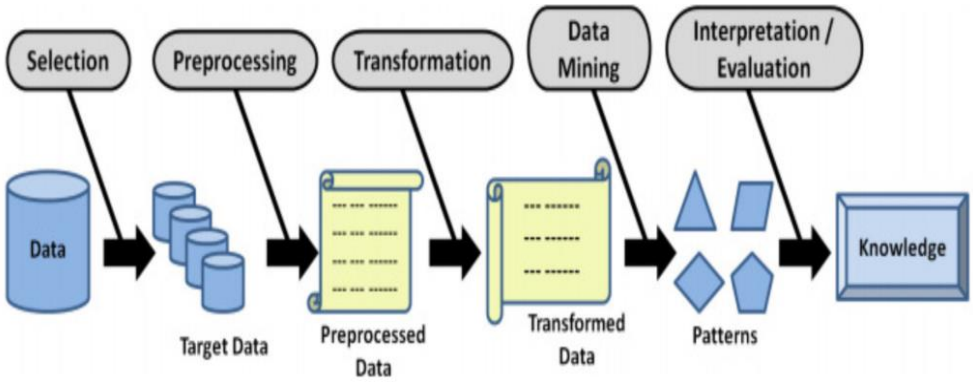
BAB II

LANDASAN TEORI

2.1 Knowledge Discovery in Database (KDD)

Untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar, data mining adalah salah satu langkah dalam proses *Knowledge Discovery in Database* (KDD) secara keseluruhan. Istilah "data mining" dan "KDD" sering disebutkan secara bersamaan. Meskipun kedua istilah tersebut berbeda, mereka saling terkait. Data mining juga merupakan bagian dari proses keseluruhan KDD. Gambar 2.1 menunjukkan proses KDD secara rinci.



Gambar 2.1 Proses *Knowledge Discovery in Database* (KDD) [1]
Sumber: Gustientiedina,2019

Keterangan Gambar:

1. *Selection*

   Pemilihan data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam *Knowledge Discovery in Databases* (KDD) dimulai. Data yang telah diseleksi dan akan digunakan dalam proses data mining disimpan dalam sebuah berkas terpisah dari basis data operasional. Proses seleksi ini bertujuan untuk menentukan variabel yang

akan diambil, sehingga menghindari adanya kesamaan dan pengulangan yang tidak diperlukan dalam pengolahan data mining.

2. *Preprocessing*

   Sebelum proses data mining dapat dilaksanakan, penting untuk melakukan pembersihan terhadap data yang menjadi fokus *Knowledge Discovery in Databases* (KDD). Proses pembersihan ini mencakup beberapa langkah, antara lain menghapus duplikasi data, memeriksa inkonsistensi data, serta memperbaiki kesalahan yang terdapat pada data, seperti kesalahan cetak (tipografi). Dalam tahap *preprocessing*, terdapat dua tahapan yang dapat dijelaskan sebagai berikut:

   a. Data *Cleaning*

      Menghilangkan data yang tidak diperlukan seperti menangani *missing value*, *noise* data serta menangani data-data yang tidak konsisten dan relevan.

   b. Data *Integration*

      Dilakukan terhadap atribut yang mengidentifikasikan entitas yang unik.

3. *Transformation*

   Mengubah data sesuai dengan format ekstensi yang sesuai dalam pengolahan data mining sangatlah penting, mengingat beberapa metode dalam data mining memerlukan format khusus sebelum dapat diproses. Proses coding merupakan transformasi terhadap data yang telah dipilih, sehingga data tersebut menjadi layak untuk diproses lebih lanjut dalam konteks data mining. Dalam kerangka *Knowledge Discovery in Databases*

(KDD), proses coding merupakan kegiatan kreatif yang sangat bergantung pada jenis atau pola informasi yang ingin dicari dalam basis data.

4. *Data Mining*

   Proses utama dalam metode yang diterapkan bertujuan untuk memperoleh pengetahuan baru dari data yang telah diproses.

5. *Evaluation/ Interpretation*

   Mengidentifikasi pola-pola yang signifikan ke dalam basis pengetahuan yang telah ditentukan. Pada tahap ini, dilakukan penghasilan pola-pola khas serta model prediktif yang dievaluasi guna menilai apakah kajian yang telah dilakukan memenuhi target yang diinginkan.

6. *Knowledge*

   Pola-pola yang dihasilkan akan dipresentasikan kepada para pengguna. Pada tahap ini, pengetahuan baru yang dihasilkan dapat dipahami oleh semua pihak yang akan dijadikan acuan dalam pengambilan keputusan.

   KDD memiliki 4 fungsi dasar yang biasa kita temui yaitu [2]:

   1. Fungsi Klasifikasi

      Fungsi ini bertujuan untuk menemukan model atau fungsi yang dapat menggambarkan suatu kelas dari sekumpulan data. Melalui klasifikasi, sistem mampu mempelajari data sehingga dapat meramalkan kecenderungan yang terjadi di dalamnya.

   2. Fungsi Prediksi

      Fungsi ini digunakan untuk mengidentifikasi pola dalam data dengan memanfaatkan variabel tertentu untuk memprediksi variabel lain yang jenisnya belum diketahui.

3. Fungsi Deskripsi

   Fungsi ini dimanfaatkan untuk mengidentifikasi karakteristik yang terdapat dalam data.

4. Fungsi Asosiasi

   Fungsi ini berfungsi untuk menemukan hubungan yang ada antara nilai atribut dari sejumlah data.

2.2 *Machine Learning*

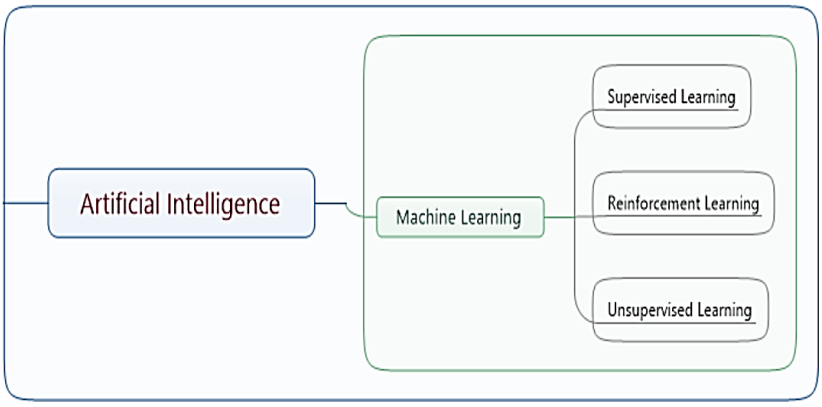
Pembelajaran mesin adalah suatu disiplin ilmu yang mempelajari algoritma komputer yang mampu mengidentifikasi pola-pola dalam data. Tujuan dari pembelajaran mesin adalah untuk mengubah berbagai jenis data menjadi tindakan yang konkret dengan minimal intervensi dari manusia. Menurut IBM, pembelajaran mesin merupakan cabang dari kecerdasan buatan *(Artificial Intelligence)* dan ilmu komputer yang berfokus pada pemanfaatan data serta algoritma untuk meniru cara manusia belajar dan secara bertahap meningkatkan tingkat akurasinya. Semakin baik algoritma pembelajaran mesin yang diterapkan, semakin optimal pula keputusan yang dihasilkan. [3].

Algoritma *machine learning* ini juga dapat mempelajari data dari data historis saja. Meskipun *machine learning* menjanjikan kesuksesan dalam pemprograman, ada beberapa masalah yang harus diperhatikan saat menggunakannya. Pembelajaran mesin merujuk pada suatu metode yang memungkinkan komputer untuk memiliki kemampuan dalam mempelajari dan melaksanakan suatu tugas secara otomatis. Proses pembelajaran mesin dilaksanakan melalui algoritma tertentu, sehingga tugas yang diberikan kepada komputer dapat diselesaikan secara otomatis. [4].

Pembelajaran mesin dilakukan melalui dua fase, yaitu fase pelatihan dan fase aplikasi. Fase pelatihan merupakan proses pemodelan di mana algoritma yang digunakan akan dipelajari oleh sistem melalui data pelatihan. Sementara itu, fase aplikasi adalah proses di mana model yang telah dipelajari oleh sistem selama fase pelatihan digunakan untuk menghasilkan keputusan tertentu, dengan memanfaatkan data uji.

2.2.1 Teknik *Machine Learning*

Penelitian terkini menunjukkan bahwa pembelajaran mesin *(machine learning)* terbagi menjadi tiga kategori, yaitu Pembelajaran Terawasi *(Supervised Learning)*, Pembelajaran Tidak Terawasi *(Unsupervised Learning)*, dan Pembelajaran Penguatan *(Reinforcement Learning)*. Skema keterkaitan antara kecerdasan buatan *(artificial intelligence)* dan pembelajaran mesin dapat dijelaskan dalam Gambar 2. 2.



Gambar 2.2 Skema *Artificial Intelligence* dan *Machine Learning* [5]
Sumber: A. Roihan,2020

Teknik yang diterapkan dalam *Supervised Learning* adalah metode klasifikasi, di mana seluruh kumpulan data telah dilengkapi dengan label untuk mengklasifikasikan kelas yang tidak diketahui. Sementara itu, teknik *Unsupervised Learning* sering kali disebut sebagai *clustering*, karena tidak ada

kebutuhan untuk memberikan label pada kumpulan data, dan hasilnya tidak mengidentifikasi contoh dalam kelas yang telah ditentukan. [6]. *Reinforcement Learning*, di sisi lain, biasanya berposisi di antara *Supervised Learning* dan *Unsupervised Learning*. Teknik ini beroperasi dalam lingkungan yang dinamis, di mana prinsip dasarnya adalah menyelesaikan suatu tujuan tanpa adanya pemberitahuan eksplisit dari komputer mengenai tercapainya tujuan tersebut.

Metode pembelajaran terawasi *(supervised learning)* didasarkan pada sekumpulan data sampel yang telah diberi label. Kumpulan sampel ini digunakan untuk meringkas karakteristik distribusi ukuran perilaku dalam setiap jenis aplikasi, sehingga membentuk model perilaku dari data tersebut. Pembelajaran terawasi dapat diklasifikasikan lebih lanjut menjadi dua kategori, yaitu masalah klasifikasi dan masalah regresi. Masalah klasifikasi terjadi ketika variabel keluaran berbentuk kategori, seperti warna (merah atau biru) atau kondisi kesehatan (ada penyakit atau tidak). Sebaliknya, masalah regresi terjadi ketika variabel keluaran berupa nilai riil, seperti dalam satuan dolar atau berat.

Pembelajaran terawasi memiliki sejumlah algoritma yang popular, di antaranya adalah *Back-propagation, Regresi Linier, Random Forest, Support Vector Machines, Naïve Bayesian, Metode Rocchio,* Pohon Keputusan*, K-Nearest Neighbor,* Jaringan Saraf, *Regresi Logistik, dan* Jaringan Saraf Tiruan. Selain itu, beberapa algoritma yang digunakan untuk klasifikasi meliputi *Support Vector Machines (SVM), Normal Bayesian Classifier (NBC), K-Nearest Neighbor (KNN), Gradient Boosted Trees (GBT), Random Trees (RT), dan Artificial Neural Networks (ANN).* Algoritma lainnya yang juga dibahas mencakup *Gaussian*

Mixture Models, Hidden Markov Models, regresi logistik, Kernel Regression, Deep Neural Networks, Deep Belief Networks, PCA, dan Kernel Perceptron.

Dalam jenis pembelajaran yang dikenal sebagai *Unsupervised Learning*, sistem diberikan sejumlah input sampel tanpa adanya output yang tersedia. Karena tidak terdapat output yang diharapkan, proses kategorisasi dilakukan agar algoritma dapat dengan tepat membedakan antara kumpulan data yang ada. Tugas utama dari pendekatan ini adalah mendefinisikan fungsi yang mencerminkan struktur tersembunyi dari data yang tidak memiliki label. Unsupervised Learning dapat dikelompokkan lebih lanjut ke dalam masalah pengelompokan *(clustering)* dan asosiasi. Masalah pengelompokan berfokus pada identifikasi kluster yang melekat dalam data, seperti mengelompokkan pelanggan berdasarkan perilaku pembelian mereka. Sementara itu, masalah asosiasi berkaitan dengan pencarian aturan yang menggambarkan hubungan dalam sebagian besar data, contohnya, individu yang membeli item A cenderung juga membeli item B.

Reinforcement Learning memiliki akar dari teori pembelajaran hewan. Metode pembelajaran ini tidak memerlukan pengetahuan sebelumnya dan mampu secara mandiri mengembangkan kebijakan optimal melalui pengalaman yang diperoleh dari interaksi berulang dengan lingkungan yang dinamis. Masalah dalam *reinforcement learning* dapat diselesaikan dengan cara mempelajari pengalaman baru melalui pendekatan coba-coba. Algoritma reinforcement learning sering kali berkaitan dengan algoritma pemrograman dinamis yang umum digunakan untuk menyelesaikan permasalahan optimisasi. Berbagai masalah dalam konteks machine learning di dunia nyata termasuk dalam kategori ini, karena proses pelabelan data dapat menjadi biaya yang tinggi dan memakan

waktu, serta sering kali memerlukan akses kepada pakar. Sebaliknya, data tanpa label dapat diperoleh dengan biaya rendah, serta lebih mudah untuk dikumpulkan dan disimpan.

Secara khusus, metode pembelajaran penguatan yang didasarkan pada model proses pengambilan keputusan Markov *(Markov Decision Process)* terdiri atas dua kategori utama. Pertama, terdapat metode berbasis model, di mana algoritma SARSA merupakan contohnya. Dalam metode ini, pembelajaran penguatan terlebih dahulu mempelajari pengetahuan mengenai model, kemudian memperoleh strategi optimal berdasarkan pengetahuan tersebut. Kedua, terdapat metode yang tidak memerlukan model, seperti algoritma Temporal Difference dan algoritma *Q-learning*, di mana pembelajaran penguatan secara langsung menghitung strategi optimal tanpa bergantung pada pengetahuan mengenai model [5].

2.3 *Naïve Bayes*

Naïve Bayes merupakan salah satu algoritma pembelajaran induktif yang sangat efektif dan efisien dalam konteks pembelajaran mesin *(machine learning)* dan penambangan data (data mining). Algoritma ini menggunakan pendekatan pengklasifikasian berdasarkan metode probabilitas dan statistik yang dikembangkan oleh ilmuwan Inggris, *Thomas Bayes*. Teknik ini memprediksi kemungkinan terjadinya suatu peristiwa di masa depan berdasarkan pengalaman yang diperoleh dari peristiwa-peristiwa sebelumnya, sehingga dikenal sebagai *Teorema Bayes.* [7]. *Naïve Bayes* merupakan suatu teknik prediksi yang berbasis probabilistik dan bersifat sederhana, yang merujuk pada penerapan *Teorema*

Bayes dengan *asumsi independensi* yang kuat. Metode ini memanfaatkan *probabilitas* bersyarat sebagai dasar utama. *Algoritma Naïve Bayes* dibangun atas dasar fungsi probabilitas untuk setiap instance, yang menguraikan pemetaan antara atribut dan klasifikasi atribut dalam suatu sistem yang menunjukkan efisiensi klasifikasi yang stabil dan kompleksitas yang rendah. Dalam *implementasinya, Naïve Bayes* mengasumsikan bahwa nilai atribut saling *independen.* [8].

Model Naïve Bayes termasuk dalam kategori pembelajaran terawasi (supervised learning), di mana pembelajaran terawasi merupakan suatu teknik dalam machine learning yang memungkinkan mesin untuk mengidentifikasi label dari suatu input dengan memanfaatkan pembelajaran yang diperoleh dari data pelatihan yang telah diberi label. Proses ini bertujuan untuk melakukan prediksi atau klasifikasi. Oleh karena itu, pada tahap pembelajaran *Naïve Bayes*, diperlukan data pelatihan agar diperoleh hasil probabilitas yang akurat. Setelah hasil probabilitas diperoleh, tahap selanjutnya adalah proses pengklasifikasian, di mana data pengujian akan digunakan untuk melakukan perhitungan lebih lanjut [9]. *Naïve Bayes* adalah suatu metode pengklasifikasian data yang berbasis probabilitas dan memiliki pendekatan yang sederhana. Metode ini menghitung sejumlah probabilitas dengan cara mengombinasikan frekuensi nilai dari sebuah dataset. Proses perhitungan dalam metode ini didasarkan pada kemiripan probabilitas antara kasus yang telah ada sebelumnya dengan kasus yang baru. [10]. *Teorema Bayes* memiliki persamaan umum sebagai berikut [7]:

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \qquad (1)$$

Dimana pada persamaan (1),

X : Data class yang belum diketahui

H : Hipotesis data X merupakan suatu class spesifik

P(H) : Probabilitas dari hipotesis H

P(X) : Probabilitas dari X

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

P(X|H) : Probabilitas X berdasarkan kondisi pada hipotesis H

Berdasarkan persamaan (1) *teorema bayes* maka inilah rumus dari *Naïve Bayes Classifier* [11]:

$$(Ci|X) = \frac{P(X|Ci)P(Ci)}{P(X)} \quad (2)$$

Dimana pada persamaan (2),

X : Data class yang belum diketahui

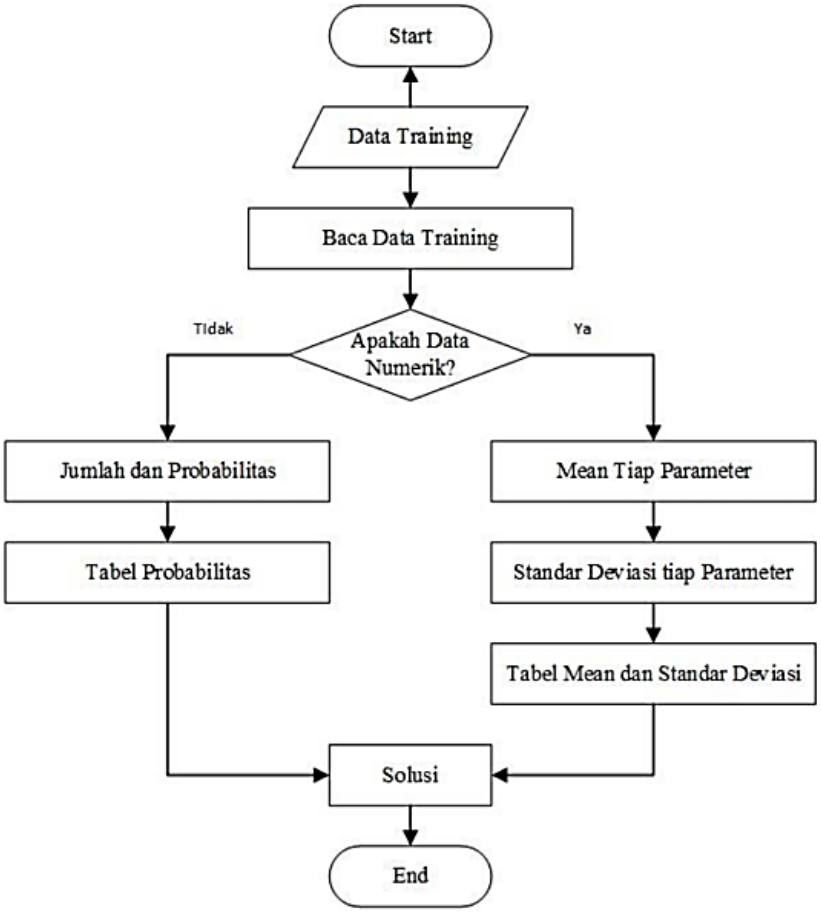
C¬i : Hipotesis data X merupakan suatu class spesifik

P(Ci|X) : Probabilitas hipotesis Ci berdasarkan kondisi X

P(Ci) : Probabilitas hipotesis Ci

P(X|Ci) : Probabilitas X berdasarkan kondisi pada hipotesis Ci

P(X) : Probabilitas X



Gambar 2.3 Alur Kerja Algoritma *Naïve Bayes* [12]

Sumber: D. Putra and A. Wibowo,2020

Dari gambar 2.3 dapat dijelaskan alur kerja dari algoritma *Naïve Bayes* dijelaskan sebagai berikut [12]:

1. Baca data *training*.
2. Hitung jumlah atau probabilitas, namun apabila data numerik maka:
3. Cari nilai mean dan standar deviasi dari masing-masing parameter yang merupakan data numerik.
4. Cari nilai probabilitas dengan cara menghitung jumlah data yang sesuai dari dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut.
5. Mendapatkan nilai dalam table mean, standar deviasi dan probabilitas.

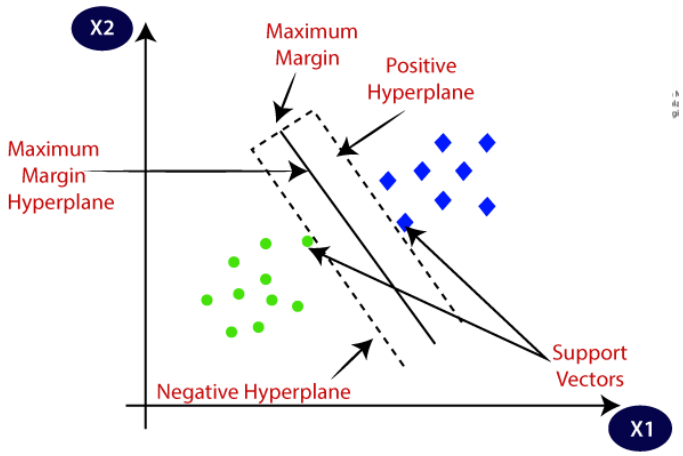
2.4 *Support Vector Machine* (SVM)

Support Vector Machine (SVM) merupakan suatu sistem pembelajaran yang memanfaatkan ruang hipotesis dalam bentuk fungsi-fungsi linier di dalam ruang fitur berdimensi tinggi. Sistem ini dilatih menggunakan algoritma pembelajaran yang berlandaskan teori optimasi, dengan menerapkan bias induktif yang bersumber dari teori pembelajaran statistik. Teori yang menjadi dasar SVM telah mengalami perkembangan sejak tahun 1960-an. Namun, SVM diperkenalkan secara resmi oleh Vapnik, Boser, dan Guyon pada tahun 1992, dan sejak saat itu, SVM telah mengalami perkembangan yang pesat. [13].

Support Vector Machine (SVM) merupakan salah satu teknik yang relatif baru jika dibandingkan dengan teknik-teknik lainnya, namun menunjukkan performa yang superior di berbagai bidang aplikasi, seperti bioinformatika, pengenalan tulisan tangan, klasifikasi teks, dan lain-lain. Proses pembelajaran dalam SVM bertujuan untuk memperoleh hipotesis yang berupa bidang pemisah optimal yang tidak hanya meminimalkan risiko empiris, yaitu rata-rata kesalahan pada data pelatihan, tetapi juga memiliki kemampuan generalisasi yang baik. Generalisasi di sini merujuk pada kemampuan sebuah hipotesis untuk mengklasifikasikan data yang tidak terdapat dalam data pelatihan secara akurat. Untuk memastikan kemampuan generalisasi ini, SVM beroperasi berdasarkan prinsip Minimasi Risiko Struktural. (SRM) [14].

Algoritma *Support Vector Machine* (SVM) merupakan salah satu metode dalam *machine learning* yang tergolong *supervised*. Algoritma ini terbukti sangat efektif dalam menganalisis data untuk tujuan klasifikasi. SVM melakukan penilaian yang mendalam terhadap data untuk mengklasifikasikannya ke dalam

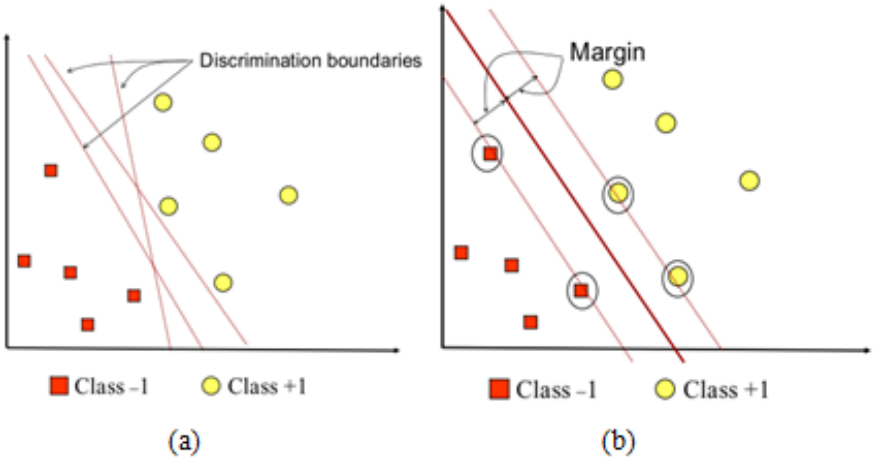
kategori data normal dan data berbahaya. Berkat kemampuan klasifikasinya, algoritma ini telah digunakan secara luas selama bertahun-tahun dalam bidang deteksi intrusi. Algoritma SVM, yang dikembangkan oleh Vladimir Vapnik, telah mendapatkan kepercayaan dari banyak administrator keamanan. Selain itu, SVM juga mampu mengklasifikasikan data yang bersifat non-linear. [15].



Gambar 2.4 Model Algoritma SVM [16]

Sumber: S. Huang,2018

SVM bekerja dengan cara mencari batasan dari dua kelas berbeda sehingga memungkinkan untuk membuat prediksi terhadap label data. Batas atau disebut juga dengan *hyperlane* ini berorientasi sedemikian rupa sehingga terletak sejauh mungkin dari data point dari masing-masing kelas. Data point terdekat ini disebut dengan *support vector*.



Gambar 2.5 Penemuan *Hyperlane* pada SVM [17]

Sumber: Marhamah,2020

Gambar 2. 5 menunjukkan beberapa pola yang termasuk dalam dua kelas, yaitu -1 dan +1. Pola pada kelas -1 dilambangkan dengan warna merah (berbentuk kotak), sedangkan pola pada kelas +1 dilambangkan dengan warna kuning (berbentuk lingkaran). Masalah klasifikasi dapat diselesaikan dengan mencari garis pemisah (hyperplane) yang memisahkan kedua kelas tersebut. Gambar (a) memperlihatkan beberapa alternatif batas diferensiasi yang berbeda. *Hyperplane* dengan pemisah terbaik dapat diidentifikasi dengan mengukur margin dari *hyperplane* dan mencari titik maksimum. Penggunaan kernel adalah penting untuk mencapai keberhasilan dalam banyak algoritma klasifikasi pada permukaan linear. Dengan demikian, dapat disimpulkan bahwa jenis kernel yang digunakan dapat mempengaruhi hasil klasifikasi yang dilakukan.

Hyperplane merupakan garis pemisah terbaik antara dua kelas. Proses pencarian *hyperplane* ini dapat dilakukan dengan cara mencari margin *hyperplan*e dan menentukan titik maksimum. Margin didefinisikan sebagai jarak antara data terdekat di antara dua kelas yang berbeda, yang disebut sebagai *support vector*. Garis solid pada gambar (b) menunjukkan *hyperplane* yang optimal, karena

terletak tepat di antara kedua kelas tersebut, sementara *support vector* dilambangkan dengan titik merah dan kuning yang berada dalam lingkaran hitam.

Hyperlane klasifikasi *linear* SVM dinotasikan:

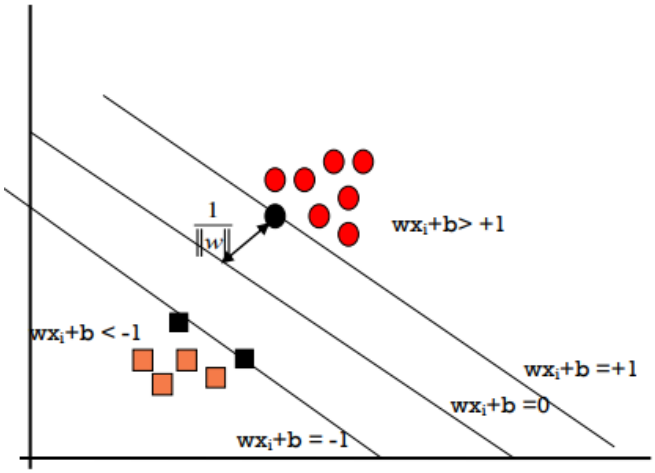
$$f(x) = w,\ x + b = 0 \qquad (3)$$

Dari persamaan di atas didapatkan pertidaksaman kelas +1 (positif)

$$w.x + b \ \leq \ +1 \qquad (4)$$

Pertidaksamaan kelas -1:

$$w.x + b \ \geq \ -1 \qquad (5)$$



Gambar 2.6 Fungsi Pemisahan optimal SVM [17]

Sumber: Marhamah,2020

W dinyatakan sebagai vektor normal, sementara b merujuk pada posisi relatif bidang terhadap pusat koordinat. Dengan mengoptimalkan jarak antara *hyperplane* dan titik terdekat, margin maksimum dapat diidentifikasi, yang besarnya adalah 1 / ||w||. Permasalahan ini dapat dirumuskan sebagai masalah pemrograman kuadratik (QP), di mana tujuan adalah untuk mencapai titik minimum dari persamaan (4) dengan mempertimbangkan kendala yang ada dalam persamaan tersebut.

$$\boldsymbol{min}\ \frac{1}{2}||w||^2 = \boldsymbol{min}\ \frac{1}{2}(w1^2 + w2^2) \quad (6)$$

$$y_i(w_{xi} + b)\ \geq 1, i = 1,2,3 \ldots, N \quad (7)$$

2.5 *Confusion Matrix*

Confusion Matrix adalah alat visualisasi yang umum digunakan dalam pembelajaran terawasi (*supervised learning*). Setiap kolom dalam matriks mewakili contoh dari kelas prediksi, sedangkan setiap baris mencerminkan kejadian dalam kelas yang sebenarnya. Metode ini menggunakan tabel matriks dalam proses evaluasi apabila dataset memiliki dua kelas, yaitu kelas yang dianggap positif dan kelas lainnya yang dianggap negatif. Evaluasi menggunakan confusion matrix ini menghasilkan nilai akurasi, presisi, dan recall terkait klasifikasi yang telah dilakukan. [11]. Akurasi dalam klasifikasi mengacu pada presentase ketepatan data yang diklasifikasikan dengan benar setelah melalui proses pengujian terhadap hasil klasifikasi. Precision adalah proporsi yang menunjukkan jumlah prediksi positif yang juga benar-benar positif berdasarkan data aktual. Di sisi lain, recall merujuk pada proporsi kasus positif yang sebenarnya berhasil diprediksi dengan benar sebagai positif. [12].

Matriks kebingungan *(confusion matrix)* merupakan sebuah tabel yang menyajikan jumlah data uji yang diklasifikasikan dengan benar serta jumlah data uji yang diklasifikasikan dengan salah. Matriks kebingungan merupakan alat yang berguna untuk menganalisis seberapa efektif suatu pengklasifikasi dalam mengidentifikasi kelas-kelas yang berbeda. Nilai *True Positive* (TP) dan *True Negative* (TN) menjelaskan situasi di mana pengklasifikator berhasil memberikan klasifikasi yang tepat, sedangkan nilai *False Positive* (FP) dan *False Negative* (FN) menunjukkan kondisi di mana pengklasifikator melakukan klasifikasi yang

keliru. [18]. Contoh *confusion matrix* untuk klasifikasi biner ditunjukkan pada tabel 2.1.

Tabel 2.1 *Confusion Table*

Confusion Matrix		Nilai Prediksi	
		Positive	Negative
Nilai Sebenarnya	Positive	TP	FN
	Negative	FN	TN

Persamaan *Accuracy* :

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad \text{.....(8)}$$

Persamaan *Recall* :

$$Recall = \frac{TP}{FN+TP} \quad \text{.....(9)}$$

Persamaan *Precision* :

$$Precision = \frac{TP}{FP+TP} \quad \text{.....(10)}$$

Keterangan:

- TP = *True Positive*
- TN = *True Negative*
- FP = *False Positive*
- FN = *False Negative*

True positive dan *false positive* merupakan istilah yang merujuk pada jumlah rekaman positif dan negatif yang diklasifikasikan sebagai positif, sedangkan *false negative* dan *true negative* merujuk pada jumlah rekaman positif dan negatif yang diklasifikasikan sebagai negatif. Selanjutnya, data uji dimasukkan dan dilakukan perhitungan untuk menentukan nilai sensitivitas, spesifikasi, presisi, dan akurasi dari data yang telah diinput. Berdasarkan isi

matriks pada tabel tersebut, dapat diketahui jumlah data dari masing-masing kelas yang diprediksi secara benar, yaitu jumlah *True Positives* ditambah dengan *True Negatives*, serta jumlah data yang diklasifikasikan secara salah, yaitu jumlah *False Positives* ditambah dengan *False Negatives.* [11].

2.6 Alat Bantu Analisis

2.6.1 *Rapidminer*

RapidMiner merupakan sebuah platform perangkat lunak di bidang ilmu data yang dikembangkan oleh perusahaan dengan nama yang sama. Platform ini menyediakan lingkungan terintegrasi untuk berbagai kegiatan, termasuk persiapan data, pembelajaran mesin, pembelajaran dalam, penambangan teks, dan analisis prediktif. *RapidMiner* digunakan tidak hanya dalam konteks bisnis dan komersial, tetapi juga untuk penelitian, pendidikan, pelatihan, pengembangan prototipe cepat, dan pengembangan aplikasi. Selain itu, platform ini mendukung seluruh langkah dalam proses pembelajaran mesin, yang mencakup persiapan data, visualisasi hasil, validasi model, dan optimasi. [19]. *RapidMiner* merupakan salah satu perangkat lunak yang digunakan untuk pengolahan data mining. Fungsi yang dilaksanakan oleh *RapidMiner* dalam bidang text mining mencakup analisis teks, ekstraksi pola-pola dari dataset yang besar, serta penggabungan teknik-teknik statistika, kecerdasan buatan, dan database. [20].

RapidMiner dikenal dengan istilah YALE (*Yet Another Learning Environment*), yang diprakarsai oleh Ralf Klinkenberg, Ingo Mierswa, dan Simon Fischer di Unit Kecerdasan Buatan Universitas Teknik Dortmund pada tahun 2001. Pada tahun 2006, pengembangan *RapidMiner* didorong oleh Rapid-I, sebuah perusahaan yang didirikan oleh Ingo Mierswa dan Ralf Klinkenberg pada

tahun yang sama. Pada tahun 2007, perangkat lunak ini mengalami perubahan nama dari YALE menjadi *RapidMiner*. Selanjutnya, pada tahun 2013, perusahaan melakukan rebranding dari Rapid-I menjadi *RapidMiner*. Saat ini, RapidMiner digunakan di berbagai bidang, termasuk bisnis, pendidikan, penelitian, pelatihan, serta pengembangan prototipe, yang mendukung proses pembelajaran mesin melalui visualisasi, pengembangan model, persiapan data, dan optimasi. [21].

Gambar 2.6 Tampilan Aplikasi *RapidMiner* [22]

Sumber: A. M. Khalimi,2022

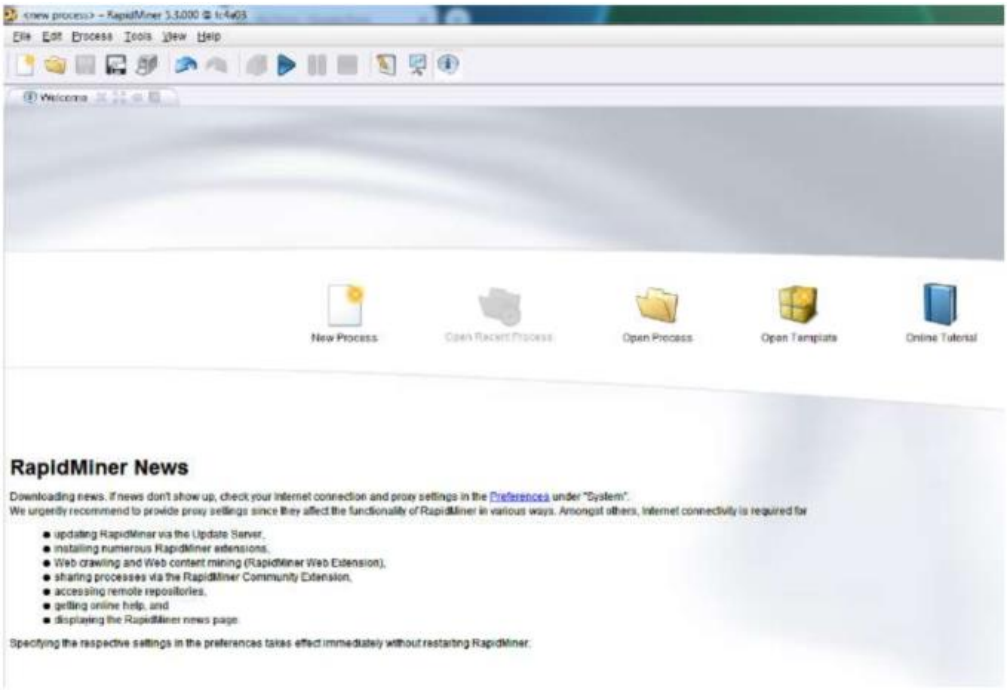
RapidMiner adalah perangkat lunak *open source* yang dirancang sebagai solusi untuk analisis data mining, text mining, dan analisis prediktif. Perangkat ini berfungsi sebagai aplikasi mandiri untuk analisis data serta sebagai mesin data mining yang dapat diintegrasikan ke dalam produk lain. *RapidMiner* dikembangkan menggunakan bahasa pemrograman Java, sehingga dapat beroperasi di berbagai sistem operasi. Dengan memanfaatkan berbagai teknik deskriptif dan prediktif, RapidMiner memberikan wawasan yang berharga kepada

pengguna, yang pada gilirannya dapat membantu pengambilan keputusan yang lebih optimal. *RapidMiner* dilengkapi dengan kurang lebih 500 operator data mining, termasuk operator untuk penginputan, pengeluaran, pra-pemrosesan data, dan visualisasi. [23].

2.6.2 Bagian-Bagian *RapidMiner*

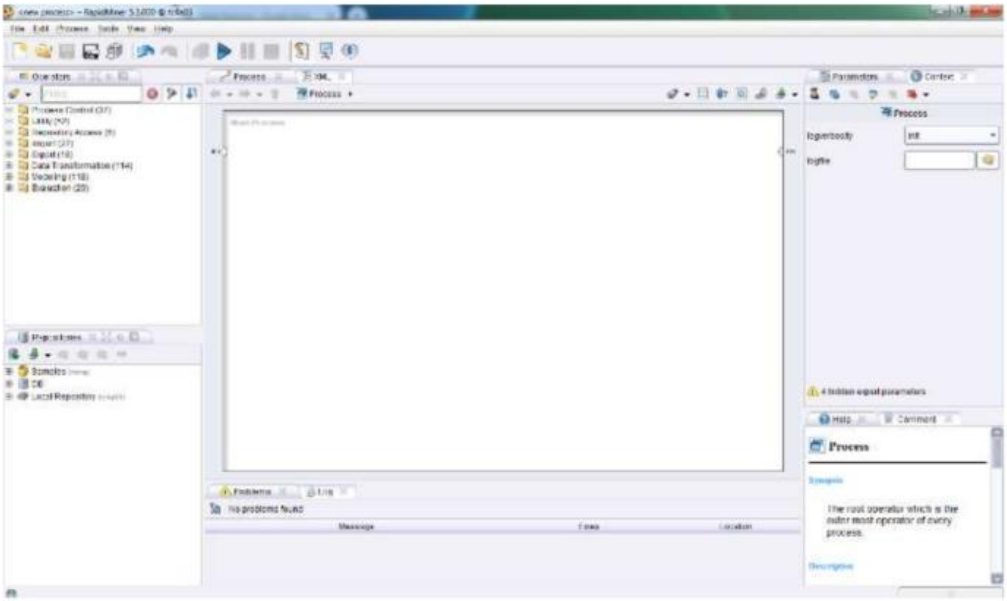
RapidMiner menyediakan tampilan (UI) yang ramah pengguna, sehingga memudahkan pengguna saat menggunakannya. Tampilan yang terdapat pada *RapidMiner* disebut *Perspective*. Terdapat 3 *Perspective*, yaitu *Welcome Perspective*, *Design Perspective* dan *Result Perspective* [24].

Ketika pengguna membuka aplikasi RapidMiner, mereka akan disambut dengan tampilan *Welcome Perspective*, seperti yang terlihat pada Gambar 2. 4. Selanjutnya, pengguna akan beralih ke *Design Perspective*, yang merupakan area utama untuk bekerja dalam *RapidMiner*. *Design Perspective* berfungsi sebagai ruang untuk merancang dan mengelola proses analisis. Dalam tampilan ini, sebagaimana ditunjukkan pada Gambar 2. 5, terdapat beberapa view dengan fungsi yang berbeda-beda, yang semuanya mendukung proses analisis data. Selain itu, terdapat juga *Result Perspective*, yang menampilkan hasil dari analisis. Hasil yang ditampilkan dapat bervariasi, mulai dari grafik, teks, pohon, tabel, dan lain sebagainya, tergantung pada jenis data dan metode yang digunakan dalam analisis.



Gambar 2.7 *Welcome Perspective RapidMiner* [24]

Sumber: V. R. Prasetyo,2021

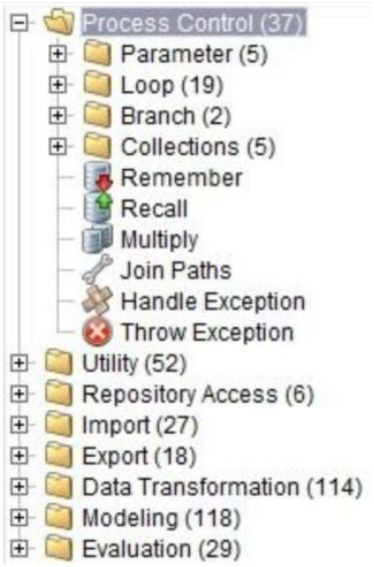


Gambar 2.8 *Design Perspective RapidMiner* [24]

Sumber: V. R. Prasetyo,2021

Dalam lingkungan kerja *Design Perspective*, terdapat beberapa tampilan yang dapat digunakan. Tampilan yang paling utama dan esensial adalah *Operator View*. Di sinilah semua operator dan langkah-langkah kerja dalam RapidMiner ditampilkan dalam bentuk grup hierarkis, memungkinkan pengguna untuk mengintegrasikannya dalam proses analisis, seperti yang terlihat pada Gambar 2. 6.Berdasarkan gambar tersebut, Operator View dibagi menjadi beberapa kelompok. Pertama, terdapat kelompok Process Control, yang berisi logika looping dan percabangan untuk mengatur alur proses analisis data. Kedua, terdapat kelompok Utility yang berfungsi sebagai operator pendukung, seperti macros, login, subproses, dan lain-lain. Kelompok selanjutnya adalah *Repository Access*, yang digunakan untuk membaca atau menulis akses ke *repository*. Berikutnya adalah kelompok Import, yang berperan untuk membaca data dan objek dari berbagai sumber seperti file dan database. Kemudian ada kelompok *Export*, yang memiliki fungsi berlawanan dengan *Import*, digunakan untuk menulis data ke dalam format tertentu.

Selanjutnya, terdapat operator Data Transformation yang berguna untuk melakukan transformasi data dan metadata. Kemudian, kita memiliki *operator Modelling*, yang mencakup berbagai metode dan teknik data mining untuk pengolahan data. Terakhir, ada kelompok Evaluation, yang berfungsi untuk mengevaluasi kualitas output yang dihasilkan.



Gambar 2.9 *Operator View RapidMiner* [24]

Sumber: V. R. Prasetyo,2021

Selain *operator view*, *repository view* juga merupakan komponen yang cukup penting dalam *Design Perspective*. Pada *view* ini, user dapat mengelola dan mengatur proses analisis data menjadi sebuah proyek, di mana pada saat yang sama *view* tersebut berperan sebagai sumber data dan hal-hal yang terkait dengan metadata. *Process view* juga berfungsi untuk menunjukkan tahaptahap tertentu dalam proses analisis dan juga berfungsi sebagai tautan ke langkah-langkah berikutnya.

2.7 Penelitian Terdahulu

Penelitian terdahulu yang menjadi referensi dalam studi ini sebagian besar berfokus pada penerapan algoritma *Naïve Bayes* dan *Support Vector Machine* (SVM) untuk berbagai kasus klasifikasi data. Beberapa penelitian relevan mencakup analisis sentimen, prediksi tren pasar, dan preferensi konsumen.

Tabel 2.2 Penelitian Terdahulu

No	Judul	Penulis	Metode dan Data	Hasil
1	Komparasi Algoritma *Naïve Bayes* Dan *Support Vector Machine* (SVM) Pada Analisis Sentimen *Spotify*.	(Rahayu et al., 2022)	• Penelitian ini menggunakan metode *Naïve Bayes* dan *Support Vector Machine.* • Dataset yang digunakan diambil dari ulasan pengguna aplikasi *Spotify* di *Play Store* sebanyak 1500 ulasan.	Penelitian ini membandingkan kinerja *Naïve Bayes* dan *Support Vector Machine* (SVM) dalam analisis sentimen ulasan pengguna *Spotify*. Hasilnya menunjukkan bahwa *Naïve Bayes* mencapai akurasi 86,4%, sementara SVM mencapai 84%, mengindikasikan bahwa *Naïve Bayes* sedikit lebih unggul dalam konteks ini.
2	Perbandingan Algoritma *Naïve Bayes* dan *Support Vector Machine* (SVM) dalam Analisis Sentimen Kebijakan Kemdikbudristek Tentang Kuota Internet Selama Pandemi Covid-19.	(Khaira et al., 2023)	• Metode *Naïve Bayes* dan *Support Vector Machine* (SVM) digunakan untuk melakukan analisis sentimen terhadap kebijakan Kemdikbudristek mengenai Kuota Kemdikbud dalam proses mengklasifikasi suatu tweet berdasarkan Tingkat emosinya. • Dataset yang digunakan diambil dari Twitter dengan menggunakan Teknik *scraing* sebanyak 1078 ulasan.	Pada penelitian ini didapatkan nilai akurasi untuk algoritma *Naïve Bayes* sebesar 64%. Sedangkan untuk algoritma *Support Vector Machine* (SVM) sebesar 80%. Dari hasil analisis tersebut didapatkan nilai akurasi pada metode SVM lebih tinggi dibandingkan dengan metode *Naïve Bayes*.

3	Perbandingan Algoritma SVM dan *Naïve Bayes* dalam Analisis Sentimen *Twitter* pada Penggunaan Mobil Listrik di Indonesia.	(Ningsih et al., 2024)	• Metode yang digunakan untuk membangun model klasifikasi berdasarkan data yang diperoleh pada penelitian ini adalah Algoritma *Naïve Bayes* dan *Support Vector Machine*. • Dataset pada penelitian ini adalah kumpulan data atau informasi analisis sentimen twitter terhadap penggunaan mobil listrik di Indonesia yang didapat daril laman Kaggle Open Dataset (kagglel.com) sebanyak 1517 data.	Hasil penelitian menunjukkan nilai akurasi implementasi klasifikasi yang diperoleh algoritma *Naïve Bayes* sebesar 63,02% dan akurasi *Support Vector Machine* sebesar 70,82%. Dapat disimpulkan bahwa algoritma *Support Vector Machine* mempunyai nilai akurasi yang paling tinggi.
4	Perbandingan Algoritma SVM dan Naïve Bayes dalam Analisis Sentimen Komentar Tiktok pada Produk Skincare.	(Setiawan et al., 2024)	• Penelitian ini membandingkan performa algoritma *Support Vector Machine* (SVM) dan *Naïve Bayes* dalam analisis sentimen komentar TikTok tentang produk skincare. • Sumber data dikumpulkan dari platform media sosial yang bernama Tiktok. Data yang dikumpulkan mencakup komentar dalam Bahasa Indonesia. Data yang diambil	Penelitian ini membandingkan kinerja SVM dan *Naïve Bayes* dalam menganalisis sentimen komentar pengguna TikTok terkait produk skincare. Hasilnya menunjukkan bahwa SVM memiliki akurasi lebih tinggi (59,43%) dibandingkan *Naïve Bayes* (47,65%), dengan *F1 Score* SVM sebesar 60,37% dan *Naïve Bayes*

			sebanyak dibagi menjadi 70% untuk train dan 30% untuk test.	54,74%.
5	Perbandingan Metode *Naïve Bayes* dan *Support Vector Machine* pada Analisis Sentimen Aplikasi Mamikos	(Nurdin et al., 2024)	• Penelitian ini mengevaluasi performa *Naïve Bayes* dan SVM dalam menganalisis sentiment ulasan pengguna aplikasi Mamikos di *Google Playstore*. • Aplikasi Mamikos yang tersedia di *Google Playstore* digunakan untuk *crawling* data ulasan, yang menghasilkan 4000 data yang diambil pada bulan November 2024. Dataset terdiri dari 4000 data, tetapi setelah proses *cleaning* data menjadi 3995 data karena konten duplikat dihapus.	Dengan menggunakan data ulasan dari *Google Playstore*, hasil menunjukkan bahwa SVM mencapai akurasi 92%, sedangkan *Naïve Bayes* mencapai 82%, mengindikasikan keunggulan SVM dalam konteks ini.

2.8 Metodologi Penelitian

Bab ini menjelaskan metodologi yang digunakan dalam penelitian untuk membandingkan algoritma *Naïve Bayes* dan *Support Vector Machine* (SVM) dalam menentukan peminat pembeli *smartphone* di Erafone Suzuya Baru Rantauprapat. Metode penelitian ini dirancang untuk menjamin proses pengumpulan, pengolahan, dan analisis data berjalan secara sistematis dan dapat menghasilkan hasil yang valid serta relevan dengan tujuan penelitian.

2.8.1 Studi Pustaka

Tahapan studi pustaka ini memastikan bahwa penelitian memiliki dasar yang kuat, metode yang relevan, dan kontribusi yang signifikan terhadap pengembangan ilmu pengetahuan. Dengan pendekatan yang sistematis, peneliti mampu menjawab tujuan penelitian secara efektif. Berikut adalah tahapan studi pustaka yang dilakukan:

1. **Identifikasi Topik Penelitian**

   Pada tahap awal, peneliti mengidentifikasi dan memahami topik utama penelitian, yaitu:

   1. Analisis Perbandingan Algoritma *Naïve Bayes* dan *Support Vector Machine* (SVM)
   2. Menentukan minat pembeli *smartphone* di Erafone Suzuya Baru Rantauprapat

   Proses ini melibatkan peninjauan literatur umum tentang:

   1. Perkembangan algoritma *machine learning* dalam analisis data konsumen.
   2. Strategi pemasaran berbasis data di industri ritel elektronik.

2. **Pengumpulan Referensi Relevan**

   Peneliti mengumpulkan referensi dari berbagai sumber untuk mendukung topik penelitian. Sumber referensi meliputi:

   1. Artikel Jurnal

      Studi terbaru dari jurnal internasional dan nasional, khususnya yang diterbitkan tahun 2023–2024, untuk memastikan relevansi data.

   2. Buku Teori

Buku yang menjelaskan konsep dasar algoritma *Naïve Bayes*, SVM, dan teknik analisis data.

3. Dokumentasi Praktis

   Panduan teknis tentang implementasi algoritma machine learning menggunakan perangkat lunak *Rapidminer*.

Referensi yang dipilih harus sesuai dengan kriteria:

1. Relevansi

   Terkait langsung dengan tujuan penelitian.

2. Keaktualan

   Berasal dari 5 tahun terakhir untuk menjaga relevansi terhadap perkembangan teknologi.

3. Kredibilitas

   Sumber yang diakui secara akademis, seperti jurnal terindeks.

3. **Analisis Penelitian Terdahulu**

Peneliti menganalisis studi yang relevan dengan membandingkan pendekatan dan hasil penelitian mereka. Contohnya:

1. Penggunaan *Naïve Bayes* dan SVM dalam Klasifikasi Data

   Studi yang menunjukkan performa kedua algoritma dalam konteks klasifikasi seperti analisis sentimen atau pola pembelian. Hasil penelitian yang membandingkan akurasi, *precision*, *recall*, dan *F1-score* antara kedua algoritma.

2. Studi tentang Minat Konsumen

   Penelitian terkait analisis preferensi konsumen di pasar ritel elektronik. Faktor-faktor yang memengaruhi minat pembelian, seperti promosi, spesifikasi produk, dan loyalitas merek.

Hasil analisis ini membantu dalam merumuskan gap penelitian dan mengidentifikasi peluang kontribusi baru.

2.8.2 Studi Wawancara

Studi wawancara dalam penelitian ini memberikan wawasan yang kaya dan mendalam tentang preferensi dan perilaku konsumen terkait minat pembeli *smartphone* di Erafone Suzuya Baru Rantauprapat. Dengan pendekatan terstruktur, data kualitatif dari wawancara membantu memahami latar belakang keputusan pembelian secara lebih komprehensif dan mendukung perancangan strategi pemasaran yang lebih tepat sasaran.

1. **Perencanaan Wawancara**

   Tahapan ini bertujuan untuk memastikan wawancara dilakukan secara terstruktur dan efektif. Aktivitas utama meliputi Penentuan tujuan wawancara, Identifikasi responden, Penyusunan pedoman wawancara, dan Menentukan jadwal serta logistik wawancara.

2. **Pelaksanaan Wawancara**

   Tahapan ini adalah proses inti dari studi wawancara. Aktivitas utama meliputi: Pendekatan kepada responden, Pengumpulan data, serta Pencatatan dan Dokumentasi,

3. **Analisis Hasil Wawancara**

   Setelah data wawancara terkumpul, tahap berikutnya adalah menganalisis hasil untuk mengidentifikasi tema utama dan pola informasi. Aktivitas utama meliputi Transkripsi data, Koding dan Kategorisasi, Identifikasi tema utama, dan Validasi temuan.

4. **Pelaporan Hasil Wawancara**

   Tahapan terakhir adalah menyusun laporan berdasarkan hasil wawancara. Komponen utama laporan meliputi: Ringkasan temuan, Kutipan responden, dan Korelasi dengan Penelitian.

2.8.3 Studi Observasi

Studi observasi memberikan wawasan yang kaya tentang perilaku konsumen secara langsung. Dengan metode yang sistematis, data yang dihasilkan dari observasi dapat memperkuat analisis dalam penelitian ini, memberikan pemahaman yang mendalam tentang minat pembeli smartphone di Erafone Suzuya Baru Rantauprapat.

1. **Perencanaan Obervasi**

   Tahapan ini memastikan bahwa observasi dilakukan secara sistematis dan sesuai dengan tujuan penelitian. Aktivitas utama meliputi: Penentuan tujuan observasi, Pemilihan lokasi observasi, Penentuan aspek yang diamati, Jenis Observasi, dan Penyusunan Alat Observasi.

2. **Pelaksanaan Observasi**

   Pada tahap ini, peneliti melakukan pengamatan langsung di lokasi penelitian. Langkah-langkahnya meliputi: Pengamatan awal, Pelaksanaan observasi terencana, Pengumpulan data kuantitatif dan kualitatif, dan Dokumentasi.

3. **Analisis Hasil Observasi**

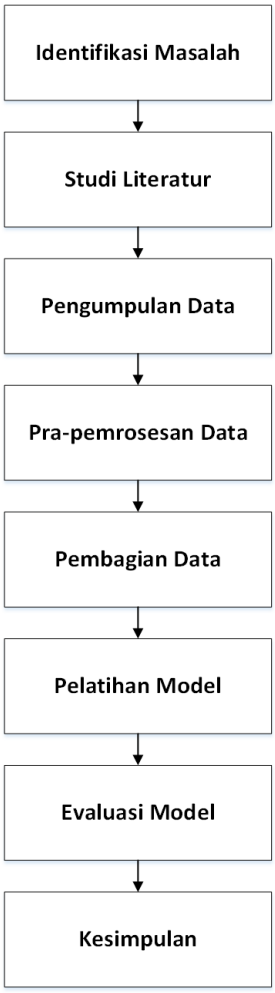
   Data yang telah dikumpulkan dianalisis untuk mengidentifikasi pola dan kesimpulan. Prosesnya meliputi Kategorisasi data, Identifikasi pola, dan validasi temuan.

4. **Pelaporan Hasil Observasi**

   Tahap terakhir adalah menyusun laporan berdasarkan temuan observasi. Komponen utama laporan meliputi Deskripsi situasi observasi, Ringkasan temuan utama, Kutipan atau contoh perilaku, dan Korelasi dengan penelitian.

2.8.4 Kerangka Kerja Penelitian

Kerangka kerja penelitian ini dirancang untuk memberikan panduan sistematis dalam menganalisis data konsumen guna menentukan minat pembeli smartphone di Erafone Suzuya Baru Rantauprapat.



Gambar 2.10 Kerangka Kerja Penelitian

Penjelasan Kerangka Kerja Penelitian:

1. **Identifikasi Masalah**

   Tahap awal penelitian ini dimulai dengan identifikasi permasalahan yang dihadapi oleh Erafone Suzuya Baru Rantauprapat terkait persaingan pasar smartphone dan perlunya strategi pemasaran berbasis data untuk memahami preferensi konsumen.

2. **Studi Literatur**

   Pada tahap ini, dilakukan pengumpulan informasi dari jurnal, buku, dan sumber lainnya terkait algoritma *Naïve Bayes*, SVM, dan penerapannya dalam analisis data konsumen.

3. **Pengumpulan Data**

   Data yang digunakan dalam penelitian ini adalah data transaksi penjualan smartphone di Erafone Suzuya Baru Rantauprapat. Data meliputi variabel demografi konsumen, produk yang dibeli, waktu pembelian, dan faktor lainnya.

4. **Pra-pemrosesan Data**

   Data yang dikumpulkan diolah untuk menghilangkan data yang tidak relevan, mengatasi nilai yang hilang, dan melakukan normalisasi agar data dapat digunakan oleh model *machine learning*.

5. **Pembagian Data**

   Data dibagi menjadi dua subset:

   1. Data Latih (*Training Data*): digunakan untuk melatih model.
   2. Data Uji (*Testing Data*): digunakan untuk mengevaluasi performa model.
   3. Proporsi data latih dan data uji ditetapkan sebesar 80:20.

6. **Pelatihan Model**

   Model *Naïve Bayes* dilatih menggunakan data latih untuk mempelajari pola preferensi pembeli. Model SVM juga dilatih menggunakan data latih yang sama untuk memastikan hasil perbandingan yang adil.

7. **Evaluasi Model**

   Model *Naïve Bayes* dievaluasi menggunakan data uji berdasarkan metrik evaluasi seperti akurasi, presisi, *recall*, dan F1-*score*. Model SVM dievaluasi dengan metrik yang sama seperti *Naïve Bayes* untuk membandingkan performa kedua algoritma.

8. **Kesimpulan**

   Hasil evaluasi kedua algoritma dibandingkan untuk menentukan algoritma yang lebih unggul dalam menentukan peminat pembeli smartphone. Berdasarkan hasil perbandingan kinerja, ditarik kesimpulan dan diberikan rekomendasi bagi Erafone terkait strategi pemasaran yang berbasis data.