

BAB II

TINJAUAN PUSTAKA

2.1. Studi Sebelumnya

Peneliti mengumpulkan hasil penelitian sebelumnya yang relevan. Penelitian-penelitian tersebut digunakan untuk menambah wawasan peneliti. Penelitian yang dikumpulkan berkaitan dengan Sistem Pendukung Keputusan atau Decision Support System. Dengan mengkaji penelitian sebelumnya, peneliti dapat memahami metode dan teknologi yang telah digunakan.

Pertama, Penelitian oleh Nata dan kawan-kawan merancang dan mendeploy Sistem Pendukung Keputusan berbasis web. Sistem ini menggunakan model klasifikasi berbasis machine learning untuk menentukan penerima Program Indonesia Pintar. Model yang digunakan dalam penelitian tersebut meliputi KNN, Naive Bayes, SVM, Boosting, dan Random Forest. Penelitian ini bertujuan untuk meningkatkan akurasi dalam penentuan penerima bantuan.

Kedua, Penelitian oleh Fahurian dan kawan-kawan merancang Sistem Pendukung Keputusan berbasis web untuk pengadaan obat. Sistem ini menggunakan metode regresi linier untuk mengatasi persediaan yang tidak terkontrol saat pemesanan. Ketidakseimbangan stok dapat menyebabkan kekurangan atau kelebihan obat di apotek. Penelitian ini mengembangkan SPK dengan database MySQL untuk memprediksi stok obat di Apotek Bina Farma. Hasil pengujian menunjukkan model mampu memberikan estimasi rata-rata 86,867, sehingga membantu pengadaan obat lebih akurat.

Ketiga, Penelitian oleh Nugroho dan kawan-kawan membangun Sistem Pendukung Keputusan untuk asesmen rehabilitasi narkoba. Sistem ini menggunakan algoritma Random Forest untuk memberikan rekomendasi asesmen rehabilitasi. SPK ini dikembangkan berbasis Rest API yang menghubungkan sistem dengan machine learning untuk BNN Kota Surabaya. Pengujian dilakukan dengan dua skenario, yaitu 100:100 dan 70:30. Hasil pengujian menunjukkan akurasi 61% pada skenario 100:100, sedangkan

pada skenario 70:30 hanya mencapai 25%, yang disebabkan oleh jumlah data yang terbatas.

Keempat, Penelitian oleh Nugroho dan kawan-kawan bertujuan untuk membangun Sistem Pendukung Keputusan (SPK) yang dapat memberikan rekomendasi asesmen rehabilitasi narkoba menggunakan algoritma Random Forest. Selain itu, penelitian ini juga bertujuan untuk 14 mengukur tingkat performa algoritma tersebut dalam penerapannya. SPK ini dikembangkan berbasis Rest API, yang memungkinkan sistem terhubung dengan machine learning untuk mendukung proses asesmen di BNN Kota Surabaya. Pengujian dilakukan dengan dua skenario, yaitu skenario 100:100 yang memperoleh akurasi 61%, serta skenario 70:30 yang hanya memperoleh akurasi 25%. Hasil penelitian menunjukkan bahwa performa sistem masih kurang optimal akibat jumlah data yang terbatas, sehingga perlu dilakukan pengembangan lebih lanjut untuk meningkatkan akurasi dan efektivitas rekomendasi asesmen rehabilitasi.

Kelima, Penelitian oleh Alan Chandra Darmawan berjudul "Pengembangan Aplikasi Berbasis Web dengan Python Flask untuk Klasifikasi Data Menggunakan Metode Decision Tree C4.5" bertujuan untuk mengembangkan aplikasi berbasis web yang mampu mengklasifikasi data menggunakan metode Decision Tree C4.5. Aplikasi ini dirancang agar dapat menerima input dalam format CSV, menampilkan hasil klasifikasi dalam bentuk diagram pohon keputusan, serta menyediakan fitur unduhan model pelatihan dalam format .pkl. Pengembangan dilakukan menggunakan Python Flask sebagai micro-framework web, dengan pengujian melalui metode Black Box Testing, Software Testing, dan User Acceptance Testing. Hasil pengujian menunjukkan bahwa aplikasi dapat mengklasifikasi data dengan akurasi tinggi dan menampilkan hasil dalam bentuk diagram pohon keputusan serta laporan klasifikasi. Dibandingkan dengan aplikasi WEKA, aplikasi ini menawarkan antarmuka berbasis web yang lebih user-friendly. Selain itu, aplikasi ini juga dirancang agar dapat digunakan oleh pengguna dengan maupun tanpa pengetahuan tentang machine learning, karena telah dilengkapi dengan petunjuk penggunaan dan glosarium istilah teknis. Aplikasi yang alanchandrad.pythonanywhere.com

Tabel 2. 1 Studi Sebelumnya

No	Penulis	Model	Tujuan	Hasil
1.	Nata, dkk.[6]	k-Nearest Neighbor , Naïve Bayes, SVM, Boosting, Random Forest	Menentukan model klasifikasi Terbaik dalam menganalisa data kelayakan penerima Program Indonesia Pintar dan merancang dan mendeployment Sistem Pendukung Keputusan dengan Model Klasifikasi Berkas Machine Learning dalam Penentuan Penerima Program Indonesia Pintar berbasis web.	Membuat Sistem Pendukung Keputusan (SPK) berbasis web untuk menentukan penerima Program Indonesia Pintar (PIP). Model Naïve Bayes memiliki akurasi tertinggi 93,3%, sehingga digunakan dalam sistem ini.
2.	Fahurian, dkk. [7]	Regresi Linier	Merancang Sistem pendukung keputusan terkait pengadaan obat menggunakan metode algoritma regresi linier untuk mengatasi	Mengembangkan SPK berbasis web dengan database MySQL untuk prediksi stok obat di Apotek Bina Farma. Hasil pengujian menunjukkan model mampu memberikan estimasi rata-rata 86,867, membantu pengadaan obat lebih akurat.

No	Penulis	Model	Tujuan	Hasil
			Persediaan yang tidak terkontrol saat pemesanan, yang dapat mengakibatkan kurang atau lebihnya stok barang	
3.	Nugroho, dkk [8]	Random Forest	a. Bagaimana membangun sistem pendukung keputusan dalam memberikan rekomendasi asesmen rehabilitasi narkotika menggunakan algoritma random forest? b. Bagaimana mengukur tingkat performa algoritma random forest pada penelitian ini? Batasan masalah	Membangun SPK asesmen rehabilitasi narkotika berbasis Rest API yang menghubungkan sistem dengan machine learning untuk BNN Kota Surabaya. Pengujian pada skenario 100:100 memperoleh akurasi 61%, sedangkan pada skenario 70:30 hanya 25%, menunjukkan performa yang kurang optimal akibat jumlah data yang terbatas.

No	Penulis	Model	Tujuan	Hasil
			pada penelitian ini sebagai berikut	
4.	Nugroho, dkk [9]	Random Forest	Membangun SPK yang dapat memberikan rekomendasi asesmen rehabilitasi narkotika menggunakan algoritma Random Forest. Dan Mengukur tingkat performa algoritma Random Forest dalam penelitian ini.	performa sistem masih kurang optimal akibat jumlah data yang terbatas, sehingga perlu dilakukan pengembangan lebih lanjut untuk meningkatkan akurasi dan efektivitas rekomendasi asesmen rehabilitasi.
5.	Darmawan [10]	Decision Tree	Penelitian ini bertujuan untuk mengembangkan aplikasi berbasis web yang dapat mengklasifikasi data menggunakan metode Decision Tree C4.5. Aplikasi ini dirancang agar mampu menerima	aplikasi dapat mengklasifikasi data dengan akurasi tinggi dan menampilkan hasil dalam bentuk diagram pohon keputusan serta laporan klasifikasi. Dibandingkan dengan aplikasi WEKA, aplikasi ini menawarkan antarmuka berbasis web yang lebih user-friendly. Selain itu, aplikasi ini juga dirancang agar dapat digunakan oleh pengguna dengan maupun

No	Penulis	Model	Tujuan	Hasil
			input berupa file CSV, menampilkan hasil klasifikasi dalam bentuk diagram pohon keputusan, serta menyediakan fitur unduhan model pelatihan dalam format	tanpa pengetahuan tentang machine learning, karena telah dilengkapi dengan petunjuk penggunaan dan glosarium istilah teknis. Aplikasi yang dikembangkan dapat diakses melalui

2.2. Dasar Teori

2.2.1. Sistem Pendukung Keputusan

Sistem pendukung keputusan atau beberapa orang mengenal dengan istilah Decision Support System merupakan sebuah sistem yang digunakan untuk membantu kita dalam membuat keputusan yang menguntungkan suatu bisnis atau perusahaan[11]. Pengambilan keputusan yang dilakukan oleh sistem menggunakan data-data sebelumnya, model, perhitungan dan analisis tertentu. SPK dapat dikembangkan menggunakan berbagai teknologi, seperti pemrosesan bahasa alami, data mining, kecerdasan buatan, serta machine learning. Dengan adanya SPK, pengambil keputusan dapat memperoleh informasi yang lebih akurat dan relevan dalam waktu singkat. Hal ini memungkinkan proses pengambilan keputusan yang lebih cepat, tepat, dan efektif, sehingga meningkatkan efisiensi dan kualitas keputusan yang diambil[12].

Secara mendasar, proses pengambilan keputusan yang didukung oleh sistem ini dapat diuraikan ke dalam beberapa tahapan, sebagaimana dikemukakan oleh Simon yang membaginya menjadi tiga fase utama: kecerdasan (pemahaman), perancangan (desain), dan pemilihan (pilihan). Tahap kecerdasan atau pemahaman adalah proses penyelidikan di mana pengambil keputusan mengamati realitas untuk memahami

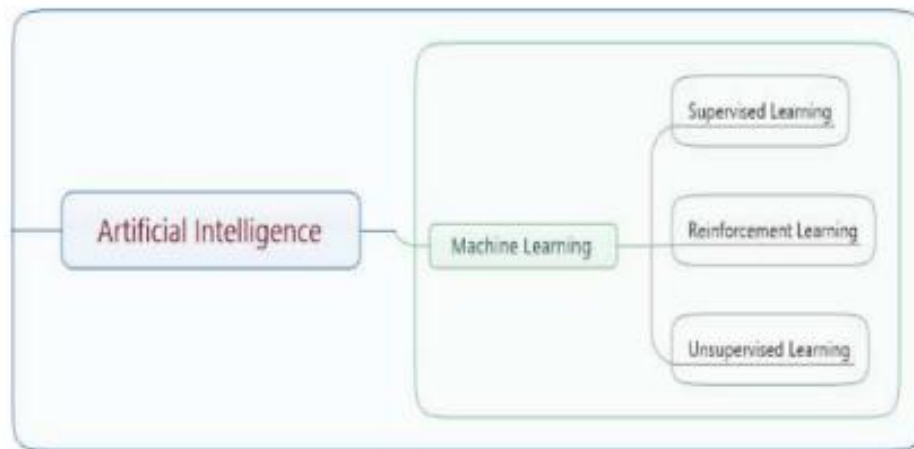
domain masalah dan mengumpulkan informasi yang diperlukan melalui pemeriksaan data. Selanjutnya, pada tahap perancangan, sistem akan membantu mengembangkan berbagai kriteria dan alternatif keputusan dengan membuat model-model berdasarkan data yang ada untuk memprakarsai pemecahan masalah. Tahap terakhir adalah pemilihan, di mana sistem menyajikan hasil rancangan dalam bentuk yang paling efektif untuk mendorong pengambilan keputusan, yang kemudian dilanjutkan dengan tahap penerapan atau implementasi[13].

2.2.2. Machine Learning

Machine Learning merupakan sebuah teknologi dan sub-ilmu dari kecerdasan buatan (Artificial Intelligence atau AI) yang memungkinkan komputer untuk belajar secara mandiri dari data tanpa perlu diprogram secara eksplisit. Dengan menggunakan algoritma yang tepat, sistem dapat mengenali pola, membuat prediksi, serta meningkatkan performa seiring bertambahnya data yang digunakan dalam proses pelatihan. Salah satu keunggulan utama machine learning adalah kemampuannya untuk menyesuaikan diri dengan data baru, sehingga sistem dapat terus berkembang dan memberikan hasil yang lebih akurat dari waktu ke waktu. Dalam penerapannya, machine learning terbagi menjadi beberapa kategori utama, yaitu Supervised Learning, Unsupervised Learning, Semi-Supervised Learning, dan Reinforcement Learning[14]

Kemampuan machine learning untuk belajar dari data secara mandiri menjadi landasan bagi evolusi Sistem Pendukung Keputusan (SPK) atau Decision Support System (DSS)[15]. Sistem ini dirancang untuk mengatasi tingginya jumlah data dan kompleksitas analisis dalam berbagai bidang, termasuk kesehatan, sehingga dapat mempermudah proses diagnosis penyakit dan pengambilan keputusan klinis. SPK modern sering kali dikategorikan sebagai sistem berbasis pengetahuan (knowledge-driven), yang secara khusus memanfaatkan paradigma dari kecerdasan buatan seperti Jaringan Syaraf Tiruan (Artificial Neural Networks) untuk memberikan keahlian dan informasi spesifik kepada pengambil keputusan[16]. Dalam praktiknya, berbagai algoritma machine learning seperti Logistic Regression, Random Forest, K-Nearest Neighbor (KNN), dan Support

Vector Machine (SVM) diterapkan dalam SPK untuk menganalisis data secara efektif dan efisien.

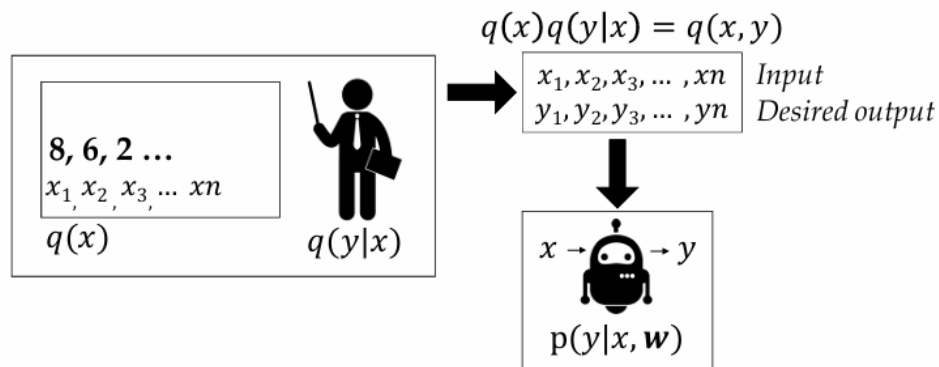


Gambar 2.1 Skema AI

2.2.2.1. Supervised Learning

Supervised learning adalah metode machine learning yang menggunakan data berlabel, di mana setiap data memiliki input dan output yang diketahui. Tujuan dari metode ini adalah membangun model yang mampu memprediksi output dari data baru berdasarkan pola yang dipelajari dari data sebelumnya. Proses pelatihan dilakukan dengan mencocokkan input terhadap output, sehingga model belajar memahami hubungan di antara keduanya. Metode ini terbagi menjadi dua jenis utama, yaitu klasifikasi dan regresi.

Klasifikasi digunakan saat output berupa kategori, seperti "ya" atau "tidak", sedangkan regresi digunakan untuk memprediksi nilai numerik, seperti jumlah penjualan atau harga. Untuk menghindari overfitting dan menilai performa model, data biasanya dibagi menjadi data training dan data testing. Algoritma yang umum digunakan dalam supervised learning antara lain decision tree, random forest, support vector machine (SVM), dan k-nearest neighbor (KNN). Metode ini banyak diterapkan dalam berbagai bidang seperti deteksi penyakit, prediksi permintaan pasar, dan klasifikasi teks[14].



Gambar 2. 2 Supervised Learning

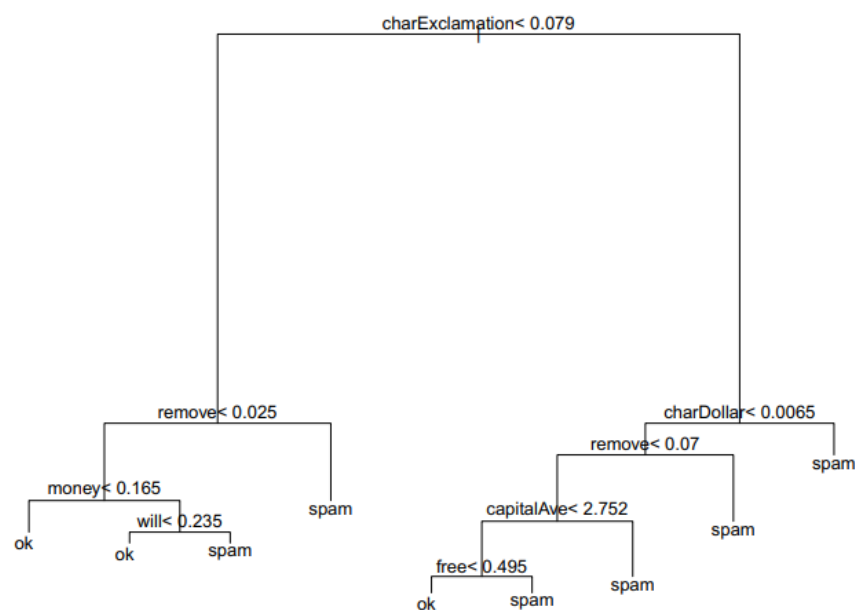
2.2.2.2. Training, Validation, and Testing

Dalam pengembangan model machine learning, data umumnya dibagi menjadi tiga bagian: training, validation, dan testing. Data training digunakan untuk melatih model agar mengenali pola dari data yang tersedia. Validation set digunakan untuk menyetel parameter model dan mencegah overfitting selama proses pelatihan. Sementara itu, testing set digunakan untuk menguji performa akhir model terhadap data yang belum pernah dilihat sebelumnya, guna menilai kemampuan generalisasi model. Pembagian ini umum digunakan dalam supervised learning. Untuk metode unsupervised seperti FP-Growth, pembagian ini biasanya tidak diperlukan karena tidak ada label yang diprediksi[14].

2.2.2.3. Random Forest

Random Forest adalah algoritma *machine learning* berbasis *ensemble* yang sangat efektif untuk tugas klasifikasi dan regresi. Prinsip kerjanya adalah dengan membangun banyak pohon keputusan (*decision trees*) di mana setiap pohon dilatih menggunakan subset data yang berbeda. Hasil akhir kemudian ditentukan melalui *voting* (pemungutan suara) dari mayoritas pohon untuk klasifikasi, atau rata-rata dari semua pohon untuk regresi. Pendekatan kolektif ini secara signifikan meningkatkan akurasi prediksi dan mengurangi risiko *overfitting*, menjadikannya lebih unggul

dibandingkan pohon keputusan tunggal. Berbagai penelitian telah menunjukkan keandalan Random Forest dalam menangani data yang kompleks, tidak terstruktur, dan berdimensionalitas tinggi, seperti data kategorikal yang telah melalui proses *one-hot encoding*. Dalam aplikasi praktis, seperti sistem pendukung keputusan untuk lokasi pengelasan, algoritma ini mampu menganalisis berbagai faktor lingkungan fisik untuk memberikan rekomendasi yang stabil dan akurat berdasarkan pola data yang dipelajari.[17]



Gambar 2. 3 Random Forest

2.2.2.4. Decision Tree

Decision Tree adalah metode klasifikasi yang memprediksi kategori dari sebuah kelas untuk kemudian dikelompokkan ke dalam kelas yang telah tersedia. Model ini memiliki stuktur berbentuk pohon di mana leaf node menunjukkan nilai kelas target, sedangkan decision node menentukan pengujian yang akan dilakukan terhadap atribut. Salah satu algoritma populer dalam pembentukan Decision Tree adalah C4.5, yang merupakan pengembangan dari algoritma ID3. Perhitungan dengan C4.5 melibatkan

parameter seperti Entropi, Info Gain, Split, dan Gain Ratio untuk membangun pohon keputusan[18].

2.2.2.5. Logistic Regression

Logistic Regression (LR) adalah teknik klasifikasi kategorikal yang digunakan untuk memetakan fitur input, baik kualitatif maupun kuantitatif, ke sebuah variabel target yang coba diprediksi. Sebagai metode supervised learning, model ini menghasilkan sebuah koefisien untuk setiap variabel prediktor, yang kemudian dapat digunakan untuk memperkirakan odds ratio (OR). Koefisien ini merepresentasikan perubahan pada variabel dependen untuk setiap satu unit perubahan pada variabel prediktor, dengan asumsi prediktor lain dalam model tetap konstan. Model ini menjadi pilihan yang lebih disukai dalam konteks medis karena menyediakan metrik statistik seperti OR, yang lebih mudah untuk dipahami oleh para dokter dan staf medis[19].

2.2.2.6. KNN

K-Nearest Neighbor (KNN) merupakan salah satu algoritma supervised learning yang digunakan untuk menyelesaikan masalah klasifikasi data. Tujuan utama dari algoritma ini adalah untuk mengklasifikasikan objek baru berdasarkan atribut dan sampel dari data latih yang ada. Prinsip kerjanya adalah dengan mencari jarak terdekat antara data yang akan dievaluasi dengan tetangga terdekatnya dalam data latih. KNN dianggap sebagai salah satu algoritma paling sederhana untuk memecahkan masalah klasifikasi dan sering kali mampu menghasilkan hasil yang kompetitif dan signifikan[18].

2.2.2.7. Membandingkan 4 Model

Tabel 2. 2 Perbandingan Model

Kriteria	Logistic Regression	Random Forest	Decision Tree	KNN
Jenis Model	Statistik & Supervised Learning	Berbasis Pohon & Supervised Learning	Ensemble Learning & Supervised Learning	Berbasis Jarak & Supervised Learning
Prinsip Kerja	Memetakan fitur ke probabilitas menggunakan fungsi logistik	Membangun struktur aturan hierarkis seperti pohon	Menggabungkan banyak pohon keputusan dan melakukan voting	Mengklasifikasikan berdasarkan mayoritas 'k' tetangga terdekat
Proses Pembelajaran	Mengoptimalkan koefisien pada fase pelatihan	Membagi data secara rekursif menggunakan metrik seperti Info Gain	Setiap pohon dilatih pada sampel data acak (bootstrapping)	Tidak ada fase pelatihan eksplisit; kalkulasi dilakukan saat prediksi
Interpretasi	Sangat Tinggi; hasil seperti Odds Ratio mudah dijelaskan	Sangat Tinggi; alur keputusan dapat divisualisasikan	Rendah; bekerja seperti "kotak hitam" (black box)	Sedang; konsepnya sederhana namun penjelasannya bisa kompleks
Kelebihan Utama	Cepat, sederhana, dan sangat mudah diinterpretasikan	Mudah dipahami dan divisualisasikan	Akurasi tinggi dan tahan terhadap overfitting	Sangat sederhana untuk diimplementasikan
Kelemahan Utama	Mengasumsikan hubungan linear; kurang	Rentan terhadap	Membutuhkan banyak sumber daya komputasi	Lambat pada dataset besar

	baik untuk masalah non-linear	overfitting dan bisa tidak stabil		
--	-------------------------------	-----------------------------------	--	--

2.2.2.8. Evaluasi

Akurasi (Accuracy) adalah metrik evaluasi yang paling dasar dalam machine learning. Akurasi menunjukkan seberapa besar proporsi prediksi model yang benar dibandingkan dengan keseluruhan jumlah data. Dengan kata lain, akurasi menghitung berapa banyak prediksi benar (baik positif maupun negatif) dibagi total jumlah prediksi. Akurasi sangat cocok digunakan ketika data memiliki distribusi kelas yang seimbang. Rumus perhitungannya adalah:

$$akurasi = \frac{TP + TN}{TP + TN + FP + FN}$$

- TP (True Positive): Prediksi positif dan benar
- TN (True Negative): Prediksi Negatif dan benar
- FP (False Positive): Prediksi Positif tapi salah
- FN (False Negative): Prediksi Negatif tapi salah

Precision adalah metrik yang mengukur seberapa akurat prediksi positif yang dilakukan oleh model. Metrik ini menunjukkan proporsi data yang diprediksi sebagai positif dan memang benar-benar positif. Precision sangat penting ketika kesalahan dalam memprediksi sesuatu sebagai positif harus dihindari, misalnya dalam sistem pendeteksi spam atau diagnosis penyakit. Precision dihitung dengan rumus:

$$Precision = \frac{TP}{TP + FP}$$

- TP (True Positive): Prediksi positif dan benar

- TN(True Negative): Prediksi Negatif dan benar
- FP(Flase Positive): Prediksi Positif tapi salah
- FN(Flase Negative): Prediksi Negatif tapi salah

Recall (atau dikenal juga sebagai sensitivity atau true positive rate) adalah metrik yang menunjukkan seberapa banyak data positif yang berhasil ditemukan oleh model. Recall menjadi sangat penting dalam kasus ketika kegagalan mendeteksi data positif (false negative) harus diminimalkan, seperti dalam sistem deteksi kanker atau alarm kebakaran. Rumus recall adalah:

$$Recall = \frac{TP}{TP + FN}$$

- TP (True Positive): Prediksi positif dan benar
- TN(True Negative): Prediksi Negatif dan benar
- FP(Flase Positive): Prediksi Positif tapi salah
- FN(Flase Negative): Prediksi Negatif tapi salah

F1-Score adalah rata-rata harmonik dari precision dan recall. Metrik ini digunakan ketika dibutuhkan keseimbangan antara precision dan recall, terutama saat data tidak seimbang. F1-score sangat berguna dalam situasi di mana peningkatan satu metrik akan menurunkan metrik lainnya, sehingga dibutuhkan kombinasi yang optimal. F1-score dihitung dengan rumus[14]:

$$akurasi = \frac{Precision \times Recall}{Precision + Recall}$$

- TP (True Positive): Prediksi positif dan benar
- TN(True Negative): Prediksi Negatif dan benar
- FP(Flase Positive): Prediksi Positif tapi salah
- FN(Flase Negative): Prediksi Negatif tapi salah

2.2.3. Database

Database merupakan komponen penting dalam sistem informasi yang berfungsi untuk menyimpan, mengelola, dan mengorganisir data secara efisien. Dalam penelitian sebelumnya, berbagai sistem telah mengadopsi database relasional seperti MySQL serta database non-relasional seperti MongoDB untuk memastikan performa yang optimal sesuai kebutuhan sistem. Database memungkinkan penyimpanan data secara terstruktur sehingga mempermudah proses pencarian, pengolahan, serta analisis informasi. Pada sistem pendukung keputusan untuk lokasi pengelasan, database digunakan untuk menyimpan data terkait lingkungan fisik, hasil analisis machine learning, serta informasi pengguna yang berinteraksi dengan sistem. Dengan pemanfaatan database yang tepat, sistem dapat berjalan secara lebih efisien dan mendukung pengambilan keputusan berbasis data yang akurat[20].

2.2.4. UI/UX

UI (User Interface) dan UX (User Experience) merupakan aspek penting dalam pengembangan sistem yang berorientasi pada pengguna. UI berfokus pada tampilan visual dan interaksi antarmuka, sedangkan UX lebih menitikberatkan pada pengalaman pengguna dalam menggunakan sistem. Penelitian sebelumnya menunjukkan bahwa desain UI/UX yang baik dapat meningkatkan keterlibatan pengguna, mempermudah navigasi, serta meningkatkan efisiensi dalam penggunaan aplikasi[21]. Dalam konteks sistem pendukung keputusan, desain UI/UX yang intuitif dan responsif memungkinkan pengguna untuk dengan mudah mengakses informasi yang dibutuhkan serta memahami hasil analisis yang disajikan. Dengan pendekatan desain berbasis pengguna, sistem dapat memberikan pengalaman yang lebih nyaman dan efektif dalam mendukung pengambilan keputusan[22].

2.2.5. Front-end

Front-end merupakan bagian dari sistem yang berinteraksi langsung dengan pengguna, mencakup elemen visual dan fungsionalitas yang memungkinkan pengguna berinteraksi dengan aplikasi. Penelitian sebelumnya menunjukkan bahwa pengembangan front-end yang baik harus mempertimbangkan aspek performa, responsivitas, serta aksesibilitas. Teknologi seperti HTML, CSS, JavaScript, dan framework seperti React atau Vue.js sering digunakan untuk membangun tampilan antarmuka yang dinamis dan menarik. Dalam sistem pendukung keputusan untuk lokasi pengelasan, front-end bertanggung jawab dalam menyajikan data hasil analisis machine learning secara interaktif dan mudah dipahami. Dengan desain yang user-friendly, pengguna dapat lebih cepat memahami informasi yang disajikan serta mengambil keputusan yang lebih tepat[23].

2.2.6. Back-end

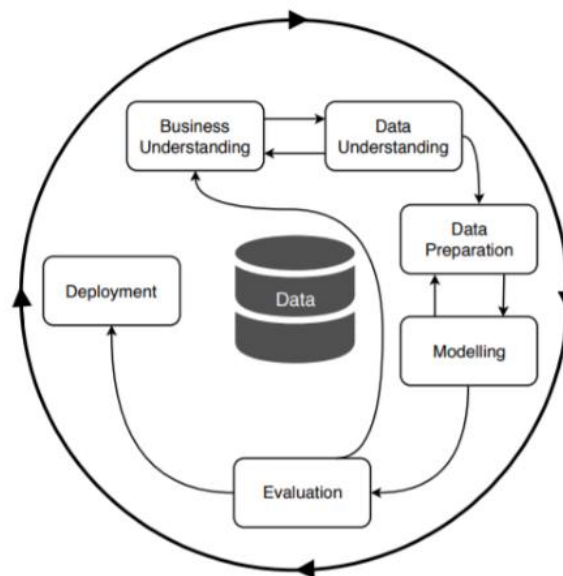
Back-end merupakan bagian dari sistem yang bertanggung jawab dalam mengelola logika bisnis, pemrosesan data, serta komunikasi dengan database. Penelitian sebelumnya menunjukkan bahwa arsitektur back-end yang baik harus mampu menangani permintaan pengguna dengan cepat dan efisien, serta menjamin keamanan data[24]. Teknologi yang sering digunakan dalam pengembangan back-end meliputi Python dengan framework Flask atau Django. Dalam sistem pendukung keputusan untuk lokasi pengelasan, back-end berperan dalam menerima input dari pengguna, memproses data dengan model machine learning, serta mengirimkan hasil analisis ke front-end untuk ditampilkan. Dengan implementasi back-end yang optimal, sistem dapat bekerja dengan lancar, memberikan respons yang cepat, serta mendukung skalabilitas untuk pengolahan data yang lebih besar[25].

2.2.7. Cross Industry Standard Process for Data Mining (CRISP-DM)

CRISP-DM (Cross Industry Standard Process for Data Mining) adalah sebuah metodologi standar yang digunakan untuk mengelola proyek data mining dan data

science. Metode ini dikembangkan pada tahun 1996 oleh beberapa perusahaan besar yaitu Integral Solutions Ltd (ISL), Teradata, Daimler AG, NCR Corporation dan OHRA untuk menyediakan kerangka kerja yang sistematis dan terstruktur. CRISP-DM dirancang agar dapat digunakan di berbagai macam industri, sehingga tidak bergantung pada satu bidang saja. Dengan mengikuti langkah-langkah CRISP-DM, proyek data dapat berjalan lebih terarah dan berfokus pada tujuan bisnis.

CRISP-DM memiliki enam tahap utama, yaitu Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, dan Deployment. Tahapan ini tidak selalu berjalan linier; sering kali prosesnya iteratif, artinya kita mungkin perlu kembali ke tahap sebelumnya jika ada yang perlu diperbaiki. Setiap tahap memiliki tujuan spesifik yang mendukung kelancaran proyek data dari awal sampai hasil akhirnya bisa digunakan oleh bisnis[26].



Gambar 2. 4 Framework CRISP-DM

2.2.7.1. Business Understanding

Tahap pertama, Business Understanding, bertujuan untuk memahami secara mendalam apa yang menjadi kebutuhan dan tujuan bisnis. Kita tidak langsung fokus

pada data, melainkan bertanya: masalah bisnis apa yang ingin diselesaikan? Dengan pemahaman yang tepat tentang kebutuhan bisnis, maka seluruh proses data mining akan menjadi lebih relevan dan efektif. Contohnya, perusahaan mungkin ingin mengetahui cara meningkatkan retensi pelanggan.

2.2.7.2. Data Understanding

Tahap kedua adalah Data Understanding. Di sini, kita mulai mengumpulkan data yang diperlukan dan melakukan eksplorasi awal. Tujuannya adalah mengenali kualitas, isi, dan struktur data yang dimiliki. Kita akan mencari pola-pola awal, anomali, dan hal-hal penting lain yang mungkin mempengaruhi langkah selanjutnya. Misalnya, mengecek apakah ada data yang hilang atau outlier yang mencurigakan.

2.2.7.3. Data Preparation

Tahap ketiga, Data Preparation, merupakan proses menyiapkan data untuk modeling. Ini meliputi pembersihan data, transformasi fitur, pembuatan variabel baru, penggabungan data dari berbagai sumber, dan lain-lain. Data yang baik dan bersih adalah fondasi model yang baik. Tahap ini biasanya memakan waktu paling banyak, sekitar 70-80% dari seluruh proyek.

2.2.7.4. Modeling

Selanjutnya, di tahap Modeling, kita mulai membangun model prediktif menggunakan algoritma statistik atau machine learning. Model ini dibuat berdasarkan data yang telah dipersiapkan. Teknik modeling harus dipilih dengan tepat sesuai jenis data dan tujuan bisnis. Kita juga perlu mengatur parameter model agar performanya optimal. Setelah model dibuat, kita lanjutkan ke tahap berikutnya.

2.2.7.5. Evaluation

Tahap Evaluation bertujuan untuk menilai seberapa baik model yang telah dibangun dalam memenuhi tujuan bisnis. Kita menggunakan metrik evaluasi seperti

Accuracy, Precision, Recall, F1-Score, atau AUC-ROC untuk mengukur performa model. Jika model belum memuaskan, kita bisa kembali ke tahap Modeling atau bahkan Data Preparation untuk memperbaiki.

2.2.7.6. Deployment

Terakhir adalah tahap Deployment. Ini adalah proses mengimplementasikan hasil proyek ke dalam operasional nyata bisnis. Deployment bisa berupa memasukkan model ke dalam sistem produksi, membuat laporan analisis, atau memberikan rekomendasi kepada manajemen. Selain itu, perlu juga merencanakan monitoring performa model secara berkala agar tetap akurat dan relevan dengan kondisi bisnis yang dinamis [26].

