BAB II
TINJAUAN PUSTAKA

2.1 Pengertian *Data Mining*

Data mining adalah suatu istilah yang digunakan untuk menemukan pengetahuan yang tersembunyi di dalam database. *Data mining* merupakan proses semi otomatik yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi pengetahuan potensial dan berguna yang bermanfaat yang tersimpan di dalam database besar. (Turban et al, 2005). Menurut Gartner Group *data mining* adalah suatu proses menemukan hubungan yang berarti, pola, dan kecenderungan dengan memeriksa dalam sekumpulan besar data yang tersimpan dalam penyimpanan dengan menggunakan teknik pengenalan pola seperti teknik statistik dan matematika (Larose, 2006).

Selain definisi di atas beberapa definisi juga diberikan seperti, "*data mining* adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual." (Pramudiono, 2006). "*Data mining* adalah analisis otomatis dari data yang berjumlah besar atau kompleks dengan tujuan untuk menemukan pola atau kecenderungan yang penting yang biasanya tidak disadari keberadaannya." (Pramudiono, 2006).

"*Data mining* merupakan analisis dari peninjauan kumpulan data untuk menemukan hubungan yang tidak diduga dan meringkas data dengan cara yang berbeda dengan sebelumnya, yang dapat dipahami dan bermanfaat bagi pemilik data." (Larose, 2006). "*Data mining* merupakan bidang dari beberapa keilmuan yang menyatukan teknik dari pembelajaran mesin, pengenalan pola, statistik, database, dan visualisasi untuk penanganan permasalahan pengambilan informasi dari database yang besar." (Larose, 2006).

Kemajuan luar biasa yang terus berlanjut dalam bidang *data mining* didorong oleh beberapa faktor, antara lain : (Larose, 2006)

1. Pertumbuhan yang cepat dalam kumpulan data.
2. Penyimpanan data dalam *data warehouse,* sehingga seluruh perusahaan memiliki akses ke dalam database yang baik.
3. Adanya peningkatan akses data melalui navigasi web dan intranet.
4. Tekanan kompetisi bisnis untuk meningkatkan penguasaan pasar dalam globalisasi ekonomi.
5. Perkembangan teknologi perangkat lunak untuk *data mining* (ketersediaan teknologi).
6. Perkembangan yang hebat dalam kemampuan komputasi dan pengembangan kapasitas media penyimpanan.

Berdasarkan definisi-definisi yang telah disampaikan, hal penting yang terkait dengan *data mining* adalah :

1. *Data mining* merupakan suatu proses otomatis terhadap data yang sudah ada.
2. Data yang akan diproses berupa data yang sangat besar.
3. Tujuan *data mining* adalah mendapatkan hubungan atau pola yang mungkin memberikan indikasi yang bermanfaat.

Hubungan yang dicari dalam *data mining* dapat berupa hubungan antara dua atau lebih dalam satu dimensi. Misalnya dalam dimensi produk, dapat di lihat keterkaitan pembelian suatu produk dengan produk yang lain. Selain itu, hubungan juga dapat dilihat antara dua atau lebih atribut dan dua atau lebih objek. (Ponniah, 2001).

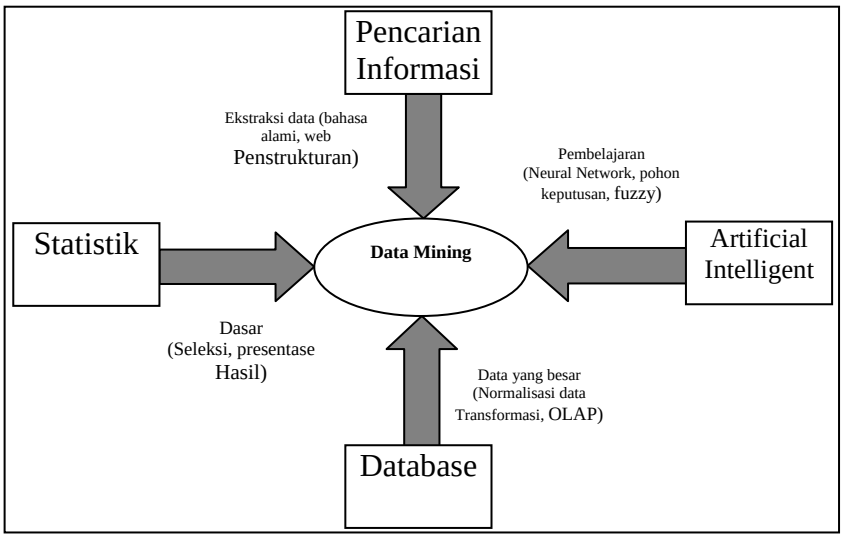
Sementara itu, penemuan pola merupakan keluaran lain dari *data mining*. Misalkan sebuah perusahaan yang akan meningkatkan fasilitas kartu kredit dari pelanggan, maka perusahaan akan mencari pola dari pelanggan-pelanggan yang ada untuk mengetahui pelanggan yang potensial dan pelanggan yang tidak potensial.

Beberapa definisi awal dari *data mining* meyertakan fokus pada proses otomatisasi. Berry dan Linoff, (2004) dalam buku *Data Mining Technique for Marketing, Sales, and Customer Support* mendefinisikan *data mining* sebagai suatu proses eksplorasi dan analisis secara otomatis maupun semi otomatis

terhadap data dalam jumlah besar dengan tujuan menemukan pola atau aturan yang berarti (Larose, 2006).

Tiga tahun kemudian, dalam buku *Mastering Data Mining* mereka memberikan definisi ulang terhadap pengertian *data mining* dan memberikan pernyataan bahwa "jika ada yang kami sesalkan adalah frasa secara otomatis maupun semi otomatis, karena kami merasa hal tersebut memberikan fokus berlebih pada teknik otomatis dan kurang pada eksplorasi dan analisis". Hal tersebut memberikan pemahaman yang salah bahwa *data mining* merupakan produk yang dapat dibeli dibandingkan keilmuan yang harus dikuasai (Larose, 2006).

Pernyataan tersebut menegaskan bahwa dalam *data mining* otomatisasi tidak menggantikan campur tangan manusia. Manusia harus ikut aktif dalam setiap fase dalam proses *data mining.* Kehebatan kemampuan algoritma *data mining* yang terdapat dalam perangkat lunak analisis yang terdapat saat ini memungkinkan terjadinya kesalahan penggunaan yang berakibat fatal. Pengguna mungkin menerapkan analisis yang tidak tepat terhadap kumpulan data dengan menggunakan pendekatan yang berbeda. Oleh karenanya, dibutuhkan pemahaman tentang statistik dan struktur model matematika yang mendasari kerja perangkat lunak (Larose, 2006).



Gambar 2.1 Bidang Ilmu *Data Mining*

Data mining bukanlah suatu bidang yang sama sekali baru. Salah satu kesulitan untuk mendefinisikan *data mining* adalah kenyataan bahwa *data mining* mewarisi banyak aspek dan teknik dari bidang-bidang ilmu yang sudah mapan terlebih dahulu. Gambar 2.1 menunjukkan bahwa *data mining* memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (*artificial intelligent*), *machine learning*, statistik, database, dan juga *information retrieval* (Pramudiono, 2006).

Istilah *data mining* dan *Knowledge Discovery in Database* (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. Dan 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 untuk proses *data mining*, disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. *Pre-processing/Cleaning*

   Sebelum proses *data mining* dapat dilaksanakan, 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 (tipografi). Juga dilakukan proses *enrichment*, yaitu proses "memperkaya" data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.

3. *Transformation*

   Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses *data mining*. Proses *coding*

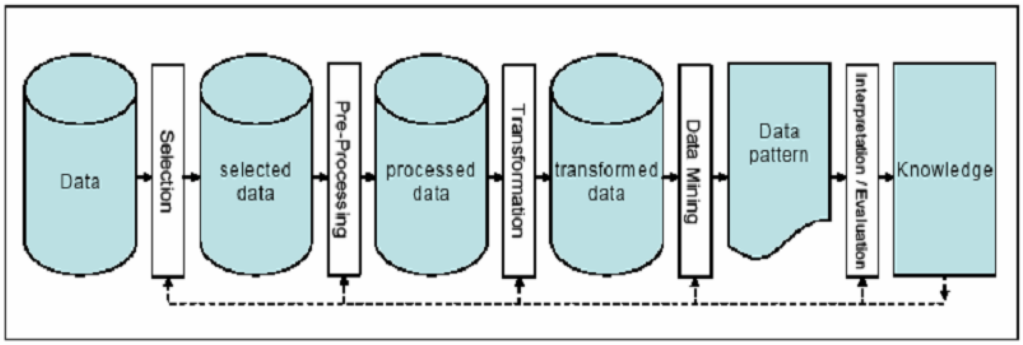
dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. *Data mining*

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

5. Interpretation/Evalution

   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. Penjelasan di atas dapat direfresentasikan pada Gambar 2.2

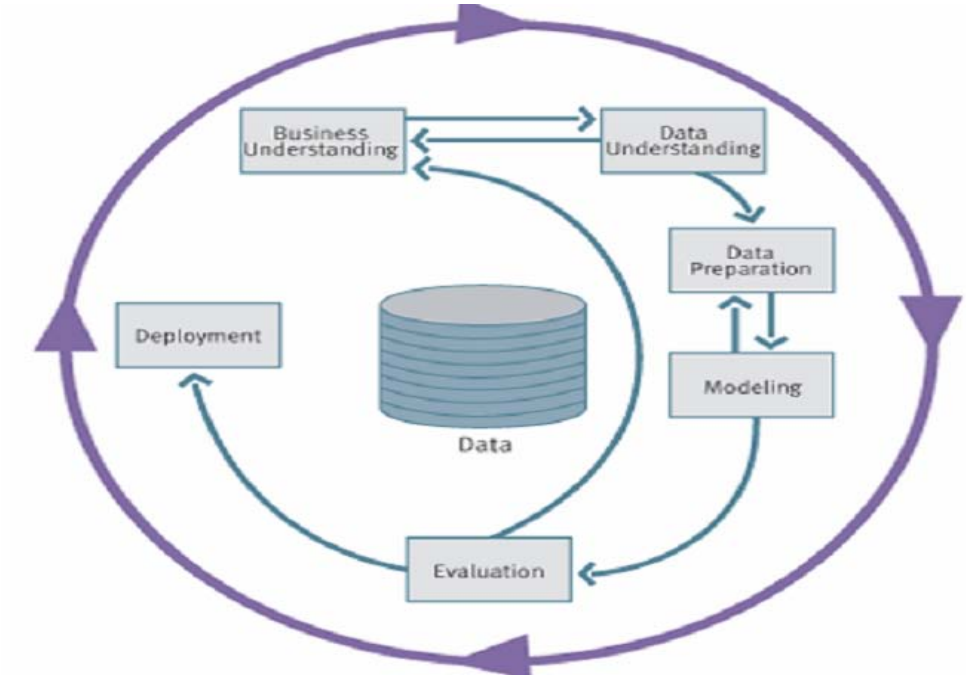


Gambar 2.2 Proses dari *Data Mining*

Sumber: SPSS, 2004

Cross-Industry Standart Process for Data Mining (CRISP-DM) yang di kembangkan tahun 1996 oleh analisis dari beberapa industri seperti Daimler Chrysler, SPSS dan NCR. CRISP-DM menyediakan standar proses *data mining* sebagai strategi pemecahan masalah secara umum dari bisnis atau unit penelitian.

Dalam CRISP-DM sebuah proyek *data mining* memiliki siklus hidup yang terbagi dalam enam fase Gambar 2.3. Keseluruhan fase berurutan yang ada tersebut bersifat adaptif. Fase berikutnya dalam urutan bergantung kepada keluaran dari fase sebelumnya. Hubungan penting antar fase digambarkan dengan panah. Sebagai contoh, jika proses berada pada fase *modeling.* Berdasar pada perilaku dan karakteristik model, proses mungkin kembali kepada fase *data preparation* untuk perbaikan lebih lanjut terhadap data atau berpindah maju kepada fase *evaluation.*



Gambar 2.3 Proses *Data Mining* Menurut CRISP-DM

Sumber: CRISP, 2005

Enam fase CRISP-DM (*Cross Industry Standard Process for Data Mining*) (Larose, 2006).

1. Fase Pemahaman Bisnis (*Business Understanding Phase*)
    a. Penentuan tujuan proyek dan kebutuhan secara detail dalam lingkup bisnis atau unit penelitian secara keseluruhan.
    b. Menerjemahkan tujuan dan batasan menjadi formula dari permasalahan *data mining*.

c. Menyiapkan strategi awal untuk mencapai tujuan.

2. Fase Pemahaman Data (*Data Understanding Phase*)
   a. Mengumpulkan data.
   b. Menggunakan analisis penyelidikan data untuk mengenali lebih lanjut data dan pencarian pengetahuan awal.
   c. Mengevaluasi kualitas data.
   d. Jika diinginkan, pilih sebagian kecil kelompok data yang mungkin mengandung pola dari permasalahan
3. Fase Pengolahan Data (*Data Preparation Phase*)
   a. Siapkan dari data awal, kumpulan data yang akan digunakan untuk keseluruhan fase berikutnya. Fase ini merupakan pekerjaan berat yang perlu dilaksanakan secara intensif.
   b. Pilih kasus dan variabel yang ingin dianalisis dan yang sesuai analisis yang akan dilakukan.
   c. Lakukan perubahan pada beberapa variabel jika dibutuhkan.
   d. Siapkan data awal sehingga siap untuk perangkat pemodelan.
4. Fase Pemodelan (*Modeling Phase*)
   a. Pilih dan aplikasikan teknik pemodelan yang sesuai.
   b. Kalibrasi aturan model untuk mengoptimalkan hasil.
   c. Perlu diperhatikan bahwa beberapa teknik mungkin untuk digunakan pada permasalahan *data mining* yang sama.
   d. Jika diperlukan, proses dapat kembali ke fase pengolahan data untuk menjadikan data ke dalam bentuk yang sesuai dengan spesifikasi kebutuhan teknik *data mining* tertentu.
5. Fase Evaluasi (*Evaluation Phase*)
   a. Mengevaluasi satu atau lebih model yang digunakan dalam fase pemodelan untuk mendapatkan kualitas dan efektivitas sebelum disebarkan untuk digunakan.
   b. Menetapkan apakah terdapat model yang memenuhi tujuan pada fase awal.

c. Menentukan apakah terdapat permasalahan penting dari bisnis atau penelitian yang tidak tertangani dengan baik.
d. Mengambil keputusan berkaitan dengan penggunaan hasil dari *data mining.*

6. Fase Penyebaran (*Deployment Phase)*
   a. Menggunakan model yang dihasilkan. Terbentuknya model tidak menandakan telah terselesaikannya proyek.
   b. Contoh sederhana penyebaran: Pembuatan laporan.
   c. Contoh kompleks Penyebaran: Penerapan proses *data mining* secara paralel pada departemen lain. Informasi lebih lanjut mengenai CRISP-DM dapat dilihat di *www.crisp-dm.org*

2.2 Pengelompokan *Data Mining*

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat di lakukan, yaitu (Larose, 2006).

1. Deskripsi

   Terkadang peneliti dan analisis secara sederhana ingin mencoba mencari cara untuk menggambarkan pola dan kecendrungan yang terdapat dalam data. Sebagai contoh, petugas pengumpulan suara mungkin tidak dapat menemukan keterangan atau fakta bahwa siapa yang tidak cukup profesional akan sedikit didukung dalam pemilihan presiden. Deskripsi dari pola dan kecendrungan sering memberikan kemungkinan penjelasan untuk suatu pola atau kecendrungan.

2. Estimasi

   Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih ke arah numerik dari pada ke arah kategori. Model dibangun menggunakan *record* lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi. Selanjutnya, pada peninjauan berikutnya estimasi nilai dari variabel target dibuat berdasarkan nilai variabel prediksi. Sebagai contoh, akan dilakukan estimasi tekanan darah sistolik pada pasien rumah sakit berdasarkan umur pasien, jenis kelamin, berat badan,

dan level sodium darah. Hubungan antara tekanan darah sistolik dan nilai variabel prediksi dalam proses pembelajaran akan menghasilkan model estimasi. Model estimasi yang dihasilkan dapat digunakan untuk kasus baru lainnya.

3. Prediksi

Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada di masa mendatang.

Contoh prediksi dalam bisnis dan penelitian adalah:

a. Prediksi harga beras dalam tiga bulan yang akan datang.
b. Prediksi presentase kenaikan kecelakaan lalu lintas tahun depan jika batas bawah kecepatan dinaikan.

Beberapa metode dan teknik yang digunakan dalam klasifikasi dan estimasi dapat pula digunakan (untuk keadaan yang tepat) untuk prediksi.

4. Klasifikasi

Dalam klasifikasi, terdapat target variabel kategori. Sebagai contoh, penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu pendapatan tinggi, pendapatan sedang, dan pendapatan rendah.

Contoh lain klasifikasi dalam bisnis dan penelitian adalah:

a. Menentukan apakah suatu transaksi kartu kredit merupakan transaksi yang curang atau bukan.
b. Memperkirakan apakah suatu pengajuan hipotek oleh nasabah merupakan suatu kredit yang baik atau buruk.
c. Mendiagnosa penyakit seorang pasien untuk mendapatkan termasuk kategori apa.

5. Pengklusteran

Pengklusteran merupakan pengelompokan *record*, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan.

Kluster adalah kumpulan *record* yang memiliki kemiripan suatu dengan yang lainnya dan memiliki ketidakmiripan dengan *record* dalam kluster lain.

Pengklusteran berbeda dengan klasifikasi yaitu tidak adanya variabel target dalam pengklusteran. Pengklusteran tidak mencoba untuk melakukan klasifikasi, mengestimasi, atau memprediksi nilai dari variabel target. Akan tetapi, algoritma pengklusteran mencoba untuk melakukan pembagian terhadap keseluruhan data menjadi kelompok-kelompok yang memiliki kemiripan (homogen), yang mana kemiripan dengan *record* dalam kelompok lain akan bernilai minimal.

Contoh pengklusteran dalam bisnis dan penelitian adalah:

a. Mendapatkan kelompok-kelompok konsumen untuk target pemasaran dari suatu produk bagi perusahaan yang tidak memiliki dana pemasaran yang besar.
b. Untuk tujuan audit akutansi, yaitu melakukan pemisahan terhadap prilaku finansial dalam baik dan mencurigakan.
c. Melakukan pengklusteran terhadap ekspresi dari gen, dalam jumlah besar.

6. Asosiasi

Tugas asosiasi dalam *data mining* adalah menemukan atribut yang muncul dalam suatu waktu. Dalam dunia bisnis lebih umum disebut analisis keranjang belanja.

Contoh asosiasi dalam bisnis dan penelitian adalah:

a. Meneliti jumlah pelanggan dari perusahaan telekomunikasi seluler yang diharapkan untuk memberikan respon positif terhadap penawaran *upgrade* layanan yang diberikan.
b. Menemukan barang dalam supermarket yang dibeli secara bersamaan dan barang yang tidak pernah dibeli bersamaan.

Untuk mendukung penelitian ini penulis menggunakan Algoritma C4.5 *decision tree*.

2.3 *Decision Tree*

Decision tree merupakan salah satu metode klasifikasi yang menggunakan representasi struktur pohon (*tree*) di mana setiap *node* merepresentasikan atribut,

cabangnya merepresentasikan nilai dari atribut, dan daun merepresentasikan kelas. *Node* yang paling atas dari *decision tree* disebut sebagai *root*.

Decision tree merupakan metode klasifikasi yang paling populer digunakan. 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*.

2.4 Algoritma C 4.5

Algoritma C 4.5 adalah salah satu metode untuk membuat *decision tree* berdasarkan training data yang telah disediakan. Algoritma C 4.5 merupakan pengembangan dari ID3. Beberapa pengembangan yang dilakukan pada C 4.5 adalah sebagai antara lain bisa mengatasi *missing value*, bisa mengatasi c*ontinue data*, dan *pruning*.

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. Dan mereka juga dapat diekspresikan dalam bentuk bahasa basis data seperti *Structured Query Language* untuk mencari *record* pada kategori tertentu. Pohon keputusan juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel *input* dengan sebuah variabel target.

Karena pohon keputusan memadukan antara eksplorasi data dan pemodelan, pohon keputusan sangat bagus sebagai langkah awal dalam proses pemodelan bahkan ketika dijadikan sebagai model akhir dari beberapa teknik lain.

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 (Berry dan Linoff, 2004).

Sebuah model pohon keputusan terdiri dari sekumpulan aturan untuk membagi sejumlah populasi yang heterogen menjadi lebih kecil, lebih homogen dengan memperhatikan pada variabel tujuannya. Sebuah pohon keputusan mungkin dibangun dengan seksama secara manual atau dapat tumbuh secara otomatis dengan menerapkan salah satu atau beberapa algoritma pohon keputusan untuk memodelkan himpunan data yang belum terklasifikasi.

Variabel tujuan biasanya dikelompokkan dengan pasti dan model pohon keputusan lebih mengarah pada perhitungan *probability* dari tiap-tiap *record* terhadap kategori-kategori tersebut atau untuk mengklasifikasi *record* dengan mengelompokkannya dalam satu kelas. Pohon keputusan juga dapat digunakan untuk mengestimasi nilai dari variabel *continue* meskipun ada beberapa teknik yang lebih sesuai untuk kasus ini.

Banyak algoritma yang dapat dipakai dalam pembentukan pohon keputusan,antara lain ID3, CART, dan C4.5 (Larose, 2006).

Data dalam pohon keputusan biasanya dinyatakan dalam bentuk tabel dengan atribut dan *record.* Atribut menyatakan suatu parameter yang dibuat sebagai kriteria dalam pembentukan pohon. Misalkan untuk menentukan main tenis, kriteria yang diperhatikan adalah cuaca, angin, dan temperatur.

Salah satu atribut merupakan atribut yang menyatakan data solusi per *item* data yang disebut target atribut. Atribut memiliki nilai-nilai yang dinamakan dengan *instance.* Misalkan atribut cuaca mempunyai *instance* berupa cerah, berawan, dan hujan (Basuki dan Syarif, 2003)

Proses pada pohon keputusan adalah mengubah bentuk data (tabel) menjadi model pohon, mengubah model pohon menjadi *rule*, dan menyederhanakan *rule* (Basuki dan Syarif, 2003).

Berikut ini algoritma dasar dari C4.5:

Input : sampel training, label training, atribut

1. Membuat simpul akar untuk pohon yang dibuat
2. Jika semua sampel positif, berhenti dengan suatu pohon dengan satu simpul akar, beri tanda (+)
3. Jika semua sampel negatif, berhenti dengan suatu pohon dengan satu simpul akar, beri tanda (-)
4. Jika atribut kosong, berhenti dengan suatu pohon dengan suatu simpul akar, dengan label sesuai nilai yang terbanyak yang ada pada label training
5. Untuk yang lain, Mulai
    a. A ------ atribut yang mengklasifikasikan sampel dengan hasil terbaik (berdasarkan *Gain* rasio)
    b. Atribut keputusan untuk simpul akar ----- A
    c. Untuk setiap nilai, *vi*, yang mungkin untuk A
        1) Tambahkan cabang di bawah akar yang berhubungan dengan A= *vi*
        2) Tentukan sampel Svi sebagai subset dari sampel yang mempunyai nilai vi untuk atrribut A
        3) Jika sampel Sv_i kosong
            i. Di bawah cabang tambahkan simpul daun dengan label = nilai yang terbanyak yang ada pada label training
            ii. Yang lain tambah cabang baru di bawah cabang yang sekarang C4.5 (sampel training, label training, atribut-[A])
    d. Berhenti

Mengubah *tree* yang dihasilkan dalam beberapa *rule*. Jumlah *rule* sama dengan jumlah *path* yang mungkin dapat dibangun dari *root* sampai *leaf node*.

Tree Pruning dilakukan untuk menyederhanakan *tree* sehingga akurasi dapat bertambah. *Pruning* ada dua pendekatan, yaitu :

a. *Pre-pruning*, yaitu menghentikan pembangunan suatu *subtree* lebih awal (yaitu dengan memutuskan untuk tidak lebih jauh mempartisi data training). Saat seketika berhenti, maka *node* berubah menjadi *leaf* (*node* akhir). *Node* akhir ini menjadi kelas yang paling sering muncul di antara *subset* sampel.
b. *Post-pruning*, yaitu menyederhanakan *tree* dengan cara membuang beberapa cabang *subtree* setelah *tree* selesai dibangun. *Node* yang jarang dipotong akan menjadi *leaf* (*node* akhir) dengan kelas yang paling sering muncul.

Untuk memudahkan penjelasan mengenai algoritma C 4.5 berikut ini disertakan contoh kasus yang dituangkan dalam Tabel 2.1

Tabel 2.1 Keputusan Bermain Tenis

No	CUACA	TEMPERATUR	KELEMBABAN	ANGIN	BERMAIN
1	Cerah	Panas	Tinggi	Tidak	Tidak
2	Cerah	Panas	Tinggi	Ya	Tidak
3	Mendung	Panas	Tinggi	Tidak	*Ya*
4	Hujan	Sedang	Tinggi	Tidak	*Ya*
5	Hujan	Dingin	Normal	Tidak	*Ya*
6	Hujan	Dingin	Normal	Ya	*Ya*
7	Mendung	Dingin	Normal	Ya	*Ya*
8	Cerah	Sedang	Tinggi	Tidak	*Ya*
9	Cerah	Dingin	Normal	Tidak	Tidak
10	Hujan	Sedang	Normal	Tidak	*Ya*
11	Cerah	Sedang	Normal	Ya	*Ya*
12	Mendung	Sedang	Tinggi	Ya	*Ya*
13	Mendung	Panas	Normal	Tidak	*Ya*
14	Hujan	Sedang	Tinggi	Ya	Tidak

Dalam kasus yang tertera pada Tabel 2.1 akan dibuat pohon keputusan untuk menentukan main tenis atau tidak dengan melihat keadaan cuaca, temperatur, kelembaban dan keadaan angin.

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

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

Untuk memilih atribut sebagai akar, didasarkan pada nilai *Gain* tertinggi dari atribut-atribut yang ada. Untuk menghitung *Gain* digunakan rumus seperti tertera dalam Rumus 1 (Craw, 2005).

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

Dengan

S : Himpunan Kasus

A : Atribut

N : Jumlah partisi atribut A

|Si| : Jumlah kasus pada partisi ke i

|S| : Jumlah kasus dalam S

Sedangkan perhitungan nilai *Entropy* dapat dilihat pada rumus 2 berikut (Craw, 2005):

$$Entropy(A) = \sum_{i=1}^{n} - pi * \log_2 pi$$

Dengan

S : Himpunan Kasus

A : Fitur

n : Jumlah partisi S

pi : Proporsi dari *Si* terhadap *S*

Berikut ini adalah penjelasan lebih rinci mengenai masing-masing langkah dalam pembentukan pohon keputusan dengan menggunakan algoritma C4.5 untuk menyelesaikan permasalahan pada Tabel 2.1

1. Menghitung jumlah kasus, jumlah kasus untuk keputusan Ya, jumlah kasus untuk keputusan Tidak, dan *Entropy* dari semua kasus dan kasus yang dibagi berdasarkan atribut cuaca, temperatur, kelembaban dan angin. Setelah itu lakukan penghitungan *Gain* untuk masing-masing atribut. Hasil perhitungan ditunjukkan oleh Tabel 2.2

Tabel 2.2 Perhitungan *Node* 1

Node			Jumlah Kasus (S)	Tidak (S1)	Ya (S2)	*Entropy*	*Gain*
1	TOTAL		14	4	10	0.863120569	
	CUACA						0.258521037
		MENDUNG	4	0	4		
		HUJAN	5	1	4	0.721928095	
		CERAH	5	3	2	0.970950594	
	TEMPERATUR						0.183850925
		DINGIN	4	0	4	0	
		PANAS	4	2	2	1	
		SEDANG	6	2	4	0.918295834	
	KELEMBABAN						0.370506501
		TINGGI	7	4	3	0.985228136	
		NORMAL	7	0	7	0	
	ANGIN						0.005977711
		TIDAK	8	2	6	0.811278124	
		YA	6	4	2	0.918295834	

Baris total kolom *Entropy* pada Tabel 2.2 dihitung dengan rumus 2, sebagai berikut:

$$Entropy(Total) = (-\frac{4}{14}*Log2(\frac{4}{14}))+(-\frac{10}{14}*Log2(\frac{10}{14}))$$

$$Entropy(Total) = 0.863120569$$

Sementara itu nilai *Gain* pada baris cuaca dihitung dengan menggunakan rumus 1, sebagai berikut :

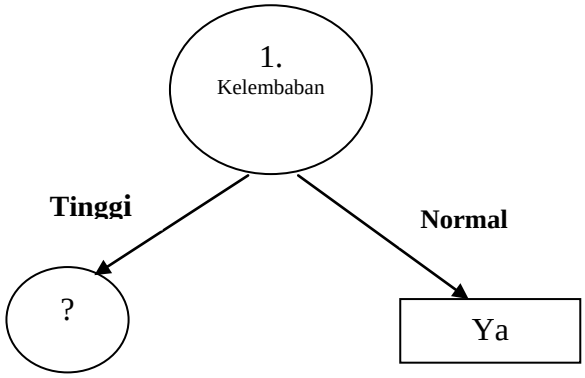
$$Gain(Total,Cuaca) = Entropy(Total) - \sum_{i=1}^{n} \frac{|Cuaca|}{|Total|} * Entropy(Cuaca)$$

$$Gain(Total,Cuaca) = 0.863120569 - ((\frac{4}{14}*0)+ ((\frac{5}{14}*0.723)+ ((\frac{45}{14}*0.97))$$

$$Gain(Total,Cuaca) = 0.23$$

Dari hasil pada Tabel 2.2 dapat diketahui bahwa atribut dengan *Gain* tertinggi adalah kelembaban yaitu sebesar 0.37. Dengan demikian kelembaban dapat menjadi *node* akar. Ada 2 nilai atribut dari kelembaban yaitu tinggi dan normal. Dari kedua nilai atribut tersebut, nilai atribut normal sudah mengklasifikasikan kasus menjadi 1 yaitu keputusannya Ya, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut tinggi masih perlu dilakukan perhitungan lagi.

Dari hasil tersebut dapat digambarkan pohon keputusan sementara, tampak seperti Gambar 2.4



Gambar 2.4 Pohon Keputusan Hasil Perhitungan *Node* 1

2. Menghitung jumlah kasus, jumlah kasus untuk keputusan Ya, jumlah kasus untuk keputusan Tidak, dan *Entropy* dari semua kasus dan kasus yang dibagi berdasarkan atribut cuaca, temperatur dan angin yang dapat menjadi node akar

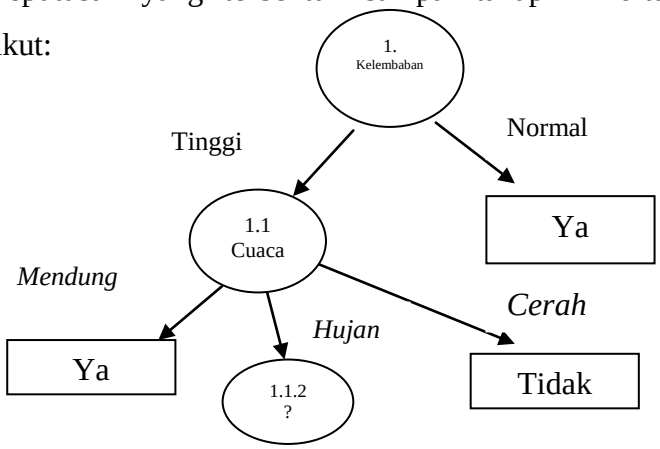
dari nilai atribut tinggi. Setelah itu lakukan penghitungan *Gain* untuk masing-masing atribut. Hasil perhitungan ditunjukkan oleh Tabel 2.3

Tabel 2.3 Perhitungan *Node* 1.1

Node			Jumlah Kasus (S)	Tidak (S1)	Ya (S2)	Entropy	Gain
1.1	KELEMBABAN-TINGGI		7	4	3	0.985228136	
	CUACA						0.69951385
		MENDUNG	2	0	2	0	
		HUJAN	2	1	1	1	
		CERAH	2	3	0	0	
	TEMPERATUR						0.020244207
		DINGIN	0	0	0	0	
		PANAS	3	2	1	0.918295834	
		SEDANG	4	2	2	1	
	ANGIN						0.020244207
		TIDAK	4	2	2	1	
		YA	3	4	1	0.918295834	

Dari hasil pada Tabel 2.3 dapat diketahui bahwa atribut dengan *Gain* tertinggi adalah cuaca yaitu sebesar 0.699. Dengan demikian cuaca dapat menjadi *node* cabang dari nilai atribut tinggi. Ada 3 nilai atribut dari cuaca yaitu mendung, hujan dan cerah. dari ketiga nilai atribut tersebut, nilai atribut mendung sudah mengklasifikasikan kasus menjadi 1 yaitu keputusannya Ya dan nilai atribut cerah sudah mengklasifikasikan kasus menjadi satu dengan keputusan Tidak, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut hujan masih perlu dilakukan perhitungan lagi.

Pohon keputusan yang terbentuk sampai tahap ini ditunjukkan pada Gambar 2.5 berikut:



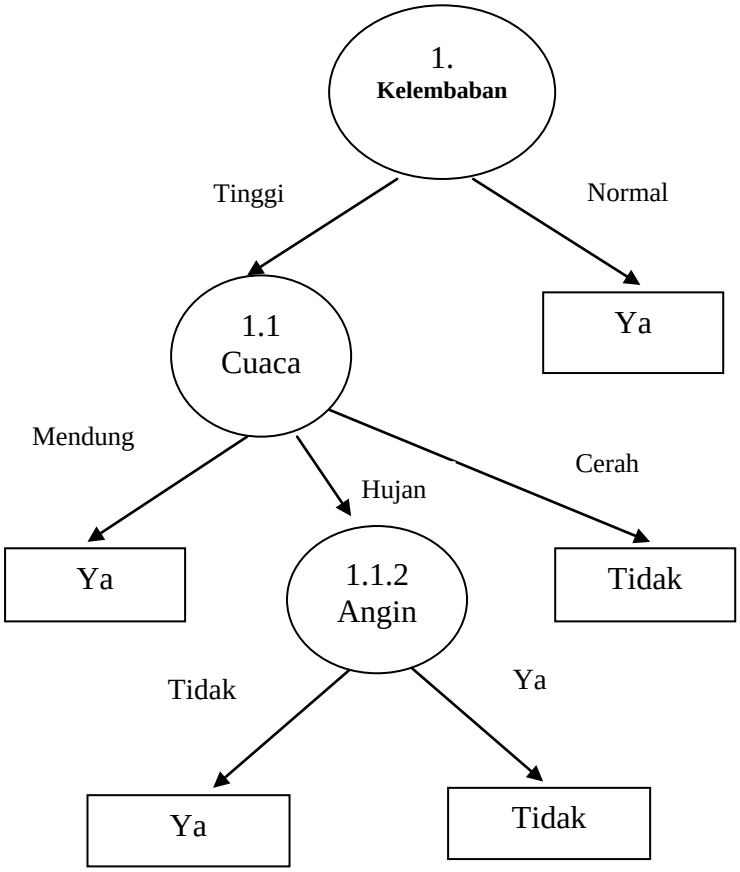
Gambar 2.5 Pohon Keputusan Hasil Perhitungan *Node* 1.1

3. Menghitung jumlah kasus, jumlah kasus untuk keputusan Ya, jumlah kasus untuk keputusan Tidak, dan *Entropy* dari semua kasus dan kasus yang dibagi berdasarkan atribut temperatur dan angin yang dapat menjadi *node* cabang dari nilai atribut hujan. Setelah itu lakukan penghitungan *Gain* untuk masing-masing atribut. Hasil perhitungan ditunjukkan oleh Tabel 2.4

Tabel 2.4 Perhitungan *Node* 1.1.2

Node			Jumlah Kasus (S)	Tidak (S1)	Ya (S2)	*Entropy*	*Gain*
1.1	KELEMBABAN-TINGGI dan CUACA – HUJAN		2	1	1	1	
	TEMPERATUR						0
		DINGIN	0	0	0	0	
		PANAS	0	0	0	0	
		SEDANG	2	1	1	1	
	ANGIN						1
		TIDAK	1	0	1	0	
		YA	1	1	0	0	

Dari hasil pada Tabel 2.4 dapat diketahui bahwa atribut dengan *Gain* tertinggi adalah angin yaitu sebesar 1. Dengan demikian angin dapat menjadi *node* cabang dari nilai atribut hujan. Ada 2 nilai atribut dari angin yaitu Tidak dan Ya. Dari kedua nilai atribut tersebut, nilai atribut Tidak sudah mengklasifikasikan kasus menjadi 1 yaitu keputusannya Ya dan nilai atribut Ya sudah mengklasifikasikan kasus menjadi satu dengan keputusan Tidak, sehingga tidak perlu dilakukan perhitungan lebih lanjut untuk nilai atribut ini. Pohon keputusan yang terbentuk sampai tahap ini ditunjukkan pada Gambar 2.6



Gambar 2.6 Pohon Keputusan Hasil Perhitungan *Node* 1.1.2

Dengan memperhatikan pohon keputusan pada Gambar 2.6 diketahui bahwa semua kasus sudah masuk dalam kelas. Dengan demikian, pohon keputusan pada Gambar 2.6 merupakan pohon keputusan terakhir yang terbentuk.

2.5 Ekstraksi *Rule* dari *Decision Tree*

Pengetahuan yang diperoleh dari *decision tree* dapat direpresentasikan dalam bentuk klasifikasi IF-THEN *rules*. Nilai suatu atribut akan menjadi bagian *anticendent* (bagian IF), sedang daun (*leaf*) dari sebuah *decision tree* akan menjadi bagian *consequent* (THEN). Aturan seperti ini akan menjadi sangat membantu manusia dalam memahami model klasifikasi terutama jika ukuran *decision tree* terlalu besar .

2.6 Support Vector Machine (SVM)

Pattern Recognition merupakan salah satu bidang dalam komputer sains, yang memetakan suatu data ke dalam konsep tertentu yang telah didefinisikan sebelumnya. Konsep tertentu ini disebut *class* atau *category*. Aplikasi pattern recognition sangat luas, di antaranya mengenali suara dalam sistem sekuriti, membaca huruf dalam OCR, mengklasifikasikan penyakit secara otomatis berdasarkan hasil diagnosa kondisi medis pasien dan sebagainya. Berbagai metode dikenal dalam pattern recognition, seperti linear discrimination analysis, hidden markov model hingga metode kecerdasan buatan seperti artificial neural network. Salah satu metode yang akhir-akhir ini banyak mendapat perhatian sebagai *state of the art* dalam pattern recognition adalah Support Vector Machine (SVM). Support Vector Machine (SVM) dikembangkan oleh Boser, Guyon, Vapnik, dan pertama kali dipresentasikan pada tahun 1992 di Annual Workshop on Computational Learning Theory. Konsep dasar SVM sebenarnya merupakan kombinasi harmonis dari teori-teori komputasi yang telah ada puluhan tahun sebelumnya, seperti *margin hyperplane* (Duda & Hart tahun 1973, Cover tahun 1965, Vapnik 1964), kernel diperkenalkan oleh Aronszajn tahun 1950, dan demikian juga dengan konsep-konsep pendukung yang lain. Akan tetapi hingga tahun 1992, belum pernah ada upaya merangkaikan komponen-komponen tersebut. Berbeda dengan strategi neural network yang berusaha mencari hyperplane pemisah antar class, SVM berusaha menemukan hyperplane yang terbaik pada input space. Prinsip dasar SVM adalah linear classifier, dan selanjutnya dikembangkan agar dapat bekerja pada problem non-linear. dengan memasukkan konsep *kernel trick* pada ruang kerja berdimensi tinggi. Perkembangan ini memberikan rangsangan minat penelitian di bidang pattern recognition untuk investigasi potensi kemampuan SVM secara teoritis maupun dari segi aplikasi. Dewasa ini SVM telah berhasil diaplikasikan dalam problema dunia nyata (real-world problems), dan secara umum memberikan solusi yang lebih baik dibandingkan metode konvensional seperti misalnya artificial neural network.

Tulisan ini memperkenalkan konsep dasar SVM, dan membahas aplikasinya di Educational data mining, yang akhir-akhir ini merupakan salah satu bidang yang berkembang cukup pesat.

2.7 Riset-Riset Terkait

Terdapat beberapa riset yang telah dilakukan oleh banyak peneliti berkaitan dengan domain pendidikan, seperti yang akan dijelaskan di bawah ini :

Yu et al. (2010) dalam risetnya menjelaskan mengenai sebuah pendekatan *data mining* dapat diaplikasikan untuk meneliti faktor-faktor yang mempengaruhi tingkat daya ingat mahasiswa. Sunjana (2010a) juga menyampaikan hasil risetnya mengenai aplikasi *data mining* mahasiswa dengan metode klasifikasi *decision tree*. Dengan kesimpulan sebagai berikut :

1. Penentuan data training sangat menentukan tingkat akurasi *tree* yang dibuat.
2. Besar prosentase kebenaran *tree* sangat dipengaruhi oleh data training yang digunakan untuk membangun model *tree* tersebut.
3. Nilai IPK seorang mahasiswa terlihat sangat terpengaruh dengan 9 (Sembilan) mata kuliah yang dianggap pokok.

Quadri dan Kalyankar (2010) juga menjelaskan tentang penggunaan teknik *decision tree* untuk mengidentifikasi berbagai faktor yang meyebabkan mahasiswa melakukan *drop out* untuk meningkatkan kinerja akademik.

She et al. (2010) dalam risetnya menjelaskan mengenai prediksi penurunan sifat sifat manusia secara cepat dan akurat dengan klasifikasi *decision tree* .

Rocha dan Junior (2010) juga dalam risetnya menjelaskan tentang bagaimana mengidentifikasi kecurangan-kecurangan yang terjadi di bidang perbankan menggunakan CRISP-DM dan *decision tree.*

Nogroho, (2008) menjelaskan dalam risetnya mengenai Implementasi *decision tree* berbasis analisis teknikal untuk pembelian dan penjualan saham, menyimpulkan sistem pendukung keputusan *decision tree* yang dibangun berdasarkan analisis teknikal mampu memberikan gambaran saat saham diperdagangkan hanya berdasarkan pergerakan trend. Perdagangan berdasarkan pergerakan trend ini bersifat spekulasi namun cukup mampu memberikan keuntungan.

Sunjana (2010b) menjelaskan dalam risetnya tentang klasifikasi data nasabah sebuah asuransi menggunakan algoritma C 4.5, berikut adalah kesimpulan yang dapat diambil dari data nasabah asuransi setelah dilakukan análisis menggunakan metode algoritma C 4.5:

1. Aplikasi dapat menyimpulkan bahwa rata-rata nasabah memiliki status L dikarenakan pembayaran premi yang melebihi 10% dari penghasilan.
2. Dengan persentase atribut premi_dasar dan penghasilan, maka dapat diketahui rata-rata status nasabah memiliki nilai P atau L.

Bhargavi at al. (2008) menjelaskan dalam risetnya tentang menguraikan pengetahuan menggunakan aturan aturan dengan pendekatan *decision tree.*

Al-Radaideh et al. (2006) menjelaskan dalam risetnya tentang pemanfaatan *data mining* terhadap data mahasiswa menggunakan *decision tree.*

Adeyemo dan Kuye (2006) menjelaskan dalam risetnya untuk memprediksi kinerja mahasiswa di bidang akademik menggunakan algoritma d*ecision tree.*

2.8 Kontribusi Riset

Penelitian ini memberikan kontribusi pada pemahaman kita tentang hubungan data mahasiswa dengan data demografi yaitu data pendukung untuk mengetahui

tingkat keinginan mahasiswa diploma untuk melanjutkan ke jenjang yang lebih tinggi yaitu jenjang sarjana.

Beberapa kemungkinan lain dianggap penting adalah pimpinan perguruan tinggi ataupun yayasan dapat menggunakan informasi yang diberikan dalam mengambil beberapa tindakan untuk meningkatkan keinginan mahasiswa dalam melanjutkan pendidikan nya. Pembuat keputusan bisa menggunakan model prediksi seberapa besar keinginan mahasiswa diploma nya unutk melanjutkan pendidikannya ke jenjang sarjana. Penelitian ini memperkenalkan aplikasi metode klasifikasi *rule decision tree* algoritma C 4.5 dan Support Vektor Machine untuk lembaga pendidikan perguruan tinggi swasta.