

BAB II

LANDASAN TEORI

2.1 Data Mining

2.1.1 Pengertian *Data Mining*

Di era sekarang ini, data merupakan sesuatu hal yang penting dalam dunia sistem teknologi informasi, khususnya di bidang pendidikan, perbankan, sumber energi dan sektor lainnya yang mengharuskan setiap perusahaan untuk memperoleh informasi penting dengan perhitungan statistik, matematika dan teknologi kecerdasan buatan (*Artificial Intelligence*). Perlu kita ketahui bahwa *data mining* sendiri dapat menggambarkan kumpulan dari beberapa teknik yang bertujuan untuk menemukan pola yang tidak diketahui dalam data yang terkumpul. Melalui data ini, pengguna dapat menemukan pengetahuan berupa *database* yang sebelumnya tidak mungkin diketahui. Berikut penulis uraikan beberapa definisi *data mining* dari beberapa sumber, baik jurnal maupun para ahli.

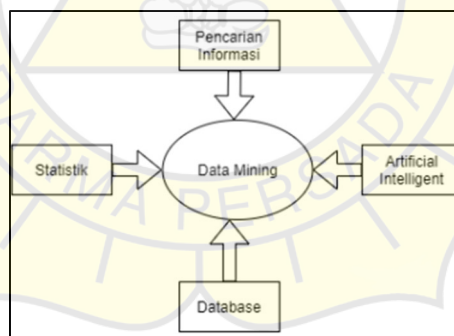
Menurut Efori Buulolo (2020) dalam *e-book* yang berjudul “*Data mining* untuk Perguruan Tinggi”. *Data Mining* atau kadang disebut juga *Knowledge Discovery in Database* (KDD) merupakan aktivitas yang berkaitan dengan pengumpulan data, pemakaian data historis untuk menemukan pengetahuan, informasi, keteraturan, pola atau hubungan dalam data yang berukuran besar. *Output* dalam *data mining* dapat dipergunakan sebagai alternatif dalam pengambilan keputusan atau untuk memperbaiki keputusan di masa yang akan datang.

Data mining salah satu bidang ilmu yang perkembangan sangat pesat, perkembangan tersebut disebabkan oleh beberapa faktor, antara lain :

1. Semakin tingginya kesadaran akan pentingnya data.

2. Semakin tingginya pemanfaatan *output* dari hasil pengolahan data dalam berbagai bidang contohnya bidang bisnis.
3. Perkembangan kumpulan data yang begitu cepat.
4. Peningkatan akses internet baik melalui navigasi web ataupun melalui *smartphone*.
5. Perkembangan *hardware* dan *software* khususnya yang berhubungan dengan *data mining*.
6. Perkembangan yang begitu cepat dalam bidang komputasi *computer*.
7. Media penyimpanan yang semakin besar dengan harga yang semakin terjangkau.

Data mining bukanlah bidang ilmu yang berdiri sendiri, tetapi sangat berkaitan dengan bidang ilmu lain seperti *database*, *statistic*, pencarian informasi, dan *artificial intelligent*.



Gambar 2. 1 Bidang Ilmu Data Mining

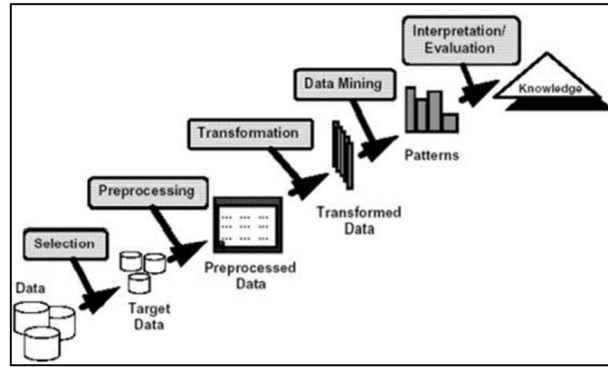
Berikut adalah Penjelasan dari gambar diatas:

1. *Database – Data mining* : Kumpulan data yang digunakan dalam *data mining* bersumber salah satunya dari *database*. Data yang digali/dicari informasinya dipisahkan dari data operasional yang di *database*.

2. *Statistic – Data mining* : Dalam pengambilan keputusan, *statistic* membutuhkan data mulai dari pengumpulan data, pengambilan sampel data dan probabilitas. *Data mining* dalam penentuan sampel data, menganalisa, dan mempresentasikan *output* menggunakan teknik *statistic*.
3. Pencarian Informasi – *Data mining* : Pencarian informasi merupakan salah satu kegiatan dalam proses *data mining* yang meliputi interpretasi, analisis dan penyimpanan data.
4. *Artificial Intelligence : Data mining* : Salah satu cabang ilmu dari *Artificial Intelligent* adalah *machine learning*. *Machine learning* merupakan disiplin ilmu yang penting dalam *data mining* dimana *system computer* belajar dari training data yang digunakan.

2.1.2 Tahapan *Data Mining*

Sebagai suatu proses pengolahan data informasi mentah menjadi data informasi yang berguna pada tingkat atas dalam menentukan kebijakan, *data mining* memerlukan serangkaian proses untuk mencapai data yang diinginkan, tidak mudah untuk mengolah data abstrak atau data yang bersumber dari banyak pihak, namun pada tahap proses *data mining* sudah mampu mengolah data tersebut menjadi data yang mudah dibaca. Penambangan data juga interaktif karena pengguna terlibat langsung atau dengan penyampaian basis pengetahuan. Berikut ini adalah tahapan dalam *data mining*. *Data Mining* mempunyai beberapa tipe model proses yang digunakan untuk mengarahkan pelaksanaan *data mining*, model proses yang biasa digunakan, yaitu *Knowledge Discovery Databases (KDD)*, *CRISP-DM* dan *SEMMA*.



Gambar 2. 2 Bidang Ilmu *Data Mining*

Berikut penjelasan dari setiap tahapan pada KDD, CRISP-DM dan SEMMA:

a. *Domain Understanding and KDD Goals*

Tujuan ditentukan dari sudut pandang *user* dan digunakan untuk mengembangkan dan pemahaman tentang *domain* aplikasi dan pengetahuan sebelumnya.

b. *Selection and Additions*

Tahap kedua berfokus pada penentuan data target dan subset dari data sampel atau variabel.

c. *Preprocessing: Data Cleaning etc*

Pembersihan dan *preprocessing* data merupakan operasi dasar untuk menyelesaikan data yang konsisten tanpa *noisy*.

d. *Transformation*

Transformasi data dari satu bentuk ke bentuk lainnya sehingga data diimplementasikan dengan mudah.

e. *Data Mining (Choosing the Suitable Data Mining Task)*

Memilih metode *data mining* yang sesuai berdasarkan tujuan tertentu yang telah didefinisikan pada tahap pertama, contoh dari metode *data mining* adalah *classification*, *regression*, *clustering* dan *summarization*.

f. *Data Mining (Choosing the Suitable Data Mining Algorithm)*

Memilih algoritma yang tepat untuk pencarian pola-pola data, algoritma yang dipilih berdasarkan kecocokan kriteria dengan metode *data mining*.

g. *Data Mining (Implying Data Mining Algorithm)*

Pada tahap ini algoritma yang telah dipilih diimplementasikan.

h. *Evaluation and Interpretation*

Tahap ini berfokus pada interpretasi dan evaluasi yang mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan hipotesa yang ada sebelumnya.

i. *Discovered Knowledge*

Penggunaan pengetahuan yang ditemukan dari proses KDD, dimana memutuskan apa yang akan dilakukan dengan pengetahuan dihasilkan.

2.1.3 Pengelompokan *Data Mining*

Pengelompokan dalam *data mining* adalah pengertian yang luas tentang pengertian *data mining* pada berbagai macam dataset yang akan diuji, pola pengelompokan pada awalnya merupakan jawaban untuk jenis dataset tersebut agar dapat dieksekusi dengan benar dan dapat maksimal mendapatkan hasil akhir untuk dicapai. *Data mining* berdasarkan fungsionalitasnya dapat dikelompokkan menjadi enam bagian yaitu (Suyanto, 2017) :

1. *Classification*

Diterapkan pada data baru untuk mengelompokkan jenis objek. Klasifikasi termasuk pada model *supervised*. Pada persoalan klasifikasi kita memiliki sampel data dan memprediksi beberapa class yang ada berdasarkan sampel yang ada. Hanya satu atribut di antara banyak atribut yang disebut dengan atribut

predator. Klasifikasi ini juga umum, digunakan untuk pemodelan bisnis dan lainnya. Misal klasifikasi menentukan penyakit tertentu atau menentukan customer berdasarkan model pembayarannya.

2. *Clustering*

Berbeda dengan klasifikasi, *clustering* termasuk model *unsupervised*. *Clustering* mengelompokkan data yang tidak diketahui labelnya. *Clustering* yang diorganisasi ke dalam struktur hierarkikal akan mendefinisikan taksonomi dari data. Penerapan metode *clustering* yang tepat akan menghasilkan *clustering* yang berkualitas. Suatu cluster dikarakterisasi oleh centroid, atribut histogram dan *clustering* model *hierarchical tree*.

3. *Regression*

Merupakan suatu fungsi yang digunakan untuk memodelkan data untuk meminimalkan hasil kesalahan prediksi. Umumnya regresi dilakukan dengan data yang bersifat *time series*.

4. *Association Rule*

Merupakan pemodelan kebergantungan. Fungsi asosiasi ini biasanya kita kenal dengan istilah “*market basket analysis*” yang merupakan fungsi untuk menemukan relasi atau korelasi antara himpunan item-item. Aturan asosiasi diartikan pada basket data yang digunakan untuk keperluan promosi, desain katalog untuk meningkatkan penjualan. Contoh penerapan asosiasi adalah ketika customer membeli pamper maka ada kemungkinan membeli bir.

5. *Anomaly Detection*

Mengidentifikasi data yang tidak umum. Bisa berupa outlier, perubahan deviasi/bias yang penting dan perlu investigasi lebih lanjut.

6. *Summarization*

Menyediakan representasi data yang lebih sederhana meliputi pelaporan, visualisasi data yang dipergunakan untuk menunjang informasi dan penguatan keputusan.

2.2 Teknik Data Mining

Menurut Vulandari (2017) teknik yang digunakan dalam *data mining* erat kaitannya dengan “penemuan” (*discovery*) dan “pembelajaran” (*learning*) yang terbagi dalam tiga metode utama pembelajaran yaitu:

1. *Supervised Learning* *Supervised Learning* adalah teknik yang paling sering digunakan. Teknik ini sama dengan “*programming by example*”. Teknik ini melibatkan fase pelatihan dimana data pelatihan historis yang karakteristiknya dipetakan ke hasil hasil yang telah diketahui diolah dalam algoritma *data mining*. Proses ini melatih algoritma untuk mengenali variabel-variabel dan nilai-nilai kunci yang nantinya akan digunakan sebagai dasar dalam membuat perkiraan-perkiraan ketika diberikan data baru.
2. *Unsupervised Learning* Teknik pembelajaran ini tidak melibatkan fase pelatihan seperti yang terdapat 6 pada *supervised learning*. Teknik ini bergantung pada penggunaan algoritma yang mendeteksi semua pola seperti *associations* yang muncul dari kriteria penting yang spesifik dari data masukan. Pendekatan ini mengarah pada pembuatan banyak aturan (*rules*) yang mengkarakterisasikan penemuan seperti *associations*, *clusters* dan *segments*. Aturan-aturan ini kemudian dianalisis untuk menemukan hal-hal penting.
3. *Reinforcement Learning* Teknik pembelajaran ini jarang digunakan dibandingkan dengan dua teknik lainnya, namun memiliki penerapan-

penerapan yang terus dioptimalkan dari waktu ke waktu dan memiliki kontrol adaptif. Reinforcement *learning* sangat tepat digunakan untuk menyelesaikan masalah-masalah yang sulit serta bergantung pada waktu.

2.3 Teknik Classification

Dito Putro Utomo & Mesran (2020) dalam jurnal Media Informatika Budidarma dengan judul “Analisis Komparasi Metode Klasifikasi *Data Mining* dan Reduksi Atribut Pada Dataset Penyakit Jantung”. Klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukkannya ke dalam kelas tertentu dari jumlah kelas yang tersedia. Klasifikasi membangun model berdasarkan data latih yang ada, kemudian menggunakan model tersebut untuk mengklasifikasikan data baru. Klasifikasi dapat didefinisikan sebagai pekerjaan yang melakukan pelatihan/pembelajaran pada fungsi target yang memetakan setiap set atribut (fitur) ke satu jumlah label kelas yang tersedia. Suatu sistem yang melakukan klasifikasi diharapkan dapat mengklasifikasikan semua kumpulan data dengan benar, namun tidak dapat dipungkiri bahwa kinerja sistem tidak dapat 100% benar sehingga suatu sistem klasifikasi juga harus mengukur kinerjanya. Umumnya, pengukuran kinerja klasifikasi dilakukan dengan matriks konfusi.

2.4 Metode Penelitian

2.4.1 Metode Penelitian Yang Digunakan

Bagi sebuah penelitian, pemilihan metode menjadi sangat penting untuk mencapai suatu tujuan dengan menggunakan perhitungan yang sistematis, alur kerja dan studi literatur yang berkaitan dengan penelitian sejenis tentang bagaimana menyelesaikan suatu masalah dengan metode pendekatan/pengarahan yang tepat.

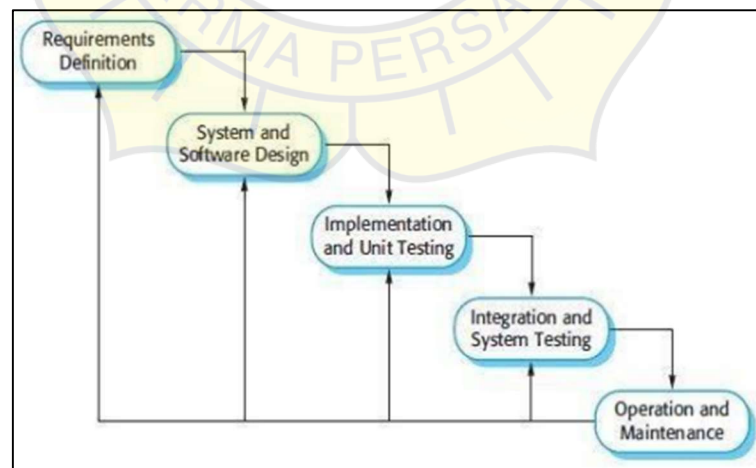
metode penelitian pada dasarnya merupakan cara ilmiah untuk mendapatkan data dengan tujuan dan kegunaan tertentu (Sugiyono, 2017).

Berdasarkan penjelasan yang dikemukakan oleh ahli di atas, maka penulis menggunakan penelitian deskriptif dengan memperhatikan rumusan masalah dalam penelitian tentang adanya variabel bebas.

2.5 Metodologi Pengembangan Sistem

2.5.1 Metodologi *Waterfall*

Metode Waterfall adalah suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. Dalam pengembangannya metode waterfall memiliki beberapa tahapan yang runtut: requirement (analisis kebutuhan), desain sistem (system design), Coding & Testing, Penerapan Program, pemeliharaan. (Chrisantus Trisianto. 2018. Jurnal Teknologi Informasi : Penggunaan Metode Waterfall Untuk Pengembangan Sistem Monitoring Dan Evaluasi Pembangunan Pedesaan).



Gambar 2. 3 Metodologi *Waterfall*

Adapun penjelasan urutan dari tahapan-tahapan yang dimiliki metodologi waterfall adalah sebagai berikut :

1. *Requirement*

Dalam langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau study literatur. Seseorang sistem analisis akan menggali informasi sebanyak-banyaknya dari user sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh user tersebut. Tahapan ini akan menghasilkan dokumen user requirement atau bisa dikatakan sebagai data yang berhubungan dengan keinginan user dalam pembuatan sistem. Dokumen inilah yang akan menjadi acuan system analisis untuk menterjemahkan kedalam bahasa pemrograman.

2. *Design System*

Proses design akan menterjemahkan syarat kebutuhan kesebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat koding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut software requirement. Dokumen inilah yang akan digunakan programmer untuk melakukan aktivitas pembuatan sistemnya.

3. *Coding & Testing (Implementation)*

Coding merupakan penerjemahan design dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan meterjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan computer akan dimaksimalkan

dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi. Tujuan testing adalah menemukan kesalahankesalahan terhadap system tersebut dan kemudian bisa diperbaiki.

4. *Integration & Testing*

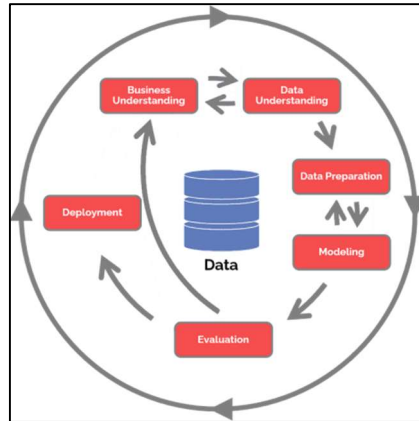
Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, design dan pengkodean maka sistem yang sudah jadi akan digunakan oleh user

5. *Operation & Maintenance*

Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, design dan pengkodean maka sistem yang sudah jadi akan digunakan oleh user

2.5.2 *Cross-Industry Standard Process for Data Mining*

Cross-Industry Standard Process for Data Mining atau CRISP-DM adalah salah satu model proses *data mining* (*data mining framework*) yang awalnya dibangun oleh 5 perusahaan yaitu Integral Solutions Ltd (ISL), Teradata, Daimler AG, NCR Corporation dan OHRA. Metodologi ini terdiri dari enam tahapan yaitu *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation*, dan *Deployment*. Proses metodologi ini terdiri dari 6 tahapan yang dapat dijelaskan sebagai berikut :



Gambar 2. 4 Alur *CRISP-DM*

1. *Business Understanding*

Ini adalah tahap pertama dalam *CRISP-DM* dan termasuk bagian yang cukup vital. Pada tahap ini membutuhkan pengetahuan dari objek bisnis, bagaimana membangun atau mendapatkan data, dan bagaimana untuk mencocokkan tujuan pemodelan untuk tujuan bisnis sehingga model terbaik dapat dibangun. Kegiatan yang dilakukan antara lain: menentukan tujuan dan persyaratan dengan jelas secara keseluruhan, menerjemahkan tujuan tersebut serta menentukan pembatasan dalam perumusan masalah *data mining*, dan selanjutnya mempersiapkan strategi awal untuk mencapai tujuan tersebut. Dalam penelitian ini, penulis menggunakan data kuesioner dari responden mahasiswa usaha

2. *Data Understanding*

Secara garis besar untuk memeriksa data, sehingga dapat mengidentifikasi masalah dalam data. Tahap ini memberikan fondasi analitik untuk sebuah penelitian dengan membuat ringkasan (*summary*) dan mengidentifikasi potensi masalah dalam data. Tahap ini juga harus dilakukan secara cermat dan tidak terburu-buru, seperti pada visualisasi data, yang terkadang *insight*-nya sangat

sulit didapat jika dihubungkan dengan summary data nya. Jika ada masalah pada tahap ini yang belum terjawab, maka akan mengganggu pada tahap *modeling*. Ringkasan atau summary dari data dapat berguna untuk mengkonfirmasi apakah data terdistribusi seperti yang diharapkan, atau mengungkapkan penyimpangan tak terduga yang perlu ditangani pada tahap selanjutnya, yaitu *Data Preparation*. Masalah dalam data biasanya seperti nilai-nilai yang hilang, outlier, berdistribusi spike, berdistribusi bimodal harus diidentifikasi dan diukur sehingga dapat diperbaiki dalam *Data Preparation*.
Data kuesioner yang diperoleh dengan memanfaatkan kuesioner

3. *Data Preparation*

Secara garis besar untuk memperbaiki masalah dalam data, kemudian membuat variabel derived. Tahap ini jelas membutuhkan pemikiran yang cukup matang dan usaha yang cukup tinggi untuk memastikan data tepat untuk algoritma yang digunakan. Bukan berarti saat *Data Preparation* pertama kali dimana masalah- masalah pada data sudah diselesaikan, data sudah dapat digunakan hingga tahap terakhir. Tahap ini merupakan tahap yang sering ditinjau kembali saat menemukan masalah pada saat pembangunan model. Sehingga dilakukan iterasi sampai menemukan hal yang cocok dengan data. Tahap sampling dapat dilakukan disini dan data secara umum dibagi menjadi dua, data training dan data testing. Kegiatan yang dilakukan antara lain: memilih kasus dan parameter yang akan dianalisis (*Select Data*), melakukan transformasi terhadap parameter tertentu (*Transformation*), dan melakukan pembersihan data agar data siap untuk tahap *modeling* (*Cleaning*). Data tweet

yang telah diperoleh kemudian diolah dan kemudian dipersiapkan sebagai data set.

4. *Modeling*

Secara garis besar untuk membuat model prediktif atau deskriptif. Pada tahap ini dilakukan metode statistika dan *Machine Learning* untuk penentuan terhadap teknik *data mining*, alat bantu *data mining*, dan algoritma *data mining* yang akan diterapkan. Lalu selanjutnya adalah melakukan penerapan teknik dan algoritma *data mining* tersebut kepada data dengan bantuan alat bantu. Jika diperlukan penyesuaian data terhadap teknik *data mining* tertentu, dapat kembali ke tahap data *preparation*. Algoritma yang digunakan yaitu SVM dan KNN.

5. *Evaluation*

Melakukan interpretasi terhadap hasil dari *data mining* yang dihasilkan dalam proses pemodelan pada tahap sebelumnya. Evaluasi dilakukan terhadap model yang diterapkan pada tahap sebelumnya dengan tujuan agar model yang ditentukan dapat sesuai dengan tujuan yang ingin dicapai dalam tahap pertama.

6. *Deployment*

Tahap *deployment* atau rencana penggunaan model adalah tahap yang paling dihargai dari proses CRISP-DM. Perencanaan untuk *Deployment* dimulai selama *Business Understanding* dan harus menggabungkan tidak hanya bagaimana untuk menghasilkan nilai model, tetapi juga bagaimana mengkonversi skor keputusan, dan bagaimana untuk menggabungkan keputusan dalam sistem operasional. Pada akhirnya, rencana sistem *Deployment* mengakui bahwa tidak ada model yang statis. Model tersebut

dibangun dari data yang diwakili data pada waktu tertentu, sehingga perubahan waktu dapat menyebabkan berubahnya karakteristik data. Model pun harus dipantau dan mungkin diganti dengan model yang sudah diperbaiki.

2.6 Algoritma Sistem

2.6.1 Algoritma *Naïve Bayes*

Probabilitas klasifikasi yang sangat sederhana yang melakukan perhitungan dengan beberapa probabilitas dengan melakukan sejumlah frekuensi dan menggabungkan nilai-nilai dari dataset yang digunakan adalah metode pengujian *naive bayes*. *Naïve Bayes* mengasumsikan atribut independen atau tidak saling ketergantungan pada nilai variabel setiap kelas. (Patil, 2013). Peneliti lain mengatakan bahwa metode Bayes naif dibuat oleh orang Inggris, Thomas Bayes, yang mengklasifikasikan probabilitas dan statistik dengan memprediksi masa depan dengan melakukannya menggunakan pengalaman sebelumnya (Bustami, 2013).

Naïve bayes tidak diberi nilai *output* dan menyederhanakan nilai atribut independen secara kondisional. Dengan kata lain, dengan mengamati probabilitas probabilitas individu produk. (Ridwan, 2013). Barang yang dihasilkan dengan menggunakan metode $n =$ adalah bahwa metode ini hanya membutuhkan sedikit data dalam melakukan pelatihan data dan dalam menentukan perkiraan parameter yang digunakan dalam proses klasifikasi data. *Naïve Bayes* bekerja sangat baik dalam kehidupan nyata global seperti yang diharapkan.

Teorema bayes digunakan untuk menghitung jumlah probabilitas untuk peristiwa yang memiliki pengaruh pada hasil pengamatan. Dalam Bayesian, parameter digunakan sebagai variabel acak sedangkan di dunia statistik sebelumnya, parameter harus selalu diperbaiki. Pastor Thomas Bayes adalah nama

theorem Bayes yang digambarkan sebagai hubungan antara peluang peristiwa A dan Z, yang dijelaskan dalam rumus berikut (Kundu, 2011):

$$\boxed{P(H | x) = \frac{P(x | H) P(H)}{P(x)}} \quad \text{Atau} \quad \boxed{P(H | x) = \frac{P(x | H) P(H)}{P(x | H)P(H) + P(x | \bar{H})P(\bar{H})}}$$

Keterangan :

X = Data dengan class yg blm diketahui

H = Hipotesis data X merupakan suatu spesifik

$P(H|X)$ = Probabilitas Hipotesis H berdasarkan kondisi X
(posteriori probability)

$P(H)$ = Probabilitas hipotesis H (proir probability)

$P(X|H)$ = Probabilitas X bedasrkan Kondisi H

$P(X)$ = Probabilitas dari X

Dalam kelas sampel data X yang labelnya belum diketahui, dan H adalah hipotesis, data sampel x ditransfer ke kelas khusus c. $P(H / x)$ adalah probabilitas yang menjelaskan data tentang data penelitian x. $P(H / x)$ adalah probabilitas posterior yang menyerupai kepercayaan pada prediksi setelah x diberikan. Sebaliknya, $P(H)$ adalah probabilitas H sebelum sampel digunakan, sebelum sampel terbentuk. Probabilitas posterior $P(H / x)$ didasarkan pada banyak informasi dari probabilitas priori $P(H)$. Teori Bayes memiliki cara menghitung probabilitas posterior $P(H / x)$ menggunakan probabilitas $P(H)$, $P(X)$ dan $P(H / x)$.

Metode Bayes adalah pendekatan statistik untuk inferensi induksi pada masalah klasifikasi. Pertama dibahas pertama tentang konsep dasar dan definisi dalam teori

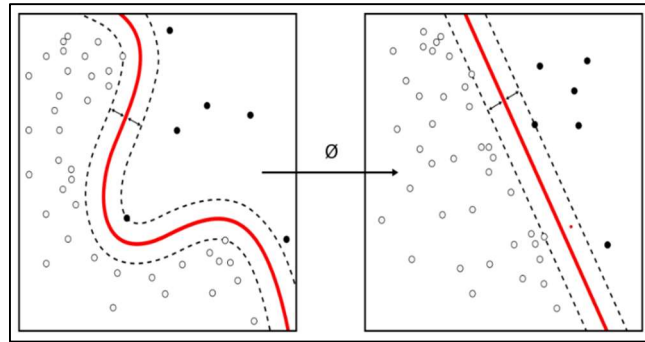
Bayes, kemudian menggunakan teorema ini untuk mengklasifikasikan dalam Penambangan Data. Metode Bayes menggunakan proporsi bersyarat sebagai dasar.

2.6.2 Algoritma *Support Vector Machine*

SVM diciptakan oleh Vapnik pada tahun 1992 menggunakan serangkaian konsep unggul yang baik di bidang pengenalan pola. SVM masih dianggap muda dalam metode pengenalan pola. Namun, kemampuannya dalam berbagai aplikasi sering digunakan sebagai seni dalam membuat pola. SVM juga merupakan metode mesin pembelajaran yang bekerja berdasarkan prinsip *Structural Risk Minimization* (SRM) yang berfungsi sebagai *hyperplane* terbaik yang memisahkan *input*.

SVM memiliki fitur multi-dimensi yang memiliki plot sebagai titik data dalam mengklasifikasikan dengan mendefinisikan batas antara titik data dari permukaan. SVM bertujuan untuk membuat garis bawah atau disebut hyperlink dengan arah partisi data yang sama satu sama lain. Dengan cara ini, pembelajaran SVM menggabungkan aspek-aspek pembelajaran dari tetangga terdekat. Untuk masalah klasifikasi biner, SVM sangat cocok.

Support Vector Machine dapat dibayangkan sebagai permukaan yang mendefinisikan batas antara berbagai titik data yang mewakili contoh yang diplot dalam ruang multidimensi sesuai dengan fiturnya. Tujuan dari SVM adalah untuk membuat batas dasar atau yang disebut dengan hyperline yang mengarah pada partisi data yang homogen di antara kedua sisi. Dengan cara ini, pembelajaran SVM yang menggabungkan aspek-aspek dari pembelajaran tetangga terdekat. Untuk permasalahan klasifikasi biner, SVM sangat cocok digunakan. Sebagai contoh pada gambar berikut ini.



Gambar 2. 5 *Support Vector Machine*

2.6.3 *Confusion Matrix*

Menurut Harani, N. H., & Hasanah, M. (2020) dalam buku yang berjudul “Deteksi Objek dan Pengenalan Plat Nomor Kendaraan Indonesia Berbasis Python” mendiskusikan bahwa “*Confusion Matrix* adalah salah satu metode yang dapat digunakan untuk mengukur kinerja suatu metode klasifikasi dan pada dasarnya *Confusion Matrix* mengandung informasi yang dapat membandingkan hasil klasifikasi yang seharusnya.”.

Kelas	Ter klasifikasi Positif	Ter klasifikasi Negatif
Positif	TP (True Positive)	FN (False Negative)
Negatif	FP (False Positive)	TN (True Negative)

Gambar 2. 6 *Confusion Matrix*

Berdasarkan nilai False Positive (FP), False Negative (FN), True Negative (TN), dan True Positive (TP) dapat diperoleh nilai akurasi, presisi dan recall. Nilai dari akurasi menggambarkan seberapa akurat/cermat suatu sistem dapat mengklasifikasikan data secara benar. Dengan kata lain, nilai akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data.

Pada tahap ini akan dijelaskan bagaimana nilai *Confusion Matrix* didapat berdasarkan penggunaan rumus-rumus yang ada di bawah ini, sebagai berikut.

$$Accuracy = (TP + TN) / (TP + FP + TN + FN)$$

$$Precision = TP / (TP + FP)$$

$$Recall = TP / (TP + FN)$$







$$F1\ Score = (2 * Precision * Recall) / (Precision + Recall)$$

2.7 Pemodelan UML

Untuk lebih menjelaskan perancangan aplikasi yang dibangun, digunakan 4 (empat) model diagram UML, yaitu : *use case diagram*, *Activity Diagram*, *sequence diagram*, *Deployment Diagram*.

2.7.1 Use Case

Tabel 2. 1 Simbol *Use Case Diagram* (Yunahar, 2018)

Simbol	Deskripsi
	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar antar unit atau aktor biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i>
	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama aktor
	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan
	Hubungan generalisasi dan sepealisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya
	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini





Dalam jurnal yang berjudul “Perancangan Sistem Aplikasi Pemesanan Makanan dan Minuman Cafeteria No Cafe di Tanjung Balai Karimun

Menggunakan Bahasa Pemrograman PHP dan MySQL” dijelaskan bahwa Use Case Diagram merupakan diagram yang bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan suatu sistem tersendiri melalui sebuah cerita bagaimana sebuah sistem dipakai (T. Bayu Kurniawan dan Syarifuddin, 2020). Use case diagram terdiri dari aktor dan interaksi yang dilakukannya, aktor tersebut dapat berupa manusia, perangkat keras, sistem lain, atau mereka yang berinteraksi dengan sistem.

Pada aplikasi pencarian jalur terpendek antar kota menggunakan algoritma genetika, use case menggambarkan hubungan antara sistem dan aktor. Hubungan ini dapat berupa *input* aktor ke sistem.

2.7.2 Activity Diagram




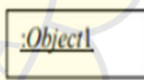

Tabel 2. 2 Simbol *Activity Diagram* (Yunahar, 2018)

Simbol	Deskripsi
	Simbol <i>start</i> untuk menyatakan awal dari suatu proses
	Simbol <i>stop</i> untuk menyatakan akhir dari suatu proses
	Simbol <i>desticion</i> digunakan untuk menyatakan kondisi dari suatu proses
	Simbol <i>action</i> menyatakan aksi yang dilakukan dalam suatu arsitektur sistem

Activity Diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas dapat dilakukan oleh sistem” (Yunahar, 2018).

2.7.3 Sequence Diagram

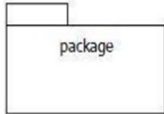
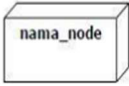


Tabel 2. 3 Simbol *Sequence Diagram* (Yunahar, 2018)

Nama Komponen	Keterangan	Simbol
<i>Lifeline</i>	Mengindikasikan keberadaan sebuah objek dalam basis waktu. Notasi untuk <i>lifeline</i> adalah garis putus-putus vertikal yang ditarik dari sebuah objek	
<i>Activation</i>	Dinotasikan sebagai sebuah kotak segi empat digambar pada sebuah lifeline mengindikasikan sebuah objek yang akan melakukan sebuah aksi	
<i>Message</i>	Digambarkan dengan anak panah horizontal antara activation. Message mengindikasikan komunikasi antara objek-objek	
<i>Object</i>	Merupakan <i>instance</i> dari sebuah class dan dituliskan tersusun secara horizontal.	
<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi	

“*Sequence Diagram* adalah tool yang sangat populer dalam pengembangan sistem informasi secara object-oriented untuk menampilkan interaksi antar objek. Berdasarkan definisi tersebut, dapat disimpulkan bahwa *Sequence Diagram* adalah tool yang digunakan dalam pengembangan sistem” (Yunahar, 2018).

2.7.4 Deployment Diagram

Tabel 2. 4 Simbol *Deployment Diagram* (Rosa dan M.Shalahuddin, 2016)

Simbol	Deskripsi
Package 	package merupakan sebuah bungkusan dari satu atau lebih <i>node</i>
Node 	biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen
Kebergantungan / dependency 	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai
Link 	relasi antar <i>node</i>

(Sukanto dan Shalahudin, 2015) mengemukakan bahwa “*Deployment Diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi”. *Deployment Diagram* dapat digunakan untuk menggambarkan hal-hal sebagai berikut, yaitu:

1. Sistem tambahan (*embedded system*) yang menggambarkan rancangan device, *node* dan *hardware*.
2. Sistem client atau server.
3. Sistem terdistribusi murni.
4. Rekayasa ulang aplikasi.

Dapat disimpulkan bahwa *Deployment Diagram* merupakan diagram yang digunakan untuk menggambarkan konfigurasi pada *system*.