META-ORCHESTRATION OF SYMBOLIC AND TEXTUAL AGENTS FOR PERSONALIZED DECISION-MAKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Easy decisions deserve simple models, while difficult decisions may benefit from richer reasoning. Utility-based models offer efficient and interpretable predictions, whereas large language models (LLMs) incorporate contextual information at greater computational cost. We propose Orchestration of Symbolic and Textual Agents (*OSTA*), a hierarchical meta-reasoning framework that adaptively combines utility models and LLMs for personalized decision modeling. Its orchestrator comprises two complementary stages: (i) Soft Inference Path Preference (*SIPP*), which assigns soft preferences between utility-only and mixed utility–LLM inference path; and (ii) Path-Conditional Agent Reliability (*PCAR*), which estimates agents' performance reliability on each path. A value-of-computation-inspired rule combines these signals with a tunable cost penalty to jointly select the inference path and agent, enabling performance and efficiency modes. Across six real-world datasets spanning transportation, health, and socioeconomic decisions, *OSTA*'s performance mode achieves the best or tied-best performance on five of them, while efficiency mode retains an comparable performance, with approximately 91% fewer estimated tokens and 94% lower estimated inference cost for selected outputs. Population-scale case studies further illustrate *OSTA*'s potential for resource-efficient personalized decision modeling. The source code will be fully released upon paper acceptance.

1 INTRODUCTION

Easy decisions deserve simple models, and hard decisions demand powerful reasoning. This principle is central to personalized decision modeling, where the reasoning required to explain a choice varies across individuals, tasks, and contexts. A familiar commuting choice may be adequately captured by a compact utility function over travel time and cost. A decision involving conflicting preferences, changing circumstances, or long-term consequences may require richer reasoning about beliefs, constraints, and contextual information. As decision models increasingly support applications in transportation, public health, education, and economic planning (Zheng et al., 2021; Wu et al., 2025; Zhao et al., 2025a; Shaham et al., 2019; Du et al., 2025), this variation creates a computational challenge: **how can we allocate sufficient reasoning to each decision without spending additional computation where it offers little benefit?**

Symbolic decision models and large language models (LLMs) provide complementary capabilities for addressing this challenge. Utility-based models, discrete choice models, and prospect-theoretic frameworks encode decision mechanisms through explicit mathematical structure (Mueller & Pearl, 2023; Rabin, 2013; Ettema et al., 2016; Berbeglia et al., 2022; Ramos et al., 2014). They offer efficient and interpretable inference when the relevant factors and trade-offs can be represented by their attributes and functional forms. LLMs provide a more flexible reasoning interface for incorporating textual descriptions of preferences, contextual constraints, and domain knowledge, but incur additional computational cost. Using symbolic inference throughout can leave relevant context unmodeled, whereas invoking LLM reasoning for every instance spends resources even when a simple model is sufficient. Combining their strengths therefore requires an instance-level mechanism for deciding where additional reasoning is useful.

We approach this problem through rational meta-reasoning: selecting computations according to their expected benefit and cost (Lieder & Griffiths, 2017; Russell & Wefald, 1991). For personalized

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.

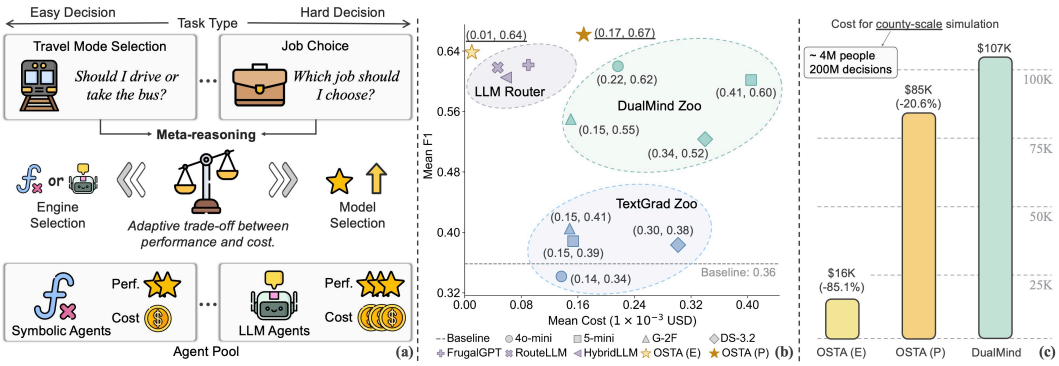


Figure 1: **Overview and performance-cost trade-off of the proposed *OSTA* framework.** 4o-mini, 5-mini, G-2F and DS-3.2 denote GPT-4o-Mini, GPT-5-Mini, Gemini-2.0-Flash and DeepSeek-V3.2, respectively. Baseline denotes the average F1 across vanilla version of these models. E and P refer to the efficiency and performance mode of *OSTA*. (a) *OSTA* realizes adaptive performance-cost trade-offs through meta-reasoning, deciding when additional reasoning is needed and which reasoning model to invoke. (b) *OSTA* (E) achieves the best accuracy-cost trade-off, while *OSTA* (P) achieves the best accuracy. (c) *OSTA* (E) reduces inference cost by 85.1% in county-scale simulation while maintaining strong predictive performance.

decision modeling, this requires answering two coupled questions: *whether to invoke additional LLM reasoning*, and *which agent to use*. The first concerns the reasoning demands of the decision; the second concerns the reliability of the available agents in meeting those demands. These questions cannot be resolved independently. A complex or unfamiliar instance may warrant further reasoning, but complexity alone does not establish that a particular agent will improve the prediction. Conversely, an agent's aggregate performance does not establish its reliability for a specific instance and inference path. Effective orchestration must therefore account jointly for decision characteristics, path-dependent agent reliability, and computational cost.

To operationalize this principle, we propose *OSTA* (Orchestration of Symbolic and Textual Agents), a multi-agent framework comprising a heterogeneous agent pool and an adaptive orchestrator. The orchestrator uses two complementary stages to inform a joint path-and-agent selection. First, Soft Inference Path Preference (*SIPP*) uses decision complexity and familiarity to assign a soft preference between utility-only inference and mixed inference that combines utility models with LLM reasoning. This stage estimates the need for additional reasoning without committing to a fixed path. Second, Path-Conditional Agent Reliability (*PCAR*) estimates each agent's predictive reliability under both paths using similarities to past successful cases, path-specific confidence signals, and historical performance. A value-of-computation (VOC)-inspired selection rule then combines path preference, agent reliability, and computational cost to choose the inference path and agent jointly. Together, these components distinguish the potential need for richer reasoning from the estimated ability of an agent to provide it.

OSTA supports a performance mode that prioritizes predictive quality and an efficiency mode that balances predictive quality with computational expenditure. These modes allow the same framework to adapt its allocation of reasoning to different resource requirements. As illustrated in Fig. 1, this adaptive allocation yields favorable performance–cost trade-offs. In county-scale simulation, the efficiency mode reduces inference cost by 85.1% while maintaining strong predictive performance. Our contributions are summarized as follows:

- We propose *OSTA*, a resource-rational framework for personalized decision modeling that dynamically orchestrates symbolic utility models and textual LLM agents, preserving the transparency of utility-based reasoning while avoiding unnecessary LLM over-reasoning.
- We design a hierarchical orchestration mechanism that combines soft path preferences and path-conditional agent reliability for joint, cost-aware path–agent selection, balancing predictive performance and computational cost.
- Across six real-world decision-making datasets spanning transportation, health, and social choice, *OSTA* demonstrates strong predictive performance while substantially reducing

inference cost and latency. Population-scale case studies further illustrate its potential for large-scale deployment.

2 Related Works

Human decision modeling. Predicting personalized human decisions requires balancing computational efficiency with the ability to capture complex, contextual behaviors. Traditional approaches, such as Discrete Choice Models (DCMs) (Lancsar et al., 2017) and utility-based functions (Ettema et al., 2016), offer mathematical rigor, high interpretability, and low latency. However, they rely on strict rationality assumptions and often struggle to capture the non-linear nuances of highly individualized contexts. Conversely, recent LLM-based architectures (OpenAI, 2025; DeepSeek-AI et al., 2025) have emerged as powerful reasoning engines capable of parsing complex preferences and executing instruction-tuned, chain-of-thought problem solving (Peng et al., 2023; Xia et al., 2024; Zhao et al., 2025b). Yet, treating every decision as a complex reasoning task introduces severe computational overhead and risks "over-reasoning" routine choices (Liu et al., 2025; Yan et al., 2025).

Model ensembling and multi-agent orchestration. To leverage the strengths of varying model architectures, the field has largely turned to model ensembling. At the parameter level, Sparse Mixture of Experts (MoE) (Jacobs et al., 1991; Chen et al., 2024; Ding et al., 2024; Ong et al., 2024) achieves an optimal balance between performance and efficiency by routing inputs to specialized sub-networks (Eigen et al., 2014; Bai et al., 2023). At the system level, Multi-Agent Systems (MAS) orchestrate distinct, role-specialized models to handle multi-step tasks (Lowe et al., 2017; Yao et al., 2023). However, existing MAS frameworks predominantly orchestrate homogeneous, heavy-weight LLM agents, compounding inference costs. Effectively bridging fast symbolic logic with heavy textual agents requires shifting from static routing to dynamic, cost-aware orchestration.

Meta-reasoning and resource-rationality. Meta-reasoning asks when costly deliberation is worthwhile (Cox & Raja, 2011). Under bounded optimality, an agent should reason only when expected decision gains exceed computational costs. This principle guides anytime planning (Hansen & Zilberstein, 2001) and reinforcement learning methods that predict the value of computation from problem features (Bhatia et al., 2021).

3 Methods

3.1 Problem Setup and Method Overview

Problem setup. We study discrete choice from a finite set $\mathcal{M} = \{1, \dots, m\}$. For each observation $(x_i, y_i) \in \mathcal{D}$, $x_i \in \mathbb{R}^d$ describes an individual and decision context, and $y_i \in \mathcal{M}$ is the observed choice. Our pool contains K heterogeneous agents. Agent k has a learned utility function f^k and a textual context $\mathcal{T}^k$, which support two inference paths:

$$\hat{y}^U_{ik} = \arg\max_{a \in \mathcal{M}} f^k_a(x_i), \qquad \hat{y}^M_{ik} = \mathcal{F}^k(x_i \mid f^k, \mathcal{T}^k), \tag{1}$$

where $f^k_a(x_i)$ is agent k's utility function for choice a, and $\mathcal{F}^k$ is its mixed inference procedure, which returns a choice using the input, learned utilities, and textual context. The utility path U predicts directly from the learned utilities; the mixed path M adds LLM reasoning (Utility+LLM). We instantiate these paths with DualMind utility-only (DM-U) and DualMind mixed (DM), respectively, based on DualMind (Yang et al., 2026). The orchestrator selects a path–agent pair from $\mathcal{A} = \{U, M\} \times \{1, \dots, K\}$.

Method overview. *OSTA* contains two main stages: Soft inference path preference (*SIPP*) assigns soft preference score between utility-only and mixed paths while path-conditional agent reliability (*PCAR*) estimates each agent's reliability of predicting correctly on each path. The final agent selection weights these reliability estimates by the path preference, subtracts the computational cost penalty, and selects the path–agent pair with the highest score.

3.2 Stage 1: Soft Inference Path Preference

Decision complexity and familiarity. For agent k and case i, decision complexity χ_{ik} summarizes the negative-log-likelihood gradient with respect to the agent's utility parameters. It consists three main components: E_{ik} (A.1 Eq. (12)) is normalized attribution entropy, measuring how broadly absolute gradient contributions are dispersed; P_{ik} (A.1 Eq. (12)) is the normalized participation

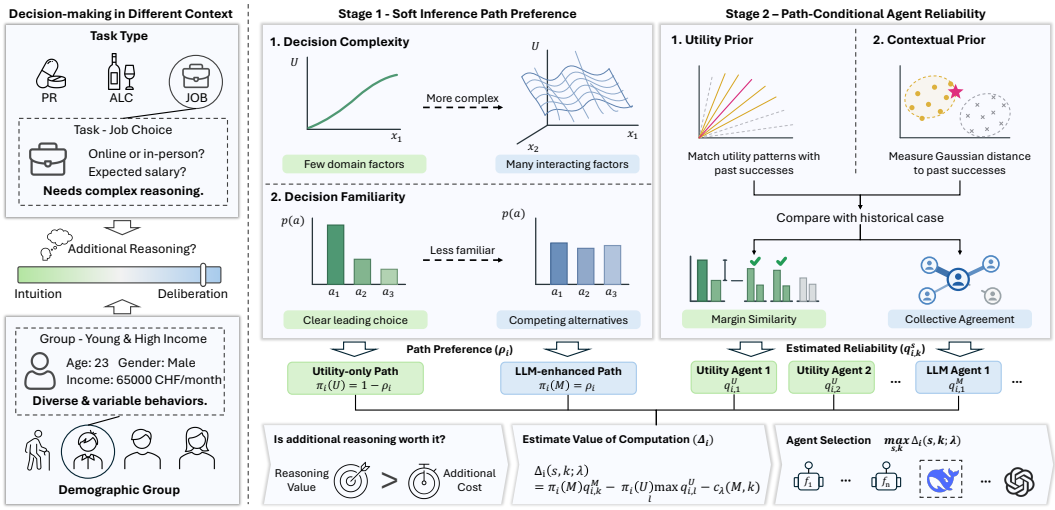


Figure 2: **System overview of *OSTA*.** *OSTA* coordinates inference through two complementary stages. Stage 1: *SIPP* uses decision complexity and familiarity to assign soft preferences to utility-only and LLM-enhanced (Utility+LLM) paths. Stage 2: *PCAR* estimates each agent's predictive reliability using utility prior and contextual similarity to past successful cases, together with margin similarity for the utility-only path or reliability-weighted collective agreement for the LLM-enhanced path. A value-of-computation-inspired selection rule combines path preferences, estimated reliability, and computational cost to jointly select the inference path and agent.

ratio, measuring the effective fraction of contributing parameters; and G_{ik} (A.1 Eq. (12)) is the mean absolute gradient magnitude, measuring overall sensitivity. For decision familiarity h_{ik}, let $p_{ik}(a)$ denote the choice probability assigned by agent k's learned utility model to choice $a \in \mathcal{M}$ for case i with input x_i. These probabilities form a distribution over the choices: $p_{ik}(a) \in [0, 1]$ and $\sum_{a \in \mathcal{M}} p_{ik}(a) = 1$. The decision familiarity refers to the largest probability in this distribution. The agent average decision complexity χ_i and familiarity h_i can be calculated as:

$$\chi_i = \frac{1}{K} \sum_{k=1}^{K} \sigma(\kappa \left[\Phi_{\mathbf{w}}(E_{ik}, P_{ik}, G_{ik}) - b\right]), \qquad h_i = \frac{1}{K} \sum_{k=1}^{K} \max_{a \in \mathcal{M}} p_{ik}(a). \tag{2}$$

Here $\Phi_{\mathbf{w}}$ is a weighted function of the three components, $\sigma(t)$ is the sigmoid function, $\kappa > 0$ controls steepness, and b sets the midpoint. Please refer to Appendix A.1 for details.

Soft inference path preference. We map decision complexity and familiarity to a soft preference score $\rho_i \in (0, 1)$ for the mixed path as: $\rho_i = S_{\boldsymbol{\theta}}(\chi_i, h_i)$, where $S_{\boldsymbol{\theta}}$ is a learned logistic regression model. We constrain the model so that, holding the other signal fixed, the preference for mixed path is nondecreasing in decision complexity and nonincreasing in decision familiarity. The model learns from calibration cases where exactly one path predicts correctly, using the successful path as the training target. Please refer to Appendix A.1 for details.

3.3 Stage 2: Path-Conditional Agent Reliability

Let $\hat{y}^s_{ik} = y_i$ denotes the event that agent k's prediction on path s matches the observed choice for case i. For each candidate (s, k), *PCAR* estimates the probability q^s_{ik} of this event by:

$$q^s_{ik} \approx P(\hat{y}^s_{ik} = y_i \mid \mathcal{E}_i, s, k), \qquad \mathcal{E}_i = \left\{ x_i, g_i, \left(p_{ik}(\cdot), \mathbf{v}_{ik}, \hat{y}^U_{ik}, \hat{y}^M_{ik}\right)_{k=1}^{K} \right\}. \tag{3}$$

where evidence $\mathcal{E}_i$ includes: g_i is demographic group (Age $\times$ Income, please refer to Sec. 4.1); $p_{ik}(\cdot)$ and $\mathbf{v}_{ik}$ denote the utility choice distribution and utility-parameter gradient of agent k, respectively; $\hat{y}^U_{ik}, \hat{y}^M_{ik}$ are its predictions on the two paths. Using these information and outcomes from calibration cases, we construct reliability features for each path–agent pair. Below, we define these features and explain how they are used to approximate q^s_{ik} at inference time for test case, in which y_i is unknown.

Reliability features. There are four designed features: utility prior u^s_{ik}, contextual prior ψ^s_{ik}, margin similarity γ_{ik} and collective agreement φ_{ik}. The utility and contextual prior mainly ask how

similar the current test case with the reference cases that candidate (s, k) predicted correctly. For a reference case j, u^s_{ik} compares the gradient directions $\mathbf{v}_{ik}$ and $\mathbf{v}_{jk}$, while ψ^s_{ik} compares the compact representations $\mathbf{e}^s_i = T_s(x_i)$ and $\mathbf{e}^s_j = T_s(x_j)$, where T_s standardizes the features and applies PCA fitted on calibration cases:

$$u^s_{ik} = \frac{1}{|\mathcal{R}^s_{ik}|} \sum_{j \in \mathcal{R}^s_{ik}} \exp\left(\frac{\cos(\mathbf{v}_{ik}, \mathbf{v}_{jk})}{\tau_v}\right), \quad \psi^s_{ik} = \frac{1}{|\mathcal{R}^s_{ik}|} \sum_{j \in \mathcal{R}^s_{ik}} K_{\tau_x}(\mathbf{e}^s_i, \mathbf{e}^s_j). \tag{4}$$

Here $\mathcal{R}^s_{ik}$ is the set of calibration-case indices that agent k predicted correctly on path s, excluding the current case i and retaining successes from its demographic group when available; otherwise, successes across groups are used (A.2 Eq. (23)). The function $\cos(\cdot, \cdot)$ denotes cosine similarity, with numerical stabilization specified in (A.2 Eq. (24)). The function K_{τ_x} is the Gaussian (RBF) kernel (A.2 Eq. (25)). The positive parameters τ_v and τ_x control sensitivity to direction and distance, respectively. Larger scores indicate greater similarity to past successes, providing higher reliability.

The margin similarity γ_{ik} and collective agreement φ_{ik} are designed for utility-only and mixed inference path, respectively. γ_{ik} is the smoothed fraction of calibration cases that agent k predicted correctly on the utility path among those with prediction margins similar to the current case's margin, while φ_{ik} denotes the fraction of total peer reliability weight contributed by other mixed-path agents that predict the same choice as agent k, where each peer's weight is its smoothed calibration accuracy:

$$\gamma_{ik} = \Gamma^U\big(p_{ik}(\cdot), k\big), \qquad \varphi_{ik} = \Gamma^M\big((\hat{y}^M_{i\ell})^K_{\ell=1}, k\big), \tag{5}$$

where Γ^U and Γ^M (A.2 are defined in Appendix A.2 Eq. (30) and Eq. (32)). We combine features for each inference path and use $\mathbf{z}^s_{ik}$ to denote the feature vector for inference path $s \in \{U, M\}$ as $\mathbf{z}^U_{ik} = (u^U_{ik}, \psi^U_{ik}, \gamma_{ik})^\top$ and $\mathbf{z}^M_{ik} = (u^M_{ik}, \psi^M_{ik}, \varphi_{ik})^\top$.

Reliability estimation. We design three functions $\mathcal{Q}^s_1$, $\mathcal{Q}^s_2$ and $\mathcal{Q}^s_3$ to estimate each agent's probability of predicting correctly:

$$q^{s,1}_{ik} = \mathcal{Q}^s_1(\mathbf{z}^s_{ik}, g_i, k), \quad q^{s,2}_{ik} = \mathcal{Q}^s_2(\mathbf{e}^s_i, \mathbf{v}_{ik}, h_{ik}, g_i, k), \quad q^{s,3}_{ik} = \mathcal{Q}^s_3(g_i, k). \tag{6}$$

The logistic model $\mathcal{Q}^s_1$ (A.3 Eq. (35)) uses logistic regression to estimate correctness from the reliability features, accounting for differences between agents and demographic groups. The local estimator $\mathcal{Q}^s_2$ (A.3 Eq. (39)) estimates reliability from the agent's performance on similar calibration cases, giving greater weight to more similar cases. The group estimator $\mathcal{Q}^s_3$ (A.3 Eq. (40)) estimates reliability from the agent's overall performance within the same demographic group, using all groups when no same-group cases are available. The final reliability estimation combines their outputs:

$$q^s_{ik} = \omega_s \Big[\alpha_s q^{s,1}_{ik} + (1 - \alpha_s) q^{s,2}_{ik}\Big] + (1 - \omega_s) q^{s,3}_{ik}. \tag{7}$$

The weights $\alpha_s, \omega_s \in [0, 1]$ are tunable parameters (A.3 Eq. (43)), in which α_s balances the feature-based and local estimation, while ω_s balances their combined estimation against the group reliability. For more details, please refer to Appendix A.2 and A.3.

3.4 Agent Selection

Value of computation. Let's define $\pi_i(M) = \rho_i$, $\pi_i(U) = 1 - \rho_i$, and $v_i(s, k) = \pi_i(s) q^s_{ik}$, which weights the agent's estimated correctness by soft inference path preference for its path: agents in U receive $(1 - \rho_i) q^U_{ik}$, and agents in M receive $\rho_i q^M_{ik}$. Thus, agents on the preferred path receive more weight, while a more reliable agent on the other path can still obtain a higher score. Let $b_i = \max_k v_i(U, k)$ be the highest score among agents inside utility-only path. The net selection advantage $\Delta_i(s, k; \boldsymbol{\lambda})$ denotes the agent's score minus the best utility-path score and its computational cost penalty and can be calculated as:

$$\Delta_i(s, k; \boldsymbol{\lambda}) = v_i(s, k) - b_i - c_{\boldsymbol{\lambda}}(s, k) \tag{8}$$

where $c_{\boldsymbol{\lambda}}(s, k)$ (A.4 Eq. (46)) is the computational cost penalty accounts for using mixed inference path and the selected agent's relative calling cost. Its nonnegative parameters $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)$ control the strength of these two penalties. The utility-only inference path receives zero penalty because it incurs no additional cost.

Agent selection and inference modes. The final prediction $\hat{y}_i$ is determined by:

$$\hat{y}_i = \hat{y}^{s^*_i}_{ik^*_i}, \qquad s.t.\ (s^*_i, k^*_i) = \arg\max_{(s,k) \in \mathcal{A}} \Delta_i(s, k; \boldsymbol{\lambda}). \tag{9}$$

Since b_i is common to every candidate, this is equivalent to maximizing $v_i(s, k) - c_{\boldsymbol{\lambda}}(s, k)$. With ties resolved in favor of the utility-only inference path, mixed inference is selected exactly when

Table 1: **Performance comparison on six datasets.** MNL refers to multinomial logit. EU denotes a utility function written by experts without optimization. NL and MXL denote nested logit and mixed logit, respectively. V, TG (TextGrad), and MAS denote vanilla LLM, textual gradient, and multi-agent system, respectively. DM denotes DualMind, while DM-U uses only the refined utility function. 4o-mini, 5-mini, G-2F, and DS-3.2 refer to GPT-4o-mini, GPT-5-mini, Gemini-2.0-Flash, and DeepSeek-V3.2, respectively. HybridLLM-C and HybridLLM-L denote class-based and label-based routing, respectively. T.N. and C. denote the average estimated token count and inference cost (10^{-3} USD) per sample, respectively. E and P denote efficiency and performance modes. Avg. denotes the unweighted average across the six datasets. Dark blue and light blue indicate the best and second-best results among all methods, respectively.

	Methods	Vaccine (VAC)				Travel Mode (SWM)				Job Choice (JOB)				Marijuana (MAJ)				Pain Reliever (PR)				Alcohol (ALC)				Average			
		F1↑	Acc↑	T.N.↓	C.↓	F1↑	Acc↑	T.N.↓	C.↓	F1↑	Acc↑	T.N.↓	C.↓	F1↑	Acc↑	T.N.↓	C.↓	F1↑	Acc↑	T.N.↓	C.↓	F1↑	Acc↑	T.N.↓	C.↓	F1↑	Acc↑	T.N.↓	C.↓
Classic	MNL-EU	0.59	0.59	–	–	0.50	0.53	–	–	0.43	0.47	–	–	0.44	0.39	–	–	0.53	0.55	–	–	0.38	0.36	–	–	0.48	0.48	–	–
	MNL-Linear	0.42	0.50	–	–	0.23	0.46	–	–	0.50	0.47	–	–	0.31	0.33	–	–	0.36	0.34	–	–	0.24	0.34	–	–	0.34	0.41	–	–
	NL	0.58	0.58	–	–	0.51	0.54	–	–	0.43	0.44	–	–	0.42	0.37	–	–	0.51	0.53	–	–	0.39	0.37	–	–	0.47	0.47	–	–
	MXL	0.57	0.58	–	–	0.34	0.46	–	–	0.44	0.48	–	–	0.48	0.43	–	–	0.52	0.54	–	–	0.38	0.40	–	–	0.46	0.48	–	–
	gplearn	0.63	0.63	–	–	0.39	0.47	–	–	0.54	0.62	–	–	0.24	0.26	–	–	0.50	0.60	–	–	0.34	0.41	–	–	0.44	0.50	–	–
	PySR	0.59	0.59	–	–	0.62	0.63	–	–	0.53	0.52	–	–	0.54	0.50	–	–	0.44	0.44	–	–	0.38	0.39	–	–	0.52	0.51	–	–
Vanilla	4o-mini-V	0.35	0.46	1,263	0.20	0.47	0.54	638	0.10	0.26	0.26	693	0.11	0.15	0.20	672	0.11	0.22	0.29	657	0.10	0.30	0.32	873	0.14	0.29	0.35	799	0.13
	5-mini-V	0.52	0.56	1,273	0.21	0.51	0.54	642	0.10	0.39	0.33	701	0.11	0.33	0.31	680	0.11	0.43	0.41	668	0.11	0.27	0.29	880	0.14	0.41	0.41	807	0.13
	G-2F-V	0.48	0.52	1,293	0.40	0.51	0.52	682	0.07	0.47	0.50	721	0.07	0.28	0.27	703	0.07	0.42	0.57	690	0.08	0.30	0.30	912	0.10	0.41	0.45	834	0.13
	DS-3.2-V	0.44	0.53	1,288	0.36	0.43	0.48	670	0.19	0.25	0.25	705	0.20	0.11	0.17	688	0.20	0.42	0.57	679	0.19	0.29	0.30	894	0.25	0.32	0.38	821	0.23
TextGrad	4o-mini-TG	0.48	0.50	1,268	0.20	0.54	0.59	639	0.10	0.32	0.30	694	0.11	0.13	0.18	673	0.11	0.29	0.32	799	0.13	0.29	0.32	972	0.17	0.34	0.37	841	0.14
	5-mini-TG	0.59	0.59	1,273	0.21	0.51	0.51	643	0.10	0.22	0.17	697	0.11	0.22	0.25	681	0.11	0.45	0.44	758	0.13	0.34	0.36	1,180	0.26	0.39	0.39	872	0.15
	G-2F-TG	0.56	0.55	1,293	0.13	0.55	0.54	860	0.09	0.47	0.47	967	0.16	0.25	0.24	926	0.10	0.41	0.36	1,526	0.30	0.19	0.25	1,078	0.11	0.41	0.40	1,108	0.15
	DS-3.2-TG	0.54	0.56	1,707	0.48	0.43	0.48	882	0.25	0.43	0.45	848	0.24	0.06	0.13	789	0.22	0.50	0.51	1,048	0.30	0.34	0.37	1,134	0.32	0.38	0.42	1,068	0.30
Agent Pool	4o-mini-DM-U	0.70	0.70	–	–	0.59	0.59	–	–	0.63	0.60	–	–	0.56	0.53	–	–	0.58	0.59	–	–	0.41	0.40	–	–	0.58	0.57	–	–
	5-mini-DM-U	0.70	0.70	–	–	0.61	0.61	–	–	0.65	0.65	–	–	0.59	0.55	–	–	0.54	0.56	–	–	0.45	0.44	–	–	0.59	0.59	–	–
	G-2F-DM-U	0.73	0.73	–	–	0.60	0.60	–	–	0.69	0.67	–	–	0.55	0.51	–	–	0.53	0.55	–	–	0.38	0.36	–	–	0.58	0.57	–	–
	DS-3.2-DM-U	0.67	0.67	–	–	0.61	0.61	–	–	0.61	0.61	–	–	0.52	0.48	–	–	0.52	0.55	–	–	0.40	0.39	–	–	0.56	0.55	–	–
	4o-mini-DM	0.75	0.75	1,488	0.22	0.69	0.69	1,480	0.23	0.67	0.65	1,377	0.21	0.62	0.60	1,449	0.22	0.53	0.53	1,358	0.21	0.46	0.48	1,368	0.21	0.62	0.62	1,420	0.22
	5-mini-DM	0.73	0.73	1,546	0.42	0.65	0.65	1,444	0.39	0.64	0.64	1,455	0.40	0.58	0.54	1,453	0.41	0.55	0.56	1,440	0.41	0.46	0.44	1,396	0.40	0.60	0.59	1,456	0.41
	G-2F-DM	0.74	0.74	1,556	0.16	0.60	0.60	1,627	0.17	0.69	0.67	1,508	0.15	0.46	0.41	1,414	0.14	0.38	0.34	1,328	0.14	0.43	0.39	1,357	0.14	0.55	0.53	1,465	0.15
	DS-3.2-DM	0.61	0.63	1,502	0.35	0.59	0.59	1,538	0.36	0.55	0.57	1,455	0.34	0.47	0.43	1,410	0.33	0.53	0.56	1,437	0.33	0.39	0.38	1,406	0.33	0.52	0.53	1,458	0.34
MAS	FrugalGPT	0.74	0.74	375	0.10	0.67	0.67	488	0.08	0.69	0.67	461	0.05	0.62	0.60	631	0.10	0.54	0.57	596	0.17	0.47	0.47	240	0.04	0.62	0.62	465	0.09
	RouteLLM	0.73	0.73	0	0.00	0.68	0.69	569	0.09	0.69	0.67	0	0.00	0.60	0.57	565	0.09	0.56	0.59	374	0.10	0.45	0.44	0	0.00	0.62	0.62	251	0.05
	HybridLLM-C	0.72	0.72	334	0.08	0.62	0.62	608	0.11	0.69	0.68	12	0.00	0.58	0.55	461	0.08	0.53	0.53	515	0.08	0.47	0.47	881	0.13	0.60	0.60	469	0.08
	HybridLLM-L	0.71	0.71	29	0.00	0.63	0.63	366	0.05	0.68	0.67	119	0.02	0.60	0.57	607	0.11	0.55	0.56	313	0.07	0.46	0.45	568	0.10	0.61	0.60	334	0.06
	Ours (E)	0.75	0.75	14	0.00	0.69	0.68	411	0.06	0.73	0.74	0	0.00	0.62	0.63	0	0.00	0.57	0.59	0	0.00	0.47	0.47	0	0.00	0.64	0.64	71	0.01
	Ours (P)	0.77	0.77	1,444	0.30	0.74	0.74	1,415	0.29	0.73	0.74	0	0.00	0.67	0.66	556	0.10	0.57	0.59	0	0.00	0.49	0.49	1,422	0.32	0.66	0.67	806	0.17

$\max_k \Delta_i(M, k; \boldsymbol{\lambda}) > 0$. For inference modes, performance mode sets $\boldsymbol{\lambda} = \mathbf{0}$, while efficiency mode selects $\boldsymbol{\lambda}$ adaptively to favor utility-only inference path use while maintaining predictive quality. Please refer to Appendix A.4. For the calibration, model fitting and parameter tuning process across *SIPP*, *PCAR* and agent selection stages, please refer to Appendix A.5.

4 Experiment

4.1 Experimental Setup

Datasets. We evaluate on six real-world human decision-making tasks: three daily socioeconomic choices: Vaccine (VAC), Travel Mode (SWM), Job Choice (JOB) and three addiction-related behaviors: Alcohol (ALC), Marijuana (MAJ), Pain Reliever (PR), each framed as a multiclass classification problem from large-scale survey data. VAC, SWM, and JOB correspond to vaccination status (*Unvaccinated, Initial, Booster*), transportation mode (*Train, Car, Swissmetro*), and labor status (*Salaried, Self-employed, Not working*). ALC, MAJ, and PR measure use of alcohol, marijuana and pain reliever in (*past 30 days, past 12 months, >12 months*). For interpretability and to reduce overfitting, we use 15-25 features per task. To enable personalized modeling, we stratify individuals into nine demographic groups defined by a 3×3 grid over Age (Young: <35, Middle: 35-65, Older: >65) and Income (Low: <50K, Middle: 50-100K, High: >100K). For each group, there are 50 training and 32 validation/test instances with balanced classes when possible. Overall, each dataset contains 876-1,008 instances, with 367-595 used for training (More details in Appendix B).

Model selection. We assess the performance of *OSTA* alongside different baseline groups: 1) Classical utility-based methods including Multinomial Logit (MNL) (Huang et al., 2024), Nested Logit (NL) (Börsch-Supan, 1990), Mixed Logit (MXL) (Hole, 2007); 2) LLM-based methods: Vanila, TextGrad (Yuksekgonul et al., 2025) and DualMind (DM) (Yang et al., 2026). For DM, we have two inference settings: utility function only (DM-U) and utility+LLM (DM). For DM-U, it only employs an MNL model with utility functions refined through DM's pipeline. For DM, it makes predictions based on both the optimized utility functions and knowledge set. For all LLM-based methods, we evaluate them on GPT-4o-mini (Hurst et al., 2024), GPT-5-mini (OpenAI, 2025), Gemini-2.0-

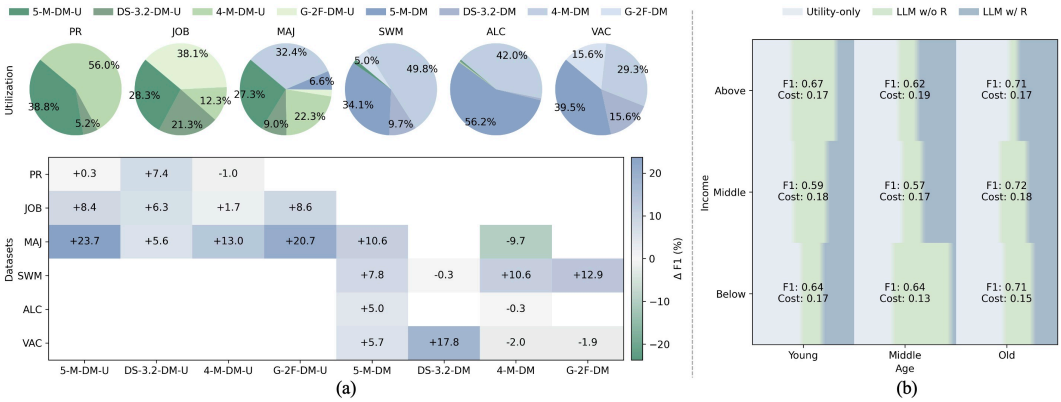


Figure 3: **Agent utilization, orchestration effectiveness, and demographic heterogeneity using performance mode.** (a, top) Agent utilization rates across datasets. (a, bottom) ΔF1, defined as the difference between each agent's average F1 score on the samples assigned to it and its average F1 score on the full test set. A larger Δ F1 indicates that the orchestrator routes samples to agents whose strengths are well aligned with those samples, demonstrating the ability of *OSTA* to exploit complementary agent capabilities effectively. For each dataset, we only show agents assigned with ≥ 5 samples. (b) Accuracy and average per-sample inference cost (10^{-3} USD) across age and income groups, aggregated over six datasets. We also report the average utilization rates of utility-only and mixed agents with reasoning (LLM w/ R) and without reasoning (LLM w/o R).

Flash (Gemini et al., 2023), DeepSeek-V3.2 (DeepSeek-AI et al., 2025). The heterogeneous agent pool of our proposed *OSTA* consists all DM and DM-U models. We adapt three established LLM-only routers—HybridLLM (Ding et al., 2024), FrugalGPT (Chen et al., 2024), and RouteLLM (Ong et al., 2024) (Details in Appendix D)—to select between utility-only and utility+LLM agents from the same agent pool, and evaluate them under the same experimental protocol.

Evaluation metrics. We use F1-score (F1), accuracy (Acc), token number and price in U.S. dollar (USD). The detailed calculation of these metrics are illustrated in Appendix C.

4.2 Results

State-of-the-art performance with computational efficiency. As shown in Tab. 1, *OSTA* in performance mode achieves the best or tied-best F1 and accuracy on five of the six datasets, trailing the best PR results by only 0.01 on both metrics. Fig. 3(a) bottom provides evidence of effective orchestration: different agents are utilized across datasets, and 17 of the 23 reported agent–dataset pairs exhibit positive ΔF1, indicating that agents generally perform better on their assigned samples than on the full test set. Efficiency mode offers a favorable computational trade-off, achieving an average F1 of 0.64 versus 0.66 in performance mode while reducing average token usage by approximately 91% and inference cost by 94%. It requires no LLM inference on JOB, MAJ, PR, and ALC, and achieves the lowest or tied-lowest token usage and cost among the evaluated routing methods across all six datasets. Together, these results demonstrate that *OSTA* combines strong predictive performance with substantial reductions in LLM inference requirements.

Statistical analysis. Our statistical analysis covers 60K bootstrapped test samples (10K per dataset) and reports 95% confidence intervals with Holm-corrected significance tests. As shown in Fig. 4, the performance mode achieves the highest average F1 and significantly outperforms all baselines ($p < 0.0001$), while the efficiency mode outperforms all but the strongest DM baseline using approximately 94% fewer tokens and incurring 95% lower cost than that baseline.

Ablation study. We conduct ablation study at both the component and stage levels across all six datasets in Tab. 2. For component-level ablation, removing any component reduces aggregate performance, with γ and φ producing the largest declines, highlighting the importance of calibrated utility-path reliability and mixed-path agreement. Among the routing cues, familiarity h contributes more than decision complexity χ, while the alignment priors u and ψ provide smaller but consistent gains. For stage-level ablation, the complete *SIPP +PCAR* (Ours) architecture performs best, whereas random routing at both stages performs worst. Replacing either orchestrator with random or validation-best selection consistently degrades performance, demonstrating the importance of instance-adaptive, hierarchical routing.

Table 2: **Ablation studies at the component and stage levels (performance mode).** At the component level, χ and h denote decision complexity and familiarity, respectively; u is the utility alignment prior, ψ is the contextual similarity prior, γ is the utility-path calibrated-margin reliability, and φ is the mixed-path reliability-weighted agreement. At the stage level, R denotes random selection of the mode or agent, while B denotes selection of the best mode or agent based on the average F1-score and accuracy on the validation set. *SIPP* and *PCAR* refer to the paradigm orchestrator and deliberation orchestrator, respectively. Average means unweighted averages across all six datasets.

Type	Methods	VAC		SWM		JOB		MAJ		PR		ALC		Average	
		F1↑	Acc↑	F1↑	Acc↑	F1↑	Acc↑	F1↑	Acc↑	F1↑	Acc↑	F1↑	Acc↑	F1↑	Acc↑
Component-Level	w/o χ	0.76	0.76	0.71	0.71	0.73	0.74	0.65	0.65	0.55	0.57	0.47	0.47	0.65	0.65
	w/o h	0.77	0.77	0.72	0.72	0.72	0.73	0.64	0.63	0.53	0.55	0.46	0.46	0.64	0.64
	w/o u	0.73	0.73	0.74	0.74	0.72	0.73	0.63	0.62	0.57	0.59	0.49	0.49	0.65	0.65
	w/o ψ	0.76	0.76	0.73	0.73	0.71	0.72	0.67	0.66	0.54	0.55	0.47	0.47	0.65	0.65
	w/o γ (utility)	0.77	0.77	0.68	0.68	0.69	0.70	0.62	0.61	0.53	0.55	0.49	0.49	0.63	0.63
	w/o φ (mixed)	0.74	0.74	0.65	0.64	0.73	0.74	0.64	0.63	0.57	0.59	0.44	0.45	0.63	0.63
Stage-Level	R+R	0.71	0.71	0.62	0.62	0.64	0.63	0.55	0.51	0.52	0.53	0.43	0.41	0.58	0.57
	R+B	0.75	0.75	0.65	0.65	0.73	0.72	0.61	0.59	0.57	0.58	0.48	0.47	0.63	0.63
	R+*PCAR*	0.76	0.76	0.70	0.70	0.73	0.74	0.62	0.60	0.56	0.58	0.48	0.48	0.64	0.64
	B+R	0.70	0.70	0.64	0.64	0.64	0.63	0.56	0.52	0.54	0.56	0.42	0.41	0.59	0.58
	B+B	0.75	0.75	0.68	0.67	0.73	0.73	0.61	0.60	0.58	0.60	0.49	0.49	0.64	0.64
	B+*PCAR*	0.75	0.75	0.72	0.72	0.73	0.74	0.64	0.63	0.57	0.59	0.48	0.48	0.65	0.65
	SIPP +R	0.72	0.71	0.64	0.63	0.64	0.63	0.56	0.52	0.54	0.56	0.44	0.42	0.59	0.58
	SIPP +B	0.77	0.77	0.67	0.67	0.73	0.72	0.63	0.63	0.58	0.60	0.49	0.49	0.64	0.65
	Ours	0.77	0.77	0.74	0.74	0.73	0.74	0.67	0.66	0.57	0.59	0.49	0.49	0.66	0.67

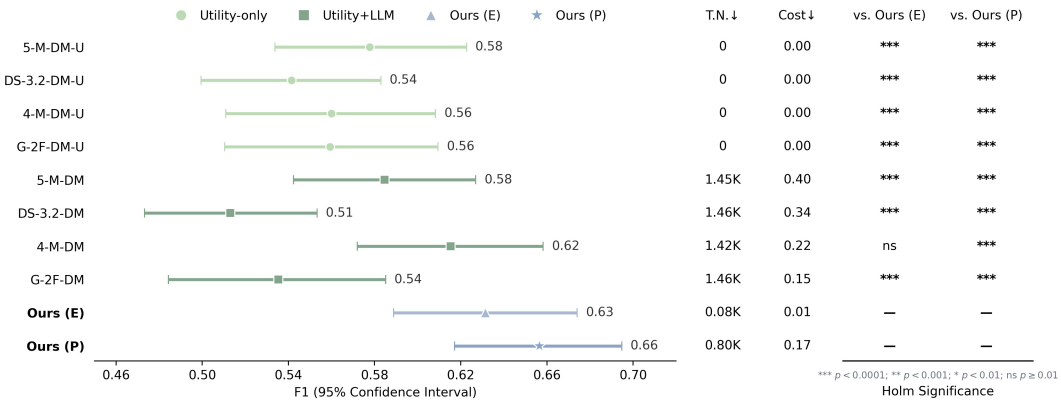


Figure 4: **Performance comparison and statistical analysis.** We report results across six datasets comprising 60K test samples, together with 95% bootstrap confidence intervals and Holm-corrected significance tests. The Holm significance indicator denotes whether each baseline differs significantly from OSTA based on p-values adjusted for multiple comparisons. T.N. denotes the number of tokens, while Cost represents the average inference cost per sample in units of 10^{-3} USD. E and P denote the Efficiency and Performance modes of *OSTA*, respectively.

5 Findings and Discussion

Over-reasoning in LLMs can be detrimental for personalization. While internal reasoning mechanisms in LLMs can improve performance in some tasks, they may interfere with utility-function optimization and knowledge-context integration in personalized decision making. As shown in Tab. 1, GPT-4o-mini-DM consistently outperforms GPT-5-mini-DM across all datasets except PR. Moreover, for reasoning-capable models (*e.g.*, GPT-5-mini and DeepSeek-V3.2), integrating optimized knowledge context can degrade performance relative to their utility-based variants. In contrast, for non-reasoning models (*e.g.*, Gemini-2.0-Flash and GPT-4o-mini), incorporating optimized knowledge context consistently improves or maintains performance.

Behavioral Heterogeneity across Decision Domains and Demographic Groups. Fig. 3 illustrates how predictive performance and inference allocation vary across decision domains and demographic groups. In Fig. 3(a) top, PR relies entirely on utility-only agents, while MAJ assigns approximately 59% of cases to this path, consistent with learned utility functions capturing regularities in substance-use behavior. Conversely, SWM and VAC exclusively employ mixed agents, suggesting benefits from contextual interpretation of travel trade-offs and vaccination beliefs. The contrasting allocations for JOB (utility-only) and ALC (mixed) indicate that broad domain categories alone do not explain

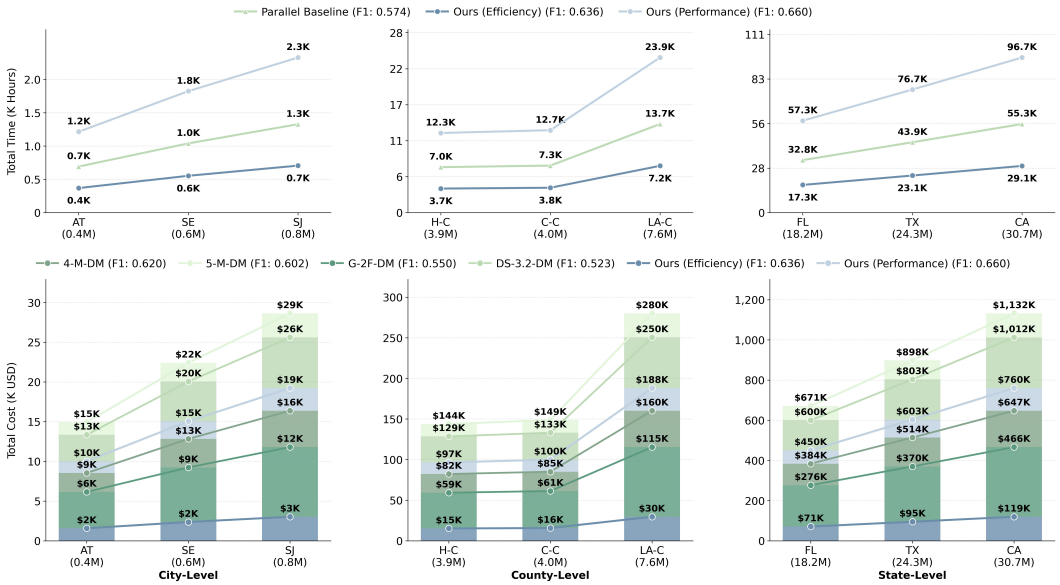


Figure 5: **Demographic heterogeneity and large-scale deployment study.** Total inference time and cost for a population-scale case study at city, county, and state levels in the U.S., comparing *OSTA* with LLM-based baselines. The F1 denotes the average F1 across six datasets illustrated in Tab. 1. AT, SE, and SJ denote Atlanta, Seattle, and San Jose; H-C, C-C, and LA-C denote Harris County, Cook County, and Los Angeles County; FL, TX, and CA denote Florida, Texas, and California. Detailed experiment settings in Appendix E.

these patterns. Across demographic groups, Fig. 3(b) shows that older individuals exhibit greater utility-only agents utilization than younger individuals at the same income level and consistently higher F1 scores, suggesting that their observed choices are more readily captured by structured utility functions. These findings connect *OSTA* 's inference allocation to variation in how effectively available agents represent domain- and group-specific behavioral patterns.

Scalable personalized decision modeling. We evaluate *OSTA* at city-, county-, and state-level population scales, comparing total inference cost and time against LLM-based baselines (Fig. 5), detailed settings in Appendix E. Inference time is estimated under a throughput limit of 5,000 tokens/s per LLM, with a parallel baseline distributing the workload across four LLMs. Efficiency mode consistently achieves the lowest cost and shortest inference time across all scales, while attaining an average F1 of 0.636, exceeding all individual LLM-based baselines (0.523–0.620). At the California scale, it reduces total cost to \$119K, approximately 74–89% below the individual baselines, and inference time to 29.1K hours, approximately 47% below the parallel baseline. Performance mode further improves F1 to 0.660, but requires greater expenditure and longer inference time; at the same scale, its cost and time reach \$760K and 96.7K hours, respectively. These results demonstrate a clear trade-off between predictive performance and computational expenditure, with efficiency mode of *OSTA* offering a particularly favorable balance for population-scale personalized decision modeling.

6 Conclusion and Limitations

In this work, we presented *OSTA*, an adaptive hierarchical multi-agent system that adaptively orchestrates utility-based and LLM-based agents for personalized decision making. It first uses decision complexity and familiarity to estimate the need for LLM reasoning, then combines path-conditional agent reliability with inference cost to select a path and agent. Across six real-world datasets, the performance mode achieved an average F1 of 0.662 and the best or tied-best results on five tasks. The efficiency mode retained an average F1 of 0.638 while using approximately 91% fewer tokens and incurring 94% lower inference cost than the performance mode. Population-scale cost projections further suggest that selective LLM use can make inference more practical at scale. A limitation is that utility representations are learned at the demographic-group level and remain fixed after training. We will explore dynamically personalized utility representations in the future.

AI USE STATEMENT

In this work, we used generative AI tools for assist in the writing of proofs, design or provide feedback on research methodology or experiments, implement methods, assist with translation, support qualitative and thematic data analysis, interpret results. We have not used generative AI tools for generate synthetic data sets, help develop theoretical models or conceptual frameworks, propose or refine hypotheses, clean and reformat dataset and formulate mathematical claims, provide critical ingredients for proving mathematical claims are not applicable to this work. We have reviewed all AI-assisted work. We thoroughly checked all LLM-generated texts inside this paper and LLM-generated code was verified and tested for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This paper has no ethical concerns.

REPRODUCIBILITY STATEMENT

The code will be open sourced upon acceptance.

REFERENCES

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report, 2023. URL https://arxiv.org/abs/2309.16609.

Gerardo Berbeglia, Agustín Garassino, and Gustavo Vulcano. A comparative empirical study of discrete choice models in retail operations. *Management Science*, 68(6):4005–4023, 2022.

Abhinav Bhatia, Justin Svegliato, and Shlomo Zilberstein. Tuning the hyperparameters of anytime planning: A deep reinforcement learning approach. In *ICAPS 2021 Workshop on Heuristics and Search for Domain-independent Planning*, 2021.

Axel Börsch-Supan. On the compatibility of nested logit models with utility maximization. *Journal of Econometrics*, 43(3):373–388, 1990. ISSN 0304-4076. doi: https://doi.org/10.1016/0304-4076(90)90126-E. URL https://www.sciencedirect.com/science/article/pii/030440769090126E.

Lingjiao Chen, Matei Zaharia, and James Zou. Frugalgpt: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024.

Michael T Cox and Anita Raja. *Metareasoning: Thinking about thinking*. MIT Press, 2011.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, and et al. Deepseek-v3.2: Pushing the frontier of open large language models, 2025. URL https://arxiv.org/abs/2512.02556.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Ruhle, Laks V. S. Lakshmanan, and Ahmed Hassan Awadallah. Hybrid llm: Cost-efficient and quality-aware query routing, 2024. URL https://arxiv.org/abs/2404.14618.

Hongru Du, Yang Zhao, Jianan Zhao, Shaochong Xu, Xihong Lin, Yiran Chen, Lauren M. Gardner, and Hao Frank Yang. Advancing real-time infectious disease forecasting using large language models. *Nature Computational Science*, 5(6):467–480, 2025. doi: 10.1038/s43588-025-00798-6.

David Eigen, Marc'Aurelio Ranzato, and Ilya Sutskever. Learning factored representations in a deep mixture of experts, 2014. URL https://arxiv.org/abs/1312.4314.

Dick Ettema, Margareta Friman, Tommy Gärling, and Lars E. Olsson. *Travel Mode Use, Travel Mode Shift and Subjective Well-Being: Overview of Theories, Empirical Findings and Policy Implications*, pp. 129–150. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016. ISBN 978-3-662-48184-4. doi: 10.1007/978-3-662-48184-4_7. URL https://doi.org/10.1007/978-3-662-48184-4_7.

Team Gemini, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Eric A Hansen and Shlomo Zilberstein. Monitoring and control of anytime algorithms: A dynamic programming approach. *Artificial Intelligence*, 126(1-2):139–157, 2001.

Arne Risa Hole. Fitting mixed logit models by using maximum simulated likelihood. *The Stata Journal*, 7(3):388–401, 2007. doi: 10.1177/1536867X0700700306. URL https://doi.org/10.1177/1536867X0700700306.

Pengpeng Huang, Lei Gong, Tian Lei, Jia Wang, Cheng Zhu, and Zheng Zhang. A comparative study of mnl and machine learning methods for travel mode choice of medical travel. In *International Conference on Traffic and Transportation Studies*, pp. 305–313. Springer, 2024.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. doi: 10.1162/neco.1991.3.1.79.

Emily Lancsar, Denzil G Fiebig, and Arne Risa Hole. Discrete choice experiments: a guide to model specification, estimation and software. *Pharmacoeconomics*, 35:697–716, 2017.

Falk Lieder and Thomas L Griffiths. Strategy selection as rational metareasoning. *Psychological review*, 124(6):762, 2017.

Ryan Liu, Jiayi Geng, Addison J. Wu, Ilia Sucholutsky, Tania Lombrozo, and Thomas L. Griffiths. Mind your step (by step): Chain-of-thought can reduce performance on tasks where thinking makes humans worse. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025. URL https://arxiv.org/abs/2410.21333.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6382–6393, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Scott Mueller and Judea Pearl. Personalized decision making–a conceptual introduction. *Journal of Causal Inference*, 11(1):20220050, 2023.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M Waleed Kadous, and Ion Stoica. Routellm: Learning to route llms with preference data, 2024. URL https://arxiv.org/abs/2406.18665.

OpenAI. Gpt-5 system card. Technical report, OpenAI, August 2025. URL https://cdn.openai.com/gpt-5-system-card.pdf.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*, 2023.

Matihew Rabin. Risk aversion and expected-utility theory: A calibration theorem. In *Handbook of the fundamentals of financial decision making: Part I*, pp. 241–252. World Scientific, 2013.

Giselle Moraes Ramos, Winnie Daamen, and Serge Hoogendoorn and. A state-of-the-art review: Developments in utility theory, prospect theory and regret theory to investigate travellers' behaviour in situations involving travel time uncertainty. *Transport Reviews*, 34(1):46–67, 2014. doi: 10.1080/01441647.2013.856356. URL https://doi.org/10.1080/01441647.2013.856356.

Stuart Russell and Eric Wefald. Principles of metareasoning. *Artificial intelligence*, 49(1-3):361–395, 1991.

Adir Shaham, Gabriel Chodick, Varda Shalev, and Dang Yamin. Personal and social patterns predict influenza vaccination decision. *BMC Public Health*, 20, 2019. URL https://api.semanticscholar.org/CorpusID:211089201.

Guanlin Wu, Boyan Su, Yang Zhao, Pu Wang, Yichen Lin, and Hao Frank Yang. Towards physics-informed spatial intelligence with human priors: An autonomous driving pilot study. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

Yu Xia, Rui Wang, Xu Liu, Mingyan Li, Tong Yu, Xiang Chen, Julian McAuley, and Shuai Li. Beyond chain-of-thought: A survey of chain-of-x paradigms for llms. *arXiv preprint arXiv:2404.15676*, 2024.

Kai Yan, Yufei Xu, Zhengyin Du, Xuesong Yao, Zheyu Wang, Xiaowen Guo, and Jiecao Chen. Recitation over reasoning: How cutting-edge language models can fail on elementary school-level reasoning problems? In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (IJCNLP-AACL)*, 2025. URL https://aclanthology.org/2025.ijcnlp-long.17/.

Hao Frank Yang, Yang Zhao, Yichen Lin, and Takeru Igusa. Thinking, fast and slow in ai agents for personalized decision-making. 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Pan Lu, Zhi Huang, Carlos Guestrin, and James Zou. Optimizing generative ai by backpropagating language model feedback. *Nature*, 639(8055):609–616, 2025.

Yang Zhao, Pu Wang, Yibo Zhao, Hongru Du, and Hao Frank Yang. Safetraffic copilot: adapting large language models for trustworthy traffic safety assessments and decision interventions. *Nature Communications*, 16(1):8846, 2025a. doi: 10.1038/s41467-025-64574-w.

Yibo Zhao, Yang Zhao, Hongru Du, and Hao Frank Yang. Personalized decision modeling: Utility optimization or textualized-symbolic reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL https://openreview.net/forum?id=RDt0crdC7N.

Yunhan Zheng, Shenhao Wang, and Jinhua Zhao. Equality of opportunity in travel behavior prediction with deep neural networks and discrete choice models. *Transportation Research Part C: Emerging Technologies*, 132:103410, 2021.

Meta-Orchestration of Symbolic and Textual Agents for Personalized Decision-making

Supplementary Material

CONTENTS

A Technical Details for the Two-Stage Orchestrator **14**
A.1 Stage 1: Soft Inference Path Preference 14
A.2 Stage 2: Reliability Features 15
A.3 Stage 2: Path-Conditional Agent Reliability 18
A.4 Agent Selection and Inference Modes 20
A.5 Calibration, Model Fitting, and Parameter Tuning 21

B Dataset details **22**

C Evaluation Metrics **22**

D Baseline Adaptations **22**
D.1 FrugalGPT 22
D.2 RouteLLM 23
D.3 HybridLLM 23

E Large-scale Personalized Decision Making Deployment **23**

A TECHNICAL DETAILS FOR THE TWO-STAGE ORCHESTRATOR

We detail the three parts of *OSTA* introduced in the method: Stage 1, soft inference path preference; Stage 2, path-conditional agent reliability; and agent selection. The method equations are repeated beside their detailed definitions so that this appendix can be read independently. Indices i, k, and $s \in \{U, M\}$ denote cases, agents, and paths. Let $\mathcal{V}$ denote the pooled set of validation-case indices used for calibration; $\mathcal{V}_s$ denotes the indices available to a particular path fit. Superscripts $1, 2, 3$ identify the logistic, local, and group reliability estimates. A check accent, as in $\check{q}^s_{ik}$, denotes a prediction constructed from cross-fitted components. The fitting order and the distinction between component cross-fitting and subsequent calibration are given in Appendix A.5.

Problem setup and inference paths. An input $x_i \in \mathbb{R}^d$ describes an individual and decision context, and $y_i \in \mathcal{M} = \{1, \dots, m\}$ is the observed choice. Each of the K agents provides a learned utility function f^k and textual context $\mathcal{T}^k$. The two predictions are

$$\hat{y}^U_{ik} = \arg\max_{a \in \mathcal{M}} f^k_a(x_i), \qquad \hat{y}^M_{ik} = \mathcal{F}^k(x_i \mid f^k, \mathcal{T}^k), \tag{10}$$

Here $f^k_a(x_i)$ is the utility for choice a, and $\mathcal{F}^k$ is the mixed inference procedure. The utility-only path U uses DualMind utility-only (DM-U), while the mixed path M uses DualMind mixed (DM), which adds LLM reasoning. The candidate set is $\mathcal{A} = \{U, M\} \times \{1, \dots, K\}$.

A.1 STAGE 1: SOFT INFERENCE PATH PREFERENCE

Utility sensitivity. Let $\vartheta^k \in \mathbb{R}^{D_k}$ contain agent k's D_k utility parameters, and let $p_{ik}(a)$ denote its utility-based probability of choice a. For each case and agent, $p_{ik}(a) \in [0, 1]$ and $\sum_{a \in \mathcal{M}} p_{ik}(a) = 1$. The sensitivity vector and its normalized absolute entries are

$$\mathbf{v}_{ik} = -\nabla_{\vartheta^k} \log p_{ik}(\hat{y}^U_{ik}), \qquad \nu_{ikt} = \frac{|v_{ikt}|}{\|\mathbf{v}_{ik}\|_1 + \epsilon}. \tag{11}$$

Here $t \in \{1, \dots, D_k\}$ indexes a utility parameter, and v_{ikt} is the corresponding entry of $\mathbf{v}_{ik}$. The predicted choice is held fixed when taking the gradient. Thus, both paths can use the same utility sensitivity without consulting y_i. Throughout this appendix, $\epsilon = 10^{-12}$ stabilizes numerical division. The gradient summary has three components:

$$E_{ik} = -\frac{\sum_{t=1}^{D_k} \nu_{ikt} \log(\nu_{ikt} + \epsilon)}{\log D_k + \epsilon},$$
$$P_{ik} = \frac{\|\mathbf{v}_{ik}\|_2^4}{D_k(\|\mathbf{v}_{ik}\|_4^4 + \epsilon)}, \qquad G_{ik} = \frac{\|\mathbf{v}_{ik}\|_1}{D_k}. \tag{12}$$

Here E_{ik} is normalized attribution entropy, P_{ik} is the normalized participation ratio, and G_{ik} is the mean absolute gradient magnitude. They measure the dispersion, effective fraction, and overall magnitude of the gradient contributions, respectively. The weighted aggregation function is

$$\Phi_{\mathbf{w}}(E_{ik}, P_{ik}, G_{ik}) = w_1 E_{ik} + w_2 P_{ik} + w_3 G_{ik}, \qquad \mathbf{w} = (0.5, 0.3, 0.2), \tag{13}$$

The per-agent decision complexity and familiarity are

$$\chi_{ik} = \sigma(\kappa[\Phi_{\mathbf{w}}(E_{ik}, P_{ik}, G_{ik}) - b]), \qquad h_{ik} = \max_{a \in \mathcal{M}} p_{ik}(a), \tag{14}$$

where σ is the sigmoid function, $\kappa = 5$ controls steepness, and $b = 0.5$ sets the midpoint. A zero gradient is assigned $\chi_{ik} = 0$; this convention also applies to its contribution to the average below. The resulting decision-complexity score depends on the utility parameterization and describes sensitivity. Decision familiarity measures predictive concentration; a high value alone does not establish correctness.

Agent-average decision complexity and familiarity. Stage 1 uses the averages over all K agents.

$$\chi_i = \frac{1}{K} \sum_{k=1}^{K} \sigma(\kappa\,[\Phi_{\mathbf{w}}(E_{ik}, P_{ik}, G_{ik}) - b]), \qquad h_i = \frac{1}{K} \sum_{k=1}^{K} \max_{a \in \mathcal{M}} p_{ik}(a). \tag{15}$$

Signal standardization. The quantities χ_i and h_i in Eq. (15) average the per-agent decision complexity and familiarity defined in Eq. (14). For a scalar signal t, write $\widetilde{t}_i = (t_i - \mu_t)/s_t$, where μ_t and s_t are its mean and standard deviation on the current fitting population. We replace s_t by one if it is smaller than 10^{-8}. This convention also applies component-wise to the Stage 2 reliability features.

Soft inference path preference. The preference for the mixed path is

$$\rho_i = S_{\boldsymbol{\theta}}(\chi_i, h_i), \tag{16}$$

The complete logistic regression function, parameterized by $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2)^\top$, is

$$S_{\boldsymbol{\theta}}(\chi_i, h_i) = \frac{1}{1 + \exp\Big[-\Big(\theta_0 + \theta_1 \widetilde{\chi}_i + \theta_2 \widetilde{h}_i\Big)\Big]}, \qquad \theta_1 \geq 0, \quad \theta_2 \leq 0. \tag{17}$$

Thus $\rho_i \in (0, 1)$. Holding the other input fixed, its preference for the mixed path is nondecreasing in decision complexity and nonincreasing in decision familiarity.

Learning from decisive cases. Let $\check{q}^s_{ik}$ be the reliability obtained by pooling cross-fitted components for candidate (s, k), and define the selected agent and its correctness within each path:

$$\check{k}^s_i = \arg\max_k \check{q}^s_{ik}, \qquad R^s_i = \mathbb{1}[\hat{y}^s_{i\check{k}^s_i} = y_i], \qquad \mathcal{I} = \{i \in \mathcal{V} : R^U_i \neq R^M_i\}. \tag{18}$$

For $i \in \mathcal{I}$, the target R^M_i is one precisely when the mixed path succeeds and the utility path fails. Cases where both paths succeed or both fail carry no preference label.

We use binary cross-entropy $\ell(t, p) = -t \log p - (1-t)\log(1-p)$ for binary outcome t and predicted probability p. The router solves

$$\boldsymbol{\theta}^* = \arg \min_{\theta_0 \in \mathbb{R},\ \theta_1 \geq 0,\ \theta_2 \leq 0} \left\{ \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \ell(R^M_i, S_{\boldsymbol{\theta}}(\chi_i, h_i)) + \lambda_\theta(\theta_1^2 + \theta_2^2) \right\}, \qquad \lambda_\theta = 0.05. \tag{19}$$

The coefficient λ_θ controls regularization of the two slopes. The intercept θ_0 is initialized to the log-odds $\log[(1 + \sum_{i \in \mathcal{I}} R^M_i)/(1 + \sum_{i \in \mathcal{I}}(1 - R^M_i))]$ of the smoothed success rate. After fitting, $\boldsymbol{\theta}$ in the method denotes $\boldsymbol{\theta}^*$. If $\mathcal{I}$ is empty, all three coefficients are set to zero, giving $\rho_i = 1/2$.

A.2 Stage 2: Reliability Features

Stage 2 estimates each candidate's probability of predicting correctly on a specified inference path.

$$q^s_{ik} \approx P(\hat{y}^s_{ik} = y_i \mid \mathcal{E}_i, s, k), \qquad \mathcal{E}_i = \Big\{ x_i, g_i, \big(p_{ik}(\cdot), \mathbf{v}_{ik}, \hat{y}^U_{ik}, \hat{y}^M_{ik}\big)_{k=1}^K \Big\}. \tag{20}$$

The event $\hat{y}^s_{ik} = y_i$ means that agent k predicts the observed choice correctly on path s. The current-case information $\mathcal{E}_i$ contains the input x_i, demographic group g_i, and each agent's utility choice distribution, utility-parameter gradient, and predictions on the two paths. Demographic groups are defined by Age $\times$ Income. The observed choice y_i is unknown for a test case; labeled calibration cases supply the reference outcomes. Mixed predictions are available in the cached-inference setting described in Appendix A.5.

Four reliability features are defined across the two paths: utility prior u^s_{ik}, contextual prior ψ^s_{ik}, margin similarity γ_{ik}, and collective agreement φ_{ik}. Each path uses three features: the two priors and its corresponding path-specific feature. The utility-only path uses margin similarity, and the mixed path uses collective agreement. Appendix A.3 explains how they inform q^s_{ik}. Throughout this subsection, j indexes a reference case, whereas ℓ indexes a peer agent.

Reference populations and correctness. Let $\mathcal{V}_s$ be the set of calibration-case indices available for fitting path s in the current fit. For example, during cross-fitting it contains only the cases outside the held-out fold. Write $\mathcal{V}_s^{-i} = \mathcal{V}_s \setminus \{i\}$ for the same set with case i removed: $\setminus\{i\}$ denotes set removal, and the superscript $-i$ is shorthand for this exclusion. This prevents the current case's own outcome from contributing to its reference evidence. If i is already held out or belongs to the test set, it is absent from $\mathcal{V}_s$, so the removal leaves the set unchanged.

For any reference case j, r^s_{jk} records whether agent k predicted its observed choice correctly using path s. For a set J of reference indices, $\mu^s_k(J)$ summarizes that candidate's correctness on J:

$$r^s_{jk} = \mathbb{1}[\hat{y}^s_{jk} = y_j], \qquad \mu^s_k(J) = \frac{1 + \sum_{j \in J} r^s_{jk}}{2 + |J|}. \tag{21}$$

Here $\mathbb{1}[\cdot]$ is one when the event in brackets holds and zero otherwise. Thus r^s_{jk} is a binary outcome, $\sum_{j \in J} r^s_{jk}$ counts correct predictions, and $|J|$ is the number of reference cases. The function μ^s_k is a smoothed correctness rate: adding one success and one failure corresponds to a Beta$(1, 1)$ prior and pulls small-sample estimates toward $1/2$. For example, eight correct predictions among ten references give $(1 + 8)/(2 + 10) = 0.75$. An empty set gives $\mu^s_k(J) = 1/2$. We reuse this same function for margin calibration, peer weights, and group reliability, with a different reference set for each purpose.

Choosing the successful-case memory. The demographic-group label g_j identifies the group of reference case j, and g_i identifies the current case's group. For any reference set J, let $J_g = \{j \in J : g_j = g\}$ be the subset belonging to group g. The function $\mathcal{G}_i$ chooses a reference set as follows:

$$\mathcal{G}_i(J) = \begin{cases} J_{g_i}, & J_{g_i} \neq \emptyset, \\ J, & J_{g_i} = \emptyset. \end{cases} \tag{22}$$

It returns cases from the current case's group if that subset is nonempty; otherwise it returns the entire supplied set J. The symbol $\emptyset$ denotes the empty set. Thus $\mathcal{G}_i(J)$ is a set of case indices, not a numerical score or a learned model.

The memory $\mathcal{R}^s_{ik}$ contains reference cases that candidate (s, k) predicted correctly. To construct it, first retain that candidate's successes and then apply the group-selection function:

$$\mathcal{R}^s_{ik} = \mathcal{G}_i\big(\{j \in \mathcal{V}^{-i}_s : r^s_{jk} = 1\}\big). \tag{23}$$

This order matters: a group with no successful reference uses successes from other groups, even if the group contains unsuccessful cases. The indices on $\mathcal{R}^s_{ik}$ reflect all three dependencies: the current case determines the group and exclusion, and the agent and path determine which past predictions count as successes. The memory may therefore differ between agents and between the two paths.

Two reusable similarity kernels. We compare the current case i with each reference case j in two ways: their input contexts and their utility-gradient directions. For contextual comparison, we write the context representation as $\mathbf{e}^s_i = T_s(x_i)$. Here x_i is the input feature vector, and T_s is a transformation that first centers and rescales the input features, then uses principal component analysis (PCA) to reduce the number of features while retaining the main patterns of variation. Both steps are fitted on the calibration cases in $\mathcal{V}_s$. The resulting vector $\mathbf{e}^s_i$ is a compact representation used to compare contexts. We apply the same fitted transformation to each reference case, giving $\mathbf{e}^s_j = T_s(x_j)$.

For gradient comparison, $\mathbf{v}_{ik}$ and $\mathbf{v}_{jk}$ are agent k's utility-parameter gradients on the current and reference cases, defined in Eq. (11). The cosine similarity used for the utility prior includes numerical stabilization:

$$\cos(\mathbf{v}_{ik}, \mathbf{v}_{jk}) = \frac{\langle \mathbf{v}_{ik}, \mathbf{v}_{jk} \rangle}{(\|\mathbf{v}_{ik}\|_2 + \epsilon)(\|\mathbf{v}_{jk}\|_2 + \epsilon)}. \tag{24}$$

For contextual comparison, we use the Gaussian kernel

$$K_\tau(\mathbf{u}, \mathbf{v}) = \exp\left(-\frac{\|\mathbf{u} - \mathbf{v}\|_2^2}{2\tau^2}\right), \qquad \tau > 0. \tag{25}$$

Here $\mathbf{u}$ and $\mathbf{v}$ are two vectors in the same embedding space, and τ is the bandwidth. In the contextual score, we use $\tau = \tau_x$ and the embeddings $\mathbf{e}^s_i$ and $\mathbf{e}^s_j$. We denote the pairwise kernels used for the utility and contextual priors by ζ_{ijk} and κ^s_{ij}:

$$\zeta_{ijk} = \exp\left(\frac{\cos(\mathbf{v}_{ik}, \mathbf{v}_{jk})}{\tau_v}\right), \qquad \kappa^s_{ij} = K_{\tau_x}(\mathbf{e}^s_i, \mathbf{e}^s_j). \tag{26}$$

Here $\|\cdot\|_2$ denotes Euclidean length and $\langle \cdot, \cdot \rangle$ denotes the inner product. The contextual kernel κ^s_{ij} is one for identical embeddings and decreases toward zero as their distance grows. The bandwidth

$\tau_x > 0$ controls how quickly it decreases: larger bandwidths give appreciable similarity to more distant cases. Its value is task-specific.

The gradient kernel ζ_{ijk} exponentiates a numerically stabilized cosine similarity. It is larger when the two gradients point in similar directions, indicating a similar pattern of local utility sensitivity. Approximately orthogonal gradients give a value near one, whereas opposing directions give a value below one. The temperature τ_v controls the contrast between directions; we use $\tau_v = 1$. The stabilizer ϵ keeps the expression defined for zero-length gradients, and the implementation clips the exponent to $[-20, 20]$. There is no path index on ζ_{ijk} because both paths use the same underlying utility gradients. Their selected reference memories can still differ.

Utility prior and contextual prior. The utility prior u^s_{ik} averages gradient similarity to the successful-case memory, while the contextual prior ψ^s_{ik} averages contextual similarity to that same memory.

$$u^s_{ik} = \frac{1}{|\mathcal{R}^s_{ik}|} \sum_{j \in \mathcal{R}^s_{ik}} \exp\left(\frac{\cos(\mathbf{v}_{ik}, \mathbf{v}_{jk})}{\tau_v}\right), \quad \psi^s_{ik} = \frac{1}{|\mathcal{R}^s_{ik}|} \sum_{j \in \mathcal{R}^s_{ik}} K_{\tau_x}(\mathbf{e}^s_i, \mathbf{e}^s_j). \tag{27}$$

Using the pairwise kernels from Eq. (26), these same averages can be written as

$$u^s_{ik} = \frac{1}{|\mathcal{R}^s_{ik}|} \sum_{j \in \mathcal{R}^s_{ik}} \zeta_{ijk}, \qquad \psi^s_{ik} = \frac{1}{|\mathcal{R}^s_{ik}|} \sum_{j \in \mathcal{R}^s_{ik}} \kappa^s_{ij}. \tag{28}$$

The denominator $|\mathcal{R}^s_{ik}|$ makes each quantity an average rather than a count of similar successes. A larger u^s_{ik} indicates closer alignment with past successful utility responses; a larger ψ^s_{ik} indicates closer contextual resemblance to past successes. Both are set to zero when the memory is empty. These priors are reliability features rather than correctness probabilities: in particular, the utility prior can exceed one. Stage 2 learns how informative they are about correctness.

Path-specific reliability features. Margin similarity is used on the utility-only path, and collective agreement is used on the mixed path. Their functional definitions from Eq. (5) are

$$\gamma_{ik} = \Gamma^U\big(p_{ik}(\cdot), k\big), \qquad \varphi_{ik} = \Gamma^M\big((\hat{y}^M_{i\ell})_{\ell=1}^K, k\big), \tag{29}$$

The functions incorporate the fitted calibration records and the case-exclusion rules described below.

Margin similarity. The function Γ^U maps agent k's utility choice distribution to margin similarity γ_{ik}: the smoothed historical correctness rate among cases with similar prediction margins. Recall that $p_{ik}(a)$ is agent k's utility-based probability for alternative a on case i. Let $p^{(1)}_{ik}$ and $p^{(2)}_{ik}$ denote the largest and second-largest values in this distribution; the superscripts indicate ranks, not fixed alternative labels. Their difference $d_{ik} = p^{(1)}_{ik} - p^{(2)}_{ik}$ is the prediction margin. A larger margin indicates greater separation between the two leading choices, but does not by itself establish correctness.

For each agent, quantiles of its calibration margins divide the margin range into at most five bins with roughly equal counts. Let B_{ik} be the margin interval containing the current value d_{ik}. Define

$$\mathcal{B}_{ik} = \{j \in \mathcal{V}^{-i}_U : d_{jk} \in B_{ik}\}, \qquad \gamma_{ik} = \Gamma^U\big(p_{ik}(\cdot), k\big) = \mu^U_k(\mathcal{B}_{ik}). \tag{30}$$

The interval B_{ik} contains margin values, whereas $\mathcal{B}_{ik}$ contains indices of calibration cases whose margins lie in that interval. The margin similarity γ_{ik} is the smoothed fraction of those cases that agent k predicted correctly on the utility path. It therefore translates the current margin into evidence based on historical outcomes. These bins use successes and failures across groups, excluding case i when needed. Duplicate quantile boundaries are removed, and a constant margin yields one bin. If the selected bin has no reference cases after exclusion, the smoothing gives $\gamma_{ik} = 1/2$. Thus Γ^U includes margin extraction, bin assignment, and the smoothed correctness calculation. Its fitted bins and calibration records are implicit; the reference set excludes the current case when applicable.

Collective agreement. The function Γ^M in Eq. (29) takes the mixed predictions of all K agents and the candidate index k, then measures agreement with the other agents. Here a peer $\ell \neq k$ is another agent in the pool. We weight its agreement by its own historical correctness:

$$\varpi_{i\ell} = \mu^M_\ell(\mathcal{V}^{-i}_M). \tag{31}$$

Thus $\varpi_{i\ell}$ is peer ℓ's smoothed correctness rate on the mixed-path calibration set, excluding the current case when necessary. It depends on past observed outcomes, not on whether the peer's current prediction is known to be correct. The candidate's collective agreement is

$$\varphi_{ik} = \Gamma^M\big((\hat{y}^M_{i\ell})_{\ell=1}^K, k\big) = \frac{\sum_{\ell \neq k} \varpi_{i\ell} \mathbb{1}[\hat{y}^M_{ik} = \hat{y}^M_{i\ell}]}{\sum_{\ell \neq k} \varpi_{i\ell}}. \tag{32}$$

The indicator is one when agents k and ℓ predict the same alternative for case i. The numerator adds the reliability weights of agreeing peers; the denominator adds the weights of all peers. Hence $\varphi_{ik} \in [0, 1]$ is the fraction of peer weight supporting the candidate's prediction. It equals one when all peers agree and zero when none agree. A historically more reliable peer contributes more weight, but agreement alone does not guarantee correctness because agents can share errors. The calibration-derived weights $\varpi_{i\ell}$ are implicit in Γ^M and use the same case-exclusion rule as above.

For a single-agent pool, we set $\varphi_{ik} = 0$. The experimental calculation uses all other mixed agents; it assumes their predictions have already been generated. Appendix A.5 explains the resulting accounting condition.

Inputs to reliability calibration. For each case and candidate, the three reliability features are collected into a column vector.

$$\mathbf{z}^U_{ik} = (u^U_{ik}, \psi^U_{ik}, \gamma_{ik})^\top, \qquad \mathbf{z}^M_{ik} = (u^M_{ik}, \psi^M_{ik}, \varphi_{ik})^\top. \tag{33}$$

Here $\top$ denotes transpose, so $\mathbf{z}^s_{ik}$ has three entries; $\mathbf{z}^U_{ik}$ and $\mathbf{z}^M_{ik}$ are its utility- and mixed-path instances, respectively. The first two entries describe similarity to past successes; the third is margin similarity for the utility-only path and collective agreement for the mixed path. These vectors are supplied to the logistic model $\mathcal{Q}^s_1$, which standardizes each feature before estimating correctness. The three reliability functions are repeated in Eq. (34). The scores themselves are not multiplied together or interpreted as independent probabilities. The local estimator $\mathcal{Q}^s_2$ uses the embeddings $\mathbf{e}^s_i$, gradients $\mathbf{v}_{ik}$, and utility familiarities h_{ik} directly, together with the group and agent indices. The group estimator $\mathcal{Q}^s_3$ uses only the group and agent indices to retrieve the corresponding calibration record.

A.3 Stage 2: Path-Conditional Agent Reliability

Stage 2 combines a logistic model, a local estimator, and a group estimator.

$$\begin{aligned} q^{s,1}_{ik} &= \mathcal{Q}^s_1(\mathbf{z}^s_{ik}, g_i, k), \\ q^{s,2}_{ik} &= \mathcal{Q}^s_2(\mathbf{e}^s_i, \mathbf{v}_{ik}, h_{ik}, g_i, k), \\ q^{s,3}_{ik} &= \mathcal{Q}^s_3(g_i, k). \end{aligned} \tag{34}$$

Their arguments describe the current case and candidate; fitted parameters, feature transformations, and stored calibration records are implicit. During calibration and cross-fitting, each function uses the appropriate fit and the reference-set exclusions described below and in Appendix A.5.

Logistic model. With component-wise standardized reliability features $\widetilde{\mathbf{z}}^s_{ik}$, the first component is

$$\begin{aligned} \eta^s_{ik} &= b^s_k + d^s_{g_i k} + (\boldsymbol{\beta}^s)^\top \widetilde{\mathbf{z}}^s_{ik}, \\ q^{s,1}_{ik} &= \mathcal{Q}^s_1(\mathbf{z}^s_{ik}, g_i, k) = \sigma(\eta^s_{ik}), \qquad \boldsymbol{\beta}^s \succeq \mathbf{0}. \end{aligned} \tag{35}$$

Here η^s_{ik} is the logistic model's score before the sigmoid, and $\boldsymbol{\beta}^s \in \mathbb{R}^3$ contains the coefficients for the three reliability features. The constraint $\boldsymbol{\beta}^s \succeq \mathbf{0}$ requires every coefficient to be nonnegative. The intercept b^s_k captures the agent's overall record, and $d^s_{g_i k}$ captures group specialization. An unseen group has zero offset. Let $\mathcal{H}_s$ be the groups represented in $\mathcal{V}_s$. For each path, we minimize

$$\begin{aligned} \mathcal{L}_s = \frac{1}{K|\mathcal{V}_s|} \sum_{i \in \mathcal{V}_s} \sum_{k=1}^K \ell(r^s_{ik}, q^{s,1}_{ik}) + \lambda_\beta \|\boldsymbol{\beta}^s\|_2^2 \\ + \frac{\lambda_b}{K} \sum_{k=1}^K (b^s_k - b^s_0)^2 + \frac{\lambda_d}{K|\mathcal{H}_s|} \sum_{g \in \mathcal{H}_s} \sum_{k=1}^K (d^s_{gk})^2, \end{aligned} \tag{36}$$

subject to the nonnegative coefficient constraint. We use $(\lambda_\beta, \lambda_b, \lambda_d) = (0.05, 0.01, 0.10)$. These are regularization strengths for the feature coefficients, agent intercepts, and group offsets, respectively; they are separate from the operating-cost parameters $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)$. The center $b_0^s = \log[p_0^s/(1-p_0^s)]$ is the logit of $p_0^s = (1+\sum_{i,k} r_{ik}^s)/(2+K|\mathcal{V}_s|)$, where the sum is over the fitting cases and all agents. The two penalties on intercepts stabilize agent estimates and shrink group effects toward zero.

Local estimator. The local reference pool contains both successes and failures:

$$\mathcal{J}_i^s = \mathcal{G}_i(\mathcal{V}_s^{-i}). \tag{37}$$

Let $\mathcal{N}_i^s \subseteq \mathcal{J}_i^s$ contain the L cases with the largest contextual similarity κ_{ij}^s, or all references if fewer than L are available. The neighborhood size L is task-specific. Within the full reference pool, normalize gradient similarity and combine the three sources of proximity:

$$\begin{aligned}\widetilde{\zeta}_{ijk}^s &= \frac{\zeta_{ijk} - \min_{j' \in \mathcal{J}_i^s} \zeta_{ij'k}}{\max_{j' \in \mathcal{J}_i^s} \zeta_{ij'k} - \min_{j' \in \mathcal{J}_i^s} \zeta_{ij'k} + \epsilon}, \\ w_{ijk}^s &= 0.4\kappa_{ij}^s + 0.3\widetilde{\zeta}_{ijk}^s + 0.3\exp\left(-\frac{|h_{ik} - h_{jk}|}{0.10}\right).\end{aligned} \tag{38}$$

Neighborhood membership depends on contextual similarity; the weights within that neighborhood additionally reflect gradient and familiarity similarity. Let $\bar{r}_k^s = |\mathcal{V}_s|^{-1} \sum_{j \in \mathcal{V}_s} r_{jk}^s$ be the agent's unsmoothed fitting accuracy. The second component is

$$\begin{aligned}q_{ik}^{s,2} &= \mathcal{Q}_2^s(\mathbf{e}_i^s, \mathbf{v}_{ik}, h_{ik}, g_i, k) \\ &= \frac{\sum_{j \in \mathcal{N}_i^s} w_{ijk}^s r_{jk}^s + \tau_0 \bar{r}_k^s}{\sum_{j \in \mathcal{N}_i^s} w_{ijk}^s + \tau_0}, \qquad \tau_0 = 2.\end{aligned} \tag{39}$$

The two pseudo-observations shrink weak neighborhood evidence toward the agent's overall record. If the reference pool is empty, we use $\bar{r}_k^s$.

Group estimator. The group estimator uses agent k's smoothed correctness rate within the current case's demographic group, using all groups when no calibration cases are available in that group:

$$q_{ik}^{s,3} = \mathcal{Q}_3^s(g_i, k) = \mu_k^s(\mathcal{G}_i(\mathcal{V}_s)). \tag{40}$$

The reference population here is the current fit, which excludes the entire held-out fold when producing cross-fitted components.

Combining the reliability estimates. The final reliability estimate is

$$q_{ik}^s = \omega_s \left[\alpha_s q_{ik}^{s,1} + (1-\alpha_s) q_{ik}^{s,2}\right] + (1-\omega_s) q_{ik}^{s,3}. \tag{41}$$

The tunable parameter α_s controls the relative weights of the feature-based logistic estimate and the local estimate. The parameter ω_s controls how much their combined estimate contributes compared with the group estimate. Equivalently,

$$q_{ik}^s = \sum_{t=1}^{3} \upsilon_{st} q_{ik}^{s,t}, \qquad \boldsymbol{\upsilon}_s = (\omega_s\alpha_s,\ \omega_s(1-\alpha_s),\ 1-\omega_s). \tag{42}$$

Here $t \in \{1, 2, 3\}$ indexes the reliability components, and υ_{st} is entry t of the weight vector $\boldsymbol{\upsilon}_s$. The weights sum to one. This makes clear that the final correctness estimate remains in $[0, 1]$ and that $\omega_s = 0$ recovers the group record.

Selecting the pooling weights. Let $\mathcal{W} = \{0, 0.1, \ldots, 1\}$. For each $(\alpha, \omega) \in \mathcal{W}^2$, form $\check{q}_{ik}^s(\alpha, \omega)$ by blending the three cross-fitted components with the weights in Eq. (42), and set $\check{k}_i^s(\alpha, \omega) = \arg\max_k \check{q}_{ik}^s(\alpha, \omega)$. The chosen weights maximize selected-agent accuracy:

$$(\alpha_s^*, \omega_s^*) = \arg \max_{(\alpha,\omega) \in \mathcal{W}^2} \frac{1}{|\mathcal{V}_s|} \sum_{i \in \mathcal{V}_s} r_{i\check{k}_i^s(\alpha,\omega)}^s. \tag{43}$$

Ties favor ω closest to $1/2$, then α closest to $1/2$; remaining ties follow grid order. We write $\check{q}_{ik}^s = \check{q}_{ik}^s(\alpha_s^*, \omega_s^*)$ for the score supplied to Stage 1. In the method and deployment equations, α_s and ω_s denote these selected values α_s^* and ω_s^*.

A.4 Agent Selection and Inference Modes

Candidate score and net selection advantage. The soft inference path preference weights each path-conditional agent reliability. The definitions in the method are

$$\pi_i(M) = \rho_i, \qquad \pi_i(U) = 1 - \rho_i, \qquad v_i(s,k) = \pi_i(s) q_{ik}^s, \qquad b_i = \max_k v_i(U,k). \tag{44}$$

The baseline b_i is the highest score among utility-only candidates. The net selection advantage subtracts that baseline and the computational cost penalty from the candidate's score.

$$\Delta_i(s,k;\boldsymbol{\lambda}) = v_i(s,k) - b_i - c_{\boldsymbol{\lambda}}(s,k) \tag{45}$$

This score supports the value-of-computation comparison in the method: a mixed candidate must improve on the best utility-only score sufficiently to offset its cost penalty. It is a selection score formed from path preference, estimated correctness, and cost; it does not represent a new calibrated probability of correctness.

Computational cost penalty. The charge has two nonnegative coefficients, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)$:

$$c_{\boldsymbol{\lambda}}(s,k) = \mathbb{1}[s = M] \left(\lambda_1 + \lambda_2 \frac{c_k}{c_0} \right), \qquad c_0 = \max_k c_k, \tag{46}$$

where c_k is a per-output API price proxy. The coefficient λ_1 penalizes using the mixed inference path, while λ_2 distinguishes agents by relative price. These coefficients convert resource preferences to the dimensionless score scale; the utility-only path has no additional calling cost.

Agent selection.

$$\hat{y}_i = \hat{y}_{ik_i^*}^{s_i^*}, \qquad (s_i^*, k_i^*) = \arg\max_{(s,k)\in\mathcal{A}} \Delta_i(s,k;\boldsymbol{\lambda}). \tag{47}$$

The common baseline does not change which candidate has the highest score. Since the best utility-only candidate has zero net selection advantage, resolving ties in favor of that path selects mixed inference exactly when $\max_k \Delta_i(M,k;\boldsymbol{\lambda}) > 0$.

Performance and Efficiency modes. With the computational cost penalty in Eq. (46), the two candidate scores, before subtracting their common reference b_i, are

$$v_i(U,k) = (1-\rho_i) q_{ik}^U, \qquad v_i(M,k) - c_{\boldsymbol{\lambda}}(M,k) = \rho_i q_{ik}^M - \lambda_1 - \lambda_2 c_k / c_0. \tag{48}$$

Performance mode uses $\boldsymbol{\lambda} = \mathbf{0}$. Efficiency-mode calibration evaluates this same rule on $\mathcal{V}$, the pooled validation set, using the initial cross-fitted reliabilities $\check{q}$ and the fitted router. The prices c_k are per-output proxies, since cached predictions lack provider-side token telemetry.

Quality and resource summaries. For a candidate operating point $\boldsymbol{\lambda}$, let $\hat{y}_i(\boldsymbol{\lambda})$ and $(s_i^*(\boldsymbol{\lambda}), k_i^*(\boldsymbol{\lambda}))$ be the resulting validation prediction and selection. Let $A(\boldsymbol{\lambda})$ be accuracy and $F(\boldsymbol{\lambda})$ be class-frequency-weighted F1, as defined in Appendix C. The calibration quality and its target are

$$Q(\boldsymbol{\lambda}) = \tfrac{1}{2}[A(\boldsymbol{\lambda}) + F(\boldsymbol{\lambda})], \qquad Q_0 = \max_{(s,k)\in\mathcal{A}} Q_{sk} - \delta. \tag{49}$$

Here Q_{sk} is the quality of always selecting candidate (s,k) on the same validation set, and $\delta \geq 0$ is the allowed shortfall. Utility-only path use and mean selected calling cost are

$$U(\boldsymbol{\lambda}) = \frac{1}{|\mathcal{V}|} \sum_{i\in\mathcal{V}} \mathbb{1}[s_i^*(\boldsymbol{\lambda}) = U], \qquad C(\boldsymbol{\lambda}) = \frac{1}{|\mathcal{V}|} \sum_{i\in\mathcal{V}} \mathbb{1}[s_i^*(\boldsymbol{\lambda}) = M]\, c_{k_i^*(\boldsymbol{\lambda})}. \tag{50}$$

All quantities in this subsection are validation quantities; no test labels enter the search.

Quality-constrained selection. Let Ξ be the finite grid of operating points and $\Xi_0 = \{\boldsymbol{\lambda} \in \Xi : Q(\boldsymbol{\lambda}) \geq Q_0\}$. When the outer-fold router diagnostic supports the target Q_0, select

$$\boldsymbol{\lambda}^* \in \arg\max_{\boldsymbol{\lambda}\in\Xi_0} U(\boldsymbol{\lambda}). \tag{51}$$

Among maximizers, prefer lower C, then higher Q. If Ξ_0 is empty, maximize Q over Ξ, breaking ties by higher U and then lower C. Thus the quality requirement and the preference for fewer mixed calls have distinct roles.

Balanced frontier selection. If the outer-fold diagnostic falls below Q_0, retain the nondominated operating points $\mathcal{P} \subseteq \Xi$: a point is removed only if another has at least as large Q and U, with a strict improvement in one. Normalize each objective over this frontier:

$$\widetilde{Q}(\boldsymbol{\lambda}) = \frac{Q(\boldsymbol{\lambda}) - \min_{\xi \in \mathcal{P}} Q(\xi)}{\max_{\xi \in \mathcal{P}} Q(\xi) - \min_{\xi \in \mathcal{P}} Q(\xi)}, \qquad \widetilde{U}(\boldsymbol{\lambda}) = \frac{U(\boldsymbol{\lambda}) - \min_{\xi \in \mathcal{P}} U(\xi)}{\max_{\xi \in \mathcal{P}} U(\xi) - \min_{\xi \in \mathcal{P}} U(\xi)}. \tag{52}$$

A normalized objective is set to one if its range is at most ϵ. Select the point with the largest harmonic mean:

$$\boldsymbol{\lambda}^* \in \arg\max_{\boldsymbol{\lambda} \in \mathcal{P}} \frac{2\widetilde{Q}(\boldsymbol{\lambda})\widetilde{U}(\boldsymbol{\lambda})}{\widetilde{Q}(\boldsymbol{\lambda}) + \widetilde{U}(\boldsymbol{\lambda})}. \tag{53}$$

The ratio is zero when its denominator is at most ϵ. Ties favor lower C, higher Q, then higher U, in that order. This criterion favors progress on both objectives; it does not impose the quality floor in Eq. (51).

Both regimes search $\lambda_1 \in [0, 0.75]$ on an equally spaced grid: 31 points for quality-constrained selection and 151 for frontier selection. They use $\lambda_2 \in \{0, 0.01, 0.025, 0.05, 0.10\}$. The implementation uses a tolerance of 10^{-12} for feasibility and dominance comparisons and rounds ranking quantities to 12 decimal places.

A.5 Calibration, Model Fitting, and Parameter Tuning

Data roles. Base utility models, textual contexts, and mixed predictions are obtained from each task's training split and are group-specific. The orchestration models are calibrated on pooled validation cases, retaining group–agent effects. The disjoint test split is used only for final evaluation.

Cross-fit the reliability components. Within each path's fitting population, partition individual cases into five folds and repeat this procedure with three independent shuffles. Groups are not held out as units. Each fold fit rebuilds the feature scaling, PCA, successful-case memories, margin bins, group statistics, and logistic coefficients using only the other folds.

Let $F_r(i)$ be the fold containing i in repeat r. Here a fold is a set of case indices, and $r \in \{1, \dots, R\}$ indexes the independent shuffles. Write $\mathbf{q}_{ik}^s = (q_{ik}^{s,1}, q_{ik}^{s,2}, q_{ik}^{s,3})^\top$ and use a superscript $(-F)$ for an estimator fitted without fold F. The cross-fitted components are

$$\check{\mathbf{q}}_{ik}^s = \frac{1}{R} \sum_{r=1}^{R} \mathbf{q}_{ik}^{s,(-F_r(i))}, \qquad R = 3. \tag{54}$$

First average the components across repeats, then select their pooling weights by Eq. (43). The resulting within-path predictions define the decisive cases for Stage 1.

Fit the router and operating point. Fit the soft router by Eq. (19). In efficiency mode, an additional five-fold router diagnostic holds out router-fitting cases while reusing the previously constructed Stage 2 scores. If its uncharged joint quality reaches Q_0, use the quality-constrained search; otherwise use frontier selection. The operating-point search itself uses the router fitted on all calibration cases.

The fold exclusion above protects each raw Stage 2 component from its own case's label. Pooling weights and operating points are subsequently selected using validation labels, however, and the router diagnostic reuses the Stage 2 calibration. These are calibration quantities, not an independent, fully nested evaluation of the complete tuned system. Final performance is measured on the untouched test split.

Fit the deployment selectors. In performance mode, assign calibration case i to the mixed subset when $\rho_i \geq 1/2$ and to the utility subset otherwise. In the quality-constrained efficiency regime, use the penalized joint selections instead. Fit each path's selector on its assigned subset if it contains at least $\max(20, 5K)$ cases; otherwise use the pooled validation set. Each such fit repeats the Stage 2 cross-fitting procedure to select its pooling weights. In the frontier regime, retain the initial pooled selector so that conditional refitting does not change the reliability scale used to select the operating point. These refits retain the same correctness targets r_{ik}^s.

Availability and accounting. Collective agreement uses predictions from the other mixed agents. The cached evaluation therefore measures selection among available outputs. The summaries $C(\boldsymbol{\lambda})$ and $U(\boldsymbol{\lambda})$ describe the selected output and selected path. Generating all peer outputs on demand would incur their costs as well, even if the calls were concurrent. A deployment seeking to avoid those calls would need reusable predictions or a separately specified acquisition or shortlisting policy.

B Dataset details

VAC models vaccination decisions based on demographic attributes and belief factors, such as trust in science; SWM examines transportation mode selection under varying travel time and cost conditions; JOB captures work arrangement choices during the COVID-19 pandemic, conditioned on occupational characteristics and commuting constraints. For the other three tasks, they share a common set of demographic variables (*e.g.* age, education, and income), they each incorporate task-specific habit and health data, such as sleep issues or previous attempts to quit. The output categories of ALC, MAJ and PR are: *Past 30 Days*, *Past 12 Month* and *More than 12 months*. Although the raw survey data contains 2,636 features, we empirically select the top 15–25 features per task to prevent overfitting and maintain interpretability.

C Evaluation Metrics

We report multiclass accuracy and class-frequency-weighted precision and F1. Let n_{ab} count cases with true class a and predicted class b, let $n_a = \sum_b n_{ab}$, and let $n = \sum_a n_a$. Accuracy and the per-class precision, recall, and F1 are

$$A = \frac{\sum_a n_{aa}}{n}, \qquad P_a = \frac{n_{aa}}{\sum_b n_{ba}}, \qquad R_a = \frac{n_{aa}}{n_a},$$
$$F_a = \frac{2P_aR_a}{P_a + R_a}. \tag{55}$$

A ratio with a zero denominator is assigned zero. Weighting by observed class frequency gives

$$P = \sum_{a=1}^{m} \frac{n_a}{n} P_a, \qquad F = \sum_{a=1}^{m} \frac{n_a}{n} F_a. \tag{56}$$

The calibration quality $Q = (A + F)/2$ uses this weighted F1. In particular, we average per-class F1 scores, rather than taking the harmonic mean of averaged precision and recall.

D Baseline Adaptations

We adapt FrugalGPT, RouteLLM, and HybridLLM to a pool of four utility-only and four utility+LLM (mixed) experts. The routers use raw individual and decision attributes x_i, with supervision from each expert's final-answer correctness on validation data. Router fitting and threshold selection use validation data; test data are reserved for evaluation.

D.1 FrugalGPT

FrugalGPT uses an answer-conditioned scorer to decide whether to accept a model's response or continue to another model (Chen et al., 2024). We adapt this principle to a fixed utility-to-mixed cascade. The utility expert e_w and mixed expert e_s are selected by validation accuracy within their respective pools. After obtaining the utility prediction $\hat{y}_{i,e_w}$, a class-balanced logistic model estimates its correctness:

$$q_i = P(\hat{y}_{i,e_w} = y_i \mid x_i, \mathrm{onehot}(\hat{y}_{i,e_w})).$$

The cascade accepts the utility prediction when $q_i \geq \tau$ and invokes the mixed expert otherwise. The acceptance threshold is selected from out-of-fold validation scores under the cost budget. This preserves the answer-dependent stopping rule while using structured attributes and a discrete decision as the scorer's inputs.

D.2 RouteLLM

RouteLLM learns to choose between weak and strong models before generating an answer (Ong et al., 2024). We use the same validation-selected utility and mixed experts as above, replacing response-preference supervision with the indicator that the mixed expert is correct while the utility expert is wrong:

$$r_i = \mathbb{I}[\hat{y}_{i,e_s} = y_i \ \wedge \ \hat{y}_{i,e_w} \neq y_i].$$

A class-balanced logistic router estimates $P(r_i = 1 \mid x_i)$ from raw attributes. It selects the mixed expert when this probability exceeds a threshold chosen on out-of-fold validation predictions under the cost budget, and selects the utility expert otherwise. Candidate answers are used to construct training targets but are unavailable to the router at inference.

D.3 HybridLLM

We adapt HybridLLM's query-based model selection (Ding et al., 2024) to all eight experts using two variants. Both route from raw attributes before invoking the selected expert.

N-class routing. A class-balanced multinomial logistic model predicts an expert directly. Its validation target is the first correct expert in an ordering that prioritizes utility-only experts, then validation accuracy. If no expert is correct, the first expert in that ordering supplies the target. This encodes a preference for utility-only decisions when they suffice.

N-label routing. A separate binary logistic model estimates each expert's probability of correctness. The router selects the cheapest expert whose predicted correctness is at least 0.5, using validation accuracy to resolve ties. If none reaches that threshold, it selects the expert with the highest predicted correctness. This variant retains separate quality estimates for all candidates and uses them for cost-aware selection.

E Large-scale Personalized Decision Making Deployment

In this study, we characterize the system-level deployment cost of personalized decision modeling at population scale. Specifically, we estimate the total number of model inferences required to support personalized behavioral decisions over a one-month horizon at the city-, county-, and state-level across the United States..

Population Data. Population statistics for representative geographic regions were obtained from the 2024 U.S. Census Bureau[1]. At the city level, we selected Atlanta (AT), Seattle (SE), and San Jose (SJ). AT has 404,600 adults (age $\geq$ 18), SE has 607,600, and SJ has 776,000. At the county level, Harris County (H-C) has 3,897,200 adults, Cook County (C-C) has 4,032,100, and Los Angeles County (LA-C) has 7,591,100. At the state level, Florida (FL) has 18,183,600 adults, Texas (TX) has 24,344,300, while California (CA) has 30,677,500.

Table 3: **The number of population aged $\geq$ 18 at representative region.**

Level	Name	Population
City	Atlanta	404,600
	Seattle	607,600
	San Jose	776,000
County	Harris	3,897,200
	Cook	4,032,100
	Los Angeles	7,591,100
State	Florida	18,183,600
	Texas	24,344,300
	California	30,677,500

[1] https://data.census.gov

Decision Data. For SWM dataset, Bureau of Transportation Statistics[2] reports that U.S. citizens make approximately 2.28 mobility-related decisions per day (e.g., trip routing or mode choice). According to the survey made by Substance Abuse and Mental Health Services Administration (SAMHSA), roughly 16.3% of adults consumed marijuana and 51.1% consumed alcohol in the past month. The National Health and Nutrition Examination Survey[3] shows that 10.7% used pain relievers last month. For these sub-populations, we assume one daily binary decision regarding their respective consumption behaviors, while an additional baseline utility category captures daily routine decision-making. Furthermore, we assume that each adult makes job- or vaccine-relative decision per month to account for low-frequency yet impactful decision types. Aggregating these quantities over a 30-day month yields the following per-adult monthly decision load D_d.

Experiment Settings. We report two deployment-oriented quantities for each region: (i) the monthly monetary inference cost and (ii) the monthly total inference time required to serve all personalized decisions.

Cost modeling. Let $C_{d,k}$ denote the per-decision inference cost of model k on dataset d (Tab. 1), and let D_d denote the per-adult monthly number of decisions associated with dataset d. Based on the decision statistics, we set $D_1 = 1$, $D_2 = 30 \times 2.28$, $D_3 = 1$, $D_4 = 30 \times 0.163$, $D_5 = 30 \times 0.107$, and $D_6 = 30 \times 0.511$, corresponding to the substance-use datasets. The monthly per-person cost of model k is then $\sum_{d=1}^{6} D_d\, C_{d,k}$, and for a region with adult population P, the total cost becomes

$$\mathrm{Cost}^{(k)}_{\mathrm{total}}(P) = P \sum_{d=1}^{6} D_d\, C_{d,k} = P\Big(C_{1,k} + 30 \times 2.28\, C_{2,k} + C_{3,k} + 30 \times 0.163\, C_{4,k} + 30 \times 0.107\, C_{5,k} + 30 \times 0.511\, C_{6,k}\Big). \tag{57}$$

Inference-time modeling. To obtain a comparable estimate of wall-clock deployment time, we assume that all LLM-based experts (4o-mini-DM, 5-mini-DM, G-2F-DM, DS-3.2-DM) are constrained by a shared token-generation rate of $r = 5000$ tokens/second. Let TN_k denote the average token usage per decision of expert k, and recall that each adult generates D decisions per month (i.e., $D = \sum_d D_d$). In the baseline setting, decision requests are uniformly split across the four experts and executed in parallel; the total inference time is therefore determined by the slowest expert:

$$T_{\mathrm{base}}(P) = \max_{k \in \{1,\dots,4\}} \left(\frac{P\, D\, \mathrm{TN}_k}{4\, r} \right). \tag{58}$$

For our method, inference time additionally includes the overhead of the adaptive router. Let $\tau_{\mathrm{route}}(P)$ denote the effective routing time per decision, and let ρ_k denote the utilization of expert k under our method. The total inference time is then

$$T_{\mathrm{ours}}(P) = P\, D\, \tau_{\mathrm{route}}(P) \; + \; \max_{k \in \{1,\dots,4\}} \left(\frac{P\, D\, \rho_k\, \mathrm{TN}_k}{r} \right). \tag{59}$$

In the simulation, we instantiate D using the same decision statistics as in the cost modeling, take TN_k from Tab. 1, and parameterize $\tau_{\mathrm{route}}(P)$ and ρ_k based on the empirical utilization and profiling results reported in the main text. The populations P correspond to the city-, county-, and state-level regions in Tab. 3. The final results are illustrated in Fig. 5

[2] https://www.fhwa.dot.gov/policyinformation/2022Summary_of_Travel_Trends_Report.pdf

[3] https://www.cdc.gov/nchs/products/databriefs/db369.htm