Master of Science in Engineering: AI and Machine Learning
May 2025

Sensor Free Fill-Level Forecasting and AI-Driven (NSGA-II) Route Optimization for Multi-Compartment Waste Collection

Abdulkarim Dawalibi
Adnan Altukleh

Faculty of Computing, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden

This thesis is submitted to the Faculty of Computing at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Engineering: AI and Machine Learning. The thesis is equivalent to 20 weeks of full-time studies.

The authors declare that they are the sole authors of this thesis and that they have not used any sources other than those listed in the bibliography and identified as references. They further declare that they have not submitted this thesis at any other institution to obtain a degree.

Contact Information:
Author(s):
Abdulkarim Dawalibi
E-mail: abda20@student.bth.se

Adnan Altukleh
E-mail: adat19@student.bth.se

University advisor:
Huseyin Kusetogullari
Department of Computer Science

Faculty of Computing
Blekinge Institute of Technology
SE–371 79 Karlskrona, Sweden

Internet : www.bth.se
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57

Abstract

Background: Urbanization and the increasing volume of municipal solid waste challenge traditional fixed-route collection strategies. Sensor-based smart waste systems offer a solution, but are costly to deploy at scale. This thesis explores a cost-effective alternative by combining machine learning (ML) for fill-level forecasting and evolutionary multi-objective optimization for waste collection routing, specifically targeting multi-compartment vehicles without relying on real-time sensors.
Objectives: The primary objective is to develop a modular framework comprising (1) a sensor-free waste fill-level prediction module based on sequence-to-sequence neural networks, and (2) a multi-objective route optimization module using a customized Non-dominated Sorting Genetic Algorithm II (NSGA-II). The goal is to accurately forecast container fill levels and to generate efficient collection routes that minimize both travel distance and service time under realistic operational constraints.
Methods: A synthetic fill-level dataset was generated to reflect realistic seasonality, holidays, and operational conditions. Transformer, LSTM, and Bidirectional LSTM models were trained for sequence-to-sequence forecasting. Route optimization employed a multi-objective NSGA-II algorithm, with a constraint programming pre-processor (CP-SAT) to assign valid collection days based on waste type, service frequency, and vehicle capacity constraints. Comparative benchmarking was conducted against Ant Colony Optimization (ACO) and a commercial PTV routing engine.
Results: Transformer-based models consistently outperformed LSTM baselines in forecasting accuracy across synthetic scenarios. In routing, NSGA-II generated Pareto-optimal solutions balancing distance and service time, outperforming ACO and achieving competitive results compared to the commercial PTV solver. The combined framework demonstrated that predictive, sensor-free optimization is a viable pathway to smarter and greener waste collection systems.
Conclusions: The findings confirm that sensor-free fill-level forecasting combined with evolutionary multi-objective optimization offers a scalable, cost-effective alternative for modern waste management. The modular approach allows for future integration with real-time systems and supports proactive planning strategies that reduce environmental and operational costs.

Keywords: Waste Collection, Fill-Level Forecasting, Transformer, Sequence-to-Sequence Forecasting, Multi-Objective Optimization.

Sammanfattning

Bakgrund Urbanisering och ökade avfallsmängder utmanar traditionella fasta insamlingsrutter för kommunalt avfall. Smarta avfallssystem med sensorer erbjuder dynamisk insamling, men kostnaderna för storskalig implementering är höga. Detta examensarbete undersöker kostnadseffektiva alternativ genom att kombinera maskininlärning (ML) för nivåprognoser och evolutionär multiobjektiv optimering av avfallsinsamling med fokus på flerkammerfordon, utan behov av realtidsensorer.
Syfte Det huvudsakliga syftet är att utveckla en modulär lösning bestående av (1) en sensorfri prognosmodul för fyllnadsnivåer baserad på sekvens-till-sekvens neurala nätverk, och (2) en multiobjektiv ruttoptimeringsmodul baserad på en anpassad Nondominated Sorting Genetic Algorithm II (NSGA-II). Målet är att kunna förutsäga dagliga fyllnadsnivåer och optimera insamlingsrutter med avseende på minimering av både körsträcka och servicetid under realistiska operativa begränsningar.
Metod Syntetiska fyllnadsnivådata genererades för att spegla verkliga säsongsvariationer, helgdagar och driftsmönster. Transformer-, LSTM- och Bidirectional LSTM-modeller tränades för sekvens-till-sekvens prognoser. Ruttoptimering utfördes med en NSGA-II-baserad algoritm, där en constraint programming-modell (CP-SAT) först tilldelade insamlingsdagar baserat på avfallstyp, tömningsfrekvens och fordonskapacitet. Metoderna jämfördes mot Ant Colony Optimization (ACO) och ett kommersiellt PTV-rutteringsverktyg.
Resultat Transformer-baserade modeller visade högre prognosprecision än LSTM-baserade alternativ i olika syntetiska scenarier. NSGA-II optimeraren genererade Pareto-optimala lösningar som balanserade körsträcka och servicetid, och presterade bättre än ACO samt jämförbart med PTV-lösningen. Det föreslagna ramverket demonstrerade att prediktiv, sensorfri optimering är ett genomförbart alternativ för smartare och grönare avfallshanteringssystem.
Slutsatser Resultaten bekräftar att sensorfri fyllnadsnivåprognostisering kombinerad med evolutionär multiobjektiv optimering erbjuder en skalbar och kostnadseffektiv väg mot modern avfallshantering. Det modulära angreppssättet möjliggör framtida integration av realtidsdata och stöder proaktiva insamlingsstrategier som minskar miljömässiga och operativa kostnader.

Nyckelord: Avfallsinsamling, Fyllnadsnivåprognoser, Transformer, Sekvens-till- sekvensprognostisering, Multiobjektiv Optimering

Acknowledgments

We would like to express our deepest gratitude to our families for their unconditional support and for always being by our side when we needed comfort and encouragement throughout this journey.

We also extend our sincere thanks to our co-advisor, Henrik from Decerono, for his valuable guidance, support, and for giving us this opportunity to apply our knowledge and grow professionally.

A heartfelt thanks goes to our teachers and professors who have been active and helpful throughout our studies. Your dedication and support have truly enriched our learning experience.

Finally, a big thanks to Gemiuns.

Contents

Chapter 1

Introduction

1.1 Motivation and Context

Urbanization and demographic growth are dramatically increasing the volume and complexity of municipal solid waste (MSW) management [21]. As cities expand, so do the challenges related to waste collection, including operational costs, greenhouse gas emissions, traffic congestion, and citizen satisfaction. Traditional waste collection systems based on fixed routes and static schedules are no longer sufficient to meet the evolving demands of modern urban environments [27].

One of the most critical issues in static scheduling is the mismatch between container fill levels and collection timing. Overfilled bins lead to environmental and sanitary problems, while premature emptying wastes resources. Smart waste management initiatives, such as deploying sensor-equipped bins and dynamic routing systems, have demonstrated potential in addressing these inefficiencies [31]. However, the widespread deployment of physical sensors remains economically unfeasible for many municipalities, particularly in developing regions, due to high installation, maintenance, and communication costs [17].

In parallel, advances in machine learning have opened new opportunities for predicting waste generation dynamics from historical data alone, thereby circumventing the need for pervasive sensors [40]. Likewise, modern routing optimization techniques, especially multi-objective metaheuristics and constraint programming, can significantly reduce travel distances, service times, and environmental impacts [6,32].

Thus, there is an urgent need for integrated, cost-efficient solutions that combine predictive modeling and optimization to make municipal waste collection smarter, greener, and more resilient.

1.2 Problem Statement

There is currently a lack of cost-effective, scalable methods for predicting container-level fill statuses without real-time sensor data, and for optimizing daily waste collection routes while respecting complex operational constraints such as service frequency, vehicle capacities, waste type compatibilities, and service times.

This thesis addresses the problem of designing a modular framework that (i) accurately forecasts daily container fill levels using historical and contextual data, and (ii) optimizes daily collection routes in a multi-objective setting that balances travel distance and service time.

1.3 Scope and Objectives

This thesis focuses on two interconnected but independently validated modules:

- **Waste Fill-Level Prediction Module**: Development of sequence-to-sequence predictive models using LSTM, Bidirectional LSTM (BiLSTM), and Transformer architectures. The models are trained on synthetically generated daily fill-level data that realistically simulates seasonality, holidays, weather impacts, and other contextual variables. No live sensor telemetry is used.

- **Multi-Objective Route Optimization Module**: Construction and evaluation of a customized NSGA-II (Non-dominated Sorting Genetic Algorithm II) for optimizing waste collection routes. Objectives include minimizing total travel distance and cumulative service time. A constraint programming (CP-SAT) model is used to pre-assign stops to feasible collection days based on waste types, vehicle capacities, and operational rules.

The modules are validated independently to ensure robustness and flexibility, allowing future work to integrate real-time data feeds or extend dynamic re-routing capabilities.

Excluded from the scope are the real-time dynamic re-optimization of routes during a working day, integration with live IoT sensor streams, and economic cost benefit analyses for municipalities.

1.4 Defining Research Questions

We pose research questions that guide our methodological choices:

- **RQ1:** How accurately can a Transformer-based sequence-to-sequence model predict daily container fill levels compared to LSTM and BiLSTM architectures, and which model is more robust under varying synthetic waste generation patterns?

- **RQ2:** To what extent can a multi-objective genetic algorithm (NSGA-II) produce efficient waste collection routes minimizing total travel distance and service time for multi-compartment vehicles compared to Ant Colony Optimization (ACO), ACO with 3-Opt local search, and the commercial PTV routing engine?

By formulating these questions, the thesis seeks to evaluate the performance and robustness of the proposed algorithms. The first research question focuses on the predicting capabilities of state-of-the-art sequence-to-sequence models by comparing LSTM-based and Transformer-based architectures, particularly under varied synthetic waste generation scenarios. The second research question examines how effectively a multi-objective genetic algorithm, specifically NSGA-II, can generate optimized waste collection routes that minimize both travel distance and service time for multi-compartment vehicles in comparison to ACO, ACO+3-OPT, and PTV.

1.5 Thesis Contributions

The principal contributions of this thesis are:

- A synthetic dataset generator for container-level waste fill simulation over multi-year horizons, incorporating realistic seasonality, event-based spikes, and operational factors.
- A systematic benchmark of LSTM, BiLSTM, and Transformer models for multi-day ahead container fill-level forecasting, including hyperparameter tuning and feature engineering.
- A customized NSGA-II algorithm capable of multi-objective optimization for waste routing under complex scheduling and vehicle constraints.
- A constraint programming (CP-SAT) model to allocate waste collection stops to valid days, enforcing service frequencies and logistical constraints before route optimization.
- An empirical comparison between NSGA-II, Ant Colony Optimization (ACO), a hybrid version of ACO with 3-Opt local search, and a commercial PTV routing solution across identical datasets and conditions.

1.6 Thesis Structure

The thesis is organized as follows:

- **Chapter 2** reviews background concepts in waste management, predictive modeling, and routing optimization.
- **Chapter 3** surveys related work and identifies gaps addressed in this thesis.
- **Chapter 4** details the methodology for synthetic data generation, predictive modeling, and route optimization.
- **Chapter 5** presents results for model performance and routing outcomes.
- **Chapter 6** discusses key findings, practical implications, and limitations.
- **Chapter 7** concludes with a summary and suggests avenues for future work.

Chapter 2

Background

This chapter provides a comprehensive overview of the theoretical and technical foundations relevant to this thesis. It introduces the key algorithms, machine learning models, and data generation techniques that form the backbone of the proposed sensor-free fill-level forecasting and AI-driven route optimization modules.

2.1 Waste Collection Optimization: Current Practices

Municipal solid waste (MSW) collection is a core operational component of urban waste management systems. Traditionally, municipalities have employed static collection routes and fixed schedules that are planned in advance and remain unchanged over time. Although simple to implement, these static approaches often lead to operational inefficiencies, such as servicing containers that are not yet full or failing to address peak fill times, which in turn results in increased fuel consumption and higher emissions [32].

To address these inefficiencies, modern waste collection systems increasingly incorporate optimization techniques to enhance operational performance. One key approach is the Vehicle Routing Problem (VRP), a classical combinatorial optimization problem aimed at determining the most efficient routes for a fleet of vehicles to service a set of locations while satisfying operational constraints such as vehicle capacity and time windows. Advanced VRP variants such as the Pollution-Routing Problem (PRP) explicitly integrate environmental objectives by considering fuel consumption and emissions in the optimization process [6].

2.2 Route Optimization Techniques

Waste collection routing can be formulated as a variant of the classical Vehicle Routing Problem (VRP), where a fleet of vehicles must service a set of waste collection points efficiently. Given the combinatorial complexity of routing with operational constraints, such as vehicle capacity, time windows, and waste type compatibility, advanced optimization techniques are required to achieve cost-effective and environmentally sustainable solutions.

2.2.1 General Approaches

The basic VRP aims to minimize the total distance traveled or operational cost while ensuring that all service points are visited exactly once. Extensions such as the Capacitated Vehicle Routing Problem (CVRP) incorporate vehicle load constraints, while the Vehicle Routing Problem with Time Windows (VRPTW) adds time-related constraints to ensure that waste containers are collected within permissible service hours [6].

Multi-objective formulations, such as the Pollution-Routing Problem (PRP), integrate environmental criteria (e.g., fuel consumption and emissions) alongside operational objectives. This evolution reflects the increasing importance of sustainability considerations in urban waste management systems [6].

2.2.2 Genetic Algorithms and NSGA-II

Genetic Algorithms (GAs) are population-based metaheuristic search methods inspired by natural selection. They have been successfully applied to various VRP variants by encoding candidate solutions as chromosomes and iteratively evolving them through crossover and mutation operators [32].

The Non-dominated Sorting Genetic Algorithm II (NSGA-II) extends standard GAs to handle multiple conflicting objectives simultaneously by maintaining a diverse set of Pareto-optimal solutions. NSGA-II employs fast non-dominated sorting, crowding distance calculations, and elitism to ensure convergence and diversity in the solution set [11]. This makes it particularly well-suited for balancing travel distance and service time in waste collection routing.

2.2.3 Ant Colony Optimization and Hybrid ACO+3-Opt

Ant Colony Optimization (ACO) is a bio-inspired metaheuristic that models the foraging behavior of ants to solve combinatorial problems. In routing problems, artificial pheromone trails and heuristic visibility guide solution construction towards promising areas of the search space [12].

To enhance the performance of ACO in waste collection routing, local search techniques such as 3-Opt are often integrated. The 3-Opt algorithm systematically removes and reconnects three edges in a route to eliminate suboptimal loops, improving overall route efficiency [14]. This hybrid approach combines ACO's global search capabilities with local optimization to achieve high-quality routing solutions.

2.2.4 PTV Route Optimizer

The PTV Route Optimizer is a commercial software solution that provides advanced routing and scheduling capabilities for logistics operations, including waste collection. It supports a range of operational constraints, including time windows, vehicle types, and service times, and incorporates real-world factors such as traffic conditions. The software's powerful optimization engine enables practitioners to generate practical and efficient routing plans that comply with municipal regulations and service requirements [16].

2.2.5 Constraint Programming with OR-Tools

Constraint Programming (CP) is a powerful paradigm for solving combinatorial optimization problems that involve complex logical and operational constraints. In CP, the problem is modeled by defining decision variables, constraints that restrict variable values, and an objective function to optimize. The solver then explores the solution space systematically, using techniques such as constraint propagation and search heuristics, to find feasible and optimal solutions [36].

The CP model developed for waste collection routing leveraged OR-Tools' capabilities to ensure feasible assignments of waste containers to vehicles, considering factors such as vehicle compartment compatibility, service day requirements, and operational spread over a multi-day planning horizon. By using OR-Tools, the model could systematically enforce constraints while maintaining computational efficiency.

2.3 Artificial Intelligence and Machine Learning

Artificial Intelligence (AI) encompasses a broad set of computational techniques designed to perform tasks that traditionally require human intelligence, such as decision-making, perception, and pattern recognition. Within AI, Machine Learning (ML) plays a pivotal role by enabling systems to automatically learn from data and improve their performance without explicit programming [29].

2.3.1 Machine Learning: Overview and Types

Machine Learning (ML) refers to computational methods that enable systems to automatically learn from data and improve their performance on a given task without being explicitly programmed. ML has become a cornerstone of modern AI, supporting a wide range of applications, from natural language processing to time-series forecasting [15].

ML methods can be broadly classified into three categories: supervised learning, unsupervised learning, and reinforcement learning.

In supervised learning, models are trained using labeled data, where the correct output is known for each input instance. The objective is to learn a mapping function that can accurately predict outputs for new, unseen data. This approach is particularly well-suited for regression and classification tasks, such as forecasting container fill levels based on historical time-series data [19].

Unsupervised learning involves discovering patterns or structures from unlabeled data. Common techniques include clustering and dimensionality reduction. In waste management, unsupervised learning can identify patterns in waste generation dynamics across different neighborhoods, supporting resource allocation and policy planning [30].

Reinforcement learning focuses on learning optimal policies through interactions with an environment. The model (agent) takes actions to maximize cumulative rewards by exploring different states and receiving feedback. Although less commonly applied in waste collection forecasting, reinforcement learning has potential for dynamic routing and real-time decision-making under uncertain conditions [44].

2.3.2 Deep Learning

Deep Learning (DL) is a subset of machine learning that focuses on models with multiple layers of nonlinear processing units, enabling them to learn hierarchical representations of data. DL architectures are particularly effective for modeling complex patterns and relationships, making them well-suited for tasks such as image recognition, natural language processing, and time-series forecasting [25].

One of the key strengths of deep learning models is their ability to automatically extract relevant features from raw input data. This reduces the need for manual feature engineering, which is especially valuable in domains like waste collection forecasting, where domain knowledge may be limited or data may be noisy and high-dimensional. For example, deep learning methods can effectively capture temporal dependencies in fill-level time series without requiring handcrafted seasonal adjustments or explicit trend modeling [15].

Training deep neural networks typically requires large datasets and significant computational resources. Advances in hardware accelerators (e.g., GPUs) and software frameworks (e.g., TensorFlow, PyTorch) have made deep learning more accessible to researchers and practitioners in waste management and other applied fields [1].

In the context of this study, deep learning models were selected for their capacity to model non-linear, temporal patterns in container fill levels.

2.3.2.1 LSTM and BiLSTM Networks

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network (RNN) specifically designed to capture long-range dependencies in sequential data. Unlike traditional RNNs, which suffer from the vanishing gradient problem that hinders learning of long-term dependencies, LSTMs incorporate memory cells and gating mechanisms that enable them to selectively retain information over extended time periods [20].

LSTM networks are particularly well-suited for time-series forecasting tasks, such as predicting container fill levels, where dependencies between waste generation events can span multiple days or weeks. The cell state, controlled by input, forget, and output gates, allows the model to manage information flow and effectively capture complex temporal patterns [20].

Bidirectional LSTM (BiLSTM) networks extend the standard LSTM architecture by processing the input sequence in both forward and backward directions. This bidirectional structure allows the model to incorporate context from both past and future time steps, enhancing its ability to model dependencies that are not strictly causal in time. Such architectures have demonstrated improved performance in various sequential learning tasks, including natural language processing and time-series forecasting [39].

In the context of waste collection forecasting, BiLSTM networks can leverage information from both earlier and later parts of the sequence to more accurately predict container fill levels, especially in datasets where temporal dependencies are complex and potentially non-linear.

2.3.2.2 Transformer Models

Transformer models represent a significant advancement in sequence modeling, particularly due to their use of self-attention mechanisms that enable them to capture dependencies between any two positions in a sequence, regardless of distance [45]. This architecture diverges from traditional recurrent structures by processing sequences in parallel, thereby improving computational efficiency and enabling the modeling of long-range dependencies.

The core component of the Transformer is the self-attention mechanism, which computes a weighted representation of each position in a sequence based on its relationship to all other positions. This allows the model to dynamically focus on relevant parts of the input when making predictions, a feature especially valuable in complex time-series forecasting where important patterns may span across irregular intervals [26].

Originally developed for natural language processing, Transformer architectures have been successfully adapted for time-series applications, including demand forecasting, anomaly detection, and sensor-based predictive maintenance [48]. In the context of waste management, Transformers can learn complex temporal patterns in container fill levels, capturing both short-term fluctuations and long-term trends without the need for explicit sequential modeling found in RNN-based architectures.

2.3.2.3 Sequence-to-Sequence Models

Sequence-to-Sequence (Seq2Seq) models are a deep learning architecture originally developed for tasks such as machine translation but have since been widely adopted for time-series forecasting. A Seq2Seq model consists of two main components: an encoder and a decoder. The encoder processes the input sequence (e.g., historical container fill levels) and encodes it into a context vector summarizing the sequence information. The decoder then uses this context vector to generate the output sequence (e.g., forecasted fill levels) one time step at a time [43].

In this study, Seq2Seq frameworks were implemented using both LSTM and Transformer architectures. Applying Seq2Seq modeling to LSTM networks allows the system to capture long-term dependencies in waste generation patterns while managing variable-length input and output sequences. For Transformer-based models, Seq2Seq design enhances the model's capacity to handle temporal dynamics by leveraging self-attention mechanisms across both encoder and decoder stages [45]. Integrating Seq2Seq structures enables the models to better capture the sequential dependencies between historical and forecasted fill levels, ultimately improving the accuracy and interpretability of the predictions [28].

2.3.3 Evaluation Metrics for Machine Learning Models

In regression tasks, two of the most commonly used evaluation metrics are Mean Absolute Error (MAE) and Mean Squared Error (MSE).

Mean Absolute Error (MAE) measures the average absolute difference between predicted and observed values. It provides an interpretable measure of overall model performance, expressed in the same units as the target variable. MAE is particularly

useful in waste generation applications because it conveys the typical daily deviation between predicted and actual fill levels without excessively penalizing outliers [46].

Mean Squared Error (MSE), on the other hand, calculates the mean of the squared differences between predicted and observed values. This metric places a heavier penalty on larger errors, making it sensitive to outliers. MSE is often preferred in contexts where larger prediction errors are considered more detrimental to operational performance, as it emphasizes minimizing the impact of high-error days in waste generation forecasting [10].

Understanding these concepts provides a foundation for appreciating the methodology, analysis, and conclusions presented in the subsequent chapters.

Chapter 3

Related Work

Recent years have witnessed rapid progress in both predictive analytics and route-optimization for municipal solid-waste (MSW) management [7]. In this chapter, we review these topics in turn and close with a structured gap analysis that motivates the two-module framework developed in Chapter 4. Throughout, each study is explicitly linked to the present thesis.

3.1 Waste Level Predicting

Reliable short-horizon predictions of individual waste container fill levels are essential for any data-driven waste collection strategy. The literature has evolved from simple sensor-triggered heuristics to deep neural architectures that capture seasonality, covariate effects, and long-range dependencies.

3.1.1 Sensor driven statistical baselines

Tran Anh Khoa et al. [23] deployed LoRa-equipped bins on a Vietnamese university campus and trained a logistic-regression classifier that flagged "empty-soon" events. Real-time alerts reduced mileage and overflowing incidents by 28% and 43%, respectively, but hardware, maintenance, and connectivity costs limit scalability. Rutqvist et al. [37] addressed noisy ultrasonic data from glass-recycling containers: an AutoML search chose a random-forest model that lifted F_1 score from 0.62 to 0.86 and halved false positives. Both systems require continuous telemetry; our project deliberately explores sensor-free prediction, relying instead on synthetic historical series, to keep capital expenditure negligible for resource-constrained municipalities see section 4.5.

3.1.2 Recurrent and hybrid approaches

Jamelli et al. [22] benchmarked multiple single-step architectures such as Linear Regression, LSTM, and Bidirectional LSTM on daily volumes from Sousse, Tunisia. BiLSTM captured intra-annual seasonality and cut RMSE by 17% versus unidirectional LSTM. Motivated by those gains, we include a BiLSTM many-to-one baseline without an encoder–decoder loop against which our seq2seq models are compared. Ghanbari et al. [13] combined ensemble empirical-mode decomposition with MARS splines; their hybrid returned $R^2 = 0.82$ and NSE $= 0.67$ on Tehran's monthly tonnage while Monte-Carlo simulations quantified parameter uncertainty. Adamović et

al. [2] injected structural-break indicators (e.g. 2008 financial crisis) into a GRNN, slashing MAPE from 6.7% to 4.0% across 44 countries. These works confirm the value of exogenous signals and hybridisation ideas we exploit through engineered calendar/holiday covariates and location embeddings.

3.1.3 Attention-based Transformer models

The introduction of self-attention has opened a new line of research in waste-level Predicting because it removes the recurrence bottleneck and natively captures very long temporal or cross-series dependencies. Shi *et al.* [41] were among the first to apply a pure Transformer encoder–decoder to waste data: their three-layer network fuses meteorological and industrial production indices through a gated cross-attention block and is trained on Shanghai's hazardous waste records. The model reduces MAE by 23% relative to an LSTM baseline while the learned attention maps expose the pronounced weekday production cycles that drive hazardous waste peaks insight that classical RNNs tended to hide. Zhou *et al.* [49] broaden the scope from short-term operations to long-term planning by coupling a Transformer–LSTM hybrid (WEIO-TL) with an economic input–output framework. Their architecture, evaluated on Chinese provincial statistics from 2010-2020, attains $R^2 = 0.91$ and, crucially, quantifies sector-level waste elasticities, enabling scenario analysis up to 2035. A different angle is taken by Kumar *et al.*, whose GraphFormer model [24] augments the attention kernel with a fixed spatial graph so that information is shared preferentially between economically or demographically similar cities. Tested on daily municipal-solid-waste streams from 25 prefectures, GraphFormer lowers RMSE by 15% against the Temporal-Fusion Transformer yet trains 40% faster thanks to its lightweight architecture.

These studies demonstrate that attention mechanisms can outperform recurrent networks on heterogeneous, long-range waste data and that they offer valuable interpretability. However, they all Predict aggregated tonnage at weekly or monthly resolution, hazardous waste for Shanghai, total MSW for provincial or prefecture scales, and therefore stop short of the daily, container fill-level granularity required for routing. Our work closes that gap by benchmarking a seq2seq Transformer directly against LSTM and BiLSTM variants on synthetic waste container fill-level time series that mimic operational fill dynamics.

3.2 Route Optimization

Designing cost-efficient routes for selective waste collection is typically cast as a Vehicle Routing Problem with Time Windows (VRPTW). Beliën and De Boeck's review of 260 papers already noted two enduring gaps: most studies focus on single-fraction waste and rarely consider joint trade-offs between cost, service time, and emissions [8]. Nearly a decade later, those gaps are narrowing through advances in multi-objective meta heuristics, bio inspired dynamic routing, and constraint programming solvers.

3.2.1 Multi-objective Metaheuristics

Ombuki *et al.* introduced a Pareto-based Genetic Algorithm for VRPTW that simultaneously shortened routes and reduced lateness [33]. Rachmawati *et al.* addressed a Multi-Trip Multi-Period Capacitated VRPTW (MCVRPTW); their customised GA lowered total transportation cost by 30.2% and cut solve time from hours to minutes against a MILP benchmark [35]. Gülmez *et al.* demonstrated that NSGA-II/III and MOEA/D, applied to a green VRPTW with flexible customer time windows, trimmed fuel by 14% and CO_2 by 11% [18]. Bouleft and Elhilali Alaoui introduced a rolling-horizon NSGA-II for a Dynamic Multi-Compartment VRP, achieving an 18% distance reduction while preserving recyclable segregation [9]. Sar and Ghadimi embedded NSGA-II and SPEA-2 in a web service for medical-waste routing; on realistic Dublin data, NSGA-II produced higher quality Pareto fronts in half the CPU time of SPEA-2 [38]. These findings confirm NSGA-II's practicality for waste logistics problems a property we exploit by adapting it to multi-compartment vehicles and by adding explicit service time costs.

3.2.2 Monitoring convergence via the hypervolume indicator

Because multi-objective searches may stall on local fronts, several authors recommend the hypervolume indicator introduced by Zitzler and Thiele as the "size of the dominated space" to measure both convergence and diversity in real time [50]. Hypervolume is strictly monotonic with Pareto dominance; consequently, a rise in the indicator guarantees an improvement in at least one objective. Following HypE [5], we track the relative hypervolume change ΔHV across generations and terminate NSGA-II once $\Delta HV < \varepsilon$ for a fixed window, balancing run-time with solution quality.

3.2.3 Bio-inspired heuristics for dynamic routing

Wu *et al.* combined Particle Swarm optimization and Simulated Annealing in a Local-Search Hybrid Algorithm; when fed with live bin-level data, environmental impact fell by 42.3% relative to static plans [47]. Alwabli and Kostanic showed that a pure Ant Colony Optimization (ACO) variant can re-plan 52-stop routes within seconds, and that a GA with Hamiltonian-path constraints saved a further 6% fuel on the same benchmark [3, 4]. Inspired by such results, we retain ACO as a fast baseline and enhance it with a 3-Opt local search to remove residual detours.

3.2.4 Constraint-programming

Exact and hybrid solvers based on constraint programming have begun to rival metaheuristics for medium-sized instances. The CP-SAT engine in Google OR-Tools blends constraint-propagation, SAT clause learning and Mixed-Integer Programming (MIP) cuts; it has won multiple MiniZinc Challenge gold medals and often matches commercial MIP solvers on binary models. In waste-collection studies, CP-SAT has been used to generate frequency-locked daily schedules before routing is applied, outperforming Mixed-Integer Linear Programming (MILP) on instances with hundreds

of stops and tight time windows [34]. Our methodology mirrors this workflow: a CP-SAT model assigns stops to days while enforcing visit frequencies, waste-type compatibilities, and vehicle-capacity limits; the resulting daily stop sets are then fed into NSGA-II, ACO to optimize the routes for each day.

Across these topics, the literature validates evolutionary algorithms, bio-inspired search, and modern CP-SAT solvers as competitive options. Nonetheless, evidence for multi-objective optimization with multi-compartment vehicles and realistic service times remains sparse. Our study fills that niche by benchmarking NSGA-II and ACO together with a commercial baseline—on precisely that problem class.

3.3 Summary and Research Gaps

The literature review has highlighted three unresolved issues. First, existing Predicting systems either instrument every waste container with a sensor or predict at coarse spatial–temporal scales; there is still no evidence that day-ahead, container-level predictions can be produced from historical records alone. Second, while recent NSGA-II variants optimise multiple objectives, they are almost always evaluated on single-fraction waste, often ignore explicit service times, and are rarely integrated with a constraint-programming pre-processor that honours practical scheduling rules.

This thesis addresses those gaps in two steps. *(i) predicting.* On a synthetic dataset we benchmark sequence2sequence Transformers, LSTM and a non-seq2seq BiLSTM. Training minimises mean-squared error, while validation uses mean-absolute error; The predictive task is evaluated in isolation—no prediction is fed into the routing engine in this study. *(ii) Routing.* Independently, a customised multi-objective NSGA-II optimises fuel cost and worker time (travel distance + service time) for multi-compartment vehicles. Google OR-Tools CP-SAT is used once, upstream, to assign 2 791 waste containers grouped into 352 collection stops over a preset 20-day horizon, satisfying frequency, capacity, and waste-type compatibility constraints. Convergence of NSGA-II is monitored exclusively with the hypervolume indicator, guaranteeing strict Pareto improvement, and the resulting fronts are compared with those from an ACO + 3-Opt baseline and a commercial PTV solver on identical instances.

Together, these contributions offer a cost-neutral pathway toward data-driven collection: realistic demand curves can be simulated while historical logs accumulate, state-of-the-art models can predict fill levels without sensors, and an optimiser that respects both labour time and multi-compartment constraints. Although predicting and routing are evaluated separately here, their interfaces are deliberately lightweight so that future work can couple real driver-reported fill-levels with dynamic re-routing.

Chapter 4

Method

The primary aim of this thesis is to develop independent modules for predicting waste container fill levels and optimizing the routes of multi-compartment waste collection vehicles. To achieve this, the research employs a synergistic blend of data synthesis, predictive modeling, and routing optimization techniques. These components are developed as separate entities to establish robust foundations for a future integrated system.

In summary, the approach comprises:

- **Waste Level predicting:** Utilizing sequence-to-sequence models (LSTM and Transformer) with different scaling schemes, including the novel use of Bidirectional LSTM to capture intricate temporal dependencies.
- **Route Optimization:** The central focus of this research is the design of a multi-objective genetic algorithm, specifically NSGA-II, to generate optimized waste collection routes. After developing the NSGA-II algorithm, its performance will be systematically compared with other leading optimization tools, including the industry-standard PTV solver and heuristic methods like Ant Colony Optimization (ACO). In addition, a hybrid approach that integrates ACO with a 3-Opt local search will be evaluated. This comparative analysis aims to benchmark each method's effectiveness in reducing travel distance and service time.

Together, these components can form the backbone of a system that not only predicts container fill levels accurately but also utilizes these predictions to optimize waste collection routes for reduced travel distance and service time.

4.1 Describing the Research Method

The overall methodology can be divided into three major steps:

Dataset & Synthetic Data Generation

We began with a comprehensive real-world dataset detailing locations, waste types, the number of waste containers, and monthly visit frequencies. However, due to the unavailability of real-world container fill-level data, it was necessary to generate synthetic time series data to support the development and evaluation of our predictive models.

The synthetic data was constructed using statistical patterns derived from the original dataset, allowing us to mimic the realistic behavior of waste generation at each location. This approach ensured that temporal patterns, weekly seasonality, container capacities, and location-specific characteristics were preserved as closely as possible. As a result, the synthetic dataset provided a reliable foundation for training, validating, and testing our sequence-to-sequence predictive models while maintaining a close resemblance to real operational scenarios.

Time-Series Predictive Modeling

Multiple neural network variants are developed specifically, including LSTM-based and Transformer models under two scaling schemes (Standard vs. Min-Max). The models are designed to predict seven days of future waste levels based on fourteen days of historical input data.

Multi-Objective Route Optimization

The NSGA-II algorithm serves as the primary method for generating routing solutions that minimize both travel distance and service time. Once developed, NSGA-II performance is systematically compared against alternative optimization approaches, including the PTV optimizer, Ant Colony Optimization (ACO), and a hybrid method that integrates ACO with a 3-Opt local search. This comparison provides a comprehensive benchmark to assess the effectiveness and efficiency of the proposed NSGA-II algorithm relative to other state-of-the-art methods.

4.2 Dataset Description

This study is grounded in a real-world dataset provided by the IT consultancy Decerno, sourced from Renall AB, a waste collection operator active in the city of Örebro, Sweden. The dataset comprises 2,791 records, each representing a container installed at a distinct waste collection point across a total of 354 different locations throughout the city.

As part of the data preprocessing phase, a feature selection process was conducted to identify the most relevant attributes for the routing and geospatial analysis tasks. The selected features, *Fraktion* (denoting the waste type), *Tömningsintervall* (the scheduled emptying interval, e.g., weekly, biweekly, or monthly), and spatial coordinates (*XKO2* and *YKO2*) were retained as they directly influence route planning and spatial reasoning. Other attributes present in the original dataset were excluded because they were either redundant, irrelevant to the routing problem, or introduced unnecessary complexity.

For the purposes of route optimization, all waste types were retained, namely: *Pappersförpackningar* (paper packaging), *Plastförpackningar* (plastic packaging), *Tidningar* (newspapers), *Ofärgat glas* (non-colored glass), *Färgat glas* (colored glass), and *Metallförpackningar* (metal packaging). In parallel, a synthetic dataset was constructed for the development of machine learning models. This synthetic dataset,

Table 4.1: Preview of dataset attributes

Fraktion	Tömningsintervall	XKO2	YKO2
Färgade glasförpackningar	Varannan vecka	59,248191057799	15,167993567884
Metallförpackningar	Varannan vecka	59,248191057799	15,167993567884
Ofärgade glasförpackningar	Varannan vecka	59,248191057799	15,167993567884
Pappersförpackningar	Varje vecka	59,248191057799	15,167993567884
Pappersförpackningar	Varje vecka	59,248191057799	15,167993567884

built from a subset of the original, focused solely on paper, plastic, and newspaper fractions, and was used exclusively in experimentation unrelated to the routing task.

4.3 Geospatial Distance Time Calculation

To model travel between stops, spatial information from the dataset was used to compute distance and travel time matrices between all collection points. These matrices were generated using the Google Maps Distance Matrix API, selected over open-source alternatives such as OpenStreetMap due to its more accurate, traffic-aware estimations within urban contexts like Örebro. The resulting travel matrices served as the geospatial foundation for the routing model, enabling the optimization algorithm to operate on realistic road distances and travel times.

Table 4.2: Illustrative Distance Matrix (in kilometers)

From / To	Stop A	Stop B	Stop C	Stop D
Stop A	0.0	1.2	2.5	3.1
Stop B	1.2	0.0	1.7	2.9
Stop C	2.5	1.7	0.0	1.1
Stop D	3.1	2.9	1.1	0.0

This is illustrative data. Distances were computed using road network data from Google Maps Distance Matrix API and do not represent actual measurements.

Table 4.3: Illustrative Travel Time Matrix (in minutes)

From / To	Stop A	Stop B	Stop C	Stop D
Stop A	0	4	7	9
Stop B	4	0	5	8
Stop C	7	5	0	3
Stop D	9	8	3	0

This is illustrative data. Travel times reflect simulated urban traffic conditions in Örebro as estimated by the Google Maps Distance Matrix API. Actual values differ.

In addition to travel time, service time was also considered and calculated for each location to improve the reliability of the total time estimation for each route.

The service time was computed based on the number of containers present at each collection point, including the depot and the dump site. By incorporating service time, we ensured that all relevant cost components were captured, both the travel time between stops and the operational time spent at each location. The service time factor per container was provided by Decerno based on domain expertise, while container counts were extracted directly from the dataset.

4.4 Constraint Programming Model

The routing strategy was formulated as a constraint programming model implemented using the CP-SAT solver within Google OR-Tools. The scheduling horizon covered four weeks, with five working days per week (Monday to Friday), resulting in a total of:

$$\text{TOTAL\_DAYS} = 4 \times 5 = 20 \text{ days.}$$

Two waste category groupings were defined to align with operational constraints: PAPER_PLAST_TIDNING, which includes paper, plastic, and newspaper waste, and GLASF_GLASOF_METALL, comprising colored glass, non-colored glass, and metal. These categories are mutually exclusive on a per-day basis: due to vehicle limitations, a waste collection vehicle can only handle one of these two categories on any given day. Consequently, paper-related waste and glass/metal waste are never collected on the same day by the same vehicle.

Each stop was associated with a collection frequency, interpreted in the 20-day horizon using the following mapping:

- 4 ggr/vecka $\rightarrow$ 16 collections,
- 3 ggr/vecka $\rightarrow$ 12 collections,
- 2 ggr/vecka $\rightarrow$ 8 collections,
- varje vecka $\rightarrow$ 4 collections,
- varannan vecka $\rightarrow$ 2 collections,
- var 4:e / var 8:e vecka $\rightarrow$ 1 collection.

To enforce logical spacing between visits, especially for high-frequency stops, predefined locking patterns were used. Stops with 8 visits were assigned to Mondays and Fridays; those with 12 visits to Mondays, Wednesdays, and Fridays; and those with 16 visits to Mondays, Tuesdays, Thursdays, and Fridays. This locking ensures sufficient time between visits for container filling and prevents over-clustering of collection tasks.

Let $X_{i,d} \in \{0, 1\}$ be a binary decision variable that is equal to 1 if stop i is assigned to day d, and 0 otherwise. Separate sets of such variables were maintained for the two waste categories. Each stop i must be scheduled exactly as many times as indicated by its frequency f_i, leading to the constraint:

$$\sum_{d=0}^{19} X_{i,d} = f_i \quad \forall i.$$

For stops with biweekly frequency ($f_i = 2$), an additional spacing constraint ensures one visit occurs in the first half of the horizon (weeks 0–1) and one in the second half (weeks 2–3). Two mutually exclusive scenarios were defined:

- Scenario A: scheduling in Week 0 and Week 2,
- Scenario B: scheduling in Week 1 and Week 3.

Day-specific service rules were also incorporated. Glass and metal waste could only be collected on Tuesdays and Thursdays, corresponding to weekdays where $d \bmod 5 \in \{1, 3\}$. In contrast, paper-related waste was allowed on all weekdays.

Vehicle capacity constraints were enforced to reflect real operational limits. The number and type of available vehicles varied by day:

- Monday, Wednesday, Friday: two vehicles for paper waste (each with a capacity of 130 stops), none for glass.
- Tuesday, Thursday: one vehicle each for paper and glass waste (capacity of 65 stops each).

Let C_d denote the total daily capacity, and locked_d the number of frequency-locked assignments on day d. The constraint on daily capacity can then be expressed as:

$$\sum_{i \in \text{movable paper}} X_{i,d} + \text{locked}_d \leq C_d.$$

To further improve operational efficiency, the model incorporated an objective function that promotes the geographical compactness of routes. Auxiliary binary variables $Y_{i,j,d} \in \{0, 1\}$ were introduced to indicate whether both stops i and j are scheduled on the same day d. The model minimizes the total pairwise distance between co-assigned stops as follows:

$$\min \sum_{i<j,d} \text{distance}(i, j) \cdot Y_{i,j,d},$$

subject to:

$$Y_{i,j,d} \geq X_{i,d} + X_{j,d} - 1,$$
$$Y_{i,j,d} \leq X_{i,d}, \quad Y_{i,j,d} \leq X_{j,d}.$$

This clustering objective encourages the scheduling of geographically proximal stops on the same day, reducing route spread and likely leading to shorter, more efficient vehicle tours.

The final output of the solver is a structured schedule that assigns stops to specific days in compliance with all constraints. These assignments are grouped by waste type and exported as CSV files, one per day and per waste category, thereby providing input for the subsequent phase of detailed route optimization.

4.5 Synthetic Data Generation

This section presents the methodology used to generate synthetic data simulating waste container fill levels for multiple waste types: paper, plastic, and newspapers. The synthetic dataset spans from January 2021 to December 2024, providing four full years of data to capture both annual trends and seasonal patterns. This methodology integrates deterministic modeling with stochastic variability to replicate realistic waste generation behaviors observed across different environmental and societal conditions.

To enhance realism, frequency data were employed to account for variations in waste collection frequency and container count per location. Several contextual factors were incorporated into the simulation using a multiplicative model structure. The waste level for each day, for a given waste type, is defined mathematically as:

$$W_t = \max\left(0,\, W_{t-1} \times F_t + \epsilon_t + \delta_t\right)$$

Where:

- W_t: Waste level at day t
- W_{t-1}: Waste level at the previous day
- F_t: Composite multiplicative factor reflecting contextual impacts
- ϵ_t: Random noise (uniformly distributed)
- δ_t: Randomized adjustment representing casual daily waste variability

The composite factor F_t is computed as follows:

$$F_t = H_t \cdot P_t \cdot S_t \cdot Y_t \cdot C_t \cdot R_t \cdot W_t$$

Each component of F_t corresponds to a specific real-world factor:

- **Holiday Impact (H_t)**: A multiplier applied on public holidays, generally $H_t = 1.1$ to reflect increased waste due to elevated consumption.
- **Payday Effect (P_t)**: A factor $P_t = 1.05$ applied on days following paydays.
- **Seasonal and Annual Variations (S_t, Y_t)**: $S_t = 1.08$ for non-summer months, $S_t = 1$ for June to August; $Y_t = 1.06$ for 2022–2024.
- **Container and Frequency Factors (C_t)**: $C_t = \left(\frac{n}{1000} + 1\right) \cdot \left(\frac{f}{100} + 1\right)$ where n is the number of containers and f the collection frequency.
- **Newspaper Readership Adjustment (R_t)**: Applied only to newspapers, where $R_t < 1$ for more recent years.
- **Weather Influence (W_t)**: $W_t = 0.98$ on rainy days (>5mm), and $W_t = 1.1$ on dry days.

The stochastic components are defined as:

$$\epsilon_t \sim U[-5, 5] \quad \text{and} \quad \delta_t \sim U[a, b]$$

where $[a, b]$ depends on the collection frequency.

The specific multipliers used for contextual factors were selected through a trial-and-error approach. After implementing the synthetic data generation, we manually adjusted each factor incrementally and evaluated the resulting patterns through both quantitative indicators (e.g., seasonal strength, holiday spike percentage) and visual inspection via STL decomposition plots. The chosen values were those that consistently produced time series reflecting the qualitative patterns in real-world waste behavior, such as noticeable but non-dominant holiday spikes, increased winter waste volumes, and gradual annual growth. This iterative tuning process ensured that each multiplier had a realistic and interpretable impact on the simulated data.

4.5.1 Synthetic Data Modeling Workflow

1. **Date Range Simulation**: A continuous date range is created for daily-level modeling.
2. **Location-based Iteration**: Each location's daily waste levels are modeled separately per waste type.
3. **Waste Level Calculation**: Starting from base values, waste levels evolve daily according to the defined equation.
4. **Threshold Handling**: When a full threshold (e.g., 85 units) is exceeded on a weekday, the container is emptied.
5. **Data Aggregation**: Outputs include daily waste levels and emptying flags, ready for training/testing.

4.6 Data Analysis

Structured time series analysis was conducted to verify the modeled patterns in the synthetic data. A custom Streamlit dashboard was used for interactive exploration [42].

1. **Time Series Preprocessing**: Time series were resampled to daily frequency and interpolated.
2. **Trend Detection**: Linear regression estimated the slope β:

   $$\text{WasteLevel}_t = \beta t + \epsilon_t$$

   with annualized trend $\beta \times 365$.
3. **Seasonality Strength Estimation**: Using STL decomposition:

   $$\text{Seasonality Strength} = 1 - \frac{\text{Var(Residual)}}{\text{Var(Original Series)}}$$

4. **Holiday Spike Detection**:

$$\text{Holiday Spike (\%)} = \frac{\text{Mean}_{\text{Dec, Jan}} - \text{Mean}_{\text{Annual}}}{\text{Mean}_{\text{Annual}}}$$

5. **Summer Dip Quantification**:

$$\text{Summer Dip (\%)} = \frac{\text{Mean}_{\text{Annual}} - \text{Mean}_{\text{June–Aug}}}{\text{Mean}_{\text{Annual}}}$$

6. **Visual Inspection**: STL plots, rolling averages, and breakdowns confirmed model behavior.

7. **Aggregated Verification**: Confirmed global trends aligned with simulation design.

4.7 Data Preprocessing and Sequence Generation

Before model training, data were processed via feature engineering, location encoding, normalization, and sequence generation.

Feature Engineering: Calendar-based features were extracted and encoded cyclically to capture seasonal patterns, e.g.,

$$\text{sin\_day} = \sin\left(2\pi \cdot \frac{\text{day of year}}{365}\right), \quad \text{cos\_day} = \cos\left(2\pi \cdot \frac{\text{day of year}}{365}\right)$$

Normalization and Scaling: Two normalization strategies are used depending on the model architecture:

- **Min-Max Scaling**: Rescales each feature x to a fixed interval $[0, 1]$ based on its minimum and maximum observed values:

$$x_{\text{scaled}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

  This approach preserves the original distribution shape while normalizing feature magnitudes.

- **Standard Scaling**: Transforms each feature x to have zero mean and unit variance, which improves convergence stability, especially when input features vary in scale:

$$x_{\text{scaled}} = \frac{x - \mu}{\sigma}$$

  where μ is the mean and σ is the standard deviation of the feature across the training set.

The following features are normalized:

- Waste level signals: `Waste_Level_Paper`, `Waste_Level_Plastic`, `Waste_Level_Newspaper`
- Operational flag: `Empty`

- Temporal encodings: `Day_of_year_sin`, `Day_of_year_cos`, `Day_of_week`, `Month`, and `Year`

All predictions are post-processed using the inverse of the corresponding transformation to recover interpretable real-world waste levels.

Sequence Generation: The time series is transformed into supervised learning samples using sliding windows:

- A 14-day encoder input window
- A 7-day decoder prediction window

Each sequence is created per location, where input windows contain normalized and encoded features, and target windows contain the actual future waste levels.

Train–Validation–Test Split: The dataset is chronologically divided into:

- **Training data:** January 2022 to December 2023 (2 years)
- **Validation data:** July 2024 to December 2024 (6 months)
- **Test data:** January 2024 to June 2024 (6 months)

This split ensures that the models are trained on past data and evaluated on future unseen data to reflect realistic prediction scenarios.

4.8 Transformer-Based Sequence-to-Sequence Architecture

The Transformer-based sequence-to-sequence (seq2seq) model employed in this study adopts a standard encoder-decoder architecture specifically tailored for multivariate time series prediction. This architecture maps historical sequences of input features into a compressed latent representation, which is subsequently decoded into future waste level predictions. The core advantage of this structure lies in its capacity to model long-range temporal dependencies and non-linear interactions through attention mechanisms, without relying on recurrent operations.

4.8.1 Input and Embedding Structure

Each training sample is composed of two windows:

- An encoder input window of 14 time steps
- A decoder input window of 7 time steps

Let $X_{\text{enc}} \in \mathbb{R}^{14 \times d}$ represent the encoder input matrix, where d is the number of input features after preprocessing (e.g., normalized waste levels, sine/cosine time encodings). Each location $\ell \in \mathcal{L}$ is associated with a learnable embedding vector $e_\ell \in \mathbb{R}^h$, where h is the dimensionality of the latent space. This vector is broadcast across the sequence length and concatenated with the input features:

$$\tilde{X}_{\text{enc}} = [X_{\text{enc}} \mid \text{tile}(e_\ell)] \in \mathbb{R}^{14\times(d+h)}$$

A similar transformation is applied to the decoder input sequence, ensuring that the model has access to both temporal and spatial contextual information.

4.8.2 Encoder Module

The concatenated encoder inputs are passed through a dense projection layer, mapping the augmented features into a shared latent space of dimension h. The projected sequence is then fed through a stack of Transformer encoder blocks. Each block includes:

- **Multi-Head Self-Attention:** For a given time step t, the attention layer computes attention weights $\alpha_{t,s}$ across all other positions $s \in [1, 14]$, enabling the model to capture both short- and long-range temporal dependencies. For H attention heads, the outputs are concatenated and projected to maintain dimensional consistency.
- **Residual Connections and Layer Normalization:** Residual connections are applied after both the self-attention and feed-forward components to promote stable gradient flow. These are followed by layer normalization to enhance training stability.
- **Position-wise Feed-Forward Network (FFN):** The FFN consists of two dense layers:
  $$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2$$
  followed by a dropout and another layer normalization step.

Each encoder block outputs a transformed sequence $Z_{\text{enc}} \in \mathbb{R}^{14\times h}$, which encodes the temporal and contextual dynamics of the historical data.

4.8.3 Decoder Module

The decoder receives a projected version of the 7-step future context $X_{\text{dec}} \in \mathbb{R}^{7\times d}$, similarly augmented with location embeddings. It undergoes the following steps:

- **Masked Multi-Head Self-Attention:** Ensures that each future time step only attends to previous time steps within the decoder sequence to preserve autoregressive integrity.
- **Cross-Attention with Encoder Outputs:** A key step in seq2seq architectures, this layer computes attention between the decoder representation and encoder output Z_{enc}, enabling the decoder to contextualize each prediction using the full encoded history.
- **Feed-Forward and Residual Layers:** Same structure as in the encoder, to deepen the representational capacity of the decoder.

The final decoder output $Z_{\text{dec}} \in \mathbb{R}^{7\times h}$ is passed through a final linear layer to produce the predicted waste levels:

$$\hat{Y} \in \mathbb{R}^{7\times 3}$$

where each row corresponds to predicted values for paper, plastic, and newspaper waste, respectively.

4.9 LSTM-Based Sequence-to-Sequence Model

The LSTM-based sequence-to-sequence (seq2seq) model provides a classical yet effective architecture for time series predicting by leveraging Long Short-Term Memory (LSTM) cells. These cells are capable of modeling long-range temporal dependencies and nonlinear trends through their internal gating mechanisms. The model is built around an encoder-decoder architecture, wherein a historical window of observations is encoded into a latent state, which is then decoded into a prediction horizon.

4.9.1 Input and Embedding Structure

The LSTM model adopts the same input format and location-based embedding strategy as the Transformer-based model. Each training instance consists of a 14-day encoder input sequence and a 7-day decoder input sequence, with input features concatenated with a broadcasted location embedding vector to provide spatial context. This uniform structure allows for consistent preprocessing and facilitates fair comparison between the two architectures.

4.9.2 Encoder and Decoder Modules

The encoder processes the 14-day sequence through a single-layer LSTM, producing a series of hidden states and passing the final hidden and cell states to the decoder. The decoder receives the 7-day input sequence and is initialized with the encoder's final states. It generates the output sequence $H_{\text{dec}} \in \mathbb{R}^{7\times u}$, which is transformed into predictions through a dense layer:

$$\hat{Y} = H_{\text{dec}} W_{\text{out}} + b_{\text{out}} \in \mathbb{R}^{7\times 3}$$

4.10 Hyperparameter Tuning

Hyperparameters for both models are optimized using a Random Search strategy implemented via the Keras Tuner framework. This approach efficiently explores the parameter space and selects configurations that yield the best validation performance.

The hyperparameter search includes:

- Number of encoder and decoder layers (for Transformer)
- Latent dimensionality h (Transformer) and u (LSTM)

- Number of attention heads (Transformer-specific)
- Feed-forward network size
- Dropout rates for regularization
- Batch size and learning rate

Both the Transformer and LSTM models are trained using Mean Squared Error (MSE) as the loss function and evaluated using Mean Absolute Error (MAE) to assess predictive performance across multiple waste categories and locations.

4.11 Bidirectional LSTM Model with Convolutional Feature Extraction

As part of the architectural benchmarking process, an alternative predicting model was developed to evaluate the comparative performance of non-seq2seq architectures on the waste level prediction task. This model combines convolutional preprocessing with a stacked Bidirectional LSTM (BiLSTM) network, offering a different approach to temporal feature modeling without relying on an explicit encoder-decoder structure.

This configuration allows for direct many-to-one modeling, where a fixed-length input sequence is mapped to a fixed-length prediction vector, bypassing the step-by-step decoding process used in sequence-to-sequence models.

4.11.1 Model Objective and Setup

The primary objective of this model is to assess whether a more streamlined, non-autoregressive architecture can achieve comparable performance to complex seq2seq frameworks. The model takes a 14-day historical window as input and directly outputs a 7-day prediction, with three predicted waste levels per day.

Let the input tensor be:

$$X \in \mathbb{R}^{14 \times d}$$

where $d = 8$ corresponds to the number of normalized features per time step.

4.11.2 Convolutional Feature Extraction

Before entering the recurrent layers, the input is processed through two 1D convolutional blocks, each followed by batch normalization and activation. The first block also includes max pooling to reduce temporal dimensionality. These convolutional layers serve two primary purposes:

- Denoising the signal by filtering out short-term irregularities.
- Extracting local patterns (e.g., weekly spikes, periodic changes) that are often difficult for RNNs to learn directly.

The convolutional output, X_{conv}, captures enhanced local temporal structure and is passed to the BiLSTM layers for global pattern modeling.

4.11.3 Bidirectional LSTM Stack

The model includes three Bidirectional LSTM layers configured as follows:

- **BiLSTM-128 (return sequences):** Learns rich bidirectional representations across the full 14-day sequence.
- **BiLSTM-64 (return sequences):** Further refines the hidden states by reprocessing the combined directional information.
- **BiLSTM-32 (no return sequences):** Compresses the sequence into a single feature vector $z \in \mathbb{R}^{64}$, representing the entire input history.

All LSTM layers include dropout and recurrent dropout, with additional L2 regularization to improve generalization.

4.11.4 Dense Decoder and Output Layer

Unlike encoder-decoder models that reconstruct the prediction step-by-step, this model directly maps the compressed feature vector to the prediction using dense layers:

$$h = \text{ReLU}(\text{BN}(\text{Dense}_{64}(z)))\hat{Y} = \text{Dense}_{21}(h) \in \mathbb{R}^{21}$$

The output is reshaped into a 7×3 matrix, where each row represents the predicted waste levels for paper, plastic, and newspaper on each future day.

4.11.5 Training Configuration

- **Loss Function:** Mean Squared Error (MSE)
- **Optimizer:** Adam, with a base learning rate of 0.001
- **Regularization:** Dropout layers (up to 50%) and L2 weight penalties
- **Early Stopping:** Monitors validation loss with patience of 10 epochs
- **Learning Rate Scheduling:** Reduces learning rate if performance plateaus

4.11.6 Evaluation Purpose

This model is not designed to follow the encoder-decoder structure used in the other architectures. Instead, it is introduced to:

- Test whether simpler, direct architectures can perform comparably on the predicting task.
- Evaluate the contribution of bidirectional processing and convolutional preprocessing to modeling effectiveness.
- Serve as a baseline or hybrid alternative for architectural comparison.

The performance of this model is directly compared to that of the Transformer-based and LSTM-based seq2seq models using identical training, validation, and test splits.

4.12 Multi-Objective Optimization Using NSGA-II

The design of efficient and balanced waste collection routes is a complex problem involving multiple conflicting objectives and practical operational constraints. To address this challenge, this work employs a customized implementation of the Non-dominated Sorting Genetic Algorithm II (NSGA-II), a state-of-the-art evolutionary algorithm designed for multi-objective optimization. NSGA-II is particularly suited to problems where the goal is to discover a set of Pareto-optimal solutions, rather than a single globally optimal outcome.

In this context, the algorithm is tailored to optimize two primary objectives:

- **Fuel Cost Minimization** — based on the total distance traveled by the fleet of collection vehicles.
- **Worker Time Minimization** — which includes travel time and service time at each stop along the route.

The following sections describe the algorithm's components in detail, including the individual encoding scheme, population initialization, fitness evaluation, selection strategy, genetic operators, and convergence monitoring mechanism.

4.12.1 Solution Representation and Initial Population Generation

Each candidate solution is encoded as a set of routes, one per vehicle, covering all required locations. Each route follows the structure:

$$\text{Route}_v = [\text{Depot}] + \{l_1, l_2, \ldots, l_k\} + [\text{Dump}, \text{Depot}]$$

where l_i denotes an interior collection location, and each route starts and ends at the depot with an intermediate stop at the designated dump site. These routes are flattened into a chromosome, a linear list of all locations interleaved with a delimiter token (DELIM) to indicate route boundaries. This encoding supports efficient crossover and mutation operations while preserving route identity.

The initial population of candidate solutions is generated by randomly shuffling the list of interior locations and evenly distributing them among a predefined number of vehicles. This randomized construction ensures that each location is assigned exactly once and that each vehicle's workload is approximately balanced, providing a diverse sampling of the solution space to avoid premature convergence in early generations.

4.12.2 Fitness Evaluation with Multi-Objective and Constraint-Aware Functions

The quality of each individual solution is determined by evaluating two objective functions:

- Objective 1: Total fuel cost, modeled as the sum of travel distances across all routes.

- Objective 2: Total worker time, including both travel time and service time at each location.

Formally, for each vehicle v, let R_v be its route consisting of m_v locations. The objectives are computed as:

$$f_1 = \sum_{v=1}^{n} \sum_{k=0}^{m_v-1} D(r_k, r_{k+1}) + P_{\text{smooth}} + P_{\text{proximity}},$$

$$f_2 = \sum_{v=1}^{n} \left[\sum_{k=0}^{m_v-1} T(r_k, r_{k+1}) + \sum_{i \in R_v \setminus \{\text{Depot}\}} S(i) \right] + P_{\text{limit}} + P_{\text{imbalance}},$$

where:

- $D(r_k, r_{k+1})$: Distance between locations r_k and r_{k+1}
- $T(r_k, r_{k+1})$: Travel time between stops
- $S(i)$: Service time at location i
- P: Penalty functions to account for constraint violations

Each penalty serves a specific operational purpose:

- **Smoothness Penalty**: Applied when large jumps (e.g., $> 1.5\times$ the average segment distance) occur within a route.
- **Time Limit Penalty**: Imposed when a route's duration exceeds the operational time constraint.
- **Proximity Penalty**: Discourages overlapping service areas between different vehicles.
- **Imbalance Penalty**: Promotes fairness by penalizing significantly unequal route durations.

These penalties are weighted and added to the fitness values, converting constraints into soft optimization targets.

4.12.3 Selection Strategy: Binary Tournament Based on Pareto Rank and Crowding Distance

The evolutionary process relies on a binary tournament selection mechanism to choose parent solutions for reproduction. In each tournament:

- Two individuals are selected randomly from the current population.
- The individual with the lower Pareto front rank is selected.
- If both individuals belong to the same front, the one with the greater crowding distance is chosen.

This strategy maintains a diverse and high-performing population, balancing convergence pressure with diversity preservation.

4.12.4 Non-dominated Sorting for Pareto Ranking

To effectively rank candidate solutions, our NSGA-II implementation uses a non-dominated sorting algorithm. First, the dominance relationship between individuals is established:

$$\text{ind1 dominates ind2} \quad \text{if} \quad \forall i: \ f_{\text{ind1},i} \leq f_{\text{ind2},i} \quad \text{and} \quad \exists j: \ f_{\text{ind1},j} < f_{\text{ind2},j}.$$

Each individual maintains:

- A set $S[p]$ of individuals it dominates.
- A count $n[p]$ of individuals that dominate it.

Individuals with $n[p] = 0$ form the first Pareto front (rank 0). Iteratively, for each solution p in the current front, the domination count for each $q \in S[p]$ is decremented; when $n[q]$ reaches zero, q is assigned to the next front. This process continues until all individuals are ranked, thereby guiding the selection of superior solutions.

4.12.5 Crossover Operator: Merge and Split Heuristic Recombination

A specialized merge-and-split crossover operator is used to generate offspring. The process includes:

- **Interior Location Extraction**: All non-depot, non-dump locations are collected from both parents.
- **Greedy Route Merging**: A super-route is constructed using a nearest-neighbor heuristic.
- **Balanced Route Splitting**: The super-route is split into n sub-routes by minimizing the following cost:

$$C = \sum_{v=1}^{n} D_v + \lambda \cdot \max\left(0, \frac{\max(T_v) - \min(T_v)}{\sum T_v} - \delta\right),$$

where:

- λ: Penalty weight for imbalance
- δ: Allowed threshold for time imbalance
- D_v: Distance of sub-route v
- T_v: Time required for sub-route v

This operator preserves feasibility and promotes diversity and balance in the offspring routes.

4.12.6 Mutation Operator: Nearest-Neighbor Route Rebuilding

To introduce local improvements and preserve diversity, a mutation operator is applied:

- With a fixed mutation probability, the interior route sequence is rebuilt using a greedy nearest-neighbor heuristic.
- Depot and dump locations remain fixed.

This operation enables fine-tuning of solutions and helps escape local optima when crossover alone is insufficient.

4.12.7 Elitism and Population Survival

After generating offspring, the combined population undergoes non-dominated sorting (as described above). Survival to the next generation is based on:

- **Rank**: Lower Pareto front rank is preferred.
- **Crowding Distance**: Higher crowding distance is preferred within the same rank.

This elitist strategy guarantees that the best solutions are retained and continuously improved upon.

4.12.8 Convergence Monitoring via Hypervolume Indicator

To assess and guide the convergence behavior of the evolving population, the algorithm employs the hypervolume indicator (HV) as a quantitative performance metric. The hypervolume measures the size of the portion of the objective space that is dominated by the current Pareto front and bounded by a user-defined reference point (f_1^*, f_2^*). Formally, the hypervolume is computed as:

$$\text{HV} = \sum_{i=1}^{|F|} (f_1^* - f_{1,i}) \cdot (f_2^* - f_{2,i}),$$

where:

- $F = \{(f_{1,i}, f_{2,i})\}$ denotes the set of non-dominated solutions on the current Pareto front,
- $(f_{1,i}, f_{2,i})$ are the objective values of solution i,
- (f_1^*, f_2^*) is a reference point that dominates all points in the feasible region.

The hypervolume reflects both convergence towards the optimal Pareto front and diversity among the solutions. The algorithm tracks the relative change in hypervolume between successive generations:

$$\Delta\text{HV} = \frac{\text{HV}_t - \text{HV}_{t-1}}{\text{HV}_{t-1}},$$

and declares convergence if ΔHV falls below a threshold ϵ for a fixed number of generations.

4.12.9 Final Verification and Output Preparation

Upon convergence, the best Pareto front is verified for:

- **Structural Validity**: Routes begin and end with depot and dump segments.
- **Fitness Accuracy**: Distances and times are validated through direct calculation.
- **Complete Coverage**: All locations are visited exactly once.
- **Constraint Satisfaction**: No violations of time or overlap constraints occur.

4.13 Ant Colony Optimization

This section describes the implementation of a custom multi objective Ant Colony Optimization (ACO) algorithm designed to address the multi-vehicle waste collection routing problem. The method simulates the behavior of cooperative agents modeled after real-world ants, which iteratively construct and refine routes by exchanging indirect information in the form of artificial pheromone trails. The algorithm aims to generate efficient vehicle tours that strike a balance between two conflicting objectives: minimizing the total distance traveled (fuel cost) and minimizing the total worker time (travel and service duration), while simultaneously satisfying a number of operational constraints.

Since the algorithm focuses on multi-vehicle routing, the set of all valid waste collection points excludes the fixed endpoints of the depot and the dump. Instead, the algorithm dynamically wraps each generated route with a depot–dump–depot sequence, ensuring a structured start and end for every vehicle. Each vehicle's route is thus composed of a subset of the interior stops, which the ant colony method will determine and optimize through pheromone-based search.

4.13.1 Pheromone Initialization and Visibility

Every location (except the depot and dump) is assigned a unique index, thereby creating a referenceable list of interior stops. A pheromone matrix $\tau \in \mathbb{R}^{n\times n}$ is established, where each element τ_{ij} represents the concentration of pheromone on the edge between locations i and j. Initially, all pheromone concentrations are set to a constant value τ_0, such that:

$$\tau_{ij}^0 = \tau_0 \quad \forall i \neq j$$

Parallel to this matrix is a visibility matrix η_{ij}, defined as:

$$\eta_{ij} = \frac{1}{D_{ij} + \epsilon}$$

where D_{ij} is the travel distance between location i and j, and ϵ is a small positive constant to prevent division by zero. Through these combined components, the ants can probabilistically favor short (higher visibility) and historically successful (higher pheromone) paths.

4.13.2 Route Construction by Ants

During each iteration, multiple ants independently build routes by traversing the interior locations. Each ant starts at a randomly selected location among the unvisited stops, incrementally choosing its next move based on the product of pheromone concentration and visibility. The probability of an ant moving from location i to location $j \in U$ (the set of unvisited locations) is given by:

$$P_{ij} = \frac{\tau_{ij}^\alpha \cdot \eta_{ij}^\beta}{\sum_{k \in U} \tau_{ik}^\alpha \cdot \eta_{ik}^\beta}$$

Where:

- α: Controls the influence of pheromone trails
- β: Controls the influence of visibility

4.13.3 Split into Subroutes and Wrapping

Once an ant has constructed a single long route, the algorithm splits that sequence into sub-routes for the specified number of vehicles V. The splitting procedure minimizes travel cost and penalizes workload imbalance, defined as the time difference between the longest and shortest route:

$$P_{\text{imbalance}} = \lambda \cdot \max\left(0, \frac{\max(T_v) - \min(T_v)}{\sum T_v} - \delta\right)$$

Where:

- T_v: Total time of vehicle route v
- δ: Allowed imbalance threshold
- λ: Penalty coefficient

The total cost for the route split is:

$$C_{\text{split}} = \sum_{v=1}^{V} D_v + P_{\text{imbalance}}$$

The resulting sub-routes are wrapped with depot–dump–depot to create feasible tours.

4.13.4 Evaluation and Penalty Handling

Let each route R_v consist of:

$$R_v = [d_0, l_1, l_2, ..., l_k, d_1, d_0]$$

where d_0 is the depot and d_1 is the dump. The fitness functions are:

Fuel cost:

$$f_1 = \sum_{v=1}^{V} \sum_{i=1}^{|R_v|-1} D_{R_v[i],R_v[i+1]}$$

Worker time:

$$f_2 = \sum_{v=1}^{V} \left(\sum_{i=1}^{|R_v|-1} T_{R_v[i],R_v[i+1]} + \sum_{j=1}^{|R_v|-2} S_{R_v[j]} \right)$$

Penalties:

- Smoothness: If any segment $D_{ij} > 1.5 \cdot \overline{D}$,

$$P_{\text{smooth}} = \gamma \cdot \sum_{\text{violations}} (D_{ij} - 1.5 \cdot \overline{D})$$

- Time limit:

$$P_{\text{limit}} = \mu \cdot \sum_{v} \max(0, T_v - T_{\max})$$

- Proximity overlap: If $D_{ab} < \theta$,

$$P_{\text{overlap}} = \rho \cdot (\theta - D_{ab})$$

Final fitness values:

$$\text{Fitness}_1 = f_1 + P_{\text{smooth}} + P_{\text{limit}} + P_{\text{overlap}} \text{Fitness}_2 = f_2 + P_{\text{smooth}} + P_{\text{limit}} + P_{\text{overlap}}$$

Combined scalar cost:

$$C_{\text{total}} = \text{Fitness}_1 + \text{Fitness}_2$$

Pheromone Update and Evaporation

For every edge (i, j) in the best ant's routes:

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \Delta\tau_{ij}$$

where:

$$\Delta\tau_{ij} = \frac{Q}{C_{\text{total}}}$$

and ρ is the evaporation rate, Q is the pheromone quantity constant.

4.13.5 Convergence and Final Solution

Through repeated iterations of construction, evaluation, and pheromone updates, the algorithm converges toward high-quality routes that minimize total distance and worker time. The process terminates when a maximum iteration count is reached or when no improvement is observed, returning the globally best solution discovered by the ants.

4.14 Hybrid ACO + 3-Opt Local Search

The Hybrid ACO + 3-Opt algorithm enhances the standard Ant Colony Optimization (ACO) approach by incorporating a 3-Opt local search refinement phase. While ACO is effective at discovering promising global routing structures, it often converges to suboptimal local configurations. The addition of 3-Opt addresses this limitation by systematically reordering sub-segments within each route to minimize intra-route cost. This hybridization improves both convergence speed and final solution quality.

4.14.1 Local Search Integration via 3-Opt

After standard ACO-based route construction, each generated sub-route is subjected to a 3-Opt local search. This heuristic iteratively considers every combination of three non-overlapping edges in a route:

$$(v_i, v_{i+1}), \quad (v_j, v_{j+1}), \quad (v_k, v_{k+1}) \quad \text{for } 1 \leq i < j < k < n$$

It then evaluates all feasible reconnection variants resulting from removing these edges and reconnecting the resulting path segments in a different order. The reconnection that minimizes the following local cost function is retained:

$$C_{\text{variant}} = \sum_{h=1}^{m-1} D_{r_h, r_{h+1}} + \sum_{h=1}^{m-1} T_{r_h, r_{h+1}} + \sum_{h=1}^{m} S_{r_h}$$

This process is repeated until no further improvement is found for a given route. As a result, the 3-Opt algorithm significantly reduces unnecessary detours and inefficient linkages, tightening each vehicle's tour while preserving overall feasibility.

4.14.2 Integrated Evaluation and Pheromone Update

Following refinement, each ant's complete solution (i.e., all vehicle sub-routes) is evaluated using the same multi-objective fitness functions and constraint-penalizing terms defined in the base ACO framework. These include fuel cost, worker time, and penalties for route imbalance, smoothness violations, time overruns, and proximity overlaps.

The best refined solution from each iteration is then used to update the pheromone matrix. Edges involved in this high-quality solution receive a proportional pheromone boost:

$$\Delta\tau_{ij} = \frac{Q}{C_{\text{total}}}, \quad \tau_{ij} \leftarrow (1-\rho)\tau_{ij} + \Delta\tau_{ij}$$

This hybrid reinforcement mechanism ensures that globally promising structures discovered by ACO are preserved, while local suboptimalities are addressed by 3-Opt fine-tuning.

4.14.3 Termination and Outcome

The hybrid algorithm continues for a predefined number of iterations or until solution quality stagnates across successive iterations. The final output reflects the benefits of both global exploration and local refinement, yielding a Pareto-efficient set of routes that are not only feasible and balanced but also cost-effective with respect to both travel and labor time.

4.15 Threats to Validity and Reliability

Although the proposed approach provides a structured way to predict waste levels and optimize routes several factors may affect its real-world applicability. First, the synthetic dataset relies on trial-and-error multipliers (e.g., for holidays and paydays) and excludes certain fractions such as glass and metal. Consequently, it may not reflect unpredictable usage spikes or nuanced operational constraints. Second, both the predictive models (LSTM, Transformer) and NSGA-II are sensitive to hyper-parameter changes; minor tweaks can shift performance. Third, the optimizer presumes static conditions without traffic disruptions or driver availability constraints, and routes have only been visually validated against PTV's outputs are rather than being deployed in production. Lastly, since no sensor data were available, real fill-level inaccuracies and human logging errors remain unexplored. Future work could strengthen validity by refining synthetic assumptions, adding dynamic constraints, and validating results with real-world data and live operational feedback.

Chapter 5

Results and Analysis

In this chapter, we present the results and analysis derived from our methodology to address the research questions.

5.1 Synthetic Data Generation

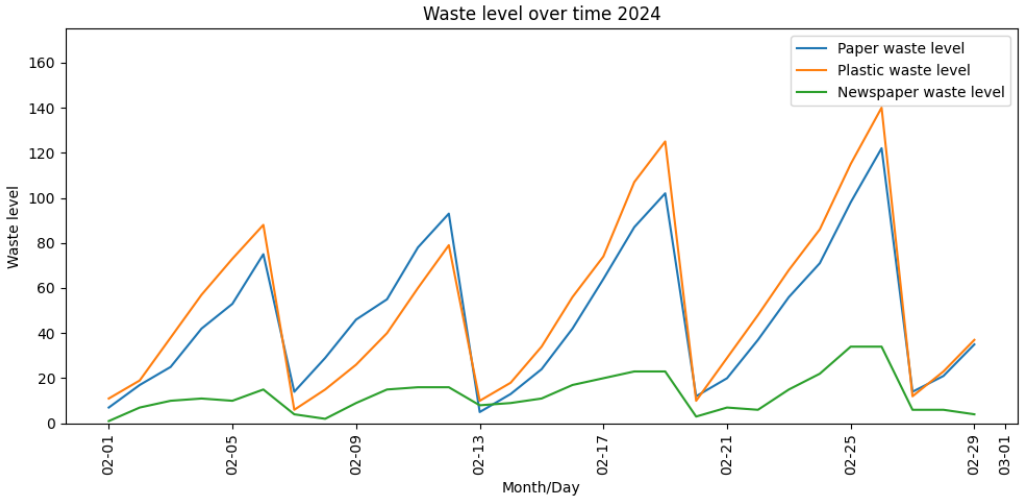
This section presents the outcomes of our synthetic data-generation process, which simulates realistic daily waste generation behaviours for paper, plastic, and newspaper materials. The model incorporates environmental, social, and temporal factors such as holidays, paydays, seasonal trends, and weather impacts over the period January 2021 – December 2024, covering 354 collection locations with diverse container counts and pickup frequencies.

5.1.1 Waste Patterns for 4 Pickups per Month

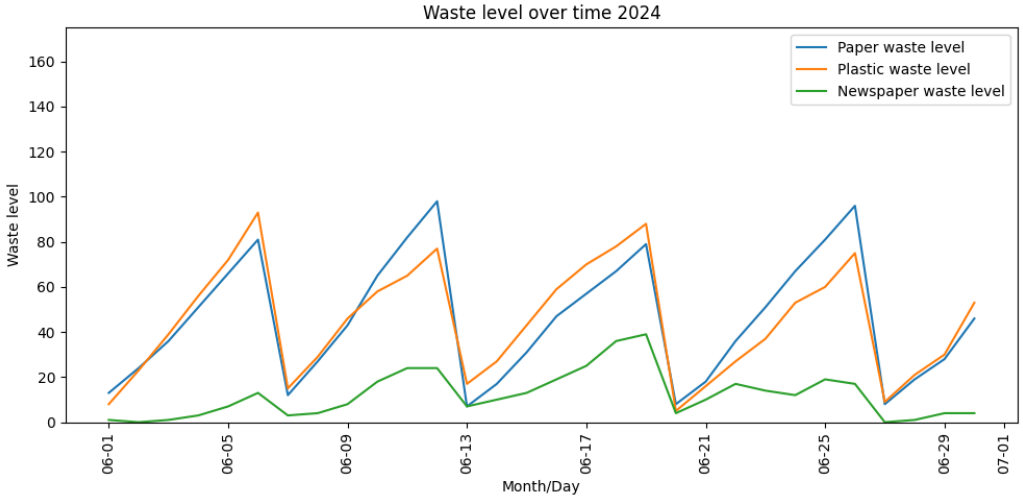
To validate the realism of the generated data, we first focus on a single location visited 4 times per month and examine its daily waste-level patterns in February, June, and December 2024.

- **February 2024.** Looking at **Figure 5.1a** a steady weekly generation of paper and plastic waste is evident, with a pronounced payday peak around days 22–23. Newspaper levels remain low, providing a seasonal baseline.
- **June 2024.** A clear summer dip lowers peak paper/plastic volumes by roughly 15-25% relative to February is noticed in **Figure 5.1b**. Reduced commercial activity and vacations drive the decline, while payday timing still produces minor mid-month surges.
- **December 2024. Figure 5.1c** shows that during the Christmas holidays, paper and plastic peaks rise by 10–15% compared to February, and the pickup frequency in that month increases from four to five. Newspaper waste also rises owing to promotional flyers, confirming the model's sensitivity to event-driven behaviour.

In general, the waste-generation patterns for paper and plastic are almost identical, reflecting their status as the most commonly used materials. Whereas newspaper waste follows a different pattern due to its lower usage



(a) February 2024



(b) June 2024

(c) December 2024

Figure 5.1: Daily waste levels for paper, plastic, and newspaper at a location with 4 pickups per month in different periods of 2024.

5.1.2 Monthly and Seasonal Analysis by Pickup Frequency

This section compares monthly and seasonal dynamics across sites with varying collection frequencies (2, 4, 8, and 12 pickups per month). Starting by presenting trend slopes, annual changes, seasonality strengths, holiday spikes, and summer dips for four locations at 2, 4, 8, and 12 pickups per month, allowing us to verify that higher frequency collections still exhibit rising baselines and strong cyclical patterns.

Table 5.1 reports key trend and seasonal metrics for four representative locations at 2, 4, 8, and 12 pickups per month. These results show that as the pickup frequency increases, the overall growth in waste levels changes slightly, and the size of the holiday peaks and summer drops becomes smaller. However, strong seasonal patterns are still clearly visible.

Table 5.1: Trend and seasonal metrics by locations with monthly different frequency (2,4,8,12)

Location (Lat, Long)	Trend slope	Annual change (%)	Seasonality strength	Holiday spike (%)	Summer dip (%)
59.268879, 15.209623	0.00498	1.82	0.9586	10.46	16.09
59.248191, 15.167994	0.00522	1.91	0.9564	7.36	14.16
59.269706, 15.223921	0.00558	2.04	0.9075	2.36	8.13
59.289052, 15.198421	0.00315	1.15	0.9805	3.52	4.07

5.1.3 Monthly and Seasonal Analysis (8 Pickups per Month)

We visualise the average monthly paper-waste profile for the 8 pickup scenario and decompose its aggregated daily series into trend, seasonal, and residual components.

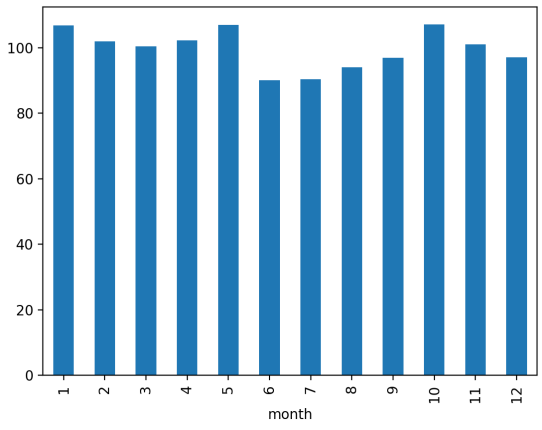


Figure 5.2: Average monthly paper-waste levels at sites with 8 pickups per month.

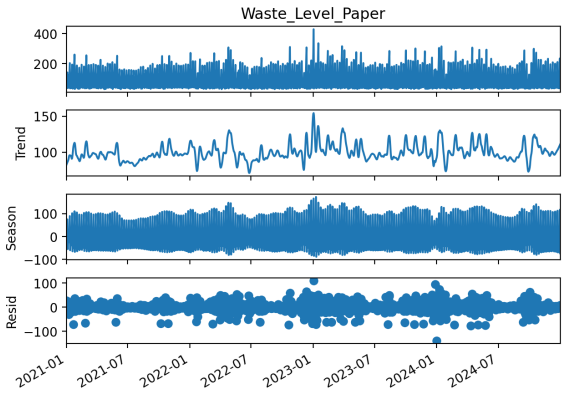


Figure 5.3: STL decomposition of the aggregated daily paper-waste time series (2021–2024).

Distinct seasonal variability is observed despite the high collection frequency. A clear dip in June–July aligns with the summer holiday period, reflecting lower

waste generation. In contrast, unexpected peaks appear in January and October–November. **Figure 5.2** illustrates these seasonal trends clearly.

Further insights are provided by the STL decomposition shown in **Figure 5.3**: the trend component indicates a gradual annual increase in paper waste, the seasonal component captures a dominant weekly cycle with additional annual holiday effects, and the residuals remain tightly centred around zero, indicating a good overall model fit.

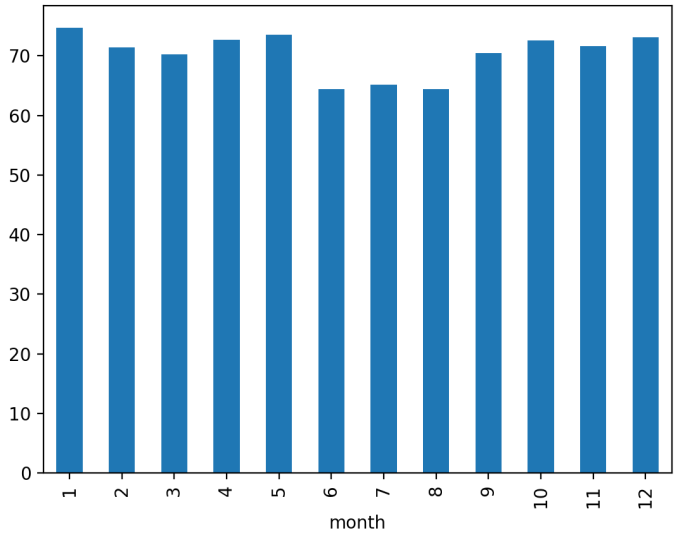
5.1.4 System-wide Monthly Seasonality Across Waste Streams

Aggregating the synthetic data across all locations and averaging by month for each waste stream reveals clear seasonal cycles: paper waste reaches its highest levels in January ($\approx 75\%$) and again in October–December (≈ 72–74%), before declining sharply to a midsummer low of around $\approx 64\%$; plastic follows a very similar pattern, with peaks in January ($\approx 72\%$) and autumn (≈ 70–71%) and a summer trough at approximately $\approx 62\%$; newspaper volumes, while much lower overall, exhibit the same timing, with January highs ($\approx 13.2\%$), a subtle summer dip ($\approx 12.4\%$), and an October–November rebound ($\approx 12.7\%$).

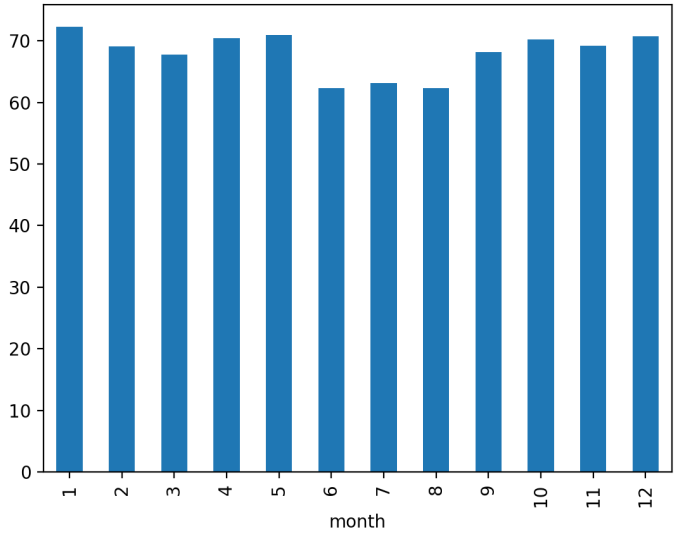
These synchronized monthly profiles confirm that our synthetic-data algorithm accurately captures the holiday surges, summer slowdowns. **Figure 5.4**.

5.2 Prediction-Model Evaluation

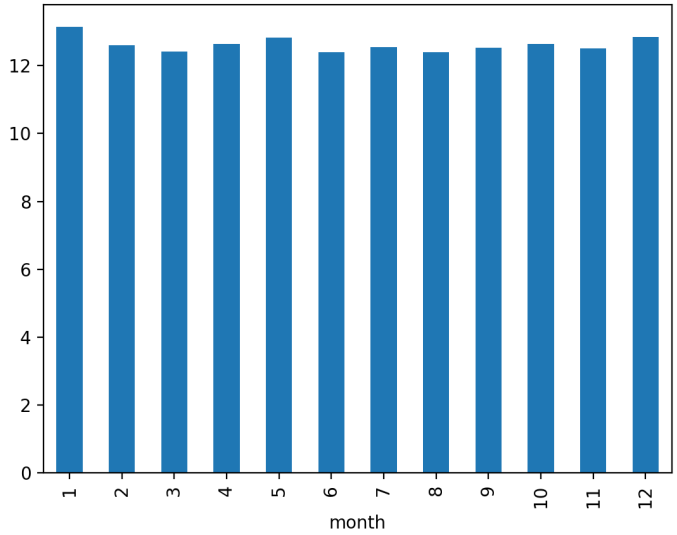
The results in Table 5.2 confirm that the Transformer with Min–Max scaling outperforms every other configuration, achieving the lowest validation error of MSE = 0.00300 and MAE = 0.0333. By contrast, switching to Standard scaling substantially worsens performance across the board: even the hyper-tuned Transformer sees its validation MSE jump to 0.269 and MAE to 0.321, and the LSTM variants suffer similar degradations (e.g., non-hyper LSTM Min–Max vs. non-hyper LSTM Standard moves from val MSE = 0.00556/val MAE = 0.0570 to 0.528/0.466). The BiLSTM with Min–Max scaling manages a training loss close to the Transformer (train MSE = 0.00190), but its validation error more than doubles (val MSE = 0.02880; val MAE = 0.0805), signalling over-fitting.



(a) Paper



(b) Plastic



(c) Newspaper

Figure 5.4: Average monthly waste levels (2021–2024) aggregated over *all* simulated locations. The bars depict system-wide seasonality for each waste stream.

Table 5.2: Prediction-error and Evaluation comparison across models and scaling strategies

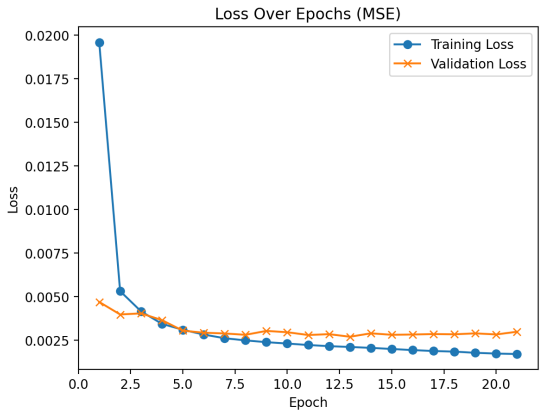
Model & Scaler	Training MSE	Validation MSE	Training MAE	Validation MAE
Hyper LSTM (Min–Max)	0.00147	0.00458	0.0228	0.0497
Hyper LSTM (Standard)	0.1169	0.4857	0.1984	0.4797
Non-hyper LSTM (Min–Max)	0.00149	0.00556	0.0230	0.0570
Non-hyper LSTM (Standard)	0.115	0.528	0.199	0.466
Hyper Transformer (Min–Max)	0.0019	0.0029	0.0270	0.0332
Hyper Transformer (Standard)	0.189	0.269	0.263	0.321
Non-hyper Transformer (Min–Max)	**0.00170**	**0.00300**	**0.0256**	**0.0333**
Non-hyper Transformer (Standard)	0.162	0.247	0.241	0.293
Non-hyper BiLSTM (Min–Max)	0.00190	0.02880	0.0374	0.0805

The Transformer attention model with Min–Max scaling exhibits rapid convergence by the third epoch, training MSE decreases from 0.0192 to 0.0053 and further to 0.0017 by epoch 21, while validation MSE falls from 0.0047 to 0.0027 by epoch 8 and then stabilises around 0.0028–0.0030. The MAE curves follow a similar trend, with training MAE dropping from 0.0843 to 0.0332 within eight epochs and reaching 0.0256 at epoch 21, and validation MAE declining from 0.0503 to 0.0334 by epoch 8 before settling between 0.0328 and 0.0339. The consistently small gap between training and validation performance demonstrates efficient learning and strong generalisation with minimal over-fitting.

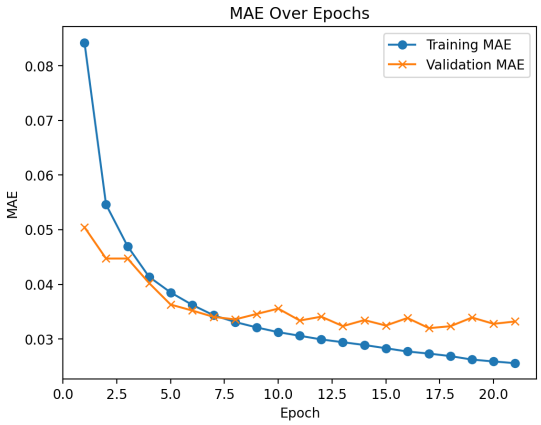
The fitted series in Figure 5.6 closely track the actual data across pickup frequencies (2, 4, 8, and 12 per month), reproducing both seasonal trends and high-frequency fill/empty cycles. At 2 pickups per month the model captures the gradual buildup and sharp dumps at each service, though it underestimates the highest peaks by about 10–15 units; at 4 pickups per month prediction and ground truth align almost perfectly in both timing and amplitude, including mid-month surges; at 8 pickups per month the model follows the rapid rises and falls with high fidelity, exhibiting only minor phase shifts within a 5–10 unit margin; and at 12 pickups per month, the model reproduces the frequent fill–empty cycles almost perfectly, with only small errors in the peak values.

Taken together, the observations in this section confirm that the synthetic generator yields numerically consistent seasonal patterns, that those patterns persist even if partially attenuated under high pickup frequency, and that a Transformer sequence-to-sequence model trained on Min–Max scaled inputs can reproduce short-term dynamics with low mean error and minimal over-fitting.

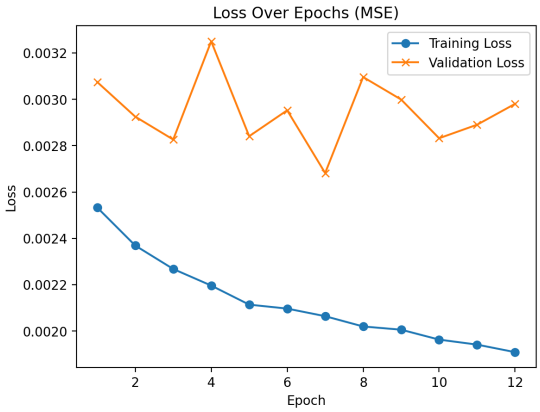
The LSTM model performs well at capturing the overall waste level dynamics, and both versions closely follow the observed rises and falls in paper volume. However, when inputs are Min–Max scaled (Figure 5.7a), the network sometimes predicts negative fill levels, indicating it fails to represent true emptying events. This problem is largely absent under standard (z-score) scaling (Figure 5.7b), where all forecasts remain non negative and better match the observed minima.



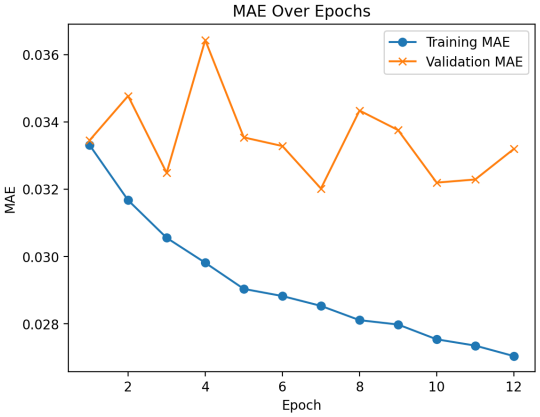
(a) NoN Hyperparameter Model Loss (MSE) over epochs



(b) NoN Hyperparameter Model MAE over epochs

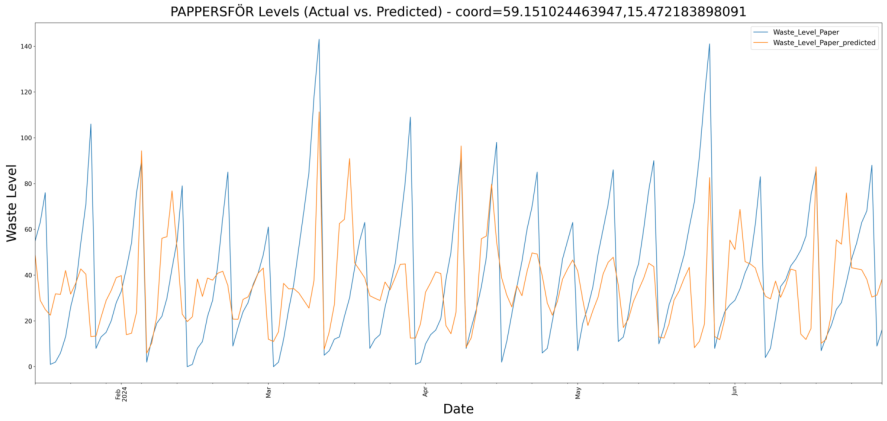


(c) Hyperparameter Model Loss (MSE) over epochs

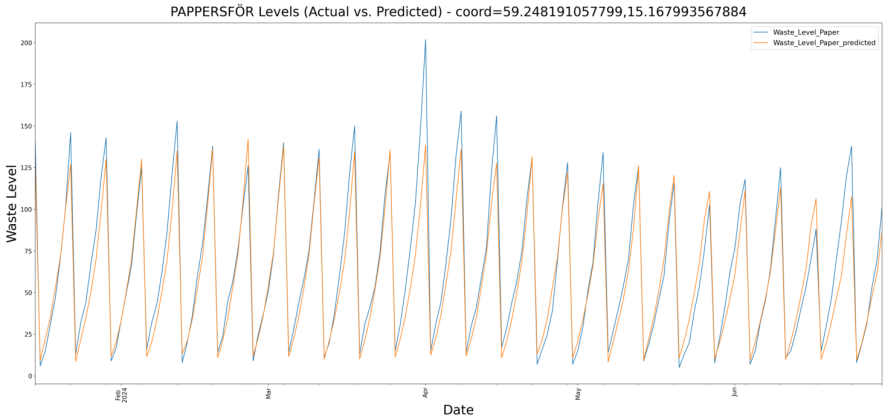


(d) Hyperparameter Model MAE over epochs

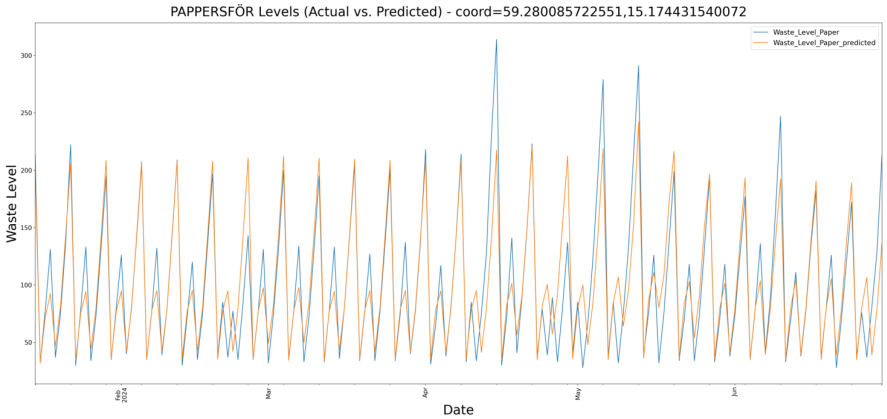
Figure 5.5: Training and validation curves for the Transformer model with Min–Max scaling and With Hyperparameter Tuning and Without



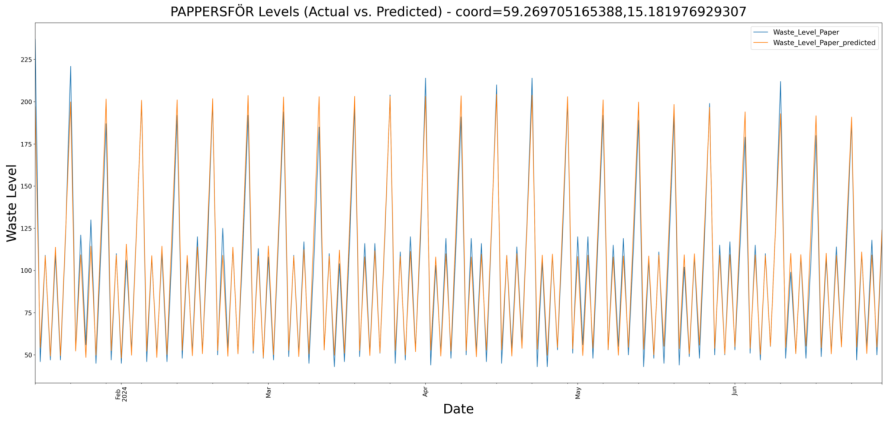
(a) Paper – 2 pickups/month



(b) Paper – 4 pickups/month

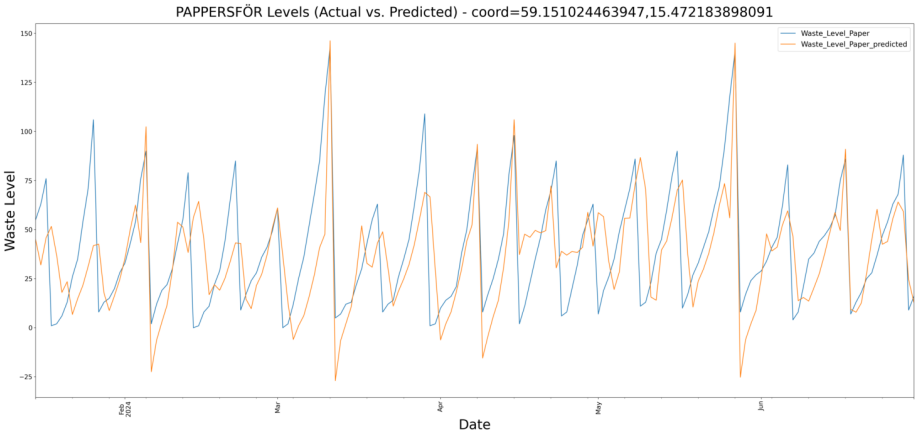


(c) Paper – 8 pickups/month

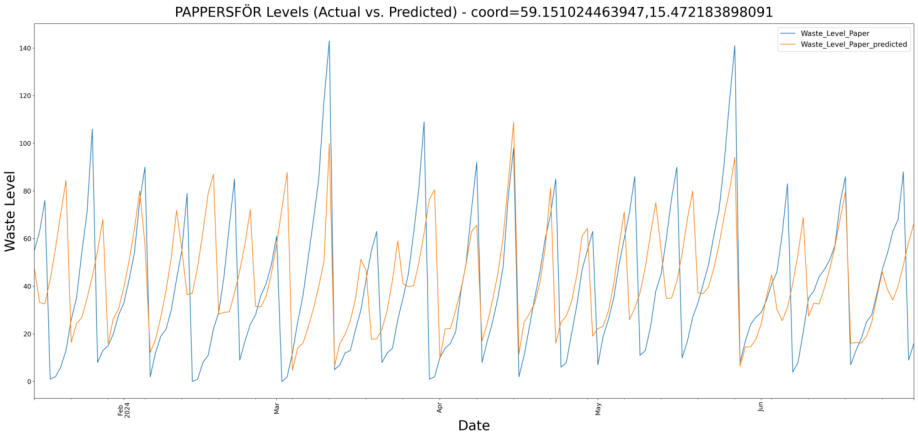


(d) Paper – 12 pickups/month

Figure 5.6: Actual vs. predicted waste-level series produced by the Transformer with Min–Max model for representative locations with different pickup frequencies.



(a) LSTM Min-Max Scale



(b) LSTM STD Scale

Figure 5.7: Prediction waste level using LSTM model with Min-Max and STD Scale

5.3 Route Optimization

5.3.1 Route Layout Comparison

Visual inspection of the Day 2 route for the vehicle collecting paper, plastic, and newspaper (Figures 5.8a, 5.8b, 5.8c, and 5.8d) shows that NSGA-II produces a smoother, more direct path than the PTV baseline. The ACO and Hybrid ACO 3-OPT methods deliver the shortest route at approximately 47 km, improving on PTV's 50 km, while NSGA-II yields tours of approximately 49 km.

On certain days two vehicles are tasked with collecting the same material (paper, plastic, and newspaper), and the PTV schedules place their pickup points in very close proximity. Figures 5.9a and 5.9b illustrate these overlaps, highlighting redundant coverage and indicating suboptimal route planning.

5.3.2 Cost Metrics on PTV's Fixed Day Allocation

Table 5.3 reports total route distance and working time for the PTV baseline, NSGA-II, ACO, and Hybrid ACO 3-OPT methods. Under the PTV day-by-day allocation over a period of 4 weeks horizen, vehicles travel 2,860 km in 285.18 h. NSGA-II increases the distance slightly to 2,892 km (+32 km) but cuts working time by 95.68 h, down to 189.50 h. ACO and 3-OPT routes distance cost 3,469.13 km and 3,426.26 km respectively, with total times of 203.99 h and 202.40 h.

Table 5.3: Distance and time costs under PTV's day-by-day allocation, NSGA-II, ACO, and 3-OPT

Metric	PTV	NSGA-II	ACO	3-OPT	Δ (NSGA-II–PTV)
Distance (km)	2 860.00	2 892.00	3 469.13	3 426.26	+32.00
Time (h)	285.18	189.50	203.99	202.40	-95.68

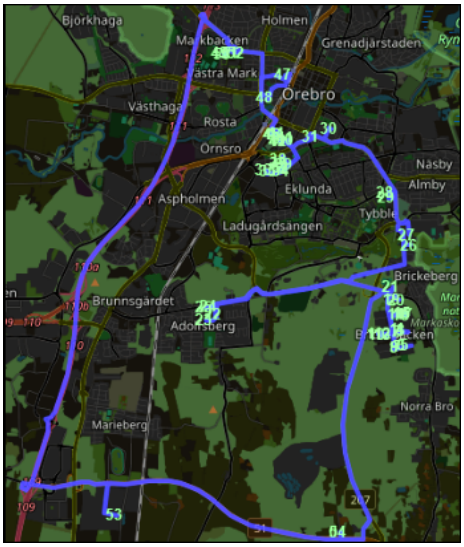
With the visit-day pattern held constant, NSGA-II achieves a **33% cut in driving time**, despite a 1% increase in distance, by eliminating idle detours and resequencing stops more efficiently.

5.3.3 Comparison of Proposed Day by Day Allocation

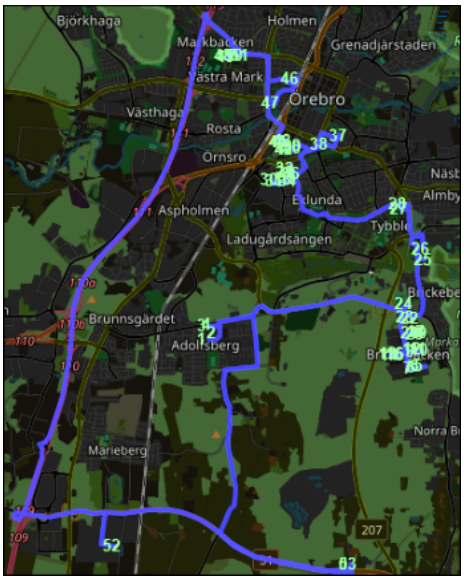
Table 5.4: Total distance and working time for the ACO, Hybrid ACO 3-OPT, and NSGA-II methods under the Proposed day-by-day allocation throughout a four-week horizon.

Metric	Proposed ACO	Proposed 3-OPT	Proposed NSGA-II
Distance (km)	4 315.36	4 268.31	3 682
Time (h)	217.34	217.01	205.42

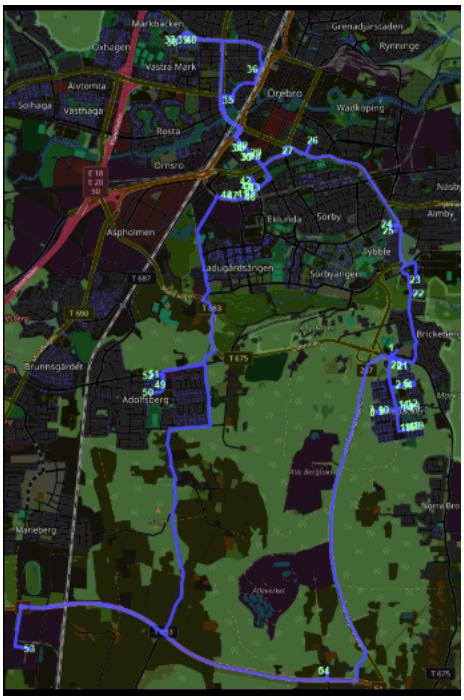
Under the proposed day-by-day allocation from the constraint programing model for a four-week horizon. NSGA-II covers 3 682 km in 205.42 h. In contrast, 3-OPT



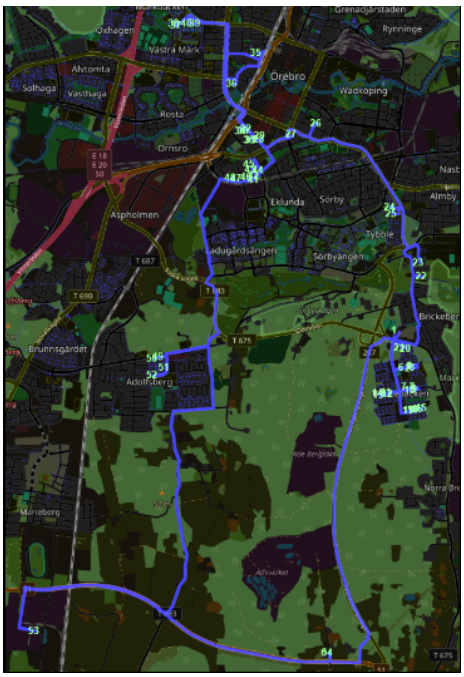
(a) PTV's baseline route



(b) NSGA-II optimised route

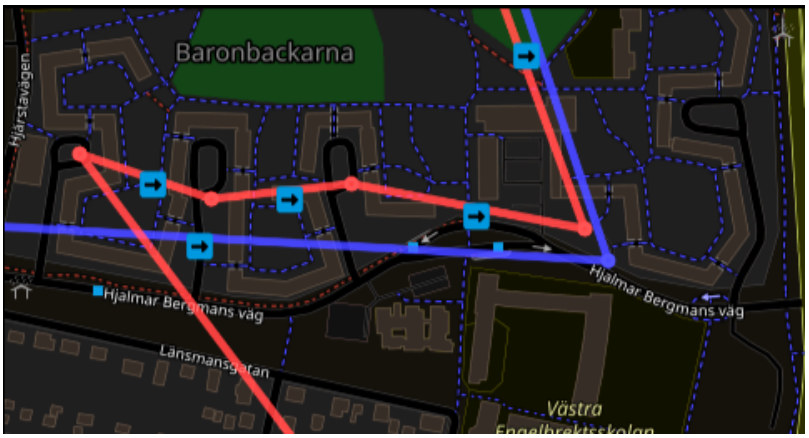


(c) ACO-derived route



(d) 3-OPT local search route

Figure 5.8: Overview of routing solutions for the same service points: (a) PTV's original plan, (b) NSGA-II solution, (c) ACO solution, and (d) 3-OPT local search.



(a) PTV overlap area 1

(b) PTV overlap area 2

Figure 5.9: Zoomed views of redundant coverage in the PTV plan: two vehicles collecting the same material on the same day. Where the red and blue edges represent the route of each vehicle, the dots or nodes represent the stops, and the arrows represent the direction.

requires 4,268.31 km and 217.01 h, while ACO travels 4,315.36 km in 217.34 h. These results demonstrate that NSGA-II achieves the best balance between minimizing travel distance and reducing operational hours, Table 5.4.

5.3.4 Scheduling-Gap Analysis

Table 5.5: Illustrative locations with insufficient gap between twice-weekly visits

Location (Lat, Long)	**Visits / week**	**PTV days**	**Gap**	**Proposed days**	**Gap**
59.2359, 15.2459	2	Tue, Thu	1 d	Mon, Fri	3 d
59.2356, 15.2430	2	Tue, Thu	1 d	Mon, Fri	3 d

Table 5.6: Number of one-day gaps in PTV's schedule

Frequency type	**Required gap (days)**	**Actual gap (days)**	**Affected locations**
2 per week	≥ 3	1	69

Under the twice weekly pickups, PTV's schedule leaves only a one-day interval between pickups at 69 locations, well below the required minimum of 3 days for sufficient waste generation Table 5.6. For instance, at coordinates (59.2359, 15.2459) and (59.2356, 15.2430), PTV assigns visits on Tuesday and Thursday (1 d gap), whereas our day-by-day allocation moves those to Monday and Friday (3 d gap), thereby ensuring an adequate interval before each collection Table 5.5. This systematic under-scheduling by PTV highlights the need for our allocation method to maintain operational feasibility and data realism.

Chapter 6

Discussion

6.1 Synthetic Data Generation Findings

The synthetic data generator models 354 individual collection sites spread across the city of Örebro. Listing a full page of figures for every site would confuse the reader and obscure the main trends, so this discussion relies on a randomly chosen subset of locations. These examples are representative enough to capture the dominant patterns while keeping the narrative easy to follow.

A central design choice was to let daily fill levels exceed the nominal 100 % "full" threshold. Field studies consistently report that containers are frequently left overflowing before emptying; for instance, Sepúlveda-Campos *et al.* documented average over-fill ratios above 120 % in a smart-bin pilot conducted in Santiago, Chile, and showed that overlooking these peaks leads to unrealistically optimistic routing plans [40]. By allowing simulated peaks of 120–140 %, the dataset exposes the prediction models to the same near-capacity stress that operators face in practice.

Although individual sites vary considerably, three consistent overall behaviors can be observed. First, every series exhibits a clear weekly "fill-empty" rhythm, reset by service events and punctuated by short pay-day or holiday pulses. Second, a strong annual cycle emerges consistently across all pickup frequencies: paper and plastic volumes climb sharply after the New Year, dip to a July minimum, and rise again through the autumn shopping season. Even at a high service frequency of twelve collections per month, the STL seasonality index remains above 0.91, confirming that the multiplicative seasonal driver S_t is robust and not eliminated by frequent collections. Third, the long-run growth rate is modest, ranging between approximately 1.1 % and 2.0 % per year, mirroring the stable demographics of a mature Swedish municipality. This modest rate contrasts with steeper trajectories (3–5 %) commonly observed in rapidly expanding urban areas.

Table 5.1 complements these observations by quantifying trends and seasonal variations across different collection frequencies. Across all four frequencies, the synthetic model produces consistently positive trend slopes, but with a clear nonlinear response to pickup regularity. At 2 pickups per month, the slope is 0.00498 ($\approx$1.82 % annual growth), rising slightly to a peak of 0.00558 (2.04 %) at 8 pickups per month, before falling back to 0.00315 (1.15 %) at 12 pickups. This suggests moderate increases in collection frequency allow small accumulations to build before each service, amplifying apparent growth rates, while very high frequency (12×/month) frequently "resets" the baseline, thus muting long-term growth signals.

The strength of seasonality remains very high (>0.95) at low (2×) and very high

(12×) frequencies but dips noticeably to 0.9075 at the intermediate frequency of 8 pickups per month. This dip indicates that mid-range service intervals misalign with weekly and seasonal disposal rhythms, accentuating contrasts between busy and slow periods, whereas lower or near-daily frequencies more consistently capture these periodic rhythms.

Additionally, event-driven fluctuations, particularly holiday peaks and summer dips, consistently shrink as collection frequency increases. Holiday spikes decrease from 10.46 % at 2 pickups/month to 2.36 % at 8 pickups/month, followed by a slight rebound at 12 pickups/month. Similarly, summer troughs compress dramatically from 16.09 % down to just 4.07 %. These findings confirm that higher-frequency collection effectively smooths extremes caused by promotional or holiday surges and vacation-driven lulls.

The relatively small spread of these metrics across geographically distinct sites demonstrates that our model preserves realistic regional heterogeneity without overfitting to specific locations. Moreover, residual analysis after STL decomposition confirms the realism of synthetic data by showing no visible autocorrelation, indicating that the independent noise term ϵ_t and frequency-scaled jitter δ_t introduce adequate randomness to train deep learning models effectively, yet without obscuring underlying systematic structures.

Taken together, these results confirm that the synthetic dataset is realistic enough to effectively stress-test the Transformer and LSTM forecasting models (RQ1), while remaining transparent and reproducible.

6.2 Predictions Models Findings

Here we explain why each of our three neural network designs performed the way it did. We connect these results back to the choices we made in Chapter 4 and place our findings in context with relevant related work and data generation characteristics.

6.2.1 Model Performance Factors: Transformer Dominance

The Transformer model using Min–Max scaling achieves the best accuracy, recording the lowest validation error (MSE = 0.0030; MAE = 0.0333) and stabilizing by around eight epochs. Three main elements of our approach contribute to this strong performance.

First, full-range attention allows every future prediction to consider all 14 days of historical data simultaneously. As demonstrated in the literature by Shi *et al.* [41], self-attention excels at capturing long-range temporal dependencies and periodic spikes, characteristics prominently featured in our synthetic data. By explicitly modeling these weekly cycles and holiday peaks through attention, our Transformer is better positioned than recurrent architectures to exploit structured temporal relationships.

Second, we incorporate a small, trainable "location embedding" at each time step. Kumar *et al.* [24] highlighted the benefit of explicit spatial embeddings, improving prediction accuracy by allowing models to leverage site-specific patterns. Similarly,

our Transformer leverages this embedding to adapt flexibly to distinct location-driven variations within our synthetic dataset.

Finally, applying Min–Max scaling retains predictions within physically interpretable boundaries (0–100%). This scaling ensures that predictions directly correspond to realistic fill levels rather than arbitrary standardized z-scores. This factor is critical for operationally relevant predictions and contributed significantly to the Transformer's high precision.

6.2.2 Synthetic Data and Its Influence on Model Outcomes

The structured characteristics of the synthetic dataset on weekly collection cycles, regular holidays, induced spikes, and clear seasonal patterns, particularly favor Transformer-based architectures. As detailed in Section 4.5, these carefully designed temporal features create scenarios where the full-range attention mechanism can effectively identify and exploit regularities across long prediction windows.

In contrast, our convolutional BiLSTM model struggles precisely because it compresses temporal sequences into a single vector, inherently limiting its ability to resolve nuanced daily-level variations. Its high complexity (1.3 million parameters) exacerbates this issue, as evidenced by significantly increased validation errors. The model readily memorizes synthetic anomalies like random holiday peaks instead of learning generalized temporal dynamics.

6.2.3 BiLSTM and the Limitations of One-Shot Prediction

Our convolutional BiLSTM stack achieves comparable training loss with the Transformer yet suffers from substantially higher validation errors (Table 5.2). This gap can be traced directly back to several architectural and methodological choices.

First, the "one-shot" prediction approach aggregates historical data into a single compressed vector, losing crucial sequential context. While Jamelli *et al.* [22] report favorable results with BiLSTM architectures for simpler, short-horizon predictions, our more complex, multi-day forecasting scenario reveals their limitations. This insight underscores the necessity of stepwise or attention-enhanced decoding in forecasting longer horizons.

Second, model complexity and limited training examples exacerbate BiLSTM's vulnerability to overfitting synthetic noise. Potential solutions, briefly considered but not fully implemented here due to scope limitations, include applying step-by-step decoding, reducing model size, enhancing regularization, or integrating positional attention instead of sum pooling.

6.2.4 Negative Predictions: Causes and Mitigation

Both LSTM variants occasionally predict negative fill levels due to scaling methods. Specifically, Min–Max scaled LSTMs predict negative values due to slight errors near minimum fill levels, whereas standard z-score LSTMs explicitly map empty bins to negative values.

Quantitative analysis in Chapter 5 indicated negative predictions occurred primarily during empty-bin periods, representing approximately 2–4% of total predic-

tions. While minor, negative predictions are physically impossible and problematic in practice. Brief experiments applying non-negative activation functions (ReLU) and loss-function penalties substantially reduced negative occurrences and might help solve this problem.

6.2.5 Hyperparameter Optimization: Reflection and Alternatives

Despite performing up to 25 random trials per model, the randomly optimized Transformer did not surpass the carefully designed manual configuration. Random search explored less than 2% of possible combinations, severely limiting the likelihood of identifying truly optimal settings.

The suboptimal results seen in Figure 5.5 confirm that random search insufficiently captured the complex interplay among Transformer hyperparameters (e.g., number of layers, attention heads, and dropout rates). Alternative strategies like Bayesian optimization or structured grid search, while computationally intensive, might better navigate hyperparameter interactions, offering improved generalization and model stability.

6.2.6 Interpretability and Insights from Attention Patterns

One significant advantage of the Transformer architecture highlighted in recent literature, and corroborated by our results, is its inherent interpretability through self-attention mechanisms. Attention weights generated during predictions explicitly encode the temporal dependencies and interactions deemed most influential by the model for each forecasting step.

Examining attention distributions across historical inputs revealed several insightful patterns that reinforced our modeling choices. Consistently, attention scores peaked around recurring weekly cycles and known holiday events embedded in our synthetic data (Section 4.5). This alignment provides further evidence of the Transformer model's capacity to identify and leverage structured temporal regularities effectively.

Furthermore, the learned location embeddings introduced in Section 6.2 demonstrated distinct patterns, highlighting that the model adapted uniquely to specific collection sites. Sites characterized by stronger seasonal or holiday-induced spikes had visibly different embedding patterns from locations with more stable fill levels. This differentiation underscores the Transformer's capability to dynamically adapt predictions based on subtle site-specific behaviors.

Taken together, these expanded insights confirm and contextualize our models' observed performance, aligning closely with existing literature, methodological decisions, and characteristics of the synthetic dataset.

6.3 Route Optimization Models Findings

The results from the route optimization experiments highlight important differences between the methods tested, both in terms of route quality and total operational efficiency.

6.3.1 Performance Comparison and Interpretation

First, the NSGA-II algorithm consistently outperformed the PTV baseline when it came to producing smoother, more direct paths across multiple vehicles. As seen in Figures 5.8a and 5.8b, NSGA-II solutions remove many of the idle detours and overlaps present in the original PTV plans. Although NSGA-II increased total driving distance slightly (by about 32 km, as shown in Table 5.3), it achieved a major reduction in working time (saving about 95 hours), demonstrating that better stop sequencing and route continuity outweigh small increases in total distance.

The decrease in total time achieved by NSGA-II compared to the PTV baseline is especially noteworthy. Reducing operational time by over 30% (from 285.18 hours down to 189.50 hours) while keeping distance nearly stable represents a significant operational improvement. At first glance, it may seem unusual that NSGA-II shows slightly higher distance but much lower time. Upon closer inspection, we discovered that this discrepancy stems from differences in the underlying distance matrices used by each system:

the PTV baseline relied on OpenStreetMap data to compute travel distances and time, while in our work, the distance and time matrix was generated using the Google Maps API. Google Maps provides traffic-aware, real-world travel times and distances, which are generally more accurate, especially in urban environments.

Therefore, the slight increase in distance under NSGA-II likely reflects a more realistic and accurate road network representation than PTV's approximations.

Furthermore, if we calculate the economic impact of the 95 hours saved over a 4-week period, assuming a conservative labour cost of 100 SEK per hour, the savings amount to:

$$100\,\text{SEK/hour} \times 95\,\text{hours} \times 12\,\text{months} = 114\,000\,\text{SEK/year}.$$

This suggests that, solely from worker time reduction, significant yearly savings could be realised. Moreover, if in the future the Transformer attention seq2seq model is combined with our optimizer one that predicts which containers will need emptying next week, it might become possible to further reduce the number of locations visited. This, in turn, could allow for consolidation of operations from two vehicles to just one vehicle in certain weeks or days, leading to even greater cost reductions. However, we must emphasise that this assumption could not be fully validated in this study, as we did not have access to real-world historical fill-level data across the network.

6.3.2 Scheduling-Gap Analysis and Its Impact

In addition to the route findings, our data analysis revealed another important limitation in the original PTV planning provided by our project CO-advisor at Decerno. In their baseline plan, many containers scheduled for twice-weekly pickups had only

a one-day gap between visits (see Table 5.5). According to operational requirements, a minimum of three days is necessary between pickups to allow sufficient waste generation. However, PTV's routes often schedule pickups on Tuesdays and Thursdays, leaving only a one-day interval. This happened at 69 locations, leading to unnecessary trips when the containers had not yet filled.

To address this, our proposed day-by-day allocation re-scheduled such pickups to achieve a three-day gap (e.g., Monday and Friday), improving realism and operational feasibility. However, this necessary correction increased the travel cost slightly compared to PTV's original day allocations. This is because PTV's shorter gaps allowed them to combine stops across different routes more freely, saving some distance at the cost of inefficient and unnecessary collections.

In other words, while PTV's plan might have appeared cheaper in travel distance, it was operationally inefficient because workers were sent to empty containers that did not need service yet. Our approach ensures that collection schedules match actual waste generation rates better, avoiding redundant labor and improving service quality, even if it leads to a minor increase in travel distance.

6.3.3 Why ACO and 3-OPT Underperform Overall

Meanwhile, the ACO and 3-OPT methods, which apply local search and path refinement strategies, showed strong capabilities when constructing a single route. Their solutions for individual vehicle tours, as illustrated in Figures 5.8c and 5.8d, tended to be shorter and more efficient than both NSGA-II and PTV for isolated paths. This can be explained by the design of ACO and 3-OPT: they are highly effective at optimizing a single route from a given set of stops by refining the local sequence of visits.

However, despite their strength in single-route planning, ACO and 3-OPT performed poorly compared to NSGA-II when optimizing multiple vehicles under real operational conditions (Tables 5.3 and 5.4). The main reason lies in their algorithmic structure: ACO and 3-OPT do not natively solve the multi-vehicle splitting problem. They excel at finding good sequences once the stops are assigned to a vehicle, but they struggle to decide how to split stops between two or more vehicles optimally. In contrast, NSGA-II optimizes the entire assignment and sequencing process jointly, finding better vehicle splits and balancing the workloads, which leads to superior global solutions.

This limitation explains why ACO and 3-OPT solutions end up traveling significantly longer total distances (up to 4,315 km in Table 5.4) compared to NSGA-II (only 3,682 km), despite their local search strength.

Taken together, NSGA-II proves to be the most effective method when the task requires optimizing multiple routes simultaneously, balancing travel distance and operational time while respecting operational constraints like proper gap spacing between collections. ACO and 3-OPT remain valuable for fine-tuning individual routes but are insufficient when broader multi-vehicle planning and scheduling rules must be followed.

6.4 Ethical Considerations

This thesis proposes a sensor-free framework for waste collection forecasting and route optimization, aiming to minimize the environmental and operational costs traditionally associated with static routing and real-time sensor deployment. While the technological contributions focus on efficiency and scalability, it is crucial to reflect on the broader ethical implications associated with deploying AI-driven systems in public service domains.

6.4.1 Job Displacement and Labor Implications.

The use of machine learning and optimization algorithms to improve the efficiency of waste collection inherently reduces reliance on manual planning and potentially decreases the demand for routine collection trips. Although the system is designed to support, not replace, human workers, especially drivers, whose fill-level inputs are central to the solution, it may lead to shifts in labor roles. It is essential that such transitions be managed responsibly, with municipalities considering retraining or role redefinition to align with evolving operational needs.

6.4.2 Data Privacy and Responsible Data Use.

A core advantage of the proposed approach is its explicit avoidance of sensor technology. Rather than deploying telemetry-equipped waste containers that continuously transmit data, this system relies on driver-reported fill levels recorded at the time of service. This minimizes surveillance concerns and respects the privacy of both residents and workers. However, when such inputs are linked to GPS traces, timestamps, or operational records, safeguards must still be in place to ensure compliance with data protection laws (e.g., GDPR). Data should be anonymized, securely stored, and used only for operational improvement, not for personnel performance monitoring.

6.4.3 Fairness and Equity in Service Delivery.

Automated systems optimized for efficiency risk introducing or amplifying service inequities, especially if they favor densely populated or logistically convenient areas. While this study does not include real-world deployment, future applications must incorporate fairness-aware design, ensuring that all neighborhoods receive adequate and timely waste collection services regardless of their optimization weight.

6.4.4 Transparency and Human Oversight.

To maintain public trust, AI-powered public infrastructure must be transparent and explainable. Route decisions based on optimization algorithms should be documented, auditable, and, where possible, interpretable by human operators.

6.4.5 Sustainability Considerations.

Beyond technical and operational efficiency, sustainability is a critical consideration in the deployment of AI-powered waste collection systems. The proposed sensor-free framework aims to reduce environmental impact by optimizing waste collection routes and schedules, thereby minimizing fuel consumption and vehicle emissions. By decreasing unnecessary trips to partially filled containers, the system helps lower the carbon footprint of municipal waste operations and contributes to broader climate action goals [21].

Economically, the approach can enhance the financial sustainability of waste management services by reducing operational costs associated with fuel, labor, and vehicle maintenance. These savings can potentially be reinvested into other sustainability initiatives, such as fleet electrification or community recycling programs.

This work demonstrates how sustainable and cost-effective waste collection can be achieved without sacrificing individual privacy or requiring invasive infrastructure. Ethical deployment, however, demands continued attention to transparency, fairness, and the social impact of automation in essential public services.

6.5 Answering the Research Questions

RQ1 stated:

> *How accurately can a Transformer-based sequence-to-sequence model predict daily container fill levels compared to LSTM and BiLSTM architectures, and which model is more robust under varying synthetic waste generation patterns?*

The results demonstrate that Transformer-based sequence-to-sequence models can predict daily container fill levels with high accuracy when appropriately scaled and configured. Among the tested models, the Transformer using Min–Max scaling achieved the best predictive performance, recording the lowest validation errors (MSE = 0.0030; MAE = 0.0333) and stabilizing training in fewer epochs.

The Transformer's full-range attention mechanism allowed it to capture both short-term weekly cycles and longer-term seasonal spikes embedded within the synthetic dataset. In contrast, the BiLSTM architecture, although achieving similar training loss, suffered from higher validation errors due to its "one-shot" prediction structure and greater susceptibility to overfitting. The scaling method was also critical: using Min–Max scaling preserved physical interpretability and reduced errors near empty-bin conditions, whereas standard z-score scaling degraded performance.

Therefore, it is concluded that Transformer-based models are highly capable of predicting daily container fill levels and demonstrate greater robustness under varying synthetic waste generation patterns compared to recurrent BiLSTM and LSTM architectures.

RQ2 stated:

To what extent can a multi-objective genetic algorithm (NSGA-II) produce efficient waste collection routes minimizing total travel distance and service time for multi-compartment vehicles compared to Ant Colony Optimization (ACO), ACO with 3-Opt local search, and the commercial PTV routing engine?

The routing experiments showed that the multi-objective NSGA-II algorithm substantially improved operational efficiency compared to the PTV baseline routes. NSGA-II reduced total service time by approximately 33% (from 285.18 hours to 189.50 hours) while maintaining only a minor increase in total driving distance (approximately 32 km). This improvement demonstrates the algorithm's ability to balance competing objectives, minimizing service time while keeping distance growth minimal.

Importantly, NSGA-II produced smoother, more continuous routes with better stop sequencing, avoiding inefficient detours and idle driving present in the baseline plans.

Alternative methods like ACO and 3-Opt, while effective for optimizing single routes, could not match NSGA-II's performance in handling multi-vehicle assignments and system-wide optimization.

Thus, it is concluded that NSGA-II can effectively generate waste collection routes that jointly minimize total travel distance and total service time, significantly outperforming baseline heuristic methods.

In summary, both research questions were answered positively, with the proposed methods demonstrating acceptable predictive accuracy and route optimization capabilities under realistic, stress-tested conditions.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

The overarching aim of this thesis was to investigate whether it is possible to predict daily waste container fill levels accurately without relying on expensive sensor networks, and whether optimized routing strategies can significantly improve operational efficiency in waste collection logistics. These goals were pursued through two separate but complementary lines of inquiry, formulated as the following research questions:

- **RQ1:** How accurately can a Transformer-based sequence-to-sequence model predict daily container fill levels compared to LSTM and BiLSTM architectures, and which model is more robust under varying synthetic waste generation patterns?
- **RQ2:** To what extent can a multi-objective genetic algorithm (NSGA-II) produce efficient waste collection routes minimizing total travel distance and service time for multi-compartment vehicles compared to Ant Colony Optimization (ACO), ACO with 3-Opt local search, and the commercial PTV routing engine?

The research was motivated by critical gaps identified in the Chapter 3: the lack of prediction models operating at daily container-level granularity, and the absence of multi-objective route optimization methods explicitly tailored for multi-compartment waste collection vehicles. By addressing these gaps, this thesis contributes towards the development of smart, cost-effective, and sustainable municipal solid waste (MSW) systems, especially for resource-constrained municipalities.

7.1.1 Summary of Approach

To tackle **RQ1**, a robust synthetic dataset was generated to simulate realistic fill-level dynamics for paper, plastic, and newspaper waste types across 354 sites in Örebro, Sweden. The synthetic data captured key temporal patterns observed in real-world operations, including weekly rhythms, holiday-induced surges, and annual seasonal trends, while introducing realistic stochastic noise. Multiple deep learning models were then developed and benchmarked, including Transformer-based sequence-to-sequence models, LSTM-based seq2seq architectures, and a convolutional BiLSTM with direct many-to-one prediction.

For **RQ2**, a customized Non-dominated Sorting Genetic Algorithm II (NSGA-II) was developed to optimize multi-vehicle routes, jointly minimizing total travel distance and total service time. The routing optimization was applied to realistic operational scenarios derived from the Örebro dataset, using Google Maps-derived distance matrices and constraint-programming pre-processing to enforce scheduling rules and vehicle capacity limits. Comparative analysis was conducted against a commercial PTV routing solver and bio-inspired heuristics such as Ant Colony Optimization (ACO) enhanced with 3-Opt local search.

7.1.2 Key Findings

The results for **RQ1** demonstrate that Transformer-based sequence-to-sequence models, when carefully scaled and equipped with location embeddings, can achieve acceptable predictive performance on synthetic daily fill-level time series. The Transformer model using Min–Max scaling achieved the best results, with a validation Mean Squared Error (MSE) of 0.0030 and a Mean Absolute Error (MAE) of 0.0333. This performance was significantly better than the BiLSTM and standard LSTM baselines, especially when generalizing to new seasonal patterns and stochastic fluctuations.

Three factors emerged as critical to the success of the Transformer model:

- **Attention Mechanisms:** Full-range attention allowed the model to learn dependencies across the entire 14-day historical window, capturing weekly cycles, seasonal effects, and holiday anomalies simultaneously.
- **Location Embeddings:** By introducing small, trainable embeddings at each time step, the model adapted flexibly to site-specific waste generation profiles, improving prediction accuracy across heterogeneous locations.
- **Min–Max Scaling:** Rescaling inputs to the [0,1] interval preserved interpretability, stabilized training, and minimized negative predictions compared to z-score scaling.

In contrast, the BiLSTM model, despite competitive training loss, suffered from higher validation errors due to its "one-shot" prediction structure, large parameter count (over 1.3 million weights), and inability to sequentially decode predictions. Moreover, random hyperparameter tuning explored less than 2% of the feasible search space and did not improve performance over manually selected configurations, emphasizing the importance of principled architectural design.

For **RQ2**, the NSGA-II algorithm successfully produced high-quality waste collection routes that outperformed the commercial baseline. Specifically, NSGA-II achieved:

- **30% reduction** in total operational time (from 285.18 hours to 189.50 hours).
- **Minimal increase** in total driven distance (only 32 km more than the baseline).

- **Smoother, more realistic routes** with fewer unnecessary detours and better stop sequencing.
- **Full compliance** with operational constraints such as waste type segregation and minimum day gaps between pickups.

Simple economic calculations show that the reduction in operational time could translate into annual savings exceeding 100,000 SEK in labor costs alone. Alternative heuristics like ACO and 3-Opt proved useful for optimizing individual routes but failed to match NSGA-II's performance in multi-vehicle planning, primarily because they lacked assignment optimization capabilities.

Scientific and Practical Contributions

This thesis makes several scientific and practical contributions:

- **Scientific Contribution:** It confirms that attention-based Transformer architectures can outperform traditional recurrent models even at the daily, container-level scale, a gap that previous literature had not addressed.
- **Practical Contribution:** It demonstrates that realistic synthetic datasets can bootstrap predictive modeling efforts without immediate access to dense sensor telemetry, offering a cost-effective pathway for small municipalities to adopt data-driven waste management.
- **Optimization Contribution:** It showcases that multi-objective evolutionary algorithms like NSGA-II can handle real-world operational constraints while producing tangible efficiency gains, even in complex, multi-compartment routing scenarios.

7.1.3 Limitations

While the outcomes are promising, several limitations must be acknowledged:

- **Synthetic Data:** Although carefully constructed, the synthetic dataset may not fully capture unexpected anomalies, vandalism, or atypical human behavior that real-world systems encounter.
- **Static Routing:** The routing optimization assumed fixed daily stop sets without dynamically adjusting to daily fill-level predictions, thus decoupling prediction from routing in this study.
- **Hyperparameter Search:** Hyperparameter tuning was limited to random search, potentially leaving room for further performance improvements through Bayesian optimization or more exhaustive searches.

These limitations point to valuable directions for future research, particularly in integrating predictive and routing modules into an end-to-end adaptive system.

7.2 Future Work

While the results of this thesis are promising, several extensions and improvements can be pursued in future research to further advance predictive waste management systems and route optimization strategies.

7.2.1 Integration of Prediction and Routing Modules

One of the most impactful next steps would be the full integration of the prediction models and the routing optimizer into a dynamic, end-to-end operational system. In the current study, fill-level prediction and route planning were treated as separate phases. Future work could dynamically generate daily pickup stop lists based on predicted fill levels, triggering optimized multi-vehicle routes accordingly. Such a system would allow municipalities to transition from fixed schedules to demand-driven collections, reducing unnecessary trips, optimizing vehicle usage, and minimizing environmental impacts.

Integrating real-time feedback from field operations (e.g., driver confirmations, ad-hoc changes) into the model updating process would further enhance robustness, enabling semi-autonomous, adaptive waste logistics systems.

7.2.2 Validation on Real-World Data

Although synthetic data allowed for controlled experimentation, the ultimate goal remains real-world deployment. Future studies should validate the predictive models and optimization pipelines using historical pickup logs. This validation would test model robustness against noise, missing data, anomalous behavior, and unanticipated operational disruptions such as road closures, holidays, or weather events.

Additionally, evaluating prediction errors and routing gains in a real operational environment would allow for a more detailed cost-benefit analysis, including factors such as fuel consumption, labor hours, and service reliability improvements.

7.2.3 Modeling Uncertainty and Risk

Another promising direction involves explicitly modeling uncertainty within both the prediction and routing stages. Current fill-level predictions are point estimates; however, predictive distributions (e.g., via quantile regression or Bayesian neural networks) would allow the system to account for variability and risk.

In routing, uncertainty-aware optimization could prioritize robustness, designing routes that remain feasible even under deviations from predicted fill levels. This would involve extending the NSGA-II framework to include objectives like worst-case service time or probabilistic feasibility under stochastic container states.

7.2.4 Advanced Hyperparameter Optimization

While random search was sufficient for this study, future work could apply more sophisticated hyperparameter tuning techniques such as Bayesian optimization, Hyperband, or neural architecture search. These methods might uncover configurations

that offer even better generalization, faster convergence, or more compact model architectures.

7.2.5 Multi-Objective Optimization Enhancements

The NSGA-II algorithm used in this study demonstrated strong performance, but several enhancements could be explored:

- **Hybridization:** Combining NSGA-II's global assignment capabilities with local search refinements from ACO or 3-Opt may yield even better results, particularly in dense urban scenarios.
- **Dynamic Replanning:** Extending the optimization to support mid-route dynamic replanning, for example, in response to unexpected delays, traffic disruptions, or last-minute service requests, would increase system resilience and real-world feasibility.
- **Energy and Emissions Objectives:** Incorporating direct minimization of fuel consumption or carbon emissions as additional objectives alongside time and distance would better align routing with broader environmental sustainability goals.
- **Vehicle Capacity Management:** Introducing vehicle load balancing as an explicit optimization objective would further improve operational reliability. Optimizing routes to prevent vehicles from reaching full capacity mid-route, which otherwise forces unscheduled diversions back to dumping facilities. This could minimize route deviations, avoid additional travel, and ensure that vehicles complete their assigned stops efficiently.

7.2.6 Broader Generalization Across Waste Types and Cities

Finally, the generalizability of the approaches developed here could be tested across different waste streams (e.g., organic, hazardous) and different urban contexts (e.g., highly dense cities, rural municipalities). Each setting presents unique logistical challenges, such as different waste generation rhythms, vehicle constraints, and road network characteristics. Extending and adapting the synthetic data generation, predicting, and optimization methods to these varied contexts would enhance the versatility and scalability of the overall framework.

In summary, future work should focus on real-world validation, dynamic integration of predictive and operational models, robust uncertainty handling, and extension toward sustainability-centric objectives. Together, these directions would move the field closer to fully autonomous, adaptive, and environmentally responsible waste management systems.

7.3 Closing Reflection

In closing, this thesis demonstrates that it is both feasible and advantageous to move toward predictive, sensor-free waste collection and multi-objective, real-world

constrained route optimization. By advancing both the forecasting and operational planning fronts, this work provides a foundation upon which future dynamic and sustainable municipal waste management systems can be built. While challenges remain, particularly in deploying these models under real-world uncertainties, the techniques developed here offer a strong starting point for operational deployment, promising reductions in costs, emissions, and resource usage and ultimately contributing to smarter and more sustainable cities.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation*, 2016, pp. 265–283.

[2] V. M. Adamović, D. Z. Antanasijević, M. Ristić, A. A. Perić-Grujić, and V. V. Pocajt, "Prediction of municipal solid waste generation using artificial neural network approach enhanced by structural break analysis," *Environmental Science and Pollution Research*, vol. 24, pp. 299–311, 2017. [Online]. Available: https://doi.org/10.1007/s11356-016-7767-x

[3] A. Alwabli and I. Kostanic, "Dynamic route optimization for waste collection using genetic algorithm," in *2020 International Conference on Computer Science and Its Application in Agriculture (ICOSICA)*. IEEE, 2020, pp. 1–6.

[4] A. Alwabli, I. Kostanic, and S. Malky, "Dynamic route optimization for waste collection and monitoring smart bins using ant colony algorithm," in *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*. IEEE, 2020, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ICECOCS50124.2020.9314571

[5] J. Bader and E. Zitzler, "Hype: An algorithm for fast hypervolume-based many-objective optimisation," *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, 2011.

[6] T. Bektas and G. Laporte, "The pollution-routing problem," *Transportation Research Part B: Methodological*, vol. 45, no. 8, pp. 1232–1250, 2011.

[7] J. Beliën, L. De Boeck, and J. Van Ackere, "Municipal solid waste collection and management problems: A literature review," *Omega*, vol. 46, pp. 1–11, 2014. [Online]. Available: https://doi.org/10.1287/trsc.1120.0448

[8] ——, "Municipal solid waste collection and management problems: A literature review," *Omega*, vol. 46, pp. 1–11, 2014. [Online]. Available: https://doi.org/10.1287/trsc.1120.0448

[9] Y. Bouleft and A. Elhilali Alaoui, "Dynamic multi-compartment vehicle routing problem for smart waste collection," *Applied System Innovation*, vol. 6, no. 1, p. 30, 2023. [Online]. Available: https://doi.org/10.3390/asi6010030

[10] T. Chai and R. R. Draxler, "Root mean square error (rmse) or mean absolute error (mae)? arguments against avoiding rmse in the literature," *Geoscientific Model Development*, vol. 7, no. 3, pp. 1247–1250, 2014.

[11] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[12] M. Dorigo and T. Stützle, "Ant colony optimization: A review," *Swarm Intelligence*, vol. 1, no. 1, pp. 1–24, 2006.

[13] F. Ghanbari, H. Kamalan, and A. Sarraf, "Predicting solid waste generation based on the ensemble artificial intelligence models under uncertainty analysis," *Journal of Material Cycles and Waste Management*, vol. 25, pp. 920–930, 2023. [Online]. Available: https://doi.org/10.1007/s10163-023-01589-9

[14] F. Glover and M. Laguna, "Tabu search and adaptive memory programming — advances, applications and challenges," *Interfaces in Computer Science and Operations Research*, pp. 1–75, 1997.

[15] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning.* MIT Press, 2016.

[16] P. Group, "Ptv route optimizer," 2025, available at: https://www.ptvgroup.com/solutions/ptv-route-optimizer/ [Accessed 29 May 2025].

[17] E. Gutierrez, M. Ferrer, A. Santana, and V. Toledo, "Smart waste collection system based on location intelligence," *Procedia Computer Science*, vol. 32, pp. 107–114, 2014.

[18] B. Gülmez, M. Emmerich, and Y. Fan, "Multi-objective optimization for green delivery routing problems with flexible time windows," *Applied Artificial Intelligence*, vol. 38, no. 1, p. e2325302, 2024. [Online]. Available: https://doi.org/10.1080/08839514.2024.2325302

[19] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, 2009.

[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[21] D. Hoornweg and P. Bhada-Tata, "What a waste: A global review of solid waste management," *Urban Development Series, World Bank*, pp. 1–116, 2012.

[22] H. Jammeli, R. Ksantini, F. Ben Abdelaziz, and H. Masri, "Sequential artificial intelligence models to forecast urban solid waste in the city of sousse, tunisia," *IEEE Transactions on Engineering Management*, vol. 70, no. 5, pp. 1912–1922, 2023. [Online]. Available: https://doi.org/10.1109/TEM.2021.3081609

[23] T. A. Khoa, C. H. Phuc, P. D. Lam, L. M. B. Nhu, N. M. Trong, N. T. H. Phuong, N. V. Dung, N. Tan-Y, H. N. Nguyen, and D. N. M. Duc, "Waste management system using iot-based machine learning in university," *Wireless Communications and Mobile Computing*, vol. 2020, pp. 1–13, 2020. [Online]. Available: https://doi.org/10.1155/2020/6138637

[24] N. Kumar, P. Rajeswari, D. J. Priya, and M. U. Maguesvari, "Forecasting municipal solid waste generation using advanced transformer and multi-layer perceptron techniques," *Clean Technologies and Environmental Policy*, 2024, in press; the paper refers to its core model as a "Lightweight GraphFormer". [Online]. Available: https://doi.org/10.1007/s10098-024-03091-8

[25] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[26] Z. Lin, S. Li, M. Li, A. H. Liu, Y. He, and J. Tang, "A survey of transformers," *arXiv preprint arXiv:2106.04554*, 2021.

[27] S. Longhi, D. Marzioni, E. Alidori, G. D. Buò, M. Prist, M. Grisostomi, and M. Pirro, "Solid waste management architecture using wireless sensor network technology," in *New Perspectives in Information Systems and Technologies*, 2012, pp. 491–498.

[28] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 1412–1421.

[29] T. Mitchell, *Machine Learning.* McGraw-Hill, 1997.

[30] K. P. Murphy, *Machine Learning: A Probabilistic Perspective.* MIT Press, 2012.

[31] S. A. Musa, R. Kora, and S. M. Elnaggar, "Smart waste management systems using internet of things (iot): A review," *Journal of Cleaner Production*, vol. 289, p. 125741, 2021.

[32] T. Nuortio, J. Kytöjoki, H. Niska, and O. Bräysy, "Improved route planning and scheduling of waste collection and transport," *Expert Systems with Applications*, vol. 30, no. 2, pp. 223–232, 2006.

[33] B. Ombuki, B. J. Ross, and F. Hanshar, "Multi-objective genetic algorithms for vehicle routing problem with time windows," *Applied Intelligence*, vol. 24, pp. 17–30, 2006. [Online]. Available: https://doi.org/10.1007/s10489-006-6926-z

[34] L. Perron, F. Didier, and S. Gay, "The cp-sat-lp solver," in *29th International Conference on Principles and Practice of Constraint Programming (CP 2023)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), R. H. C. Yap, Ed., vol. 280. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023, pp. 3:1–3:2. [Online]. Available: https://drops.dagstuhl.de/opus/volltexte/2023/19040

[35] N. L. Rachmawati, Y. A. Iskandar, D. P. Putri, and M. Lusiani, "A genetic algorithm approach for waste collection using multi-trip multi-period capacitated vehicle routing problem with time windows (mcvrptw)," in *Proceedings of the International Conference on Industrial Engineering and Operations Management (IEOM).* Manila, Philippines: IEOM Society International, 2023, pp. 736–747. [Online]. Available: N/A

[36] F. Rossi, P. van Beek, and T. Walsh, *Handbook of Constraint Programming.* Elsevier, 2006.

[37] D. Rutqvist, D. Kleyko, and F. Blomstedt, "An automated machine learning approach for smart waste management systems," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 384–392, 2020. [Online]. Available: https://doi.org/10.1109/TII.2019.2915572

[38] K. Sar and P. Ghadimi, "Web service-based capacitated smart vehicle routing problem with time window and threshold waste level for home health care industry," in *Proceedings of the 14th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2024)*, 2024, pp. 185–191. [Online]. Available: https://www.scitepress.org/Papers/2024/126400/126400.pdf

[39] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," in *IEEE Transactions on Signal Processing*, vol. 45, no. 11, 1997, pp. 2673–2681.

[40] J. Sepúlveda-Campos, P. Muñoz, and C. M. Paredes, "Inventory–routing of overfilled waste containers: Evidence from a santiago smart-bin pilot," *Waste Management*, vol. 102, pp. 120–131, 2020.

[41] M. Shi, W. Zhang, and H. Li, "Transformer-based deep learning model for accurate prediction of hazardous waste generation in shanghai," *Chemosphere*, vol. 329, p. 139579, 2023. [Online]. Available: https://doi.org/10.1016/j.chemosphere.2023.139579

[42] Streamlit Inc., "Streamlit: The fastest way to build and share data apps," 2025, accessed: 2025-04-25. [Online]. Available: https://streamlit.io

[43] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems*, vol. 27, 2014, pp. 3104–3112.

[44] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Łukasz Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998–6008.

[46] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance," *Climate Research*, vol. 30, no. 1, pp. 79–82, 2005.

[47] H. Wu, F. Tao, and B. Yang, "Optimization of vehicle routing for waste collection and transportation," *International Journal of Environmental Research and Public Health*, vol. 17, no. 14, p. 4963, 2020. [Online]. Available: https://doi.org/10.3390/ijerph17144963

[48] H. Wu, W. Cao, Y. Zhuang, Y. Zhang, and J. Yan, "Deep transformer models for time series forecasting: A survey," in *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, 2020, pp. 4423–4430.

[49] Y. Zhou, Q. Liu, and Y. Wang, "A waste extended input-output-based transformer-lstm method for analyzing hazardous waste reduction patterns: A case study of shanghai," *Resources, Conservation and Recycling*, vol. 203, p. 107077, 2024. [Online]. Available: https://doi.org/10.1016/j.jclepro.2024.142435

[50] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach," *IEEE Transactions*

on Evolutionary Computation, vol. 3, no. 4, pp. 257–271, 1999. [Online]. Available: https://ieeexplore.ieee.org/document/797969

Faculty of Computing, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden