

PERBANDINGAN METODE KLASIFIKASI PENGUJIAN KENDARAAN BERMOTOR DENGAN ALGORITMA C4.5 DAN NAÏVE BAYES

Fahreza Adams Lazuardy¹⁾, Ahmad Homaidi²⁾, Ahmad Lutfi³⁾

Jurusan Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Ibrahimy
Jl. KHR. Syamsul Arifin No.1-2, Sukorejo, Situbondo 68374, Jawa Timur, Indonesia
e-mail : ¹⁾fa.lazuardy@gmail.com, ²⁾ahmadhomaidi@ibrahimiy.ac.id, ³⁾ahmadlutfi.14@gmail.com

ABSTRAK

Penelitian ini bertujuan untuk membandingkan metode klasifikasi pengujian kendaraan bermotor pada algoritma C4.5 dengan Naïve Bayes di Dinas Perhubungan Kabupaten Situbondo. Dataset yang digunakan berjumlah 3.264 record yang dibagi menjadi 2 subset data, data training dan data testing. Atribut yang dianalisis meliputi tahun pembuatan, kepemilikan kendaraan, emisi opasitas timbal, total gaya pengereman, standarisasi klakson, kelayakan ban kendaraan, kekuatan sinar lampu, penyimpangan pancar lampu, dan hasil pengujian. Hasil perbandingan ini menghasilkan dalam algoritma C4.5 memiliki akurasi senilai 99.69% dengan precision 99.68% dan recall 100% untuk kelas “LULUS”, serta precision 100% dan recall 90.48% untuk kelas “TIDAK LULUS”. Sementara algoritma Naïve Bayes menunjukkan akurasi senilai 99.13% dengan precision 99.59% dan recall 99.51% untuk kelas “LULUS” serta precision 81.82% dan recall 84.38% untuk kelas “TIDAK LULUS”. C4.5 lebih unggul mendeteksi kendaraan yang “TIDAK LULUS”, sedangkan Naïve Bayes lebih efisien dalam komputasi. Temuan ini menyimpulkan bahwa algoritma C4.5 lebih efektif untuk pola interaksi data yang kompleks, sedangkan Naïve Bayes menawarkan kecepatan komputasi yang lebih tinggi. Pemilihan algoritma harus mempertimbangkan karakteristik dataset dan kebutuhan spesifik dari aplikasi pengujian kendaraan bermotor.

Kata Kunci: Data mining, C4.5, Naïve Bayes, Klasifikasi, Pengujian Kendaraan Bermotor, Akurasi.

ABSTRACT

This study compares vehicle inspection classification methods using the C4.5 and Naïve Bayes algorithms at the Department of Transportation in Situbondo Regency. The dataset comprises 3,264 records, divided into training and testing subsets. Analyzed attributes include year of manufacture, vehicle ownership, lead opacity emissions, total braking force, horn standardization, tire feasibility, headlight strength, headlight deviation, and test results. The C4.5 algorithm achieves 99.69% accuracy, with 99.68% precision and 100% recall for the “PASS” class, and 100% precision and 90.48% recall for the “FAIL” class. Naïve Bayes shows 99.13% accuracy, with 99.59% precision and 99.51% recall for the “PASS” class, and 81.82% precision and 84.38% recall for the “FAIL” class. C4.5 is superior in detecting “FAIL” vehicles, while Naïve Bayes is more computationally efficient. These findings suggest that the C4.5 algorithm is more effective for complex data interaction patterns, whereas Naïve Bayes offers higher computational speed. Algorithm selection should consider dataset characteristics and the specific needs of vehicle inspection applications.

Keywords: Data mining, C4.5, Naïve Bayes, Classification, Motor Vehicle Testing, Accuracy.

I. PENDAHULUAN

1.1 Latar Belakang Penelitian

Dinas Perhubungan Kabupaten Situbondo, sebagaimana diatur oleh Peraturan Bupati Situbondo

Nomor 24 Tahun 2022 tentang Kedudukan, Susunan Organisasi, Uraian Tugas, dan Fungsi Dinas Perhubungan Kabupaten Situbondo, adalah sebuah lembaga yang berfungsi membantu Bupati dalam melaksanakan urusan pemerintahan yang menjadi

kewenangan daerah di bidang perhubungan. Selain itu, lembaga ini juga bertugas melaksanakan tugas pembantuan yang diberikan kepada daerah. Salah satu unit penting di dalamnya adalah Unit Pelaksana Teknis Dinas yang bertugas memberikan layanan pengujian kendaraan bermotor. Unit ini bertanggung jawab atas pelaksanaan kegiatan teknis operasional dan/atau teknis penunjang, serta urusan pemerintahan yang bersifat pelaksanaan dari organisasi induknya, yang pada prinsipnya tidak berkaitan langsung dengan perumusan dan penetapan kebijakan daerah [1].

Unit Pelaksana Teknis Pengujian Kendaraan Bermotor bertanggung jawab menyelenggarakan layanan pengujian kendaraan bermotor secara cuma-cuma bagi masyarakat. Proses ini mencakup pendaftaran, pengisian formulir, dan pemeriksaan teknis untuk memastikan kendaraan memenuhi standar keselamatan yang telah ditetapkan. Meskipun layanan ini diberikan secara gratis, masih ditemukan sejumlah kendaraan yang tidak laik jalan dan gagal dalam pengujian. Kondisi ini sering kali disebabkan oleh kurangnya perawatan rutin serta usia kendaraan yang telah melebihi satu dekade. Kendaraan yang tidak dirawat dengan baik cenderung mengalami berbagai kerusakan, seperti emisi gas buang yang berlebihan, kondisi ban yang terlalu tipis, pancar lampu yang redup atau tidak berfungsi, sistem pengereman yang tidak optimal, serta tingkat kebisingan klakson yang tidak sesuai dengan standar pabrik. Keadaan ini tidak hanya membahayakan pengendara, tetapi juga para pengguna jalan lainnya. Pengujian kendaraan bermotor merupakan upaya esensial untuk menjamin keselamatan dan efisiensi operasional kendaraan di jalan raya. Menurut data dari Kepolisian Resor Situbondo yang menunjukkan bahwa persentase kecelakaan lalu lintas di wilayah Kabupaten Situbondo meningkat sebesar 575,92% dimana sebagian besar disebabkan oleh kendaraan yang tidak laik jalan [2].

Dalam konteks ini, teknologi data mining berperan sangat penting dalam meningkatkan akurasi dan efisiensi pengujian kendaraan bermotor. Dengan melakukan perbandingan pada algoritma C4.5 dengan Naïve Bayes, proses klasifikasi dapat dianalisa dan dapat ditemui algoritma mana yang

lebih tepat dan efisien. Hal ini memungkinkan identifikasi kendaraan yang tidak laik jalan dilakukan dengan lebih cepat dan akurat. Teknologi ini diharapkan dapat mendukung Dinas Perhubungan Kabupaten Situbondo dalam menjalankan tugasnya dengan lebih efektif, serta memberikan kontribusi yang signifikan terhadap peningkatan keselamatan lalu lintas.

1.2. Tinjauan Penelitian

Penelitian sebelumnya pernah dilakukan oleh Eka Fitriani dalam jurnal "SISTEMASI: Jurnal Sistem Informasi" melakukan perbandingan antara algoritma C4.5 dan Naïve Bayes dalam menilai kualifikasi penerima pada Program Keluarga Harapan (PKH). Penelitian ini menyoroti masalah verifikasi data yang belum tepat sasaran dalam pemilihan penerima PKH, sehingga sering terjadi ketidaktepatan sasaran. Melalui analisis komparatif, penelitian ini menemukan bahwa algoritma C4.5 menghasilkan akurasi sebesar 91.25% dengan AUC 0.930, sedangkan algoritma Naïve Bayes memiliki akurasi 87.11% dengan AUC 0.923. Hasil ini menunjukkan bahwa algoritma C4.5 lebih unggul dalam menentukan kelayakan penerima PKH dibandingkan dengan Naïve Bayes. Penelitian ini juga menegaskan pentingnya pemilihan algoritma yang tepat berdasarkan karakteristik data dan tujuan aplikasi, di mana C4.5 lebih efektif untuk data dengan pola interaksi yang kompleks, sementara Naïve Bayes lebih menawarkan pada kecepatan komputasi yang tinggi [3].

Penelitian ini memiliki relevansi langsung dengan perbandingan metode klasifikasi pengujian kendaraan bermotor menggunakan algoritma C4.5 dan Naïve Bayes di Dinas Perhubungan Kabupaten Situbondo. Kedua penelitian ini menggarisbawahi pentingnya pemilihan algoritma yang sesuai untuk meningkatkan akurasi dan efisiensi dalam proses klasifikasi. Dalam konteks pengujian kendaraan bermotor, pemanfaatan algoritma yang tepat dapat membantu dalam mengidentifikasi kendaraan yang tidak laik jalan dengan lebih cepat dan akurat. Dengan demikian, hasil penelitian Fitriani memberikan dasar yang kuat untuk menerapkan metode yang sama dalam konteks yang berbeda, yakni pengujian

kendaraan bermotor, untuk mencapai tujuan yang serupa yaitu meningkatkan ketepatan dan efisiensi proses klasifikasi.

1.3. Landasan Teori

a. Data Mining

Data Mining adalah kombinasi secara logis antara pengetahuan data, dan Analisa statistic yang dikembangkan dalam pengetahuan bisnis atau suatu proses yang menggunakan Teknik statistic, matematika, kecerdasan buatan, tiruan dan machine-learning untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat bagi pengetahuan yang terkait dari berbagai database besar [4].

b. Klasifikasi

Klasifikasi adalah tipe analisis data yang dapat membantu orang dalam menentukan kelas label dari sampel yang ingin di klasifikasi untuk menemukan hubungan antara atribut masukan dan atribut target [5].

c. Algoritma C 4.5

Algoritma C 4.5 menghasilkan beberapa rule dan pohon Keputusan dengan tujuan untuk meningkatkan keakuratan dari prediksi yang sedang dilakukan, disamping itu algoritma C 4.5 merupakan algoritma yang mudah dimengerti [6].

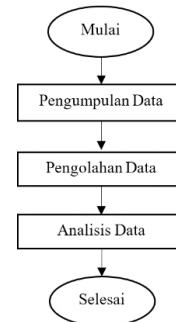
d. Naïve Bayes

Metode Naïve Bayes merupakan pendekatan statistik untuk melakukan inferensi induksi pada persoalan klasifikasi. Keuntungan menggunakan Naïve Bayes adalah metode ini hanya membutuhkan jumlah data pelatihan (Training Data) yang kecil untuk menentukan parameter yang diperlukan dalam proses klasifikasi [7].

II. METODE PENELITIAN

2.1 Tahapan Penelitian

Langkah-langkah metodologis yang diambil oleh peneliti akan diuraikan pada gambar 1 [8].



Gambar 1. Tahapan Penelitian

Berikut penjabaran gambar 1

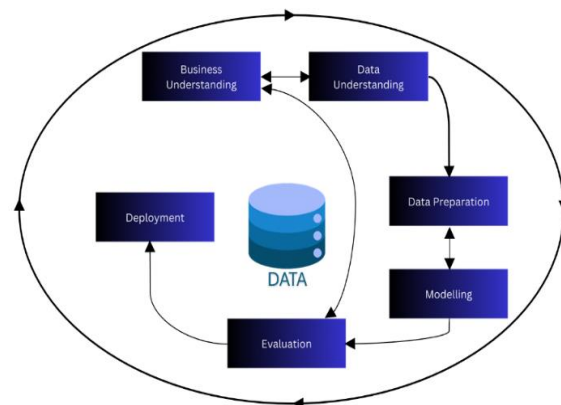
1. Data dikumpulkan melalui teknik ekspor database SQL dari aplikasi Pengujian Kendaraan Bermotor yang digunakan oleh Dinas Perhubungan Kabupaten Situbondo.

2. Pemrosesan data mencakup pengolahan data mentah yang mengandung beberapa informasi yang tidak relevan. Oleh karena itu, diperlukan tahap persiapan data untuk memastikan data siap dianalisis.

3. Data yang telah diproses sebelumnya dianalisis menggunakan metode CRISP-DM.

2.2 Metode Analisa Data

Pendekatan analisis data mining dalam penelitian ini menggunakan model CRISP-DM [9].



Gambar 2. Tahapan CRISP – DM.

Tahapan CRISP – DM sebagai berikut :

1. Fase Pemahaman Bisnis

Penelitian ini bertujuan untuk menerapkan metode data mining dengan Algoritma C4.5 Decision Tree dan Naïve Bayes guna membandingkan hasil klasifikasi pengujian kendaraan bermotor dengan akurasi tinggi. Dengan demikian, pola atau aturan

yang dihasilkan dari pohon keputusan C4.5 dan hasil klasifikasi dari Naïve Bayes dapat digunakan sebagai dasar untuk pengambilan keputusan dan kriteria yang jelas di masa depan.

2. Fase Pemahaman Data

Pada fase ini, data dikumpulkan menggunakan basis data dari aplikasi pengujian kendaraan bermotor di Dinas Perhubungan Kabupaten Situbondo untuk memperoleh informasi hasil pengujian kendaraan yang mencakup 3.264 record.

3. Fase Persiapan Data

Langkah-langkah dalam fase persiapan data adalah sebagai berikut :

a. Tahap Seleksi Data

Pada tahapan penyeleksian data, perlu dilakukan seleksi atribut dalam dataset karena tidak semua atribut pada dataset awal akan diolah dalam proses data mining. Dataset yang telah melalui seleksi dapat dilihat pada Tabel 1.

Tabel 1. Atribut Dataset yang siap diolah

No.	Atribut
1.	Tahun produksi kendaraan
2.	Kepemilikan kendaraan
3.	Emisi opasitas timbal
4.	Total gaya pengereman
5.	Standarisasi suara klakson
6.	Kelayakan ban
7.	Kekuatan sinar lampu utama
8.	Penyimpangan pancar lampu utama
9.	Hasil Pengujian

b. Tahap Pembersihan Data

Pada tahap ini, Data yang tidak lengkap atau tidak relevan dihapus melalui proses pembersihan data.

c. Diskretisasi data

Pada tahap ini, tahun produksi kendaraan diklasifikasikan menjadi dua kategori: kurang dari 10 tahun dan lebih dari 10 tahun, sebagaimana ditampilkan pada Tabel 2.

Tabel 2. Atribut hasil diskretisasai

Tahun Produksi Kendaraan	Atribut
1979 - 2013	< 10 TAHUN
2014 – 2024	> 10 TAHUN

d. Penentuan Atribut Target

Setelah proses diskretisasi, langkah berikutnya adalah menetapkan atribut dataset yang akan digunakan sebagai label, yakni atribut dataset hasil yang

mencakup nilai keputusan 'LULUS' dan 'TIDAK LULUS'.

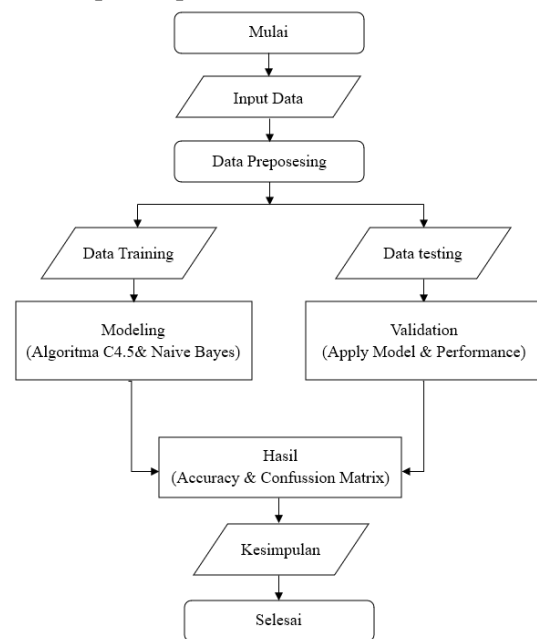
Pembagian Data

Penelitian ini melibatkan 3.264 record data yang dibagi menjadi 80% untuk data training dan 20% untuk data testing. Data tersebut akan diterapkan dalam penghitungan secara manual dengan tools RapidMiner.

Fase Pemodelan

Teknik pemodelan data mining yang diterapkan adalah metode klasifikasi dengan algoritma C 4.5 Decision Tree dan Naïve Bayes, di mana pemodelan dan desain pengujian model dilakukan menggunakan tools RapidMiner untuk memungkinkan analisis dan penilaian akurat terhadap setiap model.

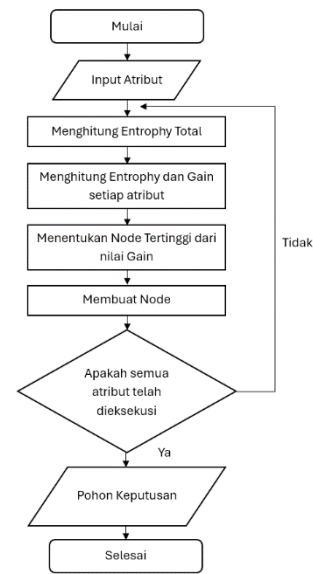
Flowchart analisis data mining dengan algoritma C4.5 ditampilkan pada Gambar 3.



Gambar 3. Flowchart Analisa Data Mining Penjabaran dari Flowchart Analisa Data Mining adalah sebagai berikut :

1. Mulai: Tahap awal di mana proses dimulai.
2. Input Data: Data yang akan digunakan dalam analisis dimasukkan ke dalam sistem.
3. Data Preprocessing: Data yang telah dimasukkan kemudian diproses untuk mempersiapkan data yang bersih dan sesuai untuk analisis. Tahap ini mencakup pemilihan atribut, pembersihan data, dan pendiskretan atribut.

4. Data Training: Data yang telah diproses dibagi menjadi dua bagian: data training dan data testing. Data training digunakan untuk melatih model.
5. Modeling (Algoritma C4.5 & Naïve Bayes): Pada tahap ini, data training digunakan untuk membangun model dengan menggunakan algoritma C4.5 Decision Tree dan Naïve Bayes. Model ini dibuat untuk memahami pola dalam data.
6. Data Testing: Data testing digunakan untuk menguji model yang telah dibangun. Tujuan dari tahap ini adalah untuk mengevaluasi seberapa baik model dapat memprediksi hasil berdasarkan data yang tidak terlihat sebelumnya.
7. Validation (Apply Model & Performance): Model yang telah dibangun diaplikasikan pada data testing untuk mengukur performa model. Evaluasi dilakukan dengan melihat akurasi dan confusion matrix untuk memahami seberapa baik model melakukan klasifikasi.
8. Hasil (Accuracy & Confusion Matrix): Hasil dari evaluasi model ditampilkan, termasuk tingkat akurasi dan confusion matrix yang menunjukkan detail performa model dalam mengklasifikasikan data.
9. Kesimpulan: Berdasarkan hasil evaluasi, kesimpulan diambil mengenai performa model dan pola yang dihasilkan. Kesimpulan ini digunakan untuk membuat keputusan yang informatif.
10. Selesai: Proses analisis data mining berakhir setelah kesimpulan diambil.



Gambar 4. Flowchart Algoritma C4.5

Berikut penjabaran dari Flowchart Algoritma C4.5 untuk membangun pohon keputusan.

1. Mulai: Tahap awal di mana proses dimulai.
2. Input atribut : Data atribut dimasukkan ke dalam sistem untuk analisis.
3. Menghitung entropy total : Entropy total dari dataset dihitung untuk mengukur ketidakpastian dalam data. Entropy ini akan menjadi acuan untuk menentukan seberapa baik atribut memisahkan data . Dalam penghitungan entropy total menggunakan persamaan berikut.

$$\text{Entropy}(S) = - \sum_{i=1}^n p_i \log_2(p_i)$$
 dimana :
 - S adalah himpunan data atau dataset
 - n adalah jumlah kelas dalam dataset
 - p_i adalah probabilitas atau proporsi kasus dalam kelas i .
4. Menghitung entropy beserta gain pada tiap atribut : Setiap atribut dalam dataset dianalisis untuk menghitung entropy dan information gain. Information gain digunakan untuk mengukur seberapa banyak ketidakpastian berkurang saat data dipisahkan oleh atribut tersebut. Penghitungan entropy beserta gain untuk setiap atribut menggunakan persamaan berikut
 - a. Menghitung entropy untuk setiap subset atribut menggunakan persamaan berikut.

b. Flowchart Algoritma C4.5

Pada diagram ini dapat dijelaskan bahwa langkah-langkah proses dalam algoritma C4.5 dapat dimulai dari pengambilan data mentah, preprocessing data, pemilihan atribut terbaik untuk membangun pohon keputusan, hingga pembentukan cabang-cabang pohon berdasarkan nilai atribut, serta evaluasi dan validasi model untuk menghasilkan pohon keputusan yang optimal. Dapat dilihat di gambar 4.

$$\text{Entropy}(S_i) = - \sum_{j=1}^m p_{ij} \log_2 (p_{ij})$$

dimana :

- S_i adalah subset data hasil pemisahan berdasarkan atribut A .
 - m adalah jumlah kelas dalam subset S_i .
 - p_{ij} adalah probabilitas atau proporsi kasus dalam kelas j di subset S_i .
- b. Menghitung Gain untuk setiap atribut menggunakan persamaan berikut.

$$\text{Gain}(A) = \text{Entropy}(S) - \sum_{i=1}^k \left(\frac{|S_i|}{|S|} \right) \text{Entropy}(S_i)$$

dimana :

- A adalah atribut yang digunakan untuk pemisahan
 - S adalah himpunan data atau dataset
 - S_i adalah subset data hasil pemisahan berdasarkan atribut A .
 - $|S_i|$ adalah jumlah kasus dalam subset S_i
 - $|S|$ adalah jumlah total kasus dalam dataset
 - k adalah jumlah subset yang dihasilkan dari pemisahan oleh atribut A .
5. Menentukan root node : Atribut dengan nilai gain tertinggi dipilih sebagai root node (node akar) karena atribut ini paling baik dalam memisahkan data berdasarkan ketidakpastian yang dikurangi.
 6. Melakukan perulangan proses : Node dibuat berdasarkan atribut yang telah dipilih sebagai root node. Proses ini mencakup pembagian data menjadi subset yang lebih kecil berdasarkan nilai atribut.
 7. Pohon Keputusan : Proses ini memeriksa apakah semua atribut telah dieksekusi dan dipertimbangkan dalam pembagian data. Jika belum, proses kembali ke langkah perhitungan entropy dan gain untuk atribut berikutnya.
 - Tidak: Jika belum semua atribut dieksekusi, kembali ke langkah menghitung entropy dan gain untuk atribut yang tersisa.
 - Ya: Jika semua atribut telah dieksekusi, lanjutkan ke langkah berikutnya.
 8. Setelah semua atribut dieksekusi dan node-node dibuat, pohon keputusan lengkap dapat dibangun.

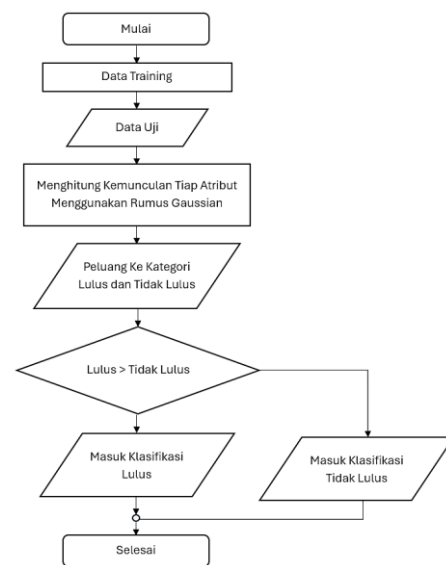
Pohon keputusan ini digunakan untuk memprediksi hasil berdasarkan atribut yang ada.

9. Selesai: Proses penerapan algoritma C4.5 berakhir setelah pohon keputusan selesai dibangun.

Flowchart ini secara keseluruhan menjelaskan langkah-langkah sistematis dalam menerapkan algoritma C4.5, dimulai dari input data atribut, perhitungan entropy dan information gain, hingga pembentukan pohon keputusan yang dapat digunakan untuk klasifikasi data.

Flowchart Algoritma Naïve Bayes

Berikut adalah flowchart algoritma Naïve Bayes dapat dilihat pada Gambar 5.



Gambar 5. Flowchart Algoritma Naïve Bayes
 Berikut penjabaran Flowchart Algoritma Naïve Bayes pada setiap tahapannya.

- Mulai: Tahap awal di mana proses dimulai.
- Data Training: Data yang akan digunakan untuk melatih model Naïve Bayes dimasukkan ke dalam sistem. Data ini digunakan untuk memahami pola dan hubungan antara atribut dan label.
- Data Uji: Data yang akan digunakan untuk menguji model dimasukkan ke dalam sistem. Data ini berfungsi untuk mengevaluasi seberapa baik model dapat memprediksi hasil pada data yang belum pernah dilihat sebelumnya.
- Menghitung Kemunculan Tiap Atribut Menggunakan Rumus Gaussian: Pada tahap ini, model menghitung probabilitas kemunculan setiap atribut dalam data training menggunakan rumus Gaussian. Rumus Gaussian digunakan untuk memperkirakan distribusi

data dan menghitung probabilitas berdasarkan distribusi normal dengan penjabaran rumus sebagai berikut:

$$P(x|C) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

dimana :

- $P(x|C)$ adalah probabilitas likelihood dari nilai x diberikan ke kelas C .
 - μ adalah rata-rata (mean) dari atribut untuk kelas C .
 - σ adalah simpangan baku (standart deviation) dari atribut untuk kelas C .
 - exp adalah fungsi eksponensial (e adalah basis logaritma natural, sekitar 2.71828).
5. Peluang ke Kategori Lulus dan Tidak Lulus
Berdasarkan perhitungan probabilitas atribut, model menghitung peluang setiap record untuk masuk ke dalam kategori "Lulus" atau "Tidak Lulus". Peluang ini dihitung dengan mengalikan probabilitas atribut untuk setiap kategori.
 6. Lulus > Tidak Lulus: Model membandingkan peluang "Lulus" dan "Tidak Lulus" untuk menentukan kategori akhir.
 - Jika peluang "Lulus" lebih besar dari peluang "Tidak Lulus", maka record diklasifikasikan sebagai "Lulus".
 - Jika peluang "Lulus" kurang dari atau sama dengan peluang "Tidak Lulus", maka record diklasifikasikan sebagai "Tidak Lulus".
 7. Masuk Klasifikasi Lulus: Jika peluang "Lulus" lebih besar, record masuk ke dalam kategori "Lulus".
 8. Masuk Klasifikasi Tidak Lulus: Jika peluang "Tidak Lulus" lebih besar atau sama, record masuk ke dalam kategori "Tidak Lulus".
 9. Selesai: Proses klasifikasi selesai setelah semua data uji diklasifikasikan ke dalam kategori yang sesuai.

Flowchart ini menjelaskan proses langkah demi langkah penerapan algoritma Naïve Bayes, mulai dari memasukkan data training dan data uji, menghitung probabilitas atribut, menentukan peluang kategori, hingga mengklasifikasikan data uji ke dalam kategori "Lulus" atau "Tidak Lulus".

5. Fase Evaluasi : Evaluasi ini dilakukan terhadap hasil perbandingan klasifikasi data mining dengan algoritma C4.5 dengan Naïve Bayes yang bertujuan untuk mengetahui apakah pola atau aturan yang dihasilkan telah memenuhi tujuan model bisnis yang

diterapkan, sehingga dapat menentukan apakah ada langkah yang terlewatkan atau perlu adanya kebutuhan tambahan dalam mengidentifikasi pola data sehingga dapat dilakukan perbandingan.

6. Fase Deployment : Hasil penelitian klasifikasi data mining telah disusun dalam bentuk laporan yang siap dipresentasikan kepada pihak terkait. Laporan ini akan menjelaskan perbandingan dari proses data mining yang telah dilakukan, termasuk perbandingan pola yang dihasilkan oleh Algoritma C4.5 dan Naïve Bayes.

III. HASIL DAN PEMBAHASAN

3.1 Perhitungan Manual Menggunakan Excel dengan Algoritma C4.5

Dalam penelitian ini, perhitungan manual menggunakan algoritma C4.5 dilakukan untuk memahami proses klasifikasi dan membangun pohon keputusan berdasarkan dataset yang digunakan. Dataset yang digunakan terdiri dari 3.264 record, yang kemudian dibagi menjadi data training (80%) dan data testing (20%). Atribut-atribut yang dianalisis meliputi tahun produksi, kelayakan ban, kekuatan sinar lampu utama, total gaya pengereman, dan atribut lainnya yang relevan. Berikut adalah tahapan perhitungan manual menggunakan algoritma C4.5

1. Menentukan Root Node:

Root node dalam pohon keputusan adalah titik awal dari pohon tersebut. Root node dipilih berdasarkan atribut yang memiliki nilai gain terbesar setelah perhitungan entropy. Menentukan root node adalah langkah krusial karena node ini akan mengarahkan bagaimana data akan dibagi dalam langkah-langkah selanjutnya dalam pohon keputusan. Atribut yang dihitung meliputi tahun produksi kendaraan, kepemilikan kendaraan, emisi opasitas timbal, total gaya pengereman, standarisasi suara klakson, kelayakan ban, kekuatan sinar lampu utama, penyimpangan pancar lampu utama, dan hasil pengujian. Sebagai contoh, perhitungan manual untuk atribut kelayakan ban telah dilakukan, dan berikut hasilnya ditampilkan pada Tabel 3.

NODE	ATRIBUT	VALUE	LABEL		JUMLAH KASUS	SPLIT INFO	ENTROPY	GAIN
			LULUS	TIDAK LULUS				
ROOT NODE	TOTAL		2523	88	2611	0.21		
	TAHUN PRODUKSI							
		> 10 TAHUN	1058	20	1078	0.13		
		< 10 TAHUN	1465	68	1533	0.26		
	****	****	****	****	2611		0.20	0.01
	KELAYAKAN BAN							
		ALUR BAN > 1.6MM	2523	28	2551	0.09		
		ALUR BAN < 1.6MM	0	60	60	0		
	****	****	****	****	2611		0.09	0.13
	KEKUATAN SINAR LAMPU UTAMA							
		> 12.000 cd	2523	67	2590	0.17		
		< 12.000 cd	0	21	21	0		
	****	****	****	****	2611		0.17	0.04

Tabel 3. Perhitungan Root Node

Pada Tabel 3, atribut kelayakan ban memiliki nilai gain terbesar yaitu 0.13, sehingga dipilih sebagai root node untuk pembentukan pohon keputusan. Ini menunjukkan bahwa atribut kelayakan ban paling signifikan dalam menentukan apakah sebuah kendaraan lulus atau tidak dalam pengujian.

Sementara pada atribut tahun produksi dan kekuatan sinar lampu utama memiliki nilai gain yang lebih kecil, masing-masing 0.01 dan 0.04, menunjukkan bahwa mereka kurang signifikan dibandingkan dengan Kelayakan Ban dalam proses klasifikasi.

Hasil perhitungan tersebut menunjukkan langkah-langkah nilai entropy dan gain yang digunakan untuk menentukan root node dalam pohon keputusan menggunakan algoritma C4.5. Atribut kelayakan ban dipilih sebagai root node berdasarkan nilai gain terbesar. Struktur pohon keputusan dengan root node tersebut dapat dilihat pada Gambar 5.



Gambar 5. Root node pada decision tree

Dapat dilihat pada gambar 5. Root node yang dipilih adalah atribut Kelayakan Ban karena atribut ini memiliki nilai gain terbesar, yaitu 0.13, yang menunjukkan bahwa atribut ini paling signifikan dalam menentukan apakah sebuah kendaraan akan lulus atau tidak dalam pengujian. Node ini menghasilkan dua struktur cabang:

- Alur Ban < 1.6mm. Kendaraan dengan alur ban kurang dari 1.6mm memiliki kemungkinan besar tidak lulus pengujian.
- Alur Ban > 1.6mm. Kendaraan dengan alur ban lebih dari 1.6mm memiliki kemungkinan besar lulus pengujian.

2. Perhitungan Entropy dan Gain pada Node 1

Setelah menentukan root node dengan atribut Kelayakan Ban, langkah selanjutnya adalah menghitung entropy dan gain untuk atribut-atribut berikutnya pada node pertama. Atribut yang akan dihitung selanjutnya meliputi tahun produksi, total gaya pengereman dan kekuatan sinar lampu utama. Hasil perhitungan node pertama digambarkan pada Tabel 4.

Tabel 4. Hasil Perhitungan Node 1

NODE	ATRIBUT	VALUE	LABEL		JUMLAH KASUS	SPLIT INFO	ENTROPY	GAIN
			LULUS	TIDAK LULUS				
NODE 1	KELAYAKAN BAN = ALUR BAN > 1.6MM		2523	28	2551	0.09		
	TAHUN PRODUKSI							
		> 10 TAHUN	1465	23	1488	0.12		
		< 10 TAHUN	1058	5	1063	0.04		
	****	****	****	****	2551		0.09	0.00
	TOTAL GAYA Pengereman							
		> 50%	2523	14	2537	0.05		
		< 50%	0	14	14	0		
	****	****	****	****	2551		0.05	0.04
	KEKUATAN SINAR LAMPU UTAMA							
		> 12.000 cd	2523	12	2535	0.04		
		< 12.000 cd	0	16	16	0		
	****	****	****	****	2551		0.04	0.05

Pada Tabel 4, perhitungan entropy dan gain untuk setiap atribut pada Node 1 adalah sebagai berikut:

1. Perhitungan Entropy dan Gain untuk Setiap Atribut pada Node 1:

Tahun Produksi:

Value: > 10 Tahun dan < 10 Tahun

Label (LULUS): 1465 (> 10 Tahun) dan 1058 (< 10 Tahun)

Label (TIDAK LULUS): 23 (> 10 Tahun) dan 5 (< 10 Tahun)

- Jumlah Kasus: 1488 (> 10 Tahun) dan 1063 (< 10 Tahun)

- Entropy: 0.09

- Gain: 0.00

b. Total Gaya Pengereman:

Value: > 50% dan < 50%

Label (LULUS): 2523 (> 50%) dan 0 (< 50%)

Label (TIDAK LULUS): 14 (> 50%) dan 14 (< 50%)

Jumlah Kasus: 2537 (> 50%) dan 14 (< 50%)

Entropy: 0.05

- Gain: 0.04

c. Kekuatan Sinar Lampu Utama:

Value: > 12.000 cd dan < 12.000 cd

Label (LULUS): 2523 (> 12.000 cd) dan 0 (< 12.000 cd)

Label (TIDAK LULUS): 12 (> 12.000 cd) dan 16 (< 12.000 cd)

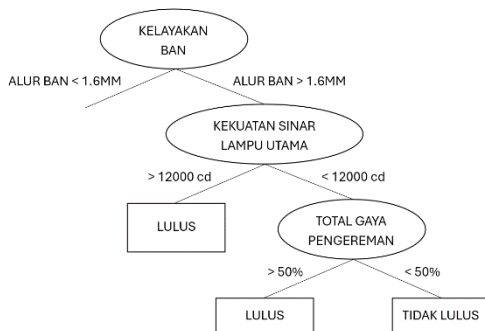
- Jumlah Kasus: 2535 (> 12.000 cd) dan 16 (< 12.000 cd)

- Entropy: 0.04
- Gain: 0.05

Pada atribut Kekuatan Sinar Lampu Utama memiliki nilai gain terbesar yaitu 0.05 pada node pertama, yang berarti atribut ini paling signifikan dalam menentukan kelayakan kendaraan setelah Kelayakan Ban. Kemudian pada atribut Total Gaya Pengereman memiliki nilai gain 0.04, menunjukkan bahwa atribut ini juga cukup signifikan namun tidak sebesar Kekuatan Sinar Lampu Utama. Sementara Tahun Produksi memiliki nilai gain 0.00, menunjukkan bahwa atribut ini kurang signifikan dalam menentukan kelayakan kendaraan pada node pertama ini. Dengan demikian, atribut Kekuatan Sinar Lampu Utama dipilih sebagai node berikutnya, proses klasifikasi akan lebih akurat dan efisien dalam mengidentifikasi kendaraan yang laik jalan nantinya.

2. Struktur Node 1

Adapun Struktur pohon keputusan selanjutnya dijabarkan pada Gambar 6.



Gambar 6. *Decision tree node 1*

Pada *decision tree node 1* berikut dari masing-masing cabang dapat dihasilkan bahwa :

1. Kelayakan Ban (Root Node):
 - Alur Ban < 1.6mm: Semua kendaraan dengan alur ban kurang dari 1.6mm dipastikan TIDAK LULUS pengujian karena kondisi ban yang tidak memenuhi standar minimum keselamatan.
 - Alur Ban > 1.6mm: Kendaraan dengan alur ban lebih dari 1.6mm akan diproses lebih lanjut untuk atribut Kekuatan Sinar Lampu Utama.
2. Kekuatan Sinar Lampu Utama:
 - > 12000 cd:

Kendaraan dengan kekuatan sinar lampu utama lebih dari 12000 cd dipastikan LULUS pengujian karena memenuhi standar pencahayaan yang ditetapkan.

< 12000 cd:

Kendaraan dengan kekuatan sinar lampu utama kurang dari 12000 cd akan diproses lebih lanjut untuk atribut Total Gaya Pengereman.

Total Gaya Pengereman (untuk kendaraan dengan Kekuatan Sinar Lampu Utama < 12000 cd):

> 50%:

Kendaraan dengan total gaya pengereman lebih dari 50% dipastikan LULUS pengujian karena memenuhi standar pengereman yang ditetapkan.

< 50%:

Kendaraan dengan total gaya pengereman kurang dari 50% dipastikan TIDAK LULUS pengujian karena tidak memenuhi standar pengereman yang ditetapkan.

Pada struktur node pertama ini kelayakan ban merupakan atribut pertama (root node) yang paling signifikan dalam menentukan kelulusan pengujian kendaraan. Kendaraan dengan alur ban kurang dari 1.6mm langsung dikategorikan sebagai TIDAK LULUS.

Kendaraan dengan alur ban lebih dari 1.6mm selanjutnya dievaluasi berdasarkan Kekuatan Sinar Lampu Utama. Jika kekuatan sinar lampu lebih dari 12000 cd, kendaraan tersebut LULUS pengujian.

Kendaraan dengan kekuatan sinar lampu utama kurang dari 12000 cd masih perlu dievaluasi lebih lanjut pada atribut Total Gaya Pengereman. Jika total gaya pengereman lebih dari 50%, kendaraan tersebut LULUS pengujian; jika kurang dari 50%, kendaraan tersebut TIDAK LULUS

3. Perhitungan Node kedua

Pada analisis struktur node pertama, yaitu untuk alur ban < 1.6 mm, terdapat 60 kasus dengan hasil 0 “LULUS” dan 60 “TIDAK LULUS”. Karena tidak ditemukan kasus “LULUS” pada node ini, maka perhitungan untuk atribut lainnya tidak diperlukan. Seluruh kasus pada node ini secara otomatis diklasifikasikan sebagai “TIDAK LULUS”.

4. Confusion Matrix

Dalam analisis ini, confusion matrix digunakan untuk mengukur akurasi model dalam memprediksi hasil pengujian kendaraan bermotor. Adapun hasil perhitungan terdapat pada tabel 5.

Tabel 5. Confusion Matrix

	PREDIKSI: LULUS	PREDIKSI: TIDAK LULUS
AKTUAL: LULUS	635	2
AKTUAL: TIDAK LULUS	0	18

Pada Tabel 5 menghasilkan hasil dari data testing dengan jumlah 653 record menunjukkan bahwa model memiliki akurasi yang sangat tinggi dalam memprediksi “Lulus” dengan 635 prediksi yang benar dan hanya 2 kasus yang salah. Model juga sangat akurat dalam memprediksi “Tidak Lulus” dengan 18 prediksi yang benar dan tidak ada prediksi yang salah sebagai “Lulus”. Berikut hasil dari penghitungan menggunakan rumus akurasi.

Akurasi

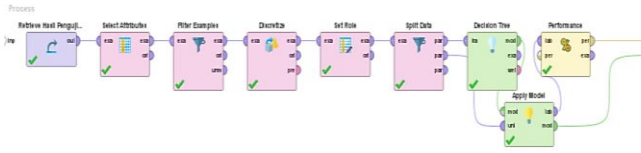
$$\begin{aligned}
 &= \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \\
 &= \frac{635 + 18}{635 + 18 + 0 + 2} \times 100\% \\
 &= \frac{653}{655} \times 100\% \\
 &= \frac{653}{655} \times 100\% \\
 &= 99.69\%
 \end{aligned}$$

3.2 Implementasi Algoritma C4.5 menggunakan RapidMiner

Dalam penelitian ini, algoritma C4.5 diimplementasikan menggunakan tools RapidMiner versi 9.10.008. Dataset yang terdiri dari 3.264 record dibagi menjadi dua bagian dengan proporsi 80:20. Tujuan dari pembagian ini adalah untuk melatih model decision tree menggunakan 80% data training, dan kemudian menguji model tersebut dengan 20% data testing untuk menilai kinerjanya. Langkah-langkah dalam proses ini meliputi:

1. Retrieve Hasil Pengujian (Retrieve)
 - Fungsi: Mengambil dataset yang berisi hasil pengujian kendaraan.
 - Input: Dataset yang akan digunakan dalam analisis.
 - Output: Data yang siap untuk diproses lebih lanjut.
2. Select Attributes
 - Fungsi: Memilih atribut yang relevan dari dataset.
 - Input: Dataset lengkap.
 - Output: Dataset dengan atribut yang dipilih untuk analisis.

3. Filter Example
 - Fungsi: Memfilter data berdasarkan kondisi tertentu.
 - Input: Dataset dengan atribut yang dipilih.
 - Output: Dataset yang telah difilter sesuai dengan kriteria yang ditentukan.
 4. Discretize
 - Fungsi: Mengubah atribut kontinu menjadi atribut diskret.
 - Input: Dataset yang telah difilter.
 - Output: Dataset dengan atribut yang telah didiskretisasi.
 5. Set Role
 - Fungsi: Menentukan peran atribut dalam model, seperti menentukan atribut target (label) dan atribut prediktor (features).
 - Input: Dataset yang telah didiskretisasi.
 - Output: Dataset dengan peran atribut yang telah ditetapkan.
 6. Split Data
 - Fungsi: Membagi dataset menjadi data training dan data testing.
 - Input: Dataset dengan peran atribut yang telah ditentukan.
 - Output: Dua subset data, satu untuk training dan satu untuk testing.
 7. Decision Tree (C4.5)
 - Fungsi: Membuat model pohon keputusan menggunakan algoritma C4.5.
 - Input: Data training.
 - Output: Model pohon keputusan yang terlatih.
 - Apply Model
 - Fungsi: Menerapkan model yang telah dilatih ke data testing untuk melakukan prediksi.
 - Input: Model pohon keputusan dan data testing.
 - Output: Hasil prediksi pada data testing.
 9. Retrieve Hasil Pengujian (Retrieve)
 - Fungsi: Mengevaluasi kinerja model dengan menghitung metrik evaluasi seperti akurasi, precision, recall, dan F1-score.
 - Input: Hasil prediksi dan data testing yang sebenarnya.
 - Output: Metrik kinerja model yang menunjukkan seberapa baik model bekerja.
- Model pengujian dapat dilihat pada gambar 7.



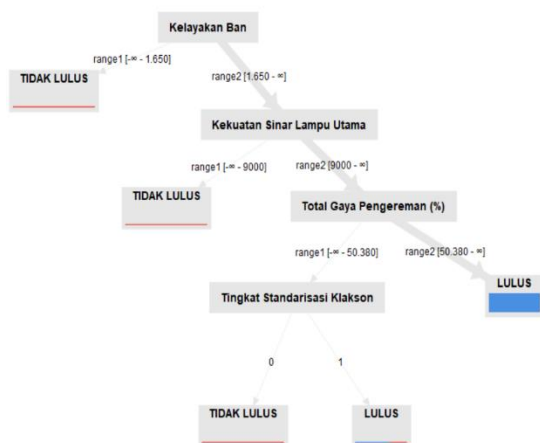
Gambar 7. Proses pemodelan menggunakan algoritma C4.5 pada RapidMiner

Pada gambar 7 dijelaskan bahwa proses ini dimulai dengan pengambilan dataset, seleksi atribut yang relevan, dan pemfilteran data. Selanjutnya, atribut secara langsung di diskretisasi dan peran tiap atribut ditentukan. Dataset kemudian dilakukan pembagian untuk dijadikan data training dan data testing sebagai pengolahan nilai prediksi. Sehingga kinerja model dapat dievaluasi untuk menentukan seberapa baik model tersebut dalam mengklasifikasikan data uji.

Dengan model langkah ini, kita dapat membangun model klasifikasi yang efektif menggunakan algoritma C4.5 dalam RapidMiner dan mengevaluasi kinerjanya secara komprehensif.

a. *Decision Tree*

Hasil pengujian algoritma C4.5 menggunakan tools RapidMiner menunjukkan pohon keputusan atau *Decision Tree* yang dibuat digunakan untuk memprediksi apakah suatu kendaraan akan "LULUS" atau "TIDAK LULUS" uji berdasarkan beberapa atribut seperti pada gambar 8.



Gambar 8. *Decision Tree* pada aplikasi RapidMiner algoritma C4.5

Pada Gambar 8 menunjukkan atribut yang paling penting dalam menentukan apakah kendaraan lulus atau tidak adalah nilai pengukuran kelayakan ban. Jika kelayakan ban memenuhi syarat, atribut berikutnya yang diperiksa adalah kekuatan sinar lampu utama, total gaya pengereman, dan terakhir

tingkat standarisasi klakson. Setiap keputusan yang dibuat berdasarkan nilai atribut pada node tersebut, sehingga mencapai keputusan akhir dari pengujian yang bernilai "LULUS" atau "TIDAK LULUS". Berikut hasil dari keterangan *Decision Tree* yang diperoleh dengan tools RapidMiner pada gambar 9.

Tree

```
Kelayakan Ban = range1 [-∞ - 1.650]: TIDAK LULUS (LULUS=0, TIDAK LULUS=56)
Kelayakan Ban = range2 [1.650 - ∞]
| Kekuatan Sinar Lampu Utama = range1 [-∞ - 9000]: TIDAK LULUS (LULUS=0, TIDAK LULUS=17)
| Kekuatan Sinar Lampu Utama = range2 [9000 - ∞]
| | Total Gaya Pengereman (%) = range1 [-∞ - 50.380]
| | | Tingkat Standarisasi Klakson = 0: TIDAK LULUS (LULUS=0, TIDAK LULUS=6)
| | | Tingkat Standarisasi Klakson = 1: LULUS (LULUS=2, TIDAK LULUS=1)
| | Total Gaya Pengereman (%) = range2 [50.380 - ∞]: LULUS (LULUS=2524, TIDAK LULUS=5)
```

Gambar 9. Keterangan *Decision Tree*

Hasil Pengukuran Akurasi

Hasil pengukuran akurasi menggunakan tools RapidMiner dengan data testing sebanyak 20% dari total record, serta confusion matrix, dapat dilihat pada Gambar 10.

accuracy: 99.69%			
	true LULUS	true TIDAK LULUS	class precision
pred LULUS	632	2	99.68%
pred TIDAK LULUS	0	19	100.00%
class recall	100.00%	90.48%	

Gambar 10. Hasil akurasi algoritma C4.5 dengan confusion matrix

Pada gambar 10. Model yang dinilai memiliki akurasi yang sangat tinggi (99.69%), menunjukkan bahwa model ini sangat efektif dalam memprediksi hasil uji kendaraan. Selain itu tingkat presisi untuk kelas "LULUS" juga bernilai sangat baik (99.68%) dan untuk kelas "TIDAK LULUS" bernilai sempurna (100%), hal ini menunjukkan bahwa model sangat jarang melakukan suatu kesalahan. Begitu pula pada class recall untuk kelas "LULUS" adalah 100.00%, yang berarti model mendeteksi semua kasus "LULUS" dengan benar. Recall untuk kelas "TIDAK LULUS" adalah 90.48%, menunjukkan bahwa ada beberapa kasus "TIDAK LULUS" yang tidak terdeteksi dengan benar. Dengan hasil ini, model algoritma C4.5 yang digunakan sangat efektif dan akurat dalam memprediksi hasil uji kendaraan, baik untuk kelas "LULUS" maupun "TIDAK LULUS".

3.3 Perhitungan Manual Menggunakan Excel dengan Algoritma Naïve Bayes

Pada perhitungan algoritma Naïve Bayes, dataset yang digunakan tetap menggunakan 3.264 record dimana untuk pembagian data training disini menggunakan 2.000 record dan untuk data test sebanyak 1.264 record. Berikut adalah langkah-

langkah untuk melakukan perhitungan manual menggunakan Excel dengan algoritma Naïve Bayes.

1. Perhitungan Probabilitas Prior (P(Class))

Dalam Probabilitas Prior (P(Class)) diperlukan penghitungan jumlah total instance untuk setiap kelas dalam data training. Setelah itu probabilitas prior untuk setiap kelas dapat dihitung dengan membagi jumlah instance dari masing-masing kelas dengan jumlah total instance dalam data training dengan hasil pada Tabel 6.

Tabel 6. Probabilitas Prior

Kelas	Jumlah	Probabilitas Prior
Lulus	1926	0.963
Tidak Lulus	74	0.037

2. Perhitungan Probabilitas Likelihood (P(Attribute|Class))

Pada penghitungan probabilitas Likelihood (P(Attribute|Class)) di setiap atributnya perlu menghitung jumlah instance yang terjadi pada setiap kelas dalam data training. Setelah itu probabilitas likelihood untuk setiap atribut dalam setiap kelas dapat dihitung dengan membagi jumlah instance dari atribut tersebut dalam kelas dengan jumlah total instance dalam kelas seperti pada Tabel 7 berikut.

Tabel 7. Probabilitas Likelihood (P(Attribute|Class))

Atribut	Nilai	Kelas	Jumlah	Probabilitas Likelihood
Kelayakan Ban	> 1.6mm	LULUS	1926	1.00
Kelayakan Ban	< 1.6mm	LULUS	0	0.00
Kelayakan Ban	> 1.6mm	TIDAK LULUS	26	0.35
Kelayakan Ban	< 1.6mm	TIDAK LULUS	48	0.65
Total Gaya Pengereman	> 50%	LULUS	1926	1.00
Total Gaya Pengereman	< 50%	LULUS	0	0.00
Total Gaya Pengereman	> 50%	TIDAK LULUS	0	0.53
Total Gaya Pengereman	< 50%	TIDAK LULUS	74	0.47
Kekuatan Sinar Lampu	> 12.000 cd	LULUS	1926	1.00
Kekuatan Sinar Lampu	< 12.000 cd	LULUS	0	0.00
Kekuatan Sinar Lampu	> 12.000 cd	TIDAK LULUS	58	0.78
Kekuatan Sinar Lampu	< 12.000 cd	TIDAK LULUS	16	0.22

Pada Tabel 7 memperlihatkan bahwa Probabilitas likelihood memberikan informasi tentang seberapa sering suatu nilai atribut muncul dalam masing-masing kelas. Atribut "Alur Ban", "Total gaya pengereman", dan "Kekuatan Pancar" memiliki nilai yang sangat mempengaruhi klasifikasi "LULUS" dan "TIDAK LULUS". Probabilitas likelihood ini digunakan dalam perhitungan posterior dalam algoritma Naïve Bayes untuk memprediksi kelas suatu instance berdasarkan nilai-nilai atributnya.

3. Perhitungan Probabilitas Posterior (P(Class|Attributes))

Pada perhitungan Probabilitas posterior (P(Class|Attributes)) dalam algoritma Naïve Bayes dihitung dengan mengalikan probabilitas prior (P(Class)) dengan probabilitas likelihood (P(Attribute | Class)) dalam setiap atribut untuk kemudian menormalkan hasilnya seperti pada Tabel 8.

Tabel 8. Probabilitas Posterior (P(Class|Attributes))

Kelas	Probabilitas Prior	P(> 1.6mm Class)	P(> 50% Class)	P(> 12.000 cd Class)	Probabilitas Posterior (tidak ternormalisasi)	Probabilitas Posterior (Ternormalisasi)
LULUS	0.963	1.00	1.00	1.00	0.96	0.99
TIDAK LULUS	0.037	0.35	0.53	0.78	0.01	0.01

Dari probabilitas posterior tersebut dapat disimpulkan bahwa untuk memprediksi kelas dengan instance baru menggunakan atribut "Kelayakan Ban > 1.6mm", "Total gaya pengereman > 50%", dan "Kekuatan sinar lampu utama > 12.000 cd" diperoleh hasil probabilitas berikut :

Probabilitas Posterior untuk kelas "LULUS":

$$P(LULUS|Attributes) \propto P(LULUS) \times P(> 1.6 mm|LULUS) \times P(> 50%|LULUS) \times P(> 12.000cd|LULUS)$$

$$P(LULUS|Attributes) \propto 0.973 \times 1.00 \times 1.00 \times 1.00 = 0.973$$

b. Probabilitas Posterior untuk kelas "TIDAK LULUS":

$$P(TIDAK LULUS|Attributes) \propto P(TIDAK LULUS) \times P(> 1.6 mm | TIDAK LULUS) \times P(> 50% | TIDAK LULUS) \times P(> 12.000cd | TIDAK LULUS)$$

$$P(TIDAK LULUS|Attributes) \propto 0.027 \times 0.35 \times 0.53 \times 0.78 = 0.0039$$

c. Normalisasi Probabilitas Posterior

Untuk mendapatkan nilai total probabilitas posterior yang ternormalisasi adalah sebagai berikut :

1. Perhitungan Total Probabilitas Posterior :

$$Total = 0.973 + 0.0039 = 0.9769$$

2. Normalisasi probabilitas tiap kelas :

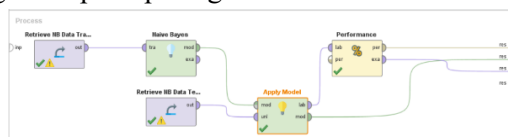
$$P(LULUS | Attributes) = \frac{0.973}{0.9769} = 0.9960$$

$$P(TIDAK LULUS | Attributes) = \frac{0.0039}{0.9769} = 0.0040$$

Dari perhitungan di atas, probabilitas posterior yang ternormalisasi menunjukkan bahwa instance baru dengan atribut Kelayakan Ban > 1.6 mm, Total gaya pengereman > 50%, dan Kekuatan sinar lampu utama > 12.000 cd lebih mungkin termasuk dalam kelas "LULUS" dengan probabilitas sekitar 99.60%, dibandingkan dengan kelas "TIDAK LULUS" yang hanya memiliki probabilitas sekitar 0.40%.

3.4 Implementasi Algoritma Naïve Bayes Menggunakan RapidMiner

Implementasi algoritma Naïve Bayes diaplikasikan dengan tools RapidMiner versi 9.10.008 dengan menggunakan split data. Adapun pembagian data tersebut sebesar 2.000 record data training dan 1.264 data testing. Adapun proses pemodelan menggunakan algoritma Naïve Bayes dalam suatu alat analisis data yang diterapkan pada gambar 11.

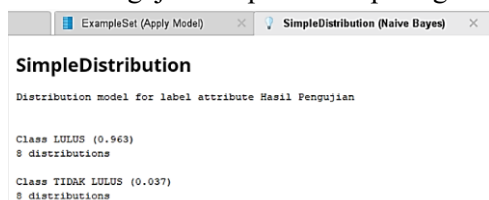


Gambar 11. Proses pemodelan menggunakan algoritma Naïve Bayes pada RapidMiner

Pada gambar 11. Proses yang ditampilkan adalah proses standar dalam membangun, menerapkan, dan mengevaluasi model klasifikasi menggunakan algoritma Naïve Bayes. Model ini pertama-tama dilatih dengan data training, kemudian diterapkan pada data testing, dan akhirnya dievaluasi menggunakan metrik kinerja untuk menentukan efektivitas model dalam melakukan prediksi.

a. Simple Distribution

Representasi dari distribusi probabilitas untuk kelas yang diprediksi oleh model Naïve Bayes berdasarkan data training. Hasil model distribusi sederhana yang dihasilkan oleh algoritma Naïve Bayes untuk atribut label "Hasil Pengujian" dapat dilihat pada gambar 12.



Gambar 12. Simple Distribution model algoritma Naïve Bayes

Hasil Simple Distribution dari model Naïve Bayes memberikan informasi tentang bagaimana model

menghitung probabilitas prior untuk setiap kelas berdasarkan data training. Dengan probabilitas prior yang lebih tinggi untuk kelas "LULUS" (0.963), model akan cenderung memprediksi "LULUS" kecuali ada bukti kuat dari atribut yang menunjukkan sebaliknya. Probabilitas prior ini dikombinasikan dengan distribusi atribut untuk membuat prediksi akhir menggunakan algoritma Naïve Bayes. "8 distributions" menunjukkan bahwa model Naïve Bayes menggunakan 8 atribut (fitur) untuk menentukan distribusi probabilitas. Setiap atribut akan memiliki distribusi probabilitas tersendiri yang digunakan untuk menghitung likelihood (kemungkinan) bahwa data baru termasuk dalam salah satu kelas. Distribusi ini digunakan dalam rumus Naïve Bayes untuk menghitung probabilitas posterior $P(\text{Class} | \text{Attributes})$, yang kemudian digunakan untuk membuat prediksi

b. Hasil pengukuran akurasi

Hasil evaluasi model Naïve Bayes berdasarkan confusion matrix dan metrik kinerja lainnya dapat dilihat pada gambar 13.

accuracy: 99.13%			
	true LULUS	true TIDAK LULUS	class precision
pred. LULUS	1225	5	99.59%
pred. TIDAK LULUS	5	27	81.82%
class recall	99.51%	84.38%	

Gambar 13. Hasil akurasi algoritma Naïve Bayes dengan confusion matrix

Model Naïve Bayes memiliki akurasi yang sangat tinggi (99.13%) sehingga menunjukkan bahwa model ini sangat efektif dalam memprediksi hasil uji kendaraan. Pada tingkat precision untuk kelas "LULUS" adalah 81.82%, menunjukkan bahwa model jarang melakukan kesalahan positif pada kelas "LULUS", tetapi masih ada beberapa kesalahan positif pada kelas "TIDAK LULUS". Recall pada kelas "LULUS" bernilai 99.51%, yang berarti model mendeteksi hampir semua kasus "LULUS" dengan benar. Recall untuk kelas "TIDAK LULUS" adalah 84.38%, menunjukkan bahwa ada beberapa kasus "TIDAK LULUS" yang tidak terdeteksi dengan benar. Dengan hasil ini, model Naïve Bayes yang digunakan sangat efektif dan akurat dalam memprediksi hasil uji kendaraan, khususnya dalam mendeteksi kendaraan yang "LULUS". Namun, ada ruang untuk perbaikan dalam mendeteksi kendaraan

yang "TIDAK LULUS", meskipun hasilnya sudah cukup baik.

IV. KESIMPULAN DAN SARAN

4.1 KESIMPULAN

Berdasarkan data hasil perbandingan pada model pengujian kendaraan bermotor menggunakan algoritma C4.5 dan Naïve Bayes dapat disimpulkan bahwa :

1. Tingkat Akurasi

Pada algoritma C4.5 menunjukkan bahwa akurasi yang sangat tinggi dalam memprediksi hasil pengujian kendaraan bermotor, dengan akurasi sebesar 99.69%. Sedangkan pada algoritma Naïve Bayes sedikit lebih rendah yang bernilai 99.13%.

2. Precision dan Recall

Untuk kelas "LULUS" pada algoritma C4.5 memiliki precision sebesar 99.68% dan recall sempurna 100%. Sedangkan pada algoritma Naïve Bayes memiliki precision sebesar 81.82% dan recall sebesar 84.38%.

3. Kemampuan Deteksi

Algoritma C4.5 lebih unggul dalam mendeteksi kendaraan yang "TIDAK LULUS" dibandingkan dengan Naïve Bayes, hal ini terlihat pada precision dan recall yang lebih tinggi pada kelas "TIDAK LULUS". Sedangkan Naïve Bayes memiliki performa yang sangat baik dalam mendeteksi kendaraan yang "LULUS", namun sedikit lebih rendah dalam mendeteksi kendaraan yang "TIDAK LULUS" dibandingkan dengan C4.5.

4. Kecepatan Komputasi dan Kompleksitas Model

Untuk algoritma Naïve Bayes umumnya lebih cepat dalam komputasi dikarenakan asumsi independensi antar atribut, yang membuatnya lebih efisien dalam menghitung probabilitas. Kemudian untuk algoritma C4.5, meskipun sedikit lebih kompleks dan lebih memerlukan waktu komputasi yang lebih lama, algoritma ini dapat memberikan pohon keputusan yang dapat diinterpretasikan dengan mudah dan memberikan wawasan yang lebih jelas mengenai aturan-aturan klasifikasi.

5. Penggunaan Atribut

Algoritma C4.5 mampu menghasilkan pohon keputusan yang mengidentifikasi atribut-atribut yang paling berpengaruh dalam klasifikasi, seperti

kelayakan ban, kekuatan sinar lampu utama, dan total gaya pengereman. Sementara pada algoritma Naïve Bayes lebih kepada penghitungan probabilitas berdasarkan distribusi atribut, yang juga menunjukkan pentingnya atribut-atribut seperti kelayakan ban dan total gaya pengereman.

4.2 SARAN

Dari hasil perbandingan ini, pemilihan algoritma dapat disesuaikan dengan kebutuhan spesifik dari aplikasi pengujian kendaraan bermotor. Algoritma C4.5 lebih cocok digunakan jika interpretabilitas dan deteksi kendaraan yang "TIDAK LULUS" sangat penting, karena mampu memberikan wawasan yang jelas melalui pohon keputusan yang dihasilkan. Sebaliknya, jika kecepatan komputasi menjadi prioritas utama, algoritma Naïve Bayes lebih unggul karena efisiensinya dalam menghitung probabilitas. Peneliti dan praktisi disarankan untuk mempertimbangkan karakteristik dataset dan tujuan analisis dalam memilih algoritma yang paling sesuai untuk aplikasi pengujian kendaraan bermotor ini

REFERENSI

- [1] P. R. Strategis and P. K. Situbondo, "Perubahan Rencana Strategis 2021 – 2026 Dinas Perhubungan Kabupaten Situbondo," 2024.
- [2] R. Kerja, "Rencana Kerja Dinas Perhubungan Kabupaten Situbondo 2024," 2024.
- [3] E. Fitriani, "Perbandingan Algoritma C4.5 Dan Naïve Bayes Untuk Menentukan Kelayakan Penerima Bantuan Program Keluarga Harapan," *Jurnal Sistem Informasi*, Volume 9 No 1, pp. 103 – 115, 2020.
- [4] A. Irfan, "Implementasi Data Mining dalam Pemeliharaan Jalan," 2021.
- [5] S. Hendrian, "Algoritma Klasifikasi Data Mining Untuk Memprediksi Siswa Dalam

- Memperoleh Bantuan Dana Pendidikan,” *Faktor Exacta*, vol. 11, no. 3, Oct. 2018.
- [6] M. A. Puspa, “Implementasi Data Mining Klasifikasi Algoritma C4.5 Dalam Perekrutan Perangkat Desa,” *Technology and Science (BITS)*, vol. 1, no. 2, 2019.
- [7] E. Apriliyani and Y. Salim, “Analisis performa metode klasifikasi Naïve Bayes Classifier pada Unbalanced Dataset,” *Indonesian Journal of Data and Science (IJODAS)*, vol. 3, no. 2, pp. 47–54, 2022.
- [8] R. Reza Andarista and A. Jananto, “Penerapan Data Mining Algoritma C4.5 Untuk Klasifikasi Hasil Pengujian Kendaraan Bermotor,” vol. 16, no. 2 pp. 29-43, 2021.
- [9] D. A. Munawwaroh and A. H. Primandari, “Implementasi Crisp-Dm Model Menggunakan Metode Decision Tree Dengan Algoritma Cart Untuk Prediksi Lila Ibu Hamil Berpotensi Gizi Kurang,” *Jurnal Ilmiah Pendidikan Matematika* Vol. 10, No. 2, pp. 367 – 380, 2022.