BAB 2
TINJAUAN PUSTAKA

Bab ini menjelaskan teori-teori yang menjadi dasar dalam melakukan penelitian, yang bersumber dari buku, jurnal ataupun artikel. Tinjauan pustaka berfungsi sebagai dasar dalam penelitian yang berisikan penjelasan teori-teori terkait dengan penelitian yaitu *data mining*, pohon keputusan (*decision tree)* dan *Knowledge Discovery in Database* (KDD) yang bersumber dari buku, jurnal dan skripsi.

2.1 Kelulusan Mahasiswa ITK

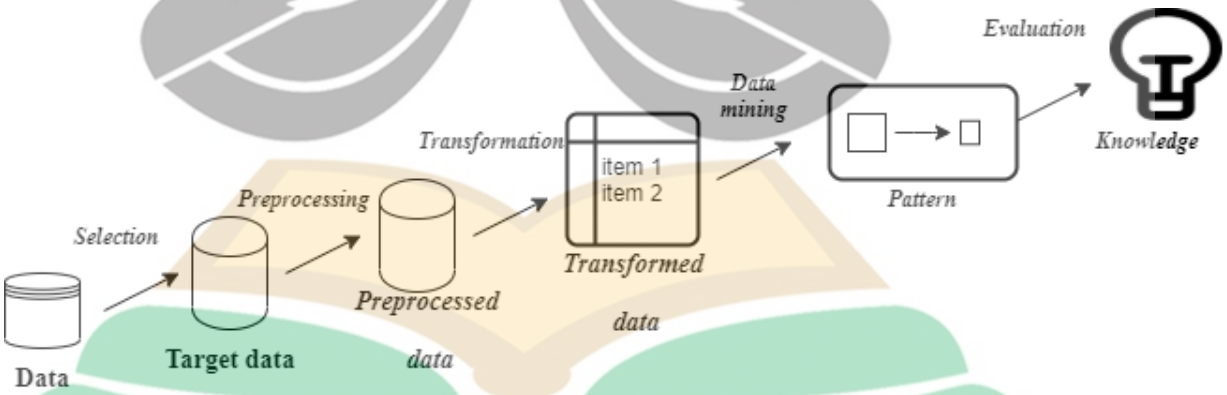
Kelulusan pada mahasiswa merupakan tanda berakhirnya mahasiswa dalam menyelesaikan pendidikan sarjana. Mahasiswa memiliki tingkat kelulusan yang berbeda yaitu lulus tepat waktu dan tidak tepat waktu. Mahasiswa dapat dikatakan lulus apabila telah memenuhi persyaratan pada setiap program studi di perguruan tinggi. Salah satu syarat kelulusan mahasiswa adalah dengan mengerjakan Tugas Akhir (TA). Tugas Akhir (TA) adalah sebuah mata kuliah yang harus ditempuh oleh seorang mahasiswa menjelang akhir studinya (Maulani, Simbolon, & Amirullah, 2019).

Pada Kementerian Pendidikan dan Kebudayaan Direktorat Jenderal Pendidikan Tinggi tentang Sistem Pendidikan Tinggi Institut Teknologi Kalimantan disebutkan bahwa untuk memenuhi standar kelulusan bagi mahasiswa program sarjana (S1) harus menempuh paling sedikit 144-160 satuan kredit semester (sks) dengan masa studi selama 8-12 semester atau 4-6 tahun. Kelulusan mahasiswa merupakan hal penting yang harus diperhatikan, karena penurunan jumlah kelulusan pada perguruan tinggi akan memberikan dampak negative kepada jumlah kelulusan dan akan berpengaruh pada penilaian pemerintah dalam bentuk status akreditasi pada perguruan tinggi (Himawan, 2014).

2.2 Data mining

Data mining adalah sebuah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar (Kusrini, 2019). *Data mining* merupakan proses menggali dan menganalisis sejumlah data yang sangat besar dan banyak untuk memperoleh hal-hal baru, benar, bermanfaat, hingga akhirnya dapat menemukan corak atau pola tertentu dalam data tersebut.

Analisis data dengan *data mining* menggunakan alat atau *tool* untuk menemukan pola dan aturan dalam himpunan data. Alat tersebut berupa perangkat lunak yang bertugas dalam mengidentifikasi aturan dan fitur pada data. Alat *data mining* diharapkan dapat mengenali pola dalam data dengan input minimal dari pengguna (Marcos & Hidayah, 2014).. KDD merupakan proses dalam menganalisis data-data untuk menemukan informasi dan pengetahuan yang bermanfaat dan dapat digunakan untuk melakukan pengambilan keputusan. KDD bersifat implisit, potensial dari data yang berukuran besar tidak diketahui sebelumnya. Langkah dalam melakukan proses KDD digambarkan pada gambar 2.1. (Nwagu, Omankwu, & Inyiama, 2017).



Gambar 2.1 Tahapan KDD (Nwagu, Omankwu, & Inyiama, 2017)

Berikut merupakan penjelasan terhadap tahapan KDD yang berada pada Gambar 2.1

1. *Data selection*

Data seleksi (*data selection)* merupakan proses pemilihan atau seleksi data dari sekumpulan data yang harus dilakukan sebelum tahap lain dilaksanakan. Tahap ini melibatkan pemilihan atribut yang diperlukan untuk proses data mining.

2. *Pre-processing*

Tahap pembersihan data (*data cleaning*) data akan dibersihkan dengan menemukan *missing value* dan menghilangkan *redundant data.* Dalam memperbaiki *missing value* pada data dilakukan dengan cara mencari rata-rata pada nilai yang ada pada atribut tersebut. Dalam proses memperbaiki *redundant data* dapat dilakukan dengan menghapus nilai ganda pada sebuah atribut.

3. *Transformation*

Tahap ini dapat dilakukan dengan cara yang beragam. Transformasi data bergantung terhadap kebutuhan bentuk data dari algoritma *data* yang dipilih..

4. *Data mining*

Pada tahap ini dilakukan pembangunan model *data mining* yang disesuaikan dengan tujuan yang ingin dicapai dari proses KDD. Pada tahap ini juga dilakukan pemilihan algoritma yang sesuai terlebih dahulu sebelum proses pembangunan pola dilakukan.

5. *Interpretation / Evaluation*

Langkah terakhir adalah interpretasi dan dokumentasi hasil dari langkah sebelumnya. Teknik interpretative yang umum digunakan adalah visualisasi dari pola yang diekstrak.

Metode split data yang terdapat dalam *data mining* terdapat 3 teknik *sampling* yang dapat digunakan dalam membangung model *data mining* diantaranya *linier sampling, stratified sampling, shuffle sampling* (Rapid, 2018).

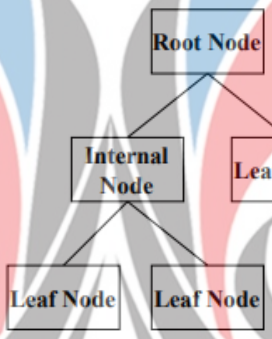
2.3 Pohon keputusan (*Decision tree)*

Pohon keputusan (*Decision tree*) merupakan salah satu metode yang populer digunakan dalam klasifikasi. Disebut pohon keputusan karena model yang dihasilkan untuk memprediksi sebuah data yaitu berupa pohon. Untuk melakukan klasifikasi data, setiap *attribute* dari data tersebut akan diuji melalui serangkaian node yang berada pada pohon keputusan dan setelah data sampai pada leaf node,

data tersebut akan terklasifikasi sesuai dengan kelas yang terdapat pada leaf node (Iskandar, Hiryanto, & Hendryli, 2018).

Gambar 2.2 Merupakan konsep dasar dalam pembuatan pohon keputusan yang terdapat 3 jenis *node, node* tersebut adalah:

1. *Root Node* merupakan node yang letaknya berada diawal *tree*. *Root Node* tidak memiliki *input* yang berarti tidak ada cabang yang masuk ke *node* ini. *Root Node* dapat memiliki *output* lebih dari satu atau tidak memiliki *output* sama sekali.
2. *Internal Node* merupakan *node* percabangan, pada *node* ini hanya terdapat satu *input* (satu cabang masuk) dan dapat memiliki *output* satu atau lebih.
3. *Leaf Node* atau terminal *node*, merupakan *node* akhir pada *decision tree*. *Node* ini memiliki satu *input* dan tidak memiliki *output*. *Node* berperan untuk menunjukan kelas akhir dari pengklasifikasian.



Gambar 2.2 Konsep dasar pohon keputusan

2.3.1 Algoritma C4.5

Metode pohon keputusan atau *decision tree* ini sangat populer karena mampu melakukan klasifikasi sekaligus menunjukkan hubungan antar atribut. Banyak algoritma yang digunakan untuk membangun suatu Decision tree, salah satunya ialah algoritma C4.5. Algoritma C4.5 menggunakan rasio perolehan (gain ratio) (Kurniawan, 2019).

Algoritma C4.5 merupakan algoritma pengembangan dari algoritma ID3 yang digunakan untuk membentuk pohon keputusan. Pohon keputusan dapat membagi kumpulan data yang besar menjadi himpunan-himpunan record yang lebih kecil dengan menerapkan serangkaian aturan keputusan.

Secara umum, langkah untuk membangun *decision tree* pada algoritma C4.5 adalah sebagai berikut (Iskandar, Hiryanto, & Hendryli, 2018):

1. Menyiapkan *training data*. *Training data* yaitu data yang akan dilatih untuk membuat sebuah prediksi atau menjalankan fungsi dari sebuah algoritma. Data *training* 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* Membagi atribut sebagai internal *node* pada setiap cabang
3. Kemudian hitung nilai *Gain, split info* dan *gain ratio*
4. Ulangi proses 2 dan 3 hingga setiap cabang berakhir pada *leaf node*
5. Proses partisi pohon keputusan akan berhenti saat:
   a. Semua *record* dalam simpul N mendapat kelas yang sama.
   b. Tidak ada atribut atau variabel didalam *record* yang dipartisi lagi.
   c. Tidak ada *record* didalam cabang yang kosong.

Untuk membangun pohon keputusan dibutuhkan nilai *entropy, information gain, split gain* dan *gain ratio*

1. *Entropy*

Entropy digunakan dalam menghitung kemiripan data pada *dataset training.* Untuk melakukan perhitungan nilai *entropy* dapat dilihat pada persamaan 2.1.

$$Entropy(S) = \sum_{i=1}^{n} -Pi\ log2(Pi) \qquad 2.1$$

Keterangan:

S = *dataset training*

n = jumlah kelas dalam S

Pi = perbandingan jumlah data pada masing masing kelas dengan total data yang terdapat dalam S

2. *Information Gain*

Information Gain digunakan dalam menentukan berapa banyak informasi yang dapat diberikan oleh *attribute* terhadap kelas yang ada. Untuk melakukan perhitungan nilai *information gain* dapat dilihat pada persamaan 2.2.

$$Gain(A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} X\ Entropy(S_i) \qquad 2.2$$

Keterangan:

A = *attribute*

S = *dataset training*

n = jumlah partisi pada *attribute*

Si = Partisi ke-i pada *attribute*

|S| = Jumlah data pada *attribute*

|Si| = Jumlah data pada partisi ke-i *attribute*

Untuk menentukan *root node*, nilai $Entropy(S)$ yang digunakan adalah *entropy* dari keseluruhan data. Pada pengulangan selanjutnya pada proses untuk menentukan *node* hasil dari *root node*, nilai $Entropy(S)$ yang digunakan adalah nilai *entropy* dari *attribute* yang menjadi *root node*. Sehingga dapat dikatakan nilai $Entropy(S)$ yang digunakan untuk mencari *Information Gain* sebuah *attribute* adalah *entropy* dari *node* sebelumnya.

3. *Split* info

Split Info digunakan dalam menghitung kemungkinan informasi yang dihasilkan dari pembagian. Semakin seragam pembagian nilai dari sebuah *attribute* nilai *split info* semakin besar. Untuk melakukan perhitungan nilai *split info* dapat dilihat pada persamaan 2.3.

$$Split(A) = -\sum_{i=1}^{n} \frac{|S_i|}{|S|} X\ log2\ \frac{|S_i|}{|S|} \qquad 2.3$$

Keterangan:

A = attribute

n = jumlah partisi pada *attribute*

Si = Partisi ke-i pada *attibute*

|S| = Jumlah data pada *attribute*

|Si| = Jumlah data pada partisi ke-i *attribute*

4. *Gain Ratio*

Gain Ratio digunakan untuk mengurangi bias/kesalahan dari *information gain*. Untuk melakukan perhitungan *gain ratio* dapat dilihat pada persamaan 2.4.

$$GainRatio(A) = \frac{Gain(A)}{Split(A)} \qquad 2.4$$

Keterangan:

A = *attribute*

Gain(A) = nilai *information gain* pada *attribute S*

Split(A) = nilai *Split information* pada *attribute S*

Berikut beberapa kelebihan dari *decision tree* dengan algoritma Klasifikasi C4.5, antara lain:

1. Hasil analisa berupa diagram pohon yang mudah dimengerti.
2. Mudah untuk dibangun, serta membutuhkan data percobaan yang lebih sedikit dibandingkan algoritma klasifikasi lainnya.
3. Mampu mengolah data nominal dan *continue*.
4. Model yang dihasilkan dapat dengan mudah dimengerti.
5. Menggunakan teknik statistik sehingga dapat divalidasikan.
6. Waktu komputasi relatif lebih cepat dibandingkan teknik klasifikasi lainnya.
7. Akurasi yang dihasilkan mampu menandingi teknik klasifikasi lainnya

(Turnip & Wijaya, 2016)

2.4 Evaluasi model

Evaluasi model adalah alat ukur berbentuk *matrix* 2x2 yang dapat digunakan untuk mendapatkan jumlah ketepatan klasifikasi *dataset* terhadap kelas lulus tepat waktu dan lulus tidak tepat pada algoritma yang dipakai tiap kelas yang diprediksi memiliki empat kemungkinan keluaran yang berbeda, yaitu *true positive* (TP) dan *true negatives* (TN) yang menunjukkan ketepatan klasifikasi. Jika prediksi keluaran bernilai positif sedangkan nilai aslinya adalah negatif maka dapat disebut dengan *false positive* (FP) dan jika prediksi keluaran bernilai negatif sedangkan nilai aslinya adalah positif maka dapat disebut dengan *false negative* (FN) (Priati, 2016).

Tabel menyajikan bentuk *confusion matrix* seperti yangtelah dijelaskan sebelumnya.

Tabel 2.1 Evaluasi model untuk Klasifikasi Kelas

		Predicated Class	
Actual Class	*yes*	*True Positive* (TP)	*False Negative* (FN)
	no	*False Positive* (FP)	*True Negative* (TN)

Untuk melakukan perhitungan akurasi yang didapatkan dengan menggunakan rumus dari persamaan 2.5

$$\text{Akurasi} = \frac{TP + TN}{\text{Jumlah Data}} \qquad 2.5$$

Untuk melakukan perhitungan tingkat kesalahan (*error rate*) yang didapatkan dengan menggunakan rumus dari persamaan 2.6

$$\text{kesalahan} = \frac{FP + FN}{\text{Jumlah Data}} \qquad 2.6$$

2.5 Dataset

Dataset Himpunan data (*dataset*) merupakan kumpulan dari objek dan atributnya. Dalam *dataset*, jenis data dapat dibagi menjadi 2 bagian yaitu *data training* dan *data testing*. *Data training* adalah data yang digunakan untuk menentukan pola klasifikasi yang dengan melakukan perhitungan *gain ratio. Data testing* merupakan data yang akan atau sedang terjadi dan dipergunakan sebagai bahan uji yang sebelumnya sudah didapatkan pada *data training*. Data ini digunakan untuk mengukur sejauh mana klasifikasi berhasil melakukan klasifikasi dengan benar. (Anggraini, Widagdo, & Arief , 2019).

Data training dan testing memiliki perbandingan persentase yang lebih besar dari data testing. Data training memiliki persentase sebesar 70% - 90% dan data testing memiliki persentase sebesar 10% - 30% (Ardiansyah, Majid, & Zain, 2016) Sehingga pada umumnya jika data memiliki skala yang besar maka perbandingannya adalah 80% : 20%. Dengan menggunakan linear sampling. Linier

sampling adalah suatu metode dalam membagi data yang tidak mengubah urutan datanya (Rapid, 2018).

2.6 Penelitian Terdahulu

Dalam penyusunan tugas akhir ini mengacu dari penelitian sebelumnya terkait *data mining* menggunakan *decision tree*. Tabel 2.2 menunjukkan rangkuman hasil penelitian terdahulu.

Tabel 2.2 Penelitian terdahulu

No	Nama dan Tahun	Data	Model	Evaluation Model	Hasil
1	Fahri, 2019	Nilai IPK 1-6	Decision tree c4.5, ID3, KNN	Confusion Matrix	Metode decision tree C4.5 mendapatkan nilai akurasi yang optimal
2	Romadhona dkk, 2017	Data usia, jenis kelamin, dan indeks prestasi mahasiswa	Decision tree c4.5, Neural network	x-validasi	Metode decision tree C4.5 mendapatkan nilai akurasi yang tertinggi
3	Fiastantyo, 2009	Data NIM, program studi, jenis kelamin, umur, status, indeks prestasi mahasiswa dan label	C4.5 Naïve Bayes	Confusion Matrix	C4.5 lebih baik dalam nilai *recall* dan akurasi

Pada tabel 2.2 terdapat tiga penelitian terkait prediksi kelulusan mahasiswa tepat waktu menggunakan metode klasifikasi. Ketiga penelitian tersebut digunakan untuk memperoleh hasil terkait masing masing algoritma.

Fahri (2019) melakukan penelitian prediksi tingkat kelulusan mahasiswa menggunakan metode data mining dengan perbandingan tiga algoritma yaitu *decision tree C4.5*, *ID3*, dan *KNN*. Dengan menggunakan data berdasarkan nilai IPK 1 hingga 6. Dalam penelitian tersebut membuktikan bahwa algoritma C4.5 memberikan hasil akurasi yang lebih baik daripada algoritma ID3 dan KNN dalam melakukan prediksi kelulusan. Perbandingan akurasi yang didapatkan adalah

92.20% : 90.80% : 83.90 % dengan menggunakan 1000 data yang telah dibagi menjadi data training 80% dan data testing 20% dengan menggunakan metode evaluasi cunfosion matrix. Sehingga mendapatkan faktor-faktor yang paling mempengaruhi tingkat siswa adalah IPK Semester 2

Romadhona dkk (2017) melakukan penelitian prediksi kelulusan mahasiswa tapat waktu menggunakan algoritma *decision tree* yaitu algoritma C4.5, ID3 dan Chaid. Dengan menggunakan data usia, jenis kelamin dan indeks prestasi mahasiswa. Dalam penelitian ini algoritma C4.5 memiliki kompleksitas yang lebih baik dari ID3 dan Chaid, karena setiap nilai dalam suatu atribut diproses untuk mendapatkan entropy dan mendapatkan information gain. Yang akan menghasilkan pohon keputusan.

Fiastantyo (2019) melakukan penelitian perbandingan kinerja metode klasifikasi data mining menggunakan naïve bayes dan algoritma C4.5 Untuk prediksi ketepatan waktu kelulusan mahasiswa menggunakan dua algoritma dalam yaitu *decision tree C4.5,* dan *Naïve Bayes.* Dengan menggunakan data berdasarkan Data NIM, program studi, jenis kelamin, umur, status, IPS 1-4 dan label (lulus tepat waktu dan lulus tidak tepat waktu atau terlambat). Dalam penelitian tersebut membuktikan bahwa algoritma C4.5 memberikan hasil akurasi yang lebih baik daripada algoritma naïve bayes dalam melakukan prediksi kelulusan. Perbandingan akurasi yang didapatkan adalah 77.534% : 74.094% dengan menggunakan 1919 dari angkatam 2008 dan 2009.

Setelah melakukan peninjauan terhadap tiga penelitian tersebut, dalam penelitian ini akan dilakukan prediksi kelulusan mahasiswa tepat waktu pada Institut Teknologi Kalimantan. Predisi ini akan memanfaatkan teknik *data mining* dengan algoritma *decision tree C4.5*. Hasil dari prediksi diharapkan dapat dimanfaatkan sebagai bahan acuan untuk standard bagi mahasiswa untuk lulus tepat waktu, sehingga mampu menyelesaikan permasalahan yang dihadapi oleh perguruan tinggi.