Comparison of Machine Learning Algorithms in Predivting Student
[Perbandingan Algoritma Machine Learning Dalam Memprediksi Kelulusan Mahasiwa]

M Cholis Afandi[1)], Uce Indahyanti[2)]

[1,)] Program Studi Teknik Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

[2,)] Program Studi Teknik Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

Email Penulis Korespondensi: uceindahyanti@umsida.ac.id

Abstract. *This study aims to predict student graduation in the Informatics Study Program at Universitas Muhammadiyah Sidoarjo using Machine Learning classification algorithms, namely Naïve Bayes, Decision Tree, and Random Forest. The dataset consists of academic records from the 2020–2021 cohort, including GPA scores and the number of credits (SKS) taken from semesters 1 to 6. The data analysis process follows the CRISP-DM methodology, covering business understanding, data understanding, data preparation, modeling, evaluation, and deployment. Model evaluation is carried out using confusion matrices along with accuracy, precision, recall, and F1-score to compare the performance of each algorithm. The results show that Random Forest achieved the highest accuracy of 97.50% in the 80:20 scenario, followed by Decision Tree at 96.25%, and Naïve Bayes at 86.25%. In addition, Random Forest demonstrated high precision and recall values with an F1-score of 97%, confirming its stability and effectiveness in academic data classification. Based on these findings, Random Forest is considered the most optimal algorithm and is recommended as a decision support tool for accurately monitoring and predicting student graduation in higher education institutions.*

Keywords – *Data Mining, Graduation Prediction, Machine Learning, CRISP-DM, Confusion Matrix.*

Abstrak.. *Penelitian ini bertujuan untuk memprediksi kelulusan mahasiswa Program Studi Informatika Universitas Muhammadiyah Sidoarjo menggunakan algoritma klasifikasi Machine Learning, yaitu Naïve Bayes, Decision Tree, dan Random Forest. Data yang digunakan merupakan data akademik mahasiswa angkatan 2020–2021, mencakup nilai IPS dan jumlah SKS dari semester 1 hingga 6. Proses analisis mengikuti tahapan CRISP-DM, mulai dari pemahaman bisnis hingga evaluasi model. Evaluasi dilakukan menggunakan confusion matrix serta pengukuran akurasi, presisi, recall, dan F1-score untuk membandingkan performa tiap algoritma. Hasil menunjukkan bahwa Random Forest memiliki akurasi tertinggi yaitu 97.50% pada skenario 80:20, disusul Decision Tree dengan 96.25%, dan Naïve Bayes sebesar 86.25%. Selain itu, Random Forest juga mencatatkan nilai presisi dan recall yang tinggi serta F1-score sebesar 97%, menunjukkan kestabilan dan keunggulan model dalam menangani data akademik. Berdasarkan temuan ini, Random Forest dinilai paling optimal dan direkomendasikan untuk digunakan sebagai sistem pendukung keputusan dalam memantau kelulusan mahasiswa secara prediktif dan akurat.*

Kata Kunci – *Data Mining, Prediksi Kelulusan, Machine Learning, CRISP-DM, Confusion Matrix.*

I. PENDAHULUAN

Pendidikan tinggi memiliki peranan yang sangat strategis dalam menghasilkan sumber daya manusia yang berkualitas untuk pembangunan bangsa. Salah satu indikator utama dalam menilai keberhasilan perguruan tinggi adalah tingkat kelulusan mahasiswa. Jika tingkat kelulusan rendah, hal tersebut dapat mencerminkan adanya masalah pada proses pembelajaran, kurikulum, sistem administrasi akademik, atau faktor eksternal lainnya yang mempengaruhi mahasiswa [1]. Oleh karena itu, penting untuk mengembangkan sistem prediksi kelulusan yang dapat memberikan intervensi tepat waktu guna meningkatkan kualitas pendidikan di perguruan tinggi.

Dengan kemajuan teknologi informasi dan data science, semakin banyak institusi pendidikan yang memanfaatkan algoritma Machine Learning (ML) untuk menganalisis dan memprediksi kelulusan mahasiswa berdasarkan data akademik, seperti Indeks Prestasi Kumulatif (IPK), jumlah Satuan Kredit Semester (SKS) yang telah ditempuh, serta faktor-faktor pendukung lainnya [2].

Sejumlah penelitian terdahulu telah membuktikan bahwa algoritma seperti Naïve Bayes, Decision Tree, dan Random Forest mampu memberikan hasil prediksi yang akurat terhadap status kelulusan mahasiswa. Model-model tersebut bekerja dengan memanfaatkan berbagai atribut, baik akademik maupun non-akademik, seperti IPK, jumlah SKS, nilai per semester, kehadiran, dan keterlibatan mahasiswa dalam kegiatan kampus. Pendekatan ini tidak hanya meningkatkan efisiensi analisis data akademik, tetapi juga memberikan dukungan penting dalam pengambilan keputusan untuk meningkatkan mutu pendidikan di perguruan tinggi [3]. Melalui penggunaan algoritma-algoritma ini, institusi pendidikan dapat melakukan analisis prediktif untuk mengetahui kemungkinan kelulusan mahasiswa dengan lebih akurat dan efisien. Beberapa algoritma Machine Learning yang sering diterapkan dalam prediksi kelulusan mahasiswa adalah Decision Tree, Random Forest, dan Naïve Bayes. Decision Tree merupakan metode yang populer karena kemampuannya dalam menghasilkan model yang mudah dipahami dan diterjemahkan dalam bentuk pohon keputusan [4]. Meskipun begitu, algoritma ini cenderung rentan terhadap masalah overfitting jika diterapkan pada dataset yang kompleks.

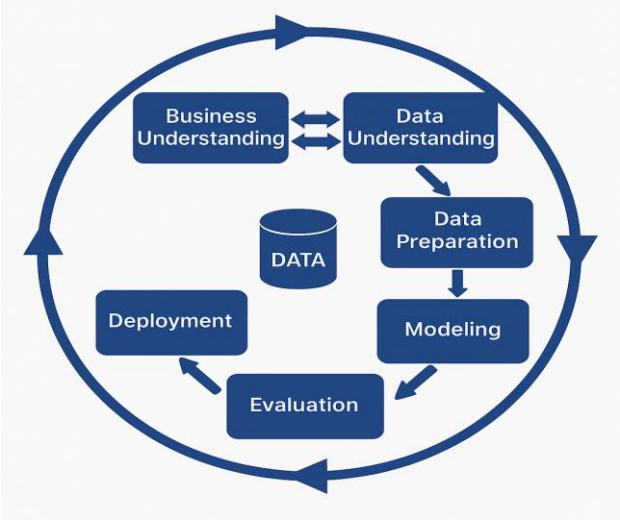
Untuk mengatasi keterbatasan tersebut, Random Forest dikembangkan sebagai metode ensemble yang menggabungkan beberapa pohon keputusan untuk meningkatkan akurasi dan mengurangi risiko overfitting [5]. Random Forest memiliki keunggulan dalam menangani data yang lebih besar dan lebih kompleks, serta mampu memberikan prediksi yang lebih andal. Sementara itu, Naïve Bayes, meskipun lebih sederhana, tetap menunjukkan performa yang kompetitif dalam berbagai kasus prediksi akademik. Algoritma ini didasarkan pada prinsip probabilistik dengan asumsi independensi antar variabel, yang memungkinkan proses pelatihan yang cepat dan efektif, terutama pada dataset yang besar [6].

Berdasarkan kajian literatur yang ada, Random Forest seringkali memberikan hasil yang lebih akurat dibandingkan dengan algoritma lain, terutama dalam konteks prediksi kelulusan mahasiswa di lingkungan pendidikan tinggi [7]. Oleh karena itu, penting untuk melakukan penelitian yang lebih mendalam untuk membandingkan kinerja ketiga algoritma ini dalam prediksi kelulusan mahasiswa di Program Studi Informatika Universitas Muhammadiyah Sidoarjo.

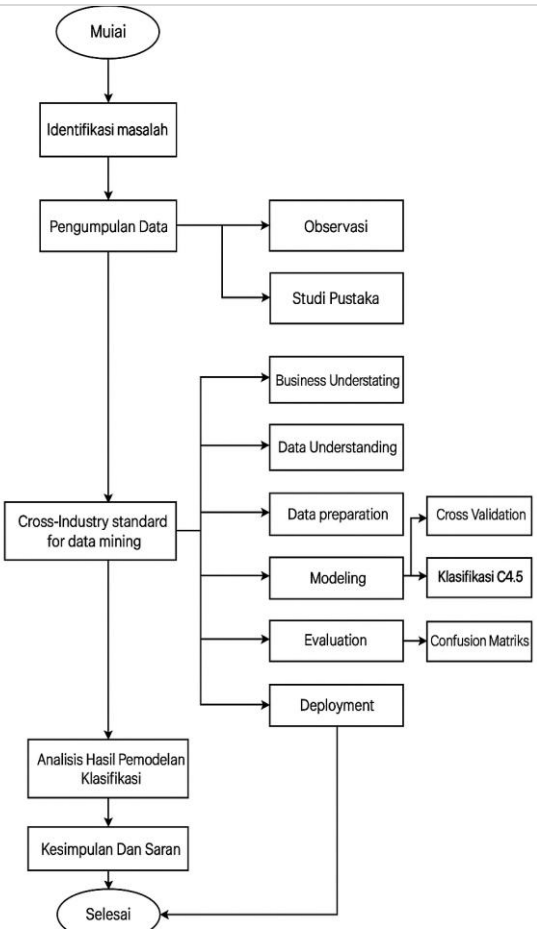
Penelitian ini bertujuan untuk membandingkan kinerja Naïve Bayes, Decision Tree, dan Random Forest dalam memprediksi kelulusan mahasiswa Program Studi Informatika Universitas Muhammadiyah Sidoarjo. Hasil penelitian ini diharapkan dapat memberikan informasi yang berguna bagi pengelola pendidikan tinggi dalam meningkatkan sistem prediksi kelulusan dan mendukung pengambilan keputusan berbasis data.

II. METODE

Bab ini menjelaskan tahapan penelitian yang akan dilaksanakan, Metodologi dan alur diagram penelitian. Metodologi yang digunakan pada penelitian adalah CRISP-DM (Cross Industry Standard Process for Data Mining) dengan enam tahapan: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, dan Deployment. Dan juga sebagaimana diilustrasikan pada gambar 1 dan 2 berikut :



Gambar 1. Diagram Alur Penelitian



Gambar 2. Alur Penelitian

Identifikasi Masalah

Tingkat kelulusan mahasiswa Program Studi Informatika Universitas Muhammadiyah Sidoarjo masih belum mencapai hasil yang optimal. Selain itu, belum tersedia sistem prediksi yang mampu mengidentifikasi secara dini mahasiswa yang berisiko tidak lulus tepat waktu. Kondisi ini menjadi kendala dalam upaya peningkatan mutu akademik. Oleh karena itu, diperlukan penerapan algoritma *machine learning* untuk membangun model prediktif yang akurat guna mendukung pengambilan keputusan akademik secara lebih efektif.

Studi Literatur

Tahap studi literatur dalam penelitian ini diawali dengan pengumpulan referensi dari penelitian-penelitian sebelumnya yang relevan. Proses ini melibatkan pencarian sistematis melalui berbagai sumber, termasuk basis data ilmiah, jurnal, buku teks, serta publikasi lainnya. Tujuan dari studi literatur ini adalah untuk mengidentifikasi tren penelitian, metodologi yang umum digunakan. Hasil dari studi literatur ini kemudian akan dianalisis, disintesis, dan diintegrasikan untuk menghasilkan tinjauan pustaka yang komprehensif dan relevan dengan topik penelitian.

Business Understanding

Tujuan dari tahap ini adalah memahami masalah bisnis yang ingin diselesaikan. Fokus utama penelitian ini adalah untuk meningkatkan tingkat kelulusan mahasiswa program studi informatika universitas muhammadiyah sidoarjo. dengan menggunakan algoritma *Machine Learning* [8].

Data Understanding

Data yang digunakan dalam penelitian ini adalah data akademik mahasiswa dari Program Studi Informatika Universitas Muhammadiyah Sidoarjo angkatan 2020-2021, yang terdiri dari informasi Indeks Prestasi Kumulatif (IPK) dan Jumlah SKS yang diambil dari semester 1 hingga semester 6. Data ini digunakan untuk membangun model prediksi kelulusan [9].

Data Preparation

Tahap ini melibatkan proses *data preprocessing* yang meliputi pembersihan data (cleaning), penanganan nilai yang hilang, serta normalisasi data agar siap digunakan dalam proses pemodelan [10] yang dapat dilihat pada gambar 3. Sedangkan pada gambar 4 merupakan tahapan Proses transformasi merupakan tahapan selanjutnya setelah proses pembersihan data. Proses transformasi dari atribut data yang memiliki angka dilakukan untuk memudahkan penambangan data. Proses transformasi dari atribut data yang digunakan ditentukan dengan membuat kategori dari setiap data yang akan digunakan untuk mentransformasi data dengan menentukan setiap kategori pada data. Ips lebih besar dari 2.70 ditentukan sebagai kategori "Lulus" sedangkan IPS kurang dari 2.70 ditentukan sebagai kategori "Tidak Lulus".

Dataset:

	NIM	NAMA	IPS_1	SKS_1	IPS_2	SKS_2	\
0	2.010802e+11	MOKHAMAD DIKI ARMANDA	3.83	20.0	3.80	20.0	
1	2.010802e+11	MUHAMMAD ALAIKA ASYROF	3.78	20.0	3.30	20.0	
2	2.010802e+11	ARI TOPAN IQBAL MADAGASKAR	3.80	20.0	3.57	20.0	
3	2.010802e+11	FAHRI FAUZI HIDAYAT	3.72	20.0	3.78	18.0	
4	2.010802e+11	RENO FAIZAL FAHMI	2.53	20.0	3.13	18.0	

	IPS_3	SKS_3	IPS_4	SKS_4	IPS_5	SKS_5	IPS_6	SKS_6
0	3.84	20.0	3.60	21.0	3.80	23.0	3.91	21.0
1	3.35	23.0	3.55	21.0	2.67	23.0	3.60	15.0
2	3.26	23.0	3.62	21.0	3.55	23.0	3.72	18.0
3	3.47	23.0	3.65	18.0	3.60	23.0	3.70	23.0
4	3.47	20.0	3.52	20.0	3.60	23.0	3.76	21.0

Gambar 3. Contoh Hasil Pembersihan Data Akademik

Gambar diatas menunjukkan hasil pembersihan data akademik mahasiswa, di mana pada tahap ini data masih dalam bentuk awal setelah diekstraksi dari sumber. Data tersebut terdiri atas kolom NIM, Nama, nilai IPS dan jumlah SKS dari semester 1 hingga semester 5. Proses pembersihan data dilakukan untuk memastikan bahwa seluruh nilai memiliki format yang seragam, seperti penggunaan titik desimal pada data numerik, serta menghilangkan duplikasi dan nilai kosong yang dapat mengganggu proses analisis. Pada tahap ini, belum terdapat atribut target seperti status kelulusan mahasiswa.

	NIM	NAMA	IPS_1	SKS_1	IPS_2	SKS_2	\
0	2.010802e+11	MOKHAMAD DIKI ARMANDA	3.83	20.0	3.80	20.0	
1	2.010802e+11	MUHAMMAD ALAIKA ASYROF	3.78	20.0	3.30	20.0	
2	2.010802e+11	ARI TOPAN IQBAL MADAGASKAR	3.80	20.0	3.57	20.0	
3	2.010802e+11	FAHRI FAUZI HIDAYAT	3.72	20.0	3.78	18.0	
4	2.010802e+11	RENO FAIZAL FAHMI	2.53	20.0	3.13	18.0	

	IPS_3	SKS_3	IPS_4	SKS_4	IPS_5	SKS_5	IPS_6	SKS_6	Avg_IPS	Status
0	3.84	20.0	3.60	21.0	3.80	23.0	3.91	21.0	3.796667	Lulus
1	3.35	23.0	3.55	21.0	2.67	23.0	3.60	15.0	3.375000	Lulus
2	3.26	23.0	3.62	21.0	3.55	23.0	3.72	18.0	3.586667	Lulus
3	3.47	23.0	3.65	18.0	3.60	23.0	3.70	23.0	3.653333	Lulus
4	3.47	20.0	3.52	20.0	3.60	23.0	3.76	21.0	3.335000	Lulus

Gambar 4. Contoh Hasil Transformasi Data Akademik

Gambar 4 menampilkan hasil transformasi data akademik setelah dilakukan proses feature engineering. Perubahan utama yang terlihat adalah penambahan dua kolom baru, yaitu avg_IPS yang merupakan rata-rata dari nilai IPS semester 1 hingga 5, dan kolom Status yang menunjukkan label kelulusan mahasiswa (Lulus atau Tidak Lulus). Nilai rata-rata IPS dihitung untuk memberikan informasi agregat yang dapat digunakan sebagai fitur prediktif, sedangkan status kelulusan ditentukan berdasarkan kriteria tertentu, misalnya mahasiswa dinyatakan lulus jika rata-rata IPS memenuhi ambang batas yang ditentukan. Transformasi ini dilakukan agar data siap digunakan dalam proses pelatihan model machine learning, khususnya untuk keperluan klasifikasi.

Modeling

Penelitian ini menggunakan tiga algoritma Machine Learning untuk memprediksi kelulusan mahasiswa, yaitu *Naïve Bayes, Decision Tree, dan Random Forest.* Ketiga algoritma ini digunakan untuk membangun model prediksi berdasarkan data akademik yang telah diproses melalui tahap pembersihan dan normalisasi. *Naïve Bayes* adalah algoritma klasifikasi berbasis probabilistik yang bekerja dengan prinsip Teorema Bayes serta asumsi independensi antar fitur, dengan rumus umum:.

$$P(H|X) = \frac{P(X|H) \cdot P(H)}{P(X)}$$

di mana P(H|X) adalah probabilitas hipotesis H(misalnya, lulus) terhadap data X. Decision Tree adalah metode klasifikasi berbasis struktur pohon yang membagi data ke dalam simpul-simpul berdasarkan atribut dengan nilai *information gain* tertinggi, menggunakan rumus:

$$Information\ Gain = Entropy(S) - i=1 \sum n \frac{|Si|}{|S|} / \cdot Entropy(Si)$$

di mana Entropy(S) mengukur tingkat ketidakpastian dalam himpunan data S Adapun Random Forest merupakan algoritma ensemble learning yang membangun banyak pohon keputusan (decision tree) dan menggabungkan hasilnya melalui mekanisme voting untuk meningkatkan akurasi dan mengurangi risiko overfitting. Proses ini melibatkan teknik bootstrap sampling serta pemilihan fitur secara acak pada setiap pohon yang dibangun. Dengan keunggulan dan karakteristik masing-masing, ketiga algoritma ini dibandingkan untuk menentukan model terbaik dalam memprediksi kelulusan mahasiswa. Evaluasi dilakukan dengan dua skenario pembagian data, yaitu: 80% data training dan 20% data testing. Juga 60% data training dan 40% data testing. Hasil evaluasi model menunjukkan bahwa pembagian data yang berbeda mempengaruhi performa algoritma, namun Random Forest tetap menunjukkan performa yang superior.

Persentase data training dan data testing yang digunakan dapat dilihat pada Tabel 1.

Tabel 2.1. Kelompok Data Training dan Data Testing

Kelompok	Data Training	Jumlah Data	Data Testing	Jumlah Data
1	80%	480	20%	120
2	60%	360	40%	240

Evaluation

Setelah model dibangun, tahap ini akan mengukur kinerja model menggunakan *confusion matrix* untuk menghitung akurasi, *Precision*, r*ecall*, dan F1-score. Evaluasi ini bertujuan untuk menentukan algoritma mana yang memberikan performa terbaik dalam memprediksi kelulusan mahasiswa [11]. Adapun rumus perhitungan masing-masing metrik evaluasi adalah sebagai berikut:

- Perhitungan *Accuracy*
  Rumus Umum : $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
- Perhitungan *Precision*
  Rumus Umum : $Precision = \frac{TP}{TP+FP}$
- Perhitungan *Recall*
  Rumus Umum : $Recall = \frac{TP}{TP+FN}$
- Perhitungan F1-*Score*
  Rumus Umum: $F1_1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$

Deployment

Hasil dari penelitian ini diharapkan dapat diimplementasikan dalam sistem prediksi kelulusan mahasiswa di universitas, memberikan alat bagi pihak akademik untuk memantau dan meningkatkan tingkat kelulusan [12].

III. HASIL DAN PEMBAHASAN

Perbandingan performa ketiga algoritma Machine Learning dilakukan berdasarkan dua skenario pembagian data, yaitu 80% data training dan 20% data testing (skenario 1), serta 60% data training dan 40% data testing (skenario 2). Evaluasi dilakukan berdasarkan nilai akurasi yang dihasilkan oleh masing-masing algoritma dalam kedua skenario tersebut. Berdasarkan perancangan yang telah dijelaskan sebelumnya, maka hasil pengujian dibagi dalam 2 Skenario pembagian data Pada masing-masing algoritma diantaranya:

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

Proses Klasifikasi Naïve Bayes

model Naïve Bayes dibangun menggunakan bahasa pemrograman Python dengan library *scikit-learn*, khususnya fungsi Gaussian NB(). Data dibagi ke dalam dua skenario: 80% training dan 20% testing (Skenario 1), serta 60% training dan 40% testing (Skenario 2). Setelah proses training, data testing dimasukkan untuk menghasilkan prediksi. Hasil prediksi dibandingkan dengan label aktual untuk membentuk confusion matrix. Dari matrix tersebut, akurasi dihitung menggunakan rumus

Perhitungan dilakukan menggunakan fungsi accuracy_score() dari pustaka *scikit-learn*, yang menghitung rasio jumlah prediksi benar terhadap seluruh data. Nilai akurasi dari masing-masing skenario ditampilkan pada Tabel Pada algoritma Naïve Bayes, hasil akurasi yang diperoleh pada skenario pertama maupun skenario kedua adalah 86.25%. Tidak adanya perubahan nilai akurasi ini menunjukkan bahwa performa Naïve Bayes tidak terlalu sensitif terhadap variasi proporsi data training dan testing. Hal ini sejalan dengan karakteristik algoritma Naïve Bayes yang berbasis probabilistik dan cenderung menghasilkan performa yang stabil pada dataset berukuran sedang. Meskipun demikian, dibandingkan dua algoritma lainnya, akurasi Naïve Bayes masih berada pada posisi terendah. Hal ini mengindikasikan bahwa meskipun stabil, model ini kurang optimal dalam menangkap kompleksitas pola data akademik mahasiswa.

Tabel 3.1 Hasil Akurasi Model Naïve Bayes pada Skenario 1 dan 2

Skenario Pembagian Data	Akurasi
80% Training : 20% Testing	86.25%
60% Training : 40% Testing	86.25%

Tabel 3.2 Hasil Akurasi Model Naïve akurasi terbaik

Classification Report:

	precision	recall	f1-score	support
Lulus	1.00	0.84	0.91	70
Tidak Lulus	0.48	1.00	0.65	10
accuracy			0.86	80
macro avg	0.74	0.92	0.78	80
weighted avg	0.93	0.86	0.88	80

Akurasi NB : 86.25%

Confusion Matrix:

[[59 11]
[0 10]]

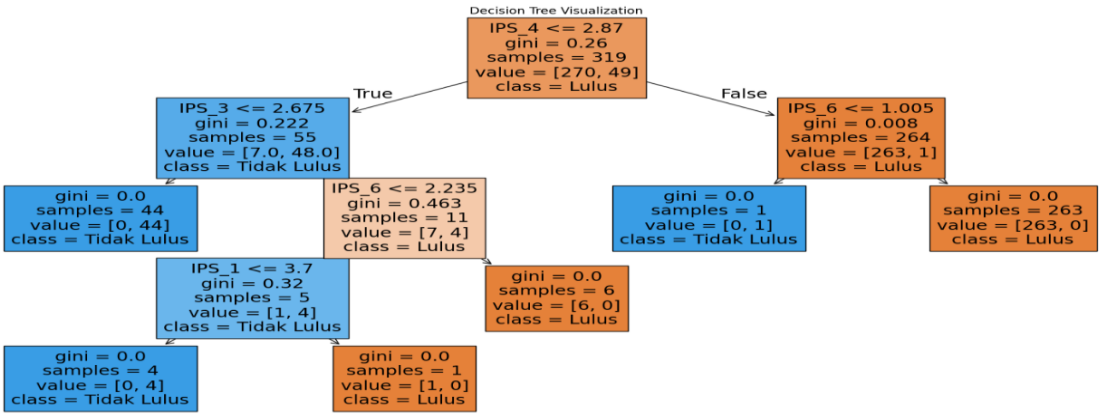
Pada algoritma Naïve Bayes, hasil akurasi yang diperoleh pada skenario pertama maupun skenario kedua adalah 86.25%. Tidak adanya perubahan nilai akurasi ini menunjukkan bahwa performa Naïve Bayes tidak terlalu sensitif terhadap variasi proporsi data training dan testing. Hal ini sejalan dengan karakteristik algoritma Naïve Bayes yang berbasis probabilistik dan cenderung menghasilkan performa yang stabil pada dataset berukuran sedang. Meskipun demikian, dibandingkan dua algoritma lainnya, akurasi Naïve Bayes masih berada pada posisi terendah. Hal ini mengindikasikan bahwa meskipun stabil, model ini kurang optimal dalam menangkap kompleksitas pola data akademik mahasiswa.

Proses Klasifikasi *Decision Tree*

model Naïve Bayes dibangun menggunakan bahasa pemrograman Python dengan library *scikit-learn*, khususnya fungsi Gaussian NB(). Data dibagi ke dalam dua skenario: 80% training dan 20% testing (Skenario 1), serta 60% training dan 40% testing (Skenario 2). Setelah proses training, data testing dimasukkan untuk menghasilkan prediksi. Hasil prediksi dibandingkan dengan label 7eputu untuk membentuk confusion matrix. Dari matrix tersebut, akurasi dihitung menggunakan rumus

Metode Elbow atau metode siku digunakan dalam pengelompokkan data karena metode elbow berperan dalam menentukan banyaknya cluster yang akan dipilih[10]. Metode elbow berbentuk lengkungan garis dengan sudut tajam yang menunjukkan cluster optimal[11.

Berikut merupakan pohon 7eputusan (Decision Tree) yang dibuat menggunakan algoritma Decision Tree Classifier beserta penjelasannya:



Gambar 5. Pohon Keputusan

Dalam pohon keputusan ini, algoritma hanya memilih fitur-fitur yang paling membantu memisahkan data dengan baik. Karena IPS_2 tidak memberikan pemisahan yang signifikan dibandingkan fitur lain seperti IPS_3, IPS_4, dan IPS_6, maka IPS_2 tidak digunakan dalam pembentukan pohon. Jadi, IPS_2 diabaikan karena atribut lain lebih efektif dalam membedakan mahasiswa yang lulus dan tidak lulus.Evaluasi model Decision Tree dilakukan dengan pendekatan serupa. Model dibangun menggunakan fungsi Decision Tree Classifier () dari pustaka *scikit-learn*. Data dibagi menjadi dua skenario: 80% training dan 20% testing, serta 60% training dan 40% testing Model dilatih menggunakan data training, lalu diuji pada data testing untuk menghasilkan prediksi. Confusion matrix kemudian dibentuk dari hasil prediksi, dan akurasi dihitung berdasarkan jumlah prediksi benar menggunakan rumus:

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

Proses evaluasi ini menghasilkan akurasi yang dihitung secara otomatis melalui fungsi accuracy_score(), dengan hasil yang ditampilkan pada Tabel 3.3.

Tabel 3.3 Hasil Akurasi Model *Decion Tree* pada Skenario 1 dan 2

Skenario Pembagian Data	**Akurasi**
80% Training : 20% Testing	96.25%
60% Training : 40% Testing	96.25%

Perhitungan *Precision*
Rumus Umum :

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

- *Precision* Kelas Lulus (Kelas 1):

TP = 69 (benar diprediksi lulus)
FP = 2 (tidak lulus tapi diprediksi lulus)

$\frac{69}{69+2} = \frac{69}{71} = 0.97$

- Precision Kelas Tidak Lulus (Kelas 0):

TP = 8 (benar diprediksi tidak lulus)
FP = 1 (lulus tapi diprediksi tidak lulus)

$P0 = \frac{8}{8+1} = \frac{8}{9} = 0.89$

Perhitungan *Recall*

Rumus Umum : $Recall = \frac{TP}{TP+FN}$

TP = 69
FN = 1 (lulus tapi salah diprediksi tidak lulus)

- *Recall* Kelas Lulus (Kelas 1):

$R1 = \frac{69}{69+1} = \frac{69}{70} = 0.99$

- *Recall* Kelas Tidak Lulus (Kelas 0):

TP = 8
FN=2 (Tidak lulus tapi diprediksi lulus)

$P0 = \frac{8}{8+2} = \frac{8}{10} = 0.80$

Perhitungan F1-*Score*

Rumus Umum: $F1_1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$

- F1 *Score* Kelas Lulus (Kelas 1):
  $F1_1 = \frac{2 \times 0.97 \times 0.99}{0.97 + 0.99} = \frac{1.9206}{1.96} = 0.98$

- F1 *Score* Kelas Tidak Lulus Kelas 0):
  $F1_0 = \frac{2 \times 0.89 \times 0.80}{0.89 + 0.80} = \frac{1.424}{1.69} = 0.84$

Tabel 3.4 Hasil Akurasi Model *Decision Tree* akurasi terbaik

Classification Report:

	precision	recall	f1-score	support
Lulus	0.97	0.99	0.98	70
Tidak Lulus	0.89	0.80	0.84	10
accuracy			0.96	80
macro avg	0.93	0.89	0.91	80
weighted avg	0.96	0.96	0.96	80

Akurasi NB: 96.25%

Confusion Matrix:
[[69 1]
[2 8]]

Pada algoritma Decision Tree, akurasi yang dihasilkan juga tetap sama pada kedua skenario, yaitu sebesar 96.25%. Konsistensi nilai akurasi ini menandakan bahwa Decision Tree mampu membentuk aturan klasifikasi yang solid dan efektif, meskipun proporsi data latih dikurangi. Ketahanan model terhadap perbedaan proporsi data menunjukkan bahwa struktur pohon keputusan yang terbentuk tidak mengalami overfitting maupun underfitting. Dengan demikian, Decision Tree menjadi algoritma yang kuat dalam menangani prediksi kelulusan pada variasi jumlah data pelatihan.

Proses Klasifikasi *Random Forest*

model Naïve Bayes dibangun menggunakan bahasa pemrograman Python dengan library *scikit-learn*, khususnya fungsi Gaussian NB(). Data dibagi ke dalam dua skenario: 80% training dan 20% testing (Skenario 1), serta 60% training dan 40% testing (Skenario 2). Setelah proses training, data testing dimasukkan untuk menghasilkan prediksi. Hasil prediksi dibandingkan dengan label 9eputu untuk membentuk confusion matrix. Dari matrix tersebut, akurasi dihitung menggunakan rumus

Berbeda halnya dengan algoritma Random Forest, di mana terjadi perbedaan akurasi antara kedua skenario. Pada skenario 80:20, Random Forest mencatat akurasi tertinggi yaitu 97.50%, sedangkan pada skenario 60:40, akurasinya sedikit menurun menjadi 96.88%. Meskipun terjadi penurunan, selisih akurasi ini relatif kecil dan tidak berdampak signifikan terhadap keandalan model. Penurunan ini dapat dijelaskan melalui karakteristik ensemble learning yang digunakan oleh Random Forest, di mana semakin banyak data pelatihan yang tersedia, semakin baik model dapat membangun variasi pohon keputusan untuk menghasilkan prediksi yang akurat. Dengan demikian, Random Forest tetap menjadi algoritma terbaik dalam penelitian ini berdasarkan hasil akurasi, baik pada skenario 1 maupun skenario 2. Model Random Forest dikembangkan menggunakan fungsi Random Forest Classifier dari pustaka *scikit-learn*. Proses dilakukan dalam dua skenario pembagian data: 80% training dan 20% testing, serta 60% training dan 40% testing.

Model dilatih dengan metode ensemble yang membangun sejumlah pohon keputusan dari data training yang diacak secara bootstrap. Setelah pelatihan, model diuji menggunakan data testing dan menghasilkan prediksi yang dibandingkan dengan label aktual untuk membentuk confusion matrix. Akurasi dihitung menggunakan rumus:

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

Perhitungan dilakukan secara otomatis melalui fungsi accuracy_score(). Hasil dari kedua skenario evaluasi disajikan pada Tabel 3.5.

Tabel 3.5 Hasil Akurasi Model *Random Forest* pada Skenario 1 dan 2

Skenario Pembagian Data	Akurasi
80% Training : 20% Testing	97.50%
60% Training : 40% Testing	96.88%

Tabel 3.6 Hasil Akurasi Model *Random Forest* akurasi terbaik

Classification Report:

	precision	recall	f1-score	support
Lulus	0.97	1.00	0.99	70
Tidak Lulus	1.00	0.80	0.89	10
accuracy			0.97	80
macro avg	0.99	0.90	0.94	80
weighted avg	0.98	0.97	0.97	80

Akurasi Random Forest: 97.50%

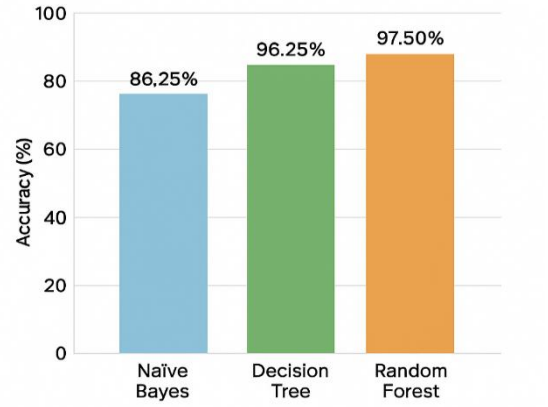
Confusion Matrix:
[[70 0]
[2 8]]

Analisis Hasil Evaluasi Akurasi Dari Tiga Algoritma

Berikut merupakan grafik dan tabel dari algoritma *naïve bayes, decision tree* dan *random forest* setelah dilakukan evaluasi pada setiap algoritma dan juga skenario :

Tabel 3.6 Hasil Akurasi tiga Model Algoritma dengan akurasi terbaik

Algoritma	Akurasi	Presisi	Recall	F1-Score
1. Naïve Bayes	86.25%	93.00%	86.00%	88.00%
2. Decision Tree	96.25%	96.00%	96.00%	96.00%
3. Random Forest	97.50%	98.00%	97.00%	97.00%



Gambar 6. Grafik Akurasi tiga Algoritma

Berdasarkan tabel evaluasi, Random Forest mencatat performa tertinggi di semua metrik, dengan akurasi 97.50%, presisi 98%, recall 97%, dan F1-score 97%, menjadikannya algoritma terbaik untuk prediksi kelulusan. Decision Tree berada di posisi kedua dengan nilai yang stabil pada semua metrik, yaitu 96.25% akurasi serta presisi, recall, dan F1-score masing-masing 96%. Ini menunjukkan bahwa model bekerja konsisten dan efektif. Sementara itu, Naïve Bayes memiliki akurasi paling rendah (86.25%), meskipun presisinya cukup tinggi (93%). Recall dan F1-score yang lebih rendah menunjukkan bahwa model kurang mampu menangkap keseluruhan pola kelulusan secara optimal.

IV. SIMPULAN

Hasil penelitian menunjukkan bahwa algoritma Random Forest memiliki performa terbaik dalam memprediksi kelulusan mahasiswa, dengan akurasi tertinggi sebesar 97.50% pada skenario 80:20 dan 96.88% pada skenario 60:40. Decision Tree juga menunjukkan performa yang konsisten dengan akurasi 96.25% di kedua skenario, menandakan model yang stabil dan andal. Sementara itu, Naïve Bayes menghasilkan akurasi stabil sebesar 86.25%, namun masih berada di bawah dua algoritma lainnya. Hal ini menunjukkan bahwa meskipun Naïve Bayes cukup stabil, algoritma ini kurang optimal dalam menangani data akademik yang lebih kompleks.Secara keseluruhan, Random Forest direkomendasikan sebagai algoritma terbaik untuk prediksi kelulusan mahasiswa dalam penelitian ini. Penelitian sebelumnya juga menunjukkan bahwa pendekatan ensemble learning yang menggunakan Random Forest mampu memberikan akurasi prediksi akademik yang tinggi dan nilai RMSE yang rendah, sehingga efektif digunakan dalam klasifikasi kelulusan mahasiswa [13] [14].

UCAPAN TERIMAKASIH

Penulis menyampaikan terima kasih yang sebesar-besarnya kepada Program Studi Informatika Universitas Muhammadiyah Sidoarjo atas kesempatan, fasilitas, dan dukungan yang telah diberikan selama pelaksanaan penelitian ini. Ucapan terima kasih juga disampaikan kepada pihak-pihak yang telah memberikan izin penelitian serta membantu dalam proses pengumpulan data, baik secara langsung maupun tidak langsung, sehingga penelitian ini dapat berjalan dengan lancar. Dukungan berupa akses terhadap data akademik, bantuan teknis, serta saran dalam pelaksanaan kegiatan penelitian sangat berarti bagi keberhasilan penyusunan karya ilmiah ini.Penulis juga menghargai kontribusi seluruh civitas akademika dan staf yang telah meluangkan waktu untuk membantu dalam proses klarifikasi data dan dokumentasi teknis yang dibutuhkan selama penelitian berlangsung. Semoga segala bantuan dan kerjasama yang telah diberikan menjadi amal kebaikan serta mendapatkan balasan yang setimpal dari Tuhan Yang Maha Esa.

REFERENSI

[1] U. Putra and I. Yptk, “Penerapan algoritma klasifikasi untuk prediksi tingkat kelulusan mahasiswa menggunakan rappidminer,” vol. 6, no. 1, pp. 376–388, 2025, doi: 10.46576/djtechno.

[2] D. Mardinah and M. Thoriq, “Algoritma Multi Layer Perceptron sebagai Prediksi Kelulusan Mahasiswa Universitas Adzkia Tepat Waktu berdasarkan jenis kelamin , Indeks Prestasi Semester , dan Jumlah SKS,” vol. 1, no. 2, pp. 26–35, 2024

[3] R. M. Ubaidilah, “Prediksi Kelulusan Mahasiswa Berdasarkan Data Kunjung dan Peminjaman Buku menggunakan Rapid Miner dengan Metode C.45 dan Random Forest,” Int. Res. Big-Data Comput. Technol. IRobot, vol. 7, no. 2, pp. 14–20, 2023, doi: 10.53514/ir.v7i2.410.

[4] A. Wahyudi, “Prediksi Kelulusan Mahasiswa Tepat Waktu Menggunakan Metode Decision Tree Dan Naïve Bayes,” J. Permata Indones., vol. 14, no. 2, pp. 132–138, 2023, doi: 10.59737/jpi.v14i2.276.

[5] E. Ismanto and M. Novalia, “Komparasi Kinerja Algoritma C4.5, Random Forest, dan Gradient Boosting untuk Klasifikasi Komoditas,” Techno.Com, vol. 20, no. 3, pp. 400–410, 2021, doi: 10.33633/tc.v20i3.4576.

[6] T. Wahyuningsih, D. Manongga, I. Sembiring, and S. Wijono, “Comparison of Effectiveness of Logistic Regression, Naive Bayes, and Random Forest Algorithms in Predicting Student Arguments,” Procedia Comput. Sci., vol. 234, pp. 349–356, 2024, doi: 10.1016/j.procs.2024.03.014.

[7] J. M. Polgan et al., “Optimasi Prediksi Kelulusan Mahasiswa Menggunakan Random Forest untuk Meningkatkan Tingkat Retensi,” vol. 13, pp. 2364–2374, 2025.

[8] M. A. Nurrohmat, “Aplikasi Pemrediksi Masa Studi dan Predikat Kelulusan Mahasiswa Informatika Universitas Muhammadiyah Surakarta Menggunakan Metode Naive Bayes,” Khazanah Inform. J. Ilmu Komput. dan Inform., vol. 1, no. 1, pp. 29–34, 2015, doi: 10.23917/khif.v1i1.1179.

[9] Oon Wira Yuda, Darmawan Tuti, Lim Sheih Yee, and Susanti, “Penerapan Penerapan Data Mining Untuk Klasifikasi Kelulusan Mahasiswa Tepat Waktu Menggunakan Metode Random Forest,” SATIN - Sains dan Teknol. Inf., vol. 8, no. 2, pp. 122–131, 2022, doi: 10.33372/stn.v8i2.885.

[10] A. Performance, E. Alshdaifat, D. Alshdaifat, A. Alsarhan, F. Hussein, and S. Moh, “The Effect of Preprocessing Techniques , Applied to Numeric,” Data, vol. 6, no. 11, 2021.

[11] H. Yuliansyah, R. A. P. Imaniati, A. Wirasto, and M. Wibowo, “Predicting Students Graduate on Time Using C4.5 Algorithm,” J. Inf. Syst. Eng. Bus. Intell., vol. 7, no. 1, p. 67, 2021, doi: 10.20473/jisebi.7.1.67-73.

[12] N. Sitohang, “Jurnal Sains Informatika Terapan (JSIT),” Penerapan Data Min. Untuk Peringatan Dini Banjir Menggunakan Metod. Klastering K-Means, vol. 2, no. 1, pp. 16–20, 2023

[13] R. Bakri, N. P. Astuti, and A. S. Ahmar, “Evaluating Random Forest Algorithm in Educational Data Mining: Optimizing Graduation on-time prediction using Imbalance Methods,” ARRUS J. Soc. Sci. Humanit., vol. 4, no. 1, pp. 108–116, 2024, doi: 10.35877/soshum2449.

[14] U. Indahyanti, N. L. Azizah, and H. Setiawan, “Pendekatan Ensemble Learning Untuk Meningkatkan Akurasi Prediksi Kinerja Akademik Mahasiswa,” J. Sains dan Inform., vol. 8, no. 2, pp. 160–169, 2022, doi: 10.34128/jsi.v8i2.459.

Conflict of Interest Statement:

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.