GRADUATION PREDICTION OF GUNADARMA UNIVERSITY STUDENTS USING ALGORITHM AND NAIVE BAYES C4.5 ALGORITHM

Marselina Silvia Suhartinah, Ernastuti

Undergraduate Program, Faculty of Industrial Engineering, 2010

Gunadarma University

http://www.gunadarma.ac.id

Keywords: Gunadarma University, Data Mining, Naive Bayes, Decision Tree, C4.5

ABSTRACT

Gunadarma University is one of the private universities in Indonesia, which has a rather large number of students. This can be seen from the increasing number of prospective new students in each year ajaran.Untuk know the graduation rate of students in one school year can be made a prediction based on student data on the level or the first academic year. Data mining, also called knowledge discovery in databases (KDD), is an activity includes the collection, use historical data to discover regularities, patterns or relationships in large data sets. Naive Bayes Classifiers (NBC) is a simple probability classifier are applying Bayes theorem with the assumption of independence (independent) high. The advantages of using NBC is that this method only requires the amount of training data (training data) are small to estimate the parameters needed in the process of classification. Decision tree method to change the fact that a very large decision tree representing a rule. C4.5 is an algorithm that is widely known and used for data classification that has a numeric attributes and categorical. The results of the classification process in the form of rules can be used to predict the value of discrete type attribute of the new record.

PENDAHULUAN

Latar Belakang Masalah

Universitas Gunadarma merupakan salah satu perguruan tinggi swasta di Indonesia yang memiliki jumlah mahasiswa yang cukup banyak. Hal ini terlihat dari peningkatan jumlah calon mahasiswa baru pada setiap tahun ajaran. Namun, kendala yang sering terjadi adalah banyaknya mahasiswa yang tidak lulus sesuai dengan waktu studi yang telah ditetapkan. Untuk mengetahui tingkat kelulusan mahasiswa dalam satu tahun ajaran dapat dilakukan suatu prediksi berdasarkan data-data mahasiswa pada tingkat atau tahun ajaran pertama. Beberapa faktor yang mempangaruhi prediksi kelulusan mahasiswa yang sesuai dengan waktu studi, diantaranya : NEM SMA, IP semester 1 dan IP semester 2, IPK DNU semester 1 dana 2, gaji orang tua dan pekerjaan orang tua.

Perkembangan teknologi saat ini semakin meningkat dan memberi pengaruh yang besar hampir disetiap sektor kehidupan dan kenegaraan. Proses globalisasi yang terjadi disetiap negara di dunia saat ini juga mendukung perkembangan dan penggunaan teknologi.

Kebutuhan akan informasi pada saat ini semakin meningkat bersamaan dengan perkembangan teknologi yang semakin pesat. Semakin banyak informasi yang dibutuhkan maka data yang dibutuhkan juga semakin banyak dan jumlahnya akan semakin besar. Kebutuhan akan jumlah data yang besar dapat kita temukan dalam dunia pendidikan. Hal ini dikarenakan, setiap tahun ajaran terjadi peningkatan data. Terutama data-data siswa yang terus bertambah dari tahun ke tahun.

Jumlah data yang terus meningkat ini memerlukan beberapa metode untuk mengolah dan mengambil kesimpulan dan informasi dari data tersebut. Beberapa metode yang digunakan untuk mengolah data yang sifatnya besar untuk menemukan pola yang terdapat didalamnya diantaranya adalah : teknik klastering, analisis diskriminan, teorema bayes, decision tree artificial neural networks, support vector machine, regresi linear, support vector regresi. Setiap metode tersebut memiliki algoritma-algoritma yang digunakan untuk memproses data yang ada. Namun pada kesempatan kali ini penulis mengangkat mengenai penggunaan algoritma naive bayes dan algoritma C4.5 yang merupakan algoritma dari metode teorema bayes dan decision tree.

Berdasarakan masalah yang telah diuraikan maka penulis mengangkat judul ”Aplikasi Algoritma Naive Bayes dan Algoritma C4.5 dalam Prediksi Kelulusan Mahasiswa Universitas Gunadarma”

Batasan Masalah

Pembahasan masalah yang diangkat yaitu:

1. Penerapan algoritma naive bayes dan C4.5 dalam prediksi kelulusan mahasiswa yang dapat lulus sesuai dengan waktu studi menggunakan Java Netbeans.
2. Analisa perbandingan hasil dan akurasi algoritma naive bayes dan algoritma C4.5.

Tujuan Penulisan

Tujuan dari penelitian ini adalah mencari dan menemukan pola yang terdapat pada data mahasiswa berdasarkan data NEM, IP DNS semester 1, IP DNS semester 2, IPK DNU semester 1-2, gaji orang tua

dan pekerjaan orang tua, untuk memprediksi mahasiswa yang lulus atau tidak lulus sesuai dengan waktu studi dengan menggunakan algoritma naive bayes dan C4.5, kemudian membandingkan hasil dan akurasi kedua algoritma tersebut.

LANDASAN TEORI

Data Mining

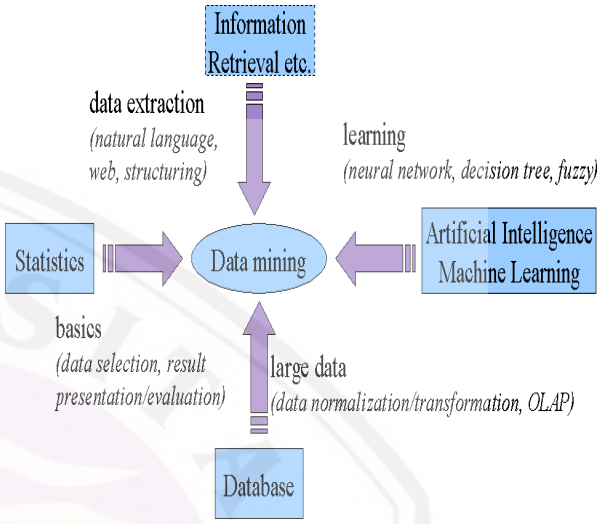
Data mining adalah proses yang menggunakan statistik, matematika, kecerdasan buatan, dan machine learning untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar [Turban, dkk.2005].

Menurur Gartner Group data mining adalah suatu proses menemukan hubungan yang berarti, pola dan kecendrungan dengan memeriksa dalam sekumpulan besar data yang tersimpan dalam penyimpanan dengan menggunakan teknik pengenalan pola seperti teknik statistik dan matematika [Larose, 2005].

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, sering juga disebut knowledge discovery in database (KDD), adalah kegiatan meliputi pengumpulan, pemakaian data historis untuk menemukan keteraturan, pola atau hubungan dalam set data berukuran besar. Keluaran dari data mining ini bisa dipakai untuk memperbaiki pengambilan keputusan di masa depan.

Sejarah Data mining bukanlah suatu bidang yang sama sekali baru. Gambar 1 menunjukkan bahwa data mining memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (artificial intelligent), machine learning, statistic, database dan juga information retrieval



Gambar 1 Hubungan Data Mining dengan bidang ilmu lain

Metode Pelatihan

Metode pelatihan adalah cara berlangsungnya pembelajaran atau pelatihan dalam data mining. Secara garis besat metode pelatihan dibedakan ke dalam dua pendekatan :

a. Pelatihan yang terawasi (*Supervised learning*)
Pada pembelajaran terawasi, kumpulan input yang digunakan, output-outputnya telah diketahui.

b. Pelatihan Tak terawasi (*Unsupervised Learning*)
Dalam pelatihan tak terawasi, metode diterapkan tanpa adanya latihan (training) dan tanpa ada guru (teacher). Guru disini adalah label dari data.

Jenis Nilai Variabel

Variabel berdasarkan nilainya bisa dikelompokkan sebagai berikut :

1. Nominal
   Variabel yang nilainya berupa simbol, nilainya sendiri hanya berfungsi sebagai label atau memberi nama, tidak ada hubungan antar nilai nominal, tidak bisa diurutkan atau diukur jaraknya dan hanya uji persamaan yang bisa dilakukan.

2. Ordinal
   Variabel yang nilainya berupa simbol tetapi bisa diurutkan, tidak bisa diukur jaraknya, tidak bisa dijumlahkan. Kadang perbedaannya dengan variabel nominal kurang tegas.

3. Interval
   Variabel yang nilainya bisa diurutkan, dan diukur dengan tetap dan unit yang sama.

4. Rasio
   Variabel yang mempunyai nilai nol yang mutlak. Nilai variabel rasio diperlakukan sebagai bilangan riil. Semua operasi matematika, seperti penjumlahan, pengurangan, pembagian dan sebagainya, bisa dilakukan terhadap nilai rasio.

Pengelompokan Data Mining

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu [Larose, 2005] :

a. Deskripsi

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

b. 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 ada yang telah diklasifikasi dan dengan menggunakan hasilnya untuk memberikan sejumlah aturan. Klasifikasi menggunakan *supervised learning.*

c. Estimasi

Estimasi hampir sama dengan klasifikasi, perbedaaanya 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.

d. Prediksi

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

e. Klastering

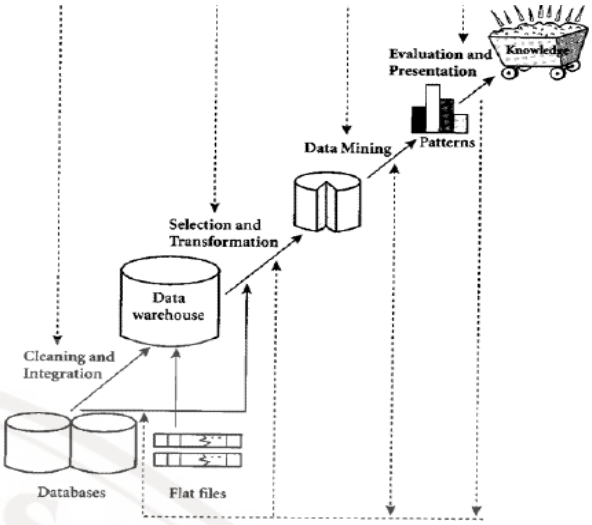
Klastering merupakan pengelompokan record, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan dengan record-record dalam kluster lain. Klastering menggunakan un*supervised learning.*

f. 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 assosiatif dapat diketahui dengan dua parameter, *support* dan *confidence.*

Proses Data Mining

Dalam penerapan data mining, diperlukan pemahaman terhadap data, proses diperolehnya data, alasan menerapkan data mining dan target yang ingin dicapai. Sehingga secara garis besar sudah ada hipotesa mengenai aksi-aksi yang dapat diterapkan dari hasil penerapan data mining. Pemahaman-pemahaman tersebut akan sangat membantu dalam mendesain proses data mining dan juga pemilihan teknik data mining yang akan diterapkan. Pada umumnya proses data mining berjalan interaktif karena tidak jarang hasil data mining pada awalnya tidak sesuai dengan harapan analisnya sehingga perlu dilakukan desain ulang prosesnya.



Gambar 2 Tahapan proses dalam data mining

Tahap-Tahap Data Mining

Sebagai suatu rangkaian proses, data mining dapat dibagi menjadi beberapa tahap yang diilustrasikan di Gambar 2. Tahap-tahap tersebut bersifat interaktif di mana pemakai terlibat langsung atau dengan perantaraan knowledge base.

a Pembersihan data (untuk membuang data yang tidak konsisten dan noise)

Pada umumnya data yang diperoleh, baik dari database suatu perusahaan maupun hasil eksperimen, memiliki isian-isian yang tidak relevan dengan hipotesa data mining yang kita miliki. Pembersihan data yang tidak relevan akan mempengaruhi performasi dari sistem data mining karena data yang ditangani akan berkurang jumlah dan kompleksitasnya.

b Integrasi data (penggabungan data dari beberapa sumber)

Integrasi data dilakukan pada atribut-aribut yang mengidentifikasikan entitas-

entitas yang unik. Integrasi data perlu dilakukan secara cermat karena kesalahan pada integrasi data bisa menghasilkan hasil yang menyimpang dan bahkan menyesatkan pengambilan aksi nantinya. Dalam integrasi data ini juga perlu dilakukan transformasi dan pembersihan data karena seringkali data dari dua database berbeda tidak sama cara penulisannya atau bahkan data yang ada di satu database ternyata tidak ada di database lainnya.Hasil integrasi data sering diwujudkan dalam sebuah data warehouse

c **Transformasi data (data diubah menjadi bentuk yang sesuai untuk di-mining)**

Beberapa teknik data mining membutuhkan format data yang khusus sebelum bisa diaplikasikan. Karenanya data berupa angka numerik yang berlanjut perlu dibagi-bagi menjadi beberapa interval. Proses ini sering disebut binning. Transformasi dan pemilihan data ini juga menentukan kualitas dari hasil data mining nantinya karena ada beberapa karakteristik dari teknik-teknik data mining tertentu yang tergantung pada tahapan ini.

d **Aplikasi teknik data mining**

Aplikasi teknik data mining sendiri hanya merupakan salah satu bagian dari proses data mining. Beberapa teknik data mining sudah umum dipakai. Ada kalanya teknik-teknik data mining umum yang tersedia di pasar tidak mencukupi untuk melaksanakan data mining di bidang tertentu atau untuk data tertentu.

e **Evaluasi pola yang ditemukan (untuk menemukan yang menarik/bernilai)**

Dalam tahap ini hasil dari teknik data mining berupa pola-pola yang khas maupun model prediksi dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai. Bila ternyata hasil yang diperoleh tidak sesuai hipotesa ada beberapa alternatif yang dapat diambil seperti : menjadikannya umpan balik untuk memperbaiki proses data mining, mencoba teknik data mining lain yang lebih sesuai, atau menerima hasil ini sebagai suatu hasil yang di luar dugaan yang mungkin bermanfaat.

f **Presentasi pola yang ditemukan untuk menghasilkan aksi**

Tahap terakhir dari proses data mining adalah bagaimana memformulasikan keputusan atau aksi dari hasil analisa yang didapat. Ada kalanya hal ini harus melibatkan orang-orang yang tidak memahami data mining. Karenanya presentasi hasil data mining dalam bentuk pengetahuan yang bisa dipahami semua orang adalah satu tahapan yang diperlukan dalam proses data mining.

Teorema Bayes

Teorema keputusan bayes adalah pendekatan statistik yang fundamental dalam pengenalan pola (pattern recoginition). Pendekatan ini didasarkan

pada kuantifikasi trade-off antara berbagai keputusan klasifikasi dengan menggunakan probabilitas dan ongkos yang ditimbulkan dalam keputusan-keputusan tersebut.

Ide dasar dari bayes adalah menangani masalah yang bersifat hipotesis yakni mendesain suatu klasifikasi untuk memisahkan objek. Misalkan terdapat dua jenis objek dengan kemungkinan kemunculan random, selanjutnya ingin diprediksi objek apa yang akan lewat selanjutnya. Objek pertama diwakili oleh h_1 dan objek kedua diwakili oleh h_2. Karena apa yang akan muncul bersifat probablistik maka *h* adalah suatu variable yang harus di deskripsikan secara probabilistik. Selanjutnya *probabilitas a priori*, $P(h_1)$ dan $P(h_2)$ masing-masing menyatak peluang munculnya objek 1 dan objek 2. Walaupun probabilitas kemunculan kedua objek tersebut tidak diketahui dengan pasti tapi setidaknya dapat diestimasi dari data yang tersedia. Misalkan N adalah jumlah total kedua objek, kemudian N1 dan N2 masing-masing menyatakan jumlah objek 1 dan objek 2, selanjutnya $P(h_1) \approx \frac{N1}{N}$ dan $P(h_2) \approx \frac{N2}{N}$.

Jika $P(h_1)$ jauh lebih besar dibanding $P(h_2)$ maka logis bila diprediksi bahwa kemunculan yang paling sering adalah objek 1. Tetapi bila $P(h_1) = P(h_2)$, maka peluang prediksi akan menjadi 50-50.

Selanjutnya dalam kasus ini digunakan informasi warna, x, sebagai tambahan informasi untuk meningkatkan keakuratan prediksi. Perbedaan warna ini dinyatakan dalam term probabilistik, x dianggap sebagai variable random kontinyu yang didistribusikan bergantung pada kemunculan objek dan dinyatakan dengan $P(x \setminus h)$ yang menyatakan peluang muncul x jika diketahui *h*. Sehingga $P(x \setminus h_1)$ dan $P(x \setminus h_2)$ menyatakan perbedaan distribusi dalam hal warna antara objek 1 dan objek 2.

Fungsi padat peluang, $P(x \setminus h_j)$, sering juga disebut dengan istilah fungsi likelihood dari h_j terhadap x. dari tambahan informasi berupa likelihood, $P(x \setminus h_j)$, bisa didapatkan *probabilitas posterior*

$$P(h_j \mid x) = \frac{p(x \setminus h_j)P(h_j)}{p(x)}$$

dimana $P(h_j \mid x)$ menyatakan probabilitas muncul h_j jika diketahi *x*. Sedangkan evidence dalam kasus kategori dua kelas adalah

$$P(x) = \sum_{i=1}^{2} p(x \mid h_j)P(h_i)$$

Sehingga secara umum, rumus bayes bisa diberikan sebagai berikut

$$posterior = \frac{likelihood \times prior}{evidence}$$

Jadi, rumus bayes sebenarnya adalah dengna mengetahui nilai x maka *probabilitas prior* $P(h_j)$ dapat diubah menjadi *probabilitas posterior* $P(h_j \mid x)$ yaitu probabilitas keluarnya keluarnya hasil h_j jika diketahui nilai x tertentu. Perkalian *likelihood* dengan *prior* adalah hal paling penting untuk menemukan *posterior*. Karena *evidence* bisa dianggap sebagai factor skala sehingga hasil penjumlahan probabilitas posterior sama dengan 1. Aturan bayes bisa ditetapkan sebgai berikut

$P(h_1 \mid x) < P(h_2 \mid x)$, maka x diklasifikasikan sebagai h_2

Aturan tersebut juga dapat dilihat ketika diambil suatu keputusan. Misalkan pengamatan nilai x tertentu, maka probabilitas error adalah :

$$P(error \mid x) = \begin{cases} P(h_1 \mid x), \textit{ jika } h_2 \\ P(h_2 \mid x), \textit{ jika } h_1 \end{cases}$$

Jadi jelasnya peluang error bisa diminimalkan jika diberikan nilai x dengan memutuskan h_1 jika $P(h_1 \mid x) > P(h_2 \mid x)$ dan memtuskan h_2 jika $P(h_2 \mid x) > P(h_1 \mid x)$.

Bayes Learning

Misalkan terdapat beberapa alternatif hipotesis $h \in H$. Dalam bayes learning, dimaksimalkan hipótesis yang paling mungkin, *h*, atau *maximum a priori* (MAP), jira diberi data, x. Secara matematis ini bisa dirumuskan

$$h_{MAP} = \arg \max P(h \mid x)$$

$$= \arg \max \frac{P(x \mid h)P9h)}{P(x)}$$

$$= \arg \max P(x \mid h)P(h)$$

Dalam banyak kasus diasumsikan bahwa setiap hipotesis *h* dalam H mempunyai peluang prior yang sama ($P(h_i) = P(h_j)$ untuk semua h_i dan h_j dalam H). $P(x \mid h)$ sering disebut likelihood dari data *x* diberikan *h* dan sembarang hipotesis yang memaksimalkan $P(x \mid h)$ dinamakan hipotesis maximum likelihood, yang dinotasikan

$$h_{ML} = \arg \max_{h \in H} P(h \mid x)$$

Dalam konteks data mining atau machine learning, data *x* adalah set training dan H adalah ruang dimana fungsi *f(.)* yang ingin ditemukan letaknya.

Algoritma Naive Bayes

Klasifikasi Bayesian adalah pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu class. Klasifkasi bayesian didasarkan pada teorema bayes. Dari hasil studi perbandingan algoritma klasifikasi, didapatkan bahwa hasil klasifikasi bayesian atau lebih dikenal dengan Naive Bayes Classification dari segi performa lebih baik dari algoritma decisión tree dan algoritma selected neural networks classifiers. Naive Bayesian Classifiers juga memilki kecepatan dan keakuratan yang tinggi bila di implementasikan ke dalam database yang ukurannya besar.

Naive bayesian classifiers berasumsi bahwa efek dari statu pada kelas yang diberikan adalah independent terhadap nilai atribut yang lainnya. Asumsi ini biasa disebut dengan class condicional independence. Itu dibuat untuk menyederhanakan komputasi yang terkait dan dalam hal ini disebut sebagai ‘naive’. Bayesian belief network adalah model grafik yang tidak seperti naive bayesian classifiers, yang memperbolelhkan representasi dari ketergantungan diantara atribut dari sebuah subset. Bayesian belief network dapat juga digunakan dalam pengklasifikasian.

Naive Bayes Classifiers (NBC) merupakan sebuah pengklasifikasi probabilitas sederhana yang mengaplikasikan Teorema Bayes dengan asumsi ketidaktergantungan (independent) yang tinggi.

Keuntungan penggunaan NBC adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan

(training data) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian.

Karena yang diasumsikan sebagai variable independent, maka hanya varians dari statu variable dalam sebuah kelas yang dibutuhkan untuk menentukan klasifikasi, bukan keseluruhan dari matriks kovarians.

Salah satu penerapan teorema bayes adalah naive bayes. Naive bayes didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara konditional saling bebas jika diberikan nilai output. Atau dengan kata lain, diberikan nilai output, probabilitas mengamati secara bersama adalah produk dari probabilitas individu atau $P(a_1, a_2, a_3, \ldots, a_n \mid \upsilon_j) = \prod_i P(a_i \mid \upsilon_j)$. Persamaan ini bisa digunakan untuk mendapatkan pendekatan yang dipakai dalam klasifier Naive Bayes dengan memasukannya ke dalam persamaan :

$$n_{NB} = \arg\max_{\upsilon j \in V} P(\upsilon_j) \prod_i P(a_i \mid \upsilon_j)$$

dimana υ_{NB} adalah nilai output dari hasil klasifikasi Naïve Bayes.

$P(a_i \mid \upsilon_j)$ adalah rasio antara $\frac{n_c}{n}$ dimana n_c adalah jumlah data training dimana $v = v_j$ dan $a = a_i$ dan n adalah total kemungkinan output. Kadang-kadang untuk banyak kasus, estimasi ini kurang akurat terutama jika jumlah kejadian yang diperhatikan sangat kecil. Misalnya terdapat 5 buah sample data dengan 3 buah atribut : a_1, a_2, a_3 dan 2 kemungkinan output ya dan tidak, pada atribut a_1 5 sampel tersebut menghasilkan keputusan tidak. Maka, nilai n_c untuk keputusan ya dalam atribut a_1 adalah 0. Hal ini akan memunculkan dua kesulitan. Pertama, $\frac{n_c}{n}$ akan menghasilkan under estimate probabilitas bias. Yang kedua, jika estimasi probabilitas ini sama dengan *nol*, probabilitas ini akan mendominasi klasifier bayes jika ada data baru. Alasannya adalah nilai yang dihitung dari sample tersebut, semua term akan dikalikan dengan nol. Untuk menghindari kesulitan ini maka digunakan pendekatan Bayesian untuk estimasi probabilitas. Untuk mengestimasi probabilitas digunakan rumus, yang sering disebut m-estimate :

$$P(a_i \mid \upsilon_j) = \frac{n_c + mp}{n + m}$$

dimana n = jumlah data training dimana $v = v_j$, n_c = jumlah data training dimana $v = v_j$ dan $a = a_i$, p = prior estimaste untuk $P(a, i \setminus v_j)$ dan m = ukuran sample ekuivalen

Cara yang biasa digunakan untuk memilih nilai *p* jika informasi lain tidak ada adalah asusmsi keseeragaman, yaitu jika ada *k* nilai yang mungkin maka *p=l/k*. Nilai m bisa diberi nilai sembarang, tetapi konsisten untuk semua atribut. Jika n dan m keduanya tidak nol, maka fraksi yang diamati adalah $\frac{n_c}{n}$ dan probabilitas prior *p* akan dikombinasikan menurut bobot *m*. Jadi alasan mengapa m dinamakan *ukuran sample ekuivalen* bahwa dalam rumus m-estimate terjadi penguatan observasi actual n dengan tambahan sample virtual *m* yan terdistribusi menurut *p*.

Decision Tree

Decision tree merupakan metode klasifikasi dan prediksi yang sangat kuat dan terkenal. Metode decision tree mengubah fakta yang sangat besar

menjadi pohon keputusan yang merepresentasikan aturan. Aturan dapat dengan mudah dipahami dengan bahasa alami. Selain itu aturan juga dapat diekspresikan dalam bentuk bahasa basis data seperti Structured Query Language (SQL) untuk mencari record pada kategori tertentu.

Decision tree juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel input dengan sebuah variabel target. Karena decision tree memadukan antara eksplorasi data dan pemodelan. Decision tree digunakan untuk kasus-kasus dimana outputnya bernilai diskrit.

Sebuah decison tree 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 dengan yang lain [Berry & Linoff, 2004].

Proses pada decision tree adalah mengubah bentuk data (tabel) menjadi model pohon, mengubah model pohon menjadi rule, dan menyederhanakan rule [Basuki & Syarif, 2003].

Sebuah model decision tree terdiri dari sekumpulan aturan untuk membagi sejumlah populasi yang heterogen menjadi lebih kecil, lebih homogen dengan memperhatikan pada variabel tujuannya. Variabel tujuan biasanya dikelompokkan dengan pasti dan lebih mengarah pada perhitungan probabilitas dari tiap-tiap record terhadap kategori-kategori tersebut atau untuk mengklasifikasi record dengan mengelompokkannya dalam satu kelas.

Data dalam decision tree biasanya dinyatakan dalam bentuk tabel dengan atribut dan record. Atribut menyatakan suatu parameter yang dibuat sebagai kriteria dalam pembentukan pohon. Atribut ini juga memiliki nilai-nilai yang terkandung didalamnya yang disebut instance. Dalam decision tree setiap atribut akan menempati posisi simpul. Selanjutnya setiap simpul akan memiliki jawaban yang dibentuk dalam cabang-cabang, jawaban ini adalah instance dari atribut (simpul) yang ditanyakan. Pada saat penelusuran, pertanyaan pertama akan ditanyakan pada simpul akar. Selanjutnya akan dilakukan penelusuran ke cabang-cabang simpul akar dan simpul-simpul berikutnya. Penelusuran setiap simpul ke cabang-cabangnya akan berakhir ketika suatu cabang telah menemukan simpul kelas atau obyek yang dicari.

Ada beberapa hal yang perlu diperhatikan dalam membuat decision tree, yaitu :

a. Atribut mana yang akan dipilih untuk pemisahan obyek.
b. Urutan atribut mana yang akan dipilih terlebih dahulu.
c. Struktur tree.
d. Kriteria pemberhentian.
e. Pruning.

Ada beberapa macam algoritma yang dapat dipakai dalam pembentukan pohon keputusan, antara lain : ID3, CART, dan C4.5. Algortima C4.5 merupakan pengembangan dari algoritma ID3.

Kriteria Pemilihan Atribut

Saat menyusun sebuah decision tree pertama yang harus dilakukan adalah menentukan atribut mana yang akan menjadi simpul akar dan atribut mana yang akan menjadi simpul selanjutnya. Pemilihan atribut yang baik adalah atribut yang memungkinkan

untuk mendapatkan decision tree yang paling kecil ukurannya. Atau atribut yang bisa memisahkan obyek menurut kelasnya. Secara heuristik atribut yang dipilih adalah atribut yang menghasilkan simpul yang paling ”purest” (paling bersih). Ukuran purity dinyatakan dengan tingkat *impurity*. Pengukuran tingkat impurity dapat dilakukan dengan beberapa kriteria, diantaranya :

- **Information Gain**
  Information gain adalah kriteria yang paling populer untuk pemilihan atribut. Information gain dapat dihitung dari output data atau variabel dependent y yang dikelompokkan berdasarkan atribut A, dinotasikan dengan gain (y,A). Information gain, gain(y,A), dari atribut A relatif terhadap output data y adalah :

$$gain(y, A) = entropi(y) - \sum_{c \in nilai(A)} \frac{y_c}{y} entropi(y_c)$$

  dimana nilai(A) adalah semua nilai yang mungkin dari atribut A, dan y_c adalah subset dari y dimana A mempunyai nilai c.

- **Gain Ratio**
  Untuk menghitung gain ratio diperlukan suatu term SplitInformation.
  SplitInformation dapat dapat dihitung dengan formula sebagai berikut :

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

  dimana S_1 sampai S_c adalah c subset yang dihasilkan dari pemecahan S demgan menggunakan atribut A yang mempunyai sebanyak c nilai.

  Selanjutnya gain ratio dihitung dengan cara :

$$Gamratio(S, A) = \frac{Gain(S, A)}{SplitInformation(S, A)}$$

- **Indeks Gini**
  Jika kelas obyek dinyatakan dengan k, k-1,2, ...C, dimana C adalah jumlah kelas untuk variabel/output dependent y, Indeks Gini untuk suatu cabang atau kotak A dihitung sebagai berikut :

$$IG(A) = 1 - \sum_{k=1}^{c} p_k^2$$

  dimana p_k adalah ratio observasi dalam kotak A yang masuk dalam kelas k. Jika IG(A) = 0 berarti semua data dalam kotak A berasal dari kelas yang sama. Nilai IG(A) mencapai maksimum jika dalam kelas A proporsi data dari masing-masing kelas yang ada mencapai nilai yang sama.

Algoritma C4.5

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

Algortima C4.5 sendiri merupakan pengembangan dari algortima ID3, dimana pengmabangan dilakukan dalam hal : bisa mengatasi missing data, bisa mengatasi data kontiyu, pruning. Secara umum algoritma C4.5 untuk membangun pohon keputusan adalah sebagai berikut:

a. Pilih atribut sebagai akar.
b. Buat cabang untuk tiap-tiap nilai.
c. Bagi kasus dalam cabang.

d. Ulangi proses untuk setiap cabang sampai semua kasus pada cabang memiliki kelas yang sama.

Dalam algortima C4.5 digunakan information gain untuk memilih atribut yang akan digunakan untuk pemisahan obyek. Atribut yang mempunyai information gain paling tinggi dibanding atribut yang lain relatif terhadap set y dalam suati data, dipilih untuk melakukan pemecahan.

Sebelum melakukan perhitungan information gain terlebih dahulu harus dicari nilai informasi dalam satuan bits dari suatu kumpulan obyek. Entropi menyatakan impurity suatu kumpulan obyek. Jika diberikan sekumpulan obyek dengan label/output y yang terdiri dari obyek berlabel 1,2 sampai n, entropi obyek dengan n kelas ini dihitung dengan rumus berikut :

$$Entropi(y) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 ..... - p_n \log$$

dimana p_1, p_2, p_n masing-masing menyatakan proporsi kelas 1, kelas 2, ..., kelas n dalam output. Jika perbandingan dua kelas, rasionya sama maka nilai entropinya 1. Jika satu set hanya terdiri dari satu kelas maka entropinya 0.

Untuk memilih atribut sebagai akar, didasarkan pada nilai information gain teringgi dari atribut-atribut yang ada. Untuk menghitung gain digunakan rumus berikut :

$$gain(y, A) = entropi(y) - \sum_{c \in nilai(A)} \frac{y_c}{y} entro$$

Data Pengujian

Pada proses pengujian ini penulis menggunakan 22 data set mahasiswa universitas gunadarma jurusan teknik informatika, 15 data yang digunakan berasal dari data angkatan 2005 yang didapat saat proses pengumpulan data. Dimana 10 data yang digunakan merupakan data yang dipakai dalam proses training sementara itu 5 data lainnya tidak digunakan untuk proses training. Sedangkan 6 data lainnya merupakan data yang diperoleh dari angkatan 2006 yang pada saat ini telah dinyatakan lulus ujian sidang yang diperoleh melalui tanya-jawab singkat kepada mahasiswa yang bersangkutan mengenai data-data yang dibutuhkan untuk proses pengujian ini. Penulis akan membandingkan akurasi hasil algoritma naive bayes dan algoritma C4.5 menggunakan aplikasi yang dibuat penulis.

Tabel 1 Data Hasil pengujian

NPM	Keterangan	Prediksi dengan Naive Bayes	Prediksi dengan algoritma C4.5
50405696	Tidak Lulus	Tidak Lulus	Tidak Lulus
50405782	Lulus	Tidak Lulus	Tidak Lulus
50405761	Tidak Lulus	Tidak Lulus	Tidak Lulus
50405762	Lulus	Tidak Lulus	Tidak Lulus
50405779	Tidak Lulus	Tidak Lulus	Tidak Lulus
50406659	Lulus	Lulus	Lulus
50406145	Lulus	Lulus	Lulus
50406168	Lulus	Lulus	Tidak Lulus
50406737	Lulus	Lulus	Lulus
50406574	Lulus	Lulus	Lulus
50406650	Lulus	Lulus	Lulus
50405006	Tidak Lulus	Tidak Lulus	Tidak Lulus
50405054	Lulus	Tidak Lulus	Lulus
50405204	Lulus	Lulus	Lulus
50405218	Tidak Lulus	Tidak Lulus	Tidak Lulus
50405137	Lulus	Lulus	Lulus
50405407	Tidak Lulus	Lulus	Tidak Lulus
50405408	Lulus	Lulus	Lulus
50405425	Tidak Lulus	Tidak Lulus	Tidak Lulus
50405427	Lulus	Lulus	Lulus
50405438	Tidak Lulus	Tidak Lulus	Tidak Lulus

Hasil Pengujian Naive Bayes

Berdasarkan hasil pengujian dari 22 data diatas,terdapat 2 data yang hasil prediksi dengan menggunakan algoritam naive bayes yang tidak sama dengan hasil data yang sebenarnya. Pada data dengan npm : 50405782, 50405762 dan 50405054. Pada hasil yang sebenarnya ketiga mahasiswa pertama dinyatakan lulus tepat waktu ,sedangkan pada proses prediksi dengan naive bayes adalah tidak lulus. Sementara itu mahasiswa 50405407 pada kenyataannya tidak lulus namun hasil prediksinya bertolak belakang.

Berdasarkan hasil pengujian didapat akurasi ketepatan hasil prediksi naive bayes adalah : ((17/21) x 100%) = 80,85%. Sementara Presentase kesalahan adalah : ((4/21) x 100%) = 19,05%.

Hasil Pengujian C4.5

Hasil penelitian menggunakan algoritma C4.5 didapatkan 3 data yang hasil prediksi tidak sama dengan hasil kelulusan yang sebenarnya, yaitu data mahasiswa untuk npm 50405782, 50405762, 50406168. Ketiga mahasiswa tersebut pada hasil sebenarnya telah dinyatakan lulus tepat waktu sedangkan hasil prediksi C4.5 adalah tidak lulus.

Akurasi ketepatan hasil prediksi C4.5 adalah : ((18/21) x 100%) = 85,7%. Sedangkan nilai kesalahan pada penelitian dengan algoritma C4.5 adalah : ((3/21) x 100%) = 14,3%.

Perbandingan Akurasi Naive Bayes dan C4.5

Berdasarkan hasil pengujian dari masing-masing algoritma maka dapat dilihat pada tabel 2, perbedaan dari akurasi hasil dan kesalahan kedua algoritma.

Tabel 2 Perbandingan akurasi naive bayes dan C4.5

	Prediksi Naive Bayes	Prediksi C4.5
Akurasi Ketepatan	80,85%	85,7%
Kesalahan	19,05%	14,3%

PENUTUP

Kesimpulan

Program Prediksi kelulusan mahasiswa universitas gunadarma algoritma naive bayes dan algoritma C4.5 dalam prediksi kelulusan mahasiswa dibuat untuk membantu pihak kampus memprediksi mahasiswa yang dapat lulus tepat waktu studi sehingga beberapa faktor yang paling mempengaruhi untuk tingkat kelulusan dapat diperhatikan. Setelah dilakukan uji coba dapat disimpulkan beberapa hal sebagai berikut :

1. Proses pengklasifikasian nilai sangat penting karena dapat mengelompokkan nilai-nilai yang akan diuji.
2. Dengan menggunakan algoritma C4.5 kesalahan yang dihasilkan dalam proses prediksi lebih sedikit karena C4.5 melakukan klasifikasi record-record ke dalam kelas tujuan yang ada.
3. Algoritma decision tree memiliki kompleksitas yang lebih besar. Karena pada algoritma C4.5 setiap nilai dalam suatu atribut ditelusuri dan diproses untuk mendapatkan entropi masing-masing nilai yang akan

digunakan untuk mencari ukuran purity masing-masing atribut yang dinyatakan dengan information gain. Proses penelusuran ini akan membentuk sebuah pola berupa pohon keputusan.

4. Algoritma naive bayes bila diimplementasikan menggunakan data yang digunakan dalam proses training akan menghasilkan nilai kesalahan yang lebih besar karena pada naive bayes nilai suatu atribut adalah independent terhadap nilai lainnya dalam satu atribut yang sama. Namun memiliki akurasi akurasi yang lebih tinggi bila dimplementasikan ke data yang berbeda dari data training dan kedalam data yang jumlahnya lebih besar.

Saran

Pada penelitian kali ini data training yang digunakan terbatas yaitu sebanyak 65 record data dan ketidaklengkapan data yang diperoleh penulis. Untuk melihat kinerja yang lebih baik dalam hasil akurasi masing-masing algoritma maka jumlah record data yang digunakan untuk proses training sebaiknya ditingkatkan mendekati jumlah data sesungguhnya.

DAFTAR PUSTAKA

[1]Achmad,Basuki. dan Iwan Syarif. 2003. *Decision Tree.* http://lecturer.eepisits.edu/~entin/Machine%20Learning/Minggu%204%20Decision%20Tree. pdf. Tanggal akses 21 Mei 2010.

[2] Anonim. 2008. *Data Mining*. http://e-learning.myhut.org/public/dss-05.ppt. Tanggal akses 19 Mei 2010.

[3] Anonim. *Konsep Data Mining.* http://student.eepisits.edu/~prara/DM/1Konsep%20Data%20Mining.pdf. Tanggal akses 19 Mei 2010.

[4] Budi, Raharjo., Imam Heryanto, dan Arif Haryono. 2007. *Mudah Belajar JAVA*. INFORMATIKA. Bandung.

[5] Budi, Santosa. 2007. Data Mining Teknik Pemanfata Data untuk Keperluan Bisnis. Graha Ilmu. Yogyakarta.

[6] Cen. 2010. *Syntax dasar dalam SQL (Structured Query Language)*. http://blackcurrant-community.blogspot.com/2010/04/syntax-dasar-dalam sql-structured-query.html. Tanggal akses 30 Juli 2010

[7] Kusnawi. 2007. *Pengantar Solusi Data Mining*.http://p3m.amikom.ac.id/p3m/56%20%20PENGANTAR%20SOLUSI%20DATA%20MINING.pdf . Tanggal akses 15 april 2010.

[8] Kusrini., dan Emha Taufiq Luthfi. 2009. *Algoritma Data Mining*. ANDI. Yogyakarta.

[9] Larose, Daniel .T. 2005. *Discovering Knowledge in Data.* John Willey & Sons. New Jersey

[10] Mardhiya, Hayaty. *Pemanfaatan Teknologi Data Mining Sebagai Pendukung Penyusun Strategi Bisnis.* http://p3m.amikom.ac.id/p3m/dasi/2010/dasimaret2009/8%20%20stmik%20amikom%20yogyakarta%20%20perbandingan%20metode%20nearest%20neighbor%20dan%20algoritma%20c4.5%20untuk%20menganalisis%2

0
kemungkinan%20pengunduran%20diri%20calon%20mahasiswa%20di%20
0stmik%20amikom%20yogyakarta.pdf. Tanggal akses 17 Juli 2010.
[11] Moore, Andrew W. *A gentle introduction to the mathematics of biosurveillance: Bayes Rule and Bayes Classifiers.* http://www.autonlab.org/tutorials/prob_and_naive_bayes.pdf. Tanggal akses 24 Maret 2010.
[12] Windy, Gambetta. 2003. *Pohon Keputusan (Decision Tree.*, http://kur2003.if.itb.ac.id/file/pohon.pdf. Tanggal akses 21 Mei 2010.
[13] Yudho, Giri Sucahyo. 2003. *Data Mining Menggali Informasi yang Terpendam.*http://wsilfi.staff.gunadarma.ac.id/Downloads/files/4413/yudh_odatamining.pdf. Tanggal akses 18 Mei 2010.