

Komparasi Algoritma Pincer Search dan Algoritma FP-Growth pada Stok Barang di Toko X

Comparison of Pincer Search Algorithm and FP-Growth Algorithm on Goods Stock at Shop X

Putu Ratih Wulandari¹, I Made Agus Dwi Suarjaya², Ni Kadek Dwi Rusjyanthi³

^{1,2,3}Jurusan Teknologi Informasi, Fakultas Teknik, Universitas Udayana

E-mail: ¹ratihwulandarii21@gmail.com, ²agussuarjaya@it.unud.ac.id,

³dwi.rusjyanthi@unud.ac.id

Abstrak

Jumlah pembelian barang setiap harinya berbeda-beda karena itu permasalahan kekurangan stok barang dapat terjadi dan mengakibatkan ketidakpuasan pelanggan dalam berbelanja karena tidak tersedianya produk yang diinginkan. Permasalahan kekurangan stok barang dapat diminimalisir dengan melakukan penelitian mengenai data mining asosiasi menggunakan data transaksi penjualan dari Toko X berdasarkan metode algoritma *pincer search* dan algoritma *FP-Growth*. Penelitian ini bertujuan untuk mendapatkan *association rule* dan jumlah kemunculan *frequent item set* dalam data transaksi melalui minimum *support* yang dimanfaatkan untuk mengatasi permasalahan kekurangan stok barang di Toko X serta melakukan komparasi algoritma *pincer search* dan algoritma *FP-Growth* terhadap waktu pemrosesan data, *frequent item set*, *rule*, *confidence* dan *lift ratio* dengan bahasa pemrograman Python. Komparasi algoritma *pincer search* dan algoritma *FP-Growth* terhadap *frequent item set*, *rule*, *confidence* dan *lift ratio* dengan bahasa pemrograman Python memperoleh hasil yang sama, tetapi waktu yang dibutuhkan dalam pemrosesan data berbeda yang disebabkan oleh minimum *support*, jumlah transaksi dan jumlah *item* serta alur proses data yang berbeda dari kedua metode.

Kata kunci: *Data Mining*, Aturan Asosiasi, Algoritma *Pincer Search*, Algoritma *FP-Growth*

Abstract

The number of purchases of goods every day is different because the problem of lack of stock of goods can occur and result in customer dissatisfaction in shopping because of the unavailability of the desired product. The stock goods shortages can minimize by conducting research association data mining using sales transaction data from Shop X based on the Pincer Search algorithm and the FP-Growth algorithm. The study aims to obtain association rule and the number of occurrences of frequent itemsets in transaction data through the minimum support value which uses to overcome the problem of lack of goods stock at Shop X and compare the Pincer Search algorithm and the FP-Growth algorithm on data processing time, frequent itemset, rule, confidence value, and lift ratio value using Python programming languages. The comparison of the pincer search algorithm and the FP-Growth algorithm against the frequent itemset, rule, confidence value, and the lift ratio value with Python programming language obtains the same results, but the time required for data processing is different due to the minimum support value, the number of transactions and the number of items and different data processing flows of the two methods.

Keywords: *Data Mining*, Association Rule, *Pincer Search Algorithm*, *FP-Growth Algorithm*

1. PENDAHULUAN

Manajemen stok barang merupakan sebuah strategi untuk memajukan pelayanan terhadap pelanggan sehingga dapat menjamin ketersediaan barang yang diperlukan [1]. Toko X

termasuk salah satu UMKM yang belum menerapkan manajemen pada stok barang sehingga kekurangan stok barang pernah terjadi pada bulan-bulan tertentu di Toko X. Permasalahan stok barang yang terjadi di Toko X dapat diminimalisir dengan melakukan data mining asosiasi menggunakan data transaksi dari Toko X berdasarkan algoritma *pincer search* dan algoritma *FP-Growth*.

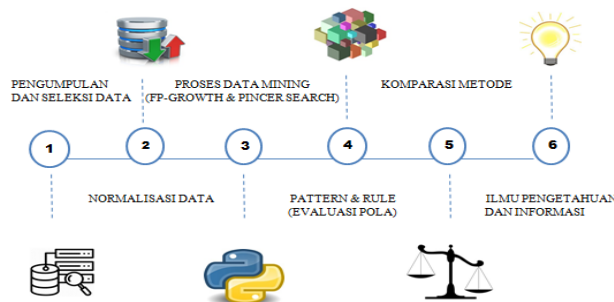
Data mining merupakan teknik perhitungan dengan kecerdasan buatan untuk menyaring data guna mendapatkan pengetahuan dan informasi dalam menentukan keputusan yang dapat dilakukan kedepannya [2,3]. Data mining tidak hanya dapat membantu dalam permasalahan asosiasi tetapi juga prediksi, deskripsi, estimasi, pengklasteran dan klasifikasi [4]. Data mining asosiasi menggunakan model *association rule* untuk mengetahui keterkaitan *item* dengan *item* yang lainnya. Data mining asosiasi berbasis algoritma *FP-Growth* yang menjadi salah satu algoritma yang dikembangkan dari algoritma apriori untuk menemukan *frequent item set* dengan membentuk *FP-Tree*, melakukan tahap *conditional pattern base*, tahap *conditional FP-Tree* serta mencari *frequent item set* [5,6]. Data mining asosiasi berbasis algoritma *pincer search* juga menjadi algoritma lainnya yang dikembangkan dari algoritma apriori dalam menemukan *frequent item set* yang memiliki cara yang hampir sama dengan algoritma apriori yaitu membentuk dan menghitung *support* dari kombinasi *item* tetapi algoritma *pincer search* menggunakan cara *bottom up* dan *top down* [7].

Penelitian data mining asosiasi menggunakan *association rule* juga pernah dilakukan oleh Anthony Anggrawan, Mayadi dan Chistofor Satria dengan algoritma apriori dan algoritma *FP-Growth*. Pengujian data mining dengan algoritma apriori dan algoritma *FP-Growth* dilakukan menggunakan data transaksi sebanyak 131 dengan jumlah item sebanyak 8. Data mining dengan algoritma *FP-Growth* menghasilkan nilai akurasi yang lebih besar dibandingkan dengan algoritma apriori. Nilai akurasi yang dihasilkan dari algoritma *FP-Growth* yaitu sebesar 178% sedangkan nilai akurasi yang dihasilkan dari algoritma apriori yaitu sebesar 78% [8].

Penelitian data mining asosiasi menggunakan *association rule* lain dilakukan oleh Imam Fahrur Rozi, Hendra Pradibta dan Syilvia Windy Kharisma Putri dengan algoritma *pincer search*. Data mining dengan algoritma *pincer search* menghasilkan *rule* terbanyak pada *maximum item* senilai 9 dan nilai minimum *support* sebesar 1% [9]. *Rule* yang dihasilkan memenuhi nilai minimum *lift ratio* dan bersifat valid karena memiliki nilai sama atau lebih dari 1. Penelitian mengenai komparasi algoritma *pincer search* dan algoritma *FP-Growth* dilakukan karena belum pernah ada yang membandingkan kedua metode tersebut untuk mengetahui pengaruh perbedaan parameter terhadap waktu pemrosesan data, *frequent item set*, *rule*, *confidence* dan *lift ratio*.

2. METODE PENELITIAN

Metode penelitian mengenai komparasi algoritma *pincer search* dan algoritma *FP-Growth* terbagi menjadi beberapa tahapan. Berikut metode penelitian mengenai komparasi algoritma *pincer search* dan algoritma *FP-Growth* yang dapat dilihat pada Gambar 1.



Gambar 1 Metode Penelitian Yang Digunakan

2.1 Pengumpulan dan Seleksi Data

Pengumpulan dan seleksi data menjadi tahapan awal dalam data mining untuk mendapatkan data transaksi pada Toko X dan data lainnya [10,11]. Data yang dikumpulkan dibagi menjadi dua yaitu data primer dan data sekunder [12]. Data primer dari penelitian data mining asosiasi pada Toko X yaitu data transaksi penjualan barang, sedangkan data sekunder dari penelitian data mining asosiasi pada Toko X yaitu data visi dan misi Toko X.

2.2 Normalisasi Data

Normalisasi data dilakukan saat data transaksi pada Toko X terkumpul untuk menghilangkan data yang tidak diperlukan dalam data mining asosiasi lalu merangkai kembali data tersebut menjadi dataset dengan format .csv dalam bentuk data biner. Dataset dengan format csv terdiri dari 1 dan 0 [13].

2.3 Data Mining (Algoritma Pincer Search dan Algoritma FP-Growth)

Data mining berdasarkan algoritma *pincer search* dan algoritma *FP-Growth* dengan bahasa pemrograman Python terdapat 3 nilai yang memiliki fungsi berbeda, terdiri dari: *Support*, berfungsi sebagai nilai propabilitas kemunculan *item* dan *item set* dalam data transaksi [14]. *Confidence*, berfungsi sebagai nilai persentase propabilitas *item* dibeli secara bersamaan dengan *item* lainnya [15]. *Lift ratio*, berfungsi sebagai nilai kevalidasian pemrosesan data [16]. Tiga nilai *association rule* diperoleh dengan 5 rumus berbeda, antara lain:

$$Support (A) = \frac{\sum Transaksi Mengandung A}{\sum Transaksi} \times 100\% \quad (1)$$

Nilai *support* satu *item* diperoleh menggunakan rumus 1 dengan *support* (A) adalah *support* dari *item* A, A adalah transaksi yang memiliki *item* A. Nilai *support* dari satu *item* sudah diperoleh, maka selanjutnya menghitung *support* dari *item set* untuk mendapatkan *frequent item set* [17].

$$Support P(A|B) = \frac{\sum Transaksi Mengandung A dan B}{\sum Transaksi} \times 100 \quad (2)$$

Nilai *support* dari *itemset* diperoleh menggunakan rumus 2 dengan *support* P adalah *support* dari *item* A dan B, A adalah transaksi yang memiliki *item* A, B adalah *item* yang memiliki *item* B. Nilai *support* dari *item set* sudah diperoleh, maka selanjutnya membentuk *rule* dan menghitung nilai *confidence* dari masing-masing *item set* [17].

$$Confidence = P(B|A) = \frac{\sum Transaksi Mengandung A dan B}{\sum Transaksi Mengandung A} \quad (3)$$

Nilai *confidence* diperoleh menggunakan rumus 3 dengan P adalah peluang *item* A dibeli secara bersamaan dengan *item* B, A adalah transaksi yang memiliki *item* A, B adalah *item* yang memiliki *item* B. Nilai *confidence* sudah diperoleh, maka selanjutnya membutuhkan nilai *expected confidence* untuk perhitungan *lift ratio* [17].

$$Expected Confidence = \frac{\sum Transaksi Mengandung B}{\sum Transaksi} \quad (4)$$

Nilai *expected confidence* diperoleh menggunakan rumus 4 dengan membagi jumlah nilai transaksi mengandung B dengan jumlah seluruh transaksi. Nilai *expected confidence* dibutuhkan dalam perhitungan *lift ratio* setelah nilai *confidence*. Nilai *expected confidence* sudah diperoleh, maka dibutuhkan nilai *lift ratio* mengetahui masing-masing *rule* bersifat valid atau tidak [17].

$$Lift Ratio = \frac{Confidence}{Expected Confidence} \quad (5)$$

Nilai *lift ratio* diperoleh menggunakan rumus 5 dengan membagi nilai *confidence* dan *expected confidence*. Nilai minimum *lift ratio* yaitu 1 sehingga jika *rule* memiliki nilai *lift ratio* sama atau lebih dari 1 maka *rule* bersifat valid dan masuk dalam list frequent pattern set tetapi jika *rule* memiliki nilai *lift ratio* dibawah 1 maka *rule* bersifat tidak valid [17].

2.4 Rule dan Pattern

Pemrosesan data dengan metode algoritma *pincer search* dan algoritma *FP-Growth* menggunakan bahasa pemrograman Python. Python menjadi salah satu bahasa pemrograman yang dapat membantu manusia dalam menyelesaikan permasalahan di beberapa kebutuhan seperti data mining asosiasi hingga menghasilkan *frequent item set* [18]. *Frequent item set* yang dihasilkan dari kedua metode kemudian membentuk *rule* setelah *rule* dari masing-masing *frequent item set* terbentuk lalu menghitung nilai *lift ratio* dari *rule*, jika *rule* memiliki nilai *lift ratio* sama atau lebih dari 1 maka *rule* dinyatakan valid dan masuk kedalam daftar *frequent pattern set* [19].

2.5 Ilmu Pengetahuan dan Informasi

Hasil pemrosesan data dari algoritma *pincer search* dan algoritma *FP-Growth* kemudian dianalisis dan dikomparasi untuk mengetahui hasil komparasi pengaruh perbedaan parameter terhadap waktu pemrosesan data, *frequent item set*, *rule*, *confidence* dan *lift ratio* pada Toko X. Hasil pemrosesan data dari algoritma *pincer search* dan algoritma *FP-Growth* dapat memberikan ilmu pengetahuan serta informasi yang dibutuhkan oleh Toko X [20,21].

3. HASIL DAN PEMBAHASAN

Hasil komparasi algoritma *pincer search* dan algoritma *FP-Growth* diperoleh dengan beberapa pengujian. Berikut beberapa pengujian yang dilakukan dalam penelitian mengenai komparasi algoritma *pincer search* dan algoritma *FP-Growth*.

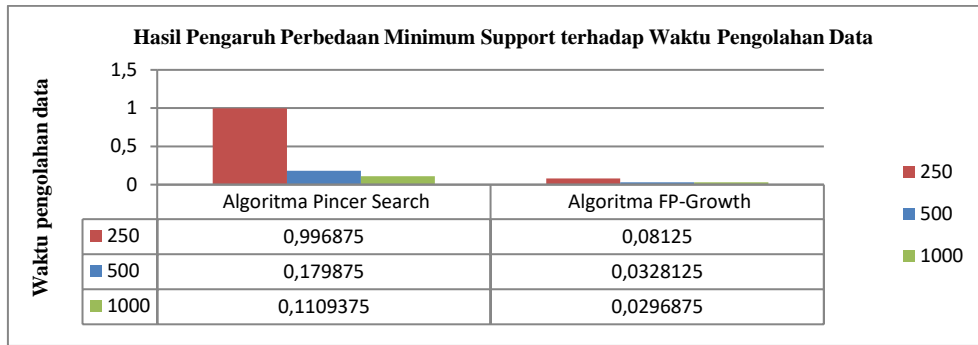
3.1 Pengaruh perbedaan parameter terhadap Waktu Pemrosesan Data

Pengujian pengaruh perbedaan parameter terhadap waktu pemrosesan data dilakukan untuk mengetahui perbedaan waktu pemrosesan data dari algoritma *pincer search* dan algoritma *FP-Growth*.

Tabel 1 Hasil Pengaruh Perbedaan Minimum Support Waktu Pemrosesan Data

Parameter			<i>Pincer Search</i>	<i>FP-Growth</i>
Transaksi	Item	Minimum Support		
10.535	1.242	250	0.996875	0.08125
		500	0.179875	0.0328125
		1000	0.1109375	0.0296875

Tabel 1 menunjukkan bahwa waktu pemrosesan data algoritma *FP-Growth* lebih cepat dari algoritma *pincer search* dengan 3 minimum *support* berbeda dan data sampel dengan jumlah data transaksi lebih dominan dari data *item* karena jumlah *item* dan *item set* yang diproses semakin sedikit yang disebabkan semakin berkurangnya jumlah *item* dan *item set* yang memenuhi minimum *support* maka waktu pemrosesan data dari kedua metode menjadi lebih cepat. Perbedaan waktu pemrosesan data dari kedua metode pada pengujian perbedaan minimum *support* terhadap waktu pemrosesan data dijelaskan dalam grafik.



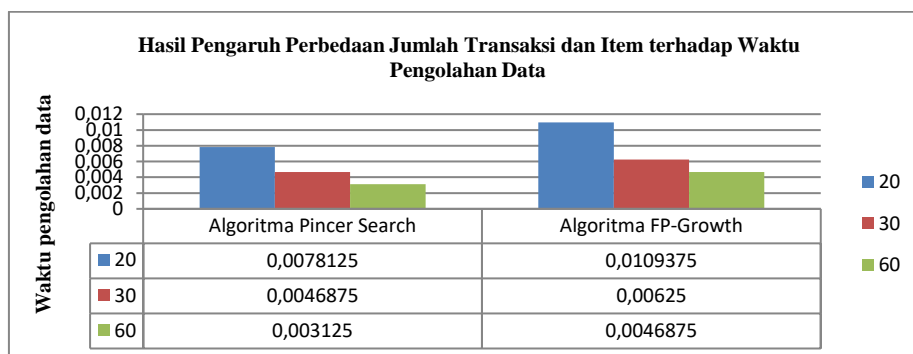
Gambar 2 Grafik Hasil Pengaruh Perbedaan Minimum Support Waktu Pemrosesan Data

Gambar 2 menunjukkan bahwa algoritma *FP-Growth* lebih cepat dari algoritma *pincer search* dalam waktu pemrosesan data karena algoritma *FP-Growth* memangkas *item* yang tidak memenuhi minimum *support* dan menyimpan *item* yang memenuhi minimum *support* pada tabel baru sehingga algoritma *FP-Growth* hanya memproses *item* pada tabel baru dalam membentuk *FP-Tree* maka membuat proses pemrosesan data algoritma *FP-Growth* menjadi lebih cepat dari algoritma *pincer search* karena algoritma *pincer search* tidak ada proses pemangkasan *item* seperti algoritma *FP-Growth*.

Tabel 2 Hasil Pengaruh Perbedaan Jumlah Transaksi dan Item Waktu Pemrosesan Data

Parameter			<i>Pincer Search</i>	<i>FP-Growth</i>
Transaksi	Item	Minimum Support		
300	1.023	20	0.0078125	0.0109375
		30	0.00625	0.0046875
		60	0.0046875	0.003125

Tabel 2 menunjukkan bahwa waktu pemrosesan data algoritma *pincer search* lebih cepat dari algoritma *FP-Growth* dengan 3 minimum *support* berbeda dan data sampel dengan jumlah data *item* lebih banyak dari jumlah data transaksi. Minimum *support* 60 memperoleh waktu pemrosesan data lebih cepat dari minimum *support* 20 dan 30. Perbedaan waktu pemrosesan data dari kedua metode pada pengujian perbedaan jumlah transaksi dan *item* terhadap waktu pemrosesan data dijelaskan dalam grafik.



Gambar 3 Grafik Hasil Pengaruh Perbedaan Jumlah Transaksi dan *Item* Waktu Pemrosesan Data

Gambar 3 menunjukkan bahwa waktu pemrosesan data dari algoritma *pincer search* lebih cepat dari algoritma *FP-Growth* karena jumlah *item* lebih banyak dibandingkan jumlah transaksi maka proses membentuk dan menghitung kombinasi *item* menjadi lebih cepat.

3.2 Pengaruh perbedaan parameter terhadap Frequent Item Set

Pengujian pengaruh perbedaan parameter terhadap *frequent item set* dilakukan untuk mengetahui perbedaan jumlah *frequent item set* yang dihasilkan dari algoritma *pincer search* dan algoritma *FP-Growth*.

Tabel 3 Hasil Pengaruh Perbedaan Minimum Support Waktu Pemrosesan Data

Parameter				Pincer Search	FP-Growth
Max Sup	Transaksi	Item	Minimum Support		
3366	10535	1242	250	5	5
			500	4	4
			1000	0	0

Tabel 3 menunjukkan bahwa perbedaan minimum *support* berpengaruh terhadap jumlah *frequent item set* yang dihasilkan kedua metode karena semakin besar minimum *support* yang ditentukan maka semakin sedikit *item* dan *item set* yang memenuhi minimum *support* dan masuk kedalam *list frequent item* dan *frequent item set*.

Tabel 4 Hasil Pengaruh Perbedaan Jumlah Transaksi dan Item Frequent Item Set

Parameter				Pincer Search	FP-Growth
Max Sup	Transaksi	Item	Minimum Support		
126	300	1023	20	5	5
			30	4	4
			60	0	0

Tabel 4 menunjukkan bahwa pengujian pengaruh perbedaan jumlah transaksi dan item pada data sampel dengan jumlah *item* lebih dominan dari jumlah transaksi memperoleh *frequent item set* dengan jumlah yang sama dengan pengujian pengaruh perbedaan minimum *support* dari algoritma *pincer search* dan algoritma *FP-Growth*.

3.3 Pengaruh perbedaan parameter terhadap Rule

Pengujian pengaruh perbedaan parameter terhadap *rule* dilakukan untuk mengetahui perbedaan jumlah *rule* yang dihasilkan dari algoritma *pincer search* dan algoritma *FP-Growth* dengan dua sampel data.

Tabel 5 Hasil Pengaruh Perbedaan Minimum Support Rule

Parameter				Pincer Search	FP-Growth
Max Sup	Transaksi	Item	Minimum Support		
3366	10535	1242	250	14	14
			500	12	12
			1000	0	0

Tabel 5 menunjukkan bahwa perbedaan minimum *support* berpengaruh terhadap jumlah *rule* yang dihasilkan karena minimum *support* yang ditentukan mempengaruhi jumlah *frequent item set* yang dihasilkan dan jumlah *frequent item set* mempengaruhi jumlah *rule* yang dihasilkan dari kedua metode maka semakin banyak *frequent item set* maka semakin banyak pula *rule* yang dihasilkan.

Tabel 6 Hasil Pengaruh Perbedaan Jumlah Transaksi dan *Item Rule*

Parameter				Pincer Search	FP-Growth
Max Sup	Transaksi	Item	Minimum Support		
126	300	1023	20	14	14
			30	12	12
			60	0	0

Tabel 6 menunjukkan bahwa pengujian pada data sampel dengan jumlah *item* yang lebih dominan dari jumlah transaksi dengan pengujian pada data sampel dengan jumlah transaksi yang lebih dominan dari jumlah *item* menghasilkan *rule* terbanyak yaitu sejumlah 14 *rule* pada minimum *support* 20.

3.4 Pengaruh perbedaan parameter terhadap Confidence

Pengujian pengaruh perbedaan parameter terhadap *confidence* dilakukan untuk mengetahui perbedaan jumlah *confidence* yang dihasilkan dari algoritma *pincer search* dan algoritma FP-Growth.

Tabel 7 Hasil Pengaruh Perbedaan Minimum Support Confidence

Nilai Conf	Algoritma Pincer Search				Algoritma FP-Growth			
	627	719	954	997	627	719	954	997
10%	6	2	2	2	6	2	2	2
20%	5	2	2	2	5	2	2	2
30%	5	2	1	2	5	2	1	2
40%	4	1	1	1	4	1	1	1
50%	3	0	1	1	3	0	1	1
60%	3	0	1	0	3	0	1	0
70%	1	0	0	0	1	0	0	0
80%	1	0	0	0	1	0	0	0
90%	0	0	0	0	0	0	0	0

Tabel 7 menunjukkan bahwa nilai *support* 627 menjadi nilai *support* yang memperoleh jumlah *confidence* paling banyak dari nilai *support* lainnya karena *item set* dengan nilai *support* 627 membentuk *rule* paling banyak dari *item set* lainnya.

Tabel 8 Hasil Pengaruh erbedaan Jumlah Transaksi dan *Item Confidence*

Nilai Conf	Algoritma Pincer Search				Algoritma FP-Growth			
	32	33	41	44	32	33	41	44
20%	6	2	2	2	6	2	2	2
30%	5	2	2	2	5	2	2	2
40%	5	2	1	1	5	2	1	1
50%	4	1	1	1	4	1	1	1
60%	3	0	1	1	3	0	1	1
70%	3	0	1	0	3	0	1	0
80%	1	0	0	0	1	0	0	0
90%	1	0	0	0	1	0	0	0
100%	0	0	0	0	0	0	0	0

Tabel 8 menunjukkan bahwa jumlah *confidence* yang dihasilkan kedua metode dari pengujian pengaruh perbedaan minimum *support* sama dengan jumlah *confidence* dari pengujian pengaruh jumlah transaksi dan *item* yang dilakukan dengan data sampel yang berbeda tetapi nilai *confidence* yang dihasilkan berbeda.

3.5 Pengaruh perbedaan parameter terhadap Lift Ratio

Pengujian pengaruh perbedaan parameter terhadap lift ratio dilakukan untuk mengetahui perbedaan jumlah nilai lift ratio yang dihasilkan dari algoritma pincer search dan algoritma FP-Growth.

Tabel 9 Hasil Pengaruh Perbedaan Minimum Support Lift Ratio

Nilai Conf	Algoritma Pincer Search				Algoritma FP-Growth			
	627	719	954	997	627	719	954	997
10%	52	12	10	10	52	12	10	10
20%	49	12	10	10	49	12	10	10
30%	49	12	7	10	49	12	7	10
40%	43	7	7	6	43	7	7	6
50%	36	0	7	6	36	0	7	6
60%	36	0	7	0	36	0	7	0
70%	15	0	0	0	15	0	0	0
80%	15	0	0	0	15	0	0	0
90%	0	0	0	0	0	0	0	0

Tabel 9 menunjukkan bahwa jumlah nilai lift ratio yang dihasilkan algoritma pincer search sama dengan jumlah nilai lift ratio yang dihasilkan algoritma FP-Growth dengan minimum support 500.

Tabel 10 Hasil Pengaruh Perbedaan Jumlah Transaksi dan Item Lift Ratio

Nilai Conf	Algoritma Pincer Search				Algoritma FP-Growth			
	32	33	44	45	32	33	41	44
20%	35	10	8	7	35	10	8	7
30%	32	10	8	7	32	10	8	7
40%	32	10	5	4	32	10	5	4
50%	28	5	5	4	28	5	5	4
60%	23	0	5	4	23	0	5	4
70%	23	0	5	0	23	0	5	0
80%	9	0	0	0	9	0	0	0
90%	9	0	0	0	9	0	0	0
100%	0	0	0	0	0	0	0	0

Tabel 10 menunjukkan bahwa pengujian pengaruh jumlah transaksi dan item menghasilkan jumlah nilai lift ratio paling banyak dari nilai support 32 pada algoritma pincer search dan algoritma FP-Growth.

3.6 Pengaruh perbedaan parameter terhadap Hasil Rule dan Hasil Lift Ratio

Pengujian pengaruh perbedaan parameter terhadap output hasil dilakukan untuk mengetahui perbedaan output hasil dari algoritma pincer search dan algoritma FP-Growth menggunakan bahasa pemrograman Python.

Tabel 11 Hasil Pengaruh Perbedaan Minimum Support Hasil Rule dan Hasil Lift Ratio

Hasil Lift Ratio dan Rule			
Algoritma Pincer Search		Algoritma FP-Growth	
Rule	Lift Ratio	Rule	Lift Ratio
(PINSET',)=>('CAIRAN SOFTLENS 60ML',)	6	(PINSET',)=>('CAIRAN SOFTLENS 60ML',)	6
('CAIRAN SOFTLENS 60ML',)=>('PINSET',)	3	('CAIRAN SOFTLENS 60ML',)=>('PINSET',)	3
('TETES SOFTLENS',)=>('CAIRAN SOFTLENS 60ML',)	7	('TETES SOFTLENS',)=>('CAIRAN SOFTLENS 60ML',)	7
('CAIRAN SOFTLENS 60ML',)=> ('TETES SOFTLENS',)	3	('CAIRAN SOFTLENS 60ML',)=> ('TETES SOFTLENS',)	3
('TETES SOFTLENS',)=>('PINSET',)	7	('TETES SOFTLENS',)=>('PINSET',)	7

('PINSET',)=>('TETES SOFTLENS',)	6	('PINSET',)=>('TETES SOFTLENS',)	6
('TETES SOFTLENS',)=>('CAIRAN SOFTLENS 60ML', 'PINSET')	7	('TETES SOFTLENS',)=>('CAIRAN SOFTLENS 60ML', 'PINSET')	7
('PINSET',)=>('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS')	6	('PINSET',)=>('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS')	6
('PINSET', 'TETES SOFTLENS')=>('CAIRAN SOFTLENS 60ML',)	15	('PINSET', 'TETES SOFTLENS')=>('CAIRAN SOFTLENS 60ML',)	15
('CAIRAN SOFTLENS 60ML',)=>('PINSET', 'TETES SOFTLENS')	3	('CAIRAN SOFTLENS 60ML',)=>('PINSET', 'TETES SOFTLENS')	3
('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS')=>('PINSET')	11	('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS')=>('PINSET')	11
('CAIRAN SOFTLENS 60ML', 'PINSET')=>('TETES SOFTLENS',)	11	('CAIRAN SOFTLENS 60ML', 'PINSET')=>('TETES SOFTLENS',)	11

Tabel 11 menunjukkan bahwa *rule* yang terbentuk dari algoritma *pincer search* dan algoritma *FP-Growth* masuk dalam *list frequent pattern set* karena seluruh *rule* memiliki nilai *lift ratio* diatas nilai 1 maka *rule* yang dihasilkan valid namun jika nilai *lift ratio* dibawah dari nilai 1, maka *rule* yang dihasilkan tidak valid.

Tabel 12 Hasil Pengaruh Perbedaan Jumlah Transaksi dan Item Hasil *Rule* dan Hasil *Lift Ratio*

Hasil <i>Lift Ratio</i> dan <i>Rule</i>			
Algoritma <i>Pincer Search</i>		Algoritma <i>FP-Growth</i>	
<i>Rule</i>	<i>Lift Ratio</i>	<i>Rule</i>	<i>Lift Ratio</i>
('PINSET',)=>('CAIRAN SOFTLENS 60ML',)	4	('PINSET',)=>('CAIRAN SOFTLENS 60ML',)	4
('CAIRAN SOFTLENS 60ML',)=>('PINSET',)	2	('CAIRAN SOFTLENS 60ML',)=>('PINSET',)	2
('TETES SOFTLENS',)=>('CAIRAN SOFTLENS 60ML',)	5	('TETES SOFTLENS',)=>('CAIRAN SOFTLENS 60ML',)	5
('CAIRAN SOFTLENS 60ML',)=>('TETES SOFTLENS')	2	('CAIRAN SOFTLENS 60ML',)=>('TETES SOFTLENS')	2
('TETES SOFTLENS',)=>('PINSET',)	5	('TETES SOFTLENS',)=>('PINSET',)	5
('PINSET',)=>('TETES SOFTLENS',)	4	('PINSET',)=>('TETES SOFTLENS',)	4
('TETES SOFTLENS',)=>('CAIRAN SOFTLENS 60ML', 'PINSET')	5	('TETES SOFTLENS',)=>('CAIRAN SOFTLENS 60ML', 'PINSET')	5
('PINSET',)=>('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS')	4	('PINSET',)=>('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS')	4
('PINSET', 'TETES SOFTLENS')=>('CAIRAN SOFTLENS 60ML',)	9	('PINSET', 'TETES SOFTLENS')=>('CAIRAN SOFTLENS 60ML',)	9
('CAIRAN SOFTLENS 60ML',)=>('PINSET', 'TETES SOFTLENS')	2	('CAIRAN SOFTLENS 60ML',)=>('PINSET', 'TETES SOFTLENS')	2
('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS')=>('PINSET')	7	('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS')=>('PINSET')	7
('CAIRAN SOFTLENS 60ML', 'PINSET')=>('TETES SOFTLENS')	7	('CAIRAN SOFTLENS 60ML', 'PINSET')=>('TETES SOFTLENS')	7

Tabel 12 menunjukkan bahwa perbedaan jumlah transaksi dan *item* pada data sampel yang digunakan dalam pengujian pengaruh perbedaan minimum *support* dan pengujian pengaruh jumlah transaksi dan *item* menghasilkan sebagian besar *lift ratio* dengan nilai yang berbeda tetapi *rule* yang sama.

3.7 Program Asosiasi Algoritma *Pincer Search* dan Algoritma *FP-Growth*

Program algoritma asosiasi berdasarkan algoritma *pincer search* dan algoritma *FP-Growth* menggunakan bahasa pemrograman Python terbagi menjadi 3 bagian tahapan. Berikut 3 tahapan didalam program asosiasi algoritma *pincer search* dan algoritma *FP-Growth*.

```
C:\WINDOWS\py.exe  
loading data...  
Selamat datang di Program Algoritma Asosiasi  
  
#####  
help      : Menampilkan daftar perintah  
quit     : Mengakhiri program  
fp       : Memulai program FP Growth  
pincer   : Memulai program Pincer  
  
#####  
  
-----  
fp  
-----
```

Gambar 4 Tahapan Pemilihan Metode

Gambar 4 menunjukkan bahwa tahapan pemilihan metode algoritma yang dapat membantu pengguna dalam memilih metode yang tersedia didalam program, yaitu metode algoritma *pincer search* dan metode algoritma *FP-Growth*. Tahapan lain juga terdapat Help berfungsi yang untuk menampilkan daftar perintah sebelumnya dan Quit yang berfungsi untuk mengakhir program.

```
C:\WINDOWS\py.exe  
mulai  
-----  
#####  
Perhatian!!!  
- Pastikan data yang ingin diolah berada dalam 1 folder dengan file ini  
- Pastikan data dalam format csv  
- Pastikan baris 1 pada data tidak terdapat tanda "," karena dapat menyebabkan masalah  
- Pastikan tidak terdapat data kosong didalam data. Kosongnya data memicu error  
#####  
  
Lanjutkan?  
(Y/N)  
  
-----  
Y  
-----  
#####  
Jumlah Transaksi : 10535  
Jumlah Item      : 1242  
Support tertinggi: 3366  
#####  
  
Masukan nilai support!  
Nilai 3366>x>0  
  
-----  
250  
-----  
#####  
Nilai support yang dimasukkan adalah 250  
  
help      : Menampilkan Daftar Perintah  
lanjut    : Lanjutkan ke proses FP growth
```

Gambar 5 Tahapan Memulai Program *Data Mining*

Gambar 5 menunjukkan bahwa tahapan memulai program data mining. Program data mining dapat dimulai setelah file dataset diproses oleh program dan pengguna menentukan minimum *support* kemudian minimum *support* tersebut diproses oleh program. Program memulai proses data mining saat pengguna meng-*input*-kan kata "Lanjut" ke dalam program.

```
C:\Users\Gek\AppData\Local\Programs\Python\Python37-32\python.exe  
760  
-----  
#####  
Nilai support yang dimasukkan adalah 250  
  
help      : Menampilkan Daftar Perintah  
lanjut    : Lanjutkan ke proses Pincer  
#####  
  
-----  
lanjut  
-----  
Running Time: 0.125  
#####  


| No | SUP  | FREQUENT ITEM dan FREQUENT ITEM SET        | FREQUENT PATTERN SET dan Confidence                 | LIFT Ratio |
|----|------|--------------------------------------------|-----------------------------------------------------|------------|
| 1  | 3366 | (CAIRAN SOFTLENS 60ML , )                  |                                                     |            |
| 2  | 1885 | (PINSET , )                                |                                                     |            |
| 3  | 1529 | (TETES SOFTLENS , )                        |                                                     |            |
| 4  | 997  | (CAIRAN SOFTLENS 60ML , PINSET , )         | (PINSET , )->(CAIRAN SOFTLENS 60ML , )=0.53         | 5.59       |
| 5  | 997  | (CAIRAN SOFTLENS 60ML , PINSET , )         | (CAIRAN SOFTLENS 60ML , )->(PINSET , )=0.39         | 3.13       |
| 6  | 954  | (CAIRAN SOFTLENS 60ML , TETES SOFTLENS , ) | (TETES SOFTLENS , )->(CAIRAN SOFTLENS 60ML , )=0.62 | 6.89       |
| 7  | 954  | (CAIRAN SOFTLENS 60ML , TETES SOFTLENS , ) | (CAIRAN SOFTLENS 60ML , )->(TETES SOFTLENS , )=0.28 | 3.13       |
| 8  | 719  | (PINSET , TETES SOFTLENS , )               | (TETES SOFTLENS , )->(PINSET , )=0.47               | 6.89       |
| 9  | 719  | (PINSET , TETES SOFTLENS , )               | (PINSET , )->(TETES SOFTLENS , )=0.38               | 5.59       |

  
#####  
Program selesai...  
Tekan Enter Untuk Keluar
```

Gambar 6 Output Hasil Algoritma *Pincer Search*

Gambar 6 menunjukkan *output* hasil dari algoritma *pincer search* berupa *item* dan *item set* yang memenuhi minimum *support* yang masuk dalam *frequent item* dan *frequent item set* serta *rule* yang bernilai diatas 1 bersifat valid dan masuk ke dalam *list frequent pattern set*..

```

C:\Users\Gek\AppData\Local\Programs\Python\Python37-32\python.exe
700 --
=====
Nilai support yang dimasukkan adalah 700
help : Masukkan Daftar Perintah
lanjut : Lanjutkan ke proses FP-Growth
=====
lanjut
=====
Running Time: 0.03125
=====
| No | Support | Frequent Item dan Frequent Item Set | Frequent Pattern Set dan Confidence | Lift Ratio |
|---|---|---|---|---|
| 1 | 366 | ('CAIRAN SOFTLENS 60ML',) | ('CAIRAN SOFTLENS 60ML',) | 5.59 |
| 2 | 1885 | ('PINSET',) | ('PINSET',) | 3.13 |
| 3 | 1529 | ('TETES SOFTLENS',) | ('TETES SOFTLENS',) | 6.89 |
| 4 | 997 | ('CAIRAN SOFTLENS 60ML', 'PINSET') | ('PINSET',)->('CAIRAN SOFTLENS 60ML',)-0.53 | 5.59 |
| 5 | 997 | ('CAIRAN SOFTLENS 60ML', 'PINSET') | ('CAIRAN SOFTLENS 60ML',)->('PINSET',)-0.30 | 3.13 |
| 6 | 954 | ('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS') | ('TETES SOFTLENS',)->('CAIRAN SOFTLENS 60ML',)-0.62 | 6.89 |
| 7 | 954 | ('CAIRAN SOFTLENS 60ML', 'TETES SOFTLENS') | ('CAIRAN SOFTLENS 60ML',)->('TETES SOFTLENS',)-0.28 | 3.13 |
| 8 | 719 | ('PINSET', 'TETES SOFTLENS') | ('TETES SOFTLENS',)->('PINSET',)-0.42 | 6.89 |
| 9 | 719 | ('PINSET', 'TETES SOFTLENS') | ('PINSET',)->('TETES SOFTLENS',)-0.38 | 5.59 |
=====
Program selesai...
Tekan Enter Untuk Keluar
    
```

Gambar 7 Output Hasil Algoritma FP-Growth

Gambar 7 menunjukkan output hasil algoritma FP-Growth berupa *frequent item*, *frequent item set* dan *rule* dengan nilai *support*, nilai *confidence* dan nilai *lift ratio* menunjukkan output hasil yang sama dengan output hasil algoritma *pincer search* tetapi *output* hasil kedua metode memiliki perbedaan waktu pemrosesan data karena alur proses data, jumlah transaksi dan jumlah *item* dari data sampel yang diproses berbeda.

4. KESIMPULAN DAN SARAN

Nilai hasil yang diperoleh dari algoritma *pincer search* dan algoritma FP-Growth dapat digunakan sebagai bahan pertimbangan untuk mengantisipasi permasalahan pada stok barang yang dialami oleh Toko X dan komparasi algoritma *pincer search* dan algoritma FP-Growth memperoleh beberapa kondisi, diantaranya: jika data transaksi lebih dominan dari data *item*, maka waktu pemrosesan data dari algoritma FP-Growth lebih cepat dari algoritma *pincer search* dan nilai *support maximum* yang dapat diproses oleh program semakin banyak serta *lift ratio* yang dihasilkan juga semakin banyak tetapi jika data *item* lebih dominan dari data transaksi, maka waktu pemrosesan data dari algoritma *pincer search* lebih cepat dari algoritma FP-Growth dan nilai *support maximum* yang dapat diproses oleh program semakin sedikit serta *lift ratio* yang dihasilkan juga semakin sedikit, sedangkan semakin besar minimum *support* yang ditentukan, maka *frequent item set* dan *rule* yang dihasilkan semakin sedikit sehingga nilai *confidence* yang dihasilkan juga semakin sedikit namun waktu pemrosesan data dari algoritma *pincer search* dan algoritma FP-Growth semakin cepat.

Penelitian mengenai komparasi metode *association rule*, selanjutnya dapat dikembangkan atau dikombinasikan dengan metode lainnya dan melakukan penerapan metode dengan data yang lain seperti data peminjaman buku dipertustakaan dan lainnya.

DAFTAR PUSTAKA

- [1] C. Mashuri *et al.*, “Sistem Inventory Manajemen Dengan Metode Safety Stock,” in *Sainteknopak* 5, 2021, hal. 1–9.
- [2] D. P. S. P. Putu Githa Pratiwi, I Ketut Gede Darma Putra, “Peramalan Jumlah Tersangka Penyalahgunaan Narkoba Menggunakan Metode Multilayer Perceptron,” *Menara Penelit. Akad. Teknol. Inf.*, vol. 7, no. 2, hal. 0–7, 2019.
- [3] T. M. Satrio Junaidi, “Metode Data Mining Association Rule Dengan Algoritma FP-Growth Untuk Mengetahui Kelulusan Tepat Waktu Mahasiswa (Studi Kasus STIKIP PGRI Sumatera Barat),” *J. Edik Inform.*, vol. 5, no. 1, hal. 41–53, 2018.

- [4] I. M. S. P. Ni Putu Ratindia Apriyanti, I Ketut Gede Darma Putra, “Peramalan Jumlah Kecelakaan Lalu Lintas Menggunakan Metode Support Vector Regression,” *Menara Penelit. Akad. Teknol. Inf.*, vol. 8, no. 2, hal. 72–80, 2020.
- [5] A. B. P. Rusnandi, Suparni, “Penerapan Data Mining Untuk Analisis Market Basket Dengan Algoritme FP-Growth Pada PD Pasar Tohaga,” *J. Nas. Pendidik. Tek. Inform.*, vol. 9, no. 1, hal. 119–133, 2020.
- [6] A. R. Wijaya dan A. Jananto, “Mencari Pola Pembelian Konsumen Menggunakan Algoritma FP-Growth,” *Din. Inform.*, vol. 10, no. 1, hal. 1–7, 2018.
- [7] M. N. Biman Giri, Sivagami M, “Frequent Item Set Mining for Data Streaming using Spark with Pincer Search Algorithm,” *Int. J. Recent Technol. Eng.*, vol. 7, no. 6, hal. 40–45, 2019.
- [8] A. Anggrawan dan C. Satria, “Menentukan Akurasi Tata Letak Barang dengan Menggunakan Algoritma Apriori dan Algoritma FP-Growth,” *J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 1, hal. 125–138, 2021.
- [9] I. F. Rozi, H. Pradibta, S. Windy, dan K. Putri, “Penerapan Market Basket Analysis Pada Data Penjualan Menggunakan Algoritma Pincer Search,” in *Seminar Informatika Aplikatif Polinema*, 2020, hal. 30–34.
- [10] A. K. Siregar *et al.*, “Perbandingan Algoritme FP-Growth dan Eclat untuk Analisis Pola Pembelian Konsumen pada Toko ‘ X ,’” in *Conference on Information Technology, Information System and Electrical Engineering*, 2018, hal. 125–128.
- [11] S. N. Krisna Nata Wijaya, Reza Firsandaya Malik, “Analisa Pola Frekuensi Keranjang Belanja Dengan Perbandingan Algoritma Fp-Growth (Frequent Pattern Growth) Dan Eclat Pada Minimarket,” *J. Tek. Inform. dan Sist. Inf.*, vol. 7, no. 2, hal. 364–373, 2020.
- [12] F. Teknik dan U. M. Semarang, “Deteksi Penyakit Algoritma ID3 Gagal Ginjal Kronis Menggunakan,” *J. Iimiah Elektron. Dan Komput.*, vol. 13, no. 1, hal. 8–17, 2020.
- [13] N. A. Novalia Barkah, Entin Sutinah, “Metode Asosiasi Data Mining Untuk Analisa Persediaan Fiber Optik Menggunakan Algoritma Apriori,” *J. Kaji. Ilm.*, vol. 20, no. 3, hal. 237–248, 2020.
- [14] S. S. Fironika Mayuni, “Perbandingan Algoritma Apriori dan Algoritma Eclat Dalam Penentuan Pola Peminjaman Buku Pada Perpustakaan Universitas Bina Darma Palembang,” in *Bina Darma Conference on Computer Science*, 2018, hal. 87–100.
- [15] H. Maulidiya dan A. Jananto, “Asosiasi Data Mining Menggunakan Algoritma Apriori dan FP-Growth Sebagai Dasar Pertimbangan Penentuan Paket Sumbako,” in *Seminar Nasional Multi Disiplin Ilmu*, 2020, hal. 978–979.
- [16] J. R. Informatika, “Analisis Pola Penjualan Dengan Menggunakan Algoritma Apriori Pada Koperasi Karyawan Yayasan Anakku,” vol. 2, no. 1, hal. 2–7, 2019.
- [17] A. N. Rahmi, Y. A. Mikola, dan K. Kunci, “Implementasi Algoritma Apriori Untuk Menentukan Pola Pembelian Pada Customer (Studi Kasus : Toko Bakoel Sembako),” vol. 4, no. 1, hal. 14–18, 2021.
- [18] N. Nyoman, P. Pinata, I. M. Sukarsa, N. Kadek, dan D. Rusjyanthi, “Prediksi Kecelakaan Lalu Lintas di Bali dengan XGBoost pada Python,” vol. 8, no. 3, hal. 188–196, 2020.
- [19] S. Laili Zami Saputri, “Penerapan Data Mining Pembuatan Produk Baru Tinta Menggunakan Metode Algoritma Apriori,” *J. Manaj. Inform. Jayakarta*, vol. 1, no. 9, hal. 320–335, 2021.
- [20] L. A. W. S, “Penerapan Data Mining Pada Penjualan Tiket Pesawat Menggunakan Algoritma Apriori (Studi Kasus : PT . Sinar Jagad Indonesia),” *J. Inform. dan Sist. Inf.*, vol. 06, no. 01, hal. 11–22, 2020.
- [21] J. Nasional, I. Komputer, E. T. Naldy, F. Teknik, I. Komputer, dan U. B. Darma, “Penerapan Data Mining Untuk Analisis Daftar Pembelian Konsumen Dengan Menggunakan Algoritma Apriori Pada Transaksi Penjualan Toko Bangunan MDN,” *J. Nas. Ilmu Komput.*, vol. 2, no. 2, hal. 89–101, 2021.