

BAB II

TINJAUAN PUSTAKA

2.1 Penerimaan Mahasiswa Baru

Penerimaan Mahasiswa Baru (PMB) merupakan salah satu kegiatan rutin yang dilaksanakan oleh seluruh perguruan tinggi di Indonesia. Kegiatan ini adalah titik awal dalam pencarian mahasiswa baru yang berkualitas (Kurniawan, 2015).

Proses PMB pada perguruan tinggi berguna untuk menyaring mahasiswa baru yang sesuai dengan kriteria yang telah ditentukan oleh Institusi. Pelaksanaan PMB umumnya dilakukan melalui beberapa tahapan di antara, pendaftaran, pengisian identitas, tes tertulis, hingga pengumpulan berkas. Tahapan dari proses PMB ini juga dilakukan oleh IIB Darmajaya Lampung.

Kampus IIB Darmajaya memiliki visi dan misi sebagai berikut, visi:

Menjadi perguruan tinggi pembelajaran unggul berbasis riset dan teknologi informasi.

Misi:

- a. Menyelenggarakan pembelajaran yang kreatif, inovatif dan adaptif terhadap kemajuan teknologi informasi berdasarkan pendekatan riset dan nilai-nilai kejujuran.
- b. Mengembangkan penelitian dan pengabdian pada masyarakat yang unggul dan inovatif serta relevan dengan kebutuhan *stakeholder*.
- c. Mengembangkan budaya profesionalisme dalam tata kelola perguruan tinggi berbasis teknologi informasi.
- d. Mengembangkan lingkungan kampus yang kondusif.

2.2 Promosi

Menurut Charles W. Lamb (2001) Promosi merupakan pengiriman informasi oleh pemasar kepada calon pembeli, untuk mendapatkan suatu respon atau pendapat sehingga menghasilkan peningkatan calon pembeli. Sedangkan menurut Irwan Dani (1999) Promosi ialah kegiatan yang dilaksanakan supaya calon pelanggan minat terhadap usaha, jasa atau barang yang sedang ditawarkan, sehingga terdorong untuk membeli.

Menurut (Kotler, 2007: 205) Unsur-unsur dalam *promotional mix* atau bauran promosi ada lima, yaitu:

- a. Pemasaran Langsung (*Direct Marketing*)
- b. Hubungan Masyarakat (*Public Relation*)
- c. Promosi Penjualan (*Sales Promotion*)
- d. Penjualan Personal (*Personal Selling*)
- e. Periklanan (*Advertising*)

2.3 Data Mining

Data *mining* merupakan proses penggalian informasi yang sangat berguna dari simpanan basis data yang besar. Data *mining* dapat diartikan proses pengekstrakan informasi yang baru dengan menggunakan data yang cukup besar sehingga dapat membantu dalam pengambilan keputusan. *Knowledge discovery* merupakan istilah lain dari data *mining* yang sering digunakan oleh banyak orang. Salah satu teknik dalam data *mining* adalah proses penelusuran dari data set untuk membangun sebuah model, setelah membangun model kemudian menggunakan model tersebut untuk mengidentifikasi data yang tidak tersimpan di dalam *database* tersebut. (Eko Prasetyo, 2012: 2)

Data *mining* juga dapat diartikan sebagai penentuan pola dari sejumlah data untuk mendapatkan informasi yang berguna. Algoritma data *mining* memiliki 2 kategori, yakni deskriptif dan predikatif. Deskriptif dalam data *mining* merupakan mengukur hubungan yang belum teridentifikasi dan mengukur kesamaan antar objek, sedangkan predikatif merupakan penyimpulan *rule* prediksi berdasarkan data pelatihan, kemudian *rule* tersebut diterapkan dalam data yang belum terprediksi (sartika & sensuse, 2017).

Hermawan Fajar Astuti (2013: 3) berpendapat bahwa Proses data *mining* yaitu menentukan model dan pola baru yang sempurna (sahih), dapat dimengerti dan bermanfaat dalam suatu basis data yang sangat besar (*massive databases*).

1. Sahih: dapat digenerasi untuk masa yang akan datang.
2. Bermanfaat: dapat digunakan untuk melakukan suatu tindakan.
3. Baru: apa yang sedang tidak diketahui.
4. Interaktif: memerlukan interaksi manusia dalam prosesnya.
5. Interaktif: memerlukan sejumlah proses yang diulang.

2.3.1 Teknik Data Mining

Hermawan Fajar Astuti (2013: 14-19) Beberapa sifat dan teknik dalam data *mining* terdapat 6 teknik, yaitu:

a. Klasifikasi (*Predictive*)

Klasifikasi atau *classification* merupakan penentuan sebuah *record* atau pengelompokan data baru kedalam sebuah katagori yang telah didefinisikan sebelumnya. Biasa disebut dengan *supervised learning*.

b. Klasterisasi (*Descriptive*)

Klasterisasi atau *clustering* memproses dengan cara mempartisi data set menjadi kelompok atau beberapa sub-set sedemikian rupa sehingga menjadi elemen-elemen dari suatu kelompok tertentu memiliki set properti yang di bagikan bersama, dengan tingkat persamaan yang tinggi dalam satu kelompok dan tingkat persamaan yang rendah untuk antar kelompok. biasa disebut dengan *unsupervised learning*.

c. Regresi (*Predictive*)

Regresi atau *regression* merupakan proses prediksi berdasarkan nilai variabel untuk mendapatkan variabel kontinyu, dengan menggunakan sebuah model ketergantungan nonlinier dan linier. Teknik ini sering digunakan dalam bidang jaringan saraf tiruan (*neural network*) dan statistika.

d. Kaidah Asosiasi (*Descriptive*)

Kaidah asosiasi atau *association rules* yaitu Mengidentifikasi kumpulan atribut-atribut yang muncul secara bersamaan (*co-occur*) dalam siklus yang kerap dan membentuk sejumlah aturan dari kumpulan data tersebut. Contoh, dalam supermarket 80% konsumen yang membeli kopi dan membeli gula, 50% dari semua yang berbelanja membeli keduanya.

e. Pencarian Pola sekuensial (*Descriptive*)

Proses mencari sejumlah *event* yang secara umum terjadi secara bersamaan. Contoh, dalam suatu set urutan DNA, ACGTC diikuti oleh GTCA sesudah suatu celah selebar 9 dengan probabilitas sebesar 30%.

f. Deteksi Devinasi (*Predictive*)

Supriyanta & Hasan (2013) Deteksi Devinasi atau *deviation detection* merupakan menerapkan deteksi anomali secara otomatis, yang berujuan agar dapat mendeteksi. Kebiasaan suatu entitas serta menetapkan beberapa *norm* melalui *pattern discovery*. Percobaan dari *norm* yang berdeviasi kemudian dideteksi seperti selaku tidak biasa.

2.3.2 Tahapan Data Mining

Ayudhitama & pujianto (2020) menegaskan bahwa dalam data *mining* terdapat tahapan-tahapan yang perlu dilakukan, diantaranya:

1. Seleksi data, dalam tahapan ini akan dilakukan proses penyeleksian data dari *database* yang berarti bahwa data yang ada didalam *database* tidak semuanya digunakan. Data yang sesuai dengan tujuan saja yang diambil dan diproses.
2. Pembersihan data, proses ini merupakan proses yang akan menghilangkan data dari *noise* atau data hilang (*missing*), dengan menggunakan data *mining* sehingga data *missing* dapat selesai.
3. Transformasi data, pada tahapan yang ketiga yaitu proses untuk menggabungkan dan mengubah data kedalam format yang sesuai dengan yang diharapkan.
4. Integrasi data, tahapan ini merupakan proses dalam penggabungan data dari beberapa *database* kemudian menjadi *database* yang baru.
5. Evaluasi pola, proses ini merupakan proses pendeteksian model atau pola prediksi dalam data *mining*.
6. Proses *mining*, pada tahapan ini bertujuan untuk mendapatkan ilmu pengetahuan dari data yang diproses.
7. Presentasi pengetahuan, proses yang terakhir yaitu bertujuan untuk memvisualisasikan atau menampilkan pengetahuan perihal metode yang digunakan.

2.4 Klasifikasi

klasifikasi atau Classification merupakan upaya untuk mendapatkan model yang membedakan dan menjelaskan class data, atau proses pengklasifikasian satu atau

beberapa class yang telah diidentifikasi sebelumnya. Berikut algoritma klasifikasi yang telah banyak digunakan yaitu, Naïve buyes, *decision tree*, K-nearest neighbor, Neural rough sets, dan sebagainya (Saputra & Primadasa, 2018).

Klasifikasi juga dapat diartikan sebuah proses untuk penentuan kelas label dari data yang ingin diklasifikasi sehingga dapat membantu seseorang dalam penentuan keputusan. Tujuan dari klasifikasi dapat meningkatkan keakuratan hasil yang diperoleh berdasarkan data yang digunakan (Hendrian, 2018).

(Sugianto, 2017) Klasifikasi memiliki komponen-komponen penting, diantaranya:

1. Kelas adalah variabel yang tidak diperbolehkan bebas dikarenakan akan dijadikan label untuk hasil klasifikasi.
2. prediktor, berbeda dengan kelas pada komponen ini diperbolehkan bebas suatu model, dengan menggunakan acuan dari karakteristik atribut data yang akan diklasifikasikan.
3. Set data latih adalah sekelompok data yang lengkap yang terdiri dari kelas dan prediktor. Dalam proses ini data latih perlu diolah agar menghasilkan model pengklasifikasian yang tepat.
4. Set data tes bergungsi untuk mengetahui akurasi berdasarkan model yang sudah terbentuk.

2.4.1 K-Nearest Neighbor

Algoritma *K-nearest neighbor* salah satu algoritma dengan teknik klasifikasi data yang cukup baik. Algoritma ini menggunakan data latih kemudian disimpan, sehingga jika ada data baru atau data *testing* maka dapat mengidentifikasi berdasarkan data yang disimpan sehingga mendapatkan nilai kemiripan (Saputra & Primadasa, 2018)

Algoritma KNN (*K-Nearest Neighbor*) juga dapat dikatakan bekerja secara sederhana, yaitu dengan cara kasus baru diklasifikasikan berdasarkan jarak data baru ke tetangga terdekat atau beberapa data. Algoritma KKN mengklasifikasikan dan menyimpan kasus-kasus baru berdasarkan fungsi jarak atau kemiripan (Sagala & Tampubolon 2018). Kelebihan dari algoritma ini pelatihan dari data trening (latih) sangat sederhana, cepat, mudah dipelajari, tetap efektif meski data treningnya besar. Kekurangan dari algoritma ini adalah mudah tertipu

apabila ada atribut yang tidak relevan, serta komputasi yang dibutuhkan kompleks dan memori terbatas (Ayudhitama & Pujianto, 2020).

Saputra & Primadasa (2018) berpendapat bahwa Terdapat banyak cara dalam pentuan jarak kedekatan antar data pada algoritma *K-nearest neighbor* salah satunya dengan memanfaatkan *Euclidean distance*. *Euclidean distance* merupakan cara untuk menghitung jarak antar data yang telah banyak digunakan ilmuwan. Jarak ini digunakan menguji interpretasi korelasi jarak antar 2 objek. Berikut rumus dari *Euclidean distance*:

$$\sqrt{\sum_{i=1}^n (X_{i2} - X_{i1})^2} \dots \dots \dots (1)$$

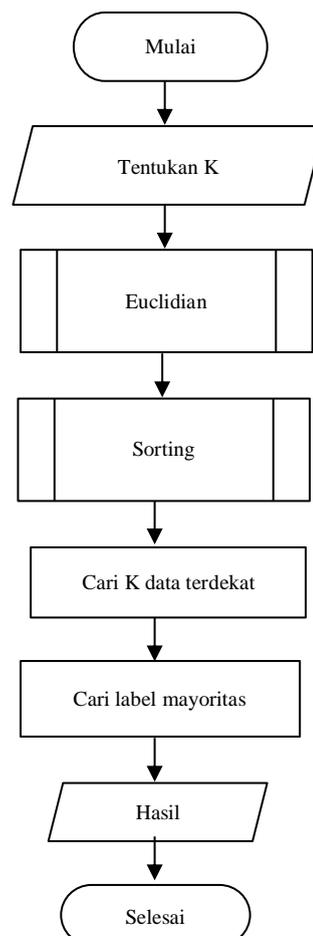
X_{i2} : data uji

X_{i1} : data *training*

i : record (baris) ke- i dari tabel

n : jumlah data *training*

Alur dalam algoritma K- *Nearest Neighbour* dapat dilihat pada gambar di bawah ini:



Gambar 2.1 Alur *K-Nearest Neighbor*

Langkah-langkah *K-Nearest Neighbor*:

1. Menentukan nilai parameter *K* pada saat testing, nilai *K* ditentukan menggunakan nilai *K* optimum pada saat *training*.
2. Menghitung nilai *euclidean distance* dari setiap objek dari setiap data *training* yang digunakan.
3. Nilai *euclidean distance* yang didapat kemudian diurutkan berdasarkan nilai yang terkecil.
4. Mengelempokan klasifikasi *Nearest Neighbour* atau mengumpulkan kategori *Y*
5. Untuk mendapatkan hasil klasifikasi maka menggunakan kategori mayoritas.

2.4.2 *Decision Tree*

Pohon keputusan atau sering disebut *decision tree* merupakan algoritma yang sering digunakan untuk mengambil keputusan. Konsep algoritma ini akan menjadikan struktur hirarki atau pohon yang kemudian dijadikan sebagai alur untuk menentukan prediksi. Setiap pohon mempunyai cabang, untuk menuju cabang yang lain maka ada beberapa atribut yang harus di lengkapi dan berakhir ketika tidak ada cabang lagi atau biasa disebut daun (Sartika & Indra, 2017).

Larose (dalam Septiani, 2017) menjelaskan bahwa algoritma C4.5 memakai konsep *entropy reduction* atau *information gain* untuk penentuan pembagian yang terbaik. Tahapan dalam membuat pohon keputusan menggunakan algoritma C4.5 yaitu:

1. Menentukan data training, data set atau data *training* dapat menggunakan data history yang sudah ada pada sebelumnya, dan data kemudian dikelompokan dalam kelas-kelas tertentu.
2. Menentukan akar dari pohon berdasarkan nilai *index entropy* terendah atau menghitung nilai *gain* yang tertinggi dari masing-masing atribut. Berikut

rumus untuk menghitung nilai *indek entropy*:

$$Entropy(s) = \sum_{i=1}^n -P_i \text{Log}_2 P_i \dots\dots\dots(2)$$

- s : himpunan kasus
 k : jumlah pasrtisi s
 P_j : proporsi S_i terhadap S

3. Menghitung nilai *gain* dengan rumus:

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * Entropy(S_i) \dots\dots\dots(3)$$

- S : himpunan kasus
 A : Atribut
 n : jumlah pada atribut A
 $|S_i|$: jumlah kasus pada partisi ke- i
 $|S|$: jumlah kasus dalam S

4. Lakukan perulangan dari langkah kedua hingga semua data terpartisi. Akan berhenti proses partisi apabila:
- Dalam record yang dipartisi tidak ada atribut lagi.
 - Dalam cabang yang sama tidak memiliki *record*.
 - Pada simpulan N semua tupel dalam record mendapat kelas yang sama.

Menurut Hendrian (2018) Pohon keputusan terbentuk berdasarkan himpunan dari data yang digunakan untuk pembelajaran (data *training*). Himpunan yang berbeda digunakan untuk proses pengujian dalam pengecekan model yang telah terbentuk, data pengujian sering disebut dengan data *testing*. Pohon keputusan atau *decision tree* memiliki beberapa kelebihan, diantaranya:

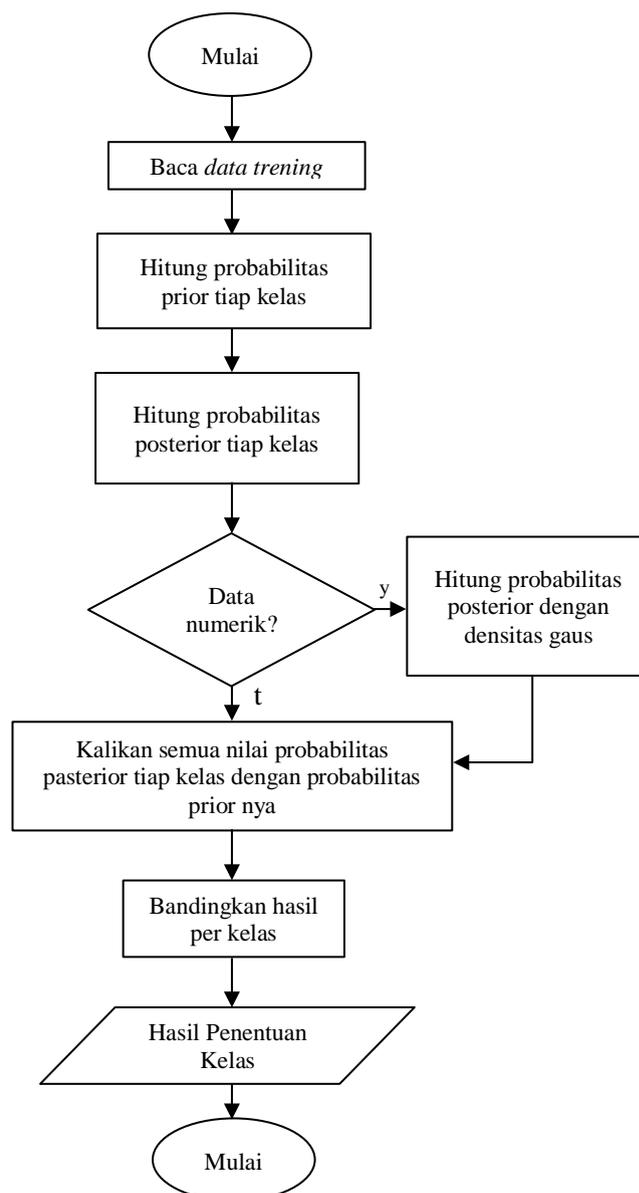
- Dapat memproses dengan data besar (dalam kerangka *ensemble*).
- Memiliki kinerja prediktif yang tinggi untuk upaya komputasi yang relatif kecil.
- Tersedia dalam berbagai paket data *mining* melalui berbagai *platform*.
- Dapat menyesuaikan dalam data set yang terdapat nilai yang hilang, atau kesalahan lainnya.
- mudah diikuti (ketika dipadatkan) dan cukup jelas.
- Dapat menangani berbagai *type* data baik numerik, tekstual, maupun nominal.

2.4.3 Naïve Bayes

Naïve Bayes merupakan algoritma yang mudah diimplementasikan dalam data *mining* dibandingkan algoritma lainnya dalam lingkup klasifikasi. Algoritma ini juga dapat memproses teks dan angka (Sagala & Tampubolon, 2018).

Bramer (dalam Septiani, 2017) menjelaskan bahwa kata *naïve*, yang memiliki kesan merendahkan bermula dari asumsi independensi pengaruh nilai suatu atribut dari probabilitas berdasarkan kelas yang diberikan kepada nilai atribut lainnya. Penerapan teorema *bayes* dalam algoritma ini dengan mengkombinasikan probabilitas bersyarat dan *prior probability* dalam sebuah rumus, untuk mendapatkan klasifikasi yang mungkin maka dapat menghitung nilai probabilitas. Klasifikasi bayes memproses sesuai dengan *teorema bayes*.

Berikut adalah alur kerja dari algoritma *Naïve bayes*:



Gambar 2.2 Alur Metode Algoritma *Naïve Bayes*

Algoritma ini memiliki kelebihan diantaranya hasil akurasi relatif tinggi serta dalam penggunaan dan pelatihannya termasuk kategori efisien. Sedangkan kekurangan algoritma ini wajib memberi batasan (threshold) dengan manual (Ayudhitama & Pujianto, 2020).

Berikut adalah rumus dalam algoritma *naïve bayes*:

$$P(H|X) = \frac{P(X|H) P(H)}{P(X)} \dots\dots\dots(4)$$

Keterangan:

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

Dapat dijelaskan yaitu $P(H|X)$ merupakan probabilitas dari H yang di dalam X atau bahasa lain dari $P(H|X)$ merupakan presentase dengan banyaknya H yang di dalam X , $P(X|H)$ merupakan probabilitas X di dalam H , $P(H)$ merupakan probabilitas dari H dan $P(X)$ merupakan probabilitas prior dari X .

2.5 *Cross Validation*

(Ayudhitama & Pujianto, 2020) *Cross validation* atau cross validasi merupakan sebuah metode statistik yang dapat digunakan dalam mengevaluasi kinerja algoritma atau model, dengan cara data akan dipisahkan menjadi 2 subset yaitu data proses pembelajaran dan data evaluasi atau validasi. Algoritma atau model dilatih oleh subset pembelajaran dan divalidasi oleh subset validasi dengan

pemilihan jenis *Cross Validation* yang dapat didasarkan pada ukuran dataset. Biasanya *Cross Validation K-fold* digunakan karena dapat mengurangi waktu komputasi dengan tetap menjaga keakuratan estimasi. Berikut merupakan tabel *confusion matrix* untuk mendapatkan performa *accuracy*, *precision*, dan *recall*:

Tabel 2.1 Confusion Matrix

| Actual Class | Predicted Class | |
|--------------|-----------------|----|
| | + | - |
| + | TAPI | FN |
| - | FP | TN |

Keterangan:

- TP : True Positif
- TN : True Negatif
- FP : False Positif
- FN : False Negatif

a. *Accuracy*

Accuracy atau akurasi digunakan untuk mengukur dan mengevaluasi keakuratan dari hasil klasifikasi, semakin besar nilai akurasi maka semakin bagus tingkat pengklasifikasiannya, berikut merupakan rumus perhitungan akurasi:

$$Accuracy = \frac{TP+TN}{TP+FN+TP+TN} \dots\dots\dots(5)$$

b. *Precision*

Precision atau biasa disebut presisi atau biasa dikenal dengan nama *confidence* merupakan sebuah model perhitungan untuk mencari hasil dari proporsi jumlah kasus dengan hasil diagnosa positif. Berikut merupakan rumus perhitungan nilai presisi:

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots(6)$$

c. *Recall*

Recall atau *sensitivity* merupakan sebuah model perhitungan untuk mencari hasil

proporsi jumlah kasus positif yang diidentifikasi dengan benar. Berikut merupakan rumus perhitungan nilai *recall*:

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots(7)$$

2.6 Rapid miner

Rapid Miner adalah perangkat lunak (software) yang dibuat menggunakan bahasa java dibawah lisensi GNU Public License dan bersifat open source, sehingga *rapid miner* dapat dijalankan diberbagai sistem operasi. Perangkat lunak ini dibuat oleh Dr. Markus Hofmann dari Institute of Technology Blanchardstown dan Raif Klinkenberg dari rapid-i.com. perangkat lunak ini khususkan untuk penerapan data *data mining*, dalam penerapannya tidak dibutuhkan kodingan khusus disebabkan fasititis telah tersedia untuk user (Hendrian, 2018).

2.7 Penelitian Terkait

Tahun 2016 Rony Setiawan menghasilkan penelitian yang berjudul Penerapan Data Mining Menggunakan Algoritma *K-means Clustering* untuk Menentukan Strategi Promosi Mahasiswa Baru. Penelitian ini membahas tentang penerapan data *mining*, dengan menggunakan satu algoritma yaitu *K-means Clustering* untuk mendapatkan profil yang sama melalui kemiripan atribut. Penelitian yang dilakukan Rony Setiawan menggunakan metode CIRSP-DM dengan melewati proses *business understanding, modelling, evaluation, data understanding, data preparation, dan deployment*. Atribut yang digunakan dalam penelitian ini terdiri dari jenis kelamin, usia, asal kota, agama, pekerjaan orang tua, jurusan, nilai tes, gelombang pendaftar, program studi, lokasi kampus, pembayaran uang registrasi, jenis sekolah, status ujian, gelombang registrasi, sumber informasi. Penelitian ini menghasilkan 4 kluster, yang terdiri dari kluster 1 dengan kelompok calon mahasiswa yang tidak lulus, kluster 2 kelompok yang lulus dengan kategori ekonomi atas, kluster 3 kelompok ekonomi rendah, dan kluster 4 digolongkan masyarakat dengan ekonomi menengah.

Penelitian Ramdani Budiman & Rudianto (2019) yang berjudul Penerapan Data Mining untuk Menentukan Lokasi Promosi Penerimaan Mahasiswa Baru pada

Universitas Baten Jaya. Penelitian ini bertujuan untuk *menentukan* atau mengelompokkan daerah yang berminat untuk melanjutkan keperguruan tinggi di kampus tersebut. Algoritma yang digunakan dalam penelitian ini yaitu *K-means clustering* serta atribut yang digunakan terdiri dari asal sekolah, minat prodi, dan asal kecamatan. Hasil dari penelitian ini terdiri dari 3 kluster yaitu kluster rendah, cluster sedang, dan yang terakhir kluster tinggi.

Penelitian yang dilakukan Aries Saifudin (2018) dengan judul Metode Data Mining untuk Seleksi Calon Mahasiswa pada Penerimaan Mahasiswa Baru di Universitas Pamulang. Data yang digunakan dalam penelitian ini menggunakan data mahasiswa yang masa belajarnya lebih dari 8 semester dan menggunakan data alumni. Penelitian ini menggunakan 6 atribut yaitu terdiri dari, status ketepatan waktu, nilai matematika, jenis kelamin, nilai ujian Bahasa Indonesia, jurusan, nilai ujian Bahasa Inggris. Penelitian ini menggunakan metode data mining untuk memprediksi, yaitu *Rule Induction, K-Nearest Neighbor, Support Vector Machine, Naïve Bayes, linear discriminant analysis, DecisionStump, neural network, Decision tree, Randem Forest, dan Linear Regression*. Hasil dari penelitian ini diperoleh algoritma terbaik yaitu *Support Vector Machine* dengan akurasi 65.00% namun nilai ini masih jauh dari nilai *excellent*.

Jurnal penelitian Annisa Putri Ayudhitama & Utomo Pijianto (2020) yang berjudul Analisa 4 Algoritma dalam Klasifikasi Penyakit Liver Menggunakan Rapidminer. Penelitian ini membahas tentang mendiagnosa penyakit liver dan Analisa algoritma. Penelitian ini menggunakan algoritma *neural network, Naïve Bayes, Decision tree, Nearest Neighbor*. Atribut yang digunakan dalam penelitian ini terdiri dari *Ratio, Age, A/G, Sgot, DB, Gender, ALB, Alkpho, TB, Dataset, Sgpt, hingga TP*. Berdasarkan hasil penelitian yang telah dilakukan algoritma yang paling tepat dalam dalam pengklasifikasian pasien liver merupakan algoritma *Decision tree*, dengan akurasi sebesar 72,89%.

Tabel 2.2 Penelitian Terkait.

| Peneliti | Domain | Metode | Hasil |
|------------------|---|-------------------------------------|---|
| Rony Setiawan | Strategi promosi mahasiswa baru, dengan indikator jenis kelamin, usia, asal kota, agama, pekerjaan orang tua, jurusan, nilai tes, gelombang pendaftar, program studi, lokasi kampus, pembayaran uang registrasi, jenis sekolah, status ujian, gelombang registrasi, sumber informasi. | <i>K-means</i> <i>Clustering</i> | Penelitian ini menghasilkan 4 klaster yang terdiri dari tidak lulus, lulus dengan ekonomi atas, lulus dengan ekonomi rendah, dan lulus dengan ekonomi menengah. |

| | | | |
|-------------------------------------|--|--|---|
| Ramdani Budiman & Rudianto | Penentuan lokasi promosi, dengan indikator asal sekolah, minat prodi, dan asal kecamatan. | <i>K-means clustering</i> | Penelitian ini terdiri dari 3 kluster yaitu kluster rendah, cluster sedang, dan yang terakhir kluster tinggi. |
| Aries Saifudin | Seleksi calon mahasiswa baru, dengan indikator status ketepatan waktu, nilai matematika, jenis kelamin, nilai ujian Bahasa Indonesia, jurusan, nilai ujian Bahasa Inggris. | <ul style="list-style-type: none"> - <i>Rule Induction</i> - <i>K-Nearest Neighbor</i> - <i>Support Vector Machine</i> - <i>Naïve Bayes</i> - <i>Linear Discriminant Analysis</i> - <i>Decition Stump</i> - <i>Neural Network</i> - <i>Decision Tree</i> - <i>Randem Forest</i> - <i>Linear Regression</i> | <ul style="list-style-type: none"> - <i>Rule Induction</i> memiliki akurasi 62,03% - <i>K-Nearest Neighbor</i> memiliki akurasi 57,25% - <i>Support Vector Machine</i> memiliki akurasi 65,00% - <i>Naïve Bayes</i> memiliki akurasi 60,85% - <i>linear Discriminant Analysis</i> memiliki akurasi 31,44% - <i>DecisionStump</i> memiliki akurasi 62,65% - <i>Neural Network</i> memiliki akurasi 55,65% - <i>Decision tree</i> memiliki akurasi 62,65% |

| | | | |
|---|---|--|--|
| | | | <ul style="list-style-type: none"> - <i>Random Forest</i> memiliki akurasi 62,65% - <i>Linear Regression</i> memiliki akurasi 57,91% |
| Annisa Putri Ayudhita ma & Utomo Pijianto | Klasifikasi penyakit liver, dengan indikator <i>Ratio, Age, A/G, Sgot, DB, Gender, ALB, Alkpho, TB, Dataset, Sgpt,</i> hingga <i>TP</i> . | <ul style="list-style-type: none"> - <i>Neural Network</i> - <i>Naïve Bayes</i> - <i>Decision tree</i> - <i>K-Nearest Neighbor</i> | <ul style="list-style-type: none"> - <i>Neural Network</i> memiliki akurasi 70,81% - <i>Naïve Bayes</i> memiliki akurasi 55,44% - <i>Decision tree</i> memiliki akurasi 72,89% - <i>K-Nearest Neighbor</i> memiliki akurasi 66,32% |
| Robby Toro | Penentuan lokasi promosi dengan klasifikasi, indikator nama, asal sekolah, pengasihan orang tua, wilayah, total alumni | <ul style="list-style-type: none"> - <i>K-Nearest Neighbor</i> - <i>Decision Tree</i> - <i>Naïve Bayes</i> | <ul style="list-style-type: none"> - <i>K-Nearest Neighbor</i> memiliki akurasi 99,61% - <i>Decision Tree</i> memiliki akurasi 100% - <i>Naïve Bayes</i> memiliki akurasi 84,78% |