

Rekomendasi Paket Produk Guna Meningkatkan Penjualan Dengan Metode FP-Growth

Asrul Abdullah*

Teknik Informatika

Universitas Muhammadiyah Pontianak

Pontianak

*asrul.abdullah@unmuhpnk.ac.id

Abstrak-Setiap pedagang atau pemilik tempat usaha harus dapat menemukan strategi agar tempat usahanya mampu bertahan dan terus berkembang. Keinginan konsumen dalam membeli suatu produk memang sangat beragam. Ada pembeli yang gemar membeli produk yang telah dipaket, tetapi ada juga pembeli yang membeli produk yang mendapat diskon, dan masih banyak lagi. Keragaman perilaku konsumen dalam memilih barang yang ingin dibeli tentunya menjadi permasalahan tersendiri bagi para penjual jika ingin meningkatkan penjualannya. Algoritma FP-Growth sendiri merupakan hasil perbaikan dari algoritma Apriori. FP-Growth membangkitkan struktur data *tree* atau yang lebih dikenal sebagai *frequent pattern tree*. FP *tree* memerlukan dua kali *scanning database* untuk menentukan *frequent itemset* sehingga membuatnya lebih efisien dibandingkan Apriori. Hasil dari *association rules* menggunakan FP-Growth dijadikan rekomendasi bagi para penjual/*retailer* dalam memberikan paket penjualan barang bagi konsumennya dan hasil dari penelitian ini adalah ditemukan dua pasangan item barang yakni kopi, gula dan teh, susu yang memiliki *support* sebesar 30% dan *confidence* sebesar 70%.

Kata Kunci: FP-Growth; frequent itemset; association rules

1. Pendahuluan

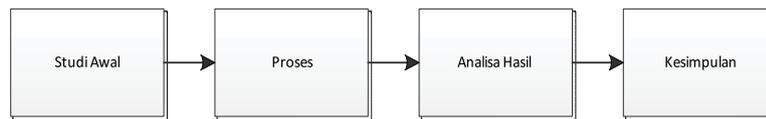
Perkembangan pasar maya (*online*) dan pasar tradisional membuat persaingan dalam menarik konsumen menjadi semakin ketat. Setiap pedagang atau pemilik usaha harus dapat menemukan strategi agar tempat usahanya mampu bertahan dan terus berkembang. Salah satu strategi yang bisa diterapkan adalah pedagang harus dapat mengetahui keinginan konsumen atau pedagang dapat memberikan pilihan produk yang sesuai dengan kebutuhan konsumen [1]. Kemampuan pedagang dalam melihat dua hal tersebut yakni keinginan dan kebutuhan konsumen pastinya berimbas pada peningkatan penjualan dari sebuah produk. Keinginan dan kebutuhan konsumen dalam membeli suatu produk memang sangat beragam. Sebagai contoh, saat kita membeli produk di toko *retail*, pastinya calon pembeli ditawarkan beragam produk mulai dari produk paketan, produk pelengkap, dan produk yang mendapat diskon. Berbagai cara menawarkan produk akan memicu calon pembeli untuk membeli produk tersebut. Ada pembeli yang gemar membeli produk yang telah dipaket, tetapi ada juga pembeli yang membeli produk yang mendapat diskon, dan masih banyak lagi. Keragaman keinginan konsumen dalam memilih barang yang ingin dibeli tentunya menjadi permasalahan tersendiri bagi para penjual jika ingin meningkatkan penjualannya.

Masalah lainnya berupa kesulitan para penjual/penentu keputusan untuk memilih paket produk yang akan dipasarkan sesuai dengan keragaman konsumen. Data penjualan yang berupa transaksi barang yang telah dibeli dan keinginan konsumen yang besar, harusnya

dapat diolah agar menghasilkan sebuah informasi yang berguna sebagai dasar untuk menentukan strategi apa yang cocok agar penjualan semakin meningkat. Menariknya, kehadiran teknologi informasi seakan menjadi alat bantu yang cocok dan tepat dalam menentukan keputusan bagi para pembuat keputusan ataupun penjual. Dengan menggunakan pendekatan/metode yang cocok diharapkan dapat menjadi jalan keluar bagi penjual dalam mengatasi permasalahan tersebut. Metode yang dapat diterapkan di dalam penelitian ini adalah *market basket analysis*. *Market basket analysis* merupakan salah satu metode di dalam *data mining* yang menentukan produk-produk yang dibeli konsumen secara bersamaan berdasarkan dari data transaksi konsumen tersebut [2].

Analisis keranjang pasar telah banyak digunakan di perusahaan sebagai alat untuk menemukan asosiasi produk. Produk-produk yang memiliki hubungan bisa disatukan dalam paket penjualan. Beberapa aspek yang dipelajari di *market basket analysis* antara lain profil minat pelanggan dengan minat produk untuk pemasaran[3], pola pembelian di lingkungan toko yang menjual berbagai macam *item* barang untuk meningkatkan penjualan[4]. Di dalam *market basket analysis*, poin penting yang menjadi acuan para pengambil keputusan adalah *association rules*. *Association rules* diturunkan dari *frequent itemset* menggunakan *support* dan *confidence* sebagai *threshold levels*. Aturan asosiasi yang dihasilkan sangat tergantung pada nilai *confidence* dari sebuah *itemset*.

Beberapa algoritma yang menggunakan *frequent itemset* antara lain Apriori[5], FP-Growth[6], Eclat[7] dan K-Apriori[8]. Khusus di dalam penelitian ini menggunakan



Gambar 1. Tahapan penelitian

algoritma FP-Growth. Pemilihan algoritma FP-Growth mengacu pada penelitian sebelumnya yang dilakukan oleh Widiastuti dan Sofi [9]. Mereka membandingkan dua buah algoritma yakni Apriori dan FP-Growth untuk mengukur waktu proses pada 2500 data transaksi. Hasilnya algoritma FP-Growth memerlukan waktu 1 jam 20 menit sedangkan Apriori memerlukan waktu 3 jam 12 menit dengan minimum *support* 40%. Beberapa penelitian serupa dengan penelitian ini seperti yang dilakukan oleh Nugroho, dkk [10] tentang rekomendasi pemilihan SMA menggunakan metode *naive bayes*. Hasil dari penelitian mereka adalah berupa daftar sekolah yang diurutkan berdasarkan probabilitas tertinggi dari hasil perhitungan dan prioritas favorit dari masing-masing sekolah. Penelitian yang dilakukan Virrayani dan Sutikno [11] tentang prediksi penjualan barang menggunakan metode *Adaptive Neuro-Fuzzy Inference System* (ANFIS). Hasil dari penelitian mereka yakni nilai MAPE (*Mean Absolute Percentage Error*) untuk pengujian seluruh produk di bawah batas toleransi *error* yakni 20%. Artinya metode ANFIS cukup baik digunakan untuk memprediksi penjualan barang.

Sementara itu, tujuan dari penelitian ini adalah memberikan rekomendasi bagi para penjual/*retailer* dalam memberikan paket penjualan barang bagi konsumennya, sedangkan manfaat dari penelitian ini adalah membantu para penjual untuk memilih strategi yang tepat dalam pemasaran produk sehingga berdampak pada peningkatan penjualan.

2. Metode Penelitian

Adapun tahapan atau proses dari penelitian ini dapat dilihat pada Gambar 1. Pada Gambar 1 dapat dikelompokkan dalam empat kategori besar, yakni:

1. Studi Awal

Pokok permasalahan di dalam penelitian ini adalah kesulitan penjual untuk menentukan paket produk yang akan ditawarkan kepada konsumen. Data yang digunakan adalah data transaksi penjualan di minimarket XYZ.

2. Proses

Sebelum dilakukan proses perhitungan dengan menggunakan metode FP-Growth, data transaksi penjualan di minimarket XYZ harus dilakukan pra-proses berupa mengisi *missing value*. Kemudian dilakukan transformasi data. Hasil dari transformasi data dapat dilihat pada Tabel 1.

Pada Tabel 1, terdapat 15 transaksi penjualan barang kebutuhan pokok di minimarket XYZ. Dari 15 transaksi tersebut, ada 3 transaksi dari konsumen yang hanya membeli 1 produk saja. Selanjutnya, data transaksi tersebut akan dilakukan *association rule mining*.

]

Tabel 1. Data transaksi minimarket XYZ

| Kode Transaksi | Nama Barang |
|----------------|---|
| TRX001 | Minyak Goreng, Gula, Kopi, Teh, Susu |
| TRX002 | Air Mineral, Susu, <i>Snack</i> |
| TRX003 | Buku, Pensil |
| TRX004 | Permen, Gula |
| TRX005 | Pensil |
| TRX006 | <i>Snack</i> , Air Mineral |
| TRX007 | Gula, Kopi |
| TRX008 | Susu, Teh |
| TRX009 | Gula, Buku, Pulpen |
| TRX010 | Air Mineral, Minyak Goreng, Kopi |
| TRX011 | <i>Snack</i> , Gula, Kopi, Buku, Pulpen |
| TRX012 | Buku |
| TRX013 | Permen, <i>Snack</i> |
| TRX014 | Kopi, Teh, Susu |
| TRX015 | Susu |

a. Association Rule Mining

Analisis asosiasi atau *association rule mining* adalah teknik *data mining* untuk menemukan aturan asosiatif antara suatu kombinasi item. Analisis asosiasi dikenal juga sebagai salah satu teknik *data mining* yang menjadi dasar dari salah satu teknik *data mining* lainnya. Secara khusus, salah satu tahap analisis asosiasi yang menarik perhatian banyak peneliti untuk menghasilkan algoritma yang efisien, yaitu analisis pola frekuensi tinggi (*frequent pattern mining*) [12][7][6][13].

Analisis pola frekuensi tinggi. Tahap ini mencari kombinasi item yang memenuhi syarat minimum dari nilai *support* dalam *database*. Nilai *support* sebuah *item* diperoleh dengan persamaan (1). Sedangkan nilai *support* dari 2 *item* diperoleh dari persamaan (2).

$$\text{support (A)} = \frac{\text{jumlah transaksi mengandung A}}{\text{Total transaksi}} \quad (1)$$

$$\text{support (A} \cap \text{B)} = \frac{\text{jumlah transaksi mengandung A dan B}}{\text{total transaksi}} \quad (2)$$

Pembentukan aturan asosiatif. Setelah semua pola frekuensi tinggi ditemukan, kemudian dicari aturan asosiatif yang memenuhi syarat minimum untuk *confidence* dengan menghitung *confidence* aturan asosiatif $A \rightarrow B$. Nilai *confidence* dari aturan $A \rightarrow B$ diperoleh dari persamaan (3)

$$\text{confidence P(A|B)} = \frac{\text{jumlah transaksi mengandung A dan B}}{\text{jumlah transaksi mengandung A}} \quad (3)$$

Tabel 2. Frekuensi semua item barang untuk semua transaksi

| Kode | Item Barang | Frekuensi |
|------|---------------|-----------|
| 1 | Gula | 5 |
| 2 | Kopi | 5 |
| 3 | Teh | 3 |
| 4 | Susu | 5 |
| 5 | Air Mineral | 3 |
| 6 | Snack | 4 |
| 7 | Buku | 4 |
| 8 | Minyak Goreng | 2 |
| 9 | Permen | 1 |
| 10 | Pulpen | 2 |
| 11 | Pensil | 2 |

Tabel 3. Frekuensi item produk yang mengikuti minimum support count

| Kode | Item Barang | Frekuensi |
|------|-------------|-----------|
| 2 | Gula | 5 |
| 3 | Kopi | 5 |
| 4 | Teh | 3 |
| 5 | Susu | 5 |
| 6 | Air Mineral | 3 |
| 7 | Snack | 4 |
| 8 | Buku | 4 |

Tabel 4. Urutan item produk berdasarkan frekuensi terbesar

| Kode | Item Barang | Frekuensi | Prioritas |
|--------|-------------|-----------|-----------|
| 2 | Gula | 5 | 1 |
| 3 | Kopi | 5 | 2 |
| 5 | Susu | 5 | 3 |
| 7 | Snack | 4 | 4 |
| 8 | Buku | 4 | 5 |
| 4 | Teh | 3 | 6 |
| 6 | Air Mineral | 3 | 7 |
| Jumlah | | 29 | |

3. Hasil dan Pembahasan

Sebelum melihat hasil perhitungan dari sistem yang dibangun dengan menggunakan bahasa pemrograman Java, terlebih dahulu dilakukan perhitungan secara manual. Untuk itu ada beberapa langkah yang harus ditempuh yakni :

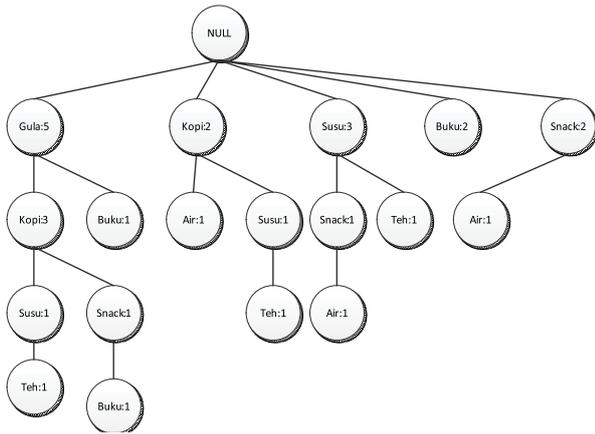
a. Pembuatan Frequent Pattern (FP) Tree

Untuk mencari *frequent itemset* dan *association rule* menggunakan algoritma FP-Growth ada beberapa langkah-langkah sebagai berikut:

- 1) Hitung terlebih dahulu *minimum support*
Asumsi *minimum support* adalah 20%, maka *minimum support count* adalah $(20\% * 15) = 3$, maka *minimum support count* adalah 3. Karena *minimum support count* adalah 3, maka ada item barang yang harus dibuang. Untuk lebih jelasnya dapat dilihat pada Tabel 2. Pada Tabel 2, di baris 8 hingga baris 11, frekuensi item barang dibawah *minimum support count* yang telah ditetapkan yakni 3. Itu berarti, *item* produk minyak goreng, permen, pulpen, dan pensil harus dihilangkan.
- 2) Pembuatan *Header Item*
Pembuatan *header* item mengikuti *minimum support count* yang telah ditentukan di awal. *Minimum support count* yang telah ditentukan adalah 3. Item barang yang memiliki frekuensi < 3, akan dibuang dari *header list*. Untuk lebih jelasnya ditunjukkan pada Tabel 3. Pada Tabel 3 terlihat ada 4 baris yang telah hilang. Yakni baris 1, 9, 10 dan 11. Produk tersebut tidak diikutkan di dalam proses berikutnya. Jadi produk yang ikut pada tahapan berikutnya adalah produk yang memiliki *minimum support count* ≥ 3 .
- 3) Prioritas item yang memiliki frekuensi terbesar
Setelah menghilangkan item produk yang di bawah *minimum support count*, langkah selanjutnya adalah memilih item prioritas yang memiliki frekuensi terbesar seperti yang ditunjukkan pada Tabel 4. Pada Tabel 4 terlihat produk gula, kopi, dan susu yang memiliki frekuensi paling banyak yaitu 5. Produk tersebut menempati urutan 1, 2, dan 3. Frekuensi paling sedikit ada pada produk air mineral yakni 3, sehingga mendapatkan prioritas terendah.
- 4) Susun *item* berdasarkan prioritas
Setelah frekuensi tertinggi dan terendah ditentukan, *item* tadi kembali diurukan berdasarkan prioritas. Di awal proses, data transaksi penjualan berjumlah 15. Hasil *order* (susun) ditunjukkan pada Tabel 5.

Tabel 5. Hasil order item produk berdasarkan prioritas

| Kode Transaksi | Kode Barang | Hasil Order |
|----------------|-------------|-------------|
| TRX001 | 2,3,4,5 | 2,3,4,5 |
| TRX002 | 6,5,7 | 5,7,6 |
| TRX003 | 8 | 8 |
| TRX004 | 2 | 2 |
| TRX006 | 7,6 | 7,6 |
| TRX007 | 2,3 | 2,3 |
| TRX008 | 5,4 | 5,4 |
| TRX009 | 3,8 | 3,8 |
| TRX010 | 6,3 | 3,6 |
| TRX011 | 7,2,3,8 | 2,3,7,8 |
| TRX012 | 8 | 8 |
| TRX013 | 7 | 7 |
| TRX014 | 3,4,5 | 3,5,4 |
| TRX015 | 5 | 5 |



Gambar 2. Pembuatan FP Tree

Tabel 6. Frequent itemset dengan 1 dan 2-itemsets

| Pattern | Item | Min Support |
|-----------|------|-------------|
| Pattern 0 | 6 | 0.2 |
| Pattern 1 | 4 | 0.2 |
| Pattern 2 | 8 | 0.3 |
| Pattern 3 | 7 | 0.3 |
| Pattern 4 | 5 | 0.4 |
| Pattern 5 | 3 | 0.4 |
| Pattern 6 | 2 | 0.4 |
| Pattern 7 | 4 5 | 0.2 |
| Pattern 8 | 2 3 | 0.2 |

Tabel 7. Rule yang dihasilkan

| Pattern | Min Support | Min Confidence |
|---------|-------------|----------------|
| 3 → 2 | 0.3 | 0.6 |
| 2 → 3 | 0.3 | 0.6 |
| 5 → 4 | 0.3 | 0.6 |
| 4 → 5 | 0.3 | 1 |

```
input_fp_growth.txt - Notepad
File Edit Format View Help
1 2 3 4 5
6 5 7
8 10
11 2
10
7 6
2 3
5 4
2 8 9
6 1 3
7 2 3 8 9
8
11 7
3 4 5
5
```

Gambar 3. Item barang diubah menjadi angka

```
public static void main(String [] arg) throws IOException{
    //String input = fileToPath("contextIGB.txt");
    String input = fileToPath("input_fp_growth.txt");
    String output = "://output.txt";

    // By changing the following lines to some other values
    // it is possible to restrict the number of items in the antecedent and
    // consequent of rules
    int maxConsequentLength = 40;
    int maxAntecedentLength = 40;
```

Gambar 4. Baris perintah untuk penyisipan inputan di FP-Growth

```
F HeaderList: [2, 3, 5, 7, 8, 4, 6]
-1 (count=1)
 2 (count=5)
  3 (count=3)
   5 (count=1)
    4 (count=1)
     7 (count=1)
      8 (count=1)
       8 (count=1)
        5 (count=3)
         7 (count=1)
          6 (count=1)
           4 (count=1)
            8 (count=2)
             7 (count=2)
              6 (count=1)
               3 (count=2)
                6 (count=1)
                 5 (count=1)
                  4 (count=1)
```

Gambar 5. Hasil pembentukan FP tree

```
----- FREQUENT ITEMSETS -----
L0
L1
pattern 0: 6 support : 3
pattern 1: 4 support : 3
pattern 2: 8 support : 4
pattern 3: 7 support : 4
pattern 4: 5 support : 5
pattern 5: 3 support : 5
pattern 6: 2 support : 5
L2
pattern 7: 4 5 support : 3
pattern 8: 2 3 support : 3
-----
```

Gambar 6. Hasil pembentukan frequent itemset

| Pattern | #SUP: | #CONF: |
|---------|-------|--------|
| 3 ==> 2 | 3 | 0.6 |
| 2 ==> 3 | 3 | 0.6 |
| 5 ==> 4 | 3 | 0.6 |
| 4 ==> 5 | 3 | 1 |

Gambar 7. Hasil association rule

Pada Tabel 5, jika diperhatikan dengan saksama ada 1 transaksi yang hilang yaitu TRX005. Kode transaksi ini tidak dimasukkan dikarenakan *item* produk dari transaksi ini hanya pensil. Sedangkan produk pensil sendiri memiliki *minimum support count* di bawah 3.

- 5) Bentuk FP tree dari tiap transaksi Tahap selanjutnya adalah membuat FP (*frequent pattern*) tree dari item transaksi. Hasil FP Tree seperti yang ditunjukkan pada Gambar 2. Pada Gambar 2 pohon *frequent pattern* dihasilkan dari semua *item* produk yang ada pada Tabel 5. *Item* produk seperti minyak goreng, pulpen, pensil, dan permen tidak ada di pohon tersebut. Gambar pohon juga tidak *absolute* atau wajib seperti gambar di atas. Yang terpenting adalah panah atau garis dan frekuensinya harus sama dengan dengan yang ada di Tabel 4.

6) Validasi

Setelah pohon FP (*frequent pattern*) terbentuk, kita terlebih dahulu harus melakukan validasi apakah pohon FP yang dibuat cocok dengan frekuensi yang ada pada tabel prioritas *item*. Tabel 4 memiliki jumlah frekuensi sebesar 29. Sedangkan pada Gambar 2, jumlah frekuensi yang ada di pohon FP berjumlah 29.

b. Penerapan Algoritma FP-Growth

Setelah *FP tree* terbentuk dan telah tervalidasi sesuai dengan jumlah frekuensi setiap *item*, maka langkah selanjutnya adalah penerapan algoritma FP-Growth.

- Langkah 1, menentukan *frequent itemset* menggunakan algoritma FP-Growth. Dari sekian banyak *frequent itemset* yang dihasilkan diambil *item* dari pola yang memiliki *minimum support (min support)* ≥ 0.3 atau 30%. *Frequent itemset* yang terbentuk ada 9 yang terbagi atas 1-*itemset* dan 2-*itemset* yang ditunjukkan pada Tabel 6.

$Pattern\ 0 = Support\ (Air\ Mineral) = 3 \div 14 = 0.21$
pembulatan ke atas (*ceiling*) menjadi 0.2.

$Pattern\ 1 = Support\ (Teh) = 3 \div 14 = 0.21$ pembulatan ke atas (*ceiling*) menjadi 0.2.

$Pattern\ 2 = Support\ (Buku) = 4 \div 14 = 0.28$
pembulatan ke atas (*ceiling*) menjadi 0.3.

$Pattern\ 3 = Support\ (Snack) = 4 \div 14 = 0.28$
pembulatan ke atas (*ceiling*) menjadi 0.3.

$Pattern\ 4 = Support\ (Susu) = 5 \div 14 = 0.35$
pembulatan ke atas (*ceiling*) menjadi 0.4.

$Pattern\ 5 = Support\ (Kopi) = 5 \div 14 = 0.35$
pembulatan ke atas (*ceiling*) menjadi 0.4.

$Pattern\ 6 = Support\ (Gula) = 5 \div 14 = 0.35$
pembulatan ke atas (*ceiling*) menjadi 0.4.

$Pattern\ 7 = Support\ (Teh)\ Support\ (Susu) = 3 \div 14 = 0.21$

$Pattern\ 8 = Support\ (Gula) \cap Support\ (Kopi) = 3 \div 14 = 0.21$.

Untuk lebih jelasnya dapat dilihat pada Tabel 6.

Pada Tabel 6 ada 9 buah *pattern* yang dihasilkan dari *frequent itemset*. Namun karena telah ditentukan *minimum support* ≥ 0.3 atau 30%. Maka *pattern* yang tersisa tinggal 9.

- Langkah 2, pembentukan *association rule* dari *frequent itemset* yang telah terbentuk. Dari 9 *frequent itemset*, *association rule* yang dihasilkan ada 4 seperti yang ditunjukkan pada Tabel 7.

Pada Tabel 7 ada 4 *association rule* (aturan) yang terbentuk yakni *item* kopi dan gula dapat menjadi satu paket dengan *minimum support* sebesar 0.3 dan *minimum confidence* sebesar 0.6 begitu juga produk gula dan kopi. Produk lainnya yakni teh dan susu dapat menjadi satu paket dengan *minimum support* sebesar 0.3 dan *minimum confidence* sebesar 0.6.

c. Perhitungan dengan Aplikasi

Aplikasi FP-Growth ini dibangun dengan menggunakan bahasa pemrograman Java. Bahasa ini dipilih karena merupakan bahasa pemrograman berbasis objek. Selain itu, Java juga dapat dijalankan di semua sistem operasi. Langkah-langkah menggunakan aplikasi FP-Growth adalah sebagai berikut:

- Ubah nama *item* barang menjadi deretan angka, misalnya gula diberi kode 2, dan seterusnya seperti

yang ditunjukkan pada Gambar 3.

Pada Gambar 3, aplikasi yang dibuat tidak dapat mengolah data berupa *string* atau kalimat. Jika setiap produk harus dikonversikan ke dalam nilai *integer* (bilangan bulat). Misalnya *item* gula diberi nilai setara dengan 2.

- Inputan *item* barang disimpan dalam ekstensi *.txt*. File berekstensi *.txt* kemudian disimpan di dalam *folder* yang digabung dalam satu *package* dengan aplikasi FP-Growth seperti yang ditunjukkan pada Gambar 4. Pada Gambar 4 ada sejumlah baris perintah di bahasa pemrograman Java untuk membuka file berekstensi *.txt*. Di baris berikutnya baris perintah untuk menampilkan *output* dari hasil perhitungan di algoritma FP-Growth.

- Hasil dari perhitungan algoritma FP-Growth akan menghasilkan pohon *frequent pattern*, *frequent itemset*, dan *association rule*.

- FP *tree* untuk dataset ini ditunjukkan pada Gambar 5. Pada Gambar 5 terlihat *header list* yang menampilkan *item* barang yang sudah mengikuti *minimum support count* $\geq 30\%$. Baris berikut terlihat ada indentasi atau baris menjorok ke dalam. Baris yang menjorok tersebut mewakili anak (*child*) dari *parent* di atasnya. Jika digambar akan persis sama seperti yang ditunjukkan pada Gambar 2.

- Frequent itemset* ditunjukkan pada Gambar 6.

Pada Gambar 6 yang merupakan hasil *frequent itemset* yang dihasilkan dari aplikasi yang telah dibuat. *Frequent itemset* juga dibagi dalam 2 yakni *frequent itemset* 1 *item* dan 2 *item*. Jika melihat dari Tabel 6 memang sedikit ada selisih *minimum support*. Hal ini terjadi karena pembulatan antara perhitungan manual dan perhitungan dengan aplikasi berbeda.

- Hasil *association rule* untuk *dataset* ini ditunjukkan pada Gambar 7.

Dari Gambar 7 dapat diinterpretasikan hasil *association rule* sebagai berikut.

- Rule* 1, jika membeli kopi maka membeli gula dengan tingkat kepercayaan mencapai 60% dan didukung oleh 30% dari data keseluruhan.
- Rule* 2, jika membeli gula maka membeli kopi dengan tingkat kepercayaan mencapai 60% dan didukung oleh 30% dari data keseluruhan.
- Rule* 3, jika membeli susu maka membeli teh dengan tingkat kepercayaan mencapai 60% dan didukung oleh 30% dari data keseluruhan.
- Rule* 4, jika membeli teh maka membeli susu dengan tingkat kepercayaan mencapai 100% dan didukung oleh 30% dari data.

Jika dipersempit, ditemukan dua pasangan *item* barang yakni kopi, gula dan teh, susu yang memiliki *support* sebesar 30% dan *confidence* sebesar 70%.

4. Diskusi

Pengembangan aplikasi berbasis *desktop* untuk memberikan rekomendasi yang tepat bagi para penjual atau para penentu keputusan. Penggunaan metode FP-Growth dalam menentukan pilihan pasangan *item* produk sangat tepat. Seperti pasangan *item* produk gula dan kopi yang memang saling membutuhkan. Jika ingin membuat

kopi, pastinya membuat gula untuk membuat rasa kopi menjadi manis. Didukung dengan nilai *minimum confidence* yang berada di level 60% seperti yang ditunjukkan pada Gambar 7 membuat pasangan dari *item* produk gula dan kopi bisa direkomendasikan untuk dijual secara bersama-sama. Pilihan produk lain seperti teh dan susu juga dapat dijual secara bersama-sama.

Fokus dari algoritma ini adalah memberikan pilihan pasangan *item* produk berdasarkan hasil transaksi penjualan. Para pengguna aplikasi ini tidak hanya digunakan oleh penjual, tetapi juga para penentu keputusan yang mempunyai kebijakan yang dapat diterapkan di tempat usahanya. Kekurangan dari penelitian ini adalah aplikasi masih menggunakan *desktop*, sehingga kesulitan untuk dapat diakses oleh banyak orang dari berbagai tempat. *Association rule* yang didapatkan masih belum dikonversikan ke dalam barang, sehingga para penjual/penentu keputusan harus dapat mengingat nilai bilangan bulat yang sesuai dengan nama barang.

5. Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah sistem yang dibangun telah dapat memberikan rekomendasi paket produk yang sesuai dengan keragaman keinginan konsumen. Penentuan rekomendasi paket produk dihitung berdasarkan tingkat frekuensi *item* yang dibeli oleh konsumen dengan memperhatikan *minimum support* yang telah ditetapkan. Aplikasi rekomendasi paket produk menggunakan metode FP-Growth dapat menjadi alternatif bagi para penentu keputusan/penjual untuk memilih produk yang bisa digabungkan di dalam satu paket yang diharapkan berdampak pada peningkatan penjualan.

6. Daftar Pustaka

- [1] E. Supariyani, "Pengaruh Biaya Pelaksanaan Promosi Melalui Pameran terhadap Tingkat Volume Penjualan pada PT Astra Internasional Tbk Isuzu Cabang Bogor," *J. Ilm. Ranggad.*, vol. 4, no. 1, pp. 69–74, 2004.
- [2] A. Muzakir and L. Adha, "Market Basket Analysis (MBA) Pada Situs Web E-Commerce Zakiyah Collection," *J. SIMETRIS*, vol. 7, no. 2, pp. 459–466, 2016.
- [3] S. . Weng and J. L. Liu, "Feature-based recommendations for one-to-one marketing," *Expert Syst. Appl.*, vol. 26, pp. 493–508, 2004.
- [4] Y.-L. Chen, K. Tang, R.-J. Shen, and Y.-H. Hu, "Market basket analysis in a multiple store environment," *Decis. Support Syst.*, vol. 40, no. 2, pp. 339–354, 2004.
- [5] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," *Proc. 20th VLDB Conf.*, pp. 487–499, 1994.
- [6] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *Data Min. Knowl. Discov.*, vol. 8, no. 1, pp. 53–87, 2004.
- [7] H. Jiawei, K. Micheline, and P. Jian, *Data Mining Concepts and Techniques*, Third Edit. USA: Morgan Kaufmann Publishers, 2012.
- [8] L. C. Annie and A. Kumar, "Frequent Item set mining for Market Basket Data using K-Apriori algorithm," *Int. J. Comput. Intell. Informatics*, vol. 1, no. 1, pp. 14–18, 2011.
- [9] D. Widiastuti and N. Sofi, "Analisis Perbandingan Algoritma Apriori dan FP-Growth Pada Transaksi Koperasi," *UGJ*, vol. 8, no. 1, pp. 21–24, 2014.
- [10] Y. S. Nugroho, T. D. Salma, and S. Rokhanudin, "Implementasi Data Warehouse dan Data Mining Untuk Pengembangan Sistem Rekomendasi Pemilihan SMA," *Khazanah Informatika.*, vol. 2, no. 2, pp. 49–56, 2016.
- [11] A. Virrayyani and Sutikno, "Prediksi Penjualan Barang Menggunakan Metode Adaptive Neuro-Fuzzy Inference System (ANFIS)," *Khazanah Informatika.*, vol. 2, no. 2, pp. 57–63, 2016.
- [12] C. Zhang and S. Zhang, *Association Rule Mining: Models and Algorithms*. Heidelberg: Springer-Verlag Berlin, 2002.
- [13] Kusriani and E. T. Luthfi, *Algoritma Data Mining*. Yogyakarta: Andi Publisher, 2009.