Computer Science
&
Data Science

Capstone Report - Fall 2024

Forecasting Spatial-Temporal Demand for Food Delivery

Feiluan Feng,
Mingxuan Xie

supervised by
Zhibin Chen

Preface

This report is inspired by the Meituan-INFORMS-TSL-Research-Challenge, which is one of the questions raised by China's largest food delivery platform Meituan, focusing on optimization of the on-demand food delivery system. As Computer Science students, we are interested in transportation development and optimization. Along with the background knowledge of machine learning, we select the topic of forecast the spatial and temporal distributions of future orders and do deeper research in it. We hope our research could proposed some ideas for optimizing the food delivery system in this direction, achieving system efficiency improve.

Acknowledgements

We express our heartfelt gratitude to our supervisor, Zhibi Chen, and his PhD student, Zhi Li, for their invaluable guidance throughout this project. Their expertise and insights have been instrumental in shaping our research. We would also like to thank our mentor, Promethee Spathis, for his continuous encouragement and the additional guidance he provided, which greatly motivated us along the way.

We are deeply appreciative of Meituan for providing the dataset that formed the foundation of our study. Special thanks to NYU Shanghai for offering resources and fostering an environment conducive to research, as well as the entire CSDS department for creating opportunities and supporting our academic growth.

Additionally, we are grateful for the safe and comfortable 24/7 study spaces provided by the university, which became our second home during this project. Lastly, a special thanks to Meituan's late-night food delivery service, which kept us fueled during countless long nights of dedication and discovery.

Abstract

In China, the food delivery industry is well-developed and the market is growing yearly. In the complex food delivery system, the operator, system, can optimize and plan the delivery time, route, etc. of food orders based on the collected historical data of users, delivery drivers, and merchants. Prediction of the spatiotemporal distribution of food orders is one of the many prediction optimizations that aims to achieve higher efficiency in operations, delivery, and personnel scheduling. In this paper, we develop a Convolution LSTM neural network framework combined with a double-column mechanism to predict the demand distribution of food orders. We train and evaluate our model using real data provided by Meituan, China's largest food delivery platform. The results show that our model can capture the temporal and geographical sequence characteristics of the training data and make predictions. We also propose ideas for the future development of the model.

Keywords

Machine Learning, Demand forecasting, Transportation, Estimation, Predictive models

Contents

1 Introduction

1.1 Background

On-demand food delivery (OFD) is a real-time local service that allows users to order meals anytime and have them delivered promptly to a designated location. This service has revolutionized access to food, providing convenience and time savings while creating new social and economic opportunities.

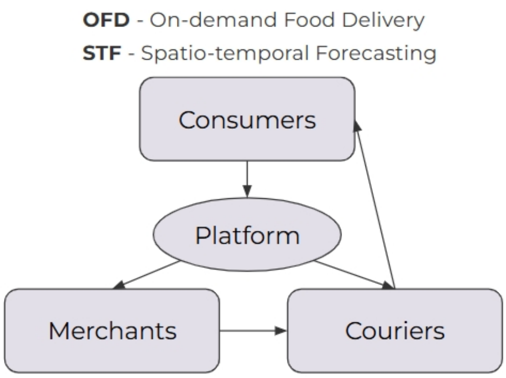

Figure 1: Four Skateholders in OFD system

OFD platforms consist of four kinds of stakeholders, which are consumers, merchants, platforms, and couriers. In this service system, consumers place orders from various locations to respective merchants through the platform. The platform then assigns these orders to couriers for delivery, ensuring that food arrives within a promised timeframe(figure 1) [1].

The global online food delivery market is projected to expand significantly, with an expected compound annual growth rate of 10.3% from 2023 to 2030, reaching an estimated value of USD 505.50 billion by 2030 [2]. In China, Meituan, the largest OFD platform, operates with over 5 million active couriers, serving more than 3,000 cities. Its daily operations include 60 million deliveries, involving 9.3 million merchants and 687 million consumers [1].

While this rapid growth has driven economic benefits, it also introduces operational challenges for OFD platforms, such as complex real-time order allocation and routing, balancing stakeholder interests, and ensuring service efficiency.

1.2 Significance

Accurate order demand prediction for an OFD platform is critical for addressing the following challenges:

- **Operational Efficiency:** By optimizing courier positioning and resource allocation, platforms can reduce costs and enhance scalability. For instance, pre-allocating couriers to areas of predicted high demand ensures faster response times.
- **Stakeholder Benefits:** Effective forecasting supports profitability for platforms, increases courier productivity, improves service reliability for customers, and boosts order volumes for merchants, enhancing overall operational efficiency.
- **Logistical Applications:** Beyond food delivery, the developed methodologies can be applied to other domains such as retail logistics, public transportation, and emergency services, addressing broader real-time resource management challenges.

While platforms such as Meituan have adopted advanced algorithms for forecasting orders, the dynamics and complexity of demand require continuous improvement in forecasting technology.

1.3 Objectives

After addressing the challenges, our research mainly focus on the area of spatial-temporal forecasting of OFD order. The research aims to address the following key questions:

- What is the spatial-temporal distribution of real-world online food order demands?
- What are the major factors affecting this distribution?
- How can machine learning models be developed to accurately forecast demand distributions?
- What are the impacts of prediction models on food delivery systems?

The intended outcome is to develop predictive methodologies that provide accurate forecasts of the spatial-temporal distribution of food delivery demand. These insights aim to support decision-making in optimizing delivery operations and resource allocation, ultimately improving system efficiency and user satisfaction.

While prior research has made progress in forecasting delivery times, relatively few studies have focused on predicting the spatial and temporal distribution of order demand. Liang et al.'s work on Poisson-based demand range prediction provides a valuable foundation, but it primarily focuses on vehicle-based transportation during specific contexts, such as the pandemic and in specific countries like Singapore [3]. Our research extends this by addressing lightweight, electric mobile couriers typical of platforms like Meituan, in China‘s OFD contexts, incorporating both spatial and temporal prediction techniques.

1.4 Our Contributions

In this research project, we have achieved the following:

- **Development of Forecasting Methods:** We designed and tested machine learning models, including LSTM, ConvLSTM, and a hybrid Poisson-based framework, to predict OFD order demand distributions with improved accuracy and adaptability.
- **Integration of Spatial-Temporal Features:** Using innovative spatial partitioning techniques like Uber's H3 grid system, we modeled complex spatial dependencies while accounting for temporal dynamics, enabling more robust predictions.
- **Exploration of Methods for Small Datasets:** Given the constraints of working with a relatively small dataset provided by Meituan, we explored techniques to mitigate data sparsity challenges, including windowing approaches and hybrid modeling strategies, demonstrating how predictive accuracy can still be achieved under limited data scenarios.
- **Insights into System Optimization:** Our findings contribute to improving resource allocation strategies, potentially enhancing system efficiency and user satisfaction.

In the following section 2 of the report, we delve into a review of related work to contextualize our research within existing studies on OFD and Demand Forecasting Models. Then, in section 3 the methodology section outlines our data processing strategies, predictive modeling approaches and developments, and evaluation metrics. Subsequently, in section 4 we will analyze the result we produced. Section 5 and Section 6 will be the discussion and the conclusion of the topic.

2 Related Work

Our research builds upon a range of spatio-temporal forecasting (STF) methodologies and demand prediction models. In this section, we review foundational approaches, including statistical models, machine learning methods, and deep learning frameworks, with a focus on their relevance to on-demand food delivery (OFD) systems. Additionally, we analyze the role of spatial indexing systems like H3 and explore advancements in label distribution learning (LDL) and Poisson-based distribution prediction (PDP). Throughout, we compare these works to highlight their contributions and limitations, as well as how our research diverges and builds upon them.

2.1 Demand Forecasting Models

2.1.1 Statistical Models

Traditional statistical models, such as AutoRegressive Integrated Moving Average (ARIMA) [4] and exponential smoothing (ETS) [5], have been extensively used for time-series forecasting. Extensions like Space-Time Autoregressive Integrated Moving Average (STARIMA) [6] incorporate spatial dependencies, allowing for spatio-temporal forecasting in applications such as traffic flow prediction [7] and public health monitoring [8]. However, STARIMA assumes fixed spatial correlations and homogeneous parameter values across locations, making it unsuitable for dynamic systems like OFD, where demand patterns vary significantly across regions and time.

2.1.2 Machine Learning Methods

Machine learning techniques, such as Support Vector Machines (SVMs) [9] and ensemble methods like Random Forest [10] and XGBoost [11], improve upon statistical models by handling nonlinear relationships and higher-dimensional data. These methods are robust and computationally efficient, making them attractive for initial forecasting tasks. However, their reliance on static feature engineering and inability to model intricate spatio-temporal dependencies limit their applicability in OFD systems.

In contrast, our approach employs deep learning methods capable of learning spatio-temporal features directly from raw data, reducing the need for manual feature selection while improving scalability and adaptability.

2.1.3 Deep Learning-Based Models

Deep learning models have revolutionized STF by simultaneously capturing spatial and temporal dependencies. Recurrent neural networks (RNNs), particularly Long Short-Term Memory (LSTM) networks [12], are widely used for sequence modeling, while convolutional neural networks (CNNs) excel at spatial feature extraction. Temporal Convolutional Networks (TCNs) [13] further enhance temporal modeling by replacing sequential processing with causal convolutions, improving computational efficiency and prediction accuracy.

ConvLSTM [14], a hybrid model combining LSTM with convolutional layers, preserves spatial structures while modeling temporal dynamics, making it particularly suitable for STF. ConvLSTM has shown success in applications like traffic prediction [15]. The advantage of ConvLSTM in capturing spatial and temporal dependencies makes it a strong candidate for predicting the spatiotemporal order volume of OFD systems. Combining ConvLSTM, we aim to integrate temporal and spatial information for the OFD order distribution, learning and analyzing the spatiotemporal dependencies.

2.2 H3 Spatial Representation

In on-demand food delivery (OFD) systems, spatial representation is essential for modeling geographically distributed demand. However, residents' location, specific latitude, and longitude, are undoubtedly sensitive information. In this project, the dataset we used provided by the Chinese OFD platform Meituan includes shifted latitude and longitude data and could not be decoded. This preventing direct alignment with actual city blocks, so that the geographic location information is encrypted for protection.

To address this problem, we introduce the H3 spatial indexing system [16] on the shifted geologic data over the simple latitude-longitude grid. H3 divides geographic areas into uniform hexagonal cells, offering superior spatial accuracy by minimizing edge effects. Its hierarchical structure supports seamless scaling across resolutions, which aligns well with ConvLSTM's grid-based input requirements. By preprocessing the dataset with H3, our approach captures spatial relationships effectively despite the limitations of the shifted geographic data.

2.3 Poisson-Based Distribution Learning Framework

Label Distribution Learning, LDL [17] produces outputs as probability distributions, making it suitable for tasks like age estimation and sentiment analysis. LDL places distributions on ambiguous labels, which can be helpful in preventing over-fitting [18].

In 2023, Liang et al. [19] extended LDL to OFD systems with the Poisson-based distribution prediction (PDP) framework and addressed data imbalance through a double-hurdle mechanism. PDP predicts demand ranges rather than single values, providing more actionable insights for resource allocation, which is one of the few algorithms focused on OFD order forecasting.

2.4 Summary

The reviewed methods highlight advancements in spatial-temporal forecasting (STF) and demand prediction, particularly using ConvLSTM for dynamic spatiotemporal modeling and H3 for efficient spatial representation. However, existing models face limitations, such as static spatial representations and difficulties in handling long-tailed distributions common in on-demand food delivery (OFD) systems.

Our research addresses these gaps through two key approaches: integrating ConvLSTM with H3 to preprocess datasets into effective grid-based formats and exploring LDL/PDP algorithms to enhance prediction performance on small, long-tailed datasets. These complementary strategies provide a robust framework for improving the accuracy and adaptability of STF in OFD contexts.

3 Methdology

In the previous section, we introduced the main content of our research, research questions, and related work. In this section, we will introduce the architecture of the model we designed.

3.1 Data Preprocessing

The Dataset we use for our research is the Meituan TSL challenge dataset [20]. The dataset contains 654343 lines of ordering data in China, which describes the process of food order dispatching and categorizing into three parts: order, courier, and assignment inputs.

3.1.1 OFD order demand distribution

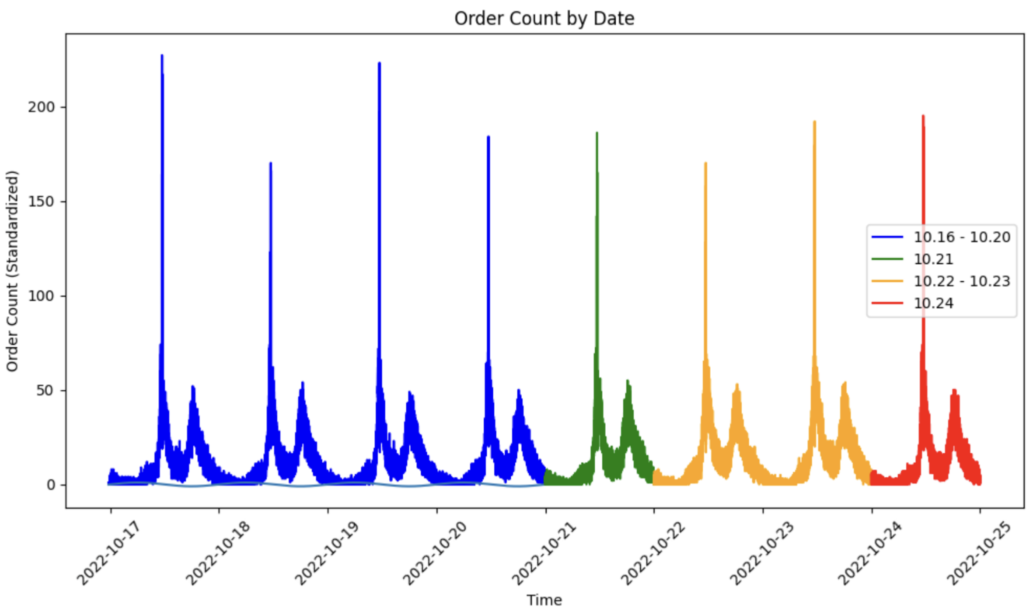

Figure 2: Orders counted by dates(temporal patterns)

Different from general demand forecasting such as logistics supply chain, OFD demand distribution shows strong temporal patterns. From the Dataset of Meituan, it is easy to recognize

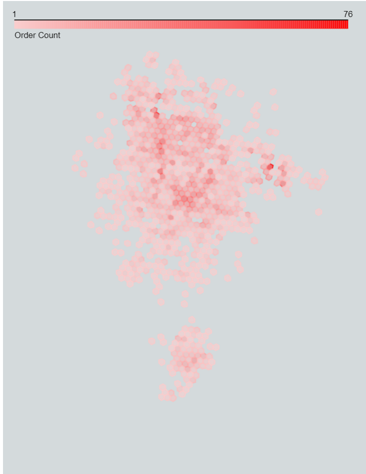

Figure 3: The H3 cells map of the food demand in 30 min on Monday noon, resolution=9

there are a large number of extreme values in the order demands, which concentrate on lunch and dinner time from 11:00 AM. to 6:00 PM, while the order demand during other times are relatively small, even close to zero The OFD demand appeared a typical long-tail distribution. The time intervals with large order demand volumes do not account for a large proportion of the day, but their existence has a significant impact on the characteristics of the overall distribution. This kind of data feature makes OFD demand forecasting a challenging task.

3.1.2 geographic information processing

As we have mentioned before, the location information is sensitive information, and the longitude and latitude coordinates provided in the Meituan dataset have been desensitized for privacy problems and couldn't be decoded. This makes it impossible for us to combine the dataset content with real geographic areas and traffic data for analysis.

However, considering the spatiotemporal prediction of order demand is expected to serve the optimization of the food delivery system, based on the experience of using food delivery platforms and relevant literature, we applied the H3 developed by Uber on the dataset, which is a discrete global grid system that divides geographic areas into hexagonal cells and indexed into unique IDs [21]. According to the actual geographic space size corresponding to the resolution index, we believe that when resolution=12, the area of 307 square meters can basically cover most personal living spaces, and when resolution=9, the area of 105,332 square meters basically matches the block area of Chinese cities(figure 3) [16].

3.2 LSTM - PDP Framework

To reduce the impact of extreme values in the long-tail distribution on the training prediction neural network, we choose to train a demand distribution neural network, which is a double-hurdle mode function and suitable for cases with many zero values.

The model receives input features and generates two outputs. At the same corresponding time, one output predicts the probability of estimating the order demand to be zero, and another predicts the label distribution for the order demand at the same time.

3.2.1 Notation Definition

Table 1: Notation Definitions

Notation	**Definition**
X	Input feature matrix
y_{b}	Binary labels indicating whether the demand is zero (1) or non-zero (0).
y	Continuous demand values for non-zero demand samples.
$\hat{y}_b$	Predicted probability of zero demand.
$\hat{y}$	Predicted demand for non-zero demand samples.
$\mathbf{p}$	Proxy label distribution representing the probabilities of demand falling into discretized ranges.
$\hat{\mathbf{p}}$	Predicted label distribution.
$\mathcal{L}_{\mathrm{BCE}}$	Binary cross-entropy loss for the binary classification task.
$\mathcal{L}_{\mathrm{MSE}}$	Mean squared error loss for continuous count prediction.
$\mathcal{L}_{\mathrm{KL}}$	KL divergence loss to align predicted and true count distributions.
$\mathcal{L}_{\mathrm{total}}$	Total loss combining binary, MSE, and KL divergence losses.

3.2.2 Model Definition

Given an input dataset (X, y), where X represents the input features and y denotes the corresponding order demands indicating the order count. In the feature matrix X, it contains the historical demand sequence, time, order count, and the h3 indexes.

We utilize a neural network function, denoted as $f(X;\theta)$, where θ encapsulates the learnable parameters of the network. This function takes the input feature X and transforms it into a latent feature representation through a series of non-linear operations. During the input, the h3 indexes are embedded into the neural network as an embedding feature and merged to the next hidden layer, which saves space and ensures index transitivity.

The first output $\hat{y}_b$ predicts the probability that the order demand at this time is zero. We have y_b from y which indicating whether the demand y is zero ($y_b = 1$) or non-zero ($y_b = 0$). To predict the probability that the count of order demand y is zero ($y_b = 1$), the latent representation produced by $f(X;\theta)$ is passed through a single dense layer equipped with sigmoid activation. This layer maps the extracted features to a probability score, computed as:

$$\hat{y}_b = \frac{1}{1 + \exp\left(-\left(w_s^\top f(X;\theta) + b_s\right)\right)},$$

where w_s and b_s represent the weights and bias of the fully connected layer.

The second output is a probability distribution over possible demand counts $\hat{p}$ passed through a softmax function of the neural network function. For the direct output, we denote it as z, then shape it into the distribution $\hat{p}$:

$$\mathbf{z} = \left[z_0, z_1, \ldots, z_k, \ldots, z_{|l|}\right]^\top = \mathbf{W}^\top f(\mathbf{X};\theta) + \mathbf{b},$$

where $\mathbf{W}$ is the weight matrix of the fully connected layer, $\mathbf{b}$ is the bias vector of the fully connected layer, and the length of z is $|\mathbf{l}|$, where $\mathbf{l}$ denotes the length of ordered label set of our order demands.

The softmax function is applied to $\mathbf{z}$ to compute the predicted probability distribution $\hat{\mathbf{p}}$, where each element and the predicted probability distribution could be defined as: $\hat{p}_k$ is defined as:

$$\hat{p}_k = \frac{\exp(z_k)}{\sum_{i=0}^{|l|} \exp(z_i)},$$

$$\hat{\mathbf{p}} = \left\{\hat{p}_0, \hat{p}_1, \ldots, \hat{p}_k, \ldots, \hat{p}_{|l|}\right\}.$$

Using this probability distribution, the expected count $\hat{\mathbf{y}}$ is computed as:

$$\hat{y} = \frac{\sum_{j=1}^{|l|} j e^{\beta \hat{p}_j}}{\sum_{i=1}^{|l|} e^{\beta \hat{p}_i}},$$

which ensures the $\hat{y}$ could be calculated, and $\beta \geq 1$ is a hyper parameter.

This ensures the model provides both a discrete probability distribution for count predictions and a continuous expected count value.

3.2.3 Model Loss Evaluation

The model described is a multi-task deep learning model designed for a time-series prediction task involving binary classification and count-based regression.

Binary Cross-Entropy Loss The Binary Cross-Entropy Loss is used for the binary classification task, which predicts whether the demand is zero ($y = 0$) or non-zero ($y > 0$). The binary cross-entropy loss is defined as:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_b^i \log(\hat{y}_b^i) + (1 - y_b^i)\log(1 - \hat{y}_b^i)\right],$$

where N is the Total number of samples, T is the sequence length, $\hat{y}_b^i$ is Predicted probability that $y > 0$, and y_b^i is Binary ground truth label.

Mean Squared Error (MSE) Loss The Mean Squared Error Loss is used for the regression task of predicting demand counts. Using the softmax operation, the model converts softmax probabilities ($\hat{\mathbf{p}}$) into continuous-valued predictions ($\hat{y}$).

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}^i - y^i)^2,$$

where $\hat{y}^i$ predicted continuous demand for sample i, y^i is the ground truth demand for sample i, and N represent the total number of samples.

Poisson Negative Log-Likelihood Loss The Poisson negative log-likelihood (NLL) loss is used to measure the likelihood of the observed count data y under the predicted Poisson distribution $P_{\text{Poisson}}(\lambda_{\text{pred}})$, also the $\hat{\mathbf{p}}$, our output. The loss is defined as:

$$\mathcal{L}_{\text{Poisson}} = -\frac{1}{N}\sum_{i=1}^{N}\log P_{\text{Poisson}}(y_i \mid \lambda_{\text{pred},i}),$$

where N is the total number of samples, $P_{\text{Poisson}}(y_i \mid \lambda_{\text{pred},i})$ represent the Poisson probability mass function, $\lambda_{\text{pred},i}$ is the predicted mean (rate) of the Poisson distribution.

Total Loss Function The total loss function combines the three components:

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{BCE}} + \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{Poisson}}.$$

This combination ensures a balance between binary classification and count regression tasks, enabling the model to perform well across both objectives.

Class weight for Loss Function Considering the data imbalance problem, we employ a class weighting scheme during the training process, where the class weights are defined as class_weights = $\{1 : 5.0, 0 : 1.0\}$, giving higher importance to the non-zero order class ($y = 0$).

3.3 ConvLSTM Framework

The ConvLSTM framework is a crucial part of our approach to spatiotemporal demand prediction, addressing both spatial and temporal dependencies. By integrating convolutional operations into the LSTM structure, ConvLSTM effectively models spatial patterns, such as clustering in urban demand, and temporal dynamics, such as daily trends. This makes it highly adaptable to the hierarchical and dynamic nature of on-demand food delivery systems.

3.3.1 Model Architecture

The ConvLSTM framework processes input sequences structured as 4D tensors:

$$X \in \mathbb{R}^{T \times H \times W \times C},$$

where$T$ is the number of time steps, H, W are dimensions of the spatial grid, and $C$is the number of input channels (e.g., demand values, temporal features).

ConvLSTM integrates convolutional operations into LSTM to jointly model spatial and temporal dependencies. The key operations include:

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + b_i)$$

$$f_t = \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + b_f)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c)$$

$$o_t = \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + b_o), \quad H_t = o_t \odot \tanh(C_t)$$

where i_t, f_t, o_t represent the input, forget, and output gates; C_t and H_t are the cell state and hidden state, respectively.

3.3.2 Notation Definition

Table 2: Notation Definitions for ConvLSTM Framework

Notation	Definition
X	Input tensor representing spatiotemporal data
T	Number of time steps in the input sequence
H, W	Dimensions of the spatial grid
C	Number of input channels (e.g., demand values, temporal features)
H_t, C_t	Hidden state and cell state at time t
i_t, f_t, o_t	Input, forget, and output gates
W_{xi}, W_{hf}	Convolutional kernels for input and hidden states
σ	Sigmoid activation function

3.3.3 Special Preprocessing for ConvLSTM

To adapt the raw dataset for ConvLSTM input requirements and optimize it for spatial-temporal demand forecasting, the following preprocessing steps were applied:

Input Preparation The dataset was divided into uniform grids using the H3 indexing system. Each grid cell aggregated demand values into 10-minute intervals, ensuring temporal uniformity. To handle missing intervals within the overall time range, zero values were used for imputation, ensuring that all time buckets were accounted for. Additionally, a fixed spatial resolution was maintained by mapping the H3 grids into a consistent 97×97 matrix. This transformation allowed seamless spatial representation, reshaping raw hexagonal grid coordinates into a format suitable for ConvLSTM.

Sliding Windows To capture temporal dependencies, a sliding window approach was employed. For a given time step t, a sequence of k past time steps was used to predict the demand at the next step. The sliding window can be expressed as:

$$X_t = \{D_{t-k}, D_{t-k+1}, \ldots, D_{t-1}\}, \quad \hat{D}_t = f(X_t),$$

where D_t represents the demand matrix at time t and $\hat{D}_t$ is the predicted demand. This reorganized the dataset into sequences that the ConvLSTM model could process effectively.

Spatial and Temporal Features Hexagonal grids ensured consistent spatial representation, avoiding distortions common with rectangular grids. The time dimension was segmented into fixed-length time buckets, aligning demand patterns across grids for temporal modeling. By combining spatial grids with temporal buckets, the dataset was transformed into a structured spatial-temporal format suitable for deep learning models.

Data Normalization and Scaling To handle the long-tailed nature of demand distributions, a logarithmic transformation was applied:

$$x' = \log(1 + x),$$

where x represents the raw demand. Afterward, Min-Max scaling normalized the data into the range $[0, 1]$:

$$x'' = \frac{x' - \min(x')}{\max(x') - \min(x')}.$$

Data Imputation Two specific imputation strategies were applied to maintain the integrity of the dataset:

- **Empty Time Buckets**: Time intervals with no recorded data were imputed with zero values to maintain temporal continuity.
- **Static Spatial Regions**: Regions that never received any orders during the dataset's timeframe were explicitly retained as zero-filled grids, ensuring consistency across the 97×97 matrices.

3.3.4 Loss Functions

The ConvLSTM model is optimized using a weighted Mean Squared Error (MSE) loss function to address the sparsity of the data:

$$\mathcal{L}_{\mathrm{MSE}} = \frac{1}{N} \sum_{i=1}^{N} w_i \cdot (\hat{y}^i - y^i)^2,$$

where:

- $w_i = 10$ for non-zero demand ($y_i > 0$) and $w_i = 1$ otherwise.
- $\hat{y}^i$: Predicted demand for sample i.
- y^i: Ground truth demand for sample i.
- N: Total number of samples.

3.3.5 Outputs

The ConvLSTM model generates one output:

Continuous Demand Prediction ($\hat{y}$): Expected demand for each grid cell.

The output provide detailed and actionable insights into both the presence and magnitude of demand, making the model suitable for food delivery demand forecasting.

4 Results and Discussion

In our actual operation, we combined the LSTM and PDP frameworks we developed to predict the time and space of orders. The advantage of doing so is that it saves space. We divide the space into h3 indexes and embed them into the network, save the index data, and predict the order volume.

4.1 Evaluation Matrices

4.1.1 Evaluation Matrices of LSTM-PDP Frame Work

We used three common evaluation metrics for the evaluation of the LSTm-PDP framework's distribution output: mean absolute error (MAE), root mean square error (RMSE), and Pearson Correlation Coefficient (PCC).

The formulas are given as follows:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

$$\mathrm{PCC} = \frac{\sum_{i=1}^{N}(y_i - \bar{y})\left(\hat{y}_i - \bar{\hat{y}}\right)}{\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}\sqrt{\sum_{i=1}^{N}\left(\hat{y}_i - \bar{\hat{y}}\right)^2}}$$

4.1.2 Evaluation Matrices of ConvLSTM network

We also used three common evaluation metrics for the evaluation of the LSTm-PDP framework's distribution output: mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE).

The formula for MAPE is given as follows:

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100$$

4.2 Data tables

The evaluations of the following two figures are performed after the model training has basically converged. Table 3 presents the performance of the LSTM-PDP framework on two different datasets: 5-minute clip data and 10-minute clip data. In two different training sessions, we process training data differently. In the first session, we split our dataset into 5-minute time segments and counted the number of orders at each h3 index grid every five minutes. Similarly, the second session is split into 10-minute time segments and trained. In comparison, the data in the second session fluctuates more and the model performance is relatively worse. It should be noted that because our dataset contains limited data, both have the possibility of over-fitting.

Table 3: Evaluation Metrics for LSTM-PDP Framework

	MAE	**RMSE**	**PCC**
5 min clip data	0.0017	0.0023	0.0214
10 min clip data	0.0132	0.0148	0.0257

Table 4 only evaluates the performance of the ConvLSTM network on the same dataset using the mentioned three metrics. Compared to the LSTM-PDP Framework, the performance dropped significantly.

Table 4: Evaluation Metrics for ConvLSTM network

MAE	**RMSE**	**MAPE**
0.0621	0.0783	0.1027

4.3 Graphs

Time Series Predictions: The line graphs (Figure 4 represent the predicted values versus true values for specific grid points. The model performs relatively well during peak hours (morning, midday, and evening), capturing the general trends and magnitude of demand spikes. However, in off-peak hours, particularly during late-night or early-morning time slots, the predictions exhibit higher deviation and noise, with underestimation of certain spikes and overprediction in low-activity regions.

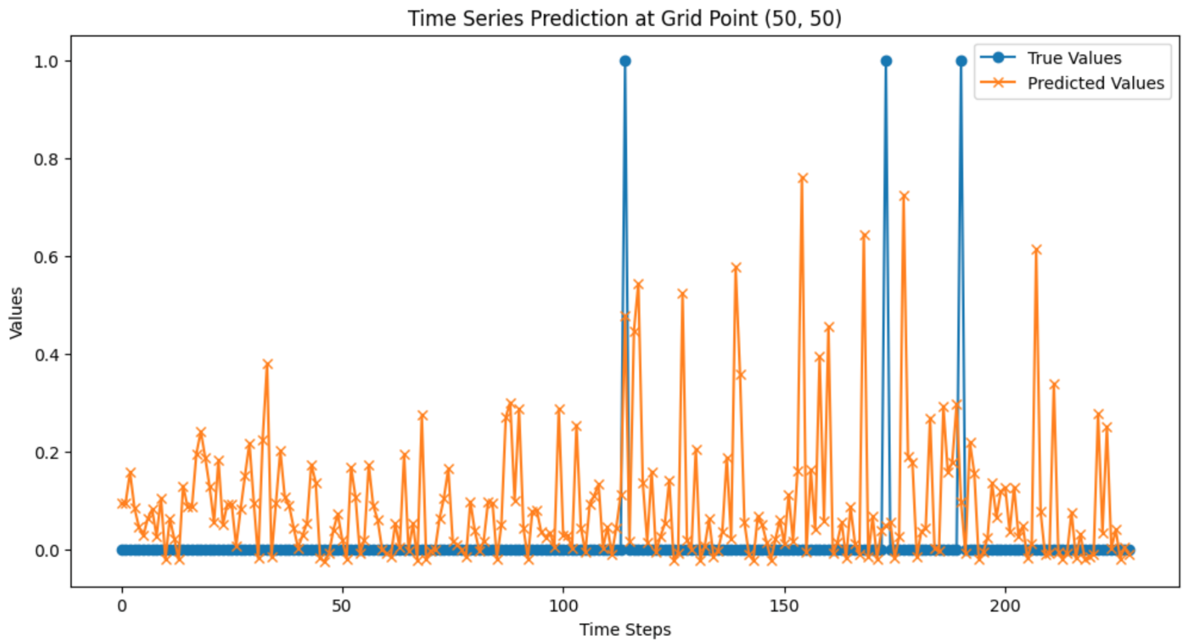

Figure 4: Time Series Prediction at Grid Point(50,50)

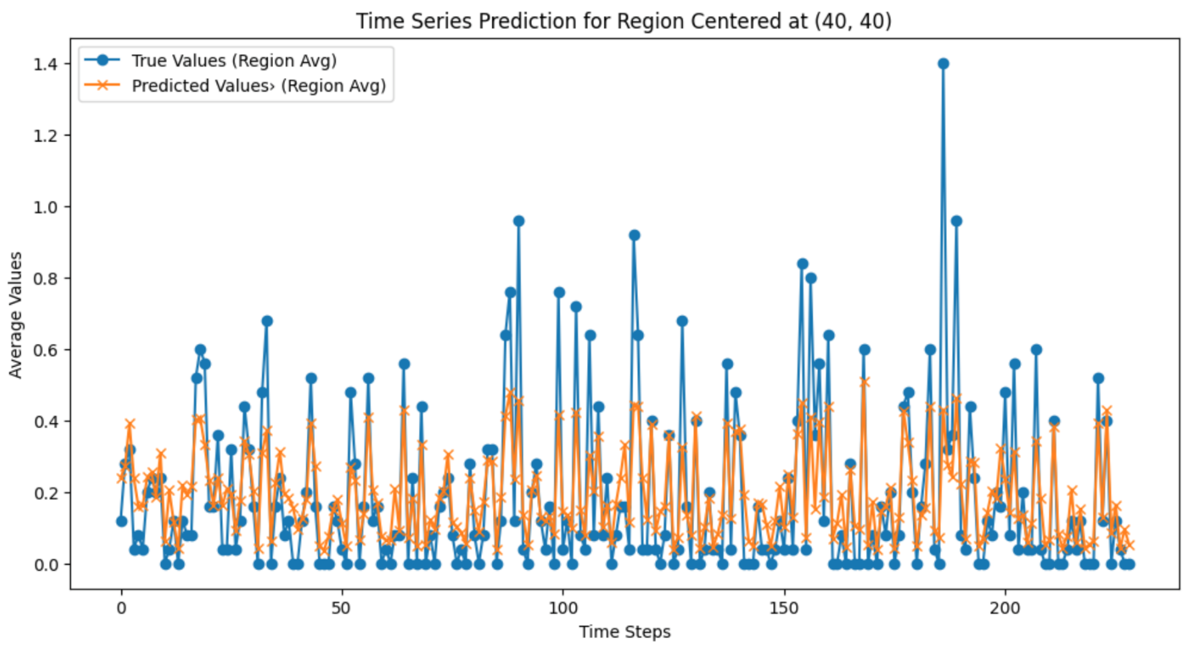

Figure 5: Time Series Prediction at Grid Point(40,40)

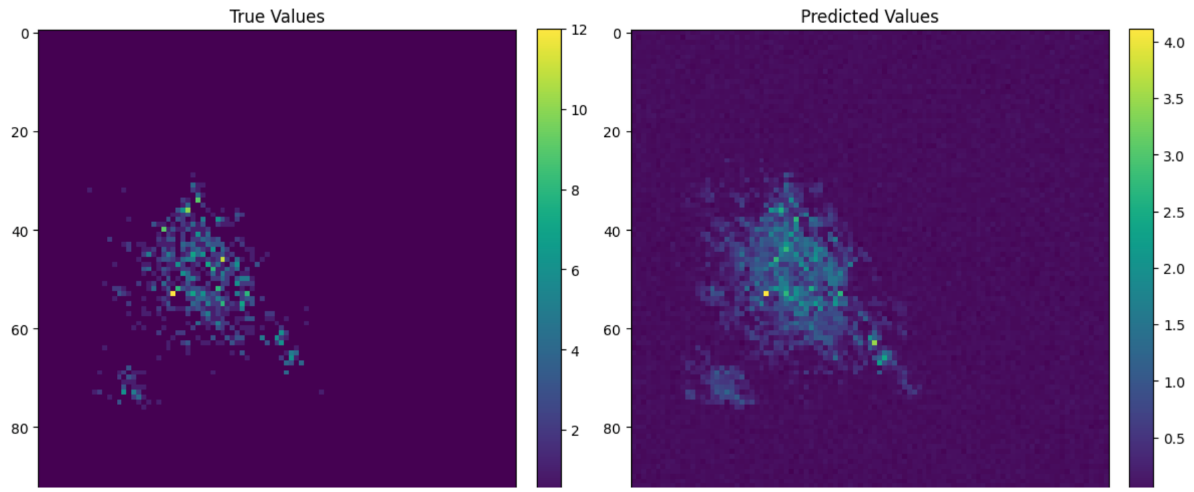

Figure 6: Spatial distribution comparison at Window index=40

Regional Average Predictions: Figure 5 shows the average predictions for a broader region centered at (40, 40). The model demonstrates robustness in aligning with the aggregated trends, providing a smoother output compared to individual grid points.

Spatiotemporal Dynamics: Figure 6 and Figure 7 Heatmap analysis confirms that the model captures demand hotspots and trends, providing meaningful predictions for high-density regions. When window_index=40, the comparing figure corresponds to Monday morning, when the demand of food delivery should be high and wide; while window_index=2, the time_bucket representing a period during a midnight. Although the renormalization of the predicted value met an unknown issue, making the predicted value range and true value range mismatch, the dispersion of points and the distribution of color depth roughly match.

5 Discussion

5.1 Dataset

During the development procedure, we met with quite a lot of problems, and the most serious problem was the limited dataset from Meituan. The dataset provided by Meituan is only about 8 ranges of days, which makes our prediction have to focus on the daily changes in the OFD order demands, but ignore the weekdays and weekends' effect. Also, it is hard to learn other features like weather in 8 days of data range. Although we successfully developed our models, we really hope there will be a larger dataset for our model.

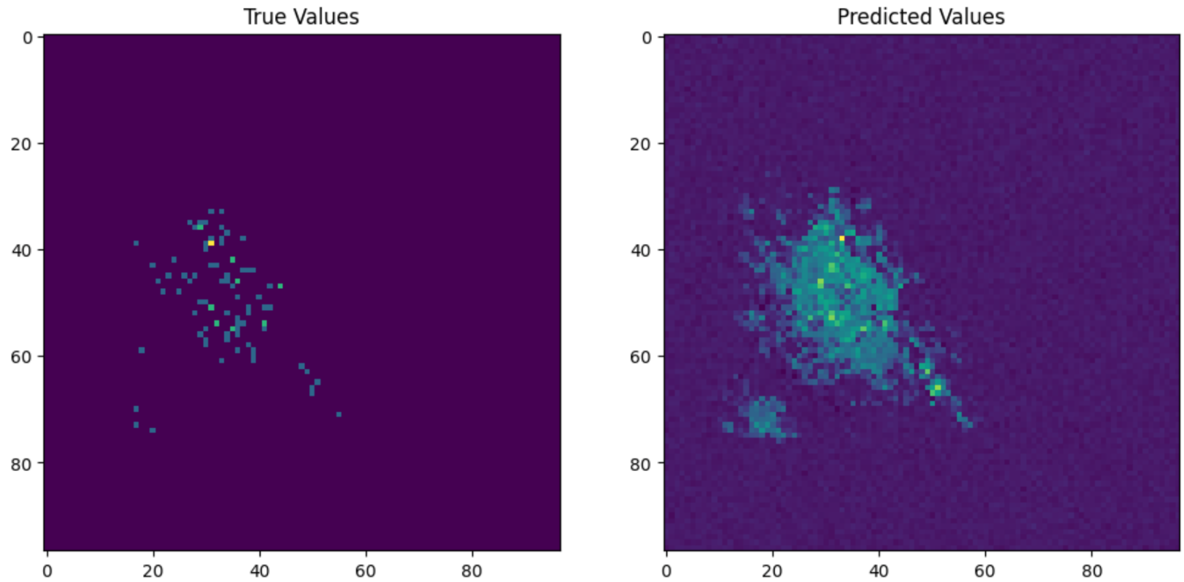

Figure 7: Spatial distribution comparison at Window index=2, left value range=[0,4],right value range=[01.5]

5.2 Limited related work

Demand forecasting in the delivery of food orders is still a field with relatively few research results. Although companies such as Meituan may have developed similar functions in their algorithms, there are very few related papers.

5.3 Models Limitations

5.3.1 ConvLSTM

The ConvLSTM model have difficulties in accurately predicting demand during off-peak hours and in low-demand regions. The lack of training data may lead to high errors in training results.

5.3.2 LSTM-PDP

The LSTM-PDP framework still has the concern of overfitting on the limited dataset.

6 Conclusion

We mainly develop two neural network models that can be used for OFD order demand spatiotemporal prediction and answer the first three research questions mentioned above.

For the ConvLSTM model, we confirmed its potential in predicting spatiotemporal demand in food delivery systems. It can capture temporal dependence and spatial correlation, and effectively deal with the long tail data distribution in delivery demand. Using the heat map or other tools allows it to be easily visualized. For the LSTM-PDP model, we confirm its strong learning ability on our dataset, and the double-hurdle framework applied by the model is very effective in processing sparse data.

However, there is still a long way to go for accurately prediction for the OFD order demand service. Also, we didn't solve the fourth research question we raised. We propose possible future research directions, including combining LSTM and the double-hurdle framework, considering how to combine them with the real transportation network and perform artificial partitioning instead of h3. Optimize the model to be able to be combined with the food delivery algorithm. Consider larger datasets and additional weather and city data that affects the order demands, etc. In summary, the model we developed provides a solid foundation for spatiotemporal demand forecasting of takeout orders and is expected to be improved and applied in practical scenarios.

7 Code

https://drive.google.com/drive/u/0/folders/1D1$_u 0ILvMouPMZMb - HhcE86P09nd - bWg$

References

[1] Meituan, "Meituan's real-time intelligent dispatching algorithms build the world's largest minute-level delivery network," *Informs journal on applied analytics*, vol. 54, no. 01, pp. 84–101, 2024. [Online]. Available: https://doi.org/10.1287/inte.2023.0084

[2] "Online food delivery services market size & outlook, 2030," Aug 2024. [Online]. Available: https://www.grandviewresearch.com/horizon/outlook/online-food-delivery-services-market-size/global

[3] J. Liang, J. Ke, H. Wang, H. Ye, and J. Tang, "A poisson-based distribution learning framework for short-term prediction of food delivery demand ranges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 12, pp. 14 556–14 569, 2023.

[4] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. Hoboken, NJ, USA: John Wiley & Sons, 2015.

[5] R. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder, *Forecasting With Exponential Smoothing: The State Space Approach*. Cham, Switzerland: Springer, 2008.

[6] P. E. Pfeifer and S. J. Deutsch, "A starima model-building procedure with application to description and regional forecasting," *Transactions of the Institute of British Geographers*, vol. 5, no. 3, pp. 330–349, 1980.

[7] P. Duan, G. Mao, W. Yue, and S. Wang, "A unified starima based model for short-term traffic flow prediction," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 1652–1657.

[8] Z. Malki, E.-S. Atlam, A. Ewis, G. Dagnew, A. R. Alzighaibi, G. ELmarhomy, M. A. Elhosseini, A. E. Hassanien, and I. Gad, "Arima models for predicting the end of covid-19 pandemic and the risk of second rebound," *Neural Computing and Applications*, vol. 33, pp. 2929–2948, 2021.

[9] A. Boukerche, Y. Tao, and P. Sun, "Artificial intelligence-based vehicular traffic flow prediction methods for supporting intelligent transportation systems," *Computer Networks*, vol. 182, p. 107484, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128620311567

[10] D. Yang, T. Heaney, A. Tonon, and et al., "Crimetelescope: Crime hotspot prediction based on urban and social media data fusion," *World Wide Web*, vol. 21, pp. 1323–1347, 2018.

[11] B. Ashwini, R. Sumathi, and H. Sudhira, "A dynamic model for bus arrival time estimation based on spatial patterns using machine learning," *arXiv preprint arXiv:2210.00733*, 2022.

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

[13] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *CoRR*, vol. abs/1803.01271, 2018. [Online]. Available: http://arxiv.org/abs/1803.01271

[14] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in neural information processing systems*, vol. 28, 2015.

[15] Y.-Y. Hong and T. R. A. Satriani, "Day-ahead spatiotemporal wind speed forecasting using robust design-based deep learning neural network," *Energy*, vol. 209, p. 118441, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544220315498

[16] Uber, "Tables of cell statistics across resolutions." [Online]. Available: https://h3geo.org/docs/core-library/restable/

[17] X. Geng, "Label distribution learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1734–1748, 2016.

[18] E. Imani and M. White, "Improving regression performance with distributional losses," in *International conference on machine learning*. PMLR, 2018, pp. 2157–2166.

[19] J. Liang, J. Ke, H. Wang, H. Ye, and J. Tang, "A poisson-based distribution learning framework for short-term prediction of food delivery demand ranges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 12, pp. 14 556–14 569, 2023.

[20] Meituan, "Meituan-informs-tsl-research-challenge[repository]," https://github.com/meituan/Meituan-INFORMS-TSL-Research-Challenge, 2024, accessed: 2024-09-02.

[21] Uber, "Introduction." [Online]. Available: https://h3geo.org/docs/