Jurnal Manajemen Teknologi dan Sistem Informasi (JMS)
Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

Perbandingan Algoritma *C4.5* Dan *Naïve Bayes* Dalam *Machine Learning* Untuk Klasifikasi Performa Pelajar

Muhammad Bilal Alfayyadh[1], Setiawan Assegaff[2], Fachruddin[3,*]

[1] Fakultas Ilmu Komputer, Magister Sistem Informasi, Universitas Dinamika Bangsa, Jambi, Indonesia
Email: [1]bilalalfayyadh@gmail.com, [2]setiawanassegaff@yahoo.com, [3,*]fachruddin.stikom@gmail.com
Email Penulis Korespondensi: bilalalfayyadh@gmail.com

Submitted :
16 April 2025

Revision :
24 Juni 2025

Accepted:
25 September 2025

Published:
30 September 2025

Abstrak– Perkembangan teknologi informasi dan komunikasi yang pesat telah membawa perubahan signifikan dalam berbagai bidang, termasuk pendidikan. Salah satu inovasi teknologi yang telah memberikan kontribusi besar adalah *machine learning*. Pendidikan merupakan aspek penting dalam pembangunan sumber daya manusia. Dalam konteks ini, memahami faktor-faktor yang mempengaruhi performa pelajar dan mengklasifikasi hasil belajar mereka menjadi sangat penting. Penelitian ini bertujuan untuk membandingkan kinerja terbaik dari dua model yaitu Algoritma *C4.5* dan *Naïve Bayes*, kemudian menghasilkan pohon keputusan untuk mempermudah klasifikasi performa pelajar. Dengan menggunakan *dataset* performa pelajar yang berjumlah 2.392 data, penelitian ini mengklasifikasi performa pelajar dari berbagai aspek seperti nilai, partisipasi dalam kelas, keterampilan belajar, serta kontribusi dalam kegiatan ekstrakurikuler. Pada penelitian ini, penulis melakukan *data splitting* dengan rasio sebesar 70:30 dan 80:20, kemudian melakukan evaluasi model dengan *confusion matrix* dan validasi model dengan 10-*fold cross-validation*. Hasil terbaik dari pengujian model adalah sebesar 85.82% menggunakan Algoritma *C4.5* dengan 10-*fold cross-validation*. Hasil penelitian ini diharapkan tidak hanya mampu mengklasifikasi performa pelajar dengan akurasi yang baik, tetapi juga memberikan *insight* yang berharga bagi pendidik dan pengelola sekolah maupun kampus untuk meningkatkan kualitas pendidikan secara keseluruhan.

Kata Kunci: Klasifikasi, *C4.5*, *Naïve Bayes*, *Confusion Matrix*, *Cross-Validation*

Abstract – The rapid development of information and communication technology has brought significant changes in various fields, including education. One of the technological innovations that has made a major contribution is machine learning. Education is an important aspect in human resource development. In this context, understanding the factors that influence student performance and classifying their learning outcomes is very important. This research aims to compare the best performance of two models, namely the C4.5 Algorithm and Naïve Bayes, then produce a decision tree to facilitate the classification of student performance. Using a student performance dataset totaling 2,392 data, this research classifies student performance from various aspects such as grades, class participation, learning skills, and contributions to extracurricular activities. In this research, the author performed data splitting with a ratio of 70:30 and 80:20, then evaluated the model with a confusion matrix and validated the model with 10-fold cross-validation. The best result of model testing was 85.82% using the C4.5 Algorithm with 10-fold cross-validation. The results of this research are expected to not only be able to classify student performance with good accuracy, but also provide valuable insights for educators and school and campus administrators to improve the overall quality of education.

Keywords: *Classification, C4.5, Naïve Bayes, Confusion Matrix, Cross-Validation*

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi yang pesat telah membawa perubahan signifikan dalam berbagai bidang, termasuk pendidikan. Salah satu inovasi teknologi yang telah memberikan kontribusi besar adalah *machine learning*. *Machine learning* adalah sistem yang mampu belajar sendiri untuk memutuskan sesuatu tanpa harus berulangkali diprogram oleh manusia [1]. Oleh karena itu, hal ini telah membuka peluang baru dalam menganalisis dan mengklasifikasi berbagai fenomena, termasuk performa pelajar.

Pendidikan merupakan aspek penting dalam pembangunan sumber daya manusia. Dalam konteks ini, memahami faktor-faktor yang mempengaruhi performa pelajar dan mengklasifikasi hasil belajar mereka menjadi sangat penting. Performa pelajar merupakan tingkat kemampuan seorang pelajar dalam aktivitas akademik dan non-akademik. Performa pelajar dapat mencakup berbagai aspek yang menggambarkan pencapaian akademik, keterampilan, dan perilaku belajar mereka. Adapun hal yang berkaitan dengan performa pelajar, seperti nilai, partisipasi dalam kelas, keterampilan belajar, serta kontribusi dalam kegiatan ekstrakurikuler.

Penelitian ini menggunakan Algoritma *C4.5* dan *Naïve Bayes* dalam mengklasifikasi performa pelajar berdasarkan data yang ada, dengan menggunakan data historis tentang pelajar, seperti nilai IPK/GPA, jumlah kehadiran, jumlah waktu belajar, ekstrakurikuler, dan faktor-faktor lainnya. Algoritma *C4.5* dan *Naïve Bayes* merupakan dua metode yang sering digunakan dalam proses klasifikasi karena keunggulannya masing-masing. Algoritma *C4.5* dikenal karena kemampuannya membangun pohon keputusan yang mudah dipahami dan memiliki performa yang baik dalam menangani data dengan atribut numerik maupun kategorikal. Sementara itu, *Naïve*

Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

Bayes merupakan algoritma berbasis probabilistik yang terkenal karena kesederhanaannya dan efisiensi dalam menangani *dataset* berukuran besar. Selain Algoritma *C4.5* dan *Naïve Bayes*, ada beberapa metode klasifikasi yang dapat digunakan seperti *Support Vector Machine* (SVM) dan *Random Forest*. Dengan demikian, penelitian ini membandingkan Algoritma *C4.5* dan *Naïve Bayes* agar dapat memberikan rekomendasi metode yang paling sesuai untuk mengklasifikasi performa pelajar.

Machine learning adalah cabang dari kecerdasan buatan (*artificial intelligence*) yang memungkinkan sistem untuk belajar dan membuat keputusan berdasarkan data tanpa diprogram secara eksplisit [1] [2]. *Machine learning* akan belajar dari data yang terdapat dalam sebuah *dataset* [3]. Konsep *machine learning* berfokus pada pengembangan model yang dapat mengenali pola dalam data dan melakukan klasifikasi atau membuat keputusan. Ada tiga kategori utama dalam *machine learning*, yaitu *supervised learning* (pembelajaran terawasi), *unsupervised learning* (pembelajaran tak terawasi) dan *reinforcement learning* (pembelajaran penguatan) [2] [4]. *Supervised learning* atau pembelajaran terawasi artinya pada pembelajaran ini ada acuan yang digunakan sebagai pengarah untuk mengarahkan sesuatu hal, misalnya melakukan klasifikasi [5]. Berdasarkan teknik pembelajarannya, *supervised learning* menggunakan *dataset* yang sudah berlabel [1]. Klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukannya kedalam kelas tertentu dari jumlah kelas yang tersedia. Klasifikasi dapat didefinisikan sebagai pekerjaan yang melakukan pelatihan/pembelajaran terhadap fungsi target yang memetakan setiap set atribut (fitur) ke satu jumlah label kelas yang tersedia [6]. Secara singkat, klasifikasi merupakan pengelompokan objek kedalam kelas tertentu berdasarkan kelompoknya yang biasanya disebut dengan kelas (*class*) [7]. *Python* adalah salah satu bahasa pemrograman tingkat tinggi yang mudah dipelajari karena sintaks yang jelas dan elegan, yang dikombinasikan dengan penggunaan modul-modul yang mempunyai struktur data tingkat tinggi, efisien, dan siap langsung digunakan [8].

Ada beberapa penelitian yang relevan dengan penelitian ini, diantaranya adalah Perbandingan Model Klasifikasi untuk Evaluasi Kinerja Akademik Mahasiswa oleh Rinna Rachmatika dan Achmad Bisri (2020) [9], Perbandingan Algoritma *Decision Tree* dan *Naive Bayes* dalam Klasifikasi Data Pengaruh Media Sosial dan Jam Tidur Terhadap Prestasi Akademik Siswa oleh Ifani Hariyanti, dkk. (2024) [10], Implementasi Algoritma *Naive Bayes* dan Algoritma *C4.5* dalam Klasifikasi Kelayakan Bantuan UMKM oleh Aldy Novriandy dan Winarsih (2023) [11], Perbandingan Algoritma *C4.5* dan *Naive Bayes* dalam Klasifikasi Tingkat Kepuasan Mahasiswa Terhadap Pembelajaran Daring oleh Fatmawati dan Narti (2022) [12], dan Klasifikasi Performa Akademik Siswa Menggunakan Metode *Decision Tree* dan *Naive Bayes* oleh Abdul Rahman (2023) [13].

Beberapa studi telah menunjukkan kontribusi yang signifikan pada *machine learning* dalam evaluasi performa pelajar. Sebagai contoh, *machine learning* dapat digunakan untuk personalisasi pembelajaran, identifikasi risiko akademik, evaluasi otomatis seperti *Natural Language Processing* (NLP), dan peningkatan efektivitas pengajaran.

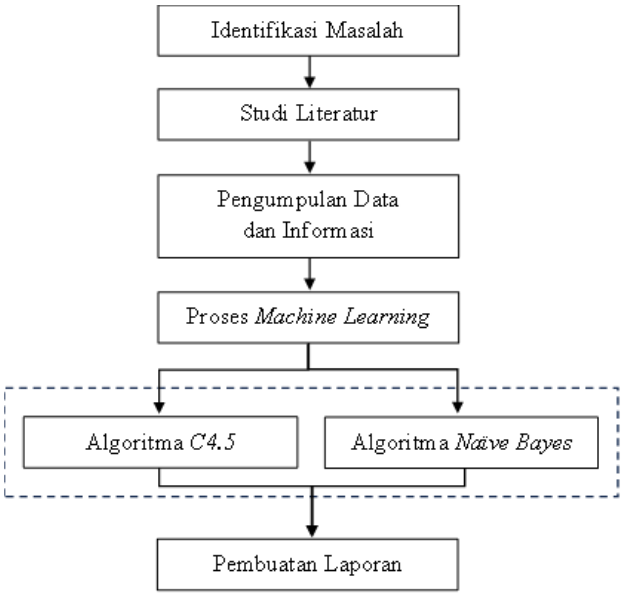
Penelitian ini bertujuan untuk mengklasifikasi data performa pelajar, mendapatkan perbandingan akurasi terbaik dari dua metode dalam mengklasifikasi performa pelajar, mendapatkan atribut yang berpengaruh sebagai faktor utama dalam mengklasifikasi performa pelajar, menghasilkan pohon keputusan yang mempermudah dalam mengklasifikasi performa pelajar. Dengan demikian, penerapan *machine learning* menggunakan Algoritma *C4.5* dan *Naïve Bayes* diharapkan tidak hanya mampu mengklasifikasi performa pelajar dengan akurasi yang baik, tetapi juga memberikan *insight* yang berharga bagi pendidik dan pengelola sekolah maupun kampus untuk meningkatkan kualitas pendidikan secara keseluruhan.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Untuk mempersiapkan penelitian ini diperlukan tahapan penelitian dalam bentuk sebuah kerangka kerja (*framework*) dengan langkah-langkah yang jelas. Tahapan penelitian ini terdapat aktivitas atau tugas yang harus dilakukan pada setiap langkah atau prosesnya. Tahapan penelitian yang digunakan adalah sebagai berikut :

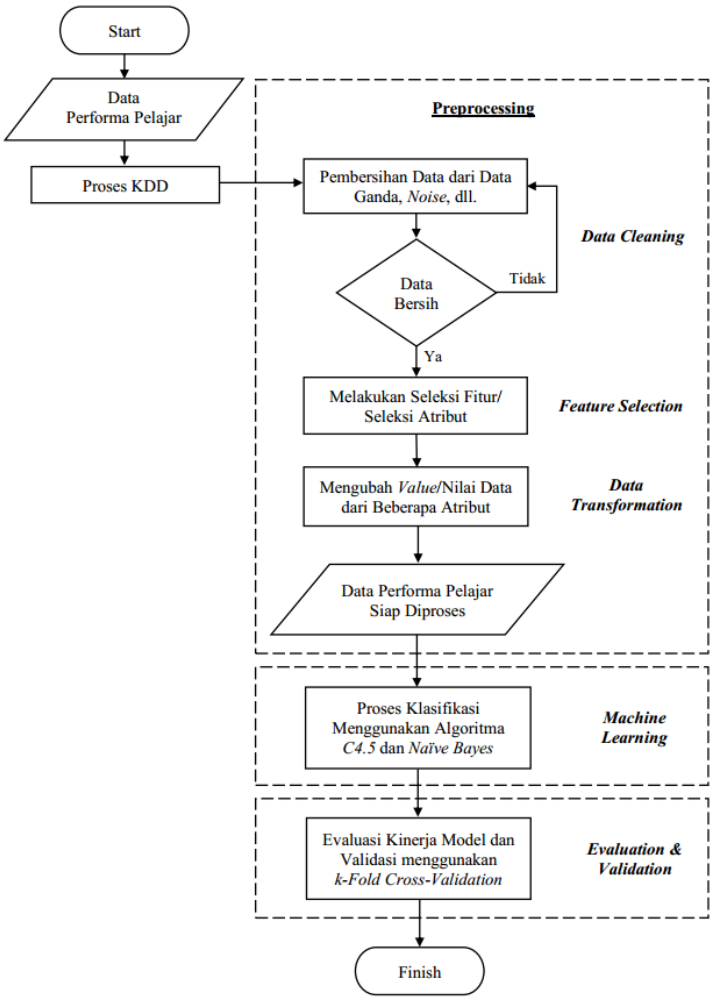
Jurnal Manajemen Teknologi dan Sistem Informasi (JMS)
Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2



Gambar 1. Tahapan Penelitian

2.2 Proses *Machine Learning*

Pada bagian ini, peneliti melakukan proses KDD (*Knowledge Discovery in Database*) yang bertujuan untuk mengekstrak pengetahuan yang berguna dari kumpulan data yang besar. Adapun proses KDD yang akan digunakan adalah sebagai berikut :



Gambar 2. Proses *Knowledge Discovery in Database*

Jurnal Manajemen Teknologi dan Sistem Informasi (JMS)
Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

2.3 Proses Algoritma *C4.5*

Pada bagian ini, data dan informasi yang diperoleh akan diproses menggunakan model Algoritma *C4.5* untuk mendapatkan hasil yang sesuai. Algoritma *C4.5* akan menghasilkan sebuah pohon keputusan (*Decision Tree*). Pohon keputusan merupakan hasil dari proses perhitungan *entropy* dan *gain* setelah perhitungan berulang-ulang hingga semua atribut pohon memiliki kelas dan tidak dapat lagi melakukan proses perhitungan [14].

2.4 Proses Algoritma *Naive Bayes*

Pada bagian ini data dan informasi yang diperoleh, kemudian diproses menggunakan model Algoritma *Naïve Bayes* untuk mendapatkan hasil yang sesuai. Algoritma *Naïve Bayes* akan melakukan perhitungan probabilitas berdasarkan fitur atau atribut yang ada didalam *dataset*. *Naive Bayes* merupakan sebuah metode klasifikasi yang berakar pada *teorema bayes*, metode klasifikasi ini dengan menggunakan metode probabilitas dan statistik [15].

2.5 *Dataset* Penelitian

Bahan penelitian yang digunakan adalah *dataset* dengan judul "Student Performance Data" yang diambil (diunduh) dari sebuah *website* yang bernama "*Kaggle*" di internet. *Dataset* ini memiliki jumlah 2.392 data dan 15 atribut. *Dataset* yang digunakan akan dipisahkan menjadi beberapa bagian atau biasa disebut dengan *data splitting*. Rasio atau bobot *data splitting* yang umum digunakan adalah 80:20, yang berarti 80% data untuk pelatihan (*training*) dan 20% data untuk pengujian (*testing*). Rasio atau bobot lain seperti 70:30, 60:40, dan bahkan 50:50 juga digunakan dalam praktik [16]. Pada penelitian ini, penulis menggunakan bobot atau rasio pembagian data sebesar 70:30 dan 80:20, yang dimana 70% dan 80% data untuk *training set*, 30% dan 20% data untuk *testing set*.

Tabel 1. *Student Performance Dataset*

StudentID	Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tutoring
1001	17	1	0	2	19,83372281	7	1
1002	18	0	0	1	15,40875606	0	0
1003	15	0	2	3	4,210569769	26	0
1004	17	1	0	3	10,02882947	14	0
1005	17	1	0	2	4,672495273	17	1
1006	18	0	0	1	8,191218545	0	0
1007	15	0	1	1	15,60168047	10	0
1008	15	1	1	4	15,42449631	22	1
….	….	….	….	….	….	….	….
3392	16	1	0	2	17,81990749	13	0

Tabel 2. *Student Performance Dataset*

ParentalSupport	Extracurricular	Sports	Music	Volunteering	GPA	GradeClass
2	0	0	1	0	2,929195592	2
1	0	0	0	0	3,042914833	1
2	0	0	0	0	0,112602254	4
3	1	0	0	0	2,05421814	3
3	0	0	0	0	1,288061182	4
1	1	0	0	0	3,084183614	1
3	0	1	0	0	2,748237415	2
1	1	0	0	0	1,360142712	4
….	….	….	….	….	….	….
2	0	0	0	1	2,140013878	1

3. HASIL DAN PEMBAHASAN

3.1 Analisis Data Performa Pelajar

3.1.1 Representasi Data

Pada *website* yang bernama *kaggle.com* (*https://www.kaggle.com/datasets/rabieelkharoua/students-performance-dataset*), penulis mengambil *dataset* performa pelajar yang dimana terdapat 2.392 data. Data ini terdapat 5 kategori *class*/label, yaitu 0 (GPA >= 3.5); 1 (3.0 <= GPA < 3.5); 2 (2.5 <= GPA < 3.0); 3 (2.0 <= GPA < 2.5); 4 (GPA < 2.0).

Jurnal Manajemen Teknologi dan Sistem Informasi (JMS)
Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

Dataset merupakan sekumpulan data yang terorganisir dalam format tertentu, biasanya berupa tabel yang terdiri dari baris dan kolom. *Dataset* digunakan sebagai sumber informasi utama dalam analisis, penelitian, atau pengembangan model. Adapun peran *dataset* dalam penelitian ini sebagai sumber informasi utama, dasar untuk analisis, serta evaluasi dan validasi model pembelajaran mesin (*machine learning*). *Dataset* dengan judul "*Student Performance Data*" dari *kaggle.com* dipilih karena sesuai dengan kebutuhan penelitian yaitu klasifikasi performa pelajar, *kaggle.com* juga merupakan tempat atau sumber *dataset* yang memiliki beragam pilihan *dataset*, dapat diakses secara gratis, memiliki kualitas yang baik, kemudahan integrasi, metadata dan dokumentasi yang jelas, serta sering digunakan untuk penelitian *machine learning*.

Tabel 3. *Dataset* Performa Pelajar

Class/Label	Ketentuan	Jumlah
0	(GPA >= 3.5)	107
1	(3.0 <= GPA < 3.5)	269
2	(2.5 <= GPA < 3.0)	391
3	(2.0 <= GPA < 2.5)	414
4	(GPA < 2.0)	1211
Total		2392

3.1.2 *Data Selection* dan *Preprocessing*

Pada tahap *Data Selection*, penulis melakukan pemilihan data yang akan digunakan, data tersebut merupakan *dataset* performa pelajar yang diperoleh dari *website Kaggle* (*https://www.kaggle.com/datasets/rabieelkharoua/students-performance-dataset*) dengan judul "Student Performance Data". Setelah tahap *Data Selection* selesai, selanjutnya masuk ke tahap *Preprocessing* yang dimana pada tahap ini terdapat proses *Data Cleaning*, *Feature Selection* dan *Data Transformation*. Berikut ini merupakan tahapan dari *Preprocessing* :

a. *Data Cleaning* (Pembersihan Data)
   Pada tahap ini, dilakukan proses pembersihan data yang bertujuan untuk meningkatkan kualitas data, mengurangi bias, dan memastikan hasil model yang lebih akurat. Proses ini sering kali membutuhkan waktu, tetapi sangat penting untuk keberhasilan penelitian dibidang *machine learning*. Setelah dilakukan proses pembersihan data, diketahui tidak ada data yang bersifat data kotor (*noise*), sehingga tidak ada perubahan data ataupun jumlah data pada tahap ini. Adapun contoh dari pembersihan data, seperti menghapus atau mengisi nilai kosong (*missing values*), menghapus data duplikat (data ganda), dan menghapus data yang terdapat simbol-simbol atau tanda baca (kesalahan pengetikan).
b. *Feature Selection* (Seleksi Fitur)
   Pada tahap ini, dilakukan proses seleksi fitur, yaitu memilih atribut yang berpengaruh terhadap *class*/*label* dari *dataset* dan membuang atribut yang tidak berpengaruh, sehingga data tersebut dapat digunakan dengan baik dan mendapatkan hasil yang optimal. Seleksi fitur ini dilakukan berdasarkan informasi yang terdapat pada sumber *dataset* di *kaggle.com*, terdapat informasi bahwa IPK (GPA : *Grade Point Average*) merupakan nilai rata-rata dalam skala 2.0 hingga 4.0, yang dipengaruhi oleh kebiasaan belajar (*study habits*), keterlibatan orang tua (*parental involvement*), dan kegiatan ekstrakurikuler (*extracurricular activities*). Setelah dilakukan seleksi fitur, terdapat 10 atribut/fitur yang berpengaruh seperti *StudyTimeWeekly*, *Absences*, *Tutoring*, *ParentalSupport*, *Extracurricular*, *Sports*, *Music*, *Volunteering*, *GPA*, *GradeClass*. Terdapat 5 atribut/fitur yang tidak berpengaruh diantaranya adalah *StudentID*, *Age*, *Gender*, *Ethnicity*, *ParentalEducation*.
c. *Data Transformation* (Transformasi Data)
   Pada tahap ini, dilakukan proses transformasi data yang dimana proses ini mengubah nilai/*value* dari *class*/*label* ("*GradeClass*") agar lebih sesuai, mudah dipahami, dan siap digunakan dalam pemodelan *machine learning*. Terdapat aturan didalam transformasi data, yaitu untuk meningkatkan kesesuaian dengan model atau algoritma, contohnya mengubah nilai kategoris seperti 'Male' dan 'Female' menjadi 0 dan 1. Adapun transformasi data yang dilakukan pada penelitian ini, yaitu mengubah nilai/*value* 0 menjadi "A", 1 menjadi "B", 2 menjadi "C", 3 menjadi "D", 4 menjadi "E".

Tabel 4. Transformasi Data

GradeClass (Before Transformation)	GradeClass (After Transformation)
2	C
1	B
4	E
3	D

Jurnal Manajemen Teknologi dan Sistem Informasi (JMS)
Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

4	E
1	B
2	C
4	E
2	C
0	A

3.1.3 *Training Data* dan *Testing Data*

Pada tahap ini, dilakukan proses pembagian data (*data splitting*) yang dimana akan membagi seluruh *dataset* menjadi dua bagian, yaitu data pelatihan (*training data*) dan data pengujian (*testing data*). Rasio pembagian data yang akan digunakan adalah 70 : 30 dan 80 : 20 dari total data. Berikut ini adalah tabel pembagian data (*data splitting*) dengan rasio yang sudah ditentukan :

Tabel 5. *Data Splitting* (*Training* dan *Testing*)

***Data Splitting* (*Ratio*)**			
Training 70%	Testing 30%	Training 80%	Testing 20%
1674	718	1914	478
Total Data : 2392		Total Data : 2392	

3.2 Implementasi Algoritma *C4.5* Dengan *Python*

a. Tahapan Implementasi Model Algoritma *C4.5*

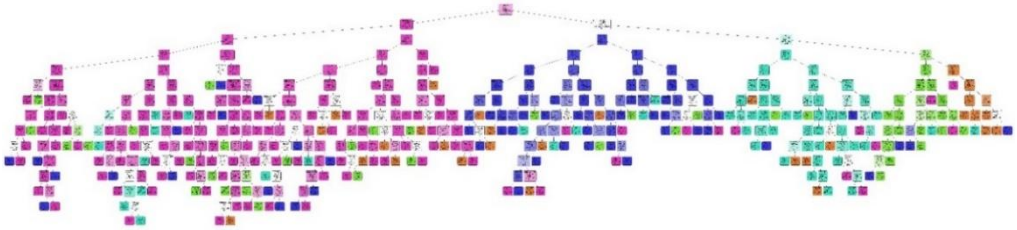
1. Persiapan Awal dan *Import Library*
   Adapun *library* yang digunakan seperti *pandas*, *numpy*, *matplotlib.pyplot*, *graphviz*, *sklearn.model_selection*, *sklearn.tree*, dan *sklearn.metrics* untuk membantu dalam pengolahan data, pemisahan *dataset*, pembuatan model Algoritma *C4.5*, serta evaluasi hasil klasifikasi.
2. *Import Dataset* dan Menampilkan Kolom *Dataset*
   Dataset yang digunakan diimpor menggunakan *pd.read_csv*() dan disimpan dalam variabel ”dataset”, kemudian menampilkan struktur data, seperti nama kolom, nilai atau isi data, dan tipe data. Setelah memuat *dataset*, penulis memeriksa kolom-kolom yang terdapat dalam *dataset* dengan menggunakan perintah *dataset.columns*.
3. Seleksi Fitur (Atribut) dan Pemilihan Target (*Class*)
   Atribut (*feature*) ini disimpan didalam variabel ”X”, sedangkan variabel ”y” akan berisi target (*class*). Dari 15 fitur yang ada, terdapat 5 fitur yang tidak berpengaruh terhadap target (*class*), yaitu *StudentID*, *Age*, *Gender*, *Ethnicity*, dan *ParentalEducation*. Untuk target (*class*) yang digunakan terdapat pada kolom *GradeClass*, yang berisi label dari masing-masing data, terdapat kategori nilai/*value* berupa 0, 1, 2, 3, 4.
4. Menampilkan Atribut (*Feature*) dan Target (*Class*)
   Proses ini penting untuk memastikan bahwa data yang digunakan sesuai dengan kebutuhan model klasifikasi. Kode perintah *X* atau *print(X)* untuk menampilkan atribut (*feature*) dan perintah *y* atau *print(y)* untuk menampilkan target (*class*).
5. *Data Transformation*
   Transformasi data untuk memastikan bahwa *dataset* berada dalam format yang sesuai untuk dianalisis dan membangun model Algoritma *C4.5*.
6. *Data Splitting* Untuk *Training Set* dan *Testing Set*
   Pembagian *dataset* menjadi dua bagian, yaitu *training set* dan *testing set*. *Training set* digunakan untuk melatih model dalam data, sedangkan *testing set* digunakan untuk menguji performa model terhadap data. Pembagian data ini menggunakan rasio 70:30 dan 80:20, yang dimana 70% dan 80% untuk *training set*, sedangkan 30% dan 20% untuk *testing set*.
7. Pembuatan dan Pelatihan Model Algoritma *C4.5* (*Decision Tree*)
   Pembuatan model *Decision Tree* (*C4.5*) menggunakan kelas *DecisionTreeClassifier* dari *sklearn.tree*. Setelah itu, model dilatih menggunakan *training data* (*X_train* dan *y_train*) dengan menggunakan metode *fit*().
8. Mengekspor Hasil Pohon Keputusan (*Decision Tree*)
   Hasil model pohon keputusan yang telah dibangun diekspor menggunakan *python* dan dapat disimpan dalam berbagai format, seperti gambar, file, atau model yang dapat digunakan kembali. Proses ekspor ini bertujuan untuk memudahkan visualisasi, dokumentasi, atau penerapan model pada sistem lain dimasa mendatang.
9. Evaluasi Model Algoritma *C4.5*
   Evaluasi model Algoritma *C4.5* dengan menggunakan 30% dan 20% dari *dataset* yang dipisahkan khusus sebagai data pengujian (*testing set*). Evaluasi ini bertujuan untuk menilai performa model pada data yang belum pernah digunakan selama pelatihan, sehingga dapat mengukur kemampuan model dalam

Jurnal Manajemen Teknologi dan Sistem Informasi (JMS)
Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

melakukan klasifikasi. Hasil evaluasi dinilai menggunakan *matrix* seperti *accuracy*, *precision*, *recall*, dan *F1-score*.

10. Validasi Model Algoritma *C4.5*
    Validasi dilakukan menggunakan *k-fold cross-validation* dengan 10 lipatan (*10-fold*) untuk menguji kinerja model Algoritma *C4.5*. Teknik ini membagi *dataset* menjadi 10 *subset* yang digunakan secara bergantian sebagai *data training* dan *data testing*. Proses ini bertujuan untuk memastikan hasil pengujian model lebih konsisten, mengurangi kemungkinan *overfitting*, dan memberikan gambaran yang lebih akurat tentang kemampuan generalisasi model.

b. Hasil Implementasi Model Algoritma *C4.5*
   1. Hasil Pohon Keputusan (*Decision Tree C4.5*)



Gambar 3. *Decision Tree* (*C4.5*)

   2. Hasil Evaluasi Model (*Confusion Matrix*)

```
Confusion Matriks :
[[ 18   1   5   3   6]
 [  2  59   4   9   6]
 [  6   4  98   5   8]
 [  3   1   6 107  10]
 [  5  18   6  12 316]]

Laporan Klasifikasi :
              precision    recall  f1-score   support

           A       0.53      0.55      0.54        33
           B       0.71      0.74      0.72        80
           C       0.82      0.81      0.82       121
           D       0.79      0.84      0.81       127
           E       0.91      0.89      0.90       357

    accuracy                           0.83       718
   macro avg       0.75      0.76      0.76       718
weighted avg       0.84      0.83      0.83       718

Akurasi = 83.28690807799443 %
```

Gambar 4. *Confusion Matrix* – 30% *Testing Set* (*C4.5*)

```
Confusion Matriks :
[[ 11   1   3   4   3]
 [  1  38   3   4   3]
 [  4   3  69   4   5]
 [  0   3   6  69   8]
 [  7   9   4   6 211]]

Laporan Klasifikasi :
              precision    recall  f1-score   support

           A       0.48      0.50      0.49        22
           B       0.70      0.78      0.74        49
           C       0.81      0.81      0.81        85
           D       0.79      0.80      0.80        86
           E       0.92      0.89      0.90       237

    accuracy                           0.83       479
   macro avg       0.74      0.76      0.75       479
weighted avg       0.83      0.83      0.83       479

Akurasi = 83.08977035490605 %
```

Gambar 5. *Confusion Matrix* – 20% *Testing Set* (*C4.5*)

   3. Hasil Validasi Model (10-*Fold Cross-Validation*)

```
Hasil Validasi 10-Fold :
85.82897489539748 %
```

Gambar 6. *Cross-Validation* (*C4.5*)

3.3 Implementasi Algoritma *Naïve Bayes* Dengan *Python*

a. Tahapan Implementasi Model Algoritma *Naive Bayes*
   1. Persiapan Awal dan *Import Library*

Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

Adapun *library* yang digunakan seperti *numpy*, *pandas*, *sklearn.model_selection*, *sklearn.naive_bayes*, dan *sklearn.metrics* untuk membantu dalam pengolahan data, pemisahan *dataset*, pembuatan model Algoritma *Naive Bayes*, serta evaluasi hasil klasifikasi.

2. *Import Dataset* dan Menampilkan Kolom *Dataset*
   Dataset yang digunakan diimpor menggunakan *pd.read_csv*() dan disimpan dalam variabel ”dataset”, kemudian menampilkan struktur data, seperti nama kolom, nilai atau isi data, dan tipe data. Setelah memuat *dataset*, penulis memeriksa kolom-kolom yang terdapat dalam *dataset* dengan menggunakan perintah *dataset.columns*.
3. Seleksi Fitur (Atribut) dan Pemilihan Target (*Class*)
   Atribut (*feature*) ini disimpan didalam variabel ”X”, sedangkan variabel ”y” akan berisi target (*class*). Dari 15 fitur yang ada, terdapat 5 fitur yang tidak berpengaruh terhadap target (*class*), yaitu *StudentID*, *Age*, *Gender*, *Ethnicity*, dan *ParentalEducation*. Untuk target (*class*) yang digunakan terdapat pada kolom *GradeClass*, yang berisi label dari masing-masing data, terdapat kategori nilai/*value* berupa 0, 1, 2, 3, 4.
4. Menampilkan Atribut (*Feature*) dan Target (*Class*)
   Proses ini penting untuk memastikan bahwa data yang digunakan sesuai dengan kebutuhan model klasifikasi. Kode perintah *X* atau *print(X)* untuk menampilkan atribut (*feature*) dan perintah *y* atau *print(y)* untuk menampilkan target (*class*).
5. *Data Transformation*
   Transformasi data untuk memastikan bahwa *dataset* berada dalam format yang sesuai untuk dianalisis dan membangun model Algoritma *Naive Bayes*.
6. *Data Splitting* Untuk *Training Set* dan *Testing Set*
   Pembagian *dataset* menjadi dua bagian, yaitu *training set* dan *testing set*. *Training set* digunakan untuk melatih model dalam data, sedangkan *testing set* digunakan untuk menguji performa model terhadap data. Pembagian data ini menggunakan rasio 70:30 dan 80:20, yang dimana 70% dan 80% untuk *training set*, sedangkan 30% dan 20% untuk *testing set*.
7. Pembuatan dan Pelatihan Model Algoritma *Naive Bayes*
   Pembuatan model Algoritma *Naive Bayes* menggunakan kelas *GaussianNB* dari *sklearn.naive_bayes*. Setelah itu, model dilatih menggunakan *training data* (*X_train* dan *y_train*) dengan menggunakan metode *fit*().
8. Evaluasi Model Algoritma *Naive Bayes*
   Evaluasi model Algoritma *Naive Bayes* dengan menggunakan 30% dan 20% dari *dataset* yang dipisahkan khusus sebagai data pengujian (*testing set*). Evaluasi ini bertujuan untuk menilai performa model pada data yang belum pernah digunakan selama pelatihan, sehingga dapat mengukur kemampuan model dalam melakukan klasifikasi. Hasil evaluasi dinilai menggunakan *matrix* seperti *accuracy*, *precision*, *recall*, dan *F1-score*.
9. Validasi Model Algoritma *Naive Bayes*
   Validasi dilakukan menggunakan *k-fold cross-validation* dengan 10 lipatan (*10-fold*) untuk menguji kinerja model Algoritma *Naive Bayes*. Teknik ini membagi *dataset* menjadi 10 *subset* yang digunakan secara bergantian sebagai *data training* dan *data testing*. Proses ini bertujuan untuk memastikan hasil pengujian model lebih konsisten, mengurangi kemungkinan *overfitting*, dan memberikan gambaran yang lebih akurat tentang kemampuan generalisasi model.

b. Hasil Implementasi Model Algoritma *Naive Bayes*
   1. Hasil Evaluasi Model (*Confusion Matrix*)

```
Confusion Matriks :
[[  2  19   4   4   4]
 [  0  46  27   2   5]
 [  0   5  98  13   5]
 [  0   2  26  90   9]
 [  0   2   0  32 323]]

Laporan Klasifikasi :
              precision    recall  f1-score   support

           A       1.00      0.06      0.11        33
           B       0.62      0.57      0.60        80
           C       0.63      0.81      0.71       121
           D       0.64      0.71      0.67       127
           E       0.93      0.90      0.92       357

    accuracy                           0.78       718
   macro avg       0.77      0.61      0.60       718
weighted avg       0.80      0.78      0.77       718

Akurasi = 77.85515320334262 %
```

Gambar 7. *Confusion Matrix* – 30% *Testing Set* (*Naïve Bayes*)

Jurnal Manajemen Teknologi dan Sistem Informasi (JMS)
Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

```
Confusion Matriks :
[[  2  13   3   2   2]
 [  0  30  14   1   4]
 [  0   7  63  11   4]
 [  0   2  16  61   7]
 [  0   1   0  24 212]]

Laporan Klasifikasi :
              precision    recall  f1-score   support

           A       1.00      0.09      0.17        22
           B       0.57      0.61      0.59        49
           C       0.66      0.74      0.70        85
           D       0.62      0.71      0.66        86
           E       0.93      0.89      0.91       237

    accuracy                           0.77       479
   macro avg       0.75      0.61      0.60       479
weighted avg       0.79      0.77      0.76       479

Akurasi = 76.82672233820459 %
```

Gambar 8. *Confusion Matrix* – 20% *Testing Set* (*Naïve Bayes*)

2. Hasil Validasi Model (10-*Fold Cross-Validation*)

```
Hasil Validasi 10-Fold :
78.3455369595537 %
```

Gambar 9. *Cross-Validation* (*Naïve Bayes*)

3.4 Hasil Evaluasi dan Validasi Model

Dalam evaluasi model klasifikasi, *Confusion Matrix* merupakan alat yang sering digunakan untuk memahami performa model secara lebih mendetail. Untuk mengamati tingkat kesalahan atau kebenaran sebuah model, *confusion matrix* adalah usulan alat yang sesuai. *Confusion matrix* memeriksa apakah jumlah prediksi untuk setiap kelas sudah sesuai [17]. *Confusion matrix* adalah tabel yang digunakan untuk melihat akurasi serta seberapa baik algoritma yang dihasilkan dari klasifikasi yang sudah dibuat, didalam *confusion matrix* terdapat *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) [18]. Dalam validasi model klasifikasi, *Cross Validation* merupakan teknik yang digunakan untuk mengevaluasi performa model dengan lebih akurat dan mengurangi risiko *overfitting*. *Cross validation* adalah sebuah metode dari teknik *machine learning* yang bertujuan untuk memperoleh hasil akurasi maksimum ketika data dibagi menjadi dua subset (data latih dan data uji). Salah satu dari jenis pengujian *Cross Validation* adalah *k-Fold Cross Validation* yang berfungsi untuk menilai kinerja proses sebuah metode algoritma dengan membagi sampel data secara acak dan mengelompokkan data tersebut sebanyak nilai *k* pada *k-Fold* [19]. Setiap subset digunakan sebagai data uji dan sisanya digunakan sebagai data latih. Setiap subset menerima nilai dan nilai total dihitung dari nilai rata-rata tiap subset [17]. Berikut ini merupakan hasil evaluasi dan validasi yang disajikan dalam bentuk tabel yang bertujuan untuk mengetahui dan membandingkan kinerja terbaik dari 2 model (Algoritma *C4.5* dan *Naïve Bayes*) dan 3 metode pengujian model (*Testing Set* 30%, *Testing Set* 20%, dan 10 *Fold-Cross Validation*) :

Tabel 6. Perbandingan Hasil Evaluasi dan Validasi

Pengujian Model		***Accuracy***	***Precision***	***Recall***
Algoritma *C4.5* (*Decision Tree*)	*Testing Set* 30%	83.28%	84.00%	83.00%
	Testing Set 20%	83.08%	83.00%	83.00%
	10 *Fold-Cross Validation*	85.82%	-	-
Algoritma *Naive Bayes*	*Testing Set* 30%	77.85%	80.00%	78.00%
	Testing Set 20%	76.82%	79.00%	77.00%
	10 *Fold-Cross Validation*	78.34%	-	-

4. KESIMPULAN

Penelitian ini menggunakan *online dataset* yang bersumber dari *website* yang bernama *kaggle* (*https://www.kaggle.com/datasets/rabieelkharoua/students-performance-dataset*) dengan jumlah 2.392 data, dan memiliki 15 atribut (*feature*) termasuk atribut target (*class*), yaitu *StudentID, Age, Gender, Ethnicity, ParentalEducation, StudyTimeWeekly, Absences, Tutoring, ParentalSupport, Extracurricular, Sports, Music, Volunteering, GPA, GradeClass*. Pada *dataset* performa pelajar yang digunakan pada penelitian ini terdapat 15 atribut, kemudian dilakukan proses seleksi fitur agar mendapatkan hasil yang baik dan optimal, sehingga atribut tersebut terseleksi menjadi 10 dari 15 atribut. Setelah dilakukan seleksi fitur, terdapat 10 atribut/fitur yang berpengaruh seperti *StudyTimeWeekly, Absences, Tutoring, ParentalSupport, Extracurricular, Sports, Music,*

Volume 5, Nomor 2, September 2025
ISSN 2808-5450 (media cetak), ISSN 2808-5019 (media online)
Available Online at https://ejournal.unama.ac.id/index.php/jms
DOI https://doi.org/10.33998/jms.v5i2

Volunteering, GPA, GradeClass. Terdapat 5 atribut/fitur yang tidak berpengaruh diantaranya adalah *StudentID, Age, Gender, Ethnicity, ParentalEducation*. Hasil evaluasi dan validasi untuk membandingkan kinerja terbaik yang dihasilkan dari 2 model *machine learning* dengan 3 metode pengujian model adalah Algoritma *C4.5* (*Testing Set* 30%) = 83.28%, Algoritma *C4.5* (*Testing Set* 20%) = 83.08%, Algoritma *C4.5* (10 *Fold-Cross Validation*) = 85.82%, Algoritma *Naïve Bayes* (*Testing Set* 30%) = 77.85%, Algoritma *Naïve Bayes* (*Testing Set* 20%) = 76.82%, Algoritma *Naïve Bayes* (10 *Fold-Cross Validation*) = 78.34%. Dapat disimpulkan bahwa model terbaik yang memperoleh hasil akurasi terbesar untuk klasifikasi performa pelajar adalah menggunakan Algoritma *C4.5* dengan 10 *Fold-Cross Validation* mendapatkan akurasi sebesar 85.82%.

REFERENCES

[1] E. Retnoningsih and R. Pramudita, "Mengenal Machine Learning Dengan Teknik Supervised Dan Unsupervised Learning Menggunakan Python," *Bina Insa. Ict J.*, vol. 7, no. 2, p. 156, 2020, doi: 10.51211/biict.v7i2.1422.

[2] Ibnu Daqiqil Id, *MACHINE_LEARNING_Teori_Studi_Kasus_dan_Implementasi*, 1st ed. Riau: UR PRESS, 2021.

[3] Fahrizal, F. O. Reynaldi, and N. Hikmah, "Implementasi Machine Learning pada Sistem PETS Identification Menggunakan Python Berbasis UBuntu," *J. Inf. Syst. Informatics Comput.*, vol. 4, no. 1, pp. 86–91, 2020, [Online]. Available: http://journal.stmikjayakarta.ac.id/index.php/jisicom/article/view/212

[4] A. U. Osarogiagbon, F. Khan, R. Venkatesan, and P. Gillard, "Review and analysis of supervised machine learning algorithms for hazardous events in drilling operations," *Process Saf. Environ. Prot.*, vol. 147, pp. 367–384, 2021, doi: 10.1016/j.psep.2020.09.038.

[5] A. P. Silalahi and H. G. Simanullang, "Supervised Learning Metode K-Nearest Neighbor Untuk Prediksi Diabetes Pada Wanita," *METHOMIKA J. Manaj. Inform. dan Komputerisasi Akunt.*, vol. 7, no. 1, pp. 144–149, 2023, doi: 10.46880/jmika.vol7no1.pp144-149.

[6] D. P. Utomo and M. Mesran, "Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung," *J. Media Inform. Budidarma*, vol. 4, no. 2, p. 437, 2020, doi: 10.30865/mib.v4i2.2080.

[7] H. F. Putro, R. T. Vulandari, and W. L. Y. Saptomo, "Penerapan Metode Naive Bayes Untuk Klasifikasi Pelanggan," *J. Teknol. Inf. dan Komun.*, vol. 8, no. 2, 2020, doi: 10.30646/tikomsin.v8i2.500.

[8] S. Ratna, "Pengolahan Citra Digital Dan Histogram Dengan Phyton Dan Text Editor Phycharm," *Technol. J. Ilm.*, vol. 11, no. 3, p. 181, 2020, doi: 10.31602/tji.v11i3.3294.

[9] R. Rachmatika and A. Bisri, "Perbandingan Model Klasifikasi untuk Evaluasi Kinerja Akademik Mahasiswa," *J. Edukasi dan Penelit. Inform.*, vol. 6, no. 3, p. 417, 2020, doi: 10.26418/jp.v6i3.43097.

[10] A.- Husaini, I. Hariyanti, and A. R. Raharja, "Perbandingan Algoritma Decision Tree dan Naive Bayes dalam Klasifikasi Data Pengaruh Media Sosial dan Jam Tidur Terhadap Prestasi Akademik Siswa," *Technol. J. Ilm.*, vol. 15, no. 2, p. 332, 2024, doi: 10.31602/tji.v15i2.14381.

[11] A. Novriandy, "Implementasi Algoritma Naive Bayes dan Algoritma C4. 5 dalam Klasifikasi Kelayakan Bantuan UMKM," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 1, pp. 208–217, 2023, doi: 10.30865/klik.v4i1.1099.

[12] F. Fatmawati and N. Narti, "Perbandingan Algoritma C4.5 dan Naive Bayes Dalam Klasifikasi Tingkat Kepuasan Mahasiswa Terhadap Pembelajaran Daring," *JTIM J. Teknol. Inf. dan Multimed.*, vol. 4, no. 1, pp. 1–12, 2022, doi: 10.35746/jtim.v4i1.196.

[13] A. Rahman, "Klasifikasi Performa Akademik Siswa Menggunakan Metode Decision Tree dan Naive Bayes," *J. SAINTEKOM*, vol. 13, no. 1, pp. 22–31, 2023, doi: 10.33020/saintekom.v13i1.349.

[14] H. Sulistiani and A. A. Aldino, "Decision Tree C4.5 Algorithm for Tuition Aid Grant Program Classification (Case Study: Department of Information System, Universitas Teknokrat Indonesia)," *Edutic - Sci. J. Informatics Educ.*, vol. 7, no. 1, pp. 40–50, 2020, doi: 10.21107/edutic.v7i1.8849.

[15] Alvina Felicia Watratan, Arwini Puspita. B, and Dikwan Moeis, "Implementasi Algoritma Naive Bayes Untuk Memprediksi Tingkat Penyebaran Covid-19 Di Indonesia," *J. Appl. Comput. Sci. Technol.*, vol. 1, no. 1, pp. 7–14, 2020, doi: 10.52158/jacost.v1i1.9.

[16] V. R. Joseph, "Optimal ratio for data splitting," *Stat. Anal. Data Min.*, vol. 15, no. 4, pp. 531–538, 2022, doi: 10.1002/sam.11583.

[17] D. K. Lailil Muflikhah, Wayan Firdaus Mahmudy, *Machine_Learning*. UB Press, 2023.

[18] Suci Amaliah, M. Nusrang, and A. Aswi, "Penerapan Metode Random Forest Untuk Klasifikasi Varian Minuman Kopi di Kedai Kopi Konijiwa Bantaeng," *VARIANSI J. Stat. Its Appl. Teach. Res.*, vol. 4, no. 3, pp. 121–127, 2022, doi: 10.35580/variansiunm31.

[19] R. R. R. Arisandi, B. Warsito, and A. R. Hakim, "Aplikasi Naïve Bayes Classifier (Nbc) Pada Klasifikasi Status Gizi Balita Stunting Dengan Pengujian K-Fold Cross Validation," *J. Gaussian*, vol. 11, no. 1, pp. 130–139, 2022, doi: 10.14710/j.gauss.v11i1.33991.