Urban Mobility Assessment Using LLMs

Prabin Bhandari
pbhanda2@gmu.edu
Department of Computer Science
George Mason University
Fairfax, Virginia, USA

Antonios Anastasopoulos
antonis@gmu.edu
Department of Computer Science
George Mason University
Fairfax, Virginia, USA

Dieter Pfoser
dpfoser@gmu.edu
Department of Geography and
Geoinformation Science
George Mason University
Fairfax, Virginia, USA

ABSTRACT

In urban science, understanding mobility patterns and analyzing how people move around cities helps improve the overall quality of life and supports the development of more livable, efficient, and sustainable urban areas. A challenging aspect of this work is the collection of mobility data through user tracking or travel surveys, given the associated privacy concerns, noncompliance, and high cost. This work proposes an innovative AI-based approach for synthesizing travel surveys by prompting large language models (LLMs), aiming to leverage their vast amount of relevant background knowledge and text generation capabilities. Our study evaluates the effectiveness of this approach across various U.S. metropolitan areas by comparing the results against existing survey data at different granularity levels. These levels include (i) pattern level, which compares aggregated metrics such as the average number of locations traveled and travel time, (ii) trip level, which focuses on comparing trips as whole units using transition probabilities, and (iii) activity chain level, which examines the sequence of locations visited by individuals. Our work covers several proprietary and open-source LLMs, revealing that open-source base models like Llama-2, when fine-tuned on even a limited amount of actual data, can generate synthetic data that closely mimics the actual travel survey data and, as such, provides an argument for using such data in mobility studies.

CCS CONCEPTS

• **Computing methodologies** → *Modeling and simulation.*

KEYWORDS

Large Language Models, Travel Data, Travel Survey, Travel Survey Data Simulation

ACM Reference Format:
Prabin Bhandari, Antonios Anastasopoulos, and Dieter Pfoser. 2024. Urban Mobility Assessment Using LLMs. In *The 32nd ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL '24), October 29-November 1, 2024, Atlanta, GA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3678717.3691221

SIGSPATIAL '24, October 29-November 1, 2024, Atlanta, GA, USA

ACM ISBN 979-8-4007-1107-7/24/10
https://doi.org/10.1145/3678717.3691221

1 INTRODUCTION

Analyzing mobility patterns in urban areas has become a critical area of research given the population growth in (mega)cities and associated space and resource constraints. Analyzing how people navigate cities is essential for several tasks that have the ultimate goal of improving the quality of life of residents and supporting the development of more livable, efficient, and sustainable urban areas. Given the right insight, city planners can design better transportation systems, reduce traffic congestion, reduce environmental impacts, and ensure equitable access to transportation for all residents.

Mobility data is the cornerstone of this assessment. However, collecting such data presents significant challenges. Current methods include user tracking [2, 18, 28] and travel surveys [21, 25]. Each has its own set of issues such as privacy concerns, participant noncompliance, and high cost. This paper proposes an innovative approach to the assessment of urban mobility using artificial intelligence and specifically large language models to tap into the existing collective wisdom and thus to overcome these challenges.

Mobility data, which captures the movement of people, is foundational for urban mobility assessment. While the rise of 5G and IoT technologies promises a great avenue to collect mobility data using GPS tracking, this approach has significant issues. Collecting GPS data raises privacy concerns, and even when the data is collected securely and ethically, many people are reluctant to opt-in to such programs. In addition, pure tracking data does not capture the context and purpose of a trip. Here, an alternative method is conducting travel surveys. These surveys collect information about an individual, and their household, and use a travel diary, which captures a person's movements on a particular day. Although different approaches exist, in general, surveys ask participants to log their start and end time, start and end location, mode of travel, and purpose of the journey in a travel diary. The quality of such surveys is often disputed given their low response rate and associated high cost, for example, the US National Highway Travel Survey (NHTS) from 2017 has an overall response rate of only 15.6% [34].

Those concerns have motivated researchers to explore simulation-based techniques to gather synthetic travel survey data [11]. In this context, our work leverages large language models (LLM) to generate travel surveys, offering a promising alternative to traditional methods and improving the evaluation of urban mobility.

LLMs have revolutionized the field of natural language processing and artificial intelligence, eliminating the need for task-specific models trained using vast amounts of labeled datasets. With the advent of pre-training techniques, LLMs, which have a large number of parameters in them, can be pre-trained without any labeled data. This pre-training allows LLMs to encode different types of

knowledge within their parameters, effectively acting as knowledge bases [23]. LLMs also encode world knowledge and exhibit common sense reasoning capabilities, enabling them to understand and generate human-like text across various contexts. Demonstrating this capability, Brown et al. [6] showed that sufficiently scaled LLMs like GPT-3 can handle diverse downstream tasks just by passing a task description, with or without a few sample task examples, as context. This technique is known as Prompting. Moreover, advancements in prompting techniques [3] have made it possible to handle complex tasks, including reasoning.

We hypothesize that since LLMs are trained on a vast amount of textual data, they could potentially encode travel-related data within their parameters. This suggests that simulating travel surveys using LLMs could present a promising alternative. In our study, we prompt LLMs, including Llama-2 [29], Gemini-Pro [9], and GPT-4 [1], to simulate a travel survey.

In our approach, an LLM generates a travel survey by completing the travel diary entries as a survey participant. To validate the results, we compare them to actual travel surveys, specifically the 2017 NHTS data. The evaluation of the results occurs at the (i) pattern level, i.e., aggregate metrics such as the number of locations visited or total travel time, (ii) trip level, i.e., comparing trips between location pairs using transition probabilities, and (iii) activity chain level, i.e., comparing daily location sequences. Our evaluation covers different US Statistical Metropolitan areas and compares them with the 2017 NHTS data. In another experiment, we also compare our LLM-based generation results with a pattern-of-life simulation, an agent-based model (ABM) simulation [15].

Our findings reveal that LLMs even without fine-tuning or alignment to better follow human instructions, encode travel-related information. However, fine-tuning these LLMs even with a small subset of actual travel surveys enables them to better mimic the real-world surveys, surpassing the performance of existing simulation techniques.

The remainder of this paper is structured as follows. Section 2 reviews related work in the field of travel survey simulation for urban mobility assessment. An overview of our proposed LLM-based travel survey simulation system is presented in Section 3. Section 4 discusses the methodology used to assess the quality of our generated data. Section 5 details our experimental setup and results. We then compare our system with an agent-based simulation approach in Section 6. Finally, Section 7 provides conclusions and directions for future work.

2 RELATED WORK

Researchers have explored various techniques to simulate travel surveys, aiming not to replace the instrument but to complement and extend collected travel survey data.

One of the first techniques was based on the Monte Carlo method [30]. Greaves and Stopher [12] proposed a method that first employed the Classification and Regression Tree method [5] to classify the households from an actual survey into clusters based on travel attributes. Subsequently, within each cluster, the variability of trip attributes was captured as a probability distribution. This approach was then applied to households within the location of interest, where Monte Carlo Sampling (MCS) was utilized to select values from the appropriate probability distribution. Initially applied in Baton Rouge, Louisiana, this technique has been validated in various locations, including the Dallas-Fort Worth and Salt Lake City metropolitan area [27], and Adelaide and Sydney in Australia [24]. A similar approach was adopted by Mohammadian et al. [22], who used a synthesized population for the target location and employed neural networks to facilitate the transferability of clusters from actual survey data to the synthesized population. Although this method accurately predicted trip rates, it failed to capture other trip characteristics like mode, departure time, and travel time. The authors attribute this to the method's inability to reflect contextual differences between locations. In contrast, our approach mitigates this issue, as recent studies have demonstrated the ability of LLMs to discern between locations when provided with the proper context [4]. In contrast, our approach addresses this issue by leveraging LLMs' ability to distinguish locations with proper context [4].

The method described by Greaves and Stopher [12] generates each trip in a chain independently of prior trips, which is also by default handled by the autoregressive nature of LLMs that we employ. Going beyond, researchers have also tried simulating trips based on previous trips taken in the chain [14, 17]. The main idea here is to generate weighted transition probabilities, as we have used in our evaluations. However, these methods typically rely on second-order transition probabilities only, whereas during our model fine-tuning, we incorporate higher-order transition probabilities.

Additionally, another technique involves simulating real-world location-based social network (LBSN) datasets, based on typical human behavior, also known as patterns of life [15, 16]. In this simulation, agents with needs similar to real individuals are tracked to generate LBSN datasets resembling actual human interactions. One advantage of this technique is that it can generate precise location information, which our system currently lacks. To evaluate our system against this 'patterns of life' based simulation, we conduct a comparative analysis in Section 6.

Lastly, recent research has explored the use of LLM-based techniques, particularly for human mobility predictions. Wang et al. [32] demonstrated that LLMs excel in accurately predicting the likely next destination in a human trajectory. Similarly, Wang et al. [31] introduced an LLM agent framework designed to generate activities based on real-world human activity data. Building on these advancements, our work shows that LLMs are not only effective at predicting subsequent destinations but also possess encoded mobility information. This inherent capability allows LLMs to assess urban mobility even in the absence of real-world data, highlighting their potential for broader applications in urban planning and mobility studies.

3 SYSTEM OVERVIEW

Our proposed LLM-based system aims to generate synthetic travel survey data to simplify and complement the data aspect of urban mobility assessment. Leveraging the extensive knowledge encoded in LLMs, the system generates travel survey responses by prompting the LLM to fill out a travel diary. The system then processes the LLM-generated output, which is structured as a travel diary table. Table 1 provides a sample table of the generated output. The overall system architecture shown in Figure 1 comprises the following

Table 1: A (reformatted) sample output generated by our LLM-based approach.

Place Visited	Arrival Time	Departure Time	Location Type
Home	00:00 AM	07:30 AM	1
Work	08:00 AM	05:00 AM	3
Home	05:30 AM	11:59 PM	1

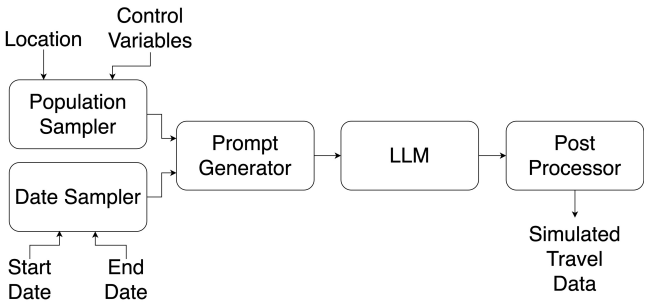

Figure 1: LLM-based travel survey generation system.

components: population sampler, date sampler, prompt generator, LLM, and a post-processor.

Population Sampler To ensure the accuracy of our LLM-based approach, it is important to account for the diverse demographics of the targeted location. To achieve this, we employ a population sampler, sampling individuals based on various control variables such as sex, age group/age, race, school enrollment, participation in the labor force, employment, occupation, marital status, household type, and presence of children under the age of 18 years. To maintain consistency with actual travel surveys, we exclude participants under the age of 16. We use the American Community Survey (ACS) [7] to define the prior probabilities used to sample demographics for locations within the United States of America.

Date Sampler Given that surveys operate within a specific duration, we employ a date sampler, which selects the survey date for each participant by uniformly sampling a date falling between the designated start and end dates of the survey.

Prompt Generator The prompt generator is responsible for crafting inputs for the LLM, using the outputs of the population and date samplers. Our prompts are hand-designed to emulate the format of the National Household Travel Survey 2017 [8] employing travel diary-based completion prompts, aiming to simulate an individual's travel diary. The prompt instructs the LLM to populate a table detailing the places visited by the participant, along with the arrival and departure times and the location types. Specifically, the pre-specified columns are: `'Place Visited'`, `'Arrival Time'`, `'Departure Time'`, and `'Location Type'`. An illustrative prompt is shown in Appendix A in Figure A1. We use the location type categorization used by the National Household Travel Survey 2017. We have provided the categorization in Appendix C in Table C1.

Large Language Model We prompt the LLMs with the output generated by the prompt generator and assess their efficacy in simulating travel surveys using three LLMs: Llama-2, Gemini, and GPT-4. In our Llama-2 setup, we use top-k sampling-based decoding with $k = 50$ and a temperature of 1. Top-k sampling limits the token pool while decoding to the k most likely options at each step. Temperature controls the randomness during token selection. A higher temperature value increases randomness, while a lower temperature value makes the output more deterministic. For Gemini and GPT-4, we use the default sampling-based decoding strategy offered in their Application programming interface (API).

NHTS-2017 Location Types	New Location Types
Regular home activities	Home
Work from home	Home
Work	Work
Work-related meeting/trip	Work
Volunteer activities	Community
Drop off /pick up someone	In Transit
Change type of transportation	In Transit
Attend school as a student	Education
Attend child care	Care
Attend adult care	Care
Buy goods	Shopping
Buy services	Shopping
Buy meals	Eat Meal
Other general errands	Other
Recreational activities	Recreational
Exercise	Recreational
Visit friends or relatives	Social
Health care visit	Social
Religious or other community activities	Community
Something else	Other

Table 2: Reclassification of the 20 location types in NHTS-2017 survey to 11 location types.

Post-processor While the prompt instructs the LLMs to produce output in a structured tabular format, their responses sometimes include extraneous text or omit required information. Hence, we need to post-process the responses generated by the LLM to extract structured travel data.

First, the post-processor utilizes regular expressions to extract the tabular data from the LLM-generated output. However, it is important to note that not all survey outputs result in accurate table formation. Specifically, for Llama-2 model outputs, the post-processor filters out surveys lacking travel time information or exhibiting travel times exceeding 2 hours between two locations. In addition, specifically for Llama-2 model outputs, we exclude surveys featuring location types categorized as 97 (Other). This decision is based on post-analysis, which revealed a notable prevalence of this location type, possibly due to the LLM trying to play it safe.

Lastly, the post-processor reclassifies the original 20 location types into a new set of 11 location types, following the method of Reia et al. [26], to facilitate an easier in-depth analysis. The reclassification is provided in Table 2.

4 EVALUATION METHODOLOGY

Comparing our generated travel data to actual travel data is inherently complex and requires multiple metrics ranging from aggregate to more detailed levels, such as daily activity chains (cf. Janssens et al. [14]). These levels include **(i) pattern level**, which examines aggregated metrics such as the average number of locations traveled and average travel time, **(ii) trip level**, which focuses on comparing all the trips using transition probabilities, and **(iii) activity-chain level**, which analyzes the sequence of locations visited by individuals.

Pattern level Pattern-level metrics compare the generated and actual travel data at the highest level by examining aggregate measures of travel behavior. At this level, we use the average number of locations visited per participant and the average travel duration as key comparison metrics. Additionally, we analyze location visit counts (visualized with histograms) and travel time distributions (visualized with box plots), comparing the characteristics of our generated data to those of actual travel survey data.

Trip level A trip refers to a movement or a journey made by an individual from one location to another (e.g., 'Home -> Work'). Trips are the fundamental unit of analysis in urban mobility. So, to compare trips, we generate transition probabilities, as shown in Table 3, from our actual and generated survey data. These transition probabilities capture the likelihood of traveling between different locations. We also generate the second-order transition probabilities (example provided in Table C2 in Appendix C). Effectively, the transition probabilities computed over a set of surveys can be presented as a square matrix. This representation allows us to easily quantify the distance between transition probabilities computed over different sets of surveys (actual and generated), by calculating the matrix norm of their difference (i.e., the norm of the subtraction –actual minus generated– of the probability matrices). In our case, we use the Forbenius norm [10]. We additionally focus on *destination* probabilities, i.e., the probabilities of a trip ending at a specific location type, to evaluate the likelihood of each location type across different LLMs and contrasting them with the destination probabilities observed in the actual survey data.

Table 3: Example of a first-order transition probabilities.

x_{t-1}	x_t Home	Work	Community	In Transit	...
Home	0	0.31	0.05	0.07	...
Work	0.21	0	0.03	0.04	...
...	...	...	...	...	...

Activity-chain level Activity chains represent the complete sequence of visited locations during a day (e.g., 'Home -> Work -> Home').

We examine different combinations of the presence of activity chains across the actual and generated data. First, for a given location we analyze if generated activity chains appear in the respective actual survey, i.e., evaluating for a specific location the portion of the survey data that is captured by the generated data. We term this metric *'Chain Precision$_{loc}$'* in our results.

Second, we consider the presence of generated chains within the actual chains across *all* locations. The assumption here is that a specific activity chain is *realistic* if it appears in an actual survey of *any* location. In our tables, we call this metric *'Chain Precision$_{all}$'*.

The previous two metrics are conceptually equivalent to *precision*. We test how many of our generated activity chains are realistic, i.e., similar to the ground truth. Thus, we also need a metric conceptually similar to *recall*. We quantify the presence of activity chains from actual surveys in generated surveys, measuring the representativeness of the generated chains. Specifically, we compute the ratio of the actual chains that are represented by the generated ones. We name this metric as *'Chain Recall'* in our tables.

Finally, we calculate the weighted overlap between the actual and generated activity chains. This weighted overlap can serve as a single measure to compare different LLMs, indicating which LLM generates activity chains that most closely match the actual ratios. Table 4 summarizes the different aspects of these activity chain level metrics for easy reference.

Table 4: Activity-chain level metrics used to compare the generated and actual travel chains. Labels used in results and a short description.

Metric	Description
Chain Precision$_{\text{loc}}$	Generated chain appears in respective actual survey (Resemblance of generated to actual)
Chain Precision$_{\text{all}}$	Generated chain appears in any actual (Presence in other locations also indicates realistic outputs)
Chain Recall	Actual chain appears in generated ones (Representatives of generated to actual)
Weighted overlap between chains	Similarity between actual and generated chains

In summary, assessing generated travel data against actual data requires a multi-level approach. We analyze aggregated metrics, individual trips, and activity chains. These evaluations help us understand how generated data aligns with actual ones, providing insights into the use of LLMs for urban mobility assessment.

5 EXPERIMENTS

Our experiments evaluate the accuracy and, essentially, the viability of our LLM-based travel survey system for various metropolitan areas in the US. We generate synthetic travel data for five different metropolitan areas and compare them to the actual 2017 National Household Travel Survey (NHTS) data. We first introduce our experimental setup and provide a detailed analysis focusing on the San Francisco-Oakland-Hayward metropolitan area. Following this case study, we extend our experiments and discussion to four more metropolitan areas. In a nutshell, the results reveal that our proposed approach is indeed viable. LLMs fine-tuned on even a limited amount of actual data can generate synthetic data that closely resemble actual survey data.

5.1 Experimental Setup

We use our LLM-based travel survey generation system to generate travel data for five different metropolitan areas in the USA: (i) San Francisco-Oakland-Hayward, (ii) Los Angeles-Long Beach-Anaheim, (iii) Washington-Arlington-Alexandria, (iv) Dallas-Fort Worth-Arlington, and (v) Minneapolis-St. Paul-Bloomington.

The population and date samplers, the first two components of our system, are tasked with sampling for participant demographics and for dates to maintain consistency with actual travel surveys. The population sampler uses the 2017 American Community Survey (5-year estimates) to sample individuals. The date sampler samples a date between April 19, 2016, and April 25, 2017, to mimic the NHTS 2017 survey timeline.

LLM is a key component of our system that generates the generated travel data. We use three different LLMs: (i) Gemini-Pro, (ii) GPT-4, and (iii) Llama-2, to compare their effectiveness.

Gemini-Pro and GPT-4-turbo are proprietary LLMs only accessible via an API. We use the Gemini-Pro and GPT-4-turbo versions.

Llama-2, an open-source LLM, is deployed on a GMU research computing GPU cluster. We employ the largest variant of Llama-2, which has 70B parameters. However, we opt for the 8-bit quantized version, as fitting it in our GPU cluster is not feasible otherwise. Model quantization involves employing low-precision data types, such as 8-bit integers (int8), for model inference, reducing computational and memory demands, albeit with a minor performance trade-off.

The pre-trained Llama-2 model can be further fine-tuned to the task-at-hand. Fine-tuning uses a subset of 10,000 randomly sampled surveys from NHTS-2017, including all possible locations. These samples are then converted to produce prompts similar to ours based on the survey metadata (Figure A1 in Appendix A), while the LLM's output should align with the actual travel diary entries (Table 1). Fine-tuning the whole model is prohibitively expensive due to its size. We instead use Low-Rank Adaptation (LoRA) [13], which enhances pre-trained LLMs by freezing their model weights and injecting trainable rank decomposition within each layer of the LLMs transformer architecture. This technique significantly reduces the number of trainable parameters, making fine-tuning LLMs feasible. We employ AdamW [20] as the optimizer, with a learning rate of 0.001 and a cosine learning rate scheduler [19], running for 3 epochs. LoRA-based hyperparameters alpha and rank are set to 128 and 256 respectively.

In the following section, we present the results of employing these different LLMs, to provide an in-depth analysis of their performance for the San Francisco-Oakland-Hayward metropolitan area. In a subsequent section, we use the insights from the detailed study to provide a comprehensive assessment for four more locations.

5.2 Detailed evaluation - San Francisco

What follows is a detailed analysis of the data generated by our LLM-based travel survey system for the San Francisco-Oakland-Hayward metropolitan area by comparing it to the actual 2017 NHTS data using the methodology outlined in Section 4.

Our findings indicate the superior performance of the fine-tuned Llama-2 model, demonstrating its effectiveness even for locations / cities where no actual travel survey data is available for training the model. Below we provide supporting arguments for all levels of analysis (pattern, trip, and activity chain). The final section more closely examines LLM fine-tuning to investigate the impact of training data choices and the model's generalization abilities.

Table 5: Comparison of the average number of visited locations and mean travel time across different models - Llama-2-trained model best matches survey data showcased by lower differences Δ.

	Num of samples	Avg num of visited locations	(Δ)	Travel hours	(Δ)
Actual	4027	5.35		1.74	
Gemini-Pro	1241	6.59	(1.24)	2.52	(0.78)
GPT-4	688	6.96	(1.61)	2.52	(0.78)
Llama-2	1279	**5.55**	**(0.20)**	7.17	(5.43)
Llama-2-trained	1452	**4.97**	**(-0.38)**	**1.53**	**(-0.21)**

Pattern-level evaluation The pattern-level evaluation compares the actual and generated travel data at an aggregate level, i.e., in our cases consider the (average) number of visited locations and total travel time.

Our findings reveal that the Llama-2-trained model closely matches the actual data in terms of aggregate metrics of the average number of visited locations and travel time.

Table 5 presents the metrics for the various models and also indicates the respective sample sizes. Among the four models, the Llama-2-trained model performs best since it produced both, a reasonable average number of visited locations *and* at the same time keeps the travel time (in hours) close to the actual average. The other three models each perform poorly on at least one of the two metrics.

Among the three base models, Llama-2 best matches the survey data in terms of visited locations but does not properly capture travel time. This discrepancy can be attributed to differences in model alignment. Model alignment refers to fine-tuning the model to better match the downstream task. GPT-4 and Gemini-Pro are fine-tuned for better alignment with human instructions, making them adept at handling numerical and temporal data. In contrast, the base Llama-2 model lacks such fine-tuning. However, this alignment of GPT-4 and Gemini-Pro, while advantageous for travel time prediction, also leads to the generation of *more text*, consequently resulting in an abnormally higher number of locations traveled. This trend is also visible in Figure 2 showing the distribution of the number of locations traveled by the survey respondents. For Gemini-Pro and GPT-4, the number of locations traveled far exceeds the actual data, particularly for longer journeys. GPT-4 produces surveys with only 5-10 locations visited. Other models output ranges from 1 to 19 locations, which resemble more closely the distributions of the actual survey data. The plot on the bottom right of Figure 2 shows how closely the fine-tuned Llama-2-trained model matches actual survey data in terms of locations visited.

The limitation of the Llama-2 base model in handling numerical data and consequently travel time is also evident in Figure 3, which shows the travel times of each model in a box plot. The travel

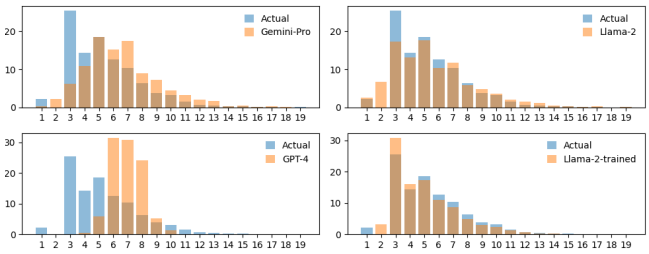

Figure 2: Distribution of the number of locations traveled by survey respondents for the San Francisco-Oakland-Hayward metropolitan area. Llama-2 and its fine-tuned variant show best alignment with actual survey data.

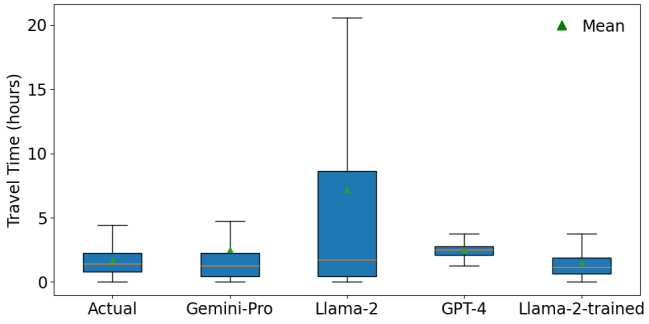

Figure 3: Travel time predictions - Gemini-Pro and GPT-4 models outperform Llama-2. The Llama-2-trained model matches survey travel times. Outliers are removed for better visualization.

time for Llama-2 is as high as 120h (in a survey that captures 24h), indicating its struggle with time-related data. Gemini-pro and GPT-4 better model travel times.

Fine-tuning the Llama-2 model addresses both of these issues. As shown in Figure 3 (right-most box), the fine-tuned model travel times resemble more closely those of the actual survey.

Overall, the pattern-level evaluation of the various models indicates that the Llama-2-based generated data, particularly the *Llama-2-trained model*, most closely matches the actual travel survey data.

Trip-level evaluation The trip-level evaluation compares all the survey's individual trips (between two locations) using transition probabilities. These results suggest that, again, the Llama-2-trained model best matches the travel survey transition probabilities.

In the trip-level comparison, the Llama-2-trained model distinguishes itself. It shows the closest resemblance to the actual transition probabilities, as indicated by the Forbenius norm results of Table 6. Among the base models, Gemini-pro performs best, whereas GPT-4 shows the poorest alignment. This variation in model performance can be further explained by examining the differences in first-order destination probabilities, which reflect the likelihood of a trip ending at a particular location type. Figure 4 shows the differences between the generated and actual survey data ordered by the probabilities of the actual survey. Gemini-Pro and GPT-4 tend to overpredict destinations typically associated with

Table 6: Frobenius norm of transition probability differences between actual and generated data (lower is better). Llama-2-trained data best matches actual survey data.

	Frobenius Norm with 1st order transition prob	Frobenius Norm with 2nd order transition prob
Gemini-Pro	0.063	0.046
GPT-4	0.166	0.130
Llama-2	0.106	0.071
Llama-2-trained	**0.044**	**0.032**

tasks such as "shopping", "dining out", and "recreational activities" (notice the large negative bars). This tendency is likely a result of the models being fine-tuned to respond to queries related to such location types. Llama-2 tends to overpredict other location types, such as "work". However, it more closely captures the actual survey data trends. The fine-tuned Llama-2-trained model overall shows the best performance and alignment (as indicated by having generally smaller positive and negative bars in the figure). The analysis of second-order destination probabilities detailed in the appendix shows a similar trend (see Figure B1 in Appendix B).

In summary, the trip-level evaluation indicates that Llama-2-trained generated data has transition probabilities best resembling those of the actual survey data. This finding is consistent with our pattern-level evaluation.

Activity-chain-level evaluation The third type of evaluation is the most detailed one and compares the generated model results at the activity-chain level by analyzing the sequence of locations visited during a 24h period. Again, the results suggest that the Llama-2-trained model generates activity chains that more closely resemble the actual activity chains of travel surveys. This analysis shows that fine-tuning is crucial: we observe a 1.5-2x improvement in emulating the activity chains from travel surveys over the base Llama-2 model.

Table 7 presents a comprehensive overview of the results for the San Francisco-Oakland-Hayward metropolitan area, showing all metrics following the explanation provided in Table 4.

According to all metrics, Llama-2-trained outperforms all other models. 61% of its generated activity chains appear in the actual survey (Chain Precision$_{loc}$), compared to only around 36% for the base Llama and even lower numbers for Gemini-pro and GPT-4. Moreover, the Llama-2-trained model generates around 80% of all (67% unique) activity chains of the NHTS survey (see Chain Precision$_{all}$). This is a significant improvement over the base models Llama-2, Gemini-pro, and GPT-4, which only reach 50%.

Taking into account the frequency of the activity chains (by appropriate weighting), we find that approximately 47% of the actual activity chains are present in the generated data of the Llama-2-trained model (see Chain Recall), compared to around 39% for the Llama-2 base model and Gemini-pro, and only 2% for GPT-4. Finally, the weighted overlap between the actual and generated activity chains is around 42% for the Llama-2-trained model, exceeding Gemini-pro by more than 20% and GPT-4 by 40%. This shows how

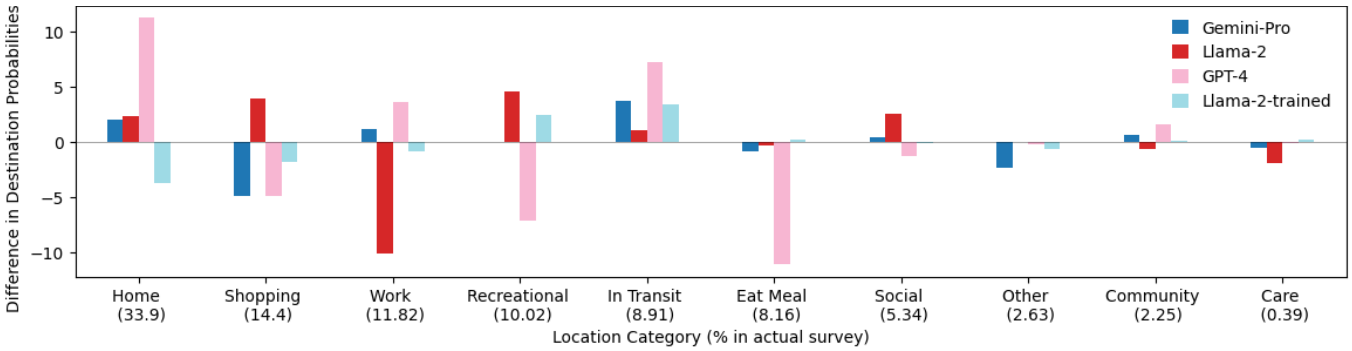

Figure 4: Difference in first order destination probabilities (actual - generated) per location category. Llama-2-trained outperforms (with shorter bars) other models.

Table 7: Activity chain analysis - Llama-2-trained model closely captures the activity chains of actual travel survey. Chain Precision$_{loc}$: presence of generated chains within actual (resemblance); Chain Precision$_{all}$: presence of generated chains within actual of all the cities (chain realism); Chain Recall: presence of actual chains within generated ones (Representativeness); Weighted overlap between chains: weighted similarity of chains.

	Chain Precision$_{loc}$		Chain Precision$_{all}$		Chain Recall		Weight Overlap
high is better for all metrics	%	weighted	%	weighted	%	weighted	of Chains
Gemini-Pro	15.55	29.81	46.22	56.79	7.77	39.21	17.25
GPT-4	7.89	11.92	40.71	52.76	1.63	2.19	1.44
Llama-2	15.54	36.12	43.06	57.39	6.88	39.14	25.42
Llama-2-trained	**30.88**	**61.16**	**65.44**	**80.10**	**11.92**	**48.22**	**42.54**

good the Llama-2-trained model is at representing the activity chains present in the actual survey.

While matching already observed activity chains is a good indicator, it is important to remember that there are many possible activity chains. Hence, we need to also establish that the remaining generated but unmatched activity chains are still realistic. To achieve this, we measure the correctness of unmatched activity chains using the Levenshtein distance [35] (also known as *edit distance*) and reporting the minimum Levenshtein distance between an unmatched activity chain and any actual one. We find that among the remaining unmatched chains, 69% have an edit distance of 1, 22.44% have an edit distance of 2, 6.29% have an edit distance of 4, and the remaining 2.75% have a higher edit distance. This means that unmatched activity chains are not too dissimilar from those found in travel surveys.

Figure 5 plots the cumulative sum of the activity chain counts in relation to the matched activity chain counts across various models, i.e., only generated activity that matches actual survey data are included. As a reference, we also plot the matching activity chains from another metropolitan area (Los Angeles-Long Beach-Anaheim (LA)). Two vertical lines in the plot indicate thresholds of activity chains with less than 10 counts (left blue line) and with only 1 count (right grey line). This shows the sparsity of many chains. From this figure, we can infer that GPT-4 exhibits the least resemblance, followed by Gemini-pro and then the Llama-2 base model (bottom three plots) to the actual data. Finally, the Llama-2-trained model best matches the actual chains, akin to how the

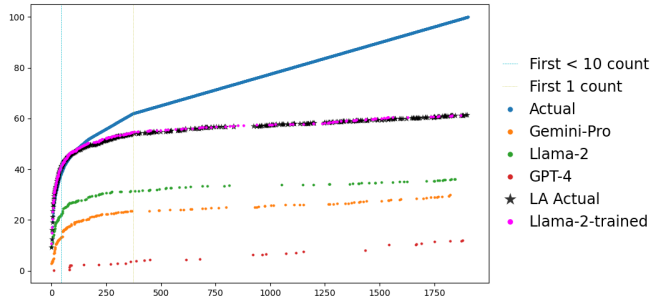

Figure 5: The cumulative sum of activity chain counts for the San Francisco-Oakland-Hayward metropolitan area's actual survey compared to the generated data. Llama-2-trained data closely matches the actual survey. Llama-2-trained and LA actual data are almost overlapping.

activity chains of the Los Angeles-Long Beach-Anaheim metropolitan area match the San Francisco-Oakland-Hayward metropolitan area data (overlapping plots of Llama-2-trained and LA data just below the actual survey data). This result is intriguing, as we will see in the next sections that the difference between the travel surveys of different cities is very nominal, suggesting a noteworthy level of generation accuracy.

We also plot the distribution of the top 100 activity chains by the count of the actual survey in comparison to other generated

surveys in Figure B2 in Appendix B. These demonstrate that the Llama-2-trained results and LA's actual survey data closely match the distribution of San Francisco's activity chains. In contrast, GPT-4 matches only two chains, while Gemini and Llama-2 show better performance than GPT-4 but still fall short of Llama-2-trained.

Overall, activity-chain-level evaluation across various models indicates that the Llama-2-trained model data shows the closest resemblance to the activity chains present in the actual survey.

Are location-specific training data important? We have now established that fine-tuning an LLM to mimic travel surveys leads to producing realistic outputs that closely match real ones. However, the data used to fine-tune the model can potentially play an important role in the model's performance. Ideally, we would want to be able to generate travel surveys even for locations for which no real travel survey data are available – this is in fact the ultimate frontier of synthetic travel data generation!

To test the ability of our generation approach to generalize beyond already-seen locations, we fine-tune another Llama-2 model, using a different fine-tuning dataset. This dataset is curated by excluding the actual travel surveys from all five specific metropolitan areas where our system will be applied (including San Francisco).

Encouragingly, our experiments show that the Llama-2-trained model without training data from the San Francisco-Oakland - Hayward area performs similarly to the model fine-tuned using such local data.

The original fine-tuned Llame-2-trained model that uses local data produces an average of 4.97 locations traveled and a travel time of 1.53h (as discussed previously in Table 5). In comparison, the model fine-tuned without local data shows an average of 4.79 visited locations and a travel time of 1.60h. We consider these slight decrease in the number of locations and increase in travel time to be rather minor. The trip-level evaluation also shows no discernible difference between the model fine-tuned with or without local data. The model fine-tuned with local data has a Forbenius norm of 0.044 for first-order transition probabilities, while the model without local data achieves a slightly better 0.043. Conversely, for second-order transition probabilities, the model fine-tuned with local data has a Forbenius norm of 0.0319, compared to 0.0360 for the model fine-tuned without local data, indicating a slightly worse performance when lacking local training data. The activity-chain level evaluation also indicates a minimal performance drop, with the weighted overlap between the actual and generated chains decreasing from 42.5% for the model fine-tuned with local data to 41% for the one fine-tuned without local data.

Overall, we find that a lack of local training data only very slightly decreases the quality of the generated data, and as such demonstrates the robust generalization capabilities of our LLM-based approach.

In conclusion, the in-depth evaluation of our LLM-based travel survey generation system for the San Francisco area demonstrates that base models like Llama-2 already produce meaningful data. However, when fine-tuned even with a limited amount of data, these models can generate synthetic travel surveys resembling actual ones, even when location-specific training data is not available.

5.3 Overall system performance

Measuring the overall system performance of our travel survey generation system requires comparing results across multiple cities. We have two desiderata for our system. First, that the generated data for the specific city match the actual travel surveys—this is in line with the analysis we performed on San Fransisco in §5.2, which we repeat here for all cities. And second, that generated surveys are *meaningfully localized*: this means that they properly capture local urban infrastructure and population characteristics. To this effect, we compare data generated for a specific city to the generated data for other cities, and we ground this discussion by comparing the actual travel survey data of different cities to each other. We also produce a clustering of the cities, the assumption being that more "similar" cities should also have more similar aggregate travel survey characteristics.

In short, we find that the Llama-2-trained model shows the smallest difference between generated and actual data among the same cities and across cities. Additionally, clustering cities shows that the Llama-2-trained model produces clusters that most closely resemble the actual city clusters, further validating the effectiveness of our approach.

Table 8 offers a consolidated pattern and activity-chain-level analysis for the additional four metropolitan areas. The table reaffirms our earlier finding that Llama-2-trained outperforms all other models. Of the base models, although it struggles with travel time, Llama-2 more closely approximates the actual data considering the pattern and activity-chain-level analyses.

The trip-level analysis for different cities yields interesting insights. Figure 6 presents box plots illustrating the Forbenius norm of the difference between the first-order transition probabilities of actual and generated surveys in various combinations for different LLMs. Plot (i) compares the Forbenius norm for the difference of first-order transition probabilities of actual and generated surveys for the same city across different models, as in the previous analysis. The distribution of norms is lowest for the Llama-2-trained model, which indicates that it best approximates the actual survey data. Plot (ii) displays the norms between the actual and generated surveys for different cities. Effectively, we want to quantify how unique (or not) the generated surveys are to a specific city. The plot is similar to Plot (i), albeit with slightly greater variability. This may seem counterintuitive, but the third plot sheds light on this phenomenon. Plot (iii) shows the norm among generated surveys between different cities, together with the norm between actual surveys for different cities, i.e., how similar is the generated output of one model across different cities and how similar are actual surveys to each other (fifth shaded box in Plot (iii)). The differences are overall much smaller than in Plots (i) and (ii), which means that the generated data of a model does not differ too much across different cities. The same trend is observed for the actual data as well. A takeaway from Figure 6 is that Llama-2-trained outperforms all other models and produces data that best matches actual survey data. Additionally, the similarity between Plots (i) and (ii) suggests that the difference among actual cities is also marginal, which is also shown in Plot (iii), indicating that the survey data for (US) cities are quite similar.

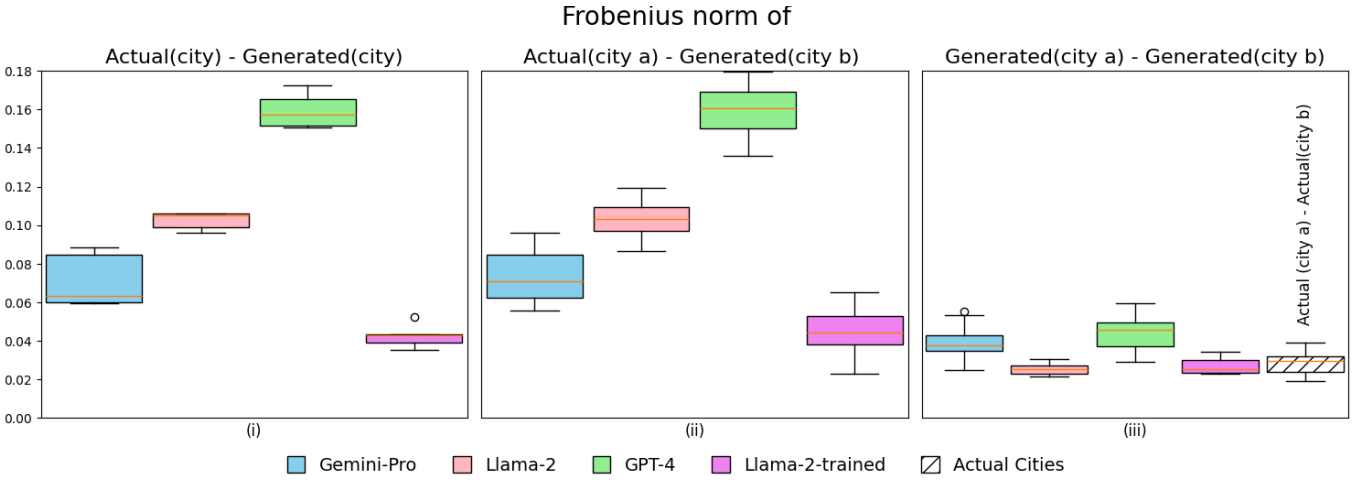

Figure 6: Forbenius norm for first-order transition probabilities of actual and generated surveys for different cities and for different LLMs. Minimal differences among actual cities themselves (see (iii): Actual) justify the minimal differences observed between actual and generated data of different cities (ii). Llama-2-trained shows best similarity (i).

With the transition probabilities from actual and generated surveys from different LLMs, we also conduct Ward *hierarchical clustering* [33]. Using the *actual* surveys, cities are clustered into two groups:

(1) San Francisco-Oakland-Hayward, Los Angeles-Long Beach-Anaheim, and *Washington-Arlington-Alexandria*
(2) Dallas-Fort Worth-Arlington and Minneapolis-St. Paul-Bloomington

Using the generated data, clustering using surveys from the GPT-4 and Llama-2-trained models produces similar results. The only difference is Washington-Arlington-Alexandria moving to the other cluster (Dallas and Minneapolis).

This similarity of clusters, combined with Llama-2-trained generated data's similarity to the actual survey, continues to make a good case for using fine-tuned models in our travel survey generation approach. LLMs fine-tuned on even small amounts of data have the potential to serve as low-cost alternatives to generate travel survey data as part of an effort to simplify urban mobility assessment.

6 COMPARISON TO PATTERNS-OF-LIFE SIMULATION

The Patterns-Of-Life (POL) simulation for location-based social network datasets presents a novel approach to simulate travel surveys. The POL approach is based on an agent based modeling (ABM) framework that uses basic agent needs to generate mobility data for arbitrary geographic locations [15, 16]. We compare the output of this approach to our LLM-based generation system using the pattern, trip, and activity level evaluation methodologies described in Section 4. We are only using Llama-2-trained as it is our best performing LLM model.

Our comparison uses the San Francisco-Oakland-Hayward Metropolitan area with the generated data generated by the Llama-2-trained model and the POL simulation. The latter produces continuous travel data for multiple agents, and to ensure consistency with

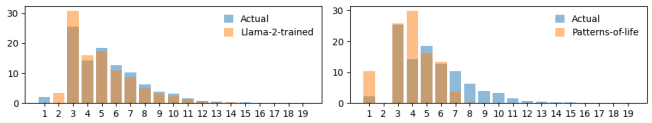

Figure 7: Distribution of the number of locations traveled. Our LLM-based approach better matches the actual distribution.

travel surveys, we sample from all the data available, to match the same size of both simulation techniques. One thing to note is that we could not extract the travel time for the POL simulation, so we exclude the associated metrics from our analysis. Another noteworthy consideration is the limited number of location types available in the POL simulation, which has only six different location types. Consequently, we reclassify the original 20 different location types from NHTS-2017 into the 6 location types provided by POL (Reclassification shown in Table C3 in Appendix C) and modify our system's post-processor to allow for a direct, fair comparison.

The evaluations reveal some stark differences between the two approaches. Our generative model better captures the average number of locations traveled by the survey respondents, as shown in Table 9. Figure 7 shows the distribution of the number of locations. Both systems overpredict the number of visited locations, but the POL simulation has less variability and fails to capture the tail end of the distribution, which the LLM-based approach handles better.

The trip-level evaluation again clearly shows the superiority of the LLM-based approach. Table 10 provides the Forbenius Norm between the first-order and second-order transition probabilities obtained from the actual survey and both techniques. Notably, our approach exhibits a significant, order-of-magnitude lower Forbenius norm than the POL simulation, indicating a substantial difference in their performance. The LLM-based approach significantly outperforms POL by a considerable margin in this metric.

Table 8: Pattern and activity-chain-level analysis for additional cities - Llama-2-trained produces results comparable to actual survey data.

	Avg. no of. locs traveled	(Δ)	Travel hours	(Δ)	Wt. overlap of Actual & Generated
	Washington-Arlington-Alexandria				
Actual	5.15		1.75		-
Gemini-Pro	6.45	(1.30)	2.82	(1.07)	18.89
GPT-4	6.91	(1.76)	2.58	(0.77)	2.7
Llama-2	**5.38**	**(0.23)**	6.43	(4.68)	26.58
Llama-2-trained	4.68	(-0.47)	**1.56**	**(-0.19)**	**40.15**
	Los Angeles-Long Beach-Anaheim				
Actual	5.11		1.63		-
Gemini-Pro	6.56	(1.46)	2.26	(0.63)	17.22
GPT-4	6.88	(1.77)	2.53	(0.90)	1.51
Llama-2	5.63	(0.52)	6.22	(4.59)	27.29
Llama-2-trained	**4.64**	**(-0.47)**	**1.69**	**(0.06)**	**43.12**
	Dallas-Fort Worth-Arlington				
Actual	5.14		1.52		-
Gemini-Pro	6.31	(1.17)	2.31	(0.79)	18.4
GPT-4	6.91	(1.77)	2.53	(1.01)	2.04
Llama-2	**5.42**	**(0.28)**	5.75	(4.23)	28.18
Llama-2-trained	4.68	(-0.45)	**1.56**	**(0.04)**	**44.42**
	Minneapolis-St. Paul-Bloomington				
Actual	5.10		1.61		-
Gemini-Pro	6.24	(1.14)	2.42	(0.81)	16.99
GPT-4	7.10	(2.00)	2.54	(0.93)	2.27
Llama-2	5.73	(0.63)	6.59	(4.98)	24.44
Llama-2-trained	**4.75**	**(0.35)**	**1.44**	**(0.17)**	**45.84**

Table 9: Average locations traveled and weighted overlap of actual vs. generated activity chains. Our LLM-based approach better aligns with actual data than POL simulation.

	No. of. Samples	Avg. no of locations traveled	(Δ)	Weighted overlap of chains
Actual	4027	5.35		-
Llama-2-trained	1435	4.98	(-0.37)	**53.35**
Pattern-of-life	1435	3.98	(-1.37)	27.49

The activity-chain-level evaluation, with the results shown in Table 9, focuses on the weighted overlap of activity chains generated by the different methods. The results in predicting activity chains as shown in Figure 8, where the cumulative sum of the activity chain counts is plotted compared to the matched activity chain counts across both approaches. While the POL approach accurately models the most prevalent activity chains of the actual survey (towards the left of the x-axis), it tends to overpredict them. Moreover, it does not capture the less common activity chains of the actual survey.

Table 10: Transition probabilities comparison (lower is better). LLM-based outputs closely match actual survey trips.

	Frobenius Norm with 1st order transition prob	2nd order transition prob
Llama-2-trained	**0.037**	**0.034**
Patterns-of-life	0.214	0.193

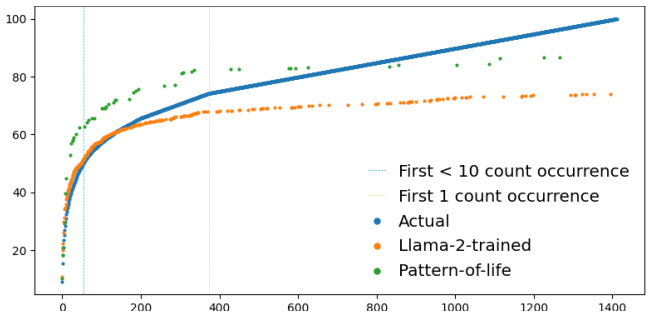

Figure 8: Cumulative sum of activity chain counts. POL simulation captures major patterns but misses less common patterns, while our LLM-based system captures all.

This is shown by the scarcity of single-count chains in the POL result compared to the LLM approach.

In conclusion, the survey, trip, and activity chain level analysis shows that our LLM-based travel survey simulation approach more accurately reflects the actual survey data compared to POL simulation. While the POL simulation certainly has its advantages compared to our technique (like generating precise location data), combining both techniques could lead to an even more robust mobility simulation. We leave this task for future work.

7 CONCLUSION AND FUTURE WORK

We propose a novel approach for urban mobility assessment utilizing Large Language Models (LLMs). Current urban mobility data collection, a cornerstone for urban mobility assessment, often faces significant challenges, including privacy concerns, noncompliance, and high costs. To address these issues, we introduce an LLM-based travel-survey generation system that generates synthetic travel data to facilitate enhanced urban mobility assessment. We devise a set of evaluation criteria to assess the quality of the generated data at the pattern, trip, and activity-chain levels. Our findings highlight the effectiveness of our system, particularly when using base models like Llama-2, which has not been fine-tuned to better follow human instructions and is trained with a limited amount of actual travel data.

Future work will focus on extending this approach to locations outside the US and refining the system to generate more precise location data. This will enhance the transferability and applicability of our system towards global urban mobility assessment.

Code, dataset, and models: https://github.com/gmuggs/Urban-Mobility-LLM

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (Award 2127901) and by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/ Interior Business Center (DOI/IBC) contract number 140D0419C0050. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or the U.S. Government. Additionally, this work was supported by resources provided by the Office of Research Computing, George Mason University and by the National Science Foundation (Award Numbers 1625039, 2018631).

REFERENCES

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report.

[2] Armando Bazzani, Bruno Giorgini, Sandro Rambaldi, Riccardo Gallotti, and Luca Giovannini. 2010. Statistical laws in urban mobility from microscopic GPS data in the area of Florence. *Journal of Statistical Mechanics: Theory and Experiment* 2010, 05 (May 2010), P05001. https://doi.org/10.1088/1742-5468/2010/05/p05001

[3] Prabin Bhandari. 2024. A Survey on Prompting Techniques in LLMs. arXiv:2312.03740 [cs.CL] https://arxiv.org/abs/2312.03740

[4] Prabin Bhandari, Antonios Anastasopoulos, and Dieter Pfoser. 2023. Are Large Language Models Geospatially Knowledgeable?. In *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems* (Hamburg, Germany) *(SIGSPATIAL '23)*. Association for Computing Machinery, New York, NY, USA, Article 75, 4 pages. https://doi.org/10.1145/3589132.3625625

[5] L Breiman, JH Friedman, R Olshen, and CJ Stone. 1984. *Classification and Regression Trees*. Wadsworth, New York. https://doi.org/10.1201/9781315139470

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[7] USC Bureau. 2021. American community survey (acs). *Census. gov. Retrieved November* 1 (2021).

[8] Federal Highway Administration. 2017. *National Household Travel Survey*. Technical Report. U.S. Department of Transportation, Washington, DC. https://nhts.ornl.gov

[9] Gemini Team. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. arXiv:2403.05530 [cs.CL] https://arxiv.org/abs/2403.05530

[10] Gene H. Golub and Charles F. Van Loan. 2013. *Matrix Computations* (4th edition ed.). Johns Hopkins University Press, Philadelphia, PA. https://doi.org/10.1137/1.9781421407944 arXiv:https://epubs.siam.org/doi/pdf/10.1137/1.9781421407944

[11] Stephen Peter Greaves. 2000. *Simulating household travel survey data in metropolitan areas*. Louisiana State University and Agricultural & Mechanical College, Baton Rouge, Louisiana. https://doi.org/10.31390/gradschool_disstheses.7357

[12] Stephen P Greaves and Peter R Stopher. 2000. Creating a synthetic household travel and activity survey: Rationale and feasibility analysis. *Transportation Research Record* 1706, 1 (2000), 82–91.

[13] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685 [cs.CL] https://arxiv.org/abs/2106.09685

[14] Davy Janssens, Geert Wets, Tom Brijs, and Koen Vanhoof. 2004. The simulation of activity diary data using sequential probability distributions. In *Forthcoming in proceedings of the ISCTSC Triennial International Conference on Transport Survey Methods*. Citeseer.

[15] Joon-Seok Kim, Hyunjee Jin, Hamdi Kavak, Ovi Chris Rouly, Andrew Crooks, Dieter Pfoser, Carola Wenk, and Andreas Züfle. 2020. Location-based social network data generation based on patterns of life. In *2020 21st IEEE International Conference on Mobile Data Management (MDM)*. IEEE, 158–167.

[16] Joon-Seok Kim, Hamdi Kavak, Umar Manzoor, Andrew Crooks, Dieter Pfoser, Carola Wenk, and Andreas Züfle. 2019. Simulating urban patterns of life: A geo-social data generation framework. In *Proceedings of the 27th ACM SIGSPATIAL international conference on advances in geographic information systems*. 576–579.

[17] Ryuichi Kitamura, Cynthia Chen, and Ram M Pendyala. 1997. Generation of synthetic daily activity-travel patterns. *Transportation research record* 1607, 1 (1997), 154–162.

[18] Liang Liu, Assaf Biderman, and Carlo Ratti. 2009. Urban mobility landscape: Real time monitoring of urban mobility patterns. In *Proceedings of the 11th international conference on computers in urban planning and urban management*. Citeseer, 1–16.

[19] Ilya Loshchilov and Frank Hutter. 2017. SGDR: Stochastic Gradient Descent with Warm Restarts. In *International Conference on Learning Representations*. OpenReview.net, Vancouver, BC, Canada. https://openreview.net/forum?id=Skq89Scxx

[20] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*. OpenReview.net, New Orleans, LA, USA. https://openreview.net/forum?id=Bkg6RiCqY7

[21] Jeremy Wade Mattson. 2012. *Travel behavior and mobility of transportation-disadvantaged populations: Evidence from the National Household Travel Survey*. Technical Report. Upper Great Plains Transportation Institute Fargo, ND, USA.

[22] Abolfazl Kouros Mohammadian, Mahmoud Javanmardi, and Yongping Zhang. 2010. Synthetic household travel survey data simulation. *Transportation Research Part C: Emerging Technologies* 18, 6 (2010), 869–878.

[23] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language Models as Knowledge Bases?. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 2463–2473. https://doi.org/10.18653/v1/D19-1250

[24] Graham Pointer, Peter R. Stopher, and Philip Bullock. 2004. Monte Carlo Simulation of Household Travel Survey Data for Sydney, Australia: Bayesian Updating Using Different Local Sample Sizes. *Transportation Research Record* 1870, 1 (2004), 102–108. https://doi.org/10.3141/1870-13 arXiv:https://doi.org/10.3141/1870-13

[25] John Pucher and John L Renne. 2003. Socioeconomics of urban travel: evidence from the 2001 NHTS. *Transportation Quarterly* 57, 3 (2003), 49–77.

[26] Sandro M. Reia, Taylor Anderson, Henrique F. Arruda, Kuldip S. Atwal, Shiyang Ruan, Hamdi Kavak, and Dieter Pfoser. 2024. Function and form of U.S. cities. arXiv:2406.04543 [physics.soc-ph] https://arxiv.org/abs/2406.04543

[27] Peter R Stopher and Graham Pointer. 2004. Monte Carlo simulation of household travel survey data with Bayesian updating. *Road & Transport Research* 13, 4 (2004), 22.

[28] Jinjun Tang, Fang Liu, Yinhai Wang, and Hua Wang. 2015. Uncovering urban human mobility from large scale taxi GPS data. *Physica A: Statistical Mechanics and its Applications* 438 (2015), 140–153.

[29] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models.

[30] John Von Neumann and Stanislav Ulam. 1951. Monte carlo method. *National Bureau of Standards Applied Mathematics Series* 12, 1951 (1951), 36.

[31] Jiawei Wang, Renhe Jiang, Chuang Yang, Zengqing Wu, Makoto Onizuka, Ryosuke Shibasaki, Noboru Koshizuka, and Chuan Xiao. 2024. Large Language Models as Urban Residents: An LLM Agent Framework for Personal Mobility Generation. arXiv:2402.14744 [cs.AI] https://arxiv.org/abs/2402.14744

[32] Xinglei Wang, Meng Fang, Zichao Zeng, and Tao Cheng. 2024. Where Would I Go Next? Large Language Models as Human Mobility Predictors. arXiv:2308.15197 [cs.AI] https://arxiv.org/abs/2308.15197

[33] Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association* 58, 301 (1963), 236–244.

[34] Westat. 2018. 2017 NHTS Data User Guide. https://nhts.ornl.gov/assets/2017UsersGuide.pdf

[35] Li Yujian and Liu Bo. 2007. A normalized Levenshtein distance metric. *IEEE transactions on pattern analysis and machine intelligence* 29, 6 (2007), 1091–1095.

A PROMPT TEMPLATE

Figure A1: An example of a travel diary-based prompt.

```
The individual is a 59-year-old female whose racial background
is 'White alone'. Currently, she is not enrolled in school and
is participating in the labor force. She is employed and working
in the 'Business and financial operations occupations' field.
Regarding her marital status, she is married, and lives in a
married couple family. She lives in San Francisco, CA. She has
been selected for a travel survey and has recorded her travel logs
for 2016-05-05 which is a Thursday. She was asked to provide a
list of all the places she visited on her travel date, including
the exact times of arrival and departure, and the location type.
The table format provided was as follows:

| Place Visited | Arrival Time | Departure Time | Location Type |
|———|———|———-|———|
| [Place Name] | [HH:MM AM/PM] | [HH:MM AM/PM] | [Location Type] |
| [Place Name] | [HH:MM AM/PM] | [HH:MM AM/PM] | [Location Type] |
| ... | ... | ... | ... |

She was instructed to fill in each row with the relevant information
for each place she visited on the specified date. If she visited
the same place multiple times on the same date, she was advised to
add a separate row for each visit to that place.

She was reminded of the following:

1. Ensure that 'Home' is included in the list if it was
part of travel activities on the specified date.
2. She was asked to input the exact arrival and departure time as
she noted in her travel diary.
3. She was advised to carefully enter the times, as gaps between
the departure time of the previous place and the arrival time of
the current place indicate the time taken to arrive at the current
location.

For the 'Location Type,' she was instructed to use the corresponding
code from the provided list:
1: Regular home activities (chores, sleep)
2: Work from home (paid)
3: Work
4: Work-related meeting / trip
5: Volunteer activities (not paid)
6: Drop off / pick up someone
7: Change type of transportation
8: Attend school as a student
9: Attend child care
10: Attend adult care
11: Buy goods (groceries, clothes, appliances, gas)
12: Buy services (dry cleaners, banking, service a car, etc)
13: Buy meals (go out for a meal, snack, carry-out)
14: Other general errands (post office, library)
15: Recreational activities (visit parks, movies, bars, etc)
16: Exercise (go for a jog, walk, walk the dog, go to the gym, etc)
17: Visit friends or relatives
18: Health care visit (medical, dental, therapy)
19: Religious or other community activities
97: Something else

The table she created is as follows:
```

B SUPPORTING FIGURES

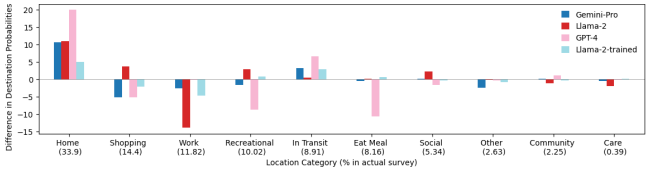

Figure B1: Difference in second-order destination probabilities (original - generated) for San Francisco-Oakland-Hayward metropolitan area .

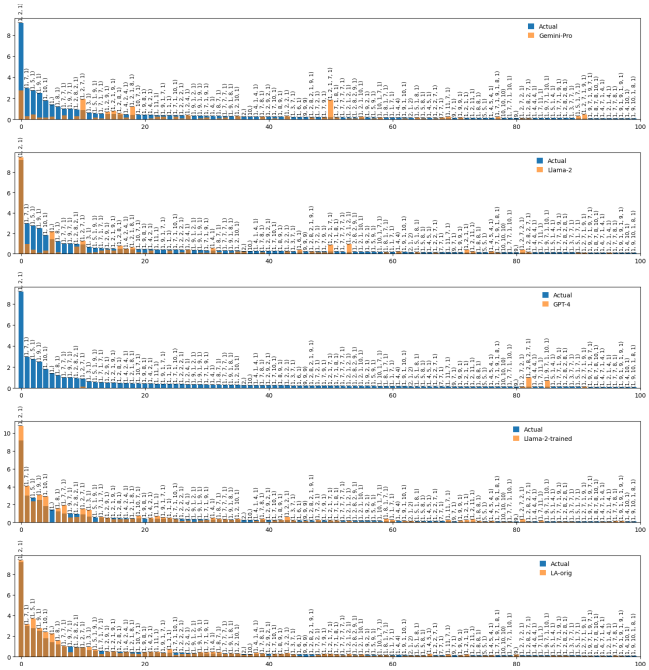

Figure B2: Distribution of top 100 activity chains of San Francisco-Oakland-Hayward metropolitan area in comparison to various LLM-based simulation techniques.

C SUPPORTING TABLES

Table C2: Example of second order transition probabilities.

x_{t-2}	x_{t-1}	x_t Home	x_t Work	x_t ...
Home	Work	0.15	0.03	...
Home	Community	0.05	0.04	...
...	...	...	...	...

Table C1: Categorization of location types done in the National Highway Travel Survey 2017.

ID	Label
1	Regular home activities (chores, sleep)
2	Work from home (paid)
3	Work
4	Work-related meeting/trip
5	Volunteer activities (not paid)
6	Drop off /pick up someone
7	Change type of transportation
8	Attend school as a student
9	Attend child care
10	Attend adult care
11	Buy goods (groceries, clothes, appliances, gas)
12	Buy services (dry cleaners, banking, service a car, etc)
13	Buy meals (go out for a meal, snack, carry-out)
14	Other general errands (post office, library)
15	Recreational activities (visit parks, movies, bars, movies, etc)
16	Exercise (go for a jog, walk, walk the dog, go to the gym, etc)
17	Visit friends or relatives
18	Health care visit (medical, dental, therapy)
19	Religious or other community activities
97	Something else

Table C3: Reclassification of the 20 location types in NHTS-2017 survey to 6 location types of patterns of life based simulation system.

NHTS-2017	Pattern of Life
Regular home activities	Home
Work from home	Home
Work	Work
Work-related meeting/trip	Work
Volunteer activities	Recreation
Drop off /pick up someone	Other
Change type of transportation	Other
Attend school as a student	School
Attend child care	Other
Attend adult care	Other
Buy goods	Other
Buy services	Other
Buy meals	Restaurant
Other general errands	Other
Recreational activities	Recreation
Exercise	Recreation
Visit friends or relatives	Other
Health care visit	Other
Religious or other community activities	Other
Something else	Other