

T.C. PİRİ REİS UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES

PARKING SPACE OPTIMIZATION
FOR EFFICIENT AND EFFECTIVE RO-RO TERMINAL OPERATIONS

PhD. THESIS

HASAN GÖLER

HASAN, GÖLER

Ph.D. Thesis

PRU 2024

NOVEMBER 2024

T.C. PİRİ REİS UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES

PARKING SPACE OPTIMIZATION
FOR EFFICIENT AND EFFECTIVE RO-RO TERMINAL OPERATIONS

PhD. THESIS

by

HASAN, GÖLER

B.BA, Business Administration, Anadolu University, 2005

M.Sc., Maritime Transportation and Management Engineering, Piri Reis University,
2017

B.Sc., Maritime Transportation and Management Engineering, Istanbul University,
2023

Maritime Transportation and Management Engineering Program

Thesis Advisor: Prof. Dr. Ergün DEMİREL

NOVEMBER 2024

APPROVAL of the INSTITUTE for GRADUATE STUDIES

PRU, Graduate Studies Institute's Doctoral student “178015003” and “Hasan GÖLER”
prepared his thesis titled

“PARKING SPACE OPTIMIZATION FOR EFFICIENT AND EFFECTIVE RO-RO
TERMINAL OPERATIONS”

by fulfilling all the necessary conditions determined by the relevant regulations and
successfully presented it to the jury whose signatures are below.

Approval of the Institute for Graduate Studies

Assoc. Prof. Dr. Özgür AYBAR
Director

Prof. Dr. H. Funda YERCAN
Head of Department

Prof. Dr. Ergün DEMİREL
Advisor

Examining Committee Members

Prof. Dr. Taner ALBAYRAK

Dr. Öğr. Üyesi Dinçer BAYER

(Title and Name)

(Title and Name)

Date of Approval: 07.11.2024

ABSTRACT

PARKING SPACE OPTIMIZATION FOR EFFICIENT AND EFFECTIVE RO-RO TERMINAL OPERATIONS

Ph. D. C. Hasan GÖLER

MARITIME TRANSPORTATION and MANAGEMENT ENGINEERING DOCTORAL
PROGRAM

Thesis Supervisor: Prof. Dr. Ergün DEMİREL

November 2024, 182 Pages

Efficient parking space allocation is a critical aspect of optimizing terminal operations and ensuring smooth cargo flow at ports. However, the optimization of parking space allocation in Ro-Ro terminals presents unique challenges due to the limited parking area and the inability to expand port facilities within urban environments. As demand for Ro-Ro shipping continues to increase, it becomes imperative to address the challenge of extended dwell times in order to utilize these ports more efficiently and increase terminal capacity. This Ph.D. study addresses the gap in the literature on dwell time prediction specifically for Ro-Ro ports and investigates how accurate dwell time predictions can contribute to the optimization of parking space allocation. By accurately predicting dwell times, it becomes possible to better allocate parking spaces and improve the overall efficiency of Ro-Ro terminal operations.

The study begins with a comprehensive literature review, highlighting the scarcity of research on dwell time prediction in port operations. While a few studies exist for container ports and terminals, there is a notable absence of research for Ro-Ro ports. This study aims to fill this research gap and contribute to the existing literature on port operations.

Real data collected from Trieste Port, encompassing various operational parameters, vessel characteristics, and temporal features, is used to develop predictive models for dwell time estimation. Multiple machine-learning algorithms are tested and compared to identify the most accurate model for dwell time prediction in the Ro-Ro terminal context.

The research investigates the influence of accurate dwell time prediction on parking space optimization. Integration with optimization algorithms enables the dynamic allocation of parking spaces based on predicted dwell times, cargo characteristics, priority, and operational constraints. The proposed system aims to enhance parking space utilization,

reduce congestion, and improve resource allocation efficiency at Trieste Port's Ro-Ro terminal.

The study's findings contribute to the literature by addressing the scarcity of research on dwell time prediction in Ro-Ro ports. By demonstrating the importance of accurate dwell time prediction for parking space optimization, this study provides practical insights and recommendations for the optimization of parking space allocation in Ro-Ro terminals. The research outcomes serve as a valuable resource for industry professionals and researchers seeking to enhance port operations and address the unique challenges of Ro-Ro ports.



Keywords: Ro-Ro Terminal, Machine Learning, Dwell Time Prediction, Parking Space Optimization

ÖZET

VERİMLİ VE ETKİN RO-RO TERMİNAL OPERASYONLARI İÇİN PARK ALANI OPTİMİZASYONU

Ph. D. C. Hasan GÖLER

DENİZ ULAŞTIRMA ve İŞLETME MÜHENDİSLİĞİ DOKTORA PROGRAMI

Tez Danışmanı: Prof. Dr. Ergün DEMİREL

Kasım 2024, 182 Sayfa

Verimli park yeri tahsisi, terminal operasyonlarını optimize etmenin ve limanlarda sorunsuz yük akışı sağlamanın kritik bir yönüdür. Ancak, Ro-Ro terminallerinde park alanı tahsisinin optimizasyonu, sınırlı park alanı ve liman tesislerinin kentsel ortamlarda genişletilememesi nedeniyle benzersiz zorluklar ortaya çıkarmaktadır. Ro-Ro taşımacılığına olan talep artmaya devam ettikçe, bu limanları daha verimli kullanmak ve terminal kapasitesini artırmak için uzun bekleme süreleri sorununun ele almak zorunlu hale gelmektedir. Bu doktora çalışması, özellikle Ro-Ro limanları için bekleme süresi tahminine ilişkin literatürdeki boşluğu ele almakta ve doğru bekleme süresi tahminlerinin park yeri tahsisinin optimizasyonuna nasıl katkıda bulunabileceğini araştırmaktadır. Bekleme sürelerinin doğru bir şekilde tahmin edilmesiyle, park alanlarının daha iyi tahsis edilmesi ve Ro-Ro terminal operasyonlarının genel verimliliğinin artırılması mümkün hale gelmektedir.

Çalışma, liman operasyonlarında bekleme süresi tahmini üzerine yapılan araştırmaların azlığını vurgulayan kapsamlı bir literatür taraması ile başlamaktadır. Konteyner limanları ve terminalleri için birkaç çalışma mevcutken, Ro-Ro limanları için kayda değer bir araştırma eksikliği bulunmaktadır. Bu çalışma, bu araştırma boşluğunu doldurmayı ve liman operasyonlarına ilişkin mevcut literatüre katkıda bulunmayı amaçlamaktadır.

Trieste Limanı'ndan toplanan ve çeşitli operasyonel parametreleri, gemi özelliklerini ve zamansal özellikleri kapsayan gerçek veriler, bekleme süresi tahminine yönelik öngörücü modeller geliştirmek için kullanılmıştır. Farklı makine öğrenimi modelleri Ro-Ro terminali bağlamında uygulanarak bekleme süresi tahmini için en doğru modeli belirlemek üzere test edilmiş ve karşılaştırılmıştır.

Arařtırma, doęru bekleme süresi tahmininin park yeri optimizasyonu üzerindeki etkisini arařtırmaktadır. Optimizasyon algoritmalarıyla entegrasyon, park yerlerinin tahmini bekleme sürelerine, kargo özelliklerine, öncelięe ve operasyonel kısıtlamalara göre dinamik olarak tahsis edilmesini saęlar. Önerilen sistem, Trieste Limanı'nın Ro-Ro terminalinde park alanı kullanımını artırmayı, sıkışıklığı azaltmayı ve kaynak tahsis verimliliğini artırmayı amaçlamaktadır.

Çalışmanın bulguları, Ro-Ro limanlarında bekleme süresi tahmini konusundaki araştırma eksikliğini gidererek literatüre katkıda bulunmaktadır. Bu çalışma, park yeri optimizasyonu için doęru bekleme süresi tahmininin önemini ortaya koyarak, Ro-Ro terminallerinde park yeri tahsisinin optimizasyonu için pratik bilgiler ve öneriler sunmaktadır. Araştırma sonuçları, liman operasyonlarını geliřtirmek ve Ro-Ro limanlarının kendine özgü zorluklarını ele almak isteyen sektör profesyonelleri ve arařtırmacılar için deęerli bir kaynak teşkil etmektedir.

Anahtar kelimeler: Ro-Ro Terminali, Makine Öğrenmesi, Bekleme Süresi Tahmini, Park Alanı Optimizasyonu

DEDICATION

To my spouse, whose love, patience, and belief in me have been my strength.

To my children, Poyraz, Ömer and Ali Osman, whose smiles and innocence remind me of
the importance of perseverance and joy in the journey of life.

To my parents, whose sacrifices and wisdom have been my guiding light.

And to all those who believed in my dreams, this work is a testament to your faith and
support.

ACKNOWLEDGMENTS

I especially would like to thank my thesis advisor Prof. Dr. Ergun DEMİREL for his inspiring orientation, guidance, support and motivation throughout all my research and study. And I would like to thank my thesis examining committee members, Prof. Dr. Taner ALBAYRAK and Asst. Prof. Dr. Dinçer BAYER for their valuable contribution and support.

I would like to thank Kemal BOZKURT, Dr. Emir EJDER, Dr. Capt. Önder AYDIN, Ahmet DERİCİOĞLU and Cankut KASBER for their valuable help and support.

Finally, I would like to thank my dear wife Gamze GÖLER, who always support and encourage me throughout all my long doctorate study.

TABLE OF CONTENTS

ABSTRACT.....	I
ÖZET	III
DEDICATION	V
ACKNOWLEDGMENTS	VI
TABLE OF CONTENTS	VII
LIST OF TABLES	XII
LIST OF FIGURES	XIII
LIST OF ABBREVIATIONS	XV
1. INTRODUCTION	1
1.1. The Aim of the Thesis	6
1.1.1 Background and Context	6
1.1.2 Overview of Trieste Port's Ro-Ro Terminal	7
1.1.3 Problem Statement	9
1.1.4 Research Objectives.....	10
1.1.5 Significance of the Study	11
1.2. Definition and Operational Characteristics of Ro-Ro Terminals.....	13
1.2.1 Characteristics of Ro-Ro Terminals	13
1.2.2 Operations and Processes at Ro-Ro Terminals.....	14
1.2.2.1 Voyage planning	14
1.2.2.2 Cargo Operations	15
1.2.2.3 Customs and Security	17
1.2.2.4 Terminal Operations	18
1.2.2.5 Yard Management.....	18
1.2.3 Challenges in Ro-Ro Terminals.....	19
1.2.3.1 Parking Space Management.....	19
1.2.3.2 Dwell Time Prediction.....	21
1.2.3.3 Congestion and Traffic Flow	21
1.2.3.4 Environmental Regulations and Compliance	21
1.2.3.5 Technological Integration and Data Management	21
1.2.3.6 Labour and Workforce Management.....	22
1.2.3.7 Safety and Security	22
1.2.3.8 Vessel Turnaround Time.....	22
1.2.3.9 Handling Increasing Volumes	22

1.2.4	Opportunities in Ro-Ro Terminals	23
1.2.4.1	Technology Integration	23
1.2.4.2	Sustainability Initiatives	24
1.2.4.3	Expansion and Diversification	24
1.2.4.4	Collaborative Partnerships	24
1.3.	Parking Space Optimization in a Ro-Ro Terminals	24
1.3.1	Automated Parking System	25
1.3.2	Smart Traffic Management	25
1.3.3	Advanced Parking Layout Design.....	26
1.3.3.1	Compact Layouts	26
1.3.3.2	Modular Parking Structures	29
1.3.3.3	Flow Optimization.....	29
1.3.4	Parking Space Allocation Algorithms.....	30
1.3.5	Use of Vertical Space	30
1.3.6	Importance of Dwell Time in Parking Space Optimization.....	31
1.3.7	Techniques to Reduce Dwell Time	31
1.3.7.1	Real-Time Monitoring	32
1.3.7.2	Traffic Flow Optimization	32
1.3.7.3	Priority-Based Parking Allocation.....	33
1.3.7.4	Advanced Planning and Coordination	33
1.3.7.5	Use of Technology and Automation	34
1.4.	Dwell Time Prediction and Machine Learning Applications.....	35
1.4.1	Dwell Time Prediction	36
1.4.1.1	Importance of Dwell Time Prediction	36
1.4.1.2	Factors Influencing Dwell Time.....	37
1.4.1.3	Methods for Dwell Time Prediction.....	38
1.4.1.4	Challenges in Dwell Time Prediction.....	39
1.4.1.5	Future of Dwell Time Prediction.....	39
1.4.2	Machine Learning Applications	39
1.4.2.1	Supervised Learning	40
1.4.2.2	Unsupervised Learning.....	41
1.4.2.3	Semi-supervised Learning.....	41
1.4.2.4	Reinforcement	41
1.4.3	Classification Analysis.....	43
1.4.3.1	Naive Bayes (NB).....	45
1.4.3.2	Linear Discriminant Analysis (LDA).....	46
1.4.3.3	Logistic regression (LR).....	46

1.4.3.4	K-nearest neighbours (KNN)	47
1.4.3.5	Support vector machine (SVM)	48
1.4.3.6	Decision tree (DT).....	49
1.4.3.7	Random forest (RF)	50
1.4.3.8	Adaptive Boosting (AdaBoost)	50
1.4.3.9	Gradient Boosting Machines (GMB).....	51
1.4.3.10	Stochastic gradient descent (SGD).....	52
1.4.3.11	Rule-based classification.....	52
1.4.3.12	Neural Networks.....	53
1.4.4	Regression Analysis	53
1.4.4.1	Simple and multiple linear regression	54
1.4.4.2	Polynomial regression	55
1.4.4.3	LASSO and ridge regression	55
1.4.5	Effects of Dwell Time on Ro-Ro Terminal Operations	56
1.4.6	Importance of Dwell Time in the Context of Energy Efficiency and Sustainability.....	57
1.5.	Analytical Hierarchy Process (AHP).....	58
2.	LITERATURE REVIEW	60
3.	MATERIALS AND METHODS	70
3.1.	Introducing the Dataset	70
3.2.	Introducing the Method.....	72
3.3.	Selecting the Machine Learning Models	74
3.3.1	Selection of Algorithms for AHP Application.....	74
3.3.1.1	Rationale for Selecting the 10 Algorithms for AHP	75
3.3.1.2	Application of AHP to the Selected Algorithms.....	77
3.3.1.3	Defining the Criteria	78
3.3.1.4	Pairwise Comparisons of Criteria	79
3.3.1.5	Results of AHP.....	85
3.3.2	Comparative Analysis.....	89
3.4.	Data Collection Process	90
3.4.1	Source Description	90
3.5.	Data Preprocessing.....	95
3.5.1	Purpose	95
3.5.2	Uploading Libraries and connect to google drive to upload raw data....	95
3.5.3	Importing Raw data.....	96
3.5.4	Data Cleaning and preprocessing.....	97
3.5.5	Date Conversion	98

3.6.	Data Transformation	99
3.6.1	Categorizing Dwell Time.....	99
3.6.2	Dropping unnecessary columns (Dwell time – Hours)	100
3.6.3	Categorizing cargo weights	101
3.6.4	Dropping unnecessary columns (Cargo Tare Weight).....	102
3.7.	Feature Engineering.....	103
3.7.1	Extracting data with new date relevant columns.....	103
3.7.2	Preprocessing Pipeline for Numeric and Categorical data	104
3.7.3	Completion of Pipeline	107
3.7.4	Label Encoding	109
3.8.	Training the model, setting test and validation datasets	111
3.8.1	Training, Validation and Testing process of Random Forest Model....	111
3.8.1.1	Import Libraries for Training process	111
3.8.1.2	Defining the Feature Matrix ‘x’ and Target Variable ‘y’	111
3.8.1.3	Splitting the Temporary Set into Validation and Test Sets	113
3.8.1.4	Training the Random Forest model.....	114
3.8.2	Process of XGBoost Model.....	116
3.8.3	Process of Light GBM Classifier	117
3.8.4	Process of CatBoost Model.....	118
3.8.5	Process of SVM (Support Vector Machines) Model.....	119
3.8.6	Process of Logistic Regression Model	120
3.8.7	Process of K-Nearest Neighbours Model	121
3.9.	Measuring Model Performance	122
3.9.1	Accuracy	122
3.9.2	Precision.....	122
3.9.3	Recall.....	123
3.9.4	F1 score	124
3.9.5	Summary of Metrics	124
3.10.	Ethical Considerations	125
3.10.1	Ethics Approval.....	125
3.10.2	Data Usage and Security.....	125
4.	RESULTS AND DISCUSSION	126
4.1.	Overview of Model Performance - Random Forest Classifier.....	126
4.2.	Overview of Model Performance - XGBoost Classifier.....	127
4.3.	Overview of Model Performance – LightGBM Classifier.....	128
4.4.	Overview of Model Performance – CatBoost Classifier.....	129

4.5. Overview of Model Performance – SVM Classifier (Support Vector Machines).....	130
4.6. Overview of Model Performance – Logistic Regression.....	131
4.7. Overview of Model Performance – KNN results (K-Nearest Neighbours)..	132
4.8. Interpretation of Results.....	133
4.8.1 Category 1 (First 24 Hours)	133
4.8.2 Category 2 (24 - 48 Hours)	134
4.8.3 Category 3 (48 - 72 Hours)	135
4.8.4 Category 4 (> 72 Hours)	137
4.8.5 Evaluation of overall results	138
4.8.6 Best Model for Each Class	138
4.8.7 Overfitting and Underfitting Evaluation.....	139
4.8.8 Validation of AHP in a Data-Driven Context	140
4.9. Discussion on results	141
4.9.1 Approach for Improvement.....	141
4.9.2 Parking slots Optimization Application	142
4.9.3 Best Model for first 24 hours and higher than 72 hours	142
4.9.4 Novel Contribution	143
4.10. Limitations of the Models/Study	143
4.10.1 Future Directions	145
5. CONCLUSION	146
5.1. Improvement of Ro-Ro Terminal Efficiency.....	146
5.2. Key Outputs of the model developed.....	146
5.3. Practical Implications	147
5.4. Contribution to the Literature	148
5.5. Suggestions for the Future Studies	148
5.6. Summary.....	149
REFERENCES.....	150
CURRICULUM VITAE.....	163

LIST OF TABLES

Table 1.1 : Types of machine Learning algorithms.	42
Table 3.1 : Data set key features	71
Table 4.1 : Result of Random Forest Classifier	126
Table 4.2 : Performance metrics of Random Forest Classifier	126
Table 4.3 : Result of XGBoost Classifier	127
Table 4.4 : Performance metrics of XGBoost Classifier	127
Table 4.5 : Result of LightGBM Classifier.....	128
Table 4.6 : Performance metrics of LightGBM Classifier.....	128
Table 4.7 : Result of CatBoost Classifier.....	129
Table 4.8 : Performance metrics of Catboost Classifier	129
Table 4.9 : Result of SVM Classifier.....	130
Table 4.10 : Performance metrics of SVM Classifier.....	130
Table 4.11 : Result of Logistic Regression.....	131
Table 4.12 : Performance metrics of Logistic Regression	131
Table 4.13 : Result of KNN Classifier.....	132
Table 4.14 : Performance metrics of KNN Classifier.....	133
Table 4.15 : Comparison of models for first 24 hours prediction results	134
Table 4.16 : Comparison of models for 24 to 48 hours prediction results.....	135
Table 4.17 : Comparison of models for 48 to 72 hours prediction results.....	136
Table 4.18 : Comparison of models for > 72 hours prediction results	137

LIST OF FIGURES

Figure 1.1 : Methodological Workflow of Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF).....	5
Figure 1.2 : Dwell time per week development of Trieste STT Terminal.	8
Figure 1.3 : Exemplification of cargo details at a Ro-Ro terminal.....	16
Figure 1.4 : Example of cargo area management at a Ro-Ro terminal.....	20
Figure 1.5 : Machine Learning Process.	35
Figure 1.6 : Machine Learning algorithms.	40
Figure 1.7 : An example of a predictive model phases.....	43
Figure 1.8 : Example of Naïve Bayes Algorithm results.....	46
Figure 1.9 : K- nearest Neighbour plotting chart.....	48
Figure 1.10 : Support Vector Machine diagram.	49
Figure 1.11 : An example of a decision tree structure.	49
Figure 1.12 : An example of a random forest structure.	50
Figure 1.13 : Differences between classification and regression	54
Figure 3.1 : Steps of Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF) approach.	73
Figure 3.2 : Criteria defined in Super Decision Software.	79
Figure 3.3 : Weights of Criteria defined in Super Decision Software.....	80
Figure 3.4 : Comparison of algorithms based on Efficiency Criteria.....	82
Figure 3.5 : Comparison of algorithms based on Generalization Criteria.....	83
Figure 3.6 : Comparison of algorithms based on Model Complexity Criteria.....	84
Figure 3.7 : Comparison of algorithms based on Data Handling Criteria.....	85
Figure 3.8 : Results of AHP.....	86
Figure 3.9 : Milos Framework.....	91
Figure 3.10 : Tableau Business Intelligence Software, view of Raw Data.	92
Figure 3.11 : Detailed Methodology Flowchart of Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF) model.....	94
Figure 3.12 : Model Train & Test & Validation Cycle.	95
Figure 3.13 : Importing Python Libraries & connect to google drive folder.....	96
Figure 3.14 : Importing Raw data.....	96
Figure 3.15 : Data cleaning and preprocessing.....	97
Figure 3.16 : Date conversion.....	98
Figure 3.17 : Categorizing dwell times.....	99
Figure 3.18 : Count of each dwell time category.....	100
Figure 3.19 : Drop “Dwell time – Hours” column.	101
Figure 3.20 : Categorizing Cargo weights.....	101
Figure 3.21 : Count of each Cargo weight category.....	102
Figure 3.22 : Drop “Cargo Tare Weight” column.	103
Figure 3.23 : Create new date relevant columns.....	104
Figure 3.24 : Importing required libraries for Pipeline processing.	105
Figure 3.25 : Defining Numeric and Categorical Features.....	106
Figure 3.26 : Imputing missing numeric values.	106
Figure 3.27 : Constructing Preprocessor.	107
Figure 3.28 : Importing Libraries for completion of pipeline and label encoding.	108
Figure 3.29 : Code for completion of pipeline for machine learning models.....	109
Figure 3.30 : Code of label encoding.....	110

Figure 3.31 : Code of label encoding.....	111
Figure 3.32 : Definition of Feature Matrix X and Y variables.	112
Figure 3.33 : Splitting Validation and Test Datasets.	113
Figure 3.34 : Code to run training process of model and provides results of predictions.....	114
Figure 3.35 : Code of XGBoost Model (Training, Validation and Testing).	117
Figure 3.36 : Code of Light GBM Classifier (Training, Validation and Testing).....	118
Figure 3.37 : Code of CatBoost Model (Training, Validation and Testing).....	119
Figure 3.38 : Code of SVM Model (Training, Validation and Testing).....	120
Figure 3.39 : Code of Logistic Regression Model (Training, Validation and Testing)....	121
Figure 3.40 : Code of K-Nearest Neighbours (Training, Validation and Testing).....	121



LIST OF ABBREVIATIONS

ADABOOST	Adaptive Boosting
AHP	Analytical Hierarchy Process
AI	Artificial Intelligence
ANN	Artificial Neural Network
ANOVA	Analysis of Variance
CDT	Container Dwell Time
DT	Decision Tree
DT	Dwell Time
EU	European Union
GBM	Gradient Boosting Machine
IOT	Internet of Things
KNN	K-Nearest Neighbours
KPI	Key Performance Indicator
LASSO	Least absolute shrinkage and selection operator
LDA	Linear Discriminant Analysis
LIGHT GBM	Light Gradient Boosting Machine
LR	Logistic Regression
ML	Machine Learning
MLFNN	Multilayer Feedforward Neural Network
MMSF	Model Synergistic Framework for Machine Learning Optimization
NB	Naïve Bayes
RF	Random Forest
RIDOR	Ripple Down Rule learner
RIPPER	Repeated Incremental Pruning to Produce Error Reduction
Ro-Ro	Roll On – Roll Off
SGD	Stochastic gradient descent
SVM	Support Vector Machine
TOS	Terminal Operating System
UND	Uluslararası Nakliyeciler Derneği
XGBOOST	Extreme Gradient Boosting

1. INTRODUCTION

Ro-Ro terminal operations are a vital part of the multinational maritime transportation system, providing reliable and competitive transport services for wheel-based cargo. (Tiago A. Santos and C. Guedes Soares, 2020). Ro-Ro terminals are critical in areas with a significant concentration of vehicle or wheeled cargo traffic, particularly Europe. Existing Ro-Ro terminals, on the other hand, have been limited by operational inefficiencies and resource bottlenecks stunting a more organized development in this sector. Although Ro-Ro transportation has plenty of benefits, the lack of operational efficiency and resource constraints in Ro-Ro terminals have been very bottleneck to its overall advancement (Arof, 2018).

Europe has the most important Ro-Ro ports which integrate its regional logistics and supply chain (Lennerfors, 2022). In this regard, the importance of Trieste Port in Italy comes to limelight as it is located strategically and has potentials. Located at the northernmost tip of the Adriatic Sea, it represents an important crossroad between the maritime routes from the East and land routes across Europe (Borruso Giuseppe and Cociancich, 2023). In particular, Trieste's Ro-Ro terminal handles a significant volume of trailer traffic which requires an optimal utilization of available parking space to improve operational efficiency.

Trieste Port, which acts as a key logistics base for Europe, also supports Türkiye's export-oriented economy. The port serves as a main entry site for Turkish exports into the European market, in particular by Ro-Ro vessels (European Commission, 2023). Trieste Port has emerged as a strategic junction on the Europe-Türkiye trade axis that connects Italy to Türkiye, especially with regards Ro-Ro cargo), owing not only in part to Italian -Turkish alliance but also due Turkish premier position among other producer of Ro-Ro cargoes. (Demirel and Demir, 2024).

Further, the Ro-Ro terminal of Trieste Port represents an unusual compound because it has diversified commodities. (Torbianelli, 2000). From cars and trucks to trailers, the heavy machineries are some of wheeled cargo handled by a terminal. The way these diverse cargo

types are managed efficiently, and the necessity of rapidly loading/unloading operations create significant challenges for port operations. (Kurtuluş and Çetin, 2020).

Despite its central role, size and importance of Trieste Ro-Ro terminal is typically congested due to parking allocation issue. (Stamatovic, 2020). Considering the terminal's importance in the current logistics network and its potential impact on the European and Turkish economies, it is crucial to improve its operational efficiency.

There is a challenge related to parking spaces management in Ro-Ro (roll-on/roll-off) terminals, for example as previously stated with Trieste port, where the focus is on the optimization of trailer storage. (Tadini Marcello and Borruso, 2022). Existing research has explored various aspects of Ro-Ro terminal operations, including storage location assignment (Russo et al., 2023), loading and unloading operations and storage yard management in transshipment ports (Stamatovic, 2020). However, the specific issue of machine learning driven dwell time prediction and using this for better parking space optimization has not been extensively addressed.

Ro-Ro terminals serving the automotive industry have been found to be very inefficient with diminishing storage capacity, also hampering its usage in other markets and development. (Abourraja et al., 2023). Unlike import-export container terminals, which perform loading and unloading operations over time without interruption throughout the year, transshipment ports must bundle these jobs within a short period of time, complicating storage yard management. Currently, the research on storage yard management mainly concentrates on dividing yards, designing yard templates and short-term allocation strategies additional stacking slots are shared adaptively when containers cannot be stacked in a more reasonable way yet should locate at their position for quick loading to vessel. (Zhang et al., 2023). Another research conducted about the arrival of cars at a yard for Ro-Ro terminals has suggested solving the storage location assignment problem to enhance ship-loading efficiency and improve storage efficiency in the terminal. (Luo, 2024).

Machine learning has revolutionized the ports in optimizing efficiency as well as decision-making. This is realized by using technologies that automate functions, analyse big data instantly and give information with deep insights to help ports serve on complex operations. By using machine learning for everything from predictive maintenance of port equipment to

dynamic scheduling and resource allocation, ports are able to work more efficiently resulting in smoother operation with lower operational costs and increased throughput. Ports can anticipate challenges using machine learning algorithms that respond to both historical and real-time data, optimize cargo handling as well as their berthing etc., which is not possible if the right access and correspondence are managed among ports, shipping lines or other stakeholders. This leap in technology is revolutionizing traditional port management and addressing the surfacing needs all while keeping an eye out on creating a resilient ecosystem with sustainable operations. These innovations have the potential to transform Ro-Ro terminal management, particularly by addressing operational challenges like parking space utilization. One of the key benefits of machine learning in this context is its ability to predict dwell time – the duration trailers spend at the terminal. Accurate dwell time predictions enable better planning and resource allocation, minimizing congestion and enhancing overall terminal performance. Machine learning-driven dwell time prediction can play a crucial role in enhancing parking space utilization by enabling more accurate forecasts of trailer stay durations. (Abourraja et al., 2022). This study aims to explore parking space optimization in the Trieste port's Ro-Ro terminal by employing machine learning techniques for dwell time prediction, allowing port operators to make data-driven decisions and improve the efficiency of terminal operations.

In this area, the ability to accurately predict dwell time - the time cargo spends within the terminal - is pivotal in this regard. An accurate and precise prediction of dwell time facilitates better planning and scheduling of operations, leading to improved parking space allocation, reduced congestion, and ultimately, improved terminal performance (Hassan et al., 2020). This study thus aims to contribute in this direction by exploiting machine learning methods for predicting dwell times of trailers for optimising parking spaces at the Trieste port's Ro-Ro terminal. The accurate prediction of dwell times will enable port operators to make more informed decisions regarding the allocation of parking spaces, which will enable to improve utilization of available resources and increase overall terminal efficiency (De Wibowo Muhammad Sidik et al., 2020a).

Literature is scarce for dwell time prediction in shipping ports, especially in the Ro-Ro terminal context. While previous research has explored storage location assignment and yard

management in Ro-Ro terminals, the specific issue of dwell time prediction and parking space optimization through dwell time prediction remains largely unexplored.

Current studies in literature focus mainly on container ports and terminals (De Wibowo Muhammad Sidik et al., 2020; Feng et al., 2020; Hassan et al., 2020; Smith, 2021). Although the findings from these studies provide insights, they cannot be directly applied to Ro-Ro terminals due to the unique characteristics of Ro-Ro operations, such as diverse cargo types, loading sequence, and operational constraints. Moreover, the study by (Martín-Lorenzo et al., 2023) presented a decision-support system for managing truck stays at seaports using machine learning and simulation models to predict truck dwell times and reduce congestion in the Port of Santander. While this study demonstrates the value of predictive tools in port management, it focuses on truck arrival and transit times of container port, where trucks enters to port from gates to bring or pick up containers, rather than trailer dwell times at Ro-Ro terminals which are discharged from vessels and waiting long time at ports before leaving the terminal and involves distinct operational dynamics. Moreover, the Trieste port presents a specific case study, and the insights gained from this research can contribute to improve the operational efficiency of this strategically important Ro-Ro terminal in particular, and same approach can be used and be a best practice for similar terminals in the broader Ro-Ro transportation ecosystem.

This study aims to address the gap in the existing literature by developing a machine learning-driven approach to predict dwell times of trailers at the Trieste port's Ro-Ro terminal and highlight importance of how dwell time prediction can help and support terminal operators for better parking space optimization.

To achieve this, the study uses a clear, step-by-step method for developing and testing machine learning models, as shown in Figure 1.1. At the core of this approach is the Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF). MMSF works by first selecting the best models using the Analytic Hierarchy Process (AHP), then further testing and combining them to find the most accurate predictions. This framework helps ensure that the final models provide better results for predicting dwell times, ultimately supporting more efficient parking management at the terminal.

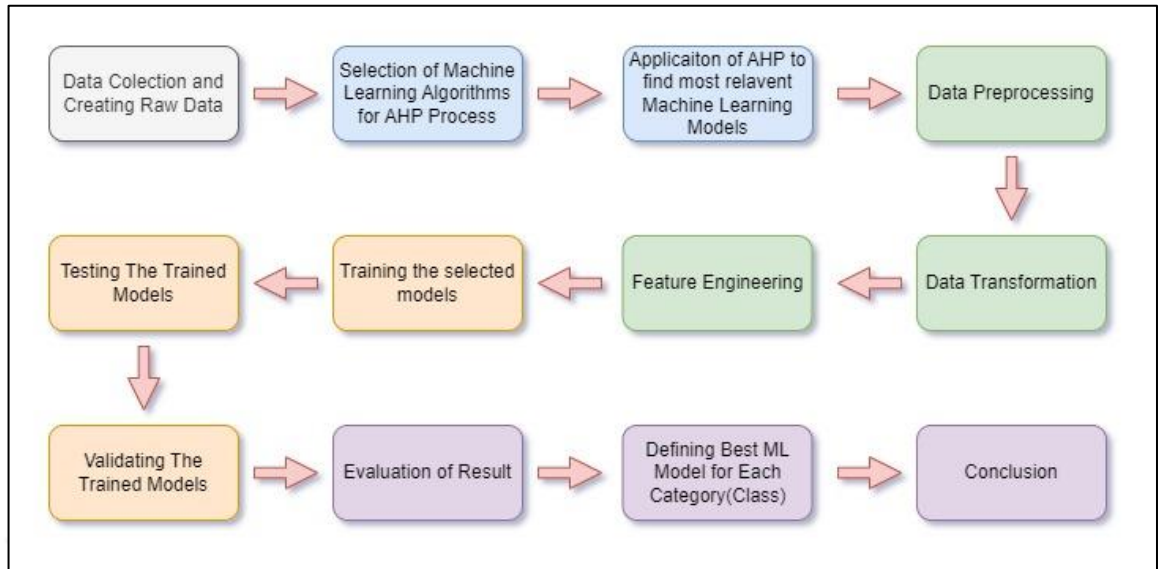


Figure 1.1 : Methodological Workflow of Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF).

The remainder of this thesis is organized as follows:

In Section 1, the Aim of the Thesis is described in detail after **Introduction**. This is followed by a comprehensive discussion of the characteristics, challenges, and opportunities associated with Ro-Ro terminals, current parking space management practices, and an exploration of dwell time prediction using machine learning models and algorithms.

Section 2 includes the **Literature Review**, encompassing studies related to Ro-Ro terminal operations, parking space optimization and machine learning applications.

Section 3 provides the **Materials and Methods**, including data collection and model selection subsections among others followed by developing dwell time prediction.

Section 4 presents **Results and Discussion** analysing the findings of the study and their implications for parking space optimization.

And lastly, the Section 5 concludes the thesis by summarizing our key findings and pointing out future research directions.

1.1. The Aim of the Thesis

1.1.1 Background and Context

Ro-Ro terminal operations are an integral part of the global maritime transportation system and provide a high-performing and unique solution for handling rolling goods. The (roll-on roll-off) Ro-Ro terminals are very important in regions where there is a large volume of wheeled cargo as vehicles, such the European territories.

Some of the key Ro-Ro ports in Europe, play vital role for the logistics and supply chain network. In Italy, the Port of Trieste is distinguished for its strategic position and operational capacity. At the northern end of the Adriatic Sea, provides vital link in transit corridors linking maritime routes to those traversing across Europe by land.

Trieste Port not only important because it performs as a crucial logistics centre for Europe but also fulfils a significant role for Türkiye's export-driven economy. The port serves as an important gate for Turkish goods to the European market and especially those which are transported by Ro-Ro vessels. The strategic co-operation between Italy and Türkiye, combined with Türkiye's strong position as a major Ro-Ro cargo exporter, has increased the significance of Trieste Port as an essential hub in the trade corridor between Europe and Türkiye.

Furthermore, the Ro-Ro terminal at Trieste Port is characterised by different types of cargo traffic. The terminal specializes in the handling of all kinds wheeled cargo: cars, trucks, trailers; heavy equipment. And to trying to manage a variety of cargo loading and unloading quickly results difficulties for port operations.

In spite of its importance, the Trieste Ro-Ro terminal frequently experiences congestion and inefficiencies, mostly due to inadequate parking space allocation. Considering the terminal's significance in the logistics network and its potential impact on European and Turkish trades, it is significant requirement to improve operational efficiency.

In this respect, being able to accurately predict dwell time (the amount of cargo remains within the terminal) becomes crucial. Accurate prediction of dwell time supports better planning and operation scheduling; thus, it can contribute to increase parking space utilization significantly, decrease congestion ultimately leading to a higher level of service. Therefore, this study will concentrate on the development of reliable predictive model for dwell time estimation in Ro-Ro terminal operations at Trieste port, aiming to address significant operational challenges and optimize Ro-Ro terminals' operations regionally as well as globally.

1.1.2 Overview of Trieste Port's Ro-Ro Terminal

At a strategic crossroads of maritime routes between Europe and Türkiye, the Port is critical in supporting trade and commerce. The Ro-Ro terminal of the Port of Trieste is considered strategic, given its role in receiving rolling goods whose handling allows for rapid switching from sea to road and rail transport.

Trieste Port's Ro-Ro Terminal holds significant importance for both Europe and Türkiye's trade, serving as a crucial hub for Ro-Ro operations between these regions. Trieste is known as the main gateway in Turkish Ro-Ro routes to reach European Union countries with its highly efficient shipping lines. With the port being strategically located, and having an excellent connection of other networks into its inland configuration, it is a vital link in the trade supply chain.

Europe's trade with Türkiye has been experiencing substantial growth, with increasing volumes of cargo being transported via Ro-Ro services. According to statistical data, Trieste Port handles approximately 72% of Ro-Ro transports from Türkiye to Europe, highlighting its pivotal role in facilitating trade flows (Unal et al., 2022). The port serves as a primary gateway for Turkish shippers aiming to connect with EU markets faster. Given that Ro-Ro transport is a short sea shipping, it could also represent an attractive alternative to road transport, aligning with the EU's goals of reducing greenhouse gas emissions and promoting sustainable transportation (European Commission, 2023).

However, the Ro-Ro Terminal at Trieste Port faces with challenges related to congestion problems caused by long dwell times. Dwell time refers to the duration of cargo remains within the port before it is either loaded onto a vessel or released for further transportation. The extended dwell times experienced at Trieste Port's Ro-Ro Terminal have a direct impact on congestion levels, resulting in operational inefficiencies, delays, and potential disruptions to the supply chain.

Statistical data in Figure 1.2 presents the magnitude of the congestion issue at Trieste Ro-Ro Terminal. While terminal has target to achieve average 1,5 days dwell time, the actual dwell time exceeds three days, reaching up to 4.5 days during peak seasons. These long dwell times significantly creates difficulty and inefficiency for the terminal operations and limit parking space availability for incoming cargo. If cargo does not leave the terminal within the expected dwell time, the terminal becomes congested, which leads to space constraints and delays in vessel discharge. Then congestion creates further consequences, affecting cargo transit times, results inefficiencies in Ro-Ro transport operations, and potentially driving transporters to favour road transport alternatives due to extended waiting times at Trieste Port.

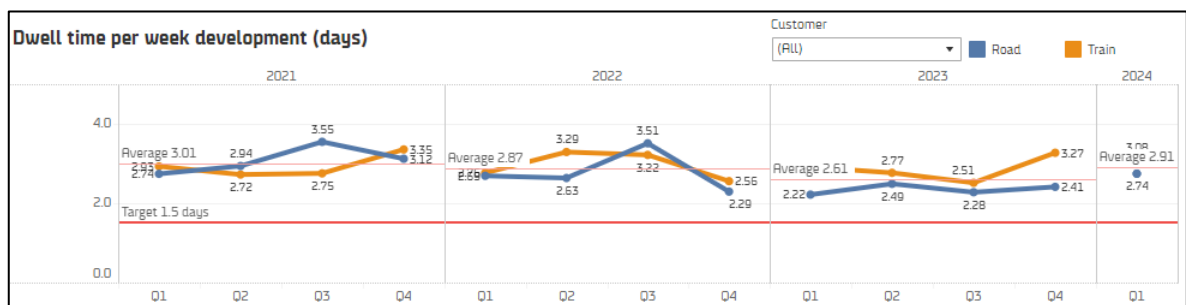


Figure 1.2 : Dwell time per week development of Trieste STT Terminal.

Such congestion problems resulting from long dwell activities pose a critical challenge which must be addressed with the optimal solutions of terminal operations, reduced handling times for cargo and to manage mobility more efficiently. Solving these challenges is crucial with respect not only to the efficiency and competitiveness of Ro-Ro transport, but also for maintaining overall supply chain performance and reliability.

The proposed research aims to address these challenges by developing a predictive model for dwell time and integrating this model into a parking space optimization framework. By doing so, it aims to increase port productivity, reduce congestion and improve service quality at Trieste Port's Ro-Ro terminal.

The primary challenge in ro-ro transportation lies in the limited storage space and capacity constraints of ports. As road transport gradually transitions to short sea shipping, the demand for ro-ro transport has increased significantly and will continue to increase. However, due to the limited expansion opportunities of the Ro-Ro ports, which are often located within urban areas, capacity problems have emerged, leading to congestion and extended waiting times.

1.1.3 Problem Statement

The major issue in ro-ro transport is the availability of limited space and ports tend to have capacity constraints, particularly in high-volume ports such as the Trieste Port. The nature of Ro-Ro operations and the various cargo types being handled contribute to the complexity of this problem. Effective parking space allocation is not only required to maintain operational efficiency, but also plays a major role to reduce terminal congestion and ensure seamless cargo flow.

The fundamental problem is that one of the key elements affecting parking allocation in a Ro-Ro terminal namely, type and size of cargo, arriving and departing vessel schedules and the operational constraints of the terminal. Among these factors, the dwell time – the time a cargo unit remains within the terminal – is of particular importance. The higher the dwell time, longer is parking space blocked and less number of terminal capacity and space available for incoming cargo.

Despite its significance, dwell time in Ro-Ro terminals can be difficult to forecast reliably because of its dependency to a wide range of variables, such as operational delays, changes in vessel schedules, deadline of cargo delivery to end customers and fluctuations in cargo volumes. This difficulty of predictability complicates the task of parking space allocation, mostly due to holding onto slots that might not be used, increased terminal congestion, and decreased operational efficiency.

Moreover, most research and predictive models have been developed for container terminals and may not be directly applicable to the unique characteristics and challenges of Ro-Ro terminals.

This research aims to address the specific problem of parking space optimization in Ro-Ro terminals by developing a predictive model for dwell time that is specifically designed for the unique operational context of Ro-Ro terminals, using the Trieste Port as a case study. By improving the accuracy of dwell time prediction, this study aims to contribute to the optimization of parking space allocation to reduce overall dwell time of terminal, thereby enhancing the overall efficiency of Ro-Ro terminal operations.

1.1.4 Research Objectives

The aim of this study is to increase port operational efficiency by reducing dwell time.

Research objectives:

- Grouping and structuring real data effecting dwell times
- Developing model based on this real data
- Performing test and acceptance of this model
- Fill gap in literature about ro-ro port optimizations and dwell time prediction

In order to solve these problems and the rising demands, port capacity will need to be addressed by reducing wait times and general improvements in efficiency. The dwell time, which means the stay of a cargo at port, is one such key parameter that affects both the operations and the specially concerning for cargo owners. Therefore, by analysing the drivers of dwell time and with the assistance of advanced technologies such as machine learning techniques to generate effective predictions, it becomes possible to predict the expected dwell time of cargo that has not yet arrived at the port.

The ability to predict dwell time provides valuable advantages to optimize port operations, parking allocation, customs procedures, and other logistical planning. By accurately

forecasting how long cargo is expected to stay at the port, these activities can be performed more effectively, which will result increased capacity in ports and reduced congestion.

In conclusion, the use of machine-learning approaches to predict dwell time is an important step forward in dealing with the capacity constraints experienced by ro-ro terminals. By using accurate predictions based on historical data, port management teams can optimize their operations to prevent congestion at ports which would lead to increasing effective capacity and improve operational efficiency.

The overall aim of this research is to investigate how accurate dwell time prediction can contribute to parking space optimization to reduce dwell time at Trieste Port's Ro-Ro terminal, a crucial node in the Europe-Türkiye trade corridor. To achieve this aim, the research will develop a predictive model for dwell time at the Ro-Ro terminal, which will then be possible to integrate this model into a parking space optimization framework and increase effectiveness of parking space allocation.

1.1.5 Significance of the Study

Optimization of port operations, mainly in terms of the work related to loading and unloading has been researched extensively. Despite the wealth of research in this area, there is only a limited amount of academic knowledge on Ro-Ro terminals which are the essential parts of global maritime transport systems. Moreover, dwell time prediction studies in the literature for ports and terminals predominantly focus on container ports, with a lack of specific studies dedicated to Ro-Ro ports, an area that presents significant opportunities for contribution and operational improvement.

Therefore, the present study is of both academic and practical importance. Primarily, it is poised to contribute to the academic field by providing insights into the application of predictive models for dwell time in the context of Ro-Ro terminals. While there is a limited literature dealing with terminal allocation in container terminals based on dwell time, this paper represents the first application of such modelling approach into Ro-Ro terminals. Accordingly, the findings of this study will offer an original addition to the related literature and a deeper insight into favourable strategies for optimizing port operation. Not only does

it fill a gap in the field but is concrete and actionable for how Ro-Ro ports can improve their operational efficiency.

Moreover, this study will be the first to utilize a machine learning approach for estimating dwell time of Ro-Ro terminals using historical data. As such, it will provide methodological toolkit available for future research in the field.

This study will significantly help to improve the performance of Ro-Ro port beyond its academic contributions. The proposed model aims to reduce dwell times and port congestion which are increasing due to higher volumes of global sea trade, thereby improving service efficiency and, potentially, increasing revenue for customers and terminals.

Furthermore, this study specifically targets Trieste Ro-Ro Terminal which is a key bottleneck in the Europe-Türkiye trade corridor, grappling with congestion issues. By proposing solutions to reduce the unexpected waiting times of cargo at the port, it aims to address the congestion problem by providing possible measures for reducing cargo waiting times in inconceivable queues and therefore enhancing operational efficiency and contributing to the economic well-being of the region.

Consequently, this research fills a crucial research gap by providing valuable insights into the optimization of Ro-Ro terminal operations while it pioneers the prediction of dwell times in Ro-Ro terminals. To the best of our knowledge, it is the first research endeavour dedicated to this aspect. By developing and applying machine learning models to predict dwell times based on historical data, this research presents a novel approach tailored specifically to the unique characteristics of Ro-Ro terminal operations.

This study stands to contribute to not just the academic field but also the broader maritime industry by offering novel and practical solutions for improving operational efficiency at Ro-Ro ports.

1.2. Definition and Operational Characteristics of Ro-Ro Terminals

Ro-Ro terminals are specially constructed facilities for the handling of Ro-Ro vessels. Ro-Ro vessels are specially designed to comfortably carry wheeled cargo, which can include anything that is loaded and unloaded on its own wheels, including cars, trucks, semi-trailers, trailers and railway wagons (Abourraja et al., 2021). Ro-Ro terminals have different characteristics and operating procedures to better serve these specialised cargoes and related operations.

Roll-On/Roll-Off (Ro-Ro) terminals are vital in the worldwide shipping and logistics chain. They thus allow for the main line between maritime transport and land-based distribution for wheeled cargo in virgin transfer (Chen et al., 2021a). Ro-Ro is the ability of cargo to roll on and off a vessel under its own power and is the one characteristic that distinguishes a Ro-Ro terminal from other types of cargo facilities such as a container terminal or a bulk terminal.

Rolled on an Rolled off facilities and services are essential in the transporting of vehicles, trucks, trailers, buses, and, sometimes, railcars. It is different way to handle the cargo as it makes loading and unloading process shorter and easier, cargo is loaded and discharged using large doors and ramps (Wang, 2024). This approach reduces the requirements for cranes or other sophisticated lifting gear, consequentially leading to quicker turnaround times and lowered handling costs.

1.2.1 Characteristics of Ro-Ro Terminals

Ro-Ro terminals are required to store a large amount of vehicular movement and need huge torch space for traffic movement. These terminals are often characterized by wide parking areas, vast road networks, and numerous loading/unloading ramps, to ensure seamless functioning in regular use (Iannone et al., 2016). This has been done to provide a more orderly queue from when the vehicles enter the docks, pass through the storage area, before finally being loaded onto vessels, all while keeping the process as safe as possible and at its most efficient (Unal et al., 2022).

There is a fair amount of infrastructure that supports Ro-Ro terminals in the form of the following:

- *Berths and Ramps:* Ro-Ro terminals have specialised berths that accommodate the unique design of Ro-Ro vessels. These berths are typically equipped with adjustable ramps to cater for different vessel heights and configurations. Ramps are critical for efficient loading and unloading of vehicles.
- *Storage and Parking Areas:* Terminals provide large areas for parking and storage of vehicles before or after they are loaded onto vessels. Proper organisation of these areas is crucial to avoid congestion and ensure smooth operations.
- *Traffic Management Systems:* Effective traffic management is very important at Ro-Ro terminals. Systems are in place to direct vehicles through the terminal, ensuring that they follow designated routes and reach the correct storage or loading areas.

1.2.2 Operations and Processes at Ro-Ro Terminals

Ro-Ro terminals are dynamic environments where the flow of vehicles is continuous and often synchronised with tight shipping schedules (Dragovic et al., 2015; Maksimavičius, 2004). The success of these operations relies on meticulous planning, effective communication and a high degree of operational efficiency. The main processes involved in Ro-Ro terminal operations include the following sections.

1.2.2.1 Voyage planning

Voyage planning is the first step for Ro-Ro operations, which requires collaboration between terminal operators, shipping companies, and port authorities (Feng et al., 2020). Voyage planning involves scheduling vessel arrivals and departures, considering factors such as:

- a. *Coordination with Shipping Lines:* Terminals work closely with shipping companies to set vessel schedules, considering capacity and required turnaround times.
- b. *Berth Allocation:* Proper allocation of berths to incoming vessels based on size and cargo load is significantly important for avoiding port congestion.
- c. *Vessel Capacity:* Size and structure of Ro-Ro vessels influences the amount of cargo they can carry, and the time required for loading and unloading.
- d. *Port Congestion:* Terminal operators must account for traffic within the port and coordinate with other terminals to avoid bottlenecks.
- e. *Weather Conditions:* Adverse weather can also impact voyage planning, affecting vessel arrival times and safety considerations.

During voyage planning, terminals also ensure that suitable berths and ramps are available to accommodate arriving vessels. This stage is critical to minimise delays and maximise the efficiency of Ro-Ro operations.

1.2.2.2 Cargo Operations

Cargo operations are those activities associated with loading and unloading of vehicles to and from the roads and water transport services. Figure 1.3 is a good example of how Ro-Ro Terminal layout is. They are high-paced and in need of careful management to avoid delays and to maintain safety (Iannone et al., 2016; Wang et al., 2022). Some of the important aspects of cargo operations are as follows:



Figure 1.3 : Exemplification of cargo details at a Ro-Ro terminal.

- a. *Loading and Unloading:* Vessel entrances and exits are provided with more than one ramp on dedicated Ro-Ro terminals for efficient replenishing and discharging of automobiles. This includes driving vehicles on and off the vessel while being directed by terminal personnel to ensure safety.
- b. *Turnaround Time:* Turnaround time is the most important KPI of Ro-Ro operations so, it must be as low as possible. The goal of terminals is to load and discharge as fast as possible to maximize vessel utilization and minimize port congestion.
- c. *Traffic Management:* Terminal automation should cover traffic management to avoid congestion within the terminal. Terminal operators handle vehicles along assigned routes by means of traffic lights and other devices to facilitate a proper distribution of its flow through the terminal.
- d. *Vehicle Handling:* Terminal personal must handle vehicles with care to prevent damage during loading and unloading. The service also involves checking vehicles for previous damages, and ensuring they are driven and parked safely within the terminal.
- e. *Safety Procedures:* In Cargo operations safety is always on the first place due to prevent any accidents & damaging of cars while handling them.

Cargo operations also include managing storage and parking of vehicles in or around the terminal. The terminals need to have enough room, as the incoming goods and traffic should be formed as harmoniously as possible.

1.2.2.3 Customs and Security

Customs and security services form the most important part of terminal operations for any Ro-Ro terminal. Terminals must meet customs requirements and need to be very secure to protect the cargo and the terminal itself (Maksimavičius, 2004; Özkan et al., 2016a). Significantly Surrounding the Customs and Security Factors Include:

- a. *Customs Inspection:* All vehicles that pass-through Ro-Ro terminals must go through customs inspection. It includes verifying papers, contraband checking, and looking after the import/export rules.
- b. *Security Measures:* Terminals are secured to avoid theft, smuggling as well as prevention of unauthorized access. This involves surveillance, access controls and on-site security guards.
- c. *Compliance and Regulations:* Ro-Ro terminals must comply with a wide range of regulations such as environment, safety and customs requirements. Regulatory compliance is integral for the terminal to operate properly and not incur any penalties.

Customs and security processes need to be incorporated into the total operation to facilitate transitions between the vessel and the terminal and back again. Because of this, terminal operators need to cooperate closely with customs authorities to ensure that cars can quickly move through the inspection step.

1.2.2.4 Terminal Operations

Terminal operations are the functions which relate with the arrival of Ro-Ro ships at the terminal and the departure of such ships upon completion of operations including the use of terminal infrastructure (Zhang et al., 2023). This section covers:

- a. *Vessel Scheduling*: Terminal operations decide vessel arrivals and departures schedules to maximise throughput while reducing delays.
- b. *Resource Allocation*: Smooth running of loading ramps, space, and personnel with efficient allocation of resources conduct the terminal operations efficiently.
- c. *Coordination with Port Authorities*: Terminal operators cooperate with the port authorities to manage the vessel traffic and adhere to port regulations.
- d. *Equipment Maintenance*: This is concerned with keeping things running efficiently, such as ramps and vehicles used within the terminal.

The objective of terminal operators is to ensure high levels of productivity, orchestrating various functions to reduce downtime and increase the cargo throughput.

1.2.2.5 Yard Management

Yard management includes the process of movement, storage and handling of the vehicles within the Ro-Ro terminal (Tsoukos Georgios and Giannopoulos, 2023). Efficient cargo operations and prevention of the congestion needs a convenient yard management (Yavuz et al., 2013). Key aspects include:

- a. *Parking Space Optimization*: Advanced yard management ensures proper utilization of your parking slots so that trucks and trailers are parked in such a way that canines perform fast loading and unloading.

- b. *Traffic Flow*: Controlling the traffic flow inside of the yard is critical to avoiding bottlenecks and ensuring safety. Vehicle movements are controlled using designated routes with visible signages.
- c. *Inventory Tracking*: Yard management manages the location and status of vehicles in the terminal. This can help control; item loading, unloading, and errors.
- d. *Safety and Security*: It is priority to ensure safety of workers and vehicles in the yard. Precautions - Safety Preparedness, Signage and Secured Monitoring to avoid accidents and theft.

Positive implications relating yard management are found to be essential in efficiency aspects of Ro-Ro terminals to operationally synchronize activities concerning cargo-related operations and terminal functions. It also needs proper coordination and management to allow vehicles to pass through the terminal effortlessly.

1.2.3 Challenges in Ro-Ro Terminals

1.2.3.1 Parking Space Management

The parking space is one of the most essential challenges that Ro-Ro terminals need to address on priority. Figure 1.4 is good example to present parking space arrangement in a Ro-Ro Terminal. As the number of vehicles grows, terminals need to make the best use of their parking lots to prevent terminal congestion and keep up with their productivity.



Figure 1.4 : Example of cargo area management at a Ro-Ro terminal.

This challenge is influenced by several factors:

- a. *Capacity Constraints:* Limited number of parking spaces can cause overcrowding and impacts vehicle flow raising throughput times. With increasing number of vessel berths to terminal, demand for parking space can exceed capacity quickly.
- b. *Dynamic Traffic Patterns:* Vehicle flow in the terminal is changing, with peaks during load and unload. Adaptive solutions are necessary to handle different traffic flows.
- c. *Efficient Use of Space:* Terminals should utilise the available space efficiently, account for vehicle size and type and be future-ready to avoid bottlenecks at a later stage.
- d. *Vehicle Turnaround:* Reducing dwell time is critical for optimizing vehicle turnaround and hence, for terminal capacity management. The longer vehicles remain in the terminal, the more it affects overall throughput.

To counter these challenges, terminals are shifting focus to solutions like dynamic space allocation in parking bays through automation. These systems can mitigate the need for manual inputs and improve overall efficiency. There is also dynamic space allocation, which utilizes real-time data in order to modify parking configurations as needed in response to traffic flow.

1.2.3.2 Dwell Time Prediction

Predicting the dwell time of vehicles is challenging due to the dynamic nature of terminal operations. Inaccurate dwell time predictions can result in congestion and delays, reducing overall terminal efficiency. Machine learning models can help address this issue by improving the accuracy of dwell time predictions, but they require high-quality data and constant tuning.

1.2.3.3 Congestion and Traffic Flow

Terminals often face congestion at peak times, particularly during vessel loading and unloading. Managing vehicle traffic within the terminal while maintaining a smooth flow of operations requires advanced traffic management systems and better coordination between different operational teams.

1.2.3.4 Environmental Regulations and Compliance

Ro-Ro terminals must comply with strict environmental regulations, especially regarding emissions and waste management. Adapting to new regulatory requirements, such as carbon reduction initiatives, can impose operational challenges, particularly for terminals handling large volumes of vehicles and cargo.

1.2.3.5 Technological Integration and Data Management

The integration of various technologies such as automated systems, IoT devices, and AI-based tools presents a challenge, especially in ensuring data consistency and system

interoperability. Ro-Ro terminals handle massive amounts of data (e.g., vehicle tracking, operational metrics), and integrating these systems without disruptions can be difficult.

1.2.3.6 Labour and Workforce Management

The operational complexity of Ro-Ro terminals requires a skilled workforce, but training personnel to handle increasingly automated systems while maintaining operational efficiency presents another challenge. Additionally, labour shortages or strikes can heavily disrupt terminal operations.

1.2.3.7 Safety and Security

Ensuring the safety of both workers and vehicles within the terminal is an ongoing challenge. With a high volume of vehicles, cargo, and personnel, ensuring compliance with safety protocols and minimizing accidents is critical. Moreover, securing the terminal from external threats, including cybersecurity risks in digital systems, is also a growing concern.

1.2.3.8 Vessel Turnaround Time

Reducing vessel turnaround time is a major challenge in Ro-Ro terminals. The efficiency of loading and unloading operations, coupled with the coordination of vessel schedules, impacts overall terminal performance. Delays in vessel turnaround can lead to backlogs in parking space usage and increased dwell times.

1.2.3.9 Handling Increasing Volumes

Ro-Ro terminals are experiencing an increase in cargo volumes, driven by global trade and expanding fleets. Accommodating this growth while maintaining operational efficiency is a key challenge. Terminals may need to expand their physical infrastructure or improve operational processes to handle larger volumes without congestion or delays.

1.2.4 Opportunities in Ro-Ro Terminals

Despite the challenges, Ro-Ro terminals have several opportunities to enhance their operations and improve overall efficiency (Tsoukos Georgios and Giannopoulos, 2023).

1.2.4.1 Technology Integration

It provides enormous opportunities to streamline operations, reduce human error by implementing automation, artificial intelligence and newer technologies etc. Automated systems can help improve traffic management and increase the throughput (Abourraja et al., 2021; Iannone et al., 2016; Özkan et al., 2016a; Yavuz et al., 2013).

- a. *Machine Learning for Traffic Management:* Machine learning algorithms can be useful to analyse traffic patterns within the terminal, such as predicting peak times and suggesting optimal routes. This will help to reduce dwell time by improving traffic flow.
- b. *Automation and AI Integration:* Automation and AI-driven automated systems can streamline vehicle movement and reduce the dwell time. These manage the parking of vehicles and or loading and unloading of vehicles in a more efficient manner.
- c. *Collaborative Machine Learning Partnerships:* Ro-Ro terminals could partner with shipping companies and logistics providers to exchange data and develop machine learning models that enhance schedule optimization and decrease dwell time. This cooperation may result in improved scheduling and, as appropriate, higher productivity at the terminal.
- d. *Predictive Maintenance:* ML is used to anticipate when maintenance needs to be performed on equipment, down to the individual piece of equipment, thereby avoid unscheduled maintenance. This minimizes downtime and ensures that key infrastructure, such as ramps and berths, always remains operational, and help lower the dwell time.

1.2.4.2 Sustainability Initiatives

There are several eco-friendly practices in-built with Ro-Ro terminals, including electric vehicles for internal transport, energy-efficient lighting, etc. The implementation of these programs will lead to reducing the environmental footprint, as well as, decreasing operational expenses (Tsoukos Georgios and Giannopoulos, 2023).

1.2.4.3 Expansion and Diversification

With the increase in demand for Ro-Ro services, terminals have the option to look at new berths or expansion of existing berths. Related services such as vehicle maintenance and pre-delivery inspection can diversify the income of the distributor (Jia et al., 2022).

1.2.4.4 Collaborative Partnerships

To optimize operational efficiency, Ro-Ro terminals can collaborate with shipping companies, logistics providers, and port authorities. This can result in shared resources, lower costs and collaboration.

By addressing these challenges and leveraging these opportunities, Ro-Ro terminals can certainly lay claim to the crucial role they play in maritime logistics, as well as the growth and prosperity that provides in the rapidly changing space. There are several opportunities that Ro-Ro terminals can tap into to aid in better capacity management and operational efficiency (Tadini Marcello and Borruso, 2022). While testing machine learning as a part of their strategies, terminals can maintain low dwell time and improve the safety as well as the security. This strategy can make Ro-Ro terminal stronger and more competitive, creating favourable conditions for growth and sustainability of marine logistics.

1.3. Parking Space Optimization in a Ro-Ro Terminals

Parking space optimization in Ro-Ro terminals is significantly important for efficient operations, especially if vehicle volumes increase and space constraints become more

challenging (Abourraja et al., 2021; Chen et al., 2021a; Dragovic et al., 2015; Özkan et al., 2016a). Effective parking space optimization can help to reduced congestion, improve turnaround times, and enhance overall terminal efficiency. Below is a detailed overviews of various methods and techniques used for parking space optimization in Ro-Ro terminals.

1.3.1 Automated Parking System

New technologies are being used to make more efficient use of parking space in terminals (Jia et al., 2022). The options are endless, with systems that can be configured for different vehicle types and sizes, as well as space utilization needs. Key features include:

- a. *Automated Storage and Retrieval:* Vehicles being stored in a multi-level structure, and automated systems retrieve them as needed. This way, it's possible to save space on a large parking area.
- b. *Dynamic Space Allocation:* Automated systems can alter parking configurations on demand, responding in real-time to changes in traffic and the number of vehicles headed their way. This flexibility enables the terminal operator to quickly respond to changing circumstances.

1.3.2 Smart Traffic Management

By applying smart traffic management, the terminal is equipped with sensors and monitored through technology to read and adjust the traffic flow within the terminal to minimize congestion and maximize parking space utilization (Yavuz et al., 2013; Zhang et al., 2023). This approach includes:

- a. *Real-Time Monitoring:* Cameras and sensors that track vehicle movement inside the terminal. It is employed in traffic process operation and bottleneck identification.
- b. *Predictive Analytics:* Machine learning algorithms analyse data from multiple sources to predict patterns in traffic and identify peak times. It gives terminal

operators a clear overview of parking arrangements and helps plan the space accordingly before the arrival of trucks.

- c. *Traffic Signals and Guidance Systems*: The installation of smart traffic lights and automated guidance systems helps safely guide vehicles through the terminal, thereby ensuring swift parking and reduced dwell time.

1.3.3 Advanced Parking Layout Design

The layout of the parking area plays a significant role in optimization. Ro-Ro terminals often use advanced design techniques and models to maximize space, such as:

1.3.3.1 Compact Layouts

Parking areas with narrow lanes and compact spaces allows for greater vehicle density. This requires careful planning to ensure safety and ease of movement.

- a. *Fishbone Parking Model*: Fishbone is a very common parking layout in Ro-Ro terminal. The parking structure is a type of multilevel car park that is essentially stacked in the shape of a fishbone; with a central spine road under which each parking spot lies on either side parking at an angle. This has several advantages:
 - i. *Efficient Space Utilization*: The angled parking arrangement allows more vehicles to be parked within a given area to maximize capacity.
 - ii. *Improved Manoeuvrability*: Vehicles can enter and exit parking spaces more easily due to the angle, which reduces the risk of congestion and accidents.
 - iii. *Reduced Congestion*: Traffic will flow better if vehicles are able to slide in and out of their row without huge disruptions.

- b. *Parallel Parking Model:* Grid or parallel parking is aligned grid wise making vehicle rows or blocks. It is especially effective in a big Ro-Ro terminal with good areas. Key advantages of block parking include:
 - i. *Straightforward Layout:* The grid structure is simple and easy to navigate, with clear routes for vehicles to follow.
 - ii. *Scalability:* Block parking can be easily scaled up or down depending on vehicle volume and terminal size, providing flexibility for future expansion.
 - iii. *Minimized Dead-Ends:* Reduced dead-ends help unproductive KM, allowing better traffic flow helping in lower dwell times.
- c. *Drive-Through Model:* In the drive-through model, allows vehicles to enter and exit without reversing. This kind of model is generally adopted by busy traffic terminals where more train services and rapid turn-around are supposed to be required. Key benefits include:
 - i. *Quick Access:* Vehicles can directly enter or exit so it will take less time to park and take them out.
 - ii. *Reduced Congestion:* With no as such option of reversing or turning around, the movement of traffic in and out of spaces is smoother and thus less chance of any sort of congestion and hence less dwell time.
 - iii. *Improved Safety:* Decreases potential for accidents (common where reversing is involved such as in the terminal) which can lead to a safer terminal.
- d. *Stacked Parking:* As the name suggests, stacked parking refers stacking the vehicles on top of each other, especially on those terminals where there is a constraint on ground area. It is frequently paired with an automated system

in order to maximize the efficient storage and retrieval of vehicles.

Advantages include:

- i. *Maximized Space Utilization:* It increases terminal capacity without having to expand or add more space on the footprint
 - ii. *Automated Systems:* Stacked parking often relies on automated systems, which improve efficiency and reduce human error.
 - iii. *Reduced Ground-Level Congestion:* Storing vehicles vertically requires less ground-level parking, resulting in less congestion and better traffic flow.
- e. *Circular or Spiral Parking:* This model is used to park vehicles along a circular or spiral path, either at grade or as part of a multi-level structure. This last model is employed in bigger terminals with various levels. Key features include:
- i. *Continuous Flow:* The round nature of vehicles to travel slowly throughout the round without stopping, which helps decrease bottlenecks and results in better traffic progression.
 - ii. *Efficient Use of Vertical Space:* When it comes to using empty vertical space, the spiral structure is by far the most efficient way to build a system that stores cars in the smallest number of parking spaces.
 - iii. *Reduced Dwell Time:* Continuous flow and multiple levels mean quicker movement of vehicles through the terminal.
- f. *Double Row or Tandem Parking:* Double row or tandem parking includes the parking of vehicles in two rows, in which one row is parked in front of the other row. It is used in places where there is very restricted space, and the

vehicles come in and out in a precise manner. This model offers advantages such as:

- i. *Increased Capacity:* As parking vehicles side by side also means better use of space.
- ii. *Quick Turnaround:* This type of arrangement is useful in terminals where vehicles are required to spend little time in the terminal since it is much quicker to clear space in front than it is to clear space in the back.
- iii. *Flexible Arrangement:* Tandem parking can be penetrating to an array of terminal layouts thereby imparting the trait of flexibly in both designing and operation.

The various parking models give Ro-Ro terminals room to manoeuvre to maximize space, and efficiency. Every model is customizable to reflect the terminal's specific needs and characteristics, which in turn offers a more flexible, and efficient, solution for the parking space management. Ro-Ro terminals improve efficiency and capacity management by using more compact machines and blending multiple models together.

1.3.3.2 Modular Parking Structures

The modular design allows terminals to add parking as needed instead of constructing a large new lot. This offers little flexibility for expanding terminals in different vehicle volume growth stages.

1.3.3.3 Flow Optimization

Parking layouts are optimally designed by terminals to prevent messing up with traffic. This includes fewer sharp turns, less street clutter and better access for vehicles to get on and off the car park.

1.3.4 Parking Space Allocation Algorithms

These are algorithms that use data and technology to determine how parking spaces are deployed. The algorithms take into account various factors, exhaustive list are vehicle type, expected dwell time and the traffic patterns to find the best parking layout. Key aspects include:

- i. *Dynamic Allocation:* Parking space allocation by algorithms is adjusted dynamically according to changing real-time data which makes sure available space is always used best.
- ii. *Priority-Based Allocation:* Some algorithms allocate parking spaces based on priority for e.g. nearest spots to loading ramps for cars that are going to be loaded soon.
- iii. *Vehicle Segmentation:* Vehicles are segmented by type (e.g. car, truck, trailer) to maximize the use of space per vehicle type based on size and shape.

1.3.5 Use of Vertical Space

Ro-Ro terminals would use Vertical Space for better Parking! To do so would require them to construction of multi-level parking structures with stacker systems, which in turn allow them to store cars in a vertical format. Benefits of Vertical Space:

- i. *Increased Capacity:* Multi-level structures can more than double the terminal's parking area without increasing.
- ii. *Reduced Congestion:* By separating vehicles onto different levels, terminals can reduce congestion in key areas and improve traffic flow.

Listed above are some of the common ways through which parking space utilization is enhanced within a Ro-Ro terminal. Which method is best boils down to the individual requirements of the terminal, the number of vehicles and the resources available. Ro-Ro

terminals manage to use them effectively, it increases the efficiency of the port terminal and reduces congestion on the roads and explains the improved commercial operation of the terminal.

1.3.6 Importance of Dwell Time in Parking Space Optimization

Dwell time plays an important role for the optimization of parking space in Ro-Ro terminals. It refers to the duration that vehicles remain in the terminal before loading onto a vessel or after unloading to leaving the terminal (Chen et al., 2021a; Jia et al., 2023; Özkan et al., 2016a; Wang et al., 2021). Managing dwell time effectively is crucial for several reasons:

- i. *Capacity Management:* Reducing the time vehicles are held in the terminal allows them to come in and go as fast as possible freeing up parking space. This results in good capacity utilization, preventing congestion.
- ii. *Improved Terminal Efficiency:* Reduce dwell time to overall improve the terminal operations. Less idle time for vehicles which translates to better loading and unloading operations.
- iii. *Reduced Congestion and Delays:* Longer dwell times may result in congestion and delays while decreasing terminal efficiency and increasing operational costs. Terminals can be the intermediary to keep the traffic flowing on the terminals by reducing the duration of time spent in one and therefore bottlenecks.

1.3.7 Techniques to Reduce Dwell Time

There are several methods and strategies available to streamline operations and optimize capacity management in Ro-Ro terminals to reduce dwell time and help struggling operators by being a partner with them in the recession phase (Farzadmehr et al., 2024; Sunardi and Somakila, 2020; Yoon et al., 2023). Techniques are designed to make vehicles and equipment move faster around the terminal, shorten transit times, and improve efficiency in general.

1.3.7.1 Real-Time Monitoring

Real-time monitoring involves the use of technology, such as cameras, sensors, and tracking systems, to continuously monitor vehicle movements within the terminal. This technique offers several benefits:

- i. *Immediate Detection of Delays:* Real-Time monitoring tool helps terminal operators spotting the delays on the line and act before the situation dwells.
- ii. *Data-Driven Decision-Making:* Terminals can make informed decisions on traffic flow and space allocation by collecting and analysing real-time data.
- iii. *Improved Safety:* Real-time monitoring also contributes to improve safety by allowing operators to detect potential hazards and respond quickly, preventing accidents and further delays.

1.3.7.2 Traffic Flow Optimization

This is important for minimizing the dwell time and maximizing the parking space occupancy, so optimizing the traffic flow is key. This process is part of routing traffic flows, to improve bottlenecks and optimize free flowing traffic movements. Some key aspects behind traffic flow optimization involves:

- i. *Well-Defined Traffic Routes:* With clearly marked traffic lanes, and instructions, help direct vehicles through the terminal to prevent confusion and bottlenecks.
- ii. *Minimization of Dead-Ends:* Places with dead-ends can be reduced so that vehicles can move constantly instead of facing any congestion to reach the destined location faster.
- iii. *Use of Traffic Signals and Barriers:* Traffic signals and barriers used to control the flow of vehicles orderly and to prevent accidents or delays.

1.3.7.3 Priority-Based Parking Allocation

Parking allocation based on priority, which is a kind of parking-space management method that allocates parking spaces according to the urgency of the vehicle turnaround is called Priority-based Parking Allocation. This process can greatly reduce the time that a vehicle needs to load or unload. Some of the key factors of this technique are:

- i. *Designated Priority Zones:* Allocating specific areas for high turnaround priority vehicles to ensure ready access for quicker turnaround reducing dwell time.
- ii. *Dynamic Allocation:* Terminals can allocate parking slots dynamic basis using real-time data and predictive analytics on the type of vehicle expected time of dwell and traffic patterns.
- iii. *Integration with Shipping Schedules:* Slot is Integrated with ship schedules to prioritize parking allocation in vehicle gate so that the truck is waiting time ready for load and unload, want to avoid dual handling time to avoid the delays.

1.3.7.4 Advanced Planning and Coordination

In Ro-Ro terminals, developing better systems of advanced planning and coordination are key strategies to reduce dwell time. These require complex planning and coordination of multiple collaborating parties' Key elements include:

- i. *Coordinated Vessel Arrivals and Departures:* Planning vessel arrivals and departures with precision can help terminals to shrink dwell time by minimizing wait times and avoiding congestion
- ii. *Cross-Functional Collaboration:* Effective Communication and collaboration between terminal operators, shipping companies and port authorities is just as important in order to make operations smoother and reduce dwell time.

- iii. *Proactive Problem-Solving:* Advance planning will identify anticipated issue and create a plan for their prompt resolution with minimum dwell-time.

1.3.7.5 Use of Technology and Automation

One of the effective ways of cutting down on the dwell time in Ro-Ro terminals is by automating the operations and integrating the same with technology. This technique employs automated processes to facilitate the business functions more efficiently. Techniques include:

- i. *Automated Parking Systems:* These systems can automatically park or bring back vehicles, thereby using the space more efficiently thereby reducing the dwell time of the vehicles.
- ii. *Artificial Intelligence (AI) for Traffic Management:* AI algorithms can be used to examine traffic patterns, as well as recommend the most efficient routes to mitigate congestion and improve the traffic.
- iii. *Predictive Analytics:* Terminals can predict peak times, allocate parking spaces accordingly, and reduce dwell time using predictive analytics. There is also the predictive analysis of expected dwell time of each import cargo for each house bill of lading with Machine learning model to carry out the further operations in next plan which bring more efficiency in terminal side and reduce the overall dwell time.

Maximization of parking space is a prime concern in Ro-Ro terminal operations. Implementing with these techniques allows Ro-Ro terminals to dramatically reduce dwell time so they can make better use of their available capacities, increase efficiency and improve their overall operational effectiveness. This model allows Ro-Ro terminals to satisfy the needs of maritime logistics and manage operations in a dynamic environment without interruptions.

1.4. Dwell Time Prediction and Machine Learning Applications

Dwell time prediction is the process of forecasting cargo, vehicle or container dwell time in a port or terminal before the cargo is transferred to the next stage transportation in the supply chain (Hassan et al., 2020; Meloni et al., 2021; Moini et al., 2012a; Veenstra and de Waal, 2024; Yoon et al., 2023). In the case of Ro-Ro (Roll-On/Roll-Off) terminals this means a prediction of how long vehicles will spend in the terminal before leaving terminal by vessel or gates. Predicting dwell time with high accuracy is a key for terminal operations optimization, capacity management and congestion reduction. Figure 1.5 presents flow of Machine Learning process.

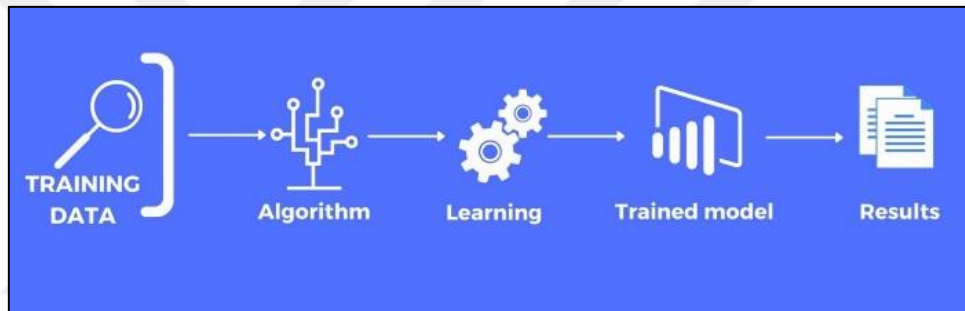


Figure 1.5 : Machine Learning Process.

Machine learning algorithms can analyse big data and decipher patterns, trends, and relationships that would be difficult for humans to spot or may not be identified at all (Smith, 2021). Using this information, these algorithms can predict future dwell times with high accuracy.

Dwell time prediction using machine learning is a game changer for port operations, providing a competitive advantage through increased efficiency, reduced delays and improved customer satisfaction.

The successful application of machine learning to dwell time prediction requires high-quality data, robust algorithms, and integration with existing terminal systems (Veenstra and de Waal, 2024). As the maritime industry increasingly embraces digitalization, machine learning applications in dwell time prediction are likely to become more sophisticated and accurate.

In summary, dwell time prediction is a critical component of port operations, and machine learning offers a powerful approach to improve its accuracy and effectiveness. By leveraging machine learning, Ro-Ro terminals and other port facilities can enhance their efficiency, reduce dwell times, and provide better service to customers. This represents a significant step forward in the ongoing efforts to optimize port operations and meet the demands of the maritime logistics industry.

1.4.1 Dwell Time Prediction

Dwell time prediction is a process which estimates how long cargo, vehicles or containers will stay in a certain region of the port or terminal before they move on to somewhere elsewhere down the logistics chain (Aminatou et al., 2018; Kourounioti et al., 2016). This involves predicting, for example, what time a vehicle will leave terminal after unloading from a vessel.

In Ro-Ro terminals, this involves predicting how much time cars, lorries or trailers can expect to wait in the terminal before being loaded onto or after being unloaded from a ship.

Dwell time prediction is a key part of port operations, helping capacity management, operational efficiency and resource allocation (Cuong et al., 2023; Kourounioti et al., 2016). The accuracy of these predictions can significantly impact a terminal's performance, as it influences everything from staffing and equipment usage to traffic flow and safety.

1.4.1.1 Importance of Dwell Time Prediction

There are five reasons why dwell time prediction is a key part of operating a port efficiently (Myriam Gaete et al., 2017a; Tong and Nachtmann, 2017):

- i. *Management:* Prediction of dwell time helps port operators understand the movement of vehicles inside the terminal. Correct predictions enable the establishment of the quantity of space needed, and capacity required for the anticipated flow so that bottlenecks will be avoided.

- ii. *Operational Efficiency:* Accurate dwell time prediction optimizes terminal operations resulting in reduced bottlenecks and faster vessel turnaround. This brings the efficiency of the port overall up.
- iii. *Resource Allocation:* If terminals know the anticipated dwell will be lengthy, they can allocate resources more efficiently, from personnel to equipment. This can make sure the resources are available at the right time and the right place and helps in avoiding overstaffing or understaffing.
- iv. *Traffic Flow Management:* Predicting exact dwell times helps terminals plan safer truck traffic routes to minimize congestion. This means that operations are more coherent and that it takes less time to process.
- v. *Customer Satisfaction:* Shorter dwell times enable terminals to meet and exceed service levels, resulting in satisfied customers. Faster turnaround times - Whether in giving a trailer back to the customer, through their logistics operation on delay is reduced.

1.4.1.2 Factors Influencing Dwell Time

Several factors can influence dwell time in Ro-Ro terminals, and understanding these factors is crucial for accurate prediction:

- i. *Vessel Schedules:* The arrival and departure times of vessels play a significant role for dwell time. Delays of vessel schedules can lead to longer dwell times for vehicles in the terminal.
- ii. *Vehicle Volume:* The vehicle volume in and out of the terminal impacts the dwell time. Increased volumes can cause congestion and higher dwell times.
- iii. *Terminal Layout:* Terminal design, such as parking, loading ramps, traffic, can influence dwell time. Poor layouts can cause bottlenecks and longer dwell times.

- iv. *Operational Practices:* How well loading and unloading processes are executed, as well as the synchronization between the various teams, impacts dwell time. Effective operational practices can reduce dwell time by streamlining processes.
- v. *External Factors:* Weather conditions, port congestion, and regulatory requirements also influence dwell time. All this means that these external factors can lead to delays and longer dwelling time of vehicles at the terminal.

1.4.1.3 Methods for Dwell Time Prediction

There are many ways to predict dwell time, and all of them have their strengths and limitations (El Mekkaoui et al., 2022; Stepec et al., 2021). These methods mostly rely on historical data, statistical analysis, and machine learning methods to predict dwell time. Here are some common methods:

- i. *Statistical Analysis:* Statistical methods, such as regression analysis, can be used to recognize trends and find out patterns of relationships between various factors and dwell time. Statistical methods are useful to provide a baseline for understanding dwell time patterns.
- ii. *Machine Learning:* Machine learning algorithms can process large volumes of data to identify complex patterns and make predictions. These models can be trained on historical data to improve their accuracy over time.
- iii. *Simulation and Modelling:* Simulation techniques create a virtual model of the terminal and simulate operations to estimate dwell time. This approach allows terminals to experiment with different scenarios and identify optimal operational strategies.
- iv. *Time Series Analysis:* Time series methods analyse data over time to detect trends, seasonality, and cycles that may influence dwell time. These methods are useful for understanding long-term patterns and predicting future dwell times.

1.4.1.4 Challenges in Dwell Time Prediction

Despite its importance, dwell time prediction presents several challenges:

- i. *Data Quality and Availability:* Accurate prediction relies on high-quality data. Incomplete or inaccurate data can result to wrong predictions, which will impact operational decisions in wrong directions.
- ii. *Complexity of Operations:* Ro-Ro terminals involve complex operations with many variables, making it difficult to create accurate prediction models.
- iii. *External Factors:* Unpredictable factors which data belong to these factors cannot be used for training the model, can affect dwell time, making prediction challenging.
- iv. *Integration with Terminal Systems:* Dwell time prediction models must be integrated with existing terminal systems to be effective. This requires coordination and technical expertise.

1.4.1.5 Future of Dwell Time Prediction

Technological advancements and data analytics will probably lead to more accurate dwell time predictions. Machine learning and AI provide a way to take the more precise prediction models to the next level and combine real-time monitoring data collected from Internet of Things (IoT) devices for continuous analysis. Dwell time prediction is more important now than ever, especially as the maritime supply chain is increasingly seeing the benefits of going digital and streamlining of port operations to become more efficient.

1.4.2 Machine Learning Applications

Machine Learning algorithms are mainly divided into four categories (Sarker, 2021):

- Supervised learning,
- Unsupervised learning,
- Semi-supervised learning,
- Reinforcement learning,

as shown in figure 1.6.

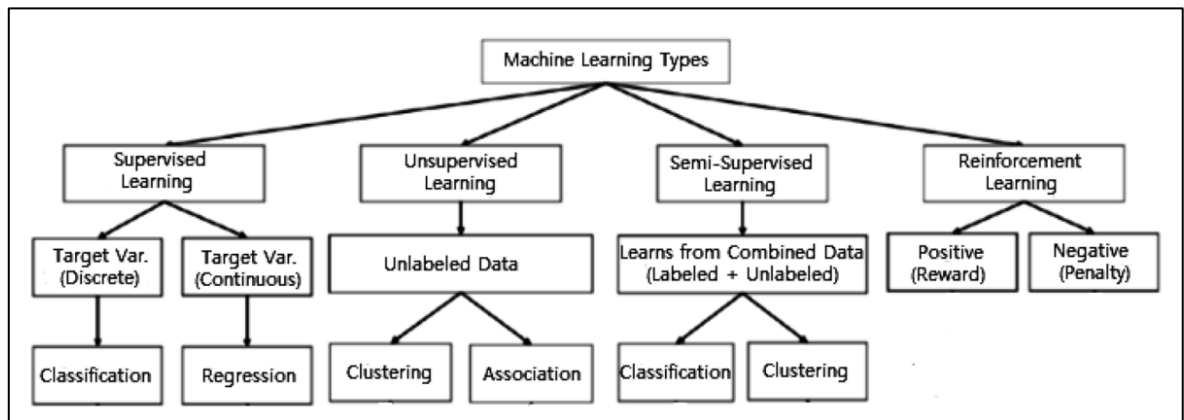


Figure 1.6 : Machine Learning algorithms.

In the following, we briefly discuss each type of learning technique with the scope of their applicability to solve dwell time prediction problem for Ro-Ro Terminals.

1.4.2.1 Supervised Learning

The supervised learning is one of most common tasks in machine learning and allows the model to estimate a function that maps an input to an output using example input-output pairs (Shetty et al., 2022). It learns the function f by using labelled training data and the labelled training examples. The task-driven approach used to complete some specific goals from some fixed set of inputs is known as Supervised learning (Suthaharan, 2016). Usually, supervised tasks identify “classification” to division the data and “regression” to fit the retrieved data.

1.4.2.2 Unsupervised Learning

Unsupervised learning is applied to unlabelled datasets, all of which are processed in accordance with the data-dependent procedure (Ghahramani, 2004; Wong Ka-Chun and Li, 2016). It is predominated to learn generative featurings, and discovering meaningful patterns and structures, result-level groupings, and it serves for exploratory. Clustering, Density estimation, feature learning, dimensionality reduction, finding association rules Anomaly detection... etc are some common unsupervised tasks in machine learning.

1.4.2.3 Semi-supervised Learning

Semi-supervised learning operates on both labelled and unlabelled data; therefore, it is a combination of the supervised and unsupervised methods described above. In this manner, it sits halfway between unsupervised and supervised learning. On the other hand, in many real-world applications labelled data may be scarce and unlabelled data are abundant, where semi-supervised learning helps. The concept of semi-supervised learning is that it would improve the prediction better than already trained model with labelled data only (van Engelen and Hoos, 2020). Machine translation, fraud detection, annotating data and spam/non-spam classification are some of the application areas where semi-supervised learning is employed.

1.4.2.4 Reinforcement

Reinforcement learning is another type of machine learning algorithm where software agents and machines assess the best course of action within a particular context or environment to increase their efficiency, i.e., an environment-driven approach (Shakya et al., 2023). It is the learning pattern explaining its action and consequences which proceeds well with reward or penalty based on the outcome at last use this feedback from environmental activists to seek more of its rewards or reduce risk (Benjamin Rolfya Jackson and Ivanov, 2023). It can be a good choice to build AI models, which will serve for automation or to streamline operations in complex systems such as robotics, autonomous driving tasks, industry 4.0 and supply chain logistics but not for simple or basic problems to solve.

Hence it can be inferred that a wide range of Machine Learning techniques may help to build accurate models on many application areas due to its learning ability in which the input data and the expected output are different but have shown their utility across different landscapes. Table 1.1 provides a broad overview of machine learning algorithms that can be leveraged to make a smarter and more competent data-driven application.

Table 1.1 : Types of machine Learning algorithms.

Learning type	Model building	Examples
Supervised	Algorithms or models learn from labeled data (task-driven approach)	Classification, regression
Unsupervised	Algorithms or models learn from unlabeled data (Data-Driven Approach)	Clustering, associations, dimensionality reduction
Semi-supervised	Models are built using combined data (labeled + unlabeled)	Classification, clustering
Reinforcement	Models are based on reward or penalty (environment-driven approach)	Classification, control

The next section is about some of the machine learning algorithms which fall under the category of supervised learning, namely classification analysis and regression analysis. Since some goal defined conditions is used in supervised learning applications, machine learning over port operation related applications fits well to supervised learning. This study will only specify a structure for supervised learning methodologies so data clustering, association rule learning methods for unsupervised learning applications and reinforcement learning applications will not be discussed in this study since these techniques are not used. In this study, we will stick to Classification and Regression methods and algorithms. As Fig. 1.7 shows a basic structure using machine learning algorithm to build a predictive model. Step 3 is from Phase 1 to Phase 2, where the model is trained on historical data (Phase-1) and predicted for a new test data.

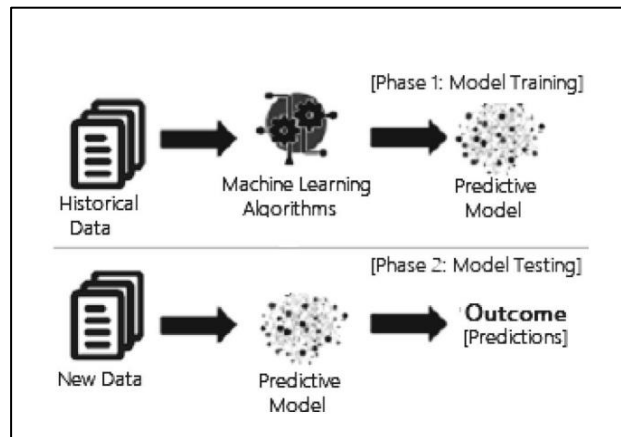


Figure 1.7 : An example of a predictive model phases.

1.4.3 Classification Analysis

Classification Algorithms are used to predict the target class by training the data. This is used for the training dataset in order to ascertain a more precise boundary condition, it might be used to define every target class (Patel and Prajapati, 2018). This will be followed by prediction of the target class after fixing the boundary conditions.

Classification entails recognition, comprehension and grouping ideas and objects into pre-existing categories or "sub-populations". Such problems can be solved with help of a pre-categorized training dataset by analysing it using the classification algorithms so that future datasets.

In machine learning classification is a supervised problem of predictive modelling where a class label is predicted for given example. Mathematically, it relates a learner (f) from the input variables (X) and output variables (Y), with target/outcome labels or categories. It is used for anticipating the class of massive data points whether they belong to one category or the other one and can be processed on structured as well as unstructured data (Gomes Mantovani et al., 2024; Rivera-Lopez et al., 2022). For instance, to classify emails as spam or not spam in an email service provider that is also a classification problem.

By leveraging classification algorithms, the model can identify recurring patterns (identical words or feelings, sequences of numbers) from future data sets.

Although the methodology is same, each algorithm varies with respect to the content of our data set and what we try to achieve.

- i. *Binary Classification:* It refers to binary classification tasks with classes that are two class labels as true and false or yes and no. In this type of binary classification tasks, one class could be for normal state and another for abnormal state. For example, a task that includes the result of a medical test could have normal state form — “cancer not detected” and abnormal state form — “cancer detected”. In a similar manner, the above email service providers example usually categorizes its messages in two different classes such as spam or not spam which are treated to binary classification.
- ii. *Multiclass classification:* This type of classification model is used when there are more than two class labels in your classification tasks. Unlike binary classification tasks, multiclass classification does not follow the principle of normal and abnormal outcomes. Instead, examples are assigned to one of a specified number of classes.
- iii. *Multi-label classification:* Multilabel classification is a key concept in machine learning where there are multiple classes or labels for a given sample. Therefore, a multi-label classification problem concerns more than one class at the same time or simultaneously-such that all possible classes in every level are represented (like checking skills), which represents related by categories not exclusive about the records-both for more hints; classic multiclass classification is a broader term to multi-level text classification where you have multiple hierarchical: lower levels influence higher levels. Example: Categorical variables like a "city name", "technology" or "latest news", etc., to which Google news can be classified multi-label classification builds on top of the traditional class label assignment where multiple class labels are assigned to each example. Unlike

normal classification stuff like this, the classes are mutually exclusive for each instance.

Many classification algorithms have been proposed in the machine learning and data science literature. In the following, we summarize the most common and popular methods widely used in various application areas.

1.4.3.1 Naïve Bayes (NB)

Naïve Bayes is based on using the Bayes' theorem with the assumption that all pairs of features in this case are independent. It works well and can be applied to many real-world situations - such as document or text classification, e-mail spam filtering.... for any binary or multi-class category (Awogbemi et al., 2021; L. Jiang and Wang, 2010). Use NB classifier, systematically which lets you effectively classify the noisy instances in a data set, and it also allows to build up a robust prediction model. The main advantage is that it can be trained on less data and fast to estimate the required parameters (in comparison to more complicated methods). But its performance may degrade for strong assumptions on independence of features. NB classifier is typically known under five common variants (Langseth and Nielsen, 2006): Gaussian, Multinomial, Complement, Bernoulli and Categorical.

Bayes theorem gives a way to calculate posterior probability $P(A|B)$ from $P(A)$, $P(B)$ and $P(B|A)$.

The formula is given in equation 1.1:

$$P(A|B) = P(B|A) * P(A) / P(B) \quad (1.1)$$

Where $P(A|B)$ is the posterior probability of A given B, $P(A)$ is the prior probability, $P(B|A)$ is the likelihood which is the probability of B given A and $P(B)$ is the prior probability of B. Example of Naïve Bayes algorithm shown in Figure 1.8.

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

Figure 1.8 : Example of Naïve Bayes Algorithm results.

1.4.3.2 Linear Discriminant Analysis (LDA)

LDA is a linear decision boundary classifier, which simply performs class conditional densities fitting and then applies Bayes rule (Xanthopoulos Petros and Pardalos, 2013). Particularly, it is recognized as a generalization of Fisher's linear discriminant, as the function tries to create features that are linear combinations of input vectors (a transformation from high-dimensional feature space to a low-dimensional one) to simplify the extracted features (i.e., reducing complexity and computational cost) (Park and Park, 2008). The standard LDA model fits a Gaussian density to each class and assumes that all classes share the same covariance matrix. LDA can also be easily related to other techniques, such as ANOVA (analysis of variance) and regression analysis, where one dependent variable can be explained in terms of the others.

1.4.3.3 Logistic regression (LR)

Logistic Regression is also a statistical model that can be used to solve classification problems in machine learning in a probabilistic way (Zedda, 2024). In general, logistic regression is based on the standard logistic function; in other words, a mathematically defined sigmoid function as shown in following equation. It may be overfit on high-dimensional databases and is efficient when the database can be linearly separated (Wahyuningsih et al., 2024). In such case we can use some of the regularization techniques to escape from overfitting. Linearity assumption BS Between dependent and independent

variables are one major disadvantage of Logistic Regression. It is used for classification and regression type of problems but often, in classification problems. Equation 1.2 show the calculation of the probability of dependent variable Y, given independent variable X.

$$P(Y = 1|X) \text{ or } P(Y = 0|X) \quad (1.2)$$

This can be used to calculate the probability of a word having a positive or negative connotation (0, 1, or on a scale between). Or it can be used to determine the object contained in a photo (tree, flower, grass, etc.), with each object given a probability between 0 and 1.

1.4.3.4 K-nearest neighbours (KNN)

K-Nearest Neighbours (KNN) is an Instance-based learning or non-generalizing learning, also called lazy learning algorithm. Instead, it simply remembers all the instances of the training data in n-dimensional space (Al-Dahidi et al., 2024). It does not bother to construct a general internal model. KNN makes use of data and uses distance measures (for example Euclidean distance function) to classify new data points. Same classification is computed by a very simple decision taken by majority of the immediate k neighbours to each point. The model is relatively robust to noisy training data, where the accuracy will vary depending on how noisy/messy your input recognition text is (Yuk Carrie Lin, 2024). The worse problem with KNN is the correct choice of how many neighbours to contemplate. KNN is being used for both classification and regression.

When k-NN is used in classification, you calculate to place data within the category of its nearest neighbour. If $k = 1$, then it would be placed in the class nearest 1st K is classified by a plurality poll of its neighbours. Figure 1.9 presents the example of classes with showing distance to neighbours.

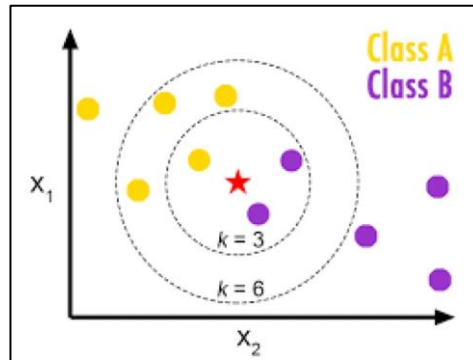


Figure 1.9 : K- nearest Neighbour plotting chart.

1.4.3.5 Support vector machine (SVM)

Another widely known technique among machine learning algorithms which can prove to be effective in handling tasks like classification, regression etc is support vector machines or SVM (Cervantes et al., 2020). A support vector machine (SVM) constructs a hyper-plane or set of hyper-planes in a high- or infinite-dimensional space. The hyper-plane, which is furthest from the nearest training data points on either side of any class will best separate the classes because if it has a large margin then it is more likely to have lower generalization error. It is well suited for high dimensional spaces and can perform different mathematical functions known as the kernel (Zhang, 2012). Kernel functions like linear, polynomial, radial basis function (RBF), sigmoid, etc., are some of the well-known kernel functions used in SVM classifier. The SVM does not work very well when the data has noise, e.g. overlapping target classes (Gold and Sollich, 2003).

The SVM then assigns a hyperplane that best separates the tags. In two dimensions this is simply a line. Anything on one side of the line is red and anything on the other side is blue. Figure 1.10 presents visualisation of SVM methodology.

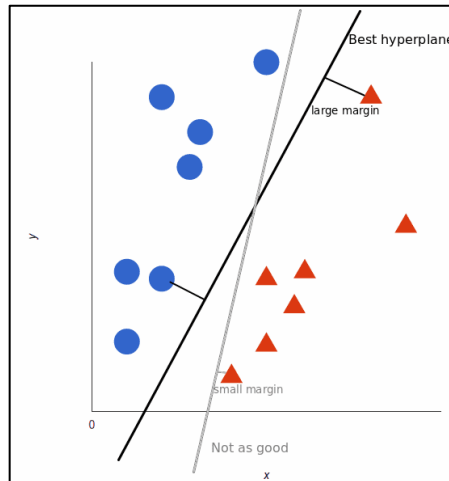


Figure 1.10 : Support Vector Machine diagram.

1.4.3.6 Decision tree (DT)

Decision Tree is one of the popular non-parametric machine learning algorithms (Gomes Mantovani et al., 2024). DT learning methods are used for classification and regressions. Sort down the tree from the root to some leaf nodes, as illustrated in Figure 1.11, to classify the instances by DT. Classifiers apply the root node of the tree to test the value of some instance attribute and move through specific branch that this attribute is leading to (Rivera-Lopez et al., 2022).

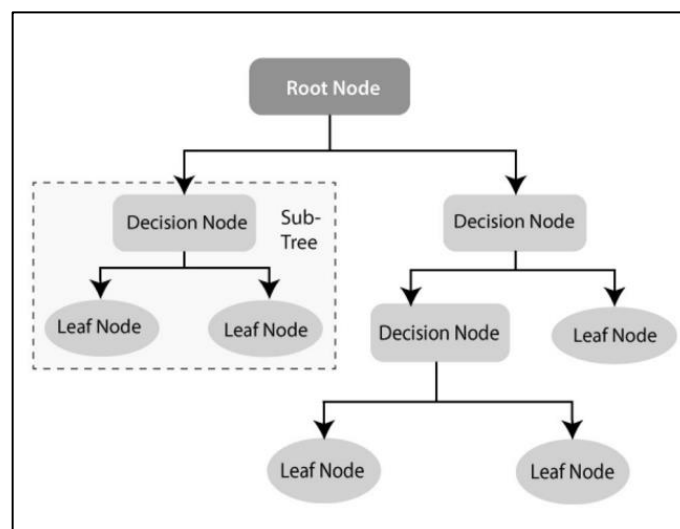


Figure 1.11 : An example of a decision tree structure.

1.4.3.7 Random forest (RF)

Random forest classifier is a well-known ensemble classification method used in the field of machine learning & data science. In this approach, several decision trees combined in parallel (Ward et al., 2021). This is called as "parallel ensembling" of decision tree classifiers depicted in Figure 1.12. Learning algorithm applied on different sub-samples of the data set and perform majority voting or averages to get the final answer. It reduces the problem of over-fitting and makes the prediction better as well. With the conclusion to this, a model of RF learning consisting of multiple decision trees is generally better than single decision tree-based models (Aqil Tariq Jianguo Yan and Mumtaz, 2023). The method is a way to construct decision trees with controlled variance, and it combines bootstrapping (as in bagging) and feature randomization. It is data-type agnostic and works equally well for both classification and regression problems, as well as for categorical and continuous values.

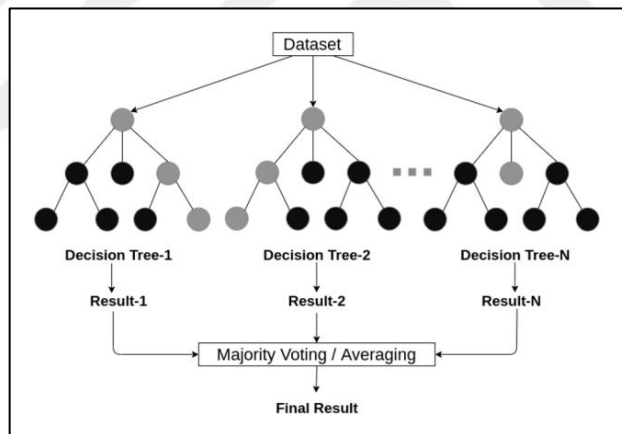


Figure 1.12 : An example of a random forest structure.

1.4.3.8 Adaptive Boosting (AdaBoost)

AdaBoost is an ensemble learning method that iteratively trains weak classifiers and improves them by focusing on their errors (Sai et al., 2023). Adaboost uses “sequential ensembling” in place of parallel ensembling like random forest. Merge many low-performing classifiers to get high-performance accuracy ratio classifier. This is why AdaBoost is known as an adaptive classifier because it effectively increases the efficiency of the classifier, however, sometimes it can cause overfit. Typically, AdaBoost is best used

to boost the performance of decision trees on binary classification problems but can also be used with other types of base estimators. The case of 2 classes is important because it only permits one estimator to come down on the negative class and all others to come down on the positive class (Wang et al., 2024).

1.4.3.9 Gradient Boosting Machines (GBM)

Gradient Boosting Machines (GBM) are powerful ensemble models that iteratively combine weak learners, typically decision trees, to minimize the loss function. GBM algorithms are highly effective for both binary and multiclass classification tasks, often yielding state-of-the-art results across various datasets. The following subsections detail three popular variants of GBM:

XGBoost: This is a scalable, distributed GBM implementation known for its high speed and accuracy (Sarkhani Benemaran, 2023). It includes regularization techniques (L1 and L2) to prevent overfitting and uses a second-order approximation of the loss function to enhance predictive performance. XGBoost is suitable for both binary and multiclass classification tasks and has built-in support for handling missing values.

CatBoost: CatBoost (Categorical Boosting) is designed to handle categorical variables efficiently without the need for extensive preprocessing. It implements ordered boosting, which reduces overfitting by considering different permutations of data during training. CatBoost is particularly effective for high-cardinality categorical features and provides robust performance for classification tasks, including both binary and multiclass scenarios.

Light GBM: LightGBM (Light Gradient Boosting Machine) is an optimized GBM variant that uses a histogram-based technique for faster training and lower memory consumption. It is well-suited for large-scale data and handles sparse data effectively. LightGBM is highly accurate for classification tasks, supporting both binary and multiclass classification, and is known for its ability to handle large datasets efficiently.

1.4.3.10 Stochastic gradient descent (SGD)

Stochastic gradient descent (often shortened to SGD) is a kind of iterative method for optimizing an objective function with suitable smoothness properties, i.e. differentiable or subdifferentiable (Ye et al., 2023). Here in the Stochastic term, it also has some significance which suggest random probability. This cuts the computation time especially in high-dimensional optimization problems, so you can iterate faster at the expense of slower convergence. Gradient means the slope of a function that computes how a variable change with differs in throughout another. From a mathematical perspective, a Gradient Descent is a convex function returning the partial derivative of a collection of its parameters (Tian et al., 2023). SGD has been widely used in large-scale and sparse applications common in text classification and natural language processing. But SGD is sensitive to feature scaling and requires several hyperparameters such as the regularization parameter and the number of iterations.

1.4.3.11 Rule-based classification

The term rule-based classification is used to refer to any classification scheme that uses IF-THEN rules for doing class prediction. Zero-R, One-R, decision trees, Ripple Down Rule learner (RIDOR), Repeated Incremental Pruning to Produce Error Reduction (RIPPER) are some classification algorithms that can generate rules (Li et al., 2024). A decision tree (Top-Down Tree Induction) is one of the most popular rule-based classification techniques among them for a lot of advantages such as ease of interpretability, suitability for handling large feature dimensionality, simplicity and processing speed, good level accuracy, generation of easily readable by human rules for understandable classification. The decision tree-based rules also give a good accuracy in prediction model as well for new test cases (Saha and Pal, 2024). As the interpretation of rules is intuitive these rule-based classifiers are also widely used for generating descriptive models that explain a system, such as by identifying the most important entities and relationships between them.

1.4.3.12 Neural Networks

Neural networks are an essential building block of machine learning, derived from the way in which our brain works with information. They are formed by layers of neurons that process input data through different mathematical operations to identify patterns and make predictions. Since neural networks can model very complicated relationships in data, they are used for image recognition, natural language processing as well as time-series.

We can use Neural Networks in machine learning to regression and classification tasks. One of the use-cases for multinomial and binary classifiers is to determine what category input should go into. Neural networks work through learning from labelled data. In binary classification, the output layer should contain only one neuron using a sigmoid activation function to predict two class probabilities. For multi-class classification, its output layer contains multiple neurons representing a class and used to predict probabilities across all classes.

The model is trained by minimizing a loss function (often the cross-entropy loss in classification) using backpropagation. This entails fine-tuning the weights of the neurons to increase prediction power. Neural networks can detect complex patterns by learning from data, they are very suited for both binary and multi-class problems in predicting.

1.4.4 Regression Analysis

The regression analysis is a set of machine learning technique used to predict the continuous output variable (y) based on the one or more independent variable (x) values (Srikiran et al., 2023). The main difference between classification and regression is that classification predicts the category of the data, whereas the continuous value is predicted by regression. Example of how classification is unlike regression models shown in Figure 1.13. There is some overlap between these two types of machine learning algorithms. Regression models are now a days widely used in many domains, like financial forecasting or prediction cost estimation trend analysis marketing time series estimation drug response modelling etc (Işıklı et al., 2020). There are a few types, some of the most common ones include linear, polynomial, lasso and ridge regression, etc... which are explained briefly below.

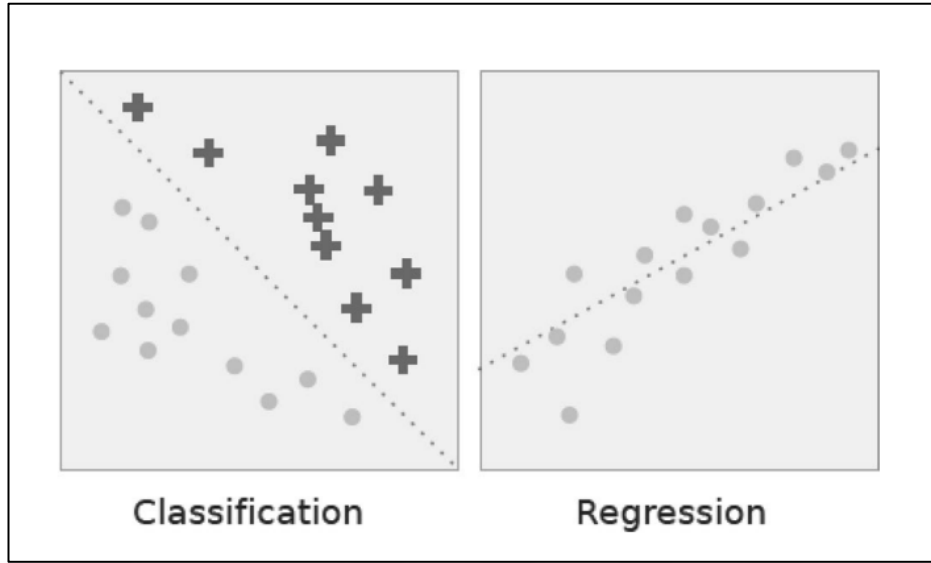


Figure 1.13 : Differences between classification and regression

1.4.4.1 Simple and multiple linear regression

A Standard ML modelling Technique, that is more in-demand and it also a regression technique (Maulud and Abdulazeez, 2020). In this method, dependent variable is continuous, independent variable(s) can be continuous or discrete and form of the regression line is linear. Linear regression attempts to draw a linear line which would go through the best-fit straight line for one or more independent variables Y, concluding in the dependent variable X (also called regression line) (James Garethand Witten, 2023). It is defined by the following Equations 1.3 and Equation 1.4 :

$$y = a + bx + e \quad (1.3)$$

$$y = a + b_1x_1 + b_2x_2 + \dots + b_nx_n + e, \quad (1.4)$$

In the equations above, ‘a’ is the intercept, ‘b’ is the slope of the line, and ‘e’ is the error term. This equation can be used to predict the value of the target variable based on the given predictor variable or variables. (Piekutowska et al., 2021). Multiple linear regression is an extension of simple linear regression that allows two or more predictor variables to model a response variable, ‘y’, as a linear function defined in Equation 1.3, whereas simple linear regression has only 1 independent variable, defined in Equation 1.4.

1.4.4.2 Polynomial regression

Polynomial regression can be defined as a type of regression analysis in which the relationship between x and y is not linear but equals to n th degree polynomial on x (Kim, Oh 2021). The polynomial regression also can be build from Linear Regression (degree 1 of Polynomial Regression) Equation 1.5 as follows:

$$y = b_0 + b_1x + b_2x^2 + b_3x^3 + \dots + b_nx^n + e. \quad (1.5)$$

In the equation above, ‘ y ’ is the predicted/target output, $b_0, b_1, \dots, b_n$ are the regression coefficient, ‘ x ’ is an independent/ input variable. Basically, we can say that if data are not distributed linearly, instead it is n th degree of polynomial then we use polynomial regression to get desired output (Qiu et al., 2020).

1.4.4.3 LASSO and ridge regression

Lasso and Ridge regression are common methods as they useful to build learning models when number of features is large, because these tend to avoid over-fitting the data by reducing model complexity (Uyanık et al., 2020). The LASSO (least absolute shrinkage and selection operator) regression model uses L1 regularization technique that uses shrinkage, which penalizes “absolute value of magnitude of coefficients” (L1 penalty). Thus, LASSO seems to be a method of setting coefficients effectively at zero. In other words, LASSO regression seeks to determine which predictors are most influential in predicting a quantitative response variable so that the subset can be utilized for prediction. On the other hand, ridge regression uses L2 regularization, which is the “squared magnitude of coefficients” (L2 penalty). Thus, ridge regression forces the weights to be small but never sets the coefficient value to zero and does a non-sparse solution. As a result, LASSO regression is useful to obtain a subset of predictors by eliminating less important features, and ridge regression is useful when a dataset has “multicollinearity” which refers to the predictors that are correlated with other predictors (Deprez and Robinson, 2024).

1.4.5 Effects of Dwell Time on Ro-Ro Terminal Operations

Dwell time, the amount of time a cargo stays at the terminal, either before loading onto a vessel or after being discharged from a vessel and before leaving the terminal. The dwell time of cargo within the terminal is the biggest determinant factor in Ro-Ro terminal capacity utilization. For import trailers detained in the terminal for a long time, their storage capacity is extremely limited, because of the lack of space available for shunting activities which lowers the overall effectivity on how these terminal functions and generating irrelevant extra costs (Chen et al., 2021). Previous research has also identified two main bottlenecks in Ro-Ro terminals which are low operational efficiency and insufficient storage resources which are greatly restricted the development of Ro-Ro transportation in different countries (Kourounioti et al., 2016). Better management and estimation of dwell time at Ro-Ro terminals lead to more optimum resource utilization, decrease the level of congestion and higher performance of the terminal as a while.

Seemingly, these data-driven machine learning techniques hold great promise in closing the gap between the accuracy of dwell time predictions at Ro-Ro terminals. By using historical data and current operational information, machine learning models can be made to predict dwell time more accurately than conventional prediction techniques. Thus, the terminal operators being informed of estimated dwell time variability can be in a position to manage resources, and plan for gate operations and optimize cargo handling processes on site. The Ro-Ro terminals will increase the overall efficiency, decrease waiting times and boost the sustainability development of the Ro-Ro transportation if dwell time prediction can be integrated as part of their operational strategies.

Therefore, the optimization of storage location assignment and truck arrival and trailer discharge management are crucial in enhancing the operation efficiency of Ro-Ro terminals. Predicting the dwell time through machine learning can further improve the proactive planning for terminals - they can better forecast truck arrivals and assign storage locations proactively, thus improving efficiency in Ro-Ro operations(Lennerfors, 2022).

Ro-Ro terminals are directly related to the capacity, volume of cargoes and resource deployment will increase production efficiency. Low Dwell Time is important to improve

capacity utilization. Machine-learning-based dwell time prediction can be employed to avoid overcapacity and shortage of terminal resources, resulting in the efficient operation of the terminal (Luo, 2024; Tsoukos Georgios and Giannopoulos, 2023). Ro-Ro terminals, which can be better prepared discharging trailers according to predicted dwell times, as well as optimized storage location assignments, thus becoming less congested and providing more efficient service to customers.

1.4.6 Importance of Dwell Time in the Context of Energy Efficiency and Sustainability

The maritime transport and port logistics industry is increasing its importance in global economic growth; however, it is also one of the main sectors that contributes to climate change (Ejder et al., 2024b, 2024a). Some of the largest ships-scrappers and tankers, for example-require lots of energy to remain in operation while at birth at ports: they usually rely on onboard diesel generators which create tonnes of air pollutants and greenhouse gases. It is of greatest importance to enhance the energy efficiency of port operations in order to reduce these environmental impacts (Styhre and Winnes, 2019).

There is energy efficiency unemployment in ports, one important aspect is the normal activity will be cargo dwell time that it is the most used parameter to represent the duration of cargo inside a port, from input at this point to being loaded or unloaded on a vessel (Aminatou et al., 2018a; Beuran et al., 2012a). The prediction of cargo permanence temporal order at ports will facilitate the optimization of energy consumption of port operators to house higher number of containers/trailers.

Prior research has examined various mechanisms to mitigate this issue. This was observed in a Port of Dar-es Salaam in Tanzania study where longer container waiting/service time, vessel clearance delays and low inbound/outbound traffic flow were identified as the main challenges constraining port efficiency (Anselem Tengecha and Zhang, 2021). (Li et al., 2018) also looked at research on container terminals and found a key problem in truck arrivals hours fluctuating peak hours that results congestion, long queue of trucks and high air pollution.

In order to solve these problems, some studies have proposed a framework for time efficiency assessment of vessel movements at ports through AIS (Automatic Identification System) data. The studies contrast the space-time trajectories of different types of ship to argue how time efficiency can be measured in different port zones and ports.

Last but not the least if port operators can predict dwell time for different types of cargo, like as we have shown in this article, it will be very beneficial in planning and improving energy efficiency and sustainability of the port. Port operators can optimize their resource use, cut emissions and help drive the shipping industry towards a greener future using advanced analytics and innovative technologies.

By using ML and AI techniques, port authorities can build predictive models that are more accurate to determine the amount of time to predict cargo dwell time. The data can then be used to optimise port operations, minimise use of energy and cut down on the ecological footprint.

1.5. Analytical Hierarchy Process (AHP)

The Analytical Hierarchy Process (AHP) is a structured decision-making tool that helps in making complex decisions by breaking them down into simpler components. It was developed by Thomas L. Saaty in the 1970s while he was at the Wharton School of Business. AHP uses a multi-level hierarchical structure of objectives, criteria, sub-criteria, and alternatives, allowing decision-makers to assign weights to various factors and rank the alternatives based on the criteria. The process involves pairwise comparisons of the elements in each level of the hierarchy, where the relative importance of one element over another is determined through a matrix.

AHP has been widely used in numerous fields such as Business and Economics for evaluating investment opportunities, supplier selection, and project prioritization, Engineering to select technologies, materials, and system designs, Environmental Decision-Making for Applied in sustainability assessments, resource management, and environmental impact analysis, Healthcare for diagnosis, treatment selection, and prioritizing healthcare

projects and Transportation and Logistics for selecting optimal routes, prioritizing infrastructure projects, and in traffic management systems.

The strength of AHP lies in its ability to handle both qualitative and quantitative criteria, making it ideal for decision-making in complex, multi-criteria problems such as selecting machine learning models for specific tasks like dwell time prediction.

In this thesis, AHP will be used to select the most appropriate machine learning algorithm for predicting dwell time in Ro-Ro terminal operations, which is a classification and prediction problem. The goal is to optimize parking space utilization by accurately predicting the time that vehicles will occupy terminal spaces. Machine learning models such as Random Forest, XGBoost, LightGBM, CatBoost, SVM, Logistic Regression, and K-Nearest Neighbours (KNN) will be evaluated based on specific criteria to determine which algorithm is most suitable for this task.

2. LITERATURE REVIEW

Literature related to Ro-Ro terminals is scarce. Existing studies focused on capacity calculation and determination of some quality indicators, capacity calculation of the terminal gateways, customs and border control facilities, calculation of occupancy rates of cargoes at the terminal facilities.

Due to lack of literature about Ro-Ro Terminals, and this research is about application of machine learning techniques to Ro-Ro operation, also includes parking space optimization which is called storage space assignment problem in literature, and final aim of study is to achieve dwell time prediction for import cargo arriving to Trieste Ro-Ro Terminal, literature review focused also on these main subjects including below:

- Ro-Ro Terminal Operations
- Application of Machine Learning to Port Operations
- Storage Location Assignment problem
- Dwell time prediction for Ro-Ro and Container Terminals
- Dwell time driven storage locations management at ports
- Port storage area allocation models
- Container Terminal storage space management
- Capacity Management in Ro-Ro Terminals
- Port Congestion and Solutions
- Cargo prioritization

Regarding dwell time and dwell time predictions in Ro-Ro terminals, no available study found in the literature, there are only four study in literature which are about dwell time of containers where only (Kourounioti et al., 2016c) and (Gaete et al., 2017b) studied dwell time-based terminal allocation model for Container Terminals. (Sidik et al., 2020b) aimed to reduce dwell time by process optimization and (Moini et al., 2012b) identified determinant factors of dwell time by classification algorithms, but any of them did not focus on Ro-Ro Terminals.

Recently, (Martín-Lorenzo et al., 2023) presented a decision-support system for managing truck stays at a container port Santander, using machine learning and simulation models to predict truck dwell times and reduce congestion in the Port of Santander. Due to nature of Container Ports, thousands of trucks are entering from gate to pick up or leave containers and high amount of truck traffic creates congestion in port area. They tried to optimize the arrival and transit times of trucks at ports, predict congestion, and manage the planning of port resources more effectively. Even this study is about container ports and truck traffic coming from land side, after study of (Gaete et al., 2017b) it is the most relevant literature about dwell time prediction. They used Decision Tree, XGBoost and Random Forest models to predict the dwell time of trucks which is called transit time in Port in the study, and applied data engineering to data collected as data transformation and feature engineering methods to make data available to use in machine learning models. Due to this study was a conference paper, auditors did not share any data belong to models' accuracy about their predictions, just mentioned how output used in daily port operations. Still this study is one of the most important and relevant one due to outcomes of machine learning techniques being tried to use in daily port operation and initial outputs shared in a conference paper. This study highlights how can new technologies improve port operations with using power of data and predictions.

As mentioned above, the most related and important study carried by (Gaete et al., 2017b) with the subject of “A Dwell Time-based Container Positioning Decision Support System at a Port Terminal”. Defining a container storage space allocation policy problem for import containers is addressed by considering the case of a container terminal that faces a high level of uncertainty in the dispatching process of import containers. The aim of the study was reducing rehandles of import containers by segregation based on predicted dwell times. Rehandles described as, if some containers may be blocking the target container and needs to be removed to be able to reach the target container. These non-value-added movements are refereed as “rehandles” or “reshuffles” of containers. Rehandles represent a high cost with no value for the container terminal and increase the truck turnaround times of the external trucks at the container terminal, generating congestion and affecting service levels of to the users of the container terminal. To assign a storage space for the import containers in the yard, they suggested a dwell time segregated storage policy. In this case, segregations of import containers are defined based on dwell time intervals, and containers of the same

segregation are assigned to close locations. The aim is to reduce potential container rehandles. With this approach, containers with the same interval of dwell time located at close positions in the yard, which is expected to result in less rehandles. In order to estimate dwell times of import containers, classification algorithms are used. They tried K-Nearest Neighbours, Naïve Bayes, OneR , ZeroR, JRip and Decision Table models to predict the dwell time. For classification, authors divided the continuous dwell time into different classes or intervals. These intervals represent different ranges of dwell time (e.g., 1-7 days, 8-14 days, etc.). The motivation for this was that the exact dwell time is not required; rather, containers with similar dwell time estimates can be grouped together into categories. This helps in optimizing container placement in the yard to reduce rehandles. When we compare Container Ports with Ro-Ro, dwell times at container ports are significantly high, but stacking of container allows container terminals to use storage space efficiently even high number of containers and long dwell times. However, stacking is not possible for Ro-Ro Terminals and its possible to park only one cargo in a single parking slot. And due to parking area is also limited, managing dwell times at a Ro-Ro Terminals is even more crucial. As a result, they reached about low accuracy as max %7 of predictions was correctly classified. Even this low accuracy, based on their calculations, measured %17 reductions in reshuffles. But due to this measured as a yearly performance, there could be other operational improvements effecting on the reductions of reshuffles. Still, study of (Gaete et al., 2017b) is the most relevant and valuable one due to it was one the first studies about dwell time prediction and the first one which used machine learning driven dwell time predictions to optimize storage location assignment decisions.

While Gaete and others studied only container terminal, we will study Ro-Ro terminal where trailers and containers handled together but focus of this study will be only dwell times of trailers. The final goal of their study was to reduce rehandles, where we aim to reduce dwell time directly to use terminal efficiently and reduce congestion. Container Terminals and Ro-Ro Terminals dynamics are completely different, and we aim to contribute literature with our study where literature is so scarce for Ro-Ro Terminals, and this will be first study in literature which utilizes machine learning techniques for dwell time prediction of import trailers in a Ro-Ro Terminal.

Other valuable related study carried out by (Kourounioti et al., 2016c) about “Development of models predicting dwell time of import containers in port container terminals – an Artificial Neural Networks application”. The goal of study is to support terminal operators make more informed decisions about stacking policies and resource allocation by improving the prediction of when containers will be picked up, which is almost same approach of this research, prediction how long import container expected to stay in Ro-Ro Terminal and supporting port operators to use this info to make more informed decisions for better parking space usage to improve overall efficiency of port and reduce congestion. They proposed methodological approach to analyse the various factors affecting the Dwell Time (DT) of containers in container terminals. The suggested methodology requires the collection of aggregate data and the application of Artificial Neural Networks (ANN) to identify the determinants of Dwell Time (DT). But data size for very limited with 13.733 import containers and reached %65 accuracy, however, due to size of data was too small, results provide an insight for further studies. Therefore, in this study, we added ANN to AHP progress, but all neural network approaches got low score at AHP results. However, some of them added to analysis to ensure validation of models. According to the results, ANN showed that the most important factors affecting significantly the model’s accuracy are the following: containers size and type, the day and month of the container’s discharge, the vessel’s port of origin and the commodities transported.

Another related and valuable study carried out by (Moini et al., 2012c) with a subject of “Estimating the determinant factors of container dwell times at seaports”. They claimed reducing the amount of time a container spends at the terminal, container dwell time (CDT), may prove to be one of the least costly solutions. For this strategy to be successful, it is essential that terminal operators be able to define factors impacting the CDT and estimate how long a container remains in the yard. They analysed data belong to 11.000 containers which is also small size for a machine learning project. They compared the performance of three algorithms to estimate CDT: Naive Bayes, decision tree and a NB-decision tree hybrid. Then they suggested to focus on determinant factors that mostly effects dwell time and improve process to reduce dwell time. They reached % 83 accuracy with decision tree, which we also used even more improved decision tree models in this study as Random Forest and XGboost. Also, Moini and others did not study dwell time-based terminal allocation.

Last study directly related with dwell time carried out by (Sidik et al., 2020b) with a subject of “Modelling and Optimization Containers Dwell-Time in Tanjung Perak Port Indonesia” which addressed the critical issue of long container dwell times at Indonesian ports, particularly the Tanjung Perak Port. The study intends to simulate and optimize the container handling process in order to bring down average dwell time from 6.2 days towards an international benchmark of 3 days. To address these challenges, this paper puts forward an approach using queue modelling and optimization methods to manage the container dwell time by employing heuristic algorithms. Therefore, they haven’t focused on application of machine learning techniques but used heuristic algorithm and simulation together to optimize port processes to reduce average dwell time. Also, their study assumes that shippers always ready for pick up but internal processes in the terminal causes long dwell time. They did not follow any method to predict the dwell time like previous studies but still valuable and relevant with our research due to highlighting importance of dwell time for improving operation efficiency of ports.

(Jacomino et al., 2021) studied "Dwell Time Estimation of Import Containers as an Ordinal Regression Problem" which they argue that using ordinal regression for estimating dwell time outperforms traditional supervised classification methods, which have been commonly employed in this field. They used data belong to 4790 containers from 2014 to 2017 which is significantly low number, especially for the 2 machine learning models they compared with ordinal regression approach. Like other container dwell time focused studies, also this study specifically focuses on reducing reshuffles (unproductive crane movements) in container stacking by improving dwell time predictions. They also tried Lazy-IBK and Lazy-Kstar models with WEKA software which are old algorithms respectively and claimed that ordinal regression performed better than these models.

(Aminatou et al., 2018b) studied "Evaluating the Impact of Long Cargo Dwell Time on Port Performance: An Evaluation Model of Douala International Terminal in Cameroon", which addresses the significant issue of long container dwell times at the Port of Douala, Cameroon, and its negative impact on port operational performance and economic development. The paper discusses the determinants of long dwell times and provides a statistical analysis to understand the underlying factors that contribute to the delays, with the goal of offering potential solutions. Different than other studies trying to predict the dwell

time of containers, they used a logistic regression analysis to identify the significant factors influencing long dwell times. They categorized dwell times into three levels: less than 30 days, between 31 and 90 days, and over 90 days and investigated relation of these dwell times with determinants defined as Fiscal regime, Container type, density of value and port of origin.

(Lee et al., 2024) recently studied Key Factors Influencing Import Container Dwell Time Using Explainable Artificial Intelligence. They used data belong to 160.000 containers from the HJNC yard which is a good data size for machine learning projects. They explored 27 factors could affect dwell time of containers, then reduced to 12 after first analysis. They used Random Forest and XGBoost as machine learning models and reached %80 accuracy with XGBoost in terms of defining most relevant factors impacting container dwell time. They concluded that Temperature, Weight of Container, Voyage Number, Block location, Shipping company and Month of discharge are the factors mostly impacting dwell time. Inspiring from this study and other, we also added and categorized cargo weight into dataset.

There are 3 other studies not directly with dwell time but focused on slot allocations based on time-sensitive cargoes. (Yao et al., 2021) studied “Container Slot Allocation for Time-Sensitive Cargo in Maritime Transportation: A One-Phase Model with consideration of Port Congestion”. Study was about container market and aims to prioritize urgent cargo with higher price and guarantee needed slot on board of vessel to maximize revenue. Queuing theory assumed for port operations, but not studied. One Phase model suggested for prioritizing time sensitive cargo. Vessels schedule also considered in the model to achieve higher revenue through the model. Qi Yao and others did not focus on terminal allocation but targeted to guarantee slot on board of vessel for urgent cargo.

(Wang et al., 2013) studied “Containership scheduling with transit-time-sensitive container shipment demand”. Study focused on scheduling of container ships on the liner route to reduce transit time required for cargoes to arrive destination. They used mixed integer non-linear non-convex programming method to evaluate better vessel schedule to reduce deadlines. The study was not considering dwell time and terminal allocation problems.

Then (Wang et al., 2016) developed new approach by “Liner container assignment model with transit-time-sensitive container shipment demand and its applications”. This time they focused on slot on board of vessels instead of ship schedules and routes. Study was about guaranteeing slot on board of vessel for time-sensitive cargo by using two link based linear programming to formulate price and ship slots arrangement decisions. Same as Qi Yao, Wang and others also did not consider dwell time and terminal storage space allocation.

(Chen et al., 2021c) studied storage location assignment problem of automotive Ro-Ro Terminal. They suggested Assignment Rule based on Car Group (ARCG) instead of using Assignment Rule based on Car Type (ARCT) which is mostly used method in automotive terminals. Hierarchical two-stage exchange strategy (HTSES) proposed to avoid deadlocks in terminal. Chen and others explained why ARCG type assignment rule would avoid parking space problems. The study very well prepared, but the model cannot be applied to standard Ro-Ro terminals where trailers parked in the terminal with fishbone parking systematic. But cars parked with block systematic in automotive terminals. One trailer should not avoid movement of other trailer in standard Ro-Ro terminals, however there isn't same requirement for automotive terminals.

Regarding capacity calculation of Ro-Ro ports, (Morales-Fusco et al., 2010) carried out valuable study and with (Özkan et al., 2016b) there are only 2 studies in literature regarding capacity analysis and calculation of Ro-Ro ports to the best of our knowledge. Fusco's study investigated whether already in operation or during its design process and relate it to some quality standards by means of quality indicators, which should lead to the definition of Levels of Service similar to those already used in roads and airports. They defined capacity as maximum number of Intermodal Transport Units (ITUs) a RoRo terminal can handle, considering optimal operation of all its equipment and a convenient demand pattern to reach this goal. Also defined quality indicators specifically for pure Ro-Ro port which explained as needed parameters to calculate capacity of the terminal. This study is the only one in literature which focused on capacity calculation and quality indicators calculation methodology for pure Ro-Ro port. However, due to scope of the study, they did not relate capacity of terminal with dwell time or cargo at port and not suggested any allocation model for better capacity utilization.

(Özkan et al., 2016b) highlighted lack of study for Ro-Ro terminals and developed a model for making capacity analysis in Ro-Ro terminals by using simulation modelling method. Investigated effect of three variables to terminal capacity: 'number of trucks arriving to terminals', 'distance between terminals' and 'Ro-Ro ship capacity'. They concluded that the variable which affect terminal capacity mostly is 'number of trucks arriving to terminals'.

(Xie and Huynh, 2010) used two kernel-based machine learning methods for the prediction of truck trips generated at Houston Container terminals. The aim of the study was to design traffic control strategies based on predictions to mitigate the adverse impact of truck traffic on road networks and port authorities can better size their facilities. They used Support vector machines and Gaussian process algorithms for the terminal operation data for predictions and confirmed results with multilayer feedforward neural network (MLFNN) model. The importance of study is, like studies which aimed to predict dwell times, it proofs that machine learning algorithms and neural network models can create significant efficiency for port operations, not only gate activities or dwell time predictions, but these algorithms can also play important role for all level of processes in ports even for safety precautions, vessel schedules, predicting stowaway cases etc.

(Amrou et al., 2018) studied possible advantaged of process mining towards Big data analytics for the Ro-Ro Terminals. They used simulation for define the system should be able to relieve congestion and confront the future demand volume flows estimated at RORO port terminal to help to identify the bottlenecks and deadlocks conditions as well as construct a clear picture of latency in the system. They proposed Process Mining techniques to analyse and picture the reality of what is hidden in RORO event for each model by gathering process statistics.

(Muravev et al., 2016) compared model development in discrete event simulation on Ro-Ro terminal example comparison of two discrete event simulation modelling software programs, which Arena and AnyLogic are used. Operations of a Ro-Ro terminal is modelled with Arena and AnyLogic independently and results are compared. There was no significant difference with the output of different simulations. This study observed if there are any changes observed on the result when different simulation tools used in the same set up.

(Keceli et al., 2013) developed a simulation model for Ro-Ro terminal operations to provide decision making and resource allocation. The model is tested by using the data from a Ro-Ro terminal in Türkiye. As a result, it was discovered that the maximum utilization rate of one of the waiting areas exceeded its capacity, which shows simulation can provide better understanding of actual operations in a Ro-Ro terminal to support improvement decisions.

Aksoy (2011) studied and modelled an existing Ro-Ro terminal under four processes as import process/export process of trucks and import/export process of trailers. Aksoy calculated occupancy rates of cargoes for predefined four processes at the terminal gate, gamma-ray station, import/export truck weighbridge, fuel station, waiting areas and on the ramp. In the study, "Arena 11.0" simulation program was used.

Olivares et al. (2018) studied stacking alternatives in the container terminal with mathematical programming and heuristic algorithm. They concluded that diagonal stacking gives better results compared to horizontal stacking in terms of reduction of rehandling or reshuffles.

There are plenty of studies in literature about container terminals and most of them are related with storage location assignment problem of containers in the yard area.

(Said and El-Horbaty, 2016) studied storage location assignment problem of Damietta port with genetic algorithm and performed discrete event simulation to optimize suggested model.

(Fancello et al., 2019) proposed an optimized and integrated network for the maritime Ro-Ro freight services currently operating in the Italian Tyrrhenian Sea to improve the overall supply of shipping services. Their methodological approach was based on the integration of timetabled frequencies of existing freight shipping services using an original mathematical model that assigns demand flows to the network as well and aims at minimizing also multiple objectives of a weighted sum of travel times and tariffs. The quantitative findings, originated from rescheduling and coordination in liner shipping, indicate that a "Tyrrhenian Network System" envisaged is collectively more attractive than single services.

(Puisa, 2018) studied optimal utilization of cargo space of Ro-Ro vessel which he described as inherently problematic. He suggested three concrete improvements relative to the state of the art. The improvements lead to a finer approach to ship stability, fire safety, and cargo handling efficiency when optimizing cargo stowage on Ro-Ro decks. He described stowage problem as a mixed-integer linear programming (MILP) problem and tried to solve it to optimality. This study was being outlined of the mathematical formulation, numerical example and practical application aspects.

(Beuran et al., 2012b) analysed long dwell times causes in ports in Sub-Saharan Africa from a shipper's perspective. The findings point to the crucial importance of private sector practices and incentives. They argued that private operators which many defined themselves as champions of reforms in this sector, could be the ones needed to be blamed for most of these failures. They claimed Sub-Saharan Africa importers' and freight forwarders' professionalism, cash constraints and operators' strategies are some of the factors that have a major impact on cargo dwell time. Low competency, cash constraints and low storage tariffs explain why most importers have little incentive to reduce cargo dwell time since in most cases, this would increase their input costs.

(Maksimavičius, 2004) conducted a theoretical study aimed to reduce the cargo processing time in Ro-Ro terminals providing required number of terminal gates, customs control units and parking areas.

In order to ensure research covers recent literature, IAMU 2024 (International Association of Maritime Universities) proceedings, MT'24 (Maritime Transportation Conference) Proceedings, IMLA 2024 (International Maritime Lecturers Association) proceedings and GMC 2024 (Global Maritime Conference) proceedings checked for potential studies about dwell time predictions at Ro-Ro Terminals using machine learning techniques. But couldn't find any relevant study so far.

As a result of literature review, to the best of my knowledge, this is the first study focusing on dwell time prediction using machine learning techniques for Ro-Ro Terminals and this study will fill significant gap in literature, also hopefully will open pathway for future studies using machine learning techniques for Ro-Ro Terminals.

3. MATERIALS AND METHODS

The purpose of this study is to evaluate the effectiveness of machine learning (ML) techniques in predicting dwell time at Ro-Ro terminals. This study intends to analyse how well various ML algorithms can predict dwell times based on historical cargo data and derive meaningful insights to optimize port operations. By leveraging these advanced technologies, the study aims to enhance the accuracy, efficiency, and effectiveness of port operations, leading to reduced congestion, improved resource allocation, and enhanced customer satisfaction.

To achieve these goals, the Multi-Model Synergistic Framework for Machine Learning Optimisation (MMSF) has been implemented. This framework helps to select, test and combine different machine learning models to achieve optimal accuracy and reliability in dwell time prediction. MMSF not only identifies the most appropriate models but also ensures their effective integration to maximise the prediction performance in different scenarios.

3.1. Introducing the Dataset

The dataset used in this study is derived from detailed records of cargo movements at a Ro-Ro terminal. It encompasses various attributes related to the handling and logistics of cargo, collected using the terminal's operating system, booking system, vessel voyage reporting system and business intelligence tools used by DFDS company which operates both the Ro-Ro terminal and the vessels serving it. This dataset is essential for building and evaluating machine learning models to predict dwell times, which are critical for optimizing port operations.

The dataset includes the following key features presented in Table 3.1.

Table 3.1 : Data set key features

Feature	Description
Release Number	A unique identifier assigned to each cargo shipment
Unit Number	The identification number of the cargo unit
Unit Type Name	The specific type of cargo unit
Date of Departure (Actual) UTC	The actual departure date and time of the cargo unit from the loading port
Year of Date of Departure (Actual) UTC	The year when the cargo unit departed
Month of Date of Departure (Actual) UTC	The month when the cargo unit departed
ISO Week of Date of Departure (Actual) UTC	The ISO week number when the cargo unit departed
ISO Weekday of Date of Departure (Actual) UTC	The ISO weekday number when the cargo unit departed
Vessel Name	The name of the vessel transporting the cargo
Voyage Number	The specific voyage number for the vessel
Departure Port Name	The name of the port where the cargo was loaded
Destination Port Name	The name of the port where the cargo is to be unloaded
Agent Code	A unique code assigned to the customer or agent responsible for the cargo
Agent Country Name	The country of the customer or agent
Cargo Tare Weight	The weight of the cargo unit
Departure Country Name	The country from which the cargo is departing
Load Location	The specific location within the departure country where the cargo was loaded
Delivery Location	The final destination code of the cargo within the destination country
Avg. Avg Weekly Dwell Time	The average weekly dwell time at the destination port for that week
Dwell time - Hours	The duration in hours that the cargo waited at the destination port before leaving the port facility

The target variable, 'Dwell time - Hours,' indicates the time cargo waits at the terminal, which was categorized into predefined bins.

The dataset utilized in this study consists of historical operational data from the Trieste SST Terminal, covering the years 2022, 2023, and up to mid-May 2024. This comprehensive dataset includes a total of 143,393 rows, representing the detailed records of 143,393 trailers. These records provide valuable insights into the terminal's operations, enabling a thorough analysis and prediction of cargo dwell times.

This dataset is vital for understanding the various factors that influence cargo dwell times and for building predictive models to forecast these times accurately. By analysing and modelling this data, the study aims to improve the efficiency and effectiveness of port operations, ultimately leading to reduced congestion and enhanced service quality at Ro-Ro terminals.

3.2. Introducing the Method

The framework used in this study is named as “Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF)” model. It is basically a combination of machine learning models through an organized preprocess, feature engineering, and validation over different datasets. By applying Analytical Hierarchy Process (AHP) for model selection and supplementing it with additional models, the framework aims to identify the best-performing combinations of models tailored for specific categories. This ensures optimal performance across diverse classification tasks, balancing accuracy, efficiency, and generalization.

Using MMSF approach balances the differences and benefits of multiple models and ensures robust predictions tailored for different categories or subgroups of data.

Model consists below main steps show in Figure 3.1 which details presented in section 3.4 “Data Collection Process”.

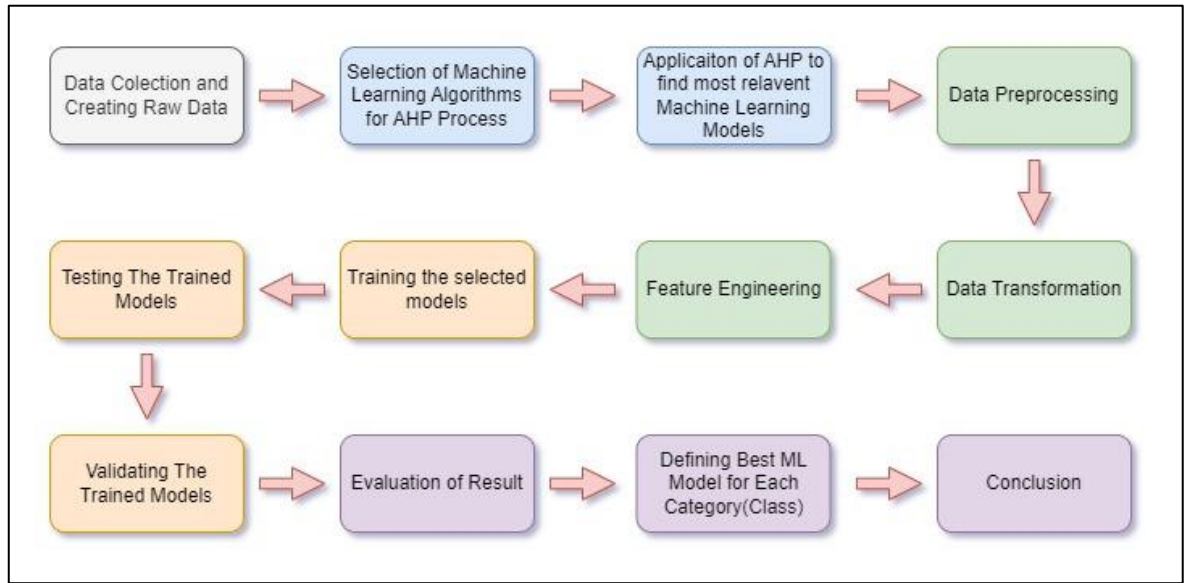


Figure 3.1 : Steps of Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF) approach.

Step 1: Data Collection and Creating Raw Data

Real-world data collected belong to Trieste Port from related systems and raw data created with size of approximately 150K records.

Step 2: AHP-Driven Model Selection

The AHP (analytic hierarchy process) is used to analyse the machine-learning models, it assists in identifying which are more appropriate for our study. It also gives a reason behind the choice of additional models besides AHP for enhanced analysis.

Step 3: Data Preprocessing, Transformation and Feature Engineering

The collected raw data undergoes preprocessing to clean and format it for analysis. This step also includes data transformation and feature engineering, creating suitable features for model training and improving the predictive accuracy of the models.

Step 4: Model Testing and Validation

After selecting models through AHP, the framework validates their performance across test and validation datasets. Highlight that MMSF integrates additional models (e.g., CatBoost, LightGBM, Logistic Regression) to cross-validate the initial results from AHP.

Step 5: Evaluation of Results and Synergistic Model Combination

Evaluation of the best-performing models for each specific category or task. By doing so, it leverages the strengths of different models, potentially combining their outputs to achieve optimal performance. By using the outputs of multiple models, making use of their strengths to return a good performance in all categories. This type of synergistic model creates a more accurate projection for dwell time, which in turn allows to make better decisions.

3.3. Selecting the Machine Learning Models

The primary objective of this study is to predict the dwell time of trailers at the Trieste SST Terminal using machine learning techniques. Given that the dwell time prediction was framed as a classification problem, various classification algorithms were selected to compare their performance and understand their strengths and weaknesses in this context. Additionally, some algorithms traditionally not suited for classification tasks were included for comparative purposes. The selected algorithms and the rationale behind their selection are as follows.

3.3.1 Selection of Algorithms for AHP Application

In the introduction section, 11 different classification algorithms and 3 regression algorithms were outlined, providing a comprehensive overview of possible approaches for dwell time prediction. However, for the AHP process, a selection of 10 classification algorithms was made, based on their relevance to the specific objectives of this study. These models were chosen to compare their performance and applicability to the classification task of predicting dwell time at the Trieste SST Terminal.

3.3.1.1 Rationale for Selecting the 10 Algorithms for AHP

a) Applicability to Classification Problems:

Since the dwell time prediction problem was framed as a classification problem, the primary focus was on algorithms capable of performing classification tasks. Algorithms that are more suited to regression problems were excluded from the AHP process, as the objective is to predict categorical dwell times (e.g., short, medium, long) rather than continuous values.

b) Performance Across AHP Criteria:

The ten algorithms selected were chosen based on their strong performance across the four AHP criteria: Data Handling, Efficiency, Generalization, and Complexity. These criteria reflect the operational demands of the task, ensuring that the models chosen can handle complex and noisy datasets, generalize well to unseen data, and be efficiently deployed in a practical terminal setting.

c) Inclusion of Decision Tree, Gradient Boosting Machines (GBM), and Neural Network:

In addition to the initial set of models (Random Forest, XGBoost, LightGBM, CatBoost, SVM, Logistic Regression, and KNN), Decision Tree, GBM (Gradient Boosting Machine), and Neural Network were included in the AHP process. These models offer distinct advantages in classification tasks:

- *Decision Tree:* While simpler than ensemble methods, it provides high interpretability, which is valuable for understanding model predictions in real-time terminal operations.
- *GBM:* Known for its high predictive accuracy, GBM provides a robust performance through boosting techniques, making it well-suited for classification tasks with complex, imbalanced data.
- *Neural Network:* Although Neural Networks tend to be more complex, they are capable of capturing deep, non-linear relationships in the data, potentially leading to better performance in certain classification tasks.

d) Balance Between Interpretability and Predictive Power:

The selection includes both simpler, interpretable models (such as Logistic Regression and Decision Tree) as well as more complex, high-performing models (such as XGBoost, LightGBM, and Neural Network). This range allows for a thorough comparison of models that provide transparent decision rules versus those that offer higher predictive accuracy at the cost of complexity.

e) Exclusion of Models Not Relevant to the Problem:

Some algorithms that were considered initially but were either:

- *Not suited for classification:* Models like Linear Regression and other regression-based models were excluded since the dwell time problem is defined as a classification task.
- *Niche or too specialized:* Certain algorithms that are highly specialized or tailored to specific types of problems were not included as they would not generalize well in the context of this study.

The ten algorithms selected for the AHP process are:

- Random Forest
- XGBoost
- LightGBM
- CatBoost
- Support Vector Machines (SVM)
- Logistic Regression
- K-Nearest Neighbours (KNN)
- Decision Tree
- Gradient Boosting Machines (GBM)
- Neural Network

These models were selected for their demonstrated capabilities in handling classification tasks, especially in scenarios similar to this study's focus on terminal operations and dwell time prediction.

3.3.1.2 Application of AHP to the Selected Algorithms

The AHP method was applied to compare these algorithms based on their performance in terms of Data Handling, Efficiency, Generalization, and Complexity. This structured process allowed for a quantitative assessment of each algorithm's strengths and weaknesses, ensuring that the final selection aligns with the practical needs of the Trieste SST Terminal. The inclusion of a diverse set of models ensures that both simple and complex approaches are evaluated, providing a robust analysis of the best-suited algorithm for the dwell time classification task.

By applying AHP to these 10 algorithms, the study aims to identify the most suitable machine learning model, balancing predictive power, operational feasibility, and ease of implementation.

To perform AHP analysis, Super Decisions software is used. Super Decisions is a specialized software developed by the Creative Decisions Foundation, which was founded by Thomas L. Saaty, the originator of the Analytical Hierarchy Process (AHP). This software is widely recognized for its robust capabilities in conducting multi-criteria decision-making analyses using AHP and related methodologies. Super Decisions was chosen for this study due to its user-friendly interface, comprehensive support for pairwise comparisons, consistency checks, and its ability to handle complex decision-making hierarchies effectively. This made it particularly well-suited for the detailed model evaluation and ranking required in this study, providing accurate and reliable results that align with the structured decision-making goals inherent to AHP.

3.3.1.3 Defining the Criteria

In this study, four main criteria were selected to assess the performance of the machine learning models which are shown in Figure 3.2 and below:

- *Data Handling*: This criterion assesses the algorithm's ability to manage complex and potentially incomplete datasets. Given the variability in trailer arrivals, operations, and external factors, the data can be noisy and prone to missing values. A model that handles such data with minimal preprocessing is highly valued.
- *Efficiency*: This measures the computational cost and time required by each model for both training and inference. Efficiency is crucial in operational environments like a Ro-Ro terminal, where real-time predictions may be necessary to optimize parking space and manage traffic flows.
- *Generalization*: This criterion evaluates how well the model can generalize to unseen data, ensuring that it performs consistently across different operational scenarios. The dwell time data at the Trieste SST Terminal may vary significantly over time due to changes in operational patterns, seasonality, or external factors, making generalization a critical factor.
- *Complexity*: This criterion takes into account the ease of implementation and interpretability of the models. While complex models might offer better performance, they could also be harder to interpret and more resource-intensive to deploy. Striking a balance between performance and simplicity is key for effective implementation in terminal operations.

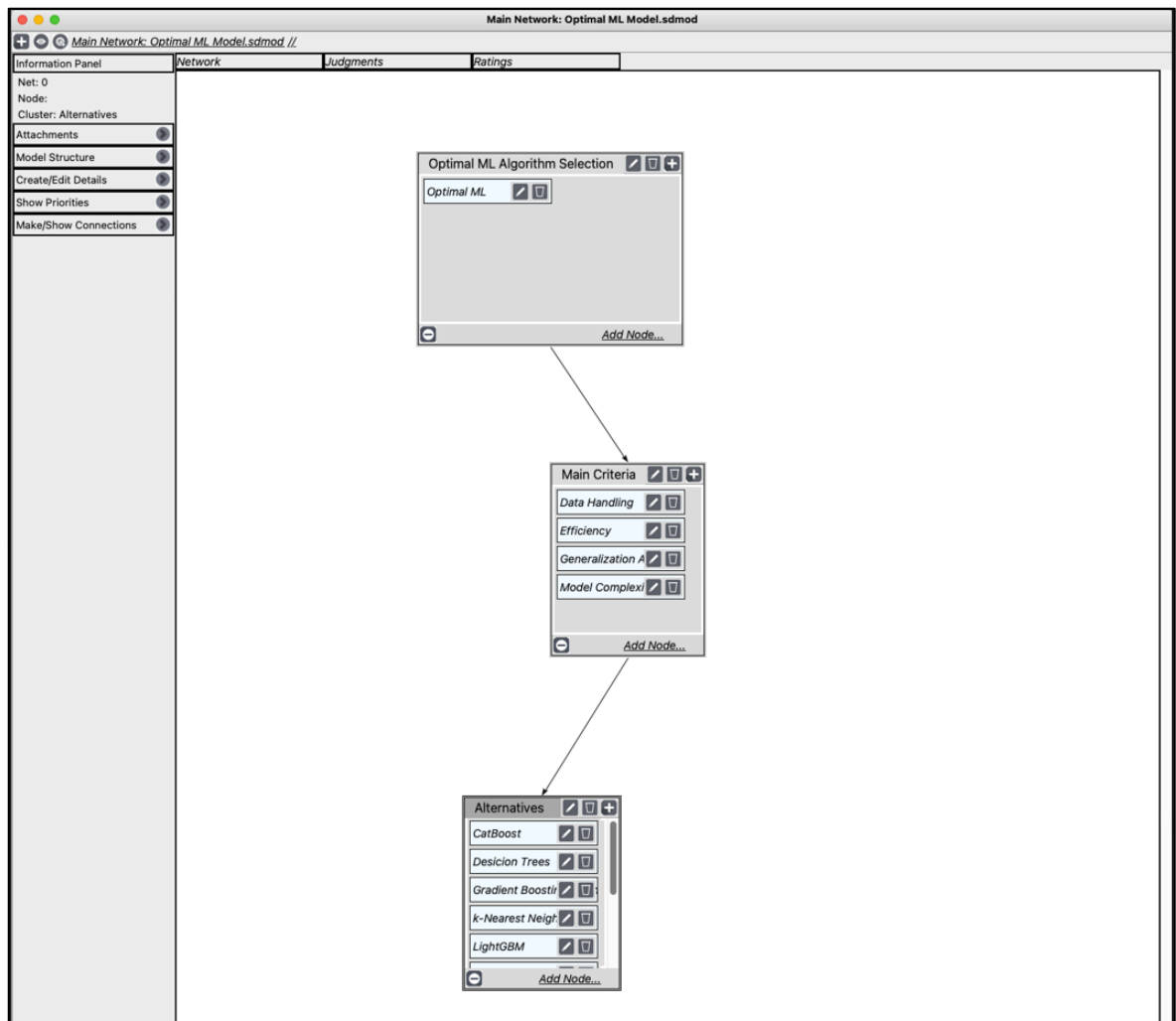


Figure 3.2 : Criteria defined in Super Decision Software.

3.3.1.4 Pairwise Comparisons of Criteria

The AHP methodology involves performing pairwise comparisons between the defined criteria to assess their relative importance. This comparison is based on the specific needs of this study and its operational context. The following judgments were made during the comparison process:

Data Handling was considered more important than Efficiency due to the variability in the quality and completeness of the available data.

Generalization was prioritized over Complexity because ensuring consistent performance across different time periods and operational conditions is critical for the long-term utility of the model.

Efficiency was prioritized over Generalization because, in the context of Ro-Ro terminal operations, the ability to make quick and efficient predictions is vital to maintaining operational fluidity. While generalization is necessary to ensure that the model performs well across different scenarios, the primary focus is on processing large datasets swiftly to support real-time decision-making.

The pairwise comparisons were entered into a software, and the relative weights of each criterion were calculated as shown in Figure 3.3, normalizing the matrix to obtain the priority vector. These weights reflect the relative importance of each criterion in selecting the most appropriate machine learning model.

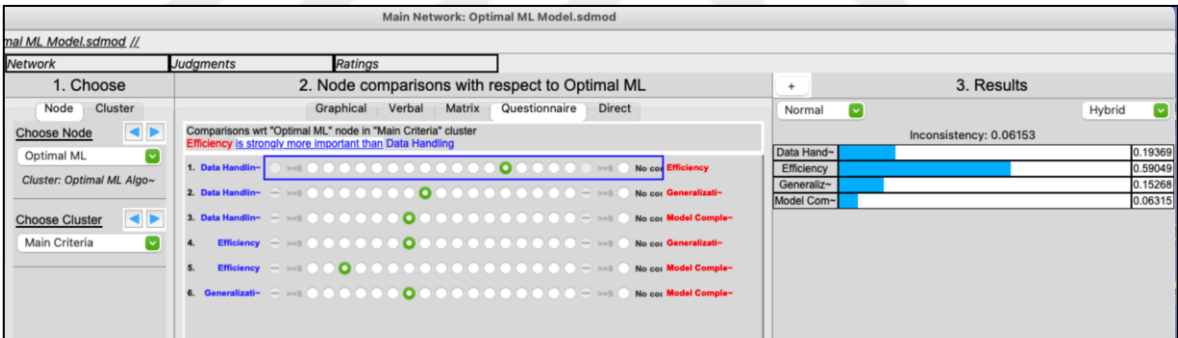


Figure 3.3 : Weights of Criteria defined in Super Decision Software.

With the highest weight of **0.59049**, **Efficiency** was determined to be the most critical criterion in the selection of the machine learning model for dwell time prediction. The second most important criterion is **Data Handling**, with a weight of **0.19369**. Although Efficiency is the top priority, the ability of a model to handle complex, noisy, or incomplete data remains a key consideration. **Generalization** carries a weight of **0.15268**, reflecting its importance in ensuring that the model performs well across various scenarios and time periods. While Generalization is less critical than Efficiency and Data Handling, it remains a significant factor. With a weight of **0.06315**, **Model Complexity** is the least important criterion in the decision-making process. This result suggests that interpretability and ease

of implementation, while still factors to consider, are less critical compared to the operational speed and data handling capabilities of the models.

The results from the AHP process show a clear prioritization of **Efficiency** as the dominant criterion, indicating that the selected model must excel in processing data quickly and efficiently to meet the operational needs of the terminal. **Data Handling** and **Generalization** are also important but hold less weight, highlighting the need for models that can manage complex datasets and perform well in different scenarios, though these factors are secondary to Efficiency. **Model Complexity** is of minimal concern in this context, meaning more complex models are acceptable as long as they meet the requirements for Efficiency and Data Handling.

Next step is to compare each selected algorithms based on importance of defined criteria. Below Figure 3.4, Figure 3.5, Figure 3.6, Figure 3.7 are presenting results of these comparisons.

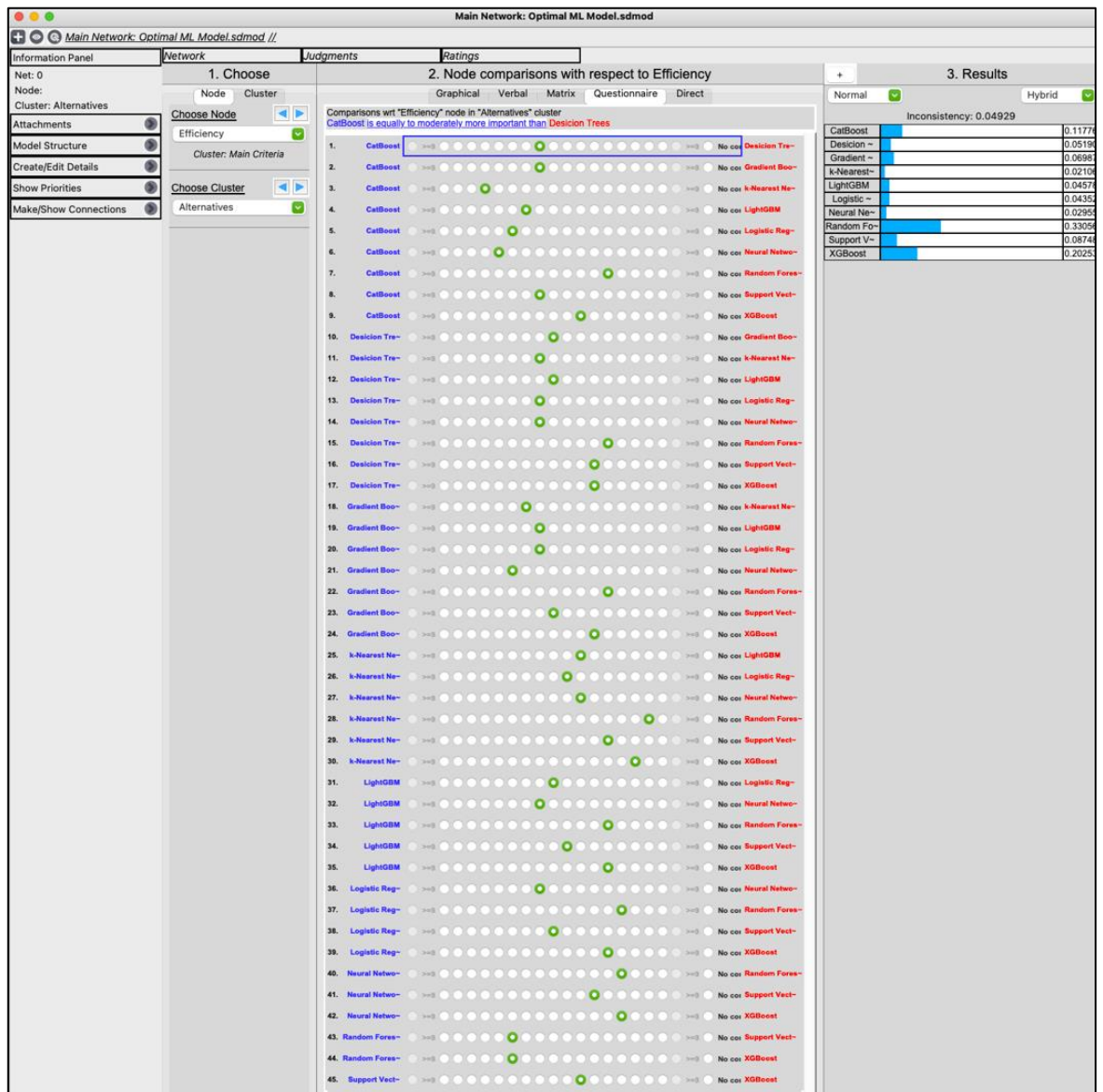


Figure 3.4 : Comparison of algorithms based on Efficiency Criteria.

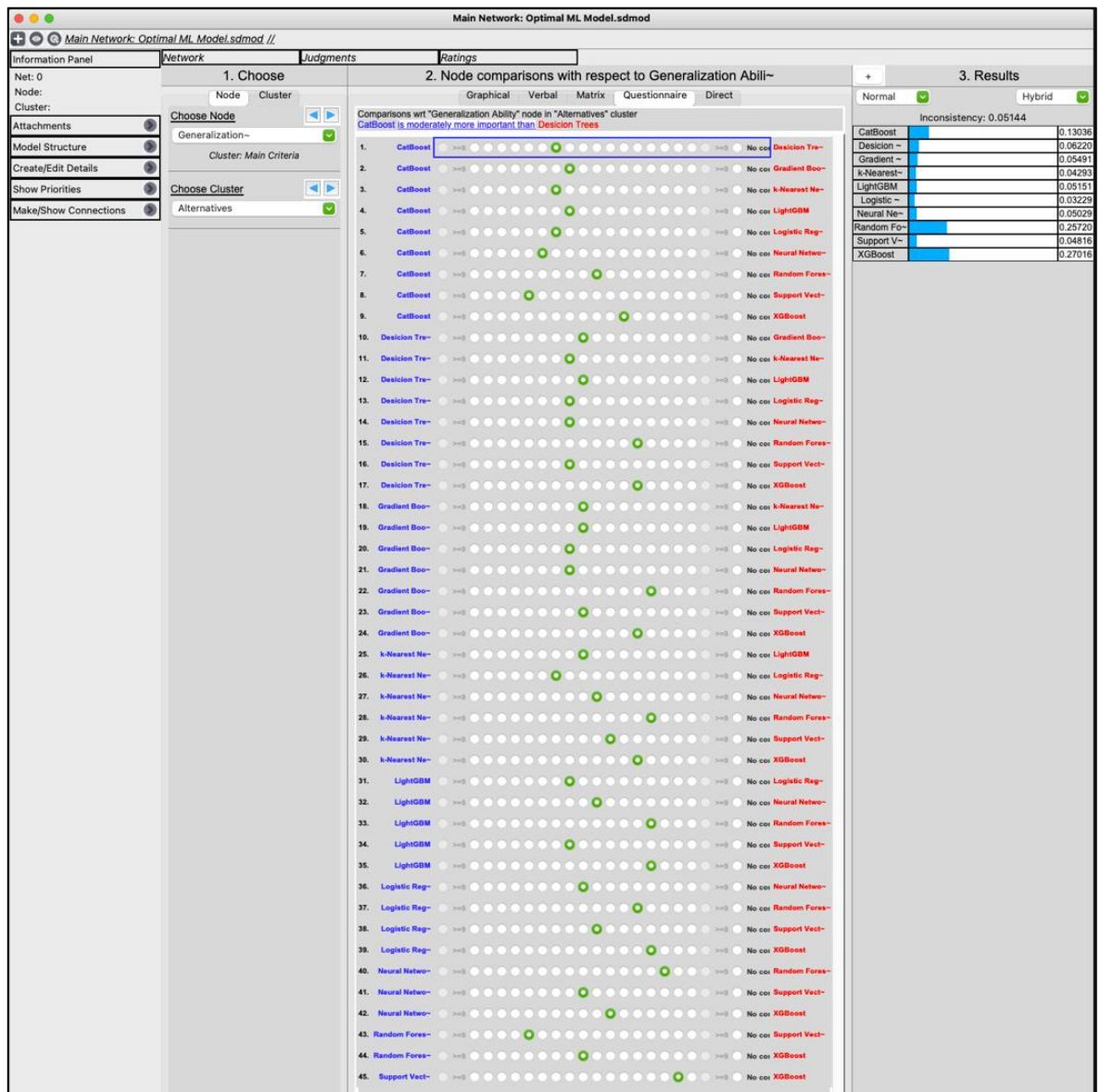


Figure 3.5 : Comparison of algorithms based on Generalization Criteria.

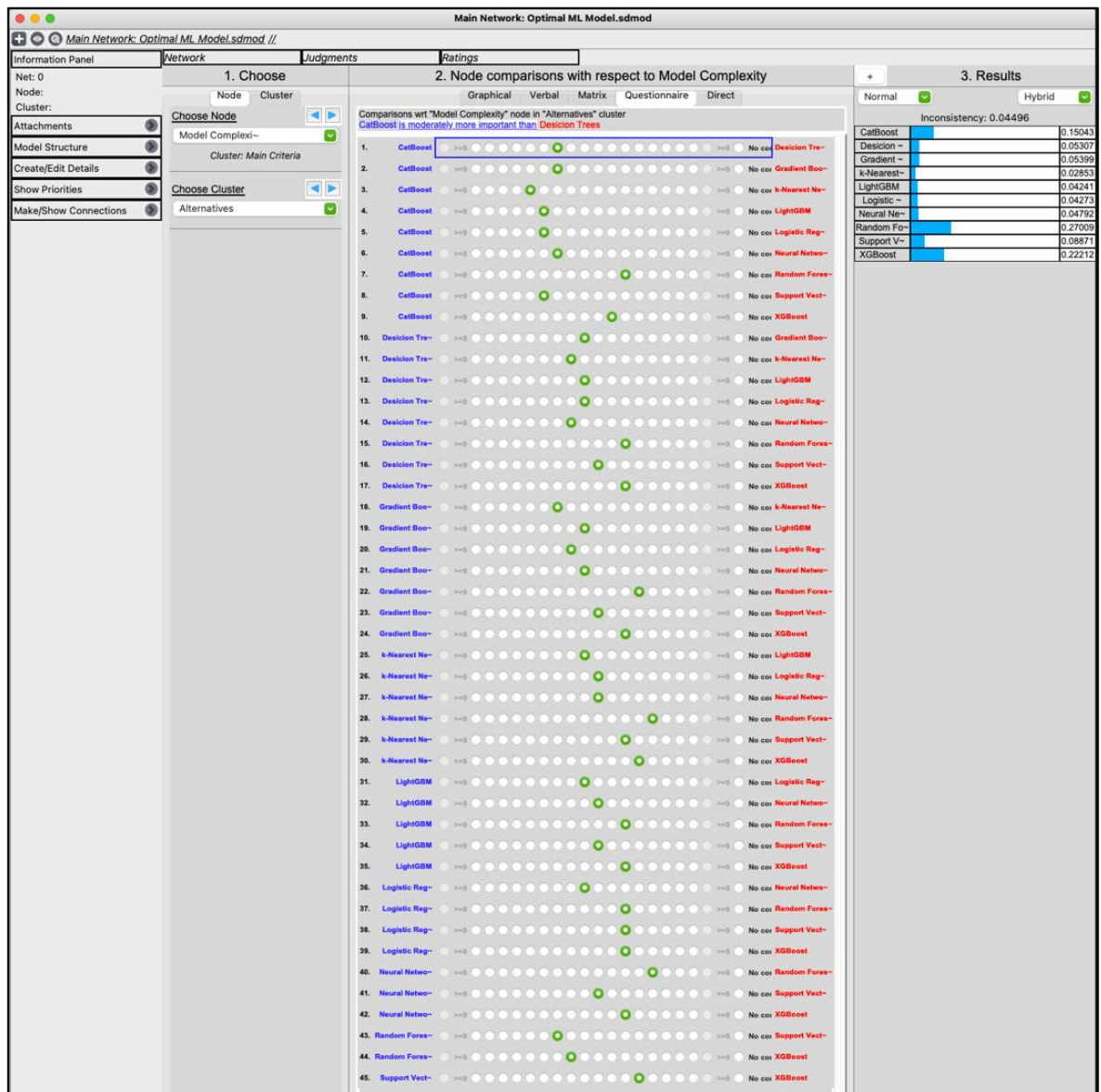


Figure 3.6 : Comparison of algorithms based on Model Complexity Criteria.

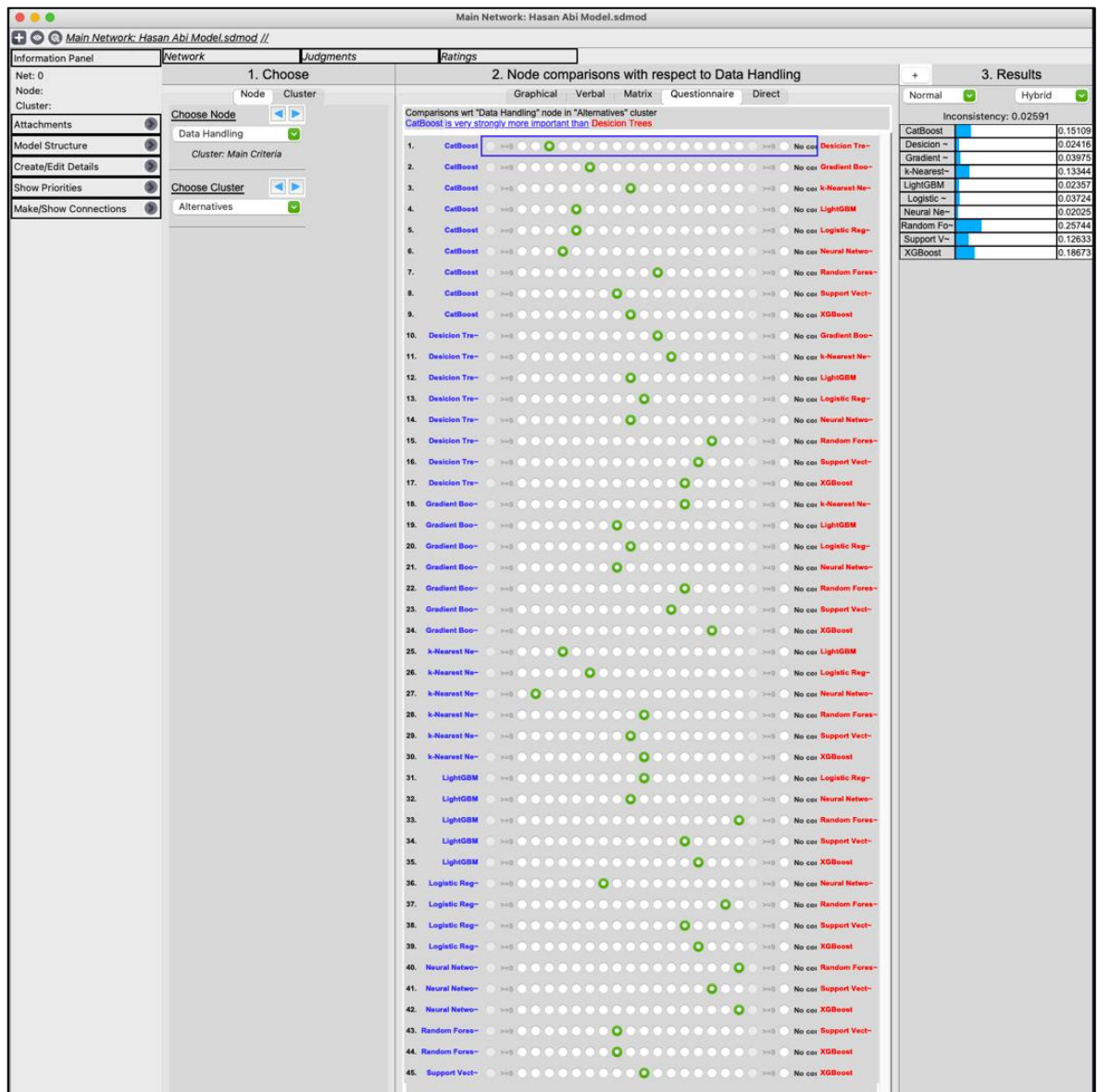


Figure 3.7 : Comparison of algorithms based on Data Handling Criteria.

3.3.1.5 Results of AHP

After completing the AHP analysis, the results for comparing the 10 machine learning algorithms based on the four criteria—Data Handling, Efficiency, Generalization, and Complexity—were synthesized using the Super Decisions software. The final rankings for each model were based on three types of scores: Ideals, Normals, and Raw values.

Here's a brief explanation of these terms:

Raw values represent the direct outcome of the synthesis, reflecting the weighted performance of each model based on the defined criteria.

Normals are the normalized scores, which rescale the raw values to a more interpretable range, usually from 0 to 1, showing the relative performance of each alternative.

Ideals reflect the ratio of each alternative's performance to the best-performing model, where the highest-performing model receives a score of 1.000.

The final scores for the machine learning algorithms are presented in Figure 3.8, with **Random Forest** achieving the highest score across all categories:

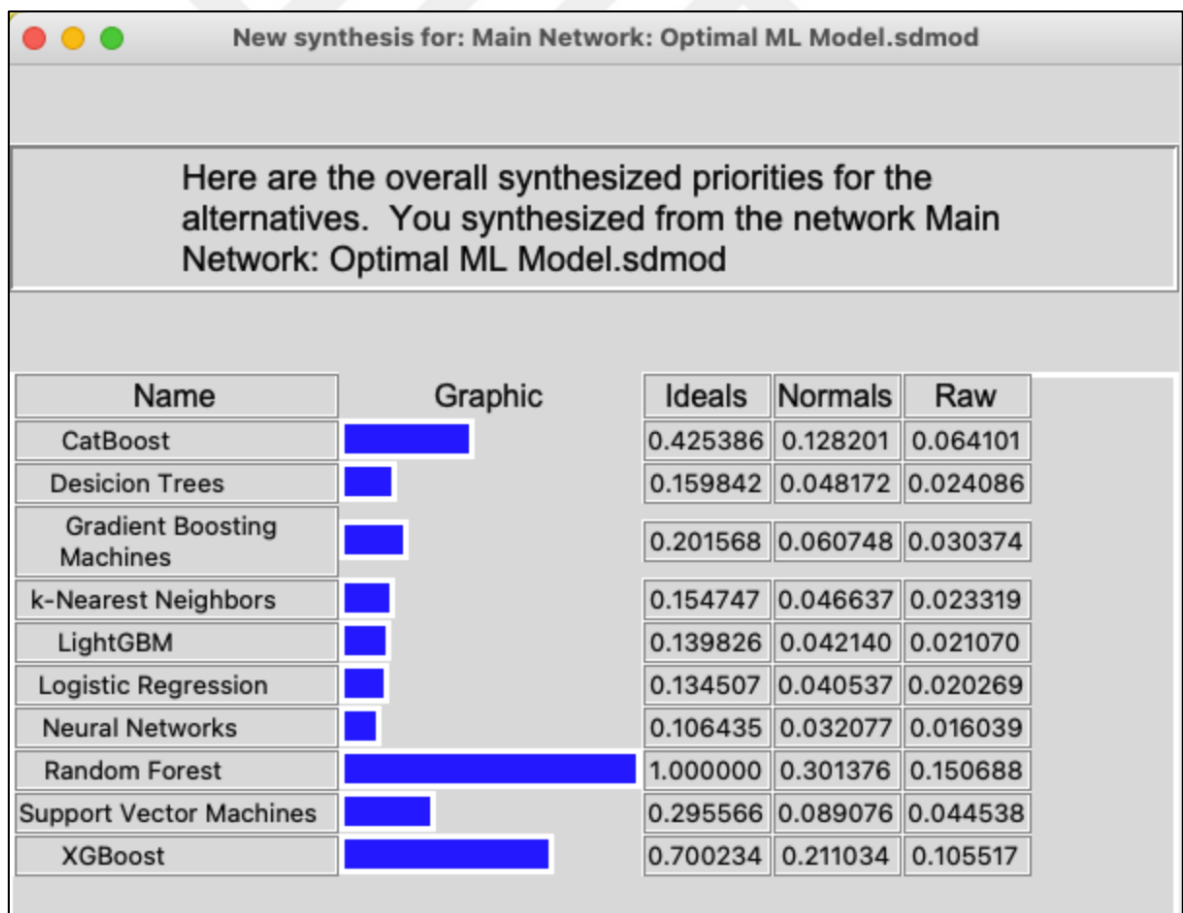


Figure 3.8 : Results of AHP.

Random Forest:

- **Ideal score:** 1.000
- **Normal score:** 0.301376
- **Raw score:** 0.150688

Random Forest was ranked the highest, excelling particularly in **Efficiency** and **Generalization**. Its strong data handling, high predictive accuracy, and computational efficiency make it the best-suited model for dwell time prediction in real-time Ro-Ro terminal operations.

XGBoost:

- **Ideal score:** 0.700234
- **Normal score:** 0.211034
- **Raw score:** 0.105517

XGBoost performed well, particularly in **Efficiency** and **Generalization**, though it was slightly behind Random Forest in terms of complexity and ease of deployment. Despite this, it remains a strong alternative for tasks requiring high performance.

CatBoost:

- **Ideal score:** 0.425386
- **Normal score:** 0.128201
- **Raw score:** 0.064101

CatBoost's ability to handle categorical variables efficiently and perform well across various data types placed it third, though it could not outperform the top two models in terms of efficiency and simplicity.

Gradient Boosting Machines (GBM):

- **Ideal score:** 0.201568
- **Normal score:** 0.060748
- **Raw score:** 0.030374

GBM's predictive power was recognized, but its computational complexity and efficiency limitations affected its overall ranking.

Support Vector Machines (SVM) and k-Nearest Neighbours (KNN):

- **SVM Ideal score:** 0.295566
- **KNN Ideal score:** 0.154747

Both SVM and KNN performed moderately well but fell behind in terms of computational efficiency and data handling capabilities, making them less suited for real-time applications.

LightGBM and Logistic Regression:

- **LightGBM Ideal score:** 0.139826
- **Logistic Regression Ideal score:** 0.134507

LightGBM had reasonable efficiency but did not perform as well as XGBoost or Random Forest, while Logistic Regression, though easy to implement, lacked the predictive power of more complex models.

Decision Trees and Neural Networks:

- **Decision Tree Ideal score:** 0.159842
- **Neural Network Ideal score:** 0.106435

Decision Trees, although highly interpretable, scored lower in efficiency and generalization. Neural Networks, despite their strength in capturing complex patterns, were penalized for their high complexity and computational demands.

Based on the AHP analysis, Random Forest emerged as the most suitable model for predicting dwell time at the Trieste SST Terminal. Its high efficiency, excellent data handling, and strong generalization capabilities make it ideal for real-time applications in a dynamic operational environment. XGBoost and CatBoost also performed well but fell short in terms of overall efficiency compared to Random Forest.

While the AHP process guided the selection of the most relevant machine learning models for this study, to further validate the AHP results, the analysis was extended by testing an additional five algorithms as CatBoost, Logistic Regression, LightGBM, Support Vector Machines (SVM), and K-Nearest Neighbours (KNN). Even though these algorithms were not considered the best fit for this specific classification problem, their inclusion enriched

the analysis and offered valuable insights into their performance in the context of dwell time prediction. Testing these additional models not only confirmed the validity of the AHP-based model selection but also contributed to the literature by highlighting the limitations and applicability of these models in similar classification prediction tasks.

The AHP study demonstrated that Efficiency, as the most weighted criterion, significantly influenced the final rankings, making models like Random Forest and XGBoost optimal choices for this specific application. The inclusion of additional algorithms, even those less suited for this classification problem, provided a more comprehensive view of the model landscape, reinforcing the robustness of the AHP results and contributing to the broader understanding of machine learning applications in Ro-Ro terminal operations.

3.3.2 Comparative Analysis

Including a diverse range of algorithms offers a comprehensive evaluation of various techniques for predicting dwell times, allowing for a detailed analysis of their performance. This broad comparison helps in identifying the strengths and weaknesses of each approach, ensuring that the most suitable model is selected for deployment in Ro-Ro terminal operations.

- **Ensemble Methods (Random Forest, XGBoost, LightGBM, CatBoost):** These algorithms are expected to perform particularly well due to their ability to reduce overfitting and improve accuracy through model averaging and boosting techniques. They are efficient at handling large, complex datasets, which makes them ideal candidates for the dwell time prediction task.
- **Support Vector Machine (SVM):** SVM provides insight into how a boundary-based classifier, which excels in high-dimensional data, performs on the dwell time prediction dataset. Its robustness in handling clear decision boundaries offers a contrast to the ensemble models.
- **Logistic Regression:** This serves as a fundamental baseline, allowing for comparison with more complex models. While it is simple and easy to interpret, it

provides valuable context for understanding how more advanced models, like Random Forest or XGBoost, improve upon basic classification tasks.

- **K-Nearest Neighbours (KNN):** KNN brings a perspective on how instance-based learning, which relies on the local structure of the data, compares with model-based approaches. Its inclusion offers a useful contrast to more computationally intensive methods like gradient boosting and ensemble models.

By including this range of algorithms, the study ensures a thorough evaluation of machine learning techniques, not only validating the results from the **AHP** process but also enriching the analysis. This comprehensive approach enhances the ability to determine the most effective algorithm for optimizing port operations and improving the efficiency of dwell time predictions. The additional algorithms, though not all perfectly suited for this task, contribute to the study by offering a broader understanding of model performance in similar classification problems, ultimately supporting both the practical application and the literature on machine learning in logistics.

3.4. Data Collection Process

3.4.1 Source Description

Collected data consists historical operational data from the Trieste SST Terminal which is operated by DFDS. This dataset includes detailed records of cargo movements and associated metadata for the years 2022, 2023, and up to mid-May 2024, totalling 143,393 rows.

- Terminal Operating System (Milos) (Figure 3.9)
- Business Intelligence Tool (Tableau)
- Booking and Agency System (Phoenix)
- Vessel Voyage Reporting System (Sertica)

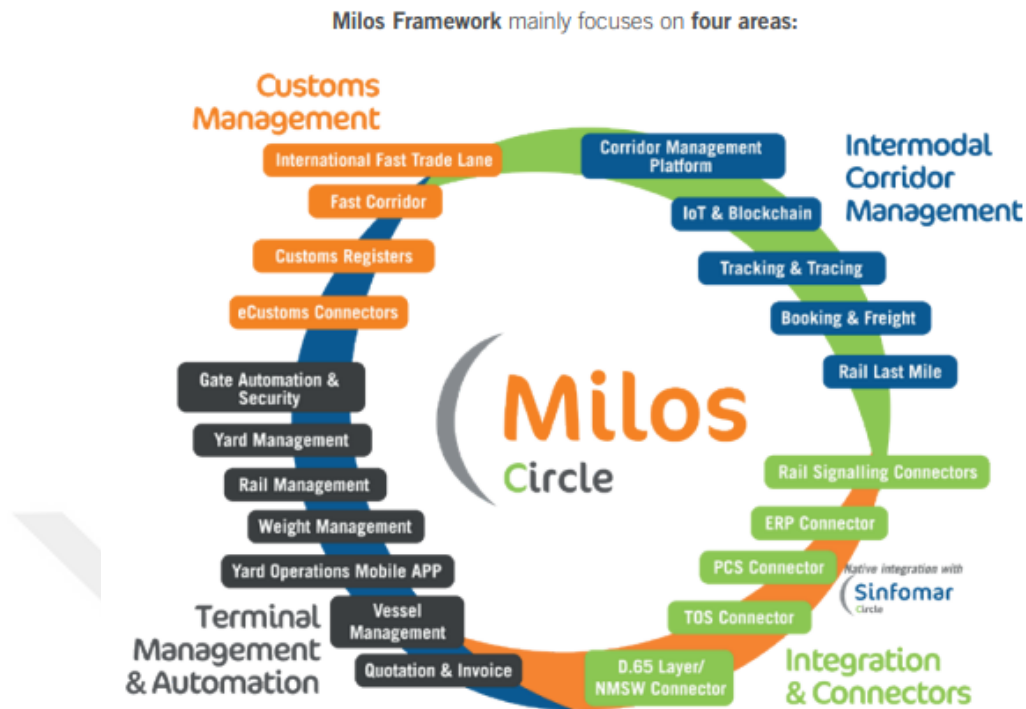


Figure 3.9 : Milos Framework.

All data belong to Terminal Operating System (MILOS), Booking and Agency System (Phoenix) and Voyage Reporting System (SERTICA) were available in Tableau business intelligence system as data sources, relevant query created to combine them in the same dataset. Terminal operational data, such as cargo discharge and exit timestamps, as well as cargo-specific attributes like cargo type, plate number, and customer information (designated as Agency Code), were collected from MILOS and the booking system Phoenix. Additionally, data pertaining to vessel information, such as vessel names, departure timestamps, and related attributes, were sourced from the Voyage Reporting System Sertica.



The methodology for predicting dwell times involves several coding steps, including data collection, data cleaning, data preprocessing, data transformation, feature engineering, model training, and evaluation of results. Each step is crucial for building an effective machine learning model. Below is a structured breakdown of the coding and followed with a graphical flowchart summarizing the entire process presented in Figure 3.11 and Figure 3.12.

Coding Methodology flow:

1. Import Libraries
2. Import Raw Data
3. Data Cleaning and Preprocessing
4. Handling Missing / Invalid Values
5. Convert Data Types
6. Date Conversion
7. Categorizing Dwell Time
8. Dropping Columns (Dwell Time - Hours)
9. Categorizing Cargo Weights
10. Dropping Columns (Cargo Tare Weight)
11. Feature Engineering (Extract Date Features)
12. Conversion of relevant features to Numeric and Categorical features
13. Creating pipeline for models
14. Using SMOTE techniques
15. Label Encoding
16. Split Data into Training, Validation, and Testing Sets
17. Model Training, Validation and Testing with Random Forest
18. Model Training, Validation and Testing with XGBoost
19. Model Training, Validation and Testing with Light GBM
20. Model Training, Validation and Testing with CatBoost
21. Model Training, Validation and Testing with SVM
22. Model Training, Validation and Testing with Logistic Regression
23. Model Training, Validation and Testing with KNN
24. Evaluation of results of the tested Models

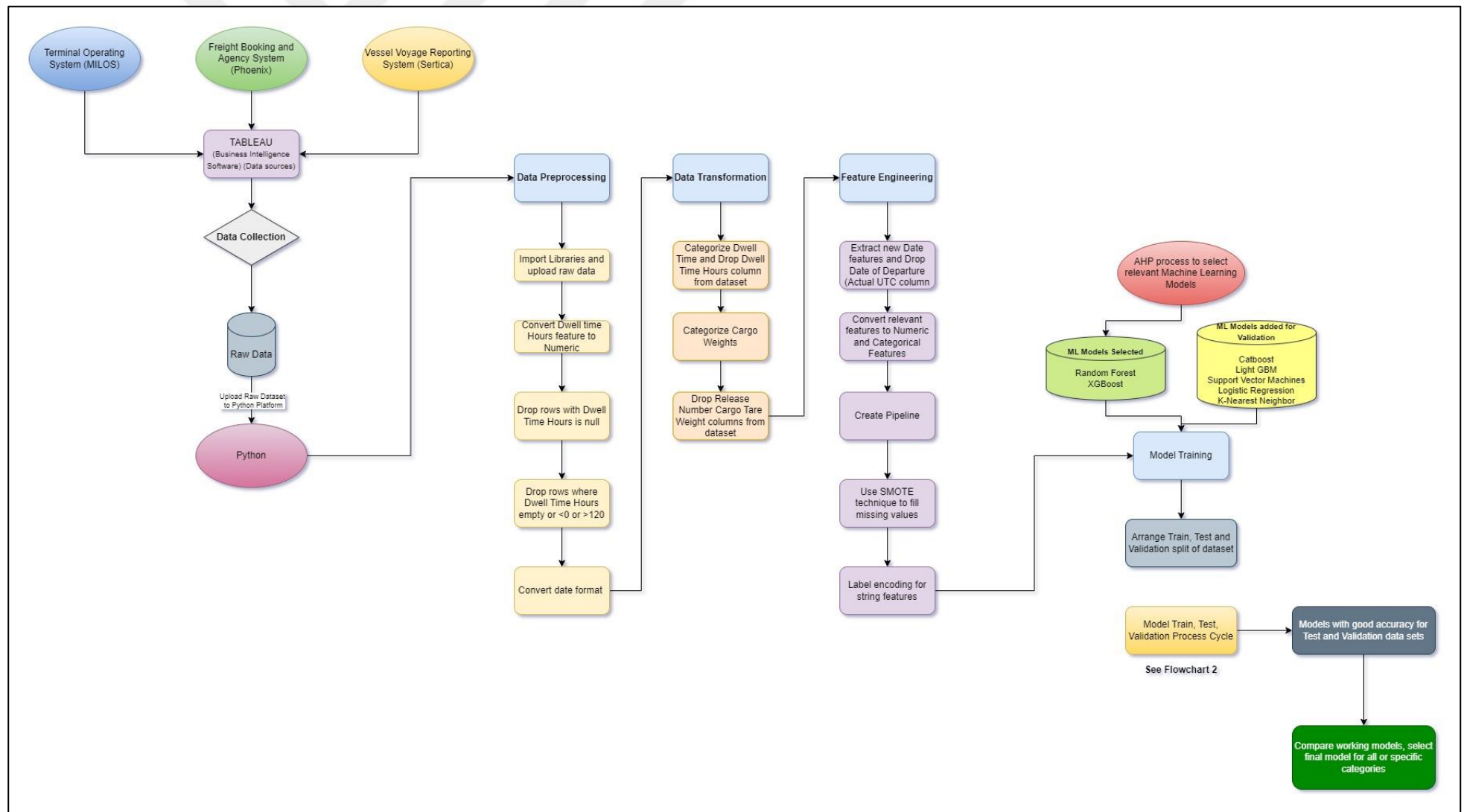


Figure 3.11 : Detailed Methodology Flowchart of Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF) model.

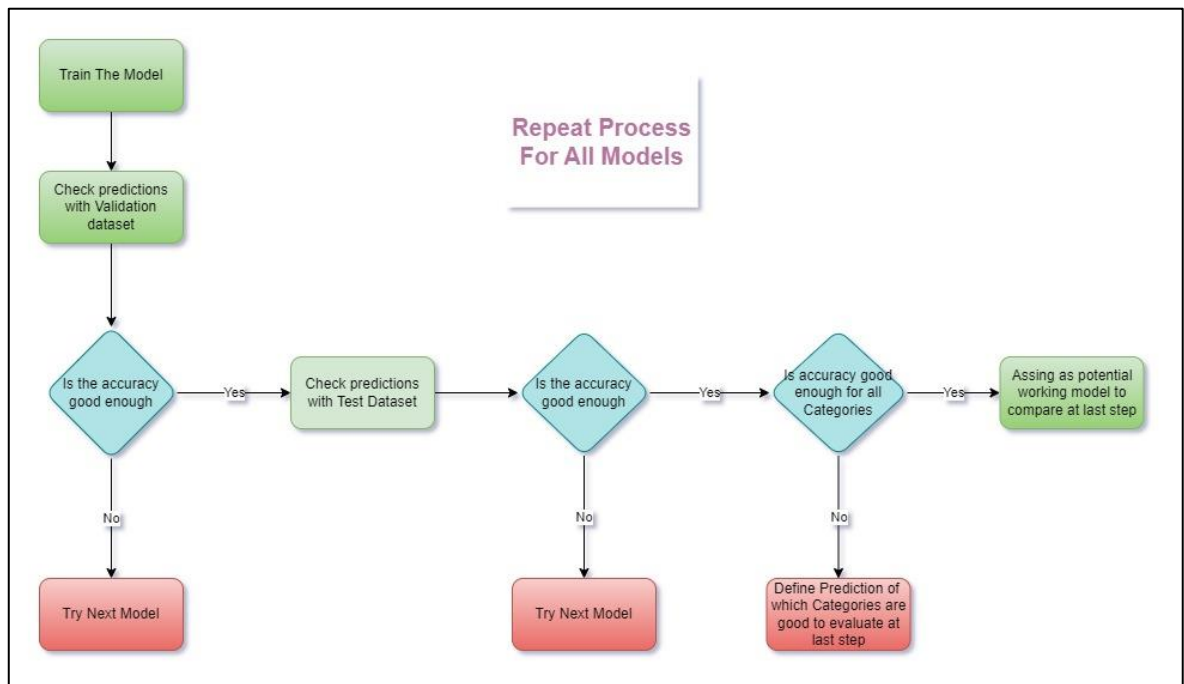


Figure 3.12 : Model Train & Test & Validation Cycle.

3.5. Data Preprocessing

3.5.1 Purpose

Preprocessing aims to clean and transform the raw dataset into a suitable form for machine learning analysis.

3.5.2 Uploading Libraries and connect to google drive to upload raw data

To begin, we need to import the necessary libraries. This usually includes libraries for data manipulation, visualization, and machine learning. In this example, we will use pandas for data manipulation, matplotlib for data visualization, and sklearn for machine learning tasks.

To import raw data into coding platform (Python), also connection with google drive folder established. Figure 3.13 shows related code to import Pandas library and connecting Python to Google Drive to give permission to reach data.

```
python

import pandas as pd
from google.colab import drive
drive.mount('/content/drive')
```

Figure 3.13 : Importing Python Libraries & connect to google drive folder.

Explanation

pandas: A powerful data manipulation and analysis library used for data cleaning, transformation, and analysis.

google.colab.drive: A module to mount Google Drive in Colab, allowing access to files stored in Google Drive. This is necessary for reading and writing data files from Google Drive.

3.5.3 Importing Raw data

Figure 3.14 presents code to import Raw data into coding platform.

```
python

# Replace 'path_to_your_file.csv' with the actual path to your CSV file
df = pd.read_csv("/content/drive/MyDrive/Dwelltime2/data2.csv", sep=';')
```

Figure 3.14 : Importing Raw data.

Explanation

pd.read_csv: Reads a comma-separated values (CSV) file into a DataFrame. The DataFrame is a two-dimensional labeled data structure with columns.

sep=';': Specifies the delimiter used in the file. In this case, the semicolon is used as the separator instead of the default comma.

3.5.4 Data Cleaning and preprocessing

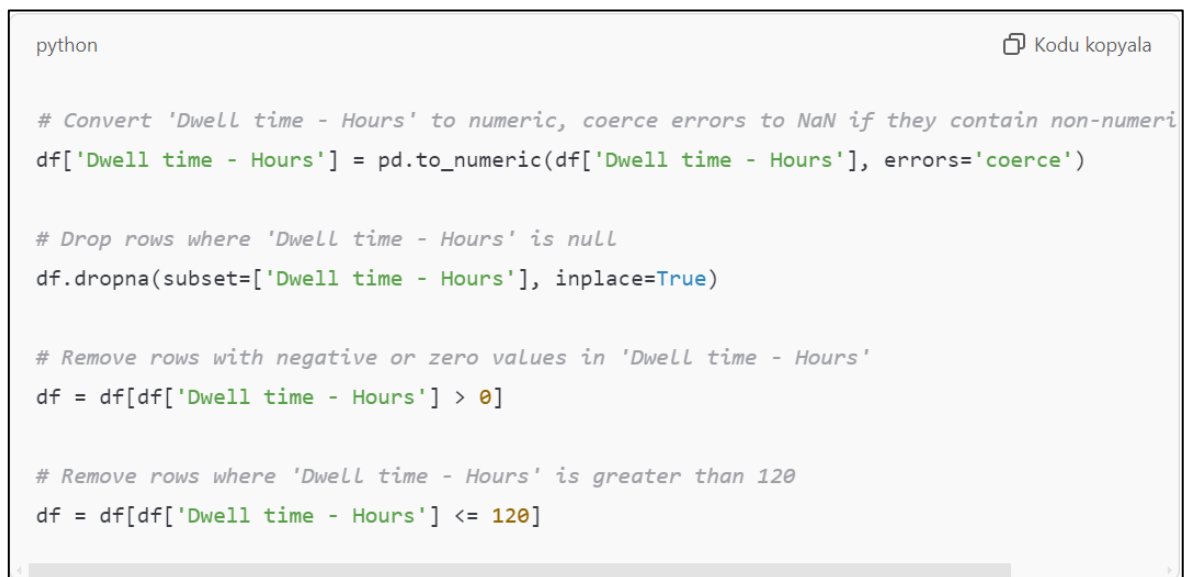
Coerced numeric columns converted into numeric types, converting errors to NaN.

Rows with null 'Dwell time - Hours' were removed. Other columns were imputed or removed if missing values were significant.

Removed negative or zero values from 'Dwell time - Hours.'

Data with dwell times above 120 hours was removed to ensure that the model is trained on the most relevant subset of data. Out of the original dataset containing 143,903 rows, 16,823 rows correspond to cargo with dwell times exceeding 120 hours. These dwell times range from 121 hours to 2,600 hours, potentially indicating the presence of anomalous data at the higher end of the range. Furthermore, 88% of the data had dwell times under 120 hours, emphasizing that most cargo stays at the terminal for up to 120 hours, making it more practical and actionable to focus on predicting dwell times within this range.

Figure 3.15 presents the code related to above steps of data preprocessing.



```
python

# Convert 'Dwell time - Hours' to numeric, coerce errors to NaN if they contain non-numeric
df['Dwell time - Hours'] = pd.to_numeric(df['Dwell time - Hours'], errors='coerce')

# Drop rows where 'Dwell time - Hours' is null
df.dropna(subset=['Dwell time - Hours'], inplace=True)

# Remove rows with negative or zero values in 'Dwell time - Hours'
df = df[df['Dwell time - Hours'] > 0]

# Remove rows where 'Dwell time - Hours' is greater than 120
df = df[df['Dwell time - Hours'] <= 120]
```

Figure 3.15 : Data cleaning and preprocessing.

Explanation

pd.to_numeric: Converts the 'Dwell time - Hours' column to a numeric type, coercing any errors (non-numeric values) to NaN. This ensures that only valid numeric data is retained for analysis.

df.dropna: Drops rows where 'Dwell time - Hours' is NaN. This is crucial to ensure the dataset only contains valid numeric entries for further analysis.

df[df['Dwell time - Hours'] > 0]: Filters out rows where 'Dwell time - Hours' is less than or equal to zero. Negative or zero dwell times are likely erroneous and should be excluded.

df[df['Dwell time - Hours'] <= 120]: Filters out rows where 'Dwell time - Hours' is greater than 120. This threshold is set to remove extreme outliers that could skew the analysis.

3.5.5 Date Conversion

Converting date strings to datetime objects helps in standardizing date formats and facilitates time-based analysis which is shown in below Figure 3.16.

```
python

# Correct the datetime conversion using the appropriate format
df['Date of Departure (Actual) UTC'] = pd.to_datetime(df['Date of Departure (Actual) UTC'])

# Check for any conversion errors or NaT values
print(df['Date of Departure (Actual) UTC'].isnull().sum())
```

Figure 3.16 : Date conversion.

Explanation

pd.to_datetime: Converts the 'Date of Departure (Actual) UTC' column to datetime format, using the specified format string. This is essential for time-based analysis and ensures that date information is in a usable format.

errors='coerce': Any errors during the conversion (e.g., incorrect format) will result in NaT (Not a Time) values. This prevents the analysis from breaking due to format inconsistencies.

.isnull().sum(): Counts the number of NaT values in the 'Date of Departure (Actual) UTC' column. This helps identify any issues with the date conversion process.

3.6. Data Transformation

Data transformation is essential to convert raw data into a more suitable format for machine learning analysis.

3.6.1 Categorizing Dwell Time

Dwell times were categorized and classified into bins to facilitate easier analysis and interpretation. Figure 3.17 presents related code for classifying Dwell Times.

```
python Kodu kopyala  
  
# Define the bins and their labels  
bins = [0, 24, 48, 72, float('inf')]  
labels = [1, 2, 3, 4]  
  
# Create a new column for the binned categories  
df['Dwell Time Category'] = pd.cut(df['Dwell time - Hours'], bins=bins, labels=labels, right=False)  
  
# Check the distribution of the new categorical variable  
print(df['Dwell Time Category'].value_counts())
```

Figure 3.17 : Categorizing dwell times.

Explanation

pd.cut: Segments and sorts of data values into bins. In this case, 'Dwell time - Hours' is categorized into intervals (bins).

bins: Specifies the edges of the bins. For example, [0, 24) represents 0 to 24 hours.

labels: Assigns labels to the bins, continuous dwell time into categorical data.

right=False: Indicates that the intervals are left-inclusive and right-exclusive. For instance, 24 is included in the bin [0, 24).

value_counts(): Counts the occurrences of each category, providing insight into the distribution of dwell times across the defined categories.

Code output presented below in Figure 3.18.

```
yml
1    44123
2    32886
4    23196
3    22377
Name: Dwell Time Category, dtype: int64
```

Figure 3.18 : Count of each dwell time category.

This output shows the count of entries in each dwell time category:

Category 1: 44,123 entries with dwell times between 0 and 24 hours.

Category 2: 32,886 entries with dwell times between 24 and 48 hours.

Category 3: 22,377 entries with dwell times between 48 and 72 hours.

Category 4: 23,196 entries with dwell times greater than 72 hours. (72 hours to 120 hours)

3.6.2 Dropping unnecessary columns (Dwell time – Hours)

The original 'Dwell time - Hours' column is dropped after converting it into categorical bins due to this column is not needed anymore as we will train models based on classified Dwell Time Categories as 0-24 hours, 24-48 hours, 48-72 hours and greater than 72 hours. Figure 3.19 shows the code to remove 'Dwell time - Hours' column from dataset.

python

Kodu kopyala

```
df = df.drop(['Dwell time - Hours'], axis=1)
```

Figure 3.19 : Drop “Dwell time – Hours” column.

Explanation

df.drop: Removes the specified column ('Dwell time - Hours') from the DataFrame.

axis=1: Indicates that a column (not a row) should be dropped. This step is performed because 'Dwell time - Hours' has been transformed into 'Dwell Time Category', making the original column redundant.

3.6.3 Categorizing cargo weights

Also 'Cargo Tare Weight' column is converted into categorical data for easier analysis and faster machine learning process. Figure 3.20 presents the code of this conversion.

python

Kodu kopyala

```
# Categorize 'Cargo Tare Weight'
weight_bins = [0, 1000, 5000, 10000, float('inf')]
weight_labels = [1, 2, 3, 4]
df['Cargo Tare Weight Category'] = pd.cut(df['Cargo Tare Weight'], bins=weight_bins, labels=weight_labels)

# Check the distribution of the new categorical variable
print(df['Cargo Tare Weight Category'].value_counts())
```

Figure 3.20 : Categorizing Cargo weights.

Explanation

pd.cut: Segments and sorts data values into bins.

weight_bins: Specifies the edges of the bins for cargo weight.

weight_labels: Assigns labels to the bins, turning continuous cargo weights into categorical data.

Figure 3.21 presents the results of code output for categorization of cargo weights.

Code Output:

Cargo Tare Weight Category	
30000	42521
10000	26129
15000	24103
20000	20534
5000	8615
higher than 30000	680
Name: count, dtype: int64	

Figure 3.21 : Count of each Cargo weight category.

Category higher than 30000: 680 entries with Cargo weights higher than 30000 kg

Category 30000: 42,521 entries with Cargo weights between 20000 to 30000 kg

Category 20000: 20,534 entries with Cargo weights between 15000 to 20000 kg

Category 15000: 24,103 entries with Cargo weights between 10000 to 15000 kg.

Category 10000: 26,129 entries with Cargo weights between 5000 to 10000 kg

Category 5000: 8,615 entries with Cargo weights between 1 to 5000 kg

3.6.4 Dropping unnecessary columns (Cargo Tare Weight)

The original 'Cargo Tare Weight' column is dropped after converting it into categorical bins due to this column is not needed anymore as we will train models based on classified Cargo Weight categories. Figure 3.22 presents the code of removing 'Cargo Tare Weight' column from the dataset.

```
python

df = df.drop(['Cargo Tare Weight'], axis=1)
```



Figure 3.22 : Drop “Cargo Tare Weight” column.

Explanation

df.drop: Removes the specified column ('Cargo Tare Weight') from the DataFrame.

axis=1: Indicates that a column (not a row) should be dropped. This step is performed because 'Cargo Tare Weight' has been transformed into 'Cargo Tare Weight Category', making the original column redundant.

3.7. Feature Engineering

Feature engineering is performed to create new and meaningful features that can improve the model's predictive capability.

3.7.1 Extracting data with new date relevant columns

'Date of Departure (Actual) UTC' column extracted to columns as Year, Month, Day and Weekday to enrich the dataset for analysis and modelling, then Original 'Date of Departure (Actual) UTC' column removed from the dataset. Figure 3.23 presents the code relevant to this process.

python

Kodu kopyala

```
# Assume 'Date of Departure (Actual) UTC' is your datetime column
# Convert datetime to more useful features
df['Departure_Year'] = df['Date of Departure (Actual) UTC'].dt.year
df['Departure_Month'] = df['Date of Departure (Actual) UTC'].dt.month
df['Departure_Day'] = df['Date of Departure (Actual) UTC'].dt.day
df['Departure_Weekday'] = df['Date of Departure (Actual) UTC'].dt.weekday

# Drop the original datetime column if it's no longer needed
df = df.drop(['Date of Departure (Actual) UTC'], axis=1)
```

Figure 3.23 : Create new date relevant columns.

Explanation

df['Date of Departure (Actual) UTC'].dt.year: Extracts the year from the datetime column.

df['Date of Departure (Actual) UTC'].dt.month: Extracts the month from the datetime column.

df['Date of Departure (Actual) UTC'].dt.weekday: Extracts the day of the week (0=Monday, 6=Sunday) from the datetime column.

df.drop(['Date of Departure (Actual) UTC'], axis=1): Removes the original datetime column if it's no longer needed, simplifying the dataset.

3.7.2 Preprocessing Pipeline for Numeric and Categorical data

This code sets up a preprocessing pipeline for preparing numerical and categorical data before feeding it into a machine learning model. The pipeline helps ensure that the data is clean, consistent, and ready for modelling.

Preprocessing steps transform the data before feeding into machine learning model. Missing values in numeric features are imputed and scaled using the median. To address categorical

columns, convert them into binary (using one hot encoding). Pipelines guarantee processing steps are consistently and efficiently executed.

This standardizes the data preprocessing pipeline. Specifies that numerical features are properly scaled, and categorical features transformed into a format suitable for model training. This way, we would apply other transformations if only the specific data type of that column is matched and as before even here same code can be used to process almost all kinds of datasets with various datatypes using different transformers for different columns via Column Transformer along with a Pipeline. The pipeline is quite flexible and there are potentially endless ways to extend or alter it in the future if more preprocessing steps needed.

Figure 3.24 presents the code for importing required libraries for pipeline preprocessing.

```
python Kodu kopyala  
  
from sklearn.impute import SimpleImputer  
import pandas as pd  
from sklearn.preprocessing import OneHotEncoder, StandardScaler  
from sklearn.compose import ColumnTransformer  
from sklearn.pipeline import Pipeline
```

Figure 3.24 : Importing required libraries for Pipeline processing.

SimpleImputer: Fills missing values in the dataset.

Pandas: A popular data manipulation library.

OneHotEncoder: Converts categorical variables into binary columns.

StandardScaler: Standardizes numerical features by scaling them.

ColumnTransformer: Applies different preprocessing steps to different columns.

Pipeline: Chains multiple preprocessing steps together.

Figure 3.25 presents the code for defining numeric and Categorical features.

```
python Kodu kopyala  
  
numeric_features = ['Avg. Avg Weekly Dwell Time']  
numeric_transformer = StandardScaler()  
  
categorical_features = ['Vessel Name', 'Agent Code', 'Departure Port Name', 'Destinati  
    'Agent Country Name', 'Departure Country Name', 'Load Location',  
    'Delivery Location', 'Departure_Year', 'Departure_Month', 'Departure_Day',  
    'Departure_Weekday', 'Cargo Tare Weight Category']  
categorical_transformer = OneHotEncoder(handle_unknown='ignore')
```

Figure 3.25 : Defining Numeric and Categorical Features.

numeric_features: Identifies the columns containing numeric data that need scaling (in this case, just 'Avg. Avg Weekly Dwell Time').

categorical_features: Identifies the columns with categorical data that need to be one-hot encoded.

numeric_transformer and categorical_transformer: Define how each type of feature will be processed.

Figure 3.26 presents the code for imputing missing numeric values using the median value of the column.

```
python Kodu kopyala  
  
numeric_transformer = Pipeline(steps=[  
    ('imputer', SimpleImputer(strategy='median')), # Impute missing values with media  
    ('scaler', StandardScaler()) # Then scale features  
)
```

Figure 3.26 : Imputing missing numeric values.

Pipeline: Chains together multiple steps to handle numeric data.

SimpleImputer(strategy='median'): Imputes missing numeric values using the median value of the column.

StandardScaler(): Standardizes numeric features to have a mean of 0 and a standard deviation of 1.

Figure 3.27 presents the code about constructing preprocessor

```
python Kodu kopyala  
  
preprocessor = ColumnTransformer(  
    transformers=[  
        ('num', numeric_transformer, numeric_features),  
        ('cat', categorical_transformer, categorical_features)  
    ]  
)
```

Figure 3.27 : Constructing Preprocessor.

ColumnTransformer: Combines different transformations for different columns.

(*'num'*, *numeric_transformer*, *numeric_features*): Applies numeric transformations (imputation and scaling) to the specified numeric columns.

(*'cat'*, *categorical_transformer*, *categorical_features*): Applies categorical transformations (one-hot encoding) to the specified categorical columns.

3.7.3 Completion of Pipeline

In this step, code is designed to build a complete machine learning pipeline that includes preprocessing, handling class imbalance, and training a model. It also includes label encoding for categorical features before the main pipeline is applied.

Figure 3.28 presents the code for completion of pipeline and label encoding processes.

```
python Kodu kopyala  
  
from sklearn.impute import SimpleImputer  
from sklearn.compose import ColumnTransformer  
from sklearn.preprocessing import OneHotEncoder, StandardScaler, LabelEncoder  
from sklearn.pipeline import Pipeline  
from imblearn.pipeline import Pipeline as ImbPipeline  
from imblearn.over_sampling import SMOTE  
from sklearn.ensemble import RandomForestClassifier
```

Figure 3.28 : Importing Libraries for completion of pipeline and label encoding.

SimpleImputer: Used to handle missing values in numeric columns.

ColumnTransformer: Allows different preprocessing steps for different columns.

OneHotEncoder: Encodes categorical variables into binary columns.

StandardScaler: Scales numerical features to have a mean of 0 and a standard deviation of 1.

Pipeline: Chains multiple preprocessing steps together.

ImbPipeline: A version of the pipeline from imblearn to integrate with oversampling techniques like SMOTE.

SMOTE: Handles class imbalance by oversampling the minority class.

RandomForestClassifier: The classification model used in this pipeline.

Figure 3.29 presents the code of completing pipeline to make dataset ready for training the models.

```
python

pipeline = ImbPipeline(steps=[
    ('preprocessor', preprocessor),
    ('smote', SMOTE(random_state=42)),
])
```

Figure 3.29 : Code for completion of pipeline for machine learning models.

ImbPipeline: A version of Pipeline from the imblearn library that can handle resampling techniques like SMOTE.

(*'preprocessor', preprocessor*): Applies preprocessing to the data, where preprocessor is *'ColumnTransformer'* defined earlier. This function imputes missing values in numeric columns using median. It scales numeric features to have a mean of 0 and standard deviation of 1. Finally, this function encodes categorical features using one-hot encoding, converting them to binary columns.

(*'smote', SMOTE(random_state=42)*): This step applies SMOTE (Synthetic Minority Over-sampling Technique) to the data after preprocessing but before model training. SMOTE is used to handle class imbalance by generating syntetic samples for the minority class. It creates new data points that resemble existing data points but with slight variations, ensuring that the minority class is better represented in the training set. *'random_state=42'* ensures reproducibility by setting a fixed seed for random number generation.

3.7.4 Label Encoding

This code performs label encoding on specific categorical columns in a 'DataFrame'. Label encoding is a preprocessing step that converts categorical text values into numerical labels, making them suitable for many machine learning models.

Figure 3.30 presents the code required for label encoding.

```
python Kodu kopyala  
  
cat_cols = ['Unit Type Name', 'Vessel Name', 'Departure Port Name', 'Destination Port Name',  
            'Departure Country Name', 'Load Location', 'Delivery Location', 'Cargo Tare Weight Category']  
le = LabelEncoder()  
for col in cat_cols:  
    df[col] = le.fit_transform(df[col].astype(str))  
  
df.head()
```

Figure 3.30 : Code of label encoding

cat_cols: This is a list of column names that represent the categorical variables in the DataFrame. These columns contain text values that need to be transformed into numeric labels for modeling.

le: An instance of the LabelEncoder is created to transform categorical values into numeric labels.

df[col].astype(str): Converts all values in the column to strings, ensuring consistent data types.

le.fit_transform(): fit: Finds all unique categories in the column and assigns them a numeric label (e.g., "A" = 0, "B" = 1, "C" = 2, etc.).

transform: Converts the original text values in the column to their respective numeric labels.

This code efficiently converts categorical columns to numeric labels using label encoding, making the data suitable for many machine learning models. It's a common step in data preprocessing when dealing with categorical data that needs to be converted to a numerical format for modelling.

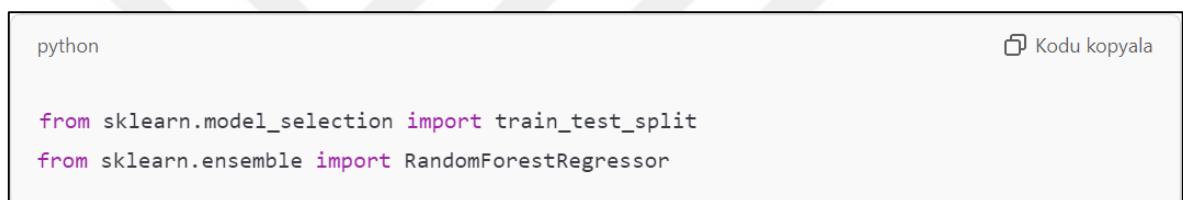
3.8. Training the model, setting test and validation datasets

3.8.1 Training, Validation and Testing process of Random Forest Model

This code splits a dataset into training, validation, and test sets to train and evaluate a machine learning model, specifically using a Random Forest Regressor.

3.8.1.1 Import Libraries for Training process

Code given in Figure 3.31 imports required libraries for model Training process.



```
python

from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
```

Figure 3.31 : Code of label encoding.

train_test_split: This function from scikit-learn splits a dataset into subsets, which are commonly used for training, validation, and testing.

RandomForestRegressor: A machine learning algorithm that uses ensemble learning with multiple decision trees to perform regression tasks.

3.8.1.2 Defining the Feature Matrix 'x' and Target Variable 'y'

Figure 3.32 presents the code for defining X and Y variables, also splitting dataset for Train and test sets.

python
Kodu kopyala

```

# Assuming 'X' is your feature matrix and 'y' is your target variable
X_train, X_temp, y_train, y_temp = train_test_split(
    df.drop(['Dwell Time Category'], axis=1),
    df['Dwell Time Category'].astype(int),
    test_size=0.2,
    random_state=42
)

```

Figure 3.32 : Definition of Feature Matrix X and Y variables.

X (Feature Matrix) creates a DataFrame containing all the features used to predict the target.

df.drop(['Dwell Time Category'], axis=1): All columns except for the target column 'Dwell Time Category'.

Y (Target Variable) represents the values the model is trying to predict.

df['Dwell Time Category'].astype(int): The column 'Dwell Time Category' is converted to integer type for modelling.

The data is split into a training set (80% of the original data) and a temporary set (20%).

test_size=0.2: Indicates that 20% of the original data will go to the temporary set.

random_state=42: Sets a fixed seed to ensure reproducibility of the split.

This split results in:

X_train: Feature matrix for training.

y_train: Target variable for training.

X_temp: Feature matrix for the temporary set.

y_temp: Target variable for the temporary set.

3.8.1.3 Splitting the Temporary Set into Validation and Test Sets

This step further splits the *temporary set* into *validation* and *test sets*, each containing 10% of the original data (since we are splitting the temporary set, which is already 20% of the original).

Figure 3.33 presents the code for splitting temporary dataset to validation and test datasets.

```
python Kodu kopyala  
  
X_val, X_test, y_val, y_test = train_test_split(  
    X_temp,  
    y_temp,  
    test_size=0.5,  
    random_state=42  
)
```

Figure 3.33 : Splitting Validation and Test Datasets.

test_size=0.5: Divides the temporary set into equal halves for validation and testing.

random_state=42: Ensures reproducibility of this split.

The resulting variables are:

X_val: Feature matrix for validation.

y_val: Target variable for validation.

X_test: Feature matrix for testing.

y_test: Target variable for testing.

Training Set (X_train, y_train) used to train the model. The model learns patterns from this data.

Validation Set (X_val, y_val) used to tune hyperparameters and evaluate the model during training. It helps prevent overfitting by evaluating model performance on unseen data during model tuning.

Test Set (X_{test} , y_{test}) used to evaluate the final model performance after training is complete. It provides an unbiased evaluation of the model's generalization ability to new, unseen data.

Overall, this section provides three-way split of the dataset:

- *80% Training Data: Used to fit the model.*
- *10% Validation Data: Used for model tuning and evaluation during training.*
- *10% Test Data: Used for final model evaluation.*

The use of a training-validation-test split is a standard approach in machine learning to ensure that the model is properly trained, tuned, and tested, leading to better generalization and performance.

3.8.1.4 Training the Random Forest model

Below Figure 3.34 presents the code for training the Random Forest Model.

```
python Kodu kopyala  
  
# Run the training process  
pipeline.fit(X_train, y_train)  
y_pred = pipeline.predict(X_test)  
  
# Evaluate the model  
from sklearn.metrics import classification_report, accuracy_score  
print("Classification Report:\n", classification_report(y_test, y_pred))  
print("Accuracy:", accuracy_score(y_test, y_pred))  
●
```

Figure 3.34 : Code to run training process of model and provides results of predictions.

pipeline.fit(X_train, y_train): This line trains (or "fits") the pipeline on the training data (X_{train} and y_{train}).

The pipeline combines preprocessing steps, any data balancing (like SMOTE), and a classifier into a single process.

During the fitting process, the pipeline applies preprocessing to the training data and trains the model using the processed data.

pipeline.predict(X_test): This line uses the trained pipeline to make predictions on the test data (X_test).

X_test: The test feature matrix, representing new, unseen data.

The model generates predictions for each instance in X_test, storing the results in y_pred.

The code imports evaluation metrics from scikit-learn which are '*classification_report*' and '*accuracy_score*'.

classification_report(y_test, y_pred): This function compares the true labels (y_test) with the predicted labels (y_pred).

It outputs below metrics:

Precision: The proportion of true positive predictions among all positive predictions. It measures how many selected items are relevant.

Recall: The proportion of true positive predictions among all actual positives. It measures how many relevant items are selected.

F1-score: The harmonic mean of precision and recall, providing a single metric that balances both concerns.

Support: The number of true instances for each class in the test set.

The classification report provides detailed metrics for each class, helping to understand which classes are predicted well and which are not.

Finally, last section of the code calculates the accuracy of the model.

`accuracy_score(y_test, y_pred)`: This function calculates the accuracy of the model, defined as the ratio of correctly predicted instances to the total number of instances.

Accuracy gives a quick overall measure of model performance but may not always provide a complete picture, especially for imbalanced datasets.

Overall, this section of the code performs three critical steps in the machine learning workflow:

- Train the pipeline with the training data.
- Predict outcomes using the trained model on the test data.
- Evaluate performance using classification metrics, which provides a comprehensive understanding of the model's accuracy and effectiveness in classifying the data.

3.8.2 Process of XGBoost Model

Due to detailed explanation of training, validation and testing processes already written in section 3.8.1, for following models, code and purpose will be explained briefly.

Below Figure 3.35 presents the relevant code for XGBoost Model. This code trains and evaluates an XGBoost classifier for a classification task, while also adjusting the class labels to ensure compatibility with the model.

```
python

from xgboost import XGBClassifier
from sklearn.metrics import classification_report, accuracy_score

# Adjust Labels to Start from 0
y_train_adjusted = y_train - 1
y_test_adjusted = y_test - 1

# Train XGBoost Classifier
xgb_model = XGBClassifier(random_state=42)
xgb_model.fit(X_train, y_train_adjusted)

# Make Predictions and Adjust Back
y_pred_xgb = xgb_model.predict(X_test)
y_pred_xgb_adjusted = y_pred_xgb + 1

# Evaluate the XGBoost Classifier
print("XGBoost Classifier Report:\n", classification_report(y_test, y_pred_xgb_adjusted))
print("Accuracy:", accuracy_score(y_test, y_pred_xgb_adjusted))
```

Figure 3.35 : Code of XGBoost Model (Training, Validation and Testing).

As summary of XGBoost Model's code performs following functions:

- It adjusts labels to ensure compatibility with XGBoost.
- It trains and predicts using XGBoost.
- Finally, evaluates performance with a classification report and accuracy score.

3.8.3 Process of Light GBM Classifier

Below Figure 3.36 presents the relevant code for Light GBM Classifier Model. This code trains and evaluates a Light GBM Classifier for a classification task.

```
python Kodu kopyala  
  
from lightgbm import LGBMClassifier  
from sklearn.metrics import classification_report, accuracy_score  
  
# Train LightGBM Classifier  
lgb_model = LGBMClassifier(random_state=42)  
lgb_model.fit(X_train, y_train)  
y_pred_lgb = lgb_model.predict(X_test)  
  
# Evaluate the LightGBM Classifier  
print("LightGBM Classifier Report:\n", classification_report(y_test, y_pred_lgb))  
print("Accuracy:", accuracy_score(y_test, y_pred_lgb))
```

Figure 3.36 : Code of Light GBM Classifier (Training, Validation and Testing).

As summary of XGBoost Model's code performs following functions:

- It trains a Light GBM model on the Training dataset
- Makes predictions on the test dataset
- Finally, evaluates performance with a classification metrics and accuracy score.

3.8.4 Process of CatBoost Model

Below Figure 3.37 presents the relevant code for CatBoost Model which is a gradient boosting model known for handling categorical features well. This code imports related library of CatBoost Model, then trains and evaluates CatBoost Model for a classification task.

```
python

# Install CatBoost library
!pip install catboost

from catboost import CatBoostClassifier
from sklearn.metrics import classification_report, accuracy_score

# Train CatBoost Classifier
cat_model = CatBoostClassifier(random_state=42, verbose=0)
cat_model.fit(X_train, y_train)
y_pred_cat = cat_model.predict(X_test)

# Evaluate the CatBoost Classifier
print("CatBoost Classifier Report:\n", classification_report(y_test, y_pred_cat))
print("Accuracy:", accuracy_score(y_test, y_pred_cat))
```

Figure 3.37 : Code of CatBoost Model (Training, Validation and Testing).

This code trains a CatBoost classifier, makes predictions on the test set, and evaluates performance using classification metrics and accuracy. CatBoost is especially effective in handling categorical data and imbalanced datasets.

3.8.5 Process of SVM (Support Vector Machines) Model

Below Figure 3.38 presents the relevant code for SVM Model. This code imports related library of SVM Model, then trains and evaluates the results.

```
python Kodu kopyala  
  
from sklearn.svm import SVC  
from sklearn.metrics import classification_report, accuracy_score  
  
# Train SVM Classifier  
svc_model = SVC(random_state=42)  
svc_model.fit(X_train, y_train)  
y_pred_svc = svc_model.predict(X_test)  
  
# Evaluate the SVM Classifier  
print("SVM Classifier Report:\n", classification_report(y_test, y_pred_svc))  
print("Accuracy:", accuracy_score(y_test, y_pred_svc))
```

Figure 3.38 : Code of SVM Model (Training, Validation and Testing).

This code trains a Support Vector Machine (SVM) classifier, makes predictions on the test set, and evaluates the performance using classification metrics and accuracy. SVM works well for both linear and non-linear classification tasks, making it a versatile choice for many types of data.

3.8.6 Process of Logistic Regression Model

Below Figure 3.39 presents the relevant code for Logistic Regression Model. This code imports related library of Logistic Regression Model, then trains and evaluates the results. As hyperparameter, maximum number of iterations set as 1000 for convergence and random state kept same as 42 as other models which is used to set a seed for reproducibility.

```
python Kodu kopyala  
  
from sklearn.linear_model import LogisticRegression  
from sklearn.metrics import classification_report, accuracy_score  
  
# Train Logistic Regression  
log_model = LogisticRegression(max_iter=1000, random_state=42)  
log_model.fit(X_train, y_train)  
y_pred_log = log_model.predict(X_test)  
  
# Evaluate the Logistic Regression  
print("Logistic Regression Report:\n", classification_report(y_test, y_pred_log))  
print("Accuracy:", accuracy_score(y_test, y_pred_log))
```

Figure 3.39 : Code of Logistic Regression Model (Training, Validation and Testing).

Logistic Regression is a simple yet effective model for linear classification tasks, often serving as a baseline for comparison with more complex models.

3.8.7 Process of K-Nearest Neighbours Model

Below Figure 3.40 presents the relevant code for K-Nearest Neighbours classifier Model.

```
python Kodu kopyala  
  
from sklearn.neighbors import KNeighborsClassifier  
from sklearn.metrics import classification_report, accuracy_score  
  
# Train KNN Classifier  
knn_model = KNeighborsClassifier()  
knn_model.fit(X_train, y_train)  
y_pred_knn = knn_model.predict(X_test)  
  
# Evaluate the KNN Classifier  
print("KNN Classifier Report:\n", classification_report(y_test, y_pred_knn))  
print("Accuracy:", accuracy_score(y_test, y_pred_knn))
```

Figure 3.40 : Code of K-Nearest Neighbours (Training, Validation and Testing).

This code trains a K-Nearest Neighbours (KNN) model, makes predictions, and evaluates performance using classification metrics and accuracy. KNN is a simple, instance-based algorithm that predicts a label based on the majority class among its closest neighbours in the training set.

3.9. Measuring Model Performance

To measure performance of machine learning algorithms, especially for classification models, several key metrics are used. These metrics include aspects of a model identifying true predictions or balancing between the classes. These include accuracy, precision, recall and F1 score which are four popular metrics.

3.9.1 Accuracy

This metric measures the percentage of correct predictions made by the algorithm on a test dataset. It is calculated by dividing the number of correct predictions to the total number of predictions. Accuracy is a simple and direct metric, but it can be misleading if the dataset is imbalanced, as it does not consider the relative importance of different classes.

Equation 3.1 presents calculation formula of Accuracy metric:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (3.1)$$

TP (True Positives): Number of positive instances correctly predicted.

TN (True Negatives): Number of negative instances correctly predicted.

FP (False Positives): Number of negative instances incorrectly predicted as positive.

FN (False Negatives): Number of positive instances incorrectly predicted as negative.

3.9.2 Precision

This metric measures the percentage of correct positive predictions made by the algorithm on a test dataset. It is calculated as the number of true positive predictions divided by the

total number of positive predictions. Precision is useful for evaluating the quality of the positive predictions made by the algorithm, but it can be affected by the relative rarity of the positive class in the dataset.

Equation 3.2 presents below of Precision calculation:

$$Precision = \frac{TP}{TP + FP} \quad 3.2$$

TP (True Positives): Number of positive instances correctly predicted.

FP (False Positives): Number of negative instances incorrectly predicted as positive.

Precision answers the question: "Of all the instances predicted as positive, how many were actually positive?"

3.9.3 Recall

Recall, also known as sensitivity or true positive rate, is the ratio of correctly predicted positive instances to the total actual positive instances. It indicates the model's ability to capture all positive instances. It is calculated as the number of true positive predictions divided by the total number of positive instances. Recall is useful for evaluating the completeness of the predictions made by the algorithm, but it can be affected by the relative rarity of the positive class in the dataset.

Equation 3.3 presents formulation of Recall metric:

$$Recall = \frac{TP}{TP + FN} \quad 3.3$$

Recall answers the question: "Of all the actual positive instances, how many were correctly predicted by the model?"

3.9.4 F1 score

The F1 score is the harmonic mean of precision and recall. It provides a balance between precision and recall, especially useful when the class distribution is imbalanced. This metric is a weighted average of precision and recall and is calculated as the harmonic mean of precision and recall. It is a balanced measure of the performance of the algorithm and is useful for comparing the performance of different algorithms on the same dataset. However, it can be affected by the relative rarity of the positive class in the dataset.

Equation 3.4 presents formulation of F1 score:

$$F1\ Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad 3.4$$

Explanation:

- The F1 score is a single metric that combines both precision and recall.
- It is particularly useful in scenarios where an equal balance between precision and recall is required.
- The harmonic mean penalizes extreme values, providing a more balanced measure.

3.9.5 Summary of Metrics

- **Accuracy:** Main metric which can be misleading with imbalanced datasets.
- **Precision:** Focuses on the accuracy of positive predictions, important when false positives are costly.
- **Recall:** Focuses on capturing all positive instances, important when false negatives are costly.
- **F1 Score:** Balances precision and recall, useful for imbalanced datasets.

Each of these metrics provides different insights into the performance of a classification model. By considering all these metrics, you can gain a comprehensive understanding of

how well your model performs and where it might need improvement, especially in the context of specific application requirements.

In addition to these metrics, there are other benchmarks and measures that can be used to evaluate the performance of machine learning algorithms, such as AUC-ROC, log-loss, and mean squared error. These metrics and benchmarks can be used to evaluate the performance of different algorithms on the same dataset, and to compare the performance of different algorithms on different datasets. However, it is important to note that the performance of an algorithm on a particular dataset may not always generalize to other datasets or tasks, and that the choice of metrics and benchmarks should be carefully considered based on the specific problem and context.

3.10. Ethical Considerations

3.10.1 Ethics Approval

Permissions: Before accessing and analysing operational data from the Ro-Ro terminal, we obtained the necessary permissions from the terminal management and the compliance department. This ensured our research followed internal regulations and maintained data security.

3.10.2 Data Usage and Security

- **Confidentiality:** Sensitive information such as customer data was anonymized to protect privacy and confidentiality. Personal identifiers and specific shipment details were excluded from the final dataset, and numerical IDs replaced identifiable data.
- **Compliance:** The data collection and analysis processes were fully compliant with international data privacy regulations, including GDPR. Secure data storage and encryption mechanisms were implemented to prevent unauthorized access.

4. RESULTS AND DISCUSSION

4.1. Overview of Model Performance - Random Forest Classifier

The Random Forest model was tested on its ability to predict the dwell time categories of import cargo. The performance metrics obtained from the classification report and overall accuracy are summarized below in Table 4.1 and Table 4.2.

Table 4.1 : Result of Random Forest Classifier

Class	Precision	Recall	F1-Score	Support
1 (0-24 hours)	0.60	0.69	0.64	4444
2 (24-48 hours)	0.46	0.45	0.45	3277
3 (48-72 hours)	0.43	0.37	0.40	2208
4 (> 72 hours)	0.55	0.49	0.52	2330

Table 4.2 : Performance metrics of Random Forest Classifier

Metric	Value
Accuracy	0.53
Macro Avg Precision	0.51
Macro Avg Recall	0.50
Macro Avg F1-Score	0.50
Weighted Avg Precision	0.52
Weighted Avg Recall	0.53
Weighted Avg F1-Score	0.52

The model achieved an overall accuracy of 52.92%.

The most critical result of the Random Forest model's performance is its ability to accurately predict the dwell time for the first 24 hours (Class 1). The precision, recall, and F1-score for this class were 0.60, 0.69, and 0.64, respectively. These metrics are higher compared to other

classes, indicating that the model performs best in predicting whether a cargo will dwell for 0-24 hours.

The Random Forest model, with an accuracy of 52.92%, shows a reasonable ability to predict cargo dwell times, with the best performance observed for the critical 0-24 hours category. This ability is essential for improving parking space optimization and overall operational efficiency in port management.

4.2. Overview of Model Performance - XGBoost Classifier

The performance of the XGBoost classifier was tested on its ability to predict the dwell time categories of cargo. The performance metrics obtained from the classification report and overall accuracy are summarized below in Table 4.3 and Table 4.4.

Table 4.3 : Result of XGBoost Classifier

Class	Precision	Recall	F1-Score	Support
1 (0-24 hours)	0.54	0.79	0.64	8686
2 (24-48 hours)	0.42	0.35	0.38	6395
3 (48-72 hours)	0.47	0.21	0.29	4475
4 (>72 hours)	0.57	0.51	0.54	4583

Table 4.4 : Performance metrics of XGBoost Classifier

Metric	Value
Accuracy	0.51
Macro Avg Precision	0.50
Macro Avg Recall	0.47
Macro Avg F1-Score	0.46
Weighted Avg Precision	0.50
Weighted Avg Recall	0.51
Weighted Avg F1-Score	0.49

The model achieved an overall accuracy of 51.34%.

The XGBoost classifier shows a significant ability to predict the dwell time for the first 24 hours (Class 1). The precision, recall, and F1-score for this class were 0.54, 0.79, and 0.64, respectively. These metrics indicate that the model is particularly strong in identifying cargos that will dwell for 0-24 hours, which is essential for operational efficiency.

The XGBoost classifier, with an accuracy of 51.34%, shows a reasonable ability to predict cargo dwell times, with the good performance observed for the critical 0-24 hours category.

4.3. Overview of Model Performance – LightGBM Classifier

The performance of the LightGBM classifier was tested on its ability to predict the dwell time categories of cargo. The performance metrics obtained from the classification report and overall accuracy are summarized below in Table 4.5 and Table 4.6.

Table 4.5 : Result of LightGBM Classifier

Class	Precision	Recall	F1-Score	Support
1 (0-24 hours)	0.50	0.83	0.63	8686
2 (24-48 hours)	0.41	0.28	0.33	6395
3 (48-72 hours)	0.47	0.17	0.25	4475
4 (>72 hours)	0.53	0.45	0.49	4583

Table 4.6 : Performance metrics of LightGBM Classifier

Metric	Value
Accuracy	0.49
Macro Avg Precision	0.48
Macro Avg Recall	0.43
Macro Avg F1-Score	0.42
Weighted Avg Precision	0.48
Weighted Avg Recall	0.49
Weighted Avg F1-Score	0.45

The model achieved an overall accuracy of 48.95%.

The LightGBM classifier shows a notable performance to predict the dwell time for the first 24 hours (Class 1). The precision, recall, and F1-score for this class were 0.50, 0.83, and 0.63, respectively. These metrics indicate that the model is particularly strong in identifying cargos that will dwell for 0-24 hours, which is essential for operational efficiency.

The LightGBM classifier, with an accuracy of 48.95%, shows a reasonable performance to predict cargo dwell times, with the good performance observed for the critical 0-24 hours category.

4.4. Overview of Model Performance – CatBoost Classifier

CatBoost classifier was tested based on performance of the dwelling time categorization for cargo. Table 4.8 and Table 4.9 below presents the summary performance metrics from classification report and overall accuracy.

Table 4.7 : Result of CatBoost Classifier

Class	Precision	Recall	F1-Score	Support
1 (0-24 hours)	0.54	0.80	0.65	8686
2 (24-48 hours)	0.43	0.34	0.38	6395
3 (48-72 hours)	0.46	0.21	0.29	4475
4 (> 72 hours)	0.56	0.52	0.54	4583

Table 4.8 : Performance metrics of Catboost Classifier

Metric	Value
Accuracy	0.51
Macro Avg Precision	0.50
Macro Avg Recall	0.47
Macro Avg F1-Score	0.46
Weighted Avg Precision	0.50
Weighted Avg Recall	0.51
Weighted Avg F1-Score	0.49

Overall model accuracy of 51.46%.

The CatBoost model does a good job of predicting the dwell time <24 hours (Class 1) with Precision: 0.54, Recall: 0.80 and F1-score: 0.65. In particular, the model has really good results when it comes to predicting cargos that will dwell 0-24 hours - a vital part of service operations.

While not high, the CatBoost classifier similarly offers modest predictive power with an overall accuracy of 51.46%, best observed in the 0-24 hours crucial category. Ability to enhance port management in terms of efficient, secure parking space optimization.

4.5. Overview of Model Performance – SVM Classifier (Support Vector Machines)

The performance of SVM classifier in prediction of cargo dwell time categories. The summary of the performance metrics from the classification report and the overall accuracy shown in below Table 4.9 and Table 4.10.

Table 4.9 : Result of SVM Classifier

Class	Precision	Recall	F1-Score	Support
1 (0-24 hours)	0.36	1.00	0.53	8686
2 (24-48 hours)	0.00	0.00	0.00	6395
3 (48-72 hours)	0.00	0.00	0.00	4475
4 (> 72 hours)	0.00	0.00	0.00	4583

Table 4.10 : Performance metrics of SVM Classifier

Metric	Value
Accuracy	0.36
Macro Avg Precision	0.09
Macro Avg Recall	0.25
Macro Avg F1-Score	0.13
Weighted Avg Precision	0.13
Weighted Avg Recall	0.36
Weighted Avg F1-Score	0.19

The Model had 35.98% in accuracy.

For the prediction in the first 24 hours (Class 1), the SVM classifier is very predictive for dwell time, but not for other classes. For that class, the precision, recall and f1 score were 0.36, 1.00 and 0.53 respectively. These metrics suggest that although the model is fairly good at predicting dwell times of 0-24 hours; it struggles with predictions of the other classes.

The accuracy of the SVM classifier which is 35.98 % has limited capacity to predict ferry cargo dwell time targets correctly for all categories. It does well predict the 0-24 hours critical category, but it fails to do this for other categories.

4.6. Overview of Model Performance – Logistic Regression

The performance of the Logistic Regression classifier was evaluated on its ability to predict the dwell time categories of cargo. The performance metrics obtained from the classification report and overall accuracy are summarized below in Table 4.11 and Table 4.12.

Table 4.11 : Result of Logistic Regression

Class	Precision	Recall	F1-Score	Support
1 (0-24 hours)	0.37	0.94	0.53	8686
2 (24-48 hours)	0.25	0.02	0.04	6395
3 (48-72 hours)	0.00	0.00	0.00	4475
4 (> 72 hours)	0.31	0.11	0.16	4583

Table 4.12 : Performance metrics of Logistic Regression

Metric	Value
Accuracy	0.36
Macro Avg Precision	0.23
Macro Avg Recall	0.27
Macro Avg F1-Score	0.18
Weighted Avg Precision	0.26
Weighted Avg Recall	0.36
Weighted Avg F1-Score	0.23

The model achieved an overall accuracy of 36.43%.

The Logistic Regression classifier shows a notable ability to predict the dwell time for the first 24 hours (Class 1) but performs poorly for other categories. The precision, recall, and F1-score for this class were 0.37, 0.94, and 0.53, respectively. These metrics indicate that while the model is strong in identifying cargos that will dwell for 0-24 hours, it struggles to predict the other classes accurately.

The Logistic Regression classifier, with an accuracy of 36.43%, shows limited ability to predict cargo dwell times accurately for all categories. It performs well in predicting the critical 0-24 hours category but fails to predict other categories.

4.7. Overview of Model Performance – KNN results (K-Nearest Neighbours)

The K-Nearest Neighbours (KNN) classifier was trained to predict the dwell time categories for cargos. Table 4.13 and Table 4.14 below presents the summary for performance metrics we have received from classifier and overall accuracy.

Table 4.13 : Result of KNN Classifier

Class	Precision	Recall	F1-Score	Support
1 (0-24 hours)	0.42	0.61	0.50	8686
2 (24-48 hours)	0.28	0.28	0.28	6395
3 (48-72 hours)	0.23	0.14	0.17	4475
4 (>72 hours)	0.31	0.18	0.23	4583

Table 4.14 : Performance metrics of KNN Classifier

Metric	Value
Accuracy	0.35
Macro Avg Precision	0.31
Macro Avg Recall	0.30
Macro Avg F1-Score	0.29
Weighted Avg Precision	0.33
Weighted Avg Recall	0.35
Weighted Avg F1-Score	0.33

The model obtained a final accuracy of 35.16%.

However, the KNN is only average in predicting Class 1 for the first 24 hours, with a poor performance in other places. 0.42 (recall), 0.61 (precision), F1-score of 0.50 of this class. These metrics show that the model can do great at predicting cargos will spend 0-24 hours but is not able to predict the other classes as well.

The performance of the KNN classifier is not good enough, accounting for 35.16% from all classifiers in predicting the cargo dwell times. This can easily catch the critical 0-24 hours category, which is good, but poorly predict any other categories.

4.8. Interpretation of Results

4.8.1 Category 1 (First 24 Hours)

Prediction results for the first 24 hours are very promising. Models found significant relation for the cargo which are expected to stay in terminal less than 24 hours.

Table 4.15 : Comparison of models for first 24 hours prediction results

Model	Precision	Recall	F1-Score	Support
Random Forest	0.60	0.69	0.64	4444
XGBoost	0.54	0.79	0.64	8686
LightGBM	0.50	0.83	0.63	8686
CatBoost	0.54	0.80	0.65	8686
SVM	0.36	1.00	0.53	8686
Logistic Reg.	0.37	0.94	0.53	8686
KNN	0.42	0.61	0.50	8686

Recall: LightGBM (0.83) and CatBoost (0.80) have the highest recall, meaning they are the best at identifying cargos that will dwell for 0-24 hours.

F1-Score: CatBoost (0.65) slightly outperforms other models in terms of F1-score, indicating a good balance between precision and recall.

Precision: Random Forest (0.60) and CatBoost (0.54) show the highest precision, meaning they have fewer false positives in predicting this category.

SVM and Logistic Regression, these models have the highest recall but lowest precision, indicating they predict almost all cargos as 0-24 hours but with many false positives.

KNN shows moderate performance with a balance between precision and recall.

4.8.2 Category 2 (24 - 48 Hours)

Prediction results for the 2nd 24 hours were not as good as first 24 hours. But models could find relation for the cargo which are expected to stay in terminal from 24 to 48 hours which is still valuable insight for terminal operators.

Table 4.16 : Comparison of models for 24 to 48 hours prediction results.

Model	Precision	Recall	F1-Score	Support
Random Forest	0.46	0.45	0.45	3277
XGBoost	0.42	0.35	0.38	6395
LightGBM	0.41	0.28	0.33	6395
CatBoost	0.43	0.34	0.38	6395
SVM	0.00	0.00	0.00	6395
Logistic Reg.	0.25	0.02	0.04	6395
KNN	0.28	0.28	0.28	6395

Best Performance: Random Forest (0.46) shows the highest precision and recall for this class, indicating better performance than other models.

XGBoost and CatBoost: Both have similar performance with moderate precision and recall, with F1-scores around 0.38.

LightGBM: Shows lower recall (0.28) for this class, leading to lower overall performance.

SVM and Logistic Regression: Perform poorly for this class, with SVM failing to predict any instance correctly.

KNN: Performs moderately with balanced precision and recall.

4.8.3 Category 3 (48 - 72 Hours)

Prediction results for the 3rd 24 hours were also not as good as first 24 hours. But models could find relation for the cargo which are expected to stay in terminal from 48 to 72 hours which is still valuable insight for terminal operators.

Table 4.17 : Comparison of models for 48 to 72 hours prediction results

Model	Precision	Recall	F1-Score	Support
Random Forest	0.43	0.37	0.40	2208
XGBoost	0.47	0.21	0.29	4475
LightGBM	0.47	0.17	0.25	4475
CatBoost	0.46	0.21	0.29	4475
SVM	0.00	0.00	0.00	4475
Logistic Reg.	0.00	0.00	0.00	4475
KNN	0.23	0.14	0.17	4475

Random Forest: Shows the best performance in this category with the highest F1-score (0.40), indicating a balanced prediction.

XGBoost and CatBoost: Both have similar precision but lower recall, leading to moderate F1-scores (0.29).

LightGBM: Has a similar performance to XGBoost and CatBoost but with slightly lower recall.

SVM and Logistic Regression: Fail to predict any instance correctly, indicating poor performance for this class.

KNN: Shows the lowest performance with low precision and recall.

4.8.4 Category 4 (> 72 Hours)

Prediction results for greater than 72 hours were also promising like first 24 hours. Models found significant relation for the cargo which are expected to stay in terminal less than 24 hours.

Table 4.18 : Comparison of models for > 72 hours prediction results

Model	Precision	Recall	F1-Score	Support
Random Forest	0.55	0.49	0.52	2330
XGBoost	0.57	0.51	0.54	4583
LightGBM	0.53	0.45	0.49	4583
CatBoost	0.56	0.52	0.54	4583
SVM	0.00	0.00	0.00	4583
Logistic Reg.	0.31	0.11	0.16	4583
KNN	0.31	0.18	0.23	4583

XGBoost and CatBoost: Show the best performance with high precision and recall, leading to the highest F1-scores (0.54).

Random Forest: Has slightly lower recall but similar precision, resulting in a slightly lower F1-score (0.52).

LightGBM: Shows moderate performance with balanced precision and recall.

SVM and Logistic Regression: Perform poorly for this class, failing to predict instances accurately.

KNN: Shows the lowest performance with low precision and recall.

4.8.5 Evaluation of overall results

The evaluation of various models for predicting cargo dwell times reveals distinct strengths and weaknesses across different classes. Following summary describes which model performs best for each class and how these models can be leveraged for parking space optimization.

4.8.6 Best Model for Each Class

- *Class 1 (0-24 hours):*

Best Model: CatBoost with a precision of 0.54, recall of 0.80, and F1-score of 0.65.

Reason: CatBoost demonstrates a high recall and balanced F1-score, making it highly effective in identifying cargos that will dwell for 0-24 hours.

- *Class 2 (24-48 hours):*

Best Model: Random Forest with a precision of 0.46, recall of 0.45, and F1-score of 0.45.

Reason: Random Forest shows the highest precision and recall for this class, providing the most reliable predictions.

- *Class 3 (48-72 hours):*

Best Model: Random Forest with a precision of 0.43, recall of 0.37, and F1-score of 0.40.

Reason: Random Forest achieves the best balance between precision and recall, resulting in the highest F1-score for this class.

- *Class 4 (>72 hours):*

Best Model: XGBoost and CatBoost both with high precision (0.57 and 0.56) and recall (0.51 and 0.52), and similar F1-scores (0.54).

Reason: Both models perform well with balanced precision and recall, making them suitable for predicting cargos with long dwell times.

4.8.7 Overfitting and Underfitting Evaluation

The Random Forest model achieved an accuracy of approximately 52.92% and performed good performance, especially for predicting dwell times within the first 24 hours. This strong performance for the critical category suggests that the model has captured important data patterns effectively. However, the lower performance of Random Forest in other categories indicates that while the model generalizes well for the primary category, it might not have captured the relationships for longer dwell times comprehensively. This could potentially indicate a slight underfitting issue for those classes, where the model fails to capture complex patterns across all categories due to limited feature relationships.

With an overall accuracy of 51.34%, XGBoost performs well for Class 1 predictions but has lower performance for mid-range categories (24-72 hours). Similar to Random Forest, this could suggest underfitting in predicting specific nuances for these categories. XGBoost's ability to handle complex relationships typically reduces overfitting risks, but its slightly poorer performance in broader classes implies either limited data representativity or insufficient feature capturing for those ranges.

CatBoost achieved a balanced performance with high precision and recall for predicting dwell times in the first 24 hours. The balanced results across precision and recall suggest that CatBoost maintains good generalization capabilities and does not exhibit strong signs of overfitting for this class. The limited effectiveness for mid-range classes could indicate challenges in feature complexity representation, suggesting slight underfitting for classes outside the first 24 hours.

SVM and Logistic Regression models displayed limited overall accuracy (~35-36%), with a notable strength in predicting the first 24 hours dwell category but poor results for other classes. The high recall but low precision for the first class indicates that these models predict a high number of false positives, suggesting potential overfitting to the prevalent patterns in this category while failing to generalize well for others. This is a clear indication of

underfitting for mid-to-longer dwell time classes due to their linear or simpler decision boundaries, which may not capture complex feature interactions well.

KNN showed only moderate performance in predicting the first 24 hours and poor results in other categories, indicating a tendency to underfit due to its local neighbour-based approach, which struggles with high-dimensional or complex feature spaces.

LightGBM also demonstrated moderate results but indicated better generalization in predicting shorter dwell times. Its performance drop for more complex categories points toward potential underfitting due to limited feature interaction complexity.

As a summary, there is minimal evidence of significant overfitting across models, as none displayed disproportionately high training performance without comparable test performance. And the more notable challenge appears to be underfitting for mid-range dwell categories, where models failed to effectively capture and generalize patterns beyond the first 24 hours predictions. Improving feature richness, data preprocessing, and capturing nuanced patterns across all categories could help address this limitation.

4.8.8 Validation of AHP in a Data-Driven Context

The use of the Analytical Hierarchy Process (AHP) at the initial stage of this study, despite having a robust dataset, was a deliberate decision to provide a systematic and strategic framework for model selection. AHP is traditionally employed in scenarios where limited data is available; however, in this context, it served as a valuable decision-making tool to evaluate and rank machine learning algorithms against predefined criteria critical for port terminal operations, such as Data Handling, Efficiency, Generalization, and Complexity. This strategic approach was used to ensure that model selection was guided not only by empirical performance but also by operational priorities and constraints inherent to Ro-Ro terminal management.

By applying AHP, Random Forest was initially identified as the most suitable model due to its high generalization capability and efficiency, while CatBoost ranked highly for its performance in capturing complex, non-linear relationships. The subsequent data-driven

analysis confirmed the validity of these selections. Random Forest performed particularly well for predicting intermediate dwell times (24-72 hours), while CatBoost excelled in both short-term (0-24 hours) and long-term (>72 hours) predictions. This alignment between AHP-derived priorities and empirical results reinforces the strength of using AHP as a guiding framework, even in a data-rich environment, to ensure that model choices are aligned with broader operational goals.

This dual-stage approach strategic model selection using AHP followed by rigorous data-driven testing highlights the importance of combining structured decision-making methodologies with data analysis. It underscores how AHP can complement data-driven modelling by ensuring that complex, multi-faceted criteria are considered from the outset, leading to better-aligned and more robust predictive models for optimizing terminal operations.

4.9. Discussion on results

4.9.1 Approach for Improvement

According to the results of the framework used in this study which is named as “Multi-Model Synergistic Framework for Machine Learning Optimization (MMSF)” model, a hybrid model that utilises the best performing models for each class may in turn improve performance of predictions for all dwell time classes/categories. The important output and value of this study is, achieving accurate predictions for the cargoes which are expected to stay in terminal up to 24 hours or more than 72 hours.

According to results of MMSF model, suggested approach would involve:

- Class 1 (0-24 hours) Dwell Times Prediction using CatBoost
- Random Forest model fits to predict Class 2 (24-48) & Class 3 (48-72 hrs) dwell times
- Prediction of Class 4 dwell times (>72 hours) - XGBoost or CatBoost

4.9.2 Parking slots Optimization Application

Accurate prediction of dwell times can significantly enhance parking space optimization in the following ways:

- **Efficient Space Allocation:** By predicting which cargos will have short (0-24 hours) or long (>72 hours) dwell times, port operators can allocate parking spaces more effectively, reducing congestion and maximizing space utilization.
- **Operational Efficiency:** Knowing dwell times in advance allows for better planning and resource allocation, facilitating quicker turnaround times and smoother cargo flow through the port.
- **Cost Savings:** Optimized parking space usage reduces the need for moving cargos multiple times, saving time and operational costs.
- **Customer Satisfaction:** Accurate predictions improve service quality by minimizing delays and ensuring reliable cargo handling and delivery schedules.

For port operations, focusing on higher accuracy results for the first 24 hours and more than 72 hours can be a straightforward approach to implement in real-life operations. This strategy targets the most critical periods for space optimization and operational efficiency.

4.9.3 Best Model for first 24 hours and higher than 72 hours

Considering the performance across all classes, CatBoost emerges as the best model for both Class 1 (0-24 hours) and Class 4 (>72 hours). CatBoost demonstrates high recall and balanced F1-scores for both short-term and long-term dwell time predictions, making it the most suitable model for practical application in port operations.

4.9.4 Novel Contribution

This study is the first attempt to incorporate a feature indicating present congestion ratio of the terminal, which is reflected by average weekly dwell time of the terminal. This inclusion is expected to improve model accuracy as it provides more information about final states (terminal conditions) that affect dwell times. Finally, to the best of our knowledge, this is the first study in the literature that has focused on the prediction of dwell time for Ro-Ro terminals using machine learning-based models. The previous few limited studies on port operations were exclusively confined to container terminals. Hence, the implications of this study are immense, and it will enable further research in the field by building a base for future progress in the literature as well as in the shipping industry.

In summary, CatBoost and Random Forest models show strong performance in predicting different waiting time classes, while a hybrid or ensemble approach can further improve the prediction accuracy. Accurate waiting time forecasts are crucial to optimise parking space allocation, improve operational efficiency and ensure customer satisfaction in port management. The inclusion of terminal congestion as a feature represents a new contribution that is expected to improve model accuracy and overall forecast performance. By focusing on higher accuracy for the first 24 hours and more than 72 hours, port operations can implement effective strategies for space optimisation and resource management.

4.10. Limitations of the Models/Study

While this study offers valuable insights and contributions to the field of cargo dwell time prediction and parking space optimization in port operations, several limitations must be acknowledged:

- **Imbalanced Dataset:** The dataset used in this study is highly imbalanced, with significantly more instances in the 0-24 hours category compared to other categories. This imbalance can lead to biased model performance, where models may perform well on the majority class but poorly on minority classes.

- **Feature Limitations:** Although this study introduces the novel feature of the terminal's current congestion ratio, other potentially influential features might not have been included. Factors such as weather conditions, economic factors, cargo content and unexpected events could impact dwell times but are not captured in this dataset.
- **Temporal Dynamics:** The models do not account for the temporal dynamics and seasonality that might affect cargo dwell times. Patterns in dwell times may vary across different times of the year or under different operational conditions, which are not explicitly modelled in this study.
- **Model Generalization:** The study's findings are based on the specific dataset and features used. The generalizability of the models to other ports or different operational contexts may be limited without further validation and adaptation to new datasets.
- **Model Complexity:** Some of the models used, particularly ensemble methods like CatBoost and Random Forest, are complex and computationally intensive. This complexity may pose challenges for real-time implementation and scalability in operational environments with limited computational resources.
- **External Factors:** External factors such as policy changes, labour strikes, and technological advancements in port operations are not considered in this study. These factors could have a significant impact on dwell times and should be taken into account in future research.
- **Real-World Constraints:** Implementing the proposed models in real-world operations may face practical constraints such as data collection challenges, integration with existing systems, and resistance to change from stakeholders.

4.10.1 Future Directions

To address these limitations, future research could focus on the following areas:

- Incorporating a broader range of features and external factors to capture the complexity of cargo dwell times.
- Exploring temporal and seasonal effects on dwell times through time series analysis.
- Simplifying model architectures for better scalability and real-time application in port operations.
- Engaging with port stakeholders to ensure practical feasibility and integration of predictive models into operational workflows.

By addressing these limitations, future studies can build on the findings of this research to further optimize port operations and improve the accuracy and utility of cargo dwell time predictions.

5. CONCLUSION

5.1. Improvement of Ro-Ro Terminal Efficiency

This study, entitled "Parking Space Optimisation for Efficient and Effective Ro-Ro Terminal Operations", represents a pioneering effort in this field by applying machine learning techniques to predict waiting times specifically for ro-ro terminals. Unlike previous research that has predominantly focused on container terminals, this study addresses a critical gap in the literature by leveraging real field data to improve the operational efficiency of ro-ro ports.

The comprehensive analysis includes extensive data collection and pre-processing, providing a solid foundation for model training. Key attributes such as cargo details, vessel information and operational data were rigorously processed and transformed, laying the groundwork for effective feature engineering. The novel inclusion of the terminal's current congestion rate, represented as average weekly dwell time, provided additional context on terminal conditions, significantly improving model accuracy.

5.2. Key Outputs of the model developed

Multiple Machine Learning models, including CatBoost, Random Forest, XGBoost, LightGBM, SVM, Logistic Regression and KNN, were evaluated for their performance in predicting waiting time categories. In particular, the CatBoost model emerged as the most effective model in predicting short-term (0-24 hours), Random Forest model worked better for mid-term (24-48 and 48-72 hours) and Catboost and XGBoost emerged as the most effective model for long-term (>72 hours) waiting times with CatBoost achieved high precision and recall compared to XGBoost.

- ***First 24 Hours Wait Time:***

CatBoost was the most reliable model for predicting short-term waiting times, showing the highest recall (0.80) and a balanced F1 score (0.65).

Operational Impact: High accuracy in predicting the first 24 hours of waiting time is crucial for immediate operational planning and resource allocation, leading to efficient space utilisation and congestion reduction.

- ***24-48 Hours and 48-72 Hours Waiting Time:***

Random Forest performed best for these categories with balanced precision and recall, making it suitable for medium-term forecasts.

Operational Impact: Accurate medium-term forecasts help optimise turnaround times and improve overall cargo flow at the port.

- ***More than 72 Hours Standby Time:***

XGBoost and CatBoost performed well with high accuracy and recall, providing reliable long-term forecasts.

Operational Impact: Reliable long-term forecasts help plan long-term storage needs and manage long-term parking space allocation, avoiding bottlenecks.

5.3. Practical Implications

The developed forecasting models offer several strategic benefits for parking space optimisation.

- **Efficient Space Allocation:** By forecasting cargo with short (0-24 hours) and long (>72 hours) dwell times, port operators can more efficiently allocate parking spaces, reduce congestion and maximise space utilisation.
- **Operational Efficiency:** Knowing waiting times in advance allows for better planning and resource allocation, resulting in faster turnaround times and smoother cargo flow through the port.

- **Cost Savings:** Optimised parking space utilisation reduces the need to move cargo multiple times, saving time and operational costs.
- **Customer Satisfaction:** Accurate forecasts improve service quality by minimising delays and ensuring reliable cargo handling and delivery schedules.

For port operators, focusing on higher accuracy results for the first 24 hours and beyond 72 hours can be an easy approach to implement in real-life operations. The CatBoost model is recommended for this dual focus due to its strong performance in both categories.

5.4. Contribution to the Literature

This study is the first to include the terminal's current congestion rate, represented as the terminal's average weekly waiting time, as a feature in the dataset. The inclusion of this feature is expected to improve the accuracy of the models by providing additional context about the terminal conditions that may affect waiting times. Furthermore, this is the first study in the literature to address dwell time prediction for Ro-Ro terminals using machine learning techniques. Previous studies have only focused on container terminals. Therefore, this study will significantly contribute to the literature and the maritime industry, providing a foundation for future research and advancements in the field.

5.5. Suggestions for the Future Studies

Based on the findings of this study, future research could explore the following:

- **Hybrid and Ensemble Methods:** Combining the strengths of different models to further improve prediction accuracy.
- **Including feature as categorized Cargo Content** to evaluate if cargo content has an impact on prediction accuracy.
- **Temporal Dynamics:** Accounting for temporal variations and seasonality in waiting time predictions.

- Real Time Application: Development of real-time forecasting systems for dynamic port management.

5.6. Summary

In conclusion, this thesis successfully demonstrates the potential of machine learning techniques in optimising parking space allocation for Ro-Ro terminal operations. By integrating advanced prediction models into terminal management systems, port operators can achieve greater operational efficiency, resource utilisation and customer satisfaction. This study sets a precedent for future research and practical applications in this field, paving the way for more effective and efficient Ro-Ro terminal operations.

REFERENCES

- Abourraja, M.N., Kringos, N., Meijer, S., 2022. Exploiting simulation model potential in investigating handling capacity of Ro-Ro terminals: The case study of Norvik seaport. *Simul Model Pract Theory* 117. <https://doi.org/10.1016/j.simpat.2022.102513>
- Abourraja, M.N., Meijer, S., Boukachour, J., 2021. A model-driven design approach for Ro-Ro and container terminals: from requirements analysis down to simulation model implementation, in: 20th International Conference on Modeling and Applied Simulation, MAS 2021. Dime University of Genoa, pp. 9–20. <https://doi.org/10.46354/i3m.2021.mas.002>
- Abourraja, M.N., Rouky, N., Kornevs, M., Meijer, S., Kringos, N., 2023. A simulation-based decision support framework devoted to Ro–Ro terminals: Design, implementation and evaluation. *Comput Ind Eng* 180, 109248. <https://doi.org/https://doi.org/10.1016/j.cie.2023.109248>
- Al-Dahidi, S., Hammad, B., Alrbai, M., Al-Abed, M., 2024. A novel dynamic/adaptive K-nearest neighbour model for the prediction of solar photovoltaic systems' performance. *Results in Engineering* 22, 102141. <https://doi.org/https://doi.org/10.1016/j.rineng.2024.102141>
- Aminatou, M., Jiaqi, Y., Okyere, S., 2018a. Evaluating the impact of long cargo dwell time on port performance: An evaluation model of Douala International Terminal in Cameroon. *Archives of Transport* 46, 7–20. <https://doi.org/10.5604/01.3001.0012.2098>
- Aminatou, M., Jiaqi, Y., Okyere, S., 2018b. Evaluating the impact of long cargo dwell time on port performance: An evaluation model of Douala International Terminal in Cameroon. *Archives of Transport* 46, 7–20. <https://doi.org/10.5604/01.3001.0012.2098>
- Amrou, M., Boulmakoul, A., Hassan, B., 2018. Process Mining towards Big data analytics in logistics environments: RORO Terminals PhD-DELAB/CRC View project Meeting-days < Distributed Artificial Intelligence < doing innovation 2019 View project.

- Anselem Tengecha, N., Zhang, X., 2021. Status, Constraints and Strategies of Marine Traffic Flow on Dar es Salaam Port, Tanzania., in: Proceedings of the 2020 2nd International Conference on Robotics Systems and Vehicle Technology, RSVT '20. Association for Computing Machinery, New York, NY, USA, pp. 54–59.
<https://doi.org/10.1145/3450292.3450314>
- Aqil Tariq Jianguo Yan, A.S.G.M.R.K., Mumtaz, F., 2023. Mapping of cropland, cropping patterns and crop types by combining optical remote sensing images with decision tree classifier and random forest. *Geo-spatial Information Science* 26, 302–320.
<https://doi.org/10.1080/10095020.2022.2100287>
- Arof, A.M., 2018. Decision Making Model for Ro-Ro Short Sea Shipping Operations in Archipelagic Southeast Asia. *The Asian Journal of Shipping and Logistics* 34, 33–42.
<https://doi.org/10.1016/J.AJSL.2018.03.005>
- Awogbemi, O., Kallon, D.V. Von, Onuh, E.I., Aigbodion, V.S., 2021. An Overview of the Classification, Production and Utilization of Biofuels for Internal Combustion Engine Applications. *Energies (Basel)* 14. <https://doi.org/10.3390/en14185687>
- Benjamin Rolf Ilya Jackson, M.M.S.L.T.R., Ivanov, D., 2023. A review on reinforcement learning algorithms and applications in supply chain management. *Int J Prod Res* 61, 7151–7179. <https://doi.org/10.1080/00207543.2022.2140221>
- Beuran, M., Bank, W., Raballand, G., 2012a. The impact of demand on cargo dwell time in ports in SSA Zambia economics and governance View project Aid effectiveness View project.
- Beuran, M., Bank, W., Raballand, G., 2012b. The impact of demand on cargo dwell time in ports in SSA Zambia economics and governance View project Aid effectiveness View project.
- Borruso Giuseppe and Cociancich, M. and G.A. and S.A. and S.F. and T.L. and T.M., 2023. Rail Ports as Nodal Gateways in the Sea – Land Connections and the Challenges of Sustainable Globalized Markets: The Case of Adriafer and the Port of Trieste, in: Gervasi Osvaldo and Murgante, B. and R.A.M.A.C. and G.C. and S.F. and K.Y. and T.C.M. (Ed.), *Computational Science and Its Applications – ICCSA 2023 Workshops*. Springer Nature Switzerland, Cham, pp. 425–441.
- Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., Lopez, A., 2020. A comprehensive survey on support vector machine classification: Applications,

- challenges and trends. *Neurocomputing* 408, 189–215.
<https://doi.org/https://doi.org/10.1016/j.neucom.2019.10.118>
- Chen, X., Li, F., Jia, B., Wu, J., Gao, Z., Liu, R., 2021a. Optimizing storage location assignment in an automotive Ro-Ro terminal. *Transportation Research Part B: Methodological* 143, 249–281.
<https://doi.org/https://doi.org/10.1016/j.trb.2020.10.009>
- Chen, X., Li, F., Jia, B., Wu, J., Gao, Z., Liu, R., 2021b. Optimizing storage location assignment in an automotive Ro-Ro terminal. *Transportation Research Part B: Methodological* 143, 249–281. <https://doi.org/10.1016/j.trb.2020.10.009>
- Chen, X., Li, F., Jia, B., Wu, J., Gao, Z., Liu, R., 2021c. Optimizing storage location assignment in an automotive Ro-Ro terminal. *Transportation Research Part B: Methodological* 143, 249–281. <https://doi.org/10.1016/j.trb.2020.10.009>
- Cuong, T.N., Long, L.N.B., Kim, H.S., You, S.S., 2023. Data analytics and throughput forecasting in port management systems against disruptions: a case study of Busan Port. *Maritime Economics and Logistics* 25, 61–89. <https://doi.org/10.1057/s41278-022-00247-5>
- De Wibowo Muhammad Sidik, A., Ramdani, D., Sopandita, D., Zam Fadilah, A.Z., Efendi, E., 2020a. Modelling and Optimization Containers Dwell-Time in Tanjung Perak Port Indonesia, in: 2020 6th International Conference on Computing Engineering and Design (ICCED). pp. 1–4.
<https://doi.org/10.1109/ICCED51276.2020.9415805>
- De Wibowo Muhammad Sidik, A., Ramdani, D., Sopandita, D., Zam Fadilah, A.Z., Efendi, E., 2020b. Modelling and Optimization Containers Dwell-Time in Tanjung Perak Port Indonesia, in: 6th International Conference on Computing, Engineering, and Design, ICCED 2020. Institute of Electrical and Electronics Engineers Inc.
<https://doi.org/10.1109/ICCED51276.2020.9415805>
- Demirel, E., Demir, U., 2024. DEVELOPMENT OF RO-RO TRANSPORT IN THE MEDITERRANEAN.
- Deprez, M., Robinson, E.C., 2024. Chapter 3 - Regression, in: Deprez, M., Robinson, E.C. (Eds.), *Machine Learning for Biomedical Applications*. Academic Press, pp. 67–86.
<https://doi.org/https://doi.org/10.1016/B978-0-12-822904-0.00008-X>
- Dragovic, B., Kofjac, D., Meštrović, R., Dragović, B., Kofjač, D., Nikitakos, N., Škurić, M., Škraba, A., 2015. Scenario-based traffic modeling of Ro-Ro automobile terminal

in the port of Bar SCENARIO-BASED TRAFFIC MODELING OF RO-RO
AUTOMOBILE TERMINAL IN THE PORT OF BAR SCENARIO-BASED
TRAFFIC MODELING OF RO-RO AUTOMOBILE TERMINAL IN THE PORT
OF BAR.

- Ejder, E., Ceylan, B.O., Celik, M.S., Arslanoğlu, Y., 2024a. Sustainability in maritime transport: Selecting ballast water treatment for a bulk carrier. *Mar Environ Res* 106511. <https://doi.org/https://doi.org/10.1016/j.marenvres.2024.106511>
- Ejder, E., Dinçer, S., Arslanoglu, Y., 2024b. Decarbonization strategies in the maritime industry: An analysis of dual-fuel engine performance and the carbon intensity indicator. *Renewable and Sustainable Energy Reviews* 200, 114587. <https://doi.org/10.1016/j.rser.2024.114587>
- El Mekkaoui, S., Benabbou, L., Berrado, A., 2022. Machine Learning Models for Efficient Port Terminal Operations: Case of Vessels' Arrival Times Prediction, in: *IFAC-PapersOnLine*. Elsevier B.V., pp. 3172–3177. <https://doi.org/10.1016/j.ifacol.2022.10.217>
- European Commission, 2023. EU trade relations with Türkiye [WWW Document]. URL https://policy.trade.ec.europa.eu/eu-trade-relationships-country-and-region/countries-and-regions/turkiye_en (accessed 6.21.24).
- Fancello, G., Serra, P., Mancini, S., 2019. A network design optimization problem for Ro-ro freight transport in the Tyrrhenian area. *Transport Problems* 14, 63–75. <https://doi.org/10.20858/tp.2019.14.4.6>
- Farzadmehr, M., Carlan, V., Vanelslander, T., 2024. How AI can influence efficiency of port operation specifically ship arrival process: developing a cost–benefit framework. *WMU Journal of Maritime Affairs*. <https://doi.org/10.1007/s13437-024-00334-9>
- Feng, M., Shaw, S.-L., Peng, G., Fang, Z., 2020. Time efficiency assessment of ship movements in maritime ports: A case study of two ports based on AIS data. *J Transp Geogr* 86, 102741. <https://doi.org/https://doi.org/10.1016/j.jtrangeo.2020.102741>
- Ghahramani, Z., 2004. Unsupervised Learning, in: Bousquet Olivier and von Luxburg, U. and R.G. (Ed.), *Advanced Lectures on Machine Learning: ML Summer Schools 2003*, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 72–112. https://doi.org/10.1007/978-3-540-28650-9_5

- Gold, C., Sollich, P., 2003. Model selection for support vector machine classification. *Neurocomputing* 55, 221–249. [https://doi.org/https://doi.org/10.1016/S0925-2312\(03\)00375-8](https://doi.org/https://doi.org/10.1016/S0925-2312(03)00375-8)
- Gomes Mantovani, R., Horváth, T., Rossi, A.L.D., Cerri, R., Barbon Junior, S., Vanschoren, J., Carvalho, A.C.P.L.F. de, 2024. Better trees: an empirical study on hyperparameter tuning of classification decision tree induction algorithms. *Data Min Knowl Discov*. <https://doi.org/10.1007/s10618-024-01002-5>
- Hassan, R., Oloan, R., Gurning, S., Handani, D.W., 2020. Analysis of the Container Dwell Time at Container Terminal by Using Simulation Modelling, *International Journal of Marine Engineering Innovation and Research*.
- Iannone, R., Miranda, S., Prisco, L., Riemma, S., Sarno, D., 2016. Proposal for a flexible discrete event simulation model for assessing the daily operation decisions in a Ro–Ro terminal. *Simul Model Pract Theory* 61, 28–46. <https://doi.org/https://doi.org/10.1016/j.simpat.2015.11.005>
- Işıklı, E., Aydın, N., Bilgili, L., Toprak, A., 2020. Estimating fuel consumption in maritime transport. *J Clean Prod* 275, 124142. <https://doi.org/https://doi.org/10.1016/j.jclepro.2020.124142>
- Jacomino, L.D.A., Medina-Pérez, M.A., Monroy, R., Valdes-Ramirez, D., Morell-Pérez, C., Bello, R., 2021. Dwell time estimation of import containers as an ordinal regression problem. *Applied Sciences (Switzerland)* 11. <https://doi.org/10.3390/app11209380>
- James Gareth and Witten, D. and H.T. and T.R. and T.J., 2023. Linear Regression, in: *An Introduction to Statistical Learning: With Applications in Python*. Springer International Publishing, Cham, pp. 69–134. https://doi.org/10.1007/978-3-031-38747-0_3
- Jia, B., Tierney, K., Reinhardt, L.B., Pahl, J., 2022. Optimal dual cycling operations in roll-on roll-off terminals. *Transp Res E Logist Transp Rev* 159, 102646. <https://doi.org/https://doi.org/10.1016/j.tre.2022.102646>
- Jia, Q., Li, R., Li, J., Li, Z., Liu, J., 2023. Vessel traffic scheduling optimization for passenger RoRo terminals with restricted harbor basin. *Ocean Coast Manag* 246, 106904. <https://doi.org/https://doi.org/10.1016/j.ocecoaman.2023.106904>

- Keceli, Y., Aksoy, S., Aydogdu, Y.V., 2013. A simulation model for decision support in Ro-Ro terminal operations. *International Journal of Logistics Systems and Management* 15, 338–358. <https://doi.org/10.1504/IJLSM.2013.054896>
- Kim, Y., Oh, H., 2021. Comparison between Multiple Regression Analysis, Polynomial Regression Analysis, and an Artificial Neural Network for Tensile Strength Prediction of BFRP and GFRP. *Materials* 14. <https://doi.org/10.3390/ma14174861>
- Kourounioti, I., Polydoropoulou, A., Tsiklidis, C., 2016a. Development of Models Predicting Dwell Time of Import Containers in Port Container Terminals - An Artificial Neural Networks Application, in: *Transportation Research Procedia*. Elsevier B.V., pp. 243–252. <https://doi.org/10.1016/j.trpro.2016.05.061>
- Kourounioti, I., Polydoropoulou, A., Tsiklidis, C., 2016b. Development of Models Predicting Dwell Time of Import Containers in Port Container Terminals - An Artificial Neural Networks Application, in: *Transportation Research Procedia*. Elsevier B.V., pp. 243–252. <https://doi.org/10.1016/j.trpro.2016.05.061>
- Kourounioti, I., Polydoropoulou, A., Tsiklidis, C., 2016c. Development of Models Predicting Dwell Time of Import Containers in Port Container Terminals - An Artificial Neural Networks Application, in: *Transportation Research Procedia*. Elsevier B.V., pp. 243–252. <https://doi.org/10.1016/j.trpro.2016.05.061>
- Kurtuluş, E., Çetin, İ.B., 2020. Analysis of modal shift potential towards intermodal transportation in short-distance inland container transport. *Transp Policy (Oxf)* 89, 24–37. <https://doi.org/https://doi.org/10.1016/j.tranpol.2020.01.017>
- L. Jiang, Z.C., Wang, D., 2010. Improving Naive Bayes for Classification. *International Journal of Computers and Applications* 32, 328–332. <https://doi.org/10.2316/Journal.202.2010.3.202-2747>
- Langseth, H., Nielsen, T.D., 2006. Classification using Hierarchical Naïve Bayes models. *Mach Learn* 63, 135–159. <https://doi.org/10.1007/s10994-006-6136-2>
- Lee, S., Park, J., Kim, N., Lee, T., Quagliato, L., 2023. Extreme gradient boosting-inspired process optimization algorithm for manufacturing engineering applications. *Mater Des* 226, 111625. <https://doi.org/https://doi.org/10.1016/j.matdes.2023.111625>
- Lee, Y., Park, K., Lee, H., Son, J., Kim, S., Bae, H., 2024. Identifying key factors influencing import container dwell time using eXplainable Artificial Intelligence. *Maritime Transport Research* 7. <https://doi.org/10.1016/j.martra.2024.100116>

- Lennerfors, T.T., 2022. Rooting for ro-ro: Exploring how strategic choices by pulp and paper companies contributed to Sweden's specialized maritime export systems, 1960–2015. *Int J Marit Hist* 34, 63–82. <https://doi.org/10.1177/08438714221079555>
- Li, N., Chen, G., Govindan, K., Jin, Z., 2018. Disruption management for truck appointment system at a container terminal: A green initiative. *Transp Res D Transp Environ* 61, 261–273. <https://doi.org/10.1016/j.trd.2015.12.014>
- Li, Y., Pérez, I.J., Cabrerizo, F.J., Garg, H., Morente-Molinera, J.A., 2024. A belief rule-based classification system using fuzzy unordered rule induction algorithm. *Inf Sci (N Y)* 667, 120462. <https://doi.org/10.1016/j.ins.2024.120462>
- Luo, J., 2024. Intelligent Stowage Expert Decision-Making System for Ro-Ro Passenger Ships, in: *Frontiers in Artificial Intelligence and Applications*. IOS Press BV, pp. 143–149. <https://doi.org/10.3233/FAIA231186>
- Maksimavičius, R., 2004. Some elements of the Ro-Ro terminals. *Transport* 19, 75–81. <https://doi.org/10.1080/16484142.2004.9637957>
- Martín-Lorenzo, A., Expósito-Márquez, A., López-Plata, I., Expósito-Izquierdo, C., Castilla-Rodríguez, I., 2023. Decision-support system for the management of truck stays at seaports, in: *Procedia Computer Science*. Elsevier B.V., pp. 1114–1123. <https://doi.org/10.1016/j.procs.2023.10.099>
- Maulud, D., Abdulazeez, A.M., 2020. A Review on Linear Regression Comprehensive in Machine Learning. *Journal of Applied Science and Technology Trends* 1, 140–147. <https://doi.org/10.38094/jastt1457>
- Meloni, C., Pranzo, M., Samà, M., 2021. Risk of delay evaluation in real-time train scheduling with uncertain dwell times. *Transp Res E Logist Transp Rev* 152, 102366. <https://doi.org/10.1016/j.tre.2021.102366>
- Moini, N., Boile, M., Theofanis, S., Laventhal, W., 2012a. Estimating the determinant factors of container dwell times at seaports. *Maritime Economics & Logistics* 14, 162–177. <https://doi.org/10.1057/mel.2012.3>
- Moini, N., Boile, M., Theofanis, S., Laventhal, W., 2012b. Estimating the determinant factors of container dwell times at seaports. *Maritime Economics and Logistics* 14, 162–177. <https://doi.org/10.1057/mel.2012.3>
- Moini, N., Boile, M., Theofanis, S., Laventhal, W., 2012c. Estimating the determinant factors of container dwell times at seaports. *Maritime Economics and Logistics* 14, 162–177. <https://doi.org/10.1057/mel.2012.3>

- Morales-Fusco, P., Saurí, S., Spuch, B., 2010. Quality indicators and capacity calculation for RoRo terminals. *Transportation Planning and Technology* 33, 695–717.
<https://doi.org/10.1080/03081060.2010.527179>
- Muravev, D., Aksoy, S., Rakhmangulov, A., Aydogdu, Y.V., 2016. Comparing model development in discrete event simulation on Ro-Ro terminal example. *International Journal of Logistics Systems and Management* 24, 283–297.
<https://doi.org/10.1504/IJLSM.2016.076890>
- Myriam Gaete, G., González-Araya, M.C., González-Ramírez, R.G., César Astudillo, H., 2017a. A dwell time-based container positioning decision support system at a port terminal, in: *ICORES 2017 - Proceedings of the 6th International Conference on Operations Research and Enterprise Systems*. SciTePress, pp. 128–139.
<https://doi.org/10.5220/0006193001280139>
- Myriam Gaete, G., González-Araya, M.C., González-Ramírez, R.G., César Astudillo, H., 2017b. A dwell time-based container positioning decision support system at a port terminal, in: *ICORES 2017 - Proceedings of the 6th International Conference on Operations Research and Enterprise Systems*. SciTePress, pp. 128–139.
<https://doi.org/10.5220/0006193001280139>
- Özkan, E.D., Nas, S., Güler, N., 2016a. Capacity Analysis of Ro-Ro Terminals by Using Simulation Modeling Method. *The Asian Journal of Shipping and Logistics* 32, 139–147. <https://doi.org/https://doi.org/10.1016/j.ajsl.2016.09.002>
- Özkan, E.D., Nas, S., Güler, N., 2016b. Capacity Analysis of Ro-Ro Terminals by Using Simulation Modeling Method. *Asian Journal of Shipping and Logistics* 32, 139–147.
<https://doi.org/10.1016/j.ajsl.2016.09.002>
- Park, C.H., Park, H., 2008. A comparison of generalized linear discriminant analysis algorithms. *Pattern Recognit* 41, 1083–1097.
<https://doi.org/https://doi.org/10.1016/j.patcog.2007.07.022>
- Patel, H.H., Prajapati, P., 2018. Study and Analysis of Decision Tree Based Classification Algorithms. *International Journal of Computer Sciences and Engineering* 6, 74–78.
<https://doi.org/10.26438/ijcse/v6i10.7478>
- Piekutowska, M., Niedbała, G., Piskier, T., Lenartowicz, T., Pilarski, K., Wojciechowski, T., Pilarska, A.A., Czechowska-Kosacka, A., 2021. The Application of Multiple Linear Regression and Artificial Neural Network Models for Yield Prediction of Very

- Early Potato Cultivars before Harvest. *Agronomy* 11.
<https://doi.org/10.3390/agronomy11050885>
- Puisa, R., 2018. Optimal stowage on Ro-Ro decks for efficiency and safety.
- Qiu, S., Dooley, L.M., Xie, L., 2020. How servant leadership and self-efficacy interact to affect service quality in the hospitality industry: A polynomial regression with response surface analysis. *Tour Manag* 78, 104051.
<https://doi.org/https://doi.org/10.1016/j.tourman.2019.104051>
- Rivera-Lopez, R., Canul-Reich, J., Mezura-Montes, E., Cruz-Chávez, M.A., 2022. Induction of decision trees as classification models through metaheuristics. *Swarm Evol Comput* 69. <https://doi.org/10.1016/j.swevo.2021.101006>
- Russo, F., Musolino, G., Assumma, V., 2023. Ro-ro and lo-lo alternatives between Mediterranean countries: Factors affecting the service choice. *Case Stud Transp Policy* 11, 100960. <https://doi.org/https://doi.org/10.1016/j.cstp.2023.100960>
- Saha, R., Pal, S., 2024. A Hybrid Metaheuristic Algorithm Using Elitist Chemical Reaction Optimization and Learning from Knowledge Assimilation for Improving Rule-based Classification Models. *Procedia Comput Sci* 235, 701–712.
<https://doi.org/https://doi.org/10.1016/j.procs.2024.04.067>
- Sai, M.J., Chettri, P., Panigrahi, R., Garg, A., Bhoi, A.K., Barsocchi, P., 2023. An Ensemble of Light Gradient Boosting Machine and Adaptive Boosting for Prediction of Type-2 Diabetes. *International Journal of Computational Intelligence Systems* 16, 14. <https://doi.org/10.1007/s44196-023-00184-y>
- Said, G.A.E.N.A., El-Horbaty, E.S.M., 2016. An intelligent optimization approach for storage space allocation at seaports: A case study, in: 2015 IEEE 7th International Conference on Intelligent Computing and Information Systems, ICICIS 2015. Institute of Electrical and Electronics Engineers Inc., pp. 66–72.
<https://doi.org/10.1109/IntelCIS.2015.7397198>
- Sarker, I.H., 2021. Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Comput Sci* 2, 160. <https://doi.org/10.1007/s42979-021-00592-x>
- Sarkhani Benemaran, R., 2023. Application of extreme gradient boosting method for evaluating the properties of episodic failure of borehole breakout. *Geoenergy Science and Engineering* 226, 211837.
<https://doi.org/https://doi.org/10.1016/j.geoen.2023.211837>

- Shakya, A.K., Pillai, G., Chakrabarty, S., 2023. Reinforcement learning algorithms: A brief survey. *Expert Syst Appl* 231, 120495.
<https://doi.org/https://doi.org/10.1016/j.eswa.2023.120495>
- Shetty, S.H., Shetty, S., Singh, C., Rao, A., 2022. Supervised Machine Learning: Algorithms and Applications, in: *Fundamentals and Methods of Machine and Deep Learning*. John Wiley & Sons, Ltd, pp. 1–16.
<https://doi.org/https://doi.org/10.1002/9781119821908.ch1>
- Smith, D., 2021. Big Data Insights into Container Vessel Dwell Times. *Transp Res Rec* 2675, 1222–1235. <https://doi.org/10.1177/03611981211015248>
- Srikiran, S., Pavani, P.N.L., Palani, K., 2023. Modeling of Nanolubricant-Assisted Machining Process by using Multiple Regression Analysis. *J Nanomater* 2023, 3669399. <https://doi.org/https://doi.org/10.1155/2023/3669399>
- Stamatovic, K., 2020. THE ANALYSIS OF PORT GROWTH FACTORS AND RECENT TRENDS IN PORT COMPETITION IN THE NORTHERN ADRIATIC REGION.
- Stepic, D., Martincic, T., Klein, F., Vladusic, D., Costa, J.P., 2021. Machine Learning based System for Vessel Turnaround Time Prediction.
<https://doi.org/10.1109/MDM48529.2020.00060>
- Styhre, L., Winnes, H., 2019. Emissions From Ships in Ports, Green Ports. Elsevier Inc.
<https://doi.org/10.1016/b978-0-12-814054-3.00006-2>
- Sunardi, O., Somakila, Y.Y., 2020. Prioritizing Criteria for Dwell Time Efficiency in Port Logistic Process, in: *Proceedings of the 2020 International Conference on Engineering and Information Technology for Sustainable Industry, ICONETSI '20*. Association for Computing Machinery, New York, NY, USA.
<https://doi.org/10.1145/3429789.3429845>
- Suthaharan, S., 2016. Supervised Learning Algorithms, in: *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*. Springer US, Boston, MA, pp. 183–206. https://doi.org/10.1007/978-1-4899-7641-3_8
- Tadini Marcello and Borruso, G., 2022. Sea-Rail Intermodal Transport in Italian Gateway Ports: A Sustainable Solution? The Examples of La Spezia and Trieste, in: Gervasi Osvaldo and Murgante, B. and M.S. and R.A.M.A.C. and G.C. (Ed.), *Computational Science and Its Applications – ICCSA 2022 Workshops*. Springer International Publishing, Cham, pp. 156–172.

- Tiago A. Santos, C. Guedes Soares, 2020. Short Sea Shipping in the Age of Sustainable Development and Information Technology.
- Tian, Y., Zhang, Y., Zhang, H., 2023. Recent Advances in Stochastic Gradient Descent in Deep Learning. *Mathematics* 11. <https://doi.org/10.3390/math11030682>
- Tong, J., Nachtmann, H., 2017. Cargo prioritization and terminal allocation problem for inland waterway disruptions. *Maritime Economics and Logistics* 19, 403–427. <https://doi.org/10.1057/mel.2015.34>
- Torbianelli, V.A., 2000. When the road controls the sea: a case study of Ro-Ro transport in the Mediterranean. *Maritime Policy & Management* 27, 375–389. <https://doi.org/10.1080/030888300416568>
- Tsoukos Georgios and Giannopoulos, A. and B.A., 2023. Efficient Management of Operations in Ro-Ro/Ro-Pax Terminals Using a Cloud-Based Yard Management Platform, in: Nathanail Eftihia G. and Gavanas, N. and A.G. (Ed.), *Smart Energy for Smart Transport*. Springer Nature Switzerland, Cham, pp. 1205–1217.
- Unal, A.U., Arslan, O., Arıcan, O.H., 2022. Türkiye’de Ro-Ro Taşımacılığının Önemi ve Geleceği Hakkında Örnek Bir Çalışma. *Denizcilik Araştırmaları Dergisi: Amfora* 1, 60–79.
- Uyanık, T., Karatuğ, Ç., Arslanoğlu, Y., 2020. Machine learning approach to ship fuel consumption: A case of container vessel. *Transp Res D Transp Environ* 84, 102389. <https://doi.org/https://doi.org/10.1016/j.trd.2020.102389>
- van Engelen, J.E., Hoos, H.H., 2020. A survey on semi-supervised learning. *Mach Learn* 109, 373–440. <https://doi.org/10.1007/s10994-019-05855-6>
- Veenstra, A., de Waal, A., 2024. The impact of container call size evidence from simulation modelling. *Maritime Transport Research* 6. <https://doi.org/10.1016/j.martra.2024.100109>
- Wahyuningsih, T., Manongga, D., Sembiring, I., Wijono, S., 2024. Comparison of Effectiveness of Logistic Regression, Naive Bayes, and Random Forest Algorithms in Predicting Student Arguments. *Procedia Comput Sci* 234, 349–356. <https://doi.org/https://doi.org/10.1016/j.procs.2024.03.014>
- Wang, J., Wang, D., Zhang, F., Yoo, C., Liu, H., 2024. Soft sensor for predicting indoor PM2.5 concentration in subway with adaptive boosting deep learning model. *J Hazard Mater* 465, 133074. <https://doi.org/https://doi.org/10.1016/j.jhazmat.2023.133074>

- Wang, K., 2024. Port Ro-Ro Connection Bridges, in: Tao Dexin and Yan, Y. and D.D. and Z.D. (Ed.), *Handbook of Port Machinery*. Springer Nature Singapore, Singapore, pp. 1839–1858. https://doi.org/10.1007/978-981-99-4848-2_40
- Wang, S., Meng, Q., Lee, C.-Y., 2016. Liner container assignment model with transit-time-sensitive container shipment demand and its applications. *Transportation Research Part B: Methodological* 90, 135–155. <https://doi.org/https://doi.org/10.1016/j.trb.2016.04.014>
- Wang, S., Meng, Q., Liu, Z., 2013. Containership scheduling with transit-time-sensitive container shipment demand. *Transportation Research Part B: Methodological* 54, 68–83. <https://doi.org/https://doi.org/10.1016/j.trb.2013.04.003>
- Wang, X., Liu, Z., Loughney, S., Yang, Z., Wang, Y., Wang, J., 2022. Numerical analysis and staircase layout optimisation for a Ro-Ro passenger ship during emergency evacuation. *Reliab Eng Syst Saf* 217, 108056. <https://doi.org/https://doi.org/10.1016/j.res.2021.108056>
- Wang, Y., Niu, B., Wang, H., Alotaibi, N., Abozinadah, E., 2021. Neural network-based adaptive tracking control for switched nonlinear systems with prescribed performance: An average dwell time switching approach. *Neurocomputing* 435, 295–306. <https://doi.org/https://doi.org/10.1016/j.neucom.2020.10.023>
- Ward, D., Miller, R., Nikolaev, A., 2021. Evaluating three stuttering assessments through network analysis, random forests and cluster analysis. *J Fluency Disord* 67, 105823. <https://doi.org/10.1016/j.jfludis.2020.105823>
- Wong Ka-Chun and Li, Y. and Z.Z., 2016. Unsupervised Learning in Genome Informatics, in: Celebi M. Emre and Aydin, K. (Ed.), *Unsupervised Learning Algorithms*. Springer International Publishing, Cham, pp. 405–448. https://doi.org/10.1007/978-3-319-24211-8_15
- Xanthopoulos Petros and Pardalos, P.M. and T.T.B., 2013. Linear Discriminant Analysis, in: *Robust Data Mining*. Springer New York, New York, NY, pp. 27–33. https://doi.org/10.1007/978-1-4419-9878-1_4
- Xie, Y., Huynh, N., n.d. Kernel-Based Machine Learning Models for Predicting Daily Truck Volume at Seaport Terminals. <https://doi.org/10.1061/ASCETE.1943-5436.0000186>

- Yao, Q., Xu, L., Zhang, Q., 2021. Container slot allocation for time-sensitive cargo in maritime transportation: A one-phase model with consideration of port congestion. *Discrete Dyn Nat Soc* 2021. <https://doi.org/10.1155/2021/6622291>
- Yavuz, K., Aksoy, S., Aydogdu, Y.V., 2013. A simulation model for decision support in Ro-Ro terminal operations. *International Journal of Logistics Systems and Management* 15, 338–358. <https://doi.org/10.1504/IJLSM.2013.054896>
- Ye, Y., Huang, Q., Rong, Y., Yu, X., Liang, W., Chen, Y., Xiong, S., 2023. Field detection of small pests through stochastic gradient descent with genetic algorithm. *Comput Electron Agric* 206, 107694. <https://doi.org/https://doi.org/10.1016/j.compag.2023.107694>
- Yoon, J.-H., Kim, S.-W., Jo, J.-S., Park, J.-M., 2023. A Comparative Study of Machine Learning Models for Predicting Vessel Dwell Time Estimation at a Terminal in the Busan New Port. *J Mar Sci Eng* 11. <https://doi.org/10.3390/jmse11101846>
- Yuk Carrie Lin, K., 2024. Optimizing variable selection and neighbourhood size in the K-nearest neighbour algorithm. *Comput Ind Eng* 191, 110142. <https://doi.org/https://doi.org/10.1016/j.cie.2024.110142>
- Zedda, S., 2024. Credit scoring: Does XGboost outperform logistic regression? A test on Italian SMEs. *Res Int Bus Finance* 70, 102397. <https://doi.org/https://doi.org/10.1016/j.ribaf.2024.102397>
- Zhang, D., Chen, F., Mei, Z., 2023. Optimization on joint scheduling of yard allocation and transfer manpower assignment for automobile RO-RO terminal. *Transp Res E Logist Transp Rev* 177, 103256. <https://doi.org/https://doi.org/10.1016/j.tre.2023.103256>
- Zhang, Y., 2012. Support Vector Machine Classification Algorithm and Its Application, in: Liu Chunfeng and Wang, L. and Y.A. (Ed.), *Information Computing and Applications*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 179–186.