

©Copyright 2026

Bin Han

In Pursuit of High-Quality Urban Data

Bin Han

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2026

Reading Committee:

Bill Howe, Chair

Aylin Caliskan

Jon E. Froehlich

Program Authorized to Offer Degree:
Information Science

University of Washington

Abstract

In Pursuit of High-Quality Urban Data

Bin Han

Chair of the Supervisory Committee:
Bill Howe
Information School

Urban data are central to transportation planning, mobility modeling, accessibility analysis, infrastructure assessment, and many other forms of urban decision-making. However, the usefulness of urban data depends not only on their availability, but also on their quality. In practice, urban datasets are often too coarse to support fine-scale analysis, incomplete with respect to important infrastructure, inconsistent across sources, or insufficiently faithful to the real-world systems they are meant to represent. These limitations constrain what urban phenomena can be observed, modeled, integrated, and acted upon. This dissertation argues that high-quality urban data are a prerequisite for reliable and equitable urban analytics, and that improving urban data quality requires targeted methods designed around the specific deficiencies, structures, and intended uses of the data.

This dissertation studies urban data quality through four complementary dimensions: granularity, coverage, coherence, and fidelity. Each dimension corresponds to a different kind of data deficiency and motivates a different methodological study. Granularity concerns whether urban phenomena are represented at sufficient spatial or temporal resolution. Coverage concerns whether important urban features are represented in the data at all. Coherence concerns whether heterogeneous data sources can be reconciled into consistent and unified representations. Fidelity concerns whether generated or extracted data preserve the real-world structure and function needed for downstream applications.

To refine granularity, this dissertation introduces the Structurally-Aware Recurrent Net-

work (SARN), a model that recovers fine-resolution urban signals from coarse, privacy-preserving aggregates by jointly modeling temporal dynamics, global spatial interactions, and hierarchical containment relationships. To increase coverage, it develops a zero-shot annotation pipeline that combines class-agnostic segmentation, candidate filtering, and Set-of-Mark prompting to identify underrepresented built-environment features from aerial imagery using vision-language models. To improve coherence, it investigates large-language-model-assisted spatial data integration for spatial join and spatial union tasks, showing that LLMs can support integration decisions more effectively when paired with human-interpretable geometric features. To enhance fidelity, it introduces TraversRL, a vision-conditioned iterative generator for pedestrian pathway graphs that is improved through reinforcement learning objectives aligned with graph-level geometry and traversability.

Overall, these studies show that urban data quality is multidimensional and that different deficiencies require different computational strategies. More broadly, the dissertation demonstrates that urban data should not be treated only as fixed input to downstream models. Urban data themselves can be the object of methodological innovation. By improving how urban data are refined, expanded, reconciled, and structured, this dissertation advances a broader agenda for building high-quality urban data in support of more reliable and actionable urban analytics.

TABLE OF CONTENTS

	Page
List of Figures	iii
List of Tables	vii
Chapter 1: Introduction	1
1.1 Research Objectives and Study-Specific Contributions	4
1.2 Dissertation-Level Contributions	7
1.3 Dissertation Organization	8
Chapter 2: Background	9
2.1 Urban Applications & Learning-Based Techniques	10
2.2 Large Foundation Models	17
Chapter 3: Refining Urban Data Granularity Through Spatio-Temporal Disaggregation	23
3.1 Introduction	23
3.2 Related Work	27
3.3 Datasets	29
3.4 Models	30
3.5 Experiments	36
3.6 Results	40
3.7 Summary	47
Chapter 4: Increasing Urban Data Coverage Through Feature Annotation With Vision-Language Models	48
4.1 Introduction	48
4.2 Related Work	51
4.3 Method	52
4.4 Evaluation & Results	55
4.5 Summary	60

Chapter 5:	Improving Urban Data Coherence Through Spatial Data Integration .	61
5.1	Introduction	61
5.2	Related Work	64
5.3	Task & Datasets	65
5.4	Experiments	68
5.5	Results	72
5.6	Ablation Studies	79
5.7	Summary	82
5.8	Appendix	83
Chapter 6:	Enhancing Urban Data Fidelity Through Pedestrian Pathway Generation	87
6.1	Introduction	87
6.2	Related Work	90
6.3	Method	92
6.4	Experiment	99
6.5	Results	101
6.6	Summary	106
Chapter 7:	Conclusion	107
7.1	Summary of the Dissertation	107
7.2	Cross-Cutting Insights	109
7.3	Limitations & Future Directions	110
7.4	Final Remarks	112

LIST OF FIGURES

Figure Number		Page
3.1	Disaggregation problem setting and its utilities. Our proposed Structurally-Aware Recurrent Network (SARN) is trained with coarse, aggregated data and structural containment maps. The trained model can be utilized in downstream applications, such as traffic prediction. It also can be adopted to disaggregate another variable through weight-sharing and fine-tuning with only a limited amount of data.	25
3.2	Four distinct geographic aggregation levels for the same NYC area. PUMA regions are the largest and most coarse. BLOCK regions are smallest and finest.	30
3.3	(a) Containment relationships between geographic level g^i and g^j . Each region from g^j is wholly contained in a region from g^i . (b) Binary containment map $M^{i,j}$ between geographic level g^i and g^j . (c) Structural-aware spatial attention is calculated among connected regions from two levels.	32
3.4	Model architecture of our proposed SARN. (a) Original architecture of the GRU model. Fully-connected layers are applied on input X^i . (b) SARN architecture. The fully-connected layers are replaced with our structurally-aware spatial attention (SASA) layers. (c) SASA layer. A SASA layer is composed with multiple spatial attention heads, each calculating global and structural attention across regions from two different geographic levels. The containment map is used to reflect the real-world structural hierarchy between geographic levels, and ensures the generation of coherent values across different resolutions.	34
3.5	Convergence plots of neural models. Top three plots: show NYC Taxi dataset. Bottom three plots: shows NYC Bikeshare dataset. Red line represents our proposed SARN model, and green line indicates the best results. In five out of six tasks, our SARN converged faster than other neural methods.	43
3.6	Transfer learning results on bikeshare dataset. The models are pre-trained with taxi data. The orange bar indicates the best performance for each disaggregation task.	45
3.7	Synthetic individual-level taxi points. (Left) Randomly selected single tract region, containing multiple block regions. (Right) The distributions of real and synthesized taxis ride over the block regions are qualitatively similar. . .	46

4.1	Pipeline of our proposed automated annotation process. Users input a pair of (satellite image, annotation guidance). The image will go through a set of processes including segmentation, filtering, and Set-of-Mark generation. Then the image and guidance will go through a vision-language model, the output of which is post-processed to produce the final annotation results. The procedure requires no fine-tuning, and can be applied on different features with minimal adjustment on the guidance.	50
4.2	(a) Filtered segmented candidates using SAM model. (b) No-Context. Candidate objects are presented separately in a new image. (c) In-Context. Candidate objects are labeled with numbers and bounding boxes within the original image.	53
4.3	Examples of annotated stop lines. Green and yellow outlines indicate perfect and approximate annotations, respectively. Red outlines indicate inaccurate annotations.	58
4.4	Examples of annotated raised tables. Green and yellow outlines indicate perfect and approximate annotations, respectively. Red outlines indicate inaccurate annotations.	59
5.1	Positive and negative examples for the spatial join and union task. (1) For the spatial join task, the blue line represents the road, and the red line represents the sidewalk. The task classifies whether a sidewalk “runs alongside” a road from a pedestrian’s perspective. (2) For the union task, both lines represent sidewalks. The task determines whether two sidewalks fully or partially represent the same real-world entity.	63
5.2	Plain prompting (left) and our heuristic prompting with features (right) for the spatial integration tasks. In the plain prompting, only the basic task description is provided. In the heuristic prompting, features derived from heuristics are included to further enhance performance.	70
5.3	Average results of heuristic prompting method for the spatial join and union task. Each row represents prompting with no heuristic (plain) or different heuristic information (hints or features), under either zero- or few-shot prompting.	74
5.4	Accuracies of review-and-refine method on both spatial join and union task. Each row represents different ways of generating initial answers, either randomly or using heuristics. We report accuracies when heuristic hints or features are provided.	78
5.5	Left: Distribution of <code>min_angle</code> in positive and negative spatial join data. Right: train accuracy on different feature thresholds for <code>min_angle</code> , for spatial join.	84

5.6	Left: Distribution of <code>min_distance</code> in positive and negative spatial join data. Right: train accuracy on different feature thresholds for <code>min_distance</code> , for spatial join.	84
5.7	Left: Distribution of <code>max_area</code> in positive and negative spatial join data. Right: train accuracy using different feature thresholds for <code>max_area</code> , for spatial join.	84
5.8	Left: Distribution of <code>min_angle</code> in positive and negative spatial union data. Right: train accuracy on different feature thresholds for <code>min_angle</code> , for spatial union.	84
5.9	Left: Distribution of <code>max_area</code> in positive and negative spatial union data. Right: train accuracy using different feature thresholds for <code>max_area</code> , for spatial union.	84
6.1	Qualitative examples of pedestrian pathway generation. Representative aerial intersection scenes illustrating common challenges for pedestrian mapping: {short segments, curvilinear segments, implicit paths (no crossing marks), occlusions from trees, and complex connectivity}. Top: ground-truth graph (blue; yellow dots mark vertices). Middle: Tile2Net prediction (red). Bottom: TraversRL-Local prediction (red). These examples highlight the topology-sensitive nature of pedestrian networks, and demonstrate TraversRL’s robust generation across diverse real-world challenges.	88
6.2	Supervised node expansion in TraversRL pretraining. Given the current node (yellow star), the set of ground-truth neighbors (red) are determined. During supervised pretraining, the policy is trained to sequentially emit one direction–distance action per ground-truth edge (blue arrows), adding the corresponding edges to the canvas. After all edges at the current node have been generated, the policy emits STOP and the algorithm moves to the next node for expansion (steps 1–5).	94
6.3	Local stepwise reward. We compare the buffered ground-truth region G (blue) with the current predicted buffer P_{t-1} (gray). Adding a new edge at step t contributes newly added buffered area ΔA_t (orange), of which the overlapped portion with ground-truth is ΔI_t . We define the stepwise reward $r_t = \Delta I_t / (\Delta A_t + \epsilon)$: it is high when the added edge aligns better with ground-truth (good step), moderate under partial alignment, and near zero when the edge adds off-target geometry ($\Delta I_t = 0$).	97

6.4	RL fine-tuning with local stepwise reward shaping (TraversRL-Local). For each intersection, we sample K rollouts (here $K = 4$) from π_θ , producing action sequences and generated graphs. A global reward assigns each rollout a terminal reward R_i , which is converted into group-relative advantages $\{A_i\}_{i=1}^K$. In parallel, a local reward computes per-step rewards $\{r_{i,t}\}$, which are normalized within each rollout to obtain $\{\tilde{r}_{i,t}\}$ and combined with A_i to form shaped advantages $\hat{A}_{i,t} = A_i + \lambda \tilde{r}_{i,t}$. The policy is updated using $\hat{A}_{i,t}$ together with $\text{KL}(\pi_\theta \parallel \pi_{\theta_0})$; only π_θ is updated, while π_{θ_0} is frozen. Without local shaping (TraversRL-Global), the update reduces to $\hat{A}_{i,t} = A_i$	98
6.5	RL fine-tuning produces cleaner, sparser graphs. Qualitative examples comparing the ground-truth graph (top, blue) to predictions from TraversRL-Pre (middle, red, “messy”) and TraversRL-Local (bottom, red, “cleaner”). Yellow dashed boxes highlight artifacts of the supervised policy — redundant branches, spurious short connectors, and noisy junction geometry —that are pruned after RL fine-tuning.	105

LIST OF TABLES

Table Number		Page
3.1	Disaggregation results from traditional and neural-based methods. The unit of the error is count/(block * hour), for comparison purpose across different disaggregation levels. Bold and <u>underline</u> numbers indicate the best and second result in the corresponding disaggregation task respectively. * indicates statistical significance.	41
3.2	Traffic demand prediction results at different geographic level. SARN column shows the results from models trained with the disaggregated data from SARN model, while Ground Truth columns shows the results from models trained with the ground truth value. $\%Difference = \frac{SARN - GroundTruth}{GroundTruth}$. Negative % Diff means better prediction results with SARN. STSGCN results at the BLOCK level is not reported due to out-of-memory issue.	44
4.1	Example text prompts for the VLM model to annotate stop lines. Pink highlighted texts represent the descriptive information of the features. Blue highlighted texts represent the general task questions. Direct Prompting method: given an image, we directly ask the VLM to annotate the feature of interest and return the bounding boxes. SoM Prompting method: we input VLM with the image from the SoM generation process.	56
4.2	Quantitative IoU results for direct prompting and the three Set-of-Mark prompting variants on stop lines and raised tables. Higher values indicate better spatial alignment with the ground-truth bounding boxes.	57
5.1	Baseline heuristic results for the spatial join and union tasks. The worst, the best, and the average accuracy (standard deviation) are reported for single, duo, and trio (spatial join only) heuristics. Standard deviations are calculated given different thresholds. For the spatial join task, duo results are averaged across all three duo combinations.	72

5.2	Common mistake examples in the reasoning steps from {4o-mini, qwen-plus, 4o}. The <i>Wrong Logic</i> category describes logical mistakes made by the models when analyzing spatial relationships between the paired objects. <i>Unclear Calculation</i> describes mistakes where there is no explicit demonstration by the models of the calculation steps used to arrive at an answer, though some numerical evidence may be provided. In such cases, models tend to use words such as “appears” and “approximately” rather than fully demonstrating their computations, casting uncertainty upon whether the models actually execute the calculation, and if so, how they do so. The <i>No Calculation</i> category is similar to the <i>Unclear Calculation</i> category. The difference is that, in this category, the model does not provide any numerical evidence whatsoever for its answers. The models tend to verbalize their assumptions, which they use to arrive at a final conclusion for a task.	76
5.3	Plain prompting only uses textual information, while plain+img prompting presents an additional satellite image as visual context. Heuristic prompting with features still has the best performance, indicating the importance of geometrical information.	80
5.4	Comparative performance of LLMs on fundamental computational geometry tasks that require spatial reasoning abilities. Accuracies are reported. . . .	81
5.5	Accuracy of heuristic-driven prompting method for the spatial join task. We evaluate both zero-shot and few-shot prompting, with plain task description, heuristic-hints or heuristic features.	85
5.6	Accuracy of heuristic-driven prompting method for the spatial union task. We evaluate both zero-shot and few-shot prompting, with plain task description, heuristic-hints or heuristic features.	86
6.1	Quantitative results on WashingtonInter (in-domain) and PathwayBench test cities (zero-shot transfer), reported as mean $\pm$ std over five seeds. <i>pre</i> , <i>global</i> , and <i>local</i> denote supervised pretraining, RL fine-tuning with a terminal graph-level reward, and RL fine-tuning with additional local stepwise reward shaping, respectively. Bold indicates the best value within each dataset/metric column. [†] denotes an improvement over both Tile2Net and the corresponding <i>pre</i> checkpoint (same backbone). * denotes an improvement over the corresponding <i>global</i> checkpoint (same backbone).	102
6.2	Graph cleanliness and connectivity agreement averaged over backbones and five seeds. AvgDeg denotes average node degree (lower indicates sparser graphs), and TravSim denotes traversability similarity (higher indicates closer agreement with ground-truth boundary connectivity). For Global and Local, we report the relative change compared to Pre on the same dataset.	104

ACKNOWLEDGMENTS

I would first like to express my deepest gratitude to my advisor, Dr. Bill Howe, for his guidance and support throughout my doctoral study. When I began my Ph.D., I had limited research experience in the urban domain. Bill gave me the space to explore different directions, helped me discover the questions that interested me most, and connected me with scholars who broadened my perspective. He taught me how to identify meaningful research questions, think critically about methods and evidence, and communicate my work more clearly. This dissertation would not have been possible without his mentorship.

Beyond my advisor, I have benefited from the guidance of many mentors. I am grateful to Dr. Aylin Caliskan for serving on both my advisory and supervisory committees and for her thoughtful feedback on my general exam and dissertation proposal. I thank Dr. Bo Zhao for serving as the Graduate School Representative on my committee and for his valuable suggestions throughout the examination process. I am also grateful to Dr. Jon E. Froehlich for kindly joining my committee on short notice. I thank Dr. Anat Caspi for her support of my urban studies and for helping me access data resources. I am grateful to Dr. Lucy Lu Wang for serving on my advisory committee and for offering guidance on research challenges I encountered along the way. I also thank Dr. Nic Weber for his regular support and check-ins, as well as for organizing the Freshplots event, where I received helpful feedback from colleagues. I am also grateful to Dr. Ott Toomet, with whom I served as a teaching assistant multiple times. Through those experiences, he gave me opportunities to teach class sessions early in my Ph.D., which helped me grow as an instructor.

I am also fortunate to have been surrounded by many amazing peers at the iSchool. My cohort began during the pandemic, when we were only able to interact with one another virtually. Even so, those circumstances did not stop us from building community, supporting one another, and learning from one another. I am grateful to the peers with whom I

have collaborated, including Bingbing Wen, Yiwei Yang, Zening Qu, Robert Wolfe, Lucas Rosenblatt, Chris Fu, Isaac Slaughter, Bernease Herman, and Eva Brown. Working with them taught me how valuable collaboration can be: each project brought different questions, methods, and ways of thinking. Their vision, perspectives, and expertise expanded my understanding of different research domains and helped me become a more thoughtful and open-minded researcher. I especially want to thank Sarah Nguyen and Hyeyoung Ryu. Their support and companionship, through all the hikes, trips, and conversations, made the journey feel less lonely and more meaningful. I am deeply grateful for their friendship.

I am also grateful to the close friends outside of school who have accompanied me through the ups and downs of this journey, including Jinyu Chen, Xiaoyu Jiang, Luman Huang, Nansu Xu, Qingpu Lv, Weijie Xu, Dr. Xi Fang, Dr. Di Liu, Hangbo Li, Hanxi Hu, Wei Yuan, Dr. Shuai Han, Yiman Wang, and Rongzichen Song. Our weekly gatherings, weekend activities, shared meals, and conversations were a constant source of joy. They reminded me that life during a Ph.D. is not only about research, but also about friendship. Their support gave me strength and helped me through the long path of graduate school.

I am deeply grateful to my family. My mother, Xiaoyan Zhou, and my father, Yushun Han, have supported me throughout this journey with their love, trust, and encouragement. No words can fully express my gratitude for everything they have given me, not only during my Ph.D., but throughout the many years I have spent pursuing my education far from home. Their selfless support made it possible for me to study in another country, keep going through difficult times, and become the person I am today.

Finally, to my partner, Jeffrey J. Campbell: thank you for being by my side through it all. Thank you for sharing my joys and celebrating with me when things went well. Thank you for comforting me when things were difficult and for being patient with my stress and frustration when I felt overwhelmed. Thank you for your positivity, support, patience, and unwavering belief in me. You made the hardest parts of this journey easier to carry and the happiest moments even more meaningful.

DEDICATION

To my parents and Jeffrey.

Chapter 1

INTRODUCTION

Urbanization is reshaping social, economic, and environmental systems worldwide [239, 207]. Cities concentrate population, infrastructure, services, and opportunities, but they also concentrate vulnerability. Challenges such as infrastructure strain, housing affordability, environmental risk, and unequal access to essential services often emerge most sharply in urban settings [10, 240, 241]. These challenges are not evenly distributed. They vary across neighborhoods, times of day, population groups, and infrastructure systems. As a result, the ability to observe, measure, and understand urban environments at appropriate levels of spatial, temporal, and semantic detail has become increasingly important for urban planning, especially in support of accessibility, equity, sustainability, and public well-being.

Understanding and resolving these challenges is fundamentally data-driven. Contemporary urban analysis relies on a wide range of data sources, including spatio-temporal mobility records, aerial imagery, infrastructure inventories, and graph-structured transportation networks. These data support urban planning through tasks such as traffic forecasting, land-use classification, map construction, and infrastructure monitoring. Advances in artificial intelligence have substantially improved the ability to leverage such data and to model complex urban phenomena, ranging from classical predictive methods to modern deep neural networks and advanced transformer-based architectures [246, 288, 178, 113, 152, 140, 130, 60, 28].

Yet for many urban analytics, modeling, and decision-making tasks, the main bottleneck is often not the lack of computational models, but the lack of urban data with sufficient quality. Urban data quality is multidimensional: useful datasets should be sufficiently fine-grained, complete in coverage, coherent across sources, and faithful to the real-world structures they represent. In practice, data may be too coarse to support neighborhood-scale analysis; important infrastructure may be absent from existing records; datasets from different sources may disagree or be difficult to reconcile; and map records may fail to

preserve the geometry and connectivity required for routing applications. These problems do more than reduce benchmark performance. They shape what can be seen, measured, and acted upon in the urban environment. Poor-quality urban data can obscure infrastructure gaps, misrepresent mobility patterns, and propagate downstream errors into decision-making workflows [231, 122, 290, 68, 9]. In this sense, urban data quality is not merely a technical concern. It is a prerequisite for reliable and equitable urban analytics.

Recent advances in deep learning and large foundation models provide more powerful tools for addressing these data quality challenges. Task-specific neural models are especially useful when the data structure and target output are well defined, such as in tasks that involve spatio-temporal dynamics, spatial hierarchies, or graph-based relationships commonly found in urban data. These models can be designed to capture the structure of a specific urban problem and optimize for the form of the desired output. In contrast, general-purpose large language models (LLMs) and vision-language models (VLMs) provide greater flexibility when tasks involve heterogeneous inputs, zero- or few-shot adaptation with limited labeled data, or context-dependent reasoning over text, imagery, and structured spatial information [205, 81, 270, 125, 109, 159, 306, 38, 311, 147].

This dissertation builds on these complementary capabilities to study how AI-based methods can improve urban data quality across diverse data forms, quality deficiencies, and intended uses. To organize this investigation, the dissertation focuses on four dimensions of urban data quality: **granularity**, **coverage**, **coherence**, and **fidelity**. Each dimension corresponds to a different kind of data deficiency that can limit urban analysis:

- **Granularity** refers to the spatial or temporal resolution at which urban phenomena are represented. Data that are overly aggregated may still be useful for capturing broad trends, but they often suppress neighborhood-scale patterns, local variation, and short-term dynamics. This limitation is especially consequential for mobility analysis, traffic modeling, and other applications that depend on fine-scale heterogeneity.
- **Coverage** refers to whether the relevant entities, features, or events are represented in the dataset at all. Urban datasets often omit long-tail or specialized infrastructure

elements because they are difficult or expensive to annotate. As a result, parts of the built environment can become digitally invisible even when they are important for accessibility, safety, or transportation planning.

- **Coherence** refers to whether multiple data sources can be reconciled into a consistent and interpretable representation. Urban data are frequently assembled from heterogeneous sources that differ in geometry, semantics, temporal scope, or data collection practices. Even when each source is useful in isolation, lack of coherence across sources can hinder integration and propagate uncertainty into downstream analysis.
- **Fidelity** refers to how faithfully urban data represent the real-world structure and function of urban systems. For graph-structured infrastructure such as pedestrian pathways, it is not enough for data to appear plausible. They must also preserve the connectivity and traversability needed for navigation and accessibility analysis.

There are two main reasons to focus on these four quality dimensions. First, they are common data quality concerns in urban datasets. Urban data are often released at coarse spatial or temporal resolutions, omit important features, differ across sources, or fail to preserve the structure needed for real-world use. Second, these concerns appear in high-impact urban applications. Mobility modeling and demand forecasting require fine-grained spatio-temporal data; accessibility and safety analysis require broad coverage of infrastructure features; planning and integration workflows require coherent records across sources; and routing or navigation requires faithful network geometry and connectivity. These dimensions therefore provide a practical way to organize recurring urban data problems and to study methods that improve urban data quality for downstream applications.

Building on these four dimensions, the central claim of this dissertation is that *high-quality urban data are a prerequisite for reliable and equitable urban analytics, and improving urban data quality requires targeted methods designed around the specific deficiencies, structures, and intended uses of the data.*

1.1 Research Objectives and Study-Specific Contributions

The dissertation is organized around four research objectives (ROs), each corresponding to one dimension and one methodological study in Chapters 3–6.

1.1.1 RO1: To Refine Urban Data Granularity Through Spatio-Temporal Disaggregation

Many open urban datasets are released in aggregated form because fine-grained observations can raise privacy concerns. Aggregation makes data easier to share, but it also removes the detailed information needed for fine-scale modeling, such as traffic demand forecasting at the neighborhood scale and cross-scale data integration. The objective of Chapter 3 is therefore to study whether coarse, irregular, privacy-preserving spatio-temporal data can be disaggregated into finer spatial units while preserving useful urban dynamics.

Chapter 3 introduces SARN, a Structurally-Aware Recurrent Network for spatio-temporal disaggregation of urban mobility data. SARN combines recurrent temporal modeling with structurally-aware spatial attention layers that capture both global spatial interactions and hierarchical containment relationships between coarse and fine geographic units. The chapter evaluates SARN on New York City taxi and bikeshare data across multiple disaggregation resolutions and shows that it outperforms both classical and neural baselines. The chapter further shows that the recovered fine-resolution data are practically useful: they support downstream demand prediction with performance close to models trained on ground-truth fine-scale data, transfer effectively across urban variables through fine-tuning, and can be used to synthesize realistic individual-level event patterns. Overall, the results demonstrate that learned spatio-temporal disaggregation can recover fine-grained urban information from privacy-preserving aggregates, thereby improving data granularity and expanding the practical value of open urban data.

1.1.2 RO2: To Increase Urban Data Coverage Through Zero-Shot Feature Annotation

Urban datasets often describe common infrastructure, such as roads, but omit many long-tail features that matter for accessibility, safety, and transportation planning, such as curb ramps, crosswalks, and other localized built-environment elements. These features are not

necessarily rare in importance, but they are often rare in available labels. Manual annotation is expensive, and supervised computer vision methods require sufficient training data that may not exist for specialized feature types or for every city. The objective of Chapter 4 is to examine whether vision-language models can help expand urban data coverage by producing feature annotations from aerial imagery without task-specific fine-tuning.

Chapter 4 proposes a modular zero-shot annotation pipeline that combines class-agnostic segmentation, lightweight candidate filtering, and vision-language-model prompting. Instead of asking a model to locate urban features directly from an image, the pipeline first generates candidate regions and then uses structured prompting to guide the model’s selection among those candidates. The chapter evaluates the pipeline on stop lines and raised tables, two transportation-relevant features that are useful for urban analysis but are not well represented in standard datasets. The experiments show that direct prompting is largely ineffective, while candidate-based prompting substantially improves annotation quality. Overall, the results demonstrate that general-purpose vision-language models, when embedded in a structured annotation pipeline, can provide a practical starting point for expanding urban data coverage and supporting downstream use.

1.1.3 RO3: To Improve Urban Data Coherence Through Spatial Data Integration

Urban datasets are frequently assembled from multiple sources that are complementary but inconsistent in geometry, scale, or representation. These inconsistencies complicate data fusion and fragment the digital description of the same underlying infrastructure. The objective of Chapter 5 is to study whether LLMs can support the accurate integration of urban data from multiple sources when provided with different forms of information, including natural-language task descriptions, spatial hints, and precomputed geometric features.

Chapter 5 studies two spatial integration tasks over pathway network data: spatial join and spatial union. The spatial join task asks whether a sidewalk runs alongside a road from a pedestrian perspective, while the spatial union task asks whether two sidewalk annotations from different sources refer to the same real-world sidewalk. The chapter evaluates multiple large language models under different prompting settings. The core finding

is that LLMs have useful spatial reasoning abilities, but they are unreliable at translating natural-language task descriptions into the precise computational geometry needed for spatial matching. However, when provided with precomputed geometric features such as distance, angle, and overlap, the models can infer context-sensitive decision rules and achieve strong performance. Overall, the chapter shows that LLMs are most effective for spatial data integration not as standalone geometry solvers, but as flexible decision interfaces built on human-interpretable geometric features. In this role, they can improve the coherence and utility of heterogeneous urban data by supporting more accurate integration.

1.1.4 RO4: To Enhance Urban Data Fidelity Through Pedestrian Pathway Generation

For routing analysis, it is not enough to know where pedestrian infrastructure appears in an image. The data must also represent how pathway segments connect and whether the resulting network supports traversal. Existing segmentation-based methods can identify visible pathway surfaces, but they often produce fragmented graphs that appear plausible while remaining incomplete for navigation. The objective of Chapter 6 is to generate pedestrian pathway graphs that preserve both real-world geometry and traversability.

Chapter 6 introduces TraversRL, a vision-conditioned iterative generator for pedestrian pathway networks. TraversRL constructs a graph step by step from aerial imagery. At each decision point, it observes a local satellite crop and a generation canvas encoding the partial graph, then predicts the next action. The model is first trained through supervised pretraining to imitate ground-truth graph expansion, and then improved through reinforcement learning with graph-level and local stepwise rewards. The chapter evaluates TraversRL across multiple backbones and datasets, compares it with segmentation-based pathway extraction methods, and shows that reinforcement learning improves both buffered geometry and traversability-oriented graph quality. It also demonstrates zero-shot generalization to other cities and produces cleaner graphs with fewer redundant edges. Overall, the results show that generating pedestrian pathways as connected graphs, rather than only detecting visible pathway surfaces, can improve the fidelity and practical utility of urban network data for routing and accessibility analysis.

1.2 *Dissertation-Level Contributions*

Beyond the individual methods developed in each chapter, this dissertation makes three broader contributions to the improvement of urban data quality.

First, it develops a data-quality-centered framework for understanding urban data problems. Rather than treating data quality as a one-dimensional issue, the dissertation organizes it around four dimensions: granularity, coverage, coherence, and fidelity. These dimensions provide a structure for identifying why urban datasets may be insufficient for analysis and what kinds of methods are needed to improve them. Through this framework, the dissertation connects four technical problems (spatio-temporal disaggregation, feature annotation, spatial integration, and pathway graph generation) under the shared goal of producing high-quality urban data that better support analysis and decision-making.

Second, it broadens the methodological scope of urban data quality improvement. The dissertation shows that improving urban data quality can involve multiple forms of data transformation, including recovering hidden fine-scale signals, identifying missing built-environment features, reconciling heterogeneous spatial records, and constructing network data from imagery. This perspective moves beyond viewing urban data as a fixed input to downstream models. Instead, it treats urban data itself as an object of computational improvement, where AI-based methods can help improve the quality of urban data.

Third, it identifies the importance of structured intermediate information in urban AI methods. Across the four studies, strong performance does not come from applying learning models to raw data alone. SARN uses geographic containment and spatial interaction structure to guide disaggregation. The annotation pipeline uses segmentation, filtering, and candidate marking to constrain visual reasoning. The integration framework uses geometric features to support language-model decisions. TraversRL uses local and graph-level reward signals to guide pathway graph generation. Overall, these findings suggest that AI-based methods for urban data quality improvement are most effective when they are paired with cues, constraints, or feedback signals that reflect the structure of the urban problem. This supports the dissertation’s broader claim that improving urban data quality requires methods designed around the specific deficiencies, structures, and intended uses of the data.

1.3 *Dissertation Organization*

The remainder of this dissertation is organized as follows. Chapter 2 reviews the background literature relevant to this dissertation, including learning-based urban applications, urban remote sensing, large language models, vision-language models, and reinforcement-learning-based post-training. Chapter 3 focuses on the granularity dimension and presents SARN, a Structurally-Aware Recurrent Network for spatio-temporal disaggregation of aggregated mobility data. Chapter 4 focuses on the coverage dimension and presents a zero-shot annotation pipeline for extracting underrepresented urban features from aerial imagery. Chapter 5 focuses on the coherence dimension and investigates LLM-assisted spatial integration through spatial join and spatial union tasks. Chapter 6 focuses on the fidelity dimension and presents TraversRL, a vision-conditioned iterative generator for pedestrian pathway networks from aerial imagery. Finally, Chapter 7 concludes the dissertation by summarizing the main findings, discussing limitations, and outlining directions for future research.

Chapter 2

BACKGROUND

This chapter introduces the background needed to understand the four core studies of the dissertation. It is organized around two bodies of literature. First, it reviews learning-based techniques for urban applications, with a focus on spatio-temporal prediction and urban remote sensing. Spatio-temporal prediction provides background for Chapter 3, where spatial and temporal patterns are used to refine the granularity of aggregated mobility data. Urban remote sensing provides background for Chapters 4 and 6, where aerial imagery is used to improve urban data coverage through feature annotation and urban data fidelity through pedestrian pathway graph generation.

Second, the chapter reviews recent developments in large foundation models and post-training methods, including large language models (LLMs), vision-language models (VLMs), and reinforcement-learning-based fine-tuning. These methods are relevant because the dissertation uses both task-specific neural models and foundation-model-based approaches. LLMs provide background for Chapter 5, which studies spatial data integration with language-model reasoning. VLMs provide background for Chapter 4, which uses them to annotate urban features, as well as for the visual ablation studies in Chapter 5. Reinforcement learning provides background for Chapter 6, where pathway generation models are fine-tuned using rewards defined over graph-level quality.

At a high level, these literatures show why improving urban data quality requires methods that are sensitive to the structure of the data, the type of deficiency being addressed, and the intended use of the resulting data. This chapter therefore provides the technical foundation for the dissertation’s broader argument: improving urban data quality requires methods designed around the limitations and use cases of the data itself.

2.1 Urban Applications & Learning-Based Techniques

Urban computing spans a wide range of applications, from mobility forecasting to infrastructure mapping. Although these tasks differ in their specific goals and data sources, many share a common challenge: strong performance requires learning from data that are noisy, incomplete, heterogeneous, spatially structured, and temporally dynamic. As a result, learning-based techniques have become central to modern urban analytics. This section reviews two groups of applications and techniques relevant to this dissertation. The first group focuses on spatio-temporal prediction tasks, including traffic prediction, air quality forecasting, and energy consumption estimation. These tasks provide background for Chapter 3, which improves urban data granularity through spatio-temporal disaggregation. The second group focuses on urban remote sensing, with attention to two types of extraction tasks from aerial imagery: surface feature and network feature extraction. Surface feature extraction provides background for Chapter 4, which improves urban data coverage through VLM-based annotation. Network feature extraction provides background for Chapter 6, which improves urban data fidelity through pedestrian pathway graph generation.

2.1.1 Spatio-Temporal Prediction Tasks

Many important urban prediction tasks are inherently spatio-temporal. They involve variables that change over time, vary across locations, and interact through spatial relationships such as distance, adjacency, network connectivity, or shared regional context. In a typical setting, observations are collected over a set of spatial units, such as road segments, traffic sensors, monitoring stations, neighborhoods, or grid cells, and repeated across discrete time intervals. The goal is to estimate future values from historical patterns while accounting for both temporal dynamics and spatial dependencies. Representative examples include traffic prediction, air quality forecasting, and energy consumption estimation. These tasks provide useful background for Chapter 3, which studies spatio-temporal disaggregation. Although disaggregation is not a forecasting task in the strict sense, it shares a key modeling challenge with prediction: both require learning how urban signals vary across space and time, especially when observations are noisy, sparse, aggregated, or irregularly structured.

Traffic Prediction: Traffic prediction is a central task in urban computing and Intelligent Transportation Systems (ITS). Accurate traffic prediction supports both real-time operations and long-term transportation decision-making [69], including route planning [137], traffic signal optimization [261], congestion pricing [257], and travel time forecasting [44]. These applications require timely and reliable estimates of how traffic conditions change across road segments, regions, and time periods. Traffic prediction includes several related tasks, such as traffic flow, traffic demand, and speed prediction. Although these tasks differ in their data sources and spatial units, they share a common objective: estimating future traffic conditions based on historical observations and spatial context.

Traffic prediction has been studied since the late 1970s [244]. Early approaches relied on statistical and traditional machine learning methods, including ARMA [209], ARIMA [268], SVM [184], Random Forest [91], and Gradient Boosting [280]. These methods can be effective in relatively simple settings, but they often struggle to capture the non-linear and time-varying dependencies found in mobility data. Recent deep learning methods have addressed these limitations by learning spatial and temporal representations directly from data. Convolutional neural networks (CNNs) have been used to model spatial patterns, while recurrent neural networks (RNNs) and related sequence models have been used to capture temporal dependencies [166, 298, 292, 283, 312]. However, CNN-based models are most natural for grid-like data and can be less suitable for non-Euclidean structures such as road networks or irregular administrative boundaries. This limitation has motivated graph neural network (GNN)-based methods, which represent spatial units as nodes and spatial relationships as edges. Early spatio-temporal GNN models combine graph-based spatial feature extraction with temporal modules, such as STGCN [289], GraphWave [167], ASTGCN [74], AGCRN [8], and STGNN [254]. Recent work has incorporated attention mechanisms and transformer architectures, such as MegaCRN [111], DDGCRN [267], STWave [59], STAEformer [151], and STD-MAE [63], to better model long-range temporal dependencies and dynamic spatial interactions. Recent studies continue to develop more expressive spatio-temporal architectures, including pivotal graph neural networks [123], efficient graph transformers [250], transformer–multi-graph GCN hybrids [297], multi-scale spatio-temporal GNNs [26], and causal-attention-based spatio-temporal transformers [86].

Air Quality Forecasting: Air quality forecasting is also a representative spatio-temporal urban prediction task. It aims to forecast future pollution levels for a region using historical measurements from monitoring stations. Common prediction targets include pollutant concentrations such as Particulate Matter (PM2.5), Sulfur Dioxide (SO2), and other regulated pollutants, as well as composite indicators such as the Air Quality Index (AQI) [105]. Air quality forecasting is important because air pollution directly affects public health and environmental quality. Accurate forecasts can support early warning systems, help agencies respond to hazardous pollution events, and guide interventions to protect public health [12].

Air quality forecasting is challenging because pollution patterns are shaped by multiple interacting factors, including emissions, meteorology, land use, transportation activity, and atmospheric diffusion. Traditional methods include statistical models [128, 100, 31] and machine learning approaches [118, 183, 204]. More recent studies use deep learning models such as CNNs, GNNs, and RNNs to capture spatial variation and temporal dependencies across monitoring stations [196, 308, 58, 101, 61]. Attention-based and transformer-based methods have further improved performance by modeling long-range dependencies and heterogeneous spatial interactions [248, 249, 186, 146, 49, 93, 192]. Recent work has further explored graph-Transformer hybrids for PM2.5 prediction [287], spatio-temporal inverted transformers for ambient air pollution forecasting [299], GNN-Transformer models [88], graph-based spatio-temporal feature fusion [20], and cross-regional deep learning frameworks for multiple pollutants [16]. As with traffic prediction, the quality and density of observations strongly affect model performance. Sparse monitoring stations, uneven spatial coverage, and mismatched spatial units can limit model performance.

Energy Consumption Estimation: Energy consumption estimation focuses on forecasting future energy use within specific regions or systems, often at hourly or daily intervals [64]. Prediction targets include electricity demand, heating and cooling loads, or total energy consumption across residential or commercial sectors. Accurate energy forecasting is important for resource planning, grid management, sustainability analysis, and smart city operations. Reliable forecasts help utilities balance supply and demand, support renewable energy integration, and guide policies for energy efficiency and emissions reduction.

Energy consumption forecasting also exhibits strong temporal and spatial structure. Demand varies by time of day, day of week, season, weather conditions, land use, building type, and human activity patterns. Traditional forecasting approaches have increasingly been complemented by deep learning methods that can model non-linear temporal dependencies and interactions among multiple covariates [52, 195, 144, 129]. Recent transformer-based architectures have also been applied to energy forecasting because of their ability to capture long-range temporal dependencies and complex contextual patterns [23, 164, 179]. More recent studies have examined city-scale electricity forecasting with modern machine learning methods [194], spatio-temporal attention models for short-term energy consumption [82], transformer-based multi-building energy prediction [180], transfer learning for transformer-based building energy forecasting [223], and adaptive ensemble models for smart-building energy consumption forecasting [182]. Similar to traffic and air quality prediction, the usefulness of these methods depends not only on model design, but also on the spatial and temporal resolution, completeness, and consistency of the underlying data.

Summary: Across these prediction tasks, a common lesson emerges: strong models still depend on appropriate data resolution and data quality. When observations are too sparse, too coarse, or inconsistent across spatial and temporal units, predictive performance and generalization can suffer. This lesson directly motivates the dissertation’s focus on improving urban data quality. Chapter 3 builds on the spatio-temporal modeling literature, but shifts the problem from predicting future values to refining the spatial granularity of observed data. Instead of treating coarse or aggregated data as a fixed limitation, it asks whether models can learn spatial interactions and temporal dynamics from aggregated observations to recover finer-resolution urban signals that better support downstream analysis.

2.1.2 Urban Remote Sensing

Remote sensing imagery provides a scalable view of urban form and infrastructure. High-resolution aerial and satellite imagery make it possible to observe buildings, roads, sidewalks, vegetation, pavement markings, and other elements of the built environment over large geographic areas. These data support applications such as land-use classification, building

and roadway extraction, environmental monitoring, infrastructure inventory construction, and map updating. Compared with field surveys or manual digitization, remote sensing offers broader spatial coverage and more frequent updates, making it especially valuable for cities where infrastructure conditions change over time. These capabilities are directly relevant to Chapter 4 and 6: Chapter 4 uses aerial imagery to annotate underrepresented urban features, and Chapter 6 generates pedestrian pathway graphs from imagery.

Surface Feature Extraction: Historically, urban feature extraction from remote sensing imagery relied heavily on manual annotation and expert interpretation. Although such workflows can produce accurate results, they are labor-intensive, difficult to scale, and slow to update. Deep learning has greatly expanded the feasibility of large-scale urban scene understanding by enabling automated segmentation, detection, and classification from imagery. CNN-based models such as dilated CNNs, encoder-decoder networks, and attention-enhanced architectures have been widely used for extracting roads, buildings, and other common urban features [286, 83, 251, 75, 46, 157, 230, 36]. Transformer-based segmentation models have further improved the ability to capture global context, long-range spatial dependencies, and multi-scale urban structure [282, 276, 95, 296, 67, 293]. These methods have produced strong results for common categories, especially roads and building footprints, where labeled training examples are relatively more available.

However, the success of deep learning for common infrastructure does not fully solve the coverage problem in urban data. Many transportation- and accessibility-relevant features remain underrepresented in standard datasets and benchmarks. Existing work has studied road extraction, building footprint extraction, sidewalk extraction, and crosswalk detection from aerial or satellite imagery [213, 215, 265, 185, 245, 163]. Some recent studies have begun to address more specialized pedestrian and roadway features, such as crosswalk changes from bi-temporal aerial imagery, turning-lane pavement markings, curb ramps, and tactile paving [5, 4, 171, 237]. Nevertheless, these efforts are still much narrower than the full range of infrastructure elements needed for accessibility, safety, and transportation planning. Detailed features such as stop lines, raised tables, curb ramps, tactile paving, walk signals, and other localized design elements are often small, visually subtle, partially occluded, or

shaped by local design standards. They are also expensive to annotate at scale. As a result, there remains a practical gap between what deep learning methods could extract in principle and what urban datasets actually contain. This gap motivates the zero-shot annotation strategy in Chapter 4, which uses segmentation, candidate filtering, and vision-language prompting to reduce the dependence on task-specific labeled training data.

Network Feature Extraction: Beyond surface-based feature extraction, remote sensing is also used to extract network features represented as lines or graphs. Roads, sidewalks, trails, and pedestrian pathways are important because they define possible movement through the city. For these infrastructures, accurate geometry alone is insufficient; usable digital data must also preserve connectivity and traversability. Existing network extraction methods broadly follow two paradigms. The first is a segmentation-first pipeline, where models predict dense road or pathway masks and then recover graph structure through vectorization, thinning, skeletonization, or heuristic post-processing [43, 92, 273, 89, 227, 314, 162, 173]. This approach benefits from mature segmentation architectures, but small mask errors can lead to disconnected links, noisy branches, or incorrect intersections after graph conversion. The second paradigm directly constructs graph structure from imagery. RoadTracer [11] introduced a crop-and-canvas tracing framework that grows a road graph step by step using local image evidence and the already-generated graph. VecRoad [229] improved local exploration at complex junctions, while RNGDet and RNGDet++ [274, 275] developed transformer-based iterative graph detection with topology-oriented supervision. More recently, NETracer [148] proposed a topology-aware tracing framework that explicitly models node–edge connectivity. These studies show the importance of moving beyond pixel-level masks when the final output must support routing or network analysis.

Similar representation challenges arise for pedestrian infrastructure, but they are often harder than road extraction. Pedestrian pathways are typically narrower, contain many short and curvilinear segments, are more susceptible to occlusions from trees, vehicles, or shadows, and require accurate crossing and intersection connectivity for accessibility and navigation applications [97, 305]. Compared with road extraction, pedestrian pathway extraction from overhead imagery remains relatively understudied. Early work by Karimi et

al. [117] explored pedestrian map generation at small scale and highlighted the importance of data quality. Tile2Net [97] detects sidewalks, crosswalks, and footpaths from aerial imagery and converts them into pedestrian networks. Prophet [304] combines road-network priors with aerial-image segmentation and rasterized street maps to infer pathway graphs. PathwayBench [305] further argues that pixel-level metrics can understate mobility-critical errors and instead emphasizes routability and intersection-scale connectivity. Other studies focus on specific pedestrian assets rather than full connected networks, including zebra-crossing detection [1], crosswalk detection from satellite imagery [245, 14], and curb-ramp detection from street-view imagery [84, 187]. These limitations are especially consequential for pedestrian applications, where a short missing link, an incorrect crossing, or a broken intersection can substantially affect routability and accessibility analysis. This problem motivates Chapter 6, which treats pathway generation as a graph-construction problem and optimizes graph-level fidelity rather than only pixel-level segmentation quality.

Summary: Across urban remote sensing tasks, a common lesson is that visual extraction alone is not sufficient for high-quality urban data. Deep learning has greatly improved the ability to detect and segment common urban features from aerial and satellite imagery, but many important infrastructure elements remain underrepresented in available datasets. Moreover, for networked infrastructure such as roads, sidewalks, and pedestrian pathways, useful data must preserve not only geometry but also connectivity and traversability. These challenges motivate the dissertation’s focus on improving urban data coverage and fidelity. Chapter 4 builds on remote-sensing-based feature extraction, but shifts attention toward zero-shot annotation of underrepresented urban features. Chapter 6 builds on network extraction research, but treats pedestrian pathway mapping as a graph-construction problem rather than a segmentation-only task.

2.2 *Large Foundation Models*

In addition to task-specific models designed for particular urban prediction or extraction tasks, researchers can now draw on large foundation models with broader and more flexible capabilities. These models can be adapted through prompting, instruction tuning, tool use, and reinforcement-learning-based post-training. They can also operate over diverse inputs, including text, imagery, and structured spatial data. These capabilities are especially relevant to urban data problems because urban datasets are often heterogeneous: a single task may require combining natural-language descriptions, aerial images, and geographic objects such as points, lines, and polygons. This section reviews three developments most relevant to the dissertation. First, large language models (LLMs) provide flexible text-based interfaces for context-dependent reasoning, which motivates Chapter 5 on spatial data integration. Second, vision-language models (VLMs) extend these capabilities to image-text settings, providing background for Chapter 4 on zero-shot urban feature annotation and for the visual ablation studies in Chapter 5. Third, reinforcement-learning-based post-training provides a way to optimize generated outputs using rewards defined over final artifacts, which motivates Chapter 6 on pedestrian pathway graph generation.

2.2.1 *Large Language Models*

Large language models (LLMs) are foundation models designed to process and generate natural language. Most contemporary LLMs are based on the Transformer architecture [243] and are pretrained on large text corpora using self-supervised objectives, most commonly causal language modeling, in which the model predicts the next token conditioned on prior context [199, 202, 18]. This training process allows LLMs to acquire broad linguistic, semantic, and procedural knowledge from large-scale corpora. As model size & data scale, and training techniques have improved, LLMs have become increasingly capable of performing diverse tasks through prompting. Major model families, including LLaMA [234, 235, 71], Qwen [278, 277], Mistral [108], DeepSeek [41], GPT [18, 188, 218], Gemini [66], Claude [3], Grok [271], and others, illustrate the rapid development of general-purpose language models across both open-weight and proprietary settings.

A key capability of LLMs is in-context learning: the ability to adapt to a task from instructions and examples provided in the prompt. Zero-shot prompting asks the model to perform a task from a natural-language instruction alone, while few-shot prompting provides a small number of input–output examples to guide the model’s behavior [18]. Chain-of-thought prompting further improves performance on reasoning tasks by encouraging the model to produce intermediate reasoning steps before giving a final answer [263]. Related strategies extend this idea in different ways. Self-consistency samples multiple reasoning paths and selects the most consistent answer [256], while Tree-of-Thoughts prompting explores multiple candidate reasoning trajectories and allows the model to evaluate, backtrack, and search over intermediate steps [284]. Another line of work studies self-correction and iterative refinement, where an LLM first generates an output and then critiques or revises it, either using its own feedback or feedback from another model [168, 120, 116]. These prompting methods show that LLMs can often be adapted to new tasks through input design and inference-time reasoning strategies, even when labeled training data are limited.

Instruction tuning methods have further improved the practical usability of LLMs. During pretraining, models learn general capabilities for understanding and generating text by modeling large-scale language corpora. Instruction tuning then teaches models to use these capabilities to follow explicit task requirements, perform requested actions, and produce outputs in formats expected by users [262]. As a result, modern LLMs are used not only for text generation [188], but also for summarization [301], information extraction [193], text classification [53], code generation [316], planning [119], and structured decision-making [24]. In this dissertation, the most relevant property of LLMs is their ability to combine task instructions with auxiliary information and produce context-dependent judgments.

Recent work has also expanded LLMs beyond standalone text generation by connecting them to external knowledge sources and tools. Retrieval-augmented generation combines parametric knowledge in the model with external retrieved information, improving factual grounding for knowledge-intensive tasks [134]. Tool-use and agent-style methods allow LLMs to call external functions, search systems, calculators, or domain-specific tools when their internal reasoning is insufficient [210, 285]. This is especially important for tasks that require exact computation, database lookup, or interaction with structured environments.

Related work on structured data reasoning shows that LLM performance depends strongly on how tables, graphs, or other structured inputs are represented in the prompt. Studies on structured table understanding [226], as well as frameworks such as StructGPT [110] and Chain-of-Table [259], demonstrate that LLMs can reason more effectively over structured data when the input is transformed into task-relevant intermediate representations.

These developments are highly relevant to geospatial and urban data tasks. Urban data often combine natural-language descriptions, geographic geometries such as points, lines, and polygons, and domain-specific rules. LLMs offer a promising interface for such heterogeneous information because they can interpret task instructions, compare alternatives, and apply flexible decision rules. As a result, researchers have increasingly explored the use of LLMs in geospatial contexts, including studies on geographic knowledge encoded in LLMs, natural-language interfaces for geospatial analysis, and LLM-based geospatial copilots for GIS workflows [169, 253, 220]. These studies suggest that LLMs have substantial potential to help make geospatial analysis more accessible and flexible, especially when tasks involve a mixture of natural language, structured data, and domain-specific reasoning.

Motivated by this growing interest, Chapter 5 studies the use of LLMs for spatial data integration. Spatial integration is a natural setting for exploring LLMs in the geospatial domain because it often requires reasoning across heterogeneous sources, interpreting task-specific criteria, and making context-dependent decisions about whether spatial records should be linked or merged. Specifically, Chapter 5 examines whether LLMs can assist spatial join and spatial union tasks when provided with different forms of task information, including natural-language descriptions, spatial hints, and precomputed geometric features. In doing so, the chapter contributes to the broader effort to understand how LLMs can support geospatial workflows that combine language, geometry, and domain-specific reasoning.

2.2.2 Vision-Language Models

In this dissertation, the term vision-language model (VLM) is used broadly to refer to models that connect visual and textual information. This category includes both earlier image-text representation models and more recent LLM-based VLMs for open-ended visual instruction

following. Some modern VLMs also belong to broader multimodal foundation-model systems and may support image generation. However, this dissertation focuses on their visual understanding capabilities rather than their image synthesis capabilities. Specifically, VLMs are used here as models that take images and text as input and produce text-based outputs, such as descriptions, answers, classifications, or candidate selections.

Early and influential VLMs, such as CLIP, learn aligned visual and textual representations by training on large-scale image–text pairs [201]. This alignment enables models to connect visual content with natural-language concepts and supports tasks such as image–text retrieval [201], zero-shot classification [201], image captioning [139], visual question answering [228], and referring expression understanding [115]. Compared with conventional supervised vision models, VLMs are attractive because they can use language as a flexible interface for specifying visual concepts, including categories or descriptions that may not appear in a fixed training label set. Recent LLM-based VLMs, such as Flamingo [2], BLIP-2 [138], InstructBLIP [37], LLaVA [153], GPT-4V [189], Gemini [66], Qwen-VL [7], and InternVL [30] illustrate the rapid development of general-purpose VLMs. These models differ in architecture and training strategy, but they share a common goal: aligning visual perception with language-based reasoning and instruction following. As a result, VLMs make it increasingly feasible to query image content using flexible natural-language instructions rather than training a dedicated detector or classifier for every target category.

Recent remote-sensing VLMs and datasets, such as RemoteCLIP [149], GeoRSCLIP [307], GeoChat [127], and EarthGPT [302], reflect growing interest in adapting VLMs to overhead imagery and geospatial interpretation in the urban setting. These studies show that VLMs can support remote-sensing tasks such as image–text retrieval, scene understanding, visual question answering, grounding, and region-level dialogue. They also highlight an important challenge: remote-sensing images differ substantially from natural images in viewpoint, scale, object size, and spatial context, so effective use of VLMs often requires task-specific prompting, domain adaptation, or structured intermediate inputs.

In this dissertation, Chapter 4 uses VLMs for low-label urban annotation, where the goal is to identify underrepresented built-environment features from aerial imagery without task-specific fine-tuning. Instead of asking a VLM to locate a feature directly from the raw

image, the chapter frames annotation as a constrained candidate-selection problem. Segmentation and filtering first produce candidate regions, and visual marks then make those candidates explicit to the VLM. This design uses the visual understanding and language-based reasoning capabilities of VLMs while reducing the open-ended localization burden. Chapter 5 also briefly evaluates VLMs for spatial integration by providing visual inputs for spatial join and spatial union tasks. Together, these uses connect VLMs to both urban data coverage and visual support for spatial reasoning.

2.2.3 Reinforcement Learning for Large Foundation Models

Beyond pretraining and supervised fine-tuning, modern foundation models are often improved through post-training methods that optimize output-level objectives. A central motivation is that some desired behaviors are difficult to specify through step-by-step supervision alone. In supervised fine-tuning, the model is usually trained to imitate local targets, such as the next token in a reference response. However, the quality of a generated response, caption, plan, or graph often depends on the completed output rather than on each individual token or action in isolation. In these settings, reinforcement learning (RL) provides a way to optimize models using rewards defined over generated sequences, trajectories, or final outputs. The model generates candidate outputs, receives a reward or preference signal, and is updated to make higher-reward outputs more likely.

In large foundation models, this idea is most commonly discussed through reinforcement learning from human feedback (RLHF). A typical RLHF pipeline begins with supervised instruction tuning, then trains a reward model from human preference comparisons, and finally fine-tunes the policy to produce outputs preferred by the reward model [33, 317, 225, 191]. Proximal Policy Optimization (PPO) has been one of the most widely used policy optimization methods in this setting because it stabilizes updates by limiting how far the updated policy can move from the previous policy in a single step [211]. RLHF and related approaches have been used to improve helpfulness, harmlessness, summarization quality, instruction following, and general alignment with user preferences. More recent post-training methods have explored alternatives to classical PPO-based pipelines. Direct Preference Optimiza-

tion (DPO) [203] and Kahneman–Tversky Optimization (KTO) [57] optimize preferences without an explicit PPO-style reinforcement-learning loop. Other methods, such as Group Sequence Policy Optimization (GSPO) [310] and Group Reward–Decoupled Normalization Policy Optimization (GDPO) [156], continue to develop RL-based optimization objectives for improving stability, efficiency, or multi-reward training.

Group Relative Policy Optimization (GRPO) is especially relevant to this dissertation. GRPO was introduced in DeepSeekMath as a variant of PPO designed to improve mathematical reasoning while reducing the memory cost associated with actor–critic methods [214]. Instead of estimating a separate value function, GRPO samples a group of candidate outputs for the same prompt, scores each output with a reward function, and computes the advantage of each output relative to the group. Outputs that perform better than the group baseline are encouraged, while weaker outputs are discouraged. This relative comparison is useful when the quality of a generated output is easier to measure after completion than to supervise at each local step. GRPO has also been used in reasoning-oriented models such as DeepSeek-R1, where reinforcement learning is used to encourage stronger reasoning behaviors without relying only on supervised reasoning traces [42].

Although this dissertation does not use reinforcement learning to fine-tune an LLM, the same principle is central to Chapter 6. Pedestrian pathway generation is a sequential graph-construction problem: the model repeatedly chooses actions that extend or stop a partial graph, and the quality of the final output depends on graph-level geometry, connectivity, and traversability. These objectives are difficult to optimize through supervised learning alone because a locally plausible action may still lead to a poor final graph, while the value of a useful action may only become clear after several later steps. Reinforcement learning therefore provides a natural mechanism for fine-tuning the generator toward the quality measures that matter for the completed pathway network. Chapter 6 adapts this idea by using rewards over generated pathway graphs, including global rewards that evaluate the completed graph and local stepwise rewards that provide more direct credit assignment during rollout. In this way, RL serves as a bridge between local generation decisions and the final fidelity of the generated urban network.

Chapter 3

REFINING URBAN DATA GRANULARITY THROUGH SPATIO-TEMPORAL DISAGGREGATION

While urban open data is critical to modern city science, it is frequently released in spatially aggregated forms to safeguard individual privacy. This necessary protection, however, creates a “granularity gap” that hinders fine-scale analysis and the integration of disparate datasets. This chapter addresses granularity, the first dimension of urban data quality, through the task of spatio-temporal disaggregation. We seek to recover high-resolution signals from coarse, irregular administrative regions. To achieve this, we introduce the Structurally-Aware Recurrent Network (SARN), a novel architecture designed to jointly model cross-region spatial interactions, temporal dependencies, and the hierarchical containment relations inherent in geographic boundaries. Through extensive experiments on New York City taxi and bikeshare data, we demonstrate that SARN recovers fine-grained urban patterns with significantly higher accuracy than existing heuristic and neural baselines, effectively restoring the utility of fine-grained urban data. This chapter is based on our work published at ACM SIGSPATIAL 2024 [77].

3.1 Introduction

Open data affords the study of urban dynamics [246, 291], but, with few exceptions [303, 219], models assume access to individual-level data, while privacy concerns motivate the release of aggregated data [72] such that aggregated data tends to be the norm in some fields [19, 160, 131]. Although aggregation can protect individual privacy [238, 55, 90] (without strong guarantees [107, 17]), it complicates integration of multiple datasets (e.g., aligning tract-level to block-level data) and destroys information useful for training. For example, demographers have studied disaggregation techniques to uncover local hotspots erased by tract-level aggregation [177]. The New York City (NYC) taxi open datasets serve as another notable example, widely used in spatio-temporal traffic forecasting liter-

ature. Much of the work has relied on NYC taxi data collected before June 2016, when individual-level records were accessible [174, 300, 112, 76]. Subsequently, taxi records have been released at aggregated taxi zone levels, forcing applications to rely on pre-2016 data.

Motivated by these issues, we study the spatio-temporal disaggregation problem — to break down the aggregated urban data from a coarse, irregular spatial partitioning to a finer, irregular partitioning. The disaggregation problem has been studied in different domains. Spatial disaggregation methods in the demography literature tend to rely on heuristics: areal weighting, which assumes all variables are constant per unit area [35], and dasymetric mapping [35], which assumes sufficient auxiliary data is available to predict the target variable using linear methods. Disaggregation has also been considered in the machine learning literature, outside of the spatiotemporal context, where the goal is to learn individual-level models given group-level aggregated data [303, 175]. However, these approaches explicitly seek a model where the target variable at the individual level is determined *only* by its parent group’s features as opposed to learning from global patterns across groups. In a spatiotemporal context, any region may influence any other, even remotely: traffic on the bridge can increase wait times in midtown, for instance.

In this study, we evaluate neural architectures for the disaggregation problem, and demonstrate the utility of the disaggregated data (See Figure 3.1). We proposed an overarching model, Structurally-Aware Recurrent Network (SARN), which integrates structurally-aware spatial attention layers (SASA) into the Gated Recurrent Unit (GRU) model. The SASA layers capture complex spatial interactions among regions, while the gated recurrent module captures the temporal dependencies. Specifically, each SASA layer calculates two types of attention — global and structural. Global attention attends each coarse region to all fine regions, facilitating a comprehensive, macro-level geographic interactions. To enhance learning efficiency and effectiveness, we leverage the structural property exhibiting between geographic levels, *containment* — each fine level region is entirely contained in a coarse level region. This phenomenon is practically seen in political boundaries. Exploiting this structural property, we create binary containment maps to be used in structural attention calculation, promoting the generation of coherent and consistent values across varying geographic levels. We conduct three experiments targeting various disaggregation levels

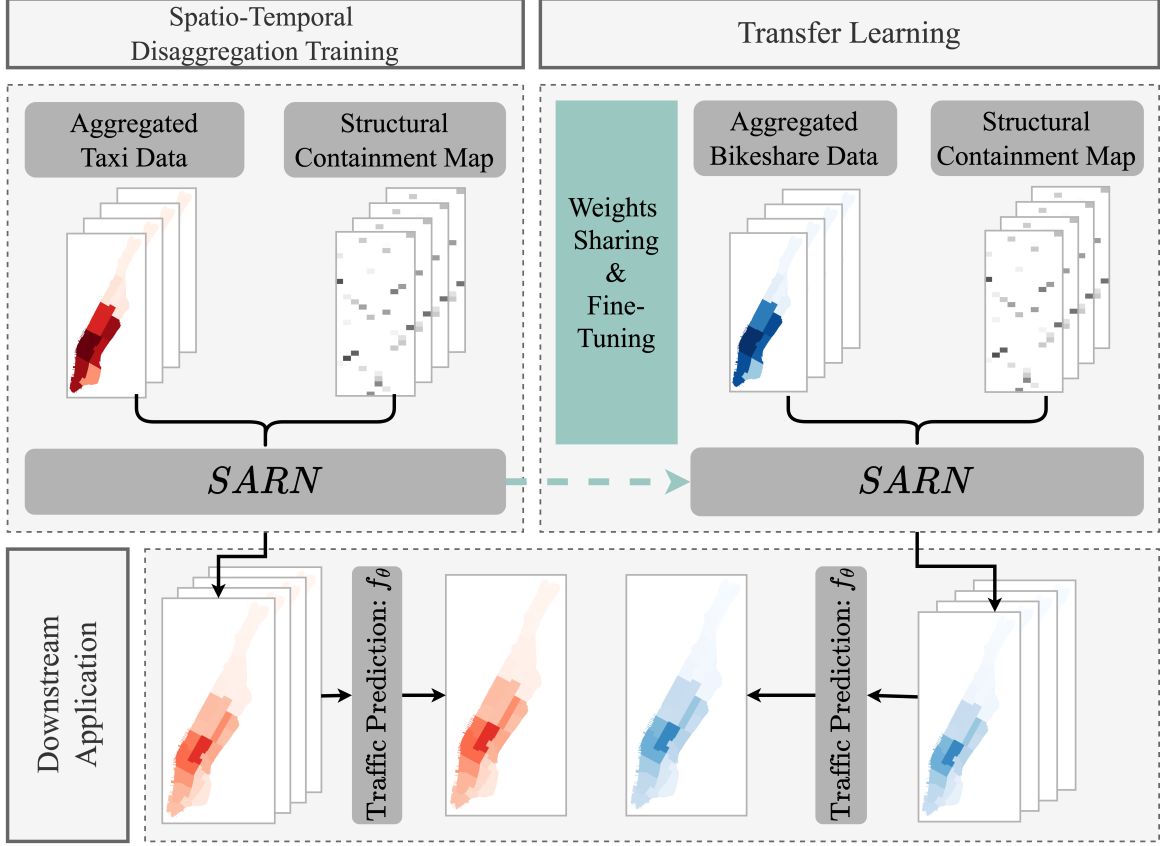


Figure 3.1: Disaggregation problem setting and its utilities. Our proposed Structurally-Aware Recurrent Network (SARN) is trained with coarse, aggregated data and structural containment maps. The trained model can be utilized in downstream applications, such as traffic prediction. It also can be adopted to disaggregate another variable through weight-sharing and fine-tuning with only a limited amount of data.

with two mobility datasets from NYC. Our results demonstrate that SARN significantly enhances disaggregation performance and converges faster than baselines.

In addition to our primary disaggregation findings, we conduct three supplementary experiments to enhance the practicality of the disaggregation task and demonstrate the utility of our model: **(1) Transfer Learning.** We relax the assumption that temporal historical data is always available and explore pre-training on variable A , then fine-tuning on vari-

able B . Our results demonstrate the remarkable effectiveness of transfer learning, yielding competitive outcomes compared with specialized models, with only a minimal number of samples. **(2) Downstream Application.** The disaggregated data can be valuable for downstream applications. We use traffic demand prediction as an example, as accurate traffic demand prediction is essential for cities to allocate transportation resources effectively, especially in fine regions (e.g., city blocks), potentially reducing traffic congestion. We train several models with the disaggregated taxi data and compare their performance to those trained with ground truth data. Our results show remarkable authenticity of the disaggregated data compared to the real data. **(3) Individual Points Synthesis.** We quantitatively show that our disaggregated values have a similar distribution to the ground truth values over a set of regions, via mutual information and Kullback–Leibler divergence statistics. Then we qualitatively demonstrate that a simple sampling scheme can be used to generate a realistic individual-level points within the finer regions.

Overall, this chapter makes four primary contributions toward improving urban data granularity and improving its utility for downstream applications:

- We propose the Structurally-Aware Recurrent Network (SARN), designed to disaggregate spatio-temporal data. The structurally-aware spatial attention (SASA) layers model spatial interactions, while the gated recurrent module captures temporal dependencies. Within each SASA layer, global attention is calculated to enable macro-level interactions between levels. By employing containment maps, structural attention is calculated to produce coherent and consistent values between different levels.
- We test three disaggregation resolutions and two mobility datasets. On NYC taxi data, SARN achieves a 5% improvement over the best neural models and a 40.1% improvement over the best classical methods on average. For NYC bikeshare data, the average improvements are 1.2% and 13.6%, respectively. Furthermore, SARN converges faster than other neural models, showing better training efficiency.
- We consider a transfer learning scenario, acknowledging that sufficient training data may not always be available to train an effective neural model for every city variable.

Our results demonstrate that, within the same city, parameters learned from one variable can be used to make predictions on another variable through rapid fine-tuning (1-2 hours versus 12 hours of pre-training), achieving competitive performance.

- We demonstrate the utility of disaggregated data in downstream traffic demand prediction. The average prediction difference between models trained with disaggregated values and ground truth values is trivial (0.78%), indicating that the disaggregated data closely matches the ground truth values. Additionally, we synthesize individual-level data via sampling and show that synthesized points have similar distributions to the true values, with high mutual information (0.736) and low KL-divergence (0.058).

3.2 Related Work

3.2.1 Spatial Disaggregation

Spatial disaggregation is frequently used in demography to infer individual records from spatially aggregated regions such as census data [198, 260], frequently applied in population analysis. Dynamic data on human population distribution at high resolution is important for a wide spectrum of activities and real-life applications, such as urban applications of business locations, transportation planning, city service management etc. [318] Other applications of spatial disaggregation include disease mapping [6] and vaccination coverage [242].

A popular heuristic is areal weighting (AW), which assumes the value per unit area is constant [70]. This method is common in practice due to its simplicity, general reliability, and ability to be computed from a single variable. Pycnophylactic interpolation (PI) [233] works similarly to areal weighting, but distributes the values from source regions to target regions iteratively, considering a weighted average of its nearest neighbors. Dasymetric mapping/interpolation/weighting (DM) approach is a family of methods that use the relationships between multiple variables to improve the estimated distribution. Unlike areal weighting, dasymetric mapping uses auxiliary information to generate a weighted distribution of source values onto the target regions. Recent methods incorporate machine learning approaches to model the relationships between auxiliary data and source values and determine the weights for disaggregation [198, 177, 176]. Sources of auxiliary information

include satellite images [224], 3D building information [198], mobile phone usage data [45], etc. Despite their popularity in spatial disaggregation, there are several potential limitations of them [35]. They are wholly dependent on auxiliary variables, which are not always available (e.g., remote sensing images or land type data in population disaggregation). Additionally, the performance depends on the correlation between auxiliary data and source values, which may be low due to lack of relevant datasets or noisy measurements.

Deep learning methods have also been directly applied to the disaggregation problem in recent years. For example, Zong et al. [318] proposed the DeepDPM model, which uses a super-resolution CNN to analyze both spatial and temporal patterns using coarse data and point-of-interest information. In the urban computing domain, several recent studies have focused on generating fine-grained traffic flow from coarse-grained data using super-resolution techniques. Ouyang et al. [190] introduced the UrbanFM model, which employs a feature extraction module and a distributional upsampling module to infer fine-grained crowd flow from coarse observations. Liu et al. [155] proposed the RATFM model, which leverages prior knowledge of road networks to learn the road-aware spatial distribution of fine-grained traffic flow. Both studies use super-resolution techniques, but their problem setting differs slightly from ours. Their coarse observations are at the pixel level, while our coarse values are at predefined geopolitical region levels. In an urban image, each region may cover numerous pixels that share the same value.

3.2.2 Learning From Aggregated Observations

In addition to the spatial disaggregation methods, it is also possible to learn individual level probabilistic models from aggregated data, without first decomposing aggregated data into individual instances. The problem is generally referred to as learning from aggregated observations, where supervision labels are given to sets of instances instead of individual instances, while the goal is still to make inferences on unseen individuals [303, 219, 131]. Zhang et al. [303] present a general probabilistic framework capable of accommodating a variety of aggregate observations for various tasks, such as classification and regression. Ma et al. [165] studied the problem of learning nonlinear dynamics from aggregated data, where

individual trajectory data is not available. Wei et al. [264] studies a more specific problem — classification from aggregate observations (CFAO), where the supervision is provided to groups of instances, instead of individual instances.

3.3 Datasets

In this study, we use two representative mobility datasets in our study — NYC taxi data and NYC bikeshare data. Each dataset contains individual-level records.

- **NYC Taxi Data:** NYC taxi trip data is collected from NYC Open Data portal¹ from 01/01/2016 to 06/30/2016. The raw data are presented in tabular format. Each record summarizes the information for one single taxi trip, which contains the longitude and latitude of the location of the trip start.
- **NYC Bikeshare Data:** NYC bikeshare data is collected from NYC DOT² 01/01/2021 to 12/31/2021. Similar to the taxi data, the raw data are presented in tabular format. Each data point represents one bike trip, including the longitude and latitude of the location where the bike was unlocked. The dataset spans the entire year, allowing the inspection of seasonal variations.

These datasets represent two examples of mobility events in the same city (taxi and bikeshare in NYC). This collection enables study of variance of our method (same city, same application domain, but different variable). Moreover, NYC taxi and bikeshare datasets afford transfer learning studies (e.g., training on taxi data and evaluating on bikeshare data) to address the situation where available data may be limited. We focus exclusively on Manhattan, employing four geographic resolutions from 2010: Public Use Microdata Areas (PUMA), Neighborhood Tabulation Areas (NTA), Census Tracts (TRACT), and Census Blocks (BLOCK), with boundaries from the NYC Open Data portal (see Figure 3.2).

Given a dataset of individual events (lat, lon, time), we construct raster representations (for CNN models) and vector (for other neural models) representations of spatiotemporal

¹<https://opendata.cityofnewyork.us/data/>

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

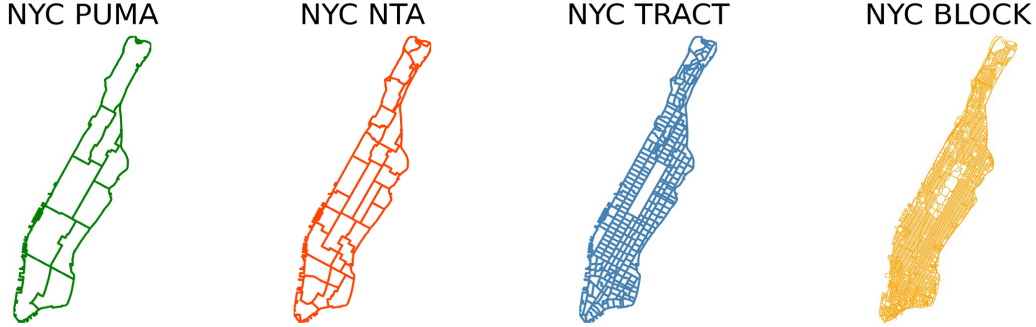


Figure 3.2: Four distinct geographic aggregation levels for the same NYC area. PUMA regions are the largest and most coarse. BLOCK regions are smallest and finest.

event data as follows: **(1) Vector Representation.** We aggregate the event counts by hour and region for each of the four geographic levels. For example, there are 10 Public Use Microdata Areas (PUMA); a single hour of data is represented as simply a 10-element vector. The order of elements is arbitrary. More generally, the dataset is represented as $X \in \mathbb{R}^{N \times d}$, where N is the total number of hours, and d is the number of regions at a given geographic level (e.g., $d = 10$ for PUMA; $d=32$ for NTA). **(2) Raster Representation.** We first create a 128×256 grid and plot the N regions at each geographic level. Each region has a corresponding value from the vector input — $X_n, n \in \{1, 2, \dots, N\}$, and the number of pixels located within the region — $C_n, n \in \{1, 2, \dots, N\}$. Then each pixel in each region takes the value of $\frac{X_n}{C_n}$. Pixels shared by more than one region are assigned to whichever region is smaller to avoid ambiguity. The cost is that we lose some information during translation since the raster representation does not represent region boundaries precisely.

3.4 Models

In §3.4.1 we formally formulate the disaggregation problem. In §3.4.2, we introduce the structural property and containment maps that connect different levels of the aggregation hierarchy. In §3.4.3, we detail the architecture of the structurally-aware spatial attention (SASA) layer, including the calculation of global and structural attention using the containment maps. In §3.4.4, we present our proposed SARN model, which integrates SASA

layers into the GRU layer. The model architecture is visualized in Figure 3.4

3.4.1 Notations & Problem Formulation

We adopt the following notations to be used throughout this study:

- g^i for $i \in \{0, 1, \dots, G\}$ denotes a geographic aggregation level. The levels are labeled in ascending order in terms of number of regions, i.e., the lowest resolution is assigned 0 and the highest resolution is assigned G . In our experiments, the input is always the lowest resolution, e.g. PUMA. Therefore, it is assigned with the level g^0 . Level g^G has the largest number of regions, which is BLOCK. Level $(g^1, g^2, \dots, g^{G-1})$ are intermediate levels, in incremental order.
- $|g^i|$ denotes the number of regions at geographic level g^i . $|g^G|$ denotes the number of regions at the BLOCK level, which we use to normalize the loss terms. That is, the loss for all levels is expressed in average error per city block to allow direct comparisons across different geographic levels.
- $X^i \in \mathbb{R}^{N \times |g^i|}$ denotes the ground truth $N \times |g^i|$ vector at geographic level g^i assuming N total hours. We include a subscript $(t) \in [1, N]$ to indicate a single timestep $X_t^i \in \mathbb{R}^{|g^i|}$. To indicate a range of timesteps t to $t + T$, we will write $X_{t:t+T}^i$.
- $\hat{X}^i \in \mathbb{R}^{N \times |g^i|}$ denotes the predicted data at geographic level g^i assuming N total hours. We will indicate slices of this vector using the same subscript notation as with ground truth.

The disaggregation problem is defined as — given a time series of values in the past T time steps at coarse level g^i , we learn a function f that disaggregates the data to a finer level g^j :

$$f([X_{t-(T-1)}^i, \dots, X_t^i]; \theta) \rightarrow [X_{t-(T-1)}^j, \dots, X_t^j], \forall j > i$$

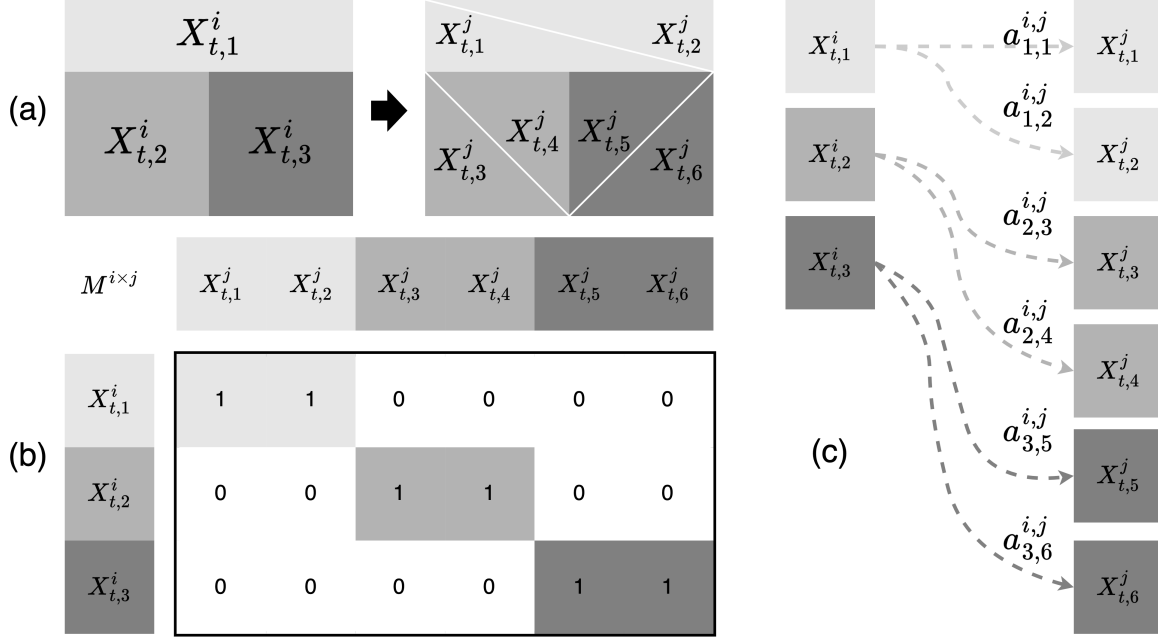


Figure 3.3: (a) Containment relationships between geographic level g^i and g^j . Each region from g^j is wholly contained in a region from g^i . (b) Binary containment map $M^{i,j}$ between geographic level g^i and g^j . (c) Structural-aware spatial attention is calculated among connected regions from two levels.

3.4.2 Structural Property — Containment

The geographic levels we use exhibit the structural property of *containment* — each region at level $g^i, i \geq 1$ is wholly contained in a region at level g^{i-1} such that regions are hierarchical. For example, one census tract comprises multiple census blocks. Assuming containment, we can identify a one-to-many map among the regions from any two geographic levels. Let $M^{i,j} \in \mathbb{R}^{|g^i| \times |g^j|}, \forall j > i$ be a $|g^i| \times |g^j|$ binary matrix representing the containment relationship between geographic levels g^i and g^j . A value of 1 at position $M^{i,j}[c, r]$ indicates that region r at level g^j is contained in region c at level g^i . Figure 3.3(a) shows an example where three regions from level g^i at a single timestamp t are split into six regions from level g^j . Figure 3.3(b) shows the corresponding binary containment map between the two levels.

3.4.3 Structurally-Aware Spatial Attention Layer

Spatio-temporal transformers have been applied to urban prediction tasks. The multi-head spatial attention layers in the transformer blocks can capture complex spatial relationships. In our disaggregation task, we also leverage the attention mechanism to extract spatial information. The spatial attention is calculated across two geographic levels with different spatial dimensions. We propose a structurally-aware spatial attention (SASA) layer, which consists of multiple structurally-aware spatial attention heads ($SASA_{head}$). Each head performs two attention calculations — a global attention and a structural attention.

Global Attention: given input $X_t^i \in \mathbb{R}^{1 \times |g^i|}$ from geographic level g^i , at a single timestamp t , the global attention score is calculated as:

$$\begin{aligned} Q &= X_t^i W_Q, \quad W_Q \in \mathbb{R}^{|g^i| \times |g^i|} \\ K &= X_t^j W_K, \quad W_K \in \mathbb{R}^{|g^j| \times |g^j|} \\ Att_g &= \text{Softmax} \left(Q^T K / \sqrt{|g^j|} \right) \in \mathbb{R}^{|g^i| \times |g^j|} \end{aligned}$$

The global attention enables each coarse region to attend to all finer regions, facilitating comprehensive interactions across different geographic levels. This macro-level interaction allows the model to capture broader dependencies between two geographic levels.

Structural Attention: the structural attention leverages the binary containment map $M^{i,j}$ to constrain attention only among connected regions across geographic levels (Figure 3.3(c)). The containment maps impose a constraint that requires the values of child regions to sum up to the value of their parent region. This ensures that the predicted values remain coherent and consistent across different geographic resolutions. By enforcing this hierarchical relationship, the model maintains the integrity of the spatial dependencies, allowing for accurate disaggregation results. Structural attention is computed as:

$$Att_s = \text{Softmax}(M^{i,j} \odot (Q^T K / \sqrt{|g^j|})) \in \mathbb{R}^{|g^i| \times |g^j|}$$

The global and structural attention outputs are added, multiplied with the value matrix, and go through a feed-forward projection layer:

$$H = FNN(V(Att_g + Att_s)) \in \mathbb{R}^{1 \times |g^j|}, V = X_t^j W_V$$

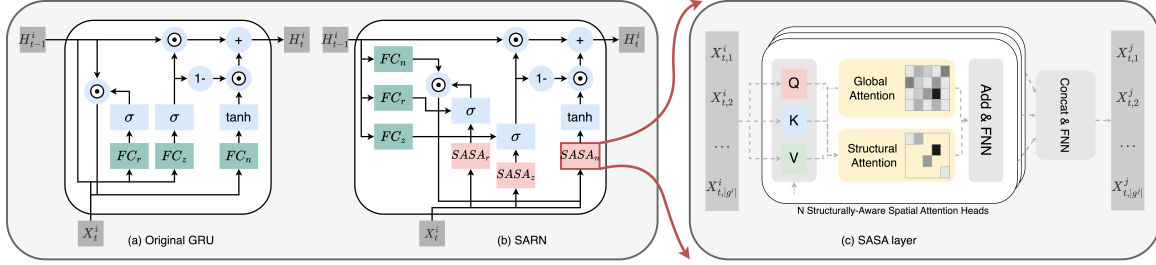


Figure 3.4: Model architecture of our proposed SARN. (a) Original architecture of the GRU model. Fully-connected layers are applied on input X^i . (b) SARN architecture. The fully-connected layers are replaced with our structurally-aware spatial attention (SASA) layers. (c) SASA layer. A SASA layer is composed with multiple spatial attention heads, each calculating global and structural attention across regions from two different geographic levels. The containment map is used to reflect the real-world structural hierarchy between geographic levels, and ensures the generation of coherent values across different resolutions.

$$V = X_t^i W_V, \quad W_V \in \mathbb{R}^{|g_i| \times |g_i|}$$

A structurally-aware spatial attention layer (SASA) is composed with N spatial attention heads, enhancing the model capacity:

$$SASA = FNN(\oplus(H_1, H_2, \dots, H_N))$$

$$H_i = SASA_{head,i}(Q, K, V)$$

Given T hours of temporal input $X_{t:t+T}^i$, we reshape the dimension $X_{t:t+T}^i \in \mathbb{R}^{T \times 1 \times |g^i|}$ and calculate spatial attention. In our disaggregation task, the input and output have different spatial dimensions. Therefore, residual connection is not applicable.

3.4.4 Structurally-Aware Recurrent Network

The spatial attention layer only captures spatial dependencies across different geographic levels. The temporal information, which is also critical for spatio-temporal prediction tasks, is not represented except as a set of independent samples. Recent research has increasingly employed spatio-temporal transformers for spatio-temporal traffic forecasting. The

temporal attention layers capture the temporal dependencies in the input sequences. Temporal attention layers are typically preferred to conventional recurrent models due to their ability to learn long-distance dependencies and their efficient implementations in parallel settings. Unlike recurrent models, where information encoded at each time step is retained solely for the subsequent step and quickly diminishes in influence, attention layers calculate attention scores across the entire sequence concurrently, thereby promoting shared sequential information. However, GRUs are still widely used in spatio-temporal prediction tasks [161, 29, 206, 255]. Additionally, our setting involves relatively short temporal lengths, and we empirically find that GRUs outperform temporal transformer-based models.

The comparison of SARN and the original GRU model is visualized in Figure 3.4(a) and (b). Given input data $X_{t:t+T}^i$ from geographic level g^i , the original GRU computes the hidden state at each timestep t :

$$\begin{aligned} r_t &= \sigma(X_t^i W_{xr} + h_{t-1} W_{hr} + b_r) \\ z_t &= \sigma(X_t^i W_{xz} + h_{t-1} W_{hz} + b_z) \\ n_t &= \tanh(X_t^i W_{xn} + r_t \odot (h_{t-1} W_{hn}) + b_n) \\ h_t &= z_t \odot h_{t-1} + (1 - z_t) \odot n_t \end{aligned}$$

where $X_t^i \in \mathbb{R}^{1 \times |g^i|}$ is the input at time t , h_{t-1} is the previous hidden state (initially at time 0), and h_t is the updated hidden state, which serves as input for the subsequent timestep. In the original GRU model, matrix multiplications $X_t^i W_{xr}$, $X_t^i W_{xz}$, $X_t^i W_{xn}$ transform the spatial dimension from coarse $|g^i|$ to finer $|g^j|$. In our SARN model, we replace these linear transformations with SASA layers, enabling simultaneous modeling of spatial and temporal dependencies. The updated gate computations are:

$$\begin{aligned} r_t &= \sigma(\text{SASA}_{xr} + h_{t-1} W_{hr} + b_r) \\ z_t &= \sigma(\text{SASA}_{xz} + h_{t-1} W_{hz} + b_z) \\ n_t &= \tanh(\text{SASA}_{xn} + r_t \odot (h_{t-1} W_{hn}) + b_n) \\ h_t &= z_t \odot h_{t-1} + (1 - z_t) \odot n_t \end{aligned}$$

3.4.5 Loss

We calculate ℓ_1 loss at the target disaggregation level. However, since ℓ_1 loss divides by the size of the vector, meaning that the ℓ_1 loss will have different units at each aggregation level. For example, the ℓ_1 loss at the TRACT level is in units of “count per TRACT region” while the ℓ_1 loss at the BLOCK level is in units of “count per BLOCK region”, leaving the two losses incomparable. Therefore, to meaningfully compare the losses at different target disaggregation level, we normalize all loss terms to be count-per-region at a selected aggregation level. In our experiments, we normalize to the BLOCK level to assist with interpretation: we consider taxi rides per city block. For prediction at a single timestamp t , at level g^i : $\hat{X}_t^i \in \mathbb{R}^{1 \times |g^i|}$, the loss is calculated as:

$$\ell_1^i = \frac{1}{|g^i|} \sum |\hat{X}_t^i - X_t^i|$$

3.5 Experiments

In this section, we describe the baseline models to compare against for the disaggregation task in 3.5.1. Additionally, we describe the setup for the supplementary studies of downstream application (3.5.2), transfer learning (3.5.3) and point synthesis (3.5.4).

3.5.1 Baseline Models

As our baseline, we test several heuristic-based disaggregation methods used commonly in the literature and in practice, as well as more advanced neural approaches:

- **Constant Weighting (CW)**: The event count for a region in the output is assumed to be $\frac{1}{N}$ of the count in the containing region in the input, where N is the number of output regions contained in the input region. That is, this heuristic assumes the events are uniformly distributed, regardless of size and shape of the region.
- **Areal Weighting (AW)**: The event count for an output region r contained in an input region c is assumed to be proportional to the fractional area of r in c . That is, $|r| = \frac{A(r)|c|}{A(c)}$, where $A(r)$ is the area of region r and $|c|$ is the event count in region c . That is, this heuristic assumes events are uniform per unit area.

- **Historical Ratios (HR)**: The event count of an output region r is assumed to be a fixed proportion of the count of an input region c . The fixed proportion is computed based on ALL historical timestamps in the training set.
- **Feedforward Neural Network (FNN)**: The event count for an output region r is assumed to be a (non-linear) function of all input regions, not just the containing region. We apply two linear layers connected with a ReLU activation function. The first layer has the number of PUMA regions ($d=10$) as the input dimension and the number of fine regions as the output dimension (e.g., $d=32$ for NTA). In the second layer, the input and output dimensions are equal and set to the number of fine regions.
- **Convolutional Neural Network (CNN)**: The event count at each output region r is extracted from the raster representation at the output aggregation level (e.g., city blocks). That raster representation in turn is learned from the raster representation of the input aggregation level (e.g., PUMA). This approach models the non-linear relationships between regions, while emphasizing local effects through convolutional layers. We use the UNet architecture [208], which is frequently employed in computer vision tasks. Please see [76] for model details.
- **Gated Recurrent Unit (GRU)**: Where FNN only models the spatial connections (by assuming that each timestep is an independent sample), GRU can capture temporal patterns. We chose $T=5$ as the temporal sequence length across all our experiments based on the study [76]. We use two GRU layers with the same input and output dimensions as FNN models.
- **UrbanFM (UFM)[190]**: a disaggregation model that generates fine traffic flow data from coarse, aggregated data using a super-resolution technique. It adopts grid representations of traffic flow, which is similar to the CNN baseline. We set the input PUMA image to be 64×32 , and slightly modified the model architecture.³.

³Their coarse observations are at pixel level, while our coarse values are at predefined geopolitical regions. When we process rasterized image inputs, pixels lying in the same region share the same value. Given

- **Spatio-Temporal Transformer (ST-Transformer)**: a model consisting a spatial and a temporal transformer. The spatial transformer shares the same spatial attention layer as described above, with global attention only. The temporal transformer follows the set up in [150] — given the output from the spatial transformer $\hat{X}_{t:t+T}^i \in \mathbb{R}^{T \times |g^j| \times 1}$, we add hour-of-day and day-of-week index to the features, and generate a feature embedding. Then regular multi-head self-attention mechanism is applied. We chose five timesteps ($T=5$) as the temporal sequence length.

We test three disaggregation tasks — from PUMA to NTA, TRACT and BLOCK level respectively. For all models and experiments, batch size, learning rate, early stop tolerance are set to 32, 1e-4, and 50 respectively. For STT model, the feature embedding size, number of temporal attention head, and number of temporal transformer layer are set to 64, 4, and 4 respectively. For our SARN, we only use one GRU layer, and each SASA layer uses 4 $SASA_{head}$. We use 1xNVIDIA T4 GPU, 16 GB RAM, x4 vCPUs, for each task.

3.5.2 Downstream Application

To demonstrate the high quality and utility of the disaggregated data, we use traffic demand prediction as the downstream application task. Traffic demand prediction involves forecasting the number of vehicles required in a region within a specific time frame. Accurate traffic demand prediction enables efficient allocation of transportation resources, reducing mismatches between supply and demand, alleviating traffic congestion, and enhancing mobility service experiences. Coarse-level predictions (e.g., PUMA) are less useful as they tend to miss fine-grained human mobility patterns. Therefore, using disaggregated values at finer resolutions is more beneficial and insightful. The specific task is to predict the next hour’s taxi demand given the previous twelve hours of traffic data.

We use the same NYC taxi dataset used in the main disaggregation problem. The NYC taxi data contains the number of taxis requested in each region for each hour, representing hourly taxi demand. We use the validation ($T=744$) and test ($T=720$) splits of the taxi data

this preprocessing, the N_2 normalization and distribution recovery layer can no longer be applied in the original UrbanFM model, which might negatively impact the results.

for traffic demand prediction, as they are not used in the disaggregation model training. We first disaggregate taxi demand values in the validation split from PUMA to NTA, TRACT, and BLOCK levels using the trained SARN models. We then train several traffic prediction models (LSTM[94], AGCRN[8], STSGCN[222]) with both the disaggregated data and the original ground truth data at each level. Finally, Mean Absolute Error (MAE) at each geographic level, for each model, is calculated across predictions on 708 hours from 06/01/2016 to 06/30/2016 in the test split.

3.5.3 *Transfer Learning*

As mentioned in the introduction, there are urban settings where the availability and quality of fine-grained data are limited for model training, posing a barrier to implement our model. This data availability issue is not just a limitation to our model, but across all neural models that require sufficient training; it is also universal to all urban computing tasks where neural models excel. In these cases, we can consider whether a model pre-trained on one variable can be used to make decent predictions for another variable.

We can use our models in this transfer learning setting by training on NYC taxi data, then fine-tuning on a limited amount of NYC bikeshare data. This approach can significantly increase the value of open data, by allowing one dataset to be used in different applications. The idea is reasonable, given the interconnectedness of city dynamics [98]: everything depends on everything else. Moreover, the main signals being learned by any model are influenced by the built environment, human activity, and population density patterns, all of which tend to be exhibited in any spatio-temporal dataset in the same domain.

For transfer learning experiments, we use the same NYC taxi data and NYC bikeshare data used in the main disaggregation problem. We first train the disaggregation models using the NYC taxi data. Then, these trained models are transferred and evaluated on the bikeshare dataset. We fine-tune the disaggregation models with varying amounts of bikeshare data — one day (24 hours), three days (72 hours), one week (168 hours), two weeks (336 hours), and one month (720 hours). The fine-tuning process follows the same setup as the training process, with a batch size of 64, a learning rate of $1e-4$, and a tolerance

of 50 epochs. Then the fine-tuned models are tested on the test split of the bikeshare data.

3.5.4 Event Data Synthesis

While the BLOCK aggregation level is high-resolution (smaller than census tracts for example), they remain spatially aggregated and cannot be directly utilized by applications requiring individual point data. Additionally, if we can synthesize individual taxi rides for NYC Taxi data after June 2016 given the aggregated taxi zone data, it would significantly enhance the utility of the datasets. A simple approach to synthesize points is to learn a high-resolution aggregation at BLOCK level, then randomly sample within these regions to generate individual points, considering the physical built environment (e.g., building, hotel etc.). If a region has k events, we can sample k events within its boundaries. Quality control is important for the synthesized data. Quantitatively, we can compare the distributions of real and disaggregated values across a set of BLOCK regions. We consider two statistics to measure similarities — Mutual Information (MI) and KL-divergence. MI measures the probabilistic dependency between two variables, where a higher value indicates greater shared information between the variables. KL divergence is a measure of entropy increase due to the use of an approximation distribution to the true distribution. A lower KL divergence value signifies a better match between the true distribution and our approximation. We also assess the distribution similarity qualitatively through visualization.

3.6 Results

3.6.1 All Neural Methods Outperform Heuristics

Table 3.1 presents the disaggregation results using both heuristic and neural methods. The error unit is normalized to be $\text{count}/(\text{block} \times \text{hour})$, so that errors at different target disaggregation levels can be meaningfully compared. We observe that: **(1)** As the target disaggregation resolution increases, the error increases across all methods, as expected. Predicting city block dynamics from large regional aggregates is a difficult problem. **(2)** Popular heuristic-based disaggregation methods, such as CW and AW, are non-competitive across all resolutions and datasets. While the underlying heuristics are reasonable, they

Dataset	Levels	CW	AW	HR	UFM	CNN	FNN	GRU	STT	SARN
NYC Taxi	NTA	1.8295	1.4072	0.4652	0.3989	0.3601	0.2300	<u>0.2255</u>	0.2729	0.2073*
	TRACT	1.8416	2.0651	0.9650	0.8049	0.7324	0.5673	<u>0.5631</u>	0.6473	0.5443*
	BLOCK	3.0947	2.9705	1.7609	1.8576	1.5306	1.4515	<u>1.4278</u>	1.4305	1.3763*
NYC Bikeshare	NTA	0.1678	0.1212	0.0619	0.0834	0.0587	0.0497	<u>0.0478</u>	0.0601	0.0473*
	TRACT	0.2973	0.2945	0.1650	0.1787	0.1591	<u>0.1497</u>	0.1505	0.1584	0.1483*
	BLOCK	0.7882	0.7574	0.2488	0.3726	0.2481	0.2492	<u>0.2341</u>	0.2513	0.2316*

Table 3.1: Disaggregation results from traditional and neural-based methods. The unit of the error is count/(block * hour), for comparison purpose across different disaggregation levels. **Bold** and underline numbers indicate the best and second result in the corresponding disaggregation task respectively. * indicates statistical significance.

cannot well capture the urban traffic dynamics. The distribution of taxi rides and bike-share trips follows human activity patterns influenced by temporal effects and non-linear interactions between regions, which simple heuristics cannot adequately capture. **(3)** HR significantly outperforms other heuristics (53.5% average increase over AW), indicating that temporal patterns are generally predictive. However, HR still overlooks the non-linear temporal and spatial variations present in the data, limiting its accuracy. **(4)** Neural-based methods outperform all common heuristics in nearly all disaggregation tasks. GRU models capture additional temporal dependencies, thus outperforming models that solely rely on spatial information. GRU emerges as the best-performing baseline model (2.3% average increase over FNN). **(5)** UrbanFM is outperformed by CNN models. We attribute the huge performance gap to the difference in setting of two disaggregation problems. The N_2 normalization and distribution recovery layer proposed for the original problem can no longer be applied in our setting, thus negatively impacting the results. Overall, the results show that neural models, especially GRU, outperform heuristic-based approaches, and temporal information helps.

3.6.2 SARN Significantly Improves Neural Results

From Table 3.1, we observe that our proposed SARN has the best performance across all six tasks! With taxi data, it delivers a 5.0% improvement over the best neural models (GRU) on average and a 40.1% improvement over the best classical methods (HR) on average. With bikeshare data, we see relatively smaller but still notable improvements of 1.2% and 13.6%, respectively. The largest improvement is observed when disaggregating from PUMA to NTA with NYC Taxi data, where SARN reduces the average hourly taxi count error in the entire Manhattan region by 301 compared to the GRU model! For statistical significance tests, we use bootstrap sampling for 10,000 iterations [114, 221]: On each iteration, we sample the predictions, then calculate error and compute the proportion that are better than the baseline to provide an empirical estimate of the p-value. We use 0.05 as the significance level. With both the taxi and bikeshare dataset, SARN consistently outperforms GRU with statistically significant improvement.

We notice that STT models generally perform poorly on both datasets, even worse than FNN models. The performance gap is noticeably large on simpler tasks, from PUMA to NTA. We conjecture two reasons: **(1)** STT has overly complicated architectures for simpler disaggregation tasks. STT has over 150K parameters when disaggregating PUMA to NTA, while GRU only has 10K and SARN has 35K. Models with more parameters do not necessarily enhance performance, but could possibly contain noisy signals that hurt the results. **(2)** STT shares the same spatial transformer layer as SARN. The only difference is the temporal module. Temporal attention is not effective and is incapable of producing consistent results compared with recurrent modeling unit in the disaggregation task. Although attention-based solutions are increasingly popular due to their success, we find that relatively short sequences and smaller capacity models can still motivate recurrent models.

Additionally, we observe that SARN converges faster than other neural model. Figure 3.5 shows the convergence plots of neural models on two datasets. We notice that other than the disaggregation from PUMA to TRACT on NYC Bikeshare dataset, SARN models always start with, and maintain lower validation error throughout training stage, thus converging faster than other models. Potential explanations are — **(1)** given the same recurrent

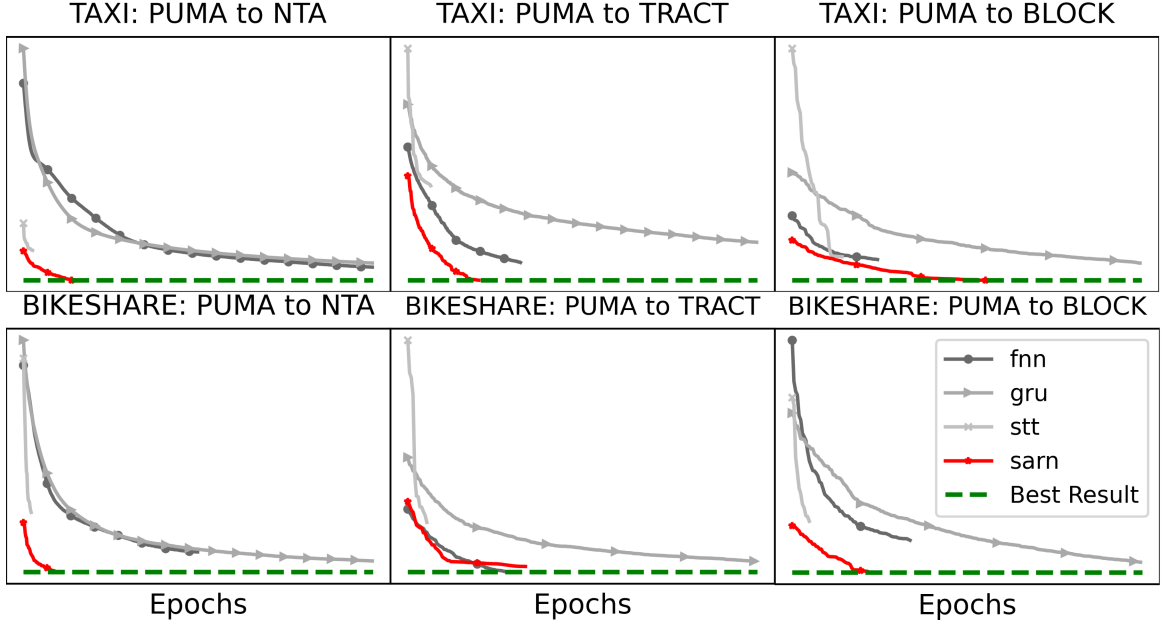


Figure 3.5: Convergence plots of neural models. Top three plots: show NYC Taxi dataset. Bottom three plots: shows NYC Bikeshare dataset. Red line represents our proposed SARN model, and green line indicates the best results. In five out of six tasks, our SARN converged faster than other neural methods.

module, SARN converges faster than GRU, indicating the power of spatial transformer layer.

(2) The spatial attention layers in SARN models are aware of the spatial structures from the containment maps, which provide “hints” for training, steering the learning toward the correct solutions (which always exhibit the containment property), thus converging faster.

3.6.3 Disaggregated Data is Authentic and Benefits Downstream Application

Table 3.2 presents the next-hour traffic demand prediction results using disaggregated data from SARN at three geographic levels. These results are compared to those from models trained with ground truth data. We observe that, all models trained with disaggregated data exhibit performance that is very close to those trained with ground truth data, with minor and sometimes negligible differences. In some cases, the models perform even better with disaggregated data (e.g., NTA with AGCRN; TRACT with AGCRN; BLOCK with

Levels	Forecasting Models	SARN	Ground Truth	% Difference
NTA	LSTM	184.665	184.035	0.03
	AGCRN	124.911	137.625	-9.24
	STSGCN	50.441	50.134	0.61
TRACT	LSTM	23.945	23.973	-0.11
	AGCRN	9.764	9.808	-0.45
	STSGCN	10.586	10.145	4.35
BLOCK	LSTM	2.328	2.393	-2.72
	AGCRN	1.648	1.452	13.47
	STSGCN	-	-	-
Average % Diff				0.78

Table 3.2: Traffic demand prediction results at different geographic level. SARN column shows the results from models trained with the disaggregated data from SARN model, while Ground Truth columns shows the results from models trained with the ground truth value. $\%Difference = \frac{SARN - GroundTruth}{GroundTruth}$. Negative % Diff means better prediction results with SARN. STSGCN results at the BLOCK level is not reported due to out-of-memory issue.

LSTM). Overall, the percentage difference is only 0.78% on average, which is very trivial. These findings indicate that the disaggregated values produced by the SARN model are authentic, of high quality, and closely mimic the distribution of the ground truth data. This has significant real-world implications. Disintegrating aggregated data into finer resolutions and then forecasting traffic demand can capture hotspots and dynamic human mobility patterns that coarse, aggregated data cannot provide.

3.6.4 Transfer Learning is Highly Effective

Figure 3.6 shows the transfer learning results with different amount of bikeshare data for fine-tuning: adding more fine-tuning samples enhances disaggregation performance. Surprisingly, fine-tuning with only one month of data achieves results comparable with the specialized model trained from scratch with bikeshare data. Additionally, even the performance slightly dropped, the benefit gain in saving computational costs is significant — fine-tuning

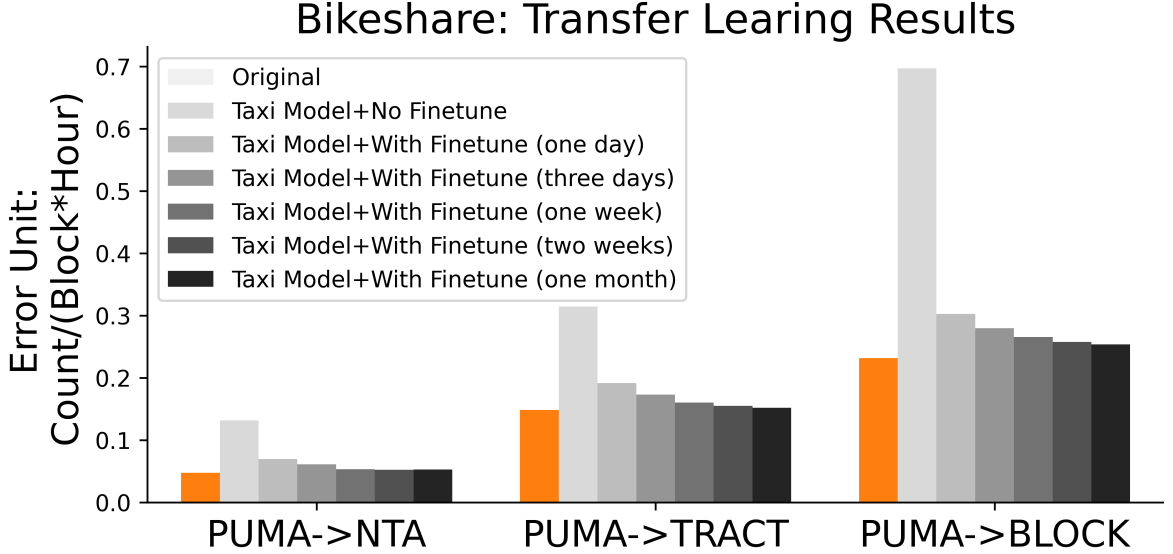


Figure 3.6: Transfer learning results on bikeshare dataset. The models are pre-trained with taxi data. The orange bar indicates the best performance for each disaggregation task.

a pre-trained model greatly reduces the time costs (1~2 hours v.s. 12 hours). This result has significant implications for practical settings: Training data is not uniformly available for all variables due to technical, privacy, and political effects. But because cities are complex systems, all variables are interdependent [98]. This result shows that pre-training can capture city dynamics resulting from the built environment and human activity, and that those signals tend to be predictive for multiple variables. We envision developing a core model pre-trained with ample historical records, then fine-tuning for different tasks in the same city, analogous to a language model being fine-tuned for more specialized tasks.

3.6.5 Realistic Synthesis of Individual Events

To generate individual point data, we use SARN to disaggregate from PUMA to BLOCK with NYC taxi data. Then, we measure the mutual information and KL divergence between the distribution of disaggregated values and the original distribution in the same time period and region set. Then, for each BLOCK region with k taxi counts, we randomly draw k points within the region. We work with a set of block regions in Manhattan as shown in Figure

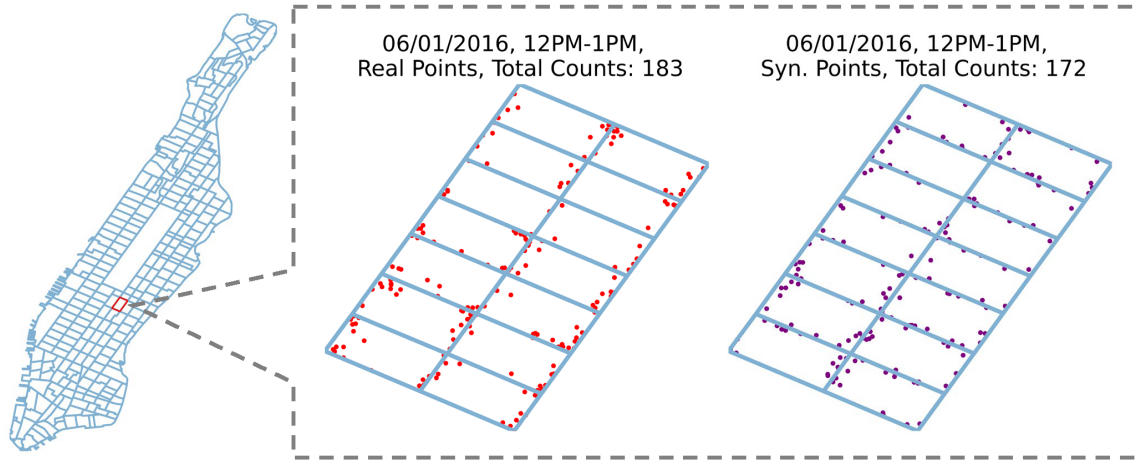


Figure 3.7: Synthetic individual-level taxi points. (Left) Randomly selected single tract region, containing multiple block regions. (Right) The distributions of real and synthesized taxis ride over the block regions are qualitatively similar.

3.7. We draw samples from this set of regions for the on 12PM-1PM, 06/01/2016.

The normalized MI score is 0.7360, greater than 0.5, indicating that two distributions share more than half their information. The KL-divergence is 0.0582. The low KL divergence indicates a reasonable approximation of the ground truth distribution using our disaggregated values. Qualitatively, Figure 3.7 depicts the distributions of both real and synthesized taxi location points within the region set from 12PM to 1PM. The model generated slightly smaller numbers of taxi rides compared to the real counts, with differences of 11 rides. While these differences are present, they are not significant compared with all taxi counts within the whole regions. In terms of the spatial distributions, the synthesized points capture the major clusters observed in the real taxi rides for both time periods, such as the clusters in the bottom left and center. This suggests that the synthesized points closely approximate the general patterns and trends present in the real data. On the other hand, the presence of discrepancies between the real and synthesized data points can have benefits, particularly in terms of privacy protection. As highlighted in the introduction, individual privacy is a significant motivator for *not* releasing individual-level data. A closely matching

distribution could be used as the basis of a privacy attack. For instance, clusters within very small areas may reveal the address of frequent riders. We consider our methods as providing better control over the trade-offs between faithful distribution and privacy (though without formal privacy guarantees, which are atypical for transportation data anyway).

3.7 Summary

This chapter improves urban data quality along the dimension of granularity by disaggregating coarse spatio-temporal observations into finer irregular regions. It introduces the Structurally-Aware Recurrent Network (SARN), which combines structurally-aware spatial attention with recurrent temporal modeling so that global spatial interactions and cross-level containment structures can be learned jointly. Across multiple disaggregation tasks on NYC taxi and bikeshare data, SARN consistently outperforms both classical and neural baselines while also converging more quickly. The chapter further shows that the generated fine-resolution data is practically useful: it supports downstream traffic demand prediction with performance close to models trained on ground-truth fine-scale data, transfers effectively across urban variables through fine-tuning, and can be used to synthesize realistic individual-level event patterns. Overall, the results demonstrate that learned spatio-temporal disaggregation is an effective way to recover fine-grained urban information from privacy-preserving aggregates, substantially improving data granularity and expanding the practical value of urban open data for urban applications.

While the refinement of granularity empowers significant analytical power, it introduces potential risk of privacy re-identification that must be acknowledged. Although our method does not use individual records, the ability to synthesize realistic, fine-grained individual points could potentially be leveraged for privacy attacks. Highly accurate distributions in small census blocks may allow for the inference of sensitive mobility patterns or the identification of frequent riders' locations. Unlike differential privacy, our disaggregation technique provides an empirical approximation rather than a formal mathematical guarantee. Therefore, our technique should be deployed with caution, accompanied by clear communication of uncertainty, appropriate access controls, and careful evaluation of downstream use.

Chapter 4

INCREASING URBAN DATA COVERAGE THROUGH FEATURE ANNOTATION WITH VISION-LANGUAGE MODELS

Urban data quality depends not only on how accurate existing records are, but also on whether the digital record covers the full range of infrastructure that shapes mobility and accessibility. In practice, many transportation-relevant features in the built environment remain absent from urban datasets because manual annotation is costly and scalable supervised pipelines require labeled data that rarely exists for rare or specialized features. This chapter addresses coverage as a core dimension of urban data quality by studying zero-shot annotation of urban features from aerial imagery with vision-language models. It presents a modular annotation pipeline that combines class-agnostic segmentation, heuristic candidate filtering, and Set-of-Mark prompting so that a general-purpose vision-language model can identify target features from a constrained set of candidates. Experiments on stop lines and raised tables show that direct prompting is ineffective, whereas the proposed pipeline produces substantially better annotations without task-specific fine-tuning, demonstrating a practical path toward expanding the coverage of urban data. This chapter is based on our work published at ACM SIGSPATIAL 2024 [80].

4.1 Introduction

The inaccessible urban infrastructure reinforces systemic exclusion of people with disabilities and negatively impacts public health and overall quality of life for everyone [54]. As laws evolve to make the urban environment more accessible, the digital representation must also evolve to include not just roads and sidewalks, but the hundreds of features that enable safe and equitable mobility: curb ramps, walk signals, tactile paving, crosswalks, light poles, and many more [172]. For example, mobility applications that provide accessible pedestrian routes rely on accurate, comprehensive annotations of the dynamic urban infrastructure [15].

The gold standard for data collection involves in-person inspections, typically admin-

istered by local governments with the help of community volunteers or contractors [62]. Human annotations are high quality but are labor-intensive, costly, and difficult to scale. For instance, in 2017, the city of Seattle employed 14 people for their first-ever audit of 2,300 miles of sidewalks, costing \$400,000 and identifying 92,000 uplifts, 38,000 surface problems, and 20,000 obstructions [62]. Sensing platforms deployed to navigate the built environment can partially automate annotation via machine learning, but these initiatives are also expensive to scale [13]. Aerial imagery offers nearly complete coverage at low cost, as long as the computer vision models to extract the relevant features can be trained. In practice, model development tends to focus on the needs of the “typical” traveler, e.g., roads [21] and in a few cases sidewalks [304]. Expanding coverage to the long-tail of accessibility features is rarely viable due to the cost in obtaining training data.

Pre-trained vision-language models (VLMs) (e.g., [154, 200]), exhibit impressive multi-modal scene understanding ability that is suitable for identifying objects in diverse visual domains [216]. If these models could be used to bootstrap feature annotation in the built environment, a more complete digital record could be developed enabling equity and accessibility at scale [99]. However, these models are trained on text paired with images from the internet, such that common objects in everyday (US-centric) experiences tend to be dominant. For atypical scenes and images, including satellite images, as well as granular urban features, VLMs’ performance on them remains to be tested.

In this study, we consider using VLMs with aerial imagery as a source of low-cost, extensible annotations for arbitrary features in the built environment. By reducing the cost of producing annotations and expanding the coverage of diverse features, mobility applications can keep pace with an evolving building code and commensurate new construction, faithfully representing the current built environment. With this capability, we envision a larger roadmap to scale this annotation process to any feature in any region, dramatically lowering costs for data acquisition and therefore enable a broad class of community-focused, AI-based mobility applications that have previously been too specialized to justify the cost. The ability to make services for specialized needs as economically viable as services for typical needs is the foundation of equity.

To demonstrate the proof-of-concept, we develop a procedure borrowing techniques from

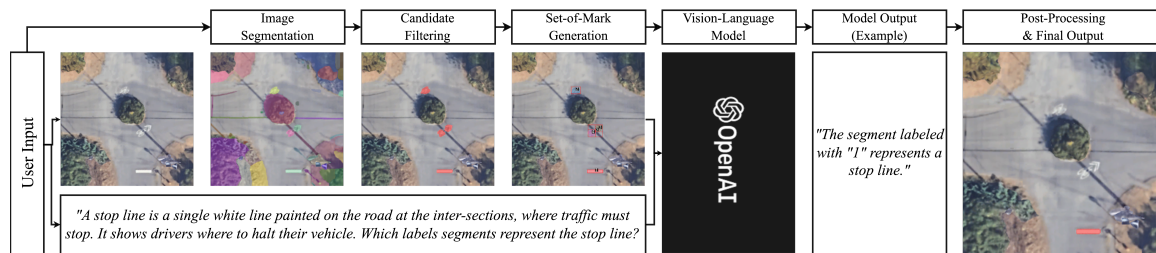


Figure 4.1: Pipeline of our proposed automated annotation process. Users input a pair of (satellite image, annotation guidance). The image will go through a set of processes including segmentation, filtering, and Set-of-Mark generation. Then the image and guidance will go through a vision-language model, the output of which is post-processed to produce the final annotation results. The procedure requires no fine-tuning, and can be applied on different features with minimal adjustment on the guidance.

visual grounding [281, 315] and visual prompting [217, 279] that generates bounding box annotations on satellite images based only on high-level descriptions of features. The procedure requires no fine-tuning or even labeled examples, demonstrating that closed, general-purpose models are potentially viable even in a zero-shot setting. Following the state-of-the-art (SOTA) visual prompting method [279], we select GPT-4o (the best VLM at the time of the work) as the VLM and SAM [121] as the general-purpose segmentation model. The SAM model segments a satellite image into a gallery of candidate objects, from which the GPT-4o model can select the appropriate feature of interest (see Figure 4.1 for details).

We test our procedure on two urban features: stop lines (white lines indicating where cars should stop at an intersection, which are important for safely deriving pedestrian routes through unmarked crossings) and raised tables (speed control features that can be used to estimate traffic flow and therefore pedestrian risk.) We consider multiple variants of this procedure, showing that presenting the segmented “options” to the model in different ways can improve performance. Quantitatively, we show that a simple direct prompting strategy fails dramatically at this task, while the proposed procedures achieve on average about 40% intersection-over-union (IoU) accuracy. The takeaway is that this problem is difficult, but not impossible. Qualitatively, we visualize several correct and incorrect annotations to

inform a discussion of the challenges and vision.

Overall, this chapter makes three primary contributions toward enhancing urban data coverage through feature annotation and supporting equitable, accessible transportation:

- We introduce a prompting procedure for generating annotations of diverse urban features from aerial imagery. By combining VLM, class-agnostic segmentation, lightweight candidate filtering, and natural-language feature guidance, the pipeline can be adapted to new feature types without collecting labeled training data or retraining a specialized detector for each feature.
- We adapt Set-of-Mark prompting to aerial imagery and study multiple strategies for presenting segmented candidates to the vision-language model, including no-context, in-context, and combined views. This design makes it possible to test how local appearance cues and surrounding spatial context each contribute to identifying fine-grained urban features that are difficult to annotate through direct prompting alone.
- We evaluate the pipeline on two transportation-relevant features, stop lines and raised tables, and show that although direct prompting is essentially ineffective, the proposed prompting strategies substantially improve annotation quality. Beyond the quantitative evaluation, we analyze qualitative examples and failure cases to clarify both the promise and the current limitations of using general-purpose models to expand the coverage of urban accessibility data at scale.

4.2 *Related Work*

Urban Data Collection: In our annotation work, we primarily focus on collecting urban features from aerial satellite imagery, such as sidewalks, stop lines, and raised tables. Zhang et al. [304] introduced a novel dataset comprising aerial satellite imagery, street maps, and rasterized annotations of street objects in cities. Even after years of work, existing models recognize only the three most important features of the pedestrian network. These models require expansive, diverse training data to perform well. Our goal is to develop a scalable process for generating training data that captures these less frequent urban features.

Visual Prompting: Visual prompting is a technique that uses visual cues or examples to guide multimodal AI models in understanding and responding to visual tasks, similar to how textual prompts are used to instruct LLMs [217]. Our proposed procedure is based on Set-of-Mark (SoM) prompting [279], leveraging segmentation model SAM to partition images into smaller regions, and overlays each region with a mark to provide visual hints. Their work demonstrated extraordinary visual grounding ability of GPT-4.

Visual Grounding: our problem aligns with the VG task, which involves annotating objects in images by providing textual prompts to VLMs [281, 315]. However, VG tasks are typically evaluated on standard images (e.g., sightseeing, animals, buildings). To our knowledge, only a few studies have utilized VLMs to annotate objects in aerial images. Studies [295, 47] annotated generic objects from aerial images across a wide range of categories, such as basketball courts, harbors, and stadiums. They trained small VLMs from scratch, instead of using pre-trained large-scale VLMs. Studies [294, 142] created benchmark datasets for annotation, which only include commonly seen features such as roads, cars, and buses. Our study is original in that we focus on the scalability, generalizability, and cost-effectiveness of annotating diverse urban features, instead of just common objects. We leverage the SOTA VLM, generic segmentation model (SAM), and visual prompting strategy to enhance annotation performance, without requiring any labeled data.

4.3 Method

Our proposed annotation procedure is illustrated in Figure 4.1. The key idea is to transform open-ended object localization into a constrained candidate-selection task. Directly asking a vision-language model to identify and localize a small urban feature from a dense satellite image is difficult because the model must simultaneously recognize the target feature, distinguish it from visually similar structures, and return a precise spatial annotation. Instead, we first generate a broad set of candidate regions and then ask the model to choose the region that best matches a textual description of the feature of interest. This decomposition reduces ambiguity, makes the task more structured, and allows the same procedure to be reused across feature types with only limited feature-specific customization. More

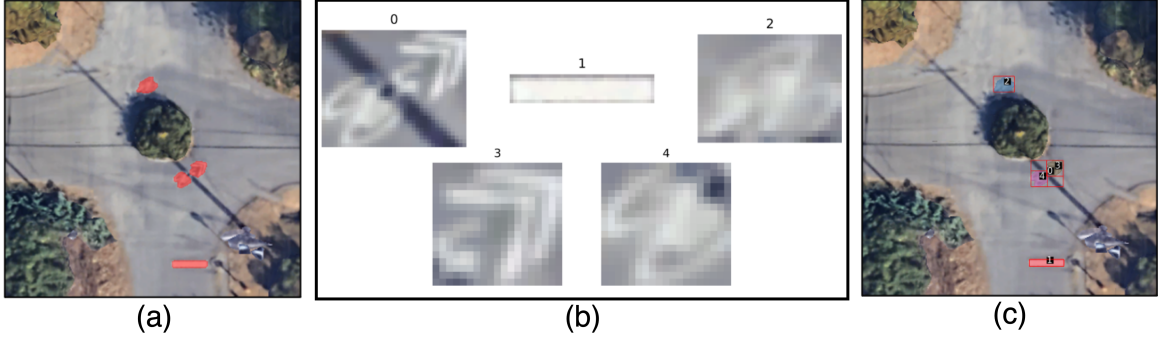


Figure 4.2: (a) Filtered segmented candidates using SAM model. (b) No-Context. Candidate objects are presented separately in a new image. (c) In-Context. Candidate objects are labeled with numbers and bounding boxes within the original image.

concretely, the pipeline consists of the following stages:

1. **User Input.** The procedure begins with a pair of inputs: a satellite image and an annotation guidance prompt in text. The satellite image provides the visual scene to be analyzed, while the guidance specifies the target feature and the annotation task. In practice, the guidance includes both semantic and operational information: it describes what the feature is, highlights its distinguishing visual characteristics, and instructs the model on how to respond. See Table 4.1 for examples.
2. **Image Segmentation.** We next apply a general-purpose image segmentation model to partition the image into candidate objects. In our implementation, we use SAM because it can generate segments without task-specific supervision and therefore serves as a flexible front end for zero-shot annotation. Importantly, segmentation is performed independently of the textual prompt. This choice keeps the procedure general: rather than requiring a feature-specific detector, we first produce a rich set of candidate components that may include the object of interest. Although this stage may over-segment the image and produce many irrelevant regions, high recall is desirable at this point because later stages can remove implausible candidates.

3. **Candidate Filtering.** The raw set of segments generated by SAM is often too large and noisy to present directly to the vision-language model. We therefore apply a lightweight filtering stage to narrow the search space. In our experiments, we use simple heuristics based on color and component size, removing green, yellow, brown, and red regions as well as components below a minimum area threshold.¹ These rules are simple: they reduce the number of clearly irrelevant segments while preserving feature candidates that should still be considered by the model. More broadly, this stage is where feature-specific domain knowledge can be incorporated in a transparent and low-cost way. Different urban features may require different thresholds, color exclusions, or additional geometric constraints.

4. **Set-of-Mark Generation.** After filtering, we convert the remaining candidate components into a representation that is easier for the vision-language model to reason over. Following Set-of-Mark prompting [279], we assign each candidate a unique visual identifier. This step turns the task from free-form localization into a structured selection problem: the model no longer needs to describe an arbitrary region in the image, but instead chooses among explicitly marked candidates. We evaluate three marking strategies, shown in Figure 4.2.
 - **No-Context (NC):** all candidates are cropped and presented together against a plain white background, with a numeric mark attached to it. This representation emphasizes the local appearance of the candidate while removing potentially distracting background content.
 - **In-Context (IC):** each candidate is marked with a number and bounding box within the original satellite image. This preserves spatial context, such as nearby lanes, intersections, and surrounding road geometry, which may be important for distinguishing visually similar objects.
 - **Combination (Comb):** both no-context and in-context views are generated and provided to the model. This representation is intended to combine the

¹200 pixel area for stop lines and 400 pixel area for raised tables.

benefits of localized appearance cues and broader scene context.

5. VLM Inference and Post-Processing. The marked image or images are then sent to the vision-language model together with the textual annotation guidance. The guidance has two roles: it defines the feature of interest and it specifies the expected annotation behavior. In other words, the prompt tells the model what to look for and how to report its decision. In our experiments, the model responds by identifying the mark corresponding to the most likely candidate. We then post-process the response by mapping the selected mark back to the original segmented component and returning its associated bounding box as the final annotation result.

This procedure is modular: each step can each be modified independently, making the pipeline extensible to new urban features and new application settings. In this sense, the method is designed not as a single-purpose detector, but as a general framework for low-cost, zero-shot annotation of built-environment features from aerial imagery.

4.4 Evaluation & Results

We conduct a proof-of-concept evaluation on two transportation-relevant urban features: stop lines and raised tables. We manually cropped eight satellite images that contain stop lines and ten images that contain raised tables from Google Earth. Each image is paired with a manually created ground-truth bounding box. Although this evaluation set is modest in size, it is appropriate for the goal of this chapter: to determine whether a zero-shot VLM-based pipeline can generate meaningful annotations at all for fine-grained built-environment features that are difficult to annotate at scale.

As a baseline, we compare against a **Direct Prompting (DP)** strategy. In this baseline, GPT-4o receives the original image directly, without SAM-based segmentation or Set-of-Mark prompting, and is asked to identify the target feature and return its bounding box coordinates. The corresponding prompt is shown in Table 4.1. This baseline tests whether a strong general-purpose vision-language model can solve the task in an unconstrained way, without first reducing the search space into explicit candidate objects. We compare this

Method	Stop Line
Direct Prompting	A stop line is a single white line painted on the road at intersections where traffic must stop. It shows drivers where to halt their vehicles. Please identify the bounding box of the stop line in the image in the format of (x1, y1, x2, y2).
SoM Prompting	A stop line is a single white line painted on the road at intersections where traffic must stop. It shows drivers where to halt their vehicles. Which labeled images represent the stop line?

Table 4.1: Example text prompts for the VLM model to annotate stop lines. **Pink** highlighted texts represent the descriptive information of the features. **Blue** highlighted texts represent the general task questions. Direct Prompting method: given an image, we directly ask the VLM to annotate the feature of interest and return the bounding boxes. SoM Prompting method: we input VLM with the image from the SoM generation process.

baseline against the three variants of our proposed procedure: **SoM-NC**, **SoM-IC**, and **SoM-Comb**. All three variants use the same segmentation and candidate-filtering stages and differ only in how the candidate regions are presented to the model. This design isolates the effect of visual prompting and the role of contextual information in zero-shot annotation.

For quantitative evaluation, we use the intersection-over-union (IoU) metric, $IoU = \frac{G \cap P}{G \cup P}$, where G is the ground-truth bounding box and P is the predicted bounding box. Higher IoU indicates stronger spatial agreement between the generated annotation and the manual reference annotation. Because the task is not only to recognize the correct feature type but also to localize it accurately, IoU provides a direct measure of annotation quality. The quantitative results are summarized in Table 4.2.

We observe that: **(1)** Direct prompting is largely ineffective for both feature types. It achieves an IoU of 0.0000 on stop lines and only 0.0190 on raised tables, indicating that GPT-4o alone is generally unable to localize these small and specialized urban features from aerial imagery when the task is posed in an unconstrained manner. In other words, the model struggles not only with precise localization, but with identifying the correct target

Feature	DP	SoM-NC	SoM-IC	SoM-Comb
Stop Line	0.0000	0.2483	0.3354	0.3657
Raised Table	0.0190	0.3315	0.4069	0.4189

Table 4.2: Quantitative IoU results for direct prompting and the three Set-of-Mark prompting variants on stop lines and raised tables. Higher values indicate better spatial alignment with the ground-truth bounding boxes.

region at all. **(2)** Introducing Set-of-Mark prompting substantially improves performance. Even the simplest prompted variant, SoM-NC, increases IoU to 0.2483 for stop lines and 0.3315 for raised tables. This result shows that reformulating the problem as selecting from a structured set of segmented candidates is already highly beneficial. Rather than requiring the model to infer an arbitrary bounding box directly from a dense aerial image, the procedure narrows the problem into a more manageable candidate-selection task. **(3)** Preserving scene context further improves annotation quality. SoM-IC increases performance to 0.3354 for stop lines and 0.4069 for raised tables. This suggests that surrounding visual context plays an important role in recognizing these features. For example, curb geometry, lane layout, road orientation, and intersection structure may help the model distinguish a true target feature from other road markings that look locally similar. The SoM-Comb variant achieves the best overall performance, reaching 0.3657 on stop lines and 0.4189 on raised tables. The additional gains over SoM-IC are modest, but they indicate that isolated object appearance and full-scene context can provide complementary information.

We also observe that raised tables are consistently easier to annotate than stop lines across all three SoM variants. One plausible explanation is that raised tables typically occupy a larger spatial region and have a stronger geometric footprint, whereas stop lines are thinner, smaller, and more easily confused with other painted road markings. This difference is important from an urban feature generation perspective: it suggests that zero-shot annotation may be more immediately effective for larger or more visually distinctive infrastructure elements, while thinner and more ambiguous features may require stronger

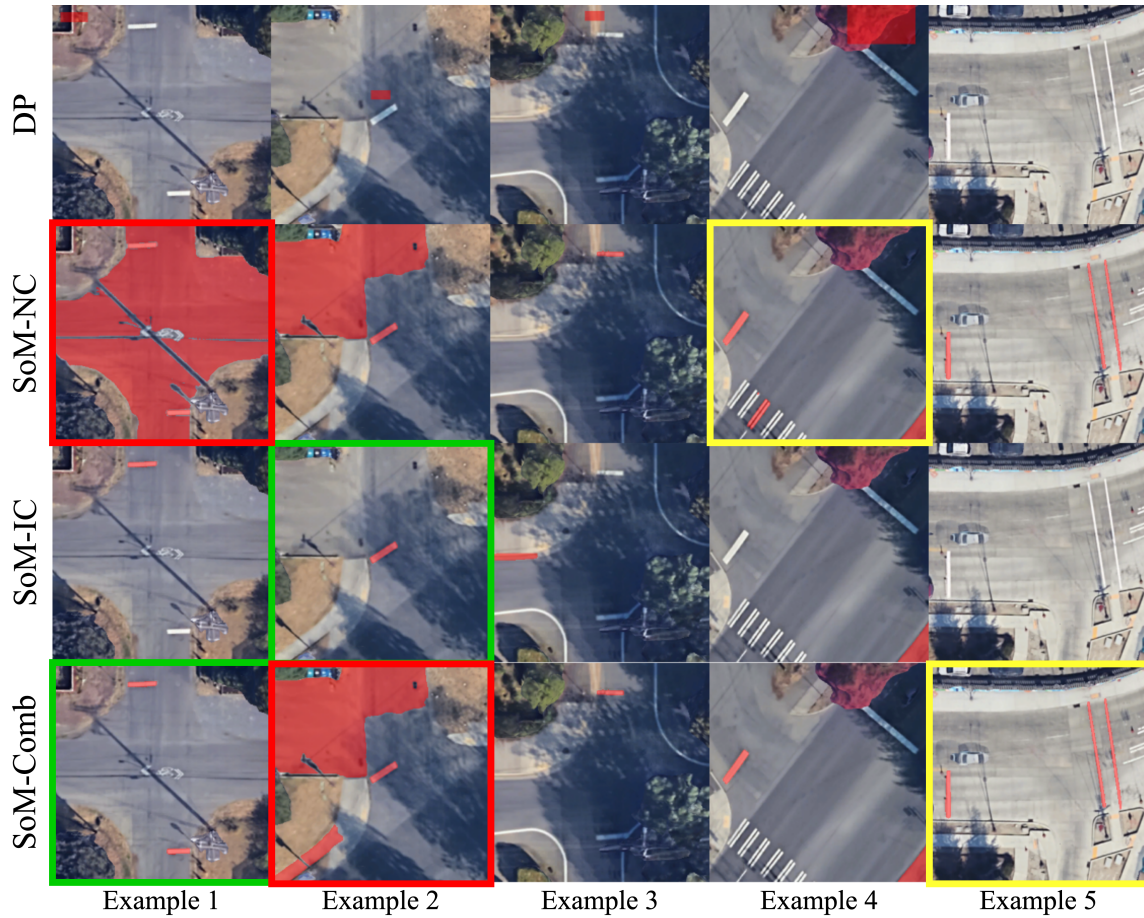


Figure 4.3: Examples of annotated stop lines. Green and yellow outlines indicate perfect and approximate annotations, respectively. Red outlines indicate inaccurate annotations.

filtering, improved prompting, or human verification.

Figure 4.3 and 4.4 provide qualitative examples. Across both feature types, the direct prompting baseline often produces boxes that are detached from the correct object or only weakly related to it, reinforcing the conclusion that the task is not amenable to naive prompting. In contrast, the SoM-based methods typically select regions closer to the correct feature, even when the final annotation is imperfect. Some examples show nearly exact matches (green), while others show partial matches (yellow), that still capture the correct general location. For instance, in the stop-line examples, the in-context view performs

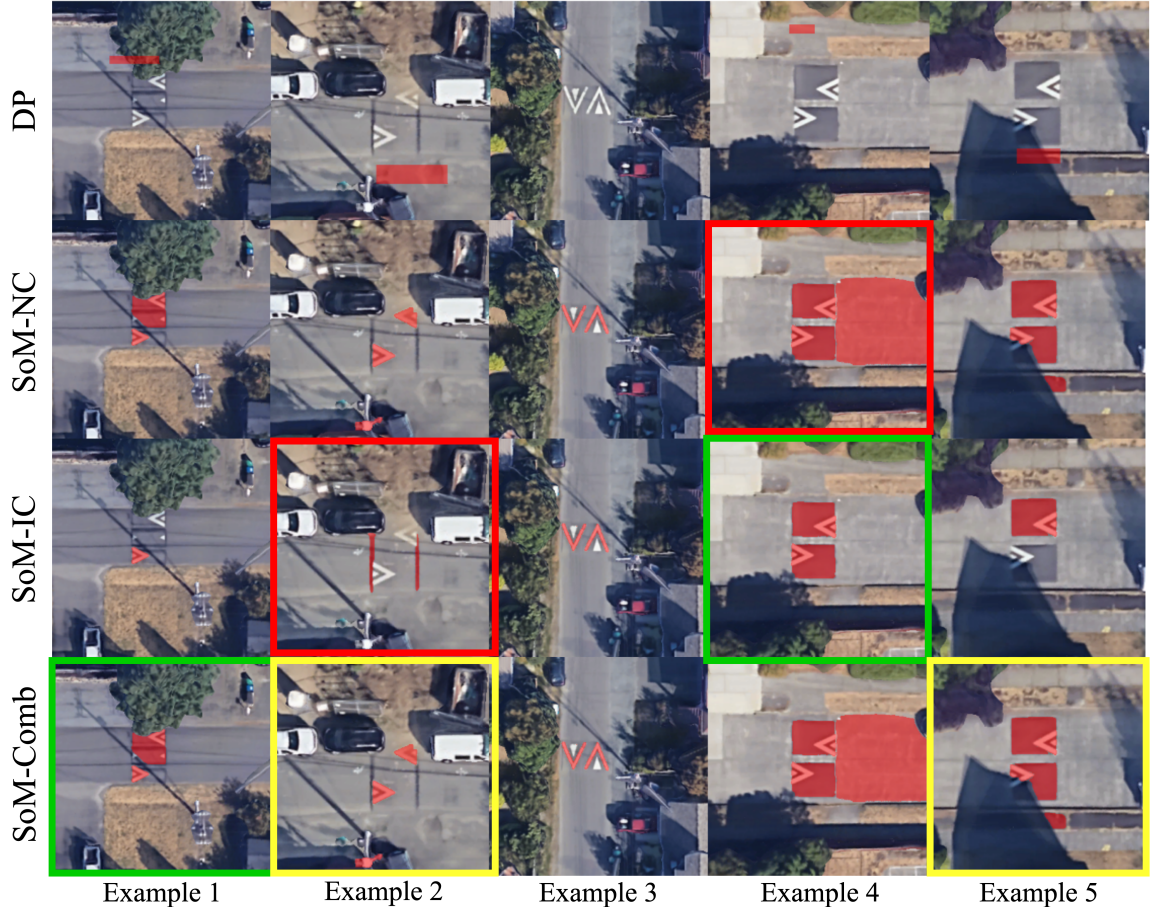


Figure 4.4: Examples of annotated raised tables. Green and yellow outlines indicate perfect and approximate annotations, respectively. Red outlines indicate inaccurate annotations.

especially well in Example 2, where local road geometry appears to help the model isolate the correct marking. In the raised-table examples, the in-context view also performs well in Example 4, while the no-context view over-selects a much larger pavement region.

The failure cases are also informative. A common error is that the selected candidate is spatially close to the true feature but is too large, too small, or merged with adjacent regions. This indicates that part of the remaining error comes not only from the VLM, but also from the quality of the candidate segmentation itself. In other cases, the model selects a visually plausible but semantically incorrect candidate, revealing that some road features

remain difficult to distinguish even after the search space has been constrained.

Overall, the evaluation shows that zero-shot annotation of fine-grained urban features from aerial imagery is difficult but feasible. The proposed prompting pipeline does not yet eliminate the need for human review, but it substantially outperforms direct prompting and demonstrates that general-purpose vision-language models can be repurposed to expand the coverage of urban feature datasets. In practice, the most realistic near-term role of this method is likely model-assisted annotation, where the system proposes candidate annotations that humans review and refine rather than drawing every label from scratch.

4.5 Summary

This chapter improves urban data quality along the dimension of coverage by addressing the difficulty of annotating diverse and often underrepresented built-environment features at scale. It proposes a zero-shot annotation framework that combines aerial imagery, class-agnostic segmentation, heuristic candidate filtering, and vision-language-model prompting to identify urban features without requiring labeled training data or task-specific fine-tuning. Evaluations on stop lines and raised tables show that direct prompting is essentially ineffective, while the proposed prompting strategies produce substantially better annotations. These results demonstrate that general-purpose vision-language models, when embedded in a structured annotation pipeline, can serve as a practical starting point for expanding the digital representation of transportation-relevant urban features.

Automated VLM-based annotation also introduces potential risks. (1) Error propagation. Because these models are approximate and zero-shot, incorrect annotations could be integrated into navigation or accessibility applications. For example, a false positive for a curb ramp or crossing could lead to unsafe routing decisions in the real world. (2) Surveillance and monitoring. A scalable pipeline for identifying fine-grained built-environment features could also be used for purposes beyond public benefit. For example, it could be used to monitor structures in specific neighborhoods for punitive enforcement rather than infrastructure improvement. For these reasons, generated annotations should be treated as preliminary outputs, not official records. Any deployment should include targeted auditing, human validation, and clear documentation of where the system is likely to make mistakes.

Chapter 5

IMPROVING URBAN DATA COHERENCE THROUGH SPATIAL DATA INTEGRATION

Urban data quality depends not only on how much data is available, but also on whether data from different sources can be reconciled into a coherent representation of the built environment. This chapter addresses urban data quality along the dimension of coherence by studying spatial data integration as a cross-source matching problem. Focusing on spatial join and spatial union tasks over road and pedestrian pathway networks, it evaluates whether large language models can support integration directly from natural-language task descriptions without task-specific labeled data. The results show that LLMs struggle when required to derive and solve the relevant computational geometry on their own, but become highly effective when provided with human-interpretable heuristic features. The chapter further introduces a review-and-refine prompting strategy that improves weak initial answers while preserving strong ones, demonstrating a practical path toward generating more coherent and integrated urban datasets. This chapter is based on our work published at the Conference on Language Modeling (COLM) 2025 [78].

5.1 Introduction

In addition to their language generation capabilities, Large Language Models (LLMs) provide intuitive natural language interfaces for advanced computational tasks, such as code generation [27], data manipulation [85], context understanding [56] and more. These capabilities reduce reliance on software engineers for domain experts, democratizing computational solutions into under-resourced domains.

In this context, we are interested in supporting experts in the spatial domain to perform macro-scale spatial data *integration* — the process of integrating multiple datasets from different sources into a higher quality dataset [181]. We particularly target scenarios involving network data representing roads and pedestrian pathways [212], where individual sources,

though sparse, biased, and erroneous, offer complementary information when integrated effectively [22]. A core primitive for integration is to match two elements from different datasets, which is traditionally achieved through either optimization and machine learning-based methods or heuristic approaches rooted in geometrical properties [212]. Rule-based heuristic approaches handle common, predictable spatial patterns [25, 124] but struggle to address edge cases in practice [313, 197]. Optimization and machine learning methods, while potentially more robust, require extensive labeled data and complex problem formulation.

In this study, we explore the potential of LLMs to perform paired-elements matching based on natural language specifications provided by domain experts, and without the need for labeled data. Effective matching relies not only on computational geometry primitives [106, 236] but also on a common sense understanding of the built environment. LLMs, pre-trained on vast diverse datasets [48], possess the ability to discern semantic relationships and infer implicit patterns [132], making them potentially useful for reasoning about spatial data represented in text-based formats. However, their ability to interpret natural language specifications as computational geometry problems and then solve them remains unclear.

We examine two element matching tasks — *spatial join*, which associates pairs of elements according to real-world relationships, and *spatial union*, which determines whether two objects fully or partially represent the same real-world entity. In both tasks, the input comprises candidate pairs of elements. Each element is a linestring, a sequence of contiguous edges connected end-to-end. The output is an indication of whether these pairs match according to the user-specified semantic relationship (e.g., a sidewalk adjacent to a road for the join task) or whether they represent the same spatial feature (union task) (Figure 5.1).

We evaluate three heuristic-based baselines on both tasks. Heuristic methods identify relevant features derived from the pair (e.g., minimum angle between pairs of sub-edges, or minimum distance between points) and an appropriate threshold for each feature. We test against heuristic-based methods rather than broader machine learning-based methods for three reasons — First, heuristics are effective on both tasks, allowing us to compare LLMs’ ability. Second, simpler heuristic methods help isolate and clearly identify the fundamental limitations of LLMs in spatial reasoning tasks. Third, domain experts can easily express matching criteria using natural language rather than formulating complex algorithms.



Figure 5.1: Positive and negative examples for the spatial join and union task. (1) For the spatial join task, the blue line represents the road, and the red line represents the sidewalk. The task classifies whether a sidewalk “runs alongside” a road from a pedestrian’s perspective. (2) For the union task, both lines represent sidewalks. The task determines whether two sidewalks fully or partially represent the same real-world entity.

We assess five LLMs using two methods — **(1) Heuristic prompting method.** We first assess LLM performance using zero-shot and few-shot natural language prompts, directly instructing models to generate labels without further explanation. We then incorporate heuristic guidance in the prompt, first in natural language (*hints*) and then as pre-computed numeric quantities (*features*). We find that natural language alone performs poorly even with hints (<60% accuracy), due to frequent computational and logical errors in identifying geometric features or thresholds. But when pre-computed features are included, sidestepping the spatial reasoning task, the models are able to derive an appropriate threshold based on understanding, thereby significantly improving performance (>90% accuracy). This method is valuable in practice — reducing reliance on precise threshold specifications from prior data or domain knowledge, and enabling domain experts to run task-agnostic computations. **(2) Review-and-refine prompting method.** Motivated by self-correction research in LLMs, we employ a prompting approach where models first receive an initial answer (from heuristics, random guesses, or other methods) and are then prompted to review and refine it. This two-step method consistently improves poor initial answers and preserves accurate answers, enhancing overall accuracy.

Overall, this chapter makes five primary contributions toward improving urban data

coherence through spatial data integration with large language models:

- We conduct comprehensive empirical evaluations of LLMs on spatial integration tasks, an area that remains largely underexplored. Our study serves as an initial step toward understanding how LLMs can be effectively used in spatial applications.
- We demonstrate that, while LLMs struggle with natural language instruction alone, prompting with geometrical features derived from heuristics significantly enhances performance, reaching peak accuracies of 98.4% and 96.0% for each task, respectively.
- We perform qualitative analysis of spatial reasoning on real-world geometries instead of abstract shapes and textbook math problems, better reflecting practical challenges and new opportunities. The analysis reveals that while LLMs possess spatial reasoning abilities, they tend to commit logical and computational errors.
- We propose a two-step review-and-refine prompting strategy that enhances poor initial answers (average improvements of 38% and 7.4% on the two tasks) while preserving the accuracy of strong initial answers. This approach reaches best accuracies of up to 99.5% and 96.5%, surpassing the performance of the most effective heuristic methods.
- We discuss the advantages and limitations of employing LLMs for spatial data integration tasks. Furthermore, we outline future research to address these limitations and enhance the applicability of LLMs for spatial data applications.

5.2 *Related Work*

Generative Language Models & Diverse Applications: We study autoregressive LLMs pretrained on large text datasets, particularly those instruction fine-tuned for natural, chat-based interactions [262, 191]. LLMs have been studied and applied across diverse domains, including medicine [205, 81], finance [270, 125], law [109, 159], education [306, 38], software development [311, 147], behavioral analysis [32, 272], translation [126, 79], and fact-checking [40, 269]. Our research focuses on urban analytics, building upon prior work

by [252], who developed personalized urban mobility strategies using LLMs, and [80], who introduced a zero-shot urban annotation framework with vision-language models (VLM).

Self-Correction Using LLMs: Recent studies have demonstrated that prompting LLMs to self-correct can substantially enhance task performance [168, 120, 116]. The core idea involves an LLM initially generating an output; and subsequently using the same model to iteratively provide feedback and refine the initial output. Similarly, cross-model correction methods utilize feedback provided by different models than those that generated the initial responses [34, 141, 145]. Importantly, both correction strategies eliminate the need for supervised training or fine-tuning. A related concept, the “LLM-as-a-Judge” paradigm, uses LLMs to evaluate or rank outputs across various applications, including retrieval [143], alignment [133], and reasoning [309]. Detailed surveys are available for further exploration [116, 135, 136].

Spatial Data Integration: *Spatial data join* merges two spatial datasets by evaluating geographic features. Rule-based methods include geometrical and distance-based joins [103]. Geometrical joins utilize qualitative spatial predicates (e.g., intersects, contains, overlaps) to link features [96], while distance-based joins rely on quantitative distance measurements. *Spatial data union*, a spatial overlay operation, combines multiple spatial datasets into one comprehensive dataset covering the combined area [170]. Our study differs from prior research in that — rather than optimizing efficiency or scalability for large-scale datasets, using known rules [51, 247], we address ambiguous scenarios requiring human judgment. Additionally, we use LLMs to evaluate joinability and unionability without explicit rules.

5.3 Task & Datasets

5.3.1 Task Definition

Spatial join and union tasks integrate datasets from heterogeneous geospatial resources. In our context, the two tasks determine whether two geometric objects should be linked based on specific criteria. Denote $\mathcal{S}_i = \{s_1, s_2, \dots, s_n\}$ as a LineString geometry representing a

sidewalk annotation, connecting each pair of points (s_i, s_{i+1}) with a line. We will refer to each sub-line as an *edge*. Similarly, let $\mathcal{R}_i = \{r_1, r_2, \dots, r_m\}$ to represent a road annotation.

The Spatial Join task classifies whether a sidewalk “runs alongside” a road from a pedestrian’s perspective. This relationship is not precisely defined, but is important for screen readers used by vision impaired travelers. If the reader makes a mistake, the audible directions could be misleading or in the worst case dangerous: the road should be called out when sighted travelers would tend to agree that they are walking along the road. Computationally, this relationship corresponds, approximately, to adjacency and parallelism between road and sidewalk, defined by a binary classification function $f_\theta(\mathcal{S}_i, \mathcal{R}_i) \rightarrow \{0, 1\}$.

The Spatial Union task determines whether two sidewalk annotations from different sources, $\mathcal{S}_i$ and $\mathcal{S}'_i$, refer fully or partially to the same sidewalk, defined similarly as $f_\theta(\mathcal{S}_i, \mathcal{S}'_i) \rightarrow \{0, 1\}$. Complications include partial matches and errors in one or both sources, and heterogeneity in different built environments and relevant laws.

We chose these two tasks because: **(1)** while they appear simple, they are foundational to more complex spatial analysis, such that focusing on these ubiquitous, fundamental tasks provides us with a highly interpretable view of LLMs’ basic capabilities in handling spatial tasks, and helps us identify the strengths and weaknesses of LLMs in working with spatial data; and **(2)** a small number of heuristic rules perform well for these two tasks, which allows us to test LLM spatial reasoning in a controlled way. In cases where heuristics are insufficient and do not generalize well, we can assess LLMs ability on those specific cases.

5.3.2 Datasets

We obtained two datasets from the Transportation Data Exchange Initiative [102]: **(1)** Bellevue-Sidewalk, comprising 14,200 sidewalk-specific annotations of attributes such as footway type and geometry; and **(2)** Bellevue-City, containing 73,700 annotations of both roads and sidewalks sourced from OpenStreetMaps, annotated with footway type, highway type, and geometry. To construct prediction datasets for our analysis, we implemented the following pre-processing steps for each dataset:

Spatial Join Dataset. We first filter roads from Bellevue-City, selecting only those

with highway types in {secondary, residential, tertiary, primary, living_street} (types likely to have paired sidewalks), resulting in 17,800 road annotations. We add a 10-meter buffer around each road. The 10-meter buffer reflects the origins of our dataset in accessibility research, which is typically concerned only with roads that have sidewalks. Thus, while there is substantial variation in real-world road widths, our dataset includes only road types likely to have adjacent sidewalks – primary, secondary, tertiary, residential, living_street, and excludes highways. The selection of the 10-meter buffer is based on typical road widths in Seattle and Bellevue, and applied across all types. Next, we perform a spatial join operation between the buffered roads and sidewalks using an intersection predicate. This step generates 14,556 potential $(\mathcal{S}_i, \mathcal{R}_i)$ candidate pairs, under the assumption that a sidewalk running alongside a road should fall within the buffer zone. Each $(\mathcal{S}_i, \mathcal{R}_i)$ pair was then manually labeled through visual inspection, resulting in 6,034 positive pairs and 4,206 negative pairs. We split the labeled dataset into training, validation, and test sets, containing 6,442; 716; and 1,000 samples, respectively. The remaining pairs were deemed ambiguous and required case-by-case discussion; they were thus excluded from this study.

Spatial Union Dataset. We use the 17,100 sidewalk annotations from Bellevue-City. A spatial join operation is performed between the two sources of sidewalk annotations using an intersection predicate. This step generates 5,496 $(\mathcal{S}_i, \mathcal{S}'_i)$ unionable candidate pairs, under the assumption that if two annotations represent the same sidewalk, either fully or partially, they should overlap. Each pair was manually labeled through visual inspection, resulting in 886 positive pairs and 443 negative pairs. The final dataset was split into training, validation, and test sets, containing 837; 93; and 399 samples.

Ultimately, for the join task, we have 6,442, 716, and 1,000 $(\mathcal{S}_i, \mathcal{R}_i)$ training, validation and test pairs. For the spatial union task, we have 837, 93, and 399 $(\mathcal{S}_i, \mathcal{S}'_i)$ pairs. Figure 5.1 illustrates a positive and a negative example for each task. Note that the candidate pairs generation process uses certain spatial conditions (e.g., 10-meter buffer for join data and intersection for union data). We use those loose conditions to narrow down the search space for the potential pairs. We are not checking all possible pairs.. For all object pairs in both datasets, we compute three features based on geometrical relationships between the objects (which are predictably and demonstrably relevant to the task):

- **min_angle**: the smallest angle (in degrees) between two spatial objects, computed by dividing each annotation into smaller segments, calculating pairwise acute angles between segments, and selecting the smallest angle as an indication of object alignment.
- **min_distance**: the smallest distance (in meters) between any point on one annotation and any point on the other annotation. This metric reflects the proximity between geometries.
- **max_area**: the percentage of overlapping area between two objects, calculated after applying a 10-meter buffer around each object. This metric reflects the potential similarity between geometries.

Appendix 5.8.1 describes the distributions of all geometrical features in both datasets.

5.4 Experiments

In this section, we describe the heuristic baselines (§5.4.1), our proposed heuristic prompting method (§5.4.2), and the review-and-refine method (§5.4.3). In addition, we describe the experimental setup and language models assessed in our study in §5.4.4.

5.4.1 Baseline Heuristics

We use relevant heuristics commonly applied in the join or union task that leverage geometrical relationships between the pairs. Their effectiveness helps us compare LLMs’ ability:

- **Parallelism (p)**: evaluates whether two linestrings are approximately parallel, quantified by the minimum angle (**min_angle**) between the pairs. We test five thresholds for each task, $\alpha = \{1, 2, 5, 10, 20\}$ degrees for the join task and $\alpha = \{1, 2, 3, 4, 5\}$ degrees for the union task. The heuristic is computed as: $\hat{y} = \mathbb{1}[\text{min_angle} \leq \alpha]$, where α is a pre-defined threshold value.
- **Clearance Heuristic (c)**: assesses whether two linestrings maintain a minimum distance (**min_distance**) between them. For example, a sidewalk likely runs alongside a road if the distance between is above a given threshold. We test five thresholds $\{1, 2, 3, 4, 5\}$ exclusively in the join task (candidate pairs for union intersect, such that

`min_distance` is always zero). The heuristic is computed as: $\hat{y} = \mathbb{1}[\text{min_distance} \geq \alpha]$, where α is a threshold value.

- **Overlap Heuristic (o):** determines the extent of overlap between linestrings using `max_area`. Two annotations likely represent the same object if they exhibit a high overlap ratio. We report $\{0.1, 0.2, 0.3, 0.4, 0.5\}$ percent thresholds for join and $\{0.5, 0.6, 0.7, 0.8, 0.9\}$ for union. The heuristic is computed as: $\hat{y} = \mathbb{1}[\text{max_area} \geq \alpha]$
- **Duo & Trio Heuristics:** We combine two or three heuristics for each task. We use abbreviations $\{(p), (c), (o)\}$ to denote the parallel, clearance, and overlap heuristics respectively. For join, we evaluate three duo heuristics $\{(p,c), (p,o), (c,o)\}$ and one trio heuristic $\{(p,c,o)\}$. For union, we evaluate one duo heuristic $\{(p,o)\}$. We use the same thresholds mentioned above.

5.4.2 Heuristic Prompting Method

As a baseline, we first evaluate pre-trained LLMs using **plain** prompting. In this setting, the model receives only a brief task description together with the corresponding GeoJSON of the objects, and is asked to predict a binary label indicating whether the pair should be matched. We evaluate both zero-shot and few-shot versions of this baseline. In the zero-shot setting, no labeled examples are provided. In the few-shot setting, we include one positive and one negative example in the prompt. This plain prompting baseline is intentionally challenging. It requires the model to infer the relevant spatial criteria directly from raw coordinates, translate the natural-language task into an implicit geometrical rule, and then carry out the corresponding reasoning correctly. This setup allows us to test whether general-purpose LLMs can solve spatial integration tasks directly from task descriptions alone.

We then introduce **heuristic prompting**, which augments the plain prompt with heuristic guidance derived from standard spatial integration practice. The intuition is that spatial integration often depends on a small set of interpretable geometrical criteria, such as alignment, clearance, and overlap, but users may not know in advance which criteria or thresholds are most appropriate for a given task. Rather than hard-coding a single rule, we provide the model with information about one or a few candidate heuristics and ask it

Plain Prompting	Heuristic Prompting
### Instruction: You are a helpful geospatial analysis assistant. Please determine whether the sidewalk runs alongside the road. ### User: Sidewalk: {'coordinates': [[-122.2, 47.5], [-122.1, 47.8]]} Road: {'coordinates': [[-122.5, 47.2], [-122.6, 47.4]]}	### Instruction: You are a helpful geospatial analysis assistant. Please determine whether the sidewalk runs alongside the road. ### User: Sidewalk: {'coordinates': [[-122.2, 47.5], [-122.1, 47.8]]} Road: {'coordinates': [[-122.5, 47.2], [-122.6, 47.4]]} min_angle: 9.9738 min_distance: 8.7260
### Response: 0  No Geometrical Features	### Response: 1  With Geometrical Features

Figure 5.2: Plain prompting (left) and our heuristic prompting with features (right) for the spatial integration tasks. In the plain prompting, only the basic task description is provided. In the heuristic prompting, features derived from heuristics are included to further enhance performance.

to use that information to decide the label. For the join task, the guidance may involve parallelism, clearance, overlap, or their combinations. For the union task, where the two geometries already intersect, the guidance focuses on parallelism and overlap. We consider two forms of heuristic guidance:

- **Natural Language Hints (hints):** We describe one or more relevant heuristics qualitatively in natural language, without supplying their numeric values. For example, the prompt may indicate that a sidewalk running alongside a road is typically approximately parallel to it and separated by a reasonable distance. This setting tests whether the model can use heuristic descriptions alone to identify the relevant spatial relationships and reason from the raw geometries.
- **Geometrical Features (features):** We provide the same natural-language heuristic descriptions, but additionally include pre-computed numerical feature values, such as `min_angle`, `min_distance`, and `max_area`. This setting changes the task substantially. Instead of deriving the relevant geometrical quantities from the coordinates itself, the model can treat the problem as one of heuristic evaluation: it must interpret the provided feature values, infer whether they satisfy an appropriate threshold or

combination of thresholds, and then predict the binary label. In this sense, feature-based prompting separates feature extraction from decision-making, allowing us to test whether LLMs are more effective as flexible rule interpreters than as standalone computational geometry solvers.

Across all experiments, we keep the prompt format as consistent as possible and vary only the amount and type of heuristic information provided. It allows us to directly compare plain prompting, hint-based prompting, and feature-based prompting, and to assess how much performance gain comes from adding qualitative guidance versus explicit numerical evidence. Figure 5.2 illustrates examples of the plain and heuristic prompting templates.

5.4.3 *Review-And-Refine Method*

Heuristic performance varies significantly with threshold selection. We consider whether LLMs can automatically select an appropriate threshold without labeled data. That is, can we improve on a “bad” guess of a threshold using the parameterized knowledge of an LLM? For each task, we generate initial responses using the best- and worst-performing heuristics, as well as random guesses. Following Kim et al [120], we provide initial guesses in a prompt and ask the model first to review the guess and identify potential issues. Then in a second pass, we include the review with the initial prompt and request a refined answer. Unlike Kim et al, we perform these steps only once rather than iteratively. All experiments are conducted under few-shot prompting, utilizing either heuristic hints or features.

5.4.4 *Models & Experimental Setup*

For our primary experiments, we evaluate five pretrained LLMs. These include two smaller, open-source models, {Meta-Llama-3.1-8B-Instruct (llama3), Mistral-7B-Instruct-v0.3 (mistral)}, and three larger proprietary, API-based models, {GPT-4o-mini, 2024-07-18 snapshot (4o-mini), Qwen-plus (qwen-plus), GPT-4o, 2024-08-06 snapshot (4o)}. Models were selected based on their scale, performance on public LLM leaderboards, and inference costs. For a qualitative ablation study, we also assess a proprietary, API-based reasoning model: Deepseek-R1. Reasoning models are post-trained using reinforcement learning for complex

heuristics	spatial join			spatial union		
	worst	best	avg (std.)	worst	best	avg (std.)
single	0.654	0.948	0.822 (0.10)	0.842	0.927	0.880 (0.03)
duo	0.735	0.987	0.910 (0.06)	0.870	0.962	0.933 (0.02)
trio	0.715	0.990	0.904 (0.08)	-	-	-
overall	0.654	0.990	0.900 (0.08)	0.842	0.962	0.918 (0.04)

Table 5.1: Baseline heuristic results for the spatial join and union tasks. The worst, the best, and the average accuracy (standard deviation) are reported for single, duo, and trio (spatial join only) heuristics. Standard deviations are calculated given different thresholds. For the spatial join task, duo results are averaged across all three duo combinations.

reasoning tasks, and they exhibit the capacity to improve their answers by drawing on internal chains of thought before responding to a user [73].

The experimental setups remain consistent across models and two tasks. For llama3 and mistral, we employ Unsloth [39] for accelerated inference with 4-bit quantization, configuring *max_sequence_length* and *top_p* to 4096 and 10. For proprietary models, *temperature* and *top_p* are set to 0 and 1. For the heuristic-driven prompting method, all models generate 10 new tokens per prompt. For the review-and-refine method, each model produces 500 new tokens. Outputs are post-processed and evaluated using accuracy as the metric.

5.5 Results

5.5.1 Baseline heuristics can be highly effective, but performance depends on task-specific selection and meticulous tuning of feature thresholds.

Summary of baseline heuristic results for both tasks are presented in Table 5.1. We observe that these tasks are amenable to effective heuristic solutions: For the join task, the best single heuristic achieves 94.8% accuracy, whereas duo and trio heuristics achieve 98.7% and 99.0%. We observe a similar trend for the union task, with duo heuristics outperforming single heuristics. Despite their effectiveness, the ability for target users to select good heuristics and good thresholds cannot be assumed, as they vary from task to task. For

example, the worst-performing join feature (`max_area` overlap) is up to 24% lower accuracy than the best (`min_angle` parallelism), even with good thresholds. For the best feature, the difference in accuracy between the best and worst threshold is 33.6%. Similar observation applies to the union task. Finally, the best choice of feature and threshold vary from task-to-task suggesting pressure on users to develop intuition about every possible integration scenario. Even with user expertise, practically identifying the optimal configuration remains a challenging task likely to involve trial and error.

5.5.2 While spatial integration tasks inherently challenge LLMs, incorporating heuristic features significantly enhances performances.

Average results of heuristic-driven prompting for join and union are visualized in Figure 5.3. Detailed numerical results are provided in Tables 5.5 and 5.6, respectively. We observe that: prompting with natural language (plain), even with feature hints (hints), is not effective. The models achieve results comparable to the worst heuristics. For the join task, the five models reach average accuracies of 54.2% and 57.5% under zero- and few-shot prompting, which are 39.8% and 36.1% lower than the overall heuristic average accuracy. For union, the models attain average accuracies of 60.1% and 59.6%, 34.6% and 35.1% lower than heuristic averages. These results indicate that models struggle to understand the scenario, translate the scenario into a computational problem, or accurately solve the problem.

However, prompting with heuristic features is remarkably effective! The availability of pre-computed features may afford sidestepping the computational geometry and provide a more direct link between the natural language specification and the identification of an appropriate threshold. We notice that — **(1)** Larger proprietary models deliver the most competitive performance. For the join task, smaller open models (llama3 and mistral) exhibit low accuracy, suggesting their limitations for spatial reasoning. The average accuracies of qwen-plus (90%) and 4o (90.1%) match the overall heuristic baseline (90%), with peak performance of 98.4% (qwen-plus) and 97.4% (4o) from Table 5.5. Qwen-plus aligns closely with the best-performing duo heuristic (98.7%) and nearly matches the top trio heuristic (99.0%). Similar trends appear in the spatial union task, with qwen-plus and

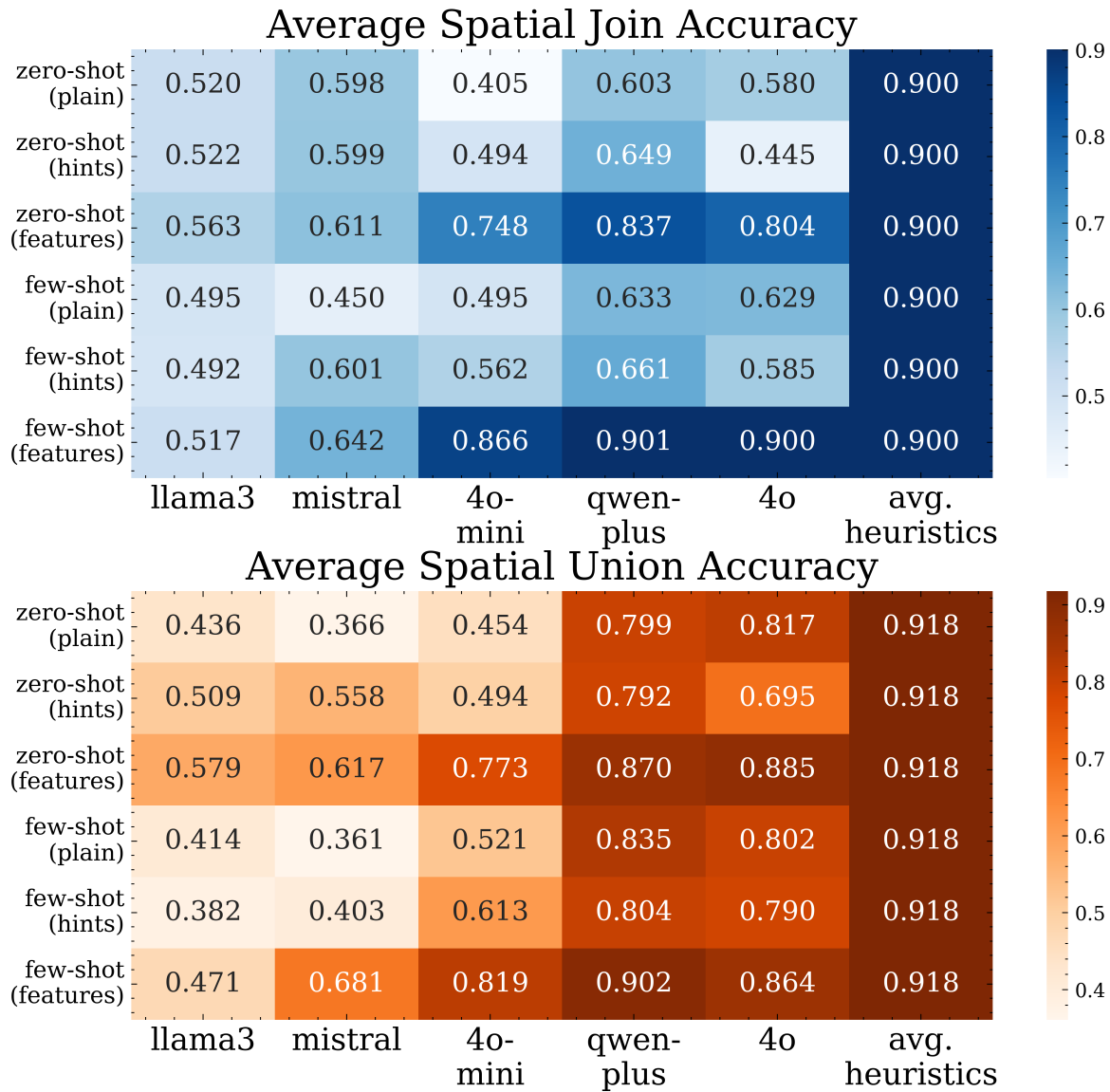


Figure 5.3: Average results of heuristic prompting method for the spatial join and union task. Each row represents prompting with no heuristic (plain) or different heuristic information (hints or features), under either zero- or few-shot prompting.

4o achieving peak accuracies of 96.0% and 95.5%, respectively, closely matching the best duo heuristic (96.2%). The subsequent observations focus exclusively on these proprietary models. **(2)** Prompting with more heuristic features improves accuracy. Table 5.5 shows

that `min_distance` or `max_area` heuristics are ineffective on their own but combinations improve performance. While expected, this result provides empirical evidence that LLMs can derive complex rule-based systems from simple geometric primitives. **(3)** LLMs eliminate the need for feature selection and threshold tuning. The selection of different features and corresponding thresholds significantly mediates heuristic-based performance; LLMs are effective in selecting both feature and threshold based on their spatial awareness. For the spatial join task, there are 215 possible heuristic combinations (15 single, 75 duo, and 125 trio combinations). The highest qwen-plus accuracy (98.4%) surpasses 93% of these heuristic combinations, while the top 4o model accuracy (97.4%) outperforms 85%.

5.5.3 LLMs exhibit fundamental spatial reasoning abilities, but struggle to derive and solve the relevant computational geometry.

We consider how LLMs approach the element matching task in order to understand why providing numeric features is required for competitive performance. For the join task, we employ few-shot, chain-of-thought (CoT) prompting on the {4o-mini, qwen-plus, 4o} models, using the standard “step by step” CoT instruction and setting the maximum generation tokens to 2,000. We inspected the outputs across 20 samples, under no heuristics (plain), with hints, and with features. We have the following observations.

Models exhibit spatial reasoning capabilities. Even without heuristic features, models understand the task by evaluating spatial conditions such as proximity (distance) and alignment (angle or orientation) between sidewalk-road pairs. Specifically, model 4o-mini mentions {proximity, distance} in 19 of 20 instances and {alignment, angle} in 5 of 20 instances (20/20 and 19/20, respectively, for qwen-plus; 20/20 and 11/20 for 4o). These findings suggest LLMs can identify spatial concepts relevant to the task.

Models are ineffective at computational geometry. All three models interpret the spatial join as a computational task involving geometrical calculations but consistently fail to execute these correctly. Errors fall into three categories: **(1)** Incorrect computational logic, with fundamentally flawed reasoning; **(2)** Unclear computational explanations, characterized by vague details, often indicated by terms like {appears, approximately}; **(3)**

common mistakes	examples
wrong logic	- orientation of road/sidewalk with first and last point - distance calculation with first point of road and last point of sidewalk
unclear calculation	- the closest approach between them appears to be around 2-3 meters - the closest sidewalk point to the road is approximately 5-10 meters away - the distance is approximately 3 meters
no calculation	- let's assume the calculations have been performed - let's assume the computed distance is approximately 1 meter - the sidewalk appears to be running parallel

Table 5.2: Common mistake examples in the reasoning steps from {4o-mini, qwen-plus, 4o}. The *Wrong Logic* category describes logical mistakes made by the models when analyzing spatial relationships between the paired objects. *Unclear Calculation* describes mistakes where there is no explicit demonstration by the models of the calculation steps used to arrive at an answer, though some numerical evidence may be provided. In such cases, models tend to use words such as “appears” and “approximately” rather than fully demonstrating their computations, casting uncertainty upon whether the models actually execute the calculation, and if so, how they do so. The *No Calculation* category is similar to the *Unclear Calculation* category. The difference is that, in this category, the model does not provide any numerical evidence whatsoever for its answers. The models tend to verbalize their assumptions, which they use to arrive at a final conclusion for a task.

Conclusions without computation, where results are asserted without computational backing, signaled by keywords like {assume, appear}. Examples of these errors are shown in Table 5.2. Occasionally, correct conclusions occur by chance despite wrong computations.

Providing heuristic features fundamentally changes the approach to the task. Including heuristic features transforms the spatial join task from a computational geometry problem into a simpler heuristic evaluation problem, where models verify heuristic conditions based on provided values. It significantly reduces task difficulty and enhances model performance. We also notice that — First, models typically apply reasonable heuristic thresholds (e.g.,

10 degrees for angles, 5 meters for distance, 30% for overlap). However, some thresholds do not clearly differentiate between provided positive and negative examples, indicating that decisions are not solely data-driven, but more relying on their inherent understanding of the task and real-world context. Second, thresholds chosen by models vary considerably across examples, ranging from 2–10 meters for distance and 20%–70% for overlap. This variability indicates that LLMs employ a flexible heuristic approach with potentially distinct thresholds per sample rather than a fixed universal threshold.

We conducted the same CoT prompting on 20 samples using Deepseek-R1, a reasoning model known for generating detailed reasoning steps. We avoid processing all 1000 samples to reduce costs, computational time, manual review time, and because the patterns began to converge. We observed that — **(1)** Deepseek-R1 exhibits strong computational geometry capabilities. Similar to other models, Deepseek-R1 interprets the task computationally without heuristic features. However, Deepseek-R1 performs precise geometrical calculations, including point-by-point and segment-by-segment evaluations, occasionally utilizing advanced measures like the Haversine distance. **(2)** However, its conclusion logic is fundamentally flawed. Despite advanced computational skills, Deepseek-R1 employs unreliable logic when arriving at conclusions. For instance, to determine parallelism, it cites evidence such as “The road’s vector is southeast, and sidewalk’s vector is northeast; thus, they are not parallel.” To assess proximity, it references questionable logic such as “The sidewalk’s start is 23 meters west of the road’s start, beyond typical adjacency for sidewalks.” This suggests that even models with powerful computational skills may struggle with spatial reasoning and logical coherence essential for spatial integration tasks.

5.5.4 Review-and-refine method is highly effective — significantly improving poor initial answers while preserving good ones.

Figure 5.4 presents the results of the review-and-refine method for both spatial integration tasks. We observe that — **(1)** Review-and-refine with only hints is ineffective. As previously discussed, direct prompting with natural language hints, yields poor results. This trend continues here — in the spatial join task, average accuracy for 4o-mini, qwen-plus, and

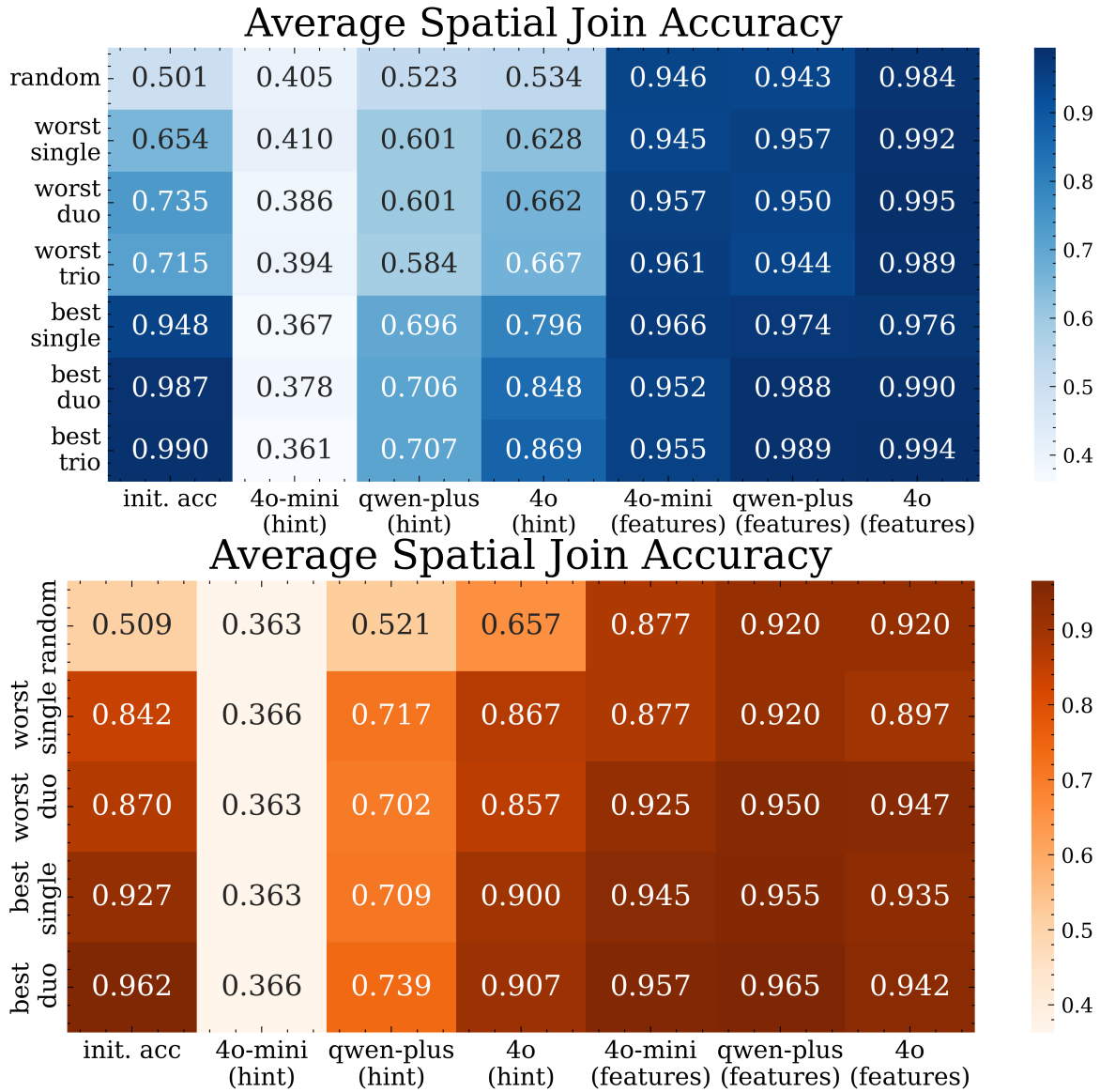


Figure 5.4: Accuracies of review-and-refine method on both spatial join and union task. Each row represents different ways of generating initial answers, either randomly or using heuristics. We report accuracies when heuristic hints or features are provided.

4o decreases by 47.9%, 17.8%, and 8.1%, respectively, from initial responses, indicating that heuristic hints alone are insufficient for effective revisions. **(2)** Review-and-refine with heuristic features is highly effective. Providing heuristic features significantly improves

poor initial responses and maintains or enhances strong answers. For the join task, poor initial responses (random, worst heuristics), the three models achieve average accuracy improvements of 36.4%, 35.9%, and 41.8%, respectively. When initial answers are strong, qwen-plus and 4o maintain or increase accuracy. Notably, 4o surpasses the best heuristic baseline accuracy (99%) by an additional 0.4%. For the spatial union task, when initial answers are poor, the three models improve average accuracy by 5.2%, 9.2%, and 7.7%.

We dive deeper into how the review-and-refine process makes decisions. We find out that: **(1)** Without features, the review-and-refine process aims to solve the task using computational geometry. With heuristic features included, the review-and-refine process systematically checks heuristic conditions against provided features, significantly reducing task difficulty and improving performance. Results demonstrate that LLMs perform well regardless of the initial answer quality, even outperforming the best heuristic. **(2)** Initial answers variably influence final results. Review-and-refine results remain relatively stable and consistent for 4o-mini (95.5%) and 4o (98.9%), indicating these models remain unaffected by initial answers. However, refined results from poor initial answers for qwen-plus (average 94.9%) are notably lower compared to results derived from strong initial answers (average 98.4%). A manual review of several correction examples reveals that qwen-plus typically considers overlap thresholds (`max_area`) between 0.1 and 0.6 sufficient for positive labels. However, with an initial negative answer (0), the review process sometimes judges a threshold around 0.25 as insufficient.

5.6 Ablation Studies

In this section, we present two ablation studies. In Section 5.6.1, we study spatial integration using vision-language models (VLMs), where visual information is incorporated as contextual input. In Section 5.6.2, we provide results from applying LLMs to four additional spatial analysis tasks, providing insights on LLMs’ generalization in the spatial context.

5.6.1 Spatial Integration With Vision-Language Models

In our core experiments, we relied solely on textual information extracted from GeoJSON files. However, visual information may also assist models in spatial integration tasks, as

models	spatial join			spatial union		
	plain	plain+img	features	plain	plain+img	features
qwen-vl	-	0.566	-	-	0.383	-
4o-mini	0.405	0.627	0.938	0.454	0.624	0.797
4o	0.580	0.728	0.948	0.817	0.889	0.955

Table 5.3: Plain prompting only uses textual information, while plain+img prompting presents an additional satellite image as visual context. Heuristic prompting with features still has the best performance, indicating the importance of geometrical information.

images provide direct cues about orientation and proximity. To explore this possibility, we conducted preliminary experiments using vision-language models (VLMs) on the two integration tasks. We evaluated one open-source model, {Qwen-2.5-VL-7B-Instruct (qwen-vl)}, and two API-based models, {gpt-4o-mini and gpt-4o} in a zero-shot setting, supplying only the plain task description with an image input. The results are summarized in Table 5.3. We observe two key findings: **(1)** Incorporating visual information improves performance on spatial integration tasks compared to text-only prompting, demonstrating the effectiveness of visual cues and **(2)** heuristic prompting using geometric features (features) still outperforms prompting with visual inputs. These results suggest that while visual context is beneficial, explicitly providing geometric features remains more effective for these tasks.

5.6.2 Diverse Spatial Reasoning Tasks

Beyond evaluating LLMs on the two core spatial integration tasks, we extend our analysis to a broader set of spatial reasoning tasks on real geometry objects. These additional experiments aim to provide more general insights into the spatial reasoning and computational capabilities of LLMs. We focus on four tasks that differ in geometry types, computational complexity, and reasoning logic, thereby broadening the evaluation scope while remaining grounded within the spatial analysis domain:

- **Point-to-Segment Distance (P2S):** To compute the shortest Euclidean distance between a point and a segment. It involves determining whether the point’s projection

Models	Point-to-Segment	Spatial Containment	Spatial Intersection	Convex Hull
qwen-plus	0.94	0.82	0.80	0.62
4o-mini	0.82	0.76	0.64	0.46
4o	0.94	0.84	0.66	0.64

Table 5.4: Comparative performance of LLMs on fundamental computational geometry tasks that require spatial reasoning abilities. Accuracies are reported.

falls onto the segment or if the distance is defined by one of the segment’s endpoints.

- **Spatial Containment (SC):** A fundamental Point-in-Polygon problem where the model determines whether a point lies strictly inside a triangle defined by three vertices. This tests the model’s ability to reason about interiority and bounded regions.
- **Spatial Intersection (SI):** This task requires checking if two 2D line segments intersect at any point. Unlike simple line intersection, this requires reasoning about the specific span of the segments and their relative orientations.
- **Convex Hull (CH):** Given a set of points, the model selects the minimal subset of points that forms a convex polygon containing all points in the set. This is a computationally complex task, requiring an understanding of global boundary structures.

Each task was evaluated on 50 synthetic geometry samples defined within a unit square. We tested the same API-based models qwen-plus, 4o-mini, and 4o, and report accuracy. For P2S, a tolerance of 1×10^{-3} was used for correctness. Results are shown in Table 5.4. We observe that the models perform well on simpler tasks (P2S: 90%, SC: 81%) but show notable performance drops on more complex tasks (SI: 70%, CH: 57%). A simple qualitative analysis of GPT-4o’s outputs revealed promising indications of spatial reasoning: the model consistently applied projection logic for P2S, ray casting for SC, orientation-based logic for SI, and referenced algorithms such as Graham’s scan or Jarvis March for CH. However, inaccurate computations still frequently led to incorrect results. These findings are consistent with our observations from the integration tasks: while LLMs demonstrate spatial reasoning capabilities that extend beyond basic integration, they still face challenges in executing the geometric computations required for general spatial analysis tasks.

5.7 Summary

This chapter improves urban data quality along the dimension of coherence by reconciling cross-source spatial datasets into a more unified representation. It studies two core integration tasks, spatial join and spatial union, and evaluates whether pre-trained large language models can support these tasks without requiring labeled training data. The results show that LLMs possess useful spatial reasoning abilities, but are unreliable at translating natural-language task descriptions into the precise computational geometry required for spatial matching. When provided with pre-computed geometric features, however, the models can infer context-sensitive decision rules and achieve strong performance, reaching peak accuracies of 98.4% on the join task and 96.0% on the union task. The proposed review-and-refine prompting strategy further improves weak initial answers while preserving strong ones, achieving up to 99.5% and 96.5% accuracy, respectively. Overall, the chapter shows that LLMs are most effective for spatial data integration not as standalone geometry solvers, but as flexible decision interfaces built on top of human-interpretable geometric features, thereby improving the coherence and practical utility of urban data.

However, using LLMs to integrate spatial data from different sources also introduces potential risks. (1) Safety risks from integration errors. If an LLM incorrectly joins or merges spatial records, it may create a misleading integrated dataset. These errors can be dangerous in high-stakes applications such as routing or navigation for users, where a wrong sidewalk-road relationship could lead to unsafe directions. (2) Unclear decision logic. LLM decisions can be difficult to audit or explain at scale, especially when compared with explicit rule-based systems. Without clear provenance and source traceability, an integrated dataset may appear coherent while hiding uncertainty or disagreement among the original sources. For these reasons, integrated outputs should be treated as preliminary integration results, not as automatic replacements for the original datasets. Deployment should preserve source information, document decision rules and model prompts, and include human review for high-stakes use cases.

5.8 Appendix

5.8.1 Geometrical Feature Distributions

In this section, we present the distributions of the geometric features, `min_angle`, `min_distance`, and `max_area`, in the training set for each task. We also show the accuracies of applying different threshold values, which helps us select appropriate thresholds for the test sets.

Feature Distributions For The Spatial Join Task

Parallel Heuristic (`min_angle`): As shown in Figure 5.5, positive pairs predominantly cluster at small angular differences. `min_angle=7` achieves a peak training accuracy of 93.76%, confirming its strength as a feature for the spatial join task.

Clearance Heuristic (`min_distance`): Most negative pairs yield smaller clearance values as they tend to overlap or intersect. Figure 5.6 shows that a `min_distance=3` threshold provides a modest classification accuracy of 86.57%.

Overlap Heuristic (`max_area`): Figure 5.7 identifies overlap as a poor heuristic for this task, given that positive and negative pairs exhibit similar distributions when `max_area` is above 0.2. The best training accuracy remains limited to 71.00 at `max_area=0.4`.

Feature Distributions For The Spatial Union Task

Parallel Heuristic (`min_angle`): it is less effective for the union task, as both classes share similar distributions (Figure 5.8). `min_angle=2` yields a training accuracy of 85.66%.

Overlap Heuristic (`max_area`): Figure 5.9 confirms overlap as a primary indicator of spatial union. While some negative pairs show significant overlap, positive instances cluster strongly at high percentage values, with `max_area=0.8` reaching 92.83% accuracy.

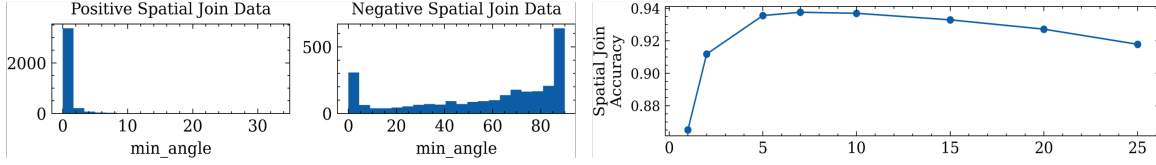


Figure 5.5: Left: Distribution of `min_angle` in positive and negative spatial join data. Right: train accuracy on different feature thresholds for `min_angle`, for spatial join.

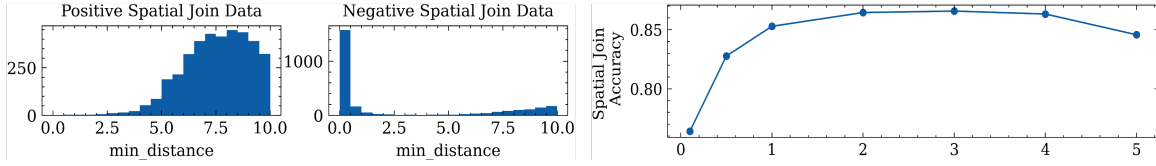


Figure 5.6: Left: Distribution of `min_distance` in positive and negative spatial join data. Right: train accuracy on different feature thresholds for `min_distance`, for spatial join.

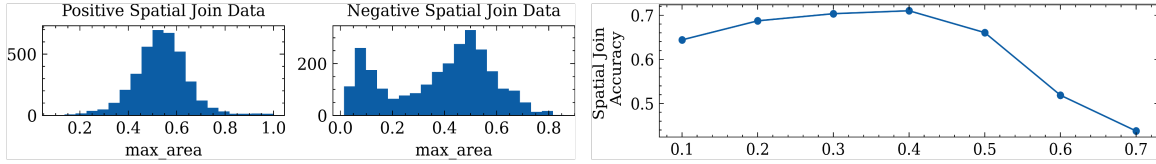


Figure 5.7: Left: Distribution of `max_area` in positive and negative spatial join data. Right: train accuracy using different feature thresholds for `max_area`, for spatial join.

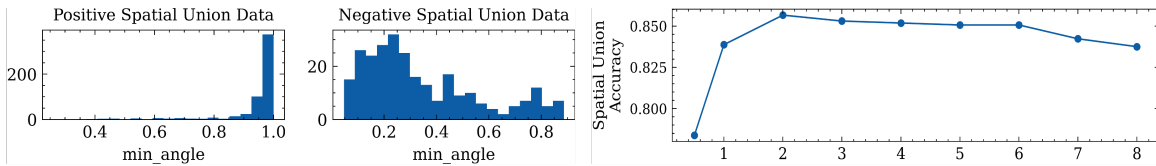


Figure 5.8: Left: Distribution of `min_angle` in positive and negative spatial union data. Right: train accuracy on different feature thresholds for `min_angle`, for spatial union.

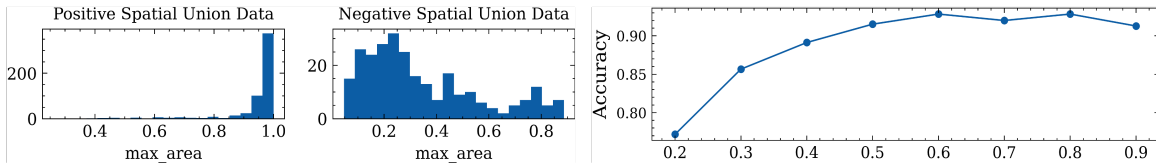


Figure 5.9: Left: Distribution of `max_area` in positive and negative spatial union data. Right: train accuracy using different feature thresholds for `max_area`, for spatial union.

5.8.2 Heuristic-Driven Prompting Results

prompting method	heuristic information	heuristic	models				
			llama3	mistral	4o-mini	qwen-plus	4o
zero-shot	plain	-	0.520	0.598	0.405	0.603	0.580
		parallel (p)	0.520	0.600	0.608	0.645	0.546
	hints	clearance (c)	0.518	0.599	0.516	0.609	0.394
		overlap (o)	0.528	0.599	0.596	0.603	0.549
		(p,c)	0.493	0.599	0.479	0.676	0.399
		(p,o)	0.505	0.600	0.458	0.669	0.413
		(c,o)	0.536	0.599	0.402	0.633	0.410
		(p,c,o)	0.556	0.599	0.400	0.705	0.406
	values	(p)	0.575	0.675	0.938	0.946	0.948
		(c)	0.596	0.599	0.742	0.732	0.670
		(o)	0.570	0.599	0.601	0.601	0.696
		(p,c)	0.523	0.599	0.877	0.950	0.948
		(p,o)	0.561	0.599	0.664	0.949	0.974
		(c,o)	0.551	0.604	0.734	0.733	0.529
		(p,c,o)	0.563	0.599	0.679	0.949	0.861
few-shot	plain	-	0.495	0.450	0.495	0.633	0.629
		(p)	0.472	0.522	0.589	0.682	0.628
	hints	(c)	0.526	0.668	0.512	0.679	0.589
		(o)	0.486	0.618	0.607	0.644	0.636
		(p,c)	0.516	0.579	0.548	0.663	0.606
		(p,o)	0.459	0.559	0.575	0.655	0.591
		(c,o)	0.474	0.652	0.590	0.654	0.483
		(p,c,o)	0.508	0.611	0.514	0.653	0.565
	values	(p)	0.606	0.867	0.929	0.936	0.931
		(c)	0.485	0.597	0.822	0.845	0.837
		(o)	0.505	0.657	0.760	0.687	0.727
		(p,c)	0.517	0.676	0.916	0.941	0.940
		(p,o)	0.479	0.618	0.858	0.968	0.959
		(c,o)	0.490	0.457	0.906	0.948	0.939
		(p,c,o)	0.539	0.625	0.872	0.984	0.967

Table 5.5: Accuracy of heuristic-driven prompting method for the spatial join task. We evaluate both zero-shot and few-shot prompting, with plain task description, heuristic-hints or heuristic features.

prompting method	heuristic information	heuristic	models				
			llama3	mistral	4o-mini	qwen-plus	4o
zero-shot	plain	-	0.436	0.366	0.454	0.799	0.817
	hints	parallel (p)	0.544	0.431	0.586	0.772	0.784
		overlap (o)	0.506	0.619	0.481	0.852	0.777
		(p,o)	0.476	0.624	0.414	0.752	0.524
	values	(p)	0.622	0.639	0.777	0.817	0.837
		(o)	0.549	0.579	0.744	0.865	0.862
		(p,o)	0.566	0.632	0.797	0.927	0.955
few-shot	plain	-	0.414	0.361	0.521	0.835	0.802
	hints	(p)	0.383	0.363	0.629	0.822	0.777
		(o)	0.391	0.421	0.647	0.827	0.779
		(p,o)	0.373	0.424	0.564	0.762	0.815
	values	(p)	0.491	0.792	0.769	0.835	0.820
		(o)	0.456	0.471	0.882	0.910	0.890
		(p,o)	0.466	0.779	0.805	0.960	0.882

Table 5.6: Accuracy of heuristic-driven prompting method for the spatial union task. We evaluate both zero-shot and few-shot prompting, with plain task description, heuristic-hints or heuristic features.

Chapter 6

ENHANCING URBAN DATA FIDELITY THROUGH PEDESTRIAN PATHWAY GENERATION

High-quality urban data must go beyond visual representation to achieve fidelity—the degree to which digital records faithfully mirror the functional connectivity and geometric nuances of the physical environment. In the context of pedestrian infrastructure, traditional segmentation-based methods often fail this test, producing fragmented or noisy graphs that appear visually correct but are topologically broken for navigation. This chapter addresses fidelity as the ultimate measure of urban data utility. We introduce TraversRL, a vision-conditioned iterative graph generator that simulates a traveler’s decision-making process to reconstruct pathway networks step-by-step. By optimizing the generator through reinforcement learning (RL) aligned with global graph outcomes, we demonstrate how to bridge the gap between pixel-level detection and traversable connectivity. Through evaluations across diverse urban settings, we show that TraversRL produces high-fidelity networks that are not only geometrically precise but also provide the robust connectivity required for accessible routing and urban planning. This chapter is based on our work currently under review.

6.1 Introduction

Accurate maps of pedestrian pathways are foundational for accessible navigation, mobility analysis, and urban planning. Yet these data are difficult to collect, and often incomplete or inaccurate, creating failure cases that disproportionately affect pedestrians and accessibility-critical users [304]. Inferring connected, traversible networks of sidewalks and crossings is a challenging vision problem: paths vary in width, shape, and material; they are frequently occluded by trees in overhead imagery; visual cues can be subtle or absent; and small geometric errors can induce large topological failures that degrade end-to-end routability (Fig. 6.1). Existing mapping pipelines rely heavily on segmentation: predicting pixel-level raster labels followed by post-processing. However, segmentation-based methods tend to



Figure 6.1: Qualitative examples of pedestrian pathway generation. Representative aerial intersection scenes illustrating common challenges for pedestrian mapping: {short segments, curvilinear segments, implicit paths (no crossing marks), occlusions from trees, and complex connectivity}. Top: ground-truth graph (blue; yellow dots mark vertices). Middle: Tile2Net prediction (red). Bottom: TraversRL-Local prediction (red). These examples highlight the topology-sensitive nature of pedestrian networks, and demonstrate TraversRL’s robust generation across diverse real-world challenges.

produce fragmented graphs that are unreliable for navigation, as paths may not be visible from overhead, and travelers may cross the street at unlabeled crosswalks.

An alternative approach is iterative graph construction, where a model repeatedly extends a partial network conditioned on local image information and the generated context. While successful for road extraction [11, 229, 274, 275], adapting these methods to pedestrian pathways introduces substantial new challenges. Pedestrian networks consist of many short and curvilinear segments; they can have weak or absent visual cues in overhead im-

agery (e.g., crossings); sidewalks are frequently occluded by trees or shadows. Moreover, urban scenes often contain clusters of intersections, so small geometric mistakes at junctions can induce large topological failures that directly harm routability. These properties amplify compounding rollout errors and make stepwise supervision a weak proxy for the global quality of the resulting network.

Reinforcement learning (RL) fine-tuning provides a principled way to improve long-horizon generation by optimizing outcome-level objectives rather than only next-step likelihood. In recent advances in large language model (LLM) reasoning, RL fine-tuning has been a key ingredient for enhancing multi-step problem solving abilities, particularly in settings where intermediate supervision is ambiguous but final outcomes are easily scored [104, 73, 232, 65, 266, 258]. This paradigm closely matches pathway generation — the metric of interest is defined on the final graph, is non-differentiable due to decoding and rendering, and success depends on many sequential decisions whose errors can compound.

In this chapter, we introduce **TraversRL**, a vision-conditioned generator that iteratively constructs pedestrian graphs from satellite imagery. TraversRL takes a local satellite crop and a generation canvas (encoding the partial graph) to predict the next network extension, using a discretized direction–distance action space tailored to short, fragmented pedestrian segments. We train TraversRL in two stages: **(1)** Supervised pretraining. The model learns core generation behavior by imitating ground-truth graph expansions step by step via cross-entropy loss, augmented with localization noise to improve robustness against drift during iterative rollouts. We denote this pretrained model as TraversRL-Pre. **(2)** RL fine-tuning. Initializing from TraversRL-Pre as a reference policy, we optimize outcome-level graph rewards under a KL-regularized objective, implemented via group-relative policy optimization (GRPO) [214, 73].

We investigate two RL reward designs. *(1)* Global reward uses a terminal, graph-level objective based on buffered geometry intersection-over-union (IoU), which aligns with final network quality. We denote this fine-tuned model as TraversRL-Global. *(2)* Local stepwise reward augments the terminal objective with a per-edge geometric quality signal (overlap efficiency). Local rewards shape rollout advantages to improve credit assignment and encourage locally precise geometry, yielding TraversRL-Local.

Overall, this chapter makes four primary contributions toward enhancing urban data fidelity through pedestrian pathway generation:

- Iterative pedestrian pathway generation. We introduce TraversRL, a vision-conditioned generator that iteratively extracts complex pathway graphs from aerial imagery, using a discretized direction–distance action space tailored to short pedestrian segments.
- RL fine-tuning for long-horizon graph generation. We demonstrate that iterative generation with both supervised pretraining and RL fine-tuning is a natural fit for pathway tracing and consistently improves graph quality. Averaged across three backbones and three datasets, TraversRL-Global outperforms a state-of-the-art segmentation-based method Tile2Net in IoU by 43.6% and a sidewalk connectivity metric by 205%.
- Local stepwise reward shaping for RL. We propose a local stepwise reward that provides a per-edge geometric quality signal for advantage shaping during RL fine-tuning. On average, TraversRL-Local further improves IoU by 2.6% and TravSim by 1.6% relative to TraversRL-Global.
- Backbone-agnostic gains and generalization. We show that RL fine-tuning improves performance across three backbones and generalizes zero-shot to unseen cities without additional training, outperforming a SoTA method trained on the target city. RL fine-tuning also yields cleaner graphs similar to those produced by human editors. In addition, we observe that RL fine-tuning yields cleaner, simpler graphs similar to those produced by human editors by reducing spurious branches and redundant edges while guaranteeing intersection connectivity for navigation.

6.2 Related Work

Street and Road Network Generation. Road and street network extraction from aerial imagery has been studied extensively. Existing methods broadly fall into two paradigms. Segmentation-first pipelines predict dense road masks and then recover graph structure through vectorization and post-processing [43, 92, 273, 89, 227, 314, 162, 173]. In contrast,

iterative graph construction grows a graph step-by-step by conditioning on local image evidence and the already-generated context. RoadTracer [11] introduced this crop-and-canvas tracing paradigm. VecRoad [229] improved local exploration at complex junctions, while RNGDet and RNGDet++ [274, 275] further developed transformer-based iterative graph detection with topology-oriented supervision. NETracer [148] proposed a topology-aware tracing framework by explicitly modeling node–edge connectivity.

While these approaches establish strong foundational techniques for extracting road graphs, they are not directly transferable to the pedestrian environment. Pedestrian pathways exhibit different challenges: they are typically narrower, contain many short and curvilinear segments, are more susceptible to occlusions (e.g., tree canopies), and require the explicit modeling of crossings and intersection connectivity for accessibility and navigation use cases [97, 305]. These inherent differences make pedestrian mapping more sensitive to rollout drift and localized geometric mispredictions during iterative generation.

Pedestrian Pathways From Aerial Imagery. Pedestrian pathway extraction from overhead imagery remains relatively understudied. Karimi et al. [117] explored pedestrian map generation at small scale and highlighted the importance of data quality. Tile2Net [97] detects sidewalks, crosswalks, and footpaths from aerial imagery and converts them into pedestrian networks. Prophet [304] combines road-network priors with aerial-image segmentation and rasterized street maps to infer pathway graphs (we do not compare with Prophet as neither the code nor trained model was available at the time of this writing). PathwayBench [305] argues that pixel-level metrics can understate mobility-critical errors and instead emphasizes routability and intersection-scale connectivity. Other work focuses on specific pedestrian assets rather than full connected networks, specifically zebra-crossing detection [1], crosswalk detection from satellite imagery [245, 14], and curb-ramp detection from street-view imagery [84, 187]. In contrast, our goal is direct generation of connected pedestrian pathway graphs.

Reinforcement Learning (RL) and GRPO. Reinforcement learning is increasingly used to improve long-horizon decision-making. Reasoning LLMs trained with RL show

strong gains with verifiable rewards (RLVR), where policies are optimized using outcome-level signals because intermediate supervision is ambiguous [104, 73, 232, 65, 266, 258]. More broadly, RL is a natural choice for optimizing non-differentiable trajectory-level objectives when errors can compound over many decisions. Among policy-gradient methods, PPO [211] is widely used for its stability. GRPO [214] is a PPO-style variant that replaces an explicit critic with group-relative baselines computed from multiple outputs, and typically uses KL regularization to a reference policy for stable updates.

These developments suggest RL is a natural fit for improving sequential generators when the desired objective is best expressed at the trajectory level. In pedestrian pathway generation, local geometric choices (e.g., sidewalk continuations) interact across many steps and must ultimately produce a globally coherent, routable network, while evaluation can be defined using geometry and connectivity aware graph metrics. We therefore adopt RL for fine-tuning and use GRPO as a practical, stable instantiation in our experiments.

6.3 Method

We propose TraversRL, a vision-conditioned generator that constructs pedestrian pathway graphs step-by-step from aerial imagery. TraversRL is trained in two stages — supervised pretraining (Sec. 6.3.1), followed by RL fine-tuning using either a global graph-level reward (Sec. 6.3.3) or RL fine-tuning with additional local stepwise reward (Sec. 6.3.4). The supervised pretraining phase yields a reference policy, denoted as TraversRL-Pre. During RL fine-tuning, the trainable policy π_θ is initialized from TraversRL-Pre while a frozen copy π_{θ_0} serves as the KL reference.

6.3.1 Supervised Pretraining of TraversRL (TraversRL-Pre)

Graph formulation. Given a satellite image of an urban region and vector annotations of pedestrian infrastructure, we convert the annotations into an undirected graph $\mathcal{G} = (V, E)$. Each node $v \in V$ is a 2D point in normalized coordinates $v = (x, y) \in [0, 1]^2$, and each edge $e = \{u, v\} \in E$ is a straight LineString segment connecting its endpoints. Our goal is to learn a policy that incrementally reconstructs E by predicting local edge, conditioned on both image evidence and the partial graph generated up to that point.

State: satellite crop + generation canvas. At each decision step, we expand from a current node location u and form a local observation centered at u , including: (1) an RGB satellite crop I_u , and (2) a single-channel generation canvas C_u that rasterizes the partial generated graph. The canvas renders all previously generated edges and marks the current node (as a filled dot). Both I_u and C_u are extracted using a fixed normalized window (e.g., 0.5×0.5 of the full extent) and resized to a fixed input resolution (256×256 ; 224×224 for ViT backbones). We concatenate channels to obtain a 4-channel tensor $X_u = \text{concat}(I_u, C_u) \in \mathbb{R}^{4 \times H \times W}$, so that each decision is explicitly conditioned on local appearance and the evolving topology encoded by the canvas.

Action space: discretized direction and distance. TraversRL predicts a discrete movement action from u . Given a ground-truth neighbor v of u , the displacement $\Delta = v - u$ is mapped to a class label by discretizing both angle and distance. We use $N_\theta = 36$ angle bins (10° each) and $N_r = 10$ distance bins uniformly spaced over $[0, 0.2]$ in normalized coordinates. This yields $|\mathcal{A}| = 360$ movement classes plus a STOP class. We write $a \in \mathcal{A}$ with $a = \text{STOP}$ indicating that no ungenerated edges remain incident to the current node.

Policy network. TraversRL uses an ImageNet-pretrained backbone $f_\phi(\cdot)$ adapted to accept 4-channel inputs. We replace the first convolution with a 4-channel variant by copying the pretrained RGB weights and initializing the additional (canvas) channel to zero. Given X_u , the backbone produces a feature vector $h \in \mathbb{R}^d$, which is mapped to action probabilities by an MLP head:

$$\pi_\theta(a \mid X_u) = \text{softmax}(W_2 \sigma(W_1 \text{Dropout}(h))) \quad (6.1)$$

where $\sigma(\cdot)$ is ReLU and $\pi_\theta(\cdot \mid X_u) \in \mathbb{R}^{|\mathcal{A}|}$.

Supervised training. For each training graph, we supervise a sequence of actions at each node: the policy emits one action per incident ground-truth edge, then emits STOP and moves to the next node (Fig. 6.2). To improve robustness to rollout drift and mitigate error accumulation, we inject localization noise by perturbing the query location while

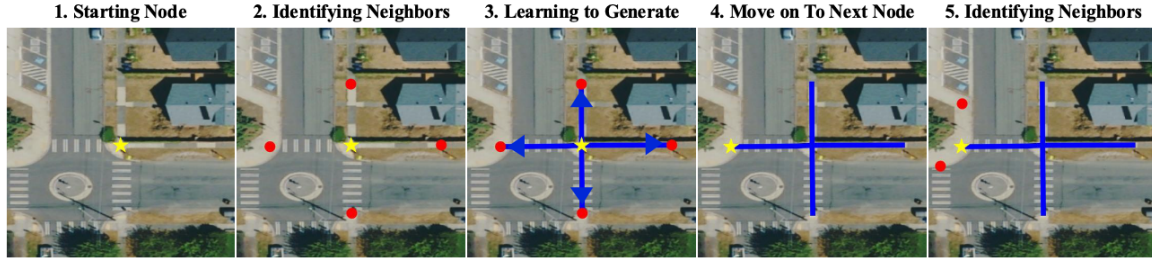


Figure 6.2: **Supervised node expansion in TraversRL pretraining.** Given the current node (yellow star), the set of ground-truth neighbors (red) are determined. During supervised pretraining, the policy is trained to sequentially emit one direction–distance action per ground-truth edge (blue arrows), adding the corresponding edges to the canvas. After all edges at the current node have been generated, the policy emits STOP and the algorithm moves to the next node for expansion (steps 1–5).

following the ground-truth edge:

$$\tilde{u} = u + \epsilon, \quad \epsilon \sim \mathcal{U}([- \eta, \eta]^2). \quad (6.2)$$

We extract crops centered at $\tilde{u}$ and compute the target class using the perturbed displacement $\Delta = v - \tilde{u}$, aligning supervision with the model’s observation under drift. We minimize cross-entropy over the supervised action sequence. The supervised training approach is adapted from a similar technique for road extraction [11], but includes variable distances and an architecture to adapt to topologically complex, occluded, high-density sidewalk paths.

6.3.2 Inference and Rollout Hyperparameters

At inference time, we roll out π_θ to construct a graph in continuous space. We initialize a start node (for training and evaluation, a random node in the ground-truth graph; in practice, a point selected by a user or based on segmentation pre-processing) and maintain a stack of active nodes. For each popped node u , we repeatedly call the policy until it emits STOP or reaches a per-node expansion budget. Each non-STOP action is decoded into a displacement using the corresponding angle–distance bin midpoints, creating a new node at

$u + \Delta$ (clipped to $[0, 1]^2$). We add the edge, render it onto the global canvas, and push the new node onto the stack. To avoid trivial cycling, we disallow executing the same discrete action twice from the same node within one expansion. Because backbones can differ in termination behaviors, we tune a small set of rollout hyperparameters on the validation set. We configure the following:

1. *Top-k backoff* (k): enumerate the k highest-logit actions and execute the first geometrically valid action. $k = 1$ equates to standard greedy decoding.
2. *Restarts per graph* (R): run the full rollout R times from independently sampled start nodes and retain the best result under an internal confidence score (defined as the sum of top-1 minus top-2 logit margins across all steps).
3. *STOP THRESHOLD* (τ_{stop}): accept STOP only when $\pi_{\theta}(\text{STOP} \mid s_t) \geq \tau_{\text{stop}}$.
4. *Maximum per-node expansions* (M): cap on the total number of actions permitted when expanding any single node.

We select the configuration that maximizes buffered-geometry IoU on the validation set, and reuse the resulting configuration for all subsequent inference and RL fine-tuning.

6.3.3 RL Fine-Tuning with Global Graph Reward (TraversRL-Global)

Motivation. Supervised pretraining optimizes per-step classification accuracy, but TraversRL is evaluated via an iterative rollout whose quality is judged globally. Small angle or distance-bin errors can compound over many steps, producing large geometric and topological deviations in the final graph. Moreover, our target evaluation metrics are non-differentiable with respect to network parameters because they depend on decoding and rendering an entire trajectory. These properties motivate RL fine-tuning, which directly optimizes an outcome-level, graph-level objective.

RL with terminal graph reward. We formalize TraversRL as a policy π_{θ} over actions $a_t \in \mathcal{A}$, initialize π_{θ} from TraversRL-Pre, and keep a frozen reference policy π_{θ_0} . Fine-tuning with a terminal graph reward yields TraversRL-Global. For each training sample, we draw a group of K rollouts $\{\tau_i\}_{i=1}^K$ from π_{θ} using the same decoding configuration as inference. Each trajectory $\tau_i = (a_{i,1}, \dots, a_{i,T_i})$ induces a predicted edge set $\hat{E}_i$ and receives

a terminal reward $R_i = R(E, \hat{E}_i)$, where $R(\cdot, \cdot)$ is the buffered-IoU objective computed on the final generated graph. We then compute standardized group-relative advantages:

$$\bar{R} = \frac{1}{K} \sum_{i=1}^K R_i, \quad A_i = \frac{R_i - \bar{R}}{\text{std}(\{R_j\}_{j=1}^K) + \epsilon}. \quad (6.3)$$

We update π_θ to increase the likelihood of actions from higher-advantage rollouts while constraining drift from the frozen reference π_{θ_0} via a KL penalty. Concretely, our RL fine-tuning objective is defined as:

$$\begin{aligned} \mathcal{L}_{\text{global}}(\theta) = & -\frac{1}{K} \sum_{i=1}^K \frac{A_i}{T_i} \sum_{t=1}^{T_i} \log \pi_\theta(a_{i,t} \mid s_{i,t}) \\ & + \beta \cdot \frac{1}{K} \sum_{i=1}^K \frac{1}{T_i} \sum_{t=1}^{T_i} D_{\text{KL}}(\pi_\theta(\cdot \mid s_{i,t}) \parallel \pi_{\theta_0}(\cdot \mid s_{i,t})) \end{aligned} \quad (6.4)$$

where $s_{i,t}$ denotes the rollout state at step t (i.e., the local image crop and canvas crop centered at the current node), β is the KL coefficient, and the $1/T_i$ factors normalize trajectories of different lengths. We compute the full-distribution KL divergence at each visited state using the policy and reference logits.

6.3.4 RL Fine-Tuning with Local Stepwise Reward (TraversRL-Local)

Motivation. TraversRL-Global aligns training with the final objective, but it assigns a single rollout-level advantage A_i to all actions within a trajectory, leading to coarse credit assignment. In practice, a rollout often contains a mixture of locally correct and incorrect steps, yet all steps are reinforced (or penalized) uniformly. As a result, terminal-only RL can improve global overlap while under-optimizing fine-grained geometric choices (e.g., selecting a nearby but incorrect direction bin that yields similar buffered overlap).

Local stepwise reward. To inject a localized training signal without discarding the stable terminal objective, we define a stepwise reward that quantifies the local geometric quality of each newly added edge and use it for advantage shaping. Let P_t be the buffered union of predicted edges up to step t and let G be the buffered union of ground-truth edges. Define predicted area $A_t = \text{area}(P_t)$ and overlap area $I_t = \text{area}(P_t \cap G)$. With a new edge,

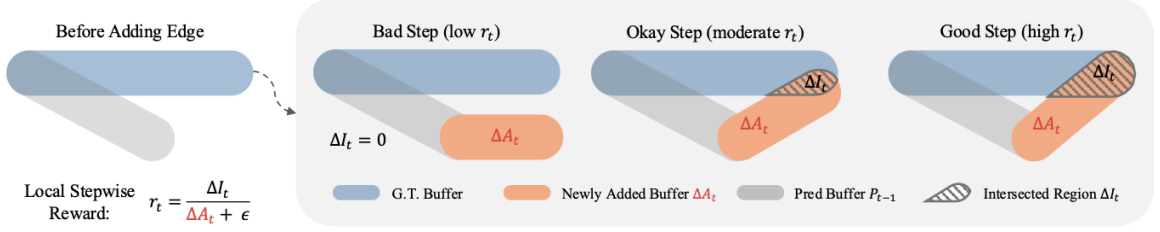


Figure 6.3: Local stepwise reward. We compare the buffered ground-truth region G (blue) with the current predicted buffer P_{t-1} (gray). Adding a new edge at step t contributes newly added buffered area ΔA_t (orange), of which the overlapped portion with ground-truth is ΔI_t . We define the stepwise reward $r_t = \Delta I_t / (\Delta A_t + \epsilon)$: it is high when the added edge aligns better with ground-truth (good step), moderate under partial alignment, and near zero when the edge adds off-target geometry ($\Delta I_t = 0$).

we compute $\Delta A_t = A_t - A_{t-1}$ and $\Delta I_t = I_t - I_{t-1}$, formulating the stepwise reward as:

$$r_t = \frac{\Delta I_t}{\Delta A_t + \epsilon}, \quad (6.5)$$

with $r_t = 0$ for non-edge actions (e.g., STOP). Intuitively, r_t measures what fraction of newly generated buffered area contributes to ground-truth overlap; edges that add mostly off-target geometry receive near-zero reward (Fig 6.3).

Additive advantage shaping. As with global terminal reward, we sample K rollouts and compute the terminal advantages A_i from the final rewards R_i . We then calculate and normalize the stepwise rewards within each individual rollout:

$$\tilde{r}_{i,t} = \frac{r_{i,t} - \mu_i}{\sigma_i + \epsilon}, \quad \mu_i = \frac{1}{T_i} \sum_{t=1}^{T_i} r_{i,t}, \quad \sigma_i = \text{std}(\{r_{i,t}\}_{t=1}^{T_i}), \quad (6.6)$$

We clip $\tilde{r}_{i,t}$ for stability and define an additively shaped stepwise advantage:

$$\hat{A}_{i,t} = A_i + \lambda \tilde{r}_{i,t}, \quad (6.7)$$

where λ controls the strength of shaping. The additive formulation preserves the outcome-level signal, while simultaneously encouraging locally high-quality steps even within imperfect rollouts. Since both A_i and $\tilde{r}_{i,t}$ are standardized, their magnitudes are comparable,

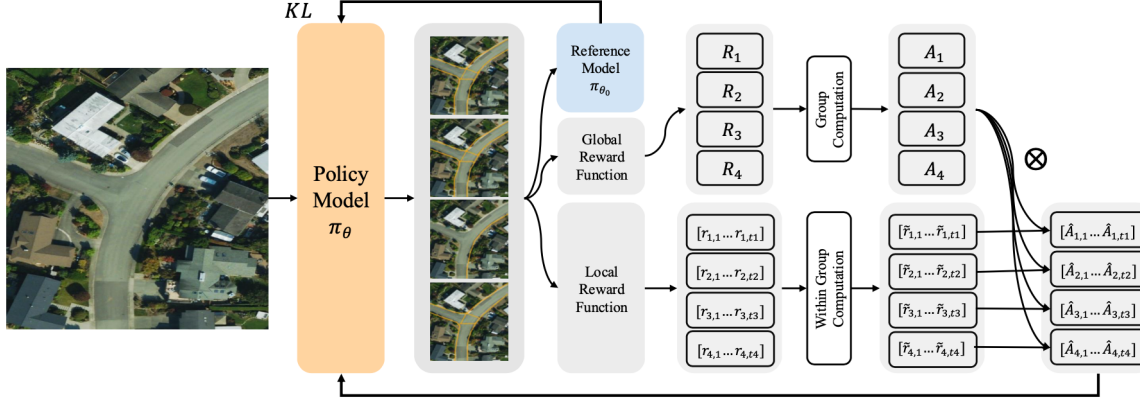


Figure 6.4: RL fine-tuning with local stepwise reward shaping (TraversRL-Local). For each intersection, we sample K rollouts (here $K = 4$) from π_θ , producing action sequences and generated graphs. A global reward assigns each rollout a terminal reward R_i , which is converted into group-relative advantages $\{A_i\}_{i=1}^K$. In parallel, a local reward computes per-step rewards $\{r_{i,t}\}$, which are normalized within each rollout to obtain $\{\tilde{r}_{i,t}\}$ and combined with A_i to form shaped advantages $\hat{A}_{i,t} = A_i + \lambda \tilde{r}_{i,t}$. The policy is updated using $\hat{A}_{i,t}$ together with $\text{KL}(\pi_\theta \parallel \pi_{\theta_0})$; only π_θ is updated, while π_{θ_0} is frozen. Without local shaping (TraversRL-Global), the update reduces to $\hat{A}_{i,t} = A_i$.

making the shaping term well-scaled in practice. We then substitute the uniform rollout-level weighting in Eq. (6.4) with $\hat{A}_{i,t}$ while keeping the same KL regularizer:

$$\begin{aligned} \mathcal{L}_{\text{local}}(\theta) = & -\frac{1}{K} \sum_{i=1}^K \frac{1}{T_i} \sum_{t=1}^{T_i} \hat{A}_{i,t} \log \pi_\theta(a_{i,t} \mid s_{i,t}) \\ & + \beta \cdot \frac{1}{K} \sum_{i=1}^K \frac{1}{T_i} \sum_{t=1}^{T_i} D_{\text{KL}}(\pi_\theta(\cdot \mid s_{i,t}) \parallel \pi_{\theta_0}(\cdot \mid s_{i,t})) \end{aligned} \quad (6.8)$$

This procedure yields TraversRL-Local and is visually demonstrated in Fig. 6.4.

Note: A natural alternative is to replace the terminal advantage with stepwise advantages normalized across rollouts at each step. However, we found this to be empirically unstable in our task. Step rewards are highly sensitive to small geometric perturbations and vary substantially across rollouts and step indices due to different rollout lengths and explo-

ration paths. This high variance makes cross-rollout, per-step normalization to be noisy. In practice, using purely stepwise advantages causes rapid drift from the reference policy (KL increases sharply early in training) and leads to collapse, whereas advantage shaping preserves the stability of terminal-reward RL while still providing informative local credit.

6.4 Experiment

We evaluate TraversRL in both in-domain and zero-shot transfer settings. Sec. 6.4.1 details the evaluation datasets, Sec. 6.4.2 outlines the visual backbones and baseline models, and Sec. 6.4.3 specifies the training and evaluation protocols.

6.4.1 Datasets

We evaluate across three urban-intersection pathway datasets:

- **WashingtonInter (in-domain)**. A curated dataset of three- and four-way urban intersections from 13 cities in Washington State. We first extract intersection bounding boxes from OpenStreetMap and prune overlapping regions to avoid spatial train–test leakage. We then retrieve the corresponding pedestrian pathway annotations from our internal database and discard scenes whose ground-truth graph is disconnected. After a final manual visual inspection to remove severe aerial-to-map misalignments, the dataset contains 2,287 training scenes, 144 validation scenes, and 426 test scenes.
- **PathwayBench Seattle (zero-shot)** [305]. A benchmark for pathway graph generation at urban intersections. We evaluate 342 intersection scenes from Seattle. Since PathwayBench does not provide training/validation splits, we evaluate in a *zero-shot* setting: models are trained on WashingtonInter and applied directly to PathwayBench without fine-tuning
- **PathwayBench Washington, D.C. (zero-shot)** [305]. 589 intersection scenes from Washington, D.C., evaluated under the same zero-shot setting.

Although WashingtonInter includes intersections from Seattle, PathwayBench uses different scene extents and spatial resolutions (zoom level of roughly 19, compared to approximately 20 in our dataset). Therefore, we refer to all PathwayBench evaluations as zero-shot transfer.

6.4.2 Backbones and Baseline

We instantiate TraversRL with three visual backbones: ResNet-18 [87], ViT-S [50], and Swin-T [158]. All are adapted to 4-channel inputs by modifying the first projection layer to accept an additional canvas channel while retaining pretrained RGB weights (the extra channel is initialized to zero). ViT uses 224×224 inputs, while ResNet and Swin use 256×256 . We compare against Tile2Net [97], a state-of-the-art pipeline that predicts sidewalks and crosswalks from aerial imagery and converts them into pedestrian networks.

6.4.3 Training & Evaluation

Supervised pretraining. For each backbone, we train a reference model following Sec. 6.3.1. Unless otherwise noted, we apply localization error with $\eta = 0.02$ (in normalized coordinates), and optimize with AdamW (learning rate 1×10^{-4} , weight decay 1×10^{-4} , gradient clipping at 1.0). Models are trained for 50 epochs with gradient accumulation over 16 node-expansions, with max_steps=200. For ResNet-18, BatchNorm statistics are frozen. We select the best checkpoint by validation loss and use it to initialize RL fine-tuning.

RL fine-tuning. We fine-tune each pretrained checkpoint using GRPO (Secs. 6.3.3, 6.3.4). Unless otherwise noted, we use AdamW (learning rate 3×10^{-5} , weight decay 1×10^{-4} , gradient clipping at 1.0), train for 15 epochs with gradient accumulation over 8 graphs, and set the KL coefficient to $\beta = 0.05$. For each training graph, we sample a group of $K = 4$ rollouts, capping the maximum rollout length at 200 steps. Rewards are computed using a buffer radius $r = 0.01$ (normalized units, approximately 1 meter). For the local shaping variant, we set $\lambda = 0.1$ and clip $\tilde{r}_{i,t}$ to be within $[-3, 3]$ for stability. We select the best RL checkpoint by validation buffered IoU.

Metrics. We evaluate each method over five randomly sampled start nodes and report averages. We use two metrics: (1) *Buffered geometry IoU*, computed between the predicted and ground-truth edge sets by buffering each polyline with a 1-meter radius and calculating the IoU of the resulting regions, and (2) *Traversability similarity (TravSim)* [305], a routability-oriented metric that compares boundary-to-boundary connectivity within local tiles. For each tile, we treat the boundary as a set of sides and identify terminating nodes where the graph intersects the boundary. A boundary pair is considered traversable if a valid path exists within the clipped tile connecting a terminating node on one boundary to a terminating node on another. Each tile thus induces a set of traversable boundary pairs for both prediction and ground-truth, and TravSim is the Jaccard similarity between these two sets, averaged over tiles. This metric does not require node/edge correspondence and directly measures whether the generated graph affords the same traversal possibilities as the ground-truth within intersection-scale regions.

6.5 Results

6.5.1 Supervised Policies Strongly Outperform the Baseline

Table 6.1 compares pretrained checkpoints (TraversRL-Pre) against the Tile2Net across three datasets. The results show a clear pattern: Even prior to RL fine-tuning, directly learning an iterative graph generator yields substantially higher geometric overlap and connectivity. We observe that: **(1) WashingtonInter (in-domain).** Tile2Net achieves 0.264 IoU and 0.114 TravSim. In contrast, TraversRL-Pre improves both metrics for every backbone, reaching 0.531 IoU and 0.605 TravSim with Swin-T. Averaged across backbones, TraversRL-Pre exceeds Tile2Net by 0.258 IoU and 0.481 TravSim. The particularly large gain in TravSim suggests that the iterative formulation better preserves intersection topology than raster-to-network post-processing, where small geometric artifacts can disconnect crossings and sidewalk continuations. **(2) PathwayBench (zero-shot transfer).** The same trend holds under distribution shift. On Seattle data, TraversRL-Pre outperforms Tile2Net by an average of 0.026 IoU and 0.215 TravSim. On D.C. data, TraversRL-Pre improves TravSim by 0.196 on average, while achieving competitive IoU: Swin-T exceeds

	Model	Stage	WashingtonInter		Seattle (PB)		D.C. (PB)	
			IoU	TravSim	IoU	TravSim	IoU	TravSim
	Tile2Net	–	0.264 $\pm$ 0.046	0.114 $\pm$ 0.017	0.169 $\pm$ 0.021	0.175 $\pm$ 0.037	0.226 $\pm$ 0.018	0.317 $\pm$ 0.023
TraversRL	ResNet18	pre	0.510 $\pm$ 0.003	0.586 $\pm$ 0.005	0.186 $\pm$ 0.003	0.386 $\pm$ 0.003	0.212 $\pm$ 0.003	0.506 $\pm$ 0.004
		global	0.539 $\uparrow$ $\pm$ 0.001	0.591 $\uparrow$ $\pm$ 0.008	0.189 $\uparrow$ $\pm$ 0.002	0.387 $\uparrow$ $\pm$ 0.004	0.223 $\pm$ 0.002	0.507 $\uparrow$ $\pm$ 0.003
		local	0.540 $\uparrow$ $\pm$ 0.002	0.605 $\uparrow$ $\pm$ 0.004	0.199 $\uparrow$ $\pm$ 0.003	0.403 $\uparrow$ $\pm$ 0.002	0.227 $\uparrow$ $\pm$ 0.002	0.513 $\uparrow$ $\pm$ 0.002
	ViT-S	pre	0.525 $\pm$ 0.003	0.593 $\pm$ 0.005	0.194 $\pm$ 0.003	0.382 $\pm$ 0.006	0.208 $\pm$ 0.002	0.499 $\pm$ 0.002
		global	0.548 $\uparrow$ $\pm$ 0.003	0.597 $\uparrow$ $\pm$ 0.003	0.211 $\uparrow$ $\pm$ 0.002	0.401 $\uparrow$ $\pm$ 0.005	0.226 $\uparrow$ $\pm$ 0.001	0.506 $\uparrow$ $\pm$ 0.004
		local	0.548 $\uparrow$ $\pm$ 0.001	0.605 $\uparrow$ $\pm$ 0.005	0.214 $\uparrow$ $\pm$ 0.003	0.411 $\uparrow$ $\pm$ 0.003	0.221 $\pm$ 0.002	0.515 $\uparrow$ $\pm$ 0.004
	Swin-T	pre	0.531 $\pm$ 0.004	0.605 $\pm$ 0.006	0.205 $\pm$ 0.004	0.401 $\pm$ 0.004	0.231 $\pm$ 0.002	0.535 $\pm$ 0.004
		global	0.559 $\uparrow$ $\pm$ 0.002	0.607 $\uparrow$ $\pm$ 0.004	0.218 $\uparrow$ $\pm$ 0.001	0.416 $\uparrow$ $\pm$ 0.004	0.236 $\uparrow$ $\pm$ 0.001	0.535 $\pm$ 0.004
		local	0.584 $\uparrow$ $\pm$ 0.003	0.611 $\uparrow$ $\pm$ 0.008	0.230 $\uparrow$ $\pm$ 0.005	0.418 $\uparrow$ $\pm$ 0.007	0.252 $\uparrow$ $\pm$ 0.003	0.535 $\pm$ 0.003

Table 6.1: Quantitative results on WashingtonInter (in-domain) and PathwayBench test cities (zero-shot transfer), reported as mean $\pm$ std over five seeds. *pre*, *global*, and *local* denote supervised pretraining, RL fine-tuning with a terminal graph-level reward, and RL fine-tuning with additional local stepwise reward shaping, respectively. **Bold** indicates the best value within each dataset/metric column. $\uparrow$ denotes an improvement over both Tile2Net and the corresponding *pre* checkpoint (same backbone). * denotes an improvement over the corresponding *global* checkpoint (same backbone).

Tile2Net in IoU (0.231 vs. 0.226), whereas ResNet-18 and ViT-S fall marginally behind. Overall, these results suggest that cross-city transfer is more robust for connectivity.

6.5.2 RL Fine-Tuning Consistently Improves Performance Across Backbones and Datasets

Table 6.1 shows that outcome-level RL fine-tuning improves TraversRL across backbones and evaluation settings, even though TraversRL-Pre is already strong. Importantly, under the evaluation with a strict 1m buffer, IoU is substantially more sensitive to small geometric misalignments, making relative gains more diagnostic of true geometric precision.

Global reward improves graph quality. With a terminal graph-level reward, TraversRL-Global improves IoU for every backbone on every dataset over TraversRL-Pre. Averaged across backbones, the relative IoU gains are 5.1% on WashingtonInter, 5.6% on Seattle, and 5.2% on D.C. TravSim also increases by 0.6%, 3.0%, and 0.5%, respectively. These consistent trends indicate that optimizing a terminal objective aligns the policy toward decisions that improve both geometric overlap and connectivity patterns.

Local reward shaping adds further gains. Relative to TraversRL-Global, TraversRL-Local yields additional IoU gains of 1.6% (WashingtonInter), 4.1% (Seattle), and 2.1% (D.C.), with one minor exception (ViT-S on D.C.). Relative to TraversRL-Pre, these gains surge to 6.7%, 9.8%, and 7.5%. Local shaping also consistently improves TravSim, by 1.5% on WashingtonInter, 2.4% on Seattle, and 1.0% on D.C. Overall, these results validate the intended mechanism of the local reward: it provides finer credit assignment that encourages locally aligned edge placement (and discourages off-target geometry) while preserving the stability of terminal-reward optimization. Additionally, the standard deviations in Table 6.1 are generally small, indicating stable training across five random seeds. In most settings, the RL gains in IoU are comparable to, or larger than, the seed-to-seed variation, indicating that the gains are not an artifact of favorable random seed but reflect a systematic improvement in rollout-time graph quality from outcome-level RL objectives.

Buffer-size ablation: stricter buffers reveal larger RL gains. We re-evaluated the fixed predictions using wider geometric buffers. On WashingtonInter, averaged across backbones and seeds, TraversRL-Global improves IoU over TraversRL-Pre by 2.1% at 4m, which grows to 3.1% at 2m and 5.2% at 1m. The same pattern holds for TraversRL-Local, whose improvements increase by 2.5%, 4.0% and 6.8% respectively. These observations show that outcome-level RL fine-tuning improves geometric precision in a way that becomes increasingly visible as the overlap criterion tightens, with local shaping providing the strongest improvements under strict alignment. The effect is strongest for local shaping: relative to global-reward RL, TraversRL-Local gains widen as the buffer shrinks, consistent with the reward encouraging locally aligned edge extensions and discouraging off-target geometry.

Dataset	AvgDeg↓			TravSim↑		
	Pre	Global (Δ)	Local (Δ)	Pre	Global (Δ)	Local (Δ)
WashingtonInter	0.429	0.416 (-3.0%)	0.407 (-5.1%)	0.595	0.599 (+0.6%)	0.607 (+2.0%)
Seattle (PB)	0.574	0.541 (-5.7%)	0.519 (-9.6%)	0.389	0.401 (+3.1%)	0.411 (+5.7%)
D.C. (PB)	0.553	0.528 (-4.5%)	0.520 (-6.0%)	0.513	0.516 (+0.6%)	0.521 (+1.6%)

Table 6.2: Graph cleanliness and connectivity agreement averaged over backbones and five seeds. AvgDeg denotes average node degree (lower indicates sparser graphs), and TravSim denotes traversability similarity (higher indicates closer agreement with ground-truth boundary connectivity). For Global and Local, we report the relative change compared to Pre on the same dataset.

TraversRL-Local recovers subtle missing paths. Recall that TravSim compares the set of traversable boundary-to-boundary pairs induced by the predicted vs. ground-truth graphs within each tile (Sec. 6.4.3). Let S_{pred} and S_{gt} denote these per-tile sets of traversable boundary-pair relations for the prediction and ground-truth, respectively. We decompose their disagreement into: $\text{TP} = |S_{\text{pred}} \cap S_{\text{gt}}|$, $\text{FP} = |S_{\text{pred}}| - \text{TP}$, and $\text{FN} = |S_{\text{gt}}| - \text{TP}$. On WashingtonInter, averaging per-tile set sizes over tiles and then over seeds/backbones, TraversRL-Pre predicts approximately 1.86 boundary-pair relations per tile, with $\text{TP} \approx 1.55$, $\text{FP} \approx 0.31$, and $\text{FN} \approx 1.09$ (i.e., missed connectivity relations dominate spurious ones by $\sim 3.5\times$). RL fine-tuning primarily reduces this missed-connectivity term: TraversRL-Local increases TP from $1.55 \rightarrow 1.60$ (+3.5%) and reduces FN from $1.09 \rightarrow 1.04$ (-5.0%). Consistent with this, at the scene level TravSim improves in 59% of intersections; among those improved scenes, 92% exhibit increased TP and 91% exhibit reduced FN. Overall, RL improves TravSim primarily by recovering missing ground-truth paths rather than pruning.

6.5.3 TraversRL Produces Cleaner, Sparser Graphs While Preserving Connectivity.

Beyond buffered IoU, we analyze how RL changes graph structure. Following Pathway-Bench, we compute all graph statistics within each tile on the clipped predicted graph and

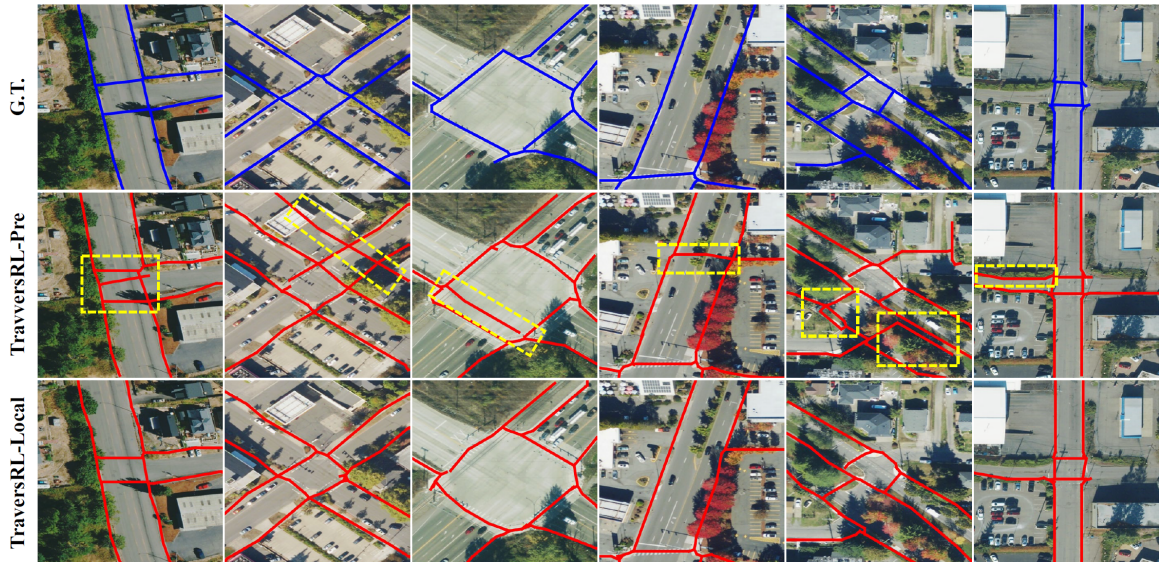


Figure 6.5: RL fine-tuning produces cleaner, sparser graphs. Qualitative examples comparing the ground-truth graph (top, blue) to predictions from TraversRL-Pre (middle, red, “messy”) and TraversRL-Local (bottom, red, “cleaner”). Yellow dashed boxes highlight artifacts of the supervised policy — redundant branches, spurious short connectors, and noisy junction geometry —that are pruned after RL fine-tuning.

then average over tiles. We report: (1) average node degree (AvgDeg) as a proxy for graph density (lower typically indicates fewer branches) and (2) traversability similarity (TravSim) as a proxy for whether the predicted graph preserves ground-truth connectivity. AvgDeg alone cannot distinguish “clean pruning” from simply missing edges; we therefore interpret it jointly with TravSim, which is sensitive to missing or incorrectly connected links.

We observe that RL reduces graph density while improving connectivity agreement. A consistent pattern emerges across all datasets (Table 6.2): RL fine-tuning yields sparser graphs while maintaining or improving connectivity. Averaged across backbones, TraversRL-Global reduces AvgDeg relative to TraversRL-Pre by 3.0% on WashingtonInter, 5.7% on Seattle, and 4.5% on D.C., while increasing TravSim by 0.6%, 3.1%, and 0.6%, respectively. TraversRL-Local strengthens this trend, further reducing AvgDeg (5.1%, 9.6%, 6.0%) while improving TravSim (2.0%, 5.7%, 1.6%). Fig. 6.5 shows the same effect qualitatively:

RL suppresses redundant edges and noisy branches while preserving intersection structure. Overall, RL improves network utility not by adding more edges, but by making more selective branching and stopping decisions that prune low-value extensions while preserving (and often strengthening) traversability-relevant connectivity.

6.6 Summary

This chapter improves urban data fidelity by generating pedestrian pathway graphs that accurately mirror real-world geometry and connectivity. It introduces TraversRL, a vision-conditioned generator that constructs networks from aerial imagery using a specialized direction-distance action space. By employing reinforcement learning alongside supervised pretraining, the model optimizes long-horizon graph generation through outcome-level rewards that bypass the limitations of imperfect intermediate supervision. TraversRL consistently exceeds the performance of segmentation-based baselines, with RL fine-tuning significantly enhancing both geometric IoU and traversability. While global rewards elevate overall graph quality across diverse settings, local stepwise rewards refine edge placement and recover subtle missing paths. Additionally, RL produces cleaner, sparser graphs by pruning redundant branches without compromising traversability. Ultimately, this chapter proves that high-quality pathway generation depends on reconstructing the functional network structure required for navigation, rather than merely identifying visible surfaces.

The generation of high-fidelity pathway graphs also introduces several potential risks. (1) Navigation and accessibility failures. Hallucinated sidewalk links or crossing segments can lead routing systems to recommend unsafe or impossible paths. Conversely, missing curb connections or walkway segments may remove genuinely traversable areas from the dataset and make accessibility appear worse than it actually is. (2) Informal paths mistaken as official infrastructure. The model may incorrectly treat desire paths, or informal trails created by repeated foot traffic, as official pedestrian infrastructure. Presenting these paths as legitimate sidewalks without validation could create legal, safety, and liability risks for planning agencies. Given these risks, generated pathway graphs should be treated as preliminary outputs, not authoritative records. Future deployment should include targeted validation procedures for safety-critical and accessibility-critical systems.

Chapter 7

CONCLUSION

High-quality urban data are fundamental to urban analysis, planning, and decision-making. Cities are increasingly studied through spatio-temporal records, aerial imagery, and graph-structured infrastructure data. Yet the usefulness of these resources depends not only on their availability, but also on their quality. In practice, urban datasets are often too coarse, incomplete, inconsistent across sources, or insufficiently faithful to the real-world systems they are meant to describe. These limitations constrain what can be measured, modeled, integrated, and acted upon by practitioners. This dissertation has argued that high-quality urban data are a prerequisite for reliable and equitable urban analytics, and that improving urban data quality requires targeted methods designed around the specific deficiencies, structures, and intended uses of the data. It studies four complementary dimensions of urban data quality: granularity, coverage, coherence, and fidelity. Together, these dimensions provide a framework for understanding different ways urban data can fall short and for developing methods that address the specific limitations of each data source.

7.1 *Summary of the Dissertation*

Chapter 3 addressed **granularity** through the problem of spatio-temporal disaggregation. Many urban datasets are released only in aggregated form, often for privacy or publication reasons, which reduces their utility for fine-scale analysis. To address this limitation, the chapter introduced the Structurally-Aware Recurrent Network (SARN), a model that combines structurally-aware spatial attention with recurrent temporal modeling to recover fine-resolution signals from coarse irregular regions. The results showed that SARN more accurately reconstructs fine-grained urban dynamics than strong baseline methods and that the recovered outputs retain practical value for downstream urban modeling tasks.

Chapter 4 addressed **coverage** through zero-shot annotation of underrepresented built-

environment features from aerial imagery. Many transportation-relevant and accessibility-relevant features remain absent from urban datasets not because they are unimportant, but because they are expensive to annotate and rarely supported by labeled training data. To address this limitation, the chapter developed a modular annotation pipeline that combines class-agnostic segmentation, heuristic candidate filtering, and Set-of-Mark prompting for a vision-language model. The results showed that direct prompting is ineffective, while structured prompting over segmented candidates substantially improves annotation quality.

Chapter 5 addressed **coherence** through spatial data integration. Urban data are often assembled from heterogeneous sources that differ in representation, geometry, or collection practice. Chapter 5 studied whether large language models can assist with two core integration tasks: spatial join and spatial union. The results showed that plain prompting is not sufficient for these tasks, largely because models struggle to derive the necessary computational geometry from natural-language descriptions and raw coordinates alone. However, when provided with interpretable geometric features, the models became more effective at making context-sensitive matching decisions. The review-and-refine prompting strategy further improved weak initial answers while preserving strong ones.

Chapter 6 addressed **fidelity** through pedestrian pathway generation. For pedestrian routing, accessibility analysis, and mobility planning, it is not enough for generated data to look visually plausible. The resulting graph must faithfully represent both the geometry and the traversable connectivity of the real built environment. The chapter introduced TraversRL, a vision-conditioned iterative graph generator that constructs pedestrian pathway networks from aerial imagery and improves them through reinforcement learning objectives aligned with graph-level geometric overlap and traversability. The results showed that TraversRL consistently improves over a strong segmentation-based baseline and that reinforcement learning further improves both graph quality and routability.

Taken together, these four studies support the central claim of the dissertation: *high-quality urban data are a prerequisite for reliable and equitable urban analytics, and improving urban data quality requires targeted methods designed around the specific deficiencies, structures, and intended uses of the data.*

7.2 *Cross-Cutting Insights*

Several broader insights emerge across the four chapters. First, improving urban data quality can involve both refining existing data and producing new data representations. The common goal is not generic data generation, but targeted improvement for specific analytical needs. Disaggregation recovers fine-scale structure from coarse releases. Annotation creates records for previously unmapped infrastructure features. Integration reconciles fragmented sources into a more coherent view. Pathway generation constructs graph representations from imagery. Together, these studies show that high-quality urban data often require task-specific methods that address both input limitations and intended downstream use.

Second, different data quality problems require different modeling strategies. Chapters 3 and 6 use specialized neural architectures because their tasks involve strong structural constraints and domain-specific outputs. Chapters 4 and 5 use foundation models because their tasks benefit from flexibility, prompting, and low-label adaptation. This contrast shows that computational methods in the urban domain should not be organized around one preferred model family. Instead, model choice should be driven by the structure of the data problem, the available supervision, and the nature of the desired output.

Third, strong performance depends on structured intermediate information. Across the dissertation, methods perform better when intermediate cues make the relevant urban structure more explicit. SARN uses containment relationships and spatial interactions. The annotation pipeline uses segmentation, filtering, and candidate marking. The integration framework uses geometric features rather than raw coordinates alone. TraversRL uses local stepwise rewards during graph generation. Together, these findings show that models benefit from cues, constraints, or feedback signals that reflect the structure of the problem.

Fourth, evaluation should match the intended use of the improved data. Data quality cannot be reduced to one universal metric. Granularity matters when the goal is to recover fine-scale dynamics. Coverage matters when important features are missing from the digital record. Coherence matters when multiple sources need to be reconciled. Fidelity matters when network data must preserve functional connectivity. A useful evaluation must therefore ask what the data are meant to support.

7.3 *Limitations & Future Directions*

Although this dissertation demonstrates that urban data quality can be improved through targeted computational methods, each study also has limitations. These limitations clarify where the proposed methods work well, where they remain constrained, and what future work is needed to make them more general, scalable, and reliable.

For Chapter 3, the main limitations concern geographic structure, temporal resolution, and transferability. The disaggregation framework was evaluated on geographic levels with clear containment relationships, where each finer unit belongs to a coarser unit. This is an important setting, but many real spatial datasets involve overlapping, changing, or mis-aligned boundaries. Future work should test whether structurally aware disaggregation can extend to these more complex geographic relationships. The chapter also focused on hourly counts, leaving open how the method performs at finer temporal resolutions, where local variation may be stronger. In addition, the transfer experiments were limited to related settings within the same city and domain. Future work should examine transfer across cities and across urban variables with different dynamics.

For Chapter 4, the main limitations come from the pipeline’s dependence on candidate quality, heuristic filtering, and current vision-language model capabilities. The annotation framework depends heavily on the segmentation and filtering stages. If SAM misses a target feature or merges it with nearby objects, the VLM may not be able to correct the error. The current filtering rules also require some feature-specific design, which limits how fully automated and general the pipeline is. Future work should improve both candidate generation and selection. One direction is to use remote-sensing-specialized segmentation models that are more robust to shadows, occlusion, small objects, and visually ambiguous markings. Another direction is to develop more principled candidate-ranking or pruning methods so that the pipeline depends less on manually designed heuristics. More broadly, the pipeline should be evaluated across more cities, imagery sources, and feature types.

For Chapter 5, the main limitation is that LLMs still need structured geometric information to perform spatial integration reliably. The experiments show that they can make useful spatial join and spatial union decisions when given features such as distance, angle,

and overlap. However, they perform much worse when asked to reason from raw coordinates and natural-language instructions alone. This means the current approach still depends on conventional spatial computation to provide geometric evidence before the LLM makes the final decision. Future work should reduce this dependence by improving the geospatial reasoning ability of foundation models. This could include post-training on spatial matching tasks or agentic approaches in which models call external tools to compute spatial relationships when needed. The current study also focuses on a limited setting: spatial joins and unions over GeoJSON LineStrings. Future work should test other geometry types, spatial formats, and integration problems, including points, polygons, rasters, inconsistent schemas, and more complex conflation tasks.

For Chapter 6, the main limitations concern deployment realism, scale sensitivity, and graph coverage. TraversRL is evaluated in a controlled setting where each rollout starts from a node on the ground-truth graph. This is useful for studying graph expansion, but it is not realistic for deployment, where the system would need to choose starting locations directly from imagery. Future work should therefore develop practical seeding methods, such as lightweight detectors that identify likely pathway locations. The method also uses fixed crop inputs, which may make it sensitive to changes in image scale, pathway width, and scene extent across cities. Future work should explore scale-aware inputs, such as adaptive crop sizes, multi-scale image context, or resolution conditioning. In addition, the current rollout process is better suited to connected graph components than to pathway systems with multiple disconnected parts. Future work should study multi-start generation strategies that identify unexplored regions and recover disconnected components more systematically.

Across all four studies, a shared deployment concern is that improved urban data can create risks when used without appropriate validation. The methods in this dissertation can make urban data more useful, but they can still produce errors, and those errors may affect downstream applications. Future systems should therefore not only produce improved datasets, but also communicate uncertainty, support calibration, and identify where human validation is most needed. This is especially important for public-facing or safety-critical applications such as mapping, routing, infrastructure assessment, and planning support.

7.4 *Final Remarks*

In pursuit of high-quality urban data, this dissertation has argued that high-quality urban data are a prerequisite for reliable and equitable urban analytics. Urban data are not simply inputs to models or planning workflows. They shape what can be observed, measured and acted upon. When data are too coarse, incomplete, inconsistent, or structurally inaccurate, even strong analytical models can produce limited or misleading results.

The dissertation has also argued that improving urban data quality requires targeted methods designed around the specific deficiencies, structures, and intended uses of the data. No single approach can address all urban data problems. Instead, different quality concerns require different methods: some recover missing detail, some expand feature coverage, some reconcile heterogeneous sources, and some preserve functional network structure. Across granularity, coverage, coherence, and fidelity, the studies in this dissertation show that urban data can be meaningfully improved when the method is aligned with the data problem.

The broader contribution of this work is therefore to shift attention from treating urban data as fixed inputs toward treating urban data quality as a central object of research. Improved data cannot replace careful measurement, documentation, or human oversight, but it can make urban datasets more useful, informative, and actionable. As cities continue to rely on data-driven systems for planning, accessibility, and decision support, developing methods for high-quality urban data will become increasingly important.

BIBLIOGRAPHY

- [1] Dragan Ahmetovic, Roberto Manduchi, James M Coughlan, and Sergio Mascetti. Zebra crossing spotter: Automatic population of spatial databases for increased safety of blind travelers. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility*, pages 251–258, 2015.
- [2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: A visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 23716–23736, 2022.
- [3] Anthropic. The Claude 3 model family: Opus, sonnet, haiku. Model card, 2024.
- [4] Richard Boadu Antwi, Michael Kimollo, Samuel Yaw Takyi, Eren Erman Ozguven, Thobias Sando, Ren Moses, and Maxim A. Dulebenets. Turning features detection from aerial images: Model development and application on florida’s public roadways. *Smart Cities*, 7(3):1414–1440, 2024.
- [5] Richard Boadu Antwi, Samuel Takyi, Alican Karaer, Eren Erman Ozguven, Michael Kimollo, Ren Moses, Maxim A. Dulebenets, and Thobias Sando. Automated geographic-information-system-based framework for detecting crosswalk changes from bi-temporal high-resolution aerial images. *Transportation Research Record: Journal of the Transportation Research Board*, 2679(4):353–370, 2025.
- [6] Rohan Arambepola, Tim C. D. Lucas, Anita K. Nandi, Peter W. Gething, and Ewan

- Cameron. A simulation study of disaggregation regression for spatial disease mapping. *Statistics in Medicine*, 41:1 – 16, 2020.
- [7] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.
- [8] Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive graph convolutional recurrent network for traffic forecasting. *ArXiv*, abs/2007.02842:, 2020.
- [9] World Bank. Global smart city partnership program, 2020.
- [10] World Bank. Urban development. <https://www.worldbank.org/en/topic/urbandevelopment>, 2023. Accessed May 2025.
- [11] Favyen Bastani, Songtao He, Sofiane Abbar, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Sam Madden, and David DeWitt. Roadtracer: Automatic extraction of road networks from aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4720–4728, 2018.
- [12] Abdellatif Bekkar, Badr Hssina, Samira Douzi, and Khadija Douzi. Air-pollution prediction in smart city, deep learning approach. *Journal of big Data*, 8(1):1–21, 2021.
- [13] Andrew R Benson, Ian GM Lawson, Matthew K Clifford, and Sean M McBride. Using robotics to detect footpath displacement caused by tree roots: A proof of concept. *Urban Forestry & Urban Greening*, 65:127312, 2021.
- [14] Zubin Bhuyan, Yuanchang Xie, AngkeaReach Rith, Xintong Yan, Nasko Apostolov, Jimi Oke, and Chengbo Ai. Crosswalknet: An optimized deep learning framework for pedestrian crosswalk detection in aerial images with high-performance computing. *arXiv preprint arXiv:2506.07885*, 2025.

- [15] Nicholas Bolten and Anat Caspi. Accessmap website demonstration: Individualized, accessible pedestrian trip planning at scale. In , pages 676–678, 10 2019.
- [16] Adam Booth, Philip James, Stephen McGough, and Ellis Solaiman. Cross-regional deep learning for air quality forecasting: A comparative study of CO, NO₂, O₃, PM_{2.5}, and PM₁₀. *Forecasting*, 7(4):66, 2025.
- [17] Alissa Brauneck, Louisa Schmalhorst, Mohammad Mahdi Kazemi Majdabadi, Mohammad Bakhtiari, Uwe Völker, Jan Baumbach, Linda Baumbach, and Gabriele Buchholtz. Federated machine learning, privacy-enhancing technologies, and data protection laws in medical research: Scoping review. *J Med Internet Res*, 25:e41588, Mar 2023.
- [18] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- [19] J. Burrell, Tim Brooke, and R. Beckwith. Vineyard computing: Sensor networks in agricultural production. *Pervasive Computing, IEEE*, 3:38– 45, 02 2004.
- [20] J. Cai, X. Fu, B. Su, and H. Fang. Air quality prediction based on spatio-temporal feature fusion over graphs. *Processes*, 13(11):3586, 2025.
- [21] Luca Caltagirone, Mauro Bellone, Lennart Svensson, and Mattias Wahde. Lidar–camera fusion for road detection using fully convolutional neural networks. *Robotics and Autonomous Systems*, 111:125–131, 2019.
- [22] Stefano Cavazzi, Brendan D. Mason, Neil Kirk, Gobe E. Hobona, Roger C. Brackin, and David Barber. Conflation of crowdsourced and authoritative data to enhance geospatial intelligence. In *Proceedings of the NATO Science and Technology Organization (STO) Meeting Proceedings IST-SET-126*, pages 7–1–7–14. NATO, 2023.
- [23] Jun Wei Chan and Chai Kiat Yeo. A transformer based approach to electricity load forecasting. *The Electricity Journal*, 37(2):107370, 2024.

- [24] Dingyang Chen, Qi Zhang, and Yinglun Zhu. Efficient sequential decision making with large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9157–9170, Miami, Florida, USA, 2024. Association for Computational Linguistics.
- [25] Hainan Chen and Volker Walter. Hierarchical quality inspection of spatial data by data integration. In *Proceedings of the ASPRS 2010 Annual Conference, San Diego, CA, USA*, pages 26–30. Citeseer, 2010.
- [26] Hui Chen, Jian Huang, Yong Lu, and Jijie Huang. Multi-scale spatio-temporal graph neural network for urban traffic flow prediction. *Scientific Reports*, 15:26732, 2025.
- [27] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé, Jared Kaplan, Harrison Edwards, Yura Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, David W. Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William H. Guss, Alex Nichol, Igor Babuschkin, Suchir Balaji, Shantanu Jain, Andrew Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew M. Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *ArXiv*, abs/2107.03374, 2021.
- [28] Weihuang Chen, Fangfang Wang, and Hongbin Sun. S2tnet: Spatio-temporal transformer networks for trajectory prediction in autonomous driving. In *Asian Conference on Machine Learning*, pages 454–469, , 2021. PMLR, .
- [29] Yibi Chen, Kenli Li, Chai Kiat Yeo, and Keqin Li. Traffic forecasting with graph spatial–temporal position recurrent network. *Neural Networks*, 162:340–349, 2023.
- [30] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong,

- Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.
- [31] Ching-Hsue Cheng and Sue-Fen Huang. Forecasting of ozone concentration using frequency ma-owa model. In *2009 IEEE International Conference on Systems, Man and Cybernetics*, pages 3461–3466. IEEE, 2009.
- [32] Yu Ying Chiu, Ashish Sharma, Inna Wanyin Lin, and Tim Althoff. A computational framework for behavioral assessment of llm therapists. *arXiv preprint arXiv:2401.00820*, 2024.
- [33] Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30, pages 4299–4307, 2017.
- [34] Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. Lm vs lm: Detecting factual errors via cross examination. *arXiv preprint arXiv:2305.13281*, 2023.
- [35] Alexis J. Comber and Wen Zeng. Spatial interpolation using areal features: A review of methods and opportunities using new forms of data with coded illustrations. *Geography Compass*, :, 2019.
- [36] P. Dabove, M. Daud, and L. Olivotto. Revolutionizing urban mapping: deep learning and data fusion strategies for accurate building footprint segmentation. *Scientific Reports*, 14:13510, 2024.
- [37] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N. Fung, and Steven C. H. Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.
- [38] Aayushi Dangol, Michele Newman, Robert Wolfe, Jin Ha Lee, Julie A Kientz, Jason Yip, and Caroline Pitt. Mediating culture: Cultivating socio-cultural understanding of

- ai in children through participatory design. In *Proceedings of the 2024 ACM Designing Interactive Systems Conference*, pages 1805–1822, 2024.
- [39] Michael Han Daniel Han and Unsloth team. Unsloth. , 2023.
- [40] Anubrata Das, Houjiang Liu, Venelin Kovatchev, and Matthew Lease. The state of human-centered nlp technology for fact-checking. *Information processing & management*, 60(2):103219, 2023.
- [41] DeepSeek-AI. DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.
- [42] DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in large language models via reinforcement learning. *Nature*, 645:633–638, 2025.
- [43] Ligao Deng, Yupeng Deng, Yu Meng, Jingbo Chen, Zhihao Xi, Diyou Liu, and Qifeng Chu. Gld-road: A global–local decoding road network extraction model for remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 228:741–755, 2025.
- [44] Austin Derrow-Pinion, Jennifer She, David Wong, Oliver Lange, Todd Hester, Luis Perez, Marc Nunkesser, Seongjae Lee, Xueying Guo, Brett Wiltshire, et al. Eta prediction with graph neural networks in google maps. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 3767–3776, , 2021. .
- [45] Pierre Deville, Catherine Linard, Samuel Martin, Marius Gilbert, Forrest R. Stevens, Andrea E. Gaughan, Vincent D. Blondel, and Andrew J. Tatem. Dynamic population mapping using mobile phone data. *Proceedings of the National Academy of Sciences of the United States of America*, 111(45):15888–15893, September 2014.
- [46] Lei Ding, Hao Tang, and Lorenzo Bruzzone. Lanet: Local attention embedding to improve the semantic segmentation of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 59(1):426–435, 2020.

- [47] Yueli Ding, Haojie Xu, Di Wang, Ke Li, and Yumin Tian. Visual selection and multi-stage reasoning for rsvg. *IEEE Geoscience and Remote Sensing Letters*, .
- [48] Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1286–1305, 2021.
- [49] Jie Dong, Yaoli Zhang, and Jiang Hu. Short-term air quality prediction based on emd-transformer-bilstm. *Scientific Reports*, 14(1):20513, 2024.
- [50] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- [51] Zhenhong Du, Xianwei Zhao, Xinyue Ye, Jingwei Zhou, Feng Zhang, and Renyi Liu. An effective high-performance multiway spatial join algorithm with spark. *ISPRS International Journal of Geo-Information*, 6(4):96, 2017.
- [52] Daniela Durand, Jose Aguilar, and Maria D R-Moreno. An analysis of the energy consumption forecasting problem in smart buildings using lstm. *Sustainability*, 14(20):13358, 2022.
- [53] Aleksandra Edwards and Jose Camacho-Collados. Language models for text classification: Is in-context learning enough? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10058–10072, Torino, Italia, 2024. ELRA and ICCL.
- [54] Yochai Eisenberg, Kerri A Vanderbom, and Vijay Vasudevan. Does the built environment moderate the relationship between having a disability and lower levels of physical activity? a systematic review. *Preventive medicine*, 95, 2017.

- [55] Zekeriya Erkin, Juan Ramon Troncoso-pastoriza, R.L. Lagendijk, and Fernando Perez-Gonzalez. Privacy-preserving data aggregation in smart metering systems: an overview. *IEEE Signal Processing Magazine*, 30(2):75–86, 2013.
- [56] Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, 2019.
- [57] Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. KTO: Model alignment as prospect theoretic optimization. In *International Conference on Machine Learning*, 2024.
- [58] Wei Fang, Runsu Zhu, and Jerry Chun-Wei Lin. An air quality prediction model based on improved vanilla lstm with multichannel input and multiroute output. *Expert systems with applications*, 211:118422, 2023.
- [59] Yuchen Fang, Yanjun Qin, Haiyong Luo, Fang Zhao, Bingbing Xu, Liang Zeng, and Chenxing Wang. When spatio-temporal meet wavelets: Disentangled traffic forecasting via efficient spectral graph attention networks. *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, ():517–529, 2023.
- [60] Yuchen Fang, Fang Zhao, Yanjun Qin, Haiyong Luo, and Chenxing Wang. Learning all dynamics: Traffic forecasting via locality-aware spatio-temporal joint transformer. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):23433–23446, 2022.
- [61] Marjan Faraji, Saeed Nadi, Omid Ghaffarpasand, Saeid Homayoni, and Kay Downey. An integrated 3d cnn-gru deep learning method for short-term prediction of pm2. 5 concentration in urban environment. *Science of The Total Environment*, 834:155324, 2022.
- [62] Jon E Froehlich, Yochai Eisenberg, Maryam Hosseini, Fabio Miranda, Marc Adams, Anat Caspi, Holger Dieterich, Heather Feldner, Aldo Gonzalez, Claudina De Gyves,

- et al. The future of urban accessibility for people with disabilities: Data collection, analytics, policy, and tools. In *Proceedings of the 24th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–8, 2022.
- [63] Haotian Gao, Renhe Jiang, Zheng Dong, Jinliang Deng, and Xuan Song. Spatio-temporal-decoupled masked pre-training for traffic forecasting. *arXiv preprint arXiv:2312.00516*, (), 2023.
- [64] Jiaxin Gao, Wenbo Hu, Dongxiao Zhang, and Yuntian Chen. Tgdlf2. 0: Theory-guided deep-learning for electrical load forecasting via transformer and transfer learning. *arXiv preprint arXiv:2210.02448*, 2022.
- [65] Jiaxuan Gao, Shusheng Xu, Wenjie Ye, Weilin Liu, Chuyi He, Wei Fu, Zhiyu Mei, Guangju Wang, and Yi Wu. On designing effective rl reward at training time for llm reasoning. *arXiv preprint arXiv:2410.15115*, 2024.
- [66] Gemini Team. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.
- [67] Mohamed Barakat A. Gibril, Rami Al-Ruzouq, Abdallah Shanableh, Ratiranjan Jena, Jan Bolcek, Helmi Zulhaidi Mohd Shafri, and Omid Ghorbanzadeh. Transformer-based semantic segmentation for large-scale building footprint extraction from very-high resolution satellite images. *Advances in Space Research*, 73(10):4937–4954, 2024.
- [68] Ekaterina Gilman. Development and evaluation of an urban data governance model. *Government Information Quarterly*, 41(1):101–112, 2024.
- [69] Bernardo Gomes, José Coelho, and Helena Aidos. A survey on traffic flow prediction and classification. *Intelligent Systems with Applications*, ():200268, 2023.
- [70] M F Goodchild, Luc Anselin, and Uwe Deichmann. A framework for the areal interpolation of socioeconomic data. *Environment and Planning A*, 25:383 – 397, 1993.
- [71] Aaron Grattafiori et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

- [72] Ben Green, Gabe Cunningham, Ariel Ekblaw, Paul M. Kominers, Andrew Linzer, and Susan P. Crawford. Open data privacy: A risk-benefit, process-oriented approach to sharing and protecting municipal data. In , page , , 2017. .
- [73] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- [74] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 922–929, , 2019. .
- [75] Ryuhei Hamaguchi, Aito Fujita, Keisuke Nemoto, Tomoyuki Imaizumi, and Shuhei Hikosaka. Effective use of dilated convolutions for segmenting small object instances in remote sensing imagery. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 1442–1450. IEEE, 2018.
- [76] Bin Han and Bill Howe. Adapting to skew: Imputing spatiotemporal urban data with 3d partial convolutions and biased masking, 2023.
- [77] Bin Han and Bill Howe. Sarn: Structurally-aware recurrent network for spatio-temporal disaggregation. In *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems, SIGSPATIAL '24*, page 338–349, New York, NY, USA, 2024. Association for Computing Machinery.
- [78] Bin Han, Robert Wolfe, Anat Caspi, and Bill Howe. Can large language models integrate spatial data? empirical insights into reasoning strengths and computational weaknesses. *arXiv preprint arXiv:2508.05009*, 2025.
- [79] Bin Han, Sean T Yang, and Christopher LuVogt. Cross-lingual text classification with large language models. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 1005–1008, 2025.

- [80] Bin Han, Yiwei Yang, Anat Caspi, and Bill Howe. Towards zero-shot annotation of the built environment with vision-language models. In *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*, pages 601–604, 2024.
- [81] Bin Han, Haotian Zhu, Sitong Zhou, Sofia Ahmed, Md Mushfiqur Rahman, Fei Xia, and Kevin Lybarger. Huskyscribe at mediqua-sum 2023: Summarizing clinical dialogues with transformers. In *CLEF (Working Notes)*, pages 1488–1509, 2023.
- [82] Mingdong Han and Lingyan Fan. A short-term energy consumption forecasting method for attention mechanisms based on spatio-temporal deep learning. *Computers and Electrical Engineering*, 114:109063, 2024.
- [83] Renlong Hang, Ping Yang, Feng Zhou, and Qingshan Liu. Multiscale progressive segmentation network for high-resolution remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–12, 2022.
- [84] Kotaro Hara, Jin Sun, Robert Moore, David Jacobs, and Jon Froehlich. Tohme: detecting curb ramps in google street view using crowdsourcing, computer vision, and machine learning. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, pages 189–204, 2014.
- [85] Hossein Hassani and Emmanuel Sirimal Silva. The role of chatgpt in data science: How ai-assisted conversational interfaces are revolutionizing the field. *Big Data Cogn. Comput.*, 7:62, 2023.
- [86] Hengyuan He, Zhengtao Long, Yingchao Zhang, and Xiaofei Jiang. Spatio-temporal transformer traffic prediction network based on multi-level causal attention. *PLOS ONE*, 20(9):e0331139, 2025.
- [87] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

- [88] Siyuan He, Yuhao Liu, Jin Peng, Dean Xu, and Yu Wang. A CEEMDAN-GNN-Transformer hybrid model for air quality index forecasting: a case study of chang'an town, dongguan, china. *Frontiers in Environmental Science*, 13:1653446, 2025.
- [89] Songtao He, Favyen Bastani, Satvat Jagwani, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Mohamed M Elshrif, Samuel Madden, and Mohammad Amin Sadeghi. Sat2graph: Road graph extraction through graph-tensor encoding. In *European Conference on Computer Vision*, pages 51–67. Springer, 2020.
- [90] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher. Pda: Privacy-preserving data aggregation in wireless sensor networks. In *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, pages 2045–2053, , 2007. .
- [91] Tomislav Hengl, Madlene Nussbaum, Marvin N Wright, Gerard BM Heuvelink, and Benedikt Gräler. Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables. *PeerJ*, 6:e5518, 2018.
- [92] Congrui Hetang, Haoru Xue, Cindy Le, Tianwei Yue, Wenping Wang, and Yihui He. Segment anything model for road network graph extraction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2556–2566, 2024.
- [93] Kethmi Hirushini Hettige, Jiahao Ji, Shili Xiang, Cheng Long, Gao Cong, and Jingyuan Wang. Airphynet: Harnessing physics-guided neural networks for air quality prediction. *arXiv preprint arXiv:2402.03784*, 2024.
- [94] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [95] Danfeng Hong, Zhu Han, Jing Yao, Lianru Gao, Bing Zhang, Antonio Plaza, and Jocelyn Chanussot. Spectralformer: Rethinking hyperspectral image classification with transformers. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2021.

- [96] Sue Hope and Allison Kealy. Using topological relationships to inform a data integration process. *Transactions in GIS*, 12(2):267–283, 2008.
- [97] Maryam Hosseini, Andres Sevtsuk, Fabio Miranda, Roberto M Cesar Jr, and Claudio T Silva. Mapping the walk: A scalable computer vision approach for generating sidewalk network datasets from aerial imagery. *Computers, Environment and Urban Systems*, 101:101950, 2023.
- [98] Bill Howe, Jackson Brown, Bin Han, Bernease Herman, Nic Weber, An Yan, Sean Yang, and Yiwei Yang. Integrative urban ai to expand coverage, access, and equity of urban data. *The European Physical Journal Special Topics*, 231, 04 2022.
- [99] Bill Howe, Jackson Maxfield Brown, Bin Han, Bernease Herman, Nic Weber, An Yan, Sean Yang, and Yiwei Yang. Integrative urban ai to expand coverage, access, and equity of urban data. *The European Physical Journal Special Topics*, 231(9):1741–1752, 2022.
- [100] Jianlin Hu, Jianjun Chen, Qi Ying, and Hongliang Zhang. One-year simulation of ozone and particulate matter in china using wrf/cmaq modeling system. *Atmospheric Chemistry and Physics*, 16(16):10333–10350, 2016.
- [101] Yu Huang, Josh Jia-Ching Ying, and Vincent S Tseng. Spatio-attention embedded recurrent neural network for air quality prediction. *Knowledge-Based Systems*, 233:107416, 2021.
- [102] Transportation Data Equity Initiative. Transportation data equity initiative. <https://tdei.cs.washington.edu/>, 2025. Accessed: 2025-03-27.
- [103] Edwin H Jacox and Hanan Samet. Spatial join techniques. *ACM Transactions on Database Systems (TODS)*, 32(1):7–es, 2007.
- [104] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

- [105] R Janarthanan, P Partheeban, K Somasundaram, and P Navin Elamparithi. A deep learning approach for prediction of air quality index in a metropolitan city. *Sustainable Cities and Society*, 67:102720, 2021.
- [106] John Jensen, Alan Saalfeld, Fred Broome, Dave Cowen, Kevin Price, Doug Ramsey, Lewis Lapine, and E Lynn Usery. Spatial data acquisition and integration. In *A research agenda for geographic information science*, pages 17–60. CRC Press, 2004.
- [107] Bin Jia, Xiaosong Zhang, Jiewen Liu, Yang Zhang, Ke Huang, and Yongquan Liang. Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in iiot. *IEEE Transactions on Industrial Informatics*, 18(6):4049–4058, 2022.
- [108] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.
- [109] Hang Jiang, Xiajie Zhang, Robert Mahari, Daniel Kessler, Eric Ma, Tal August, Irene Li, Alex’Sandy’ Pentland, Yoon Kim, Deb Roy, et al. Leveraging large language models for learning complex legal concepts through storytelling. *arXiv preprint arXiv:2402.17019*, 2024.
- [110] Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Xin Zhao, and Ji-Rong Wen. StructGPT: A general framework for large language model to reason over structured data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9237–9251, Singapore, 2023. Association for Computational Linguistics.
- [111] Renhe Jiang, Zhaonan Wang, Jiawei Yong, Puneet Jeph, Quanjun Chen, Yasumasa Kobayashi, Xuan Song, Shintaro Fukushima, and Toyotaro Suzumura. Spatio-temporal meta-graph learning for traffic forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 8078–8086, , 2023. .

- [112] Yilun Jin, Kai Chen, and Qiang Yang. Selective cross-city transfer learning for traffic prediction via source city region re-weighting. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 731–741, , 2022. .
- [113] Yilun Jin, Kai Chen, and Qiang Yang. Transferable graph structure learning for graph-based traffic forecasting across cities. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1032–1043, , 2023. .
- [114] Daniel Jurafsky and James Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, volume 2. , , 02 2008.
- [115] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. MDETR: Modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1780–1790, 2021.
- [116] Ryo Kamoi, Yusen Zhang, Nan Zhang, Jiawei Han, and Rui Zhang. When can llms actually correct their own mistakes? a critical survey of self-correction of llms. *Transactions of the Association for Computational Linguistics*, 12:1417–1440, 2024.
- [117] Hassan A Karimi and Piyawan Kasemsuppakorn. Pedestrian network map generation approaches and recommendation. *International Journal of Geographical Information Science*, 27(5):947–962, 2013.
- [118] Hamed Karimian, Qi Li, Chunlin Wu, Yanlin Qi, Yuqin Mo, Gong Chen, Xianfeng Zhang, and Sonali Sachdeva. Evaluation of different machine learning approaches to forecasting pm2. 5 mass concentrations. *Aerosol and Air Quality Research*, 19(6):1400–1410, 2019.
- [119] Michael Katz, Harsha Kokel, Kavitha Srinivas, and Shirin Sohrabi. Thought of search: Planning with language models through the lens of efficiency. In *Advances in Neural Information Processing Systems*, 2024.

- [120] Geunwoo Kim, Pierre Baldi, and Stephen McAleer. Language models can solve computer tasks. *Advances in Neural Information Processing Systems*, 36:39648–39677, 2023.
- [121] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.
- [122] Rob Kitchin. Data-driven urbanism. In *The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences*. SAGE Publications, 2014.
- [123] Weiyang Kong, Ziyu Guo, and Yubao Liu. Spatio-temporal pivotal graph neural networks for traffic flow forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8627–8635, 2024.
- [124] Xiangfu Kong and Jiawen Yang. A scenario-based map-matching algorithm for complex urban road network. *Journal of intelligent transportation systems*, 23(6):617–631, 2019.
- [125] Yaxuan Kong, Yuqi Nie, Xiaowen Dong, John M Mulvey, H Vincent Poor, Qingsong Wen, and Stefan Zohren. Large language models for financial and investment management: Applications and benchmarks. *Journal of Portfolio Management*, 51(2), 2024.
- [126] Roman Koshkin, Katsuhito Sudoh, and Satoshi Nakamura. Transllama: Llm-based simultaneous translation system. *arXiv preprint arXiv:2402.04636*, 2024.
- [127] Kartik Kuckreja, Muhammad Sohail Danish, Muzammal Naseer, Abhijit Das, Salman Khan, and Fahad Shahbaz Khan. GeoChat: Grounded large vision-language model for remote sensing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.
- [128] Ujjwal Kumar and VK Jain. Arima forecasting of ambient air pollutants (o₃, no,

- no 2 and co). *Stochastic Environmental Research and Risk Assessment*, 24:751–760, 2010.
- [129] Chaoan Lai, Yina Wang, Jianhua Zhu, and Xuequan Zhou. Deep learning-based energy consumption prediction model for green industrial parks. *Applied Artificial Intelligence*, 39(1):2462375, 2025.
- [130] Shiyong Lan, Yitong Ma, Weikang Huang, Wenwu Wang, Hongyu Yang, and Pyang Li. Dstagnn: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting. In *International conference on machine learning*, pages 11906–11917, , 2022. PMLR, .
- [131] Ho Chung Law, Dino Sejdinovic, Ewan Cameron, Tim Lucas, Seth Flaxman, Katherine Battle, and Kenji Fukumizu. Variational learning on aggregate outputs with gaussian processes. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, , 2018. Curran Associates, Inc.
- [132] Gaël Le Mens, Balázs Kovács, Michael T Hannan, and Guillem Pros. Uncovering the semantics of concepts using gpt-4. *Proceedings of the National Academy of Sciences*, 120(49):e2309350120, 2023.
- [133] Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, et al. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023.
- [134] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020.

- [135] Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, et al. From generation to judgment: Opportunities and challenges of llm-as-a-judge. *arXiv preprint arXiv:2411.16594*, 2024.
- [136] Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. Llms-as-judges: a comprehensive survey on llm-based evaluation methods. *arXiv preprint arXiv:2412.05579*, 2024.
- [137] Jinglin Li, Dawei Fu, Quan Yuan, Haohan Zhang, Kaihui Chen, Shu Yang, and Fangchun Yang. A traffic prediction enabled double rewarded value iteration network for route planning. *IEEE Transactions on Vehicular Technology*, 68(5):4170–4181, 2019.
- [138] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR, 2023.
- [139] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 12888–12900. PMLR, 2022.
- [140] Mengzhang Li and Zhanxing Zhu. Spatial-temporal fusion graph neural networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 4189–4196, , 2021. .
- [141] Ruosen Li, Teerth Patel, and Xinya Du. Prd: Peer rank and discussion improve large language model based evaluations. *arXiv preprint arXiv:2307.02762*, 2023.
- [142] Xiang Li, Jian Ding, and Mohamed Elhoseiny. Vrsbench: A versatile vision-language benchmark dataset for remote sensing image understanding. *arXiv preprint arXiv:2406.12384*, 2024.

- [143] Xiaonan Li and Xipeng Qiu. MoT: Memory-of-thought enables ChatGPT to self-improve. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6354–6374, Singapore, December 2023. Association for Computational Linguistics.
- [144] Yunfan Li. Energy consumption forecasting with deep learning. In *Journal of Physics: Conference Series*, volume 2711, page 012012. IOP Publishing, 2024.
- [145] Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*, 2023.
- [146] Yuxuan Liang, Yutong Xia, Songyu Ke, Yiwei Wang, Qingsong Wen, Junbo Zhang, Yu Zheng, and Roger Zimmermann. Airformer: Predicting nationwide air quality in china with transformers. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 14329–14337, 2023.
- [147] Feng Lin, Dong Jae Kim, et al. When llm-based code generation meets the software development process. *arXiv preprint arXiv:2403.15852*, 2024.
- [148] Chao Liu, Yangbo Jiang, and Nenggan Zheng. Nettracer: A topology-aware iterative tracing approach for tubular structure extraction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20593–20602, 2025.
- [149] Fan Liu, Delong Chen, Zhangqingyun Guan, Xiaocong Zhou, Jiale Zhu, Qiaolin Ye, Liyong Fu, and Jun Zhou. RemoteCLIP: A vision language foundation model for remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–16, 2024.
- [150] Hangchen Liu, Zheng Dong, Renhe Jiang, Jiewen Deng, Jinliang Deng, Quanjun Chen, and Xuan Song. Spatio-temporal adaptive embedding makes vanilla transformer sota for traffic forecasting. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 4125–4129, , 2023. .

- [151] Hangcheng Liu, Zheng Dong, Renhe Jiang, Jiewen Deng, Jinliang Deng, Quanjun Chen, and Xuan Song. Staeformer: Spatio-temporal adaptive embedding makes vanilla transformer sota for traffic forecasting. *ArXiv*, abs/2308.10425:, 2023.
- [152] Hao Liu, Jindong Han, Yanjie Fu, Yanyan Li, Kai Chen, and Hui Xiong. Unified route representation learning for multi-modal transportation recommendation with spatiotemporal pre-training. *The VLDB Journal*, 32(2):325–342, 2023.
- [153] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.
- [154] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.
- [155] Lingbo Liu, Mengmeng Liu, Guanbin Li, Ziyi Wu, Junfan Lin, and Liang Lin. Road network-guided fine-grained urban traffic flow inference. *IEEE Transactions on Neural Networks and Learning Systems*, 34(6):, 2023.
- [156] Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Minh Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, et al. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. *arXiv preprint arXiv:2601.05242*, 2026.
- [157] Yifan Liu, Qigang Zhu, Feng Cao, Junke Chen, and Gang Lu. High-resolution remote sensing image segmentation framework based on attention mechanism and adaptive weighting. *ISPRS International Journal of Geo-Information*, 10(4):241, 2021.
- [158] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.
- [159] Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. Interpretable long-form legal question answering with retrieval-augmented large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22266–22275, 2024.

- [160] Aurelie Lozano, Hongfei Li, Alexandru Niculescu-Mizil, Yan Liu, Claudia Perlich, Jonathan Hosking, and Naoki Abe. Spatial-temporal causal modeling for climate change attribution. In , volume , pages 587–596, 06 2009.
- [161] Bin Lu, Xiaoying Gan, Weinan Zhang, Huaxiu Yao, Luoyi Fu, and Xinbing Wang. Spatio-temporal graph few-shot learning with cross-city knowledge transfer. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1162–1172, , 2022. .
- [162] Xiaoyan Lu, Yanfei Zhong, Zhuo Zheng, and Liangpei Zhang. Gamsnet: Globally aware road detection network with multi-scale residual learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 175:340–352, 2021.
- [163] IV Luttrell, Joseph, Yuanyuan Zhang, and Chaoyang Zhang. Automatically detect crosswalks from satellite view images: A deep learning approach with ground truth verification. *International Journal of Transportation Science and Technology*, 16:165–176, 2024.
- [164] Alexandra L’Heureux, Katarina Grolinger, and Miriam AM Capretz. Transformer-based model for electrical load forecasting. *Energies*, 15(14):4993, 2022.
- [165] Shaojun Ma, Shu Liu, Hongyuan Zha, and Haomin Zhou. Learning stochastic behaviour of aggregate data. In *International Conference on Machine Learning*, page , , 2020. .
- [166] Xiaolei Ma, Zhuang Dai, Zhengbing He, Jihui Ma, Yong Wang, and Yunpeng Wang. Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. *Sensors*, 17(4):818, 2017.
- [167] Xiaolei Ma, Zhuang Dai, Zhengbing He, Jihui Ma, Yong Wang, and Yunpeng Wang. Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction. *Sensors*, 17(4):818, 2017.
- [168] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-

- refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594, 2023.
- [169] Rohin Manvi, Samar Khanna, Gengchen Mai, Marshall Burke, David Lobell, and Stefano Ermon. GeoLLM: Extracting geospatial knowledge from large language models. In *International Conference on Learning Representations*, 2024.
- [170] Avraham Margalit and Gary D Knott. An algorithm for computing the union, intersection or difference of two polygons. *Computers & Graphics*, 13(2):167–183, 1989.
- [171] Tomás E. Martínez-Chao, Agustín Menéndez-Díaz, Silverio García-Cortés, and Pierpaolo D’Agostino. Urban pedestrian routes’ accessibility assessment using geographic information system processing and deep learning-based object detection. *Sensors*, 24(11):3667, 2024.
- [172] CHRISTINE MENDOZA and ANAT CASPI. Development and taxonomization of an accessibility-first open data gis standard. UniversityofNorthCarolina at Chapel Hill and University of Washington, 2023.
- [173] Lu Mi, Hang Zhao, Charlie Nash, Xiaohan Jin, Jiyang Gao, Chen Sun, Cordelia Schmid, Nir Shavit, Yuning Chai, and Dragomir Anguelov. Hdmapgen: A hierarchical graph generative model of high definition maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4227–4236, 2021.
- [174] Jiqian Mo and Zhiguo Gong. Cross-city multi-granular adaptive transfer learning for traffic flow prediction. *IEEE Transactions on Knowledge and Data Engineering*, ():, 2022.
- [175] João Monteiro, Bruno Martins, Patricia Murrieta-Flores, and João M. Pires. Spatial disaggregation of historical census data leveraging multiple sources of ancillary information. *ISPRS International Journal of Geo-Information*, 8(8):, 2019.
- [176] João Monteiro, Bruno Martins, Patricia Murrieta-Flores, and João M. Pires. Spatial disaggregation of historical census data leveraging multiple sources of ancillary information. *ISPRS International Journal of Geo-Information*, 8(8):, 2019.

- [177] João Monteiro, Bruno Martins, and João Pires. A hybrid approach for the spatial disaggregation of socio-economic indicators. *International Journal of Data Science and Analytics*, 5, 03 2018.
- [178] Luis Moreira-Matias, Joao Gama, Michel Ferreira, Joao Mendes-Moreira, and Luis Damas. Predicting taxi-passenger demand using streaming data. *IEEE Transactions on Intelligent Transportation Systems*, 14(3):1393–1402, 2013.
- [179] Samuel Moveh, Emmanuel Alejandro Merchán-Cruz, Maher Abuhussain, Yakubu Aminu Dodo, Saleh Alhumaid, and Ali Hussain Alhamami. Deep learning framework using transformer networks for multi building energy consumption prediction in smart cities. *Energies*, 18(6):1468, 2025.
- [180] Samuel Moveh, Emmanuel Alejandro Merchán-Cruz, Maher Abuhussain, Yakubu Aminu Dodo, Saleh Alhumaid, and Ali Hussain Alhamami. Deep learning framework using transformer networks for multi building energy consumption prediction in smart cities. *Energies*, 18(6):1468, 2025.
- [181] Sébastien Mustière and Jérôme Jolivet. Large scale geospatial data conflation: A feature matching optimization-based approach. *Computers, Environment and Urban Systems*, 87:101606, 2021.
- [182] Mehdi Neshat, Menasha Thilakaratne, Mohammed El-Abd, Seyedali Mirjalili, Amir H. Gandomi, and John Bolland. Smart buildings energy consumption forecasting using adaptive evolutionary bagging extra tree learning models. *Energy*, 333:137130, 2025.
- [183] PJ García Nieto, Elías F Combarro, JJ del Coz Díaz, and Elena Montañés. A svm-based regression model to study the air quality at local scale in oviedo urban area (northern spain): A case study. *Applied Mathematics and Computation*, 219(17):8923–8937, 2013.
- [184] Ali Yadavar Nikraves, Samuel A Ajila, Chung-Horng Lung, and Wayne Ding. Mobile network traffic prediction using mlp, mlpwd, and svm. In *2016 IEEE International Congress on Big Data (BigData Congress)*, pages 402–409, , 2016. IEEE, .

- [185] Huan Ning, Xinyue Ye, Zhihui Chen, Tao Liu, and Tianzhi Cao. Sidewalk extraction using aerial and street view images. *Environment and Planning B: Urban Analytics and City Science*, 49(1):7–22, 2022.
- [186] Changan Niu, Chuanyi Li, Vincent Ng, Jidong Ge, Liguang Huang, and Bin Luo. Spt-code: Sequence-to-sequence pre-training for learning source code representations. In *Proceedings of the 44th international conference on software engineering*, pages 2006–2018, 2022.
- [187] John S O’Meara, Jared Hwang, Zeyu Wang, Michael Saugstad, and Jon E Froehlich. Rampnet: A two-stage pipeline for bootstrapping curb ramp detection in streetscape images from open government metadata. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6656–6665, 2025.
- [188] OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- [189] OpenAI. GPT-4V(ision) system card. System card, 2023.
- [190] Kun Ouyang, Yuxuan Liang, Ye Liu, Zekun Tong, Sijie Ruan, Yu Zheng, and David S Rosenblum. Fine-grained urban flow inference. *IEEE transactions on knowledge and data engineering*, 34(6):2755–2770, 2020.
- [191] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.
- [192] Madhurima Panja, Tanujit Chakraborty, Anubhab Biswas, and Soudeep Deb. Estgcn: Extreme spatiotemporal graph convolutional networks for air quality forecasting. *arXiv preprint arXiv:2411.12258*, 2024.
- [193] Vincent Perot, Kai Kang, Florian Luisier, Guolong Su, Xiaoyu Sun, Ramya Sree Boppana, Zilong Wang, Zifeng Wang, Jiaqi Mu, Hao Zhang, Chen-Yu Lee, and Nan

- Hua. LMDX: Language model-based document information extraction and localization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15140–15168, Bangkok, Thailand, 2024. Association for Computational Linguistics.
- [194] Darius Peteleaza, Alexandru Matei, Radu Sorostinean, Arpad Gellert, Ugo Fiore, Bala-Constantin Zamfirescu, and Francesco Palmieri. Electricity consumption forecasting for sustainable smart cities using machine learning methods. *Internet of Things*, 27:101322, 2024.
- [195] Aida Mehdipour Pirbazari, Ekanki Sharma, Antorweep Chakravorty, Wilfried Elmenreich, and Chunming Rong. An ensemble approach for multi-step ahead energy forecasting of household communities. *IEEE Access*, 9:36218–36240, 2021.
- [196] Yanlin Qi, Qi Li, Hamed Karimian, and Di Liu. A hybrid model for spatiotemporal forecasting of pm2. 5 based on graph convolutional neural network and long short-term memory. *Science of the Total Environment*, 664:1–10, 2019.
- [197] Hanchen Qiu, Haojia Zhu, Zhicheng Li, and Jiahui Jin. Mrg-ser: Self-supervised spatial entity resolution based on multi-relational graph. *Proceedings of the VLDB Endowment. ISSN*, 2150:8097, .
- [198] Yue Qiu, Xuesheng Zhao, Deqin Fan, and Songnian Li. Geospatial disaggregation of population data in supporting sdg assessments: A case study from deqing county, china. *ISPRS Int. J. Geo Inf.*, 8:356, 2019.
- [199] Alec Radford. Improving language understanding by generative pre-training. , 2018.
- [200] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [201] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen

- Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.
- [202] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [203] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, 2023.
- [204] Jasim M Rajab, MZ MatJafri, and HS Lim. Combining multiple regression and principal component analysis for accurate predictions for column ozone in peninsular malaysia. *Atmospheric Environment*, 71:36–43, 2013.
- [205] Giridhar Kaushik Ramachandran, Yajuan Fu, Bin Han, Kevin Lybarger, Nicholas J Dobbins, Özlem Uzuner, and Meliha Yetisgen. Prompt-based extraction of social determinants of health using few-shot learning. *arXiv preprint arXiv:2306.07170*, 2023.
- [206] Qianqian Ren, Yang Li, and Yong Liu. Transformer-enhanced periodic temporal convolution network for long short-term traffic flow forecasting. *Expert Systems with Applications*, 227:120203, 2023.
- [207] Hannah Ritchie, Veronika Samborska, and Max Roser. Urbanization. *Our world in data*, 2024.
- [208] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597:, 2015.
- [209] Said E Said and David A Dickey. Testing for unit roots in autoregressive-moving average models of unknown order. *Biometrika*, 71(3):599–607, 1984.

- [210] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, 2023.
- [211] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- [212] Erfan Hosseini Sereshgi and Carola Wenk. Map stitcher: Graph sampling-based map conflation. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Spatial Big Data and AI for Industrial Applications (GeoIndustry 2024)*, pages 5–15. ACM, 2024.
- [213] Pourya Shamsolmoali, Masoumeh Zareapoor, Huiyu Zhou, Ruili Wang, and Jie Yang. Road segmentation for remote sensing images using adversarial spatial pyramid networks. *IEEE Transactions on Geoscience and Remote Sensing*, 59(6):4673–4688, 2020.
- [214] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL <https://arxiv.org/abs/2402.03300>, 2(3):5, 2024.
- [215] Yilei Shi, Qingyu Li, and Xiao Xiang Zhu. Building segmentation through a gated graph convolutional neural network with deep structured feature embedding. *ISPRS Journal of Photogrammetry and Remote Sensing*, 159:184–197, 2020.
- [216] Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. What does clip know about a red circle? visual prompt engineering for vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11987–11997, 2023.
- [217] Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. What does clip know about a red circle? visual prompt engineering for vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11987–11997, 2023.

- [218] Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.
- [219] Rahul Singh and Yongxin Chen. Learning gaussian hidden markov models from aggregate data. *IEEE Control Systems Letters*, 7:478–483, 2023.
- [220] Simranjit Singh, Michael Fore, and Dimitrios Stamoulis. GeoLLM-Engine: A realistic environment for building geospatial copilots. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 585–594, 2024.
- [221] Derek Sonderegger and Robert Buscaglia. *Introduction to Statistical Methodology, Second Edition.* , , 12 2020.
- [222] Chao Song, Youfang Lin, S. Guo, and Huaiyu Wan. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *AAAI Conference on Artificial Intelligence*, page , , 2020. .
- [223] Robert Spencer, Surangika Ranathunga, Mikael Boulic, Andries van Heerden, and Teo Susnjak. Transfer learning on transformers for building energy consumption forecasting—a comparative study. *Energy and Buildings*, 336:115632, 2025.
- [224] Forrest R. Stevens, Andrea E. Gaughan, Catherine Linard, and Andrew J. Tatem. Disaggregating census data for population mapping using random forests with remotely-sensed and ancillary data. *PLOS ONE*, 10(2):1–22, 02 2015.
- [225] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback. In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021, 2020.
- [226] Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. Table meets LLM: Can large language models understand structured table data? a benchmark

- and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 645–654, 2024.
- [227] Tao Sun, Zonglin Di, Pengyu Che, Chun Liu, and Yin Wang. Leveraging crowdsourced gps data for road extraction from aerial imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7509–7518, 2019.
- [228] Hao Tan and Mohit Bansal. LXMERT: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 5100–5111, Hong Kong, China, 2019. Association for Computational Linguistics.
- [229] Yong-Qiang Tan, Shang-Hua Gao, Xuan-Yi Li, Ming-Ming Cheng, and Bo Ren. Vecroad: Point-based iterative graph exploration for road graphs extraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8910–8918, 2020.
- [230] Jingjing Tao, Zhe Chen, Zhongchang Sun, Huadong Guo, Bo Leng, Zhengbo Yu, Yanli Wang, Ziqiong He, Xiangqi Lei, and Jinpei Yang. Seg-road: A segmentation network for road extraction based on transformer and cnn with connectivity structures. *Remote Sensing*, 15(6):1602, 2023.
- [231] Jonathon Taylor, Anna-Kaisa Viitanen, and Alonso Espinosa. Community-based monitoring of urban environmental data. *Buildings and Cities*, 2019.
- [232] Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.
- [233] Waldo R. Tobler. Smooth pycnophylactic interpolation for geographical regions. *Journal of the American Statistical Association*, 74(367):519–530, 1979.
- [234] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal

- Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.
- [235] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- [236] Abraham Cárdenas Tristán, Eduardo Javier Treviño Garza, Oscar Alberto Aguirre Calderón, Javier Jiménez Pérez, Marco Aurelio González Tagle, and Xanat Antonio Némiga. Spatial technologies to evaluate vectorial samples quality in maps production. *Investigaciones Geográficas, Boletín del Instituto de Geografía*, 2013(80):111–128, 2013.
- [237] Elisavet Tsiranidou, Daniele Treccani, Andrea Adami, Antonio Fernández, and Lucía Díaz-Vilariño. Automatic detection of podotactile pavements in urban environments through a deep learning-based approach on MLS/HMLS point clouds. *ISPRS International Journal of Geo-Information*, 14(12):492, 2025.
- [238] Ata Ullah, Muhammad Azeem, Humaira Ashraf, Abdullellah A. Alaboudi, Mamoon Humayun, and NZ Jhanjhi. Secure healthcare data aggregation and transmission in iot—a survey. *IEEE Access*, 9:16849–16865, 2021.
- [239] UN-Habitat. *World Cities Report 2020: The Value of Sustainable Urbanization*. United Nations Human Settlements Programme (UN-Habitat), Nairobi, Kenya, 2020. Accessed: 2025-04-15.
- [240] United Nations Department of Economic and Social Affairs. The sustainable development goals report 2022, 2022.
- [241] United Nations Framework Convention on Climate Change. Un climate change annual report 2022, 2022.

- [242] CE Utazi, Julia Thorley, V. Alegana, Mjgs Ferrari, Kristine Nilsen, Saki Takahashi, C. Jessica E. Metcalf, Justin Lessler, and AJ Tatem. A spatial regression model for the disaggregation of areal unit based data to high-resolution grids with application to vaccination coverage mapping. *Statistical Methods in Medical Research*, 28:3226 – 3241, 2018.
- [243] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30():, 2017.
- [244] Juan José Vázquez, Jamie Arjona, M^aPaz Linares, and Josep Casanovas-Garcia. A comparison of deep learning methods for urban traffic forecasting using floating car data. *Transportation Research Procedia*, 47:195–202, 2020.
- [245] Rajat Verma and Satish V Ukkusuri. Crosswalk detection from satellite imagery for pedestrian network completion. *Transportation research record*, 2678(7):845–856, 2024.
- [246] Patrick Vogel, Torsten Greiser, and Dirk Christian Mattfeld. Understanding bike-sharing systems using data mining: Exploring activity patterns. *Procedia-Social and Behavioral Sciences*, 20:514–523, 2011.
- [247] Tin Vu, Alberto Belussi, Sara Migliorini, and Ahmed Eldawy. A learning-based framework for spatial join processing: estimation, optimization and tuning. *The VLDB Journal*, pages 1–23, 2024.
- [248] Chunyang Wang, Yanmin Zhu, Tianzi Zang, Haobing Liu, and Jiadi Yu. Modeling inter-station relationships with attentive temporal graph convolutional network for air quality prediction. In *Proceedings of the 14th ACM international conference on web search and data mining*, pages 616–634, 2021.
- [249] Hong-Wei Wang, Xiao-Bing Li, Dongsheng Wang, Juanhao Zhao, Hong-di He, and Zhong-Ren Peng. Regional prediction of ground-level ozone using a hybrid sequence-to-sequence deep learning approach. *Journal of Cleaner Production*, 253:119841, 2020.

- [250] Hongjun Wang, Jiyuan Chen, Tong Pan, Zheng Dong, Lingyu Zhang, Renhe Jiang, and Xuan Song. STGformer: Efficient spatiotemporal graph transformer for traffic forecasting, 2024.
- [251] Jia-Xin Wang, Si-Bao Chen, Chris HQ Ding, Jin Tang, and Bin Luo. Semi-supervised semantic segmentation of remote sensing images with iterative contrastive network. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022.
- [252] Jiawei Wang, Renhe Jiang, Chuang Yang, Zengqing Wu, Makoto Onizuka, Ryosuke Shibasaki, Noboru Koshizuka, and Chuan Xiao. Large language models as urban residents: An llm agent framework for personal mobility generation. *arXiv preprint arXiv:2402.14744*, 2024.
- [253] Shaowen Wang, Tengfei Hu, Hu Xiao, Yao Li, Chuanrong Zhang, Huayi Ning, Rui Zhu, Zhenlong Li, and Xinyue Ye. GPT, large language models (LLMs) and generative artificial intelligence (GAI) models in geospatial science: A systematic review. *International Journal of Digital Earth*, 17(1):2353122, 2024.
- [254] Xiaoyang Wang, Yao Ma, Yiqi Wang, Wei Jin, Xin Wang, Jiliang Tang, Caiyan Jia, and Jian Yu. Traffic flow prediction via spatial temporal graph neural network. In *Proceedings of the web conference 2020*, pages 1082–1092, , 2020. .
- [255] Xiaoyang Wang, Yao Ma, Yiqi Wang, Wei Jin, Xin Wang, Jiliang Tang, Caiyan Jia, and Jian Yu. Traffic flow prediction via spatial temporal graph neural network. In *Proceedings of the web conference 2020*, pages 1082–1092, , 2020. .
- [256] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.
- [257] Yiheng Wang, Hexi Jin, and Guanjie Zheng. Ctrl: Cooperative traffic tolling via reinforcement learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 3545–3554, , 2022. .

- [258] Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, et al. Reinforcement learning for reasoning in large language models with one training example. *arXiv preprint arXiv:2504.20571*, 2025.
- [259] Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. Chain-of-table: Evolving tables in the reasoning chain for table understanding. In *International Conference on Learning Representations*, 2024.
- [260] N. A. Wardropa, W. C. Jochemma, T. J. Birda, H. R. Chamberlaina, David J. Clarkea, D. Kerra, Lars Bengtssona, S. Juranc, V. Seamand, and A. J. Tatema. Spatially disaggregated population estimates in the absence of national population and housing census data. In , page , , 2018. .
- [261] Hua Wei, Guanjie Zheng, Huaxiu Yao, and Zhenhui Li. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2496–2505, , 2018. .
- [262] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. *ArXiv*, abs/2109.01652, 2021.
- [263] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- [264] Zixi Wei, Lei Feng, Bo Han, Tongliang Liu, Gang Niu, Xiaofeng Zhu, and Heng Tao Shen. A universal unbiased method for classification from aggregate observations, 2023.
- [265] Galen Weld, Esther Jang, Anthony Li, Aileen Zeng, Kurtis Heimerl, and Jon E Froehlich. Deep learning for automatically detecting sidewalk accessibility problems

- using streetscape imagery. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility*, pages 196–209, 2019.
- [266] Xumeng Wen, Zihan Liu, Shun Zheng, Shengyu Ye, Zhirong Wu, Yang Wang, Zhijian Xu, Xiao Liang, Junjie Li, Ziming Miao, et al. Reinforcement learning with verifiable rewards implicitly incentivizes correct reasoning in base llms. *arXiv preprint arXiv:2506.14245*, 2025.
- [267] Wenchao Weng, Jin Fan, Huifeng Wu, Yujie Hu, Hao Tian, Fu Zhu, and Jia Wu. A decomposition dynamic graph convolutional recurrent network for traffic forecasting. *Pattern Recognition*, 142:109670, 2023.
- [268] Billy M Williams, Priya K Durvasula, and Donald E Brown. Urban freeway traffic flow prediction: application of seasonal autoregressive integrated moving average and exponential smoothing models. *Transportation Research Record*, 1644(1):132–141, 1998.
- [269] Robert Wolfe and Tanushree Mitra. The impact and opportunities of generative ai in fact-checking. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, pages 1531–1543, 2024.
- [270] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023.
- [271] xAI. Open release of Grok-1. Technical release, 2024.
- [272] Chenjun Xu, Bingbing Wen, Bin Han, Robert Wolfe, Lucy Lu Wang, and Bill Howe. Do language models mirror human confidence? exploring psychological insights to address overconfidence in llms. *arXiv preprint arXiv:2506.00582*, 2025.
- [273] Qianxiong Xu, Cheng Long, Liang Yu, and Chen Zhang. Road extraction with satellite images and partial road maps. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–14, 2023.

- [274] Zhenhua Xu, Yuxuan Liu, Lu Gan, Yuxiang Sun, Xinyu Wu, Ming Liu, and Lujia Wang. Rngdet: Road network graph detection by transformer in aerial images. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–12, 2022.
- [275] Zhenhua Xu, Yuxuan Liu, Yuxiang Sun, Ming Liu, and Lujia Wang. Rngdet++: Road network graph detection by transformer with instance segmentation and multi-scale features enhancement. *IEEE Robotics and Automation Letters*, 8(5):2991–2998, 2023.
- [276] Zhiyong Xu, Weicun Zhang, Tianxiang Zhang, Zhifang Yang, and Jiangyun Li. Efficient transformer for remote sensing image segmentation. *Remote Sensing*, 13(18):3585, 2021.
- [277] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.
- [278] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiayi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.
- [279] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.
- [280] Jie Yang, Linjiang Zheng, and Dihua Sun. Urban traffic flow prediction using a gradient-boosted method considering dynamic spatio-temporal correlations. In *Knowledge Science, Engineering and Management: 12th International Conference, KSEM 2019, Athens, Greece, August 28–30, 2019, Proceedings, Part II 12*, pages 271–283, , 2019. Springer, .
- [281] Li Yang, Yan Xu, Chunfeng Yuan, Wei Liu, Bing Li, and Weiming Hu. Improving visual grounding with visual-linguistic verification and iterative reasoning. In *Pro-*

- ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9499–9508, 2022.
- [282] Xiaofei Yang, Weijia Cao, Yao Lu, and Yicong Zhou. Hyperspectral image transformer classification networks. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022.
- [283] Huaxiu Yao, Fei Wu, Jintao Ke, Xianfeng Tang, Yitian Jia, Siyu Lu, Pinghua Gong, Jieping Ye, and Zhenhui Li. Deep multi-view spatial-temporal network for taxi demand prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, page , , 2018. .
- [284] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.
- [285] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.
- [286] Huanran Ye, Sheng Liu, Kun Jin, and Haohao Cheng. Ct-unet: An improved neural network based on u-net for building segmentation in remote sensing images. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 166–172. IEEE, 2021.
- [287] Yao Ye, Yong Cao, Yibo Dong, and Hua Yan. A graph neural network and transformer-based model for PM2.5 prediction through spatiotemporal correlation. *Environmental Modelling & Software*, 191:106501, 2025.
- [288] Ji Won Yoon, Fabio Pinelli, and Francesco Calabrese. Cityride: a predictive bike sharing journey advisor. In *2012 IEEE 13th International Conference on Mobile Data Management*, pages 306–311. IEEE, 2012.

- [289] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, (.), 2017.
- [290] Danlin Yu. Emerging frontiers in urban data science: Climate resilience, equity, and health risk reduction for a sustainable future. *Discover Cities*, 2025. Guest Editor’s Introduction.
- [291] Haiyang Yu, Zhihai Wu, Shuqin Wang, Yunpeng Wang, and Xiaolei Ma. Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks. *Sensors*, 17(7):1501, 2017.
- [292] Rose Yu, Yaguang Li, Cyrus Shahabi, Ugur Demiryurek, and Yan Liu. Deep learning: A generic approach for extreme condition traffic forecasting. In *Proceedings of the 2017 SIAM international Conference on Data Mining*, pages 777–785, , 2017. SIAM, .
- [293] Yuan Yuan, Xiaofeng Shi, and Junyu Gao. Building extraction from remote sensing images with deep learning: A survey on vision techniques. *Computer Vision and Image Understanding*, 251:104253, 2025.
- [294] Zhenghang Yuan, Lichao Mou, Yuansheng Hua, and Xiao Xiang Zhu. Rrsis: Referring remote sensing image segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, 2024.
- [295] Yang Zhan, Zhitong Xiong, and Yuan Yuan. Rsvg: Exploring data and models for visual grounding on remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–13, 2023.
- [296] Cheng Zhang, Wanshou Jiang, Yuan Zhang, Wei Wang, Qing Zhao, and Chenjie Wang. Transformer and cnn hybrid deep neural network for semantic segmentation of very-high-resolution remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–20, 2022.

- [297] Jin Zhang, Yimin Yang, Xiaoheng Wu, and Sen Li. Spatio-temporal transformer and graph convolutional networks based traffic flow prediction. *Scientific Reports*, 15:24299, 2025.
- [298] Junbo Zhang, Yu Zheng, Dekang Qi, Ruiyuan Li, and Xiuwen Yi. Dnn-based prediction model for spatio-temporal data. In *Proceedings of the 24th ACM SIGSPATIAL international conference on advances in geographic information systems*, pages 1–4, , 2016. .
- [299] Rui Zhang and Norhashidah Awang. A novel ST-iTransformer model for spatio-temporal ambient air pollution forecasting. *Journal of Big Data*, 12:101, 2025.
- [300] Taoyi Zhang, Yubo Wang, and Zhicheng Wei. Mcl-stgat: Taxi demand forecasting using spatio-temporal graph attention network with markov cluster algorithm. *Int. J. Innovative Comput. Inf. Control*, 19(4):1251–1264, 2023.
- [301] Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B. Hashimoto. Benchmarking large language models for news summarization. *Transactions of the Association for Computational Linguistics*, 12:39–57, 2024.
- [302] Wei Zhang, Miaoxin Cai, Tong Zhang, Yin Zhuang, and Xuerui Mao. EarthGPT: A universal multimodal large language model for multisensor image comprehension in remote sensing domain. *IEEE Transactions on Geoscience and Remote Sensing*, 62:5917820, 2024.
- [303] Yivan Zhang, Nontawat Charoenphakdee, Zhenguo Wu, and Masashi Sugiyama. Learning from aggregate observations. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS’20, Red Hook, NY, USA, 2020. Curran Associates Inc.
- [304] Yuxiang Zhang, Nicholas Bolten, Sachin Mehta, and Anat Caspi. Ape: An open and shared annotated dataset for learning urban pedestrian path networks. *arXiv preprint arXiv:2303.02323*, 2023.

- [305] Yuxiang Zhang, Bill Howe, Sachin Mehta, Nicholas-J Bolten, and Anat Caspi. Pathwaybench: Assessing routability of pedestrian pathway networks inferred from multi-city imagery. *arXiv preprint arXiv:2407.16875*, 2024.
- [306] Zheyuan Zhang, Daniel Zhang-Li, Jifan Yu, Linlu Gong, Jinchang Zhou, Zhiyuan Liu, Lei Hou, and Juanzi Li. Simulating classroom education with llm-empowered agents. *arXiv preprint arXiv:2406.19226*, 2024.
- [307] Zilun Zhang, Tiancheng Zhao, Yulong Guo, and Jianwei Yin. RS5M and GeoRSCLIP: A large-scale vision-language dataset and a large vision-language model for remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–18, 2024.
- [308] Guyu Zhao, Hongdou He, Yifang Huang, and Jiadong Ren. Near-surface pm2. 5 prediction combining the complex network characterization and graph convolution neural network. *Neural Computing and Applications*, 33:17081–17101, 2021.
- [309] Lirui Zhao, Yue Yang, Kaipeng Zhang, Wenqi Shao, Yuxin Zhang, Yu Qiao, Ping Luo, and Rongrong Ji. Diffagent: Fast and accurate text-to-image api selection with large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6390–6399, 2024.
- [310] Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.
- [311] Dewu Zheng, Yanlin Wang, Ensheng Shi, Hongyu Zhang, and Zibin Zheng. How well do llms generate code for different application domains? benchmark and evaluation. *arXiv preprint arXiv:2412.18573*, 2024.
- [312] Zibin Zheng, Yatao Yang, Jiahao Liu, Hong-Ning Dai, and Yan Zhang. Deep and embedded learning approach for traffic flow prediction in urban informatics. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3927–3939, 2019.
- [313] F Zhonglianga and W Jianhuaa. Entity matching in vector spatial data. In *Proceedings of the XXI ISPRS Congress*, pages 1467–1472. Citeseer, 2008.

- [314] Kai Zhou, Yan Xie, Zhan Gao, Fang Miao, and Lei Zhang. Funet: A novel road extraction network with fusion of location data and remote sensing imagery. *ISPRS International Journal of Geo-Information*, 10(1):39, 2021.
- [315] Chaoyang Zhu, Yiyi Zhou, Yunhang Shen, Gen Luo, Xingjia Pan, Mingbao Lin, Chao Chen, Liujuan Cao, Xiaoshuai Sun, and Rongrong Ji. Seqtr: A simple yet universal network for visual grounding. In *European Conference on Computer Vision*, pages 598–615. Springer, 2022.
- [316] Terry Yue Zhuo, Minh Chien Vu, Jenny Chim, Han Hu, Wenhao Yu, Ratnadira Widiasari, Imam Nur Bani Yusuf, Haolan Zhan, Junda He, Indraneil Paul, Simon Brunner, Chen Gong, Thong Hoang, Armel Randy Zebaze, Xiaoheng Hong, Wen-Ding Li, Jean Kaddour, Ming Xu, Zhihan Zhang, Prateek Yadav, Naman Jain, Alex Gu, Zhoujun Cheng, Jiawei Liu, Qian Liu, Zijian Wang, Binyuan Hui, Niklas Muenighoff, David Lo, Daniel Fried, Xiaoning Du, Harm de Vries, and Leandro von Werra. BigCodeBench: Benchmarking code generation with diverse function calls and complex instructions. In *International Conference on Learning Representations*, 2025.
- [317] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.
- [318] Zefang Zong, Jie Feng, Kechun Liu, Hongzhi Shi, and Yong Li. Deepdpm: Dynamic population mapping via deep neural network. In *Proceedings of the AAAI conference on artificial intelligence*, pages 1294–1301, , 2019. .