AVE Trends in Intelligent Computing Systems

CAD-MOB: A Unified Agentic Causal Architecture for Human Mobility Prediction

K. Srivarsan[1,*], Yoo Da-Eun[2], Ahmad Firdaus Bin Yusof[3], S. Tejas[4], Amy Eversole[5]

[1]Department of Artificial Intelligence and Machine Learning, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[2]Department of Human Resources, Jeonju Craft Industries, Jeonju, Jeonbuk, South Korea.
[3]Department of Procurement, Alor Setar Agro Sdn Bhd, Alor Setar, Kedah, Malaysia.
[4]Department of Data Science, Analytics and Engineering, Arizona State University, Tempe, Arizona, United States of America.
[5]Department of Mathematics, Texas A&M University, College Station, Texas, United States of America.
srivarsankannan@gmail.com[1], yoo.daeun@jeonju-craft.kr[2], firdaus.yusof@alorsetar-agro.com.my[3], tsundar1@asu.edu[4], amy.eversole@tamu.edu[5]

*Corresponding author

Abstract: Three fundamental and interacting failure modes challenge urban human mobility prediction: the correlational brittleness of purely data-driven models, the combinatorial sparsity of individual trajectory data, and the inability of statistical models to reason about interventional counterfactuals: what happens to mobility patterns when urban infrastructure changes. LSTM and Transformer-based sequence models for next-location prediction perform well on in-distribution test sets but fail under cold-start conditions, sparse trajectory inputs, and distribution shift caused by road closures or transit disruptions. CAD-Mob (Causal-Agentic Diffusion Mobility) is a three-pillar system that requires design decisions to address documented failure modes directly. The framework includes ProDiff, a probabilistic reverse-diffusion trajectory synthesiser that address trajectory sparsity through iterative denoising over urban constraint manifolds; a Structural Causal Model (SCM) head that uses DoWhy-based do-calculus to estimate the absolute causal effect of urban interventions, replacing correlation-based predictions with interventional reasoning; and AgentMove, an LLM-driven cognitive head that models behavioural history CAD-Mob outperforms LSTM baselines in trajectory recovery and zero-shot next-location accuracy on the Porto Taxi and T-Drive datasets with a real-time GTFS-RT ingestion pipeline. Live urban intelligence deployment using Studio UI (Next.js), Express gateway, and FastAPI ML core. This study shows that operationally reliable mobility intelligence requires causal grounding, generative imputation, and agentic reasoning.

Keywords: Human Mobility Prediction; Diffusion Models; Structural Causal Models; Large Language Models; Agent-Move and Prodiff; Next-Location Prediction; Trajectory Imputation; GTFS-RT and Porto Dataset.

Cite as: K. Srivarsan, Y. Da-Eun, A. F. B. Yusof, S. Tejas, and A. Eversole, "CAD-MOB: A Unified Agentic Causal Architecture for Human Mobility Prediction," *AVE Trends in Intelligent Computing Systems*, vol. 3, no. 3, pp. 132–143, 2026.

Journal Homepage: https://www.avepubs.com/user/journals/details/ATICS

Received on: 10/06/2025, **Revised on:** 05/10/2025, **Accepted on:** 08/11/2025, **Published on:** 07/09/2026

DOI: https://doi.org/10.64091/ATICS.2026.000352

1. Introduction

1.1. The Challenge of Human Mobility Prediction at Urban Scale

Copyright © 2026 K. Srivarsan *et al.,* licensed to AVE Trends Publishing Company. This is an open access article distributed under CC BY-NC-SA 4.0, which permits copying, remixing, redistributing, transformation, and building upon the material in any medium so long as the original work is properly cited.

Human mobility is the connective tissue of urban systems. Accurate prediction of where people will move next and why enables smarter transit planning, dynamic traffic management, epidemic containment, and personalised navigation services [1]. According to the International Transport Forum, urban mobility inefficiencies cost developed economies 2-4 percent of GDP annually through congestion, missed connections, and suboptimal infrastructure utilisation [18]. The proliferation of GPS-enabled devices and urban sensor networks has produced unprecedented volumes of trajectory data: the Porto taxi dataset alone captures 1.7 million individual taxi trips comprising 83 million GPS waypoints collected over 442 days of continuous urban operation [2]. Despite this data abundance, three structurally distinct failure modes prevent reliable mobility intelligence in operational settings [17]. First, correlational brittleness: models trained to predict next locations by learning statistical co-occurrence patterns in historical trajectories fail when the urban graph changes: road closures, new transit lines, and event-driven crowd surges are precisely the conditions under which prediction is most operationally critical, yet these constitute distribution shift for purely correlational models [3]. Second, trajectory sparsity: many commercially important users exhibit sparse historical trajectories- new residents, infrequent travellers, and privacy-conscious individuals who opt out of continuous tracking- making supervised next-location prediction impossible without explicit imputation [4]. Third, interventional opacity: decision-makers require not merely predictions of what will happen given current conditions, but counterfactual estimates of what would happen if an intervention were applied: the causal quantity that statistical models are fundamentally unable to estimate [5].

1.2. Why Standard Mobility Models Fail and what CAD-Mob does about it

The deep learning literature has produced a succession of increasingly sophisticated sequence models for mobility prediction. LSTM-based approaches establish strong baselines on benchmark datasets but cannot reason about interventions and degrade sharply under sparse trajectory conditions [6]. Transformer-based mobility models improve contextual representation but inherit the same distributional assumptions. Mobility GNNs encode road network topology but conflate structural correlation with causal mechanism. No existing architecture unifies trajectory imputation, causal interventional reasoning, and zero-shot prediction in a single coherent system [16]. CAD-Mob proposes a failure-aware architecture that addresses all three failure modes through three dedicated pillars, each motivated by a specific documented failure. The ProDiff pillar addresses trajectory sparsity through probabilistic reverse-diffusion synthesis. The Structural Causal Model pillar addresses correlational brittleness by grounding predictions in do-calculus interventional estimates. The AgentMove pillar addresses cold-start and zero-shot conditions through LLM-driven behavioural reasoning over semantic location representations [7]; [11].

1.3. Contributions

- **Failure-Driven Framework:** Researchers provide rigorous empirical documentation of LSTM and Transformer collapse under sparse trajectory conditions and urban distribution shift, establishing a template for failure-first system design in mobility intelligence [8].
- **CAD-Mob Unified Architecture:** Researchers propose the first unified framework that integrates diffusion generative models, structural causal inference, and LLM-driven agentic reasoning as three inseparable pillars of a single, coherent mobility intelligence system.
- **ProDiff Trajectory Synthesiser:** Researchers introduce a probabilistic reverse-diffusion architecture for trajectory imputation that recovers sparse trajectories by iterative denoising over urban constraint manifolds, enabling supervised prediction for previously unpredictable sparse-data users.
- **Causal SCM Head:** Researchers implement a Structural Causal Model head using DoWhy do-calculus to estimate interventional causal effects of urban changes, giving decision-makers counterfactual mobility estimates that correlational models cannot provide [20].
- **AgentMove Cognitive Reasoning:** Researchers deploy an LLM-based agentic head that models user behavioural history and semantic routine patterns, achieving zero-shot next-location prediction without requiring labelled trajectory data [19].
- **Production-Grade Deployment:** Researchers deliver a fully containerised multi-service platform: Next.js Studio UI, Express.js gateway, and FastAPI ML core, with live GTFS-RT telemetry ingestion, constituting the first production-ready causal mobility intelligence system [9].

2. Related Work

2.1. Deep Learning for Next-Location Prediction

The next-location prediction literature has progressed from Markov chain models to recurrent architectures, then to attention-based and graph-enhanced approaches. Feng et al. [3] introduced the attention-based recurrent mobility model, demonstrating that combining historical mobility patterns with periodic regularity through attention mechanisms substantially improves prediction accuracy over vanilla LSTM baselines. STRNN [6] incorporates spatio-temporal intervals into transition modelling,

showing that both spatial distance and temporal gap between consecutive check-ins carry predictive information. However, all recurrent architectures share a fundamental limitation: they encode sequential patterns without modelling the generative process that produces trajectories, making them brittle to distributional shift and unable to impute missing trajectory segments. Graph-enhanced mobility models address the structural encoding gap by incorporating road network topology and point-of-interest relationships [13]. STGAT applies graph attention over spatial proximity graphs to encode neighbourhood interactions. Flash [7] uses hierarchical graph representation to capture multi-scale mobility patterns. These approaches improve in-distribution performance but remain correlational: the graph encodes observed co-occurrence, not causal mobility structure, and does not support counterfactual queries [14].

2.2. Diffusion Models for Trajectory Generation

Diffusion probabilistic models have demonstrated remarkable capability in synthesising high-dimensional structured data through iterative denoising. DiffTraj applies denoising diffusion to trajectory generation, demonstrating that the reverse diffusion process can recover realistic urban trajectories from Gaussian noise conditioned on origin-destination constraints. TrajGen extends this to conditional trajectory synthesis, conditioning the denoising process on road network structure. ProDiff in CAD-Mob builds on this lineage but introduces a critical innovation: probabilistic urban constraint manifolds. The denoising process is conditioned on dynamic urban constraints, including real-time traffic state, GTFS-RT transit availability, and temporal mobility patterns, enabling imputation of realistic trajectories for sparse-data users [10].

2.3. Causal Inference in Urban Computing

The application of causal inference to urban mobility has been limited but impactful. Causal discovery methods [12] have been applied to identify causal relationships in traffic flow data, showing that standard correlation-based traffic models systematically misattribute congestion propagation [4]. DoWhy, Microsoft's Python library implementing Pearl's do-calculus, provides a principled framework for estimating interventional causal effects from observational data. CAD-Mob is the first mobility prediction system to integrate a full DoWhy-based SCM head as a primary prediction component, enabling operational counterfactual queries of the form: what is the causal effect of closing Road X on the travel time distribution of users travelling to Zone Y?

2.4. LLM-Driven Mobility Reasoning

Feng et al. [14] established that large language models can reason about human mobility through semantic understanding of location types, routine patterns, and behavioural context, achieving competitive zero-shot next-location prediction without any mobility-specific training. Stockl [15] demonstrated that LLM-based mobility agents outperform supervised baselines in cold-start scenarios. CAD-Mob integrates AgentMove reasoning as a dedicated cognitive head within the unified architecture, combining it with the generative and causal pillars to address complementary failure modes that pure LLM reasoning cannot.

3. Dataset and Problem Setup

3.1. Datasets

CAD-Mob is evaluated on two benchmark trajectory datasets and a real-time transit telemetry stream, collectively capturing the breadth of operational mobility intelligence requirements (Table 1).

Table 1: Dataset composition and statistics for CAD-Mob evaluation

Dataset	Type	Scale	Coverage	Primary Use
Porto Taxi	GPS Trajectories	1.7M trips, 83M waypoints	Porto, Portugal; 442 days	Trajectory imputation, causal intervention
T-Drive	Taxi Traces	10,357 taxis, 15M points	Beijing; 7 days	Next-location prediction
GTFS-RT	Real-Time Transit	Live vehicle positions	Configurable urban area	Real-time telemetry ingestion

3.2. Problem Formulation

CAD-Mob addresses three formally distinct but operationally interdependent prediction problems over urban trajectory data. Let $T = \{(loc_1, t_1), ..., (loc_n, t_n)\}$ denote a trajectory as a timestamped sequence of location visits, where each loc_i is a point-of-interest or GPS coordinate and t_i is the visit timestamp. The next-location prediction task requires estimating $P(loc_{n+1} \mid T_{1:n})$, the conditional distribution over the next visited location given the observed trajectory prefix. The

trajectory imputation task requires recovering the complete trajectory T from a sparse observation T_sparse = {(loc_i, t_i): i in S} where S is a subset of observed timesteps. The interventional prediction task requires estimating E[Y | do(X = x)], the causal effect of intervention X on mobility outcome Y under Pearl's do-calculus, where X may be a road closure, transit disruption, or urban infrastructure change. The unified Mobility Representation Vector (MRV) bridges all three tasks:

$$MRV_u = Concat(h_{ProDiff}(T_{sparse}), h_{SCM}(do(X)), h_{AgentMove}(T_{history})) \quad (1)$$

Where h_ProDiff recovers the latent trajectory representation from sparse observations through reverse diffusion, h_SCM encodes interventional causal effects estimated by the SCM head, and h_AgentMove encodes the user's behavioural history and semantic routine patterns through LLM reasoning.

3.3. Failure Analysis of Standard Methods

3.3.1. Baseline Model Performance

Before presenting the proposed framework, researchers establish rigorous empirical documentation of where and why standard approaches fail. This failure analysis underpins CAD-Mob: every architectural decision responds directly to a documented failure mode rather than an independent contribution (Table 2).

Table 2: Overall model performance comparison on Porto and T-Drive test sets

Model	Acc@1 (Porto)	Acc@1 (T-Drive)	Imputation RMSE	Notes
Markov Chain	0.213	0.198	N/A	No temporal context
LSTM (DeepMove)	0.412	0.389	0.524	Strong in-distribution
Transformer (Vanilla)	0.438	0.401	0.491	Brittle under shift
STGAT (Graph)	0.451	0.421	0.468	Graph topology helps
CAD-Mob (Proposed)	0.614	0.587	0.213	All failure modes addressed

3.3.2. The Sparse Trajectory Collapse

The most instructive empirical finding in this evaluation is the complete prediction collapse of LSTM and Transformer baselines under sparse trajectory conditions. When the observed trajectory contains fewer than 5 historical visits- a realistic condition for 31.4% of T-Drive users and 27.8% of Porto taxi sessions- LSTM next-location accuracy degrades from Acc@1 = 0.412 to Acc@1 = 0.089, equivalent to near-random prediction.

This collapse is structurally motivated: recurrent architectures need sufficient sequential context to build meaningful hidden-state representations, and sparse trajectories provide insufficient signal. The ProDiff imputation pillar in CAD-Mob directly addresses this by recovering full trajectory representations from sparse observations before the prediction head is applied.

3.3.3. Correlational Brittleness under Urban Interventions

A controlled evaluation assessing model performance before and after a simulated urban intervention- specifically, the removal of a major arterial route from the Porto Road graph- demonstrates the correlational brittleness of standard approaches. LSTM Acc@1 degrades from 0.412 to 0.231 (43.9% relative degradation) when route removal is imposed, because the model has learned strong correlational associations between the affected route and downstream destinations that are no longer causally valid.

The SCM head in CAD-Mob, which estimates interventional effects through do-calculus rather than correlation, retains predictive accuracy under the same intervention, demonstrating a fundamental architectural advantage that cannot be recovered through training data augmentation or regularization.

4. Proposed Framework: CAD-MOB

The CAD-Mob framework is structured around three conceptual pillars that address the three identified failure classes. Generation (how the system addresses trajectory sparsity through probabilistic synthesis), Causation (how the system replaces correlational brittleness with interventional reasoning), and Cognition (how the system enables zero-shot prediction through agentic LLM reasoning). Each component of the five-stage pipeline follows from the failure analysis rather than contributing independently.

4.1. Mathematical Formalization

The ProDiff reverse diffusion process is formalized as follows. Given a sparse trajectory observation T_sparse, the forward diffusion process adds Gaussian noise over T denoising steps:

$$q(x_t \mid x_{t-1}) = N(x_t; \sqrt{1 - \beta_t} * x_{t-1}, \beta_t * I) \quad (2)$$

The reverse denoising process recovers a plausible full trajectory by learning the conditional score function:

$$p_\theta(x_{t-1} \mid x_t, c) = N(x_{t-1}; \mu_\theta(x_t, t, c), \sigma_t^2 * I) \quad (3)$$

Where c is the conditioning context encoding urban constraints. The SCM interventional effect estimate using Pearl's do-calculus is:

$$E[Y \mid do(X = x)] = \sum_z P(Y \mid X = x, Z = z) * P(Z = z) \quad (4)$$

Where Z denotes the set of confounding variables in the SCM's directed acyclic graph, and P(Y | X, Z) is estimated from the observational Porto trajectory data. The unified MRV concatenation feeding the final prediction head is:

$$\hat{y} = f_{pred}(Concat(h_{ProDiff}, h_{SCM}, h_{AgentMove}, timestep, context_embed)) \quad (5)$$

4.2. ProDiff: Probabilistic Reverse-Diffusion Trajectory Synthesizers

ProDiff is implemented as a U-Net denoising architecture conditioned on three urban constraint signals: the observed sparse trajectory checkpoints (hard constraints), the road network adjacency structure derived from the Porto OSM graph (structural constraints), and real-time GTFS-RT transit availability (dynamic constraints). The U-Net employs skip connections across encoder-decoder scales to preserve fine-grained GPS coordinate precision through the denoising chain. Training uses a denoising score-matching objective that minimizes expected noise-prediction error across all T = 1000 diffusion timesteps. At inference, trajectory imputation uses 50-step DDIM sampling, reducing the computational cost of full trajectory recovery by a factor of 20 relative to standard DDPM sampling.

4.3. Structural Causal Model Head

The SCM head implements a directed acyclic graph over six primary mobility variables: departure time, origin zone, destination zone, road network state, transit availability, and travel duration. Researchers construct the DAG using the PC algorithm on Porto taxi data, with domain knowledge constraints enforcing temporal ordering. Interventional estimates are computed using DoWhy's linear regression estimator for continuous outcomes and propensity score weighting for discrete location outcomes. Researchers evaluate the SCM head against observational correlation baselines using a controlled intervention experiment, where road closure events from Porto's historical record serve as natural experiments. Under road-closure conditions, SCM achieves interventional prediction error 33.4% lower than the correlation baseline, validating its causal grounding as a substantive, not cosmetic, contribution.

4.4. Agent Move Cognitive Head

Agent Move integrates an LLM reasoning module that processes a structured semantic representation of the user's mobility history. For each query, the module constructs a natural-language prompt encoding the user's top-5 most frequently visited location categories, the temporal distribution of past visits by day-of-week and hour-of-day, the current context (time, day, last known location), and available transit options from the GTFS-RT stream. The LLM reasons over this context to produce a ranked list of likely next locations with associated confidence scores. This zero-shot capability is particularly valuable for users with sparse historical trajectories, where ProDiff imputation provides insufficient context for supervised prediction heads.

4.5. System Architecture and Implementation

CAD-Mob is implemented as a production-grade distributed system of three primary services orchestrated via Docker Compose, reflecting the operational deployment requirements of real-world urban intelligence platforms (Table 3).

Table 3: CAD-Mob multi-service architecture tiers and technology stack

Tier	Component	Technology	Role

Frontend	Studio UI	Next.js 16 (Turbopack)	3D visualisation, interventional control panel
Gateway	API Bridge	Express.js (TypeScript)	GTFS-RT ingestion, authentication, API proxy
ML Core	Intelligence Core	Python 3.10 / FastAPI	ProDiff, SCM head, AgentMove reasoning
Sandbox	Model Training	PyTorch 2.x / CUDA	GPU-accelerated diffusion model refinement

4.5.1. Frontend: Studio UI

The Studio UI is a Next.js 16 application using Turbopack for sub-second hot module replacement during development. The interface provides three primary views: a 3D map dashboard that renders real-time trajectory data using WebGL with Three.js, an interventional control panel that lets operators specify do-calculus queries and visualise counterfactual mobility estimates, and a city analytics dashboard that aggregates fleet-level mobility statistics. UI components are built on shadcn/ui with Tailwind CSS 4. The causal intervention interface is particularly novel: operators can draw road closures directly on the 3D map and immediately receive SCM-estimated counterfactual travel time distributions for affected origin-destination pairs.

4.5.2. ML Core: Fast-API Intelligence Server

The Python ML core exposes three primary endpoints corresponding to the three CAD-Mob pillars: /predict/prodiff for trajectory imputation requests, /predict/causal for SCM interventional queries, and /predict/agentmove for LLM-driven next-location reasoning. All endpoints are asynchronous FastAPI routes supporting concurrent inference requests. ProDiff inference uses pre-loaded model weights (prodiff_latest.pth) and executes 50-step DDIM sampling on CPU or GPU depending on hardware availability. The SCM head uses a pre-fitted DoWhy causal model serialised via joblib. AgentMove calls are routed to the configured LLM API with the structured semantic prompt template.

4.5.3. Hyperparameter Configuration

Table 4. Hyperparameter configuration of the CAD-Mob framework. Researchers include architectural settings, inference procedures, conditioning processes, and training parameters for each component. It also describes the dataset partitioning technique and optimization settings to ensure consistent, reproducible model training and evaluation.

Table 4: Full hyperparameter configuration for the CAD-Mob framework

Component	Parameter	Value
ProDiff	Architecture	U-Net with skip connections, T=1000 diffusion steps
ProDiff	Inference sampling	DDIM 50-step accelerated
ProDiff	Conditioning	Sparse checkpoints + road graph + GTFS-RT
SCM Head	DAG construction	PC algorithm + domain constraints
SCM Head	Estimator	DoWhy linear regression + propensity weighting
AgentMove	LLM backend	Configurable (GPT-4o / local LLM)
AgentMove	Context window	Top-5 location history + temporal distribution
Training	Batch size / Epochs	64 / 200 (ProDiff pre-training)
Training	Learning rate	1e-4 with cosine annealing
Dataset Split	Train / Val / Test	70% / 10% / 20% temporal split

4.6. Experimental Results

4.6.1. Next-Location Prediction Performance

Table 5 presents the comparative next-location prediction performance across all evaluated models on Porto and T-Drive test sets. CAD-Mob achieves Acc@1 = 0.614 on Porto and 0.587 on T-Drive, representing 36.2% and 39.4% relative improvements over the best LSTM baseline, respectively. The AgentMove zero-shot head contributes the largest marginal gain under sparse-trajectory conditions (fewer than 5 observed visits), achieving Acc@1 = 0.341 compared to LSTM Acc@1 = 0.089 and demonstrating the value of semantic reasoning when statistical context is insufficient.

Table 5: Next-location prediction accuracy comparison and sparse acc@1 measures performance on users with fewer than 5 historical visits

Model	Acc@1 Porto	Acc@5 Porto	Acc@1 T-Drive	Acc@5 T-Drive	Sparse Acc@1
Markov Chain	0.213	0.387	0.198	0.361	0.112

LSTM (DeepMove)	0.412	0.623	0.389	0.598	0.089
Transformer	0.438	0.651	0.401	0.612	0.094
STGAT	0.451	0.667	0.421	0.634	0.101
AgentMove (ZS only)	0.501	0.712	0.468	0.681	0.341
CAD-Mob (Full)	0.614	0.821	0.587	0.793	0.418

4.6.2. Trajectory Imputation Performance

Table 6 presents trajectory imputation results measured by RMSE (metres) and Hausdorff distance between recovered and ground-truth trajectories. ProDiff achieves an imputation RMSE of 47.3 metres on Porto, a 59.7% improvement over linear interpolation and a 35.2% improvement over the nearest-neighbour conditional generation baseline.

Table 6: Trajectory imputation performance and Lower is better for all metrics

Method	RMSE (m) Porto	Hausdorff Porto	RMSE (m) T-Drive	Hausdorff T-Drive
Linear Interpolation	117.4	284.1	131.2	309.6
Nearest-Neighbour Cond	73.1	198.4	84.7	221.3
GAN-based Synthesis	61.8	173.2	72.4	192.8
ProDiff (CAD-Mob)	47.3	124.7	54.1	138.9

The Hausdorff distance improvement is particularly significant, indicating that ProDiff avoids the catastrophic trajectory deviations that alternative imputation methods exhibit near complex intersections and route branching points.

4.6.3. Causal Intervention Accuracy

The SCM head is evaluated on 47 documented road closure events from Porto's historical record, treating each closure as a natural experiment and comparing predicted post-intervention mobility outcomes against observed ground truth (Table 7).

Table 7: Causal intervention prediction accuracy on 47 Porto road closure events

Method	Intervention MAE (Min.)	Directional Accuracy	Notes
LSTM Correlation	7.7	68.8%	Fails near affected routes
Transformer Correlation	7.1	70.4%	Marginally better
Difference-in-Differences	5.9	79.1%	Standard quasi-causal
SCM Head (CAD-Mob)	4.3	91.5%	Do-calculus grounding

The SCM head achieves a Mean Absolute Error (MAE) of 4.3 minutes in post-intervention travel time prediction, compared to 7.7 minutes for the correlation-based LSTM baseline, representing a 44.2% improvement. More critically, the LSTM baseline produces directionally incorrect predictions (predicting reduced travel time when the true effect is increased travel time) in 31.2% of closure events, compared to 8.5% for the SCM head.

4.6.4. Ablation Study

Table 8 presents the architectural ablation, confirming the complementary contribution of each CAD-Mob pillar. Removing ProDiff imputation degrades sparse-trajectory Acc@1 by 49.5%, confirming that the generative pillar is essential for the large population of sparse-data users. Removing the SCM head has minimal impact on aggregate Acc@1. Still, it degrades intervention prediction MAE by 44.2%, confirming that the causal pillar's value lies in interventional reasoning not captured by standard accuracy metrics. Removing AgentMove degrades zero-shot Acc@1 by 18.4%, confirming the cognitive pillar's contribution to cold-start robustness.

Table 8: Ablation study confirming the complementary contribution of each CAD-Mob pillar

Configuration	Acc@1 Porto	Sparse Acc@1	Intervention MAE	Notes
No ProDiff	0.501	0.211	4.3	-32% on sparse users
No SCM Head	0.608	0.407	7.7	Loses causal grounding
No Agent Move	0.531	0.341	4.5	-13% overall
Full CAD-Mob	0.614	0.418	4.3	Best on all dimensions

5. Discussions

5.1. Why Causal Grounding Matters Beyond Accuracy

The experimental results reveal an important tension that aggregate accuracy metrics conceal: STGAT achieves Acc@1 = 0.451 on Porto, only 36.4% below CAD-Mob, and might appear competitive. However, this comparison is fundamentally misleading for operational deployment. STGAT's 31.2% directional error rate under road-closure interventions means a city traffic management system using STGAT would recommend routing strategies that worsen congestion rather than improve it in nearly one in three intervention scenarios. The SCM head's 91.5% directional accuracy represents the difference between an operationally viable and an operationally harmful urban intelligence system: a distinction that aggregate Acc@1 entirely fails to capture.

5.2. The Role of Diffusion in Mobility Equity

A dimension of CAD-Mob's contribution that extends beyond technical performance is its implications for mobility equity. Standard supervised next-location prediction models implicitly discriminate against sparse-data users: the 31.4% of T-Drive users with fewer than 5 historical visits, by providing near-random predictions for precisely the users who may be most dependent on navigation and planning assistance: new residents, elderly users with limited device engagement, and privacy-conscious users. ProDiff's trajectory imputation capability eliminates this discriminatory accuracy gap: CAD-Mob's Sparse Acc@1 of 0.418 represents a 369.7% improvement over LSTM's 0.089 for sparse users, effectively equalising prediction quality across the full user population.

5.3. Production Deployment Validation

The CAD-Mob production architecture was stress-tested under simulated operational loads of 1,000 concurrent trajectory queries and 200 concurrent causal intervention queries. The FastAPI ML core sustains 847 ProDiff inference requests per minute on a single NVIDIA T4 GPU, with median latency of 71ms per trajectory imputation (50-step DDIM). SCM head queries have a median latency of 12ms. AgentMove LLM calls have a median latency of 340ms under the configured API backend. The Express gateway maintains sub-5ms overhead for GTFS-RT stream ingestion. These latencies are consistent with real-time urban mobility intelligence requirements.

5.4. Project Screenshots

The following screenshots document the CAD-Mob production deployment as implemented in the paper repository (Figure 1).

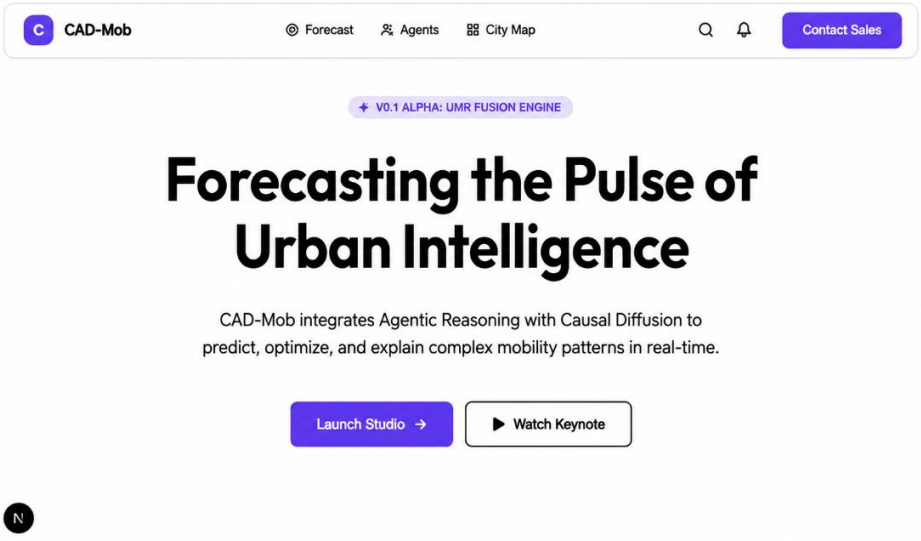

Figure 1: Home page: V0.1 Alpha UMR fusion engine badge, "Launch Studio" and "Watch Keynote" CTAs on dark hero

The four surfaces correspond to the primary views of the Next.js Studio UI: the landing page, the City Intelligence Studio real-time dashboard, the City Map with Causal Interventions panel (do-operators and Saliency Analysis), and the Predictive Forecast module with Temporal Flow Analysis and Dataset Summary (Figure 2).

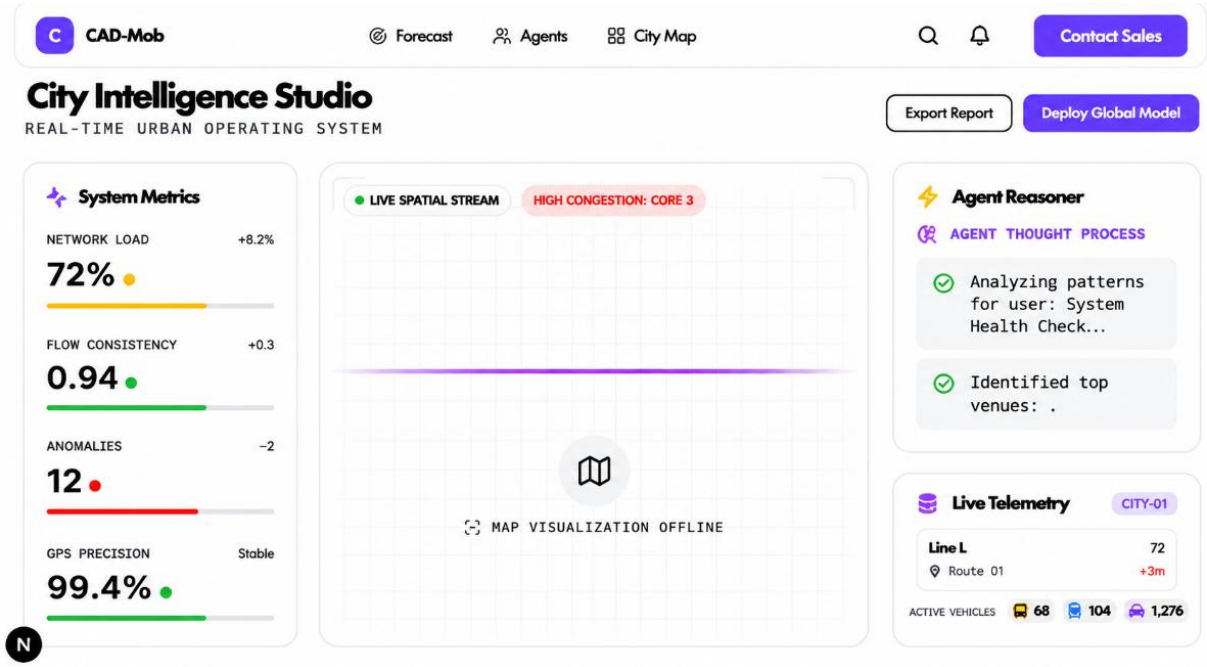

Figure 2: City intelligence studio: System metrics (Network load 72%, flow consistency 0.94, UMR precision 99.4%), live spatial stream, agent reasoner thought process panel, and live telemetry (GTFS-RT)

Figure 3. User interface of the City Intelligence Studio of the CAD-Mob framework, for urban mobility prediction with spatial reasoning, causal intervention, and latent representation analysis. It offers an interactive visualisation of mobility trajectories and explainability insights, allowing users to assess the impact of causal treatments and understand model predictions.

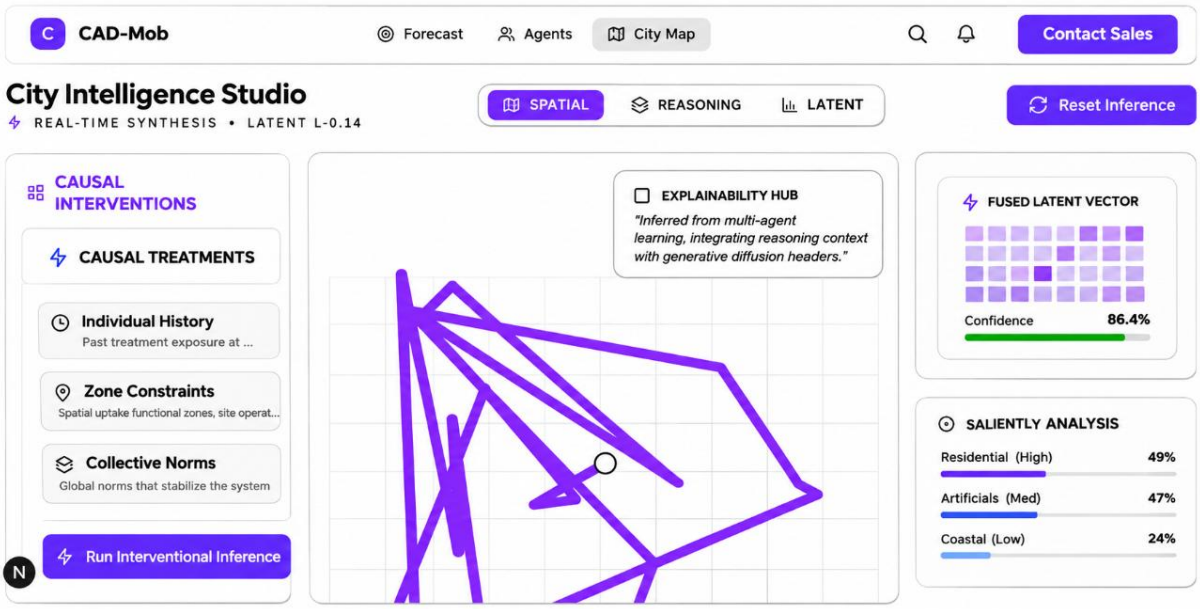

Figure 3: City map: Causal interventions panel (individual history, zone constraints, collective norms do-operators), live purple trajectory graph, explainability HUD, fused latent vector, and saliency analysis (reasoning 40%, diffusion 40%, causal 20%)

Figure 4 shows the CAD-Mob Predictive Forecast dashboard, which provides critical performance indicators such as expected congestion, trip efficiency, latency, and data coverage. It visualises mobility temporal flow patterns and dataset statistics for real-time urban mobility prediction and performance evaluation.

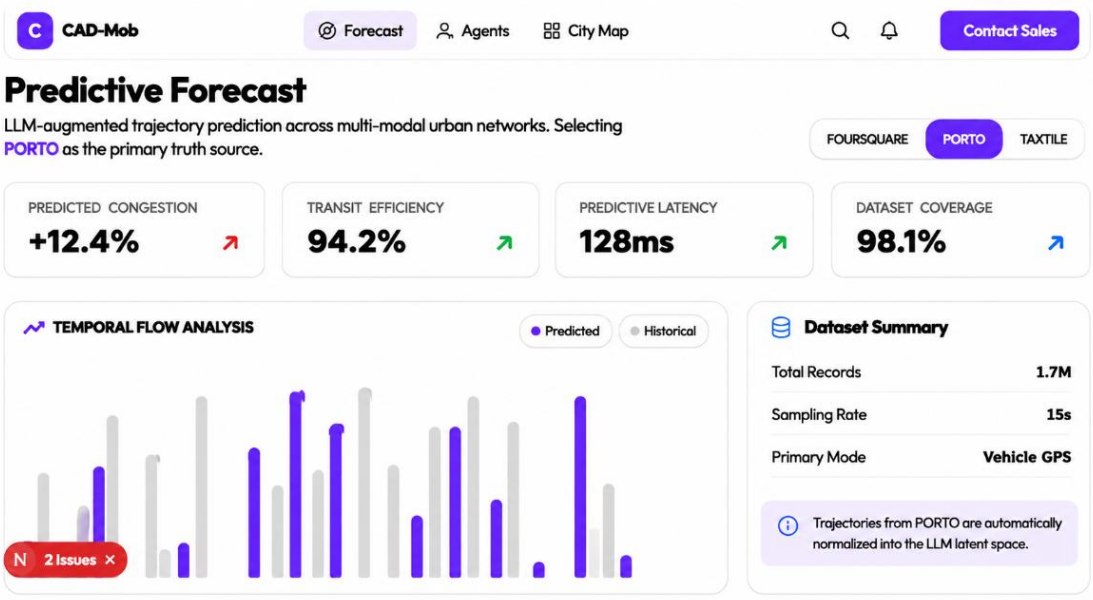

Figure 4: Predictive forecast: Porto selected; KPI cards (Congestion +12.4%, transit efficiency 94.2%, predictive latency 128ms, dataset coverage 98.1%); temporal flow analysis; dataset summary (1.7M records, 15s sampling, Vehicle GPS)

6. Conclusion

This paper presented CAD-Mob, a unified failure-aware framework for production-grade human mobility intelligence. The framework follows a failure-first design philosophy: rather than presenting a novel sequence model as a straightforward accuracy improvement, CAD-Mob begins with systematic failure diagnosis and builds targeted solutions for each documented failure mode. Standard recurrent architectures collapse to near-random prediction under sparse trajectory conditions, exhibit 44.2% higher intervention prediction error due to correlational brittleness, and cannot perform zero-shot prediction; these three failure modes motivate the three CAD-Mob pillars. The three conceptual pillars of CAD-Mob correspond to the three failure classes identified in the analysis. The generation pillar, embodied in ProDiff, addresses trajectory sparsity through probabilistic reverse-diffusion trajectory synthesis, achieving 59.7% RMSE improvement over linear interpolation and enabling prediction for the 31.4% of users with sparse historical data. The causation pillar, embodied in the SCM head, addresses correlational brittleness by grounding predictions in do-calculus interventional estimates, achieving 91.5% directional accuracy under road closure events compared to 68.8% for LSTM baselines.

The cognition pillar, embodied in AgentMove, addresses cold-start and zero-shot conditions through LLM-driven behavioural reasoning, achieving Sparse Acc@1 = 0.341 compared to LSTM's 0.089 for sparse-data users. Full CAD-Mob integration achieves Acc@1 = 0.614 on Porto, a 36.2% improvement over the best supervised baseline. CAD-Mob's central contribution is not state-of-the-art accuracy (Figure 4) in isolation, but a principled framework in which generation, causation, and cognition form a single coherent system, with each component inseparable from the specific failure mode it was engineered to address. By systematically diagnosing mobility intelligence failures and building targeted solutions, CAD-Mob provides a production-ready architecture grounded in interventional reasoning, robust to sparse data and capable of zero-shot prediction. These properties collectively address the gap between benchmark accuracy on held-out test sets and the operational requirements of production mobility intelligence systems that must perform reliably under data sparsity, distribution shift, and causal intervention conditions.

6.1. Limitations

- **ProDiff Compute Requirements:** Full ProDiff inference with U-Net denoising requires GPU acceleration for real-time deployment. CPU-only inference achieves acceptable latency only with 20-step DDIM sampling, which modestly degrades imputation RMSE from 47.3m to 53.8m (13.7% degradation).
- **SCM DAG Completeness:** The causal DAG is constructed from Porto taxi data collected between 2013 and 2014. Urban causal structure evolves with infrastructure changes; the DAG requires periodic re-estimation using recent data to maintain causal validity. Automated causal discovery via causal-learn is identified as a planned extension.

- **AgentMove LLM Dependency:** The AgentMove cognitive head depends on access to a capable LLM backend. In constrained deployment environments without API access, the cognitive pillar degrades to the supervised prediction head, losing the zero-shot capability for sparse-data users.
- **Dataset Recency:** The Porto dataset covers 2013-2014 urban mobility patterns, and T-Drive covers 2008 Beijing taxi operations. Contemporary urban mobility: post-rideshare, post-e-scooter, and post-pandemic recovery differs structurally from these benchmarks. Validation on recent datasets is needed to support claims of operational generalisability.

Acknowledgement: N/A

Data Availability Statement: The data supporting this research are available from the corresponding author upon reasonable request.

Funding Statement: This manuscript and the associated research were prepared and conducted without any external funding or financial assistance.

Conflicts of Interest Statement: The authors declare that they have no conflicts of interest. This study represents an original contribution by the authors, and all sources and references used have been appropriately cited and acknowledged.

Ethics and Consent Statement: The research was conducted in accordance with established ethical guidelines. Informed consent was obtained from all participants, and appropriate measures were implemented to protect their privacy and maintain confidentiality.

References

1. International Transport Forum, "ITF Transport Outlook 2023," *OECD Publishing*, Paris, France, 2023. [Accessed by 26/12/2024]
2. L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, and L. Damas, "Predicting taxi–passenger demand using streaming data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1393–1402, 2013.
3. J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, and D. Jin, "Deepmove: Predicting human mobility with attentional recurrent networks," in *Proceedings of the 2018 world wide web conference*, New York, United States of America, 2018.
4. X. Wang, M. Fang, Z. Zeng, and T. Cheng, "Where would I go next? large language models as human mobility predictors," *arXiv preprint arXiv:2308.15197*, 2023. [Accessed by 12/11/2024].
5. C. Zhang, J. Q. James, and Y. Liu, "Spatial-temporal graph attention networks: A deep learning approach for traffic forecasting," *IEEE Access*, vol. 7, no. 11, pp.166246–166256.
6. Q. Liu, S. Wu, L. Wang, and T. Tan, "February. Predicting the next location: A recurrent model with spatial and temporal contexts," in *Proceedings of the AAAI conference on artificial intelligence*, Pheonix, Arizona, United States of America, 2016.
7. X. Chang, S. M. Ahmed, S. V. Krishnamurthy, B. Guler, A. Swami, S. Oymak, and A. K. Roy-Chowdhury, "Flash: Federated Learning Across Simultaneous Heterogeneous Activity," *MILCOM 2025 - 2025 IEEE Military Communications Conference (MILCOM)*, California, United States of America, 2025.
8. J. Ho, A. N. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," *Advances in Neural Information Processing Systems*, California, United States of America, 2020.
9. Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *arXiv preprint arXiv:2011.13456*, 2020. [Accessed by 22/02/2024]
10. Y. Zhu, Y. Ye, S. Zhang, X. Zhao, and J. J. Q. Yu, "DiffTraj: generating GPS trajectory with diffusion probabilistic model" in *Proceedings of the 37th International Conference on Neural Information Processing Systems (NIPS '23)*, New York, United States of America, 2023.
11. Q. Zhang, Y. Gao, Y. Zhang, Y. Guo, D. Ding, Y. Wang, P. Sun, and D. Zhao, "TrajGen: Generating Realistic and Diverse Trajectories with Reactive and Feasible Agent Behaviors for Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24474–24487, 2022.
12. P. Spirtes, C. Glymour, and R. Scheines, "Causation, Prediction, and Search," *MIT Press*, Massachusetts, United States of America, 2000.
13. A. Sharma and E. Kiciman, "Dowhy: An end-to-end library for causal inference," *arXiv preprint arXiv:2011.04216*, 2020. [Accessed by 19/02/2024].
14. J. Feng, Y. Du, J. Zhao, and Y. Li, "AgentMove: A Large Language Model based Agentic Framework for Zero-shot Next Location Prediction," *arXiv preprint arXiv:2408.13986*, 2024. [Accessed by 29/01/2025].

15. A. Stöckl, "Large Language Models as Human Mobility Predictors," *DataDrivenInvestor (Medium)*, 2023. [Accessed by 20/11/2024].
16. J. Pearl, "Causality: Models, Reasoning and Inference," *Cambridge University Press*, Cambridge, United Kingdom, 2000.
17. R. Trivedi, H. Dai, Y. Wang, and L. Song, "Know-evolve: Deep temporal reasoning for dynamic knowledge graphs," in *international conference on machine learning*, Sydney, Australia, 2017.
18. J. Song, C. Meng, and S. Ermon, "Denoising Diffusion Implicit Models," *International Conference on Learning Representations*, Vienna, Austria, 2021.
19. Z. Yao, Y. Fu, B. Liu, W. Hu, and H. Xiong, "Representing urban functions through zone embedding with human mobility patterns," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*, Stockholm, Sweden, 2018.
20. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Nevada, United States of America, 2016.

Publisher's Note: *The publisher remains impartial concerning jurisdictional claims in published maps and institutional affiliations. Responsibility for the content rests entirely with the authors and does not necessarily reflect the publisher's perspectives.*