

Do LLMs Recognize Your Latent Preferences? A Benchmark for Latent Information Discovery in Personalized Interaction

Ioannis Tsaknakis¹, Bingqing Song¹, Shuyu Gan², Jiangweizhi Peng³, Dongyeop Kang², Alfredo Garcia⁴, Gaowen Liu⁵, Charles Fleming⁵, Mingyi Hong¹

¹Department of Electrical and Computer Engineering, University of Minnesota

²Department of Computer Science and Engineering, University of Minnesota

³Department of Industrial and Systems Engineering, University of Minnesota

⁴Department of Industrial and Systems Engineering, Texas A&M University

⁵Cisco Research, Cisco

{tsakn001, gan00067, mhong}@umn.edu

Abstract

Large Language Models (LLMs) excel at producing broadly relevant text, but this generality becomes a limitation when user-specific preferences are required, such as recommending restaurants or planning travel. In these scenarios, users rarely articulate every preference explicitly; instead, much of what they care about remains *latent*, waiting to be inferred. This raises a fundamental question: *Can LLMs uncover and reason about such latent information through conversation?*

We address this problem by introducing a benchmark for evaluating *latent information discovery*—the ability of LLMs to reveal and utilize hidden user attributes through multi-turn interaction. The benchmark spans three progressively realistic settings: the 20 Questions game, Personalized Question Answering, and Personalized Text Summarization. All tasks share a tri-agent framework (User–Assistant–Judge) enabling turn-level evaluation of elicitation. Our results reveal that while LLMs can indeed surface latent information through dialogue, their success varies dramatically with context: from 10% to 100%, depending on task complexity, topic, model, and number of hidden attributes. This benchmark provides the first systematic framework for studying *latent information discovery* in personalized interaction, highlighting that effective preference inference remains an open frontier for building adaptive AI systems.

1 Introduction

Large Language Models (LLMs) have achieved remarkable success across domains, such as healthcare (Wang et al., 2025; Busch et al., 2025), education (Chu et al., 2025; Xiao et al., 2023), translation (Wang et al., 2024; Zhu et al., 2024), and code generation (Jiang et al., 2024; Zan et al., 2023). Modern LLMs can now produce responses that are accurate, coherent, and contextually appropriate. Yet these strengths do not guarantee *user satisfaction*: many LLM responses remain overly *generic*, designed to appeal to a broad audience rather than reflect an individual’s intent or constraints.

As LLMs increasingly mediate personal and professional decision-making (e.g., recommending restaurants, summarizing news, or offering tailored advice), this lack of *personalization* becomes a critical limitation. In practice, user-specific information is often *latent*—unstated, contextual, or even subconscious—and therefore unavailable to the model at the outset.

A simple motivating example. Consider the classic game of *20 Questions*. One player thinks of an object, and the other must infer it by asking a series of yes-or-no questions. Success requires *strategic questioning* to reveal hidden information. Now replace the hidden object with a user’s unstated preference, like a dietary restriction or stylistic goal. Can an LLM play this game, asking the right questions to infer what the user values and tailoring its



Figure 1: Illustration of a question answering task. Strategy A: The LLM directly provides a response. This response is generic and violates the user’s preferences. Strategy B: The LLM tries to elicit the user preferences through a multi-turn conversation, requesting specific information at each turn. At the end of the interaction, the LLM produces a personalized response.

final answer accordingly? This analogy illustrates the core challenge of this work: *effective personalization requires latent information discovery*.

How can LLMs personalize? We can imagine three possible strategies: (1) User explicitly provides all preferences in the prompt; (2) LLM requests the preferences before responding; (3) LLM actively *elicits* them through *multi-turn conversation* — asking targeted questions, and interpreting answers, and adapting its final response. While the first two strategies are technically viable, they are rarely realistic: users seldom recall or articulate every relevant preference upfront. The third approach, *interactive elicitation*, is therefore the most natural and potentially powerful path toward genuine personalization. Fig. 1 illustrates a question-answering task where the contrast between providing a direct, generic answer and providing a personalized answer through multi-turn interaction with the user is highlighted.

We note that studying personalization as a latent information discovery problem is of practical importance. It captures the challenging scenario in which supplementary signals, such as a user’s profile, history, or past interactions, are unavailable, and the system must rely solely on information elicited through a multi-turn interaction. This situation is common in practice: LLMs are typically not connected by default to external online services that may contain information about users’ shopping behavior, travel plans, or habits, nor do they have access to users’ complete profiles and histories. They may, however, have access to a user’s past conversations and some user-provided context, although, as discussed above, even this assumption is not always realistic.

Our benchmark. Building on this intuition, we design a unified benchmark for evaluating LLMs’ ability to discover and use latent user information through multi-turn dialogue. The benchmark follows a consistent tri-agent framework, comprising a User (with hidden preferences), an Assistant (the model under evaluation), and a Judge (which evaluates alignment), and spans three increasingly realistic tasks: (1) *20 Questions*: a controlled reasoning setting isolating pure latent information discovery; (2) *Personalized Question Answering*: unveiling latent user preferences and utilizing them for providing personalized responses to user questions; and (3) *Personalized Text Summarization*: unveiling latent user preferences and utilizing them for providing a document summarization compatible with those preferences. This design enables systematic, turn-level evaluation of questioning strategies, reasoning efficiency, and personalization accuracy. We focus on the passive-user

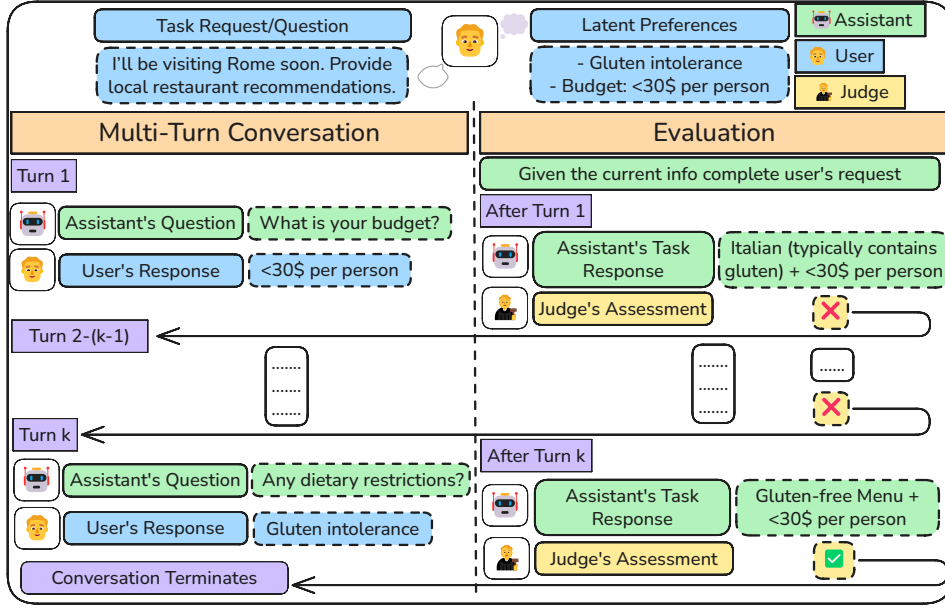


Figure 2: The benchmark’s general setting, including the example of Fig. 1. The boxes with a solid border depict the generic elements of the setting, while the boxes with a dashed border describe the example. Both the multi-turn conversation between the Assistant and the User, and the evaluation phase (where the Judge is utilized) after each turn, are depicted.

setting, where the user responds only when prompted, mirroring real-world (and in some sense worst case) interactions where users rarely volunteer all relevant context.

Benchmarking open- and closed-source models on our dataset reveals performance varying from 10% to 100%, depending on task complexity, topic, model, and number of hidden attributes. Overall, we can say that latent preference identification is neither inherently easy nor inherently difficult; instead, its level of difficulty depends on the specific setting and other factors such as the number of latent preferences. Together, these results establish the first systematic foundation for studying *latent information discovery*, a core yet underexplored skill required for building truly adaptive, user-centered LLMs.

Finally, it is worth noting that the benchmark tasks can be interpreted through the lens of preference tree models in decision theory (Tversky & Sattath, 1979), where each user’s latent preference structure defines a hierarchical set of aspects. Through dialogue, the LLM effectively performs an elimination over this latent tree, by asking targeted questions to prune inconsistent branches until the user’s intended preference leaf is revealed.

2 Related Works

Multi-Turn Interaction Benchmarks. Several studies have evaluated LLM performance in multi-turn conversations, focusing on general dialogue capabilities rather than personalization. For instance, Kwan et al. (2024) assess models’ abilities in *recollection* (retaining information from earlier turns) and *refinement* (revising previous instructions), while Bai et al. (2024) examine *perceptivity*, *adaptability*, and *interactivity* across multi-turn dialogue scenarios. Similarly, Zhang et al. (2024) employ the classic *20 Questions* game as a surrogate task to analyze LLMs’ reasoning and planning skills over multiple turns. While these studies shed light on the dynamics of conversational reasoning, they do not address how models can *actively elicit and integrate hidden user information*, which is the focus of our work. In contrast, our benchmark frames multi-turn dialogue as a means for **latent information discovery**, explicitly testing whether LLMs can uncover unspoken user attributes.

LLM Personalization, Preference Following and Elicitation. A growing body of work investigates how LLMs personalize, i.e., adapt to individual users’ preferences and needs.

This can involve several components, and some key ones are preference following and elicitation.

Several existing benchmarks evaluate how models *follow* or *apply* known preferences. For example, Zhao et al. (2025a) assess preference following and memorization abilities in long-context conversations, while Afzoon et al. (2024) evaluate both open- and closed-source LLMs on generating responses consistent with a given persona and conversational context. The LaMP (Salemi et al., 2024) and LongLaMP (Kumar et al., 2024) benchmarks extend this idea by evaluating model outputs across eleven personalization tasks, such as product rating and email completion. More recently, Zhao et al. (2025b) also study personalization through multi-turn interactions, focusing on explicit persona conditioning and interactive adaptation. Our work differs from all of the above in both goal and setting. We focus on *preference unveiling/elicitation* rather than *preference following*. Furthermore, we evaluate this capability under a *passive-user assumption*, where the user responds to only model queries, forcing the LLM to decide *what* to ask and *when* and explore a diverse set of tasks—including Personalized Question Answering, and Text Summarization, so as to evaluate LLM’s capabilities in realistic personalization scenarios.

Moreover, other works, similar to this one, address *preference elicitation*. For instance, GATE (Li et al., 2025) establishes the value of multi-turn interactive preference elicitation compared to other interactive (e.g., users are asked to respond to selected questions from a large pool) and non-interactive baselines. Our work extends this by providing a broad and systematic understanding of how multi-turn interactive preference elicitation behaves across a diverse range of settings (tasks, topics, models, etc.). Another set of works has aimed at improving model performance on preference elicitation. Hong et al. (2023) proposes an offline RL method to help an LLM achieve a goal through multi-turn interaction with a user. In this context, the authors consider preference elicitation, but not exclusively, and it is studied in a more restricted setting compared to our work, e.g., the topics are mostly related to travel. Wan et al. (2026) use RL to personalize LLMs for multi-turn conversations, focusing on comparing different RL objectives and reward designs. However, their evaluation lacks model diversity, as the LLM models are fixed across these varying designs. Gao et al. (2024) propose an interactive framework where, given a context, the model proposes an answer and the user edits that answer according to their preferences. The model elicits latent preferences and updates its answer using the user’s edit history. The efficacy of this approach is tested on two applications, text summarization and email writing. Overall, the above works *do not share the focus of our work* and the scope of their examination of preference elicitation is somehow narrower. *Our work is designed to systematically study the general abilities of LLMs in preference elicitation.*

3 Benchmark Setting and Methodology

In this section, we present the overall design of our benchmark and the methodology used to evaluate LLMs in latent information discovery. The overall benchmark structure is illustrated in Fig. 2, and the three evaluated tasks are shown in Fig. 3.

3.1 General Setting and Methodology

Tasks. The benchmark centers on tasks that require the discovery of hidden or implicit user information. A successful Assistant must go beyond producing a correct answer—it must uncover and incorporate the User’s latent preferences through interaction. We include three tasks that together capture this spectrum of behavior. Sample dataset entries for each of these tasks are presented in Appendix D.2.

(1) *Personalized Question Answering* and (2) *Personalized Text Summarization* are the two core personalization tasks. In both, the Assistant must identify which pieces of information about the User are relevant to the current task (e.g., dietary restrictions, stylistic preferences) and then use them to generate a tailored output. These tasks mimic realistic user–LLM interactions, where success depends on how effectively the model elicits, interprets, and applies latent user cues. In both tasks we employ a diverse set of topics to capture different scenarios. Specifically, we have five topics for task (1) and four topics for task (2).

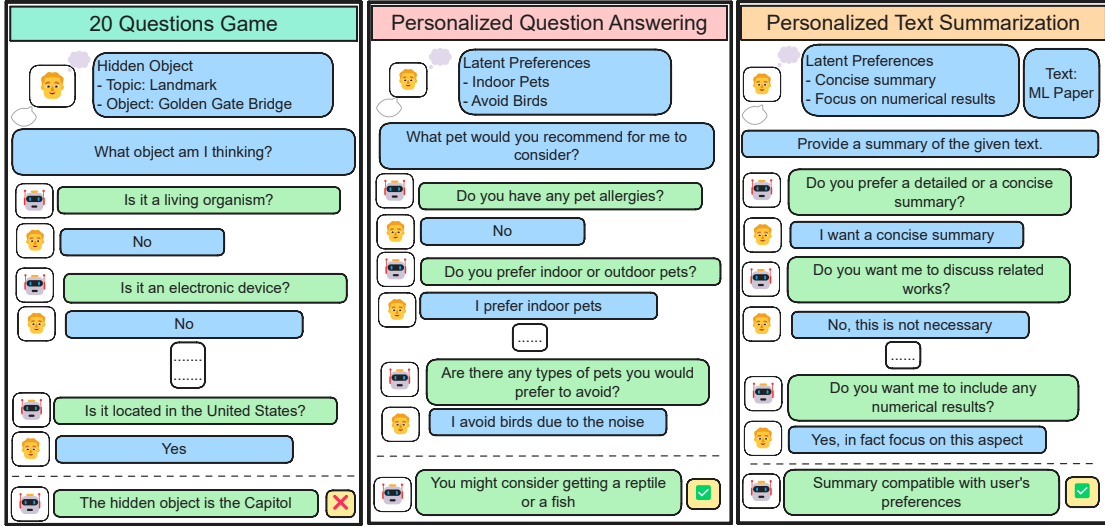


Figure 3: The two personalization tasks and the 20 Questions Game task of our benchmark. The LLM tries to elicit the user preferences (or the hidden object) through a multi-turn conversation, requesting specific information at each turn. After each turn, the Assistant produces a personalized response (here we only illustrate the response after the final turn).

(3) *The 20 Questions Game*, although not a personalization task in the conventional sense, it serves as a foundational and diagnostic component of the benchmark. The 20 Questions Game isolates the *pure reasoning process* of latent information discovery. Here, the hidden object plays the same conceptual role as a latent preference: it is information unknown to the Assistant that must be uncovered through strategic questioning. Performance in this setting illustrates how well a model can reason about and uncover hidden information before those capabilities are tested in more complex, domain-specific personalization tasks. Moreover, this task provides a simplified and controlled environment in which we can study more closely certain aspects of latent information discovery. Specifically, we analyze how the search space of hidden objects changes after each turn and assess the potential for improving the model’s performance using SFT.

Agents. Each task involves three interacting agents: the **Assistant**, the **User**, and the **Judge**. The **User** simulates a person with fixed latent preferences and provides concise, truthful responses to the Assistant’s questions. The **Assistant** acts as the model under evaluation and is the active participant in the dialogue. Its goal is to uncover the User’s hidden information and complete the task in a way that aligns with those preferences. After each conversational round, the Assistant generates a task-level output (e.g., an answer or a summary). Finally, the **Judge** operates externally, assessing whether the Assistant’s output satisfies all of the User’s latent preferences.

Multi-Turn Interaction. The dialogue unfolds as a sequence of Assistant–User exchanges. At each turn, the Assistant poses a targeted question and the User replies according to their latent preferences. The Assistant may refine its subsequent questions using all prior context. The maximum number of turns is fixed and known to the Assistant in advance, ensuring comparability across models.

Evaluation and Early Stopping. After each turn (i.e., one Assistant–User exchange), the Assistant produces a complete, personalized response to the User’s original task using all accumulated information. The Judge then evaluates whether this output satisfies every latent preference. If alignment is achieved, the conversation terminates early (*success*); otherwise, it continues until the maximum number of turns is reached (*failure*). This early-stopping mechanism allows us to measure both the quality and efficiency of latent-information discovery.

Evaluation Metrics. We use two primary metrics:

1. **Success Rate** — the proportion of instances in which the Assistant’s final output satisfies all latent preferences;
2. **Average Stop Turn** — the average number of turns required to reach success. This metric is computed over all instances and uses the maximum possible value as the stop turn in cases of failure (e.g., 20 for the 20 Questions).

Additionally, we conduct an error-type analysis in which we examine the different ways the Assistant models fail (Appendix A.3), as well as an analysis of how the search space is reduced after each turn in the 20 Questions Game (Appendix A.4). These analyses aim to take a closer look at the different models’ strategies and their effectiveness, i.e., the types of questions they ask and how effective these are at unveiling the hidden objects. Finally, we evaluate the reliability of the selected Judge and User models by examining the extent to which they adhere to their provided instructions (Appendix A.2).

Models. All agents are implemented with LLMs. To ensure reliability, the **User** and **Judge** are instantiated using strong, stable models that perform their roles consistently across runs. Specifically, we employ **GPT-4o** (Hurst et al., 2024) and **Qwen3-235B-A22B-Instruct** (Yang et al., 2025) for both, with each model handling half of the runs. For the **Assistant**, we evaluate four representative models spanning both closed- and open-source families: **GPT-4o-mini** (Hurst et al., 2024), **Claude-3.5-Haiku**, **Mistral-7B-Instruct** (Jiang et al., 2023), **Qwen2.5-7B-Instruct** and **Qwen2.5-14B-Instruct** (Yang et al., 2024).

3.2 20 Questions Game

The first component of our benchmark is the 20 Questions Game, a structured variant of the classic guessing game in which one participant selects an object and the other attempts to identify it by asking a sequence of yes-or-no questions. Each **User** is associated with a single hidden object drawn from one of five topics: *animal*, *fruit*, *landmark*, *sport*, or *weather*. The topic may either be disclosed to, or withheld from the **Assistant**, allowing us to evaluate how prior knowledge affects questioning strategies. During the interaction, the Assistant poses one question per turn, and the User responds truthfully, typically with “yes” or “no”. The dialogue lasts for at most 20 turns.

After every turn, the Assistant must, outside of the dialogue, propose its current best guess of the hidden object using all gathered information. The **Judge** then verifies whether this guess is correct. If correct, the conversation terminates early (*success*); otherwise, it continues until the 20-turn limit is reached (*failure*).

3.3 Personalized Question Answering

In Personalized Question Answering, each User is associated with a list of one to three latent preferences and a question. The preferences and the question correspond to a specific topic from the following list: medical care, personal finance, pet ownership, shopping assistance, travel restaurant.

The Assistant is given the question, and its goal is to provide a personalized answer, i.e., an answer that aligns with all the preferences in the User’s list. To achieve this, the Assistant asks the User targeted questions aimed at eliciting all relevant (to the User’s question) preferences, specifically a single question per conversation turn to which the User answers truthfully.

After each turn, we request from the Assistant a response to the User’s question (outside of the conversation), using the information gathered so far. The Judge assesses whether the Assistant’s response correctly aligns with all of the User’s latent preferences. The maximum number of conversation turns is set to 10.

3.4 Personalized Text Summarization

In this task, each user is associated with a list of three preferences and a text. The preferences and the texts correspond to a specific topic from the following list: news articles

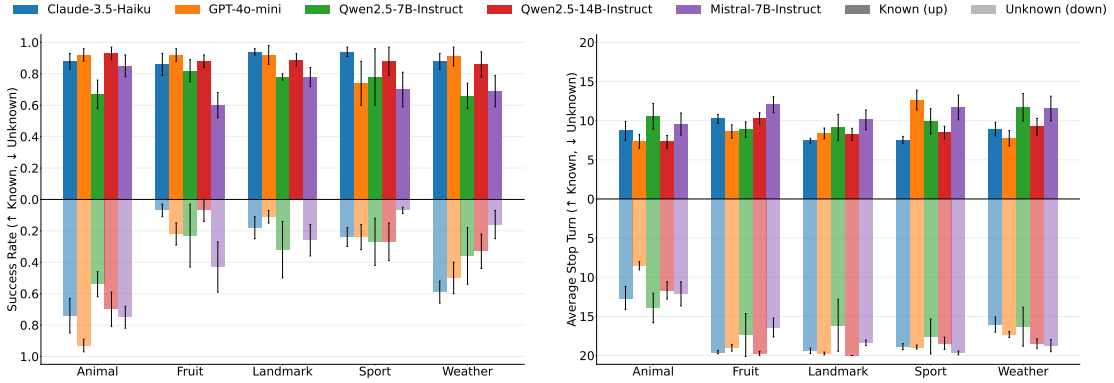


Figure 4: Results of the “20 Questions Game” experiments across different models, topics, and conditions; Topic Known $\uparrow$ (the upper part of each figure) vs Topic Unknown $\downarrow$ (the lower part of each figure).

(“cnn_dailymail”), general articles (“wikipedia”), science papers (“elsevier_PHYS”) and patent documents (“big_patent”); refer to Sec. D.1 for more details. The Assistant is given the text, and its ultimate goal is to provide a personalized summary, i.e., one that aligns with all of the User’s preferences.

To achieve this, the Assistant asks the User targeted questions aimed at understanding the User’s summarization preferences, specifically a single question per conversation turn to which the User answers truthfully. After each turn, we request the Assistant to provide a summary of the User’s text (outside of the conversation), using the information gathered so far. The Judge assesses whether the Assistant’s summary correctly aligns with all of the User’s summarization preferences. The maximum number of conversation turns is set to 10.

4 Experiments

In this section, we present our experimental setup and results across the three benchmark tasks. Our experiments aim to answer two central questions: (1) How well can current LLMs uncover hidden information through multi-turn dialogue? (2) How efficiently do they adapt newly acquired knowledge to produce personalized responses?

4.1 20 Questions Game (20QG)

We present results in Fig. 4. All numbers are averaged over six runs (three with GPT-4o as User/Judge and three with Qwen3-235B); we have 20 objects per topic.

Main observations. Performance drops sharply when the topic is unknown rather than disclosed, which is expected since the search space is much larger in the former case. The drop is less pronounced for the *animal* topic because most Assistants open with a high-yield question (e.g., “Is the object a living thing?”), quickly narrowing the hypothesis space. No single model dominates across all settings. Closed-source models (GPT-4o-mini, Claude-3.5-Haiku) tend to lead on average when the topic is known, while when the topic is unknown the situation is more mixed, as for each open model there is a topic in which it performs better than the closed-source ones. Notably, Mistral-7B-Instruct, one of the weakest models (i.e. it regularly exhibits the worst performance), outperforms the others when the topic is “fruit”, both in terms of success rate (20 percentage points higher than the second-best model) and stop-turn number (requires at least 1 less turn compared to the other models).

Although the 20QG is not a personalization task, it isolates the latent-information *discovery* process and thus offers a complementary view of models’ multi-turn reasoning: how well they plan and ask informative questions. To this end, we analyze how the size of the (hidden object) search space is reduced after each turn. In brief, we observe that the more successful models (i.e., the highest performing ones in Figure 4) consistently decrease the cardinality of the search space, whereas for the least successful models this decrease quickly slows

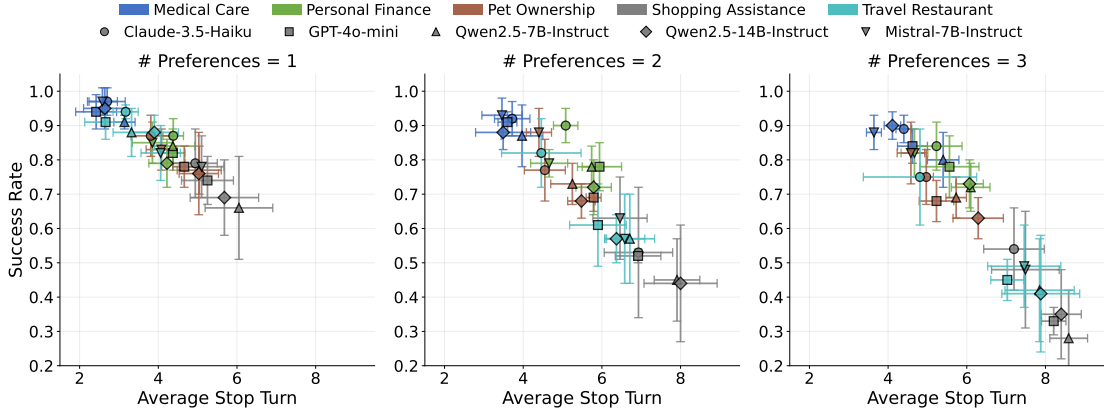


Figure 5: Results of the Personalized Question Answering experiments across different models, topics, and preference settings. The figure shows Success Rate (y-axis) vs. Average Stop Turn (x-axis) for $\# \text{Preferences} \in \{1, 2, 3\}$ (three panels). Color encodes *topic/domain* and marker shape encodes *model*; error bars reflect reported uncertainties on both axes. (i) Domain effects are prominent: clusters by topic are clearly separated and we observe easy (top left) and hard topics (bottom right); (ii) As the number of preferences increases, points generally drift toward the lower-right (lower success, higher turns), indicating that deeper personalization settings tend to be harder.

down. See Appendix A.4 for more details. We use these insights to interpret behavior on the personalization tasks.

4.2 Personalized Question Answering (PQA)

We present results in Fig. 5. All numbers are averaged over six runs (three with GPT-4o as User/Judge and three with Qwen3-235B); each topic contains 20 users.

Main observations. We observe substantial variability across topics: for a fixed model and preference count, success rates can differ by as much as 55%. For example, even with three preferences, all models achieve around 80%–90% success in *Medical Care*, whereas performance drops to 28%–54% in *Shopping Assistance*. Thus, PQA is not uniformly easy or hard; *topic* strongly determines difficulty. Claude-3.5-Haiku is usually the top performer, though Mistral-7B-Instruct is competitive and occasionally surpasses Claude-3.5 in certain instances. Average stop-turn values remain modest in most settings—in only a third of the cases we have stop turn ≥ 6 —indicating that effective preference elicitation often occurs within the first few exchanges. An instance of two conversations for the Travel Restaurant topic can be found in Fig. 6; see also Fig. 11 in the Appendix.

4.3 Personalized Text Summarization (PTS)

We report results in Fig. 7. All numbers are averaged over six runs (three with GPT-4o as User/Judge and three with Qwen3-235B); each topic contains 20 texts.

Main observations. Success rates decrease as the number of preferences increases, reflecting the added constraint burden. Closed-source models are, on average, strongest; however, Qwen2.5-7B-Instruct occasionally achieves comparable performance. Stop-turn values are generally modest—in most cases < 6 —suggesting that, for summarization, models can identify and apply key preferences with not too many clarification turns.

5 Additional Experiments and Analysis

5.1 Evaluating SFT Performance in the 20QG

We study the possibility of improving model performance on the 20QG using SFT. Specifically, we create a training set by collecting the conversation turns of the GPT-4o-mini model from successful instances (i.e., the hidden object was identified) across all five topics. Each

Preference: I avoid spicy foods.	Q: Can you recommend some restaurants to try during my visit to Mumbai?
GPT-4o-mini 1.A: What type of cuisine do you enjoy the most? 1.U: I enjoy a variety of cuisines, but I especially like Indian, Italian, and Chinese food. 2.A: Are you looking for fine dining experiences, casual eateries, or a mix of both during your visit? 2.U: I'm looking for a mix of both fine dining experiences and casual eateries during my visit. 3.A: Do you have any dietary restrictions or preferences that I should consider when recommending restaurants? 3.U: I avoid eating spicy foods entirely.	
Qwen-2.5-7B-Instruct 1.A: What kind of cuisine are you particularly interested in trying during your visit to Mumbai? 1.U: I'm open to trying a variety of cuisines, but I have a particular interest in exploring authentic Indian dishes. 2.A: Are there any specific regions or states within India that you're curious to explore through their cuisine? 2.U: I'm curious to explore the cuisine of Maharashtra. I'd love to try some authentic local dishes from this region. 3.A: Are there any specific types of dishes from Maharashtra that you're excited to try, such as vegetarian or seafood? 3.U: I'm excited to try a variety of dishes from Maharashtra, including both vegetarian and seafood options. 4.A: Are there any specific festivals or occasions in Mumbai that you're planning to experience, which might influence the type of food you'd like to try? 4.U: I'm not visiting during any specific festivals or occasions, so my focus will mainly be on exploring the local cuisine. 5.A: Are there any dietary restrictions or preferences you have, such as vegetarian, vegan, or gluten-free options? 5.U: I avoid eating spicy foods entirely, so I'd appreciate recommendations for restaurants that offer milder options.	

Figure 6: An instance of the PQA task for the “Travel Restaurant” topic, along with the conversations of two models. We note that GPT-4o begins by asking more general questions, allowing it to identify the latent preference (e.g., avoid spicy food) earlier than Qwen-2.5-7B-Instruct. In contrast, the latter model immediately specializes its questions after the first turn (e.g., asking about specific regions) and becomes “trapped” in a narrow conversation, thereby postponing the identification of more general preferences.

Objects Set	Topic	Before SFT		After SFT	
		Success Rate	Stop Turn	Success Rate	Stop Turn
Original	Fruit	0.60 ± 0.08	12.04 ± 1.02	0.71 ± 0.09	8.04 ± 0.75
	Weather	0.69 ± 0.10	11.55 ± 1.58	0.84 ± 0.05	8.18 ± 0.82
Unseen	Fruit	0.29 ± 0.03	17.72 ± 0.58	0.26 ± 0.08	10.61 ± 0.44
	Weather	0.43 ± 0.04	14.54 ± 0.64	0.53 ± 0.06	9.37 ± 0.76

Table 1: The performance of Mistral-7B-Instruct before and after SFT across two different object sets: the object set used in our main evaluation (from whose conversations the training set was built) and an unseen set of objects. (i) On the original objects set, the SFT model outperforms the pretrained model. (ii) On the unseen objects set, the SFT model performs better on the “weather” topic, while on “fruit” there is a trade-off: a small decrease in success rate but a significant improvement in the stop-turn number.

sample consists of a prompt portion (the conversation up to turn k) and a completion portion (the Assistant’s question at turn $k+1$). We perform SFT on the Mistral-7B-Instruct model. For evaluation, we construct the following object sets: (1) a set that includes only objects used in our main experiments, which we refer to as “original”; and (2) a set consisting of objects that were not used in our main experiments, i.e., an “unseen” set of objects. In both sets, we include only objects from the “fruit” and “weather” topics. We note that the above distinction means that the hidden objects included in set (1) are a subset of the objects included in the conversations from which the training set was built. As a result, object set (1) corresponds to a less challenging scenario than set (2). The results are given in Table 1.

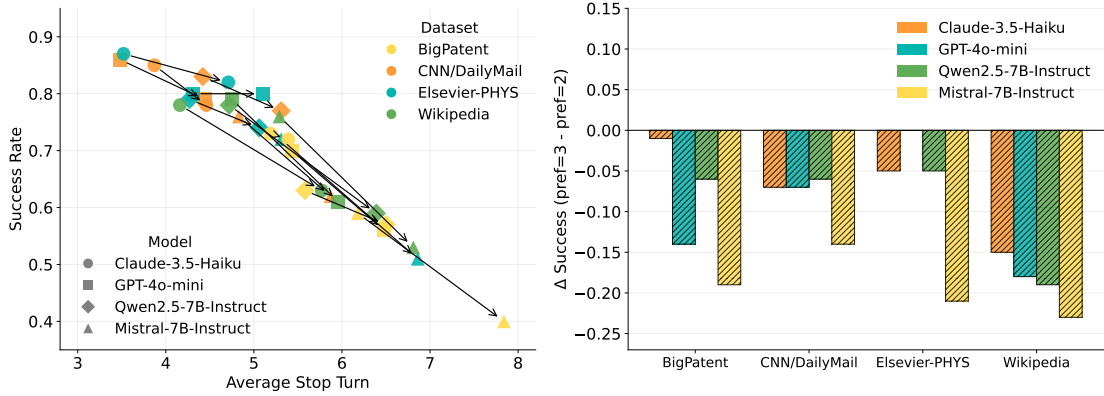


Figure 7: **Personalized Text Summarization results** across different models, topics, and preference settings. Each arrow connects the same (topic, model) pair as the number of preferences increases from 2 to 3. (i) The scatter (left) reveals an efficiency–accuracy frontier: points nearer the upper-left indicate higher success with fewer turns; as the number of preferences increases (arrows), most (model, dataset) pairs shift down/right, showing longer elicitation and lower success. (ii) Domain effects are pronounced—ELSEVIER-PHYS and CNN/DAILYMAIL clusters tend to remain higher with shorter stops, while BIGPATENT drifts lower with longer stops, indicating greater difficulty in aligning to latent preferences for technical prose. (iii) Model-wise, closed-source models typically occupy stronger regions, but open-weight QWEN2.5-7B-INSTRUCT is often competitive in efficiency (shorter stop turns). (iv) The difference-only bar chart (right) makes the penalty of increasing preference dimensionality explicit: all bars extend downward, quantifying the drop from #Prefs=2 to #Prefs=3.

We observe that, on the original objects set (set (1)), the SFT model outperforms the pre-trained model on both the “fruit” and “weather” topics. On the unseen objects set (set (2)), we see a 10% increase on the “weather” topic and a small 3% decrease on the “fruit” topic; however, we note that the SFT model is much more efficient in the latter case, as the stop turn drops from 17.7 to 10.6. Finally, we note that this is a preliminary study aimed at assessing whether there is potential for performance improvement. Indeed, we observe a positive signal, which motivates further study of SFT methods for improving latent information discovery.

We also studied the use of reinforcement learning methods to improve model performance on the 20QG task, obtaining similarly positive results. For more details, see Appendix A.5.

5.2 Utility of Multi-turn Interaction

A key aspect of our benchmark is that the Assistant–User interaction consists of multiple turns. This is necessary because the preferences are complex and cannot be uncovered with a single query. While this is intuitively clear, to quantify the exact utility of multi-turn interaction, we compute the success rate across all tasks and topics (averaged across all models) when the maximum turn number is 1. The single-turn performance can be found in Tables 7, 8 and 9. We notice that, across all tasks and topics, single-turn performance is significantly inferior, even compared to the weakest model. Notably, the difference between the success rate of the weakest model and the single-turn success rate in the PQA task varies from around 20% to 60%.

6 Conclusion

In this work we study the ability of LLMs to infer latent user preferences through conversation and to use them to generate personalized responses. To this end, we develop a benchmark consisting of three tasks. Our experiments show that in general LLMs possess the ability to uncover latent preferences. However, there are significant variations in the extent to which they demonstrate this ability, depending on the task and other factors. The next step is to explore approaches for improving the performance of LLMs on such tasks, particularly for weaker models and more challenging topics.

Acknowledgments

M. Hong and I. Tsaknakis are supported in part by NSF grant CIF-2414372 and a Cisco Research Award. A. Garcia is supported by NSF grant CCF-2414373.

References

- Saleh Afzoon, Usman Naseem, Amin Beheshti, and Zahra Jamali. Persobench: Benchmarking personalized response generation in large language models. *arXiv preprint arXiv:2410.03198*, 2024.
- Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, and Wanli Ouyang. MT-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7421–7454, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.401. URL <https://aclanthology.org/2024.acl-long.401/>.
- Felix Busch, Lena Hoffmann, Christopher Rueger, Elon HC van Dijk, Rawen Kader, Esteban Ortiz-Prado, Marcus R Makowski, Luca Saba, Martin Hadamitzky, Jakob Nikolas Kather, et al. Current applications and challenges in large language models for patient care: a systematic review. *Communications Medicine*, 5(1):26, 2025.
- Zhendong Chu, Shen Wang, Jian Xie, Tinghui Zhu, Yibo Yan, Jinheng Ye, Aoxiao Zhong, Xuming Hu, Jing Liang, Philip S Yu, et al. Llm agents for education: Advances and applications. *CoRR*, 2025.
- Ge Gao, Alexey Taymanov, Eduardo Salinas, Paul Mineiro, and Dipendra Misra. Aligning llm agents by learning latent preference from user edits. *Advances in neural information processing systems*, 37:136873–136896, 2024.
- Joey Hong, Sergey Levine, and Anca Dragan. Zero-shot goal-directed dialogue via rl on imagined conversations. *arXiv preprint arXiv:2311.05584*, 2023.
- 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.
- Dongsheng Jiang, Yuchen Liu, Songlin Liu, Jin’e Zhao, Hao Zhang, Zhen Gao, Xiaopeng Zhang, Jin Li, and Hongkai Xiong. From clip to dino: Visual encoders shout in multi-modal large language models. *arXiv preprint arXiv:2310.08825*, 2023.
- Juyong Jiang, Fan Wang, Jiasi Shen, Sungju Kim, and Sunghun Kim. A survey on large language models for code generation. *arXiv preprint arXiv:2406.00515*, 2024.
- Ishita Kumar, Snigdha Viswanathan, Sushrita Yerra, Alireza Salemi, Ryan A Rossi, Franck Dernoncourt, Hanieh Deilamsalehy, Xiang Chen, Ruiyi Zhang, Shubham Agarwal, et al. Longlamp: A benchmark for personalized long-form text generation. *arXiv preprint arXiv:2407.11016*, 2024.
- Wai-Chung Kwan, Xingshan Zeng, Yuxin Jiang, Yufei Wang, Liangyou Li, Lifeng Shang, Xin Jiang, Qun Liu, and Kam-Fai Wong. MT-eval: A multi-turn capabilities evaluation benchmark for large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 20153–20177, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1124. URL <https://aclanthology.org/2024.emnlp-main.1124/>.
- Belinda Li, Alex Tamkin, Noah Goodman, and Jacob Andreas. Eliciting human preferences with language models. In *International Conference on Learning Representations*, volume 2025, pp. 80984–81013, 2025.

- Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7370–7392, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.399. URL <https://aclanthology.org/2024.acl-long.399/>.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- Amos Tversky and Shmuel Sattath. Preference trees. *Psychological review*, 86(6):542, 1979.
- Yanming Wan, Jiaxing Wu, Marwa Abdulhai, Lior Shani, and Natasha Jaques. Enhancing personalized multi-turn dialogue with curiosity reward. *Advances in Neural Information Processing Systems*, 38:155857–155894, 2026.
- Longyue Wang, Zefeng Du, Wenxiang Jiao, Chenyang Lyu, Jianhui Pang, Leyang Cui, Kaiqiang Song, Derek Wong, Shuming Shi, and Zhaopeng Tu. Benchmarking and improving long-text translation with large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 7175–7187, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.428. URL <https://aclanthology.org/2024.findings-acl.428/>.
- Wenxuan Wang, Zizhan Ma, Zheng Wang, Chenghan Wu, Jiaming Ji, Wenting Chen, Xiang Li, and Yixuan Yuan. A survey of LLM-based agents in medicine: How far are we from baymax? In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 10345–10359, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.539. URL <https://aclanthology.org/2025.findings-acl.539/>.
- Changrong Xiao, Sean Xin Xu, Kunpeng Zhang, Yufang Wang, and Lei Xia. Evaluating reading comprehension exercises generated by LLMs: A showcase of ChatGPT in education applications. In Ekaterina Kochmar, Jill Burstein, Andrea Horbach, Ronja Laarmann-Quante, Nitin Madnani, Anaïs Tack, Victoria Yaneva, Zheng Yuan, and Torsten Zesch (eds.), *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pp. 610–625, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.bea-1.52. URL <https://aclanthology.org/2023.bea-1.52/>.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.
- Daoguang Zan, Bei Chen, Fengji Zhang, Dianjie Lu, Bingchao Wu, Bei Guan, Wang Yongji, and Jian-Guang Lou. Large language models meet NL2Code: A survey. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7443–7464, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.411. URL <https://aclanthology.org/2023.acl-long.411/>.
- Yizhe Zhang, Jiarui Lu, and Navdeep Jaitly. Probing the multi-turn planning capabilities of LLMs via 20 question games. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1495–1516, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.82. URL <https://aclanthology.org/2024.acl-long.82/>.

Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. Do llms recognize your preferences? evaluating personalized preference following in llms. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Zheng Zhao, Clara Vania, Subhradeep Kayal, Naila Khan, Shay B Cohen, and Emine Yilmaz. PersonaLens: A benchmark for personalization evaluation in conversational AI assistants. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18023–18055, Vienna, Austria, July 2025b. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.927. URL <https://aclanthology.org/2025.findings-acl.927/>.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. Multilingual machine translation with large language models: Empirical results and analysis. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2765–2781, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.176. URL <https://aclanthology.org/2024.findings-naacl.176/>.

Topic	Model	Passive User		Active User	
		Success Rate	Stop Turn	Success Rate	Stop Turn
CNN DailyMail	Claude-3.5-Haiku	0.78 ± 0.06	4.31 ± 0.41	0.98 ± 0.03	4.31 ± 0.30
	GPT-4o-mini	0.80 ± 0.05	3.82 ± 0.37	1.00 ± 0.00	4.49 ± 0.38
Wikipedia	Claude-3.5-Haiku	0.67 ± 0.05	4.89 ± 0.80	0.80 ± 0.05	5.69 ± 0.47
	GPT-4o-mini	0.71 ± 0.03	4.69 ± 0.32	0.98 ± 0.03	5.33 ± 0.83
Big Patent	Claude-3.5-Haiku	0.80 ± 0.00	4.78 ± 0.08	0.96 ± 0.06	4.53 ± 0.73
	GPT-4o-mini	0.64 ± 0.08	6.51 ± 0.60	1.00 ± 0.00	4.60 ± 0.18

Table 2: Comparison of the performance of a passive (default case) and an active user on the Text Summarization task.

A Additional Experiments and Analysis

A.1 Active vs Passive User Comparison

Our default benchmark assumes a *passive* User, defined as someone who only responds to the Assistant’s questions without volunteering additional information unless explicitly prompted (see Sec. 1). This assumption reflects the common real-world scenario in which users provide short, underspecified prompts and expect the model to take initiative in eliciting what is needed. Benchmarking under this assumption provide an assessment of the *worst case*, yet still common, scenario.

To explore how this assumption affects task difficulty, we consider a variant of the Personalized Text Summarization task with an *active* User. In this setting, the Assistant still poses a question each turn but also provides a preliminary summary. The User may either answer the question or proactively comment on the summary (e.g. by requesting a more concise summary or requesting the inclusion of specific information from the original text), whichever they judge more helpful for guiding the Assistant toward alignment. Note that in this case, since users are actively engaging the conversation, we expect that they will be directly revealing their preferences, therefore we expect that the Assistant will be able to achieve a higher success rate. See Fig. 8 for an example conversation with a passive and an active User.

Table 2 shows that success rates with an active User are higher, often approaching 100%. While this result demonstrates that most models can produce high-quality personalized outputs once preferences are explicitly surfaced, it also underscores that such an interaction style is largely impractical in ordinary use. Real users rarely maintain that level of engagement or meta-awareness about their own preferences. Hence, the *passive* setting—where the model must take the lead in discovering latent information—remains the more realistic and challenging benchmark for studying personalization in LLMs.

A.2 Judge and User LLM Evaluation

Two key components of the benchmark are the Judge and User agents. The role of the Judge is to assess whether the Assistant’s output satisfies all of the User’s latent preferences and the role of the User is to simulate a person with a certain set of latent preferences. It is clear that these agents, and especially the Judge, play a critical role in ensuring accurate and reliable results. Therefore, selecting strong and reliable LLMs for the User and the Judge agents is crucial. In our experiments, we employed GPT-4o and Qwen3-235B-A22B-Instruct to play the role of the User and the Judge agents, given their well-established capabilities. To further justify this choice, we conducted additional experiments to assess its suitability.

More precisely, we evaluate the extent to which the User and Judge LLMs follow the instructions provided to them in the Personalized Question Answering task. To do so, we employ three competent LLMs, GPT-4.1, Claude 4.5 Sonnet, and Claude 4.6 Sonnet, to assess instruction adherence. Each LLM receives as input the User-Assistant conversation, the Assistant’s responses to the User’s questions after each turn, and the Judge’s assessments of those responses. The LLMs then determine whether the Judge and the User models followed their respective instructions, and the final assessment is computed using majority

Preferences

1. Focus primarily on the medical details of Theia’s injuries and treatment rather than the emotional aspects of her story.
2. Emphasize the fundraising efforts and financial aspects of her care more than the sequence of traumatic events.
3. Give equal attention to the similar story from Tampa, FL as a parallel case rather than treating it as a brief footnote.

Text: (CNN) Never mind cats having nine lives. A stray pooch in Washington State has used up at least three of her own after being hit by a car, apparently whacked on the head with a hammer in a misguided mercy killing and then buried in a field – only to survive. (...) She suffered a dislocated jaw, leg injuries and a caved-in sinus cavity – and still requires surgery to help her breathe. The veterinary hospital’s Good Samaritan Fund committee awarded some money to help pay for the dog’s treatment, but Mellado has set up a fundraising page to help meet the remaining cost of the dog’s care. (...) Theia is not the only animal to apparently rise from the grave in recent weeks. A cat in Tampa, Florida, found seemingly dead after he was hit by a car in January (...)

Passive User

- 1.A: Would you prefer a summary that focuses more on the dog’s survival story or the broader context of animal resilience?
- 1.U: I would prefer a summary that focuses more on the medical details of the dog’s injuries and treatment.
- 2.A: Would you like the medical details presented in a clinical, technical tone or in a more narrative, empathetic style?
- 2.U: I would like the medical details presented in a clinical, technical tone.
-

Active User

- 1.A: ****Summary****: A resilient stray dog named Theia survived multiple life-threatening incidents, including being hit by a car, apparently struck with a hammer, and buried in a field. Four days after her presumed death, she crawled to a nearby farm and was rescued by Sara Mellado. Despite severe injuries like a dislocated jaw and damaged sinus cavity, Theia is receiving medical treatment at Washington State University’s Veterinary Teaching Hospital. A fundraising effort has already exceeded \$10,000 to support her recovery, with Mellado committed to finding her a loving, safe home.
- **Question****: Would you prefer a summary that focuses more on the emotional aspects of Theia’s survival story, or one that emphasizes the factual details of her medical condition and rescue?
- 1.U: I would prefer a summary that emphasizes the factual details of Theia’s medical condition and the treatment she is receiving.
- 2.A: ****Summary****: Theia, a one-year-old stray dog, suffered extensive medical trauma including a car accident, potential blunt force injury, and burial. Her specific medical conditions include a dislocated jaw, leg injuries, and a caved-in sinus cavity requiring surgical intervention. Currently receiving treatment at Washington State University’s Veterinary Teaching Hospital, her care is partially funded by the Good Samaritan Fund. Sara Mellado has established additional fundraising, which has already surpassed the \$10,000 target, to cover her extensive medical procedures necessary for nasal damage repair and jaw reset.
- **Question****: Would you like me to include more specific details about the types of medical procedures Theia requires or the potential long-term prognosis for her recovery?
- 2.U: Please include more details on the fundraising efforts and financial aspects of Theia’s care.
-

Figure 8: An instance of the Personalized Text Summarization task for the “CNN DailyMail” topic. We present two conversations illustrating cases where the User agent acts as either a passive or an active participant. A passive User only responds to the Assistant’s queries, whereas an active User either provides feedback on the summary or answers the question. Indeed, we observe that the active User answers the Assistant’s question in the first turn, while in the second turn it ignores the question and instead provides direct feedback on the summary, requesting changes to better align with the latent preferences.

voting. We report the percentage of conversations in which the Judge and the User agents adhere to the given instructions.

Model	Judge	User
GPT-4o	0.72	0.76
Qwen3-235B-A22B-Instruct	0.80	0.92
GPT-4o-mini	0.41	0.45
Mistral-7B-Instruct	0.10	0.42

Table 3: The results of evaluating different models in the roles of the Judge and User. The instruction adherence rate is the percentage of conversations in which the model adheres to the Judge/User’s instructions. GPT-4o and Qwen3-235B-A22B-Instruct clearly outperform the other models in terms of instruction adherence.

We evaluate four LLMs in the roles of the Judge and User agents: GPT-4o, Qwen3-235B-A22B-Instruct (our choices), GPT-4o-mini, and Mistral-7B-Instruct. The results are presented in Table 3. GPT-4o and Qwen3-235B-A22B-Instruct clearly outperform the other models in terms of instruction adherence, thereby validating our choice to use them as the Judge/User model.

We would like to stress here that the 72% instruction adherence rate of GPT-4o does not imply that the evaluations in 28% of the conversations are necessarily erroneous. This is because the assessments are largely subjective. Consider an example where the User’s preferences is to “avoid hospitals”. The Assistant’s answer suggests (among multiple other solutions) the following: “If your headaches persist, it would be advisable to consult a healthcare professional”. The Judge considers this answer acceptable as “The recommendation to consult a healthcare professional is general advice and does not contradict the user’s preferences, as it does not specify hospital visits.”. On the contrary the Judge evaluator flags this is assessment as incorrect because of the reference to “healthcare professional”. From our perspective, neither judgment is clearly right or wrong, rather, the disagreement reflects a matter of interpretation.

Despite the above, we note that this metric remains useful, as it (1) helps build confidence in a model’s reliability and (2) highlights measurable differences between candidate Judge/User models, enabling us to select the most suitable one (e.g. Table 3 provides a clear signal that GPT-4o is more capable than GPT-4o-mini as a Judge/User). Furthermore, the presence of such ambiguities reinforces our decision to use two different Judge/User models to compute the results (six runs in total, three runs per model). Each model exhibits its own biases and subjectivity, and using models from different families helps mitigate individual model biases, providing more robust evaluations.

A.3 Error Type Analysis

To more comprehensively evaluate model behavior in personalization tasks, we conduct an **error type analysis** on the Personalized Question Answering task. The goal is not only to assess *whether* the Assistant ultimately produces a correct answer, but also to examine *how* its reasoning trajectory evolves and where it diverges from the intended path. We categorize observed errors into two groups: *process errors*, which occur during the Assistant–User interaction, and *result errors*, which appear in the Assistant’s final output.

Process Errors. Process errors capture failures in the Assistant’s ability to discover and maintain the User’s latent preferences during conversation:

1. **Preference Elicitation Error:** The Assistant fails to elicit one or more of the User’s preferences, often because its questions are too generic, redundant, or misdirected toward irrelevant aspects.
2. **Preference Reinforcement Error:** The Assistant successfully uncovers a preference but later neglects or contradicts it in subsequent turns, indicating weak conversational memory or contextual tracking.

Result Errors. Result errors reflect issues in how the Assistant integrates the gathered preferences into its final recommendation:

1. **Preference Omission Error:** The Assistant’s final answer disregards one or more explicitly identified preferences, reverting to a generic or incomplete response.
2. **Preference Dilution Error:** The Assistant acknowledges the preferences but applies them only partially, producing responses that align loosely or inconsistently with the User’s intent.

A.4 Search Space Analysis on the 20 Questions Game

In this section, we aim to better understand how each model’s reasoning process and questioning strategy prunes the search space of hidden objects in the 20 Questions game. To this end, for each topic we generate a list of the most common objects, and we track the evolution of this set throughout each conversation. That is, after each turn, we remove from the set any objects that can be ruled out. For instance, we generate a list of 300 landmarks. These were selected such that they include the most important landmarks while covering a sufficient number of instances from each continent (e.g. it includes 22 landmarks from Oceania, a small but notable number, and 116 landmarks from Europe). If at some turn the Assistant asks, “Is it located in Europe?” and the user responds “No”, we remove all European landmarks from the object set. Finally, for each model, we repeat this process across all of its conversations (spanning all topics, assuming the topic is known), track the cardinality of the objects set, and average the results. The resulting plots are shown in Figure 9.

Overall, we observe that the questioning strategy of Claude-3.5-Haiku, GPT-4o-mini and Qwen2.5-14B-Instruct is more effective than that of Qwen2.5-7B-Instruct and Mistral-7B-Instruct. Specifically, we observe that the former set of models consistently decreases the cardinality of the object set. In contrast, the latter set of models reduces the cardinality rapidly during the first five turns, however, this reduction slows thereafter and ultimately plateaus.

To make things more concrete we take a closer look at the conversation using Claude-3.5-Haiku and Mistral-7B-Instruct when the topic is “landmark”. We notice that the questions of Claude-3.5-Haiku are more generic, and thus they prune the search space more effectively. For instance, some typical questions are about the country (“Is this landmark located in Egypt?”) or the continent (“Is this landmark located in Asia?”) or other high-level characteristics (“Is this landmark man-made?”, “Is this landmark a building?”). On the other hand, in Mistral-7B-Instruct, while the model still asks generic questions similar to those above, it also tends to ask more specialized questions, sometimes even in the early turns. When these questions receive a negative answer (which is the most likely outcome) they only slightly prune the search space. For instance, in some conversations we notice a sequence of questions such as the following (only present the first four turns): “Is this landmark located in a coastal area?”, “Is this landmark a peninsula?”, “Is this landmark an island?”, “Is this landmark a lighthouse?”. This explains why the decrease in the cardinality of the objects set slows down quickly in the case of Mistral-7B-Instruct, as illustrated in Figure 9.

A.5 Evaluating RL Training Performance in the 20QG

We investigate the use of reinforcement learning (RL) methods to improve the models’ performance in latent information discovery.

Setting. We consider the 20 Questions Game task and specifically the case where the object’s topic is “landmark” and is known at the beginning of the conversation. Here we focus on the Qwen2.5-7B-Instruct model. To simplify the analysis we note that in this section evaluation is performed directly within the conversation (i.e., the assistant provides a guess during the conversation, e.g., “Is it the Eiffel Tower?”), rather than externally (i.e., the assistant is asked to provide a guess at the end of each turn using all the information gathered to that point) as in the rest of the text. Moreover, we consider two different instruction prompt templates for the Assistant: (1) “Original” which corresponds to the template used in the main experiments (with a minor modification to accommodate the different evaluation procedure); and (2) “Structured” a template in which the Assistant is prompted to plan and

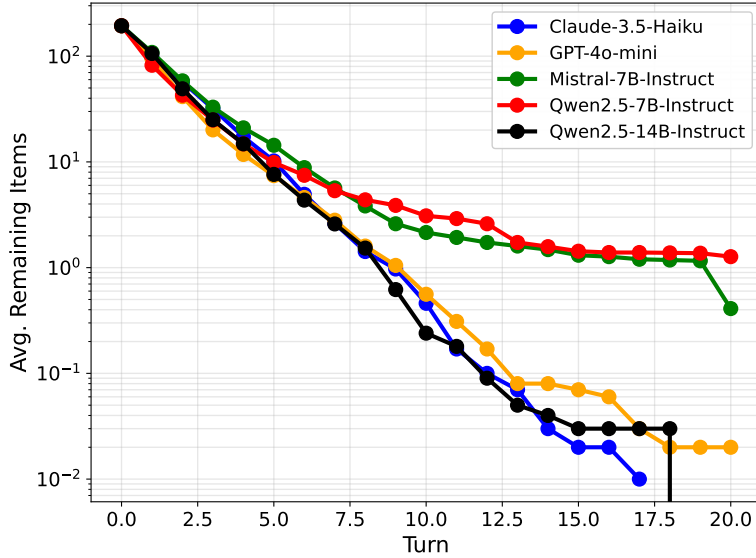


Figure 9: The evolution of the cardinality of the hidden objects set as a function of the number of turns. We observe that the questioning strategy of Claude-3.5-Haiku, GPT-4o-mini and Qwen2.5-14B-Instruct is more effective than that of Qwen2.5-7B-Instruct and Mistral-7B-Instruct.

Objects Set	Configuration	Success Rate	Stop Turn
Training	Original – Before RL	0.60 ± 0.04	13.12 ± 0.39
Training	Structured – Before RL	0.45 ± 0.04	11.35 ± 1.22
Training	Structured – After RL	0.75 ± 0.04	9.30 ± 0.41
Unseen	Original – Before RL	0.47 ± 0.12	13.68 ± 0.68
Unseen	Structured – Before RL	0.35 ± 0.08	14.59 ± 1.14
Unseen	Structured – After RL	0.50 ± 0.04	11.56 ± 0.76

Table 4: The performance of Qwen2.5-7B-Instruct before and after RL training over two different object sets (“Training” which includes the objects used for RL training, and “Unseen” for objects not used for training). We observe the following: (i) using the “Structured” template with an untrained model leads to decreased performance compared to the “Original” template. (ii) the RL-trained model using the “Structured” template outperforms the untrained models regardless of the template.

respond in a more structured manner. Specifically, the assistant is instructed to identify the hidden object by planning a sequence of high-level subgoals, each consisting potentially of multiple questions. At each turn, the assistant asks one question corresponding to the current subgoal. Finally, the model is trained using the Proximal Policy Optimization (PPO) algorithm (Schulman et al., 2017).

Results. In Table 4 we present the results. First, we observe that using the “Structured” template with an untrained model leads to decreased performance compared to the “Original” template. This is likely because the model is forced to follow a specific reasoning process without sufficient training on how to effectively leverage it. Second, the RL-trained model using the “Structured” template outperforms the untrained models regardless of the template. Overall, we observe a positive signal, which motivates further investigation of RL methods for improving latent information discovery.

A.6 Correlation Analysis Across Tasks

We study the correlations between model rankings across tasks. More precisely, for each task, we average each model’s success rate across all topics and settings (i.e., numbers of preferences in PQA and PTS) and rank the models according to the resulting average; we omit Qwen2.5-14B-Instruct as we did not test it on the PTS task. For the 20QG, we consider only the subtask in which the topic is known. This distinction is important because, in PQA and PTS, the topic, namely, the User’s question in PQA or the text in PTS, is always known. Therefore, it is more meaningful to study the correlations of the PQA and PTS tasks with the 20QG when the topic is also known. The results are presented in Table 5. We observe that the relative performance of the models is the same on 20QG (with the topic known) and PTS. We also find that the best-performing model (Claude-3.5-Haiku) on 20QG (with the topic known) is also the best-performing model on both PQA and PTS. In fact, we notice that closed-source models generally outperform their open-source counterparts. Overall, we observe that the models’ relative performance does not vary substantially and remains fairly consistent across tasks.

Model	20QG-Topic Known	PQA	PTS
Claude-3.5-Haiku	1	1	1
GPT-4o-mini	2	3	2
Qwen2.5-7B-Instruct	3	4	3
Mistral-7B-Instruct	4	2	4

Table 5: Model rankings across different tasks.

A.7 Evolution of the Models’ Success Rate Across Turns

We study how the models’ success rates evolve with the number of turns. At each turn, we compute the proportion of conversations, across all topics and numbers of preferences, that have already been successfully completed. The resulting curves are shown in Figure 10. We observe that the proportion of successful conversations increases with the number of turns, but the marginal improvement diminishes over time. This pattern suggests that many easy instances are resolved within the first few turns, whereas the remaining instances are substantially more difficult. Although some of these harder instances are gradually resolved, progress is not uniform, and many additional turns may be required. Together with the single-turn performance reported in Tables 7, 8, and 9, these results highlight the necessity and value of multi-turn interaction for preference elicitation.

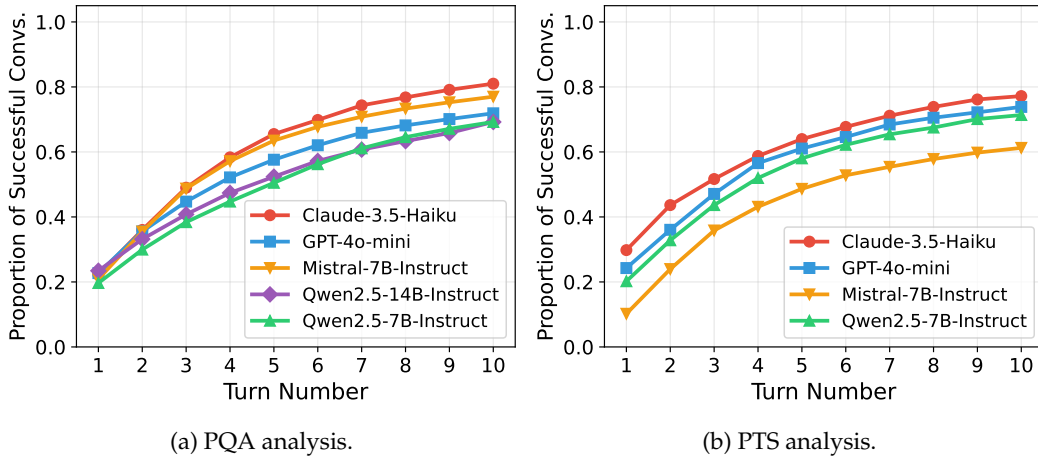


Figure 10: Evolution of the proportion of successful conversations across turns for different models on the PQA and PTS tasks.

A.8 Evaluating SFT for Cross-Topic Generalization on 20QG

We conducted additional experiments to evaluate cross-topic generalization in SFT-fine-tuned models. Specifically, we fine-tuned the Mistral-7B-Instruct model using successful conversations (i.e., those in which the hidden object was correctly identified) generated by GPT-4o-mini and Claude-3.5-Haiku on the landmark topic. The performance of the SFT-trained model on the landmark topic and two other unseen topics is shown in Table 6. We note that, for simplicity, we performed only three runs during evaluation, using GPT-4o as Judge/User. Our findings indicate that training on one topic improves performance on other unseen topics. However, at the same time, we observe some performance deterioration on the topic landmark, on which we trained the model. Overall, the results show that cross-topic generalization is possible in principle, however, a more thorough study is required to understand whether we can maintain performance on all topics and to fully understand the limitations of this generalization.

Evaluation Topic	Before SFT		After SFT	
	Success Rate	Stop Turn	Success Rate	Stop Turn
Fruit	0.65 ± 0.04	11.75 ± 0.36	0.95 ± 0.00	8.37 ± 0.25
Weather Phenomenon	0.68 ± 0.12	12.12 ± 2.06	0.78 ± 0.02	10.15 ± 0.43
Landmark	0.80 ± 0.07	9.38 ± 1.17	0.67 ± 0.02	13.75 ± 0.93

Table 6: The performance of Mistral-7B-Instruct before and after SFT on the landmark topic. The SFT-trained model is evaluated on the landmark topic and two other topics not used for training.

B Additional Experimental Details

B.1 Experiment Tables

For completeness, we include the tables of results from the experiments discussed in the main text. The results for the 20 Question Game task are given in Table 7, for the Personalized Question Answering task in Table 8, and for the Text Summarization task in Table 9.

B.2 Sample Conversations

In Figure 11 we present an instance of the Personalized Question Answering task for the “Travel Restaurant” topic, along with the conversations of two models. This is the same instance discussed in Fig 6 in the main text. Here we extend this example by providing the conversations conducted by the following models: Claude-3.5-Haiku and Mistral-7B-Instruct. The observations are similar to the ones from the main text. The model that performs best in this case, Claude-3.5-Haiku, begins with general questions, allowing it to identify the latent preference (e.g., avoid spicy food) earlier than the other model. In contrast, Mistral-7B-Instruct immediately narrows its focus after the first turn (e.g., asking about specific dishes) and becomes “trapped” in a narrow line of questioning for two turns.

C Implementation Details

C.1 LLM Models

In the experiments we used the following LLM models:

1. GPT-4o-mini (Hurst et al., 2024)
 - Link: <https://openai.com/api/>
 - License: proprietary (accessed through OpenAI API)
2. Claude-3.5-Haiku
 - Link: <https://www.claude.com/platform/api>

Topic	Model	Topic Known		Topic Unknown	
		Success Rate	Stop Turn	Success Rate	Stop Turn
Animal	Claude-3.5-Haiku	0.88 ± 0.05	8.71 ± 1.20	0.74 ± 0.11	12.66 ± 1.48
	GPT-4o-mini	0.92 ± 0.04	7.36 ± 0.89	0.93 ± 0.04	8.53 ± 0.53
	Qwen2.5-7B-Instruct	0.67 ± 0.09	10.57 ± 1.63	0.54 ± 0.08	13.93 ± 1.89
	Qwen2.5-14B-Instruct	0.93 ± 0.04	7.30 ± 0.81	0.70 ± 0.11	11.69 ± 1.10
	Mistral-7B-Instruct	0.85 ± 0.07	9.57 ± 1.41	0.75 ± 0.07	12.12 ± 1.53
	Single-Turn	0.10 ± 0.05	1.00 ± 0.00	0.02 ± 0.02	1.00 ± 0.00
Fruit	Claude-3.5-Haiku	0.86 ± 0.07	10.22 ± 0.58	0.07 ± 0.04	19.57 ± 0.20
	GPT-4o-mini	0.92 ± 0.04	8.62 ± 0.84	0.22 ± 0.07	19.02 ± 0.41
	Qwen2.5-7B-Instruct	0.82 ± 0.07	8.86 ± 0.99	0.23 ± 0.20	17.38 ± 2.74
	Qwen2.5-14B-Instruct	0.88 ± 0.04	10.31 ± 0.71	0.07 ± 0.07	19.71 ± 0.25
	Mistral-7B-Instruct	0.60 ± 0.08	12.04 ± 1.02	0.43 ± 0.16	16.43 ± 1.20
	Single-Turn	0.10 ± 0.02	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00
Landmark	Claude-3.5-Haiku	0.94 ± 0.02	7.44 ± 0.31	0.18 ± 0.07	19.41 ± 0.34
	GPT-4o-mini	0.92 ± 0.06	8.37 ± 0.68	0.11 ± 0.04	19.73 ± 0.17
	Qwen2.5-7B-Instruct	0.78 ± 0.02	9.14 ± 1.66	0.32 ± 0.18	16.14 ± 3.32
	Qwen2.5-14B-Instruct	0.89 ± 0.04	8.24 ± 0.78	0.00 ± 0.00	20.00 ± 0.00
	Mistral-7B-Instruct	0.78 ± 0.06	10.11 ± 1.26	0.26 ± 0.10	18.37 ± 0.35
	Single-Turn	0.07 ± 0.03	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00
Sport	Claude-3.5-Haiku	0.94 ± 0.03	7.53 ± 0.46	0.24 ± 0.06	18.86 ± 0.38
	GPT-4o-mini	0.74 ± 0.14	12.64 ± 1.25	0.24 ± 0.08	18.92 ± 0.25
	Qwen2.5-7B-Instruct	0.78 ± 0.18	9.92 ± 1.61	0.27 ± 0.15	17.57 ± 2.23
	Qwen2.5-14B-Instruct	0.88 ± 0.09	8.48 ± 0.80	0.27 ± 0.12	18.44 ± 0.78
	Mistral-7B-Instruct	0.70 ± 0.11	11.69 ± 1.58	0.07 ± 0.02	19.66 ± 0.23
	Single-Turn	0.07 ± 0.05	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00
Weather	Claude-3.5-Haiku	0.88 ± 0.05	8.93 ± 0.87	0.59 ± 0.07	16.05 ± 0.96
	GPT-4o-mini	0.91 ± 0.06	7.75 ± 0.99	0.50 ± 0.10	17.32 ± 0.38
	Qwen2.5-7B-Instruct	0.66 ± 0.08	11.72 ± 1.75	0.36 ± 0.18	16.32 ± 2.48
	Qwen2.5-14B-Instruct	0.86 ± 0.08	9.22 ± 1.07	0.33 ± 0.11	18.47 ± 0.60
	Mistral-7B-Instruct	0.69 ± 0.10	11.55 ± 1.58	0.16 ± 0.09	18.74 ± 0.77
	Single-Turn	0.06 ± 0.04	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00

Table 7: Results of the “20 Questions Game” experiments across different models, topics, and conditions (topic known vs unknown).

- License: proprietary (accessed through Claude API)
- 3. Mistral-7B-Instruct (Jiang et al., 2023)
 - Link: <https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2>
 - License: Apache license 2.0
- 4. Qwen2.5-7B-Instruct (Yang et al., 2024)
 - Link: <https://huggingface.co/Qwen/Qwen2.5-7B-Instruct>
 - License: Apache license 2.0
- 5. Qwen2.5-14B-Instruct (Yang et al., 2024)
 - Link: <https://huggingface.co/Qwen/Qwen2.5-14B-Instruct>
 - License: Apache license 2.0

The LLM models that were not accessed via APIs were run on NVIDIA A40 and A100 GPUs with 40GB of memory.

C.2 LLM Agent Instructions

Below we provide the instructions given to the three LLM Agents for the Personalized Question Answering Task. For the other two tasks, please refer to the code repository.

We note that expressions of the form “{...}” are substituted at execution time with the appropriate values. Specifically, “{general_instructions}” repeats the “GENERAL INSTRUCTIONS” part, “{info_num_turns}” provides information about the maximum number of turns, “{conversation_history}” provides the complete Assistant-

Topic	Model	# Preferences=1		# Preferences=2		# Preferences=3	
		Success Rate	Stop Turn	Success Rate	Stop Turn	Success Rate	Stop Turn
Medical Care	Claude-3.5-Haiku	0.97 $\pm$ 0.04	2.70 $\pm$ 0.46	0.92 $\pm$ 0.05	3.72 $\pm$ 0.45	0.89 $\pm$ 0.04	4.40 $\pm$ 0.20
	GPT-4o-mini	0.94 $\pm$ 0.05	2.42 $\pm$ 0.52	0.91 $\pm$ 0.03	3.60 $\pm$ 0.24	0.84 $\pm$ 0.05	4.62 $\pm$ 0.38
	Qwen2.5-7B-Instruct	0.91 $\pm$ 0.02	3.14 $\pm$ 0.27	0.87 $\pm$ 0.09	3.97 $\pm$ 0.53	0.80 $\pm$ 0.08	5.40 $\pm$ 0.40
	Qwen2.5-14B-Instruct	0.95 $\pm$ 0.06	2.64 $\pm$ 0.54	0.88 $\pm$ 0.05	3.49 $\pm$ 0.70	0.90 $\pm$ 0.04	4.11 $\pm$ 0.21
	Mistral-7B-Instruct	0.97 $\pm$ 0.04	2.58 $\pm$ 0.38	0.93 $\pm$ 0.05	3.46 $\pm$ 0.51	0.88 $\pm$ 0.05	3.64 $\pm$ 0.19
	Single-Turn	0.52 $\pm$ 0.08	1.00 $\pm$ 0.00	0.33 $\pm$ 0.10	1.00 $\pm$ 0.00	0.23 $\pm$ 0.08	1.00 $\pm$ 0.00
Personal Finance	Claude-3.5-Haiku	0.87 $\pm$ 0.05	4.38 $\pm$ 0.26	0.90 $\pm$ 0.05	5.08 $\pm$ 0.31	0.84 $\pm$ 0.07	5.23 $\pm$ 0.65
	GPT-4o-mini	0.82 $\pm$ 0.05	4.37 $\pm$ 0.31	0.78 $\pm$ 0.07	5.94 $\pm$ 0.56	0.78 $\pm$ 0.09	5.56 $\pm$ 0.75
	Qwen2.5-7B-Instruct	0.84 $\pm$ 0.05	4.37 $\pm$ 0.56	0.78 $\pm$ 0.06	5.74 $\pm$ 0.24	0.72 $\pm$ 0.07	6.10 $\pm$ 0.49
	Qwen2.5-14B-Instruct	0.79 $\pm$ 0.07	4.22 $\pm$ 0.46	0.72 $\pm$ 0.08	5.79 $\pm$ 0.45	0.73 $\pm$ 0.07	6.07 $\pm$ 0.35
	Mistral-7B-Instruct	0.85 $\pm$ 0.06	3.85 $\pm$ 0.51	0.79 $\pm$ 0.04	4.66 $\pm$ 0.47	0.82 $\pm$ 0.07	4.67 $\pm$ 0.34
	Single-Turn	0.34 $\pm$ 0.08	1.00 $\pm$ 0.00	0.16 $\pm$ 0.10	1.00 $\pm$ 0.00	0.15 $\pm$ 0.08	1.00 $\pm$ 0.00
Pet Ownership	Claude-3.5-Haiku	0.87 $\pm$ 0.06	3.81 $\pm$ 0.50	0.77 $\pm$ 0.09	4.55 $\pm$ 0.52	0.75 $\pm$ 0.08	4.97 $\pm$ 0.18
	GPT-4o-mini	0.78 $\pm$ 0.06	4.66 $\pm$ 0.29	0.69 $\pm$ 0.04	5.79 $\pm$ 0.19	0.68 $\pm$ 0.06	5.23 $\pm$ 0.44
	Qwen2.5-7B-Instruct	0.77 $\pm$ 0.07	5.04 $\pm$ 0.56	0.73 $\pm$ 0.06	5.25 $\pm$ 0.55	0.69 $\pm$ 0.06	5.73 $\pm$ 0.26
	Qwen2.5-14B-Instruct	0.76 $\pm$ 0.12	5.03 $\pm$ 0.50	0.68 $\pm$ 0.05	5.48 $\pm$ 0.35	0.63 $\pm$ 0.06	6.29 $\pm$ 0.64
	Mistral-7B-Instruct	0.83 $\pm$ 0.06	4.08 $\pm$ 0.39	0.88 $\pm$ 0.07	4.40 $\pm$ 0.32	0.82 $\pm$ 0.09	4.58 $\pm$ 0.35
	Single-Turn	0.35 $\pm$ 0.08	1.00 $\pm$ 0.00	0.28 $\pm$ 0.09	1.00 $\pm$ 0.00	0.23 $\pm$ 0.08	1.00 $\pm$ 0.00
Shopping Assistance	Claude-3.5-Haiku	0.79 $\pm$ 0.10	4.94 $\pm$ 0.56	0.53 $\pm$ 0.19	6.93 $\pm$ 0.87	0.54 $\pm$ 0.12	7.20 $\pm$ 0.77
	GPT-4o-mini	0.74 $\pm$ 0.07	5.25 $\pm$ 0.66	0.52 $\pm$ 0.02	6.92 $\pm$ 0.58	0.33 $\pm$ 0.04	8.21 $\pm$ 0.31
	Qwen2.5-7B-Instruct	0.66 $\pm$ 0.15	6.05 $\pm$ 0.86	0.45 $\pm$ 0.12	7.91 $\pm$ 0.58	0.28 $\pm$ 0.14	8.59 $\pm$ 0.48
	Qwen2.5-14B-Instruct	0.69 $\pm$ 0.11	5.68 $\pm$ 0.87	0.44 $\pm$ 0.17	8.00 $\pm$ 0.93	0.35 $\pm$ 0.13	8.40 $\pm$ 0.51
	Mistral-7B-Instruct	0.78 $\pm$ 0.09	5.10 $\pm$ 0.57	0.63 $\pm$ 0.12	6.46 $\pm$ 0.69	0.48 $\pm$ 0.17	7.49 $\pm$ 0.86
	Single-Turn	0.23 $\pm$ 0.06	1.00 $\pm$ 0.00	0.09 $\pm$ 0.05	1.00 $\pm$ 0.00	0.08 $\pm$ 0.04	1.00 $\pm$ 0.00
Travel Restaurant	Claude-3.5-Haiku	0.94 $\pm$ 0.02	3.17 $\pm$ 0.32	0.82 $\pm$ 0.10	4.46 $\pm$ 1.01	0.75 $\pm$ 0.14	4.81 $\pm$ 1.44
	GPT-4o-mini	0.91 $\pm$ 0.05	2.66 $\pm$ 0.53	0.61 $\pm$ 0.12	5.90 $\pm$ 0.72	0.45 $\pm$ 0.06	7.03 $\pm$ 0.42
	Qwen2.5-7B-Instruct	0.88 $\pm$ 0.07	3.32 $\pm$ 0.47	0.57 $\pm$ 0.13	6.71 $\pm$ 0.63	0.42 $\pm$ 0.15	7.85 $\pm$ 0.88
	Qwen2.5-14B-Instruct	0.88 $\pm$ 0.05	3.90 $\pm$ 0.61	0.57 $\pm$ 0.07	6.37 $\pm$ 0.25	0.41 $\pm$ 0.17	7.88 $\pm$ 0.99
	Mistral-7B-Instruct	0.82 $\pm$ 0.08	4.06 $\pm$ 0.50	0.57 $\pm$ 0.13	6.58 $\pm$ 0.51	0.49 $\pm$ 0.12	7.46 $\pm$ 0.93
	Single-Turn	0.45 $\pm$ 0.11	1.00 $\pm$ 0.00	0.16 $\pm$ 0.009	1.00 $\pm$ 0.00	0.08 $\pm$ 0.08	1.00 $\pm$ 0.00

Table 8: Results of the Personalized Question Answering experiments across different models, topics, and preference settings.

Topic	Model	# Preferences=2		# Preferences=3	
		Success Rate	Stop Turn	Success Rate	Stop Turn
Big Patent	Claude-3.5-Haiku	0.73 $\pm$ 0.02	5.19 $\pm$ 0.42	0.72 $\pm$ 0.07	5.39 $\pm$ 0.49
	GPT-4o-mini	0.70 $\pm$ 0.08	5.43 $\pm$ 0.49	0.56 $\pm$ 0.08	6.48 $\pm$ 0.48
	Qwen2.5-7B-Instruct	0.63 $\pm$ 0.09	5.58 $\pm$ 0.53	0.57 $\pm$ 0.09	6.50 $\pm$ 0.64
	Mistral-7B-Instruct	0.59 $\pm$ 0.03	6.18 $\pm$ 0.27	0.40 $\pm$ 0.07	7.84 $\pm$ 0.48
	Single-Turn	0.24 $\pm$ 0.08	1.00 $\pm$ 0.00	0.15 $\pm$ 0.10	1.00 $\pm$ 0.00
CNN DailyMail	Claude-3.5-Haiku	0.85 $\pm$ 0.04	3.87 $\pm$ 0.42	0.78 $\pm$ 0.06	4.46 $\pm$ 0.31
	GPT-4o-mini	0.86 $\pm$ 0.02	3.48 $\pm$ 0.38	0.79 $\pm$ 0.09	4.44 $\pm$ 0.35
	Qwen2.5-7B-Instruct	0.83 $\pm$ 0.07	4.42 $\pm$ 0.59	0.77 $\pm$ 0.04	5.31 $\pm$ 0.54
	Mistral-7B-Instruct	0.76 $\pm$ 0.06	4.83 $\pm$ 0.60	0.62 $\pm$ 0.08	5.87 $\pm$ 0.57
	Single-Turn	0.32 $\pm$ 0.10	1.00 $\pm$ 0.00	0.21 $\pm$ 0.13	1.00 $\pm$ 0.00
Elsevier PHYS	Claude-3.5-Haiku	0.87 $\pm$ 0.02	3.52 $\pm$ 0.46	0.82 $\pm$ 0.05	4.71 $\pm$ 0.28
	GPT-4o-mini	0.80 $\pm$ 0.08	4.31 $\pm$ 0.35	0.80 $\pm$ 0.03	5.10 $\pm$ 0.24
	Qwen2.5-7B-Instruct	0.79 $\pm$ 0.05	4.27 $\pm$ 0.70	0.74 $\pm$ 0.03	5.06 $\pm$ 0.64
	Mistral-7B-Instruct	0.72 $\pm$ 0.07	5.31 $\pm$ 0.52	0.51 $\pm$ 0.09	6.86 $\pm$ 0.50
	Single-Turn	0.31 $\pm$ 0.13	1.00 $\pm$ 0.00	0.21 $\pm$ 0.09	1.00 $\pm$ 0.00
Wikipedia	Claude-3.5-Haiku	0.78 $\pm$ 0.05	4.16 $\pm$ 0.40	0.63 $\pm$ 0.11	5.77 $\pm$ 0.54
	GPT-4o-mini	0.79 $\pm$ 0.05	4.75 $\pm$ 0.36	0.61 $\pm$ 0.03	5.96 $\pm$ 0.28
	Qwen2.5-7B-Instruct	0.78 $\pm$ 0.06	4.72 $\pm$ 0.63	0.59 $\pm$ 0.03	6.39 $\pm$ 0.36
	Mistral-7B-Instruct	0.76 $\pm$ 0.09	5.29 $\pm$ 0.67	0.53 $\pm$ 0.06	6.81 $\pm$ 0.50
	Single-Turn	0.21 $\pm$ 0.11	1.00 $\pm$ 0.00	0.14 $\pm$ 0.06	1.00 $\pm$ 0.00

Table 9: Results of the Text Summarization experiments across different models, topics, and preference settings.

User interaction up to the current turn, “{user_question}” is the User’s question, and “{assistant_answer}” is the Assistant’s answer to the User’s question.

C.2.1 Personalized Question Answering

Assistant

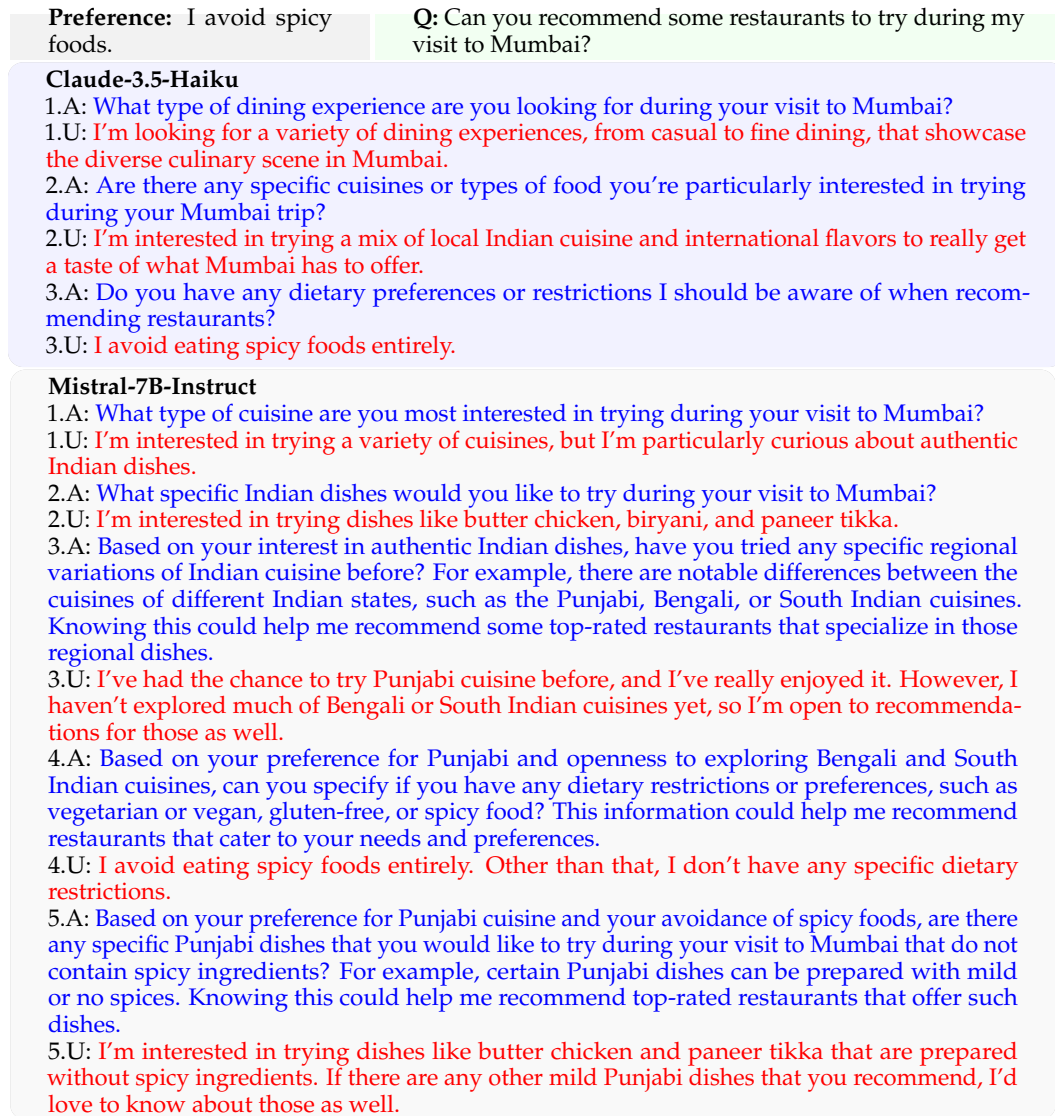


Figure 11: An instance of the Personalized Question Answering task for the "Travel Restaurant topic", along with the conversations of two models.

GENERAL_INSTRUCTIONS:

- You are an assistant participating in a multi-turn conversation with a user. The user has asked you a single question. The question is the following: "{user_question}".
- Your goal is to answer the user's question. Your answer must be a personalized one that is, it should be tailored to the user's preferences. To achieve this, you'll need to uncover the user's hidden preferences (they may have more than one) that are relevant to the question, by engaging in a multi-turn conversation. You do NOT know any of the user's hidden preferences in advance.
- Your interaction with the user works as follows. At each turn of the conversation you ask the user a question and the user responds. Do NOT simulate the user or the conversation. Just output the question.
- At each turn, provide a single, focused, and relevant question or statement do not ask multiple questions or ramble. Do not guess outside the scope of the question.
- {info_num_turns}

- Do NOT directly guess or enumerate possible preferences in your first questions. Start with open-ended, general questions to learn more about the user's relevant interests. As the conversation progresses, ask more specific questions to narrow down the preferences.

FIRST_TURN:

General Instructions:

{general_instructions}

This is the first turn, start the conversation by asking your first question. It is important to output only the question, nothing else. Your question:

SUBSEQUENT_TURNS:

General Instructions:

{general_instructions}

This is the conversation history up to this point:

{conversation_history}

This is turn {turn_number} of the conversation. Ask your next question. It is important to output only the question, nothing else.

Your question:

FINAL_PROMPT:

General Instructions:

{general_instructions}

The complete conversation history:

{conversation_history}

Based on the complete conversation provide an answer to the following question: "{user_question}".

Do not ask any more questions.

Your final answer:

User

GENERAL_INSTRUCTIONS:

- You are a user with several hidden preferences and a question whose answer depends on those preferences.
- Your list of hidden preferences is the following: "{user_preference}"
- Your question is the following: "{user_question}"
- You participate in a multi-turn conversation with an assistant which asks you questions. Do not reveal your hidden preferences unless asked a relevant question by the assistant. However, you have to answer all of the assistant's questions truthfully.
- Always provide a natural and relevant answer to each question. Never leave your answer blank. Respond as a human would in a real conversation.

PROMPT:

General Instructions:

{general_instructions}

This is the conversation history up to this point (can be empty if this is the first turn):

{conversation_history}

In the current turn the assistant asked the following question: "{assistant_question}"

Provide your answer to the question of the current turn below. Do not output anything else.

Your answer:

Judge

GENERAL_INSTRUCTIONS:

- You are a judge. You are given as input the following: 1) a user's hidden preferences; 2) a user's question; 3) an assistant's answer to the user's question. Decide whether the assistant's answer correctly responds to the user's question. Respond with 'CORRECT' or 'INCORRECT' and a brief explanation.
- A correct answer is one that aligns with ALL of the user's hidden preferences that is, it considers every preference.
- If the assistant's answer contradicts or ignores ANY of the hidden preferences, treat the answer as 'INCORRECT', even if the rest of the user's preferences are taken into account.
- If the assistant's answer is compatible with additional preferences (i.e. preferences not present on the list of user's hidden preferences) then these should not affect the judge's assessment assuming that they do not contradict with the hidden ones.

PROMPT:

General Instructions:
{general_instructions}

User's question
"{user_question}"

List of user's hidden preferences:
{user_preference}

Assistant's answer:
"{assistant_answer}"

Judge:

D Datasets

D.1 Data Sources and Generation

Details about the sources and generation process of the used data are provided below.

D.1.1 20 Questions Game

The object data were generated using LLMs. Specifically, we provided GPT-4.1 with a set of five topics and requested it to generate twenty distinct objects per topic.

D.1.2 Personalized Question Answering

For the Personalized Question Answering we used the data generation process and topics provided in Zhao et al. (2025a). We selected five topics and used their code to generate ten question-preference pairs per topic. The code we used was released under a Attribution-NonCommercial 4.0 International license.

D.1.3 Personalized Text Summarization

In personalized text summarization we used text from four different text datasets. These are the following:

- News articles: *ccdv/cnn_dailymail*
 - Link: https://huggingface.co/datasets/ccdv/cnn_dailymail
 - License: Apache license 2.0
- General knowledge articles: *wikimedia/wikipedia*

- Link: <https://huggingface.co/datasets/wikimedia/wikipedia>
- License: Creative Commons Attribution Share Alike 3.0
- Science papers: *heegyu/elsevier-oa-cc-by*
 - Link: <https://huggingface.co/datasets/heegyu/elsevier-oa-cc-by>
 - License: unspecified
- Patent documents: *satpalsr/bigpatent-test*
 - Link: <https://huggingface.co/datasets/satpalsr/bigpatent-test>
 - License: unspecified

D.2 Sample Dataset Entries

Below are sample entries from the datasets for the 20 Questions Game, the Personalized Question Answering and the Personalized Text Summarization tasks.

D.2.1 20 Questions Game

Topic : Animal

1. Cat
2. Dog
3. Eagle
4. Elephant
5. Shark
6. Frog
7. Bat
8. Dolphin
9. Penguin
10. Kangaroo

Topic : Landmark

1. Eiffel Tower
2. Statue of Liberty
3. Great Wall of China
4. Pyramids of Giza
5. Taj Mahal
6. Colosseum
7. Machu Picchu
8. Big Ben
9. Sydney Opera House
10. Mount Rushmore

D.2.2 Personalized Question Answering

Topic : Pet Ownership

```
1)
-preferences:
  -- I prefer hypoallergenic dog breeds due to allergies.
  -- I avoid adopting pets from breeders and only choose shelters or
    rescues.
  -- I don't want a pet that requires professional grooming services.
-question: What are some dog breeds you would recommend for me to adopt?
2)
-preferences:
  -- I don't like pets that bark or make loud noises.
  -- I prefer adopting older pets over young puppies or kittens.
  -- I avoid animals that require high levels of activity or exercise.
-question: What kind of pet would be best suited for my lifestyle?
3)
```

```
-preferences:
  -- I prefer pets that do not eat commercial kibble or canned food.
  -- I strongly dislike reptiles and amphibians."
  -- I avoid pets that are not allowed in apartments with pet
    restrictions.
-question: Could you suggest a few pet options suitable for apartment
  living for me?
```

Topic : Shopping Assistance

```
1)
-preferences
  -- I strictly avoid fast fashion brands due to ethical concerns.
  -- I dislike products that are heavily fragranced.
  -- I only buy clothing made from natural fibers like cotton or wool.
-question: Can you help me find some trendy clothes to refresh my
  wardrobe this summer?
2)
-preference:
  -- I strongly dislike large department stores and prefer small,
    specialty shops.
  -- I avoid products with animal testing.
  -- I only buy certified organic skincare products.
-question": Can you recommend some good skincare products for me?
3)
-preference:
  -- I only buy electronics with repairable designs.
  -- I completely avoid brands that require subscription services for
    basic functions.
  -- I do not use any product that doesn't have at least a 3-year
    warranty.
-question: What are the best tablets or laptops available in the market
  today?
```

D.2.3 Personalized Text Summarization

Topic : CNN-DailyMail

```
-preferences:
  -- Focus primarily on Andrew Getty's personal health struggles and
    medical condition rather than the family's wealth and business
    history.
  -- Minimize details about the death scene investigation and police
    findings in favor of emphasizing the family's response and
    request for privacy.
  -- Prioritize information about the recent restraining order case and
    legal troubles over the Getty family's oil fortune background.
-text: "(CNN)Andrew Getty, one of the heirs to billions of oil money,
  appears to have died of natural causes, a Los Angeles Police
  Department spokesman said. The coroner's preliminary assessment is
  there was no foul play involved in the death of Getty, grandson of
  oil tycoon J. Paul Getty, said Detective Meghan Aguilar. Andrew Getty
  , 47, had "several health issues," Aguilar said, adding that an
  autopsy will be conducted. There is no criminal investigation
  underway, he said. Some medication had also been recovered from Getty
  's home, though investigators don't know whether Getty was taking it
  or what his medical history was, Ed Winter, assistant chief in the
  Los Angeles County coroner's office, told CNN affiliate KTLA Tuesday
  night. KTLA reported that Getty was found on his side near a bathroom
  in his home. Getty's parents, Ann and Gordon Getty, released a
  statement confirming their son's death and asking for privacy. Where
  the Getty family fortune came from . Gordon Getty is one of three
  living sons of J. Paul Getty, the oil baron who was thought to be the
  richest man in the world at the time of his death in 1976. Gordon
```

Getty, 81, has a net worth of \$2.1 billion, according to Forbes. One other son died in 1958 and another died in 1973. Gordon Getty spearheaded the controversial sale of Getty to Texaco for \$10 billion in 1984. In its list of richest American families, Forbes estimated the Gettys' net worth to be about \$5 billion. Court records show Andrew Getty had recently filed to get a restraining order against an ex-girlfriend. A hearing in the case had been scheduled for next week. In his request, Getty said he had been diagnosed with a serious medical condition in 2013. "A rise in my blood pressure places me in grave risk of substantial and irreparable injury or death," he wrote in the petition. "My doctors have advised that heated arguments can cause my blood pressure to rise dangerously." Andrew Getty had three brothers and three half-sisters. People we've lost in 2015 . CNN's Doug Criss, Janet DiGiacomo, Mark Mooney, Mike Love, Julie In and Cheri Mossburg contributed to this report."