

Optimasi Sistem Muat Barang pada Layanan Pengiriman Logistik menggunakan Machine Learning pada Perusahaan Logistik

Optimization of Cargo Loading System in Logistics Delivery Services using Machine Learning at a Logistics Companies

¹Arman Dioza*, ²Utomo Budiyanto

^{1,2}Program Studi Ilmu Komputer, Fakultas Teknologi Informasi, Universitas Budi Luhur

e-mail: 12311600205@student.budiluhur.ac.id, utomo.budiyanto@budiluhur.ac.id

(received: 14 July 2025, revised: 13 October 2025, accepted: 15 October 2025)

Abstrak

Diperlukan pendekatan berbasis data yang lebih presisi untuk mengoptimalkan estimasi *leadtime* dan meningkatkan kualitas layanan. Hal ini dilakukan untuk mengevaluasi dan mengoptimalkan akurasi *leadtime* dengan melakukan optimalisasi muat barang kedalam kontainer menggunakan data pengiriman yang mencakup panjang, lebar, tinggi, berat dan volume pada barang dan juga kapasitas muat armada kemudian diolah untuk mendapatkan optimasi proses muat dengan *Genetic Algorithm* dan dikombinasikan dengan *Random Forest* dalam penentuan *Stacking* dan *Rotation* barang. Data tersebut dianalisis menggunakan teknik CRISP-DM untuk mengidentifikasi pola, tren, dan hubungan antarvariabel yang memengaruhi optimasi penempatan barang pada peti kemas (container). Algoritma-algoritma ini dipilih karena kemampuannya dalam menangkap pola hubungan serta relevansi pada data pengiriman. Evaluasi model dilakukan dengan menggunakan metrik akurasi, *Confusion Matrix* untuk menilai performa model secara menyeluruh. Selain itu, dengan juga membandingkan hasil model berbasis *machine learning* guna mengidentifikasi peningkatan yang signifikan dalam akurasi estimasi. Hasil penelitian ini mendapatkan *Genetic Algorithm* mencapai nilai *fitness* 0.836142 skenario 1 tanpa *Random Forest*, 3.127948 skenario 2 dengan kombinasi *Random Forest* dan akurasi yang didapat *Random Forest* untuk prediksi *Stacking* 99.23 % dan untuk prediksi *Rotation* 99.33 %. Sistem yang telah dikembangkan ini mampu membantu dalam proses optimasi muat barang secara optimal dengan model prediksi yang akurat untuk mendukung pengambilan keputusan berbasis data. Dengan implementasi model ini, perusahaan logistik dapat meningkatkan efisiensi operasional, meminimalkan risiko keterlambatan, dan memberikan layanan pelanggan yang lebih unggul.

Kata kunci: *genetic algorithm*, optimasi muat barang, *leadtime*, logistik, *random forest*

Abstract

A more precise data-driven approach is required to optimize lead time estimation and improve service quality. This study aims to evaluate and enhance lead time accuracy by optimizing cargo loading into containers using shipment data, including item length, width, height, weight, and volume, as well as vehicle loading capacity. The data are processed to optimize the loading process using a Genetic Algorithm, combined with a Random Forest model for determining cargo stacking and rotation. The dataset is analyzed using the CRISP-DM methodology to identify patterns, trends, and inter-variable relationships that influence the optimization of cargo placement within containers. These algorithms were selected due to their ability to capture complex relational patterns and their relevance to logistics shipment data. Model performance is evaluated using accuracy metrics and a confusion matrix to comprehensively assess predictive performance. In addition, the results of the machine learning-based models are compared to identify significant improvements in estimation accuracy. The results indicate that the Genetic Algorithm achieved a fitness value of 0.836142 in Scenario 1 without Random Forest and 3.127948 in Scenario 2 when combined with Random Forest. Furthermore, the Random Forest model achieved an accuracy of 99.23% for stacking prediction and

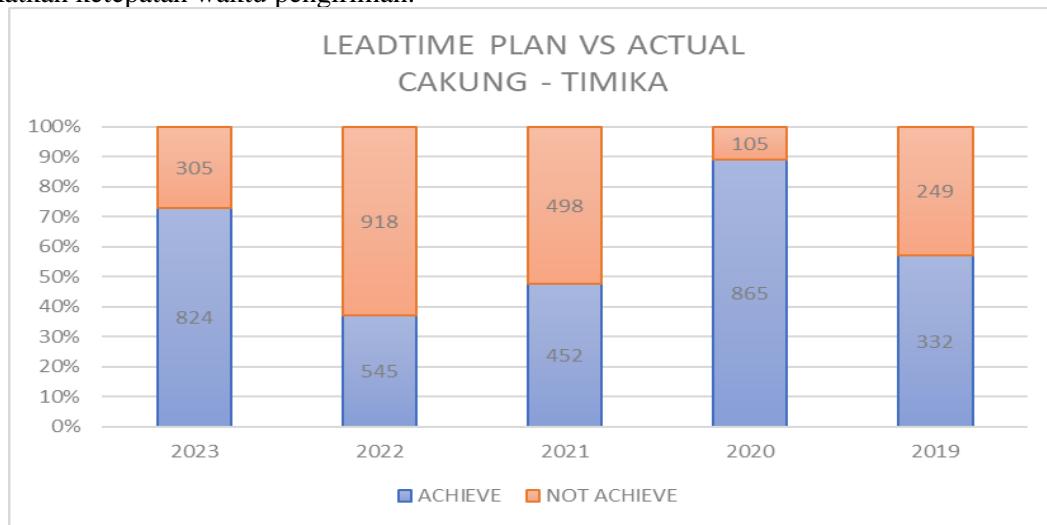
99.33% for rotation prediction. The developed system effectively supports optimal cargo loading optimization through accurate predictive models, enabling data-driven decision-making. With the implementation of this model, logistics companies can improve operational efficiency, minimize the risk of delays, and deliver superior customer service.

Keywords: genetic algorithm, load optimization, leadtime, logistics, random forest

1 Pendahuluan

Dalam sektor logistik, durasi pengiriman merupakan faktor kritis yang berdampak pada kepuasan konsumen dan efisiensi operasi. Waktu yang dibutuhkan dari pengambilan hingga penerimaan barang oleh pelanggan, yang dikenal sebagai *leadtime*, sering dijadikan standar dalam mengatur ekspektasi pelanggan dan mengoptimalkan jadwal distribusi serta manajemen sumber daya. Pelanggan selalu membutuhkan estimasi waktu tunggu yang akurat dan tepat untuk memastikan tanggal jatuh tempo produksi dan pengiriman mereka [1]. Layanan memiliki nilai lebih jika perusahaan dapat menyediakan atau memberikan produk dan jasa sesuai dengan keinginan, kebutuhan, dan harapan pengguna [2].

Salah satu penyebab layanan pengiriman kurang optimal dalam melakukan kegiatan pengiriman adalah ketika melakukan proses muat barang. Proses muat barang yang tidak optimal kedalam kontainer menyebabkan biaya pengeluaran perusahaan logistik bertambah. Hal ini dapat disebabkan belum optimalnya penggunaan ruang kosong dan penataan barang dalam kontainer. Masalah penempatan peti kemas yang sesuai di kapal peti kemas dalam perjalanan multiport, hal ini memerlukan pertimbangan konsekuensi setiap penempatan terhadap keputusan di port berikutnya [3]. Operasi peti kemas di atas kapal bertujuan untuk mengoptimalkan penempatan peti kemas yang tepat ke tempat penyimpanan yang sesuai di kapal peti kemas untuk mendapatkan peningkatan dalam hal biaya, waktu, dan stabilitas [4]. Efisiensi penyusunan barang kedalam kontainer menjadi faktor krusial dalam mengoptimalkan kapasitas muatan, keamanan barang, mengurangi biaya operasional dan meningkatkan ketepatan waktu pengiriman.



Gambar 1 Leadtime pengiriman cakung - timika

Rute Cakung-Timika merupakan salah satu rute dengan pengiriman dengan performance *leadtime* yang kurang stabil. Terlihat pada Gambar 1 fluktuasi trend rute tersebut dari tahun ke tahun dengan tertinggi Achieve pada tahun 2020 dan terendah pada tahun 2022 dan kurang optimalnya dalam penyusunan atau penempatan barang oleh *operation planner* turut menyumbang penurunan tersebut.

Penelitian ini memfokuskan pada penggunaan *Genetic Algorithm* dengan metode populasi, *fitness*, *crossover* dan *mutation* [5]. Sehingga menghasilkan generasi baru yang lebih baik dalam mencari urutan dalam proses muat barang. *Genetic Algorithm* (GA) merupakan salah satu metode heuristik yang digunakan untuk mengoptimasi sebuah proses. Terinspirasi dari teori evolusi Darwin menegaskan bahwa organisme dapat bertahan hidup dengan beberapa proses yakni reproduksi (*crossover*) dan mutasi. Konsep tersebut kemudian diadaptasi pada algoritma komputasi untuk

menemukan solusi pada masalah fungsi objektif [6]. Dikombinasikan dengan algoritma *machine learning* yaitu *Random Forest* yang bertujuan mendapatkan prediksi apakah barang boleh ditumpuk (*Stacking*) atau diputar posisinya (*Rotation*) sehingga mendapatkan hasil yang lebih baik lagi dan diharapkan secara tidak langsung dapat memberikan kontribusi dan mendukung pengambilan keputusan yang berbasis data dan teknologi dalam industri logistik.

Random Forest mampu memproses dataset besar dan variatif dengan efisien, serta mampu menangani data yang tidak seimbang dan kompleks, yang sering ditemui dalam konteks logistik dan supply chain [7]. Secara keseluruhan, *Random Forest* dipilih karena keunggulannya dalam menangani data heterogen, kegunaannya yang luas dalam masalah regresi dan klasifikasi, serta kestabilannya dalam konteks data yang beragam [8]. Teknik *Random Forest* dipilih karena karakteristik dataset yang digunakan memiliki satuan beragam (cm, kg, m³) dimana *Random Forest* yang merupakan berbasis *decision tree* tidak memerlukan normalisasi / standarisasi data. Data yang dimiliki juga berupa data campuran numerik (dimensi barang, berat barang, dimensi armada, berat armada) dan juga kategorikal (*rotation, stacking*).

2 Tinjauan Literatur

Dalam upaya mengoptimalkan optimasi muat barang untuk meningkatkan *leadtime* pengiriman, berbagai penelitian telah dilakukan dengan pendekatan yang beragam. Pada penelitian dengan masalah kurang optimalnya pengaturan barang dalam kendaraan van yang dapat menyebabkan penggunaan ruang yang tidak efisien sehingga bertujuan mengembangkan metode untuk mengoptimalkan penataan barang dalam box van [9]. Menggunakan dimensi masing-masing item, jumlah barang dari setiap jenis item solusi yang diterakan menggunakan MATLAB yang mendekati optimal dalam pengaturan barang didalam van. Pada penelitian lain dengan menggunakan algoritma genetika dengan permasalahan proses pemuatan barang ke dalam kontainer, disebabkan oleh pola penataan barang yang kurang optimal [10]. Menggunakan data ukuran dan berat barang, posisi awal cargo dalam kontainer, jumlah dan jenis barang menghasilkan konfigurasi pengisian barang ke dalam kontainer yang lebih optimal dibandingkan dengan metode manual atau pendekatan tradisional. Penelitian lain dengan tantangan dalam mengatur barang kebutuhan pokok (*ration items*) ke dalam kontainer secara optimal dan bertujuan meningkatkan efisiensi pemanfaatan ruang kontainer untuk pengiriman yang aman dan ekonomis [11]. Menggunakan algoritma Modified Forest Optimization Algorithm (MFOA) mampu menghasilkan pengaturan barang ke dalam kontainer dengan tingkat pemanfaatan ruang yang lebih tinggi menggunakan Dataset BR (Benchmark Ration items), yang berisi berbagai kumpulan contoh pengaturan bahan makanan ke dalam kontainer. Persentase pemanfaatan muat ruang meningkat dari 42% menjadi 81-83% menggunakan algoritma genetika [12]. Waktu proses menjadi lebih cepat terutama menggunakan *Random Crossover* dengan waktu tercepat sekitar 0.14 detik dibanding dengan parameter umum yang lebih lama [13]. *Random Forest* menghasilkan nilai akurasi lebih baik dibandingkan dengan regresi logistik dan *artificial neural network* sebesar 76.6% [14]. Tantangan memantau volume pengiriman harian menggunakan *Support Vector Regression* dan *Genetic Algorithm* menghasilkan MAPE terkecil sebesar 0,0969% dengan prediksi yang sangat akurat [15]

3 Metode Penelitian

Penelitian ini dilakukan melalui beberapa tahapan utama untuk memastikan optimasi muat barang yang akurat. Metode *Cross-Industry Standard Process for Data Mining* (CRISP-DM) adalah sebuah kerangka kerja atau metodologi yang digunakan dalam *data mining* dan analisis data [16]. Berikut adalah langkah-langkah penelitian dengan pendekatan metode CRISP-DM pada Gambar 2.



Gambar 2 Langkah penelitian dengan metode CRISP-DM

3.1 Business Understanding

Masalah pada penelitian pada akurasi optimasi penyusunan muat barang yang berpengaruh terhadap *leadtime* pengiriman. Sehingga penelitian ini bertujuan agar meningkatkan akurasi optimasi penyusunan barang dan menentukan model terbaik untuk digunakan.

3.2 Data Understanding

Tahapan ini dilakukan dengan mengumpulkan data historis pengiriman yang mencakup data panjang, lebar, tinggi barang serta berat dan volume barang. Untuk armada yang digunakan data yang mencakup panjang, lebar, tinggi muat armada serta volume dan berat maksimum armada.

3.3 Data Preparation

Preparation data dilakukan dengan tahapan membersihkan data seperti menangani data yang *blank/missing values*, menghapus atau memperlakukan *outlier* yang tidak wajar. Kemudian membagi dataset menjadi 80% untuk data *training*, 20% untuk data *testing*.

3.4 Modeling (Pembuatan Model)

Menggunakan *Genetic Algorithm* untuk optimasi muat dikombinasikan dengan algoritma *machine learning* yaitu *Random Forest*.

3.5 Evaluation (Evaluasi Model)

Evaluasi dilakukan dengan mengukur performa model menggunakan *Confusion Matrix*.

3.6 Deployment (Implementasi Model)

Implementasi sistem optimasi muat barang.

4 Hasil dan Pembahasan

4.1 Business Understanding

Perusahaan logistik dalam kegiatan muat barang agar memaksimalkan utilisasi ruang kontainer dan ketepatan waktu pengiriman dalam kegiatan operasionalnya dan menjaga kualitas layanan ditemukan bahwa terdapat *void space* (ruang kosong) akibat penataan barang manual yang tidak optimal sehingga *leadtime* pengiriman menjadi tidak tercapai. Penelitian ini dilakukan agar menghasilkan optimasi peningkatan utilisasi volume kontainer melalui optimasi tata letak berbasis *machine learning* dan mempercepat proses muat sehingga mengurangi *deviasi* waktu pengiriman.

4.2 Data Understanding

Tahapan selanjutnya setelah mengetahui permasalahan dan tujuan penelitian adalah dengan melakukan pengumpulan data. Pengumpulan data dilakukan dengan menarik data pengiriman barang di tahun 2023 yang tercatat pada sistem logistik perusahaan. Tujuan dari tahapan ini untuk memahami terhadap data yang dibutuhkan untuk dilakukan pemecahan masalah.

Data yang diperoleh berupa data armada dan data pengiriman. Untuk data armada dapat terlihat sampel pada Tabel 1:

Tabel 1 Sampel data armada

lic_no	model_id	model_desc	load_length_cm	load_width_cm	load_height_cm	max_volume_m3	max_weight_kg
B 9799 XW	CDE	CDE Box	300	160	160	7.68	2500
B 9782	TWB	Tronton	950	245	250	58.18	18000

EU		Wing Box					
B 9655 EI	TFT	Trailer 20 FT	606	244	259	38.29	28400
B 9656 EI	T40	Trailer 40 FT	1219	244	259	77.03	27060
B 9653 XU	CDD	CDD Box	400	185	200	14.8	4000

Kemudian untuk data pengiriman dapat terlihat sampelnya pada Tabel 2.

Tabel 2 Sampel data operasional pengiriman

manifest_no	carrier_no	item_id	item_id_lengh_cm	item_id_width_cm	item_id_height_cm	item_id_volum	item_id_weight_kg
M0259 4391	B 9214 FYW	ID01538 7793	106	75	70	0.5565	256.3
M0259 4391	B 9214 FYW	ID01538 9801	16	16	12	0.00307 2	0.14
M0259 4391	B 9214 FYW	ID01539 0873	26	22	20	0.01144	0.5
M0259 4391	B 9214 FYW	ID01539 0895	16	16	12	0.00307 2	0.5
M0259 7364	B 9887 SXT	ID00112 5915	47	36	36	0.06091 2	2.8
M0259 7364	B 9887 SXT	ID00112 5930	16	16	12	0.00307 2	0.2
M0259 7364	B 9887 SXT	ID01536 3433	55.63	26.67	26.67	0.03956 9	22.09
M0259 7364	B 9887 SXT	ID01537 6593	17.78	17.78	12.7	0.00401 48	0.72
M0259 9857	B 9946 SXT	ID02865 9464	16	16	12	0.00307 2	0.3
M0259 9857	B 9946 SXT	ID02866 0116	30	21	34	0.02142	4
M0259 9857	B 9946 SXT	ID02866 0562	16	16	12	0.00307 2	0.1
M0259 9857	B 9743 TXT	ID02869 1001	24	25	34	0.0204	1
M0259 9857	A 9484 U	ID02950 5106	35	24	14	0.01176	4
M0259 9857	B 9796 SXR	ID03350 5190	34	26	13	0.01149 2	4

4.3 Data Preparation

Pada tahapan ini pada Tabel 3 yang merupakan data armada dan Tabel 2 yang merupakan data operasional pengiriman digabungkan dengan proses join data berdasarkan lic_no (armada) dengan carrier_no (pengiriman) agar menghasilkan kebutuhan data yang nanti akan diolah dan menjadi sesuatu yang bernali, dan pada tabel hasil join carrier_no tidak perlu ditampilkan sehingga hasil penggabungan kedua tabel tersebut menjadi Tabel 3.

Tabel 3 Hasil join armada dengan pengiriman

manifest_no	carrier_no	model_id	item_id	item_id_lengh_cm	item_id_width_cm	item_id_height_cm	item_id_weight_kg
-------------	------------	----------	---------	------------------	------------------	-------------------	-------------------

M16410 0265	B 9653 XU	CDD	ID3245 9342	172	34	24	126
M16410 0265	B 9653 XU	CDD	ID3245 9353	87	54	50	263
M16410 0265	B 9653 XU	CDD	ID3245 9364	87	54	50	263
M16410 0265	B 9653 XU	CDD	ID3245 9375	146	83	69	1145
M16410 0265	B 9653 XU	CDD	ID3245 9386	308	41	55	978
M16410 0265	B 9653 XU	CDD	ID3245 9390	327	43	52	758
M16410 0265	B 9653 XU	CDD	ID3245 9401	74	67	99	531

4.4 Modeling

Tahapan dalam menerapkan model *Genetic Algorithm* dan *Random Forest* dilakukan dengan melihat sebelum dan sesudah atas optimasi muat barang yang dilakukan. Skenario 1 dilakukan dengan *Genetic Algorithm* tanpa kombinasi dengan *Random Forest* dimana semua barang pada *Stacking* dan *Rotation* bernilai N. Kemudian dilakukan optimasi menggunakan *Genetic Algorithm*. Setelah hasil optimasi dilakukan maka didapatkan hasil barang yang dimuat dan tidak dimuat dan dapat terlihat pada tampilan 3D Model atas barang yang dimuat. Skenario 2 dilakukan dengan *Genetic Algorithm* dengan sebelumnya kombinasi atas hasil *Stacking* dan *Rotation Random Forest* dimana semua barang dilakukan prediksi *Random Forest* untuk mendapatkan prediksi apakah dapat ditumpuk (*Stacking* bernilai Y) dan/atau dapat diputar posisinya (*Stacking* bernilai Y). Akan didapatkan hasil atas actual vs prediction pada *Stacking* dan *Rotation*, dan kemudian ikuti saran atas hasil prediksi jika ada yang berbeda dan dilanjutkan dengan optimasi menggunakan *Genetic Algorithm*. Setelah hasil optimasi dilakukan maka didapatkan hasil barang yang dimuat dan tidak dimuat dan dapat terlihat tampilan 3D Model atas barang yang dimuat

4.5 Genetic Algorithm

Genetic Algorithm akan digunakan untuk mengoptimasi dengan beberapa tahapan Populasi, Fitness, Crossover, Mutasi dan mendapatkan generasi baru sebagai hasil optimalnya.

4.5.1 Inisialisasi Populasi

Permasalahan muat barang disimulasikan dengan 7 barang yang akan dimuat pada kontainer. 7 barang ini akan menjadi kromosom kemudian dilakukan data acak dan dijadikan sebagai inisialisasi populasi dengan 5 populasi. Setiap kromosom merepresentasikan urutan pemuatan barang. Encoding merupakan permutasi daftar barang dengan [ID32459342, ID32459353, ID32459364, ID32459375, ID32459386, ID32459390, ID32459401] artinya barang ID32459342 dimuat pertama, lalu ID32459353, dan seterusnya). Dengan gen pada setiap gen adalah ID barang (dari kolom item_id). Sehingga dibuat 5 populasi dengan 7 kromosom terlihat pada Tabel 4:

Tabel 4 Inisialisasi 5 populasi dengan 7 kromosom

Populasi	Kromosom
Populasi 1	[ID32459342, ID32459353, ID32459364, ID32459375, ID32459386, ID32459390, ID32459401]
Populasi 2	[ID32459386, ID32459401, ID32459342, ID32459364, ID32459353, ID32459375, ID32459390]
Populasi 3	[ID32459375, ID32459386, ID32459390, ID32459342, ID32459401, ID32459364, ID32459353]
Populasi 4	[ID32459364, ID32459375, ID32459342, ID32459390, ID32459401, ID32459353, ID32459386]
Populasi 5	[ID32459390, ID32459375, ID32459401, ID32459386, ID32459364, ID32459342,

[ID32459353]

4.5.2 Fitness

Langkah berikut menggunakan Skenario 1.

Pengecekan Fitness untuk Populasi 1 dengan kapasitas kontainer 400 x 185 x 200 dengan maximum load weight 4000 kg terlihat pada Tabel 5:

Tabel 5 Muat barang populasi 1 pada skenario 1

Barang	St ac k	Ro tat e	p pakai (cm)	cek p muat	l pakai (cm)	cek l muat	t pakai (cm)	cek t muat	w pakai (kg)	cek w muat	Status Muat
ID32459342	N	N	172	Muat	34	Muat	24	Muat	126	Muat	Muat
ID32459353	N	N	259	Muat	88	Muat	74	Muat	389	Muat	Muat
ID32459364	N	N	346	Muat	142	Muat	124	Muat	652	Muat	Muat
ID32459375	N	N	492	Tidak dimuat	225	Tidak Dimuat	193	Muat	1797	Muat	Tidak dimuat
ID32459386	N	N	654	Tidak dimuat	183	Muat	179	Muat	1630	Muat	Tidak dimuat
ID32459390	N	N	673	Tidak dimuat	185	Muat	176	Muat	1410	Muat	Tidak dimuat
ID32459401	N	N	420	Tidak dimuat	209	Tidak dimuat	223	Tidak dimuat	1183	Muat	Tidak dimuat

Dari hasil Populasi 1 menghasilkan dengan barang termuat 1 dengan tidak termuat 0, sehingga pada masing – masing data acak menghasilkan [1,1,1,0,0,0,0] dengan *fitness* tertinggi 0.2349 berdasarkan volume tertinggi yang berhasil dimuat.

Kemudian selanjutnya dilakukan perhitungan Fitness untuk Populasi 2, 3, 4, 5 dengan cara yang sama yaitu dengan melihat kapasitas dan dimensi apakah masih tersedia atau tidak. Dan hasil *fitness* atas masing – masing populasi terlihat pada Tabel 6.

Tabel 6 Hasil fitness masing - masing populasi skenario 1

Populasi	Hasil Encode	Fitness
Populasi 1	[1,1,1,0,0,0,0]	0.2439
Populasi 2	[1,1,0,0,0,0,0]	0.490842
Populasi 3	[1,0,0,1,1,0,0]	0.976494
Populasi 4	[1,1,0,0,0,0,0]	0.836142
Populasi 5	[1,0,0,0,0,0,0]	0.731172

Langkah berikut menggunakan Skenario 2.

Pengecekan Fitness untuk Populasi 1 dengan kapasitas kontainer 400 x 185 x 200 dengan maximum load weight 4000 kg terlihat pada Tabel 7.

Tabel 7 Muat barang populasi 1 pada skenario 2

Barang	St ac k	Ro tat e	p pakai (cm)	cek p muat	l pakai (cm)	cek l muat	t pakai (cm)	cek t muat	w pakai (kg)	cek w muat	Status Muat
ID32459342	Y	N	172	Muat	34	Muat	24	Muat	126	Muat	Muat
ID32459353	Y	Y	259	Muat	88	Muat	74	Muat	389	Muat	Muat

ID324 59364	Y	N	346	Muat	142	Muat	124	Muat	652	Muat	Muat
ID324 59375	N	Y	492	Tidak Dimuat	225	Tidak Dimuat	193	Muat	1797	Muat	Muat
ID324 59386	Y	N	800	Tidak Dimuat	266	Tidak Dimuat	248	Tidak Dimuat	2775	Muat	Muat
ID324 59390	Y	N	1127	Tidak Dimuat	309	Tidak Dimuat	300	Tidak Dimuat	3533	Muat	Muat
ID324 59401	N	N	1201	Tidak Dimuat	376	Tidak Dimuat	399	Tidak Dimuat	4064	Tidak Dimuat	Tidak dimuat

Dari hasil Populasi 1 menghasilkan dengan barang termuat 1 dengan tidak termuat 0, sehingga pada masing – masing data acak menghasilkan [1,1,1,1,1,1,0] dengan *fitness* tertinggi 2.872006 berdasarkan volume tertinggi yang berhasil dimuat. Dan hasil *fitness* atas masing – masing populasi terlihat pada Tabel 8.

Tabel 8 Hasil fitness masing - masing populasi skenario 2

Populasi	Hasil Encode	Fitness
Populasi 1	[1,1,1,1,1,1,0]	2.872006
Populasi 2	[1,1,1,1,1,1,0]	2.631676
Populasi 3	[1,1,1,1,1,1,0]	3.127948
Populasi 4	[1,1,1,1,1,1,0]	2.668308
Populasi 5	[1,1,1,1,1,1,0]	3.127948

4.5.3 Crossover

Setelah didapatkan hasil *fitness* atas masing – masing populasi, kemudian dipilih secara acak populasi yang paling baik *fitness*nya dengan mengambil 2 populasi terbaik.

Populasi 3 dengan *fitness* 0.976494, Populasi 4 dengan *fitness* 0.836142 diambil kemudian dijadikan Orang Tua (parent) dengan populasi baru.

1. Orang Tua 1 menjadi populasi baru berisi kromosom [ID32459375, ID32459386, ID32459390, ID32459342, ID32459401, ID32459364, ID32459353]
2. Orang Tua 2 menjadi populasi baru berisi kromosom [ID32459364, ID32459375, ID32459342, ID32459390, ID32459401, ID32459353, ID32459386]

Dengan metode *crossover* pada subset Orang Tua 1 diisi sisa gen dengan urutan dari Orang Tua 2 dan hindari duplikat dengan mengambil titik pemotongan secara acak dan pada kasus ini mengambil titik potong pada posisi indeks 2 hingga 5 [ID32459390, ID32459342, ID32459401, ID32459364] untuk sehingga menjadi Child baru [... , ... , ID32459390, ID32459342, ID32459401, ID32459364, ...].

Dan hasil *crossover* ini menghasilkan Child akhir menjadi :

[ID32459386, ID32459375, ID32459390, ID32459342, ID32459401, ID32459364, ID32459353]

4.5.4 Mutasi

Anak yang telah dihasilkan dari proses *crossover* [ID32459386, ID32459375, ID32459390, ID32459342, ID32459401, ID32459364, ID32459353] selanjutnya dilakukan swap *mutation* atau pertukar posisi gen secara acak. Untuk hal ini dilakukan pengambilan posisi indeks ke 3 menjadi ke 6 , posisi indeks ke 4 menjadi ke 1, posisi indeks ke 2 menjadi ke 7,

Sebelum :

[ID32459386, ID32459375, ID32459390, ID32459342, ID32459401, ID32459364, ID32459353]

Setelah :

[ID32459401, ID32459375, ID32459342, ID32459353, ID32459386, ID32459364, ID32459390]

4.5.5 Generasi Baru

Atas hasil anak pada proses mutasi tercipta generasi baru [ID32459401, ID32459375, ID32459342, ID32459353, ID32459386, ID32459364, ID32459390].

Kemudian generasi baru ini dijadikan sebagai kromosom elite dan dapat dijadikan hasil muat barang yang paling optimal.

4.6 Random Forest

Setelah penginputan barang dilakukan, kemudian pada pilihan *Stacking* dan *Rotation* dapat menentukan bagaimana hal ini mempengaruhi hasil muat barang secara optimal. Pemilihan *Stacking* (barang boleh ditumpuk) dan *Rotation* (barang boleh diputar posisi peletakannya) berdasarkan pengalaman user dilapangan dalam menentukan barang ketika akan dimuat.

4.6.1 Data Preparation

Persiapan data dilakukan dengan membaca data historis input dari database dengan melakukan filter data yang memiliki *description_of_goods* tidak kosong dan tidak NULL untuk menentukan boleh ditumpuk atau tidak (*Stacking*) dan dimensi barang dalam menentukan apakah boleh diputar posisinya atau tidak (*Rotation*).

4.6.2 Preprocessing

Data dengan kolom pada Tabel 9 akan ditentukan untuk prediksi *Stacking* dan *Rotation*.

Tabel 9 Berat dan deskripsi barang

item_id	item_id_weight_kg	description_of_goods
ID27532394	64	BOOT
ID27565036	415	MOTOR GP-ELE
ID27574910	1	SEAL O RING
ID27911610	27	CAP SCREW
ID38962165	212	FDAO 60-208 L
ID38962176	212	FDAO 60-208 L
ID38962180	212	FDAO 60-208 L
ID38962191	212	FDAO 60-208 L
ID28621561	28	MANIFOLD
ID28644020	99	SLEEVE
ID28716735	6	BOLT HEXSOCK

Karena berat yang ada sangat bervariasi, ditentukan minimum berat yang bisa dilakukan tumpuk adalah 20 kg. Jika barang memenuhi lebih dari 20 kg maka bernilai 1, jika dibawah 20 kg maka bernilai 0 sehingga dapat terlihat datanya pada Tabel 10.

Tabel 10 Penentuan berat barang terhadap stacking

item_id	description_of_goods	item_id_weight_kg	> 20 kg
ID27532394	BOOT	64	1
ID27565036	MOTOR GP-ELE	415	1
ID27574910	SEAL O RING	1	0
ID27911610	CAP SCREW	27	1
ID38962165	FDAO 60-208 L	212	1
ID38962176	FDAO 60-208 L	212	1
ID38962180	FDAO 60-208 L	212	1
ID38962191	FDAO 60-208 L	212	1
ID28621561	MANIFOLD	28	1
ID28644020	SLEEVE	99	1
ID28716735	BOLT HEXSOCK	6	0

Perhitungan untuk Gini Tabel 10:

- a. Berat > 20 kg, terdiri dari 9 barang

$$\begin{aligned}\text{Gini} &= 1 - (1/9)^2 - (1/9)^2 - (1/9)^2 - (4/9)^2 - (1/9)^2 - (1/9)^2 \\ &= 1 - (0.1111)^2 - (0.1111)^2 - (0.1111)^2 - (0.4444)^2 - (0.1111)^2 - (0.1111)^2 \\ &= 1 - 0.01234321 - 0.01234321 - 0.01234321 - 0.19749136 - 0.01234321 - 0.01234321 \\ &= 0.74079259\end{aligned}$$

- b. Berat ≤ 20 kg, terdiri dari 2 barang

$$\begin{aligned} \text{Gini} &= 1 - (1/2)^2 - (1/2)^2 \\ &= 1 - 0.25 - 0.25 \\ &= 0.5 \end{aligned}$$

$$\begin{aligned} \text{Pembobotan Gini} &= ((9/11) \times 0.74079259) + ((2/11) \times 0.5) = 0.74079259 + 0.09091 \\ &= 0.83170259 \end{aligned}$$

Kemudian dari `description_of_goods` ditentukan apakah barang mudah pecah atau tidak (`fragile`). Jika barang termasuk fragile maka bernilai 1, jika tidak maka bernilai 0.

Tabel 11 Penentuan barang fragile

item_id	description_of_goods	fragile
ID27532394	BOOT	1
ID27565036	MOTOR GP-ELE	1
ID27574910	SEAL O RING	0
ID27911610	CAP SCREW	0
ID38962165	FDAO 60-208 L	1
ID38962176	FDAO 60-208 L	1
ID38962180	FDAO 60-208 L	1
ID38962191	FDAO 60-208 L	1
ID28621561	MANIFOLD	0
ID28644020	SLEEVE	0
ID28716735	BOLT HEXSOCK	0

Perhitungan untuk Gini Tabel 11:

- a. Barang termasuk Fragile, terdiri dari 6 barang

$$\begin{aligned} \text{Gini} &= 1 - (1/6)^2 - (1/6)^2 - (4/6)^2 \\ &= 1 - (0.1667)^2 - (0.1667)^2 - (0.6667)^2 \\ &= 1 - 0.02778889 - 0.02778889 - 0.44448889 \\ &= 0.49993333 \end{aligned}$$

- b. Barang termasuk bukan Fragile, terdiri dari 5 barang

$$\begin{aligned} \text{Gini} &= 1 - (1/5)^2 - (1/5)^2 - (1/5)^2 - (1/5)^2 - (1/5)^2 \\ &= 1 - (0.2)^2 - (0.2)^2 - (0.2)^2 - (0.2)^2 - (0.2)^2 \\ &= 1 - 0.04 - 0.04 - 0.04 - 0.04 - 0.04 \\ &= 0.8 \end{aligned}$$

$$\begin{aligned} \text{Pembobotan Gini} &= ((6/11) \times 0.49993333) + ((5/11) \times 0.8) = 0.272690907045 + 0.36364 \\ &= 0.636330907045 \end{aligned}$$

Atas hasil pembobotan gini antara kriteria Berat > 20 kg dengan nilai bobot 0.83170259 dan Fragile dengan nilai bobot 0.636330907045, maka nilai bobot terkecil yang diambil sebagai parameter untuk prediksi *Stacking* yaitu berdasarkan Fragile.

Selanjutnya untuk memprediksi apakah penempatan barang bisa dilakukan rotasi atau tidak ketika muat kedalam *container*, ditentukan dengan kriteria berat seperti yang terlihat pada Tabel 12 bahwa jika barang dengan berat melebihi 50 kg tidak disarankan untuk rotasi dan bernilai 0 dan dibawah 50 kg dapat dilakukan rotasi bernilai 1.

Tabel 12 Barang dengan pertimbangan berat untuk rotasi

item_id	description_of_goods	item_id_weight_kg	> 50 kg
ID27532394	BOOT	64	0
ID27565036	MOTOR GP-ELE	415	0
ID27574910	SEAL O RING	1	1
ID27911610	CAP SCREW	27	1
ID38962165	FDAO 60-208 L	212	0
ID38962176	FDAO 60-208 L	212	0
ID38962180	FDAO 60-208 L	212	0
ID38962191	FDAO 60-208 L	212	0
ID28621561	MANIFOLD	28	1

ID28644020	SLEEVE	99	0
ID28716735	BOLT HEXSOCK	6	1

Perhitungan untuk Gini Tabel 12:

- a. Berat > 50 kg, terdiri dari 7 barang

$$\begin{aligned} \text{Gini} &= 1 - (1/7)^2 - (1/7)^2 - (4/7)^2 - (1/7)^2 \\ &= 1 - (0.1429)^2 - (0.1429)^2 - (0.5714)^2 - (0.1429)^2 \\ &= 1 - 0.02042041 - 0.02042041 - 0.32649796 - 0.02042041 \\ &= 0.61224081 \end{aligned}$$

- b. Berat ≤ 50 kg, terdiri dari 4 barang

$$\begin{aligned} \text{Gini} &= 1 - (1/4)^2 - (1/4)^2 - (1/4)^2 - (1/4)^2 \\ &= 1 - (0.25)^2 - (0.25)^2 - (0.25)^2 - (0.25)^2 \\ &= 1 - 0.0625 - 0.0625 - 0.0625 - 0.0625 \\ &= 0.75 \end{aligned}$$

$$\begin{aligned} \text{Pembobotan Gini} &= ((7/11) \times 0.61224081) + ((4/11) \times 0.75) = 0.389607787959 + 0.272727 \\ &= 0.662334787959 \end{aligned}$$

Kemudian untuk rotasi barang dapat dipertimbangkan juga dari jenis barang apakah bersifat fleksibel jika dilakukan rotasi atau tidak.

Tabel 13 Penentuan barang yang fleksibel untuk rotasi

item_id	description_of_goods	flexible
ID27532394	BOOT	1
ID27565036	MOTOR GP-ELE	1
ID27574910	SEAL O RING	1
ID27911610	CAP SCREW	1
ID38962165	FDAO 60-208 L	0
ID38962176	FDAO 60-208 L	0
ID38962180	FDAO 60-208 L	0
ID38962191	FDAO 60-208 L	0
ID28621561	MANIFOLD	1
ID28644020	SLEEVE	1
ID28716735	BOLT HEXSOCK	1

Perhitungan untuk Gini Tabel 13:

- a. Barang termasuk Flexible, terdiri dari 7 barang

$$\begin{aligned} \text{Gini} &= 1 - (1/7)^2 - (1/7)^2 - (1/7)^2 - (1/7)^2 - (1/7)^2 - (1/7)^2 \\ &= 1 - (0.1429)^2 - (0.1429)^2 - (0.1429)^2 - (0.1429)^2 - (0.1429)^2 - (0.1429)^2 \\ &= 1 - 0.02042041 - 0.02042041 - 0.02042041 - 0.02042041 - 0.02042041 - 0.02042041 \\ &= 0.85705713 \end{aligned}$$

- b. Barang termasuk bukan Flexible, terdiri dari 4 barang

$$\begin{aligned} \text{Gini} &= 1 - (1/4)^2 \\ &= 1 - (0.25)^2 \\ &= 1 - 0.0625 \\ &= 0.9375 \end{aligned}$$

$$\begin{aligned} \text{Pembobotan Gini} &= ((7/11) \times 0.85705713) + ((4/11) \times 0.9375) = 0.545399991507 + 0.34090909 \\ &= 0.886309081507 \end{aligned}$$

Atas hasil pembobotan gini antara kriteria Berat < 50 kg dengan nilai bobot 0.662334787959 dan Flexible dengan nilai bobot 0.886309081507, maka nilai bobot terkecil yang diambil sebagai parameter untuk prediksi *Rotation* yaitu berdasarkan Berat.

4.6.3 Confusion Matrix

Visualisasi pada *Confusion Matrix* digunakan untuk validasi sistem terhadap kinerja sistem yang dibuat berdasarkan skenario pembagian data disertai dengan perhitungan untuk menentukan akurasi model pada Tabel 14.

Tabel 14 Confusion matrix stacking

Stacking	Predicted N	Predicted Y
Actual N	1679	16
Actual Y	0	395

Dengan nilai Accuracy 99.23 % dengan hasil Classification Report terlihat pada Tabel 15.

Tabel 15 Classification report stacking

Classification	Precision	Recall	F1-score	Support
N	1	0.99	1	1695
Y	0.95	1	0.98	395

Kemudian berikut untuk *Rotation Confusion Matrix* terlihat pada Tabel 16.

Tabel 16 Confusion matrix rotation

Rotation	Predicted N	Predicted Y
Actual N	1306	14
Actual Y	0	770

Dengan nilai Accuracy 99.33 % dengan hasil Classification Report pada Tabel 17.

Tabel 17 Hasil classification report

Classification	Precision	Recall	F1-score	Support
N	1	0.99	0.99	1320
Y	0.98	1	0.99	770

4.6.4 Prediction

Atas proses yang telah dilakukan maka model *Stacking* dan model *Rotation* yang dihasilkan oleh *Random Forest* dengan hasil terlihat pada Tabel 18 dan Tabel 19.

Tabel 18 Hasil prediksi stacking

Item_id	Description	Actual	Predicted Stacking
ID27532394	BOOT	Y	N
ID27565036	MOTOR GP-ELE	N	N
ID27574910	SEAL O RING	Y	N
ID27911610	CAP SCREW	N	N
ID38962165	FDAO 60-208 L	N	N
ID38962176	FDAO 60-208 L	N	N
ID38962180	FDAO 60-208 L	Y	N
ID38962191	FDAO 60-208 L	N	N
ID28621561	MANIFOLD	N	N
ID28644020	SLEEVE	N	N
ID28716735	BOLT HEXSOCK	N	N

Tabel 19 Hasil prediksi rotation

Item_id	Description	Actual	Predicted Rotation
ID27532394	BOOT	N	N
ID27565036	MOTOR GP-ELE	N	N
ID27574910	SEAL O RING	N	N
ID27911610	CAP SCREW	Y	Y
ID38962165	FDAO 60-208 L	N	N
ID38962176	FDAO 60-208 L	N	N

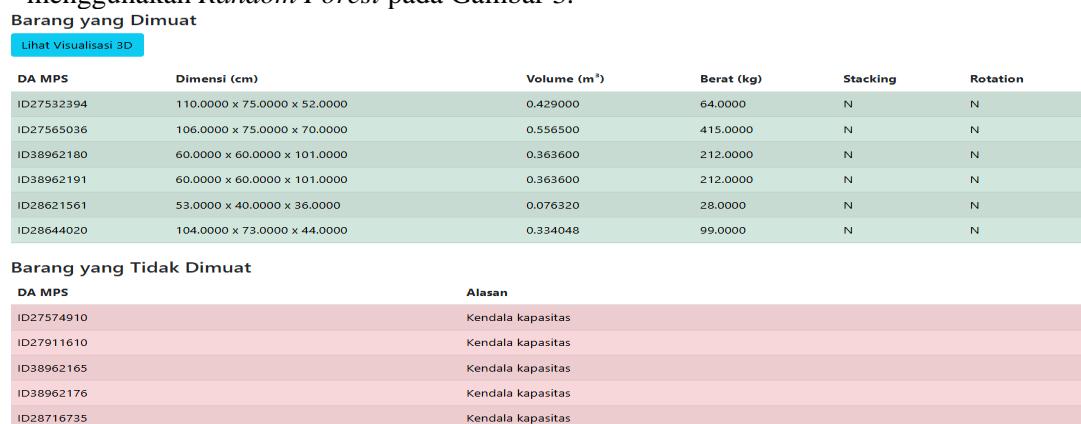
ID38962180	FDAO 60-208 L	Y	Y
ID38962191	FDAO 60-208 L	N	N
ID28621561	None	N	N
ID28644020	None	Y	Y
ID28716735	BOLT HEXSOCK	N	N

4.7 Deployment

Atas penerapan hasil yang telah dilakukan maka dikembangkan aplikasi yang digunakan untuk melakukan optimasi menggunakan *Genetic Algorithm* dan *Random Forest*.

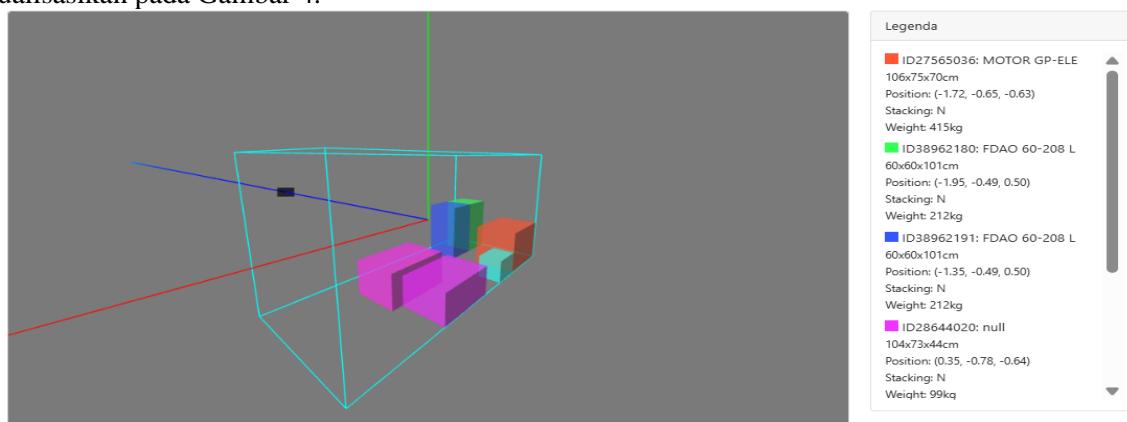
Berikut hasil yang telah diterapkan dalam bentuk aplikasi :

- Hasil barang yang dioptimasi menggunakan Genetic Algorithm dengan nilai *Stacking* dan *Rotation* kesemuanya bernilai N dengan tanpa menerapkan prediksi *Stacking* dan *Rotation* menggunakan *Random Forest* pada Gambar 3.



Gambar 3 Hasil barang dimuat dan tidak dimuat dengan *genetic algorithm* tanpa prediksi *random forest*

Dan berikut hasil visualisasi 3D muat barang kedalam kontainer sesuai hasil dari Gambar 3 divisualisasikan pada Gambar 4.



Gambar 4 Visualisasi 3D optimasi muatan dengan *genetic algorithm* tanpa prediksi *random forest*

- Hasil barang yang dioptimasi menggunakan Genetic Algorithm dengan nilai *Stacking* dan *Rotation* dengan hasil prediksi *Stacking* dan *Rotation* menggunakan *Random Forest* pada Gambar 5.

Items									
DA MPS	Desc of Goods	Dimensions (cm)	Volume (m³)	Weight (kg)	Fragile	Flexible To Rotate	Stacking	Rotation	
ID27532394	BOOT	110.000 x 75.000 x 52.0000	0.429	64.0	✓	✓	✓	X	
ID27565036	MOTOR GP-ELE	106.000 x 75.000 x 70.0000	0.5565	415.0	✓	✓	✓	X	
ID27574910	SEAL O RING	16.0000 x 16.0000 x 12.0000	0.003072	1.0	X	✓	✓	✓	
ID27911610	CAP SCREW	36.0000 x 36.0000 x 25.0000	0.0324	27.0	X	✓	X	✓	
ID38962165	FDAO 60-208 L	60.0000 x 60.0000 x 101.0000	0.3636	212.0	✓	X	✓	X	
ID38962176	FDAO 60-208 L	60.0000 x 60.0000 x 101.0000	0.3636	212.0	✓	X	✓	X	
ID38962180	FDAO 60-208 L	60.0000 x 60.0000 x 101.0000	0.3636	212.0	✓	X	✓	X	
ID38962191	FDAO 60-208 L	60.0000 x 60.0000 x 101.0000	0.3636	212.0	✓	X	✓	X	
ID28621561	None	53.0000 x 40.0000 x 36.0000	0.07632	28.0	X	✓	X	✓	
ID28644020	None	104.0000 x 73.0000 x 44.0000	0.334048	99.0	X	✓	X	X	
ID28716735	BOLT HEXSOCK	29.0000 x 24.0000 x 20.0000	0.01392	6.0	X	✓	✓	✓	

Gambar 5 Hasil prediksi stacking dan rotation random forest

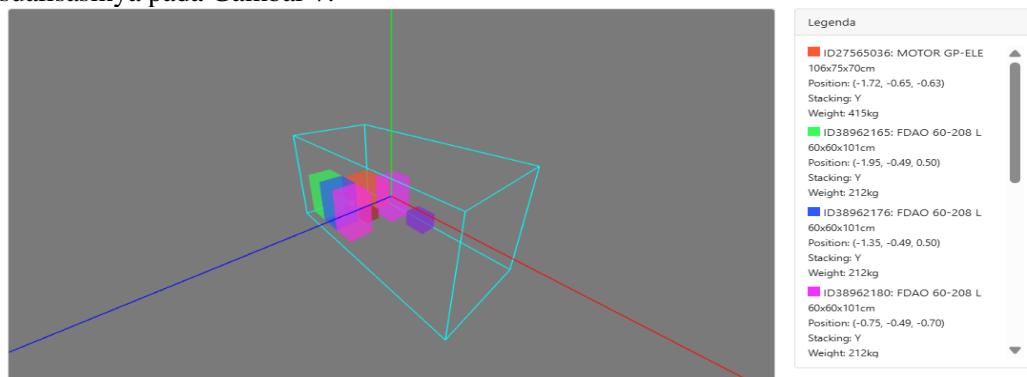
Dan berikut hasil Barang yang dimuat dan Tidak Dimuat setelah prediksi Stacking dan Rotation Random Forest pada Gambar 6.

Barang yang Dimuat					
Lihat Visualisasi 3D					
DA MPS	Dimensi (cm)	Volume (m³)	Berat (kg)	Stacking	Rotation
ID27565036	106.0000 x 75.0000 x 70.0000	0.556500	415.0000	Y	N
ID27574910	16.0000 x 16.0000 x 12.0000	0.003072	1.0000	Y	Y
ID27911610	36.0000 x 36.0000 x 25.0000	0.032400	27.0000	N	Y
ID38962165	60.0000 x 60.0000 x 101.0000	0.363600	212.0000	Y	N
ID38962176	60.0000 x 60.0000 x 101.0000	0.363600	212.0000	Y	N
ID38962180	60.0000 x 60.0000 x 101.0000	0.363600	212.0000	Y	N
ID38962191	60.0000 x 60.0000 x 101.0000	0.363600	212.0000	Y	N
ID28621561	53.0000 x 40.0000 x 36.0000	0.076320	28.0000	N	Y
ID28716735	29.0000 x 24.0000 x 20.0000	0.013920	6.0000	Y	Y

Barang yang Tidak Dimuat	
DA MPS	Alasan
ID27532394	Kendala kapasitas
ID28644020	Kendala kapasitas

Gambar 6 Hasil barang dimuat dan tidak dimuat dengan genetic algorithm dengan hasil prediksi stacking dan rotation random forest

Dan berikut hasil visualisasi 3D muat barang kedalam kontainer sesuai hasil dari Gambar 6 dapat dilihat visualisasinya pada Gambar 7.



Gambar 7 Visualisasi 3D optimasi muatan dengan genetic algorithm dengan prediksi random forest

5 Kesimpulan

Dari hasil yang didapatkan dari penelitian ini bahwa *Genetic Algorithm* mengambil *fitness* tertinggi 0.836142 skenario 1 tanpa *Random Forest*, 3.127948 Skenario 2 dengan kombinasi *Random Forest*. *Random Forest* menghasilkan prediksi *Stacking* mencapai akurasi 99.23 % dan prediksi *Rotation* mencapai akurasi 99.33 %. Simulasi menggunakan *Genetic Algorithm* tanpa menggunakan prediksi *Stacking* dan *Rotation Random Forest* menghasilkan 6 barang yang dimuat dan 5 barang tidak dimuat. Sedangkan simulasi menggunakan *Genetic Algorithm* dengan menggunakan prediksi *Stacking* dan *Rotation Random Forest* menghasilkan 9 barang yang dimuat dan 2 barang tidak dimuat.

Hal ini dapat disimpulkan *Genetic Algorithm* membantu optimasi dalam memperkirakan utilisasi muat barang dan dengan dikombinasikan dengan *Random Forest* menghasilkan hasil lebih baik dalam memperkirakan utilitas muat barang. Perlu dilakukan penerapan lebih lanjut dengan jenis data dengan dimensi atau bentuk barang yang beragam. Serta kombinasi jumlah tumpukan barang dalam batas yang diperbolehkan sesuai masing – masing barang sehingga bisa didapatkan optimasi yang sesuai untuk pengembangan lebih baik kedepannya.

Referensi

- [1] R. Y. Zhong, G. Q. Huang, Q. Y. Dai, and T. Zhang, “*Estimation of Lead Time in the RFID-Enabled Real-Time Shopfloor Production with a Data Mining Model*,” in *19th International Conference on Industrial Engineering and Engineering Management: Assistive Technology of Industrial Engineering*, Springer Science and Business Media Deutschland GmbH, 2013, pp. 321–331. DOI: 10.1007/978-3-642-38391-5_33.
- [2] P. Widiyanto *et al.*, “*The Relationship Between Service Quality, Timeliness of Arrival, Departure Flip Ship Logistics Aand People and Customer Satisfaction: A Case in Indonesia*,” *Academy of Entrepreneurship Journal*, Vol. 27, No. 6, pp. 1–10, 2021.
- [3] I. D. Wilson and P. A. Roach, “*Container Stowage Planning: a Methodology for Generating Computerised Solutions*,” 2000. [Online]. Available: www.stockton-press.co.uk/jors
- [4] M. S. Bilican, M. Karatas, Y. J. Zheng, H. H. Turan, and M. Deveci, “*A Survey of Shipping Line CONTAINER Stowage Planning Problems*,” Dec. 01, 2024, *Elsevier Ltd*. DOI: 10.1016/j.eswa.2024.124408.
- [5] Rizky Fatih Syahputra and Yahfizham Yahfizham, “Menganalisis Konsep Dasar Algoritma Genetika,” *Bhinneka: Jurnal Bintang Pendidikan dan Bahasa*, Vol. 2, No. 1, pp. 120–132, Dec. 2023, DOI: 10.59024/bhinneka.v2i1.643.
- [6] D. Suhartono, “Optimasi dengan *Genetic Algorithm* menggunakan Python.” Accessed: Jun. 04, 2025. [Online]. Available: <https://socs.binus.ac.id/2020/07/29/optimasi-dengan-genetic-algorithm-menggunakan-python/>
- [7] A. Albadrani, F. Alghayadh, M. A. Zohdy, E. Aloufi, and R. Olawoyin, “*Performance and Predicting of Inbound Logistics Processes using Machine Learning*,” in *2021 IEEE 11th Annual Computing and Communication Workshop and Conference, CCWC 2021*, Institute of Electrical and Electronics Engineers Inc., Jan. 2021, pp. 790–795. DOI: 10.1109/CCWC51732.2021.9376171.
- [8] A. Arifi, P. Bouros, and T. Chondrogiannis, “*A Study on ETA Prediction using Machine Learning and Recovered Routes*,” *EDBT/ICDT Workshops*, 2024, [Online]. Available: <https://www.openstreetmap.org/>
- [9] K. A. Santoso, I. M. Ilmiyah, and A. Pradjaningsih, “*Optimizing the Arrangement of Goods in Box Van using the Tabu Search Algorithm*,” *Statistics, Optimization and Information Computing*, Vol. 13, No. 4, pp. 1472–1479, 2025, DOI: 10.19139/SOIC-2310-5070-2151.
- [10] H.-C. Chen, A. M. Widodo, A. Wisnujati, and M. Rahaman, “*Using Genetic Algorithm to Optimize The-Loading Space in a Container*,” 2021. [Online]. Available: <https://easychair.org/cfp/FutureICT2021>
- [11] S. K. Ravichandran and A. Sasi, “*Optimal Arrangement of Ration Items into Container using Modified Forest Optimization Algorithm*,” *Indian Journal of Computer Science and Engineering*, Vol. 11, No. 4, pp. 334–346, Jul. 2020, DOI: 10.21817/indjcse/2020/v11i4/201104173.
- [12] W. Fajar Saputra, D. Diah Damayanti, and B. Santosa, “*Container Loading Allocation to Improve Space and Loading Utilization of Fleet Compartment Capacity By using Genetic Algorithm*,” 2019.
- [13] H. F. Sulaiman, B. T. Sartana, and U. Budiyanto, “*Genetic Algorithm with Random Crossover and Dynamic Mutation on Bin Packing Problem*,” *Proceeding of the Electrical Engineering Computer Science and Informatics*, Vol. 6, No. 0, Oct. 2019, DOI: 10.11591/eecsi.v6i0.1963.
- [14] H. K. Pambudi *et al.*, “*Prediksi Status Pengiriman Barang menggunakan Metode Machine Learning*,” *Jurnal Ilmiah Teknologi Infomasi Terapan*, 2020, Accessed: Jan. 21, 2025. [Online]. Available: <https://journal.widyatama.ac.id/index.php/jitter/article/view/396/304>

- [15] N. Nendi and A. Wibowo, "Prediksi Jumlah Pengiriman Barang menggunakan Kombinasi Metode *Support Vector Regression*, Algoritma Genetika dan *Multivariate Adaptive Regression Splines*," *Jurnal Teknologi Informasi dan Ilmu Komputer*, 2020, DOI: 10.25126/jtiik.202072441.
- [16] F. Sulianta, *Buku Dasar Data Mining from A to Z - Feri SLN (Free)*. 2024.