Toward Multi-Session Personalized Conversation: A Large-Scale Dataset and Hierarchical Tree Framework for Implicit Reasoning

Xintong Li, Jalend Bantupalli, Ria Dharmani, Yuwei Zhang, Jingbo Shang
University of California, San Diego
{xil240, jbantupalli, rdharmani, yuz163, jshang}@ucsd.edu

Abstract

There has been a surge in the use of large language models (LLM) conversational agents to generate responses based on long-term history from multiple sessions. However, existing long-term open-domain dialogue datasets lack complex, real-world personalization and fail to capture implicit reasoning—where relevant information is embedded in subtle, syntactic, or semantically distant connections rather than explicit statements. In such cases, traditional retrieval methods fail to capture relevant context, and long-context modeling also becomes inefficient due to numerous complicated persona-related details. To address this gap, we introduce IMPLEXCONV, a large-scale long-term dataset with 2,500 examples, each containing approximately 100 conversation sessions, designed to study implicit reasoning in personalized dialogues. Additionally, we propose TACITREE, a novel hierarchical tree framework that structures conversation history into multiple levels of summarization. Instead of brute-force searching all data, TACITREE enables an efficient, level-based retrieval process where models refine their search by progressively selecting relevant details. Our experiments demonstrate that TACITREE significantly improves the ability of LLMs to reason over long-term conversations with implicit contextual dependencies.

1 Introduction

Large language models (LLMs) have revolutionized conversational AI by enabling personalized and context-aware dialogue generation (Achiam et al., 2023; McTear, 2022). Recent advances allow LLM-based agents to recall and integrate a long-term conversational history across multiple sessions, significantly enhancing coherence and personalization (Zhong et al., 2024; Li et al., 2024; Wang et al., 2023). In this paper, we focus on *implicit reasoning*, arguably the most challenging conversational setting, where relevant information is embedded in subtle syntactic patterns or semantically distant connections rather than explicitly stated, as demonstrated in Figure 1.

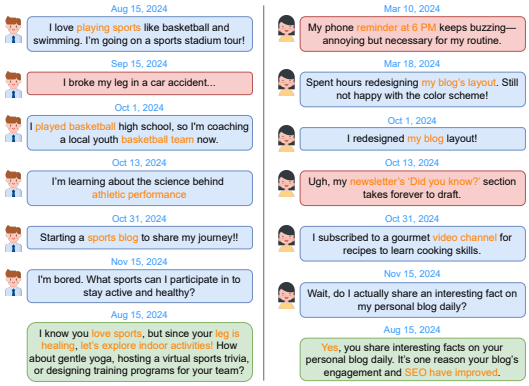

Figure 1: An example from IMPLEXCONV illustrating *opposed* (left) and *supportive* (right) implicit reasoning. The orange block is the user query, the red blocks are implicit scenarios with low semantic similarity to the query, and the blue blocks are noisy but lexically related conversations that obscure the correct response.

As shown in Table 1, none of the existing datasets incorporates implicit reasoning scenarios — Large-scale datasets (Jang et al., 2023) lack session depth, while deep but small datasets (Wu et al., 2024) lack structured personas critical for conversational consistency. To bridge this gap, we construct IMPLEXCONV, a large-scale dataset with 2,500 multi-session examples (∼100 sessions each) and 600 thousand persona traits designed to maintain session coherence. IMPLEXCONV uniquely introduces implicit reasoning scenarios, where carefully curated questions with verifiable answers reveal persona traits that subtly reinforce or oppose other personalization details, while maintaining low semantic similarity that makes such reasoning difficult to trace.

Implicit reasoning is particularly challenging to existing retrieval (Shuster et al., 2021; Fan et al., 2024) and long-context modeling (Xiong et al.,

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 11505–11518
November 4-9, 2025

Dataset	# of Conv. / # Sessions	Avg. Turns per Conv.	Avg. Tokens	Implicit Reason
Daily Dialog (Li et al., 2017)	13K / 13K	7.9	114.7	✗
PersonaChat (Zhang, 2018)	10K / 10K	14.8	245.2	✗
MSC (Xu, 2021)	4K / 12K	53.3	1,225.9	✗
CC (Jang et al., 2023)	200K / 1M	58.5	1,054.7	✗
LoCoMo (Maharana et al., 2024)	10 / 1K	304.9	9,209.2	✗
PerLTQA (Du et al., 2024)	1 / 4K	15K	1M	✗
LongMemEval (Wu et al., 2024)	500 / 50K	5K	115K	✗
ImplexConv (Ours)	2500 / 255K	2K	60K	✓

Table 1: Comparison of ImplexConv with existing datasets, highlighting its large-scale multi-session structure and unique focus on implicit reasoning.

2023; Xu et al., 2023) techniques because it requires models to move beyond surface-level pattern recognition toward deeper reasoning over long-term interactions. As dialogue history accumulates, numerous persona-related details can obscure critical implicit knowledge, making it increasingly difficult for long-context modeling techniques to extract and utilize relevant information effectively. The presence of excessive persona details often leads to retrieval inefficiencies, where dominant but less relevant traits overshadow essential implicit patterns, resulting in inconsistencies in generated responses.

We propose TaciTree, a novel framework designed to address the inefficiency of retrieving implicit knowledge in long-term conversations, as shown in Figure 3. While LLMs can inherently assess whether an implicit scenario relates to a query, brute-force retrieval (Lin, 2009) that inspects every individual fact can achieve a high recall, however, it would suffer from prohibitive computational costs. TaciTree overcomes this by structuring conversational history into a hierarchical tree, where lower-level nodes capture fine-grained details and higher-level nodes aggregate these into abstract summaries. By grouping relevant information into subtrees, our framework enables subtree skipping by evaluating high-level summaries — only when a summary is relevant does the model drill down into finer-grained details. This hierarchical approach reduces the search space by orders of magnitude compared to brute-force retrieval while retaining high accuracy, as LLMs leverage their inherent reasoning ability to navigate the tree.

We evaluate ImplexConv and TaciTree via question-answering tasks. ImplexConv exhibits 20% lower semantic similarity between queries and ground-truth answers compared to existing datasets, reflecting its unique challenge of high implicitness. TaciTree achieves 30% higher retrieval accuracy than baselines (e.g., RAG, MemoryBank), which struggle with implicit reasoning unless retrieving excessive amounts of information. Notably, TaciTree achieves this with 40–60% fewer tokens, demonstrating efficient extraction of implicit knowledge without sacrificing precision.

Our contributions are summarized as:

- We introduce ImplexConv, a large-scale multi-session dialogue dataset specifically designed to evaluate implicit reasoning in long-term personalized conversations.
- We propose TaciTree, a hierarchical tree-based framework that efficiently stores and retrieves long-term conversational history, enabling models to extract implicit knowledge with level-based retrieval.
- Experimental results demonstrate the high implicitness of our dataset and the significantly improved retrieval accuracy of our framework, achieved with a smaller retrieval token size.

Our dataset and source code can be obtained here [1].

2 Related Work

Long-term conversational AI research spans both dataset construction and memory-enhanced methodologies. Existing multi-session dialogue datasets primarily focus on continuity, personalization, or memory retention, but they lack the necessary complexity for implicit reasoning. While datasets such as MSC (Xu, 2021) and LoCoMo (Maharana et al., 2024) incorporate structured long-term interactions, they do not explicitly model implicit reasoning. Similarly, methodologies for long-term memory, including structured memory mechanisms (Zhong et al., 2024) and RAG frameworks (Lewis et al., 2020), aim to improve historical context utilization but struggle with implicit dependencies Additional discussions on related datasets and methodologies are provided in Appendix A.

3 ImplexConv Collection

We introduce ImplexConv, a large-scale dataset designed to evaluate implicit reasoning in long-term multi-session conversations as shown in Figure 2. It comprises 2,500 examples, each containing approximately 100 dialogue sessions. Unlike existing datasets, ImplexConv includes carefully constructed implicit reasoning scenarios—both opposed and supportive—that require retrieval of subtle and semantically distant connections embedded in extensive dialogue histories rather than explicit

[1] https://github.com/Kaylee0501/ImplexConv

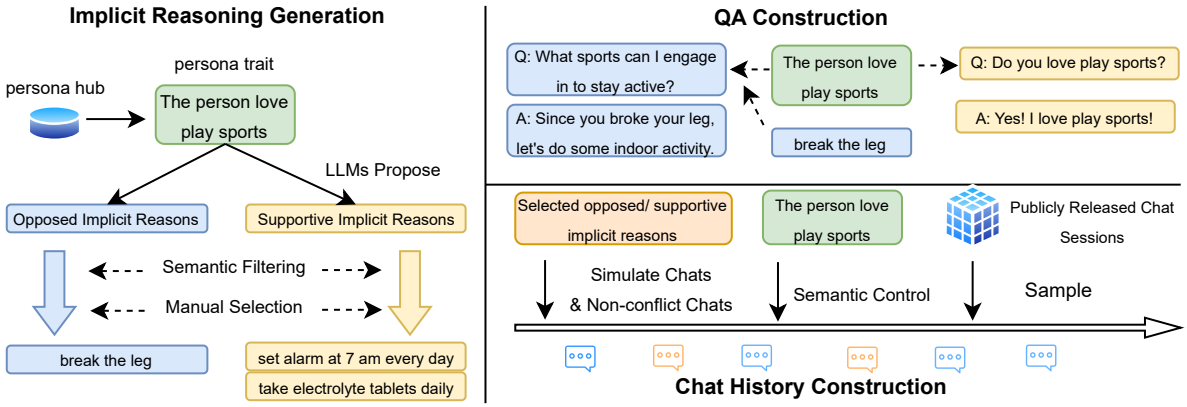

Figure 2: Overvew of IMPLEXCONV construction. Implicit reasoning is generated from persona traits, followed by QA creation and multi-session chat history construction using simulated and real-world dialogues.

statements. These properties make IMPLEXCONV a challenging benchmark for evaluating retrieval-based and long-context models. Our dataset construction consists of persona extraction, implicit reasoning generation, and multi-turn conversation formulation to ensure realism and diversity.

3.1 Persona Extraction

We begin by extracting a diverse set of personas $\mathcal{P}$ from Persona Hub (Ge et al., 2024). Each persona is described by single-sentence attributes detailing demographics, careers, personal goals, or daily activities (see Appendix B for examples). To ensure consistency for long-term coherence, we standardize the originally free-form persona descriptions into structured statements explicitly starting with "This person..." using an instruction-tuned language model (LLM) M_1 as shown in Code 1.

3.2 Implicit Reasoning

Implicit reasoning constitutes the core challenge of IMPLEXCONV, requiring models to retrieve information indirectly inferred from subtle conversational cues rather than direct mentions. For each persona trait $p \in \mathcal{P}$, we generate 20 opposed and 20 supportive implicit reasoning scenarios via LLM M_1. As illustrated in Figure 1, opposed reasoning scenarios R_o present situations that implicitly conflict with persona traits (e.g., persona: "This person enjoys sports"; opposed scenario: "I recently broke my leg"). Supportive scenarios R_s subtly reinforce these traits (e.g., persona: "This person shares facts on a personal blog"; supportive scenario: "Drafting my newsletter takes significant time"). To ensure subtlety and quality, we filter scenarios using an instruction-tuned embedding model E. We compute semantic similarities between generated scenarios and the original persona trait, retaining only those with similarity scores below a threshold β. Cases with borderline similarity scores undergo human verification, resulting in refined sets R'_o and R'_s.

3.2.1 Opposed Reasoning Scenarios

From the filtered set R'_o, we prompt M_1 to select the most implicitly opposed scenario R^*_o with Code 5. If the model outputs multiple or ambiguous candidates, human annotators manually determine the best scenario, following standardized instructions. The detailed human evaluation process is shown in Appendix C. We then formulate a corresponding question-answer task: a general daily-life question q_o is generated such that, without R^*_o, the answer a_o would naturally align with the original trait p. However, due to the implicit scenario R^*_o, a_o must reflect this changed circumstance, ensuring precise evaluation.

3.2.2 Supportive Reasoning Scenarios

Unlike opposed implicit reasoning, which presents a more challenging inference task, supportive implicit reasoning is designed to be comparatively easier to test. Supportive reasoning scenarios from R'_s are verified again using M_1, instructed explicitly to identify alignment confidently. Uncertain cases undergo additional human review similar to opposed reasoning scenarios. The verified multiple scenarios R^*_s directly inform straightforward yes/no questions designed around the original trait, ensuring clarity and ease of evaluation. The expected answer depends on whether the implicit reasoning supports the original persona trait.

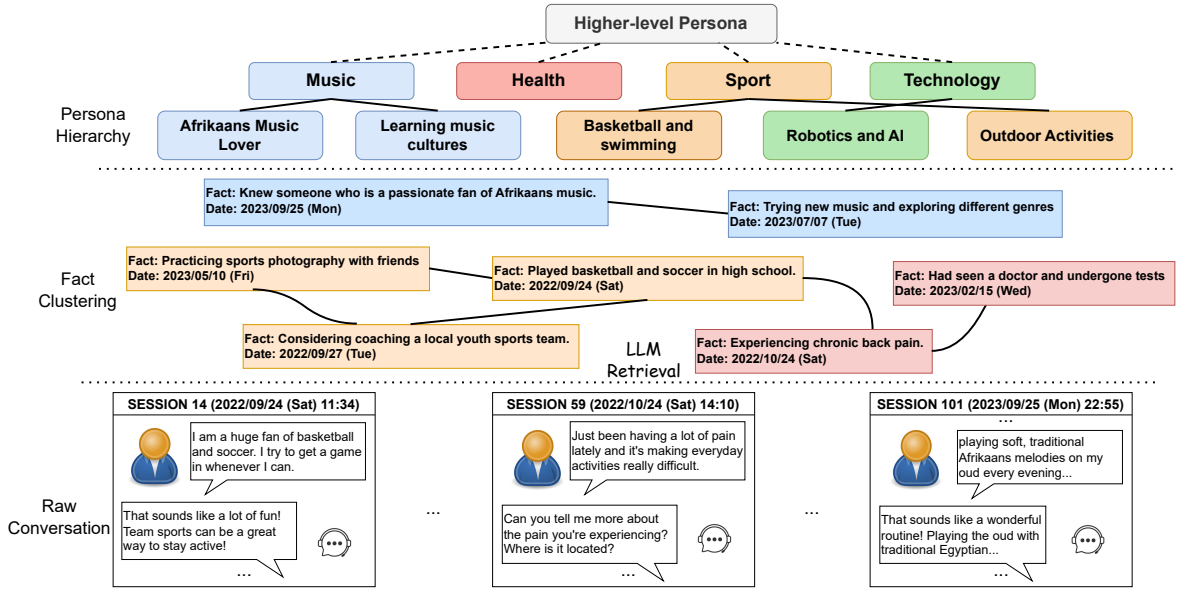

Figure 3: Overview of TACITREE framework. TACITREE organizes long-term conversational history into a hierarchical structure, clustering related facts to enable efficient retrieval of implicit reasoning. By leveraging LLMs to refine relevant information while discarding unrelated details, the framework reduces search space and improves retrieval efficiency.

3.3 Conversation Formulation

The final construction phase generates realistic multi-turn conversations based on previously selected implicit reasoning scenarios. For opposed reasoning scenarios, we increase evaluation difficulty by generating five additional scenarios with higher semantic similarity to the target question q_o than R_o^*, creating distracting but incorrect context. Each scenario, including the original trait p and the selected opposed reasoning R_o^*, is expanded into separate dialogue sessions simulating human-chat assistant interactions. Sessions are assigned timestamps ensuring temporal coherence, explicitly preventing an immediate sequence of R_o^* after p. Supportive reasoning scenarios similarly form the basis for multiple dialogue sessions.

To further increase realism and challenge, we introduce additional noisy sessions sourced from publicly available conversational datasets (CC (Jang et al., 2023), LLM-Redial (Liang et al., 2024), and UltraChat (Ding et al., 2023)). These real-world dialogues, semantically related but insufficient for correct implicit reasoning, act as challenging noise. Since CC contains human-human dialogues, we convert them into a human-assistant format to ensure consistency. All noisy sessions are randomly interleaved with implicit scenarios, enhancing realism. These inserted sessions also undergo semantic similarity checks and human verification to exclude unintentionally supportive noise, maintaining evaluation rigor. Each persona trait thus associates with over 15 dialogue sessions encompassing both targeted implicit scenarios and challenging noisy contexts. The final IMPLEXCONV instances are created by randomly sampling and merging dialogue sessions from multiple personas, containing approximately 100 sessions per example.

3.4 Implicitness of Datasets

To evaluate the implicit nature of IMPLEXCONV, we measure the semantic similarity between the query and its corresponding answer across different datasets. A higher semantic distance indicates that the answer is less explicitly stated in the conversation history, making retrieval and reasoning more challenging. We define **Implicitness Score (IS)** equals to $1 - \text{Sim}(Q, A)$, where Q is the query, A is the ground truth answer, and $\text{Sim}(\cdot)$ represents cosine similarity over sentence embeddings. Figure 4 presents the distribution of IS across multiple datasets. Traditional multi-session datasets, such as MSC and CC, exhibit relatively low implicitness scores, averaging around 37%–0.38%, indicating that their target information can often be retrieved through direct semantic similarity. LoCoMo and LongMemEval display slightly higher scores, suggesting a moderate increase in reasoning complexity but still relying on explicit contextual

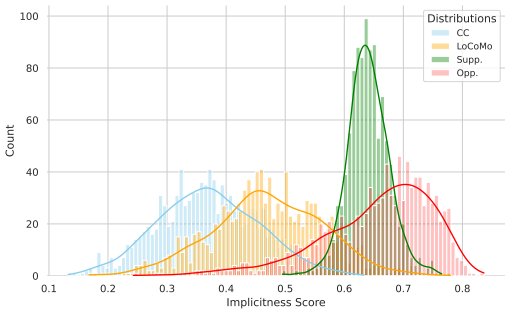

Figure 4: Distribution of implicitness scores across datasets, where Supp. and Opp. represent the supportive and opposed cases of IMPLEXCONV, respectively.

cues. In contrast, IMPLEXCONV demonstrates significantly higher IS scores, with supportive and opposed reasoning scenarios averaging 64% and 65%, respectively. These findings highlight IMPLEXCONV as a more demanding benchmark for evaluating retrieval-based models, requiring deeper inference beyond surface-level techniques.

4 Framework

We introduce TACITREE, a hierarchical retrieval framework explicitly designed for efficiently extracting implicit reasoning from long-term conversational histories. Unlike conventional retrieval methods that rely solely on direct semantic similarity checks, TACITREE organizes historical conversational facts into a hierarchical tree structure, mirroring human cognitive strategies. Our key insight is that while LLMs can identify implicit relevance, querying each fact individually is computationally infeasible. Instead, TACITREE hierarchically clusters and summarizes conversational history at multiple abstraction levels, enabling efficient navigation of relevant nodes and effective pruning of irrelevant branches. This structured retrieval not only significantly reduces search time but also enhances interpretability by clearly tracing decision pathways, closely aligning with intuitive human reasoning processes. Figure 3 illustrates our approach, with details provided in the subsequent sections.

4.1 Fact Extraction and Initial Clustering

Given a conversation session c from the long-term history, we prompt an LLM M_2 to extract all facts that capture long-term conversational context, following the strategy used in LONGMEMEVAL (Wu et al., 2024). These extracted facts typically form a comprehensive but redundant fact set $\mathcal{F}$. To enhance retrieval efficiency and maintain interpretability, we cluster semantically related facts into coherent groups. We use the embedding model described in Section 3 to encode these facts into dense vector representations, subsequently applying UMAP (McInnes et al., 2018) for dimensionality reduction. A Gaussian Mixture Model (GMM) (Reynolds et al., 2009) then clusters the reduced embeddings, effectively capturing complex semantic relationships. Each cluster's size is constrained to a maximum threshold k, resulting in an initial set of clusters,

$$H^0 = \left\lfloor \frac{|\mathcal{F}|}{k} \right\rfloor, \tag{1}$$

where H^0 denotes the total clusters at the base level. Each cluster is then summarized by M_2, yielding concise yet comprehensive representations $\{s_i^0\}_{i=1}^{H^0}$, which constitute the leaf nodes of our hierarchical retrieval structure.

4.2 Hierarchical Tree Construction

Using leaf-node summaries $\{s_i^0\}$, we iteratively construct the hierarchical tree by further clustering and summarizing nodes from lower levels. At each hierarchical level j, nodes from the previous level are clustered and summarized into progressively more abstract summaries,

$$H^j = \left\lfloor \frac{H^{j-1}}{k} \right\rfloor. \tag{2}$$

Each summary s_i^j thus offers an increasingly general abstraction of underlying conversational contexts. The iterative construction continues until reaching the top-level abstraction size L, enabling intuitive, top-down navigation and interpretation. This hierarchical structuring significantly enhances interpretability, allowing clear visualization of how high-level abstract nodes connect down to detailed, specific facts. Critical details are preserved at lower-level nodes, ensuring no essential information loss while providing high-level navigational efficiency.

4.3 Information Retrieval

TACITREE employs its hierarchical structure to facilitate efficient retrieval of implicit reasoning. Given a query q, retrieval initiates at the highest hierarchical summaries and progressively refines its search downward. At each level, instead of performing similarity comparisons, we prompt M_2 to selectively identify relevant clusters:

$$S_q^j = s_{i'}^j \mid M_2(q, s_i^j) \text{ is relevant.} \tag{3}$$

This targeted strategy drastically reduces computational complexity by pruning irrelevant subtrees early in the retrieval process, significantly accelerating response times. Relevant summaries identified at each level are recursively refined downwards until leaf-level summaries S_q^0 are retrieved. Each leaf-level summary directly connects to its original, detailed fact cluster, ensuring complete access to the critical information necessary for accurate implicit reasoning. Unlike brute-force approaches that inspect every individual fact, TACITREE groups and filters information in a structured manner, significantly improving retrieval accuracy while maintaining efficiency.

5 Experiments

This section presents our experimental setup, including the datasets, baseline methods, evaluation metrics, and implementation details. We evaluate models on multi-session conversations using retrieval accuracy, answer correctness, and token efficiency, ensuring a rigorous comparison across different approaches.

5.1 Experimental Settings

Datasets. We conduct experiments on five benchmark datasets to evaluate how well models handle long-term history in multi-session conversations. These datasets include **MSC**, **CC**, **LoCoMo**,**LongMemEval**, and our proposed dataset, **IMPLEXCONV**. Table 1 provides an overview of each dataset's information. The number of sessions per conversation ranges from 5 to 500, enabling a comprehensive evaluation of the models' ability to store and retrieve relevant information across different conversation lengths.

Compared Methods. We compare our TACITREE framework with three types of baselines: memory-based methods, RAG approaches, and long-context models. For memory-based methods, we use **MemoryBank** (Zhong et al., 2024), which is designed to store and retrieve long-term persona information. MemoryBank simulates human memory retention by dynamically updating stored information over time. RAG-based approaches retrieve relevant information using different selection strategies: a simple semantic **similarity-based** retrieval, a **summarization-based** approach that condenses raw conversations before retrieval, and **GraphRAG** (Edge et al., 2024), which organizes knowledge into a structured graph and applies community detection for modular, query-focused summarization. Long-context models, in contrast, process full conversation histories without retrieval, either using the raw dialogue in its entirety or extracting key facts before feeding them into the model.

5.2 Evaluation Metrics

There are various ways to evaluate baseline performance, including question answering, event summarization, and dialogue generation. In this paper, we focus on **question answering (QA) task**, as it provides a clear and intuitive way to assess both the implicit nature of IMPLEXCONV compared to other benchmark datasets and the performance of baseline frameworks. We use three key metrics to evaluate the performance of different frameworks.

Retrieval Accuracy. To evaluate how well a framework retrieves the necessary background information while ensuring efficiency, we use the F1 score, which balances both relevance (recall) and conciseness (precision). Given a retrieved context C_r and the ground truth context C_g, we define retrieval accuracy as,

$$\text{Retrieval Accuracy} = \frac{2 \times |C_r \cap C_g|}{|C_r| + |C_g|}, \quad (4)$$

where $|C_r \cap C_g|$ represents the overlap between the retrieved and ground-truth contexts. This formulation ensures that retrieved content is both comprehensive and efficient, avoiding excessive retrieval that may introduce noise.

Answer Correctness. Since it is difficult for the predicted response to exactly match the ground truth, we prompt an LLM to judge whether the predicted answer is semantically equivalent to the ground truth. We also conduct human evaluation to verify correctness. A detailed illustration is provided in Appendix E.

Token Efficiency. While more information generally improves performance, excessive token usage increases computational cost. We analyze the trade-off between performance and token usage by measuring the token-to-accuracy ratio, which helps assess efficiency in long-term conversation retrieval.

5.3 Implementation Details

To mitigate hallucination, we employ different LLMs for dataset generation and framework implementation. Specifically, we use

Category	Method	Datasets					
		w/o implicit reason				w/ implicit reason	
		MSC	CC	LoCoMo	LongMemEval	ImplexConv (Supp.)	ImplexConv (Opp.)
Memory-based	MemoryBank	24.96	41.24	10.97	15.42	15.90	7.95
RAG	Raw	5.12	14.64	4.64	6.33	2.31	0.65
	Summary	21.27	40.75	**26.75**	19.87	12.66	5.58
	GraphRAG	7.42	9.69	10.28	11.69	3.00	0.85
TaciTree	Facts	28.43	35.83	14.36	15.05	28.96	7.62
	Summary	**42.65**	**46.20**	16.63	**20.15**	**55.18**	**14.84**

Table 2: Retrieval Accuracy (F1 score) across Different Frameworks and Datasets

`Llama-3.1-405B-Instruct` (Touvron et al., 2023) to construct the dataset and `GPT-4o-mini` (Achiam et al., 2023) to implement the framework and evaluate its performance across various baselines. For embedding representations, we use `stella_en_1.5B_v5` (Zhang et al., 2024) as the embedding model E. All prompts used in this paper are provided in the Appendix. To ensure diverse implicit reasoning scenarios, we set the similarity threshold $\beta = 0.4$. For clustering facts, we set the maximum cluster size to $k = 6$. The root-level cluster size L is set to 15. If the number of nodes at a level drops below this threshold, we terminate clustering and designate it as the root level to preserve high-level summarization.

For the QA task, we evaluate model performance across multi-session datasets to ensure consistency. Since MSC and CC do not contain explicit QA pairs, we treat the first four sessions as conversation history and evaluate QA performance based on the response in the fifth session. As these datasets also lack annotated evidence, we consider an answer correct if it aligns with the target response, indicating successful retrieval of relevant information. Furthermore, because ImplexConv comprises two distinct reasoning types—supportive and opposed implicit reasoning—each with fundamentally different QA dynamics, we evaluate them separately. To assess whether retrieved summaries or detailed facts contribute more effectively to accurate responses, we conduct evaluations using both the retrieved summaries and their corresponding original, detailed fact sets independently.

6 Results

This section presents our results on retrieval accuracy, response accuracy, and token efficiency. We analyze how well models retrieve implicit information, the trade-off between accuracy and token usage, and the challenges of opposed implicit reasoning.

6.1 Information Retrieval Accuracy

Our evaluation of retrieval accuracy, shown in Table 2, demonstrates that TaciTree outperforms all baselines on ImplexConv, achieving the highest F1 scores in both supportive (55.18%) and opposed (14.84%) implicit reasoning scenarios. This highlights its ability to effectively retrieve relevant implicit knowledge, even when semantic similarity with the query is low. Traditional RAG-based approaches perform competitively on datasets without implicit reasoning (e.g., CC: 40.75%, LoCoMo: 26.75%), but their performance drops significantly on implicit reasoning tasks, with scores of 12.66% (supportive) and 5.58% (opposed), respectively. This indicates that standard retrieval techniques struggle with retrieving non-explicit evidence. While GraphRAG incorporates structured retrieval through graph-based knowledge representation, it still underperforms across all datasets, suggesting that graph-based retrieval alone is insufficient for handling implicit reasoning (further analysis provided in the Appendix D).

6.2 Response Accuracy and Token Size

Response accuracy typically improves with increased retrieval content, but this comes at the cost of higher computational demands due to increased token usage. To systematically test this trade-off, we compare performance across various frameworks and datasets, as illustrated in Figure 5 and detailed in Tables 5 and 6. Our analysis reveals that models using larger token sets, particularly long-context methods that use entire raw conversations, tend to achieve higher QA accuracy. However, these improvements are often marginal relative to the substantial increase in computational resources

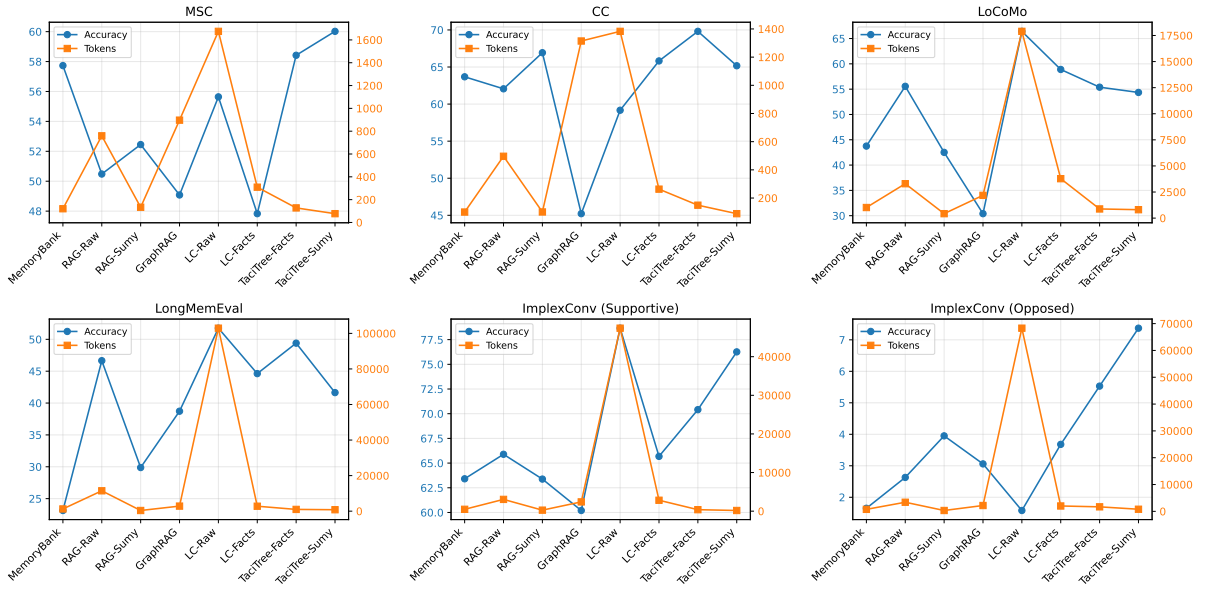

Figure 5: Response accuracy (blue) and retrieved token size (orange) across different frameworks and datasets.

Model	TaciTree		Long-Context	
	Summary	Facts	Raw	Facts
GPT-4o-mini	5.53	7.37	2.42	6.84
GPT-o3-mini	9.97	9.90	2.42	6.84
GPT-o1	**29.70**	**28.71**	**4.95**	**21.78**

Table 3: Response accuracy across different models.

	5	10	15	20	25
Supportive					
Facts	23.26	26.08	**28.96**	28.09	26.78
Summary	42.90	52.54	**55.18**	53.04	47.77
Opposed					
Facts	6.40	**7.80**	7.62	7.13	7.12
Summary	12.76	**15.40**	14.84	14.36	13.90

Table 4: Performance of TaciTree across varying root-level cluster sizes.

required. For example, in LongMemEval, retrieving the entire conversation (102,928 tokens) yields only a 2% accuracy gain compared to TaciTree, which retrieves only about 1% of the tokens used by the raw approach. TaciTree effectively balances accuracy and token efficiency by selectively retrieving relevant information. On the supportive reasoning subset of ImplexConv, TaciTree achieves a response accuracy of 76.26% while retrieving just 172.66 tokens. In contrast, RAG-Raw retrieves significantly more tokens (3045.94) but achieves lower accuracy 65.89%. This highlights TaciTree's ability to selectively retrieve crucial details efficiently, substantially reducing computational costs without compromising performance. Additionally, in opposed reasoning cases, long-context methods demonstrate poorer performance due to the excessive retrieval of semantically related yet misleading information. These methods often obscure critical implicit details with noisy contexts, significantly impacting their accuracy. TaciTree effectively mitigates this issue by selectively retrieving essential implicit reasoning details, thereby consistently outperforming long-context baselines in these challenging scenarios.

6.3 Opposed Implicit Reasoning Analysis

Table 3 presents the results for opposed reasoning, where retrieval accuracy remains high, but response accuracy is notably low. This suggests that implicit reasoning is particularly challenging, as noisy yet lexically relevant conversations can obscure the correct answer. To address this, we tested more powerful LLMs, including GPT-o3-mini and GPT-o1. The results indicate a clear trend: more powerful models perform better, with GPT-o1 achieving the highest accuracy at 29.70%, while GPT-o3-mini also outperforms GPT-4o-mini. These findings highlight the importance of stronger reasoning capabilities in handling complex implicit reasoning tasks.

6.4 Root-level Cluster Size

We further analyze the impact of the root-level cluster size L on retrieval performance for both supportive and opposed cases (Table 4). The results reveal a clear trade-off: when L is too small, overly coarse

summarization leads to information loss, while excessively large L introduces redundancy and reduces retrieval precision. In contrast, moderate cluster sizes maintain a balance between abstraction and detail. Specifically, $L = 15$ consistently achieves the best overall accuracy across both supportive and opposed settings, suggesting it provides an effective balance between high-level summarization and fine-grained retrieval. These findings validate the importance of tuning the cluster granularity in hierarchical retrieval and highlight that TACITREE 's performance is robust within a reasonable range of cluster sizes.

7 Conclusion and Future Work

In this work, we introduce IMPLEXCONV, a large-scale multi-session dataset designed to evaluate implicit reasoning in long-term personalized conversations. Unlike existing benchmarks, IMPLEXCONV incorporates subtle, semantically distant reasoning patterns that challenge traditional retrieval and long-context modeling approaches. To address these challenges, we propose TACITREE, a hierarchical tree-based framework that efficiently retrieves implicit knowledge while maintaining token efficiency. Our experiments demonstrate that IMPACT significantly improves retrieval and response accuracy, outperforming baselines with less cost. Future work includes enhancing implicit reasoning capabilities by integrating adaptive retrieval mechanisms and exploring more advanced LLM architectures to better handle complex, context-dependent reasoning in long-term dialogues.

Limitations

While IMPLEXCONV provides controlled evaluation of implicit reasoning, portions of the dataset are synthetically generated via LLM prompting, which may exhibit sensitivity to prompt design choices. Though we implement rigorous prompt engineering to ensure scenario quality, reproducibility across different LLM versions remains an open challenge. Furthermore, while our current implementation demonstrates efficient static retrieval, real-time updates to the hierarchical tree structure when integrating new dialogues could introduce computational costs. We identify promising mitigation strategies—such as incrementally assigning new dialogues to existing clusters and creating new branches only when necessary—as valuable directions for future work. These approaches would enable partial tree updates while preserving the core hierarchy, avoiding full reconstructions during dynamic deployment scenarios.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

Yiming Du, Hongru Wang, Zhengyi Zhao, Bin Liang, Baojun Wang, Wanjun Zhong, Zezhong Wang, and Kam-Fai Wong. 2024. Perltqa: A personal long-term memory dataset for memory classification, retrieval, and synthesis in question answering. *arXiv preprint arXiv:2402.16288*.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6491–6501.

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. 2024. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*.

Jihyoung Jang, Minseong Boo, and Hyounghun Kim. 2023. Conversation chronicles: Towards diverse temporal and relational dynamics in multi-session conversations. *arXiv preprint arXiv:2310.13420*.

Jiho Kim, Woosog Chay, Hyeonji Hwang, Daeun Kyung, Hyunseung Chung, Eunbyeol Cho, Yohan Jo, and Edward Choi. 2024. Dialsim: A real-time simulator for evaluating long-term dialogue understanding of conversational agents. *arXiv preprint arXiv:2406.13144*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. 2024. Hello again! llm-powered personalized agent for long-term dialogue. *arXiv preprint arXiv:2406.05925*.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*.

Tingting Liang, Chenxin Jin, Lingzhi Wang, Wenqi Fan, Congying Xia, Kai Chen, and Yuyu Yin. 2024. Llm-redial: A large-scale dataset for conversational recommender systems created from user behaviors with llms. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 8926–8939.

Jimmy Lin. 2009. Brute force and indexed approaches to pairwise document similarity comparisons with mapreduce. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 155–162.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating very long-term conversational memory of llm agents. *arXiv preprint arXiv:2402.17753*.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Michael McTear. 2022. *Conversational ai: Dialogue systems, conversational agents, and chatbots*. Springer Nature.

Douglas A Reynolds and 1 others. 2009. Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663).

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Bing Wang, Xinnian Liang, Jian Yang, Hui Huang, Shuangzhi Wu, Peihao Wu, Lu Lu, Zejun Ma, and Zhoujun Li. 2023. Enhancing large language model with self-controlled memory framework. *arXiv preprint arXiv:2304.13343*.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. 2024. Longmemeval: Benchmarking chat assistants on long-term interactive memory. *arXiv preprint arXiv:2410.10813*.

Wenhan Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajjwal Bhargava, Rui Hou, Louis Martin, Rashi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, and 1 others. 2023. Effective long-context scaling of foundation models. *arXiv preprint arXiv:2309.16039*.

J Xu. 2021. Beyond goldfish memory: Long-term open-domain conversation. *arXiv preprint arXiv:2107.07567*.

Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. 2023. Retrieval meets long context large language models. *arXiv preprint arXiv:2310.03025*.

Dun Zhang, Jiacheng Li, Ziyang Zeng, and Fulong Wang. 2024. Jasper and stella: distillation of sota embedding models. *arXiv preprint arXiv:2412.19048*.

Saizheng Zhang. 2018. Personalizing dialogue agents: I have a dog, do you have pets too. *arXiv preprint arXiv:1801.07243*.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 17, pages 19724–19731.

A Related Work

Long-term or Multi-session Dialogue Datasets Existing long-term dialogue datasets focus on different aspects of conversational memory and personalization. MSC(Xu, 2021) introduces five-session human-human dialogues with annotated summaries to enhance continuity. CC(Jang et al., 2023) and LoCoMo(Maharana et al., 2024) generate long-term multi-modal conversations with structured persona timelines. PerLTQA(Du et al., 2024) emphasizes personalized long-term QA without multi-session dialogue dynamics. Furthermore, DialSim(Kim et al., 2024) assesses dialogue similarity, focusing on coherence rather than long-term retrieval. Recent dataset LongMemEval(Wu et al., 2024) tests memory retention over multi-turn dialogues but lacks personalization. In contrast, ImplexConv (Table 1) is the first large-scale dataset designed for implicit reasoning in long-term conversations, incorporating both opposed and supportive scenarios that challenge retrieval-based and long-context models.

Long-term Memory Methodology To enhance long-term conversational reasoning, methods like MemoryBank (Zhong et al., 2024) and LDAgent (Li et al., 2024) incorporate structured memory mechanisms, while SCM (Wang et al., 2023)

utilizes structured conversational memory for efficient information retention and retrieval. RAG frameworks, including LlamaIndex, LangChain, and Haystack, enable structured retrieval to integrate relevant past context, with GraphRAG (Edge et al., 2024) further leveraging graph-based knowledge representation for improved contextual retrieval. Long-context processing remains an active research area, focusing on adapting LLMs to handle extended prompts; however, performance typically degrades as context length increases. Techniques such as hierarchical memory representations and adaptive retrieval mechanisms attempt to address this limitation. Our proposed approach, TACITREE, introduces a hierarchical tree framework that structures conversation history into multiple levels of summarization, significantly improving LLMs' ability to reason over long-term conversations with implicit contextual dependencies.

B Persona Extraction

We include some personas used to generate implicit conversations, such as, "This person enjoys listening to pop music," "This person likely engages in nostalgic experiences related to Azerbaijani culture," and "This person is a casual listener of classic rock music."All personas are formatted as short sample sentences, making it easier to create implicit reasoning.

C Dataset Evaluation and Human Verification

We initially prompt the LLM for selection using the template provided in Code 5. However, when the LLM provides multiple responses or fails to return an answer, we rely on human annotators. Specifically, in such cases, three annotators independently select the best reasoning instance using the same instructions as in Code 5. We retain only those instances agreed upon by at least two annotators.

D GraphRAG Analysis

Graph-based Retrieval-Augmented Generation (GraphRAG) extends traditional retrieval methods by structuring knowledge in a graph representation where nodes correspond to relevant entities or concepts, and edges encode relationships. This enables contextual retrieval beyond simple lexical similarity. However, in the case of our dataset, GraphRAG underperforms due to challenges in implicit reasoning, graph construction, and query generation.

The performance of GraphRAG heavily depends on the quality of the graph it constructs. In traditional explicit reasoning tasks, nodes represent well-defined entities (e.g., named entities, known facts) and edges reflect structured relationships. However, our dataset focuses on implicit reasoning, where relevant connections are syntactic, semantic, or pragmatically inferred rather than explicitly defined. As a result, the model struggles to create meaningful edges that capture indirect relationships between persona traits. Key implicit details are often embedded across multiple dialogue sessions, making single-instance graph representations insufficient. Graph sparsity leads to retrieval failures, as distant yet relevant information remains inaccessible due to missing edges.

Effective retrieval in GraphRAG depends on correctly structuring queries that retrieve the most relevant nodes. However, in our dataset, implicit reasoning requires multi-hop retrieval, yet GraphRAG often retrieves single-hop neighbors, missing deeper contextual connections. Furthermore, the LLM relies on semantic similarity-based retrieval, which fails when implicit reasoning requires retrieving conceptually related but lexically distant nodes. In addition, over-reliance on direct lexical matching leads to retrieval noise, where GraphRAG incorrectly prioritizes surface-level matches over deeper reasoning-based connections.

E Answer Correctness Evaluation

The prompt used for LLM-based evaluation is provided in Code 9. For human evaluation, we randomly sample 100 examples from each dataset and instruct human annotators to assess the validity of the model-generated responses, following the guidelines detailed in Code 10.

Category	Method	Datasets					
		MSC	CC	LoCoMo	LongMemEval	IMPLEXCONV (Supp.)	IMPLEXCONV (Opp.)
Memory-based	MemoryBank	57.73	63.68	43.75	23.14	63.41	1.65
RAG	Raw	50.48	62.06	55.56	46.65	65.89	2.63
	Summary	52.45	66.93	42.51	29.88	63.38	3.95
	GraphRAG	49.08	45.22	30.40	38.72	60.20	3.06
Long-Context	Raw	55.64	59.17	**66.42**	**51.75**	**78.66**	1.58
	Facts	47.83	65.82	58.89	44.62	65.68	3.68
IMPACT	Facts	58.42	**69.81**	55.39	49.40	70.41	5.53
	Summary	**60.02**	65.17	54.35	41.64	76.26	**7.37**

Table 5: Response Accuracy across Different Frameworks and Datasets

Category	Method	Datasets					
		MSC	CC	LoCoMo	LongMemEval	IMPLEXCONV (Supp.)	IMPLEXCONV (Opp.)
Memory-based	MemoryBank	120.22	100.55	1015.10	1242.93	479.34	743.44
RAG	Raw	759.26	496.39	3296.15	11448.38	3045.94	3395.81
	Summary	133.41	101.17	**426.17**	**374.49**	262.43	**319.33**
	GraphRAG	896.78	1314.37	2182.91	2845.27	2394.82	2178.09
Long-Context	Raw	1675.04	1382.94	17896.13	102928.90	47384.54	68299.85
	Facts	309.23	263.11	3778.62	2781.60	2825.89	2002.24
IMPACT	Facts	127.61	149.22	877.98	983.39	384.21	1682.26
	Summary	**77.07**	**89.06**	791.48	806.97	**172.66**	786.97

Table 6: Average Number of Tokens across Different Frameworks and Datasets

````
prompt = "
   Here is a brief description of a person:
    {persona}

    Please break it down into several components, including: "demographics" (including name, age, living location, birthplace,
      marital status, etc.), "career_life_and_goals" (make sure this part only contains things related with the person's career
      life), and "everyday_life_and_hobbies" (make sure this part is nothing related with the person's career life). Just list
      those information that are presented and leave others that are unknown. Below are some examples, try to make each point
      separate from each other and self-explanable. Only output a JSON object like in the following examples.
    Example 1:
    Input: An eco-friendly lifestyle podcaster who features change-makers and promotes sustainable living
    Output:
    ```json
    {{
        "demographics": {{
            "occupation": "This person is an eco-friendly lifestyle podcaster."
        }},
        "career_life_and_goals": [
            "This person features change-makers and promotes sustainable living."
        ]
    }}
    ```

    Example 2:
    Input: a nostalgic Azerbaijani pop music lover
    Output:
    ```json
    {{
        "demographics": {{
            "nationality": "This person is from Azerbaijani."
        }},
        "everyday_life_and_hobbies": [
            "This per``son enjoys listening to pop music.",
            "This person likely engages in nostalgic experiences related to Azerbaijani culture."
        ]
    }}
    ```
    "
````

Code 1: Prompt to extract persona traits

```
prompt = "
    {per_info} However, they have not been able to do it recently. Can you give me at least 20 implicit reasons why that person
      cannot do it?
    The reasons should be completely different from each other and belong to different categories.
    The reason should be specific with detailed information, like why it happens.
    The reason cannot include words related to "{traits_info}"
    Please explain the reasoning in only one sentence. Please only output the reasons with the format:
    1:
    2:
    "
```

Code 2: Prompt to generate opposed implicit reasoning

```
prompt = "
    {per_info} Can you give me at least 20 implicit reason information that supports this claim? Therefore, if I ask you, "Does
      {per_info}?", you have to answer "yes".
    The reason information should be completely different from each other and belong to different categories.
    The reason should be specific with detailed information, like why it happens.
    The reason cannot include words related to "{traits_info}"
    Please explain the reasoning in only one sentence. Please only output the reasons with the format:
    1:
    2:
    "
```

Code 3: Prompt to generate supportive implicit reasoning

```
prompt = "
    Here's the conversation between a user(speaker 1) and a chatbot assistant.
    Speaker 1 has the following persona trait: {per_info}. However, speaker 1 cannot do the trait due to the reason that
      {reason_info}.
    Now, speaker 1 asks you a question related to the trait. {reason_info} affect your answer to this question.
    You should tell speaker 1 they cannot do the trait due to the reason.
    The trait should be mentioned in the question.
    The question itself should not mention the reason or effect of the reason.
    Questions should be asked in the first person. Include "I".
    The question should not be a yes/no question.
    The question needs to be diverse.

    Please only output the question in the format of less than 20 words without any additional sentences.
    "
```

Code 4: Prompt to generate opposed implicit question

```
prompt = "
    {per_info}. Here are potential implicit reasons why this person is unable to follow this trait: {str_reason}.
    Could you select the reason that is both the most logically sound and subtly implied?
    Please select only from the provided options and output the reason only.
    "
```

Code 5: Prompt to select opposed implicit reasoning

```
prompt = "
    Consider a person with specific personality traits {persona} that could serve as responses to a given question {question}.
    Can you generate additional scenarios that reflect or align with these personality traits to support the question?
    Please output 5 scenarios that are relevant to the given traits and question.
    The scenarios should contain only one sentence.
    The scenarios can talk about both {traits_info} or other stuff that is related to {traits_info} but do not have to be the
      same.
    Please output the scenarios only with the index number.

    For example:

    Trait: I love sports
    Question: I'm bored; can you give me some suggestions?
    Scenarios:
    1. I love playing basketball.
    2. My favorite basketball player is Stephen Curry.
    "
```

Code 6: Prompt to generate noisy scenarios

```
prompt = "
    There are two speakers. Speaker 1 encounters the scenario that "{scenario}". Speaker 2 is the AI assistant.
    Based on the information. Can you generate a conversation with at least 10 turns?
    Speaker 1 shouldn't mention the scenario too early. It must be mentioned in the later section.
    Speaker 1 is exactly the person who encounters the scenario.
    The beginning turns should serve as a warm-up to introduce the scenario in a natural way.
    The conversation should be centered around the scenario without any irrelevant or extra information that is not related to
      the scenario.
    For Spearker 1, please do not start the conversation by saying something similar to "I'm feeling a bit overwhelmed lately."
      or use the same format as this sentence.
    Include diverse styles like detailed explanations, step-by-step guidance, casual small talk, humor, storytelling, and
      problem-solving.
    The conversation should feel realistic and flow naturally.
    Aim for a balance of formality and informality, capturing nuanced exchanges that go beyond simple responses.
    Please output the conversation in the following format:
    Speaker1: ...
    Assistant: ...

    Speaker1: ...
    Assistant: ...
    "
```

Code 7: Prompt to generate noisy conversations

```
prompt = "
    Can you summarize {text} in one sentence to only contain the high-level information?
    Please only output the summary without anything else.
    "
```

Code 8: Prompt to summarize facts

```
prompt = "
    Given question: {question}.

    The ideal response is: {ground_truth}. The model's response is: {answer}.

    Do you consider the model's response valid and consistent with the ideal response? The response may contain additional
      information, but we only care about whether the ideal response is present. Please answer only with "yes" or "no" and
      explain the reason.
    "
```

Code 9: Prompt to evaluate answer correctness with LLM judgement

```
prompt = "
    Given question: {question}. The ideal response is: {ground_truth}. The model's response is: {answer}.

    Do you consider the model's response a valid answer to the question? If the response matches the ideal answer, it is valid.
      Otherwise, determine whether it still validly addresses the question even if differing from the ideal response. Please
      answer with only "yes" or "no".
    "
```

Code 10: Prompt to evaluate answer correctness with human judgement