
BAB II**TINJAUAN PUSTAKA DAN LANDASAN TEORI****2.1. Tinjauan Pustaka**

Penelitian ini meninjau beberapa pustaka sebagai referensi atau acuan yang telah terangkum sebagai berikut.

Penelitian dengan judul “Penerapan Metode *Asosiasi* Data *Mining* Menggunakan Algoritma Apriori Untuk Mengetahui Kombinasi Antar *Itemset* Pada Pondok Kopi”. yang bertujuan untuk menentukan aturan *asosiasi* menggunakan metode *Market Basket Analysis* dan algoritma Apriori berdasarkan data transaksi untuk mendapatkan informasi sebagai langkah pengambilan keputusan dalam memprediksi barang untuk waktu yang akan datang yang dilakukan oleh Fitri Nurchalifatun di Fakultas Ilmu Komputer UDINUS. Hasil dari penelitian ini yaitu pertimbangan untuk menentukan keputusan akhir dalam mengendalikan barang bahan dan memberi pertimbangan dalam menyediakan barang bahan untuk penjualan selanjutnya. Obyek penelitian tersebut dilakukan di Kafe Pondok Kopi Semarang. Kesimpulan penelitian ini adalah dengan penerapan algoritma Apriori dalam menentukan kombinasi antar *itemset* dan ditemukan 7 aturan asosiasi. Dari penelitian yang telah dilakukan terdapat kekurangan yaitu jumlah data serta penggunaan *confidence* dan *support* yang bervariasi agar diperoleh hasil aturan asosiasi yang efisien. ^[5]

Penelitian lainnya dengan judul “Aplikasi Data *Mining* Analisis Data Transaksi Penjualan Obat Menggunakan Algoritma Apriori” menganalisis keranjang pasar data transaksi penjualan obat dengan menggunakan data *mining* sebagai suatu teknik analisis data yang dapat membantu apotek memperoleh pengetahuan berupa pola – pola penjualan dalam periode bulan tertentu. Aplikasi data *mining* dibangun menggunakan proses sekuensial linier dengan bahasa pemrograman *PHP* dan basis data *MySQL*. Algoritma yang digunakan sebagai proses utama dari analisis keranjang pasar adalah algoritma apriori dengan menggunakan parameter *minimum support*, *minimum confidence* dan periode bulan transaksi penjualan untuk menemukan aturan asosiasi. Penelitian ini menghasilkan aplikasi data *mining* dengan aturan asosiasi antar *item* pada bulan Februari 2012 yaitu konsumen melakukan transaksi pembelian obat jenis obat darah dan *analgesic* secara bersamaan dengan *support* sebesar 2,08%

dan *confidence* sebesar 45,45%. Dengan demikian, jika terdapat seorang konsumen membeli jenis obat darah maka kemungkinan terdapat 45,45% konsumen membeli *analgesic*. Dari penelitian yang telah dilaksanakan terdapat saran pengembangan yaitu agar sistem melakukan data *mining* terhadap atribut selain data jenis obat. [6]

Penelitian selanjutnya yang berjudul “Penerapan Data *Mining* dengan Algoritma *FP – Growth* untuk Mendukung Strategi Promosi Pendidikan” dilakukan oleh Ali Ikhwan. Penelitian ini bertujuan untuk menganalisa permasalahan promosi pendidikan di STMIK Triguna Dharma serta menerapkan algoritma *FP – Growth* dalam strategi promosi pendidikan. Penerapan algoritma *FP – Growth* dalam kasus ini menggunakan *tools Rapid Miner* untuk menemukan *rule* sehingga ditemukan strategi promosi yang tepat. Manfaat penelitian tersebut yaitu mengetahui algoritma *FP – Growth* dapat memberikan alternatif pengetahuan tentang menemukan strategi dalam promosi pendidikan, menukan strategi atau sasaran yang lebih tepat dalam promosi pendidikan, serta menjadikan *software* yang dihasilkan untuk dapat digunakan pihak kampus untuk menentukan strategi dalam promosi pendidikan. Saran dalam penelitian ini agar menjadi penelitian yang lebih baik antara lain data yang digunakan untuk *mining* diperlukan data yang lebih banyak dan lebih lengkap, dan untuk penerapan hasil maka dikonsultasikan dengan pihak pada objek penelitian. [7]

Pada penelitian dengan judul “Perbandingan Algoritma Apriori dan Algoritma *FP – Growth* Untuk Rekomendasi Pada Transaksi Peminjaman Buku di Perpustakaan Universitas Dian Nuswantoro” yang dilakukan oleh Rizky Mei Anggraeni, dibuat sebuah sistem di mana nantinya dapat dijadikan sistem rekomendasi menggunakan dua metode data *mining Association Rule mining* dengan membandingkan algoritma Apriori dan *FP-Growth (frequent pattern growth)*. Hal tersebut bertujuan untuk menentukan buku-buku yang sering dipinjam bersamaan pada transaksi peminjaman buku sebelumnya. Dari permasalahan di atas mengenai data transaksi peminjaman buku, algoritma Apriori dan *Frequent Pattern Growth* dapat digunakan dalam rekomendasi buku perpustakaan dengan teknik data *mining* menggunakan aturan asosiasi untuk menentukan buku-buku yang sering dipinjam dalam waktu bersamaan. Hasil analisis kelayakan bahwa algoritma Apriori dan *FP-Growth* membentuk aturan Asosiasi dari kedua algoritma yaitu sama.

Perbedaan terletak pada waktu eksekusi program dimana dengan menggunakan Apriori waktu yang digunakan untuk eksekusi program lebih banyak dibandingkan waktu eksekusi program menggunakan *FP - Growth*. Dari hasil kinerja kedua algoritma semakin besar *minimum supportnya* maka *itemsets* dan *Association Rule* yang terbentuk tidak ada, maka dari itu dicari *minimum support* yang paling mendekati agar *frequent itemsets* dan *Association Rule* yang terbentuk sempurna atau mendekati sempurna, dan jumlah data buku yang dipinjam ditingkatkan agar pola *mining* yang terbentuk semakin bervariasi. Maka dapat disimpulkan bahwa kedua algoritma tersebut layak diimplementasikan untuk merekomendasi peminjaman buku berdasarkan aturan asosiasi. [8]

Penelitian dengan judul “Penerapan Data Mining Untuk Mengetahui Pola Asosiasi Antara Data Mahasiswa dan Tingkat Kelulusan Menggunakan Algoritma *Fold-Growth*” melakukan pencarian pola asosiasi antara data mahasiswa dan data tingkat kelulusan dilakukan berdasarkan jalur masuk dan program studi dengan menerapkan algoritma *Fold – Growth*. Sistem akan mencari aturan asosiasi dengan tahap awal adalah mencari *frequent itemset*. Pada tahap akhir, *association rule* akan diuji dalam 3 jenis, yaitu uji pengaruh nilai *minimum support* dan *confidence* terhadap jumlah *rule*, uji *lift ratio* dan uji akurasi *Association Rule*. Perbandingan antara *minimum support* dan *minimum confidence* dengan jumlah *rule* berbanding terbalik pada program studi Ilmu Komputer. Hasil pengujian akurasi *association rule* menunjukkan tingginya prosentase *Association Rule* yang diterapkan pada data uji, sehingga dapat digunakan untuk memprediksi tingkat kelulusan mahasiswa di waktu yang akan datang. [9]

Tabel 2.1. Perbandingan Referensi

Referensi	Penelitian 1 [5]	Penelitian 2 [6]	Penelitian 3 [7]	Penelitian 4 [8]	Penelitian 5 [9]
Objek Penelitian	Café Pondok Kopi	Apotek Setya Sehat	STMIK Tri Guna Dharma	Perpustakaan UDINUS	Universitas Brawijaya
Data yang Digunakan	Transaksi Jual Beli	Transaksi Jual Beli	Data Sekolah Menengah Atas/Kejuruan	Transaksi Peminjaman Buku	Data Mahasiswa dan Data Kelulusan.

Metode yang Diterapkan	Metode Asosiasi	Metode Asosiasi	Metode Asosiasi	Metode Asosiasi	Metode Asosiasi
Algoritma yang Digunakan	Apriori	Apriori	<i>FP – Growth</i>	<i>FP – Growth</i> dan Apriori	<i>Fold - Growth</i>
Kota Penerapan	Semarang	Semarang	Padang	Semarang	Malang
Minimum Support	5%	1.7%	3%	30%	20%
Minimum Confidence	30%	10%	11%	50%	50%
Lift Ratio	11.706	0.625	1.1494	23.8	0.531

Perbedaan penelitian sebelumnya dengan penelitian yang akan dilaksanakan yaitu:

- Lokasi penelitian yang akan diterapkan yaitu di Kota Purwokerto.
- Data yang digunakan adalah data transaksi penjualan produk selama 5 tahun.
- Objek penelitian ini dilaksanakan di perusahaan dagang.
- Algoritma yang digunakan untuk mendapatkan *association rule* yaitu algoritma *FP – Growth*.

2.2. Landasan Teori

2.2.1. Data Mining

Menurut Turban dalam bukunya yang berjudul “*Decision Support System and Intelligent System*”, *Data Mining* adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam basis data. *Data Mining* adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai basis data besar. ^[10]

2.2.2. Metode Asosiasi (*Association Rule*)

Association Rule merupakan suatu proses pada data mining untuk menentukan semua aturan asosiatif yang memenuhi syarat *minimum* untuk *support* (*minsup*) dan

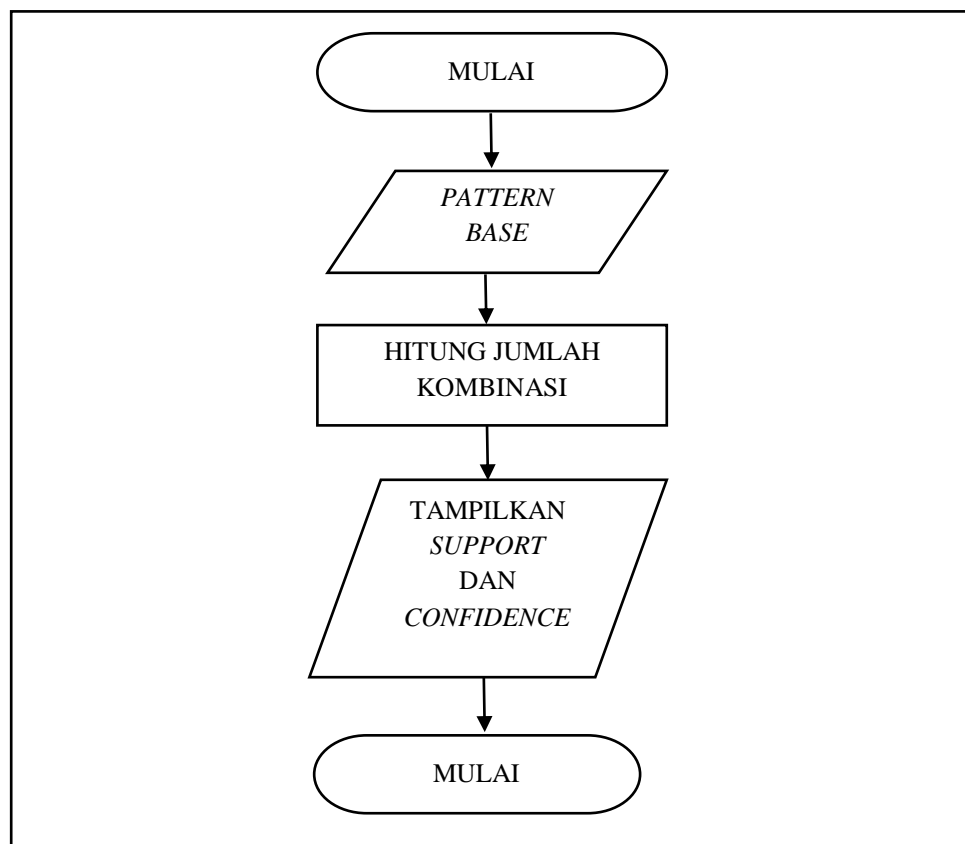
confidence (*minconf*) pada sebuah *database*. Kedua syarat tersebut akan digunakan untuk *interesting Association Rules* dengan dibandingkan dengan batasan yang telah ditentukan, yaitu *minsup* dan *minconf*. *Association Rule Mining* adalah suatu prosedur untuk mencari hubungan antar *item* dalam suatu *dataset*. Dimulai dengan mencari *frequent itemset*, yaitu kombinasi yang paling sering terjadi dalam suatu *itemset* dan harus memenuhi *minsup*. *Flowchart association rule* dapat dilihat pada Gambar 2.1 dan untuk mendapatkan nilai *support* dari suatu *item A* dapat diperoleh dengan rumus berikut: ^[11]

$$\text{Support (A)} = \frac{\text{Jumlah Transaksi Mengandung A}}{\text{Total Transaksi}} \dots\dots\dots (2.1)$$

$$\text{Support (A, B)} = P (A, B) = \frac{\text{Jumlah Transaksi Mengandung A dan B}}{\text{Total Transaksi}} \dots\dots\dots (2.2)$$

Setelah semua *frequent item* dan *large itemset* didapatkan, dapat dicari syarat *minimum confidence* (*mincof*) dengan menggunakan rumus berikut:

$$\text{Confidence (A} \rightarrow \text{B)} = P(A|B) = \frac{\text{Jumlah Transaksi Mengandung A dan B}}{\text{Jumlah Transaksi Mengandung A}} \dots\dots\dots (2.3)$$



Gambar 2.1. Flowchart Association Rule

2.2.3. Algoritma *FP – Growth*

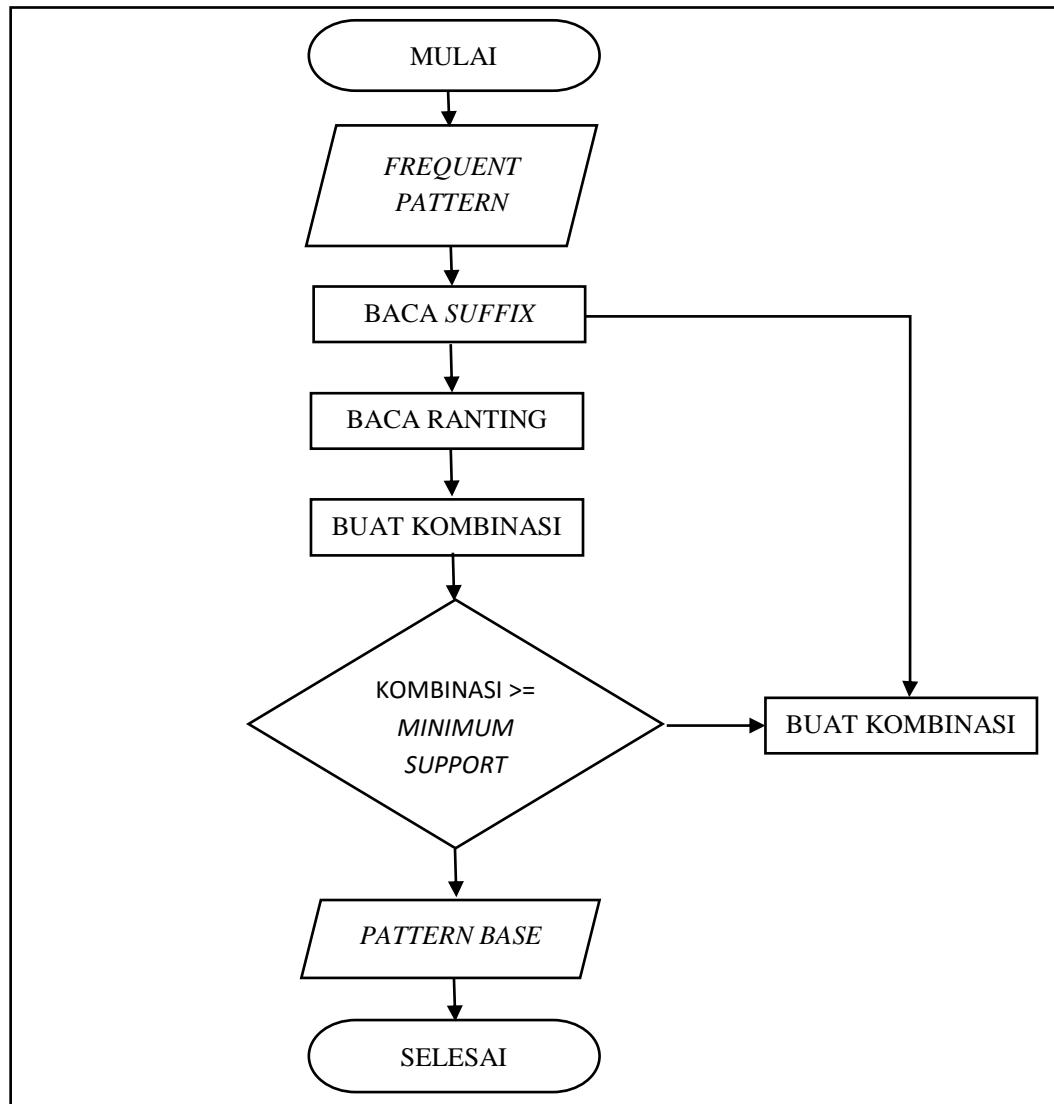
FP-Growth adalah salah satu alternatif algoritma yang dapat digunakan untuk menentukan himpunan data yang paling sering muncul (*frequent itemset*) dalam sebuah kumpulan data. Pada penentuan *frequent itemset* terdapat 2 tahap proses yang dilakukan yaitu pembuatan *FP-Tree* dan penerapan algoritma *FP-Growth* untuk menemukan *frequent itemset*. Struktur data yang digunakan untuk mencari *frequent itemset* dengan algoritma *FP-Growth* adalah perluasan dari penggunaan sebuah pohon *prefix*, yang biasa disebut adalah *FP-Tree*. Dengan menggunakan *FP-Tree*, algoritma *FP-Growth* dapat langsung mengekstrak *frequent itemset* dari *FP-Tree* yang telah terbentuk dengan menggunakan prinsip *divide and conquer*.

Pada penentuan *frequent itemset* terdapat 2 tahap proses yang dilakukan yaitu pembuatan *FP-Tree* dan penerapan algoritma *FP-Growth* untuk menemukan *frequent itemset*. Struktur data yang digunakan untuk mencari *frequent itemset* dengan algoritma *FP-Growth* adalah perluasan dari penggunaan sebuah pohon *prefix*, yang biasa disebut adalah *FP-Tree*. Dengan menggunakan *FP-Tree*, algoritma *FP-Growth* dapat langsung mengekstrak *frequent itemset* dari *FP-Tree* yang telah terbentuk dengan menggunakan prinsip *divide and conquer*.^[12]

Setelah tahap pembangunan *FP-Tree* dari sekumpulan data transaksi, maka terapkan algoritma *FP-Growth* untuk mencari *frequent itemset* yang signifikan. Algoritma *FP-Growth* dibagi menjadi tiga langkah utama, yaitu:

- a. Tahap Pembangkitan *Conditional pattern base Conditional pattern base* merupakan *subdatabase* yang berisi *prefix path* (lintasan *prefix*) dan *suffix pattern* (pola akhiran). Pembangkitan *Conditional pattern base* didapatkan melalui *FP-Tree* yang telah dibangun sebelumnya.
- b. Tahap Pembangkitan *Conditional 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 sama dengan *minimum support count* ξ akan dibangkitkan dengan *conditional FP - Tree*.
- c. Tahap Pencarian *frequent itemset* Apabila *Conditional FP-Tree* merupakan lintasan tunggal (*single path*), maka didapatkan *frequent itemset* dengan melakukan kombinasi *item* untuk setiap *conditional FP-Tree*. Jika bukan lintasan tunggal, maka dilakukan pembangkitan *FP-Growth* secara rekursif.^[12]

Proses algoritma FP – Growth dapat digambarkan dengan flowchart pada Gambar 2.2.



Gambar 2.2. Flowchart FP – Growth

2.2.4. FP – Tree

FP-Tree merupakan struktur penyimpanan data yang dimampatkan. [12] *FP-Tree* dibangun dengan memetakan setiap data transaksi ke dalam setiap lintasan tertentu dalam *FP-Tree*. Karena dalam setiap transaksi yang dipetakan, mungkin ada transaksi yang memiliki *item* yang sama, maka lintasannya memungkinkan untuk saling menimpa. Semakin banyak data transaksi yang memiliki *item* yang sama, maka proses pemampatan dengan struktur data *FP-Tree* semakin efektif. Adapun *FP-Tree* adalah sebuah pohon dengan definisi sebagai berikut:

- a. *FP-Tree* dibentuk oleh sebuah akar yang diberi label null, sekumpulan sub-*tree* yang beranggotakan *item-item* tertentu, dan sebuah tabel *frequent header*.
- b. Setiap simpul dalam *FP-Tree* mengandung tiga informasi penting, yaitu label *item*, menginformasikan jenis *item* yang direpresentasikan simpul tersebut, *support count*, merepresentasikan jumlah lintasan transaksi yang melalui simpul tersebut, dan *pointer* penghubung yang menghubungkan simpul-simpul dengan label *item* sama antar-lintasan, ditandai dengan garis panah putus-putus. ^[12]