PEMBANGUNAN PERANGKAT LUNAK *DATA MINING* UNTUK PENENTUAN PRODUKSI DI PT. CONTIN MOTOSPORT MENGGUNAKAN METODE *ASSOCIATION RULE*

Nirwan Saeful Ahmad

[1,2]Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
Email : nirwan.saeful17@gmail.com

ABSTRAK

Data mining merupakan proses penggalian data dalam jumlah besar di mana proses ini bertujuan untuk mendapatkan informasi dan pengetahuan baru yang bermanfaat. Salah satu metode *data mining* yang dapat digunakan adalah *Association Rule*. Metode ini dimulai dengan mencari *frequent itemset* dan menghasilkan *rule* yang dapat dijadikan acuan produksi jenis barang pada perusahaan.

Pendekatan *data mining* dapat digunakan sebagai solusi yang ditawarkan kepada PT. Contin Motosport untuk mengatasi kendala yang sedang dihadapi dalam meminimalisir menumpuknya jenis hasil produksi dengan memberikan rekomendasi produksi jenis barang yang tepat.

Hasil pengujian menunjukkan bahwa metode *association rule* dengan algoritma *CT-Pro* berhasil diimplementasikan untuk menentukan *rule* atau aturan terkait rekomendasi prduksi pada PT. Contin Motosport. Dari penelitian yang telah dilakukan, ditemukan juga bahwa dari data transaksi yang di dapat dan diujicobakan berdasarkan kombinasi parameter *minimum support* dan *minimum confidence* sehingga didapat kesimpulan bahwa pemilihan kombinasi nilai *minimum support* dan nilai *minimum confidence* berpengaruh terhadap jumlah *rule* ataupun *frequent item-set* yang dihasilkan.

Kata Kunci: *Data Mining*, Rekomendasi Produksi, *Association Rule*, CT-Pro.

1. PENDAHULUAN

PT. Contin Motosport adalah salah satu perusahaan yang bergerak di bidang produksi perlengkapan berkendara (*ridingware*), perusahaan ini berdiri sejak tahun 2006 silam. Perusahaan ini telah cukup berkembang walaupun pemasaran produknya belum sampai ke luar negeri, namun produk-produknya sudah cukup tersebar di Indonesia. PT. Contin Motosport telah banyak melayani transaksi pesanan produksi yang melibatkan banyak produk yang berbeda terutama jaket, karena jaket adalah produk unggulan dari perusahaan ini.

Permasalahan yang terjadi pada perusahaan ini adalah sering kali melakukan kesalahan produksi yang mengakibatkan menumpuknya jenis produk hasil produksi yang tidak sesuai dengan kebutuhan konsumen hal ini disebabkan oleh aturan produksi yang hanya berdasarkan perkiraan Kepala Produksi dan pemesanan secara langsung.

Seiring berjalannya waktu, aktivitas penjualan serta transaksi yang terjadi menghasilkan data yang banyak dan akan terus bertambah. Akan tetapi data tersebut tidak dimanfaatkan oleh pihak perusahaan secara maksimal, hanya sebatas laporan dan kemudian disimpan sebagai arsip. Dengan menggunakan *data mining*, data tersebut dapat digali hingga didapatkan nilai tambah berupa pengetahuan yang selama ini tidak diketahui secara manual [2].

Salah satu metode dalam *data mining* adalah *association rule* [3]. *Association rule* bertujuan untuk menunjukkan nilai asosiatif antara jenis-jenis produk yang dibeli oleh konsumen sehingga dapat di lihat sebuah pola berupa produk-produk apa saja yang sering dibeli secara bersamaan dalam sebuah transaksi penjualan, sehingga dapat dijadikan acuan untuk pengambilan keputusan dalam produksi jenis produk.

Sebagai solusi untuk pemecahan masalah pada perusahaan PT. Contin Motosport penulis akan mencoba menerapkan *data mining* terhadap data transaksi penjualan produk di perusahaan tersebut dengan menggunakan metode *Association Rules* dalam bentuk perangkat lunak sebagai alat bantu untuk menentukan nilai asosiatif antara jenis-jenis produk. Diharapkan dengan adanya perangkat lunak ini dapat membantu pihak perusahaan terutama bagian produksi untuk mengetahui pola pembelian konsumen, yang mana informasi tersebut dapat dimanfaatkan oleh pihak perusahaan sebagai dasar pertimbangan dan rekomendasi dalam menentukan jenis produk yang akan diproduksi.

1.1 *Data Mining*

Data mining merupakan proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai *database* besar [4].

Menurut Pramudiono [2], "*Data mining* adalah serangkaian proses untuk menggali nilai tambah berupa pengetahuan yang selama ini tidak diketahui secara manual dari suatu kumpulan data".

Menurut Larose [5], kemajuan luar biasa yang terus berlanjut dalam bidang *data mining* didorong oleh beberapa faktor, antara lain:

1. Pertumbuhan yang cepat dalam kumpulan data.
2. Penyimpanan data dalam *data warehouse*, sehingga seluruh perusahaan memiliki akses ke dalam *database* yang handal.
3. Adanya peningkatan akses data melalui navigasi web dan intranet.
4. Tekanan kompetensi bisnis untuk meningkatkan penguasaan pasar dalam globalisasi ekonomi.
5. Perkembangan teknologi perangkat lunak untuk *data mining* (ketersediaan teknologi).
6. Perkembangan yang hebat dalam kemampuan komputasi dan pengembangan kapasitas media penyimpanan.

1.1.1 Konsep *Data Mining*

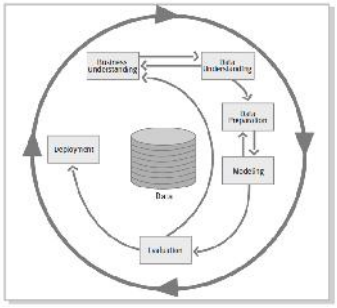
Data mining sangat diperlukan terutama dalam mengelola data yang sangat besar untuk memudahkan aktivitas *recording* suatu transaksi dan untuk proses *data warehousing* agar dapat memberikan informasi yang akurat bagi pengguna *data mining*. Alasan utama *data mining* sangat dibutuhkan dalam industri informasi karena tersedianya data dalam jumlah yang besar dan semakin besarnya kebutuhan untuk mengubah data tersebut menjadi informasi dan pengetahuan yang berguna karena sesuai fokus bidang ilmu ini yaitu melakukan kegiatan mengekstraksi atau menambang pengetahuan dari data yang berukuran atau berjumlah besar. Informasi inilah yang nantinya sangat berguna untuk pengembangan.

1.1.2 *Cross-Industry Standard Process for Data Mining* (*CRISP-DM*)

Cross-Industry Standard Process for Data Mining (*CRISP-DM*) yang dikembangkan tahun 1996 oleh analis dari beberapa industri seperti *Daimler Chrysler, SPSS dan NCR. CRISP-DM* menyediakan standar proses *data mining* sebagai strategi pemecahan masalah secara umum dari bisnis atau unit penelitian.

Dalam *CRISP-DM*, sebuah proyek *data mining* memiliki siklus hidup yang terbagi dalam enam fase. Keseluruhan fase berurutan yang ada tersebut bersifat adaptif. Fase berikutnya dalam urutan bergantung kepada keluaran dari fase sebelumnya. Hubungan antar fase digambarkan dengan panah. Sebagai contoh, jika proses berada pada fase *modelling*. Berdasar pada perilaku dan karakteristik model, proses mungkin harus kembali kepada fase *data preparation* untuk perbaikan lebih lanjut terhadap data atau berpindah maju kepada fase *evaluation*.

Enam fase yang ada dalam metode *CRISP-DM* tersebut dapat dilihat pada Gambar berikut ini:



Gambar 1 Fase metode *CRISP-DM*

Fase-fase dari *CRISP-DM* [5]:

1. *Business Understanding*
   a. Penentuan tujuan bisnis yang dilakukan oleh perusahaan dan kriteria sukses bisnis yang menjadi acuan berhasilnya perusahaan dalam menjalankan bisnis.
   b. Menjelaskan tujuan digunakannya metode *data mining* dalam bisnis yang dijalankan dan kriteria suksesnya metode *data mining*.
   c. Menilai situasi dari sumber daya yang dimiliki oleh perusahaan yang dapat mendukung proses *data mining*.
2. *Data Understanding*
   a. Mengumpulkan data.
   b. Mendeskripsikan data yang telah diperoleh.
   c. Mengevaluasi kualitas data.
   d. Jika diinginkan, pilih sebagian kecil grup data yang mungkin mengandung pola dari permasalahan.
3. *Data Preparation*
   a. Siapkan dari data awal, kumpulan data yang akan digunakan untuk keseluruhan fase berikutnya. Fase ini merupakan pekerjaan berat yang perlu dilaksanakan secara intensif.
   b. Pilih kasus dan variabel yang ingin dianalisis dan yang sesuai dengan analisis yang akan dilakukan.
   c. Lakukan perubahan pada beberapa variabel jika dibutuhkan.
   d. Siapkan data awal sehingga siap untuk perangkat pemodelan.
4. *Modelling*
   a. Pilihan dan aplikasikan teknik pemodelan yang sesuai.
   b. Kalibrasi aturan model untuk mengoptimalkan hasil.
   c. Perlu diperhatikan bahwa beberapa teknik mungkin untuk digunakan pada permasalahan *data mining* yang sama.
   d. Jika diperlukan, proses dapat kembali ke fase pengolahan data untuk menjadikan data ke dalam bentuk yang sesuai dengan spesifikasi kebutuhan teknik *data mining* tertentu.
5. *Evaluation*
   a. Mengevaluasi satu atau lebih model yang digunakan dalam fase pemodelan untuk

mendapatkan kualitas dan efektivitas sebelum disebarkan untuk digunakan.
   b. Menetapkan apakah terdapat model yang memenuhi tujuan pada fase awal.
   c. Menentukan apakah terdapat permasalahan penting dari bisnis atau penelitian yang tidak tertangani dengan baik.
   d. Mengambil keputusan berkaitan dengan penggunaan hasil dari *data mining*.
6. *Deployment*
   a. Menggunakan model yang dihasilkan. Terbentuknya model tidak menandakan telah terselesaikannya proyek.
   b. Contoh sederhana penyebaran: pembuatan laporan.
   c. Contoh kompleks penyebaran: penerapan proses *data mining* secara paralel pada departemen lain.

1.2 *Association Rule*

Association rule adalah salah satu teknik utama atau prosedur dalam *market basket analysis* untuk mencari hubungan antar-*item* dalam suatu *data set* dan menampilkan bentuk *associaiton rule* [6]. *Association rule* akan menemukan pola tertentu untuk mengasosiasikan data yang satu dengan data yang lain.

Untuk mencari *association rule* dari suatu kumpulan data, tahap pertama yang harus dilakukan adalah mencari *frequent itemset*. *Frequent itemset* adalah sekumpulan item yang sering muncul secara bersamaan. Setelah semua pola *frequent item* ditemukan, barulah mencari aturan asosiatif atau aturan keterkaitan yang memenuhi syarat yang telah ditentukan.

Sebagai contoh konsumen biasanya akan membeli kopi dan susu yang ditunjukkan sebagai berikut:

Kopi → susu [*support* = 2%, *confidence* = 60%]

Association rule diperlukan suatu variabel ukuran yang ditentukan sendiri oleh *user* untuk menentukan batasan sejauh mana atau sebanyak apa *output* yang diinginkan *user*. *Support* dan *confidence* adalah sebuah ukuran kepercayaan dan kegunaan suatu pola yang telah ditemukan, Nilai *support* 2% menunjukkan bahwa keseluruhan dari total transaksi konsumen membeli kopi dan susu secara bersamaan yaitu sebanyak 2%. Sedangkan *confidence* 60%, yaitu menunjukkan bila konsumen membeli kopi dan pasti membeli susu sebesar 60%.

Dalam menentukan suatu *association rule*, terdapat suatu *interestingness measure* (ukuran kepercayaan) yang didapat dari hasil pengolahan data dengan perhitungan tertentu. Umumnya ada dua ukuran, yaitu:

a. *Support*, suatu ukuran yang menunjukkan seberapa besar tingkat dominasi suatu *item/itemset* dari keseluruhan transaksi. Ukuran ini menentukan apakah suatu *item/itemset* layak untuk dicari *confidence*-nya (misal, dari keseluruhan transaksi yang ada, seberapa besar tingkat dominasi suatu *item* yang menunjukkan bahwa *item* A dan *item* B dibeli bersamaan).
b. *Confidence*, suatu ukuran yang menunjukkan hubungan antara 2 *item* secara *conditional* (misal, menghitung kemungkinan seberapa sering *item* B dibeli oleh konsumen jika konsumen tersebut membeli sebuah *item* A).

Kedua ukuran ini nantinya berguna dalam menentukan kekuatan suatu pola dengan membandingkan pola tersebut dengan nilai minimum kedua parameter tersebut yang ditentukan oleh pengguna. Bila suatu pola memenuhi kedua nilai minimum parameter yang sudah ditentukan sebelumnya, maka pola tersebut dapat disebut sebagai *interesting rule* atau *strong rule*.

1.2.1 Metodologi Dasar *Association Rule*

Metodologi dasar *Association Rule* terbagi menjadi dua tahap, yaitu [6]:

1. Analisis pola frekuensi tinggi

   Tahap Tahap ini mencari pola *item* yang memenuhi syarat minimum dari nilai *support* dalam *database*. Menurut *Larose*, kita bebas menentukan nilai minimum *support* (minsup) dan minimum *confidence* (mincof) sesuai kebutuhan [5]. Sebagai contoh, bila ingin menemukan data-data yang memiliki hubungan asosiasi yang kuat, minsup dan mincof-nya bisa diberi nilai yang tinggi. Sebaliknya, bila ingin melihat banyaknya variasi data tanpa terlalu mempedulikan kuat atau tidaknya hubungan asosiasi antara item-nya, nilai minsup dan mincof-nya dapat diisi rendah [7]. Nilai *support* sebuah *item* diperoleh dengan rumus berikut.

$$\text{Support (A)} = \frac{\text{Jumlah transaksi yang mengandung A}}{\text{Jumlah total transaksi}} \text{x} 100\% \quad (1)$$

   Persamaan 1 menjelaskan bahwa nilai *support* didapat dengan cara membagi jumlah transaksi yang mengandung *item* A (satu *item*) dengan jumlah total seluruh transaksi.

   Sementara itu, nilai support dari 2 *item* diperoleh dari rumus berikut

$$\text{Support (A,B)} = P(A \cap B)$$

$$\text{Support (A,B)} = \frac{\text{Jumlah transaksi yang mengandung A dan B}}{\text{Jumlah total transaksi}} \text{x} 100\% \quad (2)$$

   Persamaan 2 menjelaskan bahwa nilai *support item set* didapat dengan cara membagi jumlah transaksi yang mengandung *item* A dan *item* B (*item* pertama bersamaan dengan *item* yang lain) dengan jumlah total seluruh transaksi.

2. Pembentukan aturan asosiasi

   Dalam *antecedent* (pendahulu) dan *consequent* (pengikut) serta memenuhi syarat minimum untuk *confidence* aturan asosiatif A → B.

   Misalkan D adalah himpunan transaksi, di mana setiap transaksi T dalam D merepresentasikan himpunan *item* yang berada dalam I. I adalah

himpunan *item* yang dijual. Misalkan kita memilih himpunan *item* A dan himpunan *item* lain B, kemudian aturan asosiasi akan berbentuk:

Jika A, maka B (A→B)

Antecedent A dan *consequent* B merupakan *subset* dari I, dan A dan B di mana aturan:

Jika A, maka B

Tidak berarti

Jika B, maka A

Sebuah *item set* adalah himpunan *item* yang ada dalam I, dan i *item set*. Frekuensi *item set* merupakan *itemset* yang memiliki frekuensi kemunculan lebih dari nilai minimum yang telah ditentukan.

Nilai *confidence* dari aturan A→B diperoleh dari rumus berikut.

$$\text{Confidence} = \frac{\text{Jumlah transaksi yang mengandung A dan B}}{\text{Jumlah transaksi yang mengandung A}} \quad (3)$$

1.3 Algoritma *CT-Pro*

Algoritma ini merupakan salah satu algoritma pengembangan dari *FP-Growth*. Perbedaannya adalah di mana *FP-Growth* membuat *FP-Tree* sedangkan *CT-Pro* membuat *Compressed FP-Tree (CFP-Tree)* [8]. Pada tahap *mining* algoritma *CT-PRO* juga menggunakan pendekatan *bottom-up* dimana *item* pada *item* tabel dan *CFP-Tree* dilakukan *scan* dari jumlah terkecil hingga terbesar.

Algoritma *CT-Pro* memiliki tiga tahap yaitu [8]:

1. Mencari *Frequent Item*, pada tahap ini terjadi proses-proses sebagai berikut:
   a. Dari *dataset* yang ada, dilakukan seleksi berdasarkan *minimum support* yang ditentukan sehingga menghasilkan *frequent item.*
   b. Dari *frequent item* yang telah terbentuk, dihitung frekuensi kemunculan setiap *item* sehingga menghasilkan *Global Item tabel*.
2. Membangun *CFP-Tree*, pada tahap ini terjadi proses-proses sebagai berikut:
   a. *Frequent item* yang telah didapatkan, diurutkan berdasarkan *Global item table* yang ada secara menurun (diurutkan mulai dari item berfrekuensi terbesar hingga terkecil).
   b. Dengan *frequent item* yang telah terurut ini dibentuk *Global CFP-Tree*, berikut adalah aturan pembentukan *Global CFP-Tree*:
      1) *CFP-Tree* terdiri dari *tree* yang memiliki *root* yang mewakili indeks dari *item* dengan tingkat kemunculan tertinggi dan kumpulan *subtree* sebagai anak dari *root*.
      2) Jika I = {i1,i2, …, ik} adalah kumpulan dari *frequent item* dalam transaksi, *item* dalam transaksi akan dimasukkan ke dalam *CFP-Tree* dimulai dari *root subtree* yang merupakan i1 dalam *item* tabel.
      3) *Root* dari *CFP-Tree* merupakan *level*-0 dari *tree*.
      4) Setiap *node* dalam CFP-Tree memiliki empat *field* utama yakni *item-id, parent-id, Count* yang merupakan jumlah *item* pada *node* tersebut, dan *level* yang menunjukkan struktur data *tree* pada *node* tersebut dimulai dari *item* yang terdapat pada *item* tabel dengan *level* yang terdapat pada *CFP-Tree*.
3. *Mining*, pada tahap ini terjadi proses-proses sebagai berikut:
   a. Pada tahap *mining* ini, algoritma *CT-Pro* bekerja dengan melakukan *bottom-up mining* sehingga *Global item table* diurutkan mulai dari *item* berfrekuensi terkecil hingga terbesar.
   b. Untuk setiap *item* yang terdaftar pada *Global item table* yang telah diurutkan, dilakukan pencarian *node* yang berkaitan dengan *item* tersebut pada *Global CFP-Tree*. Dari semua *node* yang ditemukan untuk setiap *item* inilah yang disebut dengan *Local frequent item* dan digunakan untuk membuat *Local item table*.
   c. Pada pembuatan *Local item table* ini juga dilakukan berdasarkan jumlah minimum *support* yang telah ditentukan.
   d. Setelah itu, dibuat *Local CFP-Tree* berdasarkan *Local item table* yang terbentuk. Aturan pembentukan *Local* CFP-Tree sama dengan pembentukan *Global CFP-Tree*, yang membedakan adalah pada *Global CFP-Tree* yang digunakan dalam pembentukan *tree*-nya adalah *Global item table* yang terbentuk dari *Global item table* data sedangkan pada *Local CFP-Tree* yang digunakan dalam pembentukan *tree*-nya adalah *Local item table* yang terbentuk dari *Local frequent item.*
   e. Dari *Local CFP-Tree* dibentuk *frequent pattern* sesuai dengan item yang di-*mining*.
   f. Dari *frequent pattern* dihitung masing-masing *item* yang memenuhi dihitung *confidence*-nya. Apabila memenuhi *minimum confidence* maka masing-masing *item* yang bersangkutan dijadikan sebagai *knowledge*.

2. ISI PENELITIAN

2.1 Proses *Data Mining*

1. Ekstraksi Data

Ekstraksi data adalah proses pengambilan data dari sumber data untuk melanjutkan proses pengolahan data ke tingkat selanjutnya ataupun untuk menyimpan data hasil ekstrak tersebut. Data yang berasal dari file yang berformat .xlsx di ekstrak ke dalam format *sql*, kemudian disimpan dalam sebuah *database* agar memudahkan proses pengolahan data.

2. Pemilihan Atribut

Proses ini dilakukan untuk menghilangkan beberapa atribut yang tidak diperlukan agar proses *mining* menjadi lebih cepat. Berikut merupakan

atribut-atribut yang dipilih dan akan digunakan dalam proses mining.

Tabel 1 Pemilihan Atribut Data Transaksi

Atribut	Dipilih	Alasan
No. Faktur	Ya	Sebagai acuan produk-produk yang dibeli dalam suatu transaksinya
Tanggal	Tidak	Tidak diperlukan
Customer	Tidak	Tidak diperlukan
Status	Tidak	Tidak diperlukan
Kode *Type*	Tidak	Tidak diperlukan
Type	Ya	Untuk menemukan hubungan antar jenis produk agar diperoleh *rule* untuk penentuan produksi berdasarkan jenisnya
Size	Tidak	Tidak diperlukan
QTY	Tidak	Tidak diperlukan

3. Pembersihan Data

Pembersihan data (*cleaning*) dilakukan untuk menghilangkan *noise* dan menghilangkan data yang tidak relevan. Aturan yang digunakan dalam proses *cleaning* yaitu jika dalam satu transaksi hanya terdapat satu jenis produk, maka transaksi tersebut akan dibersihkan dari *data-set*. Proses pembersihan data di antaranya:

a. Memindai setiap transaksi pada *data-set* setelah pemilihan atribut.
b. Jika transaksi tersebut mengandung lebih dari satu jenis produk (*type*) maka akan dimasukan dalam proses selanjutnya.
c. Jika transaksi tersebut hanya mengandung satu jenis produk (*type*) saja maka akan dibersihkan.

Tabel 2 Dataset Sebelum *Cleaning*

No. Faktur	*Type*
140102-1	Rompi Blaster
140102-1	Centour Pants
140102-1	Rompi Cavalier
140102-2	Speedtrap
140102-3	Corble Man
140102-3	Speedtrap
140102-3	Rompi Cavalier
140102-4	Kamerad 11 jeans man
140102-5	Rompi Blaster
140103-6	Neo Lunatic
140103-6	Centour Pants

Tabel 3 Dataset Setelah *Cleaning*

No. Faktur	*Type*
140102-1	Rompi Blaster
140102-1	Centour Pants
140102-1	Rompi Cavalier
140102-3	Corble Man
140102-3	Speedtrap
140102-3	Rompi Cavalier
140103-6	Neo Lunatic
140103-6	Centour Pants

4. Pemodelan

Merupakan tahap untuk membuat model atau desain sistem yang akan dibangun dengan mengimplementasikan algoritma *CT-PRO* dengan menerapkan struktur data *CFP-Tree* pada data uji.

A. Menghitung Frekuensi Kemunculam Tiap Jenis Produk

Untuk tahapan mencari frekuensi item sebagai berikut:

1. Pindai setiap jenis produk pada tabel transaksi penjualan yang telah dilakukan pemilihan atribut dan pembersihan, kemudian kelompokkan berdasarkan jenis produk.
2. Hitung jumlah jenis produk berdasarkan kelompoknya, kemudian masukan ke tabel Frekuensi jenis produk.

Tabel 4 Frekuensi Kemunculan Setiap Jenis Produk

Type	*Support Count*
Speedtrap	26
Tank Bag Skipper	21
Hip Bag Flanker	21
Neo Lunatic	21
Centour Pants	20
Kamerad 11 jeans man	18
Rompi Cavalier	16
Corble Man	15
Hip Bag Baru 2 (sidewinder)	10
Kamerad 12 Jeans Woman	8
Rompi Blaster	7
Corble Women	7
Kamerad 21 Camo Man	6
Rain Cover Tankbag	6
Rompi Reign	6
Rompi Reign Denim	5
Magnet	4
Vivere Bella	4
Zirkon	3

B. Membangun *Global Item Tabel*

Langkah-langkah dalam membuat *Global item table* adalah sebagai berikut:

a. Menentukan *minimum support* untuk membatasi kemunculan jenis produk dalam satu transaksi. Dalam kasus ini *minimum support* yang digunakan yaitu 10 dan *minimum confidence* = 60% yang nantinya digunakan sebagai acuan pembangkitan *rule*. Hal ini dimaksudkan agar calon *rule* yang diperoleh cukup banyak sehingga dapat dilihat jenis produk apa saja yang sering dibeli secara bersamaan agar dapat dijadikan rekomendasi produksi. Oleh karena itu, batasan frekuensi jenis produk yang muncul harus sebanyak ≥ 10, jika terdapat produk yang kemunculannya < 10 maka akan dihilangkan. Berikut ini data produk yang sudah dilakukan

pemindaian hasilnya dapat dilihat pada tabel berikut.

Tabel 5 Penghapusan Data Tidak *Frequent*

Type	Frekuensi
Speedtrap	26
Tank Bag Skipper	21
Hip Bag Flanker	21
Neo Lunatic	21
Centour Pants	20
Kamerad 11 jeans man	18
Rompi Cavalier	16
Corble Man	15
Hip Bag Baru 2 (sidewinder)	10

b. Setelah diseleksi jenis produk yang frekuensi kemunculannya memenuhi nilai *minimum support*, kemudian dibentuk *Global item table*. Setiap jenis produk akan diberikan *Id* secara *ascending* sesuai dari frekuensi terbesar ke terkecil. Hasilnya dapat dilihat pada tabel dibawah ini:

Tabel 6 *Global Item Table*

Type	*Support Count*	*Id Global Item*
Speedtrap	26	1
Tank Bag Skipper	21	2
Hip Bag Flanker	21	3
Neo Lunatic	21	4
Centour Pants	20	5
Kamerad 11 jeans man	18	6
Rompi Cavalier	16	7
Corble Man	15	8
Hip Bag Baru 2 (sidewinder)	10	9

Setelah ditentukan *Global item table*, dapat ditentukan jenis produk yang *frequent*. Jenis produk dari data transaksi kemudian diurutkan dari *id global item* terkecil ke terbesar pada setiap No. Fakturnya. Berikut adalah contoh hasil pengurutan jenis produk yang *frequent.*

Tabel 7 Data Transaksi Dengan Jenis Produk yang *Frekuent*

No. Faktur	*Type*
140102-1	Corble Man
140102-1	Kamerad 11 jeans man
140102-1	Hip Bag Baru 2 (sidewinder)
140102-1	Tank Bag Skipper
140102-1	Hip Bag Flanker
140102-1	Neo Lunatic
140102-1	Speedtrap
140102-1	Centour Pants
140102-1	Rompi Cavalier
140102-3	Corble Man
140102-3	Speedtrap
140102-3	Rompi Cavalier

C. *Mapping* Data

Proses memetakan data dari terhadap id *global* item, dimana nama produk digantikan oleh *id global item*. Berikut adalah contoh hasil *mapping* data transaksi:

Tabel 8 *Mapping* Data

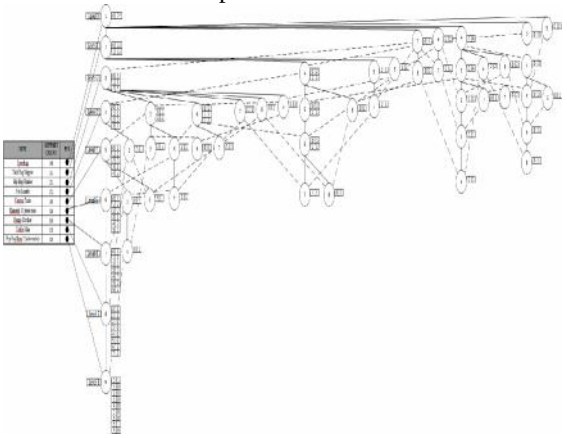
No. Faktur	Jenis Produk
140102-1	1
140102-1	2
140102-1	3
140102-1	4
140102-1	5
140102-1	6
140102-1	7
140102-1	8
140102-1	9
140102-3	1
140102-3	7
140102-3	8

D. Membangun *CFP-Tree*

Proses *CFP-Tree* memasukkan setiap data transaksi kedalam tree. Tahapan membangun *CFP-Tree* yaitu:

a. Dimulai dari Jenis Produk yang pertama kali muncul dari No.Faktur pertama pada tabel *mapping*, Jenis Produk tersebut akan menjadi *currNode*.
b. Tambahkan *Count* sebanyak 1 pada Jenis Produk yang pertama muncul sesuai dengan levelnya.
c. Jika *currNode* sudah memiliki anak, dan Jenis Produk yang muncul sama dengan anak tersebut maka tambahkan *Count* sebanyak 1 pada level yang sama dengan *currNode*.
d. Jika *currNode* sudah memiliki anak, tetapi data yang muncul berbeda dengan anak dari *currNode*, buat *node* baru dan tambahkan *Count* sebanyak 1 pada level yang sama dengan *currNode*, hubungkan *node* baru tersebut pada *currNode* melalui *linkNode*.

Berikut adalah hasil pembentukan Global CFP-Tree.



Gambar 2 *Global CFP-Tree*

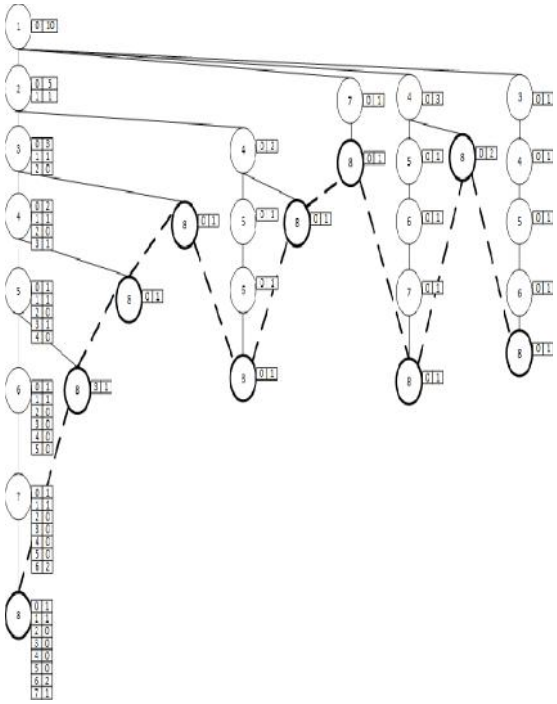
E. Mencari *Frequent Pattern*

Berikut adalah langkah-langkah untuk mencari *frequent Pattern*:

a. Dimulai dengan indeks dengan *support count* terkecil (x) dari *Global item table*.
b. Lakukan penelusuran *node* yang berkaitan dengan indeks x dengan menghitung *Count* sebanyak 1 pada setiap *node* yang dikunjungi mulai dari *Path* sampai ke *root* untuk mencari *node* yang memenuhi minimum *support Count.*
c. Masukan setiap jenis produk yang memenuhi *minimum support* ke *Local item table.*
d. Berikan Indeks baru pada *node* yang ada *di Local item table* secara *ascending*.
e. Setelah *Local item table* terbentuk lakukan pemetaan indeks baru terhadap *node* yang memenuhi *minimum support* sesuai dengan indeks di *Local item table.*
f. Setelah dilakukan pemetaan lakukan pembentukan *Local CFP-Tree* dengan aturan yang sama saat pembentukan *Global CFP-Tree*
g. lakukan pembentukan *Local CFP-Tree projection* yaitu di mana x akan menjadi *root* dan indeks yang ada di *Local CFP-Tree* menjadi anaknya.
h. Lakukan penelusuran di *Local CFP-Tree* ulang secara *bottom-up* dari *node* ke-i dengan menghitung *Count* dari setiap *Node*, jika *node* k memenuhi *minimum support* jadikan k menjadi anak dari *node* ke-i.

Contoh mencari *frequent item* untuk jenis produk (*type*) 8.

Cari jalur (*Node*) pada *Global* CFP-Tree yang hanya berakhiran di jenis produk 8 secara *bottom-up* yaitu penelusuran mulai dari *Path* sampai *root*, untuk jelasnya dapat dilihat pada gambar berikut:



Gambar 3 *CFP-Tree* Berakhiran Jenis Produk 8

Dari gambar *CFP-Tree* akhiran indeks 8 di atas, lalu dilakukan penelusuran dengan cara mencari indeks apa saja yang berhubungan dengan indeks 8 dan memasukkannya ke dalam tabel Hasil Penelusuran. Berikut adalah hasil penelusuran untuk indeks 8 pada *Global CFP-Tree* yang dapat dilihat di bawah ini:

Tabel 9 Hasil Penelusuran untuk Indeks 8

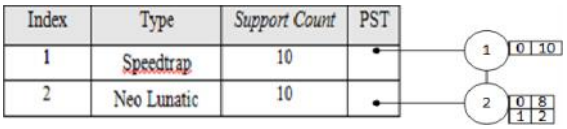
No	*Path*
1	8
2	8, 7
3	8, 7
4	8, 7, 6, 5, 4, 3, 2
5	8, 7, 6, 5, 4, 3, 2, 1
6	8, 5, 4
7	8, 4, 3, 2, 1
8	8, 3, 2 ,1
9	8, 6, 5, 4, 2, 1
10	8, 4, 2, 1
11	8, 7, 1
12	8, 7, 6, 5, 4, 1
13	8, 4, 1
14	8, 4, 1
15	8, 6, 5, 4, 3, 1

Dari tabel di atas didapat jumlah *support count* dari setiap *node* yaitu 1(10), 2(6), 3(5), 4(10), 5(6), 6(5), dan 7(6). Karena *minimum support*-nya adalah 10 maka untuk *node* 1(10) dan 4(10) akan dimasukkan kedalam *local item table*. Selanjutnya *node* yang memenuhi *minimum support* di masukkan ke *Local item table* dan diberikan indeks baru di *Local item table* yaitu penomoran secara *ascending* dari jumlah *support count* terbesar ke terkecil. Berikut merupakan *Local item table* yang terbentuk:

Tabel 10 *Local Item Table* untuk Indeks 8

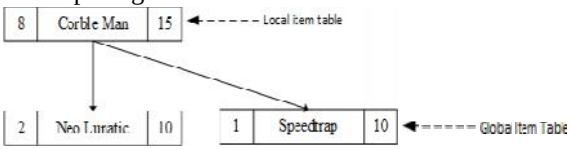
Indeks	*Node*	*Type*	*Support Count*
1	1	Speedtrap	10
2	4	Neo Lunatic	10

Selanjutnya melakukan pembentukan *Local CFP-Tree*. Aturan pembentukan *Local CFP-Tree* sama dengan pembentukan *Global CFP-Tree*, yang membedakan adalah pada *Global CFP-Tree* yang digunakan dalam pembentukan *tree-nya* adalah *Global item table* yang terbentuk dari *Global item table* data sedangkan pada *Local CFP-Tree* yang digunakan dalam pembentukan *tree-nya* adalah *Local item table* yang terbentuk dari *Local frequent item*. Berikut adalah *Local CFP-Tree* dari indeks 8:



Gambar 4 Local CFP-Tree Untuk Indeks 8

Setelah *Local CFP-Tree* indeks 8 terbentuk, langkah selanjutnya adalah pembentukan *Local CFP-Tree projection* dimana indeks 8 akan menjadi *root* dan setiap indeks di *Local item table* akan menjadi anak dari indeks 8. Untuk menemukan *Local frequent pattern tree*, lakukan penelusuran di *Local CFP-Tree* indeks 8 mulai dari *Path sampai root* secara *bottom-up*, untuk lebih jelasnya dapat dilihat pada gambar berikut:



Gambar 5 *Local CFP-Tree Projection* Indeks 8

Dari *Local CFP-Tree projection* maka didapatkan *item frequent* untuk indeks 8 dengan jenis produk (*type*) Corble Man adalah sebagai berikut: (8,2:10) dan (8,1:10).

Lakukan hal ini secara menurun dari jenis produk/indeks ke 9 hingga jenis produk/indeks ke 1 untuk mengetahui *local CFP-Tree*-nya.

F. *Generate Rule*

Dari *frequent Pattern* yang didapat, akan di-*generate* untuk mendapatkan *rule* dengan *minimum confidence* = 60%. Dihitung dengan persamaan sebagai berikut:

$$Confidence = P(B \mid A) = \frac{Jumlah\ transaksi\ yang\ mengandung\ A\ dan\ B}{Jumlah\ transaksi\ yang\ mengandung\ A}\ X\ 100\% \quad (4)$$

Berikut adalah hasil generate Rule yang memenuhi minimum confidence > 60%:

Tabel 11 Hasil *Rule*

Rules	*Rules Type*	*Confidence*
8 → 1	Corble Man → Speedtrap	(10/15) * 100% = 66.67%
8 → 4	Corble Man → Neo Lunatic	(10/15) * 100% = 66.67%
6 → 1	Kamerad 11 Jeans Man → Speedtrap	(11/18) * 100% = 61.11%
6 → 4	Kamerad 11 Jeans Man → Neo Lunatic	(12/18) * 100% = 66.67%
5 → 4	Centour Pants → Neo Lunatic	(14/20) * 100% = 70%
5 → 1	Centour Pants → Speedtrap	(12/20) * 100% = 60%
4 → 1	Neo Lunatic → Speedtrap	(16/21) * 100% = 76.19%
3 → 2	Hip Bag Flanker → Tank Bag Skiper	(14/21) * 100% = 66.67%

Berdasarkan informasi *rules* pada tabel 11 di atas maka dapat diperjelas lagi sebagai rekomendasi penentuan produksi jenis produk berdasarkan pola pembelian konsumen, untuk lebih jelasnya dapat dilihat pada tabel berikut:

Tabel 12 Hasil Pembentukan *Rules*

Rules	*Rules Type*	**Penjelasan *Rules***
8 → 1	Corble Man → Speedtrap	Berdasarkan kebiasaaan konsumen yang membeli *type* produk "Corble Man", sering bersamaan dengan *type* produk "Speedtrap". Maka sebagai rekomendasi jika perusahaan akan memproduksi *type* produk "Corble Man" maka *type* produk "Speedtrap" lebih baik diproduksi juga.
8 → 4	Corble Man → Neo Lunatic	Berdasarkan kebiasaaan konsumen yang membeli *type* produk "Corble Man", sering bersamaan dengan *type* produk "Neo Lunatic". Maka sebagai rekomendasi jika perusahaan akan memproduksi *type* produk "Corble Man" maka *type* produk "Neo Lunatic" lebih baik diproduksi juga.
6 → 1	Kamerad 11 Jeans Man → Speedtrap	Berdasarkan kebiasaaan konsumen yang membeli *type* produk "Kamerad 11 Jeans", sering bersamaan dengan *type* produk "Speedtrap". Maka sebagai rekomendasi jika perusahaan akan memproduksi *type* produk "Kamerad 11 Jeans" maka *type* produk "Speedtrap" lebih baik diproduksi juga.

Rules	*Rules Type*	Penjelasan *Rules*
6 → 4	Kamerad 11 Jeans Man → Neo Lunatic	Berdasarkan kebiasaaan konsumen yang membeli *type* produk "Kamerad 11 Jeans Man", sering bersamaan dengan *type* produk "Neo Lunatic". Maka sebagai rekomendasi jika perusahaan akan memproduksi *type* produk "Kamerad 11 Jeans Man" maka *type* produk "Neo Lunatic" lebih baik diproduksi juga.
5 → 4	Centour Pants → Neo Lunatic	Berdasarkan kebiasaaan konsumen yang membeli *type* produk "Centour Pants", sering bersamaan dengan *type* produk "Neo Lunatic". Maka sebagai rekomendasi jika perusahaan akan memproduksi *type* produk "Centour Pants" maka *type* produk "Neo Lunatic" lebih baik diproduksi juga.
5 → 1	Centour Pants → Speedtrap	Berdasarkan kebiasaaan konsumen yang membeli *type* produk "Centour Pants", sering bersamaan dengan *type* produk "Speedtrap". Maka sebagai rekomendasi jika perusahaan akan memproduksi *type* produk "Centour Pants" maka *type* produk "Speedtrap" lebih baik diproduksi juga.
4 → 1	Neo Lunatic → Speedtrap	Berdasarkan kebiasaaan konsumen yang membeli *type* produk "Neo Lunatic", sering bersamaan dengan *type* produk "Speedtrap". Maka sebagai rekomendasi jika perusahaan akan memproduksi *type* produk "Neo Lunatic" maka *type* produk "Speedtrap" lebih baik diproduksi juga.
3 →2	Hip Bag Flanker → Tank Bag Skipper	Berdasarkan kebiasaaan konsumen yang membeli *type* produk "Tank Bag Skipper", sering bersamaan dengan *type* produk "Speedtrap". Maka sebagai rekomendasi jika perusahaan akan memproduksi *type* produk "Tank Bag Skipper" maka *type* produk "Speedtrap" lebih baik diproduksi juga.

Maka berdasarkan penjelasan hasil akhir dari *rules* pada tabel 12 di atas dapat disimpulkan jenis-jenis produk (*type)* sebagai rekomendasi yang harus diproduksi adalah sebagai berikut:

Tabel 13 Rekomendasi Jenis (*type)* Produk untuk Diproduksi

Rekomendasi *Type*
Corble Man
Speedtrap
Neo Lunatic
Kamerad 11 Jeans Man
Centour Pants
Hip Bag Flanker
Tank Bag Skiper

2.2 Pengujian Hasil

Pengujian ini menguji perangkat lunak yang telah dibangun apakah menghasilkan data yang diinginkan dan sesuai dengan hasil penerapan metode *association rule* dengan algoritma CT-Pro pada bab 3. Pengujian akan dilakukan dengan membandingkan hasil *rule* dari perhitungan manual dengan hasil *rule* dari aplikasi *data mining* yang telah dibangun. Nilai *minimum support* yang akan di uji adalah 10 dan nilai *minimum confidence* yang akan di uji sebesar 60%. Berikut adalah hasil perhitungan manual metode *association rule* dengan pendekatan algoritma CT-Pro.

Tabel 14 Hasil Pengujian Perhitungan Manual

Rules Type	*Confidence*
Corble Man → Speedtrap	(10/15) * 100% = 66.67%
Corble Man → Neo Lunatic	(10/15) * 100% = 66.67%
Kamerad 11 Jeans Man → Speedtrap	(11/18) * 100% = 61.11%
Kamerad 11 Jeans Man → Neo Lunatic	(12/18) * 100% = 66.67%
Centour Pants → Neo Lunatic	(14/20) * 100% = 70%
Centour Pants → Speedtrap	(12/20) * 100% = 60%
Neo Lunatic → Speedtrap	(16/21) * 100% = 76.19%
Hip Bag Flanker → Tank Bag Skiper	(14/21) * 100% = 66.67%

Dari hasil pengujian dengan perhitungan manual didapat 8 rule yang dihasilkan.

Berikut adalah *rule* yang dihasilkan oleh aplikasi *data mining* yang dibangun dengan nilai *minimum support* sebesar 10 dan nilai *minimum confidence* sebesar 60%.

Rule	Support	Confidence
Neo Lunatic -> {Speedtrap}	16	76.19%
Tank Bag Skipper -> {Hip Bag Fla...	14	66.67%
Centour Pants -> {Speedtrap}	12	60%
Centour Pants -> {Neo Lunatic}	14	70%
Kamerad 11 jeans man -> {Speed...	11	61.11%
Kamerad 11 jeans man -> {Neo L...	12	66.67%
Corble Man -> {Speedtrap}	10	66.67%
Corble Man -> {Neo Lunatic}	10	66.67%

Gambar 6 Hasil *Rule* Aplikasi *Data Mining*



Gambar 7 Hasil Rekomendasi Jenis Produk Diproduksi

Dari hasil pengujian 1 menggunakan aplikasi *data mining* yang telah dibangun didapat 8 *rule* yang dihasilkan.

Jika melihat hasil pengujian perhitungan manual dan hasil pengujian perangkat lunak, dapat disimpulkan bahwa hasil perhitungan dari aplikasi yang dibangun telah sesuai dengan hasil perhitungan secara manual karena data *rule* yang dihasilkan oleh aplikasi dan data *rule* yang dihasilkan dengan perhitungan manual menunjukkan nilai yang sama.

Berdasarkan hasil pengujian, dapat disimpulkan bahwa aplikasi *data mining* yang telah dibangun telah sesuai dengan perhitungan manual karena aplikasi yang telah dibangun menunjukkan hasil *rule* yang sama dengan menggunakan perhitungan manual. Oleh karena itu aplikasi *data mining* ini dapat digunakan oleh pihak PT. Contin Motosport sebagai alat bantu untuk mengetahui keterkaitan antar produk dan rekomendasi produksi jenis produk yang dapat tepat.

3. PENUTUP

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan pada aplikasi *data mining* untuk rekomendasi produksi jenis produk menggunakan metode *association rule* dan algoritma *CT-Pro*, maka dapat diambil kesimpulan sebagai berikut:

1. Aplikasi yang dibangun dapat meminimalisir menumpuknya hasil produksi jenis produk karena informasi yang dihasilkan cukup akurat dengan fakta di lapangan.
2. Aplikasi yang dibangun dapat membantu Kepala Produksi dalam memberikan informasi dan memberikan rekomendasi produksi jenis produk yang baik untuk diproduksi selanjutnya.

Dengan demikian aplikasi yang dibangun telah sesuai dan mencapai tujuan dari penelitian ini.

Adapun saran untuk pengembangan aplikasi rekomendasi produksi ini lebih lanjut yaitu:

1. Pengembangan dalam memberikan rekomendasi produksi dengan *output* yang lebih inovatif sehingga lebih memudahkan dan menyenangkan Kepala Produksi.
2. Menyesuaikan aplikasi dengan perkembangan teknologi di PT. Contin Motosport, misalnya jika Contin Motosport telah menggunakan *Database* dalam rekap data penjualanya maka aplikasi bisa disesuaikan dengan menggunakan data penjualannya yang sebelumnya melalu proses *import data* menjadi pemilihan data dengan data yang tersambung *database* perusahaan sehingga Kepala Produksi langsung memilih data penjualan tanggal berapa yang akan dijadikan acuan dalam menentukan produksi selanjutnya.

4. DAFTAR PUSTAKA

[1] P. Duncan and Hollander, Manajemen Toko Eceran, Jakarta: Balai Aksara, 1981.

[2] I. Pramudiono, Pengantar *Data Mining* : Menambang Permata Pengetahuan di Gunung Data, 2007.

[3] B. Santoso, *Data Mining*: Teknik Pemanfaatan Data Untuk Keperluan Bisnis, Yogyakarta: Graha Ilmu, 2007.

[4] A. Ramon, T. Mata and C. K. Pailine, Dasar-dasar Database Relational, Jakarta: Airlangga, 2007.

[5] D. T. Larose, *DISCOVERING KNOWLEDGE IN DATA : An Introduction to DATA MINING*, Hoboken, New Jersey: Wiley Interscience, 2005.

[6] J. Santoni, Implementasi Data Mining Dengan Metode *Market Basket Analysis*, Teknologi Informasi dan Pendidikan,, 2012.

[7] Kusrini and E. T. Luthfi, in *Algoritma Data Mining*, Yogyakarta, Andi, 2009.

[8] Y. Sucahyo and R. Gopalan, *CT-PRO: A Bottom-Up Non Recursive Frequent Itemset Mining Algorithm Using Compressed FP-Tree Data Structure*, 2004.