

## **BAB II**

### **TINJUAN PUSTAKA**

#### **2.1 Penelitian Terkait**

Penelitian yang dipublikasikan pada tahun 2023 oleh Novi Rustiana Dewi, Anita Desiani, Fitri Salamah dan Yuli Andriani yang berjudul “Algoritma *K-Nearest Neighbor* (K-NN) Dan (SLP) Untuk Klasifikasi Penyakit Alzheimer” Penelitian menggunakan metode *percentage split*. Data tersebut memiliki 10 atribut dan berjumlah 373 data tentang penyakit alzheimer. Hasil penelitian menunjukkan bahwa pengujian menggunakan algoritma K-NN menghasilkan akurasi sebesar 96%. Nilai presisi dan *recall* untuk kelas 0 (*Nondemented*) berturut-turut adalah 93% dan 100%, sedangkan untuk kelas 1 (*Demented*) adalah 100% dan 91%. Di sisi lain, pengujian menggunakan algoritma SLP menghasilkan akurasi sebesar 99%. Nilai presisi dan *recall* untuk kelas 0 (*Nondemented*) berturut-turut adalah 97% dan 100%, sedangkan untuk kelas 1 (*Demented*) adalah 100% dan 98% [8].

Penelitian dengan judul “Pendekatan Algoritma Klasifikasi Machine Learning untuk Deteksi Penyakit Demensia” pada tahun 2023 yang dilakukan oleh Muhammad Iqbal, Hendri Mahmud Nawawi, Muhammad Rezki, Abdul Hammid dan Sri Rahayu. Penelitian tersebut menggunakan algoritma klasifikasi seperti *extra trees* (ET), *linear discriminant analysis* (LDA), *random forest* (RF) dan *ridge*. Data ini berasal dari koleksi *longitudinal Open Access Series of Imaging Studies* (OASIS-2), yang mencakup 373 data MRI, yang mencakup 150 subjek. Hasil evaluasi menunjukkan bahwa algoritma *random forest* (RF) menghasilkan kinerja terbaik dengan akurasi sebesar 91.56%. Algoritma *extra trees* (ET) memiliki akurasi yang hampir sebanding dengan 91.44%, sementara *ridge* dan *linear discriminant analysis* (LDA) memiliki akurasi masing-masing sebesar 90.44%. Dalam konteks deteksi penyakit demensia, kinerja algoritma *random forest* dengan metode seleksi atribut terbukti menjadi yang terbaik dengan akurasi 91.56% [16].

Penelitian sebelumnya yang dilakukan oleh Firman Akbar dan Rahmaddeni pada tahun 2022 dengan judul “Komparasi Algoritma *Machine Learning* Untuk Memprediksi Penyakit Alzheimer” melakukan perbandingan kinerja berbagai algoritma klasifikasi untuk memprediksi penyakit alzheimer menggunakan *artificial neural network* (ANN), *naïve bayes*, *random forest*, *support vector machine* (SVM), *logistic regression*. Dataset yang digunakan dalam penelitian ini terdiri dari 373 data yang dikumpulkan dari situs *Kaggle Open Datasets*. Hasil penelitian menunjukkan bahwa algoritma *logistic regression* dapat menghasilkan nilai akurasi yang paling baik yaitu 85,71% [13].

Penelitian yang dilakukan oleh Priyo Wahyu Setiyo Aji, Supriyanto dan Rohman Dijaya dengan judul “ Prediksi Stroke Menggunakan Metode *Random forest*” pada tahun 2023 dengan data yang digunakan berasal dari dataset *Kaggle* terdiri dari 40910 *record*, yang terdiri dari 11 indeks stroke dengan metode *machine learning* yaitu *random forest*. Hasil prediksi akhir yang diperoleh melalui proses voting dengan tahapan preprocessing, processing dan evaluasi hasil menghasilkan akurasi sebesar 99% [14].

Penelitian terkait yang dilakukan pada tahun 2023 oleh Roudlotul Jannah Alfirdausy dan Saiful Bahri dengan judul “Implementasi Algoritma *K-Nearest Neighbor* untuk Klasifikasi Diagnosis Penyakit Alzheimer ” penelitian tersebut menggunakan metode KNN yang digunakan untuk mengklasifikasikan penyakit alzheimer menggunakan kumpulan data dari dataset *Kaggle* yang terdiri dari 331 data. Hasil klasifikasi yang dilakukan dengan membagi data *testing* dan *training* 70:30 menggunakan metode KNN dengan K=3, K=5 dan K=7 menghasilkan nilai akurasi sebesar 99%, sensitivitas sebesar 100%, serta spesifitas sebesar 96% [9].

Pada tahun 2021, Fandi Yulian Pamuji Dan Viry Puspaning Ramadhan melakukan penelitian yang berjudul “Komparasi Algoritma *Random Forest* Dan *Decision Tree* Untuk Memprediksi Keberhasilan Immunotherapy” dalam penelitian ini menggunakan metode *random forest* dan *decision tree*. dataset yang digunakan dalam penelitian ini adalah dataset immunotherapy yang terdiri dari 90 *record* yang terdiri dari delapan atribut dan satu atribut label, diperoleh dari UCI *machine learning repository*. Hasil penelitaian metode *decision tree* masih terdapat prediksi yang tidak tepat dengan tingkat akurasi 84,4 % kemudian metode *random*

*forest* prediksinya tepat dengan tingkat akurasi 85,5 %. hasil pengujian tersebut menunjukkan bahwa metode *random forest* merupakan metode yang lebih unggul dibandingkan dengan metode *decision tree* [15].

Penelitian pada tahun 2021 dengan judul “ Komparasi Algoritma *Random forest*, *Naïve Bayes* dan *K-Nearest Neighbor* Dalam klasifikasi Data Penyakit Jantung” yang dilakukan oleh Amril Samosir, Ms Hasibuan, Wahyu Eko Justino dan Tri Hariyono. Tujuan penelitian tersebut adalah untuk membandingkan ketiga algoritma untuk mengklasifikasikan data tentang penyakit jantung. Untuk melihat perbandingan algoritma, metrik kinerja yang terdiri dari tingkat akurasi, *recall*, dan presisi setiap kelas akan digunakan. Setiap algoritma diuji dengan *cross-validation*. Menurut hasil perbandingan terhadap 304 dataset penyakit jantung, Hasil komparasi dari tiga algoritma menghasilkan nilai skor yang berbeda. Algoritma *naive bayes* mencapai 0,91 AUC, 0,84 CA, 0,84 F1, 0,839 *Precision* dan 0,84 *Recall*. Algoritma *Random forest* mencapai 0,884 AUC, 0,788 CA, 0,787 F1, 0,788 *Precision*, dan 0,788 *Recall*. Algoritma KNN mencapai 0,686 AUC, 0,645 CA, 0,641 F1, 0,642 *Precision*, dan 0,645 *Recall*. Algoritma *naive bayes* dianggap optimal untuk mengklasifikasikan penyakit jantung dan dianggap lebih baik dari pada algoritma *KNN* dan *random forest*. Hasil klasifikasi dengan algoritma *naive bayes* menunjukkan rerata akurasi sebesar 0,91 AUC, 0,84 CA, 0,84 F1, 0,839 Keakuratan, dan 0,84 *Recall* [17].

Penelitian yang dilakukan oleh Imaniar Ikko Mulya Rizky, Suhendro Yusuf Irianto dan Sriyanto pada tahun 2023 dengan judul “Perbandingan Kinerja Algoritma *Naive Bayes*, *Support Vector Machine* dan *Random forest* untuk Prediksi Penyakit Ginjal Kronis” data penelitian terdiri dari 400 *record*, 25 atribut dan target class adalah penyakit ginjal kronis (PGK) bersumber dari UCI. Tujuan penelitian ini adalah untuk membandingkan kinerja tiga algoritma: *naive bayes*, SVM, dan *random forest*. Hasil pengujian menunjukkan bahwa pemrosesan klasifikasi dengan algoritma *naive bayes* sebesar 97.14%, pemrosesan klasifikasi dengan algoritma *support vector machine* sebesar 92.50%, dan pemrosesan klasifikasi dengan algoritma *random forest* sebesar 99.64% [18].

Penelitian yang dilakukan oleh Martika Kesuma, Sriyanto dan Sutedi pada tahun 2023 yang berjudul “Prediksi Penyakit Liver Menggunakan Algoritma *Random*

*forest*” tujuan penelitian ini adalah untuk mengetahui hasil prediksi algoritma *random forest* dengan jumlah dataset yang digunakan berjumlah 583 data, Pemodelan algoritma ini memiliki nilai akurasi 0.713326 dan skor f1 81%. yang memiliki nilai akurasi cukup baik untuk melakukan prediksi tentang penyakit liver [19].

Penelitian pada tahun 2024 yang dilakukan oleh Neni Purwati, Windya Harieska Pramujati, Syakur dan Egi Safitri yang berjudul “Prediksi Pasien Pusat Kesehatan Masyarakat Menggunakan *Machine Learning*” Untuk melakukan prediksi, data yang digunakan adalah dataset kunjungan pasien berobat rawat jalan di Puskesmas Hanura, yang terletak di desa Hanura, provinsi Lampung, yang terdiri dari 500 *record* dan 6 atribut. Akurasi *random forest* (RF) dan *extreme gradient boosting* (XGBoost) digunakan dalam tahapan pemisahan data *train* dan *test* dengan presentase 70% untuk data *train* dan 30% untuk data *test*. Hasilnya adalah 0.69 untuk RF dan 0.93 untuk XGBoost. Hasil *matrix confusion* XGBoost menunjukkan nilai positif benar (TP), yang berarti data yang diprediksi dengan tepat dan benar sebanyak 53, *false negative* (FN) bernilai 3, *false positive* (FP) bernilai 2 dan 1, *true negative* (TN) bernilai 40, 4, 1, 46, dan nilai F1-Score juga 0.93. Hasil menunjukkan bahwa model yang digunakan berkinerja baik dan layak digunakan [20].

Penelitian dengan judul “Prediksi Penyakit Diabetes Melitus Menggunakan Algoritma *Machine Learning*” yang dilakukan oleh Egi Safitri, Dani Rofianto, Neni Purwati, Hendra Kurniawan dan Sri Karnila pada tahun 2024 untuk mengembangkan model prediksi yang akurat untuk diabetes melitus menggunakan tiga algoritma machine learning: *random forest*, *regresi logistik*, dan *decision tree*. Dataset Pima Indians Diabetes 768 catatan pasien Hasil penelitian menunjukkan bahwa regresi logistik mencapai akurasi tertinggi (75%) dan kinerja yang seimbang dalam mengidentifikasi kasus positif dan negatif. *decision tree* unggul dalam *recall*, sementara *random forest* menunjukkan keseimbangan yang sedikit lebih rendah antara *presisi* dan *recall*. Analisis kurva ROC mengungkapkan bahwa *random forest* memiliki AUC tertinggi (0.82), diikuti oleh *regresi logistik* (0.81) dan *decision tree* (0.73). dengan berbagai indikator kesehatan, digunakan untuk pelatihan dan evaluasi model [21].

Penelitian dengan judul "Alzheimer's Disease Detection Using m-Random Forest Algorithm with Optimum Features Extraction" yang dilakukan pada tahun 2021 oleh Md Shahin Ali, Md. Khairul Islam, Jahurul Haque, A Arjan Das, D S Duranta, dan Md Ariful Islam. Dataset diperoleh dari Open Access Series of Imaging Studies (OASIS-2), yang terdiri dari 150 subjek berusia antara 60 hingga 98 tahun. Dataset ini memiliki tiga kategori diagnosis yaitu non-demented, demented, dan converted. Menggunakan algoritma m-Random Forest, sebuah modifikasi dari algoritma Random Forest, yang diklaim mampu mendeteksi penyakit Alzheimer dengan lebih akurat, terutama pada tahap awal penyakit. Hasil pengujian menunjukkan bahwa model ini mampu mencapai akurasi sebesar 96.43%, melampaui model lain seperti AdaBoost (82.5%), Random Forest biasa (84%), dan model deep learning (87.76%) [2].

Tabel 2. 1 Penelitian Terkait

No	Judul penelitian	Metode	Evaluasi	Jumlah Dataset
1	Algoritma <i>K-Nearest Neighbor</i> ( <i>K-Nn</i> ) Dan <i>Single Layer Perceptron</i> ( <i>Slp</i> ) Untuk Klasifikasi Penyakit Alzheimer	<i>K-Nearest Neighbor</i> ( <i>K-NN</i> ) dan <i>Single Layer Perceptron</i> ( <i>SLP</i> )	KNN = 96% SLP = 99%.	373 Data
2	Pendekatan Algoritma Klasifikasi <i>Machine Learning</i> untuk Deteksi Penyakit Demensia	<i>Random forest</i> ( <i>RF</i> ), <i>Extra Trees</i> ( <i>ET</i> ), <i>Ridge</i> dan <i>Linear Discriminant Analysis</i> ( <i>LDA</i> )	<i>Random forest</i> 91.56% <i>Extra Trees</i> 91.44% <i>Ridge</i> dan <i>Linear Discriminant Analysis</i> 90.44%	373 data
3	Komparasi Algoritma <i>Machine Learning</i> Untuk Memprediksi Penyakit Alzheimer	<i>Artificial Neural Network</i> , <i>Naïve Bayes</i> , <i>Logistic Regression</i> , <i>Support Vector Machine</i>	<i>Logistic Regression</i> 85,71%, <i>Naïve Bayes</i> 83,04%, <i>Artificial Neural Network</i> 46,43%, <i>Random forest</i> 85,71%, <i>Support Vector Machine</i> 85,71%	373 data
4	Prediksi Penyakit Stroke	<i>Random forest</i>	99%	40910 data

	Menggunakan Metode <i>Random forest</i>			
5	Implementasi Algoritma <i>K-Nearest Neighbor</i> untuk Klasifikasi Diagnosis Penyakit Alzheimer	<i>K-Nearest Neighbor</i>	99%	331 data
6	Komparasi Algoritma <i>Random forest</i> Dan <i>Decision Tree</i> Untuk Memprediksi Keberhasilan Immunotherapy	<i>Random forest</i> <i>Decision Tree</i>	<i>Random forest</i> 85,5% <i>Decision tree</i> 84,4%	90 data
7	<i>Alzheimer's Disease Detection Using m-Random Forest Algorithm with Optimum Features Extraction</i>	<i>m-Random Forest</i>	96.43%	150 Data

## 2.2 Landasan Teori

### a. *Alzheimer Disease (AD)*

*Alzheimer disease (AD)* adalah gangguan *neurologis degeneratif* yang mempengaruhi otak secara permanen. Setiap penyakit yang secara signifikan mengganggu fungsi sistem saraf pusat dan otak disebut sebagai gangguan *neurologis*. AD merupakan penyakit otak *neurodegeneratif* jangka panjang yang

ditandai dengan kehilangan memori, perubahan perilaku, gangguan penilaian, dan masalah bahasa. Ini adalah jenis demensia yang berbahaya yang menyerang orang lanjut usia [22]. Beberapa faktor risiko AD diantaranya seperti usia, cedera kepala traumatis, depresi, penyakit jantung, merokok, dan riwayat demensia dalam keluarga [7].

Tabel 2. 2 Jenis – jenis *Alzheimer Disease*

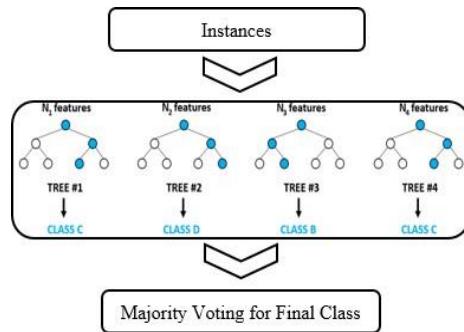
<b>Jenis Alzheimer</b>	<b>Definisi</b>	<b>Karakteristik</b>
<i>Early onset familial</i>	Mutasi genetik pada gen PS1, PS2, dan APP	Cenderung muncul pada individu yang lebih muda, biasanya sebelum usia 65 tahun, dan memiliki komponen genetik yang kuat
<i>Late-onset sporadic</i>	Dipengaruhi oleh faktor risiko lain seperti usia, jenis kelamin, dan riwayat keluarga	Muncul setelah usia 65 tahun

*Alzheimer disease* (AD) adalah penyakit otak yang berkembang secara perlahan yang dimulai bertahun-tahun sebelum gejalanya muncul. Kesulitan mengingat percakapan, nama, atau peristiwa yang baru saja terjadi sering kali merupakan gejala klinis awal, apatis dan depresi juga sering kali merupakan gejala awal. Gejala selanjutnya meliputi gangguan komunikasi, disorientasi, kebingungan, penilaian yang buruk, perubahan perilaku dan, pada akhirnya, kesulitan berbicara, menelan dan berjalan [5].

### b. *Random forest*

*Random forest* adalah algoritma pembelajaran mesin yang digunakan untuk mengklasifikasikan banyak dataset. Karena fungsinya dapat digunakan dengan berbagai dimensi, skala, dan tingkat kinerja yang tinggi. *Random forest* digunakan untuk berbagai industri, termasuk *e-commerce*, analis keuangan,

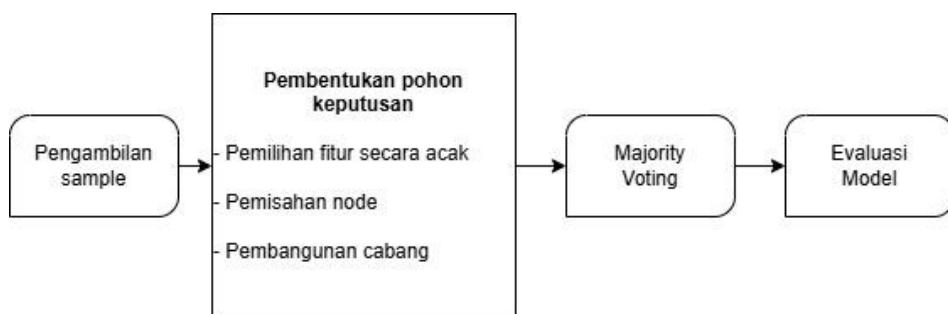
perbankan, dan kesehatan. Algoritma ini bekerja dengan membuat beberapa pohon keputusan dan menyatukannya untuk membuat prediksi yang lebih akurat dan stabil. Algoritma tersebut membuat Hutan (*Forest*) dari kumpulan pohon keputusan yang dilatih dengan metode *bagging*. Ini diharapkan akan menghasilkan model pembelajaran mesin yang lebih baik [23]. *Random forest* adalah turunan dari algoritma *decision tree* untuk mengatasi masalah dengan kemungkinan *overfitting* yang tinggi. Algoritma *Random forest* direpresentasikan pada gambar 2.1



Gambar 2. 1 Algoritma *Random Forest*

Untuk menghasilkan sejumlah n pohon keputusan, algoritma pohon acak pertama kali akan mengambil nilai atribut *instance* data yang akan diproses. Selama tahap pembangkitan pohon keputusan, dua parameter yang sangat penting diperlukan. Parameter pertama adalah jumlah pohon yang akan dibangkitkan (n), dan parameter kedua adalah jumlah fitur maksimal yang dipertimbangkan selama proses percabangan saat membangkitkan setiap pohon keputusan (k) [24].

Berikut mekanisme cara kerja *random forest* [24]:



Gambar 2. 2 Cara Kerja *Random Forest*

## 1. Pengambilan Sampel Data

Pengambilan sampel acak *random forest* dimulai dengan dari dataset pelatihan awal untuk membentuk beberapa subset data dengan tiap *subset* ini dapat berisi duplikasi dari data asli dan beberapa data mungkin tidak terpilih.

## 2. Pembentukan Pohon Keputusan

Dibangun untuk setiap *subset* data. Proses pembentukan pohon terdiri dari tahapan berikut:

### a) Pemilihan Fitur secara Acak

Sejumlah fitur dipilih secara acak dari semua fitur yang tersedia di setiap *node* (simpul) pohon, ini berbeda dari pohon keputusan biasa yang menggunakan semua fitur untuk memecahkan *node*.

### b) Pemisahan *Node*

Algoritma menggunakan *Gini Indeks* untuk klasifikasi data untuk menemukan fitur dan titik pemisahan terbaik, yang memaksimalkan pemisahan data berdasarkan kriteria tertentu menggunakan *Gini Index* untuk klasifikasi.

$$\begin{aligned} \text{Gini Index} &= 1 - \sum_{i=1}^n (p_i)^2 \\ &= 1 - [(P_+)^2 + (P_-)^2] \end{aligned}$$

Keterangan :

n = Jumlah dari masing-masing atribut

Pi = jumlah atribut dari masing-masing kelas atau labelnya

P+ = Probabilitas Positif Class

P- = Probabilitas Negatif Class

### c) Pembangunan cabang

*Node* tersebut terbagi menjadi dua cabang, dan proses ini berulang sampai pohon mencapai kedalaman tertentu atau *node* tidak dapat dipecah lagi (misalnya, semua data dalam *node* adalah homogen).

### 3. Majority Voting

Untuk menghasilkan hasil prediksi secara keseluruhan, *Random forest* akan melakukan proses *majority voting*. Proses ini didasarkan pada hasil prediksi dari setiap n pohon keputusan yang dibuat.

### 4. Evaluasi Model

Tujuan dari evaluasi model *random forest* yaitu untuk mengukur kinerja dan keakuratan model yang telah dilatih. Evaluasi model memungkinkan untuk memahami seberapa baik model bekerja dalam memprediksi data.

#### c. Metrik Evaluasi

Pada tahap ini dilakukan pada model untuk menilai kinerja algoritma yang dilatih. Dengan tabel *confusion matriks* digunakan untuk menguji kinerja model klasifikasi dalam memprediksi kelas target. Metode ini menunjukkan kemampuan model untuk mengidentifikasi kesalahan dan mengklasifikasikan data dengan benar [25].

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Gambar 2. 3 *Confusion Metrik*

Keterangan :

1. *True Positive* (TP) adalah jumlah kasus yang diklasifikasikan dengan benar sebagai positif oleh model.
2. *True Negative* (TN) adalah jumlah kasus yang diklasifikasikan dengan benar sebagai negatif oleh model.
3. *False Positive* (FP) adalah jumlah kasus yang diklasifikasikan dengan salah sebagai positif oleh model, meskipun sebenarnya negatif.
4. *False Negative* (FN) adalah jumlah kasus yang diklasifikasikan dengan salah sebagai negatif oleh model walaupun sebenarnya negatif.

Berdasarkan *Confusion Matrix*, Berikut rumus untuk menghitung matrik evaluasi klasifikasi:

- a) *Accuracy* megukur seberapa sering model memberikan prediksi yang benar secara keseluruhan. Adapun rumus dari *accuracy* yaitu :

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \times 100\%$$

- b) *Precision* mengukur sejauh mana model memberikan hasil positif yang benar dari semua hasil positif yang diberikan. Adapun rumus dari *Precision* yaitu :

$$\text{Precision} = \frac{TP}{TP+FP} \times 100\%$$

- c) *Recall* atau *Sensitivity* mengukur seberapa baik model dapat mengidentifikasi kasus positif yang sebenarnya. Adapun rumusan *Recall* atau *Sensitivity* yaitu:

$$\text{Recall} = \frac{TP}{TP+FN} \times 100\%$$

- d) *F1-Score* menggambarkan perbandingan *precision* dan *recall*. Jika kumpulan data berisi jumlah data *False Negative* dan *False Positive*

yang sangat mirip, namun jika angkanya tidak dekat, maka menggunakan F1-Score sebagai referensi. Adapun rumusan F1-Score:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

#### **d. Machine Learning**

*Machine learning* adalah pendekatan *artificial intelligence* yang berfokus pada pembuatan mesin (robot) yang dapat belajar tanpa diprogram secara khusus. Ini adalah teknik untuk melakukan inferensi data menggunakan pendekatan matematis. Maksud dari kesimpulan ini adalah lebih berkonsentrasi pada hubungan antar atribut [26]. Dalam beberapa dekade terakhir, *Machine learning* telah menjadi salah satu alat yang paling populer untuk digunakan dalam pekerjaan yang membutuhkan ekstraksi informasi dari suatu kumpulan data dengan cara yang sama dan untuk melakukan prediksi. *Machine learning* adalah kumpulan teknik yang memungkinkan penggunaannya untuk melaksanakan implementasi algoritma yang memiliki sifat yang dapat disesuaikan untuk mengorganisir data masukkan berdasarkan fitur yang sama secara otomatis. Sebenarnya, istilah pembelajaran mesin mengacu pada suatu otomasi yang dapat mendeteksi pola penting dalam data [27].

#### **e. Data Mining**

Data mining atau juga dinamakan *Knowledge Discovery in Database* (KDD) merupakan bidang ilmu yang banyak membahas pola data [28]. Data mining adalah bidang keilmuan yang terdiri dari banyak cabang keilmuan menggabungkan berbagai metode, seperti mempelajari mesin, memahami pola, dan menganalisis yang bersifat otomatis dalam kasus kumpulan data yang besar atau kompleks untuk mengumpulkan data seperti pola dan hubungan antar data, yang kemudian dapat digunakan untuk menghasilkan visualisasi, statistik, atau basis data [27]. Data mining adalah proses mengekstrak jumlah data yang sangat besar yang

sebelumnya tidak diketahui [29]. Data mining merupakan proses menemukan pola, hubungan, dan kecenderungan penting dalam sekumpulan data yang besar yang disimpan dalam penyimpanan dengan menggunakan teknik pengenalan pola [30].

#### f. Google Colaboratory

*Google Colaboratory* adalah layanan komputasi cloud yang disediakan oleh *Google* untuk mendukung pengembangan dan penelitian ilmiah. Produk kerja sama, atau "Colab", adalah produk dari penelitian *Google*. *Colab* memungkinkan pengguna menggunakan *browser* untuk menulis dan mengeksekusi kode *Python*, dan sangat cocok untuk pendidikan, analisis data, dan *machine learning* [31].

#### g. Streamlit

Streamlit merupakan sebuah framework open-source berbasis *Python*, memungkinkan pengembang membuat dan berbagi aplikasi data interaktif dengan mudah. Dengan Streamlit, pengguna dapat dengan cepat membuat aplikasi web yang menampilkan visualisasi data, analisis, dan model pembelajaran mesin tanpa memerlukan keahlian dalam pengembangan web yang mendalam. Ini menyederhanakan proses pembuatan aplikasi berbasis data dengan menyediakan *tools* dan komponen yang dapat digunakan untuk membangun antarmuka pengguna yang menarik dan interaktif. Dengan berfokus pada data, Streamlit memungkinkan pengembang berkonsentrasi pada logika bisnis dan analisis sambil menangani aspek teknis pengembangan web [32].