

PENCARIAN FREQUENT ITEMSET DENGAN ALGORITMA APRIORI DAN PYTHON. STUDI KASUS: DATA TRANSAKSI PENJUALAN ECERAN ONLINE DI UK

Astried Silvanie

Program Studi Teknik Informatika, Fakultas Ilmu Komputer
Institut Bisnis dan Informatika (IBI) Kosgoro 1957
Email: astried@ibi-k57.ac.id

ABSTRAK

Saat ini, sejumlah besar data disimpan dalam basis data di berbagai sektor bisnis. Dengan *data mining* kita dapat mengekstraksi informasi yang berguna dari data ini. Salah satu teknik *data mining* yaitu *Market Based Analysis* digunakan untuk menemukan aturan asosiasi di antara kumpulan data. Dalam penelitian ini data yang digunakan adalah data publik transaksi online eceran di UK. Dari hasil penelitian didapatkan 11 itemset dengan aturan minimum support 1.5% dan confidence $\geq 80\%$, dan 3 itemset dengan aturan minimum support 2% dan confidence $\geq 80\%$. Hasil ini dapat digunakan kepentingan bisnis di online retail untuk merekomendasikan produk yang sering dibeli bersamaan. Implementasi dilakukan menggunakan python menggunakan pustaka mlxtend, pandas dan sys. Hasil ini adalah itemset yang sering dibeli dari total 22.846 transaksi dan 4.055 produk.

Kata Kunci: *Market Based Analysis, Association Rule, Apriori, Frequent Itemset, Python, Market Based Analysis*

I. PENDAHULUAN

Saat ini, sejumlah besar data disimpan dalam basis data di berbagai sektor bisnis. Tetapi tidak seluruh data berguna bagi pengguna. Itulah mengapa sangat penting untuk mengekstraksi informasi yang berguna dari data yang sangat besar. Informasi diekstraksi dari sejumlah data yang sangat besar dengan menggunakan teknik statistik, matematika, kecerdasan buatan atau pun *machine learning* dan proses ini dikenal sebagai *Data Mining* (J, Han et al, 2006), (Kurnia et al, 2019). Salah satu teknik *data mining* yaitu *Market basket analysis* (MBA). *Market basket analysis* adalah teknik penambangan data untuk menemukan asosiasi di antara sekumpulan data yang sangat banyak (Gupta et al, 2014). Secara spesifik, MBA bertujuan mengidentifikasi barang atau kombinasi barang yang paling sering dibeli oleh pelanggan (Kurniawan et al, 2018). MBA memberitahu pengecer informasi tentang penjualan terkait berdasarkan kelompok barang yang sering dibeli pelanggan. Misalkan pelanggan yang sering membeli roti juga membeli beberapa produk yang berhubungan dengan roti seperti susu, mentega (Charlet et al, 2012). Informasi tersebut diekstraksi dengan sebuah aturan yang dikenal aturan asosiasi. Aturan asosiasi adalah teknik penggalian data untuk menemukan aturan keterkaitan antara kombinasi item dalam satu kumpulan data (Kurnia et al, 2019).

Dalam penelitian ini, MBA dilakukan terhadap data publik atas transaksi online eceran bukan toko di UK dalam kurun waktu 01/12/2010 sampai 09/12/2011. Himpunan data ini bisa diakses di <https://archive.ics.uci.edu/ml/machine-learning-databases/00352/>. Hasil yang ingin dicapai dari penelitian ini adalah:

- 1) Menemukan kombinasi produk yang paling sering dibeli dan pada batas nilai berapa minimal pembelian ini.
- 2) Bagaimana implementasi kode python untuk pencarian kombinasi produk yang memenuhi aturan asosiasi.

II. METODE PENELITIAN

2.1 Proses Data Mining

Proses penemuan pengetahuan adalah urutan iteratif dari langkah-langkah berikut ini (J, Han et.all, 2006):

- 1) *Data cleaning*. Proses menghilangkan *noise* dan data yang tidak konsisten.
- 2) *Data Integration*. Proses ini dilakukan jika perlu pengabungan data dari beberapa sumber yang berbeda.
- 3) *Data Selection*. Pemilihan data yang relevan untuk pencarian pola.
- 4) *Data Transformation*. Proses agregasi atau pengubahan format data agar sesuai dengan yang dibutuhkan.
- 5) *Data mining*. Proses pencarian pengetahuan yang tersembunyi.
- 6) *Pattern evaluation*. Pengidentifikasian pola-pola yang menarik yang merepresentasikan pengetahuan sesuai dengan ukuran-ukuran tertentu.
- 7) *Knowledge presentation*. Teknik merepresentasikan pengetahuan agar dapat dimengerti pengguna.

2.2 Aturan Asosiasi

Aturan asosiasi menemukan asosiasi yang menarik jika memenuhi nilai minimal dari dua parameter yaitu *support* dan *confidence*. *Support(A)* adalah berapa kali item x muncul dalam basis data transaksi. *Confidence(A->B)* adalah istilah yang terkait dengan aturan asosiasi dan didefinisikan secara matematis sebagai: $Confidence(A \rightarrow B) = support(A \cap B) / support(A)$ (Kaur dan Kangs, 2016).

$$Support(A) = \frac{\Sigma \text{transactions containing A}}{\text{Total transactions}}$$

Equation 1. Rumus menghitung Support (J, Han et.all, 2006)

$$Confidence(A \Rightarrow B) = P(B|A) = \frac{\Sigma \text{ transactions containing A and B}}{\Sigma \text{ transactions containing A}}$$

Equation 2. Rumus menghitung Confidence (J, Han et.all, 2006)

Terjadinya itemset A tidak tergantung pada terjadinya itemset B jika $P(A \cup B) = P(A)P(B)$; jika tidak, itemsets A dan B bergantung dan berkorelasi sebagai peristiwa. Definisi ini dengan mudah bisa diperpanjang hingga lebih dari dua itemsets. Peningkatan (*Lift*) antara terjadinya A dan B bisa diukur dengan:

$$lift(A, B) = \frac{P(A \cup B)}{P(A)P(B)}$$

Equation 3. Rumus menghitung Lift (J, Han et.all, 2006)

Aturan asosiasi diturunkan dari kumpulan item menggunakan *support* dan *confidence* sebagai tingkat ambang batas. Kumpulan item yang memiliki dukungan minimum dikenal sebagai *frequent itemsets* (Charlet et al, 2012).

2.3 Algoritma Apriori

Salah satu algoritma *data mining* untuk menemukan *frequent itemsets* atau kumpulan item yang sering muncul adalah apriori. Apriori menemukan kumpulan item yang sering menggunakan

pendekatan berbasis tingkat yang berulang pada generasi kandidat. Diketahui D adalah transaksi dalam basis data. Diketahui min_support adalah batas nilai support minimum.

```

1  Cdn : Candidate itemset of size n
2  Ln : frequent itemset of size n
3  L1 = {frequent items};
4  For (n=1; Ln != Φ; n++)
5  Do begin
6    Cdn+1 = candidates generated from Ln ;
7    For each transaction T in database do
8      Increment the count of all candidates in Cdn+1 that are
9      contained in T
10   Ln+1 = candidate in Cdn+1 with min_support
11 End
12 Return Un Ln
(Charlet et al, 2012)

```

```

L1 = find frequent 1-itemsets(D);
for (k = 2; Lk-1 ≠ ∅; k++) {
  Ck = apriori gen(Lk-1);
  for each transaction t ∈ D { // scan D for counts
    Ct = subset(Ck, t); // get the subsets of t that are candidates
    for each candidate c ∈ Ct
      c.count++;
  }
  Lk = {c ∈ Ck | c.count ≥ min sup}
}
return L = Uk Lk;

procedure apriori gen(Lk-1 : frequent (k - 1)-itemsets)
for each itemset l1 ∈ Lk-1
  for each itemset l2 ∈ Lk-1
    if (l1[1] = l2[1]) ∧ (l1[2] = l2[2])
      ∧ ... ∧ (l1[k - 2] = l2[k - 2]) ∧ (l1[k - 1] < l2[k - 1]) then {
        c = l1 ∪ l2; // join step: generate candidates
        if has infrequent subset(c, Lk-1) then
          delete c; // prune step: remove unfruitful candidate
        else add c to Ck;
      }
}
return Ck;

procedure has infrequent subset(c : candidate k-itemset; Lk-1 : frequent (k - 1)-itemsets); // use prior knowledge
for each (k - 1)-subset s of c
  if s ∉ Lk-1 then
    return TRUE;
return FALSE;

```

Kode 1. Pseudocode Algoritma Apriori

III. HASIL DAN PEMBAHASAN

Pertama kali kita harus memeriksa data publik yang sudah diunduh dari <https://archive.ics.uci.edu/ml/machine-learning-databases/00352/>. Kumpulan data terdiri dari 541.909 baris record dan 8 atribut. Berikut ini adalah atribut yang dimiliki data:

- InvoiceNo: Nomor Invoice . Berupa nominal 6-digit secara unik mengidentifikasi transaksi. Jika kode diawali dengan 'c' maka transaksi tersebut dibatalkan.

- StockCode: Kode item/ kode produk. Berupa nominal 5-digit secara unik mengidentifikasi setiap item.
- Description: Nama produk/item. Berupa deskripsi nama produk.
- Quantity: Kuantitas produk untuk setiap transaksi.
- InvoiceDate: Tanggal dan waktu transaksi.
- UnitPrice: Harga perunit dengan mata uang poundsterling.
- CustomerID: Nomor unik pelanggan berupa nominal dengan panjang 5-digit.
- Country: Nama negara dimana pelanggan berada.

3.1 Data Cleaning

Ada dua hal yang harus kita periksa pertama kali yaitu nilai yang hilang dan *noise*. *Noise* adalah kesalahan atau varian acak dalam variabel terukur. Ada beberapa metode untuk membersihkan data-data seperti ini yaitu (J, Han et.all, 2006):

- Mengacuhkan dan menghapus *record* tersebut.
- Mengisi secara manual dengan nilai acak.
- Mengisi secara manual dengan nilai *mean* atau *median*.
- Mengisi secara manual dengan nilai *mean* atau *median* dari kelas yang sama.
- Mengisi secara manual dengan nilai yang paling sering muncul.

Proses pembersihan yang dilakukan pada kumpulan data pada penelitian ini adalah:

- 1) Menghapus semua transaksi yang dibatalkan. Cirinya adalah kode transaksi diawali huruf 'C'.
- 2) Menghapus semua transaksi yang bukan proses pembelian yaitu:
 - Amazon Fee
 - Adjust bad debt
 - Bank Charges
 - CRUK Commission
 - Discount
 - Samples

Jumlah data yang dibersihkan adalah 9.307 baris record yaitu sekitar 1.7% dari keseluruhan transaksi. Sisa record setelah dibersihkan menjadi 532.602 baris record. Jumlah transaksi secara unik adalah 22.846 transaksi. Jumlah item secara unik adalah 4.055 produk. Proses pembersihan data dilakukan menggunakan fungsi filter di Excel dan menghapus baris.

3.2 Data Integration

Karena data diunduh dari sumber dan hanya terdiri dari satu *file* csv (comma separated value) maka tidak ada proses integrasi untuk data ini.

3.3 Data Selection

Tidak semua atribut akan berguna untuk mencari kombinasi item yang paling sering. Dalam penelitian ini kita hanya akan mengambil dua atribut yaitu *InvoiceNo* dan *StockCode* untuk digunakan untuk mencari *frequent itemset*. Sedangkan atribut yang lain tidak akan dimasukkan ke dalam proses penggalian data.

3.4 Data Transformation

Data transaksi tertulis per baris dengan pasangan *InvoiceNo* dan *StockCode* seperti di bawah ini. Langkah pertama sebelum ke transformasi kedua adalah mengurutkannya berdasarkan atribut *InvoiceNo*. Apakah secara menurun (*descending*) atau menaik (*ascending*) tidaklah penting.

	A	B	C	D	E	F	G	H
1	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
2	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850	United Kingdom
3	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850	United Kingdom
4	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850	United Kingdom
5	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850	United Kingdom
6	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850	United Kingdom
7	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	12/1/2010 8:26	7.65	17850	United Kingdom
8	536365	21730	GLASS STAR FROSTED T-LIGHT HOLDER	6	12/1/2010 8:26	4.25	17850	United Kingdom
9	536366	22633	HAND WARMER UNION JACK	6	12/1/2010 8:28	1.85	17850	United Kingdom
10	536366	22632	HAND WARMER RED POLKA DOT	6	12/1/2010 8:28	1.85	17850	United Kingdom
11	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	12/1/2010 8:34	1.69	13047	United Kingdom
12	536367	22745	POPPY'S PLAYHOUSE BEDROOM	6	12/1/2010 8:34	2.1	13047	United Kingdom
13	536367	22748	POPPY'S PLAYHOUSE KITCHEN	6	12/1/2010 8:34	2.1	13047	United Kingdom
14	536367	22749	FELTCRAFT PRINCESS CHARLOTTE DOLL	8	12/1/2010 8:34	3.75	13047	United Kingdom
15	536367	22310	IVORY KNITTED MUG COSY	6	12/1/2010 8:34	1.65	13047	United Kingdom
16	536367	84969	BOX OF 6 ASSORTED COLOUR TEASPOONS	6	12/1/2010 8:34	4.25	13047	United Kingdom
17	536367	22623	BOX OF VINTAGE JIGSAW BLOCKS	3	12/1/2010 8:34	4.95	13047	United Kingdom
18	536367	22622	BOX OF VINTAGE ALPHABET BLOCKS	2	12/1/2010 8:34	9.95	13047	United Kingdom
19	536367	21754	HOME BUILDING BLOCK WORD	3	12/1/2010 8:34	5.95	13047	United Kingdom
20	536367	21755	LOVE BUILDING BLOCK WORD	3	12/1/2010 8:34	5.95	13047	United Kingdom
21	536367	21777	RECIPE BOX WITH METAL HEART	4	12/1/2010 8:34	7.95	13047	United Kingdom
22	536367	48187	DOORMAT NEW ENGLAND	4	12/1/2010 8:34	7.95	13047	United Kingdom
23	536368	22960	JAM MAKING SET WITH JARS	6	12/1/2010 8:34	4.25	13047	United Kingdom
24	536368	22913	RED COAT RACK PARIS FASHION	3	12/1/2010 8:34	4.95	13047	United Kingdom
25	536368	22912	YELLOW COAT RACK PARIS FASHION	3	12/1/2010 8:34	4.95	13047	United Kingdom
26	536368	22914	BLUE COAT RACK PARIS FASHION	3	12/1/2010 8:34	4.95	13047	United Kingdom
27	536369	21756	BATH BUILDING BLOCK WORD	3	12/1/2010 8:35	5.95	13047	United Kingdom
28	536370	22728	ALARM CLOCK BAKELIKE PINK	24	12/1/2010 8:45	3.75	12583	France
29	536370	22727	ALARM CLOCK BAKELIKE RED	24	12/1/2010 8:45	3.75	12583	France
30	536370	22726	ALARM CLOCK BAKELIKE GREEN	12	12/1/2010 8:45	3.75	12583	France
31	536370	21724	PANDA AND BUNNIES STICKER SHEET	12	12/1/2010 8:45	0.85	12583	France
32	536370	21883	STARS GIFT TAPE	24	12/1/2010 8:45	0.65	12583	France
33	536370	10002	INFLATABLE POLITICAL GLOBE	48	12/1/2010 8:45	0.85	12583	France
34	536370	21791	VINTAGE HEADS AND TAILS CARD GAME	24	12/1/2010 8:45	1.25	12583	France

Gambar 1. Data Mentah Dari UK Retail Dataset

Dalam penelitian ini kita akan melakukan dua kali transformasi. Transformasi pertama mengubah dari bentuk pasangan data *InvoiceNo-StockCode* ke variabel *array of array*. Dimana untuk setiap *StockCode* dari *InvoiceNo* yang sama akan dijadikan dalam satu array dengan panjang berbeda. Hasil transformasi kedua akan terlihat seperti cuplikan data di bawah ini:

[[85123A, 71053, 84406B, 84029G, 84029E, 22752, 21730], [22633, 22632], [84879, 22745, 22748, ..], ...]

Transformasi ketiga adalah perubahan dari *array of array* menjadi matriks dengan dimensi $n \times m$. Dimana n adalah banyak transaksi dengan identifikasi secara unik dengan *InvoiceNo*. Nilai m adalah banyak item yang ada dalam proses transaksi pembelian dengan identifikasi secara unik dengan *StockCode*

	10002	10080	10120	10123C	10123G	10124A	10124G	10125	10133	10134	10135	...
0	False	False	False	False	False	False	False	False	False	False	False	...
1	False	False	False	False	False	False	False	False	False	False	False	...
2	False	False	False	False	False	False	False	False	False	False	False	...
3	False	False	False	False	False	False	False	False	False	False	False	...
4	False	False	False	False	False	False	False	False	False	False	False	...
...
22041	False	False	False	False	False	False	False	False	False	False	False	...
22042	False	False	False	False	False	False	False	False	False	False	False	...
22043	False	False	False	False	False	False	False	False	False	False	False	...
22044	False	False	False	False	False	False	False	False	False	False	False	...
22045	False	False	False	False	False	False	False	False	False	False	False	...

[22046 rows x 4055 columns]

Gambar 2. Matrix

3.5 Data Mining Dengan Algoritma Apriori

Proses penggalan data dimulai dari membaca *file* sumber yang sudah dibersihkan sampai mendapatkan *frequent itemset*. Implementasi algoritma dari proses membaca data dari sumber sampai dengan mencari *frequent itemset* dikodekan dengan bahasa pemrograman python. Berikut ini adalah kodenya:

```

1 import pandas as pd
2 import sys
3 from mlxtend.preprocessing import TransactionEncoder
4 from mlxtend.frequent_patterns import apriori, association_rules
5
6 #deklarasi variabel array of item, array of transactions, current id
7 #dan penghitung
8 myitems = []
9 transactions = []
10 currid = ""
11 k = 0
12 nRow = 0
13
14 #membaca data dari file csv
15 order = pd.read_csv('retailuk.csv', header=None, usecols=[0,1])
16
17 #fungsi pengambilan data dengan membaca perbaris dan kemudian
18 #mentransformasikan ke array of array.
19 def ambildata(order):
20     myitems = []
21     transactions = []
22     currid = ""
23     k = 0
24     nRow = 0
25     for index, row in order.iterrows():
26         if k == 0:
27             currid = row[0]
28             myitems.append(row[1])
29         else:
30             if currid != row[0]:
31                 currid = row[0]
32                 transactions.append(myitems)
33                 myitems = []
34                 myitems.clear()
35                 myitems.append(row[1])
36                 nRow += 1
37             else:
38                 myitems.append(row[1])
39                 k += 1
40
41     transactions.append(myitems)
42     myitems = []
43     myitems.clear()
44     return transactions
45
46 dataset = ambildata(order)
47
48 te = TransactionEncoder()
49 te_ary = te.fit(dataset).transform(dataset)
50 df = pd.DataFrame(te_ary, columns=te.columns_)
51 df.head()
52 print(df)
53
54 #fungsi apriori dari library mlxtend
55 #disini contoh dengan minimum support 1.5%
56 frq_items = apriori(df, min_support = 0.015, use_colnames = True)
57 print(frq_items)

```

```

58
59 #mencari aturan asosiasi
60 rules = association_rules(freq_items)
61 rules = rules.sort_values(['confidence'], ascending =[True])
62 rules2 = rules[ (rules['confidence'] >= 0.8) ]
63
64 #pencatatan hasil kombinasi item ke file
65 #deklarasi file system
66 original_stdout = sys.stdout
67
68 #proses penulisan hasil ke file teks
69 with open('result_frequentitemsets_0_015.txt', 'w') as f:
70     sys.stdout = f
71     print(rules2)
72     sys.stdout = original_stdout
73

```

Kode 2. Kode Pencarian *frequent itemset* Dari *UK Retail Online Tansaction* Dengan Python. Percobaan Dilakukan Berkali-kali Dengan Jumlah Data Berbeda dan Variabel yang Berbeda

3.6 Pattern Evaluation

Confidence aturan A->C adalah probabilitas untuk melihat konsekuensi(C) dalam transaksi yang juga mengandung anteseden(A). Perhatikan bahwa metrik tidak simetris atau diarahkan, misalnya, *Confidence* untuk A->C berbeda dari kepercayaan untuk C->A. *Confidence* bernilai 1 (maksimal) untuk aturan A->C jika konsekuensi dan anteseden selalu terjadi bersamaan.

Metrik *Lift* biasanya digunakan untuk mengukur seberapa sering anteseden(A) dan konsekuensi(C) dari aturan A->C terjadi bersama-sama daripada yang kita harapkan jika mereka independen secara statistik. Jika A dan C independen, skor Peningkatan akan tepat 1.

Nilai *Leverage* menghitung perbedaan antara frekuensi pengamatan A dan C yang muncul bersama-sama dan frekuensi yang diharapkan jika A dan C tidak saling bergantung. Nilai leverage 0 menunjukkan independensi (saling bebas).

Nilai *Conviction* yang tinggi berarti konsekuensinya sangat tergantung pada anteseden. Misalnya, dalam kasus skor *Conviction* sempurna, hasilnya menjadi 0 (karena $1 - 1$) di mana skor *Conviction* didefinisikan sebagai 'inf'. Mirip dengan *lift*, jika item independen, keyakinannya adalah 1.

Uji coba dilakukan dengan mengubah nilai minimum support untuk setiap kasus tetapi kita akan membatasi nilai confidence diatas 80%. Nilai-nilai minimum support yang diujikan adalah 1.5%, 2%, dan 3%. Berikut ini akan dibahas terlebih dahulu hasil dari setiap kasus uji coba.

1) Minimum Support = 1.5%

Nilai minimum support 1.5% menunjukkan setidaknya item-item tersebut muncul 7.989 kali dari 532.602 transaksi. Jumlah kumpulan item ini berjumlah 610 kombinasi. Itemset dengan nilai confidence mulai dari 80% keatas berjumlah 11 dan dapat dilihat pada tabel 2.

Tabel 1. Item Dengan Minimum Support 1.5%

	support	itemsets
0	0.023542	(15036)
1	0.020684	(15056N)
2	0.020321	(20676)
3	0.015105	(20677)

4	0.031616	(20685)
..
596	0.017146	(22411, 85099B, 22386)
597	0.015558	(22698, 22423, 22697)
598	0.018507	(22699, 22423, 22697)
599	0.015014	(22698, 22699, 22423)
600	0.024585	(22698, 22699, 22697)

[601 rows x 2 columns]

Tabel 2. Kumpulan Item Dengan Minimum Support 1.5% Dan Confidence >= 80%

Antecedents	Consequent	Antecedent Support	Consequent Support	Support	Confidence	Lift	Leverage	Conviction
(22698, 22699)	(22697)	0.027170	0.046040	0.024585	0.904841	19.653334	0.023334	10.024947
(22698, 22423)	(22697)	0.017781	0.046040	0.015558	0.875000	19.005172	0.014740	7.631679
(22698, 22697)	(22699)	0.028713	0.048353	0.024585	0.856240	17.707945	0.023197	6.619695
(22698, 22423)	(22699)	0.017781	0.048353	0.015014	0.844388	17.462826	0.014154	6.115499
(22698)	(22697)	0.034791	0.046040	0.028713	0.825293	17.925534	0.027111	5.460353
(21086)	(21094)	0.019414	0.023905	0.015876	0.817757	34.209243	0.015412	5.356011
(22411, 22386)	(85099B)	0.021228	0.094892	0.017146	0.807692	8.511656	0.015132	4.706559
(20723, 22355)	(20724)	0.018824	0.046902	0.015150	0.804819	17.159619	0.014267	4.883157
(84596F)	(84596B)	0.019731	0.023678	0.015831	0.802299	33.884062	0.015363	4.938374
(22386, 21931)	(85099B)	0.023360	0.094892	0.018734	0.801942	8.451055	0.016517	4.569906
(22423, 22697)	(22699)	0.023088	0.048353	0.018507	0.801572	16.577345	0.017390	4.795922

2) Minimum Support = 2%

Nilai minimum support 2% menunjukkan setidaknya item-item tersebut muncul 10.652 kali dari 532.602 transaksi. Jumlah kumpulan item ini berjumlah 310 kombinasi. Itemset dengan nilai confidence mulai dari 80% keatas berjumlah 3 dan dapat dilihat pada tabel 4.

Tabel 3. Kumpulan Item Dengan Minimum Support 2%

support	itemsets
0	0.023542 (15036)
1	0.020684 (15056N)
2	0.020321 (20676)
3	0.031616 (20685)
4	0.023814 (20711)
..	...
305	0.022907 (82482, 82494L)
306	0.026535 (85099B, 85099C)
307	0.024358 (85099F, 85099B)
308	0.021999 (85099B, DOT)
309	0.024585 (22697, 22698, 22699)

[310 rows x 2 columns]

Tabel 4. Kumpulan Item Dengan Minimum Support 2% Dan Confidence \geq 80%

Antecedents	Consequent	Antecedent Support	Consequent Support	Support	Confidence	Lift	Leverage	Conviction
(22698, 22699)	(22697)	0.027170	0.046040	0.024585	0.904841	19.653334	0.023334	10.024947
(22697, 22698)	(22699)	0.028713	0.048353	0.024585	0.856240	17.707945	0.023197	6.619695
(22698)	(22697)	0.034791	0.046040	0.028713	0.825293	17.925534	0.027111	5.460353

3) Minimum Support = 3%

Nilai minimum support 3% menunjukkan setidaknya item-item tersebut muncul 15.978 kali dari 532.602 transaksi. Jumlah kumpulan item ini berjumlah 114 kombinasi.

Tabel 5. Kumpulan Item Dengan Minimum Support 3%

	support	itemsets
0	0.031616	(20685)
1	0.039145	(20712)
2	0.030300	(20713)
3	0.037875	(20719)
4	0.032750	(20723)
..
109	0.051256	(POST)
110	0.032840	(85099B, 21931)
111	0.037422	(85099B, 22386)
112	0.030845	(85099B, 22411)
113	0.034836	(22697, 22699)

[114 rows x 2 columns]

Hasil pasangan frequent itemset dengan minimal confidence 80% tidak ada. Hasil yang dikembalikan oleh eksekusi python adalah kosong dan dapat dilihat pada tabel 6. Hal ini menunjukkan bahwa semakin tinggi supportnya maka tidak menjamin menghasilkan itemset dengan nilai confidence tinggi.

Tabel 6. Kumpulan Item Dengan Minimum Support 3% Dan Confidence \geq 80%

Empty DataFrame
Columns: [antecedents, consequents, antecedent support, consequent support, support, confidence, lift, leverage, conviction]
Index: []

3.6 Knowledge Representation

Hasil uji coba ini berupa aturan asosiasi yang menentukan kemungkinan terjadinya konsekuensi B akan dibeli jika pada transaksi tersebut A sudah dibeli. Aturan yang didapatkan dengan jumlah pembelian produk minimal pembelian 7.989 kali adalah:

- 1) Jika pelanggan membeli {PINK REGENCY TEACUP AND SAUCER dan ROSES REGENCY TEACUP AND SAUCER} => maka akan membeli {GREEN REGENCY TEACUP AND SAUCER} dengan probabilitas 90% dan lift 19.65
- 2) Jika pelanggan membeli {PINK REGENCY TEACUP AND SAUCER dan REGENCY CAKESTAND 3 TIER} => maka pelanggan akan membeli {GREEN REGENCY TEACUP AND SAUCER} dengan confidence 87.5% dan lift 19.00

- 3) Jika pelanggan membeli {PINK REGENCY TEACUP AND SAUCER dan GREEN REGENCY TEACUP AND SAUCER} => maka akan membeli {ROSES REGENCY TEACUP AND SAUCER} dengan confidence 85.6% dan lift 17.70
- 4) Jika pelanggan membeli {PINK REGENCY TEACUP AND SAUCER dan REGENCY CAKESTAND 3 TIER} => maka akan membeli {ROSES REGENCY TEACUP AND SAUCER} dengan confidence 84.4% dan lift 17.46.
- 5) Jika pelanggan membeli {PINK REGENCY TEACUP AND SAUCER} => maka akan membeli {GREEN REGENCY TEACUP AND SAUCER} dengan confidence 82.5% dan lift 17.92.
- 6) Jika pelanggan membeli {SET/6 RED SPOTTY PAPER CUPS} => maka akan membeli {SET/6 RED SPOTTY PAPER PLATES} dengan confidence 81.77% dan lift 34.2.
- 7) Jika pelanggan membeli {JUMBO SHOPPER VINTAGE RED PAISLEY dan JUMBO BAG PINK POLKADOT} => maka akan membeli {JUMBO BAG RED RETROSPOT} dengan confidence 80.76% dan lift 8.5.
- 8) Jika pelanggan membeli {STRAWBERRY CHARLOTTE BAG dan CHARLOTTE BAG SUKI DESIGN} => maka akan membeli {RED RETROSPOT CHARLOTTE BAG} dengan confidence 80.48% dan lift 17.15.
- 9) Jika pelanggan membeli {SMALL MARSHMALLOWS PINK BOWL} => maka akan membeli {SMALL DOLLY MIX DESIGN ORANGE BOWL} dengan confidence 80.22% dan lift 33.88.
- 10) Jika pelanggan membeli {JUMBO BAG PINK POLKADOT dan JUMBO STORAGE BAG SUKI} => maka akan membeli {JUMBO BAG RED RETROSPOT} dengan confidence 80.19% dan lift 8.45.
- 11) Jika pelanggan membeli {REGENCY CAKESTAND 3 TIER dan GREEN REGENCY TEACUP AND SAUCER} => maka akan membeli {ROSES REGENCY TEACUP AND SAUCER} dengan confidence 80.15% dan lift 16.57.

Aturan yang didapatkan dengan jumlah pembelian produk minimal pembelian 10.652 kali adalah:

- 1) Jika pelanggan membeli {PINK REGENCY TEACUP AND SAUCER dan ROSES REGENCY TEACUP AND SAUCER} => maka akan membeli {GREEN REGENCY TEACUP AND SAUCER} dengan confidence 90% dan lift 19.65
- 2) Jika pelanggan membeli {GREEN REGENCY TEACUP AND SAUCER dan PINK REGENCY TEACUP AND SAUCER} => maka pelanggan akan membeli {ROSES REGENCY TEACUP AND SAUCER} dengan confidence 85.62% dan lift 17.70
- 3) Jika pelanggan membeli {PINK REGENCY TEACUP AND SAUCER} => maka akan membeli {GREEN REGENCY TEACUP AND SAUCER} dengan confidence 82.5% dan lift 17.92.

Hasil dari uji coba ini dapat digunakan kepentingan bisnis di online retail untuk merekomendasikan produk yang sering dibeli bersamaan dengan kepastian tinggi.

IV. PENUTUP

Dari penelitian ini dapat disimpulkan sebagai berikut:

- 1) Didapatkan 11 aturan asosiasi untuk kumpulan Item dengan minimum support 1.5% dan confidence $\geq 80\%$. Didapatkan 3 aturan asosiasi untuk kumpulan Item dengan minimum support 2% dan confidence $\geq 80\%$
- 2) Implementasi kode python dapat dilakukan dengan penambahan beberapa pustaka yaitu:
 - mlxtend, pustaka ini mempunyai fungsi-fungsi lengkap untuk algoritma apriori dan mencari aturan asosiasi.

- pandas, pustaka ini digunakan untuk membaca input dataset dan melakukan transformasi data ke dataframe.
- sys, pustaka ini digunakan untuk mencatat output ke *file* teks.

DAFTAR PUSTAKA

- Aprianti, Winda, Anwar Hafizd, Khairul dan Rizani, M. Redhy. (2017). Implementasi Association Rules dengan Algoritma Apriori pada Dataset Kemiskinan. *J. Math. and Its Appl.* E-ISSN: 2579-8936, P-ISSN: 1829-605X Vol. 14, No. 2, Desember 2017, 145-155.
- Charlet Annie M.C, Loraine dan Kumar D, Ashok. (2012). Market Basket Analysis for a Supermarket based on Frequent Itemset Mining. *IJCSI International Journal of Computer Science Issues*, Vol. 9, Issue 5, No 3, September 2012.
- Gupta, Savi and Mamtora, Roopal. (2014). A Survey on Association Rule Mining in Market Basket Analysis. *International Journal of Information and Computation Technology* ISSN 0974-2239 Volume 4, Number 4 (2014), pp. 409-414.
- J. Han dan M. Kamber. (2006). *Data Mining Concept and Techniques*, 2nd edition, USA: Elsevier, Inc, 2006.
- Kaur, Manpreet dan Kangs, Shivani. (2016). Market Basket Analysis: Identify the changing trends of market data using association rule mining. *International Conference on Computational Modeling and Security (CMS 2016)* Elsevier.
- Kurnia, Yusuf; Isharianto, Yohanes; Ceng Giap, Yo; Hermawan, Aditiya dan Riki. (2019). Study of application of data mining market basket analysis for knowing sales pattern (association of items) at the O! Fish restaurant using apriori algorithm. *J. Phys.: Conf. Ser.* 1175 012047
- Kurniawan, Fachrul; Umayah, Binti; Hammad, Jihad; Mardi Susiki Nugroho, Supeno dan Hariadi, Mochammad. (2018). Market Basket Analysis to Identify Customer Behaviors by Way of Transaction Data. *Knowledge Engineering and Data Science (KEDS)* Vol 1, No 1, January 2018, pp. 20–25.
- Sağın, Ayşe Nur dan Ayvaz, Berk. (2018). Determination of Association Rules with Market Basket Analysis: An Application in the Retail Sector. *Southeast Europe Journal of Soft Computing* VOL.7 NO.1 March 2018 - ISSN 2233 – 1859.
- Waghare SS, Sonar S, Kawad S et al. (2018). Apriori algorithm using map reduce. *International Journal of Research and Review*. 2018; 5(5):129-132.
- Wahyudi Oktavia Gama, Adie; Gede Darma Putra, I Ketut dan Agung Bayupati, I Putu. (2016). Implementasi Algoritma Apriori Untuk Menemukan Frequent Itemset Dalam Keranjang Belanja. *Teknologi Elektro*, Vol. 15, No.2, Juli - Desember 2016.