

BAB 2

TINJAUAN PUSTAKA

2.1 Prediksi indeks prestasi mahasiswa

Prediksi adalah meramal atau memperkirakan suatu hal yang belum terjadi dengan sistematis mengenai masa depan akan tetapi berdasarkan rekam jejak yang sudah pernah terjadi dimasa lalu yang dapat membantu saat mengambil sebuah keputusan, akan tetapi prediksi tidak memiliki jaminan secara pasti atas jawaban yang dihasilkan, melainkan memberikan hasil atau output yang semirip atau sedekat mungkin apa yang akan terjadi (Wantono, 2013). Sehingga prediksi akan bermanfaat jika diterapkan dibeberapa bidang, salah satu bidang yang dapat menerapkan prediksi adalah bidang pendidikan untuk memprediksi indeks prestasi mahasiswa. Dalam memprediksi indeks prestasi mahasiswa dibutuhkan atribut-atribut tertentu untuk menghasilkan sebuah informasi sehingga dapat menangani secara dini mahasiswa yang memiliki potensi penurunan indeks prestasi (Rolansa et al., 2020). Informasi mengenai prestasi akademik mahasiswa dapat berguna bagi manajemen universitas untuk memantau proses kegiatan perkuliahan mahasiswa dan melakukan pengambilan kebijakan yang terkait untuk meningkatkan prestasi mahasiswa (Hasudungan & Pranoto, 2021). Prediksi indeks prestasi mahasiswa semester pertama dapat dilakukan dengan berbagai metode data mining seperti yang ditunjukkan pada tabel 2.1.

Tabel 2.1 Penelitian terkait indeks prestasi mahasiswa

Peneliti	Permasalahann	Atribut	Metode	Hasil
Sahat Sonang, dkk (2022)	Melakukan prediksi prestasi mahasiswa dengan menggunakan algoritma Backpropagation	9 nilai mata kuliah : pengantar teknik informasi, elektronika dasar, pancasila, matematika diskrit, agama,	Algoritma Backpropagation	Perolehan luaran terbaik dengan epoch 8149, durasi 17detik, error rate 0.1248, dan akurasi

Peneliti	Permasalahann	Atribut	Metode	Hasil
		bahasa pemrograman i, praktek bahasa pemrograman, logika algoritma, bahasa inggris		87,5% untuk data testing
Dea Alverina, dkk (2018)	Membuat perbandingan akurasi algoritma C4.5 dan CART dalam memprediksi indeks prestasi mahasiswa	Sekolah asal, status, lokasi, level berbahasa inggris, nilai numerik, nilai verbal, nilai spasial dan nilai analogi), jalur non-prestasi.	Algoritma C4.5 dan CART	Akurasi memprediksi kategori IP mahasiswa menghasilkan akurasi sebesar 86,86%
Freska Rolansa, dkk (2020)	Membuat sistem prediksi dan evaluasi prestasi akademik mahasiswa di program studi teknik informatika menggunakan data mining	Indeks prestasi semester 1 sampai semester 6	Algoritma C4.5	Data yang digunakan sebanyak 116 mahasiswa dengan akurasi sebesar 94,55%
Rofilde Hasudungan (2018)	Melakukan analisis terhadap indikator kinerja dosen terhadap prestasi mahasiswa semester satu dengan menggunakan decision tree	nilai kuis, kompetensi pedagogic, kompetensi professional, kompetensi kepribadian, kompetensi sosial	Decision Tree	Hasil akurasi yang didapat dalam penelitian ini sebesar 74%
Tommy dan Amir Mahmud (2021)	Melakukan analisis model prediksi prestasi mahasiswa	id nilai, id matkul, nilai, id dosen 1, id dosen 2, id	Logistic regression, Support Vector	Hasil akurasi pada tiap model

Peneliti	Permasalahann	Atribut	Metode	Hasil
	berdasarkan evaluasi pembelajaran menggunakan pendekatan data science	prodi, pedagogic, professional, kepribadian	Machine (SVM), Decision Tree, Naïve Bayes, dan K-nears	logistic regression 55%, SVM 55%, Decision Tree 68%, Naïve Bayes 55%, dan KNN 66%
Yusran Timur dan Chrystle Beatrix (2020)	Melakukan prediksi terhadap prestasi mahasiswa yang berkuliah sambil bekerja	Jenis kelamin, SKS, jurusan, tempat kerja, umur, jam kerja, tempat tinggal	Decision Tree, C4.5, SMOTE	Dengan hasil antara ketiganya berada di 70% sampai 80%
Hikmah Maros dan Sarah Juniar (2016)	Mengimplementasi metode k-nears prediksi indeks prestasi	indeks prestasi, faktor fisik, faktor psikologis	Algoritma K-nears neightboor	Dari 22 data tingkat akurasi sebesar 88%

2.2 Algoritma Naïve Bayes

Data mining adalah kegiatan mencari pola yang menarik dalam data yang jumlah besar untuk mendapatkan informasi yang belum diketahui (Saputro & Sari, 2020). Dalam suatu istilah data mining digunakan untuk mengurai pengetahuan dalam data yang sangat banyak sehingga data mining bermanfaat untuk digunakan mencari hubungan yang tak terduga dan kemudian meringkasnya dengan cara berbeda dan mudah dipahami (Widaningsih, 2019). Data mining dapat dikatakan juga proses penyelesaian masalah yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan machine learning untuk mengekstrak informasi dari berbagai data. Metode naïve bayes masuk dalam metode klasifikasi data mining untuk mengatasi ketidakpastian, metode ini memiliki fungsi memprediksi peluang di masa depan yang berdasarkan pengalaman dan rekam jejak yang didapat dalam masa sebelumnya (Hasudungan & Pranoto, 2021). Asumsi algoritma naïve bayes bahwa setiap nilai antar atribut berbentuk indepen atau saling bebas, hal ini dikarenakan ketika kekosongan atau

ketidakadaan pada suatu atribut tidak akan mengganggu atribut lainnya saat klasifikasi dijalankan. (Saputro & Sari, 2020) Persamaan (2.1) merupakan persamaan metode naïve bayes.

$$P(H | E) = \frac{P(E | H)P(H)}{P(E)} \quad (2.1)$$

Keterangan:

$P(H | E)$: Probabilitas sebuah hipotesis H berdasarkan bukti E yang terjadi.

$P(E | H)$: Probabilitas sebuah bukti E berdasarkan hipotesis H yang terjadi.

$P(H)$: Probabilitas hipotesis H (prior probabilitas).

$P(E)$: Probabilitas bukti E.

E : Elemen data dengan class yang belum diketahui

H : Hipotesis data merupakan suatu kelas yang spesifik

Naïve bayes adalah algoritma yang memiliki salah satu kelebihan tingkat akurasi yang tinggi, tingkat waktu komputasi yang cepat, dan dapat menggunakan data latih atau training yang sedikit (Desember et al., 2017). Jumlah penggunaan data training yang digunakan akan menentukan estimasi atribut yang diperlukan dalam pemrosesan klasifikasi. Adapun berikut penelitian yang menggunakan algoritma naïve bayes digunakan di beberapa penelitian terkait mahasiswa dapat dilihat pada tabel 2.2.

Tabel 2.2 Tabel penelitian terkait metode naïve bayes

Peneliti	Permasalahann	Hasil
Lila Setiyani, dkk (2020)	Memprediksikan kelulusan mahasiswa tepat Waktu menggunakan metode data mining naïve bayes dengan cara systematic review	Dengan 91 literatur, akan tetapi hanya 68 literatur menggunakan naïve bayes dengan akurasi 90%
Andika Mahanggara dan Arif Dwi (2019)	Prediksi Pengunduran Diri Mahasiswa Universitas Amikom Yogyakarta Menggunakan Metode Naive Bayes	Dari hasil penelitian tingkat akurasi sebesar 77,78%
Indra Kurniawan, dkk (2018)	Implementasi Data Mining untuk Prediksi Mahasiswa Pengambil Mata	Dari hasil penelitian tingkat akurasi dari 7 matakuliah sebesar 74,91%

Peneliti	Permasalahann	Hasil
	Kuliah dengan Algoritme Naive Bayes	
Irkham Widhi dan Betty Wulan (2020)	Uji Performa Algoritma Naïve Bayes untuk Prediksi Masa Studi Mahasiswa	Hasil dari penelitian ini dengan menggunakan 300 data, dan nilai akurasi 68% sampai dengan 61% paling rendah.
Endang Etriyanti, dkk (2020)	Implementasi Data Mining Menggunakan Algoritme Naive Bayes Classifier dan C4.5 untuk Memprediksi Kelulusan Mahasiswa	Dari keseluruhan 227 data mahasiswa setelah pemrosesan data menjadi 162 data dengan 9 atribut menghasilkan akurasi sebesar 78,46 % untuk algoritma Naïve bayes
Juli Desember, dkk (2017)	Klasifikasi Judul Buku dengan Algoritma Naive Bayes dan Pencarian Buku pada Perpustakaan Jurusan Teknik Elektro	Penelitian dengan menggunakan 249 dokumen memperoleh 70 dokumen relevan dan 5 tidak relevan dan algoritma naïve bayes memiliki akurasi sebesar 97,78%
Sri Widaningsih (2019)	Perbandingan metode data mining untuk prediksi nilai dan waktu kelulusan mahasiswa prodi teknik informatika dengan Algoritma C4.5, Naïve bayes, KNN, dan SVM	Hasil dari penelitian ini dengan menggunakan 500 data, dan nilai akurasi 76% untuk algoritma naïve bayes

Penggunaan metode naïve bayes pada prediksi indeks prestasi mahasiswa juga banyak dilakukan oleh beberapa peneliti seperti pada tabel 2.3.

Tabel 2.3 Tabel indeks prestasi mahasiswa menggunakan metode naive bayes

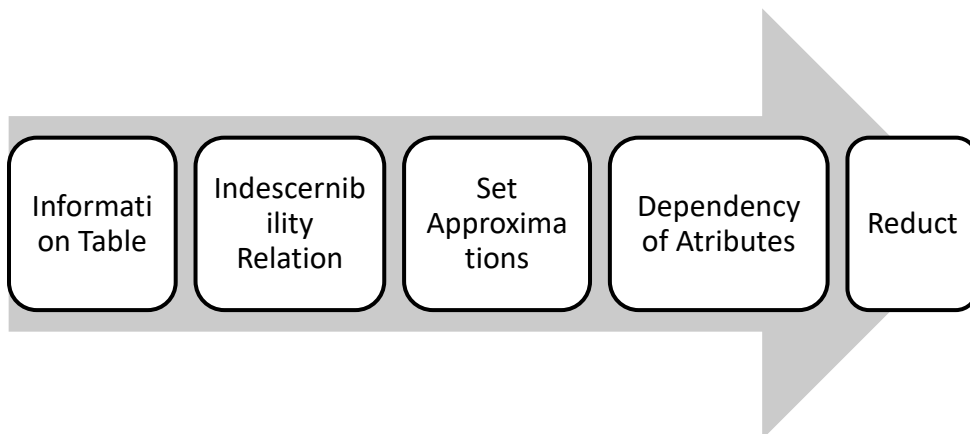
Peneliti	Atribut	Permasalahann	Hasil
Rofilde Hasudungan dan Wawan Pranoto (2021)	Umur, jenis kelamin, tempat tinggal, jarak, jenis pendidikan, status pendidikan sebelumnya, pekerjaan ayah, pekerjaan ibu, status orang tua, penghasilan ayah, penghasilan ibu, jumlah anggota keluarga, jumlah kakak, jumlah adik	Mengimplementasi teorema naïve bayes pada prediksi prestasi mahasiswa	Hasil penelitian dari banyak 40 data mahasiswa, 5 dianggap anomali, menunjukkan tingkat akurasi 77,5% dari algoritma naïve bayes
Anita Desiani, dkk (2020)	jenis kelamin, jam belajar diluar kampus, pendidikan ayah, pekerjaan ayah, pendidikan ibu, pekerjaan ibu, asal daerah, tempat tinggal selama kuliah, nilai TOEFL, dan IPS	Melakukan prediksi tingkat indeks prestasi kumulatif akademik mahasiswa dengan menggunakan teknik data mining	Dari 141 data yang terkumpul dengan 13 atribut menghasilkan akurasi sebesar 74,47%
Yoga Mochammad Firdaus (2019)	Jenis kelamin, kelas, materi pokok, pemahaman konsep dan praktikum, lima sukses santri, kegiatan ekstrakurikuler, absensi dan keaktifan, IPK	Menerapkan metode naïve bayes classifier untuk mengklasifikasi tingkat prestasi akademik santri pondok pesantren mahasiswa baitul jannah malang	Akurasi yang didapatkan pada penelitian ini sebesar 76%.

Peneliti	Atribut	Permasalahann	Hasil
Muhammad Syukri, dkk (2017)	Indeks prestasi semester 1 sampai 4	Mengimplementasikan naïve bayes pada indeks prestasi sebagai evaluasi kinerja akademik	Hasil akurasi yang didapat pada penelitian ini sebesar 92,3%

Namun pada penelitian-penelitian pada tabel 2.2 dan tabel 2.3 tidak menggunakan atau menerapkan pemilihan atribut. Sehingga perlu adanya sebuah metode untuk melakukan pemilihan atribut untuk meningkatkan hasil akurasi.

2.3 Algoritma Rough Set

Rough set dikemukakan oleh Zdzilaw Pawlak pada tahun 1980. Filosofi rough set didasarkan pada asumsi bahwa setiap objek semesta memiliki beberapa informasi data atau pengetahuan yang selalu terikat karena data itu sendiri sudah memiliki informasi yang dibutuhkan (Pawlak, 1998). Rough set salah satu metode matematika yang sering menangani data yang tidak konsisten, tidak memiliki kepastian dan samar sehingga rough set memiliki kelebihan masuk dalam algoritma efisien untuk mencari pola tersembunyi dalam data, yang kemudian juga menemukan reduksi dari himpunan datanya, lalu mengevaluasi data, serta menghasilkan keputusan aturan-aturan untuk pemilihan atribut (Samaray, 2022). Adapun skema dalam penyelesaian algoritma rough set dalam pemilihan attribute dapat dilihat pada gambar 2.1.



Gambar 0.1 Skema penyelesaian algoritma rough set

Berikut keterangan berdasarkan gambar 2.1 mengenai skema penyelesaian algoritma rough set sebagai penyelesain : (Senan et al., 2011)

- 1) *Information Table* adalah tabel yang terdiri dari kolom dan baris berisi data, dimana kolom akan diberikan label atribut, dan baris akan diisi nilai dari atribut. Dengan informasi sistem seperti berikut $S = (U, A, V, f)$, dimana U adalah himpunan dari objek, A adalah himpunan atribut yang tidak boleh kosong, $V = \bigcup_{a \in A} V_a$, V_a adalah domain atribut a , $f: U \times A \rightarrow V$ adalah fungsi total yang sedemikian rupa sehingga $f(u, a) \in V_a$, untuk setiap $f(u, a) \in U \times A$, disebut sebagai fungsi informasi atau pengetahuan. Tabel harus memiliki satu atribut keputusan (*Decision information system*) yang tidak boleh memiliki nilai kosong. Dengan informasi sistem sebagai berikut $D = (U, A \cup \{d\}, V, f)$, dimana U, A, V dan f sesuai dengan D dan $\{d\}$ adalah atribut keputusan dimana $\{d\} \cap A \neq \emptyset$.
- 2) *Indiscernibility Relation* adalah sebuah gagasan antar objek yang dapat didefinisikan memiliki kemiripan sehingga dapat disatukan. Dengan definisi $S = (U, A, V, f)$ menjadi sistem informasi dan B akan menjadi bagian dari A dua element $x, y \in U$ dikatakan *B-indiscernible* (tidak dapat dibedakan oleh himpunan atribut $B \subseteq A$ dalam S) jika hanya $f(x, a) = f(y, a)$ untuk setiap $a \in B$.
- 3) *Set Approximations* adalah mengelompokkan hasil dari *Indiscernibility relation* akan digunakan untuk mendefinisikan approximations sebagai konsep dasar dalam algoritma rough set. Untuk menentukan perkiraan terbawah dan perkiraan teratas dalam suatu himpunan sehingga dapat didefinisikan sebagai berikut $S = (U, A, V, f)$ menjadi sistem informasi dan B akan menjadi bagian dari A , X akan menjadi bagian dari U . *B-lower approximation* (perkiraan terbawah) dari X dapat dinotasikan sebagai $\underline{B}(X)$, dan *B-upper approximation* dari X dapat dinotasikan sebagai $\overline{B}(X)$. Sehingga dapat didefinisikan dengan persamaan 2.2.

$$\underline{B}(X) = \{x \in U \mid [x]_B \subseteq X\} \text{ dan } \overline{B}(X) = \{x \in U \mid [x]_B \cap X \neq \emptyset\} \quad (2.2)$$

Masalah penting lainnya adalah mencari atau menemukan dependensi antar atribut, dengan definisi $S = (U, A, V, f)$ menjadi sistem informasi, D dan C menjadi bagian dari A. Atribut D secara fungsional akan bergantung pada atribut C, sehingga dapat dinotasikan $C \Rightarrow D$, jika setiap nilai D (*decision*) terkait persis dengan nilai C.

- 4) *Dependency of Attributes* adalah langkah menghitung konsistensi setiap atribut dengan definisi sebagai berikut $S = (U, A, V, f)$ menjadi sistem informasi, D dan C menjadi bagian dari A. dependensi D pada C dalam tingkat k ($0 \leq k \leq 1$), dengan notasi $C \Rightarrow_k D$. Maka dapat didefinisikan dengan persamaan 2.3.

$$k = \frac{\sum_{x \in U/D} |\underline{C}(X)|}{|U|} \quad (2.3)$$

- 5) *Reduct* adalah proses meminimalisir himpunan dari atribut. Dengan cara menghitung ulang menggunakan langkah-langkah sebelumnya untuk diterapkan di setiap atribut yang ada, sehingga mendapatkan himpunan atribut terbaik dan tidak mengurangi nilai konsistensi atributnya. Dengan definisi sebagai berikut $S = (U, A, V, f)$ menjadi sistem informasi, dan B menjadi bagian dari A, jika B berpengaruh pada konsistensi atribut menjadi berlebihan dapat dibuang dengan notasi $B \text{ if } U / (B - \{b\}) = U / B$, jika tidak mempengaruhi konsistensi atribut maka sangat diperlukan

Tabel 2.4 Tabel penelitian terkait Rough set

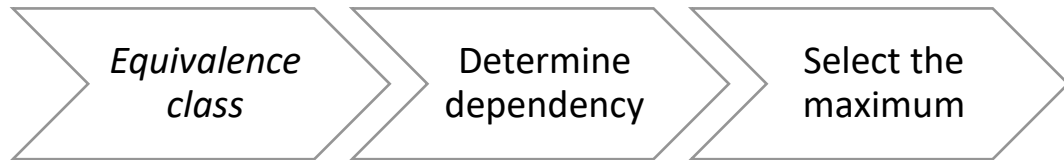
Peneliti	Permasalahann	Hasil
Rofilde Hasudungan (2021)	Menerapkan algoritma naïve bayes model untuk analisis data mahasiswa	Menggunakan 47 data mahasiswa dengan 28 atribut dan menjadi 6 atribut ketika menggunakan rough set menghasilkan akurasi sebesar 68,09% untuk naïve bayes tanpa menggunakan rough set dan meningkat menjadi 80,85% dengan

Peneliti	Permasalahann	Hasil
		menggunakan rough set.
Rofilde Hasudungan, dkk (2020)	Implementasi MDA untuk meningkatkan klasifikasi naïve bayes pada prediksi kinerja mahasiswa	Menggunakan 40 data mahasiswa dengan 28 atribut ketika menggunakan rough set menghasilkan akurasi sebesar 68% untuk naïve bayes tanpa menggunakan rough set dan meningkat menjadi 79% dengan menggunakan rough set.
Angga Putra Juledi (2022)	Analisis Algoritma Pada Penerimaan beasiswa pada program BAZNAS di Labuhanbatu. Rough Set digunakan untuk membantu penentuan penerima beasiswa menggunakan bantuan aplikasi Rosetta	Dari keseluruhan 20 data yang diambil menghasilkan 4 rules hasil reduct
Silvana Samaray (2022)	Implementasi algoritma rough set untuk prediksi hasil belajar sangat penting dikarekan salah satu yang dapat mempengaruhi indeks prestasi menggunakan bantuan aplikasi Rosetta.	Dari keseluruhan 157 data menghasilkan 14 rules hasil reduct
Suci Utari (2022)	Penerapan algoritma rough set untuk memprediksi jumlah permintaan produk pada PT. Duta Abadi Primantara menggunakan bantuan aplikasi Rosetta	Dari keseluruhan 12 data menghasilkan 12 rules hasil reduct
Aswanda Putra (2018)	Penerapan algoritma rough set dalam memprediksi kecerdasan anak dengan menggunakan bantuan aplikasi Rosetta	Menghasilkan 13 rules setelah reduct

2.4 Maximum Dependency of Attributes

Maximum dependency attributes adalah rough set berbasis pemilihan atribut yang dapat menemukan ketergantungan antar atribut dan dapat mengurangi atribut yang berlebihan. Dalam mengurangi atribut yang berlebihan dapat

menggunakan cara dengan menghitung ketergantungan antar atribut satu dengan atribut lainnya yang berdasarkan nilai ketergantungan maksimum atribut terhadap data (Hasudungan et al., 2020). Adapun dalam langkah penerapan maximum dependency attributes memerlukan beberapa tahap penyelesaian seperti terdapat pada gambar 2.2 berikut.



Gambar 0.2 Skema penyelesaian MDA

Berikut keterangan berdasarkan skema penyelesaian maximum dependency attributes sebagai metode menghitung ketergantungan atribut : (Herawan et al., 2010).

- 1) *Equivalence class* adalah tahap pertama dalam menerapkan algoritma rough set MDA untuk mencari equivalence class pada setiap atribut dari himpunan U dengan menggunakan *indiscernibility relation* pada setiap atribut dengan definisi $S = (U, A, V, f)$ menjadi sistem informasi, D dan C menjadi bagian dari A . Jika D bergantung penuh pada C , kemudian $\alpha_B(X) \leq \alpha_C(X)$, untuk semua anggota $X \subseteq U$. berdasarkan definisi itu didapat $IND(C) \subseteq IND(D)$ oleh karenanya dapat diterapkan pada persamaan 2.4.

$$D(X) \subseteq C(X) \subseteq X \subseteq \underline{C}(X) \subseteq DX \quad (2.4)$$

- 2) *Determine dependency* adalah tahap selanjutnya dalam menentukan ketergantungan maksimum atribut α^j sehubungan dengan semua atribut α_i , akan tetapi $\alpha^j \neq \alpha_i$. Adapun dalam penerapannya dapat menggunakan persamaan 2.5.

$$D(\underline{R}(X), \overline{R}(X)) = 1 - \frac{|\underline{R}(X) \cap \overline{R}(X)|}{|\underline{R}(X) \cup \overline{R}(X)|}, = 1 - \frac{|\underline{R}(X)|}{|\overline{R}(X)|}, = 1 - aR(X) \quad (2.5)$$

- 3) *Select the maximum* adalah tahap menseleksi ketergantungan maksimum dari setiap atribut Tingkat ketergantungan atribut maksimum dapat ditentukan

berdasarkan semakin banyak atribut yang bernilai sama akan memperoleh nilai ketergantungan. Dengan definisi $S = (U, A, V, f)$ menjadi sistem informasi, $S = (U, A, V, f)$ menjadi sistem informasi dan C_1, C_2, \dots, C_n sehingga D menjadi bagian dari A . Jika $C_1 \Rightarrow k_1 D, C_2 \Rightarrow k_2 D, \dots, C_n \Rightarrow k_n D$, dimana $k_n \leq k_{n-1} \leq \dots \leq k_2 \leq k_1$, sehingga $\alpha D(X) \leq \alpha C_n(X) \leq \alpha C_{n-1}(X) \leq \dots \leq \alpha C_2(X) \leq \alpha C_1(X)$ For every $X \subseteq U$. Adapun terdapat pada persamaan 2.6.

$$\alpha D(X) \leq \alpha C_n(X) \mid k_n \leq k_{n-1} \leq \dots \leq k_2 \leq k_1 \mid [x]C_n \subseteq [x]C_{n-1} \quad (2.6)$$

2.5 Pemrosesan data

Pemrosesan data adalah tahapan dalam data mining yang digunakan untuk memproses data agar dapat dijalankan pada proses pengklasifikasian. Pemrosesan data mempunyai tujuan untuk mengurangi data, mencari relasi antar data, menormalkan data, membuang data menyimpang(outliers) dan mengekstrak pengetahuan untuk melakukan hal itu memerlukan beberapa teknik yaitu Data cleaning, Data integration, Data transformation, dan Data reduction. Berikut keterangan berdasarkan teknik pemrosesan data : (Suad A. Alasadi & Wesam S. Bhaya, 2017).

- 1) *Data clening* atau pembersihan data adalah membersihkan tiap baris data yang tidak lengkap atau kurang dalam kolomnya, tipe data pada baris tidak sesuai dengan kolom, dan data yang tidak konsisten.
- 2) *Data integration* atau Integrasi data adalah teknik untuk mengkombinasikan data dari berbagai sumber data menjadi satu data yang konsisten seperti contoh data pergudangan, yang memiliki sumber dari berbagai tempat.
- 3) *Data transformation* atau Transformasi data adalah proses mentransformasikan data dengan mengubah tipe data numerik menjadi kategorikal agar data dapat diproses saat klasifikasi.
- 4) *Data Reduction* atau Reduksi data adalah teknik ini dapat digunakan untuk mereduksi data yang berjumlah besar dengan mepresentasikan data dalam jumlah yang kecil. dengan tetap mempertahankan integritas dari data yang asli.

2.6 Evaluasi

Evaluasi adalah pengukuran kinerja sebuah algoritma terhadap data. Proses evaluasi melakukan penilaian pada pola-pola menarik atau model prediksi apakah telah memenuhi hipotesa awal atau belum. Evaluasi data mining tipe klasifikasi dilakukan pengujian untuk proses prediksi kebenaran objek. Confusion matrix adalah salah satu cara yang sering digunakan pada proses evaluasi model data mining klasifikasi dengan memprediksi kebenaran objek. Proses pengujian memanfaatkan confusion matrix yang menempatkan kelas prediksi dibagian atas matrik kemudian untuk sumber yang diamati diletakkan di sebelah kiri matrik. Setiap sel matrik berisi angka yang menampilkan jumlah kasus actual dari kelas yang sedang diamati (Muslim et al., 2019). Pada tabel 2.5 dijelaskan contoh confusion matrix proses klasifikasi.

Tabel 2.5 Confusion matrix

	Action True	Action False
Predict True	TP	FP
Predict False	FN	TN

Keterangan:

- 1) TP (*True Positive*) adalah observasi kelas yang benar dan prediksi yang benar
- 2) TN (*True Negatif*) adalah observasi kelas yang benar dengan prediksi yang salah
- 3) FP (*False Positive*) adalah observasi kelas yang salah dengan prediksi yang benar
- 4) FN (*False Negatif*) adalah observasi kelas yang salah dengan prediksi yang salah

Untuk mengukur akurasi (accuracy) pada model dapat menerapkan persamaan 2.7 yang digunakan untuk menghitung hasil akurasi.

$$Akurasi = \frac{TP + TN}{TP + FP + FN + TN} \quad (2.7)$$

Sedangkan untuk menghitung tingkat kesalahan (error rate) dapat didefinisikan dengan persamaan 2.8 seperti berikut.

$$Kesalahan = \frac{FP + FN}{TP + FP + FN + TN} \quad (2.8)$$

Dan untuk menghitung ketepatan (precision) mengukur data yang telah diprediksi positif dengan kenyataan yang benar dan tidak benar dapat menggunakan persamaan 2.9 seperti berikut.

$$Ketepatan = \frac{TP}{TP + FP} \quad (2.9)$$

Terakhir untuk menghitung sensitivitas (recall) banyak data yang sukses saat diprediksi dengan perbandingan seluruh data yang pada kenyataannya positif dapat menggunakan persamaan 2.10 seperti berikut.

$$Sensitivitas = \frac{TP}{TP + FN} \quad (2.10)$$