

**APLIKASI DATA MINING
UNTUK MENAMPILKAN INFORMASI
TINGKAT KELULUSAN MAHASISWA**

Yuli Asriningtias, Rodhyah Mardhiyah

Program Studi Teknik Informatika

Fakultas Bisnis & Teknologi Informasi, Universitas Teknologi Yogyakarta

Jl. Glagahsari No 63 Yogyakarta

Telp : (0274) 373955

Email : dheeyaah@gmail.com, yuli_asriningtias@yahoo.com

Abstrak

Perguruan tinggi dituntut memiliki keunggulan bersaing dengan memanfaatkan sumber dayanya, termasuk sumber daya manusia dalam hal ini adalah mahasiswa. Tidak semua mahasiswa dapat menyelesaikan study tepat waktu, disamping IPK yang beragam. Lama waktu mahasiswa dalam menempuh studi dan IPK menjadi salah satu faktor tingkat keunggulan sebuah Perguruan Tinggi. Nilai potensi tersebut dapat digali menggunakan teknik data mining.

Data mining adalah kegiatan menemukan pola yang menarik dari data dalam jumlah besar, data dapat disimpan dalam database, data warehouse, atau penyimpanan informasi lainnya. Data warehouse merupakan penyimpanan data yang berorientasi objek, terintegrasi, mempunyai variant waktu, dan menyimpan data dalam bentuk nonvolatile sebagai pendukung manajemen dalam proses pengambilan keputusan.

Penelitian ini dikembangkan dengan cara menscan data pada database secara langsung sehingga menghasilkan informasi yang dibutuhkan. Aplikasi data mining ini dibangun menggunakan bahasa pemrograman Borland Delphi 7 dan menggunakan database SQL Server 2000 sebagai media penyimpanan data. Hasil dari penelitian bahwa dapat diketahui tingkat ketepatan waktu dan nilai kelulusan mahasiswa yang berelasi dengan atribut data masuk mahasiswa.

Kata Kunci : *Data mining, data warehouse, kelulusan mahasiswa.*

1. PENDAHULUAN

Informasi merupakan suatu elemen penting dalam kebutuhan kehidupan sehari-hari. Untuk mendapatkan sebuah informasi yang penting dan akurat sering kali tidak mudah kita dapatkan. Informasi yang tersedia dalam jumlah yang besar terkadang masih harus kita gali terlebih dahulu agar informasi tersebut dapat menyajikan informasi yang tepat dan sesuai dengan kebutuhan. Untuk dapat menggali informasi yang berpotensi dari gudang data tidak bisa hanya mengandalkan data operasional saja, namun diperlukan suatu analisis data yang tepat sehingga menghasilkan informasi yang lebih berharga sehingga dapat menunjang kegiatan operasional. Untuk membantu mempermudah para pengambil keputusan dalam menganalisis dan mengekstraksi data maka lahirlah cabang ilmu baru yang disebut *Data Mining*.

Perguruan tinggi dituntut memiliki keunggulan bersaing dengan memanfaatkan semua sumber daya yang dimiliki. Salah satunya adalah Sumber Daya Manusia (SDM), dalam hal ini adalah mahasiswa. Standar lama studi mahasiswa Program Sarjana reguler (S1) adalah 8 semester, namun teknis di lapangan banyak dijumpai mahasiswa lulus melebihi dari yang telah dijadwalkan. Untuk itu perlunya penggalian data dengan memanfaatkan data induk mahasiswa dan data kelulusan sehingga dapat diketahui informasi tingkat kelulusan mahasiswa.

Tahap-tahap *data mining* ada 6 yaitu :

1.1. Pembersihan data (*data cleaning*)

Pembersihan data merupakan proses menghilangkan noise dan data yang tidak relevan. Pada umumnya data yang diperoleh, baik dari *database* memiliki isian-isian yang tidak sempurna seperti data yang hilang, data yang tidak valid atau juga hanya sekedar salah ketik. Data-data yang tidak relevan itu juga lebih baik dibuang. Pembersihan data juga akan mempengaruhi performansi dari teknik *data mining* karena data yang ditangani akan berkurang jumlah dan kompleksitasnya.

1.2. Integrasi data (*data integration*)

Integrasi data merupakan penggabungan data dari berbagai *database* ke dalam satu *database* baru. Integrasi data perlu dilakukan secara cermat karena kesalahan pada integrasi data bisa menghasilkan hasil yang menyimpang dan bahkan menyesatkan pengambilan aksi nantinya. Sebagai contoh bila integrasi data berdasarkan jenis produk ternyata menggabungkan produk dari kategori yang berbeda maka akan didapatkan korelasi antar produk yang sebenarnya tidak ada.

1.3. Seleksi Data (*Data Selection*)

Data yang ada pada *database* sering kali tidak semuanya dipakai, oleh karena itu hanya data yang sesuai untuk dianalisis yang akan diambil dari *database*. Sebagai contoh, sebuah kasus yang meneliti faktor kecenderungan orang membeli dalam kasus *market basket analysis*, tidak perlu mengambil nama pelanggan, cukup dengan id pelanggan saja.

1.4. Transformasi data (*Data Transformation*)

Data diubah atau digabung ke dalam format yang sesuai untuk diproses. Sebagai contoh beberapa metode standar seperti analisis asosiasi dan *clustering* hanya bisa menerima input data kategorikal. Karenanya data berupa angka numerik yang berlanjut perlu dibagi-bagi menjadi beberapa interval. Proses ini sering disebut transformasi data.

1.5. Proses mining

Merupakan suatu proses utama saat metode diterapkan untuk menemukan pengetahuan berharga dan tersembunyi dari data.

1.6. Evaluasi pola (*pattern evaluation*)

Untuk mengidentifikasi pola-pola menarik kedalam *knowledge based* yang ditemukan. Dalam tahap ini hasilnya berupa pola-pola yang khas maupun model prediksi dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai.

1.7. Presentasi pengetahuan (*knowledge presentation*)

Merupakan visualisasi dan penyajian pengetahuan mengenai metode yang digunakan untuk memperoleh pengetahuan yang diperoleh pengguna. Tahap terakhir adalah bagaimana memformulasikan keputusan atau aksi dari hasil analisis yang didapat. Karenanya presentasi dalam bentuk pengetahuan yang bisa dipahami semua orang adalah satu tahapan yang diperlukan. Dalam presentasi ini, visualisasi juga bisa membantu mengkomunikasikan hasil *data mining* (Han, 2006).

2. METODE

Dengan definisi *data mining* yang luas, ada banyak jenis metode analisis yang dapat digolongkan dalam *data mining*.

2.1. Association rules

Association rules (aturan asosiasi) atau *affinity analysis* (analisis afinitas) berkenaan dengan studi tentang “apa bersama apa”. Sebagai contoh dapat berupa berupa studi transaksi di supermarket, misalnya seseorang yang membeli susu bayi juga membeli sabun mandi. Pada kasus ini berarti susu bayi bersama dengan sabun mandi. Karena awalnya berasal dari studi tentang *database* transaksi pelanggan untuk menentukan kebiasaan suatu produk dibeli bersama produk apa, maka aturan asosiasi juga sering dinamakan *market basket analysis*. Aturan asosiasi ingin memberikan informasi tersebut dalam bentuk hubungan “if-then” atau “jika-maka”. Aturan ini dihitung dari data yang sifatnya probabilistik (Santoso, 2007).

Ada beberapa algoritma yang sudah dikembangkan mengenai aturan asosiasi, namun ada satu algoritma klasik yang sering dipakai yaitu algoritma *apriori*. Ide dasar dari algoritma ini adalah dengan mengembangkan *frequent itemset*. Dengan menggunakan satu item dan secara rekursif mengembangkan *frequent itemset* dengan dua item, tiga item dan seterusnya hingga *frequent itemset* dengan semua ukuran. Untuk mengembangkan *frequent set* dengan dua item, dapat menggunakan

frequent set item. Alasannya adalah bila set satu item tidak melebihi *support minimum*, maka sembarang ukuran itemset yang lebih besar tidak akan melebihi *support minimum* tersebut. Secara umum, mengembangkan set dengan *fc-item* menggunakan frequent set dengan *k-1* item yang dikembangkan dalam langkah sebelumnya. Setiap langkah memerlukan sekali pemeriksaan ke seluruh isi *database*. Dalam asosiasi terdapat istilah *antecedent* dan *consequent*, *antecedent* untuk mewakili bagian “jika” dan *consequent* untuk mewakili bagian “maka”. Dalam analisis ini, *antecedent* dan *consequent* adalah sekelompok item yang tidak punya hubungan secara bersama (Santoso, 2007).

$$S = \frac{\Sigma(Ta+Tc)}{\Sigma(T)} \dots\dots\dots(2.1)$$

Keterangan :

S = Support

$\Sigma(Ta + Tc)$ = Jumlah transaksi yang mengandung *antecedent* dan *consequent*

$\Sigma(T)$ = Jumlah transaksi

$$C = \frac{\Sigma(Ta+Tc)}{\Sigma(Ta)} \dots\dots\dots(2.2)$$

Keterangan :

C = Confidence

$\Sigma(Ta + Tc)$ = Jumlah transaksi yang mengandung *antecedent* dan *consequent*

$\Sigma(Ta)$ = Jumlah transaksi yang mengandung *antecedent*

Langkah pertama algoritma apriori adalah, support dari setiap item dihitung dengan men-scan *database*. Setelah *support* dari setiap item didapat, item yang memiliki *support* lebih besar dari *minimum support* dipilih sebagai pola frekuensi tinggi dengan panjang 1 atau sering disingkat 1-itemset. Singkatan k-itemset berarti satu set yang terdiri dari k item.

Iterasi kedua menghasilkan 2 item set yang tiap set-nya memiliki dua item. Pertama dibuat kandidat 2 item set dari kombinasi semua 1 item set. Lalu untuk tiap kandidat 2 item set ini dihitung *support*-nya dengan men-scan *database*. *Support* artinya jumlah transaksi dalam *database* yang mengandung kedua item dalam kandidat 2 item set. Setelah *support* dari semua kandidat 2 item set didapatkan, kandidat 2 item set yang memenuhi syarat *minimum support* dapat ditetapkan sebagai 2 item set yang juga merupakan pola frekuensi tinggi dengan panjang 2 (Pramudiono, 2007).

Secara ringkas algoritma *apriori* sebagai berikut :

```

Create L1 = set of supported itemsets of cardinality one
Set k to 2
while (Lk-1 ≠ ∅) {
    Create Ck from Lk-1
    Prune all the itemsets in Ck that are not
    supported, to create Lk
    Increase k by 1
}
The set of all supported itemsets is L1 ∪ L2 ∪ ... ∪ Lk
    
```

Contoh proses *mining* untuk mengetahui hubungan tingkat kelulusan dengan jurusan yang dikelompokkan dalam beberapa kategori adalah sebagai berikut

Tabel 1. Tabel Kategori kelulusan

Kategori	Keterangan
A1	lama studi 4 tahun atau kurang dari 4 tahun dan IPK 3,51 – 4,00
A2	lama studi 4 tahun atau kurang dari 4 tahun dan IPK 2,76 – 3,50
A3	lama studi 4 tahun atau kurang dari 4 tahun dan IPK kurang dari 2,76
B1	lama studi lebih dari 4 tahun dan IPK 3,51 – 4,00
B2	lama studi lebih dari 4 tahun dan IPK 2,76 – 3,50
B3	lama studi lebih dari 4 tahun dan IPK kurang dari 2,76

Tabel 2 Tabel data

NIM	Kategori Kelulusan	Asal Jurusan
3075111051	A1	IPA
3075111052	A2	IPS
3075111053	A1	IPA
3075111054	A3	IPS
3075111055	B2	IPS
3075111056	A3	IPS
3075111057	A3	IPS

307511105 8	A2	IPA
307511105 9	A2	IPA
307511106 0	A2	IPA
307511106 1	B2	IPS

Dari data awal tersebut didapat kandidat pertama (C1) seperti pada tabel 3.

Tabel 3 Kandidat Pertama (C1)

Itemset	Count
A1	2
A2	4
A3	3
B2	2
IPA	5
IPS	6

Ditetapkan *threshold* = 3, maka kandidat yang nilainya kurang dari 3 akan dihapus. Sehingga, didapat hasil seperti pada tabel 4 (L1).

Tabel 4 hasil setelah *threshold* ditetapkan (L1).

Itemset	Count
A2	4
A3	3
IPA	5
IPS	6

Dari tabel 4 didapat kandidat kedua (C2) seperti pada tabel 5.

Tabel 5 Kandidat kedua (C2)

Itemset	Count
A2, IPA	3
A2, IPS	1
A3, IPA	0
A3, IPS	3

Setelah ditetapkan *threshold* menghasilkan data seperti pada tabel 6.

Tabel 6 Hasil kedua (L2)

Itemset	Count
A2, IPA	3
A3, IPS	3

Dari pada tabel 6 dapat diambil hasil sebagai berikut :
Support A2, IPA = $Count(A2, IPA) / \text{jumlah transaksi} = 3/11$.
Support A3, IPS = $Count(A3, IPS) / \text{jumlah transaksi} = 3/11$.
Confidence A2, IPA = $Count(A2, IPA) / Count(A2) = 3/4$.
Confidence A3, IPS = $Count(A3, IPS) / Count(A3) = 3/3$.

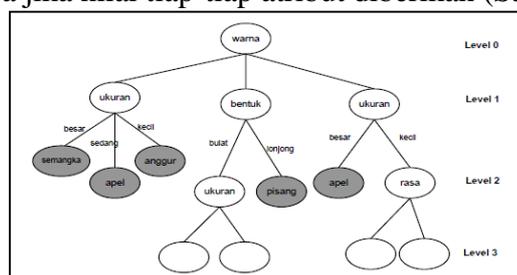
Dapat dilihat bahwa proses *mining* hubungan tingkat kelulusan dengan jurusan mahasiswa dengan *threshold* 3 menghasilkan hubungan A2, IPA mempunyai nilai *support* = 3/11, *Confidence* = 3/4 dan hubungan A3, IPS mempunyai nilai *support* = 3/11, *Confidence* = 3/3.

IPA mempunyai tingkat kelulusan A2 dan IPS mempunyai tingkat kelulusan A3 sehingga dapat disimpulkan bahwa mahasiswa yang melalui jurusan IPA mempunyai tingkat kelulusan lebih bagus dibanding mahasiswa yang melalui jurusan IPS.

Selain algoritma *apriori*, terdapat juga algoritma lain seperti *FP-Grwoth*. Perbedaan algoritma *apriori* dengan *FP-Growth* pada banyaknya *scan database*. Algoritma *apriori* melakukan *scan database* setiap kali iterasi sedangkan algoritma *FP-Growth* hanya melakukan sekali di awal (Bramer, 2007).

2.2. Decision Tree

Dalam *decision tree* tidak menggunakan *vector* jarak untuk mengklasifikasikan obyek. Seringkali data observasi mempunyai atribut-atribut yang bernilai nominal. Seperti yang diilustrasikan pada gambar 2.6, misalkan obyeknya adalah sekumpulan buah-buahan yang bisa dibedakan berdasarkan atribut bentuk, warna, ukuran dan rasa. Bentuk, warna, ukuran dan rasa adalah besaran nominal, yaitu bersifat kategoris dan tiap nilai tidak bisa dijumlahkan atau dikurangkan. Dalam atribut warna ada beberapa nilai yang mungkin yaitu hijau, kuning, merah. Dalam atribut ukuran ada nilai besar, sedang dan kecil. Dengan nilai-nilai atribut ini, kemudian dibuat *decision tree* untuk menentukan suatu obyek termasuk jenis buah apa jika nilai tiap-tiap atribut diberikan (Santoso, 2007).

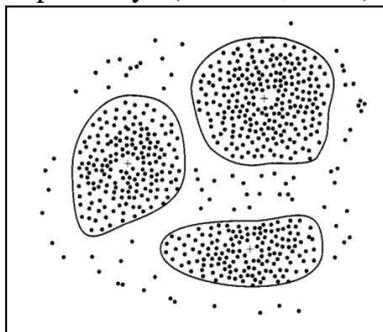


Gambar 1 Decision Tree.

2.3. Clustering

Tujuan utama dari metode *clustering* adalah pengelompokan sejumlah data/obyek ke dalam *cluster* (*group*) sehingga dalam setiap *cluster* akan berisi data yang semirip mungkin seperti diilustrasikan pada gambar 2.7. Dalam *clustering* metode ini berusaha untuk menempatkan obyek yang mirip (jaraknya dekat) dalam satu klaster dan membuat jarak antar klaster

sejauh mungkin. Ini berarti obyek dalam satu *cluster* sangat mirip satu sama lain dan berbeda dengan obyek dalam *cluster-cluster* yang lain. Dalam metode ini tidak diketahui sebelumnya berapa jumlah *cluster* dan bagaimana pengelompokannya (Santoso, 2007).



Gambar 2 *Clustering*

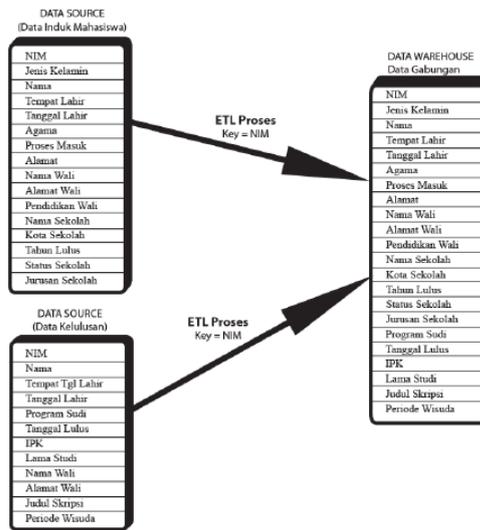
3. HASIL DAN PEMBAHASAN

Perangkat yang digunakan pada tahap pengembangan adalah sebuah perangkat komputer netbook dengan spesifikasi :AMD E450 APU with Radeon™ HD Graphics (2 CPUs) 1,6 GHz, RAM 2Gb, HD 500 Gb.

Spesifikasi Kebutuhan Fungsional

Spesifikasi kebutuhan fungsional pada Aplikasi *Data Mining* ini merujuk pada kebutuhan akan perancangan *data mining*, seperti yang tertera berikut ini :

1. Dapat menggabungkan data yang akan diproses *mining* dari data kelulusan dan data induk mahasiswa
2. Dapat menghapus data-data yang tidak relevan serta atribut yang tidak dipakai
3. Dapat merubah data menjadi data yang siap diproses
4. Dapat memproses data untuk *dimining* meliputi :
 - a. Hubungan tingkat kelulusan dengan asal sekolah.
 - b. Hubungan tingkat kelulusan dengan asal kabupaten.
 - c. Hubungan tingkat kelulusan dengan jurusan.
- d. Dapat menampilkan hasil proses *mining* dengan nilai *support* dan *confidence*

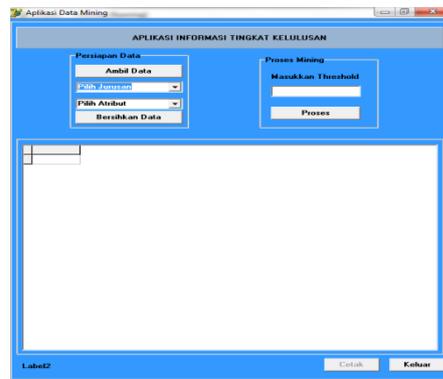


Gambar 3 Proses load data.

Pembahasan

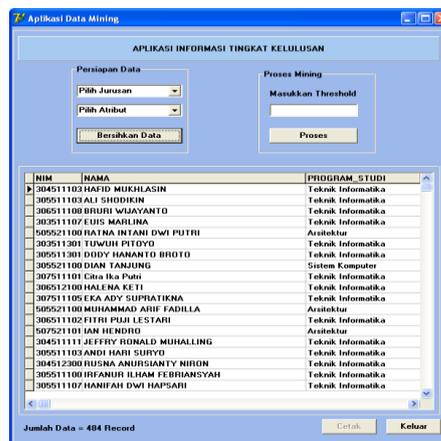
Hasil rancangan antarmuka Aplikasi *Data Mining* diimplementasikan dalam satu form. Form merupakan halaman yang berisi perintah pengambilan data pemilihan atribut data induk mahasiswa, input *threshold*, perintah proses *mining* dan tombol keluar aplikasi sekaligus hasil proses *data mining* yaitu tabel nilai *support* dan *confidence*. *Threshold* bernilai *default* yaitu 1. Tampilan dari awal form dapat dilihat pada gambar 4.

Halaman utama merupakan halaman yang pertama kali di tampilkan.



Gambar 4 Halaman Awal

Halaman aplikasi saat dilakukan proses *button* bersihkan data menampilkan data bersih seluruh mahasiswa dan seluruh angkatan serta seluruh atribut.



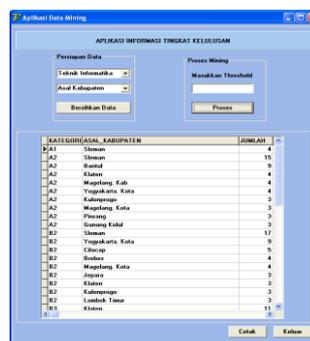
Gambar 5. Data bersih seluruh mahasiswa seluruh angkatan.

Menampilkan data mahasiswa jurusan Teknik Informatika dan menghapus data kosong pada *field* kabupaten.



Gambar 6 Data bersih mahasiswa dengan atribut kabupaten.

Menampilkan data hasil klasifikasi kelulusan mahasiswa jurusan Teknik Informatika dengan atribut asal kabupaten dan nilai *minimum support* atau *threshold* 3.



Gambar 7 Data hasil klasifikasi dengan threshold 3.

Menampilkan data laporan klasifikasi kelulusan mahasiswa jurusan Teknik Informatika dengan atribut asal kabupaten dan nilai *minimum support* 3.

NO	KTGR	ASAL_KABUPATEN	JML	SUPPORT	CONFIDENCE
1	A1	Sleman	4	0.138	0.009
2	A2	Sleman	15	0.140	0.034
3	A3	Bantul	9	0.084	0.020
4	A4	Klaten	4	0.037	0.009
5	A5	Magelang, Kab	4	0.037	0.009
6	A6	Yogyakarta, Kota	4	0.037	0.009
7	A7	Kulonprogo	3	0.028	0.007
8	A8	Magelang, Kota	3	0.028	0.007
9	A9	Pinrang	3	0.028	0.007
10	A10	Gunung Kidul	3	0.028	0.007
11	B1	Sleman	17	0.124	0.038
12	B2	Yogyakarta, Kota	9	0.066	0.020
13	B3	Cilacap	5	0.036	0.011
14	B4	Brebes	4	0.029	0.009
15	B5	Magelang, Kota	4	0.029	0.009
16	B6	Jepara	3	0.022	0.007
17	B7	Klaten	3	0.022	0.007
18	B8	Kulonprogo	3	0.022	0.007

Gambar 8 Laporan hasil klasifikasi data mahasiswa jurusan Teknik Informatika dengan atribut asal kabupaten dan *threshold* 3.

Menampilkan data laporan klasifikasi kelulusan mahasiswa seluruh program studi, seluruh angkatan, dan seluruh atribut dan nilai *minimum support* 2.

NO	KTGR	ASAL SEKOLAH	ASAL KABUPATEN	ASAL JURUSAN	JML	SUPPORT	CONFIDENCE
1	A2	AMIK 'PROACTIVE'	Klaten	IPA	2	0.020	0.004
2	B1	SMA HARMONI	Batam, Kota	IPS	2	0.012	0.004
3	B2	SMA MUHAMMADIYAH7	Yogyakarta, Kota	IPA	2	0.012	0.004
4	B3	SMA N2 NGAGLIK	Sleman	IPS	2	0.012	0.004
5	B4	SMA N2 SELONG	Lombok Timur	IPA	2	0.012	0.004
6	B5	MAN2	Kebumen	IPS	2	0.014	0.004
7	B6	SMA N1	Benyumas	IPS	2	0.014	0.004
8	B7	SMA N3	Sintang	IPS	2	0.014	0.004
9	B8	SMA PIRI 2	Yogyakarta, Kota	IPS	2	0.014	0.004
10	B9	SMK MARSUDI LUAR 2	Yogyakarta, Kota	MESIN OTOMOTIF	2	0.014	0.004

Gambar 9. Laporan hasil klasifikasi kelulusan mahasiswa seluruh angkatan, seluruh program studi, dan atribut dengan *threshold* 2.

4. KESIMPULAN

Kesimpulan yang dapat diambil adalah Aplikasi *Data Mining* ini dapat digunakan untuk menampilkan informasi tingkat kelulusan. Informasi yang ditampilkan berupa nilai *support* dan *confidence* hubungan antara tingkat kelulusan dengan data induk mahasiswa. Semakin tinggi nilai *confidence* dan *support* maka semakin kuat nilai hubungan antar atribut. Data induk mahasiswa yang diproses *mining* meliputi data ata asal sekolah, data kabupaten

mahasiswa, data jurusan, dan data program studi. Hasil dari proses *data mining* ini dapat digunakan sebagai pertimbangan dalam mengambil keputusan lebih lanjut tentang faktor yang mempengaruhi tingkat kelulusan khususnya faktor dalam data induk mahasiswa.

DAFTAR PUSTAKA

- [1] Santosa, Budi., 2007. "Data Mining Teknik Pemanfaatan Data Untuk Keperluan Bisnis Teori dan Aplikasi". Graha Ilmu : Yogyakarta.
- [2] <http://bow-masbow.blogspot.com/2010/10/algorithm-apriori-apriori-algorithm.html>, diakses pada Desember 2011.
- [3] <http://digilib.petra.ac.id/viewer.php?page=1&submit.x=0&submit.y=0&qual=high&fname=/jiunkpe/s1/info/2007/jiunkpe-ns-s1-2007-26403150-8823-clustering-chapter2.pdf>, diakses pada Desember 2011.
- [4] http://www.google.co.id/url?sa=t&rct=j&q=datawarehouse%20adalah&source=web&cd=3&ved=0CDMQFjAC&url=http%3A%2F%2Fzakki.dosen.narotama.ac.id%2Ffiles%2F2012%2F02%2FDefinisi-Data-Warehouse.doc&ei=T4FET8WSLInOrQfV8NieDw&usg=AFQjCNGO91hKU1gt5NZ_z_5VgpNIHNYLmQ&cad=rja, diakses pada Desember 2011.
- [5] <http://sarwojowo.net/component/content/article/39-database-/71-pengenalan-datawarehouse-.html>, diakses pada Desember 2011.
- [6] <http://www.docstoc.com/docs/20182632/Datawarehousing-dan-Datamining-Asosiasi-Nilai-Mahasiswa-Masa-Skripsi>, diakses pada November 2011.
- [7] http://www.gunadarma.ac.id/library/articles/postgraduate/information-system/Sistem%20Informasi%20Akuntansi/Artikel_92106032.pdf, diakses pada November 2011.
- [8] <http://www.scribd.com/vemerald/d/53676182/43-III-1-5-Penggunaan-Algorithm-Apriori>, diakses pada Desember 2011.