

Implementasi Algoritma *FP-Growth* untuk Menemukan Pola Keterkaitan Antara Matakuliah Pemrograman dan Matakuliah Matematika

Zurneli Kurnia Putri. P^{1*}, Iwan Iskandar¹, Alwis Nazir¹

¹Program Studi Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim, Indonesia
nelipanee@gmail.com (*corresponding author), iwan.iskandar@uin-suska.ac.id, alwis.nazir@uin-suska.ac.id

Abstrak. Spesifikasi keahlian pemrograman merupakan salah satu fokus pembelajaran di program studi Teknik Informatika yang menuntut mahasiswa untuk memahami dan mendapatkan nilai yang baik pada semua mata kuliah yang berkaitan dengan pemrograman. Mata kuliah yang dianggap memiliki keterkaitan dengan bidang pemrograman adalah mata kuliah Matematika. Upaya yang dapat dilakukan untuk mengetahui keterkaitan antara mata kuliah pemrograman dan mata kuliah matematika salah satunya melalui algoritma asosiasi dalam data mining yaitu algoritma FP-Growth. FP-Growth dipilih karena memiliki kecepatan eksekusi pola data yang lebih cepat dibandingkan dengan algoritma apriori. Tahap akhir KDD menghasilkan 1227 data yang kemudian diolah menggunakan algoritma FP-Growth. Pengujian dengan nilai support minimal 0,5 dan confidence minimal 0,7 menunjukkan jumlah pola yang sama antara aplikasi yang dibangun dengan aplikasi SPMF sebanyak 52250 pola. Nilai support tertinggi sebesar 51% dan nilai confidence tertinggi sebesar 98% serta nilai lift ratio tertinggi sebesar 1,1941 pada kombinasi pola itemset menunjukkan bahwa jika mahasiswa lulus mata kuliah pemrograman, maka mata kuliah matematika juga dapat lulus atau sebaliknya.

Kata kunci: Data Mining, FP-Growth, KDD, Matematika, Pemrograman.

Abstract. The specification of programming skills is one of the focuses of learning in the Informatics Engineering study program which requires students to understand and get good grades in all courses related to programming. The subject that is considered to have a relationship with the programming field is the Mathematics course. Efforts to determine the correlation between programming courses and mathematics courses through one of the association algorithms in data mining, namely the FP-Growth algorithm. FP-Growth was chosen because it has a faster data pattern execution rate than the a priori algorithm. The final stage of KDD produces 1227 data which is then processed using the FP-Growth algorithm. Tests with a minimum support value of 0.5 and a minimum confidence of 0.7 show the same number of patterns between applications built with the SPMF application of 52250 patterns. The highest support value of 51% and the highest confidence value of 98% and the highest lift ratio value of 1.1941 in the combination of itemset patterns indicate that if students pass programming courses, then mathematics courses can also pass or vice versa.

Keywords: Data Mining, FP-Growth, KDD, Mathematics Course, Programming Course.

Received August 2021 / Revised October 2021 / Accepted December 2021

This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).



PENDAHULUAN

Spesifikasi keahlian pemrograman menjadi salah satu fokus pembelajaran di prodi Teknik informatika. Hal ini mengharuskan mahasiswa memahami semua mata kuliah yang berkaitan dengan pemrograman. Dibuktikan dengan perolehan nilai yang baik pada tiap mata kuliah pemrograman. Mahasiswa yang mendapat nilai rendah (D atau E) pada mata kuliah pemrograman akan mempengaruhi kualitas studi mahasiswa tersebut. Meskipun perolehan nilai mahasiswa yang rendah tidak secara langsung mempengaruhi kualitas prodi, namun jika hal ini terus berlanjut hingga mengakibatkan banyaknya mahasiswa TIF yang terlambat lulus maka efek selanjutnya akan mempengaruhi akreditasi prodi.

Mahasiswa yang mendapatkan nilai D atau E juga diharuskan mengulang perkuliahan tersebut. Hal ini berdasarkan penetapan peraturan akademik UIN SUSKA Riau tahun 2018 [1] pada pasal 41 tentang Pengulangan Mata kuliah ayat 1, yang berbunyi: "Mahasiswa yang mendapat nilai E wajib mengulang perkuliahan secara utuh mata kuliah bersangkutan" dan juga pada ayat 2, yang berbunyi: Mahasiswa yang mendapat nilai D pada mata kuliah keahlian program studi, wajib memperbaiki nilainya dengan mengikuti seluruh kegiatan perkuliahan, praktikum, dan tugas-tugas akademik lainnya yang berkaitan dengan mata kuliah bersangkutan secara utuh". Mahasiswa juga diperbolehkan mengikuti munaqasyah jika tidak

memiliki nilai D pada mata kuliah keahlian. Hal ini juga dijelaskan pada pasal 55 tentang prosedur munaqasyah ayat 2b, yang berisikan salahsatu syarat bagi mahasiswa yang akan mengikuti munaqasyah ialah: “telah melaksanakan semua tugas akademik, kelengkapan administrasi, IPK serendah-rendahnya 2,00 dan tidak ada nilai D pada mata kuliah keahlian.

Hal yang menjadi dasar dari penelitian ini adalah timbulnya dampak lain yang diakibatkan adanya nilai rendah pada matakuliah pemrograman. Selain itu, permasalahan lain yang memperkuat penelitian ini terkait nilai mata kuliah pemrograman, terdapat 40% mahasiswa TIF dengan rentang angkatan 2014-2017 mendapatkan perolehan nilai rendah pada mata kuliah pemrograman. Data ini diperoleh setelah dilakukan survei terhadap 100 mahasiswa TIF. Data yang diperoleh dari prodi menjelaskan bahwa mata kuliah bidang matematika juga dinilai memiliki korelasi dengan mata kuliah bidang pemrograman.

Algoritma FP-Growth dipilih pada penelitian ini karena algoritma FP-Growth merupakan algoritma asosiasi penyempurna algoritma apriori. Pada algoritma ini tidak dibutuhkan terlalu banyak scanning database seperti halnya pada algoritma apriori. *Scanning database* hanya dilakukan satu atau dua kali sehingga menyebabkan algoritma FP-Growth memiliki tingkat eksekusi pola data lebih cepat dibandingkan algoritma apriori [2]. Penelitian yang dilakukan untuk menemukan pola obat juga menunjukkan bahwa algoritma *Fp-Growth* mampu menemukan pola obat dari data diagnosa penyakit [3]. Penelitian tentang analisis pola pembelian konsumen dengan algoritma *Fp-Growth* pada data transaksi penjualan *spare part* motor juga menunjukkan bahwa algoritma *Fp-Growth* mampu menemukan jenis *spare part* yang banyak dicari konsumen dan membantu merekomendasikan jumlah permintaan *spare part* yang dibutuhkan untuk kemudian disediakan oleh kantor pusat [4]. Algoritma FP-Growth memanfaatkan konsep struktur data berupa pembangunan pohon (tree) atau disebut dengan FP-Tree untuk menemukan frequent itemset pada dataset. Tahapan-tahapan yang dilakukan pada algoritma ini [5] adalah Pembangkitan *conditional pattern base*, Pembangkitan *conditional FP- Tree*, dan Pencarian *frequent itemset*.

Korelasi yang terbentuk akan dilihat dari kekuatan nilai *support* dan *confidence*-nya untuk kemudian dapat menghasilkan informasi keterkaitan antara matakuliah pemrograman dan matakuliah matematika. Selanjutnya informasi tersebut akan digunakan oleh dosen pengampu mata kuliah pemrograman dan dosen pengampu mata kuliah matematika sebagai bahan evaluasi pembelajaran dan dapat berkoordinasi dalam melakukan perbaikan pola pembelajaran untuk meningkatkan pemahaman mahasiswa terkait matakuliah pemrograman dan matakuliah matematika.

METODE

A. Teknik Asosiasi

Teknik asosiasi merupakan teknik menemukan keterhubungan antar *itemset* dengan memberikan syarat nilai minimal *support* (minsup) dan minimal *confidence* (mincof) pada data yang akan diolah. Melalui penerapan teknik ini, akan ditemukan atribut yang saling keterkaitan dan juga atribut yang memiliki frekuensi kemunculan yang sering (*frequent itemset*) [6]. Nilai *support* merupakan nilai kombinasi antar *itemset*, sedangkan nilai *confidence* adalah nilai yang menunjukkan pembentukan asosiasi yang kuat antar *itemset*. Nilai *support* pada data didapatkan melalui perhitungan berikut :

$$Support(A) = \frac{\text{jumlah transaksi yang mengandung item A}}{\text{Total Transaksi}} \quad (1)$$

Sementara untuk menentukan nilai *Support* dengan 2 *itemset*, dapat dilakukan perhitungan sebagai berikut:

$$Support(A \cap B) = \frac{\text{jumlah transaksi yang mengandung } A \cap B}{\text{Total Transaksi}} \quad (2)$$

Sementara untuk mencari nilai *confidence* dapat dilakukan dengan perhitungan sebagai berikut:

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

B. *FP-Tree*

Data transaksi dipetakan menjadi sebuah lintasan berbentuk pohon (*FP-Tree*). Memungkinkan terjadinya lintasan yang saling menimpa jika terdapat transaksi yang memiliki item sama. Proses pemampatan akan semakin efektif jika semakin banyak item yang memiliki kesamaan antar transaksinya [7]. Untuk pelabelan awal pada akar pohon, akan diberikan label *null*. Tiap kumpulan *sub-tree* memiliki anggota beberapa *item* dan *frequent header* [8]. Untuk menemukan *frequent itemset*, *FP-Tree* akan digunakan bersamaan dengan *FP-Growth*. Berikut beberapa field yang terdapat pada *FP-tree*:

1. *Item-name*: menampilkan item yang terdapat pada simpul.
2. *Count*: menampilkan jumlah transaksi yang telah melewati simpul.
3. *Node-Link*: penghubung antar simpul dengan *item* yang sama. Berlabel *Null* jika kosong.

C. Algoritma *FP-Growth*

Tahapan pencarian pola dengan menggunakan algoritma *fp-growth* adalah sebagai berikut:

1. Penentuan frekuensi dan support masing-masing item
Masing-masing item akan dihitung frekuensi kemunculan dan juga *support* dengan menggunakan rumus 2.
2. Menentukan *minimum support*
Minimum support yang digunakan dalam penelitian ini adalah senilai $\geq 10\%$. Nilai ini akan menjadi penentu item-item yang akan di proses selanjutnya. Item yang tidak memenuhi dari *minimum support* ini akan dihapus. Pada data contoh ini item yang akan dihapus adalah item DPTL, KSTL, TWTL, ALTL, MDTL, SDTL, MNTL, dan PSTL.
3. Mengurutkan data transaksi berdasarkan frekuensi
Pengurutan data dilakukan berdasarkan frekuensi tertinggi sampai ke terendah.
4. Pembentukan *Frequent Pattern-Tree*
5. Penerapan Algoritma *Fp-Growth*
 - a. Tahap pembentukan *conditional pattern base*
 - b. Tahap pembentukan *conditional fp-tree*
 - c. Tahap pembentukan *frequent pattern tree*
6. Pembentukan Kombinasi Pola

HASIL DAN PEMBAHASAN

A. Analisa Proses KDD

Tahapan-tahapan KDD ini dilakukan dengan tujuan menghasilkan data yang akurat sesuai dengan tujuan penelitian ini. Berikut tahapan-tahapan analisa proses KDD [9]:

1. *Data Selection*

Keseluruhan atribut data mahasiswa yang telah diperoleh dari PTIPD UIN SUSKA RIAU dengan total atribut sebanyak 84 atribut, hanya akan digunakan beberapa atribut saja. Atribut yang dibutuhkan pada penelitian ini hanya atribut *id_pd* yang memuat nomor induk mahasiswa, atribut *id_smt* yang memuat keterangan matakuliah pada semester ganjil-genap, atribut *nm_mk* yang memuat nama matakuliah dan juga atribut *nilai_huruf* yang memuat perolehan nilai mahasiswa pada rentang nilai A sampai dengan E. Matakuliah pemrograman yang akan digunakan meliputi: matakuliah dasar pemrograman, algoritma dan pemrograman, teknologi web, pemrograman bergerak, dan struktur data. Sementara matakuliah matematika yang digunakan meliputi: matakuliah kalkulus, aljabar linear, matematika diskrit, probabilitas dan statistika dan metode numerik. Proses akhir dari tahap *selection* ini menghasilkan perubahan keseluruhan jumlah data yang awalnya sejumlah 41728 data dengan total atribut sebanyak 84 atribut menjadi 1667 data dengan total atribut sebanyak 11 atribut.

2. *Pre-processing*

Beberapa proses yang akan dilakukan pada tahapan ini menggunakan *Microsoft Excel* sebagai *tools* pendukung dengan beberapa tahapan yakni pengecekan data yang kosong atau *missing value*, data yang duplikat, adanya format data yang berbeda antara satu dengan yang lainnya atau *inconsistent data* dan pengecekan data diluar kewajaran atau *outlier*. Proses akhir dari tahap *pre-processing* ini menghasilkan perubahan keseluruhan jumlah data yang awalnya sejumlah 1667 data dengan total atribut sebanyak 11 atribut menjadi 1227 data dengan total atribut sebanyak 10 atribut.

3. Transformation

Proses transformasi data yang dilakukan adalah sebagai berikut:

- a. Nilai huruf dengan rentang nilai A-C akan ditransformasi dengan simbol L dan nilai huruf dengan rentang nilai C-D akan ditransformasi dengan simbol TL serta penambahan kode matakuliah untuk memudahkan pengenalan nilai asal matakuliah.
- b. Nilai matakuliah dasar pemrograman dengan rentang nilai A sampai dengan C diubah menjadi DPL (Dasar Pemrograman Lulus) sementara nilai D dan E diubah menjadi DPTL (Dasar Pemrograman Tidak Lulus).
- c. Nilai matakuliah kalkulus dengan rentang nilai A sampai dengan C diubah menjadi KSL (Kalkulus Lulus) sementara nilai D dan E diubah menjadi KSTL (Kalkulus Tidak Lulus).
- d. Nilai matakuliah matematika diskrit dengan rentang nilai A sampai dengan C diubah menjadi MDL (matematika Diskrit Lulus) sementara nilai D dan E diubah menjadi MDTL (matematika Diskrit Tidak Lulus).
- e. Nilai matakuliah Algoritma Pemrograman dengan rentang nilai A sampai dengan C diubah menjadi APL (Algoritma Pemrograman Lulus) sementara nilai D dan E diubah menjadi APTL (Algoritma Pemrograman Tidak Lulus).
- f. Nilai matakuliah Aljabar Linear dengan rentang nilai A sampai dengan C diubah menjadi ALL (Aljabar Linear Lulus) sementara nilai D dan E diubah menjadi ALTL (Aljabar Linear Tidak Lulus).
- g. Nilai matakuliah Struktur Data dengan rentang nilai A sampai dengan C diubah menjadi SDL (Struktur Data Lulus) sementara nilai D dan E diubah menjadi SDTL (Struktur Data Tidak Lulus).
- h. Nilai matakuliah Pemrograman Bergerak dengan rentang nilai A sampai dengan C diubah menjadi PBL (Pemrograman Bergerak Lulus) sementara nilai D dan E diubah menjadi PBTL (Pemrograman Bergerak Tidak Lulus).
- i. Nilai matakuliah Probabilitas dan Statistik dengan rentang nilai A sampai dengan C diubah menjadi PSL (Probabilitas dan Statistik Lulus) sementara nilai D dan E diubah menjadi PSTL (Probabilitas dan Statistik Tidak Lulus).
- j. Nilai matakuliah Teknologi Web dengan rentang nilai A sampai dengan C diubah menjadi TWL (Teknologi Web Lulus) sementara nilai D dan E diubah menjadi TWTL (Teknologi Web Tidak Lulus).
- k. Nilai matakuliah Metode Numerik dengan rentang nilai A sampai dengan C diubah menjadi MNL (Metode Numerik Lulus) sementara nilai D dan E diubah menjadi MNTL (Metode Numerik Tidak Lulus).

Tampilan transformasi dapat dilihat pada Gambar 1.

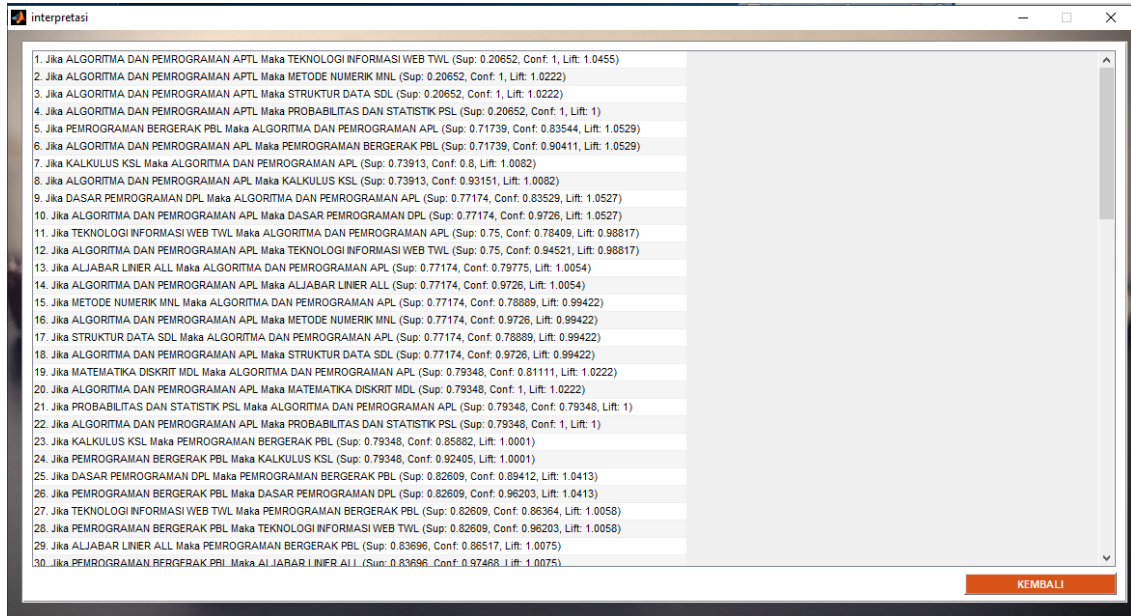
	A	B	C	D	E	F	G	H	I	
1	DASAR PEMROG	KALKULUS	MATEMATIKA DISKRIT	ALGORITMA DAN PEMROGR	ALJABAR LINIER	STRUKTUR DATA	TEKNOLOGI IN	PROBABILITAS DAN STA	PEMROGRAMAN BERGERAK	METOD
2	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
3	DPTL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
4	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
5	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
6	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
7	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
8	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
9	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
10	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
11	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
12	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
13	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
14	DPL	KSTL	MDL	APL	ALTL	SDL	TWL	PSL	PBL	MNL
15	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
16	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
17	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
18	DPL	KSL	MDTL	APL	ALL	SDL	TWL	PSL	PBL	MNL
19	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
20	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
21	DPTL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
22	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL
23	DPTL	KSL	MDL	APL	ALL	SDTL	TWL	PSL	PBL	MNTL
24	DPL	KSL	MDL	APL	ALL	SDL	TWL	PSL	PBL	MNL

Gambar 1. Transformation data

4. Interpretasi Pola

Berikut menunjukkan pola yang dihasilkan dari perhitungan sebelumnya dengan format bahasa “jika-maka”. Hal ini bertujuan agar memudahkan pengguna memahami pola asosiasi yang dihasilkan aplikasi. Gambar 4 menunjukkan hasil interpretasi pola dengan informasi yang bisa kita dapat pada informasi aturan pertama adalah: jika nilai mahasiswa pada matakuliah algoritma dan pemrograman lulus, maka nilai

mahasiswa pada matakuliah teknologi web lulus dengan tingkat kemunculan data (*support*) sebanyak 20% dengan tingkat kepercayaan aturan (*confidence*) senilai 100% serta nilai *lift ratio* sejumlah 1.04 yang menandakan bahwa aturan ini valid untuk dijadikan informasi [10]. Tampilan interpretasi aplikasi dapat dilihat pada Gambar 2.



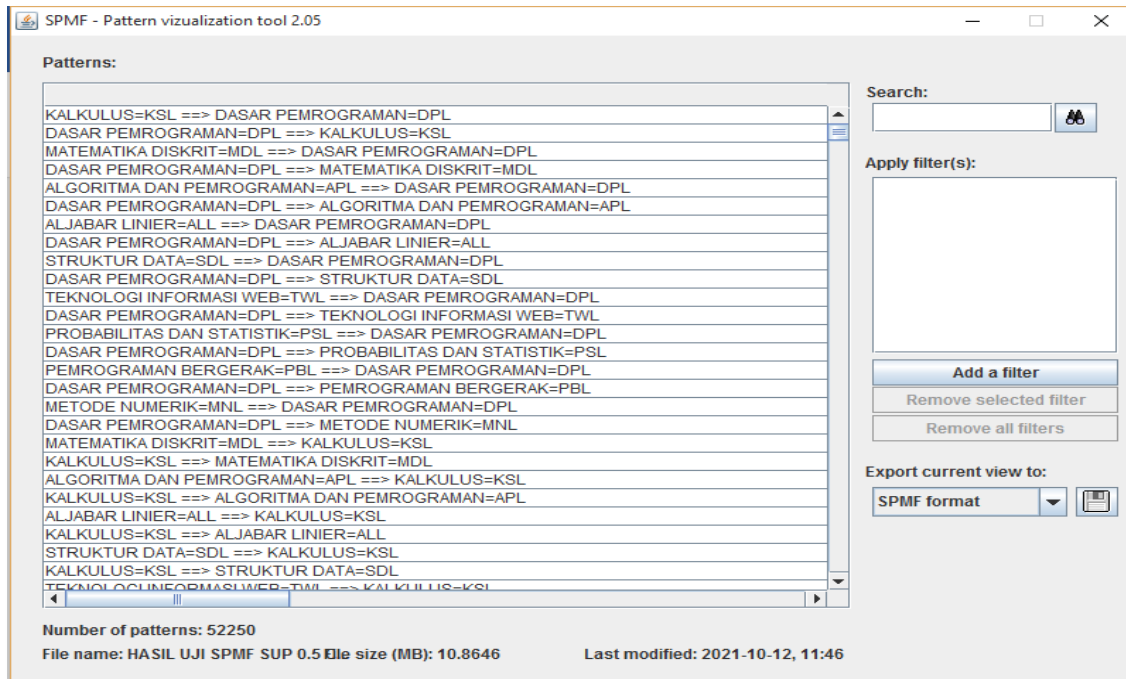
Gambar 2. Tampilan interpretasi aplikasi

5. Pengujian

Tahapan selanjutnya setelah tahap implementasi adalah tahapan pengujian. Tujuan dari tahapan pengujian ini adalah untuk memastikan bahwa aplikasi yang dibangun sudah sesuai dengan tujuan penelitian dan menunjukkan hasil sesuai dengan yang dibutuhkan. Tahap ini membutuhkan salahsatu *tools data* mining yaitu SPMF untuk membandingkan hasil yang dimunculkan pada aplikasi sesuai dengan hasil yang dimunculkan pada aplikasi *data mining* ini. Pengujian pada penelitian ini menggunakan beberapa kombinasi nilai *support* dan nilai *confidence* untuk menemukan model pola terbaik dan paling kuat untuk dijadikan informasi seperti yang terlihat pada Tabel 1, dan hasil implementasi pada aplikasi SPMF dapat dilihat pada Gambar 3.

Tabel 1 Pengujian jumlah pola

Kombinasi itemset	Nilai <i>Support</i>	Nilai <i>Confidence</i>	Jumlah pola pada aplikasi	Jumlah pola pada <i>tools</i> SPMF
5 itemset	0.7	0.9	245 pola	245 pola
6 Itemset	0.6	0.9	2118 pola	2118 pola
7 itemset	0.6	0.9	942 pola	942 pola
8 Itemset	0.5	0.9	1082 pola	1082 pola
9 Itemset	0.5	0.9	354 pola	354 pola
10 itemset	0.5	0.9	49 pola	49 pola



Gambar 3. Hasil Pola Pada SPMF

KESIMPULAN

Minimal *support* senilai 0.5 dan minimal *confidence* 0.7 menunjukkan jumlah pola yang sama antara aplikasi yang dibangun dengan aplikasi SPMF sejumlah 52250 pola. Pemilihan nilai minimal bertujuan untuk menampilkan pola dengan jumlah kombinasi 10 itemset. Nilai *support* tertinggi sejumlah 51% dan nilai *confidence* tertinggi sejumlah 98% serta nilai *lift ratio* tertinggi sejumlah 1.1941 pada pola *itemset*: jika nilai matakuliah dasar pemrograman lulus, aljabar linear lulus, struktur data lulus, dan metode numerik lulus maka matakuliah kalkulus lulus, matematika diskrit lulus, algoritma pemrograman lulus, teknologi informasi web lulus, probabilitas dan statistik lulus, dan matakuliah pemrograman bergerak lulus. Hal ini menunjukkan bahwa jika mahasiswa lulus pada matakuliah pemrograman, maka pada matakuliah matematika juga dapat lulus atau sebaliknya. Penelitian berikutnya dapat mempertimbangkan kemungkinan lain yang menyebabkan rendahnya nilai matakuliah pemrograman dari aspek pendapat/aspirasi mahasiswa.

REFERENSI

- [1] UIN Suska Riau, *Panduan Dan Informasi Akademik*. Pekanbaru: Suska Press, 2018.
- [2] B. D. Meilani and M. Asadulloh, "Data Mining untuk Menggali Pola Mahasiswa Baru Menggunakan Metode Frequent Pattern Growth (Studi Kasus: Institut Teknologi Adhi Tama Surabaya)," in *Prosiding Seminar Nasional Sains dan Teknologi Terapan*, 2015, pp. 269–276.
- [3] R. Anggrainingsih, N. R. Khoirudin, and H. Setiadi, "Discovering drugs combination pattern using FP-growth algorithm," in *2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 2017, pp. 1–4.
- [4] A. A. Fajrin and A. Maulana, "Penerapan Data Mining Untuk Analisis Pola Pembelian Konsumen Dengan Algoritma Fp-Growth Pada Data Transaksi Penjualan Spare Part Motor," *Kumpulan Jurnal Ilmu Komputer (KLIK)*, vol. 5, no. 01, pp. 1–10, 2018.
- [5] R. R. Mahmudah and E. Aribowo, "Penggunaan Algoritma FP-Growth untuk Menemukan Aturan Asosiasi pada Data Transaksi Penjualan Obat di Apotek (Studi Kasus: Apotek UAD)," *JSTIE (Jurnal Sarjana Teknik Informatika)(E-Journal)*, vol. 2, no. 3, pp. 130–139, 2014.
- [6] N. R. Ardani and N. Fitriana, "Sistem Rekomendasi Pemesanan Sparepart Dengan Algoritma Fp-Growth (Studi Kasus Pt. Rosalia Surakarta)," *SEMNASTEKNOMEDIA ONLINE*, vol. 4, no. 1, p. 3, 2016.
- [7] Ririanti, "Implementasi Algoritma FP-Growth Pada Aplikasi Prediksi Persediaan Sepeda Motor (Studi Kasus PT. Pilar Deli Labumas)," *Pelita Informatika Budi Darma*, vol. 6, no. 1, pp. 139–144, 2014.

- [8] M. I. Ghozali and W. H. Sugiharto, "Analisa Pola Belanja Menggunakan Algoritma Fp Growth, Self Organizing Map (Som) Dan K Medoids," *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, vol. 8, no. 1, pp. 317–326, 2017.
- [9] O. Maimon and L. Rokach, *Data mining and knowledge discovery handbook*. Springer, 2005.
- [10] M. Fauzy and I. Asror, "Penerapan metode association rule menggunakan algoritma apriori pada simulasi prediksi hujan wilayah kota bandung," *Jurnal Ilmiah Teknologi Infomasi Terapan*, vol. 2, no. 3, 2016.