Transportation Research Part C 181 (2025) 105383

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Uncertainty-aware probabilistic travel demand prediction for mobility-on-demand services

Tao Peng *, Jie Gao , Oded Cats

Department of Transport and Planning, Delft University of Technology, The Netherlands

ARTICLE INFO

Keywords:
Mobility-on-demand services
Probabilistic forecasting
Travel demand
Variational autoencoder

ABSTRACT

Demand prediction is essential for effective management of Mobility-on-Demand (MoD) systems, as accurate forecasts enable better resource allocation, reduced wait times, and improved user satisfaction. Beyond that, probabilistic prediction methods that explicitly account for uncertainty are particularly valuable, as it allows decision-makers to assess risk and make robust plans under uncertain operational environments. However, most existing approaches focus on point predictions, which fail to capture the full spectrum of possible future outcomes. For probabilistic prediction, many methods typically rely on strong parametric distributional assumptions that may not accurately reflect the complex real-world environments. Nonparametric methods proposed in the literature, although promising, often suffer from high computational costs and model complexity, limiting their practical applicability. To overcome these challenges, we propose the Spatial-Temporal Graph Convolutional Network Variational Autoencoder (STGCN-VAE), a novel deep learning framework designed for uncertainty-aware probabilistic travel demand prediction in MoD services. The STGCN-VAE effectively captures complex spatial-temporal dependencies and inherent uncertainties in MoD demand data, generating diverse and realistic future demand scenarios and constructing comprehensive demand distributions. Specifically, the proposed framework integrates three key components: a Spatial-Temporal Graph Convolutional Network (STGCN) to learn complex spatial-temporal dependencies, a Variational Autoencoder (VAE) to compress these patterns into a latent space, and a Kernel Density Estimation (KDE) module to accurately construct probabilistic demand distributions and quantify uncertainties. Experiments on four different real-world MoD datasets including both rideshare and bikeshare services across different cities demonstrate that STGCN-VAE consistently outperforms state-of-the-art baselines in both point and probabilistic prediction, highlighting its robustness and broad transferability across service modes and urban contexts.

1. Introduction

Mobility-on-Demand (MoD) systems, such as ride-hailing and ride-pooling provided by Uber, Lyft and DiDi, offer real-time, flexible mobility by matching passenger requests with available drivers through digital platforms. These services have rapidly gained popularity by providing convenient, point-to-point transport, especially in areas where traditional public transit is less accessible (Lucken et al., 2019). In addition, they support the development of multi-modal transport by complementing public transit and active modes like walking and cycling, offering first- and last-mile connectivity and improving access to opportunities without the need for private

* Corresponding author.
E-mail addresses: t.peng-2@tudelft.nl, t.peng-2@tudelft.nl (T. Peng), j.gao-1@tudelft.nl (J. Gao), o.cats@tudelft.nl (O. Cats).

https://doi.org/10.1016/j.trc.2025.105383
Received 19 May 2025; Received in revised form 3 September 2025; Accepted 3 October 2025
Available online 16 October 2025

car ownership (Audenhove et al., 2020). As reported by Mahajan (2025), the global ride-hailing market is projected to grow from approximately $181.72 billion in 2025 to $441.20 billion by 2032, reflecting a compound annual growth rate (CAGR) of 13.5 % .

As these services continue to expand, ensuring their operational efficiency becomes increasingly critical. A key functionality of these systems is their ability to make proactive decisions based on anticipated demand. Accurate demand forecasting enables platforms to anticipate when and where ride requests will occur, facilitating proactive decisions such as guiding drivers toward areas with expected high demand (Gao et al., 2021; Van Engelen et al., 2018) and implementing dynamic pricing strategies, such as surge pricing, in anticipation of imbalances between supply and demand (Wang and Yang, 2019). Without reliable forecasting, platforms lack the ability to optimize operations proactively and are limited to reactive and sub-optimal responses. Worse still, inaccurate predictions can mislead these decisions, sending drivers to the wrong areas, triggering inappropriate pricing responses, and thereby resulting in the inefficient matching between riders and available vehicles. These inefficiencies result in longer wait times, unnecessary travel for drivers, and missed revenue opportunities.

Given the central role of demand prediction in MoD operations, substantial research has focused on developing accurate forecasting methods. Traditional forecasting methods relied on classical time series models, such as Time-Varying Poisson Processes, ARIMA, and Vector AutoRegressive models, which provided point estimates of demand but struggled to capture complex urban dynamics (Yuan and Li, 2021). The advent of deep learning introduced neural network architectures, such as Convolutional Neural Networks (CNNs) (LeCun et al., 1998) and Recurrent Neural Networks (RNNs) (Elman, 1990), which significantly improved prediction accuracy by modeling temporal trends and local patterns. However, travel demand exhibits strong spatial and temporal dependencies. To capture spatial correlations, such as those between neighboring city zones, recent studies leverage graph neural networks (GNNs) (Kipf and Welling, 2016), including Graph Convolutional Networks (GCNs) and Graph Attention Networks (GATs) (Veličković et al., 2017), alongside temporal dynamics, achieving promising results.

MoD systems operate in highly dynamic environments influenced by external factors such as traffic congestion, weather conditions, and large-scale events (e.g., concerts or sports games). These factors introduce significant uncertainty, making single-point forecasts inadequate for robust decision-making (Gao et al., 2025). Understanding and quantifying uncertainty in demand forecasting is therefore essential, as it provides stakeholders with richer information to assess risks, supports robust planning under varying demand scenarios, and enhances transparency and fairness in system operations (Bhatt et al., 2021). To address this, probabilistic forecasting methods have emerged to estimate not only expected demand but also deliver possibility-aware insights to support decision-making under uncertainty. Parametric approaches, such as those assuming Gaussian or Negative Binomial distributions (Stoklosa et al., 2022), estimate distribution parameters but often rely on restrictive assumptions that fail to capture real-world complexity. For instance, Gaussian assumptions may poorly reflect the skewed or multimodal demand patterns typical of urban mobility systems (Gammelli et al., 2020). Nonparametric methods offer greater flexibility by avoiding predefined distributions, yet they frequently suffer from high computational costs or model complexity, limiting their applicability in real-time MoD settings.

In this study, we propose a novel deep learning framework, the Spatial-Temporal Graph Convolutional Network Variational Autoencoder (STGCN-VAE), to address these limitations and advance nonparametric probabilistic demand forecasting. Unlike traditional methods, STGCN-VAE directly learns demand distributions from data without relying on fixed parametric assumptions, enabling more accurate and flexible uncertainty quantification. Our framework integrates three key components: (i) the STGCN module captures intricate spatial-temporal dependencies across city regions, (ii) the Variational Autoencoder (VAE) maps these patterns into a latent space for diverse scenario sampling, and (iii) a Kernel Density Estimation (KDE) module constructs robust probability distributions for future demand. The integration is challenging, as it requires aligning the STGCN and VAE to bridge generative modeling and forecasting: the model must preserve multi-horizon temporal dependencies, maintain cross-region correlations, and reconstruct demand distributions that capture the empirical characteristics of demand rather than being constrained by a fixed likelihood across scenarios. We achieve this by conditioning the VAE on STGCN features, so that the latent generator is tied to recent spatio-temporal context and produces coherent multi-step samples. During prediction, we draw multiple demand samples in parallel and use KDE to assemble a smooth predictive distribution with well-calibrated uncertainty intervals. Through extensive experiments on real-world MoD datasets, we demonstrate that STGCN-VAE substantially improves both forecasting accuracy and uncertainty quantification, while maintaining computational efficiency.

The main contributions of this paper are:

- We develop STGCN-VAE, a novel deep learning framework that integrates spatial-temporal graph convolution, variational autoencoder, and nonparametric density estimation to enable uncertainty-aware, probabilistic travel demand prediction for Mobility-on-Demand (MoD) services.
- We design an architecture that is computationally efficient, modular, and scalable, enabling STGCN-VAE to generate probabilistic demand forecasts with low computational cost and high flexibility. The STGCN backbone efficiently captures spatiotemporal dependencies. Probabilistic samples are generated directly from the latent space and decoder in parallel, eliminating the need for repeated full-model runs. The framework is modular, allowing the backbone to be replaced with alternative models to accommodate different input formats or spatiotemporal structures, enhancing generalizability. It is also scalable, capable of handling large numbers of locations and time steps without a proportional increase in computation, making it suitable for city-wide demand forecasting while maintaining high-quality predictions.
- We conduct extensive experiments on four real-world datasets, including rideshare and bikeshare services across cities with varying geographical scale and demand patterns. Our model significantly outperforms the baselines, e.g., it achieves up to a 55.2 % improvement in interval score on the New York Yellow Taxi Trip Records dataset and a 15 % improvement on the Chicago rideshare dataset. In terms of interval width, it reduces uncertainty by 37.6 % on the Washington DC taxi dataset and by 57.7 %

on the Washington DC bikeshare dataset, compared to the state-of-the-art baseline. These datasets vary in geographical scale, demand patterns, and service type, providing a robust evaluation of model transferability. Across all four datasets, our proposed STGCN-VAE framework consistently surpasses existing methods in both point prediction accuracy and probabilistic forecasting metrics (e.g., CRPS, IS), highlighting its robustness and general applicability across diverse urban contexts and service modes.

The remainder of this paper is structured as follows: Section 2 reviews relevant literature. In Section 3, a formal problem statement is given, followed by details of our proposed method. Section 4 validates the proposed model's performance using public datasets and provides a comprehensive analysis. Finally, Section 5 concludes the work and presents future research directions.

2. Related work

This section reviews the literature on travel demand forecasting with a particular focus on uncertainty quantification. While accurate forecasting is essential for transport planning and resource allocation, existing methods often fail to provide reliable estimates of predictive uncertainty, especially in the presence of complex, nonlinear, and spatiotemporal dependencies. This review synthesizes recent advances in travel demand prediction and identifies key gaps that the proposed STGCN + VAE model— a hybrid Spatiotemporal Graph Convolutional Network and Variational Autoencoder–aims to address, especially in quantifying predictive uncertainty.

To facilitate clarity and guide the reader through a methodologically-driven narrative, the review is organized by problem type and analytical approach rather than by specific application domains. It begins by examining point forecasting methods, first covering traditional statistical models and then progressing to deep learning techniques. Subsequently, it transitions into probabilistic forecasting, highlighting a shift from parametric approaches, where distributional assumptions are explicitly defined, to more flexible non-parametric methods, which avoid assumptions about data distribution. Throughout this structured progression, the review highlights shared methodological challenges, particularly those involving spatiotemporal modeling, and uncertainty quantification.

2.1. Point forecasting

2.1.1. Traditional demand prediction models

Early studies in travel demand forecasting primarily utilized time series models such as ARIMA-based (Box et al., 2015) and regression-based techniques. For instance, Andreoni et al. (2006) proposed an ARIMAX model to forecast travel demand at Reggio Calabria airport using both univariate and multivariate ARIMA models. While univariate models effectively capture trends under stable conditions, the ARIMAX model—incorporating variables such as income and aircraft movements—accounts for policy impacts like the introduction of low-cost routes, predicting a 78 % increase in demand in 2006. However, limitations arise due to data scarcity, sensitivity to boundary conditions, and difficulties in integrating fare estimates into the ARIMAX framework. To that end, Chen et al. (2019) proposesd a framework to predict short-term subway passenger flow during special events using smart card data, aiming to address the volatility and nonlinearity of passenger flow to reduce delays and improve service reliability. It employs a hybrid ARIMA-NAGARCH model, combining ARIMA for mean estimation with GARCH variants to capture volatility, asymmetry, and nonlinearity, and uses k-fold cross-validation to evaluate performance. The methodology involves four steps: establishing the mean model, selecting the volatility model, estimating the hybrid model with different residual distributions, and validating predictions. However, the study focuses on only two subway stations. While the chosen stations are indeed highly impacted by special events, the findings may not generalize to other parts of the subway network, or to other cities with different infrastructures, land-use patterns, or event characteristics. In another line of work, Wu et al. (2012) proposed a sparse Gaussian Process Regression (GPR) model for forecasting tourism demand in Hong Kong using monthly arrival data (1985–2008) from 13 source regions. This model outperforms ARMA and SVM approaches by reducing computational complexity and incorporating multi-factor inputs such as income and transportation costs. However, it still faces challenges in modeling non-stationary covariance structures.

Although these approaches effectively capture linear temporal patterns, they struggle to model the nonlinear dynamics (e.g., abrupt traffic shifts) and spatial dependencies (e.g., networked congestion) inherent in urban mobility systems. These limitations restrict their predictive accuracy in such complex environments.

2.1.2. Deep learning based models

With the rise of deep learning, more expressive models have been developed to improve predictive accuracy in travel demand forecasting. Temporal dependencies—such as daily, weekly, and seasonal patterns—are commonly modeled using Recurrent Neural Networks (RNNs), including Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997) and Gated Recurrent Units (GRUs) (Cho et al., 2014). For example, Ke et al. (2017) proposed the Fusion Convolutional Long Short-Term Memory Network (FCL-Net), which integrates convolutional LSTM, standard LSTM, and convolutional layers to forecast short-term passenger demand for on-demand ride services in Hangzhou, China, using DiDi Chuxing data. FCL-Net captures temporal patterns (e.g., peak-hour demand), spatial dependencies (across a 7×7 grid), and exogenous factors (e.g., travel time rates and weather), outperforming traditional time-series models. However, its high computational complexity may limit practicality for real-time applications.

To address limitations of grid-based spatial representations, Liu et al. (2020) introduced the Physical-Virtual Collaboration Graph Network (PVCGN) for station-level metro ridership prediction. PVCGN integrates three types of graphs: a physical graph representing metro topology, a similarity graph built using Dynamic Time Warping to link stations with similar flow patterns, and a correlation graph reflecting origin-destination (OD) relationships. These are embedded into a Graph Convolutional Gated Recurrent Unit (GC-GRU) and combined with a fully connected GRU for global trends within a Seq2Seq framework. PVCGN outperforms baselines like

LSTM by capturing complex spatial-temporal dynamics and extends effectively to online OD prediction. However, its generalizability may be limited when applied to systems with different topologies, scales, or passenger behaviors.

Despite these advancements, many existing models tend to focus on either temporal or spatial dependencies, while the interaction between the two is crucial for understanding urban mobility dynamics. To better capture these interactions, spatiotemporal forecasting models have been developed. These models represent zones (e.g., administrative areas or demand clusters) as graph nodes and relationships between them (e.g., adjacency or travel flow) as edges, forming a sequence of graphs enriched with temporal information. Graph Convolutional Network (GCN)-based models have shown strong potential in this domain. For instance, Yu et al. (2017) proposed Spatio-Temporal Graph Convolutional Networks (STGCN), a deep learning framework for traffic forecasting. STGCN models traffic networks as graphs and uses a fully convolutional architecture, combining graph convolutional layers (using Chebyshev or first-order approximations) to capture spatial features and gated temporal convolutions for time dynamics. These are structured into spatiotemporal blocks. STGCN outperforms baselines like ARIMA and LSTM, achieving lower errors and faster training and support for parallelization. Its flexibility allows application to a broad range of spatiotemporal sequence prediction tasks. Extending STGCN, Guo et al. (2019) proposed the Attention-based STGCN (ASTGCN) for traffic flow forecasting. ASTGCN introduces spatial and temporal attention mechanisms within each spatiotemporal block and incorporates separate components for different time periods—hourly, daily, and weekly—fusing their outputs for the final prediction. While effective, the use of fixed time segments may limit model flexibility. Guo and Zhang (2020) introduced the Residual Spatio-Temporal Network (RSTN), a deep learning model for short-term travel demand forecasting in ride-sharing services like taxis. RSTN combines Fully Convolutional Neural Networks (FCNs) and a hybrid Conv-LSTM (CE-LSTM) module with residual connections to capture spatial, temporal, and exogenous factors (e.g., weather, time of day). It uses a Dynamic Request Vector (DRV) to model demand trends within time subintervals. While effective, RSTN's performance may struggle with incomplete data or in highly dynamic systems, as noted by the authors. Furthermore, Tian and Chan (2021) introduced the Spatial–Temporal Attention WaveNet (STAWnet), which captures complex spatiotemporal dependencies without relying on predefined graph structures. STAWnet integrates gated temporal convolutional networks (TCNs) with dilated convolutions to model long-range temporal patterns, and a dynamic attention network (DAN) with self-learned node embeddings to adaptively capture latent spatial relationships. This approach improves flexibility and interpretability via attention weight visualization. However, its lack of explicit adjacency modeling may hinder performance in short-term forecasting scenarios where local spatial dependencies are critical. Additionally, Ke et al. (2021) introduced the Spatio-Temporal Encoder-Decoder Residual Multi-Graph Convolutional Network (ST-ED-RMGC) for short-term OD ride-sourcing demand prediction. This model constructs OD graphs with adjacency matrices to capture non-Euclidean geographical and semantic correlations, employing residual multi-graph convolutional networks for spatial modeling and LSTM networks for temporal modeling. A key limitation lies in its computational overhead from processing multiple graphs and large datasets. Furthermore, the model assumes relatively stable demand patterns based on historical data, which may reduce robustness to sudden disruptions or long-term shifts in mobility behavior.

Despite significant progress in improving point prediction accuracy, these models generate deterministic forecasts, predicting a single future value. This neglects the inherent uncertainty in travel demand arising from unpredictable factors such as traffic incidents, weather events, or individual human behavior. Ignoring such uncertainty can lead to overconfident predictions, undermining the reliability of planning and decision-making processes, and ultimately compromising system efficiency and user satisfaction.

2.2. Probabilistic forecasting

2.2.1. Parametric forecasting models

To address uncertainty in travel demand forecasting, recent research has explored probabilistic approaches that quantify uncertainty by estimating prediction intervals (PIs) around forecasted values. A commonly used category of such methods is *parametric forecasting*, which assumes a specific distributional form for the target variable—such as Gaussian or Poisson—and estimates the corresponding parameters. One widely adopted technique is mean-variance estimation, which minimizes the negative log-likelihood (NLL) under the assumed distribution.

For instance, Wang et al. (2024) proposed the Probabilistic Graph Neural Network (Prob-GNN) framework, focusing on average demand forecasting. This approach combines deterministic components (GCN and GAT) with probabilistic assumptions — such as Homoskedastic and Truncated Gaussian distributions — to model public transport and ride-sharing demand in Chicago. LSTM networks capture temporal dependencies, while multi-graph structures capture spatial correlations. Their findings highlight that the choice of distributional assumptions significantly impacts uncertainty estimation. However, a key limitation of this framework is its reliance on parametric assumptions, which may not adequately capture complex or non-parametric uncertainty patterns. This suggests the need for future comparisons with non-parametric alternatives. Similarly, Zhuang et al. (2022) introduced the Spatial-Temporal Zero-Inflated Negative Binomial Graph Neural Network (STZINB-GNN) to model sparse Origin-Destination (OD) demand while quantifying uncertainty. The model integrates diffusion graph convolution and temporal convolutional networks to capture spatiotemporal correlations, and employs a zero-inflated negative binomial distribution with a sparsity parameter π to represent the high frequency of zeros in fine-grained OD matrices. While effective at fine resolutions, the model's performance deteriorates at coarser temporal scales (e.g., 60-minute intervals), where simpler models may be more appropriate, indicating model's sensitivity to data granularity. Another notable work is the DeepNegPol model by de Nailly et al. (2024), which also focuses on parametric forecasting for multivariate count data, using a "sums and shares" distribution framework combined with deep learning. It leverages recurrent neural networks (RNNs) to predict correlated and overdispersed count data, focusing on pedestrian counts at La Défense, a multimodal transport hub in Paris. The model uses two LSTMs: one to predict the total count (sum) via a negative binomial distribution, and another to distribute this sum across locations (shares) using a Dirichlet-Multinomial distribution. However, the "sums and shares" structure, while effective

for regular time series with predictable patterns (e.g., daily or weekly transportation cycles), is less suited for highly irregular or high-dimensional data, limiting the model's generalizability across diverse multivariate count datasets. In another line of work, Zhu and Laptev (2017) proposed the BNN-LSTM model for predicting daily Uber trip counts. This approach combines LSTM networks in an encoder-decoder structure with a Gaussian likelihood for regression outputs. To capture uncertainty, it uses Monte Carlo dropout for epistemic uncertainty and a residual-based estimator for inherent noise. The model outperforms baselines such as vanilla LSTM and quantile random forests in terms of predictive accuracy. However, Bayesian inference methods typically require sampling-based or variational techniques, which are computationally intensive and may hinder their scalability in real-time applications.

A major limitation shared by these parametric approaches is their reliance on strong distributional assumptions. Presuming that travel demand follows predefined distributions—such as Gaussian, Poisson, or Negative Binomial—may introduce bias and reduce predictive reliability if the actual data deviates substantially from these assumptions.

2.2.2. Nonparametric forecasting models

Non-parametric methods, on the other hand, offer a flexible alternative to the aforementioned parametric approaches by avoiding assumptions about the underlying data distribution. In the transportation domain, Gaussian Process (GP) models have been employed as non-parametric approaches for demand and traffic prediction. For example, Zhu et al. (2023b) developed a Bayesian Clustering Ensemble Gaussian Process (BCEGP) framework to capture spatiotemporal heterogeneity in network-wide traffic flow prediction, leveraging ensemble learning and Dirichlet process mixture clustering to improve scalability. Similarly, Zhu et al. (2023a) proposed additive Gaussian Process models for ride-sourcing operations, modeling matching and pickup processes with interpretable kernel structures. These studies highlight the flexibility of GPs in capturing nonlinear and complex patterns without strong distributional assumptions. As an extension of this approach, Neural Processes (NPs) have also been used as a scalable and data-driven approach for capturing complex spatiotemporal patterns in travel demand and traffic dynamics. For example, Li et al. (2024) proposed the Bidirectional Spatial-Temporal Transformer Neural Processes (Bi-STTNP) model for ride-sourcing demand forecasting, which provides probabilistic predictions and uncertainty estimates to support supply-demand management. This model integrates neural processes with bidirectional attention and spatial-temporal transformer modules to capture both supply-demand correlations and spatiotemporal dependencies. It outperforms baseline models in terms of both prediction accuracy and uncertainty quantification. Additionally, a predictability analysis categorizes regions by uncertainty levels using K-means clustering, offering insights for risk-aware operational strategies. However, the model's complexity and computational cost, along with its dependence on high-quality, fine-grained data, may limit its practical applicability.

Building on the objective of producing high-quality uncertainty estimates without distributional assumptions, Pearce et al. (2018) developed a quality-driven (QD) method that constructs prediction intervals (PIs) based on a loss function designed to minimize interval width while maintaining desired coverage. Unlike Bi-STTNP, which directly estimates probabilistic uncertainty, the QD method focuses on ensuring the quality of PIs through a likelihood-based formulation compatible with gradient descent. Despite its strength in generating narrow intervals, the method's lack of explicit probabilistic interpretation may limit its practical relevance in uncertainty-aware decision-making. Similar limitations apply to the Lower Upper Bound Estimation (LUBE) method proposed by Khosravi et al. (2010), where narrow but non-probabilistic PIs fail to reflect the underlying uncertainty dynamics of real-world systems.

Addressing a different aspect of uncertainty, Liu et al. (2023) introduced ProBTTE to improve travel time estimation (TTE) for on-demand ride-hailing by addressing uncertainty. This framework reformulates TTE as a multi-class classification problem using the Distributional Travel Time Encoding (DTTE) module, which discretizes travel time into intervals without assuming a predefined distribution shape. However, the method depends on large-scale historical data to generate route-wise priors, which may not generalize well in areas with sparse data or dynamic traffic conditions.

We also review relevant studies from the energy sector which we found inspiring and constructive in deepening our understanding of nonparametric approaches and their applications. For example, He and Li (2018) developed the QRNNE-UCV model to enhance short-term wind power forecasting by producing full probability density curves. This model combines a quantile regression neural network (QRNN), implemented via a single hidden-layer feedforward network, with kernel density estimation (KDE) using the Epanechnikov kernel. Bandwidth is optimized using unbiased cross-validation (UCV). Validated on wind power data from Ontario, Canada, the method provides detailed uncertainty quantification. However, the KDE bandwidth selection process can be computationally expensive and highly sensitive to the dataset. Similarly, Gu et al. (2021) proposed the LSTM-CM-NPKDE model, combining an improved Long Short-Term Memory (LSTM) network for multi-horizon wind power forecasting (4h, 24h, 72h), a Cloud Model (CM) for qualitative uncertainty assessment, and Non-Parametric Kernel Density Estimation (NPKDE) for deriving full probability density functions and confidence intervals of forecast errors. While the method improves accuracy and uncertainty quantification, its complexity and resource demands may hinder generalization and scalability across datasets with varying properties.

Despite their flexibility, non-parametric methods face several challenges. A common limitation is the computational burden, as these methods often estimate full distributions directly rather than a small number of parameters. Consequently, they require repeated sampling, dropout, or ensemble processes. Moreover, their performance is often highly sensitive to data quality, domain characteristics, and granularity, which can limit generalizability and robustness across different application settings.

2.3. Summmary

Parametric approaches are widely used due to their simplicity and computational efficiency. However, their reliance on predefined distributional assumptions — such as Gaussian or Poisson — can result in biased or miscalibrated uncertainty estimates, particularly

Table 1
Description of key notations.

Symbol	Description
v, V	A unique region in a city, a set of regions.
t, T	Index of time interval in a sequence, the total number of time intervals sequence.
x, X	The number of orders in a region, the set of numbers of orders in all regions.
$\mathcal{G}$	A graph representing all regions as nodes, with edges encoding inter-regional relationships.
Γ	The convolution kernel over a time series sequence.
K	The kernel size of a convolution kernel.
$2C_o$	The number of output channels produced by the kernel.
$\odot, \sigma()$	The element-wise Hadamard product, sigmoid function.
L, D, A	The graph Laplacian matrix, the degree matrix, the adjacency matrix.
$\mathcal{L}_{\text{MSE}}$	The mean squared error.

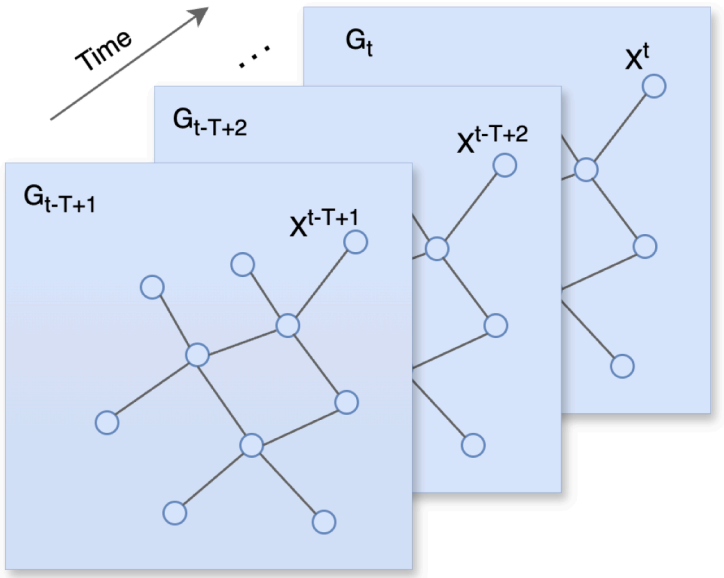


Fig. 1. Graph representation with time series. Graph G_t with travel demand X^t represents the travel demand status at each time step t.

when the actual data distribution deviates from these assumptions. Non-parametric models offer greater flexibility by directly learning uncertainty patterns from data without assuming any specific distribution, leading to improved uncertainty representation. Yet, existing non-parametric methods often suffer from high computational overhead, limited scalability, and sensitivity to data quality. To address these limitations, this study proposes a non-parametric probabilistic forecasting framework based on a Spatiotemporal Graph Convolutional Network combined with a Variational Autoencoder (STGCN + VAE). This model captures the nonlinear, spatiotemporal structure of travel demand while generating calibrated and distribution-free uncertainty estimates. Importantly, by leveraging the powerful spatiotemporal learning capabilities of STGCN and the highly efficient parallel sample generation enabled by VAE, our method achieves significantly higher computational efficiency compared to previous studies.

3. Methodology

In this section, we first formalize the learning problem of spatiotemporal travel demand distribution forecasting. Thereafter we introduce the proposed spatiotemporal graph neural network-based encoder-decoder framework, followed by two subsections that describe the training and testing procedures. Table 1 summarizes the key notations frequently used in what follows.

3.1. Travel demand distribution forecasting

We consider a travel demand distribution forecasting problem that is partitioned into n distinct regions. Let $V = \{v_1, v_2, \dots, v_n\}$ denote the set of all regions, where each $v_i \in V$ represents a unique region within the area of interest, e.g. a city. For simplicity, we will drop the index i when referring to any region as v. To represent the connections between these regions, we define the adjacency

matrix $A \in \mathbb{R}^{n \times n}$, which is represented as:

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \dots & A_{nn} \end{bmatrix}$$

where $A_{ij} = 1$ if regions $v_i \in V$ and $v_j \in V$ are adjacent, and $A_{ij} = 0$ otherwise. Based on this partitioning, we construct a graph representation $\mathcal{G} = (V, A)$, where V is a set of nodes (regions) and A is the connection between them.

Let x_v^t denote the number of orders in the region $v \in V$ during the t^{th} time interval, where $x_v^t \in \mathbb{R}$. We then define $X^t \in \mathbb{R}^{n \times 1}$ as the number of orders in all regions at the t^{th} time interval, with x_v^t as its entry for each region v.

For a sequence of T time intervals, the travel demand sequence is denoted as $X^{t-T+1:t} = [X^{t-T+1}, \dots X^{t-1}, X^t]$, as illustrated in Fig. 1. This sequence captures the historical demand over the past T intervals. Given the defined travel demand sequence, the demand forecasting problem is modeled as a function of the time dependent historical demand sequence. Formally, given $X^{t-T+1:t}$, the goal is to forecast the *conditional probability distribution* of travel demand at the next time step, X^{t+1}, which is represented as:

$$P(X^{t+1} | X^{t-T+1:t}) \tag{1}$$

In the remainder of this section, we introduce the proposed methodology for travel demand forecasting with the consideration of uncertainty.

3.2. Spatiotemporal graph convolutional network variational autoencoder(STGCN-VAE)

Fig. 2 presents the overall structure of our proposed approach, which consists of three key modules. The first module leverages a spatiotemporal graph convolutional network (STGCN) to capture the complex spatial and temporal dependencies inherent in travel demand data, modeling both spatial interactions between regions and temporal patterns across time intervals. These learned features are then processed through an encoder-decoder structure, where the encoder compresses the high-dimensional spatiotemporal information into a latent representation, effectively preserving critical features while reducing dimensionality. The decoder subsequently resamples from this latent representation to generate multiple predictions, representing a range of possible outcomes that capture the uncertainty present in real-world travel demand. To model the complete probability distribution of travel demand, we apply a non-parametric Kernel Density Estimation (KDE) technique. This module constructs a continuous, data-driven probability distribution based on the decoder's predictions, unconstrained by predefined distributional assumptions, thereby offering a flexible and accurate representation of demand uncertainty. In the following subsections, we provide a detailed description of each of these three modules.

3.2.1. STGCN for spatial and temporal information learning

The STGCN is designed to model both spatial dependencies (i.e., how regions are related in space through a graph) and temporal dependencies (i.e., how features of regions evolve over time) of travel demand data. It combines graph convolutional layers to capture spatial relationships with temporal convolutional layers to model dynamic changes over time. Our STGCN design follows the approach outlined in Yu et al. (2017).

The temporal convolutional aspect of STGCN captures patterns in travel demand data over time, as shown in Fig. 3. Specifically, given a time series sequence $X_v = [x_v^{t-T+1}, \dots, x_v^t]$ at a region v, where $X_v \in \mathbb{R}^T$, the model applies a convolution kernel $\Gamma \in \mathbb{R}^{K \times 2C_o}$. Here, K represents the kernel size, indicating the length of the time window over which the convolution operates, and $2C_o$ defines the total number of output channels produced by the kernel. As the kernel slides over the sequence, it extracts temporal features, producing an output matrix $[PQ] \in \mathbb{R}^{(T-K+1) \times (2C_o)}$. This matrix is then split along the channel dimension into two parts, P and Q, each of dimension $(T - K + 1) \times (C_o)$. Then, P, Q are applied with gated linear units (GLU) to control the information flows:

$$P \odot \sigma(Q) \in \mathbb{R}^{(T-K+1) \times C_o} \tag{2}$$

where P, Q are input of gates in GLU respectively; $\odot$ denotes the element-wise Hadamard product. The sigmoid gate $\sigma(Q)$ controls which input P of the current states are relevant for discovering compositional structure and dynamic variances in time series.

The spatial convolution block is used to extract spatial dependencies inherited from travel demand data, and it adopts the spectral-based graph convolution approach (Bruna et al., 2013). The spectral convolution relies on the graph Fourier transform, which is defined using the eigenvectors of the graph Laplacian matrix. The graph Laplacian matrix, denoted as L, is a pivotal operator in spectral graph theory, given by:

$$L = D - A$$

where A is the adjacency matrix as defined in Section 3.1, $D = diag(d_1, d_2, \dots, d_n)$ is the degree matrix, which is a diagonal matrix where its element $d_i = \sum_{j=1}^{n} A_{ij}$, representing the degree (number of connections) of region v_i. Then, the eigen-decomposition of the normalized Laplacian, denoted as L_{norm}, is given by:

$$L_{norm} = D^{-1/2} L D^{-1/2} = U \Lambda U^\top$$

where U is the matrix of eigenvectors (orthonormal basis), and $U^\top$ is the transpose of U, Λ is the diagonal matrix of eigenvalues. The simplified representation of the spectral convolution $\mathbf{g}$ is provided by:

$$\mathbf{g} * \mathbf{s} = U \mathbf{g}(\Lambda) U^\top \mathbf{s}$$

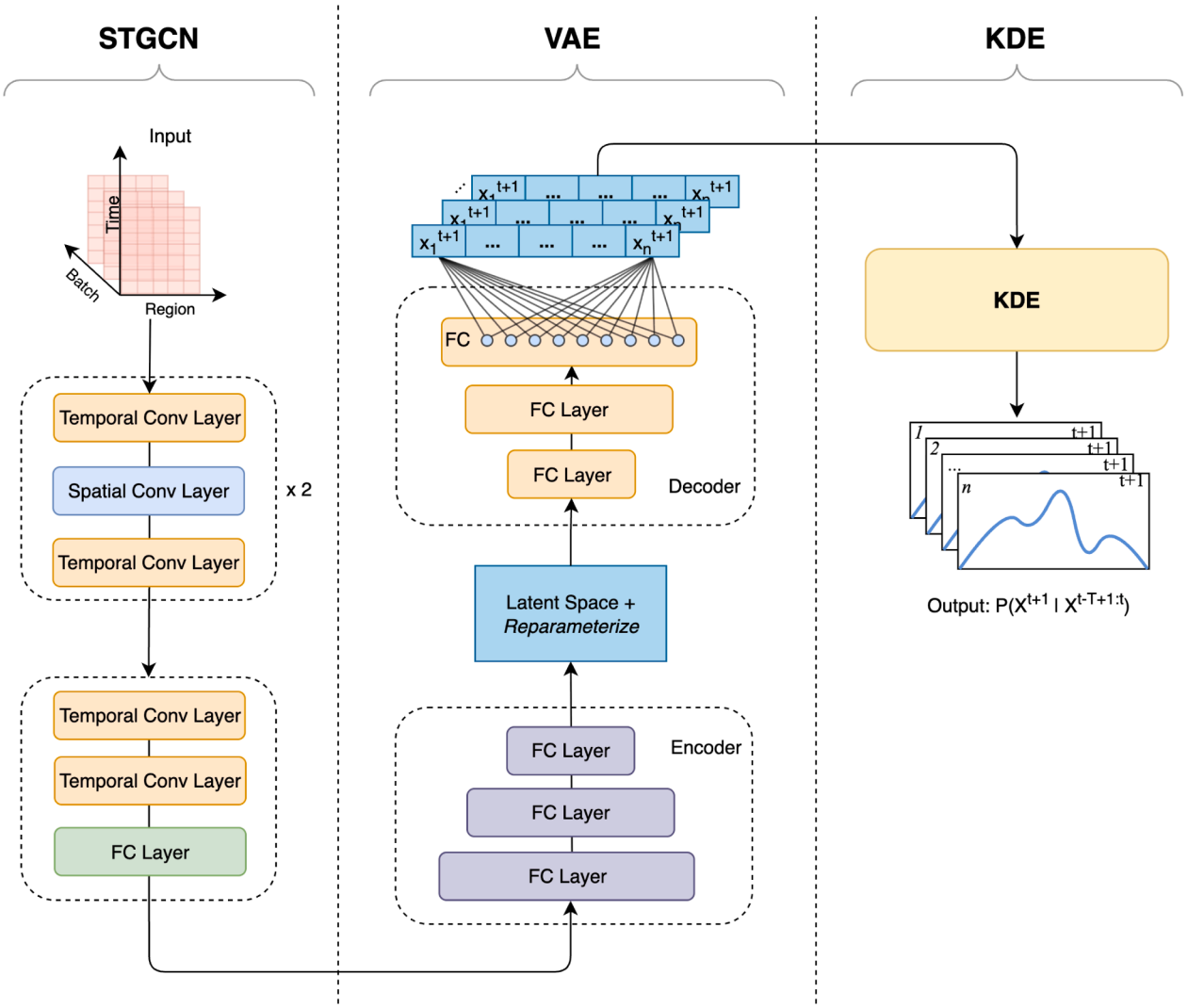


Fig. 2. Overall architecture of the proposed method. During training, the STGCN and VAE modules are trained, while the latent space, decoder, and KDE are employed in the generation step. In the generation step, samples of predicted demand for all n regions at time $t + 1$ are drawn from the latent space and converted into region-wise predictive distributions using KDE.

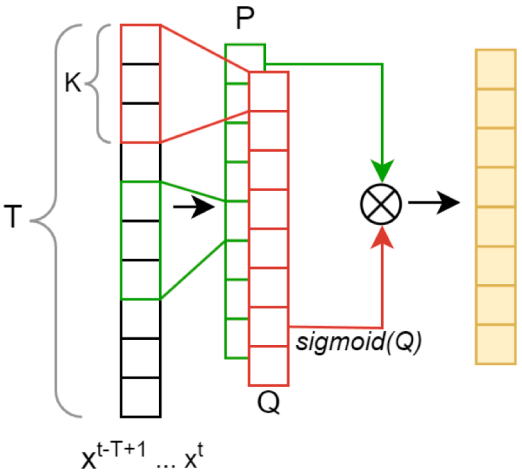


Fig. 3. Temporal convolution block structure.

where each element s_i of $\mathbf{s}$ is the signal value of the region v_i, which is the output of the previous temporal convolution block. The drawback of the implementation of spectral convolution is that the eigen-decomposition involves computing U and Λ is computationally expensive, especially for large graphs. Therefore, Chebyshev polynomials approximation is used to approximate the spectral convolution (Defferrard et al., 2016), which is computationally efficient and avoids the need for explicit eigen-decomposition, given by:

$$\mathbf{g} * \mathbf{s} \approx \sum_{k=0}^{K} \theta_k \mathcal{T}_k(\tilde{L})\mathbf{s}$$

where θ_k are the coefficients of the Chebyshev polynomials; $\mathcal{T}_k(\tilde{L})$ are the Chebyshev polynomials of the rescaled Laplacian $\tilde{L} = 2L/\lambda_{\max} - I$, which normalizes the Laplacian such that its eigenvalues lie in the range $[-1, 1]$; K is the order (or degree) of the polynomial approximation.

In order to fuse features from both spatial and temporal domains, the spatio-temporal convolutional block (ST-Conv block) is constructed to jointly process graph-structured time series. The input $X^{t-T+1:t}$ is uniformly processed by ST-Conv blocks to explore spatial and temporal dependencies coherently. An output layer integrates comprehensive features to feed into the next module for multiple samples prediction. In the subsequent subsection, we explain the motivation and approach of shifting from generating a single forecast to producing a series of potential outcomes which jointly constitute the distribution of possible demand levels.

3.2.2. Variational autoencoder (VAE) for information compression and samples generation

The VAE architecture in the model compresses information and generates new data samples. As shown in Fig. 2, the encoder, consisting of three fully connected layers, receives the learned features $F \in \mathbb{R}^{n \times c}$ from the STGCN module, where n is the number of regions and c is the output channel number, and compresses them into a latent distribution parameterized by $\mu \in \mathbb{R}^{i \times j}$ and $\sigma \in \mathbb{R}^{i \times j}$, denoting as:

$$[\mu, \sigma] = Encoder(F) \tag{3}$$

Here, i and j are determined by data complexity. Using the reparameterization trick, we sample $z = \mu + \sigma \cdot \epsilon$, where $\epsilon \sim \mathcal{N}(0, 1)$, transforming F into a dense latent representation z that captures essential travel demand patterns, discards redundancies, and introduces randomness to explore the underlying distribution. The decoder reconstructs the next-step travel demand:

$$\hat{X}^{(t+1)} = Decoder(z) \tag{4}$$

where $\hat{X}^{(t+1)}$ is the next step travel demand we want to generate. With the predicted next-step travel demand, the loss function can be written as:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{n} \sum_{i=1}^{n} \left(x_i^{t+1} - \hat{x}_i^{t+1} \right)^2$$

where n is the number of regions and x_i^{t+1} is the real demand value in region v_i at next time step, $\hat{x}_i^{t+1}$ is the prediction value. During the training process, we try to minimize the loss and test the model on the testing dataset.

Once training is complete, the model efficiently generates multiple demand samples using the VAE decoder. Since the generation process does not require running the entire model, it significantly reduces the computational load. This approach allows us to generate a vector of potential travel demand for each location in the next time step, enabling construction of a travel demand distribution via a kernel density estimation (KDE), which will be explained in the next subsection.

3.2.3. Kernel density estimation for distribution construction

To estimate the density function that characterizes the distribution of the observed samples, we apply a non-parametric method known as Kernel Density Estimation (KDE). KDE approximates the probability density function of a random variable by smoothing the distribution over the observed samples, offering a continuous and flexible representation of the density. For each region, the respective KDE with a Gaussian kernel is expressed as:

$$\hat{f}(x) = \frac{1}{N h \sqrt{2\pi}} \sum_{i=1}^{N} \exp\left(-\frac{(x - x_i)^2}{2h^2} \right)$$

where N is the number of samples, x_i represents the i^{th} sample at the next time step, and $\hat{f}(x)$ is the estimated density of travel demand x. The parameter h is the bandwidth, which controls the smoothness of the estimate. The term $\exp\left(-\frac{(x-x_i)^2}{2h^2} \right)$ is the Gaussian kernel, which determines each sample's contribution to the density at point x, and the summation aggregates the contributions of all samples.

In KDE, bandwidth is the key parameter that influences the smoothness of the estimate. Both overly narrow and overly wide bandwidths result in reduced accuracy. While careful bandwidth tuning is essential for optimal KDE performance, our study focuses on integrating KDE into the STGCN-VAE model to forecast taxi demand. We therefore leave extensive parameter optimization to future work, but include a sensitivity analysis in the experimental section to demonstrate the impact of bandwidth choice.

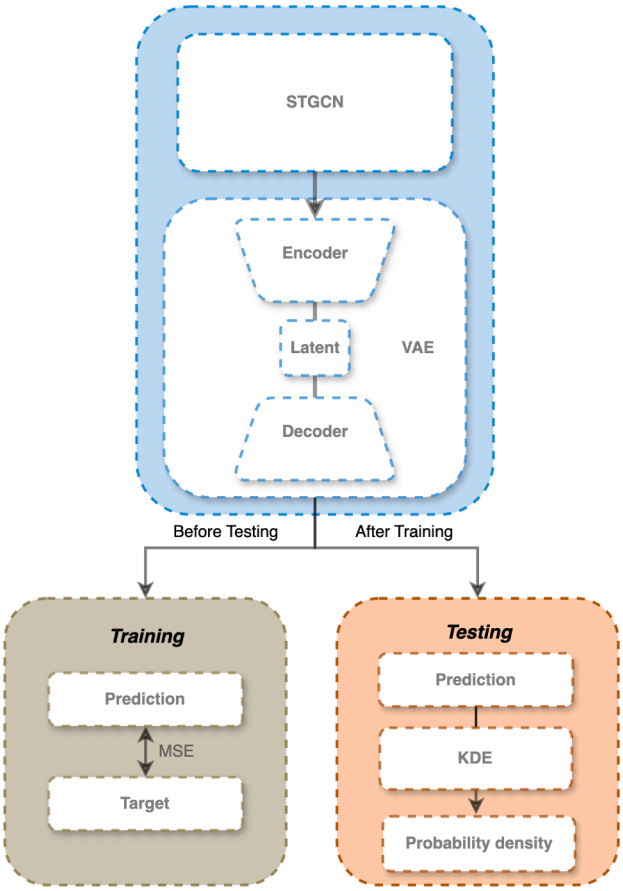


Fig. 4. Training and testing process of the STGCN-VAE.

3.3. *Training process: Spatiotemporal learning and latent representation construction*

The training process begins with time series data $X^T \in \mathbb{R}^{T\times N}$ and Chebyshev polynomials $\mathcal{T}_k(L)$ of the graph Laplacian L, which capture spatial dependencies. These inputs are processed by two stacked Spatio-Temporal Convolution (ST-Conv) blocks, each consisting of a temporal convolution, a spatial convolution using $\mathcal{T}_k(L)$, another temporal convolution, and a normalization layer. The resulting features are passed to an output block with two temporal convolution layers and a fully connected layer. The output is fed into the encoder of a Variational Autoencoder (VAE), which compresses it into a latent space of dimension d. The model applies the reparameterization trick to sample latent vectors $z \sim \mathcal{N}(\mu, \sigma)$, decodes them into a vector of predictions, and computes the Mean Squared Error (MSE) loss between the predicted mean and the ground-truth future time steps. The parameters are updated via backpropagation until convergence is achieved. A simplified illustration of the training process is shown in the left 'Training' branch of Fig. 4, and the formalized workflow of the training process is summarized in Algorithm 1.

3.4. *Testing process: Probabilistic forecasting*

The testing process leverages the trained STGCN-VAE model to generate probabilistic forecasts of travel demand. The testing phase follows the same data processing steps as in the training phase, ensuring consistency in the input representation. Specifically, the input time series data X^T and Chebyshev polynomials $\mathcal{T}_k(L)$ are fed through the ST-Conv blocks and VAE encoder to obtain the latent representation. However, unlike in the training phase, the objective during testing is to generate multiple samples for each time step, thereby capturing a range of potential future outcomes. This is achieved by repeatedly sampling latent vectors z using the reparameterization trick. The decoder then produces multiple demand predictions for each sampled z, effectively generating a set of potential future demand values.

These samples are then fed into the Kernel Density Estimation (KDE) module, which constructs a continuous probability density function for the forecasted demand. The KDE module processes the set of generated samples to estimate the probability distribution of travel demand at the next time step. This distribution provides a comprehensive representation of forecast uncertainty, rather than a single-point estimate. A schematic representation of the testing process is shown in the right 'Testing' branch of Fig. 4, and the detailed workflow is outlined in Algorithm 2.

Algorithm 1 STGCN-VAE training process.

Input: Time series data $X^T \in \mathbb{R}^{T \times N}$, Chebyshev polynomials $\mathcal{T}_k(L)$
Output: A list of predicted samples.
for each training epoch **do**
    Feed X^T and $\mathcal{T}_k(L)$ into the first ST-Conv block
    for each ST-Conv block **do**
        Apply temporal convolution
        Apply spatial convolution using $\mathcal{T}_k(L)$
        Apply another temporal convolution
        Apply normalization
    end for
    Feed output into the next ST-Conv block
    Feed ST-Conv output into the output block
        Apply two temporal convolution layers
        Apply a fully connected layer
    Feed result into VAE encoder
    Encode into latent representation
    Sample latent vectors using the reparameterization trick
    Decode latent vectors to reconstruct predicted samples
    Compute the mean of predicted samples
    Compute $\mathcal{L}_{\text{MSE}}$ between the mean and the ground-truth future target
    Backpropagate to update parameters
end for

It is worth noting that, in our framework, we train the model by minimizing the MSE between the mean of generated samples and the observed demand, while evaluation employs probabilistic metrics to assess the full predictive distribution. KDE is then applied post-training solely for visualization and summary statistics, not during optimization. We use MSE as a surrogate loss because it provides stability, scales well, and reduces computational load compared to distribution-aware losses such as CRPS or NLL. In a non-parametric, sample-based setting, those losses are resource-intensive because they require extensive calculations over the entire predictive distribution. By adopting MSE, our large graph-based model consistently delivers robust results, effectively supporting diverse scenario generation while maintaining computational efficiency.

Algorithm 2 STGCN-VAE testing process: Probabilistic forecasting.

Input: Time series data $X^T \in \mathbb{R}^{T \times N}$, Chebyshev polynomials $\mathcal{T}_k(L)$, trained model parameters $\mathcal{W}^*$
Output: Probabilistic distribution of forecasted values
Obtain latent representation as in Algorithm 1 (steps 3–16) using trained parameters $\mathcal{W}^*$
Sample multiple latent vectors using reparameterization trick
Decode latent vectors to obtain multiple predicted samples
Feed predicted samples into Kernel Density Estimation (KDE)
Construct continuous probability density function from KDE
Return the probabilistic distribution

4. Experiments

We conduct a series of extensive experiments to evaluate the proposed method introduced in Section 3. We assess its performance on real-world data and compare it with eight baseline models using six evaluation metrics, covering both point and probabilistic forecasting accuracy.

4.1. Experimental setup

4.1.1. Data and experimental settings

We conduct experiments using four different datasets (all for the months of January-March):

- Yellow Taxi Trip Records (New York City TLC, 2024)[1]: a public dataset that includes all MoD travel orders in New York city, containing 9,554,778 trip records.

[1] https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page

Table 2
Distribution of zone-level travel demand aggregated during the PM peak hour (18:00) across all days in the NYC and Chicago datasets.

Distribution	NYC Counts	NYC High Demand	Chicago Counts
Normal	13	5	10
Gamma	8	8	28
Beta	0	0	22
Lognormal	48	20	17
None	193	30	0
Total	**262**	**63**	**77**

- Washington DC Open Data Taxi Trips (2024)[2]: a public dataset that contains all taxi trips in Washington DC, containing 640,720 trip records.
- Chicago Rideshare Trips (2025)[3]: a public dataset reported by Transportation Network Providers (rideshare companies), containing 22,934,065 records.
- Capital Bikeshare Data (Washington DC, 2025)[4]: a public dataset recording bikeshare trips from the Capital Bikeshare system, containing 1,281,301 records.

All datasets record individual trips with metadata such as start time, start location, end time, and end location. In our study, we retain only the start time and start location, discarding other fields. For the first three datasets, trips are first mapped to regions based on postcode. Since the original data are collected at precise time points, we aggregate them into one-hour intervals. A new field, demand number, is then created to indicate the total number of trips in each interval.

For the Capital Bikeshare data, unlike the others which record pickup locations, the dataset reports origin stations (over 800 in total). To reduce data sparsity and improve consistency, we first aggregate stations by postcode and then apply the same hourly aggregation procedure. This significantly reduces both the data scale and sparsity, as many stations are geographically close and individually exhibit very low demand.

Fig. 5 visualizes the spatial divisions and average hourly demand across all datasets. Even though the absolute demand levels are not directly comparable due to differences in region definitions, the figure illustrates the distinct spatial demand patterns present in each dataset, highlighting the diversity of service modes and urban contexts considered in our evaluation.

After the data processing, we split the entire data into training set (Jan. 1st - Mar. 13th), validation set (Mar. 14th - Mar. 21st) and test set (Mar. 22nd - Mar. 31st) for training, validation, and testing, respectively. For each time series training round, the input data is structured as a $T \times n$ table, where T represents time steps and n denotes the regions. Each element in the table represents the travel demand at a given time and region.

To better understand regional travel demand distributions, we tested the New York and Chicago datasets against several candidate distributions using a p-value threshold of 0.05. We focused on the peak hour of 18:00, where demand was aggregated at the region level across all days. For each zone, the value corresponds to the total number of requests observed during this hour. As shown in Table 2, most New York regions do not conform to standard distributional forms. By contrast, in Chicago all 77 zones admit a parametric fit, but they split across families—10 normal, 28 gamma, 22 beta, and 17 lognormal—so no single form describes the city.

All experiments are conducted on a single Apple Silicon M1 Pro-with 32 GB of memory, using the PyTorch framework. Baselines are implemented based on their official source code and parameter settings as described in the respective original studies. The datasets are split into training, validation, and test sets in an 8:1:1 ratio. The best-performing model on the validation set is selected for testing. The prediction horizon is set to 3 steps. We use the RMSProp optimizer with an initial learning rate of 1×10^{-3}. The latent space embedding dimension for both μ and σ is set to 64.

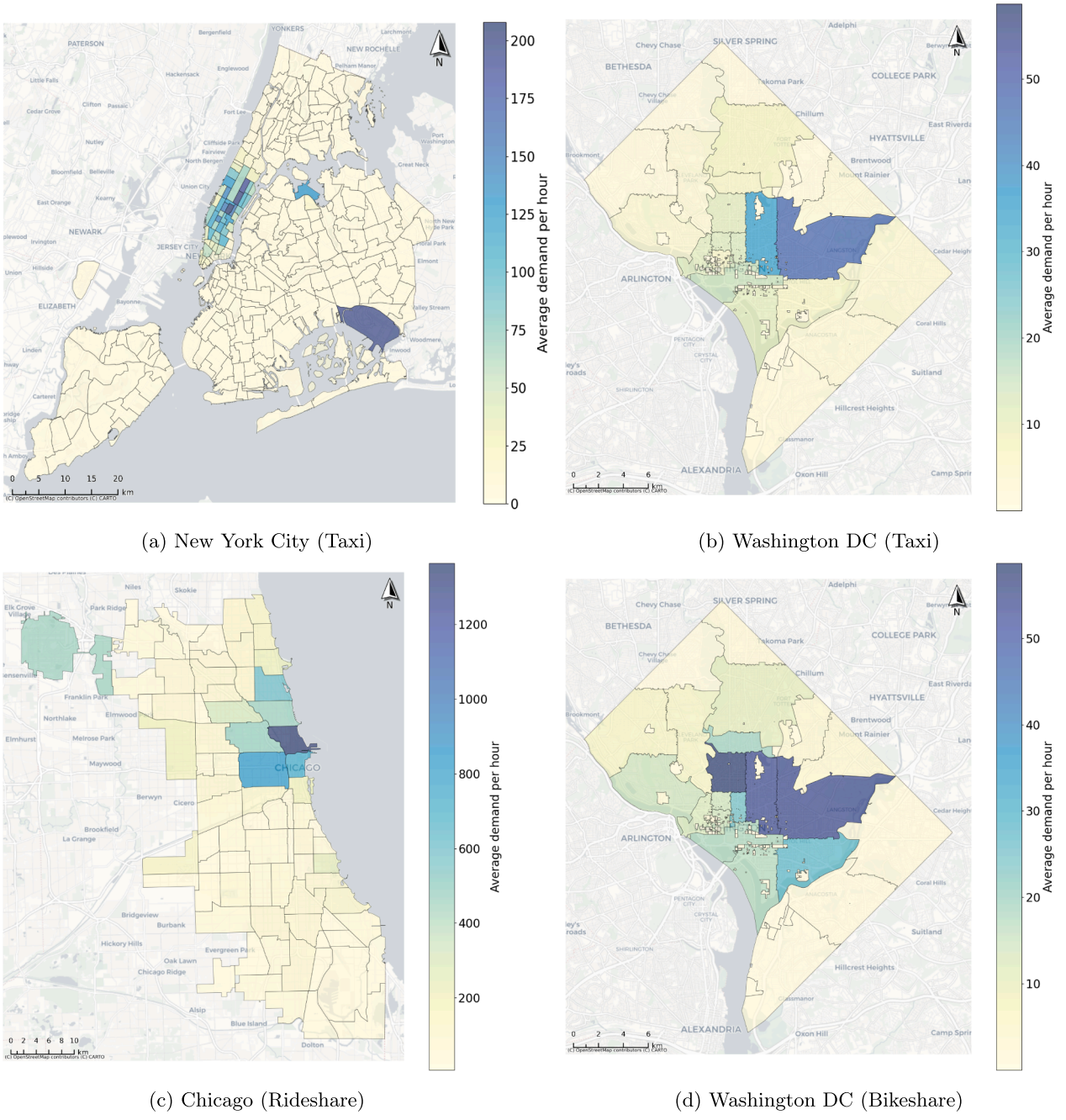
4.1.2. Evaluation metrics

Six metrics are used to evaluate the performance of our approach. First, we use the mean absolute error (MAE) and root mean squared error (RMSE) to evaluate the performance for point estimation based on the mean. These two metrics evaluate the prediction accuracy and are defined as:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|x_i - \hat{x}_i\right| \tag{5}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \hat{x}_i\right)^2} \tag{6}$$

[2] https://dcgov.app.box.com/v/TaxiTrips2024
[3] https://data.cityofchicago.org/Transportation/Transportation-Network-Providers-Trips-2025-/6dvr-xwnh/about_data
[4] https://capitalbikeshare.com/system-data



(a) New York City (Taxi)

(b) Washington DC (Taxi)

(c) Chicago (Rideshare)

(d) Washington DC (Bikeshare)

Fig. 5. Average hourly travel demand heatmaps across four datasets. Darker regions represent higher demand.

where $\hat{x}_i$ and x_i are the predicted and ground-truth demand values of the i^{th} region, respectively, and n is the total number of regions.

In addition, to quantify uncertainty, we use three metrics, namely Mean Prediction Interval Width (MPIW), Continuous Ranked Probability Score (CRPS), and Interval Score (IS) to evaluate the performance of probabilistic forecasting (Gneiting and Raftery, 2007).

- MPIW assesses the average width of the prediction intervals, indicating interval tightness. A smaller MPIW suggests more confident predictions. It is defined as:

$$\text{MPIW} = \frac{1}{n}\sum_{i=1}^{n}\left(U_i - L_i\right) \tag{7}$$

where L_i and U_i represent the lower and upper bounds of the prediction interval for region v_i, respectively, which are computed with 10 % to 90 % confidence interval.

- CRPS treats the observed value as a step function in the form of a 0-1 CDF, providing a quadratic measure of discrepancy. It is formulated as:

$$\mathrm{CRPS}(F, x) = \int \left(F(x_i) - \mathbf{1}_{\{x_i \geq x\}} \right)^2 dx \tag{8}$$

where F is the predicted CDF, x_i is the predicted travel demand and x is the ground truth travel demand. $\mathbf{1}(\cdot)$ is an indicator function, the value is 1 when $x_i > x$, otherwise 0. A lower CRPS indicates that the predicted distribution is closer to the observed outcome.

- IS evaluates the quality of prediction intervals, it balances the width of the prediction interval and the penalties for observed values falling outside the interval. It is defined as:

$$\mathrm{IS}_\alpha(L, U, x) = (U - L) + \frac{2}{\alpha}(L - x) \cdot \mathbf{1}(x < L) + \frac{2}{\alpha}(x - U) \cdot \mathbf{1}(x > U) \tag{9}$$

where L and U represent the lower and upper bounds of the prediction interval for region v_i, for simplicity, we omit the subscript v_i. x is the ground truth travel demand, α is the confidence level, and $\mathbf{1}(\cdot)$ is an indicator function that equals 1 when the condition inside is true and 0 otherwise. A lower Interval Score indicates better calibration and sharpness of the prediction interval.

- Running time measures computational efficiency, in which we record both training and testing times (in minutes). Training time is measured from model initialization to convergence, while testing time refers to the total duration required to generate predictions for the entire test set.

4.1.3. Baselines

We compare our proposed STGCN-VAE model with several benchmarks, including one point forecasting model and six probabilistic forecasting models:

- **STGCN** (Yu et al., 2017): A spatiotemporal graph convolutional network designed for point forecasting on graph-structured time series data. It serves as a strong baseline for deterministic forecasting and forms the foundation for our proposed STGCN-VAE model.
- **STAWnet** (Tian and Chan, 2021): Spatial-Temporal Attention Wavenet, an end-to-end multi-step traffic prediction model that captures complex spatial-temporal dependencies using temporal convolution and self-attention with self-learned node embeddings.
- **STGCN + Normal**: An extension of STGCN that incorporates a normal distribution over its predictions. By modeling the output as a Gaussian, it provides probabilistic forecasts, estimating uncertainty while retaining STGCN's spatiotemporal modeling capabilities.
- **STGCN + Log-normal**: A variant of STGCN that uses a log-normal distribution for probabilistic forecasting. This approach is suited for data with positive values and skewed distributions, offering an alternative uncertainty estimation method compared to the normal distribution variant.
- **DeepAR** (Salinas et al., 2020): A deep autoregressive recurrent network designed for probabilistic forecasting. It uses an RNN architecture to model global time series, incorporating a likelihood term to produce uncertainty estimates across all input features. DeepAR is particularly effective for capturing temporal dependencies in large-scale time series datasets.
- **BNN-LSTM** (Zhu and Laptev, 2017): A Bayesian LSTM network with two LSTM layers, enabling probabilistic forecasting through Bayesian inference. By applying Monte Carlo dropout during inference, it estimates predictive uncertainty, making it suitable for time series tasks requiring robust uncertainty quantification.
- **DGGP** (Jiang et al., 2022): Deep graph Gaussian processes that combine graph-based spatial modeling with Gaussian processes for probabilistic forecasting. It uses RBF and Cosine kernels to capture complex spatiotemporal patterns, providing a flexible framework for uncertainty estimation on graph-structured data.
- **Prob-GNNs** (Wang et al., 2024) A probabilistic graph neural network framework that quantifies spatiotemporal uncertainty in travel demand, robustly predicting ridership patterns under domain shifts and revealing peak-hour and high-volume area uncertainties.

4.2. Experimental results

In the following we present an in-depth analysis of the experimental results, focusing on the predictive accuracy, uncertainty quantification, and computational efficiency of the proposed STGCN-VAE model. We compare its performance with eight baseline models under various demand conditions using both point and probabilistic forecasting metrics. To better understand the model's behavior, we also examine how results vary across high- and low-demand regions, explore how uncertainty varies by urban context and time, and investigate the differences between parametric and nonparametric modeling approaches.

4.2.1. Predictive performance

A detailed comparison of the predictive performance is presented in Table 3. MAE and RMSE are evaluated across all seven models, while CRPS, MPIW, and IS are assessed for the seven probabilistic models. To better analyze performance differences, regions are categorized into low- and high-demand groups based on an average hourly demand threshold of 10, mitigating the dominance effect of high-demand regions. Each model's results are averaged over five runs. The best performing model per evaluation metric is highlighted in bold.

Table 3
Comparison of predictive results across models on the *New York City Trip Record dataset*. (S + N: STGCN + Normal, S + L: STGCN + Lognormal, S + V: STGCN + VAE, our proposed model; model names abbreviated due to space constraints).

Metric	Region	Point forecasting		Probabilistic forecasting						
				Parametric				Non-parametric		
		STGCN	STAWnet	S + N	S + L	BNN-LSTM	Prob-GNNs	DeepAR	DGGP	S + V
MAE	All	5.7487	11.9538	5.8504	24.3598	10.0087	16.9312	11.0107	8.4483	**5.4094**
	Low demand	0.9431	2.0633	0.9548	1.7435	1.6095	14.8306	**0.7750**	1.4297	0.8815
	High demand	27.2737	26.9945	27.7788	125.6621	41.8215	26.3403	56.8581	37.5961	**25.6908**
RMSE	All	29.3497	21.5188	26.0853	64.9564	32.5472	24.1395	35.6128	35.2685	**19.9317**
	Low demand	2.4184	3.4835	2.3556	4.0361	3.4655	15.0214	2.7630	3.2143	**2.1532**
	High demand	68.5096	28.7667	60.8557	151.8075	56.0011	46.7131	83.1558	62.3789	**46.2324**
MPIW	All	–	–	29.4285	13761.1612	31.2340	39.8369	29.1572	34.8921	**23.8823**
	Low demand	–	–	3.3382	16234.3577	4.6783	7.8204	**1.2152**	5.1234	6.0574
	High demand	–	–	146.2911	2683.3019	154.9125	114.9107	154.3143	167.4532	**103.7229**
CRPS	All	–	–	22.8911	24.4993	19.8754	34.6295	20.9385	21.3467	**17.1134**
	Low demand	–	–	3.5585	2.2019	1.5432	29.8844	**0.9861**	1.6789	1.0643
	High demand	–	–	109.4855	124.3728	98.7654	155.8835	110.3097	105.4321	**88.9996**
IS	All	–	–	76.7607	13761.1699	38.5476	39.8369	41.9638	45.1234	**19.4128**
	Low demand	–	–	35.5485	16234.3681	4.8765	23.8254	**3.5794**	5.2345	3.8665
	High demand	–	–	261.3572	2683.3020	198.7654	114.7159	213.8939	220.9876	**89.0475**
Time	Train(min)	15	40	15	15	35	50	45	210	17
	Test(min)	1	5	1	1	5	8	5	20	2

As presented in Table 3, our proposed STGCN-VAE model (shaded in grey, last column) outperforms the baseline models across most metrics. For *point forecasting*, the results are presented in the first two rows of the table. The first row shows that our model achieves an MAE of 5.41 across all regions, a 5.9 % improvement over STGCN, where STGCN is the strongest baseline for this metric. In high-demand regions, it achieves an MAE of 25.69, reducing the error by 5.8 % compared to STGCN, which outperforms other baselines. For RMSE, the second row indicates that our model achieves 19.93 across all regions, a 23.6 % reduction relative to STGCN + Normal, the best baseline for this metric. In high-demand regions, it yields 46.23, improving performance by 24 % over STGCN + Normal. Although our model is primarily designed for probabilistic forecasting, it consistently outperforms STGCN in point forecasting across various metrics. This result underscores the robustness of our approach, as it not only captures uncertainty more effectively but also delivers more accurate point estimates than a model specifically optimized for deterministic predictions.

For *probabilistic forecasting*, our model consistently outperforms on average all baselines across regions, with particularly strong performance in high-demand areas. As shown in the third row of Table 3, it achieves an 18 % improvement in MPIW over STGCN + Normal across all regions and a 29 % improvement over STGCN + Normal in high-demand regions, where STGCN + Normal is the best-performing baseline for this metric. For CRPS in the fourth row, it improves by 13.8 % overall and by 9.8 % in high-demand regions over BNN-LSTM, with BNN-LSTM being the strongest baseline for this metric. Finally, for IS in the fifth row, it achieves a 49.6 % improvement across all regions and a 55.2 % improvement in high-demand regions over BNN-LSTM, the strongest baseline here. These results highlight the proposed approach STGCN + VAE's ability to generate high-quality predictive distributions, especially in high-demand regions.

However, in low-demand regions, our model slightly underperforms compared to DeepAR. This is likely due to data sparsity. In low-demand areas, travel demand is often zero for most time steps, which makes it difficult for sample-based methods like STGCN-VAE to learn meaningful patterns. Since such methods generate predictions by sampling from learned distributions, excessive zeros hinder accurate distribution learning and can result in higher uncertainty or biased estimates. This challenge is further amplified by the graph-based nature of our model, which relies not only on a region's own history but also on information from structurally related neighbors. Empirically, low-demand regions are highly likely to be surrounded by other low-demand regions, meaning that both local and neighboring signals are sparse, which further degrades learning quality. In contrast, time series forecasting models such as DeepAR, which directly estimate the mean and variance, are generally more robust in sparse settings, as they focus primarily on capturing central tendencies in historical data rather than learning from both spatial and temporal dependencies. However, such models often underperform in high-demand regions, where uncertainty is higher and spatial-temporal correlations are more complex. Importantly, our model performs strongly in high-demand regions, which are more critical in practice as they account for the majority of ride requests and directly influence operational decisions such as fleet relocation, pricing, and matching.

Furthermore, our approach significantly outperforms DeepAR, BNN-LSTM, and DGGP in terms of computational efficiency. Table 3 records training and testing times in the last row. Training time is measured from initialization to convergence, showing that STGCN-based models are more efficient than others due to their effective spatiotemporal dependency learning mechanism. In particular, it reduces training and testing time by more than half relative to DeepAR, with a 62 % reduction in training time (17 vs. 45 mins) and a 60 % reduction in testing time (2 vs. 5 mins). Relative to other baselines, our model also achieves significant improvements: STAWnet (58 % faster training, 60 % faster testing), BNN-LSTM (51 % faster training, 60 % faster testing), DGGP (92 % faster training, 90 % faster testing), and Prob-GNNs (66 % faster training, 75 % faster testing). While parametric models predict only the mean and variance, our approach remains comparably efficient despite generating 100 samples per inference via the VAE decoder. This balance

Table 4
Comparison of predictive results across models on *Washington DC taxi dataset*. (S + N: STGCN + Normal, S + L: STGCN + Lognormal, S + V: STGCN + VAE, our proposed model; model names abbreviated due to space constraints).

Metric	Region	Point forecasting		Probabilistic forecasting						
				Parametric				Non-parametric		
		STGCN	STAWnet	S + N	S + L	BNN-LSTM	Prob-GNNs	DeepAR	DGGP	S + V
MAE	All	1.1378	4.1979	1.2021	2.0125	1.4046	1.7656	1.4483	2.0087	**0.9179**
	Low demand	0.0465	0.2552	0.0531	1.1639	**0.0054**	0.0652	0.3297	0.3095	0.0115
	High demand	2.2292	4.2634	2.3511	3.5990	3.0331	3.5183	7.5961	6.8215	**1.8243**
RMSE	All	4.9402	7.7103	5.1010	5.2927	6.1319	8.0275	7.2685	5.5472	**3.5404**
	Low demand	0.0926	0.9402	0.0860	1.1639	0.0735	0.1076	0.7143	0.5655	**0.0661**
	High demand	6.9859	7.7730	7.2134	8.8234	9.0200	11.4391	12.3789	9.0011	**5.0065**
MPIW	All	–	–	2.9804	855.6571	5.7355	2.5313	7.8921	6.2340	**1.5811**
	Low demand	–	–	0.4281	937.8258	**0.0001**	1.4026	1.1234	0.9783	0.1106
	High demand	–	–	5.5327	702.0375	12.4111	3.6947	18.4532	20.9125	**3.0516**
CRPS	All	–	–	1.8668	1.8128	**0.7744**	1.5080	5.3467	5.8754	1.8417
	Low demand	–	–	0.0453	0.9529	**0.0001**	0.0775	0.3789	0.5432	0.0177
	High demand	–	–	3.6883	3.4205	**1.6758**	3.1423	19.4321	16.7654	3.6657
IS	All	–	–	4.5671	855.6620	7.3894	52.6456	7.1234	7.5476	**3.2792**
	Low demand	–	–	0.4434	937.8311	**0.0217**	1.4638	1.2345	1.1765	0.1257
	High demand	–	–	8.6909	702.0413	15.9649	105.4023	40.9876	35.7654	**6.4328**
Time	Train(min)	13	35	15	15	40	45	180	35	15
	Test(min)	1	5	1	1	5	8	16	5	2

Table 5
Comparison of predictive results across models on *Chicago rideshare dataset*.(S + N: STGCN + Normal, S + L: STGCN + Lognormal, S + V: STGCN + VAE, our proposed model; model names abbreviated due to space constraints).

Metric	Region	Point forecasting		Probabilistic forecasting						
				Parametric				Non-parametric		
		STGCN	STAWnet	S + N	S + L	BNN-LSTM	Prob-GNNs	DeepAR	DGGP	S + V
MAE	All	40.3073	24.1664	55.8454	59.9482	50.4321	66.4161	47.5363	42.3894	**23.8814**
	Low demand	7.9235	7.1227	10.7209	7.9072	8.0470	11.4898	10.7983	7.2380	**5.8113**
	High demand	73.5434	**41.5733**	102.1573	113.3587	209.7438	122.7878	85.2411	188.9517	42.4271
RMSE	All	116.2122	69.0882	149.4167	185.1795	162.3815	207.5255	122.1309	174.2864	**67.8297**
	Low demand	11.9019	11.1874	14.7481	11.1761	17.4629	16.227	14.9733	16.0873	**8.4144**
	High demand	164.9865	97.5690	212.1673	263.3574	282.9187	294.9521	173.1888	310.3476	**96.1778**
MPIW	All	–	–	200.0194	219.7879	**156.4732**	222.3351	199.8660	171.9284	170.7719
	Low demand	–	–	37.6611	70.2834	43.9248	45.3467	58.4643	55.6125	**35.6497**
	High demand	–	–	366.6503	373.2268	772.5819	429.5074	344.9886	836.2745	**309.4499**
CRPS	All	–	–	147.0282	102.9417	**98.6527**	164.1430	170.7946	207.9842	145.3260
	Low demand	–	–	32.7299	**20.2880**	27.8546	39.9279	26.8846	38.4012	33.9377
	High demand	–	–	264.3343	287.7705	497.3825	319.7848	326.1235	528.6197	**259.6456**
IS	All	–	–	222.5982	231.3913	193.1748	2294.9114	480.1473	225.6841	**185.4911**
	Low demand	–	–	41.2715	70.7689	**24.6971**	233.4164	109.9827	26.1824	38.4222
	High demand	–	–	408.6967	396.2406	994.2816	4410.6558	860.0535	1106.4739	**336.4302**
Time	Train(min)	13	20	13	13	35	40	30	100	17
	Test(min)	1	3	1	1	5	6	3	10	2

between computational efficiency and probabilistic forecasting capability makes it well-suited for online applications, such as real-time demand prediction in ride-hailing platforms, dynamic pricing systems, and adaptive fleet management.

An interesting observation from Table 3 is that STGCN + Lognormal performs significantly worse than STGCN + Normal, even though the only difference is the use of a lognormal instead of a Gaussian distribution. In contrast, our VAE-based approach learns the predictive distribution in a data-driven, non-parametric manner, avoiding reliance on rigid distributional forms. This flexibility allows it to better capture complex, multimodal, or skewed demand patterns, leading to more reliable uncertainty quantification. This is especially valuable in real-world ride-hailing settings where demand distributions often deviate from standard parametric shapes, as we verified in Table 2.

As for the Washington DC taxi dataset and the Chicago rideshare dataset, shown in Tables 4 and 5, our proposed STGCN-VAE model demonstrates consistent performance advantages across both point and probabilistic forecasting tasks.

As reported in Table 4, our model achieves the strongest overall results on the Washington DC taxi dataset. For point forecasting, it delivers the lowest errors across both MAE and RMSE. Averaged over all regions, STGCN-VAE reaches an MAE of 0.92, improving upon STGCN—the strongest baseline—by 19.3 %. In high-demand regions, it reduces MAE to 1.82, a clear improvement over all competing methods. Similarly, for RMSE, our model achieves 3.54 overall and 5.00 in high-demand areas, corresponding to error

Table 6
Comparison of predictive results across models on *Washington DC bikeshare dataset*. (S + N: STGCN + Normal, S + L: STGCN + Lognormal, S + V: STGCN + VAE, our proposed model; model names abbreviated due to space constraints).

Metric	Region	Point forecasting		Probabilistic forecasting						
				Parametric				Non-parametric		
		STGCN	STAWnet	S + N	S + L	BNN-LSTM	Prob-GNNs	DeepAR	DGGP	S + V
MAE	All	9.3599	8.7593	9.4426	20.4495	8.9576	11.9240	12.4326	9.5373	**8.1848**
	Low demand	**1.1623**	1.9231	1.5357	2.1500	1.4478	1.4806	1.6397	1.2714	1.3609
	High demand	18.0698	**12.6734**	17.8437	39.8928	36.9532	23.0201	23.9000	33.2245	15.4351
RMSE	All	23.8387	**18.3119**	22.2216	44.9135	28.8924	29.9681	26.4479	31.4263	21.5475
	Low demand	2.4538	3.1416	**2.3496**	3.5986	3.0123	2.8551	2.9699	2.8476	2.7682
	High demand	34.1422	**22.8374**	31.8210	64.3953	49.8523	42.9377	37.8594	55.3071	30.8134
MPIW	All	–	–	19.1583	664.0077	27.6152	38.2897	30.3608	30.4265	**11.6821**
	Low demand	–	–	3.0921	947.9519	4.0937	4.3600	4.3214	4.5662	**1.6972**
	High demand	–	–	36.2287	362.3170	137.8156	112.4650	58.0277	148.5621	**22.2911**
CRPS	All	–	–	16.6771	20.0856	17.3265	21.0804	11.9945	18.7534	**11.8485**
	Low demand	–	–	2.0423	2.0019	1.3421	**1.0369**	1.4654	1.4573	1.2460
	High demand	–	–	32.2265	39.2994	87.5276	31.7516	23.1817	94.2315	**23.1136**
IS	All	-	-	33.7017	669.2824	34.1068	364.1973	45.0575	40.5634	**33.2454**
	Low demand	–	–	5.9053	947.9544	**4.3264**	23.1970	6.9978	4.6932	5.3133
	High demand	–	–	63.2354	373.1933	176.8291	726.5094	85.4959	197.4532	**62.9232**
Time	Train(min)	**8**	15	**8**	**8**	16	18	17	35	**8**
	Test(min)	**1**	5	**1**	**1**	2	4	2	8	**1**

reductions of 28.4 % and 28.3 %, respectively, compared to the best baseline. These results highlight that even though our approach is tailored for probabilistic forecasting, it also yields highly competitive point predictions.

In terms of probabilistic forecasting, our model consistently provides narrower and more reliable predictive intervals. Across all regions, it achieves the smallest MPIW (1.58), reducing interval width by 37.6 % compared to the next best model (Prob-GNNs). This improvement holds in high-demand regions, where interval sharpness is crucial. For CRPS, BNN-LSTM attains slightly lower values in sparse areas, but STGCN-VAE remains competitive, particularly in high-demand settings, where it achieves 3.67 compared to 1.68 for BNN-LSTM. Importantly, for IS, our model records 3.28 overall and 6.43 in high-demand regions, outperforming all baselines by a large margin. While parametric approaches such as BNN-LSTM benefit from robustness in extremely sparse regions, they fall short in capturing the complex dynamics of dense urban demand. By contrast, STGCN-VAE excels in high-demand regions, which dominate practical ride-hailing operations and inform key decisions such as fleet allocation, pricing, and relocation.

As shown in Table 5, similar trends are observed on the Chicago rideshare dataset. For *point forecasting*, STGCN-VAE again achieves the strongest overall performance, recording the lowest MAE (23.88 overall) and RMSE (67.83 overall), improving upon the best baselines (STAWnet) by 1.2 % and 1.8 %, respectively. In high-demand regions, which are the most challenging, our model attains an MAE of 42.43 and RMSE of 96.18, remaining competitive with STAWnet and outperforming all other models. For *probabilistic forecasting*, STGCN-VAE consistently delivers sharper and more reliable predictive intervals, achieving the lowest MPIW in both low- and high-demand regions, as well as the lowest IS overall (185.49), surpassing all baselines. While BNN-LSTM and STGCN + Lognormal show slightly better CRPS in low-demand settings, STGCN-VAE outperforms them in high-demand regions, where accuracy and uncertainty calibration are most critical. In addition to accuracy, STGCN-VAE is also computationally efficient. Training requires only 17 mins and testing 2 mins, which is significantly faster than more complex probabilistic baselines such as Prob-GNNs (40/6 mins) and DGGP (100/10 mins), while still achieving stronger predictive performance. This balance between predictive accuracy and efficiency confirms the practicality of our approach for large-scale rideshare demand forecasting.

For the experiment on the Washington DC bikeshare dataset, as reported in Table 6, our proposed STGCN + VAE model delivers strong performance, particularly in probabilistic forecasting, with competitive point forecasting and high efficiency for the Washington DC bikeshare dataset.

For *point forecasting*, STGCN + VAE achieves an overall MAE of 8.18, improving on STGCN (9.36) by 12.6 % and outperforming STAWnet (8.76, 6.6 % worse). In high-demand regions, STGCN + VAE's MAE (15.44) improves on STGCN (18.07) by 14.6 %, though STAWnet leads (12.67, 25.6 % better). For RMSE, STGCN + VAE records 21.55 overall, 9.6 % better than STGCN (23.84) but behind STAWnet (18.31, 15.1 % better). In high-demand areas, STGCN + VAE's RMSE (30.81) improves on STGCN (34.14) by 9.8 %. While not the top performer, STGCN + VAE is competitive despite its probabilistic focus.

In *probabilistic forecasting*, STGCN + VAE excels with the smallest MPIW (11.68 overall), reducing interval width by 57.7 % over BNN-LSTM (27.62), with similar gains in low-demand (1.70, 45.1 % better than STGCN + Normal) and high-demand (22.29, 38.5 % better than STGCN + Normal). For CRPS, STGCN + VAE achieves 11.85 overall, slightly outperforming DeepAR (11.99, 1.2 % better), and in high-demand areas, its 23.11 nearly matches DeepAR (23.18, 0.3 % worse) while far surpassing BNN-LSTM (87.53, 73.6 % better). For IS, STGCN + VAE delivers 33.25 overall, improving on STGCN + Normal (33.70) by 1.3 %, and in high-demand regions, its 62.92 is 0.5 % better than STGCN + Normal (63.24), while significantly lower than BNN-LSTM (176.83, 64.4 % better). STGCN + VAE matches the fastest training (8 min) and testing (1 min) times, equalling STGCN and STGCN + Normal while outperforming DGGP (35

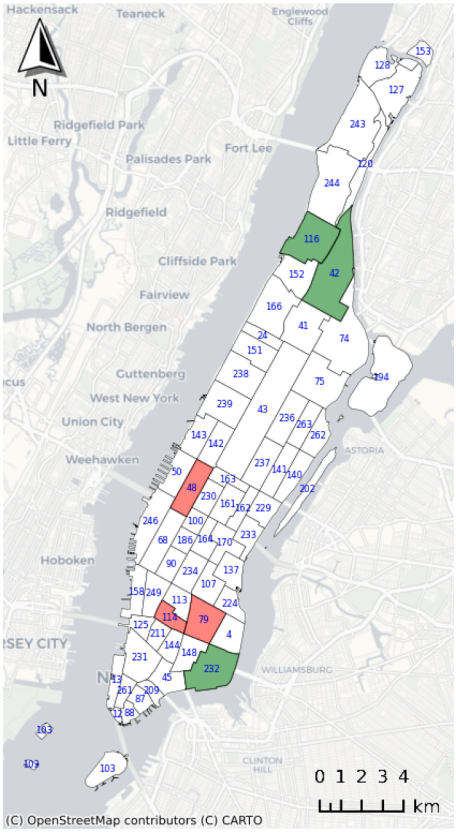


Fig. 6. Illustration of six selected regions: red denotes high-demand areas, while green denotes low-demand areas.

and 8 min). Its tight intervals and reliable uncertainty make STGCN + VAE ideal for high-demand bike-sharing scenarios, enhancing decisions like bike rebalancing, despite some trade-offs in point forecasting accuracy.

To visually illustrate the model's performance, we select six representative regions from the New York, including three high-demand and three low-demand areas, as shown in Fig. 6. This selection reflects a range of traffic conditions.

Fig. 7 compares the ground truth values with the ordered prediction intervals over a continuous 100-hour period. The orange line indicates the observed values, while the blue line and the shaded area represent the predicted values and the 90 % predictive interval, respectively. In high-demand regions, the predicted mean (blue line) closely follows the observed values (orange line), and the predictive intervals remain well-calibrated and appropriately narrow. This suggests that our model effectively captures both central tendencies and uncertainty, providing reliable probabilistic forecasts. Such accurate and uncertainty-aware predictions are essential for decision-making in dynamic ride-hailing environments, where dispatching and pricing strategies depend on precise demand estimates. A closer examination of the figure reveals that the highest predictive uncertainty does not always coincide with peak demand periods. This observation aligns with real-world travel demand patterns, where uncertainty is influenced by various factors such as traffic conditions, event-driven fluctuations, and external disruptions rather than demand magnitude alone. However, parametric models that assume a Gaussian distribution often struggle to capture this nuanced behavior. These models explicitly predict the mean and variance under a fixed functional relationship, typically leading to the assumption that uncertainty grows monotonically with demand. As a result, they may overestimate or underestimate uncertainty in certain scenarios. As further illustrated in Fig. 7, low-demand regions show lower prediction accuracy and interval coverage compared to high-demand regions. This underperformance is expected: in areas where both the region and its neighbors have very few non-zero demand values, the model has limited data to learn from, and the generated forecasts remain conservative. In such cases, short, unexpected increases in demand may not be captured by the predicted interval. However, this behavior also reflects a strength of the model: it avoids reacting to noise or overestimating demand where data are sparse. Even though there is still room to improve performance in these areas, the model remains reliable and well-calibrated in high-demand regions, which are most critical for practical applications. These observations highlight the importance of future work that specifically targets sparse regions, without compromising performance in high-demand regions.

4.2.2. Uncertainty quantification

Next, we present our findings on uncertainty quantification by displaying prediction intervals for three representative regions in Manhattan. We do so for regions with different characteristics: workplace (Region 161), tourism (Region 230), and residential (Region 238), as shown in Fig. 8.

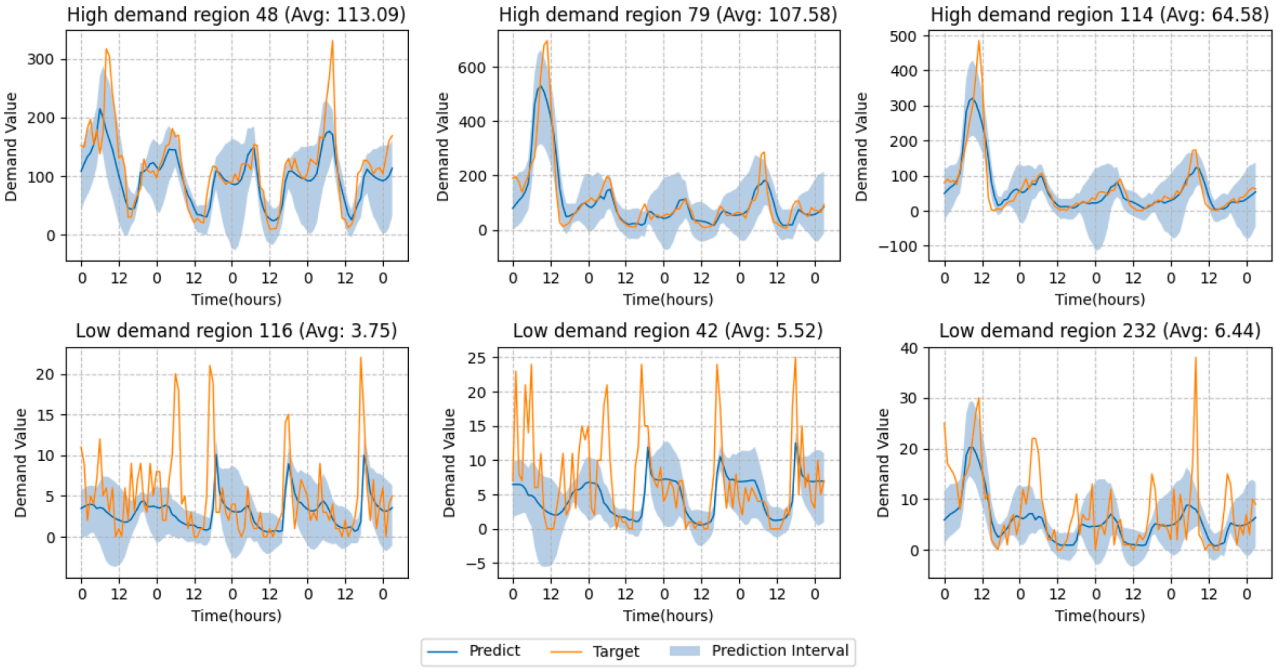


Fig. 7. Comparison of targets and predictions across six regions (top: high-demand regions; bottom: low-demand regions).

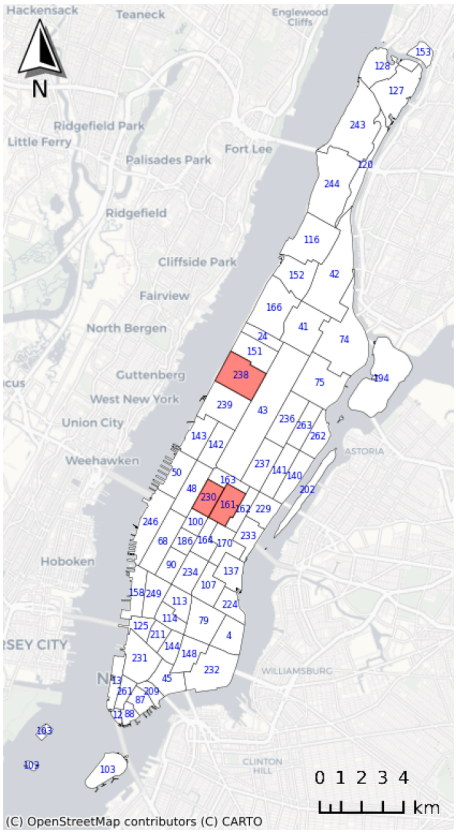


Fig. 8. Three representative regions are selected and marked in red: a workplace area (Region 161), a tourism area (Region 230), and a residential area (Region 238).

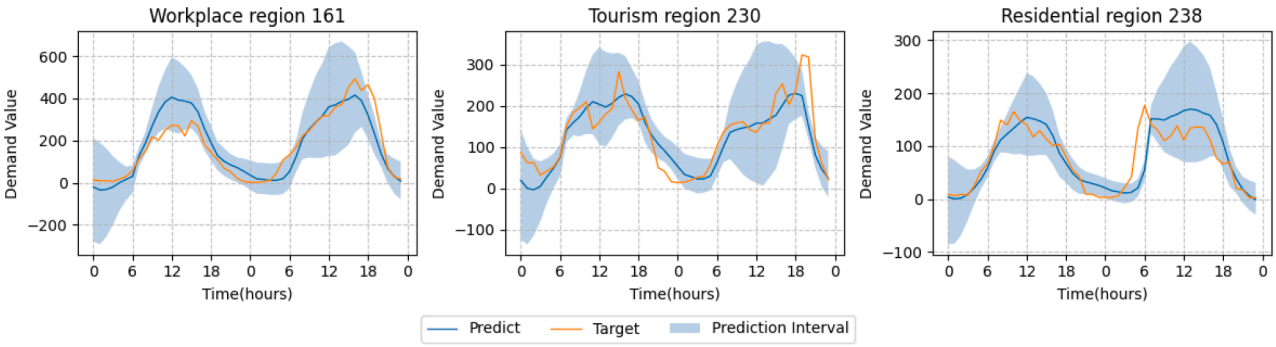


Fig. 9. Uncertainty quantification across three representative regions over two consecutive days (March 23, 2024, Sunday, and March 24, 2024, Monday).

The results are shown in Fig. 9. The figure shows the target demand (orange line), predicted demand (blue line), and associated prediction intervals (shaded blue area) over a 48-hour period, with the x-axis representing time (0–24h for each day) and the y-axis representing demand value. These regions were deliberately chosen for their distinct travel demand profiles, enabling a comprehensive exploration of how demand and its associated uncertainty fluctuate across diverse urban contexts. By examining these patterns over a weekend and a workday, we capture temporal variations that reflect differing activity rhythms, offering a robust testbed for our model's ability to quantify uncertainty effectively.

In the workplace region (161), the target demand on Sunday exhibits a modest peak around 12 PM, reaching approximately 200 trips per hour, before declining steadily into the evening. On Monday, the demand pattern shifts dramatically, with a pronounced peak at 5–6 PM, reaching nearly 600 trips, which aligns with the end of the typical workday and the evening commute. The predicted demand closely tracks the target values, with the blue line generally overlapping the orange line, indicating high predictive accuracy of our model. However, the prediction intervals widen significantly around 12 PM on both days, spanning from roughly 200 to 600 units on Sunday and 100 to 700 units on Monday. This elevated uncertainty during midday likely stems from the heterogeneity of travel purposes, such as lunch breaks, meetings, or sporadic personal trips, which lack the structured regularity of commuting. In contrast, the intervals are narrower during the Monday evening peak (17–18 PM), spanning approximately 200 to 400 units, reflecting the more predictable flow of workers leaving offices.

In the tourism region (230), the target demand displays a more consistent profile across both days, fluctuating between 10 and 300 units during daylight hours (roughly 6 AM to 6 PM) before dropping sharply to near-zero levels after 6 PM. This pattern reflects the continuous influx of visitors to attractions, largely unaffected by the weekday-weekend divide that governs workplace activity. The predicted demand again aligns well with the target, with minor deviations during the early morning hours (0–6 AM). However, the prediction intervals remain consistently wide throughout the day, often spanning from 100 to 300 units, even during peak demand periods around 12 PM. This persistent uncertainty reflects the unpredictable nature of tourist behavior, influenced by a complex interplay of factors: individual itineraries, group dynamics, weather conditions, and event schedules. Unlike the workplace region's commute-driven predictability, tourism lacks a fixed temporal regularity, resulting in a diffuse demand pattern that challenges precise forecasting. The slight narrowing of intervals after 18 PM, where demand approaches zero, aligns with a natural tapering of activity as attractions close or visitors return to accommodations.

The residential region (238) shows comparable overall demand levels on both days but with distinct temporal signatures. On Sunday, the target demand peaks around 9–10 AM, reaching approximately 150 units, likely driven by leisure-oriented trips. On Monday, the peak shifts earlier to 6 AM, with demand spiking to around 200 units, corresponding to the morning commute as residents depart for work. A secondary peak on Monday around 6 PM is also visible, reflecting the evening return commute. The predicted demand closely follows the target, with the blue line nearly overlapping the orange line during these peak periods, highlighting the model's accuracy for structured routines. Uncertainty is notably lower during commuting periods—both the Monday morning peak (6 AM) and evening peak (6 PM)—with prediction intervals spanning roughly 50 to 150 units, indicating a high degree of regularity in these movements. In contrast, the Sunday morning peak (9–10 AM) shows slightly wider intervals (approximately 90 to 200 units), possibly due to the discretionary nature of weekend activities, which vary more widely in timing and purpose than weekday routines. Across both days, midday hours (around 12 PM) exhibit broader intervals, suggesting a mix of secondary trips (e.g., shopping or social visits) that defy the predictability of peak commuting times.

A striking trend across all three regions is the consistent peak in uncertainty around 12 PM. This phenomenon likely arises from the convergence of diverse travel purposes during midday: professional errands and lunch trips in the workplace region, spontaneous sightseeing in the tourism region, and miscellaneous household activities in the residential region. Such multiplicity introduces greater stochasticity into the demand signal, complicating precise estimation. Conversely, uncertainty narrows during commuting hours (e.g., 6–7 AM and 5–6 PM on Monday), where travel is dominated by habitual, work-related trips with well-defined spatial and temporal constraints. For instance, in the workplace region, the interval at 5 PM on Monday spans only 300 to 400 units, a much tighter range than the midday peak. This contrast highlights a key insight: the predictability of demand is closely tied to the degree of underlying behavioral regularity, with structured routines yielding tighter bounds than discretionary or heterogeneous activities.

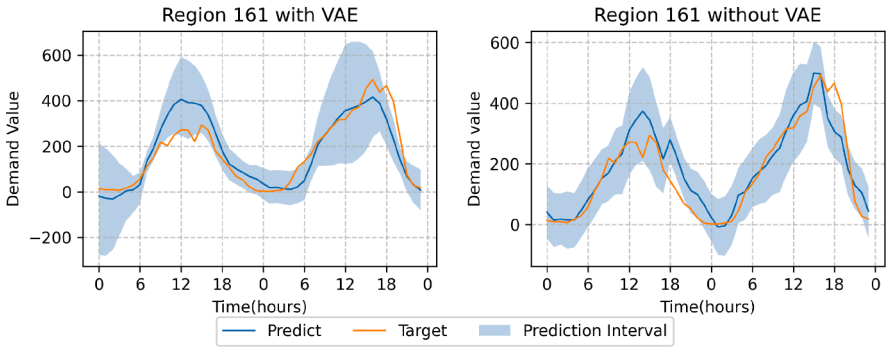


Fig. 10. Uncertainty quantification on region 161 over two consecutive days with and without VAE integration. (March 23, 2024, Sunday, and March 24, 2024, Monday).

These findings highlight the strength of our probabilistic approach, which combines predictive accuracy with explicit uncertainty quantification. The close alignment between predicted and target demand across all regions demonstrates the model's ability to capture underlying patterns, while the prediction intervals provide a transparent measure of confidence in these estimates. By modeling not just point estimates but the full distribution of possible outcomes, our method captures the inherent variability in urban travel demand across different contexts, locations, and times. This added layer of information is especially valuable for decision makers — such as transport planners, ride-sharing operators, or policymakers — who must navigate uncertainty to optimize resource allocation, design responsive services, or mitigate congestion. For example, the wider intervals in the tourism region highlight the need for flexible capacity to accommodate unpredictable visitor flows, while the tighter bounds during commuting hours in the workplace and residential regions support more targeted scheduling of public transport services.

4.3. Ablation study and hyperparameter sensitivity analysis

To further evaluate the robustness and interpretability of our proposed framework, we conduct two complementary analyses: an ablation study to isolate the contribution of the VAE module, and a hyperparameter sensitivity analysis to assess the stability of model performance under varying configurations.

Ablation study To investigate the contribution of the VAE to demand forecasting, we conducted an ablation study on region 161, over a continuous 48-hour period covering Sunday and Monday, as shown in Fig. 10. The VAE enhances interpretability by encoding temporal data into a latent space, capturing day-of-week variations, resulting in wider, more adaptive prediction intervals (left) that consistently encompass the target values. In contrast, the non-VAE variant (right) generates prediction intervals that remain nearly constant in width, failing to adjust to changes in demand volatility. This static behavior suggests that uncertainty is not adequately captured. In the non-VAE model, the uncertainty likely stems from model limitations rather than data variability. By contrast, the VAE's probabilistic modeling enables practitioners to interpret the width of prediction intervals in relation to real-world activity patterns, with narrower intervals during stable periods and wider intervals during volatile periods such as rush hours. This provides a transparent diagnostic tool to trace demand drivers such as commuting or commercial activity, which is crucial for building trust and refining models in real-world applications.

Hyperparameter sensitivity analysis To evaluate the robustness of our VAE-based forecasting framework, we conducted a sensitivity analysis on three key hyperparameters: the latent dimension size (L), the number of samples drawn during inference (S), and the bandwidth (h) used in KDE. These parameters directly influence the model's capacity to encode temporal-spatial patterns, reconstruct meaningful distributions, and quantify predictive uncertainty.

As shown in Table 7, the bandwidth parameter h, when varied under fixed latent dimension ($L = 64$) and sample size ($S = 30$), exhibits a U-shaped effect on performance. A smaller bandwidth ($h = 0.5$) yields sharp and relatively narrow predictive intervals (MPIW = 22.23) and the best RMSE (19.41), but suffers from degraded probabilistic scores (CRPS = 19.12, IS = 25.88), likely due to overfitting and increased noise sensitivity in KDE. Increasing the bandwidth to 2.0 results in smoother but overly diffused distributions, reflected in the worst IS (27.67) and the widest intervals (MPIW = 24.54). The configuration with $h = 1.0$ achieves the best balance across all metrics, providing the lowest CRPS (17.11), MAE (5.41), and IS (19.41), while maintaining reasonably tight intervals (MPIW = 23.88). These results suggest that $h = 1.0$ effectively smooths the sample distribution without losing sharpness or introducing bias. When varying the number of samples (S) under fixed $L = 64$ and $h = 1.0$, we observe that too few samples ($S = 15$) yield narrower intervals (MPIW = 15.23) and a strong IS (21.98), but slightly worse CRPS (18.66) and MAE (5.64), indicating reduced robustness in reconstructing the distribution. Increasing to $S = 60$ leads to higher errors (MAE = 6.15, CRPS = 19.23) without clear gains in uncertainty quantification, suggesting diminishing returns and possible overfitting to noise. The configuration $S = 30$ achieves a desirable trade-off between probabilistic accuracy and computational efficiency. Lastly, varying the latent dimension size L under fixed $S = 30$ and $h = 1.0$ reveals that both under- and over-parameterization degrade model performance. A small latent space ($L = 32$) fails to capture complex demand patterns, resulting in high errors (MAE = 9.63, RMSE = 32.53) and wide intervals (MPIW = 43.74). In contrast, a large latent space ($L = 128$) produces overconfident yet inaccurate predictions: although the intervals

Table 7
Parameter sensitivity (L: Latent: dimension of latent space in VAE; S: Sample: number of samples generated for distribution construction; h: bandwidth of kernel in KDE)). Metrics (↓ better). ⋆ denotes the configuration used in the main experiments.

(L, S, h)	IS ↓	CRPS ↓	MAE ↓	RMSE ↓	MPIW ↓
Varying bandwidth h (fix L=64, S=30)					
(64, 30, 0.5)	25.88	19.12	5.71	**19.41**	**22.23**
(64, 30, 1.0) ⋆	**19.41**	**17.11**	**5.41**	19.93	23.88
(64, 30, 2.0)	27.67	19.98	5.71	**19.41**	24.54
Varying samples S (fix L=64, h=1.0)					
(64, 15, 1.0)	21.98	18.66	5.64	**17.80**	**15.23**
(64, 30, 1.0) ⋆	**19.41**	**17.11**	**5.41**	19.93	23.88
(64, 60, 1.0)	26.92	19.23	6.15	20.98	21.91
Varying latent dimension L (fix S=30, h=1.0)					
(32, 30, 1.0)	50.55	22.20	9.63	32.53	43.74
(64, 30, 1.0) ⋆	**19.41**	**17.11**	**5.41**	**19.93**	23.88
(128, 30, 1.0)	29.45	18.63	8.83	28.75	**10.60**

are narrow (MPIW = 10.60), the MAE (8.83) and RMSE (28.75) are significantly worse, reflecting overfitting and bias. The best overall performance is achieved at $L = 64$, which balances representational power and generalization ability.

Overall, the experiments reveal three main patterns: (i) h shows a U-shape, with h=1.0 balancing calibration (IS/CRPS) against interval width (MPIW); (ii) fewer samples (S=15) sharpen intervals but hurt CRPS, whereas S=30 offers the best trade-off; and (iii) both too small and too large latent sizes degrade accuracy, with L=64 yielding the most favorable results. This analysis highlights the importance of carefully tuning latent representation, sampling depth, and distribution smoothing when applying generative models to travel demand forecasting. Future work could explore adaptive strategies for hyperparameter selection to improve robustness across diverse spatial and temporal contexts.

5. Conclusion

In this study, we proposed STGCN-VAE, a novel probabilistic framework for travel demand forecasting that not only quantifies uncertainty beyond state-of-the-art models but also demonstrates robustness and general applicability across diverse urban contexts and service modes through extensive real-world evaluations. Unlike parametric probability forecasting approaches, STGCN-VAE adopts a nonparametric paradigm to predict the underlying distribution of travel demand. The framework leverages a Spatio-Temporal Graph Convolutional Network (STGCN) to efficiently capture spatial and temporal dependencies, extracting robust features from the data. These features are then compressed into a latent embedding by a Variational Autoencoder (VAE) encoder. Through sampling and reparameterization in the latent space, the decoder reconstructs the demand distribution. During inference, multiple samples are generated, fitted to a Kernel Density Estimator (KDE), and subjected to statistical operations to quantify uncertainty. Extensive experiments on real-world public datasets demonstrate that STGCN-VAE significantly outperforms baseline models, delivering accurate point forecasts and robust distribution forecasts with probability distributions. This enables uncertainty quantification for travel demand, providing a reliable and interpretable forecasting solution.

Despite its strong performance, STGCN-VAE has several limitations. It does not explicitly leverage supply-demand interactions, may miss unfulfilled demand, and lacks destination information. Additionally, like many machine learning models, it underperforms in low-demand regions due to data sparsity. Recognizing these limitations highlights areas for refinement and the need for careful data integration and sparsity-aware modeling

Building on these reflections, future work should focus on further enhancing the framework's flexibility, accuracy, and applicability. First, given the modular nature of STGCN-VAE, future research can explore replacing or augmenting the STGCN component with recent advances in spatiotemporal modeling, such as attention mechanisms or transformer-based architectures, to improve scalability and representational power. Second, integrating heterogeneous data sources — such as weather conditions, special events, and socioeconomic indicators — offers a promising direction for multimodal data fusion that could enhance predictive performance. Third, while our current framework employs MSE on the mean of generated scenarios to ensure stable and scalable training, the evaluation relies on probabilistic metrics such as CRPS, IS. This creates a partial mismatch between training and evaluation objectives. In future work, we plan to explore mixed or distribution-aware training objectives, for example, by incorporating differentiable approximations of probabilistic scoring rules (e.g., CRPS) or hybrid losses that jointly balance point accuracy and distributional calibration. Such extensions could further align the optimization process with evaluation criteria and improve the methodological contribution of non-parametric generative models for travel demand prediction. Fourth, sparsity handling, like many machine learning models, our approach underperforms in low-demand regions due to data sparsity compared to high-demand regions, where statistical models often achieve better robustness. Integrating sparsity-aware mechanisms into our framework would therefore be beneficial and represents an important direction for future work. Lastly, future work may assess the generalizability of the framework by applying it to related forecasting tasks, including energy consumption, bike-sharing demand, and traffic flow prediction, thereby evaluating its robustness across domains.

CRediT authorship contribution statement

Tao Peng: Writing – original draft, Methodology, Conceptualization; **Jie Gao:** Writing – review & editing, Methodology, Supervision, Conceptualization; **Oded Cats:** Writing – review & editing, Supervision, Conceptualization.

Data availability

The datasets I used in this research are publicly available and all sources are included in the paper.

References

Andreoni, A., Postorino, M.N., et al., 2006. A multivariate ARIMA model to forecast air transport demand. Proceedings of the Association for European Transport and Contributors, 1–14.

Audenhove, F.-J.V., Ali, S., Salem, J., 2020. Rethinking on-demand mobility. https://www.adlittle.com/en/insights/report/rethinking-demand-mobility.

Bhatt, U., Antorán, J., Zhang, Y., Liao, Q.V., Sattigeri, P., Fogliato, R., Melançon, G., Krishnan, R., Stanley, J., Tickoo, O., et al., 2021. Uncertainty as a form of transparency: measuring, communicating, and using uncertainty. In: Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society, pp. 401–413.

Box, G. E.P., Jenkins, G.M., Reinsel, G.C., Ljung, G.M., 2015. Time Series Analysis: Forecasting and Control. John Wiley & Sons.

Bruna, J., Zaremba, W., Szlam, A., LeCun, Y., 2013. Spectral networks and locally connected networks on graphs. arXiv:1312.6203.

Chen, E., Ye, Z., Wang, C., Xu, M., 2019. Subway passenger flow prediction for special events using smart card data. IEEE Trans. Intell. Transp. Syst. 21 (3), 1109–1120.

Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y., 2014. On the properties of neural machine translation: Encoder-decoder approaches. arXiv:1409.1259.

Defferrard, M., Bresson, X., Vandergheynst, P., 2016. Convolutional neural networks on graphs with fast localized spectral filtering. Adv. Neural Inf. Process. Syst. 29.

Elman, J.L., 1990. Finding structure in time. Cogn. Sci. 14 (2), 179–211.

Gammelli, D., Peled, I., Rodrigues, F., Pacino, D., Kurtaran, H.A., Pereira, F.C., 2020. Estimating latent demand of shared mobility through censored gaussian processes. Transp. Res. Part C: Emerg. Technol. 120, 102775.

Gao, J., Li, X., Wang, C., Huang, X., 2021. BM-DDPG: an integrated dispatching framework for ride-hailing systems. IEEE Trans. Intell. Transp. Syst. 23 (8), 11666–11676.

Gao, J., Zhu, Y., Cats, O., 2025. Uncertainties in shared mobility optimization problems: survey and perspective. Transp. Rese. Part E: Logist. Transp. Rev. 203, 104350.

Gneiting, T., Raftery, A.E., 2007. Strictly proper scoring rules, prediction, and estimation. J. Am. Stat. Assoc. 102 (477), 359–378.

Gu, B., Zhang, T., Meng, H., Zhang, J., 2021. Short-term forecasting and uncertainty analysis of wind power based on long short-term memory, cloud model and non-parametric kernel density estimation. Renew. Energy 164, 687–708.

Guo, G., Zhang, T., 2020. A residual spatio-temporal architecture for travel demand forecasting. Transp. Res. Part C: Emerg. Technol. 115, 102639.

Guo, S., Lin, Y., Feng, N., Song, C., Wan, H., 2019. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence. 33, pp. 922–929.

He, Y., Li, H., 2018. Probability density forecasting of wind power using quantile regression neural network and kernel density estimation. Energy Convers. Manage. 164, 374–384.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Jiang, Y., Fan, J., Liu, Y., Zhang, X., 2022. Deep graph gaussian processes for short-term traffic flow forecasting from spatiotemporal data. IEEE Trans. Intell. Transp. Syst. 23 (11), 20177–20186.

Ke, J., Qin, X., Yang, H., Zheng, Z., Zhu, Z., Ye, J., 2021. Predicting origin-destination ride-sourcing demand with a spatio-temporal encoder-decoder residual multi-graph convolutional network. Transp. Res. Part C: Emerg. Technol. 122, 102858.

Ke, J., Zheng, H., Yang, H., Chen, X.M., 2017. Short-term forecasting of passenger demand under on-demand ride services: a spatio-temporal deep learning approach. Transp. Res. part C: Emerg. Technol. 85, 591–608.

Khosravi, A., Nahavandi, S., Creighton, D., Atiya, A.F., 2010. Lower upper bound estimation method for construction of neural network-based prediction intervals. IEEE Trans. Neural Netw. 22 (3), 337–346.

Kipf, T.N., Welling, M., 2016. Semi-supervised classification with graph convolutional networks. arXiv:1609.02907.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324.

Li, C., Geng, M., Chen, Y., Cai, Z., Zhu, Z., Chen, X.M., 2024. Demand forecasting and predictability identification of ride-sourcing via bidirectional spatial-temporal transformer neural processes. Transp. Res. Part C: Emerg. Technol. 158, 104427.

Liu, H., Jiang, W., Liu, S., Chen, X., 2023. Uncertainty-aware probabilistic travel time prediction for on-demand ride-hailing at didi. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 4516–4526.

Liu, L., Chen, J., Wu, H., Zhen, J., Li, G., Lin, L., 2020. Physical-virtual collaboration modeling for intra-and inter-station metro ridership prediction. IEEE Trans. Intell. Transp. Syst. 23 (4), 3377–3391.

Lucken, E., Frick, K.T., Shaheen, S.A., 2019. “Three ps in a MOD:” role for mobility on demand (MOD) public-private partnerships in public transit provision. Res. Transp. Bus. Manage. 32, 100433.

Mahajan, G., 2025. Ride-hailing market size and trends. https://www.coherentmarketinsights.com/market-insight/ride-hailing-market-5446.

de Nailly, P., Côme, E., Oukhellou, L., Samé, A., Ferrière, J., Merad-Boudia, Y., 2024. Deep probabilistic forecasting of multivariate count data with sums and shares distributions: a case study on pedestrian counts in a multimodal transport hub. IEEE Trans. Intell. Transp. Syst. 25 (11), 15687–15701.

Pearce, T., Brintrup, A., Zaki, M., Neely, A., 2018. High-quality prediction intervals for deep learning: a distribution-free, ensembled approach. In: International Conference on Machine Learning. PMLR, pp. 4075–4084.

Salinas, D., Flunkert, V., Gasthaus, J., Januschowski, T., 2020. DeepAR: probabilistic forecasting with autoregressive recurrent networks. Int. J. Forecast. 36 (3), 1181–1191.

Stoklosa, J., Blakey, R.V., Hui, F. K.C., 2022. An overview of modern applications of negative binomial modelling in ecology and biodiversity. Diversity 14 (5), 320.

Tian, C., Chan, W.K., 2021. Spatial-temporal attention wavenet: a deep learning framework for traffic prediction considering spatial-temporal dependencies. IET Intel. Transport Syst. 15 (4), 549–561.

Van Engelen, M., Cats, O., Post, H., Aardal, K., 2018. Enhancing flexible transport services with demand-anticipatory insertion heuristics. Transp. Res. Part E: Logist. Transp. Rev. 110, 110–121.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y., 2017. Graph attention networks. arXiv:1710.10903.

Wang, H., Yang, H., 2019. Ridesourcing systems: a framework and review. Transp. Res. Part B: Methodol. 129, 122–155.

Wang, Q., Wang, S., Zhuang, D., Koutsopoulos, H., Zhao, J., 2024. Uncertainty quantification of spatiotemporal travel demand with probabilistic graph neural networks. IEEE Trans. Intell. Transp. Syst. 25 (8), 8770–8781.

Wu, Q., Law, R., Xu, X., 2012. A sparse gaussian process regression model for tourism demand forecasting in hong kong. Expert Syst. Appl. 39 (5), 4769–4774.

Yu, B., Yin, H., Zhu, Z., 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. arXiv:1709.04875.

Yuan, H., Li, G., 2021. A survey of traffic prediction: from spatio-temporal data to intelligent transportation. Data Sci. Eng. 6 (1), 63–85.

Zhu, L., Laptev, N., 2017. Deep and confident prediction for time series at uber. In: 2017 IEEE International Conference on Data Mining Workshops (ICDMW). IEEE, pp. 103–110.

Zhu, Z., Xu, M., Di, Y., Chen, X., Yu, J., 2023a. Modelling ride-sourcing matching and pickup processes based on additive gaussian process models. Transport. B: Transp. Dyn. 11 (1), 590–611.
Zhu, Z., Xu, M., Ke, J., Yang, H., Chen, X.M., 2023b. A bayesian clustering ensemble gaussian process model for network-wide traffic flow clustering and prediction. Transp. Res. Part C: Emerg. Technol. 148, 104032.
Zhuang, D., Wang, S., Koutsopoulos, H., Zhao, J., 2022. Uncertainty quantification of sparse travel demand prediction with spatial-temporal graph neural networks. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 4639–4647.