uTECH-GenUrban: A Generative Agent-Based Framework for Urban Planning and Mobility Analytics

Anonymous authors
Paper under double-blind review

Abstract

Urban mobility systems exhibit complex, emergent behavior that is difficult to capture with traditional top-down demand models, which often rely on simplified behavioral assumptions and limited adaptability. We present uTECH-GenUrban, a hybrid generative-predictive framework that integrates large language model (LLM) reasoning with heterogeneous urban data to synthesize demographic cohorts, hourly activity schedules, and origin–destination (OD) flows. The framework consists of three sequential stages: `PopulationAgent`, which constructs interpretable demographic cohorts from American Community Survey (ACS) distributions; `ActivityAgent`, which generates hourly participation profiles across five broad activity domains derived from the American Time Use Survey (ATUS), namely sleep, work, meal, errand, and leisure; and `PlannerAgent`, which converts validated activity schedules into mobility-force signals and OD flow estimates using TomTom mobility records, Overture Maps building data, and a downstream Gradient Boosting Regressor. To improve reliability, each stage is wrapped in a hybrid verification pipeline combining deterministic checks, a self-correction step, and structured multi-agent debate. We evaluate uTECH-GenUrban across four heterogeneous U.S. urban testbeds—Las Vegas, Minneapolis, New York City, and San Francisco. The generated cohorts exhibit coherent and socially interpretable demographic differentiation. Aggregated evaluation against observed activity distributions demonstrates the strong capability of the mobility synthesis approach in reproducing real-world patterns. For mobility synthesis, uTECH-GenUrban generalizes well to unseen temporal windows, with an R^2 ranging from 0.80 to 0.98 across four cities. These results suggest that LLM-driven, verification-aware urban simulation offers a promising pathway toward digital twins for planning and urban analysis.

Keywords: Large Language Models, Multi-Agent Systems, Urban Mobility Modeling, Synthetic Population Generation, Activity Schedule Synthesis, Verification-Aware Generation

1 Introduction

Urban transportation systems are increasingly complex, requiring modeling approaches capable of capturing both large-scale mobility patterns and individual behavioral dynamics. Traditional spatial interaction models, such as gravity models, were among the earliest approaches used to estimate travel flows between locations by relating population size to distance-based travel costs. While these models can approximate aggregate commuting patterns, empirical studies using large-scale mobility data have shown that human travel behavior is often concentrated around a limited number of frequently visited locations and cannot be fully explained by simple distance–population relationships Csáji et al. (2013). Additionally, research has demonstrated that gravity models struggle to accurately reproduce fine-scale spatial–temporal mobility patterns, particularly when applied to high-resolution urban mobility networks. Hsu et al. (2021)

To support transportation planning, gravity-based concepts were later incorporated into the four-step travel demand forecasting framework, which sequentially models trip generation, trip distribution, mode choice, and traffic assignment to estimate travel flows across transportation networks Hensher (2008). Although the four-step model remains widely used in practice, it represents travel decisions at an aggregated zonal level and assumes that trip decisions occur independently within each modeling stage. These simplifying assumptions limit the model's ability to represent the behavioral complexity and interdependence of individual travel decisions. Building on these behavioral foundations, activity-based travel demand models were developed to derive travel demand from the daily activity patterns of individuals and households rather than modeling trips as independent events Davidson et al. (2007). These models represent travel as part of broader daily activity schedules and often employ tour-based structures and microsimulation techniques to capture interactions between activities, travel choices, and household characteristics.

More recent modeling frameworks further extend these ideas by integrating land use, behavioral modeling, and transportation system dynamics. UrbanSim, for example, models long-term urban development by simulating the location decisions of households, firms, and developers within urban land markets Waddell (2002). Similarly, agent-based transportation simulation platforms such as MATSim represent travelers as autonomous agents whose activity–travel plans evolve in response to network congestion, policy interventions, and interactions with other travelers (Adnan et al. (2021)).

Despite these methodological advances, many existing modeling approaches still rely on simplified behavioral assumptions, complex calibration procedures, or substantial computational resources. As urban systems become increasingly data-rich and dynamic, there is growing interest in developing modeling frameworks capable of capturing complex behavioral dynamics while remaining scalable for large metropolitan transportation systems.

This paper presents uTECH-GenUrban, a generative multi-agent framework that addresses these limitations by embedding large language model cognition directly into the urban mobility simulation pipeline. Instead of relying on hand-crafted behavioral rules or extensive manual calibration, uTECH-GenUrban leverages LLM-powered agents to synthesize demographic cohorts and hour-by-hour activity schedules, and to infer mobility-force signals that are then translated into OD flow estimates by a downstream regressor. A novel application of verification-guided prompt diversification and agent arrangement enables the framework to produce behaviorally realistic outputs across cities of varying density, such as our application of this framework across Manhattan, Minneapolis, San Francisco, and Las Vegas without city-specific modifications. OD matrices derived from real-time TomTom traffic data serve both as calibration targets and ground-truth benchmarks, anchoring the generative outputs in empirically observed spatiotemporal flow structures. By uniting generative AI with urban transportation modeling, uTECH-GenUrban offers a promising pathway toward responsive, explainable, and city-scale digital twins capable of informing next generation planning and resilience strategies.

uTECH-GenUrban advances urban mobility simulation along three fronts. First, we develop a prompt-based cohort modeling approach in which a PopulationAgent autonomously segments urban populations from ACS microdata and maps each cohort to hour-by-hour activity fractions across five behavioral domains. Second, we design a unified data pipeline that fuses ACS census distributions, Overture Maps building footprints, and TomTom real-time OD records to ground generative outputs observed in urban structure. Third, we

establish a hybrid mobility synthesis stage in which PlannerAgent infers behavioral movement forces and a downstream Gradient Boosting Regressor maps these signals, together with spatial features, to OD flow estimates, followed by aggregate validation against held-out TomTom data.

2 Related Work

2.1 LLM Powered Agent-Based Mobility Simulation

Prior to the integration of large language models, agent-based approaches to urban mobility simulation took one of two forms: hand authored behavioral rules encoded as finite state machines or utility functions, or data driven predictors trained on observed trajectories. Both produce agents that are, in a precise sense, memoryless. They carry no accumulated experience that would modify how a future situation is interpreted, and they cannot reflect on past outcomes to revise ongoing plans. A comprehensive survey of multi agent geosimulation methodologies spanning two decades of work formalizes the architectural components that a fully specified agent requires: perception, memory, planning, and action Padilla & D'avila (2025). The survey evaluates nine established development methodologies against this reference model. No methodology instantiates all four components. These gaps are consistent across all nine reviewed methodologies, suggesting they reflect something structural about the paradigm rather than fixable oversights in any individual framework.

The canonical demonstration of this paradigm shift is the Generative Agents framework of Park et al. (2023). Populated with 25 LLM driven agents in a sandbox environment, it established the foundational architecture underlying most subsequent work such as a memory stream with recency, importance, and relevance weighted retrieval; a reflection mechanism that synthesizes observations into higher level inferences; and a hierarchical planning module that decomposes daily intentions into moment to moment actions. A controlled ablation confirmed that each component is independently necessary. Removing reflection degraded synthesis and inference, removing planning produced temporally incoherent action sequences, and removing memory access altogether collapsed performance to near baseline levels. Park et al. (2023) also documented the principal failure modes shaping subsequent design choices such as retrieval failures that surface incomplete memory fragments, hallucinated embellishments blending factual recall with world knowledge contamination, and instruction tuning artifacts that make agents overly cooperative. Each has a direct counterpart in urban mobility simulation. A cohort generation agent that hallucinates demographic distributions or an activity scheduling agent that accepts any framing without push back, can produce outputs that appear plausible but fail empirical validation.

Building on this foundation, subsequent work has explored embedding LLM cognition specifically within mobility contexts. Gao et al. present a comprehensive survey on LLM enabled agent based simulations, highlighting their particular promise for transportation domains Gao et al. (2024) . Xu et al. propose a GenAI powered multi agent paradigm for smart urban mobility incorporating retrieval augmented generation to improve context awareness and decision making Xu et al. (2024). Wang et al. introduce LLMob, which generates realistic personal activity trajectories through semantic alignment Wang et al. (2024). Noyman et al. develop TravelAgent for simulating pedestrian way finding in built environments Noyman et al. (2026), and complementary work from Chalmers University integrates mobile data into synthetic multi day travel plans for sustainable transportation analysis Çaglar Tozluoglu (2025).

More recent frameworks have pushed toward greater cognitive richness and scale. CitySim endows agents with recursive value driven daily planning, explicit need tracking, belief aware spatial memory updated via Kalman filtering, and long term goal formation grounded in Maslow's hierarchy, validating against Japanese national time use data at populations reaching one million agents Bougie & Watanabe (2025). Calderon et al. formalize the cognitive agent loop within a validated urban mobility simulator, demonstrating through controlled ablation that structured memory is not merely a design preference but an empirically grounded engineering requirement Calderón et al. (2025). Their ablation study shows that short term episodic memory alone reduces late arrivals by nearly 11 percentage points under severe disruption and adding long term semantic abstraction further lowers tardiness to below 15 percent, while maintaining a trip completion rate above 98 percent. Calderon et al. also document limitations that persist in even well designed LLM frameworks: behavioral adaptation remains reactive rather than anticipatory, scalability is constrained by

per agent LLM invocation costs, and individual decision attribution remains partially opaque. GATSim addresses the scalability concern through hierarchical memory and reflective planning Liu et al. (2026), while CAMS encodes geospatial awareness directly into the simulation layer to improve spatial consistency Du et al. (2025).

A final line of work justifies uTECH-GenUrban's use of structured, role differential prompting. Chain-of-thought-prompting elicits stronger multi-step reasoning by making intermediate inference steps explicit . This directly motivates uTECH-GenUrban's ConversableAgent architecture, in which each stage reasons step by step through demographic interpretation, activity assignment, and mobility force inference before outputs are validated against hard constraints. Collectively, these studies define the intellectual context within which uTECH-GenUrban operates. The framework extends cognitively grounded, memory structured LLM agents beyond small-scale sandbox settings. The goal is aggregated urban mobility that transfers across heterogeneous cities without city specific calibration.

2.2 Demographic Modeling and Synthetic Population Generation

Empirical and computational evidence consistently supports demographic segmentation as a foundation for mobility modeling. A clustering-based approach by Yang et al. demonstrates that principal component analysis combined with K-means can effectively identify representative traveler types, showing strong alignment between sociodemographic segments and temporal mobility choices Yang et al. (2023). Cho & Kim further demonstrate that clustering residents by spatiotemporal mobility patterns before embedding significantly improves destination prediction accuracy by over 14 percentage points compared to random grouping, confirming that shared behavioral characteristics create more discriminative mobility representations Cho & Kim (2021). Together, these studies point to a consistent finding. Emergent demographic segmentation produces more behaviorally coherent representations. The finding is robust across methods. Sociodemographic features, spatiotemporal patterns, and agent based grouping all yield more discriminative mobility representations than ungrouped approaches. uTECH-GenUrban's unsupervised cohorting strategy builds directly on this evidence.

Generating realistic synthetic populations from census microdata is the operational counterpart to this segmentation rationale. Jiang et al. (2024) use hierarchical sampling on 2020 U.S. Census data to produce a geographically explicit synthetic population across all 50 states, reproducing demographic distributions but omitting transportation behavior and fine scale mobility dynamics. Jutras-Dubé et al. (2024) advance this by combining copula theory with generative models to enable transferability across geographic scales using ACS data, though their statistical approach relies on fixed dependency structures and cannot adapt to novel urban conditions. Lim et al. (2026) take it a step further by fine tuning a lightweight open source LLM on household travel survey data, using Bayesian Network driven topological order to guide the auto regressive generation process. Their approach achieves approximately 95 percent feasibility, significantly outperforming VAEs and GANs while maintaining comparable diversity by encoding semantic relationships between sociodemographic attributes. However, this approach remains limited to individual level tabular synthesis and does not extend to activity scheduling or mobility flows.

Connecting population attributes to daily behavior, Shi & Goulias (2024) use the ATUS to show that sociodemographic characteristics reliably shape hour by hour activity participation across canonical schedule types such as work days, mixed days, and leisure days. This validates demographic segmentation as an organizing principle for activity generation. Similarly, the OpenCity framework groups large numbers of LLM driven agents by static sociodemographic features to ensure scalability and behavioral coherence while letting the model learn patterns endogenously Yan et al. (2024). uTECH-GenUrban builds directly on these foundations. Instead of requiring fine tuning or fixed statistical structures, it uses prompt based LLM agents operating on ACS micro data to autonomously form demographic cohorts, map them to hourly activity fractions, and infer behavioral mobility signals used for OD flow prediction. This extends the synthetic population pipeline to a full mobility simulation framework that generalizes temporally and spatially across urban typologies.

3 METHODS

3.1 System Design

The first two stages are LLM-driven structured generation modules, while the final mobility stage combines PlannerAgent-derived behavioral signals with a downstream Gradient Boosting Regressor. The system consists of three sequential stages, each responsible for transforming inputs into progressively more structured representations of urban behavior.

As shown in Figure 1, our aggregated pipeline features three sequential stages, each executed by a specialized LLM-powered "manager" agent. The `PopulationAgent` ingests region-level ACS demographic data and partitions the population into a set of archetypal cohorts, each defined by a cohort ID, total count, and a compact summary of attribute distributions such as age, income, educational attainment, and related characteristics. The `ActivityAgent` takes each cohort and a fixed list of activity types (sleep, work, meal, errand, leisure). It divides the 24-hour day into hourly bins and, for each bin and activity, estimates the fraction of the cohort engaged. These fractions sum to one across all activities in each hour. Finally, the `PlannerAgent` transforms the hour-by-hour activity fractions into Mobility Forces, while the Gradient Boosting Regressor Module simulates real-world OD flows using TomTom records.

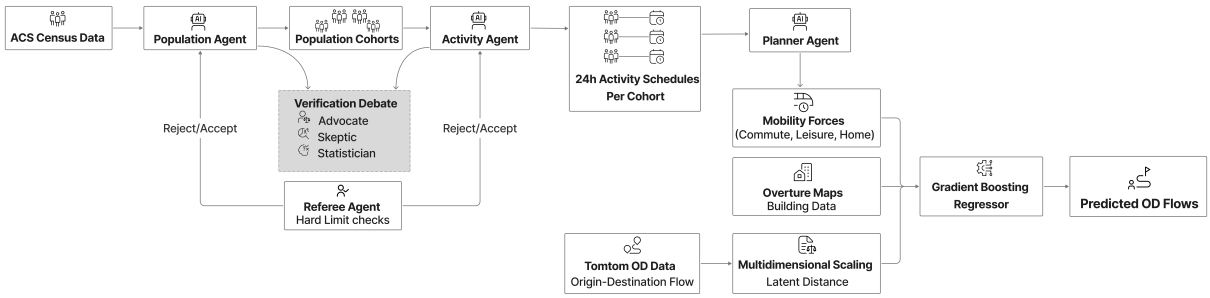


Figure 1: The system design and data flow of the uTECH-GenUrban framework

3.1.1 PopulationAgent and Population Generation Stage

The population generation stage begins with the `PopulationAgent`, which retrieves detailed demographic statistics for a target region using a custom utility function built on top of the ACS API. The returned data include a structured distribution of key population characteristics, such as age, gender, marital status, educational attainment, employment status, household composition, vehicle ownership, and income. Based on these demographic inputs, the `PopulationAgent` constructs a cohorting schema that groups residents into interpretable demographic clusters with distinct activity and mobility attributes. Each cohort is represented as a structured profile and passed to the next stage as the basis for downstream behavioral modeling.

3.1.2 ActivityAgent and Activity Generation Stage

Given the demographic cohorts generated by the `PopulationAgent`, the `ActivityAgent` produces cohort-specific lifestyle narratives and hourly activity schedules. Formally, for each city $c \in \mathcal{C}$, demographic cohort $g \in \mathcal{G}$, hour $t \in \mathcal{T}$, and activity type $a \in \mathcal{A}$, the activity module estimates the probability that cohort g performs activity a at time t:

$$p_{g,a}^{(c)}(t) = \Pr(A_t = a \mid g, c)\,, \tag{1}$$

where A_t denotes the activity state at hour t. In this work, the activity set is defined as

$$\mathcal{A} = \{\text{sleep}, \text{work}, \text{leisure}, \text{errand}, \text{meal}\}. \tag{2}$$

For each cohort g and hour t, the activity probabilities form a valid distribution:

$$\sum_{a \in \mathcal{A}} p_{g,a}^{(c)}(t) = 1, \tag{3}$$

with

$$0 \leq p_{g,a}^{(c)}(t) \leq 1, \qquad \forall a \in \mathcal{A}. \tag{4}$$

These cohort-level activity distributions provide a temporally resolved representation of behavioral patterns. To summarize the dominant behavioral state of each cohort at a given time, we define the dominant activity as

$$a_g^*(t) = \arg\max_{a \in \mathcal{A}} p_{g,a}^{(c)}(t), \tag{5}$$

with corresponding dominance score

$$s_g^*(t) = \max_{a \in \mathcal{A}} p_{g,a}^{(c)}(t). \tag{6}$$

The resulting city-time cohort summary is represented as

$$S^{(c)}(t) = \left\{ \left(g, a_g^*(t), s_g^*(t) \right) \right\}_{g \in \mathcal{G}}, \tag{7}$$

which serves as the structured behavioral input to the mobility planning stage.

3.1.3 PlannerAgent and Mobility Flow Prediction Stage

The mobility flow prediction stage begins with `PlannerAgent`, which takes the hourly validated activity schedules at the cohort-level produced by the activity generation stage. `PlannerAgent` acts as a cognitive reasoning module, inferring first a set of high-level behavioral mobility signals that reflect the dominant temporal movement tendencies under a given city, hour, and day type and produces a macro-level mobility force vector:

$$\vec{\Phi}(t) = \mathrm{LLM}_{\mathrm{Planner}} \left(\bigcup_{k=1}^{K} P(S \mid c_k, t), \ \text{context} \right) = \begin{bmatrix} \phi_{\mathrm{commute}}(t) \\ \phi_{\mathrm{leisure}}(t) \\ \phi_{\mathrm{home}}(t) \end{bmatrix} \tag{8}$$

Where $\mathcal{C} = \{c_1, \ldots, c_K\}$ represents the socio-demographic cohorts generated by the population agent, S denotes the activity schedules {sleep, work, meal, errand, leisure}, and the output vector components correspond to three movement forces: commute, leisure, and home, respectively. These vectors represent the forces at a particular instance of time; for example, at 10:00 AM, the commute force is higher than the home force as most people are commuting for work, while in the evening, the home force might be higher. This is essentially the "pull" toward a particular zone at a specific time. These vectors are scaled between 0 and 1. The planner agent analyzes the activity schedules to identify the deltas between time steps. For instance, if at 7:00 AM "Sleep" is 90% and at 8:00 AM "Sleep" is 10% while "Work" is 80%, the agent recognizes a massive transition. It then assigns a high value to $\phi_{\mathrm{commute}}(8)$ because the sociological intent has shifted from "Home/Rest" to "Productivity/Work."

To ground these behavioral intents in physical reality, the framework incorporates structured spatial and historical infrastructural constraints. From historical mobility records, we aggregate the total flow F_{ij} between origin i and destination j, and define the global maximum flow $F_{\max}$. We construct a flow-induced dissimilarity matrix $D_{i,j} = 1 - (F_{ij}/F_{\max})$, where larger historical flow implies smaller functional dissimilarity. This matrix is embedded into a latent spatial topology using multidimensional scaling (MDS) to assign a latent coordinate $\mathbf{z}_i \in \mathbb{R}^2$ to each zone:

$$\min_{\mathbf{z}_1, \ldots, \mathbf{z}_N} \sum_{i<j} \left(\|\mathbf{z}_i - \mathbf{z}_j\| - D_{i,j} \right)^2 \tag{9}$$

In parallel, we derive the activity of a physical destination using Overture Maps building footprints, defining B_i and B_j as the total capacities of commercial and residential buildings for the given origin and destination zones.

The final step fuses sociological intent with the learned physical topology to predict continuous OD flows. A Gradient Boosting Regressor (GBR) is employed to capture the complex, non-linear interactions across these highly heterogeneous features without over-fitting to regional sparsity. The final predicted mobility flow $\hat{y}_{i,j}^{(t)}$ from region i to j at time t is formulated as:

$$\hat{y}_{i,j}^{(t)} = \mathcal{M}_{\text{GBR}}\Big(\|\mathbf{z}_i - \mathbf{z}_j\|,\ B_i,\ B_j,\ \vec{\Phi}(t),\ t\Big) \tag{10}$$

To ensure physically meaningful nonnegative estimates, the prediction is clipped via $\hat{y}_{i,j}^{+} = \max(\hat{y}_{i,j}^{(t)}, 0)$. All continuous features were standardized to zero-mean and unit-variance prior to model fitting.

Accordingly, the final predicted mobility flow is jointly determined by latent functional separation, built-environment attractiveness, temporal context, and the autonomous planner's inferred behavioral movement forces.

3.2 Data Sources

Our framework relies on three core data streams for input: demographics distribution, building footprints, and real-time OD pairs over regions. These data streams enable a multi-scale and multi-modal view of urban activity and infrastructure, essential for generating behaviorally and spatially realistic simulations. We retrieve age, gender, marital status, educational attainment, employment status, household structure, vehicle ownership status, and income for any U.S. region via the American Community Survey (ACS) API U.S. Census Bureau (2025e). We use the `overturemaps` Python package Overture Maps Foundation (2025)Overture Maps (2025) to load building geometries, categories (e.g., residential, commercial, education), and geocoordinates.

The uTECH-GenUrban framework operates on a hexagonally partitioned representation of urban space Brodsky (2018), using Uber H3 resolution 7, where each hexagon covers approximately 1 square kilometer. This resolution balances computational tractability with spatial precision, allowing the simulation to effectively capture neighborhood-level activity and travel patterns. Real-time hourly OD records are then pulled between these H3 hexagonal regions through the TomTom Mobility API TomTom International BV (2025). The API reports trip volumes between geographic zones based on aggregated mobile device and vehicle telemetry data, reflecting both commuter and non-commuter patterns. We use these OD matrices both to calibrate the synthetic mobility generation process and, on disjoint held-out temporal segments, to evaluate its spatial and temporal fidelity. In our study, the TomTom data are available only for August, and the O/D analysis is further constrained by TomTom's documented maximum of 200 regions per analysis, which restricts the achievable spatial resolution of the OD data. Within each hexagonal unit, the system synthesizes population cohorts, assigns hourly activity profiles, and predicts origin-destination (OD) flows based on real-time and contextual factors.

4 Experiment and Results

To evaluate the performance and realism of the uTECH-GenUrban framework, we conducted a comprehensive simulation workflow across four urban testbeds in the United States: Las Vegas (represented by Clark County, Nevada), Minneapolis (represented by Hennepin County, Minnesota), New York City (represented by New York County/Manhattan, New York), and San Francisco (represented by San Francisco County, California). We selected these testbeds to capture substantial variation in urban form, demographic composition, housing conditions, and economic structure, thereby providing a more representative basis for evaluating the generalizability of the framework across heterogeneous U.S. cities. The four study areas span markedly different population scales and built-environment intensities, ranging from the extremely high-density core of Manhattan to the much lower-density and spatially expansive Clark County, while also differing in household structure, linguistic diversity, educational attainment, and housing markets U.S. Census Bureau (2025c;d;a;b).

These testbeds also reflect distinct urban-economic regimes that are important for mobility generation and activity simulation. Clark County captures a tourism- and hospitality-oriented metropolitan system centered

on Las Vegas Clark County, Nevada (2025); San Francisco represents a dense coastal innovation hub with strong technology, professional services, and visitor-economy sectors City and County of San Francisco (2025); New York County provides a globally connected, transit-intensive urban core with exceptionally high residential and employment density U.S. Census Bureau (2025c); and Hennepin County represents a large, diversified Upper Midwest metropolitan economy anchored by Minneapolis–Saint Paul Hennepin County (2023). Collectively, these four cases allow us to assess whether uTECH-GenUrban can remain robust under widely varying demographic, spatial, and socioeconomic conditions rather than being tuned to a single urban archetype.

4.1 Experimental Setup

The simulation environment was implemented in Python as a modular pipeline capable of interacting with large language models, external APIs, and geospatial data services. Each component of the system operates as an LLM-powered agent with structured prompting, validation logic, and a dedicated communication interface, enabling coherent data exchange and more behaviorally realistic simulation outputs.

All agents in our framework are implemented as `ConversableAgent` instances within the AutoGen framework. Instead of relying on a single backend model, we now evaluate the system using four foundation models accessed through LiteLLM: `openai.gpt-5-mini`, `meta.llama-4-maverick-17b-instruct`, `google.gemini-2.5-flash`, and `xai.grok-3-mini`. Each agent is driven by a dedicated `ChatSession` with an attached `MemoryBuffer` to preserve interaction history and maintain contextual consistency across tasks. In addition, we implement a customized `UserProxy` layer for each session to intercept manager requests, execute functional calls, and validate outputs to ensure that all intermediate and final results follow the required structure.

To compare synthetic daily schedules with observed time-use data, we aggregate the highly granular American Time Use Survey (ATUS) activity lexicon into five broad activity domains: *sleep*, *work*, *meal*, *errand*, and *leisure*. This aggregation is a deliberate modeling choice. First, the official ATUS coding system is substantially more detailed than what is needed for our task: it uses a three-tier hierarchy with 17 first-tier categories and many finer lower-tier subcategories U.S. Bureau of Labor Statistics (2025b; 2019). While this level of detail is valuable for survey analysis, it also creates a much larger and sparser output space for synthetic schedule generation. In our setting, the goal is not to reproduce the full semantic richness of ATUS, but to recover the minimum set of activity domains needed to represent the broad structure of a person's day: physiological maintenance, productive obligations, eating, maintenance/service tasks, and discretionary time.

A second motivation is methodological. Our daily schedules are generated in a structured format by LLMs, and recent work shows that structured generation under schema or format constraints remains a nontrivial reliability bottleneck for LLMs Lu et al. (2025); Geng et al. (2025). Using a coarser activity space therefore improves tractability, reduces sparsity, and yields more stable generation and comparison. In other words, the five-category abstraction is not only behaviorally interpretable but also better aligned with the current practical limits of LLM-based synthetic-data generation.

Following the official ATUS definitions, we map working and education activities to our work category so that the same domain covers both labor-force participation and student schedules U.S. Bureau of Labor Statistics (2023). We map eating and drinking to meal. We map purchasing goods and services, household services, government-service-related activities, and household-management or chore-like activities to errand, consistent with ATUS descriptions of purchasing, household maintenance, and service acquisition U.S. Bureau of Labor Statistics (2025a; 2021). We anchor sleep in the ATUS personal care category, which explicitly includes sleeping and related self-care U.S. Bureau of Labor Statistics (2023). Because our objective is to model broad daily routine structure rather than preserve every fine-grained semantic boundary, some necessary home-bound caregiving or maintenance time may also be absorbed into this minimally defined restorative category in the coarsened representation. Finally, we group socializing, relaxing, and leisure, sports, religious activities, and volunteer activities into leisure U.S. Bureau of Labor Statistics (2025a). This aggregation preserves the main behavioral rhythm of daily life while keeping the representation compact enough for robust synthetic schedule generation and cross-city comparison.

4.2 Hybrid Verification Pipeline: Self-Check + Multi-Agent Debate

To reduce prompt variability, hallucinations, and run-to-run instability, we wrap Population and Activity stages in a hybrid verification loop. Specifically, for every stage in the sequence (*Population* → *Activity*), the stage agent first produces a structured JSON output rather than free-form text. This design enables deterministic, programmatic evaluation and supports consistent downstream interfaces.

Immediately after generation, we execute a suite of deterministic validators, including (i) schema checks to ensure the output matches an expected JSON structure; (ii) constraint checks such as conservation rules and range bounds; and (iii) distribution-fit diagnostics (e.g., KL/JS divergence and RMSE) comparing synthesized outputs against reference marginals or aggregate targets. These checks produce a machine-readable `check_report` that summarizes violations, metric values, and pass/fail status.

Next, a Self-Check Verifier performs a CoVe-style verification using only the candidate JSON and the `check_report`. The verifier generates targeted verification questions and proposes a concrete patch plan to address detected issues. The stage agent then applies the patch plan to produce a revised JSON output, which is re-evaluated by the deterministic checks. This step reduces the risk of self-confirmation by grounding the verification on explicit constraints and measurable metrics.

Finally, we run a structured multi-agent debate consisting of three roles: an Advocate that justifies the candidate output, a Skeptic that stress-tests for inconsistencies and potential hallucinations, and a Statistician that focuses exclusively on quantitative metrics from the hard checks. A Referee agent then makes the final decision, accepting only candidates that satisfy hard constraints and achieve the best metric profile among available candidates. If the Referee rejects the output, the stage is rerun under a retry policy that may vary prompt templates, random seeds, or model choices.

In our implementation, the hybrid verification wrapper is inserted immediately after each stage's `initiate_chat` call (Population/Activity) and before passing results to the next stage. We replace ad-hoc parsing of free-form text with a stable JSON contract, enabling deterministic validation and systematic acceptance/rejection. This converts reliability from an implicit expectation into a measurable system property, which can be quantified via repeated runs across multiple prompts and seeds.

4.3 Population cohorts

As illustrated in Figures 2, the PopulationAgent autonomously generated a set of archetypal cohorts, each endowed with a distinct identity and demographic profile. As discussed in the previous sections, no explicit instruction was given regarding either the exact number of cohorts or the criteria by which they should be differentiated; instead, both the cohort structure and their defining attributes were inferred by the AI agent through its internal reasoning process. Figure 2 provides a representative example for New York City under `google.gemini-2.5-flash`. Across these visualizations, the generated cohorts are clearly not arbitrary partitions, but coherent demographic groupings with interpretable contrasts in age composition, education, income, gender balance, and employment status. Although the cohort names, relative sizes, and degree of separation vary across cities and model backends, the PopulationAgent consistently produces differentiated population segments that reflect meaningful demographic heterogeneity.

We can observe the distinct age distributions across cohorts, which underscore the central role of age as a demographic variable, exhibiting strong associations with most other demographic dimensions Bialowolski et al. (2025). Furthermore, the cohorts reveal a positive correlation among household income, employment status, and levels of educational attainment, aligning with empirical findings that higher education and income facilitate automobile access, which in turn promotes employment, and that greater educational attainment corresponds with increased earnings Baum (2009); National Center for Education Statistics (2024). Overall, these patterns suggest that the PopulationAgent is able to recover plausible cross-demographic relationships and organize the synthetic population into socially interpretable archetypes across diverse urban contexts and model backends.

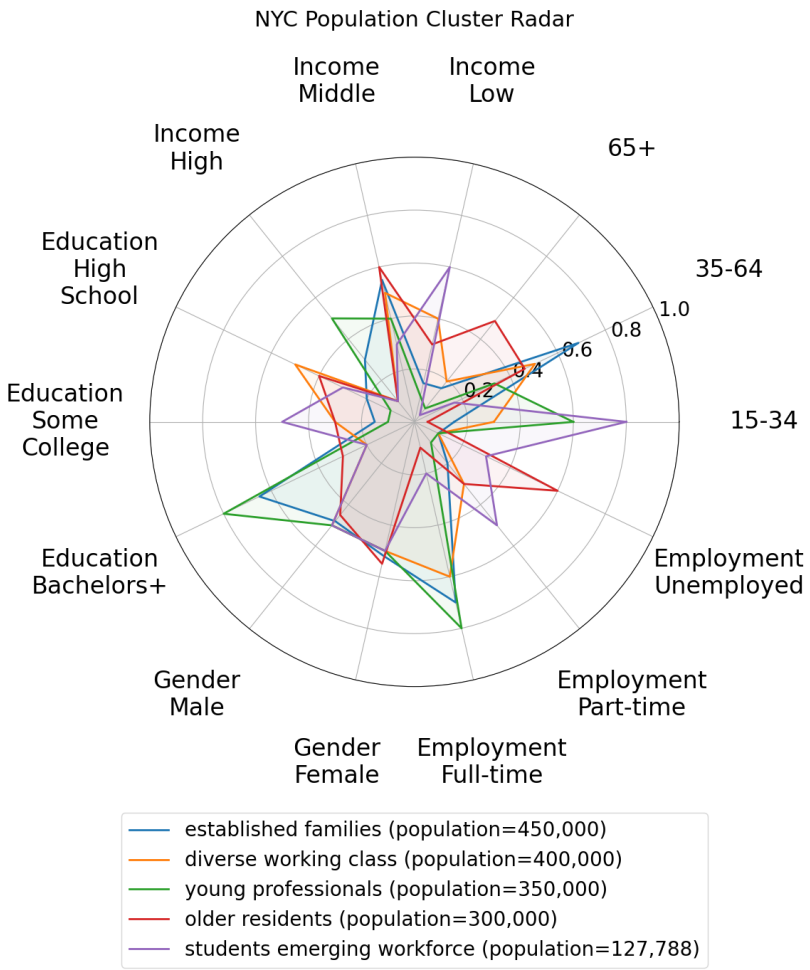


Figure 2: Representative cohort demographic radar plot for New York City generated by `google.gemini-2.5-flash`. The figure illustrates how the PopulationAgent separates the synthetic population into interpretable archetypes with distinct demographic profiles.

4.4 Activity Patterns and Scheduling

As illustrated in Figure 3, the generated activity schedules exhibit empirically plausible patterns. Sleep periods are predominantly allocated to nighttime hours, while work activities occur chiefly during the daytime. Meal times cohort around 06:00–08:00, 11:00–13:00, and 17:00–20:00, and leisure activities peak in the evening.

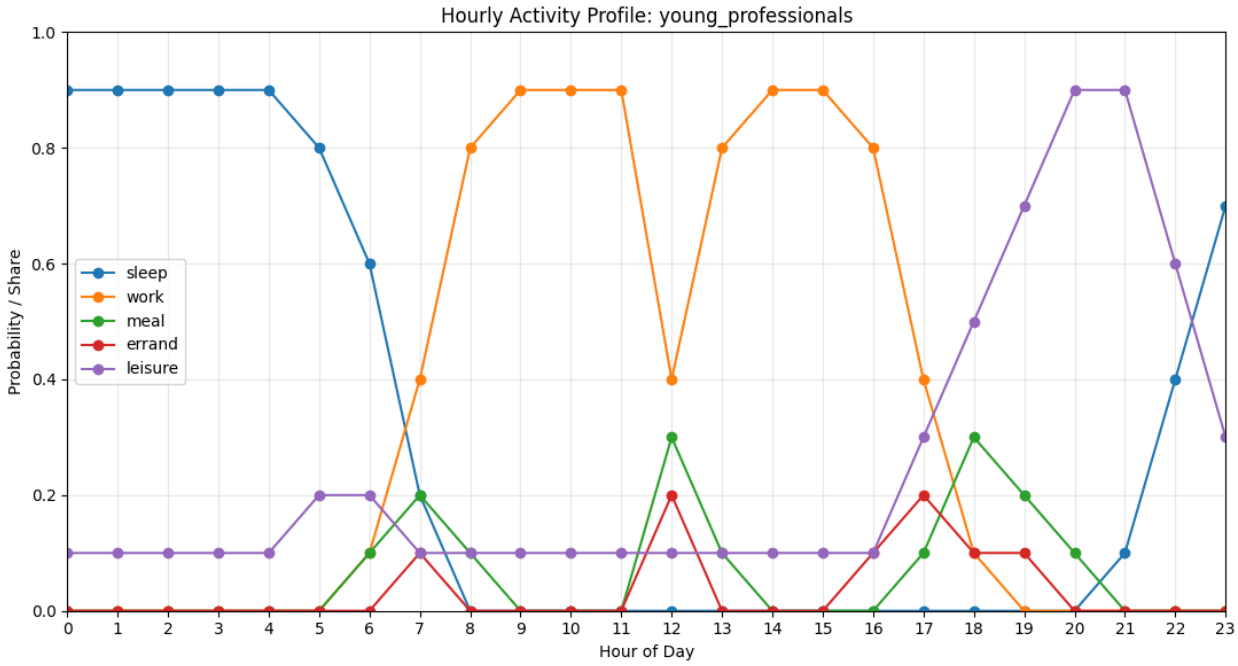


Figure 3: Representative cohort Activity distribution for New York City generated by google.gemini-2.5-flash. The figure illustrates how the activity schedules vary alongside hours.

Figures 4 and 5 provide a representative comparison between generated and observed activity patterns for New York City using `google.gemini-2.5-flash`. These two figures should be interpreted as complementary views of the same validation problem. Figure 4 reports the distribution of daily time across the five activity types, that is, the percentage of the 24-hour day allocated to each activity within each age group. By contrast, Figure 5 reports mean activity duration among participants only, excluding non-participants, so the activity durations in each age group do not necessarily sum to 24 hours. Together, these measures capture both the relative composition of the day and the intensity of engagement in each activity.

At a broad level, the generated schedules recover the main structure of the observed daily allocation across all age groups: sleep remains the largest component of the day, work is concentrated among younger and middle-aged groups, and leisure becomes relatively more prominent for older adults. This indicates that the model is able to preserve the overall ordering and relative dominance of major activity types. At the same time, both figures reveal several systematic deviations. In proportional terms, the generated schedules tend to under-allocate sleep across all age groups while slightly over-allocating errand and leisure, especially for the older population. In duration terms, the same pattern appears as fewer generated minutes devoted to sleep and somewhat more time allocated to leisure and errand activities. These discrepancies are most pronounced for the 65+ group, where the model shifts a noticeable portion of time away from sleep and toward non-work daytime activities. Meal remains the closest-matched category in both figures, with generated proportions and durations staying near the observed benchmark across all age groups, while work is reproduced more accurately for the youngest and oldest groups than for the middle-aged working population. Overall, the joint evidence from Figures 4 and 5 suggests that the models more easily reproduce

highly regular, temporally anchored activities such as meals, but face greater difficulty with broader and more behaviorally heterogeneous categories such as leisure and work.

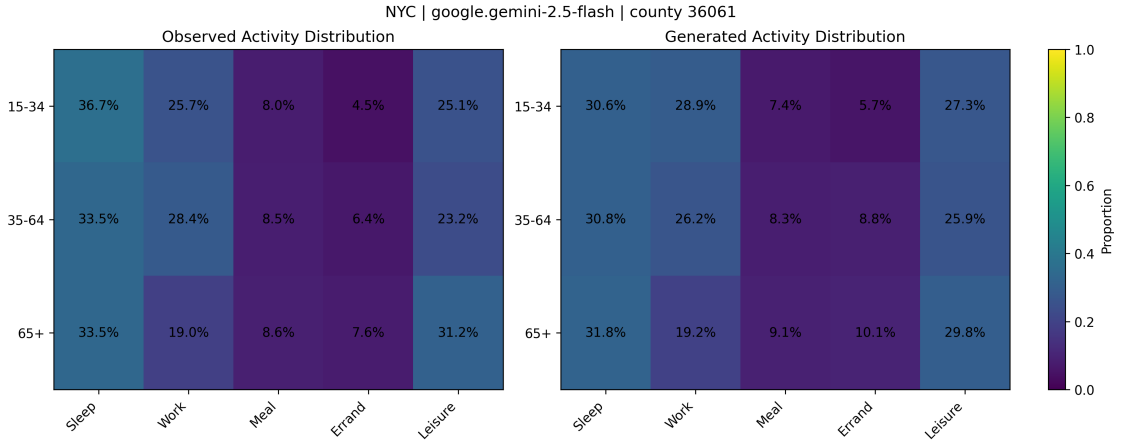


Figure 4: Representative activity distribution comparison between the generated activity data and observed data

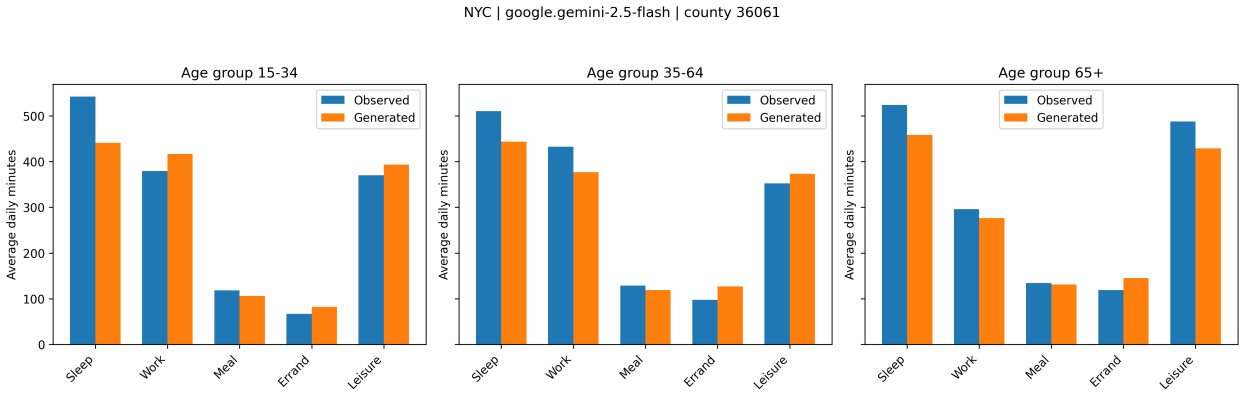


Figure 5: Representative activity duration between the generated activity schedule and observed data

Table 1: Summary statistics of evaluation metrics by model across all cities. Lower values indicate better performance. RMSE is computed on normalized activity-share distributions and is therefore reported in proportion units (0–1 scale) of one day.

	RMSE		KL		JS	
Model	Mean	Std	Mean	Std	Mean	Std
google.gemini-2.5-flash	0.048	0.024	0.035	0.035	0.008	0.008
meta.llama-4-maverick-17b-instruct	0.040	0.010	0.019	0.010	0.005	0.003
openai.gpt-5-mini	0.051	0.013	0.027	0.012	0.007	0.003
xai.grok-3-mini	0.042	0.012	0.029	0.017	0.007	0.004

These qualitative patterns are reflected in the aggregate model-level evaluation results in Table 1. Among the four tested models, `meta.llama-4-maverick-17b-instruct` achieves the best overall performance, with the lowest mean RMSE (0.0400), KL divergence (0.0193), and JS divergence (0.00485). `xai.grok-3-mini` follows closely, particularly on RMSE and JS, while `openai.gpt-5-mini` obtains intermediate results. `google.gemini-2.5-flash` shows the highest average errors and the largest variability across cases, as

Table 2: Summary statistics of evaluation metrics by city across all models. Lower values indicate better performance.

	RMSE		KL		JS	
City	Mean	Std	Mean	Std	Mean	Std
Las Vegas	0.040	0.007	0.023	0.011	0.005	0.002
Minneapolis	0.054	0.018	0.038	0.032	0.009	0.007
NYC	0.040	0.014	0.022	0.018	0.005	0.004
San Francisco	0.048	0.018	0.027	0.015	0.007	0.004

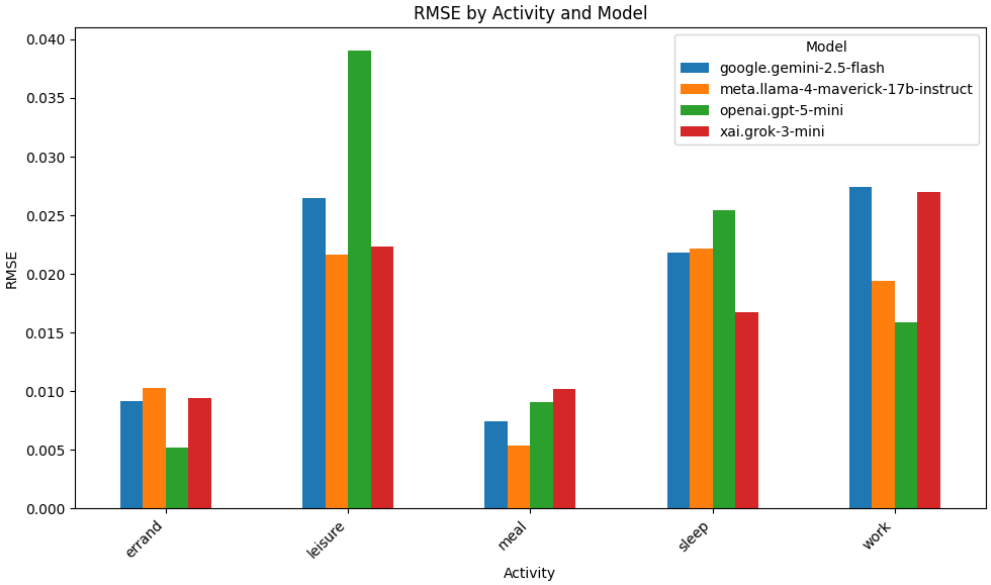


Figure 6: RMSE by activity category and model, aggregated across all four study cities. The results indicate that meal and errand are reproduced with the lowest absolute error across most models, while leisure and work exhibit comparatively higher RMSE, suggesting greater difficulty in matching their observed distributions. Among the tested models, `meta.llama-4-maverick-17b-instruct` and `openai.gpt-5-mini` generally achieve lower error for several activity types, although model performance varies by category.

indicated by its higher standard deviations, suggesting that its activity generation is less stable across cities. Overall, however, the differences across models are moderate rather than extreme, implying that all four LLMs are capable of generating broadly plausible schedules, but that the best-performing models more consistently align with observed activity distributions.

City-level results in Table 2 show that performance also varies across urban contexts. New York City and Las Vegas are the easiest settings overall, with the lowest mean RMSE and JS values, while Minneapolis is the most difficult city, exhibiting the highest mean RMSE (0.0540), KL (0.0380), and JS (0.00889). San Francisco falls between these extremes. This variation suggests that the proposed framework is sensitive not only to model choice but also to the underlying behavioral complexity of the city being represented. Cities with more heterogeneous work, leisure, and maintenance patterns may be harder to reproduce under a compact five-activity abstraction, even when the generated schedules remain qualitatively plausible.

A more detailed activity-specific view is provided by Figure 6 and Table 3. Across all city–model combinations, *meal* and *errand* are the most accurately reproduced categories, with mean RMSE values of 0.0080 and 0.0085, respectively, and correspondingly small KL and JS divergences. This is consistent with the intuition that these activities are either temporally regular (meals) or relatively low-share categories whose aggregate proportions are easier to match. In contrast, *leisure* and *work* are the most challenging categories. Leisure has the highest mean RMSE (0.0274), while work has by far the largest KL and JS values, indicating that

Table 3: Summary statistics of evaluation metrics by activity across all city–model combinations. Lower values indicate better performance.

	RMSE		KL		JS	
Activity	Mean	Std	Mean	Std	Mean	Std
errand	0.009	0.007	0.004	0.006	0.001	0.001
leisure	0.027	0.013	0.009	0.006	0.002	0.001
meal	0.008	0.004	0.003	0.005	0.001	0.001
sleep	0.022	0.010	0.004	0.005	0.001	0.001
work	0.022	0.012	0.022	0.031	0.005	0.007

its age-specific distribution is especially difficult to match. Figure 6 further shows that this pattern is robust across models: although the absolute error differs somewhat by LLM, all models find leisure and work more difficult than meal and errand. This likely reflects the fact that work and leisure are more heterogeneous across cohorts and cities, and are more sensitive to demographic composition and social context than regular maintenance activities.

4.5 Mobility Synthesis

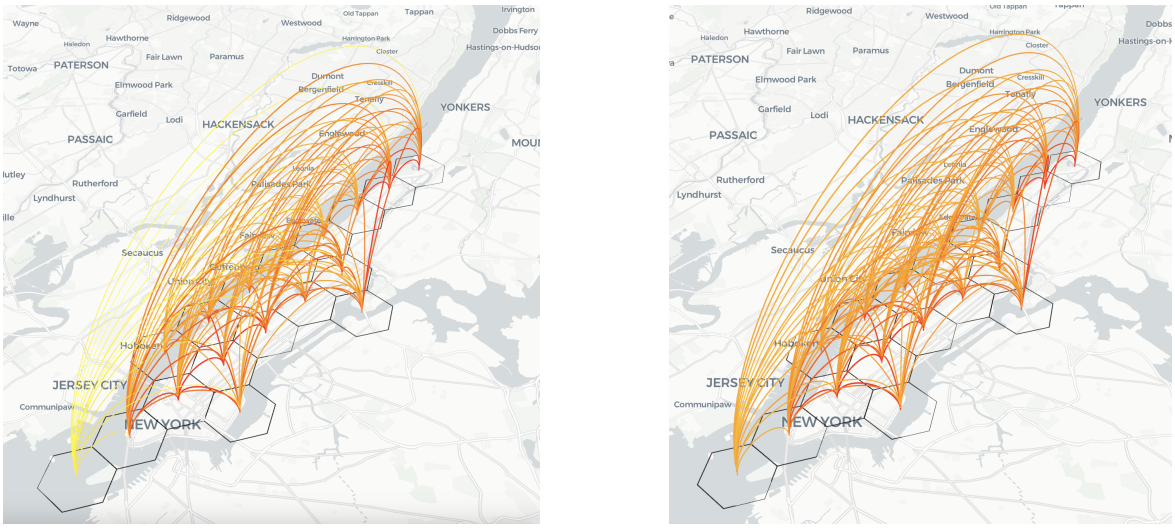
Although baseline LLM agents often struggle with the combinatorial complexity of urban movement—often failing to maintain spatial consistency or refusing tasks as the number of regions increases (Calderón et al., 2025)—uTECH-GenUrban addresses these challenges through its structured pipeline. To assess how well the framework generalizes to unseen conditions, the TomTom OD flow data for each city was split along the time axis using August as the study month. Four consecutive date windows August 6–9, August 10–13, August 14–17, and August 18–21 make up the training set. These windows together cover a broad slice of mid-month urban activity, giving the model enough exposure to the recurring spatial patterns and volume fluctuations that characterize each city's movement behavior. The final window, August 22–25, was withheld completely and used as the blind test set. No information from these dates entered the training pipeline in any form.

The choice of a forward-looking temporal split rather than a random holdout was deliberate. Urban mobility has a strong temporal structure patterns observed on one set of dates carry predictive signal for nearby future dates, but the magnitude and spatial distribution of flows can shift meaningfully across weeks due to behavioral drift, events, and day-of-week composition. Holding out the last date window therefore tests whether the framework captures something genuine about how a city moves, rather than simply memorizing the training period.

This evaluation setup was applied identically across all four cities New York City, Las Vegas, San Francisco, and Minneapolis. For NYC, the model achieved an R^2 of 0.9775 on the August 22–25 blind window, reflecting close agreement between the synthesized and ground-truth OD flows on dates the model had never seen. As illustrated in Figure 7, the predicted OD flows for Manhattan closely mirror the actual movement corridors, with the dominant north-south transit arteries and inter-borough connections reproduced with high spatial fidelity across the blind test period. Beyond R^2, performance was also measured using CPC (Common Part of Commuters), RMSE, and MAE, which together capture both the spatial distribution accuracy and the raw volumetric alignment of predicted flows against reality. Table 4 presents the full benchmark results across all four cities and all evaluated models, showing that strong generalization to unseen dates holds consistently and is not isolated to the NYC case.

The diagnostic results reveal that the GenUrban framework performs optimally in continuous, dense urban grids. In New York City and Minneapolis, ghost flows accounted for merely 3.3% and 0.0% of all routes, respectively. Correspondingly, their Normalized Root Mean Square Error (NRMSE) remained exceptionally low across both high-volume arteries (0.13 to 0.29) and sparse routes (0.46 to 0.81).

To address the performance variance observed across different cities specifically the degradation of R^2 in San Francisco (0.80) and Las Vegas (0.89) compared to New York City (0.97), we conducted a rigorous error decomposition analyzing flow sparsity and topological constraints. We segmented the evaluation by



(a) Visualizing true origin-destination flows in Manhattan, extracted from empirical TomTom datasets during the validation period.

(b) Visualizing synthetic mobility patterns for Manhattan synthesized by the uTECH-GenUrban PlannerAgent using R^2-driven behavioral reasoning.

Figure 7: Side-by-side comparison of urban mobility corridors in Manhattan: (a) actual traffic data and (b) generated synthetic flows, demonstrating high-fidelity replication of spatial arteries and volume gradients across hexagonal zones.

traffic volume (high-volume arteries vs. low-volume routes) and quantified the prevalence of "ghost flows" instances where the `PlannerAgent` synthesized significant mobility (> 50 trips) in zones where empirical TomTom sensors recorded minimal to no activity (< 5 trips).

Table 4: Comprehensive Performance Metrics Across different cities and different models

City	Model	R^2	CPC	MAE	NRMSE
NYC	openai.gpt-5-mini	0.9771	0.9180	1999.72	0.2328
	meta.llama-4-maverick-17b-instruct	0.9670	0.9100	2136.88	0.2795
	google.gemini-2.5-flash	0.9667	0.9112	2101.46	0.2809
	xai.grok-3-mini	0.9664	0.9113	2103.54	0.2820
Minneapolis	google.gemini-2.5-flash	0.9179	0.8425	717.16	0.4878
	openai.gpt-5-mini	0.9131	0.8433	710.40	0.5018
	meta.llama-4-maverick-17b-instruct	0.9098	0.8413	718.18	0.5110
	xai.grok-3-mini	0.9012	0.8357	739.71	0.5349
Las Vegas	google.gemini-2.5-flash	0.9032	0.5647	328.53	1.7693
	openai.gpt-5-mini	0.8925	0.5569	332.10	1.8647
	xai.grok-3-mini	0.8841	0.5543	332.63	1.9364
	meta.llama-4-maverick-17b-instruct	0.8818	0.5529	333.32	1.9558
San Francisco	google.gemini-2.5-flash	0.8140	0.4837	209.40	3.5712
	xai.grok-3-mini	0.8060	0.4773	210.86	3.6473
	meta.llama-4-maverick-17b-instruct	0.8059	0.4781	210.85	3.6479
	openai.gpt-5-mini	0.8054	0.4772	211.03	3.6527

In contrast, the performance degradation observed in San Francisco is predominantly concentrated within the "long-tail" of low-volume routes. The city's unique geographic profile defined by steep topography and extensive topological "dead zones" (e.g., coastal perimeters, the Presidio) this resulted in a ghost flow rate of 38.8%. The Large Language Model (LLM), which relies heavily on building footprints and commercial capacity as primary spatial attractors, over-predicted potential movement to these zones, causing the low-

volume Normalized Root Mean Square Error (NRMSE) to reach 130.84. This suggests that while the framework effectively models the theoretical capacity for movement, empirical traffic is frequently diverted from these geographically isolated, yet theoretically active, zones.

In Las Vegas, the error profile shifts from geographic constraints to demographic variance. Characterized by a ghost flow rate of 35.1% and an elevated NRMSE in high-volume arteries (0.84), the model encounters difficulties with the extreme volume variance inherent to a tourism-dominated economy. The `PlannerAgent`'s dependence on standardized census data and activity priors tends to under-represent the massive, concentrated flow of non-resident populations along the Las Vegas Strip, leading to volume mismatches despite accurate directional routing.

Ultimately, this decomposition demonstrates that while agent-based generative synthesis maintains high accuracy in standard metropolitan grids, future iterations must incorporate topographical penalty masks and transient population variables to mitigate ghost flows in geographically constrained or highly anomalous urban environments.

5 Discussion and Conclusions

uTECH-GenUrban demonstrates that LLM-driven agents can support a structured and data-grounded urban simulation pipeline across heterogeneous cities. Across Las Vegas, Minneapolis, New York City, and San Francisco, the framework generates socially interpretable demographic cohorts, temporally plausible activity schedules, and realistic aggregate OD flows by combining LLM reasoning with ACS, ATUS, Overture Maps, and TomTom data. The results show that the framework is able to recover broad daily behavioral rhythms and preserves key age-related differences, while generalizing well in mobility synthesis, including strong held-out performance in New York City. At the same time, the experiments reveal clear limits: meal and errand are reproduced more reliably than leisure and work, performance varies across model backends and city contexts, and the system still depends on a hybrid architecture that couples generative reasoning with downstream statistical flow prediction.

Where earlier modeling generations struggled with memoryless agents and city specific calibration requirements, uTECH-GenUrban addresses these limitations by replacing manually specified decision rules with LLM powered agents that reason using demographic data. These agents construct interpretable population cohorts from ACS census distributions and can be applied to different cities without recalibration. The strength of uTECH-GenUrban stems from this hybrid architecture. Its value lies not simply in using LLMs for free-form generation, but in embedding them within a verification aware simulation workflow. Structured outputs, deterministic checks, and self correction make the framework more robust and interpretable than a purely prompt driven pipeline.

However, there are still some important limitations. The framework uses a coarse five activity abstraction and lacks explicit interpersonal or household level interactions. Systematic bias introduced in prompts or benchmark aggregation is also a concern. The mobility component is trained and evaluated on TomTom data from August only, which limits temporal coverage of the framework and may not fully capture seasonal variation or broader shifts in travel behavior. The mobility analysis is also constrained by TomTom's limit of 200 regions per analysis. This restricts the achievable spatial resolution of the OD data and prevents the framework from representing finer-grained movement patterns at a more localized level. The current setup does not explicitly model atypical days, such as holidays, major events, or other disruptions, and therefore is better suited to reproducing regular daily patterns than unusual or highly perturbed mobility conditions. Future work should extend the framework toward richer behavioral representations, multi-day and disrupted condition simulation, broader temporal data coverage, finer spatial calibration, uncertainty-aware validation, and stronger integration with policy-scenario analysis. These updates would further position uTECH-GenUrban as a scalable and explainable foundation for urban digital twins.

References

Muhammad Adnan, Fatma Outay, Shiraz Ahmed, Erika Brattich, Silvana di Sabatino, and Davy Janssens. Integrated agent-based microsimulation framework for examining impacts of mobility-oriented policies. *Personal and Ubiquitous Computing 25*, 25(1):205–217, January 2021. ISSN 1617-4917. doi: 10.1007/s00779-020-01363-w. URL http://dx.doi.org/10.1007/s00779-020-01363-w.

Charles L. Baum. The effects of vehicle ownership on employment. *Journal of Urban Economics*, 66(3): 151–163, 2009. Accessed: 2025-08-02.

Piotr Bialowolski, Christos A. Makridis, Matt Bradshaw, Dorota Weziak-Bialowolska, Craig Gundersen, Noémie Le Pertel, Cristina Gibson, Sung Joon Jang, R. Noah Padgett, Byron R. Johnson, and Tyler J. VanderWeele. Analysis of demographic variation and childhood correlates of financial well-being across 22 countries. *Nature Human Behaviour*, 9:917–932, 2025. doi: 10.1038/s41562-025-02207-4. Accessed: 2025-08-02.

Nicolas Bougie and Narimawa Watanabe. Citysim: Modeling urban behaviors and city dynamics with large-scale llm-driven agent simulation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 215–229, Suzhou, China, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-industry.15. URL https://aclanthology.org/2025.emnlp-industry.15/.

Isaac Brodsky. H3: Uber's hexagonal hierarchical spatial index. https://www.uber.com/blog/h3/, 2018. Accessed: 2025-08-02.

Christian Calderón, Pasqual Martí, Jaume Jordán, Juan Palanca, and Vicente Julian. Cognitive agents in urban mobility: Integrating LLM reasoning into multi-agent simulations. *Sensors*, 25(18):5688, 2025. doi: 10.3390/s25185688.

Sung-Bae Cho and Jin-Young Kim. Clustered embedding using deep learning to analyze urban mobility based on complex transportation data. *PLoS One*, 16(4):e0249318, 2021. doi: 10.1371/journal.pone.0249318.

City and County of San Francisco. Key sectors in san francisco. https://www.sf.gov/key-sectors-in-san-francisco, 2025. Accessed: 2026-03-29.

Clark County, Nevada. Living and working in clark county. https://www.clarkcountynv.gov/residents/about_clark_county/, 2025. Accessed: 2026-03-29.

Balázs Cs. Csáji, Arnaud Browet, V.A. Traag, Jean-Charles Delvenne, Etienne Huens, Paul Van Dooren, Zbigniew Smoreda, and Vincent D. Blondel. Exploring the mobility of mobile phone users. *Physica A: Statistical Mechanics and its Applications*, 392(6):1459–1473, March 2013. ISSN 0378-4371. doi: 10.1016/j.physa.2012.11.040. URL http://dx.doi.org/10.1016/j.physa.2012.11.040.

William Davidson, Robert Donnelly, Peter Vovsha, Joel Freedman, Steve Ruegg, Jim Hicks, Joe Castiglione, and Rosella Picado. Synthesis of first practices and operational research approaches in activity-based travel demand modeling. *Transportation Research Part A: Policy and Practice*, 41(5):464–488, 2007. ISSN 0965-8564. doi: https://doi.org/10.1016/j.tra.2006.09.003. URL https://www.sciencedirect.com/science/article/pii/S0965856406001121.

Y. Du, J. Feng, J. Yuan, and Y. Li. Cams: A citygpt-powered agentic framework for urban human mobility simulation. *arXiv preprint arXiv:2506.13599*, 2025.

Chen Gao, Xiaochong Lan, Nian Li, Yuan Yuan, Jingtao Ding, Zhilun Zhou, Fengli Xu, and Yong Li. Large language models empowered agent-based modeling and simulation: A survey and perspectives. *Humanities and Social Sciences Communications*, 11:1259, 2024. doi: 10.1057/s41599-024-03611-3. URL https://www.nature.com/articles/s41599-024-03611-3.

Saibo Geng, Hudson Cooper, Michał Moskal, Samuel Jenkins, Julian Berman, Nathan Ranchin, Robert West, Eric Horvitz, and Harsha Nori. Jsonschemabench: A rigorous benchmark of structured outputs for language models. *arXiv preprint arXiv:2501.10868*, 2025. URL https://arxiv.org/abs/2501.10868.

Hennepin County. Minnesota's economic driver. https://www.hennepin.us/-/media/hennepinus/your-government/open-government/igr/economic-driver.pdf, 2023. Accessed: 2026-03-29.

Button Kenneth J. Hensher, David A. 3. the four-step model, 2008.

Chiawei Hsu, Chao Fan, and Ali Mostafavi. Limitations of gravity models in predicting fine-scale spatial-temporal urban mobility networks, 2021. URL https://arxiv.org/abs/2109.03873.

Na Jiang, Fuzhen Yin, Boyu Wang, and Andrew Crooks. A large-scale geographically explicit synthetic population with social networks for the united states. *Scientific Data*, 11, 11 2024. doi: 10.1038/s41597-024-03970-1.

Pascal Jutras-Dubé, Mohammad B. Al-Khasawneh, Zhichao Yang, Javier Bas, Fabian Bastin, and Cinzia Cirillo. Copula-based transferable models for synthetic population generation. *Transportation Research Part C: Emerging Technologies*, 169:104830, 2024. ISSN 0968-090X. doi: https://doi.org/10.1016/j.trc.2024.104830. URL https://www.sciencedirect.com/science/article/pii/S0968090X24003516.

Sung-Yoo Lim, Hyunsoo Yun, Prateek Bansal, Dong-Kyu Kim, and Eui-Jin Kim. A large language model for feasible and diverse population synthesis. *Transportation Research Part C: Emerging Technologies*, 185:105581, 2026. ISSN 0968-090X. doi: https://doi.org/10.1016/j.trc.2026.105581. URL https://www.sciencedirect.com/science/article/pii/S0968090X26000690.

Qi Liu, Can Li, and Wanjing Ma. Gatsim: Urban mobility simulation with generative agents. *Transportation Research Part C: Emerging Technologies*, 186:105576, 2026. doi: 10.1016/j.trc.2026.105576. URL https://www.sciencedirect.com/science/article/pii/S0968090X26000641.

Yaxi Lu, Haolun Li, Xin Cong, Zhong Zhang, Yesai Wu, Yankai Lin, Zhiyuan Liu, Fangming Liu, and Maosong Sun. Learning to generate structured output with schema reinforcement learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4905–4918, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.243. URL https://aclanthology.org/2025.acl-long.243/.

National Center for Education Statistics. Annual earnings by educational attainment. https://nces.ed.gov/programs/coe/indicator/cba/annual-earnings, 2024. Accessed: 2025-08-02.

Ariel Noyman, Kai Hu, and Kent Larson. Travelagent: Generative agents in the built environment. *Environment and Planning B: Urban Analytics and City Science*, 53(2):377–397, 2026. doi: 10.1177/23998083251360458. URL https://journals.sagepub.com/doi/10.1177/23998083251360458.

Overture Maps. overturemaps-py: Python client and command-line interface for overture maps. https://github.com/OvertureMaps/overturemaps-py, 2025. Accessed: 2025-07-15.

Overture Maps Foundation. Overture maps foundation: Latest open map data release. https://overturemaps.org/download, 2025. Accessed: 2025-07-15.

Virginia Padilla and Jacinto D'avila. A survey of multi-agent geosimulation methodologies: From abm to llm. *arXiv preprint arXiv:2507.23694*, 2025. doi: 10.48550/arXiv.2507.23694. URL https://arxiv.org/abs/2507.23694.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23, 2023. doi: 10.1145/3586183.3606763.

Hui Shi and Konstadinos G. Goulias. Year-to-year time allocation and spatial structure of americans' daily schedules from 2019 to 2022 and a detailed analysis of the stay-at-home all-day patterns. *Transportation Research Part A: Policy and Practice*, 190:104190, 2024. ISSN 0965-8564. doi: https://doi.org/10.1016/j.tra.2024.104190. URL https://www.sciencedirect.com/science/article/pii/S0965856424002386.

TomTom International BV. Tomtom developer portal. https://developer.tomtom.com/, 2025. Accessed: 2025-08-02.

U.S. Bureau of Labor Statistics. American time use survey user's guide. https://www.bls.gov/tus/atususersguide.pdf, 2019. Accessed April 2, 2026.

U.S. Bureau of Labor Statistics. American time use survey news release archive (2021). https://www.bls.gov/news.release/archives/atus_07222021.htm, 2021. Accessed April 2, 2026.

U.S. Bureau of Labor Statistics. American time use survey activity lexicon 2023. https://www.bls.gov/tus/lexicons/lexiconwex2023.pdf, 2023. Accessed April 2, 2026.

U.S. Bureau of Labor Statistics. American time use survey – 2024 results. https://www.bls.gov/news.release/pdf/atus.pdf, 2025a. Accessed April 2, 2026.

U.S. Bureau of Labor Statistics. Atus activity coding lexicons and coding manuals. https://www.bls.gov/tus/lexicons.htm, 2025b. Accessed April 2, 2026.

U.S. Census Bureau. Quickfacts: Clark county, nevada. https://www.census.gov/quickfacts/fact/table/clarkcountynevada/PST045224, 2025a. Accessed: 2026-03-29.

U.S. Census Bureau. Quickfacts: Hennepin county, minnesota. https://www.census.gov/quickfacts/fact/table/hennepincountyminnesota/PST045224, 2025b. Accessed: 2026-03-29.

U.S. Census Bureau. Quickfacts: New york county, new york. https://www.census.gov/quickfacts/fact/table/newyorkcountynewyork/PST045224, 2025c. Accessed: 2026-03-29.

U.S. Census Bureau. Quickfacts: San francisco county, california. https://www.census.gov/quickfacts/fact/table/sanfranciscocountycalifornia/PST045224, 2025d. Accessed: 2026-03-29.

U.S. Census Bureau. American community survey (acs) program. https://www.census.gov/programs-surveys/acs.html, 2025e. Accessed: 2025-07-15.

Paul Waddell. Urbansim: Modeling urban development for land use, transportation and environmental planning. *Journal of the American Planning Association*, 68:297–314, 12 2002. doi: 10.1080/01944360208976274.

Jiawei Wang, Renhe Jiang, Chuang Yang, Zengqing Wu, Makoto Onizuka, Ryosuke Shibasaki, Noboru Koshizuka, and Chuan Xiao. Large language models as urban residents: An llm agent framework for personal mobility generation. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/e142fd2b70f10db2543c64bca1417de8-Abstract-Conference.html.

Haowen Xu, Jinghui Yuan, Anye Zhou, Guanhao Xu, Wan Li, Xuegang Ban, and Xinyue Ye. Genai-powered multi-agent paradigm for smart urban mobility: Opportunities and challenges for integrating large language models (llms) and retrieval-augmented generation (rag) with intelligent transportation systems. *arXiv preprint arXiv:2409.00494*, 2024. doi: 10.48550/arXiv.2409.00494. URL https://arxiv.org/abs/2409.00494.

Yuwei Yan, Qingbin Zeng, Zhiheng Zheng, et al. Opencity: A scalable platform to simulate urban activities with massive llm agents. *arXiv preprint arXiv:2410.21286*, 2024. doi: 10.48550/arXiv.2410.21286.

Yang Yang, Samitha Samaranayake, and Timur Dogan. A clustering-based approach to quantifying socio-demographic impacts on urban mobility patterns. *Environment and Planning B: Urban Analytics and City Science*, 50(9):2452–2469, 2023. doi: 10.1177/23998083231159909.

Çaglar Tozluoglu. Agent-based models for evaluating sustainable transportation. Chalmers Research Publication, 2025.