

ANALISA DAN IMPLEMENTASI ASSOCIATION RULE DENGAN ALGORITMA FP-GROWTH DALAM SELEKSI PEMBELIAN TANAH LIAT (STUDI KASUS DI PT. ANVEVE ISMI BERJAYA)

Dio Prima Mulya¹⁾

1) Magister Ilmu Komputer, Universitas Putra Indonesia, Padang
email: dio.prima.mulya.dp@gmail.com

Abstract

Data Mining aims to draw abstract knowledge of a big database. Data Mining also known as Knowledge Discovery Database. FP-Growth algorithm is one of the very popular algorithms in finding frequent itemset in finding the rule of a large data base. Association rule used to find patterns in market basket analysis. Steps in the process of association rule mining is confidence and support. Clay formed from the weathering of silica by carbonic acid and partly generated by geothermal activity. In this study using FP-Growth Algorithm in purchasing decisions withdrawal clay by PT. ISMI ANVEVE BERJAYA.

Keyword: *Knowledge Discovery Database, Data Mining, Association Rule, Fp-Growth, frequent itemset.*

Abstrak

Data Mining bertujuan untuk menarik pengetahuan abstrak dari sebuah *database* yang besar. *Data Mining* juga dikenal sebagai *Knowledge Discovery Database*. Algoritma *FP-Growth* adalah salah satu algoritma yang sangat populer dalam mencari *frequent itemset* dalam menemukan rule dari basis data yang besar. *Association rule* digunakan dalam menemukan pola dalam *market basket analysis*. Langkah dalam proses *association rule mining* yaitu *confidence* dan *support*. Tanah liat terbentuk dari pelapukan silika oleh asam karbonat dan sebagian lagi dihasilkan oleh aktifitas panas bumi. Dalam penelitian ini menggunakan *Algoritma FP-Growth* dalam penarikan keputusan pembelian tanah liat oleh PT. ANVEVE ISMI BERJAYA.

Kata kunci : *Knowledge Discovery Database, Data Mining, Association Rule, Fp-Growth, frequent itemset.*

1. Pendahuluan

Penyajian informasi tidak sepadan dengan kebutuhan informasi yang sangat tinggi. Salah satu cara penggalan informasi dari sekumpulan *database* yang besar yaitu *Data Mining*. Tujuan dari *Data Mining* adalah untuk menarik pengetahuan abstrak dari sebuah *database* yang besar.

Hal terpenting dalam teknik *Data Mining* adalah aturan untuk menemukan pola frekuensi tinggi antar himpunan *itemset* yang disebut fungsi *association rules* (aturan asosiasi). Langkah-langkah dalam proses *association rule mining*, yaitu *confidence* dan *support*.

2. METODE

2.1 Knowledge Discovery in Database

Knowledge Discovery in Database (KDD) didefinisikan sebagai ekstraksi informasi potensial, implisit dan tidak dikenal dari sekumpulan data, Proses *Knowledge Discovery in Database* melibatkan hasil proses *Data Mining* (proses pengekstrak kecenderungan suatu pola data), kemudian mengubah hasilnya secara akurat menjadi informasi yang mudah dipahami (Tampubolon *et al*, 2013).

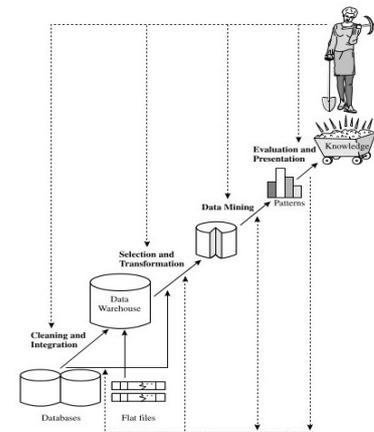
KDD terdiri dari tiga proses utama yaitu (anis Kurniawati, 2014).

- a. *Preprocessing*
Preprocessing dilakukan terhadap data sebelum algoritma Data Mining diaplikasikan. Proses ini meliputi data cleaning, integrasi, seleksi dan transformasi.
- b. *Data Mining*
Proses utama dalam KDD adalah proses Data Mining, dalam proses ini algoritma-algoritma Data Mining diaplikasikan untuk mendapatkan pengetahuan dari sumber data.
- c. *Post processing*
Hasil yang diperoleh dari proses *Data Mining* selanjutnya akan dievaluasi pada *post processing*.

Menurut Han dan Kamber (2012) *Knowledge discovery* sebagai sebuah proses yang digambarkan pada gambar 2.1, dan terdiri atas runtunan *iterative* dari langkah-langkah berikut :

- a. *Data cleaning* (untuk membuang *noise* dan data yang tidak konsisten).
- b. *Data integration* (beragam sumber data digabungkan).
- c. *Data selection* (data yang relevan untuk dianalisa di *retrive* dari *database*)
- d. *Data transformation* (dimana data diubah atau dikonsolidasi ke dalam bentuk yang sesuai untuk pertambangan misalnya dengan melakukan ringkasan atau operasi *agregasi*).
- e. *Data Mining* (sebuah proses esensi dimana metode cerdas diterapkan untuk mengekstraksi pola-pola data).
- f. *Pattern evaluation* (mengidentifikasi pola yang benar menarik merepresentasikan dasar pengetahuan pada beberapa ukuran menarik).
- g. *Knowledge presentation* (teknik representasi visualisasi

dan pengetahuan digunakan untuk memperlihatkan pengetahuan yang telah ditambang kepada pengguna).



2.2 Data Mining

Data Mining adalah penambangan atau penemuan informasi baru dengan mencari pola atau aturan tertentu dari sejumlah data yang sangat besar (Beta *et al* dalam jurnal Sarjon Defit, 2013).

2.2.1 Pengelompokan Data Mining

Data Mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu (Kusrini dan Emha Taufiq Luthfi, 2009 dalam Tampubolon, saragih, reza, 2013):

- a. *Deskripsi*
Proses pencarian data untuk menggambarkan pola dan kecenderungan yang terdapat dalam data.
- b. *Estimasi*
Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih kearah *numerik* dari pada kearah kategori
- c. *Prediksi*
Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada dimasa mendatang.
- d. *Klasifikasi*
Dalam klasifikasi, terdapat target variabel kategori.

- e. *Clustering* (Pengkusteran)
Kluster adalah kumpulan *record-record* yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan dengan *record-record* dalam *kluster* lain.
- f. Asosiasi
Tugas asosiasi dalam *Data Mining* adalah menemukan *attribut* yang muncul dalam satu waktu.

2.2.2 Tahapan Dalam Data Mining

Menurut Feen Lee & Juan Santana (2010:37-40) langkah-langkah yang dibutuhkan untuk mengerjakan implementasi *Data Mining* sebagai berikut (Tampubolon *et al*, 2013):

- a. Tahap pertama : *Precise statement of the problem* (mendefinisikan permasalahan yang ingin diketahui). Tugas dasar yang menjadi dasar algoritma *Data Mining* adalah klasifikasi, regresi, segmentasi, asosiasi dan *sequence* analisis.
- b. Tahap kedua : *Initial Exploration* (mempersiapkan data yang menjadi sumber untuk *Data Mining* termasuk data "*cleaning*" untuk mempelajari polanya). Setelah menemukan definisi masalah, langkah berikutnya adalah mencari data yang mendukung definisi masalah. Menentukan porsi data yang digunakan men-training *Data Mining* berdasar algoritma *Data Mining* yang telah dibuat. Setelah persiapan data selesai dilakukan, langkah berikutnya adalah memberikan sebagian data kedalam algoritma *Data Mining*.
- c. Tahap ketiga : *Model building and validation*. Validasi apakah *Data Mining* memberikan prediksi yang

akurat. Setelah training data selesai dilakukan, *Data Mining* tersebut perlu di-"uji" atau di-validasi keakuratannya terhadap data testing.

- d. Tahap ke-empat : *Deployment*. Tahap ini memilih aplikasi yang tepat terhadap *Data Mining* untuk membuat prediksi.

Berdasarkan beberapa pengertian diatas dapat ditarik kesimpulan bahwa *Data Mining* adalah sebuah analisa dari observasi data dalam jumlah besar untuk menemukan hubungan yang tidak diketahui sebelumnya dan metode baru untuk meringkas data agar mudah dipahami serta kegunaan untuk pemilik data.

2.3 Association Rule (Aturan Asosiasi)

Beberapa pengertian *Association rule* yang berhasil penulis himpun dari beberapa literatur adalah sebagai berikut:

- a. *Association rule* adalah salah satu metode yang bertujuan mencari pola yang sering muncul diantara banyak transaksi, dimana setiap transaksi terdiri dari beberapa item (Sarjon Defit, 2013).
- b. *Association rule* adalah salah satu teknik *Data Mining* yang mana diharapkan untuk menjadi aplikasi yang sangat bermanfaat (D. Magdelene Delighta Angeline, 2013).
- c. Menurut amirudin *et al* (2007) Analisis asosiasi atau *association rule mining* adalah teknik *Data Mining* untuk menemukan aturan asosiasi antara kombinasi item (Tampubolon *et al*, 2013).
- d. Menurut Goldie,G & Dana, I.S (2012) Analisis asosiasi adalah teknik *Data Mining* untuk menemukan aturan asosiasi antara suatu kombinasi item (Sarjon Defit, 2013).

- e. *Association rule* merupakan salah satu bentuk pola yang dihasilkan oleh *Data Mining*. *Association rule* dapat digunakan untuk menemukan hubungan atau sebab akibat. Penting tidaknya suatu asosiasi dapat diketahui dengan dua tolak ukur, yaitu : *support* dan *confidence*. *Support* (nilai penunjang) adalah persentasi kombinasi item tersebut dalam *database*, sedangkan *confidence* (nilai kepastian) adalah kuatnya hubungan antar item dalam aturan asosiasi (Eka Novita Sari, 2013).

Association rule mempunyai bentuk LHS (*Left Hand Shake*) -> RHS (*Right Hand Shake*) dengan interpretasi jika setiap item dalam LHS dibeli maka sepertinya item dalam RHS juga dibeli. Dua pengukuran penting untuk sebuah *rule* adalah *Support* dan *Confidace*.

Support (s) didefinisikan sebagai presentasi *record* terhadap total *record* keseluruhan dalam *database*. *Support* dapat ditulis dalam rumus berikut :

$$\text{Support} = \frac{\text{Banyaknya } X}{\text{Jumlah total transaksi dalam database } (D)}$$

Sementara nilai support dari 2 item dituliskan dalam rumus berikut:

$$\text{Support } (X,Y) = \frac{P(X \cap Y)}{\Sigma \text{ Banyaknya transaksi } X \text{ dan } Y} = \frac{\Sigma \text{ Transaksi}}{\Sigma \text{ Transaksi}}$$

Jadi jika misalnya *Support* sebuah item adalah 0,1 % maknanya hanya 0,1 persen dari transaksi berisikan item ini.

Sementara *Confidence* (c) didefinisikan sebagai presentasi dari sejumlah transaksi yang berisikan $X \cup Y$ terhadap jumlah *record* keseluruhan yang mengandung X. *Confidence* dapat dituliskan dalam rumus berikut :

$$\text{Confidence } (X|Y) = \frac{\text{Support } (XY)}{\text{Support } (X)}$$

Confidence merupakan ukuran kekuatan sebuah *rule* asosiasi. Jika kita anggap *confidence* $X \rightarrow Y$ adalah 80%, berarti sebanyak 80 persen transaksi yang mengandung X maka akan mengandung jika Y.

Pada umumnya, menurut Han dan Kamber (2012:245) *association rule mining* dapat dilihat sebagai proses dua langkah, yaitu:

- Temukan semua *frequent itemsets*, dengan pengertian setiap *itemset* ini akan muncul sesering sekurangnya minimum *support* (*min-sup*) yang telah ditentukan.
- Hasilkan *association rules* yang kuat dari *frequent itemsets*: dengan pengertian *rule-rule* ini haruslah memenuhi minimum *Support* dan minimum *Confidence*.

2.4 Langkah – Langkah Proses Aturan Asosiasi

Menurut Eko Wahyu Tyas d (2008) proses aturan asosiasi terdiri dari beberapa tahap sebagai berikut (Tampubolon *et al*, 2013):

- Sistem *men-scan database* untuk mendapatkan kandidat *1-itemset* (himpunan item yang terdiri dari 1 item) dan menghitung nilai supportnya. Kemudian nilai supportnya tersebut dibandingkan dengan minimum *support* yang telah ditentukan, jika nilainya lebih besar atau sama dengan minimum *support* maka *itemset* tersebut termasuk dalam *large itemset*.
- Itemset* yang tidak termasuk dalam *large itemset* tidak diikutkan dalam iterasi selanjutnya (diprun).
- Pada iterasi kedua sistem akan menggunakan hasil *large itemset* pada iterasi pertama (L1) untuk membentuk kandidat *itemset* kedua (L2). Pada iterasi selanjutnya sistem akan menggunakan hasil *large itemset* pada iterasi selanjutnya akan

- menggunakan hasil *large itemset* pada iterasi sebelumnya (Lk-1) untuk membentuk kandidat *itemset* berikut (Lk). Sistem akan menggabungkan (join) Lk-1 dengan Lk-1 untuk mendapatkan Lk, seperti pada iterasi sebelumnya sistem akan menghapus (*prune*) kombinasi *itemset* yang tidak termasuk dalam *large itemset*.
- d. Setelah dilakukan operasi join, maka pasangan *itemset* baru hasil proses *join* tersebut dihitung supportnya.
 - e. Proses pembentuk kandidat yang terdiri dari proses join dan *prune* akan terus dilakukan hingga himpunan kandidat *itemset*nya null, atau sudah tidak ada lagi kandidat yang akan dibentuk.
 - f. Setelah itu, dari hasil *frequent itemset* tersebut dibentuk *association rule* yang memenuhi nilai *support* dan *confidence* yang telah ditentukan.
 - g. Pada pembentukan *association rule*, nilai yang sama dianggap sebagai satu nilai.
 - h. *Association rule* yang terbentuk harus memenuhi nilai minimum yang telah ditentukan.
 - i. Untuk setiap *large itemset* L, kita cari himpunan bagian L yang tidak kosong. Untuk setiap himpunan bagian tersebut, dihasilkan *rule* dengan bentuk $aB(L-a)$ jika supportnya (L) dan supportnya (a) lebih besar dari minimum *support*.

2.5 Algoritma FP-Growth

Menurut Gunadi dan Sanuse (2012) *FP-Growth* adalah salah satu alternatif algoritma yang dapat digunakan untuk menentukan himpunan data yang paling sering muncul (*frequent itemset*) dalam

sebuah kumpulan data. *FP-Growth* menggunakan pendekatan yang berbeda dari paradigma yang digunakan pada algoritma apriori.

Menurut Ririanti (2014) *FP-Growth* adalah salah satu alternatif algoritma yang dapat digunakan untuk menentukan himpunan data yang paling sering muncul (*frequent itemset*) dalam sekumpulan data.

Algoritma *FP-Growth* merupakan pengembangan dari algoritma *Apriori*. *FP-Growth* adalah metode yang membangun sebuah struktur data yang sangat padat (*FP-Tree*) untuk kompres database transaksi asli.

Algoritma *FP-Growth* adalah metode yang efisien dimana *mining* dilakukan dengan struktur *prefix-tree* pada suatu *itemset* yang lengkap. Struktur *tree* menyimpan informasi tentang *frequent pattern*.

Algoritma ini dimulai dengan mengompresi database input, sehingga mengembangkan contoh dari sebuah *frequent pattern tree*. Database yang dikompresi kemudian dibagi menjadi beberapa database bersyarat. Dimana setiap database merepresentasikan sebuah *unique frequent pattern*. Oleh karena itu, biaya pencarian berkurang dan menawarkan selektivitas yang baik (Shivam *et al*, 2014).

Metode *FP-Growth* dibagi menjadi tiga tahapan utama, yaitu (Han *et al*, 2012):

1. Tahap pembangkitan *conditional pattern base*,
2. Tahapan pembangkitan *conditional FP-Tree*, dan
3. Tahapan pencarian *frequent itemset*.

2.6 Tanah Liat (Clay)

Tanah liat atau lempung adalah partikel mineral berkerangka dasar silikat yang berdiameter kurang dari 4 mikrometer. Lempung mengandung leburan silika dan/atau aluminium yang halus. Unsur-unsur ini, silikon, oksigen, dan aluminium adalah unsur yang paling banyak menyusun komposisi bumi. Lempung terbentuk

dari proses pelapukan batuan silika oleh asam karbonat dan sebagian dihasilkan dari aktivitas panas bumi (S.O. Obaje, J.I. Omada, dan U.A. Dambatta, 2013).

2.7 RAPIDMINER

Rapidminer adalah sebuah alat Data Mining yang digunakan untuk menganalisis informasi web yang diakses. Rapidminer juga digunakan untuk penelitian, pendidikan, rapid prototyping, pengembangan aplikasi dan aplikasi industri. Proyek Rapidminer dimulai pada tahun 2001 oleh Ralf Klinkenberg, Ingo Mierswa, dan Simon Fischer di Artificial Intelligence Group of Katharina Morik di Dortmund University of Technology.

Aplikasi yang merupakan Open Source ini termasuk data cleaning, data transformation, optimization, validation, and visualization. Visualisasinya dapat dilihat dalam bentuk scatter plot, bar, pie chart, dll. (M.Santhanakumar and C.Christopher Columbus, 2015).

3. Analisa dan Hasil

3.1 Persiapan Data

Dalam analisa ini penulis menganalisa dan mengelompokkan data sesuai dengan variabel yang dibutuhkan dalam menentukan pembelian tanah liat oleh PT. ANVEVE ISMI BERJAYA.

Pada Algoritma FP-Growth terbagi mejadi 3 tahapan, yaitu pembangkitan conditional pattern base, pembangkitan conditional FP-Tree, pencarian frequent itemset.

Data utama yang digunakan pada penelitian ini adalah data pembelian tanah liat oleh PT.ANVEVE ISMI BERJAYA pada bulan April sampai dengan Agustus tahun 2016 yang merupakan data yang akan dijadikan sampel dalam penerapan algoritma FP-Growth untuk penentuan pembelian tanah liat sesuai dengan pemenuhan kebutuhan perusahaan.

Tabel 3.1 Produsen Tanah

KODE	Produsen
A	Lantamal
B	Bungus
C	Gunung Sariak
D	Sayang Ibu
E	Kuranji

Tabel 3.2 Kualitas Tanah

KODE	KUALITAS TANAH
A1	PRODUSEN A MINERAL TINGGI
A2	PRODUSEN A MINERAL SEDANG
A3	PRODUSEN A MINERAL RENDAH
B1	PRODUSEN B MINERAL TINGGI
B2	PRODUSEN B MINERAL SEDANG
B3	PRODUSEN B MINERAL RENDAH
C1	PRODUSEN C MINERAL TINGGI
C2	PRODUSEN C MINERAL SEDANG
C3	PRODUSEN C MINERAL RENDAH
D1	PRODUSEN D MINERAL TINGGI
D2	PRODUSEN D MINERAL SEDANG
D3	PRODUSEN D MINERAL RENDAH
E1	PRODUSEN E MINERAL TINGGI
E2	PRODUSEN E MINERAL SEDANG
E3	PRODUSEN E MINERAL RENDAH

Tabel 3.3 Data Pembelian Setelah di-konversi berdasarkan pemenuhan kebutuhan perusahaan

	A1	A2	B2	B3	C1	C3	D1	D2	E1	E2	E3
M1	1	1	1	0	0	0	1	0	1	0	1
M2	1	0	1	1	1	0	1	0	0	0	1
M3	0	1	1	0	1	0	1	1	0	1	0
M4	1	0	0	1	0	1	1	0	0	1	0
M5	0	0	1	1	1	0	1	1	1	0	1
M6	1	1	0	0	0	0	0	0	0	0	0
M7	1	0	0	0	1	1	0	1	0	1	1
M8	1	0	1	0	1	0	1	0	1	1	0
M9	0	1	1	1	0	0	1	1	0	0	0
M10	0	1	0	1	1	1	0	0	0	0	1
M11	0	0	1	1	1	0	1	1	1	1	0
M12	1	0	0	1	1	0	0	0	0	0	1
M13	0	1	0	1	0	0	0	1	0	0	0
M14	1	0	1	0	0	1	1	1	1	1	1
M15	0	0	1	0	1	0	1	0	0	1	0
M16	1	0	1	1	1	0	0	1	0	0	1
M17	1	0	0	1	0	1	0	0	1	0	1
M18	1	1	1	0	1	0	1	1	0	1	0
M19	0	1	0	1	1	1	1	1	1	0	0
M20	0	1	0	0	1	0	0	0	0	0	1

Tabel 3.3 Pembacaan pembelian pada setiap transaksi

	Itemset
M1	A1,A2,B2,D1,E1,E3
M2	A1,B2,B3,C1,D1,E3
M3	A2,B2,C1,D1,D2,E2
M4	A1,B3,C3,D1,E2
M5	B2,B3,C1,D1,D2,E1,E3
M6	A1,A2
M7	A1,C1,C3,D2,E2,E3
M8	A1,B2,C1,D1,E1,E2
M9	A2,B2,B3,D1,D2
M10	A2,B3,C1,C3,E3
M11	B2,B3,C1,D1,D2,E1,E2
M12	A1,B3,C1,E3
M13	A2,B3,D2
M14	A1,B2,C3,D1,D2,E1,E2,E3
M15	B2,C1,D1,E2
M16	A1,B2,B3,C1,D2,E3
M17	A1,B3,C3,E1,E3
M18	A1,A2,B2,C1,D1,D2,E2
M19	A2,B3,C1,C3,D1,D2,E1
M20	A2,C1,E3

Tabel 3.4 Kemunculan Item

Item	Jumlah
A1	11
A2	9
B2	11
B3	11
C1	13
C3	6
D1	12
D2	10
E1	7
E2	8
E3	10

3.2 Perhitungan Algoritma FP-Growth

Mendata kemunculan item diurut berdasarkan frekuensi tertinggi.

Tabel 3.5 Kemunculan item berdasarkan frekuensi tertinggi

Item	Jumlah
C1	13
D1	12
A1	11
B2	11
B3	11
D2	10
E3	10
A2	9
E2	8
E1	7
C3	6

Setelah dilakukan Pengurutan didapat *item* yang memiliki frekuensi di atas *support count* ≥ 10 ($30/108=0.27$) adalah C1,D1,A1,B2,B3,D2, dan E3. Enam itemset ini yang akan berpengaruh dan akan dimasukkan ke dalam *FP-tree*

Tabel 3.6 Kemunculan Item Pada Setiap Transaksi Berdasarkan Frequent Paling Tinggi

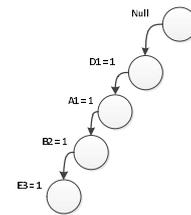
Trans.	Item
M1	D1,A1,B2,E3
M2	C1,D1,A1,B2,B3,E3
M3	C1,D1,B2,D2,E3
M4	D1,A1,B3
M5	C1,D1,B2,B3,D2,E3
M6	A1
M7	C1,A1,D2,E3
M8	C1,D1,A1,B2
M9	D1,B2,B3,D2
M10	C1,B3,E3
M11	C1,D1,B2,B3,D2
M12	C1,A1,B3,E3
M13	B3,D2
M14	D1,A1,B2,D2,E3
M15	C1,D1,B2
M16	C1,A1,B2,B3,D2,E3
M17	A1,B3,E3
M18	C1,D1,A1,B2,D2
M19	C1,D1,B3,D2
M20	C1,E3

3.3 Tahap Pembangunan Fp-Tree

Dalam proses pembangunan *FP-Tree* dengan menggunakan *frequent items* yang telah dipangkas dan diurutkan. Data pembelian tanah liat dalam 20 (dua puluh) minggu dengan

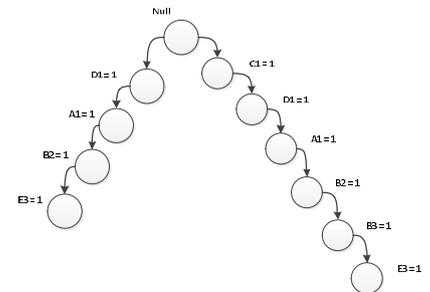
7 jenis item (tanah liat), dipindahkan ke dalam bentuk *FP-Tree* setiap minggu pembelian. Setiap simpul pada *FP-Tree* mengandung kode *item* dan *counter support* yang berfungsi untuk menghitung frekuensi kemunculan *item* tersebut dalam tiap lintasan mingguan.

Item yang tidak *frequent* dibuang, sedangkan *frequent item* dimasukkan dan disusun dengan urutan menurun, seperti yang terlihat pada gambar 4.1. Berikut adalah bentuk di minggu ke-1 :



Gambar 3.1 Hasil Pembentukan *FP-Tree* Setelah Pembacaan M1(D1,A1,B2,E3)

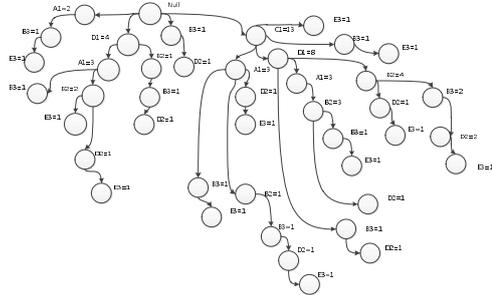
Dari pembacaan pembelian pada minggu ke-1, M1 {D1,A1,B2,E3} maka terbentuk lintasan transaksi Null→D1→A1→B2→E3. *Support count* dari setiap simpul bernilai awal 1 (satu).



Gambar 3.2 Hasil Pembentukan *FP-Tree* Setelah Pembacaan M2 (C1,D1,A1,B2,B3,E3)

Pada pemindaian kedua pada gambar 3.2, dimana pembacaan Transaksi M2 {C1,D1,A1,B2,B3,E3} akan membuat simpul C1, D1, A1, B2, B3, dan E3, sehingga terbentuk lintasan transaksi Null→C1→D1→A1→B2→B3→E3.

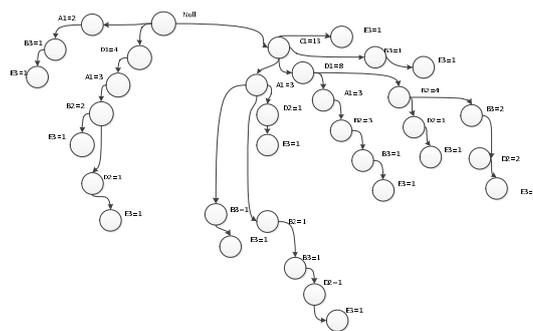
Setelah dilakukan proses pembentukan *FP-Tree* pada transaksi M1 hingga M20, maka didapatkan bentuk *Tree* pada transaksi terakhir, yaitu bentuk *FP-Tree* pada transaksi minggu ke-20 (M20), sebagai berikut:



Gambar 3.3 Hasil Pembentukan *FP-Tree* Setelah Pembacaan M20 (C1,E3)

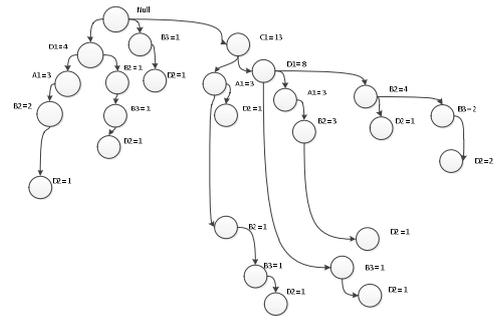
3.4 Tahap pencarian Frequent Itemset

Dalam menemukan *frequent itemset* maka diperlukan upapohon (pohon yang akarnya adalah keturunan dari akar pohon induknya) dengan lintasan yang memiliki *support count* terkecil, yaitu E3. Proses pembentukannya dapat dilihat pada gambar berikut :



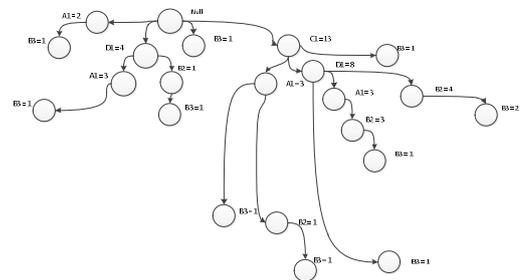
Gambar 3.3 Lintasan Yang Memiliki Nilai E3

Pada Gambar 3.3 Semua lintasan yang memiliki nilai E3 ditampilkan, lintasan yang tidak memiliki nilai E3 dihapuskan. Lintasan diambil acuan adalah *Fp-tree* lengkap yang ada pada gambar 3.2



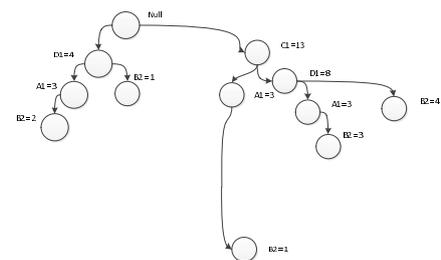
Gambar 4.22 Lintasan Yang Memiliki Nilai D2

Pada gambar 4.22 ditampilkan akar yang memiliki nilai D2.



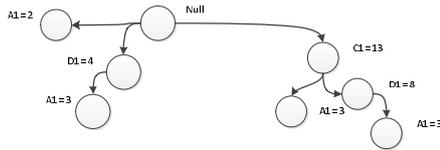
Gambar 4.23 Lintasan yang memiliki akar B3

Pada Gambar 4.23 menampilkan lintasan yang memiliki akar akhir B3.



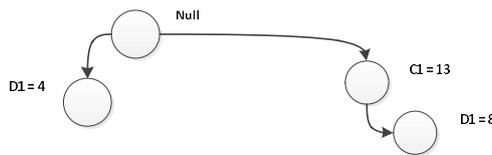
Gambar 4.24 Lintasan Yang Memiliki B2

Pada gambar 4.24 Bentuk Lintasan B2 yang ditampilkan.



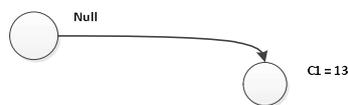
Gambar 4.25 Lintasan Yang Memiliki Nilai A1

Pada gambar 4.25 Bentuk Lintasan yang memiliki nilai A1.



Gambar 4.26 Lintasan Yang Memiliki Nilai D1

Pada gambar 4.26 Lintasan yang memiliki nilai D1.



Gambar 4.27 Lintasan Yang Memiliki Nilai C1

Pada gambar 4.27 Lintasan C1 memiliki simpul tunggal yaitu C1 yang memiliki nilai support count tertinggi.

3.5 Rule Asosiasi Algoritma FP-Growth

Dalam menentukan Rule Asosiasi Algoritma Fp-Growth, pertama yang dilakukan adalah menemukan semua frequent itemset yang berakhiran E3, D2, B3, B2, A1, D1, dan C1. Lalu cek apakah support count dari E3,D2,B3,B2,A1,D1, dan C1 memenuhi minimum support count ≥ 10 (minimum support ≥ 0.27).

Dari pembangkitan FP-tree yang telah dilakukan maka didapatkan hasil Conditional Pattern Base dan Conditional Fp-Tree sebagai berikut:

Tabel 3.7 Hasil Conditional Pattern Base dan Conditional Fp-Tree dari upapohon.

Suffix	Conditional Pattern Base	Conditional FP-Tree
E3	{(A1,B3):1, (D1,A1,B2):1, (D1,A1,B2,E3,D2):1, (C1,A1,B3):1, (C1,B3):1, (C1,A1,D2):1, (C1,A1,B2,B3,D2):1, (C1,D1,A1,B2,B3):1, (C1,D1,B2,D2):1, (C1,D1,B2,B3,D2):1}	{{(C1:7,D1:5,A1:7,B2:6, B3:6,D2:5)} (C1,D1,A1,B2,B3,D2)}
D2	{(D1,A1,B2):1, (D1,B2,B3):1, (C1,A1):1, (C1,A1,B2,B3):1, (C1,D1,B3), (C1,D1,A1,B2):1, (C1,D1,B2):1, (C1,D1,B2,B3):2}	{{(C1:7,D1:7,A1:4,B2:6, B3:5)} (C1,D1,A1,B2,B3)}
B3	{(D1,A1):1, (D1,B2):1, (C1,A1):1, (C1,A1,B2):1, (C1,D1):1, (C1,D1,A1,B2):1, (C1,D1,B2):2}	{{(C1:6,D1:6,A1:4, B2:5)} (C1,D1,A1,B2)}
B2	{(D1,A1):2, (C1,A1):1, (C1,D1,A1):3, (C1,D1):4}	{{(C1:7,D1:9,A1:6)} (C1,D1,A1)}
A1	{(C1,D1):3}	{{(C1:3,D1:3)}(C1,D1)}
D1	{{(C1:8)}(C1)}	{{(C1:8)}(C1)}
C1		

Dari hasil Conditional FP-Tree dan Conditional Pattern Base maka didapatkan kombinasi Frequent Itemset. Selanjutnya menghitung nilai support dan Confidence dari kombinasi Frequent Itemset.

Agar tingkat akurasi tinggi maka ditetapkan minimum Confidence $\geq 83\%$ (0.83).

Tabel 3.8 kombinasi frequent itemset yang memenuhi Minimum Support ≥ 0.27 dan Confidence ≥ 0.83

no	X	Y	Support	Confidence
1	D1	B2	0,5	0,83
2	E1	D1	0,3	0,86
3	D1, D2	B2	0,3	0,86
4	B2, D2	D1	0,3	0,86
5	D1, E2	B2	0,3	0,86
6	E2	D1	0,35	0,88
7	C1, D1	B2	0,35	0,88
8	C1, B2	D1	0,35	0,88
9	B2	D1	0,5	0,91
10	B2, E2	D1	0,3	1,00

Maka dari hasil tabel 3.8 diatas didapatkan pola Rule Asosiasi sebagai berikut:

- [D1] --> [B2] (confidence: 0.83)
Jika Perusahaan membeli tanah liat mineral tinggi dari produsen Sayang Ibu (D1) Maka Perusahaan

juga akan membeli tanah liat mineral sedang dari produsen Bungus (B2) dengan nilai *Confidence* 83%.

2. [E1] --> [D1] (*confidence*: 0.86)

Jika Perusahaan membeli tanah liat mineral tinggi dari produsen Kuranji (E1) maka Perusahaan akan membeli tanah liat mineral tinggi pada produsen Sayang Ibu (D1) dengan nilai *Confidence* 86%.

3. [D1, D2] --> [B2] (*confidence*: 0.86)

Jika Perusahaan membeli tanah liat mineral tinggi dari produsen Sayang Ibu (D1) dan tanah liat mineral sedang dari produsen Sayang Ibu (D2) maka Perusahaan akan membeli tanah liat mineral sedang pada produsen Bungus (B2) dengan nilai *Confidence* 86%.

4. [B2, D2] --> [D1] (*confidence*: 0.86)

Jika Perusahaan membeli tanah liat mineral sedang dari produsen Bungus (B2) dan tanah liat mineral sedang dari produsen Sayang Ibu (D2) maka Perusahaan akan membeli tanah liat mineral tinggi pada produsen Sayang Ibu (D1) dengan nilai *Confidence* 86%.

5. [D1, E2] --> [B2] (*confidence*: 0.86)

Jika Perusahaan membeli tanah liat mineral tinggi dari produsen Sayang Ibu (D1) dan tanah liat mineral sedang dari produsen Kuranji (E2) maka Perusahaan akan membeli tanah liat mineral sedang pada produsen Bungus (B1) dengan nilai *Confidence* 86%.

6. [E2] --> [D1] (*confidence*: 0.88)

Jika Perusahaan membeli tanah liat mineral sedang dari produsen Kuranji (E2) maka Perusahaan akan membeli tanah liat mineral tinggi pada produsen Sayang Ibu (D1) dengan nilai *Confidence* 88%.

7. [C1, D1] --> [B2] (*confidence*: 0.88)

Jika Perusahaan membeli tanah liat mineral tinggi dari produsen Gunung Sariak (C1) dan tanah liat mineral tinggi dari produsen Sayang Ibu (D1) maka Perusahaan akan membeli tanah liat mineral sedang

pada produsen Bungus (B1) dengan nilai *Confidence* 88%.

8. [C1, B2] --> [D1] (*confidence*: 0.88)

Jika Perusahaan membeli tanah liat mineral tinggi dari produsen Gunung Sariak (C1) dan tanah liat mineral sedang pada produsen Bungus (B2) maka Perusahaan akan membeli tanah liat mineral tinggi pada produsen Sayang Ibu (D1) dengan nilai *Confidence* 88%.

9. [B2] --> [D1] (*confidence*: 0.91)

Jika Perusahaan membeli tanah liat mineral sedang dari produsen bungus (B1) maka Perusahaan akan membeli tanah liat mineral tinggi dari produsen Sayang Ibu (D1) dengan nilai *Confidence* 91%.

10. [B2, E2] --> [D1] (*confidence*: 1.00)

Jika Perusahaan membeli tanah liat mineral sedang dari produsen Bungus (B1) dan tanah liat mineral sedang dari produsen Kuranji (E2) maka Perusahaan akan membeli tanah liat mineral tinggi pada produsen Sayang Ibu (D1) dengan nilai *Confidence* 100%.

4. Kesimpulan dan Saran

4.1 Kesimpulan

Dari uraian pada bab-bab yang sudah dibahas sebelumnya dapat ditarik kesimpulan :

1. Dengan *Algoritma FP-Growth* dapat membantu seleksi pembelian tanah liat dengan menghasilkan *rule*. *Algoritma FP-Growth* memberikan informasi eksklusif dan menggambarkan proses dalam pembelian tanah liat oleh perusahaan.
2. Penggunaan *Algoritma Fp-Growth* dan aplikasi *software Rapidminer 7.4.0* didapatkan hasil berupa aturan (*rules*) yang merupakan kumpulan dari beberapa *frequent itemset* dengan nilai *confidence* yang tinggi.
3. Hasil *rule* dari *Algoritma FP-Growth* dan *software Rapidminer Studio 7.4.0* membantu untuk pimpinan perusahaan membuat suatu perencanaan pembelian tanah liat.

4.2 Saran

Dalam implementasi *Algoritma FP-Growth* dalam membuat rule atau aturan untuk pengambilan keputusan belanja sangat membantu bagi perusahaan. Berikut adalah saran yang dapat diperhatikan untuk masa yang akan datang.

1. Implementasi *Algoritma FP-Growth* sangat praktis namun harus dilakukan perbandingan dengan *Algoritma* lain, sebagai bahan pengujian sejauh mana *Algoritma FP-Growth* dapat diandalkan dalam pengambilan keputusan dan pembuatan aturan.
2. Penerapan *Algoritma Fp-Growth* sangat tergantung pada penentuan *minimum support*, semakin besar *minimum support*-nya maka semakin besar pula *performance* dari *Algoritma*-nya. Begitupun sebaliknya.
3. Dalam penambahan data, sebaiknya ditambahkan jumlah transaksi yang lebih banyak, agar *rule* yang dihasilkan lebih akurat.

5. Daftar Pustaka

- Aggarwal, S., & Kaur, R.** (2013). Comparative Study of Various Improved Versions of Apriori Algorithm. *International Journal of Engineering Trends and Technology (IJETT)*, 4(4), 687–690.
- Angeline, D. Magdalene, D.** (2013). Association Rule Generation for Student Performance Analysis using Apriori Algorithm. *The SIJ Transactions on Computer Science Engineering & Its Applications (CSEA)*, 1(1), p12--16.
- Defit, S.** (2013). Penggunaan Algoritma Apriori Dalam Menganalisa Perilaku Mahasiswa Dalam Memilih Mata Kuliah (Studi Kasus : Fkip Upi “ Yptk ”), 8(3), 31–42.
- Gunadi, G., & Sensuse, D. I.** (2012). Penerapan Metode Data Mining Market Basket Analysis Terhadap Data Penjualan Produk Buku Dengan Menggunakan Algoritma Apriori Dan Frequent Pattern Growth (Fp-Growth): *Telematika*, 4(1), 118–132.
- Ikhwan, A., Nofriansyah, D., & Sriani.** (2015). Penerapan Data Mining dengan Algoritma Fp-Growth untuk Mendukung Strategi Promosi Pendidikan (Studi Kasus Kampus STMIK Triguna Dharma). *Saintikom*, 14(3), 211–226.
- Jiawei, H., Kamber, M., & Pei, J.** (2012). *Data Mining: Concepts and Techniques*. San Francisco, CA, itd: Morgan Kaufmann. <https://doi.org/10.1016/B978-0-12-381479-1.00001-0>
- Kurniawati, A.** (2014). Pemetaan Pola Hubungan Program Studi Dengan Algoritma Apriori Studi Kasus SPMU UNNES. *EDUKOM*, 1(1), 51–58.
- Obaje, S. O., Omada, J. I., & Dambatta, U. A.** (2013). Clays and their Industrial Applications: Synoptic Review. *International Journal of Science and Technology*, 3(5), 264–270.
- Ririanti.** (2014). Implementasi Algoritma Fp-Growth Pada Aplikasi Prediksi Persediaan Sepeda Motor (Studi Kasus Pt . Pilar Deli Labumas). *Pelia Informatika Budi Darma*, VI(1), 139–144.
- Sari, E. N.** (2013). Analisa Algoritma Apriori Untuk Menentukan Merek Pakaian Yang Paling Diminati Pada Mode Fashion Group Medan. *Jurnal Teknik Informatika*, IV, 35–39.
- Sidhu, S.** (2014). FP Growth Algorithm Implementation, 93(8), 6–10.
- Singh, J., & Ram, H.** (2013). Improving efficiency of Apriori algorithm using transaction reduction. *International Journal of Scientific and Research Publications*, 3(1), 1–4.
- Suresh, J., Rushyanth, P., & Trinath, C.** (2013). Generating associations rule mining using Apriori and FPGrowth Algorithms. *International Journal of Computer Trends and Technology*, 4(4), 887–892.
- Tampubolon, K., Saragih, H., Reza,** (2013). Implementasi Data Mining Algoritma Apriori Pada Sistem Persediaan Alat-Alat Kesehatan. *Informasi Dan Teknologi Ilmiah*, 93–106.