

# Analisis Consumer Behaviour Pada Toko Retail Dengan Metode APRIORI-SD

Nathaniel Edward, Rolly Intan, Alvin Nathaniel Tjondrowiguno  
Program Studi Informatika Fakultas Teknologi Industri Universitas Kristen Petra  
Jl. Siwalankerto 121 – 131 Surabaya 60236  
Telp. (031) – 2983455, Fax. (031) - 8417658

Email: nathaniel.edwards96@gmail.com, rintan@petra.ac.id, alvin.nathaniel@petra.ac.id

## ABSTRAK

Toko ritel perlu berkembang terutama di era digital di mana *e-commerce* menjadi semakin umum dan kebanyakan orang lebih suka kenyamanan dari *e-commerce*. Salah satu keuntungan terbesar dari *e-commerce* "baru" adalah mereka membangun model bisnis mereka di atas dasar pemrosesan data, sedangkan toko ritel yang lebih tua tidak. Pengembangan data mining dan pembelajaran mesin mendorong model bisnis yang lebih tua untuk berbuat lebih baik.

Jurnal ini akan mewakili kemungkinan menggunakan penemuan subkelompok sebagai metode analisis data transaksional. Penemuan subkelompok adalah teknik penambangan data yang mengekstraksi aturan yang menarik. *APRIORI-SD* adalah metode dalam penemuan subkelompok di mana ukuran evaluasi yang digunakan oleh *APRIORI-SD* sudah memprioritaskan distribusi keanehan dari data yang diberikan.

Hasil dari percobaan ini adalah dapat menemukan anomali seperti membedakan subkelompok dengan perbedaan setinggi 50% dari persentase keseluruhan. Dengan hasil tersebut orang mampu menciptakan strategi yang lebih baik di masa depan.

**Kata Kunci:** *APRIORI, APRIORI-SD, Subgroup discovery*

## ABSTRACT

*Retail store needs to evolve especially in digital age where e-commerce becoming more and more common and most people prefer the convenience of an e-commerce. One of the biggest advantage of a "newer" e-commerce is they build their business model on the foundation of processing data, whereas older retail store doesn't. Development of data mining and machine learning are pushing older business model to do better.*

*This journal represents the possibilities of using subgroup discovery as a method of analyzing transactional data. Subgroup discovery is a data mining technique which extracts interesting rule. APRIORI-SD is a method within subgroup discovery where evaluation measure use by APRIORI-SD already prioritizing unusualness distribution of a given data.*

*The result of this knowledge are able to find anomalies such as differentiating subgroup(s) with differences up to 50% compared to overall distribution percentage. With the result people are able to create a better strategies in the future.*

**Keywords:** *APRIORI, APRIORI-SD, Subgroup discovery*

## 1. PENDAHULUAN

Pengolahan data dalam dunia bisnis telah menjadi sebuah *trend* akhir-akhir ini. Semakin banyak bidang usaha yang secara mandiri mengolah data yang mereka miliki. Usaha mengolah data untuk mengenali *behavior* dari consumer guna memberikan layanan yang lebih baik[3]. Di tengah perkembangan ini, apabila

ada usaha yang tertinggal karena masih atau belum mengolah datanya akan tertinggal.

Salah satu usaha yang masih mengolah data secara manual adalah toko retail. Toko retail adalah bidang usaha yang sudah ada sejak lama. Beberapa toko retail raksasa sudah melakukan pengolahan data untuk memahami perpindahan *trend*, dan mengenal konsumernya[9]. Sekalipun tidak sedikit dari mereka telah melakukan pengolahan data, masih banyak pula toko retail yang belum memaksimalkan sumber daya yang dimilikinya. Berangkat dari pemikiran tersebut, tercetuslah ide untuk membantu memudahkan pengolahan data toko retail.

*Subgroup discovery* adalah suatu metode *data mining* yang dapat digunakan untuk mengenali perbedaan dari setiap subkelompok yang ada pada suatu data, dengan fokus mencari sebuah relasi data yang bersifat unik, anomali terhadap sebuah target tertentu. Hasil dari *subgroup discovery* tersebut adalah sebuah subkelompok yang memiliki deskripsi dan target. Poin atau skor yang didapat oleh subkelompok tersebut akan menunjukkan keunikan dan pengaruhnya secara signifikan.

Metode yang digunakan dalam journal ini adalah *APRIORI-SD*. Metode ini cukup mudah untuk digunakan karena *user* hanya perlu menentukan *class target* yang diinginkan. Keuntungannya, tidak ada bias pada hasil yang dikeluarkan karena pengolahan data dalam membentuk *rule* dilakukan oleh program. Metode ini merupakan turunan dari *APRIORI-C* yang merupakan metode *data mining* ditambah *classification* berdasarkan *association rules*. *Association Rules* adalah salah satu teknik data mining yang sangat populer, dan masih digunakan oleh perusahaan-perusahaan *modern* seperti *e-commerce*[7]. Selain itu *Association rules* juga digunakan untuk menemukan adanya *pattern* dalam data[5].

## 2. TINJAUAN PUSTAKA

### 2.1 *Subgroup discovery* (SD)

*Subgroup discovery* merupakan salah satu metode *data mining* yang mengekstrak *rule(s)* yang dianggap menarik terhadap sebuah *target variable*. [1] Pada dasarnya *Subgroup discovery* merupakan sebuah teknik untuk mencari sesuatu yang unik, atau bersifat *anomaly* dalam sebuah distribusi data melalui prespektif yang baru. Prespektif ini dapat dicapai dengan memberikan batasan-batasan seperti menentukan *target* dari objektif, melihat jumlah dari data yang dicakup, data yang menarik bisa berupa data yang paling besar, atau data yang memiliki *anomaly* secara *statistic*. *Subgroup discovery* menghasilkan banyak hasil yang berbeda sesuai dengan *target* yang telah ditentukan, dengan proses yang telah ditentukan[8].

*Subgroup discovery* berusaha mencari relasi pada satu atau beberapa elemen/*variable* terhadap sebuah *target*, dikarenakan *subgroup discovery* berfokus untuk mengekstrak data yang menarik, penilaian tidak selalu membutuhkan keseluruhan data untuk mencapai kesimpulan, bahkan mungkin cukup partial data saja. Relasi yang dihasilkan dikeluarkan dalam bentuk *Rule(R)*,

dan *Rule* ini bisa didefinisikan secara formal dalam persamaan (1).

$$R : Cond \rightarrow TargetValue \quad (1)$$

Dimana *TargetValue* merupakan nilai dari elemen yang ditentukan sebagai target dari pencarian (*target variable*), dan *Cond* adalah sebuah deskripsi atau poin-poin yang menentukan identitas dari subkelompok yang dihasilkan.

## 2.2 APRIORI-C

*APRIORI-C* merupakan pengembangan dari metode *APRIORI*, dengan menambahkan *classification*. [2] *APRIORI-C* dapat diimplementasikan dalam klasifikasi dengan mengikuti langkah-langkah sebagai berikut.

1. *Discretize continuous attribute.*
2. Pada setiap atribut dengan  $N$  elemen, buatlah  $N$  item, pada item yang koresponden berikan nilai 1, dan berikan nilai 0 untuk item lainnya, nilai hilang akan memiliki nilai 0 pada semua item.
3. Jalankan sebuah *Association rule learning algorithm*
4. Pilih *rule* yang pada *right hand* sidenya hanya memiliki 1 item, jadikan itu sebagai *target variable*.
5. Gunakan *set of rules* yang dihasilkan untuk mengklasifikasi data yang ada.

Klasifikasi yang dilakukan bertujuan untuk menemukan *set of rules* yang mampu mengelompokan data yang ada [6].

### 2.2.1 Post processing by rule subset selection

Dengan menurunkan angka *support* maupun *confidence*, *rule* yang dihasilkan akan sangat banyak dan mungkin menutupi *target* yang diinginkan. Ada 2 metode yang dapat digunakan untuk menentukan *rule(s)* terbaik.

- ***N best rule*** Algoritma mencari *rule* dengan nilai terbaik, dalam hal ini berarti *support*, mengeliminasi seluruh itemsets yang dicakup oleh *rule*. Urutkan *rule* berdasarkan *support* dan mengulangi hal yang sama sampai tidak ada lagi *rule(s)* yang perlu dievaluasi atau tidak ada lagi *itemset(s)* yang perlu dicover, algoritma berhenti dan mengembalikan *rule* dalam bentuk *if-then-else*

- ***N best rule for each class*** metode ini berlaku mirip dengan *N best rule* akan tetapi metode ini dilakukan pada setiap *class* yang dihasilkan, memungkinkan *class* yang berukuran kecil juga diperhitungkan sebagai hasil.

## 2.3 APRIORI-SD

*APRIORI-SD* adalah pengembangan dari *APRIORI-C* atau *APRIORI* yang diterapkan untuk *subgroup discovery* [4], komponen utama dari metode ini masih berbasis *APRIORI*. Perubahan dari *APRIORI-C* yang membuat metode ini bisa diaplikasikan pada *subgroup discovery* adalah melakukan *weighting scheme* pada *rule post-processing*, dan memodifikasi *evaluation measure* yang digunakan, menambahkan beban pada *Weight Relative Accuracy*.

Segment 1 merupakan *pseudo code* dari *APRIORI-SD*.

```

Algorithm Apriori-SD(Examples, Classes, minSup, minConf, k)
Ruleset = APRIORI-C(Examples, Classes, minSup, minConf)
Majority = the majority class in example
Resultset = {}
Repeat
BestRule =rule with the highest WRAcc value
Resultset = Resultset U BestRule
    
```

*Ruleset* = *Ruleset* \ *BestRule* decrease the weight of examples covered by *BestRule* (using the example weighting scheme) remove from *Examples* the examples covered more than  $k$ -times.

*Until* *Examples* = {} or *Ruleset* = {}

return *Resultset* = *Resultset* U "*true*  $\rightarrow$  *Majority*"

### segment 1. pseudo code APRIORI-SD

*Pseudo-Code* dari *APRIORI-SD*. Argumen yang menjadi input dari algoritma ini adalah : *Examples*, *Classes*, *MinSupport*, *MinConfidence*, dan  $k$ . *Examples* merupakan *set data* yang akan dijadikan acuan dalam *training*, *Classes* adalah *value* dari atribut yang ada di dalam *set data training*, dan parameter  $k$  menentukan batasan *threshold* yang akan digunakan.

*APRIORI-SD* menghasilkan *rule(s)* dengan menggunakan fungsi dari *APRIORI-C* yang dikurangi *post processing* dalam *selection rule* karena akan menggunakan nilai dari *weight relative accuracy* sebagai ganti *support*, *rule* terbaik diolah sama dengan cara memilih *rule* terbaik pada *APRIORI-C*.

### 2.3.1 Example Weighting Scheme

Pencarian *rule* terbaik pada metode *APRIORI-SD*, menggunakan langkah sebagai berikut. Data yang dilingkup oleh *rule* terbaik tidak dieliminasi, tetapi di kalkulasi ulang.

Kalkulasi ulang didapatkan dengan rumus  $w(e, i) = 1/i+1$  dimana  $w$  adalah *weight* atau beban yang ingin dicari,  $i$  adalah jumlah *rule* yang melingkupi data. Semakin banyak *rule* yang dapat melingkup data semakin kecil beban dari data tersebut. Semakin kecil *rule* yang mampu melingkup data semakin besar beban dari data tersebut, kesimpulan apabila sebuah data dapat dilingkup oleh banyak *rule*, maka data bisa dianggap bersifat terlalu umum dan tidak menarik.

### 2.3.2 Weight Relative Accuracy

*Weight Relative Accuracy* (WRAcc) adalah salah satu *evaluation measure* yang digunakan dalam *subgroup discovery*. Dalam pengembangannya *WRAcc* merupakan turunan dari *evaluation measure* yang disebut *novelty*, adanya *WRAcc* memungkinkan untuk menilai sebuah *rule* berdasarkan tingkat "*unusualness*" dari pada *rule(s)* yang ada [10] Dalam pengembangannya *WRAcc* merupakan turunan dari *quality of measure subgroup discovery* yang disebut *novelty*, adanya *WRAcc* memungkinkan untuk menilai sebuah *rule* berdasarkan tingkat "*unusualness*" dari pada *rule(s)* yang ada.

*WRAcc* dapat dicari dengan melakukan penghitungan sebagai berikut.  $n(X)$  merupakan jumlah data yang dicakup oleh *rule*  $X \rightarrow Y$ , dan  $n(Y)$  adalah jumlah data yang dicakup oleh *class*  $Y$ , dan  $n(X.Y)$  adalah jumlah data yang *correctly covered* (*true positive*). Digunakan  $p(X).p(X.Y)$ , dll. Sebagai probabilitas yang korespon. *Rule accuracy* atau *Rule confidence* didefinisikan sebagai  $Acc(X \rightarrow Y) = Conf(X \rightarrow Y) = p(Y|X) = p(X.Y)/p(X)$ . *WRAcc* didefinisikan sebagai berikut.

$$WRAcc(X \rightarrow Y) = p(X).(p(Y|X) - p(Y)) \quad (2)$$

Persamaan (2) terdiri dari dua komponen *generality*  $p(X)$  dan akurasi  $p(Y|X) - p(Y)$ . istilah akurasi adalah untuk mengkitu seberapa akurat sebuah *rule* terhadap *true*  $\rightarrow Y$ , sebuah *rule*  $X \rightarrow Y$  dikatakan menarik apabila berhasil memberikan nilai lebih tinggi dibanding *default* yang ada.

### 2.3.3 WRAcc with weight

*Quality measure* pada *APRIORI-SD* sudah dimodifikasi untuk bisa menghitung beban dalam rumus yang digunakan, memungkinkan pencarian pada *space* yang berbeda dalam

mencari *rule(s)* yang terbaik, WRAcc yang dimodifikasi didefinisikan pada persamaan (3).

$$wWRAcc(X \rightarrow Y) = n'(X)/N' * (n'(X.Y)/x'(X) - n'(Y)/N') \quad (3)$$

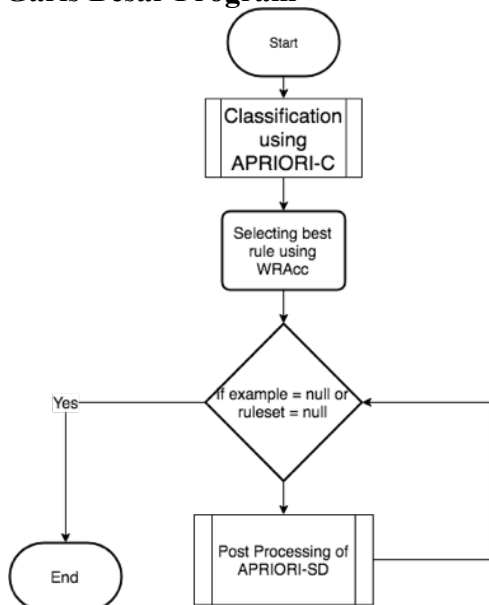
### 3. ANALISA DAN DESAIN

#### 3.1 Analisis Teori

APRIORI-SD adalah metode yang diturunkan dari APRIORI-C, dasar pemikiran dari APRIORI-SD adalah penggunaan metode *association rule* pada *subgroup discovery*. APRIORI-C bertujuan untuk mengelompokkan data (*classification*) dengan metode *association rules*. Pengembangan yang dilakukan dari APRIORI-C ke APRIORI-SD adalah adanya *weighting scheme*, dan mengubah *evaluation measure* yang digunakan yaitu, *weighted relative accuracy*(WRAcc).

Kenapa metode ini cocok, karena metode APRIORI-SD memiliki keunggulan. Yaitu, *target variable* yang ditentukan tidak harus memiliki value yang bersifat *binary* (2 output *True or false, above or below n number*), namun bisa kategorikal. dan akan dikategori berdasar *class* baru disesuaikan *target output* sesungguhnya namun harus bersifat *discrete*.

#### 3.2 Garis Besar Program

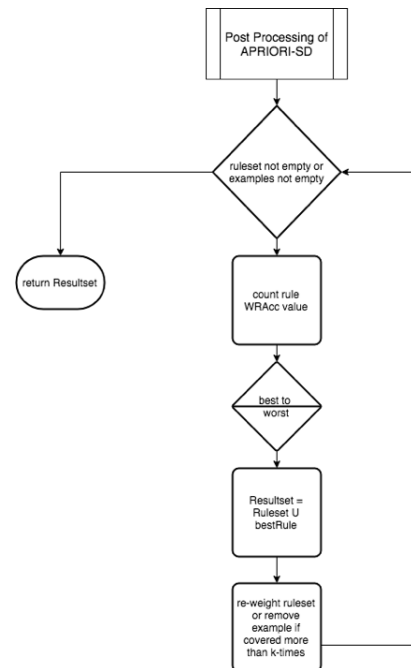


Gambar 1. Flowchart Utama Program

Gambar 1 merupakan *Flowchart* utama dari program yang dibuat. Pada tahap awal dari program ini adalah membuat *Rule(s)* berdasarkan algoritma *APRIORI* dan mengelompokkan data berdasarkan *class* yang ada, setelah *Rule(s)* terbentuk, akan disaring untuk menghilangkan *Rule(s)* yang tidak diperlukan atau mengandung *noise*, barulah masuk tahap *post processing* menggunakan *N best rules* untuk menentukan nilai dari *Rule(s)*.

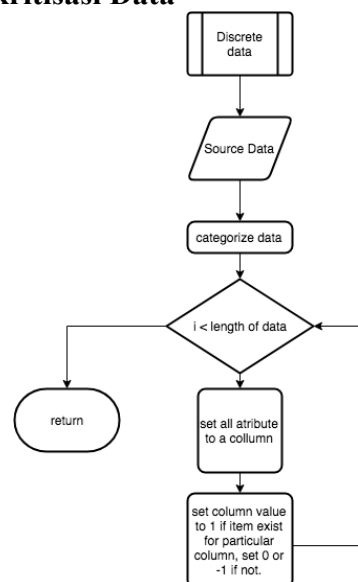
#### 3.3 Detail

Gambar 2 merupakan *Post-processing* dalam *APRIORI-SD* adalah detail dari *N best rules*. Proses akan berjalan terus sampai tidak ada *Rule(s)* atau tidak ada lagi *Item* atau *examples* untuk di cari, langkah pertama dari program ini adalah mengurutkan *Rule(s)* berdasar nilai *WRAcc* dari yang paling tinggi sampai yang paling rendah, *Rule(s)* tertinggi dipilih dan dilihat cakupannya, semua *itemset* atau *example* di hitung ulang bebannya dengan *coverage* yang baru. Eliminasi *Rule(s)* yang telah digunakan, ulangi prosesnya sampai selesai.



Gambar 2. Post Processing APRIORI-SD

#### 3.4 Diskritisasi Data



Gambar 3. Flowchart Diskritisasi Data.

Diskritisasi data diperlukan agar data dapat diolah. Karena APRIORI-C merupakan metode yang tidak dapat mengolah data yang sifatnya “*continious*”. Dalam *flowchart* ini data pertama-tama dikelompokkan berdasar kategori yang telah ditentukan. Setelah dikelompokkan, akan dibuat *file* besar yang menampung hasil diskritisasi data, dalam *file* ini, kategori yang digunakan akan menjadi nama kolom. Setiap baris mewakili 1 data. Apabila *item* pada baris mengandung nama kolom yang korespon, kolom akan diberi nilai 1, bisa tidak diberi nilai 0 atau -1. Proses digambarkan pada Gambar 3.

### 4. Result Evaluation

#### 4.1 Input data

Input data dari program ini berupa data transaksi yang telah dikelompokkan. *Data* yang disajikan bisa berbagai jenis dan terbebas dari berapapun banyaknya kolom atau baris yang ada, Gambar 2 merupakan data yang diinput pada program, data ini

berisikan 6 kolom dengan keterangan *Gender*, dan barang-barang yang dibeli dalam sebuah transaksi.

customer_gender	Food	House	Technology	Toiletries	Garden
Male	food	-1	technology	toiletries	-1
Male	food	-1	-1	toiletries	-1
Male	-1	house	-1	-1	-1
Male	-1	-1	technology	-1	-1
Male	-1	-1	technology	toiletries	garden
Female	food	house	-1	-1	garden
Female	food	house	-1	-1	-1

Gambar 4. Data Input

## 4.2 Generating rules from APRIORI

Dari *input* data yang ada, dijalankan program APRIORI, dan dilakukan eliminasi terhadap data yang mengandung -1, dimana nilai -1 berarti data tersebut bernilai *false*, setelah data dibersihkan data akan diurutkan berdasarkan nilai *support* yang dimiliki, hasil bisa dilihat pada gambar 5.

```
{000", foodi} -> {-1} (conf: 1.000, supp: 0.072, lift: 0.134,
{-1, 000"} -> {foodi} (conf: 0.909, supp: 0.072, lift: 1.242,
{000"} -> {-1, foodi} (conf: 0.909, supp: 0.072, lift: 1.242,
{400", foodi} -> {-1} (conf: 1.000, supp: 0.036, lift: 0.134,
{-1, 400"} -> {foodi} (conf: 0.833, supp: 0.036, lift: 1.139,
```

Gambar 5. Hasil rules dari APRIORI

Setelah *rules* diseleksi, maka pencarian *best rule* dimulai, dengan melihat skor *WRAcc* dari yang paling tinggi. *Rules* yang dianggap terbaik dalam iterasi tersebut akan diaplikasikan pada seluruh *itemsets*. Dicatat berapa banyak *itemsets* yang tercakup oleh *rule* tersebut. apabila *coverage area* pada *rule* cukup besar, maka akan ditambahkan ke dalam *candidate* dan *final set*. *Rule* tersebut akan menjadi deskripsi dari subkelompok, nilai *WRAcc* akan menjadi skor dari subkelompok tersebut. Gambar 6 menunjukkan hasil subkelompok.

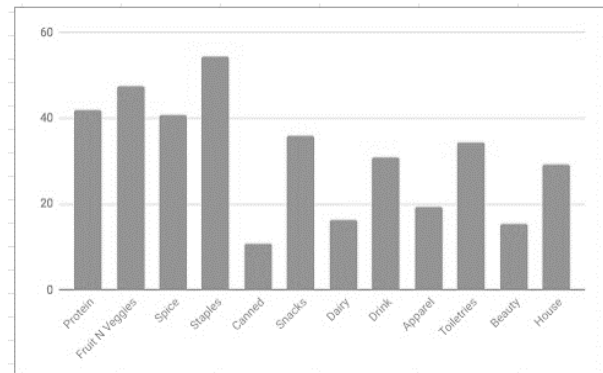
```
description = ('apparel',) --> ('toilet',)
score = 0.06906875
=====
description = ('spice',) --> ('staples',)
score = 0.06893125
=====
description = ('fruitNveggies', 'spice') --> ('staples',)
score = 0.05865625000000001
=====
description = ('female', 'spice') --> ('staples',)
score = 0.04844375000000001
=====
description = ('fruitNveggies', 'protein', 'staples') --> ('spice',)
score = 0.04739375000000001
=====
description = ('apparel', 'house') --> ('toilet',)
score = 0.04568125
=====
```

Gambar 6. Hasil Subkelompok

## 4.3 Proving Rule(s)

Hasil *rule* pada gambar 6 akan diujikan dengan data yang sama apakah benar bahwa deskripsi subkelompok yang dihasilkan memiliki distribusi yang aneh, dalam hal ini aneh diartikan sebagai distribusi yang ada mengalami perbedaan yang signifikan terhadap distribusi keseluruhan data.

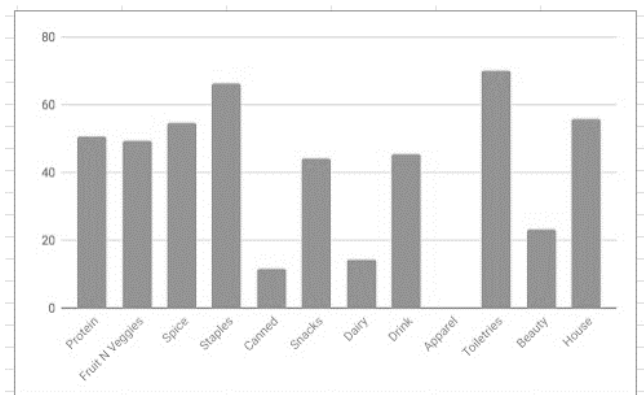
Tahap pertama dalam pembuktian ini adalah dengan membandingkan distribusi keseluruhan. Gambar 7 merupakan gambar yang berisi distribusi keseluruhan. Distribusi keseluruhan merupakan distribusi yang terbentuk oleh seluruh data transaksi yang ada. Data keseluruhan beserta data-data yang lain ditampilkan dalam bentuk persen. Persentase dibentuk dengan jumlah transaksi yang mengandung barang di dalam grafik, dibagi seluruh jumlah transaksi yang ada, dikalikan 100.



Gambar 7. Overall Distribusi pada data

Berdasarkan Gambar 7, akan dibandingkan hasil distribusi sebuah subkelompok dari *rule(s)* yang sudah dihasilkan oleh program.

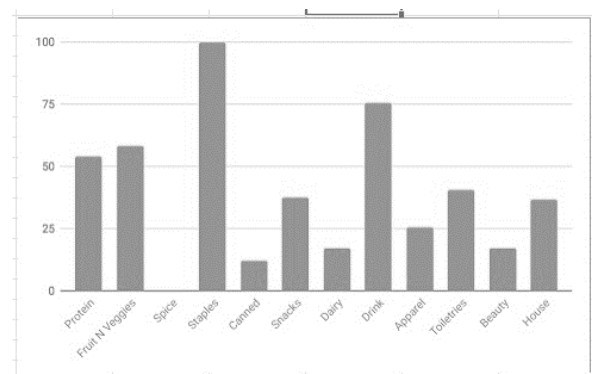
Hasil pertama adalah *apparel* → *toilet*, pada Gambar 8. Merupakan hasil dari distribusi data pada subkelompok *apparel*.



Gambar 8. Overall Distribusi pada data

Dari hasil subkelompok *apparel*, didapati bahwa pembelanjaan barang dalam kategori *toiletries* mencapai 70%. Artinya 70% orang yang membeli barang dengan kategori *apparel* juga akan membeli barang dari kategori *toiletries*. Jika kedua angka dibandingkan, penjualan *toiletries* pada keseluruhan adalah 35%, sedangkan dalam subkelompok *apparel* 70%, terjadi perbedaan sejauh 35%, atau bila dilihat secara peningkatan mencapai 100%. Hasil seperti inilah yang dikatakan aneh, atau *anomaly*.

Contoh kedua akan mengambil *rule* *spice* → *staples*



Gambar 9. Distribusi apparel yang dibagi gender

Pada Gambar 9. Disimpulkan bahwa setiap orang yang membeli barang pada *section* *spice*, orang yang sama akan membeli barang di *section* *staples* (tingkat *spice* dibeli bersama *staples* = 100%). Jika dibandingkan dengan grafik *staples* pada total distribusi hanya 50%, didapatkan perbedaan sejauh 50% di dalam subkelompok *spice* saja.

Berdasarkan hasil diatas dapat disimpulkan bahwa metode *APRIORI-SD* dapat memberikan informasi yang bisa dikategorikan menarik. Hasil bisa dikatakan menarik karena perbedaan nilai pada grafik keseluruhan dengan grafik dalam subkelompok bisa mencapai 100%. Hal menarik lain adalah. Apabila skor *WRAcc* antara 2 *rule* hampir sama, peningkatan persen terhadap *target variable* dalam subkelompok terhadap *global* juga hampir sama.

## 5. KESIMPULAN

Dari hasil pengujian sistem yang telah dilakukan, dapat diambil beberapa kesimpulan antara lain:

- Dari pengujian yang dilakukan ditemukan *rule* yang paling signifikan, memiliki perbedaan setinggi 100% atau 2 kali dibanding data keseluruhan.
- Apabila skor *WRAcc* antar 2 *rule* hampir sama, peningkatan persen terhadap *target variable* dari subkelompok terhadap *global* penjualan juga hampir sama.
- Hasil dari *WRAcc improve* akan semakin tinggi apabila data dalam subkelompok yang dihasilkan, tidak *overlapping* dengan data subkelompok yang lain.

Saran yang dapat diberikan untuk penyempurnaan dan pengembangan program lebih lanjut antara lain:

- *Reverse subkelompok*, apabila diketahui data transaksi namun tidak diketahui gendernya
- Menggunakan *evaluation measure* yang berbeda selain *WRAcc*
- Melakukan komparasi dengan metode *Subgroup discovery* lainnya.

## 6. DAFTAR REFERENSI

- [1] Herrera, Franciso., Gonzalez, Pedro. 2010. An Overview on Subgroup discovery: Foundations and Applications. London
- [2] Jovanoski, Viktor., Lavrac, Nada. 2002 Classification Rule Learning with APRIORI-C. Slovenia
- [3] Kaur, Manpreet., Kang, Shivani. 2016. Market Basket Analysis: Identify the changing trends of market data using association rule mining. India
- [4] Kavsek, Branko, Lavrac, Nada. 2006. Adapting Association Rule Learning to Subgroup discovery
- [5] Kumar, Dr.V. Srinivasa Kumar., Dr. R. Renganathan. 2018. Consumer Buying Pattern analysis using Apriori Association Rule.
- [6] Liu, Bing, Hsu, Wynne 1998. Integrating Classification and Association Rule Mining. National University of Singapore.
- [7] Mining Interest In Online Shoppers' Data: An Association Rule Mining Approach. 2018. Acta Polytechnica Hungarica, 14(7). doi:10.12700/aph.14.7.2017.7.9
- [8] Sani, M.Fani., van der Aalst, W.M.P. 2017. Subgroup Discovery in Process Mining. Eindhoven University of Technology.
- [9] Samli, A. C. 2015. Consumer Behavior and Retail Strategy. Coping with Retail Giants, 83-102. doi:10.1057/9781137476340\_8
- [10] Todorovski, Ljupco. Lavrac, Nada. 2000 Predictive Performance of Weighted Relative Accuracy. University of Bristol.