

Klasifikasi Multi-Kelas Kondisi Lalu Lintas Perkotaan Menggunakan Random Forest

Naiya Aulia Ridha¹, Roberto Kaban²

¹Program Studi Rekayasa Perangkat Lunak, Fakultas Sains dan Teknologi, Institut Teknologi dan Bisnis Indonesia, Kab. Deli Serdang, Indonesia

²Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Institut Teknologi dan Bisnis Indonesia, Kab. Deli Serdang, Indonesia

surel: naiyaauliaridha@gmail.com, roberto.kaban@yahoo.com

Info Artikel

Sejarah artikel:

Diterima 18-06-2026

Revisi 11-07-2026

Diterima 02-08-2026

Kata kunci:

klasifikasi multi-kelas

kondisi lalu lintas

Random Forest

volume kendaraan

fitur temporal

ABSTRAK

Kemacetan lalu lintas perkotaan membutuhkan pendekatan klasifikasi yang mampu membedakan kondisi jalan secara lebih rinci agar pengelolaan lalu lintas dapat dilakukan berbasis data. Penelitian ini bertujuan mengklasifikasikan kondisi lalu lintas perkotaan secara multi-kelas menggunakan algoritma Random Forest berbasis volume kendaraan dan fitur temporal. Dataset yang digunakan adalah Traffic Prediction Dataset dari Kaggle yang terdiri atas 2.976 observasi dengan empat kelas target, yaitu low, normal, high, dan heavy. Fitur prediktor meliputi jumlah mobil, sepeda motor, bus, truk, total kendaraan, serta fitur temporal seperti jam, menit, periode waktu, dan status akhir pekan. Penelitian ini menggunakan dua skenario pemodelan, yaitu skenario dengan fitur Total dan skenario tanpa fitur Total. Evaluasi model dilakukan menggunakan accuracy, precision, recall, F1-score, 5-fold cross validation, ROC-AUC One-vs-Rest, serta analisis feature importance. Hasil penelitian menunjukkan bahwa skenario dengan fitur Total menghasilkan accuracy sebesar 0,9916 dan weighted F1-score sebesar 0,9915. Skenario tanpa fitur Total tetap memperoleh performa baik dengan accuracy sebesar 0,9396 dan weighted F1-score sebesar 0,9379. Analisis feature importance menunjukkan bahwa fitur Total menjadi prediktor paling dominan pada skenario pertama, sedangkan CarCount, TruckCount, dan BusCount menjadi fitur utama setelah fitur Total dihapus. Temuan ini menunjukkan bahwa Random Forest efektif untuk klasifikasi kondisi lalu lintas multi-kelas, sementara pemisahan skenario fitur membantu menjelaskan pengaruh fitur agregatif terhadap performa dan interpretasi model.

Penulis yang sesuai:

Naiya Aulia Ridha

Program Studi Rekayasa Perangkat Lunak, Fakultas Sains dan Teknologi, Institut Teknologi dan Bisnis Indonesia

Email: naiyaauliaridha@gmail.com

1. PENDAHULUAN

Kemacetan lalu lintas perkotaan menjadi persoalan transportasi yang berulang karena peningkatan volume kendaraan tidak selalu diikuti oleh kemampuan jaringan jalan dalam menampung pergerakan harian. Kondisi ini memengaruhi waktu tempuh, konsumsi bahan bakar, efisiensi mobilitas, dan kualitas hidup Masyarakat [1]. Dalam konteks sistem transportasi cerdas, pengelola lalu lintas membutuhkan pendekatan berbasis data untuk membaca perubahan kondisi jalan secara lebih terukur. Attioui dan Lahby melalui systematic review terhadap penelitian congestion forecasting periode 2010-2024 menunjukkan bahwa machine learning banyak digunakan dalam kajian prediksi kemacetan, dengan 115 studi memenuhi kriteria seleksi, dan tugas klasifikasi menjadi salah satu pendekatan yang paling banyak digunakan dalam penelitian kemacetan lalu lintas [2]. Temuan tersebut menunjukkan bahwa klasifikasi kondisi lalu lintas masih menjadi ruang penelitian yang relevan dalam pengembangan sistem pemantauan dan pengambilan keputusan transportasi berbasis data.

Penelitian terkait klasifikasi kondisi lalu lintas menunjukkan bahwa algoritma machine learning mampu membaca pola jalan dari data historis. Nasution et al. mengembangkan competitive machine learning system untuk klasifikasi kondisi lalu lintas di Kota Bandung dengan membandingkan Neural Network, K-Nearest Neighbor, Logistic Regression, Bayesian Network, Decision Tree, dan Random Forest. Hasil penelitian tersebut menunjukkan bahwa Random Forest memperoleh akurasi individual sebesar 72,6%, lebih tinggi dibanding K-Nearest Neighbor sebesar 70,5%, Decision Tree sebesar 72,2%, Logistic Regression sebesar 44,4%, dan Bayesian Network sebesar 46,1%. Sistem voting berbobot yang mereka usulkan menghasilkan akurasi 71,9% sampai 72,5% [3]. Studi tersebut menegaskan bahwa pemilihan algoritma perlu disesuaikan dengan karakteristik data lalu lintas, karena setiap model menghasilkan performa berbeda pada struktur data yang berbeda.

Random Forest banyak digunakan dalam klasifikasi lalu lintas karena mampu menangani data tabular, relasi non-linear, dan interaksi antarvariabel. Imam dan Hardianto menggunakan Random Forest untuk mengklasifikasikan tingkat kemacetan di Pekanbaru berdasarkan tiga fitur utama, yaitu kecepatan, kepadatan lalu lintas, dan rasio volume terhadap kapasitas jalan. Penelitian tersebut membagi tingkat kemacetan ke dalam kelas rendah, sedang, dan tinggi, lalu memperoleh akurasi sebesar 98,86% pada data uji. Analisis feature importance dalam studi tersebut juga menunjukkan bahwa kecepatan dan rasio volume terhadap kapasitas menjadi variabel paling dominan dalam menentukan tingkat kemacetan [4]. Hasil ini memperlihatkan bahwa Random Forest tidak hanya dapat digunakan untuk memperoleh prediksi kelas, tetapi juga untuk menilai kontribusi fitur terhadap keputusan model.

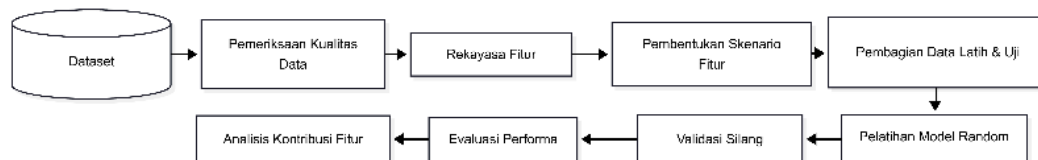
Traffic Prediction Dataset yang dipublikasikan oleh Aman di Kaggle menyediakan data lalu lintas berbasis hasil deteksi kendaraan, dengan fitur waktu, tanggal, hari, jumlah mobil, jumlah sepeda motor, jumlah bus, jumlah truk, total kendaraan, dan label Traffic Situation. Kolom Total merepresentasikan jumlah seluruh jenis kendaraan yang terdeteksi dalam durasi 15 menit, sehingga fitur ini memiliki posisi penting dalam pembacaan kondisi lalu lintas. Dataset yang sama telah digunakan oleh Febrianto et al. dalam penelitian prediksi situasi lalu lintas menggunakan algoritma K-Nearest Neighbors. Penelitian tersebut menggunakan 2.976 baris data dan 9 kolom, menerapkan pembagian data 80:20, serta memperoleh akurasi sebesar 0,8896 atau 89% pada data uji. Hasil pencarian grid dengan validasi silang menghasilkan akurasi terbaik sebesar 0,8503 atau 85% [5]. Studi tersebut memberi dasar awal bahwa Traffic Prediction Dataset dapat digunakan untuk klasifikasi kondisi lalu lintas, tetapi pengujian masih berfokus pada K-Nearest Neighbors dan belum membahas kontribusi fitur secara mendalam.

Berdasarkan penelitian terdahulu, kajian klasifikasi lalu lintas telah berkembang melalui berbagai algoritma machine learning, tetapi masih terdapat ruang untuk menguji model berbasis ensemble pada Traffic Prediction Dataset dan menafsirkan kontribusi fitur yang memengaruhi hasil klasifikasi. Fitur Total perlu diuji secara hati-hati karena bersifat agregatif dan dapat mendominasi keputusan model. Penelitian ini bertujuan mengklasifikasikan kondisi lalu lintas perkotaan secara multi-kelas menggunakan algoritma Random Forest berbasis volume kendaraan dan fitur temporal. Penelitian ini membandingkan dua skenario pemodelan, yaitu skenario dengan fitur Total dan skenario tanpa fitur Total, untuk menilai pengaruh fitur agregatif terhadap performa klasifikasi. Selain itu, penelitian ini menganalisis kontribusi fitur melalui feature importance agar

hasil model tidak berhenti pada nilai akurasi, tetapi juga menjelaskan peran komposisi kendaraan dan fitur waktu dalam membedakan kondisi low, normal, high, dan heavy.

2. METODE

Penelitian ini menggunakan pendekatan kuantitatif eksperimental berbasis supervised machine learning. Model yang digunakan adalah Random Forest untuk mengklasifikasikan kondisi lalu lintas perkotaan ke dalam empat kelas, yaitu low, normal, high, dan heavy. Tahapan penelitian meliputi pengumpulan dataset, prapemrosesan data, rekayasa fitur temporal, pembentukan skenario fitur, pelatihan model, evaluasi performa, dan analisis kontribusi fitur.



Gambar 1. Alur penelitian klasifikasi kondisi lalu lintas menggunakan Random Forest

2.1 Dataset Penelitian

Dataset yang digunakan adalah Traffic Prediction Dataset dari Kaggle yang dipublikasikan oleh Aman (<https://www.kaggle.com/datasets/hasibullahaman/traffic-prediction-dataset>). Dataset ini berisi data lalu lintas hasil deteksi kendaraan berbasis computer vision, dengan fitur waktu, tanggal, hari, jumlah mobil, jumlah sepeda motor, jumlah bus, jumlah truk, total kendaraan, dan label kondisi lalu lintas. Kolom Total merepresentasikan jumlah seluruh jenis kendaraan yang terdeteksi dalam interval 15 menit.

Dataset terdiri atas 2.976 baris dan 9 kolom utama, yaitu Time, Date, Day of the week, CarCount, BikeCount, BusCount, TruckCount, Total, dan Traffic Situation. Variabel prediktor dalam penelitian ini meliputi jumlah kendaraan berdasarkan jenis kendaraan dan fitur temporal. Variabel target adalah Traffic Situation, yang terdiri atas empat kelas, yaitu low, normal, high, dan heavy. Pemilihan dataset ini didasarkan pada ketersediaan fitur volume kendaraan dan informasi waktu yang relevan untuk membangun model klasifikasi kondisi lalu lintas secara multi-kelas.

2.2 Prapemrosesan dan Rekayasa Fitur

Prapemrosesan data dilakukan untuk memastikan dataset siap digunakan dalam pemodelan. Tahap ini mencakup pemeriksaan nilai kosong, pemeriksaan data duplikat, pengecekan tipe data, serta pengubahan variabel kategorik menjadi bentuk numerik. Variabel Day of the week dan TimePeriod diproses menggunakan one-hot encoding agar dapat digunakan oleh algoritma Random Forest tanpa membentuk asumsi urutan antar kategori.

Rekayasa fitur temporal dilakukan dengan mengekstraksi informasi waktu dari kolom Time. Fitur yang dibentuk meliputi Hour, Minute, TimeInMinutes, IsWeekend, dan TimePeriod. Fitur TimeInMinutes digunakan untuk merepresentasikan waktu dalam satuan menit sejak awal hari, dengan rumus sebagai berikut:

$$TimeInMinutes = (Hour \times 60) + Minute \quad (1)$$

Fitur IsWeekend digunakan untuk membedakan hari kerja dan akhir pekan. Nilai 1 diberikan pada hari Sabtu dan Minggu, sedangkan nilai 0 diberikan pada hari kerja. Rumus pengkodeannya ditulis sebagai berikut:

$$IsWeekend = \begin{cases} 1, & \text{jika hari Sabtu atau Minggu} \\ 0, & \text{jika hari kerja} \end{cases} \quad (2)$$

Fitur TimePeriod dibentuk untuk mengelompokkan waktu pengamatan ke dalam periode harian, seperti pagi, siang, sore, dan malam. Fitur ini digunakan karena kondisi lalu lintas perkotaan cenderung berubah mengikuti ritme aktivitas harian.

2.3 Skenario Pemodelan Random Forest

Penelitian ini menggunakan dua skenario pemodelan. Skenario pertama menggunakan seluruh fitur, termasuk fitur Total. Skenario kedua menghapus fitur Total dari variabel prediktor. Pemisahan skenario ini dilakukan karena Total merupakan fitur agregatif yang berasal dari penjumlahan CarCount, BikeCount, BusCount, dan TruckCount. Dengan membandingkan dua skenario tersebut, penelitian dapat menilai sejauh mana performa model dipengaruhi oleh fitur agregatif dan sejauh mana model tetap mampu membaca kondisi lalu lintas berdasarkan komposisi kendaraan serta fitur temporal.

Random Forest dipilih karena algoritma ini membangun banyak pohon keputusan dan menggabungkan hasil prediksi dari setiap pohon untuk menghasilkan keputusan akhir [6], [7]. Dalam implementasi scikit-learn, Random Forest menggunakan beberapa decision tree pada subsampel data dan mendukung kriteria pemisahan seperti Gini impurity [8], [9]. Rumus Gini impurity yang digunakan dalam proses pemisahan node adalah sebagai berikut:

$$Gini = 1 - \sum_{i=1}^C p_i^2 \quad (3)$$

Keterangan: p_i adalah proporsi data pada kelas ke- i , sedangkan C adalah jumlah kelas. Semakin kecil nilai Gini, semakin homogen data pada suatu node.

Model Random Forest dibangun menggunakan parameter `n_estimators=200`, `random_state=42`, dan `class_weight="balanced"`. Parameter `class_weight="balanced"` digunakan karena distribusi kelas target tidak sepenuhnya seimbang. Bobot kelas dihitung dengan rumus:

$$w = \frac{N}{C \times n} \quad (4)$$

Keterangan: w adalah bobot kelas, N adalah jumlah seluruh data latih, C adalah jumlah kelas, dan n adalah jumlah data pada kelas tertentu. Penggunaan bobot kelas membantu model mengurangi kecenderungan prediksi yang terlalu dominan terhadap kelas mayoritas.

2.4 Validasi dan Evaluasi Model

Dataset dibagi menjadi data latih dan data uji dengan proporsi 80:20. Pembagian dilakukan menggunakan stratified split agar proporsi setiap kelas target tetap terjaga pada data latih dan data uji. Setelah itu, model divalidasi menggunakan 5-fold cross validation untuk menilai kestabilan performa model pada beberapa pembagian data [10].

Rata-rata skor cross validation dihitung dengan rumus:

$$CV_{mean} = \frac{1}{K} \sum_{i=1}^K S_i \quad (5)$$

Keterangan: CV_{mean} adalah rata-rata skor validasi silang, K adalah jumlah fold, dan S_i adalah skor pada fold ke- i . Pada penelitian ini, nilai K ditetapkan sebesar 5.

Evaluasi performa model dilakukan menggunakan confusion matrix, accuracy, precision, recall, dan F1-score. F1-score merupakan rata-rata harmonik dari precision dan recall, serta dapat digunakan pada klasifikasi multi-kelas melalui pengaturan average seperti macro dan weighted [11], [12]. Rumus evaluasi yang digunakan adalah sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

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

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

$$F1\text{-score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (9)$$

Keterangan: TP adalah true positive, TN adalah true negative, FP adalah false positive, dan FN adalah false negative. Karena target penelitian terdiri atas empat kelas, hasil precision, recall, dan F1-score dilaporkan dalam bentuk macro average dan weighted average. Macro average digunakan untuk melihat rata-rata performa antar kelas, sedangkan weighted average mempertimbangkan jumlah data pada setiap kelas [13].

2.5 Analisis Feature Importance

Analisis feature importance dilakukan untuk mengetahui fitur yang paling berkontribusi terhadap hasil klasifikasi [14]. Penelitian ini menggunakan dua pendekatan, yaitu Mean Decrease in Impurity dan Permutation Importance. Mean Decrease in Impurity digunakan untuk membaca kontribusi fitur berdasarkan penurunan impurity pada pohon keputusan. Permutation Importance digunakan sebagai pembanding karena metode ini mengukur penurunan skor model setelah nilai suatu fitur diacak.

Menurut dokumentasi scikit-learn, permutation importance didefinisikan sebagai selisih antara skor baseline model dan skor setelah kolom fitur tertentu dipermutasi [15]. Rumus permutation importance ditulis sebagai berikut:

$$PI_j = S_{baseline} - S_{permuted(j)} \quad (10)$$

Keterangan: PI_j adalah nilai permutation importance fitur ke- j , $S_{baseline}$ adalah skor model sebelum fitur diacak, dan $S_{permuted(j)}$ adalah skor model setelah fitur ke- j diacak. Pada penelitian ini, skor yang digunakan dalam permutation importance adalah weighted F1-score agar kontribusi fitur tetap mempertimbangkan distribusi kelas target.

3. HASIL DAN PEMBAHASAN

3.1 Karakteristik Dataset

Dataset penelitian terdiri atas 2.976 observasi dengan target klasifikasi Traffic Situation yang terbagi ke dalam empat kelas, yaitu normal, heavy, high, dan low. Dataset ini berasal dari Traffic Prediction Dataset Kaggle yang memuat data hasil deteksi kendaraan, meliputi jumlah mobil, sepeda motor, bus, truk, total kendaraan, serta informasi waktu pengamatan. Pada dataset tersebut, fitur Total merepresentasikan jumlah seluruh jenis kendaraan yang terdeteksi dalam interval 15 menit.

Tabel 1. Distribusi Kelas dan Kualitas Data

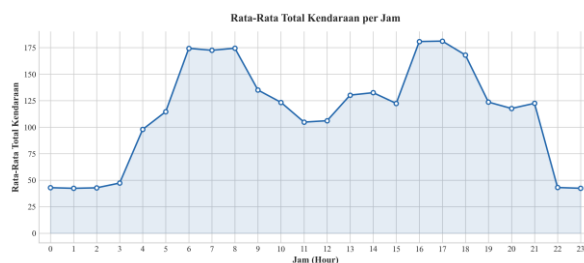
Aspek	Kategori	Jumlah	Persentase	Keterangan
Distribusi kelas	normal	1.669	56,08%	Kelas mayoritas
Distribusi kelas	heavy	682	22,92%	Kelas kedua terbesar
Distribusi kelas	high	321	10,79%	Kelas minoritas
Distribusi kelas	low	304	10,22%	Kelas minoritas
Kualitas data	Missing value	0	0%	Tidak ditemukan nilai kosong
Kualitas data	Duplikasi	0	0%	Tidak ditemukan data ganda

Distribusi kelas pada Tabel 1 menunjukkan bahwa kelas normal mendominasi data, sedangkan kelas high dan low memiliki jumlah observasi lebih kecil. Kondisi ini perlu diperhatikan dalam evaluasi model karena akurasi saja dapat memberi gambaran yang kurang lengkap pada data multi-kelas dengan proporsi kelas yang tidak seimbang. Oleh karena itu, penelitian ini juga menggunakan macro F1 dan weighted F1 untuk membaca performa model secara lebih seimbang.

Analisis volume kendaraan memperlihatkan hubungan yang jelas antara jumlah kendaraan dan kondisi lalu lintas. Kelas heavy memiliki rata-rata total kendaraan tertinggi, yaitu 198,86 unit. Pada kelas ini, mobil menjadi komponen dominan dengan rata-rata 130,89 unit, diikuti bus 33,01 unit dan sepeda motor 29,78 unit. Kelas high memiliki rata-rata total kendaraan sebesar 142,43 unit, sedangkan kelas low mencatat rata-rata total kendaraan sebesar 64,99 unit. Pola tersebut menunjukkan bahwa peningkatan total kendaraan berkaitan langsung dengan perubahan tingkat kepadatan lalu lintas.



Pola temporal juga menunjukkan variasi yang kuat. Rata-rata total kendaraan meningkat pada pagi hari sekitar pukul 06.00 sampai 08.00 dan kembali meningkat pada sore hari sekitar pukul 16.00 sampai 18.00. Nilai rata-rata tertinggi muncul sekitar pukul 17.00 dengan 181,05 unit. Pada pukul 06.00 dan 08.00, rata-rata total kendaraan berada pada kisaran 174 unit, sedangkan pada dini hari volume kendaraan berada pada kisaran 42 sampai 47 unit. Pola ini mengindikasikan bahwa fitur waktu membantu model membaca perubahan kondisi lalu lintas berdasarkan ritme aktivitas harian.



Gambar 2. Tren Rata-Rata Total Kendaraan per Jam

Gambar 2 menampilkan perubahan rata-rata total kendaraan dari pukul 00.00 sampai 23.00. Grafik ini cukup penting karena penelitian menggunakan fitur temporal. Puncak pagi dan sore pada grafik dapat memperkuat argumen bahwa kondisi lalu lintas tidak hanya dipengaruhi oleh volume kendaraan, tetapi juga oleh waktu pengamatan.

3.2 Kinerja Model pada Dua Skenario

Pengujian model dilakukan pada dua skenario. Skenario pertama menggunakan seluruh fitur, termasuk Total. Skenario kedua menghapus fitur Total untuk menilai kemampuan model dalam membaca kondisi lalu lintas berdasarkan komposisi kendaraan dan fitur temporal. Perbandingan performa kedua skenario disajikan pada Tabel 2.

Tabel 2. Perbandingan Kinerja Model Random Forest

Metrik	Dengan Total	Tanpa Total
Accuracy	0,9916	0,9396
Macro Precision	0,9952	0,9397
Macro Recall	0,9836	0,8941
Macro F1	0,9892	0,9122
Weighted Precision	0,9917	0,9406
Weighted Recall	0,9916	0,9396
Weighted F1	0,9915	0,9379
CV Accuracy Mean	0,9903	0,9392
CV F1 Weighted Mean	0,9902	0,9382
ROC-AUC Weighted OVR	0,9999	0,9951

Tabel 2 menunjukkan bahwa skenario dengan fitur Total menghasilkan performa tertinggi pada seluruh metrik. Accuracy mencapai 0,9916 dan weighted F1-score mencapai 0,9915. Nilai ini menunjukkan bahwa model mampu mengklasifikasikan empat kelas kondisi lalu lintas dengan tingkat kesalahan yang rendah. Nilai cross-validation juga stabil, dengan CV accuracy mean sebesar 0,9903 dan CV F1 weighted mean sebesar 0,9902.

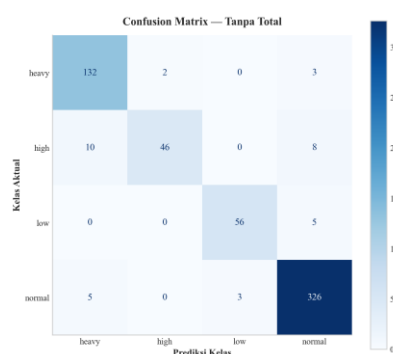
Skenario tanpa fitur Total mengalami penurunan performa, tetapi model tetap menunjukkan hasil yang kuat. Accuracy sebesar 0,9396 dan weighted F1-score sebesar 0,9379 menunjukkan bahwa komposisi kendaraan dan fitur temporal masih mampu membentuk pola klasifikasi yang baik. Penurunan performa setelah penghapusan Total menunjukkan bahwa fitur agregatif tersebut berperan besar dalam memisahkan kelas kondisi lalu lintas.

Hasil ini memperlihatkan dua fungsi skenario yang berbeda. Skenario dengan Total lebih kuat untuk kebutuhan prediksi cepat karena model memperoleh ringkasan langsung dari intensitas kendaraan. Skenario tanpa Total lebih berguna untuk analisis karena kontribusi masing-masing jenis kendaraan dapat terlihat lebih jelas.

3.3 Analisis Kesalahan Klasifikasi

Analisis confusion matrix menunjukkan bahwa model dengan fitur Total memiliki kesalahan klasifikasi yang sangat kecil. Kelas heavy terklasifikasi benar sebanyak 137 dari 137 observasi, sedangkan kelas low terklasifikasi benar sebanyak 61 dari 61 observasi. Pada kelas normal, model memprediksi 333 dari 334 observasi dengan benar. Kesalahan paling terlihat terdapat pada kelas high, yaitu 60 dari 64 observasi diprediksi benar dan 4 observasi bergeser ke kelas normal.

Pada skenario tanpa fitur Total, kesalahan klasifikasi meningkat, terutama pada kelas high. Kelas normal terklasifikasi benar sebanyak 326 dari 334 observasi. Kelas heavy terklasifikasi benar sebanyak 132 dari 137 observasi. Kelas low terklasifikasi benar sebanyak 56 dari 61 observasi. Kelas high menjadi kelas dengan performa paling rendah, dengan 46 dari 64 observasi terprediksi benar. Sebanyak 10 observasi high bergeser ke heavy, sedangkan 8 observasi high bergeser ke normal.



Gambar 3. Confusion Matrix Skenario tanpa Total

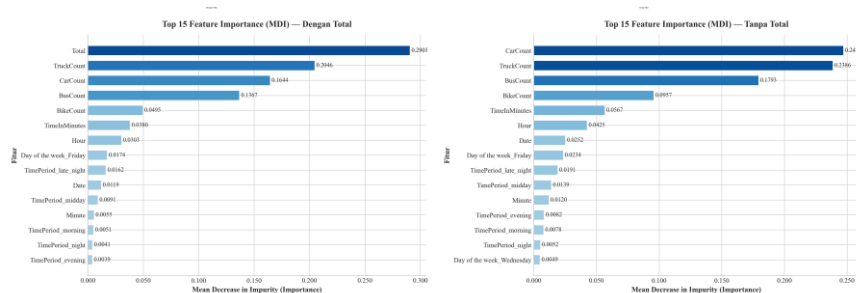
Gambar 3 lebih penting ditampilkan dibanding confusion matrix skenario dengan Total karena kesalahan model lebih terlihat pada skenario tanpa fitur agregatif. Kesalahan pada kelas high menunjukkan bahwa kelas ini memiliki karakter transisi antara normal dan heavy. Secara lalu lintas, kondisi high menggambarkan kepadatan yang meningkat, tetapi belum mencapai karakteristik heavy secara penuh. Karena itu, sebagian observasi high memiliki pola volume kendaraan yang mendekati normal atau heavy.

Hasil ini menjelaskan mengapa macro recall dan macro F1 mengalami penurunan lebih besar pada skenario tanpa Total. Kelas dengan jumlah observasi lebih kecil dan batas karakteristik berdekatan lebih sulit dikenali ketika fitur agregatif dihapus. Dalam konteks operasional, kelas high perlu mendapat perhatian karena kelas ini dapat menjadi sinyal awal perubahan menuju kondisi heavy.

3.4 Analisis Feature Importance

Analisis feature importance menunjukkan bahwa fitur Total menjadi prediktor paling dominan pada skenario pertama. Berdasarkan Mean Decrease in Impurity, fitur Total memperoleh nilai 0,2905, diikuti TruckCount sebesar 0,2046, CarCount sebesar 0,1644, dan BusCount sebesar 0,1367. Pada permutation importance, fitur Total juga berada pada posisi tertinggi dengan nilai rata-rata 0,4085. Permutation importance mengukur kontribusi fitur dengan cara mengacak nilai suatu fitur dan melihat penurunan skor model setelah fitur tersebut diacak, sehingga metode ini berguna untuk memeriksa pengaruh fitur pada model yang sudah dilatih [9].

Pada skenario tanpa fitur Total, susunan fitur utama berubah. Berdasarkan MDI, CarCount menjadi fitur paling penting dengan nilai 0,2470, diikuti TruckCount sebesar 0,2386, BusCount sebesar 0,1793, dan BikeCount sebesar 0,0957. Berdasarkan permutation importance, TruckCount menempati posisi tertinggi dengan nilai 0,2263, diikuti CarCount sebesar 0,1907 dan BusCount sebesar 0,1636.



Gambar 4. Feature Importance Dua Skenario

Gambar 4 dibuat dalam dua panel agar pembaca dapat melihat perubahan kontribusi fitur secara langsung. Panel pertama menunjukkan dominasi Total. Panel kedua menunjukkan pergeseran kontribusi ke CarCount, TruckCount, dan BusCount setelah Total dihapus.

Dominasi Total menunjukkan bahwa total volume kendaraan menjadi indikator paling kuat dalam membedakan kelas kondisi lalu lintas. Hal ini wajar karena fitur tersebut merangkum seluruh kendaraan yang melintasi titik pengamatan. Namun, dominasi tersebut juga perlu dibaca secara hati-hati. Fitur Total dapat memperkuat performa prediktif, tetapi pada saat yang sama dapat menutupi kontribusi jenis kendaraan lain karena fitur tersebut merupakan akumulasi dari CarCount, BikeCount, BusCount, dan TruckCount.

Skenario tanpa Total memberikan hasil interpretatif yang lebih rinci. CarCount menunjukkan pengaruh besar karena mobil menjadi komponen dominan pada kelas heavy. TruckCount dan BusCount juga memberi kontribusi penting karena kendaraan besar membutuhkan ruang jalan lebih besar dan dapat memengaruhi kelancaran arus. Perbedaan urutan antara MDI dan permutation importance menunjukkan bahwa kontribusi fitur dapat muncul melalui mekanisme yang berbeda. MDI membaca kontribusi fitur pada pemisahan node, sedangkan permutation importance membaca dampak pengacakan fitur terhadap skor model [16].

3.5 Pembahasan Temuan

Hasil penelitian menunjukkan bahwa Random Forest mampu mengklasifikasikan kondisi lalu lintas perkotaan secara multi-kelas dengan performa tinggi. Skenario dengan fitur Total menghasilkan accuracy sebesar 0,9916 dan weighted F1-score sebesar 0,9915. Skenario tanpa fitur Total tetap menghasilkan performa kuat dengan accuracy sebesar 0,9396 dan weighted F1-score sebesar 0,9379. Temuan ini memperlihatkan bahwa Random Forest dapat membaca pola volume kendaraan dan fitur temporal pada data lalu lintas tabular.

Jika dibandingkan dengan penelitian Febrianto et al. yang menggunakan dataset Kaggle yang sama dengan algoritma K-Nearest Neighbors, penelitian tersebut melaporkan accuracy sebesar 0,8896 pada data uji [5]. Perbandingan ini menunjukkan bahwa Random Forest memberi hasil yang lebih tinggi pada penelitian ini, meskipun perbandingan tersebut tidak dapat dibaca sebagai perbandingan eksperimental langsung karena konfigurasi fitur, rekayasa fitur, dan proses evaluasi dapat berbeda.

Hasil ini juga sejalan dengan penelitian Imam dan Hardianto yang menggunakan Random Forest untuk klasifikasi kemacetan lalu lintas di Pekanbaru. Penelitian tersebut menggunakan fitur kecepatan, kepadatan, dan rasio volume terhadap kapasitas, lalu memperoleh accuracy sebesar 98,86% pada data uji [4]. Kesamaan pola temuan terdapat pada kemampuan Random Forest dalam menangani data lalu lintas berbasis parameter operasional dan dalam menjelaskan kontribusi fitur melalui feature importance.

Perbedaan utama penelitian ini terletak pada pengujian dua skenario fitur. Fitur Total memberi performa tertinggi karena menjadi ringkasan langsung dari intensitas kendaraan. Skenario tanpa Total memberi informasi yang lebih analitis karena model harus membaca kondisi lalu lintas dari komposisi kendaraan dan fitur waktu. Dengan demikian, kedua skenario tidak perlu diposisikan sebagai pilihan yang saling menggantikan. Skenario dengan Total berguna untuk prediksi cepat, sedangkan skenario tanpa Total berguna untuk memahami jenis kendaraan yang paling memengaruhi perubahan kondisi jalan.

Pola temporal memperkuat hasil klasifikasi. Puncak kendaraan pada pagi dan sore hari menunjukkan bahwa kondisi heavy berkaitan dengan jam aktivitas tinggi. Pada malam hari, kelas heavy hampir tidak muncul,

sedangkan kelas normal lebih dominan. Temuan ini memperlihatkan bahwa fitur waktu membantu model membedakan jam lengang, jam transisi, dan jam puncak. Dalam penerapan sistem pemantauan lalu lintas, informasi ini dapat digunakan untuk mengenali jam rawan kepadatan dan menyiapkan respons pengaturan arus kendaraan.

Keterbatasan penelitian terletak pada ruang lingkup dataset. Dataset yang digunakan belum memuat informasi lokasi ruas jalan, kapasitas jalan, geometri jalan, cuaca, sinyal lalu lintas, hari libur, dan kejadian khusus. Padahal, variabel tersebut dapat memengaruhi kondisi lalu lintas di lapangan. Penelitian lanjutan perlu menguji model pada dataset real-time dan lokasi yang lebih beragam. Pengujian terhadap algoritma lain, seperti XGBoost, Support Vector Machine, atau model deret waktu, juga dapat dilakukan untuk memperoleh pembandingan yang lebih komprehensif.

4. KESIMPULAN

Penelitian ini mengklasifikasikan kondisi lalu lintas perkotaan secara multi-kelas menggunakan algoritma Random Forest pada Traffic Prediction Dataset Kaggle yang memuat fitur jumlah kendaraan berdasarkan jenis kendaraan, total kendaraan, waktu pengamatan, dan label kondisi lalu lintas. Hasil pengujian menunjukkan bahwa Random Forest mampu menghasilkan performa klasifikasi yang tinggi pada dua skenario pemodelan. Skenario dengan fitur Total memperoleh accuracy sebesar 0,9916 dan weighted F1-score sebesar 0,9915, sedangkan skenario tanpa fitur Total tetap menghasilkan performa baik dengan accuracy sebesar 0,9396 dan weighted F1-score sebesar 0,9379.

Perbandingan kedua skenario menunjukkan bahwa fitur Total memiliki pengaruh besar terhadap performa model karena merepresentasikan akumulasi seluruh jenis kendaraan. Namun, skenario tanpa Total memberikan pemahaman yang lebih rinci terhadap kontribusi masing-masing jenis kendaraan. Pada skenario tersebut, fitur CarCount, TruckCount, dan BusCount menjadi prediktor utama dalam membedakan kondisi lalu lintas. Hasil confusion matrix juga menunjukkan bahwa kelas high menjadi kelas yang paling sulit diklasifikasikan, karena karakteristiknya berada di antara kondisi normal dan heavy.

Secara keseluruhan, Random Forest dapat digunakan sebagai model klasifikasi kondisi lalu lintas berbasis volume kendaraan dan fitur temporal. Penelitian ini juga menunjukkan bahwa analisis feature importance penting untuk menafsirkan hasil model, terutama ketika dataset memuat fitur agregatif seperti Total. Penelitian selanjutnya disarankan untuk menguji model pada data real-time, menambahkan variabel kontekstual seperti lokasi ruas jalan, cuaca, kapasitas jalan, dan kondisi sinyal lalu lintas, serta membandingkan Random Forest dengan algoritma lain seperti XGBoost, Support Vector Machine, atau model berbasis deret waktu agar hasil evaluasi menjadi lebih komprehensif.

REFERENSI

- [1] R. Pradana dan I. P. Handayani, "PERBANDINGAN EFEKTIVITAS WAKTU FIXED-TIME TRAFFIC CONTROL DAN ACTUATED TRAFFIC CONTROL DALAM PENGURAIAN KEMACETAN," *JIKA (Jurnal Informatika)*, vol. 7, no. 4, hlm. 440, Jul 2023, doi: 10.31000/jika.v7i4.8668.
- [2] M. Attioui dan M. Lahby, "Congestion Forecasting Using Machine Learning Techniques: A Systematic Review," *Future Transportation*, vol. 5, no. 3, hlm. 76, Jul 2025, doi: 10.3390/futuretransp5030076.
- [3] S. M. Nasution, R. R. Septiawan, R. M. Fikri, dan B. Dirgantoro, "Traffic Management Enhancement: A Competitive Machine Learning System for Traffic Condition Classification," *International Journal of Transport Development and Integration*, vol. 8, no. 4, hlm. 553–567, Jul 2024, doi: 10.18280/ijtdi.080407.
- [4] K. Imam dan R. Hardianto, "Random Forest based traffic congestion classification in Pekanbaru, Indonesia," *ComniTech: Journal of Computational Intelligence and Informatics*, vol. 2, no. 2, hlm. 80–89, Jul 2026, [Daring]. Tersedia pada: <https://journal.unilak.ac.id/index.php/ComniTech/article/view/31577>
- [5] M. F. Febrianto dkk., "Prediksi Situasi Lalu Lintas Menggunakan Machine Learning Dengan Algoritma K-Nearest Neighbors Classifier," *Informatik Jurnal Ilmu Komputer*, vol. 20, no. 1, hlm. 28–34, Jul 2024, doi: 10.52958/iftk.v20i1.7042.
- [6] L. Breiman, "Random Forests," *Mach. Learn.*, vol. 45, no. 1, hlm. 5–32, Jul 2001, doi: 10.1023/a:1010933404324.
- [7] H. A. Salman, A. Kalakech, dan A. Steiti, "Random Forest Algorithm Overview," *Babylonian Journal of Machine Learning*, vol. 2024, hlm. 69–79, Jul 2024, doi: 10.58496/bjml/2024/007.
- [8] L. Camadini, Y. Bouzid, M. Merlet, dan L. Carron, "Randomness control and reproducibility study of random forest algorithm in R and Python," *Cornell University*, Jul 2024. doi: 10.48550/arxiv.2408.12184.
- [9] D. S dan B. Praveen, "Random Forest-Based Forensic Investigation of Non-Fungible Tokens: for Enhanced Detection and Anomaly Identification," *Int. J. Sci. Res. Sci. Technol.*, vol. 11, no. 6, hlm. 212–219, Jul 2024, doi: 10.32628/ijrsrst2411497.
- [10] H. Mulyo dan A. K. Zyen, "Pengaruh Hyperparameter Tuning Gradient Boosting Terhadap Prediksi Pemilihan Program Studi Mahasiswa Baru," *Juli 2025*. doi: 10.47065/bulletincsr.v5i2.454.



- [11] A. Singh, A. Joshi, J. Jiang, dan H.-Y. Paik, "A survey of classification tasks and approaches for legal contracts," *Artif. Intell. Rev.*, vol. 58, no. 12, Jul 2025, doi: 10.1007/s10462-025-11359-8.
- [12] A. F. Communications, E. A. 2026, dan J. Onners, "Machine Learning Applications to Predict Military Readiness," Juli 2026. doi: 10.48448/5y85-gx12.
- [13] K. Takahashi, K. Yamamoto, A. Kuchiba, dan T. Koyama, "Confidence interval for micro-averaged F1 and macro-averaged F1 scores," *Applied Intelligence*, vol. 52, no. 5, hlm. 4961–4972, Jul 2021, doi: 10.1007/s10489-021-02635-5.
- [14] R. A. Azizah, F. A. Bachtiar, dan S. Adinugroho, "Klasifikasi Kinerja Akademik Siswa Menggunakan Neighbor Weighted K-Nearest Neighbor dengan Seleksi Fitur Information Gain," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 9, no. 3, hlm. 605–614, Jul 2022, doi: 10.25126/jtiik.2022935751.
- [15] R. Espinosa, G. Sánchez, J. M. Palma, dan F. Jiménez, "Permutation-based multi-objective evolutionary feature selection for high-dimensional data," Jul 2025. doi: 10.48550/arxiv.2501.14310.
- [16] K. Neupane dan I. Griva, "A Random Forest approach to detect and identify Unlawful Insider Trading," Cornell University, Jul 2024. doi: 10.48550/arxiv.2411.13564.