

LAPORAN TESIS

**PENERAPAN *ASSOCIATION RULE* DENGAN
ALGORITMA APRIORI PADA TRANSAKSI
PENJUALAN MAKANAN *ONLINE*
(STUDI KASI: DAPUR TIARA)**

Jihan Tiara Amanda

NIM. 2156102011

DOSEN PEMBIMBING

Dr. Yohana Dewi Lulu Widyasari, S.Si., M.T.

Dini Nurmalasari, S.T., M.T.

PROGRAM STUDI

MAGISTER TERAPAN TEKNIK KOMPUTER

POLITEKNIK CALTEX RIAU

2025

LAPORAN TESIS

**PENERAPAN *ASSOCIATION RULE* DENGAN
ALGORITMA APRIORI PADA TRANSAKSI
PENJUALAN MAKANAN *ONLINE*
(STUDI KASI: DAPUR TIARA)**

Jihan Tiara Amanda

NIM. 2156102011

DOSEN PEMBIMBING

Dr. Yohana Dewi Lulu Widyasari, S.Si., M.T.

Dini Nurmalasari, S.T., M.T.

PROGRAM STUDI

MAGISTER TERAPAN TEKNIK KOMPUTER

POLITEKNIK CALTEX RIAU

2025

HALAMAN PENGESAHAN

PENERAPAN ASSOCIATION RULE DENGAN ALGORITMA APRIORI PADA TRANSAKSI PENJUALAN MAKANAN ONLINE (STUDI KASUS: DAPUR TIARA)

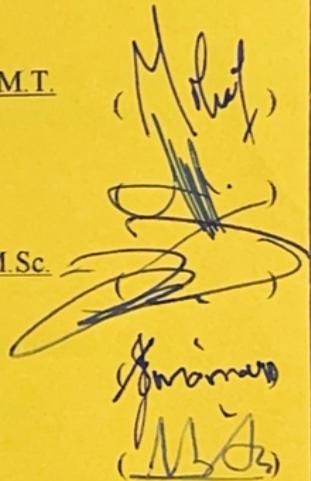
Oleh:

Jihan Tiara Amanda
NIM. 2156102011

Tesis ini disusun untuk memenuhi salah satu persyaratan memperoleh
Gelara Magister Terapan Komputer (M.Tr.Kom.)
di
Politeknik Caltex Riau
2025

Disetujui oleh:

- Pembimbing : Dr. Yohana Dewi Lulu Widyasari, S.Si., M.T.
Utama NIP. 007717
- Pembimbing : Dini Nurmalasari, S.T., M.T.
Pendamping NIP. 048108
- Penguji : Dr. Eng. Yoanda Alim Syahbana, S.T., M.Sc.
NIP. 148809
- Penguji : Dr. Ir. Emansa Hasri Putra, S.T., M.Eng.
NIP. 017501
- Penguji : Memem Akbar, S.Si, M.T.
NIP. 078313



Mengetahui,

Ketua Program Studi Magister Terapan Teknik Komputer
Politeknik Caltex Riau


Dr. Eng. Yoanda Alim Syahbana, S.T., M.Sc.
NIP. 148809

HALAMAN PERNYATAAN ORISINALITAS

Dengan ini saya menyatakan bahwa bagian atau keseluruhan tesis ini:

1. Adalah hasil karya sendiri dan tidak mengandung unsur plagiat dari pihak lain
2. Tidak pernah diajukan untuk mendapatkan gelar akademis pada suatu Perguruan Tinggi
3. Tidak pernah dipublikasikan atau ditulis oleh pihak lain
4. Mencantumkan rujukan dan kutipan dengan jujur dan benar terhadap sumber referensi lain yang menunjang pembahasan pada tesis.

Apabila ditemukan bukti bahwa pernyataan saya diatas tidak benar, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku di Politeknik Caltex Riau.

Pekanbaru, 20 Agustus 2025

Yang menyatakan,

Jihan Tiara Amanda

HALAMAN KESEPAKATAN PUBLIKASI

Demi pengembangan ilmu pengetahuan, dengan ini saya menyatakan:

1. Memberikan persetujuan kepada Politeknik Caltex Riau untuk menyimpan, mengolah dalam bentuk pangkalan data, merawat, mengalih media/formatkan dan mempublikasikan tesis ini selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.
2. Tidak melakukan alih media/format dan publikasi dalam bentuk makalah ilmiah dari bagian atau keseluruhan tesis ini ke suatu publikasi ilmiah, pada seminar ataupun jurnal, skala nasional maupun internasional, kecuali ada persetujuan dari saya dan Dosen Pembimbing Utama, dan mencantumkan nama saya, Dosen Pembimbing Utama dan nama-nama lain (jika ada) yang berkontribusi pada makalah.

Pekanbaru, 20 Agustus 2025

Yang menyatakan,

Jihan Tiara Amanda

KATA PENGANTAR

Alhamdulillah, segala puji dan syukur penulis panjatkan kehadirat Allah Subhanahu wa Ta'ala, Tuhan semesta alam, atas limpahan rahmat, taufik, dan hidayah-Nya. Shalawat serta salam semoga selalu tercurah kepada Nabi Muhammad Shallallahu Alaihi Wasallam, keluarga, sahabat, dan seluruh umatnya. Amin. Sehingga penulis dapat menyelesaikan penulisan Tesis ini dengan judul “Penerapan *Association Rule* dengan Algoritma Apriori pada Transaksi Penjualan Makanan *Online* (Studi Kasus: DAPUR TIARA)”. Tesis ini disusun sebagai salah satu syarat untuk menyelesaikan jenjang Pendidikan Magister Terapan pada Program Studi Teknik Komputer Politeknik Caltex Riau. Pada kesempatan ini, penulis ingin mengucapkan terima kasih kepada pihak yang telah banyak memberikan bantuan dan dukungan yang tiada terhingga baik secara langsung maupun tidak langsung. Ucapan terima kasih tersebut penulis tujukan kepada:

1. Kedua orang tua, abang, kakak ipar dan adik penulis, atas dukungan dan kasih sayang tak terhingga, sehingga penulis mampu menyelesaikan tesis ini tepat waktu.
2. Bapak Dr. Dadang Syarif Sihabudin, S.Si,M.Sc. selaku Direktur Politeknik Caltex Riau.
3. Ibu Dr. Yohana Dewi Lulu Widyasari, S.Si.,M.T. selaku Dosen Pembimbing I yang telah memberikan masukan dan arahan dengan penuh kesabaran dalam menyelesaikan tesis ini.
4. Ibu Dini Nurmalasari, S.T., M.T. selaku Dosen Pembimbing II yang telah banyak memberikan bimbingan dan arahan serta bantuan untuk menyelesaikan tesis ini dengan penuh kesabaran.
5. Bapak Dr. Eng. Yoanda Alim Syahbana, S.T., M.Sc. selaku Ketua Program Studi Magister Terapan Teknik Komputer. Sekaligus Dosen Penguji I yang selalu mengingatkan, memberi saran dan memberikan arahan untuk penyelesaian tesis ini.

6. Bapak Ananda, S.Kom., M.T., Ph.D. selaku Koordinator Tesis yang selalu memberikan motivasi dan semangat untuk menyelesaikan tesis ini tepat waktu.
7. Bapak Dr. Ir. Emansa Hasri Putra, S.T.,M.Eng selaku Dosen Penguji II dan Bapak Memen Akbar, S.Si, M.T. selaku Dosen Penguji III penulis yang telah banyak memberikan saran dan arahan serta bantuan untuk menyelesaikan tesis ini.
8. Seluruh dosen khususnya Program Studi Magister Terapan Teknik Komputer dan seluruh dosen di Politeknik Caltex Riau pada umumnya yang telah memberikan bekal ilmu kepada penulis dalam menyelesaikan penulisan tesis ini.
9. Teman seperjuangan Angkatan 21MTTKB Program Studi Magister Terapan Teknik Komputer yang telah memberikan semangat dan masukkan dalam penulisan tesis ini.

Penulis sangat menyadari sepenuhnya bahwa laporan ini masih jauh dari sempurna, oleh karena itu segala jenis kritik, saran dan masukan yang membangun sangat penulis harapkan agar dapat memberikan wawasan bagi pembaca dan yang paling utama penulis sendiri.

Pekanbaru, 20 Agustus 2025

Jihan Tiara Amanda

ABSTRAK

Perkembangan teknologi digital telah membawa perubahan besar dalam gaya hidup masyarakat, termasuk dalam cara berbelanja yang kini banyak dilakukan secara daring. Salah satu sektor yang terdampak langsung adalah bisnis makanan online, di mana pelaku usaha perlu menerapkan strategi yang tepat untuk tetap bersaing dan meningkatkan penjualan. Penelitian ini dilakukan pada *Dapur Tiara*, sebuah usaha makanan yang memasarkan produknya melalui platform Instagram dan WhatsApp. Selama ini, data transaksi hanya disimpan sebagai arsip, padahal memiliki potensi besar untuk dianalisis guna menggali pola pembelian konsumen.

Penelitian ini menggunakan algoritma Apriori untuk menganalisis data transaksi penjualan dan mengidentifikasi produk-produk yang sering dibeli secara bersamaan. Tujuan utamanya adalah untuk membentuk aturan asosiasi yang dapat digunakan sebagai dasar strategi *cross-selling*. Proses yang dilakukan meliputi pembersihan data, transformasi ke format basket, pembentukan *frequent itemset*, evaluasi aturan asosiasi dengan menggunakan FP-Growth dan ECLAT sebagai pembanding, serta penyajian hasil melalui aplikasi web berbasis Flask. Eksperimen dilakukan pada nilai *minimum support* dari rentang 0 hingga 1 dengan kenaikan 0.1 dan *minimum confidence* dari rentang 0 hingga 1 juga.

Hasil penelitian menunjukkan bahwa terdapat beberapa kombinasi produk yang memiliki keterkaitan kuat, seperti Mini Pizza dan Pepperoni Pizza dengan confidence sebesar 0.48 dan lift 1.872. Aplikasi web yang dikembangkan memungkinkan pengguna mengunggah data transaksi dan secara otomatis menampilkan hasil analisis dalam bentuk tabel dan kalimat rekomendasi. Sistem ini dapat digunakan untuk menyusun strategi penjualan berbasis data, meningkatkan nilai transaksi per pelanggan, serta memberikan rekomendasi produk yang lebih personal dan relevan.

Kata Kunci : Algoritma Apriori, Transaksi Penjualan, Dapur Tiara, Rekomendasi Produk, Analisis Asosiasi, Visualisasi Web

ABSTRACT

The development of digital technology has brought significant changes to people's lifestyles, including in shopping habits, which are now largely conducted online. One of the sectors directly affected is the online food business, where entrepreneurs need to implement the right strategies to remain competitive and increase sales. This research was conducted on Dapur Tiara, a food business that markets its products through Instagram and WhatsApp. So far, transaction data has only been stored as an archive, even though it has great potential to be analyzed in order to uncover consumer purchasing patterns.

This study applies the Apriori algorithm to analyze sales transaction data and identify products that are frequently purchased together. The main objective is to generate association rules that can serve as the foundation for cross-selling strategies. The process includes data cleaning, transformation into basket format, generation of frequent itemsets, evaluation of association rules using FP-Growth and ECLAT as comparisons, and presenting the results through a web application built with Flask. Experiments were conducted with minimum support values ranging from 0 to 1 with an increment of 0.1, and minimum confidence values ranging from 0 to 1 as well.

The results show that several product combinations have strong relationships, such as Mini Pizza and Pepperoni Pizza with a confidence of 0.48 and a lift of 1.872. The developed web application allows users to upload transaction data and automatically display the analysis results in the form of tables and recommendation sentences. This system can be used to design data-driven sales strategies, increase the transaction value per customer, and provide more personalized and relevant product recommendations.

Keywords: *Apriori Algorithm, Sales Transactions, Dapur Tiara, Product Recommendation, Association Analysis, Web Visualization*

DAFTAR ISI

HALAMAN JUDUL	
HALAMAN PENGESAHAN	
HALAMAN PERNYATAAN ORISINALITAS	i
HALAMAN KESEPAKATAN PUBLIKASI	ii
KATA PENGANTAR	iii
ABSTRAK	v
<i>ABSTRACT</i>	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Permasalahan	2
1.3 Batasan.....	3
1.4 Tujuan	3
1.5 Manfaat	3
1.6 Sistematika Penulisan	3
BAB 2 KAJIAN PUSTAKA	5

2.1	Teori Penunjang	5
2.3.1	<i>Online Shop</i>	5
2.3.2	Data Mining	6
2.3.3	<i>Market Basket Analysis</i>	9
2.3.4	Algoritma Apriori	10
2.3.5	Flask	15
2.2	Penelitian Terkait	16
BAB 3	PERANCANGAN	21
3.1	Alat Penelitian	22
3.2	Arsitektur Sistem	22
3.2.1	Data Source	22
3.2.2	<i>Pre-Processing</i>	25
3.2.3	Modeling dengan Algoritma Apriori	28
3.2.4	Visualisasi berbasis Web	31
BAB 4	PEMBAHASAN DAN ANALISIS	34
4.1	Deskripsi Data	34
4.2	Proses Transformasi Data	37
4.3	Implementasi Model Algoritma Apriori	38
4.4	Hasil Rule Asosiasi	42
4.5	Analisis terhadap Hasil Rule	44

4.6	Analisis Hasil Evaluasi Model.....	44
4.7	Visualisasi Hasil	46
4.8	Perhitungan Manual Apriori	49
4.9	Interpretasi Aturan Asosiasi.....	52
BAB 5 KESIMPULAN DAN SARAN		56
5.1	Kesimpulan	56
5.2	Saran	57
DAFTAR PUSTAKA		59

DAFTAR GAMBAR

Gambar 2.1 <i>The Taxonomy of Data Mining</i>	6
Gambar 3.1 Tahapan Penelitian	21
Gambar 3.2 Arsitektur Sistem.....	22
Gambar 3.3 Visualisasi berbasis Web.....	32
Gambar 3.4 Skema Visualisasi Web.....	32
Gambar 4.1 Grafik Batang Total Penjualan per Produk	34
Gambar 4.2 Perbandingan Waktu Apriori vs FP-Growth vs ECLAT	45
Gambar 4.3 Perbandingan Jumlah Aturan Apriori vs FP-Growth vs ECLAT	45
Gambar 4.4 Tampilan awal <i>Interface Web</i>	48
Gambar 4.5 <i>Interface Web</i> Hasil Rancangan.....	49

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	17
Tabel 3.1 Contoh <i>Dataset</i>	23
Tabel 4.1 7 Kombinasi teratas Uji Sensitivitas	40
Tabel 4.2 10 Transaksi Teratas	50
Tabel 4.3 Pembentukan Aturan Manual.....	52
Tabel 4.4 Uji Kualitatif terhadap Pelaku Usaha.....	55

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pada era digitalisasi saat ini menyebabkan banyak perubahan dan perkembangan dalam bidang ekonomi, *social* budaya, teknologi maupun politik. Saat ini perkembangan teknologi telah menciptakan banyak terobosan baru diberbagai bidang sehingga merubah gaya hidup masyarakat. Salah satunya aktivitas berbelanja, saat ini berbelanja dapat melalui *online* yang membantu masyarakat untuk berbelanja tanpa tatap muka yang dibatasi oleh jarak dan waktu. *Online shopping* ialah kegiatan pembelian barang dan jasa melalui media Internet. Melalui belanja lewat Internet seorang pembeli bisa melihat terlebih dahulu barang dan jasa yang hendak ia belanjakan melalui web yang dipromosikan oleh penjual. *Platform* yang digunakan penjual untuk mempromosikan dagangannya yaitu *Instagram* dan *WhatsApp*.

Instagram dan *WhatsApp* adalah salah satu dari banyaknya platform *online shopping* yang dapat digunakan masyarakat untuk transaksi jual beli. Saat ini masyarakat senang berbelanja menggunakan *online shopping* karena mudah dan hemat waktu. Salah satu yang dibeli oleh masyarakat ialah *makanan*. Pada kasus ini diambil sebuah usaha yang bernama “Dapur Tiara”, ini adalah salah satu toko yang menjual berbagai macam makanan dan *pastry* dari Negara Indonesia, Jepang hingga *Western*. Persaingan di dunia bisnis tidaklah mudah, khususnya pada industri seperti menjual makanan ini. Sehingga perlu dilakukannya suatu strategi untuk memenangkan pasar khususnya pada penjualan makanan ini secara *online*. Dengan strategi yang baik sebuah produk akan terjual dengan cepat agar penjualan dapat meningkat, sehingga laba yang dihasilkan akan mencapai maksimal sebagaimana tujuan dari Dapur Tiara.

Pemanfaatan data transaksi penjualan Dapur Tiara selama ini hanya disimpan sebagai arsip. Padahal, data tersebut dapat dimanfaatkan dan diolah menjadi sebuah informasi penting dan berguna untuk peningkatan penjualan produk-produk maupun inovasi

produk. Dalam hal ini dapat dilakukannya analisis data transaksi untuk mendapatkan pola penjualan. Pola penjualan ini bisa menjadi informasi bagi Dapur Tiara untuk mengetahui apa saja produk yang paling sering dibeli oleh kosumen. Sehingga dari informasi pola penjualan tersebut pihak penjual dapat mengambil keputusan yang berhubungan dengan produk yang akan dijual.

Karena banyaknya data transaksi penjualan yang ada, tentu akan sulit data tersebut dianalisis atau diolah secara manual. Maka dari itu, perlu dilakukan dengan bantuan sistem sehingga mempermudah untuk mendapatkan pola penjualan tersebut. Adapun *output* dari pemrosesan tersebut akan menghasilkan informasi transaksi untuk membantu mengetahui pola penjualan atau produk apa yang diinginkan selanjutnya untuk dijual (*open pre-order*) berikutnya.

Penelitian ini menggunakan Algoritma Apriori yang mana dengan menggunakan metode ini akan memberikan hubungan antar item dalam data penjualan. Dalam hal ini produk makanan yang dibeli akan menjadi pola pembelian konsumen. Sehingga penerapan Algoritma Apriori akan membantu dalam membentuk kandidat kombinasi item yang mungkin akan terjadi. Lalu dilakukannya pengujian terhadap kombinasi-kombinasi tersebut apakah memenuhi parameter *minimum support* dan *minimum confidence* yang merupakan nilai ambang batas yang diberikan oleh pengguna. Output ini akan menghasilkan informasi yang nantinya bisa menjadi dasar pengambilan keputusan oleh pihak Dapur Tiara.

1.2 Permasalahan

Berdasarkan latar belakang yang telah diuraikan adapun permasalahan dalam Tesis ini adalah data transaksi dari penjualan makanan pada usaha Dapur Tiara tidak digunakan semaksimal mungkin. Sehingga mengakibatkan kesulitan dalam menemukan pola kombinasi item atau produk-produk makanan yang paling sering dibeli oleh konsumen. Dari hasil pola penjualan ini dapat digunakan sebagai bahan pertimbangan dalam menyusun strategi penjualan yakni rekomendasi pembuatan paket atau *bundle* menu dan rekomendasi penawaran produk tambahan atau biasa disebut sebagai *cross-selling*.

1.3 Batasan

Adapun batasan masalah dalam pembuatan tesis ini adalah:

1. Data yang dianalisis adalah data transaksi penjualan dari Dapur Tiara pada Bulan Agustus 2021 – Januari 2024
2. Metode *data mining* yang digunakan adalah metode asosiasi *Market Basket Analysis*.
3. Algoritma yang digunakan yaitu Algoritma Apriori.

1.4 Tujuan

Berdasarkan permasalahan yang ada, tujuan dari penelitian ini adalah membuat sistem rekomendasi produk bundling atau mengetahui implikasi lainnya berdasarkan pola transaksi penjualan makanan yang sering dibeli secara bersamaan.

1.5 Manfaat

Manfaat yang diharapkan dari penelitian ini yaitu dapat membantu pihak Dapur Tiara mengetahui rekomendasi kombinasi produk makanan apa saja yang dibeli konsumen secara bersamaan dalam satu waktu.

1.6 Sistematika Penulisan

Sistematika penulisan tesis ini secara keseluruhan terdiri dari empat bab, masing-masing terdiri dari beberapa sub bab. Adapun pokok pembahasan dari masing-masing bab tersebut secara garis besar sebagai berikut:

Bab 1 Pendahuluan

Bab ini menguraikan tentang latar belakang masalah, permasalahan, batasan masalah, tujuan, manfaat penelitian, dan sistematika penulisan.

Bab 2 Kajian Pustaka

Bab ini menguraikan beberapa hasil penelitian terdahulu dan landasan teori yang diperlukan untuk merancang sistem.

Bab 3 Desain Sistem

Bab ini menjelaskan tentang perancangan sistem terdiri dari perancangan sistem yang akan dibangun dimulai dari tahapan-tahapan pembangunan sistem berdasarkan analisis masalah.

Bab 4 Eksperimen dan Analisis

Bab ini diuraikan secara detail mengenai penerapan sistem berdasarkan desain sistem yang telah dirancang dan melakukan proses pengujian terhadap sistem yang dibuat dan menganalisis hasil yang telah dilakukan.

Bab 5 Penutup

Pada bab ini berisi kesimpulan yang merupakan jawaban atas rumusan masalah serta eksperimen dan analisis yang telah dilakukan, apabila sekiranya ada yang kurang dalam penelitian ini memungkinkan untuk pemberian saran setelah penarikan kesimpulan.

BAB 2

KAJIAN PUSTAKA

Bab ini membahas mengenai kajian teori dari permasalahan yang akan diteliti yaitu menerapkan penerapan *association rule* dengan algoritma apriori pada transaksi penjualan makanan melalui Instagram dan WhatsApp pada Usaha rumahan yaitu Dapur Tiara. Dimana istilah-istilah tersebut tentu perlu dipahami dengan teori-teori yang ada. Selain itu pada bab ini juga menjelaskan mengenai penelitian-penelitian terdahulu yang memiliki keterkaitan, agar didapatkan pemahaman keterbaruan dari penelitian ini.

2.1 Teori Penunjang

2.3.1 *Online Shop*

Online shop adalah sistem belanja secara online di mana pembeli bisa melakukan pembelian secara daring. Dari kemajuan internet ini pembeli tidak hanya bisa melakukan pembayaran secara daring, melainkan juga melakukan transaksi, melihat review produk, mengetahui kredibilitas suatu toko dan lainnya. Dengan kemajuan teknologi ini penjual maupun pembeli tidak perlu bertemu dan lebih praktis.

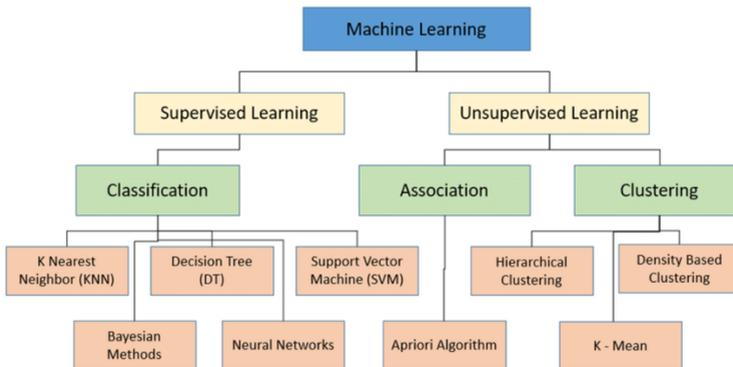
Membangun bisnis secara *online* melalui *online shop* terbilang lebih aman bagi anda yang masih pemula. Alasannya adalah karena tidak perlu membutuhkan modal yang begitu besar. Anda hanya perlu membeli beberapa stock barang untuk dijual kemudian sebar di internet. Lalu resikonya juga minim dari kerugian dan promosi yang lebih mudah.

Untuk membangun *online shop* bisa di lakukan di mana saja, baik sosmed ataupun anda bisa membangun *marketplace* sendiri di berbagai aplikasi-aplikasi *Online Shop*.(Alda Shabriani 2023)

2.3.2 Data Mining

Untuk prediksi masa depan. Teknik data mining menjadi bidang penelitian yang penting untuk analisis big data yang efektif seiring dengan meningkatnya kompleksitas dan ukuran data. Data mining digunakan di banyak domain yaitu *image mining*, *opinion mining*, *web mining*, *text mining*, *graph mining*, *medical data systems*. Ada dua kategori utama dalam data mining yang dikenal sebagai *supervised dan unsupervised*. Kedua pendekatan tersebut memiliki penerapan dan efisiensi yang berbeda dalam menganalisis dan memprediksi penyakit. Klasifikasi adalah teknik supervised di mana objek ditugaskan dalam koleksi ke kelas sasaran. Decision tree, ANN, SVM, NB, dll adalah klasifikasi. Pendekatan yang berbeda digunakan untuk tujuan yang berbeda dalam perawatan kesehatan.

Dalam pengelompokan, jenis objek yang serupa dikategorikan dalam kelompok yang sama. K-means, K-medoids, agglomerative, divisive, DBSCAN dll adalah beberapa teknik clustering. Asosiasi adalah kemungkinan terjadinya suatu benda dalam suatu himpunan. Klasifikasi selanjutnya dari Apriori adalah asosiasi. Hierarki data mining ditunjukkan pada Gambar 2.1. (Nabeel et al. 2021)



Gambar 2.1 The Taxonomy of Data Mining

Berdasarkan peran data mining dalam melakukan proses prediksi dan mendeskripsikan data, tugas data mining dapat dibagi ke dalam empat kelompok utama, yaitu:

1. Estimasi

Teknik untuk melakukan estimasi terhadap sebuah data baru yang tidak memiliki keputusan berdasarkan histori data yang telah ada, dimanavariabel target estimasi lebih ke arah numerik daripada ke arah kategori. Model dibangun menggunakan record lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi. Selanjutnya, pada peninjauan berikutnya estimasi nilai dari variabel target dibuat berdasarkan nilai dari variabel prediksi. Contohnya melakukan estimasi tekanan darah sistolik pada pasien rumah sakit berdasarkan umur pasien, jenis kelamin, indeks berat badan, dan level sodium darah. Hubungan antara tekanan darah sistolik dan nilai variabel prediksi dalam proses pembelajaran akan menghasilkan model estimasi. Model estimasi yang dihasilkan dapat digunakan untuk memprediksi kasus baru lainnya.

2. Klasifikasi

Suatu teknik dengan melihat pada kelakuan dan atribut dari kelompok yang telah didefinisikan. Teknik ini dapat memberikan klasifikasi pada data baru dengan memanipulasi data yang ada yang telah diklasifikasi dan dengan menggunakan hasilnya untuk memberikan sejumlah aturan. Salah satu contoh yang mudah dan populer adalah dengan decision tree yaitu salah satu teknik klasifikasi yang paling populer karena mudah untuk interpretasi seperti Algoritma C4.5, ID3 dan lain-lain. Contoh pemanfaatannya misalnya pada bidang akademik terkait klasifikasi siswa yang layak masuk kedalam kelas unggulan atau akselerasi di sekolah tertentu.

3. Asosiasi

Teknik untuk mengenali kelakuan dari kejadian-kejadian khusus atau proses dimana hubungan asosiasi muncul pada setiap kejadian. Adapun teknik pemecahan masalah yang sering digunakan seperti Algoritma Apriori. Contoh pemanfaatan Algoritma apriori yaitu pada bidang Marketing ketika sebuah

Minimarket melakukan tata letak produk yang dijual berdasarkan produk-produk mana yang paling sering dibeli konsumen, selain itu seperti tata letak buku yang dilakukan pustakawan di perpustakaan.

4. Klasterisasi

Teknik untuk mengelompokkan data dan membentuk kelas objek-objek yang memiliki kemiripan. Klaster adalah kumpulan record yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan dengan record-record dalam kluster lain. Proses klasterisasi berbeda dengan proses klasifikasi yaitu tidak adanya variable target dalam pengklusteran. Proses klasterisasi mencoba untuk melakukan pembagian terhadap keseluruhan data menjadi kelompok-kelompok yang memiliki kemiripan (homogen), dimana kemiripan data dalam satu kelompok tinggi (maksimal) dan kemiripan data dengan data pada kelompok lain rendah (minimal). Contoh klasterisasi misalkan proses mendapatkan kelompok-kelompok konsumen untuk target pemasaran dari suatu produk bagi perusahaan yang tidak memiliki dana pemasaran yang besar.(Gede et al. n.d.)

Berikut merupakan tahapan dari Data Mining di bawah ini.

1. **Pembersihan Data (*Data Cleaning*)**

Sebelum data diproses dan dibentuk menjadi sebuah *knowledge*, data yang ada harus dibersihkan terlebih dahulu. Jika ada yang mengandung eror, maka data-data tersebut harus dibuang. Sehingga tersisa data yang 'bagus' untuk diolah dalam tahap selanjutnya.

2. **Integrasi Data (*Data Integration*)**

Langkah kedua dalam tahapan data mining adalah integrasi data. Proses menggabungkan data dari berbagai sumber yang mungkin berbeda format, struktur, atau jenisnya menjadi satu set data yang konsisten dan terstruktur. Tujuannya adalah untuk menciptakan kumpulan data yang lebih lengkap dan bermakna yang dapat digunakan untuk analisis lebih lanjut.

3. **Transformation dan Aggregation**

Dalam tahap *transformation* data akan dipilih dan diubah formatnya agar sesuai dengan teknik atau metode yang dipakai. Pada tahap ini pula kualitas data mining akan terlihat. Agregasi (*aggregation*) adalah proses mengkombinasikan dua atau lebih objek ke dalam sebuah objek tunggal. Agregasi data sangat berguna ketika pada set data ada sejumlah nilai dalam satu fitur yang sebenarnya satu kelompok, yang tidak akan menyimpang dari deskripsi fitur tersebut jika nilainya digabungkan. Agregasi yang dapat dilakukan adalah sum (jumlah), average (rata-rata), min (terkecil), max (terbesar). (Radliya Nizar Rabbi 2015)

Ada beberapa alasan mengapa sebaiknya melakukan agregasi, diantaranya adalah:

- 1) Set data yang lebih kecil akan membutuhkan memori penyimpanan yang lebih sedikit (pengurangan data atau perubahan skala).
- 2) Waktu pemrosesan dalam algoritma data mining menjadi lebih cepat.
- 3) Agregasi bertindak untuk mengubah cara pandang terhadap data dari level rendah menjadi level tinggi.
- 4) Perilaku pengelompokan objek atau atribut sering kali lebih stabil dari pada objek individu itu sendiri (lebih sedikit variasinya)

4. Data Mining

Tahapan data mining selanjutnya adalah proses penambangan data itu sendiri. Perlu penentuan metode *mining* yang tepat.

5. Evaluasi Pola (*Pattern Evaluation*)

Setelah selesai melakukan proses data mining, pola-pola yang dihasilkan dari proses tersebut perlu untuk dievaluasi. Tujuan dari dilakukannya evaluasi adalah untuk menguji hipotesis awal. Setelah teruji data bisa dipresentasikan kepada pengguna.

2.3.3 Market Basket Analysis

Istilah *Market Basket Analysis* sendiri datang dari kejadian yang sudah sangat umum terjadi di dalam pasar swalayan, yakni

ketika para konsumen memasukkan semua barang yang mereka beli ke dalam keranjang yang umumnya telah disediakan oleh pihak swalayan itu sendiri. Informasi mengenai produk-produk yang biasanya dibeli secara bersamaan oleh para konsumen dapat memberikan “wawasan” tersendiri bagi para pengelola toko atau swalayan untuk menaikkan laba bisnisnya (Albion Research).

Informasi atau pengetahuan seperti di atas tentunya tidak hanya bermanfaat di dalam lingkungan pemasaran untuk pasaran swalayan saja. Beberapa bisnis yang bergerak di luar wilayah ini pun bisa menikmati manfaat dari adanya *Market Basket Analysis* ini. Sebut saja misalnya toko-toko virtual yang menjual produk-produknya secara online, bank-bank yang memberikan fasilitas layanan kartu kredit untuk para nasabahnya, perusahaan penyedia jasa asuransi, restoran *fast-food*, toko baju, toko buku, dan lain-lain.

Untuk beberapa kasus, pola dari barang-barang yang dibeli secara bersamaan oleh konsumen mudah untuk ditebak, misalnya susu dibeli bersamaan dengan roti. Namun, mungkin saja terdapat suatu pola pembelian barang-barang yang tidak pernah terpikirkan sebelumnya. Misalnya, pembelian minyak goreng dengan deterjen. Mungkin saja pola seperti ini tidak pernah terpikirkan sebelumnya karena minyak goreng dan deterjen tidak mempunyai hubungan sama sekali, baik sebagai barang pelengkap maupun barang pengganti. Hal ini mungkin tidak pernah terpikirkan sebelumnya sehingga tidak dapat diantisipasi jika terjadi sesuatu, seperti kekurangan stok deterjen misalnya. Inilah salah satu manfaat yang dapat diperoleh dari melakukan *Market Basket Analysis*. Dengan melakukan proses ini secara otomatis, seorang manajer tidak perlu mengalami kesulitan untuk menemukan pola barang-barang apa saja yang mungkin dibeli secara bersamaan. (Xandra Albert Sim et al. 2015)

2.3.4 Algoritma Apriori

Algoritma Apriori adalah salah satu algoritma yang melakukan pencarian frekuensi itemset dengan menggunakan teknik *association rule*. Algoritma Apriori menggunakan pengetahuan frekuensi atribut yang telah diketahui sebelumnya untuk memproses informasi selanjutnya. Pada algoritma apriori menentukan kandidat yang mungkin muncul dengan cara memperhatikan *minimum support*

dan *minimum confidence*. Support adalah nilai pengunjung atau persentase kombinasi sebuah item database. (Saefudin and DN 2019)

Cara algoritma ini bekerja adalah algoritma akan menghasilkan kandidat baru dari k-itemset dari frequent itemset pada langkah sebelumnya dan menghitung nilai support k-itemset tersebut. Itemset yang memiliki nilai *support* di bawah dari *minimum support* akan dihapus. Algoritma berhenti ketika tidak ada lagi *frequent itemset* baru yang dihasilkan. Kedua, dari hasil *frequent itemset* tersebut, langkah selanjutnya dihitung *minimum confidence* mengikuti rumus sesuai yang telah ditentukan. Nilai support tidak perlu dilihat lagi, karena *generate frequent itemset* didapatkan dari *minimum support*-nya. Bila *rule* yang didapatkan memenuhi batasan yang ditentukan dan batasan itu tinggi, maka *rule* tersebut tergolong *strong rules* (Adri Ahmad, Rumlaklak Nelci Dessy, and Sina Derwin R. 2021)

Adapun 2 tolak ukur dalam membentuk *rules* atau aturan dalam penerapan algoritma apriori adalah sebagai berikut: (HAIDAR 2021a)

a. *Support*

Support atau bisa juga disebut nilai penunjang adalah persentase dari laporan atau *record* yang didalamnya mengandung kombinasi item.

Persamaan (1) adalah rumus untuk mendapatkan nilai *support*.

$$Support (A) = \frac{Jumlah\ Transaksi\ Mengandung\ A}{Total\ Transaksi} \times 100\% \quad (1)$$

Persamaan (2) adalah rumus untuk mendapatkan nilai *support* dari suatu kombinasi item.

$$Support (A, B) = \frac{Jumlah\ Transaksi\ Mengandung\ A\ dan\ B}{Total\ Transaksi} \times 100\% \quad (2)$$

Yang mana nilai support A dan B diperoleh dari pembagian antara jumlah transaksi yang mengandung A dan B dengan total transaksi yang akhirnya dikalikan dengan 100%.

b. *Confidence*

Confidence atau bisa disebut nilai kepastian adalah kuatnya hubungan antar item dalam aturan asosiasi. Adapun rumus untuk mendapatkan nilai *confidence* adalah

$$\text{Confidence}(A, B) = \frac{\text{Jumlah Transaksi Mengandung A dan B}}{\text{Total Transaksi Mengandung A}} \quad (3)$$

Atau,

$$\text{Confidence}(A \rightarrow B) = \frac{\text{Support}(A, B)}{\text{Support}(A)} \quad (4)$$

Sedangkan rumus mendapatkan nilai persentase confidence ialah:

$$\text{Confidence}(A \rightarrow B) = \frac{\text{Support}(A, B)}{\text{Support}(A)} \times 100\% \quad (5)$$

Nilai *confidence* $A \rightarrow B$ diperoleh dengan membagi jumlah transaksi yang mengandung A dan B dengan total transaksi yang dikalikan dengan 100%.

Menurut(Ashari et al. 2022), Algoritma Apriori memiliki kekurangan atau kelemahan dari hal performa. Karena pada setiap penentuan frequent k-itemset harus melakukan scan database. Hal ini akan menjadi masalah apabila kandidat k-itemset memiliki dimensi yang banyak. Proses scan database yang besar akan memakan waktu yang lama dan berpengaruh pada penggunaan memori dan prosesor.

Menurut (Delima Sitanggang 2023), adapun kelebihan dan kelemahan dari algoritma Apriori adalah sebagai berikut, kelebihan algoritma Apriori:

1. Sederhana dan mudah dipahami di antara algoritma asosiasi.
2. Aturan yang dihasilkan intuitif dan mudah dikomunikasikan ke pengguna awam.
3. Tidak memerlukan data berlabel karena merupakan algoritma unsupervised .
4. Algoritmanya lengkap, sehingga dapat menemukan semua aturan dengan support dan confidence yang ditentukan.

Kelemahan dari algoritma apriori di antaranya:

1. Membutuhkan banyak perhitungan jika itemset sangat besar dan nilai support minimal dipertahankan seminimal mungkin.
2. Perlu melakukan pemindaian penuh seluruh database. Hal ini membuat proses pada algoritma Apriori berjalan sangat lama dan lambat serta menghabiskan banyak sumber daya.
3. Karena perlu melakukan pemindaian penuh, kompleksitas waktu dan ruang dari algoritma apriori adalah $O(2D)$, yang berarti sangat tinggi. Simbol D mewakili lebar horizontal yang ada dalam database.

Algoritma apriori digunakan untuk menemukan *association rules* yang memenuhi batas nilai *support* dan *confidence*. Cara kerja apriori adalah dengan menganalisa kumpulan item yang diambil atau dipilih secara bersamaan pada beberapa transaksi. penghitung kumpulan pola data yang muncul di dalam *database* melalui beberapa iterasi atau perulangan. Iterasi i menghitung semua kumpulan data i (kumpulan data yang mengandung elemen i) yang sering muncul. Setiap iterasi terdiri dari 2 langkah yaitu *candidate generation* (penentuan kandidat) dan *candidate counting and selection* (pemilihan sertapenghitungan kandidat).

Dalam tahap penghitungan, algoritma apriori menghitung nilai *support* terhadap tiap elemen dalam *database*. Hanya kumpulan data i dengan batas minimal yang telah ditentukan akan dimasukkan ke dalam kandidat barang yang sering muncul.

Pencarian pola kombinasi barang dalam apriori tidak dapat secara langsung menggabungkan elemen i dengan seluruh jenis barang yang diterima. Bila pencarian kombinasi barang dilakukan secara langsung maka waktu yang dibutuhkan untuk mencari kombinasi barang akan lebih lama. Cara yang dilakukan dalam algoritma apriori ini adalah dengan melakukan *self-joining* iterasi sebelumnya.

Proses utama yang dilakukan dalam algoritma apriori untuk menemukan frequent itemset yaitu

1. *Join* (Penggabungan)
Pada proses ini dilakukan pengkombinasian item dengan item yang lainnya hingga tidak dapat terbentuk kombinasi lagi.
2. *Prune* (Pemangkasan)

Proses ini merupakan hasil dari item yang telah dikombinasikan kemudian dipangkas dengan menggunakan *minimum support* yang telah ditentukan *user*.

Langkah pertama pada algoritma apriori yaitu, support dari setiap item dihitung dengan men-*scan database*. Setelah *support* dari setiap item didapat, item yang memiliki *support* lebih besar dari *minimum support* dipilih sebagai pola frekuensi tinggi dengan panjang 1 atau sering disingkat 1- itemset. Singkatan k-itemset berarti satu set yang terdiri dari k item.

Iterasi kedua menghasilkan 2-itemset yang tiap set-nya memiliki dua item. Pertama dibuat kandidat 2-itemset dari kombinasi semua 1-itemset. Lalu untuk tiap kandidat 2-itemset ini dihitung *support*-nya dengan men-*scan database*. *Support* artinya jumlah transaksi dalam database yang mengandung kedua item dalam kandidat 2-itemset. Setelah support dari semua kandidat 2- itemset didapatkan, kandidat 2-itemset yang memenuhi syarat minimum support dapat ditetapkan sebagai 2-itemset yang juga merupakan pola frekuensi tinggi.

Untuk selanjutnya iterasi-iterasi ke-k dapat dibagi lagi menjadi beberapa bagian :

1. Pembentukan kandidat itemset Kandidat k-itemset dibentuk dari kombinasi (k-1)- itemset yang didapat dari iterasi sebelumnya. Satu ciri dari algoritma apriori adalah pemangkasan kandidat kitemset yang subset-nya yang berisi k-1 item tidak termasuk dalam pola frekuensi tinggi dengan panjang k-1.
2. Penghitungan support dari tiap kandidat k-itemset Support dari tiap kandidat k-itemset didapat dengan men-*scan database* untuk menghitung jumlah transaksi yang memuat semua item di dalam kandidat k-itemset tersebut. Ini adalah juga ciri dari algoritma apriori yaitu diperlukan penghitungan dengan scan seluruh database sebanyak k-itemset terpanjang.
3. Tetapkan pola frekuensi tinggi Pola frekuensi tinggi yang memuat k item atau kitemset ditetapkan dari kandidat k-itemset yang support-nya lebih besar dari minimum support. Kemudian dihitung confidence masing-masing kombinasi

item. Iterasi berhenti ketika semua item telah dihitung sampai tidak ada kombinasi item lagi. (HAIDAR 2021b)

2.3.5 Flask

Flask adalah *framework web* yang ditulis menggunakan bahasa pemrograman Python. Seperti framework lain, Flask memudahkan developer membuat aplikasi, karena menyediakan template script yang bisa dipakai oleh developer kapan saja. Secara kategori, Flask termasuk dalam micro framework. Artinya, Flask adalah *framework* minimalis yang fiturnya tidak selengkap full stack framework. Namun, karena ukuran framework Python ini cukup ringan, kinerja Flask menjadi cenderung lebih gesit. Selain itu, Flask sangat cocok dengan berbagai macam extensions yang bisa dimanfaatkan oleh developer untuk menambah fitur-fiturnya. Salah satu contoh *extensions* untuk Flask adalah *Google App Engine*. (Mufid 2023)

Mayoritas fungsi dan komponen umum, seperti *database*, validasi form, dan lain-lain bukanlah fungsi bawaan Flask. Hal ini karena fungsi dari komponen-komponen tersebut telah disediakan oleh pihak ketiga. Flask dapat memakai ekstensi yang membuat fitur serta komponen-komponen tersebut seolah-olah diterapkan oleh Flask sendiri. Karena termasuk *micro framework*, Flask dapat membuat core aplikasi sesederhana mungkin sehingga tetap dapat dengan mudah ditambahkan. Hasilnya, fleksibilitas dan skalabilitas Flask cukup tinggi jika dibandingkan dengan framework lain.

Semua *framework* pasti memiliki kelebihan dan kekurangan yang dapat menjadi bahan pertimbangan, dan disesuaikan dengan kebutuhan. Flask adalah framework yang mempunyai beberapa kelebihan bila diterapkan dalam sebuah aplikasi web, di antaranya: (Mufid 2023)

1. Sederhana

Flask adalah *framework* yang cocok digunakan untuk pemula. Selain mudah dipahami, Flask juga memiliki tampilan yang cukup sederhana. Sebagai developer, Anda akan dimudahkan dalam membuat navigasi serta aplikasi. Flask memberi Anda kendali penuh dalam pengembangan web. Inilah yang membedakan Flask dengan framework yang lain.

2. Fleksibel Berbeda dengan framework lain, Flask adalah framework minimalis yang setiap bagiannya dapat dibuka dan diubah. Flask juga menawarkan template yang memungkinkan Anda memakai user interface yang sama untuk halaman berbeda.
3. Testing Dengan Flask, Anda dapat melakukan pengujian unit melalui dukungan terintegrasi, debugger cepat, server pengembangan bawaan dan permintaan RESTful. Selain itu, Anda juga bisa transit ke kerangka web secara mudah, cukup dengan beberapa ekstensi. Hal ini dapat terjadi karena Flask yang sangat ringan.

2.2 Penelitian Terkait

Penelitian ini merupakan studi literatur yang merujuk beberapa penelitian terdahulu yang relevan dengan permasalahan yang akan diteliti mengenai Algoritma Apriori. Berikut merupakan penelitian-penelitian terdahulu yang merujuk ke penelitian penulis.

Tabel 2.1 Penelitian Terdahulu

No	<i>Author</i> (Tahun)	Judul Penelitian	Tujuan Penelitian	Hasil Penelitian
1	(HAIDAR 2021)	Implementasi Algoritma Apriori untuk Mencari Pola Transaksi Penjualan Carroll Kitchen	Untuk membuat sistem data mining untuk pola penjualan dengan metode algoritma apriori untuk memberikan gambaran keterkaitan antar barang dengan menganalisis data transaksi penjualan berdasarkan dari produk yang sering dibeli secara bersamaan.	Hasil analisis yang didapatkan setelah menggunakan perbandingan minimum support dan minimum confidence yang berbeda-beda berdasarkan data transaksi yang ada adalah dengan menggunakan minimum support 4% (kuatnya kombinasi item tersebut dalam database) dan minimum confidence 30% (kuatnya hubungan antar item dalam aturan asosiasi) menghasilkan dua puluh empat aturan asosiasi. Salah satu contohnya yaitu jika konsumen membeli menu Teh manis (dingin) maka 71,05% (kepastian konsumen dalam membeli item) akan membeli Nasi goreng jambal.

2	(Erfina, Melawati, and Destria Arianti 2020)	Penerapan Metode Algoritma Apriori Terhadap Data Transaksi Penjualan Toko Fasentro Fancy	Untuk membantu toko mengetahui produk atau barang yang sering dibeli secara bersamaan, sehingga memudahkan mereka dalam penataan tata letak.	<p>Berdasarkan data transaksi penjualan pada toko fasentri fancy dilakukan analisis menggunakan algoritma apriori dengan parameter minimum support 33,33% dan minimum confidence 80%. Dalam penelitian ini dihasilkan 4 aturan asosiasi kombinasi itemset yaitu 2 kombinasi itemset dan 2 kombinasi 3 itemset yang memenuhi nilai minimum support dan nilai minimum confidence yang telah ditentukan.</p> <p>Berdasarkan hasil aturan asosiasi final yang memenuhi nilai support dan nilai confidence tertinggi yaitu jika Pranaya (P24TVC), Toyobu Fodu maka akan membeli Roberto Cavali.</p>
---	--	--	--	--

3	(Tida and Zubair 2022)	Analisa Transaksi Penjualan Menggunakan Metode Algoritma Apriori Studi Kasus Toko Busana Sakato	Mencari pola pembelian konsumen pada Toko Busana Sakato dengan menggunakan Algoritma Apriori.	Menghasilkan 7 aturan asosiasi terbaik dari pengujian data dengan minimum support 30% dan minimum confidence 60%.
4	(Adri Ahmad et al. 2021)	Implementasi Algoritma Apriori Untuk Analisa Data Penjualan (Studi Kasus: Toko UD. Suryani)	Menerapkan Data mining pada data transaksi satu tahun terakhir yang ada ditoko UD. Suryani	Memberikan rekomendasi bagi UD. Suryani untuk menambah jumlah stok untuk item yang sering dibeli, penempatan rak yang berdekatan untuk kombinasi item yang dibeli bersamaan dan dapat mengadakan promo terhadap item tersebut.

Berdasarkan keempat penelitian terkait, dapat disimpulkan bahwa algoritma Apriori telah banyak digunakan untuk menemukan pola keterkaitan antar produk yang sering dibeli secara bersamaan di berbagai bidang usaha. Penelitian (HAIDAR 2021b) berhasil menemukan 24 aturan asosiasi dengan parameter *minimum support* 4% dan *minimum confidence* 30%, sedangkan (Erfina et al. 2020) menemukan 4 aturan asosiasi dengan nilai tertinggi pada kombinasi produk tertentu. Penelitian (Tida and Zubair 2022) menghasilkan 7

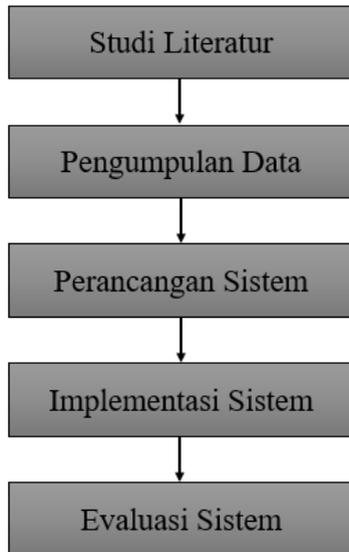
aturan asosiasi terbaik dengan minimum support 30% dan minimum confidence 60%, sementara (Adri Ahmad et al. 2021) memanfaatkan hasil Apriori untuk memberikan rekomendasi penambahan stok, penempatan rak, dan promo produk.

Dari keempat penelitian terkait tersebut, terlihat bahwa penerapan algoritma Apriori secara konsisten mampu mengidentifikasi hubungan keterkaitan antar produk yang sering dibeli secara bersamaan, sehingga memberikan manfaat nyata bagi pengambilan keputusan, seperti pengaturan tata letak produk, penambahan stok, ataupun strategi promosi. Setiap penelitian memiliki perbedaan pada parameter *minimum support* dan *minimum confidence* yang digunakan, jumlah aturan yang dihasilkan, serta jenis usaha yang menjadi objek studi. Hal ini menunjukkan bahwa fleksibilitas algoritma Apriori memungkinkan penerapannya pada berbagai konteks bisnis dengan hasil yang relevan. Penelitian ini memosisikan diri untuk melanjutkan dan mengembangkan temuan-temuan sebelumnya dengan studi kasus pada bisnis makanan online Dapur Tiara, yang bertujuan menghasilkan aturan asosiasi sebagai dasar strategi *cross-selling* yang efektif, sehingga dapat mendukung peningkatan penjualan dan efisiensi pemasaran.

BAB 3

PERANCANGAN

Desain sistem adalah suatu kegiatan perancangan atau *design* sebelum aplikasi dibuat dan merupakan tahapan setelah Analisis sistem. Dimulai dari latar belakang masalah penelitian yaitu dari adanya data transaksi usaha ini tidak digunakan semaksimal mungkin. Sehingga mengakibatkan kesulitan dalam menemukan kombinasi item atau produk-produk makanan yang paling sering dibeli oleh konsumen. Dalam bab ini akan dijelaskan mengenai analisis dan perancangan sistem tentang penelitian ini pada Gambar 3.1 berikut:



Gambar 3.1 Tahapan Penelitian

3.1 Alat Penelitian

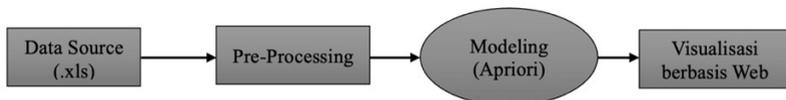
Alat yang digunakan dalam penelitian ini adalah sebuah Laptop. Dengan spesifikasi yang cukup dan mendukung untuk menjalankan Sistem Operasi. Serta pengembangan *software*, dan bisa terkoneksi dengan jaringan internet.

Sistem Operasi, program-program, dan alat-alat yang akan digunakan dalam pengembangan penelitian ini meliputi *hardware* dan *software* sebagai berikut:

- a. Sistem operasi macOS Sequoia 15.3.1.
- b. Python sebagai bahasa pemrograman.
- c. Flask sebagai *framework* python
- d. Google colabs sebagai perangkat *notebook* python untuk melakukan pengujian
- e. Safari sebagai web browser internet
- f. Visual studio sebagai *editor* bahasa pemrograman

3.2 Arsitektur Sistem

Arsitektur Sistem merupakan implementasi dari perancangan Sistem Penerapan Algoritma Apriori Untuk Mencari Pola Penjualan pada Usaha Dapur Tiara. Arsitektur Sistem yang digunakan untuk memudahkan penelitian melakukan penelitian dapat dilihat pada gambar dibawah ini:



Gambar 3.2 Arsitektur Sistem

3.2.1 Data Source

Informasi mengenai pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Koleksi Data

Meminta secara langsung data transaksi penjualan pada periode tertentu kepada *Owner* Dapur Tiara untuk digunakan dalam penelitian.

2. Karakteristik Data

Data yang digunakan dalam penelitian ini adalah data hasil penjualan Dapur Tiara yang diambil dengan rentan waktu 2021 hingga 2024 yang berjumlah lebih dari 500 data yang belum diolah sama sekali. Data tersebut akan menjadi data yang akan diimplementasikan metode asosiasi dengan menggunakan *Association Rules* dan pembentukan pola menggunakan *Apriori Algorithm*. Data berformat (.xls). Rancangan ini berisikan tahapan data mining.

3. Parameter Data

Parameter data pada penelitian ini ada ID Transaksi, Daftar Produk, Jumlah, dan Tanggal Transaksi. Jenis data dapat dilihat pada Tabel 3.2 dan contoh sebagian data dari Dapur Tiara dapat dilihat pada Tabel 3.1 dibawah ini.

Tabel 3.1 Contoh Dataset

ID Transaksi	Produk	Jumlah	Tanggal Transaksi
1	{PP,BP}	2	26/08/21
2	{PP,BP,CP}	3	26/08/21
3	{PP}	2	26/08/21
4	{'TP', 'NUB'}	3	26/08/21
5	{'BP', 'MP', 'NUB', 'TP', 'TNK'}	3	26/08/21
6	{'PP', 'CP', 'TP', 'BP'}	2	26/08/21
7	{'NUB', 'CP'}	3	26/08/21
8	{'BP', 'MP', 'TNK'}	3	26/08/21
9	{'BP', 'MP', 'CP', 'TP'}	3	26/08/21
10	{'MP', 'PP', 'NUB'}	3	26/08/21
11	{'PP', 'CP', 'BP', 'TP', 'MP'}	1	26/08/21

12	{'PP', 'NUB', 'NKB', 'TNK', 'CP'}	2	26/08/21
13	{'NKB', 'CP', 'PP', 'MP', 'TNK'}	2	26/08/21
14	{'PP', 'MP', 'CP', 'TNK', 'NUB'}	2	26/08/21
15	{'CP'}	3	26/08/21
16	{'NUB', 'TP', 'NKB', 'TNK'}	2	26/08/21
17	{'PP', 'TNK', 'BP'}	2	26/08/21
18	{'TNK', 'BP', 'TP'}	2	26/08/21
19	{'PP', 'NUB', 'NKB'}	3	26/08/21
20	{'NKB', 'PP', 'BP', 'TNK'}	2	24/09/21

Ket	
MP=Mini Pizza	L=Lasagna
PP=Pepperoni Pizza	CT=Chicken Takoyaki
BP=Beef Pizza	BT=Beef Takoyaki
TP=Tuna Pizza	SCW=Soft Cookies Walnut
CP=Chicken Pizza	SCM=Soft Cookies Marshmallow
NKB=Nasi Kuning Bento	SCDC=Soft Cookies Dark Chocolate
TNK=Tumpeng Nasi Kuning	SCRV=Soft Cookies Red Velvet
NUB=Nasi Uduk Bento	SCMt=Soft Cookies Matcha
BO=Beef Onigiri	RT=Roti Tawar
TO=Tuna Onigiri	RG=Roti Gandum
CO=Chicken Onigiri	

Untuk menggunakan *Apriori Algorithm* dibutuhkan data yang berjumlah banyak. Dimana setiap dataset harus memenuhi syarat untuk dilakukan implementasi data. Dapur Tiara menyediakan data yang bersifat *raw* dan setelah melakukan riset bahwasanya data tersebut memenuhi syarat atau cocok untuk dilakukan implementasi *Market Basket Analysis* dan *Apriori Algorithm*.

3.2.2 *Pre-Processing*

Dalam penelitian ini, data transaksi penjualan online perlu melalui beberapa tahap *pre-processing* sebelum dilakukan analisis menggunakan algoritma Apriori. Tahapan awal dalam *pre-processing* ini bertujuan untuk membersihkan, mentransformasi, dan memformat data agar sesuai dengan kebutuhan algoritma. Adapun tahapan-tahapannya adalah sebagai berikut:

1. *Data Collection*

Proses *data collection* dilakukan dengan mengumpulkan data transaksi penjualan sebagai *dataset* yang akan digunakan dalam penelitian. *Dataset* yang akan digunakan adalah data dari Dapur Tiara dari bulan agustus 2021 hingga januari 2024. Jumlah data yang digunakan adalah sebanyak 2432 baris data penjualan makanan berformat (.xls).

Adapun fitur yang digunakan adalah data yang berisi ID transaksi, daftar produk, jumlah, dan tanggal transaksi.

2. *Filtering Data*

Tahap ini bertujuan untuk menyaring data transaksi sehingga hanya data yang relevan dan mendukung proses pembentukan aturan asosiasi yang digunakan dalam analisis. Pada penelitian ini, data transaksi yang hanya memiliki satu jenis produk dihapus dari dataset karena transaksi tersebut tidak dapat membentuk kombinasi item (*itemset*) yang diperlukan dalam penerapan algoritma Apriori.

Proses ini dilakukan dengan menghitung jumlah produk pada setiap transaksi, kemudian hanya mempertahankan transaksi dengan jumlah produk lebih dari satu. Penyaringan ini

bertujuan untuk memastikan bahwa algoritma dapat mengidentifikasi hubungan antarproduk secara optimal, karena setidaknya dibutuhkan dua item dalam satu transaksi untuk dapat menghasilkan aturan asosiasi. Implementasi penyaringan data dalam kode program adalah sebagai berikut:

```
df['jumlah_item'] = df['Produk'].apply(count
_itemset)
df_multi_item = df[df['jumlah_item'] >
1].drop(columns=['jumlah_item'])
```

Hasil dari tahap ini adalah dataset yang lebih terfokus, yang hanya berisi transaksi dengan jumlah item lebih dari satu. Dataset hasil penyaringan ini kemudian digunakan pada tahap selanjutnya yaitu *data transformation* untuk diubah menjadi format biner yang sesuai dengan kebutuhan algoritma Apriori.

3. Data Cleaning

Data cleaning pada penelitian ini sangat penting karena dapat melakukan pembersihan data yang memiliki *noise* dan data yang tidak konsisten seperti salah memberi kode, *human error*, *typo*, maupun data yang duplikat.

Setelah penyaringan data, proses selanjutnya yaitu pembersihan data dilakukan untuk menghilangkan karakter-karakter yang tidak diperlukan pada kolom Produk, seperti tanda kutip ('), tanda kurung kurawal ({ }) dan tanda kurung biasa (()). Karakter-karakter tersebut berpotensi mengganggu pemrosesan data lebih lanjut. Berikut adalah salah satu baris kode yang melakukan *data cleaning*:

```
df['Produk'] = df['Produk'].apply(lambda x:
re.sub(r"[\{\}\(\)\"]", '', x)).str.split(',')
```

Setelah proses ini, data pada kolom Produk menjadi lebih bersih dan siap untuk diproses ke tahap berikutnya.

4. Data Integration

Dalam penelitian ini tidak dilakukan proses *data intergration* atau penggabungan data karena seluruh data yang digunakan berasal dari satu sumber Tunggal yaitu dari File Excel berisi Riwayat transaksi penjualan online Dapur Tiara. Tidak ada penggabungan (merge) atau konsolidasi dari beberapa sumber data lain.

5. Transformation dan Aggregation

Proses *data selection* dan *transformation* adalah dua proses penting dalam analisis data. Transformasi data diperlukan untuk mengubah data transaksi menjadi format yang sesuai untuk analisis asosiasi. Pada tahapan ini, data pada kolom Produk yang sebelumnya berupa string dipisahkan berdasarkan tanda koma menjadi list item. Kemudian, digunakan fungsi `explode()` untuk memecah list tersebut menjadi beberapa baris transaksi individual. Tahapan ini dilakukan dengan kode berikut:

```
basket = df_transactions.pivot
_table(index='ID Transaksi',
columns='Produk', values='Value',aggfunc=
'sum', fill_value=0)
basket = basket.applymap(lambda x: 1 if x >
0 else 0).astype(bool)
```

Setelah data dipecah perproduk, selanjutnya dilakukan pembuatan pivot table untuk membentuk matriks transaksi-produk, dengan setiap baris mewakili satu transaksi dan setiap kolom mewakili satu item produk. Nilai 1 menunjukkan bahwa produk tersebut ada dalam transaksi tersebut, sedangkan 0 berarti produk tidak ada. Ringkasnya data transaksi yang semula berupa teks produk dipisah menjadi list item per transaksi, kemudian diubah menjadi format *pivot table* biner (0 dan 1). Dimana:

- Setiap baris mewakili satu transaksi,
- Setiap kolom mewakili satu produk,

- Nilai 1 menunjukkan bahwa produk tersebut ada dalam transaksi.

Data ini kemudian dikonversi ke dalam format Boolean (True/False) agar sesuai dengan input yang dibutuhkan oleh algoritma Apriori.

6. *Data Mining*

Metode *data mining* yang digunakan adalah metode asosiasi dengan menggunakan Algoritma Apriori. Algoritma Apriori digunakan untuk menemukan pola-pola asosiasi dalam data yang berisi banyak transaksi atau itemset. Algoritma ini dipilih karena pada algoritma ini melakukan *scan database* setiap kali iterasi.

3.2.3 Modeling dengan Algoritma Apriori

Modeling yang digunakan adalah menggunakan Teknik Algoritma Apriori untuk menemukan aturan asosiasi antar produk yang sering dibeli bersamaan berdasarkan data transaksi yang telah melalui tahap *pre-processing*. Proses *pre-processing* mencakup *data collection*, *data cleaning*, *data transformation* dan *aggregation* dimana data mentah diubah menjadi format yang sesuai untuk analisis asosiasi, sebagaimana telah dijelaskan pada bagian sebelumnya.

Setelah data berupa dalam bentuk pivot table biner (0 dan 1), proses modelling dilakukan melalui beberapa langkah berikut:

1. Pembentukan *Frequent Itemsets*

Pembentukan *frequent itemsets* merupakan tahapan awal dalam algoritma Apriori yang bertujuan mengidentifikasi kombinasi item yang paling sering muncul bersamaan dalam data transaksi. Proses dimulai dengan menentukan nilai *minimum support*, kemudian menghitung frekuensi relatif setiap kombinasi item (itemsets). Pada tahap pembentukan frequent itemsets, algoritma Apriori bekerja secara iteratif (*level-wise search*) untuk membangun

kombinasi item mulai dari 1-itemset, kemudian 2-itemset, dan seterusnya.

Proses ini diawali dengan menghitung nilai *support* untuk setiap item tunggal (*1-itemset*). Item yang tidak memenuhi ambang *minimum support* akan dieliminasi. Selanjutnya, algoritma membentuk kandidat kombinasi item baru (*k-itemset*) dari itemset yang lolos seleksi pada tahap sebelumnya (*k-1 itemset*). Pada setiap iterasi, nilai *support* kembali dihitung, dan itemset yang tidak memenuhi *minimum support* kembali dibuang. Proses ini terus berulang hingga tidak ada lagi kombinasi item yang memenuhi syarat, atau tidak mungkin membentuk kandidat kombinasi baru.

Dengan mekanisme ini, pada setiap pasangan nilai *minimum support* dan *minimum confidence* yang diuji, model akan secara otomatis berhenti ketika seluruh kandidat itemset yang tersisa tidak memenuhi syarat. Walaupun dalam penelitian ini misalkan hasilnya hanya menghasilkan dua kombinasi parameter yang membentuk aturan asosiasi, proses pencarian tetap melalui iterasi penuh sampai kondisi berhenti (*stop condition*) terpenuhi.

Untuk memperoleh hasil yang optimal, proses pembentukan *frequent itemsets* dilakukan dengan mencoba beberapa nilai *minimum support*. Penelitian ini menerapkan proses pembentukan *frequent itemsets* dengan menggunakan variasi nilai *minimum support* pada rentang 0–1, yaitu dengan rentang 0.1, 0.2, hingga 1. Variasi nilai parameter ini bertujuan untuk mengevaluasi dan membandingkan performa model pada setiap tingkat pengaturan, sehingga dapat diketahui pengaruhnya terhadap jumlah kombinasi item (*frequent itemsets*) yang terbentuk serta kualitas aturan asosiasi yang dihasilkan. Penggunaan rentang parameter yang luas diharapkan memberikan gambaran yang lebih komprehensif mengenai sensitivitas model terhadap perubahan nilai *support* dan *confidence*.

2. Pembentukan Aturan Asosiasi

Setelah *frequent itemsets* terbentuk, langkah selanjutnya adalah pembentukan aturan asosiasi (*association rules*). Aturan asosiasi merupakan hasil utama dari algoritma Apriori yang menggambarkan hubungan keterkaitan antar item dalam transaksi. Proses ini dilakukan menggunakan fungsi *association_rules* dari pustaka *mlxtend*, yang memungkinkan pembuatan aturan berdasarkan *frequent itemsets* yang telah diperoleh sebelumnya.

Proses pembentukan aturan asosiasi menghasilkan kumpulan aturan yang memenuhi nilai *minimum confidence* tertentu. Aturan-aturan asosiasi ini dibentuk menggunakan nilai *confidence* sebagai acuan utama. Dalam penelitian ini digunakan beberapa variasi *minimum confidence* untuk mengamati pengaruhnya terhadap pembentukan aturan asosiasi. Nilai *confidence* ini yang di coba yaitu mulai dari 0.1 hingga 1. Tujuan dari variasi ini adalah untuk melihat sejauh mana kekuatan hubungan antar produk berubah berdasarkan ambang *confidence* yang berbeda.

3. Evaluasi Aturan

Setelah aturan asosiasi terbentuk, langkah selanjutnya adalah melakukan evaluasi terhadap performa model dengan membandingkan hasil aturan asosiasi yang dihasilkan oleh algoritma Apriori dengan *-Growth* dan ECLAT. Tujuan perbandingan ini adalah untuk melihat perbedaan jumlah aturan yang dihasilkan, tingkat efisiensi, serta kesesuaian pola yang ditemukan antara kedua metode tersebut.

FP-Growth dan ECLAT dipilih sebagai metode pembandingan karena keduanya memiliki kemampuan untuk menemukan *frequent itemset* tanpa perlu melakukan proses pencarian kandidat itemset seperti pada Apriori. Algoritma *FP-Growth* memanfaatkan struktur data *FP-Tree (Frequent Pattern Tree)* yang mampu mengompresi dataset dan mempercepat pencarian pola, sedangkan algoritma ECLAT

menggunakan pendekan *tidlist intersection* untuk menghitung dukungan (support) secara efisien.

Dengan demikian, perbandingan ini diharapkan dapat menunjukkan apakah FP-Growth dan ECLAT mampu memberikan hasil aturan asosiasi yang serupa atau bahkan lebih optimal dibandingkan Apriori pada data transaksi penjualan Dapur Tiara.

3.2.4 Visualisasi berbasis Web

Setelah proses pre-processing, pembentukan model asosiasi menggunakan algoritma Apriori, dan evaluasi model selesai dilakukan, tahap selanjutnya adalah menampilkan hasil tersebut melalui visualisasi berbasis web. Tujuannya untuk menyajikan hasil analisis algoritma Apriori secara interaktif dan mudah dipahami, baik oleh peneliti maupun pihak-pihak lain yang berkepentingan.

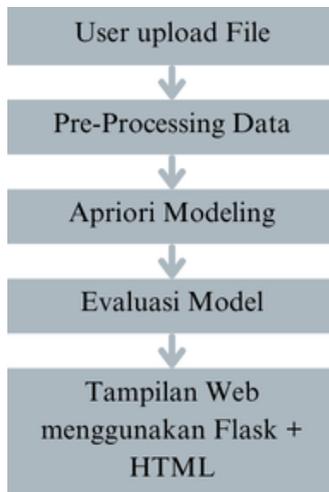
Pada tahapan ini visualisasi dilakukan dengan teknologi berbasis web menggunakan *framework* Flask, yaitu framework web berbasis Python yang ringan dan fleksibel. Pada halaman awal sistem web, *user* diarahkan untuk meng-*upload* file dataset dalam format Excel (.xls atau .xlsx). Setelah proses *upload* dilakukan, sistem secara otomatis akan memproses data tersebut dan menampilkan hasil analisis. Mulai dari *pre-processing*, pembentukan *frequent itemsets*, pembuatan aturan asosiasi, hingga evaluasi model dengan menggunakan data latih dan data uji.

Hasil aturan asosiasi yang diperoleh ditampilkan dalam bentuk tabel HTML yang berisi informasi mengenai kombinasi produk, nilai *support*, *confidence*, dan lift. Kalimat-kalimat *interpretative* juga disediakan untuk memudahkan pemahaman terhadap pola-pola yang ditemukan. Berikut merupakan skema visualisasi web:



Gambar 3.3 Visualisasi berbasis Web

Dengan pendekatan ini, proses visualisasi menjadi lebih dinamis karena *user* dapat langsung meng-*upload* data baru dan melihat hasil analisis tanpa perlu menjalankan ulang script Python secara manual.



Gambar 3.4 Skema Visualisasi Web

Arsitektur visualisasi web menjelaskan bagaimana sistem dibangun untuk menampilkan hasil analisis algoritma Apriori secara

dinamis melalui antarmuka berbasis web. Gambar 3.4 di atas merupakan skema visualisasi web. Tujuannya adalah untuk memperjelas pemahaman alur data dari *backend* ke *frontend*, serta menggambarkan peran masing-masing komponen dalam sistem.

Proses perancangan mencakup pemilihan alat dan teknologi yang digunakan, arsitektur sistem secara keseluruhan, serta tahapan-tahapan dalam data mining seperti *collecting data*, *filtering data*, *pre-processing*, *modelling*, evaluasi model, dan visualisasi hasil analisis.

BAB 4

PEMBAHASAN DAN ANALISIS

4.1 Deskripsi Data

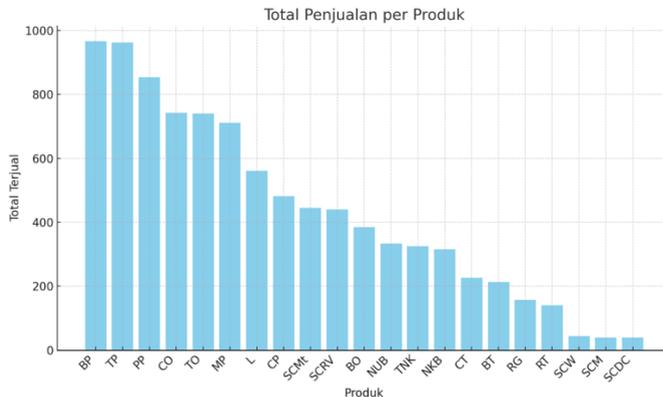
Pada tahap ini dijelaskan karakteristik dari data yang digunakan dalam penelitian.

1. Sumber Data

Data diperoleh dari riwayat transaksi penjualan pada bisnis makanan online Dapur Tiara.

2. Jumlah Transaksi

Dataset yang akan digunakan adalah data dari Dapur Tiara dari bulan agustus 2021 hingga januari 2024. Jumlah transaksi yang digunakan adalah sebanyak 2432 data penjualan makanan berformat (.xls).



Gambar 4.1 Grafik Batang Total Penjualan per Produk

Grafik batang di atas menunjukkan total penjualan untuk setiap produk berdasarkan data asli. BP (Beef Pizza) dan TP (Tuna Pizza) menjadi dua produk dengan jumlah transaksi tertinggi, masing-masing mendekati 1.000 kali penjualan.

Produk seperti SCW, SCM, dan SCDC berada di urutan terbawah, dengan penjualan di bawah 50 kali.

3. Parameter yang Digunakan

Parameter utama dalam dataset ini adalah:

- ID Transaksi : sebagai identifikasi unik tiap transaksi.
- Produk : berisi daftar nama produk yang dibeli dalam satu transaksi

4. Proses Penyaringan Data (*Filtering Data*)

Pada tahap ini dilakukan proses *filtering data* untuk memastikan bahwa data transaksi yang digunakan dalam pembentukan *frequent itemsets* memenuhi syarat analisis algoritma Apriori. Filtering data merupakan bagian dari *data preprocessing* yang bertujuan untuk menghapus data yang tidak relevan atau tidak memenuhi kriteria tertentu, sehingga data yang digunakan menjadi lebih bersih, konsisten, dan siap untuk dianalisis.

Dataset yang digunakan pada penelitian ini berisi informasi mengenai ID Transaksi, Tanggal, dan Produk yang dibeli. Setiap transaksi dapat terdiri dari satu atau lebih produk. Namun, dalam konteks algoritma Apriori, transaksi yang hanya memiliki satu item tidak dapat membentuk kombinasi *itemset* yang bermanfaat untuk aturan asosiasi, sehingga transaksi tersebut perlu dihilangkan.

Proses filtering data dilakukan melalui langkah-langkah berikut:

1) Menghapus Baris Kosong pada Kolom Produk

Baris yang tidak memiliki nilai pada kolom *Produk* dihapus untuk menghindari kesalahan dalam proses analisis. Hal ini dilakukan menggunakan perintah:

```
df = df.dropna(subset=['Produk'])
```

Dengan demikian, hanya transaksi yang memiliki data produk akan dipertahankan.

2) Menghitung Jumlah Item dalam Satu Transaksi

Untuk mengetahui jumlah produk yang dibeli dalam setiap transaksi, dilakukan perhitungan jumlah item menggunakan fungsi:

```
def count_itemset(Produk):  
    return len([item.strip() for item  
in str(Produk).split(',') if  
item.strip() != ''])
```

Fungsi ini memisahkan produk berdasarkan tanda koma, menghapus spasi yang tidak diperlukan, dan menghitung jumlah item dalam satu transaksi.

3) Menyaring Transaksi dengan Jumlah Item ≤ 1

Transaksi yang hanya berisi satu item tidak dapat digunakan dalam pembentukan kombinasi dua item (*2-itemsets*), sehingga transaksi tersebut dihapus. Penyaringan dilakukan dengan perintah:

```
df_multi_item = df[df['jumlah_item'] >  
1]
```

Dengan langkah ini, dataset akhir hanya berisi transaksi yang memiliki minimal dua produk, sehingga memungkinkan terbentuknya aturan asosiasi yang bermakna.

4) Menyimpan Dataset Hasil Filtering

Setelah proses filtering selesai, dataset hasil penyaringan disimpan ke dalam file Excel:

```
df_multi_item.to_excel('transaksi_buang  
itemset1.xlsx', index=False)
```

Penyimpanan ini bertujuan agar data hasil filtering dapat digunakan kembali tanpa perlu melakukan proses yang sama dari awal.

Berdasarkan hasil *filtering data*, jumlah transaksi yang awalnya 2432 berkurang menjadi 2403 transaksi setelah penghapusan data kosong dan transaksi dengan satu item. Pengurangan jumlah data ini wajar terjadi karena hanya

transaksi yang relevan untuk pembentukan aturan asosiasi yang dipertahankan.

5. Pembersihan Data (*Data Cleaning*)

Setelah proses penyaringan dilakukan, selanjutnya pembersihan data dilakukan untuk menghilangkan karakter-karakter yang tidak diperlukan pada kolom Produk, proses ini menggunakan fungsi `re.sub()` dari pustaka *regular expression* untuk menghapus karakter seperti tanda kutip ('), tanda kurung kurawal ({ }) dan tanda kurung biasa (()). Karakter-karakter tersebut berpotensi mengganggu pemrosesan data lebih lanjut. Berikut adalah baris kode yang melakukan *data cleaning*:

```
df['Produk'] = df['Produk'].apply(lambda
x: re.sub(r"[{}()]", '', x)).str.split
(',')
```

Selanjutnya, data yang semula berupa string item yang digabung dengan tanda koma diubah menjadi format list menggunakan metode `.str.split(',')`. Dengan demikian, setiap transaksi akan memiliki representasi berupa daftar item individual yang siap untuk diproses lebih lanjut.

4.2 Proses Transformasi Data

Setelah dilakukan proses penyaringan dan *data cleaning* terhadap data transaksi untuk hanya menyertakan transaksi yang memiliki lebih dari tiga produk, langkah selanjutnya adalah melakukan proses *transformation data*, yaitu mengubah data transaksi dari format teks menjadi struktur list agar lebih mudah diproses.

Data pada kolom Produk yang sebelumnya berupa string dipisahkan berdasarkan tanda koma menjadi list item. Kemudian, digunakan fungsi `explode()` untuk memecah list tersebut menjadi beberapa baris transaksi individual. Tahapan ini dilakukan dengan kode berikut:

```
df_transactions= df_multi_item.explode
('Produk').reset_index(drop=True)
df_transactions['Value'] = 1
```

Setelah data dipecah perproduk, selanjutnya dilakukan *aggregation data* berupa pembentukan *pivot table*, yaitu sebuah matriks biner dimana setiap baris mewakili transaksi dan setiap kolom mewakili produk. Berikut adalah baris koding dalam proses pembuatan *pivot table*:

```
basket = df_transactions.pivot_table(index='ID
Transaksi', columns='Produk', values='Value',
aggfunc='sum', fill_value=0)
basket = basket.applymap(lambda x: 1 if x >
```

Nilai 1 menunjukkan bahwa produk tersebut ada dalam transaksi tersebut, sedangkan 0 berarti produk tidak ada. Ringkasnya data transaksi yang semula berupa teks produk dipisah menjadi list item per transaksi, kemudian diubah menjadi format *pivot table* biner (0 dan 1). Dimana:

- Setiap baris mewakili satu transaksi,
- Setiap kolom mewakili satu produk,
- Nilai 1 menunjukkan bahwa produk tersebut ada dalam transaksi.

Data ini kemudian dikonversi ke dalam format Boolean (True/False) agar sesuai dengan input yang dibutuhkan oleh algoritma Apriori.

Data hasil *pre-processing* ini menjadi masukan utama dalam proses modelling untuk mencari *frequent itemsets* dan aturan asosiasi. Dengan pendekatan ini, kualitas aturan yang dihasilkan menjadi lebih optimal karena hanya berdasarkan pada transaksi yang benar-benar relevan dan memiliki potensi keterkaitan antar item.

4.3 Implementasi Model Algoritma Apriori

Pada tahap ini, dilakukan proses pembentukan model asosiasi dengan menggunakan algoritma Apriori. Tujuan dari proses ini adalah untuk menemukan pola asosiasi yang tersembunyi dari data transaksi

yang telah melalui tahap *pre-processing*. Dalam implementasi, digunakan fungsi `apriori()` dari Pustaka `mlxtend.frequent_patterns`, yang menerima parameter berupa data transaksi dalam bentuk pivot table (`df_pivot`) dan nilai ambang *minimum support*. Nilai *minimum support* ini berfungsi sebagai filter, yaitu hanya kombinasi item dengan tingkat kemunculan melebihi nilai tersebut yang akan dipertahankan. Contoh kode untuk proses ini adalah sebagai berikut:

```
frequent_itemsets = apriori(basket,  
min_support=support, use_colnames=True)
```

Output dari proses ini berupa daftar kombinasi item beserta nilai *support*nya, yang menunjukkan seberapa sering kombinasi tersebut muncul dalam seluruh transaksi. Proses ini juga merupakan bentuk dari *data aggregation*, yaitu pengelompokan nilai berdasarkan frekuensi kemunculan item dalam keseluruhan data.

4.3.1 Uji Sensitivitas Minimum Support dan Confidence

Untuk memperoleh pemahaman yang lebih mendalam terhadap pengaruh parameter, dilakukan pengujian (uji sensitivitas) dengan mencoba semua kombinasi nilai *minimum support* dan *minimum confidence* dalam rentang 0 hingga 1 dengan kenaikan 0.1 (0.1, 0.2, 0.3, ..., 1.0). Pada setiap kombinasi parameter *minimum support* dan *minimum confidence*, algoritma Apriori secara otomatis melakukan proses iteratif untuk membentuk aturan asosiasi.

Proses ini dimulai dari pembentukan frequent 1-itemset (kombinasi satu produk) yang memenuhi batas *minimum support*. Selanjutnya, algoritma membentuk frequent 2-itemset dari kombinasi 1-itemset yang lolos, lalu mengevaluasi kembali dukungan (*support*) masing-masing kombinasi. Itemset yang tidak memenuhi *minimum support* akan dihapus (*pruning*).

Hasil uji sensitivitas parameter *minimum support* dan *minimum confidence* ditampilkan secara lengkap pada **Lampiran I** mengingat ukuran tabel yang sangat besar. Pada Tabel 4.1 ini hanya menampilkan 7 kombinasi teratas dari hasil pengujian sebagai representasi, antara lain:

Tabel 4.1 7 Kombinasi teratas Uji Sensitivitas

min_support_param	min_confidence_param	jumlah_aturan
0.1	0.09	12
0.1	0.1	12
0.1	0.2	12
0.1	0.3	12
0.1	0.4	8
0.1	0.5	0
0.1	0.6	0

Pada Lampiran I yang memuat hasil lengkap uji sensitivitas parameter *minimum support* dan *minimum confidence*, dapat diperoleh beberapa temuan yaitu:

- Support rendah (0.1–0.2) : menghasilkan lebih banyak *frequent itemsets* dan aturan asosiasi yang relevan.
- Support menengah (0.3–0.4) : jumlah aturan berkurang, sebagian konfigurasi tidak menghasilkan aturan sama sekali.
- Support tinggi (≥ 0.5) : mayoritas konfigurasi tidak menghasilkan *frequent itemsets* atau aturan.
- Confidence rendah–sedang (0.5–0.6) : memunculkan aturan dengan variasi lebih banyak.
- Confidence tinggi (≥ 0.8) : aturan yang terbentuk sedikit, hanya hubungan paling kuat yang tersisa.

Pendekatan ini membantu melihat keseimbangan antara jumlah aturan yang dihasilkan dengan kekuatan asosiasi, sehingga parameter yang dipilih pada tahap akhir benar-benar optimal secara analisis bisnis. Meskipun pada hasil akhir untuk parameter *min support* = 0.1 dan *min confidence* = 0.4 hanya ditemukan sejumlah aturan (misalnya 12 aturan), hal tersebut merupakan hasil dari proses pencarian menyeluruh terhadap semua kemungkinan kombinasi produk yang ada dalam dataset, dan berhenti ketika tidak ada variasi baru yang memenuhi syarat.

4.3.2 Analisis Metrik Tambahan

Selain metrik utama *support* dan *confidence*, evaluasi kualitas aturan asosiasi dalam penelitian ini juga didukung oleh metrik tambahan seperti lift, conviction, dan leverage:

- Lift
 - Nilai lift seluruh aturan berada di atas 1, dengan kisaran sekitar 1,51–1,87.
 - $Lift > 1$ menandakan hubungan antara produk antecedent dan consequent bersifat positif (*positively correlated*), artinya pembelian produk A meningkatkan kemungkinan pembelian produk B dibandingkan pembelian acak.
 - Nilai tertinggi (1,87) ditemukan pada aturan $PP \rightarrow MP$ dan $MP \rightarrow PP$, yang menunjukkan hubungan kuat antara kedua produk ini.
- Leverage
 - Nilai leverage pada aturan yang terbentuk berkisar antara 0,0389–0,0493.
 - Leverage mengukur seberapa besar perbedaan probabilitas pembelian bersama dibandingkan pembelian acak. Semakin tinggi nilainya, semakin besar dampak keterkaitan antar produk.
 - Nilai leverage tertinggi (0,0493) juga terdapat pada aturan $PP \rightarrow MP$ dan $MP \rightarrow PP$, mengindikasikan adanya keterkaitan yang relatif kuat meskipun support tidak terlalu besar.
- Conviction
 - Nilai conviction pada semua aturan berada di atas 1, dengan kisaran sekitar 1,21–1,43.
 - $Conviction > 1$ menunjukkan bahwa kesalahan prediksi aturan (pelanggan membeli A tetapi tidak membeli B) lebih jarang terjadi daripada yang diharapkan secara acak.
 - Nilai conviction tertinggi (1,43) ditemukan pada aturan $MP \rightarrow PP$, yang berarti aturan ini memiliki

kestabilan yang baik untuk digunakan dalam prediksi pembelian.

Kesimpulannya berdasarkan ketiga metrik tambahan ini, aturan dengan kombinasi nilai lift, leverage, dan conviction tertinggi (terutama PP ↔ MP) dapat dianggap sebagai kandidat kuat untuk strategi *cross-selling*. Lift dan leverage memberikan gambaran kekuatan hubungan, sementara conviction menegaskan konsistensi aturan. Kombinasi ketiganya membantu memilih aturan yang tidak hanya signifikan secara statistik, tetapi juga relevan untuk implementasi bisnis.

Penggunaan metrik tambahan ini bertujuan memberikan gambaran yang lebih komprehensif mengenai kekuatan dan relevansi aturan yang terbentuk, sehingga hasil analisis dapat dipertanggungjawabkan baik secara statistik maupun dalam konteks bisnis.

4.4 Hasil Rule Asosiasi

Hasil dari proses pembentukan aturan asosiasi menghasilkan sekumpulan aturan yang memenuhi nilai *minimum confidence* tertentu. Setiap aturan terdiri dari bagian *antecedent* dan *consequent* yang menunjukkan hubungan keterkaitan antar item. Evaluasi kualitas setiap aturan dilakukan menggunakan metrik utama seperti *support*, *confidence*, dan *lift*, serta metrik tambahan seperti *leverage* dan *conviction* yang memberikan gambaran lebih rinci mengenai kekuatan dan stabilitas asosiasi.

Proses pembentukan aturan asosiasi ini menggunakan nilai *confidence* sebagai acuan utama dengan variasi *minimum confidence* tertentu untuk melihat pengaruhnya terhadap jumlah aturan yang terbentuk. Pada potongan kode di bawah, parameter (`min_threshold=min_confidence`) berfungsi untuk mengatur batas minimum nilai *confidence* yang digunakan dalam pemilihan aturan. Berikut adalah potongan kode yang digunakan untuk menghasilkan aturan asosiasi:

```
rules = association_rules(frequent_itemsets,
metric="confidence
",min_threshold=min_confidence)
```

Aturan-aturan asosiasi ini dibentuk menggunakan nilai *confidence* sebagai acuan utama. Dalam penelitian ini telah digunakan beberapa variasi *minimum confidence* untuk mengamati pengaruhnya terhadap pembentukan aturan asosiasi. Dari pengujian, diketahui bahwa aturan asosiasi hanya terbentuk pada kombinasi nilai *minimum support* sebesar 0.1 dengan *minimum confidence* antara 0.1 hingga 0.4. Kombinasi nilai *minimum support* yang lebih tinggi atau *minimum confidence* $\geq 0,5$ tidak menghasilkan aturan karena jumlah *frequent itemset* yang memenuhi kriteria terlalu sedikit.

Aturan yang terbentuk menunjukkan keterkaitan antar produk seperti BP (Beef Pizza), MP (Mini Pizza), PP (Pepperoni Pizza), dan TP (Tuna Pizza). Beberapa aturan dengan metrik terbaik antara lain:

1. Jika membeli MP, maka cenderung membeli PP (*confidence* 0,48; *lift* 1,87; *leverage* 0,0493; *conviction* 1,43).
2. Jika membeli PP, maka cenderung membeli MP (*confidence* 0,41; *lift* 1,87; *leverage* 0,0493; *conviction* 1,33).
3. Jika membeli BP, maka cenderung membeli TP (*confidence* 0,46; *lift* 1,57; *leverage* 0,0489; *conviction* 1,31).

Seluruh aturan memiliki nilai *lift* > 1 , menunjukkan adanya asosiasi positif antar produk. Nilai *conviction* berkisar antara 1,21 hingga 1,43, yang menandakan adanya kecenderungan hubungan, meskipun kekuatannya tergolong sedang. Nilai *leverage* berada di kisaran 0,038–0,049, yang berarti frekuensi kemunculan bersamaan sedikit lebih tinggi dibandingkan jika produk dipilih secara acak.

Berdasarkan pola ini, dapat disimpulkan bahwa strategi *cross-selling* sebaiknya difokuskan pada produk-produk yang sering muncul bersama pada nilai *support* rendah namun memiliki *lift* yang signifikan.

4.5 Analisis terhadap Hasil Rule

Berdasarkan hasil analisis, nilai minimum confidence sebesar 40% menghasilkan aturan asosiasi yang paling kuat dan andal, ditunjukkan oleh tingkat keyakinan (confidence) yang tinggi diantara yang lain. Namun, secara keseluruhan hasil evaluasi model menunjukkan bahwa aturan asosiasi yang bermakna hanya terbentuk saat menggunakan nilai *minimum support* dan *minimum confidence* yang rendah. Ketika dilakukan pengujian pada nilai *minimum support* diatas 0.1 atau *minimum confidence* diatas 0.4, jumlah aturan menurun drastis atau bahkan tidak terbentuk sama sekali.

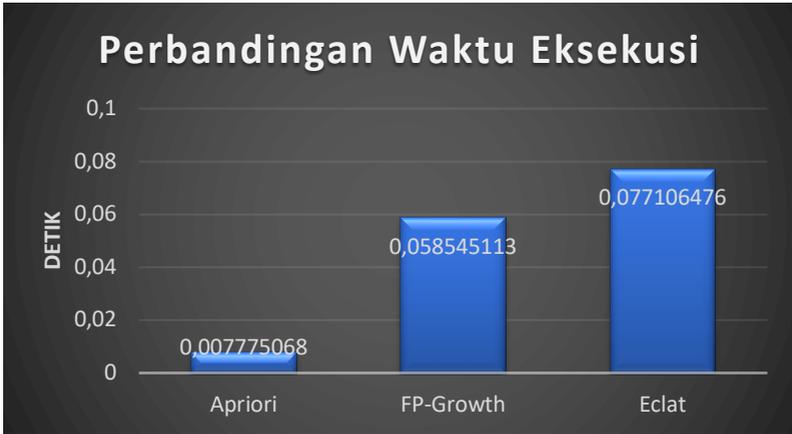
Hal ini mengindikasikan bahwa kualitas asosiasi antar produk dalam data transaksi masih lemah. Kondisi ini kemungkinan disebabkan oleh struktur data yang tidak terlalu besar, distribusi pembelian yang tidak merata, serta banyaknya transaksi yang hanya mengandung sedikit item. Oleh karena itu, model tidak mampu menghasilkan aturan yang kuat dan konsisten pada parameter yang ideal (*support* dan *confidence* yang tinggi).

Dengan demikian, dapat disimpulkan bahwa meskipun proses modelling berhasil membentuk beberapa aturan asosiasi, hasil yang diperoleh belum optimal dan model hanya efektif bekerja dalam parameter yang longgar.

4.6 Analisis Hasil Evaluasi Model

Setelah aturan asosiasi terbentuk menggunakan algoritma Apriori, langkah selanjutnya adalah melakukan evaluasi terhadap model yang dihasilkan. Evaluasi dilakukan dengan membandingkan aturan yang terbentuk dengan hasil yang diperoleh dari algoritma FP-Growth dan ECLAT pada parameter *minimum support* sebesar 0.1 dan *minimum confidence* sebesar 0.4. Perbandingan ini bertujuan untuk melihat perbedaan kinerja dan hasil antara ketiga algoritma asosiasi dalam menemukan pola pembelian yang konsisten. Perbandingan ini penting dilakukan karena ketiga algoritma sama-sama digunakan untuk analisis asosiasi, namun memiliki perbedaan mendasar dalam

metode pencarian *frequent itemset*. Dengan menggunakan dataset dan parameter identik, dapat dianalisis perbedaan waktu eksekusi, jumlah aturan yang dihasilkan, serta nilai metrik seperti *support*, *confidence*, dan *lift*.



Gambar 4.2 Perbandingan Waktu Apriori vs FP-Growth vs ECLAT



Gambar 4.3 Perbandingan Jumlah Aturan Apriori vs FP-Growth vs ECLAT

Berdasarkan hasil pengujian, dapat dilihat pada diagram bahwa algoritma Apriori memerlukan waktu eksekusi sebesar 0.007292 detik untuk menghasilkan 8 aturan asosiasi, sedangkan algoritma FP-Growth memerlukan waktu eksekusi sebesar 0.059910 detik untuk menghasilkan jumlah aturan asosiasi yang sama. Sementara itu, algoritma ECLAT menghasilkan aturan yang sama yaitu 8 aturan dan waktu eksekusi sebesar 0,077106476 detik, yang berada di atas Apriori dan FP-Growth.

Hasil ini menunjukkan bahwa pada dataset transaksi Dapur Tiara dengan ukuran dan kompleksitas seperti yang digunakan dalam penelitian ini, Apriori justru lebih cepat dibandingkan FP-Growth maupun ECLAT. Hal ini dimungkinkan karena jumlah transaksi dan variasi produk relatif sedikit sehingga proses pembangkitan kandidat kombinasi pada Apriori tidak memerlukan banyak iterasi. Sementara itu, FP-Growth dan Eclat memiliki overhead dalam membangun struktur data (FP-Tree untuk FP-Growth dan *tid-list intersection* untuk Eclat) yang pada dataset kecil justru membuat waktu proses lebih lama.

Dengan demikian, meskipun secara teori FP-Growth dan ECLAT lebih efisien pada dataset besar, pada kasus ini Apriori menjadi pilihan yang lebih cepat tanpa mengurangi kualitas aturan asosiasi yang dihasilkan.

4.7 Visualisasi Hasil

Setelah proses pembentukan model dan evaluasi selesai dilakukan, hasil akhir ditampilkan dalam bentuk visualisasi berbasis web untuk memudahkan pemahaman dan interpretasi terhadap pola-pola asosiasi yang ditemukan.

Visualisasi ini dibuat menggunakan *framework* Flask yang terintegrasi dengan *template engine* Jinja2. Melalui aplikasi web ini terdapat dua halaman, yaitu pada halaman awal sistem web, *user* diarahkan untuk meng-*upload* file dataset dalam format Excel (.xls atau .xlsx). Halaman kedua dari web yaitu menampilkan hasil pemrosesan data dalam bentuk tabel aturan asosiasi, metrik evaluasi model, dan interpretasi berbasis kalimat.

Beberapa fungsi penting yang diimplementasikan dalam sistem visualisasi ini antara lain:

```
@app.route('/', methods=['GET', 'POST'])
def upload_file():
```

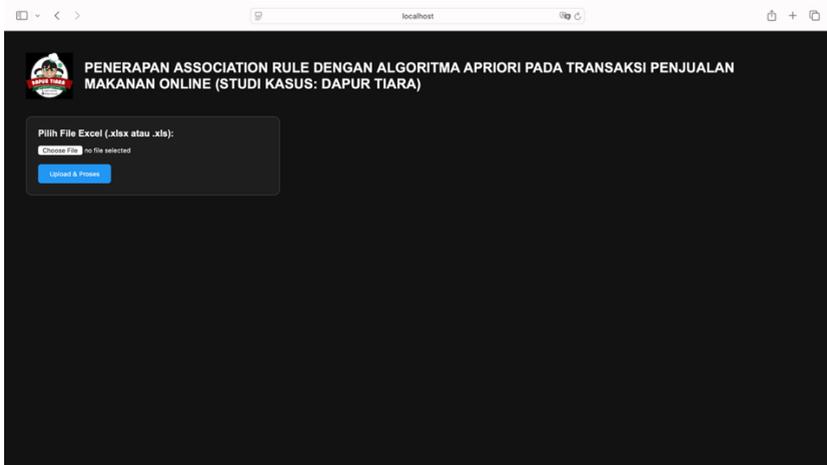
Fungsi ini bertugas untuk menangani permintaan *user* saat meng-*upload* file excel berisi data transaksi. Sistem akan memeriksa apakah file valid dan menyimpannya di folder uploads/. Selanjutnya ada fungsi validasi format file yang akan mengecek apakah file yang di-*upload* memiliki ekstensi .xls atau .xlsx. Berikut kodingnya:

```
def allowed_file(filename):
    return '.' in filename and
    filename.rsplit('.', 1)[1].lower() in
    ALLOWED_EXTENSIONS
```

Lalu, fungsi untuk mengirimkan hasil analisis ke halaman HTML (web) berupa tabel, kalimat deskriptif dari aturan, nilai evaluasi agar ditampilkan kepada *user* ialah sebagai berikut:

```
return render_template('results.html',
    results=results, filename=filename)
```

Fungsi `render_template` digunakan untuk *me-render* (menyusun dan menampilkan) file HTML yang berada dalam folder *templates*, dan menyisipkan data dari *backend* (Python) ke dalam halaman web tersebut secara dinamis. Berikut Gambar 4.2 dibawah ini merupakan tampilan awal halaman/*interface* web:



Gambar 4.4 Tampilan awal *Interface Web*

Setelah proses *upload* dilakukan, sistem secara otomatis akan memproses data tersebut dan menampilkan hasil analisis berupa tabel nilai *aturan asosiasi*, *minimum support*, *confidence*, dan *lift* dari berbagai parameter yang diuji. Selain itu, aturan-aturan asosiasi yang terbentuk juga disajikan secara terstruktur dalam bentuk kalimat. Gambar 4.3 menampilkan *interface web* hasil rancangan, yang memuat tabel evaluasi model dan aturan asosiasi.

File yang diunggah: DataDT.xlsx

Min Support: 0.1

Aturan	Support	Confidence	Lift
'MP' → 'BP'	0.102	0.454	1.589
'PP' → 'BP'	0.112	0.441	1.544
'BP' → 'TP'	0.138	0.483	1.597
'TP' → 'BP'	0.138	0.456	1.597
'PP' → 'MP'	0.107	0.42	1.872
'MP' → 'PP'	0.107	0.477	1.872
'MP' → 'TP'	0.11	0.491	1.622
'PP' → 'TP'	0.116	0.455	1.502

Jika membeli 'MP', maka cenderung membeli 'BP' (Confidence: 45.36%)
 Jika membeli 'PP', maka cenderung membeli 'BP' (Confidence: 44.06%)
 Jika membeli 'BP', maka cenderung membeli 'TP' (Confidence: 48.33%)
 Jika membeli 'TP', maka cenderung membeli 'BP' (Confidence: 45.58%)
 Jika membeli 'PP', maka cenderung membeli 'MP' (Confidence: 41.96%)
 Jika membeli 'MP', maka cenderung membeli 'PP' (Confidence: 47.75%)
 Jika membeli 'MP', maka cenderung membeli 'TP' (Confidence: 49.07%)
 Jika membeli 'PP', maka cenderung membeli 'TP' (Confidence: 45.45%)

Gambar 4.5 Interface Web Hasil Rancangan

Tampilan *interface* web dirancang sederhana dan informatif, dengan menampilkan tabel yang tersusun rapi menggunakan HTML dan CSS. Dengan adanya visualisasi ini, hasil analisis dapat diakses secara interaktif dan lebih mudah dipahami oleh *user*, terutama dalam mengevaluasi kualitas aturan yang terbentuk berdasarkan parameter yang digunakan.

4.8 Perhitungan Manual Apriori

Tujuan dari poin ini ialah untuk menunjukkan cara menghitung support, confidence, lift, leverage, dan conviction secara manual pada 10 transaksi pertama dengan *minimum support* 0.3 dan *minimum confidence* 0.4. Sekaligus menjelaskan proses iterasi Apriori dari L1, L2 sampai berhenti.

Langkah 1 – Ambil 10 transaksi teratas

Tabel 4.2 10 Transaksi Teratas

ID Transaksi	Produk
1	{PP,BP}
2	{PP,BP,CP}
3	{PP}
4	{'TP', 'NUB'}
5	{'BP', 'MP', 'NUB', 'TP', 'TNK'}
6	{'PP', 'CP', 'TP', 'BP'}
7	{'NUB', 'CP'}
8	{'BP', 'MP', 'TNK'}
9	{'BP', 'MP', 'CP', 'TP'}
10	{'MP', 'PP', 'NUB'}

Total transaksi $N = 10$

Langkah 2 – Frequent 1-itemset (hitungan support)

Menghitung berapa kali tiap item muncul di 10 transaksi dan Support

(i) – (jumlah transaksi yang memuat i) / 10:

- BP: 6 → 0.6
- PP: 5 → 0.5
- CP: 4 → 0.4
- TP: 4 → 0.4
- NUB: 4 → 0.4
- MP: 4 → 0.4
- TNK: 2 → 0.2

Semua diatas 0.1 artinya seluruh 1-itemset LOLOS.

Langkah 3 – Frequent 2-itemset (hitung support)

Support (A,B) = jumlah transaksi yang memuat A dan B) / 10

- (BP, MP) = 3 → 0,30
- (BP, TP) = 3 → 0,30
- (BP, CP) = 3 → 0,30
- (BP, PP) = 3 → 0,30

- (BP, NUB) = 1 → 0,10 ----- tidak lolos
- (BP, TNK) = 2 → 0,20 ----- tidak lolos
- (PP, CP) = 2 → 0,20 ----- tidak lolos
- (PP, TP) = 1 → 0,10 ----- tidak lolos
- (PP, MP) = 1 → 0,10 ----- tidak lolos
- (PP, NUB) = 1 → 0,10 ----- tidak lolos
- (PP, TNK) = 0 → 0 ----- tidak lolos
- (CP,TP) = 2 → 0,20 ----- tidak lolos
- (CP, MP) = 1 → 0,10 ----- tidak lolos
- (CP, NUB) = 1 → 0,10 ----- tidak lolos
- (CP, TNK) = 0 → 0 ----- tidak lolos
- (TP, MP) = 2 → 0,20 ----- tidak lolos
- (TP, NUB) = 2 → 0,20 ----- tidak lolos
- (TP, TNK) = 1 → 0,10 ----- tidak lolos
- (MP, NUB) = 2 → 0,20 ----- tidak lolos
- (MP, TNK) = 2 → 0,20 ----- tidak lolos
- (NUB, TNK) = 1 → 0,10 ----- tidak lolos

Hanya beberapa yang lolos karena memenuhi support dengan *minimum support* yaitu 0.3 yaitu (BP,MP), (BP,TP), (BP,CP), dan (BP,PP) Semua pasangan di atas adalah frequent 2-itemset.

Langkah 4 – Frequent 3-itemset (hitung support)

Selanjutnya dilakukan iterasi ketiga untuk menemukan frequent 3-itemset.

- (BP,MP,TP) = 2 → 0,20 ----- tidak lolos
- (BP,MP,CP) = 1 → 0,10 ----- tidak lolos
- (BP,MP,PP) = 0 → 0 ----- tidak lolos
- (BP,TP,CP) = 2 → 0,20 ----- tidak lolos
- (BP,TP,PP) = 1 → 0,10 ----- tidak lolos
- (BP,CP,PP) = 2 → 0,20 ----- tidak lolos

Tidak ada 3-itemset yang lolos karena semua dibawah 0,3. Itu berarti iterasi berhenti di 2-itemset.

Langkah 5 – Pembentukan Aturan Asosiasi dan Metrik

Rumus yang dipakai:

- $confidence(A \rightarrow B) = support(A,B) / support(A)$
- $lift(A \rightarrow B) = confidence / support(B)$
- $leverage(A \rightarrow B) = support(A,B) - support(A) \cdot support(B)$
- $conviction(A \rightarrow B) = (1 - support(B)) / (1 - confidence)$

Aturan yang memenuhi *min-support* 0,1 dan *min-confidence* 0,4 dapat dilihat pada Tabel 4.3 dibawah ini:

Tabel 4.3 Pembentukan Aturan Manual

A → B	Support	Confidence	Lift	Leverage	Conviction
BP → MP	0,3	0,5	1,25	0,06	1,2
MP → BP	0,3	0,75	1,25	0,06	1,6
BP → TP	0,3	0,5	1,25	0,06	1,2
TP → BP	0,3	0,75	1,25	0,06	1,6
BP → CP	0,3	0,5	1,25	0,06	1,2
CP → BP	0,3	0,75	1,25	0,06	1,6
BP → PP	0,3	0,5	1	0	1
PP → BP	0,3	0,6	1	0	1

Langkah 6 – Ringkasan

Dengan *minimum support* 0.3 dan *minimum confidence* 0.4, model Apriori pada 10 transaksi ini menghasilkan aturan yang paling relevan:

- MP→BP, TP→BP, CP→BP (conf 0,75; lift 1,25; lev 0,06; dan conv 1,6)
- Aturan BP ↔ PP punya lift = 1, ini hanya kebetulan tidak ada hubungan lebih kuat dari random.

4.9 Interpretasi Aturan Asosiasi

Dari hasil pemodelan dengan algoritma Apriori, ditemukan beberapa aturan asosiasi yang memiliki nilai *confidence* dan *lift* yang cukup tinggi, khususnya pada *minimum support* 0.1 dan *minimum confidence* 0.4. salah satu aturan yang paling menonjol adalah:

Jika membeli MP, maka cenderung membeli PP (*confidence* 0,48; *lift* 1,87; *leverage* 0,0493; *conviction* 1,43).

Aturan ini menunjukkan adanya kecenderungan yang kuat bahwa kedua produk tersebut sering dibeli bersamaan dalam satu transaksi. Nilai lift diatas 1 memperkuat bahwa asosiasi tersebut bukan terjadi secara kebetulan, melainkan lebih sering daripada yang diharapkan secara acak.

Selain itu, terdapat juga asosiasi yang lain seperti:

- Jika membeli 'MP', maka cenderung membeli 'BP' (Confidence: 45.36%)
- Jika membeli 'PP', maka cenderung membeli 'BP' (Confidence: 44.06%)
- Jika membeli 'BP', maka cenderung membeli 'TP' (Confidence: 48.33%)
- Jika membeli 'TP', maka cenderung membeli 'BP' (Confidence: 45.58%)
- Jika membeli 'PP', maka cenderung membeli 'MP' (Confidence: 41.96%)
- Jika membeli 'MP', maka cenderung membeli 'PP' (Confidence: 47.75%)
- Jika membeli 'MP', maka cenderung membeli 'TP' (Confidence: 49.07%)
- Jika membeli 'PP', maka cenderung membeli 'TP' (Confidence: 45.45%)

Aturan-aturan ini mencerminkan kebiasaan pelanggan dalam membeli produk secara bersamaan, yang dapat dimanfaatkan lebih lanjut untuk strategi penjualan.

Temuan dari aturan asosiasi ini memberikan beberapa implikasi strategis bagi pemilik bisnis, dalam hal ini Dapur Tiara:

1. Promosi *Bundling*

Aturan seperti 'MP' → 'BP' atau 'PP' → 'BP' dapat digunakan untuk membuat paket *bundling* atau diskon gabungan. Misalnya, jika membeli 'Mini Pizza', maka bisa diberikan diskon harga untuk 'Beef Pizza'.

2. Peningkatan *Cross-Selling*

Dengan mengetahui produk-produk yang sering dibeli secara bersamaan, sistem penjualan online dapat menerapkan strategi

cross-selling dengan memberikan rekomendasi otomatis kepada pelanggan. Strategi ini memungkinkan peningkatan penjualan tanpa harus menambah biaya akuisisi pelanggan baru, karena pelanggan yang sudah ada diarahkan untuk membeli lebih banyak item dalam satu transaksi. Oleh karena itu, meskipun aturan yang dihasilkan pada nilai *confidence* 40% memiliki tingkat kepastian yang sedikit lebih rendah, aturan tersebut tetap memberikan nilai tambah yang signifikan dalam mengidentifikasi peluang penjualan silang (*cross-selling*).

3. Manajemen Stok yang Lebih Tepat

Mengetahui produk mana yang sering dibeli bersamaan memungkinkan manajemen Gudang untuk memprediksi stok secara lebih akurat.

Sebagai bentuk validasi terhadap hasil analisis asosiasi yang telah diperoleh, peneliti melakukan wawancara langsung dengan pelaku usaha Dapur Tiara. Wawancara ini bertujuan untuk mengetahui apakah *bundling* produk yang dihasilkan dari analisis algoritma Apriori relevan dan dapat diterapkan dalam strategi penjualan nyata. Berikut merupakan tabel hasil uji kualitatif terhadap pelaku usaha melalui wawancara atau observasi:

Tabel 4.4 Uji Kualitatif terhadap Pelaku Usaha

No	Bundling Produk	Disetujui oleh Pelaku Usaha	Alasan Disetujui/Tidak
1	TP + BP	Iya	Sudah dilakukan
2	SCDC+SCM+SCW+SCMt+SCRV	Iya	Cocok untuk paket hemat
3	BP + L	Iya	Akan dilakukan bundling
4	PP + SCDC	Tidak	Kurangnya minat

Dari wawancara tersebut, diperoleh tanggapan sebagai berikut:

1. Bundling TP (Tuna Pizza) + BP (Beef Pizza) telah disetujui karena sudah pernah diterapkan sebelumnya dan terbukti efektif dalam meningkatkan penjualan.
2. Kombinasi bundling produk SCDC + SCM + SCW + SCMt + SCRv, yang merupakan varian cookies, disetujui karena dinilai cocok untuk dijadikan paket hemat.
3. Kombinasi BP (Beef Pizza) + L (Lasagna) juga mendapat persetujuan dan direncanakan akan diterapkan sebagai bundling dalam waktu dekat.
4. Sebaliknya, bundling PP (Pepperoni Pizza) + SCDC (*Soft Cookies Dark Chocolate*) tidak disetujui karena dinilai kurang diminati oleh konsumen.

Hasil wawancara ini memperkuat bahwa beberapa aturan asosiasi yang terbentuk dari analisis data transaksi tidak hanya valid secara statistik, namun juga dapat diterapkan secara praktis dalam strategi bisnis. Validasi ini menjadi bagian penting dari implementasi data mining untuk mendukung pengambilan keputusan berbasis data.

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan hal-hal sebagai berikut:

1. Algoritma Apriori terbukti mampu mengidentifikasi pola pembelian produk makanan yang sering muncul secara bersamaan dalam data transaksi penjualan. Beberapa kombinasi produk yang ditemukan menunjukkan nilai *support*, *confidence*, dan *lift* yang tinggi, yang mengindikasikan adanya hubungan kuat antar produk tertentu. Aturan asosiasi yang dihasilkan memberikan dasar yang valid untuk pengembangan sistem rekomendasi bundling produk.
2. Dari data transaksi penjualan Dapur Tiara, ditemukan pola kebiasaan belanja pelanggan. Beberapa produk sering dibeli bersamaan, misalnya orang yang membeli *Mini Pizza (MP)* cukup sering juga membeli *Beef Pizza (BP)*, *Pepperoni Pizza (PP)*, atau *Tuna Pizza (TP)*. Begitu juga sebaliknya, pembelian PP, BP, atau TP sering diikuti dengan pembelian produk lain yang sudah disebutkan.
3. Sistem rekomendasi berbasis Apriori ini memiliki potensi untuk diintegrasikan ke dalam strategi penjualan Dapur Tiara, dengan manfaat sebagai berikut:
 - Meningkatkan nilai transaksi rata-rata per pelanggan melalui teknik *cross-selling* atau paket *bundling*.
 - Pola pembelian ini bisa dipakai untuk membuat penawaran paket atau *bundling*, misalnya menggabungkan dua atau tiga produk yang sering dibeli bersama, sehingga pembeli merasa lebih hemat dan penjualan bisa meningkat.

- Hasil wawancara ini memperkuat bahwa beberapa aturan asosiasi yang terbentuk dari analisis data transaksi tidak hanya valid secara statistik, namun juga dapat diterapkan secara praktis dalam strategi bisnis. Validasi ini menjadi bagian penting dari implementasi data mining untuk mendukung pengambilan keputusan berbasis data.
4. Visualisasi berbasis web yang dikembangkan menggunakan *framework* Flask memungkinkan *user* untuk meng-*upload* dataset dan langsung melihat hasil aturan asosiasi secara interaktif dan informatif.
 5. Tiga metode pencarian pola Apriori, FP-Growth, dan ECLAT telah diuji. Pada data Dapur Tiara yang tidak terlalu besar, metode Apriori justru bekerja paling cepat. FP-Growth dan ECLAT sebenarnya dirancang untuk data yang sangat besar, sehingga pada kasus ini tidak memberi keuntungan waktu yang signifikan.

5.2 Saran

Berdasarkan proses dan hasil penelitian, beberapa saran yang dapat diberikan adalah sebagai berikut:

1. Perluasan dataset transaksi, baik dari segi jumlah maupun periode waktu, agar model dapat menemukan pola yang lebih kuat dan general.
2. Menambahkan atribut lain dalam analisis, seperti waktu transaksi, kategori produk, atau data pelanggan, untuk menghasilkan pola asosiasi yang lebih kompleks dan relevan. Dengan begitu, pola pembelian yang ditemukan bisa lebih spesifik, misalnya paket promo berdasarkan musim, hari tertentu, atau segmen pelanggan tertentu.
3. Meskipun pada penelitian ini algoritma Apriori terbukti paling cepat untuk dataset Dapur Tiara, tetap disarankan mencoba metode lain seperti FP-Growth dan ECLAT pada dataset yang lebih besar untuk membandingkan kecepatan dan efisiensi masing-masing metode.

4. Terapkan hasil pola pembelian ini ke promo nyata, lalu lihat apakah penjualannya meningkat. Misalnya, coba buat paket hemat berdasarkan produk yang sering dibeli bareng.

DAFTAR PUSTAKA

- Adri Ahmad, Rumlaklak Nelci Dessy, and Sina Derwin R. 2021. "IMPLEMENTASI ALGORITMA APRIORI UNTUK ANALISA DATA PENJUALAN (STUDI KASUS: TOKO UD. SURYANI)." *ICON* 9 No 2:182–88.
- Alda Shabriani. 2023. "Apa Itu Online Shop? Pengertian Dan Ide Jualan ."
- Ashari, Imam Ahmad, Anggit Wirasto, Deny Nugroho Triwibowo, and Purwono Purwono. 2022. "Implementasi Market Basket Analysis Dengan Algoritma Apriori Untuk Analisis Pendapatan Usaha Retail." *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer* 21(3):701–9. doi:10.30812/matrik.v21i3.1439.
- Delima Sitanggang, Penulis. 2023. *ALGORITMA APRIORI*.
- Erfina, Adhithia, Melawati Melawati, and Nunik Destria Arianti. 2020. "Penerapan Metode Data Mining Terhadap Data Transaksi Penjualan Menggunakan Algoritma Apriori (Studi Kasus : Toko Fasentro Fancy)." *SANTIKA Is a Scientific Journal of Science and Technology* 10(1):11–17. doi:10.37150/jsa.v10i1.1359.
- Gede, Aditra, S. Pradnyana, M. Kom, Ketut Kom, S. Agustini, and M. Si Si. n.d. *Konsep Dasar Data Mining*.
- HAIDAR, I. 2021a. "Implementasi Algoritma Apriori Untuk Mencari Pola Transaksi Penjualan (Studi Kasus: Carroll Kitchen)."

- Haidar, I. 2021b. "Implementasi Algoritma Apriori Untuk Mencari Pola Transaksi Penjualan (Studi Kasus: Carroll Kitchen)." [https://dspace.uui.ac.id/handle/123456789/36068%0Ahttps://dspace.uui.ac.id/bitstream/handle/123456789/36068/17523130Ibnu Haidar.pdf?sequence=1](https://dspace.uui.ac.id/handle/123456789/36068%0Ahttps://dspace.uui.ac.id/bitstream/handle/123456789/36068/17523130Ibnu%20Haidar.pdf?sequence=1).
- Mufid, Ahmad. 2023. "Flask: Pengertian, Fungsi, Kelebihan, Dan Cara Setup." <https://blog.rumahweb.com/flask-adalah/>.
- Nabeel, Muhammad, Shumaila Majeed, Mazhar Javed Awan, Hooria Muslih-Ud-din, Mashal Wasique, and Rabia Nasir. 2021. "Review on Effective Disease Prediction through Data Mining Techniques." *International Journal on Electrical Engineering and Informatics* 13(3):717–33.
- Radliya Nizar Rabbi. 2015. *Data Mining*.
- Saefudin, Saefudin, and Septian DN. 2019. "Penerapan Data Mining Dengan Metode Algoritma Apriori Untuk Menentukan Pola Pembelian Ikan." *JSiI (Jurnal Sistem Informasi)* 6(2):36. doi:10.30656/jsii.v6i2.1587.
- Tida, A. M., and A. Zubair. 2022. "Analisa Transaksi Penjualan Menggunakan Metode Algoritma Apriori (Studi Kasus Toko Busana Sakato, Gajah Mada Plaza B-19" *Jurnal Teknik Informatika Kaputama* ... 6(1). <http://jurnal.kaputama.ac.id/index.php/JTIK/article/viewFile/753/530>.
- Xandra Albert Sim, Alex, Fandi Halim, M. Hawari Simanullang, and M. Firkhan Siregar. 2015. *Pengembangan Aplikasi Market*

*Basket Analysis Menggunakan Algoritma Generalized
Sequential Pattern Pada Supermarket.*