BAB II

LANDASAN TEORI

II.1 Data, Informasi, dan Pengetahuan

II.1.1 Data

Data adalah sejumlah fakta, angka, atau teks yang dapat diproses oleh komputer. Sekarang ini, organisasi-organisasi mengumpulkan/ mengakumulasikan data dalam format yang berbeda dalam basis data yang berbeda pula. Ini meliputi :

1. Data operasional atau transaksional, seperti : penjualan, biaya, persediaan, penggajian, dan akuntansi.
2. Data non-operasional, seperti : penjualan industri, data perkiraan, dan data makro ekonomi.
3. Meta data - data tentang data itu sendiri, seperti : desain logika basis data atau definisi kamus data.

II.1.2 Informasi

Pola, asosiasi, atau hubungan diantara semua data-data ini dapat menghasilkan informasi. Sebagai contoh, analisis data transaksi penjualam retail dapat menghasilkan informasi tentang produk mana yang dijual dan kapan produk tersebut dijual.

II.1.3 Pengetahuan

Informasi dapat diubah menjadi pengetahuan tentang pola historis dan tren masa depan/ yang akan datang. Sebagai contoh, summary dari informasi penjualan supermarket retail dapat dianalisis dalam usaha promosi untuk menyediakan pengetahuan tentang tingkah laku beli para konsumen. Selain itu, perusahaan atau retailer dapat menentukan item-item mana yang paling mempengaruhi usaha promosional.

II.2 Basis Data

Basis data adalah himpunan kelompok data yang saling berhubungan yang disimpan secara bersama pada media elektronis tanpa redundansi yang tidak perlu, dengan aturan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan mudah dan cepat. Prisip utamanya adalah pengaturan data dengan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan data kembali (Fathansyah, 2001).

Secara lebih lengkap, pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (objektif) seperti berikut ini :

1. Kecepatan dan kemudahan (Speed)

   Menyimpan, memanipulasi, dan menampilkan kembali data dengan cepat dan mudah.

2. Efisiensi ruang penyimpanan (Space)

   Penekanan terhadap adanya redundansi data.

3. Keakuratan (Accuracy)

   Pemanfaatan pengkodean/pembentukan relasi untuk menghindari ketidak-akuratan.

4. Ketersediaan (Availability)

   Pertumbuhan data yang membutuhkan space yang besar, maka data harus dipilah sebagai data master/referensi/transaksi/historis.

5. Kelengkapan (Completeness)

   Lengkap adalah relatif, sehingga penambahan data, struktur data diakomodasi.

6. Keamanan (Security)

   Penentuan siapa yang berhak mengakses obyek-obyek tertentu dan operasi apa yang diperbolehkan.

7. Kebersamaan pemakai (Sharability)

   Pemakai basis data tidak terbatas pada satu pemakai saja.

II.3 Knowledge Discovery in Database (KDD)

II.3.1 Definisi

Teknologi komputasi dan media penyimpanan telah memungkinkan manusia untuk mengumpulkan dan menyimpan data dari berbagai sumber dengan jangkauan yang amat luas. Fenomena ini terjadi dalam banyak bidang kehidupan, seperti bisnis, perbankan, pemasaran, produksi, pengetahuan, dan sebagainya. Dalam bidang pengetahuan misalnya, berbagai teknologi memungkinkan pengambilan data yang dilakukan secara kontinu hingga dalam jumlah bertera-tera (1012) byte. Salah satu contohnya adalah Sistem Observasi Bumi milik NASA yang mampu mengirimkan berbagai jenis data berkaitan dengan objek-objek yang diamatinya hingga berpuluh-puluh gigabyte setiap jamnya.

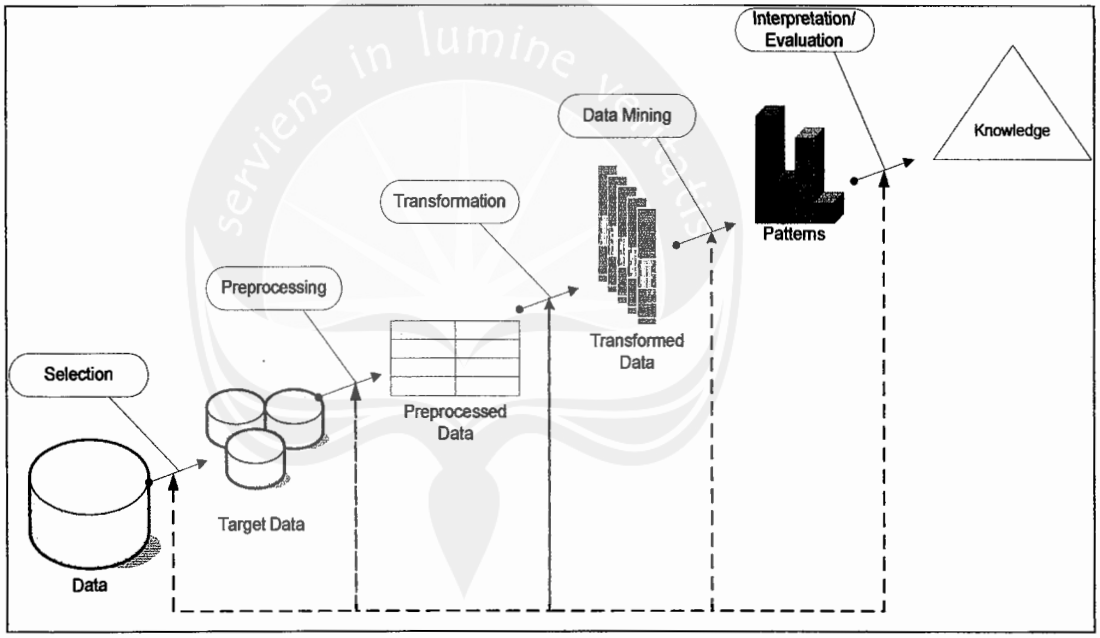
Meskipun teknologi basis data modern telah menghasilkan media penyimpanan yang ekonomis bagi aliran data yang sangat besar ini, namun teknologi untuk membantu kita menganalisis, memahami, atau bahkan memvisualisasikannya belumlah tersedia. Hal inilah yang melatar-belakangi dikembangkannya konsep Pengambilan Pengetahuan dari Basis data (PPB) atau *Knowledge Discovery in Database (KDD)*. Penggunaan istilah Pengambilan Pengetahuan dari Basis data (PPB) atau *Knowledge Discovery in Database (KDD)* dan istilah Penambangan Data (*Data Mining*) seringkali dianggap memiliki arti yang sama, namun beberapa tahun terakhir ini KDD digunakan untuk menunjuk pada suatu proses yang terdiri dari beberapa banyak tahap, sementara *data mining* hanya merupakan salah satu tahap di dalamnya. Definisi sederhana dari KDD adalah pencarian pengetahuan dalam basis data dalam proses identifikasi pola-pola yang valid, berpotensi manfaat, dan dapat dipahami secara mudah. KDD merupakan suatu proses yang meliputi beberapa tahap yang berbeda. Input dari proses ini adalah data dan output yang dihasilkan adalah informasi yang diharapkan dapat berguna bagi user.

Sedangkan *data mining*, pada dasarnya merupakan suatu perolehan pengetahuan dari data atau dapat diartikan sebagai ekstraksi informasi atau pola yang menarik (tidak sepele, implisit, tak-diketahui sebelumnya, mungkin bermanfaat) dari data didalam basis data yang besar. Sebagai cabang ilmu baru di bidang komputer, cukup banyak penerapan yang dapat dilakukan oleh *data mining*. Apalagi ditunjang kekayaan dan keaneka-ragaman berbagai bidang ilmu (kecerdasan

buatan, basis data, statistik, pemodelan matematika, pengolahan citra, dsb), membuat aplikasi, tren, dan penerapan *data mining* menjadi makin luas.

II.3.2 Proses dalam KDD

Proses yang ada dalam KDD dapat digambarkan seperti dibawah ini, dan terdiri dari tahapan-tahapan berikut :



Gambar 2.1 Proses dalam KDD

1. Selection (Pemilihan).

   Data yang diperlukan untuk proses data mining dapat diperoleh dari banyak sumber data yang berbeda-beda dan heterogen. Tahap pertama ini mengambil data dari basis data, file-file, dan sumber non-elektronis yang bermacam-macam.

2. Preprocessing (Pemrosesan awal).

   Data yang akan dipakai dalam proses mungkin memiliki kesalahan ataupun ada data yang hilang. Pada tahap ini terdapat beberapa aktifitas berbeda yang dilakukan. Data yang mengandung kesalahan (error) dapat dibenarkan ataupun dihilangkan, namun data yang hilang harus disediakan ataupun diperkirakan.

3. Transformation (Transformasi).

   Dari dari sumber yang berbeda-beda harus diubah ke dalam suatu format/bentuk yang umum untuk diproses. Beberapa jenis data mungkin dapat diubah kedalam format yang lebih bermanfaat. Reduksi data dapat digunakan untuk mengurangi banyaknya nilai data yang mungkin yang diketahui.

4. Data Mining (Penambangan data).

   Berdasarkan pada tugas data mining yang dikerjakan, tahap ini mengaplikasikan algoritma kedalam data yang telah ditransformasikan untuk mendapatkan hasil yang diharapkan.

5. Interpretation and Evaluation (Interpretasi dan evaluasi).

   Bagaimana hasil penambangan data disajikan kepada user merupakan hal yang sangat penting, karena kegunaan hasil yang diperoleh tergantung pada hal ini. Strategi visualisasi dan GUI yang bermacam-macam digunakan pada tahap terakhir ini.

II.4 Penambangan Data (Data Mining)

II.4.1 Definisi

Penambangan data menghadirkan suatu proses yang dikembangkan untuk menguji sejumlah data besar yang secara rutin dikumpulkan. Istilah ini juga mengacu pada suatu koleksi tool yang digunakan untuk melaksanakan proses tersebut. Penambangan data kebanyakan digunakan pada area dimana data dikumpulkan, seperti pemasaran, kesehatan, komunikasi, dan lain-lain. Sebagai contoh, toko eceran secara rutin menggunakan tool penambangan data untuk mempelajari kebiasaan belanja dari para pelanggan.

Data mining merupakan proses mengidentifikasikan kebenaran, ide baru, potensial berguna, dan pada akhirnya dapat dipahami, dan sebagai dasar pengetahuan dari basis data untuk membuat keputusan bisnis yang penting (G.Piatetsky-Shapiro, 2004). Secara teknis, data mining merupakan proses menemukan korelasi/pola data dari banyak field pada basis data relasional yang berukuran besar.

Ketika teknologi informasi skala besar telah melibatkan transaksi terpisah dari sistem analisis, *data mining* menyediakan *link* antar keduanya. Software penambangan data mengasanalisis hubungan dan pola pada transaksi yang tersimpan dalam basis data pada query user yang *open-ended*. Umumnya, ada empat jenis hubungan yang digunakan, antara lain :

1. *Classes*

   Data yang tersimpan digunakan untuk menempatkan data dalam kelompok yang telah ditetapkan sebelumnya (*predetermined group*). Sebagai

contoh, suatu cabang restaurant dapat menambang data pengeluaran konsumen untuk menentukan kapan konsumen akan berkunjung dan pesanan apa yang biasanya dipesan. Informasi ini dapat digunakan untuk meningkatkan trafik dengan menyediakan menu spesial/khusus harian.

2. *Clusters*

   Item-item data dikelompokkan berdasarkan hubungan logika atau pilihan pengguna. Sebagai contoh, data dapat ditambang untuk mengidentifikasi segmen pasar atau afinitas konsumen.

3. *Associations*

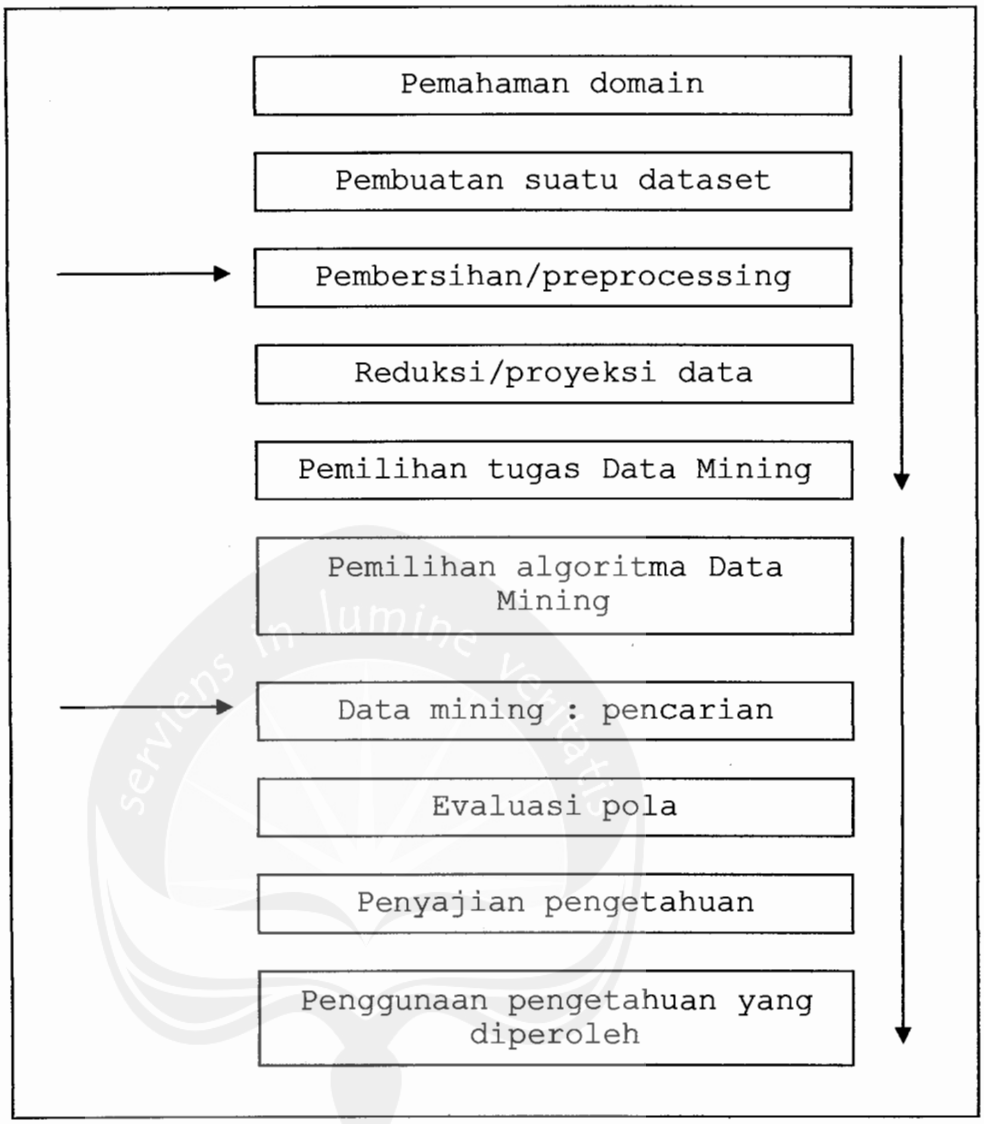
   Data dapat ditambang untuk mengidentifikasi asosiasi. Contoh bir-popok merupakan suatu contoh penambangan asosiatif.

4. *Sequential pattterns*

   Data ditambang untuk mengantisipasi pola tingkah laku dan tren. Sebagai contoh, retailer peralatan outdoor dapat memperkirakan tas punggung seperti apa yang akan dibeli berdasarkan pengeluaran konsumen terhadap kantong tidur (*sleeping bag*) dan sepatu *hiking.*

II.4.2 Tahapan Proses Data Mining

Mengacu pada proses Knowledge Discovery in Database (KDD), tahap-tahap proses data mining dijelaskan sebagai berikut :



Gambar 2.2 Tahapan proses Data Mining

II.4.2.1 Pemahaman Domain

Pada proses pemahaman domain, dilakukan pemetaan pertanyaaan bisnis dan objektif projek kedalam perumusan masalah dan objektif yang dapat diselesaikan dengan teknik *data mining*. Untuk mengidentifikasi pertanyaan-pertanyaan bisnis atau tujuan (*goal*) yang akan dicapai dapat dilakukan

melalui kolaborasi antara analisis data, analisis bisnis, dan expert domain.

Dalam pemahaman domain ditentukan output apa yang ingin didapatkan dari data mining. Hal ini akan membantu dalam menentukan teknik visualisasi dan *data mining* yang sesuai dengan goal yang akan dicapai. Proses ini melibatkan beberapa hal, yaitu :

(a) Pemahaman batasan goal.
(b) Menentukan goal yang tepat.
(c) Menentukan elemen yang tepat untuk mendapatkan goal.
(d) Pemahaman data.

II.4.2.2 Pembuatan Data Set Target

Dalam pembuatan data set target, ditentukan data-data apa saja yang diperlukan untuk mendapatkan goal. Data-data dapat diperoleh dari tool pengoleksi data. Data-data tersebut dikumpulkan dan disimpan pada suatu sistem. Berikut ini merupakan beberapa cara untuk mendapatkan sumber data yang dapat digunakan untuk data mining :

(a) Data warehouse.
(b) Relational transaction/PC-Based Databases.
(c) Data conversion utility.
(d) Query tools.
(e) Flat files.

II.4.2.3 Pembersihan/Preprocessing Data

Beberapa issue pada tahap preprocessing data adalah sebagai berikut :

(a) Konsistensi data.

Nilai kolom yang mengacu pada nilai yang sama tetapi sistem membaca sebagai nilai yang berbeda karena isi masukan berbeda. Contoh : kolom yang berisi jus minuman ringan dapat berisi Pepsi, Pepsi Cola, atau Cola.

(b) *Stale data.*

Nilai yang tidak lagi dapat dipakai karena data tidak lagi merujuk pada keadaan yang sebenarnya. Contoh : *mailing list* yang secara kontinu di-*update*, karena orang-orang cenderung berpindah/mengubah alamat e-mailnya sehingga alamat yang lama tidak lagi benar.

(c) *Typographical error.*

Pemasukan nilai data dengan ejaan yang salah.

(d) *Data redundant.*

Pemasukan data dengan nilai yang sama pada suatu kolom.

(e) *Missing values.*

Missing values atau data hilang adalah adanya sel-sel kosong pada satu atau beberapa variabel, karena hal-hal sebagai berikut :

- Data tidak diberikan.
- Data sulit didapatkan.
- Informasi tidak ada.

II.4.2.4 Reduksi Data

Reduksi/proyeksi data merupakan akhir dari proses persiapan data, dimana pada proses ini dilakukan pencarian *feature-feature* yang berguna, dimensi data dan mengatasi representasi data yang berbeda. Dari proses ini didapatkan data-data yang siap untuk di-*mining*.

II.4.2.5 Pemilihan Tugas Data Mining

Pemilihan tugas data mining didasarkan pada domain problem yang akan diselesaikan sesuai dengan yang telah didefinisikan pada langkah pertama. Berikut ini merupakan beberapa tugas data mining :

(a) Asosiasi.

Mencari asosiasi dan korelasi antar setiap item data pada dataset.

(b) Klasifikasi.

Mengklasifikasi (membuat suatu model) berdasarkan himpunan pelatihan dan nilai-nilai (label kelas) dalam suatu atribut klasifikasi dan menggunakannya didalam mengklasifikasikan data baru. Model itu sendiri bisa berupa aturan "if-then", berupa pohon keputusan, formula matematis, atau jaringan saraf (*neural network*).

(c) Klasterisasi.

Proses pembagian populasi objek kedalam subgroup yang memiliki kemiripan sifat.

II.4.2.6 Pemilihan Algoritma Data Mining

Untuk pemilihan algoritma *data mining*, hal-hal yang perlu dipertimbangkan adalah tipe data dan

tugas data mining yang akan diselesaikan. Untuk data karyawan ini, digunakan algoritma klasifikasi dengan menggunakan metode naive Bayesian. Algoritma ini dipilih dengan mempertimbangkan bahwa algoritma ini relatif lebih mudah digunakan dan proses pengklasifikasian ini akan digunakan untuk mengklasifikasikan tuple baru yang ada, apakah termasuk dalam kelas prestasi yang baik, cukup, ataupun kurang. Kelas-kelas prestasi ini akan dapat digunakan untuk melihat efektivitas kerja karyawan pada perusahaan dan dapat digunakan sebagai bahan pertimbangan bagi perusahaan untuk memberikan bonus kepada karyawan.

II.4.2.7 Evaluasi Pola

Tahap ini digunakan untuk mengenali pola yang benar-benar menarik yang menggambarkan pengetahuan berdasarkan beberapa pengukuran kemenarikan.

II.4.2.8 Penyajian Pengetahuan

Pada tahap penyajian pengetahuan ini, berbagai macam teknik visualisasi dan representasi pengetahuan digunakan untuk menyajikan pengetahuan yang telah digali kepada user.

II.5 Klasifikasi

II.5.1 Definisi

Metode klasifikasi mungkin merupakan salah satu teknik penambangan data yang paling umum dan sering digunakan. Beberapa contoh aplikasi metode ini antara

lain pengenalan pola dan gambar, diagnosa medis, aplikasi pendeteksian kesalahan dalam industri, dan pengelompokan finansial tren pasar. Perkiraan dan prediksi dapat dipandang sebagai jenis-jenis klasifikasi. Ketika seseorang memperkirakan umur kita atau menebak jumlah kelereng dalam toples, sebenarnya ini termasuk dalam klasifikasi. Prediksi dapat diartikan dengan mengelompokkan sebuah nilai atribut ke dalam suatu kumpulan kelas yang ada/memungkinkan. Selain itu, juga sering dipandang sebagai meramalkan suatu nilai yang kontinu, sementara klasifikasi meramalkan suatu nilai diskret.

Semua pendekatan untuk menampilkan klasifikasi mengambil beberapa pengetahuan dari data. Seringkali kumpulan *training data* (data percobaan) digunakan untuk mengembangkan parameter spesifik yang diperlukan oleh teknik yang digunakan. Data percobaan (*training data*) terdiri dari contoh data masukan/input dan juga penempatan klasifikasi data. Permasalahan klasifikasi dinyatakan dengan definisi yang ditunjukkan sebagai berikut :

> Diberikan sebuah basis data $D = \{t_1, t_2, \ldots, t_n\}$, dengan t adalah tuple-tuple (*items, records*) dan sekumpulan kelas $C = (C_1, \ldots, C_m)$, permasalahan klasifikasinya adalah untuk mendefinisikan sebuah pemetaan $f : D \rightarrow C$ dimana setiap t_i dinyatakan sebagai satu kelas. Sebuah kelas, C_j, terdiri tepat hanya tuple-tuple yang dipetakan kepadanya; yaitu $C_j = \{t_i \mid f(t_i) = C_j , 1 \leq i \leq n,$ dan $t_i \in D\}$.

Definisi tersebut memandang klasifikasi sebagai suatu pemetaan dari basis data ke dalam suatu kumpulan

kelas. Dengan catatan bahwa kelas-kelas tersebut telah didefinisikan sebelumnya, tidak *overlap*, dan mencakup/membagi seluruh basis data. Setiap tuple dalam basis data memiliki tepat satu kelas. Kelas-kelas yang ada dalam permasalahan klasifikasi merupakan kelas-kelas yang ekuivalen. Dalam kenyataannya, permasalahan tersebut biasanya diimplementasikan dalam dua fase :

1. Menciptakan suatu model yang spesifik dengan mengevaluasi data percobaan. Pada langkah ini, data percobaan digunakan sebagai input (termasuk klasifikasi yang didefinisikan untuk setiap tuple) dan suatu definisi dari model yang dikembangkan sebagai outputnya. Model yang tercipta mengklasifikasikan data percobaan seakurat mungkin.

2. Mengaplikasikan model yang telah dikembangkan pada langkah 1 dengan mengelompokkan tuple-tuple dari basis data tujuan.

Ada 3 metode dasar yang digunakan untuk menyelesaikan permasalahan klasifikasi, yaitu :

a. Menspesifikasikan batasan (*specifying boundaries*). Klasifikasi di sini dinyatakan dengan membagi ruang input dari tuple basis data yang potensial ke dalam wilayah-wilayah, dimana setiap wilayah diasosiasikan dengan satu kelas.

b. Menggunakan distribusi probabilitas *(using probability distributions*). Untuk setiap kelas

yang diberikan, C_j, $P(t_i \mid C_j)$ adalah nilai distribusi probabilitas untuk kelas yang dievaluasi pada satu titik, t_i. Jika probabilitas suatu kejadian untuk setiap kelas, P(Cj) diketahui, maka $P(C_j) P(t_i \mid C_j)$ digunakan untuk memperkirakan probabilitas t_i berada pada kelas C_j.

c. Menggunakan probabilitas posterior *(using posterior probabilities)*. Diberikan suatu nilai data t_i, kita hendak menentukan probabilitas bahwa t_i termasuk dalam kelas C_j. Ini ditunjukkan dengan $P(C_j \mid t_i)$ dan disebut sebagai probabilitas posterior. Pendekatan satu klasifikasi dapat digunakan untuk menentukan probabilitas posterior untuk masing-masing kelas dan kemudian menempatkan t_i pada kelas dengan probabilitas tertinggi.

Suatu pokok persoalan utama yang diasosiasikan dengan klasifikasi adalah ketidaksesuaian (*overfitting*). Jika strategi klasifikasi yang digunakan tepat sesuai dengan data percobaan, hal tersebut tidak dapat diaplikasikan pada populasi data yang lebih luas. Sebagai contoh, seharusnya data percobaan memiliki kesalahan atau data yang *noisy*. Namun pada kasus ini tentunya, penyesuaian data tidak diharapkan sama sekali.

II.5.2 Persoalan dalam klasifikasi

Pokok persoalan yang ada dalam klasifikasi antara lain sebagai berikut :

(a) Data yang hilang (missing data).

Nilai data yang hilang dapat menimbulkan masalah pada saat fase percobaan dan proses klasifikasi itu sendiri. Nilai yang hilang dalam data percobaan harus ditangani dan dapat menghasilkan hasil yang tidak akurat. Data yang hilang dalam sebuah tuple yang akan diklasifikasikan harus sudah ditangani pada saat menghasilkan pola klasifikasi. Ada banyak pendekatan untuk menangani data yang hilang, diantaranya :

- Abaikan data yang hilang.
- Asumsikan suatu nilai untuk data yang hilang. Hal ini dapat ditentukan dengan menggunakan beberapa metode untuk memprediksikan nilai apa yang dapat digunakan.
- Asumsikan suatu nilai tertentu untuk data yang hilang. Ini berarti bahwa nilai dari data yang hilang diambil untuk menjadi suatu nilai yang spesifik untuk keseluruhan nilai itu sendiri.

Terdapat kesamaan antara data yang hilang pada permasalahan klasifikasi dengan *nulls* pada basis data tradisional.

(b) Pengukuran kinerja (measuring performance).

Kinerja dari algoritma klasifikasi biasanya dilihat dengan mengevaluasi keakuratan klasifikasi tersebut. Namun bagaimanapun juga, dikarenakan klasifikasi ini seringkali merupakan masalah yang tidak jelas (*fuzzy problem*), maka jawaban yang tepat tergantung pada penggunanya. Pendekatan evaluasi dengan algoritma tradisional seperti menetukan ruang dan waktu *overhead* dapat digunakan, namun pendekatan ini biasanya bersifat sekunder.

Akurasi klasifikasi biasanya diperhitungkan dengan menentukan persentase dari tuple-tuple yang ditempatkan pada kelas yang tepat. Hal ini mengabaikan fakta bahwa mungkin saja juga terdapat pengeluaran/biaya yang diasosiasikan dengan suatu penempatan yang tidak tepat pada kelas yang salah. Ini juga mungkin harus ditetapkan.

Kita dapat melihat kinerja dari klasifikasi sebanyak yang telah dilakukan dengan sistem pencarian informasi (*retrieval information system*). Hanya dengan dua kelas, ada empat hasil yang memungkinkan dengan klasifikasi, seperti yang terlihat di bawah ini.

RET REL	NOTRET REL
RET NOTREL	NOTRET NOTREL

(a) Pencarian informasi

Assigned Class A in Class A	Assigned Class B in Class A
Assigned Class A in Class B	Assigned Class B in Class B

(b) Klasifikasi ke dalam kelas A

True positive	False negative
False positive	True negative

(c) Prediksi kelas

Gambar 2.3 Perbandingan kinerja klasifikasi terhadap pencarian informasi

Kuadran kiri atas dan kuadran kanan bawah pada gambar 2(a) dan 2(b) menunjukkan aksi yang benar. Dan dua sisanya menunjukkan aksi yang salah. Kinerja klasifikasi dapat ditentukan dengan mengasosiasikan biaya dengan masing-masing kuadran. Namun, hal ini dapat menjadi sulit dilakukan karena jumlah biaya total yang diperlukan adalah m^2, dimana m adalah jumlah kelas.

Diberikan suatu kelas yang spesifik, C_j, dan sebuah tuple basis data, t_i, dimana tuple tersebut dapat/tidak dapat ditempatkan pada kelas tersebut meskipun keanggotaan aktual dari tuple tersebut termasuk/tidak termasuk dalam kelas tersebut. Hal ini memberikan lagi kepada kita empat kuadran seperti yang ditunjukkan pada gambar 2(c), yang dapat dideskripsikan sebagai berikut :

- True positive (TP) :
  t_i diperkirakan berada pada kelas C_j dan benar-benar berada pada kelas tersebut.
- False positive (FP) :

t_i diperkirakan berada pada kelas C_j dan tidak benar-benar berada pada kelas tersebut.

- True negative (TN) :
  t_i tidak diperkirakan berada pada kelas C_j dan tidak benar-benar berada pada kelas tersebut.
- False negative (FN) :
  t_i tidak diperkirakan berada pada kelas C_j dan benar-benar berada pada kelas tersebut.

II.6 Klasifikasi Bayesian

Dengan kesimpulan statistikal, informasi tentang suatu distribusi data dapat disimpulkan dengan memeriksa data yang menyertai distribusi tersebut. Diberikan suatu kumpulan data $X = \{x_1, x_2, \ldots, x_n\}$, suatu permasalahan penambangan data adalah untuk menemukan properti-properti distribusi darimana kumpulan data tersebut berasal. Aturan Bayes adalah suatu teknik untuk memperkirakan kemiripan dari sebuah properti yang diberikan kepada suatu kumpulan data sebagai keterangan atau input. Diharuskan terjadinya salah satu dari hipotesis h_1 atau hipotesis h_2, namun tidak keduanya. Dan diharuskan juga bahwa x_i adalah suatu kejadian yang dapat diamati. Teorema Bayes atau aturan Bayes adalah sebagai berikut :

$$P(h_1 \mid x_i) = \frac{P(x_i \mid h_1)P(h_1)}{P(x_i \mid h_1)P(h_1) + P(x_i \mid h_2)P(h_2)}$$

$P(h_1 \mid x_i)$ disebut sebagai probabilitas posterior, sedangkan $P(h_1)$ adalah probabilitas prior yang diasosiasikan dengan hipotesis h_1. $P(x_i)$ adalah probabilitas terjadinya nilai data x_i dan $P(x_i \mid h_1)$ adalah probabilitas kondisional dimana tuple memenuhi hipotesis.

Untuk m hipotesis berbeda, kita memiliki probabilitas $P (x_i)$:

$$P(x_i) = \sum_{j=1}^{m} P(x_i \mid h_j) P(h_j)$$

Jadi, kita memiliki probabilitas $P(x_i \mid h_1)$:

$$P(h_1 \mid x_i) = \frac{P(x_i \mid h_1) P(h_1)}{P(x_i)}$$

Aturan Bayes memungkinkan kita untuk menentukan probabilitas dari hipotesis yang diberikan oleh suatu nilai data, $P(h_j \mid x_i)$. Yang dibicarakan disini adalah tuple meskipun pada kenyataannya setiap x_i dapat berupa suatu nilai atribut atau label data lainnya. Setiap h_i dapat berupa suatu nilai atribut (seperti suatu jangkauan/range), atau bahkan suatu kombinasi nilai-nilai atribut.

Dengan mengasumsikan bahwa kontribusi oleh semua atribut adalah independen dan bahwa setiap atribut mengkontribusikan secara bersama-sama pada permasalahan klasifikasi, suatu pola klasifikasi sederhana yang disebut *naive Bayes classification*, yang didasarkan pada aturan Bayes tentang probabilitas kondisional, telah dikemukakan. Dengan menganalisis kontribusi dari setiap atribut independen, suatu probabilitas kondisional dapat ditentukan. Suatu klasifikasi dibuat dengan mengkombinasikan pengaruh dari atribut-atribut

berbeda yang dimiliki dalam prediksi yang akan dibuat. Pendekatan ini disebut "*naive*" karena pendekatan ini mengambil keindependensian diantara berbagai nilai atribut. Diberikan suatu nilai data x_i, probabilitas suatu tuple yang berhubungan, t_i, termasuk dalam kelas C_j dideskripsikan dengan $P(C_j \mid x_i)$. Data percobaan dapat digunakan untuk menentukan $P(x_i)$, $P(x_i \mid C_j)$, dan $P(C_j)$. Dari nilai-nilai ini, teorema Bayes memungkinkan kita untuk memperkirakan probabilitas posterior $P(C_j \mid x_i)$ dan kemudian $P(C_j \mid t_i)$.

Ketika diberikan suatu kumpulan data, pertama-tama teorema Bayes memperkirakan probabilitas prior $P(C_j)$ untuk setiap kelas dengan menghitung seberapa sering setiap kelas muncul dalam data percobaan. Untuk setiap atribut, x_i, banyaknya kemunculan dari setiap nilai atribut x_i dapat dihitung untuk menentukan $P(x_i)$. Demikian pula dengan probabilitas $P(x_i \mid C_j)$ dapat diperkirakan dengan menghitung seberapa sering setiap nilai muncul di dalam kelas pada data percobaan. Sebuah tuple dalam data percobaan mungkin dapat memiliki banyak atribut yang berbeda, dan masing-masing dengan banyak nilai. Hal ini harus dilakukan untuk semua atribut dan semua nilai dari atribut. Kemudian kita menggunakan probabilitas yang telah kita peroleh ini pada saat sebuah tuple baru akan diklasifikasikan. Itulah sebabnya mengapa *naive Bayes classification* dapat dipandang baik sebagai suatu jenis algoritma deskriptif maupun suatu jenis algoritma prediktif. Probabilitas-probabilitas yang ada ini bersifat deskriptif yang kemudian digunakan untuk memperkirakan keanggotaan kelas untuk tuple tujuan.

Ketika mengklasifikasikan tuple tujuan, probabilitas kondisional dan probabilitas prior yang dihasilkan dari kumpulan data percobaan digunakan untuk membuat suatu prediksi/perkiraan. Hal ini dilakukan dengan menggabungkan sifat-sifat dari nilai-nilai atribut yang berbeda dari tuple tersebut. Misalkan tuple t_i memiliki p nilai atribut independen $\{xi_1, xi_2, ..., xi_p\}$. Dari tahap deskriptif, kita tahu $P(x_{ik} \mid C_j)$ untuk setiap kelas Cj dan atribut x_{ik}, kemudian kita memperkirakan $P(t_i \mid C_j)$ dengan :

$$P(t_i \mid C_j) = \prod_{k=1}^{p} P(x_{ik} \mid C_j)$$

Dalam algoritma pada titik ini, kemudian kita memerlukan probabilitas prior $P(C_j)$ untuk masing-masing kelas dan probabilitas kondisional $P(t_i \mid C_j)$. Untuk menghitung $P(t_i)$, kita dapat memperkirakan kemiripan t_i dalam setiap kelas. Hal ini dapat dilakukan dengan menemukan kemiripan pada kelas dimana t_i berada dan menambahkan seluruh nilai-nilai ini. Probabilitas bahwa berada di sebuah kelas adalah hasil dari probabilitas kondisional untuk setiap nilai atribut. Kemudian probabilitas posterior $P(C_j \mid t_i)$ untuk setiap kelas dapat ditemukan. Kelas dengan probabilitas tertinggi adalah yang terpilih untuk tuple.

Pendekatan *naive Bayes* memiliki beberapa keuntungan. Yang pertama ialah mudah untuk digunakan. Kedua, tidak seperti pendekatan klasifikasi yang lain, dalam klasifikasi Bayes ini hanya diperlukan satu kali pemeriksaan terhadap data percobaan. Pendekatan Bayes ini dapat dengan mudah menangani nilai-nilai yang hilang dengan hanya mengabaikan probabilitas tersebut

ketika menghitung kesamaan keanggotaan dalam setiap kelas. Dalam kasus dimana terdapat relasi yang sederhana, teknik ini seringkali menghasilkan hasil-hasil yang bagus.

Meskipun pendekatan *naive Bayes* ini seringkali digunakan, namun hasil yang dihasilkan belum tentu memuaskan. Hal ini dikarenakan artibut-atribut yang ada biasanya tidak independen. Kita dapat menggunakan suatu subset atribut dengan mengabaikan atribut yang dependen dengan atribut yang lain. Teknik ini tidak dapat menangani data yang kontinu/berkesinambungan. Memecah-mecah suatu nilai yang kontinu kedalam range-range tertentu dapat dilakukan untuk memecahkan persoalan ini, namun pembagian domain ke dalam range-range bukanlah hal yang mudah untuk dilakukan, dan bagaimana hal ini dilakukan pasti akan dapat mempengaruhi hasilnya.

Untuk lebih memperjelas pemahaman tentang algoritma naive Bayesian ini, maka perhatikan contoh sebagai berikut :

Dilakukan pengukuran tinggi badan terhadap sekumpulan orang dengan pengelompokan secara sederhana sebagai berikut ini :

2 m <= tinggi badan	tinggi
1.7 m < tinggi badan < 2 m	sedang
Tinggi badan <= 1.7 m	pendek

Dan dari hasil pengukuran tinggi badan 15 orang diperoleh data sebagai berikut :

Nama	Jns kelamin	Tinggi badan	Hasil pengukuran
Cindy	F	1.6 m	pendek

Jeremiah	M	2 m	tinggi
Obelia	F	1.9 m	sedang
Mischa	F	1.88 m	sedang
Nadia	F	1.7 m	pendek
Billy	M	1.85 m	sedang
Vera	F	1.6 m	pendek
Dave	M	1.7 m	pendek
Kennard	M	2.2 m	tinggi
Steven	M	2.1 m	tinggi
Debbie	F	1.8 m	sedang
Jason	M	1.95 m	sedang
Kim	F	1.9 m	sedang
Amy	F	1.8 m	sedang
Tasha	F	1.75 m	sedang

Pada tabel diatas, ada 4 tuple yang dikelompokkan sebagai pendek, 8 tuple sebagai sedang, dan 3 tuple sebagai tinggi. Untuk melakukan pengklasifikasian, kita membagi nilai atribut tinggi badan menjadi enam range, yaitu :

(0 - 1.6), (1.6 - 1.7), (1.7 - 1.8), (1.8 - 1.9), (1.9 - 2.0), (2.0 - ∞)

Dengan training data ini, kita memperkirakan nilai probabilitas proir :

P(pendek) = 4/15 = 0.267

P(sedang) = 8/15 = 0.533

P(tinggi) = 3/15 = 0.2

Kemudian dengan menghitung jumlah tuple dan probabilitas subsequent yang diasosiasikan dengan nilai atribut, maka :

Atribut	Nilai	Count			Probabilitas		
		Pendek	Sedang	Tinggi	Pendek	Sedang	Tinggi
Jenis kelamin	M	1	2	3	1/4	2/8	3/3
	F	3	6	0	3/4	6/8	0/3
Tinggi badan	0 - 1.6	2	0	0	2/4	0	0
	1.6 - 1.7	2	0	0	2/4	0	0
	1.7 - 1.8	0	3	0	0	3/8	0
	1.8 - 1.9	0	4	0	0	4/8	0
	1.9 - 2.0	0	1	1	0	1/8	1/3
	2.0 - ∞	0	0	2	0	0	2/3

Dengan menggunakan nilai-nilai tersebut, maka tuple baru akan diklasifikasikan. Sebagai contoh, tuple baru yang akan diklasifikasikan adalah t = (Adam, M, 1.95 m). Dengan menggunakan nilai-nilai diatas dan nilai probabilitas yang diasosiasikan dengan atribut jenis kelamin dan tinggi badan, maka diperoleh perkiraan sebagai berikut :

P(t | pendek) = 1/4 * 0 = 0

P(t | sedang) = 2/8 * 1/8 = 0.031

P(t | tinggi) = 3/3 * 1/3 = 0.333

Dengan mengkombinasikan nilai-nilai tersebut, maka didapat :

Kemungkinan pendek = 0 * 0.267 = 0

Kemungkinan sedang = 0.031 * 0.533 = 0.0166

Kemungkinan tinggi = 0.333 * 0.2 = 0.066

Nilai P(t) diperkirakan dengan menjumlahkan masing-masing nilai kemungkinan tersebut :

P(t) = 0 + 0.0166 + 0.066 = 0.0826

Akhirnya, nilai probabilitas aktual dari setiap event diperoleh dari :

$$P(\text{pendek} \mid t) = \frac{0 * 0.0267}{0.0826} = 0$$

$$P(\text{sedang} \mid t) = \frac{0.031 * 0.533}{0.0826} = 0.2$$

$$P(\text{tinggi} \mid t) = \frac{0.333 * 0.2}{0.0826} = 0.799.$$

Berdasarkan nilai probabilitas yang ada, tuple yang baru tersebut diklasifikasikan kedalam kelas tinggi, karena kelas tersebut memiliki nilai probabilitas tertinggi.