

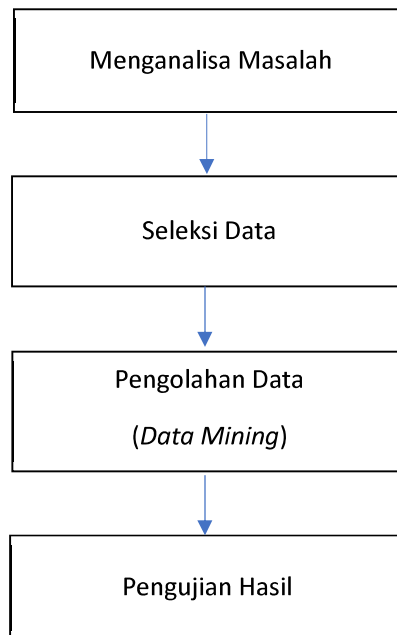
BAB III

METODOLOGI PENELITIAN

3.1 Menganalisa Masalah

Pada bab ini dilakukan analisis masalah yang lebih mendalam terkait dengan pengelolaan stok barang dan bagaimana algoritma *FP-Growth* dapat diterapkan untuk memberikan rekomendasi stok barang berdasarkan pola pembelian konsumen. Tahap menganalisa masalah mencakup identifikasi masalah utama, penentuan metode yang akan digunakan, dan langkah-langkah penyelesaian yang dirancang.

Gambar 3.1 adalah tahapan perencanaan penelitian yang menyajikan hasil model penelitian yang terstruktur, sehingga berfungsi sebagai panduan referensi untuk membantu peneliti



Gambar 3.1 Desain Penelitian

3.1.1. Identifikasi Masalah

Pada penelitian ini, masalah utama yang dihadapi adalah bagaimana menyediakan stok barang secara optimal berdasarkan pola pembelian konsumen. Tantangan yang diidentifikasi meliputi:

- a. Fluktuasi permintaan: Tidak adanya pola pembelian yang stabil, yang membuat perencanaan stok barang menjadi sulit.
- b. Kelebihan atau kekurangan stok: Tanpa prediksi yang baik, manajemen sering kali mengalami kelebihan stok barang yang menumpuk atau kekurangan stok yang mengakibatkan hilangnya penjualan.
- c. Kurangnya pemanfaatan data transaksi penjualan: Banyak perusahaan memiliki data transaksi yang besar namun belum dimanfaatkan secara optimal untuk menganalisis dan memprediksi pola pembelian yang relevan bagi perencanaan stok.

3.1.2 Penentuan Metode Penyelesaian Masalah

Untuk memecahkan masalah yang telah diidentifikasi, digunakan pendekatan berbasis data mining dengan algoritma *FP-Growth*. *FP-Growth* dipilih karena kemampuannya dalam menambang pola pembelian dari data transaksi penjualan secara efisien tanpa harus melalui eksplorasi berulang yang memakan waktu. Langkah-langkah utama dalam penyelesaian masalah adalah sebagai berikut:

- a. Mengumpulkan data transaksi penjualan dari sistem yang ada (misalnya dari sistem manajemen gudang atau ERP) yang mencatat kode barang yang dibeli dalam setiap transaksi.
- b. Melakukan pembersihan dan seleksi data untuk memastikan bahwa hanya data yang relevan dan valid yang digunakan dalam analisis.
- c. Mengaplikasikan algoritma *FP-Growth* pada data transaksi untuk menemukan *frequent itemset*, yaitu kombinasi barang yang sering dibeli bersama.
- d. Membuat aturan asosiasi dari *frequent itemset* tersebut untuk mengetahui hubungan antar barang yang dibeli secara bersamaan,

yang dapat membantu memprediksi permintaan barang di masa mendatang.

- e. Mengembangkan sistem rekomendasi stok barang berdasarkan hasil analisis pola pembelian, yang dapat digunakan oleh manajemen untuk mengoptimalkan stok barang di gudang.

3.2 Seleksi Data

Seleksi data merupakan langkah penting dalam proses penelitian, bertujuan untuk memilih data yang relevan dari kumpulan data transaksi penjualan untuk dianalisis lebih lanjut menggunakan algoritma *FP-Growth*. Seleksi data memastikan bahwa hanya data yang relevan, valid, dan berkualitas yang digunakan dalam proses analisis.

3.2.1 Kriteria Seleksi Data

Kriteria seleksi data yang digunakan dalam penelitian ini meliputi:

- a. Periode waktu transaksi: Data yang dipilih harus mencakup periode waktu yang cukup panjang untuk mengidentifikasi pola pembelian yang konsisten.
- b. Frekuensi pembelian: Hanya barang-barang yang sering dibeli bersama yang akan dianalisis, karena item yang jarang dibeli bersama mungkin tidak memberikan insight yang signifikan.
- c. Kategori barang: Fokus pada kategori barang yang strategis, seperti barang dengan perputaran cepat, barang yang memiliki nilai tinggi, atau barang yang penting bagi manajemen stok.

3.2.2 Proses Seleksi Data

- a. Pengumpulan Data: Mengambil data transaksi dari sistem manajemen gudang atau database penjualan.
- b. Pembersihan Data: Menghilangkan data duplikat, transaksi yang tidak lengkap, atau transaksi yang mengandung kesalahan (misalnya, kode barang tidak valid).
- c. Seleksi Data Berdasarkan Frekuensi: Memprioritaskan data barang yang sering dibeli bersama oleh konsumen dalam transaksi.

- d. Seleksi Berdasarkan Kategori Barang: Mengelompokkan barang berdasarkan kategori yang relevan dengan tujuan penelitian.

3.2.3 Tantangan Seleksi Data

Beberapa tantangan dalam seleksi data meliputi:

- a. Volume data yang besar: Data transaksi yang besar membutuhkan proses seleksi yang efisien untuk mendapatkan data yang relevan tanpa mengorbankan kualitas analisis.
- b. Variabilitas dalam pola pembelian: Pola pembelian konsumen yang berbeda-beda dari waktu ke waktu dapat mempersulit pemilihan data yang paling relevan untuk dianalisis.

3.3 Pengolahan Data

Setelah proses seleksi data, data yang terpilih kemudian diolah untuk menghasilkan informasi yang berguna. Pengolahan data dalam penelitian ini mencakup transformasi data transaksi menjadi format yang bisa digunakan oleh algoritma *FP-Growth* dan proses analisis pola pembelian.

3.3.1 Praproses Data

Praproses data dilakukan untuk memastikan bahwa data yang akan diolah oleh algoritma *FP-Growth* memiliki format yang sesuai dan kualitas yang baik. Langkah-langkah dalam praproses data meliputi:

- a. Normalisasi Data: Menstandarkan format kode barang dan informasi transaksi agar seragam, seperti memastikan konsistensi kode barang di semua transaksi.
- b. Penghapusan Data Tidak Relevan: Menghapus data transaksi yang tidak relevan seperti transaksi yang tidak lengkap atau transaksi yang berkaitan dengan barang yang tidak diikutsertakan dalam analisis.
- c. Transformasi Data: Mengubah data transaksi menjadi format yang sesuai untuk diolah oleh algoritma *FP-Growth*. Data transaksi diubah menjadi bentuk list barang yang dibeli dalam setiap transaksi.

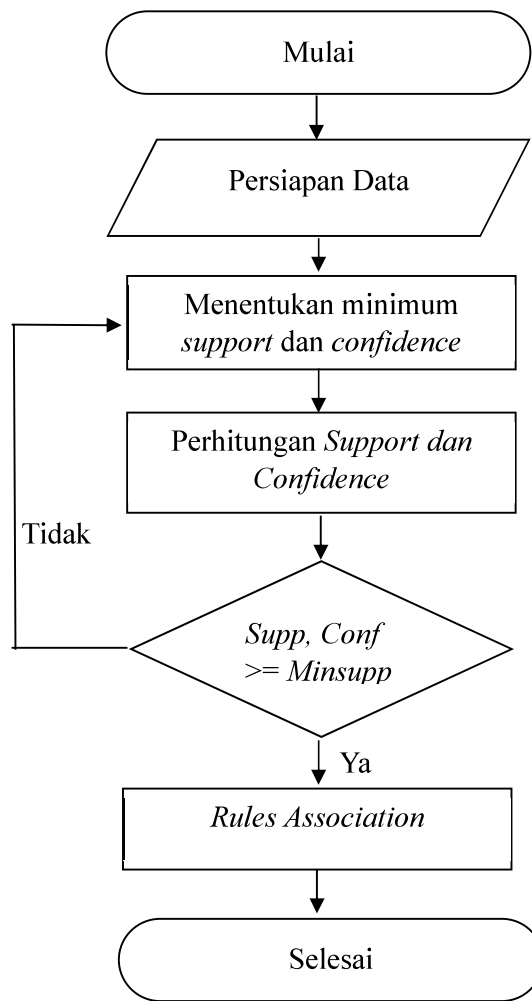
3.3.2 Aplikasi Algoritma *FP-Growth*

Algoritma *FP-Growth* dikembangkan dari algoritma *Apriori* untuk memperbaiki kekurangannya. *FP-Growth* tidak memerlukan *generate*

candidate karena menggunakan konsep pembangunan *tree* dalam pencarian *frequent itemset*. *Frequent itemset* dapat langsung diekstrak dari struktur *FP-Tree* dan hasilnya dapat diketahui. *FP-Tree* dibangun dengan memetakan setiap data transaksi ke dalam setiap lintasan tertentu dalam *FP-Tree*, yang memungkinkan untuk saling menimpa karena transaksi mungkin memiliki item yang sama. Semakin banyak data transaksi yang memiliki item yang sama, maka proses pemanpatan dengan struktur data *FP-Tree* semakin efektif. Kelebihan dari *FP-Tree* adalah hanya memerlukan dua kali pemindaian data transaksi yang terbukti sangat efisien.

Metode FP-Growth terbagi menjadi 3 tahapan yaitu:

- a. Pembentukan *FP-Tree*: Pada tahap ini, *support count* dari setiap item pada setiap *conditional pattern base* dijumlahkan, lalu setiap item yang memiliki jumlah *support count* lebih besar atau sama dengan minimum *support count* yang akan dibangkitkan dengan *conditional FP-Tree*
- b. Pembentukan *Conditional Pattern Base*: *Conditional Pattern Base* merupakan *sub database* yang berisi *prefix path* (lintasan *prefix*) dan *suffix path* (pola akhiran). Pembangkitan *conditional pattern base* didapatkan melalui *FP-Tree* yang telah dibangun sebelumnya.
- c. Ekstraksi *Frequent itemset*: Apabila *conditional FP-Tree* merupakan lintasan Tunggal (*single path*), maka didapatkan *frequent pattern* dengan melakukan kombinasi item untuk setiap *conditional FP-Tree*. Jika bukan lintasan Tunggal, maka dilakukan pembangkitan *FP-Growth* secara rekursif



Gambar 3.2 Flowchart Algoritma FP-Growth

Berikut penjelasan gambar 3.2 untuk tahapan alur informasi dari algoritma *FP-Growth* :

1. Persiapan data : Memasukan data kedalam dataset untuk dilakukan analisis
2. Menentukan minimum *support* dan *confidence*: Tentukan nilai minimum dari nilai *support* dan nilai *confidence* yang akan menjadi parameter untuk melakukan analisis.
3. Perhitungan *Support* dan *confidence* : Algoritma *FP-Growth* akan menganalisis data berdasarkan parameter dari nilai minimum *support* dan *confidence* yang dimasukan.
4. Setelah dilakukan perhitungan analisis algoritma akan mengecek kesesuaian data jika data yang dihasilkan lebih besar atau sama dengan *minimum support* dan

minimum confidence maka akan dibentuk aturan assosiasinya, jika tidak maka akan kembali ke pengisian nilai minimum

5. *Rules Association*: Merupakan hasil dari analisis data yang dilakukan oleh algoritma FP-Growth, hasilnya berupa pola pembelian.

```
Input:
- Dataset transaksi D
- Minimum Support (min_sup)
- Minimum Confidence (min_conf)

Output:
- Frequent Itemsets L
- Association Rules dengan Confidence dan Lift

1. Mulai

2. // Fase 1: Hitung Frequent Itemsets
   a. Hitung support setiap item tunggal di D
   b. Buang item dengan support < min_sup
   c. Simpan item yang lolos sebagai L1 (Frequent 1-Itemset)

3. K = 2 // Inisialisasi ukuran itemset
   While L(K-1) ≠ ∅:
       a. Hasilkan kandidat itemset Ck dari kombinasi itemset L(K-1)
       b. Hitung support setiap kandidat itemset Ck dalam D
       c. Simpan itemset yang memiliki support ≥ min_sup sebagai Lk
       d. K = K + 1

4. Gabungkan semua Lk untuk mendapatkan Frequent Itemsets L

5. // Fase 2: Generate Association Rules
   For setiap frequent itemset F ∈ L:
       a. Bagi F menjadi semua kombinasi antecedent (X) dan consequent (Y)
          - Syarat:  $X \cup Y = F$  dan  $X \cap Y = \emptyset$ 
       b. Hitung Confidence:
           $Confidence(X \rightarrow Y) = Support(F) / Support(X)$ 
       c. Jika Confidence ≥ min_conf:
          - Hitung Lift:
              $Lift(X \rightarrow Y) = Confidence(X \rightarrow Y) / Support(Y)$ 
          - Simpan aturan asosiasi (X → Y) beserta nilai Confidence dan Lift

6. Selesai
```

Gambar 3.3 proses *Algoritma FP-Growth*

3.3.3 Pembuatan *Association Rules*

Setelah *frequent itemset* ditemukan, langkah selanjutnya adalah membuat *association rules*. *Association rules* menunjukkan hubungan antara barang-barang tertentu yang sering dibeli bersama, sehingga dapat digunakan untuk memprediksi barang yang mungkin dibutuhkan dalam stok. Aturan asosiasi ini akan menjadi dasar untuk membuat rekomendasi stok barang, ada dua ukuran ketertarikan yang umum digunakan, yaitu:

- Support*, adalah dukungan atau probabilitas bahwa konsumen membeli lebih dari satu produk secara bersamaan dari jumlah total transaksi. Ukuran ini menentukan apakah suatu *Items/Itemset* layak untuk dicari nilai *Confidence*-nya.
- Confidence* atau tingkat kepercayaan merupakan probabilitas terjadinya beberapa produk yang dibeli secara bersamaan dimana salah satu produk sudah pasti dibeli.

Cara untuk memperoleh nilai dukungan dari suatu item A adalah melalui rumus berikut :

Cara untuk memperoleh nilai dukungan dari suatu item A adalah melalui rumus (1) berikut :

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

Nilai support dari suatu item A

Setelah itu, untuk memperoleh nilai dukungan dari dua item dapat dilakukan menggunakan rumus (2) berikut :

$$Support(A, B) = \frac{\Sigma Transaksi\ mengandung\ A\ dan\ B}{\Sigma Transaksi} \times 100\% \quad (2)$$

Nilai support dari dua item

Setelah memperoleh semua item yang sering muncul (*frequent item*) dan himpunan item besar (*large item set*), kita bisa mencari kepercayaan minimum (*mincof*) dengan menggunakan rumus (3) berikut :

$$Confidence = P(B | A) = \frac{\Sigma \text{Transaksi mengandung A dan B}}{\Sigma \text{Transaksi mengandung A}} \times 100\% \quad (3)$$

Minimum confidence (mincof)

3.3.4 Korelasi antara *Support* dan *Confidence*

a. *Support* sebagai Dasar *Confidence*:

Nilai *support* digunakan sebagai komponen dasar untuk menghitung *confidence*. *Confidence* adalah rasio antara *support itemset* A dan B dengan *support antecedent* (A).

b. *Support* Rendah, *Confidence* Tinggi:

Dalam beberapa kasus, meskipun *support* suatu *itemset* rendah, nilai *confidence* masih bisa tinggi jika *antecedent* memiliki *support* rendah tetapi *itemset A U B* sering muncul bersama.

Contoh: Jika produk A jarang dibeli, tetapi ketika A dibeli, B selalu muncul, maka *confidence* tetap tinggi.

c. *Support* Tinggi, *Confidence* Rendah:

Jika *antecedent* memiliki *support* yang tinggi tetapi *itemset* lengkap (**A U B**) memiliki frekuensi yang relatif rendah, maka *confidence* akan lebih kecil.

Contoh: Produk A sering dibeli, tetapi tidak selalu bersamaan dengan produk B.

3.3.5 Validasi Hasil Pengolahan Data

Setelah proses pengolahan data selesai dan *association rules* terbentuk, hasil dari pengolahan data diuji untuk memastikan keakuratannya. Pengujian dilakukan dengan menggunakan data transaksi historis yang belum dianalisis, untuk melihat seberapa baik sistem dapat memprediksi pola pembelian dan kebutuhan stok di masa mendatang.