Exploiting Pre-Trained Language Model for Cross-City Urban Flow Prediction Guided by Information-Theoretic Analysis

Qiang Zhou[1,2*], **Xudong Tong**[1,2*], **Yuting Liu**[1], **Chuanxing Liu**[1], **Jingjing Gu**[1,2†]

[1]Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
[2]MIIT Key Laboratory of Pattern Analysis and Machine Intelligence, Nanjing 211106, China
{zhouqnuaacs, xudongtong, yuting_liu, lcx_defender, gujingjing}@nuaa.edu.cn

Abstract

Cross-city urban flow prediction is critical for democratizing smart application benefits in data-scarce developing cities. However, existing methods face an inherent performance ceiling, constrained by both the inevitably finite samples from the source city and the distributional gap between cities. In this paper, we present PLM-CUP, the first theoretically-grounded framework that breaks this bottleneck by leveraging a pre-trained language model (PLM) as an additional source domain. Through an information-theoretic analysis of the generalization error bound, we reveal that the key challenge lies in constructing a semantic bridge encoder and a task-specific adapter to enable cross-domain alignment when incorporating a PLM. Accordingly, PLM-CUP adopts a three-stage architecture, including a semantic bridge encoder that transforms spatiotemporal flow patterns into language-aligned representations via trend-periodicity decomposition, a PLM fine-tuned for knowledge transfer, and a task adapter with spatiotemporal self-attention to conduct multi-step prediction. We further introduce GDAConv, a graph convolution module with dual activation functions that enhances spatial modeling throughout the framework. Experiments on real-world datasets demonstrate that PLM-CUP significantly outperforms state-of-the-art baselines, validating the effectiveness of the proposed PLM enhanced cross-city transfer paradigm for urban flow prediction.

Code — https://github.com/SINCOSLab/PLM-CUP

Introduction

As the popularization of smart urbanization, urban flow prediction has become a hot spot for optimizing resource allocation and management of cities (Zhang et al. 2024). With various flow data obtained from different transportation modes in many developed cities, machine learning based methods (Gao et al. 2025; Kong, Guo, and Liu 2024) have achieved advanced performance in the task of urban flow prediction. To democratize the benefits of smart urbanization for a vast number of developing cities and regions, achieving accurate urban flow prediction remains a critical prerequisite for enabling various smart city applications. Consequently, significant research efforts (Zhang et al. 2025b; Wang, Lin, and Li 2025) sought to effectively leverage mobility data from data-rich urban centers to empower data-scarce regions with accurate modeling, which have led to wide attention on cross-city urban flow prediction. Recently, researches in cross-city urban flow prediction primarily confronts the significant domain shift between source and target cities. For example, (Shi, Zhou, and Gu 2024; Chen et al. 2022) refines the city-level task to regional urban flow prediction, transferring knowledge by matching regions with similar flow dynamics to mitigate negative transfer. (Zhang et al. 2025b) employs a one-stage learning framework, training a spatiotemporal forecasting model on both domains while strategically filtering gradients to retain only beneficial information from the source city. From an information-theoretic perspective, these state-of-the-art methods operate by extracting a matched subset of the source distribution to minimize the domain discrepancy. This paradigm, however, is self-limiting, since performance is fundamentally capped by the inherent distributional gap and remains constrained by the number of samples within the source domain. Consequently, a performance bottleneck is reached nowadays the distributional gap is minimized. Extending this paradigm, (Yang et al. 2025) suggests that it is worth introducing multiple source cities for cross-city transfer learning, which inspired us to boldly regard other sequential data as a source domain. As a type of sequential data, natural language has been well-studied through Pre-trained Language Models (PLM) (Li et al. 2024a). Can we leverage natural language as an additional source domain for cross-city urban flow prediction? ST-LLM (Liu et al. 2024) lent support to this idea by introducing PLM for urban traffic flow prediction. The experimental improvement of ST-LLM+ (Liu et al. 2025) even proved the potential ability of PLM for few and zero shot urban flow prediction. However, these studies only draw the empirical inference from experimental results. To date, the application of PLM to cross-city urban flow prediction lacks clear theoretical guidance. Therefore, how to fully exploit the generalization capabilities of PLM to realize efficient cross-city transfer learning remains an open and interesting problem. To this end, we derive the generalization error bound (GEB) based on information-theoretic analysis in this paper. The result shows that when treating large-

*These authors contributed equally.
†Corresponding author.

scale corpora as a new source domain, the main term of GEB stems from the distribution shift between the new source domain and the target city. Consequently, It is indicated that further improvements in fine-tuning are inefficient, which yield sharply diminishing returns on the mutual-information components in GEB. In contrast, we should highlight the importance of training a semantic bridge encoder and a task-specific adapter to align cross-domain distributions. Accordingly, we develop the first simple but effective transfer paradigm that leverages PLM for Cross-city Urban flow Prediction (PLM-CUP), which adopts a three-stage architecture comprising a semantic bridge encoder, a fine-tuned PLM, and a task-specific adapter. To enhance spatial generalization, we introduce GDAConv, a novel graph convolutional module with dual activation functions, which is extensively employed across the whole workflow. Within the semantic bridge encoder, we extract and fuse spatiotemporal and feature-specific embeddings, which are then fused and processed via trend and period semantic decomposition modules to produce semantical inputs for the PLM. The task adapter further decodes the PLM-derived representation through spatiotemporal self-attention module and a predictor to generate multi-step prediction of urban flow. During training, PLM-CUP is first pre-trained on the source city to establish distributional alignment between language and urban flow representation roughly, and subsequently fine-tuned on target city for the cross-city transfer learning. In summary, our work makes the following contributions:

- *Theoretically,* guided by information-theoretic insights into the generalization error bound, we introduce the first cross-city transfer paradigm that leverages the pre-trained language model for urban flow prediction.
- *Methodologically,* we instantiate the paradigm as PLM-CUP, a novel cross-city urban flow prediction model, which adopts a three-stage architecture comprising a semantic bridge encoder, a fine-tuned PLM, and a task-specific adapter for cross-domain distributional alignment by modeling spatiotemporal dependencies.
- *Empirically,* extensive experiments conducted on three real-world urban flow datasets demonstrate that PLM-CUP achieves superior prediction performance compared to other state-of-the-art models.

Related Work

Urban Flow Prediction. Urban flow prediction have been paid much attention to. Early methods employed traditional models (Fu, Zhang, and Li 2016), while recent approaches leverage deep neural networks to capture complex spatiotemporal dynamics (Cao et al. 2025; Kong, Guo, and Liu 2024). More recently, pre-trained language models (Liu et al. 2024) and urban foundation models (Li et al. 2024b) have been introduced to improve generalization. However, these methods typically require substantial training data, and few-shot urban flow prediction with PLM (Liu et al. 2025) remains underexplored. **Cross-city Transfer Learning.** Cross-city transfer learning has been a primary approach to few-shot urban flow prediction, where models are typically pre-trained on a source city and adapted to a target city with limited supervision. Early studies aimed to construct transferable spatiotemporal frameworks with deep learning (Wang et al. 2019), attentive adaptation network (Wang et al. 2022), and meta-learning (Zhang et al. 2022) to bridge domain discrepancies. More recent efforts emphasize proper knowledge matching (Shi, Zhou, and Gu 2024) and selective transfer (Zhang et al. 2025b) to enhance source information utilization while suppressing negative transfer. Despite these advances, this paradigm remains inherently constrained by the distributional gap between cities, leading to a performance bottleneck even when domain alignment is optimized. However, extending the source modality to natural language promises richer knowledge transfer, yet the resulting semantic gap poses new challenges that remain unstudied.

Preliminaries

In this section, we formally define basic concepts and the cross-city urban flow prediction problem.

Definition 1 (Urban Flow) *We define the urban flow as $x_t \in \mathbb{R}^{N \times D}$ for N nodes (i.e. regions) in a city at time step t, where D is the number of features. Each element in x_t represents the crowd or traffic volume at time t in region $n \in [1, N]$. The unit of time t is flexible regarding the granularity of prediction, e.g., 30 minutes.*

Definition 2 (Target & Source City) *The target city has only a small amount of urban flow data, denoted as $\mathcal{D}^{\mathcal{T}}$, where we expect to train an urban flow prediction model. In contrast, the source city is relatively rich in urban flow records, denoted as $\mathcal{D}^{\mathcal{S}}$. Then, we denote the datasets of two cities as $S^{\mathcal{T}} = \{S_1^{\mathcal{T}}, \cdots, S_{n_1}^{\mathcal{T}}\}$ and $S^{\mathcal{S}} = \{S_1^{\mathcal{S}}, \cdots, S_{n_2}^{\mathcal{S}}\}$, where $n_2 > n_1$ in this scenario. Note that we assume both source and target cities demonstrate common flow patterns in their normalized urban flow data, such as morning and evening peaks, which serve as transferable information.*

Definition 3 (Natural Language Domain) *We use natural language as an auxiliary source domain, denoted as $\mathcal{D}^{\mathcal{L}}$. Although the samples are neither accessible nor directly used, for notational clarity, we denote the natural language dataset as $S^{\mathcal{L}} = \{S_1^{\mathcal{L}}, \cdots, S_{n_3}^{\mathcal{L}}\}$, where $n_3 \gg n_2$.*

Definition 4 (Pre-trained Language Model) *We utilize a language model h, pre-trained on $\mathcal{D}^{\mathcal{L}}$, as a component for constructing the cross-city urban flow prediction model.*

Problem 1 (Cross-city Urban Flow Prediction) *Given T time steps of urban flow and distance-based weighted matrix A_d of the source or target city as widely defined in (Yu, Yin, and Zhu 2018; Li et al. 2018), we aim to learn a function $\mathcal{F}(\cdot, \omega)$ to predict urban flow for the next T' time steps:*

$$\left[X_{(t-T+1):t}, A_d\right] \xrightarrow{\mathcal{F}(\cdot,\omega)} \left[Y_{(t+1):(t+T')}\right], \tag{1}$$

where $[X_{(t-T+1):t}, Y_{(t+1):(t+T')}] = S_i^{\mathcal{T}}$ is a sample in the target city dataset, $\omega \in \mathcal{W}$ is a hypothesis from the hypothesis space corresponding to $\mathcal{F}$. Differ from the traditional urban flow prediction, the initialization of ω is adapted from $\mathcal{D}^{\mathcal{S}}$ and $\mathcal{D}^{\mathcal{L}}$ in the transfer paradigm.

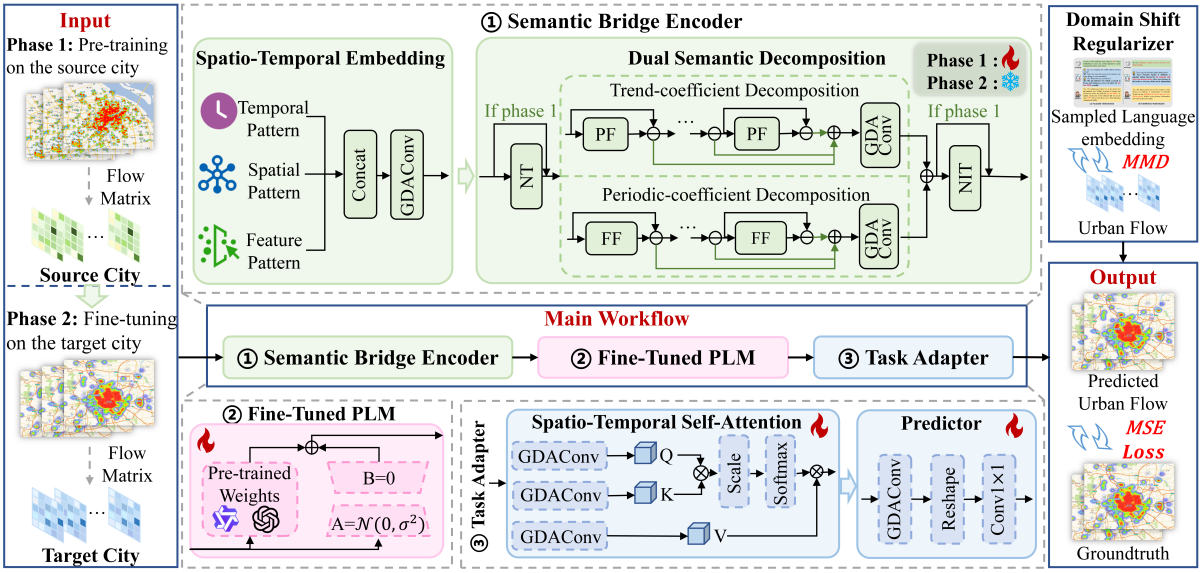

Figure 1: **Framework overview.** The main workflow of PLM-CUP is trained in two phases with a theoretically guided loss. PLM-CUP comprises three components. The encoder performs a dual semantic decomposition over the aggregation embeddings of urban flow to extract trend and periodic semantics, thereby enhancing semantic alignment with natural language. A PLM is fine-tuned with LoRA throughout the two training phases. The task adapter models spatiotemporal features from the semantic representations through self-attention and generates urban flow prediction via a predictor.

Methodology

In this section, we first derive the generalization error bound regarding Problem 1 in the transfer paradigm. Guided by the derivation, the proposed multi-source pre-trained transfer learning model PLM-CUP for cross-city urban flow prediction will be detailed.

Information-theoretic Analysis

First, we examine the property of loss functions by referring to the sub-Gaussianity introduced in (Wu et al. 2020).

Lemma 1 (Sub-Gaussianity of a loss function) *Regarding an instance space $\mathcal{S}$ and a hypothesis space $\mathcal{W}$, a non-negative loss function ℓ is $\mathcal{W}\times\mathcal{S}\mapsto\mathbb{R}^+$. For any hypothesis ω from the hypothesis space $\mathcal{W}$, if the loss function $\ell(\omega,\mathcal{S})$ takes value in $[a,b]$, then $\ell(\omega,\mathcal{S})$ is $\frac{(b-a)^2}{4}$-subgaussian.*

For Problem 1, standard practice involves normalizing both input and output flow values during training, with predictions further clipped to suppress outliers. This implies the existence of an upper bound B such that both groundtruth and predicted flows lie in $[0,B]$. Under this assumption, the commonly used mean-square error loss $\ell(\omega,\mathcal{S})=(\mathcal{F}(X,\omega)-Y)^2$ is $\frac{B^2}{4}$-subgaussian. Let us consider the cross-city transfer learning with a target city and a source city first. The empirical risk induced by $S^{\mathcal{T}}$ and $S^{\mathcal{S}}$ (Ben-David et al. 2010) can be defined as

$$\hat{L}_\alpha(\omega):=\frac{\alpha}{n_1}\sum_{i=1}^{n_1}\ell(\omega,S_i^{\mathcal{T}})+\frac{1-\alpha}{n_2}\sum_{i=1}^{n_2}\ell(\omega,S_i^{\mathcal{S}}),\quad(2)$$

where the weight $\alpha\in[0,1]$ is to be determined. Then, we define $W_{\mathsf{ERM}}=\operatorname{argmin}_\omega\hat{L}_\alpha(\omega)$ as the empirical risk minimization (ERM) solution trained by one ERM algorithm. Also, we denote μ' as the probability distribution defined on samples in $S^{\mathcal{T}}$. The generalization error can be defined as

$$\mathrm{gen}(W_{\mathsf{ERM}})=\mathbb{E}_{S_i\sim\mu'}\{\ell(W_{\mathsf{ERM}},\mathcal{S})\}-\hat{L}_\alpha(W_{\mathsf{ERM}}).\quad(3)$$

With the probability distribution μ_1 defined on samples in $S^{\mathcal{S}}$, the generalization error bound for subgaussian loss functions is derived (Wu et al. 2020).

Theorem 1 (Generalization error for subgaussian loss functions in cross-city transfer learning) *As $\ell(\omega,\mathcal{S})$ is r^2-subgaussian under the distribution $P_\omega\otimes\mu'$, the generalization error expectation of the ERM solution is bounded as*

$$\begin{aligned}|\mathbb{E}_{\omega S^{\mathcal{T}}S^{\mathcal{S}}}\{\mathrm{gen}(W_{\mathsf{ERM}})\}|&\le\frac{\alpha\sqrt{2r^2}}{n_1}\sum_{i=1}^{n_1}\sqrt{I(W_{\mathsf{ERM}};S_i^{\mathcal{T}})}\\&+\frac{(1-\alpha)\sqrt{2r^2}}{n_2}\sum_{i=1}^{n_2}\sqrt{I(W_{\mathsf{ERM}};S_i^{\mathcal{S}})+D(\mu_1||\mu')}.\end{aligned}\quad(4)$$

If we adopt the former transfer paradigm, i.e., $\mathcal{F}$ is a fine-tuned PLM h, the generalization error expectation which obeys additivity principle can be extended to incorporating another source domain $\mathcal{D}^{\mathcal{L}}$. Assuming μ_2 as the probability distribution defined on samples in $S^{\mathcal{L}}$, the generalization error expectation is bounded as

$$\begin{aligned}|\mathbb{E}_{\omega S^{\mathcal{T}}S^{\mathcal{S}}S^{\mathcal{L}}}\{\mathrm{gen}(W_{\mathsf{ERM}})\}|&\le\frac{\alpha\sqrt{2r^2}}{n_1}\sum_{i=1}^{n_1}\sqrt{I(W_{\mathsf{ERM}};S_i^{\mathcal{T}})}\\&+\frac{(1-\alpha-\beta)\sqrt{2r^2}}{n_2}\sum_{i=1}^{n_2}\sqrt{I(W_{\mathsf{ERM}};S_i^{\mathcal{S}})+D(\mu_1||\mu')}\\&+\frac{\beta\sqrt{2r^2}}{n_3}\sum_{i=1}^{n_3}\sqrt{I(W_{\mathsf{ERM}};S_i^{\mathcal{L}})+D(\mu_2||\mu')},\end{aligned}\quad(5)$$

where $I(W_{\mathsf{ERM}}; S_i)$ denotes the mutual information (MI) between the hypothesis W_{ERM} and a sample S_i, and $D(\mu_1\|\mu')$, $D(\mu_2\|\mu')$ represent KL-divergence terms capturing domain shift, with β a balancing weight. In Eq. (5), MI terms decrease as sample sizes increase with the introduction of $S^{\mathcal{L}}$, since larger datasets dilute individual sample impact and weaken statistical dependency on the hypothesis. It is noteworthy that each term $I(W_{\mathsf{ERM}}; S_i^{\mathcal{L}})$ approaches zero due to large n_3. In contrast, the KL-divergence terms remain non-vanishing as n_1, n_2 and n_3 go to infinity. Hence, reducing domain shift becomes central to improving performance. Prior efforts mostly focus on minimizing $D(\mu_1\|\mu')$, such as (Chen et al. 2022) using region-level prediction to align transferable knowledge, and (Zhang et al. 2025b) filtering gradients for beneficial source signals. However, the KL-divergence $D(\mu_2||\mu')$, which reflects the domain shift between urban flow data and natural language, is considerably large. Worse still, due to inaccessible PLM corpora, adjusting μ_2 is infeasible, posing a fundamental barrier to effectively leveraging PLMs in cross-city urban flow prediction. To tackle this problem, we propose a novel and simple cross-city transfer paradigm in which $\mathcal{F}$ is no longer a merely fine-tuned PLM. Instead, a semantic encoder g_1 and a task-specific adapter g_2 are introduced to align the cross-domain distributions, forming $\mathcal{F}(\cdot) := g_2(h(g_1(\cdot)))$. In our formulation, μ' and μ_2 correspond to the joint distribution $\mu' := P_{X^{\mathcal{T}},Y^{\mathcal{T}}}$ and $\mu_2 := P_{X^{\mathcal{L}},Y^{\mathcal{L}}}$, respectively. Consequently, the KL-divergence $D(\mu_2||\mu')$ can be replaced by

$$D[(g_1^{-1} \times g_2)_{\#P_{X^{\mathcal{L}},Y^{\mathcal{L}}}}||P_{X^{\mathcal{T}},Y^{\mathcal{T}}}], \tag{6}$$

where $(g_1^{-1} \times g_2)_{\#P}$ denotes the push-forward measure of the PLM-side distribution $P_{X^{\mathcal{L}},Y^{\mathcal{L}}}$ under the transformation $g_1^{-1} \times g_2$. The divergence can be further converted without ambiguity for bypassing the inversion of g_1 in the generalization error expectation as

$$\begin{aligned}|\mathbb{E}_{\omega S^{\mathcal{T}} S^{\mathcal{S}} S^{\mathcal{L}}}\{\mathrm{gen}(W_{\mathsf{ERM}})\}| \leq \frac{\alpha\sqrt{2r^2}}{n_1}\sum_{i=1}^{n_1}\sqrt{I(W_{\mathsf{ERM}}; S_i^{\mathcal{T}})} \\ + \frac{(1-\alpha-\beta)\sqrt{2r^2}}{n_2}\sum_{i=1}^{n_2}\sqrt{I(W_{\mathsf{ERM}}; S_i^{\mathcal{S}}) + D(\mu_1||\mu')} \\ + \frac{\beta\sqrt{2r^2}}{n_3}\sum_{i=1}^{n_3}\sqrt{\sigma + D[(\mathrm{I}\times g_2)_{\#P_{X^{\mathcal{L}},Y^{\mathcal{L}}}}||(g_1\times \mathrm{I})_{\#P_{X^{\mathcal{T}},Y^{\mathcal{T}}}}]},\end{aligned} \tag{7}$$

where I is the identity transformation function, σ is a vanishingly small positive constant and $\sigma \geq I(W_{\mathsf{ERM}}; S_i^{\mathcal{L}})$. Based on this paradigm, the KL-divergence term is diminishable now through constructing the good g_1 and g_2. Moreover, the KL-divergence term can be incorporated as a regularizer to explicitly constrain the mappings of the semantic encoder g_1 and task-specific adapter g_2, thereby tightening the generalization error bound.

Cross-city Transfer Learning Model with PLM

Inspired by the information-theoretic analysis, we introduce the PLM-CUP framework with a semantic encoder and a task-specific adapter, as depicted in Fig. 1. For simplicity, we use z_{in} and z_{out} as the input and output of each component in this section.

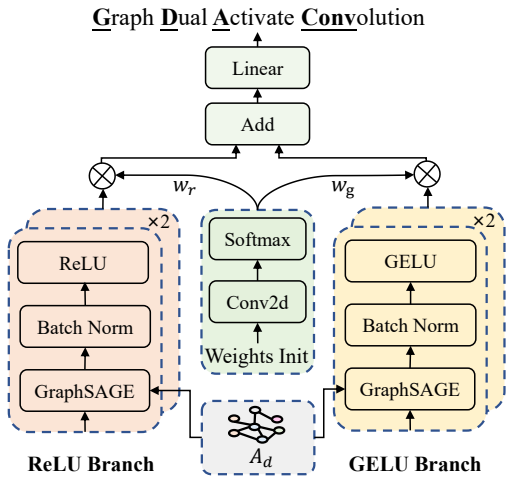

Figure 2: Graph dual activate convolution module.

Spatial Representation Module. Modeling spatial correlations is vital for urban flow prediction (Li et al. 2024b), as mobility in a region is often influenced by its neighbors. In cross-city settings with limited target city data, aggregating information from related areas becomes more critical. Thus, we propose a novel graph convolution module with dual activation functions, termed GDAConv, extensively employed in Fig. 1 to capture complex spatial dependencies. As shown in Fig. 2, GDAConv has two branches modeling distinct spatial patterns. The ReLU (Huang et al. 2020) branch effectively captures linear and simple nonlinear patterns due to its sparsity, while the GELU (Hendrycks and Gimpel 2016; Zheng, Yao, and Zhang 2025) branch captures complex and rare features through its smooth and probabilistic formulation. The process is formulated as

$$z_r = \mathrm{ReLU}(\mathrm{BN}(\mathrm{GraphSAGE}(z_{\text{in}}, A_d, \psi))) \tag{8}$$

$$z_g = \mathrm{GELU}(\mathrm{BN}(\mathrm{GraphSAGE}(z_{\text{in}}, A_d, \psi))) \tag{9}$$

$$z_{\text{out}} = W \cdot (w_r * z_r + w_g * z_g) \tag{10}$$

where BN is a batch normalization layer. We adopt GraphSAGE (Hamilton, Ying, and Leskovec 2017) for inductive spatial dependencies learning, since the graph architecture will change in the training phases, where ψ is the sampling iteration. Outputs from two-layer ReLU and two-layer GELU branches are fused by a weighted sum and then linearly transformed. The two weights w_r and w_g with shape of $(N \times 1)$ are derived from initialized learnable parameters.

Semantic Bridge Encoder. Inspired by (Liu et al. 2024), we treat timesteps at each urban region as the tokens, i.e., $x \in \mathbb{R}^{N\times(T\times D)}$. Multiple embeddings encode feature, spatial, and temporal patterns into a unified representation. The feature embedding is produced via a GDAConv layer. For spatial encoding, we apply GraphSAGE to learnable parameters, ensuring spatial patterns are explicitly unique. Temporal embedding encodes hour-of-day and day-of-week via two learnable matrices to model urban flow dynamics. These embeddings are concatenated along the timestep axis and passed through another GDAConv layer to yield the final

representation $z_{\text{in}} \in \mathbb{R}^{N \times d}$, where d is the latent dimension. It is then processed by the dual semantic decomposition module, inspired by and extending (Cao et al. 2025; Godfrey and Gashler 2017). The module includes two Neural Decomposition (ND) branches: a trend-coefficient branch using polynomial basis functions to extract multi-scale trends, and a periodic-coefficient branch using Fourier basis functions to isolate cyclic patterns (e.g., commuting). The polynomial basis matrix is constructed by raising a normalized vector $\phi_t \in \mathbb{R}^{1 \times d}$ to powers defined by $\lambda \in \mathbb{R}^{d_{\text{ND}} \times 1}$, where d_{ND} is the ND latent dimension. The Fourier basis matrix is formed by concatenating sine and cosine functions applied to position encodings ϕ_p across multiple harmonics k (Godfrey and Gashler 2017). The coefficients for both branches are learned from the residuals of z_{in}, refined through several feed-forward layers. Formally,

$$p_i = \text{FFN}(z_{\text{in}} - \sum_{m=1}^{i-1} p_m) \cdot \phi_t^{\lambda}, \tag{11}$$

$$q_i = \text{FFN}(z_{\text{in}} - \sum_{m=1}^{i-1} q_m) \cdot [(\sin(\text{k} \cdot \phi_{\text{p}}) | \cos(\text{k} \cdot \phi_{\text{p}})]. \tag{12}$$

where $[\cdot, \cdot]$ means the concatenation operator along the last dimension. Consequently, after υ iterations of ND, the decomposed parts at every iteration are concatenated with z_{in} and passed through GDAConv in each branch. The outputs are then concatenated to form the z_{out} formulated as

$$z_{\text{out}} = [\text{GDAConv}([z_{\text{in}} | q_{1 \sim \upsilon}]) | \text{GDAConv}([z_{\text{in}} | q_{1 \sim \upsilon}])], \tag{13}$$

where z_{out} encodes rich urban flow semantics, including spatial correlations, trends, and periodic patterns, for alignment with natural language semantics, and is fed into a LoRA-tuned PLM during training.

Task-specific Adapter. Beyond the semantic representations of natural language, it is believed that the bottleneck between latent representations and the urban flow prediction task lies in decoding the complex spatiotemporal flow patterns (Li et al. 2025; Cao et al. 2025). To bridge the semantic gap between the PLM representations and the task, we design an adapter that enhances the transformation of semantic representations into urban flow aware ones, which is composed of a multi-head spatiotemporal self-attention module and a predictor for urban flow prediction. Specifically, the self-attention module leverages GDAConv layers for generating Q, K, V. Formally, given the semantic representation $z_{\text{in}} \in \mathbb{R}^{N \times d}$, the transformation is simply defined as

$$z_{\text{out}} = \text{Self\_Attention}(\text{GDAConv}(z_{\text{in}})), \tag{14}$$

where $z_{\text{out}} \in \mathbb{R}^{N \times (T' \times d)}$, and T' denotes the number of time steps to be predicted in Problem 1. In the predictor, z_{out} is reshaped and passed through another GDAConv layer followe123 d by a 1×1 convolutional predictor formulated as

$$Y = \text{Conv}_{1 \times 1}(\text{GDAConv}(z_{\text{out}}).\text{reshape}(N, T', d)) \tag{15}$$

By adapting PLM-extracted representations to the target task space through spatiotemporal self-attention, the adapter serves as a key enabler of effective semantic transfer from semantic representations to the urban flow domain, improving the performance of cross-city urban flow prediction.

Transfer and Optimize. In the cross-city setting, we adopt a standard fine-tuning-based transfer learning approach. As our contribution lies beyond the transfer strategy itself, recent methods can be seamlessly integrated into our PLM-based paradigm. Specifically, Phase 1 pre-trains the entire PLM-CUP workflow on the source city, followed by fine-tuning on the target city in Phase 2. We observe that fine-tuning the dual semantic decomposition degrades performance, as it already learns effective decomposition patterns from the source city. Thus, we freeze its parameters in Phase 2 and introduce Node Transform (NT) layers around the module to project into the source domain for decomposition and map them back, ensuring semantic consistency. Guided by our information-theoretic analysis, we treat the KL-divergence term in Eq. (7) as a regularizer to constrain the encoder and adapter mappings. Since the real sample distributions are inaccessible, we approximate this term using Maximum Mean Discrepancy (MMD). The resulting loss ℓ is defined as:

$$\ell = (\mathcal{F}(X) - Y)^2 + \kappa \text{MMD}(\mathbf{Z}^{\mathcal{L}}, \mathbf{Z}^{\mathcal{T}}), \tag{16}$$

where κ is the balancing factor between the two terms. To approximate the natural language dataset $S^{\mathcal{L}}$, we obtain a natural language dataset $S^{\mathcal{L}}_{\text{appr}}$ by inputting dialogue samples, randomly sampled from another large language model, into the PLM. Then, $\mathbf{Z}^{\mathcal{L}}$ can be derived by passing Y in $S^{\mathcal{L}}_{\text{appr}}$ to the task-specific adapter. Meanwhile, $\mathbf{Z}^{\mathcal{T}}$ is obtained from a batch of target-city data by encoding its inputs X via the semantic bridge encoder.

Experiments

Experimental Setup

Datasets. We evaluate our method on three spatiotemporal traffic datasets collected from Nanjing (NJ), Shanghai (SH), and Haikou (HK). These datasets from different urban areas with varying scales and characteristics, providing a comprehensive evaluation of our proposed cross-city urban flow prediction framework. The time span of each flow dataset: 02/01/2018-03/27/2018 in Shanghai with 8,779,200 records, 12/01/2018-01/31/2019 in Nanjing with 12,371,062 records, and 05/01/2017-10/31/2017 in Haikou with 13,160,170 records. Each urban area is divided into 15×15 rasterized regions.

Baselines. We compare PLM-CUP with 13 baseline approaches of three categories: statistical methods, typical deep-learning methods, and transfer-learning methods:

- **Statistical Methods**: Historical Averages(HA).
- **Deep-learning Methods**: ASTGCN (Guo et al. 2019), AGCRN (BAI et al. 2020), ST-LLM (Liu et al. 2024), and ST-LLM-Plus (Liu et al. 2025) are spatiotemporal modeling SOTAs designed for the single-city scene.
- **Transfer-learning Methods**: StepDeep (Shen et al. 2018), ConvLSTM (SHI et al. 2015), RegionTrans (Wang et al. 2019), ST-DAAN (Wang et al. 2022), CCMHC (Chen et al. 2022), CrossGNN (Huang et al. 2023), CrapEpic (Shi, Zhou, and Gu 2024) and TSJT(STGCN) (Zhang et al. 2025b) are suitable or well-designed for cross-city urban flow prediction.

	Model	MAE		RMSE		MAE		RMSE		MAE		RMSE	
		SH->NJ	HK->NJ	SH->NJ	HK->NJ	NJ->SH	HK->SH	NJ->SH	HK->SH	NJ->HK	SH->HK	NJ->HK	SH->HK
Target Only	HA	46.423		71.427		17.421		24.815		7.713		11.274	
	StepDeep	40.894		64.879		15.684		20.074		6.497		10.31	
	ConvLSTM	39.341		63.143		15.237		19.968		6.581		10.47	
	ASTGCN	30.809		54.801		10.541		17.985		4.483		8.124	
	AGCRN	25.366		45.494		9.863		16.672		3.94		8.017	
	CrossGNN	32.055		57.764		11.727		19.709		5.013		9.44	
	ST-LLM	24.417		43.498		9.154		15.107		4.013		7.874	
	ST-LLM-Plus	23.876		42.642		8.967		14.7		3.872		6.951	
	TSJT(STGCN)	23.217		41.623		9.143		15.263		4.013		6.862	
	PLM-CUP-TG	23.315		41.682		8.734		14.078		3.743		6.578	
	PLM-CUP-TQ	22.879		40.348		8.183		13.125		3.729		6.595	
Transfer	StepDeep	35.419	36.974	58.429	57.817	13.856	14.191	18.084	17.93	5.903	5.786	9.561	9.632
	ConvLSTM	34.797	36.213	59.147	60.418	13.342	13.852	18.424	18.59	5.873	5.651	9.614	9.368
	RegionTrans	29.47	30.13	48.719	48.423	11.125	11.406	16.141	16.249	4.713	4.591	9.078	8.99
	ST-DAAN	28.543	28.014	47.374	46.427	10.243	10.651	15.691	15.912	4.362	4.183	8.603	8.483
	CCMHC	26.428	27.928	46.913	46.289	9.785	10.205	15.558	15.629	4.231	4.287	8.418	8.513
	CrossGNN	27.439	28.141	47.048	47.013	9.934	9.873	15.614	15.323	4.413	4.505	8.812	8.703
	CrapEpic	25.44	26.138	43.227	41.479	9.313	9.178	15.342	15.233	4.128	4.093	8.18	8.093
	TSJT(STGCN)	22.532	22.417	39.89	39.416	8.869	8.903	14.083	13.745	3.864	3.901	6.643	6.518
	PLM-CUP-G	**21.913**	22.346	**39.215**	39.342	8.272	8.449	13.085	13.411	3.688	3.7	6.559	6.589
	PLM-CUP-Q	22.017	**22.124**	39.442	**39.145**	**7.761**	**7.809**	**12.552**	**12.494**	**3.619**	**3.601**	**6.451**	**6.423**

	Model	MAPE		WMAPE		MAPE		WMAPE		MAPE		WMAPE	
		SH->NJ	HK->NJ	SH->NJ	HK->NJ	NJ->SH	HK->SH	NJ->SH	HK->SH	NJ->HK	SH->HK	NJ->HK	SH->HK
Target Only	HA	0.793		0.327		0.811		0.354		0.683		0.32	
	StepDeep	0.597		0.289		0.617		0.314		0.591		0.287	
	ConvLSTM	0.584		0.274		0.585		0.292		0.603		0.292	
	ASTGCN	0.577		0.238		0.481		0.254		0.407		0.218	
	AGCRN	0.442		0.196		0.457		0.238		0.394		0.214	
	CrossGNN	0.585		0.248		0.554		0.282		0.458		0.237	
	ST-LLM	0.445		0.208		0.463		0.227		0.378		0.193	
	ST-LLM-Plus	0.413		0.184		0.441		0.216		0.372		0.182	
	TSJT(STGCN)	0.393		0.18		0.473		0.225		0.378		0.19	
	PLM-CUP-TG	0.377		0.18		0.441		0.211		0.363		0.176	
	PLM-CUP-TQ	0.362		0.176		0.422		0.197		0.355		0.176	
Transfer	StepDeep	0.527	0.551	0.254	0.266	0.521	0.547	0.273	0.281	0.473	0.498	0.263	0.259
	ConvLSTM	0.533	0.526	0.261	0.258	0.515	0.492	0.282	0.275	0.469	0.485	0.266	0.262
	RegionTrans	0.448	0.46	0.237	0.224	0.466	0.454	0.253	0.246	0.421	0.43	0.238	0.236
	ST-DAAN	0.422	0.453	0.231	0.223	0.468	0.461	0.249	0.252	0.409	0.415	0.235	0.242
	CCMHC	0.427	0.435	0.214	0.208	0.473	0.472	0.234	0.229	0.42	0.418	0.218	0.215
	CrossGNN	0.431	0.455	0.223	0.219	0.481	0.453	0.253	0.235	0.413	0.405	0.221	0.218
	CrapEpic	0.434	0.445	0.201	0.197	0.475	0.463	0.223	0.231	0.382	0.397	0.211	0.209
	TSJT(STGCN)	0.384	0.391	0.173	0.177	0.426	0.433	0.212	0.209	0.366	0.371	0.178	0.183
	PLM-CUP-G	0.376	0.374	**0.169**	0.172	0.409	**0.406**	0.207	0.203	**0.34**	0.35	0.173	0.174
	PLM-CUP-Q	**0.358**	**0.361**	0.172	**0.167**	**0.392**	0.408	**0.186**	**0.188**	0.346	**0.343**	**0.17**	**0.171**

Table 1: Performance comparison overall. The best results are in bold, and the second-best results are underlined.

- **PLM-CUP variants**: PLM-CUP-G and PLM-CUP-Q leverage GPT-2 (Radford et al. 2019) and Qwen3-0.6B (Zhang et al. 2025a) as the PLM backbone, respectively. PLM-CUP-TG and PLM-CUP-TQ are variants trained only on the target city.

Implementation. We use $T = 6$ historical time steps to predict the next $T' = 1$ time step. All datasets use 1-hour intervals. PLM-CUP is trained on an A100 Tesla GPU with 80GB memory. The learning rate is set to 0.0005, the regularization parameter κ in Eq. (16) is set to 0.01, and the number of decomposition modules m is set to 3. All determined through grid search. The batch size is set to 64. Performances are reported via the averaged scores of 10 runs in terms of MAE, RMSE, MAPE, and WMAPE.

Performance Comparison

The performance comparison is presented in Table 1. PLM-CUP demonstrates consistently superior performance across all evaluated scenarios. In particular, PLM-CUP-Q secures top results in 18 out of 24 metrics. For example, PLM-CUP-G achieves an MAE of 21.913 on SH→NJ, marking a 2.7% improvement over TSJT, while PLM-CUP-Q attains 7.761 on NJ→SH, reducing error by 12.5%. Compared to existing transfer learning approaches, although TSJT (based on STGCN) yields strong baseline results, PLM-CUP consistently outperforms all such methods by leveraging the representational power of pre-trained language models. The advantage is especially pronounced in HK→SH, where PLM-CUP-Q improves upon TSJT by 12.3%. Traditional deep learning baselines, such as ST-LLM and ST-LLM-Plus, perform worse due to their lack of theoretical cross-domain alignment; PLM-CUP-Q surpasses ST-LLM-Plus by an average of 4.2% in MAE, validating the efficacy of our semantic bridging encoder and task adapter. Furthermore, PLM-CUP achieves stable improvements across both geographically close (e.g., SH and NJ) and distant (e.g., HK to NJ/SH) domain pairs, underscoring the robustness of our information-theoretic design.

Ablation Study

Component Analysis To investigate the contribution of each component in PLM-CUP, we conduct ablation studies of PLM-CUP-Q on the Shanghai dataset by systematically removing key modules to evaluate their impact. The results

Ablation Model	MAE			RMSE			MAPE			WMAPE		
	Target Only	NJ->SH	HK->SH	Target Only	NJ->SH	HK->SH	Target Only	NJ->SH	HK->SH	Target Only	NJ->SH	HK->SH
PLM-CUP-Q w/o TCB	8.574	8.223	8.227	13.792	13.263	13.188	0.442	0.413	0.414	0.206	0.193	0.194
PLM-CUP-Q w/o PCB	8.512	8.073	8.081	13.853	13.458	13.394	0.44	0.414	0.411	0.205	0.195	0.199
PLM-CUP-Q w/o SBE	8.606	8.215	8.228	13.903	13.503	13.542	0.451	0.42	0.423	0.208	0.198	0.204
PLM-CUP-Q w/o PLM	9.582	9.033	9.131	16.13	15.441	15.428	0.47	0.428	0.424	0.231	0.214	0.221
PLM-CUP-Q w/o TA	8.382	8.013	7.978	13.6	13.218	13.297	0.418	0.391	0.403	0.202	0.198	0.189
PLM-CUP-Q w/o GDA-A	8.618	8.226	8.2	13.87	13.458	13.343	0.434	0.413	0.415	0.223	0.212	0.214
PLM-CUP-Q w/o GDA-S	8.484	8.111	8.17	13.527	13.134	13.251	0.427	0.408	0.407	0.211	0.192	0.199
PLM-CUP-Q w/o GDA-T	8.535	8.231	8.269	13.8	13.484	13.492	0.429	0.417	0.422	0.208	0.183	0.21
PLM-CUP-Q w/o GDA-R	8.456	7.979	7.987	13.529	13.047	13.218	0.431	0.413	0.415	0.204	0.191	0.195
PLM-CUP-Q	8.183	7.761	7.809	13.125	12.552	12.494	0.422	0.392	0.408	0.197	0.186	0.188

Table 2: Ablation study of PLM-CUP components on Shanghai (SH) as target city.

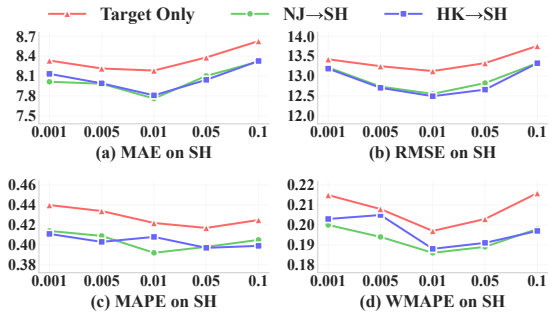

Figure 3: Performance comparison with different MMD regularization coefficients κ on Shanghai (SH) as target city.

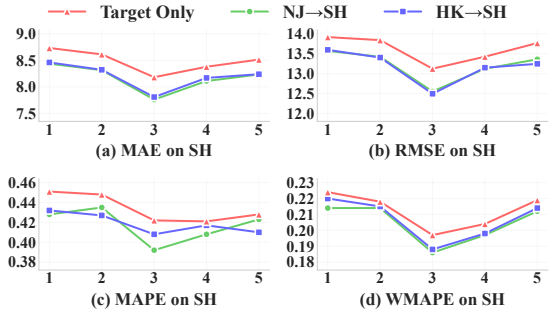

Figure 4: Performance comparison with different decomposition encoder iterations m on Shanghai (SH) as target city.

are presented in Table 2. The variants are:

- **w/o TCB/PCB**: Remove trend-coefficient or period-coefficient branches from semantic bridge encoder.
- **w/o SBE**: Remove the semantic bridge encoder.
- **w/o PLM**: Remove the pre-trained language model.
- **w/o TA**: Remove the task-specific adapter.
- **w/o GDA-A/S/T/R**: Remove GDAConv from all components (A), semantic bridge encoder only (S), task adapter only (T), or regression layer only (R).

The results in Table 2 reveal several key insights. The PLM emerges as the most critical component, with its removal causing the largest performance degradation (16.4% for NJ→SH and 16.9% for HK→SH), confirming that leveraging PLM knowledge makes essential effects. The semantic bridge encoder also plays a vital role, removing the trend-coefficient branch increases MAE by 6.0% and 5.4% for NJ→SH and HK→SH, respectively, and the period-coefficient branch performs similarly. The GDAConvs vary in importance in different components, within the task adapter being the most critical. Overall, the ablation study validates each component contributes meaningfully to the success in cross-city urban flow prediction.

Hyperparameters Sensitivity Analysis

We analyze the impact of two key hyperparameters on PLM-CUP's performance: the regularization coefficient for MMD loss and the number of decomposition encoder iterations.

Regularization Coefficient The sensitivity of MMD coefficient κ is shown in Fig. 3. Optimal performance occurs at $\kappa = 0.01$ (MAE: 7.761/7.809, RMSE: 12.552/12.494). Smaller values (0.001–0.005) offer insufficient alignment, limiting PLM transfer, while larger ones (0.05–0.1) over-constrain the encoder, hindering target-specific pattern learning. Performance drops more sharply with over regularization, causing MAE rises 7.2% from $\kappa = 0.01$ to 0.1, versus a 2.4% drop to 0.001, which calls for moderate regularization to balance alignment and adaptation.

Number of Decomposition Encoders Fig. 4 shows how decomposition iterations affect the performance. The best results occur at $m = 3$ iterations (MAE: 7.761/7.809, RMSE: 12.552/12.494; MAPE: 0.392/0.408; WMAPE: 0.186/0.188). Fewer iterations (1–2) lead to incomplete trend/periodic decomposition, limiting alignment with language semantic, e.g., 1 iteration increases MAE to 8.438/8.462 (+8.7%). Excessive iterations (4–5) cause over-decomposition and noise fitting, with MAE rising 6.1% from 3 to 5 iterations. This supports our theory that effective semantic decomposition minimizes domain shift.

Conclusion

In this paper, we proposed PLM-CUP, the first cross-city transfer paradigm that leverages pre-trained language models for urban flow prediction. Grounded in information-theoretic insights, our approach highlights the importance of aligning domain representations through a semantic bridge encoder and a task-specific adapter. PLM-CUP adopts a simple three-stage architecture and introduces GDAConv to enhance spatial modeling. Extensive experiments demonstrate its superior cross-city generalization, pioneering for integrating the PLM into cross-city urban flow prediction.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities, NO.NJ2024031, the Natural Science Foundation of China (62402220), and the Natural Science Foundation of Jiangsu Province, China (BK20241402).

References

BAI, L.; Yao, L.; Li, C.; Wang, X.; and Wang, C. 2020. Adaptive Graph Convolutional Recurrent Network for Traffic Forecasting. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 17804–17815. Curran Associates, Inc.

Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; and Vaughan, J. W. 2010. A theory of learning from different domains. *Machine learning*, 79: 151–175.

Cao, L.; Wang, B.; Jiang, G.; Yu, Y.; and Dong, J. 2025. Spatiotemporal-aware Trend-Seasonality Decomposition Network for Traffic Flow Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 11463–11471.

Chen, Y.; Gu, J.; Zhuang, F.; Lu, X.; and Sun, M. 2022. Exploiting Hierarchical Correlations for Cross-City Cross-Mode Traffic Flow Prediction. In *2022 IEEE International Conference on Data Mining (ICDM)*, 891–896.

Fu, R.; Zhang, Z.; and Li, L. 2016. Using LSTM and GRU neural network methods for traffic flow prediction. In *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 324–328.

Gao, Q.; Wang, Z.; Huang, L.; Trajcevski, G.; Liu, G.; and Chen, X. 2025. Responsive Dynamic Graph Disentanglement for Metro Flow Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 11690–11698.

Godfrey, L. B.; and Gashler, M. S. 2017. Neural decomposition of time-series data for effective generalization. *IEEE transactions on neural networks and learning systems*, 29: 2973–2985.

Guo, S.; Lin, Y.; Feng, N.; Song, C.; and Wan, H. 2019. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01): 922–929.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive Representation Learning on Large Graphs. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Hendrycks, D.; and Gimpel, K. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Huang, Q.; Shen, L.; Zhang, R.; Ding, S.; Wang, B.; Zhou, Z.; and Wang, Y. 2023. CrossGNN: Confronting Noisy Multivariate Time Series Via Cross Interaction Refinement. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 46885–46902. Curran Associates, Inc.

Huang, Z.; Shen, X.; Tian, X.; Li, H.; Huang, J.; and Hua, X.-S. 2020. Spatio-temporal inception graph convolutional networks for skeleton-based action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, 2122–2130.

Kong, W.; Guo, Z.; and Liu, Y. 2024. Spatio-temporal pivotal graph neural networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 8627–8635.

Li, J.; Tang, T.; Zhao, W. X.; Nie, J.-Y.; and Wen, J.-R. 2024a. Pre-Trained Language Models for Text Generation: A Survey. *ACM Comput. Surv.*, 56(9).

Li, Y.; Yu, R.; Shahabi, C.; and Liu, Y. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations*.

Li, Z.; Hu, Z.; Han, P.; Gu, Y.; and Cai, S. 2025. SSL-STMFormer Self-Supervised Learning Spatio-Temporal Entanglement Transformer for Traffic Flow Prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 12130–12138.

Li, Z.; Huang, W.; Zhao, K.; Yang, M.; Gong, Y.; and Chen, M. 2024b. Urban region embedding via multi-view contrastive prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 8724–8732.

Liu, C.; Hettige, K. H.; Xu, Q.; Long, C.; Xiang, S.; Cong, G.; Li, Z.; and Zhao, R. 2025. ST-LLM+: Graph Enhanced Spatio-Temporal Large Language Models for Traffic Prediction. *IEEE Transactions on Knowledge and Data Engineering*.

Liu, C.; Yang, S.; Xu, Q.; Li, Z.; Long, C.; Li, Z.; and Zhao, R. 2024. Spatial-Temporal Large Language Model for Traffic Prediction. In *2024 25th IEEE International Conference on Mobile Data Management (MDM)*, 31–40.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Shen, B.; Liang, X.; Ouyang, Y.; Liu, M.; Zheng, W.; and Carley, K. M. 2018. StepDeep: A Novel Spatial-temporal Mobility Event Prediction Framework based on Deep Neural Network. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, 724–733. New York, NY, USA: Association for Computing Machinery. ISBN 9781450355520.

Shi, G.; Zhou, Q.; and Gu, J. 2024. Exploring Idealized Regional Match for Cross-City Cross-Mode Traffic Flow Prediction. In *International Conference on Database Systems for Advanced Applications*, 54–69. Springer.

SHI, X.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-k.; and WOO, W.-c. 2015. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. In Cortes, C.; Lawrence, N.; Lee, D.; Sugiyama, M.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Wang, J.; Lin, Y.; and Li, Y. 2025. GTG: Generalizable Trajectory Generation Model for Urban Mobility. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 834–842.

Wang, L.; Geng, X.; Ma, X.; Liu, F.; and Yang, Q. 2019. Cross-city transfer learning for deep spatio-temporal prediction. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, 1893–1899. AAAI Press. ISBN 9780999241141.

Wang, S.; Miao, H.; Li, J.; and Cao, J. 2022. Spatio-Temporal Knowledge Transfer for Urban Crowd Flow Prediction via Deep Attentive Adaptation Networks. *IEEE Transactions on Intelligent Transportation Systems*, 23(5): 4695–4705.

Wu, X.; Manton, J. H.; Aickelin, U.; and Zhu, J. 2020. Information-theoretic analysis for transfer learning. In *2020 IEEE International Symposium on Information Theory (ISIT)*, 2819–2824. IEEE.

Yang, M.; Li, X.; Xu, B.; Nie, X.; Zhao, M.; Zhang, C.; Zheng, Y.; and Gong, Y. 2025. STDA: Spatio-Temporal Deviation Alignment Learning for Cross-city Fine-grained Urban Flow Inference. *IEEE Transactions on Knowledge and Data Engineering*.

Yu, B.; Yin, H.; and Zhu, Z. 2018. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 3634–3640. International Joint Conferences on Artificial Intelligence Organization.

Zhang, W.; Han, J.; Xu, Z.; Ni, H.; Liu, H.; and Xiong, H. 2024. Urban Foundation Models: A Survey. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, 6633–6643. New York, NY, USA: Association for Computing Machinery. ISBN 9798400704901.

Zhang, Y.; Li, M.; Long, D.; Zhang, X.; Lin, H.; Yang, B.; Xie, P.; Yang, A.; Liu, D.; Lin, J.; et al. 2025a. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176*.

Zhang, Y.; Li, Y.; Zhou, X.; Kong, X.; and Luo, J. 2022. STrans-GAN: Spatially-Transferable Generative Adversarial Networks for Urban Traffic Estimation. In *2022 IEEE International Conference on Data Mining (ICDM)*, 743–752.

Zhang, Y.; Wang, X.; Yu, X.; Sun, Z.; Wang, K.; and Wang, Y. 2025b. Drawing informative gradients from sources: A one-stage transfer learning framework for cross-city spatiotemporal forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 1147–1155.

Zheng, Q.; Yao, Z.; and Zhang, Y. 2025. ST-ReP: Learning Predictive Representations Efficiently for Spatial-Temporal Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 13419–13427.