BAB II
TINJAUAN PUSTAKA

2.1 Deskripsi Teoritik

2.1.1 *Data mining*

Data mining adalah proses untuk mengidentifikasi pola dan pengetahuan menarik dari kumpulan data yang besar. Sumber data yang digunakan bisa berasal dari *database, data warehouse, web*, dan tempat penyimpanan informasi lainnya [7]. *Data mining*, yang juga dikenal sebagai *Knowledge Discovery in Database (KDD)*, mencakup aktivitas mengumpulkan dan menggunakan data historis untuk menemukan keteraturan, pola, atau hubungan alami dalam data berukuran besar [8].

Metode dalam *data mining* dapat dikategorikan menjadi dua jenis, yaitu deskriptif dan prediktif. Metode deskriptif digunakan untuk menemukan pola-pola yang mudah dipahami oleh manusia dan yang menggambarkan karakteristik data. Sebaliknya, metode prediktif digunakan untuk membuat model pengetahuan yang dapat digunakan untuk melakukan prediksi .

Terdapat beberapa teknik dalam *data mining* berdasarkan jenis tugas yang dapat dilakukan [8], yaitu :

1. Klasifikasi :

Klasifikasi adalah proses untuk menemukan serangkaian model yang menggambarkan kelas-kelas data sehingga model tersebut dapat digunakan untuk memprediksi kelas yang belum diketahui dari suatu objek. Untuk memperoleh model ini, analisis dilakukan terhadap data latih, sedangkan data uji digunakan untuk mengukur tingkat akurasi dari model yang dihasilkan. Klasifikasi dapat digunakan untuk memprediksi nama atau nilai suatu objek data.

2. Klastering:

Klastering adalah proses pengelompokan data tanpa label kelas ke dalam beberapa kelompok tertentu berdasarkan tingkat kemiripannya.

3. Prediksi:

Model prediksi digunakan untuk meramalkan nilai-nilai dimasa depan berdasarkan data yang ada saat ini.

4. Estimasi:

Estimasi digunakan untuk menghasilkan perkiraan nilai yang tidak diketahui dalam data berdasarkan pola yang terlihat pada data yang ada.

5. Asosiasi:

Metode ini bertujuan untuk menghasilkan aturan-aturan yang menjelaskan hubungan kuat antara data yang berbeda.

2.1.2 Klasifikasi

Klasifikasi adalah proses analisis data yang mengekstrak model-model yang menggambarkan kelas-kelas data penting. Model-model ini, yang disebut sebagai classifier, digunakan untuk memprediksi label kelas kategorikal dari data yang diberikan [7].

Proses klasifikasi didasarkan pada empat komponen fundamental [9]:

1. Kelas

Variabel dependen dalam model adalah variabel kategorikal yang menunjukkan 'label' yang diberikan pada objek setelah proses klasifikasi. Contoh dari kelas tersebut meliputi: status loyalitas pelanggan, klasifikasi bintang (galaksi), atau kategori gempa bumi (badai), dan sebagainya.

2. Prediktor

Variabel independen dalam model adalah karakteristik (atribut) data yang digunakan untuk melakukan klasifikasi dan yang mempengaruhi hasil klasifikasi. Contoh dari prediktor ini antara lain kebiasaan merokok, konsumsi alkohol, tekanan darah, frekuensi pembelian, status pernikahan, fitur gambar satelit, catatan geologi tertentu, arah dan kecepatan angin, musim, serta lokasi kejadian fenomena, dan lain-lain.

3. Dataset Latih

Merupakan kumpulan data yang mencakup nilai untuk kedua komponen sebelumnya, dan digunakan untuk 'melatih' model agar dapat mengenali kelas yang tepat berdasarkan prediktor yang tersedia. Contoh dataset ini meliputi: kelompok pasien yang diuji untuk risiko serangan jantung, kelompok pelanggan dari sebuah supermarket yang dianalisis melalui survei internal, serta basis data yang berisi gambar untuk pemantauan dan pelacakan objek astronomi menggunakan teleskop.

4. Dataset pengujian

Dataset pengujian berisi data baru yang akan diklasifikasikan oleh model classifier yang telah dibangun, serta memungkinkan evaluasi akurasi klasifikasi dan kinerja model.

2.1.3 *Recursive Feature Elimination*

Metode *Recursive Feature Elimination* (*REF*) pada dasarnya bekerja dengan cara melakukan proses rekursif berdasarkan hasil pemeringkatan fitur berdasarkan nilai kepentingannya terhadap sebuah proses prediksi. Pada setiap iterasinya, setiap fitur akan diukur tingkat kepentingannya, kemudian sekumpulan fitur yang dianggap kurang relevan akan dihilangkan dan berdasarkan kumpulan fitur yang tersisa akan dilatih model klasifikasi yang baru [10]. *Metode RFE* merupakan salah satu metode eliminasi fitur secara otomatis. *Metode RFE* menjadi relevan untuk digunakan ketika jumlah fitur yang digunakan untuk melatih sebuah model memiliki jumlah yang besar sehingga proses eliminasi fitur secara manual sulit untuk dilakukan.

2.1.4 *Naïve Bayes*

Algoritma *Naive Bayes* merupakan salah satu algoritma yang sering di implementasikan dalam teknik klasifikasi. Algoritma ini mengaplikasikan metode probabilitas dan statistik yang dikembangkan oleh ilmuwan Inggris, Thomas Bayes. Metode ini memperkirakan kemungkinan kejadian di masa depan berdasarkan data dan pengalaman dari masa lalu, sehingga dikenal dengan nama Teorema Bayes.

Teorema ini digabungkan dengan asumsi Naive, di mana diasumsikan bahwa kondisi antara masing-masing atribut adalah independen satu sama lain. Dalam klasifikasi model *Naive Bayes*, diasumsikan bahwa keberadaan atau ketidakhadiran suatu ciri dalam sebuah kelas tidak berkaitan dengan ciri-ciri pada kelas lainnya [11].

Persamaan/rumus yang digunakan pada teorema Bayes adalah [11]:

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

Keterangan :

- X : Data dengan kelas yang belum diketahui
- H : Hipotesis data X merupakan suatu kelas spesifik
- $P(H|X)$: Probabilitas hipotesis H berdasar kondisi X (posteriori probability)
- $P(H)$: Probabilitas hipotesis H (prior probability)
- $P(X|H)$: Probabilitas X berdasarkan kondisi pada hipotesis H
- $P(X)$: Probabilitas X

Proses klasifikasi membutuhkan sejumlah indikator untuk menentukan kelas yang sesuai bagi sampel yang akan dianalisis. Oleh karena itu, teorema Bayes diadaptasi menjadi sebagai berikut:

$$P(C|F1 \ldots Fn) = \frac{\mathrm{P(C)P(F1 \ldots .Fn\,|C)}}{\mathrm{P(F1.... Fn)}}$$

Variabel C merepresentasikan kelas, sedangkan variabel F1...Fn menggambarkan karakteristik yang diperlukan untuk melakukan klasifikasi. Persamaan tersebut menunjukkan bahwa probabilitas sebuah sampel dengan karakteristik tertentu termasuk dalam kelas C (Posterior) adalah hasil dari perkalian antara probabilitas awal kemunculan kelas C (prior), dengan probabilitas kemunculan karakteristik sampel dalam kelas C (likelihood), kemudian dibagi dengan probabilitas kemunculan karakteristik sampel secara umum (evidence). Oleh karena itu, rumus tersebut dapat disederhanakan sebagai berikut:

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

Nilai Evidence tetap sama untuk setiap kelas pada satu sampel. Nilai posterior tersebut nantinya akan dibandingkan dengan nilai-nilai posterior dari kelas lainnya untuk menentukan kelas mana yang cocok untuk suatu sampel. Penjelasan lebih lanjut dari rumus atau persamaan Bayes dilakukan dengan menguraikan $P(C|F_1,..,F_n)$ menggunakan persamaan sebagai berikut:

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

Dapat dilihat bahwa penjelasan tersebut menambah jumlah dan kompleksitas faktor-faktor yang mempengaruhi nilai probabilitas, sehingga menyulitkan analisis setiap faktor secara terpisah. Oleh karena itu, asumsi independensi yang tinggi (naif) perlu digunakan, di mana setiap petunjuk ($\boldsymbol{F_1}$, $\boldsymbol{F_2}$ … $\boldsymbol{F_n}$) saling bebas (independen) antara satu dengan lainnya. Dengan asumsi tersebut, maka berlaku suatu persamaan :

$$\boldsymbol{P(Pi\ |Fj)} = \frac{\mathrm{P(F1 \cap Fj)}}{\mathrm{P(Fj)}} = \frac{\mathrm{P(Fi\ )P(Fj)}}{\mathrm{P(Fj)}} = \boldsymbol{P(F_i)}$$

Untuk $\boldsymbol{i \neq j}$, sehingga

$\boldsymbol{P(F_i\ |C, F_j)} = \boldsymbol{P(F_i\ |C)}$

Selanjutnya, penjabaran $\boldsymbol{P(C|F1, \ldots, Fn)}$ dapat disederhanakan menjadi :

$$\boldsymbol{P(C|F_1, \ldots, F_n)} = \boldsymbol{P(C)} \coprod_{i=1}^{n} P(F_i|C)$$

Rumus atau persamaan tersebut adalah model dari Teorema *Naive Bayes* yang akan diterapkan dalam proses klasifikasi. Sedangkan rumus Densitas Gauss digunakan untuk klasifikasi dengan data kontinu :

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

Keterangan :

P : Peluang

X_i : Atribut ke i

X_i : Nilai atribut ke i

Y : Kelas yang dicari

Y_j : Sub kelas Y yang dicari

μ : Mean, menyatakan rata-rata dari seluruh atribut

σ : Deviasi standar, menyatakan varian dari seluruh atribut

2.1.5 *Random Forest*

Random Forest adalah salah satu algoritma supervised learning yang digunakan untuk menganalisis dan memprediksi data. Sebagai *algoritma ensemble*, *Random Forest* menggabungkan hasil dari berbagai pohon keputusan sebagai classifier dasar yang dibangun dan digabungkan.[12].

Random Forest menggunakan pohon keputusan sebagai pengklasifikator dasar dan menghasilkan beberapa pohon keputusan untuk membentuk ensemble. Proses randomisasi dalam *Random Forest* terjadi dalam dua cara yaitu pertama, pengambilan sampel data secara acak untuk bootstrap samples, dan kedua, pemilihan atribut input secara acak untuk menghasilkan pohon keputusan dasar individual [13].

Random Forest dapat beroperasi dengan efisien pada basis data besar, mampu menangani ribuan variabel input tanpa menghapus variabel, memberikan estimasi terhadap variabel penting, menghasilkan estimasi internal yang tidak bias dari kesalahan generalisasi seiring pertumbuhan, memiliki metode yang cukup efektif untuk mengestimasi data hilang, serta mempertahankan performa akurasi meskipun proporsi data yang hilang besar. Selain itu, sifat paralel inheren dari *Random Forest* telah memungkinkan implementasi paralelnya menggunakan multithreading, multi-core, dan arsitektur paralel [13].

2.1.6 *K-Nearest Neighbor*

K-Nearest Neighbor adalah metode klasifikasi yang berdasarkan pada kedekatan jarak antara data yang satu dengan yang lainnya [22]. Dalam *K-Nearest Neighbor*, nilai K mengacu pada jumlah data terdekat dari data uji. Karena kesederhanaannya dalam proses klasifikasi, metode *K-Nearest Neighbor* menjadi salah satu metode pengenalan pola yang umum dan sering digunakan. Cara kerja *K-Nearest Neighbor* melibatkan pencarian jarak antara dua titik, yaitu titik pelatihan dan titik uji, kemudian penilaian dilakukan dengan K tetangga terdekat dari data pelatihan. Pada penelitian ini, jarak akan diukur menggunakan Euclidean distance. Rumus Euclidean distance ditunjukkan pada persamaan berikut [23].

$$d_{(xixj} = \sqrt{\sum_{r}^{n} (xi - xj)^{2}}$$

Ada beberapa faktor yang dapat mempengaruhi hasil *K-Nearest Neighbor*, termasuk penentuan nilai K. Jika nilai K terlalu kecil, hasil prediksi bisa menjadi

sensitif terhadap noise. Sebaliknya, jika nilai K terlalu besar, tetangga terdekat yang dipilih mungkin terlalu banyak dari kelas lain yang tidak relevan karena jaraknya terlalu jauh. Pemilihan nilai K genap atau ganjil juga penting. Dengan K genap dan jumlah klasifikasi genap, ada kemungkinan terjadi voting dengan jumlah suara yang sama. Namun, dengan K ganjil dan jumlah klasifikasi genap, kemungkinan ini dihindari karena kedua kelas tidak akan mendapat jumlah suara yang sama [3].

2.1.7 Pengujian dan Evaluasi Model

Hasil klasifikasi pada model *Naïve Bayes*, *Random Forest*, dan *K-Nearest Neighbor* kemudian dilakukan evaluasi dengan *confusion matrix*. Confusion matrix adalah alat evaluasi yang digunakan dalam model klasifikasi untuk menilai apakah prediksi objek benar atau salah. Matriks ini mencakup prediksi yang dibandingkan dengan kelas sebenarnya, sehingga berisi informasi tentang nilai aktual dan hasil prediksi dalam klasifikasi. [14].

Tabel 2.1 Confusion Matrix Dua Kelas

Classification	Predicted Class	
	Class = Yes	Class = No
Class = Yes	a (true positive)	b (false negative)
Class = No	c (false positive)	d (true negative)

Pada tabel *confusion matrix* di atas, *true positive (TP)* adalah jumlah data positif yang diklasifikasikan sebagai positif, sementara *false positive (FP)* adalah jumlah

data negatif yang salah diklasifikasikan sebagai positif. *False negative (FN)* mengacu pada jumlah data positif yang salah diklasifikasikan sebagai negatif, dan *true negative (TN)* adalah jumlah data negatif yang benar diklasifikasikan sebagai negatif. Setelah data uji diklasifikasikan, *confusion matrix* dapat digunakan untuk menghitung nilai sensitivitas, spesifisitas, dan akurasi.

2.2 Kajian Hasil Penelitian yang Relevan

Penelitian relevan dengan menggunakan algoritma *Random Forest* atau *Naive Bayes* telah menghasilkan temuan yang signifikan dalam berbagai domain. Algoritma ini telah terbukti efektif dalam memprediksi berbagai kejadian, seperti diagnosis penyakit, penipuan keuangan, dan preferensi konsumen. Berikut adalah rangkuman hasil kajian penelitian yang relevan dengan penggunaan *Random Forest* dan *Naive Bayes* dalam memprediksi berbagai hal.

Tabel 2.2 Penelitian yang Relevan

Peneliti & Tahun	**Judul Penelitian**	**Tujuan Penelitian**	**Hasil**
Sarah, N. Al, Rifat, F. Y., Hossain, M. S., & Narman, H. S., 2021	An Efficient Android Malware Prediction Using Ensemble machine learning algorithms	Melakukan pendekatan malware Android prediction menggunakan beberapa algoritma ensemble machine learning. Dengan	Hasil penelitiannnya didapatkan model LightGBM memperoleh akurasi tertinggi (sebesar 99,5%) dengan

		total feature pada dataset berjumlah 215 feature. Kemudian diterapkannya *Recursive Feature Elimination* (*REF*) dan *Recursive Feature Elimination* with *Cross Validation* (*REFCV*) pada Feature Selections.	menggunakan 100 fitur optimal. [15]
Agung Purwanto, Handoyo Widi Nugroho, 2023	Analisa Perbandingan Kinerja Algoritma C4.5 Dan Algoritma *K-Nearest Neighbor*s Untuk Klasifikasi Penerima Beasiswa	Menentukan penerima beasiswa di Universitas Muhammdiyah Pringsewu dengan membandingkan performa Algoritma C.45 dan Algoritma *K-Nearest Neighbor*s	Dari beberapa data sampel calon penerima dari jurusan algoritma *K-Nearest Neighbor*s memiliki performansi yang lebih baik yaitu presisi 98,08%, akurasi 98,30% dan nilai recall 98,00%, dengan hasil AUC sebesar 1,000 sedangkan C4,5 algoritma. mencapai 97,23% dengan nilai

			precision 94.43%, nilai recall 100,00% dan hasil AUC 0,956. [6]
Wibisono A, Fahrurozi A Jurnal Ilmiah Teknologi dan Rekayasa (2019)	Perbandingan Algoritma Klasifikasi Dalam Pengklasifikasian Data Penyakit Jantung Koroner	Membandingkan algoritma *Naïve Bayes, K-NEAREST NEIGHBOR , Decision Tree* dan *Random Forest* dalam mengklasifikasikan data penyakit jantung koroner	Berdasarkan analisis terhadap 300 dataset penyakit jantung koroner, algoritma *Random Forest* terbukti lebih unggul dan optimal dibandingkan dengan algoritma lainnya. Klasifikasi menggunakan algoritma *Random Forest* mencapai akurasi sebesar 85,668%, sedangkan *Naïve Bayes* dan *Decision Tree* masing-masing memiliki akurasi sebesar 80,33%. *K-Nearest Neighbor* memiliki akurasi terendah, yaitu 69,67%.[5].

Kumar Y, Saini S, Payal R SSRN Electronic Journal (2021) (February 2022)	Comparative Analysis for Fraud Detection Using Logistic Regression, *Random Forest* and Support Vector Machine	Membandingkan algoritma *Logistic Regression, Random Forest* dan *support vector machine* untuk mendeteksi penipuan dalam transaksi menggunakan kartu kredit	Penelitian ini menggunakan 75% data training dan 25% data testing. Perbandingan *Logistic Regression, Random Forest and support vector machine* dalam deteksi kasus penipuan tersebut menghasilkan akurasi masing-masing akurasi masing-masing 77,97%, 81,79%, dan 65,16%. Sehingga *algoritma Random Forest* memberikan hasil yang lebih baik untuk klasifikasi penipuan dengan akurasi 81,79% [16].
Virgina V, Wibowo P, Aurelia J (2020) 618-622	Comparison between Support Vector Machine and *Random*	Melakukan perbandingan Support Vector Machine dan	Data yang digunakan terdiri dari 192 sampel dengan 66 sampel

	Forest for Hepatocellular Carcinoma (HCC) Classification	*Random Forest* untuk mengkalisikasikan penyakit Hepatocellular Carcinoma (HCC).	HCC dan 126 sampel non-HCC yang diperoleh dari Rumah Sakit Al Islam Bandung. Hasil penelitian menunjukkan bahwa SVM dan RF memiliki nilai akurasi tertinggi masing-masing sebesar 90% dan 100%. Oleh karena itu, metode RF merupakan model yang lebih baik dibandingkan dengan SVM dan disarankan untuk digunakan dalam klasifikasi HCC [14].
Prajarini D (2021) 15(3) 1-5	Perbandingan Algoritma Klasifikasi *Data mining* Untuk Prediksi Penyakit Kulit	Membandingkan algoritma *Decision Tree, Naive Bayes*, *K-Nearest Neighbor* dan support vector machine	Data yang digunakan berasal dari UCI, khususnya dataset penyakit kulit, yang terdiri dari 366 instances dan 35 atribut. Metode uji

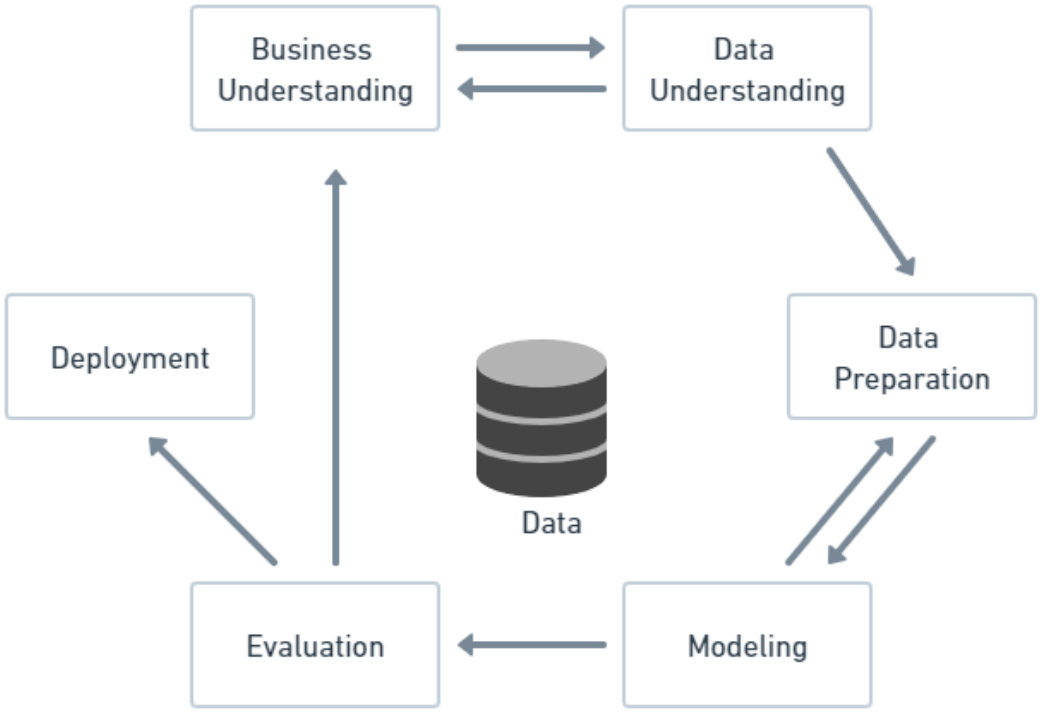
			yang diterapkan meliputi persentase split 50%, 60%, 70%, 80%, dan 90%. Berdasarkan hasil penelitian, akurasi rata-rata tertinggi diperoleh oleh Support Vector Machine dan *Naive Bayes*, masing-masing mencapai 98,1%. *K-Nearest Neighbor* memperoleh akurasi sebesar 95,3%, sementara *Decision Tree* mencapai 94,7%. [28]
Kusrini K, Luthfi E, Abdullah R 2019. 4th International Conference on Information Technology, Information	Comparison of *Naive Bayes* and *K-NEAREST NEIGHBOR* method on tuition fee payment overdue prediction	Membandingkan akurasi antara kedua metode *Naive Bayes* dan *K-Nearest Neighbor* (*K-NEAREST NEIGHBOR*) dalam	Data yang digunakan untuk praktikum adalah data pokok pendidikan kedinasan SMK Al-Islam Surakarta tahun 2017/2018 sebanyak 236 data.

Systems and Electrical Engineering, ICITISEE 2019 (2019) 6 125-130		memprediksi keterlambatan pembayaran SPP	Untuk meningkatkan akurasinya, penelitian ini juga menggabungkan metode prediksi dengan teknik pemilihan fitur *Recursive Feature Elimination* yang biasa digunakan untuk memilih parameter optimal untuk proses prediksi. Pada akhirnya sistem diuji dengan menggunakan metode Confusion Matrix. Hasil penelitian menunjukkan bahwa Metode *Naive Bayes* dengan pemilihan atribut *Recursive Feature Elimination* menghasilkan akurasi tertinggi sebesar 69% [29]

2.3 Kerangka Berpikir

2.3.1 *Metode CRISP-DM*

Metode yang digunakan dalam penelitian ini menggunakan standard *Cross-Industry Standard Process* for *Data mining* (*CRISP-DM*) yang terdiri dari beberapa tahapan yaitu *Business Understanding*, *Data Understanding*, *Data Preparation*, Modelling, *Evaluation*, dan *Deployment* [30].



Gambar 2.1 Metode *CRISP-DM*

Berikut adalah penjelasan dari tahapan Metode *CRISP-DM*:

1. Business Understanding

Tahap pertama pada metode *CRISP-DM* ini merupakan tahap yang sangat penting. Pada tahap ini, diperlukan pemahaman tentang objek bisnis, cara memperoleh atau mengumpulkan data, dan bagaimana menyelaraskan tujuan pemodelan dengan tujuan bisnis untuk membangun model yang optimal. Aktivitas yang dilakukan antara lain dengan menetapkan tujuan dan persyaratan secara menyeluruh secara jelas, menerjemahkan tujuan tersebut, menentukan batasan dalam formulasi masalah *data mining*, dan mempersiapkan strategi awal untuk mencapai tujuan tersebut.

2. Data Understanding

Secara umum, tujuan dari data understanding adalah untuk mengidentifikasi masalah yang mungkin ada. Tahap ini membangun fondasi analitis untuk penelitian dengan membuat ringkasan dan mengidentifikasi potensi masalah dalam data. Proses ini harus dilakukan dengan cermat dan tidak tergesa-gesa, terutama dalam visualisasi data, di mana sering kali sulit untuk mendapatkan wawasan ketika dikaitkan dengan ringkasan data. Jika terdapat masalah yang tidak terpecahkan pada tahap ini, maka akan mengganggu proses di tahap Modeling.

Data understanding dilakukan untuk memastikan apakah distribusi data sesuai harapan atau untuk mengungkapkan penyimpangan yang tidak terduga yang perlu ditangani pada tahap berikutnya, yaitu melalui Data Preparation. Masalah dalam

data, seperti nilai yang hilang atau *missing value* dan outlier, harus diidentifikasi dan diukur agar dapat diperbaiki dalam tahap Data Preparation.

3. Data Preparation

Tahap data preparation dilakukan untuk memperbaiki masalah dalam data dan membuat variabel turunan. Proses ini memerlukan pemikiran yang mendalam dan upaya yang signifikan untuk memastikan bahwa data sesuai dengan algoritma yang akan digunakan.

Namun, meskipun pada tahap awal Data Preparation masalah data mungkin telah diatasi, bukan berarti data tersebut bisa langsung digunakan hingga tahap akhir. Tahap ini dapat ditinjau kembali ketika ada masalah yang ditemukan selama pelatihan model. Oleh karena itu, proses ini dilakukan secara iteratif hingga menemukan solusi yang sesuai dengan tujuan.

Pada tahap ini, proses sampling akan dilakukan, dan data akan dibagi menjadi dua bagian yaitu data training dan data testing. Aktivitas yang dilakukan antara lain dengan memilih kasus dan parameter yang akan dianalisis (Select Data), melakukan transformasi terhadap parameter tertentu (Transformation), dan membersihkan data agar siap untuk tahap Modeling (Cleaning).

4. Modeling

Tujuan dari tahap ini adalah membuat model prediktif atau deskriptif. Pada tahap ini, metode statistik dan Machine Learning digunakan untuk menentukan teknik *data mining*, serta algoritma *data mining* yang akan diterapkan. Selanjutnya, teknik dan algoritma *data mining* tersebut diterapkan pada data dengan tools atau alat bantu yang sesuai. Proses dapat Kembali ke tahap data preparation jika diperlukan penyesuaian data untuk teknik *data mining* tertentu.

5. Evaluation

Tahap ini berfokus pada analisis dan pemahaman hasil yang diperoleh dari proses *data mining* yang telah dilakukan pada tahap pemodelan sebelumnya. Pada tahap ini, dilakukan evaluasi terhadap model yang diterapkan untuk memastikan bahwa model tersebut efektif dan sesuai dengan tujuan yang telah ditetapkan. Evaluasi mencakup pengecekan akurasi, validitas, dan relevansi model dalam konteks tujuan bisnis atau penelitian. Jika ditemukan bahwa model tidak sepenuhnya sesuai dengan tujuan awal, maka perlu dilakukan revisi atau penyesuaian lebih lanjut untuk meningkatkan kualitas dan kesesuaian model tersebut. Dengan demikian, tahap ini sangat penting untuk menjamin bahwa model yang dihasilkan dapat memberikan insight yang berharga dan dapat diandalkan dalam pengambilan keputusan.

6. Deployment

Tahap *Deployment*, atau tahap penerapan model, merupakan bagian yang paling penting dalam proses *CRISP-DM*. Perencanaan untuk *Deployment* sebenarnya dimulai sejak tahap *Business Understanding*. Proses ini juga perlu mempertimbangkan cara menerjemahkan skor model menjadi keputusan yang dapat diambil. Hal ini memastikan bahwa model yang dikembangkan dapat memberikan nilai nyata dan digunakan secara efektif dalam praktik bisnis. Selain itu, *Deployment* juga melibatkan pemantauan berkelanjutan untuk menilai kinerja model dan melakukan penyesuaian jika diperlukan untuk menjaga relevansi dan efektivitasnya. Dengan pendekatan ini, model yang dihasilkan tidak hanya bersifat teoretis tetapi juga dapat memberikan kontribusi langsung terhadap tujuan bisnis.