IJAI (Indonesian Journal of Applied Informatics)
Vol. 9 No. 2 Tahun 2025 pISSN: 2548-3846, eISSN: 2598-5981

Klasifikasi Tindakan Korektif Iregularitas dengan Penerapan Algoritma CART pada Perusahaan Logistik

Juwita Stefany Hutapea[1]*, Nisa Hanum Harani[1]
[1]Program Studi D4 Teknik Informatika, Sekolah Vokasi, Universitas Logistik dan Bisnis Internasional, Indonesia
*Email: juwitastefany13@gmail.com

Info Artikel

Kata Kunci :
iregularitas, algoritma cart, tindakan korektif, logistik, CRISP-DM

Keywords :
irregularities, cart algorithm, corrective actions, logistics, CRISP-DM

Tanggal Artikel
Dikirim : 23 Januari 2025
Direvisi : 27 Mei 2025
Diterima : 31 Mei 2025

Abstrak

Perusahaan logistik memiliki peran penting dalam menjaga kelancaran rantai pasok dan distribusi barang. Namun, sering kali terjadi iregularitas dalam proses pengiriman yang dapat menghambat operasional dan menurunkan kualitas layanan. Klasifikasi tindakan korektif iregularitas bertujuan untuk mengklasifikasikan dan memprediksi tindakan korektif yang dibutuhkan dalam melakukan evaluasi iregularitas menggunakan algortima *Classification and Regression Tree (CART)* dan metodologi *CRISP-DM* untuk tahapan penelitian. Data historis iregularitas dikumpulkan dari sebuah perusahaan logistik dan diolah. Model *CART* diterapkan dengan berbagai parameter tuning untuk mendapatkan performa terbaik. Hasil evaluasi menunjukkan bahwa model yang dibangun memiliki akurasi 90%, *precision* 0.97, *recall* 0.90, dan *F1-score* 0.92. Penelitian ini membuktikan bahwa algoritma *CART* cukup baik dalam mengklasifikasikan tindakan korektif dan memprediksi tindakan yang tepat. Sistem yang dihasilkan dapat membantu meningkatkan efisiensi operasional perusahaan logistik.

Abstract

Logistics companies play a crucial role in ensuring the smooth flow of the supply chain and distribution of goods. However, irregularities in the delivery process often occur, which can disrupt operations and reduce service quality. The classification of corrective actions for irregularities aims to classify and predict the corrective actions needed in evaluating irregularities using the Classification and Regression Tree (CART) algorithm and the CRISP-DM methodology for the research phases. Historical irregularity data was collected from a logistics company and processed. The CART model was applied with various parameter tuning to achieve optimal performance. Evaluation results show that the developed model has an accuracy of 90%, with precision, recall, and F1-score values to be specified. This study demonstrates that the CART algorithm is effective in classifying corrective actions and predicting the appropriate actions. The resulting system can help improve the operational efficiency of the logistics company.

1. PENDAHULUAN

Perusahaan logistik memiliki peran penting dalam rantai pasok global dengan memastikan pengiriman barang yang tepat waktu dan efisien. Namun, dalam operasionalnya masih saja terjadi kegagalan atau ketidaksesuai layanan kinerja. Ketidaksesuaian atau kegagalan yang terjadi dalam operasional sebuah perusahaan logistik disebut sebagai Iregularitas. Iregularitas merujuk pada berbagai masalah atau penyimpangan yang ditemukan dalam proses *first mile, middle mile*, dan *last mile*. Setiap proses ini memiliki tantangan sendiri, di mana ketidaksesuaian dapat terjadi baik pada pengumpulan dan pengiriman barang maupun dalam proses pengantaran akhir ke konsumen [1]. Iregularitas tersebut meliputi kerusakan barang, kesalahan dokumentasi, ketidaksesuaian barang yang dikirim, salah salur kiriman, salah tempel resi, hingga selisih berat kiriman. Masalah-masalah ini tidak hanya merugikan pelanggan tetapi juga berpotensi menurunkan reputasi perusahaan.

Di sebuah perusahaan logistik, sebelumnya menangani iregularitas secara manual menggunakan laporan *Excel*. Metode ini memakan waktu lama, rawan kesalahan, dan kurang efisien terutama dalam menangani volume data yang besar. Keterbatasan ini sejalan dengan temuan Kerdprasop et al. (2019) yang menunjukkan bahwa proses manual dalam deteksi anomali di rantai pasok dapat menghambat respons cepat dan akurat [2]. Oleh karena itu, dibutuhkan solusi berbasis data untuk otomatisasi yang bertujuan mempercepat dan mempermudah evaluasi iregularitas secara akurat.

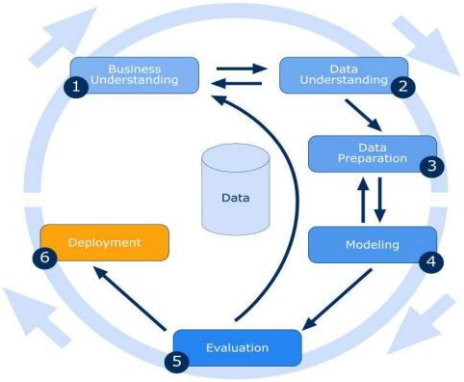
Untuk mengatasi masalah tersebut diperlukan pendekatan yang mampu mengklasifikasikan tindakan korektif secara tepat dan cepat berdasarkan jenis dan penyebab iregularitas yang terjadi. Salah satu metode yang efektif untuk tujuan ini adalah *decision tree*. *Decision tree* adalah metode analisis yang memanfaatkan struktur hierarkis menyerupai pohon, di mana setiap cabang merepresentasikan keputusan atau aturan tertentu [3]. Penelitian yang dilakukan oleh Purnomo et al. (2023) menunjukkan bahwa algoritma C4.5 dapat digunakan untuk mengklasifikasikan tren pelanggaran kendaraan angkutan barang dengan akurasi sebesar 86,34% [4]. Selain itu, Aribowo et al. (2021) menerapkan algoritma CART untuk menentukan kelayakan penerima bantuan sosial di Desa Ngadirejo dan menunjukkan bahwa decision tree mampu menghasilkan aturan klasifikasi yang jelas dan dapat digunakan sebagai dasar pengambilan keputusan [5]. Penerapan algoritma lain seperti *Naïve Bayes* dan J48 telah dilakukan untuk tujuan klasifikasi dan prediksi. Namun, CART menunjukkan keunggulan dalam hal interpretabilitas dan performa prediksi [6].

Penelitian ini mengusulkan penerapan algoritma *Classification and Regression Tree (CART)* sebagai alat klasifikasi tindakan korektif berdasarkan jenis dan penyebab iregularitas yang terjadi. Algoritma *CART* memiliki keunggulan dalam mengolah data besar, membuat keputusan berdasarkan aturan hierarkis, serta mendukung variabel kategori maupun numerik [7].

Tujuan utama penelitian ini adalah membangun model klasifikasi menggunakan algoritma *Classification and Regression Tree (CART)* untuk mengklasifikasikan tindakan korektif sehingga dapat meningkatkan efisiensi evaluasi dan mempercepat pengambilan keputusan [5][8].

2. METODE PENELITIAN

Penelitian ini menggunakan metode pengumpulan data dengan menerapkan pendekatan *CRISP-DM (Cross-Industry Standard Process for Data Mining)*, yang merupakan standar dalam data mining [9][10]. Alur proses penelitian ini ditampilkan pada Gambar 1. Setiap langkah dalam proses tersebut memiliki peran penting untuk dilaksanakan. Penjelasan detail mengenai masing-masing langkah akan dijelaskan dalam bab ini. Berikut adalah Gambar 1 yang memperlihatkan alur metodologi penelitian.



Gambar 1. Metodologi CRISP-DM [9]

2.1 *Business Understanding*

Iregularitas dalam sektor logistik merujuk pada penyimpangan proses pengiriman barang dari standar operasional seperti keterlambatan, salah tujuan, kerusakan, atau kehilangan barang. Masalah ini berdampak pada kepuasan pelanggan, reputasi perusahaan, dan efisiensi operasional [11]. Sebelumnya, Penanganan iregularitas dilakukan manual melalui laporan cabang yang dianalisis menggunakan *Excel*, namun metode ini memakan waktu, rawan kesalahan, dan kurang efektif untuk memenuhi kebutuhan logistik modern [1]. Penelitian ini menawarkan solusi berbasis data mining untuk mengidentifikasi pola iregularitas dan menentukan tindakan korektif secara efisien [12]. Dengan pendekatan ini, pengambilan keputusan menjadi lebih cepat, akurat, dan berdampak positif terhadap operasional logistik perusahaan [13].

2.2 *Data Understanding*

Tahap awal penelitian ini melibatkan pengumpulan data yang menjadi dasar untuk analisis dan pengambilan kesimpulan [9]. Penelitian ini menggunakan dataset dari historis evaluasi iregularitas di PT Pos Indonesia dengan 100 entri dan 19 fitur (Tabel 1) yang merepresentasikan informasi penting terkait kejadian iregularitas dan tindakan korektif, dan telah disintesis untuk keperluan analisis. Berikut adalah tahap-tahap data *understanding*.

Tabel 1. Dataset

Atribut	Deskripsi
ID_Sistem	Nomor unik sebagai identifikasi tiap entri dalam sistem
ZonaAsal	Regional atau zona asal tempat kiriman dikirim
Nama_Kantor_Asal	Nama lengkap kantor asal pengiriman
KantorAsal	Kode atau nama singkat kantor asal
Tanggal_Berita_Acara	Tanggal dibuatnya berita acara iregularitas
ZonaTujuan	Regional atau zona tujuan pengiriman
Nama_Kantor_Tujuan	Nama lengkap kantor tujuan pengiriman
KantorTujuan	Kode atau nama singkat kantor tujuan
Deskripsi	Keterangan umum terkait kasus iregularitas
DNLN	Status pengiriman: Domestik (DN) atau Luar Negeri (LN)
Nomor_Kiriman	Nomor resi atau nomor pengiriman barang yang terkait kasus
Uraian_Berita_Acara	Penjabaran isi berita acara secara detail
Deskripsi_Iregularitas	Penjelasan spesifik mengenai bentuk atau jenis iregularitas
Tahun_BA	Tahun kejadian atau tahun pembuatan berita acara
Atribut	**Deskripsi**
Bulan_BA	Bulan kejadian berita acara dalam format angka
Week	Minggu ke-berapa dalam tahun saat kejadian terjadi
month_name	Nama bulan dari tanggal berita acara
referensi_root_cause	Penyebab utama dari kejadian iregularitas berdasarkan analisis
corrective_action	Tindakan korektif yang diberikan untuk menangani iregularitas

1. Mengimpor dataset

```
[ ]  # Mendefinisikan path file dan membaca file CSV
     file_path = 'dummy_new.csv'
     data = pd.read_csv(file_path)
```

Keterangan : *File path* digunakan untuk mendefinisikan lokasi atau nama file CSV yang akan dibaca. Dan fungsi *pd.read_csv( )* dipanggil untuk membaca isi file menggunakan pustaka *Pandas* yang kemudian disimpan ke dalam variabel data sebagai sebuah *DataFrame*.

2. Menampilkan Informasi Data dan Data Awal

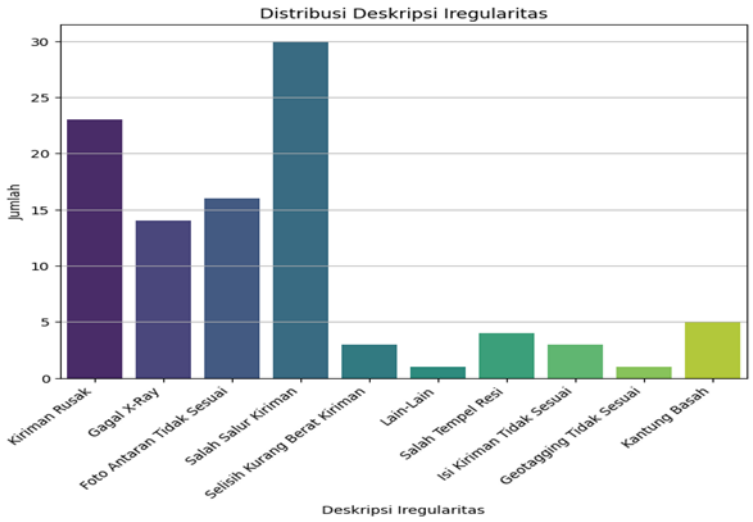
```
[ ]  # Menampilkan informasi tentang DataFrame dan lima baris gambaran awal data
     print("Info Data:")
     print(data.info())
     print("Lima Data Teratas:")
     data.head()
```

Keterangan : Kode ini menampilkan informasi tentang *DataFrame* data. Fungsi *.info ( )* memberikan gambaran umum tentang struktur data seperti jumlah baris, kolom, tipe data, dan jumlah nilai yang tidak kosong. Sementara fungsi print ("Info Data:") digunakan untuk menampilkan teks sebelum informasi dari data ditampilkan. Fungsi *.head ( )* digunakan untuk melihat sampel data teratas sebagai gambaran umum tentang isi dataset.

3. Eksplorasi Data

```
[ ]  # visualisasi distribusi Deskripsi Iregularitas
     plt.figure(figsize=(8, 6))
     sns.countplot(data=data, x='Deskripsi_Iregularitas', palette='viridis')
     plt.title('Distribusi Deskripsi Iregularitas')
     plt.xlabel('Deskripsi Iregularitas')
     plt.ylabel('Jumlah')
     plt.xticks(rotation=45, ha='right')
     plt.grid(axis='y')
     plt.show()
```

Keterangan : Dalam melakukan eksplorasi data dilakukan visualisasi distribusi nilai pada Deskripsi_Iregularitas dengan ukuran plot 8x6 dan warna palet viridis. Judul dan label sumbu X serta Y ditambahkan, sementara label pada sumbu X diputar agar lebih mudah dibaca. Grafik yang ditampilkan berupa grafik batang dengan menambahkan grid pada sumbu Y untuk membantu pembacaan nilai.



Gambar 2. Visualisasi Distribusi Deskripsi Iregularitas

Pada Gambar 2 diatas menunjukkan distribusi frekuensi berbagai jenis iregularitas yang terjadi dalam proses pengiriman. Berdasarkan visualisasi tersebut, **Kiriman Rusak** merupakan jenis iregularitas yang paling dominan, diikuti oleh **Salah Salur Kiriman** dan **Selisih Kurang Berat Kiriman**. Sebaliknya, jenis iregularitas seperti **Salah Tempel Resi** dan **Kantung Basah** memiliki frekuensi kejadian yang relatif rendah. Hasil ini memberikan informasi penting mengenai pola permasalahan yang paling sering muncul, sehingga dapat menjadi dasar dalam merumuskan prioritas perbaikan serta strategi peningkatan kualitas layanan pengiriman.

2.3 *Data Preparation*

Fase persiapan data merupakan serangkaian proses yang bertujuan untuk mempersiapkan data agar siap digunakan dalam analisis lebih lanjut. Tahapan ini mencakup beberapa proses seperti:

1. Encoding Data

```
# Encoding Kategorikal
encoded_data = data.copy()
object_columns = encoded_data.select_dtypes(include=['object']).columns
label_encoders = {}

for col in object_columns:
    le = LabelEncoder()
    encoded_data[col] = le.fit_transform(encoded_data[col])
    label_encoders[col] = le
```

Keterangan : Membuat salinan data dengan *encoded_data = data.copy()* dan kemudian mencari kolom dengan tipe data objek menggunakan *encoded_data.select_dtypes(include=['object']).columns*. Untuk setiap kolom objek, menerapkan *LabelEncoder* untuk mengubah nilai-nilai unik menjadi angka dengan *le = LabelEncoder()* dan *encoded_data[col] = le.fit_transform(encoded_data[col])*. Hasil encoding disimpan kembali pada data yang telah disalin, dan setiap *encoder* disimpan dalam *dictionary label_encoders* menggunakan *label_encoders[col] = le*, sehingga dapat digunakan kembali.

2. Uji Korelasi Semua Atribut

```
# Uji Korelasi
correlation_matrix = encoded_data.corr()
```

Keterangan : Setelah data di-*encode* dan dihitung korelasinya dengan *.corr()*, matriks korelasi menunjukkan hubungan antar kolom dengan nilai antara -1 hingga 1, di mana nilai 1 menunjukkan hubungan yang positif sempurna (kedua variabel bergerak searah), nilai -1 menunjukkan hubungan yang negatif sempurna (kedua variabel bergerak berlawanan arah), dan nilai 0 menunjukkan tidak adanya hubungan linier antara kedua variabel. Nilai ini membantu menganalisis kekuatan dan arah hubungan antar variabel. Berdasarkan hasil korelasi dan eksplorasi, seleksi atribut dilakukan untuk memilih fitur yang paling relevan dan signifikan dalam memprediksi tindakan korektif iregularitas.

3. Pemilihan Fitur dan Target

```
# Menentukan variabel X (fitur/atribut) dan Y (target/kelas)
X = encoded_data.drop(columns=['Kantor_Asal', 'Nopend_Asal', 'Kantor_Tujuan',
                               'Nopend_Tujuan', 'Deskripsi','Week', 'Tahun_BA',
                               'month_name','corrective_action'])
y = encoded_data['corrective_action']
```

Keterangan : Variabel X berisi fitur yang digunakan untuk memprediksi target yang dibuat dengan menghapus beberapa kolom fitur yang tidak diperlukan dari *encoded_data* menggunakan *.drop( )*. Variabel Y merupakan kolom target yang ingin diprediksi atau sebagai kelas.

4. Pembagian Dataset

```
# Split Data menjadi Data Latih dan Data Uji
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Keterangan : Variabel *X_train, X_test, y_train*, dan *y_test* akan menyimpan data latih dan data uji untuk fitur (X) dan target (y) secara terpisah. Fungsi *train_test_split(X, y, test_size=0.2, random_state=42)* digunakan untuk membagi dataset menjadi data latih dan uji. Di sini, X berisi fitur-fitur, dan y adalah target yang ingin diprediksi. Parameter *test_size=0.2* berarti 20% data akan digunakan sebagai data uji, sementara 80% sisanya untuk data latih. Sedangkan *random_state=42* memastikan pembagian data yang konsisten setiap kali kode dijalankan.

2.4 *Modeling*

Dalam penelitian ini, *machine learning* diterapkan untuk menentukan teknik dan algoritma yang sesuai untuk proses data mining. Model klasifikasi yang dipilih adalah *Decision Tree* dengan algoritma CART *(Classifiction and Regression Tree)*. Algoritma ini membangun struktur pohon keputusan berdasarkan kriteria *Gain Information* [13], dengan tahapan

Machine Learning digunakan secara langsung untuk menentukan teknik data mining, alat bantu data mining, dan algortima yang akan ditetapkan. Dalam penelitian ini, model klasifikasi yang digunakan adalah *Decision Tree* dengan algoritma *CART (Classification and Regression Tree)*. Algoritma *CART* membentuk sebuah pohon keputusan dengan menggunakan kriteria pemecah *gain information* [13], yang terdiri dari langkah-langkah berikut:

1. Menentukan akar (root) pohon keputusan
2. Menghitung nilai *gain information* untuk setiap kandidat atribut cabang
3. Memilih atribut dengan nilai *gain information* tertinggi sebagai atribut pemisah yang akan digunakan sebagai cabang
4. Mengulangi proses perhitungan dan pemilihan hingga mencapai daun atau *leaf node*

2.5 *Evaluation*

Fase evaluasi adalah tahap di mana hasil pemodelan *data mining* diinterpretasikan. Pada fase ini, beberapa kegiatan dilakukan, salah satunya adalah mengevaluasi akurasi yang diperoleh dari fase sebelumnya. Proses evaluasi dilakukan dengan berbagai metrik, seperti akurasi, *presisi, recall*, dan *F1-score* untuk masalah klasifikasi. Sebagai alat bantu, *confusion matrix* digunakan untuk mengevaluasi performa model klasifikasi dengan menghitung tingkat prediksi yang tepat. Penilaian kinerja model kemudian dilakukan berdasarkan perhitungan berikut:

1. Akurasi : Metrik sederhana yang menunjukkan persentase prediksi model yang benar dari total data uji. Dengan mengetahui jumlah data yang diklasifikasikan dengan tepat, tingkat akurasi dari model prediksi dapat ditentukan. Rumus akurasi tercantum pada Rumus 1.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \quad (1)$$

2. Presisi : Metrik yang mengukur seberapa banyak dari prediksi untuk suatu kelas yang benar dibandingkan dengan semua prediksi positif yang dibuat oleh model. Presisi menunjukkan akurasi model dalam memprediksi kelas positif.. Rumus presisi tercantum pada Rumus 2.

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

3. *Recall* : Metrik yang mengukur seberapa banyak data positif yang benar-benar berhasil diprediksi oleh model dibandingkan dengan semua data positif yang ada. Rumus *recall* tercantum pada Rumus 3.

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

4. *F1-Score* : Metrik kombinasi antara presisi dan recall, memberikan gambaran yang lebih baik ketika ada ketidakseimbangan antara kelas positif dan negatif. *F1-score* berguna untuk memperhatikan keseimbangan antara presisi dan *recall*. Rumus *F1-score* tercantum pada Rumus 4.

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

2.6 *Deployment*

Deployment merupakan proses mengimplementasikan model yang telah dibangun pada lingkungan produksi agar dapat memberikan prediksi yang dapat diakses oleh sistem. Tahap ini model yang telah dilatih dan diuji disimpan dalam format *joblib* kemudian diintegrasikan ke dalam sistem berbasis web *dashboard* sederhana yang dikembangkan menggunakan *Python* dan *Streamlit* untuk mendapatkan prediksi yang andal dan memaksimalkan nilai model [14][15]. Tujuan dari *deployment* ini adalah

mengembangkan antarmuka prediksi yang dapat digunakan oleh pengguna sehingga dapat memanfaatkan teknologi ini dalam evaluasi iregularitas pada perusahaan logistik.

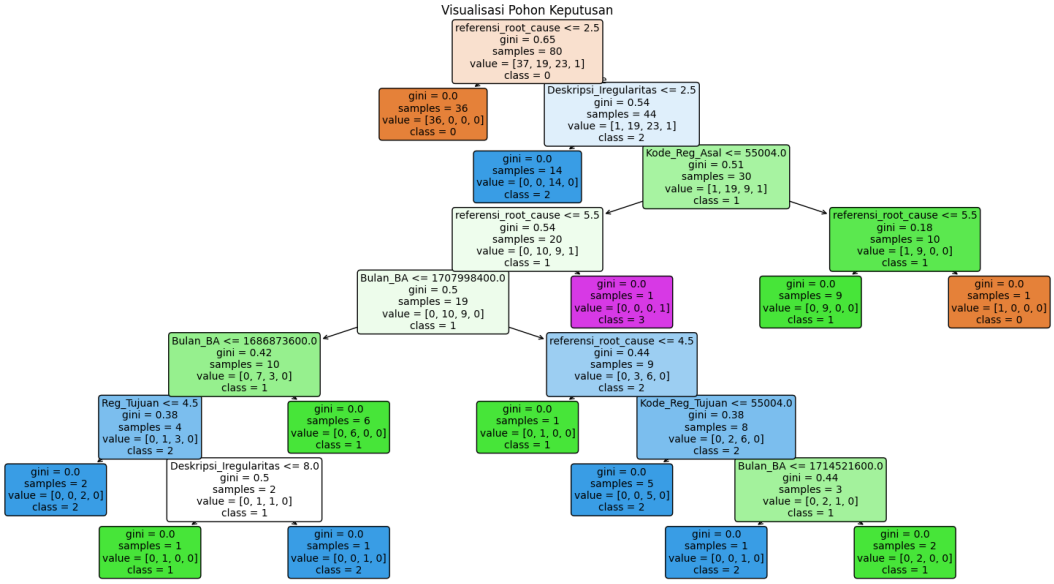
3. HASIL DAN PEMBAHASAN

Penelitian ini menghasilkan model klasifikasi menggunakan algoritma CART yang mampu memberikan prediksi Tindakan korektif iregularitas dengan akurasi yang cukup tinggi. Berdasarkan evaluasi menggunakan data uji, model mencapai akurasi sebesar 90%. Hal ini menunjukkan bahwa model memiliki kinerja yang baik dalam mengidentifikasi dan mengklasifikasikan data iregularitas. Hasil pemilihan fitur ditampilkan dalam tabel 2 berikut ini:

Tabel 2. Hasil Pemilihan Fitur

Fitur	Korelasi dengan Target	Keterangan
Reg_Asal	*-0.07*	*memiliki relevansi kontekstual*
Kode_Reg_Asal	*-0.16*	*memiliki relevansi kontekstual*
Reg_Tujuan	*-0.06*	*relevan secara domain*
Kode_Reg_Tujuan	*-0.07*	*mendukung informasi geografis*
DNLN	*0.04*	*signifikan secara domain*
Deskripsi_Iregularitas	*-0.06*	*memberikan konteks penting*
Bulan_BA	*0.11*	*untuk pengaruh temporal*
Referensi_Root_Cause	*0.71*	*penting dan relevan*

Setelah dilakukan pemilihan fitur dengan metode *Correlation Matrix* terdapat 8 fitur yang paling relevan terhadap klasifikasi. Walaupun korelasi dengan target beberapa rendah namun tetap dipertahankan karena memiliki relevansi untuk digunakan ke dalam pelatihan model. Selanjutnya *Decision Tree* dilakukan untuk memahami struktur pengambilan keputusan model, dimana atribut utama muncul pada *node* pertama sebagai faktor pemisah paling signifikan. Dapat dilihat juga visualisasi dari *Decision Tree* pada dalam Gambar 3 berikut ini:



Gambar 3. Visualisasi *Decision Tree*

Kemudian, model dievaluasi menggunakan *cross-validation* dengan membandingkan berbagai kombinasi ukuran data uji (*test_size*) dan jumlah lipatan (*fold*). Untuk *test_size* sebesar 0.2 dengan 5 *fold*, rata-rata akurasi *cross-validation* adalah 0.85 dengan standar deviasi 0.0637 yang menunjukkan stabilitas dan konsistensi model. Meskipun *test_size* 0.3 dengan 5 atau 10 *fold* menghasilkan rata-rata akurasi yang lebih tinggi, keputusan akhir didasarkan pada kinerja model pada data evaluasi. Model dengan *test_size* 0.2 dan 5 *fold* dipilih karena memberikan akurasi evaluasi lebih tinggi yaitu 90% yang menunjukkan

kemampuan model dalam memprediksi dengan lebih baik pada data baru. Perbandingan *cross-validation* ditampilkan pada Tabel 3 di bawah ini:

Tabel 3. Perbandingan Cross-Validation

Test_size	*Fold (cv)*	*Akurasi Rata-Rata*	*Standar Deviasi*	*Akurasi Evaluasi*
0.2	3	0.83	0.032	90%
	5	0.85	0.063	90%
	10	0.81	0.115	90%
0.3	3	0.89	0.021	83%
	5	0.9	0.057	83%
	10	0.9	0.091	83%
0.5	3	0.91	0.057	74%
	5	0.9	0.063	74%
	10	0.88	0.097	74%

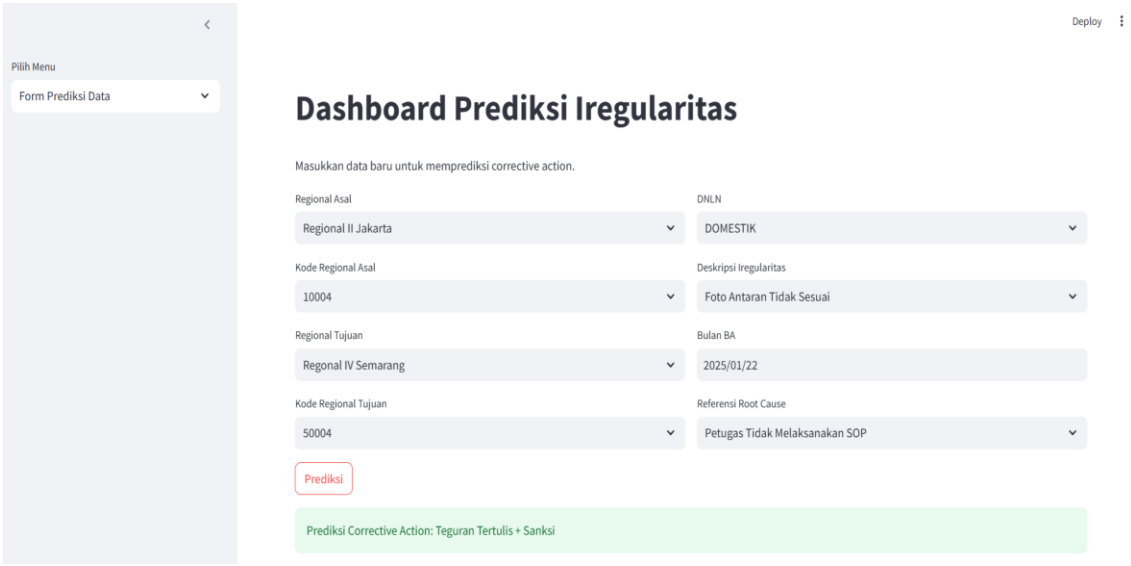
Berdasarkan hasil perbandingan cross-validation yang ditunjukkan pada Tabel 3, hasil terbaik ditunjukkan oleh kombinasi **test_size sebesar 0.2 dengan 5-fold cross-validation**, yang menghasilkan **akurasi rata-rata sebesar 0.85** dan **akurasi evaluasi tertinggi sebesar 90%** dengan standar deviasi yang relatif kecil yaitu 0.063. Hal ini menunjukkan bahwa model memberikan performa klasifikasi yang baik dan stabil pada skenario pembagian data 80:20 dan validasi silang 5 lipatan.

Kinerja model di evaluasi dengan metode *confusion matrix* untuk menganalisis performa prediksi pada data uji, menghasilkan akurasi sebesar 90%, *precision* 0.97, *recall* 0.90, dan *F1-score* 0.92, yang menunjukkan bahwa model mampu memprediksi tindakan korektif dengan baik. Hasil dari analisis performa model disajikan dalam Tabel 4, menunjukkan kinerja model pada masing-masing kelas.

Tabel 4. Performa Kinerja Model

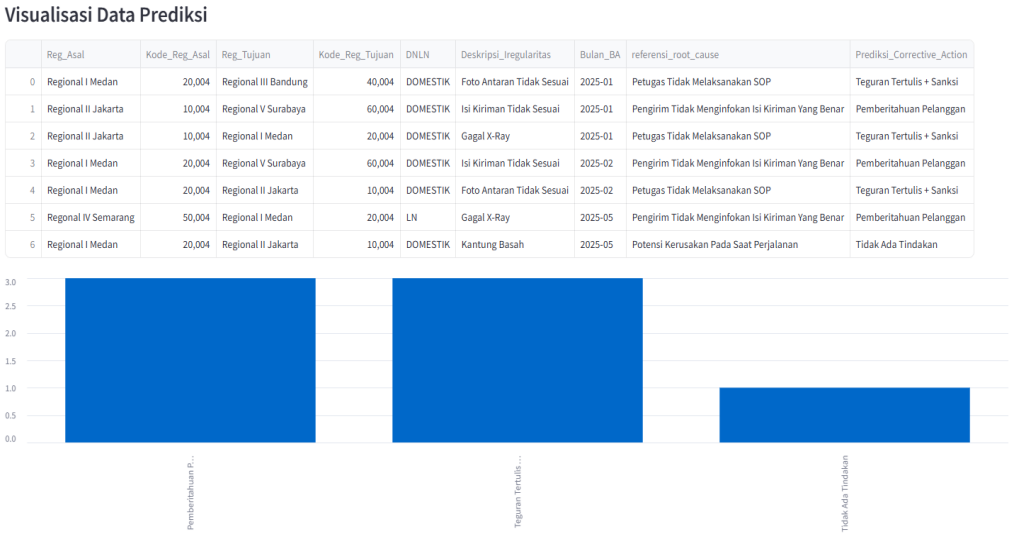
Kelas	*Precision*	*Recall*	*F1-Score*	*Support*
0	1.00	1.00	1.00	9
1	0.33	1.00	0.50	1
2	1.00	0.78	0.88	9
3	1.00	1.00	1.00	1
Accuracy	0.90			
Macro Avg	0.83	0.94	0.84	20
Weighted Avg	0.97	0.90	0.92	20

Model yang telah dibangun dan dianalisis hingga mendapatkan tingkat akurasi kinerja model sebesar 90%, kemudian diimplementasikan dalam *dashboard* sederhana yang memungkinkan pengguna untuk menginputkan data iregularitas, mendapatkan hasil prediksi secara *real-time*, serta melihat visualisasi data dan kinerja model sehingga mempercepat proses evaluasi iregularitas pada perusahaan logistik. Tampilan dari dashboard tercantum pada Gambar 4.



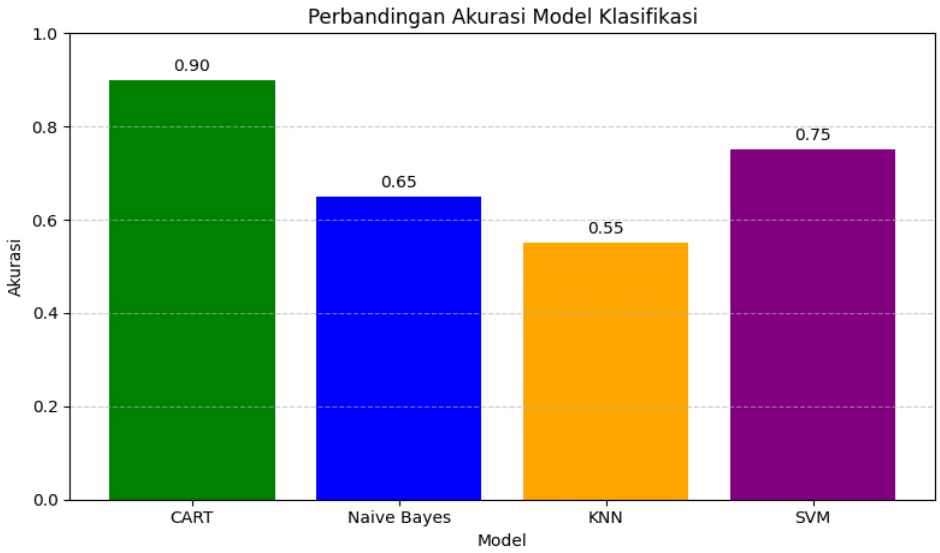
Gambar 4. Implementasi *Dashboard*

Gambar 4 menunjukkan antarmuka yang dirancang menggunakan metode klasifikasi untuk memprediksi tindakan korektif berdasarkan data yang diinput. Pengguna dapat menginputkan informasi seperti Regional Asal, Kode Regional Asal, Regional Tujuan, Kode Regional Tujuan, DNLN (Domestik/Luar Negeri), Deskripsi Iregularitas, Bulan BA dan Referensi Root Cause. Setelah data diinputkan, pengguna dapat mengklik tombol Prediksi untuk mendapatkan rekomendasi tindakan korektif. *Output* dari hasil prediksi adalah model klasifikasi yang memetakan input data ke kategori tindakan yang paling relevan, sehingga membantu pengambilan keputusan yang lebih cepat dan akurat. Dapat dilihat pada Gambar 5 berikut ini.



Gambar 5. Dashboard Hasil Prediksi

Model yang digunakan dalam penelitian ini adalah algoritma CART yang berhasil bekerja dengan membagi data ke dalam simpul-simpul berdasarkan fitur input yang paling optimal dalam memisahkan kelas target. Sebagaimana ditampilkan pada Gambar 5, model CART mampu mengklasifikasikan data ke dalam tiga kategori tindakan korektif. Hasil visualisasi tersebut menunjukkan bahwa tindakan korektif yang paling banyak direkomendasikan oleh model adalah Pemberitahuan Pelanggan, diikuti oleh Teguran Tertulis + Sanksi, dan yang paling sedikit adalah Tidak Ada Tindakan. Sebagai bagian dari proses analisis komparatif, algoritma lain seperti *Naïve Bayes*, *K-Nearest Neighbor* (K-NN), dan *Support Vector Machine* (SVM) juga diterapkan untuk tujuan klasifikasi yang serupa. Analisis ini dilakukan agar dapat diketahui optimasi dari algoritma yang digunakan, baik dari segi akurasi maupun interpretabilitas hasil.



Gambar 6. Perbandingan Akurasi Model

Meskipun masing-masing algoritma memiliki kekuatan tersendiri, hasil perbandingan yang ditunjukkan pada Gambar 6 memperlihatkan bahwa algoritma CART memiliki performa yang lebih unggul dibandingkan algoritma lainnya, menjadikannya pilihan yang tepat dalam klasifikasi tindakan korektif terhadap kasus iregularitas di sektor logistik.

4. KESIMPULAN

Penelitian ini menunjukkan bahwa penerapan algoritma CART dalam klasifikasi tindakan korektif iregularitas dapat menghasilkan model yang akurat dan andal. Dari hasil evaluasi kinerja model pada Tabel 3, model yang dikembangkan berhasil mencapai akurasi sebesar 90%, dengan tingkat *precision* 0.97, *recall* 0.90, dan *F1-score* 0.92 yang menjadikan model ini efektif dalam memprediksi tindakan korektif yang tepat untuk mengatasi iregularitas. Proses seleksi fitur dan evaluasi *cross-validation* dengan *test_size 0.2* dan 5 *fold* menunjukkan stabilitas dan konsistensi model dengan hasil yang optimal. Implementasi model dalam bentuk dashboard berbasis web memberikan solusi praktis dan efisien untuk mempermudah proses evaluasi dan pengambilan keputusan terkait iregularitas di perusahaan logistik. Dengan demikian, model ini dapat membantu penanganan iregularitas dan meningkatkan kinerja operasional perusahaan.

DAFTAR PUSTAKA

[1] N. K. Putri, "Sistem Penanganan Iregularitas Dalam Proses Kirimanpos Pada Pos Indonesia Regioanal IV Semarang," 2022.

[2] N. Kerdprasop, K. Chansilp, K. Kerdprasop, and P. Chuaybamroong, "Anomaly Detection with Machine Learning Technique to Support Smart Logistics," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2019, pp. 461–472. doi: 10.1007/978-3-030-24289-3_34.

[3] S. Firmansyah, J. Gaol, and S. B. Susilo, "Comparison of SVM and Decision Tree Classifier with Object Based Approach for Mangrove Mapping to Sentinel-2B Data on Gili Sulat, Lombok Timur," *Jurnal Pengelolaan Sumberdaya Alam dan Lingkungan*, vol. 9, no. 3, pp. 746–757, 2019, doi: 10.29244/jpsl.9.3.746-757.

[4] N. H. Purnomo, B. Pamungkas, and C. Juliane, "Penerapan Algoritma C4.5 Untuk Klasifikasi Tren Pelanggaran Kendaraan Angkutan Barang dengan Metode CRISP-DM," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 7, no. 1, p. 30, Jan. 2023, doi: 10.30865/mib.v7i1.5247.

[5] A. Aribowo, R. Kuswandhie, and Y. Primadasa, "Implementasi Algoritma CART Dalam Penentuan Implementation of the CART Algorithm in Determining the Eligibility of PKH Aid Recipients in Ngadirejo Village," *Cogito Smart Journal* |, vol. 7, no. 1, 2021.

[6] A. N. Ikhsan, A. N. Fadilah, and A. D. Iftinani, "Performance Comparison of Decision Tree J48, CART, and Naïve Bayes Algorithms for Predicting Chronic Kidney Disease," *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 7, no. 1, p. 64, Nov. 2023, doi: 10.24014/ijaidm.v7i1.26472.

[7] H. D. Darmawan, D. Yuniarti, and Y. N. Nasution, "Klasifikasi Lama Masa Studi Mahasiswa Menggunakan Perbandingan Metode Algoritma C.45 dan Algoritma Classification and Regression Tree," *Jurnal EKSPONENSIAL*, vol. 8, no. 2, 2017.

[8] Nurahman and R. Sumanti, "Klasifikasi Data Tingkat Kerawanan Kebakaran Menggunakan Algoritma CART," *Jurnal Informatika & Rekayasa Elektronika)*, vol. 7, no. 1, 2024, [Online]. Available: http://e-journal.stmiklombok.ac.id/index.php/jire

[9] M. A. Hasanah, S. Soim, and A. S. Handayani, "Implementasi CRISP-DM Model Menggunakan Metode Decision Tree dengan Algoritma CART untuk Prediksi Curah Hujan Berpotensi Banjir," *Journal of Applied Informatics and Computing (JAIC)*, vol. 5, no. 2, p. 103, 2021, [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC

[10] E. Novalia and A. Voutama, "Prediction of Rice Field Planted Area with CRISP-DM Using Classification and Regression Tree (Cart) Algorithms," *SYSTEMATICS*, vol. 5, no. 1, p. 578, 2023.

[11] Somadi and A. Muzakki Ijlal, "Proposed Repair to Minimize the Difference of Delivery Items Using Fishbone and 5W + 1H," *Jurnal Logistik Indonesia*, vol. 5, no. 2, pp. 174–182, 2021, [Online]. Available: http://ojs.stiami.ac.id

[12] A. Amrullah, I. Purnamasari, B. Nurina Sari, and A. Voutama, "Analisis Cluster Faktor Penunjang Pendidikan Menggunakan Algoritma K-Means (Studi Kasus: Kabupaten Karawang)," *Jurnal Informatika & Rekayasa Elektronika)*, vol. 5, no. 2, pp. 244–252, 2022, [Online]. Available: http://e-journal.stmiklombok.ac.id/index.php/jirehttp://e-journal.stmiklombok.ac.id/index.php/jire

[13] R. Wisnu Nugraha and R. Hadiansah, "Penerapan Data Mining Pada Data Transaksi Distribusi Untuk Menganalisa Penempatan Buku Menggunakan Algoritma Apriori (Studi Kasus PT.Duta Bandung)," *Jurnal Teknologi dan Informasi (JATI)* , vol. 7, no. 1, pp. 29–36, 2017.

[14] A. Saddam, "[Python] Streamlit for Data Mining Visualization," Medium. Accessed: May 26, 2025. [Online]. Available: https://medium.com/learning-about-data-mining/streamlit-for-data-mining-visualization-4a3efec1455b

[15] A. Tholib, *Buku Refrensi Implementasi Algoritma Machine Learning Berbasis Web dengan Framework Streamlit*, 1st ed. Probolinggo: Pustaka Nurja, 2023. [Online]. Available: https://pustakanurja.unuja.ac.id