Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25)
Special Track on AI4Tech: AI Enabling Critical Technologies

HygMap: Representing All Types of Map Entities via Heterogeneous Hypergraph

Yifan Yang[1,4], **Jingyuan Wang**[1,2,3*], **Xie Yu**[1,4] and **Yibang Tang**[1,4]
[1]School of Computer Science and Engineering, Beihang University, Beijing, China
[2]MIIT Key Laboratory of Data Intelligence and Management, Beihang University, Beijing 100191, China
[3]School of Economics and Management, Beihang University, Beijing 100191, China
[4]Engineering Research Center of Advanced Computer Application Technology, Ministry of Education
{yfyang, jywang, yuxie_scse, tangyb}@buaa.edu.cn

Abstract

Maps are crucial for various smart city applications as a core component of city geographic information systems (GIS). Developing effective Map Entity Representation Learning methods can extract semantic information for downstream tasks like crime rate prediction and land use classification, with significant application potential. A map comprises three entity types: land parcels, road segments, and points of interest. Most existing methods focus on a single entity type, losing inter-entity relationships and weakening representation effectiveness for real-world applications. Thus, jointly modelling and representing multiple map entity types is essential. However, designing a unified framework is challenging due to map data's unstructured, complex, and heterogeneous nature. We propose a novel method, HygMap, to represent all map entity types. We model the map as a heterogeneous hypergraph, design an encoder for map entities, and introduce a hybrid self-supervised training scheme. This architecture comprehensively captures the heterogeneous relationships among map entities at different levels. Experiments on nine downstream tasks with two real-world datasets show that our framework outperforms all baselines, with good computational efficiency and scalability.

1 Introduction

Geographic Information Systems (GIS) are indispensable tools for urban management and daily life, supporting location-based applications such as intelligent transportation [Wang *et al.*2021a, Liu *et al.*2024], urban planning [Wang *et al.*2018, Wang *et al.*2017b], and emergency response [Wang *et al.*2021b, Wang *et al.*2017a]. A core component of these systems is *Map*, which comprises points of interest (POIs), road segments, and land parcels (regions). Given their critical role in urban management, developing effective methods to characterize and model map data in a generalizable manner is essential.

*Corresponding author

Traditional maps rely on specialised spatial-temporal data formats, such as Shapefile and GeoJSON, which store geographic coordinates as attributes for map entities. However, these formats cannot directly model relationships between entities, necessitating manual feature engineering for downstream tasks. This process is labour-intensive and limits the effectiveness of downstream applications due to insufficient data mining.

In recent years, deep neural network-based representation learning has achieved significant success in various domains, including text representation learning [Yan *et al.*2021] ,image representation learning [Chen *et al.*2020] and spatio-temporal representation learning [Ren *et al.*2021, Hettige *et al.*2024, Ji *et al.*2022]. As a critical form of unstructured data, map data has also benefited from representation learning. Emerging Map Entity Representation Learning methods [Han *et al.*2025, Jiang *et al.*2024, Cheng *et al.*2025] use deep learning to extract structural and semantic information from maps, converting it into generic representation vectors for downstream tasks such as traffic prediction [Ji *et al.*2025, Wang *et al.*2016], route planning [Wang and Wong2023, Wang *et al.*2021c], and land-use classification [Zhang *et al.*2020].

Maps are unified wholes comprising multiple interactive entity types. For instance, a commercial land parcel may house numerous POIs (e.g., stores, restaurants, entertainment venues), attracting tourists and boosting economic vitality, transportation demand, and traffic flow on nearby roads. Transportation hubs (e.g., train stations, subway stations, bus stops) connect to multiple roads, forming complex networks where high traffic volume significantly affects accessibility and flow in surrounding areas. These inter-entity relationships hold valuable structural and semantic information. However, existing map representation learning methods focusing on single-entity categories often lose such relationships, weakening representation effectiveness for real-world applications. Therefore, it is crucial to model and represent multiple map entity categories jointly.

However, designing a unified framework to represent multiple types of map elements is non-trivial. As a form of unstructured, highly complex, and heterogeneous data, Maps pose significant challenges for uniform representation. First, the entities in maps are diverse and heterogeneous. Land parcels, roads, and points of interest (POIs) have distinct data structures and attributes. Second, the relationships among

these entities are also heterogeneous across different entity levels. Previous work on map entity representation has focused on capturing different semantic information at each level. For example, at the land parcel level, the adjacency relationships of polygons are captured; at the road segment level, the connectivity of the road network and traffic transfer relationships are represented; and at the POI level, the semantic relationships implied by check-in sequences are revealed. Finally, at the overall map level, as illustrated in Fig. 1, multiple roads enclose a parcel, a road may be adjacent to many parcels, and multiple POIs are distributed within a parcel, forming a natural high-order topological relationship.

Driven by these, we propose a unified framework for representing **all types of map entities**, denoted as **Heterogeneous Hypergraph-based Map Representation Learning (HygMap)**, which unifies their diverse characteristics and complex relationships. Firstly, we put forward a map modelling method, which models the unstructured data into a map hypergraph structure composed of three types of nodes and four types of hyperedges. Then, we designed a map entity encoder for this graph structure. It includes the message passing among the same type of entities and covers the message passing between heterogeneous entities. Finally, we proposed a hybrid training strategy for the map entity encoder we have put forward, comprehensively capturing and balancing entities' relationships while ensuring the representations' robustness. Experimental results show that HygMap can achieve the optimal performance while also possessing excellent efficiency after we conduct extensive experiments on two real-world datasets and nine types of tasks. Compared with the 22 existing methods, our method is superior in all downstream tasks, with an average performance increase of over 7%, validating the effectiveness and versatility of HygMap in handling multiple categories of map entity representation learning challenges.

2 Related Work

POI Representation Learning. POI representation learning typically involves capturing users' mobility patterns and generating POI representations based on check-in data. Early approaches utilised generic word representation learning methods [Mikolov *et al.*2013a], and later researchers often used sequence models such as LSTM and Transformer to capture temporal patterns from POI check-in sequences and generate representations for predicting the next POI [Lin *et al.*2021, Gong *et al.*2023].

Road Segment Representation Learning. At first, researchers relied on topology-aware graph embedding methods [Perozzi *et al.*2014], but these methods have limitations in capturing complex road attributes and traffic semantics. Subsequently, more graph-based road network representation methods emerged. These methods consider topology and geospatial attribute information [Wu *et al.*2020, Chen *et al.*2021, Zhang and Long2023]. In the latest development, two self-supervised methods have been proposed, namely JCLRNT [Mao *et al.*2022] and SARN [Chang *et al.*2023].

Land Parcel (Region) Representation Learning. Early approaches utilised the skip-gram method to learn representations of parcels from sequences of human mobility data [Wang and Li2017]. Subsequent research has focused on modelling parcel features using multi-view data (e.g. geographic distance and human mobility data) [Zhang *et al.*2019]. More recently, researchers have explored the graph structure of parcels with multiple views for representation learning [Zhang *et al.*2020, Wu *et al.*2022]. In the latest study, the ReCP model [Li *et al.*2024] presents a self-supervised representation learning scheme incorporating cross-view comparison learning, showing promising results.

Previous studies represented map data at three levels with distinct methods, overlooking interactions among urban entities at different scales. CityFM [Balsebre *et al.*2024] is a pioneering work to represent map entities uniformly but overlooks their complex relationships. In contrast, our model uniformly represents three entity types via comprehensive map modelling, preserving map integrity and capturing inter-entity relationships.

3 Preliminaries

3.1 Definitions of Basic Map Entities

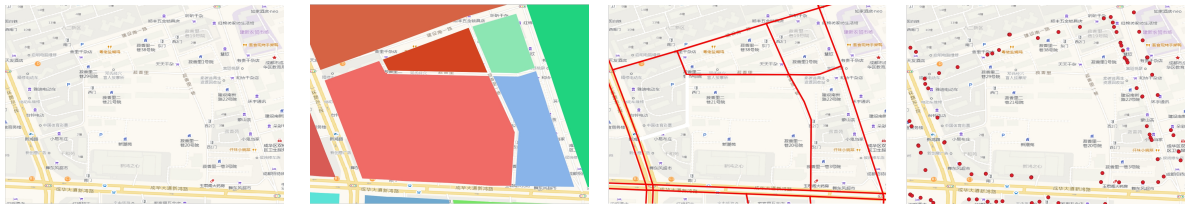
Definition 1 (Point of Interest)**.** *A point of interest is defined as a location in the city, such as a building or landmark (in Fig. 1(d)). A typical coordinate is $[x, y]$, with x and y being the geographic longitude and latitude, respectively. The POI set within a city can be denoted as $\mathcal{P} = \{p_1, p_2, \cdots, p_{N_p}\}$, where N_p is the number of POIs. For a specific POI $p_i \in \mathcal{P}$, its raw features are encoded as the vector $\boldsymbol{f}_{p_i}^{raw} \in \mathbf{R}^{D_P}$, where D_P is the dimension of the raw feature vectors for POIs. The raw feature of $\mathcal{P}$ is $\boldsymbol{F}_P^{raw} \in \mathbf{R}^{N_p \times D_P}$.*

Definition 2 (Road Segment)**.** *A road segment is a polyline of the road network (in Fig. 1(c)). A typical coordinate is $[[x_0, y_0], [x_1, y_1], \cdots]$, with x_i and y_i being the geographic longitude and latitude of the i-th point on the road segment, respectively. The road segment set is denoted as $\mathcal{S} = \{s_1, s_2, \cdots, s_{N_s}\}$, where N_s is the number of road segments. For a specific road segment $s_i \in \mathcal{S}$, its raw features are encoded as the vector $\boldsymbol{f}_{s_i}^{raw} \in \mathbf{R}^{D_S}$, where D_S is the dimension of the raw feature vectors for roads. The raw feature of $\mathcal{S}$ is $\boldsymbol{F}_S^{raw} \in \mathbf{R}^{N_s \times D_S}$.*

Definition 3 (Land Parcel)**.** *A land parcel is a polygon area divided by road segments (in Fig. 1(b)). A typical coordinate is $[[x_0, y_0], [x_1, y_1], \cdots, [x_0, y_0]]$, with x_i and y_i being the geographic longitude and latitude of the i-th point on the parcel boundary, respectively. The land parcel set is denoted as $\mathcal{R} = \{r_1, r_2, \cdots, r_{N_r}\}$, where N_r is the number of land parcels. For a specific land parcel $r_i \in \mathcal{R}$, its raw features are encoded as the vector $\boldsymbol{f}_{r_i}^{raw} \in \mathbf{R}^{D_R}$, where D_R is the dimension of the raw feature vectors for parcels. The raw feature of $\mathcal{R}$ is $\boldsymbol{F}_R^{raw} \in \mathbf{R}^{N_r \times D_R}$.*

3.2 Problem Statement

Given the map entity set $\mathcal{M} = \mathcal{S} \cup \mathcal{R} \cup \mathcal{P}$, map representational learning aims to encode each POI p_i, road segment s_i, and land parcel r_i to the dense vector $\boldsymbol{f}_{p_i} \in \mathbf{R}^D$, $\boldsymbol{f}_{s_i} \in \mathbf{R}^D$, and $\boldsymbol{f}_{r_i} \in \mathbf{R}^D$ within a unified representation space, respectively. Here, D is the embedding dimension. These embed-



(a) City Map (b) Land Parcels(Polygons) (c) Road Segments(Lines) (d) Points of Interest(Points)

Figure 1: City map with three types of entities.

dings are expected to integrate the essential features of $\mathcal{M}$, thereby supporting various downstream tasks.

4 Methods

The overall framework is shown in Figure 2. Firstly, we constructed a heterogeneous map hypergraph with three types of nodes and four types of hyperedges to model the map data. Then we proposed a map entity encoder with a hybrid training strategy which leverages both generative and contrastive training paradigms for robust representations.

4.1 Map Hypergraph Builder

The graph structure is suitable for modelling map entities and their relationships, with entities as nodes and relationships as edges. Specifically, relationships among map entities exhibit two key characteristics: *i) Relationships are heterogeneous and more complex than geographic adjacency. ii) Relationships do not have to be one-to-one relationships; they can be high-order.* Therefore, we designed a hypergraph-based modelling approach to model the map as the map hypergraph. The map hypergraph we designed encompasses three types of heterogeneous entities as nodes and constructs four high-order relationships as hyperedges. Specifically, high-order relationships are concluded from different views, such as explicit connections (geographic view) and implicit connections (functional view, mobility view).

Geographic View. From the spatial topological perspective of the map, the relationships among different types of entities exhibit a hierarchical structure, which emphasizes the many - to - many dependencies. Specifically, a land parcel may be enclosed by multiple road segments. Meanwhile, a road can be adjacent to numerous parcels, and many Points of Interest (POIs) are located within a land parcel. Therefore, from the geographical view, we will construct two types of hyperedges, "located" and "enclose".

For POIs and land parcels, we take all the POIs within a land parcel and the land parcel itself together as a hyperedge of the type "located". To reflect the importance and connection strength of the hyperedge, we normalize the POI density within the parcel and use it as the weight of the "located" hyperedge. $\mathcal{E}^{located}$ denotes the "located" hyperedge set and $w^{located}_{\{p_{i_1},p_{i_2}....,r_j\}}$ denotes the weight of the "located" hyperedge that encompasses the entity set $\{p_{i_1}, p_{i_2}...., r_j\}$. The calculation process is expressed as follows

$$w^{located}_{\{p_{i_1},p_{i_2}....,r_j\}} = \frac{\text{count}\left(\{p_{i_1}, p_{i_2}....\}\right)}{\text{area}\left(r_j\right)}. \tag{1}$$

For all located hyperedge weight set $\mathcal{W}^{located}$, we normalize their weight into $[0, 1]$ by min-max normalization.

For roads and land parcels, we consider all the road segments that enclose a land parcel and the land parcel itself as a hyperedge of the "enclose" type. The weight of the hyperedge is calculated based on the average geographical distance between the two types of entities. The closer the distance between the two entities, the stronger their connection. $\mathcal{E}^{enclose}$ denotes the "enclose" hyperedge set and $w^{enclose}_{\{s_{i_1},s_{i_2}....,r_j\}}$ denotes the weight of the "enclose" hyperedge that encompasses the entity set $\{s_{i_1}, s_{i_2}...., r_j\}$. The calculation process is expressed as the inverse of the geographical distance, which is

$$w^{enclose}_{\{s_{i_1},s_{i_2}....,r_j\}} \frac{1}{\sum_{s\in\{s_{i_1},s_{i_2}...\}} \text{dist}\left(s, r_j\right)/\left|\{s_{i_1}, s_{i_2}...\}\right| + \epsilon}. \tag{2}$$

where $\text{dist}\left(s, r_j\right)$ represents the Euclidean distance between the centroid of road s and land parcel r_j. ϵ is a small positive constant to avoid division by zero. For all enclose hyperedge weight set $\mathcal{W}^{enclose}$, we also normalize their weight into $[0, 1]$ by min-max normalization.

Functional View. The basic functional unit in a city is the Point of Interest (POI). The functional view captures the functional similarity between two entities, extending beyond spatial proximity. For a POI, we first extract information from its name to serve as the functional semantics. Specifically, we use a pre-trained BERT model to encode the name of the POI to obtain the functional vector, denoted as $\boldsymbol{p}^{fun}$. Then, to reflect the functional semantic similarity among POIs, we perform clustering [Ester *et al.*1996] on the functional vectors to obtain several functional semantic clusters. We regard each functional semantic cluster as a function hyperedge, which is

$$\mathcal{E}^{function} = \text{Cluster}\left(\boldsymbol{p}^{fun}\right), \tag{3}$$

where $\mathcal{E}^{function}$ denotes the "function" hyperedge set. $w^{function}_{\{p_{i_1},p_{i_2}....,p_{i_n}\}}$ denotes the weight of the "function" hyperedge that encompasses the POI set $\{p_{i_1}, p_{i_2}...., p_{i_n}\}$, which is calculated by functional vector similarity. Specifically, the calculation process is

$$w^{function}_{\{p_{i_1},p_{i_2}....,p_{i_n}\}} = \frac{\sum_{p_j,p_k\in\{p_{i_1},p_{i_2}....,p_{i_n}\}} \text{sim}\left(\boldsymbol{p}^{fun}_j, \boldsymbol{p}^{fun}_k\right)}{n\left(n-1\right)}, \tag{4}$$

where $\text{sim}(\cdot,\cdot)$ is the cosine similarity.

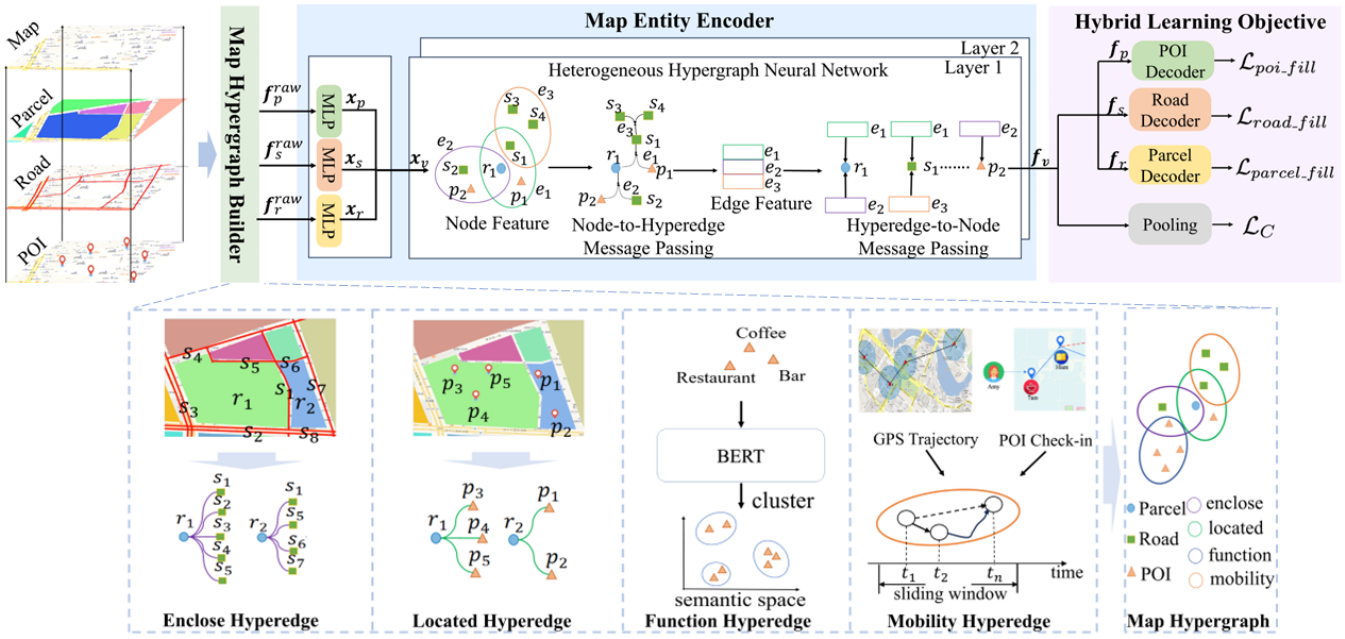


Figure 2: The overall framework of HygMap.

Mobility View. Urban human mobility patterns reflect the relationships among map entities. We utilise historical GPS trajectory data to build edges for the mobility view. Unlike prior studies that only capture binary mobility patterns from trajectories, the property of hypergraphs to capture higher-order relationships allows us to capture mobility patterns and entity co-occurrence relationships over longer distances.

First, we apply the map-matching algorithm [Yang and Gidofalvi2018] to align trajectories with corresponding road segments for road entities. Then, we sample the trajectory data using a fixed-length sliding window. Entities within each window form a mobility pattern set, and all such sets are considered mobility-type hyperedges, denoted as $\mathcal{E}^{mobility}$. Finally, we use the frequency of these mobility sets in historical trajectories as hyperedge weights. Specifically, $w^{mobility}_{\{s_{i_1},s_{i_2}\ldots,s_{i_n}\}}$ denotes the weight of the hyperedge representing the continuous road transfer sequence $\{s_{i_1}, s_{i_2}, \cdots, s_{i_n}\}$, which is calculated as

$$w^{mobility}_{\{s_{i_1},s_{i_2}\ldots,s_{i_n}\}} = \frac{\text{count}\left(s_{i_1} \rightarrow s_{i_2} \ldots \rightarrow s_{i_n}\right)}{\text{count}\left(s_i \rightarrow \ldots\right)}, \quad (5)$$

where $\text{count}\left(s_{i_1} \rightarrow s_{i_2} \ldots \rightarrow s_{i_n}\right)$ represents the statistical count of transfers along the entities in the current hyperedge, and $\text{count}\left(s_i \rightarrow \ldots\right)$ denotes the the statistical count of transfers starting from s_{i_1}. The mobility pattern of a land parcel is shaped by its nearest road segments. Specifically, each road segment is mapped to its closest land parcel. Thus, road-based trajectories are converted to parcel-based trajectories. For POIs, we employ the consecutive check-in sequences as POI-based trajectories and follow the similar calculation approach as road.

Definition 4 (Map Hypergraph)**.** *The map hypergraph is defined as* $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{W}, \boldsymbol{F}^{raw}\}$*, where* $\mathcal{V} = \mathcal{P} \cup \mathcal{S} \cup \mathcal{R}$ *is the vertex set composed of POI, road segment and parcel sets,* v_i *is the* i*-th vertex.* $\mathcal{E} = \mathcal{E}^{located} \cup \mathcal{E}^{enclose} \cup \mathcal{E}^{function} \cup \mathcal{E}^{mobility}$ *is the hyperedge set, which includes four types of hyperedges.* e_j *is the* j*-th hyperedge.* $\mathcal{W} = \mathcal{W}^{located} \cup \mathcal{W}^{enclose} \cup \mathcal{W}^{function} \cup \mathcal{W}^{mobility}$ *is the hyperedge weight set,* w_j *is the weight of* e_j*.* $\boldsymbol{F}^{raw} = \boldsymbol{F}^{raw}_P \cup \boldsymbol{F}^{raw}_S \cup \boldsymbol{F}^{raw}_R$ *is the raw feature vectors for all map entities. We denote* $\phi() : \mathcal{V} \rightarrow \{'poi','road','parcel'\}$ *as the type mapping function of the node, and* $\psi() : \mathcal{E} \rightarrow \{'located','enclose','function','mobility'\}$ *as the type-mapping function of the hyperedge.*

4.2 Map Entity Encoder

After abstracting the map into the map hypergraph, we design a map entity encoder for this structure to represent the three types of entities as vectors. The encoder is divided into two submodules. We first use a Raw Entity Feature Encoder to encode different features for heterogeneous entities and then use an HGNN for the heterogeneous map hypergraph to comprehensively capture the complex relationships and semantics on the map.

Raw Entity Feature Encoder. As defined in Definition 1, 2 and 3, each entity has a raw feature vector, *i.e.,* $\boldsymbol{f}^{raw}_p \in \mathbb{R}^{D_P}$, $\boldsymbol{f}^{raw}_s \in \mathbb{R}^{D_S}$ and $\boldsymbol{f}^{raw}_r \in \mathbb{R}^{D_R}$, respectively. Our module first encodes these raw features as dense vectors through a linear embedding layer. Then, for the poi p_i, segment s_i and the parcel r_i, we denote their raw feature embedding vectors as$\{\boldsymbol{n}_1, \ldots, \boldsymbol{n}_{D_P}\}$, $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{D_S}\}$ and $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{D_R}\}$. Then, we concatenate the raw feature embedding vectors of an entity as a long vector and adopt a multilayer perceptron network to compress the long vectors into a dense entity fea-

ture embedding vector as

$$\begin{aligned}\boldsymbol{x}_{p_i} &= \mathrm{MLP}\left(\boldsymbol{n}_1 \| \ldots \| \boldsymbol{n}_{D_P}\right), \\ \boldsymbol{x}_{s_i} &= \mathrm{MLP}\left(\boldsymbol{u}_1 \| \ldots \| \boldsymbol{u}_{D_S}\right), \\ \boldsymbol{x}_{r_i} &= \mathrm{MLP}\left(\boldsymbol{v}_1 \| \ldots \| \boldsymbol{v}_{D_R}\right),\end{aligned} \tag{6}$$

$\boldsymbol{x}_{p_i} \in \mathbb{R}^D$, $\boldsymbol{x}_{s_i} \in \mathbb{R}^D$ and $\boldsymbol{x}_{r_i} \in \mathbb{R}^D$ are the dense embedding vectors for the poi p_i, segment s_i and the parcel r_i. We set embedding vectors of all map entities with the uniform dimension D.

Heterogeneous Hypergraph Neural Network. As defined in Def. 4, our map hypergraph has three types of heterogeneous nodes and four types of heterogeneous hyperedges. We take $\boldsymbol{x}_v = \boldsymbol{x}_p \cup \boldsymbol{x}_s \cup \boldsymbol{x}_r$ as the initial feature of the node, and sequentially go through the two processes of Node-to-Hyperedge Message Passing and Hyperedge-to-Node Message Passing, in which we explicitly model the node and hyperedge heterogeneity, and finally output three entity representations.

Node-to-Hyperedge Message Passing. This process involves aggregating all node features contained within a hyperedge to obtain a representation of the hyperedge. Specifically, for node $v_i \in \mathcal{V}$, we first project the node features according to the node type to get the message to be passed, and then do the aggregation of the messages of all the nodes within a hyperedge $e_j \in \mathcal{E}$ based on the weight and type of the hyperedge to get the representation of the hyperedge:

$$\begin{aligned}\boldsymbol{m}_{v_i} &= \boldsymbol{W}_{\phi(v_i)} \boldsymbol{x}_{v_i}, \\ \boldsymbol{m}_{e_j} &= w_j \cdot \boldsymbol{W}_{\psi(e_j)} \frac{1}{|\mathcal{N}_j|} \sum_{v_k \in \mathcal{N}_j} \boldsymbol{m}_{v_k},\end{aligned} \tag{7}$$

where $\mathcal{N}_j$ is the set of nodes contained in e_j, $\boldsymbol{W}_{\phi(v_i)}$, $\boldsymbol{W}_{\psi(e_j)} \in \mathbb{R}^{D \times D}$ are learnable parameters, which are distinct for different categories of nodes and hyperedges.

Hyperedge-to-Node Message Passing. This step assigns the hyperedge representations obtained from the aggregation in the previous step to nodes and obtains the final node representations as map entity representations. Specifically for a node, all the hyperedge representations in which the node participates are aggregated and weighted according to the attention score:

$$\begin{aligned}\boldsymbol{q}_{v_i} = \boldsymbol{W}_q \boldsymbol{m}_{v_i}, \;\; \boldsymbol{k}_{e_j} = \boldsymbol{W}_k \boldsymbol{m}_{e_j}, \;\; \boldsymbol{v}_{e_j} = \boldsymbol{W}_v \boldsymbol{m}_{e_j}, \\ a_{ij} = \frac{\exp\left(\boldsymbol{q}_{v_i}^\top \boldsymbol{k}_{e_j} / \sqrt{D}\right)}{\sum_{e_{j'} \in \mathcal{H}_i} \exp\left(\boldsymbol{q}_{v_i}^\top \boldsymbol{k}_{e_{j'}} / \sqrt{D}\right)}, \\ \boldsymbol{f}_{v_i} = r \sum_{e_j \in \mathcal{H}_i} a_{ij} \boldsymbol{v}_{e_j} + (1 - r) \boldsymbol{x}_{v_i},\end{aligned} \tag{8}$$

$\mathcal{H}_i$ is the neighbor hyperedge set of node v_i, where $\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v \in \mathbb{R}^{D \times D}$ are learnable parameter matrices, r is the weight of the residual connection.

4.3 Hybrid Learning Objective

Entity Level: Entity Mask Fill Task. Three types of entities are distributed on the map and serve as the context for each other. Therefore, we hope to utilize the training paradigm of Masked Autoencoders (MAE) on the map hypergraph to infer the properties of certain entities based on the map context. We propose a hypergraph generative task named the entity mask fill task.

The Entity mask fill task starts by randomly masking some of the segments and POIs at a mask ratio for each parcel. Then we mask the parcel based on the degree of the hyperedge in the map hypergraph. We tend to mask parcels with lower degrees. For the masked POI, segment and parcel, we fill a special vector $[MASK_P]$, $[MASK_S]$ and $[MASK_R]$ that can be learned. Then, we use an MLP module as a decoder and try to recover the masked entity features from the map entity encoder's results. For a given POI, road and parcel, we denote the decoder output features as $\boldsymbol{f}_p^D, \boldsymbol{f}_s^D$ and $\boldsymbol{f}_r^D$. Our optimization objective is to maximize the similarity between reconstructed and original features, and we use scaled cosine error (SCE) to construct the loss function,

$$\begin{aligned}\mathcal{L}_{road\_fill} &= \frac{1}{|\tilde{\mathcal{S}}|} \sum_{s_i \in \tilde{\mathcal{S}}} \left(1 - \frac{\boldsymbol{f}_{s_i}^{D\top} \boldsymbol{f}_{s_i}^{raw}}{||\boldsymbol{f}_{s_i}^D|| \cdot ||\boldsymbol{f}_{s_i}^{raw}||}\right)^2, \\ \mathcal{L}_{poi\_fill} &= \frac{1}{|\tilde{\mathcal{P}}|} \sum_{p_i \in \tilde{\mathcal{P}}} \left(1 - \frac{\boldsymbol{f}_{p_i}^{D\top} \boldsymbol{f}_{p_i}^{raw}}{||\boldsymbol{f}_{p_i}^D|| \cdot ||\boldsymbol{f}_{p_i}^{raw}||}\right)^2, \\ \mathcal{L}_{parcel\_fill} &= \frac{1}{|\tilde{\mathcal{R}}|} \sum_{r_i \in \tilde{\mathcal{R}}} \left(1 - \frac{\boldsymbol{f}_{r_i}^{D\top} \boldsymbol{f}_{r_i}^{raw}}{||\boldsymbol{f}_{r_i}^D|| \cdot ||\boldsymbol{f}_{r_i}^{raw}||}\right)^2,\end{aligned} \tag{9}$$

where $\tilde{\mathcal{P}}, \tilde{\mathcal{S}}$ and $\tilde{\mathcal{R}}$ are the sets of POIs, segments and parcels being masked.

City Level: City Contrastive Learning. We want the representations we obtain to fully learn the semantics at the entity level while still maintaining global consistency at the city level. We introduce a visual global city node to the map hypergraph to achieve this. We assume the city node contains all entities of the map hypergraph. The representation of the city node is calculated as a average pooling of the entity representations, *i.e.,* $\boldsymbol{f}_c = \frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \boldsymbol{f}_{v_i}$.

City contrastive learning aims to maximize the mutual information between the city node representation and parcel representations. We consider the city representation and an entity representation as a pair of positive samples, *i.e.,* $< \boldsymbol{f}_c, \boldsymbol{f}_{v_i} >$. Besides, we construct a set of fake representations for the entities as negative samples. Specifically, we corrupt the raw feature matrix $\boldsymbol{F}^{raw}$ for entities by randomly permuting its rows, resulting in a set of fake entities. Then we feed the fake entities into the map entity encoder to get three types of entity representations, denoted as $\tilde{\boldsymbol{f}}_{v_i}$. The representation pairs $< \boldsymbol{h}_c, \tilde{\boldsymbol{f}}_{v_i} >$ are used as negative samples. The loss function for city contrastive learning is defined as

$$\begin{aligned}\mathcal{L}_C = -\Bigg(&\frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{R}} \log \mathcal{D}\left(\boldsymbol{f}_{v_i}, \boldsymbol{f}_c\right) \\ &+ \frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \log\left(1 - \mathcal{D}\left(\tilde{\boldsymbol{f}}_{v_i}, \boldsymbol{f}_c\right)\right)\Bigg),\end{aligned} \tag{10}$$

where $\mathcal{D}(\boldsymbol{f}_{v_i}, \boldsymbol{f}_c) = \text{Sigmoid}(\boldsymbol{f}_{v_i}^\top \boldsymbol{W}_C \boldsymbol{f}_c)$ and $\boldsymbol{W}_C$ is a learnable parameter matrix.

The overall learning objective is

$$\mathcal{L} = \lambda_1 \mathcal{L}_{road\_fill} + \lambda_2 \mathcal{L}_{poi\_fill} + \lambda_3 \mathcal{L}_{parcel\_fill} + \lambda_4 \mathcal{L}_C, \tag{11}$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are hyperparameters.

5 Experiments

We conduct comprehensive experiments to evaluate the performance of the HygMap. We chose two real-world cities, San Francisco and Porto, as the datasets for our experiments. Our experiments are completed depending on the VecCity [Zhang *et al.*2024]. The scale, source of the datasets, code, and more implementation details will be provided in https://github.com/yyf-buaa/HygMap.

We conduct experiments on the following three types of downstream tasks to evaluate the learned representations. The road segment-based tasks are average speed prediction, traffic flow prediction and OD flow prediction. The land parcel-based tasks are land Parcel classification, traffic flow prediction and OD flow prediction. The POI-based tasks are POI classification, next POI prediction and trajectory user link.

Our baselines fall into three categories:(1) Land parcel representation learning methods, consisting of HDGE [Wang and Li2017], ZE-Mob [Yao *et al.*2018], GMEL [Liu *et al.*2020], MVURE [Zhang *et al.*2020], MGFN [Wu *et al.*2022], ReMVC [Zhang *et al.*2022], HREP [Zhou *et al.*2023], ReCP [Li *et al.*2024];(2) Road network representation learning methods, including RN2Vec [Wang *et al.*2021d], HRNR [Wu *et al.*2020], Toast [Chen *et al.*2021], START [Jiang *et al.*2023], JCLRNT [Mao *et al.*2022], SARN [Chang *et al.*2023] and HyperRoad [Zhang and Long2023]; (3) POI representation learning methods: Skip-gram [Mikolov *et al.*2013b], Teaser [Zhao *et al.*2017], POI2Vec [Feng *et al.*2017], HIER [Shimizu *et al.*2020], Tale [Wan *et al.*2021], CTLE [Lin *et al.*2021], CACSR [Gong *et al.*2023];

5.1 Performance Comparison

The experimental results of the land parcel-based, road segment-based, and poi-based tasks are shown in Tab. 1. Although the baseline methods exhibit good performance, our proposed method outperforms all of them in all tasks and metrics on the two datasets, as confirmed by the Student's t-test at a significance level of 0.01. Our average improvement across all tasks and datasets is over 7%. The outstanding performance can be attributed to the ability of our model to integrate representations of different types of entities in a unified manner. Our proposed map hypergraph models the relationships as high-order interactions between heterogeneous entities, effectively capturing the intricate interrelationships among city map entities, enabling comprehensive representation learning. Additionally, by incorporating information from the geographic, functional, and mobility views, our model takes advantage of multiple perspectives. This comprehensive understanding of entity relationships endows our model's remarkable ability to capture diverse entity information. Finally, unlike most SOTA models that adopt contrastive learning methods, our model integrates contrastive and generative learning at both the city and entity levels. This enables our model to learn more comprehensive, rich, and robust representations, demonstrating its superiority in handling complex urban map data.

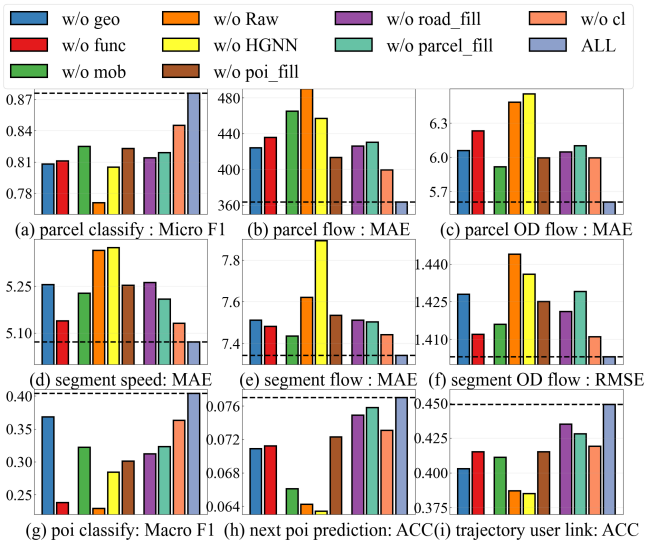


Figure 3: Ablation study on the San Francisco dataset.

5.2 Ablation Study

We conduct ablation experiments on all datasets and report the average results in Fig. 3. Due to limited space, we only present the results on the San Francisco dataset. Removing any module leads to a decline in the performance of entity representation in each task. The most significant drop occurs when the Raw Encoder and Hypergraph Neural Network (HGNN) are removed, as they capture the different characteristics of heterogeneous entities and complex heterogeneous relationships in the hypergraph structure. Removing any of the three views in the map hypergraph reduces performance, confirming the necessity of using a heterogeneous hypergraph. For the hybrid training objective, removing any entity mask filling task affects its related tasks and impacts other entity types, further demonstrating the interconnection among different types of map entities. Removing contrastive learning reduces the global consistency of the representation, making the model overly focus on the local information of nodes, leading to performance degradation.

5.3 Parameter Sensitivity

We conduct a parameter sensitivity analysis for encoder layers L and embedding dimension D on the San Francisco datasets. We present the results of 3 tasks in Fig 4. We observe that as D and L increase, the expressive power of the model becomes stronger, and the performance on various tasks all rises. However, as D and L continue to increase, the performance on all tasks decreases. This is because, under the premise of a fixed amount of data, an overly complex model will overfit. Therefore, it is necessary to select a model scale corresponding to the scale of the dataset.

Data	Porto						San Francisco					
Land Parcel-based Tasks	Classification		Flow Prediction		OD Prediction		Classification		Flow Prediction		OD Prediction	
Metrics / Models	Mi-F1	Ma-F1	MAE	RMSE	MAE	RMSE	Mi-F1	Ma-F1	MAE	RMSE	MAE	RMSE
HDGE	0.553	0.554	5523	6402	33.42	50.32	0.770	0.507	511.8	926.4	7.397	21.94
ZEMob	0.456	0.436	5532	6423	35.31	51.44	0.791	0.621	509.1	910.5	7.837	23.48
GMEL	0.607	0.604	5503	6398	36.42	52.31	0.789	0.654	514.0	916.7	7.377	22.29
MVURE	0.649	0.646	5244	5924	31.95	47.41	0.780	0.595	511.2	913.1	7.196	22.73
ReMVC	0.784	0.780	5435	6341	32.49	48.35	0.794	0.558	444.7	852.3	7.007	22.60
HREP	0.803	0.794	5034	5942	31.53	49.02	0.824	0.692	490.2	835.4	6.751	21.98
ReCP	0.813	0.810	4978	5750	30.32	48.77	0.836	0.709	438.4	799.3	6.154	21.03
HygMap	**0.842**	**0.841**	**4788**	**5440**	**29.62**	**45.79**	**0.876**	**0.742**	**363.5**	**765.8**	**5.605**	**19.61**
Improve	3.57%	3.83%	3.81%	5.39%	2.32%	5.30%	4.78%	4.65%	17.10%	4.19%	8.92%	6.75%
Road Segment-based Tasks	Speed Prediction		Flow Prediction		OD Prediction		Speed Prediction		Flow Prediction		OD Prediction	
Metrics / Models	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
RN2Vec	4.580	8.196	25.23	86.23	0.743	1.525	2.978	5.897	8.423	38.80	0.459	2.235
HRNR	4.557	8.167	24.90	86.02	0.814	1.553	2.560	5.474	8.341	38.02	0.402	2.187
Toast	4.554	7.983	24.52	85.54	0.623	1.505	2.603	5.395	8.245	38.42	0.398	2.094
START	4.475	7.928	23.88	83.84	0.546	1.458	2.500	5.433	7.945	37.04	0.339	1.805
JCLRNT	4.334	7.822	23.04	83.07	0.723	1.511	2.584	5.295	7.972	36.92	0.348	1.935
HyperRoad	4.395	7.899	23.18	83.90	0.549	1.456	2.536	5.350	8.074	36.82	0.298	1.554
SARN	4.372	7.849	23.82	84.51	0.543	1.464	2.531	5.403	7.823	36.94	0.205	1.533
HygMap	**4.156**	**7.560**	**22.05**	**81.13**	**0.529**	**1.411**	**2.411**	**5.071**	**7.342**	**35.64**	**0.172**	**1.403**
Improve	4.11%	3.35%	4.29%	2.33%	2.58%	3.09%	3.56%	4.23%	6.15%	3.21%	16.1%	8.48%
POI-based Tasks	Classification		Next Poi Prediction		Trajectory User Link		Classification		Next Poi Prediction		Trajectory User Link	
Metrics / Models	Mi-F1	Ma-F1	ACC	F1	ACC	F1	Mi-F1	Ma-F1	ACC	F1	ACC	F1
Skipgram	0.251	0.103	0.096	0.044	0.367	0.254	0.243	0.205	0.054	0.019	0.343	0.185
Teaser	0.249	0.103	0.884	0.044	0.338	0.293	0.311	0.194	0.050	0.018	0.378	0.195
POI2Vec	0.255	0.111	0.119	0.052	0.365	0.272	0.319	0.174	0.049	0.014	0.301	0.127
HIER	0.289	0.177	0.128	0.054	0.401	0.313	0.314	0.201	0.050	0.018	0.357	0.186
Tale	0.306	0.219	0.130	0.058	0.445	0.351	0.359	0.269	0.066	0.028	0.440	0.260
CTLE	0.331	0.233	0.132	0.056	0.414	0.331	0.345	0.256	0.632	0.229	0.432	0.259
CACSR	0.346	0.232	0.159	0.082	0.475	0.393	0.315	0.222	0.064	0.022	0.425	0.225
HygMap	**0.395**	**0.273**	**0.181**	**0.088**	**0.515**	**0.413**	**0.404**	**0.315**	**0.077**	**0.030**	**0.449**	**0.296**
Improve	14.16%	14.65%	13.68%	7.04%	8.43%	5.14%	12.53%	14.60%	16.77%	6.34%	2.23%	13.78%

Table 1: Performance on Downstream Tasks.

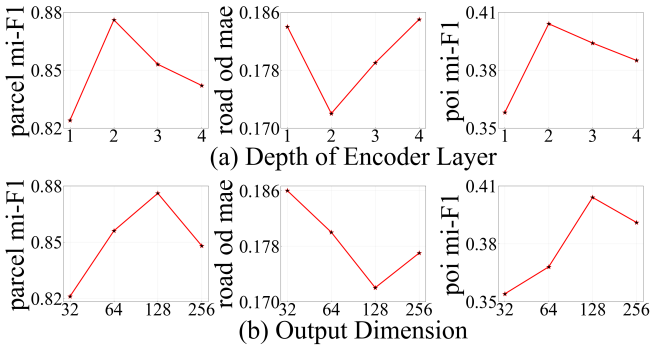


Figure 4: Parameter Sensitivity Result on the San Francisco dataset.

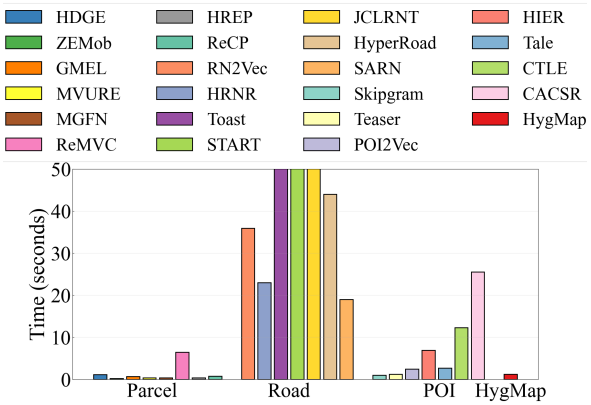


Figure 5: Model training time on San Francisco dataset.

5.4 Model Efficiency and Scalability

Fig. 5 shows the training time results. The land parcel representation models have the shortest training time because land parcels are the geographical entities with the smallest quantity and encompass the other two types of entities. Road segment representation models typically require a longer training time due to the relatively large number of road segments. Moreover, models such as in START, JCLRNT, utilize both GNNs to model the road network and Transformers to perform sequence modeling on trajectory data based on road segments incurs significant time consumption. The training time of HygMap is slightly slower than that of some land parcel representation models but faster than the POI and road segments representation models. Moreover, our model provides a unified representation for all three types of map entities, whereas the baseline models can only represent one type.

6 Conclusion and Future Work

We propose HygMap, a novel framework for uniformly representing all types of map entities through heterogeneous hypergraph modeling. Extensive experiments demonstrate its robust representation ability and efficiency across three map entity type. Future work may focus on enriching data modalities by integrating supplementary sources (e.g., satellite imagery and social media) to further enhance urban entity representations. HygMap can also support diverse heterogeneous data types in the future as a tokenizer module for spatio-temporal foundation models [Yu *et al.*2024, Yuan *et al.*2024].

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 72222022, 72171013, 72242101).

References

[Balsebre *et al.*, 2024] Pasquale Balsebre, Weiming Huang, Gao Cong, and Yi Li. City foundation models for learning general purpose representations from openstreetmap. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 87–97, 2024.

[Chang *et al.*, 2023] Yanchuan Chang, Egemen Tanin, Xin Cao, and Jianzhong Qi. Spatial structure-aware road network embedding via graph contrastive learning. In *EDBT*, pages 144–156. OpenProceedings.org, 2023.

[Chen *et al.*, 2020] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[Chen *et al.*, 2021] Yile Chen, Xiucheng Li, Gao Cong, Zhifeng Bao, Cheng Long, Yiding Liu, Arun Kumar Chandran, and Richard Ellison. Robust road network representation learning: When traffic patterns meet traveling semantics. In *CIKM*, pages 211–220. ACM, 2021.

[Cheng *et al.*, 2025] Jiawei Cheng, Jingyuan Wang, Yichuan Zhang, Jiahao Ji, Yuanshao Zhu, Zhibo Zhang, and Xiangyu Zhao. Poi-enhancer: An llm-based semantic enhancement framework for poi representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 11509–11517, 2025.

[Ester *et al.*, 1996] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231. AAAI Press, 1996.

[Feng *et al.*, 2017] Shanshan Feng, Gao Cong, Bo An, and Yeow Meng Chee. Poi2vec: Geographical latent representation for predicting future visitors. In *AAAI*, pages 102–108. AAAI Press, 2017.

[Gong *et al.*, 2023] Letian Gong, Youfang Lin, Shengnan Guo, Yan Lin, Tianyi Wang, Erwen Zheng, Zeyu Zhou, and Huaiyu Wan. Contrastive pre-training with adversarial perturbations for check-in sequence representation learning. In *AAAI*, pages 4276–4283. AAAI Press, 2023.

[Han *et al.*, 2025] Chengkai Han, Jingyuan Wang, Yongyao Wang, Xie Yu, Hao Lin, Chao Li, and Junjie Wu. Bridging traffic state and trajectory for dynamic road network and trajectory representation learning. *arXiv preprint arXiv:2502.06870*, 2025.

[Hettige *et al.*, 2024] Kethmi Hirushini Hettige, Jiahao Ji, Shili Xiang, Cheng Long, Gao Cong, and Jingyuan Wang. Airphynet: Harnessing physics-guided neural networks for air quality prediction. In *ICLR*. OpenReview.net, 2024.

[Ji *et al.*, 2022] Jiahao Ji, Jingyuan Wang, Junjie Wu, Boyang Han, Junbo Zhang, and Yu Zheng. Precision cityshield against hazardous chemicals threats via location mining and self-supervised learning. In *KDD*, pages 3072–3080. ACM, 2022.

[Ji *et al.*, 2025] Jiahao Ji, Wentao Zhang, Jingyuan Wang, and Chao Huang. Seeing the unseen: Learning basis confounder representations for robust traffic prediction. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pages 577–588, 2025.

[Jiang *et al.*, 2023] Jiawei Jiang, Dayan Pan, Houxing Ren, Xiaohan Jiang, Chao Li, and Jingyuan Wang. Self-supervised trajectory representation learning with temporal regularities and travel semantics. In *ICDE*. IEEE, 2023.

[Jiang *et al.*, 2024] Jiawei Jiang, Yifan Yang, Jingyuan Wang, and Junjie Wu. Jointly learning representations for map entities via heterogeneous graph contrastive learning, 2024.

[Li *et al.*, 2024] Zechen Li, Weiming Huang, Kai Zhao, Min Yang, Yongshun Gong, and Meng Chen. Urban region embedding via multi-view contrastive prediction. In *AAAI*, pages 8724–8732. AAAI Press, 2024.

[Lin *et al.*, 2021] Yan Lin, Huaiyu Wan, Shengnan Guo, and Youfang Lin. Pre-training context and time aware location embeddings from spatial-temporal trajectories for user next location prediction. In *AAAI*, pages 4241–4248. AAAI Press, 2021.

[Liu *et al.*, 2020] Zhicheng Liu, Fabio Miranda, Weiting Xiong, Junyan Yang, Qiao Wang, and Cláudio T. Silva. Learning geo-contextual embeddings for commuting flow prediction. In *AAAI*, pages 808–816. AAAI Press, 2020.

[Liu *et al.*, 2024] Zehua Liu, Jingyuan Wang, Zimeng Li, and Yue He. Full bayesian significance testing for neural networks in traffic forecasting. In *IJCAI*, pages 2216–2224. ijcai.org, 2024.

[Mao *et al.*, 2022] Zhenyu Mao, Ziyue Li, Dedong Li, Lei Bai, and Rui Zhao. Jointly contrastive representation learning on road network and trajectory. In *CIKM*, pages 1501–1510. ACM, 2022.

[Mikolov *et al.*, 2013a] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *ICLR (Workshop Poster)*, 2013.

[Mikolov *et al.*, 2013b] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *ICLR (Workshop Poster)*, 2013.

[Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: online learning of social representations. In *KDD*, pages 701–710. ACM, 2014.

[Ren *et al.*, 2021] Houxing Ren, Jingyuan Wang, Wayne Xin Zhao, and Ning Wu. Rapt: Pre-training of time-aware transformer for learning robust healthcare representation. In *Proceedings of the 27th ACM SIGKDD conference on*

knowledge discovery & data mining, pages 3503–3511, 2021.

[Shimizu *et al.*, 2020] Toru Shimizu, Takahiro Yabe, and Kota Tsubouchi. Learning fine grained place embeddings with spatial hierarchy from human mobility trajectories. *CoRR*, abs/2002.02058, 2020.

[Wan *et al.*, 2021] Huaiyu Wan, Yan Lin, Shengnan Guo, and Youfang Lin. Pre-training time-aware location embeddings from spatial-temporal trajectories. *IEEE Transactions on Knowledge and Data Engineering*, 34(11):5510–5523, 2021.

[Wang and Li, 2017] Hongjian Wang and Zhenhui Li. Region representation learning via mobility flow. In *CIKM*, pages 237–246. ACM, 2017.

[Wang and Wong, 2023] Libin Wang and Raymond Chi-Wing Wong. Efficient public transport planning on roads. In *ICDE*, pages 2443–2455. IEEE, 2023.

[Wang *et al.*, 2016] Jingyuan Wang, Qian Gu, Junjie Wu, Guannan Liu, and Zhang Xiong. Traffic speed prediction and congestion source exploration: A deep learning method. In *ICDM*, pages 499–508. IEEE Computer Society, 2016.

[Wang *et al.*, 2017a] Jingyuan Wang, Chao Chen, Junjie Wu, and Zhang Xiong. No longer sleeping with a bomb: A duet system for protecting urban safety from dangerous goods. In *KDD*, pages 1673–1681. ACM, 2017.

[Wang *et al.*, 2017b] Jingyuan Wang, Yating Lin, Junjie Wu, Zhong Wang, and Zhang Xiong. Coupling implicit and explicit knowledge for customer volume prediction. In *AAAI*, pages 1569–1575. AAAI Press, 2017.

[Wang *et al.*, 2018] Jingyuan Wang, Xiaojian Wang, and Junjie Wu. Inferring metapopulation propagation network for intra-city epidemic control and prevention. In *KDD*, pages 830–838. ACM, 2018.

[Wang *et al.*, 2021a] Jingyuan Wang, Jiawei Jiang, Wenjun Jiang, Chao Li, and Wayne Xin Zhao. Libcity: An open library for traffic prediction. In *SIGSPATIAL/GIS*, pages 145–148. ACM, 2021.

[Wang *et al.*, 2021b] Jingyuan Wang, Xin Lin, Yuan Zuo, and Junjie Wu. Dgeye: Probabilistic risk perception and prediction for urban dangerous goods management. *ACM Trans. Inf. Syst.*, 39(3):28:1–28:30, 2021.

[Wang *et al.*, 2021c] Jingyuan Wang, Ning Wu, and Wayne Xin Zhao. Personalized route recommendation with neural network enhanced search algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 34(12):5910–5924, 2021.

[Wang *et al.*, 2021d] Meng-xiang Wang, Wang-Chien Lee, Tao-Yang Fu, and Ge Yu. On representation learning for road networks. *ACM Trans. Intell. Syst. Technol.*, 12(1):11:1–11:27, 2021.

[Wu *et al.*, 2020] Ning Wu, Wayne Xin Zhao, Jingyuan Wang, and Dayan Pan. Learning effective road network representation with hierarchical graph neural networks. In *KDD*, pages 6–14. ACM, 2020.

[Wu *et al.*, 2022] Shangbin Wu, Xu Yan, Xiaoliang Fan, Shirui Pan, Shichao Zhu, Chuanpan Zheng, Ming Cheng, and Cheng Wang. Multi-graph fusion networks for urban region embedding. In *IJCAI*, pages 2312–2318. ijcai.org, 2022.

[Yan *et al.*, 2021] Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. Consert: A contrastive framework for self-supervised sentence representation transfer. *CoRR*, abs/2105.11741, 2021.

[Yang and Gidofalvi, 2018] Can Yang and Gyozo Gidofalvi. Fast map matching, an algorithm integrating hidden markov model with precomputation. *International Journal of Geographical Information Science*, 32(3):547–570, 2018.

[Yao *et al.*, 2018] Zijun Yao, Yanjie Fu, Bin Liu, Wangsu Hu, and Hui Xiong. Representing urban functions through zone embedding with human mobility patterns. In *IJCAI*, pages 3919–3925. ijcai.org, 2018.

[Yu *et al.*, 2024] Xie Yu, Jingyuan Wang, Yifan Yang, Qian Huang, and Ke Qu. Bigcity: A universal spatiotemporal model for unified trajectory and traffic state data analysis. *arXiv preprint arXiv:2412.00953*, 2024.

[Yuan *et al.*, 2024] Yuan Yuan, Jingtao Ding, Jie Feng, Depeng Jin, and Yong Li. 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, 2024.

[Zhang and Long, 2023] Liang Zhang and Cheng Long. Road network representation learning: A dual graph based approach. *CoRR*, abs/2304.07298, 2023.

[Zhang *et al.*, 2019] Yunchao Zhang, Yanjie Fu, Pengyang Wang, Xiaolin Li, and Yu Zheng. Unifying inter-region autocorrelation and intra-region structures for spatial embedding via collective adversarial learning. In *KDD*, pages 1700–1708. ACM, 2019.

[Zhang *et al.*, 2020] Mingyang Zhang, Tong Li, Yong Li, and Pan Hui. Multi-view joint graph representation learning for urban region embedding. In *IJCAI*, pages 4431–4437. ijcai.org, 2020.

[Zhang *et al.*, 2022] Liang Zhang, Cheng Long, and Gao Cong. Region embedding with intra and inter-view contrastive learning. *CoRR*, abs/2211.08975, 2022.

[Zhang *et al.*, 2024] Wentao Zhang, Jingyuan Wang, Yifan Yang, et al. Veccity: A taxonomy-guided library for map entity representation learning. *arXiv preprint arXiv:2411.00874*, 2024.

[Zhao *et al.*, 2017] Shenglin Zhao, Tong Zhao, Irwin King, and Michael R. Lyu. Geo-teaser: Geo-temporal sequential embedding rank for point-of-interest recommendation. In *WWW (Companion Volume)*, pages 153–162. ACM, 2017.

[Zhou *et al.*, 2023] Silin Zhou, Dan He, Lisi Chen, Shuo Shang, and Peng Han. Heterogeneous region embedding with prompt learning. In *AAAI*, pages 4981–4989. AAAI Press, 2023.