

## BAB III METODE PENELITIAN

### 3.1 Tahapan Penelitian

#### 3.1.1 Pengumpulan Data

Metode pengumpulan data yang dilakukan penelitian ini:

1. Wawancara

Wawancara dilakukan secara langsung terhadap pihak kampus STMIK Dharmawacana untuk mendapatkan informasi dan data-data yang dibutuhkan dalam memprediksi mahasiswa yang berpotensi lulus cepat atau tidak.

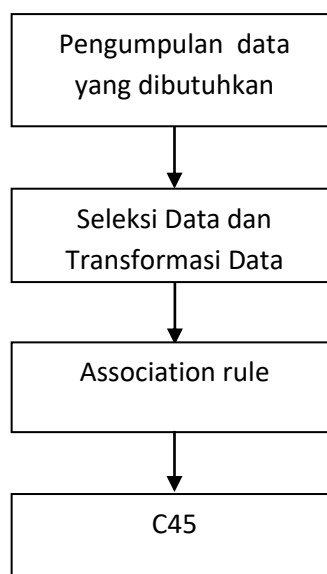
2. Observasi

Mengadakan pengamatan dan pengumpulan data history berupa data internal dan external yang telah tersusun dalam database maupun kelengkapan data yang dibutuhkan.

3. Studi Dokumentasi

Mengumpulkan data dari dokumen-dokumen yang ada dari jurusan Teknik Informatika sebagai bahan untuk melanjutkan proses penelitian terkait aktivitas akademik mahasiswa.

Tahapan penelitian dalam studi ini ditunjukkan pada gambar 3.1



**Gambar 3.1.1 Diagram Alir Penelitian**

### 3.1.2 Jenis Data

Jenis data yang digunakan pada penelitian ini adalah:

1. Data Primer

Data primer merupakan data yang didapatkan melalui observasi dan wawancara langsung dengan Ketua Program Studi Teknik Informatika STMIK Daharmawacana Metro.

2. Data Sekunder

Data sekunder merupakan sumber data yang diperoleh dari media perantara atau secara tidak langsung yaitu berupa buku dan jurnal. Untuk mencari data sekunder, peneliti melakukan pencarian di internet dan melakukan kunjungan ke perpustakaan.

Setelah melakukan observasi dan pencarian data baik primer maupun sekunder, didapatkn beberapa kategory yang nantinya dijadikan acuan dan tolak ukur dalam pengitungan klasifikasi mahasiswa. Beberapa kategory ini yang nantinya menentukan potensi akademik mahasiswa kan cepat lulus, tepat waktu, atau lulus terlambat. kategori-kategori itu yakni masa study, besaran IPK, jumlah sks, toefl, pendapatan orang tua dan hardskill .Berikut dibawah ini dilampirkan data rinci dari 6 kategori tersebut :

**Tabel 3.1 Kategori masa study yang digunakan**

Kategori	Masa Studi
Lulus Cepat	$\leq 6$ semester
Tepat waktu	$7 \leq X \leq 8$
Terlambat	$\geq 9$ semester

**Tabel 3.2 Kategori IPK yang digunakan**

Kategori	IPK
Sangat Baik (potensi lulus)	$> 3.0 X \leq 3.5$
Baik	$> 2,75 X \leq 2.99$
Buruk	$< 2.74$

**Tabel 3.3 Kategori SKS yang digunakan**

Kategori	Jumlah SKS	Keterangan
Tercukupi	144	3
Tidak Cukup	$<143 \text{ X } \geq 120$	2
Kurang	$< 120$	1

**Tabel 3.1.2.4 Toefl**

Kategori	Rangking Toefl
5	600
4	550 – 500
3	500 – 450
2	450 – 350
1	$< 350$

**Tabel 3.1.2.5 Pendapatan Orang Tua**

Kategori	Pendapatan Orang Tua
5	$> \text{Rp. } 10.000.000$
4	$> \text{Rp. } 7.000.000 - 9.000.000$
3	$> \text{Rp. } 4.000.000 - 6.000.000$
2	$> \text{Rp. } 1.000.000 - 3.000.000$
1	$< \text{Rp. } 1.000.000$

**Tabel 3.1.2.6 Hard Skill**

Kategori	Hard Skill
3	A
2	B
1	C

### 3.1.3 Analisa Data (Association Rules)

Pada penelitian ini dilakukan analisa data terhadap data-data yang diperoleh pada tahap pengumpulan data. Penelitian ini menggunakan data kelulusan tahun 2017-2018 dengan sampel 100 data mahasiswa. Adapun atribut yang digunakan yaitu masa studi, IPK dan jumlah sks yang telah diambil. Dalam algoritma apriori data yang telah diperoleh dikelompokkan kemudian dilakukan penggabungan *itemset* dan perhitungan nilai. Algoritma association rule dibagi menjadi beberapa tahap atau yang biasa disebut iterasi. Tiap iterasi menghasilkan pola frekuensi tinggi dengan panjang yang sama dimulai dari pass pertama yang menghasilkan pola frekuensi tinggi dengan panjang satu. Pada iterasi pertama ini, *support* dari setiap item dihitung dengan melihat data yang ada. Setelah *support* dari setiap item di dapat, item yang memiliki *support* diatas minimum *support* dipilih sebagai pola frekuensi tinggi dengan panjang 1 atau sering disingkat persamaan 3.1 - itemset. Iterasi kedua menghasilkan 3.2 -itemset yang tiap set-nya memiliki dua item. Pertama dibuat kandidat 2 - itemset dari kombinasi semua 1 - itemset. Sedangkan untuk tiap kandidat 2-itemset ini dihitung *support*nya dengan ,elihat data acuan kembali. *Support* disini artinya jumlah transaksi dalam data yang mengandung kedua item dalam kandidat 2-itemset. Setelah *support* dari semua kandidat 2-itemset didapatkan, kandidat 2-itemset yang memenuhi syarat minimum support dapat ditetapkan sebagai 2-itemset yang juga merupakan pola frekuensi tinggi dengan panjang 2 Persamaan yang digunakan untuk menghitung nilai presentase *support* 1 *itemset* adalah sebagai beriku pada persamaan 3.1 :

$$\text{Support (A)} = \left( \frac{\text{Jumlah transaksi mengandung A}}{\text{Jumlah transaksi A}} \right) * 100\% \dots\dots\dots(3.1)$$

Persamaan kedua yang digunakan untuk menghitung nilai presentase *support* 2 *itemset* adalah sebagai berikut dibawah ini persamaan 3.2 :

$$\text{Support (A,B)} = \left( \frac{\text{Jumlah transaksi mengandung A \& B}}{\text{Jumlah transaksi A}} \right) * 100\% \dots\dots\dots(3. 2)$$

Persamaan (3) yang digunakan untuk menghitung nilai presentase *confidence* dari sebuah kombinasi *itemset* adalah dibawah ini persamaan 3.3 sebagai berikut :

$$\text{Confidence P (B | A)} = \frac{(\text{Jumlah Transaksi mengandung A \& B}) * 100\%}{\text{Jumlah transaksi A}} \dots\dots(3.3)$$

### 3.1.4 Pengujian

Untuk mengukur kekuatan asosiasi ini digunakan ukuran *confidence*. Selain *support* ada ukuran lain yang mengukur ketidakpastian aturan “if-then”. Ukuran tersebut adalah *confidence* dari suatu aturan. *Confidence* adalah rasio jumlah transaksi yang meliputi semua item dalam antecedent dan consequent dengan jumlah transaksi yang meliputi semua item. Selain itu nilai *confidence* juga terpengaruh oleh nilai *support* yang sudah dihitung terlebih dahulu.

### 3.1.5 Proses Hitung C45

Pada akhir tahun 1970 sampai awal tahun 1980 J. Ross Quinlan, seorang peneliti di bidang machine learning, membuat sebuah algoritma decision tree yang dikenal dengan ID3 (Iterative Dichotomiser). Quinlan kemudian membuat algoritma C4.5 (sering disebut dengan pohon keputusan) yang merupakan pengembangan dari algoritma ID3 [10]. Algoritma ini memiliki kelebihan, yaitu mudah dimengerti, fleksibel, dan menarik karena dapat dalam bentuk gambar (pohon keputusan). Algoritma C4.5 merupakan struktur pohon dimana terdapat simpul yang mendeskripsikan atribut-atribut, setiap cabang menggambarkan hasil dari atribut yang diuji, dan setiap daun menggambarkan kelas. Algoritma C4.5 secara rekursif mengunjungi setiap simpul keputusan, memilih pembagian yang optimal, sampai tidak bisa dibagi lagi. Algoritma C4.5 menggunakan konsep information gain atau entropy reduction untuk memilih pembagian yang optimal. Pohon keputusan mirip sebuah struktur pohon dimana terdapat node internal (bukan daun) yang mendeskripsikan atribut-atribut, setiap cabang menggambarkan hasil dari atribut yang diuji, dan setiap daun menggambarkan kelas. Pohon keputusan bekerja mulai dari akar paling atas, jika diberikan sejumlah data uji, misalnya X

dimana kelas dari data X belum diketahui, maka pohon keputusan akan menelusuri mulai dari akar sampai node dan setiap nilai dari atribut sesuai data X diuji apakah sesuai dengan aturan pohon keputusan, kemudian pohon keputusan akan memprediksi kelas dari tupel X. Contoh Konsep Pohon Keputusan untuk memprediksi kelas dari tupel X. Ada beberapa tahap dalam membuat sebuah pohon keputusan dengan algoritma C4.5 yaitu :

1. Menyiapkan data training. Data training biasanya diambil dari data histori yang pernah terjadi sebelumnya dan sudah dikelompokkan ke dalam kelas-kelas tertentu.
2. Menentukan akar dari pohon. Akar akan diambil dari atribut yang terpilih, dengan cara menghitung nilai gain dari masing-masing atribut, nilai gain yang paling tinggi yang akan menjadi akar pertama. Sebelum menghitung nilai gain dari atribut, hitung dahulu nilai entropy. Untuk menghitung nilai entropy digunakan rumus:

$$Entropi(S) = \sum_{k=1}^n -p_k \log_2 p_k \dots\dots\dots (1)$$

Keterangan: S = himpunan kasus k = jumlah partisi S p<sub>k</sub> = probabilitas yang didapat dari jumlah (ya/tidak) dibagi total kasus. Kemudian hitung nilai gain menggunakan rumus:

$$Gain(S, A) = Entropi(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * Entropi(S_i) \dots\dots\dots (2)$$

Keterangan: S = himpunan kasus A = fitur n = jumlah partisi atribut A  $\frac{|S_i|}{|S|}$  = proporsi S<sub>i</sub> terhadap S  $|S_i|$  = jumlah kasus dalam S<sub>i</sub>. Ulangi langkah ke-2 hingga semua record terpartisi.

Populasi penelitian ini adalah data kelulusan mahasiswa yang diberikan perguruan tinggi maupun data yang berhubungan dengan kelulusan mahasiswa pada jurusan Teknik Informatika, dan melihat kondisi mahasiswa selama mengikuti masa studi diperguruan tinggi. Pada penelitian ini, algoritma C4.5 dalam membangun sebuah pohon keputusan adalah sebagai berikut:

1. Hitung jumlah data, jumlah data berdasarkan anggota atribut hasil dengan syarat tertentu. Untuk proses pertama syaratnya masih kosong.

2. Pilih atribut sebagai Node.
3. Buat cabang untuk tiap-tiap anggota dari Node.
4. Periksa apakah nilai entropy dari anggota Node ada yang bernilai nol. Jika ada, tentukan daun yang terbentuk. Jika seluruh nilai entropy anggota Node adalah nol, maka proses pun berhenti.
5. Jika ada anggota Node yang memiliki nilai entropy lebih besar dari nol, ulangi lagi proses dari awal dengan Node sebagai syarat sampai semua anggota dari Node bernilai nol. Node adalah atribut yang mempunyai nilai gain tertinggi dari atribut-atribut yang ada. Untuk menghitung nilai gain suatu atribut