menggunakan Lift Ratio | DATA MINING - Youtube,” 2021. [Online]. Tersedia: <https://youtu.be/A2jKP2vke2o?si=X3AkL3nv2qLcnQbq> [Diakses: 25 Maret 2024].