Perbandingan Performansi Algoritma *Nearest Neighbor* dan SLIQ untuk Prediksi Kinerja Akademik Mahasiswa Baru

(**Studi Kasus : Data Akademik Mahasiswa Fakultas Teknologi Informasi UNISBANK**)

Arief Jananto
Fakultas Teknologi Informasi, Universitas Stikubank Semarang
email : arief@unisbank.ac.id

Abstrak

Data akademik perguruan tinggi bertambah setiap tahunnya sejalan dengan bertambahnya jumlah mahasiswa. Data yang berlimpah menyimpan informasi yang berlimpah juga.. Teknologi data mining merupakan alat bantu untuk penambangan informasi pada basis data berukuran besar dan telah banyak digunakan pada banyak domain. Memprediksi kinerja (evaluasi belajar) mahasiswa adalah suatu kegiatan untuk menentukan suatu kondisi dimasa depan berdasarkan data yang telah ada.

SLIQ merupakan algoritma yang dikembangkan oleh tim proyek IBM's Quest pada tahun 1996 dapat digunakan untuk dataset yang besar. Penggunaan algoritma SLIQ untuk mengklasifikasikan dan memprediksi kinerja mahasiswa sudah digunakan pada penelitian sebelumnya dengan hasil tingkat akurasi yang masih rendah dikarenakan banyaknya pembatasan. Selanjutnya dilakukan implementasi algoritma Nearest Neighbor yang menggunakan pendekatan untuk mencari kasus dengan menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada. Dari hasil penelitian ini kemudian dibandingkan tingkat akurasi dari hasil prediksi tersebut.

Dari sistem yang dihasilkan dapat disimpulkan bahwa algoritma SLIQ dengan teknik pohon keputusan mempunyai tingkat akurasi prediksi yang lebih rendah dibandingkan dengan tingkat akurasi dari penggunaan algoritma nearest neighbor.

Kata kunci : SLIQ, Nearest Neighbor, prediksi kinerja, akurasi

PENDAHULUAN

Pada sebuah perguruan tinggi data akan bertambah setiap tahunnya dengan data mahasiswa baru. Selain itu data juga bertambah dengan terjadinnya pendaftaran matakuliah yang harus ditempuh pada setiap semester sejumlah mahasiswa aktif pada perguruan tinggi tersebut. Jumlah data yang berlimpah dan dalam ukuran yang cukup besar ini sebenarnya membuka banyak peluang untuk dihasilkan informasi yang berguna bagi manajemen khususnya maupun bagi mahasiswa itu sendiri secara umum.

Teknologi data mining merupakan salah satu alat bantu untuk penambangan data pada basis data berukuran besar dan dengan spesifikasi tingkat kerumitan yang telah banyak digunakan pada banyak domain aplikasi seperti perbankan, deteksi kecurangan maupun bidang telekomunikasi .Beberapa penelitian juga telah banyak dilakukan dengan menggunakan teknik data mining untuk menggali berbagai informasi dari sebuah database, seperti untuk menganalisa kinerja (*performance*) mahasiswa dalam proses belajar mengajar (Kalles dan Pierrakeas, 2006) maupun untuk membantu pengajar untuk mengelola kelas yang diampunya (Agathe dan Kalina, 2005) serta dimungkinkan untuk menganalisa dan mengevaluasi data akademik untuk mengetahui kualitas perguruan tinggi (Al-Radaideh, dkk, 2006). Dalam penelitian dengan algortma Nearest Neighbor dan C4.5 menunjukan hasil pengujian kami menunjukkan bahwa klasifikasi dengan menggunakan metode nearest neighbor tidak lebih akurat dari algoritma C4.5 tetapi proses klasifikasi membutuhkan waktu yang lebih banyak dan memerlukan proses yang lebih panjang (Kusrini, 2009).

Nearest Neighbor adalah pendekatan untuk mencari kasus dengan menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada. Misalkan diinginkan untuk mencari solusi terhadap seorang pasien baru dengan menggunakan solusi dari pasien terdahulu. Untuk mencari kasus pasien mana yang akan digunakan maka dihitung kedekatan kasus pasien baru dengan semua kasus pasien lama. Kasus pasien lama dengan kedekatan terbesar-lah yang akan diambil solusinya untuk digunakan pada kasus pasien baru (Kusrini, 2009).

Pada penelitian sebelumnya peneliti telah mengimplementasikan algoritma SLIQ untuk menghasilkan suatu model yang kemudian model tersebut digunakan untuk meprediksi kinerja akademik mahasiswa baru. Kemudian pada penelitian ini peneliti akan mencoba membangun aplikasi sejenis dengan menggunakan algoritma nearest neighbor dan kemudian membandingkan hasilnya dengan performansi dari aplikasi sebelumnya.

Permasalahan yang akan dibahas dalam penelitian ini adalah "Bagaimana implementasi algoritma nearest neighbor ke dalam bentuk aplikasi sederhana sehingga dapat digunakan untuk memprediksi kinerja akademik mahasiswa baru dan kemudian membandingkan hasil kinerja aplikasi (performansi) dari kedua algoritma yang digunakan yaitu Nearest Neighbor dan SLIQ".

Tujuan yang ingin dicapai dari pelaksanaan penelitian ini adalah Studi terhadap algoritma Nearest Neighbor pada proses klasifikasi untuk prediksi nilai suatu atribut tujuan (kelas). Membangun sebuah aplikasi data mining sederhana dengan mengimplementasikan algoritma Nearest Neighbor, membandingkan hasil kinerja(performansi) dari aplikasi yang dibangun tersebut dengan aplikasi sejenis lain yang telah peneliti buat sebelumnya.

Metode Penelitian

Untuk dapat membandingkan kinerja dari algoritma SLIQ dan algoritma Nearest Neighbor, peneliti memembangun aplikasi dengan menerapkan metode nearest neighbor untuk memprediksi kinerja akademik dari mahasiswa baru berdasarkan data training dari mahasiswa lama.

Variabel yang dipakai dalam aplikasi ini disesuaikan dengan variabel yang dipakai dalam ujicoba penelitian "Prediksi Kinerja Mahasiswa Menggunakan Teknik Data Mining" yaitu : Usia,Kategori_s, Asal_smu, Jenkel,tpa, inggris. Pra proses yang dilakukan dalam penelitian ini juga sama dengan yang dilakukan pada tahap pra proses dalam penelitian sebelumnya.

DATA MINING

Menurut Han dan Kamber (2001) alasan utama mengapa data mining diperlukan adalah karena adanya sejumlah besar data yang dapat digunakan untuk menghasilkan informasi dan knowledge yang berguna. Informasi dan knowledge yang didapat tersebut dapat digunakan pada banyak bidang, mulai manajemen bisnis, control produksi, kesehatan, dan lain-lain.

Secara sederhana, data mining dapat diartikan sebagai proses mengekstrak atau "menggali" knowledge yang ada pada sekumpulan data. Banyak orang yang setuju bahwa data mining adalah sinonim dari Knowledge Discovery in Database, atau yang biasa disebut KDD. Dari sudut pandang yang lain, data mining dianggap sebagai statu langkah yang penting di dalam proses KDD. Han dan Kamber (2001) menyatakan bahwa KDD terdiri dari langkah-langkah *Data cleaning, Data integration, Data selection, Data transformation, Data mining, Pattern evaluation, Knowledge presentation.*

Beberapa penelitian yang terkait seperti, Candra dan Nandhini (2005) melakukan penelitian untuk memprediksi kinerja siswa dengan teknik klasifikasi dengan algoritma digunakan adalah induksi pohon keputusan dan naive bayes.

Kalles dan Pierrakeas (2006) melakukan penelitian untuk menganalisa kinerja siswa pada sistem pembelajaran jauh dengan algoritma genetik dan pohon keputusan. Pada penelitiannya, Kalles dan Pierrakeas (2006) melakukan analisa kinerja suatu kelompok siswa pada sebuah perguruan tinggi di Hellenic Open

University (HOU) melalui kemampuan pengerjaan pekerja rumah yang pada akhirnya akan diperoleh hubungannya dengan tingkat keberhasilan di ujian akhir (*final exam)*.

Al-Radaideh, dkk.,(2006) menganalisa dan mengevaluasi data akademik untuk mendapatkan kinerja dari siswa yang selanjutnya dapat digunakan mengetahui kualitas perguruan tinggi.

Pramudyo (2008) melakukan penelitian tentang klasifikasi mahasiswa baru berdasarkan prediksi indeks prestasi pada semester I(studi kasus program studi teknik informatika universitas bina darma palembang) dengan menggunakan metoda case base reasoning**.**

Algoritma SLIQ juga telah dimanfaatkan pada beberapa bidang masalah seperti salah satu diantaranya pembuatan system pengambilan keputusan penawaran di pasar kelistrikan. Menurut Yan dkk.(2005) sebuah metode yang didasarkan pada SLIQ diterapkan pada unit generasi sistem keputusan penawaran pasar listrik. Pengetahuan bahwa kemampuan satuan penawaran tersebut diperoleh, mengambil permintaan pasar, penawaran harga dan kapasitas satuan penawaran mempertimbangkan untuk merancang proyek perdagangan yang optimal.

Menurut Kusrini(2009), untuk memudahkan dalam melakukan pengambilan keputusan dalam proses penjaringan calon mahasiswa baru di STMIK AMIKOM Yogyakarta diperlukan alat analisis bagi manajemen untuk mengetahui kemungkinan pengunduran diri calon mahasiswa baru. Analisis ini dapat dilakukan dengan memanfaatkan teori penalaran berbasis kasus, yaitu membandingkan kasus calon mahasiswa bariu dengan kasus-kasus yang pernah terjadi di tahun-tahun sebelumnya. Hasil pengujian kami menunjukkan bahwa klasifikasi dengan menggunakan metode nearest neighbor tidak lebih akurat dari algoritma C4.5 tetapi proses klasifikasi membutuhkan waktu yang lebih banyak dan memerlukan proses yang lebih panjang.

1. Klasifikasi

Masih menurut Han dan Kamber (2001) *data classification* memiliki dua tahap proses. Tahap pertama adalah membangun suatu model yang berdasarkan serangkaian data class, yang disebut *learned model*. Model tersebut dibangun dengan menganalisa database tuple. Setiap tuple diasumsikan menjadi *predefined class* yang ditentukan oleh satu atribut yang disebut class label attribute. Akibat terdapat class label maka tahap ini juga dikenal dengan *supervised learning*. Berbeda dengan unsupervised learning atau dikenal dengan clustering.

2. Pengukuran Akurasi Model

Masih menurut Han dan Kamber (2001) bahwa untuk mengukur akurasi atau ketepatan model dapat dilakukan dengan menghitung perbandingan jumlah prediksi benar terhadap total seluruh record yang dapat diprediksi (presentase dari data test yang diprediksi dengan benar oleh model)

acc(M) = percentage of test set tuples that are correctly classified by the model M

Error rate (misclassification rate) of M = 1 – acc(M)

Given *m* classes, $CM_{i,j}$, an entry in a *confusion matrix*, indicates # of tuples in class *i* that are labeled by the classifier as class *j*

Alternatif pengukuran akurasi (*Alternative accuracy measures*)

sensitivity = t-pos/pos /* true positive recognition rate */

specificity = t-neg/neg /* true negative recognition rate */

precision = t-pos/(t-pos + f-pos)

accuracy = sensitivity * pos/(pos + neg) + specificity * neg/(pos + neg)

Tabel 1. Confusion Matrix

	C1	C2
C1	True Positive	False Negative
C2	False Positive	True Negative

Tabel 1. adalah tabel yang digunakan untuk memetakan hasil prediksi dari model terhadap data testing. True Positive (C1,C1) artinya adalah jumlah prediksi benar terhadap kelas pertama dan False Positive (C2,C1) adalah jumlah prediksi salah terhadap kelas pertama. False Negative (C1,C2) adalah jumlah prediksi salah pada kelas kedua dan True Negative (C2,C2) adalah prediksi benar terhadap kelas yang kedua.

Acc(M)=
((C1,C1)+(C2,C2))/((C1,C1)+(C1,C2)+(C2,C1)
+(C2,C2)) x 100%

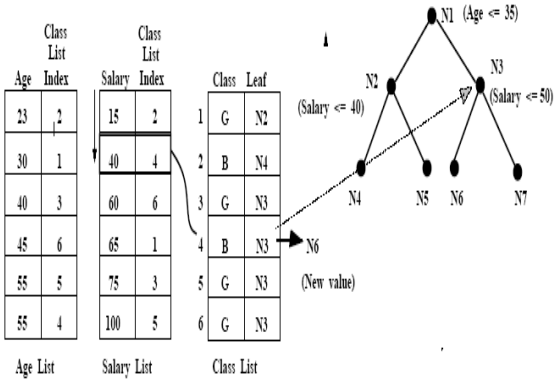
3. Algoritma SLIQ

Algoritma SLIQ ini dikembangkan oleh tim proyek IBM's Quest pada sekitar tahun 1996. Kemunculan algoritma ini merupakan jawaban dari kekurangan algoritma-algoritma sebelumya yang memiliki keterbatasan memori untuk dataset dalam jumlah yang besar. (Mehta dkk.,1996)

Algoritma SLIQ menggunakan modifikasi dari tree classifier sehingga bisa dipakai juga untuk dataset yang besar. SLIQ bisa dipakai untuk atribut dengan tipe numerik dan kategorikal. Decision tree dalam SLIQ menggunakan teknik novel untuk mempersingkat learning time dengan tetap mempertahankan tingkat akurasi yang tinggi.

Selain itu SLIQ juga memungkinkan untuk dilakukan pada sistem dengan jumlah memori yang terbatas tanpa kehilangan performanya. Algoritma ini juga tidak mempunyai batasan jumlah training data atau jumlah atribut yang digunakan. Dengan demikian SLIQ mempunyai potensi menghasilkan klasifikasi yang lebih akurat pada training dataset yang lebih besar, yang tidak dapat dilakukan oleh algoritma-algoritma sebelumnya.

SLIQ menggunakan teknik pre-sorting di fase tree-growth untuk mengurangi cost evaluasi atribut numerik. Prosedur sorting ini diintegrasikan dengan strategi breadth-first tree growing untuk memungkinkan SLIQ melakukan klasifikasi pada dataset yang ada dalam disk. Selain itu SLIQ menggunakan algoritma fast subsetting untuk menentukan split point pada atribut kategorikal. SLIQ juga menggunakan algoritma baru tree-pruning yang berbasiskan pada prinsip Minimum Description Length. Pada aplikasi yang disusun tidak menggunakan proses pruning.

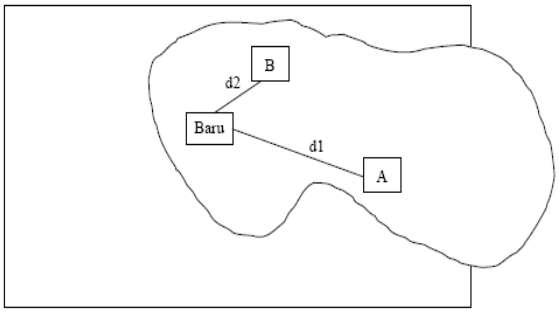


Gambar 1. Proses Update Daftar Kelas (Mehta, dkk., 1996)

Tampak pada gambar 1. merupakan proses update terhadap daftar kelas terhadap pohon yang dihasilkan.

4. Algoritma Nearest Neighbor

Nearest Neighbor adalah pendekatan untuk mencari kasus dengan menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada[8]. Misalkan diinginkan untuk mencari solusi terhadap seorang pasien baru dengan menggunakan solusi dari pasien terdahulu. Untuk mencari kasus pasien mana yang akan digunakan maka dihitung kedekatan kasus pasien baru dengan semua kasus pasien lama. Kasus pasien lama dengan kedekatan terbesar-lah yang akan diambil solusinya untuk digunakan pada kasus pasien baru.



Gambar 2. Ilustrasi Kedekatan Kasus

Seperti tampak pada Gambar 2, terdapat dua pasien lama A dan B. Ketika ada pasien Baru, maka solusi yang akan diambil adalah solusi dari pasien terdekat dari pasien Baru. Seandainya d1 adalah kedekatan antara pasien Baru dan pasien A, sedangkan d2 adalah kedekatan antara pasien Baru dengan pasien B. Karena d2 lebih dekat dari d1 maka solusi dari pasien B lah yang akan digunakan untuk memberikan solusi pasien Baru.

Adapun rumus untuk melakukan penghitungan kedekatan antara dua kasus adalah sebagai berikut[8]:

$$similarity(T,S) = \frac{\sum_{i=1}^{n} f(T_i, S_i)\ x\ w_i}{w_i}$$

dengan

T : kasus baru

S : kasus yagn ada dalam penyimpanan

n : jumlah atribut dalam masing-masing kasus

i : atribut individu antara 1 s/d n

f : fungsi similarity atribut i antara kasus T dan kasus S

w : bobot yang diberikan pada atribut ke i

Kedekatan biasanya berada pada nilai antara 0 s/d 1. Nilai 0 artinya kedua kasus mutlak tidak mirip, sebaliknya untuk nilai 1 kasus mirip dengan mutlak.

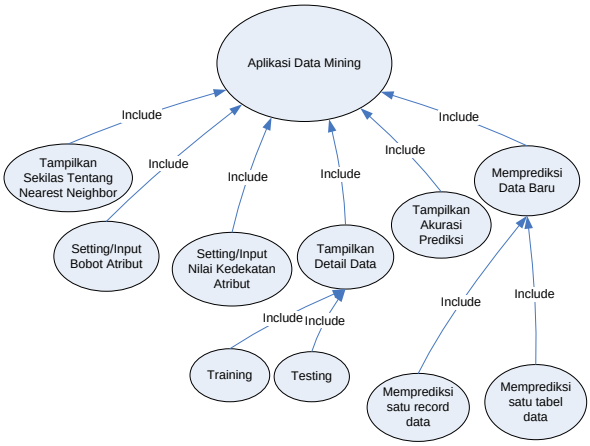
5. Kinerja Mahasiswa

Kinerja yang dimaksud adalah suatu evaluasi terhadap kemajuan belajar mahasiswa, apakah mahasiswa mampu mencapai suatu kondisi yang ditentukan sebelumnya. Evaluasi Kemajuan Studi Mahasiswa, dimana disebutkan bahwa untuk mengetahui kemajuan studi mahasiswa, dilakukan evaluasi setiap empat semester dengan kriteria selain pencapaian jumlah sks adalah pencapaian Indeks Prestasi Kumulatif (IPK). Jika IPK>=2.00 maka mahasiswa dianggap **mampu** dan sebaliknya jika IPK < 2.00 maka mahasiswa dianggap **tidak mampu.** Penggunaan IPK disini hanya sebagai atribut untuk nantinya ditransformasikan menjadi label kelas (atribut target) ’Mampu’ dan ’Tidak Mampu’, sedangkan sebagai atribut untuk memprediksi (*predictor)* terdiri dari jenis kelamin, usia, asal sekolah, kategori sekolah, nilai test potensi akademik dan nilai tes bahasa inggris.

HASIL DAN PEMBAHASAN

1. Fungsi-fungsi Produk

Produk Aplikasi dibangun dalam bentuk form-form, sehingga dalam sebuah form bisa berisi satu atau lebih fungsi yang digunakan untuk mengelola suatu tabel data menjadi suatu aturan. Adapun fungsi-fungsi yang ada dapat dimanfaatkan oleh pengguna dengan cepat. Gambar 3 merupakan diagram fungsi dari produk aplikasi data mining.

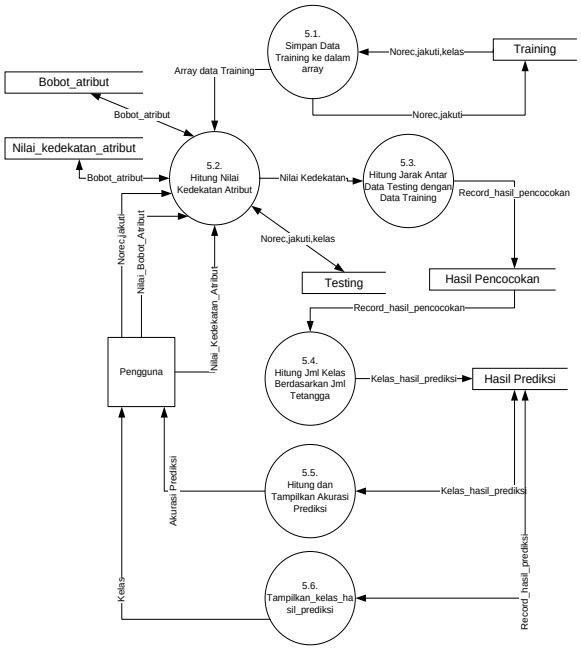


Gambar 3. Diagram hirarki fungsi produk

2. Diagram Alir Data Level 2 Proses Mining Data

Diagram alir data level 2 proses mining data diperlihatkan pada gambar 4.

Pada gambar 4. terdapat 6 proses yaitu simpan data training ke dalam array, hitung nilai kedekatan atribut, hitung jarak antar data testing dengan data training, hitung jumlah kelas berdasarkan jumlah tetangga, hitung dan tampilkan akurasi prediksi, tampilkan kelas hasil prediksi.



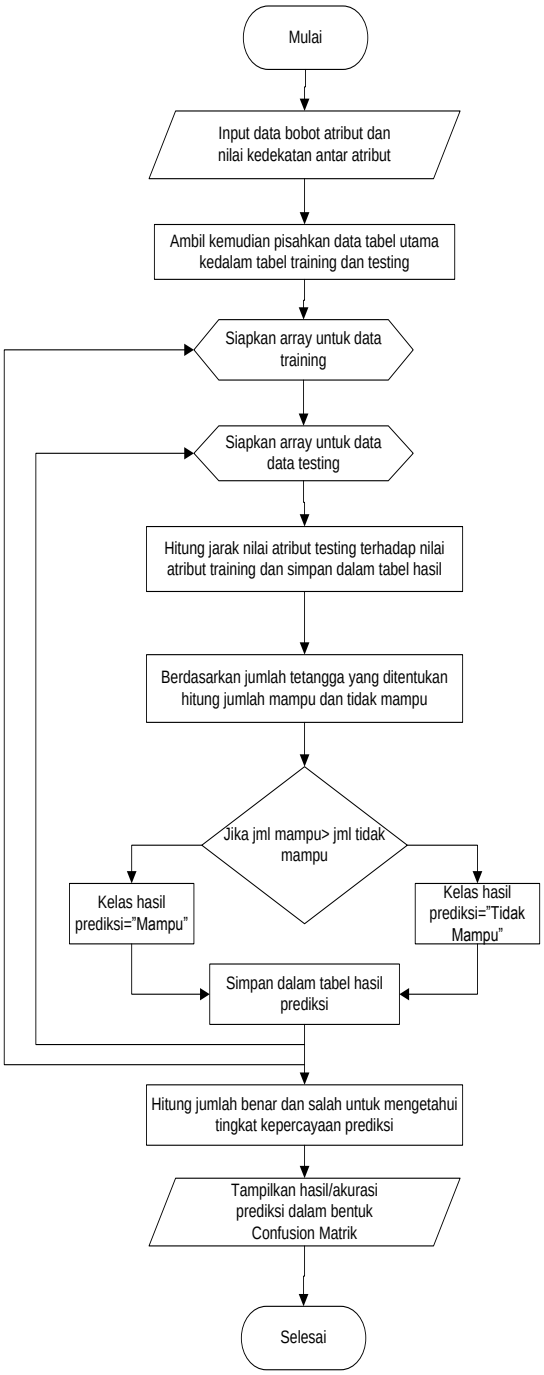
Gambar 4. Diagram Alir Data Level 2 Proses Mining Data

5.6. Diagram Alir Proses Data Mining

Aliran proses data mining diperlihatkan pada gambar 5.

Pertama-tama pengguna diminta menentukan nilai bobot dari atribut-atribut yang digunakan serta nilai kedekatan antar atribut. Selanjutnya system aplikasi akan menggunakan data utama yang berisi seluruh data akademik dan penerimaan mahasiswa baru dari mahasiswa angkatan lama (yaitu tahun 2005 dan 2006) yang nantinya akan dipecah menjadi sebagian(75%) data training dan sisanya(25%) sebagai data testing.

Proses berikutnya adalah menyimpan semua data training kedalam variable array begitu juga juga data testing juga dilakukan pemindahan ke dalam variable array juga. Selanjutnya akan dilakukan perhitungan jarak antar antara atribut data testing terhadap data training. Record data testing ke i akan dibandingkan terhadap seluruh data training. Hasil perhitungan jarak disimpan dalam sebuah tabel data Hasil.dbf yang kemudian berdasarkan penentuan atau jumlah tetangga yang diinginkan sebagai pembanding terdekat, maka dapat ditentukan banyaknya jumlah yang berlabel kelas "Mampu" maupun "Tidak Mampu". Dengan mengambil jumlah yang lebih banyak (berarti lebih mirip) maka label tersebut menjadi label kelas dari record data testing maupun data baru yang diprediksi.



Gambar 5. Diagram Aliran Proses Data Mining

Selanjutnya dari jumlah data kelas hasil prediksi dibandingkan dengan data kelas dari data sebenarnya dari data testing maka akan dapat diperoleh tingkat akurasi atau kepercayaan dari proses prediksi tersebut.

Sedangkan untuk proses prediksi data baru baik berupa satu buah record data maupun sejumlah record data, sama dengan proses data mining pada data testing. Hanya pada proses akhir tidak ada proses perbandingan terhadap tingkat akurasi prediksi karena memang sebelumnya tidak terdapat label kelasnya.

HASIL DAN PEMBAHASAN

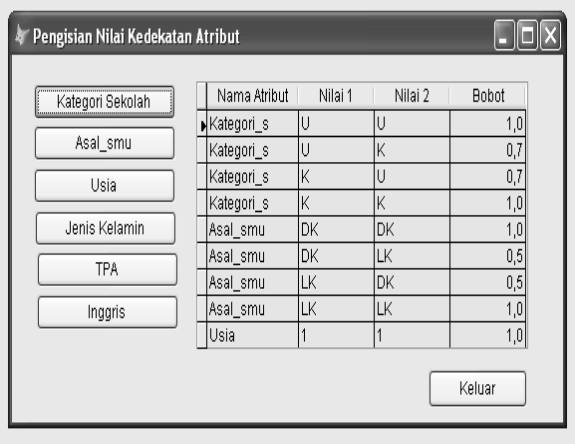
Perangkat lunak pada penelitian ini, dibuat dengan menggunakan bahasa pemrograman Visual Foxpro Versi 9.0. Pada bagian ini akan dijelaskan mengenai implementasi dari proses data mining meliputi pembentukan data training dan data testing, pembentukan atribut list dan sorting, perhitungan nilai gini index setiap atribut, partisi tabel dan pemeriksaan kelas dan penyusunan aturan serta implementasi beberapa rancangan form.

1. Input Data Bobot dari Atribut dan Nilai Kedekatan Antar Nilai Atribut

Data tentang bobot adalah data angka yang menunjukkan tentang nilai bobot dari sebuah atribut dalam penentuan label kelas prediksi, dimana dalam hal ini ditentukan berdasarkan urutan nilai kepentingan terhadap penentuan nilai suatu label kelas yang berhubungan dengan kemampuan akademik (kinerja belajar) dari seorang mahasiswa. Pada penelitian ini urutan tingkat bobot didasarkan pada hasil gini indeks pada penelitian sebelumnya, namun demikian pada aplikasi ini dapat dilakukan perubahan pada nilai bobot dari atribut-atribut tersebut.

Selain itu juga nilai kedekatan antar nilai atribut juga dapat dimasukkan melalui sebuah jendela yang telah disiapkan. Kedekatan nilai antar atribut pada aplikasi ini saat didasarkan pada asumsi dari peneliti, sehingga nilai kedekatan antar nilai atribut hanya diberikan sejumlah nilai yang sekiranya mendekati sesuai kondisi sebenarnya. Namun demikian nilai kedeketan antar nilai atribut dapat dipermbaiki(edit) melalui sebuah jendela yang telah disiapkan.

Gambar 6. Form Penentuan Nilai Bobot Atribut



Gambar 7. Form Pengisian Nilai Kedekatan Atribut

2. Implementasi Proses Pembentukan Data Training dan Data Testing

Data training dapat dibentuk dengan dasar dari pemilihan tahun angkatan yang kemudian dilakukan pemilihan field-field yang dibutuhkan sebagai input dan target/kelas dari tabel datautama.dbf dan disimpan sebagai table dataawal.dbf. Selanjutnya dari table dataawal.dbf dipilih 75 persen dari total jumlah record dari record data yang sesuai sebagai tabel training.dbf dan selebihnya sebanyak 25 persen atau nomor record lebih besar dari 75 persen sebagai tabel testing.dbf. Pembentukan tabel menggunakan fungsi *select* dalam bentuk query seperti terlihat pada gambar 8.

```
SELECT
IdRec,Jenkel,Usia,Asal_smu,Kategori_s,Nt_tpa as tpa, Nt_inggris as inggris,kelas FROM dataawal WHERE
RECNO()<=(ROUND(0.75*RECCOUNT(),0))
ORDER BY idrec INTO TABLE training
SELECT
IdRec,Jenkel,Usia,Asal_smu,Kategori_s,Nt_tpa as tpa, Nt_inggris as inggris,kelas FROM dataawal WHERE
RECNO()>(ROUND(0.75*RECCOUNT(),0))
ORDER BY idrec INTO TABLE testing
```

Gambar 8. Fungsi *select* dalam query pembentuk tabel training dan testing

Adapun data yang digunakan pada penelitian ini adalah data akademik dari mahasiswa angkatan tahun 2005/2006 dan 2006/2007 di Fakultas Teknologi Informasi, sejumlah 1067 record data. Sehingga setelah melalui pembentuk atau pemecahan data, maka akan diperoleh 800 record data untuk data training dan 267 record data untuk data testing..

3. Proses Perhitungan Jarak Antara Nilai Atribut Testing terhadap Nilai Atribut Training.

Setelah data training dan data testing terbentuk maka kemudian dilakukan perhitungan jarak antara nilai atribut testing terhadap nilai atribut training pada atribut yang sama. Perhitungan jarak ditujukan untuk mendapatkan nilai tertinggi terhadap jarak dengan mengalikan nilai kedekatan atribut dengan nilai bobot masing-masing atribut kemudian menjumlahkannya dan selanjutnya dibagi dengan total dari nilai bobot semua atribut, sehingga akan diperoleh suatu nilai tertentu. Dengan ketentuan semakin besar nilai jarak yang diperoleh berarti nilai atribut testing dengan nilai atribut training berarti lebih mirip. Sehingga untuk menentukan label kelas prediksi untuk data record testing ditentukan dengan mengambil sejumlah k (tetangga) nilai jarak yang mempunyai nilai kemiripan yang tinggi. Setelah diambil sejumlah k tetangga nilai jarak , maka dilakukan perhitungan terhadap jumlah masing-masing label kelas tersebut dan untuk label kelas yang lebih banyak, maka label kelas tersebut yang akan dijadikan sebagai label kelas dari record data testing. Hal ini dilakukan untuk setiap data testing terhadap seluruh data training.

Untuk itu sebelum dilakukan perhitungan jarak, maka semua record data training maupun testing dimasukkan kedalam suatu variable array, dengan harapan akan lebih efiesien dalam proses pembacaan data untuk dilakukan pembandingan nilai data.

Pemetaan record data training dan testing kedalam varibel array dilakukan dengan prosedur seperti tampak pada gambar 9.

```
USE training
a=RECCOUNT()
DIMENSION tr(a,7)
FOR i=1 TO a
SELECT
kategori_s,asal_smu,usia,jenkel,tpa,inggris,kelas
FROM trainingx INTO ARRAY tr
NEXT i
USE testing
b=RECCOUNT()
DIMENSION br(b,7)
FOR j=1 TO b
SELECT
kategori_s,asal_smu,usia,jenkel,tpa,inggris,kelas
FROM testing INTO ARRAY br
NEXT j
```

Gambar 9. query pembentukan array data training dan array data testing

Kemudian selain data training dan data testing, data mengenai nilai bobot dan nilai kedekatan antar atribut juga sekaligus dipindahkan ke dalam variable array. Pemindahan juga tujukan untuk mempercepat pembacaan data. Adapun prosedur/query untuk pemindahan data nilai bobot atribut dan nilai kedekatan atribut ke dalam variable array adalah seperti tampak pada gambar 10.

```
DIMENSION bobotfield(7)
SELECT kategori_s,asal_smu,usia, jenkel, tpa, inggris FROM bobot INTO ARRAY bobotfield
b1=bobotfield(1)
b2=bobotfield(2)
b3=bobotfield(3)
b4=bobotfield(4)
b5=bobotfield(5)
b6=bobotfield(6)
DIMENSION nk(39)
SELECT bobot FROM nilai_kedekatan INTO ARRAY br
```

Gambar 10.Query pemindahan data nilai bobot atribut dan nilai kedekatan antar atribut.

Pada gambar 10. setiap nilai bobot atribut dipindahkan ke dalam satu variable array. Untuk nilai bobot atribut disiapkan sebanyak 6 buar variable array, hal ini dikarenakan jumlah atribut yang digunakan untuk penentuan label kelas adalah 6 buah atribut. Atribut yang digunakan dalam penentuan label kelas adalah kategori sekolah (kategori_s), asal sekolah (asal_smu), usia (usia), jenis kelamin (jenkel), nilai tes potensi akademik (tpa), nilai tes bahasa inggris(inggris). Semua nilai atribut tersebut selanjutnya dipindahkan kedalam array bobotfield(1) hingga bobotfield(6), dan kemudian dipindahkan lagi kedalam varibel bebas dari b1 sampai dengan b6.

Untuk nilai kedekatan antar atribut disiapkan sebanyak 39 indek array. Hal ini dikarenakan setiap atribut dapat memiliki sejumlah nilai kedekatan yang dalam penelitian ini diambil sejumlah asumsi, hingga terbentuk nilai kedekatan atribut dari 6 buah atribut menjadi 39 nilai kedekatan.

4. Data Pelatihan

Sebagai data pelatihan digunakan data mahasiswa fakultas teknologi informasi tahun angkatan 2005 dan 2006. Untuk tahun 2005 terdapat jumlah record sebanyak 577 record setelah melalui proses praprosesing secara manual. Dan untuk tahun 2006 terdapat data sebanyak 490 record data. Untuk data training mahasiswa angkatan 2005, yang dijadikan data training adalah sebanyak 433 record atau sebanyak 75% dari total record data 577, dengan jumlah atribut sebanyak 8 buah dimana ada 6 atribut yang digunakan sebagai atribut predictor dan 1 atribut tujuan.

Atribut predictor terdiri dari jenis kelamin yang berisi nilai ‘0’ untuk jenis kelamin Perempuan dan ‘1’ untuk jenis kelamin ‘Laki-laki’. Atribut kategori sekolah berisi nilai ‘KEJURUAN’ untuk nilai data nama sekolah selain smu/sma sedangan nilai ‘UMUM’ merupakan nilai untuk data nama sekolah yang berisi smu/sma. Kemudian atribut asal_smu berisi nilai data ‘Dalam Kota’ untuk smu yang beralamat atau berada diwiliayah semarang dan sekitarnya. Dan ‘Luar Kota’ untuk smu yang beralamat diluar semarang. Atribut Usia berisi nilai data usia dari siswa yang diperoleh dari tanggal lahir. Atribut nilai tes potensi akademik (TPA) dan atribut nilai tes bahasa inggris diperoleh dari data tes masuk mahasiswa baru pada saat mahasiswa tersebut mendaftar.

Sedangkan label kelas/target yang berisi nilai ‘Mampu’ dan ‘Tidak Mampu’ berasal dari nilai IPK sampai dengan semester empat dari mahasiswa. Dimana untuk nilai IPK kurang dari 2.00 diberi label kelas ‘Tidak Mampu’ dan untuk IPK lebih dari atau sama dengan 2.00 diberi label kelas ‘Mampu’.

5. Proses Training

Training dilakukan dengan menggunakan 75% dari keseluruhan data populasi yang digunakan yang bersasal dari 2 tahun angkatan yaitu 2005/2006 dan 2006/2007. Dimana data training yang diperoleh sebanyak 800 record data dan 267 data testing atau 25%. Selanjutnya data training sebanyak 800 record akan dipindahkan(salin) kedalam variabel memory untuk memudahkan dan mengefesienkan waktu pembacaan data, kemudian dibaca dan dihitung secara simultan untuk membandingkannya dengan data testing yang berjumlah 267 record untuk mendapatkan nilai kemiripan(similarity) dalam bentuk jarak antara satu nilai data yang satu dengan nilai data yang lain.

6. Hasil Pengujian

Pengujian pada aplikasi pada penelitian sebelumnya dengan menggunakan data training dari tahun angkatan 2005/2006 dan 2006/2007 diperoleh hasil sebagai berikut :

Pada data tahun angkatan 2005 diperoleh urutan atribut berdasarkan tingkat pengaruh terhadap pencapaian kinerja adalah jenis kelamin, usia, asal_smu, Nilai TPA, Nilai Tes Inggris dan kategoris. Dari hasil pengujian dengan data testing menunjukan akurasi model untuk tahun angkata 2005 adalah sebesar 41,67 % seperti tampak pada gambar 11.(a).

```
Hasil perhitungan tanpa memperhitungkan kelas 'Tidak Terklasifikasi'
pada saat pengklasifikasian/pencocokan terhadap data testing

              CONFUSION MATRIK
-----------------------------------------------
              |    Kelas Sebenarnya        |
-----------------------------------------------
 Prediksi     |   Mampu    | Tidak Mampu   |
-----------------------------------------------
 Mampu        |         34 |          5    |
 Tidak Mampu  |         79 |         26    |
-----------------------------------------------
Jumlah record =         144

Akurasi model (prediksi benar) = 41.67 %
```

(a)

```
Hasil perhitungan tanpa memperhitungkan kelas 'Tidak Terklasifikasi'
pada saat pengklasifikasian/pencocokan terhadap data testing

              CONFUSION MATRIK
-----------------------------------------------
              |    Kelas Sebenarnya        |
-----------------------------------------------
 Prediksi     |   Mampu    | Tidak Mampu   |
-----------------------------------------------
 Mampu        |         68 |          8    |
 Tidak Mampu  |         37 |          9    |
-----------------------------------------------
Jumlah record =         122

Akurasi model (prediksi benar) = 63.11 %
```

(b)

Gambar 11. Matrik hasil prediksi terhadap data tahun angkatan 2005 dan 2006

Berdasarkan gambar 11.(a), dari 113 record berlabel kelas 'Mampu' dapat terprediksi dengan tepat sebagai kelas 'Mampu' sebanyak 34 record sedangkan yang dapat terprediksi sebagai kelas 'Tidak Mampu' sebanyak 79 record. Untuk record berlabel kelas 'Tidak Mampu' dari jumlah 31 record dapat terprediksi dengan tepat sebanyak 26 record sebagai record berlabel kelas 'Tidak Mampu' dan sebanyak 5 record terprediksi sebagai record berlabel kelas 'Mampu'.

Selanjutnya dilakukan uji coba untuk data angkatan tahun 2006, dengan jumlah record data sebanyak 368. Pada data tahun angkatan 2006 urutan atribut berdasarkan pengaruhnya terhadap kinerja (evaluasi belajar) mahasiswa adalah jenis kelamin, usia kemudian berikunya atribut nt_tpa, nt_inggris, asal_smu dan kategori_s. Untuk dua atribut terakhir dapat bervariasi. Diperoleh hasil perhitungan seperti tampak pada gambar 11.(b). Akurasi model aturan berdasarkan data tahun 2006 memiliki tingkat akurasi model 63,11%.

Berikut akan disajikan hasil pengujian dengan menggunakan aplikasi baru yang menerapakan algoritma nearest neighbor dengan menggunakan data tahun akademik 2005/2006 dan 2006/2007.

Adapun hasil pengujian aplikasi dengan menggunakan data akademik mahasiswa tahun angkatan 2005/2006, diperoleh hasil sebagiamana tampak pada tabel 2.

Tahun 2006/2007

Jumlah data training 433

Jumlah data testing 144

Tabel 2. Rekapitulasi Hasil Uji Coba

No	Jumlah Tetangga	Confusion Matrix				Jumah Benar	Jumlah Salah	Waktu
		Mampu		Tidak Mampu		Akurasi	Error	
		Mampu	Tidak Mampu	Mampu	Tidak Mampu			
1	1	113	0	31	0	78,5	21,5	6
2	2	113	0	31	0	78,5	21,5	6
3	3	113	0	31	0	78,5	21,5	6
4	4	113	0	31	0	78,5	21,5	7
5	5	113	0	31	0	78,5	21,5	6
6	6	113	0	31	0	78,5	21,5	6
7	7	113	0	31	0	78,5	21,5	7
8	8	113	0	31	0	78,5	21,5	6
9	9	113	0	31	0	78,5	21,5	7
10	10	113	0	31	0	78,5	21,5	6

Berdasarkan tabel 2, dapat dilihat bahwa setelah dilakukan ujicoba dengan menggunakan data tahun akademik 205/2006 dengan jumlah record training sebanyak 433 dan jumlah record testing sebanyak 144 diperoleh hasil tingkat akurasi prediksi terhadap data testing sebesar 78,5 % dengan 113 dari 144 yang diprediksi tepat sesuai dengan label kelas sebenarnya. Dan 21 % dengan 31 dari 144 record data yang salah dalam memprediksinya. Adapun waktu yang digunakan rata2 adalah 6 detik untuk data sebanyak itu.

Sedangkan hasil pengujian aplikasi dengan menggunakan data akademik mahasiswa tahun angkatan 2006/2005, diperoleh hasil sebagiamana tampak pada tabel 3.

Tahun 2006/2007

Jumlah data training 368

Jumlah data testing122

Tabel 3. Rekapitulasi Hasil Uji Coba

No	Jumlah Tetangga	Confusion Matrix				Jumah Benar	Jumlah Salah	Waktu
		Mampu		Tidak Mampu		Akurasi	Error	
		Mampu	Tidak Mampu	Mampu	Tidak Mampu			
1	1	105	0	17	0	86,5	13,5	4
2	2	105	0	17	0	86,5	13,5	4
3	3	105	0	17	0	86,5	13,5	3
4	4	105	0	17	0	86,5	13,5	4
5	5	105	0	17	0	86,5	13,5	4
6	6	105	0	17	0	86,5	13,5	4
7	7	105	0	17	0	86,5	13,5	4
8	8	105	0	17	0	86,5	13,5	4
9	9	105	0	17	0	86,5	13,5	4
10	10	105	0	17	0	86,5	13,5	4

Berdasarkan tabel 3, dapat dilihat bahwa setelah dilakukan ujicoba dengan menggunakan data tahun akademik 2006/2007 dengan jumlah record training sebanyak 368 dan jumlah record testing sebanyak 122 diperoleh hasil tingkat akurasi prediksi terhadap data testing sebesar 86,5 % dengan 105 dari 122 yang diprediksi tepat sesuai dengan label kelas sebenarnya. Dan 13,5 % dengan 17 dari 122 record data yang salah dalam memprediksinya. Adapun waktu yang digunakan rata2 adalah 4 detik untuk data sebanyak itu.

Sedangkan pengujian pada aplikasi yang saat ini dilakukan dengan menggunakan data yang digabungkan dari 2 tahun angkatan maka diperoleh hasil seperti tampak pada tabel 4.

Jumlah data training 800

Jumlah data testing 267

Tabel 4. Rekapitulasi Hasil Uji Coba

No	Jumlah Tetangga	Confusion Matrix				Jumah Benar	Jumlah Salah	Waktu
		Mampu		Tidak Mampu		Akurasi	Error	
		Mampu	Tidak Mampu	Mampu	Tidak Mampu			
1	1	172	63	23	9	67.8	32.2	119
2	2	158	77	23	9	62.5	37.5	108
3	3	209	26	29	3	79.4	20.6	130
4	4	183	52	25	7	71.2	28.8	111
5	5	221	14	31	1	83.1	16.9	115
6	6	221	14	31	1	83.1	16.9	113
7	7	223	12	31	1	83.9	16.1	105
8	8	119	36	27	5	76.4	23.6	105
9	9	228	7	31	1	85.8	14.2	106
10	10	228	7	31	1	85.8	14.2	109
11	15	228	7	31	1	85.8	14.2	100
12	20	235	0	32	0	88.0	12.0	95

Terlihat pada tabel 4. bahwa tingkat akurasi maupun kesalahan prediksi akan berbeda atau bervariasi untuk sejumlah data tetangga yang berbeda. Uji coba dilakukan dengan menggunakan jumlah tetangga mulai dari 1 (satu) tetangga terdekat yang memiliki kemiripan dengan data yang ditesting sampai dengan 50 tetangga terdekat ya ng memiliki kemiripan. Berdasarkan uji coba yang dilakukan dengan jumlah data tetangga tersebut diatas dapat diambil tinggak akurasi tertinggi adalah 88% dengan erri 12% dan waktu yang dibutuhkan adalah 95 detik.

Hal ini berarti bahwa semakin banyak jumlah data training dan data tetangga yang digunakan maka akan semakin banyak kemungkinan nilai-nilai atribut yang sama, sehingga diperoleh banyak prediksi yang sama atau prediksi yang memiliki ketepatan yang tinggi.

Dari hasil ujicoba yang telah dilakukan terdapat beberapa hal yang perlu diperhatikan :

1. Aplikasi yang menggunakan algoritma SLIQ selanjutnya disebut sebagai aplikasi lama, dan aplikasi yang menggunakan algoritma Nearest Neighbor selanjutnya disebut sebagai aplikasi baru.
2. Pada aplikasi yang baru, tingkat akurasi diperoleh lebih tinggi dari aplikasi lama. Dimana nilai tingkat akurasi dari kedua aplikasi dengan data yang sama dapat dilihat pada tabel 5.

Tabel 5. Nilai Tingkat Akurasi Prediksi Data Testing

No	Tahun Akademik			
	2005		2006	
	Alg. SLIQ	Alg. NN	Alg. SLIQ	Alg. NN
	41,67 %	78,5 %	63,11 %	86,5 %

3. Pada aplikasi yang baru ,tidak terdapat label kelas lain yang dapat mengganggu dalam perhitungan akurasi
4. Pada aplikasi yang baru, atribut dapat dibatasi hanya muncul satu kali.
5. Pada aplikasi yang baru, semakin banyak data testing dapat diperoleh tingkat akurasi prediksi yang lebih tinggi. Hal ini dapat dilihat dari tingkat akurasi yang diperoleh pada uji coba dengan menggabungkan data tahun akademik tahun 2005/2006 dengan tahun 2006/2007 dan diperoleh tingkat akurasi hingga mencapai 88.00 %.

7. Hasil Prediksi Pada Data Mahasiswa Baru

Dengan menggunakan data training yang telah ada yaitu data akademik tahun 2005/2006 dan 2006/2007 maka dapat dilakukan prediksi terhadap data uji coba untuk memprediksi kinerja mahasiswa baru (dalam hal ini data yang diperoleh adalah tahun 2007). Dari hasil ujicoba prediksi diperoleh hasil akurasi prediksi seperti dapat dilihat pada tabel 6.

Tabel 6. Tingkat akurasi prediksi data mahasiswa baru

	Training 2005		Training 2006	
	Sebenarnya		Sebenarnya	
Prediksi	Mampu	Tidak Mampu	Mampu	Tidak Mampu
Mampu	412	101	412	101
Tidak Mampu	0	0	0	0
	412	101	412	101
Akurasi Prediksi	80,3%		80,3%	
Jumlah Data Training	433		368	
Jumlah Data Testing	513		513	
Jumlah Tetangga	10		10	
Waktu	63 Detik		54 Detik	

PENUTUP

Dari hasil penelitian yang telah dilakukan dapat disimpulkan hal-hal sebagai berikut:

1. Penelitian menghasilkan sebuah aplikasi datamining dengan menggunakan algoritma Nearest Neighbor yang dapat digunakan untuk memprediksi kinerja akademik mahasiswa baru.
2. Data akademik setiap angkatan mempunyai pola yang berbeda-beda yang ditunjukkan oleh tingkat akurasi yang berbeda untuk penentuan dari prediksi kinerja akademik dari setiap data testing yang digunakan.
3. Hasil pengujian tingkat akurasi model tiap data tahun angkatan ternyata mempunyai nilai akurasi yang berbeda, dan hal ini dapat disebabkan karena sebaran nilai data yang berbeda-beda.
4. Hasil pengujian dengan aplikasi sejenis yang telah peneliti buat pada penelitian sebelumnya, menunjukkan skor yang lebih rendah pada aplikasi yang dibuat sebelumnya. Hal ini dapat dikarenakan adanya perbedaan algoritma yang digunakan, dimana dalam penelitian sebelumnya algoritma yang digunakan adalah SLIQ. Dalam algoritma SLIQ dilakukan pembatasan jumlah kemunculan dari atribut pada pohon oleh peneliti.
5. Dari sisi waktu yang dibutuhkan untuk proses mining, jika dilihat dari data flow diagram yang ada, maka pada algoritma Nearest Neigbor lebih sederhana namun membutuhkan waktu yang lebih lama karena harus dilakukan proses training secara berulang untuk setiap data yang akan diprediksi.

DAFTAR PUSTAKA

Agathe, dan Kalina, 2005, *Educational Data Mining: a Case Study*,Pôle Universitaire Léonard de Vinci, France

Al-Radaideh, Q.A., Al-Shawakfa, E.M., dan Al-Najjar, M.I., 2006, *Mining Student Data Using Decision Trees,* The 2006 International Arab Conference on Information Technology (ACIT'2006).

Chandra, E., dan Nandhini, K., 2005, *Predicting Student Performance using Classification Techniques,* Proceedings of SPIT – IEEE Colloquium and International Conference, Mumbai, India Volumen 5, 83.

Han, J., dan Kamber, M., 2001, *Data Mining: Concepts and Techniques,* SunFransisco : Morgan Kaufmann Publishers

Kalles, D., dan Pierrakeas, C., 2006 , *Analyzing Student Performance in Distance Learning with Genetic Algorithms and Decsion Trees,* Hellenic Open University.

Kusrini., Hartati, S.,. 2009. *Perbandingan Metode Nearest Neighbor dan Algoritma C4.5 untuk menganalisis Kemungkinan Pengunduran Diri Calon Mahasiswa Di*

STMIK AMIKOM Yogyakarta. JURNAL DASI ISSN: 1411-3201, Vol. 10 No. 1 Maret 2009

Mehta, M., Agrawal, R., dan Rissanen, J., 1996, *SLIQ: A Fast Scalable Classifier for Data Mining*

Purba, E., 2003, *Pengantar Data Mining*, Universitas Islam Indonesia, Yogyakarta.

Romero, C., Ventura, S., Hervás, C., Gonzales, P., 2006, *Data Mining Algorithms to Classfy Students* ,

Shaufiah, 2005 *klasifikasi dalam data mining menggunakan algoritma SLIQ*. STTTelkom

Tan, P., Steinbach, M., Kumar, V., 2006, *Introduction to Data Mining*, Pearson Education.