

Sistem Rekomendasi Mata Kuliah Pilihan Menggunakan Association Rule dan Ant Colony Optimization (Studi Kasus Mata Kuliah di Jurusan Teknik Informatika Universitas Telkom)

Aditia Rafif Khoerullah¹, Dade Nurjanah² Ade Romadhony³

^{1,2,3}Fakultas Informatika, Universitas Telkom, Bandung

¹sayaadit@students.telkomuniversity.ac.id, ²dadenurjanah@telkomuniversity.ac.id,

³aderomadhony@telkomuniversity.ac.id

Abstrak

Dalam proses perkuliahan, setiap mahasiswa diwajibkan untuk menyelesaikan setiap mata kuliah dengan sejumlah sks yang ditentukan oleh institusi. Ada mata kuliah yang bersifat wajib dan pilihan. Mata kuliah pilihan berdampak pada fokus bidang penelitian yang akan diambil. Pemilihan hal tersebut membantu dalam pengerjaan tugas akhir yang lebih baik. Pemilihan mata kuliah yang tidak sesuai dengan riwayat nilai dan ekspektasi mahasiswa bisa menyebabkan kesulitan dalam penyelesaian tugas akhir. Pada penelitian ini akan dirancang sistem rekomendasi mata kuliah pilihan menggunakan metode *brute force*, *association rule* dan metode *ant colony optimization* (ACO). Metode *brute force* digunakan untuk mendapatkan rekomendasi mata kuliah pilihan berdasarkan riwayat nilai. Algoritma apriori pada metode *association rule* digunakan untuk menemukan asosiasi setiap mata kuliah pilihan. Sedangkan algoritma *ant cycle* pada metode ACO digunakan untuk mengoptimasi hasil pencarian *rule* sebagai rekomendasi *learning path* pengambilan mata kuliah pilihan. Kemudian hasil dari ketiga algoritma tersebut digabungkan untuk hasil rekomendasi yang lebih baik. Hasil penelitian menunjukkan bahwa algoritma *brute force* dapat memberikan rekomendasi sesuai riwayat nilai. Algoritma apriori pada metode *association rule* dapat menghasilkan *rule* untuk setiap kelompok keahlian dan algoritma *ant cycle* pada algoritma ACO dapat memberikan rekomendasi berupa *rule* pemilihan mata kuliah pilihan dan *learning path* mata kuliah pilihan. Hasil penggabungan tiga algoritma dapat menghasilkan rekomendasi mata kuliah pilihan dengan baik.

Kata kunci: *course recommendation, association rule, ant colony optimization (ACO), apriori, ant cycle, brute force*

Abstract

In the lecture process, each student is required to complete each course with several credits determined by the institution. Some courses are mandatory and optional. Elective courses have an impact on the focus of the research area to be taken. The selection of these things helps in the completion of the final project. Selection of courses that are not by the history of grades and expectations of students can cause difficulties in completing the final project. In this study, a recommendation subject system will be designed using the brute force method, association rule, and the ant colony optimization (ACO) method. The brute force method is used to obtain elective course recommendations based on historical grades. A priori algorithm in the association rule method is used to find the association of each chosen subject. While the ant cycle algorithm in the ACO method is used to optimize the rule search results as a recommendation for learning path taking elective courses. Then the results of the three algorithms are combined for better recommendation results. The results showed that the brute force algorithm can provide recommendations according to historical values. A priori algorithm in the association rule method can produce rules for each group of expertise and the ant cycle algorithm in the ACO algorithm can provide recommendations in the form of rules for selecting the subject and learning path for selected subjects. The results of combining the three algorithms can produce a choice of elective courses well.

Keywords: *course recommendation, association rule, ant colony optimization (ACO), apriori, ant cycle, brute force*

1. Pendahuluan

Latar Belakang

Rencana pengambilan mata kuliah merupakan sebuah kegiatan yang lazim dilakukan oleh setiap mahasiswa. Perencanaan yang dilakukan berpengaruh terhadap bidang spesifik yang akan diambil untuk menyelesaikan tugas akhir mahasiswa sebagai syarat kelulusan studinya. Beberapa institusi telah menyediakan

skema untuk menyusun perencanaan mata kuliah tersebut dengan memberikan sistem paket agar capaian pembelajaran dapat lebih terstruktur dan terukur. Sistem pengambilan mata kuliah ini berlaku untuk setiap mata kuliah wajib dan juga mata kuliah pilihan. Mata kuliah wajib merupakan wawasan dan capaian belajar yang bersifat wajib diambil. Sedangkan, mata kuliah pilihan bersifat opsional sesuai minat dari mahasiswa yang bersangkutan. Selain diambil berdasarkan minat, mata kuliah pilihan juga diambil berdasarkan nilai mata kuliah wajib yang bersifat sebagai dasar penunjang. Setiap jurusan dalam setiap institusi pendidikan tinggi memiliki pengelompokan untuk setiap bidang penelitian yang biasa disebut kelompok keahlian (KK).

Pada jurusan Teknik Informatika Universitas Telkom terdapat tiga KK, yaitu SIDE, ICM, dan Telematika [1]. Permasalahan dapat terjadi ketika mahasiswa mengambil mata kuliah pilihan tanpa mempertimbangkan riwayat nilai mata kuliah wajib penunjangnya. Hal ini sering terjadi karena kurangnya pengawasan dari dosen wali yang masih bersifat manual dan ekspektasi mahasiswa yang tidak relevan terhadap mata kuliah pilihan yang diinginkan sehingga menyebabkan pengambilan mata kuliah pilihan yang tidak sesuai kemampuan.

Pada penelitian sebelumnya, sistem rekomendasi menggunakan beberapa metode diantaranya metode rekomendasi, association rule, content-based, collaborative filtering, dan menggabungkan beberapa metode atau hybrid [4]. Association rule merupakan metode rekomen sistem yang melakukan rekomendasi berdasarkan aturan aturan yang dirancang sesuai dengan kebutuhan aturan pembelajaran [9]. Metode content-based merupakan metode rekomen sistem yang melakukan rekomendasi berdasarkan deskripsi feedback penggunaannya di masa lalu, akan tetapi akurasi rekomen sistemnya sangat bergantung pada preferensi pengguna sehingga sistem hanya merekomendasikan yang diinginkan namun belum tentu dibutuhkan. Collaborative filtering merupakan metode dalam rekomen sistem yang memberikan rekomendasi berdasarkan rating atau pembobotan yang dilakukan oleh pengguna lain [8]. Penghitungan similarity berdasarkan rating orang lain bisa memunculkan hasil rekomendasi yang tidak sesuai dengan karakter pengguna. Hybrid merupakan metode dalam rekomen sistem dengan menggabungkan beberapa metode untuk menutupi kekurangan dari setiap metode.

Penelitian tentang course recommendation menggunakan algoritma berbasis kecerdasan kolektif telah dilakukan. Beberapa algoritma seperti ant colony optimization (ACO), particle swarm optimization (PSO), intelligent weed optimization (IWO), bee colony optimization (BCO), dan bat algorithm (BA) digunakan untuk menghitung similarity yang dilakukan pada data set yang sama menunjukkan hasil yang menjanjikan. Terutama penggunaan algoritma ACO yang menghasilkan akurasi hingga 86% pada kasus klasifikasi [7]. Hal tersebut dapat dijadikan sebagai acuan untuk membangun sebuah sistem rekomendasi yang dapat merekomendasikan mata kuliah pilihan yang tepat. Sistem dapat dibangun dengan menggunakan metode aturan asosiasi [2] dan metode optimasi menggunakan Ant Colony Optimization (ACO) [3][5]. Hal tersebut bertujuan agar hasil rekomendasi sesuai dengan aturan aturan mata kuliah pilihan yang berlaku. Kemudian, hasil rekomedasi dioptimasi menggunakan algoritma ACO sehingga menghasilkan hasil rekomendasi yang lebih sesuai dengan riwayat nilai mahasiswa [6].

Topik dan Batasannya

Tugas Akhir ini menggunakan algoritma Apriori untuk mencari *association rule* dari mata kuliah pilihan setiap mahasiswa dan *Ant Colony Optimization (ACO)*.

Masalah yang akan dibahas pada Tugas Akhir ini adalah:

1. Bagaimana cara membangun rekomen sistem untuk pengambilan mata kuliah pilihan berdasarkan riwayat nilai menggunakan algoritma *brute force*?
2. Bagaimana cara membangun rekomen sistem untuk mendapatkan *rule* asosiasi antar mata kuliah pilihan menggunakan algoritma apriori pada metode *association rule*?
3. Bagaimana cara membangun rekomen sistem untuk mengoptimasi hasil algoritma apriori menjadi sebuah *learning path* menggunakan algoritma *ant cycle* pada metode ACO?

Tugas Akhir ini memiliki batasan dan ruang lingkup penilaian yang mencakup:

1. Tidak membandingkan algoritma pencarian ACO terhadap algoritma pencarian lainnya.
2. Penelitian ini berfokus memberikan rekomendasi mata kuliah pilihan bagi mahasiswa.
3. Data yang digunakan adalah data nilai mahasiswa aktif 2015-2016 yang sudah lulus mata kuliah dasar.

Tujuan

Berdasarkan masalah yang ada, tujuan yang ingin dicapai dalam tugas akhir ini adalah membangun sistem rekomendasi untuk melakukan terhadap pengambilan mata kuliah pilihan menggunakan algoritma *brute force*, metode *association rule* menggunakan algoritma apriori, dan metode ACO menggunakan algoritma *ant cycle* yang sesuai dengan *data set* yang digunakan.

Organisasi Tulisan

Tugas Akhir ini disusun dengan struktur sebagai berikut. Setelah dijelaskan pendahuluan pada bagian pertama, pada bagian kedua menyantumkan studi terkait, selanjutnya dijelaskan pemodelan sistem pada bagian ketiga, setelah itu, dijelaskan evaluasi terhadap sistem yang dibangun pada bagian keempat. Pada bagian kelima dijelaskan kesimpulan dan saran untuk penelitian selanjutnya.

2. Studi Terkait

2.1 Algoritma Brute force

Algoritma ini merupakan algoritma pencarian paling sederhana untuk menemukan sebuah pola yang diinginkan. Kebutuhan dari algoritma brute-force adalah deskripsi kondisi, seperangkat operator logika, keadaan awal, dan deskripsi keadaan tujuan. Pencarian *brute force* juga disebut pencarian pencarian buta. Algoritma *brute force* harus dilanjutkan dengan cara yang sistematis dengan menjelajahi node dalam beberapa urutan yang telah ditentukan. Setelah melakukan seluruh pencarian pembobotan disimpan [20]. Algoritma pencarian *brute force* dapat dilihat pada gambar 1.

```

c ← first(P)
while c ≠ Λ do
  if valid(P,c) then output(P, c)
  c ← next(P,c)
end while

```

Gambar 1. Pseudocode Algoritma Brute Force

2.2 Metode Association Rule

Algoritma ini mulai diperkenalkan pada tahun 1993 dan memberikan dampak yang signifikan. Disaat banyak pola data mining masih menggunakan konsep *Knowledge Discovery in Databases* (KDD). Secara garis besar aturan asosiasi merupakan ekspresi dari $X \rightarrow Y$, dimana X dan Y merupakan data *set item* [8]. Ide dari *mining association rule* adalah mengenali pola kemunculan itemset pada sebuah dataset sehingga bisa diambil informasi. [9].

2.2.1 Algoritma Apriori

Algoritma Apriori merupakan salah satu algoritma untuk mendapatkan kandidat item set dengan menggunakan metode *association rule* [11]. Algoritma ini memiliki dua tahapan utama yaitu, melakukan pencarian kandidat *itemset*, lalu kandidat itemset yang tidak diinginkan akan buang atau biasa disebut *prune* [12]. Penentuan itemset terpilih dan yang dibuang ditentukan oleh nilai minimum support, minimum confidence [13][14]. Nilai *support* merupakan indikasi seberapa sering itemset muncul di dataset. Sedangkan *confidence* adalah indikasi seberapa sering aturan itu terbukti benar. Formula nilai *support* (1) dan nilai *confidence* (2).

$$\text{Support}(A) = \frac{\text{Record yang Mengandung}(A)}{\text{Total Record}} \quad (1)$$

$$\text{Confident}(A \rightarrow B) = \frac{\text{Record yang Mengandung}(A \text{ dan } B)}{\text{Total Record Mengandung}(A)} \quad (2)$$

Pada penelitian [12], algoritma apriori diterapkan untuk menentukan mata kuliah pilihan yang sesuai dengan kebutuhan mahasiswa Telkom University, penentuan rekomendasi berdasarkan penjurusan yang tersedia, algoritma apriori menghasilkan 35 rule kuat yang didapat dari data nilai 660 data nilai [12].

Berikut contoh pencarian rule menggunakan nilai minimum support dan minimum confidence pada pengambilan mata kuliah pilihan tabel 1:

Tabel 1. Contoh Data Mahasiswa Mengambil Mata Kuliah

id_Mahasiswa	Mengambil Mata Kuliah
1	C, E, D
2	A, F, D
3	D, G, B, F
4	E, D, G, B
5	B, A, C
6	F, A, B, G
7	G, D

8	C, G, E
9	F, A, B
10	B, D

Setelah itu memisahkan masing-masing contoh mata kuliah yang diambil sehingga menghasilkan itemset yang ada pada tabel 2.

Tabel 2. Contoh Data Mahasiswa Mengambil Mata Kuliah

Mata Kuliah Diambil
A
B
C
D
E
F
G

Berdasarkan itemset yang didapat, dibuatlah tabel untuk diolah menggunakan algoritma Apriori seperti pada tabel 3.

Tabel 3. Contoh Tabel Data Mahasiswa Siap Diolah Menggunakan Algoritma Apriori

id_mahasiswa	A	B	C	D	E	F	G
1	0	0	1	1	1	0	0
2	1	0	0	1	0	1	0
3	0	1	0	1	0	1	1
4	0	1	0	1	1	0	1
5	1	1	1	0	0	0	0
6	1	1	0	0	0	1	1
7	0	0	0	1	0	0	1
8	0	0	1	0	1	0	1
9	1	1	0	0	0	1	0
10	0	1	0	1	0	0	0
Σ	4	6	3	6	3	4	5

Kemudian dilakukan perhitungan nilai support dan confidence untuk mendapatkan *rule* sesuai dengan nilai confident=0,60 yang ditentukan sehingga didapat *rule* dengan perhitungan pada tabel 4.

Tabel 4. Contoh Data Mahasiswa Mengambil Mata Kuliah

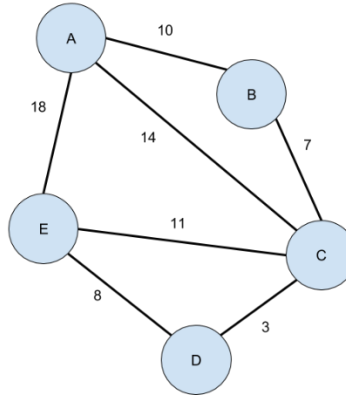
Rule Mata Kuliah	Support	Support%	Confidence	Confidence%
A $\rightarrow$ B	3/10	30%	3/4	75%
A $\rightarrow$ F	3/10	30%	3/4	75%
F $\rightarrow$ A	3/10	30%	3/4	75%
F $\rightarrow$ B	3/10	30%	3/4	75%
G $\rightarrow$ B	3/10	30%	3/5	60%
G $\rightarrow$ D	3/10	30%	3/5	60%

2.3 Metode Ant Colony Optimization

Kebiasaan sosial yang kompleks dari semut sudah dipelajari dan para peneliti sudah menemukan pola pergerakan semut ketika mencari makanan. Kebiasaan tersebut dibuat menjadi pemodelan algoritma untuk menyelesaikan permasalahan kombinatorial optimisasi yang sulit. Percobaan untuk mengembangkan algoritma yang terinspirasi dari kebiasaan semut yang memiliki kemampuan untuk menentukan jalur terpendek, kini sudah menjadi sebuah bidang riset tersendiri yaitu, *ant colony optimization* (ACO) [3]. ACO pertama kali diperkenalkan

oleh Marco Dorigi, pada tahun 1992 dan dirancang untuk menyelesaikan permasalahan optimasi diskrit. Kini sedang dikembangkan untuk bisa menyelesaikan permasalahan kontinu [10].

Ant colony optimization (ACO) merupakan salah satu dari algoritma kecerdasan kolektif. ACO merupakan algoritma yang bersifat *meta heuristic* dan cocok digunakan untuk menyelesaikan berbagai masalah optimasi. Pada awal pengujian metode ini, ACO diajukan untuk menyelesaikan permasalahan diskrit. Rekomendasi mata kuliah menggunakan nilai sebagai data set bisa dikategorikan kedalam permasalahan diskrit bila dimodelkan kedalam konsep *graph* seperti pada gambar 2.



Gambar 2. *Graph Sales Traveling Problem*

2.3.1 Algoritma Ant Cycle

Ant cycle merupakan algoritma dari ACO yang merupakan hasil modifikasi yang dapat digunakan untuk kasus kasus lainnya [10]. Implementasinya memiliki beberapa tahapan, diantaranya [10]:

1. Inisialisasi ruang solusi dan inisialisasi jumlah semut,
2. Perhitungan perpindahan semut,
3. Menghitung nilai probabilistik perpindahan semut menggunakan persamaan (3),

$$p_{xy}^k = \frac{(\tau_{xy}^\alpha)(\eta_{xy}^\beta)}{\sum_{z \in \text{allowed}_x} (\tau_{xz}^\alpha)(\eta_{xz}^\beta)} \quad (3)$$

4. *Update* pheromone dengan parameter evaporasi menggunakan persamaan (4),

$$\tau_{ij}(t+n) = \rho \tau_{ij}(t) + \Delta \tau_{ij} \quad (4)$$

5. Melakukan iterasi sampai mendapatkan nilai stagnan.

Pada penelitian [6] *date set* yang digunakan diambil dari *University Information System EdukacjaCL* sebanyak 479359 nilai dari 9055 mata kuliah yang diikuti oleh 13574 mahasiswa. Dengan memberikan parameter $\alpha = 0.4$, $\beta = 0.4$, dan dengan jumlah semut sebanyak 100 ekor. Hasil perhitungan eksekusi algoritma menggunakan *Prediction Accuracy (PA)* sebesar 0,88 [6].

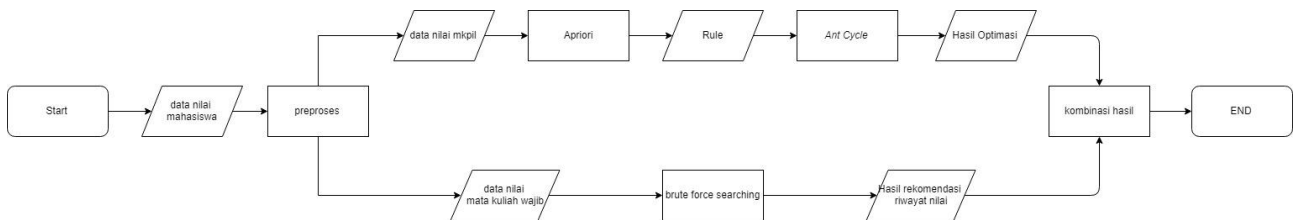
Pada penelitian [15], pendekatan ACO digunakan untuk membantu penentuan *rule*. Selain itu, ACO juga dapat membantu penentuan *rule* dengan lebih efisien seperti pada penelitian [19]. Pendekatan ACO yang dilakukan adalah dengan memodifikasi update bobot dengan menggunakan nilai *support* dan *confidence* yang didapat pada persamaan (5).

$$\text{Jarak} = \frac{2 \cdot \text{support} \cdot \text{confidence}}{\text{support} + \text{confidence}} \quad (5)$$

Persamaan (5) dapat digunakan untuk mendapatkan jarak antar node antar *rule* sehingga membentuk *graph* yang saling terhubung [18]. Setelah itu *rule* hasil algoritma apriori dapat dioptimasi dengan menggunakan *ant cycle* [16] dengan pemodelan pada *traveling salesman problem* untuk mendapatkan hasil jarak terbaik.

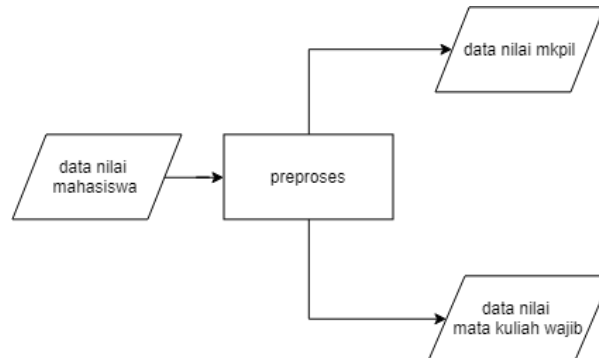
3. Rancangan Sistem Rekomendasi Association Rule dan Ant Colony Optimization

Sub-bab ini menjelaskan langkah langkah mengolah data sampai mendapatkan hasil rekomendasi dalam bentuk kombinasi hasil pencarian ACO dan *rule* untuk setiap mahasiswa yang ada di Universitas Telkom. Gambar 3 menunjukkan gambaran umum sistem yang dibangun pada tugas akhir ini.



Gambar 3. Gambaran umum sistem

3.1 Preproses Data Set Nilai Mahasiswa



Gambar 4. Alur Preprocessing Data Set Nilai Mahasiswa

Pada gambar 4, masukan dataset yang digunakan berupa nilai mahasiswa program S1 Teknik Informatika tahun 2015-2016 yang sesuai dengan kurikulum 2016 Teknik Informatika dengan ketentuan mahasiswa telah lulus pada mata kuliah dasar. Data didapat dari direktorat Sistem Informasi (SISFO) Universitas Telkom, dengan rincian NIM, nama mata kuliah, nilai, dan indeks nilai. Ukuran data keseluruhan 838x79 dengan baris menyatakan mahasiswa dan kolom menyatakan mata kuliah. Sample data dapat dilihat pada tabel 5.

Tabel 5. Sample data

NIM	ANALISIS PERFORMANSI JARINGAN KOMPUTER	ARSITEKTUR BERORIENTASI LAYANAN	CONVOLUTIONAL NEURAL NETWORK	TOPIK KHUSUS 3 SIDE	TOPIK KHUSUS 5 SIDE	...	...	MATEMATIKA DISKRIT	KALKULUS
s5a9a85a179e979c			A			...	...	B	A
sc6931d0f81c7ec9	B					...	...	AB	C
sc1d3694bbd94d33		BC	AB			...	...	AB	AB
...	...	...	...	...	...	...	...	...	...

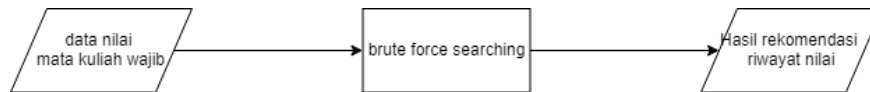
Preproses *data set* dilakukan dengan membagi dua kolom pada *data set* berdasarkan mata kuliah wajib dan mata kuliah pilihan. Hasil dari tahapan ini berupa dua data set, yaitu matriks mata kuliah wajib dan matriks mata kuliah pilihan. Ukuran matriks mata kuliah wajib adalah 838x26 dan ukuran matriks mata kuliah pilihan adalah 838x52. Representasi baris pada setiap matriks menunjukkan indeks mahasiswa sedangkan representasi baris pada setiap matriks adalah nilai yang dicapai oleh mahasiswa.

Setiap matriks memiliki tahapan *encoding* yang berbeda disesuaikan dengan kebutuhan. Matriks mata kuliah wajib dengan mengganti nilai A=8, AB=7, B=6, BC=5, C=4, D=2, dan E,T = 0. Hal tersebut bertujuan agar proses rekomendasi mata kuliah pilihan berdasarkan riwayat nilai mendapatkan hasil dengan bobot tertinggi.

Matriks mata kuliah pilihan dengan mengganti nilai A, AB, B, BC, dan C dengan nilai 1. Sedangkan nilai D, E, T dengan nilai 0. Representasi dari nilai 1 adalah menunjukkan bahwa mahasiswa tersebut telah mengambil dan lulus mata kuliah pilihan tersebut. Sedangkan, representasi nilai 0 adalah menunjukkan bahwa mahasiswa tersebut tidak mengambil mata kuliah pilihan tersebut atau mengambil mata kuliah pilihan namun tidak lulus mata kuliah pilihan tersebut.

Setelah proses *encoding* selesai matriks mata kuliah wajib dan mata kuliah pilihan sudah siap diolah menggunakan algoritma *brute force* dan algoritma apriori pada tahap selanjutnya.

3.2 Rekomendasi Mata Kuliah Pilihan Berdasarkan Riwayat Nilai Menggunakan Algoritma Brute Force



Gambar 5. Alur Preprocessing Data Set Nilai Mahasiswa

Pada gambar 5, masukan untuk algoritma *brute force* adalah matriks mata kuliah wajib dengan ukuran 838x26. Matriks tersebut diolah oleh algoritma *brute force* sehingga menghasilkan luaran daftar mata kuliah pilihan dengan bobot tertinggi berdasarkan riwayat nilai. Setelah data nilai mata kuliah wajib siap digunakan, pencarian mata kuliah pilihan menggunakan algoritma *brute force* dicocokkan dengan kondisi yang sesuai dengan aturan *prerequisite* atau syarat pengambilan mata kuliah pilihan terkait [1].

```

hasil_rekom = []
list_mk_temp = []

for i to 838
  x = i
  if (Nilai[x][1]>3) & (Nilai[x][3]>3) & (Nilai[x][4]>3) ...
    list_mk_temp <-- [nama_mk, bobot_nilai]
  if (Nilai[x][10]>3) & (Nilai[x][11]>3) & (Nilai[x][18]>3) ...
    list_mk_temp <-- [nama_mk, bobot_nilai]
  ...
  sorted_10 (list_mk_temp)
endfor
  
```

Gambar 6. Algoritma Brute Force

Setiap nilai mahasiswa, akan ditinjau nilainya sesuai dengan aturan yang berlaku, apabila mahasiswa memenuhi kondisi maka algoritma akan menyimpan data mata kuliah pilihan dan menyimpan bobot yang didapatkan. Setelah peninjauan semua nilai selesai, dilakukan pengurutan berdasarkan bobot terbesar sehingga setiap indeks mahasiswa mendapatkan sepuluh nilai terbaik [20]. Algoritma *brute force* untuk mendapatkan rekomendasi berdasarkan riwayat nilai dapat dilihat pada gambar 6.

3.3 Pencarian Rule Menggunakan Algoritma Apriori



Gambar 7. Alur Tahap Pencarian Rule oleh Algoritma Apriori

Pada gambar 7, masukan untuk algoritma apriori adalah matriks mata kuliah pilihan. Diolah oleh algoritma apriori sehingga menghasilkan luaran rule asosiasi antar mata kuliah pilihan dengan kriteria lebih dari nilai *minimum support* dan *minimum confidence* yang ditentukan. Algoritma Apriori merupakan salah satu algoritma untuk mendapatkan kandidat item set dengan menggunakan metode association rule [11]. Algoritma ini memiliki dua tahapan utama yaitu, melakukan pencarian kandidat *itemset*, lalu kandidat *itemset* yang tidak diinginkan akan buang atau biasa disebut *prune* [12]. Penentuan itemset terpilih dan yang dibuang ditentukan oleh nilai minimum support, minimum confidence [13][14]. Algoritma dapat dilihat pada gambar 8.

```

Ck: kandidat mata kuliah pilihan sebanyak k
Lk : frequent mata kuliah pilihan sebanyak k
L1 = {frequent items};
for (k = 1; Lk !=0; k++) do begin
  Ck+1 = menghasilkan kandidat dari Lk;
  for nilai mahasiswa t in database do
    menambah jumlah semua kandidat dalam Ck + 1 yang terkandung dalam t
  Lk+1 = kandidat dalam Ck+1 dengan min_support
  end
return Uk Lk;
  
```

Gambar 8. Algoritma Apriori Untuk Pencarian Rule

Nilai *support* merupakan indikasi seberapa sering itemset muncul di dataset. Sedangkan *confidence* adalah indikasi seberapa sering aturan itu terbukti benar. Proses perhitungan nilai support dan nilai confidence suatu mata kuliah pilihan dapat dilihat pada persamaan (1) dan (2).

$$Support(A) = \frac{\sum \text{Mahasiswa Mengambil MK (A)}}{\sum \text{Mahasiswa Mengambil MK}} \quad (1)$$

$$\text{Confident } (A \rightarrow B) = \frac{\Sigma \text{ Mahasiswa Mengambil MK (A dan B)}}{\Sigma \text{ Total Mahasiswa Mengambil MK (A)}} \quad (2)$$

Skenario pengujian untuk algoritma ini adalah membandingkan jumlah *rule* yang dibentuk dengan melakukan kombinasi nilai *minimum support* dan *minimum confidence* pada algoritma apriori. Rentang nilai yang digunakan diambil pada penelitian sebelumnya [12]. Setelah semua tahapan selesai, akan mendapatkan *rule* untuk setiap mata kuliah pilihan yang dapat digunakan sebagai untuk tahap optimasi selanjutnya menggunakan *ant cycle*.

3.4 Optimasi Hasil Rule Menggunakan Algoritma Ant Cycle



Gambar 9. Alur Tahap Optimasi Rule Oleh Algoritma Ant Cycle

Pada gambar 9, masukan untuk algoritma *ant cycle* adalah *rule* hasil algoritma apriori. Sebelum diolah oleh algoritma *ant cycle*, *rule* hasil algoritma apriori dipisahkan berdasarkan kelompok keahlian (KK) pada lampiran 2. Kemudian, *rule* hasil algoritma apriori yang telah dipisah sesuai KK, diubah menjadi *graph* dengan node yang saling terhubung menggunakan tahapan preproses [18]. Hasil dari matriks yang saling terhubung direpresentasikan dalam bentuk matriks agar siap diolah menggunakan algoritma *ant cycle* [17] [19]. Penentuan jarak untuk setiap node mata kuliah pilihan dalam *graph* ditentukan oleh persamaan (5). Setelah jarak setiap node mata kuliah pilihan selesai, matriks *rule* setiap KK siap dioptimasi oleh *ant cycle*.

$$\text{Jarak} = \frac{2 \cdot \text{support} \cdot \text{confidence}}{\text{support} + \text{confidence}} \quad (5)$$

Hasil preproses menghasilkan 3 matriks *rule* mata kuliah pilhan setiap KK. Matriks *rule* mata kuliah pilihan akan diolah sehingga menghasilkan luaran berupa *rule learning path* berupa alur pengambilan tiga mata kuliah pilihan dalam KK yang sama. *Learning path* tersebut diharapkan menjadi referensi perencanaan pengambilan mata kuliah pilihan. Kemudian ketiga matriks tersebut diolah menggunakan algoritma *ant cycle* dapat dilihat pada gambar 10.

```

procedure AntCycle-LearningPath-MKpil
PencarianJalur
    SemutMemilihJalurMKPil ()
    EvaporasiPheromone ()
End PencarianJalur
end AntCycle-LearningPath-MKpil
  
```

Gambar 10. Algoritma Ant Cycle

Pada tahapan *SemutMemilihJalurMKPIL* dilakukan perhitungan peluang setiap kali perpindahan semut dihitung oleh persamaan (3). Hasil dari persamaan (3) adalah nilai peluang yang akan menentukan keputusan perpindahan semut menuju *node* terbaik.

$$p_{xy}^k = \frac{(\tau_{xy}^\alpha)(\eta_{xy}^\beta)}{\sum_{z \in \text{allowed}_x} (\tau_{xz}^\alpha)(\eta_{xz}^\beta)} \quad (3)$$

Dimana, P_{xy} adalah nilai peluang terpilihnya *node* y. Nilai τ_{ij} merupakan jumlah *pheromone* yang ada pada setiap *edge* x,y. $\eta_{x,y}$ merupakan nilai kecenderungan untuk memilih sebuah *node*. Nilai τ_{ij} dipengaruhi oleh parameter α , sedangkan nilai $\eta_{x,y}$ dipengaruhi oleh parameter β .

Pada tahapan *EvaporasiPheromone*, semut yang telah menentukan nilai peluang terbaik akan mengupdate *pheromone* yang dipengaruhi oleh nilai evaporasi. Semakin besar nilai *pheromone* agar menunjukkan bahwa semut telah melewati jalur tersebut. Perhitungan update *pheromone* dan parameter evaporasi dihitung dengan menggunakan persamaan (4).

$$\tau_{ij}(t+n) = \rho \tau_{ij}(t) + \Delta \tau_{ij} \quad (4)$$

Dimana, τ_{ij} merupakan jumlah *pheromone* yang ada di dalam tepi i,j. Sedangkan, ρ adalah nilai *rate* evaporasi, dan $\Delta \tau_{ij}$ adalah performansi pencarian dari semut. Selama pencarian belum selesai, nilai *pheromone*

akan terus diperbaharui.

Performansi algoritma *ant cycle* sangat bergantung terhadap nilai parameter yang diberikan pada algoritma tersebut. Parameter parameter yang ditentukan pada algoritma ini, diantaranya nilai α , β , banyaknya semut, dan nilai evaporasi [21]. Oleh karena itu skenario pengujian yang dilakukan pada algoritma *ant cycle* adalah dengan menemukan parameter yang sesuai dengan jenis implementasi *ant cycle* pada kasus yang optimasi [12].

4. Evaluasi

4.1 Rekomendasi Berdasarkan Riwayat Nilai Menggunakan Algoritma *Brute Force*

Proses pencarian mata kuliah pilihan berdasarkan riwayat nilai dilakukan setelah data nilai mata kuliah telah melewati tahap preproses dan tahap encoding. Luaran dari preproses untuk tahap ini adalah sebuah matrik dengan dimensi 838x27. Baris pada matriks menunjukkan jumlah mahasiswa sedangkan kolom pada matriks menunjukkan daftar nilai mata kuliah wajib yang diambil oleh mahasiswa. Setelah data nilai mata kuliah wajib siap digunakan, pencarian mata kuliah pilihan menggunakan algoritma *brute force* dicocokkan dengan kondisi yang sesuai dengan aturan *prerequisite* mata kuliah pilihan terkait. Setiap nilai mahasiswa, akan ditinjau nilainya sesuai dengan aturan yang berlaku, apabila mahasiswa memenuhi kondisi maka algoritma akan menyimpan data mata kuliah pilihan dan menyimpan bobot yang didapatkan. Setelah peninjauan semua nilai selesai, dilakukan pengurutan berdasarkan bobot terbesar sehingga setiap indeks mahasiswa mendapatkan sepuluh nilai terbaik [20]. Kemudian algoritma mengulangi tahapan yang sama untuk setiap mahasiswa sampai dengan data mahasiswa terakhir. Hasil rekomendasi berdasarkan riwayat nilai dapat dilihat pada gambar 11.

Index	0	1	2	3	4
0	MIKROKONTROL...	SISTEM KEAMANAN	SISTEM PENDUKUNG PE...	KECERDASAN KOLEKTIF	KOMPUTASI EVOLUSI...
1	MIKROKONTROL...	SISTEM KEAMANAN	SISTEM PENDUKUNG PE...	KOMPUTASI EVOLUSIONER	KECERDAS KOLEKTII
2	SISTEM KEAMANAN	MIKROKONTROL...	SISTEM PENDUKUNG PE...	KOMPUTASI EVOLUSIONER	KECERDAS KOLEKTII
3	SISTEM PENGENALAN	TOPIK KHUSUS 2 ICM - PEMR...	KOMPUTASI FINANSIAL	KOMPUTASI LUNAK	KECERDAS KOLEKTII
4	SISTEM PENGENALAN	KOMPUTASI FINANSIAL	TOPIK KHUSUS 2 ICM - PEMR...	KOMPUTASI LUNAK	KOMPUTASI EVOLUSI...
5	SISTEM PENGENALAN	KOMPUTASI LUNAK	KOMPUTASI FINANSIAL	TOPIK KHUSUS 2 ICM - PEMR...	KOMPUTASI EVOLUSI...
6	SISTEM PENGENALAN	KOMPUTASI LUNAK	KOMPUTASI FINANSIAL	TOPIK KHUSUS 2 ICM - PEMR...	KOMPUTASI EVOLUSI...
7	SISTEM PENGENALAN	KOMPUTASI LUNAK	TOPIK KHUSUS 2 ICM - PEMR...	KOMPUTASI FINANSIAL	KECERDAS KOLEKTII
8	SISTEM PENGENALAN	KOMPUTASI FINANSIAL	KOMPUTASI LUNAK	TOPIK KHUSUS 2 ICM - PEMR...	KOMPUTASI EVOLUSI...
9	SISTEM PENGENALAN	KOMPUTASI LUNAK	KOMPUTASI EVOLUSIONER	TOPIK KHUSUS 2 ICM - PEMR...	FINANSI...
10	SISTEM PENGENALAN	KOMPUTASI LUNAK	KOMPUTASI FINANSIAL	KOMPUTASI EVOLUSIONER	KECERDAS KOLEKTII
11	SISTEM PENGENALAN	KOMPUTASI LUNAK	KOMPUTASI FINANSIAL	TOPIK KHUSUS 2 ICM - PEMR...	KECERDAS KOLEKTII
12	SISTEM PENGENALAN	KOMPUTASI EVOLUSIONER	KECERDASAN KOLEKTIF	KOMPUTASI LUNAK	KOMPUTASI FINANSI...
13	SISTEM	KOMPUTASI	KECERDASAN	KOMPUTASI	KOMPUTASI

Gambar 11. Hasil Rekomendasi Berdasarkan Riwayat Nilai

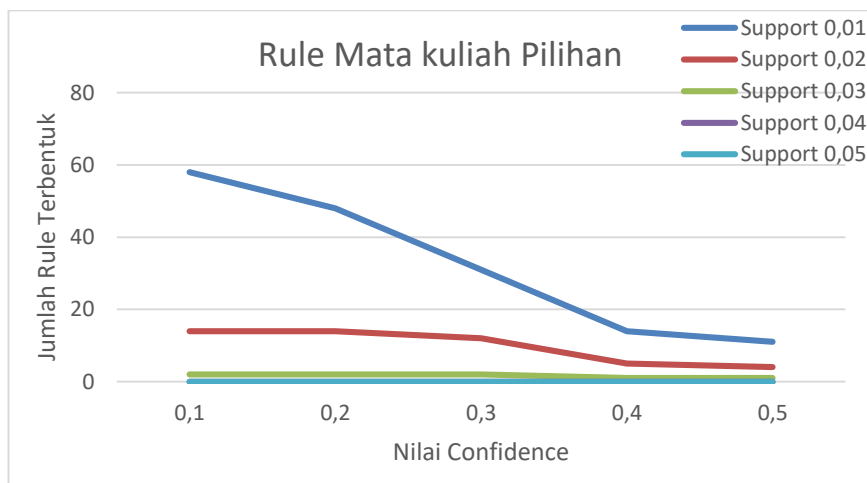
4.2 Pengujian Rule Menggunakan Nilai *Support* dan *Confidence*

Pembentukan rule sebelumnya dimulai dengan menentukan nilai minimum support dan minimum confidence yang akan digunakan. Nilai tersebut akan mempengaruhi hasil algoritma apriori ketika pembentukan rule. Setiap dataset memiliki karakter yang berbeda beda, oleh karena itu diperlukan pengujian untuk mendapatkan kombinasi nilai support dan confidence yang diinginkan. Berikut adalah hasil pengujian kombinasi nilai *minimum support* dan *minimum confidence* dengan rentang nilai pengujian seperti pada penelitian [12]. Hasil pengujian nilai *minimum support* dan *minimum confidence* terhadap rule yang dibentuk dapat dilihat pada tabel 6 dan gambar 12.

Tabel 6. Hasil pengujian nilai minimum support dan minimum confidence

Support	Confidence	Itemset	Rule yang Dibentuk
0.05	0.1	16	0
	0.2		0
	0.3		0
	0.4		0
	0.5		0

0.04	0.1	21	0
	0.2		0
	0.3		0
	0.4		0
	0.5		0
0.03	0.1	26	2
	0.2		2
	0.3		2
	0.4		1
	0.5		1
0.02	0.1	50	14
	0.2		14
	0.3		12
	0.4		5
	0.5		4
0.01	0.1	112	58
	0.2		48
	0.3		31
	0.4		14
	0.5		11



Gambar 12. Diagram Perbandingan Rule Terbentuk

Pengaruh nilai *minium support* dan *minimum confidence* pada *dataset* terhadap jumlah rule yang terbentuk untuk semua mata kuliah pilihan:

1. *Minimum support* = 0.05 dan *minimum confidence* = 0.1; menghasilkan 16 *itemset* dan 0 *rule*. Dikarenakan *itemset* yang terlalu sedikit sehingga tidak menghasilkan *rule*. Oleh karena itu, parameter *minimum support* = 0.05 dan *minimum confidence* = 0.1 tidak dapat digunakan.
2. *Minimum Support* = 0.04 dan *minimum confidence* = 0.1; menghasilkan 21 *itemset* dan 0 *rule*. Dikarenakan *itemset* yang terlalu sedikit sehingga tidak menghasilkan *rule*. Oleh karena itu, parameter *minimum support* = 0.04 dan *minimum confidence* = 0.1 tidak dapat digunakan.
3. *Minimum Support* = 0.03 dan *minimum confidence* = 0.1; menghasilkan 26 *itemset* dan 2 *rule*. Dikarenakan *itemset* yang terlalu sedikit sehingga menghasilkan *rule* yang terlalu sedikit pula bila dibandingkan dengan jumlah mata kuliah pilihan yang tersedia. Oleh karena itu, parameter *minimum Support* = 0.03 dan *minimum confidence* = 0.1 tidak dapat digunakan.
4. *Minimum Support* = 0.02 dan *minimum confidence* = 0.1; menghasilkan 50 *itemset* dan 14 *rule*. Dikarenakan *itemset* yang terlalu sedikit sehingga menghasilkan *rule* yang terlalu sedikit pula bila dibandingkan dengan jumlah mata kuliah pilihan yang tersedia. Oleh karena itu, parameter *minimum Support* = 0.02 dan *minimum confidence* = 0.1 tidak dapat digunakan.
5. *Minimum Support* = 0.01 dan *minimum confidence* = 0.2; menghasilkan 112 *itemset* dan 48 *rule*. Penggunaan nilai tersebut dapat menghasilkan *rule* yang sesuai dengan data mata kuliah pilihan yang tersedia seperti pada penelitian [12], sehingga nilai *minimum support* = 0.01 dan *minimum confidence* = 0.2 dipilih sebagai parameter terbaik pada penelitian ini.

6. *Minimum Support* = 0.01 dan *minimum confidence* = 0.1; menghasilkan 112 *itemset* dan 58 *rule*. Dikarenakan komposisi KK tidak seimbang dengan perbandingan untuk KK TELE, KK SIDE, dan KK ICM adalah 3:2:1 dan seringnya muncul *rule* berulang. Oleh karena itu, parameter *minimum support* = 0.01 dan *minimum confidence* = 0.1 tidak dapat digunakan.

Berdasarkan pengamatan diatas, nilai *minimum support* yang digunakan adalah 0.01 dan *minimum confidence* 0.2, setelah didapat *itemset* kemudian dilakukan proses pencarian *rule* untuk data mata kuliah pilihan. Hasil *rule* dapat dilihat pada tabel 7.

Tabel 7. Rule Mata Kuliah Pilihan yang Didapatkan

No	antecedents	consequents	kk ante	kk consq	status
1	FORENSIK KOMPUTER DAN JARINGAN	ANALISIS PERFORMANSI JARINGAN KOMPUTER	TELE	TELE	SESUAI
2	ANALISIS PERFORMANSI JARINGAN KOMPUTER	FORENSIK KOMPUTER DAN JARINGAN	TELE	TELE	SESUAI
3	JARINGAN SENSOR NIRKABEL	ANALISIS PERFORMANSI JARINGAN KOMPUTER	TELE	TELE	SESUAI
4	KOMPUTASI AWAN	ANALISIS PERFORMANSI JARINGAN KOMPUTER	TELE	TELE	SESUAI
5	NETCENTRIC	ANALISIS PERFORMANSI JARINGAN KOMPUTER	TELE	TELE	SESUAI
6	ANALISIS PERFORMANSI JARINGAN KOMPUTER	NETCENTRIC	TELE	TELE	SESUAI
7	PEMULIHAN BENCANA	ANALISIS PERFORMANSI JARINGAN KOMPUTER	TELE	TELE	SESUAI
8	ANALISIS PERFORMANSI JARINGAN KOMPUTER	PEMULIHAN BENCANA	TELE	TELE	SESUAI
9	PENAMBANGAN DATA	PEMROSESAN BAHASA ALAMI	SIDE	ICM	LINTAS KK
10	PEMROSESAN BAHASA ALAMI	PENAMBANGAN DATA	ICM	SIDE	LINTAS KK
11	PENAMBANGAN DATA	PENAMBANGAN TEKS	SIDE	SIDE	SESUAI
12	PENAMBANGAN TEKS	PENAMBANGAN DATA	SIDE	SIDE	SESUAI
13	SISTEM PENDUKUNG PENGAMBILAN KEPUTUSAN	PENAMBANGAN DATA	SIDE	SIDE	SESUAI
14	DATA WAREHOUSE	PENAMBANGAN TEKS	SIDE	SIDE	SESUAI
15	E-GOVERNMENT	SISTEM INFORMASI KORPORAT	SIDE	SIDE	SESUAI
16	SISTEM INFORMASI KORPORAT	E-GOVERNMENT	SIDE	SIDE	SESUAI
17	FORENSIK KOMPUTER DAN JARINGAN	PEMULIHAN BENCANA	TELE	TELE	SESUAI

18	PEMULIHAN BENCANA	FORENSIK KOMPUTER DAN JARINGAN	TELE	TELE	SESUAI
19	FORENSIK KOMPUTER DAN JARINGAN	SISTEM KEAMANAN	TELE	TELE	SESUAI
20	SISTEM KEAMANAN	FORENSIK KOMPUTER DAN JARINGAN	TELE	TELE	SESUAI
21	SOFTWARE DEFINED NETWORKS	FORENSIK KOMPUTER DAN JARINGAN	TELE	TELE	SESUAI
22	FORENSIK KOMPUTER DAN JARINGAN	SOFTWARE DEFINED NETWORKS	TELE	TELE	SESUAI
23	KONSEP PENGEMBANGAN E-LEARNING	SISTEM INFORMASI KORPORAT	SIDE	SIDE	SESUAI
24	SISTEM INFORMASI KORPORAT	KONSEP PENGEMBANGAN E-LEARNING	SIDE	SIDE	SESUAI
25	TEKNOLOGI PEMBELAJARAN KREATIF	KONSEP PENGEMBANGAN E-LEARNING	SIDE	SIDE	SESUAI
26	MIKROKONTROLER	JARINGAN KOMPUTER LANJUT (INTERNET OF THINGS)	TELE	TELE	SESUAI
27	JARINGAN KOMPUTER LANJUT (INTERNET OF THINGS)	MIKROKONTROLER	TELE	TELE	SESUAI
28	TOPIK KHUSUS 1 TELEMATIKA	MIKROKONTROLER	TELE	TELE	SESUAI
29	MIKROKONTROLER	TOPIK KHUSUS 1 TELEMATIKA	TELE	TELE	SESUAI
30	PEMROSESAN BAHASA ALAMI	PENAMBANGAN TEKS	ICM	SIDE	LINTAS KK
31	SISTEM BERBASIS PENGETAHUAN	PEMROSESAN BAHASA ALAMI	ICM	ICM	SESUAI
32	PEMROSESAN BAHASA ALAMI	SISTEM BERBASIS PENGETAHUAN	ICM	ICM	SESUAI
33	SISTEM PENGENALAN	PEMROSESAN BAHASA ALAMI	ICM	ICM	SESUAI
34	PEMROSESAN BAHASA ALAMI	SISTEM PENGENALAN	ICM	ICM	SESUAI
35	SISTEM PENGENALAN	PENGOLAHAN CITRA DIGITAL	ICM	ICM	SESUAI
36	PENGOLAHAN CITRA DIGITAL	SISTEM PENGENALAN	ICM	ICM	SESUAI
37	SISTEM TERTANAM UNTUK ROBOTIKA	JARINGAN KOMPUTER LANJUT (INTERNET OF THINGS)	TELE	TELE	SESUAI
38	JARINGAN KOMPUTER LANJUT (INTERNET OF THINGS)	SISTEM TERTANAM UNTUK ROBOTIKA	TELE	TELE	SESUAI
39	TOPIK KHUSUS 1 TELEMATIKA	JARINGAN KOMPUTER LANJUT (INTERNET OF THINGS)	TELE	TELE	SESUAI
40	DESAIN INTERAKSI	SISTEM INFORMASI KORPORAT	SIDE	SIDE	SESUAI

41	SISTEM INFORMASI KORPORAT	DESAIN INTERAKSI	SIDE	SIDE	SESUAI
42	PENAMBANGAN DATA , KONSEP PENGEMBANGAN E-LEARNING	PENAMBANGAN TEKS	SIDE	SIDE	SESUAI
43	PENAMBANGAN TEKS , KONSEP PENGEMBANGAN E-LEARNING	PENAMBANGAN DATA	SIDE	SIDE	SESUAI
44	SOFTWARE DEFINED NETWORKS , FORENSIK KOMPUTER DAN JARINGAN	SISTEM KEAMANAN	TELE	TELE	SESUAI
45	SOFTWARE DEFINED NETWORKS , SISTEM KEAMANAN	FORENSIK KOMPUTER DAN JARINGAN	TELE	TELE	SESUAI
46	FORENSIK KOMPUTER DAN JARINGAN , SISTEM KEAMANAN	SOFTWARE DEFINED NETWORKS	TELE	TELE	SESUAI
47	SOFTWARE DEFINED NETWORKS	FORENSIK KOMPUTER DAN JARINGAN , SISTEM KEAMANAN	TELE	TELE	SESUAI
48	FORENSIK KOMPUTER DAN JARINGAN	SOFTWARE DEFINED NETWORKS , SISTEM KEAMANAN	TELE	TELE	SESUAI

4.3 Pengujian Hasil Pencarian Menggunakan ACO

Penentuan parameter pertama adalah parameter evaporasi yang dapat dilihat pada Tabel 6 dengan jumlah semut sebanyak 100 dan 1 semut terbaik yang menyimpan *pheromone*. Menjalankan algoritma *Ant Cycle* perlu menentukan parameter agar bisa menghasilkan *learning path* dengan hasil jarak tertinggi sebagai tolak ukur optimasi *rule* hasil algoritma apriori. Parameter yang ditentukan adalah jumlah semut (n), jumlah semut terbaik (t), tingkat evaporasi, α , dan β . Selanjutnya pengujian akan dilakukan berdasarkan rentang parameter dan kombinasi parameter sesuai dengan [21].

4.3.1 Pengujian Skenario 1: Parameter Evaporasi

Setelah mendapatkan *rule*, data dibagi ke dalam 4 kelompok data KK ICM, KK SIDE, KK TELE dan semua data *rule*. Setelah itu, data *rule* setiap kelompok diproses menjadi matriks yang saling terhubung berdasarkan metode pada paper [18]. Hasil dari preproses menghasilkan 3 matriks dengan ukuran, matriks icm = 4x4, matriks side = 9x9, matriks tele = 12x12. Pengujian nilai evaporasi dengan rentang seperti [21] terhadap jarak setiap KK dapat dilihat pada tabel 8.

Tabel 8. Pengujian Parameter Evaporasi

Skenario	Kelompok keahlian	Evaporasi	α	β	Hasil Jarak	Jarak Total
1	ICM	0,10	1	1	9,803	50,807
	SIDE				17,061	
	TELE				23,943	
2	ICM	0,2	1	1	9,803	50,724
	SIDE				14,633	
	TELE				26,288	
3	ICM	0,3	1	1	12,291	55,464

	SIDE				21,802	
	TELE				21,371	
4	ICM	0,4	1	1	9,803	49,594
	SIDE				11,655	
	TELE				28,136	
5	ICM	0,5	1	1	12,291	50,201
	SIDE				13,967	
	TELE				23,943	
6	ICM	0,6	1	1	15,492	55,656
	SIDE				16,169	
	TELE				23,996	
7	ICM	0,7	1	1	15,492	56,495
	SIDE				17,061	
	TELE				23,943	
8	ICM	0,8	1	1	9,803	49,755
	SIDE				15,660	
	TELE				24,292	
9	ICM	0,95	1	1	9,803	47,332
	SIDE				18,360	
	TELE				19,169	
10	ICM	1	1	1	12,291	53,295
	SIDE				17,061	
	TELE				23,943	

Pada hasil pengujian pada tabel 7 didapat hasil jarak terbanyak sebesar 56,495 dengan parameter evaporasi 0,7. Dilihat dari hasil pengujian cost bergerak meningkat pada parameter evaporasi 0,1 sampai dengan 0,7. Pada parameter evaporasi 0,8 sampai dengan 1 mengalami penurunan. Karena alasan demikian parameter evaporasi 0,7 dipilih sebagai parameter pengujian pada skenario pengujian selanjutnya [21].

4.3.2 Pengujian Skenario 2: Parameter α dan β

Pengaruh nilai α dan β terhadap hasil jarak dapat dilihat pada tabel 9.

Tabel 9. Pengujian Parameter α dan β

Skenario	Kelompok keahlian	Evaporasi	α	β	Hasil Jarak	Jarak Total
1	ICM	0,7	0,1	0,1	21,409	79,277
	SIDE				24,616	
	TELE				33,252	
2	ICM	0,7	0,3	0,3	21,409	74,026
	SIDE				21,952	
	TELE				30,666	
3	ICM	0,7	0,5	0,5	21,409	65,727
	SIDE				20,551	
	TELE				23,766	
4	ICM	0,7	0,7	0,7	12,291	53,413
	SIDE				17,061	
	TELE				24,061	
5	ICM	0,7	0,9	0,9	13,761	59,901

	SIDE				17,583	
	TELE				28,556	
6	ICM	0,7	0,1	0,9	21,409	73,802
	SIDE				23,558	
	TELE				28,835	
7	ICM	0,7	0,3	0,7	21,409	75,161
	SIDE				21,766	
	TELE				31,987	
8	ICM	0,7	0,5	0,5	15,492	69,394
	SIDE				21,257	
	TELE				32,645	
9	ICM	0,7	0,7	0,3	21,409	71,865
	SIDE				25,349	
	TELE				25,107	
10	ICM	0,7	0,9	0,1	21,409	62,985
	SIDE				17,288	
	TELE				24,288	
11	ICM	0,7	0,9	0,1	13,761	67,738
	SIDE				18,814	
	TELE				35,163	
12	ICM	0,7	0,7	0,3	13,761	56,841
	SIDE				17,061	
	TELE				26,019	
13	ICM	0,7	0,5	0,5	15,492	67,248
	SIDE				24,616	
	TELE				27,140	
14	ICM	0,7	0,3	0,7	21,409	76,378
	SIDE				20,551	
	TELE				34,418	
15	ICM	0,7	0,1	0,9	21,409	72,792
	SIDE				22,028	
	TELE				29,355	

Berdasarkan perhitungan nilai hasil jarak pada tabel 7, parameter dengan $\alpha = 0,1$ dan $\beta = 0,1$ menghasilkan hasil jarak dengan nilai 79,277. Pada pengujian parameter α dan β lainnya, cenderung mengalami penurunan. Sehingga parameter $\alpha = 0,1$ dan $\beta = 0,1$ diambil sbagai paramter untuk pengujian skenario selanjutnya.

4.3.3 Pengujian Skenario 3: Pengujian Konstan

Tujuan dari pengujian ini adalah melihat pergerakan konstan nilai jarak total sebanyak 10 kali dengan menggunakan parameter yang sudah di dapat pada pengujian skenario 1 dan skenario 2. Paramter yang digunakan adalah jumlah semut = 100, jumlah semut terbaik = 1, nilai evaporasi = 0,7, $\alpha = 0,1$ dan $\beta = 0,1$. Hasil pengujian dapat dilihat pada tabel 10.

Tabel 10. Pengujian Konstan

Skenario	Kelompok keahlian	Evaporasi	α	β	Hasil Jarak	Jarak Total
1	ICM	0,7	0,1	0,1	21,409	79,277
	SIDE				24,616	
	TELE				33,252	
2	ICM	0,7	0,1	0,1	21,409	90,086

	SIDE				30,213	
	TELE				38,464	
3	ICM	0,7	0,1	0,1	21,409	93,928
	SIDE				31,699	
	TELE				40,820	
4	ICM	0,7	0,1	0,1	21,409	83,478
	SIDE				26,410	
	TELE				35,660	
5	ICM	0,7	0,1	0,1	21,409	91,197
	SIDE				27,540	
	TELE				42,248	
6	ICM	0,7	0,1	0,1	21,409	92,248
	SIDE				30,946	
	TELE				39,893	
7	ICM	0,7	0,1	0,1	21,409	86,402
	SIDE				28,616	
	TELE				36,378	
8	ICM	0,7	0,1	0,1	21,409	84,532
	SIDE				25,474	
	TELE				37,650	
9	ICM	0,7	0,1	0,1	21,409	86,940
	SIDE				30,213	
	TELE				35,318	
10	ICM	0,7	0,1	0,1	21,409	90,589
	SIDE				26,874	
	TELE				42,306	

Berdasarkan hasil pengujian pada tabel 8, didapat nilai hasil jarak maksimal dari hasil parameter yang ditentukan adalah 93,928. Sedangkan nilai minimal dan rata rata dari hasil pengujian adalah 79,277 dan 87,868. Setelah mendapatkan hasil terbaik, maka didapatkan juga *learning path* hasil optimasi ACO terhadap *rule* yang diperoleh. Hasil optimasi *learning path* dapat dilihat pada tabel 11.

Tabel 11. *Learning Path* Mata Kuliah Pilihan

Nama Kelompok Keahlian	<i>Learning Path</i>
ICM	1. PEMROSESAN BAHASA ALAMI → SISTEM BERBASIS PENGETAHUAN → PENGOLAHAN CITRA DIGITAL 2. PEMROSESAN BAHASA ALAMI → PENGOLAHAN CITRA DIGITAL → SISTEM PENGENALAN

SIDE	1. DATA WARE HOUSE → PENAMBANGAN DATA → PENAMBANGAN TEKS → KONSEP PENGEMBANGAN E-LEARNING 2. KONSEP PENGEMBANGAN E-LEARNING → SISTEM INFORMASI KORPORAT → DESAIN INTERAKSI 3. DESAIN INTERAKSI → E-GOVERNMENT → SISTEM PENDUKUNG PENGAMBILAN KEPUTUSAN 4. DATA WARE HOUSE → PENAMBANGAN DATA → TEKNOLOGI PEMBELAJARAN KREATIF
TELE	1. ANALISIS PERFORMANSI JARINGAN KOMPUTER → PEMULIHAN BENCANA → FORENSIK KOMPUTER DAN JARINGAN 2. JARINGAN SENSOR NIRKABEL → SISTEM TERTANAM UNTUK ROBOTIKA → TOPIK KHUSUS 1 TELEMATIKA 3. FORENSIK KOMPUTER DAN JARINGAN → NETCENTRIC → MIKROKONTROLER 4. TOPIK KHUSUS 1 TELEMATIKA → JARINGAN KOMPUTER LANJUT (INTERNET OF THINGS) → KOMPUTASI AWAN

4.3 Analisis Hasil Pengujian

Berdasarkan hasil pengujian pada 838 data nilai mata tahun ajaran 2015-2016 S1 Teknik Informatika, algoritma *brute force* dapat memberikan rekomendasi berdasarkan bobot tertinggi yang sesuai dengan aturan *prerequisite* atau syarat pengambilan mata kuliah pilihan yang berlaku [1]. Selain dapat memberikan hasil yang sesuai dengan aturan yg berlaku, hasil algoritma *brute force* juga sesuai dengan riwayat nilai mahasiswa. Dengan demikian hasil rekomendasi dapat dijadikan sebagai referensi atau acuan pengambilan mata kuliah pilihan berdasarkan kemampuan setiap mahasiswa. Kekurangan dari algoritma ini adalah waktu eksekusinya. Apabila mendapatkan data set yang besar, eksekusi algoritma ini akan memakan waktu karena melakukan pencarian secara sekuensial hingga selesai.

Berdasarkan hasil pengujian pada 838 data nilai mata kuliah pilihan tahun ajaran 2015-2016 S1 Teknik Informatika, penentuan nilai *minimum support* dan *minimum confidence* yang sesuai [12] dengan karakter data yang dimiliki dapat menghasilkan rekomendasi *learning path* sesuai dengan mata kuliah kelompok keahlian. Hal tersebut dibuktikan oleh hasil algoritma apriori yang mendapatkan 58 *rule*. Dapat dilihat pada tabel 3, pada kelompok ICM terdapat 8 *rule* dengan 2 *rule* lintas kelompok keahlian, pada kelompok SIDE terdapat 14 *rule* dengan 1 *rule* lintas kelompok keahlian, dan pada kelompok TELE terdapat 26 *rule* dengan kriteria 100% *rule* kelompok TELE.

Selain itu, penentuan nilai *minimum support* dan *minimum confidence* pada algoritma Apriori sangat berpengaruh pada hasil akhir *rule* yang terbentuk dikarenakan setiap dataset memiliki karakter yang berbeda beda, dengan pemilihan nilai *minimum support* dan *minimum confidence* yang tepat dapat menghasilkan *rule* yang sesuai dengan kebutuhan rekomendasi.

Berdasarkan hasil 3 skenario pengujian pada *rule* hasil dari algoritma apriori didapatkan hasil jarak terbesar. Data nilai pun tersebar dari nilai 79,277 sampai dengan 92,248 [21], dapat dilihat bahwa hasil pencarian belum sepenuhnya konstan pada suatu titik. Penentuan jumlah semut mempengaruhi proses eksplorasi dan eksploitasi dalam pencarian node terbaik. Apabila semut terlalu banyak dan kandidat semut terbaik terlalu sedikit, bisa menyebabkan pencarian yang belum konstan pada suatu nilai global optimum. Begitu pula konstanta evaporasi, α dan β sangat mempengaruhi perpindahan semut ke node lainnya. Pencarian pada sedikit node yang dilakukan oleh banyak agen bisa bekerja dengan efektif dalam pencarian. Namun, kurang efisien karena melibatkan banyak agen. Parameter parameter tersebut bisa bekerja dengan lebih baik ketika menggunakan data dalam jumlah besar agar algoritma *ant cycle* dapat bekerja lebih efektif dan efisien.

Selain itu, penentuan nilai parameter pada algoritma *ant cycle* sangat berpengaruh pada hasil pencarian dikarenakan setiap *dataset* memiliki karakter berbeda beda, dengan pemilihan parameter pada penelitian dengan masalah serupa bisa dijadikan sebagai referensi parameter. Pemilihan referensi parameter berdasarkan penelitian sebelumnya dapat menghasilkan pencarian yang sesuai dengan kebutuhan rekomendasi.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil penelitian dan analisis yang dilakukan, maka ditarik kesimpulan sebagai berikut:

1. Dengan menggunakan algoritma *brute force* didapatkan 10 mata kuliah pilihan dengan bobot tertinggi berdasarkan riwayat nilai untuk 838 data nilai mahasiswa.
2. Dengan menggunakan nilai *minimum support* 0.01 dan *minimum confidence* 0.1 didapatkan 58 *rule* yang dapat digunakan sebagai rekomendasi pemilihan mata kuliah pilihan, dengan penggunaan nilai tersebut setiap *rule* menggambarkan pola pengambilan mata kuliah tahun 2015 – 2016 sesuai dengan kelompok keahlian, dari 48 *rule* terdapat 3 *rule* yang memiliki rekomendasi lintas kelompok keahlian.
3. Dengan menggunakan nilai parameter banyak semut = 100, $\alpha = 0,1$ dan $\beta = 0,1$ didapatkan jarak perjaanan kelompok keahlian ICM dengan nilai 21,408, jarak perjaanan kelompok keahlian SIDE dengan nilai 30,945, dan jarak perjaanan kelompok keahlian TELE dengan nilai 39,893. Dari nilai hasil jarak dapat digunakan sebagai rekomendasi pengambilan mata kuliah pilihan dalam bentuk *learning path* atau *road map* berdasarkan *rule* yang didapat, untuk mahasiswa merencanakan pengambilan mata kuliah pilihannya.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan, terdapat beberapa pengembangan yang dapat dilakukan untuk penelitian selanjutnya.

1. Melengkapi setiap aturan *prerequisite* agar setiap rekomendasi mata kuliah pilihan bisa sesuai dengan kompetensi mahasiswa.
2. Menambah data nilai mahasiswa agar mendapatkan rekomendasi *rule* dan *learning path* dengan lebih baik.
3. Menggunakan algoritma pencarian yang efektif dan efisien untuk data nilai mahasiswa yang lebih banyak.

Daftar Pustaka

- [1] F. Informatika, "Fakultas Informatika, School of Computing Telkom University," 30 11 2018. [Online]. Available: <https://bif.telkomuniversity.ac.id/pengumuman/buku-kurikulum-2016/struktur-kurikulum-2016/>. [Accessed 30 11 2018].
- [2] Bendakir, N., & Aïmeur, E. (2006, July). Using association rules for course recommendation. In Proceedings of the AAAI Workshop on Educational Data Mining (Vol. 3).
- [3] Katiyar, S., Ibraheem, N., & Ansari, A. Q. (2015). Ant colony optimization: a tutorial review. MR International Journal of Engineering and Technology, 7(2), 35-41.
- [4] Gulzar, Z., Leema, A. A., & Deepak, G. (2018). PCRS: Personalized Course Recommender System Based on Hybrid Approach. Procedia Computer Science, 125, 518-524.
- [5] Gil, A. B., & García-Peñalvo, F. J. (2008). Learner course recommendation in e-learning based on swarm intelligence. J. UCS, 14(16), 2737-2755.
- [6] Sobecki, J., & Tomczak, J. M. (2010, March). Student courses recommendation using ant colony optimization. In Asian Conference on Intelligent Information and Database Systems (pp. 124-133). Springer, Berlin, Heidelberg.
- [7] Sobecki, J. (2014). Comparison of selected swarm intelligence algorithms in student courses recommendation application. International Journal of Software Engineering and Knowledge Engineering, 24(01), 91-109.
- [8] Sohail, S. S., Siddiqui, J., & Ali, R. (2017). Classifications of Recommender Systems: A review. Journal of Engineering Science & Technology Review, 10(4).
- [9] Hipp, J., Güntzer, U., & Nakhaeizadeh, G. (2000). Algorithms for association rule mining—a general survey and comparison. ACM sigkdd explorations newsletter, 2(1), 58-64.
- [10] Suyanto, in Swarm Intelligence: Komputasi Modern untuk Optimasi dan Big Data Mining, Bandung, Informatika, 2017, pp. 20-30.
- [11] Zhao, Q., & Bhowmick, S. S. (2003). Association rule mining: A survey. Nanyang Technological University, Singapore.
- [12] Afifuddin, R. N., & Nurjanah, D. (2019). Sistem Rekomendasi Pemilihan Mata Kuliah Peminatan Menggunakan Algoritma K-means Dan Apriori (studi Kasus: Jurusan S1 Teknik Informatika Fakultas Informatika). eProceedings of Engineering, 6(1).

- [13] Song, C. (2016, March). Research of association rule algorithm based on data mining. In 2016 IEEE International Conference on Big Data Analysis (ICBDA) (pp. 1-4). IEEE.
- [14] Özseyhan, C., Badur, B., & Darcan, O. N. (2012). An association rule-based recommendation engine for an online dating site. *Communications of the IBIMA*, 2012, 1.
- [15] Olmo, J. L., Luna, J. M., Romero, J. R., & Ventura, S. (2013). Mining association rules with single and multi-objective grammar guided ant programming. *Integrated Computer-Aided Engineering*, 20(3), 217-234.
- [16] Stützle, T., & Dorigo, M. (1999). ACO algorithms for the traveling salesman problem. *Evolutionary algorithms in engineering and computer science*, 163-183.
- [17] Asmar, D., Elshamli, A., & Areibi, S. (2005). A comparative assessment of ACO algorithms within a TSP environment. *Dynamics of Continuous Discrete and Impulsive Systems-Series B-Applications & Algorithms*, 1, 462-467.
- [18] Nuutila, E., & Soisalon-Soininen, E. (1994). On finding the strongly connected components in a directed graph. *Information Processing Letters*, 49(1), 9-14.
- [19] Patel, B., Chaudhari, V. K., Karan, R. K., & Rana, Y. K. (2011). Optimization of association rule mining apriori algorithm using ACO. *International Journal of Soft Computing and Engineering*, 1(1), 24-26.
- [20] Trakhtenbrot, B. A. (1984). A survey of Russian approaches to perebor (brute-force searches) algorithms. *Annals of the History of Computing*, 6(4), 384-400.
- [21] Stützle, T., López-Ibáñez, M., Pellegrini, P., Maur, M., De Oca, M. M., Birattari, M., & Dorigo, M. (2011). Parameter adaptation in ant colony optimization. In *Autonomous search* (pp. 191-215). Springer, Berlin, Heidelberg.

Lampiran

1. Tabel data mata kuliah wajib sebagai *prerequisite* mata kuliah pilihan berdasarkan aturan kurikulum yang berlaku.

MK Dasar			MK Wajib ICM		
No	Kode MK	Nama MK	No	Kode MK	Nama MK
1	CCH1D4	Algoritma Struktur Data	1	CSH2G3	Desain Analisis Algoritma
2	CCH1A4	Dasar Algoritma Pemrograman	2	CCH3F3	Kecerdasan Buatan
3	MSH1B3	Logika Matematika	3	CSH483	Pengolahan Citra Digital
4	CSH4C3	Manajemen Proyek Teknologi Informasi	4	CSH3L3	Pembelajaran Mesin
5	MSH2A3	Matematika Diskrit	5	CSH2B3	Teori Bahasa Automata
6	MUH1G3	Matriks dan Ruang Vektor			
7	CSH3H2	Pemodelan dan Simulasi			
8	SEH2A3	Pemrograman Berorientasi Objek			
9	MUH1F3	Probabilitas dan Statistika			
MK Wajib SIDE			MK Wajib TELE		
No	Kode MK	Nama MK	No	Kode MK	Nama MK
1	CSH2E4	Analisis Perancangan Perangkat Lunak	1	CSH3A4	Jaringan Komputer
2	CSH3I3	Sistem Informasi	2	CSH4P3	Jaringan Komputer Lanjut
3	CSH3B3	Interaksi Manusia Komputer	3	CSH2F3	Sistem Digital
4	CSH3E3	Implementasi dan Pengujian Perangkat Lunak	4	CSH2I3	Organisasi dan Arsitektur Komputer
5	CSH2C3	Pemodelan Basis Data	5	CSH3A3	Sistem Operasi
6	CSH2D3	Sistem Basis Data	6	CSH3J3	Sistem Terdistribusi dan Paralel

2. Tabel data berisi mata kuliah pilihan berdasarkan kelompok keahlian

ICM		SIDE		TELE	
Kode Mata Kuliah	Nama Mata Kuliah	Kode Mata Kuliah	Nama Mata Kuliah	Kode Mata Kuliah	Nama Mata Kuliah
CSH4103	Metode Formal	CSH4113	Sistem Pendukung Pengambilan Keputusan	CSH4183	Sistem Nirkabel
CSH4313	Kecerdasan Kolektif	CSH4123	Teknologi Pembelajaran Kreatif	CSH4203	Software Defined Networks
CSH483	Pengolahan Citra Digital	CSH413	Topik Khusus 2 SIDE	CSH423	Topik Khusus 1 Telematika (IOT for Healthcare)
CSH493	Sistem Berbasis Pengetahuan	CSH4133	Sistem Informasi Korporat	CSH4233	Forensik Komputer Dan Jaringan
CSH4F3	Kriptografi	CSH4143	Topik Khusus 3 SIDE - Mesin	CSH4243	Rekayasa Protokol
CSH4I3	Sistem Pengenalan	CSH4153	Penjaminan Mutu Perangkat Lunak	CSH4253	Mikrokontroler
CSH4M3	Komputasi Finansial	CSH4163	Desain Interaksi	CSH4283	Netcentric
CSH4N3	Komputasi Lunak	CSH4213	Basis Data Non Relasional	CSH433	Topik Khusus 2 Telematika
CSH4O3	Pemrosesan Bahasa Alami	CSH4223	Audit Sistem Informasi	CSH4333	Kompresi Data
CSH4R3	Convolutional Neural Network	CSH4273	Perencanaan Strategis Sistem Informasi	CSH4343	Analisis Performansi Jaringan Komputer
CSH4U3	Komputasi Kinerja Tinggi	CSH4353	Data Warehouse	CSH443	Sistem Waktu Nyata

CSH4X3	Topik Khusus 1 ICM (mobile programming)	CSH4363	E-Government	CSH463	Jaringan Sensor Nirkabel
CSH4Y3	Topik Khusus 2 ICM - Pemrograman Logika	CSH453	Basis Data Spasial	CSH4D3	Sistem Keamanan
CSH4K3	Komputasi Evolusioner	CSH4G3	Penambangan Data	CSH4P3	Jaringan Komputer Lanjut (IOT)
CSH4173	Aplikasi Intelegensia Buatan Untuk Edukasi	CSH4H3	Penambangan Teks	CSH4Q3	Sistem Operasi Lanjut
CSH4293	Sistem Berbasis Agen	CSH4T3	Arsitektur Berorientasi Layanan	CSH4S3	Komputasi Awan
CSH4J3	Model Graf Probabilistik	CSH4Z3	Topik Khusus 1 SIDE	CSH473	Pemulihan Bencana
		CSH4303	Konsep Pengembangan E-Learning	CSH4C3	Sistem Tertanam Untuk Robotika