BAB II
LANDASAN TEORI

2.1 Tinjauan Studi

Dalam penulisan skripsi ini penulis menggali informasi dari penelitian sebelumnya sebagai bahan referensi, baik mengenal permasalahan, metode, dan hasil yang fungsinya untuk memudahkan peneliti dalam melakukan penelitianya sesuai dengan tema. Beberapa penelitian terkait sebagai berikut :

Dalam suatu penelitian [3] membahas masa studi mahasiswa menggunakan metode klasifikasi algoritma Naive Bayes. Penelitian tersebut menggunakan data alumni Fakultas Teknik tahun 2008-2012 Universitas Sam Ratulangi Manado dengan menggunakan 9 (sembilan) atribut dengan rincian 1 (satu) atribut ID yaitu nama, 7 (tujuh) atribut reguler yaitu prodi, semester, IP semester, IPK, SKS, lulus, jumlah semester, dan 1 (satu) atribut kelas yaitu status kelulusan. Hasil penelitian tersebut menunjukkan bahwa algoritma Naive Bayes dapat menentukan prediksi masa studi mahasiswa dengan tingkat akurasi sebesar 85,17%.

Penelitian selanjutnya [5] membahas perbandingan antara algoritma C4.5 dan Naive Bayes dalam memprediksi penyakit hepatitis. Penelitian tersebut menggunakan data sekunder yang diperoleh dari *Machine Learning Repository* UCI dengan menggunakan 12 (dua belas) atribut dengan rincian 1 (satu) atribut ID yaitu nomor, 10 (sepuluh) atribut reguler yaitu *age, steroid, malaise, liver_big, spiders, varices, bilirubin, sgot, albumin, protime*, dan 1 (satu) atribut kelas yaitu keterangan. Hasil penelitian ini menunjukkan bahwa algoritma Naive Bayes memiliki nilai akurasi yang lebih tinggi untuk prediksi penyakit hepatitis yaitu 83,71%, sedangkan nilai akurasi algoritma C4.5 adalah 77,29%.

Penelitian selanjutnya [6] membahas perbandingan dari algoritma C.45, K-Nearest Neighbor, dan Naive Bayes untuk mengklasifikasi penanggung jawab BSI entrepreneur center. Penelitian tersebut menggunakan data primer yang diperoleh dari bagian administrasi BSI entrepreneur center Universitas Bina Sarana Informatika dengan menggunakan 12 (dua belas) atribut. Dari hasil pengujian masing-masing algoritma menunjukan bahwa algoritma Naive Bayes merupakan

algoritma yang paling tepat untuk klasifikasi penanggung jawab BSI entrepreneur center karena memiliki nilai akurasi yang paling tinggi yaitu 80%, sedangkan nilai akurasi algoritma C.45 adalah 73,33% dan algoritma K-Nearest Neighbor memiliki nilai akurasi 70%.

Penelitian selanjutnya [4] membahas penerapan algoritma C4.5 untuk memprediksi heregistrasi calon mahasiswa baru. Penelitian tersebut menggunakan data calon mahasiswa baru Politeknik Negeri Bengkalis tahun 2016-2017 dengan menggunakan 10 (sepuluh) atribut dengan rincian 1 (satu) atribut ID yaitu nama, 8 (delapan) atribut reguler yaitu jenis kelamin, kota asal, jurusan sekolah, pilihan kampus 1, pilihan kampus 2, pilihan prodi 1, pilihan prodi 2, prodi diterima, dan 1 (satu) atribut kelas yaitu status heregistrasi. Hasil penelitian tersebut menunjukkan bahwa algoritma C4.5 dapat menentukan prediksi heregistrasi calon mahasiswa baru dengan tingkat akurasi sebesar 68,93%.

Berdasarkan penjelasan diatas tentang perbedaan dari beberapa penelitian yang telah dilaksanakan sebelumnya, maka perbedaan penelitian yang akan dilaksanakan peneliti adalah penerapan Algoritma Bayes dengan permasalahan dan data dengan atribut yang berbeda. Peneliti akan menerapkan algoritma Naive Bayes untuk melakukan prediksi heregistrasi calon mahasiswa baru di UNISNU Jepara. Dataset yang akan digunakan adalah data calon mahasiswa baru UNISNU Jepara tahun 2019-2020 dengan 14 (empat belas) atribut dengan rincian 1 (satu) atribut ID yaitu nama, 12 (dua belas) atribut reguler yaitu tahun pendaftaran, program kelas, jenis kelamin, usia, prodi, kota asal, pekerjaan ayah, pekerjaan ibu, penghasilan orangtua, jurusan sekolah asal, nilai UN, informasi pendaftaran, dan 1 (satu) atribut kelas yaitu status heregistrasi. Algoritma Naive Bayes diharapkan mampu memprediksi heregistrasi calon mahasiswa baru dengan lebih akurat sehingga dapat membantu pihak pengelola penerimaan mahasiswa baru UNISNU Jepara untuk mengetahui calon mahasiswa baru cenderung berpotensi melakukan heregistrasi atau tidak serta membantu dalam pengambilan keputusan dan menentukan langkah di penerimaan mahasiswa baru yang akan datang.

2.2 Tinjauan Pustaka

2.2.1 Data Mining

Salah satu bidang ilmu komputer yang sedang berkembang pesat saat ini adalah data mining. Hal ini disebabkan karena data mining menghasilkan suatu output berupa model yang bisa memperkirakan kelompok dari suatu objek. Model bisa berbentuk aturan "jika-maka" yang dihasilkan dari proses latihan atau biasa disebut *Model Training* dengan data yang telah ada sebelumnya.

Data mining atau biasa disebut *Knowledge Discovery in Database* (KDD) merupakan sebuat kegiatan yang berhubungan dengan pengumpulan data atau penggunaan data historis dengan tujuan menemukan informasi, pengetahuan, keteraturan, pola atau hubungan data yang berukuran besar. Hasil dari data mining dapat digunakan sebagai alternatif dalam penentuan atau penentuan keputusan pada masa yang akan datang [1].

Data Mining juga digunakan untuk mencari suatu pola dari sekumpulan data yang berjumlah cukup banyak. Sehingga dengan data mining, data diam yang tersimpan di *database* bisa dimanfaatkan untuk diolah lebih lanjut hingga menghasilkan pengetahuan atau informasi yang berharga. Data diam ini bisa dijadikan data latih untuk membuat suatu model yang nantinya bisa menjadi bantuan dalam pengambilan keputusan [7].

Tahap pembentukan model pada data mining terbagi menjadi 2 tahap, yaitu: tahap *training* dan tahap *testing*. Pada tahap *training*, data yang telah diberi label atau telah diketahui kelompoknya digunakan untuk melatih model dengan menggunakan suatu algoritma tertentu. Setelah model terlatih telah terbentuk, maka dilanjutkan dengan tahap uji coba atau *testing* yang berguna untuk mengetahui akurasi dari model tersebut. Model yang memiliki akurasi baik bisa digunakan untuk melakukan prediksi kelompok dari data yang belum diketahui kelompoknya.

Perbedaan antara proses pemrograman konvensional dengan pemrograman *machine learning* melalui data mining adalah jika pada *machine learning*, model dilatih dari data yang sudah ada sebelumnya (*Learning by example*) sehingga menghasilkan suatu aturan baru. Sedangkan pemrograman konvensional,

aturannya diprogram secara manual tanpa melalui proses pelatihan/pembelajaran terlebih dahulu [8].

Berdasarkan definisi-definisi diatas tentang Data Mining dapat disimpulkan bahwa Data Mining adalah sebuah proses pencarian secara otomatis untuk menemukan pola atau model dari suatu *database* yang besar.

2.2.1.1 Metode Pelatihan Model

Terdapat 2 macam pendekatan yang dilakukan dalam proses pelatihan model pada data mining, yaitu [8]:

1. *Supervised Learning*

   Pelatihan dengan pendekatan *supervised learning*, prosesnya memerlukan sejumlah data yang memiliki input yang berupa features dan output yang biasa disebut label. Dengan menggunakan suatu algoritma tertentu, mesin akan belajar melalui observasi data yang telah disediakan. Sehingga nantinya mesin dapat memetakan dari *features* menjadi sebuah *output* (label) dari data baru atau data yang belum pernah dipelajari oleh mesin sebelumnya. Secara garis besar *supervised learning* adalah pelatihan model dengan data yang telah ditentukan kelas atau kelompoknya sebelumnya.

2. *Unsupervised Learning*

   Sedangkan pada pelatihan model dengan pendekatan unsupervised learning mampu mencari pola dari kumpulan data yang diberikan secara otomatis. *Unsupervised learning* mengelompokkan data berdasarkan suatu pola tertentu yang belum ditentukan sebelumnya. Perbedaan antara *supervised learning* dan *unsupervised learning* adalah data yang digunakan untuk *training* tidak perlu diberi label, mesin yang akan mencari polanya sendiri.

2.2.1.2 Teknik Data Mining

Menurut [9] data mining memiliki beberapa teknik dan sifat yang biasa dilakukan. Teknik dan sifat tersebut diantaranya sebagai berikut:

1. Klastering

   Klastering merupakan sebuah metode dalam pengelompokkan obyek ke dalam beberapa kelompok berdasarkan kemiripan antar obyek. Dalam satu klaster, harus didapati obyek yang mirip dan obyek salin tidak mirip. Berbeda dengan metode klasifikasi, dalam pengolahannya klastering tidak memerlukan data latih yang sudah diberi label.

2. Klasifikasi

   Klasifikasi merupakan sebuah metode untuk mengelompokkan obyek berdasarkan kelompok yang sudah tersedia. Seperti yang dijelaskan sebelumnya, jika klastering tidak memerlukan data latih yang sudah diberi label, maka pada klasifikasi data latih harus sudah diberi label. Proses prediksi dilakukan dengan membangun model melalui proses pelatihan dengan menggunakan *data training*.

3. Regresi/Estimasi

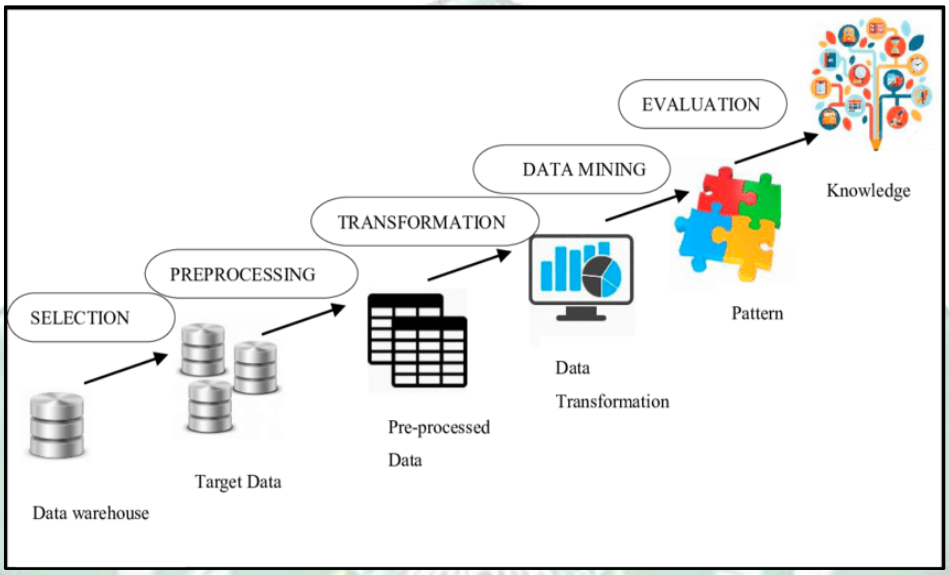
   Regresi merupakan sebuah metode untuk mengetahui atau mencari model hubungan antara atribut predictor dan atribut dependen yang berupa nilai kontinyu. Berbeda dengan klasifikasi, output yang dihasilkan pada metode regresi bersifat nilai kontinyu. Sedangkan pada klasifikasi merupakan nilai bersifat diskrit.

4. Asosiasi

   Asosiasi merupakan sebuah metode yang melakukan asosiasi antar obyek dalam suatu set data. Data yang digunakan biasanya merupakan data transaksional. Asosiasi dilakukan dengan menghitung berapa kali set data suatu transaksi mengandung dua item atau lebih yang memiliki hubungan. Metode ini sering disebut *Market Basket Analysis*.

2.2.2 *Knowledge Discovery in Database*

Knowlegde Discovery in Databases (KDD) kerap kali digunakan secara bergantian untuk menguraikan proses pencarian informasi dalam suatu data yang berukuran besar. Data mining dan *Knowlegde Discovery in Databases* (KDD) memiliki konsep yang berbeda, namun memiliki keterkaitan satu sama lain [10]. *Knowlegde Discovery in Databases* (KDD) memiliki proses yang dapat dijelaskan sebagai berikut:



Gambar 2.1 Tahapan Dalam KDD

1. *Data Selection*

   Tahap awal dalam menjalankan kegiatan adalah dengan menyeleksi data dari sekumpulan data yang berskala besar, sebelum nantinya akan diproses oleh data mining. Data tersebut nanti akan disimpan pada suatu berkas yang terpisah dari basis data operasional yang nantinya akan digunakan untuk proses data mining.

2. *Pre-processing/Cleaning*

   Setelah proses pemilihan data, selanjutnya data akan melewati proses cleaning. Proses cleaning dilakukan untuk membuang data yang tidak lengkap atau untuk menghindari duplikasi data (ganda), data yang tidak valid, dan memperbaiki kesalahan-kesalahan data seperti salah cetak. Selain itu dilakukan proses enrichment yang memiliki tujuan untuk

memperkaya data yang sudah tersedia dengan data atau informasi yang relevan.

3. *Transformation*

Tahap ketiga dalam proses data mining adalah transformasi dalam pemilihan data. Peneliti harus mengenali data mining yang akan diterapkan dalam pengolahan data. Seperti pada algoritma *Iterative Dichotomiser Three* (ID3), tipe data yang digunakan adalah string. Jika data awal yang diperoleh peneliti bersifat *numeric*, maka perlu dilakukan pengolahan untuk mengubah tipe data tersebut.

4. *Data Mining*

Data mining adalah proses pencarian data atau informasi dari data yang terpilih menggunakan teknik atau metode tertentu. Pemilihan metode atau algoritma dapat menentukan hasil perhitungan dalam data mining. Hal tersebut karena karakteristik dari tiap metode yang berbeda.

5. *Interpretation*

Tahap ini memberikan pola informasi yang dihasilkan dari proses sebelumnya sehingga pihak berkepentingan bisa dengan mudah memahami. Hal tersebut dikarenakan mencakup pemeriksaan pola atau informasi yang ditemukan bersifat fakta atau hipotesis yang ada sebelumnya.

2.2.3 Jenis-Jenis Variabel

Pada sebuah data kuantitatif yang diolah terdapat berbagai macam jenis variabel. Sebelum mengolahnya melalui pendekatan data mining dengan menggunakan suatu metode tertentu, baiknya telah dikenali jenis-jenis variabel yang ada di dalamnya. Hal ini bertujuan agar memudahkan dalam proses pemilihan algoritma yang tepat. Adapun beberapa jenis-jenis variabel yaitu [11] :

1. Variabel Diskrit

Merupakan variabel yang berupa data yang sifatnya memiliki kategori atau dapat dikelompokkan pada suatu kelas tertentu. Kategori atau kelas yang terbentuk memiliki nilai yang setara. Contohnya atribut yang

memiliki jenis variabel diskrit adalah jenis kelamin. Data jenis kelamin memiliki 2 kategori yaitu pria dan wanita. Dua kategori tersebut memiliki nilai yang setara secara data.

2. Variabel Kontinyu

   Variabel kontinyu adalah data yang berbentuk angka, bisa berbentuk bilangan pecahan maupun bilangan bulat. Sehingga variabel kontinyu dapat digunakan untuk proses operasi hitung. Contoh atribut yang berjenis variabel kontinyu adalah umur. Karena umur bisa digunakan untuk proses operasi hitung seperti dihitung rata-ratanya.

3. Variabel Ordinal

   Variabel Ordinal merupakan data yang berbentuk angka yang memiliki peringkat. Dengan adanya peringkat, memungkinkan variabel ordinal tidak setara satu sama lain. Contoh variabel ordinal adalah pada atribut ranking kelas. Data rangking kelas 1 tidak sama dengan data rangking kelas 2, kedua angka tersebut memiliki peringkat yang tidak setara.

4. Variabel Interval

   Data angka juga bisa dikatakan sebagai variabel interval apabila datanya memiliki rentang dengan jarak yang jelas. Contoh atribut yang memliki jenis variabel interval adalah rentang gaji ataupun rentang usia. Dimisalkan terdapat data rentang gaji dengan nilai 1 juta-3 juta, 5 juta-7 juta, dan lain sebagainya. Data- data tersebut memiliki rentang dan jarak yang jelas yaitu 2 juta.

5. Variabel Rasio

   Variabel rasio merupakan data angka yang telah diproses melalui operasi hitung yang kompleks. Angka pada variabel rasio bukanlah sebuah simbol ataupun kategori melainkan merupakan angka yang sebenarnya. Contoh variabel rasio adalah data tinggi badan. Dimisalkan data tinggi badan A bernilai 100 cm dan data tinggi badan B bernilai 200 cm, maka skala rasio tinggi badan A adalah setengah dari tinggi badan B.

2.2.4 Klasifikasi

Klasifikasi adalah salah satu teknik yang ada pada data mining yang mampu mengelompokkan data berdasarkan pelatihan yang telah dilakukan sebelumnya menggunakan data yang telah tersedia hingga terbentuk aturan baru pada model [12]. Proses *training* yang dilakukan tentunya menggunakan data yang berlabel karena jika tidak, maka disebut *clustering*.

Proses untuk menemukan pola yang menjelaskan data yang penting dikenal sebagai klasifikasi [2]. Metode klasifikasi dalam data mining ada banyak, diantaranya decision tree, k-nearest neighbor, neural network dan naïve bayes :

1. Decision Tree

   Decision Tree merupakan metode klasifikasi dalam bentuk diagram yang direpresentasikan seperti struktur pohon.

2. K-Nearest Neighbor

   Metode K-Nearest Neighbor merupakan metode klasifikasi pertama yang dijabarkan pada awal tahun 1950.

3. Neural Network

   Neural Network terinspirasi pengenalan sistem pembelajaran kompleks otak binatang yang terdiri atas kumpulan neuron yang saling berhubungan.

4. Naïve Bayes

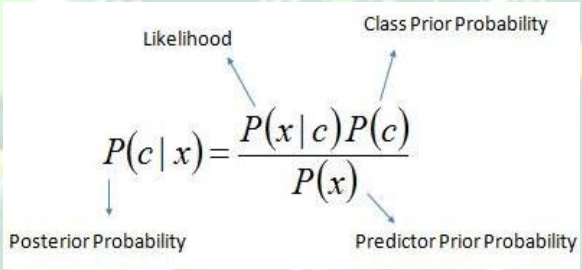
   Klasifikasi naïve bayes adalah salah satu teknik data mining yang paling popular untuk mengklasifikasikan data dalam jumlah besar dan dapat digunakan untuk memprediksi probabilitas keanggotaan suatu class.

2.2.5 Naive Bayes

Naïve Bayes adalah salah satu algoritma pembelajaran induktif yang paling efektif dan efisien untuk *machine learning* dan data mining. Performa naïve bayes yang kompetitif dalam proses klasifikasi walaupun menggunakan asumsi keidependenan atribut (tidak ada kaitan antar atribut). Asumsi keidependenan atribut ini pada data sebenarnya jarang terjadi, namun walaupun asumsi keidependenan atribut tersebut dilanggar performa pengklasifikasian naïve bayes cukup tinggi, hal ini dibuktikan pada berbagai penelitian empiris.

Definisi lain mengatakan Naïve Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai teorema Bayes. Teorema tersebut dikombinasikan dengan ”naive” dimana diasumsikan kondisi antar atribut saling bebas [13].

Naive Bayes didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara kondisional saling bebas jika diberikan nilai *output*. Dengan kata lain, diberikan nilai *output*, probabilitas mengamati secara bersama adalah produk dari probabilitas individu. Keuntungan penggunaan *Naive Bayes* adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (*Training Data*) yang kecil untuk menentukan estimasi paremeter yang diperlukan dalam proses pengklasifikasian. Naive Bayes sering bekerja jauh lebih baik dalam kebanyakan situasi dunia nyata yang kompleks dari pada yang diharapkan [14].



Naive Bayes merupakan pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu *class*. Naive Bayes memiliki akurasi dan kecepatan yang sangat tinggi saat diaplikasi ke dalam *database* dengan data yang besar. Berikut merupakan keterangan dari teorema *bayes* [15]:

Keterangan :

x : data dengan *class* yang belum diketahui

c : hipotesis data x merupakan suatu *class* spesifik

P(c|x) : probabilitas hipotesis c berdasar kondisi x (*posteriori probability*)

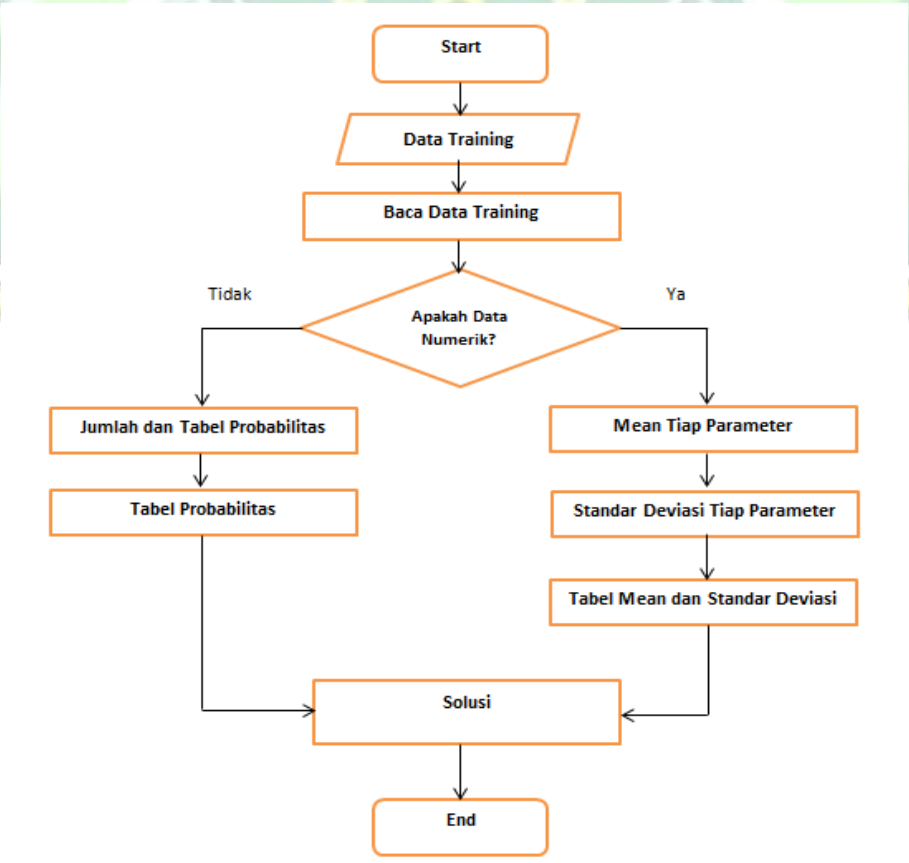
P(c) : probabilitas hipotesis c (*prior probability*)

P(x|c) : probabilitas x berdasarkan kondisi pada hipotesis c

P(x) : probabilitas dari x.

Alur dari metode Naïve Bayes sebagai berikut [16]:

a. Baca *data training*
b. Hitung jumlah data probabilitas, namun apabila data numerik maka:
c. Cari nilai *mean* dan standar deviasi dari masing-masing parameter yang merupakan data numerik
d. Cari nilai probabilitas dengan cara menghitung jumlah data yang sesuai dari dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut.
e. Mendapatkan nilai dalam tabel mean, standar deviasi dan probabilitas.



Gambar 2.2. Alur Naïve Bayes.

Untuk mengetahui lebih lanjut mengenai algoritma Naive Bayes, berikut adalah contoh perhitungannya [3]:

Sampel Data yang akan dikelola dengan perhitungan manual :

Tabel 2.1 Sampel Data

Id	Tahun	Prodi	Lulus	ips_1	ips_2	.....	sks_4	sks_5	sks_6
12	2009	T. Mesin	Cepat	0	2.13	.....	30	0	\N
13	2009	Arsitek	Cepat	4	0	.....	8	\N	\N
14	2009	T. Sipil	Cepat	0	2.97	.....	0	\N	\N
…						.....			
15	2010	T. Sipil	Cepat	4	2.79	.....	10	34	24
29	2010	Informatika	Tidak Lulus	2.56	2.85	.....	18	17	17
30	2010	Arsitektur	Tidak Lulus	1.8	0.5	.....	17	16	15
31	2010	T. Sipil	Tidak Lulus	0.73	2.38	.....	15	20	17

1. Menghitung Standar Deviasi dan Mean

Nilai Standar Deviasi dan Mean diambil dari masing – masing variable yang bernilai kontinu antara lain IPS_1, IPS_2, IPS_3, IPS_4, IPS_5, IPS_6, SKS_1, SKS_2, SKS_3, SKS_4, SKS_5, dan SKS_6 pada setiap kategori. Berikut proses perhitungan yang menunjukan nilai Mean dan Standar Deviasi.

Tabel 2.2. Perhitungan Mean dan Standar Deviasi

kelas	*Mean*/sttde	ips_1	ips_2	.	ipk_4	ipk_5	ipk_6
cepat	*Mean*	2.186	2.122	.	15.6	6.8	4.8
	Standar Deviasi	2.042787	1.227383	.	13.66748	15.20526	10.73313

tepat	*Mean*	3.39	3.032	.	15.8	23.8	22.2
	Standar Deviasi	0.601457	0.693376	.	9.038805	2.48998	4.969909
terlambat	*Mean*	1.51	2.198	.	15.4	19.8	21
	Standar Deviasi	1.46506	1.691544	.	8.619745	0.447214	4.301163
tidak lulus	*Mean*	1.246	0.908	.	14.2	15.2	16
	Standar Deviasi	1.154807	1.243411	.	8.043631	8.58487	9.192388

Dengan aturan tersebut jika diberikan data baru yang terdapat pada tabel di bawah ini maka prediksi masa studi mahasiswa dapat dikategorikan dengan menggunakan langkah-langkah sebagai berikut :

Tabel 2.3 Data hitung manual

NIM	Prodi	IPS1	IPS2	..	SKS3	SKS4	SKS5
12345	Sipil	2.52	2.98	..	19	9	14

2. Menghitung Probabilitas untuk setiap kelas berdasarkan fitur

Dari kasus diatas maka akan dilakukan perhitungan nilai probabilitas dari variabel yang bersifat kontinu yaitu IPS1 s/d IPS6. Berikut proses perhitungan masing-masing IPS :

a. Probabilitas IPS dan SKS untuk kelas Tepat dari IPS1 s/d IPS6

1. Probabilitas IPS1 untuk kelas Tepat

Standar Deviasi $\sigma = 0.601456565347825$

Mean $\mu = 3.39$

$f(ips1 = 2.52 \mid Tepat) =$

$$\frac{1}{\sqrt{2\pi}(0.601456565347825)} e^{\frac{(2.52-3.39)^2}{2(0.601456565347825)^2}}$$

$= 0.216730648 \times 2{,}7183^{-0.20084401}$

$= 0.177294107$

2. Probabilitas IPS2 untuk kelas Tepat

Standar Deviasi σ = 0.693375799981511

Mean μ = 3.032

f(ips2 = 2.98 | Tepat) =

$$\frac{1}{\sqrt{2\pi(0.693375799981511)}} e^{\frac{(2.98-3.032)^2}{2(0.693375799981511)^2}}$$

= 0.22917 x $2{,}7183^{-0.5947}$

= 0.12644

b. Probabilitas IPS dan SKS untuk kelas Terlambat dari IPS1 s/d IPS6

1. Probabilitas IPS1 untuk kelas Terlambat

Standar Deviasi σ = 1.46506

Mean μ = 1.51

f(ips1 = 2.52 | Tepat) =

$$\frac{1}{\sqrt{2\pi(1.46506)}} e^{\frac{(2.52-1.51)^2}{2(1.46506)^2}}$$

= 0.329679933 x $2{,}7183^{-0.009160196}$

= 0.177294107

3. Membandingkan hasil perkalian ketiap kelas

- *Likelihood* Cepat :
  P(Sipil|Cepat)*P(IPS1|Cepat)*P(IPS2|C epat)*P(IPS3|Cepat)*P(IPS 4|Cepat)*P(IPS5|Cepat)*P(SKS1|Cepat)*P(SKS2|Cepat)*P(SKS3|Cepat) *P(SK S4|Cepat)*P(SKS5|Cepat)
  = 0.4 * 0.27549 * 0.28211 * 0.22815 * 0.30901 * 0.31946 * 0.08218 * 0.06121 * 0.06275 * 0.09606 * 0.09148 = 0.954276465
- *Likelihood* Tepat = 2.81285E-05
- *Likelihood* Terlambat = 0
- *Likelihood* Tidak Lulus = 0.045695407

Probabilitas of Cepat = 82 %

Probabilitas of Tepat = 13 %

Probabilitas of Terlambat = 0 %

Probabilitas of Tidak Lulus = 5 %

Dari contoh kasus di atas bisa diprediksi bahwa mahasiswa dengan Program Studi Teknik Sipil, dengan data IP Semester 1 s/d 5 memiliki Prediksi = Cepat.

2.2.6 *Information Gain*

Information Gain merupakan metode seleksi atribut paling sederhana dengan melakukan perangkingan atribut dan banyak digunakan dalam aplikasi kategorisasi teks, analisis data *microarray* dan analisis data citra. *Information Gain* dapat membantu mengurangi *noise* yang disebabkan oleh fitur-fitur yang tidak relevan. *Information Gain* mendeteksi fitur-fitur yang paling banyak memiliki informasi/nilai berdasarkan kelas tertentu [17]. *Information gain* dapat digunakan untuk mengetahui pengaruh atribut dataset terhadap klasifikasi. Nilai *information gain* diperoleh dari nilai *entropy* sebelum pemisahan dikurangi dengan nilai *entropy* setelah pemisahan. Pengukuran nilai ini digunakan sebagai tahap awal untuk penentuan atribut yang nantinya akan digunakan atau dibuang. Atribut yang memenuhi kriteria/memiliki nilai tinggi dalam pembobotan nantinya akan digunakan dalam proses klasifikasi sebuah algoritma [18].

2.2.7 *Confusion Matrix*

Confusion Matrix adalah tabel matriks yang dapat digunakan untuk mengevaluasi model klasifikasi yang telah dilatih atau dibuat [19]. Terdapat 2 kelas pada tabel *confusion matrix* yaitu kelas positif dan negatif. Pada tabel *confusion matrix* juga terdapat 4 *cell* yang memiliki label masing-masing: *True Positive* (TP), *False Positive* (FP), *False Negative* (FN) dan *True Negative* (TN) [20].

Tabel 2.4 *Confussion matrix*

		Kelas predisksi	
		1	**0**
Kelas sebenarnya	**1**	TP	FN
	0	FP	TN

Label-label pada tabel matriks memiliki maksud untuk membedakan hasil prediksi dari model. Label *True Positive* (TP) digunakan untuk mempresentasikan jumlah data berkelas positif yang diklasifikasikan sebagai kelas positif juga. Sedangkan label *False Positive* (FP) adalah jumlah data berkelas positif yang diklasifikasikan sebagai kelas negatif. Adapun label *False Negative* (FN) yang menampilkan jumlah data berkelas negatif yang diklasifikasikan sebagai kelas positif. Yang terakhir adalah label *True Negative* (TN) adalah data berkelas negatif yang diklasifikasikan sebagai kelas negatif juga.

Akurasi dihitung berdasarkan tingkat kedekatan antara nilai prediksi dengan nilai aktual untuk metode pengujian akurasi. Dengan mengetahui secara benar jumlah data klasifikasi maka akurasi hasil prediksi dapat diketahui. Nilai akurasi dapat dicari dengan menggunakan persamaan berikut :

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 100\%$$

Proporsi jumlah kasus yang diprediksi positif yang juga positif benar pada data yang sebenarnya disebut nilai presisi. Proporsi jumlah kasus positif yang sebenarnya yang diprediksi positif secara benar pula disebut nilai recall. Sedangkan presentase jumlah data yang diklasifikasi secara benar secara keseluruhan disebut nilai *accuracy* [21]. Nilai presisi bisa diperoleh dengan persamaan berikut :

$$Presisi = \frac{TP}{(FP + TP)} \times 100\%$$

Dan nilai *recall* bisa diperoleh dengan menggunakan persamaan berikut :

$$Recall = \frac{TP}{(FN + TP)} \times 100\%$$

2.2.8 Kurva ROC (*Receiver Operating Characteristic*)

Kurva ROC *(Receiver Operating Characteristic)* merupakan kurva yang dibagi dalam dua dimensi yang digunakan untuk mengetahui tingkat keberhasilan dari setiap metode klasifikasi. Kurva *ROC (Receiver Operating Characteristic)* digunakan untuk mengekspresikan data *confusion matrix*. Garis horizontal mewakili nilai *false positive* (FP) dan garis vertikal mewakili nilai *true positive* (TP). Akurasi AUC (*Area Under Curve*) dari kurva ROC (*Receiver Operating Characteristic*) dikatakan sempurna apabila nilai AUC (*Area Under Curve*) mencapai 1.000 dan akurasinya buruk jika nilai AUC (*Area Under Curve*) dibawah 0.500. Untuk klasifikasi data mining, nilai AUC (*Area Under Curve*) dapat dibagi menjadi beberapa kelompok [22]:

a. 0,90 – 1,00 = Klasifikasi Sangat Baik (*Excellent Classification*)
b. 0,80 – 0,90 = Klasifikasi Baik (*Good Classification*)
c. 0,70 – 0,80 = Klasifikasi Cukup (*Fair Classification*)
d. 0,60 – 0,70 = Klasifikasi Buruk (*Poor Classification*)
e. 0,50 – 0,60 = Klasifikasi Gagal (*Failure*)

2.2.9 RapidMiner

RapidMiner merupakan sebuah perangkat lunak yang bersifat open source. RapidMiner adalah sebuah upaya penyelesaian dalam melakukan tahapan analisis terhadap beberapa keilmuan seperti data mining, teks mining, dan memprediksi. Dalam memberikan pemahaman pada pengguna, RapidMiner memanfaatkan beberapa teknik seperti prediksi dan deskriptif yang nantinya mampu menghasilkan keputusan paling baik. Jumlah operator dalam RapidMiner berkisar 500 yang diantaranya merupakan operator untuk *input, output, data preprocessing* dan visualisasi. RapidMiner dapat digunakan dan bekerja pada semua sistem operasi karena menggunakan bahasa java.

RapidMiner sebelumnya bernama YALE (Yet Another Learning Enviroment), dimana versi awalnya mulai dikembangkan pada tahun 2001 oleh Ralf Klinkenberg, Ingo Mierswa, dan Simon Fischer di Artificial Intelligence Unit dari University of Dortmund. RapidMiner didistribusikan di bawah lisensi AGPL

(GNU Affero General Public License) versi 3. RapidMiner menyediakan GUI (Graphic User Interface) untuk merancang sebuah *pipeline* analisis. GUI ini akan menghasilkan file XML (Extensible Markup Language) yang mendefinisikan proses analitis keinginan pengguna untuk diterapkan ke data. File ini kemudian dibaca oleh RapidMiner untuk menjalankan analisis secara otomatis [23].

Beberapa sifat dari rapidminer, antara lain [24]:

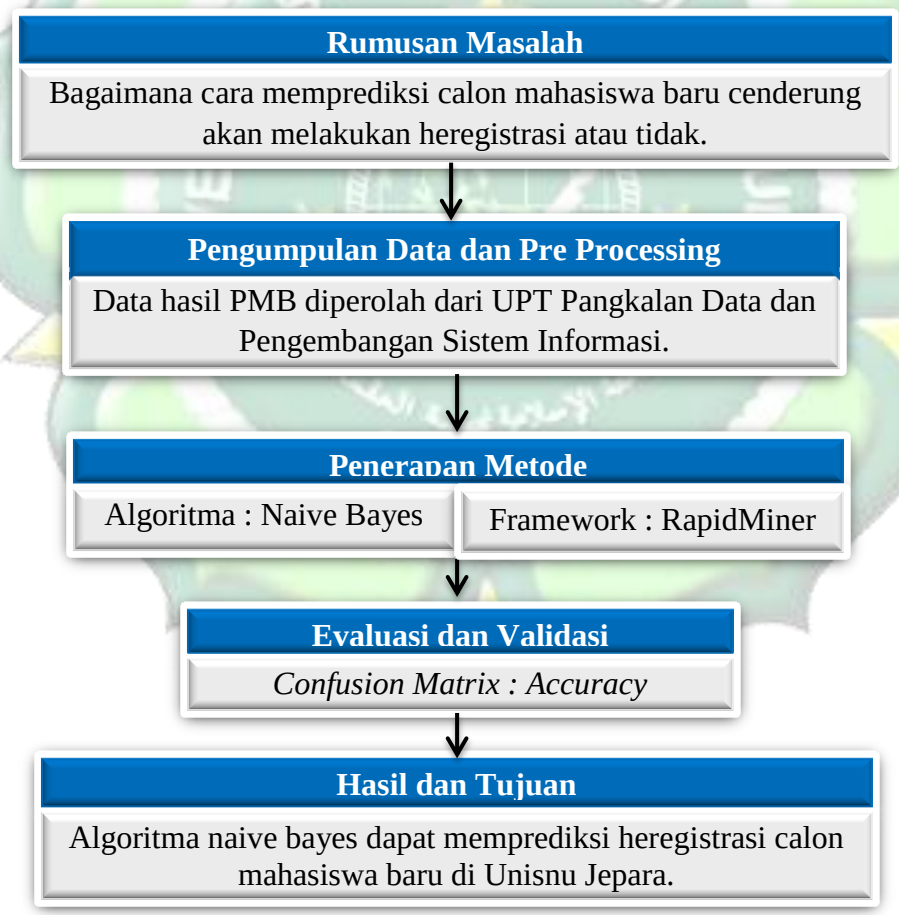
a. Berlisensi gratis (*open source*).
b. Multiplatform karena diprogram dalam bahasa Java.
c. Proses penemuan pengetahuan dimodelkan sebagai operator *trees*.
d. Internal data berbasis XML sehingga memudahkan pertukaran data eksperimen.
e. Dilengkapi dengan *scripting language* untuk otomatisasi eksperimen
f. Mempunyai konsep multi – layer yang bisa menjamin tampilan data yang efektif dan terjaminnya penanganan data.
g. Mempunyai Java API, GUI, dan *command line mode*.

Rapid Miner mempunyai beberapa fitur, yaitu:

a. Terdapat banyak algoritma data mining, seperti *self- organization map* dan *decision tree* .
b. Gambar grafis yang canggih, seperti diagram histogram yang tumpang tindih, *3D Scatter plots* dan *tree chart*.
c. Terdapat banyak puglin, seperti *text plugin* untuk melakukan analisis teks.
d. Mempunyai fitur *machine learning* dan proses data mining termasuk: data *preprocessing*, ETL *(extraction, transformation, loading), modelling*, evaluasi, dan visualisasi.
e. Proses data mining tersusun atas operator-operator yang *nestable*, dibuat dengan GUI dan dideskripsikan dengan XML.
f. Mengintegrasikan proyek data mining statistika R dan Weka.

2.3 Kerangka Pemikiran

Kerangka pemikiran dalam penelitian ini adalah dasar pemikiran dalam penerapan algoritma Naive Bayes untuk prediksi heregistrasi calon mahasiswa baru di UNISNU Jepara. Oleh sebab itu, penelitian ini menggunakan data mining metode klasifikasi algoritma Naive Bayes dengan menggunakan dataset hasil penerimaan mahasiswa baru pada tahun 2019-2020 dengan 14 (empat belas) atribut dengan rincian 1 (satu) atribut ID yaitu nama, 12 (dua belas) atribut reguler yaitu tahun pendaftaran, program kelas, jenis kelamin, usia, prodi, kota asal, pekerjaan ayah, pekerjaan ibu, penghasilan orangtua, jurusan sekolah asal, nilai UN, informasi pendaftaran, dan 1 (satu) atribut kelas yaitu status heregistrasi yang akan dimasukkan ke dalam RapidMiner dan akan dihitung akurasinya menggunakan *Confusion Matrix*. Adapun kerangka pemikiran sebagai berikut :



Gambar 2.3 Kerangka Pemikiran