MOBILENS: A HUMAN-CENTERED BENCHMARK FOR PERSONALIZED MOBILITY AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models have enabled mobility agents to interpret complex travel intents, use navigation tools, and tailor route and destination recommendations to individual users' needs, preferences, and contexts. However, existing benchmarks provide limited support for jointly assessing behavioral alignment and broader human-centered outcomes under a common evaluation protocol. To address this gap, we introduce MobiLens, a human-centered benchmark built from real-world user queries, pre-query user context, and observed mobility decisions. Inspired by naturalistic decision-making theory, our framework evaluates mobility agents across five human-centered outcome dimensions: personalization, diversity, fairness, explainability, and robustness, alongside process-level measures of tool-use success and operational efficiency. We benchmark 11 agent configurations spanning reasoning, memory augmentation, multi-agent coordination, and recommendation-oriented designs under a shared offline navigation-tool replay protocol. Our results reveal an execution–alignment gap: most agents reliably execute tools and produce format-compliant answers, yet show limited alignment with users' observed route and destination choices. Performance also varies across user groups and mobility scenarios, underscoring the need to evaluate behavioral alignment beyond aggregate execution success. Code and benchmark resources are available at `https://anonymous.4open.science/r/MobiLens-F917`.

1 INTRODUCTION

Recent advances in large language models have transformed navigation systems into conversational mobility agents that can interpret open-ended requests, interact with navigation tools, and make use of users' historical behavior and context (Schick et al., 2023; Patil et al., 2024; Chen et al., 2024a; Zhe et al., 2025). These capabilities enable navigation systems to move beyond finding feasible routes (Dijkstra, 1959; Hart et al., 1968) toward providing mobility services tailored to individual users. Existing benchmarks have made substantial progress in evaluating mobility agents, including tool use, task completion, constraint satisfaction, personalized decision making, and behavior-grounded evaluation (Xie et al., 2024; Song et al., 2026; Bai et al., 2026). However, they provide limited support for jointly assessing behavioral alignment and broader human-centered outcomes across users and mobility scenarios under a common evaluation protocol. This limitation is particularly consequential in mobility, where user decisions depend not only on explicit constraints, such as time or cost, but also on prior experience and situational context. Naturalistic decision-making theory (NDM) studies how people make decisions under uncertainty and dynamic conditions by drawing on prior experience and contextual cues (Klein, 1993), providing a useful perspective for behavior-grounded mobility evaluation. This perspective motivates grounding mobility-agent evaluation in pre-query personalization context and observed mobility decisions. Inspired by this perspective, we argue that evaluating personalized mobility agents should go beyond feasible task completion to assess alignment with individual decision patterns, recommendation diversity, explanation quality, robustness across temporal conditions and query difficulty levels, and disparities in recommendation performance across user groups.

To address these limitations, we introduce MobiLens, a human-centered benchmark that evaluates personalized mobility agents along five complementary dimensions: personalization, diversity, fairness, explainability, and robustness (Figure 1). The benchmark contains 1,608 real-world episodes

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

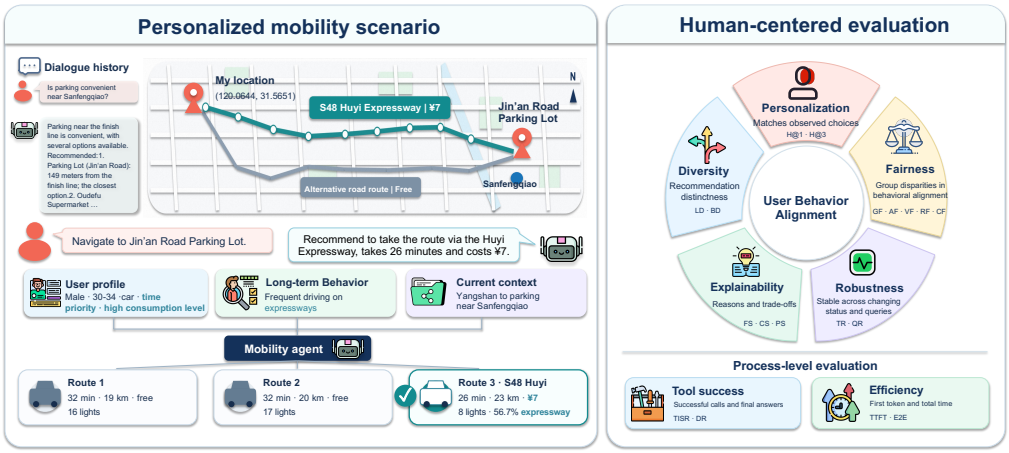


Figure 1: Overview of MobiLens: A Personalized Mobility Scenario and the Human-Centered Evaluation Framework

from 1,458 users, with an average of 400 historical mobility records per episode. Each episode integrates de-identified user attributes, pre-query behavioral history, and request-specific dialogue context as personalization inputs. For POI-related episodes, we use LLM-based matching to associate queries with subsequent user behaviors, whereas route-planning decisions are identified directly from navigation records and click events. The resulting observed mobility decisions serve as behavior-grounded evaluation references rather than being inferred from LLM judgments of agent outputs.

To support reproducible evaluation, we construct a deterministic offline tool environment that serves 19 navigation tools by replaying a frozen cache of API responses, preventing temporal changes in traffic and external services from confounding cross-system comparisons.

Our main contributions are as follows:

- We introduce MobiLens, a human-centered benchmark containing 1,608 real-world episodes from 1,458 users, with de-identified pre-query context as agent inputs and observed post-query mobility decisions as evaluation references.
- We develop a human-centered evaluation framework covering personalization, diversity, fairness, explainability, and robustness, complemented by process-level measures of tool-use success and operational efficiency.
- We benchmark 11 agent configurations spanning reasoning, memory augmentation, multi-agent coordination, and recommendation-oriented designs under a shared offline navigation-tool replay protocol.

Across three LLM backbones, our results reveal a consistent execution–alignment gap: most agents reliably execute tools and produce format-compliant answers, yet show limited behavioral alignment with users' observed route and destination choices. Performance also varies across user groups and mobility scenarios, revealing limitations that aggregate execution metrics alone do not capture.

2 Related Work

2.1 Personalized Mobility Agent Paradigms

LLM-based mobility systems combine language models with symbolic planners to interpret user constraints and support executable search (Yuan et al., 2025; Zhe et al., 2025; Meng et al., 2024), or explore route generation, spatial reasoning, and decision making with LLMs (Chen et al., 2024b; Li et al., 2025; Panagopoulou et al., 2025; Feng et al., 2026; Zhang et al., 2026). Learning-based approaches such as GenMRP instead estimate personalized road costs from road features and user histories before applying shortest-path search (Wang et al., 2026b).

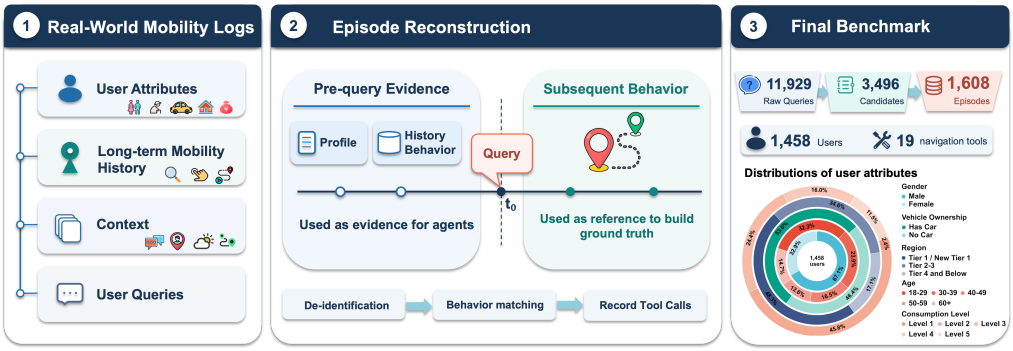


Figure 2: Construction of Behavior-Grounded Evaluation Episodes from Real-World Mobility Logs

General-purpose agents and recommender systems provide reusable designs spanning reasoning, memory augmentation, multi-agent coordination, and recommendation-oriented mechanisms (Yao et al., 2023; Packer et al., 2023; Chen et al., 2026a; Huang et al., 2025; Xu et al., 2025; Wang et al., 2024). However, their evaluation on different tasks and datasets makes it difficult to compare how these designs serve individual users in mobility settings. MobiLens compares configurations spanning these paradigms under a shared offline mobility protocol.

2.2 Benchmarks and Human-Centered Evaluation

Mobility benchmarks assess tool use, constraint satisfaction, intent understanding, spatial reasoning, and task completion (Xie et al., 2024; Chaudhuri et al., 2025; Cheng et al., 2026a;b; Guo et al., 2026). Beyond mobility, personalized-agent benchmarks examine personal-information understanding, preference-aware assistance, and long-term interaction, largely in general recommendation or assistant settings (Qian et al., 2025; Li et al., 2026; Tan et al., 2025; Zhao et al., 2025; Chen et al., 2026b; Duan et al., 2026; Liu et al., 2025; Shang et al., 2026).

Within mobility, MobilityBench provides deterministic API replay over real-world navigation queries (Song et al., 2026), while MapSatisfyBench uses behavioral evidence and pre-query information to assess whether map agents recover and satisfy implicit decision factors in underspecified requests (Bai et al., 2026). MobiLens complements these efforts by jointly assessing behavioral alignment and broader human-centered outcomes across agent configurations, user groups, and mobility scenarios.

Transportation research emphasizes equitable accessibility (Fransen, 2019; Martens, 2006; Martens et al., 2012), while research on recommender systems and AI assistants studies diversity (Vargas & Castells, 2011), explanation faithfulness (Chen et al., 2022), robustness (Ovaisi et al., 2022), fairness (Wang et al., 2026a; Deldjoo et al., 2024; Vassøy & Langseth, 2023), and user experience (Hong et al., 2026). These perspectives motivate our multidimensional evaluation of personalized mobility agents, complemented by process-level measures of tool-use success and operational efficiency.

3 Method

MobiLens evaluates personalized mobility agents using real-world navigation queries, pre-query user context, and observed mobility decisions. Figure 2 summarizes the benchmark pipeline.

3.1 Problem Formulation

For each evaluation episode i associated with user u and query time t_i, we consider three sources of personalization information: a **user profile** $a_{u,i}$ containing user attributes and mobility-related information, a **behavior history** $b_{u,i}$ containing prior searches, clicks, routes, and trajectories, and a **request context** z_i containing information available when the query is issued, such as conversation history, navigation state, user location, and weather. All personalization information exposed to the agent is restricted to information available no later than t_i.

Together with the query x_i, cached tool state S_i, and observed user decision y_i, an episode is

$$e_i = (x_i, a_{u,i}, b_{u,i}, z_i, S_i, y_i). \tag{1}$$

During evaluation, the agent interacts with S_i and produces an ordered recommendation list R_i, explanation ξ_i, and tool-interaction trace $\mathcal{L}_i$:

$$(R_i, \xi_i, \mathcal{L}_i) = A^{S_i}(x_i, a_{u,i}, b_{u,i}, z_i). \tag{2}$$

The observed decision y_i and the post-query evidence used to construct it are withheld from the agent and used only for evaluation.

3.2 Behavior-Grounded Data Construction

Data sources. We sample candidate queries from interaction logs of a navigation agent deployed on a major commercial navigation platform in China. Each query is linked to de-identified user information and request context. An LLM-based classifier removes user-independent factual requests. The retained episodes cover POI recommendation, driving, and public-transit planning.

Observed user decisions. For each query, we construct a behavioral reference from subsequent interactions within a fixed post-query observation window. For driving episodes, subsequent navigation trajectories identify the route actually taken; for transit episodes, recorded click events identify the selected plan. For POI episodes, where multiple subsequent interactions may be relevant, we use Qwen3.8-Max-Preview to associate the query with the most relevant subsequent POI behavior using temporal, semantic, and contextual evidence. The model is used only for query–behavior alignment and never scores outputs from the evaluated agents. The resulting alignments undergo quality-control filtering before inclusion.

These observations are behavior-grounded references rather than exhaustive labels of all acceptable recommendations. The observation window, alignment prompts, evidence fields, and filtering procedures are detailed in Appendix A.

Dataset statistics and validation. The construction pipeline reduces 11,929 raw queries to 3,496 candidate episodes and yields 1,608 benchmark episodes from 1,458 users: 450 POI, 686 driving, and 472 transit episodes. Each episode contains approximately 400 pre-query historical mobility records on average. Figure 2 summarizes the user attributes covered by the benchmark, while Appendix A reports detailed filtering and retained-sample statistics.

Because POI references involve LLM-assisted alignment, we additionally validate this stage with a human audit. Two annotators independently judge the Qwen3.8-Max-Preview links on a random sample of 100 retained POI episodes; a link counts as correct only when both annotators judge it correct, and 96% of the sampled links pass this rule (inter-annotator agreement 99/100; protocol in Appendix A).

3.3 Agent Interaction Environment

All tool interactions are executed in a deterministic offline environment backed by cached responses from 19 navigation APIs. Requests are resolved by tool name and canonicalized arguments, and identical requests within the same replay state receive identical responses. Before evaluation, the response store is frozen as a versioned, read-only snapshot; no live API calls or cache updates are permitted during scored runs. Requests absent from the snapshot are recorded as cache misses and counted as failed invocations in TISR.

This design prevents changes in traffic, POI data, and external services from confounding cross-system comparisons. Details of the tool suite, cache construction, coverage, and resolution statistics are provided in Appendix D.11.

3.4 Human-Centered Evaluation Framework

We evaluate personalization, diversity, fairness, explainability, and robustness, together with process-level tool-use success and operational efficiency. Detailed scoring rules are provided in Appendix C.

3.4.1 TOOL-USE SUCCESS AND OPERATIONAL EFFICIENCY

Tool Invocation Success Rate (TISR) is the mean per-episode success rate over episodes with at least one agent-initiated call, including navigation and framework-local operations. Errors, cache misses, and missing or malformed responses count as failures; navigation-only replay statistics are reported separately in Appendix D.11. **Delivery Rate (DR)** is the fraction of episodes for which the system produces a compliant final answer, treating crashes, empty outputs, format violations, and apology-only refusals as failures. We further report **Time to First Token (TTFT)** and **End-to-End Latency (E2E)** for operational efficiency. Implementation details are provided in Appendix C.

3.4.2 PERSONALIZATION

Personalization measures whether recommendations recover the user's observed mobility decision. Let $\tau_i \in \mathcal{T} = \{\text{POI}, \text{driving}, \text{transit}\}$ denote the task type, and let $m_{\tau_i}(r, y_i) \in \{0, 1\}$ denote its task-specific matching function. For an ordered recommendation list R_i, the top-K hit is

$$h_i^{(K)} = \mathbf{1}\left[\exists\, 1 \le k \le \min(K, |R_i|) : m_{\tau_i}(r_{ik}, y_i) = 1\right]. \tag{3}$$

Overall Hit@K is the equal-weight macro-average of task-specific hit rates:

$$\text{Hit@}K = \frac{1}{|\mathcal{T}|} \sum_{\tau \in \mathcal{T}} \frac{1}{N_\tau} \sum_{i:\tau_i = \tau} h_i^{(K)}. \tag{4}$$

We report Hit@1 and Hit@3.

POI matching uses canonicalized POI-name and metadata matching. Driving recommendations are matched against the observed trajectory using road-level mileage coverage with threshold 0.8. Transit uses the selected plan index when the original candidate list is preserved; otherwise, a candidate matches when it covers at least half of the observed plan's transit-line set. Full matching rules are provided in Appendix C.

3.4.3 DIVERSITY

Diversity evaluates whether the top-K recommendations provide distinct alternatives; we use $K = 3$ and let $n_i = \min(K, |R_i|)$.

Label Diversity (LD) measures descriptive diversity using the average pairwise cosine distance between recommendation-label embeddings:

$$\text{LD}_i = \frac{2}{n_i(n_i - 1)} \sum_{1 \le p < q \le n_i} \left[1 - \cos(\mathbf{e}(r_{ip}), \mathbf{e}(r_{iq}))\right], \qquad n_i \ge 2. \tag{5}$$

We set $\text{LD}_i = 0$ for $n_i < 2$ and macro-average task-wise LD means.

Behavior-Matching Diversity (BD) measures the non-redundancy of the behavior-matching recommendation, conditioned on a successful Top-3 match. For a matched episode, let $k_i^\star$ denote the rank of the highest-ranked matching recommendation among the first $n_i = \min(3, |R_i|)$ candidates:

$$\text{BD}_i = 1 - \max_{\substack{1 \le j \le n_i \\ j \ne k_i^\star}} \text{sim}_{\text{beh}}(r_{ik_i^\star}, r_{ij}), \qquad n_i \ge 2. \tag{6}$$

For matched episodes with $n_i = 1$, we set $\text{BD}_i = 0$. Here, $\text{sim}_{\text{beh}}(\cdot, \cdot) \in [0, 1]$ is task specific: route-planning tasks use route overlap, while POI recommendation uses normalized POI-feature similarity. BD is averaged over Top-3-matched episodes within each task and then macro-averaged across tasks; we report it together with Hit@3 and the number of matched episodes for each task. If a task contains no Top-3-matched episodes, its task-level BD and the overall macro-averaged BD are reported as N/A.

3.4.4 FAIRNESS

Fairness quantifies disparities in behavioral alignment across user groups. We consider gender, age, vehicle ownership, geographic region, and consumption level, with region operationalized using city tiers. For attribute γ and groups $h \in \mathcal{H}_\gamma$, let $\widehat{p}_{\gamma,h}$ denote task-standardized Hit@3. We define

$$F(\gamma) = 1 - \left(\max_{h \in \mathcal{H}_\gamma} \widehat{p}_{\gamma,h} - \min_{h \in \mathcal{H}_\gamma} \widehat{p}_{\gamma,h} \right). \tag{7}$$

We report gender fairness (GF), age fairness (AF), vehicle-ownership fairness (VF), region fairness (RF), and consumption-level fairness (CF). Higher scores indicate smaller group disparities. Because high parity can coexist with uniformly poor performance, we additionally report absolute and worst-group performance. Standardization, missing-cell handling, and bootstrap uncertainty estimation are described in Appendix C and Appendix D.9.2.

3.4.5 EXPLAINABILITY

We use GLM-5.2 with temperature 0 to evaluate three dimensions of explanation quality. **Faithfulness (FS)** measures whether factual claims such as distance, travel time, traffic conditions, and POI attributes are supported by tool outputs. **Completeness (CS)** measures whether relevant considerations underlying the recommendation are covered. **Persuasiveness (PS)** measures whether the rationale clearly communicates why the recommendation is suitable for the user. Each dimension is scored from 0 to 5 and normalized by dividing by 5. These scores reflect judge-assessed explanation quality rather than direct user understanding or acceptance.

Judge prompts, rubrics, aggregation, and failure handling are provided in Appendix E.4.

3.4.6 ROBUSTNESS

Robustness measures variation in behavioral alignment across temporal conditions and query difficulty levels. For each condition group $g \in \mathcal{G}$, task-specific Hit@3 values are standardized using fixed task weights:

$$\widehat{p}_g^{(3)} = \sum_{\tau \in \mathcal{T}} \omega_\tau \widehat{p}_{g,\tau}^{(3)}, \qquad \omega_\tau \geq 0, \quad \sum_{\tau \in \mathcal{T}} \omega_\tau = 1. \tag{8}$$

Using the overall standardized Hit@3, $\bar{p}^{(3)}$, we define

$$\mathrm{Rob} = \mathrm{clip}\left(1 - \frac{\max_{g \in \mathcal{G}} \widehat{p}_g^{(3)} - \min_{g \in \mathcal{G}} \widehat{p}_g^{(3)}}{\bar{p}^{(3)}}, 0, 1\right), \tag{9}$$

for $\bar{p}^{(3)} > 0$; otherwise the score is N/A.

Temporal Robustness (TR) applies this formula with $\mathcal{G}$ the time-of-day groups. **Query Robustness (QR)** is a retention ratio instead: the standardized Hit@3 on hard-to-parse requests, including ambiguous descriptions, implicit intents, and ASR noise. divided by that on the clearest queries, so its numerator and denominator share the same task weights and the ratio is not confounded by task composition. Both metrics quantify variation across naturally occurring conditions rather than paired counterfactual perturbations. We report condition-level Hit@3 alongside robustness scores; grouping criteria are provided in Appendix C.

3.5 BASELINE SYSTEMS AND INTEGRATION PROTOCOL

We benchmark 11 agent configurations spanning reasoning, memory augmentation, multi-agent coordination, and recommendation-oriented designs under a shared offline navigation-tool replay protocol. All configurations are evaluated on the same episodes, tool environment, output-parsing rules, and behavior-matching criteria.

All configurations share the same request context. Behavioral history is withheld only from the no-memory ReAct baseline; every other configuration, including Plan-and-Execute, receives the same underlying pre-query records while differing in how this information is represented, summarized, retrieved, stored, or incorporated into decision making. The no-memory baseline retains the user profile and request context but removes access to behavioral-history records.

We adapt prompts, tool interfaces, and memory mechanisms to mobility without task-specific fine-tuning; implementation details appear in Appendix E. Additional reference comparisons use recorded production-system responses and last-round tool candidates (Appendices D.2 and D.3).

Table 1: Results on MobiLens with Qwen3.7-Max across five outcome dimensions. All scores are multiplied by 100; higher is better. H@K equally averages hit rates across the three task types (Table 3); BD measures behavior-matching diversity conditional on a Top-3 match. VF/CF denote vehicle-ownership and consumption-level fairness; TR measures temporal stability; QR measures retention on ambiguous or noisy queries. Best scores are bold and second-best underlined. Full results across three backbones appear in Table 2.

	Personalization		**Diversity**		**Fairness**					**Explanation**			**Robustness**	
	H@1	H@3	LD	BD	GF	AF	VF	RF	CF	FS	CS	PS	TR	QR
Qwen3.7-Max														
ReAct (no mem)	31.8	48.1	43.1	49.1	97.0	86.6	83.9	92.7	95.3	91.1	90.9	92.5	87.9	38.3
Plan-and-Execute	30.1	45.4	37.8	**49.7**	**99.0**	83.4	79.4	91.6	84.8	89.8	92.2	93.6	**95.2**	31.3
ReAct (last-k)	33.3	50.0	43.8	48.4	96.6	**92.8**	77.4	92.0	**95.5**	90.0	91.2	92.6	89.2	39.4
ReAct (summary)	35.5	52.7	47.8	49.4	96.4	86.4	84.7	95.4	94.5	89.6	92.8	93.6	90.0	50.2
ReAct (retrieval)	33.2	49.0	44.9	48.8	96.6	88.0	84.1	91.6	94.9	91.6	91.8	93.0	88.5	39.2
Letta (MemGPT)	36.2	53.7	46.5	49.3	96.0	86.8	86.6	94.1	93.0	**92.9**	95.4	95.8	82.4	43.0
MetaGPT	35.5	52.0	47.7	49.5	96.7	86.0	79.9	**96.5**	89.4	92.6	**95.9**	**96.3**	90.5	51.6
CAMEL	**38.4**	**56.8**	**50.3**	**49.7**	96.5	82.3	88.5	94.1	91.6	86.9	93.8	93.9	89.2	**61.6**
InteRecAgent	36.6	53.0	43.0	46.9	94.7	89.4	93.5	95.3	91.0	87.8	92.9	94.1	87.8	59.4
iAgent	37.1	51.9	50.2	47.7	97.2	86.1	82.0	92.7	88.7	76.3	91.6	90.8	80.3	45.3
MACRec	**38.4**	54.9	46.2	45.3	95.6	91.4	**94.6**	94.2	95.2	88.5	91.9	92.8	95.0	56.0

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

Dataset. All configurations are evaluated on the same 1,608 Chinese-language episodes, collected over 19 days, with personalization histories drawn from a three-month pre-query window.

Evaluated Agents. We compare 11 configurations: reasoning baselines (ReAct and Plan-and-Execute), memory-centric agents (three ReAct memory variants and Letta), multi-agent systems (MetaGPT and CAMEL), and recommendation-oriented agents (InteRecAgent, iAgent, and MACRec). We follow the common input and integration protocol described above; framework adaptations are detailed in Appendix E.1.

Implementation. Each configuration uses the shared 19-tool offline environment and is evaluated with Qwen3.7-Max, GPT-5.4, and DeepSeek-V4-Pro at temperature 0, yielding 33 configuration–backbone combinations. Prompts share a navigation template while retaining framework-specific extensions (Appendix E.2).

4.2 OVERALL RESULTS

Table 1 reports results with Qwen3.7-Max; complete results across all three backbones are provided in Appendix D.1. No configuration leads across all five human-centered dimensions. With Qwen3.7-Max, CAMEL achieves the highest Hit@3 and LD, while Plan-and-Execute and CAMEL tie for the highest BD. The highest fairness and explanation scores are attained by other configurations.

The results reveal an execution–alignment gap. Under Qwen3.7-Max, all configurations except Plan-and-Execute achieve TISR above 94% and DR above 96%, yet the highest Hit@3 is 56.8%. High tool-invocation and delivery success therefore coexist with limited alignment with users' observed mobility choices.

4.3 PROCESS-LEVEL PERFORMANCE

Tool invocation succeeds in most cases: TISR exceeds 95% for most configuration–backbone combinations. Delivery rates are also generally high. GPT-5.4 achieves a mean DR of 98.2% across the 11 configurations (Appendix D.1). These process-level results characterize execution reliability, while behavioral alignment additionally depends on the recommendations produced.

Latency differs substantially across agent categories. With Qwen3.7-Max, mean E2E latency increases from 9.86 s for reasoning baselines to 14.30 s for memory-centric agents, 21.64 s for multi-agent systems, and 29.43 s for recommendation-oriented agents. The same category ordering holds for the other two backbones. Across the 11 Qwen3.7-Max configurations, E2E latency is positively correlated with Hit@3 (Spearman $\rho = 0.72$, $p = 0.013$). In this comparison, higher-alignment configurations tend to exhibit greater latency, although the association does not isolate the effect of additional computation.

4.4 Personalization

4.4.1 Comparison across Paradigms

Under Qwen3.7-Max, multi-agent and recommendation-oriented configurations achieve mean Hit@3 scores of 54.4% and 53.3%, respectively, exceeding the reasoning baselines. Their relative strengths differ across tasks. iAgent achieves the highest POI Hit@3 at 30.7%, while CAMEL leads driving and transit Hit@3 at 59.6% and 82.4%, respectively (Appendix D.1). These task-specific differences show why aggregate alignment scores should be accompanied by results for individual mobility scenarios.

4.4.2 Memory Variants

Among the ReAct variants with Qwen3.7-Max, summary memory provides the largest improvement over the no-memory baseline, increasing Hit@3 from 48.1% to 52.7%. Last-k memory reaches 50.0%, while retrieval memory reaches 49.0%. Letta achieves 53.7%, although its agent architecture also differs from ReAct. The comparison among ReAct variants suggests that access to history alone does not determine its benefit; how that history is organized and used also matters.

A request for places in Longde County suitable for a trip with children illustrates these differences. Other memory variants suggested children's play venues, such as indoor amusement parks. ReAct with summary memory instead incorporated the user's prior visits to scenic areas and parks into recommendations for outdoor family activities. This example illustrates how different history representations can lead agents to emphasize different aspects of a user's past behavior in their recommendations.

4.5 Diversity

LD measures descriptive variation across the Top-3 recommendations, whereas BD measures how distinct the behavior-matching recommendation is from the remaining Top-3 recommendations, conditional on a successful Top-3 match. Under Qwen3.7-Max, CAMEL achieves the highest LD (50.3), while Plan-and-Execute ties CAMEL for the highest BD (49.7), despite having lower Hit@3 than CAMEL. The two metrics therefore capture different aspects of recommendation diversity.

Among the memory configurations, summary memory achieves the highest LD (47.8) and BD (49.4). Summary memory exceeds retrieval memory on both LD (47.8 versus 44.9) and BD (49.4 versus 48.8). Because BD is computed on each configuration's successfully matched episodes, comparisons must be interpreted alongside Hit@3 and task-level matched-sample counts. A higher BD does not by itself establish that a system recovers more observed choices or provides more suitable alternatives.

4.6 Fairness

We assess fairness at two levels: task-standardized group differences pooled across 11 frameworks within each backbone, and each group's task-standardized Hit@3 relative to the sample-size-weighted mean of the other groups in individual framework–backbone cells (Appendix D.9.2)

Disparities depend on both user attributes and task type. Car owners significantly outperform non-car owners after task-type standardization. Consumption-level differences show a similar pattern: Level-1 users score significantly below the best-performing group on all three backbones, as do Level-3 users on Qwen3.7-Max and DeepSeek-V4-Pro. Because these gaps persist after accounting

for the mix of POI, driving, and bus episodes, task composition alone does not explain them. The results suggest that agents may serve users with different mobility resources unevenly.

Age reveals why aggregate fairness scores require further scrutiny. Pooled across frameworks, users aged 40–44 and 55–59 score approximately 8–12 percentage points below the best-performing age group on all three backbones ($q < 0.05$). For users aged 60+, however, the direction changes with the task: they score approximately 14–17 points below the mean across age groups in POI episodes, but 13–15 points above it in driving-route episodes ($q < 10^{-4}$ in both cases). An agent can therefore serve an age group well in route planning while underserving the same group in destination recommendation; a single aggregate age gap would obscure this distinction.

Gender and region show directional patterns without comparably strong statistical evidence. Female users score higher than male users in 31 evaluated configurations, and Tier 2–3 city users have lower relative Hit@3 in 32 of 33 configurations. Yet the gender differences are small, and no individual regional contrast survives BH-FDR correction.

4.7 Explainability

Explanation scores vary systematically across backbones. Averaged across configurations, GPT-5.4 achieves the highest scores in faithfulness, completeness, and persuasiveness, followed by Qwen3.7-Max and DeepSeek-V4-Pro (Appendix Table 2).

We summarize explanation quality as $\mathrm{EQ} = (\mathrm{FS} + \mathrm{CS} + \mathrm{PS})/3$. Across the 33 configuration–backbone combinations, EQ is weakly correlated with both Hit@1 (Spearman $\rho = 0.04$) and Hit@3 ($\rho = 0.20$). High judge-assessed explanation quality therefore does not consistently coincide with stronger agreement with observed mobility choices. These results motivate reporting explanation quality alongside behavioral alignment as complementary aspects of agent performance.

4.8 Robustness

TR measures variation across time-of-day groups; QR measures the retention of task-standardized Hit@3 on hard-to-parse requests relative to clear requests. They characterize performance across naturally occurring conditions rather than responses to controlled perturbations.

The two dimensions produce different rankings. With Qwen3.7-Max, Plan-and-Execute achieves the highest TR (95.2), but its QR is 31.3. CAMEL achieves the highest QR (61.6), with a TR of 89.2. Thus, stability across time-of-day groups does not necessarily accompany performance retention on hard-to-parse requests, which include underspecified requests and ASR noise.

Both scores should be interpreted alongside condition-level Hit@3. A high score on either metric does not by itself imply high behavioral alignment.

5 Conclusion

We introduce MobiLens, a human-centered benchmark for personalized mobility agents built from real-world navigation queries, pre-query user context, and observed mobility decisions. Across 11 agent configurations and three LLM backbones, our results reveal an execution–alignment gap: most agents achieve high tool-invocation and delivery success, yet show limited alignment with users' observed mobility choices. Performance also varies across human-centered dimensions, user groups, and mobility scenarios, showing that execution success alone provides an incomplete view of personalized mobility-agent performance.

The benchmark has several limitations. Observed mobility decisions provide behavioral references but do not exhaust all recommendations that may be suitable for a user. LLM-assisted POI reference construction may still introduce alignment errors despite validation against independent human annotations on a random sample. The data are drawn from a single navigation platform and Chinese-language interactions, which may limit generalizability. Future work will extend MobiLens to broader mobility settings and investigate agent designs that better use user history and context to improve behavioral alignment and robustness while mitigating observed performance disparities across user groups.

AI USE STATEMENT

We used generative AI tools to assist with agent-framework adaptation, evaluation-code development, data cleaning and reformatting, organization of experimental outputs, manuscript editing, and reference formatting. These tools also provided feedback on evaluation methodology and the interpretation of results. LLM-based query filtering, query–behavior alignment, and explanation-quality assessment are explicitly described in the Method section and accompanying appendices as part of the benchmark methodology. The authors reviewed and verified all AI-assisted work and take full responsibility for the final manuscript, code, data processing, and reported findings.

ETHICS STATEMENT

MobiLens is constructed from existing, de-identified interaction logs of a commercial navigation platform in China. No additional end-user interactions were collected for this study. The de-identification procedures and the scope of information included in the planned release are described in Appendix B. Raw operational logs and direct personal identifiers are excluded from the release.

Group-level analyses characterize disparities in behavioral alignment across user groups. These findings are reported alongside absolute and worst-group performance and do not, by themselves, establish causal discrimination. Independent human annotations are used to assess the reliability of LLM-assisted POI query–behavior alignment on a random sample.

REPRODUCIBILITY STATEMENT

We plan to release the 1,608 de-identified benchmark episodes, the frozen navigation-tool response cache, the evaluation harness, agent-framework adaptations, scoring code, prompts, configurations, and a data card. The Method section and Appendix C specify the evaluation metrics, task-specific matching rules, and aggregation procedures. Data construction and processing are documented in Appendix A and Appendix B, while Appendix D.11 describes cache construction, tool coverage, and request-resolution behavior. The versioned, read-only cache fixes navigation-tool responses during evaluation, although LLM-generated outputs may still vary across runs.

REFERENCES

Lubin Bai, Mengyu Cao, Sixue Wang, Zhongwei Wan, Yue Pan, Jiale Hou, Xiang Li, and Xiuyuan Zhang. Mapsatisfybench: Benchmarking satisfaction-aware map agents through behavior-grounded implicit decision factors. *arXiv preprint arXiv:2606.17453*, 2026.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 01 1995. ISSN 0035-9246. doi: 10.1111/j.2517-6161.1995.tb02031.x. URL https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.

Soumyabrata Chaudhuri, Pranav Purkar, Ritwik Raghav, Shubhojit Mallick, Manish Gupta, Abhik Jana, and Shreya Ghosh. Tripcraft: A benchmark for spatio-temporally fine grained travel planning. *arXiv preprint arXiv:2502.20508*, 2025.

Aili Chen, Xuyang Ge, Ziquan Fu, Yanghua Xiao, and Jiangjie Chen. Travelagent: An ai assistant for personalized travel planning. *arXiv preprint arXiv:2409.08069*, 2024a.

Wanyi Chen, Meng-Wen Su, Nafisa Mehjabin, and Mary L Cummings. Can llms plan paths in the real world? *arXiv preprint arXiv:2411.17912*, 2024b.

Weixin Chen, Yuhan Zhao, Jingyuan Huang, Zihe Ye, Mingxuan Ju, Tong Zhao, Neil Shah, Li Chen, and Yongfeng Zhang. Memrec: Collaborative memory-augmented agentic recommender system. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 44515–44544, 2026a.

Xu Chen, Yongfeng Zhang, and Ji-Rong Wen. Measuring "why" in recommender systems: A comprehensive survey on the evaluation of explainable recommendation. *arXiv preprint arXiv:2202.06466*, 2022. URL https://arxiv.org/abs/2202.06466.

Yuxin Chen, Yi Zhang, Zhengzhou Cai, Yaorui Shi, Zhiyuan Yao, Chenhang Cui, Jingnan Zheng, Yaqi Huo, Xi Su, Qi Gu, et al. Vitabench 2.0: Evaluating personalized and proactive agents in long-term user interactions. *arXiv preprint arXiv:2605.27141*, 2026b.

Xiang Cheng, Yulan Hu, Xiangwen Zhang, Lu Xu, Lide Tan, Zheng Pan, Xin Li, and Yong Liu. Beyond itinerary planning—a real-world benchmark for multi-turn and tool-using travel tasks. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29200–29251, 2026a.

Xiang Cheng, Yulan Hu, Lulu Zheng, Zheng Pan, Xin Li, and Yong Liu. Grouptravelbench: Benchmarking llm agents on multi-person travel planning. *arXiv preprint arXiv:2605.25200*, 2026b.

Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogin, Alessandro Difonzo, and Dario Zanzonelli. Fairness in recommender systems: research landscape and future directions: Y. deldjoo et al. *User Modeling and User-Adapted Interaction*, 34(1):59–108, 2024.

Edsger W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959. doi: 10.1007/BF01386390. URL https://doi.org/10.1007/BF01386390.

Feiyu Duan, Xuan-Jing Huang, and Zhongyu Wei. Lifesim: Long-horizon user life simulator for personalized assistant evaluation. *arXiv preprint arXiv:2604.07720*, 2026.

Sicheng Feng, Song Wang, Shuyi Ouyang, Lingdong Kong, Zikai Song, Jianke Zhu, Huan Wang, and Xinchao Wang. Reasonmap: Towards fine-grained visual reasoning from transit maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 41077–41088, 2026.

Koos Fransen. Transport justice: Designing fair transportation systems. *Transfers*, 9(2):119–121, 2019.

Hanyu Guo, Jiedong Yang, Chao Chen, Longfei Xu, Kaikui Liu, and Xiangxiang Chu. Transitlm: A large-scale dataset and benchmark for map-free transit route generation. *arXiv preprint arXiv:2605.22355*, 2026.

Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

Mengze Hong, Xia Zeng, Zeyang Lei, Sheng Wang, Chen Jason Zhang, Di Jiang, Taiming Fu, Jinfeng Huang, Mengqiao Liu, Qinghe Chang, et al. Uxbench: Benchmarking user experience in ai assistants. *arXiv preprint arXiv:2606.09570*, 2026.

Xu Huang, Jianxun Lian, Yuxuan Lei, Jing Yao, Defu Lian, and Xing Xie. Recommender ai agent: Integrating large language models for interactive recommendations. *ACM Transactions on Information Systems*, 43(4):1–33, 2025.

Gary A Klein. A recognition-primed decision (rpd) model of rapid decision making. In *Decision making in action: Models and methods*, pp. 138–147. Ablex Publishing Corporation, 1993.

Kechen Li, Yaotian Tao, Ximing Wen, Quanwei Sun, Zifei Gong, Chang Xu, Xizhe Zhang, and Tianbo Ji. Gridroute: A benchmark for llm-based route planning with cardinal movement in grid environments. *arXiv preprint arXiv:2505.24306*, 2025.

Stella Li, Avinandan Bose, Faeze Brahman, Simon Du, Pang Wei Koh, Maryam Fazel, and Yulia Tsvetkov. Prefdisco: Benchmarking proactive personalized reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 80844–80908, 2026.

Qijiong Liu, Jieming Zhu, Lu Fan, Kun Wang, Hengchang Hu, Wei Guo, Yong Liu, and Xiao-Ming Wu. Can llms outshine conventional recommenders? A comparative evaluation. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/129292da836f35bc051ccef07a245ab7-Abstract-Datasets_and_Benchmarks_Track.html.

Karel Martens. Basing transport planning on principles of social justice. *Berkeley Planning Journal*, 19(1), 2006.

Karel Martens, Aaron Golub, and Glenn Robinson. A justice-theoretic approach to the distribution of transportation benefits: Implications for transportation planning practice in the united states. *Transportation research part A: policy and practice*, 46(4):684–695, 2012.

Silin Meng, Yiwei Wang, Cheng-Fu Yang, Nanyun Peng, and Kai-Wei Chang. Llm-a*: Large language model enhanced incremental heuristic search on path planning. *arXiv preprint arXiv:2407.02511*, 2024.

Zohreh Ovaisi, Shelby Heinecke, Jia Li, Yongfeng Zhang, Elena Zheleva, and Caiming Xiong. RGRecSys: A toolkit for robustness evaluation of recommender systems. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining (WSDM '22)*, New York, NY, USA, 2022. ACM. doi: 10.1145/3488560.3502192. URL https://doi.org/10.1145/3488560.3502192. arXiv:2201.04399.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G Patil, Ion Stoica, and Joseph E Gonzalez. Memgpt: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Artemis Panagopoulou, Aveek Purohit, Achin Kulshrestha, Soroosh Yazdani, and Mohit Goyal. Maptrace: Scalable data generation for route tracing on maps. *arXiv preprint arXiv:2512.19609*, 2025.

Shishir G Patil, Tianjun Zhang, Xin Wang, and Joseph E Gonzalez. Gorilla: Large language model connected with massive apis. *Advances in Neural Information Processing Systems*, 37:126544–126565, 2024.

Cheng Qian, Zuxin Liu, Akshara Prabhakar, Zhiwei Liu, Jianguo Zhang, Haolin Chen, Heng Ji, Weiran Yao, Shelby Heinecke, Silvio Savarese, et al. Userbench: An interactive gym environment for user-centric agents. *arXiv preprint arXiv:2507.22034*, 2025.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551, 2023.

Yu Shang, Peijie Liu, Yuwei Yan, Zijing Wu, Leheng Sheng, Yuanqing Yu, Chumeng Jiang, An Zhang, Fengli Xu, Yu Wang, et al. Agentrecbench: Benchmarking llm agent-based personalized recommender systems. *Advances in Neural Information Processing Systems*, 38, 2026.

Zhiheng Song, Jingshuai Zhang, Chuan Qin, Chao Wang, Chao Chen, Longfei Xu, Kaikui Liu, Xiangxiang Chu, and Hengshu Zhu. Mobilitybench: A benchmark for evaluating route-planning agents in real-world mobility scenarios. *arXiv preprint arXiv:2602.22638*, 2026.

Juntao Tan, Liangwei Yang, Zuxin Liu, Zhiwei Liu, Rithesh RN, Tulika Manoj Awalgaonkar, Jianguo Zhang, Weiran Yao, Ming Zhu, Shirley Kokane, et al. Personabench: Evaluating ai models on understanding personal information through accessing (synthetic) private user data. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 878–893, 2025.

Saúl Vargas and Pablo Castells. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the Fifth ACM Conference on Recommender Systems (RecSys '11)*, pp. 109–116, New York, NY, USA, 2011. ACM. doi: 10.1145/2043932.2043955. URL https://doi.org/10.1145/2043932.2043955.

Bjørnar Vassøy and Helge Langseth. Consumer-side fairness in recommender systems: a systematic survey of methods and evaluation. *arXiv preprint arXiv:2305.09330*, 2023.

Bohao Wang, Yu Cui, Zhenxiang Xu, Jujia Zhao, Chenxiao Fan, Jizhi Zhang, Weiqin Yang, Shengjia Zhang, Sirui Chen, Yang Zhang, et al. Trustworthy recommendation in the era of large language models: Opportunities and challenges. *arXiv preprint arXiv:2606.00540*, 2026a.

Chengzhang Wang, Chao Chen, Jun Tao, Tengfei Liu, He Bai, Song Wang, Longfei Xu, Kaikui Liu, and Xiangxiang Chu. Genmrp: A generative multi-route planning framework for efficient and personalized real-time industrial navigation. *arXiv preprint arXiv:2602.04174*, 2026b.

Zhefan Wang, Yuanqing Yu, Wendi Zheng, Weizhi Ma, and Min Zhang. Macrec: A multi-agent collaboration framework for recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2760–2764, 2024.

Jian Xie, Kai Zhang, Jiangjie Chen, Tinghui Zhu, Renze Lou, Yuandong Tian, Yanghua Xiao, and Yu Su. Travelplanner: A benchmark for real-world planning with language agents. *arXiv preprint arXiv:2402.01622*, 2024.

Wujiang Xu, Yunxiao Shi, Zujie Liang, Xuying Ning, Kai Mei, Kun Wang, Xi Zhu, Min Xu, and Yongfeng Zhang. iagent: Llm agent as a shield between user and recommender systems. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18056–18084, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Liangqi Yuan, Dong-Jun Han, Christopher G Brinton, and Sabine Brunswicker. Llmap: Llm-assisted multi-objective route planning with user preferences. *arXiv preprint arXiv:2509.12273*, 2025.

Wenjie Zhang, Chen Yang, Xin Lu, Zhen Wang, Yue Liu, Bobo Xi, and Pengbo Zhang. Smap: Semantic route planning with map-grounded multimodal alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40108–40118, 2026.

Zheng Zhao, Clara Vania, Subhradeep Kayal, Naila Khan, Shay B Cohen, and Emine Yilmaz. Personalens: A benchmark for personalization evaluation in conversational ai assistants. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18023–18055, 2025.

Tao Zhe, Rui Liu, Fateme Memar, Xiao Luo, Wei Fan, Xinyue Ye, Zhongren Peng, and Dongjie Wang. Constraint-aware route recommendation from natural language via hierarchical llm agents. *arXiv preprint arXiv:2510.06078*, 2025.

A Dataset Statistics

Dataset composition. The benchmark contains 1,608 episodes (450 POI, 686 driving, 472 transit) drawn from 1,458 users, and every evaluated configuration is scored on all of them.

Construction details. The stage-by-stage filtering counts, the three-day post-query observation window, and the second collection window are documented in Appendix B; the behavioral-evidence requirement for POI alignment and the released alignment prompts are in Appendix E.4.

Human validation of POI alignment. Because only POI references rely on LLM-assisted query–behavior alignment (driving and transit references are constructed by deterministic rules), we validate this stage with a human audit. We draw a random sample of 100 retained POI episodes. Two

annotators independently examine, for each sampled episode, the Qwen3.8-Max-Preview query–behavior link and judge whether it correctly associates the user's request with the observed post-query behavior; each annotator records a binary correct/incorrect verdict per episode. The two annotators agree on 99 of the 100 sampled episodes. Aggregating the two verdicts per episode (a link counts as correct only when both annotators judge it correct), 96% of the sampled links are judged correct. This audit evaluates the correctness of retained POI links; it does not estimate the recall of the construction pipeline, and it does not calibrate the explanation-quality judge of Appendix E.4, which is a separate evaluation-time component.

B Data Construction and Ethics

Construction funnel. The episodes are produced in three stages. First, the 11,929 raw queries are cleaned (removing empty utterances, internal test accounts, and sessions lacking navigation identifiers) and classified by an LLM to filter out user-independent factual requests, leaving 3,496 user-dependent candidates. Second, the LLM-based query–behavior alignment described in the main text removes composite behaviors that are not cleanly classifiable and retains single-category driving and POI episodes after behavioral-evidence filtering. Third, transit episodes are constructed independently by deterministic click-log rule matching, and a second collection window of two additional days contributes 390 driving episodes constructed by the same deterministic rules. Every ground truth is drawn from a three-day post-query observation window. Episodes whose ground-truth labels are flagged as unreliable are dropped at the end of construction, so the benchmark comprises 450 POI, 686 driving, and 472 transit episodes, that is 1,608 in total, sampled over 19 days in two non-contiguous windows.

De-identification. The benchmark is constructed from logs of a production navigation platform. Data collection and use follow the platform's terms of service and privacy policy, and all processing is performed on de-identified records. Before construction, direct identifiers (user and device IDs) are removed, coordinates are truncated to three decimal places (two for sensitive locations), and timestamps are generalized to day type and time period, preserving temporal context without revealing absolute dates. The fairness attributes are limited to aggregated demographic bins (age group, gender, vehicle ownership, consumption level, city tier), and no free-text profile fields are used.

Release scope. We plan to release the 1,608 benchmark episodes with the anonymized fields required for evaluation, the cached tool responses for deterministic replay, and the evaluation code; raw platform logs and users' full behavioral histories are not released. We acknowledge that even truncated mobility records may carry residual re-identification risk; the generalization above, together with the exclusion of raw logs and the restriction of released episodes to the behavioral window linked to each query rather than unbounded histories, is intended to keep this risk low.

C Evaluation Dimensions

Tool invocation and delivery. The Tool Invocation Success Rate averages the per-episode success rate of agent-initiated tool calls, and the Delivery Rate is the proportion of episodes that deliver a compliant answer:

$$\begin{aligned} \mathrm{TISR} &= \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|T_i|}\sum_{t\in T_i}\mathbf{1}\left[\mathrm{success}(t)\right], \\ \mathrm{DR} &= \frac{1}{M}\sum_{i=1}^{M}\mathbf{1}\left[\epsilon_i = \varnothing \wedge a_i \neq \varnothing \wedge f_i = 0 \wedge r_i = 0\right], \end{aligned} \tag{10}$$

where T_i is the set of tool calls in episode i and N counts only episodes that issue at least one tool call (a configuration issuing none receives N/A), while M counts all episodes; ϵ_i is the episode-level error (crash or timeout), a_i the final answer, and f_i and r_i are binary flags for a format violation and an apology-only refusal (an answer that provides no candidate). TISR is therefore an episode-averaged, configuration-level rate over all agent-initiated calls, including framework-local memory and profile operations when present; it is neither a pooled call-level rate nor the sandbox cache-resolution rate of Appendix D.11. All LLM clients use a 120-second timeout with at most 5 retries;

episodes that still fail after retries are recorded as errors, and their elapsed wall-clock time is included in E2E.

POI name matching. A recommended POI counts as matched if its name hits a ground-truth POI name under a five-level fuzzy waterfall: bidirectional substring, sequence ratio ≥ 0.75, longest-common-substring ratio ≥ 0.70, core-token substring, and a discriminative sliding window. Every level is gated by an identity-evidence check, so sharing a facility or brand term (a charging-station brand, a fuel retailer) does not by itself establish that two records denote the same entity.

Route coverage and route hits. Road names are normalized by stripping ramp-variant suffixes before comparison. For transit episodes, a candidate whose plan index equals the user's clicked plan (a rank match) scores 1.0; otherwise the line-mileage share of the user's picked transit plan covered by the candidate supplies the continuous transit RC@K score. The main-table Hit@K applies the task-type-specific thresholds of the main text, with two implementation details: for driving, coverage is taken as the more favorable of two comparisons, the order-preserving distance-weighted longest common subsequence and the unordered distance coverage; for transit, the line-set criterion replaces the rank criterion when the replay does not preserve the original plan index (this path decides 8.2% of the pooled transit hits across the 33 agent runs). The continuous RC@K is retained as a diagnostic and reported in Appendix D.12; its denominator includes only episodes that delivered at least one comparable route candidate, since episodes without any candidate are counted as non-delivered by DR rather than as zero coverage.

Diversity. Label embeddings are produced by BAAI bge-small-zh-v1.5. The Label Diversity denominator is the set of episodes with comparable ground truth, and an episode that yields fewer than two labels scores zero rather than being dropped, so that recommending a single candidate is not rewarded by exclusion. The behavioral similarity kernel $\mathrm{sim}_{\mathrm{beh}}$ of behavior-matching diversity is instantiated per task type: for route episodes, with $\mathcal{L}_i^j$ the road segments of the j-th recommended route and $d(e)$ the segment length, it is the length-weighted route-overlap kernel $\mathrm{IoU}(\mathcal{L}_i^j, \mathcal{L}_i^l) = \sum_{e \in \mathcal{L}_i^j \cap \mathcal{L}_i^l} d(e) / \sum_{e \in \mathcal{L}_i^j \cup \mathcal{L}_i^l} d(e)$; for POI episodes it is computed over the candidates' attribute fields (category and tags, distance and detour, price), averaging a type-hierarchy and tag-overlap layer with a numeric layer over the distance and price fields.

Fairness standardization. Group hit rates are standardized over task types before any gap is computed; the definition and the task weights are given in Appendix D.9.1. Groups with fewer than 30 episodes are excluded from the gap to avoid small-sample noise, and episodes whose attribute is unavailable are likewise excluded from the corresponding dimension. Attribute values are read from the benchmark's fixed user-profile snapshot, so every configuration is evaluated on the same population strata. Gaps are reported as absolute percentage-point differences, and the resulting confound with overall accuracy is analyzed for CF in Appendix D.9.

Robustness details. TR partitions episodes into five traffic periods: early peak $[6, 9)$, daytime off-peak $[9, 17)$, evening peak $[17, 20)$, night $[20, 24)$, and late night $[0, 6)$. Periods containing fewer than 25 episodes are excluded. For each retained period, we compute task-standardized Hit@3 using the same fixed task weights as in the main evaluation. TR is then computed from the relative gap between the maximum and minimum standardized Hit@3 across retained periods, normalized by the overall standardized Hit@3 and clipped to $[0, 1]$. QR is the retention ratio of the same standardized Hit@3 between the hard levels (L4∪L5, 213 episodes) and level-1 queries, so numerator and denominator share the task weights and the ratio is not confounded by task composition.

D Detailed Main-Result Analyses

D.1 Complete Results and Per-Type Breakdown

Table 1 in the main text reports the Qwen3.7-Max backbone over the five outcome dimensions (personalization, diversity, fairness, explanation, robustness). Table 2 below adds the tool-invocation (TISR, DR) and efficiency (TTFT, E2E) dimensions and extends the grid to all three backbones.

Table 3 decomposes the personalization scores from Table 2 by the three ground-truth task types (POI / driving / transit).

Table 2: Full main results on MobiLens over all three LLM backbones (Qwen3.7-Max / GPT-5.4 / DeepSeek-V4-Pro): eighteen metrics per configuration (tool invocation TISR/DR, efficiency TTFT/E2E, personalization Hit@1/Hit@3, diversity LD/BD, fairness GF/AF/VF/RF/CF, explanation FS/CS/PS, robustness TR/QR), grouped by backbone block. All rate-type metrics are in percent ($\times 100$); TTFT and E2E in seconds; higher is better except TTFT/E2E. Best in **bold** and second best underlined, compared across all backbones rather than within each block.

	Tool		**Efficiency**		**Personalization**		**Diversity**		**Fairness**					**Explanation**			**Robustness**	
	TISR	DR	TTFT	E2E	H@1	H@3	LD	BD	GF	AF	VF	RF	CF	FS	CS	PS	TR	QR
Qwen3.7-Max																		
ReAct (no mem)	95.3	96.8	6.98	11.69	31.8	48.1	43.1	49.1	97.0	86.6	83.9	92.7	95.3	91.1	90.9	92.5	87.9	38.3
Plan-and-Execute	88.4	88.6	**3.94**	**8.03**	30.1	45.4	37.8	49.7	99.0	83.4	79.4	91.6	84.8	89.8	92.2	93.6	**95.2**	31.3
ReAct (last-k)	95.7	97.0	7.04	11.79	33.3	50.0	43.8	48.4	96.6	**92.8**	77.4	92.0	95.5	90.0	91.2	92.6	89.2	39.4
ReAct (summary)	96.9	98.3	8.96	15.66	35.5	52.7	47.8	49.4	96.4	86.4	84.7	95.4	94.5	89.6	92.8	93.6	90.0	50.2
ReAct (retrieval)	96.2	97.4	8.81	14.92	33.2	49.0	44.9	48.8	96.6	88.0	84.1	91.6	94.9	91.6	91.8	93.0	88.5	39.2
Letta (MemGPT)	**98.2**	96.8	7.96	14.84	36.2	53.7	46.5	49.3	96.0	86.8	86.6	94.1	93.0	92.9	95.4	95.8	82.4	43.0
MetaGPT	97.6	98.1	10.55	16.11	35.5	52.0	47.7	49.5	96.7	86.0	79.9	96.5	89.4	92.6	95.9	96.3	90.5	51.6
CAMEL	**98.2**	99.2	20.75	27.17	38.4	**56.8**	50.3	49.7	96.5	82.3	88.5	94.1	91.6	86.9	93.8	93.9	89.2	61.6
InteRecAgent	95.2	97.1	9.94	17.79	36.6	53.0	43.0	46.9	94.7	89.4	93.5	95.3	91.0	87.8	92.9	94.1	87.8	59.4
iAgent	94.6	**99.8**	16.41	27.33	37.1	51.9	50.2	47.7	97.2	86.1	82.0	92.7	88.7	76.3	91.6	90.8	80.3	45.3
MACRec	97.5	99.7	36.69	43.16	38.4	54.9	46.2	45.3	95.6	91.4	**94.6**	94.2	95.2	88.5	91.9	92.8	95.0	56.0
GPT-5.4																		
ReAct (no mem)	96.0	99.7	9.31	13.96	31.2	47.6	42.6	49.5	99.5	80.6	86.6	92.9	90.5	95.6	94.5	96.4	87.0	35.6
Plan-and-Execute	94.4	98.8	8.11	12.84	33.2	49.5	44.3	49.4	98.8	83.7	82.2	95.8	89.2	**97.8**	**98.0**	**98.8**	92.5	40.8
ReAct (last-k)	94.9	99.2	9.49	14.58	30.5	47.0	41.7	48.3	98.7	86.4	81.0	93.0	93.2	96.5	95.5	97.0	83.9	25.3
ReAct (summary)	96.2	98.1	14.36	21.02	32.9	49.1	44.9	48.5	96.5	87.8	88.7	98.1	90.6	95.7	95.6	97.2	86.8	33.4
ReAct (retrieval)	97.6	99.6	13.66	20.23	31.9	47.7	42.6	47.7	98.0	84.6	86.4	94.6	92.2	96.4	95.6	97.3	85.9	40.1
Letta (MemGPT)	97.6	96.6	11.45	16.02	35.0	52.3	46.1	48.4	94.8	85.3	88.2	95.5	92.0	96.8	96.5	97.8	88.6	45.4
MetaGPT	97.7	97.7	13.84	20.58	35.4	52.3	49.5	50.4	**99.6**	87.0	89.6	**99.1**	91.4	97.3	97.9	98.3	86.6	45.7
CAMEL	97.2	92.8	30.10	34.05	35.1	53.5	47.6	47.6	96.8	83.9	85.1	94.6	92.7	96.4	96.2	97.5	87.2	64.4
InteRecAgent	98.1	98.9	10.02	18.46	34.1	51.9	44.4	47.3	95.6	87.6	89.1	95.5	91.8	94.4	95.0	97.0	92.1	41.3
iAgent	82.5	**99.8**	38.08	50.24	**38.9**	55.1	**54.5**	45.7	99.4	90.3	92.5	94.0	**96.6**	91.3	95.1	95.5	90.1	62.7
MACRec	98.1	98.3	35.82	41.97	37.0	56.5	49.2	**51.5**	97.4	89.3	89.7	97.4	89.6	96.5	96.9	98.5	85.5	60.6
DeepSeek-V4-Pro																		
ReAct (no mem)	96.3	92.4	10.54	16.08	30.3	46.7	41.3	47.6	97.7	82.7	77.4	93.9	91.1	92.6	93.1	94.3	86.2	35.1
Plan-and-Execute	94.7	90.5	4.85	10.77	28.0	44.0	39.8	49.2	98.6	83.1	77.6	93.1	93.8	89.8	92.7	94.2	80.3	32.9
ReAct (last-k)	96.6	93.1	11.05	16.45	31.6	47.7	41.7	48.7	96.3	87.2	81.7	95.7	91.8	90.2	92.4	93.8	82.5	39.9
ReAct (summary)	97.5	95.1	13.34	19.33	34.5	50.7	43.2	47.6	98.1	85.9	84.6	96.9	88.3	90.9	94.5	95.4	91.9	56.8
ReAct (retrieval)	97.5	94.5	13.28	19.16	31.6	47.3	42.1	46.7	97.7	85.7	81.7	92.5	94.2	91.2	94.0	95.2	84.4	33.9
Letta (MemGPT)	97.7	89.9	10.40	19.05	34.2	52.5	43.6	49.7	98.9	90.4	85.6	95.3	96.3	94.2	96.4	97.4	86.1	40.1
MetaGPT	97.8	94.2	11.53	17.38	28.8	45.4	43.6	48.8	98.1	82.7	89.4	94.7	88.6	90.9	92.5	93.7	83.4	58.8
CAMEL	97.1	89.0	55.83	60.94	33.8	47.4	37.7	41.3	94.6	87.8	87.8	97.8	88.2	90.0	92.0	92.8	84.3	**73.3**
InteRecAgent	96.6	92.0	9.42	16.36	32.0	47.1	36.3	43.5	98.9	87.1	94.5	92.7	89.2	88.1	89.7	91.4	73.9	64.3
iAgent	96.6	98.6	26.42	35.63	31.9	47.0	45.8	48.2	99.2	83.7	83.5	89.6	92.3	78.3	88.4	88.5	86.8	60.7
MACRec	97.0	92.4	77.76	83.13	32.7	47.9	42.6	44.2	95.2	91.1	91.9	96.7	91.5	78.7	86.3	87.8	75.1	59.1

Table 3: Behavioral alignment (Hit@1 / Hit@3, in percent) by task type: the per-type decomposition whose equal-weight average is the main-table Hit@K. Driving hits require the best candidate to cover at least 0.8 of the ground-truth mileage. Columns are grouped by LLM backbone; each backbone reports POI / Drive / Transit. Higher is better. Best in **bold**, second-best underlined within each backbone.

	Qwen3.7-Max						**GPT-5.4**						**DeepSeek-V4-Pro**					
	POI		Drive		Transit		POI		Drive		Transit		POI		Drive		Transit	
	H@1	H@3	H@1	H@3	H@1	H@3	H@1	H@3	H@1	H@3	H@1	H@3	H@1	H@3	H@1	H@3	H@1	H@3
ReAct (no mem)	19.3	26.7	34.4	47.4	41.7	70.3	15.6	22.9	33.7	45.8	44.5	74.2	16.9	24.0	35.9	48.5	38.1	67.6
Plan-and-Execute	14.4	20.0	32.1	43.4	43.6	72.9	16.7	22.9	36.4	48.7	46.6	76.9	14.9	21.6	32.9	44.2	36.0	66.3
ReAct (last-k)	18.7	27.1	36.3	49.3	44.9	73.5	13.8	20.7	34.3	46.6	43.4	73.7	17.1	25.8	37.9	49.4	39.8	68.0
ReAct (summary)	17.8	24.4	41.3	53.9	47.5	79.7	16.2	22.0	36.3	48.3	46.2	77.1	17.6	24.4	40.1	50.9	**45.8**	76.9
ReAct (retrieval)	17.8	24.9	37.6	50.6	44.1	71.4	15.8	22.9	35.0	46.5	44.9	73.7	16.0	24.7	36.3	48.3	42.4	68.9
Letta (MemGPT)	18.4	25.3	42.6	56.3	47.5	79.4	16.9	23.3	40.5	53.6	47.7	79.9	16.9	25.6	40.8	54.2	44.9	**77.8**
MetaGPT	20.2	28.0	37.9	51.3	48.5	76.7	18.0	25.6	37.3	49.9	50.8	81.4	17.1	24.9	30.9	42.6	38.3	68.6
CAMEL	18.7	28.2	45.0	**59.6**	**51.5**	**82.4**	17.6	24.7	41.3	55.8	46.4	79.9	13.1	20.2	**48.3**	**59.3**	40.0	62.5
InteRecAgent	17.8	23.6	41.7	55.1	50.4	80.3	15.8	24.0	39.9	51.7	46.6	80.1	17.1	24.2	35.1	44.8	43.9	72.2
iAgent	**22.7**	**30.7**	40.7	51.3	47.9	73.7	**21.8**	**31.1**	42.6	55.4	**52.3**	78.8	**18.0**	**26.7**	34.4	44.8	43.4	69.7
MACRec	16.9	23.6	**47.4**	59.5	50.8	81.8	16.0	24.9	**46.2**	**63.0**	48.9	**81.8**	12.7	20.0	39.8	52.6	**45.8**	71.2

Evaluation denominators. All evaluated configurations are scored on the full benchmark under one shared pipeline; the zero-LLM reference of Appendix D.2 is the only exhibit that also reports a number on a subset, because no default output exists for 557 episodes.

D.2 HIT@K OF THE NAVIGATION SYSTEM'S ORIGINAL RESPONSES

This reference is not an agent: it is the response the production navigation system originally returned to each query, namely its recorded top-3 candidates taken verbatim, re-scored by the identical Hit@K pipeline on the same full test set (N=1,608) with no LLM involved. Episodes whose recorded session produced no original response (557 in total, of which 390 driving episodes from the additional collection window carry no original recordings) count as undelivered and therefore as misses. On the full set this reference delivers 65.4% of episodes and reaches Hit@1 17.0 and Hit@3 32.1 (Pers. 32.1; by type POI 22.0, driving 24.9, transit 49.4).

Two readings follow. First, the transit Hit@3 of 49.4 restates that transit ground truth is the user's selection among the presented candidates, so this reference records the production ranking as presented to the user rather than a competitor to beat. Second, on identical episodes the reference is beatable: over the 227 driving episodes with an original response it scores 75.3 while the evaluated configurations reach a median of 78.9 and a best of 87.7 (Letta on Qwen3.7-Max), and over all 686 driving episodes, 459 of which carry no original response, it falls to 24.9 against 42.6–63.0 (median 50.6). POI selection is where the reference is weakest (28.1 on the 352 POI episodes with an original response).

Excluding the 557 episodes without a recorded response, the reference reaches pooled Hit@1 26.6 and Hit@3 47.9 on the remaining 1,051 episodes (Pers. 50.9), and on this common subset the evaluated configurations attain Pers. 51.9–65.6 with a median of 60.4: the reference stays below every configuration even when the episodes without a recorded response are dropped from the denominator. POI selection is the exception: the reference's 28.1 exceeds the configuration median of 25.3 and is beaten by only 2 of the 33 cells, whereas driving (75.3 against a median of 78.9, beaten in 28 of 33 cells) and transit (49.4 against 74.2, beaten in all 33) account for the overall margin.

D.3 HIT@K OF LAST-ROUND TOOL CANDIDATES

The previous subsection fixes the reference at the dataset level. Table 4 instead reports a reference specific to each configuration: the first three candidates returned by its *final* candidate-producing tool call, delivered verbatim, with no selection, reranking, or personalization. For POI ground truth that call is the last invocation of any POI search tool; for driving and transit ground truth, the last `plan_route` call. Scoring uses the identical Hit@K pipeline over the same 1,608 episodes, so the only difference from Table 2 is the source of the prediction, and the table reads as a reference for tool orchestration rather than an upper bound on retrieval.

Table 4: Hit@K when each configuration's own last-round tool candidates are delivered verbatim: the first three candidates returned by its final candidate-producing tool call of the matching type (the POI search tools for POI ground truth, `plan_route` for driving and transit ground truth), re-scored by the identical Hit@K pipeline on the full test set (N=1,608) with no agent-side selection, reranking, or personalization. Values are in percent ($\times$100); each cell is the equal-weight average of the POI, driving, and transit hit rates. Across the 33 backbone–configuration cells this reference reaches Hit@1 18.1–26.0 and Hit@3 33.7–45.6, and it stays below the delivered answers in every cell (mean gap 10.8 points at Hit@1 and 9.6 at Hit@3), so candidate selection contributes positively throughout. Best in **bold**, second best underlined.

	Qwen3.7-Max		**GPT-5.4**		**DeepSeek-V4-Pro**	
	Hit@1	Hit@3	Hit@1	Hit@3	Hit@1	Hit@3
ReAct (no mem)	22.0	39.8	21.7	39.4	22.1	39.6
Plan-and-Execute	19.7	35.1	22.1	39.5	20.1	36.0
ReAct (last-k)	22.8	40.3	21.3	39.1	22.2	39.5
ReAct (summary)	24.0	42.3	23.0	40.1	22.9	40.9
ReAct (retrieval)	22.7	40.2	21.5	38.6	22.2	38.9
Letta (MemGPT)	23.9	42.6	24.0	42.8	<u>24.0</u>	**43.0**
MetaGPT	22.4	38.0	23.0	38.9	18.1	33.7
CAMEL	**26.0**	**45.6**	23.5	<u>43.5</u>	**25.8**	<u>42.6</u>
InteRecAgent	23.8	42.3	23.4	41.2	20.4	37.9
iAgent	24.2	42.1	**25.8**	43.0	23.5	41.1
MACRec	<u>25.6</u>	<u>45.3</u>	<u>25.4</u>	**45.2**	22.5	40.2

Episodes whose trace contains no matching tool call (24.7% of driving episodes pooled over the 33 cells), and calls whose result yields no parseable candidate list (a further 0.8%), count as misses.

D.4 Paradigm-Level Comparison

Averaging the member configurations of each category (Table 13) within each backbone yields the paradigm-level view behind Table 2.

Category means show stable structure and one rotating effect. Two regularities hold on every backbone: reasoning baselines trail all other categories on driving, and recommendation-specialized agents explain least while delivering at or near the top. What rotates with the backbone is which paradigm leads alignment. On Qwen3.7-Max, multi-agent systems attain the highest POI Hit@1 (19.44) and the highest driving alignment (driving Hit@3 = 55.47), with recommendation-specialized agents a close second (19.11 / 55.30) and memory-enhanced agents third (18.17 / 52.51). The leadership differs by backbone: recommendation-specialized agents lead driving on GPT-5.4 (56.71), multi-agent systems lead it on DeepSeek-V4-Pro (50.95), and memory-enhanced agents lead POI selection on DeepSeek-V4-Pro (16.89). On Qwen3.7-Max, multi-round coordination and domain-specific reranking co-occur with the best POI selection and the best route recovery; on the other two backbones the leading paradigm differs, so no single paradigm owns either task family.

High delivery coexists with the weakest explanations. Recommendation-specialized agents explain poorly: their delivery is at or near the top of the paradigm ranking on every backbone (DR = 98.88 / 99.03 / 94.34, exceeded only by the reasoning baselines on GPT-5.4 at 99.25), while their explanation quality (the mean of FS, CS, and PS) is the lowest on every backbone (89.64 / 95.57 / 86.36) and their latency the highest (E2E = 29.43 / 36.89 / 45.04 s). Their robustness scores rank first or second of the four categories on every backbone (the mean of TR and QR: 70.64 / 72.05 / 69.99).

Latency increases monotonically with paradigm complexity on every backbone (9.86 $\rightarrow$ 14.30 $\rightarrow$ 21.64 $\rightarrow$ 29.43 s on Qwen3.7-Max from baselines to recommendation-specialized agents), foreshadowing the efficiency–quality analysis of Appendix D.8.

D.5 Within-Category Comparisons

Reasoning baselines. ReAct (no mem) is the stable all-rounder on Qwen3.7-Max (Hit@1 = 31.8, TISR = 95.3, DR = 96.8), whereas Plan-and-Execute attains the fastest execution of all configurations (E2E = 8.03 s) at the cost of fragility: on Qwen3.7-Max it records the lowest TISR (88.4), DR (88.6), and hard-query retention (QR = 31.3) of the backbone block, driven by structural execution failures (Appendix D.10).

Memory-enhanced. On Qwen3.7-Max the three ReAct memory variants stay within 1.4–3.7 points of the no-memory baseline on Hit@1 (33.2–35.5 vs. 31.8), so the three injection strategies differ little from one another on this backbone while sharing a common cost: the summarization variant adds roughly a third of prompt-side latency (TTFT = 8.96 vs. 6.98 s) and of end-to-end latency (15.66 vs. 11.69 s). Where the choice of strategy does matter is summarization against the no-memory baseline, the only comparison with a repeatable positive effect on two of the three backbones (Appendix D.6). Letta is the strongest memory variant on Qwen3.7-Max (Hit@1 = 36.2, Hit@3 = 53.7, TISR = 98.2).

Multi-agent. On Qwen3.7-Max the two members split the category's strength. CAMEL's role-play negotiation tops the block on alignment (Hit@1 = 38.4, Hit@3 = 56.8, including the best driving Hit@3 and transit Hit@1 of the Qwen block, Table 3) at substantial latency (E2E = 27.17 s). MetaGPT's SOP pipeline is the reliability workhorse (TISR = 97.6, DR = 98.1) with mid-field alignment (Hit@1 = 35.5) and slightly lower diversity than CAMEL (LD = 47.7 vs. 50.3). CAMEL's advantage does not carry over to GPT-5.4 or DeepSeek-V4-Pro, where it falls to third and fourth place on Hit@3.

Recommendation-specialized. On Qwen3.7-Max the three members divide the category's strengths. iAgent is the POI specialist (POI Hit@1 = 22.7, the highest of the block) with near-perfect delivery (DR = 99.8) but the least faithful explanations (FS = 76.3). MACRec matches the block's best alignment (Hit@1 = 38.4) at the highest latency of the Qwen block (E2E = 43.16 s). InteRecAgent persuades (PS = 94.1) while trailing on POI selection (POI Hit@1 = 17.8).

D.6 Memory Injection Analysis

This appendix details the paired flip and mechanism analysis behind the aggregate memory scores of Table 2; all statistics are computed over the 1,608 episodes of the benchmark.

Paired flip setup. For each ReAct memory variant (last-k, summary, retrieval) we form paired episodes against the no-memory baseline within the same backbone over all 1,608 episodes, where an undelivered answer counts as a miss exactly as in the Hit@K pipeline. A *win* is an episode where the memory variant hits at rank 3 while the baseline misses; a *loss* is the reverse. Each variant therefore contributes 1,608 pairs per backbone, that is 4,824 pairs per backbone pooled over the three variants and 14,472 pairs in total.

Flip statistics. Wins exceed losses on every backbone, but the margin is backbone-dependent: Qwen3.7-Max records 254 wins against 126 losses, DeepSeek-V4-Pro 321 against 236, and GPT-5.4 234 against 210. An exact McNemar test per variant and backbone, with Benjamini–Hochberg correction (Benjamini & Hochberg, 1995) across the nine cells, leaves three significant at $q = 0.05$: the summary variant on Qwen3.7-Max (134 wins / 55 losses, $p < 0.001$) and on DeepSeek-V4-Pro (132 / 70, $p < 0.001$), and the last-k variant on Qwen3.7-Max (60 / 30, $p = 0.002$). The retrieval variant is never significant (net $+19$, $+3$, $+7$; $p \geq 0.07$), and no GPT-5.4 cell is significant (net -6, $+27$, $+3$). Summarization is therefore the only memory strategy with a repeatable positive effect, and it holds on two of the three backbones.

Where flips occur. Decomposing the 14,472 pairs by ground-truth task type shows that memory injection is not neutral across tasks. POI episodes are the only type with a net negative balance (150 wins against 167 losses), whereas driving (314 / 186) and transit episodes (345 / 219) are net positive. Transit episodes are also the most volatile, carrying both the highest win rate (8.12% of transit pairs) and the highest loss rate (5.16%), against 3.70% and 4.12% on POI and 5.09% and 3.01% on driving. Injected history therefore helps most where the ground truth is a user-selected itinerary and hurts most where it is a discrete POI choice.

Win mechanisms. We label each win by whether the ground-truth target appears verbatim in the injected memory text (POI names; road names and road sequence for driving; line names for transit).

Only 6.4% of wins do (52 of the 807 wins whose ground truth carries an extractable reference string), which is indistinguishable from the 6.2% observed among episodes that both configurations miss (410 of 6,579). Memory therefore rarely wins by placing the answer in the context. The check is a lexical proxy, so the absence of a ground-truth string does not exclude inference from related history.

Letta. The self-managed design behaves differently from the three injection variants: it attains the highest Hit@3 of the DeepSeek-V4-Pro block (52.5).

D.7 Backbone-Level Comparison

Averaging all eleven framework configurations within each LLM backbone complements the framework-level views above.

Alignment is nearly backbone-invariant. POI Hit@1 spans only 2.30 points (16.12–18.42) and driving Hit@3 3.47 points (49.05–52.52) across backbones, so the spread in the main table is driven by framework architecture rather than backbone capability, consistent with the failure analysis attributing route-task failures to candidate selection.

Backbones separate on peripheral quality instead. GPT-5.4 is the quality-oriented backbone: the highest delivery (DR = 98.15) and the highest explanation quality (96.45), at moderate latency (E2E = 24.00 s). Qwen3.7-Max is the balanced one: the fastest (E2E = 18.95 s) while keeping the highest alignment (POI Hit@1 = 18.42, driving Hit@3 = 52.52). DeepSeek-V4-Pro is reliable and steady, with the highest tool success rate (TISR = 96.85) and fairness mean (90.92), but pays for it with the lowest delivery (DR = 92.88), explanations trailing GPT-5.4 by 5.19 points (91.26 vs. 96.45), and the highest latency (E2E = 28.57 s).

D.8 Efficiency–Quality Trade-offs

Frameworks cluster into three regimes along the latency–alignment plane. **Fast and effective** configurations (Plan-and-Execute, Letta, and the plain ReAct variants; E2E = 8.0–15.7 s on Qwen3.7-Max, Hit@1 = 30–36) offer real-time responsiveness with competitive personalization. **Balanced** configurations (MetaGPT, InteRecAgent; E2E = 16.1–17.8 s) trade some alignment for tool reliability, delivery, and explanation quality. **Slow but thorough** configurations (CAMEL, iAgent, MACRec; E2E = 27.2–43.2 s) spend extended negotiation or reranking chains on broader candidate exploration and reach the highest alignment on Qwen3.7-Max (Hit@1 = 37–38). Latency and alignment are positively associated across the 33 configurations (Spearman $\rho(\mathrm{E2E}, \mathrm{Hit@1}) = 0.49$, $p = 0.004$, on the main-table macro Hit@1; within the Qwen3.7-Max block the same association on Hit@3 is $\rho = 0.72$ as reported in the main text): the coordination-heavy paradigms that spend more time also align better, though the return per added second collapses beyond the balanced regime (on Qwen3.7-Max, $+2.7$ Hit@1 points for $+4.1$ s from the fast to the balanced regime against $+1.9$ points for $+15.6$ s into the slow regime). Among memory-centric agents, Letta is the efficiency frontier on Qwen3.7-Max (Hit@1 = 36.2 at 14.84 s), within 2.2 points of the best alignment in the block at about one third of the slowest configuration's latency.

D.9 Detailed Fairness Analysis

The standardization is defined in Appendix D.9.1, and leave-one-out effects with significance tests are tabulated in Appendix D.9.2. The five dimensions separate cleanly: gender sits close to parity and consumption level near it; city tier, two age bins (40–44 and 55–59), and vehicle ownership are under-served in nearly every cell, with the vehicle-ownership gap being the largest and most consistent; and the remaining dimensions localize to specific groups. Read together they identify which individual systems diverge on which groups, with vehicle ownership and city tier marking the clearest population-level axes.

Gender Fairness (GF) scores are consistently high (94.6–99.6); female users are under-served in only 2 of 33 cells and by at most 1.2 points in both, so gender is effectively at parity across the grid (Appendix D.9.2).

Age Fairness (AF) is discriminative (80.6–92.8) and its direction depends on the bin: the 55–59 and 40–44 bins are under-served in all 33 cells (mean effect -8.10 and -5.61 points; no individual

cell survives BH-FDR correction, consistent with single-cell power of about 7–11 points), whereas the 60+ group is under-served in only 13 of 33 cells and favored in 20, with a slightly positive mean effect per backbone ($+0.81$ to $+1.61$ points), and the 25–29 bin is favored on GPT-5.4 and DeepSeek-V4-Pro (11/11 and 10/11 cells) but mixed on Qwen3.7-Max (6/11) (Appendix D.9.2). Pooled over the eleven configurations within each backbone, the 55–59 bin trails the strongest age group by 10.8–12.1 points of standardized Hit@3 (all $q < 10^{-7}$ after BH-FDR), the 40–44 bin by 8.3–9.2 points (all $q < 10^{-7}$), and the 60+ group by only 1.74–3.33 points (Qwen3.7-Max $q = 0.030$, GPT-5.4 $q = 0.052$, DeepSeek-V4-Pro $q = 0.294$); this vs-best contrast uses the single strongest group as reference, whereas the leave-one-out effects above use the mean of all other groups, which is why the two views can point in different directions for the 60+ group. Age therefore separates systems cell by cell, and its population-level deficit concentrates in the 55–59 and 40–44 bins rather than in the oldest group. The demand-composition analysis below describes how query intent differs across age groups.

Table 5 decomposes the query-intent composition of the 450 POI episodes by age group; intents are assigned by keyword rules with no LLM involved, 376 episodes (83.6%) match a rule, and the denominators are the rule-classifiable episodes of each age bin. Along-route search is the hardest intent class for agents (a pooled Hit@3 of 16.0% across the 33 runs, against 39.1% for fueling/parking/charging and 21.7% for nearby), and the two older bins issue it most often (66.7% for 60+ and 54.5% for 55–59, against 42.0% overall); the classifiable count for the 55–59 bin (11 episodes) is too small to attribute its service deficit to this composition.

Table 5: Query-intent composition of POI episodes by age group (% of each group's rule-classifiable POI episodes; n = episode count including the 16.4% that match no rule, so rows need not sum to 100). The along-route share rises with age to 66.7% in the 60+ group against 42.0% overall, while that group's fueling/parking/charging share is the lowest of any bin (11.1% against 44.1%): the oldest cohort concentrates on the intent class that agents handle weakest (pooled Hit@3 16.0%).

Age	n	Along-route	Nearby	Fuel/Park/Charge
[18,24]	75	36.4	16.4	47.3
[25,29]	113	34.3	13.7	52.0
[30,34]	69	51.7	5.0	43.3
[35,39]	57	38.6	11.4	50.0
[40,44]	58	42.6	12.8	44.7
[45,49]	25	47.6	9.5	42.9
[50,54]	19	38.9	33.3	27.8
[55,59]	11	54.5	27.3	18.2
$\geq$60	23	66.7	22.2	11.1
All	450	42.0	13.8	44.1

Vehicle-Ownership Fairness (VF) shows the widest cell spread of the five dimensions (77.4–94.6) and, together with region fairness, the most consistent direction: non-owners are under-served in all 33 cells, with a mean deficit of about 14 points and every cell individually significant (Appendix D.9.2). The gap therefore reflects a population-level difference between car owners and non-owners rather than a few framework–backbone combinations.

Region and Consumption-Level Fairness. Under the city-tier grouping (T1/NT1 = first-tier and new first-tier cities, T2–3 = second and third tier, T4+ = fourth tier and below), region fairness shows the most consistent direction of the five dimensions: users in T2–3 cities are under-served in 32 of 33 cells (no individual cell survives BH-FDR correction, consistent with the moderate effect size), while T1/NT1 users are favored in 32 of 33 (Appendix D.9.2). The aggregate RF scores remain high (89.6–99.1) because the typical effect is moderate in magnitude (median 4.4 points), not because the direction is absent. Consumption-level fairness (CF, 84.8–96.6) is more cell-specific and correlates only weakly with overall accuracy (Spearman $\rho(\mathrm{CF}, \mathrm{Hit@3}) = 0.26$, $p = 0.14$, over the 33 cells), so the consumption spread cannot be reduced to a general quality deficit. Our data do not identify the direction of causality; one hypothesis, which we do not test here, is that high-consumption users issue simpler queries that tools resolve more easily.

D.9.1 TASK-TYPE STANDARDIZATION

All group-level hit rates are task-type-standardized before any gap or leave-one-out effect is computed, because raw group Hit@3 mixes service disparities with query-mix differences across groups.

For each group g and framework–backbone cell c, the standardized Hit@3 is the weighted average of that group's per-task-type Hit@3:

$$H_g^{(c)} = \sum_t w_t \cdot \overline{\text{hit3}}_{g,t}^{(c)}. \tag{11}$$

This gives $w_{\text{POI}} = 28.0\%$, $w_{\text{drive}} = 42.7\%$, and $w_{\text{transit}} = 29.4\%$ in the benchmark. Each aggregate fairness score is one minus the max–min spread of the group-level $H_g^{(c)}$ within a backbone, and the leave-one-out effects in Appendix D.9.2 are computed from the same standardized values rather than from raw Hit@3.

D.9.2 LEAVE-ONE-OUT FAIRNESS EFFECTS (D_g)

Tables 6–11 report the leave-one-out effect size

$$D_g^{(c)} = H_g^{(c)} - H_{-g}^{(c)}, \quad H_{-g}^{(c)} = \frac{\sum_{g' \neq g} n_{g'}^{(c)} H_{g'}^{(c)}}{\sum_{g' \neq g} n_{g'}^{(c)}}, \tag{12}$$

where $H_g^{(c)}$ is the task-type-standardized Hit@3 of group g for framework–backbone cell c and $H_{-g}^{(c)}$ is the weighted average of all other groups. Negative values indicate that group g is under-served relative to the rest of the population; significance is assessed by a two-proportion z-test with BH-FDR correction (Benjamini & Hochberg, 1995) within each dimension ($^{*}q < 0.05$, $^{**}q < 0.01$, $^{***}q < 0.001$). Group sizes range from 85 to 1,084 episodes (median 251), so at the 5% level a single cell resolves differences of about 6.7 points or larger for a median-sized group and about 10.9 points for the smallest; because each cell is tested individually and then corrected within its dimension, the stars mark effects large enough to survive multiple-comparison correction within one cell, and the directional conclusions rest on consistency across cells rather than on individual p-values. For gender only Female is shown because Male is the exact mirror image of Female; for vehicle ownership the disadvantaged group is "No car". Per-backbone D_g breakdowns for consumption level (CF) are omitted for space: its gaps concentrate in isolated framework–backbone cells rather than showing a directional pattern, as analyzed above, and are summarized by the aggregate CF scores.

Table 6: Leave-one-out fairness effect $D_g = H_g - H_{-g}$ for Gender (GF). Only Female is shown because Male is its mirror image. Values are percentage points ($\times 100$). Negative values indicate the group is under-served relative to everyone else. Significance is assessed by a two-proportion z-test with BH-FDR correction within each dimension: $^{*}q < 0.05$, $^{**}q < 0.01$, $^{***}q < 0.001$.

	Qwen3.7-Max	**GPT-5.4**	**DeepSeek-V4-Pro**
ReAct (no mem)	+2.97	+0.48	+2.35
Plan-and-Execute	+1.04	-1.21	+1.45
ReAct (last-k)	+3.40	+1.31	+3.68
ReAct (summary)	+3.60	+3.55	+1.93
ReAct (retrieval)	+3.38	+2.00	+2.31
Letta (MemGPT)	+3.99	+5.17	+1.11
MetaGPT	+3.33	+0.37	+1.86
CAMEL	+3.51	+3.17	+5.45
iAgent	+2.79	+0.64	-0.83
InteRecAgent	+5.33	+4.44	+1.15
MACRec	+4.39	+2.55	+4.80

Table 7: Leave-one-out fairness effect $D_g = H_g - H_{-g}$ for Vehicle ownership (VF). The 'No car' row reports the effect for non-car owners relative to car owners. Values are percentage points ($\times 100$). Negative values indicate the group is under-served relative to everyone else. Significance is assessed by a two-proportion z-test with BH-FDR correction within each dimension: $^{*}q < 0.05$, $^{**}q < 0.01$, $^{***}q < 0.001$.

	Qwen3.7-Max	**GPT-5.4**	**DeepSeek-V4-Pro**
ReAct (no mem)	-16.15***	-13.38***	-22.55***
Plan-and-Execute	-20.60***	-17.77***	-22.37***
ReAct (last-k)	-22.56***	-19.00***	-18.32***
ReAct (summary)	-15.34***	-11.31***	-15.38***
ReAct (retrieval)	-15.87***	-13.64***	-18.25***
Letta (MemGPT)	-13.36***	-11.80***	-14.44***
MetaGPT	-20.12***	-10.36***	-10.64***
CAMEL	-11.46***	-14.93***	-12.23***
iAgent	-18.05***	-7.52**	-16.50***
InteRecAgent	-6.48*	-10.94***	-5.49*
MACRec	-5.42*	-10.28***	-8.11**

Table 8: Leave-one-out fairness effect $D_g = H_g - H_{-g}$ for Region / city tier (RF). Values are percentage points ($\times 100$). Negative values indicate the group is under-served relative to everyone else. Significance is assessed by a two-proportion z-test with BH-FDR correction within each dimension: $^{*}q < 0.05$, $^{**}q < 0.01$, $^{***}q < 0.001$.

	Qwen3.7-Max			**GPT-5.4**			**DeepSeek-V4-Pro**		
	T1	T2–3	T4+	T1	T2–3	T4+	T1	T2–3	T4+
ReAct (no mem)	+5.44	-6.73	+0.66	+6.88	-5.33	-3.73	+3.07	-5.54	+2.88
Plan-and-Execute	+6.80	-7.26	-0.83	+3.74	-2.37	-2.79	+5.72	-5.85	-1.07
ReAct (last-k)	+5.70	-7.49	+1.33	+6.42	-5.59	-2.60	+2.67	-4.29	+1.75
ReAct (summary)	+3.86	-3.91	-0.77	+1.60	-0.95	-1.28	+2.69	-2.55	-0.79
ReAct (retrieval)	+6.61	-7.38	-0.34	+5.26	-3.75	-3.32	+2.41	-5.78	+4.31
Letta (MemGPT)	+4.69	-5.22	-0.27	+4.33	-3.47	-2.19	+3.49	-4.32	+0.43
MetaGPT	+3.00	-2.87	-0.85	+0.56	-0.85	+0.30	+4.96	-3.35	-3.40
CAMEL	+2.45	-4.95	+3.05	+4.12	-4.86	+0.15	-0.43	+1.50	-1.45
iAgent	+0.50	-4.31	+5.36	+0.08	-3.28	+4.58	+1.66	-6.80	+7.02
InteRecAgent	+3.32	-4.50	+0.97	+3.07	-4.35	+1.17	+5.54	-6.60	+0.30
MACRec	+1.65	-4.36	+3.53	+2.05	-2.26	-0.16	+0.59	-2.24	+2.23

Table 9: Leave-one-out age fairness effect $D_g = H_g - H_{-g}$ for Qwen3.7-Max. Values are percentage points. Negative values indicate the age group is under-served relative to all other age groups. Significance is assessed by a two-proportion z-test with BH-FDR correction within each dimension: $^{*}q < 0.05$, $^{**}q < 0.01$, $^{***}q < 0.001$.

	[18,24]	[25,29]	[30,34]	[35,39]	[40,44]	[45,49]	[50,54]	[55,59]	$\geq$60
ReAct (no mem)	+0.87	-1.09	+7.04	+0.21	-2.96	+0.72	+2.84	-7.66	-2.81
Plan-and-Execute	+3.00	+2.74	+7.36	+0.11	-2.31	-6.67	-0.43	-10.78	-3.80
ReAct (last-k)	+0.53	+0.11	+3.09	-0.23	-1.75	+0.45	+0.65	-4.76	-0.66
ReAct (summary)	+0.31	-0.67	+6.93	-2.62	-6.32	-4.23	+4.67	-8.05	+3.77
ReAct (retrieval)	+1.53	+2.09	+5.24	-2.88	-4.19	-2.99	+3.91	-7.91	-0.79
Letta (MemGPT)	+3.48	-1.58	+6.32	-4.97	-8.58	-5.52	+2.68	-4.06	+5.57
MetaGPT	+2.10	+1.62	+1.67	-3.73	-3.24	-7.94	-3.99	-5.21	+7.69
CAMEL	+0.06	-0.11	+7.79	-3.15	-4.13	-4.66	+2.54	-11.52	+4.06
iAgent	+3.48	+0.21	+5.34	-0.70	-8.49	+0.85	-0.31	-9.82	+1.79
InteRecAgent	+6.16	-2.70	+4.12	-2.61	-6.08	-5.35	+2.78	-3.76	+1.99
MACRec	+3.36	+2.01	+3.38	-0.36	-4.82	-4.64	-0.11	-5.97	-0.44

Table 10: Leave-one-out age fairness effect $D_g = H_g - H_{-g}$ for GPT-5.4. Values are percentage points. Negative values indicate the age group is under-served relative to all other age groups. Significance is assessed by a two-proportion z-test with BH-FDR correction within each dimension: $^{*}q < 0.05$, $^{**}q < 0.01$, $^{***}q < 0.001$.

	[18,24]	[25,29]	[30,34]	[35,39]	[40,44]	[45,49]	[50,54]	[55,59]	≥60
ReAct (no mem)	+2.78	+4.03	+4.69	+2.83	-8.03	-2.99	+6.47	-14.11	-4.30
Plan-and-Execute	+2.73	+5.18	+0.76	+1.26	-5.04	-4.66	+2.70	-12.68	-0.79
ReAct (last-k)	+2.03	+2.92	+4.87	-0.78	-8.08	-6.01	+4.11	-9.86	+1.42
ReAct (summary)	+2.03	+5.24	+2.28	-1.22	-8.74	-5.45	+4.04	-6.84	+0.58
ReAct (retrieval)	+4.15	+8.00	+2.13	-4.62	-8.27	-9.38	+2.21	-7.32	+0.64
Letta (MemGPT)	+3.10	+1.30	+0.54	-1.62	-7.43	-6.47	+6.13	-9.45	+5.37
MetaGPT	+1.59	+6.74	-0.26	-6.83	-5.05	-4.94	-0.85	-7.86	+6.08
CAMEL	-1.39	+5.33	-0.04	+1.62	-4.57	-1.60	+7.33	-9.75	-1.44
iAgent	+1.83	+4.22	+3.37	-4.29	-3.68	+2.54	-1.79	-6.58	-1.29
InteRecAgent	+1.34	+4.71	+1.26	-2.05	-8.33	-5.61	+4.11	-8.97	+4.15
MACRec	+3.54	+0.51	-0.57	-2.75	-1.02	-1.21	+6.36	-5.07	-1.47

Table 11: Leave-one-out age fairness effect $D_g = H_g - H_{-g}$ for DeepSeek-V4-Pro. Values are percentage points. Negative values indicate the age group is under-served relative to all other age groups. Significance is assessed by a two-proportion z-test with BH-FDR correction within each dimension: $^{*}q < 0.05$, $^{**}q < 0.01$, $^{***}q < 0.001$.

	[18,24]	[25,29]	[30,34]	[35,39]	[40,44]	[45,49]	[50,54]	[55,59]	≥60
ReAct (no mem)	-1.81	+4.05	+7.40	-4.67	-5.57	-0.33	+5.08	-11.48	+0.62
Plan-and-Execute	+2.76	+3.03	+2.99	-1.80	-4.88	-6.25	+6.81	-11.09	-0.09
ReAct (last-k)	+0.38	+5.94	+0.65	-1.63	-8.77	-6.80	+5.27	-5.92	+3.05
ReAct (summary)	+2.81	+2.18	+1.77	-1.04	-6.58	-6.09	+7.45	-7.53	+0.79
ReAct (retrieval)	+0.31	+6.70	+3.34	-5.12	-9.75	-0.74	+5.13	-2.11	-1.04
Letta (MemGPT)	+1.48	+2.58	+3.58	-4.15	-7.24	-2.48	+1.14	-4.52	+3.37
MetaGPT	+2.22	+8.21	+0.31	-0.31	-6.38	-7.11	-4.64	-11.08	+3.22
CAMEL	-3.03	-2.15	+3.89	+0.95	-2.48	+0.32	+4.41	-8.56	+4.30
iAgent	+2.74	+0.61	+2.13	-0.56	-4.73	-1.39	+2.49	-14.01	+3.50
InteRecAgent	+4.76	+6.94	+1.34	-2.48	-7.27	-7.08	-0.96	-7.59	+0.11
MACRec	-1.34	+3.34	+1.25	-2.66	-0.24	-2.08	+4.01	-5.45	-0.08

Worst-group absolute levels. A small gap can also arise when every group is served poorly, so the aggregate fairness scores are read together with the absolute level of the worst-served group in each cell. Across the 33 cells the worst group's standardized Hit@3 spans 33.6 to 49.9 (median 42.4) against a same-basis overall level of 44.5 to 58.2, which places the worst-served group 5.7 to 14.3 points below the overall rate (median 8.5). The worst-served group is the 55–59 age bin in 20 of the 33 cells, followed by non-car owners (6 cells), 40–44 (4 cells), the lowest consumption level (2 cells), and the 45–49 bin (1 cell); the lowest cell (iAgent on DeepSeek-V4-Pro, 33.6 for the 55–59 bin with $n = 85$) still reaches an overall standardized Hit@3 of 47.3. A high aggregate fairness score therefore never coincides with uniformly poor service in this batch, and the per-cell values are released with the evaluation code.

Uncertainty for the largest gaps. The widest gap in each backbone block is quantified by a user-level bootstrap over the 1,458 users (sampled with replacement, $B = 5{,}000$, recomputing the task-type-standardized group rates in each replicate): ReAct (last-k) on Qwen3.7-Max (VF) 22.6 points, 95% CI 11.4–31.6; the no-memory ReAct baseline on GPT-5.4 (AF) 19.4, 12.9–35.2; and the same baseline on DeepSeek-V4-Pro (VF) 22.6, 14.3–30.5. All three intervals exclude zero. Resampling users instead of episodes moves the three lower bounds by 1.7, 0.0, and 1.2 points (episode-level intervals 13.1–30.6, 12.9–35.4, and 13.1–30.6), as expected when each user contributes only 1.10 episodes.

D.10 Failure Analysis: Tool-Call Errors and Failures

We decompose all Hit@1 = 0 episodes into four failure types with a deterministic precedence rule over the recorded tool trace (first match wins; no LLM or human judgment involved): *no tool call* (the trace is empty); *tool failures* (every call in the trace fails); *tool-selection errors* (at least one call succeeds, but the trace contains no call to the task-appropriate tool family—`query_poi`, `search_around_poi`, or `search_destination_parking` for POI episodes; `plan_route`, `search_along_route`, or `query_current_route_facts` for driving and transit episodes); and *reasoning errors*, the residual bucket (the trace calls the task-appropriate tool family and records at least one successful call, yet the final answer misses the ground truth). Table 12 reports the reasoning-error share by task type for all three backbones; on Qwen3.7-Max the reasoning share spans 28–82% (POI), 53–84% (driving), and 82–100% (transit), so its complement, the tool-side failures, concentrates on POI tasks. Two patterns deserve emphasis:

Tool-side failures are POI-specific. On POI tasks, tool-side failures (tool-selection errors, tool failures, and missing calls jointly, the complement of the reasoning share in Table 12) account for 18–72% of failed POI episodes on Qwen3.7-Max, versus at most 47% on driving and 18% on transit tasks; one illustrative tool-selection error invokes `plan_route` when the user asks for a nearby restaurant. The correct tool type is more obvious from the query on route tasks, which is where tool selection goes right.

Plan-and-Execute exhibits structural execution failures. Its POI reasoning share on Qwen3.7-Max is the lowest of the backbone block (28% of failed episodes, Table 12), so most of its failed POI episodes end in a tool-side breakdown rather than in wrong candidate selection: the decomposition step generates sub-plans that fail to execute or skip tool invocation entirely.

Table 12: Reasoning-error share (% of failed episodes) by task type; columns grouped by LLM backbone. Reasoning errors (a deterministic-rule residual: at least one tool call succeeds but the delivered answer misses the ground truth) dominate driving and transit tasks across all backbones.

	Qwen3.7-Max			**GPT-5.4**			**DeepSeek-V4-Pro**		
	POI	Drive	Transit	POI	Drive	Transit	POI	Drive	Transit
ReAct (no mem)	42	68	89	34	60	97	42	68	87
Plan-and-Execute	28	53	82	36	72	93	30	64	78
ReAct (last-k)	42	67	87	33	58	95	41	68	87
ReAct (summary)	48	72	97	43	61	96	49	68	98
ReAct (retrieval)	40	68	88	39	60	97	42	67	86
Letta (MemGPT)	43	75	99	43	71	98	46	75	98
MetaGPT	82	67	89	86	64	98	80	63	82
CAMEL	53	84	100	53	79	98	69	91	100
InteRecAgent	41	74	97	46	71	99	45	58	85
iAgent	52	71	97	80	91	99	51	69	97
MACRec	55	82	99	34	93	100	55	72	90

D.11 Tool Usage Patterns

Agents gravitate toward task-specific tools: `search_along_route` on POI tasks, `plan_route` on driving, and transit-specific tools on transit episodes. Per-episode call volume differs several-fold across configurations, with MetaGPT and MACRec issuing the most sub-calls; the per-task-type counts are released with the evaluation code.

Sandbox cache resolution. The response store holds 137,758 cached tool responses covering all 19 navigation tools. Across the 33 framework–backbone runs, agents issued 124,709 tool calls, of which 96,896 entered the sandbox path; the remainder are framework-local lookups (memory and profile tools) that never reach the store. Of the sandbox-path calls, 6,315 (6.5%) returned no usable response during execution, so the frozen store resolves 93.5% of them. The store is frozen before scoring, so re-evaluation of any run replays identical responses.

D.12 Robustness and Route-Coverage Breakdowns

The main results table reports aggregate scores; this appendix reports the remaining per-group breakdowns for robustness and route coverage (group fairness effects are in Appendix D.9.2). Delivery

needs no table: undelivered episodes are dominated by apology-only responses rather than crashes or format violations in most configurations, the exception being Plan-and-Execute on Qwen3.7-Max, whose POI delivery collapses behind its structural execution failures (Appendix D.10). All hit-rate-based entries use the task-type-standardized Hit@3 of Appendix D.9.1, which removes task-composition confounds across groups.

Robustness by group. On Qwen3.7-Max the standardized Hit@3 declines steeply with query difficulty: 60.0–67.1 at L1 (clearest expression), 24.7–44.1 at L4, and 7.0–43.8 at the sparsest L5 level (71 episodes), where two recommendation-specialized configurations reverse the order. QR separates frameworks most on Qwen3.7-Max: CAMEL retains hard queries best (L4 = 44.1) while Plan-and-Execute retains worst (L4 = 24.7). Across the 33 configurations the two robustness dimensions differ in how much they discriminate: TR is high and comparatively uniform (73.9–95.2) while QR is the widest (25.3–73.3).

Route coverage by task type. RC@1 and RC@3 follow the definitions and denominator of Appendix C, and confirm the binary picture with one addition: on Qwen3.7-Max driving coverage is high and uniform (RC@3 = 70.8–80.7, best ReAct (last-k) at 80.7), whereas transit coverage splits the configurations by over 25 points (RC@3 = 60.0–86.2, with iAgent and MACRec at 60.0–60.3 and Plan-and-Execute at 86.2).

E Experimental Details

E.1 Framework Adaptation Protocol

All evaluated frameworks are adapted under a single constraint: adaptation is restricted to the tool layer, the environment interface, and domain-specific prompt content, while each framework's native agent loop and memory mechanism are preserved, so measured differences reflect architecture rather than adaptation quality. Every framework operates through the deterministic replay sandbox over the shared navigation tool suite (Appendix E.3) with the unified navigation template (Appendix E.2). Table 13 lists the evaluated frameworks with their categories and personalized-context sources.

The per-framework adaptation surface is as follows. The ReAct variants keep the thought–action–observation loop and differ only in context supply: the latest 100 raw records, a scene-conditioned LLM summary of the user profile and behavioral history (hereafter the profile–history summary), or on-demand retrieval over cleaned behavior records. Plan-and-Execute keeps its plan-then-execute pipeline, rewriting only the plan and synthesize instructions. Letta keeps LLM-managed core, recall, and archival memory; the profile–history summary occupies the Core Memory human block and archival is prefilled with cleaned behavior records. MetaGPT keeps its SOP role-action-environment architecture and message routing, redefining the three role prompts and prefilling a retrievable store of behavior summaries that the Advisor queries. CAMEL keeps the RolePlaying dual-agent loop, prefilling the assistant's chat-history memory with the profile–history summary. MACRec keeps Manager-Analyst-Searcher coordination and the shared scratchpad, storing the profile and the profile–history summary in the InfoDatabase/Retriever pair that only the Analyst reads. InteRecAgent keeps its Plan-and-Solve ToolBox, dialogue and profile memories, and candidate buffer, swapping the raw history dump in the user message for the profile–history summary while UserProfileMemory still feeds the ranking pipeline. iAgent keeps its sequential chain and three-layer processing memory, feeding the profile–history summary into its native profile-generation steps.

Every configuration that receives a summarized profile draws it from one shared summarization module; ReAct (no mem) receives no behavioral history, while Plan-and-Execute and the last-k variant receive raw records. Cross-framework fairness differences partly reflect how explicitly each architecture uses demographic attributes, which we treat as an architectural property rather than an adaptation artifact.

E.2 Prompt Specifications

All prompts are deployed in Chinese (all benchmark queries are Chinese); we show English translations and release the originals with the code. The unified navigation template (Appendix E.2.1) governs every tool-invoking step; the system prompts of ReAct (all four variants), Plan-and-Execute,

Table 13: Evaluated agents. All agents receive the current trip context and dialogue history; the last column lists each agent's personalized context source (profile and/or behavioral history).

Category	Agent	Paradigm	Personalized Context Acquisition
Baseline	ReAct (no mem)	ReAct	None
	Plan-and-Execute	Plan-and-Solve	Raw profile + latest 100 behaviors
Memory	ReAct (last-k mem)	ReAct	Raw profile + latest 100 behaviors
	ReAct (summary mem)	ReAct	Scene-conditioned LLM summary of profile and behaviors
	ReAct (retrieval mem)	ReAct	Raw profile + on-demand retrieval over cleaned behavior records
	Letta (MemGPT)	ReAct (function calling)	Self-managed core/archival memory (summary in core, cleaned records in archival)
Multi-agent	MetaGPT	SOP pipeline	Message memory + RAG over cleaned behavior records
	CAMEL	Role-play dual-agent negotiation	Prefilled chat-history memory (profile–history summary)
Recommendation	InteRecAgent	Plan-and-Solve (rerank)	Profile–history summary in user message + UserProfileMemory for ranking
	iAgent	Sequential chain (rerank)	Three-layer processing memory (pre-computed summary into profile steps)
	MACRec	Multi-agent ReAct	Analyst-mediated retrieval of profile–history summaries + shared scratchpad

Letta, and InteRecAgent use it directly, differing only in how the tool phase terminates (`finish`, `send_message`, or `Final Answer`). The Searcher of MACRec and of MetaGPT, and CAMEL's assistant, embed the same tool-routing rules in condensed form. Framework-specific prompts (Appendix E.2.2) govern the non-tool steps and preserve each framework's native logic. Memory-augmented frameworks additionally receive a per-case, scene-trimmed personalized-usage strategy (the `mem_personalize` prompt) injected alongside their profile–history summary, so the same personalization guidance is applied consistently across architectures.

E.2.1 Unified Navigation Template

The template opens with the role instruction "You are a professional mobility route-planning assistant" and keeps the tool-calling phase free of user-facing content: no markdown, summaries, or reasoning traces, with the phase ending in a `finish` call.

The routing rules govern all nineteen tools and impose three families of constraints. Dialogue history is used only for reference resolution and argument inheritance: a continued trip copies the start, end, and route types of the most recent `plan_route` call verbatim. Tool use is minimal and grounded: place names only, no repeated identical calls, no model-filled coordinates or POI ids, and missing endpoints or failed calls terminate in `finish`. Guardrails require every route, POI, transit, parking, and geographic fact to come from tool returns, bar fabricated distances, durations, fares, or congestion, and forbid the final answer from exposing coordinates, ids, raw tool inputs and outputs, or details of the user's history and profile.

After all tool calls complete, the agent outputs a single JSON object:

```
{
  "reason": "...",
  "answer": "...",
  "labels": ["...", ...],
  "memory": "..."
}
```

where `reason` explains and ranks the candidates, `answer` lists at most three candidates in descending priority, `labels` gives a short tag per candidate, and `memory` records which user context or history the answer used. Route candidates must begin with "Route N:" using the index from the tool-returned text, and multi-day itineraries with "Plan N (theme):", with all names drawn from tool returns.

E.2.2 Framework-Specific Prompts

Framework-specific prompts. The non-tool steps keep each framework's native logic; the full prompts are released with the code. Plan-and-Execute's planner must emit at least one tool call for any mobility request, including vague continuations such as "the nearest one along the way", and its synthesizer may not add further plans. The role-based frameworks differ in where personalization enters: MetaGPT's Advisor ranks tool-returned candidates against the profile (chargers for EVs, venue tier by consumption level), CAMEL's assistant reranks under a user agent limited to one instruction and two corrections per turn, MACRec confines profile and history access to the Analyst's

five commands, InteRecAgent consumes a JSON profile of `history`/`like`/`unwanted` lists with a repair prompt for invalid output, and iAgent reranks with memory inside its six-step chain.

E.3 Navigation Tool Interface

All frameworks share the identical set of 19 navigation tools, executed through the deterministic replay sandbox and injected into the prompts via the {`tool_descriptions`} placeholder of the unified template (Appendix E.2.1). The interface groups into five families: route planning (`plan_route` over car, transit, walking, cycling, and EV modes; loop routes; multi-waypoint resolution); POI and parking search (by name, nearby a location, along a planned route, and at the destination); transit information (line and station lookup, transfers, nearby-station search, and real-time arrivals); combined multi-modal itineraries resolved in a single call (transit-plus-riding, park-and-ride, cycling-plus-metro); and environmental context (weather, current-route facts, restriction rules, traffic events, and reverse geocoding). All tools return structured natural-language summaries consumed by the agent.

E.4 Evaluation Prompts

Three evaluation-time components of the pipeline rely on LLM judgment (a fourth, the query filtering at construction time, is described in Appendix B), and all their prompts are deployed in Chinese and released verbatim with the code.

POI ground-truth matching. The alignment model must cite the behavioral evidence (click, arrival, or route selection) supporting its link, so a match without evidence is rejected; driving and transit ground truths bypass judging entirely.

Explanation-quality judge (FS/CS/PS). A single judge (GLM-5.2, temperature 0) scores each explanation on three dimensions with integer 0–5 rubrics. *Faithfulness* checks whether every key claim (distance, duration, traffic, POI attributes) is supported by the tool returns provided to the judge, with 0 reserved for fabricated or contradictory reasons; the judge is instructed not to bring in external knowledge. *Completeness* checks whether the explanation states why each candidate is recommended together with the scenario-relevant attributes. *Persuasiveness* checks whether the explanation is clear and comparative enough for a user to decide. Failed or refusal answers are scored by a separate calibration in which an honest refusal can receive high scores, so the metric measures explanation quality rather than task success.

Query-difficulty labeler (QR). Each query receives a 1–5 expression-clarity level (1 = fully explicit, 5 = hardly interpretable), labeled once and reused across all frameworks. The rubric measures clarity of expression rather than execution difficulty: a query scores high when it is ambiguous or underspecified even after considering the dialogue context. The judgment anchor is the agent's real action space: the rubric enumerates the ten action categories the navigation system actually supports, so that queries whose intent does not map to a single supported action (e.g., a bare "stop" that could mean either finding a parking lot or pulling over) are rated as unclear.

E.5 Case Study

Table 14 shows a transit episode selected under the persona-necessity criterion: without any persona the case is unsolvable, whereas with persona it is solved on every backbone. The query is a speech-recognition transcript whose destination phrase matches no POI name; the only referent that resolves it is the home community stored in the user profile, and the ranking additionally follows the user's daily bus-commute preference. The ground truth is the transit plan the user selected afterwards, recovered by click-log rule matching rather than by any judge. The context-absent condition of this case-level comparison removes profile and history jointly, and is therefore stricter than the no-memory baseline of the main table, which retains the profile. Coordinates and exact timestamps are omitted here, as in the released episodes.

Table 14: A strictly persona-required transit episode: the destination is resolvable only from the profile, and the ground truth is the user's observed post-query transit choice rather than a generated reference answer.

Field	Content
User query	"I want to go from where I am now ... back to this Hailong–Hailonghong Vanke property compound ... please recommend public transit for me." (verbatim speech transcript; the destination phrase is corrupted and matches no POI name)
Request context	current location only; no resolvable destination and no prior route in the session
Personalization signals	profile: home community in Daya Bay, Huizhou, age 55–59, bus as the frequent commute mode; history: recurring bus commutes to the same community
Agent output and memory	three transit itineraries labeled fastest / cheapest / full-bus, the full-bus option justified by the commute habit; memory field: "origin taken from the current location, destination from the profile home community, ranking by the daily bus-commute preference"
Behavior-grounded GT	within the post-query window the user's recorded click selects the Daya Bay 317B bus line, i.e. the full-bus itinerary among the three recommended; recovered by click-log rule matching (transit), with no judge involved
Persona necessity	Hit@3 $= 0$ without persona on all three backbones and $= 1$ with the profile–history summary or latest-k records on all three; the episode belongs to the strictly persona-required case set