

BAB 2

LANDASAN TEORI

2.1 Prediksi

Pengertian prediksi adalah sama dengan ramalan atau perkiraan. Menurut kamus besar bahasa Indonesia, prediksi adalah hasil dari kegiatan memprediksi atau meramal atau memperkirakan nilai pada masa yang akan datang dengan menggunakan data masa lalu. Prediksi menunjukkan apa yang akan terjadi pada suatu keadaan tertentu dan merupakan input bagi proses perencanaan dan pengambilan keputusan. Prediksi bisa berdasarkan metode ilmiah ataupun subjektif belaka. Ambil contoh, prediksi cuaca selalu berdasarkan data dan informasi terbaru yang didasarkan pengamatan termasuk oleh satelit. Begitupun prediksi gempa, gunung meletus ataupun bencana secara umum. Namun, prediksi seperti pertandingan sepak bola, olah raga, dan lain-lain umumnya berdasarkan pandangan subektif dengan sudut pandang sendiri yang memprediksinya.

2.2 Prestasi Akademik

Prestasi akademik mahasiswa bisa diukur dengan nilai Indeks Prestasi (IP) yang didapat oleh mahasiswa. Indeks Prestasi sendiri adalah tingkat keberhasilan mahasiswa dalam satu semester. Sedangkan tingkat keberhasilan mahasiswa sejak semester pertama sampai dengan suatu semester tertentu dinyatakan dengan Indeks Prestasi Kumuatif. Perhitungan IPK menggunakan rumus seperti IP, tetapi dengan K adalah besarnya seluruh SKS mata kuliah yang telah ditempuh dan N adalah seluruh nilai matakuliah yang diperoleh. Berikut rumus perhitungannya:

$$IP = \frac{\sum_{i=0}^n N_i * K_i}{\sum_{i=1}^n K_i} \quad (2.1)$$

dimana N_i = nilai angka hasil evaluasi masing-masing mata kuliah.

K_i = Besar SKS masing-masing mata kuliah

n = Jumlah mata kuliah yang telah diambil.

2.3 Data Mining

Data mining adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam database. Data mining adalah 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 (Turban, dkk. 2005).

Data mining merupakan bidang dari beberapa bidang keilmuan yang menyatukan teknik dari pembelajaran mesin, pengenalan pola, statistik, database, dan visualisasi untuk penanganan permasalahan pengambilan informasi dari database yang besar (Daniel T. Larose, 2005).

2.3.1 Pengelompokan Data Mining

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu (Daniel T. Larose, 2005):

1. Deskripsi

Deskripsi adalah menggambarkan pola dan kecenderungan yang terdapat dalam data secara sederhana. Deskripsi dari pola dan kecenderungan sering memberikan kemungkinan penjelasan untuk suatu pola atau kecenderungan.

2. Klasifikasi

Suatu teknik dengan melihat pada kelakuan dan atribut dari kelompok yang telah didefinisikan. Teknik ini dapat memberikan klasifikasi pada data baru dengan memanipulasi data yang telah diklasifikasi dan dengan menggunakan hasilnya untuk memberikan sejumlah aturan. Klasifikasi menggunakan *supervised learning*.

3. Estimasi

Estimasi hampir sama dengan klasifikasi, perbedaannya adalah variabel target estimasi lebih ke arah numerik daripada ke arah kategori. Model dibangun dengan menggunakan *record* lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi.

4. Prediksi

Prediksi memiliki kesamaan dengan klasifikasi dan estimasi, perbedaannya adalah hasil dari prediksi akan ada dimasa mendatang. Beberapa teknik yang digunakan dalam klasifikasi dan estimasi dapat juga digunakan (untuk keadaan yang tepat) untuk prediksi.

5. Klastering

Klastering merupakan pengelompokan *record*, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan dengan *record-record* dalam kluster lain. Klastering menggunakan *unsupervised learning*.

6. Asosiasi

Tugas asosiasi atau sering disebut juga sebagai *market basket analysis* dalam data mining adalah menemukan relasi atau korelasi diantara himpunan item-item dan menemukan atribut yang muncul dalam satu waktu. Asosiasi menggunakan *unsupervised learning*. Penting tidaknya suatu aturan asosiatif dapat diketahui dengan dua parameter, *support* dan *confidence*.

Metode yang akan digunakan pada penelitian ini termasuk kedalam kelompok prediksi, karena menggunakan teknik klasifikasi yang hasilnya akan ada dimasa mendatang.

2.3.2 Knowledge Discovery in Databases (KDD)

Knowledge Discovery in Databases (KDD) adalah keseluruhan proses untuk mengkonversi data mentah menjadi suatu pengetahuan yang bermanfaat. Istilah data mining dan *Knowledge Discovery in Databases* (KDD) sering kali digunakan secara bergantian untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar. Sebenarnya kedua istilah tersebut memiliki konsep yang berbeda, tetapi berkaitan satu sama lain. Salah satu tahapan dalam keseluruhan proses KDD adalah data mining.

Proses KDD secara garis besar dapat dijelaskan sebagai berikut (Fayyad, 1996).

1. Data Selection

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang akan digunakan, disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. Pre-processing/Cleaning

Sebelum proses data mining, perlu dilakukan proses *cleaning* pada data yang menjadi fokus KDD. Proses *cleaning* mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak.

3. Transformation

Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses data mining. Proses *coding* dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. Data mining

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik tertentu. Teknik, metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

5. Interpretation Evaluation

Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut *interpretation*. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya.

2.4 Decision Tree (Pohon Keputusan)

Pohon keputusan merupakan metode klasifikasi dan prediksi yang sangat kuat dan terkenal. Metode pohon keputusan mengubah fakta yang sangat besar menjadi pohon keputusan yang merepresentasikan aturan. Aturan dapat dengan mudah dipahami dengan bahasa alami. Selain itu dapat diekspresikan dalam bentuk bahasa basis data seperti *Structure Query Language* untuk mencari *record* pada kategori tertentu (Kusrini dan Emha, 2009).

Pohon keputusan juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel input dengan variabel target.

Sebuah pohon keputusan adalah sebuah struktur yang dapat digunakan untuk membagi kumpulan data yang besar menjadi himpunan-himpunan *record* yang lebih kecil dengan menerapkan serangkaian aturan keputusan, dengan masing-masing rangkaian pembagian, anggota himpunan hasil menjadi mirip satu dengan yang lain.

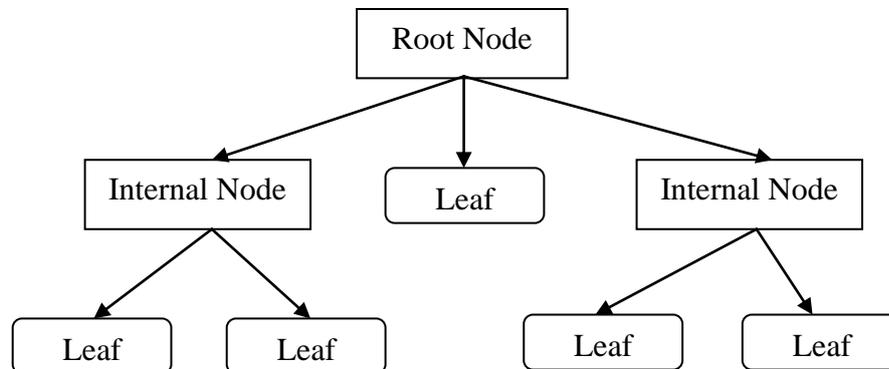
2.4.1 Model Decision Tree

Decision tree adalah *flow-chart* seperti *struktur tree*, dimana tiap *internal node* menunjukkan sebuah test pada sebuah atribut, tiap cabang menunjukkan hasil dari test, dan *leaf node* menunjukkan *class-class* atau *class distribution*.

Selain karena pembangunannya relatif cepat, hasil dari model yang dibangun mudah untuk dipahami. Pada *decision tree* terdapat 3 jenis *node*, yaitu:

- a. *Root Node*, merupakan *node* paling atas, pada *node* ini tidak ada *input* dan bisa tidak mempunyai *output* atau mempunyai *output* lebih dari satu.
- b. *Internal Node*, merupakan *node* percabangan, pada *node* ini hanya terdapat satu *input* dan mempunyai *output* minimal dua.
- c. *Leaf node* atau *terminal node*, merupakan *node* akhir, pada *node* ini hanya terdapat satu *input* dan tidak mempunyai *output*.

Contoh dari model pohon keputusan yaitu seperti pada **gambar 2.1** berikut:



Gambar 2.1 Model *Decision Tree*

2.4.2 Decision Tree C4.5

C4.5 adalah algoritma yang sudah banyak dikenal dan digunakan untuk klasifikasi data yang memiliki atribut-atribut numerik dan kategorikal. Hasil dari proses klasifikasi yang berupa aturan-aturan dapat digunakan untuk memprediksi nilai atribut bertipe *diskret* dari *record* yang baru.

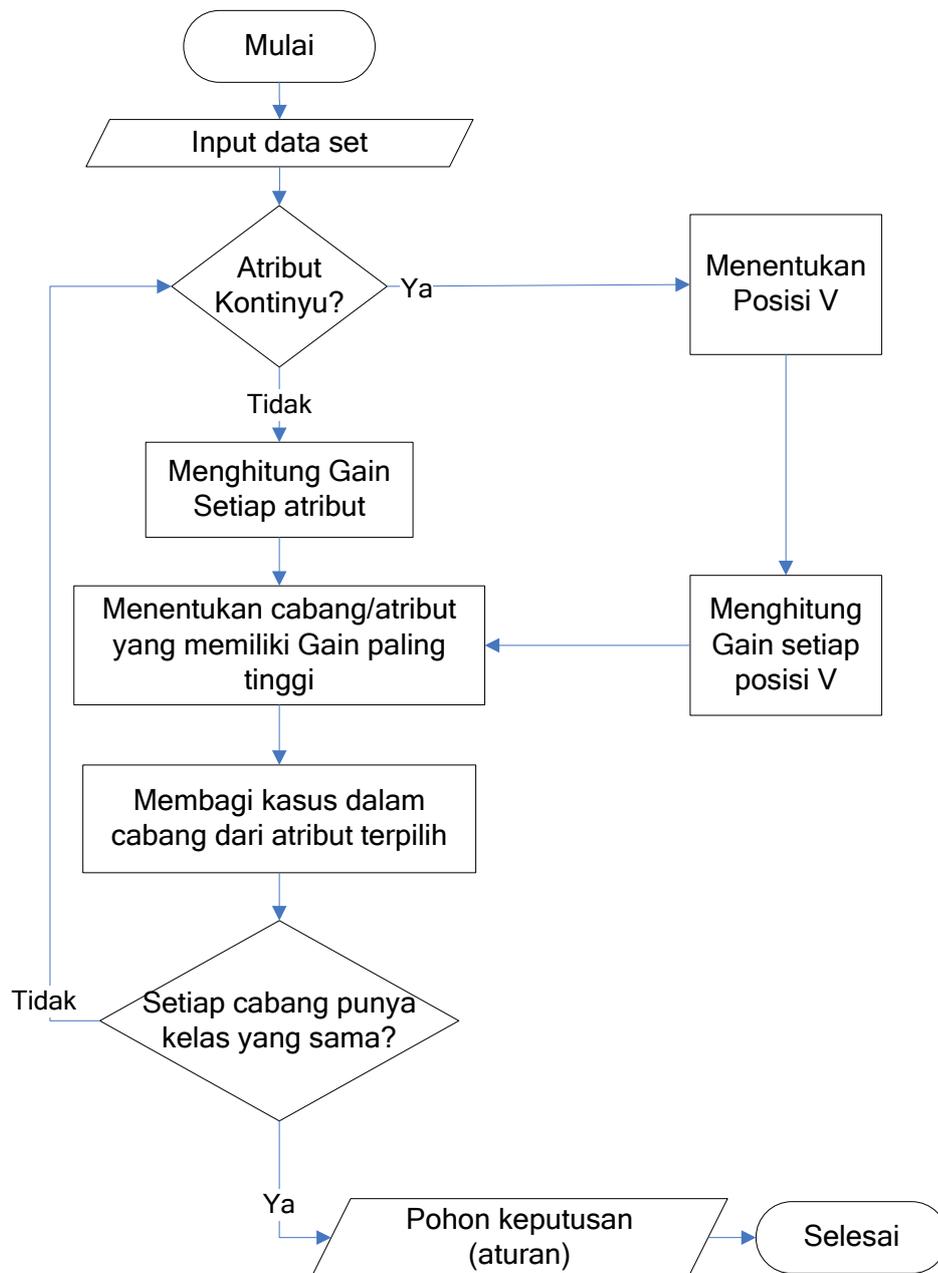
Algoritma C4.5 sendiri merupakan pengembangan dari algoritma ID3, dimana pengembangan dilakukan dalam hal bisa mengatasi *missing* data, bisa mengatasi data *kontinyu*, dan *pruning*.

2.4.3 Algoritma

Secara umum algoritma C4.5 untuk membangun pohon keputusan adalah sebagai berikut:

1. Pilih atribut sebagai akar.
2. Buat cabang untuk tiap-tiap nilai.
3. Bagi kasus dalam cabang.
4. Ulangi proses untuk setiap cabang sampai semua kasus pada cabang memiliki kelas yang sama.

Berikut ini akan dijelaskan secara lebih detail algoritma C4.5 menggunakan *flowcart* yang disajikan pada **gambar 2.2**



Gambar 2.2 Flowchart algoritma *Decision Tree C4.5*

Untuk memilih atribut sebagai simpul akar (*root node*) atau simpul dalam (*internal node*), didasarkan pada nilai *information gain* tertinggi dari atribut-atribut yang ada. Sebelum perhitungan *information gain*, akan dilakukan perhitungan *entropy*. *Entropy* merupakan distribusi probabilitas dalam teori informasi dan diadopsi kedalam algoritma *C4.5* untuk mengukur tingkat

homogenitas distribusi kelas dari sebuah himpunan data (*data set*). Semakin tinggi tingkat *entropy* dari sebuah data maka semakin homogen distribusi kelas pada data tersebut. Perhitungan *information gain* menggunakan rumus 2.2, sedangkan *entropy* menggunakan rumus 2.3.

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

dimana,

S : Himpunan kasus

A : Atribut

n : Jumlah partisi atribut A

|S_i| : Jumlah kasus pada partisi ke i

|S| : Jumlah kasus dalam S

$$Entropy(S) = - \sum_{i=1}^n p_i * \log_2 p_i \quad (2.3)$$

dimana,

S : Himpunan kasus

A : Fitur

n : Jumlah partisi S

p_i : Proporsi dari S_i terhadap S

Selain *Information Gain* kriteria yang lain untuk memilih atribut sebagai pemecah adalah *Rasio Gain*. Perhitungan rasio gain menggunakan rumus 2.4, sedangkan split information menggunakan rumus 2.5.

$$GainRasio(S, A) = \frac{Gain(S, A)}{SplitInformation(S, A)} \quad (2.4)$$

$$SplitInformation(S, A) = - \sum_{i=1}^c \frac{S_i}{S} \log_2 \frac{S_i}{S} \quad (2.5)$$

dimana S₁ sampai S_c adalah c subset yang dihasilkan dari pemecahan S dengan menggunakan atribut A yang mempunyai sebanyak c nilai.

2.4.4 Contoh Perhitungan

Berikut ini akan dijelaskan ilustrasi dari alur proses perhitungan algoritma *Decision Tree C4.5*. Data set yang digunakan pada contoh ini adalah data untuk menentukan *Play* atau *Don't Play* dengan beberapa atribut yaitu atribut *outlook*, *temperature*, *humidity*, dan *windy*. Dimana atribut *temperature* dan *humidity* bertipe kontinyu sedangkan *outlook* dan *windy* bertipe kategorikal. Sedangkan kolom *Class* adalah kelas tujuannya atau label kelas-nya.

Tabel 2.1 Contoh data set

Outlook	Temperature	Humidity	Windy	Class
sunny	75	70	TRUE	Play
sunny	80	90	TRUE	Don't Play
sunny	85	85	FALSE	Don't Play
sunny	72	95	FALSE	Don't Play
sunny	69	70	FALSE	Play
overcast	72	90	TRUE	Play
overcast	83	78	FALSE	Play
overcast	64	65	TRUE	Play
overcast	81	75	FALSE	Play
rain	71	80	TRUE	Don't Play
rain	65	70	TRUE	Don't Play
rain	75	80	FALSE	Play
rain	68	80	FALSE	Play
rain	70	96	FALSE	Play

Pada contoh ini rumus yang digunakan untuk memilih atribut sebagai *node* adalah rumus *information gain*. Proses pertama adalah menghitung *entropy* untuk semua data.

Jumlah class play = 9

Jumlah class don't play = 5

Berikut adalah perhitungan *entropy* untuk semua data:

$$\begin{aligned} Entropy(S) &= -\frac{9}{14} * \log_2 \left(\frac{9}{14} \right) - \frac{5}{14} * \log_2 \left(\frac{5}{14} \right) \\ &= 0.940 \end{aligned}$$

Selanjutnya menghitung *gain* untuk setiap atribut. Berikut adalah contoh perhitungan gain untuk atribut *outlook*:

Tabel 2.2 Distribusi jumlah atribut *outlook*

Nilai Outlook	Σ Play	Σ Don't Play	Total
Sunny	2	3	5
Overcast	4	0	4
Rain	3	2	5

Berdasarkan tabel 2.2, maka nilai *information gain* untuk atribut *outlook* adalah sebagai berikut:

$$\begin{aligned}
 Gain(outlook) &= 0.940 - \left(\frac{5}{14} * \left(-\frac{2}{5} * \log_2 \left(\frac{2}{5} \right) - \frac{3}{5} * \log_2 \left(\frac{3}{5} \right) \right) \right. \\
 &\quad + \frac{4}{14} * \left(-\frac{4}{4} * \log_2 \left(\frac{4}{4} \right) - \frac{0}{4} * \log_2 \left(\frac{0}{4} \right) \right) \\
 &\quad \left. + \frac{5}{14} * \left(-\frac{3}{5} * \log_2 \left(\frac{3}{5} \right) - \frac{2}{5} * \log_2 \left(\frac{2}{5} \right) \right) \right) \\
 &= 0.940 - 0.694 \\
 &= 0.246
 \end{aligned}$$

Untuk perhitungan atribut yang bertipe kontinyu, harus menentukan *posisi V* terbaik yang dinyatakan dalam perbandingan ($A \leq V$) atau ($A > V$). Berikut akan dijelaskan contoh perhitungan dari atribut *temperature*.

Misal posisi *V* yang akan digunakan pada atribut *temperature* adalah 65,70,75,dan 80, kemudian dihitung nilai *information gain*-nya.

Contoh perhitungan *temperature* posisi $v=65$:

$$\begin{aligned}
 Gain(temp) &= 0.940 - \left(\frac{2}{14} * \left(-\frac{1}{2} * \log_2 \left(\frac{1}{2} \right) - \frac{1}{2} * \log_2 \left(\frac{1}{2} \right) \right) \right. \\
 &\quad \left. + \frac{12}{14} * \left(-\frac{8}{12} * \log_2 \left(\frac{8}{12} \right) - \frac{4}{12} * \log_2 \left(\frac{4}{12} \right) \right) \right) \\
 &= 0.940 - 0.930 \\
 &= 0.010
 \end{aligned}$$

Berikut hasil perhitungan atribut numerik untuk setiap posisi yang telah ditentukan:

Tabel 2.3 Hasil perhitungan posisi V untuk atribut *temperature*

Temperature	65		70		75		80	
	≤	>	≤	>	≤	>	≤	>
Play	1	8	4	5	7	2	7	2
Don't Play	1	4	1	4	3	2	4	1
Jumlah	2	12	5	9	10	4	11	3
Entropy	1.000	0.918	0.722	0.991	0.881	1.000	0.946	0.918
Gain	0.010		0.045		0.025		0.0005	

Berdasarkan tabel 2.3, nilai gain tertinggi adalah 70, maka nilai information gain pada atribut temperature adalah 0.045. Hasil perhitungan pada setiap atribut disajikan pada tabel 2.4

Tabel 2.4 Hasil perhitungan *Information gain* untuk setiap atribut

		Jumlah	Play	Don't Play	Entropy	Gain
Total		14	9	5	0.940	
Outlook	Sunny	5	2	3	0.971	0.247
	Overcast	4	4	0	0.000	
	Rain	5	3	2	0.971	
Temperature	≤ 70	5	4	1	0.722	0.045
	> 70	9	5	4	0.991	
Humidity	≤ 80	9	7	2	0.764	0.102
	> 80	5	2	3	0.971	
Windy	TRUE	6	3	3	1.000	0.048
	FALSE	8	6	2	0.811	

Berdasarkan tabel 2.4 menunjukkan bahwa atribut *outlook* memiliki nilai gain tertinggi, maka atribut *outlook* akan menjadi *node*. Karena atribut outlook memiliki tiga nilai atribut atau lebih dari dua, maka dilakukan perhitungan rasio gain untuk memilih pilihan percabangan terbaik. Berikut adalah contoh perhitungan rasio gain untuk pilihan percabangan {sunny, overcast, rain}.

$$\begin{aligned}
 \text{Split info}(\text{Semua}, \text{overcast}) &= \left(-\frac{5}{14} * \log_2 \left(\frac{5}{14} \right) \right) + \left(-\frac{4}{14} * \log_2 \left(\frac{4}{14} \right) \right) \\
 &\quad + \left(-\frac{5}{14} * \log_2 \left(\frac{5}{14} \right) \right) \\
 &= 0.531 + 0.516 + 0.531 = 1.577
 \end{aligned}$$

$$\begin{aligned} \text{Rasio Gain}(\text{Semua}, \text{overcast}) &= \frac{0.247}{1.577} \\ &= 0.156 \end{aligned}$$

Hasil untuk perhitungan *rasio gain* lainnya ada pada tabel 2.5.

Tabel 2.5 Hasil perhitungan *Rasio gain* untuk setiap pilihan cabang

		Jumlah	Split Inf	Gain	Rasio Gain
Total		14		0.247	
Pilihan 1	sunny	5	1.577		0.156
	overcast	4			
	rain	5			
Pilihan 2	sunny	5	0.940		0.262
	overcast rain	9			
Pilihan 3	sunny overcast	9	0.940		0.262
	rain	5			
Pilihan 4	sunny rain	10	0.863		0.286
	overcast	4			

Dari tabel 2.5 pilihan 4 yaitu {*sunny, rain*} dan {*overcast*} memiliki nilai *rasio gain* tertinggi, maka atribut terpilih (*outlook*) akan dibagi menjadi dua cabang. Pembagian cabang disajikan pada tabel 2.6 dan tabel 2.7.

Tabel 2.6 Pembagian cabang (*sunny, rain*)

Outlook	Temperature	Humidity	Windy	Class
sunny	75	70	TRUE	Play
sunny	80	90	TRUE	Don't Play
sunny	85	85	FALSE	Don't Play
sunny	72	95	FALSE	Don't Play
sunny	69	70	FALSE	Play
rain	71	80	TRUE	Don't Play
rain	65	70	TRUE	Don't Play
rain	75	80	FALSE	Play
rain	68	80	FALSE	Play
rain	70	96	FALSE	Play

Tabel 2.7 Pembagian cabang (*overcast*)

Outlook	Temperature	Humidity	Windy	Class
overcast	72	90	TRUE	Play
overcast	83	78	FALSE	Play
overcast	64	65	TRUE	Play
overcast	81	75	FALSE	Play

Pada cabang *overcast* memiliki kelas yang sama yaitu *Play*, maka *node* ini akan menjadi daun dengan nilai *Play*. Sedangkan cabang *sunny* dan *rain* masih ada kelas yang berbeda, maka akan memilih atribut sebagai *node*. Proses tersebut akan berulang sampai semua kasus pada cabang memiliki kelas yang sama atau menjadi daun (*leaf*).

2.5 Penelitian Sebelumnya

Penelitian sebelumnya yang menggunakan metode pohon keputusan C4.5 adalah penelitian yang berjudul “*Aplikasi Data Mining untuk Memprediksi Masa Studi Mahasiswa menggunakan Algoritma C4.5 (Studi kasus: Jurusan Teknik Komputer UNIKOM)*”, dibuat oleh Ida Hamidah (Universitas Komputer Indonesia Bandung, 2012). Tujuan dari penelitian tersebut adalah untuk mengimplementasikan algoritma pembentukan pohon keputusan C4.5 dalam sebuah aplikasi untuk memprediksi masa studi mahasiswa. Atribut-atribut yang digunakan adalah nilai matakuliah Algoritma Pemrograman 1, Fisika Dasar 1, Fisika Dasar 2, Kalkulus 1, Kalkulus 2, Pengantar Sistem Komputer, dan label kelasnya adalah masa studi mahasiswa yang dibagi menjadi dua yaitu ≤ 5 tahun dan > 5 tahun. Hasil dari penelitian tersebut, aplikasi yang dirancang menggunakan algoritma C4.5 dapat memprediksi masa studi mahasiswa.

Penelitian menggunakan algoritma C4.5 juga dilakukan oleh Angga Raditya (Universitas Gunadarma, 2011), yaitu penelitian tentang pencarian pola prediksi hujan menggunakan algoritma C4.5. Data yang digunakan adalah data cuaca yang tersimpan di *World Meteorologi Organization* (organisasi pengawas cuaca dunia). Dari data tersebut akan diolah untuk pola prediksi hujan dengan menggunakan algoritma C4.5. akurasi pola prediksi yang didapat mampu

mencapai 79%. Kelebihan algoritma C4.5 dalam membangun pohon keputusan prediksi cuaca adalah kemampuannya menangani data kontinyu maupun data nominal, karena hampir seluruh atribut cuaca yang digunakan bertipe data kontinyu. Selain itu dalam membangun keputusan tingkat *error*-nya lebih sedikit.

Khafizh Hastuti (Universitas Dian Nuswantoro, Semarang, 2012) melakukan penelitian yaitu penelitian komparasi algoritma klasifikasi data mining untuk prediksi mahasiswa non aktif. Algoritma yang digunakan adalah *logistic regression*, *decision tree*, *naïve bayes* dan *neural network*. Hasil proses klasifikasi dievaluasi dengan menggunakan *cross validation*, *confusion matrix*, *ROC Curve* dan *T-Test* untuk mengetahui algoritma klasifikasi data mining yang paling akurat. Hasil dari komparasi algoritma klasifikasi tersebut *decision tree* merupakan algoritma yang paling akurat, namun tidak dominan terhadap algoritma lain. Algoritma yang paling dominan adalah *logistic regression* namun akurasi paling rendah.

Penelitian lainnya yang terkait dengan Sistem prediksi prestasi yaitu penelitian yang berjudul “*Sistem Inferensi Fuzzy untuk Memprediksi Prestasi Belajar Mahasiswa Berdasarkan Nilai Ujian Nasional, Tes Potensi Akademik, dan Motivasi Belajar*”, dibuat oleh Hindayati Mustafidah dan Dwi Aryanto (Universitas Muhammadiyah Purwokerto, 2012). Tujuan dari penelitian tersebut adalah melakukan rancang bangun sistem yang bersifat interaktif yang bisa berfungsi sebagai sistem prediksi otomatis terhadap prestasi belajar mahasiswa dalam mengikuti perkuliahan di perguruan tinggi yang didasarkan atas nilai UN, nilai TPA, dan tingkat motivasi menggunakan *system inferensi fuzzy* metode *Mamdani*. Simpulan yang didapat dari penelitian tersebut adalah dengan menggunakan aplikasi logika fuzzy *system inferensi fuzzy* metode *Mamdani* dapat diprediksi prestasi belajar mahasiswa berdasarkan nilai TPA, NEM, dan tingkat motivasi belajar mahasiswa.