Information Fusion 126 (2026) 103692

Contents lists available at ScienceDirect

Information Fusion

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

# Spatio-temporal data fusion framework based on large language model for enhanced prediction of electric vehicle charging demand in smart grid management

Yitong Shang [a], Wen-Long Shang [b,c,*], Dingsong Cui [c], Peng Liu [d,e,f], Haibo Chen [c], Dongdong Zhang [g], Runsen Zhang [h], Chengcheng Xu [i], Ye Liu [c], Chenxi Wang [c], Mohannad Alhazmi [j]

[a] Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong, China
[b] Department of Civil and Environmental Engineering, Imperial College London, UK
[c] Institute for Transport Studies, University of Leeds, 34-40 University Road, Leeds LS2 9JT, UK
[d] Collaborative Innovation Centre for Electric Vehicles, Beijing, China
[e] National Engineering Research Centre for Electric Vehicles, Beijing Institute of Technology, Beijing 100081, China
[f] Beijing Institute of Technology Chongqing Innovation Centre, Chongqing 401120, China
[g] Renewable Energy School, Inner Mongolia University of Technology, Ordos City, China
[h] Graduate School of Frontier Sciences, The University of Tokyo, Kashiwa 2778563, Japan
[i] School of Transportation, Southeast University, Nanjing 210096, China
[j] Electrical Engineering Department, College of Applied Engineering, King Saud University, Riyadh, Saudi Arabia

## ARTICLE INFO

*Keywords:*
Electric vehicle
Charging demand prediction
Spatiotemporal data fusion
Large language models
Model fusion
Low-rank adaptation

## ABSTRACT

Accurate prediction of electric vehicle (EV) charging demand is pivotal for effective smart grid management and renewable energy integration. However, predicting spatio-temporal EV charging patterns remains challenging due to complex data fusion requirements arising from heterogeneous temporal, spatial, and contextual features, as well as difficulties in effectively integrating multiple modeling approaches. This paper introduces EV-STLLM, a novel spatio-temporal data fusion framework based on Large Language Model explicitly designed for accurate short-term EV charging demand forecasting through innovative integration of data-level and model-level fusion techniques. At the data level, a multi-source embedding module is developed to seamlessly fuse temporal features (e.g., time slots, weekdays), spatial heterogeneity (e.g., geographical location), and contextual charging behaviors into a unified representation via embedding convolutional network. At the model level, a large language model (LLM) is employed to capture global spatiotemporal dependencies, enhanced with Low-Rank Adaptation (LoRA) for parameter-efficient fine-tuning, substantially reducing computational costs while maintaining prediction robustness. Using a comprehensive real-world dataset comprising over 830,000 EV charging records across 16 districts and 331 subdistricts in Beijing, we validate EV-STLLM across multiple forecasting scenarios (district and subdistrict levels, one-step and two-step ahead predictions). Extensive comparative evaluations demonstrate that EV-STLLM consistently outperforms classical, graph-based, and deep learning baselines. Specifically, in one-step ahead district-level forecasting, EV-STLLM achieves up to a 15.41% reduction in MAE and a 53.51% reduction in MAPE compared to the leading baseline, underscoring its potential to significantly enhance data-driven smart grid operations.

## 1. Introduction

With the accelerated global transition towards carbon neutrality, Electric Vehicles (EVs) are shifting from being an "option" in the transportation revolution to a "necessity" in urban low-carbon transformation [1]. As of 2023, the number of EVs in China has surpassed 20 million, and it is expected to reach 160 million by 2030 [2]. However, the rapid growth of EV ownership has led to unprecedented scheduling pressures on urban power systems [3]. EV charging behavior exhibits significant spatiotemporal characteristics, with high volatility and concentration in different time periods and spatial regions, creating new challenges for grid stability [4]. Particularly during peak electricity

* Corresponding author.
*E-mail address:* wenlong.shang12@imperial.ac.uk (W.-L. Shang).

https://doi.org/10.1016/j.inffus.2025.103692
Received 26 April 2025; Received in revised form 14 August 2025; Accepted 1 September 2025
Available online 9 September 2025

usage periods or in specific areas, concentrated EV charging may cause localized load surges, even leading to power supply bottlenecks [5].

In this context, conducting spatiotemporal predictions of short-term EV charging demand is of great importance for smart grid management and pricing strategies [6]–[7]. By accurately predicting the charging demand distribution in different regions over the next hour or next few hours, power companies can devise time-of-use pricing strategies, guiding users to avoid peak load periods and improving the operational efficiency of the power system [8]. Additionally, the prediction results can support grid scheduling, enabling dynamic load adjustment and optimized energy allocation, reducing the peak–valley difference, and enhancing grid security and flexibility [9]. Moreover, short-term predictions can improve user experience, such as by intelligently recommending charging times or locations, reducing queuing time, and improving charging convenience and cost-effectiveness [10]. Therefore, short-term spatiotemporal prediction of EV charging demand is not only an effective means of addressing grid operational pressure, but also an indispensable technological support in the development of green transportation and smart energy [11]. However, despite extensive research on modeling and predicting EV charging demand, achieving high-accuracy short-term spatiotemporal predictions at the urban scale still faces several challenges, especially in data fusion and model fusion [12]. Specifically, data-level fusion focuses on preprocessing and integrating raw or intermediate data from different sources to create a unified representation for prediction models, while model-level fusion emphasizes the combination of outputs or intermediate representations from multiple models to improve overall accuracy and robustness. The details of data-level fusion and model-level fusion are described as follows.

On the one hand, although existing research has extracted a large number of features from temporal, spatial, and user behavior dimensions, efficiently integrating these multi-dimensional and multi-modal features remains one of the core challenges in short-term EV charging demand forecasting. Firstly, charging demand exhibits evident temporal periodicity (e.g., daily and weekly cycles) and temporal burstiness (e.g., during commuting hours). Traditional time series modeling approaches often struggle to capture such nonlinear trends and sudden fluctuations. For example, one study reported that traditional methods (e.g., ARIMA) underperformed their proposed method (TSAGE) by a factor of 3.25 in terms of RMSE [8]. Additionally, historical data often include anomalous time points such as holidays, extreme weather, or unexpected events, further complicating the modeling and fusion of temporal features [13]. Secondly, the spatial distribution of charging stations in urban environments is highly uneven. Charging behaviors across different regions are influenced by various factors such as geographic location, traffic conditions, and surrounding facilities. Effectively modeling spatial correlations (e.g., influence propagation between neighboring regions) and spatial heterogeneity (e.g., behavioral differences between urban centers and suburban areas) is a key challenge for spatial feature fusion [14]. Finally, fundamental features such as user type, charging station category, and pricing mechanisms represent static information whose influence varies under different spatiotemporal contexts. These static features interact in complex ways with dynamic spatiotemporal features. Therefore, dynamically adjusting the weights of static information in the modeling process is an urgent issue in current data fusion research. Consequently, constructing a data fusion framework that can flexibly adapt to multi-source heterogeneous data and dynamically adjust feature weights is crucial for improving prediction accuracy.

On the other hand, in spatiotemporal forecasting tasks for short-term EV charging demand, model fusion has emerged as a key strategy for enhancing prediction accuracy and system robustness. The current mainstream prediction models include small models, large models, and, more recently, large language models (LLMs). Each model category offers advantages in feature modeling, representation capacity, and computational efficiency, but also faces specific challenges in practical application. Small models, such as linear regression (LR), decision trees, and support vector machines, offer high training efficiency, low computational cost, simple structure, and ease of deployment. However, they are limited in capturing complex nonlinear relationships or long-term dependencies, making them less suitable for large-scale, multi-dimensional urban-level charging demand forecasting [15]. Large models, such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and Transformers, possess significant advantages in spatiotemporal modeling, capable of uncovering deeper sequential patterns and spatial dependencies in charging behaviors. Nevertheless, these models require substantial data and computational resources for training, are costly to implement, and are prone to overfitting [16]. Recently, LLMs, such as GPT and BERT, have achieved remarkable progress in natural language processing, especially in understanding complex semantics and generating context-aware content. However, these models were not originally designed for spatiotemporal sequence prediction tasks. Directly applying LLMs to structured time series modeling may encounter structural mismatches and task adaptation challenges [17]. In summary, although integrating small models, large models, and LLMs could leverage their complementary strengths, in practice, model fusion faces several difficulties, including inconsistencies in model architectures, heterogeneity in feature input formats, and complexity in designing effective fusion strategies.

To address the aforementioned challenges, this paper proposes a spatiotemporal fusion framework that integrates both data-level and model-level strategies, aiming to enhance the accuracy and robustness of short-term electric vehicle charging demand forecasting at the urban scale. Leveraging real-world EV charging data from Beijing—covering 331 sub-districts across 16 administrative districts—this study demonstrates how multi-source, city-scale data fusion can effectively support intelligent demand forecasting and urban energy scheduling, providing critical insights for the development of smart grids and low-carbon cities. The main contributions of this paper include:

(1) Unified spatiotemporal data fusion through collaborative embeddings. To address the heterogeneity of multi-source data in EV charging demand forecasting, we design a collaborative embedding mechanism that enables effective data-level fusion across temporal, spatial, and behavioral dimensions. Specifically, temporal patterns are encoded via time-slot and weekday embeddings; spatial heterogeneity is captured through nonlinear transformations of urban region features; and localized charging behaviors are represented using pointwise convolution. These components are integrated via an Embedding Convolutional Network (ECN), constructing a unified spatiotemporal representation capable of capturing both static and dynamic feature interactions across time and space.

(2) Model-level fusion via LoRA-enhanced large language model. To fully leverage the complementary strengths of traditional feature embeddings and large-scale pretrained models, we propose a model fusion strategy that integrates lightweight spatiotemporal encodings with a parameter-efficient fine-tuned LLM. By adopting Low-Rank Adaptation (LoRA), we insert trainable low-rank matrices into the Transformer's Query and Value projections while keeping the pretrained backbone frozen. This approach significantly reduces training overhead, enhances scalability, and enables the LLM to adapt to structured forecasting tasks without compromising its generalization ability.

(3) Extensive real-world validation with superior quantitative performance across scales. We construct a large-scale EV charging dataset spanning 16 districts and 331 subdistricts in Beijing, and evaluate the proposed model across multiple forecasting scenarios. Compared to strong baselines (e.g., GRU, GraphSAGE), EV-STLLM achieves up to 15.45% reduction in MAE and 53.51% reduction in MAPE for district-level 1-step prediction, and up to 19.61% and 26.39% reductions in MAE and RMSE respectively

at the subdistrict level. These results demonstrate the model's superior accuracy, robustness, and fine-grained adaptability across both coarse and fine spatial resolutions.

The rest of this paper is organized as follows. Section 2 reviews the related literature on EV charging demand prediction, spatiotemporal modeling, and the application of LLMs in forecasting. Section 3 formally defines the problem of short-term EV charging demand prediction. Section 4 details the proposed methodology, including the overall framework, the spatiotemporal feature embedding process, the Transformer-based architecture, and the parameter-efficient fine-tuning strategy. Section 5 describes the experimental setup, presents the results of comparative and ablation studies, and analyzes the model's sensitivity. Finally, Section 6 concludes the paper, summarizing the key findings and discussing the practical implications and limitations of the work.

## 2. Related work

### 2.1. EV charging demand prediction

EV charging demand prediction has become an important research direction in the fields of intelligent transportation and smart energy [18]. Existing EV charging demand prediction methods can be broadly divided into three categories: statistical models, traditional machine learning methods, and deep learning methods. Statistical models, such as autoregressive integrated moving average (ARIMA) [19] and seasonal ARIMA (SARIMA) [20] are characterized by simplicity in modeling and strong interpretability, making them suitable for relatively stable time series prediction tasks. However, their ability to model nonlinear relationships is limited, and they perform poorly when dealing with complex and variable charging behaviors. Machine learning methods, such as support vector regression (SVR), random forest, and gradient boosting decision trees (GBDT) are capable of capturing nonlinear features to some extent. These methods are suitable for medium- to small-scale datasets, but they heavily depend on feature engineering [21]. In recent years, deep learning models based on RNN [22], CNN [23], and their variants have been widely applied in charging demand prediction. These models possess strong feature extraction and pattern recognition capabilities, especially in handling high-dimensional complex data and capturing temporal and spatial dependencies.

The variation in EV charging demand is driven by multiple factors, and constructing a high-quality prediction model requires comprehensive consideration and modeling of key influencing factors [24]. Temporal features, such as hours, days of the week, holidays, etc., reflecting the periodicity and regularity of charging behavior. Spatial features, including districts and subdistricts, which reflect regional differences in population, traffic, and infrastructure, thereby influencing charging behavior. Central urban areas tend to have higher charging demands due to dense commuting, while suburban areas are significantly influenced by residential distribution and charging station coverage. Proper modeling of spatial hierarchy helps improve spatial accuracy and generalization ability in predictions. Environmental factors, such as weather (temperature, precipitation, etc.) and air quality, which may indirectly affect vehicle travel frequency and charging demand. Therefore, the fusion of multi-source heterogeneous data has become an important way to enhance model performance, and establishing effective correlations between different data types remains a critical research topic.

### 2.2. Spatiotemporal prediction

Charging demand prediction falls under the broader category of spatiotemporal prediction. Therefore, when constructing prediction models, time and space features should not be treated separately but should be understood in terms of their coupling relationship. During the development of spatiotemporal modeling methods, researchers have proposed various architectures to enhance the ability to capture complex spatiotemporal features. Early methods often employed combinations of CNNs and RNNs, where the former was used to capture spatial dependencies and the latter was used to model temporal dynamics. Models such as CNN-long short-term memory networks (LSTM) [25] and ConvLSTM [26] have improved the accuracy of spatiotemporal sequence modeling to some extent but have limitations when dealing with non-Euclidean spatial structures, such as urban road networks.

To address these issues, graph neural networks (GNNs) have been introduced into spatiotemporal modeling in recent years. Representative models such as spatiotemporal graph convolutional networks (STGCN) [27] and diffusion convolutional recurrent networks (DCRNN) [28] build spatial graph structures to capture non-Euclidean relationships between nodes and integrate time series modeling techniques to efficiently predict spatiotemporal data like traffic flow and energy consumption. Xing et al. proposed a spatiotemporal fusion network that integrates GCN with LSTM-Attention, specifically designed for urban rail transit OD flow prediction [29]. Xu et al. introduced a novel transportation prediction model, the HSTGODE, which employs a dual-layer structure combined with spatiotemporal ordinary differential equation modules to address the over-smoothing problem in GNNs and effectively capture hierarchical spatiotemporal features at both regional and node levels [30]. These methods have achieved significant results in urban traffic, power load, and other fields, further expanding the technical boundaries of spatiotemporal prediction.

With breakthroughs in the Transformer model in natural language processing (NLP) for sequence modeling, researchers have begun to introduce these models into spatiotemporal prediction, developing spatiotemporal Transformer architectures [31]. These models utilize self-attention mechanisms, dynamically capturing global spatiotemporal dependencies, and are better at modeling long-range features. For example, models like spatio-temporal attention network (STAN) have shown superior performance in multiple transportation and energy prediction tasks compared to traditional CNN–RNN structures [32]. Yu et al. proposed the MGSFformer, which utilizes a residual redundancy elimination module to remove information redundancy across different granularities of data. Furthermore, it incorporates spatiotemporal attention and dynamic fusion modules to achieve efficient air quality prediction [33]. Zhang et al. introduced the EF-former, a deep learning-based multistep passenger flow prediction model. By integrating the parallel interactive attention module and multi-scale causal multi-Head self-attention module, EF-former extracts both global and local temporal dependencies, enabling precise modeling of spatiotemporal dynamics during large-scale events and realizing accurate multistep forecasting of passenger flow [34].

### 2.3. Prediction-oriented applications of large language models

In recent years, LLMs such as the GPT series and Llama series have achieved remarkable success in the field of NLP. As the capabilities of LLMs continue to expand, researchers have begun exploring their potential applications in structured data modeling, especially in spatiotemporal prediction tasks. These applications can be summarized into the following three pathways [35].

The first is knowledge extraction and feature enhancement [36]. This approach uses LLMs to perform deep semantic understanding and implicit knowledge extraction from multimodal data (such as text, images, etc.). The extracted semantic features are then input

into traditional prediction models (such as LSTM, GNN, etc.) to enhance the model's ability to perceive background information, behavioral patterns, and environmental factors. In this process, LLMs act as knowledge encoders, effectively supplementing and enhancing feature semantics without changing the original model structure.

The second is text-to-LLM approach [37]. In this method, raw spatiotemporal structured data (such as time series, spatial locations, weather, etc.) is transformed into natural language descriptions, which are then tokenized and input into frozen or fine-tuned LLMs for prediction. This approach has good generality and interpretability, as it can flexibly adapt to any pre-trained language model and improve the human readability of model outputs. However, due to limitations in expressing complex spatiotemporal dependencies, this method faces issues such as high token usage costs, limited context windows, and insufficient expression accuracy, making it unsuitable for high-dimensional and multi-scale complex prediction tasks. Kang et al. proposed an enhanced multi-level health event prediction framework, LLM-DG, which improves prediction accuracy and robustness by semantically enhancing the representation of patients and discharge summaries, injecting domain knowledge, capturing higher-order correlations, and integrating dynamic and static features to simultaneously model inter-patient clustering and intra-patient disease evolution characteristics [38]. Shen et al. introduced a framework that employs reinforcement learning to provide decisive subgraph information for Graph LLMs. By utilizing a reinforcement subgraph detection module and a node-guidance network, the framework searches for and delivers critical neighborhood and node information in textual form to assist LLM predictions, without requiring model retraining [39].

The third is fine-tuning LLMs for spatiotemporal tasks [40]. This method encodes spatiotemporal data as structured token sequences (such as timestamps, region codes, numerical attributes, etc.) and fine-tunes LLMs under supervision, allowing them to understand the spatiotemporal meaning behind these tokens and perform prediction tasks. Compared to natural language descriptions, this approach has higher token efficiency and expression accuracy, better modeling higher-order temporal dependencies and spatial relationships, and stronger adaptability. However, the fine-tuning process typically requires high computational resources and data quality.

### 2.4. Research gap

Although significant progress has been made in EV charging demand prediction, spatiotemporal modeling, and the application of LLMs, several interconnected challenges remain. In EV charging demand prediction, existing methods struggle to effectively model the nonlinear and high-dimensional interactions between temporal, spatial, and behavioral features, particularly in short-term urban-scale tasks. While deep learning models have improved predictive performance, they often fail to adequately fuse multi-source heterogeneous data, such as spatial hierarchies, temporal periodicity, and environmental factors, leading to suboptimal generalization and scalability. In the broader domain of spatiotemporal prediction, traditional architectures like CNN–RNN hybrids and ConvLSTM face difficulties in explicitly capturing complex feature interactions and balancing the trade-off between global generalization and local detail fitting. Advanced models, such as GNNs and spatiotemporal Transformers, have addressed some of these issues but still struggle with multi-scale and highly dynamic scenarios, particularly in non-Euclidean spatial structures like urban road networks. Meanwhile, the emerging application of LLMs in prediction tasks highlights their potential for semantic knowledge extraction and feature enhancement, yet challenges such as efficient tokenization, representation of spatiotemporal dependencies, and the computational cost of fine-tuning remain significant barriers. These gaps collectively indicate the need for a unified framework that integrates the strengths of spatiotemporal architectures with the semantic understanding capabilities of LLMs. The focus should be on achieving efficient data fusion across multi-source heterogeneous datasets and model fusion, providing high-performance prediction systems for intelligent energy and urban-scale applications.

## 3. Problem definition

This work focuses on the problem of short-term EV charging demand prediction, which aims to estimate the future charging demand across multiple spatial locations in a city over discrete time intervals. This is a typical spatiotemporal forecasting task, where both temporal dynamics and spatial heterogeneity must be jointly modeled.

Formally, let the city be partitioned into $N$ spatial units (district or subdistrict), and time be divided into discrete intervals (30 min). For each location and time slot, we observe a set of features describing charging behavior and contextual factors. The objective is to predict the EV charging demand for each location in the next few time steps based on historical data. Let $\mathcal{X} \in \mathbb{R}^{T_{in} \times N_{nodes} \times C_{in}}$ denote the historical multivariate input tensor, where: $T_{in}$ is the number of historical time steps, $N_{nodes}$ is the number of spatial locations, $C_{in}$ is the number of features per location per time step (e.g., past charging demand, time slot index, week of day, etc.).

Let $\hat{Y} \in \mathbb{R}^{T_{out} \times N_{nodes}}$ be the predicted charging demand over a forecasting horizon of $T_{out}$ time steps for $N_{nodes}$ locations. The goal is to learn a mapping function $f(\cdot)$ such that:

$$\hat{Y}_{T+1:T+H} = f(\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_T) \tag{1}$$

Here, $f$ is a predictive model capable of learning complex spatiotemporal dependencies from the input data. This prediction function should capture: Temporal dependencies, such as daily/weekly charging patterns, peak vs. off-peak hours, and holiday effects. Spatial dependencies, such as mobility patterns across districts, charging infrastructure density, and population distribution. External factors, such as weather conditions, public events, and road traffic status, which may impact trip frequency and charging behavior.

## 4. Methodology

### 4.1. Framework

The proposed framework for EV charging demand prediction utilizes a Spatiotemporal Large Language Model (EV-STLLM) as shown in Fig. 1. This framework combines spatiotemporal embeddings with a transformer-based architecture to model complex dependencies across time and space, aiming to enhance the accuracy and robustness of short-term charging demand forecasts in urban environments. Traditional spatio-temporal GCN-based prediction methods typically rely on fixed graph structures or predefined attention patterns, which limit their ability to adapt to dynamic spatial relationships. Our proposed framework is graph-free and leverages position-aware embeddings and Transformer attention to model global spatiotemporal dependencies without assuming any rigid structure. Furthermore, by integrating LoRA for efficient fine-tuning, EV-STLLM achieves a better balance between adaptability and computational efficiency, which is especially important for real-world urban computing scenarios

The historical input data, $\mathcal{X} \in \mathbb{R}^{T_{in} \times N_{nodes} \times C_{in}}$, is converted into tokens representing spatial and temporal features. Three types of embeddings—Auxiliary, Temporal, and Spatial—are applied and fused into a unified spatiotemporal representation. The spatiotemporal embeddings are reshaped and passed into a transformer architecture with a masked multi-head self-attention mechanism. This self-attention enables the model to capture both global and local dependencies in time and space, improving the model's ability to learn complex patterns. To adapt the pretrained language model to the EV charging demand task while minimizing overfitting, the LoRA technique is used. LoRA allows the model to adjust only a small set of parameters, effectively transferring knowledge with fewer computational resources. After processing through the transformer layers, the model produces the final predictions using a regression convolution layer. The loss function combines the prediction error with a regularization term to prevent overfitting.

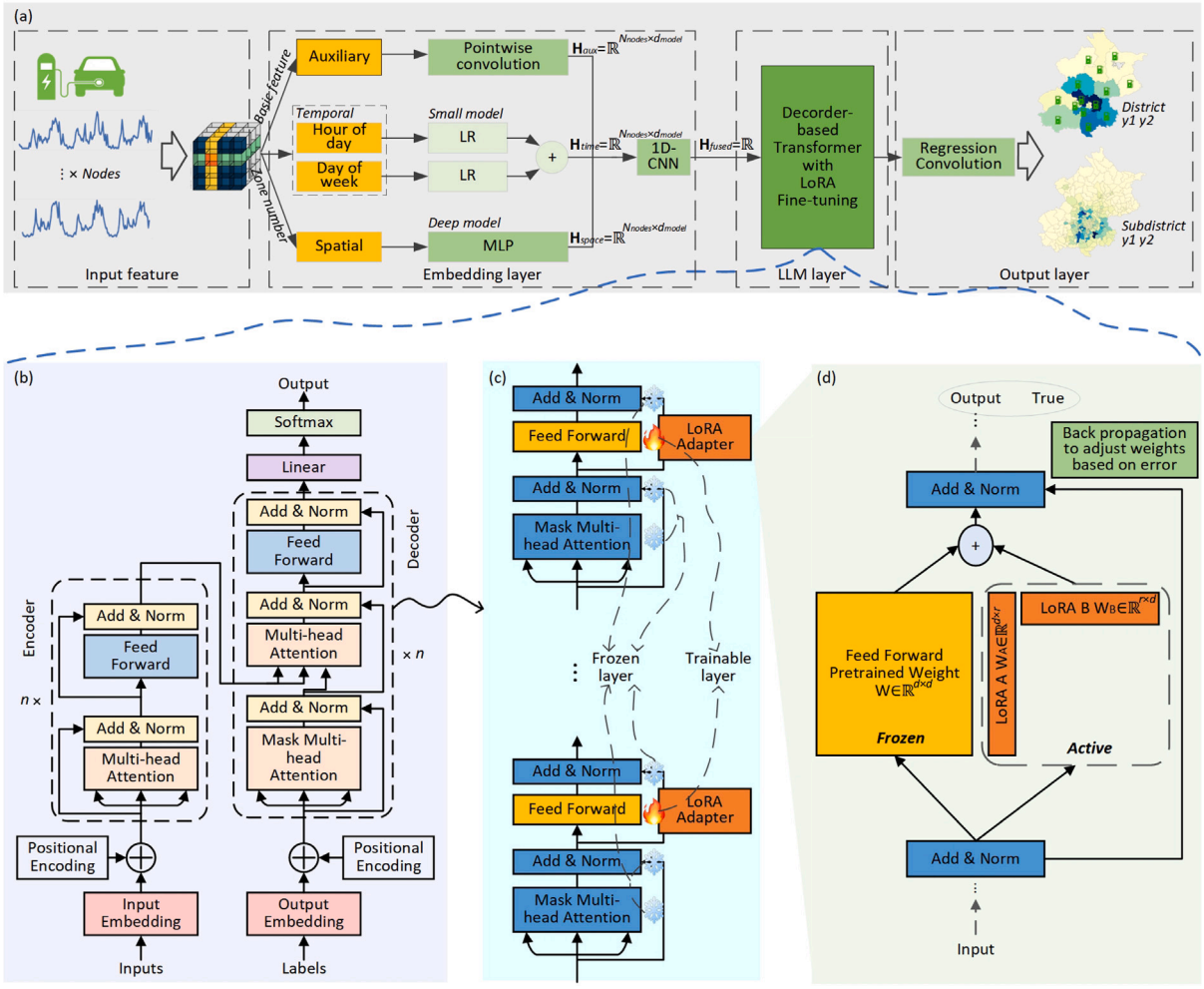


**Fig. 1.** Framework of this work, (a) EV-STLLM, (b) Transformer network,(c) decoder-based Transformer, and (d) LoRA mechanism.

### 4.2. Spatiotemporal feature embedding

As illustrated in Fig. 1(a), we consider the input data $\mathcal{X} \in \mathbb{R}^{T_{in} \times N_{nodes} \times C_{in}}$ as a sequence composed of spatial and temporal information. To comprehensively capture its underlying patterns, we devise three distinct embedding mechanisms.

*Auxiliary feature embedding.* We first process the raw input through a pointwise convolutional layer to generate an auxiliary feature embedding:

$$\mathbf{H}_{aux} = \text{PointwiseConv}(\mathcal{X}; \Theta_{aux}), \quad \mathbf{H}_{aux} \in \mathbb{R}^{N_{nodes} \times d_{model}} \tag{2}$$

where PointwiseConv$(\cdot)$ denotes a $1 \times 1$ convolution operation with trainable parameters $\Theta_{aux}$. $\mathbf{H}_{aux}$ is the resulting node embedding matrix, and $d_{model}$ defines the dimension of the embedding vectors. We employ pointwise convolution due to its exceptional computational efficiency. It effectively captures local inter-feature relationships and compresses the high-dimensional input into a more compact embedding space, providing a refined feature representation for the subsequent spatiotemporal fusion.

*Temporal information embedding (hour and day-of-week).* To encode temporal periodicities, we create learnable embedding vectors for different time slots of the day and days of the week:

$$\mathbf{H}_{time} = \mathbf{W}_{hour}(\mathcal{X}_{hour}) + \mathbf{W}_{day}(\mathcal{X}_{day}), \quad \mathbf{H}_{time} \in \mathbb{R}^{N_{nodes} \times d_{model}} \tag{3}$$

Here, $\mathcal{X}_{hour}$ and $\mathcal{X}_{day}$ are the hour-of-day and day-of-week indices for each node, respectively. $\mathbf{W}_{hour} \in \mathbb{R}^{T_{hour} \times d_{model}}$ and $\mathbf{W}_{day} \in \mathbb{R}^{T_{day} \times d_{model}}$ are two learnable embedding lookup tables. $\mathbf{H}_{time}$ is the final temporal embedding matrix.

*Spatial information embedding.* We extract spatial information directly from the input features using a fully connected layer:

$$\mathbf{H}_{space} = \phi(\mathcal{X} \cdot \mathbf{W}_{space} + \mathbf{b}_{space}), \quad \mathbf{H}_{space} \in \mathbb{R}^{N_{nodes} \times d_{model}} \tag{4}$$

where $\mathbf{W}_{space} \in \mathbb{R}^{C_{in} \times d_{model}}$ and $\mathbf{b}_{space} \in \mathbb{R}^{d_{model}}$ are the weight matrix and bias vector of the layer, respectively. $\phi(\cdot)$ represents a non-linear activation function (e.g., ReLU), and $\mathbf{H}_{space}$ is the spatial embedding matrix derived from the raw features.

Finally, we integrate the three aforementioned embeddings to form a unified, multi-faceted feature representation:

$$\mathbf{H}_{fused} = \text{FusionLayer}(\mathbf{H}_{aux} || \mathbf{H}_{time} || \mathbf{H}_{space}; \Theta_{fuse}), \quad \mathbf{H}_{fused} \in \mathbb{R}^{N_{nodes} \times 3d_{model}} \tag{5}$$

In this equation, $||$ denotes the concatenation operation along the feature dimension. FusionLayer$(\cdot)$ is an ECN function used for feature fusion, with $\Theta_{fuse}$ as its learnable parameters. $\mathbf{H}_{fused}$ is the final fused spatiotemporal embedding representation. We opt for ECN as the spatiotemporal feature fusion method because it efficiently models local dependencies, has low computational complexity, and features flexible learnable parameters. In our experiments, it demonstrated superior performance compared to both more complex and simpler fusion alternatives.

### 4.3. Transformer-based decoder-only architecture

As depicted in Fig. 1(b) and (c), we adopt a Transformer-based decoder-only architecture to model the complex spatiotemporal dependencies in EV charging demand. The model takes the fused spatiotemporal embeddings $\mathbf{H}_{fused} \in \mathbb{R}^{N_{nodes} \times 3d_{model}}$ and reshapes them into a sequence for processing:

$$\mathbf{Z}^{(0)} = \text{Reshape}(\mathbf{H}_{fused}) \in \mathbb{R}^{S \times d}, \quad \text{where } S = N_{nodes},\ d = 3d_{model} \tag{6}$$

Each decoder block consists of a masked multi-head self-attention layer and a feed-forward network (FFN). During spatiotemporal modeling, we leverage the self-attention mechanism of the Transformer to capture global spatiotemporal dependencies by computing attention weights across different time steps and spatial positions in the input sequence. Specifically, for the $l$th layer, the attention mechanism computes the following result:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}} + \mathbf{M}\right)\mathbf{V} \tag{7}$$

$$\mathbf{Z}^{(l)}_{attn} = \text{Concat}(\text{head}_1, \ldots, \text{head}_H)\mathbf{W}_O + \mathbf{b}_O \tag{8}$$

$$\mathbf{Z}^{(l)}_{res1} = \text{LayerNorm}(\mathbf{Z}^{(l)}_{attn} + \mathbf{Z}^{(l-1)}) \tag{9}$$

$$\mathbf{Z}^{(l)}_{ffn} = \phi(\mathbf{Z}^{(l)}_{res1}\mathbf{W}^{(l)}_1 + \mathbf{b}^{(l)}_1)\mathbf{W}^{(l)}_2 + \mathbf{b}^{(l)}_2 \tag{10}$$

$$\mathbf{Z}^{(l)} = \text{LayerNorm}(\mathbf{Z}^{(l)}_{ffn} + \mathbf{Z}^{(l)}_{res1}) \tag{11}$$

Through this mechanism, the model is able to simultaneously capture long-range dependencies across multiple time steps and spatial regions. Moreover, to enable the LLM to handle heterogeneous information, we serialize temporal context, spatial identifiers, and auxiliary behavioral features into token sequences as inputs to the model, thereby implicitly integrating information across scales. After $L$ layers, the final output is:

$$\mathbf{H}^{(L)} = \mathbf{Z}^{(L)} \in \mathbb{R}^{S \times d_{seq}} \tag{12}$$

We then apply a regression layer to generate the final predictions:

$$\hat{\mathbf{Y}}_{T+1:T+H} = \text{RConv}(\mathbf{H}^{(L)}; \Theta_r) \tag{13}$$

### 4.4. Parameter-efficient LoRA fine-tuning

To adapt the pretrained model to the charging demand prediction task while retaining its prior knowledge, we employ Low-Rank Adaptation (LoRA) for efficient fine-tuning, as shown in Fig. 1(d). LoRA injects low-rank matrices into specific weight matrices (e.g., the **Q** and **V** projections in multi-head attention) and exclusively trains these newly introduced matrices while keeping the pretrained weights frozen. Specifically, for a given weight matrix $\mathbf{W}_0 \in \mathbb{R}^{d \times d}$, LoRA modifies it as:

$$\mathbf{W}_{\text{LoRA}} = \mathbf{W}_0 + \alpha \cdot \mathbf{A}\mathbf{B} \tag{14}$$

where $\mathbf{A} \in \mathbb{R}^{d \times r}$ and $\mathbf{B} \in \mathbb{R}^{r \times d}$ are low-rank matrices with $r \ll d$, and $\alpha$ is a scaling factor. The original weight matrix $\mathbf{W}_0$ remains frozen, and only **A** and **B** are trainable. The key advantages of this design are as follows:

- Computational Efficiency: By significantly reducing the number of trainable parameters (less than 1% of the full model), LoRA drastically decreases GPU memory usage and training time.
- Robustness: LoRA preserves the original knowledge and capacity of the pretrained model, mitigating the risk of overfitting in low-data scenarios.
- Stability: By only fine-tuning the Query and Value matrices while keeping other components (e.g., $K$ and FFN) frozen, LoRA further reduces the risk of catastrophic forgetting.

In our model, we apply LoRA fine-tuning to the Query and Value projection matrices of each Transformer block:

$$\mathbf{Q}_{\text{LoRA}} = \mathbf{Q}_0 + \alpha_q \cdot \mathbf{A}_q\mathbf{B}_q, \quad \mathbf{V}_{\text{LoRA}} = \mathbf{V}_0 + \alpha_v \cdot \mathbf{A}_v\mathbf{B}_v \tag{15}$$

The remaining matrices, such as for the Key (**K**), Output (**O**), and the FFN, are kept frozen to minimize the number of trainable parameters and prevent catastrophic forgetting. Through this approach, we achieve efficient customization for domain-specific tasks without sacrificing model performance.

### 4.5. Loss function

After obtaining the output representations $\mathbf{H}^{(L)} \in \mathbb{R}^{N_{nodes} \times d_{seq}}$ from the Transformer backbone, a regression convolution layer is applied to produce the final prediction:

$$\hat{\mathbf{Y}}_{T+1} = \text{RConv}(\mathbf{H}^{(L)}; \Theta_r) \tag{16}$$

The loss function is defined as the sum of the prediction error and a regularization term on the LoRA parameters:

$$\mathcal{L} = \left\| \hat{\mathbf{Y}}_{T+1} - \mathbf{Y}_{T+1} \right\|_2^2 + \lambda \cdot \|\Theta_{\text{LoRA}}\|_2^2 \tag{17}$$

where $\Theta_{\text{LoRA}}$ denotes the set of all trainable LoRA parameters and $\lambda$ is the regularization coefficient.

## 5. Experiments

### 5.1. Experiment setup

#### 5.1.1. Dataset description

In this section, we demonstrate the effectiveness of the proposed EV-STLLM framework through a case study based on Beijing, China. As one of the leading cities in China for EV adoption, Beijing has developed a robust electric transportation network. We utilized EV charging log data from December 2020, collected at a 30 min interval, covering 16 districts and 331 subdistrict regions across the city. The dataset includes 833,439 charging events, each with an average of 22.06 kWh of energy consumed.

The spatial distribution of charging demand across various districts and subdistrict regions is depicted in Fig. 2. Specifically, Fig. 2(a) shows the total EV charging demand aggregated by district. Darker colors indicate districts with higher total energy consumption, highlighting that central districts such as Chaoyang and Haidian exhibit significantly higher demand. Fig. 2(b) provides a finer granularity by illustrating the spatial distribution of charging demand at the subdistrict level. Brighter colors represent higher demand densities, revealing hot spots of EV activity within the city.

Table 1 summarizes the maximum instantaneous charging demand and the cumulative total charging demand for each district during the observation period. In addition, Fig. 3 displays the temporal dynamics of EV charging demand throughout December 2020. Specifically, Fig. 3(a) illustrates the charging demand over time at each district, segmented into training, validation, and testing periods, with a ratio of 8:1:1. The split is temporally consistent, meaning earlier days are allocated to the training set, and more recent days are assigned to the testing set. This strategy prevents data leakage and ensures the model is trained on historical data and evaluated on future data, mirroring real-world forecasting scenarios. The training set is shown in blue, the validation set in orange, and the testing set in green. Distinct daily patterns and weekly seasonality can be observed. For feature standardization, we employed the z-score normalization method. This

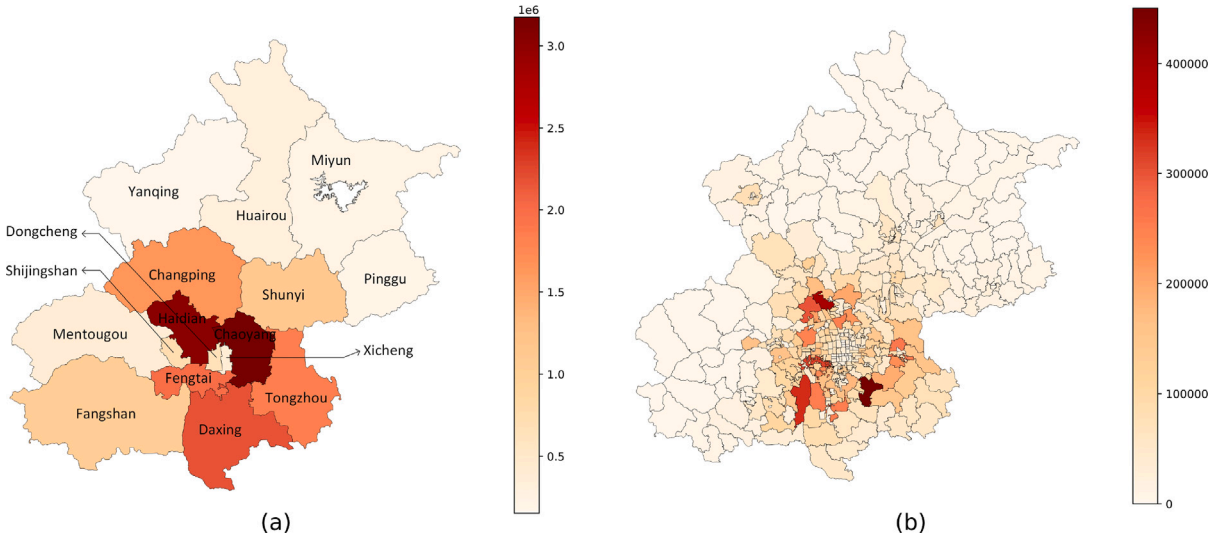


**Fig. 2.** Distribution of the EV charging demand in (a) different districts and (b) different subdistrict.

**Table 1**
Maximum and total EV charging demand by Beijing districts.

| Index | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Name | Dongcheng | Xicheng | Chaoyang | Haidian | Shijingshan | Fengtai | Mentougou | Fangshan |
| Mean (kWh) | 245.16 | 434.66 | 2123.46 | 2021.02 | 490.84 | 1329.92 | 260.63 | 693.42 |
| Std | 219.65 | 379.34 | 1766.87 | 1695.40 | 418.29 | 1129.79 | 230.64 | 597.07 |
| Max (kWh) | 1776.53 | 2875.16 | 13 993.56 | 14 218.98 | 3367.72 | 10 184.80 | 1991.60 | 4878.33 |
| Sum (kWh) | 364 546.60 | 646 338.50 | 3157578.00 | 3005260.00 | 729 876.70 | 1977598.00 | 387 558.10 | 1031116.00 |
| Index | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| Name | Changping | Daxing | Shunyi | Tongzhou | Miyun | Pinggu | Huairou | Yanqing |
| Mean (kWh) | 1094.24 | 1451.24 | 748.59 | 1232.15 | 143.99 | 123.60 | 212.20 | 103.14 |
| Std | 918.04 | 1211.85 | 641.65 | 1030.00 | 137.96 | 119.03 | 191.22 | 105.14 |
| Max (kWh) | 8057.20 | 9853.18 | 6074.08 | 8354.02 | 1539.35 | 966.94 | 1786.56 | 878.67 |
| Sum (kWh) | 1627137.00 | 2158001.00 | 1113159.00 | 1832207.00 | 214 116.00 | 183 791.30 | 315 541.20 | 153372.30 |

approach transforms each feature to have zero mean and unit variance, calculated as:

$$z = \frac{x - \mu}{\sigma} \tag{18}$$

where $x$ is the original feature value, $\mu$ is the mean, and $\sigma$ is the standard deviation of the feature across the training dataset. Fig. 3(b) presents violin plots of normalized charging demand for each of the 16 districts. Each violin plot shows the distribution, median, and interquartile range of the normalized demand, highlighting the variability and central tendency across different zones.

### 5.1.2. Evaluation metrics

To evaluate the accuracy of parking demand prediction, we employ four widely used metrics: mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). These metrics are defined as follows:

$$\text{MAE} = \frac{1}{N \times T'} \sum_{t=1}^{T'} \sum_{i=1}^{N} \left| \hat{Y}_{t,i} - Y_{t,i} \right| \tag{19}$$

$$\text{RMSE} = \sqrt{\frac{1}{N \times T'} \sum_{t=1}^{T'} \sum_{i=1}^{N} \left( \hat{Y}_{t,i} - Y_{t,i} \right)^2} \tag{20}$$

$$\text{MAPE} = \frac{100\%}{N \times T'} \sum_{t=1}^{T'} \sum_{i=1}^{N} \left| \frac{\hat{Y}_{t,i} - Y_{t,i}}{Y_{t,i}} \right| \tag{21}$$

where $T'$ denotes the number of time steps in the evaluation period (validation or testing set), $N$ is the number of parking locations, $\hat{Y}_{t,i}$ is the predicted demand, and $Y_{t,i}$ is the ground truth demand at time $t$ and location $i$.

### 5.1.3. Benchmark models

To assess the robustness and predictive power of the proposed model, we compare it with a wide range of classical and deep learning-based forecasting methods:

- RNN [41]: A neural network architecture designed for sequential data modeling, capable of capturing temporal dynamics through recurrent connections, suitable for tasks such as parking demand prediction.
- LSTM [42]: An enhanced RNN architecture that introduces memory cells and gating mechanisms to effectively preserve long-term dependencies, widely used in complex time series modeling tasks.
- Gated Recurrent Unit (GRU) [43]: A lightweight variant of LSTM that retains gating mechanisms with fewer parameters, offering efficient training and effective modeling of short- to medium-term dependencies.
- Graph Convolutional Network (GCN) [44]: Captures spatial dependencies among parking locations using a graph structure and extracts spatial features via graph convolution operations.
- Graph Attention Network (GAT) [45]: Extends GCN by incorporating attention mechanisms that assign different weights to neighboring nodes, allowing more flexible modeling of spatial dependencies.
- GraphSAGE [46]: A scalable graph neural network approach that employs neighborhood sampling and aggregation strategies, enabling efficient representation learning on large-scale graphs.
- Temporal Graph Convolutional Network (TGCN) [47]: Combines GCN and GRU to jointly model spatial and temporal dependencies, suitable for spatio-temporal sequence prediction tasks.

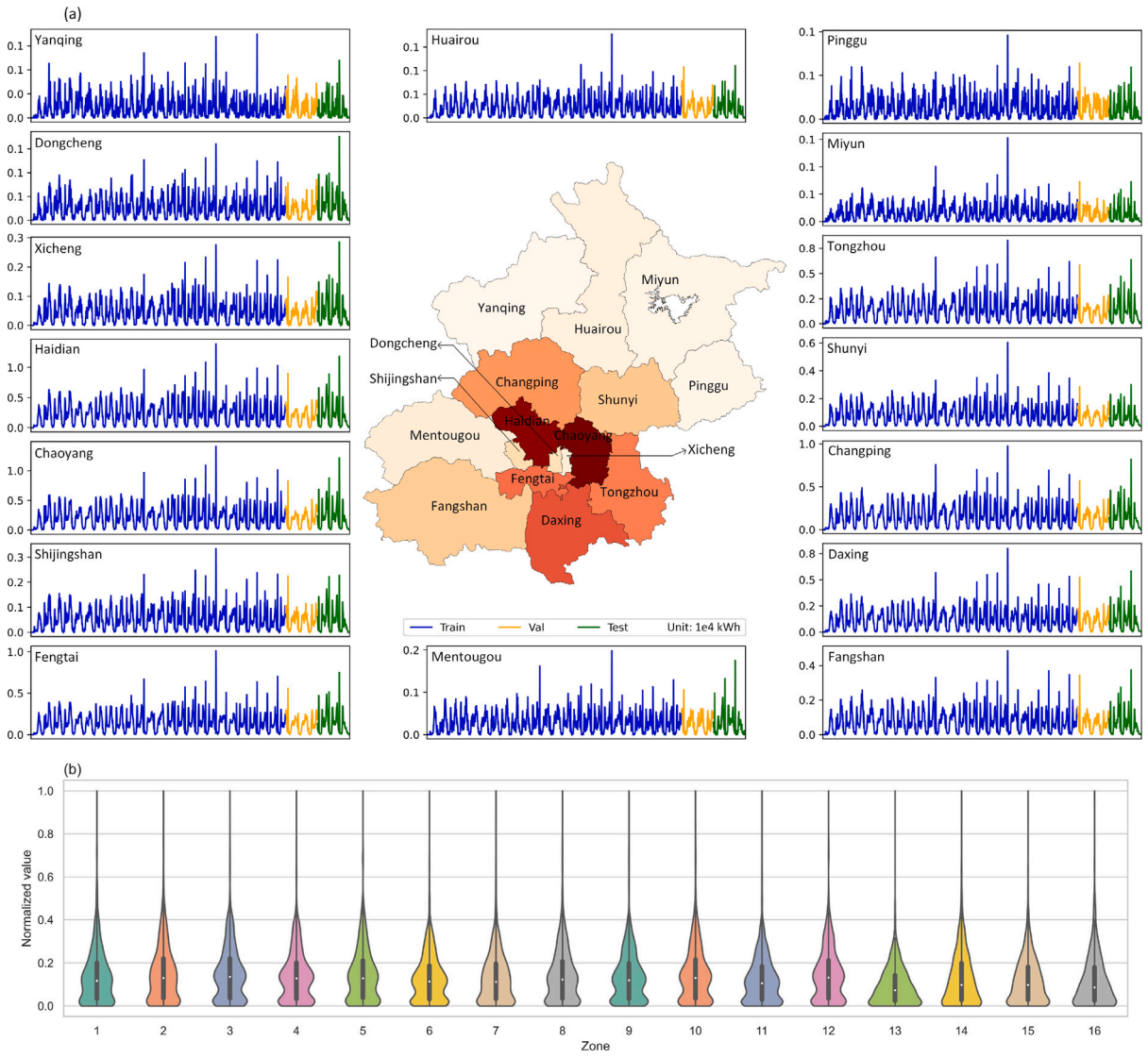


**Fig. 3.** Distribution of the EV charging demand in one month by Beijing districts, (a) temporal distribution of charging demand, (b) violin plot of normalized charging demand by zone.

- Temporal Graph Attention Network (TGAT) [48]: Enhances TGCN by introducing time encoding and attention mechanisms to capture the temporal dynamics affecting node representations more precisely.
- Temporal GraphSAGE (TSAGE) [8]: An extension of GraphSAGE to temporal graphs, using time-aware sampling and aggregation to capture dynamic features in evolving graph structures.

### 5.1.4. Model settings

The main hyperparameters of the proposed model are shown in Table 2. Besides, to ensure a comprehensive and unbiased comparison, we structured the experiment as follows: (1) The model's output comprises the charging demand of all districts or subdistrict s for the subsequent 1 timeslot and 2 timeslots. (2) For fair comparison, all baseline models use the same hyperparameters as the proposed model. Additionally, four prediction scenarios (S) are proposed, as follows:

- S1: Predicting the charging demand for the next time interval across 16 districts;
- S2: Predicting the charging demand for the next two time intervals across 16 districts;
- S3: Predicting the charging demand for the next time interval across 331 subdistricts;
- S4: Predicting the charging demand for the next two time intervals across 331 subdistricts.

(3) The proposed model is implemented using PyTorch on a workstation equipped with a GeForce RTX 3090 Ti GPU. The model utilizes a mean squared error loss function. The Ranger optimizer, which integrates the RAdam and LookAhead strategies, is known for its ability to retain the efficient convergence properties of Adam while enhancing model generalization through the LookAhead mechanism. In this study, the learning rate is set to 0.001, with a weight decay coefficient of 0.0001. The training process is conducted over 100 epochs. The libraries utilized in code and their exact versions used in the experiments are specified in Table 2.

## 5.2. Comparison results

The comparison of performance of different models on various metrics at district and subdistrict scales are shown in Table 3. In the task of multi-scale EV charging demand forecasting, the EV-STLLM model demonstrates outstanding performance, particularly in district-level

**Table 2**
Detailed Python libraries and their exact versions.

| Library | Version |
|---|---|
| numpy | 1.26.4 |
| pandas | 2.2.3 |
| Python | 3.12.3 |
| torch | 2.3.0+cu121 |
| torchvision | 0.18.0+cu121 |
| transformers | 4.49.0 |
| matplotlib | 3.9.0 |

and subdistrict-level predictions across different forecasting horizons. For the district-level one-step prediction (S1), EV-STLLM achieves a mean absolute error (MAE) of 218.13, representing a reduction of approximately 15.4% compared to the next-best model, GRU, which records an MAE of 257.86. In terms of root mean square error (RMSE), EV-STLLM scores 680.54, significantly outperforming all other models. The mean absolute percentage error (MAPE) is only 1.39%, a reduction of 53.5% compared to GraphSAGE's 2.99%. These results indicate that EV-STLLM is more capable of capturing complex spatial dependencies and short-term temporal dynamics across regions. Furthermore, in the more challenging district-level two-step prediction (S2), EV-STLLM continues to lead with an MAE of 242.35, which is approximately 34.0% lower than GRU's 366.93. The RMSE reaches 693.39, again outperforming all graph- and sequence-based models. The MAPE further drops to 1.29%, more than halving that of GraphSAGE (2.94%). This controlled increase in error across time steps highlights EV-STLLM's superior ability in long-term temporal dependency modeling and spatiotemporal trend learning, surpassing traditional RNN/LSTM and graph-based methods in multi-step forecasting tasks.

At a finer granularity, EV-STLLM also exhibits strong generalization capability in subdistrict-level predictions. For one-step forecasting at the subdistrict scale (S3), EV-STLLM achieves an MAE of 33.80, which is 19.6% lower than that of RNN (42.04), and an RMSE of 53.00, 26.4% lower than RNN's 71.99. Although MAPE slightly exceeds that of GRU (1.82% vs. 1.21%), this can be attributed to the smaller magnitude of subdistrict-level demand, making MAPE more sensitive to small values. Consequently, MAE and RMSE remain the more reliable indicators for operational decision-making in this context. In the two-step subdistrict-level prediction task (S4), EV-STLLM maintains its advantage, with MAE and RMSE of 36.91 and 66.13, respectively—representing reductions of approximately 16.7% and 14.0% when compared to GRU. While the MAPE is slightly higher (1.78% vs. 1.23%), its practical impact can be mitigated through weighted business metrics that emphasize absolute error control and stability in scheduling.

From a comparative perspective, traditional sequence models such as RNN, LSTM, and GRU exhibit certain strengths in temporal dependency modeling. However, they often suffer from information loss and struggle to simultaneously extract both local and global features in complex urban spatiotemporal interactions. Graph-based models like GCN, GAT, and GraphSAGE, while effective in capturing spatial adjacency, tend to lack the capacity to model temporal dynamics, focusing primarily on static topological relationships. Although spatiotemporal graph models such as TGCN, TGAT, and TSAGE attempt to integrate both spatial and temporal information, their reliance on localized convolutional or gated mechanisms constrains their ability to comprehensively fuse global dependencies. This can be attributed to several inherent limitations:

- Static Graph Structures: These models rely on fixed, predefined spatial graphs, which fail to capture dynamic inter-region relationships arising from temporal shifts in EV charging demand. For instance, regions that are spatially disconnected might exhibit strong temporal correlations that static adjacency matrices cannot represent.
- Local Dependency Bias: Traditional GNNs aggregate information from local neighborhoods, lacking the capacity to model global patterns. While extensions like TGAT attempt to incorporate temporal dynamics, their performance is constrained by the localized nature of their graph convolution mechanisms.
- Dependency on Graph Construction: The effectiveness of these models heavily depends on the quality of the constructed graphs. In real-world urban scenarios, accurately modeling human mobility or vehicle flow through adjacency matrices is challenging and often leads to suboptimal graph structures. Such inaccuracies propagate through the model, degrading its performance.

In contrast, our proposed EV-STLLM framework eliminates the dependency on explicit graph structures by adopting an attention mechanism capable of capturing long-range spatiotemporal dependencies directly from raw data. This not only addresses the limitations of graph-based models but also allows EV-STLLM to generalize effectively across regions with ambiguous or dynamically shifting relationships. The incorporation of embedding techniques further enhances its ability to integrate heterogeneous features, offering a more comprehensive understanding of urban dynamics. Moreover, the incorporation of LoRA enables efficient parameter tuning, facilitating flexible adaptation across diverse application scenarios. Overall, EV-STLLM achieves lower errors and higher robustness across multiple forecasting levels and time horizons, showcasing not only technical sophistication in model architecture and learning mechanisms but also practical value in supporting large-scale urban demand forecasting.

Fig. 4 presents a comparison between the EV-STLLM predicted charging demand curves (orange) and the actual observed curves (blue) across 16 different districts. It can be observed that the overall trend fitting is satisfactory, with the model accurately capturing the periodic variations in daily charging demand, such as the rise in the morning and the decline at night. The model also exhibits strong adaptability to pattern shifts between weekdays and weekends. During major peak periods (e.g., morning and evening peaks) and trough periods, the predicted curves closely align with the ground truth, demonstrating the model's excellent capability in extracting temporal features. Although certain deviations occur during extreme surges (e.g., around holidays), the magnitude remains well-controlled, and the fluctuation trends are consistent, indicating strong robustness in handling abnormal demand variations. In addition, the model is capable of promptly responding to sudden load changes, accurately reflecting turning points in the demand curves, which highlights its sensitivity and rapid adaptability to load fluctuations. In summary, at the district scale, EV-STLLM effectively captures the macroscopic temporal trends of electric vehicle charging demand across wide urban spaces, providing reliable support for city-level energy scheduling and charging infrastructure optimization.

Fig. 5 illustrates the prediction results for 16 consecutive subdistricts from zone 279 to zone 294. Compared to the district scale, the subdistrict data exhibit greater randomness and sparsity, with more frequent small-scale fluctuations. Due to the smaller base values within subdistricts, minor variations are amplified, resulting in sharper and more irregular curves. Nevertheless, EV-STLLM maintains good trend fitting across most subdistricts, demonstrating strong stability. For subdistricts with extensive periods of zero or very low demand, the model effectively avoids overfitting to zero values and preserves reasonable nonlinear fitting, reflecting sound regularization capabilities. The predicted curves generally synchronize with the actual changes at turning points of sudden demand surges or drops, although slight smoothing is observed in extremely sparse regions. When facing occasional demand spikes (e.g., caused by local events), the model partially captures the surges but tends to slightly underfit, suggesting that future improvements could incorporate anomaly detection mechanisms. Overall, the subdistrict-scale evaluation verifies EV-STLLM's generalization ability

**Table 3**
Comparison of performance of different models on various metrics at district and subdistrict scales.

| Zone | District | | | | | | Subdistrict | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Output | 1 | | | 2 | | | 1 | | | 2 | | |
| Metric | MAE | RMSE | MAPE | MAE | RMSE | MAPE | MAE | RMSE | MAPE | MAE | RMSE | MAPE |
| RNN | 322.13 | 700.32 | 3.12 | 357.78 | 732.39 | 3.52 | 42.04 | 71.99 | 1.26 | 47.28 | 80.58 | 1.58 |
| LSTM | 260.57 | 656.21 | 6.32 | 377.50 | 766.39 | 3.10 | 47.48 | 79.99 | 1.60 | 47.91 | 79.13 | 1.59 |
| GRU | 257.86 | 611.10 | 3.22 | 366.93 | 748.39 | 3.40 | 42.16 | 72.56 | **1.21** | 44.31 | 76.89 | **1.23** |
| GCN | 518.94 | 956.72 | 4.63 | 538.07 | 994.02 | 4.28 | 50.31 | 93.94 | 1.38 | 51.69 | 96.97 | 1.37 |
| GAT | 464.51 | 925.70 | 3.28 | 501.10 | 948.86 | 3.38 | 50.87 | 91.93 | 1.33 | 50.70 | 92.33 | 1.44 |
| GraphSAGE | 362.85 | 828.52 | 2.99 | 394.88 | 813.19 | 2.94 | 47.54 | 83.35 | 1.27 | 47.48 | 83.30 | 1.28 |
| TGCN | 567.90 | 992.81 | 5.61 | 550.34 | 932.13 | 6.37 | 54.55 | 102.84 | 1.41 | 52.32 | 97.05 | 1.57 |
| TGAT | 557.77 | 966.54 | 3.47 | 499.07 | 885.13 | 4.94 | 54.15 | 101.84 | 1.46 | 51.47 | 90.69 | 1.63 |
| TSAGE | 468.83 | 872.62 | 3.64 | 474.96 | 859.45 | 5.08 | 46.45 | 82.62 | 1.25 | 47.78 | 81.71 | 1.37 |
| EV-STLLM | **218.13** | 680.54 | **1.39** | **242.35** | **693.39** | **1.29** | **33.80** | **53.00** | 1.82 | **36.91** | **66.13** | 1.78 |

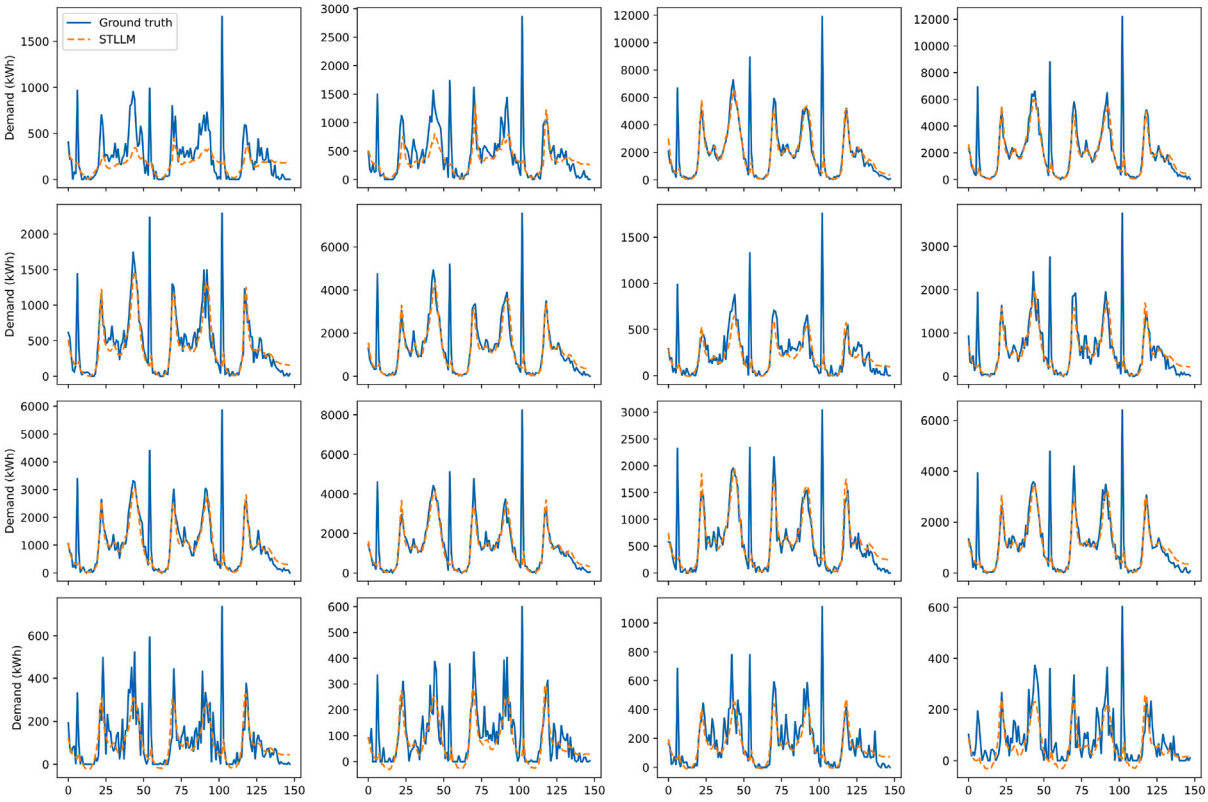


**Fig. 4.** Comparison between ground truth and EV-STLLM predictions across 16 districts.

under small-sample, high-noise environments, confirming its adaptability and robustness in fine-grained spatial predictions, thus providing highly reliable support for regional charging strategy formulation.

Fig. 6 shows the spatial distribution comparisons of charging demand across 16 districts at three typical timeslots: 9 AM (weekday peak), 3 PM (weekday valley), and 8 PM (evening peak). It is evident that EV-STLLM effectively reconstructs the spatial distribution characteristics of high-demand (central urban areas) and low-demand (suburban areas) regions at different times. Throughout the day, the model accurately captures the dynamic shifts of demand centers, such as concentration in business areas during the morning and expansion towards residential areas during the evening. Regarding color scale variations, the model successfully reflects the demand intensity differences between regions, further demonstrating its strong capability in modeling inter-regional demand disparities. Moreover, EV-STLLM not only maintains the consistency of overall spatial distribution trends but also replicates certain local spatial details, such as small high-demand clusters in specific districts during particular times, showcasing its high spatial resolution. These results further confirm that at the district scale, EV-STLLM possesses outstanding capabilities in geographic spatial perception and spatiotemporal evolution modeling, providing a scientific basis for optimizing the layout of urban charging infrastructure, balancing loads, and energy scheduling.

Fig. 7 presents the heatmap comparisons of charging demand for 331 subdistricts at the same three typical timeslots. Despite the large number of subdistricts and the complex spatial distribution, EV-STLLM accurately captures the locations and intensities of most major high-demand subdistricts. During peak times (9 AM and 8 PM), the model precisely identifies demand hotspots, such as the core urban areas and surrounding hotspot subdistricts. At valley times (3 PM), it reasonably reflects the overall sparse demand characteristics and accurately locates small localized hotspots. It is noteworthy that the predictions

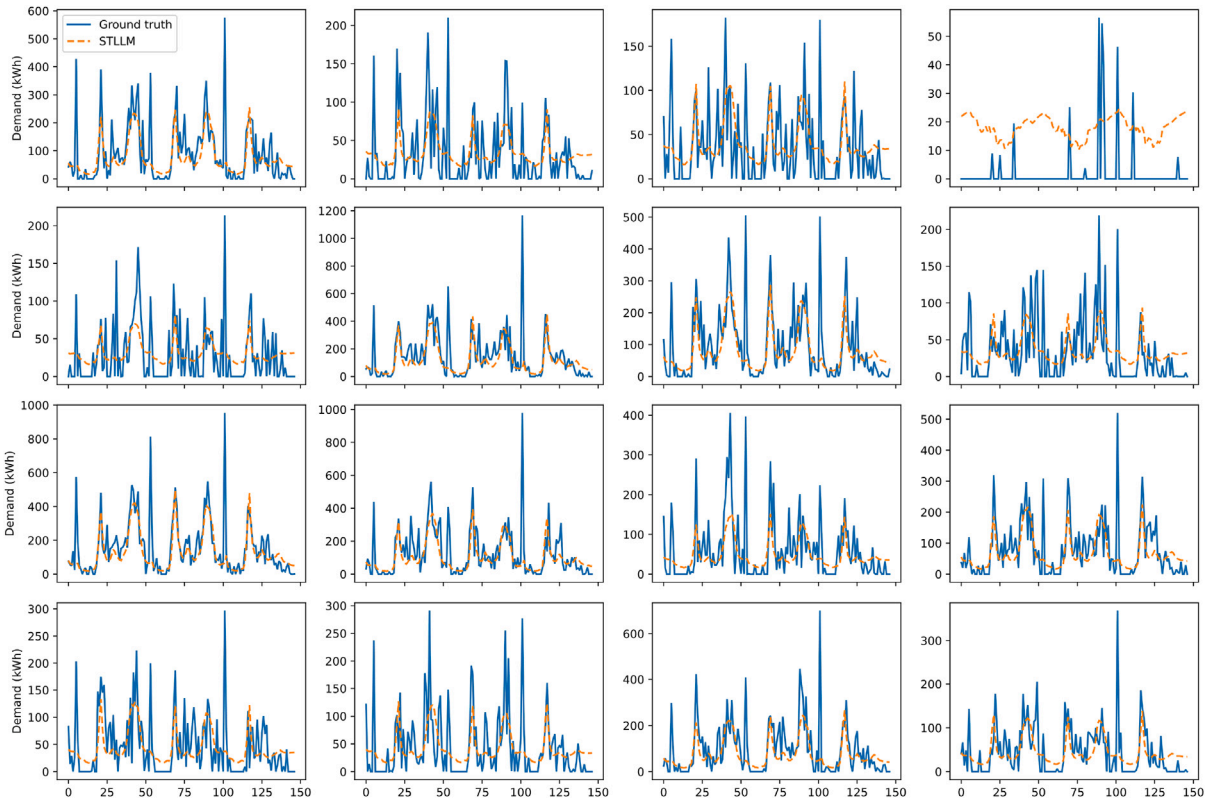


**Fig. 5.** Comparison between ground truth and EV-STLLM predictions across 16 consecutive subdistricts (zone 279 to zone 294).

**Table 4**
Ablation study of the proposed framework.

| Zone | District | | | | | | Subdistrict | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Output | 1 | | | 2 | | | 1 | | | 2 | | |
| Metric | MAE | RMSE | MAPE | MAE | RMSE | MAPE | MAE | RMSE | MAPE | MAE | RMSE | MAPE |
| EV-STLLM | **218.13** | **680.54** | **1.39** | 242.35 | 693.39 | **1.29** | **33.80** | **53.00** | **1.82** | 36.91 | 66.13 | **1.78** |
| FFT | 235.25 | 682.73 | 1.76 | **240.39** | **688.89** | 1.62 | 38.83 | 74.28 | 2.85 | **36.56** | **62.80** | 1.98 |
| FF | 226.77 | 687.08 | 1.53 | 253.78 | 704.42 | 1.34 | 38.39 | 69.97 | 2.08 | 38.14 | 69.84 | 1.95 |
| NonLLM | 349.10 | 813.01 | 4.05 | 437.87 | 845.46 | 5.43 | 43.90 | 78.47 | 1.92 | 44.02 | 77.62 | 2.33 |

remain stable across numerous low-demand subdistricts without significant overestimation or underestimation, demonstrating the model's excellent adaptability and stability in sparse data spaces. Furthermore, EV-STLLM effectively captures both spatial continuity and local spatial heterogeneity, such as sudden demand changes in peripheral subdistricts. Overall, the subdistrict-scale heatmap analysis verifies EV-STLLM's robust modeling capabilities in ultra-large-scale, micro-spatial prediction tasks, laying a solid foundation for future dynamic load forecasting, demand-driven deployment of charging infrastructure, and intelligent scheduling optimization based on geographic units.

### 5.3. Ablation study

We conduct a comprehensive comparison among the following methods:

- EV-STLLM: The proposed method, utilizing LoRA for efficient fine-tuning by adjusting only a small subset of parameters, enabling effective knowledge transfer while maintaining a lightweight model design.
- Full Fine-Tuning (FFT): A conventional approach where all parameters of the pre-trained LLM are fine-tuned, resulting in higher computational cost and risk of overfitting.
- Full Frozen LLM (FF): A variant where the LLM backbone is kept frozen, and only the task-specific modules are trained, aiming to utilize the frozen knowledge of LLM without modification.
- NonLLM: A baseline model without any LLM component, relying solely on task-specific modules for spatial prediction, serving as a control to assess the contribution of LLMs.

As shown in Table 4 and Fig. 8, we observe the following key findings. In Scenario S1 (Fig. 8(a)), EV-STLLM achieves the best performance across all metrics, achieving the lowest MAE (218.13), RMSE (680.54), and MAPE (1.39). The performance gap is especially evident in MAPE, highlighting EV-STLLM's superior capability in controlling relative errors. FFT and FF perform slightly worse, while NonLLM shows the worst results with significantly higher errors, confirming the indispensable role of LLM-based spatial feature extraction. In Scenario S2 (Fig. 8(b)), although FFT slightly outperforms EV-STLLM in MAE

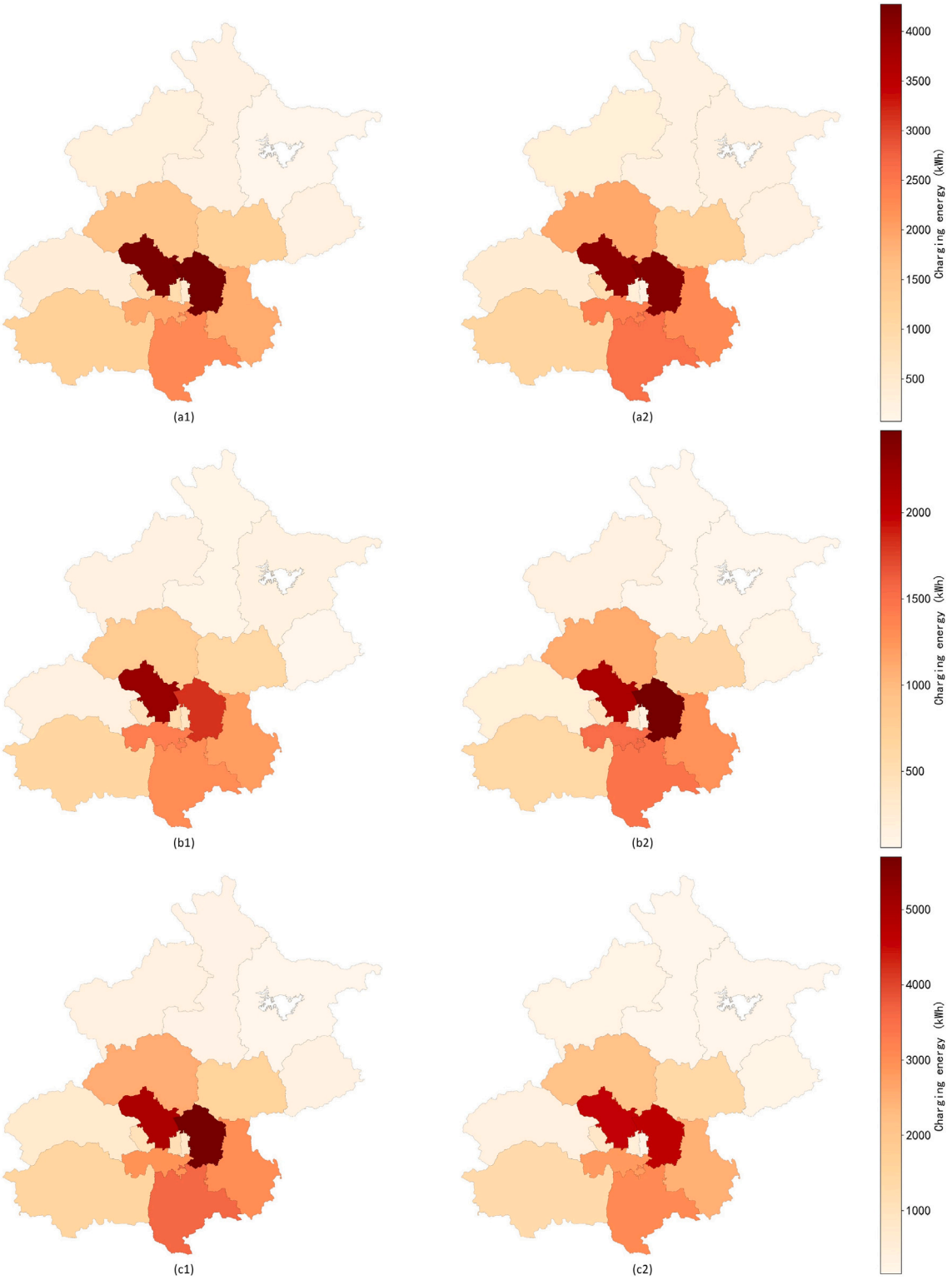


**Fig. 6.** Heatmap comparisons for 16 districts at different timeslots between ground truth (left) and prediction (right): (a) 9 AM peak; (b) 3 PM valley; (c) 8 PM peak.

(240.39 vs. 242.35) and RMSE (688.89 vs. 693.39), EV-STLLM maintains the lowest MAPE (1.29), suggesting better robustness. This indicates that EV-STLLM is more reliable for controlling relative deviations, which is crucial in heterogeneous spatial prediction tasks.

In Scenario S3 (Fig. 8(c)), EV-STLLM consistently leads with the lowest MAE (33.80) and RMSE (53.00). Although NonLLM achieves a comparable MAPE (1.92 vs. EV-STLLM's 1.82), its MAE and RMSE are much higher, indicating poor stability and weaker generalization

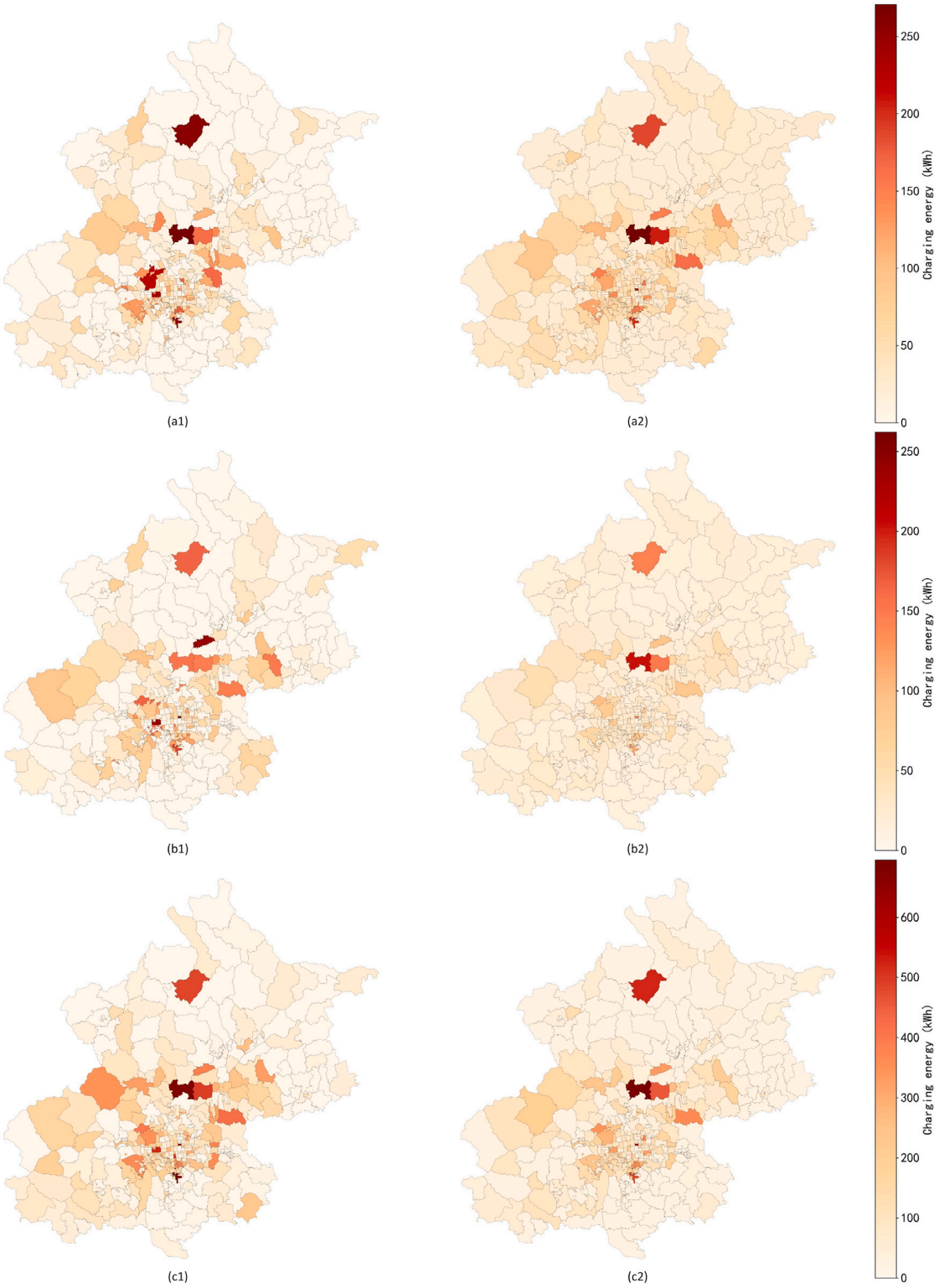


**Fig. 7.** Heatmap comparisons for 331 subdistricts at different timeslots between ground truth (left) and prediction (right): (a) 9 AM peak; (b) 3 PM valley; (c) 8 PM peak.

capabilities. In Scenario S4 (Fig. 8(d)), both EV-STLLM and FFT perform closely in terms of MAE (36.91 vs. 36.56) and RMSE (66.13 vs. 62.80), but EV-STLLM achieves the best MAPE (1.78). FF performs moderately, but NonLLM consistently underperforms across all metrics.

Overall, these results consistently demonstrate that EV-STLLM strikes the best trade-off between accuracy, efficiency, and scalability. While FFT occasionally achieves slightly better absolute metrics, it requires significantly higher computational resources, making it less

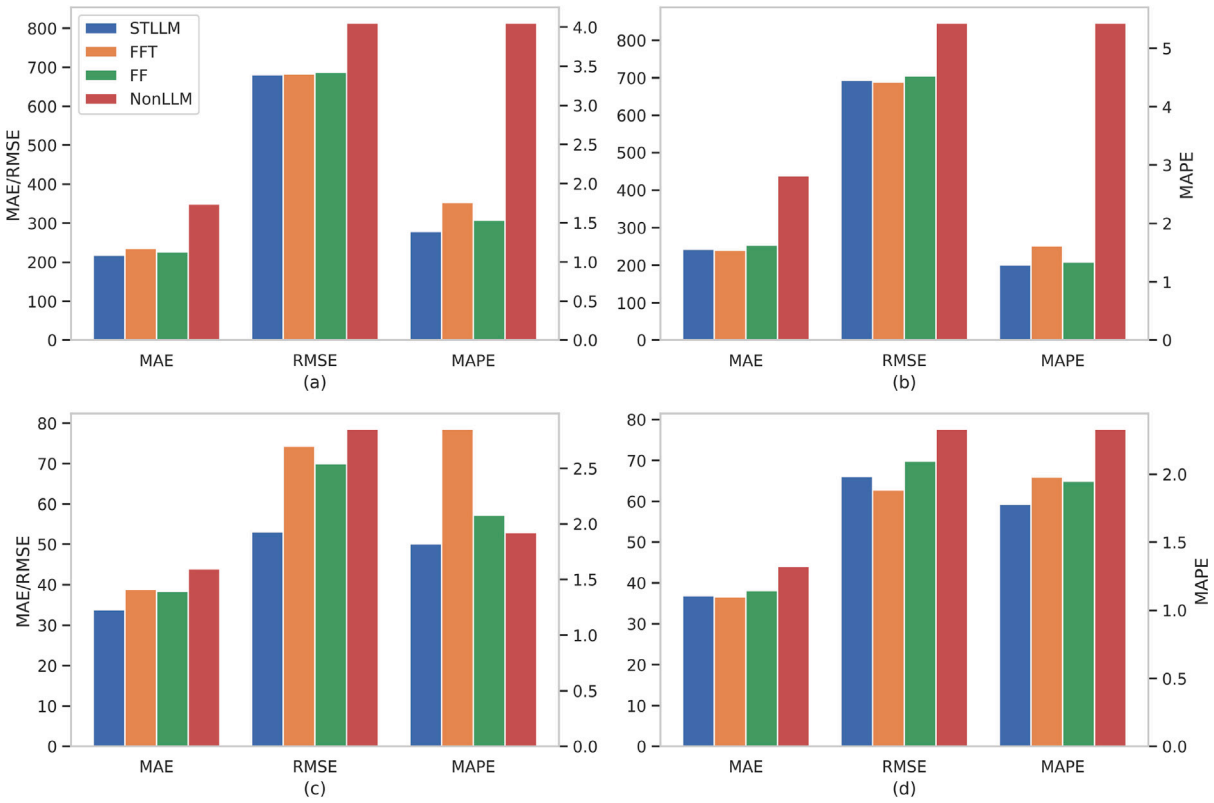


**Fig. 8.** Ablation study across different scenarios: (a) S1, (b) S2, (c) S3, (d) S4.

**Table 5**
Comparison of GPU memory and parameter update fraction across different ablation study Strategies.

| Zone | District | | Subdistrict | | Parameter Update Fraction |
|---|---|---|---|---|---|
| Output | 1 | 2 | 1 | 2 | |
| EV-STLLM | 1.41GB | 1.41GB | 18.87GB | 18.85GB | 0.49% |
| FFT | 2.07GB | 2.07GB | 18.84GB | 18.76GB | 100% |
| FF | 1.28GB | 1.28GB | 15.93GB | 15.85GB | 0% |

preferable for practical usage. On the other hand, EV-STLLM, with its parameter-efficient LoRA-based design, achieves competitive or superior performance at a much lower cost. Moreover, the consistently poor performance of NonLLM across all scenarios validates the necessity of incorporating LLMs for effective spatial modeling. In conclusion, EV-STLLM stands out as a robust, efficient, and scalable solution for fine-grained spatial prediction tasks.

To further evaluate the efficiency of the proposed EV-STLLM framework, we present a comparison of GPU memory usage and parameter update fraction across different ablation strategies, as shown in Table 5. This analysis highlights the computational cost and fine-tuning efficiency associated with each approach. Specifically, EV-STLLM demonstrates remarkable memory efficiency, requiring only 1.41 GB of GPU memory for District-level outputs and approximately 18.86 GB for Subdistrict-level outputs. Despite its low memory footprint, EV-STLLM updates merely 0.49% of the total parameters, owing to its LoRA-based fine-tuning strategy. This minimal update fraction enables efficient adaptation while preserving the general knowledge of the frozen LLM backbone. In contrast, the FFT strategy consumes more GPU memory than EV-STLLM at the District level (2.07 GB vs. 1.41 GB), and requires a full 100% parameter update, which significantly increases the computational burden. At the Subdistrict level, however, the GPU memory usage of FFT (18.84 GB and 18.76 GB) is slightly lower than that of EV-STLLM (18.87 GB and 18.85 GB), suggesting comparable memory demands in larger-scale scenarios. Nevertheless, the full parameter update requirement of FFT still makes it computationally intensive and less scalable for deployment in resource-constrained environments. The FF variant is the most memory-efficient approach, requiring only 1.28 GB (District) and 15.89 GB (Subdistrict) of GPU memory. However, since it keeps the LLM backbone entirely frozen (0% parameter update), its performance is generally inferior to EV-STLLM, as it lacks adaptability to task-specific spatial patterns.

Overall, these results reinforce that EV-STLLM strikes an optimal balance between performance and efficiency. By leveraging LoRA for lightweight fine-tuning, it significantly reduces memory consumption and training overhead while maintaining or exceeding the predictive accuracy of more resource-intensive methods. This makes EV-STLLM a practical and scalable solution for large-scale spatial prediction tasks.

### 5.4. Sensitive analysis

Table 6 and Fig. 9 present the sensitivity analysis results with regard to different input sequence lengths (6, 12, 18, 24, 30). The evaluation is conducted for both district- and subdistrict-level tasks across two output scenarios.

For the district-level tasks, as shown in Figs. 9 (a1) and (b1), the MAE and RMSE metrics generally fluctuate within a moderate range with varying sequence lengths. In Output 1, the lowest MAE (231.82) and RMSE (681.53) are observed at a sequence length of 18, indicating that moderate-length sequences provide more stable and

**Table 6**
Sensitivity analysis of different sequence lengths.

| Zone | Output | Metric | Statistic | 6 | 12 | 18 | 24 | 30 |
|---|---|---|---|---|---|---|---|---|
| District | 1 | MAE | Mean | 234.96 | 237.98 | 231.82 | 258.57 | 249.07 |
| | | | Std | 6.62 | 13.20 | 11.16 | 22.50 | 15.87 |
| | | RMSE | Mean | 695.44 | 691.72 | 681.53 | 708.96 | 701.95 |
| | | | Std | 3.13 | 12.14 | 7.98 | 13.56 | 12.39 |
| | | MAPE | Mean | 1.53 | 1.55 | 1.69 | 1.93 | 1.76 |
| | | | Std | 0.31 | 0.27 | 0.18 | 0.45 | 0.26 |
| | 2 | MAE | Mean | 252.60 | 258.87 | 248.06 | 259.33 | 244.53 |
| | | | Std | 8.69 | 26.30 | 11.42 | 20.75 | 14.10 |
| | | RMSE | Mean | 700.96 | 704.94 | 695.69 | 708.57 | 702.81 |
| | | | Std | 11.35 | 26.56 | 11.98 | 18.63 | 16.07 |
| | | MAPE | Mean | 1.39 | 1.41 | 1.52 | 1.64 | 1.52 |
| | | | Std | 0.17 | 0.21 | 0.13 | 0.22 | 0.13 |
| Subdistrict | 1 | MAE | Mean | 37.56 | 36.67 | 36.23 | 36.23 | 36.78 |
| | | | Std | 1.07 | 0.80 | 1.26 | 1.43 | 1.57 |
| | | RMSE | Mean | 69.29 | 63.43 | 62.15 | 64.33 | 65.98 |
| | | | Std | 3.59 | 4.51 | 6.15 | 6.01 | 6.01 |
| | | MAPE | Mean | 1.96 | 2.16 | 2.11 | 1.93 | 1.98 |
| | | | Std | 0.35 | 0.23 | 0.27 | 0.14 | 0.28 |
| | 2 | MAE | Mean | 37.61 | 37.42 | 37.02 | 37.97 | 37.59 |
| | | | Std | 0.68 | 0.53 | 0.70 | 0.84 | 0.89 |
| | | RMSE | Mean | 67.36 | 67.49 | 65.72 | 68.45 | 66.99 |
| | | | Std | 3.20 | 2.08 | 3.27 | 2.38 | 2.97 |
| | | MAPE | Mean | 2.00 | 1.94 | 2.02 | 2.09 | 2.09 |
| | | | Std | 0.14 | 0.14 | 0.12 | 0.13 | 0.19 |

accurate predictions. However, overly long sequences (24 or 30) tend to slightly deteriorate the performance, possibly due to the introduction of noise and overfitting issues. The MAPE metric follows a similar trend, with longer sequences leading to higher relative errors. For Output 2 at the district level, a similar pattern is observed. The best performance is obtained around sequence lengths of 18 and 30, while 24 shows a noticeable increase in both MAE and MAPE. The standard deviations are also larger for longer sequences, suggesting less stability across different runs.

In the subdistrict-level tasks shown in Figs. 9 (c1) and (d1), the system demonstrates higher robustness to sequence length variations. The MAE and RMSE remain relatively stable across different lengths, but the shortest sequence (6) shows slightly inferior performance. In particular, the RMSE is minimized at a length of 18 for Output 1 and at 18–30 for Output 2, indicating that moderately long input sequences are advantageous for subdistrict predictions as well. The MAPE trends at the subdistrict level (Figs. 9 (c2) and (d2)) show that sequence lengths around 18 yield slightly better relative error control. However, the differences are minor compared to district-level results, showcasing the relatively smooth dynamics at finer spatial granularity.

The sensitivity analysis results provide practical insights that can guide the deployment of the proposed model in real-world scenarios:

- Recommended default configuration: Based on our results, we recommend using an input window of 12 to 18 time steps (i.e., 6 to 9 h) as a reliable and generalizable setting for short-term EV charging demand forecasting in urban environments. This range provides a practical balance between capturing sufficient temporal dependencies and maintaining computational efficiency, making it suitable for deployment in both real-time and resource-constrained scenarios.
- Long Sequences can Cause Prediction Instability: When input sequences exceed 18 time steps (9 h), prediction instability emerges, particularly at the district level, as indicated by increased variability in performance metrics (MAE, RMSE, MAPE). This instability stems from factors such as noise and redundancy introduced by irrelevant or outdated time dependencies, increased risk of overfitting due to longer sequences, and higher computational complexity, which amplifies inference time and resource challenges in real-time scenarios. Practically, this insight suggests that longer sequences should be avoided in deployment, especially for district tasks, to ensure stability and scalability in operational systems.
- Subdistrict Tasks are Less Sensitive to Sequence Length: Sensitivity analysis reveals that subdistrict-level predictions are more robust to sequence length variations than district-level predictions. This is due to smoother temporal dynamics in subdistrict data, which often exhibit less external interference and more stable demand patterns. Furthermore, subdistrict tasks mitigate the noise amplification associated with aggregated district-level data, offering clearer and more distinctive temporal signals. These findings can guide future deployment strategies by advocating for the adoption of adaptive sequence length approaches at the subdistrict level, leveraging shorter sequences to enhance computational efficiency during periods of low variability while extending sequence lengths to maintain accuracy during high-variability periods. These findings can guide future deployment by supporting the use of adaptive sequence length strategies at the subdistrict level, enabling computational efficiency during low-variability periods and maintaining accuracy during high-variability periods.

These interpretations underline the importance of tailoring the input sequence length to specific operational conditions. They also provide actionable guidelines to enhance both the accuracy and efficiency of the forecasting model in deployment scenarios.

### 5.5. Comparative evaluation of PEFT methods

This Section presents a comparative analysis of six mainstream Parameter-Efficient Fine-Tuning (PEFT) methods in terms of their prediction performance. The selected methods represent current research hotspots, including LoRA, IA3, Prefix Tuning, P-Tuning, P-Tuning-v2, and BitFit. Overall, LoRA consistently achieves the best results across all evaluation metrics and testing dimensions. In the District level, it attains the lowest MAE (218.13 in 1 output length and 242.35 in 2 output lengths), and also significantly outperforms other methods in terms of RMSE and MAPE. This suggests that LoRA possesses strong modeling ability in capturing local structural variations of the target function. Particularly at the Subdistrict level, LoRA achieves a remarkably low RMSE of 53.00 in 1 output length, substantially outperforming competitors such as IA3 (68.50) and BitFit (69.72), demonstrating its superior capacity to adapt to fine-grained spatial heterogeneity.

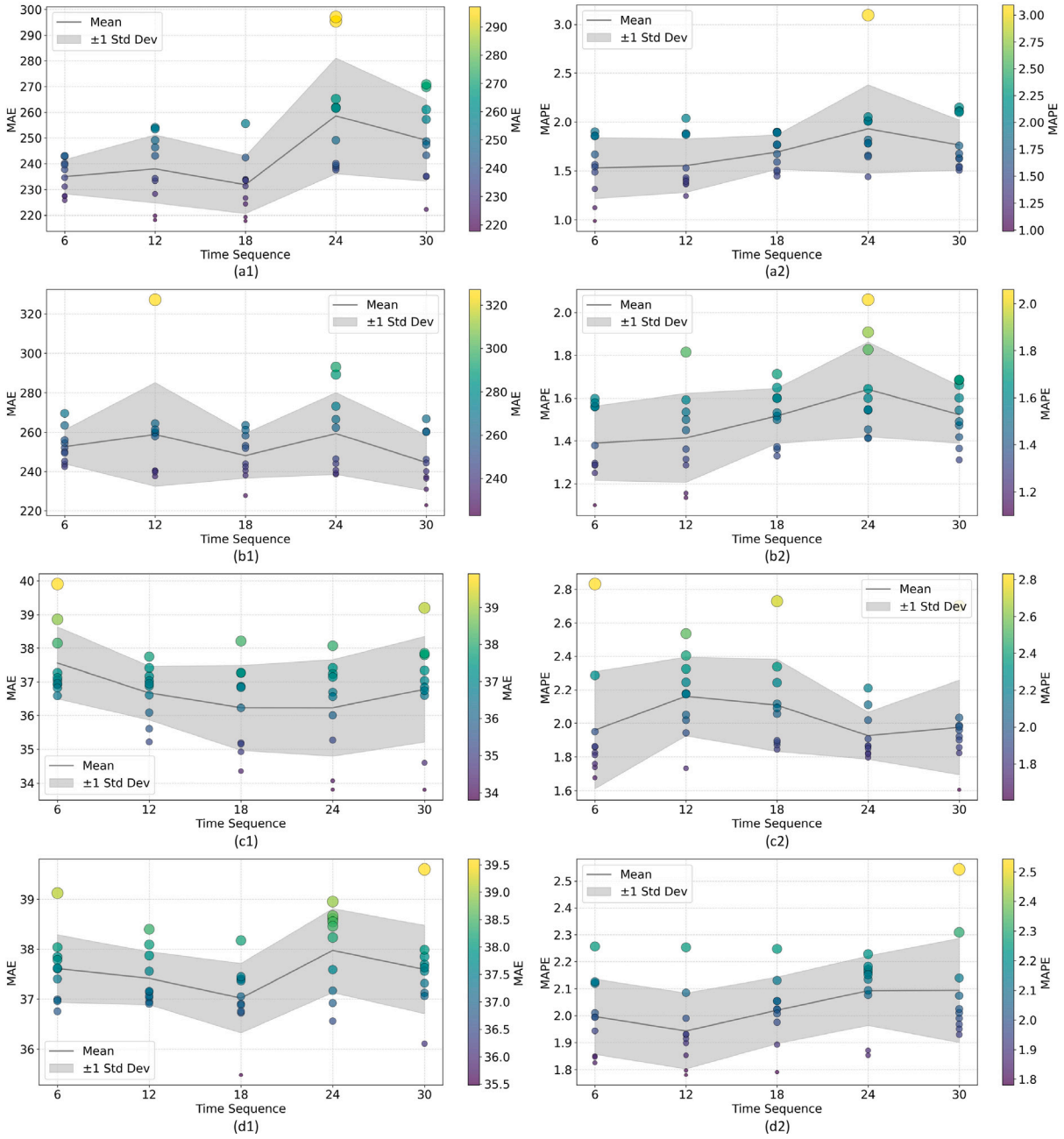


**Fig. 9.** Sensitivity analysis with MAE (1) and MAPE (2) across different time sequences lengths, (a) S1, (b) S2, (c) S3, (d) S4.

**Table 7**
Comparison of prediction performance of different parameter efficient fine-tuning methods.

| Zone | District | | | | | | Subdistrict | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Output | 1 | | | 2 | | | 1 | | | 2 | | |
| Metric | MAE | RMSE | MAPE | MAE | RMSE | MAPE | MAE | RMSE | MAPE | MAE | RMSE | MAPE |
| IA3 | 234.17 | 690.95 | 1.53 | 256.22 | 695.52 | 1.73 | 37.56 | 68.50 | 1.97 | 38.44 | 70.65 | 1.84 |
| P-Tuning-v2 | 585.26 | 998.78 | 8.08 | 580.53 | 993.93 | 7.55 | 39.67 | 72.70 | 1.97 | 38.68 | 70.04 | 1.89 |
| P-Tuning | 579.21 | 997.06 | 7.82 | 572.94 | 993.51 | 7.00 | 38.78 | 70.57 | 2.12 | 39.70 | 72.79 | 1.80 |
| Prefix | 574.49 | 996.52 | 7.53 | 569.65 | 993.61 | 6.85 | 40.16 | 73.13 | 2.16 | 40.07 | 72.01 | 2.06 |
| BitFit | 242.02 | 688.16 | 1.74 | 255.79 | 705.31 | 1.41 | 38.10 | 69.72 | 1.97 | 38.13 | 70.28 | 1.79 |
| LoRA | **218.13** | **680.54** | **1.39** | **242.35** | **693.39** | **1.29** | **33.80** | **53.00** | **1.82** | **36.91** | **66.13** | **1.78** |

IA3, which introduces a different parameter injection mechanism, performs comparably to LoRA at the District level, but its RMSE and MAPE at the Subdistrict level are slightly worse, indicating limited generalization when modeling at finer spatial scales. BitFit, despite modifying only a small subset of bias parameters, performs moderately or even second-best in many settings. It shows better MAE performance than the P-Tuning family, highlighting its relatively high

**Table 8**
Comparison of training and inference time (in seconds) of different parameter-efficient fine-tuning methods.

| Zone | District | | | | Subdistrict | | | |
|---|---|---|---|---|---|---|---|---|
| Output | 1 | | 2 | | 1 | | 2 | |
| Type | Training | Inference | Training | Inference | Training | Inference | Training | Inference |
| IA3 | 0.4605 | 0.0547 | 0.4532 | 0.0528 | 8.5828 | 0.8325 | 8.5588 | 0.8330 |
| P-Tuning-v2 | 0.7531 | 0.0865 | 0.7498 | 0.0860 | 8.3898 | 0.8287 | 8.3490 | 0.8274 |
| P-Tuning | 0.6846 | 0.0822 | 0.6853 | 0.0824 | 8.3174 | 0.8257 | 8.2929 | 0.8241 |
| Prefix | 0.6769 | 0.0815 | 0.6744 | 0.0816 | 8.3057 | 0.8233 | 8.2796 | 0.8228 |
| BitFit | 0.4332 | 0.0491 | 0.4299 | 0.0489 | 8.1539 | 0.7955 | 8.1389 | 0.7962 |
| LoRA | 0.5042 | 0.0581 | 0.5062 | 0.0582 | 8.9191 | 0.8816 | 8.2929 | 0.8241 |

cost-effectiveness in constrained parameter-update scenarios. The performance of P-Tuning and its variant P-Tuning-v2 is relatively poor in this experimental setup, especially at the District level, where their MAE and RMSE scores are significantly higher than those of other methods—sometimes approaching the performance of an unadapted base model. This may be due to the heavy reliance of these methods on extensive prompt token tuning, which is sensitive to task structure and less effective. Prefix Tuning, while enhancing prompt expressiveness relative to P-Tuning, still falls short of matching the performance of LoRA or IA3. Notably, at the Subdistrict level in 1 output length, it yields the highest MAPE of 2.16 among all methods, indicating potential limitations in modeling complex hierarchical spatial dependencies. Further analysis of prediction errors across different levels reveals that all methods exhibit significantly lower MAE and RMSE at the Subdistrict level compared to the District level. This may partially reflect the fact that fine-grained spatial prediction tasks are associated with smoother target functions or are easier to fit. However, the MAPE at the Subdistrict level shows greater variability, suggesting that normalized error metrics are more sensitive to prediction targets with low magnitude. This highlights the need to carefully choose evaluation metrics based on specific business requirements in real-world applications.

In addition to prediction accuracy, computational efficiency is a crucial factor when selecting PEFT strategies, particularly for deployment in resource-constrained environments. Table 8 presents a detailed comparison of training and inference time. We observe that BitFit consistently exhibits the fastest training and inference times across all scenarios, owing to its minimal parameter update design—only tuning bias parameters. IA3 also demonstrates low computational overhead, particularly in the District-level tasks, with training times under 0.5 s per epoch and inference times around 0.05 s. LoRA, while not the fastest, strikes a compelling balance between efficiency and performance. At the District level, its training and inference times (approximately 0.5 and 0.058 s respectively) are only marginally higher than those of IA3 and BitFit, but it far surpasses all other methods in prediction accuracy (see Table 7). For instance, LoRA achieves the lowest MAE and RMSE across all regions and output lengths, and delivers particularly strong results at the Subdistrict level—demonstrating its ability to model fine-grained spatial heterogeneity.

At the Subdistrict level, LoRA's training time ( 8.9 s per epoch for 1 output length) is slightly higher than other methods, but this overhead is justifiable given its substantial gains in predictive performance. Inference times remain comparable with other methods (e.g., 0.8816 s vs. 0.8325 for IA3), ensuring that LoRA remains practical for real-time or near-real-time applications. On the other hand, the P-Tuning family and Prefix Tuning methods, despite their expressiveness through prompt-based parameterization, incur longer training times (around 0.68–0.75 s at the District level and over 8.3 s at the Subdistrict level) and fail to offer competitive prediction accuracy. This suggests that their computational cost is not well-compensated by corresponding gains in model performance, making them less favorable in this context. In summary, combining both predictive accuracy and computational efficiency, LoRA emerges as the most balanced and effective PEFT method for the studied spatial–temporal prediction tasks.

## 6. Conclusions

In this paper, we propose a novel spatiotemporal learning framework, EV-STLLM, for short-term EV charging demand prediction in urban environments to tackle the core challenges of data fusion and model fusion. At the data level, our model constructs a collaborative embedding mechanism that fuses token-level, temporal, and spatial features, enabling fine-grained modeling of the nonlinear and dynamic patterns inherent in EV charging behavior. At the model level, we integrate a pretrained LLM as the backbone for deep spatiotemporal dependency modeling while significantly reducing training cost through LoRA, which freezes the bulk of model parameters and only tunes a small set of low-rank matrices.

Extensive experiments conducted on a large-scale real-world dataset from Beijing—covering 16 districts and 331 subdistricts, with over 830,000 charging records—demonstrate the superior performance of EV-STLLM across multiple evaluation metrics and prediction scenarios. Compared to classical sequence models (RNN, LSTM, GRU), graph-based models (GCN, GAT, GraphSAGE), and spatiotemporal graph models (TGCN, TGAT, TSAGE), EV-STLLM achieves consistent improvements across all tasks and scales: In district-level one-step prediction, EV-STLLM reduces MAE by 15.41% and MAPE by 53.51% compared to the best-performing baseline. In subdistrict-level prediction, despite the greater spatial granularity, EV-STLLM maintains a significant lead in both MAE and RMSE, showcasing its strong generalization capacity and robustness at fine spatial resolutions. To better understand the impact of input sequence length on prediction performance, we conduct a temporal sensitivity analysis by varying the length of historical time windows used in the model input. The results reveal the importance of selecting an appropriate temporal window that balances context depth and relevance. They also suggest potential for adaptive sequence learning, where the model dynamically adjusts its receptive field based on forecast horizon or local temporal patterns. Additionally, our ablation study confirms the effectiveness of each key component. Removing the LLM component (NonLLM) leads to substantial performance degradation, especially in MAPE, indicating the critical role of LLMs in capturing global dependencies. Compared with full fine-tuning (FFT) and fully frozen (FF) setups, our LoRA-based approach achieves comparable or even better prediction accuracy while significantly reducing computational cost, validating its practical value for scalable deployment.

### 6.1. Practical implications and limitations

The proposed EV-STLLM framework holds significant promise for real-world deployment in smart grid management and urban energy systems. However, its practical application also entails several considerations and limitations that must be carefully addressed.

On the one hand, for practical implications, Firstly, EV-STLLM enables accurate short-term forecasts of charging demand at both district and subdistrict levels. This allows grid operators to proactively allocate electricity resources, mitigate peak loads, and implement dynamic load balancing strategies. The fine-grained spatial resolution also supports zonal demand-response mechanisms. Secondly, by predicting temporal

variations in charging activity, EV-STLLM can inform adaptive time-of-use pricing strategies. This enables utility providers to incentivize off-peak charging, reduce grid stress, and align user behavior with system-level optimization objectives. And then, EV-STLLM facilitates hotspot detection and demand clustering at subdistrict scale, supporting optimal siting of new charging stations or mobile charging units. Long-term deployment planning can benefit from short-term demand dynamics, especially in rapidly evolving urban environments. Finally, given its efficient architecture with LoRA-based fine-tuning, EV-STLLM can be integrated into real-time decision support systems, such as charging station management platforms or urban energy digital twins, offering timely and localized predictions.

On the other hand, for limitations and future works, Firstly, while EV-STLLM performs well on Beijing data, its generalization to other cities with different urban topologies, charging behaviors, or infrastructure densities may be limited. Domain adaptation or federated learning approaches may be required for cross-city deployment. Secondly, the effectiveness of EV-STLLM depends heavily on the availability and accuracy of fine-grained charging logs, spatial metadata, and contextual features. In data-scarce regions, performance may degrade. Synthetic data generation or transfer learning could be explored to mitigate this issue. Thirdly, although LLM-based models offer strong representation capabilities, their decision-making process remains relatively opaque. For high-stakes applications (e.g., grid reliability or public safety), augmenting the model with explainable AI modules is crucial to enhance transparency and stakeholder trust. Last but not least, urban charging patterns are dynamic, affected by policy changes, infrastructure upgrades, and behavioral shifts. The model requires periodic retraining or online learning capabilities to remain accurate over time. Automated model updating pipelines should be considered in large-scale deployments.

## CRediT authorship contribution statement

**Yitong Shang:** Writing – original draft, Visualization, Methodology, Conceptualization. **Wen-Long Shang:** Writing – review & editing, Writing – original draft, Project administration, Methodology, Funding acquisition, Conceptualization. **Dingsong Cui:** Investigation, Conceptualization. **Peng Liu:** Data curation, Conceptualization. **Haibo Chen:** Validation, Project administration, Investigation. **Dongdong Zhang:** Visualization, Validation, Software. **Runsen Zhang:** Writing – review & editing. **Chengcheng Xu:** Writing – review & editing, Investigation. **Ye Liu:** Writing – review & editing, Methodology. **Chenxi Wang:** Writing – review & editing. **Mohannad Alhazmi:** Validation.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

This work was partially supported by the ZEV-UP and ePowerMove projects co-funded by the European Union under Grant agreement ID: 101138721 and 101192753. Besides, this research is supported in part by the Beijing Natural Science Foundation, China (No. 9232003). Also, this work would like to acknowledge the support provided by Researchers Supporting Project (Project number: RSPD2025R635), KingSaud University, Riyadh, Saudi Arabia.

## Data availability

The authors do not have permission to share data.

## References

[1] Martino Tran, David Banister, Justin D.K. Bishop, Malcolm D. McCulloch, Realizing the electric-vehicle revolution, Nat. Clim. Chang. 2 (5) (2012) 328–333.

[2] Aqib Zahoor, Yajuan Yu, Saima Batool, Muhammad Idrees, Guozhu Mao, The carbon-neutral goal in China for the electric vehicle industry with solid-state battery's contribution in 2035 to 2045, J. Environ. Eng. 149 (12) (2023) 04023082.

[3] Zhaoyao Bao, Jiapei Li, Xuehan Bai, Chi Xie, Zhibin Chen, Min Xu, Wen-Long Shang, Hailong Li, An optimal charging scheduling model and algorithm for electric buses, Appl. Energy 332 (2023) 120512.

[4] Hang Yu, Songyan Niu, Yitong Shang, Ziyun Shao, Youwei Jia, Linni Jian, Electric vehicles integration and vehicle-to-grid operation in active distribution grids: A comprehensive review on power architectures, grid connection standards and typical applications, Renew. Sustain. Energy Rev. 168 (2022) 112812.

[5] Yanchong Zheng, Songyan Niu, Yitong Shang, Ziyun Shao, Linni Jian, Integrating plug-in electric vehicles into power grids: A comprehensive review on power interaction mode, scheduling methodology and mathematical foundation, Renew. Sustain. Energy Rev. 112 (2019) 424–439.

[6] Zaoli Yang, Qin Li, Yamin Yan, Wen-Long Shang, Washington Ochieng, Examining influence factors of Chinese electric vehicle market demand based on online reviews under moderating effect of subsidy policy, Appl. Energy 326 (2022) 120019.

[7] Wen-Long Shang, Junjie Zhang, Kun Wang, Hangjun Yang, Washington Ochieng, Can financial subsidy increase electric vehicle (EV) penetration—evidence from a quasi-natural experiment, Renew. Sustain. Energy Rev. 190 (2024) 114021.

[8] Yitong Shang, Duo Li, Yang Li, Sen Li, Explainable spatiotemporal multi-task learning for electric vehicle charging demand prediction, Appl. Energy 384 (2025) 125460.

[9] Alexis Pengfei Zhao, Shuangqi Li, Zhengmao Li, Zhaoyu Wang, Xue Fei, Zechun Hu, Mohannad Alhazmi, Xiaohe Yan, Chenye Wu, Shuai Lu, et al., Electric vehicle charging planning: A complex systems perspective, IEEE Trans. Smart Grid (2024).

[10] Zhijie Lai, Sen Li, Towards a multimodal charging network: Joint planning of charging stations and battery swapping stations for electrified ride-hailing fleets, Transp. Res. Part B: Methodol. 183 (2024) 102928.

[11] Wenhao Wang, Aihong Tang, Feng Wei, Huiyuan Yang, Li Xinran, Jiao Peng, Electric vehicle charging load forecasting considering weather impact, Appl. Energy 383 (2025) 125337.

[12] Chahinez Ounoughi, Sadok Ben Yahia, Data fusion for ITS: A systematic literature review, Inf. Fusion 89 (2023) 267–291.

[13] Laura Melgar-García, David Gutiérrez-Avilés, Cristina Rubio-Escudero, Alicia Troncoso, A novel distributed forecasting method based on information fusion and incremental learning for streaming time series, Inf. Fusion 95 (2023) 163–173.

[14] Shams Forruque Ahmed, Sweety Angela Kuldeep, Sabiha Jannat Rafa, Javeria Fazal, Mahfara Hoque, Gang Liu, Amir H. Gandomi, Enhancement of Traffic Forecasting Through Graph Neural Network-Based Information Fusion Techniques, Elsevier, 2024.

[15] Niyaz Ahmad Wani, Ravinder Kumar, Jatin Bedi, Imad Rida, et al., Explainable AI-driven IoMT fusion: Unravelling techniques, opportunities, and challenges with explainable AI in healthcare, Inf. Fusion (2024) 102472.

[16] Fatima Hassan, Syed Fawad Hussain, Saeed Mian Qaisar, Fusion of multivariate EEG signals for schizophrenia detection using CNN and machine learning techniques, Inf. Fusion 92 (2023) 466–478.

[17] Hui Huang, Bing Xu, Xinnian Liang, Kehai Chen, Muyun Yang, Tiejun Zhao, Conghui Zhu, Multi-view fusion for instruction mining of large language model, Inf. Fusion 110 (2024) 102480.

[18] Lingshu Zhong, Ziling Zeng, Zikang Huang, Xiaowei Shi, Yiming Bie, Joint optimization of electric bus charging and energy storage system scheduling, Frontiers of Engineering Management 11 (4) (2024) 676–696.

[19] M. Hadi Amini, Amin Kargarian, Orkun Karabasoglu, ARIMA-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation, Electr. Power Syst. Res. 140 (2016) 378–390.

[20] Fei Ren, Chenlu Tian, Guiqing Zhang, Chengdong Li, Yuan Zhai, A hybrid method for power demand prediction of electric vehicles based on SARIMA and deep learning with integration of periodic features, Energy 250 (2022) 123738.

[21] Eiman ElGhanam, Mohamed Hassan, Ahmed Osman, Machine learning-based electric vehicle charging demand prediction using origin-destination data: A UAE case study, in: 2022 5th International Conference on Communications, Signal Processing, and their Applications, ICCSPA, IEEE, 2022, pp. 1–6.

[22] Shengyou Wang, Chengxiang Zhuge, Chunfu Shao, Pinxi Wang, Xiong Yang, Shiqi Wang, Short-term electric vehicle charging demand prediction: A deep learning approach, Appl. Energy 340 (2023) 121032.

[23] Zhiyan Yi, Xiaoyue Cathy Liu, Ran Wei, Xi Chen, Jiangpeng Dai, Electric vehicle charging demand forecasting using deep learning model, J. Intell. Transp. Syst. 26 (6) (2022) 690–703.

[24] Lijing Zhu, Wen-Long Shang, Jingzhou Wang, Yixin Li, Chulung Lee, Washington Ochieng, Xunzhang Pan, Diffusion of electric vehicles in Beijing considering indirect network effects, Transp. Res. Part D: Transp. Environ. 127 (2024) 104069.
[25] Nikolaos Tsalikidis, Paraskevas Koukaras, Dimosthenis Ioannidis, Dimitrios Tzovaras, Hybrid CNN-LSTM forecasting model for electric vehicle charging demand in smart buildings, in: 2024 6th Global Power, Energy and Communication Conference, GPECOM, IEEE, 2024, pp. 590–595.
[26] Zhiju Chen, Kai Liu, Jiangbo Wang, Toshiyuki Yamamoto, H-ConvLSTM-based bagging learning approach for ride-hailing demand prediction considering imbalance problems and sparse uncertainty, Transp. Res. Part C: Emerg. Technol. 140 (2022) 103709.
[27] Shengyou Wang, Anthony Chen, Pinxi Wang, Chengxiang Zhuge, Predicting electric vehicle charging demand using a heterogeneous spatio-temporal graph convolutional network, Transp. Res. Part C: Emerg. Technol. 153 (2023) 104205.
[28] Shu Wang, Yang Yang, Yisong Chen, Xuan Zhao, Trip pricing scheme for electric vehicle sharing network with demand prediction, IEEE Trans. Intell. Transp. Syst. 23 (11) (2022) 20243–20254.
[29] Xue Xing, Bing Wang, Xin Ning, Gang Wang, Prayag Tiwari, Short-term OD flow prediction for urban rail transit control: A multi-graph spatiotemporal fusion approach, Inf. Fusion (2025) 102950.
[30] Tao Xu, Jiaming Deng, Ruolin Ma, Zixiang Zhang, Yingying Zhao, Zhilong Zhao, Juntao Zhang, Hierarchical spatio-temporal graph ODE networks for traffic forecasting, Inf. Fusion 113 (2025) 102614.
[31] Zhenghong Wang, Yi Wang, Furong Jia, Fan Zhang, Nikita Klimenko, Leye Wang, Zhengbing He, Zhou Huang, Yu Liu, Spatiotemporal fusion transformer for large-scale traffic forecasting, Inf. Fusion 107 (2024) 102293.
[32] Junyi Gao, Rakshith Sharma, Cheng Qian, Lucas M. Glass, Jeffrey Spaeder, Justin Romberg, Jimeng Sun, Cao Xiao, STAN: spatio-temporal attention network for pandemic prediction using real-world evidence, J. Am. Med. Inform. Assoc. 28 (4) (2021) 733–743.
[33] Chengqing Yu, Fei Wang, Yilun Wang, Zezhi Shao, Tao Sun, Di Yao, Yongjun Xu, MGSFformer: A multi-granularity spatiotemporal fusion transformer for air quality prediction, Inf. Fusion 113 (2025) 102607.
[34] Jinlei Zhang, Shuai Mao, Shuxin Zhang, Jiateng Yin, Lixing Yang, Ziyou Gao, EF-former for short-term passenger flow prediction during large-scale events in urban rail transit systems, Inf. Fusion 117 (2025) 102916.
[35] Bo Zhang, Hui Ma, Jian Ding, Jian Wang, Bo Xu, Hongfei Lin, Distilling implicit multimodal knowledge into large language models for zero-resource dialogue generation, Inf. Fusion (2025) 102985.
[36] Hao Xue, Flora D. Salim, Promptcast: A new prompt-based learning paradigm for time series forecasting, IEEE Trans. Knowl. Data Eng. 36 (11) (2023) 6851–6864.
[37] Haiteng Zhao, Shengchao Liu, Ma Chang, Hannan Xu, Jie Fu, Zhihong Deng, Lingpeng Kong, Qi Liu, Gimlet: A unified graph-text model for instruction-based molecule zero-shot learning, Adv. Neural Inf. Process. Syst. 36 (2023) 5850–5887.
[38] Yan Kang, Mingjian Yang, Yue Peng, Jingwen Cai, Lei Zhao, Zhan Gao, Ningshu Li, Bin Pu, LLM-DG: Leveraging large language model for enhanced disease prediction via inter-patient and intra-patient modeling, Inf. Fusion 121 (2025) 103145.
[39] Tiesunlong Shen, Erik Cambria, Jin Wang, Yi Cai, Xuejie Zhang, Insight at the right spot: Provide decisive subgraph information to graph LLM with reinforcement learning, Inf. Fusion 117 (2025) 102860.
[40] Chenxi Liu, Sun Yang, Qianxiong Xu, Zhishuai Li, Cheng Long, Ziyue Li, Rui Zhao, Spatial-temporal large language model for traffic prediction, in: 2024 25th IEEE International Conference on Mobile Data Management, MDM, IEEE, 2024, pp. 31–40.
[41] Wei Zhao, Haoran Zhang, Jianqin Zheng, Yuanhao Dai, Liqiao Huang, Wenlong Shang, Yongtu Liang, A point prediction method based automatic machine learning for day-ahead power output of multi-region photovoltaic plants, Energy 223 (2021) 120026.
[42] Zhile Yang, Tianyu Hu, Juncheng Zhu, Wenlong Shang, Yuanjun Guo, Aoife Foley, Hierarchical high-resolution load forecasting for electric vehicle charging: A deep learning approach, IEEE J. Emerg. Sel. Top. Industrial Electron. 4 (1) (2022) 118–127.
[43] Haosen Yang, Xin Shi, Linyun Xiong, Ziqiang Wang, Zipeng Liang, Robust hierarchical grouping learning immune to missing data for voltage stability assessment, IEEE Trans. Power Syst. (2024).
[44] Wen-Long Shang, Xiaoming Tao, Huibo Bi, Yanyan Chen, Hui Zhang, Washington Y. Ochieng, Audio related quality of experience evaluation in urban transportation environments with brain inspired graph learning, IEEE Trans. Intell. Transp. Syst. 24 (12) (2023) 13841–13851.
[45] Wenlong Liao, Dechang Yang, Qi Liu, Yixiong Jia, Chenxi Wang, Zhe Yang, Data-driven reactive power optimization of distribution networks via graph attention networks, J. Mod. Power Syst. Clean Energy 12 (3) (2024) 874–885.
[46] Jielun Liu, Ghim Ping Ong, Xiqun Chen, GraphSAGE-based traffic speed forecasting for segment network with sparse data, IEEE Trans. Intell. Transp. Syst. 23 (3) (2020) 1755–1766.
[47] Lijun Sun, Mingzhi Liu, Guanfeng Liu, Xiao Chen, Xu Yu, FD-TGCN: Fast and dynamic temporal graph convolution network for traffic flow prediction, Inf. Fusion 106 (2024) 102291.
[48] Chaofan Dai, Qideng Tang, Huahua Ding, TGAT: Temporal graph attention network for blockchain phishing scams detection, in: 2024 International Conference on Computer, Information and Telecommunication Systems, CITS, IEEE, 2024, pp. 1–7.