BAB 2

TINJAUAN PUSTAKA

2.1 Prediksi Tingkat Pemahaman Mahasiswa

Tingkat pemahaman mahasiswa adalah derajat atau tinggi rendahnya seseorang dalam menanggapi hal yang sangat penting dalam mempelajari sesuatu. Tingkat pemahaman yang dimiliki oleh seorang mahasiswa sangat berpengaruh dalam menerima suatu materi kuliah yang sedang diikutinya. Tingkat pemahaman mahasiswa sangat dipengaruhi oleh banyak faktor seperti kesiapan pembelajaran, ketertiban pembelajaran dan seterusnya (Siltonga & Dewi, 2019). Oleh karena itu pentingnya suatu analisa dalam memprediksi tingkat pemahaman mahasiswa membuat banyak peneliti melakukan penelitian terhadap hal ini dengan menggunakan berbagai macam algoritma seperti algoritma Rough set, algoritma C4.5, algoritma Naive bayes dan seterusnya. Berikut adalah tabel penelitian terkait yang membahas tentang prediksi tingkat pemahaman mahasiswa tertera pada tabel 2.1

Tabel 0.1 Penelitian Terdahulu

Author	Metode	Hasilnya	Keterangan
Nurul Rofiqo, Dkk(2019)	Algoritma C4.5	Nilai akurasi 87,10%	Menerapkan algoritma C4.5 untuk memprediksi tingkat pemahaman mahasiswa terhadap mata kuliah
(Algoritma et al., 2022)	Algoritma C4.5	Nilai akurasi 84,38 %	Menggunakan algoritma C4.5 untuk menetukan klasifikasi tingkat pemahaman mahasiswa terhadap mata kuliah bahasa pemprograman
(Raharjo & Windarto, 2021)	Rough Set	Nilai akurasi 53%	Prediksi tingkat pemahaman mahasiswa terhadap mata kuliah
(Mutamainnah & INFOKAM, 2019)	Naïve bayes	Nilai akurasi	Menggunakan Naïve Bayes untuk memprediksi masa studi mahasiswa

		85,17%	berdasarkan faktor yang berhubungan dengan akademik mahasiswa.
(Astuti et al., 2022)	Naïve Bayes	Nilai akurasi 69,23%	Naïve bayes untuk memprediksi tingkat pemahaman mahasiswa terhadap mata kuliah algoritma struktur data
(Siltonga & Dewi, 2019)	Naïve Bayes	Nilai akurasi 88,24%	Analisis metode naïve bayes memprediksi tingkat pemahaman mahasiswa terhadap mata kuliah berdasarkan posisi duduk
Eka Sabna, Muhardi (2016)	Decision Tree	Nilai akurasi 65%	Menggunakan algoritma Decision Tree untuk memprediksi prestasi akademik berdasarkan sosial ekonomi, motivasi, peran dosen, disiplin dan hasil belajar
Abdul Rohman, Sri Mujiyono (2021)	Decision Tree	Nilai akurasi 71,67%	Menggunakan Decision Tree C4.5 agar mendapat model pohon keputusan dengan variabel atau atribut indeks prestasi yang berpengaruh pada predikat kelulusan mahasiswa.
Riski Annisa dan Agung Sasongko (2020)	Naïve Bayes	Nilai akurasi 96,24%	Menggunakan Naïve Bayes untuk prediksi nilai akademis mahasiswa dengan memanfaatkan perhitungan probabilitas dan statistik data sebelumnya untuk memprediksi data di masa depan berdasarkan pada data sebelumnya.
Ahmad Fauzi dan	Naïve	Nilai	Menggunakan Decision Tree dan

Tukiyat (2019)	Bayes	akurasi 94,47%	Naïve Bayes hasil akurasi dari metode Naive Bayes tetap yang terbesar, meskipun peningkatan nilai akurasi setelah dioptimasi lebih rendah dari pada metode Decision Tree.
(Aspiah & Taghfirul Azhima Yoga Siswa, 2022)	C4.5	Nilai akurasi 97,22%	Implementasi correlation based feature selection (CFS) untuk peningkatan akurasi algoritma C4.5 dalam prediksi perfoma akademik mahasiswa berbasis learning management system

2.2 Teori Naïve Bayes

Naïve Bayes adalah salah satu algoritma klasifikasi berdasarkan teorema Bayesian pada statistika dan dapat digunakan untuk memprediksi probabilitas keanggotaaan suatu kelas. Naïve bayes menghitung nilai *posterior probability* P(H|X) menggunakan probabilitas P(H), P(X), dan P(X|H) dimana nilai X adalah data testing yang kelasnya belum di ketahui. Nilai H adalah hipotesis data X yang merupakan suatu kelas yang lebih spesifik. Nilai P(X|H) atau disebut juga *likelihood*, adalah probabilitas hipotesis X berdasarkan kondisi H. Nilai P(H) atau disebut juga dengan *prior probability* adalah probabilitas hipotesis H. Sedangkan nilai P(X) yang disebut juga dengan *predictor* prior *probability,* adalah probabilitas X (Suntoro, 2019).

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

Keterangan:

X : Data dengan *class* yang belum diketahui

H : Hipotesis data merupakan suatu *class* spesifik

P(H|X) : Probabilitas hipotesis H berdasar kondisi X (posteriori probabilitas)

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

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

P(X) : Probabilitas dari X

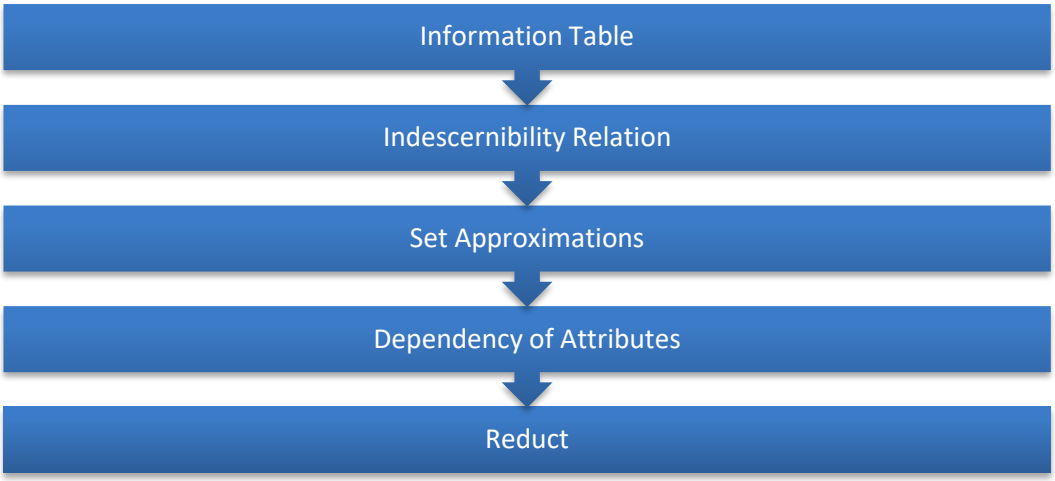
Algoritma naïve bayes memiliki kelebihan yang dianggap cepat dan kuat terutama ketika berhadapan dengan data besar. Serta naïve bayes menganggap semua atribut sama, dan karna itulah naïve bayes disebut naïf. Berikut adalah tabel penelitian terdahulu yang menggunakan roughset sebagai fitur seleksi atribut seperti tertera pada tabel 2.2

Tabel 0.2 Penelitian Terdahulu Naive Bayes

Author	Metode	Akurasi	Keterangan
(Siltonga & Dewi, 2019)	Naïve Bayes	Nilai Akurasi 88,24%	Analisis metode naïve bayes dalam memprediksi tingkat pemahaman mahasiswa terhadap mata kuliah berdasarkan posisi duduk
(Astuti et al., 2022)	Naïve Bayes	Nilai Akurasi 69,23%	Menggunakan naïve bayes untuk memprediksi tingkat pemahaman mahasiswa terhadap mata kuliah algoritma struktur data
(Mutamainnah & INFOKAM, 2019)	Naïve Bayes	Nilai Akurasi 85,17%	Menggunakan naïve bayes untuk memprediksi masa studi mahasiswa berdasarkan faktor yang berhubungan dengan akademik mahasiswa
(Hasudungan & Pranoto, 2021)	Naïve Bayes	Nilai Akurasi 77,05%	Impementasi algoritma naïve bayes pada prediksi prestasi mahasiswa
(Salmu & Solichin, 2017)	Naïve Bayes	Nilai Akurasi 80,72%	Menggunakan naïve bayes untuk memprediksi kelulsan mahasiswa tepat waktu

2.3 Teori Rough Set

Rough Set teori ini pertama kali diperkenalkan oleh Pawlak, yang menyatakan bahwa Rough set merupakan salah satu metode matematika untuk menangani data yang tidak konsisten dan samar (Pawlak, 1982). Selain itu, keunggulan dari metode ini ialah tidak memerlukan parameter atau masukan karena informasi terkait dengan data diambil dari data itu sendiri (Pawlak, 1991). Serta Pawlak mengusulkan bahwa teori himpunan kasar didirikan pada asumsi bahwa dengan setiap anggota alam semesta wacana kita menghubungkan beberapa informasi. Konsep himpunan kasar adalah teknik matematika baru untuk mengatasi ketidakjelasan, ketidaktepatan, dan ketidakpastian (Pawlak & Skowron, 2007. Berikut alur penyelesaian algoritma rough set dapat dilihat pada bagan 2.1



Gambar 0.1 Alur penyelesaian rough set

Berikut keterangan berdasarkan alur penyelesaian algoritma rough set sebagai penyelesain : (Senan et al., 2011)

1. *Information Table* adalah tabel yang terdiri dari kolom dan baris yang berisikan data, dimana kolom diberikan label attribut, dan baris akan diisi nilai dari attribut. Dengan informasi sistem seperti S = (U,A,V, *f*), dimana U adalah himpunan dari objek, A adalah himpunan attribut yang tidak boleh kosong, $V=U_{a\in A}V_a$, V_a adalah domain attribut A, f:U×A → 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 attribut keputusan *(Decision information system)* yang tidak boleh memiliki nilai kosong. Dengan infomasi sistem sebagai berikut D = (U, A U {d}, V, *f*, dimana U, A, V dan *f* sesuai dengan D dan {d} adalah attribut keputusan dimana {d} ∩ *A* ≠ ∅).

2. *Indescernibility Relation* ialah 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 ∈ U dikatakan *B-indescernible*(tidak dapat dibedakan oleh himpunan attribut B ⊆ A dalam S) jika hanya *f*(x,a) = *f* (y,a) untuk setiap a ∈ B.
3. *Set Appromaximations* adalah mengelompokkan hasil dari *Indescernibility relation* yang digunakan untuk mendefinisikan approximations sebagai konsep dasar dalam algoritma rough set, untuk menentukan perkiraan terbawah dan perkiraan teratas dalam suatu himpunan 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.

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

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

4. *Dependency of Attributes* ialah langkah menghitung konsistensi setiap attribut 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≤k≤1), dengan notasi C $\Rightarrow_k$ D. Maka dapat didefiniskan dengan persamaan 2.3.

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

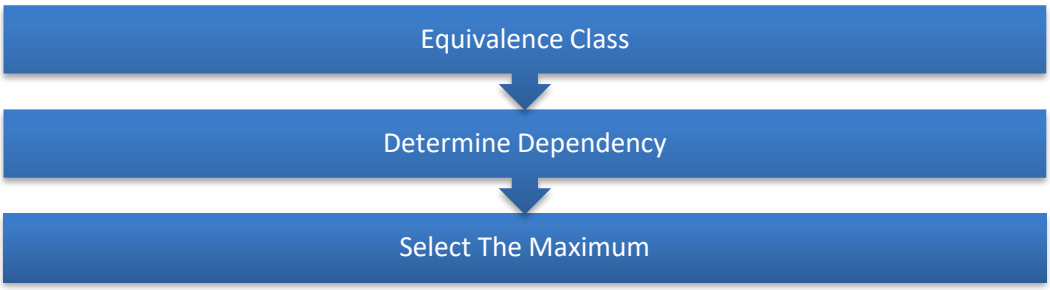
5. *Reduct* merupakan proses meminimalisir himpunan dari attribut. Dengan cara menghitung ulang menggunakan langkah-langkah sebelumnya untuk diterapkan di setiap attribut yang ada, sehingga mendapatkan attribut terbaik dan tidak mengurangi nilai konsistensi attribut tersebut. Dengan definisi sebagai berikut S = (U,A,V, f) menjadi sistem informasi, dan B menjadi bagian dari A, jika B berpengaruh pada konsistensi attribut menjadi berlebihan dapat dibuang dengan notasi B if U / (B – {b}) = U / B, jika tidak mempengaruhi konsistensi attribut maka sangat diperlukan. Berikut adalah tabel penelitian terdahulu yang menggunakan roughset sebagai fitur seleksi atribut seperti tertera pada tabel 2.3.

Tabel 0.3 Penelitian Terdahulu Rough Set

Author	Metode	Hasilnya	Keterangan
(Hasudungan & Pranoto, 2021)	Rough set	Nilai Akurasi 68,09%	Menerapkan naïve bayes model untuk analisis data mahasiswa
(Raharjo & Windarto, 2021)	Rough set	Hasil nya 90 rules	Penerapan maching learning dengan konsep data meaning roughset untuk memprediksi tingkat pemahaman mahasiswa terhadap mata kuiah
(Samaray, 2022)	Rough Set	Hasilnya 14 rules	Implementasi algoritma rough set dengan softwere rosetta untuk prediksi hasil belajar
(Aziz, 2020)	Rough Set	Hasil Akurasi 92%	Iplementasi algoritma rough set dana naïve bayes untuk mendapat rule dalam menyeleksi pemohon bantuan fasilitas rumah ibadah
(Putra et al., 2018)	Rough Set	Hasilnya 13 rules	Implementasi algoritma rough set dalam memprediksi kecerdasan anak

2.4 Maximum Dependency of Attributes

Metode Maximum dependency attributes ialah algoritma 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 Alur penyelesaian MDA

Berikut keterangan 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.

$$D(X) \subseteq C(X) \subseteq X \subset \bar{C}(X) \subseteq \bar{D}X \qquad 2.4$$

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

$$D\left(\underline{R}(X), \bar{R}(X)\right) = 1 - \frac{|\underline{R}(X) \cap \bar{R}(X)|}{|\underline{R}(X) \cup \bar{R}(X)|}, = 1 - \frac{|\underline{R}(X)|}{|\bar{R}(X)|}, = 1 - \alpha R(X) \quad 2.5$$

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 $C1$, $C2$, … , Cn sehingga D menjadi bagian dari A. Jika $C1 \Rightarrow k1\ D$, $C2 \Rightarrow k2\ D$, … $Cn \Rightarrow kn\ D$, dimana $kn \leq kn{-}1 \leq \cdots \leq k2 \leq k1$, sehingga $\alpha D\ (X) \leq \alpha Cn\ (X) \leq \alpha Cn{-}1\ (X) \leq \cdots \leq \alpha C2\ (X) \leq \alpha C1\ (X)$ For every $X \subseteq$ U. Adapun terdapat pada persamaan 2.6.

$$\alpha D\ (X) \leq \alpha Cn\ (X) \quad | \quad kn \leq kn\text{-}1 \leq \cdots \leq k2 \leq k1 \quad | \quad [x]Cn \subseteq [x]Cn\text{-}1 \qquad 2.6$$

2.5 Pemprosesan Data

Pemprosesan data merupakan tahapan di dalam data maining yang digunakan untuk memproses data agar dapat dijalankan pada proses klasifikasi. Pemprosesan data memiliki tujuan untuk mengurangi data, membuang outliers, dan mengekstrak pengetahuan, untuk itu memerlukan beberapa teknik yaitu pembersihan data, integrasi data, transformasi data, reduksi data. Berikut keterangan pemprosesan data : (Suad A. Alasadi & Wesam S. Bhaya, 2017)

1. Pembersihan data adalah proses 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. Integrasi data adalah teknik untuk mengkombinasikan data dari berbagai sumber data menjadi satu data yang konsisten seperti data pergudangan yang memiliki sumber dari berbagai tempat.
3. Transfomasi data adalah proses transformasi data dengan mengubah tipe data numerik menjadi kategoril agar data dapat diproses saat klasifikasi.
4. Reduksi data adalah teknik yang digunakan untuk mereduksi data dalam jumlah besar dengan mempresentasikan data dalam jumlah kecil, dengan tetap mempertahankan integritas data yang asli.

2.6 Evaluasi

Evaluasi ialah suatu proses didalam analisa data untuk mengukur model yang telah dihasilkan. Terdapat banyak alat yang dapat digunakan untuk mengukur kinerja sebuah algoritma salah satunya menggunakan akurasi, pengukuran evaluasi pada peranan data maining klasifikasi adalah mengukur akurasi dan perhitungan akurasi berdasarkan pada confusion matrix. *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 2.4 dijelaskan contoh *confusion matrix* proses klasifikasi. Untuk mengukur akurasi *(accuracy)* pada model dapat menerapkan persamaan 2.5 yang digunakan untuk menghitung hasil akurasi, sedangkan untuk menghitung tingkat kesalahan *(error rate)* dapat didefinisikan dengan persamaan 2.5, dan untuk menghitung ketepatan *(precison)* mengukur data yang telah diprediksi positif dengan kenyataan yang benar dan tidak benar dapat menggunakan persamaan 2.6, Terakhir untuk menghitung sentivitas *(recall)* banyak data yang sukses saat diprediksi dengan perbandingan seluruh data yang pada kenyataanya positif dapat menggunakan persamaan 2.7.

Tabel 0.4 Contoh 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.

$$Akurasi = \frac{Jumlah\ Prediksi\ yang\ Benar}{Jumlah\ Prediksi\ yang\ dilakukan} = \frac{\mathrm{TP+TN}}{\mathrm{TP+FP+FN+TN}} \quad 2.7$$

$$Kesalahan = \frac{Jumlah\ Prediksi\ yang\ Salah}{Jumlah\ Prediksi\ yang\ dilakukan} = \frac{\mathrm{FP+FN}}{\mathrm{TP+FP+FN+TN}} \quad 2.8$$

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

$$Sentivitas = \frac{TP}{TP+FN} \quad 3.0$$