RESEARCH-ARTICLE

M3: Recommendation via Attention-Graph Cluster Q-Learning with Multi-Scale Spatial Heterogeneity for Multi-Purpose, Multi-Stakeholder Green Attractions in Transportation

SHIHYU LAI, National Taiwan University, Taipei, Taiwan

TZUHSIN HSIEH, Delft University of Technology, Delft, Zuid-Holland, Netherlands

PEICHI TSAI, University College London, London, U.K.

CHAOCHUN KUNG, Architectural Association (AA) School of Architecture, London, U.K.

SING KAI LING

HSUN-PING HSIEH, National Cheng Kung University, Tainan, Taiwan

Published: 03 November 2025

SIGSPATIAL '25: 33rd ACM International Conference on Advances in Geographic Information Systems
November 3 - 6, 2025
MN, Minneapolis, USA

Conference Sponsors:
SIGSPATIAL

M3: Recommendation via Attention-Graph Cluster Q-Learning with Multi-Scale Spatial Heterogeneity for Multi-Purpose, Multi-Stakeholder Green Attractions in Transportation

Shih-Yu, Lai
Department of Computer Science and Information Engineering, National Taiwan University
Taipei, Taiwan
akinesia112@gmail.com

Tzu-Hsin, Hsieh
Faculty of Electrical Engineering, Mathematics and Computer Science, Technische Universiteit Delft
Delft, Nederland
celinehsieh68@gmail.com

Pei-Chi, Tsai
The Bartlett Centre for Advanced Spatial Analysis, University College London
London, UK
a230129520a@gmail.com

Chao-Chun, Kung
OAOA Architecture Associates
London, UK
kungjack43@gmail.com

Sing-Kai, Ling
104 Corporation
Taipei, Taiwan
juliuslsk@gmail.com

Hsun-Ping Hsieh*
Department of Electrical Engineering, National Cheng Kung University
Tainan, Taiwan
hphsieh@mail.ncku.edu.tw

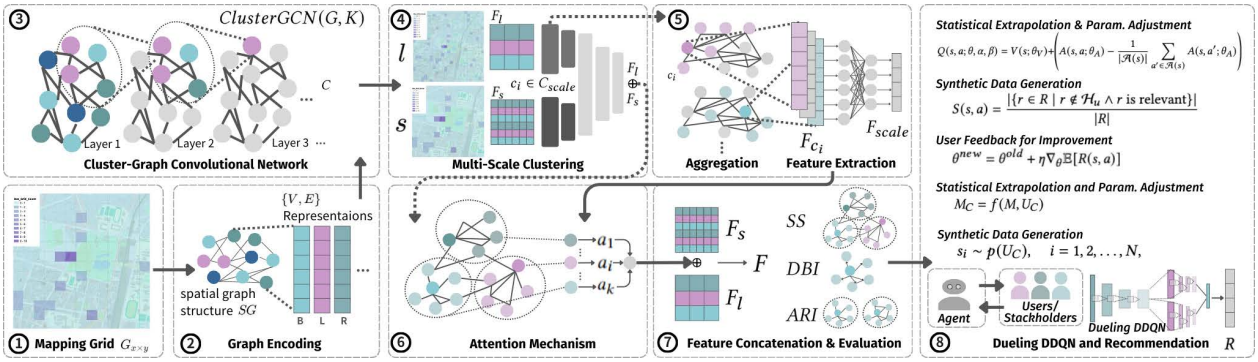

Figure 1: The architecture enhances graph modeling by boosting intra-cluster similarity and uncover patterns in Cluster-GCN. Attention mechanisms refine this clustering by weighting edges, capturing nuanced structures and improving accuracy. The Dueling DDQN generate the contextually relevant and personalized recommendations by user interactions and feedbacks.

Abstract

With growing environmental concerns and the push for sustainable urban development, promoting green travel has become a critical initiative. Urban transit systems face the challenge of integrating green initiatives with efficient transport routes, while sophisticated graph modeling enhances travel efficiency. However, blending historical and contemporary elements introduces complex variations in traffic networks, complicating feature extraction and clustering for information retrieval due to multi-scale spatial heterogeneity. Traditional methods often overlook key nuances by oversimplifying data relationships. We proposed **M3** and validated the integration of GIS-based Attention-Cluster-GCN with Dueling Double Deep Q Network across various cities, enhancing urban travel with detailed information on green attraction recommendations, considering the usage of **M**ulti-Purpose and **M**ulti-Stakeholder for **M**ulti-Scale Spatial Heterogeneity scenarios. Utilizing Attention-Based Reinforcement Graph Clustering refines modeling and emphasizes vital connections, enhancing personalized recommendation precision and clustering performance. Our method surpasses both conventional and advanced GNN methods, even in graph convolution-based deep reinforcement learning, achieving superior cluster separation and accuracy. Our sampling and ablation studies confirm the pivotal role of the attention mechanism and multi-scale features, showing a significant performance decline without attention. Our findings underscore the potential of graph clustering in making public transport more engaging and aligned with green attractions policies by recommendations, even amidst significant spatial heterogeneity.

*Corresponding author

SIGSPATIAL '25, Minneapolis, MN, USA

ACM ISBN 979-8-4007-2086-4/2025/11
https://doi.org/10.1145/3748636.3762708

CCS Concepts

• **Computing methodologies** → **Artificial intelligence**.

Keywords

Graph Clustering, Reinforcement Learning, Attention Mechanism, Public Transport, Spatial Heterogeneity, Recommendation systems

ACM Reference Format:

Shih-Yu, Lai, Tzu-Hsin, Hsieh, Pei-Chi, Tsai, Chao-Chun, Kung, Sing-Kai, Ling, and Hsun-Ping Hsieh. 2025. M3: Recommendation via Attention-Graph Cluster Q-Learning with Multi-Scale Spatial Heterogeneity for Multi-Purpose, Multi-Stakeholder Green Attractions in Transportation. In *The 33rd ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL '25), November 3–6, 2025, Minneapolis, MN, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3748636.3762708

1 Introduction

Urban areas with abundant greenery—such as parks, scenic walkways, and eco-friendly spaces—offer significant value to both residents and visitors. We refer to such locations as "**Green Attractions**". Imagine a local citizen or tourist planning an eco-conscious day trip: their travel needs vary widely across city zones. In well-connected cities like New York or Tokyo, subways provide easy access to green attractions. However, in many smaller or developing cities, *public transport is limited to buses due to the* ***absence of a subway system***, which complicates access the green attractions due to inconsistent routes and coverage.

The first problem comes from the bottleneck of **Multi-Scale Spatial Heterogeneity**, referring to the varying characteristics of urban areas that can include differences in land use, population density, and infrastructure among various city zones. These discrepancies present unique challenges for transportation and urban planning, especially when integrating sustainable initiatives like green travel. Second, **Multi-Purpose** indicates that the green attraction recommendation system serves diverse purposes, catering to both environmental sustainability and recreational planning. For instance, a user might wish to visit green attractions for relaxation, exercise, or education, each requiring distinct itineraries and transport planning. Third, the system is designed to integrate and balance the needs of **Multi-Stakeholder**, including tourists, residents, public transport passengers, and end-users. For example, tourists might prioritize maximizing public transport utilization, while residents seek convenience and eco-friendliness.

Our system tackles these challenges by mapping green attractions to bus stations based on their locations, adapting recommendations to time and place, and providing several optimal places through clustering, considering road networks and green spaces to enhance recommendation accuracy in the diverse spatial contexts of different urban environments. This approach not only meets diverse travel needs but also supports sustainable travel initiatives, navigating the complexities of cities that blend historic neighborhoods with modern developments and have varied traffic patterns.

We introduce **M3**, a GIS-based Attention based Cluster-GCN [7] to enhance feature extraction with urban spatial heterogeneity, with Dueling Double Deep Q Network (Dueling DDQN) [13, 26] to improve the accuracy of green attraction recommendations in different user groups with policy adaptation. The system promotes public transport and environmental awareness via a spatial graph data pipeline for Point-Of-Interest (POI) recommendation while handling the complex relationships in multi-scale spatial heterogeneity. Our model accommodates such multi-purpose scenarios by providing clustered recommendations that adapt to user intent and spatiotemporal factors, ensuring that results align with varied personal or societal goals. By incorporating stakeholder objectives through adaptive clustering and reinforcement learning (RL) techniques, our model optimizes for both policy and user-centered outcomes. This ensures a cohesive framework that aligns with broader urban sustainability with multiple UN Sustainable Development Goals (**SDGs**)[19], particularly targeting SDG11 for sustainable cities and communities and SDGs 9 by enhancing infrastructural sustainability and fostering a positive environment for local development.

For example, in cities served only by buses, sharp variations in land use and population density ("multi-scale spatial heterogeneity"), diverse user purposes (e.g. exercise vs. sightseeing), and competing stakeholder needs (tourists vs. residents) make green-space recommendations challenging. Our **goal** is to convert a spatial graph of bus stops and green POIs (**input**) into personalized, context-aware ranked suggestions (**output**). With simplified example of parks A, B, C and bus stops 1 & 2, we first use an Attention-Cluster-GCN to group POIs by different geographic scales, then encode a user's intent (e.g. jogging vs. photography), and finally train a Dueling DDQN agent to learn top-k recommendations.

This method integrates advanced graph processing and Q-learning with green attraction recommendations for public transportation in multi-scale spatial heterogeneity, showcasing the potential for improved urban information management in tourism and transportation with 3 main contributions, including:

- **Refined Modeling with Attention and Dueling DDQN Mechanisms:** Utilizing attention mechanisms to weigh graph edges enhances structural detail capture and clustering accuracy. Dueling DDQN addresses the challenges of large, complex action spaces by using a differentiable exploration strategy that thoroughly evaluates all potential actions, enabling the system to uncover patterns in the data for more effective recommendations.
- **Enhanced Recommender by Multi-scale Clustering:** Our model enhances graph clustering with high intra-similarity and supports detailed information on green attractions. We adopts to environmental contexts at each bus stop, precisely recommending green attractions in transportation.
- **Scalability and Adaptability in Spatial Heterogeneity:** Designed to efficiently scale with urban data demands, our system adjusts to changes without performance loss.

2 Related Works

2.1 Bus System in Spatial Heterogeneity

Research on planning bus routes based on spatial feature heterogeneity is abundant [6, 15, 20, 25], yet studies focusing on city-specific or local characteristics for service-oriented approaches are rare. This gap highlights the potential for integrating local distinctiveness into public transportation planning, aiming to enhance both service quality and user experience. One study [9] proposes

a new zoning approach based on local traffic characteristics, addressing the gap in research on bus service zoning and advocating for a shift towards localized, service-oriented planning. This contrasts with the prevalent focus on spatial feature heterogeneity in bus route planning. Another piece of research [24] delves into customized bus demand, employing spatial dynamics to underscore the need for nuanced service strategies that consider a variety of factors, highlighting a move towards addressing specific community needs rather than just spatial heterogeneity. Additionally, an examination of urban customized bus companies suggests a time-dependent planning method to tackle route efficiency challenges [11], illustrating a departure from conventional planning based on spatial features alone and moving towards adaptive solutions that consider temporal and spatial demand variations.

2.2 Cluster-Graph Convolutional Network

Prior to Cluster-GCN. In spatial data mining and recommendation systems, graph analysis methods like FastGCN (Chen et al.[4]) or GraphSAGE (Hamilton et al.[12]) enhance analysis through neighbor aggregation or node sampling, accurately capturing graph data's inherent complexities and significantly outperforming traditional methods. Cluster-GCN (Graph Convolutional Network) , developed by Chiang et al. [7], enhances GNN scalability by partitioning graphs into smaller clusters using algorithms such as METIS [16] for effective mini-batch training. This method lowers computational needs and maintains graph integrity, broadening GNN's utility in fields like social network analysis and recommendation systems. Its key contributions are: **(1) Efficiency in Large Graphs:** Cluster-GCN's clustering significantly lowers memory and computational demands, allowing for effective training on large datasets. **(2) Preservation of Graph Structure:** It maintains structural integrity within clusters, unlike some sampling methods, enhancing performance and accuracy. **(3) Compatibility with Various GNN Architectures:** Its versatile clustering technique supports diverse GNN architectures, expanding its utility and influence.

Improvements over Previous Methods. Previous methods like GraphSAGE's neighbor sampling and FastGCN's layer-wise propagation struggled with computation efficiency and model accuracy, often sacrificing information integrity or failing to scale for large graphs. Cluster-GCN addresses these issues by ensuring structural integrity within each cluster, preventing information loss and enhancing training efficiency and accuracy on large-scale graphs, redefining GNN scalability standards. Following Cluster-GCN's debut, subsequent studies like GraphSAINT (Zeng et al. [28]) and VR-GCN (Chen et al. [5]) have refined graph analysis, targeting efficiency and accuracy across fields such as recommendation systems and bioinformatics. Yet, Cluster-GCN's major limitation lies in its struggle to balance global and local graph information, heavily influenced by the clustering algorithm's effectiveness. This issue, alongside the intensive computational demands of initial clustering and challenges with dynamic graphs, underscores the need for advancements in real-time, adaptable graph analysis solutions.

Limitations. Cluster-GCN significantly advances GNN research, addressing scalability and broadening use in large-scale graphs. Despite its advancements, Cluster-GCN faces limitations, notably its reliance on the clustering algorithm's quality. Performance can fluctuate with the graph's partitioning quality, leading to potential inaccuracies if it doesn't precisely reflect the graph's structure. Additionally, the initial clustering can be resource-intensive, posing challenges for large, complex graphs. Dynamic graphs requiring frequent cluster updates introduce further computational demands, potentially hindering the efficacy in real-time applications due to these evolving structures. The issues with graph clustering quality and adapting to dynamic graphs highlight the need for ongoing innovation for full GNN potential in analyzing complex systems.

Table 1: Statistics of Datasets.

City	Bus Stops	POIs	Green Centroids	100 m Grids (s)	500 m Grids (l)
Tainan, Taiwan	4,624	525,364	330,127	469,971	8,768
Hsinchu, Taiwan	3,328	482,749	263,473	10,415	416
Keelung, Taiwan	2,775	293,278	283,374	13,276	531
Cheongju, South Korea	2,107	142,365	200,256	94,043	3,761
Melaka, Malaysia	1,231	85,742	150,345	164,300	6,572
Chiang Mai, Thailand	956	68,590	310,789	4,020	161
Okinawa, Japan	1,650	97,834	220,501	488,108	9,124
New York, USA	14,855	850,123	400,678	77,431	3,135
London, UK	13,597	789,456	379,234	116,936	4,420
Tokyo, Japan	16,250	812,367	419,890	469,524	8,776

3 Method

Datasets. Three types of input data are mapped into grids according to their coordinates, including (1) Points of Interest (POIs) S from point-shaped urban amenities, (2) bus stations, and (3) centroid point locations calculated from the polygon-shaped data of plant species and green attractions. The dataset in Tainan, Hsinchu, Keelung, Cheongju, Melaka, Chiang Mai, Okinawa, New York, London, and Tokyo include a total of **57** categories across 10 cities with **bus stops**, **POIs of urban amenities**, and **centroid points of plant species and green attractions**, see the details in **Table 1**. These point data will all be referred to as POIs S in the following sections, while they were mapped into grids $G_{x\times y}$ in 100 m square for small scale s unit, and grids in 500 m square for large scale l unit. To determine the number of grids, divide the total area by the area of one grid using the formula: Number of Grids = $\frac{\text{Total Area in square meters}}{\text{Area of one grid in square meters}}$. To reduce data dependency, avoiding to relies heavily on high-quality, geo-spatially detailed datasets, which may not be universally available, the sparse POI data sampled from the sources include Google Map API, OpenStreetMap API, iNaturalist, web crawlers, and open data from local gvernments [1–3, 18]. These POIs S cover a broad range of categories such as dining venues, shopping places, public facilities, educational institutions, medical facilities, undesirable amenities, financial institutions, churches, and more. This data type contains a variety of common POIs in urban areas, sufficient to represent the heterogeneity between different areas.

Architecture. Each module in the architecture is grounded in real urban mobility challenges. For example, the grid mapping simplifies handling irregular city layouts; multi-scale clustering reflects the actual variety of green spaces; attention prioritizes context-relevant features; and Q-learning captures diverse preferences over time. See model training details in **Appendix A**.

The advancement of urban travel through the detailed recommendation of green attractions is predicated on the integration of sophisticated geospatial analysis with user-centric data interpretation. Our system utilizes a Cluster Graph Convolutional Network

(Cluster-GCN) enhanced with multi-scale clustering and an attention mechanism to achieve this integration (**Figure 1** and **Table 2**):

(1) POI Data and Grid Mapping: Urban Points of Interest (POIs), especially green attractions, are unevenly distributed and vary widely in form—from neighborhood parklets to large botanical gardens. To model this **diversity** efficiently, we discretize the city into spatial grids, which not only support geospatial reasoning but also enable uniform computation across variable urban landscapes. This approach makes it scalable to cities with complex and inconsistent road or POI layouts, especially in regions lacking subway systems. The decision to structure POI data into a grid format corresponds with the need for a scalable approach to data management that can accommodate the vast and varied nature of urban environments. This method enhances the system's ability to process and analyze data efficiently, ensuring that green attractions are accurately represented within the urban matrix. **(2) Multi-Scale Clustering in Cluster-GCN:** To **reflect the heterogeneity of urban green spaces**, we employ multi-scale clustering within a Cluster-GCN framework. This allows the system to distinguish between localized pocket parks and larger green zones, which users may access differently depending on their location, travel intent, and time constraints. The clustering model supports context-aware grouping of POIs, allowing for recommendations that match user preferences at the appropriate attraction **granularity** level. Multi-scale clustering allows the system to recognize and categorize these attractions at different scales, improving the granularity of the analysis and ensuring more precise recommendations. This approach not only enhances pattern recognition but also adapts to the diverse preferences of urban travelers. **(3) Attention Mechanism:** Not all features in a city graph are equally relevant to a user's intent. The attention mechanism highlights **key spatial features**—such as connectivity, environmental quality, or proximity to transit hubs—based on user queries or inferred interests. This selective focus not only improves clustering interpretability but also enables a personalized recommendation experience that adapts dynamically to user context. Incorporating an attention mechanism enables the system to focus on the most relevant features of the green attractions based on user queries. This focus is crucial for personalizing recommendations, which in turn, enhances user satisfaction and engagement with the recommended green spaces. By prioritizing significant features, the system tailors its outputs to the specific needs and interests of its users, promoting a more targeted exploration of urban greenery. **(4) Evaluation Metrics:** To ensure clustering is not only computationally valid but also **spatially and semantically meaningful**, we evaluate using Silhouette Score [22], Davies-Bouldin Index [8], and Adjusted Rand Index [14]. These metrics quantify how well POIs are grouped and whether clusters align with human-perceivable green zone patterns. High scores directly translate to higher quality of recommendation for different spatial scales. The use of sophisticated metrics enables rigorous evaluation of the clustering quality. These metrics are crucial for optimizing the system's performance and ensuring that the recommendations are both accurate and meaningful, thus directly contributing to enhanced urban travel experiences. **(5) Dueling DDQN and Recommendation:** Recommendation actions are guided by a Dueling Double Deep Q-Network (Dueling DDQN), which treats each green attraction suggestion as an action in a spatial decision space. Through interactions (e.g., clicks, rejections), the system effectively learned **preferences** from feedback. The dueling architecture efficiently separates the value of a state (e.g., user near a dense green cluster) from the advantage of each action (e.g., recommending a nearby eco-park), allowing more robust adaptation across users and city layouts. The RL model of the system learns from user interactions, such as which recommendations are accepted or ignored, and adjusts future recommendations accordingly. This feedback loop continuously refines the recommendation engine, improving user satisfaction and system efficiency over time.

Table 2: Attention based Cluster-GCN for plant species and bus station mapping with Dueling DDQN in multi-scale.

Algorithm 1	
Step	Description
	Input: S: POIs; $G_{x\times y}$: grid; K: clusters.
	Output: R: recommendations.
1	Input $S, G_{x\times y}, K$
2	$GridMap \leftarrow$ Map S to $G_{x\times y}$
3	$SG \leftarrow$ Convert $GridMap$ to graph
4	$\{V, E\} \leftarrow ENCODE(SG)$
5	$G \leftarrow GRAPH(L_p)$
6	$C \leftarrow ClusterGCN(G, K)$
7	Init F_l, F_s
8	For each scale scale $\in \{l, s\}$:
9	$C_{scale} \leftarrow ClusterGCN(\{V, E\}, scale)$
10	For $c_i \in C_{scale}$:
11	$F_{c_i} \leftarrow AGG(c_i)$
12	$F_{scale} \cup = \{F_{c_i}\}$
13	End For
14	$F_{scale} \leftarrow ATT(F_{scale}, scale)$
15	End For
16	$F \leftarrow F_l \oplus F_s$
17	Metrics $\alpha \leftarrow EVAL(F, \{SS, DBI, ARI\})$
18	Init Dueling DDQN with parameters θ_V, θ_A
19	Define reward function:
20	$R(s, a) = \lambda_p P(s, a) + \lambda_s S(s, a) + \lambda_u U(s, a)$
21	For each state $s \in F$:
22	$Q(s, a; \theta, \alpha, \beta)$
23	$= V(s; \theta_V) + \left(A(s, a; \theta_A) - \frac{1}{\lvert\mathcal{A}(s)\rvert} \sum_{a' \in \mathcal{A}(s)} A(s, a'; \theta_A)\right)$
24	Derive policy $\pi(s) = \text{softmax}\left(\frac{Q(s,a;\theta,\alpha,\beta)}{\tau}\right)$
25	Update parameters $\theta \leftarrow \theta + \eta \nabla_\theta \mathbb{E}[R(s, a)]$
26	End For
27	$R \leftarrow RECOMMEND(F)$ based on $\pi(s)$

These technical choices are driven by the need to effectively parse and prioritize urban green spaces, which are essential for improving the quality of urban life and promoting sustainable tourism practices. Our system's design is tailored to enhance the exploration and enjoyment of urban green spaces. By leveraging detailed graph analysis, spatial searches, and personalized recommendation, which supported by machine and deep learning, it seeks to transform urban travel into a more engaging and sustainable activity.

3.1 POI Data and Grid Mapping

Mapping POIs S to Grids $G_{x\times y}$. The mapping of POIs (S) onto the grid $G_{x\times y}$ involves several steps: (1)**Intersection:** For each polygon p, represented as an element in the set of polygons P, assign p to a grid cell g within the grid G if the centroid $C(p)$ of polygon p lies within grid cell g. (2)**Filtering:** Retain each polygon p in P

unless it is verifiably irrelevant or its data is incomplete. Specifically, remove a polygon if: The area $A(p) = 0$ and it lacks other defining attributes (e.g., historical significance or designated use). (3)**Area Calculation:** Compute the area $A(p)$ for each polygon p in P. (4)**Centroid Conversion:** Convert the geometric center of each polygon p into a centroid point $C(p)$. (5)**Area Calculation for Points:** For each grid cell g in G, calculate the total area contributed by points within the grid cell as $\sum_{p \in g} A(p) \cdot N(g)$, where $N(g)$ is the number of points within grid cell g. (6)**Total Area Calculation:** Calculate the total area for each grid cell g as $\sum_{p \in g} A(p)$. (7)**Area-Weighted Count:** Compute the area-weighted count for each grid cell g as $\frac{\sum_{p \in g} A(p) \cdot N(g)}{\sum_{p \in g} A(p)}$. P represent the set of all polygons. G represent the grid into which the urban space is divided, with $G_{x \times y}$ indicating specific grid cells. S denote the set of POIs S centroids. $A(p)$ denote the area of polygon p. $C(p)$ denote the centroid point of polygon p. $N(g)$ denote the number of points within grid cell g.

Graph Encoding. The grid map is converted into a graph format SG, encoding spatial relationships. We construct graph with vertices V represent the various POIs S, including bus stations, green attractions and plant centroids, and other urban amenities. Edges E between nodes are established based on spatial proximity and categorical relevance, emphasizing connections that are likely to influence urban mobility and green space accessibility. The graph, represented by V and E, processes clustering preparation. Each vertex V incorporates location coordination, counting points in each grid, and the spatial attributes. The Haversine formula calculates distances d between two POIs S to establish edges E:

$$d = 2r \arcsin\left(\sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\Delta\lambda}{2}\right)}\right)$$

d is the distance between two points on a sphere's surface, r is the sphere's radius (Earth's mean radius is about 6371 km), ϕ_1 and ϕ_2 are the points' latitudes in radians, $\Delta\phi = \phi_2 - \phi_1$ is the latitude difference, and $\Delta\lambda$ is the longitude difference in radians.

Data Augmentation. The system processes the encoding data into a graph that captures spatial features related to bus transportation and plant landscapes in urban spaces. Depending on data characteristics, distinct scale modules in clustering are applied. To enhance the model's performance and robustness: **(1) Sampling and Area Selection:** Used as split criteria to ensure diverse and comprehensive training samples. **(2) Oversampling and Undersampling** are applied to address imbalance dataset, ensuring that less frequent but significant categories such as specific types of green attractions or bus stations are appropriately represented.

3.2 Cluster-GCN with Multi-Scale Clustering

Our method (**Figure 1**) enhances higher intra-cluster similarity by grouping vertices into clusters, ideal for revealing hidden patterns in graph convolutional networks. The integration of attention mechanisms further refines clustering quality by weighting edges, capturing nuanced graph structures for improved performance.

Feature Initialization. Features at large (F_l) and small (F_s) scales are initialized to capture various levels of spatial information with: $F_l, F_s \leftarrow InitializeFeature(G, scale)$ indicating the process of initializing features for G at different scales. The Cluster-GCN algorithm is applied to the graph G, segmenting it into smaller sections based on similarity, with the designated number of clusters K,where G is the input graph and C_i represents each cluster formed within the graph: $\boldsymbol{G \leftarrow ClusterGCN(G, K) = \{C_1, C_2, \ldots, C_K\}}$.

Cluster-GCN Model Design. Given a graph $G = (V, E)$ where V and E represent vertices and edges, Cluster-GCN partitions G into K clusters for efficient processing. Let $S = \{S_1, S_2, \ldots, S_K\}$ denote the set of clusters. C_K represents the clusters generated by the Cluster-GCN algorithm for efficient graph processing, while S_K may denote an initial or specialized subset of clusters used for validation, testing, or other specific analytical purposes. The adjacency matrix A and feature matrix X of graph G are input into the model. The goal is to learn a function $f(\cdot)$ that maps node features X to labels Y with the graph structure for regularization: $\boldsymbol{H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)})}$,where $\tilde{A} = A + I_N$ (with I_N being the identity matrix), $\tilde{D}$ is the diagonal node degree matrix of $\tilde{A}$, $H^{(l)}$ is the activation in the l-th layer (with $H^{(0)} = X$), $W^{(l)}$ is the weight matrix for the l-th layer, and $\sigma(\cdot)$ represents the non-linear activation function.

Hyperparameters and Loss Design. Hyperparameters include the number of clusters K, learning rate η, and the dimensions of each hidden layer. The loss function is designed to minimize the difference between the predicted labels $\hat{Y}$ and the true labels Y, often using Cross-Entropy Loss for classification tasks: $\mathcal{L} = -\sum_{i=1}^{N}\sum_{c=1}^{M} Y_{ic}\log(\hat{Y}_{ic})$,where N is the number of nodes, M is the number of classes, Y_{ic} is the ground truth, and $\hat{Y}_{ic}$ is the predicted probability of node i belonging to class c. For mapping POIs S, introduce a spatial attention layer, where A' is the adjusted adjacency matrix focusing on more relevant spatial connections, and W_{att} is the weight matrix of attention mechanism: $\boldsymbol{A' = \text{softmax}(\text{LeakyReLU}(A \cdot W_{att}))}$. To accommodate the clustering of POIs S, the model could incorporate the multi-modal inputs: $\boldsymbol{F_{fusion} = \sigma(W_{fusion} \cdot [F_{spatial}; F_{cate}])}$,where $F_{spatial}$ and F_{cate} represent the features related to spatial and categorical data, respectively, and W_{fusion} is the weight matrix for the fusion layer. The tailored Cluster-GCN architecture becomes more suited for tasks requiring nuanced understanding of spatial and categorical data within urban environments.

Multi-Scale Clustering. For each scale, large (l) and small (s), the graph undergoes a distinct clustering process, with features F_{c_i} aggregated for each cluster c_i within that scale: $\boldsymbol{\forall\, scale \in \{l, s\}}$, $\boldsymbol{C_{scale} \leftarrow ClusterGCN(G, scale)}$. Here, C_{scale} represents the set of clusters obtained for each scale, and F_{c_i} symbolizes the aggregated features for cluster c_i: $\forall\, c_i \in C_{scale}, \quad F_{c_i} \leftarrow AGG(c_i)$. Given a set of features F_{c_i} for each cluster c_i, we update the feature set F_{scale} by: $F_{scale} \cup = \{F_{c_i}\}$. This denotes the aggregation of features from individual clusters into the comprehensive set F_{scale}.

3.3 Attention Mechanism

Attention Mechanism Model Design. Incorporating an attention mechanism within the Cluster-GCN framework refines structural details for tasks through specific edge emphasis. The formulations include attention coefficients, feature aggregation, hyperparameters, and loss function design. Given a set of node features F_{c_i} for each cluster c_i, it calculates attention coefficients for the edges, highlighting the importance of node features as follows: (1) **Attention Coefficients**: The coefficients between nodes i and j: $\boldsymbol{a_{ij}} = \text{LeakyReLU}\left(\mathbf{a}^T[\mathbf{W}f_i \,\|\, \mathbf{W}f_j]\right)$, where $\mathbf{W}$ represents a weight

matrix applied to the node features f_i and f_j, $\mathbf{a}^T$ is the transpose of the weight vector for the attention mechanism, and $\|$ indicates concatenation. (2) **Normalization**: The coefficients are normalized using the softmax function to facilitate comparison across nodes: $\boldsymbol{\alpha_{ij}} = \frac{\exp(a_{ij})}{\sum_{k\in N(i)}\exp(a_{ik})}$, $N(i)$ means the neighbors of node i.

Loss Design for Clustering. The design of the loss function in clustering with an attention mechanism aims to enhance intra-cluster similarity while reducing inter-cluster similarity. This is achieved by integrating a clustering-specific loss component and a regularization term for attention weights. Using the $ClusterGCN(G, K)$, we evaluate the clustering effectiveness with established metrics α, including Silhouette Score (SS), Davies-Bouldin Index (DBI), and Adjusted Rand Index (ARI). The loss function combines these elements to effectively capture and emphasize the structural patterns crucial for high-quality clustering: (1) **Clustering Loss**: Focused on clustering quality by maximizing intra-cluster similarity and minimizing inter-cluster differences: $L_{\text{cluster}} = \text{ClusteringQualityMetric}(C)$, drawing from metrics such as SS, DBI, and ARI in a gradient-optimizable form. (2) **Attention Regularization**: Aims to distribute attention discriminatively across the graph based on the entropy of attention weights: $L_{\text{reg}} = -\sum_{i=1}^{N}\sum_{j\in N(i)}\alpha_{ij}\log(\alpha_{ij})$. (3) **Combined Loss Function**: Integrates clustering and attention regularization: $L = L_{\text{cluster}} + \lambda L_{\text{reg}}$, where λ balances the two components. This loss design ensures that the Attention-Cluster-GCN model effectively learns to identify and emphasize the graph's structural patterns most relevant for high-quality clustering.

3.4 Feature Aggregation

After calculating attention coefficients, they weight the aggregation of neighboring node features, enhancing the distinction and capturing the nuanced structural information necessary for differentiating clusters: $f_i' = \sigma\left(\sum_{j\in N(i)}\alpha_{ij}\mathbf{W}f_j\right)$, The combination of large (F_l) and small (F_s) scale features forms a comprehensive spatial representation: $F = F_l \oplus F_s$, which is instrumental for clustering by integrating diverse scales of spatial information, facilitating a more accurate and meaningful distinguishing of the graph clustering purposes.

3.5 Dueling DDQN and Recommendation

From Clustering to Recommendations. The process of clustering and its role in generating recommendations for visitation locations are intrinsically connected. The clustering algorithm organizes and groups POIs based on their features which are refined through the attention mechanism described earlier. This clustering is not just a grouping by proximity or type, but a sophisticated categorization based on a composite of spatial and semantic similarities, ensuring that each cluster represents a coherent subset of attractions that are contextually related.

Dueling DDQN Architecture. Dueling DDQN (**Figure 1** and **Table 2**) offer a refined approach for addressing the complexities involved in recommendation systems, particularly when working with graph-based data structures while considering multi-purpose and multi-stakeholder. This section details the mathematical underpinnings of Dueling DDQN's integration into recommendation systems, highlighting its role in enhancing policy exploration, compensating for limitations in attention-based feature aggregation, and balancing key recommendation metrics.

The Dueling DDQN separates the estimation of the state-value function and the advantage for each action, which is particularly beneficial in the context of a graph where the state space and action space are large and complex:

$$Q(s,a;\theta,\alpha,\beta) = V(s;\theta_V) + \left(A(s,a;\theta_A) - \frac{1}{|\mathcal{A}(s)|}\sum_{a'\in\mathcal{A}(s)}A(s,a';\theta_A)\right),$$

where:

- $V(s;\theta_V)$ estimates the value of being in state s,
- $A(s,a;\theta_A)$ computes the advantage of action a over other actions in state s,
- θ_V and θ_A are parameters specific to the value and advantage streams, respectively,
- $\mathcal{A}(s)$ represents the action set available in state s.

Compensating Feature Aggregation with Differentiable Policy Exploration. Feature aggregation in graph-based data structures may miss capturing weaker yet critical signals due to the dominance of more salient features. Dueling DDQN mitigates this by employing a differentiable policy that encourages exploration of underappreciated features: $\pi(s) = \text{softmax}\left(\frac{Q(s,a;\theta,\alpha,\beta)}{\tau}\right)$, where the recommendation policy $\pi(s)$ as a *differentiable* function of the Q-values, allowing gradient-based optimization to directly tune which actions to explore; τ is a temperature parameter that controls the level of exploration, encouraging the system to investigate less obvious paths that might yield novel recommendations.

Reward Function for Precision, Serendipity, and Engagement. The reward function $R(s,a)$ is designed to incorporate different aspects of the recommendation quality, specifically aiming to balance precision, serendipity, and user engagement. The function is defined as: $R(s,a) = \lambda_p P(s,a) + \lambda_s S(s,a) + \lambda_u U(s,a)$, where $P(s,a)$, $S(s,a)$, and $U(s,a)$ measure the precision, serendipity, and user engagement in Hit Rate of action a in state s, respectively. λ_p, λ_s, λ_u are weighting factors that prioritize these aspects according to system goals.

Serendipity is defined as the measure of how surprising the recommendations are to the user, relative to the user's typical interaction or historical preferences, yet still being relevant and appreciated. Let $\mathcal{H}_u$ represent the historical set of items interacted with by user u, and let R represent the set of recommended items. Serendipity $S(s,a)$ can then be quantified as:

$$S(s,a) = \frac{|\{r \in R \mid r \notin \mathcal{H}_u \wedge r \text{ is relevant}\}|}{|R|},$$

where:

- $r \in R$ are the items recommended in the current session,
- $r \notin \mathcal{H}_u$ ensures that the recommended items are not part of the user's historical interactions,
- r is relevant confirms that the items are still relevant to the user's preferences.

Optimization from User Feedback for Improvement. The Dueling DDQN framework is optimized using a gradient-based approach, updating parameters to maximize expected rewards while

maintaining a balance among the competing objectives:

$$\theta^{new} = \theta^{old} + \eta \nabla_\theta \mathbb{E}[R(s,a)],$$

where η is the learning rate. This optimization ensures that the recommendation system dynamically adapts to user feedback and evolving data patterns, enhancing overall recommendation quality across precision, serendipity, and engagement. By incorporating Dueling DDQN into graph-based recommendation systems, the approach effectively addresses the challenges of large and complex action spaces typical of such environments. The use of a differentiable exploration strategy within this framework ensures that all potential actions are thoroughly evaluated, allowing the system to discover and leverage intricate patterns within the data for more effective and engaging recommendations This feedback mechanism ensures that the model continuously adapts to new data, improving accuracy and relevancy across different urban environments. This comprehensive and mathematically rigorous approach ensures the recommendation system is robust and adaptable, capable of handling the complexities of various urban settings internationally.

Implementation of Recommendations. After the successful aggregation of features and the formation of clusters, the system utilizes these clusters to inform its recommendations: $R = \{r_1, r_2, \ldots, r_n\}$ where each r_i is derived from cluster C_K. Each recommendation r_i is generated based on the characteristics of a cluster C_K. The selection of POIs from these clusters for recommendations is based on several criteria: **(1) User Preferences:** Recommendations are tailored to match the specific preferences and historical interactions of the user. If a user shows a preference for certain types of green spaces or cultural sites, the recommendation engine prioritizes POIs from clusters enriched with these features. **(2) Contextual Relevance:** The current context of the user, including spatial areas and current location within the city, affects which clusters are tapped for recommendations. **(3) Diversity and Novelty:** To enhance user experience, the system also considers diversity and novelty in its recommendations, ensuring that suggestions from clusters provide a range of experiences, even suggesting lesser-known POIs that still fit the user's overall preferences. The final recommendation result R for bus stops and their nearby green attractions are stored in a database through a structured and automated spatial graph data pipeline, enhancing the precision of our urban mobility and green attraction recommendations. Then, R are presented to the user through an interactive interface, which not only displays the suggested POIs but also provides detailed information about why these locations were recommended, based on the cluster attributes. This transparency helps in building trust and enhancing the user engagement with the system. Additionally, the interface allows users to provide immediate feedback on the recommendations, which the system uses to refine further clustering and recommendation processes, and leveraging insights derived from the spatial data analysis to inform decision-making or suggestions.

Personalization Mechanism. To ensure truly personalized recommendations, our system models each user with: (1) **User Embedding:** Each user's historical interactions (clicks or accepted recommendations) are aggregated into a fixed-dimensional vector, which serves as the initial state for the Dueling DDQN. The user embedding is updated incrementally after every interaction. (2) **Online Preference Update:** After each interaction (click, skip, or bookmark), we perform a small-batch incremental learning step to adjust the user embedding, allowing the policy to quickly adapt to changing behaviors. We employ a lightweight online fine-tuning (incremental gradient) mechanism to learn from user actions in real time. (3) **Personalization vs. Generalization:** To compare personalization against a purely generalized model based on global statistics, we introduce user-specific vectors. In our experiments, we evaluate *Personalized* versus *Global Dueling DDQN*, quantifying the difference between using user embeddings and relying solely on map-level statistics. See details in **Appendix B.1, B.2, B.3**, **Figure 5**,**Figure 6**.

Temporal Dynamics. Urban traffic exhibits strong collective patterns and clear periodicity—daily rush hours, weekday/weekend differences, even seasonal shifts. To capture this, we enrich our spatial graph with temporal embeddings (e.g. time-of-day, day-of-week, month-of-year) at each node. During offline clustering, we group POIs not only by location but also by similar traffic–time profiles, yielding spatiotemporal clusters that reflect, for example, morning-commuter green spaces vs. weekend-leisure parks.

Continuous Adaptation. To handle ever-changing traffic, we adopt a sliding-window update strategy: every 6 hours (configurable), we incrementally update node features and edge weights based on the latest transit data—avoiding full retraining. The attention mechanism is fine-tuned online via lightweight incremental gradient steps, and Cluster-GCN parameters are synchronized within minutes. The Dueling DDQN periodically retrains on recent interaction logs in past 24 hrs so that recommendation policies adapt to current user flow without catastrophic forgetting.

Generalizability. To validate the generalizability of the proposed recommendation system beyond initial test sites in Taiwan, a methodical approach is employed to generate representative and synthetic user samples from diverse international urban contexts, including Cheongju, Melaka, Chiang Mai, and Okinawa, where are also the cities without subway system (**Table 6**). This includes collecting and simulating data on urban structure, public transport systems, and population density to reflect the unique characteristics of each city. Adjustments are made to model parameters to fit local contexts, emphasizing cultural and environmental factors relevant to each location: **(1) Statistical Extrapolation and Parameter Adjustment:** Given a city C with urban parameters U_C, the baseline model M is adapted: $M_C = f(M, U_C)$, where f is the transformation function that adjusts M to fit U_C, optimizing for local transportation modalities and urban mobility patterns. **(2) Synthetic Data Generation:** Using M_C, synthetic user interactions I_C are generated to simulate decision processes within C: $s_i \sim p(U_C), \quad i = 1, 2, \ldots, N$, where $p(U_C)$ is the probability distribution of user choices derived from U_C, and N is the number of synthetic samples. See details in **Appendix B.4, B.5**, **Figure 6(d)**.

4 Experiments

4.1 Baseline and Performance Comparison

Clustering Metrics. To comprehensively assess the effectiveness of the clustering, we employ a suite of established metrics α, each highlighting different facets of clustering quality: (1) **Silhouette Score (SS)**: Measures the cohesion and separation of clusters. A higher SS indicates that clusters are dense and well-separated,

Table 3: Baseline and ours performance comparison in clustering and recommendation.

α	SS			DBI			ARI			Precision			Recall			F1-Score		
Batch size	16	32	64	16	32	64	16	32	64	16	32	64	16	32	64	16	32	64
GMM [21]	0.3968	0.3759	0.3771	0.3492	0.3511	0.3487	0.3256	0.3247	0.3269	0.43	0.42	0.43	0.40	0.39	0.40	0.41	0.40	0.41
FastGCN [4]	0.4612	0.4607	0.4628	0.2487	0.2479	0.2495	0.3917	0.3928	0.3905	0.50	0.49	0.50	0.48	0.47	0.48	0.49	0.48	0.49
GraphSAGE [12]	0.4665	0.4674	0.4659	0.2396	0.2381	0.2403	0.3961	0.3972	0.3954	0.52	0.51	0.52	0.50	0.49	0.50	0.51	0.50	0.51
GraphSAINT [28]	0.4713	0.4722	0.4704	0.2349	0.2357	0.2338	0.4015	0.4023	0.4006	0.53	0.52	0.53	0.51	0.50	0.51	0.52	0.51	0.52
VR-GCN [5]	0.4819	0.4828	0.4807	0.2598	0.2604	0.2586	0.3873	0.3864	0.3885	0.55	0.54	0.55	0.53	0.52	0.53	0.54	0.53	0.54
GATT [23]	0.4417	0.4403	0.4425	0.2649	0.2655	0.2637	0.3672	0.3661	0.3683	0.49	0.48	0.49	0.47	0.46	0.47	0.48	0.47	0.48
GaAN [29]	0.4868	0.4893	0.4869	0.2543	0.2559	0.2531	0.3725	0.3737	0.3714	0.56	0.55	0.56	0.54	0.53	0.54	0.55	0.54	0.55
Cluster-GCN [7]	0.4968	0.4954	0.4876	0.2231	0.2285	0.2740	0.4034	0.4053	0.3845	0.57	0.56	0.57	0.55	0.54	0.55	0.56	0.55	0.56
DDSGNN [27]	0.4972	0.4903	0.4903	0.2195	0.2277	0.2484	0.4177	0.3983	0.4215	0.58	0.57	0.58	0.58	0.55	0.55	0.57	0.56	0.57
TSP-GNN [10]	0.4986	0.4975	0.4914	0.2178	0.2293	0.2577	0.4083	0.4039	0.4092	0.59	0.58	0.59	0.57	0.56	0.57	0.58	0.57	0.58
GCRL [17]	0.5132	**0.5013**	0.4972	0.2084	0.2138	**0.2120**	0.4014	0.4075	0.4147	0.69	0.64	0.65	0.62	0.58	0.56	0.66	**0.69**	0.60
Ours	**0.5267**	0.4971	**0.5113**	**0.1992**	**0.2063**	0.2304	**0.4281**	**0.4269**	**0.4274**	**0.75**	**0.67**	**0.68**	**0.65**	**0.60**	**0.61**	**0.70**	0.63	**0.66**

which is desirable for high-quality clustering: $SS = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}}$, where $a(i)$ is the average distance from the i-th data point to other points in the same cluster, and $b(i)$ is the smallest average distance from the i-th data point to points in a different cluster, minimized over clusters. (2) **Davies-Bouldin Index (DBI)**: Compares the within-cluster similarity to the between-cluster differences. Lower values indicate better clustering: $DBI = \frac{1}{K}\sum_{i=1}^{K} \max_{j\neq i}\left(\frac{\sigma_i+\sigma_j}{d(c_i,c_j)}\right)$, where σ_i is the average distance of all points in cluster i to the centroid c_i, and $d(c_i, c_j)$ is the distance between centroids c_i and c_j. (3) **Adjusted Rand Index (ARI)**: Measures the similarity between two clusterings, accounting for chance groupings. ARI values near 1 indicate perfect agreement: $ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]/\binom{N}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2}+\sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]/\binom{N}{2}}$, where n_{ij} is the number of points in common between cluster i in the first clustering and cluster j in the second clustering, a_i is the total points in cluster i of the first clustering, and b_j is the total points in cluster j of the second clustering.

Baselines and Ours Performance . Our model excels in SS, DBI, ARI in clustering and Precision, Recall, F1-Score in recommendation for batch sizes of 16, 32, and 64, surpassing most baselines across all metrics (**Table 3**). For benchmarking, state-of-the-art methods in baseline models of GCN techniques like FastGCN [4], GraphSAGE [12], and GraphSAINT [28] demonstrate superior performance with higher SS and ARI, and lower DBI, while these baselines designed for scalable and efficient graph clustering, provide high-quality results with optimized clustering performance. Particularly, Cluster-GCN [7] and enhances efficiency and scalability, while GCRL [17] achieves higher SS up to 0.5013 for batch size of 32, and higher DBI in 0.2120 for 64 batch size. Our model modifies Cluster-GCN by integrating attention mechanisms with Dueling DDQN, enabling more precise structure capture and pattern detection, leading to the highest SS of 0.5267, the lowest DBI of 0.1992, and the highest ARI of 0.4281 at batch size 16, thus our model with the optimal cluster separation and accuracy in robustness; Our model consistently outperforms all baselines in recommendation effectiveness—achieving a peak Precision of 0.75, a Recall of 0.65, and an F1-Score of 0.70 at batch size 16—demonstrating substantially improved accuracy and balance between precision and coverage (**Table 3**).These improvements position our model as the optimal method for analyzing complex graph data, effectively utilizing underlying patterns, and outperforming the baselines.

Table 4: Clustering metrics with diverse sampling methods.

Sampling Method (with batch size = 16, Tainan)	SS	DBI	ARI
Ours (Multi-Scale Sampling)	**0.5267**	**0.1992**	**0.4269**
Sampling by Ring Network	0.4201	0.9526	0.3866
Sampling by Village Areas	0.3621	1.2126	0.3761
Sampling by Axial Lines	0.3921	1.1126	0.3961
Random Sampling	0.3147	1.6929	0.3346

Table 5: Clustering performance of five ablation study.

α (with batch size = 16)	SS	DBI	ARI
Ours	**0.5267**	**0.1992**	**0.4269**
Ours w/o Dueling DDQN	0.3310	2.2618	0.3116
Ours w/o ATT	0.2421	0.4826	0.3636
Ours w/o Multi-Scale (F)	0.2009	3.1126	0.3661
Ours w/o Large Scale (F_l)	0.2147	2.6929	0.3846
Ours w/o Small Scale (F_s)	0.2214	3.1135	0.3812

Sampling for Clustering. In **Table 4**, we evaluate the impact of using different sampling methods on our model's performance. The experiments show the importance of our chosen approach, evidenced by significant differences in SS, DBI and ARI with other methods. Each method influences the model differently: (1) ***Ring Network***: This method defines the inner ring as training data, the middle ring as validation, and the outer ring as testing, resulting in a lower SS = 0.4201, higher DBI = 0.9526, and a reduced ARI = 0.3866. (2) ***Village Areas***: This method focuses on sampling from distinct village areas to represent rural settings in training, validation, and testing. It results in SS = 0.3621, DBI = 1.2126, and ARI = 0.3761. This method captures the heterogeneity within rural settings but still shows a decrease in clustering performance compared to our approach. (3) ***Axial Lines***: This method, which involves splitting data along axial lines (6:2:2 ratio for training, validation, and test splitting), means that each axis will include city, suburb, and rural areas. The results in a performance decrease: SS drops to 0.3921, DBI rises to 1.1126, and ARI to 0.3961, indicating its less severe but still impactful differences. (4)) ***Random Sampling***: Randomly assigning data to Train, Val, and Test sets leads to the notable performance degradation: SS decreases to 0.3147, DBI increases to 1.6929, and ARI to 0.3346, showing that our method outperforms random sampling by capturing more relevant spatial features.

Table 6: Comparison across 10 cities without subway system (spatial heterogeneity in city functionalities and cross-countries)

City	Cluster Metrics			Recommendation Effectiveness			User Indexes of Recommendations			
	SS	DBI	ARI	Precision	Recall	F1-Score	Sample Size	Hit Rate (%)	Satisfaction (%)	Serendipity (%)
Tainan, Taiwan (a historical city)	**0.5267**	**0.1992**	**0.4281**	**0.75**	**0.65**	**0.70**	100	**82**	**80**	**48**
Hsinchu, Taiwan (a high-tech city)	0.5241	0.2054	0.4197	0.72	0.68	0.67	100	78	75	46
Keelung, Taiwan (a harbor city)	0.5203	0.2031	0.4233	0.70	0.62	0.66	100	80	77	38
Cheongju, South Korea	0.5150	0.2132	0.4177	0.68	0.57	0.62	100	74	72	28
Melaka, Malaysia	0.5113	0.2157	0.4043	0.65	0.58	0.61	100	75	73	45
Chiang Mai, Thailand	0.5182	0.2085	0.4154	0.69	0.60	0.64	100	76	74	42
Okinawa, Japan	0.5123	0.2127	0.4055	0.67	0.59	0.63	100	77	76	36
New York, USA	0.5050	0.2201	0.4102	0.66	0.60	0.63	100	70	68	30
London, UK	0.5082	0.2175	0.4125	0.67	0.61	0.64	100	72	70	32
Tokyo, Japan	0.5105	0.2123	0.4148	0.68	0.62	0.65	100	74	73	35

Validated across Various Cities. Due to our method outperforming previous models, this table furthur indicates how our method performs in different urban contexts, providing a more comprehensive assessment of its efficacy and applicability. For cities with more primitive forms of transportation, the proposed method is extended here to include results from different city functionalities and across countries to better verify its generalizability and performance across different urban settings; for more advanced cities (NYC, London, and Tokyo), which are larger, have more complex forms and structures of transportation, large populations, and need to consider the relationship with the surrounding urban network, the model still maintain the same level performance. **Table 6** shows the clustering metrics across these cities with recommendation effectiveness (Precision, Recall, and F1-Score), and user indexes of recommendations (Sample Size, Hit Rate (%), and Satisfaction (%)).

4.2 Ablation Study

In **Table 5**, we evaluate the impact of omitting specific features on our model's clustering performance. The results demonstrate the importance of each component. The full model achieves the highest SS = 0.5267, lowest DBI = 0.1992, and highest ARI = 0.4269. Removing the Dueling DDQN leads to a significant performance drop (SS = 0.3310, DBI = 2.2618, ARI = 0.3116), highlighting its crucial role in balancing policy exploration and feature aggregation. Excluding the attention mechanism (w/o ATT) drastically decreases effectiveness, as seen with SS = 0.2421, the second high DBI = 0.4826, and ARI = 0.3636, emphasizing its role in refining spatial-semantic relationships. The absence of multi-scale features (w/o F) further degrades performance (SS = 0.2009, DBI = 3.1126, ARI = 0.3661), showing their importance in capturing spatial heterogeneity. Comparatively, removing small-scale features (w/o F_s) has a slightly more negative impact (SS = 0.2214, DBI = 3.1135, ARI = 0.3812) than excluding large-scale features (w/o F_l, SS = 0.2147, DBI = 2.6929, ARI = 0.3846). This indicates the small-scale features' critical role in capturing fine-grained spatial details, while large-scale features contribute to broader contextual understanding. These results validate the necessity of each component in achieving clustering performance.

4.3 Sensitivity Analysis

The sensitivity analysis reveals notable differences in the performance metrics across scales and parameter settings, as depicted in **Figure 2**. (1) For the **Learning Rate (η)**, the results indicate that the *Small Scale (100m)* achieves optimal performance at 10^{-6}, while the *Large Scale (500m)* exhibits lower sensitivity, with more stable

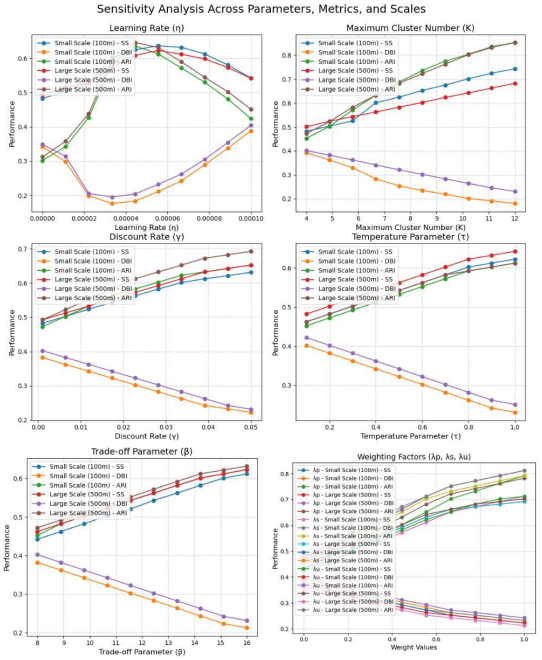

Figure 2: Sensitivity analysis: parameters, metrics, and scales.

trends across the parameter range. (2) For the **Maximum Cluster Number (K)**, the *Small Scale (100m)* demonstrates heightened sensitivity, reaching peak performance at $K = 12$. Conversely, the *Large Scale (500m)* displays relatively stable performance throughout the range. (3) The **Trade-off Parameter (β)** shows a consistent improvement in performance for the *Large Scale (500m)* as β increases, peaking in the range of 12 to 14. (4) For the **Discount Rate (γ)**, the *Small Scale (100m)* experiences a pronounced improvement, peaking at approximately 1.3×10^{-2}. The *Large Scale (500m)* also improves with γ, but the trend is more gradual and exhibits less variability. These results highlight the importance of scale-specific tuning for achieving optimal performance. The *Small Scale (100m)* consistently demonstrates greater sensitivity across all parameters, emphasizing the need for fine-grained adjustments in high-resolution scenarios. In contrast, the *Large Scale (500m)* benefits from broader parameter ranges and exhibits more robust performance.

Figure 3: The app interface with real-world deployment.

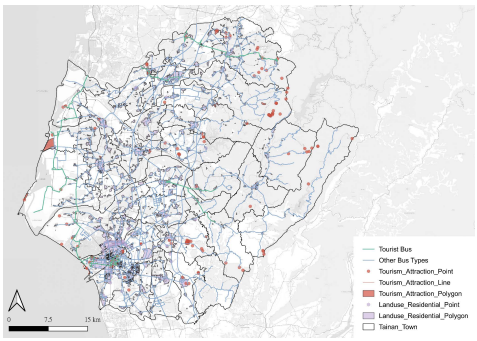

Figure 4: One of the visualized results illustrates the multi-purpose and multi-stakeholder characteristics of tourist and resident activity areas alongside bus routes, highlighting clustering results optimized through attention-based RL.

4.4 Multi-Purpose/Stakeholder: User Groups

Our method effectively captures the multi-scale spatial heterogeneity inherent in bus network, providing a deeper understanding of how factors such as the geographic separation of residential and tourist areas impact transit demand and usage patterns. The integration of Dueling DDQN with graph clustering enables differentiated and customized recommendations tailored to the needs of both residents and tourists. For instance, tourists benefit from recommendations prioritizing access to green attractions and efficient sightseeing routes, while residents gain enhanced transit options for daily commuting, focusing on accessibility and reduced congestion. Additionally, attention-based RL refines clustering performance by improving the separation and identification of routes based on passenger behavior, ensuring that routes are optimized for both efficiency and specific user groups with the multi-purpose and multi-stakeholder characteristics (**Figure 4**). These advancements have practical implications for urban planning, as they address disparities in bus network usage and support the goals for sustainable urban transit by aligning public transportation strategies with environmental policies, thereby promoting green initiatives without compromising user satisfaction. See details in **Appendix B.5, B.6**, **Figure 5(e)**,**Figure 6(c)**.

5 Interfaces and Deployment

Our system integrates GIS-based Attention-Cluster-GCN with Dueling Double Deep Q Network (Dueling DDQN), a form of RL, enhancing the web-based app deployment of urban travel recommendations across various cities. This integration allows the system to dynamically adapt and optimize recommendation strategies based on user interactions and environmental changes, ensuring high-quality, context-aware suggestions for urban mobility and green travel. The web-based app interface demonstrates the following functionalities (**Figure 3**): (1) Clustering recommendations for green attractions based on mapping to a single bus stop; (2) Plant POI recommendations, reflecting urban spatial heterogeneity, enhanced gamified experience with AR-based realistic virtual plants; (3) Integration of green attraction recommendations with real-time transportation information for seamless travel planning; (4) A localized recommendation system tailored to user preferences for nearby green attractions. (5) The user scenarios in real-world deployment at the bus station: Applying this framework integrates Attention-Cluster-GCN with Dueling DDQN to dynamically adjust attention weights based on user feedback and environmental data. The system prioritizes key urban features and POIs, adapts to user preferences and urban changes, and provides optimized route and POI recommendations. Leveraging RL, it updates suggestions in real time considering user feedback related to traffic, public transport, and user location, addressing urban spatial heterogeneity effectively. By combining dynamic clustering and personalized recommendations, the app promotes green travel, efficient urban planning, and an engaging user experience.

Real-Time Responsiveness & Computational Load. Given the latency-sensitive nature of transport suggestions, our system splits processing into two stages. In the **(1) offline preprocessing** stage (batch updates, $\leq$1 hour), we build and update spatiotemporal clusters using multi-scale and temporal embeddings, precompute attention scores into lookup tables, and train or fine-tune the Dueling DDQN model on accumulated interaction data. For **(2) online processing** (end-to-end <200 ms): encoding the user's current location and time into a state vector ($\approx$ 5 ms), retrieving precomputed cluster and attention weights ($\approx$ 10 ms), and performing a Dueling DDQN forward pass to generate top-K recommendations ($\approx$ 50 ms on a mid-range GPU/CPU). Benchmarking on a standard edge server shows an **average end-to-end latency** of 150 ms, **peak memory usage** of 600 MB, and **throughput** of 120 queries per second. See details in **Appendix B.6, B.7** and **Limitations** in **Appendix C**.

6 Conclusion

Our approach significantly enhances urban travel efficiency by leveraging advanced graph clustering with attention and Q-Learning in data pipeline for green attraction recommendations in public transportation. Achieving results that outperform other baselines, this model excels in precision, scalability, and adaptability. By incorporating an attention mechanism into the Cluster-GCN with Dueling DDQN policy, it effectively identifies and emphasizes structural patterns, enhancing graph clustering performance across multiple scales of spatial heterogeneity and driving improvements in urban sustainability and information management.

Acknowledgments

This work was partially supported by National Science and Technology Council (NSTC) of Taiwan under Grants 114-2628-E-006 -005 -MY3 and 112-2221-E-006 -150 -MY3.

References

[1] Tainan City Government Agriculture Bureau. 2024. *Distribution and production period of agricultural products in Taiwan.* Retrieved April 01, 2024 from https://agron.tainan.gov.tw/cp.aspx?n=34637

[2] Tainan City Government Agriculture Bureau. 2024. *Tainan Old Trees Maps and Database.* Retrieved April 01, 2024 from https://oldtree.tainan.gov.tw/#/

[3] Taiwan Bureau of Transportation, Tainan City Government. 2024. *Tainan Bus Stop Information.* Retrieved April 01, 2024 from https://data.tainan.gov.tw/dataset/tainan-bus-stop

[4] Jie Chen, Tengfei Ma, and Cao Xiao. 2018. FastGCN: Fast Learning with Graph Convolutional Networks via Importance Sampling. *ICLR.*

[5] Jianfei Chen, Jun Zhu, and Le Song. 2017. Stochastic Training of Graph Convolutional Networks with Variance Reduction. In *International Conference on Machine Learning.* https://api.semanticscholar.org/CorpusID:3636539

[6] Jing Cheng, Ruiyun Yan, and Yueer Gao. 2020. Exploring spatial heterogeneity in accessibility and transit mode choice. *Transportation Research Part D-transport and Environment* 87 (2020), 102521. https://api.semanticscholar.org/CorpusID:224847564

[7] Wei-Lin Chiang, Xuanqing Liu, Si Si, Yang Li, Samy Bengio, and Cho-Jui Hsieh. 2019. Cluster-GCN: An Efficient Algorithm for Training Deep and Large Graph Convolutional Networks. 257–266. doi:10.1145/3292500.3330925

[8] David Davies and Don Bouldin. 1979. A Cluster Separation Measure. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* PAMI-1 (05 1979), 224 – 227. doi:10.1109/TPAMI.1979.4766909

[9] Fanghao Fu, Jiemin Xie, Ming Cai, and Wei Huang. 2024. A Bus-Service-Based Zone Division Approach for the Spatial Analysis of Public Bus: A Case Study in South China. *Journal of Advanced Transportation* 2024 (02 2024), 1–19. doi:10.1155/2024/8681100

[10] Qixu Gong, Huiying Chen, Huiping Cao, and Jiefei Liu. 2024. Backbone Index and GNN Models for Skyline Path Query Evaluation over Multi-cost Road Networks. *ACM Trans. Spatial Algorithms Syst.* (April 2024). doi:10.1145/3660632 Just Accepted.

[11] Rongge Guo, Wenyi Zhang, Wei Guan, and Bin Ran. 2021. Time-Dependent Urban Customized Bus Routing With Path Flexibility. *IEEE Transactions on Intelligent Transportation Systems* 22, 4 (2021), 2381–2390. doi:10.1109/TITS.2020.3019373

[12] William Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. (06 2017).

[13] Hado van Hasselt, Arthur Guez, and David Silver. 2016. Deep reinforcement learning with double Q-Learning. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (Phoenix, Arizona) *(AAAI'16)*. AAAI Press, 2094–2100.

[14] Lawrence J. Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification* 2 (1985), 193–218. https://api.semanticscholar.org/CorpusID:189915041

[15] Pawinee Iamtrakul and Sararad Chayphong. 2024. Exploring Spatial Accessibility to Urban Activities Based on the Transit-Oriented Development Concept in Pathum Thani, Thailand. *Sustainability* 16 (03 2024). doi:10.3390/su16052195

[16] George Karypis and Vipin Kumar. 1999. Kumar, V.: A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs. SIAM Journal on Scientific Computing 20(1), 359-392. *Siam Journal on Scientific Computing* 20 (01 1999). doi:10.1137/S1064827595287997

[17] Qi Liu, Zirui Li, Xueyuan Li, Jingda Wu, and Shihua Yuan. 2022. Graph convolution-based deep reinforcement learning for multi-agent decision-making in mixed traffic environments. *arXiv preprint arXiv:2201.12776* (2022).

[18] Taiwan Ministry of Agriculture. 2024. *Taiwan Agricultural and Fishery Production Map.* Retrieved April 01, 2024 from https://kmweb.moa.gov.tw/theme_list.php?theme=production_map

[19] Department of Economic and United Nations Social Affairs. 2024. *THE 17 GOALS – Sustainable Development.* Retrieved April 01, 2024 from https://sdgs.un.org/goals

[20] Dmitry Pavlyuk. 2015. *Spatial Allocation of Bus Stops: Advanced Econometric Modelling.* 331–339. doi:10.1007/978-3-319-19216-1_31

[21] Douglas Reynolds. 2008. Gaussian Mixture Models. *Encyclopedia of Biometrics* (01 2008). doi:10.1007/978-0-387-73003-5_196

[22] Peter J. Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* 20 (1987), 53–65. doi:10.1016/0377-0427(87)90125-7

[23] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio', and Yoshua Bengio. 2017. Graph Attention Networks. *ArXiv* abs/1710.10903 (2017). https://api.semanticscholar.org/CorpusID:3292002

[24] Jiangbo Wang, Toshiyuki Yamamoto, and Kai Liu. 2021. Spatial dependence and spillover effects in customized bus demand: Empirical evidence using spatial dynamic panel models. *Transport Policy* 105 (2021), 166–180. doi:10.1016/j.tranpol.2021.03.004

[25] Yujing Wang, Yi Deng, Fu Ren, Ruoxin Zhu, Pei Wang, Tian Du, and Qingyun Du. 2019. Analysing the spatial configuration of urban bus networks based on the geospatial network analysis method. *Cities* 96 (08 2019). doi:10.1016/j.cities.2019.102406

[26] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Van Hasselt, Marc Lanctot, and Nando De Freitas. 2016. Dueling network architectures for deep reinforcement learning. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48* (New York, NY, USA) *(ICML'16)*. JMLR.org, 1995–2003.

[27] Hanqing Zeng, Muhan Zhang, Yinglong Xia, Ajitesh Srivastava, Andrey Malevich, Rajgopal Kannan, Viktor Prasanna, Long Jin, and Ren Chen. 2024. Decoupling the depth and scope of graph neural networks. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NIPS '21)*. Curran Associates Inc., Red Hook, NY, USA, Article 1504, 15 pages.

[28] Hanqing Zeng, Hongkuan Zhou, Ajitesh Srivastava, Rajgopal Kannan, and Viktor Prasanna. 2020. GraphSAINT: Graph sampling based inductive learning method.

[29] Jiani Zhang, Xingjian Shi, Junyuan Xie, Hao Ma, Irwin King, and D. Y. Yeung. 2018. GaAN: Gated Attention Networks for Learning on Large and Spatiotemporal Graphs. In *Conference on Uncertainty in Artificial Intelligence.* https://api.semanticscholar.org/CorpusID:3973810

Appendix

A Training Details

Numerical experiments validate the proposed models. The model runs on a desktop with a 16-core Intel® Core™ i7-13700KF CPU (5.40-GHz), NVIDIA® GeForce RTX™ 3070 Ti GPU (8 GB), and 32 GB RAM via implementation in the PyTorch library. Training on the GPU lasts about 16 hours for 100 epochs, with a mini-batch size of 16. For benchmarking, we replicated and reproduced outcomes from state-of-the-art baselines. Our model with multi-scale clustering and an attention network underwent 100 and 300 epochs of training, respectively. The learning rate l_r starts at 10^{-7}. The maximum cluster number K is 8, with a trade-off parameter β at 12 and a discount rate γ set to 1.3×10^{-2}. The low computational intensity in training time and hardware enhacing practical scalability.

B Detailed Experimental Analysis

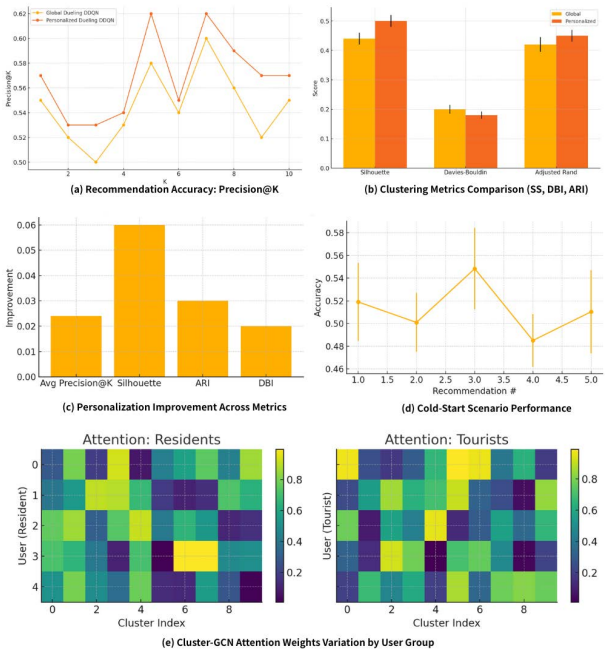

Figure 5: (a) Recommendation accuracy: Precision@K for global vs. personalized Dueling DDQN. (b) Comparison of clustering quality metrics between global and personalized models. (c) Absolute improvements in Precision@K and clustering metrics due to personalization. (d) Cold-start scenario performance over the first five recommendations. (e) Cluster-GCN attention weights variation by user groups.

B.1 Recommendation Accuracy: Precision@K. In **Figure 5(a)**, Precision@K for Global vs. Personalized Dueling DDQN shows the latter consistently higher. Both models peak at $K = 5, 7$ (Global $\approx 0.58, 0.60$, Personalized ≈ 0.62), and the gap grows from ≈ 0.02 at $K = 1$ to ≈ 0.05 at $K = 9$. A minimum occurs at $K = 3$ (Global ≈ 0.50, Personalized ≈ 0.53), underscoring the benefit of user embeddings for adaptive ranking.

B.2 Clustering Metrics Comparison. In **Figure 5 (b)**, we compare three clustering quality metrics—SS, DBI, and ARI—between the global and personalized clustering pipelines. The personalized approach achieves a higher average in 10 times for all cities than the global approach in SS (0.50 vs. 0.44) and ARI (0.45 vs. 0.42), while yielding a lower (better) DBI (0.18 vs. 0.20). Error bars indicate one standard deviation over repeated runs. These results confirm that attention-driven, personalized embeddings produce tighter, more coherent clusters of POIs.

B.3 Personalization Improvement. In **Figure 5 (c)**, it summarizes the absolute improvements gained by **personalization** over the global baseline. On average, Precision@K increases by 0.024, SS improves by 0.06, DBI by 0.02, and ARI by 0.03. The largest relative gain appears in clustering coherence (SS), highlighting the critical role of user-specific information in grouping relevant POIs.

B.4 Cold-Start Scenario Performance. Figure 5 (d) shows the recommendation accuracy for the first five suggestions made to new users, plotted with mean and standard deviation. Initial accuracy starts at 0.52 and dips to 0.49 for the forth recommendation, before recovering slightly. The error bars (±0.03–0.04) reflect variability due to limited interaction data. These results suggest that while cold-start performance is inherently lower, our model still achieves a near 0.5 accuracy within five interactions, demonstrating reasonable effectiveness even with minimal user history.

B.5 Attention Weights Variation by User Group. In **Figure 5(e)**, attention heatmaps (rows: users; columns: cluster indices) are shown for two groups. **Residents** concentrate on commute-related clusters (yellow) with moderate variance. **Tourists** focus on attraction-rich clusters, displaying higher, more dispersed weights. This clear separation demonstrates that the attention mechanism captures group-specific preferences for tailored recommendations.

B.6 M3-Specific Evaluation Metrics. To verify that M3 effectively addresses **Multi-Scale**, **Multi-Purpose**, and **Multi-Stakeholder**, we define corresponding metric groups:

- **Multi-Scale Consistency (MSC):** Measures clustering quality at each geographic scale $s \in \{s, l\}$. $\text{MSC}_s = \frac{1}{|C_s|} \sum_{c \in C_s} \text{SS}(c)$, $\Delta_{\text{MSC}} = |\text{MSC}_s - \text{MSC}_l|$. A low Δ_{MSC} indicates balanced performance across scales.
- **Multi-Purpose Effectiveness (MPE):** For each user intent $p \in \{\text{relax}, \text{exercise}, \text{learn}\}$, compute Precision@K and Recall@K: $\text{MPE}_p = \left(\text{Precision@K}_p,\ \text{Recall@K}_p\right)$. Variance across intents σ_{MPE} should be small to demonstrate stable performance.
- **Stakeholder Fairness Index (SFI):** Let R_{tourist} and R_{resident} be average F1-scores for two groups. Define $\text{SFI} = 1 - \frac{|R_{\text{tourist}} - R_{\text{resident}}|}{\max(R_\cdot)}$. Values close to 1 indicate equitable recommendations.

Figure 6(a) plots Δ_{MSC} for ten cities. Okinawa has the smallest $\Delta_{\text{MSC}} \approx 0.001$, Keelung the largest ($\approx 0.020$), with Chiang Mai ($\approx 0.018$) and Melaka ($\approx 0.014$) also high. Others like Tainan (≈ 0.005), Hsinchu (≈ 0.010), Cheongju (≈ 0.0045), New York (≈ 0.012), London (≈ 0.007), Tokyo (≈ 0.0135) are mid-range. This confirms M3's robust clustering across scales, despite a few extreme cases. **Figure 6 (b)** presents the Stakeholder Fairness Index (SFI) for each model, plotted on a log scale to highlight fine-grained

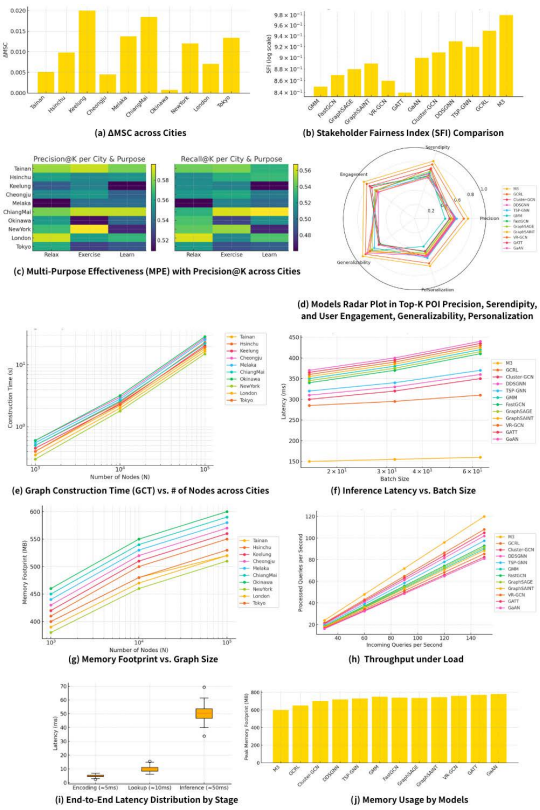

Figure 6: (a) Δ_{MSC} across cities. (b) Stakeholder Fairness Index (SFI) comparison across models. (c) Multi-Purpose Effectiveness (MPE) with Precision@K across Cities. (d) Models radar plot in Top-K POI Precision, Serendipity, and User Engagement, Generalizability, Personalization. (e) Graph Construction Time (GCT) vs. number of nodes across cities. (f) Inference latency vs. batch size. (g) Memory footprint vs. graph size. (h) Throughput under load. (i) End-to-end latency distribution by stage. (j) Memory usage by models.

differences. Our proposed M3 achieves the highest fairness (SFI ≈ 0.98), followed by GCRL (≈ 0.95), DDSGNN (≈ 0.93), and TSP-GNN (≈ 0.92). Classic GCN variants (e.g. Cluster-GCN, GraphSAINT) and attention-based models (e.g. GaAN) exhibit intermediate SFI values (0.89–0.91), whereas simpler baselines (GMM, GATT) fall below 0.87. These results confirm that M3 delivers the most equitable recommendations between tourists and residents, with GCRL as the closest competitor. **Figure 6(c)** displays Precision@K and Recall@K for Relax, Exercise, and Learn across ten cities. Exercise peaks at Precision≈ 0.59 (NYC) and Recall≈ 0.57 (Chiang Mai); Relax records Precision≈ 0.52–0.58, Recall≈ 0.48–0.56; Learn shows Precision≈ 0.51–0.58, Recall≈ 0.47–0.54. This confirms M3's stable, high-quality recommendations across purposes and locations. **Figure 6(d)** contrasts five diversity metrics—Precision, Serendipity, Engagement, Generalizability, Personalization—between M3 and eleven baselines. M3 outperforms all, especially in Engagement and Generalizability, with GCRL second, confirming M3 not only achieves superior accuracy but also delivers the most novel, engaging, adaptable, and personalized recommendations.

B.7 Scalability and Robustness Evaluation. While M3 performs well on moderate city graphs, we must demonstrate scalability as data volumes grow. **(1) Graph Construction Time:** Time to build spatiotemporal graph as a function of number of nodes N and edges E. **(2) Inference Latency** End-to-end online recommendation time vs. graph size and batch size, measured in milliseconds. **(3) Memory Footprint:** Peak RAM/VRAM usage during clustering and Dueling DDQN inference, reported for increasing N. **(4) Throughput:** Queries per second supported under load.

Figure 6(e) plots graph construction time vs. number of nodes $N \in \{10^3, 10^4, 10^5\}$ on a log–log scale. Cheongju peaks at ≈ 30s, NYC at ≈ 15s; all cities span 0.3–0.6s for 10^3 and 15–30s for 10^5. **Figure 6(f)** shows inference latency for $B \in \{16, 32, 64\}$. M3 consistently achieves the lowest latency (≈150 ms at $B = 16$, rising only to ≈ 160 ms at $B = 64$), demonstrating its efficiency and scalability. GCRL is the second fastest: 285 → 310ms; while other methods: 300–450ms range and exhibit a roughly linear increase with batch size. **Figure 6(g)** reports memory footprint vs. N. Usage grows from 380–470MB at 10^3 to 510–600MB at 10^5; Okinawa/Chiang Mai ≈ 600MB, NY/London ≈ 510MB. This confirms that M3's memory requirements scale predictably across diverse urban contexts. **Figure 6 (h)** plots the processed queries per second against incoming load (30–150 qps) for M3 and eleven baselines. M3 (gold) exhibits the highest throughput, scaling linearly from ≈24 qps processed at 30 incoming to ≈120 qps at 150 incoming. GCRL (orange) follows closely, processing ≈22–110 qps over the same range. Other models remain below 100 qps at peak load, confirming that M3 delivers the best end-to-end serving capacity under heavy query traffic. **Figure 6 (i)** presents the latency distribution for the three stages of our online pipeline—encoding, lookup, and inference—using boxplots over 100 runs. Encoding (≈5 ms) exhibits minimal variability (IQR ≈3–7 ms), lookup (≈10 ms) shows moderate spread (IQR ≈8–12 ms), and inference (≈50 ms) has the highest median (≈50 ms) and variability (IQR ≈45–55 ms, with occasional outliers up to ≈70 ms). All stages remain well under the 200 ms end-to-end target, confirming the system's suitability for real-time recommendations. **Figure 6 (j)** compares the peak memory footprint of twelve models. Our M3 has the lowest requirement at approximately 600 MB, with GCRL a close second at around 650 MB. Other GNN and clustering baselines consume between 700 MB and 800 MB, peaking with GaAN at about 790 MB. These results confirm that M3 offers the most memory-efficient deployment among all evaluated methods.

C Limitations

Despite M3's strong empirical performance, it has several limitations: (1) potential oversampling bias in POI balancing; (2) dependence on bus-stop mapping, limiting applicability in cities without comprehensive bus networks; (3) grid-cell collisions when multiple POIs occupy the same cell; (4) sharp cold-start performance decline (accuracy drops to 0.49 by the 4th suggestion)—indicating that early-stage user preference modeling requires further enhancement; and (5) the global setting of reward-weight hyperparameters λ_p, λ_s, and λ_u does not account for differing priorities among stakeholders (e.g., policymakers versus end-users).