IDENTIFIKASI JENIS KECELAKAAN LALU LINTAS DI KOTA SAMARINDA MENGGUNAKAN METODE *RANDOM FOREST*

Nanda Betris Dea Maretta, Fendy Yulianto, Taghfirul Azhima Yoga Siswa
Teknik Informatika, Universitas Muhammadiyah Kalimantan Timur
Jln.Ir H Juanda, Samarinda 75124, Indonesia
2111102441103@umkt.ac.id

ABSTRAK

Kecelakaan lalu lintas di Kota Samarinda merupakan permasalahan yang berdampak signifikan, baik dari segi kerugian material, cedera, maupun korban jiwa. Oleh karena itu, mengklasifikasikan tingkat kecelakaan menjadi langkah penting dalam mengidentifikasi pola kejadian serta mendukung upaya pencegahan dan penanganan yang lebih optimal. Penelitian ini bertujuan untuk mengelompokkan tingkat kecelakaan di Kota Samarinda dengan menerapkan metode *Random Forest*. Data yang digunakan terdiri dari 1.004 kasus kecelakaan yang diperoleh dari Kepolisian Sektor Samarinda Kota dalam rentang waktu 2021 hingga 2024. Data tersebut diproses melalui tahap *Pre-Processing* dan dianalisis menggunakan *K-Fold Cross-Validation* guna memastikan kualitas serta akurasi model. Metode *Random Forest* bekerja dengan membangun sejumlah pohon keputusan dan menentukan hasil akhir berdasarkan voting mayoritas, sehingga efektif dalam mengklasifikasikan data yang kompleks seperti tingkat kecelakaan. Hasil penelitian menunjukkan bahwa metode ini mampu mengelompokkan kecelakaan ke dalam tiga kategori, yaitu ringan, sedang, dan berat, dengan nilai *F1-Score* mencapai 95%. Dengan demikian, penelitian ini dapat menjadi landasan dalam merumuskan kebijakan keselamatan lalu lintas yang lebih berbasis data serta membantu pihak berwenang dalam mengurangi angka kecelakaan dan meningkatkan kesadaran masyarakat terhadap keselamatan berkendara.

Kata kunci : Kecelakaan Lalu Lintas, *K-Fold Cross- Validation,* Klasifikasi, *Random Forest, F1-Score*

1. PENDAHULUAN

Kecelakaan lalu lintas adalah peristiwa tak terduga antara kendaraan dan penggunanya, terkadang menimbulkan kematian maupun kerusakan harta benda yang disebabkan oleh pelanggaran, kondisi jalan, kendaraan, cuaca buruk, berkurangnya jarak pandang semuanya dapat berujung pada kecelakaan lalu lintas [1].

Kecelakaan lalu lintas merupakan salah satu permasalahan yang sering terjadi di jalan dan harus dihindari sebagai bagian dari manajemen jalan karena berpotensi membahayakan keselamatan pengguna jalan [2].

Keselamatan pengguna jalan merupakan hal penting, namun sering kali kecelakaan terjadi karena adanya pelanggaran tata tertib oleh pengendara yang membahayakan pengguna jalan lainnya, sehingga diperlukan kedisiplinan dalam berlalu lintas, terutama bagi pengguna kendaraan bermotor [3].

Penggunaan kendaraan bermotor di Indonesia membutuhkan edukasi yang mencakup peraturan lalu lintas, standar kendaraan, serta pengaturan perilaku pengemudi melalui implementasi tindakan penegakan hukum [4].

Implementasi penegakan hukum telah diterapkan seperti memberikan sanksi kepada pelanggar, meningkatkan patroli, mengadopsi E-Tilang untuk pemanfaatan teknologi digital proses tilang, maupun memberikan edukasi tentang disiplin berlalu lintas, namun tingkat kecelakaan masih tinggi [5].

Tingkat kecelakaan yang tinggi dapat diatasi dengan memanfaatkan informasi sebagai acuan kebijakan keselamatan lalu lintas melalui penerapan konsep prediksi, klastering, serta klasifikasi [6].

Klasifikasi merupakan teknik *Data Mining* yang menggunakan data pelatihan guna membuat model dan data pengujian untuk menilai kemampuan model dalam mengklasifikasi kategori baru secara akurat maupun efesien, serta menunjukkan prediksi dapat dipercaya pada data yang belum diketahui sebelumnya [7].

Mengklasifikasi kategori baru berdasarkan kemampuan model dalam menggunakan konsep klasifikasi memerlukan penggunaan metode seperti *K-Nearest Neighbors, Random Forest, Decision Tree,* dan *Classification and Regression Tree (CART)* [8].

Penelitian yang dilakukan oleh [9] tentang analisis kecelakaan lalu lintas didapatkan hasil akurasi tertinggi metode *Random Forest* sebesar 72% dibandingkan dengan metode *K-Nearset Neighbor* sebesar 66%.

Penelitian lain dilakukan oleh [10] tentang analisis tingkat keparahan kecelakaan lalu lintas menggunakan tiga metode pemodelan didapatkan hasil akurasi tertinggi metode *Random Forest* sebesar 73,38% dibandingkan dengan metode *Logistic Regression* sebesar 73,07% dan metode *Classification and Regression Tree* (*CART*) sebesar 72,65%.

Metode yang akan digunakan pada penelitian ini ialah metode *Random Forest*, karena kemampuannya dalam menangani data kompleks menggunakan teknik *agregasi bootstrap* (*bagging*), dengan peningkatan akurasi seiringnya bertambahnya jumlah pohon dan klasifikasi bergantung pada hasil pemungutan suara dari bentuk pohon saat diterapkan [11].

Menurut penelitian yang dilakukan [12] membuktikan bahwa *Random Forest* menghasilkan akurasi yang lebih optimal dengan nilai akurasi sebesar 75,5% dibandingkan metode *Naive Bayes* sebesar 73,1% dan *Logistic Regression* sebesar 74,*5%*, menujukkan metode *Random Forest* cocok digunakan dalam memprediksi tingkah keperahan kecelakaan lalu lintas.

Pada penelitian sebelumnya yang dilakukan oleh [9] telah membandingkan metode *Random Forest* dan *K-Nearest Neighbor* untuk analisis risiko kecelakaan, namun akurasi yang diperoleh masih terbatas karena atribut yang digunakan hanya mencakup jenis kelamin, usia, waktu kecelakaan, bulan dan cuaca, fatalitas dan data numerik. Penelitian ini memiliki perbedaan dengan peneliti sebelumnya karena menerapkan pendekatan yang lebih mendalam dengan memanfaatkan berbagai atribut seperti kondisi cahaya, cuaca, kelas jalan, tipe jalan, kondisi permukaan jalan, batas kecepatan jalan, status jalan dan kemiringan jalan, bertujuan memahami pola kecelakaan untuk mendukung kebijakan keselamatan dan pencegahan yang tepat [13].

Dalam penelitian ini, *Confusion Matrix* digunakan untuk menilai kinerja suatu model klasifikasi, selain itu *Confusion Matrix* dapat memberikan nilai evaluasi terhadap model untuk performa seperti akurasi, *Precision*, *Recall*, dan *F1-Score* serta menyajikan informasi prediksi pada setiap kelas [14].

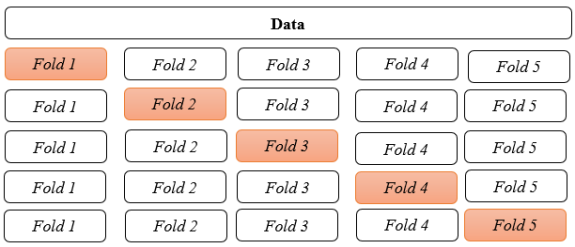
Confusion Matrix dipilih karena mampu memberikan nilai evaluasi menyeluruh, membantu menentukan kesalahan prediksi, menangani masalah ketidakseimbangan data, dan berfungsi sebagai standar umum dalam penelitian, sehingga hasil evaluasi lebih mudah dibandingkan dan lebih dapat dipercaya [15].

2. TINJAUAN PUSTAKA

2.1. Klasifikasi

Klasifikasi adalah proses mengelompokkan data atau objek ke dalam kategori tertentu berdasarkan karakteristik yang telah ditentukan sebelumnya, menggunakan metode tertentu, dengan tujuan memisahkan data secara jelas dan menentukan kriteria digunakan untuk mengklasifikasikan data baru [16].

2.2. *K-Fold Cross-Validation*



Gambar 1. K-Fold Cross-Validation

K-Fold Cross-Validation merupakan metode statistik untuk menguji kinerja model dengan membagi data menjadi data pelatihan dan validasi, mengulanginya sebanyak lima kali guna mengevaluasi kemampuan model dalam menggeneralisasi data [17], dapat dilihat pada Gambar 1.

Pada Gambar 1 menunjukkan pengujian 5-*Fold Cross-Validation*, di mana data dibagi menjadi 5 bagian, 4 untuk pelatihan dan 1 untuk validasi. Rata-rata hasil 5 iterasi digunakan untuk menilai kinerja model, memberikan keseimbangan antara akurasi, efisiensi komputasi, dan mencegah *overfitting*.

2.3. *Binary Class*

Binary Class merupakan teknik klasifikasi dengan membagi data ke dalam dua kelas seperti kelas 0 dan 1, proses ini lebih sederhana dan efektif dalam menangani masalah ketidak seimbangan data. Pendekatakan ini digunakan sebagai langkah awal untuk memahami pola data sebelum melanjutkan klasifikasi yang lebih mendalam [18].

2.4. *Multi-Class*

Multi-Class merupakan konsep klasifikasi yang memiliki lebih dari dua kelas, dengan beberapa pendekatan seperti *One-vs-One* dan *One-vs-All* dapat membantu dalam proses klasifikasi [19].

2.5. *One-vs-One*

One-vs-One mengklasifikasikan data dengan membandingkan setiap pasangan kelas secara berpasangan dalam data, yang menghasilkan beberapa klasifikasi biner [20].

2.6. *One-vs-All*

One-vs-All mengklasifikasikan data dengan membandingkan setiap kelas terhadap gabungan semua kelas lainnya dalam data, kemudian diulang untuk setiap kelas yang ada [21]**.**

2.7. *Multi-Label*

Multi-Label adalah teknik klasifikasi yang memungkinkan data memiliki lebih dari satu label atau kelas dengan mengizinkan penugasan beberapa label berbeda pada setiap sampel data, memfasilitasi prediksi beberapa kelas dengan sesuai [22].

2.8. *Random Forest*

Random Forest merupakan pengembangan dari *Classification and Regression Tree (CART),* dimana metode *Random Forest* menggunakan teknik *agregasi bootstrap (bagging)* dalam memilih fitur secara acak untuk membuat banyak pohon klasifikasi. Pemilihan fitur secara acak untuk meningkatkan metode *Random Forest* membangkitkan simpul anak secara acak pada setiap titik keputusan dalam pohon [23]. Berikut merupakan perhitungan *Entropy* dari *Random Forest* pada Persamaan 1.

$$Entropy\ (Y) = -\sum_i p(c|Y) log^2 p(c|Y) \quad (1)$$

Keterangan:
Y = Himpunan Kasus
P(c|Y) = Proporsi nilai Y terhadap kelas c.

Kemudian menghitung *Information Gain* (Y,a), untuk menentukan atribut yang paling berpengaruh dalam proses klasifikasi menggunakan Persamaan 2.

$$Gain\ (Y) = -\sum_v evalues(a) \frac{|Y_v|}{|Y_a|} Entropy\ (Y_V\ ) \quad (2)$$

Keterangan:

Values(a) = Nilai yang mungkin dalam himpunan kasus a.

Y_v = Subkelas dari Y dengan kelas v yang berhubungan dengan kelas a.

Y_a = Semua nilai yang sesuai dengan a.

2.9. Evaluasi

Evaluasi mengukur kinerja model dengan membandingkan hasilnya dengan data asli. *Confusion Matrix* digunakan untuk menilai akurasi dan kinerja klasifikasi pada data uji [24].

2.10. *Precision*

Precision merupakan matrik evaluasi dalam statistika yang digunakan untuk mengukur nilai suatu model maupun sistem secara akurat [25]. Berikut perhitungan nilai *Precision* dapat dilihat pada Persamaan 3.

$$Precision = \frac{TP}{TP+FP} \times 100\% \quad (3)$$

Keterangan:

a. *True Positive* (TP) = Jumlah sampel positif yang diprediksi dengan benar oleh model.
b. *False Positive* (FP) = Jumlah sampel negatif yang salah diprediksi sebagai positif oleh model.

2.11. *Recall*

Recall merupakan perbandingan jumlah dokumen yang berhasil diambil oleh sistem pencarian, kemudian nilai *Recall* menampilkan untuk menghitung nilai positif dari metode-metode yang digunakan [26]. Berikut perhitungan nilai *Recall* dapat dilihat pada Persamaan 4.

$$Recall = \frac{TP}{TP+FN} \times 100\% \quad (4)$$

Keterangan:

a. *True Positive* (TP) = Jumlah sampel positif yang diprediksi dengan benar oleh model.
b. *False Negative* (FN) = Jumlah sampel positif yang salah diprediksi sebagai negatif oleh model.

2.12. *F1-Score*

F1-Score adalah perhitungan matrik untuk menilai keseimbangan antara *Precision* dan *Recall* dengan menghitung rata-rata dalam menentukan kinerja model klasifikasi [27]. Berikut perhitungan nilai *F1-Score* dapat dilihat pada Persamaan 5.

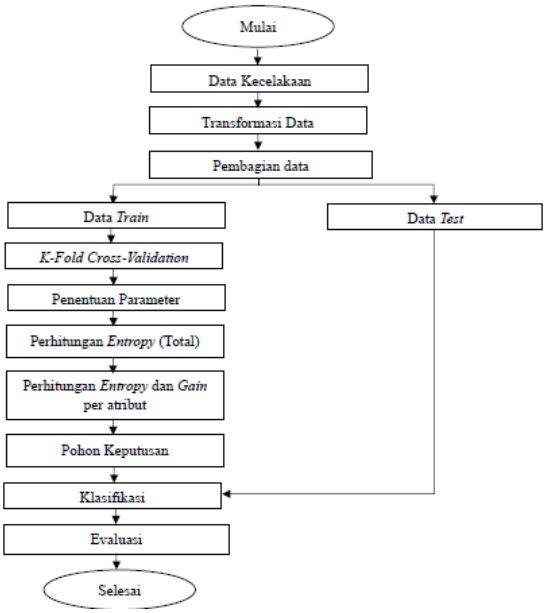
$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \times 100\% \quad (5)$$

Keterangan:

a. *Precision* = Mengukur seberapa akurat prediksi positif model.
b. *Recall* = Mengukur seberapa baik model menangkap semua data yang relevan.

3. METODE PENELITIAN

Alur metode adalah rangkaian langkah terstruktur yang digunakan untuk menerapkan teknik atau pendekatan dalam mencapai tujuan penelitian dengan memastikan bahwa proses dilakukan secara teratur dan terencana. Berikut diagram alur metode *Random Forest* yang digunakan dalam penelitian klasifikasi kecelakaan lalu lintas dapat dilihat pada Gambar 2.



Gambar 2. Alur Metode

Berdasarkan pada Gambar 2 alur metode dalam penelitian ini terdiri dari beberapa tahapan yang dilakukan dengan urutan sebagai berikut:

a. Data Kecelakaan
   Data kecelakaan lalu lintas dari Polsek Samarinda (2021–2024) telah divalidasi dan diproses, mencakup validasi, pemilihan atribut, serta pra-pemrosesan, untuk melatih dan menguji model *Random Forest* dalam klasifikasi tingkat kecelakaan.
b. Transformasi Data
   Transformasi data dalam penelitian ini mencakup beberapa langkah yaitu:
   - Pemilihan fitur dalam transformasi data mencakup seleksi untuk memilih atribut relevan dan menghapus yang tidak perlu, serta ekstraksi untuk menentukan fitur baru yang lebih relevan.
   - *Missing value* ditangani dengan memeriksa, menghapus, atau mengganti nilai yang hilang dalam transformasi data.
   - Pendeteksian dan perbaikan kesalahan memastikan kualitas data dengan mengidentifikasi dan memperbaiki ketidaktepatan, seperti salah ketik atau inkonsistensi.
   - Transformasi data mengubah kategori ke numerik dengan *One-Hot Encoding* dan menyesuaikan ukuran data untuk meningkatkan konsistensi dan efektivitas model.

c. Pembagian Data
Data awal dibagi menjadi dua bagian utama yaitu data pelatihan (*Train*) 80% untuk menyusun model dan data pengujian (*Test*) 20% bertujuan mengukur performa model tersebut bertujuan untuk mengevaluasi performa model [28].

d. *K-Fold Cross-Validation*
Proses ini membagi data pelatihan (*Train*) menjadi beberapa lipatan (*Fold*) untuk validasi silang, tujuannya untuk mengevaluasi kinerja model terhadap subset data sehingga dapat mengurangi risiko *Overfitting*, pada proses ini menggunakan K = 5 [29].

e. Penentuan Parameter
Parameter model yang mempengaruhi kinerja model seperti *n_estimator*, *max_depth*, *min_samples_split* dan *min_samples_leaf* [28].

f. Perhitungan *Entropy* (Total)
Entropy (Total) digunakan untuk mengukur ketidakpastian atau keragaman sebelum data dipisahkan berdasarkan atribut, semakin tinggi nilai *Entropy* maka semakin besar ketidakpastian data.

g. Perhitungan *Entropy* dan *Gain* per atribut
Selanjutnya menentukan atribut terbaik dalam proses pembagian data, dinilai dengan menggunakan *Entropy* untuk mengukur ketidakpastian dalam data dan *Gain* menentukan seberapa besar atribut tersebut mampu mengurangi ketidakpastian, proses ini membantu memilih atribut yang memberikan pembagian data terbaik dalam klasifikasi.

h. Pohon Keputusan
Pohon keputusan dibangun dengan menentukan atribut yang memiliki nilai *Gain* tertinggi dipilih sebagai *node*, proses ini diulang hingga setiap cabang selesai lalu digunakan untuk membuat keputusan berdasarkan data yang diberikan.

i. Klasifikasi
Klasifikasi merupakan proses pengelompokkan data kedalam kategori berdasarkan atribut tertentu, untuk mengidentifikasi tingkat kecelakaan lalu lintas yang terbagi menjadi 3 yaitu ringan, sedang, serta berat.

j. Evaluasi
Evaluasi model *Random Forest* dilakukan menggunakan *Confusion Matrix* dengan metrik *Precision*, *Recall*, dan *F1-Score* untuk menilai performanya secara keseluruhan.

4. HASIL DAN PEMBAHASAN

4.1. Transformasi Data

Proses transformasi data mengubah kategori menjadi label numerik (0, 1, atau 2) untuk pemrosesan oleh model *Random Forest*, meningkatkan efisiensi dan akurasi prediksi. Hasil pra-pemrosesan dapat dilihat pada Tabel 1.

Tabel 1. Transformasi Data

Atribut	**Spesifikasi**	**Label**
Kondisi Cahaya	Terang	0
	Gelap	1
Cuaca	Cerah	0
	Hujan	1
	Hujan disertai Angin Kencang	1
	Angin Kencang	1
	Berawan	0
Kondisi Permukaan Jalan	Berombak	1
	Baik	0
	Licin	1
	Berlubang	1
	Basah	1
Kelas Jalan	I (Jalan Besar utk beban 10 ton & max 18 m Panjang Ran)	1
	II (Jalan Sedang utk beban s/d 10 ton & 12m Panjang Ran)	1
	III (Jalan Kecil utk max beban 8 ton & 9 m Panjang Ran)	0
Kemiringan Jalan	Datar	0
	Menanjak	1
Tipe Jalan	1/1 (1 Lajur/1 Arah)	0
Atribut	**Spesifikasi**	**Label**
	2/2 TB (2 Lajur/2 Arah Tanpa Batas Median)	0
	4/2 B (4 Lajur/2 Arah dengan Batas Median)	1
Batas Kecepatan di Lokasi	20	0
	30	0
	40	0
	50	1
	60	1
	80	1
Status Jalan	Jalan Provinsi	1
	Jalan Kota	0
	Jalan Nasional	1
	Jalan Desa	0

Pada Tabel 1 menunjukkan bahwa setiap atribut diberikan label berbeda, dengan label 0 menunjukkan atribut yang tidak berpengaruh dan label 1 menunjukkan atribut yang berpengaruh terhadap kecelakaan lalu lintas. Label ini membantu memahami kontribusi atribut terhadap kecelakaan dan memungkinkan analisis penyebab yang lebih mendalam.

4.2. Pembagian Data

Data dibagi menjadi 80% untuk pelatihan model dan 20% untuk evaluasi kinerja [28], yang ditampilkan pada Gambar 3.

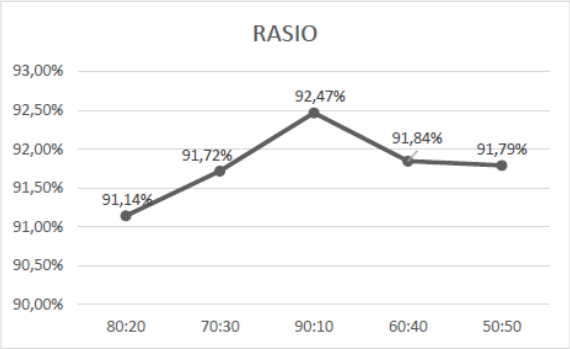
```
(X_train): 803 baris
(y_train): 803 baris
(X_test): 201 baris
(y_test): 201 baris
```

Gambar 3. Pembagian Data

Gambar 3 menunjukkan pembagian data, dengan 803 data pelatihan untuk melatih model dan 201 data pengujian untuk evaluasi kinerja model.

4.3. Pengujian Pembagian Data

Pengujian rasio pembagian data dilakukan untuk menilai dampaknya terhadap performa model, dengan hasil yang ditampilkan pada Gambar 4.



Gambar 4. Pengujian Pembagian Data

Pada Gambar 4 menunjukkan bahwa rasio pembagian data 90:10 menghasilkan akurasi tertinggi sebesar 92,47%, yang mengindikasikan bahwa rasio ini memberikan hasil optimal dan meningkatkan kinerja model secara signifikan.

4.4. *K-Fold Cross-Validation*

Penerapan K-Fold Cross-Validation dengan berbagai nilai K (9, 12, 21, 22, dan 27) menunjukkan K=9 memberikan hasil terbaik, hasil pengujian dapat dilihat pada Gambar 5.

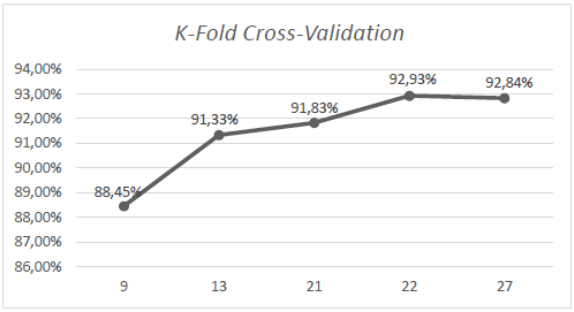
	Fold	Index	Actual (Biner Train)	Predicted (Biner Train)
0	1	0	1	1
1	1	1	1	1
2	1	2	1	1
3	1	3	1	1
4	1	4	1	1
...	...	...	...	...
7222	9	798	1	1
7223	9	799	0	0
7224	9	800	1	0
7225	9	801	1	1
7226	9	802	1	1

Gambar 5. K-Fold Cross-Validation

Pada Gambar 5 menampilkan hasil evaluasi *K-Fold Cross-Validation*, kolom *Fold* menunjukkan data uji, Index mencantumkan nomor data, *Actual* (*Biner*) menampilkan nilai asli dalam format *biner* (0 untuk ringan, 1 untuk sedang/berat), dan *Predicted* (*Biner*) berisi hasil prediksi model. Setiap baris membandingkan nilai asli dengan prediksi untuk menilai kinerja model.

4.5. Pengujian *K-Fold Cross-Validation*

Proses pengujian menggunakan *K-Fold Cross-Validation* dengan menentukan nilai K yang berbeda. Setiap nilai K diuji sebanyak 5 kali dan menunjukkan penurunan akurasi diikuti peningkatan seiring bertambahnya nilai K, hasil perubahan akurasi ini dapat dilihat pada Gambar 6.

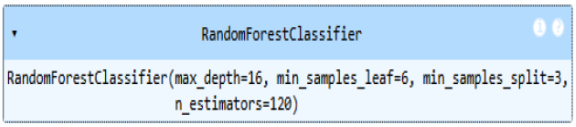


Gambar 6. Pengujian K-Fold Cross-Validation

Pada Gambar 6 pengujian *K-Fold Cross-Validation* menunjukkan bahwa nilai akurasi tertinggi tercapai pada K=22 dengan hasil sebesar 92,93%.

4.6. Parameter *Random Forest*

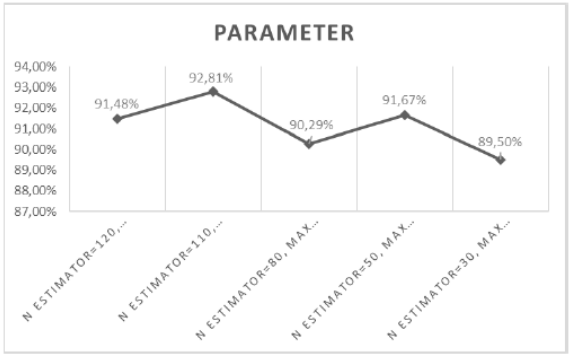
Random Forest adalah metode dalam *Machine Learning* yang menggunakan pendekatan e*nsemble* dengan membuat beberapa pohon keputusan dan menggabungkan hasilnya untuk meningkatkan efektivitas model serta mengurangi terjadinya *Overfitting*. Beberapa parameter dalam *Random Forest* untuk mengoptimalkan model dapat dilihat pada Gambar 7.



Gambar 7. Parameter Random Forest

Pada Gambar 3.5, parameter untuk metode *Random Forest* ditentukan *N_estimator* sebesar 120 untuk meningkatkan kestabilan hasil, *Max_depth* sebesar 16 untuk menangkap pola kompleks, *Min_Samples_Leaf* sebesar 6 untuk mencegah *Overfitting*, dan *Min_Samples_Split* sebesar 2 untuk memungkinkan pembagian node yang lebih detail.

4.7. Pengujian Parameter *Random Forest*



Gambar 8. Pengujian Parameter Random Forest

Pengujian parameter model *Random Forest* dengan berbagai nilai parameter menunjukkan

pengaruh perubahan parameter terhadap kinerja model, hasil pengujian dapat dilihat pada Gambar 8.

Pada Gambar 8 parameter yang mencapai akurasi tertinggi yaitu *n_estimator* = 110, *max_depth* = 12, *min_sample_leaf* = 3 dan *min_sample_split* =5 mencapai akurasi sebesar 92,81%.

4.8. Perhitungan nilai *Entropy* dan *Gain* peratribut

Perhitungan *Entropy* dan *Gain* untuk setiap atribut dalam data kecelakaan lalu lintas, memilih atribut yang paling efisien dalam mengurangi ketidakpastian dan meningkatkan akurasi klasifikasi pada pohon keputusan, memastikan model optimal, dilihat pada Gambar 9.

```
Entropy Total: 1.109
Gain Seluruh Atribut Kondis Cahaya: 0.12
Gain Seluruh Atribut Cuaca: 0.084
Gain Seluruh Atribut Kelas Jalan: 0.098
Gain Seluruh Atribut Tipe Jalan: 0.1
Gain Seluruh Atribut Kondisi Permukaan Jalan: 0.098
Gain Seluruh Atribut Batas Kecepatan Dilokasi: 0.206
Gain Seluruh Atribut Kemiringan Jalan: 0.04
Gain Seluruh Atribut Status Jalan: 0.097
```

Gambar 9. Perhitungan nilai Entropy dan Gain peratribut

Gambar 9 menunjukkan hasil perhitungan *Entropy* total sebesar 1.109, dengan atribut batas kecepatan di lokasi memiliki *Gain* tertinggi 0.206, yang berkontribusi besar dalam mengurangi ketidakpastian. Atribut tipe jalan, kondisi cahaya, dan kondisi permukaan jalan memberikan *Gain* signifikan, sementara atribut cuaca, kelas jalan, status jalan, dan kemiringan jalan menunjukkan kontribusi lebih kecil dalam pemisahan data.

4.9. Klasifikasi

Hasil prediksi klasifikasi yang diperoleh dari metode *Random Forest* serta data aktual yang digunakan dalam analisis dan memberikan gambaran tentang kemampuan model dalam memprediksi kelas serta membandingkan hasil prediksi dengan data sebenarnya dapat dilihat pada Gambar 10.

```
177        Ringan    Ringan
178        Sedang    Sedang
179        Ringan    Ringan
180        Sedang    Ringan
181        Sedang    Sedang
182        Sedang    Sedang
183        Ringan    Ringan
```

Gambar 10. Hasil Klaisifikasi

Pada Gambar 9 menunjukkan perbandingan nilai aktual dan prediksi untuk setiap data uji, sebagai contoh data uji 179 dengan label aktual ‘Ringan’ berhasil diprediksi tepat, sementara data uji nomor 180 yang seharusnya memiliki label ‘Sedang’ justru diprediksi ‘Ringan’, hal ini memberikan pemahaman tentang seberapa baik model *Random Forest* dalam melakukan klasifikasi.

4.10. Evaluasi

Hasil evaluasi menunjukkan pada rasio data, *K-Fold Cross-Validation*, parameter *Random Forest*, dan evaluasi model, dapat dilihatt pada Tabel 2.

Tabel 2. Evaluasi

Kelas	F1-Score (Total)	precision	Recall	F1-Score
0	95%	98%	94%	96%
1		90%	97%	94%
2		100%	100%	100%

Pada Tabel 2 menunjukkan hasil evaluasi dengan *K-Fold* = 22, rasio data 90:10 dan parameter *Random Forest n_estimator* = 110, *max_depth* = 12, *min_sample_leaf* = 3, *min_sample_split* = 5 memiliki hasil sebesar 95%, kelas ringan (0) *Precision* 98%, *Recall* 94%, *F1-Score* 96%, kelas sedang (1) *Precision* 90%, *Recall* 97%, *F1-Score* 94%, kelas berat (2) *Precision, Recall, F1-Score* 100%.

Secara keseluruhan, metode *Random Forest* terbukti efektif dalam mengklasifikasikan kecelakaan lalu lintas dan dapat menjadi dasar dalam pengambilan keputusan untuk kebijakan keselamatan yang lebih baik. Meskipun demikian, masih terdapat peluang untuk meningkatkan performa model dengan mengoptimalkan parameter lebih lanjut atau menerapkan metode perbaikan tambahan.

5. KESIMPULAN DAN SARAN

Penelitian klasifikasi kecelakaan lalu lintas di Kota Samarinda dengan metode *Random Forest* menunjukkan bahwa parameter optimal yang diperoleh adalah *n_estimator* = 110, *max_depth* = 12, *min_samples_leaf* = 3, dan *min_samples_split* = 5, yang meningkatkan performa model. Evaluasi dengan *F1-Score* menghasilkan 95%, membuktikan keefektifan *Random Forest* dalam mengklasifikasikan kecelakaan ke dalam tiga kategori. Peneliti selanjutnya disarankan menambah variabel signifikan, mengeksplorasi metode *SVM* dan *Neural Network*, serta menggunakan data lebih luas untuk meningkatkan akurasi dan ketahanan model.

DAFTAR PUSTAKA

[1] M. F. Pradana, D. E. Intari, and D. Pratidina D, “Analisa Kecelakaan Lalu Lintas Dan Faktor Penyebabnya Di Jalan Raya Cilegon,” *J. Kaji. Tek. Sipil*, vol. 4, no. 2, pp. 165–175, 2019, doi: 10.52447/jkts.v4i2.1492.

[2] Y. B. Utomo, I. Kurniasari, and I. Yanuartanti, “Penerapan Knowledge Discovery in Database Untuk Analisa Tingkat Kecelakaan Lalu Lintas,” *JTIK (Jurnal Tek. Inform. Kaputama)*, vol. 7, no. 1, pp. 171–180, 2023, doi: 10.59697/jtik.v7i1.61.

[3] T. H. M. Gultom, L. Sofia, T. Tjahjono, and S.

Sulistyono, "Gambaran Perilaku Disiplin Berlalu Lintas Dan Penyebab Kecelakaan Lalu Lintas Di Jalan Nasional Kota Samarinda," *J. Indones. Road Saf.*, vol. 2, no. 1, p. 56, 2019, doi: 10.19184/korlantas-jirs.v2i1.15044.

[4] B. Amyrulloh and Samuji, "Analisa Penyebab Pelanggaran Lalu Lintas Oleh Pengendara Kendaraan Bermotor," *Kult. J. Ilmu Sos. dan Hum.*, vol. 2, no. 2, pp. 81–103, 2024.

[5] Ni Putu Krisna Dewi, Ni Putu Rai Yuliartini, and Komang Febrinayanti Dantes, "Implementasi Undang-Undang Nomor 22 Tahun 2009 Tentang Lalu Lintas Dan Angkutan Jalan Terhadap Penegakan Hukum Pelaku Balapan Liar Di Kabupaten Jembrana," *J. Komunitas Yust.*, vol. 5, no. 2, pp. 383–399, 2022, doi: 10.23887/jatayu.v5i2.51631.

[6] S. Hartati and H. A. SAN, "Algoritma Naive Bayes untuk Prediksi Kelulusan Mahasiswa," *J. Cakrawala Inf.*, vol. 2, no. 2, pp. 42–50, 2022, doi: 10.54066/jci.v2i2.234.

[7] Ardi Ramdani, Christian Dwi Sofyan, Fauzi Ramdani, Muhamad Fauzi Arya Tama, and Muhammad Angga Rachmatsyah, "Algoritma Klasifikasi Data Mining Untuk Memprediksi Masyarakat Dalam Menerima Bantuan Sosial," *J. Ilm. Sist. Inf.*, vol. 1, no. 2, pp. 39–47, 2022, doi: 10.51903/juisi.v1i2.363.

[8] K. Tingkat *et al.*, "Classification of The Severity 0f Traffic Accident Victims In The City of Samarinda Uses The K-Nearest Neighbor and Naive Bayes Algorithms," *J. EKSPONENSIAL*, vol. 14, no. 2, pp. 99–106, 2023, [Online]. Available: http://jurnal.fmipa.unmul.ac.id/index.php/eksponensial99

[9] C. Naveen Kumar and M. Kumar, "Road Accident Analysis Using Random Forest Compared to K Nearest Neighbor to Increase Accuracy," *J. Surv. Fish. Sci.*, vol. 10, no. 1S, pp. 3014–3023, 2023, [Online]. Available: https://sifisheriessciences.com/journal/index.php/journal/article/view/535

[10] M. M. Chen and M. C. Chen, "Modeling road accident severity with comparisons of logistic regression, decision tree and random forest," *Inf.*, vol. 11, no. 5, 2020, doi: 10.3390/INFO11050270.

[11] M. R. U. Pulungan, D. E. Ratnawati, and B. Rahayudi, "Analisis Sentimen Ulasan Aplikasi PeduliLindungi dengan Metode Random Forest," *J. Pengemb. Teknol. Inf. dan Ilmu Komun.*, vol. 6, no. 9, pp. 4378–4385, 2022, [Online]. Available: http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/download/11582/5142

[12] R. E. Almamlook, K. M. Kwayu, M. R. Alkasisbeh, and A. A. Frefer, "Comparison of machine learning algorithms for predicting traffic accident severity," *2019 IEEE Jordan Int. Jt. Conf. Electr. Eng. Inf. Technol. JEEIT 2019 - Proc.*, no. April, pp. 272–276, 2019, doi: 10.1109/JEEIT.2019.8717393.

[13] Y. Priantama and T. A. Yoga Siswa, "Optimasi Correlation-Based Feature Selection Untuk Perbaikan Akurasi Random Forest Classifier Dalam Prediksi Performa Akademik Mahasiswa," *JIKO (Jurnal Inform. dan Komputer)*, vol. 6, no. 2, p. 251, 2022, doi: 10.26798/jiko.v6i2.651.

[14] Wartumi, R. Kurniawan, and A. Y. Wijaya, "Analisis Data Sentimen Ulasan Pengguna Aplikasi Shopee di Google Play Store dengan Klasifikasi Algoritma Naïve Bayes," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 6, no. 1, pp. 164–170, 2024.

[15] R. R. Adhitya, Wina Witanti, and Rezki Yuniarti, "Perbandingan Metode Cart Dan Naïve Bayes Untuk Klasifikasi Customer Churn," *INFOTECH J.*, vol. 9, no. 2, pp. 307–318, 2023, doi: 10.31949/infotech.v9i2.5641.

[16] F. Yulianto, R. Arifando, and A. A. Supianto, "Improved Eliminate Particle Swarm Optimization on Support Vector Machine for Freshwater Fish Classification," *Proc. 2019 4th Int. Conf. Sustain. Inf. Eng. Technol. SIET 2019*, pp. 38–43, 2019, doi: 10.1109/SIET48054.2019.8986119.

[17] Y. N. FUADAH, I. D. UBAIDULLAH, N. IBRAHIM, F. F. TALININGSING, N. K. SY, and M. A. PRAMUDITHO, "Optimasi Convolutional Neural Network dan K-Fold Cross Validation pada Sistem Klasifikasi Glaukoma," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 10, no. 3, p. 728, 2022, doi: 10.26760/elkomika.v10i3.728.

[18] J. L. Leevy, J. Hancock, T. M. Khoshgoftaar, and A. Abdollah Zadeh, "Investigating the effectiveness of one-class and binary classification for fraud detection," *J. Big Data*, vol. 10, no. 1, pp. 1–16, 2023, doi: 10.1186/s40537-023-00825-1.

[19] D. Alita, Y. Fernando, and H. Sulistiani, "Implementasi Algoritma Multiclass Svm Pada Opini Publik Berbahasa Indonesia Di Twitter," *J. Tekno Kompak*, vol. 14, no. 2, p. 86, 2020, doi: 10.33365/jtk.v14i2.792.

[20] R. S. Tantika and A. Kudus, "Penggunaan Metode Support Vector Machine Klasifikasi Multiclass pada Data Pasien Penyakit Tiroid," *Bandung Conf. Ser. Stat.*, vol. 2, no. 2, pp. 159–166, 2022, doi: 10.29313/bcss.v2i2.3590.

[21] A. I. Chandra, Yulia, and R. Adipranata, "Aplikasi Penentu Subyek Skripsi Menggunakan Metode Support Vector Machine," *J. Infra*, vol. 8, no. 2, 2020.

[22] I. G. A. Purnajiwa Arimbawa and N. A. Sanjaya ER, "Penerapan Metode Adaboost Untuk Multi-Label Classification Pada Dokumen Teks," *JELIKU (Jurnal Elektron. Ilmu Komput. Udayana)*, vol. 9, no. 1, p. 127, 2020, doi:

10.24843/jlk.2020.v09.i01.p13.
[23] G. A. Sandag, "Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest," *CogITo Smart J.*, vol. 6, no. 2, pp. 167–178, 2020, doi: 10.31154/cogito.v6i2.270.167-178.
[24] 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.
[25] H. A. Dwi Fasnuari, H. Yuana, and M. T. Chulkamdi, "Penerapan Algoritma K-Nearest Neighbor Untuk Klasifikasi Penyakit Diabetes Melitus," *Antivirus J. Ilm. Tek. Inform.*, vol. 16, no. 2, pp. 133–142, 2022, doi: 10.35457/antivirus.v16i2.2445.
[26] F. R. Lumbanraja, R. A. Saputra, K. Muludi, A. Hijriani, and A. Junaidi, "Implementasi Support Vector Machine Dalam Memprediksi Harga Rumah Pada Perumahan Di Kota Bandar Lampung," *J. Pepadun*, vol. 2, no. 3, pp. 327–335, 2021, doi: 10.23960/pepadun.v2i3.90.
[27] L. Britanthia Christina Tanuwijaya, B. Susanto, and A. Saragih, "Perbandingan Metode Regresi Logistik dan Random Forest untuk Klasifikasi Fitur Mode Audio Spotify," *Indones. J. Data Sci.*, vol. 1, no. 3, pp. 68–78, 2020.
[28] H. Hidayat, A. Sunyoto, and H. Al Fatta, "Klasifikasi Penyakit Jantung Menggunakan Random Forest Clasifier," *J. SISKOM-KB (Sistem Komput. dan Kecerdasan Buatan)*, vol. 7, no. 1, pp. 31–40, 2023, doi: 10.47970/siskom-kb.v7i1.464.
[29] M. A. As Sarofi, I. Irhamah, and A. Mukarromah, "Identifikasi Genre Musik dengan Menggunakan Metode Random Forest," *J. Sains dan Seni ITS*, vol. 9, no. 1, pp. 79–86, 2020, doi: 10.12962/j23373520.v9i1.51311.