Prediction of delivery and pickup times and routes in urban areas using spatio-temporal neural networks and basic reinforcement learning

MSc Research Project
Master of Science in Data Analytics

Hugo Francisco Alejo Cárdenas
Student ID: x23323442

School of Computing
National College of Ireland

Supervisor: Qurrat UI Ain

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Hugo Francisco Alejo Cárdenas
Student ID:	x23323442
Programme:	Master of Science in Data Analytics — **Year:** 2024
Module:	MSc Research Project
Supervisor:	Qurrat Ul Ain
Submission Due Date:	11/08/2025
Project Title:	Prediction of Delivery and Pickup Times and Routes in Urban Areas Using Spatio-Temporal Neural Networks and Basic Reinforcement Learning
Word Count:	8,757 — **Page Count** 23

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Hugo Francisco Alejo Cárdenas

Date: 11/08/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Prediction of delivery and pickup times and routes in urban areas using spatio-temporal neuronal networks and basic reinforcement learning

Hugo Francisco Alejo Cárdenas
X23323442

Abstract

This thesis investigates how well spatio-temporal neural networks may help to better forecast pickup and delivery times and routes in cities and examines the use of basic reinforcement learning to maximize these choices. The study examines advanced models including Spatio-Temporal Graph Neural Networks (STGNN), hybrid LSTM with residual learning, and Long Short-Term Memory (LSTM) networks on actual metropolitan logistics data from five Chinese cities. With predictions confirmed via standard error measurements, a modular system was created to process, organize, and forecast trip times between back-to-back logistics nodes. Though not totally carried out inside the purview of this study, an offline reinforcement learning experiment was also conceived to choose routes according to these predictions. Results show that hybrid LSTM models do well in places where things aren't normal in cities, but STGNN models do better in cities where the roads are organised. This research provides insightful information for the best possible urban logistics and helps to expand the body of knowledge about how spatio-temporal neural networks and reinforcement learning can be used together in last-mile delivery systems.

1 Introduction

This thesis investigates the question: can spatio-temporal neural networks improve the prediction of delivery and pickup times and routes in urban areas, and how can basic reinforcement learning optimize these decisions?

The general objective is to evaluate spatio-temporal models for predicting delivery and pickup times and routes, and to explore the potential of basic reinforcement learning for improved decision-making. Specific objectives include: (a) processing urban delivery and pickup data to build a spatio-temporal prediction model; (b) developing and training models capable of accurate time and route predictions; (c) implementing a basic offline reinforcement learning experiment for route selection based on these predictions; and (d) comparing the performance of standalone prediction models with combined prediction and reinforcement learning approaches. Using real-world data from five Chinese cities, this work develops a modular architecture that structures logistic trips into temporal sequences and dynamic graphs to train LSTM-based and graph-based models. The research highlights the strengths of hybrid LSTM models with residual learning in complex, irregular urban contexts, while STGNN models demonstrate advantages in cities with regular road networks. Although reinforcement learning was not fully integrated, the system design facilitates future work in this area.

This thesis contributes to both academic and practical fields by providing a robust framework for urban logistics prediction and by exploring the intersection of spatio-temporal neural networks and reinforcement learning in last-mile delivery optimization.

2 Related Work

In recent years, the growth of e-commerce, along with accelerated urbanization and the pressure to offer increasingly faster and more efficient deliveries, has made last-mile logistics a key point of innovation. Within this context, some research has begun to explore ways to anticipate routes and delivery times by combining models that work with spatial and temporal information and using reinforcement learning techniques to improve decisions. This chapter critically reviews these proposals, analysing the extent to which they could be applied to predict delivery and pickup routes and times in urban environments, in line with the objectives of the thesis.

2.1 Prediction Models and Machine Learning in Urban Logistics

Although traditional methods, such as heuristics or nonlinear regressions, have proven useful, they present significant limitations when facing the great fluctuation of urban environments. The author (Zhou *et al.*, 2023) shows that CNNs can identify spatial patterns in urban maps to predict traffic. Similarly,(Zhao, Xing and Mao, 2022) and (Shi and Du, 2022) demonstrate that the combination of CNN, XGBoost, and LSTM achieves superior accuracy compared to classic methods.

This trend is also observed in studies where hybrid models are applied to urban service data. However, in most cases, the spatial and temporal components are addressed separately, which reduces the capacity of these models to generalize in environments characterized by constant changes.

2.2 LSTM and Spatiotemporal Models

The author (Mahajan, Pottigar and Fahad, 2025) highlights the use of LSTM networks, which were created to recognize long-term relationships in data that change over time. A practical example is found in (Wang *et al.*, 2025) where an LSTM is used to calculate the estimated time of arrival (ETA) by combining traffic and location information.

To improve these capabilities, other versions have been tested, as noted in (Crivellari *et al.*, 2022), including GRUs, which are usually faster and perform better when sequences are short or when data changes frequently. On the other hand, (Bing Yu, Yin and Zhu, 2018) study TCN networks (Li *et al.*, 2023) as an alternative that can be executed more in parallel than LSTMs, although they do not reach the same level of performance when the spatial aspect of the data is very important, as is the case in urban logistics.

2.3 Graph Neural Networks and STGNN

Spatio-Temporal Graph Neural Networks (STGNN) have become increasingly important because they handle spatial and temporal information simultaneously, overcoming the limitation of models that analyze each dimension separately. An example of early developments is the DCRNN, which combines GRUs with directed graphs to predict traffic conditions. The flexibility of this type of network is evident in works like (Shi *et al.*, 2025) where GCNs are used to anticipate demand at delivery points, or (Ieva et al., 2025), where GATs help manage fleet movement in real time. Among the most well-known models is STGCN, described in (Yuan, Li and Ji, 2021) which has been widely replicated because it achieves a good balance between accuracy and complexity. Added to this is ASTGCN, explained in (Guo *et al.*, 2025) which incorporates spatial and temporal attention mechanisms to focus analysis on important events and key zones.

2.4 Reinforcement Learning (RL) in Logistics and Routing

Reinforcement learning (RL) has greatly changed the way in which logistics processes are optimized. The DQN model, explained in (Adi, Iskandar and Bae, 2020) was one of the first to pave the way for many of the systems that exist today. Its usefulness can be seen in cases such as (Zhang *et al.*, 2020), where DQN is adapted for dynamic vehicle routing, or (Liang et al., 2024), where it is used to manage fleet operation.

Another interesting approach is combining RL with supervised models, as in (Shokrollahi and Dehghan, 2023) where it is applied to dynamically assign orders. The challenge is that training these systems directly in real-world scenarios is not always safe, particularly due to the exploration issue. To address this, research has been conducted on offline reinforcement learning. In (Mahajan, Pottigar and Fahad, 2025) BCQ is presented as a method designed to learn from previously recorded data, without needing to interact with the environment in real time, which is highly useful in urban logistics.

2.5 Integration of STGNN and Reinforcement Learning

A recent trend in research is the combination of STGNNs for prediction tasks with reinforcement learning for decision-making. For example, (Jin *et al.*, 2022) present a system where the predictions from the model serve as input for an RL agent to choose the best route.

Coordination between these two modules is key, and (Jiang and Feofilova, 2025) analyze how synchronization between them influences system stability. Additionally, (Qi *et al.*, 2022) show that by integrating a GAT network to predict demand with a DQN model, delivery times can be reduced by up to 18% in urban areas.

2.6 Gaps in the Literature

Despite the advances that have been achieved, there are still several important gaps to be covered:

- On the one hand, very little has been explored about combining STGNN with offline reinforcement learning in real applications.
- Also, the availability of public and realistic datasets is limited, which makes it difficult to validate the models outside the environments where they are developed.
- There is also a lack of specific research in Latin American urban contexts, where infrastructure and mobility patterns are different from those that have been more commonly studied.

- On the other hand, many of the existing models require a lot of computing power and are not designed to perform well under real conditions or on mobile devices.

This thesis aims to help close some of these gaps, analyzing how to use STGNN to make predictions and applying a simple offline reinforcement learning model to improve decision-making in routing.

Table 1: Comparative Table

Model	Type	Advantages	Limitations
LSTM	Temporal	Captures long-term dependencies	Computationally expensive
Bi-LSTM	Temporal	Bidirectional, higher accuracy	More parameters
GRU	Temporal	More efficient than LSTM	Lower long-term capacity
TCN	Temporal	Parallelizable, efficient	Does not model spatial relationships
DCRNN	STGNN	Excellent accuracy in traffic	Expensive and dependent on topological data
STGCN	STGNN	Accuracy and robustness	Intensive training
ASTGCN	STGNN	Spatial and temporal attention	Computationally demanding
DQN	RL	Sequential learning, flexible	Requires simulation
PPO	RL	Stability and convergence	Complex in real-world environments
BCQ	Offline RL	Uses historical data, avoids real exploration	Sensitive to policy design
STGNN + RL	Hybrid	Prediction + decision integration	Still experimental

3 Research Methodology

This chapter outlines the step-by-step process followed during the research to develop models aimed at predicting and optimizing delivery and pickup logistics operations in the city. The work adopts a quantitative approach, using real-world data and modern artificial intelligence techniques such as spatiotemporal neural networks and reinforcement learning methods. The main goal is to understand how much these tools can improve time and route prediction, and whether even simple versions of reinforcement learning can support better decision-making in daily operations. The methodological design is quantitative and experimental, based on the analysis of historical data from real operations in five Chinese cities: Shanghai, Hangzhou, Chongqing, Jilin, and Yantai. This approach was chosen because logistics data have a sequential and geospatial nature, which requires models capable of capturing complex patterns in both time and space. For prediction tasks, Long Short-Term Memory (LSTM) models were used due to their proven effectiveness in time series. Spatiotemporal graph neural networks (STGNN) were also incorporated to better capture structural and spatial relationships between urban nodes. Finally, a basic reinforcement learning (RL) approach was added to simulate and optimize routing decisions, completing a robust and hybrid methodological framework.

3.1 Data Collection and Preprocessing

The research uses three main types of data. Geospatial data are stored in .csv files containing road geometries in WKT format, representing the city's street and road network. Logistics data come from historical .parquet records of real delivery and pickup operations, including timestamps, locations, node sequences, duration, operation type, and couriers. Additionally, derived temporal variables were computed, such as time of day, day of the week, time slots, and weekend indicators.

The dataset covers five cities and contains complete records for both deliveries and pickups. Its level of detail allows for individual trajectory analysis, which provides a strong foundation for building sequential and graph-based models. The information comes from the LaDe (Large-scale Dataset for Urban Last-mile Delivery)[1] dataset, published by Cainiao-AI and available on the Hugging Face platform. This source compiles a large amount of real-world logistics data from several Chinese cities, enabling training and validation with high realism and representativeness.

Geospatial files were transformed from EPSG:3857 to EPSG:4326 (WGS84) to ensure compatibility with web-based tools and spatial analysis systems. City names were translated from Chinese to English, and interactive maps were created using Folium to visually inspect data quality. Logistics operations were cleaned through several steps: time columns were converted to datetime format, irrelevant columns were removed, routes with fewer than two nodes were filtered, new temporal variables were derived, and nodes were encoded as latitude-longitude tuples with unique identifiers. Cleaned data were exported in .parquet format.

Feature engineering included generating dummy variables for time slots, organizing node sequences for temporal analysis, and ensuring the consistency of trajectory structures. These steps prepared the data for use in both sequence-based and graph-based models.

3.2 Predictive Modelling

3.2.1 LSTM and Hybrid Residual Adjustment

LSTM models were trained individually for each city and each task type (delivery or pickup), using enriched temporal sequences as input. Inputs combined temporal and logistics variables, and the output was the estimated time for each operation. Mean Squared Error (MSE) was used as the loss function, and evaluation metrics included Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and the R^2 coefficient. Early stopping, batch processing, and masking for missing data were applied as training strategies.

In a second phase, residual adjustment models were introduced. New LSTM models were trained to predict the errors of the base model, using normalized sequences of fixed length

[1] https://huggingface.co/datasets/Cainiao-AI/LaDe

created through truncation and padding. Bidirectional models were trained per city and task type, and the original predictions were corrected based on the estimated residuals. This hybrid strategy improved performance without redesigning the base model.

3.2.2 Spatiotemporal Graph Neural Networks (STGNN)

Structured data were represented as graphs using PyTorch Geometric, where each node represented a location and the edges represented temporal transitions between them. The process included loading and filtering relevant columns, dividing the data into training, validation, and testing groups while ensuring that couriers did not overlap across splits, scaling features with StandardScaler, and converting the data into graphs with special handling of missing nodes.

The STGNN model was defined using GATConv layers and MLP blocks. It was trained using the Adam optimizer and validated frequently with early stopping to prevent overfitting.

3.3 Evaluation and Comparative Analysis

The models were evaluated using standard metrics: MAE, RMSE, MAPE, and R^2. These metrics were applied to all predictive models and compared against the results obtained from the reinforcement learning experiment.

A comparative analysis was conducted between the LSTM models, the hybrid residual adjustment model, and the STGNN architecture. The models were evaluated based on their prediction accuracy, generalization capacity, and error patterns. In addition, the reinforcement learning approach was assessed for its effectiveness in route optimization and reduction of operational times.

3.4 Strengths, Limitations, and Visualization

The suggested approach has several advantages that render it adaptable and technically strong. Combining spatial, temporal, and logistical data in a unified way, its modular design lets modifications match every task type and every city. Including graph-based models lets one depict difficult inter-node connections, while residual error correction improves prediction accuracy without demanding a whole redesign of the base model. The framework also looks at how to make better decisions by using a simple type of reinforcement learning. This means that it puts a layer for making decisions on top of the part that predicts things.

The approach does, nonetheless, have some drawbacks. Multiple models need a lot of computer power to train, and how well they do depend a lot on how good and how much old data they have. Furthermore, the reinforcement learning module needs careful calibration and adaptation to work effectively in real-world situations.

Though a whole Offline RL (Offline Reinforcement Learning) framework was not applied precisely, a basic idea from this approach was included in the modelling process. The core idea of policy refinement based on historical data, which is a key feature of Offline RL, is the inspiration for the hybrid LSTM model with residual adjustment. By learning from past prediction errors and fixing them without having to go out into the real world and explore, the model mimics the way that Offline RL improves decisions by using only data that has already been logged. Given the sensitivity of Offline RL to policy design and distribution shift, this technique provided a more controlled, computer-efficient, and interpretable alternative. Offline RL was not used in its original form because it was hard to use and unstable in real-world situations. However, its basic logic helped to guide the residual correction method that was used in this study.

The following graphs are recommended to help the methodological explanation be clear and supported. First, draw.io or Lucidchart are advised for a basic flow chart showing the whole pipeline from data gathering to last evaluation. Second, matplotlib or Netron can be used to make a diagram of the STGNN model architecture that shows which nodes are important, which layers are GAT layers, and which connections are likely to happen. Finally, draw.io can be used to make a conceptual diagram that shows how the reinforcement learning cyclestate, action, reward, and agent update work. This will help people understand how the model learns.

4 Design Specification

Grounded in the application of deep learning models for predicting travel time between metropolitan logistic nodes, the technical design specification of this research is based on a modular and task-oriented approach. The overall design is intended to directly answer the research question by concentrating on how time estimation for urban delivery and pickup routes may be improved by sequential and spatiotemporal neural networks.

Unlike worldwide solutions that forecast the overall time of a trip as one variable, this proposal concentrates on step-by-step sequential prediction that is, the model predicts the time between each pair of adjacent nodes, therefore providing a fine reconstruction of the entire path. This method offers more flexibility, helps pinpoint bottlenecks, and enhances resilience to many metropolitan layouts.

Furthermore, a clear separation between delivery and pickup activities is created, needing totally separate datasets, training procedures, and validation workflows for each. This choice addresses the need for expertise since logistic and temporal behaviour patterns vary greatly across the two activities.

4.1 Core Technical Components

4.1.1 LSTM (Long Short-Term Memory) Models

LSTM networks are a type of recurrent neural network designed to capture long-term dependencies in sequential data. In this project, they are used to predict the time between consecutive nodes in a logistics route. Each trip is represented as a sequence enriched with temporal and logistical variables and fed into the model for step-by-step time estimation.

A separate LSTM model is trained for each city and task type (delivery or pickup), allowing for specific adaptation to local conditions.

4.1.2 Hybrid LSTM Models

To improve the accuracy of sequential models without changing their core architecture, a second residual prediction stage is added. This involves another LSTM model that learns to estimate the error made by the initial prediction. The final result is obtained by adjusting the original predictions with the learned residuals, which demonstrates significant improvements in accuracy without compromising system stability.

4.1.3 STGNN (Spatio-Temporal Graph Neural Network) Models

This type of model allows logistic trips to be represented as dynamic graphs, where nodes represent locations (intersections, pickup/delivery points), and edges reflect temporal transitions between them. STGNNs are particularly suited to capturing both spatial relationships and temporal dynamics within an urban environment.

The networks are implemented using the PyTorch Geometric library and constructed with GATConv (Graph Attention Convolutional) layers, which allow the model to weigh the relative influence of each neighbouring node during prediction.

4.2 System Requirements

From a technical standpoint, the system is implemented locally using Python 3.10 as the primary programming language. The tools and libraries used include:

- Data processing: pandas, numpy, geopandas, shapely
- Visualization and exploratory analysis: matplotlib, seaborn, folium
- Predictive modelling: PyTorch, PyTorch Geometric, scikit-learn
- Development environment: Jupyter Notebooks
- Computational capacity: High-performance CPU with optional GPU support to accelerate STGNN training

The modular design, separation by city and operation type, and spatiotemporal focus make this solution an adaptable, scalable, and robust platform for granular time prediction in urban logistics environments.

5 Implementation

The final development phase focused on the implementation, training, and evaluation of the proposed models for predicting travel time between nodes in urban logistics trips. This process was carried out independently for each city and task type (delivery and pickup), in accordance with the previously defined modular design. The resulting system is capable of making precise, detailed predictions at the route segment level and is ready to be integrated into logistics planning or simulation solutions.

5.1 Data Preparation and Trip Structuring

Data from the LaDe[2] dataset (logistics operations in five Chinese cities) was cleaned, transformed, and enriched by standardizing timestamps, removing incomplete records, encoding geospatial nodes, assigning unique IDs, creating temporal features, and separating delivery and pickup trips. The data was formatted into ordered sequences for LSTM models and dynamic graphs for STGNN models.

[2] https://huggingface.co/datasets/Cainiao-AI/LaDe

5.1.1 Training of LSTM and Hybrid LSTM Models

LSTM models were trained on enriched temporal sequences using techniques like batch training, early stopping, and dropout. A secondary LSTM modelled prediction residuals, improving accuracy by capturing predictable error patterns.

5.1.2 Implementation of STGNN

Trips were represented as spatiotemporal graphs using PyTorch Geometric, with nodes as geographic points and edges as temporal transitions. GATConv layers prioritized relevant neighbour information. Models were trained with Adam optimizer and early stopping to avoid overfitting, evaluated on separate test sets.

5.1.3 Evaluation and Results

Models were assessed with MAE, RMSE, MAPE, and R^2 metrics. STGNN performed better in cities with regular road networks (Shanghai, Hangzhou), while Hybrid LSTM excelled in more irregular cities. Jilin and Chongqing posed greater challenges due to complex logistics.

5.1.4 Exclusion of Reinforcement Learning

Though reinforcement learning for route optimization was considered, it was not implemented. The system, however, is designed to allow future integration of such approaches.

6 Evaluation

The purpose of this section is to analyse in detail, as well as rigorously, the performance of the prediction models that have been developed for the completion of this thesis, with the objective of estimating pickup and delivery times with a main focus on urban logistics scenarios. For this reason, four different architectures have been implemented and evaluated:

- LSTM (Long Short-Term Memory) model, which will be our baseline
- Adjusted version of the LSTM (Long Short-Term Memory) baseline model
- LSTM + RL (Residual Learning) model, which is an approach inspired by Reinforcement Learning
- STGNN (Spatio-Temporal Graph Neural Network) model, applied over the edges of a network

This evaluation considers different cities, each with different logistical characteristics (Chongqing, Yantai, Shanghai, Hangzhou, and Jilin), focusing on last-mile logistics for deliveries and pickups along different routes.

To ensure a strictly professional evaluation, the metrics to be used will include Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and the Coefficient of Determination (R^2). Additionally, a residual analysis will be performed, including the mean residual, median, standard deviation, and skewness, with the aim of identifying bias and dispersion patterns in our data.

6.1 Presentation of Results

6.1.1 LSTM Baseline Model

A baseline LSTM model offers performances that could be considered heterogeneous given the dependence on each city as well as the task to be performed. However, in general, this model has achieved MAE values between 2.98 minutes, recorded in the city of Hangzhou in pickup tasks, and 11.28 minutes, recorded in the city of Chongqing in delivery tasks. Regarding the values of the R^2 coefficient, the model has shown a fairly reasonable explanatory capacity, reaching up to 0.9621 for the city of Yantai based on delivery behaviours, and a minimum of 0.8497 in the city of Chongqing for delivery operations.

The model shows a strong tendency to underestimate time, generating negative mean residual values that result in positively skewed distributions. The summary table that explains this data is shown in Table 2.

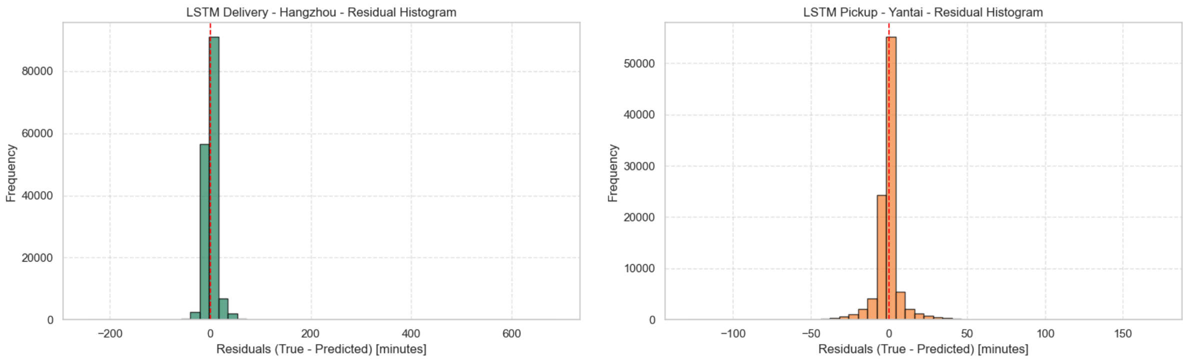
Table 2: Final evaluation of the model on the test set, with MAE, RMSE, MAPE, and R^2 metrics by city and type of movement. (LSTM Model)

City	Movement	MAE	RMSE	MAPE (%)	R^2
Chongqing	Delivery	11.28	21.55	63.34	0.8497
Chongqing	Pickup	5.19	9.03	30.58	0.9432
Yantai	Delivery	5.68	9.85	33.80	0.9621
Yantai	Pickup	4.50	8.43	25.51	0.9458
Shanghai	Delivery	7.86	13.85	57.15	0.8722
Shanghai	Pickup	3.40	6.98	23.80	0.9558
Hangzhou	Delivery	6.08	11.44	45.55	0.9286
Hangzhou	Pickup	2.98	5.93	22.18	0.9352
Jilin	Delivery	7.89	12.00	58.85	0.9385
Jilin	Pickup	7.67	12.22	51.50	0.8967

To illustrate the general tendency of the model to underestimate time, as well as the symmetrical distribution of these residuals, two representative examples have been selected:

A. Deliveries in Hangzhou: In this specific case, the model shows a clear tendency to underestimate time, given a residual mean very close to zero (0.213 minutes). However, a negative median (-1.316 minutes) indicates that more than half of our predictions are underestimating the actual time. The distribution presents a highly pronounced positive skew (skewness = 4.577), which implies the presence of some large errors leaning towards overestimation, although negative errors still predominate.

B. Pickups in Yantai: For this example, we observe a completely different behaviour. The residuals are centred very close to zero, with a residual mean very close to zero (-0.729 minutes) and a median also negative but much smaller compared to the previous city (-0.641 minutes), which suggests that our model does not present a strong bias in this city and movement. Regarding the distribution, it is defined by a slight positive skew (skewness = 0.927), indicating a mild asymmetry resulting in some slightly large errors leaning more towards overestimation. However, in general, these errors remain balanced.



(A) (B)

Figure 1: Residual Histogram (LSTM Model)

The probable cause of the observed tendency of the model to underestimate delivery times, as well as the heterogeneity in performance between one city and another, may be closely related to external factors not incorporated into our LSTM model. These factors may include traffic congestion, weather, and even the availability and quality of city-specific data, which could significantly influence the behavior of our model.

In previous studies, such as Implementing Real-Time Traffic Flow Prediction Using LSTM, it has been identified that LSTM models are prone to difficulties when capturing external events or patterns that are heterogeneous and present in temporal data, which is consistent with our current observations. It is worth noting that the hypotheses presented here were not directly explored in the current study. However, they are based on the results obtained, as well as on existing literature. Their consideration seems relevant for improving future versions of the model and ensuring more accurate application in operational contexts. As a consequence of this, the following model evaluates an alternative designed to mitigate the aforementioned limitations and thereby improve prediction accuracy.

6.1.2 Adjusted LSTM Model

This model presents some improvements in accuracy and stability compared to the baseline model, which allows us to observe heterogeneous performances depending on the quality of the city and the task. If we look at the MAE values, they range between 0.87 minutes in the city of Hangzhou for pickups and 10.62 minutes in Chongqing for delivery tasks. With this, the R^2 coefficient values reflect a high explanatory capacity, giving us a maximum of 0.9867 in Yantai for deliveries and a minimum of 0.8398 in Chongqing for deliveries.

The model still shows a clear tendency to underestimate time, which is evidenced by residuals with negative means and distributions with positive skew. However, it features less bias and more balanced errors in pickup movements. These improvements suggest that with small adjustments in training parameters, we can enhance the robustness of our model without making major modifications to the baseline architecture.

Table 3: Final evaluation of the model on the test set, with MAE, RMSE, MAPE, and R^2 metrics by city and type of movement. (Adjusted LSTM Model)

City	Movement	MAE	RMSE	MAPE (%)	R^2
Chongqing	Delivery	10.62	22.25	52.38	0.8398
Chongqing	Pickup	1.73	4.45	8.72	0.9862
Yantai	Delivery	2.34	5.83	11.92	0.9867
Yantai	Pickup	1.69	4.40	8.53	0.9852
Shanghai	Delivery	7.26	13.94	44.30	0.8705
Shanghai	Pickup	1.26	4.21	6.65	0.9839
Hangzhou	Delivery	5.67	11.60	37.61	0.9266
Hangzhou	Pickup	0.87	2.99	5.29	0.9835
Jilin	Delivery	5.29	9.63	30.64	0.9604
Jilin	Pickup	3.65	7.66	18.53	0.9594

By analysing the residuals of the adjusted model, we can observe that the general tendency to underestimate times persists, especially in delivery scenarios, where the mean values of the residuals are mostly positive yet very close to zero with negative means, which could indicate a slight inclination toward underestimation. Despite this, there are several cases where the magnitude of the bias is low. To illustrate these contrasts, two representative scenarios are presented.

A. Deliveries in Chongqing: For this case, one of the scenarios with the greatest bias in the model is represented. The residual mean (1.373 minutes) as well as the negative median (-1.490 minutes) indicate that most of the generated predictions tend to underestimate the actual time. However, the high standard deviation (22.207 minutes) and the extreme positive skew (skewness = 6.023) may reflect the presence of some exceptionally large overestimation errors, which significantly widen the dispersion of our residuals.
B. Pickups in Hangzhou: In this city, we can observe a contrast, where the model shows virtually no bias (0.345 minutes), and residuals are centred around zero (0.148 minutes), resulting in a lower standard deviation (2.974 minutes). Despite these results, the skew although positive (skewness = 5.362) suggests that, although rare, large errors tend to lean toward overestimation.

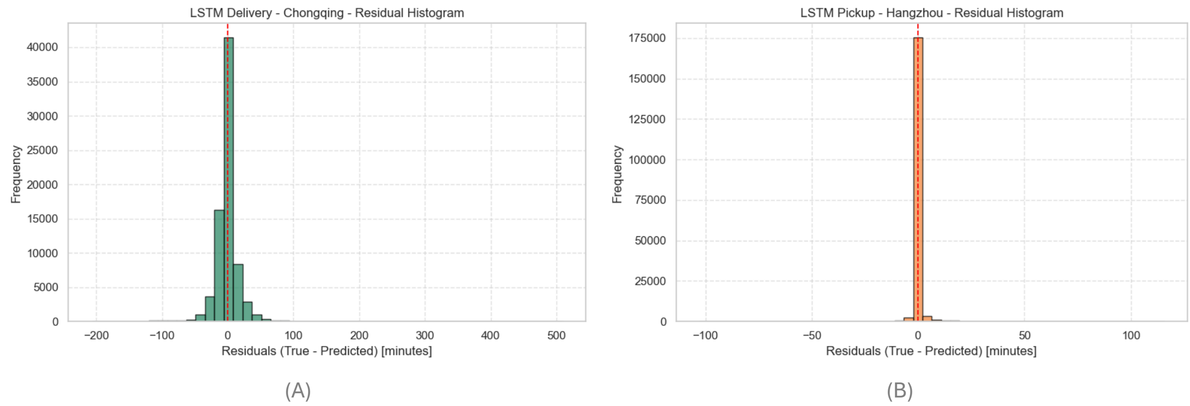


Figure 2: Residual Histogram (Adjusted LSTM Model)

In this adjusted LSTM model, although it manages to maintain part of the general trend observed in the previous model (baseline), especially in delivery scenarios, the differences in performance between each city once again seem to respond to the previously mentioned external factors, which are not explicitly included in the training. Among these, we can highlight the variability in traffic congestion, weather conditions, and even the representative quality of the data collected by each city, which very likely continue to influence the ability of our model to generalize in a homogeneous manner.

This adjustment, although it does not represent a profound redesign, has introduced strategic changes in the training parameters in order to explore what would happen if changes were applied to make the model more robust, resulting in a reduction of the identical limitations found in the baseline model and contributing improvements in prediction accuracy.

6.1.3 LSTM + Residual Learning Model

The model combines the sequential capabilities of an LSTM model with the learned correction of residuals, which results in an overall improvement in performance in terms of R^2 ranging from (0.8927) in Chongqing for deliveries up to (0.9878) in deliveries in the city of Yantai. The MAE has also been positively impacted under the conditions of this model, which has seen a notable change of 1.07 minutes for pickups in the city of Hangzhou, 1.39 minutes for deliveries in Yantai, and finally 1.89 minutes for pickups in Shanghai. Despite this, the tendency to underestimate persists, although the residuals are more centred and closer to zero, especially in pickup tasks, leading to a positive skew in the residuals, which remains, indicating long tails toward overtime errors.

Table 4: Final evaluation of the model on the test set, with MAE, RMSE, MAPE, and R^2 metrics by city and type of movement. (LSTM + Residual Learning Model)

City	Movement	MAE	RMSE	MAPE (%)	R^2
Chongqing	Delivery	7.42	18.20	34.74	0.8927
Chongqing	Pickup	2.27	5.01	14.82	0.9825
Yantai	Delivery	4.32	7.91	25.76	0.9755

Yantai	Pickup	1.39	4.00	9.09	0.9878
Shanghai	Delivery	4.70	10.75	28.64	0.9230
Shanghai	Pickup	1.89	4.79	12.40	0.9792
Hangzhou	Delivery	2.74	8.11	16.16	0.9641
Hangzhou	Pickup	1.07	2.64	8.84	0.9871
Jilin	Delivery	3.88	8.09	22.19	0.9720
Jilin	Pickup	4.19	7.80	25.27	0.9579

To illustrate the improvement, as well as the behaviour of the LSTM + Residual Learning model, along with the persistence of the tendency to underestimate time and the characteristics of the residual distribution, two representative examples have been selected.

A. Deliveries in Chongqing: In this specific case, our model shows a clear tendency to underestimate the time, resulting in a positive mean residual (1.082 minutes), which could indicate that, strictly speaking, on average the prediction is slightly below the real time. Despite this, the residual median in this city remains negative (-0.074 minutes), indicating that, in general, more than half of our predictions tend to underestimate the real time. Regarding the distribution of residuals, it presents a marked positive skew (skewness = 6.188), which indicates the presence of some significant errors leaning toward overestimation, although negative errors still predominate. This reflects that, despite the improvements in our residual learning adapted to the predictions of the baseline LSTM model in an attempt to correct them, the model continues to face difficulties in capturing certain extreme events or significant variations.

B. Pickups in Shanghai: In this example, the model presents centred residuals that are close to zero, producing a mean residual (0.178 minutes) and a median of (0.011 minutes), which indicates a low inclination toward a strong bias. This distribution of residuals becomes practically symmetrical (skewness = 2.215), showing a balance between overestimation and underestimation errors, which suggests that residual learning helps correct certain biases and improves the prediction accuracy for this city and task, resulting in a better fit compared to the baseline model.

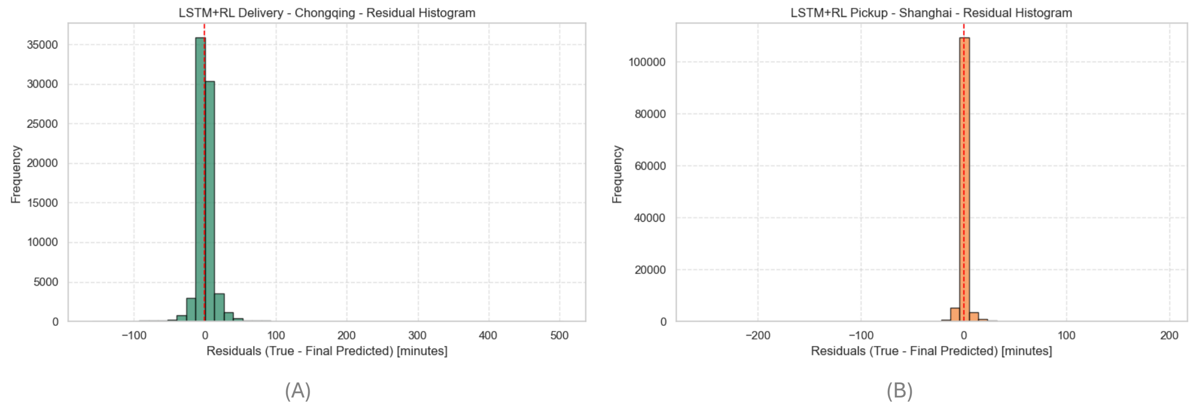


Figure 3: Residual Histogram (LSTM + Residual Learning Model)

An important point to mention in the evaluation of this model is that, although the baseline LSTM model was adjusted through residual learning, no external variables that could affect the prediction process were explicitly incorporated. Nevertheless, the results obtained show a significant improvement, consistent with the evaluation metrics compared to the baseline model. The improvement is not only reflected in the reduction of the Mean Absolute Error (MAE) or the Root Mean Square Error (RMSE) but can also be seen in the reduction of the bias observed in the residuals.

The addition of the residual learning technique has allowed the model to systematically correct the deviations of our baseline model, managing to capture with the residuals a greater number of complex and non-linear patterns that would otherwise go unmodeled. Thanks to the residual learning's capability, we can infer that it provides an effective mechanism to complement or strengthen time series prediction, especially in datasets where variability is high and the underlying relationships are difficult to model using a single approach.

Despite the progress achieved with this model, limitations still persist, many of which are significant and restrict the ability to generalize uniformly across all evaluated cities. As mentioned in other models, the absence of critical contextual variables that directly impact actual delivery and pickup times remains a limitation. Moreover, the heterogeneity in the quality and quantity of data available for each city contributes to the variability in model performance. For this reason, an important part of this work is to suggest that future research efforts should be directed toward incorporating these variables.

The improvements introduced to the baseline LSTM model with residual learning have direct and highly relevant implications for logistics management and planning within the urban context. Achieving greater accuracy in predicting delivery or pickup times enables better resource allocation, reduces operational costs, and enhances customer satisfaction by minimizing delays and allowing for proactive planning.

6.1.4 STGNN (Prediction on Edges)

For the final model, STGNN, its ability to explicitly model spatial relationships through graphs stands out. Although in this case it did not outperform the LSTM + RL model across all tasks, it offered highly competitive performance. As an example, in deliveries in Shanghai and Hangzhou, it reached an R^2 of 0.9309 and 0.952 respectively. Unlike the previous models, this one tends to overestimate delivery times, with positive mean residuals in all tasks, and distributions that are positively skewed.

Table 5: Final evaluation of the model on the test set, with MAE, RMSE, MAPE, and R² metrics by city and type of movement. (STGNN Model)

City	Movement	MAE	RMSE	MAPE (%)	R^2
Chongqing	Delivery	8.61	12.63	49.00	0.9328
Chongqing	Pickup	4.84	7.41	33.26	0.9552
Yantai	Delivery	5.17	7.95	35.51	0.9381
Yantai	Pickup	4.75	7.36	34.23	0.9522
Shanghai	Delivery	5.89	9.45	42.71	0.9309
Shanghai	Pickup	3.80	6.21	31.08	0.9623
Hangzhou	Delivery	4.94	8.29	39.41	0.9520
Hangzhou	Pickup	3.12	5.01	27.67	0.9518
Jilin	Delivery	5.73	8.92	36.65	0.8708
Jilin	Pickup	5.51	8.28	38.68	0.9430

To illustrate the behaviour of the STGNN model and its general tendency to overestimate time, as well as the characteristics observed in each of the residual distributions, we present two key representative examples:

A. Deliveries in Chongqing: In this specific case, the model shows a clear tendency to overestimate delivery time, reflected in a negative mean residual (-1.028 minutes) as well as a negative median residual (-1.907 minutes). These metrics indicate that, on average, the predictions are consistently higher than the actual delivery times, which can be considered a conservative behaviour in operational-level scenarios. The distribution of residuals displays a moderately positive skewness (skewness = 1.149), suggesting the presence of some significant errors oriented toward underestimation, although overestimations remain predominant. This behaviour may stem from the model's limitations in capturing certain complex events or patterns in the data, highlighting areas for improvement that could be addressed in future work.

B. Pickups in Shanghai: This example exhibits a similar behaviour in terms of residual metrics, with a mean residual of -0.325 minutes and a median of -1.049 minutes, indicating that predictions generated by this model tend to slightly overestimate the actual pickup times. The residual distribution presents a notably positive skewness (skewness = 2.548), pointing to the presence of outliers or errors increasingly leaning toward underestimation. However, the relatively low standard deviation (6.205 minutes) suggests greater consistency in the predictions compared to other cities. This may reflect a better model capacity in adjusting to actual times when variability is lower, though there are still opportunities to reduce bias and improve precision.

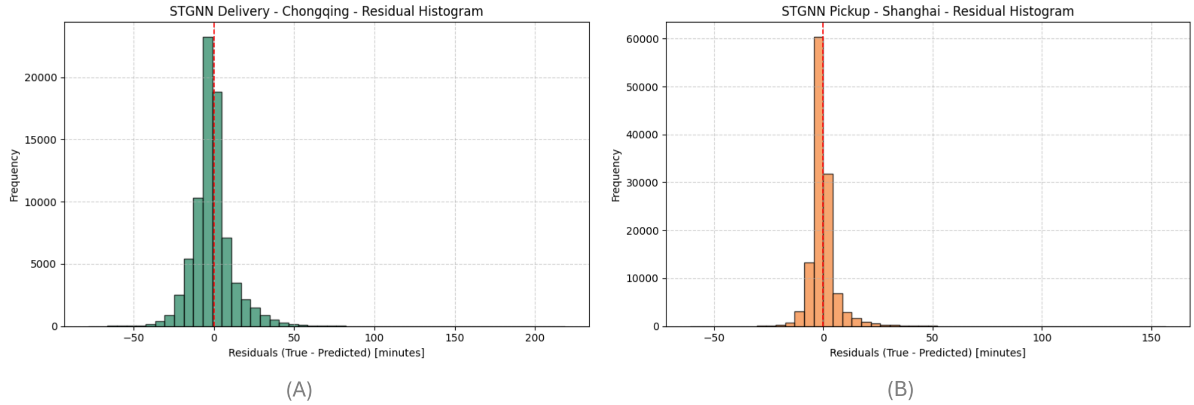


Figure 4: Residual Histogram (STGNN Model)

Based on the results reviewed, the following key observations can be highlighted:

- Our STGNN model tends to overestimate travel times in both deliveries and pickups. While this can be advantageous in operational settings to avoid missed deadlines, it comes at the cost of lower point accuracy.
- The positive skewness in the residuals where skewness > 1 in all cases indicates that the model behaves asymmetrically, with a right-tailed distribution. This means that although the average prediction is an overestimation, large residual errors tend to be underestimations.
- Regarding the standard deviation, we find that although there is variability in the model's predictions, this variability is better controlled in pickup tasks compared to deliveries.

This analysis offers valuable insights that can guide future improvements, particularly by incorporating external variables or adopting more sophisticated modelling strategies aimed at balancing prediction bias and enhancing overall model accuracy.

6.2 Comparative and Statistical Analysis

6.2.1 Global Comparison of Evaluation Metrics

The following summary table presents the overall performance of each model, using the average values of key evaluation metrics:

Table 6: Summary of Average Metrics and Bias Trends in Time Prediction Models

Model	MAE Prom.	RMSE Prom.	R^2 Prom.	Bias Trend
LSTM Baseline	~6.25 min	~11.13 min	~0.923	Significant underestimation
Adjusted LSTM	~4.04 min	~8.70 min	~0.948	Moderate underestimation
LSTM + Residual Learning	~3.39 min	~7.73 min	~0.962	Slight underestimation
STGNN	~5.24 min	~8.15 min	~0.939	Slight overestimation

In Table 6, we can observe that the LSTM + Residual Learning model achieves the best results in terms of MAE and RMSE for most cities. However, the STGNN model offers comparable R^2 values and, interestingly, an opposite behaviour in terms of bias. This means that although STGNN does not reduce MAE as effectively as LSTM + Residual Learning, it achieves similar RMSE and R^2 scores, suggesting that it is more robust to extreme variations, albeit with a slight bias toward overestimation.

Regarding the bias patterns, we can deduce that the first three models—LSTM Base, Adjusted LSTM, and LSTM + Residual Learning—tend to underestimate travel times, although the intensity of this bias decreases progressively as improvements and refinements are applied. In contrast, STGNN stands out as the only model that consistently shows a slight overestimation, which could be beneficial in scenarios where avoiding delays is more critical, even at the expense of slightly longer predicted times.

From a technical perspective, the drop in average MAE from ~6.25 minutes (in the base LSTM model) to ~3.39 minutes (in the LSTM + Residual Learning model) signifies that the predictions are considerably closer to the actual values, leading to potentially significant improvements in operational planning and logistics.

This illustrates the importance of balancing prediction accuracy and bias. While the LSTM + Residual Learning model provides the best balance with low error and reduced underestimation, the STGNN model offers an attractive alternative when the priority is to minimize the risk of delays. Ultimately, model selection should be context-dependent: in cities with high traffic variability, a model with a slight overestimation like STGNN may be more reliable, whereas in more predictable environments, the higher precision of the LSTM + Residual Learning model could be more advantageous.

6.2.2 Residual Analysis and Skewness

The residuals of the LSTM-based models consistently show a positive skewness (skewness > 1), suggesting a significant presence of underpredictions, i.e., the model tends to predict lower times than the actual values. Interestingly, while the STGNN model also exhibits positive skewness, it is the only model that produces positive residual means across all cases, indicating a systematic overestimation of travel times.

This finding is especially relevant from an operational perspective. In urban logistics, underestimation can lead to customer dissatisfaction and operational overload, while overestimation could provide a buffer or margin of flexibility. The decision to prioritize one behaviour over another will depend on the optimization criteria and specific goals of the use case.

The models have been evaluated separately for each of the five cities in China. This segmentation allows us to observe that Shanghai and Hangzhou consistently exhibit better overall performance across all architectures, while Jilin and Chongqing show higher errors, particularly in delivery tasks. These discrepancies may be hypothetically linked to road network complexity, weather conditions, or lack of standardization in traffic patterns. However, confirming these hypotheses will require future research, which opens new opportunities for enhancing current model performance.

Table 7: MAE by City and Movement: LSTM, Residual Learning, and STGNN

City	Movement	LSTM Baseline MAE	Adjusted LSTM MAE	LSTM + Residual Learning MAE	STGNN MAE
Chongqing	Delivery	11.28	10.62	7.42	8.61
Chongqing	Pickup	5.19	1.73	2.27	4.84
Yantai	Delivery	5.68	2.34	4.32	5.17
Yantai	Pickup	4.50	1.69	1.39	4.75
Shanghai	Delivery	7.86	7.26	4.70	5.89
Shanghai	Pickup	3.40	1.26	1.89	3.80
Hangzhou	Delivery	6.08	5.67	2.74	4.94
Hangzhou	Pickup	2.98	0.87	1.07	3.12
Jilin	Delivery	7.89	5.29	3.88	5.73
Jilin	Pickup	7.67	3.65	4.19	5.51

6.3 Implications

6.3.1 Academic

From an academic perspective, the results presented in this thesis demonstrate that incorporating residual learning into sequential models such as LSTM achieves significant improvements in accuracy and stability. This combination allows for better capture of the nonlinearities present in the time series of logistic times, thus contributing to the literature that seeks to merge recurrent networks with structured adjustments.

On the other hand, the use of models like STGNN opens an emerging direction in graphical modelling of urban space through GNNs, which explicitly encode spatial relationships that sequential models fail to capture. Although its performance was slightly lower in some cases, the opposite bias and its stability in cities with complex topographies make it a robust alternative with great potential for future work. Among possible improvements is the incorporation of geographic space at the street level by city, which would allow for a more accurate representation of distances compared to the geodesic distances used in this study.

6.3.2 Practical

In the operational domain, these results have direct applications in intelligent urban and last-mile logistics systems. The choice between a model with a tendency to overestimate or underestimate can align with the business strategy: for example, a startup focused on efficiency might opt for the LSTM + Residual Learning model, while a platform oriented towards customer satisfaction might prefer STGNN.

Additionally, segmentation by city highlights the importance of having personalized or regionally adjusted models, since geographic and infrastructure factors considerably influence model performance.

6.3.3 Relation to Literature Review

The findings of this thesis align with previous studies highlighting the limitations of LSTM in complex urban scenarios (reference). However, this work contributes by incorporating residual learning techniques that reduce cumulative errors in sequential predictions, achieving significant improvements that, depending on the context, can have a very relevant operational impact. On the other hand, the results obtained with the STGNN model confirm recent literature on the application of GNNs in transportation models (reference), where the graph structure at the street level becomes essential to capture congestion patterns and travel times. This model also confirms that GNNs tend to overestimate due to the propagation of information among highly connected nodes, an important aspect to consider. A key contribution of this thesis is the use of real data, although an important gap is recognized: the lack of precise information about the start and end points of each route limits the ability to model the complete cycle of each trip. Future work proposes including this information along with relevant external variables, such as weather conditions or special road states for each city.

7 Conclusion and Future Work

The main question this study tries to answer is: How can spatiotemporal neural networks help better predict delivery and pickup times and routes in cities, and how can simple reinforcement learning help make these decisions better? To address this, a methodology based on quantitative analysis of real-world data was developed and implemented combining sophisticated predictive models like LSTM networks and spatiotemporal graph neural networks (STGNN), along with developing a conceptual framework for fundamental reinforcement learning.

With regard to the particular objectives, a number of significant successes were attained:

a. Five Chinese cities' urban delivery and pickup data were thoroughly processed to produce a well-organized and enriched dataset fit for spatiotemporal analysis. Capturing the complicated urban dynamics influencing logistics depends on this dataset.
b. Efficient predictive models utilizing both LSTM and STGNN were created and trained. They showed good performance metrics (MAE, RMSE, MAPE, R) in forecasting logistic times. These findings demonstrate that, in contrast to conventional approaches, these models are able to successfully record spatial and temporal patterns, therefore enhancing prediction accuracy.
c. A reinforcement learning test without a computer was planned with states, actions, and rewards clearly defined in line with logistics goals. The conceptual groundwork establishes a strong basis for more study even if this works' deadline passed before its whole practical application could be finished.
d. The hybrid model, which combines LSTM with residual adjustments, has the advantage of being able to model complicated urban routing structures well, and the graph-based approach shows a lot of promise.

Combining reinforcement learning methods with strong spatiotemporal learning can help to solve urban logistics issues, according to these results. They also underline how properly modelling spatial relationships and including contextual information can greatly improve prediction accuracy and efficiency, therefore enabling intelligent logistics management systems.

7.1 Future Research

Based on these findings, several interesting areas of future research and development might improve the predictive capacity and practical application of this method:

The following phase is to go beyond the theoretical idea and really apply the reinforcement learning experiment. Ideally, real-time data and simulated settings would be used to evaluate how effectively it enhances timing and routing decisions. More sophisticated techniques like Deep Q-Networks (DQN) or Policy Gradient approaches could enable adaption to the changing and erratic character of metropolitan logistics.

Bringing in real-time data streams like traffic updates, weather conditions, and unforeseen occurrences would let the models dynamically adapt and maintain current predictions and judgments.

Evaluating how well the method generalizes and spotting needed changes for diverse cultural, geographic, or legal settings will help test the models in other cities with different urban designs and logistic patterns.

Improving Spatiotemporal Models: To capture more sophisticated and subtle correlations in the data, hybrid designs mixing spatiotemporal graphs with sophisticated attention mechanisms or self-supervised learning approaches may be investigated.

More sustainable and equitable logistic solutions may result from adding further goals into the reinforcement learning reward functions—such as lowering environmental effect, distributing workload across delivery agents, or raising customer satisfaction—to improve multi-objective optimization and sustainability.

Real-World Deployment and Pilot Studies: Ultimately, working with urban delivery systems or logistics firms to conduct pilot tests would offer insightful information on the actual viability of these models. Real-world deployment would make it possible to fine-tune depending on operational feedback and show actual advantages in user experience, cost reduction, and efficiency. These next actions can not only advance the research but also close the gap between academic work and practical uses, therefore supporting wiser, more resilient cities.

References

Adi, T.N., Iskandar, Y.A. and Bae, H. (2020) ‘Interterminal truck routing optimization using deep reinforcement learning’, *Sensors (Switzerland)*, 20(20), pp. 1–20. Available at: https://doi.org/10.3390/s20205794.

Bing Yu, Yin and Zhu, Z. (2018) *Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting*. Available at: https://aaafoundation.org/american-driving-survey-2014-2015/.

Crivellari, A. *et al.* (2022) ‘Multi-target CNN-LSTM regressor for predicting urban distribution of short-term food delivery demand’, *Journal of Business Research*, 144, pp. 844–853. Available at: https://doi.org/10.1016/j.jbusres.2022.02.039.

Guo, T. *et al.* (2025) ‘Learning-Aided Neighborhood Search for Vehicle Routing Problems’, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(07), pp. 5930–5944. Available at: https://doi.org/10.1109/TPAMI.2025.3554669.

Jiang, J. and Feofilova, A. (2025) ‘Research on Urban Road Traffic Flow Prediction Based on Hybrid CNN-LSTM Model’, *Applied and Computational Engineering*, 129(1), pp. 77–83. Available at: https://doi.org/10.54254/2755-2721/2025.20467.

Jin, G. *et al.* (2022) ‘STGNN-TTE: Travel time estimation via spatial–temporal graph neural network’, *Future Generation Computer Systems*, 126, pp. 70–81. Available at: https://doi.org/10.1016/j.future.2021.07.012.

Li, F. *et al.* (2023) ‘Dynamic Graph Convolutional Recurrent Network for Traffic Prediction: Benchmark and Solution’, *ACM Transactions on Knowledge Discovery from Data*, 17(1). Available at: https://doi.org/10.1145/3532611.

Mahajan, D., Pottigar, V. V and Fahad, A. (2025) *Implementing Real-Time Traffic Flow Prediction Using LSTM Networks for Urban Mobility Optimization*, *Communications on Applied Nonlinear Analysis*. Available at: https://internationalpubls.com.

Qi, Y. *et al.* (2022) ‘Status update frequency optimization for In-vehicle LAN route based on deep Q-learning’, *Results in Control and Optimization*, 8. Available at: https://doi.org/10.1016/j.rico.2022.100163.

Shi, C. *et al.* (2025) ‘Enhancing Travel Time Prediction for Intelligent Transportation Systems: A High-Resolution Origin–Destination-Based Approach with Multi-Dimensional Features’, *Sustainability (Switzerland)*, 17(5). Available at: https://doi.org/10.3390/su17052111.

Shi, R. and Du, L. (2022) ‘Multi-Section Traffic Flow Prediction Based on MLR-LSTM Neural Network’, *Sensors 2022, Vol. 22, Page 7517*, 22(19), p. 7517. Available at: https://doi.org/10.3390/S22197517.

Shokrollahi, S. and Dehghan, M. (2023) ‘TGRV: A trust-based geographic routing protocol for VANETs’, *Ad Hoc Networks*, 140. Available at: https://doi.org/10.1016/j.adhoc.2022.103062.

Wang, Y. *et al.* (2025) ‘The multi-depot pickup and delivery vehicle routing problem with time windows and dynamic demands’, *Engineering Applications of Artificial Intelligence*, 139. Available at: https://doi.org/10.1016/j.engappai.2024.109700.

Yuan, Y., Li, H. and Ji, L. (2021) ‘Application of Deep Reinforcement Learning Algorithm in Uncertain Logistics Transportation Scheduling’, *Computational Intelligence and Neuroscience*, 2021. Available at: https://doi.org/10.1155/2021/5672227.

Zhang, C. *et al.* (2020) ‘AIS data driven general vessel destination prediction: A random forest based approach’, *Transportation Research Part C: Emerging Technologies*, 118. Available at: https://doi.org/10.1016/j.trc.2020.102729.

Zhao, S., Xing, S. and Mao, G. (2022) ‘An Attention and Wavelet Based Spatial-Temporal Graph Neural Network for Traffic Flow and Speed Prediction’, *Mathematics*, 10(19). Available at: https://doi.org/10.3390/math10193507.

Zhou, X. *et al.* (2023) ‘Inductive Graph Transformer for Delivery Time Estimation’, in *WSDM 2023 - Proceedings of the 16th ACM International Conference on Web Search and Data Mining*. Association for Computing Machinery, Inc, pp. 679–687. Available at: https://doi.org/10.1145/3539597.3570409.