

BAB II

LANDASAN TEORI

2.1. Penelitian Terdahulu

Pada penelitian yang dilakukan oleh Ramadani dan Alexander (2020) mengkaji tentang penerapan *data mining* dengan menggunakan algoritma *apriori* untuk meningkatkan pola penjualan obat. Studi ini dilakukan di Apotik Pusaka Arta dengan tujuan mengatasi masalah kekosongan stok obat yang sering terjadi akibat tidak memanfaatkan data transaksi secara efektif. Pendekatan yang digunakan dalam penelitian ini adalah *association rules*, dengan metode pengembangan yang mengacu pada kerangka kerja *CRISP-DM* (*Cross Industry Standard Process for Data mining*). Dengan mengatur nilai *minimum support* sebesar 20% dan *minimum confidence* sebesar 50% dari total 117 data yang tersedia, penelitian ini menghasilkan informasi mengenai jenis obat yang sering dibeli oleh konsumen secara bersamaan [6]. Penelitian ini memiliki kesamaan dengan penelitian yang mereka lakukan, yaitu penggunaan algoritma *apriori*, metode *association rules*, dan metode pengembangan *CRISP-DM*. Namun, Penelitian ini dilakukan berbeda dari penelitian mereka dalam hal ukuran *dataset* yang digunakan. Penelitian ini menggunakan *dataset* yang terdiri dari 2733 transaksi penjualan, sedangkan penelitian mereka hanya menggunakan 117 data transaksi. Perbedaan ini menunjukkan bahwa penelitian yang dilakukan ini memiliki *volume* data yang lebih besar, yang dapat memberikan pemahaman yang lebih *komprehensif* tentang pola pembelian konsumen dan potensi pengembangan strategi promosi. Dengan jumlah transaksi penjualan yang lebih banyak, penelitian ini memiliki kemampuan

yang lebih besar untuk mengidentifikasi pola-pola yang signifikan dan mencerminkan preferensi konsumen secara lebih akurat. Hasil penelitian ini diharapkan dapat memberikan wawasan yang lebih mendalam dan solusi yang lebih dalam pengembangan strategi promosi melalui paket *bundling* produk alat musik.

Penelitian selanjutnya dilakukan oleh Gaerry, membahas mengenai aplikasi *data mining* berbasis *web* dengan menggunakan algoritma *apriori* untuk analisa pola pembelian produk pada PT Menara Bahagia Bersama, permasalahan yang terjadi ialah proses menganalisa pola pembelian konsumen yang dilakukan secara *manual* sehingga menyebabkan waktu dan tenaga lebih besar dikorbankan. Tujuan dalam penelitian yang dilakukan untuk mengetahui pola pembelian konsumen dengan algoritma *apriori* berbasis *web* yang nantinya akan dijadikan strategi promosi pada penjualan dimasa yang akan datang. Metode yang digunakan ialah *association rules* dan metode pengembangan *CRISP-DM*. Dengan memasukkan nilai *minimum support* sebesar 1% dan nilai *minimum confidence* sebesar 50% dari 816 data transaksi dengan jumlah produk sebanyak 140 periode 1 januari 2021 sampai 31 januari 2021. Hasil penelitian berupa informasi yang dapat dipergunakan untuk kebutuhan promosi penjualan pada PT Menara Bahagia Bersama [7]. Pada penelitian yang dilakukan oleh Gaerry mempunyai persamaan dengan penelitian ini yaitu penggunaan algoritma *apriori*, metode *association rules*, metode pengembangan *CRISP-DM* dan hasil dari implementasi dijadikan strategi promosi pada penjualan di masa yang akan datang. Perbedaannya pada penelitian yang dilakukan oleh Gaerry pada pemodelan menggunakan alat bantu aplikasi *RapidMiner Studio 9.9* dan *deployment* berupa *website* dengan bahasa *PHP*,

sedangkan pada penelitian ini pemodelan data menggunakan alat bantu aplikasi *jupyter notebook* dan *deployment* berupa *website* dengan bahasa *python* menggunakan *framework flask*.

2.2. Teori Pendukung

Teori pendukung merupakan istilah dalam penelitian yang dapat digunakan sebagai acuan untuk memahami penelitian yang dilakukan oleh penulis.

2.2.1. Analisa

Menurut Azwar, Analisa merupakan suatu upaya untuk memecahkan sesuatu unit menjadi beberapa unit terkecil [8]. Sedangkan menurut Minto Rahayu, Analisa adalah suatu cara membagi subjek menjadi komponen yang padu [9]. Dari definisi di atas diambil kesimpulan bahwa Analisa merupakan proses memecahkan, mengidentifikasi dan memahami suatu bagian/masalah menjadi bagian-bagian terkecil yang saling berkaitan.

2.2.2. Pola

Pola adalah suatu tampilan atau pola yang muncul dari suatu rangkaian atau sekumpulan data atau elemen, yang memiliki karakteristik atau sifat tertentu yang dapat dikenali atau dibedakan. Pola seringkali diidentifikasi berdasarkan pengamatan atau analisis terhadap data atau informasi yang tersedia. Menurut Kurniasari, pola adalah suatu bentuk model, sistem ataupun cara kerjanya. Pola dikatakan sebagai model yang berarti cara untuk menunjukkan suatu proses yang mengandung hubungan [10].

2.2.3. Pembelian

Pembelian adalah proses memperoleh suatu produk atau jasa dengan

menukarkan uang atau nilai tukar lainnya. Pembelian dapat terjadi dalam berbagai skala, mulai dari pembelian kecil yang dilakukan oleh individu, hingga pembelian besar yang dilakukan oleh perusahaan atau organisasi. Menurut Soemarso, pembelian adalah akun yang digunakan untuk mencatat semua pembelian produk dagang dalam satu periode [11].

Dari beberapa pengertian di atas mengenai pembelian, maka disimpulkan bahwa pembelian adalah proses memperoleh suatu produk dengan menukarkan uang lalu pembelian ini dicatat dalam satu periode tertentu.

2.2.4. Produk *Bundling*

Menurut Arsyad yang dikutip oleh Moeniri, *product bundle* adalah cara menjual satu atau lebih produk secara bersamaan dalam satu paket dengan satu harga [12]. Menurut Yang, T.C., Lai, H. yang dikutip oleh Jalari dkk, bahwa produk pendamping ditujukan untuk memaksimalkan keuntungan dalam jenis pemasaran yang sering dilakukan dalam bidang produk berupa penelitian produk dan harga [11].

Menurut Royan, mengemukakan bahwa ada dua hal penting dalam produk *bundling* menurut Stremersch dan Tellis dalam penelitiannya “*Strategic and Product Bundle Pricing*” [13], yaitu :

1. Price Bundling

Price Bundling atau harga bundel adalah praktik penjualan di mana dua atau lebih produk yang berbeda dijual dalam satu paket dengan diskon atau penawaran khusus. Dalam penawaran ini, harga bundel untuk beberapa produk tersebut lebih rendah daripada jika produk tersebut dibeli secara

terpisah tanpa adanya integrasi. Penting untuk dicatat bahwa produk dalam harga bundel tidak terintegrasi, yang berarti konsumen masih dapat menggunakan setiap produk secara mandiri tanpa mengurangi fungsionalitasnya [12].

2. *Product Bundling*

Product bundling adalah praktik mengintegrasikan dan menjual dua atau lebih produk terpisah dengan harga tertentu. Integrasi ini ke dalam paket produk seringkali memberikan nilai tambah bagi konsumen. Dalam konteks ini, harga reservasi mengacu pada harga *maksimum* yang siap dibayar oleh konsumen. Saat produk dikombinasikan dalam sebuah bundel, harga reservasi untuk paket produk tersebut umumnya lebih tinggi daripada total harga reservasi yang diberlakukan jika setiap produk dibeli secara terpisah. Penentuan harga reservasi berkaitan dengan preferensi pembayaran yang diinginkan oleh masing-masing konsumen dan nilai persepsi yang diterima oleh mereka. Oleh karena itu, dalam melakukan *bundling*, perusahaan harus memperhatikan segmen pasar yang menjadi targetnya. Komponen produk yang digabungkan dalam bundel tersebut harus memiliki tujuan segmentasi yang serupa. Selain itu, harga *pre-order* dapat bervariasi antara segmen pasar, sehingga menciptakan nilai persepsi yang berbeda bagi setiap bundel produk [12].

2.2.5. Jenis-Jenis Produk *Bundling*

Menurut Stremersch dan Tellis, *Bundling* dapat dilakukan dalam tiga bentuk yang berbeda, yaitu *Pure Bundling*, *Mix Bundling*, dan *Unbundling* [12]. Penjelasannya sebagai berikut :

1. *Pure Bundling*

Pure bundling merupakan strategi di mana perusahaan hanya menjual produk dalam bentuk *bundling* dan tidak menjual produk secara terpisah.

2. *Mixed Bundling*

Mixed bundling adalah strategi di mana perusahaan menjual produk-produk yang tersedia secara terpisah, tetapi juga menawarkan pilihan untuk membeli semua produk tersebut dalam satu paket terpisah.

3. *Unbundling*

Unbundling adalah strategi di mana perusahaan hanya menjual produk secara terpisah dan tidak menawarkan produk-produk tersebut dalam bentuk *bundling*.

2.2.6. Promosi

2.2.6.1. Pengertian Promosi

Menurut Laksana, Promosi merupakan komunikasi antara penjual dan pembeli yang berasal dari informasi yang tepat untuk pembeli, bertujuan untuk memberikan pengaruh kepada sikap dan tingkah laku yang dilakukan pembeli atau konsumen, yang pada akhirnya diharapkan pembeli menjadi tahu produk apa yang di beli dan tetap mengingat produk tersebut [14].

Sedangkan menurut Alma dalam Heni, Promosi adalah bentuk komunikasi pemasaran berupa kegiatan pemasaran yang berusaha untuk memberikan informasi, memberi bimbingan dan memberikan pengaruh terhadap pembeli atas perusahaan dan produknya agar dapat membeli, menerima dan loyal pada produk yang ditawarkan oleh penjual [13].

Dari definisi di atas maka dapat ditarik kesimpulan bahwa promosi adalah suatu komunikasi antara penjual dan pembeli yang bertujuan untuk memberikan pengaruh, serta bimbingan dari penjual agar produk yang ditawarkan oleh perusahaan dibeli oleh konsumen (pembeli).

2.2.6.2. Cara Promosi Penjualan Paket *Bundling*

Promosi Penjualan dapat dilakukan dengan cara sebagai berikut :

1. Menawarkan produk atau jasa yang terkait

Misalnya, jika menjual produk busana muslim, maka dapat membuat paket *bundling* yang terdiri dari beberapa produk busana muslim yang terkait, seperti jilbab baju gamis, dan ciput (bagian dalam jilbab). Hal ini dapat memudahkan dalam memilih produk, meningkatkan penjualan dan memberikan pengalaman berbelanja konsumen.

2. Menawarkan harga diskon produk terkait

Menawarkan harga diskon untuk paket *bundling* yang dilakukan oleh perusahaan dapat menjadi daya tarik bagi konsumen. Dengan menawarkan harga diskon, tentunya konsumen akan memberikan rasa nilai lebih dengan membeli paket *bundling* daripada membeli produk secara terpisah.

3. Menawarkan produk eksklusif atau bonus hadiah tambahan

Memberikan tawaran produk eksklusif atau *bonus* hadiah dalam paket *bundling* dapat meningkatkan minat konsumen untuk membeli produk. Misalnya, perusahaan dapat menambahkan produk baru, produk terlaris dalam paket *bundling* atau memberikan hadiah seperti *voucher* belanja dan produk lainnya.

4. Menawarkan paket *bundling* terbatas.

Menawarkan paket *bundling* dengan jumlah terbatas dapat menciptakan rasa kewaspadaan bagi konsumen. Hal ini dapat mendorong konsumen untuk segera membeli paket *bundling* sebelum stok habis atau penawaran berakhir.

5. Menawarkan paket *bundling* musiman.

Menawarkan paket *bundling* yang berkaitan dengan acara atau musim tertentu, seperti saat menjelang hari natal, *valentine* ataupun hari besar lainnya. Dapat meningkatkan minat konsumen untuk membeli produk. Misalnya, dengan menawarkan paket *bundling* khusus imlek yang terdiri dari produk-produk yang sesuai dengan tema imlek.

2.2.7. Data mining

Kemunculan *data mining* didasarkan pada dampak dari pertumbuhan jumlah data yang disimpan dalam basis data yang semakin meningkat.

Data mining, menurut David Hand dkk. dari MIT, merujuk pada analisis data yang umumnya berukuran besar dengan tujuan untuk menemukan hubungan yang signifikan dan mengungkapkan informasi baru yang

sebelumnya tidak diketahui. Pendekatan yang digunakan dalam *data mining* ini menggunakan metode terkini yang dapat dipahami dan memberikan manfaat bagi pemilik data tersebut [15].

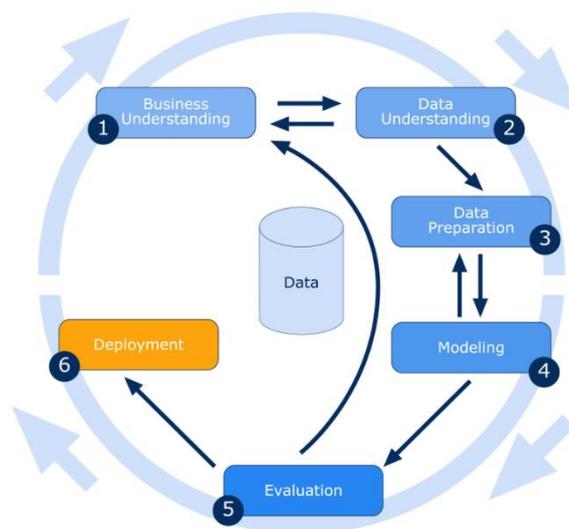
Data mining menurut Max Bramer dalam Yuli Mardi, *data mining* adalah serangkaian proses yang digunakan untuk menggali nilai tambah dari kumpulan data dengan cara otomatis, yang sebelumnya tidak diketahui melalui proses *manual* [16]

Dari pengertian yang telah dijelaskan, hal yang menjadi fokus utama terkait dengan *data mining* yaitu :

1. *Data mining* melibatkan tahapan proses otomatis terhadap data yang telah ada.
2. Proses ini dilakukan terhadap data yang memiliki volume yang sangat besar.
3. Tujuan utama dari *data mining* adalah untuk mengidentifikasi hubungan atau pola yang dapat memberikan informasi yang bernilai dan bermanfaat.

2.2.8. Cross-Industry Standard Process for Data mining

Proses *data mining* yang dikembangkan tahun 1996 oleh analisis dari beberapa industri seperti SPSS, NCR dan Daimler Chrysler. Menurut Hasanah, dkk, CRISP-DM adalah model proses pengembangan data yang sering digunakan para ahli untuk memecahkan suatu masalah [17]. Pada CRISP-DM siklus proses terbagi menjadi enam fase, fase ini dapat dilihat pada gambar 2.1 :



Gambar 2.1 Siklus Proses CRISP-DM
(Sumber: linkedin.com) [29]

Berikut penjelasan pada gambar diatas mengenai siklus proses *CRISP-DM*

[30] :

1. *Business Understanding* (Pemahaman Bisnis)

Dalam fase ini, diperlukan pemahaman mengenai esensi dari penelitian *data mining* yang akan dilakukan. Pemahaman ini harus berlandaskan pada perspektif bisnis yang relevan. Beberapa aspek yang harus dipertimbangkan antara lain memahami permasalahan dalam konteks bisnis, menetapkan sasaran atau tujuan bisnis, menentukan tujuan penelitian yang terkait dengan *data mining*, serta merencanakan strategi dan jadwal penelitian yang akan dilakukan.

2. *Data Understanding* (Pemahaman data)

Dalam fase ini peneliti Pada tahap ini, peneliti mengumpulkan data awal, mempelajari, dan memahami data yang akan digunakan. Dalam proses ini, data dianalisis untuk mendeteksi *subset* yang menarik sehingga dapat dibuat hipotesis awal, serta mengidentifikasi masalah yang berkaitan

dengan kualitas data.

3. *Data Preparation* (Persiapan data)

Pada fase ini dilakukan aktivitas pada data mentah yang didapatkan, mencakup *data cleaning*, melakukan *data selection record* dan atribut-atribut serta melakukan *data transformation* untuk dijadikan masukan dalam fase *modelling*.

4. *Modelling* (Pemodelan)

Fase selanjutnya dilakukan *machine learning* dan metode statistika untuk penentuan terhadap teknik *data mining*, algoritma *data mining* dan alat bantu *data mining* yang akan dipakai dalam penelitian. Selanjutnya melakukan penerapan algoritma *data mining* dan penerapan teknik yang digunakan kepada data dengan bantuan alat bantu.

5. *Evaluation* (Pengujian)

Pada fase ini, dilakukan interpretasi terhadap hasil dari *data mining* yang telah dilakukan pada tahap pemodelan sebelumnya. Evaluasi dilakukan terhadap model yang telah diterapkan sebelumnya dengan tujuan untuk mengkaji kesesuaian dengan tujuan yang ingin dicapai dalam tahap awal.

6. *Deployment* (Penyebaran)

Sebelum melakukan *deployment*, persiapan yang cermat diperlukan, salah satunya adalah menyusun rencana *deployment* yang terstruktur. *Deployment* dapat berupa pembuatan laporan sederhana atau pelaksanaan proses *data mining* berulang untuk keperluan perusahaan.

2.2.9. Algoritma *Apriori*

Menurut pendapat Benni R Siburian, algoritma *apriori* adalah salah satu algoritma yang digunakan untuk menemukan *itemset* yang sering muncul dengan menggunakan teknik *association rules* [18].

Dengan menerapkan teknik *association rules*, algoritma *apriori* digunakan dalam rangka menemukan aturan asosiasi yang memenuhi batasan nilai *support* dan *confidence*. Algoritma *apriori* bekerja dengan melakukan analisis terhadap sejumlah *item* yang diambil secara bersama-sama dari sejumlah transaksi yang dipilih. Pada setiap iterasi, algoritma ini melakukan perhitungan terhadap semua kumpulan data yang mengandung elemen tertentu yang selalu muncul. Proses iterasi tersebut melibatkan dua langkah utama, yakni pembentukan calon kandidat (*candidate generation*) dan pemilihan serta penghitungan kandidat (*candidate counting and selection*).

Dalam tahap perhitungan, algoritma *apriori* menghitung nilai *support* untuk setiap elemen dalam *database*. Hanya kumpulan data yang memenuhi batasan *minimal* yang ditentukan yang diperhitungkan sebagai kandidat produk yang sering muncul. Dalam pencarian pola kombinasi produk menggunakan algoritma *apriori*, tidak mungkin langsung menggabungkan suatu elemen dengan seluruh jenis produk yang ada. Pendekatan ini dilakukan untuk menghindari peningkatan waktu komputasi yang signifikan dalam mencari kombinasi produk. Sebagai gantinya, algoritma *apriori* melakukan *self-joining* berdasarkan hasil iterasi atau perulangan sebelumnya [19].

Berikut adalah tahapan utama yang dilakukan dalam algoritma *Apriori* untuk menemukan *frequent itemset* :

1. Penggabungan (*Join*)

Proses yang dilakukan adalah dengan menggabungkan *item* dengan *item* lainnya secara berulang hingga tidak ada lagi kombinasi yang dapat terbentuk.

2. Pemangkasan (*Prune*)

Pada tahap ini, hasil dari kombinasi *item* kemudian dipangkas dengan memanfaatkan nilai *minimum support* yang telah ditentukan sesuai preferensi pengguna.

Langkah awal dalam algoritma *Apriori* melibatkan perhitungan *support* untuk setiap *item* dengan memeriksa *database*. Setelah mendapatkan nilai *support* untuk setiap *item*, *item-item* yang memiliki *support* lebih besar dari *minimum support* ditandai sebagai pola frekuensi tinggi dengan panjang 1, atau dikenal sebagai *1-itemset*. Singkatan *k-itemset* mengacu pada himpunan yang terdiri dari *k-item*.

Iterasi kedua melibatkan pembentukan *2-itemset*, di mana setiap set terdiri dari dua *item*. Proses ini dimulai dengan pembuatan kandidat *2-itemset* dari kombinasi semua *1-itemset* yang ada. Selanjutnya, untuk setiap kandidat *2-itemset*, *support*-nya dihitung dengan melakukan pemindaian ulang pada *database*. *Support* di sini merupakan jumlah transaksi dalam *database* yang mengandung kedua *item* yang ada dalam kandidat *2-itemset*. Setelah *support* dari semua kandidat *2-itemset* berhasil diperoleh, jika

mereka memenuhi syarat *minimum support* yang ditentukan, maka kandidat *2-itemset* tersebut dapat dianggap sebagai *2-itemset* yang juga merupakan pola frekuensi tinggi.

Untuk *iterasi* atau perulangan selanjutnya ke k - akan dibagi lagi menjadi beberapa bagian seperti :

1. Pembentukan kandidat *itemset*

Dalam algoritma *apriori*, kandidat *k-itemset* dibentuk melalui kombinasi dari $(k-1)$ -*itemset* yang dihasilkan dari iterasi sebelumnya. Suatu ciri khas dari algoritma *apriori* adalah pemotongan kandidat *k-itemset* jika *subsetnya* yang memiliki $k-1$ *item* tidak termasuk dalam pola frekuensi tinggi dengan panjang $k-1$. Dengan kata lain, hanya kandidat *k-itemset* yang memiliki *subset* dengan $k-1$ *item* yang juga merupakan pola frekuensi tinggi yang akan dipertimbangkan pada iterasi selanjutnya.

2. Menghitung *support* untuk setiap kandidat *k-itemset*

Support dari setiap kandidat *k-itemset* diperoleh dengan memeriksa *database* untuk menghitung jumlah transaksi yang mengandung semua *item* yang ada dalam kandidat *k-itemset* tersebut. Hal ini juga merupakan ciri dari algoritma *apriori* di mana perhitungan dilakukan dengan melakukan pemindaian pada seluruh *database* untuk setiap *k-itemset* terpanjang.

3. Tetapkan pola frekuensi tinggi

Pola frekuensi tinggi yang mencakup *k-itemset* ditetapkan berdasarkan kandidat *k-itemset* yang memiliki nilai *support* lebih besar dari *minimum support* yang telah ditentukan. Selanjutnya, dilakukan penghitungan nilai

confidence untuk setiap kombinasi *item* yang ada. Iterasi atau perulangan akan berhenti jika semua kombinasi *item* telah dihitung dan tidak ada lagi kombinasi *item* yang tersisa untuk dieksplorasi lebih lanjut.

2.2.10. Association rules

Menurut S.Zhang dkk dalam Buyung SH. *Association rules* merupakan metode yang digunakan untuk mencari pola yang sering terjadi di antara sejumlah transaksi. Setiap transaksi terdiri dari beberapa *item* yang muncul. Dalam analisis *association rules*, korelasi antara *item* tersebut dalam transaksi-transaksi dapat diidentifikasi dan digunakan untuk mendapatkan wawasan yang berharga tentang pola pembelian atau kebiasaan konsumen.

Pada transaksi yang terdapat *item* X terdapat kemungkinan ada *item* Y juga di dalamnya, dinotasikan $X \rightarrow Y$. dimana X dan Y adalah *dis-joint itemset*, dinotasikan sebagai $X \setminus Y$. kumpulan informasi transaksi ini disebut dengan *itemset*, yang dinotasikan dengan I_k ($k= 1, 2, \dots, m$). jika terdapat *itemset* yang mempunyai *item* sebanyak k , maka disebut *k-itemset* [20].

Association rules ini nantinya menghasilkan rules yang menentukan seberapa besar hubungan antara X dan Y tadi, dan diperlukan ukuran untuk *rules* ini yaitu *support* dan *confidence*.

2.2.10.1. Support

Tingkat kemunculan atau *Support* adalah ukuran yang mencerminkan seberapa sering *item* atau *itemset* tertentu muncul dalam keseluruhan transaksi, perhitungan untuk mencari nilai *support* ada pada persamaan 1 dan 2 [21].

$$\text{Support}(X) = \frac{\text{Jumlah transaksi yang mengandung } X}{\text{Total transaksi}} \quad \text{persamaan (1)}$$

$$\text{Support}(X, Y) = \frac{\text{Jumlah transaksi yang mengandung } X \text{ dan } Y}{\text{Total transaksi}} \quad \text{persamaan (2)}$$

dan

$$\text{Support}(X, Y, Z) = \frac{\text{Jumlah transaksi yang mengandung } X, Y \text{ dan } Z}{\text{Total transaksi}} \quad \text{persamaan (3)}$$

2.2.10.2. Confidence

Tingkat kepercayaan atau *Confidence* sebuah nilai yang menggambarkan hubungan antara dua *item*, yaitu sejauh mana *item X* muncul secara teratur dalam transaksi yang mengandung *item Y*, untuk mencari nilai *confidence* perhitungan ini dinotasikan sebagai persamaan 3 dan 4 [20] :

$$\text{Confidence}(X, Y) = \frac{\text{Jumlah transaksi yang mengandung } X \text{ dan } Y}{\text{Jumlah transaksi yang mengandung } X} \quad \text{persamaan (4)}$$

Atau

$$\text{Confidence}(X \rightarrow Y) = \frac{\text{Support}(X, Y)}{\text{Support}(X)} \quad \text{persamaan (5)}$$

2.2.11. Lift Ratio

Lift ratio digunakan dalam menentukan kekuatan dari suatu *rule* (*aturan*). Jika nilai *lift ratio* lebih dari 1 maka nilai *rule* dikatakan mempunyai kekuatan [31]. Perhitungan nilai *lift ratio* menggunakan persamaan untuk 2 *item* pada persamaan 5 dan untuk 3 *item* pada persamaan 6 [22].

$$\text{Lift}(X \rightarrow Y) = \frac{\text{support yang mengandung } X, Y}{\text{nilai Support } X} \quad \text{persamaan (6)}$$

Atau

$$Lift(X, Y \rightarrow Z) = \frac{\text{Support yang mengandung } X, Y, Z}{(\text{Nilai support } XY) \times (\text{nilai support } Z)} \quad \text{persamaan (7)}$$

2.3. Piranti Pendukung

Adapun piranti pendukung yang digunakan dalam penelitian ini, sebagai berikut :

2.3.1. Python



Gambar 2.2 Logo Python
(Sumber: python.org) [30]

Python merupakan suatu bahasa pemrograman yang bersifat interpretatif dan memiliki kegunaan yang beragam. Berbeda dengan bahasa pemrograman lain yang cenderung sulit dibaca dan dipahami, *Python* mengutamakan keterbacaan kode sehingga lebih mudah untuk memahami sintaks-sintaks yang digunakan. Keunikan ini menjadikan *Python* sebagai bahasa pemrograman yang

```

app.py  X
app.py
1  import pandas as pd
2
3  data = pd.read_csv('data.csv')
4
5  print("Data awal:")
6  print(data)
7
8  data['Kolom Baru'] = data['Kolom A'] + data['Kolom B']
9  data = data.drop(['Kolom C'], axis=1)
10
11 data.to_csv('data_modifikasi.csv', index=False)
12
13 print("\nData setelah dimodifikasi:")

```

Gambar 2.3 Contoh Coding Python

mudah untuk dipelajari karena memiliki struktur sintaks yang teratur dan dapat dipahami dengan mudah [23].

2.3.2. *Jupyter Notebook*



Gambar 2.4 Logo Jupyter Notebook
(Sumber: jupyter.org) [32]

Jupyter notebook adalah sebuah perangkat lunak sumber terbuka yang digunakan dalam komputasi interaktif untuk berbagai bahasa pemrograman. *JupyterLab* adalah lingkungan pengembangan interaktif berbasis *web* untuk kode program dan data dalam format *notebook Jupyter*. *JupyterLab* memiliki fleksibilitas dalam mendukung alur kerja untuk data sains, komputasi ilmiah, dan pembelajaran mesin.

Jupyter notebook merupakan sebuah aplikasi *web* sumber terbuka yang ditujukan untuk melakukan berbagai tugas seperti pembersihan dan transformasi data, simulasi angka, visualisasi data, pemodelan statistik, serta pembelajaran mesin, dan masih banyak lagi [23].

```

jupyter apriori Last Checkpoint: 06/12/2023 (autosaved)
File Edit View Insert Cell Kernel Widgets Help
+ %< > Run C Code
In [47]: #import Library
import pandas as pd
import numpy as np

from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import association_rules

```

Gambar 2.5 Contoh Aplikasi Jupyter Notebook

2.3.3. Website



Gambar 2.6 Contoh Website
(Sumber: www.sevenstereo.com) [36]

Website adalah kumpulan dokumen yang disimpan di *server* dan dapat diakses oleh pengguna melalui *browser*. Dokumen tersebut dapat terdiri dari beberapa halaman yang menyediakan berbagai informasi dan interaksi. Informasi dan interaksi yang ditampilkan dapat berupa teks, gambar, video, animasi, suara, dan lainnya [24].

2.3.4. HTML



Gambar 2.7 Logo HTML
(Sumber: codeburst.io) [31]

HTML adalah sebuah format data dalam bentuk dokumen *Hyper-text* yang dapat dipindahkan dari satu sistem ke sistem lain tanpa mengalami perubahan apapun, karena pada dasarnya *HTML* hanya merupakan dokumen teks biasa. Teks atau konten yang terdapat dalam *HTML* disebut sebagai Bahasa Penanda karena mengandung simbol-simbol khusus (tag, elemen, atribut) yang

digunakan untuk mengatur tampilan teks melalui peramban (*browser*). *HTML* digunakan sebagai bahasa dalam *World Wide Web* (disingkat *www*) untuk membuat dokumen tertentu yang dapat ditampilkan dan dilihat melalui peramban [26].

```
contoh.html X
contoh.html > ...
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Contoh Halaman Web</title>
5 </head>
6 <body>
7 <header>
8   <h1>Selamat Datang di Halaman Web Saya</h1>
9 </header>
10
11 <nav>
12 <ul>
13   <li><a href="#">Beranda</a></li>
14   <li><a href="#">Tentang</a></li>
15   <li><a href="#">Kontak</a></li>
16 </ul>
17 </nav>
```

Gambar 2.8 Contoh Coding HTML

2.3.5. CSS



Gambar 2.9 Logo CSS
(Sumber: rootstack.com) [33]

CSS merupakan kependekan dari *Cascading Style Sheet*, yang merupakan dokumen *web* yang digunakan untuk mengatur elemen-elemen *HTML* dengan berbagai properti yang tersedia, sehingga dapat ditampilkan dengan gaya yang diinginkan. Meskipun strukturnya yang sederhana, CSS tidak dianggap sebagai

bahasa pemrograman oleh sebagian orang, karena terdiri dari kumpulan aturan yang mengatur gaya elemen-elemen *HTML*.

```
# contoh.css X
# contoh.css > ...
1  /* style.css */
2
3  body {
4      font-family: Arial, sans-serif;
5      margin: 0;
6      padding: 0;
7  }
8
9  header {
10     background-color: #333;
11     color: #fff;
12     padding: 20px;
13     text-align: center;
14 }
15
```

Gambar 2.10 Contoh Coding CSS

CSS bekerja dengan cara memodifikasi *HTML* melalui pemilihan elemen *HTML* yang akan diatur, kemudian memberikan properti yang sesuai untuk mencapai tampilan yang diinginkan. Dalam memberikan aturan pada elemen *HTML*, skrip CSS terdiri dari tiga bagian utama. Pertama adalah "*Selector*" yang digunakan untuk memilih elemen yang akan diberi aturan. Kedua, "*Property*" merupakan aturan yang diberikan, seperti warna, ukuran *font*, *margin*, dan sebagainya. Terakhir, "*Value*" adalah nilai dari aturan yang diberikan, misalnya nilai warna, ukuran, atau jarak. [26].

2.3.6. *Flask*



Gambar 2.11 Logo Flask
(Sumber: codersera.com) [34]

Flask merupakan sebuah *web framework* yang digunakan bersamaan dengan bahasa *python* yang tergolong kedalam jenis *microframework*. *Flask* mempunyai

fungsi sebagai kerangka kerja pada aplikasi dan tampilan pada suatu *web*. Saat mengembangkan suatu *web* dengan menggunakan *flask* dan bahasa *python*, sebuah *web* dapat terbentuk secara terstruktur dan *behaviour* pada suatu *web* akan dapat diatur dengan mudah, contoh *behaviour* ini seperti navigasi, interaksi dengan elemen, *responsive*, pencarian dan analisis perilaku pengguna. Dalam melakukan instalasi *flask* pada sebuah perangkat, dibutuhkan *Python* versi 3.4 keatas untuk menggunakan *PIP*. Perintah *PIP* digunakan untuk menginstal *flask*, dengan menggunakan *PIP*, semua hal termasuk *library* akan diunduh dan dipasang dalam satu perintah [25].