

SISTEM REKOMENDASI PAKET MAKANAN MENGGUNAKAN ALGORITMA APRIORI PADA PENYETAN BU TINI

Ade Pujiyanto¹⁾, Selvy Megira²⁾, Hasnan Afif³⁾, Kusri⁴⁾

¹⁾²⁾³⁾⁴⁾ Magister Teknik Informatika, Universitas AMIKOM Yogyakarta

Jl. Ring Road Utara, Condong Catur, Depok, Sleman, Yogyakarta 55281

Email : adepujiyanto@gmail.com¹⁾, selvy.megira@students.amikom.ac.id²⁾, hasnan.afif@gmail.com³⁾

Abstrak

Setiap penyedia jasa kuliner pasti memiliki history data transaksi pelanggan dalam proses kegiatannya. Dalam sebuah transaksi hampir semua pelanggan memilih lebih dari satu jenis makanan. Begitupun dengan rumah makan Penyetan Bu Tini dimana data transaksi yang tidak diolah hanya akan menumpuk dan tidak bermanfaat. Sebaliknya pengolahan data yang tepat menggunakan data mining akan bermanfaat dan dapat mendukung suatu pengelola dalam mengambil keputusan. Berdasarkan hal ini peneliti mencoba melakukan penelitian pada Penyetan Bu Tini dengan mengolah data transaksi dan mengekstrak informasi seperti paket menu manakah yang sering dipesan. Untuk melakukan penelitian ini peneliti menggunakan algoritma Apriori dalam pembuatan sistem rekomendasi paket makanan. Adanya penelitian ini diharapkan dapat menjadi pendukung keputusan pengelola dalam membuat dan memberikan rekomendasi paket menu makanan bagi pelanggan baru.

Kata kunci: Algoritma Apriori, Sistem Rekomendasi, Data Mining

1. Pendahuluan

Dengan meningkatnya mobilitas masyarakat, tempat-tempat kuliner ini menjadi alternatif bagi orang-orang yang tidak punya waktu untuk makan di rumah. Setiap orang memiliki kriteria sendiri untuk memilih tempat makan. Selain itu dalam penyusunan layout makanannya, Penyetan Bu Tini masih melakukannya secara acak dan belum sesuai dengan pola kebiasaan konsumen dalam mengambil makanan.

Sehingga dapat mengurangi kenyamanan dan kepuasan konsumen terhadap Penyetan Bu Tini yang dapat berdampak pada pendapatan pihak pengelola rumah makan menjadi tidak maksimal. Selain itu, di rumah makan ini belum terdapat paket makanan hemat yang dapat ditawarkan kepada konsumen dalam meminimalisir terjadinya antrian panjang pada waktu-waktu tertentu setiap harinya, terutama pada waktu makan siang.

Pada penelitian yang dilakukan oleh Belladonna Shelly Agasti dkk yang berjudul "Sistem Rekomendasi Kuliner Untuk Mahasiswa Universitas Sebelas Maret Surakarta Menerapkan Algoritma Apriori Positif Negatif Dan

Binary Hamming Distance" (2015), penelitian ini bertujuan untuk membuat sistem rekomendasi yang dapat membantu mahasiswa memilih tempat makan sesuai dengan kriteria yang diinginkan mahasiswa itu sendiri. Berdasarkan hasil penelitian *binary hamming distance* dapat diaplikasikan bersama algoritma Apriori untuk mengukur nilai kedekatan nilai *input user* dengan nilai suatu tempat makan. Dua metode ini dapat memberikan rekomendasi tempat makan yang baik [5].

Pada tahun 2014, Elsa Widiati dan Kania Evita Dewi, juga melakukan penelitian yang berjudul "Implementasi Association Rule Terhadap Penyusunan Layout Makanan Dan Penentuan Paket Makanan Hemat di RM Roso Echo Dengan Algoritma Apriori". Tujuan penelitian ini adalah menerapkan *data mining* untuk mendapatkan susunan makanan yang sesuai dengan kebiasaan pembeli dan susunan paket hemat yang mungkin dibentuk. Berdasarkan hasil *association rule* dengan minimum *support* 30% dan minimum *confidence* 62% mampu memberikan informasi kepada pihak pengelola RM. Roso Echo mengenai susunan *layout* makanan dan paket makanan hemat yang dapat diterapkan [6].

Penelitian selanjutnya dilakukan oleh Wildan Budiawan Zulfikar dkk pada tahun 2016, yang berjudul "Implementation Of Association Rules With Apriori Algorithm For Increasing The Quality Of Promotion" yang bertujuan untuk mengimplementasikan *Association rule* dengan apriori untuk meningkatkan kualitas promosi atau meningkatkan penjualan produk dengan promosi yang menghasilkan nilai *support* dan *confidence* yang mempengaruhi kualitas aturan. sehingga untuk menetapkan nilai *minimum support* dan *confidence* yaitu *minimum support* adalah 12% dan *minimal confidence* adalah 60% untuk mendapatkan peraturan yang berpotensi menghasilkan promosi yang berkualitas. Item akan dipromosikan untuk disesuaikan dengan potensi penjualan toko. Lokasi toko mempengaruhi aturan asosiasi. Toko dibagi menjadi 8 kelompok, aturan dapat diterapkan dalam *cluster* tetapi tidak dapat diterapkan pada *cluster* lain. Aturannya bisa dijadikan bahan promosi sebagai terobosan untuk meningkatkan penjualan [7].

Berdasarkan hal yang telah dijelaskan di atas, dibutuhkan suatu algoritma terapan untuk membuat sistem rekomendasi yang dapat membantu pelanggan

memilih menu makanan sesuai dengan kriteria yang diinginkan pelanggan itu sendiri.

Salah satu metode yang ada dalam *data mining* adalah metode *association rule* dimana metode ini mencari sekumpulan items yang sering muncul bersamaan [1]. Metode ini sering dianalogikan dengan keranjang belanja dimana dapat diketahui, barang apa saja yang sering dibeli bersamaan dan barang mana saja yang tidak. Keuntungan metode ini dapat dimanfaatkan dalam penentuan paket makanan ataupun stok barang pada suatu badan usaha. Dalam penelitian ini akan dibuat sebuah sistem data mining yang akan melihat kecenderungan pembeli pada Penyetan Bu Tini sehingga pengambil keputusan dapat menentukan paket makanan yang sesuai dengan kebutuhan pelanggan.

Data Mining

Data mining merupakan analisis dari peninjauan suatu kumpulan data dengan tujuan menemukan pola atau hubungan yang tidak diketahui dan meringkas data dengan cara yang berbeda dengan sebelumnya, yang dapat dipahami dan bermanfaat bagi pemilik data [2]. Dengan melakukan penggalian pola-pola dari data dalam tujuan melakukan manipulasi data menjadi informasi yang lebih dapat digunakan dalam mendukung keputusan, dengan cara mengekstraksi dan mengenali pola yang penting atau menarik dari data yang terdapat dalam basis data. *Data mining* juga dikenal dengan nama *Knowledge Discovery (mining) in Database (KDD)*, umum, definisi *data mining* dapat diartikan sebagai berikut [3]

1. Proses penemuan pola yang menarik dari data yang tersimpan dalam jumlah besar.
2. Ekstraksi dari suatu informasi yang berguna atau menarik (*non-trivial*, implisit, sebelumnya belum diketahui potensian kegunaannya) pola atau pengetahuan dari data yang di simpan dalam jumlah besar.
3. Eksplorasi dari analisa secara otomatis atau semiotomatis terhadap data-data dalam jumlah besar untuk mencari pola dan aturan yang berarti.

Association Rule

Association rule merupakan teknik *data mining* untuk menemukan aturan asosiatif antara suatu kombinasi item atau mencari hubungan antar-item pada suatu *dataset* dan menampilkan bentuk *association rule*. *Association rule* (aturan asosiasi) akan menemukan pola tertentu dalam mengasosiasikan data yang satu dengan data yang lain. Metodologi dasar dalam mencari *association rule* dari suatu kumpulan data menjadi dua tahap, tahap pertama yang harus dilakukan adalah mencari *frequent itemset* terlebih dahulu. *Frequent itemset* adalah sekumpulan *item* yang sering muncul secara bersamaan yang memenuhi syarat minimum dari nilai *support* dalam *database*. Setelah semua pola *frequent itemset* ditemukan, barulah mencari aturan asosiatif atau aturan

keterkaitan yang memenuhi syarat *confidence* dengan menghitung *confidence* aturan asosiatif [1].

Pola tersebut dirumuskan dalam sebuah *association rule*. Sebagai contoh konsumen biasanya akan membeli Susu dan gula yang ditunjukkan sebagai berikut :

Susu \rightarrow gula [*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 gula dan pasti membeli susu sebesar 60%.

Algoritma Apriori

Algoritma apriori adalah salah satu algoritma yang melakukan pencarian *frequent itemset* dengan menggunakan teknik *association rule*. Algoritma apriori menggunakan pengetahuan *frekuensi* atribut yang telah diketahui sebelumnya untuk memproses informasi selanjutnya. Pada algoritma apriori, menentukan kandidat yang mungkin muncul dengan cara memperhatikan minimum *support* dan minimum *confidence*. *Support* adalah nilai pengujung atau persentase kombinasi sebuah *item* dalam *database*. nilai *support* sebuah item diperoleh dengan rumus 1 berikut [4] :

$$Support(A) = \frac{\text{Jumlah transaksi mengandung A}}{\text{Total transaksi}} \times 100\% \quad (1)$$

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

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

Sedangkan nilai *confidence* merupakan nilai kepastian kuatnya hubungan antar item pada apriori. *Confidence* dapat dicari setelah pola frekuensi munculnya sebuah item ditemukan. Untuk menghitung *confidence* dapat digunakan rumus 3 berikut:

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

Proses perhitungan *association rule* dengan apriori terdiri dari beberapa tahap adalah sebagai berikut [4]:

1. Sistem melakukan *scan database* untuk mendapatkan kandidat 1-*itemset* (himpunan *item* yang terdiri dari 1 *item*) dan menghitung nilai *support*-nya. Kemudian nilai *support*-nya tersebut akan dibandingkan dengan *minimum support* yang telah ditentukan. Apabila nilainya lebih besar atau

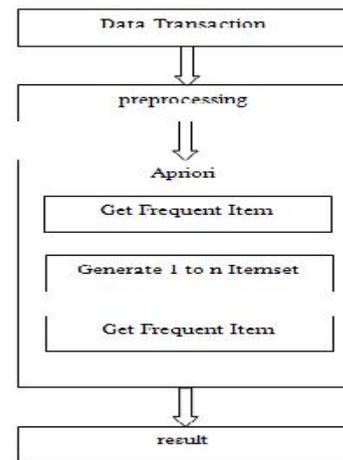
sama dengan *minimum support*, maka *itemset* tersebut termasuk *large itemset*.

2. *Itemset* yang tidak termasuk dalam *large itemset* tidak diikuti dalam iterasi selanjutnya (dipangkas).
3. 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 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 (memangkas) kombinasi *itemset* yang tidak termasuk dalam *large itemset*.
4. Setelah dilakukan operasi *join*, maka pasangan *itemset* baru hasil proses *join* tersebut dihitung *support*-nya.
5. Proses pembentuk kandidat yang terdiri dari proses penggabungan dan memangkas akan terus dilakukan hingga himpunan kandidat *itemset*-nya *null*, atau sudah tidak ada lagi kandidat yang akan terbentuk.
6. Setelah itu, dari hasil *frequent itemset* tersebut dibentuk *association rule* yang memenuhi nilai *support* dan *confidence* yang telah ditentukan.
7. Pada pembentukan *association rule*, nilai yang sama dianggap sebagai satu nilai.
8. *Association rule* yang terbentuk harus memenuhi nilai *minimum* yang telah ditentukan.
9. Untuk setiap *large itemset* L, cari himpunan bagian L yang tidak kosong. Untuk setiap himpunan bagian tersebut, dihasilkan *rule* dengan bentuk $aB(L-a)$ jika *support*-nya (a) lebih besar dari *minimum support*.

2. Pembahasan

Diagram Alir data

Diagram alir data menggambarkan proses kerja perhitungan model algoritma apriori pada sistem seperti pada Gambar 1 pada tahapan data *transaction* merupakan awal dalam menentukan data transaksi pada rumah makan Penyetan Bu Tini kemudian data tersebut akan dilakukan *preprocessing* untuk menyesuaikan format database terhadap sistem setelah itu akan dilakukan pencarian frekuensi kemunculan *itemset*, lalu akan dilakukan *generate* setiap item kedalam satu sampai n *itemset* untuk mendapatkan nilai frekuensi disetiap *itemset*. Pada tahapan *result* akan menghasilkan output berupa layout dan paket makanan.



Gambar 1. Diagram Alir Data

Pada subbab ini yang akan dijelaskan contoh kasus menggunakan algoritma apriori untuk mencari *frequent itemset*-nya. Adapun dalam tahapan proses pengerjaan algoritma apriori pada penelitian ini adalah sebagai berikut :

1. Data yang digunakan adalah data hasil *preprocessing* seperti pada Tabel 1.

Tabel 1. Data transaksi

id transaksi	Item
T0001	Bebek, Ayam, Nasi, Terong
T0002	Ayam, Nasi, Terong
T0003	Ayam, Terong
T0004	Bebek, Nasi, Terong

2. Tentukan item-item yang dibeli dalam data transaksi pada Tabel 2.

Tabel 2. Item-item yang dibeli

Item yang dibeli
Bebek
Ayam
Nasi
Terong

3. Mencari kandidat 1-*itemset* dan hitung nilai *Support*-nya sehingga memperoleh hasil seperti pada Tabel 3.

Tabel 3. Kandidat 1-*itemset*

1-Itemset	Support	Confidence
Bebek	2	66,6667
Ayam	3	100
Nasi	3	100
Terong	3	100

- Pangkas data yang memiliki nilai *support* lebih kecil dari nilai *minimum support*. Misal nilai *minimum confidence* = 75% dengan hasil pada Tabel 4.

Tabel 4. Large 1-itemset

1-Itemset	Support	Confidence
Ayam	3	100
Nasi	3	100
Terong	3	100

- Setelah itu lakukan penggabungan dari setiap data yang ada pada Tabel 5 untuk mendapatkan kandidat 2-itemset.

Tabel 5. Kandidat 2-itemset

2-Itemset	Support	Confidence
Bebek, Nasi	2	66,6667
Ayam, Nasi	2	66,6667
Ayam, Terong	3	100
Nasi, Terong	3	100

- Lakukan pemangkasan kembali terhadap data yang tidak memenuhi nilai *minimum confidence* dengan hasil pada Tabel 6.

Tabel 6. Large 2-itemset

2-Itemset	Support	Confidence
Ayam, Terong	3	100
Nasi, Terong	3	100

- Lakukan proses penggabungan dan pemangkasan hingga tidak ada lagi data yang dapat digabungkan.
- Selanjutnya hitung nilai *confidence* dari setiap *large itemset* yang didapat, mulai dari *large 2-itemset*.
- Pangkas data yang memiliki nilai *confidence* lebih kecil dari nilai *minimum confidence*. Misal nilai *minimum confidence* = 100% seperti pada Tabel 7.

Tabel 7. Confidence 2-itemset

3-Itemset	Support	Confidence
Bebek, Ayam, Nasi	1	33,3333
Bebek, Ayam, Terong	1	33,3333
Bebek, Nasi, Terong	1	33,3333
Ayam, Nasi, Terong	2	66,6667

- Setelah nilai *confidence* telah didapat, selanjutnya adalah mengurutkan data yang memiliki nilai *support* dan nilai *confidence* yang terbesar hingga terkecil untuk mendapatkan *layout* makanan seperti pada Tabel 8.

Tabel 8. Susunan Layout Makanan

Layout Letak Makanan	
Urutan	Makanan
1	Nasi
2	Ayam
3	Terong
4	Bebek

- Hitung nilai *lift* dari data yang memiliki nilai *support* dan nilai *confidence* yang memenuhi nilai *minimum support* dan nilai *minimum confidence* untuk mendapatkan informasi paket makanan hemat seperti pada Tabel 9.

Tabel 9. Paket Makanan Hemat

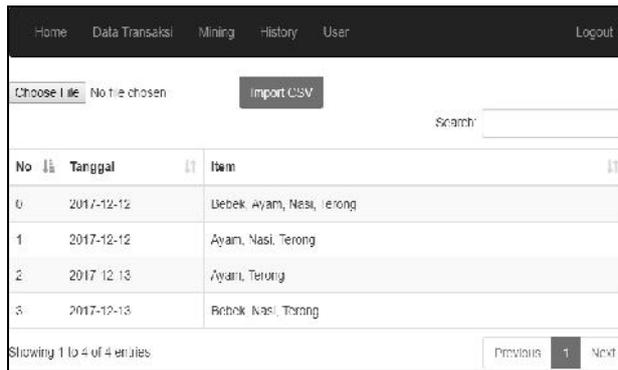
Paket Makanan	
No	Rekomendasi
1	Ayam, Terong
2	Nasi, Terong
3	Ayam, Nasi, Terong

Berdasarkan hasil *association rule* dengan menggunakan metode apriori dengan minimum *support* 3 dan minimum *confidence* 100% diperoleh susunan yaitu: Nasi, Ayam, Terong, Bebek. Sedangkan untuk paket makanan hemat bergantung kebutuhan dengan hasil pemilihan pake pada 2 *itemset* yaitu menu Ayam dan Terong serta Nasi dan Terong, kemudian pada 3 *itemset* di peroleh menu Ayam, Nasi dan Terong.

Antarmuka Sistem

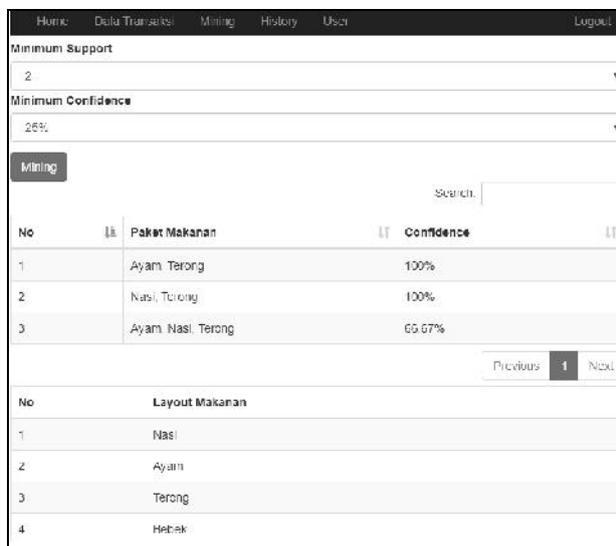
Tampilan antarmuka pada sistem rekomendasi paket makanan dengan menggunakan algoritma *apriori* dapat dilihat pada Gambar 2 dan Gambar 3. Sebagai contoh yang akan ditampilkan pada antarmuka sistem, akan menggunakan data pada pembahasan sebelumnya yaitu:

Pada Gambar 2 merupakan hasil tampilan data transaksi dengan id transaksi T0001, T0002, T0003, T0004. Pada Gambar 3 merupakan hasil tampilan sistem rekomdasi pake makanan ini berupa tampilan layout makanan serta rekomendasi paket makanan.



Gambar 2. Antarmuka Data Transaksi

Gambar 2 merupakan antarmuka untuk admin dalam mengelola atau memasukan data transaksi dalam format CSV (*Comma Separated Values*).



Gambar 3. Antarmuka Hasil Rekomendasi Paket dan Layout Makanan

Gambar 3 merupakan antarmuka untuk admin untuk melakukan proses penentuan paket makanan dengan pilihan parameter *Minimum Support* dan *Minimum Confidence*.

3. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan algoritma apriori dapat menjadi solusi pada kelemahan algoritma asosiasi rule dalam menentukan *frequent* itemset. Tahapan dalam menentukan *frequent* itemset adalah tahap yang sangat berpengaruh pada performa data mining terutama jika kandidat itemset berjumlah sangat besar. Pada algoritma apriori hanya dilakukan 1 kali *input* database yaitu pada awal iterasi. Pada tahap selanjutnya itemset tersebut akan dimasukkan kedalam iterasi *itemset* pada tabel apriori dengan menggunakan operasi apriori terhadap ketentuan setiap alamat iterasi diisi oleh 1 *itemset*. Dengan begitu, performa data

mining dapat meningkat dan pada setiap iterasi langsung dilakukan eliminasi terhadap iterasi itemset yang tidak memenuhi minimum *support count*.

Daftar Pustaka

- [1] Kusrini dan Emha T Luthfi, *Algoritma Data Mining*, Yogyakarta, Andi Offset, 2009.
- [2] Larose, Daniel T, *Discovering Knowledge in Data: An Introduction to Data Mining*, John Wiley & Sons, Inc, 2005
- [3] J. Han and M. Kamber, *Data Mining : Concept and Techniques*, 2nd ed. San Fransisco: Morgan Kauffman, 2006.
- [4] R. S. Pressman, *Software Engineering – A Practitioner’s Approach, 5th ed*, McGraw-Hill International, 2001
- [5] Shelly Agasti, Belladona, dkk, *Sistem Rekomendasi Kuliner Untuk Mahasiswa Universitas Sebelas Maret Surakarta Menerapkan Algoritma Apriori Positif Negatif Dan Binary Hamming Distance*, Seminar Nasional Ilmu Komputer (SNIK), 2015.
- [6] Widiati, Elsa dan Evita Dewi, Kania, *Implementasi Association Rule Terhadap Penyusunan Layout Makanan Dan Penentuan Paket Makanan Hemat Di Rm Roso Echo Dengan Algoritma Apriori*, Jurnal Ilmiah Komputer dan Informatika (KOMPUTA), Vol. 3, No. 2, 2014.
- [7] Zulfikar, Wildan Budiawan, dkk, *Implementation Of Association Rules With Apriori Algorithm For Increasing The Quality Of Promotion*, 4th International Conference on Cyber and IT Service Management, 2016.

Biodata Penulis

Ade Pujiyanto, Saat ini menjadi Mahasiswa di Magister Teknik Informatika Universitas AMIKOM Yogyakarta.

Selvy Megira, Saat ini menjadi Staff di AMIK Lembah Dempo, Sumatera Selatan.

Hasnan Afif, Saat ini menjadi Mahasiswa di Magister Teknik Informatika Universitas AMIKOM Yogyakarta.

Kusrini, Saat ini menjadi Dosen di Universitas AMIKOM Yogyakarta.

