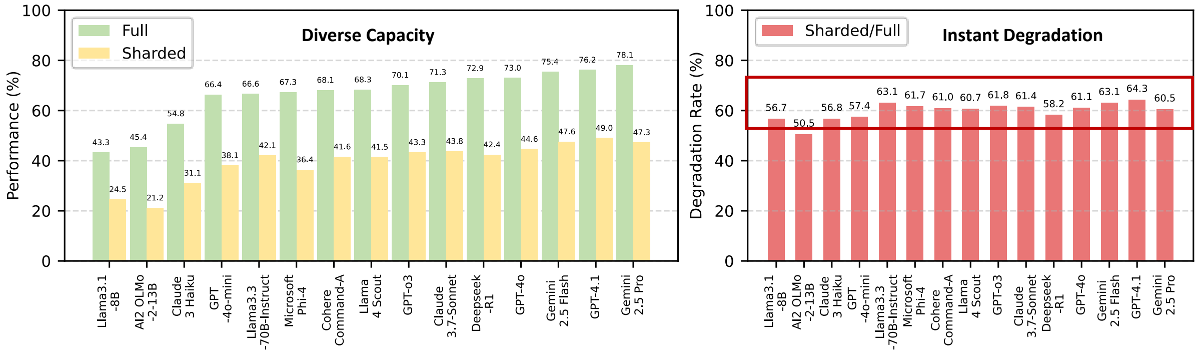

Figure 2: **Performance comparison across different LLMs on the LiC benchmark (Laban et al., 2025).** While absolute performance improves with model scale, the relative performance degradation remains strikingly constant (∼60%). This structural invariance suggests that the bottleneck lies in the alignment prior rather than model capacity.

cific individual. This manifests as generic convergence: the model assumes the most statistically probable intent, leading to the LiC phenomenon where the model confidently answers the wrong question. Scaling the model size (θ) does not solve this; it merely allows the model to better fit the prior distribution of the average user, essentially reinforcing the misalignment. When contextual uncertainty is high, a frozen LLM defaults to its pretraining prior, generating I_t that maximizes (approximately) $P_\theta(I_t \mid C_t)$. Since this distribution encodes population-level patterns rather than individual preferences, the output often reflects a stereotypical "average user," leading to confidently incorrect responses (LiC). Scaling model size only sharpens this prior, exacerbating the misalignment in the absence of personalized signals.

This theoretical limitation is corroborated by empirical evidence. As illustrated in Figure 2, a re-analysis of experimental results from Laban et al. (2025) reveals a striking pattern: while stronger models achieve higher absolute scores, the *relative performance degradation* between fully specified and underspecified settings remains remarkably constant (approximately 60%) across diverse model sizes and families. We attribute this **invariant degradation** to the homogeneity of the alignment prior. Although models differ in capacity θ, they are predominantly pre-trained and aligned on similar vast corpora, leading them to converge towards a shared representation of the "average user" (P_{avg}). In the absence of specific constraints, all models fall back to this shared prior:

$$\arg\max_{I_t} P_\theta(I_t \mid C_t) \approx \arg\max_{I_t} P_{\text{avg}}(I_t \mid C_t). \tag{4}$$

Since the specific user intents in the benchmark deviate from this population mean in a fixed manner, the divergence between the actual user intent and the average prior acts as a constant structural penalty. Consequently, merely scaling model parameters optimizes the fit to P_{avg} but does not bridge the semantic gap to the specific individual, rendering the ambiguity strictly unresolvable via scaling alone.

3.3 Reducing Entropy via History

The decomposition in Eq. (3) implies that to improve performance, we must sharpen the distribution $P(I_t \mid \dots)$. Since the entropy $H(I_t \mid C_t)$ is constrained by the information loss in C_t, we must introduce an auxiliary variable: the user's generalized interaction history, denoted as $\mathcal{H}$.

We propose that while C_t is ambiguous on its own, the extended context $(C_t, \mathcal{H})$ contains sufficient information to recover I_t. By encapsulating longitudinal evidence of user behavior, including past dialogue trajectories and interaction outcomes, $\mathcal{H}$ implicitly encodes the user's specific expression habits T and effectively acts as the "compression key".

Our method introduces a Mediator M to approximate this inference process. Instead of asking the general model to guess I_t solely from C_t, the Mediator computes:

$$\hat{U} \sim P(U \mid C_t, \mathcal{H}). \tag{5}$$

Here, $\hat{U}$ is a reconstructed, fully-specified instruction that acts as a proxy for the latent intent I_t. By conditioning on $\mathcal{H}$, the Mediator significantly reduces the conditional entropy:

$$H(I_t \mid C_t, \mathcal{H}) \ll H(I_t \mid C_t). \tag{6}$$

The downstream LLM then operates on this low-entropy input: $R \sim P_\theta(R \mid \hat{U})$. This effectively bypasses the information bottleneck, allowing the

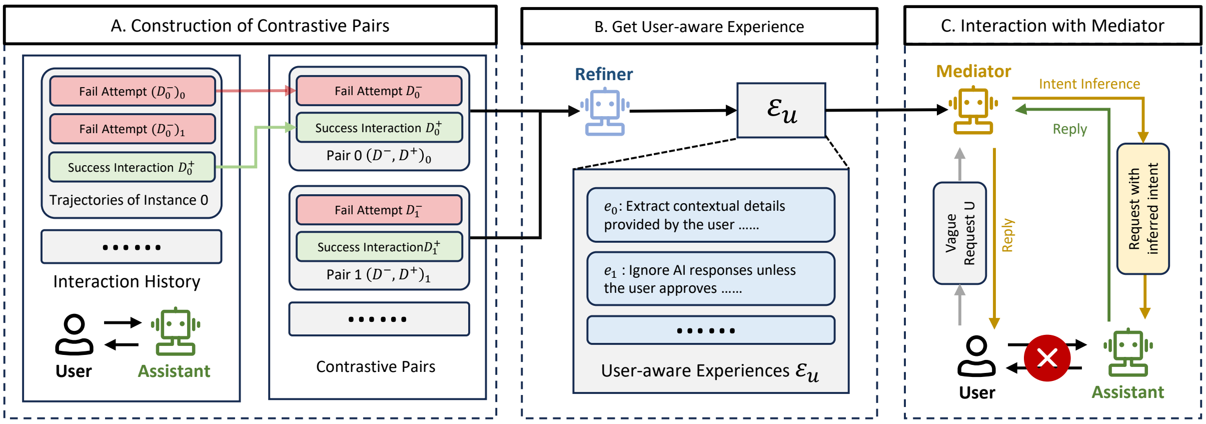
A. Construction of Contrastive Pairs
Fail Attempt $(D_0^-)_0$
Fail Attempt $(D_0^-)_1$
Success Interaction D_0^+
Trajectories of Instance 0
••••••
Interaction History
User
Assistant
Fail Attempt D_0^-
Success Interaction D_0^+
Pair 0 $(D^-, D^+)_0$
Fail Attempt D_1^-
Success InteractionD_1^+
Pair 1 $(D^-, D^+)_1$
••••••
Contrastive Pairs
B. Get User-aware Experience
Refiner
$\mathcal{E}_u$
e_0: Extract contextual details provided by the user ……
e_1 : Ignore AI responses unless the user approves ……
••••••
User-aware Experiences $\mathcal{E}_u$
C. Interaction with Mediator
Mediator
Intent Inference
Reply
Vague Request U
Reply
Request with inferred intent
User
Assistant

Figure 3: **Pipeline of the Mediator Framework.** We construct contrastive pairs by extracting a failed conversational trajectory D^- and the corresponding successful trajectory D^+ for the same task instance from the user's historical logs. The Refiner distills these pairs into explicit pragmatic experiences $\mathcal{E}$, which guide the Mediator to explicate ambiguous user contexts into precise instructions for the Assistant. The Mediator operates as a transparent alignment layer, decoupling the user from the raw execution model while maintaining a seamless interaction flow.

general model to execute tasks with the clarity and precision of a single-turn interaction.

4 Method

Building upon the analysis in §3, we propose a Mediator-Assistant framework designed to resolve the pragmatic mismatch in multi-turn dialogues. To bridge the information gap, we leverage the user's generalized history $\mathcal{H}$. In our implementation, $\mathcal{H}$ consists of raw contrastive interaction pairs (successful vs. failed trajectories). However, raw history could be noisy and high-dimensional. Therefore, we introduce a Refiner module (R) to distill $\mathcal{H}$ into a compact set of explicit Experiences ($\mathcal{E}$). These experiences serve as the "pragmatic profile" for the Mediator M. Our approach adopts a training-free, experience-driven paradigm. This strategic choice ensures immediate adaptability: the system learns from history without parameter updates, bypassing the storage and versioning overheads of per-user fine-tuning.

4.1 The Mediator Framework

The core premise, as derived in Eq. (3), is that the raw context C_t acts as a lossy compression of the true intent I_t. Direct interaction fails because the generic Assistant (A) maximizes $P(R \mid C_t)$ based on population priors rather than individual intent. The Mediator acts as an alignment layer, approximating the inference distribution by conditioning on the distilled experiences: $P(I_t \mid C_t, \mathcal{E})$.

The Mediator takes the accumulated context C_t and experiences $\mathcal{E}$. It analyzes the discrepancy between the surface utterance and the latent intent, generating a reconstructed instruction $\hat{U}$. This $\hat{U}$ is an explicit, fully specified articulation of I_t. The Assistant A then generates the response based on this low-entropy input:

$$R = A(\hat{U}), \quad \text{where } \hat{U} \approx I_t. \tag{7}$$

For a specific user, the distilled experiences are aggregated into a fixed knowledge base $\mathcal{E}_u$, which will be injected into the Mediator as a system instruction. Mathematically, the Mediator performs the mapping:

$$\hat{U} = M(C_t \mid \mathcal{E}_u). \tag{8}$$

During inference, the Mediator references this established profile to interpret the current context C_t. Since $\mathcal{E}_u$ explicitly encodes the user's specific pragmatic habits and constraints, M can accurately detect omitted information in the surface utterance and synthesize the clarified prompt $\hat{U}$.

4.2 Experience Acquisition

This subsection details how we construct the raw history $\mathcal{H}$ and how the Refiner transforms it into actionable experiences $\mathcal{E}$.

Contrastive Pair Construction. We premise our data construction on a pervasive behavioral pattern in human-computer interaction: iterative query reformulation (Huang and Efthimiadis, 2009; Odijk et al., 2015). Real-world users typically exhibit goal-directed persistence; when an initial ambiguous utterance fails to elicit the desired response, users rarely abandon the task immediately. Instead, they engage in a trial-and-error process, refining their instructions until the model successfully executes the task. This behavior naturally yields a dataset of paired trajectories within successful sessions. We view the final, successful turn as the ground-truth articulation (D^+) of the user's intent, while the preceding failed interaction sequence serves as the ambiguous context (D^-). We define the generalized history $\mathcal{H}$ as a collection of contrastive interaction pairs (D^-, D^+). From an information-theoretic perspective, the difference between D^- and D^+ explicitly reveals the latent information bits that were omitted in the ambiguous context but are required to reduce the conditional entropy $H(I_t \mid C_t)$.

To operationalize this framework within our simulation benchmark, we construct these pairs synthetically using interaction logs. We identify instances where the model exhibits performance discrepancies: specifically, instances where the model fails in the multi-turn conversational setting but succeeds when the same task is presented in a single-turn format. From these instances, we construct discrete contrastive pairs. For each task selected for experience mining, we sample a single specific failed conversational trajectory to serve as D^- and pair it directly with the corresponding successful single-turn input as D^+. We do not aggregate all possible failure modes for a task; instead, we establish a one-to-one mapping between a specific noisy history and an effective input. In this context, the successful single-turn prompt acts as a proxy for the user's "final successful turn." By analyzing this specific pair, the system can learn to bridge the gap between particular context and effective intent.

Experience Distillation via Refiner. To extract explicit inference rules from these noisy interaction logs, we employ a Refiner (R). The Refiner performs an inductive analysis on the contrastive pairs. It receives contrastive pairs (D^-, D^+) and performs a contrastive analysis: what underlying pattern can be learned to identify user's true intents from the trajectories in the future? The output is a set of structured textual guidelines $\mathcal{E} = \{e_0, e_1, ..., e_n\}$. These guidelines do not merely memorize the specific task content, but distill the pragmatic strategy (e.g., "If the user has not explicitly approved the previous solution, he is not satisfied with it."). These distilled experiences serve as the context for the Mediator.

5 Experiments

To systematically evaluate the robustness of LLMs in multi-turn interactions and validate our proposed framework, we conduct extensive experiments leveraging the simulation benchmark introduced by Laban et al. (2025). We benchmark a diverse set of state-of-the-art models, including the widely-used GPT-4o-mini (Hurst et al., 2024), the highly capable GPT-5.2, and the reasoning-specialized DeepSeek-V3.2-Thinking (DeepSeek-AI, 2025). Details of experiment setup are available at §A. In this section, we present our main results, demonstrating the universality of the LiC phenomenon and the efficacy of our Mediator across different model architectures (§5.1). We also provide an in-depth ablation analysis to distinguish pragmatic alignment from simple factual memory retrieval (§5.2).

5.1 Main Results

Table 1 presents the comprehensive evaluation results across four domains: Code, Database, Actions, and Math. We compare three settings: (1) **Full**: The idealized upper bound, where users provide complete, unambiguous instructions in a single turn. (2) **Sharded**: The baseline setting, representing simulated multi-turn conversations where context is fragmented and ambiguous. (3) **w/ Ours**: The proposed framework, where the Experience-Driven Mediator reconstructs the *Sharded* input into a specified instruction before passing it to the Assistant. To rigorously assess model robustness against the stochasticity inherent in multi-turn generation, we conducted five independent runs for every experimental instance. We report two key metrics: (1) Average Performance ($\bar{P}$): The mean score across the five runs. (2) Reliability (R): A consistency metric calculated at the instance level. For each specific instance, we measure the divergence between its best and worst outcomes across the five runs ($S_{\text{max}} - S_{\text{min}}$). The reported R is the average of $1 - (S_{\text{max}} - S_{\text{min}})$ across all instances.

The Persistence of LiC. Comparing the *Full* and *Sharded* settings reveals a severe alignment gap that transcends model architectures. First, standard instruction-tuned models exhibit dramatic degradation; for instance, **GPT-5.2** drops from a near-perfect 92.7% to 48.5%. This confirms that scaling model capabilities alone cannot resolve the ambiguity problem. In the absence of explicit intent, even the most capable models inevitably re-