

URBAN-PROTOS: Discrete Semantic Prototypes for Cross-City Region Representation Learning

Zihang Xu^{1*}, Xinshun Feng^{2*}, Haozhi Xie^{1*}, Yi Qiao¹, Liangzhe Han^{1†}, Leilei Sun¹

¹State Key Laboratory of Complex & Critical Software Environment,
Beihang University, Beijing, China

²Hangzhou International Innovation Institute,
Beihang University, Hangzhou, China

{zihangxu,xinshunfeng,haozhixie,qiaoy,liangzhehan,leileisun}@buaa.edu.cn

Abstract

Urban region representation learning plays a fundamental role in urban computing, providing compact embeddings that support a wide range of downstream analytical tasks. Existing methods are developed under a single-city setting and rely on continuous vector representations learned from city-specific spatiotemporal data. Learning transferable and semantically meaningful region representations in cross-city scenarios remains a significant challenge. In this paper, we propose URBAN-PROTOS, a two-stage framework for cross-city learning of urban region representations. We first abstract multi-city mobility data into discrete Urban Prototypes, which capture city-invariant spatiotemporal regularities from localized heterogeneous contexts. These prototypes serve as shared semantic units across cities and are further integrated into the semantic space of large language models through a prototype-guided alignment mechanism. Extensive experiments conducted demonstrate that URBAN-PROTOS consistently outperforms existing methods in cross-city transfer settings, while maintaining strong performance in single-city evaluations. Further analyses show that the learned region embeddings encode deep semantic regularities shared across cities, providing robust support for cross-city generalization.

1 Introduction

The rapid acceleration of urbanization has produced vast amounts of heterogeneous urban data, providing unprecedented opportunities to understand and model the complex characteristics of urban environments (Rathore et al., 2018; Hu et al., 2021; Zhang et al., 2023). Leveraging these data, urban region representation learning has emerged as a fundamental problem in urban computing (Fu et al., 2019; Liu et al., 2020), aiming to encode each urban region into a compact embedding. Such

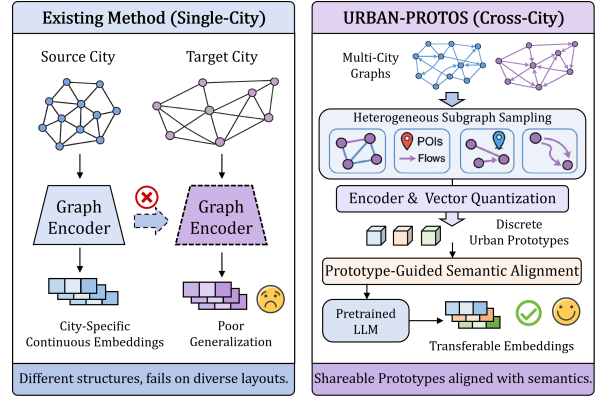


Figure 1: Illustration of cross-city urban region representation learning. Existing methods learn embeddings on fixed city graphs and generalize poorly across cities. URBAN-PROTOS derives discrete Urban Prototypes from subgraphs and aligns them with the semantic space of LLMs for transferable region embeddings.

region embeddings serve as essential foundations for downstream applications (Zhang et al., 2020; Wu et al., 2022; Zhou et al., 2023; Li et al., 2024a).

Prior work has extensively studied region representation learning from human mobility logs (Wang and Li, 2017; Yao et al., 2018), with methods typically organized as flow-based (Li et al., 2024b; Yuan et al., 2024), graph-based (Wu et al., 2022), and side-information-based approaches (Jenkins et al., 2019; Kim and Yoon, 2025; Zhang et al., 2020; Li et al., 2024a; Zhou et al., 2023). However, most methods rely on city-specific graph structures and continuous embeddings, which often lose robustness and semantic coherence under cross-city distribution shifts. As shown in Fig. 1, encoders trained on a fixed city graph may hard-code source-specific structures, leading to degraded transfer when the target city exhibits different spatial and mobility patterns. Although recent transfer-learning methods mitigate this issue by building source-city pattern banks and aligning target features via attention mechanisms (Liu et al., 2023)

*Equal contribution. †Corresponding author.

or similarity matrices (Jin et al., 2023), they remain largely flow-centric and are mainly evaluated on forecasting tasks, leaving transferable region embeddings with city-invariant structural and functional semantics underexplored.

Recent advances in LLMs (OpenAI et al., 2023; Touvron et al., 2023; Bai et al., 2023) offer a new opportunity for cross-city region representation learning. Pretrained on large and diverse corpora, LLMs provide strong semantic abstraction and generalization capabilities (Zhao et al., 2026), making their semantic space a potential unified space for heterogeneous urban environments. Although LLMs can encode spatial and temporal structure (Gurnee and Tegmark, 2023), they are primarily optimized for language and do not explicitly impose the heterogeneous graph and locality biases required by urban mobility data. This raises a key question: *how can heterogeneous urban spatiotemporal patterns be transformed and aligned with the semantic space of LLMs for transferable region representation learning?*

To address this challenge, we propose **URBAN-PROTOS**, a two-stage approach that learns discrete Urban Prototypes from multi-city spatiotemporal data and integrates them with the semantic space of pretrained large language models for cross-city region representation learning. In Stage I, URBAN-PROTOS learns *discrete Urban Prototypes* from multi-city mobility data. In Stage II, we design a *prototype-guided semantic alignment* mechanism to integrate Urban Prototypes into the semantic space of a pretrained large language model. Through a progressive alignment strategy, the LLM associates prototype-level spatiotemporal patterns with its pretrained semantic space, enabling heterogeneous urban signals to be integrated into a unified embedding representation.

In general, our contributions are summarized:

- **Framework.** We propose a unified two-stage framework for cross-city region representation learning, which learns region embeddings that generalize across heterogeneous cities by decoupling spatiotemporal pattern abstraction from semantic representation.
- **Method.** We introduce *discrete Urban Prototypes* to capture city-invariant spatiotemporal regularities, and design a *prototype-guided semantic alignment* mechanism that integrates abstractions into the embedding space of a

large language model, yielding a unified and transferable region representation space.

- **Evaluation.** Extensive experiments on multiple scenarios demonstrate that URBAN-PROTOS achieves strong performance in both cross-city transfer and single-city settings. Further analyses show that the learned embeddings encode deep semantic regularities shared across cities, providing robust support for cross-city knowledge transfer.

2 Preliminary

Definition 1 (Urban Region). A city is partitioned into N disjoint urban regions, denoted as $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$, where each region v_i represents a geographical subarea of the city.

Definition 2 (Human Mobility) Human mobility data capture the spatiotemporal patterns of population movement within urban regions. In this study, we aggregate the mobility data on a regional basis. The mobility data of the i 'th region can be denoted as $\mathbf{X}_i \in \mathbb{R}^T$ ($1 \leq i \leq N$), where T is the time steps of the mobility data and each element presents the flow numbers of one time step.

Definition 3 (Heterogeneous Regional Subgraph) To capture the local spatial and functional context of an urban region, we construct a heterogeneous regional subgraph centered at region S_i . Given neighborhood sizes $K_{\text{geo}}, K_{\text{flow}}, K_{\text{poi}} \in [1, N]$, the subgraph $\mathcal{G}_i^{\text{sub}} = (\mathcal{V}_i^{\text{sub}}, \mathcal{E}_i^{\text{sub}})$ is formed by selecting geographically proximate regions, regions with strong human mobility interactions, and regions with similar POI distributions to S_i . Edges in the subgraph encode heterogeneous relational semantics, including geographic adjacency, mobility intensity, and functional similarity, providing a multi-view description of regional relationships.

Problem (Cross-City Region Representation Learning) Given a source city $\mathcal{C}^A$ and a target city $\mathcal{C}^B$, each partitioned into urban regions with corresponding human mobility data $\{\mathbf{X}_i^A\}, \{\mathbf{X}_i^B\}$ and heterogeneous regional subgraphs $\{\mathcal{G}_i^{\text{sub},A}\}, \{\mathcal{G}_i^{\text{sub},B}\}$, the goal is to learn a transferable region encoder $\text{Enc}(\cdot)$ from the source city. The learned encoder is directly applied to the target city without further tuning to obtain region representations

$$\mathbf{z}_i^B = \text{Enc}(\mathbf{X}_i^B, \mathcal{G}_i^{\text{sub},B}),$$

which are expected to preserve essential spatiotemporal structures and semantic characteristics of urban regions across different cities.

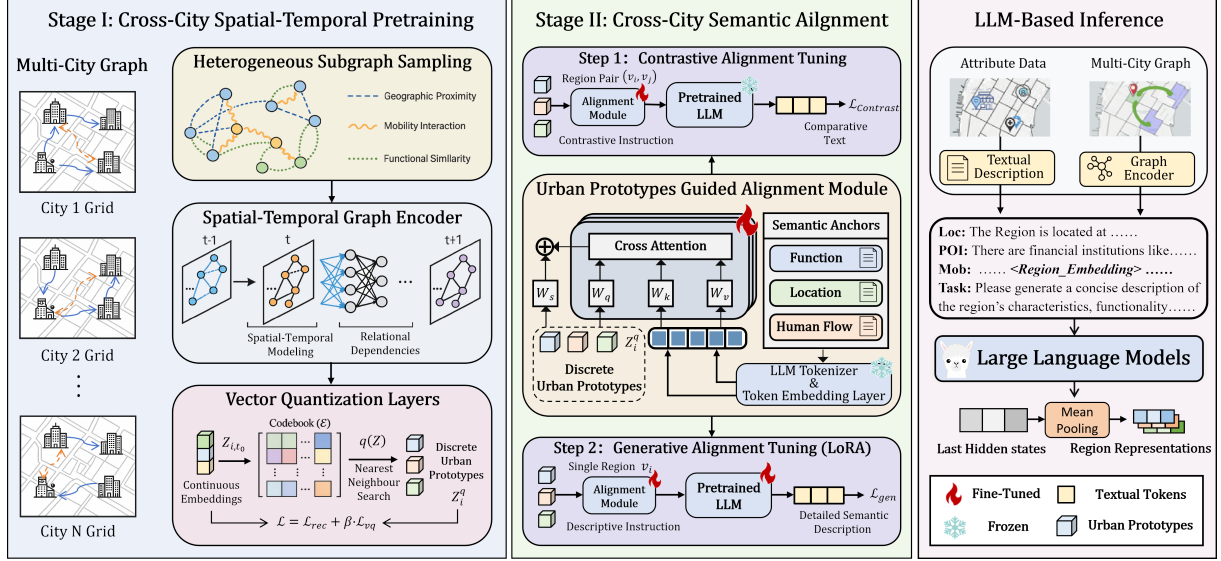


Figure 2: Overview of URBAN-PROTOS. The framework consists of two training stages. In Stage I, heterogeneous regional subgraphs and human mobility data are used to learn spatiotemporal region embeddings and discrete semantic prototypes. In Stage II, the learned prototypes are aligned with the semantic space of a large language model through instruction tuning, enabling transferable and semantically meaningful urban region representations.

3 Methodology

3.1 Overall Framework

We propose URBAN-PROTOS, a two-stage framework for cross-city urban region representation learning that integrates heterogeneous spatiotemporal urban data with large language models. As illustrated in Figure 2, the framework first learns transferable region representations from urban contexts and then aligns them with the semantic space of a pretrained LLM for cross-city generalization.

3.2 Cross-City Spatial-Temporal Pretraining

Stage I aims to learn *transferable and city-invariant urban region representations* from heterogeneous spatiotemporal data. Given multi-city human mobility observations, the goal is to extract canonical spatiotemporal patterns that are shared across cities and abstract them into a discrete latent space serving as the basis for cross-city knowledge transfer.

3.2.1 Problem Formulation

Let $\mathcal{C}$ denote a set of cities. For each city $c \in \mathcal{C}$, we consider a set of regions $\mathcal{V}^{(c)} = \{v_1, \dots, v_{N_c}\}$ and a temporal human mobility tensor

$$\mathbf{X}^{(c)} \in \mathbb{R}^{N_c \times T}, \quad (1)$$

where $X_{i,t}^{(c)}$ denotes the human flow intensity of region v_i at time t .

The objective of Stage I is to learn a mapping

$$f_{\text{proto}} : (\mathbf{X}^{(c)}, \mathcal{G}^{(c)}) \rightarrow \mathcal{Z}^{\text{disc}}, \quad (2)$$

where $\mathcal{G}^{(c)}$ denotes heterogeneous relational structures among regions, and $\mathcal{Z}^{\text{disc}}$ is a discrete latent space shared across all cities.

Heterogeneous Subgraph Sampling. To mitigate the heterogeneity of global city graphs, we adopt a localized modeling strategy. For each region v_i at time t_0 , we sample a subgraph

$$\mathcal{G}_{i,t_0}^{\text{sub}} = (\mathcal{V}_{i,t_0}^{\text{sub}}, \{\mathbf{A}_{i,t_0}^{(r)}\}_{r=1}^R), \quad (3)$$

where $\mathcal{V}_{i,t_0}^{\text{sub}}$ contains the center region and its neighbors sampled from multiple relational views (e.g., geographic proximity, mobility interaction, functional similarity), and $\mathbf{A}^{(r)}$ denotes the adjacency matrix of relation r .

Each subgraph is associated with a local mobility patch

$$\mathbf{X}_{i,t_0} \in \mathbb{R}^{|\mathcal{V}_{i,t_0}^{\text{sub}}| \times L}, \quad (4)$$

covering a temporal window of length L . This construction ensures that regions from cities are represented through comparable local contexts.

3.2.2 Spatiotemporal Graph Encoder.

We employ a spatiotemporal graph encoder δ_{enc} to jointly model temporal dynamics and relational dependencies within each sampled subgraph. Formally, the encoder maps the local mobility patch and heterogeneous graph structure to continuous node embeddings:

$$\mathbf{Z}_{i,t_0} = \delta_{\text{enc}}(\mathbf{X}_{i,t_0}, \mathcal{G}_{i,t_0}^{\text{sub}}), \quad \mathbf{Z}_{i,t_0} \in \mathbb{R}^{k \times d}, \quad (5)$$

where $k = |\mathcal{V}_{i,t_0}^{\text{sub}}|$ and d is the embedding dimension. The encoder is designed to aggregate multi-relational neighborhood information while capturing short-term and long-term mobility patterns, yielding continuous region representations that encode local spatiotemporal contexts.

3.2.3 Discrete Urban Prototype Learning.

Although continuous embeddings encode rich information, they are sensitive to city-specific distributions and hard to align across domains. To induce a transferable representation space, we introduce a vector-quantized layer that maps embeddings to a finite set of discrete Urban Prototypes. Specifically, we maintain a learnable codebook

$$\mathcal{E} = \{\mathbf{e}_1, \dots, \mathbf{e}_M\}, \quad \mathbf{e}_m \in \mathbb{R}^d, \quad (6)$$

shared across all cities. Each continuous embedding $\mathbf{z} \in \mathbb{R}^d$ is quantized to its nearest prototype:

$$q(\mathbf{z}) = \arg \min_{\mathbf{e}_m \in \mathcal{E}} \|\mathbf{z} - \mathbf{e}_m\|_2^2. \quad (7)$$

The quantized representation $\mathbf{z}^q = q(\mathbf{z})$ serves as the discrete Urban Prototype associated with the region. This vector quantization mechanism encourages regions with similar functional roles and mobility patterns to share the same prototypes, thereby inducing implicit cross-city alignment.

Optimization Objective. The spatiotemporal encoder and the Urban Prototype codebook are jointly optimized via self-supervised objectives derived from mobility data. In particular, we employ a reconstruction-based pretraining objective $\mathcal{L}_{\text{rec}}$ to preserve essential spatiotemporal information, together with a vector quantization commitment loss $\mathcal{L}_{\text{vq}}$ to stabilize the discrete latent space:

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{\text{rec}} + \beta \mathcal{L}_{\text{vq}}, \\ \mathcal{L}_{\text{vq}} &= \|\text{sg}[\mathbf{Z}] - \mathbf{Z}^q\|_2^2. \end{aligned} \quad (8)$$

where $\text{sg}[\cdot]$ denotes the stop-gradient operator and β controls the strength of commitment to discrete prototypes.

After optimization, each urban region is represented by a discrete Urban Prototype summarizing its city-agnostic spatiotemporal characteristics. These prototypes form a transferable semantic basis for alignment with LLMs in Stage II.

3.3 Cross-City Semantic Alignment

While Stage I learns city-invariant discrete Urban Prototypes that summarize spatiotemporal patterns,

these prototypes remain purely numerical and lack explicit semantics. The objective of Stage II is to align these discrete prototypes with the semantic space of a pretrained large language model (LLM), enabling semantic interpretation, reasoning, and cross-city generalization.

3.3.1 Prototype-to-Semantic Alignment Module

Let $\mathbf{z}_i^q \in \mathbb{R}^d$ denote the discrete Urban Prototype obtained for region v_i . To integrate such prototypes into the LLM, we design a prototype-to-semantic alignment module that maps them into the LLM embedding space while preserving their spatiotemporal semantics.

Direct linear projection is insufficient to ensure semantic consistency between numerical prototypes and linguistic representations. To guide the alignment process, we introduce a set of **semantic anchors**, which serve as reference concepts describing common urban characteristics and mobility patterns. These anchors are expressed as short textual phrases and grouped into three categories:

$$\mathcal{A} = \{\mathcal{A}_{\text{func}}, \mathcal{A}_{\text{loc}}, \mathcal{A}_{\text{mob}}\}, \quad (9)$$

where $\mathcal{A}_{\text{func}}$ describes urban functions (e.g., residential area, commercial district), $\mathcal{A}_{\text{loc}}$ captures spatial properties (e.g., city center, suburban area), and $\mathcal{A}_{\text{mob}}$ characterizes mobility patterns (e.g., morning peak, high population inflow).

Each anchor phrase is tokenized and embedded using the frozen LLM tokenizer and embedding layer, yielding anchor embeddings

$$\mathbf{E}_{\mathcal{A}} = \text{Embed}_{\text{LLM}}(\mathcal{A}) \in \mathbb{R}^{|\mathcal{A}| \times d_L}, \quad (10)$$

where d_L is the LLM hidden dimension.

To align Urban Prototypes with semantic concepts, we treat each discrete prototype $\mathbf{z}_i^q$ as a query and perform cross-attention over the anchor embeddings as follows:

$$\mathbf{e}_i^{\text{sem}} = \text{Softmax}\left(\frac{Q(\mathbf{z}_i^q)K(\mathbf{E}_{\mathcal{A}})^{\top}}{\sqrt{d_L}}\right)V(\mathbf{E}_{\mathcal{A}}), \quad (11)$$

where $Q(\cdot)$, $K(\cdot)$, and $V(\cdot)$ are learnable linear projections.

This operation extracts anchor-aware semantic signals aligned with the spatiotemporal characteristics encoded in the prototype. To preserve structural information learned in Stage I, we further project the prototype into the LLM space:

$$\mathbf{e}_i^{\text{proj}} = \mathbf{W}_p \mathbf{z}_i^q + \mathbf{b}_p, \quad (12)$$

where $\mathbf{W}_p \in \mathbb{R}^{d_L \times d}$ and $\mathbf{b}_p \in \mathbb{R}^{d_L}$.

The final prototype-conditioned LLM input embeddings are obtained by concatenation:

$$\mathbf{e}_i = \mathbf{e}_i^{\text{sem}} \parallel \mathbf{e}_i^{\text{proj}}. \quad (13)$$

Through anchor-guided alignment, discrete Urban Prototypes are mapped into the semantic space of the pretrained LLM. This alignment associates spatiotemporal patterns with high-level urban concepts, yielding representations that integrate transferable structural regularities with semantic priors and lay the foundation for subsequent instruction-based adaptation.

3.3.2 Two-Step Instruction Tuning

We adopt a two-step instruction tuning strategy to align Urban Prototypes with language semantics while preserving the intrinsic reasoning and generation capabilities of the LLM.

Step 1: Foundational Alignment via Contrastive Tuning. In Step 1, we freeze all LLM parameters θ_{LLM} and optimize only the alignment module. The goal is to establish a coarse semantic correspondence between prototype embeddings and textual concepts. Given a region pair (v_i, v_j) with embeddings $(\mathbf{e}_i, \mathbf{e}_j)$, we construct comparative instruction prompts that require the model to reason about their relative properties (e.g., mobility intensity or spatial centrality). The alignment module is trained to maximize the likelihood of generating the target comparison text:

$$f^* = \arg \max_f \log P(\mathbf{w} \mid \mathbf{e}_i, \mathbf{e}_j; \theta_{\text{LLM}}), \quad (14)$$

where $\mathbf{w} = (w_1, \dots, w_T)$ denotes the generated response and f represents the alignment module.

This step encourages the model to associate prototype differences with semantic distinctions, achieving an initial cross-modal alignment without modifying the LLM itself.

Step 2: Semantic Integration via Generative Tuning. In the second step, we further refine the alignment by jointly optimizing the alignment module and a small subset of LLM parameters. To avoid catastrophic forgetting, we adopt parameter-efficient fine-tuning based on low-rank adaptation (LoRA) (Hu et al., 2022). Given a single region embedding $\mathbf{e}_i$, the model is instructed to generate a detailed semantic description of the region:

$$\theta_{\text{LLM}}^*, f^* = \arg \max_{\theta_{\text{LLM}}, f} \log P(\mathbf{w} \mid \mathbf{e}_i), \quad (15)$$

where $\mathbf{w}$ is a free-form textual description capturing functional roles, spatial properties, and mobility characteristics.

Through this step, the LLM internalizes the semantics encoded in Urban Prototypes and becomes capable of producing interpretable, context-aware region representations that generalize across cities.

3.4 LLM-based Representation Inference

URBAN-PROTOS employs a pretrained LLM as the semantic backbone to integrate heterogeneous urban signals into a shared embedding space, producing unified and transferable region representations across cities.

For each region v_i , the LLM input is constructed by concatenating three components along the token dimension:

$$E_{\text{Input}} = E_{\text{Instruct}} \parallel E_{\text{Attr}} \parallel E_{\text{Proto}}, \quad (16)$$

where E_{Instruct} encodes task-level instructions for cross-city representation learning, E_{Attr} represents static regional attributes such as POI distributions, and E_{Proto} denotes the prototype-aligned embeddings from Stage II that capture semantically aligned spatiotemporal patterns.

The LLM encodes the input sequence and aggregates the hidden states into a fixed-dimensional region representation:

$$\begin{aligned} \mathbf{H}_i^{\text{LLM}} &= \text{LLM}(E_{\text{Input}}), \\ \mathbf{R}_i &= \text{MeanPooling}(\mathbf{H}_i^{\text{LLM}}), \end{aligned} \quad (17)$$

where $\mathbf{R}_i \in \mathbb{R}^{d_L}$ denotes the final representation of region v_i . By integrating discrete Urban Prototypes, static regional attributes, and task-level context within the LLM semantic space, URBAN-PROTOS produces region representations that capture transferable spatiotemporal regularities and high-level urban semantics. These representations form a unified embedding space for downstream cross-city urban analysis tasks.

4 Experiments

In this section, we conduct extensive experiments on multiple downstream urban tasks across two cities to evaluate the effectiveness of URBAN-PROTOS with following research questions (RQs):

- **RQ1:** How does URBAN-PROTOS perform in cross-city settings compared with existing region representation learning and transfer learning methods?

Table 1: Performance comparison under cross-city transfer settings.

City	Method	Business			Crime			Crash			311 Service		
		MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
CHI2NYC	PatchTST	21.78	29.16	0.26	66.74	104.12	0.09	7.05	9.61	0.16	305.18	417.62	0.11
	TSLANet	24.95	32.29	0.09	71.05	106.44	0.06	7.65	10.31	0.04	320.82	437.34	0.04
	TimesURL	22.06	29.78	0.22	63.92	99.81	0.16	7.18	9.97	0.11	298.27	412.61	0.14
	STID	24.13	31.76	0.15	72.36	108.87	0.01	7.52	10.17	0.07	313.74	428.76	0.10
	STBP	17.21	24.73	0.40	60.12	94.61	0.23	6.82	9.52	0.15	293.51	407.62	0.17
	TransGTR	20.58	28.19	0.30	63.44	97.88	0.19	7.02	9.87	0.12	289.94	414.68	0.13
	CrossST	<u>16.62</u>	<u>24.59</u>	<u>0.45</u>	59.42	94.88	0.25	6.87	9.68	0.14	290.89	413.12	0.14
	CoRE	21.74	29.20	0.26	66.68	104.21	0.09	7.06	9.63	0.16	305.41	418.03	0.11
	UrbanGPT	18.03	27.86	0.32	<u>59.21</u>	<u>92.88</u>	<u>0.26</u>	6.58	<u>9.21</u>	<u>0.18</u>	<u>280.94</u>	404.87	0.15
	UrbanCLIP	19.11	28.02	0.30	59.84	97.31	0.20	<u>6.53</u>	9.27	0.18	284.06	<u>389.42</u>	<u>0.18</u>
	Ours	15.18	22.26	0.53	58.27	89.75	0.27	6.46	9.09	0.20	277.90	387.07	0.20
NYC2CHI	PatchTST	22.94	51.03	0.22	83.06	142.81	0.14	285.96	396.12	0.15	183.61	233.21	0.18
	TSLANet	24.17	52.76	0.16	90.04	145.31	0.11	293.84	414.92	0.08	203.12	256.43	0.06
	TimesURL	19.58	51.39	0.20	86.32	143.94	0.12	252.81	380.74	0.21	189.71	241.12	0.15
	STID	23.72	52.31	0.17	92.87	149.32	0.07	291.52	410.34	0.10	197.84	249.68	0.10
	STBP	16.83	44.96	0.35	83.24	138.85	0.18	238.55	351.24	0.29	178.42	226.83	0.19
	TransGTR	17.91	45.88	0.34	86.71	141.76	0.16	257.42	372.96	0.23	194.88	244.96	0.14
	CrossST	16.54	48.19	0.30	80.96	136.51	0.20	<u>225.73</u>	348.11	0.33	184.33	237.24	0.16
	CoRE	22.87	51.17	0.21	82.94	143.21	0.13	286.74	397.03	0.14	184.02	234.07	0.17
	UrbanGPT	18.47	46.12	0.33	<u>76.89</u>	136.41	0.21	228.36	358.92	0.24	<u>170.98</u>	229.11	0.19
	UrbanCLIP	<u>15.92</u>	<u>44.01</u>	<u>0.36</u>	82.07	127.64	0.23	227.11	339.58	0.34	175.31	218.94	0.20
	Ours	15.20	42.12	0.39	72.31	123.69	0.25	207.98	308.78	0.41	159.92	202.06	0.32

- **RQ2:** What are the contributions of the proposed modules in enhancing the effectiveness and generalization?
- **RQ3:** Can URBAN-PROTOS remain competitive in single-city settings against classical urban region representation methods?
- **RQ4:** Does URBAN-PROTOS learn transferable and semantically meaningful region representations across cities?

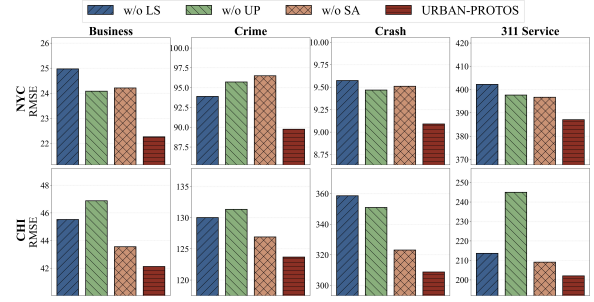


Figure 3: Ablation study of the proposed framework.

4.1 Experimental Settings

We conduct experiments on two large-scale cities, New York City (NYC) and Chicago (CHI), where each city is partitioned into grid-based regions. Detailed dataset descriptions are provided in Appendix A.1. We then compare URBAN-PROTOS with representative region representation learning methods under both single-city and cross-city settings. Detailed descriptions are provided in Appendix A.2. We evaluate model performance using MAE, RMSE, and R^2 . Additional implementation details are reported in Appendix A.3.

4.2 Cross-City Performance(RQ1)

In this section, we evaluate URBAN-PROTOS under cross-city transfer settings, where models trained on a source city are directly applied to

a target city. The results are reported in Table 1 across four downstream tasks. The baseline methods can be broadly categorized into time-series-based approaches, spatiotemporal forecasting models, graph-based transfer methods, and recent LLM-driven urban models. Conventional time-series-based approaches exhibit limited transferability, as human mobility dynamics are highly city-dependent and difficult to generalize across heterogeneous urban environments. Spatiotemporal forecasting models show mixed transferability, since they either tie region representations to city-specific indices or reuse recurring patterns in a purely numerical space without semantic grounding, and thus remain fragile under structural discrepancies between cities. Graph-based transfer methods improve performance by modeling traffic-

Table 2: Performance comparison under single-city settings.

City	Method	Business			Crime			Crash			311 Service		
		MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
NYC	GCN	20.97	28.01	0.27	64.21	100.43	0.09	6.80	9.22	0.17	294.79	401.81	0.12
	Graph WaveNet	18.42	25.92	0.38	60.42	93.67	0.23	6.72	9.35	0.16	283.46	392.98	0.14
	ZE-Mob	24.03	31.08	0.10	68.31	102.18	0.06	7.35	9.91	0.04	308.86	419.52	0.04
	MVURE	21.25	28.71	0.23	61.72	96.25	0.17	6.90	9.56	0.11	287.31	393.99	0.15
	HREP	19.83	27.15	0.31	61.29	94.05	0.20	6.76	9.49	0.12	277.63	396.76	0.14
	ReCP	16.01	23.75	0.47	57.44	90.77	0.26	6.61	9.31	0.15	282.15	394.98	0.15
	STID	20.14	27.67	0.30	65.58	100.86	0.15	7.13	9.78	0.10	305.82	418.76	0.10
	STBP	16.28	23.82	0.43	59.06	92.23	0.25	6.63	9.17	0.19	281.06	391.39	0.19
	Ours	14.97	21.45	0.55	54.01	87.53	0.33	6.44	9.01	0.24	261.97	372.86	0.31
CHI	GCN	21.11	47.07	0.23	76.59	131.77	0.15	263.21	365.96	0.17	167.75	215.02	0.23
	Graph WaveNet	16.82	43.72	0.34	76.88	128.14	0.20	233.84	343.65	0.26	174.32	222.06	0.18
	ZE-Mob	22.28	48.73	0.17	83.02	133.60	0.13	271.11	381.49	0.09	186.75	236.14	0.07
	MVURE	18.03	47.59	0.21	79.15	132.98	0.13	232.99	351.28	0.23	175.14	223.23	0.17
	HREP	16.50	42.39	0.37	79.41	130.32	0.17	237.33	345.90	0.25	178.84	224.17	0.16
	ReCP	15.18	44.30	0.32	74.68	125.47	0.23	209.03	322.66	0.35	169.41	217.88	0.21
	STID	19.36	46.03	0.26	81.64	133.82	0.12	264.53	372.61	0.12	182.73	231.24	0.11
	STBP	15.87	42.61	0.36	75.36	127.35	0.21	216.87	328.94	0.32	170.24	216.85	0.22
	Ours	14.14	41.49	0.43	65.41	116.75	0.34	199.97	293.58	0.45	155.21	193.38	0.37

flow dependencies between regions, but their reliance on mobility interactions and limited handling of structural heterogeneity restricts representation expressiveness, especially in cross-city settings. Recent LLM-based urban models incorporate language priors and show promising results. However, their effectiveness in cross-city transfer remains limited, as spatiotemporal mobility patterns are often introduced implicitly and lack an explicit, city-invariant alignment with the language semantic space. Overall, URBAN-PROTOS consistently outperforms all baselines across tasks and transfer directions, demonstrating strong robustness to cross-city distribution shifts. By abstracting spatiotemporal dynamics into city-invariant Urban Prototypes and aligning them with the semantic space of an LLM, URBAN-PROTOS captures transferable urban regularities with semantic consistency, enabling robust performance across diverse cross-city prediction tasks.

4.3 Ablation Study (RQ2)

To assess the contribution of each component in URBAN-PROTOS, we conduct an ablation study under cross-city settings, with *RMSE* results reported in Figure 3. We consider three variants: w/o LLM Semantics (LS), which removes LLM-derived semantic information; w/o Urban Prototypes (UP), which removes spatiotemporal context modeling and relies only on semantic representations; and w/o Semantic Anchors (SA), which replaces anchor-guided alignment with a simple projection. As shown in Figure 3, URBAN-PROTOS

consistently outperforms all variants, demonstrating the contribution of each component. The degradation of LS and UP indicates that semantic knowledge and mobility-driven spatiotemporal dynamics provide complementary signals for cross-city representation learning. The inferior performance of SA further verifies the role of semantic anchors in stabilizing the alignment between spatiotemporal representations and the LLM semantic space. Overall, these results confirm that the three components are jointly essential for learning transferable urban region representations.

4.4 Single-City Performance (RQ3)

We further evaluate URBAN-PROTOS under single-city settings, where models are trained and tested on the same city. As summarized in Table 2, URBAN-PROTOS achieves consistently strong and competitive performance across downstream tasks. Compared with traditional graph-based and mobility-driven methods, it obtains comparable or better results in most cases, indicating that improved transferability does not come at the cost of in-city performance. We further compare with recent spatiotemporal forecasting backbones, which are competitive on flow-related tasks yet remain inferior to URBAN-PROTOS across all metrics, as they model numerical dynamics alone without semantic abstraction. By integrating spatiotemporal context modeling with LLM-driven semantic representations, URBAN-PROTOS effectively captures both local urban patterns and high-level semantics, leading to stable performance across tasks.

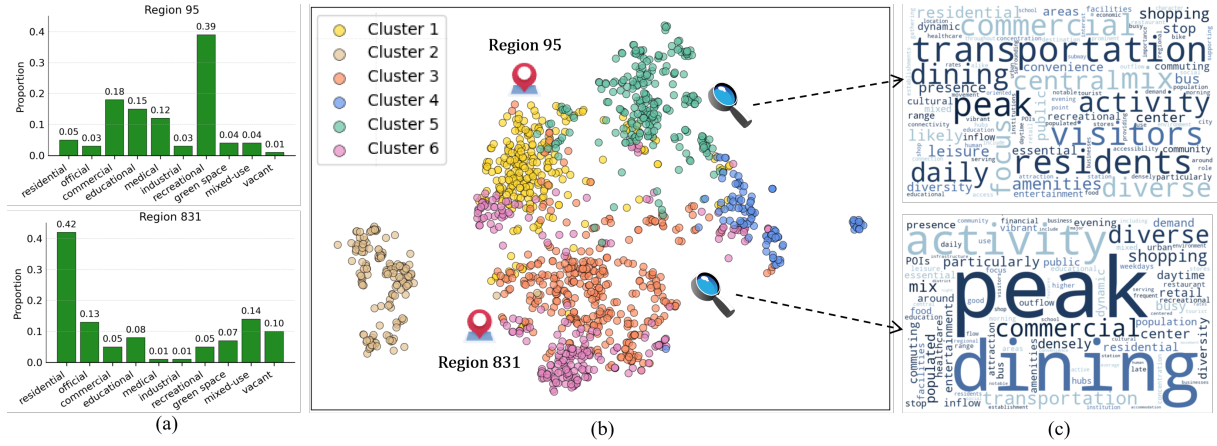


Figure 4: Clustering Analysis and Case Study. (a) Prototype attention scores for representative regions, illustrating how Urban Prototypes contribute to region characterization. (b) Overall clustering results, where representations are grouped into six clusters. (c) Word cloud analysis of clusters, highlighting their intrinsic characteristics.

4.5 Clustering Analysis and Case Study (RQ4)

We conduct a case study on NYC to examine whether URBAN-PROTOS captures interpretable urban semantics. K-Means clustering of region representations shows clear structural separation, as illustrated in Figure 4(b).

Prototype attention analysis. We visualize attention weights over Urban Prototypes for two representative regions from different clusters in Figure 4(a). Region 95 mainly attends to commercial, recreational, and high-activity prototypes, reflecting intensive daily activities and mixed commercial functions. In contrast, Region 831 focuses more on residential, public-service, and mixed-use prototypes, indicating a neighborhood-oriented area dominated by residential and routine community activities. These distinct attention patterns suggest that URBAN-PROTOS adaptively activates prototypes to encode region-specific functions.

Cluster-level semantic interpretation. We further decode clustered region representations with the LLM backbone and aggregate the generated semantic descriptions into word clouds. As shown in Figure 4(c), Cluster 3 is associated with keywords such as peak, dining, commercial, and activity, corresponding to bustling commercial districts, while Cluster 5 emphasizes transportation, residents, daily, and central, indicating well-connected residential areas. Such semantic distinctions show that URBAN-PROTOS organizes regions by coherent functional patterns rather than superficial spatial proximity. These results demonstrate that URBAN-PROTOS learns interpretable and semantically meaningful urban representations, further

supporting its cross-city transferability.

5 Related Work

Urban Region Representation Learning. Urban region representation learning aims to encode regions into compact embeddings. Early mobility-based methods, such as HDGE (Wang and Li, 2017) and ZE-Mob (Yao et al., 2018), model regions as graph nodes connected by mobility flows. Subsequent studies incorporate heterogeneous urban attributes, including POIs, check-ins, and mobility patterns, through multi-view or contrastive learning frameworks (Zhang et al., 2020; Li et al., 2024a; Zhou et al., 2023). However, most methods are still coupled with single-city spatial structures and data distributions, limiting their transferability.

Region Transfer Learning. Region transfer learning transfers knowledge from data-rich cities to data-scarce ones. Pattern-based methods, such as TPB (Liu et al., 2023) and CrossST (Liu and Zhang, 2025), construct spatial or temporal pattern banks to guide target-city learning. CrossTReS (Jin et al., 2022) adaptively re-weights source-city regions to facilitate target-city fine-tuning. Recent studies further explore selective refinement of source-city gradients (Zhang et al., 2025) or latent-manifold alignment (Chen et al., 2025). Nevertheless, these methods rely on continuous representations and comparable cross-city graphs, while semantic consistency and interpretability remain underexplored.

LLMs for Urban Computing. LLMs have demonstrated strong semantic understanding and generalization capabilities (OpenAI et al., 2023; Zhao et al., 2026), and have been extended to time-

series analysis, graph reasoning, and representation learning (Jin et al., 2024; Ren et al., 2024; Tang et al., 2024). In urban computing, recent works such as UrbanGPT (Li et al., 2024b), Mobility-LLM (Gong et al., 2024), and UrbanCLIP (Yan et al., 2024) adapt LLMs to urban analysis, mobility prediction, and multimodal representation alignment. However, existing LLM-based urban methods are largely task- or city-specific, leaving transferable, semantically consistent cross-city region representation learning underexplored.

6 Conclusion

We propose URBAN-PROTOS, a two-stage framework for cross-city learning of urban region representations. By decoupling spatiotemporal pattern abstraction from semantic representation, URBAN-PROTOS learns embeddings that generalize across heterogeneous spatial structures and mobility dynamics. It introduces Urban Prototypes to capture city-invariant spatiotemporal regularities and aligns them with the semantic space of pretrained LLMs, yielding a unified transferable representation space. Extensive experiments show that URBAN-PROTOS outperforms existing methods in cross-city transfer while maintaining strong single-city performance.

Limitations

In this section, we discuss several limitations of this work. First, due to computational constraints, our current implementation is based on a single LLM architecture and model scale. While the empirical results demonstrate the effectiveness of the proposed prototype-guided alignment framework under this setting, evaluations with additional backbone families and model sizes would further strengthen the assessment of scalability and robustness. Second, our downstream evaluation focuses on four representative urban analytics tasks, including business activity, crime, traffic accidents, and 311 service requests. These tasks cover multiple practical aspects of urban region modeling and provide a meaningful testbed for representation learning. Nevertheless, additional downstream scenarios could provide a more comprehensive view of the generality of the learned representations.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. U24B20171).

References

- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, and 29 others. 2023. [Qwen technical report](#). *Preprint*, arXiv:2309.16609.
- Meng Chen, Hongwei Jia, Zechen Li, Wenzhen Jia, Kai Zhao, Hongjun Dai, and Weiming Huang. 2025. [Cross-city latent space alignment for consistency region embedding](#). In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 8261–8274. PMLR.
- Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, and Xiaoli Li. 2024. Tslanet: rethinking transformers for time series representation learning. In *Proceedings of the 41st International Conference on Machine Learning*, pages 12409–12428.
- Yanjie Fu, Pengyang Wang, Jiadi Du, Le Wu, and Xiaolin Li. 2019. Efficient region embedding with multi-view spatial networks: A perspective of locality-constrained spatial autocorrelations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 906–913.
- Letian Gong, Yan Lin, Xinyue Zhang, Yiwen Lu, Xuedi Han, Yichen Liu, Shengnan Guo, Youfang Lin, and Huaiyu Wan. 2024. [Mobility-LLM: Learning Visiting Intentions and Travel Preference from Human Mobility Data with Large Language Models](#). In *Neural Information Processing Systems*.
- Wes Gurnee and Max Tegmark. 2023. Language models represent space and time. *arXiv preprint arXiv:2310.02207*.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *International Conference on Learning Representations*.
- Sheng Hu, Song Gao, Liang Wu, Yongyang Xu, Ziwei Zhang, Haifu Cui, and Xi Gong. 2021. Urban function classification at road segment level using taxi trajectory data: A graph convolutional neural network approach. *Computers, Environment and Urban Systems*, 87:101619.
- Porter Jenkins, Ahmad Farag, Suhang Wang, and Zhenhui Li. 2019. Unsupervised representation learning of spatial data via multimodal embedding. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1993–2002.
- Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. 2024. Time-LLM: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations (ICLR)*.

- Yilun Jin, Kai Chen, and Qiang Yang. 2022. Selective cross-city transfer learning for traffic prediction via source city region re-weighting. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 731–741.
- Yilun Jin, Kai Chen, and Qiang Yang. 2023. Transferable graph structure learning for graph-based traffic forecasting across cities. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pages 1032–1043.
- Namwoo Kim and Yoonjin Yoon. 2025. [Effective urban region representation learning using heterogeneous urban graph attention network \(hugat\)](#). *IEEE Access*, 13:102602–102612.
- Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- Zechen Li, Weiming Huang, Kai Zhao, Min Yang, Yongshun Gong, and Meng Chen. 2024a. Urban region embedding via multi-view contrastive prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8724–8732.
- Zhonghang Li, Lianghao Xia, Jiabin Tang, Yong Xu, Lei Shi, Long Xia, Dawei Yin, and Chao Huang. 2024b. Urbangpt: Spatio-temporal large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5351–5362.
- Aoyu Liu and Yaying Zhang. 2025. Crossst: An efficient pre-training framework for cross-district pattern generalization in urban spatio-temporal forecasting. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, pages 2935–2948. IEEE Computer Society.
- Aoyu Liu and Yaying Zhang. 2026. [A general spatio-temporal backbone with scalable contextual pattern bank for urban continual forecasting](#). In *International Conference on Learning Representations*, volume 2026, pages 72080–72103.
- Jiexi Liu and Songcan Chen. 2024. Timesurl: Self-supervised contrastive learning for universal time series representation learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 13918–13926.
- Zhanyu Liu, Guanjie Zheng, and Yanwei Yu. 2023. [Cross-city few-shot traffic forecasting via traffic pattern bank](#). *Preprint*, arXiv:2308.09727.
- Zhicheng Liu, Fabio Miranda, Weiting Xiong, Junyan Yang, Qiao Wang, and Claudio Silva. 2020. Learning geo-contextual embeddings for commuting flow prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 808–816.
- Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2023. [A time series is worth 64 words: Long-term forecasting with transformers](#). *Preprint*, arXiv:2211.14730.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2023. [Gpt-4 technical report](#). *Preprint*, arXiv:2303.08774.
- M. Mazhar Rathore, Anand Paul, Won-Hwa Hong, HyunCheol Seo, Imtiaz Awan, and Sharjil Saeed. 2018. Exploiting iot and big data analytics: Defining smart digital city using real-time urban data. *Sustainable cities and society*, 40:600–610.
- Xubin Ren, Wei Wei, Lianghao Xia, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024. Representation learning with large language models for recommendation. In *Proceedings of the ACM on Web Conference 2024*, pages 3464–3475.
- ZeZhi Shao, Zhao Zhang, Fei Wang, Wei Wei, and Yongjun Xu. 2022. Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4454–4458.
- Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. 2024. Graphgpt: Graph instruction tuning for large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 491–500.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. [Llama: Open and efficient foundation language models](#). *Preprint*, arXiv:2302.13971.
- Laurens van der Maaten and Geoffrey Hinton. 2008. [Visualizing data using t-sne](#). *Journal of Machine Learning Research*, 9(86):2579–2605.
- Hongjian Wang and Zhenhui Li. 2017. Region representation learning via mobility flow. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 237–246.
- Shangbin Wu, Xu Yan, Xiaoliang Fan, Shirui Pan, Shichao Zhu, Chuanpan Zheng, Ming Cheng, and Cheng Wang. 2022. [Multi-graph fusion networks for urban region embedding](#). In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 2312–2318. International Joint Conferences on Artificial Intelligence Organization. Main Track.
- Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 1907–1913.

Yibo Yan, Haomin Wen, Siru Zhong, Wei Chen, Haodong Chen, Qingsong Wen, Roger Zimmermann, and Yuxuan Liang. 2024. Urbanclip: Learning text-enhanced urban region profiling with contrastive language-image pretraining from the web. In *Proceedings of the ACM on Web Conference 2024*, pages 4006–4017.

Zijun Yao, Yanjie Fu, Bin Liu, Wangsu Hu, and Hui Xiong. 2018. [Representing urban functions through zone embedding with human mobility patterns](#). In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 3919–3925. International Joint Conferences on Artificial Intelligence Organization.

Yuan Yuan, Jingtao Ding, Jie Feng, Depeng Jin, and Yong Li. 2024. Unist: A prompt-empowered universal model for urban spatio-temporal prediction. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4095–4106.

Mingyang Zhang, Tong Li, Yong Li, and Pan Hui. 2020. [Multi-view joint graph representation learning for urban region embedding](#). In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4431–4437. International Joint Conferences on Artificial Intelligence Organization. Special track on AI for CompSust and Human well-being.

Qianru Zhang, Chao Huang, Lianghao Xia, Zheng Wang, Zhonghang Li, and Siuming Yiu. 2023. Automated spatio-temporal graph contrastive learning. In *Proceedings of the ACM Web Conference 2023*, pages 295–305.

Yudong Zhang, Xu Wang, Xuan Yu, Zhaoyang Sun, Kai Wang, and Yang Wang. 2025. 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, pages 1147–1155.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Zican Dong, Yupeng Hou, Beichen Zhang, Yingqian Min, Junjie Zhang, Peiyu Liu, Xiaolei Wang, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, and 4 others. 2026. [A survey of large language models](#). *Frontiers of Computer Science*, 20(12).

Silin Zhou, Dan He, Lisi Chen, Shuo Shang, and Peng Han. 2023. Heterogeneous region embedding with prompt learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4981–4989.

A Appendix

In this section, we provide further details on our methods and experiments.

A.1 Dataset Descriptions

In our experiments, we focus on two major metropolitan areas: New York City and Chicago. To facilitate the study, we partitioned each city into grid-based regions, resulting in 1,390 valid regions for NYC and 1,931 valid regions for CHI. We utilize Point-of-Interest data and shared bike trip records as input features to extract the functional and semantic information embedded within urban regions. The POI data, sourced from OpenStreetMap¹, includes details such as geographical coordinates, names, and categories. Meanwhile, the shared bike trip data was collected from City Bike NYC² and Divvy Bike³, covering the second half of 2023.

To evaluate the effectiveness of the learned representations, we curated four downstream tasks from official municipal portals. Specifically, we gathered datasets from NYC Open Data⁴ and Chicago Data Portal⁵, encompassing the following urban analytics tasks in practice:

(1) Business Counts: This dataset includes over 40,000 businesses in NYC and over 46,000 in Chicago. We filtered active businesses and mapped them to regions, providing insights into commercial activity distribution.

(2) Crime Counts: Covering over 200,000 arrests in NYC and over 240,000 criminal incidents in Chicago, this dataset enables crime distribution analysis by region, highlighting public safety patterns.

(3) Traffic Accidents: With nearly 100,000 reported collisions in NYC and 800,000 in Chicago for 2023, this dataset helps assess traffic safety and identify high-risk areas.

(4) 311 Service: Recording over 310,000 service requests in NYC and 1,300,000 in Chicago, this dataset reflects infrastructure issues, public concerns, and service distribution across urban regions.

A.2 Details of Baselines

We compare URBAN-PROTOS with representative urban region representation learning methods under both *single-city* and *cross-city* settings. These baselines cover graph-based urban modeling,

¹<https://www.openstreetmap.org/>

²<https://citibikenyc.com/system-data>

³<https://divvybikes.com/system-data>

⁴<https://opendata.cityofnewyork.us/data/>

⁵<https://data.cityofchicago.org/>

multi-view representation learning, spatiotemporal forecasting, temporal sequence modeling, region transfer learning, and recent LLM-driven approaches.

Single-City Urban Region Representation. In the single-city setting, all methods are trained and evaluated within the same urban environment. We consider classical urban region representation learning approaches that exploit spatial structures, human mobility patterns, and heterogeneous urban attributes.

- **GCN** (Kipf and Welling, 2016): learns region embeddings by aggregating neighborhood information over a predefined urban graph.
- **Graph WaveNet** (Wu et al., 2019): learns region embeddings by combining a self-adaptive adjacency matrix with stacked dilated causal convolutions.
- **ZE-Mob** (Yao et al., 2018): learns region embeddings by modeling origin-destination co-occurrence patterns extracted from taxi trajectories data.
- **MVURE** (Zhang et al., 2020): learns region embeddings by integrating heterogeneous urban attributes and human mobility flows through multi-view modeling.
- **HREP** (Zhou et al., 2023): learns universal region embeddings by constructing heterogeneous regional relations and prompt-based downstream objectives.
- **ReCP** (Li et al., 2024a): learns region embeddings by combining intra-view and inter-view contrastive learning across multiple urban views and modalities.
- **STID** (Shao et al., 2022): learns region embeddings by attaching spatial and temporal identity embeddings to a purely MLP-based architecture.
- **STBP** (Liu and Zhang, 2026): learns region embeddings by pairing a general spatiotemporal backbone with a scalable contextual pattern bank.

Cross-City Evaluation. For cross-city scenarios, we additionally include baselines designed to encode temporal mobility patterns or explicitly transfer urban knowledge across cities.

Sequence-based Methods. These methods model regional mobility time series and focus on capturing temporal regularities without explicit spatial transfer mechanisms.

- **PatchTST** (Nie et al., 2023): learns region embeddings by encoding human mobility time series as patch sequences with a Transformer encoder architecture.
- **TSLANet** (Eldele et al., 2024): learns region embeddings by capturing temporal and frequency patterns using adaptive convolution.
- **TimesURL** (Liu and Chen, 2024): learns region embeddings by applying self-supervised contrastive learning on augmented time-series data representations.

Spatiotemporal Forecasting Models. These models are trained on the source city and directly applied to the target city without transfer-specific designs.

- **STID** (Shao et al., 2022): transfers region embeddings learned by spatial and temporal identity embeddings over a purely MLP-based architecture.
- **STBP** (Liu and Zhang, 2026): transfers region embeddings by reusing spatiotemporal patterns stored in its scalable contextual pattern bank.

Region Transfer Learning Methods. These approaches explicitly address cross-city distribution shifts by transferring spatial or spatiotemporal knowledge.

- **TransGTR** (Jin et al., 2023): learns transferable region embeddings by reducing spatial discrepancies between source city and target city during transfer.
- **CrossST** (Liu and Zhang, 2025): learns transferable region embeddings by constructing pattern banks from the source city.
- **CoRE** (Chen et al., 2025): learns transferable region embeddings by aligning region representation manifolds across different cities.

LLM-based Urban Models. To further evaluate existing LLM-based explorations, we include:

- **UrbanGPT** (Li et al., 2024b): learns region embeddings by aligning external urban representations with the semantic space of a pre-trained large language model.
- **UrbanCLIP** (Yan et al., 2024): learns region embeddings by aligning satellite imagery with textual descriptions via contrastive learning objectives.

Overall, these baselines provide a comprehensive comparison across vector-based, spatiotemporal predictive, transfer-oriented, and LLM-based urban representation learning paradigms.

A.3 Experimental Setup and Implementations

A.3.1 Spatial-temporal Pre-training.

We conduct spatial-temporal pre-training using bike-sharing datasets from New York City and Chicago, where raw trip records are aggregated into hourly Inflow and Outflow sequences for each region. Regional mobility time series are segmented into overlapping patches of length 2016 with a sliding window stride of 168, and corresponding origin–destination (OD) matrices are constructed for each patch to facilitate heterogeneous subgraph sampling. For every target region, we sample a localized second-order neighborhood consisting of five first-order and five second-order neighbors to preserve balanced local topological contexts. The dataset is split into training, validation, and test sets with a ratio of 7:1:2. All experiments are optimized using the Adam optimizer with a fixed learning rate of 0.001, and early stopping is applied with a patience of 30 epochs during pre-training and 50 epochs during downstream evaluation. The final region representation dimension is fixed to 512 across all tasks.

A.3.2 LLM Backbone.

We employ Llama3.1-8B as the backbone model. During the instruction-tuning phase, we use a batch size of 16 and a learning rate of 1e-4 with decay, training the model for two epochs. In URBAN-PROTOS, the pretrained mobility embedding dimension is set to 512, while the final region representation dimension is configured as 4096. In the first stage of instruction tuning, all LLM parameters are frozen. In the second stage, we apply LoRA-based fine-tuning to selectively update a subset of parameters, ensuring that the model’s core language capabilities remain intact. During inference, to enhance stability and consistency in the

generated outputs, we set the temperature parameter to 0.01.

A.4 Illustration of Urban Prototypes

Urban prototypes are a set of semantic tokens that bridge the spatiotemporal context with the prior knowledge of LLMs, mapping heterogeneous urban signals into a unified semantic space. We organize them along three axes that are canonical in urban spatiotemporal modeling: *function* (land use and POI composition), *location* (how a region is positioned relative to and interacts with its surroundings), and *human flow* (finer-grained and largely implicit mobility dynamics). The concrete prototype words of each axis are obtained through an LLM-based multi-round iterative procedure. Taking function as an example, we feed the OpenStreetMap POI corpus to the LLM and prompt it to propose, revise, and filter candidates over several rounds, retaining ten prototypes that are semantically unambiguous and as mutually orthogonal as possible; the resulting prototypes are listed in Table 4. Since the procedure is corpus-driven, it is readily extensible: new city-related corpora can be plugged into the same loop to derive prototypes tailored to specific downstream tasks.

Sensitivity to prototype categories. To quantify the contribution of each prototype, we remove one category at a time under the CHI2NYC cross-city setting (Table 3). Discarding all prototypes yields the largest degradation across the four tasks, confirming that prototype-level semantics are indispensable for cross-city transfer. The reliance is further task-dependent: for function-related tasks (Business and Crime), dropping functional prototypes is more harmful than dropping human-flow ones, whereas the opposite holds for the flow-related Crash prediction. Nevertheless, removing any single category still outperforms removing all of them in most cases, indicating that the three axes provide complementary rather than redundant semantics. This observation also suggests task-adaptive weighting over prototype categories as a natural extension of our framework.

A.5 Transfer with a Distinct Source City

All cross-city experiments follow a strict data-isolation protocol: taking NYC2CHI as an example, both pretraining and instruction tuning consume only NYC data, and the target city is touched exclusively at inference time. To further stress-test

Table 3: Ablation on the three categories of urban prototypes under the CHI2NYC cross-city setting. Best results are in **bold** and the second best are underlined.

Variant	Business		Crime		Crash		311 Service	
	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2
w/o All Prototypes	24.19	0.46	97.02	0.21	9.52	0.17	396.45	0.14
w/o Function Prototypes	23.88	0.48	95.85	0.23	<u>9.17</u>	<u>0.20</u>	390.68	0.17
w/o Location Prototypes	23.38	0.50	92.65	0.24	9.23	0.18	<u>388.64</u>	<u>0.19</u>
w/o Human-flow Prototypes	<u>22.47</u>	<u>0.51</u>	88.92	0.29	9.43	0.17	395.23	0.15
Full (Ours)	22.26	0.53	<u>89.75</u>	<u>0.27</u>	9.09	0.20	387.07	0.20

Table 4: Categories and Examples of Urban Prototypes

Prototypes	Examples
Function	Residential area, Office district, Commercial area, Educational zone, Medical zone, Industrial zone, Recreational zone, Green space, Mixed-use area, Vacant land
Location	Urban core, Suburban area, Transport hub, Edge zone, City center, Residential fringe, Highway adjacent, Near railway station, Airport vicinity, Low-density outskirts
Human Flow	Morning peak commute, Evening peak commute, Weekday movement, Weekend activity flow, High transit flow, Low pedestrian flow, Inbound population flow, Outbound population flow, Commute patterns, Population concentration

transferability, we introduce Chengdu (CD) as an additional source city, which differs substantially from NYC and CHI in spatial layout, urban scale, and mobility rhythm, and is processed with the same pipeline. We transfer from CD to NYC and CHI, and report the RMSE results in Table 6, where each newly added setting is placed next to its counterpart in the main experiments.

Transferring from CD yields performance comparable to transferring between NYC and CHI across all four tasks, and the results remain stable regardless of which city serves as the source. Such stability indicates that Urban Prototypes abstract city-invariant regularities and align them with a shared semantic space, so that the transferable knowledge is not tied to the specific spatiotemporal patterns of any individual source city. Consequently, the effectiveness of our framework does not rely on the source and target cities being geographically or culturally similar.

A.6 High Quality Instruction Dataset.

To align spatiotemporal contexts with the LLM’s semantic space, we construct a high-quality instruction dataset for two-stage instruction tuning. Stage 1 focuses on comparative tasks to establish a foundational understanding of mobility patterns. The instruction dataset comprises positive and negative sample judgments based on key mobility statistics, such as human flow volume and centrality within the mobility network. Stage 2 involves generating detailed mobility descriptions using GPT-4o, where each region is characterized by multiple descriptive attributes, including net mobility flow, peak hours, deviations from citywide averages, weekday-weekend variations, and inferred semantic attributes. Table 5 provides representative examples of instructions from both stages.

A.7 Ablation on Instruction-Tuning Strategy.

We conduct an ablation study to evaluate the role of the proposed two-stage instruction-tuning strategy in *cross-city urban region representation learning*. We compare three model variants: (1) *w/o Tuning*, where the LLM backbone is used without instruction tuning; (2) *S1 Tuning*, which applies only Stage 1 instruction tuning; and (3) *S1+S2 Tuning*, which jointly employs both stages.

As shown in Table 7, instruction tuning consistently improves performance across all downstream tasks and both cities. Notably, applying Stage 1 alone already leads to clear gains over the untuned model, suggesting that the initial alignment objective effectively bridges spatiotemporal region representations with the semantic space of the LLM, which is essential for cross-city generalization. Further incorporating Stage 2 results in additional and consistent improvements, yielding the best performance in all evaluated settings. These results indicate that the staged instruction-tuning strategy progressively enhances the transferability

Table 5: Examples of Instruction Prompts for Spatiotemporal Alignment.

Example of stage 1 instruction

Input: Characterize the population mobility of regions A and B, which are sequentially represented in the embedding vector $\langle \text{region_embedding} \rangle \langle \text{region_embedding} \rangle$. Compare and determine which region is more central in the mobility network?

Output: Region B is more central in the mobility network than Region A.

Example of stage 2 instruction

Input: Given the human mobility embedding $\langle \text{region_embedding} \rangle$, describe the average population flow characteristics of this region.

Output: The region exhibits higher inflow and outflow rates compared to regional averages, particularly during peak hours. There is a concentration of inflow in the early hours, suggesting morning rush, with higher outflow during the afternoon and evening, indicating commuting and leisure activities. The inflow and outflow patterns indicate a dynamic, active urban environment with a strong flow of people in and out, possibly reflecting a busy commercial or tourist hub.

Table 6: Transfer performance (RMSE) with different source cities.

Setting	Business	Crime	Crash	311
CHI2NYC	22.26	89.75	9.09	387.07
CD2NYC	21.68	90.79	9.17	388.97
NYC2CHI	42.12	123.69	308.78	202.06
CD2CHI	41.97	125.03	310.63	201.97

Table 7: Ablation study on the Instruction-Tuning strategy.

City	Method	Business	Crime	Crash	311
CHI2NYC	w/o Tuning	24.08	95.71	9.47	397.63
	S1 Tuning	24.49	91.71	9.42	393.30
	S1+S2 Tuning	22.26	89.75	9.09	387.07
NYC2CHI	w/o Tuning	46.88	131.34	350.98	244.98
	S1 Tuning	43.45	124.79	320.46	214.78
	S1+S2 Tuning	42.12	123.69	308.78	202.06

and semantic coherence of urban region representations, thereby facilitating more robust cross-city urban analytics.

A.8 Representation Visualization

We visualize region representations from NYC and CHI by jointly projecting them into a two-dimensional space using t-SNE(van der Maaten and Hinton, 2008), as illustrated in Figure 5. This experiment compares ReCP and URBAN-PROTOS to examine their representation distributions under cross-city settings. For ReCP, region embeddings form structured and well-separated clusters within each city; however, representations from NYC and CHI are largely distributed in disjoint regions of the embedding space. This pattern indicates that, although intra-city distinctions are preserved,

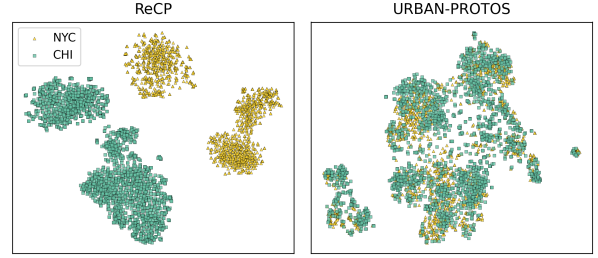


Figure 5: Cross-city visualization of urban region representations learned by different models.

the learned representations remain strongly city-dependent, limiting cross-city consistency. In contrast, URBAN-PROTOS learns region embeddings that exhibit clear cluster structures within each city, while sharing a common embedding space across cities. Regions from different cities are organized according to similar patterns rather than city identity, suggesting that URBAN-PROTOS captures higher-level urban semantics that are transferable across heterogeneous urban environments. We also conducted a detailed case study in Section 4.5.

A.9 Examples of LLM-based Inference

For URBAN-PROTOS, the large language model is employed as a semantic reasoning backbone to integrate prototype-aligned spatiotemporal representations with contextual urban attributes. To qualitatively examine the learned urban semantics, we analyze the model’s internal reasoning by decoding contextualized hidden states under a fixed analysis prompt, as illustrated in Table 8. Given a region, its aligned representation and textual attribute cues, i.e., the location and POI descriptions, are fed into

Table 8: Examples of URBAN-PROTOS’ Region Representation and Decoded Output.

Location Information: Region 95 is located at latitude 40.70530 and longitude -74.00609 in New York City. POI Information: Region 95 is a bustling area with a diverse range of Points of Interest (POIs) including various dining options such as restaurants, cafes, fast food joints, and pubs. It also features several financial institutions, shops offering services like hairdressing, cosmetics, and alcohol, as well as accommodations like hotels. Additionally, the region has health-related POIs like pharmacies and dentists, along with public services such as post offices and ATMs. Transport options are available with bus stops, and there are cultural attractions like art galleries and historic landmarks. Mobility Information: To better understand the region’s human mobility characteristics, we provide the corresponding human mobility embedding <code><region_embedding></code>. Task Prompt: Based on the provided information, generate a concise description of the region’s characteristics, functionality, and potential roles within the city.
Output Representation: <code><region_representation></code> which is derived from the last hidden states of URBAN-PROTOS. Decoded Output: The region functions as a densely populated commercial area with a wide range of amenities, including financial services, dining, healthcare, retail and accommodation facilities. It exhibits high mobility levels during peak hours, suggesting strong transportation demand and active population flow. Overall, the region functions as a central business district with substantial economic activity, supporting a substantial workforce and attracting visitors from surrounding areas.

the LLM, and the decoded responses are inspected to infer high-level properties such as functional roles, spatial centrality, and characteristic mobility patterns. These results are used solely for qualitative inspection rather than explicit generative outputs, since the actual output of our model is the region representation taken from the last hidden states, whereas the decoded text is reported only for interpretability. As shown in Table 8, the observed reasoning outcomes are consistent with the underlying mobility dynamics and regional context, providing qualitative evidence that URBAN-PROTOS captures meaningful and transferable urban semantics in the LLM latent space.