

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Penelitian Terdahulu**

Penelitian yang dilakukan oleh Rahmad Aditya dan Sarjon Defit dengan judul *Prediksi Tingkat Ketersediaan Stock Sembako Menggunakan Algoritma FP-Growth dalam Meningkatkan Penjualan*. Pada penelitian ini dilakukan proses analisis terhadap data penjualan sembako mulai dari bulan Februari hingga Mei 2020. Dari penelitian yang dilakukan terhadap data transaksi penjualan maka diperoleh 16 pola penjualan sembako yang dihitung menggunakan aplikasi Rapidminer 9.4. Dari pola-pola tersebut dapat direkomendasikan kepada pemilik Toko UD.[8].

Penelitian yang dilakukan oleh Irsyad Djamaludin dan Agus Nursikuwagus dengan judul *Analisis Pola Pembelian Konsumen Pada Transaksi Penjuala Menggunakan Algoritma Apriori*. Pada penelitian ini dilakukan proses analisis terhadap data penjualan selama bulan April 2017. Penelitian dapat disimpulkan bahwa data mining dan teknik algoritma Apriori ini dapat diimplementasikan terhadap data transaksi penjualan, salah satunya untuk mendapatkan pola pembelian konsumen. Pola yang diperoleh, akan menghasilkan suatu rule atau aturan mengenai keterkaitan dari suatu produk. Hasil atau rule yang didapatkan dipengaruhi oleh batasan yang ditentukan, baik banyaknya data, batasan itemset, batasan minimal support dan confidence, serta batasan final association rule yang ditetapkan secara manual. Hasil yang didapatkan

juga dipengaruhi oleh data transaksi penjualannya, terutama varian produk yang ada dalam setiap transaksi[9].

Penelitian yang dilakukan oleh Lalu Aldila Maulana Fajar dan Ria Rismayati dengan judul Rekomendasi Paket Menu Angkringan Waru Tanjung Bias Dengan Algoritma Frequent Pattern Growth Berbasis Web. Pada penelitian ini dilakukan proses analisis terhadap data penjualan pada Angkringan Waru dalam kurun waktu 6 bulan. Dari penelitian ini dihasilkan 57 pola menu.[10].

Persamaan penelitian yang dilakukan oleh penulis dengan penelitian yang dilakukan oleh Rahmad Aditya dan Sarjon Defit, dan penelitian yang dilakukan oleh Lalu Aldila Maulana Fajar dan Ria Rismayati, adalah algoritma yang digunakan yaitu *FP-Growth*. Terdapat perbedaan pada penelitian yang dilakukan oleh Irsyad Djamaludin dan Agus Nursikuwagus yaitu menggunakan algoritma *Apriori*. Perbedaan pada penelitian yang dilakukan penulis dengan penelitian yang dilakukan oleh Rahmad Aditya dan Sarjon Defit yaitu menggunakan aplikasi *RapidMiner*. Sedangkan penelitian yang dilakukan oleh penulis menggunakan bahasa pemrograman *Python* serta menggunakan *Text Editor VisualStudio Code*. Ada perbedaan juga pada tujuan penelitiannya yaitu penulis membuat analisis pola penjualan produk berbasis *web*.

## **2.2 Teori Pendukung**

Teori pendukung adalah istilah penelitian yang digunakan sebagai referensi untuk memahami penelitian yang dilakukan oleh penulis.

### 2.2.1 Penjualan

Pada dasarnya, setiap interaksi antara penjual dan pembeli dapat menghasilkan transaksi atau penjualan. Selama proses penjualan, uang digunakan sebagai alat tukar untuk memenuhi kebutuhan. Di pasar, penjualan sendiri biasanya dilakukan dengan bantuan pelayan, yang berfungsi sebagai perantara antara penjual dan pembeli[11]. Penjualan adalah kegiatan menjual produk atau jasa kepada konsumen dengan tujuan memperoleh keuntungan[12]. Penjualan dapat dilakukan secara langsung maupun tidak langsung melalui berbagai saluran distribusi, seperti toko fisik, toko *online*, agen, dan lain sebagainya.

### 2.2.2 Data Mining

*Data mining* adalah suatu proses ekstraksi atau penggalian data dan informasi yang sangat besar, yang belum diketahui sebelumnya, namun dapat dipahami dengan baik dari *database* yang besar dan digunakan untuk membuat keputusan penting bagi bisnis[13].

*Data mining* digunakan untuk menggambarkan penemuan pengetahuan di dalam *database*. Ini adalah proses yang mengekstraksi dan mengidentifikasi informasi penting dan terkait dari berbagai *database* yang sangat besar dengan menggunakan teknik seperti statistik, matematika, kecerdasan buatan, dan pembelajaran mesin[3].

### 2.2.3 Teknik *Data Mining*

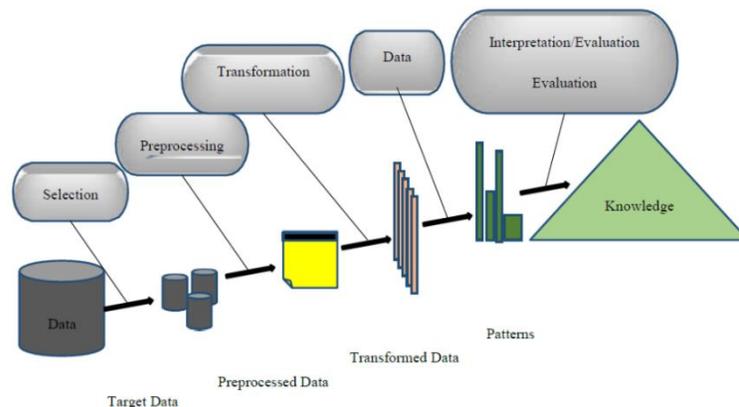
Beberapa teknik dan sifat data mining adalah sebagai berikut :

1. Klasifikasi adalah komponen *data mining* di mana sistem menyusun kelompok dan golongan menurut standar yang ditetapkan. Klasifikasi data adalah suatu proses yang menemukan karakteristik yang sama pada sekumpulan objek dalam basis data dan kemudian membaginya ke dalam kelas-kelas yang berbeda menggunakan model klasifikasi yang telah ditetapkan[14]. Ini juga disebut dengan “*supervised learning*”.
2. *Clustering* adalah pengelompokan sekumpulan objek ke dalam kelas atau pola objek yang sama . *Cluster* membagi data menjadi kelompok-kelompok atau cluster berdasarkan kesamaan karakteristik di antara sekumpulan data, yang ditunjukkan sebagai titik-titik dalam ruang multidimensi[15]. Dapat disebut juga “*unsupervised learning*”.
3. *Association rules* adalah salah satu metode yang bertujuan untuk menemukan pola yang sering muncul di antara banyak transaksi, di mana setiap transaksi terdiri dari beberapa item. Metode ini akan mendukung sistem rekomendasi dengan menemukan pola antar item dalam transaksi tersebut[16].
4. *Regresi* adalah suatu teknik statistika dalam mengidentifikasi data yang bertujuan untuk mengetahui hubungan antara sejumlah variabel. Dimana

variabel respon yang bersifat kategorik, baik nominal maupun ordinal dengan variabel penjelas yang bersifat kategorik atau kontinu[17].

#### 2.2.4 Knowledge Discovery in Database (KDD)

Untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar, istilah "*Data Mining*" dan "*Knowledge Discovery in Database*" sering digunakan secara bergantian. Padahal, istilah-istilah tersebut sebenarnya merujuk pada konsep yang berbeda, akan tetapi saling berkaitan satu sama lain. *Data mining* merupakan bagian dari keseluruhan tahapan yang ada di *Knowledge Discovery in Database*[18]. Berikut ini tahapan dari proses *Knowledge Discovery in Database (KDD)* dapat dilihat pada gambar :



**Gambar 2. 1 Proses KDD**

**(Sumber : Jurnal Nasional Ilmu Komputer [19])**

1. *Data Selection*

Sebelum tahap penggalian informasi, sekumpulan data operasional diseleksi untuk memilih data dan karakteristik yang akan digunakan untuk proses berikutnya. Hasil seleksi ini disimpan dalam berkas yang berbeda dari data operasional dan digunakan dalam proses *data mining*.

2. *Pre-processing (Cleaning)*

Pada data hasil seleksi akan dilakukan tahapan *cleaning* atau *pre-processing*, tahapan ini dilakukan sebelum proses *data mining*. Tujuan dari tahapan *cleaning* ini adalah untuk menghilangkan duplikasi data dan ketidakkonsistenan data.

3. Transformasi

Tahap transformasi dilakukan setelah tahap *pre-processing* dan *cleaning data*, yaitu tahap yang menghasilkan data yang siap ditambah dengan mengubah data dari bentuk aslinya sesuai kebutuhan. Ini dilakukan agar proses penggalian data lebih mudah menemukan informasi baru.

4. *Data Mining*

*Data mining* adalah proses menemukan pola atau informasi yang menarik pada data yang telah dipilih dengan menggunakan metode atau teknik

tertentu. Memilih metode atau algoritma yang tepat sangat penting karena berdampak pada tujuan dan proses *KDD* secara keseluruhan.

#### 5. *Interpretation (Evaluation)*

Pada tahap ini, pola yang dihasilkan dari *data mining* akan diidentifikasi. Pola informasi yang dihasilkan harus ditampilkan dalam bentuk yang mudah dipahami oleh pihak yang berkepentingan[19].

#### 2.2.5 *Association Rule*

Salah satu metode pengolahan data adalah *association rule*, yang digunakan untuk mengidentifikasi hubungan antara variabel dalam data transaksi. Pola tertentu yang dihasilkan oleh aturan asosiasi akan digunakan untuk mengasosiasikan satu data dengan data lainnya. Untuk mencari *association rule* dari suatu kumpulan data, tahap pertama yang harus dilakukan adalah mencari *frequent itemset* terlebih dahulu[20].

#### 2.2.6 *Tahapan Association Rule*

Aturan asosiasi adalah teknik *data mining* untuk menemukan aturan asosiasi antara kombinasi item, teknik ini merupakan dasar dari berbagai teknik lainnya. Tahap asosiasi dikenal juga sebagai analisis pola frekuensi tinggi (*frequent pattern mining*) yang mendorong para peneliti untuk

menghasilkan algoritma yang efektif. Tahapan metodologi dasar analisis asosiasi adalah sebagai berikut :

### 1. Analisa Pola Frekuensi Tinggi

Analisa pola frekuensi tinggi digunakan untuk menemukan kombinasi item yang memenuhi syarat minimum dari nilai pendukung yang ada dalam *database*. Nilai pendukung sebuah item diperoleh dengan menggunakan rumus berikut:

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

Sedangkan nilai dari *support* dua item diperoleh dari rumus berikut:

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

### 2. Pembentukan Aturan Asosiasi

Setelah semua pola frekuensi tinggi ditemukan, pembentukan aturan asosiasi dapat dimulai. Langkah selanjutnya adalah mencari aturan asosiasi yang memenuhi syarat minimum dengan menggunakan rumus sebagai berikut :

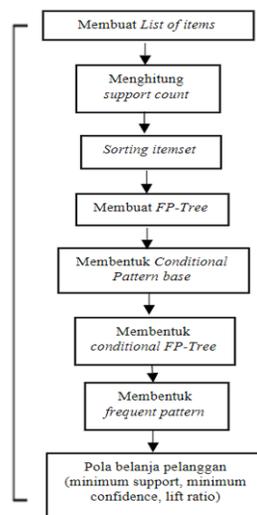
$$Confidence(A|B) = \frac{Jumlah\ Transaksi\ A\ dan\ B}{Total\ Transaksi\ A} \times 100\% \quad [21].$$

#### 2.2.7 *FP-Growth*

*FP-Growth* adalah salah satu algoritma alternatif untuk menemukan himpunan data yang paling sering muncul (*frequent itemset*) dalam Kumpulan

data. Dibandingkan dengan algoritma lain dalam metode asosiasi, seperti algoritma *apriori*, *FP-Growth* lebih cepat menemukan himpunan data yang paling sering muncul dan dapat menghasilkan aturan yang sama dengan algoritma *apriori* setelah diujikan pada transaksi penjualan. Tujuan algoritma *FP-Growth* adalah untuk menemukan pola-pola yang sering terjadi dalam data transaksi, seperti *itemset* yang paling sering dibeli bersamaan, dan untuk meningkatkan efisiensi pengelolaan data. Oleh karena itu, algoritma ini membantu bisnis meningkatkan strategi penjualan mereka dan meningkatkan keuntungan mereka[22]. *FP-growth* adalah algoritma alternatif yang dapat digunakan untuk menentukan item yang paling sering muncul dalam dataset. Ini menggunakan pendekatan yang berbeda dari paradigma yang digunakan dalam algoritma Apriori.[23].

### 2.2.8 Tahapan *FP-Growth*



**Gambar 2. 2 Tahapam *FP-Growth***

(Sumber:Jurnal of Informatics and Computer Science (JINACS) [22])

1. Membuat *List of Items*

Tahap pertama proses *FP-Growth* adalah membuat *list of item*. *List of item* adalah daftar barang apa pun yang dibeli pada setiap transaksi.

2. Menghitung *Support Count*

Setelah mendapatkan *list of item* dari setiap transaksi, hitung *support count* untuk setiap item yang dibeli pada data transaksi. *Support count* adalah jumlah kali suatu item muncul dalam seluruh data transaksi.

3. *Sorting Itemset*

Dalam proses ini, setiap itemset akan diurutkan menurut jumlah total *support count* terbesar.

4. Membuat *FP-Tree*

Pada tahap ini, *FP-Tree* akan dibuat dari data transaksi yang ada dimulai dari item dengan *support count* tertinggi hingga item dengan *support count* terkecil.

5. Membentuk *Conditional Pattern Base*

Dalam proses ini, *conditional pattern base* akan dibentuk dari tiap *node* yang ada pada *fp-tree*, kecuali *itemset* yang tidak memiliki *parent* atau tidak merupakan *children*. Pertama, kita akan mengambil semua *node* yang memiliki *parent*, atau bisa dikatakan *node* tersebut adalah *children*, kemudian kita akan mengambil *parent* dari tiap *itemset* pada pohon yang telah terbentuk. *Conditional pattern base* terdiri dari *parent* dari tiap *node* dan jumlah *support count* dari *node* yang memiliki anak.

#### 6. Membentuk *Conditional FP-Tree*

Pada tahap ini, setiap *node* yang terbentuk pada *conditional pattern base* harus memiliki *support count* lebih dari 1 atau minimal 2. Kemudian, untuk membuat *conditional FP-Tree*, ambil setiap *parent* pada *conditional pattern base* dan hitung *support count* total dari tiap *parent node* yang memiliki *support count* minimal 2.

#### 7. Membentuk *Frequent Pattern*

*Frequent pattern* dibentuk dari kombinasi yang mungkin dari tiap *Conditional FP-Tree* dengan *node children* kemudian dihitung *support count* dari setiap kombinasi.

#### 8. Pola Belanja Pelanggan

Dalam proses ini pertama dilakukan *filter* terhadap *frequent pattern* yang sesuai dengan nilai minimum *support* yang telah dimasukkan pengguna. Untuk memperoleh hasil minimum *support* dari tiap *frequent pattern* yang terbentuk dapat dirumuskan dalam :

$$\text{Minimum Support}(A) = \frac{\text{Support Count}}{\text{Total Transaksi}} \times 100\% \dots \dots \dots (1)$$

Selanjutnya, yaitu membuat *rules* dari *subset* pada *frequent pattern* yang sesuai dengan nilai minimum *support*.

Nilai minimum *confidence* akan diperoleh dari *rules* yang telah dibuat. Hasil minimum *confidence* dapat dirumuskan pada :

$$\text{Minimum Confidence} = \frac{\text{Support Count}(A,B)}{\text{Support Count } A} \times 100\% \dots \dots \dots (2)$$

Terakhir yaitu menghitung *lift ratio* dari tiap *rules* yang terbentuk. Untuk memperoleh hasil *lift ratio* dapat dirumuskan pada :

$$\text{Lift} = \frac{\text{Confidence}(A,B)}{\text{Support Count } B} \times 100\% \dots \dots \dots (3)$$

Selanjutnya, *rules* akan ditampilkan berdasarkan pola belanja pelanggan yang memenuhi nilai minimum *support*, nilai minimum *confidence* dan nilai *lift ratio* lebih dari 1[20].

### 2.2.9 Prinsip Pareto

Prinsip Pareto dikemukakan oleh seorang ahli ekonomi-sosiologi Italia bernama Alfredo Pareto pada tahun 1906. Alfredo Pareto adalah seorang Profesor dibidang Ekonomi Politik di Lausanne, Swiss. Prinsip Pareto dikenal juga dengan berbagai istilah seperti The Principle of Least Effort dan The Principle of Imbalance. Prinsip Pareto (Pareto Principle) bermula pada tahun 1906, Alfredo Pareto mengamati bahwa 80% dari pendapatan di Italia dimiliki oleh 20% dari jumlah populasi penduduk Italia. Prinsip utama dalam aturan 80:20 pareto ini adalah bahwa dengan memaksimalkan 20% usaha maka akan memberikan hasil 80% secara maksimal.

Dalam implementasinya, prinsip pareto tersebut juga dapat diterapkan di berbagai hal diantaranya seperti 80% dari keluhan pelanggan muncul dari 20% dari produk atau jasa yang diberikan; 80% dari output yang dihasilkan diperoleh dari 20% masukan;

20% dari cacat sistem akan menyebabkan 80% masalah; serta 20% dari memaksimalkan produk atau jasa maka akan mencapai 80% dari keuntungan[24].

## 2.3 Alat Pendukung

Terdapat beberapa alat pendukung yang digunakan untuk membantu penulis menyelesaikan penelitian, Diantaranya :

### 2.3.1 *Python*

*Python* adalah bahasa pemrograman yang paling banyak digunakan dan cocok untuk pemula yang ingin belajar membuat program komputer. Dalam beberapa tahun terakhir, *Python* telah menjadi salah satu bahasa pemrograman yang paling populer untuk membuat aplikasi analisis data. *Python* memiliki banyak *library* dan modul yang mendukung penggunaan algoritma klasifikasi seperti *Naive Bayes*, dan kemudahan penggunaannya membuatnya menjadi pilihan yang menarik untuk menerapkan algoritma klasifikasi *Naive Bayes* untuk klasifikasi data[21]. Berikut adalah daftar kerangka kerja paling populer untuk melakukan analisis data :

1. *Numpy* adalah kerangka kerja *Python* yang paling populer untuk perhitungan numerik.
2. *SciPy* adalah modul yang berfokus pada sains, matematika, dan Teknik.
3. *Scikit-Learn* adalah kerangka kerja pembelajaran mesin *Python* yang digunakan untuk penambangan data yang produktif, yang memungkinkan untuk

melakukan proses *regresi*, pengelompokan, pemilihan model, *preprocessing*, dan klasifikasi.

4. *Dask* adalah kerangka kerja untuk paralelisme lanjutan untuk perhitungan numerik yang kompleks[21].



**Gambar 2. 3 Logo Python**

**(Sumber:python.org[25])**

### ***2.3.2 HTML (HyperText Markup Language)***

*Hypertext Markup Language (HTML)* adalah *script* pemograman yang mengatur cara kita menyajikan informasi di internet dan bagaimana informasi itu membawa kita dari satu tempat ke tempat lain. *HTML* sangat berkembang pada awal tahun 1990. Setiap pengembangan *HTML* pasti akan meningkatkan kemampuan dan fitur dari versi sebelumnya[26].



**Gambar 2. 4 Logo Html**

(Sumber:Wikipedia.com[27])

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title></title>
6 </head>
7 <body>
8
9 </body>
10 </html>
```

**Gambar 2. 5 Contoh Coding HTML**

(Sumber:Dicoding.com[28])

### 2.3.3 CSS (*Cascading Style Sheet*)

*Cascading Style Sheet (CSS)* merupakan salah satu kode pemrograman yang bertujuan untuk membuat dan mengatur gaya tampilan atau layout halaman web agar lebih menarik. Jika dibandingkan dengan bahasa pemrograman inti seperti *HTML* dan *PHP*, *CSS* memiliki banyak fitur yang dapat dilakukan. Anda memiliki kemampuan untuk mengubah warna teks, jenis

font, baris antar paragraf, ukuran kolom, dan jenis *background* yang dilakukan dengan menggunakan *CSS*. Selain itu, *CSS* memiliki kemampuan untuk mendesain *layout*, mengubah tampilan pada berbagai perangkat, dan menerapkan efek yang berbeda pada web. *CSS* tidak hanya mudah dipelajari, tetapi juga sangat efektif karena dapat mengontrol bagaimana dokumen *HTML* ditampilkan. [26].



**Gambar 2. 6 Logo CSS**  
**Sumber: Wikipedia.com[29]**

```
1  /* CSS untuk membuat teks berwarna merah */
2  h1 {
3      color: red;
4  }
5  /* CSS untuk mengatur lebar dan tinggi gambar
6  img {
7      width: 200px;
8      height: 200px;
9  }
```

**Gambar 2. 7 Contoh Coding CSS**  
**(Sumber:rumahweb.com[30])**

#### **2.3.4 Website**

Website adalah kumpulan halaman atau data yang diakses melalui jalur internet. Selama seseorang terhubung ke internet, setiap orang dapat menggunakannya kapan saja dan di mana saja. Secara teknis, sebuah website adalah kumpulan dari halaman web yang digabungkan ke dalam domain atau subdomain tertentu. Website dapat berupa blog, toko online, portal berita, aplikasi web, dan banyak lagi. Fungsi utama sebuah website adalah untuk menyajikan informasi kepada pengunjung, memfasilitasi interaksi, dan memberikan layanan yang berguna. [31].