

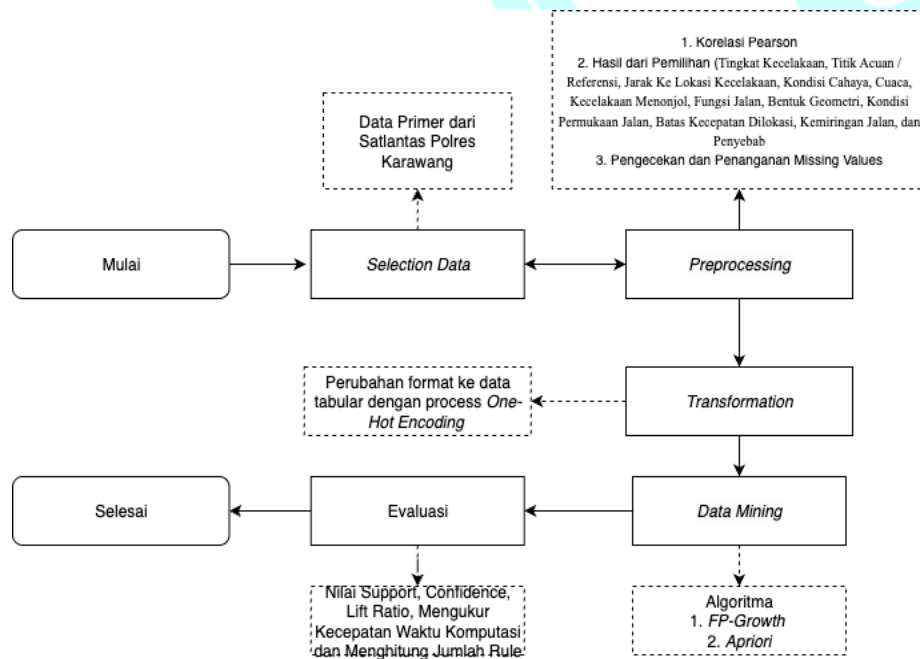
BAB III METODE PENELITIAN

3.1 Objek Penelitian

Pada penelitian ini, data yang digunakan diperoleh dari Satuan Lalu Lintas (Satlantas) Polres Karawang dengan jumlah sebanyak 4.021 data, yang mencakup periode dari tahun 2020 hingga 2024. Data ini dipilih sebagai objek penelitian karena memiliki cakupan yang menyeluruh dan relevan untuk dianalisis dalam upaya mengidentifikasi pola-pola kecelakaan lalu lintas di wilayah Karawang. Keberadaan variabel yang beragam memungkinkan dilakukannya analisis hubungan antar faktor penyebab kecelakaan menggunakan metode *association rule mining* dengan algoritma FP-Growth dan Apriori.

3.2 Prosedur Penelitian

Subab ini akan menjelaskan langkah-langkah yang akan dilakukan penelitian, Prosedur penelitian mencakup tahapan mulai dari pengumpulan data hingga evaluasi, yang dirancang untuk memastikan pelaksanaan penelitian berlangsung secara terstruktur. Langkah-langkah yang dilakukan sebagai berikut.



Gambar 3. 1 Prosedur Penelitian

Sumber : (Dokumen pribadi)

3.2.1 Selection Data

Tahap **selection data** dimulai dengan proses pengumpulan data primer dari Satuan Lalu Lintas (Satlantas) Polres Karawang. Dataset yang digunakan merupakan catatan historis seluruh insiden kecelakaan lalu lintas yang terjadi dalam rentang waktu tahun 2020 hingga 2024. Data mentah ini tersimpan dalam format file Excel dan memiliki struktur komprehensif yang terdiri dari 31 variabel dengan total 4.021 baris data, di mana setiap baris merepresentasikan satu catatan unik kejadian kecelakaan. Dataset ini mewakili kejadian laka lantas pada setiap barisnya yang terdiri dari 31 variabel yaitu Tanggal Kejadian, Waktu Kejadian, Tanggal Dilaporkan, Waktu Dilaporkan, Tingkat Kecelakaan, Jumlah Meninggal Dunia, Jumlah Luka Berat, Jumlah Luka Ringan, Koordinat GPS – Lintang, Koordinat GPS – Bujur, Titik Acuan/Referensi, Jarak Ke Lokasi Kecelakaan, Arah Dari Titik Acuan ke Lokasi Kecelakaan, Tipe Kecelakaan, Kondisi Cahaya, Cuaca, Kecelakaan Menonjol, Nomor Jalan, Nama Jalan, Titik Jalan, Fungsi Jalan, Kelas Jalan, Tipe Jalan, Bentuk Geometri, Kondisi Permukaan Jalan, Batas Kecepatan Dilokasi, Kemiringan Jalan. Visualisasi sampel dari data tersebut dapat dilihat pada Gambar 3.2.

3.2.2 Preprocessing

Preprocessing merupakan tahap awal dalam pemrosesan data yang bertujuan untuk menyiapkan data mentah menjadi bentuk yang bersih, terstruktur, dan siap untuk diolah menggunakan algoritma *data mining*. Pada preprocessing ini akan dilakukan tahapan analisis korelasi pearson, hasil pemilihan dari pearson dan pengecekan penanganan *missing values*.

Tahap korelasi Pearson merupakan proses pengukuran tingkat hubungan antara dua variabel numerik dalam suatu dataset. Metode ini digunakan untuk mengetahui sejauh mana perubahan pada satu variabel berkaitan dengan perubahan pada variabel lainnya. Nilai koefisien korelasi Pearson berada pada rentang -1 hingga 1, di mana nilai mendekati 1 menunjukkan hubungan positif yang kuat, nilai mendekati -1 menunjukkan hubungan negatif yang kuat, sedangkan nilai mendekati 0 mengindikasikan tidak adanya hubungan yang signifikan antara kedua variabel.

1	Tanggal Kejadian	Waktu Kejadian	Tanggal Dilaporkan	Waktu Dilaporkan	Tingkat Kecelakaan	Jumlah Meninggal Dunia	Jumlah Luka Berat	Jumlah Luka Ringan	Koordinat GPS -Lintang	Koordinat GPS -Bujur	Titik Acuan / Referensi	Jarak Ke Lokasi Kecelakaan	Arah Dari Titik Acuan Ke Lokasi Kecelakaan	Tipe Kecelakaan	Kondisi Cahaya	Cuaca
2	2020-01-01	14:20:00	2020-01-02	10:00:00	Berat	1	0	0	-6,334548	107,33368	Kantor/Perkantoran	500	Timur (90°)	Tabrakan samping Terang/Jelas	Berawan	
3	2020-01-01	03:00:00	2020-01-01	06:00:00	Berat	1	0	1	-6,253458	107,578894	Rumah/Perumahan/Pemukiman	23000	Utara (0°)	Tabrakan depan - Terang/Jelas	Cerah	
4	2020-01-02	11:00:00	2020-01-03	16:00:00	Ringan	0	0	2	-6,326841	107,322723	Rumah/Perumahan/Pemukiman	3000	Barat (270°)	Di ruas jalan, peja Terang/Jelas	Cerah	
5	2020-01-03	13:15:00	2020-01-03	14:00:00	Berat	1	0	0	-6,349382	107,279612	Jembatan		Barat (270°)	Tabrakan saat ger: Terang/Jelas	Cerah	
6	2020-01-04	05:30:00	2020-01-04	06:00:00	Ringan	0	0	1	-6,396588	107,434671	Rumah/Perumahan/Pemukiman	10000	Timur (90°)	Tabrakan saat ger: Terang/Jelas	Cerah	
7	2020-01-04	19:00:00	2020-01-05	06:00:00	Ringan	0	0	2	-6,26696	107,515431	Rumah/Perumahan/Pemukiman	25000	Utara (0°)	Tabrakan saat me: Terang/Jelas	Hujan/Cerah	
8	2020-01-06	06:30:00	2020-01-06	10:00:00	Ringan	0	0	1	-6,301587	107,303018	Simpang	3000	Barat (270°)	Kendaraan Out of Terang/Jelas	Hujan/Cerah	
9	2020-01-06	13:00:00	2020-01-06	14:00:00	Sedang	0	1	5	-6,251412	107,344937	Rumah/Perumahan/Pemukiman	160000	Utara (0°)	Tabrakan saat bel: Terang/Jelas	Cerah	
10	2020-01-05	16:00:00	2020-01-06	14:00:00	Ringan	0	0	1	-6,088789	107,197016	Rumah/Perumahan/Pemukiman	26000	Utara (0°)	Kendaraan Out of Terang/Jelas	Hujan/Cerah	
11	2020-01-02	16:00:00	2020-01-03	09:00:00	Ringan	0	0	1	-6,122577	107,403928	Rumah/Perumahan/Pemukiman	25000	Utara (0°)	Tabrakan depan - Terang/Jelas	Cerah	
12	2020-01-07	06:45:00	2020-01-07	08:30:00	Ringan	0	0	1	-6,348669	107,25992		5000	Barat (270°)	Tabrakan depan - Terang/Jelas	Cerah	
13	2020-01-05	22:30:00	2020-01-05	23:00:00	Ringan	0	0	1	-6,2183	107,294224	Rumah/Perumahan/Pemukiman	25000	Utara (0°)	Tabrakan depan - Terang/Jelas	Cerah	
14	2020-01-07	19:00:00	2020-01-08	08:30:00	Ringan	0	0	1	-6,412875	107,484114	Rumah/Perumahan/Pemukiman	15000	Timur (90°)	Tabrakan samping Terang/Jelas	Cerah	
15	2020-01-04	19:30:00	2020-01-05	10:00:00	Ringan	0	0	1	-6,32978	107,329378	Rumah/Perumahan/Pemukiman	2000	Barat (270°)	Tabrakan depan - Terang/Jelas	Cerah	
16	2020-01-22	14:00:00	2020-02-02	10:00:00	Berat	1	0	1	-6,20496	107,518498	Rumah/Perumahan/Pemukiman	26000	Utara (0°)	Di simpang, tabra Terang/Jelas	Cerah	
17	2020-01-11	00:30:00	2020-01-11	01:30:00	Berat	1	0	0	-6,323108	107,313519	Rumah/Perumahan/Pemukiman	3000	Barat (270°)	Di ruas jalan, peja Terang/Jelas	Cerah	
18	2020-01-11	06:00:00	2020-01-14	08:00:00	Ringan	0	0	1	-6,31953	107,388495	Rumah/Perumahan/Pemukiman	8000	Utara (0°)	Kendaraan Out of Terang/Jelas	Cerah	
19	2020-01-10	18:00:00	2020-01-11	09:30:00	Ringan	0	0	1	-6,072446	107,3442	Rumah/Perumahan/Pemukiman	26000	Utara (0°)	Di ruas jalan, peja Terang/Jelas	Cerah	
20	2020-01-09	15:00:00	2020-01-11	11:00:00	Ringan	0	0	1	-6,120008	107,31993	Rumah/Perumahan/Pemukiman	26000	Utara (0°)	Sepeda angin/ Sej Terang/Jelas	Cerah	
21	2020-01-08	10:00:00	2020-01-09	16:00:00	Berat	1	0	0	-6,106222	107,268544	Rumah/Perumahan/Pemukiman	26000	Utara (0°)	Di ruas jalan, peja Terang/Jelas	Cerah	
22	2020-01-11	23:00:00	2020-01-11	23:30:00	Berat	1	0	0	-6,383323	107,374114	Rumah/Perumahan/Pemukiman	7000	Selatan (180°)	Tabrakan depan - Terang/Jelas	Cerah	
23	2020-01-13	08:45:00	2020-01-13	11:50:00	Berat	1	0	0	-6,350272	107,293725		4000	Barat (270°)	Di ruas jalan, peja Terang/Jelas	Cerah	
24	2020-01-14	06:00:00	2020-01-15	13:00:00	Ringan	0	0	1	-6,312707	107,320883	Rumah/Perumahan/Pemukiman	3000	Barat (270°)	Kendaraan Out of Terang/Jelas	Cerah	
25	2020-01-11	05:30:00	2020-01-13	10:00:00	Berat	1	0	1	-6,333166	107,309097	Rumah/Perumahan/Pemukiman	4000	Selatan (180°)	Tabrakan samping Terang/Jelas	Cerah	
Kecelakaan Menonjol																
	Nomor Jalan	Nama Jalan	Titik Jalan	Fungsi Jalan	Kelas Jalan	Tipe Jalan	Bentuk Geometri	Kondisi Permukaan Jalan	Batas Kecepatan Dilokasi	Kemiringan Jalan	Status Jalan	Penyebab	Perkiraan Nilai Rugi Material Kendaraan	Total Nilai Rugi Material non Kendaraan		
Tidak		Jln. Raya Surotokunto de Arteri	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Nasioni	Manusia			100000		0	
Tidak		Jln. Raya Dsn. Kertasari D Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Basah				60 Datar	Jalan Kota / K	Manusia			500000		0	
Tidak		Jln. Suhut Hidayat Ds. W. Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Kota / K	Manusia			200000		0	
Tidak		Jln. Tol Jakarta-Cikampek Tol	I (Jalan Besar 6/2 B (6 La Lurus	Baik				80 Datar	Jalan Nasioni	Manusia			10000000		0	
Tidak		Jln. Raya Jend. Sudirman Arteri	I (Jalan Besar 4/2 B (4 La Lurus	Baik				60 Datar	Jalan Nasioni	Manusia			100000		0	
Tidak		Jln. Syeh Quro Kp. Kecem Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Basah				60 Datar	Jalan Kota / K	Manusia			10000000		0	
Tidak		Jln. Raya A. Yani Perempé Arteri	I (Jalan Besar 4/2 B (4 La Lurus	Basah				60 Datar	Jalan Nasioni	Manusia			100000		0	
Tidak		Jalan Raya Kp. Rawa cula Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Basah				20 Datar	Jalan Kota / K	Manusia			10000000		0	
Tidak		Jln. Raya Dsn Tengah 2 D Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Propinsi	Manusia			100000		0	
Tidak		Jln. Kosambi Batu Kec. Ci Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Kota / K	Manusia			500000		0	
Tidak		Jln. Tol Jakarta-Cikampek Tol	I (Jalan Besar 6/2 B (6 La Lurus	Baik				80 Datar	Jalan Nasioni	Manusia			5000000		0	
Tidak		Jln. Proklamasi Ds. Kalan Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Propinsi	Manusia			1000000		0	
Tidak		Jln. Raya Simpang Jomin Arteri	I (Jalan Besar 4/2 TB (4 L Lurus	Baik				60 Datar	Jalan Nasioni	Manusia			100000		0	
Tidak		Jln. Raya Surontkunto Ds Arteri	I (Jalan Besar 4/2 B (4 La Lurus	Baik				60 Datar	Jalan Nasioni	Manusia			500000		0	
Tidak		Jln. Raya Ds. Mukti Jaya K Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Kota / K	Manusia			500000		0	
Tidak		Jalan Raya Wirasaba Desi Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Kota / K	Manusia			200000		0	
Tidak		Jln. Raya Pasirmulya Ds. I Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Berlubang				60 Datar	Jalan Kota / K	Manusia			1000000		0	
Tidak		Jln. Desa Kertarahayu Ke Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Kota / K	Manusia			0		0	
Tidak		Jln. Dusun Ciagem Ds. Jay Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Kota / K	Manusia			0		0	
Tidak		Jln. Kp. Sawah Ds. Kp Saw Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Kota / Kabupaten							
Tidak		Jln. Raya Kosambi-Curug Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				60 Datar	Jalan Propinsi	Manusia			1000000		0	
Tidak		Jalan Raya Ruas Tol KM 5 Tol	I (Jalan Besar 6/2 B (6 La Lurus	Baik				80 Datar	Jalan Nasioni	Manusia						
Tidak		Jln. Tanggul Irigasi Ds. Ad Kolektor	I (Jalan Besar 2/2 TB (2 L Lurus	Baik				50 Datar	Jalan Kota / K	Manusia			200000		0	
Tidak		Jln. Galuh Mas Depan Pei Kolektor	I (Jalan Besar 4/2 B (4 La Lurus	Baik				60 Datar	Jalan Kota / K	Manusia			2000000		0	

Gambar 3. 2 Data Kecelakaan Lalu Lintas Excel

Tahap pemilihan variabel merupakan tahapan setelah analisis korelasi pearson dalam suatu dataset yang bertujuan untuk menentukan variabel yang paling relevan. Dalam penelitian ini, variabel yang digunakan berjumlah 12 item pada dataset kecelakaan lalu lintas mencakup Tingkat Kecelakaan, Titik Acuan/Referensi, Jarak Ke Lokasi Kecelakaan, Kondisi Cahaya, Cuaca, Kecelakaan Menonjol, Fungsi Jalan, Bentuk Geometri, Kondisi Permukaan Jalan, Batas Kecepatan Dilokasi, Kemiringan Jalan, dan Penyebab. Variabel yang telah dipilih tersebut dinilai cukup untuk mendukung proses pencarian pola keterkaitan antar faktor dalam kejadian kecelakaan lalu lintas melalui metode asosiasi.

Tahap pada deteksi dan penanganan *missing values* bertujuan untuk mengidentifikasi keberadaan data yang tidak memiliki nilai pada salah satu atau beberapa variabel dalam dataset. Proses ini dilakukan untuk memastikan kelengkapan data sehingga analisis dapat berjalan dengan baik. Nilai yang kosong akan dihapus agar tidak memengaruhi hasil pemrosesan dan akurasi algoritma yang digunakan.

3.2.3 Transformation

Data Transformation merupakan proses mengubah data menjadi format yang lebih sesuai untuk *data mining*. Proses ini mencakup mengubah format tabular dan encoding data kategorikal.

1. Mengubah Format Tabular

Mengubah format tabular adalah representasi data dalam bentuk **tabel** yang terdiri dari **baris dan kolom**, di mana setiap baris mewakili satu entitas atau sampel data, sedangkan setiap kolom mewakili atribut atau variabel yang menggambarkan karakteristik entitas tersebut. Format ini memudahkan analisis data, termasuk penerapan algoritma **data mining** seperti **Apriori dan FP-Growth**.

2. Encoding Data Kategorikal

Kolom seperti tanggal kejadian dan batas kecepatan dilokasi perlu diubah menjadi bentuk numerik.

One-Hot Encoding → Menciptakan kolom baru untuk setiap kategori.

Label Encoding → Mengonversi kategori menjadi angka unik.

3.2.4 Data Mining

Proses *data mining* dilakukan dengan dua algoritma yaitu Algoritma FP-Growth dan Apriori. Berikut *pseudocode data mining* yang dilakukan:

1. Algoritma FP-Growth

Algoritma **FP-Growth** digunakan untuk menemukan *frequent itemsets* dalam suatu dataset transaksi tanpa perlu membangkitkan kandidat itemset seperti pada Apriori. Proses ini langsung mencari pola yang sering muncul dalam data yang telah ditransformasikan dan menghasilkan aturan asosiasi berdasarkan nilai *support* dan *confidence*. Berikut *pseudocode* penerapannya:

BEGIN

```
// Penerapan Algoritma FP-Growth untuk Menemukan Frequent Itemsets
```

```
1. TETAPKAN nilai minimum_support (contoh: 0.1).
```

```
2. JALANKAN algoritma FP-Growth pada df_encoded dengan minimum_support yang telah ditentukan.
```

```
- Fungsi: fpgrowth(data=df_encoded, min_support=minimum_support, use_colnames=True)
```

```
3. SIMPAN hasilnya, yang merupakan daftar itemset yang sering muncul (frequent itemsets), ke dalam DataFrame frequent_itemsets_fp.
```

END

Fungsi:

1. Fungsi `fpgrowth`. Fungsi ini berasal dari library `mlxtend`. Fungsi ini digunakan untuk menemukan himpunan item (itemset) yang sering muncul dalam sebuah dataset transaksi. Algoritma ini seringkali lebih efisien daripada Apriori, terutama pada dataset yang besar, karena tidak perlu menghasilkan kandidat itemset.
2. Fungsi `data=df_encoded`. Parameter ini merujuk pada dataset yang akan dianalisis. `df_encoded` adalah sebuah DataFrame yang sudah dalam format *one-hot encoded*. Dalam format ini, setiap kolom mewakili satu item, dan baris-barisnya berisi nilai `True` (atau 1) jika item tersebut ada dalam transaksi dan `False` (atau 0) jika tidak.

3. Fungsi `min_support=minimum_support`. Ini ambang batas yang menentukan kelayakan sebuah itemset untuk dianggap "sering muncul" (frequent). Nilai `minimum_support` (misalnya 0.1) menandakan bahwa sebuah kombinasi item harus ada dalam setidaknya 10% dari total transaksi agar bisa dimasukkan ke dalam hasil.
4. Fungsi `use_colnames=True`. Parameter ini berfungsi untuk menampilkan hasil menggunakan nama kolom (nama item) dari DataFrame, bukan indeks numeriknya. Penggunaan `True` membuat output menjadi lebih mudah dibaca dan dipahami oleh pengguna.

2. Algoritma Apriori

Algoritma **Apriori** digunakan untuk menemukan *frequent itemsets* dalam suatu dataset transaksi dengan menghitung support dan confidence. Algoritma ini bekerja dengan membentuk kandidat itemset, mengevaluasi frekuensinya, dan menghasilkan aturan asosiasi yang relevan. Berikut *pseudocode* untuk penerapannya:

```
BEGIN
    // Penerapan Algoritma Apriori untuk Menemukan Frequent
    Itemsets
        1. TETAPKAN nilai minimum_support (contoh: 0.1).
        2. JALANKAN algoritma Apriori pada df_encoded dengan
           minimum_support yang telah ditentukan.
           - Fungsi:      apriori(data=df_encoded,
           min_support=minimum_support, use_colnames=True)
        3. SIMPAN hasilnya, yang merupakan daftar itemset yang
           sering muncul (frequent itemsets), ke dalam DataFrame
           frequent_itemsets_apr.
END
```

Fungsi:

1. Fungsi `apriori`. Fungsi ini berasal dari library `mlxtend` dan bertugas untuk menganalisis data transaksi untuk menemukan "itemset" (kombinasi item) yang paling sering muncul bersamaan.

2. Fungsi `data=df_encoded`. Parameter ini menentukan dataset yang akan dianalisis. `df_encoded` adalah DataFrame yang sudah diubah ke dalam format *one-hot encoding*, di mana setiap kolom mewakili satu item dan nilainya adalah `True` (1) jika item tersebut ada dalam transaksi, dan `False` (0) jika tidak.
3. Fungsi `min_support=minimum_support`. Ini ambang batas atau nilai minimum yang menentukan seberapa sering sebuah itemset harus muncul agar dianggap "sering" (frequent). Nilai `minimum_support` (contoh: 0.1) berarti itemset tersebut harus muncul dalam setidaknya 10% dari total transaksi.
4. Fungsi `use_colnames=True`. Parameter ini memastikan bahwa hasil yang ditampilkan menggunakan nama kolom (nama item) yang sebenarnya dari DataFrame, bukan hanya indeks kolom. Ini membuat output lebih mudah dibaca dan diinterpretasikan.

3.2.5 Evaluasi

Tahap evaluasi ini bertujuan untuk memvisualisasikan hasil akhir dari aturan asosiasi yang diperoleh menggunakan algoritma FP-Growth dan **Apriori**. Proses evaluasi dilakukan dengan mengukur **kecepatan waktu komputasi dan jumlah rule** dari masing-masing algoritma, yang didasarkan pada nilai **support, confidence dan lift ratio** yang dihasilkan. Berikut penjelasan dari masing masing evaluasi yang digunakan

1. Nilai *Support*

Support digunakan untuk menentukan frekuensi kemunculan suatu pola dalam dataset transaksi. Semakin besar nilai support, semakin kuat pola tersebut dalam membentuk aturan asosiasi.

Interpretasi: Itemset dengan **support tinggi** menunjukkan bahwa kombinasi tersebut sering muncul dalam transaksi.

2. Nilai *Confidence*

Confidence mengukur tingkat keyakinan dalam suatu aturan asosiasi. Nilai confidence yang tinggi menunjukkan bahwa jika suatu item muncul, maka

kemungkinan besar item yang berkaitan juga akan muncul dalam transaksi tersebut.

Interpretasi: Confidence yang tinggi menunjukkan **hubungan erat** antara dua item dalam aturan asosiasi.

3. *Lift Ratio*

Lift ratio berfungsi untuk menilai kekuatan hubungan antara item dalam aturan asosiasi. Jika nilai lift lebih dari satu, maka terdapat keterkaitan yang kuat antara item, sedangkan nilai lift di bawah satu menunjukkan hubungan yang lemah atau tidak signifikan.

Interpretasi:

Lift > 1 → Item memiliki hubungan yang lebih erat dibandingkan kemunculan acaknya.

Lift = 1 → Item muncul secara independen tanpa keterkaitan signifikan.

Lift < 1 → Hubungan antar item menunjukkan pola yang lemah atau bahkan negatif.

4. Mengukur Kecepatan Waktu Komputasi

Perbandingan waktu eksekusi antara algoritma Apriori dan FP-Growth digunakan untuk mengevaluasi efisiensi kinerja masing-masing metode dalam mengolah dataset. Algoritma yang memiliki waktu pemrosesan lebih cepat dianggap lebih efektif dalam menangani data dalam jumlah besar.

Interpretasi: Algoritma dengan waktu pemrosesan yang lebih cepat dianggap lebih efisien, terutama saat digunakan pada dataset berskala besar.

5. Menghitung Jumlah Rule

Jumlah aturan yang dihasilkan mencerminkan banyaknya pola yang ditemukan oleh algoritma. Jika jumlah aturan terlalu sedikit, kemungkinan besar parameter support dan confidence yang digunakan terlalu tinggi. Sebaliknya, jika terlalu banyak, dapat mengakibatkan informasi yang kurang spesifik dan sulit untuk dianalisis.