

BAB II

LANDASAN TEORI

2.1 Data Mining

Menurut (Ermanto, Damas Mahardi 2020) dalam jurnal yang berjudul “Penerapan *Data Mining* Untuk Klasifikasi Pandemi COVID-19 Di Indonesia Dengan Algoritma *Naive Bayes*”. *Data mining* adalah salah satu bentuk implementasi yang diterapkan untuk mencari sebuah model dan pola yang mampu melakukan prediksi pada suatu data berdasarkan data sebelumnya di periode waktu tertentu. *Data mining* adalah bentuk penggalian data yang digunakan untuk menggali pengetahuan dari jumlah data yang besar.

2.2 Fungsi Data Mining

Serta fungsi *data mining* menurut (Rinhto Rare Rerung 2018) dalam jurnal yang berjudul “Penerapan *Data Mining* dengan Memanfaatkan *Association Rule* untuk Promosi Produk” *Data mining* merupakan teknologi yang sangat berguna untuk membantu perusahaan menemukan informasi yang sangat penting dari database mereka yang selama ini hanya digunakan sebagai arsip perusahaan saja.

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas/pekerjaan yang dapat dilakukan yaitu:

- 1) Deskripsi: terkadang peneliti dan analisis secara sederhana ingin mencoba mencari cara untuk menggambarkan pola dan kecenderungan yang terdapat dalam data.

Deskripsi dari pola kecenderungan sering memberikan kemungkinan penjelasan untuk suatu pola atau kecenderungan

2) Estimasi: hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih ke arah numerik dari pada ke arah kategori. Model dibangun menggunakan baris data (record) lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi. Selanjutnya, pada peninjauan berikutnya estimasi nilai dari variabel target dibuat berdasarkan nilai variabel prediksi.

3) Prediksi: hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada di masa mendatang. Beberapa metode dan teknik yang digunakan dalam klasifikasi dan estimasi dapat pula digunakan (untuk keadaan yang tepat) untuk prediksi.

4) Klasifikasi: terdapat target variabel kategori. Sebagai contoh, penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu pendapatan tinggi, pendapatan sedang, dan pendapatan rendah.

5) Pengklasteran: merupakan pengelompokan record, pengamatan, atau memperhatikan dan membentuk kelas obyek-obyek yang memiliki kemiripan. Klaster adalah kumpulan record yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan record dalam klaster yang lain. Berbeda dengan klasifikasi, pada pengklasteran tidak ada variabel target. Pengklasteran tidak melakukan klasifikasi, mengestimasi, atau memprediksi nilai dari variabel target, akan tetapi, algoritma pengklasteran mencoba untuk melakukan pembagian terhadap keseluruhan data menjadi kelompok-kelompok yang memiliki kemiripan (homogen), yang mana kemiripan record

dalam satu kelompok akan bernilai maksimal, sedangkan kemiripan dengan record dalam kelompok lain akan bernilai minimal.

6) Asosiasi: tugas asosiasi dalam data mining adalah untuk menemukan atribut yang muncul dalam satu waktu. Salah satu implementasi dari asosiasi adalah market basket analysis sebagaimana yang akan dibahas dalam penelitian ini. Dalam bidang keilmuan data mining, terdapat suatu metode yang dinamakan association rule. Metode ini sering juga dinamakan dengan market basket analysis. Association rule mining adalah suatu prosedur untuk mencari hubungan antar item dalam suatu data set yang ditentukan.

2.3 Teknik Pembelajaran Data Mining

Teknik Pembelajaran Data Mining menurut (Juna Eska 2016) dalam jurnal yang berjudul “Penerapan Data Mining untuk Prediksi Penjualan Wallpaper menggunakan Algoritma C4.5” Penerapan data mining dalam sistem berbantuan komputer diawali dengan pengumpulan data, yang dilanjutkan dengan transformasi data, dan diakhiri dengan analisis data. Pada pengumpulan data, harus didefinisikan suatu model interaksi user dengan sistem untuk menetapkan data yang harus dicatat dari suatu proses pembelajaran. Proses transformasi data mengubah data mentah menjadi dataset yang siap dianalisis. Transformasi dapat dilakukan pada instan dataset melalui proses filtrasi, maupun pada atribut dari dataset melalui filtrasi ataupun konversi.

Beberapa teknik dan sifat data mining adalah sebagai berikut:

- a. Classification [Predictive] b. Clustering [Descriptive]
- b. AssociationRule Discovery [Descriptive]

- c. Regression [Predictive]
- d. DeviationDetection [Predictive]

Karena data mining adalah sebuah untain proses, maka pecah menjadi beberapa tahap. Tahapan tersebut akan bersifat interaktif, pengguna akan terlibat langsung atau dengan perantara KDD.

Tahapan data mining dibagi menjadi enam bagian yaitu :

1. Pembersihan data (data cleaning) Sebelum proses data mining dapat dilaksanakan, perlu dilakukan proses cleaning pada data yang menjadi fokus KDD. Proses cleaning mencakup antara lain membuang 4 duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (tipografi). Juga dilakukan proses enrichment, yaitu proses “memperkaya” data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.
2. Integrasi data (data integration) Integrasi data merupakan penggabungan data dari berbagai database ke dalam satu database baru. Tidak jarang data yang diperlukan untuk data mining tidak hanya berasal dari satu database tetapi juga berasal dari beberapa database atau file teks. Integrasi data dilakukan pada atribut-atribut yang mengidentifikasi entitas-entitas yang unik seperti atribut nama, jenis produk, nomor pelanggan lainnya. Integrasi data perlu dilakukan secara cermat karena kesalahan pada integrasi data bisa menghasilkan hasil yang menyimpang dan bahkan menyesatkan pengambilan aksi nantinya. Sebagai contoh bila integrasi data berdasarkan jenis produk

ternyata menggabungkan produk dari kategori yang berbeda maka akan didapatkan korelasi antar produk yang sebenarnya tidak ada.

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

4. Transformasi data (Data Transformation) Data diubah atau digabung ke dalam format yang sesuai untuk diproses dalam data mining. Beberapa metode data mining membutuhkan format data yang khusus sebelum bisa diaplikasikan. Sebagai contoh beberapa metode standar seperti analisis asosiasi dan clustering hanya bisa menerima input data kategorikal. Karenanya data berupa angka numerik yang berlanjut perlu dibagi menjadi beberapa interval. Proses ini sering disebut transformasi data.

5. Proses mining adalah sebuah proses yang paling utama pada saat metode diterapkan untuk mencari pengetahuan tersembunyi dan berharga dari data.

6. Evaluasi pola (pattern evaluation), Untuk mengidentifikasi pola-pola menarik kedalam knowledge based yang ditemukan. Dalam tahap ini hasil dari teknik data mining berupa pola-pola yang khas maupun model prediksi dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai.

7. Presentasi pengetahuan (knowledge presentation), Merupakan penyajian dan visualisasi pengetahuan mengenai metode yang digunakan untuk memperoleh pengetahuan yang diperoleh pengguna. Tahap terakhir dari proses data mining adalah bagaimana memformulasikan keputusan atau aksi dari hasil analisis yang didapat. Ada

kalanya hal ini harus melibatkan orang-orang yang tidak memahami data mining. Karenanya presentasi hasil data mining dalam bentuk pengetahuan yang bisa dipahami semua orang adalah satu tahapan yang diperlukan dalam proses data mining. Dalam presentasi ini, visualisasi juga bisa membantu mengkomunikasikan hasil data mining.

2.4 Pemrograman Aplikasi

2.4.1 HTML

HTML menurut . (Selli Mariko 2019) dalam jurnal yang berjudul “Aplikasi Website Berbasis HTML Dan Javascript Untuk Menyelesaikan Fungsi Integral Pada Mata Kuliah Kalkulus” Hypertext Markup Language (HTML) adalah bahasa standar yang digunakan untuk menampilkan konten pada halaman website. Fungsi-fungsi yang dapat dilakukan dengan bahasa programan HTML adalah:

- (1) Mengatur serta mendesain tampilan isi halaman website, Membuat tabel pada halaman website,
- (3) Mempublikasikan halaman website secara online, Membuat form yang dapat menjadi input serta menangani registrasi dan transaksi via website, Menampilkan area gambar pada browser.

Penanda perintah pada bahasa pemrograman Hypertext Markup Language (HTML) disebut dengan tag. Tag digunakan untuk menentukan tampilan dari dokumen HTML. Tag HTML berfungsi untuk mendefinisika bahwa isi dalam file tersebut adalah dokumen. Element head merupakan kepala dari dokumen HTML. Elemen head digunakan untuk menempatkan identitas file, sedangkan tag body digunakan untuk menentukan konten yang akan ditampilkan pada halaman website.

2.4.2 CSS

CSS menurut (Marlina, Masnur, Muh.Dirga.F 2021) dalam jurnal yang berjudul “Aplikasi E-Learning Siswa SMK Berbasis Web” mengatakan bahwa CSS atau singkatan dari Cascading Style Sheet adalah suatu aturan untuk mengatur tampilan dari website sehingga tampilan dalam web lebih terstruktur. CSS sendiri bukanlah bahasa pemrograman, CSS lebih seperti konfigurasi tampilan dari suatu tag pada website. CSS dapat merubah text, warna, background dan posisi dari suatu tag.

2.4.3 JavaScript

Javascript menurut (Ahmad Sahi 2020) dalam jurnal yang berjudul “Aplikasi Test Potensi Akademik Seleksi Saringan Masuk LP31 Berbasis Web Online Menggunakan Framework Codeigniter” mengatakan bahwa Javascript diperkenalkan pertama kali oleh Netscape pada tahun 1995. Pada awalnya bahasa ini dinamakan “LiveScript” yang berfungsi sebagai bahasa sederhana untuk browser Netscape Navigator 2. Javascript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengizinkan pengeksekusian perintah perintah di sisi user, yang artinya di sisi browser bukan di sisi server web. Javascript bergantung kepada browser (navigator) yang memanggil halaman web yang berisi skrip-skrip dari Javascript dan tentu saja terselip di dalam dokumen HTML.

JavaScript adalah bahasa pemrograman berbentuk kumpulan script yang berjalan pada suatu dokumen HTMLJavaScript dapat menyempurnakan tampilan dan sistem pada

halaman web-based application yang dikembangkan. Adapun karakteristik dari bahasa pemrograman JavaScript adalah:

- 1) Bahasa pemrograman berjenis high-level programming;
- 2) Bersifat client-side;
- 3) Berorientasi pada objek, dan
- 4) Bersifat loosely typed.

Aplikasi atau tools standar yang digunakan dalam mengimplementasikan JavaScript adalah:

- a. Software text-editor: seperti Notepad++, Adobe Dreamweaver dan NetBeans;
- b. Web browser, seperti: Mozilla Firefox, Google Chrome, Internet Explorer, OperaMini, Safari, dan lain sebagainya;
- c. HTML. Kelemahan dari bahasa pemrograman JavaScript ini sendiri adalah JavaScript tidak didukung oleh browser versi lama (Enterprise, 2017, pp. 1-3). (Selli Mariko 2019).

2.4.4 PHP

Menurut (Ahmad Sahi 2020) juga mengatakan bahwa PHP merupakan singkatan dari PHP Hypertext Preprocessor yang digunakan sebagai bahasa script server-side dalam pengembangan web yang disisipkan pada dokumen HTML. PHP merupakan software open source yang disebar dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya <http://www.php.net>.

Kelebihan dari PHP, yaitu :

- a. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. Web server yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, nginx, hingga Xitami dengan konfigurasi lebih mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
- e. PHP adalah bahasa open source yang dapat digunakan di beberapa mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.

2.4.5 Bootstrap

Bootstrap menurut (Indri Handayani, Giandari Maulani, Evira Noviandri, Hesti Widya Ningsih 2020) dalam jurnal yang berjudul “Optimilasi Sistem Pengelolaan E-Journal Berbasis Open Journal System (OJS) Menggunakan Framework CSS Bootstrap Pada Instansi dan Asosiasi” mengatakan bahwa Bootstrap adalah framework ataupun tools untuk membuat aplikasi web ataupun website yang bersifat responsive secara cepat, mudah dan gratis. Kata responsive disini berarti bahwa tampilan web (lebar dan susunan isinya dapat berubah secara otomatis sesuai dengan lebar layar yang menampilkannya). Maka dari itu untuk membuat tampilannya lebih menarik dioptimalkan dengan menggunakan OJS versi 3.1.0.1 untuk mendukung proses pengelolaan jurnal menjadi lebih efisien dan flexible serta untuk mengembangkan sebuah sistem agar memiliki

tampilan yang lebih menarik dan mudah di kustomisasi menggunakan tema Framework CSS Bootstrap.

2.4.6 Database

Database menurut (Nurhadi, Slamet, dkk. 2019) dalam jurnal yang berjudul “Sistem Basis Data: Mysql di Ubuntu” mengatakan bahwa Sistem Basis Data adalah suatu sistem menyusun dan mengelola record-record menggunakan computer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan. Berdasarkan pengertian tersebut, dapat disimpulkan bahwa Sistem Basis Data merupakan penggunaan basis data yang dimana sistem dapat menyusun dan mengola record secara komputerisasi yang dapat memudahkan sebuah perusahaan/organisasi untuk proses pengambilan keputusan.

2.4.7 MySQL

Menurut (Nurhadi, Slamet, dkk. 2019) juga mengatakan bahwa MySQL merupakan singkatan dari Structured Query Language. SQL merupakan bahasa terstruktur yang khusus digunakan untuk mengolah database. MySQL merupakan sistem manajemen database yang bersifat relational. Artinya, data yang dikelola dalam database akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan jauh lebih cepat.

2.5 Metodologi Algoritma Sistem

2.5.1 Algoritma C4.5

Pengertian Algoritma C4.5 menurut (Wira Bagye. 2020) dalam jurnal yang berjudul “Jurnal Misi (Jurnal Manajemen Informatika dan Sistem Informasi)” dijelaskan bahwa Algoritma C4.5 dipahami sebagai sebuah algoritma yang berfungsi untuk mengembangkan pohon keputusan. Konsep berpikir pohon keputusan dimulai dari atribut paling atas yang disebut sebagai akar (root) kemudian mengalir melalui cabang-cabang di bawahnya sehingga berakhir di atribut paling bawah yang disebut sebagai daun (leaf). Dalam membangun pohon, algoritma C4.5 mempunyai urutan langkah sebagai berikut:

1. Melakukan perhitungan jumlah data, dengan dasar anggota atribut hasil yang mempunyai syarat yang telah ditentukan terlebih dahulu. Pada proses pertama, syarat dalam kondisi kosong.
2. Memilih atribut yang akan digunakan node.
3. Membuat cabang pada setiap anggota yang dimiliki node.
4. Memeriksa nilai entropy yang dimiliki oleh anggota node. Jika setelah diperiksa terdapat anggota node yang nilai entropynya berisi nol, maka akan ditentukan daun yang muncul. Apabila ketika diperiksa, ditemukan kondisi dimana semua nilai entropy anggota berisi nol, maka rangkaian proses akan dihentikan.
5. Apabila terdapat anggota node yang entropynya bernilai lebih dari nol, maka akan mengulangi proses dari posisi awal, dengan node yang digunakan sebagai syarat, sampai seluruh anggota node berisi nol.

Node adalah sebuah atribut dimana nilai gain yang dimilikinya dihitung sebagai nilai tertinggi jika dibandingkan dengan atribut-atribut yang lain. Dalam proses perhitungan nilai gain pada sebuah atribut, maka akan digunakan persamaan sebagai berikut:

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

Keterangan:

S : himpunan kasus

A : atribut

n : jumlah partisi atribut A

|S_i| : jumlah kasus pada partisi ke-i

|S| : jumlah kasus dalam S

Sedangkan dalam melakukan perhitungan nilai entropy akan menggunakan persamaan berikut:

$$\text{Entropy}(S) = \sum_{i=1}^n -p_i * \log_2 p_i$$

Keterangan:

S : himpunan kasus

A : atribut

n : jumlah partisi S

P_i : proporsi dari S_i terhadap S

2.5.2 Algoritma Naive Bayes Classifier

Pengertian Algoritma Naive Bayes Classifier menurut (Alvina Felicia Watratan, Arwini Puspita. B, Dikwan Mocis,2020) dalam jurnal yang berjudul “Implementasi Algoritma Naive Bayes Untuk Memprediksi Tingkat Penyebaran Covid-19 Di Indonesia” dijelaskan bahwa Naive Bayes Classifier merupakan sebuah metoda klasifikasi yang berakar pada teorema Bayes. Metode pengklasifikasian dengan menggunakan 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.

Ciri utama dari Naive Bayes Classifier ini adalah asumsi yang sangat kuat (naif) akan independensi dari masing-masing kondisi / kejadian.

Keuntungan penggunaan adalah bahwa metoda ini hanya membutuhkan jumlah data pelatihan (training data) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian. Karena yang diasumsikan sebagai variable independent, maka hanya varians dari suatu variable dalam sebuah kelas yang dibutuhkan untuk menentukan klasifikasi, bukan keseluruhan dari matriks kovarians.

a. Kegunaan Naive Bayes :

1. Mengklasifikasikan dokumen teks seperti teks berita ataupun teks akademis
2. Sebagai metode machine learning yang menggunakan probabilitas
3. Untuk membuat diagnosis medis secara otomatis
4. Mendeteksi atau menyaring spam

b. Kelebihan Naive Bayes :

1. Bisa dipakai untuk data kuantitatif maupun kualitatif

2. Tidak memerlukan jumlah data yang banyak
 3. Tidak perlu melakukan data training yang banyak
 4. Jika ada nilai yang hilang, maka bisa diabaikan dalam perhitungan.
 5. Perhitungannya cepat dan efisien
 6. Mudah dipahami
 7. Mudah dibuat
 8. Pengklasifikasian dokumen bisa dipersonalisasi, disesuaikan dengan kebutuhan setiap orang
 9. Jika digunakan dalam bahasa pemrograman, code-nya sederhana
 10. Bisa digunakan untuk klasifikasi masalah biner ataupun multiclass
- c. Kekurangan Naive Bayes :
1. Apabila probabilitas kondisionalnya bernilai nol, maka probabilitas prediksi juga akan bernilai nol
 2. Asumsi bahwa masing-masing variabel independen membuat berkurangnya akurasi, karena biasanya ada korelasi antara variabel yang satu dengan variabel yang lain
 3. Keakuratannya tidak bisa diukur menggunakan satu probabilitas saja. Butuh bukti-bukti lain untuk membuktikannya.
 4. Untuk membuat keputusan, diperlukan pengetahuan awal atau pengetahuan mengenai masa sebelumnya. Keberhasilannya sangat bergantung pada pengetahuan awal tersebut Banyak celah yang bisa mengurangi efektivitasnya dirancang untuk mendeteksi katakata saja, tidak bisa berupa gambar.

Persamaan Metode Naive Bayes

Persamaan dari teorema Bayes adalah :

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

Di mana :

X : Data dengan class yang belum diketahui.

H : Hipotesis data merupakan suatu class spesifik.

$P(H|X)$: Probabilitas hipotesis H berdasarkan 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

Untuk menjelaskan metode Naive Bayes, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut.

Karena itu, metode Naive Bayes di atas disesuaikan:

$$P(C|F_1 \dots F_n) = \frac{P(C)P(F_1 \dots F_n|C)}{P(F_1 \dots F_n)}$$

Di mana Variabel C merepresentasikan kelas, sementara variabel $F_1 \dots F_n$ merepresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi.

Maka rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas C (Posterior) adalah peluang munculnya kelas C (sebelum masuknya sampel tersebut, seringkali disebut prior), dikali dengan peluang kemunculan karakteristik-karakteristik sampel pada kelas C (disebut juga likelihood), dibagi dengan peluang kemunculan karakteristik-karakteristik sampel secara global (disebut juga evidence).

Karena itu, rumus di atas dapat pula ditulis secara sederhana sebagai berikut:

$$\text{Posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

Nilai Evidence selalu tetap untuk setiap kelas pada satu sampel. Nilai dari posterior tersebut nantinya akan dibandingkan dengan nilai-nilai posterior kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan. Penjabaran lebih lanjut rumus Bayes tersebut dilakukan dengan menjabarkan $(C|F_1, \dots, F_n)$ menggunakan aturan perkalian sebagai berikut:

$$\begin{aligned} (C|F_1, \dots, F_n) &= P(C)P(F_1, \dots, F_n|C) \\ &= (C|F_1)P(F_2, \dots, F_n|C, F_1) \\ &= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3, \dots, F_n|C, F_1, F_2) \\ &= (C|F_1)P(F_2|C, F_1)P(F_3|C, F_1, F_2)P(F_4, \dots, F_n|C, F_1, F_2, F_3) \\ &= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1, F_2) \dots P(F_n|C, F_1, F_2, F_3, \dots, F_{n-1}) \end{aligned}$$

Dapat dilihat bahwa hasil penjabaran tersebut menyebabkan semakin banyak dan semakin kompleksnya faktor - faktor syarat yang mempengaruhi nilai probabilitas, yang hampir mustahil untuk dianalisa satu persatu. Akibatnya, perhitungan tersebut menjadi sulit untuk dilakukan. Disinilah digunakan asumsi independensi yang sangat tinggi (naif), bahwa masing-masing petunjuk (F_1, F_2, \dots, F_n) saling bebas (independen) satu sama lain.

Dengan asumsi tersebut, maka berlaku suatu persamaan :

$$P(F_i|F_j) = \frac{P(F_i \cap F_j)}{P(F_j)} = \frac{P(F_i)P(F_j)}{P(F_j)} = P(F_i)$$

Untuk sehingga

$$P(F_i|C, F_j) = P(F_i|C)$$

Persamaan di atas merupakan model dari teorema Naive Bayes yang selanjutnya akan digunakan dalam proses klasifikasi.

Untuk klasifikasi dengan data kontinyu digunakan rumus Densitas Gauss :

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

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

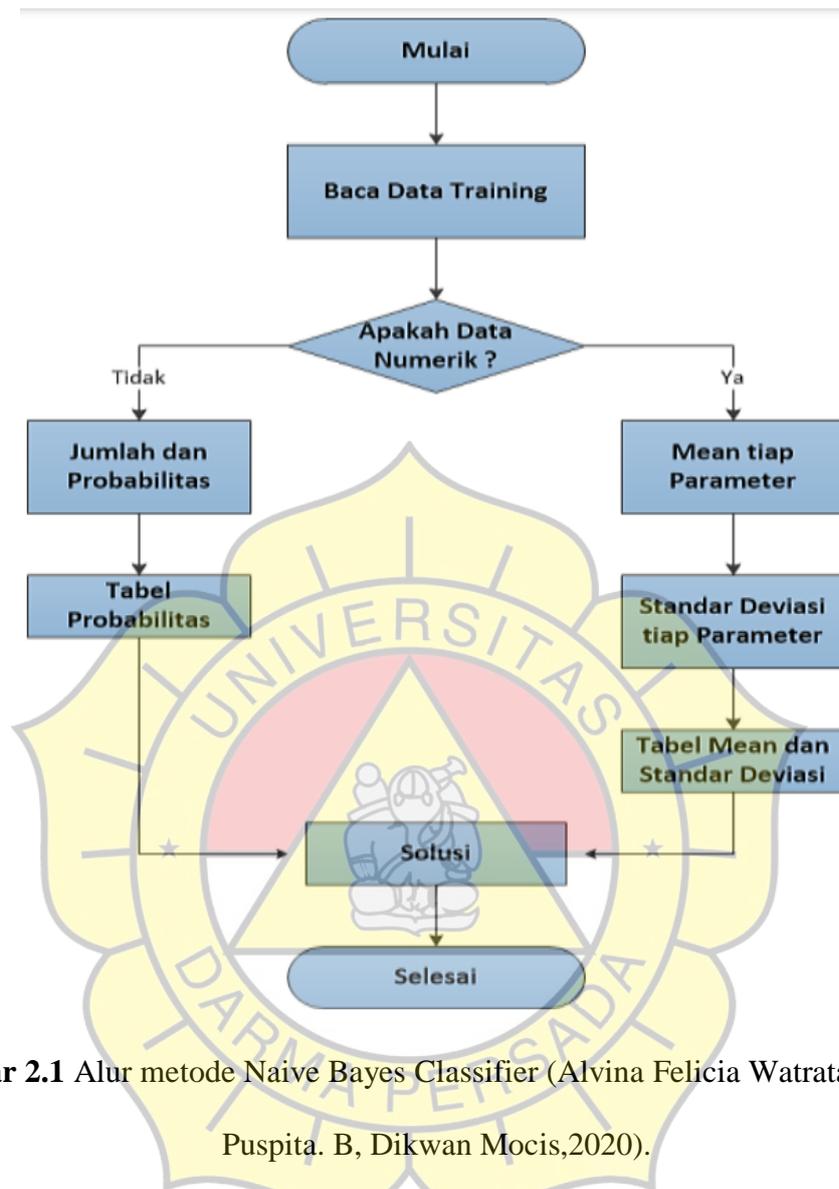
μ : mean, menyatakan rata – rata dari seluruh atribut

σ : Deviasi standar, menyatakan varian dari seluruh atribut.

Alur dari metode Naive Bayes dapat dilihat pada gambar 3.

Adapun keterangan dari gambar 3 sebagai berikut:

1. Baca data training
2. Hitung Jumlah dan probabilitas, namun apabila data numerik maka:
 - a. Cari nilai mean dan standar deviasi dari masingmasing parameter yang merupakan data numerik. Adapun persamaan yang digunakan untuk menghitung nilai rata – rata hitung (mean) dapat dilihat sebagai berikut :



Gambar 2.1 Alur metode Naive Bayes Classifier (Alvina Felicia Watratan, Arwini Puspita. B, Dikwan Mocis,2020).

$$\mu = \frac{\sum_{i=1}^n x_i}{n}$$

Atau

$$\mu = \frac{x_1 + x_2 + x_3 + \dots + x_n}{n}$$

di mana :

μ : rata – rata hitung (mean)

x_i : nilai sample ke -i

n : jumlah sampel

Dan persamaan untuk menghitung nilai simpangan baku (standar deviasi) dapat dilihat sebagai berikut:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{n - 1}}$$

di mana :

σ : standar deviasi

x_i : nilai x ke -i

μ : rata-rata hitung

n : jumlah sampel

- b. Cari nilai probabilitik dengan cara menghitung jumlah data yang sesuai dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut.
3. Mendapatkan nilai dalam tabel mean, standard deviasi dan probabilitas.
4. Solusi kemudian dihasilkan.

2.6 Pemodelan UML

Untuk lebih menjelaskan perancangan aplikasi yang dibangun, digunakan 5 (lima) model diagram UML, yaitu : use case diagram, activity diagram, sequence diagram, class diagram dan deployment diagram.

2.6.1 Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor

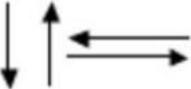
Gambar 2.2 Use Case Diagram (Sumber: Rival Andaputra, 2018)

Menurut (T. Bayu Kurniawan, Syarifuddin 2020) 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. Use case diagram terdiri dari sebuah aktor dan interaksi yang dilakukannya, aktor tersebut dapat berupa manusia, perangkat keras, sistem lain, ataupun yang berinteraksi dengan sistem.

Pada aplikasi pencarian jalur terpendek antar kota menggunakan algoritma genetika, use case menjelaskan tentang hubungan antara sistem dengan aktor. Hubungan ini dapat berupa inputaktor ke sistem ataupun output ke aktor. Use case merupakan dokumen naratif yang mendeskripsikan kasus-kasus atau kejadian-kejadian daripada aktor dalam menggunakan sistem untuk menyelesaikan sebuah proses.

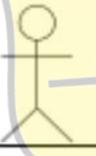
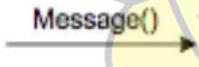
2.6.2 Activity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		Action	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		Initial Node	Bagaimana objek dibentuk atau diawali.
4		Activity Final Node	Bagaimana objek dibentuk dan diakhiri
5		Decision	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
6		Line Connector	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

Gambar 2.3 Activity Diagram (Sumber: Rival Andaputra, 2018)

(T. Bayu Kurniawan, Syarifuddin 2020) juga dijelaskan bahwa Activity diagram menggambarkan berbagai aliran aktivitas dalam sistem yang sedang di rancang, bagaimana masing-masing aliran berawal, decision yang mungkin terjadi dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

2.6.3 Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		LifeLine	Objek / entity, antarmuka yang saling berinteraksi.
		Actor	Digunakan untuk menggambarkan user / pengguna.
2		Message	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		Boundary	Digunakan untuk menggambarkan sebuah form.
4		Control Class	Digunakan untuk menghubungkan boundary dengan tabel.
5		Entity Clas	Digunakan untuk menggambarkan hubungan kegiatan yang akan dilakukan.

Gambar 2.4 Sequence Diagram (Sumber: Rival Andaputra, 2018)

Menurut (T. Bayu Kurniawan, Syarifuddin 2020) juga dijelaskan bahwa Sequence diagram digunakan untuk menggambarkan interaksi antar objek di dalam dan di

sekitar sistem yang berupa message yang digambarkan terhadap waktu. Sequence diagram terdiri antara dimensi vertical (waktu) dan dimensi horizontal (objek-objek yang terkait).

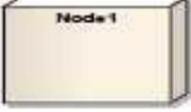
2.6.4 Class Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Class</i>	Himpunan dari objek-objek yang berbagai atribut serta operasi yang sama.

Gambar 2.5 Class Diagram (Sumber: Fransiska, 2019)

Menurut (Winda Aprianti, Umi Maliha. 2016) dalam jurnal yang berjudul “Sistem Informasi Kepadatan Penduduk Kelurahan Atau Desa Studi Kasus Pada kecamatan Bati-Bati Kabupaten Tanah Laut” dijelaskan bahwa Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan di buat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

2.6.5 Deployment Diagram

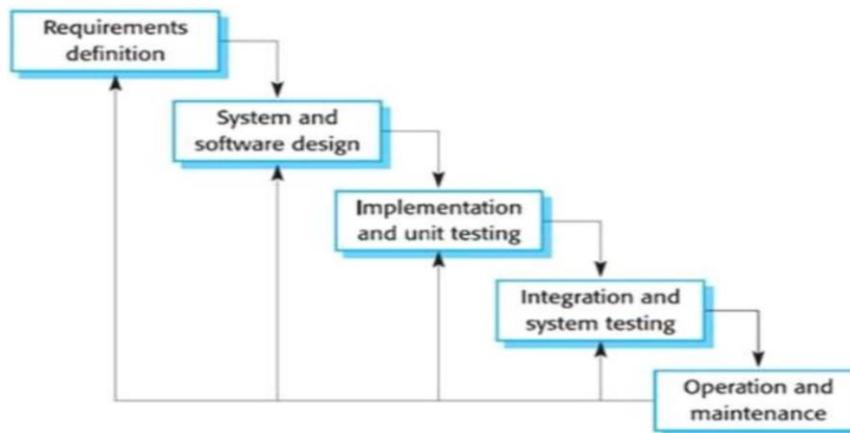
Simbol	Deskripsi
 Package	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih <i>node</i>
 Node	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan.
 Kebergantungan/ <i>dependency</i>	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang di pakai.
 Link	Relasi antar <i>node</i> .

Gambar 2.6 Deployment Diagram (Sumber: Syafira Ayushaleha, 2017)

Menurut (Kundang Karsono 2016) dalam jurnal yang berjudul “Pembangunan Aplikasi Penyewaan Apartemen Berbasis N-Tier dengan Mobile Device” dijelaskan bahwa Deployment diagram adalah penggambaran tugas-tugas kongkrit dari setiap node/ software yang terlibat dalam jaringan sistem, menampilkan keseluruhan node dalam jaringan serta hubungan dari node-node tersebut termasuk proses-proses yang terlibat di dalamnya.

2.7 Metodologi pengembangan Sistem (Waterfall)

Metode penelitian yang diterapkan pada penelitian ini adalah dengan pengembangan metode waterfall. Metode waterfall merupakan model pengembangan sistem informasi yang terstruktur dan berurutan.



Gambar 2.7 Bagan metode waterfall (Sumber: Ginanjar Wiro Sasmito,2017).

Menurut (Ginanjar Wiro Sasmito,2017) dalam jurnal yang berjudul “Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal” dijelaskan bahwa Metode Waterfall memiliki tahapan-tahapan sebagai berikut:

2.7.1 Requirements analysis and definition

Layanan sistem, kendala, dan tujuan ditetapkan oleh hasil konsultasi dengan pengguna yang kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem.

2.7.2 System and software design

Tahapan perancangan sistem mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat lunak dan hubungannya.

2.7.3 Implementation and unit testing

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

2.7.4 Integration and system testing

Unit-unit individu program atau program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak. Setelah pengujian, perangkat lunak dapat dikirimkan ke customer

2.7.5 Operation and maintenance

Biasanya (walaupun tidak selalu), tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. Maintenance melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.

