

Spatial Heterophily Aware Graph Neural Networks

Congxi Xiao*
School of Computer Science and
Technology, University of Science and
Technology of China
Baidu Research
xiaocongxi@mail.ustc.edu.cn

Jingbo Zhou†
Business Intelligence Lab,
Baidu Research
zhoujingbo@baidu.com

Jizhou Huang
Baidu Inc.
huangjizhou01@baidu.com

Tong Xu
School of Computer Science and
Technology, University of Science and
Technology of China
State Key Laboratory of Cognitive
Intelligence
tongxu@ustc.edu.cn

Hui Xiong†
The Thrust of Artificial Intelligence,
The Hong Kong University of Science
and Technology (Guangzhou),
The Department of Computer Science
and Engineering, The Hong Kong
University of Science and Technology
xionghui@ust.hk

ABSTRACT

Graph Neural Networks (GNNs) have been broadly applied in many urban applications upon formulating a city as an urban graph whose nodes are urban objects like regions or points of interest. Recently, a few enhanced GNN architectures have been developed to tackle heterophily graphs where connected nodes are dissimilar. However, urban graphs usually can be observed to possess a unique spatial heterophily property; that is, the dissimilarity of neighbors at different spatial distances can exhibit great diversity. This property has not been explored, while it often exists. To this end, in this paper, we propose a metric, named Spatial Diversity Score, to quantitatively measure the spatial heterophily and show how it can influence the performance of GNNs. Indeed, our experimental investigation clearly shows that existing heterophilic GNNs are still deficient in handling the urban graph with high spatial diversity score. This, in turn, may degrade their effectiveness in urban applications. Along this line, we propose a Spatial Heterophily Aware Graph Neural Network (SHGNN), to tackle the spatial diversity of heterophily of urban graphs. Based on the key observation that spatially close neighbors on the urban graph present a more similar mode of difference to the central node, we first design a rotation-scaling spatial aggregation module, whose core idea is to properly group the spatially close neighbors and separately process each group with less diversity inside. Then, a heterophily-sensitive spatial interaction module is designed to adaptively capture the

commonality and diverse dissimilarity in different spatial groups. Extensive experiments on three real-world urban datasets demonstrate the superiority of our SHGNN over several its competitors.

CCS CONCEPTS

• Information systems → Spatial-temporal systems.

KEYWORDS

Urban graphs, Spatial heterophily, Graph neural networks

ACM Reference Format:

Congxi Xiao, Jingbo Zhou, Jizhou Huang, Tong Xu, and Hui Xiong. 2023. Spatial Heterophily Aware Graph Neural Networks. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23)*, August 6–10, 2023, Long Beach, CA, USA. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3580305.3599510>

1 INTRODUCTION

Applying Graph Neural Networks (GNNs) in different urban applications has attracted much research attention in the past few years [9, 10, 31, 33, 39, 43, 54]. These studies usually model the city as an urban graph whose nodes are urban objects (e.g., regions or Points of Interest (POIs)) and whose edges are physical or social dependencies in the urban area (e.g., human mobility and road connection [39, 40]). Upon urban graphs, GNNs with variant architectures are proposed to achieve the classification or regression tasks.

Nevertheless, there is a serious limitation of GNNs which has been largely overlooked in previous studies but have attracted increasing research attention recently: GNNs have an implicit homophily assumption that nodes only with similar features or same labels are connected together on the graph [53, 55]. Meanwhile, the opposite assumption is heterophily that connected nodes have dissimilar features or labels. In fact, heterophily usually exists in an urban graph as it describes the complex urban system where both similar and dissimilar urban objects (e.g., regions with different functionalities) can correlate with each other in complex manners. Taking the urban graph constructed with human mobility as an example, the start node and end node of an edge could be home

*This work was done when the first author was an intern at Baidu Research under the supervision of Jingbo Zhou.

†Corresponding authors.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](https://permissions.acm.org).

KDD '23, August 6–10, 2023, Long Beach, CA, USA

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-0103-0/23/08...\$15.00

<https://doi.org/10.1145/3580305.3599510>

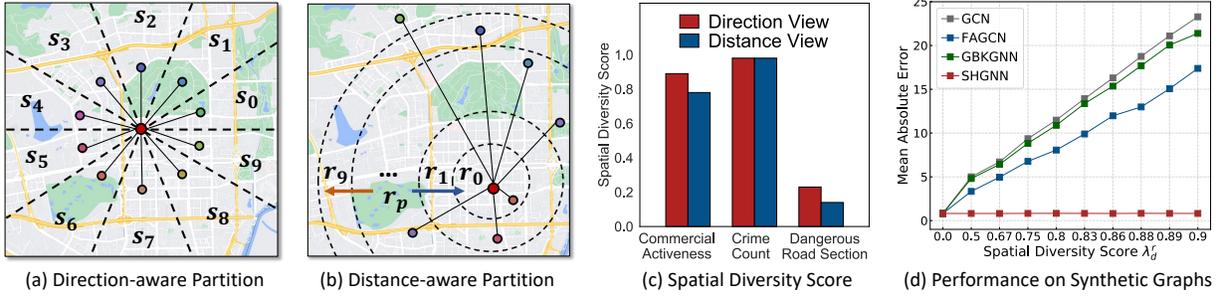


Figure 1: Analysis of spatial heterophily. (a)-(b) illustrate the space partition from two spatial views. (c) shows the spatial diversity scores calculated on three real-world urban graphs. (d) presents the results of experimental investigation.

and workplace, respectively, which are definitely heterophilic. Such difference information on the heterophilic urban graph may not be modeled well by many traditional homophilic GNNs, who tend to generate similar representations for connected nodes [2, 55]. In this way, the performance of these homophilic GNN methods on urban graphs may be largely hindered.

Our further observation is that urban graphs have a unique *Spatial Heterophily* property. To be specific, we find that the heterophily on the urban graph often presents a characteristic of *spatial diversity*. In other words, the difference (or dissimilarity) between the central node and its neighbors at different distances or directions exhibits evident discrepancy, rather than distributed uniformly. Aware of such a characteristic, an attendant question is how to measure the spatial heterophily of the urban graph. There have been various studies put forward to investigate the graph homophily and heterophily from different perspectives, including node homophily [29], edge homophily [56] and class homophily [23]. But without considering the spatial position of linking nodes, these metrics cannot describe the spatial heterophily on urban graphs.

Therefore, in this work, we propose a metric named **spatial diversity score** to analyze the spatial heterophily, and investigate its influence on the performance of existing GNN methods on urban graphs. Firstly, we divide the neighbors on an urban graph into different spatial groups according to their locations (including direction and distance), and then the spatial diversity score measures the discrepancy between different spatial groups, in terms of their label dissimilarity to the central node. A higher score (close to one) indicates a larger discrepancy between different spatial groups, and thus a higher spatial diversity of heterophily on the urban graph.

Yet, it still remains an outstanding challenge for designing powerful heterophilic GNN models over an urban graph if its spatial diversity score is high, where there is a diversity of dissimilarity distributions between the central node and its neighbors at different spatial locations. There are some recent studies to improve the GNN architectures to handle the graph heterophily [8, 14, 17, 24, 48]. Most of these methods can only work on a heterophilic graph when there is limited difference between nodes. For example, GBKGNN [8] assumes that there are only two different kinds of nodes, and FAGCN [2] assumes that node features have only two different levels of frequencies (Note that this limitation is discussed in their papers). Following this line, it is hard to model such diverse distributions of spatial heterophily on urban graphs. To provide more evidence, we conduct an experiment on synthetic urban graphs with varying levels of spatial diversity score (details are in Section

3.2). As shown in Figure 1(d), when the graph presents higher spatial diversity score, the performance of these two state-of-the-art heterophilic GNNs, GBKGNN and FAGCN, is far from optimal. We also apply the proposed spatial diversity score to analyze three real-world urban graphs under different target tasks in our experiments. As we can see from Figure 1(c), three urban graphs present different levels of spatial heterophily, where one of them can get a very high score (0.99 on the urban graph in crime prediction task). Thus, it is valuable to develop an effective GNN model that can handle the diverse spatial heterophily of the urban graph.

Through our in-depth analysis of spatial heterophily, we observe that the heterophily further exhibits a spatial tendency on urban graphs, which reveals a promising opportunity for us to tackle such diverse heterophily in a divide-and-conquer way. Different from ordinary graphs, nodes on urban graphs should follow Tobler’s First Law of Geography (TFL) [35]. As the fundamental assumption used in almost all urban analysis, TFL means *everything is related to everything else, but near things are more related than distant things*. Obeying TFL, spatially close neighbors on the urban graph present a more similar mode of difference to the central node, compared to the distant ones. We also analyze the real-world urban graph in commercial activeness prediction task to visualize such a tendency (details are in Section 4). Figure 2 clearly shows that spatially close neighbors present less discrepancy from both the direction and distance view. Thus, if we can properly group the spatially close neighbors together, it is possible to alleviate the diversity of heterophily inside groups on the urban graph.

To this end, we propose a novel Spatial Heterophily Aware Graph Neural Network (SHGNN), to tackle the spatial heterophily on urban graphs, with two specially designed modules. First, we devise a Rotation-Scaling Spatial Aggregation module. Its core idea is to properly divide the neighbors into different spatial groups according to their direction and distance to the central node, and perform a spatial-aware feature aggregation for each group, which serves as the basis of handling diverse heterophily distributions separately. Then, a Heterophily-Sensitive Spatial Interaction module with two learnable kernel functions is designed to capture the commonality and discrepancy in the neighborhood, and adaptively determine what and how much difference information the central node needs. It acts between the central node and neighbors in different groups to manage the spatial diversity of heterophily on the urban graph.

The contribution of this paper is summarized as follows:

- To the best of our knowledge, we are the first to investigate the spatial heterophily of urban graphs. We design a metric named

spatial diversity score, to analyze the spatial heterophily property, and identify the limitation of existing GNNs in handling the diverse spatial heterophily on the urban graph.

- We propose a novel spatial heterophily aware graph neural network named SHGNN, in which two techniques: rotation-scaling spatial aggregation and heterophily-sensitive spatial interaction are devised to tackle the spatial heterophily of the urban graph in a divide-and-conquer way.
- We conducted extensive experiments to verify the effectiveness of SHGNN on three real-world datasets.

2 PRELIMINARIES

In this section, we first introduce the basic concepts of the urban graph, then clarify the goal of our work. The frequently used notations are summarized in Table 4 in Appendix.

Urban Graphs. Let $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{X})$ denote an urban graph, where $\mathcal{V} = \{v_1, \dots, v_N\}$ denotes a set of nodes representing a kind of urban entity, \mathcal{E} denotes the edge set indicating one type of relation among nodes in the urban scenario, and $\mathcal{N}(v_i) = \{v_j | (v_i, v_j) \in \mathcal{E}\}$ is the neighborhood of node v_i . $\mathbf{X} \in \mathbb{R}^{N \times d}$ denotes the feature matrix, in which the i -th row is the d -dimensional node features of v_i obtained from the urban data. Different instantiations of node set and edge set will form different urban graphs, such as: (1) **Mobility Graph** with regions as nodes and human flows as edges. The node features can be some region attributes such as the distribution of POI inside the region; (2) **Road Network** where the node set is formed by road sections, and the edges denote the connectivity between them. The node features can be the structural information of a section, such as the number of branches and lanes.

Problem Formulation. Given an urban graph, our goal is to design a GNN model, that considers and alleviates the spatial heterophily, to learn the node representation $f: (v_i | \mathcal{G}) \rightarrow \hat{\mathbf{h}}_i$, where $\hat{\mathbf{h}}_i$ denotes the representation vector of node v_i . The model f will be trained in an end-to-end manner in different downstream tasks.

3 SPATIAL HETEROPHILY

In this section, an analysis of spatial heterophily on urban graphs is provided. We first introduce our metric to measure the spatial heterophily (Section 3.1). Then, Section 3.2 gives an experimental investigation on synthetic graphs. It not only demonstrates the importance for GNNs to consider the spatial heterophily on urban graphs, but also suggests a promising way to tackle this challenge.

3.1 Spatial Diversity of Heterophily

To describe the spatial diversity of heterophily on urban graphs, we design a metric named **spatial diversity score**. Typically, the graph heterophily is measured by the label dissimilarity between the central node and its neighbors (e.g., [8, 29]). Following this line, our spatial diversity score aims to further assess the discrepancy between neighbors at different spatial locations, in terms of the distributions of their label dissimilarity to the central node. Briefly, we first divide the neighborhood into different spatial groups according to their spatial locations. Then, we can measure spatial groups' discrepancy by calculating the Wasserstein distance between their label dissimilarity distributions, and the metric is further defined as the ratio of nodes with high discrepancy.

3.1.1 Dual-View Space Partition. To distinguish the spatial location of neighbors and form different spatial groups, we first partition the geographic space into several non-overlap subspaces, and each neighbor of the central node can be assigned to the group corresponding to the subspace it locates in. Note that the spatial heterophily can both present in different directions and distances, thus we propose to perform space partition from both two views.

Direction-Aware Partition. Given the central node v_i on an urban graph, we evenly partition the geographic space centered by it into ten direction sectors $\mathcal{S} = \{s_k | k = 0, 1, \dots, 9\}$. Correspondingly, nodes in the neighborhood $\mathcal{N}(v_i)$ will be divided into the sector they locate in. Neighbors belonging to the same sector are redefined as the direction-aware neighborhood $\{\mathcal{N}_{s_k}(v_i) | k = 0, 1, \dots, 9\}$, where $\bigcup_{k=0}^9 \mathcal{N}_{s_k}(v_i) = \mathcal{N}(v_i)$. In this way, the direction-aware neighborhoods can be regarded as different spatial groups associated with different spatial relations to the central node. We will then calculate the discrepancy between different spatial groups. Figure 1(a) illustrates such a sector partition.

Distance-Aware Partition. As illustrated in Figure 1(b), we also divide the neighbors based on their distance to the central node. To be specific, we first determine the distance range of the neighborhood on an urban graph, by making a statistic of the distance distribution between connected nodes. Note that we consider the 90% percentile of this distribution as the maximum distance of the neighborhood on the graph. This is based on the observation that the distance distribution often presents a long tail property, and such a distance cut-off can avoid interference from extremely distant outliers. And then, the distance range is evenly split into ten buckets, which results in distance rings $\mathcal{R} = \{r_k | k = 0, 1, \dots, 9\}$. Similarly, the original neighborhood $\mathcal{N}(v_i)$ can be divided into these ten distance-aware neighborhoods $\{\mathcal{N}_{r_k}(v_i) | k = 0, 1, \dots, 9\}$ as another view of spatial groups, where we also have $\bigcup_{k=0}^9 \mathcal{N}_{r_k}(v_i) = \mathcal{N}(v_i)$.

3.1.2 Spatial Diversity Score. After the neighborhood partition, our goal is to further define the spatial diversity score by measuring the discrepancy between different spatial groups, based on their distribution distance of label dissimilarity to the central node.

First of all, we define spatial group's label dissimilarity distribution to the central node. For a node classification task with C classes, within a spatial group, this distribution is calculated as the ratio of neighbors belonging to each class (different to the central node's). Taking the direction view as an example, for the central node v_i , the label dissimilarity distribution of s_k is formally defined as $P_i^{s_k} = [P_{i,0}^{s_k}, P_{i,1}^{s_k}, \dots, P_{i,|C-1|}^{s_k}]$ with:

$$P_{i,c}^{s_k} = \sum_{v_j \in \mathcal{N}_{s_k}(v_i) \wedge y_j \neq y_i} \mathbb{1}(y_{j,c} = 1) \cdot |\mathcal{N}_{s_k}(v_i)|^{-1}, \quad (1)$$

where $c = 0, 1, \dots, |C-1|$, $\mathbb{1}(\cdot)$ is the indicator function, and y_i is v_i 's one-hot label vector whose c -th value is denoted by $y_{i,c}$.

In addition, to improve the generality of this metric to more urban applications (e.g., regression tasks), we also extend the definition of label dissimilarity distribution above to the node regression task. To be specific, we first make a statistic of the label difference between connected nodes on the whole graph $\hat{\mathcal{Y}} = \{(y_j - y_i) | (v_i, v_j) \in \mathcal{E}\}$, and calculate the deciles $\{D_1, D_2, \dots, D_9\}$ of this distribution. These nine deciles can determine ten buckets (intervals), which will be used for the discretization of continuous label difference value, to

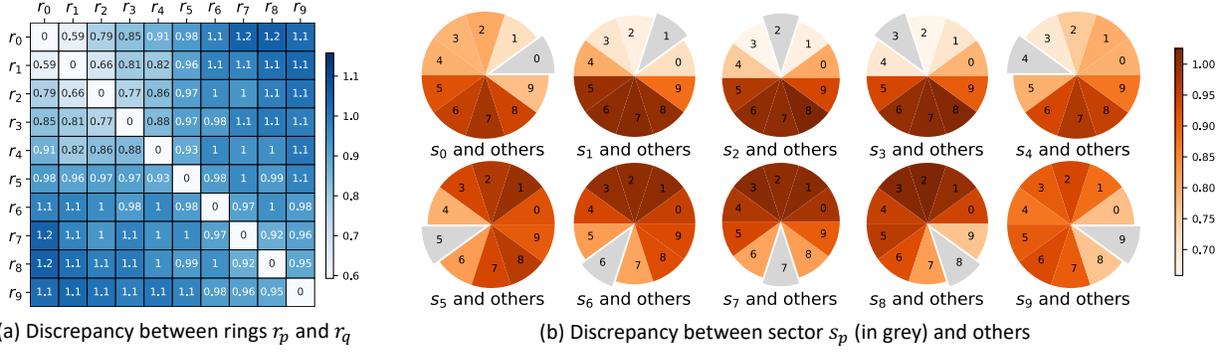


Figure 2: Illustration of spatial tendency in the distance view (a) and direction view (b).

obtain a similar form of label dissimilarity as the node classification task. Hence, still for the spatial group $\mathcal{N}_{s_k}(v_i)$ in sector s_k , its label dissimilarity distribution will be calculated as the ratio of neighbors mapped into different buckets, according to their discretized label difference to the central node, which can be also formulized as: $P_i^{s_k} = [P_{i,0}^{s_k}, P_{i,1}^{s_k}, \dots, P_{i,9}^{s_k}]$, with the c -th element computed by:

$$P_{i,c}^{s_k} = \sum_{v_j \in \mathcal{N}_{s_k}(v_i)} \mathbb{1}(D_c < y_j - y_i \leq D_{c+1}) \cdot |\mathcal{N}_{s_k}(v_i)|^{-1}, \quad (2)$$

where $c = 0, 1, \dots, 9$. The D_0 and D_{10} are used to denote the minimum and maximum of the label difference distribution on the whole graph, respectively (i.e., $D_0 = \min(\mathcal{Y})$ and $D_{10} = \max(\mathcal{Y})$).

Next, the discrepancy between different spatial groups can be defined by measuring the distance between their label dissimilarity distributions. Following a recent study on the graph heterophily [52], we adopt Wasserstein distance (WD) to measure the distribution distance between two spatial groups. Formally, consider two spatial groups $\mathcal{N}_{s_p}(v_i)$ and $\mathcal{N}_{s_q}(v_i)$ in sector s_p and s_q of node v_i , the discrepancy between them is defined as:

$$Disc(v_i, s_p, s_q) = WD(P_i^{s_p}, P_i^{s_q}), \quad (3)$$

where $WD(\cdot, \cdot)$ denotes the Wasserstein distance between two distributions, which can be approximately calculated by the Sinkhorn iteration algorithm [6].

With such a measurement, we can finally define the spatial diversity score to describe the diverse spatial heterophily of an urban graph. This metric can be computed based on the ratio of nodes with high discrepancy among different spatial groups:

$$\lambda_d^s = |\mathcal{V}|^{-1} \cdot \sum_{v_i \in \mathcal{V}} \mathbb{1}(\max_{p \neq q} Disc(v_i, s_p, s_q) \geq 1), \quad (4)$$

where $\max_{p \neq q} Disc(v_i, s_p, s_q) \geq 1$ with $p, q = 0, 1, \dots, 9$ indicates that there are at least two sectors being discrepant in terms of their label dissimilarity distributions to the central node v_i . In this way, the score λ_d^s will get higher if there are more nodes whose spatial groups present high discrepancy on the urban graph. Similarly, we can also define the score λ_d^r in the distance view, which measures the discrepancy between spatial groups formed by different rings in \mathcal{R} , with different distances to the central node:

$$\lambda_d^r = |\mathcal{V}|^{-1} \cdot \sum_{v_i \in \mathcal{V}} \mathbb{1}(\max_{p \neq q} Disc(v_i, r_p, r_q) \geq 1), \quad (5)$$

In practice, when only a part of nodes are labeled on an urban graph, it's often sufficient to use the labeled data to estimate λ_d^s and

λ_d^r . Figure 1(c) shows the scores of three real-world urban graphs. As we can see, from both the direction and distance view, urban graphs can present very high spatial diversity scores (e.g., 0.99 in crime prediction), which reveals the diverse heterophily in different spatial groups, with discrepant label dissimilarity distributions.

3.2 Experimental Investigation

Next, we conduct an experimental investigation on synthetic urban graphs to illustrate the importance of considering the diverse spatial heterophily on the urban graph. Specifically, we test the performance of several GNN models on a series of synthetic urban graphs with increasing spatial diversity of the heterophily. We generate 10 graphs containing 5000 nodes with 10-dimensional randomly generated feature vectors. For each node, we build 50 edges and assume that these neighbors locate at 10 distance rings from near to far around the central node. To increase the spatial diversity of heterophily from \mathcal{G}_1 to \mathcal{G}_{10} , we gradually enlarge the discrepancy of label dissimilarity distributions between neighbors in different distance rings. Specifically, in graph \mathcal{G}_i , we evenly divide the node set into i subsets $\mathcal{V}_i = \bigcup_{j=1}^i \mathcal{V}_{i,j}$, where the labels of nodes in $\mathcal{V}_{i,j}$ are sampled from the Gaussian distribution $\mathcal{N}(10j, 1)$. Then, the 50 neighbors connected to the node in $\mathcal{V}_{i,j}$ are randomly selected from one of subsets $\{\mathcal{V}_{i,k}\}_{k=j}^i$ by an equal probability $1/(i-j+1)$. Besides, we let neighbors' spatial distance be consistent with their label difference to the central node (i.e., neighbors with smaller differences locates in the closer distance ring). In this way, these 10 graphs will have increasing spatial diversity scores λ_d^r . Meanwhile, neighbors in the same distance ring have less discrepancy.

Figure 1(d) shows the node regression error of GCN [18] and two state-of-the-art heterophic GNNs (FAGCN [2] and GBKGN [8]) on 10 synthetic graphs. As expected, GCN gets the worst performance, since it proved unsuitable for heterophily graphs in most cases [24]. For two heterophic GNNs, despite less error than GCN, their performance is still far from optimal when the spatial diversity score increases. This is because they only model limited dissimilarity between nodes (e.g., two levels of frequencies [2] or two classes of nodes [8]), without considering diverse dissimilarity distributions. However, if the model separately processes neighbors in each distance ring whose differences to the central node are more similar, (as our SHGNN did), it can always achieve satisfactory performance. Thus, it's reasonable to consider whether we can tackle the diverse spatial heterophily, by properly grouping together the neighbors with less discrepancy on the urban graph.

4 METHODOLOGY

In this section, we first present that the heterophily on urban graphs often exhibits a spatial tendency: spatially close neighbors have more similar heterophily distributions than distant ones, which obeys TFL. This characteristic gives rise to the solution that we can properly divide neighbors according to their spatial locations, to achieve grouping neighbors with less discrepancy together. Then, we introduce our proposed SHGNN that leverages such a spatial tendency to tackle the spatial heterophily of urban graphs.

Spatial Tendency. In addition to the presence of a discrepancy between different spatial groups (discussed in Section 3.1), our in-depth investigation of spatial heterophily further reveals that such a discrepancy shows a spatial tendency on urban graphs. Specifically, we observe that the discrepancy between two spatially close groups is smaller, compared to two distant groups. We conduct a data analysis to visually present such a tendency. Figure 2 shows the pair-wise discrepancy of label dissimilarity distributions between any two spatial groups, which is computed based on real-world human mobility and regional commercial activeness data. In Figure 2(a), for the spatial group formed by distance ring r_p , we can find that its discrepancy with other rings r_q is highly correlated to their spatial distance (i.e., the discrepancy gets higher when r_q is more distant to r_p). For example, the close ring pair (r_0, r_1) are less discrepant than the distant pair (r_0, r_9) . A similar spatial tendency can also be observed from the direction view. As shown in Figure 2(b), in most cases, the discrepancy will increase along with the included angle between sectors. Taking sector s_0 as an example, its discrepancy with $s_1 \sim s_9$ increases first and then decreases, which is roughly in sync with the change of their included angles. In other words, a sector is more likely to be discrepant to another distant sector (e.g., (s_1, s_6)) than a nearby one (e.g., (s_1, s_2)). This characteristic motivates us to address the diverse spatial heterophily on urban graphs, by properly grouping spatially close neighbors and separately processing each group with less discrepancy inside.

To this end, we propose a novel GNN architecture named SHGNN, which is illustrated in Figure 3 and 4. Our model consists of two components: Rotation-Scaling Spatial Aggregation (see Section 4.1) and Heterophily-Sensitive Spatial Interaction (see Section 4.2).

4.1 Rotation-Scaling Spatial Aggregation

This component aims to properly group spatially close neighbors and alleviate the diversity of heterophily inside groups in the message passing process. In general, we first divide neighbors according to their relative positions to the central node. Then, the feature aggregation is performed in each spatial group separately.

4.1.1 Rotation-Scaling Dual-View Partition. Following the neighborhood partition in Section 3.1.1, we also partition the geographic space into non-overlap subspaces, and neighbors located at the same subspace are then grouped together. Note that there are two major differences between the space partition performed in this component and that in Section 3.1.1. First, we apply a more general partition with a variable number of subspaces. Second, we introduce a rotation-scaling multi-head partition strategy to model the neighbor’s spatial location in a more comprehensive way.

To be specific, in the direction view, we evenly partition the space into a set of sectors $\mathcal{S} = \{s_k \mid k = 0, 1, \dots, n_s - 1\}$, where n_s denotes

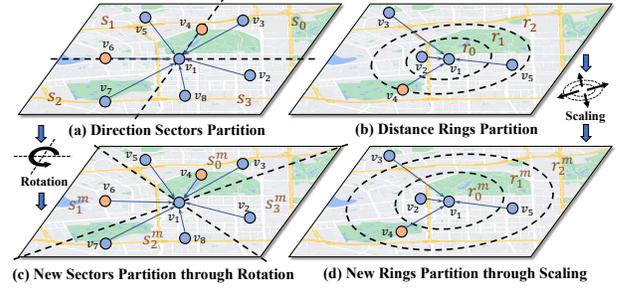


Figure 3: Illustration of Rotation-scaling Partition.

the number of partitioned sectors, which can be appropriately set for different datasets. Nodes in each direction-aware neighborhood $\mathcal{N}_{s_k}(v_i)$ of sector s_k are grouped together, which we still call spatial group. In the distance view, the space will be partitioned into n_r distance rings $\mathcal{R} = \{r_k \mid k = 0, 1, \dots, n_r - 1\}$, which is resulted from a predefined distance bucket (e.g., $< 1km$, $1 - 2km$, and $> 2km$). Note that the central node v_i itself does not belong to any sector or ring, we regard it as an additional group $\mathcal{N}_{s_{n_s}}(v_i) = \mathcal{N}_{r_{n_r}}(v_i) = \{v_i\}$.

Rotation-Scaling Multi-Head Partition. In view of the special case that a part of neighbors may locate at the boundary between two subspaces, we further propose a multi-head partition strategy to simultaneously perform multiple partitions at each view, where different heads can complement each other. For example, as shown in Figure 3(a), the orange node v_4 locates at the boundary between sectors s_0 and s_1 , which indicates that the spatial relation of the neighborhood is still inadequately excavated by the single direction-based partition. A similar situation can be found in the partition of distance rings, such as the node v_4 in Figure 3(b).

To overcome this limitation, we extend our partition strategy by devising two operations, which are *sector rotation* and *ring scaling*, to achieve multiple space partitions for capturing the diverse spatial relations comprehensively. Specifically, as illustrated in Figure 3(a) and (c), for the originally partitioned direction sectors, we turn the sector boundary a certain angle (e.g., 45 degrees) to derive another set of sectors, then the neighbors can be correspondingly reassigned in these new sectors and form a different set of direction-aware neighborhoods. Thus, we update the denotation of sectors as $\mathcal{S}^m = \{s_k^m \mid k = 0, 1, \dots, n_s\}$, and that of direction-aware neighborhoods as $\{\mathcal{N}_{s_k^m}(v_i) \mid k = 0, 1, \dots, n_s\}$, where $m = 1, 2, \dots, M_s$ denotes the m -th head partition among total M_s heads. Similarly, from the distance view, we scale the boundary of the original distance rings to obtain the supplemental partition, which is shown in Figure 3(b) and (d). The denotations are updated as $\mathcal{R}^m = \{r_k^m \mid k = 0, 1, \dots, n_r\}$ and $\{\mathcal{N}_{r_k^m}(v_i) \mid k = 0, 1, \dots, n_r\}$ for rings and distance-aware neighborhoods, where $m = 1, 2, \dots, M_r$. In this way, different heads of partitions model the spatial relation between neighbors and the central node complementarily, and thus we can avoid the improper grouping under a single partition.

4.1.2 Spatial-Aware Aggregation. After the multi-head partition from two spatial perspectives, we collect messages from neighbors to the central node. Rather than mixing the messages (e.g., through averaging) that most of the GNNs follow [47], our model performs a group-wise aggregation to handle different spatial groups with diverse heterophily. An illustration is shown in Figure 4(a).

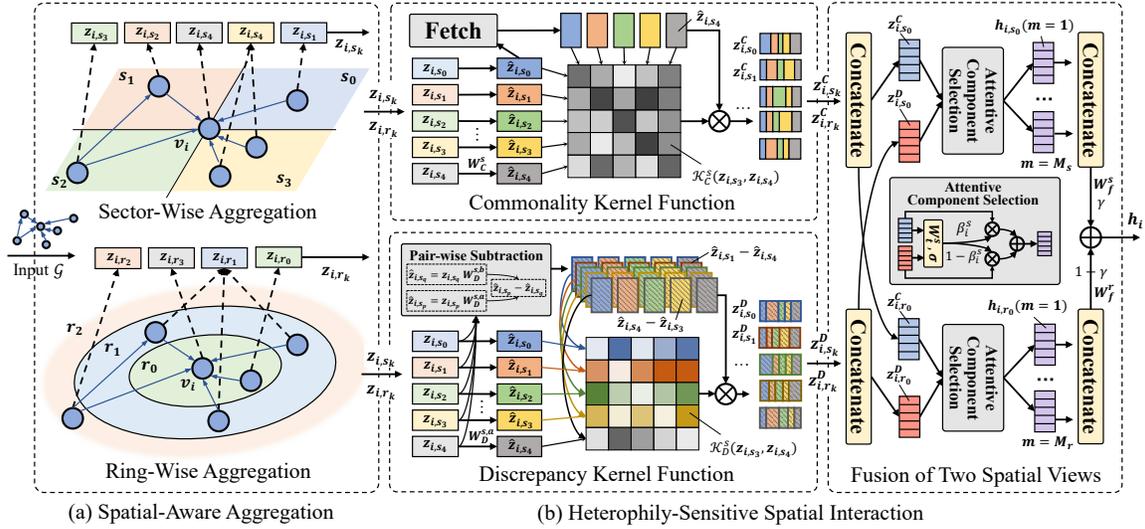


Figure 4: The architecture of SHGNN. We only detailedly present the operation under sector partition in the illustration of two kernel functions and the attentive component selection. The same operation is also performed under ring partition.

Formally, taking the direction view as an example, with the set of direction-aware neighborhoods (spatial groups) under the m -head partition $\{\mathcal{N}_{s_k}^m(v_i) \mid k = 0, 1, \dots, n_s\}$, we use graph convolution [18] to respectively aggregate the features of nodes in each neighborhood $\mathcal{N}_{s_k}^m(v_i)$ with the normalization by degree:

$$z_{i,s_k}^m(l+1) = \sum_{j \in \mathcal{N}_{s_k}^m(v_i)} (|\mathcal{N}(v_i)| \cdot |\mathcal{N}(v_j)|)^{-\frac{1}{2}} \mathbf{h}_j(l) \mathbf{W}_{s_k}^m(l), \quad (6)$$

where $z_{i,s_k}^m(l)$ denotes the l -layer aggregated message from direction sector s_k^m , $\mathbf{h}_j(l)$ denotes the l -layer features of neighbor v_j with $\mathbf{h}_j(0) = \mathbf{x}_j$, and $\mathbf{W}_{s_k}^m(l)$ is a trainable transformation extracting useful information from neighbors' features.

Similarly, at the distance view, we also perform the ring-wise aggregation in each distance ring separately by:

$$z_{i,r_k}^m(l+1) = \sum_{j \in \mathcal{N}_{r_k}^m(v_i)} (|\mathcal{N}(v_i)| \cdot |\mathcal{N}(v_j)|)^{-\frac{1}{2}} \mathbf{h}_j(l) \mathbf{W}_{r_k}^m(l), \quad (7)$$

where $\mathbf{W}_{r_k}^m(l)$ is another feature transformation for neighbors at different distances. In this way, the aggregated messages can not only capture the structure information on the graph, but also discriminate their different spatial groups. It avoids losing different distributions of spatial heterophily on the urban graph.

4.2 Heterophily-Sensitive Spatial Interaction

After the group-wise feature aggregation, SHGNN further captures the diverse spatial heterophily in different spatial groups on the urban graph. In detail, we devise two learnable kernel functions to first respectively capture the commonality and discrepancy between the central node and each spatial group. Then, an attentive gate is jointly learned to adaptively determine the ratio of two components that should be propagated to the central node. Additionally, as indicated by the analysis of spatial tendency in Figure 2, the discrepancy of heterophily distributions varies along with the distance between two spatial groups, we further consider such characteristics by allowing the kernel functions to act between groups, which

encourages the propagation of common and discrepant information among them. Since we view the central node as an additional group, this process can be regarded as an interaction among every two groups. For simplicity, we omit the index l and m of layer and head in the following discussion.

Commonality Kernel Function. Given the fact that different sectors / rings all belong to the neighborhood of the central node, they may share some common knowledge that can probably enhance the representation of each other. Thus, we first design a commonality kernel function to capture such information among them.

Formally, taking the direction view as an example, with the representation $\{z_{i,s_k} \mid k = 0, 1, \dots, n_s\}$ of n_s sectors centered by v_i , the kernel function $\mathcal{K}_C^S(\cdot, \cdot)$ that models the commonality degree between sector s_p and s_q (including v_i itself) is defined as:

$$\mathcal{K}_C^S(z_{i,s_p}, z_{i,s_q}) = \langle \hat{z}_{i,s_p}, \hat{z}_{i,s_q} \rangle, \quad \hat{z}_{i,s_k} = z_{i,s_k} \mathbf{W}_C^S, \quad (8)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product and \mathbf{W}_C^S is the learnable matrix for common knowledge extraction. The larger output value suggests a higher similarity (i.e., commonality) between inputs. Based on this measurement, we enhance the sector representation with the extracted useful information from other sectors:

$$z_{i,s_p}^C = \sum_{q=0}^{n_s} \alpha_{pq}^{C,s} z_{i,s_q} \mathbf{W}_C^S, \quad (9)$$

$$\alpha_{pq}^{C,s} = \frac{\exp(\mathcal{K}_C^S(z_{i,s_p}, z_{i,s_q}))}{\sum_{k=0}^{n_s} \exp(\mathcal{K}_C^S(z_{i,s_p}, z_{i,s_k}))}, \quad (10)$$

where the coefficient $\alpha_{pq}^{C,s}$ is the level of commonality normalized by the softmax function. In the same way, we can also obtain the representation of each distance ring z_{i,r_p}^C with the enhancement of common knowledge from other rings using a similar kernel function $\mathcal{K}_C^R(\cdot, \cdot)$ parametrized by \mathbf{W}_C^R .

Discrepancy Kernel Function. In addition to the common knowledge, modeling the difference information is critical on heterophilic urban graphs. Thus, we devise another kernel function to capture

diverse dissimilarity between the central node and every group, as well as between any two groups. We introduce it from the direction view. Specifically, taking the original representation $\{z_{i,s_k} | k = 0, 1, \dots, n_s\}$ as inputs, the discrepancy kernel is defined as:

$$\mathcal{K}_D^s(z_{i,s_p}, z_{i,s_q}) = \langle z_{i,s_p} \mathbf{W}_D^{s,a}, (z_{i,s_p} \mathbf{W}_D^{s,a} - z_{i,s_q} \mathbf{W}_D^{s,b}) \rangle, \quad (11)$$

where $\mathbf{W}_D^{s,a}$ and $\mathbf{W}_D^{s,b}$ denote two transformations that learn to extract the difference of sector s_q compared to sector s_p . According to [12], the kernel function $\mathcal{K}_D^s(z_{i,s_p}, z_{i,s_q})$ can be regarded as calculating the dissimilarity degree between two inputs, which tends to be higher when s_p and s_q are more dissimilar. Subsequently, our model facilitates each sector to be aware of the helpful difference information from others with this measurement of discrepancy:

$$z_{i,s_p}^D = \sum_{q=0}^{n_s} \alpha_{pq}^{D,s} (z_{i,s_p} \mathbf{W}_D^{s,a} - z_{i,s_q} \mathbf{W}_D^{s,b}), \quad (12)$$

$$\alpha_{pq}^{D,s} = \frac{\exp(\mathcal{K}_D^s(z_{i,s_p}, z_{i,s_q}))}{\sum_{k=0}^{n_s} \exp(\mathcal{K}_D^s(z_{i,s_p}, z_{i,s_k}))}. \quad (13)$$

Similarly, we utilize an analogous kernel function, which is denoted as $\mathcal{K}_D^r(\cdot, \cdot)$ with parameters $\mathbf{W}_D^{r,a}$ and $\mathbf{W}_D^{r,b}$ to capture such discrepancy at the distance view and derive the ring representation z_{i,r_p}^D aware of the diverse distributions of spatial heterophily.

Attentive Component Selection. With these two kernel functions, we can exploit both the common knowledge and diverse discrepancy information between the central node and neighbors in different groups. However, different nodes may possess varying levels of spatial heterophily in various applications. Thus, SHGNN learns to derive a gate that adaptively determines the ratio of common and difference information in an end-to-end manner.

Specifically, for each central node v_i , we concatenate both the commonality and discrepancy components of all sectors, to derive a scalar with a transformation:

$$\beta_i^s = \sigma(\|_{j \in \{C,D\}} \|_{k=0}^{n_s} z_{i,s_k}^j \mathbf{W}_t^s), \quad (14)$$

where \mathbf{W}_t^s denotes the trainable transformation mapping the input to a scalar and σ denotes the *Sigmoid* function restricting the output value into $(0, 1)$. Then, β_i^s serves as a gate controlling the ratio of the commonality and discrepancy components in the final representation of each sector:

$$\tilde{z}_{i,s_k} = \beta_i^s \cdot z_{i,s_k}^C + (1 - \beta_i^s) \cdot z_{i,s_k}^D. \quad (15)$$

In the same way, we learn to derive the gate β_i^r that determines the ratio in each ring's final representation \tilde{z}_{i,r_k} .

After the propagation among spatial groups (including the central node), different groups can contain diverse discrepancy information in the neighborhood, which is vital to the heterophilic urban graph. We then integrate this group-wise representation by concatenation (instead of summation, to avoid the mixing of diverse distributions of spatial heterophily) to obtain the global representation of two views. Since we adopt the multi-head partition strategy, a following concatenation is used to combine different heads at each view. The above two processes can be jointly expressed as:

$$\mathbf{h}_{i,s} = \|_{m=1}^{M_s} (\|_{k=0}^{n_s} \tilde{z}_{i,s_k}^m), \quad \mathbf{h}_{i,r} = \|_{m=1}^{M_r} (\|_{k=0}^{n_r} \tilde{z}_{i,r_k}^m). \quad (16)$$

Table 1: Statistics of three real-world datasets.

Task	Dataset	# Nodes	# Edges	# Labeled	λ_d^s	λ_d^r
CAP	Shenzhen	82,510	9,486,879	9,121	0.89	0.78
CP	Manhattan	180	3,780	180	0.98	0.98
DRSD	Los Angeles	253,985	1,365,289	15,274	0.23	0.14

Fusion of Two Spatial Views. Finally, we fuse two spatial views with a learnable weighted summation to update the central node's representation as follows:

$$\mathbf{h}_i = \gamma \mathbf{h}_{i,s} \mathbf{W}_f^s + (1 - \gamma) \mathbf{h}_{i,r} \mathbf{W}_f^r, \quad (17)$$

where \mathbf{W}_f^s and \mathbf{W}_f^r are two weight matrices transforming the representation vectors of two views into the same space, and γ is a trainable trade-off parameter activated by *Sigmoid* function, which learns to assign different importance to the direction and distance view according to the target task.

4.3 Prediction and Optimization

Consistent with general GNNs, we use the L -layer output activated by *ReLU* function $\hat{\mathbf{h}}_i = \sigma(\mathbf{h}_i^{(L)})$ as the node representation to make a prediction in different downstream tasks, and optimize the model by the appropriate loss function: $\mathcal{L}(LR(\hat{\mathbf{h}}_i), y_i)$. In node regression tasks, $LR(\cdot)$ is the linear regressor, $y_i \in \mathbb{R}$ denotes the ground truth of labeled nodes, and \mathcal{L} can be L2 loss. While for node classification tasks, $LR(\cdot)$ performs the logistic regression, $y_i \in \{0, 1\}^C$ is a one-hot label vector of C classes, and \mathcal{L} can be cross entropy loss.

5 EXPERIMENTS

In this section, we conduct extensive experiments on real-world datasets in three different tasks upon two types of urban graphs to evaluate the effectiveness of our model. The code of SHGNN is available at <https://github.com/PaddlePaddle/PaddleSpatial/tree/main/research/SHGNN>.

5.1 Experiment Settings

5.1.1 Tasks and Data Description. We first briefly introduce the three datasets corresponding to three different tasks, and details of how to build each dataset are described in Appendix A.1. Table 1 summarizes the statistical information of the three datasets.

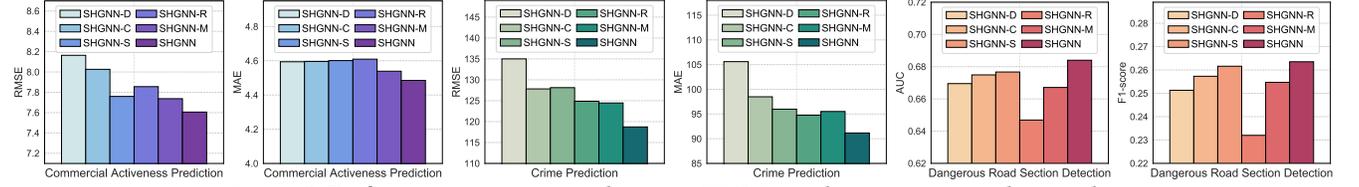
Commercial Activeness Prediction (CAP) is a node regression task on a mobility graph. Similar to [42], we use the number of comments to POIs in each region as the indicator of regional commercial activeness. To form the dataset of this task, we collect the following urban data in **Shenzhen** city in China from Baidu Maps, including POI data and satellite images in September 2019 to construct region features, the daily human flow data from July 2019 to September 2019 for building the edge set of the urban graph, and the number of regional POI comments from June 2019 to April 2020 which is regarded as the ground truth.

Crime Prediction (CP) is also a node regression task on the mobility graph. We collect the real-world dataset of **New York City** from NYC open data website¹ for this task. The dataset contains 180 regions in Manhattan, with the POI data and crime number in each region, as well as taxi trips between regions [40, 51]. We construct node features from POIs. Taxi trips are used to build the

¹opendata.cityofnewyork.us

Table 2: Performance comparison. ‘TASK-S’ refers to task-specific baseline for CAP (KnowCL), CP (NNCCRF), and DRSD (RFN). Symbol *, \blacktriangle and \triangle indicate that SHGNN achieves significant improvements with $p < 0.001$, $p < 0.01$ and $p < 0.05$, respectively.

Methods	Commercial Activeness Prediction			Crime Prediction			Dangerous Road Section Detection	
	RMSE ↓	MAE ↓	R ² ↑	RMSE ↓	MAE ↓	R ² ↑	AUC ↑	F1-score ↑
GCN	8.354 ± 0.020*	5.018 ± 0.028*	0.388 ± 0.003*	156.1 ± 1.312*	114.5 ± 1.005*	0.082 ± 0.015*	0.634 ± 0.002*	0.229 ± 0.003*
GAT	8.952 ± 0.153*	5.134 ± 0.099*	0.298 ± 0.023*	166.8 ± 5.750*	127.1 ± 4.359*	-0.040 ± 0.071*	0.613 ± 0.006*	0.213 ± 0.008 \blacktriangle
Mixhop	8.168 ± 0.038*	4.981 ± 0.010*	0.415 ± 0.005*	147.6 ± 0.576*	109.0 ± 0.398*	0.179 ± 0.006*	0.636 ± 0.003*	0.229 ± 0.005*
FAGCN	8.327 ± 0.018*	5.063 ± 0.031*	0.392 ± 0.002*	150.9 ± 3.618*	108.1 ± 2.179 \blacktriangle	0.142 ± 0.041*	0.632 ± 0.004*	0.228 ± 0.009 \blacktriangle
NLGCN	8.495 ± 0.108*	4.883 ± 0.053*	0.367 ± 0.016*	151.7 ± 4.496*	110.2 ± 3.557 \blacktriangle	0.133 ± 0.050*	0.626 ± 0.003*	0.224 ± 0.004*
GBKGNN	8.490 ± 0.017*	5.004 ± 0.074 \blacktriangle	0.368 ± 0.002*	133.6 ± 4.095 \blacktriangle	98.28 ± 3.792	0.327 ± 0.041 \blacktriangle	0.626 ± 0.011*	0.225 ± 0.013 \blacktriangle
GPRGNN	8.139 ± 0.016 \blacktriangle	4.975 ± 0.025*	0.419 ± 0.002 \blacktriangle	152.4 ± 0.338*	107.0 ± 0.319*	0.126 ± 0.003*	0.635 ± 0.003*	0.226 ± 0.004*
PRIM	8.928 ± 0.144*	5.186 ± 0.086*	0.301 ± 0.022*	151.7 ± 6.243*	118.6 ± 4.320*	0.131 ± 0.070*	0.615 ± 0.008*	0.210 ± 0.005*
SAGNN	8.843 ± 0.068*	5.305 ± 0.057*	0.315 ± 0.010*	142.2 ± 2.451*	108.6 ± 1.360*	0.238 ± 0.026*	0.630 ± 0.006*	0.216 ± 0.008*
TASK-S	8.360 ± 0.068*	4.875 ± 0.028*	0.387 ± 0.010*	138.8 ± 2.619*	100.4 ± 0.945 \blacktriangle	0.274 ± 0.027*	0.638 ± 0.012 \blacktriangle	0.240 ± 0.008 \blacktriangle
SHGNN	7.605 ± 0.131	4.485 ± 0.056	0.493 ± 0.017	118.7 ± 4.135	91.16 ± 2.754	0.469 ± 0.036	0.684 ± 0.004	0.263 ± 0.003

**Figure 5: Performance comparison between SHGNN and its variants on three tasks.**

urban graph, where we only keep the 20 most important edges for each region w.r.t. the number of trip records.

Dangerous Road Section Detection (DRSD) is a node classification task performed on the road network. In this work, the dangerous section is defined as the road section with a high incidence of traffic accidents. We build a real-world dataset in *Los Angeles* based on the road network data from OSMnx Street Networks in Harvard Dataverse² and the traffic accident records in December 2021 from Kaggle dataset website³. First, we make statistics of the number of accident records on each road section. Then, the sections containing more than 3 accident records in a month will be considered as dangerous road sections in our experiments.

To select the best hyper-parameters for all the comparing methods, we randomly split each dataset into three parts with 60% for training, 20% for validation and 20% for test.

5.1.2 Baselines. We compare SHGNN with a variety of state-of-the-art GNN models, including two classical message passing neural networks (GCN [18] and GAT [36]), five representative methods for heterophilic graphs (Mixhop [1], FAGCN [2], NLGCN [26], GPRGNN [5] and GBKGNN [8]), two spatial GNN models (SAGNN [20] and PRIM [4]), as well as three task-specific baselines (KnowCL [27] for CAP, NNCCRF [49] for CP and RFN [13] for DRSD). Detailed descriptions are introduced in Appendix A.3.

5.1.3 Evaluation Metrics. For the two regression tasks, we evaluate all methods with Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and the coefficient of determination (R²). For the node classification task, we use Area Under Curve (AUC) and F1-score.

5.2 Performance Evaluation

5.2.1 Overall Comparison. The performance comparison of our SHGNN and baselines is presented in Table 2, in which the mean and standard deviation of all metrics are obtained through five

random runs. As we can see, SHGNN consistently achieves the best performance in three tasks on two kinds of urban graphs, with 6.6% and 11.2% reductions of RMSE in commercial activeness prediction (CAP) and crime prediction (CP), as well as 7.2% improvements of AUC in dangerous road section detection (DRSD) over the most competitive baseline of each task. We also conduct a pairwise t-test between SHGNN and each baseline to demonstrate that our model outperforms all of them significantly. Note that although the spatial diversity of heterophily on the road network in DRSD task is not so strong ($\lambda_d^s = 0.23$ and $\lambda_d^r = 0.14$) as another two urban graphs, our model can still improve the accuracy in a large margin. It indicates the effectiveness of SHGNN can be general but not just limited to urban graphs with strong spatial heterophily.

Specifically, the ordinary GNN models (GCN and GAT) generally have the worst overall performance. By contrast, approaches designed to deal with graph heterophily (Mixhop, FAGCN, NLGCN, GBKGNN and GPRGNN) evidently perform better. It indicates the inappropriateness of simply treating an urban graph as a general homophilic graph. However, as a specially designed model to handle spatial heterophily, our SHGNN remarkably outperforms these general heterophilic GNNs. The spatial GNN methods (SAGNN and PRIM) perform better than GCN and GAT sometimes, but perform worse than the methods for heterophilic graphs in many cases. And for task-specific baselines, it can also be found that some heterophilic GNNs are level pegging with them (such as Mixhop and GPRGNN vs. KnowCL in CAP task, GBKGNN vs. NNCCRF in CP task and Mixhop vs. RFN in DRSD task). These results also demonstrate the importance to take the spatial heterophily into consideration when using GNNs over an urban graph. To sum up, our SHGNN is much more effective in considering and alleviating the spatial heterophily on urban graphs in all the tasks.

5.2.2 Ablation Study. To verify the effectiveness of each design in our model, we further compare SHGNN with its five variants:

- **SHGNN-S** removes the direction section partition, which only models spatial heterophily from the distance view.

²<https://dataverse.harvard.edu/dataverse/osmnx-street-networks>

³<https://www.kaggle.com/datasets/sobhanmoosavi/us-accidents>

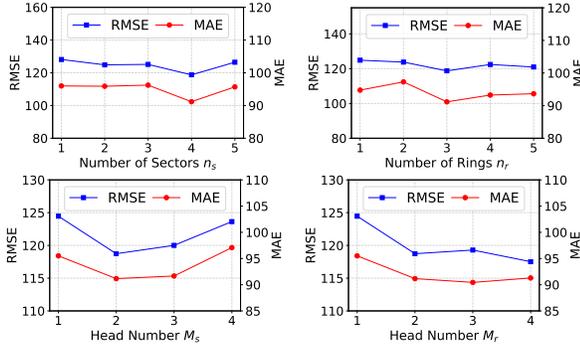


Figure 6: Parameter analysis in CP task.

- **SHGNN-R** removes the distance ring partition, which only models spatial heterophily from the direction view.
- **SHGNN-M** removes the multi-head partition strategy.
- **SHGNN-C** removes the commonality kernel function, without sharing common knowledge among spatial groups.
- **SHGNN-D** removes the discrepancy kernel function. It cannot capture the difference among spatial groups.

As shown in Figure 5, SHGNN outperforms all the variants, proving the significance of our designs in tackling spatial heterophily. Specifically, the performances get worse if we remove either the partition of direction sections or distance rings (SHGNN-S and SHGNN-R), which indicates the necessity of considering the spatial heterophily from both views. Besides, the rotation-scaling multi-head partition strategy can evidently help to model diverse spatial relations (SHGNN-M). In addition, the performance degrades if the commonality kernel function is not used, suggesting the effectiveness of sharing knowledge among neighbors. More importantly, removing the discrepancy kernel function results in a notable performance decline, which verifies the importance of further capturing and exploiting the difference information on heterophilic urban graphs.

5.2.3 Parameters Analysis. We further investigate the influence of several important hyper-parameters on the performance of SHGNN while keeping other parameters fixed. Figure 6 presents the results in CP task, and other results are in Appendix A.4.

Number of spatial groups n_s/n_r . We first analyze the effects made by the number of partitioned sectors n_s and rings n_r . By increasing n_s/n_r to partition more subspaces, SHGNN can model more diverse spatial relations and further capture the diversity of spatial heterophily in a more fine-grained manner. However, over-dense partitions bring no further improvements but even slight performance declines. A possible explanation is that some subspaces contain too few neighbors to support its representation learning.

Partition head number M_s/M_r . We also study the impact of head number M_s/M_r in the multi-head partition strategy. It can be observed that, compared to the single partition ($M_s, M_r = 1$), SHGNN using such a strategy ($M_s, M_r = 2$) can evidently perform better, thanks to the complementing role between two heads. When M_s and M_r continue to increase, the model generally gets fewer further improvements, and too many heads with additional redundancy may sometimes result in performance degradation. Thus, we recommend to set a small head number but larger than 1 (e.g., 2), which is good enough while keeping efficiency.

6 RELATED WORK

Here we briefly review two related topics: GNNs for urban applications and GNNs with heterophily.

GNNs for Urban Applications. As a powerful approach to representing relational data, GNN models are widely adapted in recent studies to learn on urban graphs, and achieve remarkable performance in various applications, including traffic forecasting [9, 25, 31, 33, 37, 43, 50], bike demand prediction [19], region embedding [40, 51], regional economy prediction [45] and special region discovery [44]. There are a few works that tend to encode the location information in GNNs’ message passing process [4, 20] in a special domain (POI relation prediction). But these methods fail to generalize to urban graphs with heterophily, which may significantly degrade their performance in other urban applications.

GNNs with Heterophily. Our research is also related to the studies of graph heterophily. Here we only make a brief introduction to heterophilic GNNs and refer readers to a recent comprehensive survey [53]. Such approaches solve the heterophily problem basically in the following two ways. The first branch is to reconstruct the homophilic neighborhood with similar nodes on the graph measured by different criteria. The used criteria include the structural similarity defined by the distance in latent space [29] or degree sequence [34], the difference of attention scores [26], cosine similarity of node attributes [15, 16], nodes’ ability to mutually represent each other [21] and so on. However, as pointed out by He et al. [11], these methods will damage the network topology and tamper with the original real-world dependencies on the urban graph.

Another branch tends to modify the GNN architecture to handle the difference information on heterophilic graphs, in contrast to the Laplacian smoothing [38] of typical GNNs, such as processing neighbors in different classes separately [7, 8], explicitly aggregating features from higher-order neighbors in each layer [1, 56], allowing the high-frequency information by passing signed messages [2, 28, 41, 46, 47], and combining outputs of each layer (including ego features) to also empower the GNNs with a high-pass ability [3, 5, 56]. However, most of these methods do not consider the diversity of dissimilarity distribution between the central node and different neighbors, especially with different spatial relations.

7 CONCLUSION

In this paper, we studied the unique spatial heterophily of the urban graph and developed a spatial heterophily-aware graph neural network. We designed a spatial diversity score to uncover the diversity of heterophily at different spatial locations in the neighborhood, and showed the limitation of existing GNNs for handling diverse heterophily distributions on urban graphs. Further, motivated by the analysis that spatially close neighbors present a more similar mode of heterophily, we proposed a novel method, named SHGNN, which can group spatially close neighbors together and separately process each group with less diversity inside, to tackle the spatial heterophily in a divide-and-conquer way. Finally, extensive evaluations demonstrate the effectiveness of our approach.

ACKNOWLEDGMENTS

This work is supported in part by Foshan HKUST Projects (FSUST21-FYTRI01A, FSUST21-FYTRI02A).

REFERENCES

- [1] Sami Abu-El-Hajja, Bryan Perozzi, Amol Kapoor, Nazanin Alipourfard, Kristina Lerman, Hrayr Harutyunyan, Greg Ver Steeg, and Aram Galstyan. 2019. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In *ICML*. PMLR, 21–29.
- [2] Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. 2021. Beyond low-frequency information in graph convolutional networks. In *AAAI*, Vol. 35. 3950–3957.
- [3] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. 2020. Simple and deep graph convolutional networks. In *ICML*. PMLR, 1725–1735.
- [4] Yile Chen, Xiucheng Li, Gao Cong, Cheng Long, Zhifeng Bao, Shang Liu, Wanli Gu, and Fuzheng Zhang. 2021. Points-of-interest relationship inference with spatial-enriched graph neural networks. *PVLDB* 15, 3 (2021), 504–512.
- [5] Eli Chien, Jianhao Peng, Pan Li, and Oljica Milenkovic. 2021. Adaptive Universal Generalized PageRank Graph Neural Network. In *ICLR*.
- [6] Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *NeurIPS* 26 (2013).
- [7] Enyan Dai, Shijie Zhou, Zhimeng Guo, and Suhang Wang. 2022. Label-Wise Graph Convolutional Network for Heterophilic Graphs. In *LoG*. PMLR, 26–1.
- [8] Lun Du, Xiaozhou Shi, Qiang Fu, Xiaojun Ma, Hengyu Liu, Shi Han, and Dongmei Zhang. 2022. GBK-GNN: Gated Bi-Kernel Graph Neural Networks for Modeling Both Homophily and Heterophily. In *WWW*. 1550–1558.
- [9] Zheng Fang, Qingqing Long, Guojie Song, and Kunqing Xie. 2021. Spatial-temporal graph ode networks for traffic flow forecasting. In *SIGKDD*. 364–373.
- [10] Jindong Han, Hao Liu, Haoyi Xiong, and Jing Yang. 2022. Semi-Supervised Air Quality Forecasting via Self-Supervised Hierarchical Graph Neural Network. *TKDE* (2022).
- [11] Dongxiao He, Chundong Liang, Huixin Liu, Mingxiang Wen, Pengfei Jiao, and Zhiyong Feng. 2022. Block modeling-guided graph convolutional neural networks. In *AAAI*, Vol. 36. 4022–4029.
- [12] Yifan Hou, Jian Zhang, James Cheng, Kaili Ma, Richard T. B. Ma, Hongzhi Chen, and Ming-Chang Yang. 2020. Measuring and Improving the Use of Graph Information in Graph Neural Networks. In *ICLR*.
- [13] Tobias Skovgaard Jepsen, Christian S Jensen, and Thomas Dyhre Nielsen. 2020. Relational fusion networks: Graph convolutional networks for road networks. *TITS* 23, 1 (2020), 418–429.
- [14] Junteng Jia and Auston R Benson. 2020. Residual correlation in graph neural network regression. In *SIGKDD*. 588–598.
- [15] Di Jin, Zhizhi Yu, Cuiying Huo, Rui Wang, Xiao Wang, Dongxiao He, and Jiawei Han. 2021. Universal graph convolutional networks. *NeurIPS* 34 (2021), 10654–10664.
- [16] Wei Jin, Tyler Derr, Yiqi Wang, Yao Ma, Zitao Liu, and Jiliang Tang. 2021. Node similarity preserving graph convolutional networks. In *WSDM*. 148–156.
- [17] Dongkwan Kim and Alice Oh. 2020. How to Find Your Friendly Neighborhood: Graph Attention Design with Self-Supervision. In *ICLR*.
- [18] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
- [19] Guanyao Li, Xiaofeng Wang, Gunarto Sindoro Njoo, Shuhan Zhong, S-H Gary Chan, Chih-Chieh Hung, and Wen-Chih Peng. 2022. A Data-Driven Spatial-Temporal Graph Neural Network for Docked Bike Prediction. In *ICDE*. IEEE, 713–726.
- [20] Shuangli Li, Jingbo Zhou, Tong Xu, Hao Liu, Xinjiang Lu, and Hui Xiong. 2020. Competitive analysis for points of interest. In *SIGKDD*. 1265–1274.
- [21] Xiang Li, Renyu Zhu, Yao Cheng, Caihua Shan, Siqiang Luo, Dongsheng Li, and Weining Qian. 2022. Finding global homophily in graph neural networks when meeting heterophily. In *ICML*. PMLR, 13242–13256.
- [22] Yuxuan Liang, Kun Ouyang, Junkai Sun, Yiwei Wang, Junbo Zhang, Yu Zheng, David Rosenblum, and Roger Zimmermann. 2021. Fine-grained urban flow prediction. In *WWW*. 1833–1845.
- [23] Derek Lim, Felix Hohne, Xiuyu Li, Sijia Linda Huang, Vaishnavi Gupta, Omkar Bhalerao, and Ser Nam Lim. 2021. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. *NeurIPS* 34 (2021), 20887–20902.
- [24] Derek Lim, Xiuyu Li, Felix Hohne, and Ser-Nam Lim. 2021. New benchmarks for learning on non-homophilous graphs. *arXiv preprint arXiv:2104.01404* (2021).
- [25] Fan Liu, Hao Liu, and Wenzhao Jiang. 2022. Practical Adversarial Attacks on Spatiotemporal Traffic Forecasting Models. In *NeurIPS*, Vol. 35. 19035–19047.
- [26] Meng Liu, Zhengyang Wang, and Shuiwang Ji. 2021. Non-local graph neural networks. *TPAMI* (2021).
- [27] Yu Liu, Xin Zhang, Jingtao Ding, Yanxin Xi, and Yong Li. 2023. Knowledge-infused contrastive learning for urban imagery-based socioeconomic prediction. In *WWW*. 4150–4160.
- [28] Sitao Luan, Chenqing Hua, Qincheng Lu, Jiaqi Zhu, Mingde Zhao, Shuyuan Zhang, Xiao-Wen Chang, and Doina Precup. 2021. Is Heterophily A Real Nightmare For Graph Neural Networks To Do Node Classification? *arXiv preprint arXiv:2109.05641* (2021).
- [29] Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. 2019. Geom-GCN: Geometric Graph Convolutional Networks. In *ICLR*.
- [30] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *SIGKDD*. 701–710.
- [31] Xuan Rao, Hao Wang, Liang Zhang, Jing Li, Shuo Shang, and Peng Han. 2022. FOGS: First-Order Gradient Supervision with Learning-based Graph for Traffic Flow Forecasting. In *IJCAI*.
- [32] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).
- [33] Chao Song, Youfang Lin, Shengnan Guo, and Huaiyu Wan. 2020. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *AAAI*, Vol. 34. 914–921.
- [34] Susheel Suresh, Vinith Budde, Jennifer Neville, Pan Li, and Jianzhu Ma. 2021. Breaking the Limit of Graph Neural Networks by Improving the Assortativity of Graphs with Local Mixing Patterns. In *SIGKDD*. 1541–1551.
- [35] Waldo R Tobler. 1970. A computer movie simulating urban growth in the Detroit region. *Economic geography* 46, sup1 (1970), 234–240.
- [36] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *ICLR*.
- [37] Zhaonan Wang, Renhe Jiang, Hao Xue, Flora D Salim, Xuan Song, and Ryosuke Shibasaki. 2022. Event-Aware Multimodal Mobility Nowcasting. In *AAAI*, Vol. 36. 4228–4236.
- [38] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *ICML*. PMLR, 6861–6871.
- [39] Ning Wu, Xin Wayne Zhao, Jingyuan Wang, and Dayan Pan. 2020. Learning effective road network representation with hierarchical graph neural networks. In *SIGKDD*. 6–14.
- [40] 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 *IJCAI*.
- [41] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. 2022. Beyond low-pass filtering: Graph convolutional networks with automatic filtering. *TKDE* (2022).
- [42] Yanxin Xi, Tong Li, Huangdong Wang, Yong Li, Sasu Tarkoma, and Pan Hui. 2022. Beyond the First Law of Geography: Learning Representations of Satellite Imagery by Leveraging Point-of-Interests. In *WWW*. 3308–3316.
- [43] Tong Xia, Junjie Lin, Yong Li, Jie Feng, Pan Hui, Funing Sun, Diansheng Guo, and Depeng Jin. 2021. 3dgc: 3-dimensional dynamic graph convolutional network for citywide crowd flow prediction. *TKDD* 15, 6 (2021), 1–21.
- [44] Congxi Xiao, Jingbo Zhou, Jizhou Huang, Hengshu Zhu, Tong Xu, Dejing Dou, and Hui Xiong. 2023. A Contextual Master-Slave Framework on Urban Region Graph for Urban Village Detection. In *ICDE*.
- [45] Fengli Xu, Yong Li, and Shusheng Xu. 2020. Attentional multi-graph convolutional network for regional economy prediction with open migration data. In *SIGKDD*. 2225–2233.
- [46] Yujun Yan, Milad Hashemi, Kevin Swersky, Yaoqing Yang, and Danai Koutra. 2021. Two sides of the same coin: Heterophily and oversmoothing in graph convolutional neural networks. *arXiv preprint arXiv:2102.06462* (2021).
- [47] Liang Yang, Mengzhe Li, Liyang Liu, Chuan Wang, Xiaochun Cao, Yuanfang Guo, et al. 2021. Diverse message passing for attribute with heterophily. *NeurIPS* 34 (2021), 4751–4763.
- [48] Yongyi Yang, Tang Liu, Yangkun Wang, Jinjing Zhou, Quan Gan, Zhewei Wei, Zheng Zhang, Zengfeng Huang, and David Wipf. 2021. Graph neural networks inspired by classical iterative algorithms. In *ICML*. PMLR, 11773–11783.
- [49] Fei Yi, Zhiwen Yu, Fuzhen Zhuang, and Bin Guo. 2019. Neural Network based Continuous Conditional Random Field for Fine-grained Crime Prediction.. In *IJCAI*. 4157–4163.
- [50] Haitao Yuan, Guoliang Li, Zhifeng Bao, and Ling Feng. 2021. An effective joint prediction model for travel demands and traffic flows. In *ICDE*. IEEE, 348–359.
- [51] Mingyang Zhang, Tong Li, Yong Li, and Pan Hui. 2021. Multi-view joint graph representation learning for urban region embedding. In *IJCAI*. 4431–4437.
- [52] Gongpei Zhao, Tao Wang, Yidong Li, Yi Jin, Congyan Lang, and Songhe Feng. 2022. Neighborhood Pattern Is Crucial for Graph Convolutional Networks Performing Node Classification. *TNNLS* (2022).
- [53] Xin Zheng, Yixin Liu, Shirui Pan, Miao Zhang, Di Jin, and Philip S Yu. 2022. Graph neural networks for graphs with heterophily: A survey. *arXiv preprint arXiv:2202.07082* (2022).
- [54] Jingbo Zhou, Tao Huang, Shuangli Li, Renjun Hu, Yanchi Liu, Yanjie Fu, and Hui Xiong. 2021. Competitive relationship prediction for points of interest: A neural graphlet based approach. *TKDE* 34, 12 (2021), 5681–5692.
- [55] Jiong Zhu, Ryan A Rossi, Anup Rao, Tung Mai, Nedim Lipka, Nesreen K Ahmed, and Danai Koutra. 2021. Graph neural networks with heterophily. In *AAAI*, Vol. 35. 11168–11176.
- [56] Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. 2020. Beyond homophily in graph neural networks: Current limitations and effective designs. *NeurIPS* 33 (2020), 7793–7804.

A APPENDIX

A.1 Datasets Construction

In this section, we make a supplementary explanation of the construction of three real-world datasets in three tasks.

A.1.1 Commercial Activeness Prediction (CAP). Following many previous studies (e.g. [22, 37]) that divide a city into grids for various urban analyses, we also divide Shenzhen city into a set of $128m \times 128m$ grids, which are regarded as the regions in this task. We construct two groups of region features for commercial activeness prediction, which are POI features and satellite image features.

A. POI Features. We construct three types of POI features from the POI data, to reflect a region’s functionality, which is highly correlated with commercial activeness [42].

(a) *Category Distribution.* For each region, we compute the ratios of different categories of POIs, to obtain the category distribution histogram as a feature vector. The total number of POIs is also appended in this vector. In our experiments, we consider the following 23 categories: Food Service, Hotel, Shopping Place, Life Service, Beauty Industry, Scenic Spot, Leisure and Entertainment, Sports and Fitness, Education, Cultural Media, Medicine, Auto Service, Transportation Facility, Financial Service, Real Estate, Company, Government Apparatus, Entrance and Exit, Topographical Object, Road, Railway, Greenland, and Bus Route. To include more surrounding information, we also compute the distribution in the 3×3 regions centered by the given region.

(b) *POI Radius.* This feature is defined as the shortest distance between a certain type of POIs and the region, such as hospital radius. The distance is discretized into different buckets ($< 0.5km$, $0.5 \sim 1.5km$, $1.5 \sim 3km$ and $> 3km$). We calculated 15 kinds of radius in total: Hospital, Clinic, College, School, Bus Stop, Subway Station, Airport, Train Station, Coach Station, Shopping Mall, Supermarket, Market, Shop, Police Station, and Scenic Spot.

(c) *Perfect Degree of Living Facility.* It’s a binary index that reflects whether residents can satisfy their daily consumption demands around the region. It will be assigned one if there are all the following types of facilities within $1km$ of the current region, which are Medical Service, Shopping Place, Sports Venue, Education Service, Food Service, Financial Service, Communication Service, Public Security Organ and Transportation Facility. We select these types of POIs according to the official document⁴.

B. Satellite Image Features. The satellite image data are 3-channel 256×256 RGB images with 0.5 spatial resolution. We use the pre-trained VGG16 [32] with the last two dense layers removed, to extract semantic features from the satellite image of each region.

A.1.2 Crime Prediction (CP). We provide more details about the Manhattan dataset. As introduced in [40, 51]: (1) *Census block data* contains the boundaries of 180 regions split by streets in Manhattan. (2) *Taxi trip data* records around 10 million taxi trips during one month among regions. (3) *POI data* includes information of nearly 20 thousand POIs, which can be divided into 14 categories. (4) *Crime data* consists of more than 35 thousand crime records during one year in these 180 regions. We construct 14-dimensional POI category distribution region features similar to CAP task.

A.1.3 Dangerous Road Section Detection (DRSD). We further introduce the road data and traffic accident record data used to build the Los Angeles dataset. The road data describes the location and connecting relationship of road sections in a city. It is used to build the graph of road networks, where nodes are road sections, and edges are added between connected sections. For each node, we apply Deepwalk [30] to generate the embedding as node features. As for the accident data, it records the occurrence time and location of nearly 20 thousand car accidents in December 2021.

A.2 Implementations

We first introduce some basic settings (e.g., input and output layers) for three tasks. For CAP task, since the input POI and satellite image features are from different modalities, we apply two dense layers with 64 hidden units to transform them into the same space, and concatenate them together to obtain the initial representation (i.e., $h_i^{(0)}$) of each region. For CP task, the 14 POI category features are directly used as the initial representation. For DRSD task, the node representation is initialized by the $64-d$ embedding vector learned by Deepwalk algorithm [30], where we set the walk length as 10, the number of random walk as 80, and the window size as 5. As for the output layer, we adopt Linear Regression to get the final prediction in CAP and CP task. For the node classification in DRSD task, we use Logistic Regression to obtain the output probability. Moreover, for all three tasks, we use Adam optimizer for model training with the learning rate set to 0.001.

Next, we introduce other hyper-parameter settings in SHGNN. For rotation-scaling spatial aggregation module, we set the head number $M_s = M_r = 2$ to divide neighbors from both the direction and distance view. Specifically, we partition $n_s = 4$ sectors under each head and the rotation angle is set to 45 degrees for all three tasks. In the distance view, we set different ring numbers in three tasks. For CAP task and CP task, we partition $n_r = 3$ rings per head, and the interval of each distance bucket is set to $1.5km$ (under the head $m = 1$) and $2.5km$ (head $m = 2$), while for DRSD task, we set $n_r = 2$ with $100m$ (head $m = 1$) and $200m$ (head $m = 2$) as bucket intervals. The hidden size is set to 32 for all the tasks. For heterophily-sensitive spatial interaction module, the hidden size of the two dense layers used to fuse two spatial views are set to 32, 128, and 32 for CAP, CP, and DRSD task, respectively. We apply one SHGNN layer between the input and output layer to learn the node representation based on the urban graph.

A.3 Baseline Descriptions

- **GCN** [18] is a classic message passing homophilic GNN model.
- **GAT** [36] uses self-attention mechanism in the feature aggregation. It also implicitly assumes that the graph is homophilic.
- **Mixhop** [1] is a GNN model for heterophilic graphs. It directly aggregates features from higher-order neighbors, which can identify feature differences among neighbors.
- **FAGCN** [2] is a heterophily GNN that extends the message from neighbors to be signed value. Its aggregation process can capture the high-frequency signals in the neighborhood.
- **NLGCN** [26] tackles the heterophily by rebuilding the homophilic neighborhood for similar nodes. The similarity of two nodes is measured by their attention scores to a calibration vector.

⁴<http://www.mohurd.gov.cn/wjfb/201811/W02018113004480.pdf>

Table 3: Efficiency comparison in the commercial activeness prediction task.

	GCN	GAT	Mixhop	FAGCN	NLGCN	GBKGN	GPRGNN	PRIM	SAGNN	KnowCL	SHGNN
Training time (s / epoch)	0.054	0.125	0.113	0.105	0.070	0.222	0.057	0.172	0.474	0.017	0.392
Inference time (s / instance)	0.003	0.004	0.033	0.005	0.031	0.005	0.003	0.006	0.008	0.001	0.018

- **GPRGNN** [5] combines intermediate representations from each GNN layer (including ego features). It achieves remarkable performance on general heterophilic graphs.
- **GBKGN** [8] tackles the graph heterophily problem by applying two different feature transformations for homophily node pairs and heterophily node pairs, respectively.
- **SAGNN** [20] includes the spatial information of neighbors in feature aggregation. It considers neighbors' direction by splitting sectors, and processes their relative position by dividing grids.
- **PRIM** [4] is a GNN-based framework working on POI graphs, with the consideration of geographical influence. When aggregating neighbor features, it modifies the attention score with a spatial distance score calculated by radial basis function (RBF).
- **KnowCL** [27] is specifically designed for socioeconomic prediction. It uses GCN encoder to extract multiple knowledge from an urban knowledge graph, and infuse it to region's visual representation through cross-modality contrastive learning.
- **NNCCRF** [49] is a crime prediction method. It integrates Continuous Conditional Random Field (CCRF) with neural network to model region interactions for crime prediction.
- **RFN** [13] is an improved GCN designed for prediction tasks on road networks. It simultaneously captures two interdependent views of relations, between both intersections and road segments.

A.4 Additional Experimental Results

Efficiency. We also conducted an experiment in CAP task to evaluate the efficiency of SHGNN. We compare SHGNN and all baselines in this task, in terms of the training time per epoch and the inference time of an instance. As shown in Table 3, the efficiency of SHGNN is affordable (0.392 seconds for training one epoch, and 0.018 seconds for inferring one instance), since our model can handle spatial heterophily on the urban graph, and achieves significant performance improvements over all baselines.

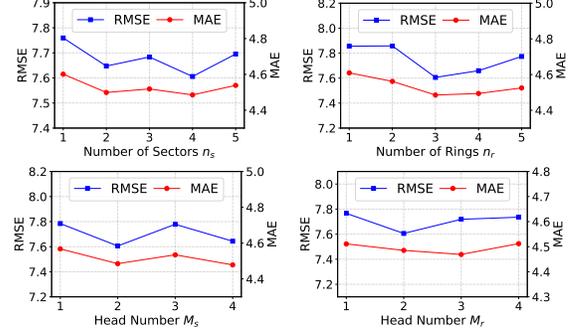
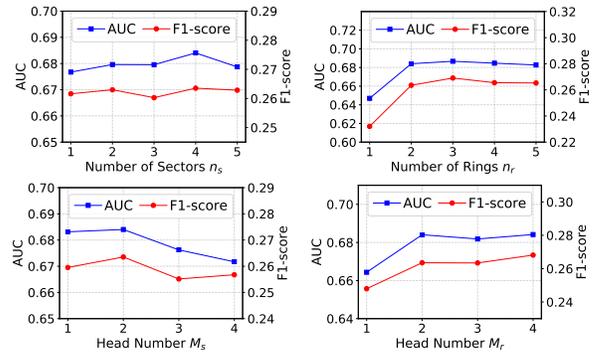
Parameter Analysis. The results of parameter analysis in CAP and DRSD task are shown in Figure 7 and 8.

A.5 Complexity Analysis

We further analyze the complexity of SHGNN. First, in the rotation-scaling spatial aggregation module, the computation cost comes from the spatial-aware aggregation process. Its complexity is:

$$O((M_s n_s + M_r n_r) |\mathcal{V}| d^2 + (M_s + M_r) |\mathcal{E}| d),$$

where $|\mathcal{V}|$ and $|\mathcal{E}|$ denote the number of nodes and edges, and d denotes the node feature dimension. n_s and n_r denote the number of sectors and rings. M_s and M_r are head numbers of the multi-head sector partition in direction view and ring partition in distance view. To explain, in the direction view, the term $O(M_s n_s |\mathcal{V}| d^2)$ is the cost of feature transformation for nodes in n_s different sectors, under M_s heads of sector partitions. And $O(M_s |\mathcal{E}| d)$ corresponds to the cost of the feature aggregation process. Similarly, the part $O(M_r n_r |\mathcal{V}| d^2 + M_r |\mathcal{E}| d)$ is for the distance view.

**Figure 7: Parameter analysis in CAP task.****Figure 8: Parameter analysis in DRSD task.****Table 4: Mathematical notations.**

Notation	Description
\mathcal{G}	The urban graph
$\mathcal{V}, \mathcal{E}, \mathbf{X}$	The node set, edge set and node attributes of \mathcal{G}
$\mathcal{N}(v_i)$	The neighborhood of node v_i on \mathcal{G}
\mathbf{h}_i	The representation of node v_i
s_k, r_k	The partitioned direction sector and distance ring
$\mathcal{N}_{s_k}(v_i)$	The direction-aware neighborhood of node v_i
$\mathcal{N}_{r_k}(v_i)$	The distance-aware neighborhood of node v_i
\mathbf{z}_{i,s_k}	The representation of sector s_k around node v_i
\mathbf{z}_{i,r_k}	The representation of ring r_k around node v_i

In the heterophily-sensitive spatial interaction module, the complexity of commonality and discrepancy kernel are both:

$$O((M_s n_s + M_r n_r) |\mathcal{V}| d^2 + (M_s n_s^2 + M_r n_r^2) |\mathcal{V}| d),$$

where the first term is to extract the common knowledge / difference information, and the second term is the cost to compute the commonality / dissimilarity degree among sectors and among rings. Then, the computation cost of attentive component selection is also $O((M_s n_s + M_r n_r) |\mathcal{V}| d^2)$. Overall, the total complexity of SHGNN is the combination of the above two modules:

$$O((M_s n_s + M_r n_r) |\mathcal{V}| d^2 + (M_s n_s^2 + M_r n_r^2) |\mathcal{V}| d + (M_s + M_r) |\mathcal{E}| d).$$