JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 13 No. 3, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v13i3.6766

PREDIKSI PENERIMAAN MAHASISWA BARU FAKULTAS ILMU KOMPUTER UNSIKA MENGGUNAKAN ALGORITMA C4.5 DAN CART

Dwi Miftahussalamah[1], Jajam haerul Jaman[2], Iqbal Maulana[3]

[1,2,3]Informatika, Universitas Singaperbangsa Karawang; Jl. HS.Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat

Keywords:
Algoritma C4.5;
Algoritma CART;
Decsision Tree;
Pruning.

Corespondent Email:
2110631170010@student.unsika.ac.id

Abstrak. Penerimaan mahasiswa baru merupakan aspek penting dalam pengelolaan pendidikan tinggi, termasuk di Universitas Singaperbangsa Karawang (Unsika) melalui jalur Seleksi Nasional Berdasarkan Prestasi (SNBP). Jalur ini menilai prestasi akademik siswa tanpa melalui tes tulis, namun masih menghadirkan tantangan dalam menentukan indikator keberhasilan seleksi. Di Fakultas Ilmu Komputer, tingkat persaingan sangat tinggi sehingga diperlukan pendekatan berbasis data untuk mendukung proses seleksi yang lebih objektif. Penelitian ini menerapkan algoritma *decision tree* C4.5 dan CART untuk memprediksi penerimaan calon mahasiswa baru berdasarkan data historis, dengan menerapkan metode SMOTE untuk mengatasi ketidakseimbangan data dan *cost complexity pruning* untuk meningkatkan generalisasi model. Proses data mining mengikuti tahapan *Knowledge Discovery in Databases* (KDD), dengan pengujian menggunakan pembagian data 80:20. Hasil evaluasi menunjukkan bahwa model C4.5 yang menerapkan SMOTE dan *pruning* memberikan performa terbaik, dengan akurasi sebesar 85% dan nilai ROC 0,92, mengungguli CART yang memiliki akurasi sama namun nilai ROC lebih rendah sebesar 0,89. Temuan ini menunjukkan bahwa algoritma C4.5 lebih unggul dalam mendukung proses prediksi penerimaan mahasiswa baru di lingkungan perguruan tinggi.

Abstract. *Admitting new students is an important aspect of managing higher education, including at Singaperbangsa University of Karawang (Unsika), which uses the National Selection Based on Achievement (SNBP) pathway. Although this pathway evaluates students' academic achievement without a written test, it still presents challenges in determining indicators of selection success. At the Faculty of Computer Science, competition is so fierce that a data-driven approach is necessary to ensure a fair selection process. This study uses C4.5 and CART decision tree algorithms to predict new student admissions based on historical data. The SMOTE method is applied to overcome data imbalance, and cost complexity pruning improves model generalization. The data mining process follows the stages of knowledge discovery in databases (KDD), and testing is performed using an 80:20 data split. Evaluation results showed that the C4.5 model with SMOTE and pruning performed best, achieving 85% accuracy and an ROC value of 0.92. This outperformed the CART model, which had the same accuracy but a lower ROC value of 0.89. These results demonstrate that the C4.5 algorithm is superior for predicting new student admissions in a university setting.*

1. PENDAHULUAN

Penerimaan mahasiswa baru merupakan aspek penting dalam pengelolaan pendidikan tinggi. Universitas Singaperbangsa Karawang (Unsika), sebagai perguruan tinggi negeri, mengikuti sistem Seleksi Nasional Penerimaan Mahasiswa Baru (SNPMB) yang dikelola oleh Balai Pengelolaan Pengujian Pendidikan (BPPP) di bawah naungan Kementerian Pendidikan Dasar dan Menengah (Kemendikdasmen). Sistem penerimaan calon mahasiswa baru ini mencakup tiga jalur utama, yaitu Seleksi Nasional Berbasis Prestasi (SNBP) dan Seleksi Nasional Berbasis Tes (SNBT) yang dikelola oleh Tim SNPMB serta Seleksi Mandiri yang sepenuhnya diatur oleh masing-masing perguruan tinggi [1].

SNBP merupakan jalur penerimaan berbasis prestasi akademik selama pendidikan menengah yang menjadi salah satu jalur tanpa tes tertulis. Namun, seleksi ini menimbulkan tantangan tersendiri dalam menentukan indikator keberhasilan yang relevan, khususnya pada program studi dengan persaingan tinggi [2].

Di era digital, industri semakin menuntut lulusan yang kompeten dalam teknologi, pengolahan data, dan pengembangan perangkat lunak. Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang perlu memastikan bahwa mahasiswa yang diterima memiliki potensi akademik yang sesuai dengan kebutuhan tersebut. Namun sayangnya, jumlah pendaftar pada jalur SNBP, jauh melebihi daya tampung yang tersedia, mencerminkan tingkat persaingan yang ketat dan perlunya proses seleksi yang efektif. Untuk menjawab tantangan ini, pendekatan berbasis data menjadi penting dalam memprediksi peluang diterimanya calon mahasiswa. Teknik data *mining* dapat digunakan untuk menggali pola dari data historis dan mengidentifikasi faktor yang memengaruhi keberhasilan seleksi. Dalam proses pengambilan keputusan berbasis data, pemilihan algoritma yang sesuai menjadi faktor penting untuk memperoleh hasil prediksi yang akurat. *Decision tree* merupakan salah satu metode yang banyak digunakan karena interpretasinya yang sederhana dengan algoritma populer seperti C4.5 dan CART [3].

Penelitian ini menggunakan algoritma C4.5 dan CART untuk memprediksi penerimaan mahasiswa baru melalui jalur SNBP di Fakultas Ilmu Komputer Unsika. Kedua algoritma tersebut digunakan dengan beberapa alasan. C4.5 unggul dalam mengolah data numerik dan kategorikal serta menghasilkan aturan yang mudah dipahami, sementara CART efektif dalam memilih variabel paling signifikan [4]. Untuk mengatasi ketidakseimbangan data, digunakan metode SMOTE, dan *cost complexity pruning* guna menyederhanakan model tanpa mengurangi akurasi. Tujuan dari penelitian ini adalah mengidentifikasi pola dan faktor utama dalam penerimaan mahasiswa, sebagai dasar pengambilan keputusan yang lebih tepat dan strategis.

2. TINJAUAN PUSTAKA

2.1. Decision tree

Decision tree adalah algoritma pembelajaran terawasi non-parametrik yang digunakan dalam tugas klasifikasi dan regresi. Algoritma ini merepresentasikan proses pengambilan keputusan dalam bentuk struktur pohon, yang terdiri dari *root node* sebagai titik awal, internal *node* yang berisi pengujian atribut, dan *leaf node* yang merepresentasikan hasil klasifikasi [5]. Setiap cabang dari *node* menggambarkan hasil dari pengujian pada atribut tertentu, dan proses pemisahan data dilakukan secara rekursif hingga terbentuk kelompok data yang homogen. Proses *decision tree* dilakukan dari atas ke bawah dengan membagi dataset ke dalam subset yang lebih kecil, hingga data dapat diklasifikasikan ke dalam kelas tertentu secara akurat [6].

2.2. Algoritma C4.5

Algoritma C4.5 merupakan metode pohon keputusan yang sering digunakan dalam klasifikasi data. Algoritma ini mampu menangani atribut numerik maupun kategorikal, serta membentuk model berdasarkan hubungan antara atribut dan kelas data [7]. Proses pembentukan pohon dilakukan secara rekursif dengan memilih atribut yang paling efektif memisahkan data, menggunakan nilai *information gain*, yaitu selisih *entropy* yang dinormalisasi untuk menghindari bias terhadap atribut dengan banyak nilai [8]. Konsep berikut akan menjelaskan bagaimana tahapan algoritma C4.5 dapat diterapkan.

a. Perhitungan *Entropy*
Entropy merupakan ukuran ketidakpastian atau keragaman data. Nilai *entropy* tinggi menunjukkan data tersebar merata di banyak kelas, sedangkan nilai rendah menunjukkan dominasi oleh satu atau beberapa kelas saja [9]. Untuk menghitung *entropy* digunakan rumus berikut.

$$Entropy(S) = \sum_{i=1}^{n} -p_i \log_2(p_i) \quad (1)$$

Keterangan:
S = Jumlah Sampel
i = Jumlah Populasi
n = Jumlah Partisi S
p_i =Probabilitas yang didapat dari Sum (Ya) atau Sum (Tidak) dibagi total kasus

b. Perhitungan *Information gain*
Information gain digunakan untuk mengukur seberapa besar pengurangan *entropy* setelah data dipisahkan berdasarkan suatu atribut. Atribut dengan nilai *information gain* tertinggi dipilih sebagai *root* karena paling efektif dalam menghasilkan kelompok data yang homogen [10]. Untuk menghitung *information gain*, digunakan rumus berikut.

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} Entropy(S_i) \quad (2)$$

Keterangan
S = Jumlah Sampel
A = Atribut
n = Jumlah Atribut Partisi A
$|S_i|$ = Jumlah Kasus pada Partisi ke *i*
$|S|$ = Jumlah Kasus dalam S

c. Pembentukan Pohon Klasifikasi
Setelah atribut dengan *information gain* tertinggi dipilih sebagai *root*, data dibagi ke dalam cabang berdasarkan nilai atribut tersebut. Proses perhitungan *information gain* dan pembagian data kemudian diulang secara rekursif pada setiap cabang hingga semua data terklasifikasi atau tidak ada lagi atribut yang bisa digunakan.

2.3. Algoritma CART

CART (*Classification and Regression Tree*) merupakan salah satu teknik klasifikasi yang membentuk model dalam bentuk pohon keputusan. CART mencakup dua pendekatan, yakni pohon klasifikasi dan pohon regresi. Apabila variabel target bersifat kategorikal, maka metode ini akan membentuk pohon klasifikasi, sedangkan jika variabel target bersifat kontinu atau numerik, maka akan dihasilkan pohon regresi [11]. Konsep berikut akan menjelaskan bagaimana tahapan algoritma CART dapat diterapkan.

a. Penetuan Pemilah
Penentuan pemilah menggunakan *Gini index*, yaitu metode untuk mengukur ketidakmurnian data dalam algoritma pohon keputusan. Atribut dengan nilai *gini index* terendah dipilih karena menghasilkan pemisahan data yang paling murni [12]. Rumus untuk perhitungan *gini index* adalah sebagai berikut.

$$i(t) = \sum_{i,j=1} p(j|t)\, p(i|t), i \neq j \quad (3)$$

Keterangan:
$p(j|t)$ = proporsi kelas j pada simpul t
$p(i|t)$ = proporsi kelas i pada simpul t

Pemilahan membentuk simpul yang akan terus dibagi secara rekursif hingga mencapai terminal *node*s. Evaluasi pemilah dilakukan menggunakan *goodness of split*, yaitu selisih nilai *gini index* sebelum dan sesudah pemisahan. Pemilah terbaik adalah yang menghasilkan nilai GOS tertinggi karena paling efektif menurunkan heterogenitas data. Berikut adalah persamaannya.

$$(\phi(s,t)) = \Delta i(s,t) = i(t) - P_L i(t_L)\, P_R i(t_R) \quad (4)$$

Keterangan :
$i(t)$= nilai *gini index* pada simpul t
$i(t_L)$ = nilai *gini index* pada simpul anak kiri
$i(t_R)$= nilai *gini index* pada simpul anak kanan
P_L = probabilitas amatan pada simpul kiri
P_R= probabilitas amatan pada simpul kanan

b. Penentuan Simpul Terminal
Simpul terminal ditentukan saat jumlah data di simpul kurang dari batas minimum (N_{min}) atau telah mencapai kedalaman maksimum pohon.

c. Penanda Kelas Label
Label kelas diberikan berdasarkan kelas yang memiliki amatan lebih banyak pada simpul terminal, yaitu apabila

$$p(j_0|t) = max_j p(j|t) = max_j \frac{N_j(t)}{N(t)} \quad (5)$$

Dimana:
$p(j|t)$ = proporsi kelas *j* pada simpul *t*
$N_j(t)$ = jumlah amatan kelas *j* pada *terminal node t*
$N(t)$ = jumlah total amatan pada *terminal node t*

2.4. *Pruning*

Pruning adalah proses penyederhanaan pohon keputusan dengan memangkas cabang yang tidak penting untuk mengurangi kompleksitas dan risiko *overfitting*. *Pruning* dilakukan agar model tetap akurat dalam memprediksi data baru. Terdapat dua pendekatan, yaitu *pre-pruning* yang menghentikan pembentukan pohon lebih awal, dan *post-pruning*, yang memangkas pohon setelah terbentuk penuh. Salah satu metode post-*pruning* yang umum digunakan adalah *Cost complexity Pruning* (CCP), yang mempertahankan subpohon dengan biaya kesalahan terendah [13].

2.5. SMOTE

Synthetic Minority Over-sampling Technique (SMOTE) adalah metode untuk menangani ketidakseimbangan kelas dengan menghasilkan sampel sintetis pada kelas minoritas. Teknik ini bekerja dengan mencari tetangga terdekat dari setiap sampel minoritas, lalu membentuk data baru melalui interpolasi. SMOTE membantu meningkatkan akurasi model dan mengurangi bias terhadap kelas mayoritas [14].

2.6. Evaluasi

Evaluasi kinerja model klasifikasi dapat dilakukan menggunakan *confusion matrix* dan ROC-AUC. *Confusion matrix* adalah tabel yang menunjukkan perbandingan antara hasil prediksi model dan nilai aktual dari data, dengan empat komponen utama, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN). Tabel 1 berikut menunjukan hasil klasifikasi *confusion matrix*.

Tabel 1. Confusion matrix.

		Kelas Prediksi	
		1	0
Kelas sebenarnya	1	TP	FN
	0	FP	TN

Dari matriks ini dapat dihitung metrik evaluasi seperti *accuracy, precision, recall* dan *F1-score*. Metrik tersebut menggambarkan tingkat ketepatan dan kemampuan model dalam mengidentifikasi kelas positif dan negatif [15]. Berikut ini merupakan rumus untuk menghitung *accuracy*, *precision*, *recall* dan *f1-score*.

$$accuracy = \frac{TP+TN}{Total} \quad (6)$$

$$precision = \frac{TP}{TP+FP} \quad (7)$$

$$recall = \frac{TP}{TP+FN} \quad (8)$$

$$F1\ score = 2 \text{ x } \frac{precission \text{ x } recall}{precission+recall} \quad (9)$$

Sementara itu, ROC (*Receiver Operating Characteristic)* adalah kurva yang memvisualisasikan kinerja model pada berbagai ambang batas, dengan memplot *True Positive Rate* (TPR) terhadap *False Positive Rate* (FPR). AUC (*Area Under Curve*) mengukur luas area di bawah kurva ROC dan mencerminkan kemampuan model dalam membedakan kelas. Semakin tinggi nilai AUC (mendekati 1), semakin baik performa model, terutama dalam kasus distribusi kelas yang tidak seimbang [16].

2.7. *Knowledge Discovery in Databases*

Knowledge Discovery in Database (KDD) adalah proses sistematis untuk menemukan informasi baru yang valid, berguna, dan dapat dipahami dari kumpulan data besar. Tujuan utamanya adalah memprediksi nilai penting dari variabel atau mengenali pola tersembunyi yang bermakna. KDD terdiri dari beberapa tahap, diantaranya yaitu *data selection, preprocessing, transformation, data mining* dan *evaluation*. Teknik data *mining* digunakan sebagai inti dari proses ini untuk mengungkap pola tersembunyi dalam data [17].

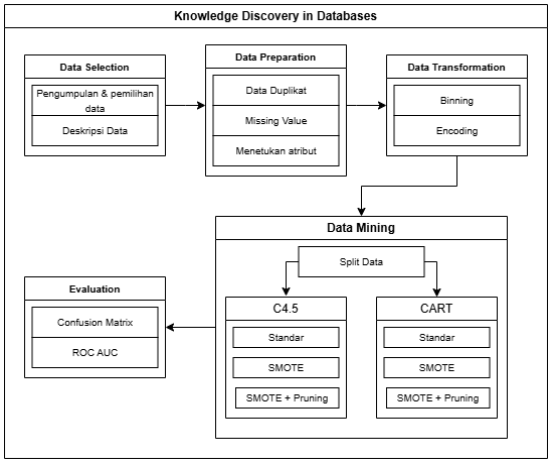
2.8. SNBP

Seleksi Nasional Berdasarkan Prestasi (SNBP) adalah jalur penerimaan mahasiswa baru di PTN yang menilai prestasi akademik

dan non-akademik siswa melalui rapor dan portofolio tanpa ujian tertulis. Peserta dipilih oleh sekolah berdasarkan rekam jejak prestasi selama pendidikan menengah atas [2][1].

3. METODE PENELITIAN

Penelitian ini bertujuan menerapkan proses data *mining* melalui tahapan *Knowledge Discovery in Databases* (KDD) untuk mengidentifikasi pola penerimaan calon mahasiswa baru Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang melalui jalur SNBP. Rancangan penelitian mengikuti lima tahapan KDD, yaitu *data selection*, data *preprpocessing*, *data transformation*, data *mining* dan evaluasi. Metode yang digunakan adalah *decision tree* dengan algoritma C4.5 dan CART, serta menerapkan metode SMOTE untuk menangani ketidakseimbangan data. Selain itu, diterapkan teknik post-*pruning* menggunakan *cost complexity pruning* guna menyederhanakan struktur pohon dan meningkatkan akurasi. Pendekatan ini diharapkan mampu memberikan wawasan yang lebih mendalam terhadap faktor-faktor yang memengaruhi seleksi penerimaan mahasiswa. Alur penelitian dengan melalui tahapan KDD dapat dilihat pada gambar 1. berikut.



Gambar 1. Metode penelitian

4. HASIL DAN PEMBAHASAN

4.1 Data Selection

Pada tahap ini dilakukan pengumpulan data awal dari arsip akademik Universitas Singaperbangsa Karawang, berupa data pendaftaran calon mahasiswa S1 Informatika dan Sistem Informasi melalui jalur SNBP tahun 2024 dan 2025. Dataset terdiri dari 2806 baris dan 9 atribut yang mencakup informasi seperti jurusan, jenis sekolah, nilai rapor, dan status kelulusan. Data ini akan digunakan untuk analisis prediksi kelulusan SNBP. Dari total 2806 calon mahasiswa yang mendaftar melalui jalur ini, hanya 138 dinyatakan lulus, sementara 2668 tidak lulus pada tahun 2024 dan 2025.

Setelah dilakukan proses pengumpulan data awal, selanjutnya dilakukan analisis atau pendeskripsian pada data dengan mendeskripsikan format serta keterangan atribut yang merujuk pada informasi dari bagian akademik Universitas Singaperbangsa Karawang. Data dideskripsikan menjadi atribut, tipe data dan keterangan seperti pada tabel 2.

Tabel 2. Deskripsi atribut

Atribut	Tipe Data	Keterangan
no peserta	Object	Nomor pendaftaran peserta SNBP
jurusan	Object	Jurusan peserta saat sekolah menengah
jenis_sekolah	Object	Jenis sekolah mengah peserta
ranking_versi _sekolah	Int	Ranking peserta di tingkat sekolah
rata_nilai_rapor	Float	Rata-rata nilai rapor peserta
Pilihan	Object	Pilihan pendaftaran
Prodi	Object	Program studi pilihan saat melakukan pendaftaran
tahun_pendaftaran	Int	Tahun pendaftaran
Status_kelulusan	Object	Status kelulusan peserta

4.2 Data Preprocessing

Setelah data dipilih, tahap berikutnya adalah data *preprocessing*. Langkah pertama yaitu memeriksa duplikasi data berdasarkan atribut "no_peserta" yang merupakan identitas unik setiap calon mahasiswa. Pemeriksaan ini penting untuk memastikan bahwa setiap entri dalam dataset benar-benar mewakili satu individu. Hasil pengecekan yang ditampilkan pada gambar 3 menunjukkan bahwa tidak terdapat data duplikat.

```
Duplikat no_peserta 0       False
1       False
2       False
3       False
4       False
        ...
2801    False
2802    False
2803    False
2804    False
2805    False
Length: 2806, dtype: bool
jumlah nilai yang duplikat 0
```

Gambar 2. Pengecekan data duplikat

Selanjutnya adalah menghapus atribut yang tidak relevan terhadap tujuan penelitian. Atribut "no peserta" dihapus karena hanya berfungsi sebagai penanda unik dan tidak memiliki kontribusi dalam proses analisis prediksi kelulusan.

Tahap terakhir adalah pengecekan *missing value* atau nilai yang hilang. Gambar 4 menunjukkan bahwa seluruh atribut dalam dataset terisi dengan lengkap, atribut yang akan digunakan juga telah terpilih sehingga data siap digunakan untuk tahap analisis berikutnya.

```
jurusan                0
jenis_sekolah          0
ranking_versi_sekolah  0
rata_nilai_rapor       0
Pilihan                0
Prodi                  0
Status_kelulusan       0
tahun_pendaftaran      0
dtype: int64
```

Gambar 3. Pengecekan missing value

4.3 Data Transformation

Pada tahap transformasi, data yang telah dibersihkan diubah ke dalam format seragam untuk memudahkan proses data *mining*. Proses transformasi meliputi konversi atribut kategorikal menjadi numerik, sekaligus penyeragaman tipe data setiap atribut untuk mendukung tahapan berikutnya. Atribut numerik seperti "ranking versi sekolah" dan "rata nilai rapor" dikategorikan menggunakan metode *binning equal-frequency*, yaitu membagi data menjadi tiga kelompok dengan jumlah observasi yang hampir sama. Untuk "ranking versi sekolah", label kategori dibuat terbalik, dimana angka kecil diberi label "tinggi" karena menunjukkan peringkat lebih baik. Sebaliknya, "rata nilai rapor" dilabeli secara berurutan dari "rendah" ke "tinggi". Setelah proses *binning*, kedua atribut diubah ke dalam tipe data *string* untuk keperluan analisis selanjutnya. Hasil *binning* tersebut dapat dilihat pada gambar 5 berikut.

```
   ranking_versi_sekolah rata_nilai_rapor
0                rendah           tinggi
1                rendah           tinggi
2                rendah           tinggi
3                tinggi           tinggi
4                sedang           sedang

=== Tipe data kedua kolom ===
ranking_versi_sekolah    object
rata_nilai_rapor         object
dtype: object
```

Gambar 4. Hasil binning

Distribusi pembagian data hasil *binning* dapat dilihat pada tabel 3 berikut.

Tabel 3. Distribusi hasil *binning*

Atribut	Kategori	Banyaknya Data
ranking_versi_sekolah	rendah	912
	sedang	948
	tinggi	946
rata_nilai_sekolah	rendah	936
	sedang	930
	tinggi	940

Selanjutnya, atribut "tahun pendaftaran" diubah dari tipe *integer* menjadi *string* untuk menyeragamkan tipe data, sehingga proses *label encoding* dapat dilakukan secara konsisten. Gambar 6 menunjukkan hasil perubahan seluruh atribut ke format yang seragam.

```
Data columns (total 8 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   jurusan               2806 non-null   object
 1   jenis_sekolah         2806 non-null   object
 2   ranking_versi_sekolah 2806 non-null   object
 3   rata_nilai_rapor      2806 non-null   object
 4   Pilihan               2806 non-null   object
 5   Prodi                 2806 non-null   object
 6   Status_kelulusan      2806 non-null   object
 7   tahun_pendaftaran     2806 non-null   object
dtypes: object(8)
```

Gambar 5. Penyamaan tipe data

Label encoding dilakukan menggunakan *LabelEncoder* dari *sklearn.preprocessing* untuk mengubah nilai kategorikal menjadi format numerik yang dapat diproses oleh algoritma. Gambar 7 menunjukan hasil *encoding* pada semua atribut pada dataset.

```
   jurusan  jenis_sekolah  ranking_versi_sekolah  rata_nilai_rapor  Pilihan  \
0        2               0                      0                 2        0
1        2               0                      0                 2        0
2        2               0                      0                 2        0
3        5               1                      2                 2        0
4        2               0                      1                 1        0

   Prodi  Status_kelulusan  tahun_pendaftaran
0      0                 1                  0
1      0                 1                  0
2      1                 1                  0
3      0                 1                  0
4      1                 1                  0
```

Gambar 6. Hasil label encoding

4.4 Data *Mining*

Pada tahap data *mining*, dilakukan pencarian pola pohon keputusan terbaik menggunakan algoritma C4.5 dan CART. Dataset yang telah dibersihkan dan ditransformasi, terdiri dari 2806 sampel, dibagi menggunakan teknik *train-test split* dalam pembagian data 80% *trining* dan 20% *testing*.

Pemodelan dimulai dengan inisialisasi, pelatihan model menggunakan data latih, lalu pengujian untuk mengevaluasi akurasi dan generalisasi model.

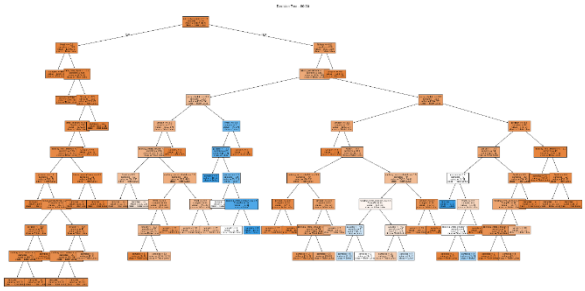
a. Algoritma C4.5

Pada algoritma C4.5, pohon keputusan dibangun berdasarkan *entropy* dan *information gain*. Atribut dengan *information gain* terbaik akan dijadikan sebagai *root node* pada pohon keputusan. Hasil dari perhitungan *entropy* dan *information gain*, lebih jelasnya dapat terlihat pada tabel 4 berikut.

Tabel 4. Hasil perhitungan *entropy* dan *information gain*

	Kategori	*Entropy*	*Information gain*
Total		0.2829	
Jurusan	BDP	0.8113	0.0069
	IPS	0.0000	
	MIPA	0.3128	
	PPLG	0.4987	
	TJKT	0.1787	
	TKJ	0.1496	
	Umum	0.3671	
Jenis sekolah	SMA	0.3109	0.0019
	SMK	0.2064	
Ranking Versi Sekolah	Rendah	0.2297	0.0016
	Sedang	0.2760	
	Tinggi	0.3363	
Rata nilai rapor	Rendah	0.0000	0.0639
	Sedang	0.0786	
	Tinggi	0.5785	
Pilihan	Pilihan 1	0.4105	0.0378
	Pilihan 2	0.0000	
Prodi	Informatika	0.2678	0.0003
	Sistem Informasi	0.3056	
Tahun pendaftaran	2024	0.3276	0.0021
	2025	0.3056	

Berdasarkan hasil perhitungan *entropy* dan *Information gain* pada tabel 5, nilai tertinggi terdapat pada atribut "rata nilai rapor" dengan nilai 0.0639. Oleh karena itu, atribut rata nilai rapor akan menjadi *node root*. Hasil pemodelan yang menunjukkan *decision tree* standar ditunjukan pada gambar 7 berikut.



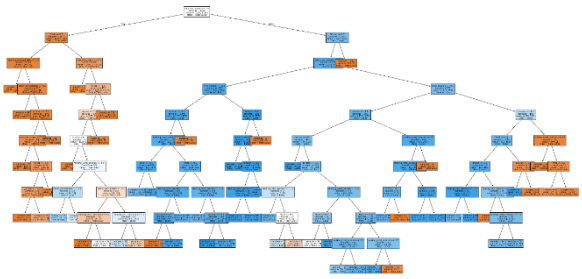
Gambar 7. *Decision tree* C4.5

Setelah membentuk *decision tree* pada tiap skenario, metode SMOTE diterapkan untuk mengatasi ketidakseimbangan kelas pada atribut "status kelulusan". Teknik ini mensintesis data baru pada kelas minoritas guna meningkatkan kinerja model dan mengurangi bias. Perbandingan distribusi data sebelum dan sesudah SMOTE ditampilkan pada tabel 6.

Tabel 5. Distribusi SMOTE

Kondisi	**0**	**1**
Sebelum SMOTE	2134	110
Setelah SMOTE	2134	2134

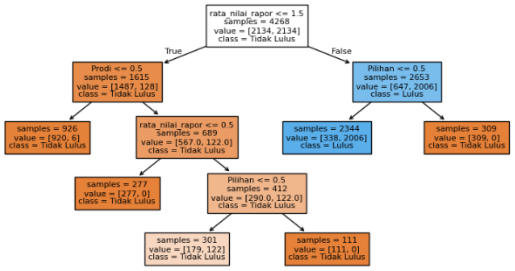
Model kemudian dievaluasi ulang untuk menilai efektivitas SMOTE dalam mengatasi ketidakseimbangan kelas serta meningkatkan akurasi dan performa model. Struktur pohon hasil SMOTE ditampilkan pada gambar 8.



Gambar 8. Hasil SMOTE C4.5 *split* data 80:20

Tahap selanjutnya adalah post-*pruning* menggunakan *cost complexity pruning* berbasis nilai *alpha* untuk menyederhanakan pohon keputusan dengan menghapus *node* yang tidak signifikan. Proses ini membandingkan biaya kompleksitas antar *node* dan menghilangkan yang tidak meningkatkan kinerja model secara signifikan. Setelah *pruning*, model dievaluasi ulang untuk menilai peningkatan akurasi dan performa. Gambar 9 menampilkan hasil

struktur pohon pasca-*pruning* setelah penerapan SMOTE.



Gambar 9. Hasil *pruning* C4.5 *split* data 80:20

Dari hasil pohon keputusan yang dihasilkan, *cost complexity pruning* berhasil mengurangi kompleksitas model agar tidak *overfitting*.

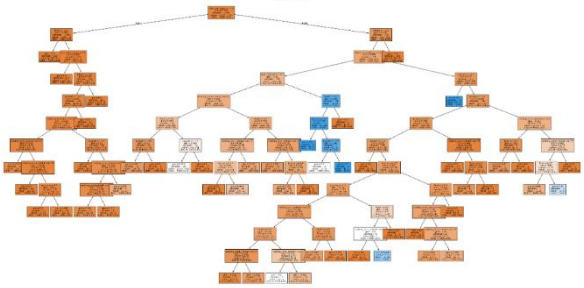
b. Algoritma CART

Algoritma CART membangun pohon keputusan berdasarkan *gini impurity*, dengan memilih atribut yang menghasilkan nilai *gini* terendah sebagai *split node* untuk menghasilkan pemisahan data yang paling murni. Pemilihan atribut terbaik dalam algoritma CART didasarkan pada *gini index*, di mana nilai yang lebih kecil menunjukkan *node* yang lebih murni. Proses dimulai dengan menghitung *gini root*, dilanjutkan dengan *gini split* untuk tiap atribut, dan selisihnya disebut *Goodness of Split* (GoS). Atribut dengan GoS tertinggi dipilih sebagai pemilah utama. Berikut adalah tabel 6, yang menunjukan perhitungan *gini root*, *gini split* dan *goodness of split*.

Tabel 6. Hasil *gini root*, *gini split* dan goodness of *split*

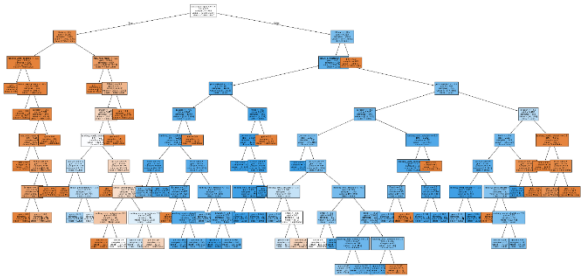
Pemilah	*Gini root*	*Gini split*	Goodness of *split*
rata_nilai_rapor	0.0935	0.0856	0.0079
Pilihan	0.0935	0.0903	0.0033
jurusan	0.0935	0.0927	0.0008
tahun_pendaftaran	0.0935	0.0933	0.0003
jenis_sekolah	0.0935	0.0933	0.0002
ranking_versi_sekolah	0.0935	0.0933	0.0002
Prodi	0.0935	0.0935	0.0000

Berdasarkan table 6, atribut "rata nilai rapor" memiliki nilai GoS tertinggi (0.0079), sehingga dipilih sebagai pemilah utama karena paling mampu mengurangi ketidakmurnian kelas. Setelah menentukan atribut yang akan digunakan dalam pemodelan berdasarkan hasil *goodess of split*. Hasil pemodelan yang menunjukkan struktur *decision tree* dapat dilihat pada gambar 10.



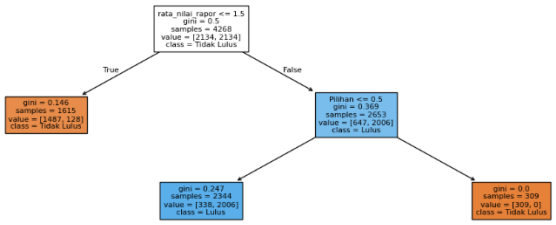
Gambar 10. *Decision tree* CART 80:20

Sama seperti pada implementasi algoritma C4.5, metode SMOTE juga diterapkan untuk mengatasi ketidakseimbangan kelas pada atribut "status kelulusan" untuk algoritma CART. Model kemudian dievaluasi ulang untuk menilai efektivitas SMOTE dalam mengatasi ketidakseimbangan kelas serta meningkatkan akurasi dan performa model. Struktur pohon hasil SMOTE ditampilkan pada gambar 11.



Gambar 11. Hasil SMOTE CART *split* data 80:20

Tahap *post-pruning* menggunakan *cost complexity pruning* juga diterapkan pada algoritma CART untuk menyederhanakan pohon keputusan dengan menghapus *node* yang tidak signifikan. Gambar 12 menampilkan hasil struktur pohon pasca-*pruning* setelah penerapan SMOTE.



Gambar 12. Hasil *Pruning* CART *split* data 80:20

Dari hasil pohon keputusan yang telah terbentuk, penerapan metode SMOTE dapat meningkatkan jumlah *node* karena menambah variasi data melalui data sintesis, sehingga memaksa pohon mempelajari pemisahan yang lebih kompleks. Sebaliknya, proses *pruning* berhasil menyederhanakan struktur pohon dengan menghapus *node-node* yang tidak terlalu berkontribusi, terutama yang hanya memisahkan sedikit data atau berpotensi menyebabkan *overfitting*.

4.5 Evaluation

Pada tahap evaluasi, performa model diukur menggunakan beberapa metrik, yaitu *confusion matrix, classification report* (yang mencakup akurasi, presisi, *recall*, dan *F1-score*) serta kurva *Receiver Operating Characteristic* (ROC) dan nilai *Area Under Curve* (AUC) untuk mengukur kemampuan model dalam membedakan antara kelas lulus dan tidak lulus. Hasil penerapan C4.5 baik itu standar, dengan SMOTE maupun dengan *pruning* dapat dilihat pada tabel 7.

Tabel 7. Hasil evaluasi

Metode	*accuracy*	*precision*	*recall*	*F1-score*	AUC
C4.5	96%	73%	55%	57%	0.89
C4.5 SMOTE	86%	61%	84%	65%	0.89
C4.5 + SMOTE + *pruning*	85%	62%	89%	65%	0.92
CART	95%	81%	57%	61%	0.89
CART+ SMOTE	86%	61%	84%	65%	0.89
CART + SMOTE + *pruning*	85%	62%	89%	65%	0.89

Berdasarkan tabel 7, hasil pengujian menunjukkan bahwa algoritma C4.5 standar mencapai akurasi tertinggi yaitu 96%, sedangkan algoritma CART mencapai akurasi hingga 95%. Namun hasil kedua algorima ini cenderung bias terhadap kelas mayoritas. Penerapan SMOTE menurunkan akurasi menjadi 86% pada algoritma C4.5 dan CART, namun meningkatkan sensitivitas terhadap kelas minoritas dengan meningkatnya *recall* yang lebih merata. Sementara itu, kombinasi SMOTE dan *pruning* menghasilkan akurasi relatif stabil di angka 85%, yang menunjukkan meskipun *pruning* menyederhanakan model, dampaknya terhadap peningkatan akurasi tidak terlalu signifikan.

Berdasarkan hasil evaluasi, model *decision tree* C4.5 dengan penerapan metode SMOTE dan *pruning* menunjukkan kinerja paling optimal dibandingkan pendekatan lainnya. Model ini menunjukkan keseimbangan antara stabilitas dan performa, terutama pada *F1-score* dan AUC dengan nilai *F1-score* tertinggi sebesar 65% dan AUC sebesar 0,92 menunjukkan kemampuan klasifikasi yang baik tanpa *overfitting*. Selain itu, algoritma C4.5 terbukti lebih unggul daripada CART dalam memprediksi penerimaan mahasiswa baru di Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang. Meskipun nilai evaluasi lainnya serupa, C4.5 mencatat AUC sebesar 0.92 dibandingkan CART yang hanya 0.89, menandakan kemampuan yang lebih baik dalam menangani data tidak seimbang. Dengan demikian, pendekatan terbaik dalam penelitian ini adalah model C4.5 dengan SMOTE dan *pruning*.

5. KESIMPULAN

a. algoritma C4.5 dan CART mampu membentuk model klasifikasi yang layak digunakan untuk mendukung proses seleksi calon mahasiswa baru di Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang. Kedua algoritma berhasil mengidentifikasi atribut-atribut yang paling berpengaruh terhadap keputusan penerimaan, dengan "rata-rata nilai rapor" sebagai atribut paling dominan dalam membedakan kelas dilanjutkan dengan atribut "pilihan" dan "prodi".
b. Model C4.5 dengan kombinasi SMOTE dan *pruning* memberikan kinerja paling seimbang, dengan akurasi 85% dan ROC-AUC 0.92, lebih baik dari CART dengan konfigurasi serupa (ROC-AUC 0.89). Temuan ini menegaskan

efektivitas C4.5 dan CART dalam prediksi penerimaan mahasiswa baru serta potensi *decision tree* sebagai alat bantu seleksi berbasis data.

DAFTAR PUSTAKA

[1] A. Prayogi and R. Nasrullah, "UTBK-SNBT Training and Information Provision for High School and Equivalent Students in Pekalongan," MID: Journal of Sustainable Community Development, vol. 2, no. 1, pp. 26–31, 2024.

[2] R. B. Riry, "Sosialisasi Perguruan Tinggi Negeri (PTN) Universitas Pattimura Jalur SNBP, SNBT, Mandiri dan KIP Kuliah Universitas Pattimura Kepada Siswa/Siswi SMA/SMK/MA Sederajad di Kabupaten Buru," Jurnal Pengabdian Arumbai, vol. 2, no. 2, pp. 138 – 145, 2024. doi: https://doi.org/10.30598/arumbai.vol2.iss2.pp138-145.

[3] I. Lestari, D. Fitria, Syafriandi, and A. Salma, "Comparison of the C5.0 Algorithm and the CART Algorithm in Stroke Classification," *UNP Journal of Statistics and Data Science*, vol. 2, no. 1, pp. 90–98, Feb. 2024, doi: 10.24036/ujsds/vol2-iss1/144.

[4] I. Jayanto, "Analisis Perbandingan Algoritma Decision Tree untuk Prediksi Karyawan dengan Potensi Atrisi di PT. XYZ," Jurnal Informatika Komputer, Bisnis Dan Manajemen (FAHMA), vol. 22, no. 1, pp. 49–59, Jan. 2024, doi: 10.61805/fahma.v22i1.112.

[5] C. Nas, "Data Mining Prediksi Minat Calon Mahasiswa Memilih Perguruan Tinggi Menggunakan Algoritma C4.5," *Jurnal Manajemen Informatika (JAMIKA)*, vol. 11, no. 2, pp. 131–145, Sep. 2021, doi: 10.34010/jamika.v11i2.5506.

[6] B. Charbuty and A. Abdulazeez, "Classification Based on Decision Tree Algorithm for Machine Learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, Mar. 2021, doi: 10.38094/jastt20165.

[7] M. A. F. Firmansyah and A. A. Setiawan, "Perbandingan Algoritma C 4.5 dan Naïve Bayes Dalam Memprediksi Kelulusan Mahasiswa," *Th*, vol. 3, no. 1, pp. 20–26, 2023, doi: https://doi.org/10.31331/jsitee.v3i1.2703.

[8] P. B. N. Setio, D. R. S. Saputro, and B. Winarno, "Klasifikasi dengan Pohon Keputusan Berbasis Algoritme C4.5," vol. 3, pp. 64–71, 2020, https://journal.unnes.ac.id/sju/index.php/prisma/

[9] G. Taufik and D. Jatmika, "Penerapan Algoritma C4.5 Untuk Klasifikasi Keberhasilan Pengiriman Barang," Jurnal Inovtek Polbeng - Seri Informatika, vol. 6, no. 1, 2021.

[10] M. Yunus, M. K. Biddinika, and A. Fadlil, "Classification of Stunting in Children Using the C4.5 Algorithm," *Jurnal Online Informatika*, vol. 8, no. 1, pp. 99–106, Jun. 2023, doi: 10.15575/join.v8i1.1062.

[11] F. Maisa Hana, W. Cholid Wahyudin, S. Ulya, and D. Setia Negara, "Implementasi Algoritma Cart dalam Klasifikasi Penyakit Diabetes," Jurnal Ilmu Komputer dan Matematika, vol. 4, no.1, pp. 1 – 8, 2023.

[12] I. Lestari, D. Fitria, Syafriandi, and A. Salma, "Comparison of the C5.0 Algorithm and the CART Algorithm in Stroke Classification," *UNP Journal of Statistics and Data Science*, vol. 2, no. 1, pp. 90–98, Feb. 2024, doi: 10.24036/ujsds/vol2-iss1/144.

[13] Y. Kustiyahningsih, B. K. Khotimah, D. R. Anamisa, M. Yusuf, T. Rahayu, and J. Purnama, "Decision Tree C 4.5 Algorithm for Classification of Poor Family Scholarship Recipients," *IOP Conf Ser Mater Sci Eng*, no. 1, pp. 1–7, May 2021, doi: 10.1088/1757-899X/1125/1/012048.

[14] H. Hairani, K. E. Saputro, and S. Fadli, "K-means-SMOTE for handling class imbalance in the classification of diabetes with C4.5, SVM, and naive Bayes," *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 2, pp. 89–93, Apr. 2020, doi: 10.14710/jtsiskom.8.2.2020.89-93.

[15] H. Yuliansyah, R. A. P. Imaniati, A. Wirasto, and M. Wibowo, "Predicting Students Graduate on Time Using C4.5 Algorithm," *Journal of Information Systems Engineering and Business Intelligence*, vol. 7, no. 1, p. 67, Apr. 2021, doi: 10.20473/jisebi.7.1.67-73.

[16] L. Qadrini, A. Seppewali, and A. Aina, "Decision Tree Dan Adaboost pada Klasifikasi Penerima Program Bantuan Sosial," Jurnal Inovasi Penelitian, vol. 2, no. 1, pp. 1959–1966, 2021, doi: https://doi.org/10.47492/jip.v2i7.1046.

[17] S. Febriani and H. Sulistiani, "Analisis Data Hasil Diagnosa untuk Klasifikasi Gangguan Kepribadian Menggunakan Algoritma C4.5," Jurnal Teknologi dan Sistem Informasi (JTSI), vol. 2, no. 4, pp. 89–95, 2021.