

Analisis Perbandingan Algoritma *Decision Tree (C4.5)* Dan *K-Naive Bayes* Untuk Mengklasifikasi Penerimaan Mahasiswa Baru Tingkat Universitas

Suyadi

Magister Teknik Informatika
Universitas Amikom Yogyakarta
E-mail: suyadia00@gmail.com

Arief Setyanto dan Hanif Al Fattah

Magister Teknik Informatika
Universitas Amikom Yogyakarta
E-mail : arief_s@amikom.ac.id, hanif.a@amikom.ac.id

ABSTRAK

Profil PMB (Penerimaan Mahasiswa Baru) mahasiswa dari beberapa periode mempunyai data yang melimpah sehingga bisa digunakan untuk penelitian. Data tersebut berupa keterangan mahasiswa dari jurusan asal, NEM dan jurusan sekarang. Mengklasifikasi data profil PMB mahasiswa di tingkat Universitas di Yogyakarta dapat mengetahui mayoritas peserta didik. Membandingkan beberapa algoritma sangat dibutuhkan untuk mengetahui algoritma terbaik. Klasifikasi merupakan algoritma pengelompokan yang mempunyai beberapa algoritma diantaranya adalah *Decision Tree (C4.5)* dan *K-Naive Bayes*. *Decision Tree (C4.5)* merupakan algoritma dengan pohon keputusan, sedangkan *K-Naive Bayes* merupakan algoritma kemungkinan yang akan terjadi. Analisa ini menggunakan *Rapidminer* yang merupakan software analisa data dengan fitur beberapa algoritma yang mudah untuk dioperasikan. Kedua algoritma mempunyai hasil dengan data yang besar yaitu 1504 mahasiswa, *Decision tree (C4.5)* mempunyai akurasi 81,84% dan akurasi error 18,16%, sedangkan *K-Naive Bayes* 85,12% dan akurasi error 14,88%. Sedangkan dengan data yang lebih kecil *Decision tree (C4.5)* mempunyai akurasi 100% sedangkan *K-Naive Bayes* mempunyai akurasi yang sama dengan *Decision Tree (C4.5)* yaitu 100%.

Kata Kunci: *Data mining*, Klasifikasi, Komparasi, *Decision Tree (C4.5)*, *K-Naive Bayes*.

ABSTRACT

Profile of PMB (New Student Admissions) students from several periods have abundant data that can be used for research. The data is in the form of student information from the majors of origin, NEM and majors now. Classifying the PMB profile data of students at the University level in Yogyakarta can know the majority of learners. Comparing some algorithms is needed to find out the best algorithm. Classification is a grouping algorithm that has several algorithms such as *Decision Tree (C4.5)* and *K-Naive Bayes*. *Decision Tree (C4.5)* is an algorithm with decision tree, while *K-Naive Bayes* is the likely algorithm that will occur. This analysis uses *Rapidminer* which is a data analysis software with features of several algorithms that are easy to operate. Both algorithms have results with large data of 1504 students, *Decision tree (C4.5)* has an accuracy of 81.84% and an error accuracy of 18.16%, while *K-Naive Bayes* 85.12% and accuracy of error 14.88%. Whereas with smaller data the *Decision tree (C4.5)* has 100% accuracy whereas *K-Naive Bayes* has the same accuracy as *Decision Tree (C4.5)* that is 100%.

Keywords: *Data mining*, Classification, Comparison, *Decision Tree (C4.5)*, *K-Naive Bayes*

1. PENDAHULUAN

Membandingkan antar algoritma Algoritma klasifikasi untuk menganalisa dan mengklasifikasikan data PMB historikal mahasiswa, contoh : Algoritma *Decision Tree (C4.5)* dan *K-Naive Bayes* paling banyak digunakan untuk mengklasifikasi dan memprediksi. Tujuan utama penelitian ini adalah mengetahui pentingnya data historikal sebagai pengaruh terhadap akurasi untuk mengklasifikasi penerimaan mahasiswa baru. Makalah ini juga bertujuan mengidentifikasi algoritma terbaik di antara dua pilihan algoritma klasifikasi *Decision Tree (C4.5)* dan *K-Naive Bayes*. Hasilnya akan menunjukkan keakurasian yang dicapai dengan *Parameter Standart* dalam mengklasifikasikan tingkat risiko secara akurat.

Decision tree merupakan suatu teknik menemukan kumpulan pola atau fungsi yang mendeskripsikan serta memisahkan kelas data yang satu dengan yang lainnya untuk menyatakan objek tersebut masuk pada kategori tertentu dengan melihat kelakuan dan atribut dari kelompok yang telah didefinisikan. *Decision Tree (C4.5)* merupakan algoritma untuk mengklasifikasikan apa yang akan terjadi di masa depan untuk memperoleh hasil yang maksimal. *Decision Tree (C4.5)* adalah algoritma pembelajaran yang mengimplementasikannya pencarian heuristik (penemuan) yang cukup sebanding menggunakan pohon keputusan [1].

Algoritma *K-Naive Bayes* bahwa dalam kumpulan data dua *attribute* mungkin saling bergantung satu sama lain tapi ini ketergantungan dapat didistribusikan secara merata di setiap atribut. *K-Naive Bayes* merupakan algoritma *data mining* yang dapat digunakan untuk melakukan klasifikasi, dan merupakan algoritma yang akan memberikan hasil kemungkinan yang akan terjadi di masa akan datang dari hasil klasifikasi. *K-Naive Bayes* pernah diterapkan untuk mengklasifikasikan data berdimensi tinggi. Data berdimensi tinggi yang dimaksud adalah data yang memiliki proporsi atribut yang tidak relevan. Percobaan dilakukan dengan menggunakan bagian dokumen dan *gen microarray* data yang menunjukkan efisiensi yang sangat memuaskan dengan presentase sekitar 98% [2].

Dalam penelitian mereka [3], menyajikan *add-on* ke kerangka *Rapidminer* yang mendukung pengguna selama tahap pemodelan dengan merekomendasikan operasi tambahan untuk dimasukkan ke dalam mesin yang saat ini dikembangkan dengan alur kerja. Model R [4] adalah software statistik yang bekerja dengan formulasi model matematis sementara *Rapidminer* dapat bekerja secara efisien dengan menggunakan *operator* yang disediakan dalam bentuk blok untuk membuat alur kerja analisis data yang spesifik atau karakteristik produk. Penelitian ini menggunakan *Rapidminer* karena data yang diteliti berupa profil PMB mahasiswa dan terdiri dari beberapa atribut yang berisi *value* yang isinya abjad. Sehingga dibutuhkan *software data mining* untuk pengolah data berupa abjad untuk memudahkan.

2. METODOLOGI PENELITIAN

2.1 Data Mining

Data Mining adalah Serangkaian proses untuk menggali nilai tambah berupa informasi yang selama ini tidak diketahui secara manual dari suatu basis data dengan melakukan penggalian pola-pola dari data dengan tujuan untuk memanipulasi data menjadi informasi yang lebih berharga yang diperoleh dengan cara mengekstraksi dan mengenali pola yang penting atau menarik dari data yang terdapat dalam basis data. *Data cleaning* (untuk menghilangkan noise data yang tidak konsisten). *Data integration* (di mana sumber data yang terpecah dapat disatukan). *Data selection* (di mana data yang relevan dengan tugas analisis dikembalikan ke dalam database). *Data transformation* (di mana data berubah atau bersatu menjadi bentuk yang tepat untuk menambang dengan ringkasan performa atau operasi agresif). *Knowledge Discovery* (proses esensial di mana metode yang intelegen digunakan untuk mengekstrak pola data). *Pattern evolution* (untuk mengidentifikasi pola yang benar-benar menarik yang mewakili pengetahuan berdasarkan atas beberapa tindakan yang menarik). *Knowledge presentation* (di mana gambaran teknik visualisasi

dan pengetahuan digunakan untuk memberikan pengetahuan yang telah ditambah kepada user).

2.2 Klasifikasi

Klasifikasi ialah suatu kegiatan yang mengelompokkan benda yang memiliki beberapa ciri yang sama dan memisahkan benda yang tidak sama. Dalam kaitannya di dunia perpustakaan klasifikasi diartikan sebagai kegiatan pengelompokkan bahan pustaka berdasarkan ciri-ciri yang sama, misalnya pengarang, fisik, isi sebagainya.. Pada dasarnya di perpustakaan dikenal ada 2 (dua) jenis kegiatan klasifikasi. Klasifikasi Fundamental (*Fundamental Classification*) yaitu klasifikasi bahan pustaka berdasarkan subyek/isi buku, sebab pada dasarnya pemakai perpustakaan lebih banyak mencari informasi tentang subyek tertentu. Klasifikasi Artifisial (*Artificial Classification*), yaitu klasifikasi bahan pustaka berdasarkan ciri-ciri yang ada pada bahan pustaka. Misalnya klasifikasi berdasarkan warna, ukuran dan sebagainya. *Decision Tree (C4.5)* adalah sebuah struktur pohon, di mana setiap *node* pohon merepresentasikan atribut yang telah diuji, setiap cabang merupakan suatu pembagian hasil uji, dan *node* daun (*leaf*) merepresentasikan kelompok kelas tertentu. *Level node* teratas dari sebuah *Decision Tree (C4.5)* adalah *node* akar (*root*) yang biasanya berupa atribut yang paling memiliki pengaruh terbesar pada suatu kelas tertentu.

Rumus untuk mencari *Entropy* :

$$Entropy(S) = \sum_{i=1}^x - p_i \times \log^2 p_i \quad (1)$$

Keterangan Konsep *Entropy* :

- S = Himpunan kasus
- i = Fitur
- n = Jumlah partisi S
- pi = Proporsi Si terhadap S

Rumus untuk mencari *Gain* :

$$Gain(S, a) = Entropy(S) \sum_{i=1}^x \frac{|S_i|}{|S|} \times Entropy(S_i) \quad (2)$$

Keterangan Konsep *Gain* :

- S = Himpunan kasus
- I = Atribut
- n = Jumlah partisi atribut A
- Si = Proporsi Si terhadap S
- S = Jumlah kasus dalam S

K-Naive Bayes merupakan sebuah pengklasifikasi probabilitas sederhana yang mengaplikasikan *Teorema Bayes* dengan asumsi ketidak tergantungan (*independent*) yang tinggi. Keuntungan penggunaan *K-Naive Bayes* adalah bahwa metoda ini hanya membutuhkan jumlah data pelatihan (*training data*) yang kecil untuk menentukan *estimasi parameter* yang diperlukan dalam proses pengklasifikasian. Untuk situasi ini, menilai kemungkinan bahwa sebuah data yang sudah ada untuk kemungkinan sebuah keputusan menggunakan algoritma *K-Naive Bayes* [5].

Rumus mencari Probabiliti :

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \quad (3)$$

Di mana :

- X : Data dengan *class* yang belum diketahui
- H : Hipotesis data merupakan suatu *class* spesifik
- $P(H|X)$: Probabilitas hipotesis H berdasar kondisi X (posteriori probabilitas)
- $P(H)$: Probabilitas hipotesis H (prior probabilitas)
- $P(X|H)$: Probabilitas X berdasarkan kondisi pada hipotesis H
- $P(X)$: Probabilitas X

Rumus Mencari Peluang :

$$P(X_i = x_i | Y = y_j) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}} \quad (4)$$

Di mana :

- P : Peluang
- X_i : *Atribut* ke i
- x_i : Nilai *atribut* ke i
- Y : Kelas yang dicari
- y_i : Sub kelas Y yang dicari
- μ : *mean*, menyatakan rata – rata dari seluruh *atribut*
- σ : *Devisasi standart*, menyatakan varian dari seluruh *atribut*

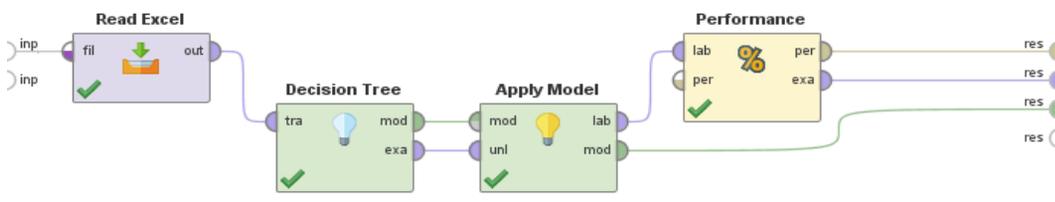
2.3 Rapidminer

Rapidminer adalah software analisis data *mining* yang digunakan untuk menganalisa data dan mendukung berbagai teknik data *mining*. *Rapidminer* ini digunakan untuk aplikasi industri, penelitian, pelatihan, pengembangan aplikasi dan pendidikan. *Rapidminer* mengandung sekitar 100 skema pembelajaran untuk analisis klastering, klasifikasi dan *regresi*. Ini mendukung sekitar 22 format file seperti .xls, .csv dan masih banyak data pendukung lainnya [6].

2.4 Implementasi Klasifikasi Profil PMB Mahasiswa tingkat

2.4.1 Decision Tree (C4.5)

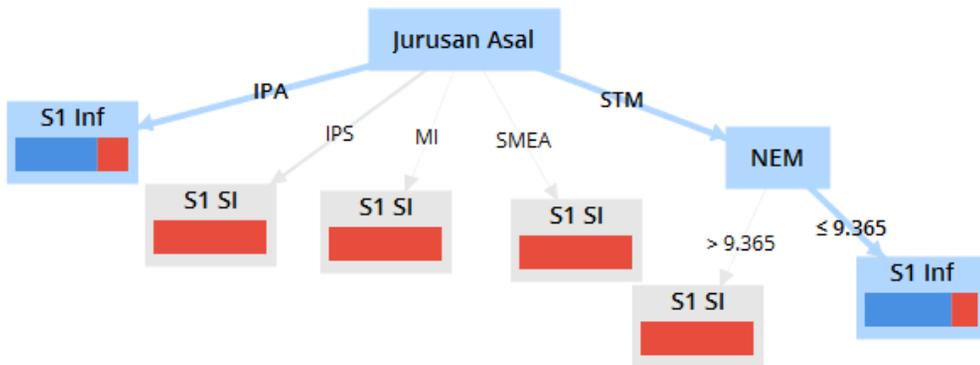
Data profil PMB mahasiswa yang sudah ada dari file excel dan dikoneksikan ke *Rapidminer* melalui *read excel*. Di *Rapidminer* fitur yang digunakan untuk menampilkan fitur yang diinginkan disebut vektor. Pada penelitian algoritma ini atribut yang digunakan adalah Jurusan sekarang sebagai klas tujuan, jurusan asal dan Nem. Data yang diteliti sebanyak 1504 profil PMB mahasiswa.



Gambar 1. Struktur Untuk Analisa Decision Tree (C4.5)

Pada Gambar 1. Data dari file excel diimport menggunakan menggunakan fitur *Read Excel*, setelah itu dikoneksikan ke fitur *Decision Tree* untuk menerapkan data yang diteliti menggunakan algoritma, selanjutnya ditambah fitur *Apply Model* untuk menyetujui model algoritma yang

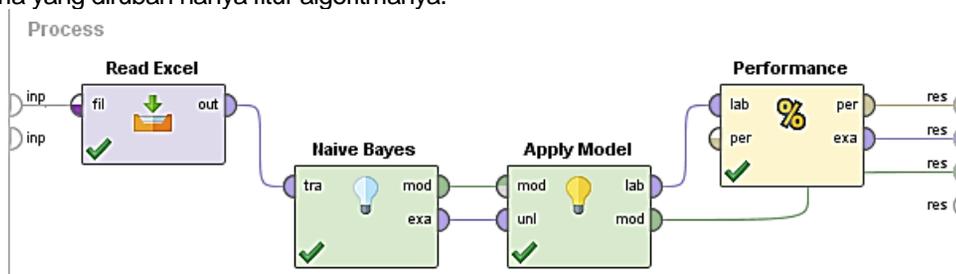
digunakan, dan ditambah fitur *Performance* untuk menampilkan performa dari algoritma yang diterapkan (contoh : akurasi). Hasil pohon keputusan menggunakan *Rapidminer* :



Gambar 2. Pohon Keputusan

2.4.2 K-Naive Bayes

Data profil PMB mahasiswa yang sudah ada dari *file excel* dan dikoneksikan ke *Rapidminer* melalui *read excel*. Susunan struktur antara *Decision Tree (C4.5)* dengan *K-Naive Bayes* hampir sama yang dirubah hanya fitur algoritmanya.



Gambar 3. Struktur Untuk Analisa K-Naive Bayes

Pada Gambar 3. Data dari file excel diimport menggunakan menggunakan fitur *Read Excel*, setelah itu dikoneksikan ke fitur *Naive Bayes* untuk menerapkan data yang diteliti menggunakan algoritma, selanjutnya ditambah fitur *Apply Model* untuk mensetujui model algoritma yang digunakan, dan selanjutnya ditambah fitur *Performance* untuk menampilkan performa dari algoritma yang diterapkan. Hasil Analisa *K-Naive Bayes* menggunakan *Rapidminer* :

SimpleDistribution

Distribution model for label attribute Jurusan Sekarang

Class S1 Inf (0.591)
4 distributions

Class S1 SI (0.409)
4 distributions

Gambar 4. Distribusi atribut jurusan sekarang

2.5 Implementasi Pembuktian SMK di Madiun

2.5.1 Decision Tree (C4.5)

Profil PMB Mahasiswa dari tahun 2014 sampai 2017 setelah diteliti terdapat 1 siswa angkatan tahun 2015 yang berasal dari SMK di Madiun, sehingga pembuktian diterapkan menggunakan data dari sekolah tersebut. Pembuktian diterapkan di SMK di Madiun yang sekarang mempunyai dua keahlian yaitu Multimedia dan Perbankan. Siswa yang di ambil lulusan tahun 2015 sebanyak 27 siswa, terdiri dari 19 jurusan Multimedia dan 8 jurusan perbankan.

Tabel 1. Mencari *entropy* dan *gain*

	JML Kasus			<i>Entropy</i>	<i>Gain</i>
	Ya	Tidak			
TOTAL	27	1	26	0,228538144	
Pekerjaan Orang Tua					0,0446195
Dagang	2	0	2	0	
Swasta	12	1	11	0,41381685	
Serabutan	1	0	1	0	
Wiraswasta	2	0	2	0	
PNS	3	0	3	0	
Buruh Tani	3	0	3	0	
Petani	3	0	3	0	
Ibu Rumah Tangga	1	0	1	0	
Jurusan					0,0192058
Multimedia	19	1	18	0,297472249	
Perbankan	8	0	8	0	

Pada Tabel 1. Menjelaskan hasil perhitungan mencari *entropy* dan *gain*, perhitungan mencari nilai *entropy* pekerjaan orang tua dengan keterangan swasta yaitu :

$$Entropy(S) = \sum_{i=1}^x - p_i \times \log^2 p_i \quad (5)$$

$$= \left(\left(-\frac{1}{12} \right) \times \log_2 \left(\frac{1}{12} \right) + \left(-\frac{11}{12} \right) \times \log_2 \left(\frac{11}{12} \right) \right) = 0,41381685 \quad (6)$$

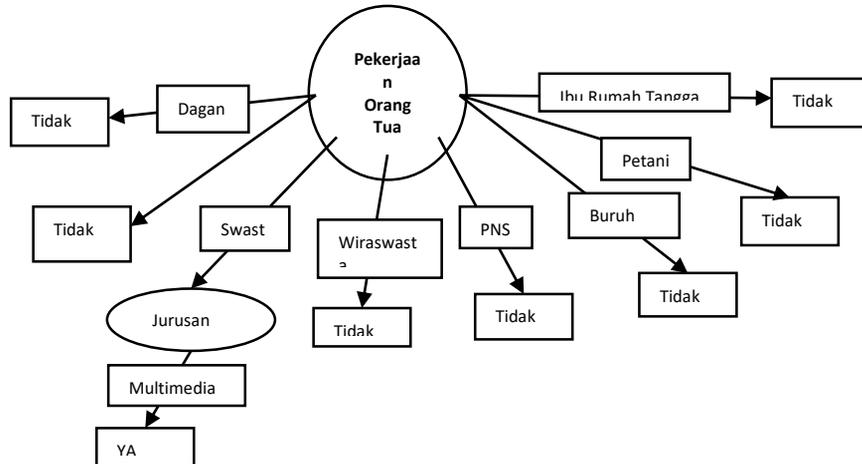
Perhitungan mencari *gain* dari pekerjaan orang tua yaitu :

$$Gain(S, a) = Entropy(S) \sum_{i=1}^x \frac{|S_i|}{|S|} \times Entropy(S_i) \quad (7)$$

$$= 0,044619544 - \left(\left(\frac{5}{27} \times 0 \right) + \left(\frac{12}{27} \times 0,41381685 \right) + \left(\frac{1}{27} \times 0 \right) + \left(\frac{2}{27} \times 0 \right) \right) \quad (8)$$

$$= 0,044619544 - \left(\left(\frac{3}{27} \times 0 \right) + \left(\frac{3}{27} \times 0 \right) + \left(\frac{3}{27} \times 0 \right) + \left(\frac{1}{27} \times 0 \right) \right)$$

Setelah selesai mencari *entropy* dan *gain* maka akan bisa digambar pohon keputusan dari tabel pekerjaan orang tua dengan keterangan pada gambar 5 yaitu :



Gambar 5. Pohon Keputusan *Decision Tree* (C4.5)

Setelah mencari *entropy* dan *gain* bisa ditarik kesimpulan pohon keputusan *Decision Tree* (C4.5) pada Gambar 5. Pada gambar tersebut dengan atribut swasta masih bisa dicari sehingga ketemu jurusan multimedia dengan keterangan “Ya”.

2.5.2 K-Naive Bayes

Pembuktian perhitungan dengan *K-Naive Bayes* ini menggunakan *excel* dengan rumus formula yang sudah tersedia dari Tabel 2. Perhitungan *K-Naive Bayes* berbeda dengan *Decision Tree* (C4.5), algoritma ini menghitung semua data bersama tanpa mencari nilai tertinggi terlebih dahulu.

Tabel 2. Mencari Probabiliti Tiap Atribut

YA/TIDAK	0,037037	0,962962963
----------	----------	-------------

P(V=↓ C=Ya)	... C=Tidak)
Dagang	0	0,074074074
Swasta	0,037037	0,407407407
Serabutan	0	0,037037037
Wiraswasta	0	0,074074074
PNS	0	0,111111111
Buruh Tani	0	0,111111111
Petani	0	0,111111111
Ibu Rumah Tangga	0	0,037037037

P(E=↓ C=Ya)	... C=Tidak)
Multimedia	0,037037	0,818181818
Perbankan	0	0,363636364

Perhitungan mencari probabiliti pada atribut swasta dengan keterangan "Ya" :

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} = \frac{1}{27} = 0,037037 \quad (9)$$

Perhitungan mencari peluang :

$$P(X_i = x_i | Y = y_j) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}} \quad (10)$$

$$= 0,962962963 + 0,407407407 + 0,818181818 = 0,320987654$$

Karena dengan keterangan nama Alil Fachrul dengan nilai TIDAK lebih tinggi dari nilai YA maka dengan keterangan YA berubah menjadi TIDAK. Dengan keterangan sebagai berikut :

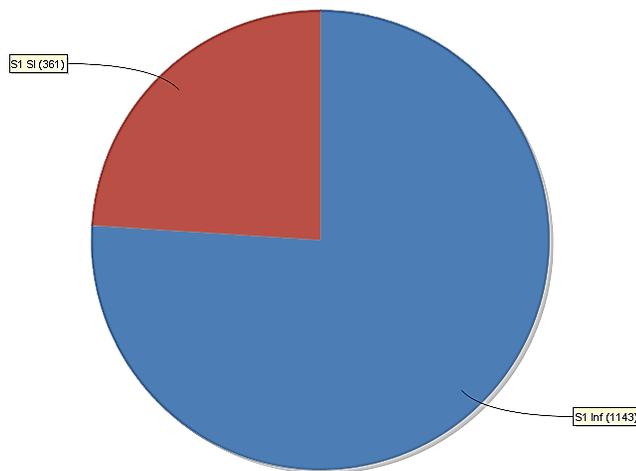
Tabel 3. Peluang K-Naive Bayes

Sebelum	Class (sesudah)	p(X Ya)	p(X Tidak)
Tidak	Tidak	0	0,14266118
Tidak	Tidak	0	0,038907594
Tidak	Tidak	0	0,14266118
Tidak	Tidak	0	0,038907594
Tidak	Tidak	0	0,038907594
Tidak	Tidak	0	0,012969198
Tidak	Tidak	0	0,14266118
Tidak	Tidak	0	0,038907594
Tidak	Tidak	0	0,058361392
Tidak	Tidak	0	0,029180696
Tidak	Tidak	0	0,320987654
Tidak	Tidak	0	0,058361392
Tidak	Tidak	0	0,087542088
Tidak	Tidak	0	0,058361392
Tidak	Tidak	0	0,320987654
Tidak	Tidak	0	0,087542088
Tidak	Tidak	0	0,087542088
Tidak	Tidak	0	0,320987654
Tidak	Tidak	0	0,058361392
Tidak	Tidak	0	0,087542088
Tidak	Tidak	0	0,320987654
Tidak	Tidak	0	0,320987654
Tidak	Tidak	0	0,320987654
Tidak	Tidak	0	0,087542088
Tidak	Tidak	0	0,320987654
Tidak	Tidak	0	0,320987654
Ya	Tidak	0	0,320987654

Pada Tabel 2, kolom **Sebelum** adalah data sebelum diteliti dan pada kolom **Class (sesudah)** adalah data setelah diteliti. Jika nilai pada kolom $p(X | Ya)$ lebih kecil dari $p(X | Tidak)$ maka kolom *Class* (sesudah) menjadi keterangan “Tidak” dan sebaliknya. Perubahan keterangan dari “Ya” menjadi “Tidak” bisa dilihat pada Tabel 2 baris yang terakhir karena kolom $p(X | Ya)$ lebih besar dari $p(X | Tidak)$ yaitu 0 lebih besar dari 0,320987654.

3. HASIL DAN PEMBAHASAN

Analisa data tingkat Universitas di Yogyakarta dan SMK di Madiun, pada data tingkat Universitas di Yogyakarta mempunyai jurusan S1 Teknik Informatika dengan jumlah 1143 sedangkan S1 Sistem Informasi berjumlah 361 semuanya berjumlah 1504 data.



Gambar 5. Grafik Profil PMB mahasiswa tingkat Universitas di Yogyakarta

(1)
Pada gambar 5 menunjukkan data histori PMB Mahasiswa baru yang mendominasi adalah dari jurusan S1 Inf. Dan pada jurusan S1 Inf yang mendominasi jurusan asal adalah IPA, sehingga pada PMB yang akan datang bisa difokuskan promosi ke SMA jurusan IPA.

Tabel 4. Perbandingan Algoritma *Decision Tree (C4.5)* dan Algoritma *K-Naive Bayes*

	Tingkat Universitas	Tingkat Universitas (Sampel)	SMK di Madiun
<i>Decision Tree (C4.5)</i>			
<i>Data Record</i>	988	20	27
Akurasi	81,84%	100%	96,30%
Akurasi <i>Error</i>	18,16%	0%	3.70%
Waktu	3 detik	0 detik	0 detik
<i>K-Naive Bayes</i>			
<i>Data Record</i>	1504	20	27
Akurasi	85,12%	100%	96.30%
Akurasi <i>Error</i>	14,88%	0%	3.70%
Waktu	0 detik	0 detik	0 detik

Penelitian ini menggunakan *Rapidminer* dengan hasil pada Tabel 4. Menunjukkan tingkat akurasi *Rapidminer* dengan data yang lebih besar yaitu 1504 data menggunakan Algoritma *Decision Tree (C4.5)* mempunyai tingkat akurasi yang lebih kecil dibanding dengan algoritma *K-Naive Bayes* waktu yang dibutuhkan, *K-Naive Bayes* lebih cepat dibanding dengan *Decision Tree(C4.5)*. Sedangkan menggunakan data yang lebih kecil yaitu 20 dan 27 data, kedua algoritma mempunyai tingkat akurasi sama.

4. KESIMPULAN

Kesimpulan dari penelitian ini adalah dengan membandingkan dua algoritma, *K-Naive Bayes* yang mempunyai tingkat akurasi yang lebih baik dibanding dengan algoritma *Decision Tree (C4.5)* mempunyai akurasi 81,84% dan akurasi *error* 18,16%, sedangkan *K-Naive Bayes* 85,12% dan akurasi *error* 14,88%. Sedangkan dengan data yang lebih kecil *Decision Tree (C4.5)* mempunyai akurasi 100% sedangkan kan *K-Naive Bayes* mempunyai akurasi yang sama dengan *Decision Tree (C4.5)* yaitu 100%. Dari segi kecepatan *Decision Tree (C4.5)* lebih lambat menganalisa dibanding dengan *K-Naive Bayes*. Karena perbandingan hanya 3 detik dengan menggunakan software *Rapidminer* sehingga waktu kurang signifikan pada penelitian ini.

UCAPAN TERIMA KASIH

Terimakasih kepada instansi terkait yang telah membantu menyediakan data, kepada teman-teman dan dosen pembimbing yang telah membimbing penelitian ini sehingga bisa selesai dengan baik walaupun penelitian ini masih kurang sempurna.

DAFTAR PUSTAKA

- [1] Angra, S. and Ahuja, S. (2016) 'Analysis of student's data using rapid miner', Journal on Today's Ideas - Tomorrow's Technologies, 4(1), pp. 49–58. doi: 10.15415/jotitt.2016.41004.
- [2] Brauning, M. et al. (2017) 'Lexicographic preferences for predictive modeling of human decision making: A new machine learning method with an application in accounting', European Journal of Operational Research, 258(1), pp. 295–306. doi: 10.1016/j.ejor.2016.08.055.
- [3] Chen, L. and Wang, S. (2012) 'Automated Feature Weighting In Naive Bayes For High-dimensional Data Classification', Proceedings of the ACM International Conference on Information and Knowledge Management, pp. 1243–1252. doi: 10.1145/2396761.2398426.
- [4] Jannach, D., Jugovac, M. and Lerche, L. (2016) 'Supporting the Design of Machine Learning Workflows with a Recommendation System', ACM Transactions on Interactive Intelligent Systems, 6(1), pp. 1–35. doi: 10.1145/2852082.
- [5] Kitcharoen, N. et al. (2013) 'RapidMiner Framework for Manufacturing Data Analysis on the Cloud', 2013 10th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 1–6. doi: 10.1109/JCSSE.2013.6567336.
- [6] S. Rana and A. Singh (2016) 'Comparative Analysis of Sentiment Orientation Using SVM and Naive Bayes Techniques', (October), pp. 106–111.