# Perbandingan Algoritma *Random Forest* dan *Gradient Boosting* dalam Memprediksi Tingkat Kemacetan Lalu Lintas Pada Kabupaten Karawang

**Siti Qoriah Ambarwati[1], Alvita Syaharani Tuswanto [2], Nina Amelina[3], Dony Oscar[4]**

[1,2,3,4]Fakultas Teknik dan Informatika, Sistem Informasi, Universitas Bina Sarana Informatika, Jakarta, Indonesia

Email: [1,*]19251338@bsi.ac.id, [2]19252672@bsi.ac.id, [3]19252247@bsi.ac.id, [4]dony.dor@bsi.ac.id

[*)] Email Penulis Utama

**Abstrak−** Kemacetan lalu lintas merupakan salah satu permasalahan yang sering terjadi di Kabupaten Karawang sebagai kawasan industri dengan tingkat mobilitas yang tinggi. Kondisi ini berdampak pada efisiensi transportasi, aktivitas ekonomi, dan kenyamanan masyarakat. Oleh karena itu, diperlukan metode prediksi yang mampu membantu mengidentifikasi tingkat kemacetan secara lebih efektif. Penelitian ini bertujuan membandingkan kinerja algoritma *Random Forest* dan *Gradient Boosting* dalam memprediksi tingkat kemacetan lalu lintas di Kabupaten Karawang menggunakan data *Google Traffic*. Metode penelitian yang digunakan adalah kuantitatif dengan pendekatan eksperimen. Data sekunder sebanyak 585 data dikumpulkan melalui observasi *Google Maps* pada delapan titik rawan kemacetan selama periode April–Mei 2026. Tahapan penelitian meliputi preprocessing data, transformasi dan encoding fitur, pembagian data latih dan data uji dengan rasio 80:20, pemodelan menggunakan algoritma *Random Forest* dan *Gradient Boosting*, serta evaluasi menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score*. Hasil penelitian menunjukkan bahwa algoritma *Gradient Boosting* memiliki performa lebih baik dibandingkan *Random Forest*. *Gradient Boosting* memperoleh *accuracy* sebesar 58,12% dan *F1-score* macro sebesar 0,44, sedangkan *Random Forest* memperoleh *accuracy* sebesar 51,28% dan *F1-score* macro sebesar 0,39. Faktor waktu dan lokasi menjadi variabel yang paling berpengaruh terhadap tingkat kemacetan.

**Kata Kunci:** Kemacetan Lalu Lintas, *Machine Learning*, *Random Forest*, *Gradient Boosting*, *Google Traffic*.

**Abstract−** Traffic congestion is one of the major problems in Karawang Regency, an industrial area with a high level of mobility. This condition affects transportation efficiency, economic activities, and public convenience. Therefore, a prediction method is needed to help identify traffic congestion levels more effectively. This study aims to compare the performance of the *Random Forest* and *Gradient Boosting* algorithms in predicting traffic congestion levels in Karawang Regency using *Google Traffic* data. This research employed a quantitative experimental approach. A total of 585 secondary data records were collected through *Google Maps* observations at eight congestion-prone locations during April–May 2026. The research stages included data preprocessing, feature transformation and encoding, splitting the *dataset* into *training* and *testing* sets with an 80:20 ratio, model development using *Random Forest* and *Gradient Boosting* algorithms, and evaluation using *accuracy*, *precision*, *recall*, and *F1-score* metrics.The results show that the *Gradient Boosting* algorithm outperformed *Random Forest*. *Gradient Boosting* achieved an *accuracy* of 58.12% and a macro *F1-score* of 0.44, while *Random Forest* achieved an *accuracy* of 51.28% and a macro *F1-score* of 0.39. Time and location were identified as the most influential variables affecting traffic congestion levels.

**Keywords:** Traffic Congestion, *Machine Learning*, *Random Forest*, *Gradient Boosting*, *Google Traffic*.

## 1. PENDAHULUAN

Kemacetan lalu lintas merupakan salah satu permasalahan utama yang dihadapi berbagai wilayah perkotaan dan kawasan industri di Indonesia. Meningkatnya jumlah kendaraan bermotor yang tidak diimbangi dengan peningkatan kapasitas infrastruktur jalan menyebabkan penurunan kualitas pelayanan transportasi, meningkatnya waktu tempuh perjalanan, pemborosan bahan bakar, serta bertambahnya pencemaran lingkungan. Kondisi tersebut juga berdampak terhadap aktivitas ekonomi, distribusi logistik, dan produktivitas masyarakat. Kabupaten Karawang sebagai salah satu kawasan industri terbesar di Indonesia mengalami pertumbuhan mobilitas yang sangat tinggi seiring berkembangnya sektor industri, perdagangan, dan permukiman. Tingginya aktivitas kendaraan pada jam-jam tertentu menyebabkan kemacetan yang sering terjadi di berbagai ruas jalan utama, terutama pada akses menuju kawasan industri, jalan nasional, dan gerbang tol.

Menurut Zheng et al., Kemacetan lalu lintas merupakan permasalahan yang kompleks karena dipengaruhi oleh interaksi berbagai faktor pada jaringan transportasi, sehingga proses prediksi kondisi lalu lintas memerlukan analisis data yang mampu menangkap hubungan spasial dan temporal secara akurat. Permasalahan tersebut berdampak pada efisiensi sistem transportasi, aktivitas ekonomi, dan kualitas hidup masyarakat [1]. Sejalan dengan hal tersebut, Febrianto dkk. menjelaskan bahwa peningkatan jumlah penduduk serta pertumbuhan kendaraan bermotor menyebabkan tingkat kepadatan lalu lintas semakin meningkat sehingga diperlukan sistem

yang mampu melakukan prediksi kondisi lalu lintas secara lebih akurat sebagai dasar pengambilan keputusan dalam pengelolaan transportasi [2].

Perkembangan teknologi kecerdasan buatan, khususnya *Machine Learning*, memberikan peluang dalam menyelesaikan berbagai permasalahan transportasi modern. *Machine Learning* mampu mempelajari pola dari data historis sehingga dapat menghasilkan prediksi terhadap kondisi yang akan terjadi pada masa mendatang. Dalam bidang transportasi, teknologi ini telah banyak dimanfaatkan untuk memprediksi tingkat kepadatan lalu lintas, memperkirakan waktu tempuh perjalanan, hingga mendukung sistem transportasi cerdas (Intelligent Transportation System). Penelitian Vindua dkk. menunjukkan bahwa penerapan algoritma Machine Learning, khususnya Random Forest, mampu memanfaatkan data historis lalu lintas untuk membangun model prediksi tingkat kemacetan secara efektif [3].

Di antara berbagai algoritma *Machine Learning* yang digunakan dalam prediksi lalu lintas, algoritma *Random Forest* dan *Gradient Boosting* merupakan dua metode *ensemble learning* yang banyak digunakan karena memiliki kemampuan klasifikasi yang baik pada data yang kompleks. *Random Forest* bekerja dengan membangun sejumlah besar decision tree secara paralel menggunakan teknik bagging sehingga mampu meningkatkan stabilitas model dan mengurangi risiko overfitting. Penelitian Kurniawan dkk,.menunjukkan bahwa algoritma Random Forest mampu menghasilkan akurasi klasifikasi hingga 96% pada prediksi tingkat kemacetan lalu lintas di Kota Bandung, sehingga efektif digunakan dalam sistem prediksi kemacetan perkotaan [4].

Sementara itu, *Gradient Boosting* membangun model secara bertahap dengan memperbaiki kesalahan prediksi pada model sebelumnya melalui pendekatan boosting. Proses pembelajaran secara bertingkat tersebut memungkinkan model menghasilkan prediksi yang lebih akurat terutama pada data yang memiliki hubungan nonlinier. Menurut Weng et al., algoritma *Extreme Gradient Boosting* (XGBoost) menerapkan proses *boosting* secara bertahap untuk memperbaiki kesalahan prediksi pada setiap iterasi, sehingga mampu meningkatkan akurasi model dalam memprediksi kondisi lalu lintas perkotaan [5]. Selain itu, Lisanthoni dkk. menyatakan bahwa algoritma *boosting* seperti *XGBoost* mampu memberikan performa yang sebanding bahkan lebih baik dibandingkan *Random Forest* pada beberapa kasus prediksi lalu lintas [6]. Oleh karena itu, kedua algoritma tersebut menarik untuk dibandingkan pada permasalahan prediksi kemacetan lalu lintas.

Berbagai penelitian sebelumnya telah membuktikan bahwa metode *ensemble learning* mampu menghasilkan performa yang lebih baik dibandingkan algoritma klasifikasi tunggal. Sinaga dkk. menjelaskan bahwa penerapan *Machine Learning* pada sistem transportasi dapat meningkatkan efisiensi pengelolaan lalu lintas melalui kemampuan memprediksi tingkat kepadatan kendaraan secara lebih akurat [8]. Penelitian Huizen juga menunjukkan bahwa *Random Forest* memperoleh nilai akurasi yang lebih tinggi dibandingkan Support Vector Machine dan Decision Tree dalam sistem pengelolaan lalu lintas [9]. Di sisi lain, penelitian Khairi dkk. mengenai XGBoost menunjukkan bahwa metode boosting memiliki kemampuan yang sangat baik dalam menangani data berskala besar dan kompleks [11]. Temuan serupa juga disampaikan oleh Tribuana dkk. yang menyatakan bahwa algoritma berbasis boosting memiliki kemampuan prediksi yang tinggi pada berbagai jenis *dataset* [12].

Meskipun demikian, penelitian yang secara khusus membandingkan performa *Random Forest* dan *Gradient Boosting* menggunakan data kondisi lalu lintas di Kabupaten Karawang masih sangat terbatas. Sebagian besar penelitian terdahulu menggunakan *dataset* lalu lintas dari kota besar atau memanfaatkan data sensor lalu lintas, sedangkan karakteristik kemacetan di Kabupaten Karawang dipengaruhi oleh aktivitas kawasan industri, distribusi logistik, mobilitas pekerja, serta akses menuju jalan tol yang memiliki pola berbeda dibandingkan wilayah perkotaan lainnya. Hidayatullah dan Sari juga menjelaskan bahwa beberapa ruas jalan di Kabupaten Karawang memiliki tingkat kerawanan kemacetan yang tinggi akibat aktivitas sekolah, kawasan industri, dan tingginya volume kendaraan [7].

Selain itu, perkembangan layanan peta digital seperti *Google Maps* menyediakan informasi kondisi lalu lintas secara *real-time* melalui indikator warna (traffic layer). Informasi tersebut dapat dimanfaatkan sebagai sumber data sekunder untuk membangun model prediksi kemacetan. Sintong dkk. menjelaskan bahwa pemanfaatan peta digital mampu membantu proses identifikasi titik kemacetan sehingga dapat dijadikan dasar dalam analisis lalu lintas [14]. Oleh karena itu, penggunaan data *Google Traffic* dipandang memiliki potensi sebagai sumber data yang praktis dan mudah diperoleh dalam penelitian prediksi kemacetan.

Berdasarkan uraian tersebut, penelitian ini dilakukan untuk membandingkan performa algoritma *Random Forest* dan *Gradient Boosting* dalam memprediksi tingkat kemacetan lalu lintas di Kabupaten Karawang menggunakan data *Google Traffic*. Perbandingan dilakukan berdasarkan metrik evaluasi *Accuracy*, *Precision*, *Recall*, dan *F1-score* sehingga dapat diketahui algoritma yang memberikan performa terbaik. Selain itu, penelitian ini juga menganalisis *feature importance* untuk mengetahui variabel yang paling berpengaruh terhadap tingkat kemacetan. Hasil penelitian diharapkan dapat memberikan kontribusi bagi pengembangan sistem prediksi lalu lintas berbasis *Machine Learning* serta menjadi referensi bagi pemerintah daerah dan instansi terkait dalam mendukung pengambilan keputusan pada pengelolaan transportasi yang lebih efektif.

# 2. METODE PENELITIAN

## 2.1 Jenis dan Pendekatan Penelitian

Pendekatan penelitian menggunakan kuantitatif eksperimen. Hardani dkk. menjelaskan bahwa penelitian kuantitatif adalah macam-macam penelitian yang detailnya sistematis, dan terencana sejak awal jelas hingga akhir desain penelitian [18]. Metode ini diterapkan mengingat fokus utama penelitian bertumpu pada analisis dan pengolahan data numerik, hasil observasi lapangan serta evaluasi kinerja model menggunakan metrik statistik yang terukur dan objektif [18].

Penelitian ini menggunakan metode eksperimen untuk mengevaluasi pengaruh penerapan dua algoritma *Machine Learning*, yaitu *Random Forest* dan *Gradient Boosting*, terhadap hasil prediksi tingkat kemacetan. Metode eksperimen memungkinkan peneliti membandingkan kinerja kedua algoritma pada dataset yang sama menggunakan skenario pengujian yang terkontrol sehingga perbedaan performa model dapat dianalisis secara objektif [13]. Sejalan dengan hal tersebut, Adlini dkk. menjelaskan bahwa studi kuantitatif eksperimental berfokus pada pembuktian hubungan sebab-akibat antarvariabel melalui penerapan perlakuan yang dikendalikan secara ketat [19].

Penggunaan metode eksperimen bertujuan untuk membandingkan efektivitas kedua algoritma secara objektif. Parameter utama model disesuaikan berdasarkan karakteristik masing-masing algoritma dengan *random_state* yang sama untuk menjaga reproduktibilitas hasil. Dengan demikian, metodologi kuantitatif-eksperimen ini memungkinkan pengujian hubungan kausal antara jenis algoritma dan tingkat akurasi prediksi kemacetan.

## 2.2 Lokasi dan Waktu Penelitian

Pengambilan data dalam penelitian ini dilakukan di Kabupaten Karawang, Jawa Barat dengan fokus pada ruas jalan nasional dan daerah yang memiliki tingkat kerawanan kemacetan tinggi. Pemilihan lokasi difokuskan pada 8 titik pengamatan strategis. Seperti pada tabel 1. Kedelapan titik tersebut dipilih secara *purposive sampling* karena memiliki volume kendaraan harian tinggi, berada di dekat kawasan industri dan pemukiman padat, serta sering dilaporkan mengalami kemacetan pada jam sibuk.

**Tabel 1.** Titik Pengamatan Penelitian

| No | Nama Jalan | Kecamatan | Karakteristik |
|---|---|---|---|
| 1 | Jl. Raya Kosambi | Klari | Jalur industri dengan volume kendaraan berat yang tinggi |
| 2 | Jl. Raya Klari–Gintungkerta | Klari | Akses utama menuju kawasan pabrik dan pergudangan |
| 3 | Jl. Jenderal Ahmad Yani | Karawang Barat | Pusat kota dengan aktivitas perdagangan dan perkantoran |
| 4 | Simpang Jomin | Kota Baru | Persimpangan utama dengan arus lalu lintas padat |
| 5 | Jl. Husni Hamid | Karawang Barat | Kawasan permukiman dan pusat aktivitas masyarakat |
| 6 | Jl. Raya Telukjambe | Telukjambe Timur | Kawasan industri dengan mobilitas kendaraan tinggi |
| 7 | Jl. Pantura | Telukjambe Timur | Jalan nasional dengan arus kendaraan antarkota dan logistik |
| 8 | Gerbang Tol Karawang Barat | Karawang Barat | Akses keluar-masuk jalan tol dengan kepadatan kendaraan tinggi |

**Sumber:** Hasil observasi dan identifikasi lokasi penelitian (2026).

Adapun waktu pelaksanaan penelitian berlangsung selama satu bulan, terhitung sejak April 2026 hingga Mei 2026. Tahapan penelitian dibagi menjadi tiga fase utama. Fase pertama adalah pengumpulan data observasi daring melalui *Google Traffic* yang dilaksanakan pada 24 April hingga 5 Mei 2026. Observasi difokuskan pada jam sibuk pukul 06.00–09.00 WIB dan 12.00–20.00 WIB setiap hari untuk menangkap pola kemacetan puncak. Sebanyak 585 data berhasil dikumpulkan. Fase kedua meliputi pra-pemrosesan data, *Feature Engineering*, dan pelatihan model algoritma *Random Forest* serta *Gradient Boosting* yang dilaksanakan pada Mei 2026. Fase ketiga adalah pengujian model, evaluasi kinerja menggunakan *Confusion Matrix* dan *classification report*, serta analisis *feature importance* yang dilakukan pada Mei 2026.

## 2.3 Sumber dan Teknik Pengumpulan Data

Penelitian ini menggunakan data sekunder yang diperoleh melalui observasi digital menggunakan fitur ***Traffic Layer*** pada *Google Maps*. Pemanfaatan peta digital dipilih karena mampu menyajikan informasi kondisi lalu lintas secara *real-time* sehingga efektif digunakan untuk mengidentifikasi tingkat kemacetan pada ruas jalan [14]. Pengumpulan data dilakukan pada delapan titik lokasi rawan kemacetan di Kabupaten Karawang selama periode April–Mei 2026. Observasi dilaksanakan pada jam sibuk, yaitu pukul 06.00–09.00 WIB dan 12.00–20.00 WIB sehingga diperoleh sebanyak **585 data observasi** yang digunakan sebagai *dataset* penelitian.

Proses pengumpulan data dilakukan dalam dua tahap. Tahap pertama adalah observasi digital dengan mengambil tangkapan layar kondisi lalu lintas pada setiap lokasi penelitian menggunakan *Google Maps*. Tahap kedua berupa dokumentasi hasil observasi ke dalam Google Sheets berdasarkan indikator warna lalu lintas yang ditampilkan *Google Maps*. Warna lalu lintas kemudian diklasifikasikan menjadi empat kategori, yaitu **Lancar (Hijau)**, **Sedang (Oranye)**, **Macet (Merah)**, dan **Lebih Macet (Merah Tua)** [14].

Sebelum proses pemodelan, seluruh data melalui tahap *data cleaning* untuk menyeragamkan penulisan label dan menghilangkan inkonsistensi data. *Dataset* terdiri atas variabel tanggal, hari, waktu, nama jalan, kecamatan, koordinat geografis, cuaca, hari libur, aktivitas sekitar, warna lalu lintas *Google Maps*, dan tingkat kemacetan sebagai variabel target. Dataset dibagi menjadi data latih dan data uji dengan rasio **80:20**, yang merupakan salah satu praktik umum dalam pengembangan model *machine learning*. Pembagian data ini bertujuan agar model dievaluasi menggunakan data yang tidak digunakan selama proses pelatihan sehingga kemampuan generalisasi model dapat dinilai secara lebih objektif. Pada skenario pertama seluruh atribut digunakan sebagai variabel masukan, sedangkan pada skenario kedua atribut **warna lalu lintas *Google Maps*** dihilangkan untuk menghindari *data leakage* sehingga kemampuan prediksi model dapat dibandingkan secara lebih objektif.

**Tabel 2.** Variabel Penelitian

| Variabel | Keterangan |
|---|---|
| Tanggal | Tanggal pengamatan |
| Hari | Hari observasi |
| Waktu | Waktu pengamatan (ditransformasikan menjadi waktu desimal) |
| Nama Jalan | Lokasi pengamatan |
| Kecamatan | Wilayah administrasi |
| Latitude, Longitude | Koordinat geografis |
| Cuaca | Kondisi cuaca saat observasi |
| Hari Libur | Status hari libur nasional |
| Aktivitas Sekitar | Karakteristik lingkungan |
| Warna Traffic *Google Maps* | Digunakan pada skenario pertama |
| Tingkat Kemacetan | Variabel target (Lancar, Sedang, Macet, Lebih Macet) |

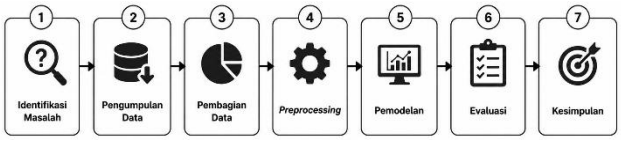
**Tabel 3.** Data Kemacetan Lalu Lintas

| No | Tanggal | Hari | Waktu | Nama Jalan | Kecamatan | Cuaca | Aktivitas Sekitar | Warna Traffic | Tingkat Kemacetan |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 24/04/2026 | Jumat | 06.30 | Jl Raya Kosambi | Klari | Cerah | Jalur Industri | Merah Tua | Lebih Macet |
| 2 | 24/04/2026 | Jumat | 06.31 | Jl Raya Klari Gintungkerta | Klari | Cerah | Akses Kawasan Pabrik | Merah Tua | Lebih Macet |
| 3 | 24/04/2026 | Jumat | 06.33 | Jl Jenderal Ahmad Yani | Karawang Barat | Cerah | Pusat Kota | Merah Tua | Lebih Macet |
| 4 | 24/04/2026 | Jumat | 06.37 | Simpang Jomin | Kota Baru | Cerah | Persimpangan | Oranye | Sedang |
| 5 | 24/04/2026 | Jumat | 06.41 | Jl Husni Hamid | Karawang Barat | Cerah | Pemukiman | Oranye | Sedang |
| 6 | 24/04/2026 | Jumat | 06.45 | Jl Raya Telukjambe | Telukjambe Timur | Cerah | Kawasan Industri | Oranye | Sedang |

Sumber: Observasi *Daring* pada *Google Maps Traffic Layer* (2026), (Sintong dkk., 2025)

### 2.5 Tahapan Penelitian

Prosedur penelitian dilaksanakan sistematis melalui tujuh tahapan seperti Gambar 1:




**Gambar 1.** Alur Penelitian

1. **Identifikasi Masalah.** Tahap ini bertujuan menentukan permasalahan utama yang diteliti. identifikasi masalah pada sistem transportasi cerdas sangat penting untuk menentukan pendekatan analisis data yang tepat. Selain itu, analisis awal terhadap pola kemacetan menjadi tahapan penting untuk memahami karakteristik data sehingga dapat meningkatkan efektivitas model machine learning dalam melakukan prediksi [1].
2. **Pengumpulan Data.** Peneliti mengumpulkan data melalui observasi daring dengan pengambilan dokumentasi visual terhadap situasi arus lalu lintas pada rentang waktu yang telah ditentukan.
3. **Preprocessing Data.** Tahap preprocessing meliputi data *cleaning*, transformasi data, dan *feature engineering*. *Feature engineering* merupakan salah satu tahapan penting untuk meningkatkan performa model *machine learning* [20].
4. **Pembagian Data.** *Dataset* dibagi menjadi 80% data latih dan 20% data uji. Pembagian *dataset* yang tepat dapat membantu mengurangi *overfitting* [15].
5. **Pemodelan.** Tahap pemodelan membangun model *Machine Learning* menggunakan algoritma *Random Forest* dan *Gradient Boosting*. Pemilihan algoritma *ensemble* dilakukan karena kemampuannya dalam memodelkan hubungan data yang kompleks pada bidang transportasi dan prediksi lalu lintas [16].
6. **Evaluasi Model.** Evaluasi dilakukan melalui *Confusion Matrix* dan *Feature importance*. Su et al. menjelaskan bahwa penggunaan *Confusion Matrix* dan *Feature importance* mampu memberikan interpretasi model yang lebih baik [17].
7. **Kesimpulan dan Rekomendasi.** Tahap kesimpulan dilakukan dengan menganalisis hasil komparatif kinerja *Random Forest* dan *Gradient Boosting* berdasarkan hasil evaluasi model.

### 2.6 Teknik Analisis Data

Analisis data dilakukan menggunakan pendekatan *supervised Machine Learning* berbasis *ensemble learning* dengan membandingkan algoritma ***Random Forest*** dan ***Gradient Boosting***. Kedua algoritma diimplementasikan menggunakan pustaka *Scikit-learn* pada bahasa pemrograman Python dengan parameter dasar yang sama sehingga perbandingan kinerja dapat dilakukan secara objektif.

Sebelum proses pemodelan, *dataset* melalui tahap *preprocessing* yang meliputi *data cleaning*, transformasi atribut waktu menjadi bentuk numerik (*feature engineering*), serta *One-hot encoding* terhadap variabel kategorikal. Selanjutnya *dataset* dibagi menjadi data latih dan data uji menggunakan rasio **80:20** agar model dapat dievaluasi menggunakan data yang belum pernah dipelajari sebelumnya.

Penelitian menerapkan dua skenario pengujian. **Skenario pertama** menggunakan seluruh atribut, termasuk variabel **warna lalu lintas *Google Maps***, sedangkan **skenario kedua** menghilangkan atribut tersebut untuk menghindari *data leakage*. Pendekatan ini bertujuan untuk mengetahui pengaruh atribut warna lalu lintas terhadap performa model sekaligus mengevaluasi kemampuan algoritma dalam memprediksi tingkat kemacetan berdasarkan variabel independen lainnya.

Kinerja model dievaluasi menggunakan ***Confusion Matrix*** dan **Classification Report** yang menghasilkan nilai ***Accuracy***, ***Precision***, ***Recall***, dan ***F1-score***. *Accuracy* digunakan untuk mengukur tingkat ketepatan prediksi secara keseluruhan, *Precision* menunjukkan ketepatan model dalam mengklasifikasikan setiap kelas, *Recall* mengukur kemampuan model mengenali seluruh data pada masing-masing kelas, sedangkan *F1-score* digunakan sebagai ukuran keseimbangan antara *Precision* dan *Recall* [23].

Selain evaluasi performa, penelitian ini juga menerapkan analisis ***Feature importance*** untuk mengidentifikasi variabel yang memberikan kontribusi terbesar terhadap hasil prediksi. Analisis tersebut membantu menjelaskan faktor-faktor yang paling berpengaruh terhadap tingkat kemacetan sehingga model yang dihasilkan tidak hanya memiliki tingkat akurasi yang baik, tetapi juga lebih mudah diinterpretasikan [17].

### 2.7 Parameter Model

Algoritma Random Forest dan Gradient Boosting diimplementasikan menggunakan *library Scikit-learn*. Seluruh parameter model dibiarkan pada nilai *default* untuk menjaga objektivitas perbandingan. Parameter utama: *n_estimators=100, random_state=42, max_depth=3* untuk *Gradient Boosting*, ditunjukkan pada Tabel 4 dan Tabel 5.

**Tabel 4.** Parameter Klasifikasi *Random Forest*

| Parameter | Nilai | Deskripsi |
|---|---|---|
| n_estimators | 100 | Jumlah pohon keputusan dalam *Random Forest*. |
| criterion | gini | Fungsi untuk mengukur kualitas pemisahan node (Gini impurity). |
| max_depth | None | Ekspansi simpul hingga seluruh daun murni atau homogen. |
| min_samples_split | 2 | Batas paling sedikit sampel dalam rangka memecah simpul internal. |

| | | |
|---|---|---|
| min_samples_leaf | 1 | Kuantitas sampel minimal yang wajib tersedia pada simpul daun. |
| random_state | 42 | Nilai penjamin reprodusibilitas hasil pengujian keacakan. |
| *n_jobs* | -1 | Jumlah *CPU* yang digunakan untuk komputasi paralel. Nilai -1 |

**Tabel 5.** Parameter Klasifikasi *Gradient Boosting*

| Parameter | Nilai | Deskripsi |
|---|---|---|
| *n_estimators* | 100 | Jumlah stage *boosting* sekuensial yang akan dijalankan. |
| *learning_rate* | 0.1 | Tingkat pembelajaran penyusutan kontribusi tiap pohon baru. |
| *max_depth* | 3 | Pembatasan kedalaman pohon regresi dasar untuk menekan overfitting. |
| *min_samples_split* | 2 | Ambang paling sedikit data dalam rangka memecah simpul |
| *min_samples_leaf* | 1.0 | Kuantitas sampel minimal yang wajib tersedia pada simpul daun. |
| *subsample* | 1.0 | Fraksi sampel yang digunakan untuk mencocokkan setiap pembelajar dasar. |
| *random_state* | 42 | Mengontrol keacakan untuk inisialisasi. Nilai 42 digunakan agar hasil penelitian dapat direplikasi. |

### 2.8 Teknik Evaluasi Model

Evaluasi model klasifikasi dilakukan untuk mengetahui seberapa baik model dalam memprediksi data berdasarkan *Confusion Matrix* menggunakan metrik *accuracy* (1), *precision* (2), *recall* (3), *dan F1-score* (4), [23]. *Feature importance* digunakan guna mengidentifikasi variabel-variabel yang memiliki signifikansi tertinggi terhadap prediksi kemacetan [17]. Pada penelitian *multiclassi* ini, *nilai precision, recall,* dan *F1-score* dihitung menggunakan pendekatan *Macro Average*.

1. ***Accuracy***
   Menghitung tingkat presisi global model dalam memprediksi keseluruhan sampel.

$$\boldsymbol{Accuracy = \frac{TP + TN}{TP + TN + FP + FN}} \quad (1)$$

2. ***Precision***
   Menunjukkan ketepatan model dalam memprediksi kelas positif. Penting untuk meminimalkan kesalahan saat model memprediksi macet padahal tidak macet.

$$\boldsymbol{Precision = \frac{TP}{TP + FP}} \quad (2)$$

3. ***Recall***
   Merepresentasikan rasio keberhasilan sistem dalam mengidentifikasi totalitas data positif. Penting agar model tidak melewatkan kondisi macet.

$$\boldsymbol{Recall = \frac{TP}{TP + FN}} \quad (3)$$

4. ***F1-score***
   Menghitung nilai rata-rata harmonis dari *precision* dan *recall*. Dipakai saat terjadi ketidakseimbangan kelas antara data macet dan tidak macet.

$$\boldsymbol{F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}} \quad (4)$$

**Keterangan:**
TP = *True Positive*: Sampel macet yang diprediksi benar sebagai macet.
TN = *True Negative*: Sampel tidak macet yang diprediksi benar sebagai tidak macet.
FP = *False Positive*: Sampel tidak macet yang diprediksi keliru sebagai macet.
FN = *False Negative*: Sampel macet yang diprediksi keliru sebagai tidak macet.

### 2.9 Alat dan Perangkat Penelitian

Penelitian ini menggunakan *Python* 3.10 yang dijalankan pada *Google Colaboratory, library Pandas 2.0, Scikit-learn 1.3, Matplotlib dan Seaborn.* Perangkat keras: Laptop *Intel Core i5,* RAM 8GB, *Windows 11*.

# 3. HASIL DAN PEMBAHASAN

## 3.1 Gambaran Umum Penelitian

Penelitian ini bertujuan membangun model prediktif kemacetan lalu lintas di Kabupaten Karawang melalui analisis perbandingan kinerja dua metode *ensemble Machine Learning*, yakni *Random Forest* dan *Gradient Boosting*. Fokus utama penelitian terletak pada pengujian kemampuan kedua algoritma dalam mengklasifikasikan kondisi lalu lintas ke dalam empat kelas, yaitu Lancar, Sedang, Macet, dan Lebih Macet, berdasarkan data yang diperoleh dari fitur *traffic layer Google Maps* [14].

Latar belakang dilakukannya penelitian ini berangkat dari permasalahan tingginya volume kendaraan di Kabupaten Karawang sebagai kawasan industri dan pemukiman yang terus berkembang. Kondisi tersebut menyebabkan kemacetan pada titik-titik strategis seperti Simpang Jomin, akses Gerbang Tol Karawang Barat, dan jalan arteri di sekitar kawasan industri. Namun, sistem manajemen lalu lintas yang ada saat ini masih berfokus pada penanganan setelah masalah terjadi dan belum memanfaatkan pendekatan prediksi yang didukung oleh data [2].

Secara metodologis, penelitian ini dilaksanakan melalui empat tahapan utama. Tahap pertama adalah pengumpulan data yang dilakukan menggunakan metode observasi daring melalui *Google Maps* pada delapan titik rawan macet di Kabupaten Karawang selama periode April hingga awal Mei 2026. Total data yang berhasil dikumpulkan sebanyak 585 data, mencakup variabel tanggal, jam, hari, nama jalan, kecamatan, dan status kemacetan berdasarkan indikator visual warna *Google Maps Traffic* [14]. Tahap kedua merupakan proses pra-pemrosesan data yang meliputi pembersihan data, standarisasi label tingkat kemacetan sehingga diperoleh empat kelas akhir, serta transformasi seluruh variabel kategorikal menggunakan teknik *One-hot encoding* [24]. Selanjutnya, dataset dibagi menjadi data latih (80%) dan data uji (20%) menggunakan *stratified sampling* untuk mempertahankan proporsi distribusi kelas pada kedua *subset* [15].

Tahap ketiga merupakan proses pemodelan yang dilakukan dengan melatih algoritma *Random Forest* dan *Gradient Boosting* menggunakan parameter *default* dari *library Scikit-learn* untuk menjaga objektivitas perbandingan. Dalam penelitian ini, diuji dua skenario: skenario pertama menggunakan seluruh fitur termasuk variabel warna lalu lintas, dan skenario kedua tanpa fitur tersebut untuk menghindari *data leakage*. Tahap keempat adalah proses evaluasi dilakukan dengan membandingkan metrik *accuracy*, *precision*, *recall*, dan *F1-score* pada data uji, serta menganalisis *Confusion Matrix* dan tingkat kepentingan fitur *feature importance* [17], [23].

Hasil awal penelitian menunjukkan adanya fenomena *data leakage* pada skenario pertama, dimana model *Gradient Boosting* mencapai akurasi mendekati 100%. Hal ini mengindikasikan bahwa fitur warna lalu lintas memiliki korelasi yang terlalu kuat dengan label target sehingga tidak realistis untuk digunakan dalam sistem prediksi dunia nyata. Oleh sebab itu, pembahasan utama pada bab ini akan difokuskan pada hasil skenario kedua tanpa fitur warna, karena lebih merepresentasikan kondisi aktual dimana prediksi harus dilakukan berdasarkan faktor-faktor penyebab kemacetan.

## 3.2 Hasil *Data Cleaning* dan Distribusi Kelas

Pada *dataset* awal yang berjumlah 585 data, ditemukan inkonsistensi penulisan label seperti Macet parah, Macet Parah, dan Macet dengan spasi di akhir. Setelah dilakukan proses standarisasi, diperoleh 4 kelas akhir dengan distribusi sebagai berikut:

### 3.2.1 Distribusi Kelas Setelah Pembersihan

Hasil distribusi kelas tingkat kemacetan setelah proses pembersihan disajikan pada Tabel 6.

**Tabel 6.** Distribusi Kelas Tingkat Kemacetan

| Label Kelas | Jumlah Data | *Persentase* |
|---|---|---|
| Lancar | 273 | 46.7% |
| Sedang | 174 | 29.7% |
| Macet | 113 | 19.3% |
| Lebih Macet | 25 | 4.3% |
| **Total** | **585** | **100%** |

Sumber: Hasil Pengolahan Data, 2026

Tabel 6 menunjukkan *dataset* bersifat *imbalanced* dengan kelas Lebih Macet hanya 4.3%. Sedangkan kelas Lancar mendominasi 46.7%. Ketidakseimbangan ini berpotensi menyebabkan model bias terhadap kelas mayoritas. Untuk mengatasi bias, proses *train-test split* dilakukan dengan menggunakan parameter *stratify=y* untuk menjaga keseimbangan proporsi setiap kelas pada data latih dan data uji.

### 3.2.2 *Feature Engineering* dan *Encoding*

Fitur Waktu yang semula dipresentasikan dalam format teks HH.MM diubah menjadi Waktu_Desimal menggunakan persamaan Jam + Menit/60 agar dapat diproses sebagai data numerik kontinu oleh model. Seluruh fitur kategorikal seperti Hari, Kecamatan, Cuaca, Aktivitas_sekitar, dan Hari_libur, kemudian dikonversi menggunakan teknik *One-hot encoding* sehingga dapat diproses oleh algoritma *Machine Learning* [21].

Setelah proses *encoding*, jumlah fitur prediktor yang digunakan pada skenario tanpa Warna_*traffic_google*_maps adalah sebanyak 52 fitur. Selanjutnya, *dataset* dibagi menjadi data latih sebesar 80% (468 data) dan data uji sebesar 20% (117 data) menggunakan metode *stratified sampling* untuk menjaga proporsi distribusi setiap kelas.

### 3.3 Hasil Pelatihan dan Pengujian Model

Pengujian dilakukan dengan dua skenario. Skenario pertama menggunakan seluruh fitur termasuk Warna_*traffic_google_maps* menghasilkan akurasi 100% pada *Gradient Boosting* yang mengindikasikan *data leakage*. Oleh karena itu, skenario kedua tanpa fitur tersebut dijadikan fokus pembahasan utama karena lebih merepresentasikan kondisi nyata di lapangan.

### 3.3.1 Hasil Pengujian Skenario 1

Pada skenario pertama, proses pelatihan dan pengujian model dilakukan dengan menggunakan seluruh fitur yang tersedia, termasuk variabel Warna_*traffic_google_maps* sebagai salah satu fitur prediktor. Skenario ini bertujuan untuk mengetahui kinerja model apabila seluruh informasi pada *dataset* dimanfaatkan tanpa menghilangkan fitur apa pun. Hasil evaluasi terhadap algoritma *Random Forest* dan *Gradient Boosting* pada data uji sebanyak 117 data disajikan pada Tabel 7 dan Tabel 8.

.

**Tabel 7.** Hasil *Random Forest* (dengan fitur *Warna_traffic_google_maps*)

| Kelas | *Precision* | *Recall* | *F1-score* | *Support* |
|---|---|---|---|---|
| Lancar | 0.95 | 0.98 | 0.96 | 55 |
| Macet | 0.95 | 0.95 | 0.95 | 22 |
| Lebih Macet | 1.00 | 1.00 | 1.00 | 5 |
| Sedang | 1.00 | 0.94 | 0.97 | 35 |
| ***Accuracy*** | - | - | **0.97** | **117** |
| ***Macro Avg*** | 0.98 | 0.97 | 0.97 | 117 |
| ***Weighted Avg*** | 0.97 | 0.97 | 0.97 | 117 |

*Accuracy RF* = 0.965811965

Sumber: Hasil Pengolahan Data, 2026

**Tabel 8.** *Gradient Boosting* (dengan fitur *Warna_traffic_google_maps*)

| Kelas | *Precision* | *Recall* | *F1-score* | *Support* |
|---|---|---|---|---|
| Lancar | 1.00 | 1.00 | 1.00 | 55 |
| Macet | 1.00 | 1.00 | 1.00 | 22 |
| Lebih Macet | 1.00 | 1.00 | 1.00 | 5 |
| Sedang | 1.00 | 1.00 | 1.00 | 35 |
| ***Accuracy*** | - | - | **1.00** | **117** |
| ***Macro Avg*** | 1.00 | 1.00 | 1.00 | 117 |
| ***Weighted Avg*** | 1.00 | 1.00 | 1.00 | 117 |

*Accuracy GB* = 1.00

Sumber: Hasil Pengolahan Data, 2026

Berdasarkan hasil pada Tabel 8 dan Tabel 9, algoritma *Random Forest* memperoleh akurasi sebesar 96,58%, sedangkan *Gradient Boosting* mencapai akurasi 100%. Nilai evaluasi yang sangat tinggi pada kedua model mengindikasikan bahwa fitur Warna_*traffic_google_maps* memiliki hubungan yang sangat kuat dengan label target. Kondisi tersebut menunjukkan adanya potensi data *leakage*, sehingga hasil pada skenario ini tidak dijadikan acuan utama dalam penelitian.

### 3.3.2 Hasil Pengujian Skenario 2

Untuk memperoleh hasil yang lebih representatif terhadap kondisi implementasi nyata, dilakukan skenario kedua dengan menghapus fitur Warna_*traffic_google_maps* dari *dataset*. Penghapusan fitur tersebut bertujuan menghindari terjadinya data *leakage*, sehingga model melakukan prediksi berdasarkan faktor-faktor penyebab kemacetan seperti waktu, lokasi, hari, cuaca, aktivitas sekitar, dan variabel pendukung lainnya. Hasil pengujian algoritma *Random Forest* dan *Gradient Boosting* pada skenario ini ditampilkan pada Tabel 9 dan Tabel 10.

**Tabel 9.** Hasil *Random Forest* (Tanpa Warna)

| Kelas | *Precision* | *Recall* | *F1-score* | *Support* |
|---|---|---|---|---|
| Lancar | 0.63 | 0.71 | 0.67 | 55 |
| Macet | 0.33 | 0.23 | 0.27 | 22 |
| Lebih Macet | 0.17 | 0.20 | 0.18 | 5 |
| Sedang | 0.44 | 0.43 | 0.43 | 35 |
| ***Accuracy*** | - | - | **0.51** | **117** |
| ***Macro Avg*** | 0.39 | 0.39 | 0.39 | 117 |
| ***Weighted Avg*** | 0.50 | 0.51 | 0.50 | 117 |

*Accuracy RF* = 0.5128205128205128

Sumber: Hasil Pengolahan Data, 2026

**Tabel 10.** Hasil *Gradient Boosting* (Tanpa Warna)

| Kelas | *Precision* | *Recall* | *F1-score* | *Support* |
|---|---|---|---|---|
| Lancar | 0.64 | 0.80 | 0.71 | 55 |
| Macet | 0.57 | 0.18 | 0.28 | 22 |
| Lebih Macet | 0.33 | 0.20 | 0.25 | 5 |
| Sedang | 0.50 | 0.54 | 0.52 | 35 |
| ***Accuracy*** | - | - | **0.58** | **117** |
| ***Macro Avg*** | 0.51 | 0.43 | 0.44 | 117 |
| ***Weighted Avg*** | 0.57 | 0.58 | 0.55 | 117 |

*Accuracy GB* = 0.5811965811965812

Sumber: Hasil Pengolahan Data, 2026

Berdasarkan Tabel 10 dan Tabel 11, terjadi penurunan performa pada kedua algoritma dibandingkan skenario pertama. *Random Forest* memperoleh akurasi sebesar 51.28%, sedangkan *Gradient Boosting* memperoleh akurasi sebesar 58.12%. Meskipun nilai akurasinya lebih rendah, hasil pada skenario kedua dinilai lebih realistis karena model melakukan prediksi tanpa memanfaatkan informasi yang secara langsung merepresentasikan kondisi kemacetan. Oleh karena itu, hasil pada skenario ini dijadikan dasar dalam proses analisis dan pembahasan selanjutnya.

Selain nilai *accuracy*, hasil *classification report* juga menunjukkan bahwa kedua model masih mengalami kesulitan dalam mengenali kelas minoritas, khususnya kelas Lebih Macet. Nilai *F1-score* untuk kelas tersebut hanya 0.18 pada RF dan 0.25 pada GB. Hal ini disebabkan oleh jumlah sampel kelas Lebih Macet yang sangat sedikit, yaitu hanya 5 data pada data uji. Oleh karena itu, analisis lebih lanjut terhadap *Confusion Matrix* dan *feature importance* dilakukan pada subbab berikutnya.

### 3.3.3 Perbandingan Kinerja Algoritma

Perbandingan metrik evaluasi utama kedua algoritma dirangkum pada Tabel 11.

**Tabel 11**.Perbandingan Kinerja Algoritma

| Metrik Evaluasi | *Random Forest* | *Gradient Boosting* |
|---|---|---|
| *Accuracy* | 51.28% | 58.12% |
| *Precision Macro Avg* | 0.39 | 0.51 |
| *Recall Macro Avg* | 0.39 | 0.43 |
| *F1-score Macro Avg* | 0.39 | 0.44 |

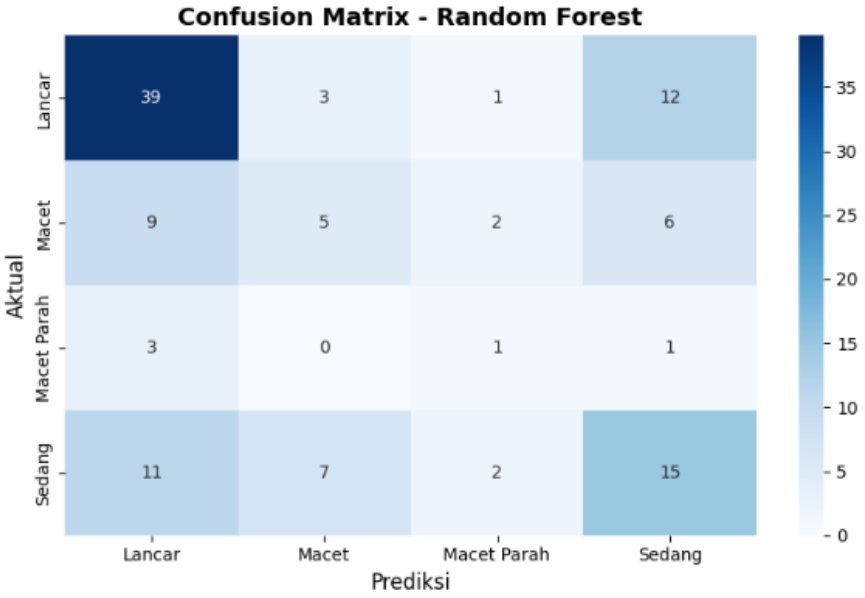
Sumber: Hasil Pengolahan Data, 2026

Berdasarkan Tabel 12, algoritma *Gradient Boosting* menunjukkan performa lebih baik dengan akurasi 58.12%, unggul 6.84% dibandingkan *Random Forest* yang memperoleh 51.28%. Nilai *F1-score Macro Avg* GB sebesar 0.44 juga lebih tinggi dari RF sebesar 0.39, menandakan GB lebih seimbang dalam memprediksi semua kelas. Keunggulan ini sejalan dengan temuan Lisanthoni dkk. yang menyatakan bahwa metode *boosting* seperti *Gradient Boosting* memiliki performa sebanding dengan *Random Forest* dalam hal efisiensi dan akurasi [6].

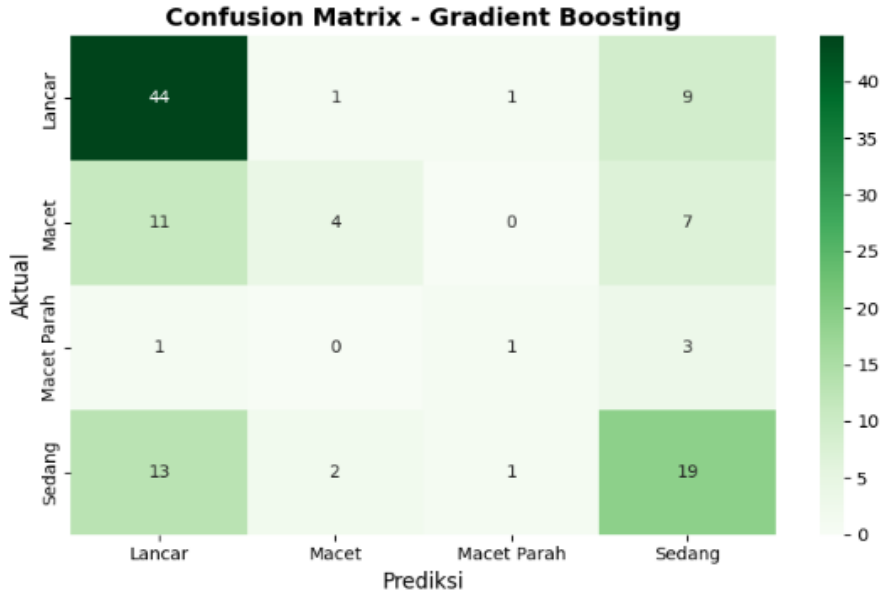
### 3.4 Hasil Evaluasi Model

#### 3.4.1 Analisis *Confusion Matrix*

Untuk mengetahui pola kesalahan klasifikasi yang dilakukan oleh masing-masing algoritma, dilakukan analisis menggunakan *Confusion Matrix*. Visualisasi *Confusion Matrix Random Forest* dan *Gradient Boosting* ditunjukkan pada Gambar 2 dan Gambar 3.




**Gambar 2.** *Confusion Matrix Random Forest*
Sumber: Hasil Pengolahan Data, 2026



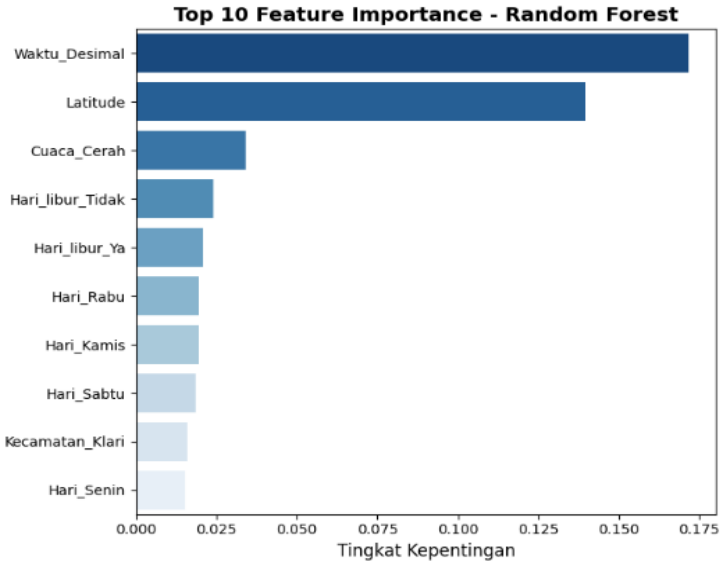

**Gambar 3.** *Confusion Matrix Gradient Boosting*
Sumber: Hasil Pengolahan Data, 2026

Berdasarkan Gambar 2, *Random Forest* berhasil memprediksi 39 dari 55 data kelas Lancar dengan benar. Namun, 9 dari 22 data kelas Macet aktual salah diklasifikasikan sebagai Lancar. Pada Gambar 3, *Gradient Boosting* memiliki performa lebih baik pada kelas Lancar dengan 44 prediksi benar. Akan tetapi, algoritma ini juga mengalami *misclassification* pada kelas Macet, dimana 11 dari 22 data diprediksi sebagai Lancar dan 4 data diprediksi sebagai Sedang.

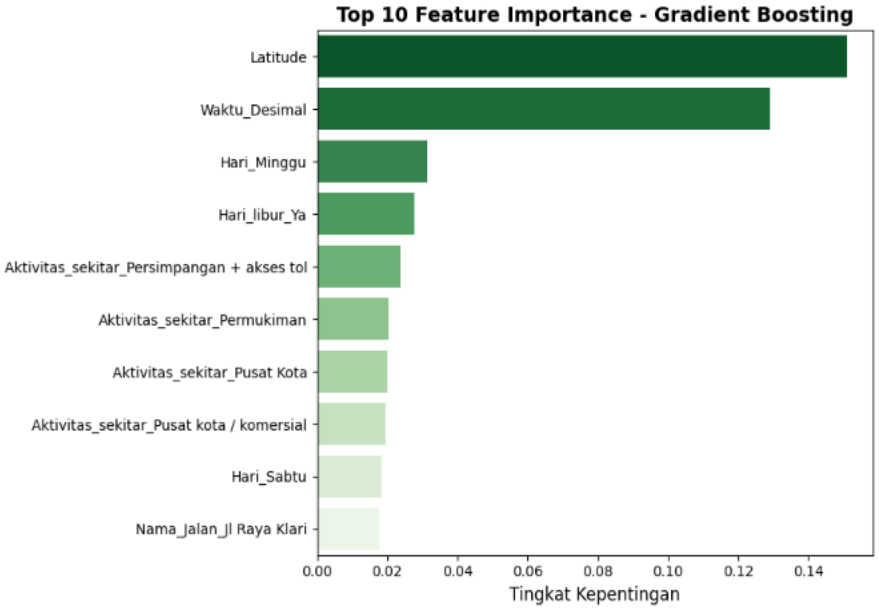
Kelas Lebih Macet menjadi yang paling sulit diprediksi kedua model karena jumlah data uji yang sangat sedikit, yaitu hanya 5 data. Pola ketidakakuratan tersebut membuktikan bahwa performa kedua model mengalami distorsi atau bias yang mengarah pada kelas dominan 'Lancar'. Hal ini merupakan konsekuensi dari *imbalanced dataset* yang telah dijelaskan pada Tabel 6. Kondisi tersebut menyebabkan model cenderung memberikan prediksi pada kelas mayoritas dibandingkan kelas minoritas.

### 3.4.2 Analisis *Feature importance*

Untuk mengidentifikasi variabel yang memiliki kontribusi terbesar terhadap hasil prediksi, peneliti melakukan analisis *feature importance* dengan hasil visualisasi yang dapat dilihat pada Gambar 4 dan Gambar 5.




**Gambar 4.** Top 10 *Feature importance Random Forest*
Sumber: Hasil Pengolahan Data, 2026




**Gambar 5.** Top 10 *Feature importance Gradient Boosting*
Sumber: Hasil Pengolahan Data, 2026

Detail nilai lima fitur terpenting disajikan melalui Tabel 12 dan Tabel 13.

**Tabel 12.** Lima Fitur Terpenting *Random Forest*

| Peringkat | Fitur | *Importance* |
|---|---|---|
| 1 | Waktu_Desimal | 0.171629 |
| 2 | Latitude | 0.139526 |
| 3 | Cuaca_Cerah | 0.034118 |
| 4 | Hari_libur_Tidak | 0.023985 |
| 5 | Hari_libur_Ya | 0.020634 |

Sumber: Hasil Pengolahan Data, 2026

**Tabel 13.** Lima Fitur Terpenting *Gradient Boosting*

| Peringkat | Fitur | Importance |
|---|---|---|
| 1 | Latitude | 0.158818 |
| 2 | Waktu_Desimal | 0.129057 |
| 3 | Hari_Minggu | 0.031353 |
| 4 | Hari_libur_Ya | 0.027666 |
| 5 | Aktivitas_sekitar_Persimpangan + akses tol | 0.023762 |

Sumber: Hasil Pengolahan Data, 2026

Berdasarkan Tabel 13 dan 14, fitur Waktu_Desimal dan *Latitude* menjadi dua prediktor paling dominan pada kedua algoritma dengan tingkat kepentingan 0.17 dan 0.15 untuk RF, serta 0.15 dan 0.13 untuk GB. Temuan ini membuktikan bahwa faktor temporal dan spasial merupakan penentu utama kemacetan di Kabupaten Karawang. Temuan ini sejalan dengan Su et al. [17] yang menunjukkan bahwa analisis feature importance mampu mengidentifikasi variabel yang paling berkontribusi terhadap hasil prediksi lalu lintas.

Perbedaan menarik terlihat pada fitur peringkat ketiga. *Random Forest* lebih sensitif terhadap Cuaca_Cerah dengan nilai 0.045, mengindikasikan bahwa kondisi cuaca berpengaruh terhadap volume kendaraan. Sementara itu, *Gradient Boosting* mengidentifikasi Hari_Minggu sebagai fitur penting dengan nilai 0.031, menunjukkan bahwa algoritma ini lebih baik dalam menangkap pola kemacetan akhir pekan yang berbeda dengan hari kerja. Selain itu, GB juga menemukan Aktivitas_sekitar_Persimpangan dan akses tol sebagai fitur signifikan, sejalan dengan kondisi geografis Karawang yang memiliki banyak persimpangan menuju kawasan industri dan pintu tol.

### 3.5 Pembahasan

Berdasarkan hasil pelatihan dan pengujian model pada subbab sebelumnya, terdapat beberapa temuan penting yang perlu dibahas secara mendalam terkait perbandingan kinerja *Random Forest* dan *Gradient Boosting* dalam prediksi kemacetan lalu lintas 4 kelas.

#### 3.5.1 Analisis Data *Leakage*

Skenario pertama dilakukan hanya sebagai analisis tambahan untuk mendeteksi kemungkinan data *leakage* dan tidak digunakan sebagai dasar pemilihan model. Hasil ini mengindikasikan terjadinya data *leakage*, yaitu kondisi ketika informasi pada fitur prediktor memiliki hubungan yang sangat kuat dengan label target sehingga model menghasilkan performa yang terlalu tinggi dan tidak merepresentasikan kondisi prediksi yang sebenarnya.

Fitur Warna_*traffic_google_maps* merupakan representasi visual langsung dari tingkat kemacetan pada *Google Maps*, sehingga penggunaannya membuat model tidak lagi melakukan prediksi, melainkan hanya mencocokkan warna dengan label. Kondisi ini tidak realistis untuk diterapkan pada sistem prediksi dunia nyata, karena dalam praktiknya sistem harus mampu memprediksi kemacetan sebelum terjadi berdasarkan faktor-faktor penyebab seperti waktu, hari, dan lokasi. Oleh karena itu, skenario kedua tanpa fitur Warna_*traffic_google_maps* dipilih sebagai fokus utama pembahasan karena lebih merepresentasikan tantangan prediksi yang sebenarnya.

#### 3.5.2 Analisis Perbandingan RF dan GB

Pada skenario kedua tanpa fitur warna, *Gradient Boosting* memperoleh akurasi 58.12%, unggul 6.84% dibandingkan *Random Forest* yang hanya mencapai 51.28% (Tabel 10 dan 11). Keunggulan *Gradient Boosting* juga terlihat pada nilai *F1-score* Macro Avg sebesar 0.44 dibandingkan *Random Forest* 0.39. Perbedaan performa ini dapat dijelaskan melalui karakteristik algoritma. Hasil penelitian sebelumnya pada data transportasi menunjukkan bahwa model *Gradient Boosting* dapat memberikan performa prediksi yang lebih baik dibandingkan *Random Forest* pada kasus tertentu [6]. Temuan serupa juga dilaporkan oleh Ramadhan et al. yang menyatakan

bahwa mekanisme *Gradient Boosting* meningkatkan performa model melalui pembelajaran bertahap, di mana setiap model baru dibangun untuk memperbaiki kesalahan prediksi dari model sebelumnya [22]. Sebaliknya, *Random Forest* memanfaatkan teknik bagging untuk mengurangi variansi model sehingga menghasilkan klasifikasi yang lebih stabil dan memiliki kemampuan generalisasi yang baik terhadap data lalu lintas [4], [16].

### 3.5.3 Dampak Imbalanced *Dataset*

Rendahnya akurasi kedua model pada skenario 2, yaitu 51-58%, disebabkan oleh karakteristik *dataset* yang sangat *imbalanced*. Berdasarkan Tabel 11 dan 12, terlihat bahwa:

1. Kelas Lancar mendominasi dengan 55 data uji dan memiliki *F1-score* tertinggi: RF 0.67 dan GB 0.71. Model mudah mengenali kelas mayoritas ini.
2. Kelas Lebih Macet hanya memiliki 5 data uji, menghasilkan *F1-score* sangat rendah: RF 0.18 dan GB 0.25. Jumlah sampel yang terlalu sedikit membuat model kesulitan mempelajari pola kelas minoritas.
3. Kelas Macet memiliki *recall* rendah: RF 0.23 dan GB 0.18. Artinya, banyak kondisi macet aktual yang gagal terdeteksi dan salah diklasifikasi sebagai Lancar atau Sedang.

Temuan ini sejalan dengan penelitian Sinaga dkk. yang menunjukkan bahwa distribusi data yang tidak seimbang menyebabkan model klasifikasi lebih cenderung memprediksi kelas mayoritas sehingga performa pada kelas minoritas menjadi lebih rendah [8]. Meskipun sudah menggunakan *stratify=y* pada *train-test split*, jumlah absolut data kelas minoritas tetap terlalu kecil untuk dipelajari secara efektif [21]. Menurut Kurniawan dkk., algoritma *Random Forest* memiliki kemampuan yang baik dalam mengklasifikasikan tingkat kemacetan lalu lintas karena mampu menghasilkan model yang akurat dan memiliki kemampuan generalisasi yang tinggi terhadap data transportasi perkotaan [4].

### 3.5.4 Keterbatasan Penelitian

Penelitian ini memiliki beberapa keterbatasan. Pertama, jumlah data hanya 585 baris dengan distribusi sangat imbalanced, terutama kelas Lebih Macet yang hanya 4.3%. Kedua, akurasi model terbaik baru mencapai 58.12%, yang belum memadai untuk diimplementasikan langsung sebagai sistem peringatan dini. Ketiga, fitur yang digunakan masih terbatas pada variabel waktu dan lokasi, belum memasukkan faktor eksternal seperti volume kendaraan aktual, kecelakaan, atau perbaikan jalan [11].

Masalah ***multiclass imbalanced*** dapat menurunkan performa model, terutama dalam mengklasifikasikan kelas minoritas. Oleh karena itu, berbagai teknik penyeimbangan data seperti ***oversampling*** menggunakan **SMOTE** banyak diterapkan untuk meningkatkan kinerja klasifikasi pada kelas minoritas [20], [15]. Berdasarkan seluruh hasil pengujian dan evaluasi yang telah dilakukan, dapat disimpulkan bahwa algoritma *Gradient Boosting* memberikan performa yang lebih baik Temuan ini konsisten dengan penelitian Weng et al. [5] yang menunjukkan bahwa mekanisme *boosting* mampu meningkatkan akurasi melalui proses koreksi kesalahan secara bertahap pada setiap iterasi. Meskipun demikian, performa kedua model masih dipengaruhi oleh kondisi *dataset* yang tidak seimbang sehingga diperlukan pengembangan lebih lanjut melalui penambahan jumlah data maupun penerapan teknik penanganan *imbalanced dataset* pada penelitian berikutnya.

## 4. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan mengenai perbandingan algoritma *Random Forest* dan *Gradient Boosting* dalam memprediksi kemacetan lalu lintas di Kabupaten Karawang, diperoleh kesimpulan sebagai berikut: (1) Proses *preprocessing* telah berhasil dilakukan pada 585 data hasil observasi *Google Maps* periode April–Mei 2026. Tahapan meliputi standarisasi label menjadi 4 kelas yaitu Lancar, Sedang, Macet, dan Lebih Macet, transformasi fitur Waktu menjadi Waktu_Desimal, serta *One-hot encoding* pada fitur kategorikal sehingga menghasilkan 52 fitur prediktor. (2) Pada skenario 1 yang menggunakan fitur Warna_traffic_google_maps, terjadi *data leakage* yang dibuktikan dengan akurasi *Gradient Boosting* mencapai 100%. Hal ini menunjukkan fitur warna memiliki korelasi langsung dengan label target sehingga tidak realistis digunakan dalam sistem prediksi dunia nyata. (3) Pada skenario 2 tanpa fitur warna, *Gradient Boosting* menunjukkan performa lebih baik dengan *accuracy* 58.12%, *Precision macro* 0.51, *Recall macro* 0.43, dan *F1-score macro* 0.44. Sementara *Random Forest* memperoleh *accuracy* 51.28% dengan *Precision macro* 0.39, *Recall macro* 0.39 dan *F1-score macro* 0,39. Selisih 6,84% menunjukkan *Gradient Boosting* menunjukkan kemampuan yang lebih baik dalam menangkap pola nonlinier pada dataset penelitian ini. (4) Kinerja kedua model masih rendah akibat *imbalanced dataset*. Kelas Lebih Macet hanya 4,3% dari total data sehingga model kesulitan mempelajari pola kelas minoritas. Hal ini terlihat dari *F1-score* kelas Lebih Macet yang hanya 0.18 pada *Random Forest* dan 0.25 pada *Gradient Boosting*. Berdasarkan analisis *Feature Importance*, Waktu_Desimal dan Latitude merupakan fitur dengan nilai *feature importance* tertinggi pada kedua model. *Random Forest* lebih sensitif terhadap faktor

cuaca dan hari libur, sedangkan *Gradient Boosting* lebih sensitif terhadap variasi hari dalam seminggu dan aktivitas di sekitar lokasi pengamatan.

# DAFTAR PUSTAKA


[1] W. Zheng, H. Yang, J. Cai, P. Wang, X. Jiang, S. S. Du, Y. Wang, and Z. Wang,"Integrating Traffic Science with Representation Learning for City-Wide Network Congestion Prediction," Information Fusion, vol. 100, Art. no. 101837, 2023, doi: 10.1016/j.inffus.2023.101837.

[2] M. F. Febrianto, A. Priyatno, H. A. Saputri, R. Amanullah, T. P. Krissella, N. I. Matondang, and others, "Prediksi Situasi Lalu Lintas Menggunakan *Machine Learning* Dengan Algoritma *K-Nearest Neighbors Classifier*," Informatik: Jurnal Ilmu Komputer, vol. 20, no. 1, 2024, doi: 10.52958/iftk.v20i1.7042.

[3] R. Vindua et al., "Prediksi kemacetan lalu lintas menggunakan algoritma Random Forest," TIN: Terapan Informatika Nusantara, vol. 6, no. 7, 2025, doi: 10.47065/tin.v6i7.8806.

[4] M. A. Angka Kurniawan, S. S. Prasetiyowati, and Y. Sibaroni, "*Prediction and Classification of Vehicle Traffic Congestion in Bandung City Using the Random Forest and K-Nearest Neighbour Algorithm*," International Journal on Information and Communication Technology (IJoICT), vol. 11, no. 2, pp. 90–110, Dec. 2025, doi:10.21108/ijoict.v11i2.9602.

[5] J. Weng, K. Feng, Y. Fu, J. Wang, and L. Mao, "*Extreme gradient boosting algorithm based urban daily traffic index prediction model: A case study of Beijing, China*," Digital Transportation and Safety, vol. 2, no. 3, pp. 220–228, 2023, doi: 10.48130/DTS-2023-0018.

[6] A. Lisanthoni, F. I. Sari, E. L. Gunawan, and C. A. Adhigiadany, "Model Prediksi Kepadatan Lalu Lintas: Perbandingan Algoritma Random Forest dan XGBoost," Pros. Semin. Nas. Sains Data, vol. 3, no. 1, pp. 296–303, 2023, doi: 10.33005/senada.v3i1.126.

[7] I. Hidayatullah and B. N. Sari, "Penentuan jalur alternatif menghindari jalan rawan macet di Kota Karawang menggunakan algoritma Dijkstra," *Jurnal Ilmiah Wahana Pendidikan*, vol. 8, no. 13, pp. 190–199, 2022.

[8] R. M. Sinaga *et al.*, "Pemanfaatan teknik *Machine Learning* dalam memprediksi kepadatan lalu lintas guna efisiensi transportasi," *PROSISKO*, vol. 12, 2025.

[9] R. R. Huizen, "Optimalisasi rekayasa lalu lintas melalui teknologi deteksi objek," *Jurnal Sistem Informatika (JSI)*, vol. 18, no. 2, p.112-120, 2024. Available: https://mail.jsi.stikom-bali.ac.id/index.php/jsi/article/view/605

[10] A. Fikri, "Prediksi kemacetan lalu lintas urban menggunakan model pembelajaran mesin dan data mobilitas real-time," *Journal of Engineering and Technological Science*, vol. 1, no. 2, pp. 61–67, 2025.https://ejournal.kalibra.or.id/index.php/jets/article/view/133/150

[11] S. A. Khairi, A. R. Adriansyah, and L. Rosyidi, "Perbandingan XGB Regressor dengan algoritma lain untuk prediksi tarif tol," Journal of Digital Business and Technology Innovation, vol. 2, no. 1, p. 128, 2024. Available: https://journal.nurulfikri.ac.id/index.php/DBESTI/article/view/1477/408

[12] D. Tribuana, Baharuddin, and A. M. Resky, "Penerapan algoritma XGBoost untuk prediksi kepuasan pelanggan pada layanan e-commerce: Studi pada *dataset* transaksi nyata," JTBC: Jurnal Teknologi dan Bisnis Cerdas, vol. 1, no. 1, pp. 50–59, 2025. Available: https://jtbcjournal.org/index.php/jtbc/article/view/5/7

[13] L. Fink, "Why and How Online Experiments Can Benefit Information Systems Research," Journal of the Association for Information Systems, vol. 23, no. 6, pp. 1333–1346, 2022, doi: 10.17705/1jais.00787.

[14] M. Sintong et al., "Analisis pemanfaatan peta digital dalam mengidentifikasi titik kemacetan di Kota Medan," Indo Green Journal, vol. 3, no. 1, pp. 18–23, 2025, doi: 10.31004/green.v3i1.90.

[15] J. A. Kamińska and J. Kajewska-Szkudlarek, "The Importance of Data Splitting in Combined NOx Concentration Modelling," Science of the Total Environment, vol. 868, Art. no. 161744, 2023, doi: 10.1016/j.scitotenv.2023.161744..

[16] N. U. Khan, M. A. Shah, C. Maple, E. Ahmed, and N. Asghar, "Traffic Flow Prediction: An Intelligent Scheme for Forecasting Traffic Flow Using Air Pollution Data in Smart Cities with Bagging Ensemble," Sustainability, vol. 14, no. 7, Art. 4164, 2022, doi: 10.3390/su14074164.

[17] Z. Su, Q. Liu, C. Zhao, and F. Sun, "*A Traffic Event Detection Method Based on Random Forest and Permutation Importance*," Mathematics, vol. 10, no. 6, Art. no. 873, 2022, doi: 10.3390/math10060873.

[18] Hardani et al., Metode Penelitian Kualitatif & Kuantitatif, Cet. 1. Pustaka Ilmu, 2022. Available: https://penerbitpustakailmu.com/product/metode-penelitian-kualitatif-kuantitatif/

[19] M. N. Adlini et al., "Metode penelitian kualitatif studi pustaka," Edumaspul: Jurnal Pendidikan, vol. 6, no. 1, pp. 974–980, 2023, doi: 10.33487/edumaspul.v6i1.3394.

[20] L. Zhang, Y. Chen, and H. Wang, "Impact of feature engineering strategies on *Gradient Boosting* models for urban congestion forecasting," Expert Systems with Applications, vol. 238, p. 121943, 2024, doi: 10.1016/j.eswa.2023.121943.

[21] M. Han, A. Li, Z. Gao, D. Mu, S. Liu, Y. Wang, and Y. Zhang, "A Survey of Multi-Class Imbalanced Data Classification Methods," *Journal of Intelligent & Fuzzy Systems*, vol. 44, no. 2, pp. 2471–2501, 2023, doi: 10.3233/JIFS-221902.

[22] A. Ramadhan, B. M. Sopha, and M. K. Ridwan,"Sensitivity Analysis of Hyperparameter in Solar Energy Prediction Model Using Gradient Boosting Method,"cASEAN Journal of Systems Engineering, vol. 9, no. 1, pp. 1–13, 2024, doi: 10.22146/ajse.v9i1.78322.

[23] Google Developers, "Classification: *Accuracy*, *recall*, *precision*, and related metrics," Google *Machine Learning* Crash Course, 2025.. Available: https://developers.google.com/machine-learning/crash-

course/classification/*accuracy-precision-recall*

[24] F. Pargent, F. Pfisterer, J. Thomas, and B. Bischl, "Regularized Target Encoding Outperforms Traditional Methods in Supervised Machine Learning with High Cardinality Features," *Computational Statistics*, vol. 37, no. 5, pp. 2671–2692, 2022, doi: 10.1007/s00180-022-01207-6.