

Identifying Healthcare Access and Enhancing Geospatial Analysis with Generative AI

Zachary Sherman

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University in
fulfillment of the requirements for the degree of

Master of Science

In

Geography

Junghwan Kim, Chair

Mengxi Zhang, Co-Chair

Tom Crawford

June 4th, 2025

Blacksburg, Virginia

Keywords: Healthcare Accessibility, 2SFCA, Spatial Inequality, Transportation, Geospatial Data;
Dashboard; Fine-tuned; ChatGPT; Large Language Model

Identifying Healthcare Access and Enhancing Geospatial Analysis with Generative AI

Zachary Sherman

Abstract

This thesis brings together geospatial modeling and generative artificial intelligence to address healthcare accessibility and automation of spatial analysis. The first study examines disparities in dental care access across six regions of Virginia by comparing driving and public transit modes for both all dental clinics and those accepting Medicaid. Using a modified two-step floating catchment area (2SFCA) method, the research quantifies access based on travel time, supply-demand ratios, and vehicle ownership. Results show that public transit accessibility is significantly lower and more unequal than driving access, particularly for Medicaid recipients, with variation across regions. Spatial error models further reveal demographic factors, such as poverty, race, and vehicle access, influence accessibility patterns.

The second study fine-tunes OpenAI's GPT-4o-mini model to convert natural language queries into executable Python code for geospatial analysis. Trained on over 600 geospatial prompt-completion pairs using Virginia health data, the model achieves an 89.7% accuracy rate, improving significantly over the baseline. It integrates spatial reasoning, fuzzy geographic matching, and modular function calls to reduce execution errors and enhance usability. Together, these studies demonstrate how AI and geospatial science can jointly address inequities in healthcare access while making spatial tools more accessible to policymakers, researchers, and the general public.

Identifying Healthcare Access and Enhancing Geospatial Analysis with Generative AI

Zachary Sherman

General Audience Abstract

Access to dental care shouldn't depend on where you live or whether you own a car—but in many parts of Virginia, it does. This thesis looks at how easy it is for people to reach dental clinics, especially those who rely on Medicaid or public transportation. It shows that driving makes it much easier to get to care than taking the bus, and that people in poverty or rural areas often face serious barriers. To help make this kind of analysis easier and more accessible, the second part of this thesis uses artificial intelligence to turn plain language questions like “How many clinics are near me?”, into computer code that can search maps and analyze data. The AI model was trained to understand real-world health and location data, making it possible for everyday users and decision-makers to explore health access through simple questions. Together, this work combines research and technology to improve how we understand and respond to health care inequality for non-technical users.

Dedication

This thesis is dedicated to my family: Mom, Dad, Katie, Joseph, Thomas, Zach, Karli, Elise, Alex, Sammy, Theo, Annie, Henry, Julianna, Dashy, Micah, Gracie and all future Sherman's.

Acknowledgments

I would like to express my deepest gratitude to my advisors, Dr. Junghwan Kim and Dr. Mengxi Zhang, for their guidance, support, and mentorship throughout the course of this research. I also sincerely thank Dr. Tom Crawford, my committee member and the Department Head of Geography, for his thoughtful feedback, continued encouragement, and providing access to resources for all students.

This work would not have been possible without the generous financial and technical support from the Virginia Tech Carilion School of Medicine, the American Dental Association, the Institute for Society, Culture, and Environment (ISCE), the Virginia Tech Poole Endowment Fund, and OpenAI's Research Access Program. I am especially grateful to Karen Bland for her instrumental help in processing and coordinating these funding efforts.

I extend my appreciation to the Department of Geography and the College of Natural Resources and Environment (CNRE) at Virginia Tech for providing a collaborative and intellectually stimulating academic environment.

Additional thanks goes to the American Association of Geographers (AAG) for awarding me a travel grant to present this research at the 2025 AAG Conference in Detroit, and to the North American Cartographic Information Society (NACIS) for funding my attendance at the 2025 NACIS Conference in Tacoma, Washington.

I also want to thank the incredible friends I've made in the Department of Geography and in the Smart Cities for Good Lab. You are some of the most genuine, fun, and down-to-earth people I've ever met, and I feel lucky to have shared this chapter of life with you. Your support, laughter, and late-night work sessions made even the most stressful days manageable. I hope we stay in touch, because each of you has left a lasting impact on my life.

Preface/Attribution

For complete disclosure, **Manuscript 1: Geospatial Access to Dental Care in Virginia: Insights from Driving and Public Transit Analysis**, which was recently submitted and currently under revision in the **Journal of the American Dental Association**, the following individuals provided contribution to improve the quality of the manuscript and are all co-authors:

- Shashank Karki: (Co-Author)
- Tegwyn Brickhouse: (Co-Author)
- Marko Vujicic: (Co-Author)
- Kamyar Nasseh: (Co-Author)
- Changzhen Wang: (Co-Author)
- Hyun Kim: (Co-Author)
- Junghwan Kim: (Co-Author | Co-Advisor)
- Mengxi Zhang: (Corresponding Author | Co-Advisor)

For complete disclosure, **Manuscript II: Generative AI for Geospatial Analysis: Fine-Tuning ChatGPT to Convert Natural Language into Python-Based Geospatial Computations**, which was recently submitted to and accepted in the **ISPRS International Journal of Geo-Information**, the following individuals provided contribution to improve the quality of the manuscript and are co-authors:

- Sandesh Sharma Dulal: (Co-Author)
- Jin-Hee Cho: (Co-Author)
- Mengxi Zhang: (Co-Advisor)
- Junghwan Kim: (Corresponding Author | Co-Advisor)

These chapters, including all other parts of my thesis, are my own work, but they benefitted from general input and ideas from these individuals.

Table of Contents

I. Introduction	01
II. Manuscript I: Geospatial Access to Dental Care in Virginia: Insights from Driving and Public Transit Analysis	03-23
1. Introduction	04
2. Methods.....	05
2.1. Study Region.....	05
2.2. Measures	06
2.3. Dataset and Data Processing	06
2.4. Data Analysis.....	06
3. Results	09
3.1. Driving and Transit Accessibility Scores	09
3.2. Accessibility Inequality	13
4. Discussion	15
5. Conclusion.....	16
6. References... ..	17
7. Supplementary Material	20
III. Manuscript II: Generative AI for Geospatial Analysis: Fine-Tuning ChatGPT to Convert Natural Language into Python-Based Geospatial Computations	24-54
1. Introduction	25
1.1 Background.....	25
1.2 State-of-the-art: The intersection of LLMs and Geospatial Analysis.....	26
1.3 Research goal and questions.....	27
2. Materials & Methods.....	28
2.1 Overview of Research Design.....	28
2.2 As-Is Model: OpenAI’s GPT-4o-mini	29
2.3 Fine-Tuned Model	31
2.4 Evaluation Setup for the Performance Comparison	35
2.5 External Validation for Geographic Generalizability... ..	36
3. Results	37
3.1 Performance Comparison between As-Is Model and Fine-Tuned Model	37

3.2 Analysis of Error Types of As-Is Model and Fine-Tuned Model.....	40
4. Discussion	42
5. Conclusions	46
6. Appendix A... ..	47
7. References... ..	51
IV. Conclusion.....	57

I. Introduction

Geographic disparities in healthcare access represent a persistent challenge in the United States, with dental care particularly affected by uneven provider availability, socioeconomic barriers, and limited transportation options. Where individuals live, whether they own a vehicle, and whether providers accept public insurance such as Medicaid are all critical factors that shape access to care. These challenges are especially pronounced for low-income populations and residents of rural or transit-scarce regions, who often face disproportionate obstacles in reaching dental services despite demonstrated links between oral health and broader systemic health outcomes (Rahman et al., 2024; Serón et al., 2023; Shariff et al., 2023).

Geospatial modeling has emerged as a powerful tool for evaluating healthcare accessibility, providing spatially explicit insights into where gaps exist and which populations are underserved. Yet, existing approaches frequently rely on overly simplified assumptions, such as using only driving time as a proxy for access, and often fail to account for real-world constraints such as transit availability, provider capacity, or Medicaid acceptance. Additionally, access disparities are rarely quantified using distributional measures like the Gini coefficient, despite growing awareness of the need to understand not just average access, but who benefits and who is left behind. These methodological shortcomings limit the ability of public health researchers and policymakers to develop targeted, equity-driven interventions (Luo & Wang, 2003; Wolfe et al., 2020).

At the same time, a parallel challenge lies in the usability of spatial analysis tools. Most geographic information system (GIS) platforms require specialized technical expertise, posing a barrier for stakeholders such as public health officials, nonprofit staff, or local planners who may lack formal GIS training. While large language models (LLMs) like ChatGPT offer promising pathways for translating plain-language questions into geospatial computations, they frequently fail to handle complex spatial reasoning tasks, such as spatial joins or drive-time calculations, without domain-specific adaptation (Tao & Xu, 2022; Zhang et al., 2024). As a result, there is a pressing need to develop AI-powered geospatial tools that are both accurate and accessible, particularly for non-technical users.

This thesis addresses both of these gaps through a dual approach. The first study investigates geospatial disparities in dental care access across six regions in Virginia, comparing driving and

public transit accessibility for both general and Medicaid-accepting dental clinics. Using a modified two-step floating catchment area (2SFCA) method and Gini coefficients, the study highlights significant inequalities, particularly for Medicaid recipients and transit-dependent populations, and identifies sociodemographic predictors of access. The second study builds on this foundation by fine-tuning OpenAI's GPT-4o-mini model to interpret natural language questions and generate executable Python code for spatial analysis. Trained on over 600 prompt-response pairs grounded in Virginia health data, the model achieves an accuracy rate of nearly 89.7%, significantly outperforming the baseline. Notably, this framework allows for real-time spatial querying within interactive dashboards or chatbot interfaces, reducing technical barriers and enabling broader participation in geospatial decision-making by policymakers and everyday people alike.

Together, these studies illustrate how rigorous geospatial analysis and generative AI can be strategically combined to (1) expose critical inequities in healthcare access and (2) democratize the tools needed to explore and address those inequities. While grounded in the case of healthcare in Virginia, the findings and methodological innovations developed here offer scalable frameworks for advancing equity-oriented spatial analysis in public health and beyond.

This thesis is structured as two standalone but complementary manuscripts. Manuscript I examines regional disparities in dental care accessibility across six Virginia regions using transit- and driving-based 2SFCA methods, Gini coefficients, and regression analysis. Manuscript II presents the development, training, and evaluation of a fine-tuned large language model capable of converting natural language into executable geospatial Python code, offering a proof of concept for scalable, AI-powered spatial tools for non-technical users for academic and industry settings.

II. Manuscript I: Geospatial Access to Dental Care in Virginia: Insights from Driving and Public Transit Analysis

Zach Sherman; Shashank Karki; Tegwyn Brickhouse; Marko Vujicic; Kamyar Nasseh; Changzhen Wang; Hyun Kim; Junghwan Kim, Mengxi Zhang*

*Corresponding Author

Abstract:

This study aims to investigate: (1) how driving and public transit access to dental clinics differ for all dental clinics versus those participating in Medicaid, and (2) how sociodemographic factors are associated with driving and public transit accessibility in Virginia.

Methods:

Six regions in Virginia were selected in this study based on the availability of transit data. A modified two-step floating catchment area (2SFCA) method considering vehicle ownership was applied to measure geospatial accessibility scores for the overall dental care providers and those participating in Medicaid. Inequality in accessibility among census block groups were analyzed using Gini coefficients.

Results:

Public transit accessibility to dental care services is lower compared to driving, regardless of Medicaid acceptance. Driving also provides more equitable access than public transit.

Conclusions:

Access to dental care services is much more challenging using public transportation compared with driving, especially in smaller regions and among Medicaid beneficiaries in Virginia.

Practical implications:

To improve access to dental care in Virginia, efforts should focus on encouraging dentists to accept Medicaid, establishing new dental clinics near public transportation hubs and major roads, utilizing ride-sharing support, and advocating for the development of robust public transportation systems.

Introduction

Geospatial accessibility means how easy it is for people to get to a location, depending on how far it is, what transportation they can use, and whether services are available. Geospatial dental care access is crucial for overall health, as limited access to dental services often leads to untreated oral health issues that can contribute to systemic health conditions.^{1,2} In the United States (U.S.), geospatial dental care accessibility varies significantly across regions and populations, with low-income individuals, Medicaid recipients, and residents of rural areas experiencing particularly low levels of access.^{1,3} In Virginia, the overall density of dentists (63.19 per 100,000 population) is higher than the national average (61.06 per 100,000 population).⁴ However, significant disparities exist across the state, with urbanized regions in Northern Virginia generally having better access than rural areas in Southwest and Central Virginia.⁴ These disparities are further compounded by differences in insurance coverage. Virginia has one of the lowest rates of dentist participation in Medicaid nationwide (26.4% in Virginia compared to the national average of 43%), raising concerns about dental care access and utilization.⁵ In 2022, only 17% of adults with Medicaid in Virginia received any dental services.⁶ Understanding those disparities could help improving dental and broader health outcomes and healthcare planning.

However, previous studies have not measured dental care geospatial access comprehensively and accurately. Most research relies solely on a single mode of transportation, primarily driving, without considering that many individuals rely on public transportation to access care, especially low-income population.⁷ Additionally, most studies measure geospatial accessibility based only on driving time or distance, overlooking the critical supply-and-demand dynamics. Patients (demand) often compete for access to limited medical resources, given the number of available dental care providers (supply).⁸ Ignoring these dynamics can result in an incomplete understanding of dental care accessibility. Furthermore, many studies fail to account for insurance acceptance in clinics, despite its significant role in determining access to care in the U.S.

This paper aims to fill those research gaps by measuring geospatial accessibility to dental care, considering both car travel and public transit, the competition between supply and demand, and whether clinics participating in Medicaid. Specially, we asked the following research question:

whether driving and public transit access to dental clinics differs between all dental clinics and those participating in Medicaid in Virginia.

Methods

Study region

The study focused on six regions within Virginia, including Altavista, Lynchburg, Richmond, Staunton-Harrisonburg, Williamsburg, and Winchester. These locations were chosen based on the availability of comprehensive transit data. Those locations also represent a mix of urban and rural settings, with diverse demographic characteristics (Table 1). The six regions are composed of 1,098 census block groups which were used as units of analysis.

Table 1. Demographic information of the selected six regions in Virginia

Region	Total Population	% Rural Population	Med. Age	% White	% High Edu.	Unemp. Rate	% Poverty	% No Vehicle	% Tran. Com.	% Uninsu.	% Medicaid
Altavista	55,518	56.86%	43.1	78.9	24.6	4.9	10.0	5.2	0.1	6.5	17.0
Lynchburg	79,166	1.62%	28.4	62.7	38.0	5.4	17.5	10.6	2.2	6.9	21.8
Richmond	227,171	0.00%	34.5	44.6	44.1	6.1	19.5	14.2	4.3	9.6	22.9
Staunton	25,581	2.14%	41.0	81.8	33.6	5.1	11.4	8.8	0.5	8.2	18.9
Harrisonburg	51,784	0.62%	25.4	71.0	35.2	6.7	27.2	7.2	1.5	10.6	18
Williamsburg	15,486	0.12%	24.6	69.2	50.6	6.0	13.7	7.1	1.2	6.0	8.5
Winchester	28,103	0.00%	36.6	72.9	32.3	5.2	16.8	10.0	1.2	12	19.1
Virginia (State)	8,624,511	24.4%	38.7	63.5	41.0	4.4	10.0	6.1	3.0	7.4	16.6
U.S. (Nation)	331,097,593	19.9%	38.5	65.9	34.3	5.3	12.5	8.3	3.8	8.7	21.2

Note. Data Source: American Community Survey (ACS) 5-Year Estimates for each city representing the study regions except for Altavista (Campbell County). % Rural Population was obtained from the County-level Urban and Rural information for the 2020 Census. Columns: Total Population. Median Age. % White Alone. % High Edu (Bachelor’s Degree or Higher) Unemployment Rate. % below Poverty Level. % No Vehicle Available (Households). % Public Transit Commuters. % Uninsured. % Medicaid/means-tested.

Dataset and data processing

A comprehensive list of dental care providers data, including locations of clinics, full-time equivalent (FTE), whether care providers accept Medicaid, were extracted from the 2022 American Dental Association office database.⁹ FTE was calculated by dividing one by the number of clinics in which each doctor's name appeared in the directory. Sociodemographic data, including population density, non-White percentage, and poverty percentage, were extracted from the American Community Survey (ACS) 5-Year Estimates at the census block group level. Data sources of all variables included in the analysis were presented in supplementary Table 1.

We generated an origin-destination matrix travel time considering both driving and public transit. For driving, the origins are the centroid of each census block group, and the destinations are the location of dental clinics. Driving time was then estimated using Google Maps Distance API which compute accurate travel times based on high-resolution real-world transportation network and traffic congestion situation.¹⁰⁻¹² For public transit, for the same origin and destination pairs, travel time was measured using General Transit Feed Specification (GTFS) data—a standardized format that provides public transportation schedules and geographic data and applying an *r5r* package—an R package designed for transit network analysis.^{12,13} We included only bus transportation in this study, as metro and other forms of public transit are not available in the selected areas. Travel time for public transit include in-vehicle travel time and out-of-vehicle travel time, including walking and waiting time for transfer, to capture realistic travel time estimation. A detailed explanation of the public transit travel time analysis is provided in Supplementary Figure 1. The origin-destination matrix travel time calculated for both driving and public transit was used to define catchment areas which will be explained for accessibility score analysis.

Data analysis

Accessibility score

First, we measured accessibility scores through both driving and taking public transit across two groups: all dental clinics and those clinics participating in Medicaid, applying the two-step floating catchment area (2SFCA) model.⁸ For the analysis of clinics participating in Medicaid, we only focused the analysis on adult Medicaid beneficiaries and none-pediatric dental care providers.

The 2SFCA method works in two steps. In the first step, a catchment area was defined around each dental provider based on 30 minutes travel time. This threshold was chosen based on accepted travel time through driving and public transit from previous studies.¹⁴⁻¹⁶ Within this catchment area, the total population that can reach the dental provider is summed up. The dental clinics capacity, represented by FTE, was then divided by the total population within the catchment area to calculate a provider-to-population ratio. This ratio indicates how much healthcare service is available relative to the population in that area. In the second step, for each census block group, all clinics within the 30 minutes travel time from that block groups' centroid were identified. The previously calculated provider-to-population ratios for these clinics were summed to produce an overall accessibility score for that block group. Lower 2SFCA accessibility scores suggest an overwhelming demand for the available dental services, indicating potential delays in treatment for patients, and vice versa. Considering the importance impact of vehicle ownership on individuals' choice of travel, we include the rate of vehicle ownership in the analysis.

Equations 1 and 2 illustrate the 2SFCA accessibility metric on driving ($A_i^{driving}$) and public transit ($A_i^{transit}$).

$$A_i^{driving} = \sum_{j=1}^N \frac{D_j \times f(DT_{ij})}{\sum_{k=1}^N \alpha_k \times P_k \times f(DT_{kj}) + \alpha_k \times P_k \times f(PT_{kj})} \dots (1)$$

$$A_i^{transit} = \sum_{j=1}^N \frac{D_j \times f(PT_{ij})}{\sum_{k=1}^N \alpha_k \times P_k \times f(DT_{kj}) + (1 - \alpha_k) \times P_k \times f(PT_{kj})} \dots (2)$$

where, i represents the specific census block, N is the total number of census block groups P_k denotes the population at census block k , and α_k reflects the vehicle ownership rate at the census tracts. Note that since vehicle ownership rate data are not available at the census block group, so census tract data are the best available alternative. D_j indicates the total FTEs at census block j , indicating the level of dental care supply. The function $f(.)$ yields a value of 1 for driving (DT_{ij}) and transit (PT_{ij}) travel times between census block group i and j within a 30-minute threshold, and 0 otherwise. This ensures that only clinics reachable within 30 minutes are included in the calculation of accessibility scores. The accessibility score can be interpreted as a provider-to-

population ratio (e.g., number of dental providers per population) and a higher score suggests better geospatial accessibility. We calculated two accessibility scores in each region. We then compared accessibility scores between driving and taking public transit in each region using a two-sample *t*-test.

Accessibility inequality

The accessibility score presents us the level of average access in each region. Thus, we utilized the Gini coefficient to measure the degree of inequality in the distribution of accessibility scores among census block groups in each region. Equation 3 illustrates the equation to calculate the Gini Coefficient (*G*). The Gini coefficient ranging from 0 (perfect equality) to 1 (maximum inequality), with a higher Gini coefficient indicates greater disparities in access, while a lower value suggests a more equitable distribution of dental services in each selected region.¹⁷

$$G = \frac{\sum_{i=1}^N \sum_{j=1}^N |x_i - x_j|}{2N^2\bar{x}} \dots (3)$$

G represents the Gini index. The variable *N* denotes the total number of census block groups analyzed. The terms x_i and x_j represent the dental care accessibility scores for two different areas, and the summation symbol Σ accounts for the differences between all pairs of areas. The term \bar{x} is the average accessibility score, obtained by dividing the total accessibility score by the number of census block groups. This coefficient provides a single, interpretable value to quantify inequality in access, allowing for straightforward comparisons across different regions or population groups.

Lorenz curves were then used to visualize Gini coefficient, which is the ratio of the area between the Lorenz curve and the line of equality to the total area under the line of equality (supplementary Figure 2). The line of equality, shown as a 45-degree diagonal, represents perfect equality. The further the Lorenz curve falls below this diagonal, the greater the inequality.

R Studio was used to build GTFS data, Python for statistical analysis and data processing, and ArcGIS Pro for spatial visualization and mapping of accessibility results.

3. Results

3.1. Driving and transit accessibility scores

Table 2 presents the descriptive statistics of driving and transit accessibility scores for the overall dental clinics and clinics participating in Medicaid in the six selected regions in Virginia.

Table 2. Descriptive statistics of driving and transit accessibility scores of all dental clinics and clinics participating in Medicaid/CHIP applying two sample *t*-test

		Accessibility Score (All Clinics)		Accessibility Score (Clinics Participating in Medicaid)	
		Driving	Transit	Driving	Transit
Altavista	Mean	0.0024	0.0002	0.0021	0.0001
	SD	0.0008	0.0003	0.0011	0.0005
Lynchburg	Mean	0.0037	0.0001	0.0085	0.0002
	SD	0.001	0.0002	0.0024	0.0005
Winchester	Mean	0.0088	0.0002	0.0345	0.0008
	SD	0.0013	0.0005	0.0052	0.0024
Williamsburg	Mean	0.0083	0.0001	0.0109	0.0002
	SD	0.0024	0.0003	0.0044	0.0010
Staunton-Harrisonburg	Mean	0.0054	0.0001	0.0079	0.0001
	SD	0.0022	0.0002	0.0035	0.0005
Richmond	Mean	0.0094	0.0001	0.0099	0.0001
	SD	0.0022	0.0003	0.0021	0.0003

Note: Accessibility scores for all clinics cannot be directly compared with those for clinics participating in Medicaid/CHIP because different population groups were used in the analysis. However, accessibility scores can be compared within each clinic type. SD refers to standard deviation. All the comparison were statistically significant; *P-value* <0.001

The results demonstrated a remarkable discrepancy in accessibility scores between driving and taking public transit. Across all six regions in the study area, the average driving accessibility score was consistently higher than the transit accessibility score. Moreover, the difference in accessibility scores between driving and transit tended to be larger in areas with higher population. For example, the average difference in driving and transit accessibility scores ranged from 24 and 21 times in Altavista, the region with highest rural population, to 94 and 99 times in Richmond, one of the urban region, across all clinics and clinics participating in Medicaid, respectively. Notably, the differences in accessibility scores between driving and transit were more pronounced

for clinics participating in Medicaid compared to all clinics, underscoring a greater disparity in access to dental clinics participating in Medicaid.

Figures 1 and 2 visualize driving and transit accessibility scores of the six regions in Virginia in census block groups. Accessibility was divided into seven categories, with darker colors indicating better accessibility. The classification intervals were manually defined based on the distribution of driving and transit accessibility scores to provide visually meaningful distinctions across different accessibility levels. Different scales were used for all dental clinics and dental clinics participating in Medicaid. These figures showed a general trend where census block groups along major transportation networks, including interstates and bus routes, tend to have higher accessibility scores than other census block groups. Also, accessibility scores were higher in census block groups closer to urban centers. Some regions, however, demonstrated more complex spatial patterns, especially for driving. For instance, in Staunton-Harrisonburg, census block groups located between Staunton and Harrisonburg had higher driving accessibility scores than the downtowns of both regions. This was because these areas have convenient access to dental clinics in both regions.

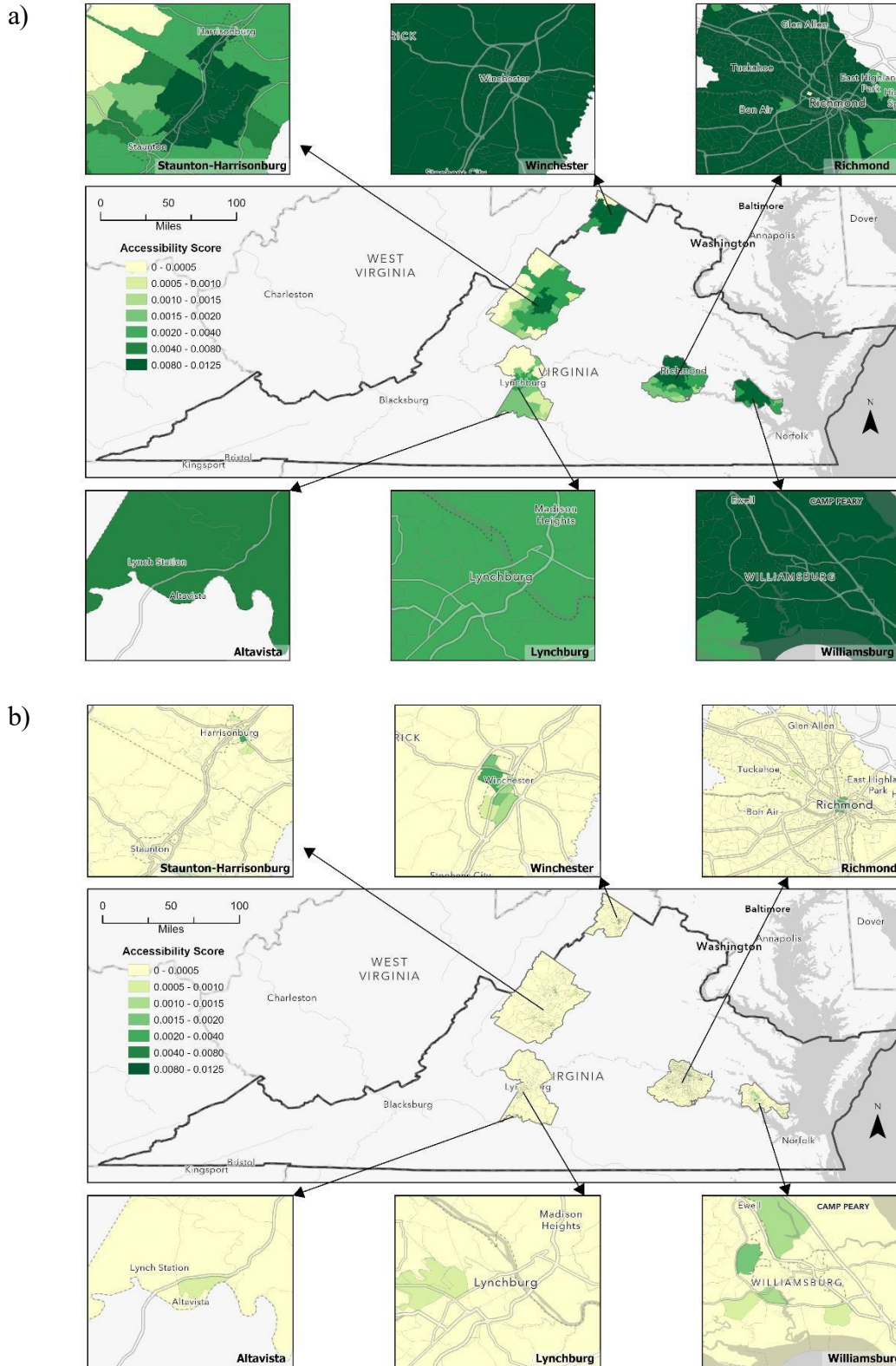


Figure 1. Maps of accessibility to all dental clinics by (a) driving and (b) public transit.

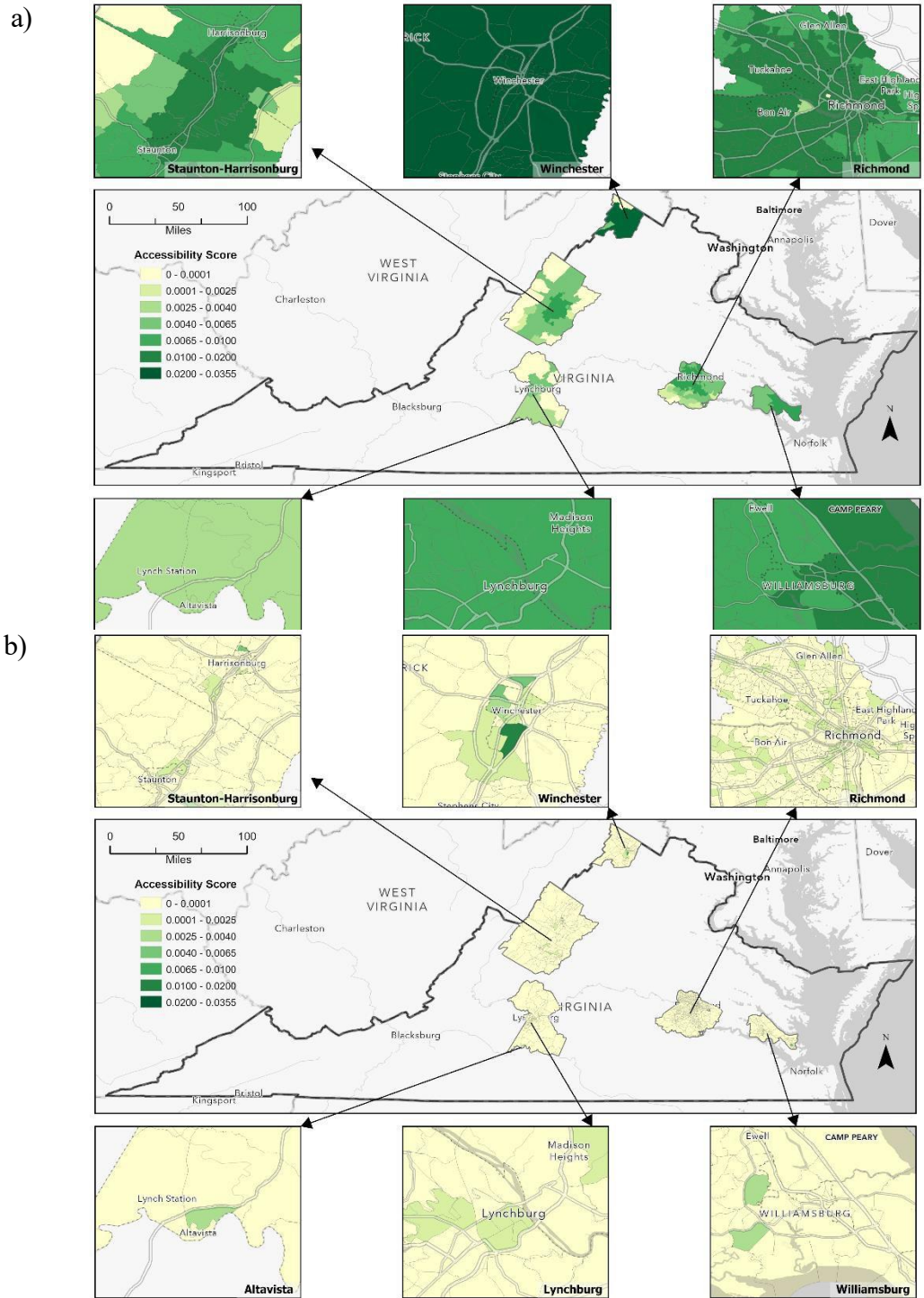


Figure 2. Maps of accessibility to dental clinics participating in Medicaid/CHIP by (a) driving and (b) public transit.

3.2. Accessibility inequality

Table 3 reports the value of Gini coefficients in each selected region. **Figure 3** illustrates Lorenz curves that visualize Gini coefficients. For driving accessibility to all dental care providers, the Gini coefficients varied widely, from as low as 0.0289 in Winchester to 0.2109 in Staunton-Harrisonburg, suggesting the greatest level of inequality across the six regions. Gini coefficients of driving accessibility to clinics participating in Medicaid showed a similar pattern but with some regions like Williamsburg and Altavista (Gini coefficients = 0.2277 and 0.3036) experiencing a greater level of overall inequality.

Table 3. Gini coefficients of driving and transit accessibility scores for all dental clinics and clinics participating in Medicaid

Region	Gini coefficients of the Accessibility Score (All Clinics)		Gini coefficients of the Accessibility Score (Clinics participating in Medicaid)	
	Driving	Transit	Driving	Transit
Altavista	0.1697	0.8037	0.3036	0.9486
Lynchburg	0.0896	0.8236	0.0701	0.8057
Winchester	0.0289	0.8679	0.0335	0.9001
Williamsburg	0.1537	0.8222	0.2277	0.9698
Staunton-Harrisonburg	0.2109	0.8323	0.1862	0.9099
Richmond	0.1288	0.8541	0.0876	0.9162

In terms of comparing inequality in driving and transit accessibility scores for all clinics, the Gini coefficients for transit were higher than those for driving, ranging from 0.8037 in Altavista to 0.8679 in Winchester. Comparing all clinics with the clinics participating in Medicaid, the Gini coefficients of accessibility scores were higher for clinics participating in Medicaid than for all clinics. For instance, in Altavista, the Gini coefficient of transit accessibility scores for clinics participating in Medicaid was 0.9486, which was higher than the Gini coefficient of transit accessibility scores for all clinics.

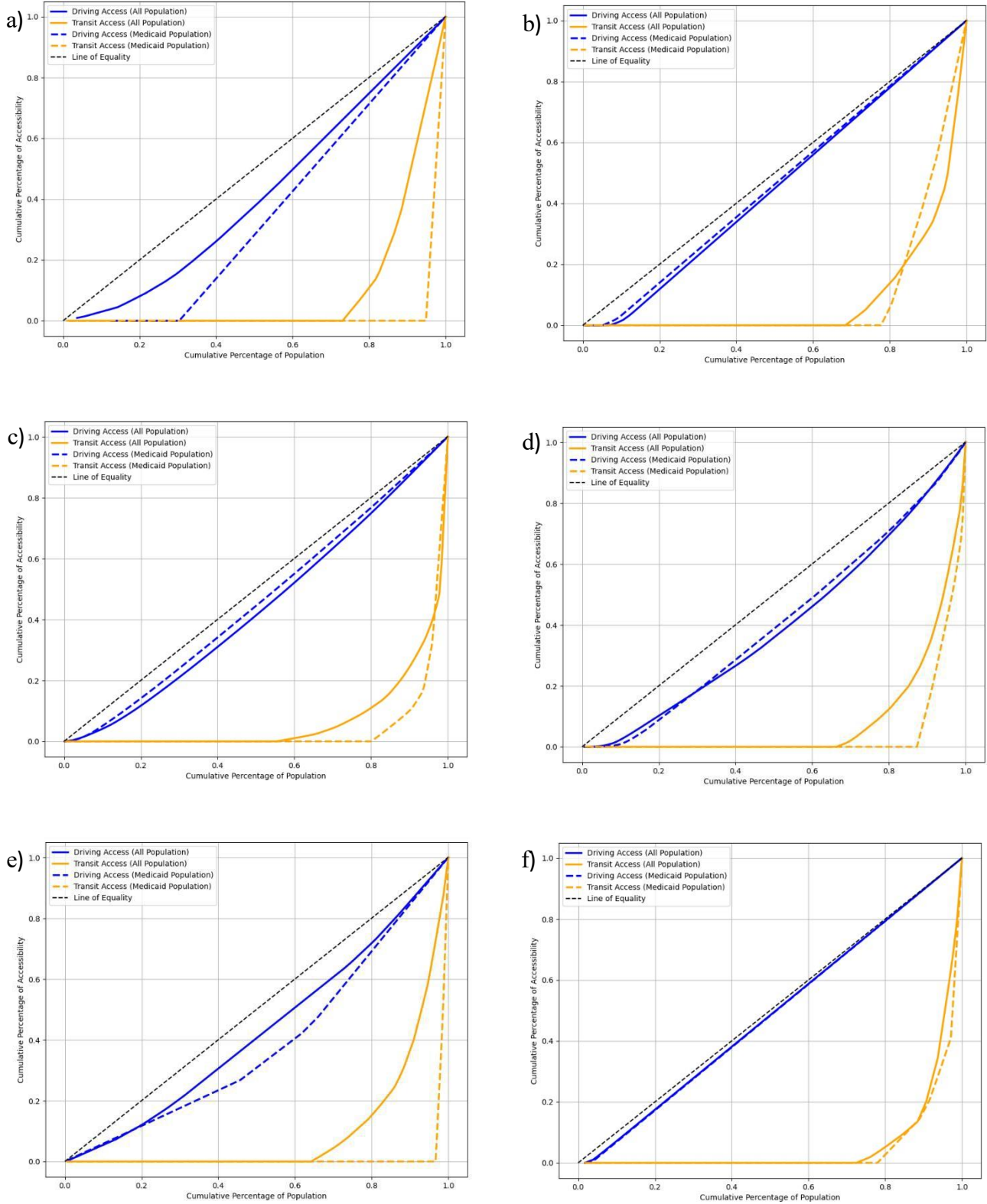


Figure 3. Lorenz curves of driving and transit accessibility scores for all dental clinics and clinics participating in Medicaid/CHIP. *a) Altavista, b) Lynchburg, c) Richmond, d) Staunton-Harrisonburg, e) Williamsburg, f) Winchester*

4. Discussion

Our results revealed that public transit accessibility is lower compared to driving accessibility to dental care services, irrespective of Medicaid participation, which is in line with the findings from previous studies measuring accessibility to hospital and pediatric clinic services in the U.S.^{19,20} The differences between driving and public transit accessibility were larger in regions with smaller geographic extent compared to those with larger geographic extent. This may reflect the undeveloped transportation system in the U.S., especially in smaller regions. Thus, to increase access to dental care, in addition to promoting Medicaid participation among dental clinics, developing public transportation systems, especially in small urban and rural regions are important. Dentists should also consider proximity to public transportation and major roads while selecting clinic locations.

The results of the Gini coefficient analysis showed inequality in access to dental services through public transit is much higher compared with driving. This is expected as the public transportation systems varies across municipalities.^{21,22} This difference was larger for clinics that participate in Medicaid. The high Gini coefficient for transit suggests taking public transportation to dental clinics might not be an option for many people, even those who live in big cities. Lacking access to personal vehicles or the ability to drive in those regions could significantly impact their access to care. Ride support from community level organizations or ride-sharing services, like Uber Health which is designed for healthcare organizations to provide rides for patients and caregivers, might be able to improve access.²³ To increase access to dental care among Medicaid population, Virginia expanded dental services to cover comprehensive care such as preventive services and essential treatments for adults starting from 2021 and increased the dental care reimbursement rate for healthcare providers by 30% in 2022 for treating both children and adults.^{7,8} Given the timing of our analysis, these recent policy changes may not yet be reflected in the current geospatial access to care. However, future research could include longitudinal analyses to examine changes in access following these policy implementations.

While exploring contributing factors, we found that a high poverty rate is associated with better access to dental clinics accepting Medicaid through public transit in Winchester and Richmond. This finding aligns with previous research in dental care and primary care. One explanation is the

high demand for dental care in regions with larger populations living in poverty, given the population served by Medicaid. This might also be related to the locations of low-income communities, which are more likely to be concentrated in urban centers where public transportation is readily available.²⁴ While we observed mixed findings regarding non-White population and population density regarding transit accessibility.

This echoes previous research that social economic factors impact geospatial accessibility differently across various regions.^{16,25,26} Thus, public health policy needs to consider the unique local sociodemographic, geographic, and transportation attributes.

The study has several limitations. First, many rural towns and small cities in Virginia are not included in this analysis, due to the lack of comprehensive transit data. Second, this study only considers driving and public transportation without considering other modes of transportation, such as ride-sharing, for example Uber or Lyft, which 13% of the Americans have been using for daily transportation.²⁷ Additionally, the analysis was conducted in census block group level which lacks of individual level information. Considering those limitations, future studies can consider building a comprehensive public transit network through manually searching public transition routes and. Collecting individual-level data may also help capture the use of alternative transportation modes and provide a more accurate reflection of real-life access to dental care.

5. Conclusion

In Virginia, access to dental care services is more challenging using public transportation compared with driving, especially in smaller regions and among Medicaid beneficiaries. In addition to continuing efforts to encourage dentists to accept Medicaid patients, it is also important to prioritize establishing new dental clinics near public transportation and major roads, utilizing ride-sharing support, and advocating for the development of robust public transportation systems.

References

1. Rahman MS, Blossom JC, Kawachi I, Tipirneni R, Elani HW. Dental Clinic Deserts in the US: Spatial Accessibility Analysis. *JAMA Network Open*. 2024;7(12):e2451625-e2451625. doi:10.1001/jamanetworkopen.2024.51625
2. Serón C, Olivero P, Flores N, et al. Diabetes, periodontitis, and cardiovascular disease: towards equity in diabetes care. Mini Review. *Frontiers in Public Health*. 2023-December-21 2023;11doi:10.3389/fpubh.2023.1270557
3. Shariff JA, Edelstein BL. Medicaid Meets Its Equal Access Requirement For Dental Care, But Oral Health Disparities Remain. *Health Aff (Millwood)*. 2016/12// 2016;35(12):2259-2267. doi:10.1377/hlthaff.2016.0583
4. Rephann T, Nicole T, Terance W, Rephann J, Wanchek T. Oral Health in Virginia: Trends, Disparities, and Policy Implications. *The Virginia News Letter*. 08/01 2012;88:1-20.
5. Association AD. *Dentist participation in Medicaid or CHIP*. 2020. Accessed February 5, 2025. https://www.ada.org/-/media/project/ada-organization/ada/ada-org/files/resources/research/hpi/hpigraphic_0820_1.pdf
6. Williams E. *Variation in use of dental services by children and adults enrolled in Medicaid or CHIP*. 2024. Accessed February 5, 2025. <https://www.kff.org/medicaid/issue-brief/variation-in-use-of-dental-services-by-children-and-adults-enrolled-in-medicaid-or-chip/>
7. Wolfe M, McDonald N, Holmes G. Transportation Barriers to Health Care in the United States: Findings From the National Health Interview Survey, 1997–2017. *American Journal of Public Health*. 04/16 2020;110:e1-e8. doi:10.2105/AJPH.2020.305579
8. Luo W, Wang F. Measures of spatial accessibility to health care in a GIS environment: synthesis and a case study in the Chicago region. *Environment and planning B: planning and design*. 2003;30(6):865-884.
9. Association AD. Methodology for developing the American Dental Association office database. *Health Policy Institute* <https://www.ada.org/-/media/project/ada-organization/ada/ada-org/files/resources/research/hpi/hpiofficedatabasemethods.pdf>. 2017;
10. Kim J, Lee B. More than travel time: New accessibility index capturing the connectivity of transit services. *Journal of Transport Geography*. 2019;78:8-18.

11. Kim J, Rapuri S, Chuluunbaatar E, et al. Developing and evaluating transit-based healthcare accessibility in a low-and middle-income country: A case study in Ulaanbaatar, Mongolia. *Habitat International*. 2023;131:102729.
12. Pereira R, Saraiva M, Herszenhut D, et al. r5r: Rapid Realistic Routing on Multimodal Transport Networks with R5 in R. *Transport Findings*. 03/12 2021;doi:10.32866/001c.21262
13. Google for Developers. GTFS Static Overview Static Transit,.
<https://developers.google.com/transit/gtfs>
14. Bosanac EM, Parkinson RC, Hall DS. Geographic access to hospital care: a 30-minute travel time standard. *Med Care*. 1976/07// 1976;14(7):616-624. doi:10.1097/00005650-197607000-00006
15. 85 Update to Access Standards: Drive Time Calculations (National Archives and Records Administration) 42724-42726
(2020).
16. Kim J, Karki S, Brickhouse T, et al. Navigating Disparities in Dental Health-A Transit-Based Investigation of Access to Dental Care in Virginia. *Community Dentistry And Oral Epidemiology*. 10/01 2024:1-8. doi:10.1111/cdoe.13015
17. Gini C. Measurement of Inequality of Incomes. *The Economic Journal*. 1921;31(121):124-126. doi:10.2307/2223319
18. Dubin RA. Spatial Autocorrelation: A Primer. *Journal of Housing Economics*. 1998/12/01/ 1998;7(4):304-327. doi:<https://doi.org/10.1006/jhec.1998.0236>
19. Alam MS, Tabassum NJ, Tokey AI. Evaluation of accessibility and equity to hospitals by public transport: evidence from six largest cities of Ohio. *BMC Health Services Research*. 2023/06/08 2023;23(1):598. doi:10.1186/s12913-023-09588-0
20. Zhou X, Yuan L, Wu C, Yu Z, Lei W. Measuring spatiotemporal accessibility to healthcare with multimodal transport modes in the dynamic traffic environment. 2023;15(1)Open Geosciences. doi:doi:10.1515/geo-2022-0461
21. Anderson K, Galaskiewicz J. Racial/Ethnic Residential Segregation, Socioeconomic Inequality, and Job Accessibility by Public Transportation Networks in the United States. *Spatial Demography*. 10/01 2021;9doi:10.1007/s40980-021-00093-8

22. Borowski E, Ermagun A, Levinson DM. DISPARITY OF ACCESS: VARIATIONS IN TRANSIT SERVICE BY RACE, ETHNICITY, INCOME, AND AUTO AVAILABILITY. 2018:
23. Uber Health. Taking the homework out of healthcare. <https://www.uberhealth.com/>
24. Glaeser EL, Kahn ME, Rappaport J. Why do the poor live in cities? The role of public transportation. *Journal of Urban Economics*. 2008/01/01/ 2008;63(1):1-24.
doi:<https://doi.org/10.1016/j.jue.2006.12.004>
25. Lawal O, Anyiam F. Modelling Geographic Accessibility to Primary Health Care Facilities: Combining Open Data and Geospatial analysis. *Geo-spatial Information Science*. 07/30 2019:1-11. doi:10.1080/10095020.2019.1645508
26. Henry K, Sherman R, Farber S, Cockburn M, Stroup A. The joint effects of census tract poverty and geographic access on late-stage breast cancer diagnosis in 10 US States. *Health & place*. 02/28 2013;21C:110-121. doi:10.1016/j.healthplace.2013.01.007
27. Sikder S. Who Uses Ride-Hailing Services in the United States? *Transportation Research Record: Journal of the Transportation Research Board*. 06/25 2019;2673:036119811985930.
doi:10.1177/0361198119859302

Supplementary Material

Supplementary Table 1. Measures of all variables included in the analysis

Variable	Measures	Data sources	Year
Dental care provider			2022
Location of clinics	Geographic coordinates of dental clinics	ADA Office Database	
Full time equivalent	Number of providers (FTE) per clinic		
Geospatial measures			
Driving time	Travel time from block group centroid to clinics via road network	Google Maps Driving API	2024
Public transit time	Travel time from block group centroid to clinics via public transit	GTFS data, r5r package	2023
Sociodemographic variables			
Total Population	Population total	American Community Survey	5 year estimation, 2022
Median Age %	Median age of population		
White Alone %	Population of those who are white		
% Higher Education	(Bachelor's Degree or Higher)		
% Unemployment Rate	Population of those who are unemployed		
% Below Poverty Level	Population of those below the poverty level		
% No Vehicle Available	Measured at Household Level		
% Public Transit Commuters	Population of those who rely on Public Transportation		
% Uninsured	Population of those who are uninsured		
% Medicaid/means tested	Population of those who rely on Medicaid		

Supplementary Table 2. The number and type of dental care providers in each of the six study regions based on data from the 2022 ADA Office Database. Providers are categorized by specialty, including general practitioners (GP Dentists), pediatric dentists, oral surgeons, and other dental specialists.

Region	Endodontist	General Practitioner Dentist	Oral Surgeon	Orthodontist	Pediatric dentist	Periodontists	Prosthodontist	Other Specialty	Total
Altavista	0	20	0	0	0	0	0	0	20
Lynchburg	4	70	9	8	2	4	3	2	102
Richmond	104	968	121	140	161	58	24	41	1617
Staunton	5	183	13	16	4	6	2	4	233
Williamsburg	4	139	8	13	43	5	2	0	214
Winchester	8	128	14	16	14	6	0	1	187

Supplementary Table 3. The association between sociodemographic variables and accessibility through driving to dental clinics participating in Medicaid/CHIP applying spatial error model

	Altavista		Lynchburg		Winchester		Williamsburg		Staunton-Harrisonburg		Richmond	
	Coeff (SE)	<i>P-value</i>	Coeff (SE)	<i>P-value</i>	Coeff (SE)	<i>P-value</i>	Coeff (SE)	<i>P-value</i>	Coeff (SE)	<i>P-value</i>	Coeff (SE)	<i>P-value</i>
Population Density	0.0089 (0.0125)	0.4756	0.0015 (0.0044)	0.7381	0.0079 (0.0099)	0.4267	0.0068 (0.0064)	0.2871	-0.0009 (0.0027)	0.7515	0.0007 (0.0004)	0.0497
Non-White Percentage	-0.0005 (0.0015)	0.7366	0.0015 (0.0010)	0.1302	0.0049 (0.0052)	0.3436	-0.0009 (0.0017)	0.6101	0.0016 (0.0013)	0.2163	-0.0003 (0.0002)	0.1891
Poverty Percentage	-0.0001 (0.0024)	0.9653	-0.0003 (0.0017)	0.8822	-0.0003 (0.0074)	0.9705	0.0047 (0.0034)	0.1599	0.0006 (0.0014)	0.6857	0.0005 (0.0003)	0.1328
Intercept	0.0020 (0.0005)	<0.001	0.0080 (0.0008)	<0.001	0.0332 (0.0010)	<0.001	0.0112 (0.0017)	<0.001	0.0075 (0.0011)	<0.001	0.0100 (0.0004)	<0.001
AIC ^a	- 267.3300		- 788.6800		- 606.8400		- 846.9400		- 1705.1000		- 6891.6000	
Log-likelihood ^b	139.6663		400.3409		309.4191		429.4707		858.5521		3451.8010	
Observations	25		81		80		94		184		635	

Coeff refers to coefficient; SE refers to standard error

^a Akaike Information Criterion (AIC) measures model fit and lower AIC indicates a better-fitting model.

^b Log-Likelihood measures how well the model explains the observed data and higher log-likelihood values indicate a better model fit.

Supplementary Table 4. The association between sociodemographic variables and accessibility through taking public transit to dental clinics participating in Medicaid/CHIP applying spatial error model

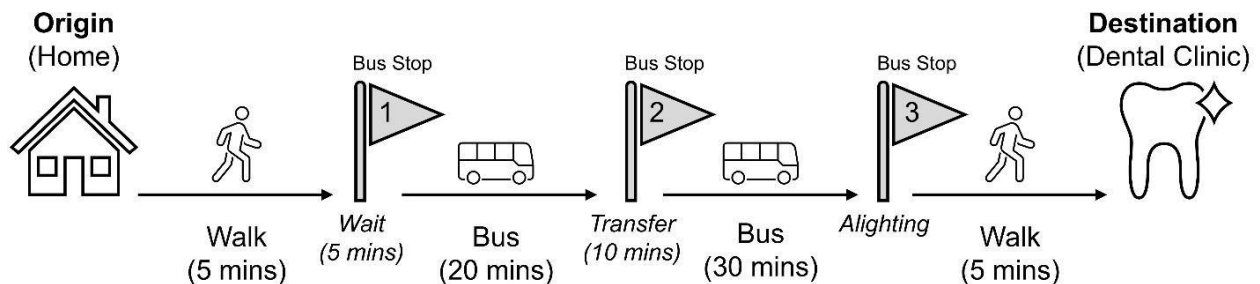
	Altavista		Lynchburg		Winchester		Williamsburg		Staunton -Harrisonburg		Richmond	
	Coeff (SE)	<i>P</i> -value	Coeff (SE)	<i>P</i> -value	Coeff (SE)	<i>P</i> -value	Coeff (SE)	<i>P</i> -value	Coeff (SE)	<i>P</i> -value	Coeff (SE)	<i>P</i> -value
Population Density	0.0011 (0.0033)	0.7499	0.0009 (0.0011)	0.4023	-0.0082 (0.0045)	0.0419	-0.0003 (0.0018)	0.8812	0.0004 (0.0006)	0.5853	0.0003 (0.0001)	0.0005
Non-White Percentage	0.0027 (0.0008)	<0.001	0.0004 (0.0003)	0.1324	0.0016 (0.0022)	0.4784	-0.0003 (0.0007)	0.6380	-0.0001 (0.0003)	0.6496	-0.0001 (0.0001)	0.0383
Poverty Percentage	-0.0017 (0.0012)	0.1669	0.0009 (0.0005)	0.0835	0.0091 (0.0032)	<0.001	-0.0002 (0.0014)	0.8613	0.0003 (0.0004)	0.4539	0.0003 (0.0001)	<0.001
Intercept	-0.0003 (0.0002)	0.0553	-0.0001 (0.0001)	0.2448	0.0003 (0.0006)	0.6594	0.0003 (0.0002)	0.1360	0.0001 (0.0001)	0.0941	0.0001 (0.0001)	0.3817
AIC ^a	-307.1900		-988.1300		-746.8700		-1030.9000		- 2240.700 0		- 8774.700 0	
Log-likelihood ^b	159.5940		505.0668		379.4366		521.4719		1125.341 0		4393.365 0	
Observations	25		81		80		94		184		635	

Coeff refers to coefficient; SE refers to standard error

^a Akaike Information Criterion (AIC) measures model fit and lower AIC indicates a better-fitting model.

^b Log-Likelihood measures how well the model explains the observed data and higher log-likelihood values indicate a better model fit.

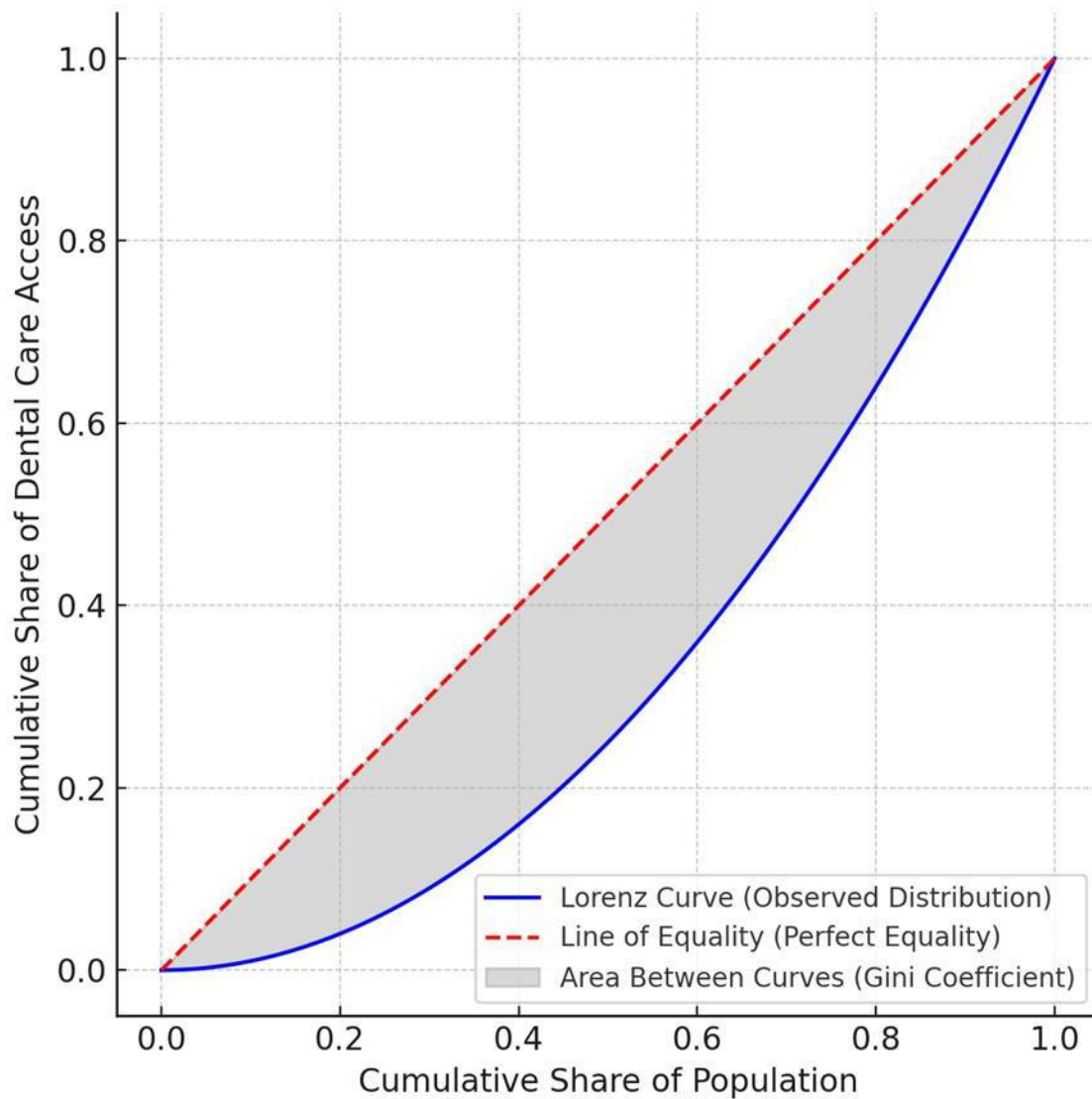
Supplementary Figure 1.



In this study, we utilized the actual time schedules and route geometries of public transit services, buses, available in the study area. Spatiotemporally detailed public transit networks were constructed to calculate segments of transit trips between given origin-destination pairs. For example, the illustrated figure depicts a representative trip from a residence to a dental clinic, where the passenger first walks five minutes to the nearest bus stop, waits five minutes for the

first bus, rides the first bus for 20 minutes, transfers to a second bus with a 10-minute wait, rides the second bus for 30 minutes, alights at the final stop, and walks an additional five minutes to the clinic. These values are presented for illustrative purposes only. For each real-world origin-destination pair, we calculated realistic and accurate public transit travel times using a shortest-path algorithm applied to the actual transit network.”

Supplementary Figure 2. Lorenz curves for dental care accessibility score



III. Manuscript II: Generative AI for Geospatial Analysis: Fine-Tuning ChatGPT to Convert Natural Language into Python-Based Geospatial Computations

Zachary Sherman, Sandesh Sharma Dulal, Jin-Hee Cho, Mengxi Zhang *, Junghwan Kim *

*Co-Corresponding Author

Highlights

This study demonstrates how fine-tuned large language models (LLMs) can translate natural language into executable geospatial Python code with high accuracy and efficiency. The fine-tuned model outperforms a baseline model across six spatial query types, showing substantial gains in correctness and reductions in execution errors and token usage. These findings highlight the potential for scalable AI-powered geospatial analytics in public health, smart cities, and spatial decision-making applications.

What are the main findings?

- Fine-tuning GPT-4o-mini on geospatial queries significantly improves Python code generation for spatial analysis tasks
- The fine-tuned model achieved an 89.7% accuracy rate, improving 49.2 percentage points over the baseline.

What is the implication of the main finding?

- Integrating LLMs into geospatial dashboards enables real-time, user-friendly analysis for smart city management.
- This framework offers scalable potential for domain-specific AI tools in geospatial science and smart urban analytics.

Abstract: This study investigates the potential of fine-tuned large language models (LLMs) to enhance geospatial intelligence by translating natural language queries into executable Python code. Traditional GIS workflows, while effective, often lack usability, and scalability for non-technical users. LLMs offer a new approach by enabling conversational interaction with spatial data. We evaluate OpenAI’s GPT-4o-mini model in two forms: an “As-Is” baseline and a fine-tuned version trained on 600+ prompt-response pairs related to geospatial Python scripting in

Virginia. Using U.S. Census shapefiles and hospital data, we tested both models across six types of spatial queries. The fine-tuned model achieved 89.7%, a 49.2 percentage point improvement over the baseline's 40.5%. It also demonstrated substantial reductions in execution errors and token usage. Key innovations include the integration of spatial reasoning, modular external function calls, and fuzzy geographic input correction. These findings suggest that fine-tuned LLMs can improve the accuracy, efficiency, and usability of geospatial dashboards when they are powered by LLMs. Our results further imply a scalable and replicable approach for future domain-specific AI applications in geospatial science and smart cities studies.

Keywords: Geospatial Data; Dashboard; Fine-tuned; ChatGPT; Large Language Model

1. Introduction

1.1. Background

Recent advances in large language models (LLMs), such as OpenAI's ChatGPT, hold transformative potential for expanding stakeholder participation in geospatial analysis and visualization while enhancing smart urban management processes [1-3]. For instance, LLM-powered chatbots can translate natural language queries into programming code and execute analytical tasks with different file types [4,5], including geospatial data [6-8]. By leveraging conversational interfaces, these LLM-powered chatbots enable non-technical users (e.g., policymakers and citizens) to interact with geospatial data more intuitively, facilitating rigorous problem-solving that ultimately leads to smarter management of cities [9-11]. However, most geographic information system (GIS) tools are not designed for conversational interaction posing a barrier to usability and accessibility.

Recent advances in LLMs can potentially fill this gap by enabling natural language interaction, opening the door to broader participation of smart decision-making based on geospatial data. This paper aims to harness the potential of LLMs to support user-friendly decision-making processes involving complex spatial data and analysis. Specifically, we conducted a methodological investigation to evaluate the capabilities of OpenAI's GPT-4o-mini model in addressing geospatial queries and examine how the refined version (i.e., fine-tuned using our proposed strategies) enhances geospatial reasoning compared to the original model. This paper begins by reviewing state-of-the-art research at the intersection of LLMs and geospatial analysis, highlighting gaps in existing studies. We then outline our research methods and materials,

followed by the presentation of results and discussion.

1.2. State-of-the-art: The intersection of LLMs and geospatial analysis

Since the launch of OpenAI's ChatGPT in 2022, a growing number of studies in the field of geography and GIScience have explored the use of LLMs for geospatial domains [12], such as location information retrieval [13], smart urban infrastructure monitoring [14,15], and geospatial analysis workflows [16-20].

For instance, Jiang and Yang [16] evaluated LLMs for generating structured spatial queries in SQL. Their results demonstrate that while LLMs could automate basic spatial tasks, LLMs struggled with more complex operations such as spatial joins and multi-table queries. Similarly, Zhang et al. [20] introduced MapGPT, a framework for generating thematic maps through natural language inputs that simplified map creation but faced challenges in handling advanced spatial logic and maintaining accuracy. Mansourian and Oucheikh [17] further advanced this area with ChatGeoAI, a platform leveraging LLMs to translate natural language queries into geospatial code using PyQGIS. Ning et al. [18] proposed LLM-Find, an autonomous GIS agent framework that allows LLMs to retrieve geospatial data by selecting sources and generating executable code. While the system showed promising results, it still struggled with lengthy handbook inputs, vague user queries, and reliable retrieval of complex spatial datasets.

Although these advancements mark significant progress in applying LLMs to geospatial data and tasks, a growing number of studies have also reported limitations in the spatial reasoning and processing capabilities of LLMs. For example, Zhang et al. [21] found that while LLMs can generate basic spatial queries, they often fail when handling complex workflows requiring multi-layered computations. Similarly, Tao and Xu [22] highlighted how LLMs frequently generate "hallucinated data", outputs that appear correct but contain fundamental spatial errors, such as misplacing counties from Missouri and Arkansas onto a population map intended for Mississippi. This occurred because the model relied on name matching (e.g., counties named "Mississippi") rather than accurate geospatial identifiers like FIPS codes, revealing a critical limitation in its ability to perform reliable spatial joins. In real-world applications where such LLMs inform decision-making processes, these inaccuracies may lead to misinformed policy decisions and the inefficient allocation of resources. Additionally, many LLMs are trained in text-based corpora, meaning they lack domain-specific geospatial reasoning. A recent study by Renshaw et al. [19] found potential geographic biases in LLMs' spatial reasoning capabilities.

For example, LLMs performed better in high-density urban areas than in low-density rural areas when asked to identify queen-type adjacent counties or K-5 nearest neighbors.

This 'uneven geography' of LLM capabilities has been noted in previous studies. For example, Kim et al. [13] examined ChatGPT's ability to provide locally specific responses related to environmental issues across more than 3,000 U.S. counties. Their findings, consistent with those of Renshaw et al. [19], revealed that ChatGPT struggled to generate local-specific information for low-density rural counties compared to high-density urban ones, highlighting a key limitation of LLMs in the geospatial context. Additionally, Jang and Kim [15] investigated multimodal LLMs and their ability to detect built environment features (e.g., trees, streetlights) from street-view images. They found that a LLM performed better on images from urban areas than rural ones, further confirming the existence of an 'uneven geography' in LLM capabilities.

1.3. Research goal and questions

Therefore, to effectively employ LLMs in geospatial analysis and visualization processes, these limitations must be addressed. Overcoming these limitations requires integrating GIS data structures, providing specialized training in spatial concepts, and developing spatial reasoning modules to improve accuracy in geospatial queries. Given these challenges, there remains a significant knowledge gap regarding how to enhance LLMs' ability to accurately and efficiently process geospatial data queries, with only a few studies addressing this issue [23,24] to our best knowledge. Therefore, to address this critical gap, this study aims to enhance the ability of LLMs to generate executable Python code for geospatial analysis by proposing and evaluating targeted fine-tuning strategies.

To achieve this research goal, we ask the following three research questions. First, how accurately can an existing LLM (an "As-Is" LLM that is not fine-tuned) generate Python code for handling various geospatial data and queries? Second, what types of errors are frequently observed for the "As-Is" LLM? Third, how does a fine-tuned LLM compare to an "As-Is" LLM in accuracy and efficiency in handling geospatial data and queries? To answer these questions, we adopt a case study approach focused on hospital locations in Virginia, United States. Virginia was selected for its diverse geographic characteristics, including a mix of urban, suburban, and rural regions [25,26]. While the case study is conducted within Virginia, the methodology is designed to be scalable and adaptable to broader applications in enhancing LLMs' performance in processing geospatial data queries. We also conducted an additional test using New York State

as a new study area to validate our approach and framework. Furthermore, we selected and compared two LLM models (OpenAI’s GPT-3.5-Turbo and GPT-4o-mini), which are accessible via API.

Our work has the following **key contributions**. First, we propose a novel fine-tuning framework that trains LLMs to generate executable Python code for geospatial analysis using natural language input, addressing a critical usability gap in traditional GIS tools. Second, we implement innovative mechanisms within the fine-tuned model, including modular external function calls, spatial reasoning enhancements, and fuzzy geographic input correction, which have not widely adopted in geospatial workflows. Third, we empirically demonstrate that fine-tuning dramatically improves model performance, achieving 89.7% accuracy (vs. 40.5% in the baseline) and reducing execution errors and token usage by over 70%. These improvements enhance the model’s effectiveness, computational efficiency, and cost-effectiveness, making LLMs more capable in geospatial reasoning and scalable for real-time, high-volume, industry-level applications.

2. Materials and Methods

2.1. Overview of Research Design

The primary objective of this study is to propose fine-tuning strategies to enhance LLMs' ability to process geospatial data queries with greater accuracy and efficiency. To achieve this, we first evaluated the performance of an existing, non-fine-tuned LLM (“As-Is” LLM) in handling geospatial data and queries. We assessed the model’s accuracy by comparing the LLM’s results with those generated by traditional geospatial analysis tools. Next, we analyzed the types of errors in the As-Is LLM to gain insights into developing fine-tuning strategies. Finally, we fine-tuned the As-Is model and compared its performance to assess the effectiveness of our fine-tuning strategies in improving LLM’s capability in handling geospatial data and queries.

We adopted a case study approach [2,23], focusing on a realistic scenario where geospatial queries are centered on a public health topic, such as understanding geospatial accessibility to hospitals. We purposefully selected this domain because geospatial data have played a critical role in public health planning, resource allocation, and addressing health disparities [27]. Public health is also a domain, where place-based decisions that are informed by geospatial data and analysis are crucial, making it a useful testbed for evaluating the utility of LLMs in geospatial analysis. Additionally, healthcare accessibility is a key concern across urban, suburban, and rural

contexts, enabling assessment of the model’s performance under diverse spatial conditions [28].

To conduct this case study, we utilized the following empirical geospatial data. The primary datasets included:

1. Virginia polygon shapefiles representing county and ZIP code boundaries, based on the U.S. Census Bureau’s TIGER/Line database, which served as the spatial units for aggregation and spatial filtering.
2. A dataset of street addresses, randomly sampled across Virginia to simulate realistic inputs for location-based accessibility analyses.
3. A point shapefile of hospital locations, obtained from the Virginia Geographic Information Network (VGIN), which provided accurate and up-to-date healthcare facility coordinates essential for proximity and network-based accessibility calculations.

By incorporating multiple geographic scales (i.e., ZIP code, county, and individual addresses) and grounding our hospital location data into an authoritative state-level source (VGIN), we ensured that the model was trained and evaluated under conditions that reflect plausible user scenarios in real world [16]. Furthermore, the use of standardized and widely available geospatial formats enhances the reproducibility and scalability of our approach for broader adoption across other applications.

2.2. As-Is Model: OpenAI’s GPT-4o-mini

The As-Is model served as a baseline, utilizing an unmodified version of OpenAI’s GPT-4o-mini model and GPT-3.5-Turbo to generate Python code in response to spatial queries (**Figure 1**).

GPT-4o-mini was selected for its optimal balance of performance, speed, and cost, making it a suitable candidate for scalable and cost-effective applications in industry and research.

According to OpenAI, GPT-4o-mini is a “fast, affordable small model for focused tasks,” and is particularly ideal for fine-tuning in domain-specific applications [29,30]. Additionally, GPT-4o-mini supports distillation from larger models such as GPT-4o, achieving similar performance with lower latency and reduced computational costs, valuable for resource-constrained or high-volume deployments [29,30]. GPT-4o-mini consistently outperformed GPT-3.5-turbo across nearly every category, including accuracy, execution success, and token efficiency, making GPT-4o-mini a reliable foundation for fine-tuning in spatial code generation tasks. Although several comparable LLMs exist, GPT-4o-mini provided the most practical balance of

accessibility, cost, and performance for this study's scope.

This As-Is model was not fine-tuned. However, to help the LLM's minimal understanding of our dataset, details (e.g., names of variables) were provided within each prompt. For example, each prompt specified the data context, including key shapefiles, relevant variable names, and their associated attributes. The contextual information included geographic information on counties, ZIP code areas, and clinic facility locations across Virginia. To improve query accuracy, essential columns from each dataset, such as common U.S. Census variable names (e.g., NAMESLAD for county names and ZCTA5CE20 for ZIP codes), were explicitly referenced. Providing this minimal context was necessary to fairly evaluate the As-Is model because, without context about variable names or the structure of the input data, the model would not have been able to generate meaningful or executable geospatial code [31-33]. Since language models inherently lack knowledge of dataset schemas, omitting this context would make it impossible to fairly assess the model's ability to translate spatial questions into valid Python queries.

Moreover, several formatting instructions were added to standardize the model's output to ensure the generated code could be executed directly without modification. These instructions specified that the model should omit wrapping the code as a string, use a consistent variable structure (e.g., referencing data variables directly without reloading the shapefiles), and end each code block with "#End of Code" for easy identification [34]. For queries involving address geocoding and driving time estimations, explicit guidance was provided to use the geopy packages [35,36] for address handling and Mapbox API [8,37] for calculating travel times. This was done to ensure the model had the necessary tools to answer location-based questions accurately. **Appendix A1** illustrates the prompts used for the As-Is model.

Despite structured prompts, formatting instructions, and domain specific context, the As-Is model exhibited several fundamental limitations, including limited spatial awareness, frequent hallucinations, and excessive token usage, all of which compromised its ability to generate accurate geospatial queries. Details of these limitations will be illustrated in the result sections. These shortcomings emphasize the necessity of fine-tuning to enhance the model's spatial awareness, reduce hallucinations, and improve consistency in query execution.

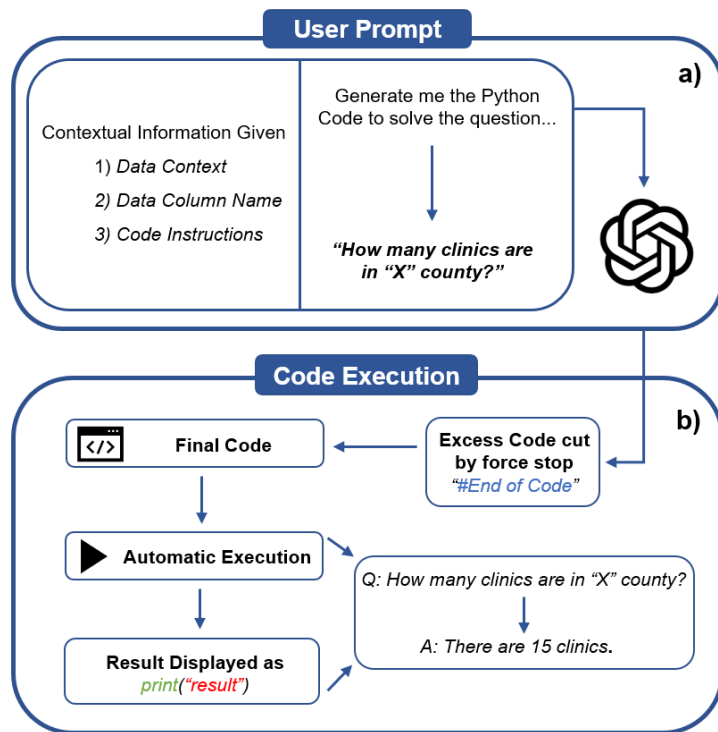


Figure 1. Workflow of the As-Is model. In step (a), the user provides a prompt including data context and instructions. In step (b), the model generates Python code, which is then automatically executed to return the final result. Excess code is removed using a forced stop tag (“#End of Code”).

2.3. Fine-tuned Model Framework

The fine-tuned model addresses key limitations of the As-Is model, specifically improving geospatial reasoning, query accuracy, and computational efficiency. **Figure 2** illustrates an overview of the fine-tuned model’s mechanism.

2.3.1 Fine-Tuning Procedure

While the As-Is model struggled with spatial joins, buffer operations, and driving distance estimations, the fine-tuned model integrates custom training, external function calls, and error handling to improve its performance. A fine-tuned model is a customized adaptation of OpenAI’s LLMs, allowing it to specialize in domain-specific tasks. In this study, a fine-tuned version of OpenAI’s GPT-4o-mini was developed to enhance geospatial analysis capabilities, addressing the As-Is model’s shortcomings.

The model was trained on a dataset of 634 prompt-response pairs, each designed to refine spatial query execution. Training data consisted of customized correct answers and validated outputs from the As-Is model [16]. To improve generalization and reduce overfitting, we employed prompt augmentation, a process in which each spatial query was rewritten in multiple semantically equivalent forms to reflect the diversity of natural language input. For example, the query ‘How many clinics are in X County?’ was rephrased as ‘What is the total number of

clinics in X County?’, ‘Can you tell me the number of clinics operating in X County?’, and other variations. This prompt augmentation approach ensured the model could interpret and process differently phrased questions while avoiding overfitting to specific patterns in the training data [38].

2.3.2 Model Usage and Operation Architecture

Fine-tuning a GPT model involves training it on pairs of prompts and completions, where the prompt is the input (e.g., a question or instruction) and the completion is the ideal response, the model should generate (e.g., correct executable python code). This process teaches the model to produce more accurate or specialized outputs based on the specific patterns in the training data. The completion responses were designed to reflect real-world geospatial workflows by implementing appropriate spatial operations depending on the query type. These workflows are foundational geospatial methods widely used in GIS practice and public health accessibility analysis. Specifically, operations such as spatial joins, buffering for proximity analysis, nearest-neighbor queries, and isochrone-based service area analyses are extensively documented in both GIS textbooks [39,40] and software documentation [41-43] as standard practice. Similarly, studies in public health and urban planning frequently use these techniques to assess accessibility, service coverage, and spatial disparities [19,27,44]. This alignment ensures that the fine-tuning dataset mirrors the operational workflows used by GIS professionals in real-world scenarios. For example, queries asking for the number of clinics within a given distance of a ZIP code or county centroid used buffering and spatial joins, implemented using the GeoPandas library.

For queries that are required to identify the closest clinic to a specific address, the model used the ArcGIS Geocoding API to convert the textual address into geographic coordinates and then calculates the nearest point using spatial distance functions. For queries involving driving-time accessibility (e.g., “clinics within a 15-minute drive”), the Mapbox Isochrone API was used to generate drivetime polygons, which were then intersected with clinic locations using GeoPandas. Across all query types, the spatial data manipulations, such as reading shapefiles, creating buffers, calculating distances, or executing spatial joins, were consistently implemented using the Python GeoPandas library to maintain reproducibility and efficiency. The core geospatial operations represented in the generated Python code are mathematically formalized in **Appendix**, which outlines the spatial logic underlying the model’s code generation outputs. By

including these realistic geospatial operations in the completion responses, the fine-tuned model learned how to accurately convert natural language queries into executable code for location-based spatial analysis. **Appendix** illustrates selective examples of prompt-response pairs that were used for fine-tuning.

Several key improvements were implemented in the fine-tuned model to overcome the As-Is model's shortcomings, including conversational adaptability, location identification, and optimized computational efficiency.

First, the fine-tuned model integrates conversational adaptability (**Figure 2(b)**), distinguishing between general inquiries and computational queries. Unlike the As-Is model, the fine-tuned model assesses whether a user query requires executable code or a conversational response. If a query asks for general information rather than a geospatial computation, the model formats responses as text within a Python print statement and then executes that result automatically. However, when a query demands a spatial computation, the model automatically generates and structures Python code for execution. This structured approach enables handling both general (non-spatial) queries and precise geospatial analytics within the same framework.

Second, the fine-tuned model includes a location identification mechanism that corrects misspelled or improperly formatted geographic names (**Figure 2(c)**). This mechanism is particularly important because, despite improvements in language model performance, LLMs are still prone to failure when user inputs do not precisely match the names found in the geospatial dataset [41]. For example, even a small typo in a county name (e.g., "Fairfak" instead of "Fairfax") can lead to the generation of code that references nonexistent data, resulting in execution errors or failed analyses. To address this, the model uses the Fuzzywuzzy package to compare user inputs against a predefined geographic dictionary containing valid names of counties and ZIP code areas [45]. Fuzzywuzzy applies the Levenshtein distance to compute the similarity between the input string and the valid geographic names [42]. If a confidence score of 75% is met, the model automatically corrects the input to the nearest valid location. If no match is found, the user is prompted to refine their input, preventing the generation of invalid code [43]. As shown in **Figure 2(c)**, user prompts containing location-based queries are processed through the spelling correction system before advancing to geospatial analysis, reducing the likelihood of errors due to minor spelling variations.

Third, a significant optimization in the fine-tuned model is the use of external function calls,

which improves computational efficiency by reducing token usage (**Figure 2(c)**). Unlike the As-Is model, which generated full geospatial scripts within each response, the fine-tuned model calls predefined functions for key operations such as geocoding, isochrone generation, and spatial joins. For instance, when a query requires an address-to-coordinate conversion, the model invokes an ArcGIS geocoder function to retrieve latitude and longitude coordinates, ensuring that address-based queries are accurately interpreted. Another example includes cases for driving distance calculations. In this case, the model integrates the Mapbox API, generating isochrones that represent areas accessible within a specified travel time. These function calls significantly reduce token consumption, lowering computational costs and improving scalability for real-world applications [46]. It is worth mentioning that while the fine-tuned model references external functions for operations such as geocoding, spatial joins, and travel-time estimation, it does not use OpenAI’s built-in function calling capabilities. Instead, the model is trained to generate Python code that invokes predefined custom functions, which are implemented externally and accessible in the execution environment.

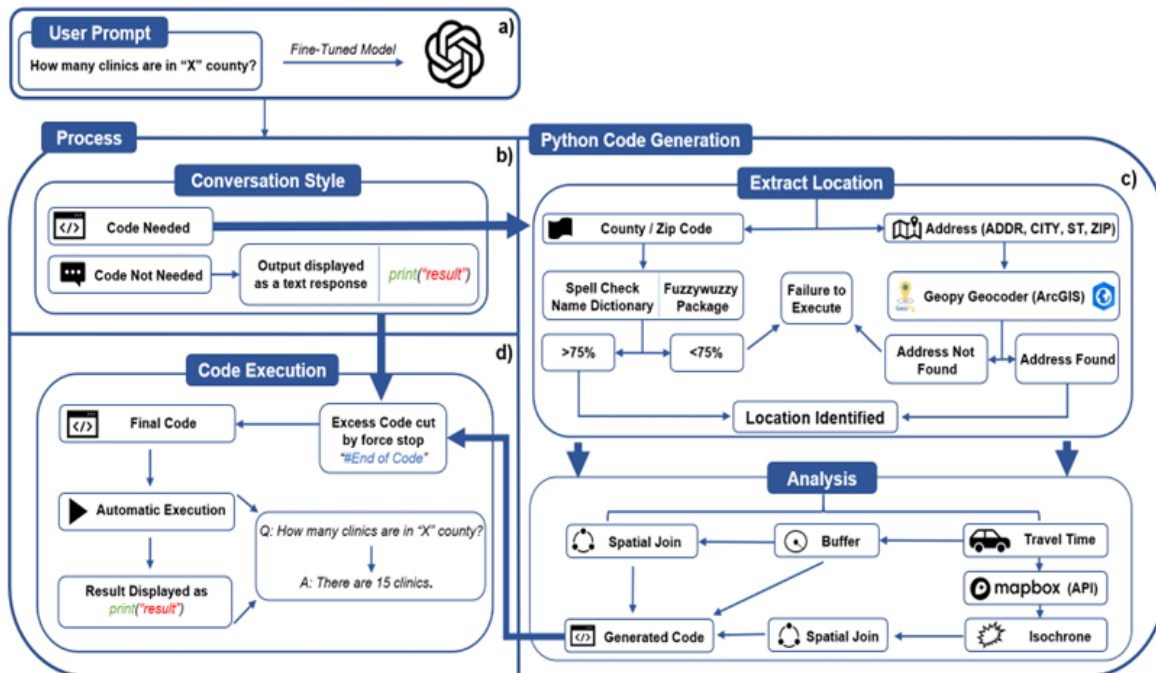


Figure 2. Workflow of the Fine-Tuned Model. (a) After receiving a user prompt, (b) the model determines whether a computational response or general information is required, formatting outputs accordingly. (c) Location inputs undergo correction and geocoding via fuzzy matching and ArcGIS integration. Relevant spatial analyses, including spatial joins, buffers, and travel-time calculations, are executed through pre-integrated Python libraries and APIs, reducing

computational overhead. (d) Excess code is removed using a forced stop tag (“#End of Code”), with final results displayed automatically. The model generates Python code, which is automatically parsed and executed within a Python environment that includes pre-integrated spatial libraries and API keys. No manual copy-pasting is needed, enabling seamless interaction between the LLM and geospatial analysis tools.

2.4. Evaluation Setup for the Performance Comparison of As-Is and Fine-tuned Model

To assess the effectiveness of the fine-tuned model, a comparative evaluation was conducted between the As-Is model and the fine-tuned model, which was specifically trained to process geospatial queries. Both models were tested on six predefined geospatial queries designed to convert natural language into accurate and executable Python. These questions include:

Q1. How many clinics are in “X” County?

Q2. How many clinics are in the area with the ZIP Code “X”?

Q3. What is the closest clinic to “X Address”?

Q4. How many clinics are within 10 miles of the Centroid of “X” County?

Q5. How many clinics are within 10 miles of the Centroid of the area with the ZIP Code “X”?

Q6. How many clinics are within a 15-minute driving time of “X Address”?

These queries were selected to reflect typical geospatial analysis tasks relevant to practitioners, such as assessing healthcare facility locations, calculating proximity-based accessibility, and determining travel times to medical providers [27,28,44]. Specifically, by including both static location-based queries (e.g., total clinics in a county) and dynamic proximity-based queries (e.g., clinics within a 15-minute drive), our evaluation approach ensures that LLMs are tested on a practical subset of core geospatial tasks.

Each query was executed 100 times using randomly selected county names, addresses, or ZIP codes from our case study region (Virginia), ensuring that the model’s performance was evaluated across multiple spatial contexts. A total of 1,200 evaluation queries were issued across both As-Is and fine-tuned models. The evaluation was structured to measure the models’ accuracy and consistency across different spatial scales (e.g., street address vs. zip code vs. county). Accuracy was defined as the model’s ability to generate correct and executable Python code, while consistency was assessed based on whether similar queries produced reliable

responses across multiple iterations. Ground truth answers were established through manual geospatial analysis using Python libraries such as GeoPandas. These analyses included spatial joins, buffering, and Mapbox API-based driving-time calculations.

Any generated code that failed to execute or produced incorrect results was classified as incorrect, while code that executed successfully and returned correct outputs was considered accurate. For incorrect answers, the code was evaluated and categorized by error type.

Additionally, token count was analyzed for all generated code to assess the computational efficiency of the model when handling spatial queries. In the context of LLMs, tokens are the basic units of text that the model reads and generates. A token can be a word, part of a word, or punctuation, depending on the language and tokenization rules used. Token usage is an important measure because it directly affects computational cost, API pricing, and the model's ability to process longer or more complex queries [29,30]. For example, the sentence 'How many clinics are in Fairfax County?' could be broken into tokens such as ['How', 'many', 'clinics', 'are', 'in', 'Fairfax', 'County', '?']. Token counts directly influence both computational costs and processing limits within LLM systems.

2.5 External Validation for Geographic Generalizability

To evaluate the geographic generalizability of the models, we conducted external validation using data from New York State. Two queries were selected to represent different spatial complexities: (1) “How many clinics are in X County?”, administrative spatial join, and (2) “What clinics are within a 15-minute driving time of X Address?”, a higher-complexity proximity analysis requiring address-level geocoding and isochrone computation. This subset was intentionally chosen to reflect both low and high spatial complexity. The New York dataset was entirely separate from the Virginia data used for training, providing a test of the model's ability to generalize spatial reasoning to unseen geographies. New York was selected for external validation in part because it has been used in previous studies as a representative test case for evaluating spatial model generalizability, and also because it presents a diverse mix of dense urban, suburban, and rural geographies, making it a robust test for assessing spatial reasoning across varying spatial contexts [16].

3. Results

This section presents a comparative evaluation of the As-Is and fine-tuned LLMs across six spatial query types. We analyze model accuracy, execution success, token efficiency, and coding errors to assess the practical benefits of fine-tuning for geospatial tasks.

3.1. Performance Comparison between As-Is and Fine-tuned Models.

Table 1 presents the total counts of correct and incorrect answers for the As-Is and fine-tuned models across all spatial queries. The fine-tuned model demonstrated substantial improvements in accuracy, reducing errors across all query types. The most notable improvement was observed for a question (Q6) that asked the LLM to calculate the number of clinics within a 15-minute driving time of a given street address. The As-Is model only achieved a 12% accuracy rate for this query, whereas the fine-tuned model reached 75% accuracy, an improvement factor of 6.25 times. Similarly, for less spatially complex tasks (Q2) such as counting clinics within a specific zip code, the fine-tuned model still exhibited a 1.24 times improvement, increasing accuracy from 76% to 94%.

Additionally, **Table 1** reports the execution failure rate, representing the percentage of incorrect responses that resulted in non-executable code due to syntax, logic, or missing resource errors. Generated code that executed but produced incorrect results is not included in the execution failure rate column. The As-Is model failed to execute queries in up to 100% in incorrect cases, whereas the fine-tuned model significantly reduced execution failures for Q3, Q4, and Q5. Notably, Q5, which required spatial joins, had a 90% execution failure rate in the As-Is model, while the fine-tuned model resolved this issue entirely, achieving 0% execution errors. Overall, the fine-tuned model achieved an accuracy rate of 89.7%, whereas the As-Is model only achieved 40.5% accuracy, proving the necessity of domain-specific training for geospatial applications.

The external validation conducted with New York data, shown in **Table 2**, confirmed the potential generalizability of the fine-tuned model beyond the Virginia training region. For the simple spatial join query (“How many clinics are in X County?”), the fine-tuned model achieved 92% accuracy, significantly outperforming GPT-4o-mini As-Is (71%) and GPT-3.5-turbo As-Is (18%). Non-executable code counts dropped from 64 (3.5-turbo) and 25 (4o-mini) to just 5 (Fine-Tuned). Token usage was reduced by 77% compared to GPT-4o-mini, demonstrating both accuracy and computational efficiency gains. For the more complex isochrone-based query

(“How many clinics are within a 15-minute drive of X Address?”), the fine-tuned model achieved 73% accuracy, a major improvement over 10% (4o-mini) and 9% (3.5-turbo). Non-executable code counts fell from 85 (3.5-tubro) and 80 (4o-mini) to 23 (Fine-Tuned), and token usage decreased by 65%.

Additionally, **Figure 3** illustrates the relationship between token count and spatial query complexity, where the number of computational steps increases token usage. In this study, computational steps are defined as distinct spatial operations, such as geolocation, spatial joins, buffering, distance calculations, etc., required to complete a given geospatial query. The fine-tuned model required an average of 217 tokens per query, compared to 822 tokens per query for the As-Is model. This represents a considerable reduction (74%) in token usage, significantly decreasing the cost of computation in large-scale applications. Specifically, the As-Is model required longer prompts with explicit shapefile references, variable names, and geospatial processing instructions to ensure correct responses, leading to excessive token consumption. In contrast, the fine-tuned model leveraged external function calls, minimizing redundant code generation while maintaining accuracy.

Overall, these findings suggest that fine-tuning significantly enhances the model’s ability to interpret and execute spatial queries. Furthermore, fine-tuning improved not only correctness but also computational efficiency, as indicated by a reduced token usage across all query types.

Table 1. Virginia Comparative Analysis of LLM Query Execution: As-Is vs Fine-Tuned Model

	Correct (%)	Incorrect (%)	Execution Failure Rate (%)	Non-Executable Count	Average Tokens
Q1. How many clinics are in “X” County?					
As-Is (3.5 Turbo)	22 %	78 %	88 %	69	775
As-Is (4o-mini)	67 %	33 %	91 %	30	746
Fine-tuned	94 %	6 %	100 %	6	169
Improvement	+27 % points improved		80 % reduction in code errors		77 % reduced
Q2. How many clinics are in the area with the ZIP code “X”?					
As-Is (3.5 Turbo)	23 %	77 %	96 %	74	764
As-Is (4o-mini)	76 %	24 %	100 %	24	747
Fine-tuned	94 %	6 %	100 %	6	156
Improvement	+18 % points improved		75 % reduction in code errors		79 % reduced
Q3. What is the closest clinic to “X Address”?					
As-Is (3.5 Turbo)	5 %	95 %	94 %	89	843
As-Is (4o-mini)	20 %	80 %	94 %	75	909
Fine-tuned	78 %	22 %	59 %	13	297
Improvement	+58 % points improved		83 % reduction in code errors		67 % reduced
Q4. How many clinics are within 10 miles of the centroid of the “X” County?					

As-Is (3.5 Turbo)	28%	72%	90 %	65	861
As-Is (4o-mini)	39 %	61 %	87 %	53	807
Fine-tuned	98 %	2 %	0 %	0	160
Improvement	+59 % points improved		100 % reduction in code errors		80 % reduced
Q5. How many clinics are within 10 miles of the centroid of the area with the ZIP code "X"?					
As-Is (3.5 Turbo)	22%	78%	88 %	69	823
As-Is (4o-mini)	29 %	71 %	39 %	28	797
Fine-tuned	99 %	1 %	100 %	1	235
Improvement	+70 % points improved		96 % reduction in code errors		71 % reduced
Q6. How many clinics are within a 15-minute driving time of "X Address"?					
As-Is (3.5 Turbo)	6%	94%	96 %	90	887
As-Is (4o-mini)	12 %	88 %	99 %	87	924
Fine-tuned	75 %	25 %	84 %	21	283
Improvement	+63 % points improved		76 % reduction in code errors		69 % reduced
Summary					
As-Is (3.5 Turbo)	17.7 %	82.3 %	92.3 %	456	826
As-Is (4o-mini)	40.5 %	59.5 %	83.2 %	297	822
Fine-tuned	89.7 %	10.3 %	75.8 %	47	217
Improvement	+49.2 % points improved		84.2 % reduction in code errors		74 % reduced

Note. Improvement column represents improvement between As-Is (4o-mini) and the Fine-Tuned model. "Non-Executable Count" represents the number of incorrect responses where the generated code failed to run. "Execution Failure Rate" is calculated as the percentage of non-executable responses relative to the total number of incorrect responses. Execution Failure Rate (%) is calculated relative to incorrect responses. High values (e.g., 100%) occur when very few errors remain, meaning all residual errors were non-executable. For each question, the variable "X" (i.e., county, ZIP code, or address) was randomly selected from the Virginia datasets. Specifically, 100 counties, 100 ZIP codes, and 100 street addresses were randomly sampled at the start of the evaluation and used consistently across both model assessments.

Table 2. New York Comparative Analysis of LLM Query Execution: As-Is vs Fine-Tuned Model

	Correct (%)	Incorrect (%)	Execution Failure Rate (%)	Non-Executable Count	Average Tokens
Q1. How many clinics are in "X" County?					
As-Is (3.5 Turbo)	18 %	82 %	78 %	64	780
As-Is (4o-mini)	71 %	29 %	86 %	25	756
Fine-tuned	92 %	8 %	63 %	5	171
Improvement	+21 % points improved		80 % reduction in code errors		77 % reduced
Q6. How many clinics are within a 15-minute driving time of "X Address"?					
As-Is (3.5 Turbo)	9%	91%	93 %	85	892
As-Is (4o-mini)	10%	90 %	89 %	80	843
Fine-tuned	73%	27 %	85 %	23	298

Improvement	+63 % points improved	71 % reduction in code errors	65 % reduced
-------------	-----------------------	-------------------------------	--------------

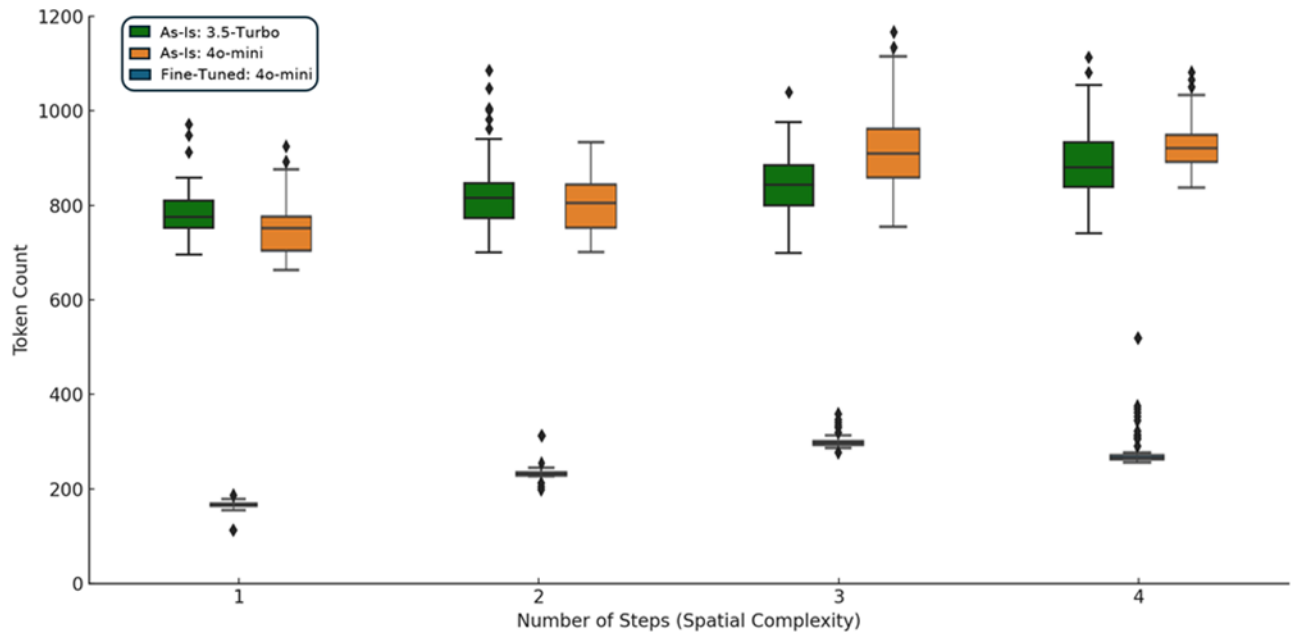


Figure 3. Comparison of token counts across increasing spatial query complexity (Steps 1–4). At each step, the Fine-Tuned model demonstrates considerably lower token usage and reduced variability, indicating greater efficiency and consistency in handling geospatial queries compared to the baseline As-Is model.

3.2. Analysis of Error Types of As-Is and Fine-tuned Models

The analysis of coding errors reveals that the fine-tuned model substantially reduced errors, such as syntax errors, incorrect variable references, and coordinate reference system (CRS) issues, across all categories compared to the baseline As-Is model (**Figure 4**). The most frequent error in the As-Is model was syntax errors, which resulted in immediate execution failures due to missing parentheses, incorrect indentation, or improperly formatted function calls. These errors illustrate the baseline LLM’s lack of structured syntax control, which frequently led to incomplete or malformed code snippets. Fine-tuning remarkably reduced syntax-related failures by refining the model’s understanding of Python’s syntax rules, ensuring that generated code adhered to proper structure and formatting. Errors related to file paths and missing resources also posed a frequent challenge in the baseline model, often arising when hallucinated file locations were referenced. The fine-tuned model eradicated these errors entirely.

Moreover, another prevalent issue in the As-Is model was undefined variables and function errors. These occurred when the model referenced nonexistent variables or attempted to call functions that were either incorrectly named or entirely hallucinated. This issue often led to execution failures, particularly in spatial queries requiring precise function calls. The fine-tuned model substantially mitigated this problem by learning to reference valid function names and predefined methods within the geospatial processing framework.

Errors involving coordinate reference systems (CRS) and geometric transformations were particularly disruptive in the As-Is model. These errors arose when the model failed to correctly assign or transform coordinate reference systems, producing failures such as "invalid projection error" and "cannot transform naive geometries". Such issues are critical in geospatial computations where accurate spatial alignment is essential. The fine-tuned model showed marked improvement in handling CRS transformations by correctly applying projection methods and ensuring appropriate geospatial references in generated queries.

It is important to note that execution accuracy, whether the generated code runs without error, is not equivalent to analytical validity. A code block may execute successfully but still produce incorrect spatial results due to semantic misunderstandings, incorrect spatial joins, or failures in geographic name resolution. While this study primarily measures execution accuracy improvements, we also report on analytical errors where outputs deviated from the correct spatial intent despite successful code execution.

To sum up, fine-tuning led to a substantial decrease in coding errors, improving the model's ability to execute geospatial queries correctly. These improvements were particularly pronounced in syntax handling, spatial data operations, and function referencing, reinforcing the effectiveness of fine-tuning for geospatial healthcare applications.

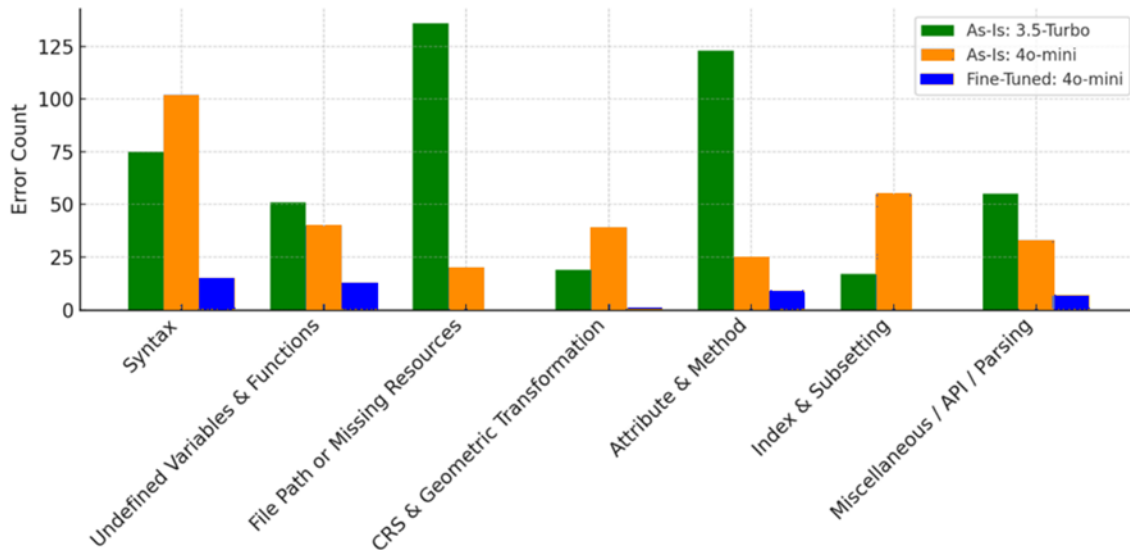


Figure 4. Distribution of coding errors across categories for As-Is and Fine-Tuned models. The fine-tuned model exhibits substantially fewer errors across all categories.

4. Discussions

Our results demonstrate substantial improvements achieved through fine-tuning a generative AI model for geospatial data queries. The fine-tuned GPT-4o-mini model exhibited higher accuracy and greater efficiency in spatial reasoning compared to the baseline As-Is model. By addressing key spatial tasks such as buffering, spatial joins, and travel time estimations, the fine-tuned model was able to produce ready-to-execute Python scripts while reducing computational costs (i.e., the number of tokens). Additionally, the fine-tuned GPT-4o-mini consistently outperformed unmodified GPT-3.5 across all tasks. The New York results further validated our approach, demonstrating similarly high accuracy and efficiency in a different geographic context. These improvements are consistent with previous research demonstrating that fine-tuning enhances LLMs' ability to handle domain-specific tasks [2,16].

A key takeaway from this study is the fine-tuned model's enhanced ability to process multi-step spatial queries, a capability that the As-Is model struggled with. For example, in response to the query "How many clinics are within a 15-minute driving time of X Address?", the fine-tuned model improved accuracy by about 6.3 times, highlighting limitations in geographic and spatial reasoning capabilities of existing LLMs [19]. This aligns with prior research that has demonstrated fine-tuned LLMs significantly outperform general models in structured query

generation [47-49]. In addition to increased accuracy, the fine-tuned model achieved substantial token efficiency, reducing average token usage by approximately 74% compared to the As-Is model. Token reduction was observed even for complex spatial queries, reinforcing that fine-tuning not only improves correctness but also enhances computational cost-efficiency.

The study also identified a marked reduction in error rates, particularly in syntax errors, attribute misapplications, CRS-related issues, and indexing errors. While the As-Is model frequently generated errors related to incorrect function calls, mis-referenced attributes, and missing geospatial transformations, the fine-tuned model effectively reduced these failures by refining its ability to reference geospatial attributes correctly and perform transformations accurately. These findings align with existing literature suggesting that LLM fine-tuning can mitigate hallucinations and improve structured reasoning in computational workflows [4]. However, despite these improvements, some persistent errors remained, particularly in handling advanced spatial operations such as chained geospatial computations and multi-step logic processing. This suggests that further refinement in fine-tuning strategies and the incorporation of reinforcement learning techniques could further improve the model's geospatial reasoning capabilities.

Overall, the results suggest our fine-tuning strategies were effective. The developed model demanded a domain-specific strategy grounded in spatial logic, data structure awareness, and computational efficiency. To that end, we developed a novel fine-tuning framework tailored specifically for spatial querying tasks, which we believe constitutes a foundational contribution to GIScience literature. Our approach included five key innovations: (1) a carefully curated dataset of natural language prompts paired with optimized Python code for solving real-world spatial queries; (2) purposeful variation in question phrasing to promote generalizability across query types and user styles; (3) integration of external function calls, which dramatically reduced token usage while maintaining code modularity and accuracy; (4) a fuzzy matching mechanism to auto-correct geographic names and reduce input errors; and (5) structured formatting protocols that ensured all generated code was executable, syntactically valid, and terminated consistently.

Importantly, the model is not trained to directly answer to given geospatial problems, it is trained to generate executable Python code that solves them. This approach is not a limitation; it is a strength. By generating clean, domain-specific code rather than producing final outputs directly, the model improves transparency, ensures reproducibility, and allows users to interact with geospatial data through natural language. This approach dramatically increases accuracy while

minimizing computational overhead, making it ideally suited for high-volume, real-time, or industry-scale applications where cost-efficiency and interpretability are paramount.

Specifically, our research moves beyond the simple general claim that “fine-tuning improves performance” by offering a replicable, purpose-built methodology for making LLMs spatially aware and providing a scalable blueprint for future applications. Thus, other studies that aim to develop LLMs that are capable of spatial querying can refer to our fine-tuning strategies that can lead to more accurate and efficient LLMs with spatial reasoning than as-is LLMs. For the application domains requiring efficient spatial analytics for decision-making, the integration of conversational AI powered by LLMs could lower technical barriers for non-expert users while streamlining computational processes for geospatial professionals.

This study has several limitations that warrant further investigation. First, the model was fine-tuned primarily on data from Virginia, which may limit generalizability to regions with different geographic structures and naming conventions. While an external validation in New York suggests reasonable transferability, expanding the training data to include more diverse geographic areas would likely improve robustness, particularly in handling ambiguous names and boundary hierarchies.

The dataset consisted of 634 prompt-response pairs (over 70,000 tokens), covering six foundational spatial query types. Although effective, expanding both geographic diversity and query complexity, including multi-constraint queries, chained operations, and raster-based analyses, would improve generalization and semantic accuracy for broader applications. Similarly, while the current benchmark addresses operational coding tasks, it does not capture the full complexity of GIS workflows like spatiotemporal analysis, multi-layer joins, or conditional logic. Consistent with our results, Mooney et al. also found that while GPT-4 significantly outperformed GPT-3.5 in overall accuracy on GIS tasks, both models struggled with advanced GIS concepts such as spatial reasoning, mathematical computations, and complex workflows beyond basic mapping and definitions [9]. While complementary, future research should consider designing hybrid benchmarks that evaluate both conceptual GIS knowledge and operational code generation to provide a more holistic assessment of LLM spatial intelligence. Future work could build on our framework by incorporating such hybrid benchmarks to assess both conceptual reasoning and practical geospatial execution in tandem.

This study focused on OpenAI’s GPT-4o-mini and compared it to GPT-3.5-Turbo. However,

performance on other models and open-source models like Mistral or DeepSeek, remains unknown [50,51]. Broader benchmarking across models, including alternative approaches like retrieval-augmented generation (RAG), prompt chaining, or tool-augmented systems (e.g., GeoAgent, ShapefileGPT), is a crucial next step to evaluate scalability, flexibility, and efficiency trade-offs [2,52]. While our focus was on evaluating fine-tuning in isolation, future research should extend this comparison by integrating empirical tests against these alternative architectures within a unified evaluation framework.

The current framework cannot handle conversational memory, restricting it to single-turn queries without context retention for iterative or chained spatial workflows. This limitation impacts integration into dashboards and decision-support tools. Additionally, reliance on commercial APIs like Mapbox and ArcGIS introduces operational costs; transitioning to open-source alternatives like OpenRouteService (ORS) and Nominatim would improve cost-efficiency and reproducibility [53,54].

Finally, the study did not formally assess output variability, whether semantically equivalent queries yield consistent outputs. Although prompt augmentation aimed to reduce this, variability remains an open question. Likewise, the validation framework focuses on execution correctness but does not detect semantic errors under ambiguity or flag anomalous outputs. Future research should develop trust, uncertainty quantification, and validation mechanisms to support safe deployment in real-world geospatial applications.

Despite the limitations, our study provides important implications for the potential of integrating geospatial data analysis and visualization with LLMs, especially to enhance usability of online geospatial data interactive dashboards. Recently, there has been growing interest in the use of online dashboards as accessible tools for communicating complex spatial information, particularly in public health, urban planning, and environmental monitoring [55-57]. However, while dashboards such as those developed during the COVID-19 pandemic [58] have proven effective at visualizing data and informing policy decisions, they remain limited in interactivity. Most dashboards require users to manually navigate filters, dropdowns, and map layers, posing challenges for non-technical users [44,59].

Our study shows that fine-tuned large language models (LLMs), when integrated with geospatial dashboards, can offer a transformative solution by enabling users to interact through natural language, asking questions like “Which counties have the fewest clinics within a 15-minute

drive?”. LLMs can act as conversational interfaces that simplify data exploration. Unlike off-the-shelf LLMs, which frequently struggle with geospatial logic and generate incorrect or non-executable code, the fine-tuned model consistently returns valid Python code tailored to spatial analysis. Importantly, these responses can be programmed not only to return textual summaries (e.g., number of clinics, access gaps, travel times), but also to dynamically visualize the results directly on the dashboard map. This dual output, visual and textual, has the potential to enhance both usability and interpretability, helping users understand where spatial patterns occur and why they matter. By embedding our fine-tuned LLM framework into dashboards, the platform evolves from a passive visualization tool into an intelligent, user-driven decision support system, unlocking new possibilities for real-time and accessible geospatial analysis.

5. Conclusions

This study investigated how generative AI can be integrated with geospatial analysis to enhance spatial querying via a fine-tuned GPT-4o-mini model. The results highlight the potential of fine-tuning to improve accuracy, reduce computational costs, and overcome technical barriers, enabling broader access to geospatial analytical tools for various stakeholders. Key findings demonstrate the fine-tuned model’s superiority over the baseline “As-Is” model, achieving higher accuracy rates across diverse spatial queries and demonstrating significant token efficiency improvements. Our study identified several effective strategies for improving LLM accuracy and efficiency in spatial tasks. These include (1) training on a curated dataset of geospatial prompts and Python completions, (2) rephrasing each question to increase generalizability and reduce overfitting, (3) enforcing structured outputs using forced stop tags to eliminate redundant generated code, (4) integrating fuzzy location matching to improve spatial input handling, and (5) using modular function calls to reduce token usage and improve code reliability. These strategies not only improved performance but also allowed non-technical users to interact with complex geospatial data in ways that were previously inaccessible, enabling natural language queries to generate accurate spatial outputs dynamically. To realize the full potential of generative AI in spatial analysis, future efforts should prioritize the development of more robust models capable of handling complex geospatial tasks and adapting to diverse geographic contexts that can contribute to a smart decision-making process.

Author Contributions: Conceptualization, Z.S., M.Z., and J.K.; Methodology, Z.S., M.Z., and J.K.; Formal Analysis, Z.S., M.Z., and J.K.; Investigation, Z.S., M.Z., and J.K.; Writing – Original Draft Preparation, Z.S., S.S.D., J.C., M.Z., and J.K.; Writing – Review & Editing, Z.S., S.S.D., J.C., M.Z., and J.K. Z.S., S.S.D., J.C., M.Z., and J.K.; Supervision, M.Z. and J.K.; Funding Acquisition, M.Z. and J.K.

Funding: This research was funded by the Institute for Society, Culture and Environment (ISCE) at Virginia Tech and OpenAI’s Research Access Program.

Acknowledgements: The authors thank the five anonymous reviewers for their thoughtful comments, which helped improve the article considerably.

Data Availability Statement: The original data presented in this study are available from publicly accessible sources, including the U.S. Census Bureau and VGIN.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

A1. Prompt Format for As-Is Model Evaluation

Below is the standardized prompt format used to evaluate the As-Is model. Each question was inserted into the {question} placeholder. Contextual information about the shapefile variables was provided to ensure the model had adequate knowledge of the geospatial data structure.

"""

Generate me the Python code to solve the question: {question}

Use the three shapefiles stored in the variables "counties", "zip_code", and "healthcare".

The variable "counties", is a polygon shapefile containing information about counties in the U.S.

Important Column names include...

[...]

Follow these additional instructions:

1) Do not include "python or wrap code as string. This code should be able to be directly executed.

2) When the code is complete, please add the line "#End of Code" at the bottom of the generated code.

[...]

6) Do not include comments in the code.

7) When asked about an address, reference “from geopy.geocoders import ArcGIS”.

8) Make sure the result is visible as a print statement.

9) Do not use "op", instead use "predicate" when generating code for spatial functions.

""""

A2. Sample Prompt-Completion Pairs Used for Fine-Tuning

Below are selected examples of natural language prompts and corresponding Python code completions used in the fine-tuning dataset. Each completion was designed to reflect proper geospatial processing techniques using Python libraries such as GeoPandas and aligned with the real data structures provided during training.

Example 1: Driving-Time Accessibility Query Using an Address

Prompt:

How many clinics are accessible within 30 minutes of 1500 Maplewood Dr, Richmond, VA?

Completion:

```
address = "1500 Maplewood Dr, Richmond, VA"
coordinates = get_coordinates(address)
if coordinates:
    mapbox_token = mapbox_api
    minutes = 30
    isochrone_geojson = get_isochrone(coordinates, mapbox_token, minutes=minutes)
    if isochrone_geojson:
        clinics_within_isochrone = find_clinics_within_isochrone(isochrone_geojson, healthcare)
        if len(clinics_within_isochrone) == 0:
            print(f'Sadly, no clinics are accessible within {minutes} minutes of {address}.')
        else:
            print(f'There are {len(clinics_within_isochrone)} clinics accessible within {minutes}
```

```

minutes of {address}.)
else:
    print(f"The address '{address}' could not be found. Please try again.")
#End of Code

```

Example 2: County-Based Query with Fuzzy Name Matching

Prompt:

How many clinics are located in Fairfax County?

Completion:

```

input_name = "Fairfax County"
corrected_name = correct_name(input_name, true_county_names)
place = counties[counties["NAMELSAD"] == corrected_name]
county_boundary = place.geometry.union_all()
clinics = healthcare[healthcare.within(county_boundary)]
if len(clinics) == 0:
    print(f"Currently, there are no clinics available in {corrected_name}.")
else:
    print(f"Good news! {corrected_name} has {len(clinics)} clinics available.")
#End of Code

```

A3. Mathematical Formalization of Geospatial Operations

This appendix presents formal mathematical descriptions of the core spatial operations that underpin the queries solved by the fine-tuned LLM in this study.

Buffer and Proximity

$$\begin{aligned}
 \mathbf{B}(\mathbf{c}, r) &= \{\mathbf{x} \in \mathbf{R}^2 \mid d(\mathbf{c}, \mathbf{x}) \leq r\} \\
 d(\mathbf{c}, \mathbf{x}) &= \sqrt{(x_1 - c_1)^2 + (x_2 - c_2)^2} \\
 \mathbf{C} &= \{\mathbf{p}_i \in \mathbf{P} \mid d(\mathbf{c}, \mathbf{p}_i) \leq r\}
 \end{aligned}$$

The buffer operation identifies all point features located within a specified straight-line distance

from a reference point. In this equation, \mathbf{c} represents the center point, such as the centroid of a county or ZIP code, with coordinates $(\mathbf{c}_1, \mathbf{c}_2)$ and \mathbf{r} represents the buffer radius (e.g., 10 miles). The function $\mathbf{d}(\mathbf{c}, \mathbf{x})$ calculates the Euclidean distance between the center point \mathbf{c} and any other point \mathbf{x} in the two-dimensional space \mathbf{R}^2 . The set $\mathbf{B}(\mathbf{c}, \mathbf{r})$ defines the area that includes all points located within the distance \mathbf{r} from point \mathbf{c} . The set \mathbf{C} specifically includes all points \mathbf{p}_i from the dataset \mathbf{P} (e.g., clinics) that satisfy the condition of being within the buffer distance \mathbf{r} from the center point.

Driving Time Isochrones

$$\mathbf{I}(\mathbf{c}, \mathbf{t}) = \{\mathbf{x} \in \mathbf{R}^2 \mid \mathbf{T}(\mathbf{c}, \mathbf{x}) \leq \mathbf{t}\}$$

$$\mathbf{C} = \{\mathbf{p}_i \in \mathbf{P} \mid \mathbf{p}_i \in \mathbf{I}(\mathbf{c}, \mathbf{t})\}$$

The driving-time isochrone operation identifies all point features that are reachable within a specified travel time along the road network from a given reference point. In this formulation, \mathbf{c} represents the origin point, typically a user-defined address, and \mathbf{t} represents the driving time threshold in minutes (for example, 15 minutes). The function $\mathbf{T}(\mathbf{c}, \mathbf{x})$ calculates the driving time between the origin point \mathbf{c} and any other location \mathbf{x} in the spatial domain \mathbf{R}^2 , based on the structure of the road network. The isochrone area $\mathbf{I}(\mathbf{c}, \mathbf{t})$ represents the set of all points that can be reached from point \mathbf{c} within the specified driving time \mathbf{t} . The set \mathbf{C} consists of all point features \mathbf{p}_i in the dataset \mathbf{P} (e.g., clinic locations) that are located within the boundaries of the isochrone $\mathbf{I}(\mathbf{c}, \mathbf{t})$, meaning those clinics are accessible within the specified travel time.

References

1. Jiang, H.; Li, M.; Witte, P.; Geertman, S.; Pan, H. Urban Chatter: Exploring the potential of ChatGPT-like and generative AI in enhancing planning support. *Cities* **2025**, *158*, 105701, doi:<https://doi.org/10.1016/j.cities.2025.105701>.
2. Zhang, Y.; Wei, C.; He, Z.; Yu, W. GeoGPT: An assistant for understanding and processing geospatial tasks. *International Journal of Applied Earth Observation and Geoinformation* **2024**, *131*, 103976, doi:<https://doi.org/10.1016/j.jag.2024.103976>.
3. Ullah, A.; Qi, G.; Hussain, S.; Ullah, I.; Ali, Z. The role of llms in sustainable smart cities: Applications, challenges, and future directions. *arXiv preprint arXiv:2402.14596* **2024**.
4. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. *Language Models are Few-Shot Learners*; 2020.
5. Fang, C.; Miao, N.; Srivastav, S.; Liu, J.; Zhang, R.; Fang, R.; Tsang, R.; Nazari, N.; Wang, H.; Homayoun, H. Large language models for code analysis: Do {LLMs} really do their job? In Proceedings of the 33rd USENIX Security Symposium (USENIX Security 24), 2024; pp. 829-846.
6. Akinboyewa, T.; Li, Z.; Ning, H.; Lessani, M.N. GIS copilot: Towards an autonomous GIS agent for spatial analysis. *arXiv preprint arXiv:2411.03205* **2024**.
7. Zhang, Y.; Wang, Z.; He, Z.; Li, J.; Mai, G.; Lin, J.; Wei, C.; Yu, W. BB-GeoGPT: A framework for learning a large language model for geographic information science. *Information Processing & Management* **2024**, *61*, 103808, doi:<https://doi.org/10.1016/j.ipm.2024.103808>.
8. Hochmair, Hartwig H.; Juhász, L.; Kemp, T. Correctness Comparison of ChatGPT-4, Gemini, Claude-3, and Copilot for Spatial Tasks. *Transactions in GIS* **2024**, *28*, 2219-2231, doi:<https://doi.org/10.1111/tgis.13233>.
9. Mooney, P.; Cui, W.; Guan, B.; Juhász, L. *Towards Understanding the Geospatial Skills of ChatGPT: Taking a Geographic Information Systems (GIS) Exam*; 2023.
10. Mahmoudi, H.; Camboim, S.; Brovelli, M.A. Development of a Voice Virtual Assistant for the Geospatial Data Visualization Application on the Web. *ISPRS International Journal of Geo-Information* **2023**, *12*, doi:10.3390/ijgi12110441.
11. Morocho, V.; Achig, R.; Bustamante, J.; Mendieta, F. Virtual Assistants to bring geospatial information closer to a smart citizen. In Proceedings of the 2022 IEEE Sixth Ecuador

Technical Chapters Meeting (ETCM), 11-14 Oct. 2022, 2022; pp. 01-06.

12. Wang, S.; Tao, H.; Huang, X.; Yun, L.; Ce, Z.; Huan, N.; Rui, Z.; Zhenlong, L.; and Ye, X. GPT, large language models (LLMs) and generative artificial intelligence (GAI) models in geospatial science: a systematic review. *International Journal of Digital Earth* **2024**, *17*, 2353122, doi:10.1080/17538947.2024.2353122.
13. Kim, J.; Lee, J.; Jang, K.M.; Lourentzou, I. Exploring the limitations in how ChatGPT introduces environmental justice issues in the United States: A case study of 3,108 counties. *Telematics and Informatics* **2024**, *86*, 102085, doi:<https://doi.org/10.1016/j.tele.2023.102085>.
14. Akinboyewa, T.; Ning, H.; Lessani, M.N.; Li, Z. Automated floodwater depth estimation using large multimodal model for rapid flood mapping. *Computational Urban Science* **2024**, *4*, 12.
15. Jang, K.M.; and Kim, J. Multimodal Large Language Models as Built Environment Auditing Tools. *The Professional Geographer* **2025**, *77*, 84-90, doi:10.1080/00330124.2024.2404894.
16. Jiang, Y.; Yang, C. Is ChatGPT a Good Geospatial Data Analyst? Exploring the Integration of Natural Language into Structured Query Language within a Spatial Database. *ISPRS International Journal of Geo-Information* **2024**, *13*, doi:10.3390/ijgi13010026.
17. Mansourian, A.; Oucheikh, R. ChatGeoAI: Enabling Geospatial Analysis for Public through Natural Language, with Large Language Models. *ISPRS International Journal of Geo-Information* **2024**, *13*, doi:10.3390/ijgi13100348.
18. Ning, H.; Zhenlong, L.; Temitope, A.; and Lessani, M.N. An autonomous GIS agent framework for geospatial data retrieval. *International Journal of Digital Earth* **2025**, *18*, 2458688, doi:10.1080/17538947.2025.2458688.
19. Renshaw, A.; Lourentzou, I.; Lee, J.; Crawford, T.; Kim, J. Comparing the Spatial Querying Capacity of Large Language Models: OpenAI's ChatGPT and Google's Gemini Pro. *The Professional Geographer* **2025**, *77*, 1-13, doi:10.1080/00330124.2024.2434455.
20. Zhang, Y.; He, Z.; Li, J.; Lin, J.; Guan, Q.; Yu, W. *MapGPT: An Autonomous Framework for Mapping by Integrating Large Language Model and Cartographic Tools*; 2024.
21. Zhang, M.; He, J.; Lei, S.; Yue, M.; Wang, L.; Lu, C.-T. Can llm find the green circle? investigation and human-guided tool manipulation for compositional generalization. In

Proceedings of the ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2024; pp. 11996-12000.

22. Tao, R.; Xu, J. Mapping with ChatGPT. *ISPRS International Journal of Geo-Information* **2023**, *12*, doi:10.3390/ijgi12070284.
23. Li, Z.; and Ning, H. Autonomous GIS: the next-generation AI-powered GIS. *International Journal of Digital Earth* **2023**, *16*, 4668-4686, doi:10.1080/17538947.2023.2278895.
24. Wei, C.; Yifan, Z.; Xinru, Z.; Ziyi, Z.; Zhiyun, W.; Jianfeng, L.; Qingfeng, G.; and Yu, W. GeoTool-GPT: a trainable method for facilitating Large Language Models to master GIS tools. *International Journal of Geographical Information Science* **2025**, *39*, 707-731, doi:10.1080/13658816.2024.2438937.
25. Rahman, F.; Oliver, R.; Buehler, R.; Lee, J.; Crawford, T.; Kim, J. Impacts of point of interest (POI) data selection on 15-Minute City (15-MC) accessibility scores and inequality assessments. *Transportation Research Part A: Policy and Practice* **2025**, *195*, 104429, doi:<https://doi.org/10.1016/j.tra.2025.104429>.
26. Kim, J.; Karki, S.; Brickhouse, T.; Vujicic, M.; Nasseh, K.; Wang, C.; Zhang, M. Navigating Disparities in Dental Health-A Transit-Based Investigation of Access to Dental Care in Virginia. *Community Dent Oral Epidemiol* **2025**, *53*, 117-124, doi:10.1111/cdoe.13015.
27. Yiannakoulis, N. Spatial intelligence and contextual relevance in AI-driven health information retrieval. *Applied Geography* **2024**, *171*, 103392, doi:<https://doi.org/10.1016/j.apgeog.2024.103392>.
28. Ong, J.C.L.; Seng, B.J.J.; Law, J.Z.F.; Low, L.L.; Kwa, A.L.H.; Giacomini, K.M.; Ting, D.S.W. Artificial intelligence, ChatGPT, and other large language models for social determinants of health: Current state and future directions. *Cell Reports Medicine* **2024**, *5*, doi:10.1016/j.xcrm.2023.101356.
29. OpenAI. Advancing cost-efficient intelligence. Available online: <https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/> (accessed on December 13).
30. OpenAI. Fine-tuning guide. Available online: <https://platform.openai.com/docs/guides/fine-tuning> (accessed on December 13).

31. Gramacki, P.; Martins, B.; Szymański, P. Evaluation of Code LLMs on Geospatial Code Generation. In Proceedings of the Proceedings of the 7th ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery, 2024; pp. 54-62.
32. Hou, S.; Shen, Z.; Zhao, A.; Liang, J.; Gui, Z.; Guan, X.; Li, R.; Wu, H. GeoCode-GPT: A Large Language Model for Geospatial Code Generation Tasks. *arXiv preprint arXiv:2410.17031* **2024**.
33. Hou, S.; Jiao, H.; Shen, Z.; Liang, J.; Zhao, A.; Zhang, X.; Wang, J.; Wu, H. Chain-of-Programming (CoP): Empowering Large Language Models for Geospatial Code Generation. *arXiv preprint arXiv:2411.10753* **2024**.
34. Ikumapayi, N. Automated Front-End Code Generation Using OpenAI: Empowering Web Development Efficiency. *SSRN Electronic Journal* **2023**, doi:10.2139/ssrn.4590704.
35. Gupta, D.V.; Ishaqui, A.S.A.; Kadiyala, D.K. Geode: A Zero-shot Geospatial Question-Answering Agent with Explicit Reasoning and Precise Spatio-Temporal Retrieval. *arXiv preprint arXiv:2407.11014* **2024**.
36. Dai, H.; Li, Y.; Liu, Z.; Zhao, L.; Wu, Z.; Song, S.; Shen, Y.; Zhu, D.; Li, X.; Li, S. Ad-autogpt: an autonomous gpt for alzheimer's disease infodemiology. *arXiv preprint arXiv:2306.10095* **2023**.
37. Xu, L.; Zhao, S.; Lin, Q.; Chen, L.; Luo, Q.; Wu, S.; Ye, X.; Feng, H.; Du, Z. Evaluating Large Language Models on Spatial Tasks: A Multi-Task Benchmarking Study. *arXiv preprint arXiv:2408.14438* **2024**.
38. Gupta, S.; Nandwani, Y.; Yehudai, A.; Mishra, M.; Pandey, G.; Raghu, D.; Joshi, S. *Selective Self-Rehearsal: A Fine-Tuning Approach to Improve Generalization in Large Language Models*; 2024.
39. Plewe, B.; Dibiase, D.; Demers, M.; Johnson, A.; Kemp, K.; Wentz, E. *Geographic Information Science & Technology Body of Knowledge*; 2006.
40. Engine, A.G. Nearest Neighbors. Available online: <https://developers.arcgis.com/geoanalytics/tools/nearest-neighbors> (accessed on July 2).
41. Masis, T.; O'Connor, B. Where on earth do users say they are?: geo-entity linking for noisy multilingual user input. *arXiv preprint arXiv:2404.18784* **2024**.
42. Bell, S.; Marlow, T.; Wombacher, K.; Hitt, A.; Parikh, N.; Zsom, A.; Frickel, S.

Automated data extraction from historical city directories: The rise and fall of mid-century gas stations in Providence, RI. *PLOS ONE* **2020**, *15*, e0220219, doi:10.1371/journal.pone.0220219.

43. Martynov, N.; Baushenko, M.; Kozlova, A.; Kolomeytseva, K.; Abramov, A.; Fenogenova, A. A methodology for generative spelling correction via natural spelling errors emulation across multiple domains and languages. *arXiv preprint arXiv:2308.09435* **2023**.
44. Shaw, N. Geographical Information Systems and Health: Current State and Future Directions. *Healthcare informatics research* **2012**, *18*, 88-96, doi:10.4258/hir.2012.18.2.88.
45. Fuzzywuzzy: Fuzzy String Matching in Python. Available online: <https://github.com/seatgeek/fuzzywuzzy> (accessed on July 2).
46. Lee, S.; Jang, S.; Jang, S.; Lee, D.; Yu, H. Exploring Language Model's Code Generation Ability with Auxiliary Functions. *arXiv preprint arXiv:2403.10575* **2024**.
47. Han, Y.; Liu, J.; Luo, A.; Wang, Y.; Bao, S. Fine-Tuning LLM-Assisted Chinese Disaster Geospatial Intelligence Extraction and Case Studies. *ISPRS International Journal of Geo-Information* **2025**, *14*, 79, doi:10.3390/ijgi14020079.
48. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), 2019; pp. 4171-4186.
49. Manvi, R.; Khanna, S.; Mai, G.; Burke, M.; Lobell, D.; Ermon, S. Geollm: Extracting geospatial knowledge from large language models. *arXiv preprint arXiv:2310.06213* **2023**.
50. Frontier AI LLMs, Assistants, Agents, Services | Mistral AI. Available online: <https://mistral.ai/> (accessed on July 2).
51. Deepseek. Available online: <https://deep-seek.chat> (accessed on July 2).
52. Lin, Q.; Hu, R.; Li, H.; Wu, S.; Li, Y.; Fang, K.; Feng, H.; Du, Z.; Xu, L. *ShapefileGPT: A Multi-Agent Large Language Model Framework for Automated Shapefile Processing*; 2024.
53. Openrouteservice. Available online: <https://openrouteservice.org/> (accessed on July 2).
54. Nominatim. Available online: <https://nominatim.org/> (accessed on July 2).
55. Jing, C.; Du, M.; Li, S.; Liu, S. Geospatial Dashboards for Monitoring Smart City Performance. *Sustainability* **2019**, *11*, doi:10.3390/su11205648.

56. Praharaj, S.; Wentz, E. BUILDING COMMUNITY RESILIENCE THROUGH GEOSPATIAL INFORMATION DASHBOARDS. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2022**, *XLVIII-4/W5-2022*, 151-157, doi:10.5194/isprs-archives-XLVIII-4-W5-2022-151-2022.
57. Lwin, K.K.; Sekimoto, Y.; Takeuchi, W.; Zettsu, K. City Geospatial Dashboard: IoT and Big Data Analytics for Geospatial Solutions Provider in Disaster Management. In Proceedings of the 2019 International Conference on Information and Communication Technologies for Disaster Management (ICT-DM), 18-20 Dec. 2019, 2019; pp. 1-4.
58. Dong, Y.; Jiang, X.; Jin, Z.; Li, G. Self-collaboration Code Generation via ChatGPT. *ACM Transactions on Software Engineering and Methodology* **2024**, *33*, doi:10.1145/3672459.
59. Bernhäuserová, V.; Krajňáková, L.; Hátlová, K.; Hanus, M. The Limits of GIS Implementation in Education: A Systematic Review. *ISPRS International Journal of Geo-Information* **2022**, *11*, 592, doi:10.3390/ijgi11120592.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

IV. Conclusion

This thesis brings together two intersecting challenges in modern geography and public health: the persistent inequity in healthcare accessibility, and the steep technical barrier that often limits broader participation in geospatial analysis. Through a combination of traditional spatial modeling and innovative artificial intelligence techniques, this research contributes both conceptual insights and practical tools for addressing these issues.

The first study highlights how access to dental care in Virginia varies significantly depending on transportation mode, region, and Medicaid status. By applying a modified two-step floating catchment area (2SFCA) method across six diverse regions, the analysis reveals that public transit access is not only lower than driving access but also far more unequal, especially for Medicaid-accepting clinics. These disparities are compounded by sociodemographic factors such as poverty and race, showcasing the need for geographically targeted policies and interventions.

The second study moves beyond analysis into tool-building, demonstrating how fine-tuned large language models can translate natural language into executable Python code for spatial analysis. Trained on over 600 prompt-response pairs rooted in Virginia health data, the fine-tuned GPT-4o-mini model achieved nearly 89.7% accuracy and significant gains in computational efficiency. Perhaps most importantly, this approach enables real-time interaction with spatial data through chatbot or dashboard interfaces, reducing reliance on technical expertise and opening the door to a more inclusive geospatial future.

Together, these studies demonstrate that geospatial science and generative AI are not isolated domains, but mutually reinforcing. When integrated thoughtfully, they can illuminate inequities, automate complex workflows, and empower both experts and non-experts to ask better spatial questions. As spatial data continues to grow in relevance for urban planning, public health, and environmental management, the ability to interact with that data in natural, intuitive ways will be critical for equitable and informed decision-making.

Looking forward, future research should expand these methods to other geographic regions, health domains, and population groups. Additionally, continued development of AI-powered spatial tools should prioritize generalizability, multimodal integration (e.g., combining raster and

vector data), and ethical safeguards. By doing so, we can ensure that spatial technology evolves not only in power, but in accessibility, supporting a more just and spatially aware society.