StructFlowBench: A Structured Flow Benchmark for Multi-turn Instruction Following

Anonymous ACL submission

Abstract

Multi-turn instruction following capability constitutes a core competency of large language models (LLMs) in real-world applications. Existing evaluation benchmarks predominantly focus on fine-grained constraint satisfaction and domain-specific capability assessment, yet overlook the crucial structural dependency between dialogue turns that distinguishes multi-turn from single-turn interactions. This structural dependency not only reflects user intent but also establishes a second dimension for instruction following evaluation beyond constraint satisfaction. To address this gap, we propose StructFlowBench, a multi-turn instruction following benchmark with structural flow modeling. The benchmark innovatively defines a structural flow framework comprising six fundamental inter-turn relationships, which not only introduces novel structural constraints for model evaluation but also serves as generation parameters for creating customized dialogue flows tailored to specific scenarios. Adopting established LLM-based automatic evaluation methodologies, we conduct systematic evaluations of 13 leading open-source and closed-source LLMs. Experimental results reveal significant deficiencies in current models' comprehension of multi-turn dialogue structures.

1 Introduction

The rapid advancement of large language models (LLMs) in multi-turn dialogue systems has elevated instruction-following capabilities to a pivotal research frontier in human-AI interaction (Chang et al., 2024). Current evaluation methodologies bifurcate into two streams: multi-turn dialogue evaluations focusing on capability evaluation (Zheng et al., 2023b; Bai et al., 2024; Kwan et al., 2024) and instruction-following analyses emphasizing fine-grained constraint compliance (Jiang et al., 2024; He et al., 2024a; Zhang et al., 2024). More recent research has started to model the composition of intra-turn constraints (Wen et al., 2024).

However, current evaluation methodologies treat multi-turn dialogues as simple concatenations of single-turn interactions, neglecting users' planning and intentionality in extended conversations. This leads to three critical limitations: (1) Failure to model complex scenarios: Multi-turn dialogue data constructed with simplistic linear thinking cannot accurately capture key characteristics of real-world complex conversations, such as logical coherence, user goal clarity, and natural transitions. (2) Methodological bias: Single-turn evaluation strategies fragment inter-turn structural connections, overlooking multi-turn structural constraints. (3) Analytical deficiency: Existing approaches overemphasize intra-turn-level constraint compliance while lacking a systematic framework to characterize dialogue structural flow.

To bridge these gaps, we introduce **StructFlowBench**, a novel instruction-following benchmark integrating a multi-turn structural flow framework. It consists of two key components: 1) **Dual-constraint evaluation system**, which combines 8 intra-turn instruction constraints with 5 newly proposed structural constraints, enabling a more comprehensive assessment of multi-turn dialogue instruction following capabilities of LLMs. These structural constraints account for inter-turn dependencies, ensuring that models are evaluated not only on their ability to satisfy individual constraints but also on their capacity to maintain logical coherence across multiple turns. 2) **Six-category structural flow taxonomy**, encompassing six fundamental inter-turn relationships: *Follow-up, Refinement, Recall, Summary, Expansion, Unrelatedness*. The illustration of the structural flow taxonomy and an example of the structural flow are presented in Figure 1. This taxonomy serves a tripartite function: (a) **Diagnostic evaluation:** It enables a structured analysis of cross-turn structural rationality, helping to identify inconsistencies in dialogue flow and ensuring that model responses align with the ex-

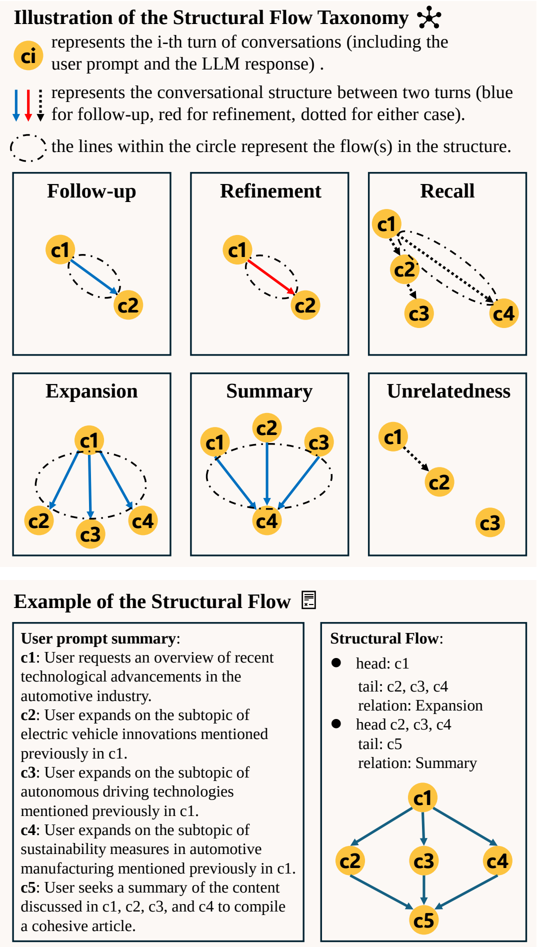


Illustration of the Structural Flow Taxonomy

ci represents the i-th turn of conversations (including the user prompt and the LLM response) .

represents the conversational structure between two turns (blue for follow-up, red for refinement, dotted for either case).

the lines within the circle represent the flow(s) in the structure.

Example of the Structural Flow

User prompt summary:
c1: User requests an overview of recent technological advancements in the automotive industry.
c2: User expands on the subtopic of electric vehicle innovations mentioned previously in c1.
c3: User expands on the subtopic of autonomous driving technologies mentioned previously in c1.
c4: User expands on the subtopic of sustainability measures in automotive manufacturing mentioned previously in c1.
c5: User seeks a summary of the content discussed in c1, c2, c3, and c4 to compile a cohesive article.

Structural Flow:

- head: c1
  tail: c2, c3, c4
  relation: Expansion
- head c2, c3, c4
  tail: c5
  relation: Summary

Figure 1: The Structural Flow Taxonomy includes six fundamental structures, each used to describe the inter-turn relationships in multi-turn dialogues. It can be applied to analyze any dialogue and generate specific structural flows.

pected discourse structure. (b) **Intent inference:** By analyzing structural patterns, this taxonomy facilitates the extraction of implicit user intent, offering a deeper understanding of how instructions evolve over multiple turns. (c) **Controlled generation:** The taxonomy provides configurable structural parameters that guide task-specific dialogue simulation, allowing for the tailored generation of multi-turn conversations with predefined structural patterns. This not only enhances dataset diversity but also supports the development of more robust instruction-following models adaptable to varied real-world applications.

We summarize our contributions as follows:

- **Structural Flow Taxonomy**: We propose a six-category structured taxonomy for multi-turn instruction-following evaluation, offering an interpretable framework for analyzing dialogue structural flow.
- **StructFlowBench**: We introduce StructFlowBench, a structurally annotated multi-turn benchmark that leverages a structure-driven generation paradigm to enhance the simulation of complex dialogue scenarios.
- **Comprehensive LLM evaluation**: We systematically evaluate 13 state-of-the-art LLMs (3 closed-source and 10 open-source), unveiling disparities in structural processing capabilities and providing empirical insights for optimizing dialogue systems.

2 Related Work

2.1 Benchmarks for Multi-Turn Dialogues

The evolution of dialogue evaluation paradigms has progressed from single-turn assessments to sophisticated multi-turn interaction analysis (Wang et al., 2023; Sun et al., 2024; Duan et al., 2024). Among these, MT-Bench (Zheng et al., 2023b) pioneered this transition by providing methodologies specifically designed to assess a model's ability to handle multi-turn interactions. Building upon this, MT-Bench-101 (Bai et al., 2024) introduces a more granular evaluation framework to assess fine-grained capabilities. Multi-IF (He et al., 2024b) expands single-turn dialogues into multi-turn interactions by following simple, predefined linear paths. However, most existing work on multi-turn dialogue evaluation does not prioritize the assessment of instruction following and overlooks the influence of structural information on the evaluation of multi-turn dialogues. The four modes of user-assistant interactions proposed by MT-Eval (Kwan et al., 2024) cover certain structural information within multi-turn dialogues. However, MT-Eval does not establish a systematic structural framework and lacks integration of various structural aspects for a comprehensive evaluation.

2.2 Benchmarks for Instruction Following

Recent instruction following evaluation predominantly employs constraint-based frameworks (Jiang et al., 2024; Zhang et al., 2024; He et al., 2024a; Zhou et al., 2023). InfoBench (Qin et al., 2024) introduces the Decomposed Requirements Following Ratio (DRFR) metric, which provides a more granular scoring system by breaking down the evaluation of complex instructions into assessments of their individual simple constraints. Furthermore, ComplexBench (Wen et al., 2024) explores instruction-following capabilities in single-turn complex dialogues through empirical studies of constraint composition. However, prior work on instruction-following evaluation has primarily focused on single-turn interactions, which do not align with the more common multi-turn dialogue scenarios observed in real-world user interactions. While some studies have attempted to split complex single-turn instructions into multi-turn dialogues, these approaches do not fully capture the intentionality and goal-oriented nature of users in real-world contexts.

3 StructFlowBench

In this section, we first introduce the structural flow framework and the constraint categories of our benchmark. Next, we detail the data construction pipeline and present an overview of the statistics for StructFlowBench. Finally, we outline the evaluation protocol.

3.1 Structural Flow Taxonomy

By analyzing existing LLM and real human multi-turn dialogue datasets (such as *WILDCHAT* (Zhao et al., 2024) and *LMSYS-Chat-1M dataset* (Zheng et al., 2023a)), we identified and categorized six structural patterns of multi-turn dialogues to enhance the understanding and analysis of conversational structural flow.

Follow-up: An adjacent-turn structure where the user's next prompt builds on the content of the previous turn, incorporating details from either the user's previous prompt or the AI's previous response. This is the most common structure in multi-turn dialogues, typically reflecting the user's intent to explore the topic more deeply.

Refinement: An adjacent-turn structure in which the user modifies or clarifies their immediate previous prompt to improve the AI's response. This structure usually signals the user's dissatisfaction with the prior response, prompting them to refine the prompt while clarifying and emphasizing their concerns to obtain a more satisfactory response.

Recall: A long-range structure in which the user refers back to content from two or more turns ago to provide context for the current prompt (long-range follow-up) or referencing prior content for clarification (long-range refinement).

Expansion: A multi-turn "fan-out" structure where the user introduces a main theme and explores related subtopics in subsequent turns. This structure suggests that the user's following turns are focused on specific subtopics derived from a particular point in the conversation.

Summary: A multi-turn "fan-in" structure in which the user requests a consolidation of content from multiple previous turns into a cohesive overview. This structure acts as the counterpart to expansion, reflecting the need to summarize and condense the information discussed in earlier turns.

Unrelatedness: A conversational structure in which the user's prompt is entirely independent of the previous turn, with no reference to prior content or context. This structure often occurs in everyday use of LLMs by non-experts, where a new topic is introduced within a previously unrelated dialogue, rather than starting a new conversation.

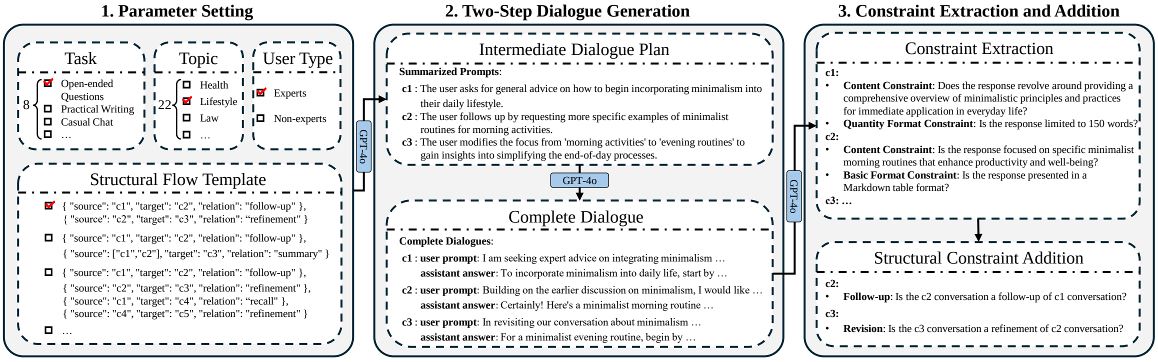
After defining the six basic dialogue structures, we can use the Structural Flow Taxonomy to analyze multi-turn dialogue data and construct the corresponding structural flows.

3.2 Constraint Categories

We categorize our constraints into **intra-turn constraints** and **multi-turn structural constraints**. Details related to constraints can be found in Appendix B.

For **intra-turn constraints**, we synthesize and refine constraint classification systems from several works in this field (e.g., IF-Eval (Zhou et al., 2023), CFBench (Zhang et al., 2024), FollowBench (Jiang et al., 2024)). Based on this synthesis, we categorize constraints into eight types: *Inverse Constraint, Style Constraint, Situation Constraint, Keyword/Element Constraint, Basic Format Constraint, Quantity Format Constraint, Template Format Constraint, and Content Constraint*.

For **multi-turn structural constraints**, we define five types of structural design constraints, excluding the "unrelatedness" structure. These constraints are specifically designed to maintain logical coherence and continuity across multiple turns in a dialogue. They ensure that the structural relationships between turns are consistent and contextually relevant, enabling a smooth flow of conversation. The five types of constraints are aimed at handling key aspects such as follow-ups, refinements, recalls, expansions, and summaries, ensuring that each turn in the dialogue properly connects to the previous ones while adhering to the intended conversational structure.



☑ { "source": "c1", "target": "c2", "relation": "follow-up" },
{ "source": "c2", "target": "c3", "relation": "refinement" }

□ { "source": "c1", "target": "c2", "relation": "follow-up" },
{ "source": ["c1","c2"], "target": "c3", "relation": "summary" }

□ { "source": "c1", "target": "c2", "relation": "follow-up" },
{ "source": "c2", "target": "c3", "relation": "refinement" },
{ "source": "c1", "target": "c4", "relation": "recall" },
{ "source": "c4", "target": "c5", "relation": "refinement" }

□ …

Summarized Prompts:

c1 : The user asks for general advice on how to begin incorporating minimalism into their daily lifestyle.

c2 : The user follows up by requesting more specific examples of minimalist routines for morning activities.

c3 : The user modifies the focus from 'morning activities' to 'evening routines' to gain insights into simplifying the end-of-day processes.

Complete Dialogues:

c1 : **user prompt**: I am seeking expert advice on integrating minimalism …
assistant answer: To incorporate minimalism into daily life, start by …

c2 : **user prompt**: Building on the earlier discussion on minimalism, I would like …
assistant answer: Certainly! Here's a minimalist morning routine …

c3 : **user prompt**: In revisiting our conversation about minimalism …
assistant answer: For a minimalist evening routine, begin by …

c1:
- **Content Constraint**: Does the response revolve around providing a comprehensive overview of minimalistic principles and practices for immediate application in everyday life?
- **Quantity Format Constraint**: Is the response limited to 150 words?

c2:
- **Content Constraint**: Is the response focused on specific minimalist morning routines that enhance productivity and well-being?
- **Basic Format Constraint**: Is the response presented in a Markdown table format?

c3: …

c2:
- **Follow-up**: Is the c2 conversation a follow-up of c1 conversation?

c3:
- **Revision**: Is the c3 conversation a refinement of c2 conversation?

Figure 2: The construction pipeline of StructFlowBench. First, tasks, topics, user types, and structural flow templates are defined. Then, dialogue data is generated in two steps: intermediate dialogue plans (i.e., the summarized prompts) are created from the structural flow, followed by generating complete dialogues from these plans. Finally, intra-turn constraints are extracted by GPT-4o, and structural constraints are added based on the structural flow information.

3.3 Data Construction Pipeline

The construction pipeline of StructFlowBench, as shown in Figure 2, comprises three main components: parameter setting, two-step dialogue generation, and constraint extraction and addition. All prompt templates used in the data construction process are included in Appendix D, and a sample data instance is provided in Appendix E.

Parameter Setting

Before dialogue generation, we select parameters such as topic, task, user characteristics, and structural flow template, ensuring comprehensive coverage of the evaluation scope for multi-turn dialogue generation. For task types, we refer to the taxonomy of ComplexBench (Wen et al., 2024), adapting it to our evaluation framework and selecting eight task types. For topics, we draw from the MT-Bench-101 (Bai et al., 2024) framework, making necessary adjustments to suit our context, and ultimately select 22 topics. For user characteristics, we consider the significant differences in questioning styles and language between experts and non-experts. For the structural flow template, we designed multiple templates based on insights from real data and specific scenarios.

Two-Step Dialogue Generation

We employ a two-step process to generate a dialogue for a parameter setting. The first step uses the structural flow template to generate an intermediate dialogue plan (i.e., summarized prompts) via GPT-4o. Locally deployed mini-models perform initial screening and manual inspection of error data to ensure the dialogue plan aligns with the structural flow. In the second step, each intermediate dialogue plan is used to generate a complete dialogue, including user prompts and LLM responses via GPT-4o. This approach ensures high-quality generation of both dialogue content and structure while minimizing manual effort.

Constraint Extraction and Addition

For the complete multi-turn dialogue data, we extract intra-turn constraints using the GPT-4o, followed by manual validation to ensure accuracy. Based on the structural flow information, we then assign the corresponding multi-turn structural constraints to each dialogue turn.

3.4 Benchmark Dataset Statistics

Table 1 presents a comparison of related benchmark datasets, evaluating them from three perspectives: fine-grained constraints, multi-turn dialogue assessment, and structural information. Our StructFlowBench encompasses 8 task types, 22 topics, and 13 constraint types. It ultimately includes 155 multi-turn dialogues, comprising a total of 643 turns and 1,775 constraints. Detailed statistics for tasks and topics are provided in the Appendix A.

3.5 Evaluation

Evaluation Criteria

Drawing on the methodology of MT-Bench-101 (Bai et al., 2024), we implemented the “Golden Context” approach in our evaluation framework. Instead of relying on model-generated contexts, this method uses carefully curated datasets as dialogue

Benchmark	#Dialogues	Avg. #Turns	#Constraint Types	Fine-grained Constraint	Multi-turn Assessment	Structural Information
IFEval	541	1	4	✓	✗	✗
CELLO	523	1	4	✓	✗	✗
FollowBench	820	1	6	✓	✗	✗
InfoBench	500	1	5	✓	✗	✗
CFBench	1000	1	10	✓	✗	✗
ComplexBench	1150	1	19	✓	✗	✗
MT-Bench-101	1388	3.03	-	✗	✓	✗
Multi-if	4501	3	24	✓	✓	✗
MT-Eval	168	6.96	-	✗	✓	△
StructFlowBench	155	4.14	13	✓	✓	✓

Table 1: Comparisons between STRUCTFLOWBENCH and other related benchmark datasets. △ represents partially satisfied.

histories. By providing accurate and consistent contexts for each dialogue turn, it minimizes biases and noise, improving the reliability, fairness, and comparability of response quality assessments across different models.

To achieve a fine-grained evaluation of multi-turn user instructions, we integrate insights from prior studies (Qin et al., 2024; Wen et al., 2024; Zhang et al., 2024; He et al., 2024a) and propose an assessment method based on constraint decomposition and binary question formulation. Specifically, we decompose each user instruction into multiple independent constraints and design concise binary questions for each, answered with a simple "Yes" or "No" to assess satisfaction. These binary questions are then aggregated into a checklist that comprehensively covers all critical constraints of the instruction.

Building on this foundation, we further adopt the approach of leveraging state-of-the-art LLMs for evaluation, as outlined in MT-Bench (Zheng et al., 2023b). In our implementation, we use the advanced GPT-4o as the LLM evaluator. By providing the evaluator with the golden context, response of the test model, the constraint checklist, and a carefully crafted prompt template, we ensure high consistency and reliability in the evaluation process. The prompt template is designed to emphasize key evaluation points, effectively enhancing the accuracy and credibility of the results.

Evaluation Metrics

We adopted several existing metrics, including Constraint Satisfaction Rate (CSR) and Instruction Satisfaction Rate (ISR) (Zhang et al., 2024), as well as Decomposed Requirements Following Ratio (DRFR) (Qin et al., 2024).

The **Constraint Satisfaction Rate (CSR)** evaluates the average proportion of satisfied constraints across all instructions, calculated as CSR $= \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{n_i}\sum_{j=1}^{n_i} s_i^j\right)$, where m represents the total number of instructions, n_i denotes the number of constraints in the i-th instruction, and $s_i^j \in \{0,1\}$ indicates whether the j-th constraint in the i-th instruction is satisfied.

The **Instruction Satisfaction Rate (ISR)** measures the proportion of instructions where all constraints are fully satisfied, computed as ISR $= \frac{1}{m}\sum_{i=1}^{m} s_i$, where $s_i \in \{0,1\}$ indicates whether all constraints in the i-th instruction are satisfied.

The **Decomposed Requirements Following Ratio (DRFR)** evaluates the overall satisfaction of requirements across all instructions, defined as DRFR $= \frac{\sum_{i,j} r'_{i,j}}{\sum_i m_i}$, where m_i is the number of scoring questions for the i-th instruction, and $r'_{i,j}$ denotes the result of the j-th scoring question in the i-th instruction.

Despite their utility, these existing metrics have limitations. For instance, CSR treats all constraints equally without considering their relative importance, while ISR provides a binary evaluation that may overlook partial fulfillment of constraints. To overcome these limitations, we introduce the **Weighted Constraint Satisfaction Rate (WCSR)**, defined as **WCSR** $= \frac{\sum_{j=1}^{n} w_j \cdot s_j}{\sum_{j=1}^{n} w_j}$, which incorporates weighted factors to account for the varying significance of different constraint types. Here, n denotes the total number of constraints, w_j represents the weight assigned to the j-th constraint, and $s_j \in \{0,1\}$ indicates whether the j-th constraint is satisfied. In our framework, intra-turn constraints are assigned a weight of $w_r = 1$, whereas structural constraints, which play a critical role in ensuring coherence and correctness, are given a higher weight of $w_s = 2$.

The introduction of WCSR provides a more nuanced evaluation by emphasizing important constraints through weighted assessments. This improves the precision and relevance of evaluations, enhancing the reliability of LLMs in meeting complex requirements.

4 Experiments

4.1 Experimental Setup

We evaluate 13 popular LLMs on StructFlow-Bench, including 3 closed-source models (GPT-4o (Hurst et al., 2024), Claude-3.5-Sonnet (Anthropic, 2024) and Gemini-1.5-Pro (Team et al., 2024)) and 10 open-source models: Llama-3.1-Instruct-8B (Dubey et al., 2024), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023), Qwen2.5-7B-Instruct, Qwen2.5-14B-Instruct (Yang et al., 2024), Yi-6B-Chat (Young et al., 2024), Phi-3.5-mini-instruct (Abdin et al., 2024), GLM-4-9B-Chat (GLM et al., 2024), Deepseek-R1-Distill-Llama-8B, Deepseek-R1-Distill-Qwen-7B (Guo et al., 2025) and DeepSeek-v3 (Liu et al., 2024). More details on these evaluated models can be found in Appendix F.

4.2 Main Results

Overall Results

Table 2 presents a comprehensive evaluation of 13 representative LLMs on StructFlowBench, covering four key metrics as well as assessments of structural constraints. The detailed results, categorized by intra-turn constraints and task types, are provided in the Appendix C.

The recently released DeepSeek-v3 outperforms all other models across all metrics, demonstrating its exceptional capability in fine-grained constraint satisfaction and multi-turn dialogue structure understanding. Gemini-1.5-Pro and GPT-4o closely follow, achieving comparable performance in intra-turn constraints but showing slightly weaker results in adhering to structural constraints for multi-turn dialogues. Claude-3.5-Sonnet, GLM-4-9B-Chat, Qwen2.5-14B-Instruct, and Qwen2.5-7B-Instruct also exhibit strong instruction-following capabilities, with CSR exceeding 94%. Notably, all seven of these models achieve high DRFR scores, indicating their strong ability to follow fine-grained instructions.

In contrast, mid-tier models such as Deepseek-R1-Distill-Llama-8B, Llama-3.1-8B-Instruct, Phi-3.5-Mini-Instruct, and Yi-6B-Chat perform reasonably well but exhibit greater instability, particularly in ISR and WCSR. While they handle simpler constraints effectively, they struggle with maintaining consistency when processing complex instructions and multi-turn dialogue structures. The weakest performers in multi-turn instruction following are Deepseek-R1-Distill-Qwen-7B and Mistral-7B-Instruct-v0.3, revealing significant deficiencies in natural interaction scenarios.

A particularly interesting observation is that Deepseek-R1-Distill-Llama-8B, distilled from Llama-3.1-8B, outperforms Llama-3.1-8B-Instruct across all metrics, demonstrating the effectiveness of the distillation process. However, Deepseek-R1-Distill-Qwen-7B, distilled from Qwen2.5-Math-7B, underperforms due to its origin from a model optimized primarily for mathematical reasoning tasks, which inherently makes it weaker in multi-turn dialogue instruction following compared to Qwen2.5-7B-Instruct.

One particularly noteworthy outcome is that DeepSeek-v3, an open-source model, surpasses its closed-source counterparts in multi-turn instruction-following evaluations. This result is encouraging for both the research community and the open-source ecosystem, suggesting that the theoretical advancements and training methodologies behind DeepSeek-v3 could offer valuable insights for improving LLMs in multi-turn instruction-following tasks.

Structural-Constraint-Categorized Performance

The evaluated LLMs exhibit strong performance in follow-up structures, with nearly all models excelling in maintaining contextual continuity and generating coherent responses. Additionally, most models handle recall structures well, demonstrating their ability to reference prior conversational turns effectively. However, performance varies when dealing with more complex structures such as summary and expansion. DeepSeek-v3 and proprietary models outperform the others, indicating their superior capability in nuanced content condensation and elaboration. In contrast, refinement tasks pose a significant challenge across all models. Even the strongest model, DeepSeek-v3, achieves only 0.8 in this category, highlighting the inherent difficulty of processing refinements accurately and maintaining coherence when adapting to modified user inputs. While LLMs exhibit strong instruction-following abilities in structured dialogue, refinement remains the most challenging task, requiring improvements in dynamic response adaptation. Future advancements should focus on enhancing models' flexi-

Model Name	follow-up	refinement	expansion	summary	recall	CSR	ISR	WCSR	DRFR
Deepseek-v3	**0.99**	**0.8**	**0.92**	**1.0**	**1.0**	**0.98**	**0.93**	**0.96**	**0.98**
Gemini-1.5-Pro	0.97	0.78	0.91	**1.0**	0.94	0.97	0.91	0.95	0.97
GPT-4o	0.98	0.78	0.88	0.97	0.91	0.97	0.9	0.95	0.97
Claude-3.5-Sonnet	0.98	**0.8**	0.88	**1.0**	0.91	0.95	0.88	0.94	0.96
GLM-4-9B-Chat	0.95	0.75	0.84	0.97	0.94	0.95	0.86	0.93	0.95
Qwen2.5-14B-Instruct	0.97	0.73	0.87	0.97	0.97	0.94	0.84	0.92	0.94
Qwen2.5-7B-Instruct	0.95	0.76	0.9	0.94	0.97	0.94	0.84	0.92	0.94
Deepseek-R1-Distill-Qwen-7B	0.91	0.62	0.85	0.86	0.78	0.81	0.69	0.8	0.82
DeepSeek-R1-Distill-Llama-8B	0.94	0.73	0.82	0.89	0.84	0.87	0.79	0.86	0.87
Llama-3.1-Instruct-8B	0.96	0.71	0.84	0.79	0.94	0.85	0.68	0.83	0.86
Phi-3.5-mini-instruct	0.94	0.68	0.87	0.94	0.94	0.88	0.73	0.87	0.88
Yi-6B-Chat	0.98	0.62	0.87	0.84	0.94	0.86	0.7	0.84	0.86
Mistral-7B-Instruct-v0.3	0.97	0.59	0.87	0.71	0.97	0.77	0.56	0.76	0.78

Table 2: StructFlowBench rated by **GPT-4o**. The left side displays the performance of various models on the five basic structural constraints, while the right side presents their performance on the four key metrics.

bility in refining responses based on iterative user feedback, ensuring more robust handling of complex multi-turn interactions.

Intra-Turn-Constraint-Categorized Performance

The evaluation of LLMs across various constraint dimensions highlights their strengths and weaknesses in following specific instructions. DeepSeek-v3, Gemini-1.5-Pro, and GPT-4o achieve near-perfect satisfaction rates, demonstrating strong capabilities in fine-grained instruction following. Most other models also perform well in rule-based constraints, such as Inverse Constraint, Keyword/Element Constraint, Style Constraint, and Situation Constraint. However, performance drops noticeably in format-related constraints, including Basic Format Constraint, Template Format Constraint, and Quantity Format Constraint, indicating that rigid format adherence remains a significant challenge, even for top-performing models. Overall, while LLMs effectively handle intra-turn constraints, their ability to maintain format consistency remains a key limitation. Addressing this challenge requires further advancements in structured output generation and adherence to strict formatting requirements.

Task-Categorized Performance

We evaluated various models across seven NLP tasks and a mixed task. Unlike the constraint-categorized evaluation, where DeepSeek-v3 led across all metrics, the task-based analysis presents a more nuanced picture. DeepSeek-v3 remains the overall best-performing model but leads only in Fact-based Questions, Professional Writing, Practical Writing, and Casual Chat. Gemini-1.5-Pro outperforms others in Open-ended Questions and Creative Writing, while Claude-3.5-Sonnet achieves the highest performance in Fact-based Questions and Task-oriented Role-playing. Meanwhile, GPT-4o excels in the Mixture task type, reflecting its strength in handling diverse instructions across domains. These results highlight the varying strengths of these top-tier models across different tasks. Following the top four models, GLM-4-9B-Chat, Qwen2.5-14B-Instruct, and Qwen2.5-7B-Instruct maintain consistently strong performance across all tasks. Their stability, combined with their significantly smaller parameter sizes compared to the leading models, makes them highly cost-effective alternatives. In contrast, the remaining models all exhibit noticeable weaknesses in at least one task category, with Mistral-7B-Instruct-v0.3 underperforming across nearly all tasks, revealing a clear performance gap.

4.3 Further Analysis

4.3.1 Complex Scenario Suitability Study

This study aims to verify whether the multi-turn dialogue dataset we have constructed more closely aligns with real-world complex use cases. To achieve this, we designed an experiment to analyze three key properties of dialogue: logical coherence, goal clarity, and transition naturalness. The datasets used in this experiment include our StructFlowBench, three other multi-turn dialogue evaluation datasets (MT-Bench-101, Multi-if, and MT-Eval), and a real-world dialogue dataset, WILDCHAT.

Data Preparation: For each dataset, we randomly selected 50 English multi-turn dialogue samples, ensuring a diverse representation of dialogue types.

Evaluation Protocol: To quantify how well the

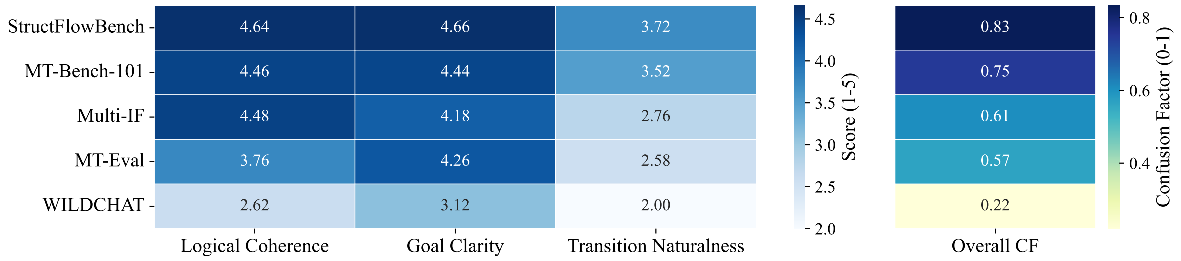


Figure 3: The comprehensive complex scenario evaluation heatmap of five multi-turn dialogue datasets.

dialogues meet complex scenario requirements, we employed GPT-4o for automated scoring. Each dialogue was evaluated based on its performance in the following areas:

- **Logical Coherence:** Evaluates whether the dialogue is logically consistent and free of abrupt or unreasonable shifts.
- **Goal Clarity:** Assesses whether the dialogue clearly communicates the task's goals and ensures both the user's and system's intentions are transparent.
- **Transition Naturalness:** Judges whether transitions between dialogue turns are smooth and natural, without awkward or forced shifts.

Each property was scored on a scale from 1 to 5, where 1 indicates complete failure to meet the expected standard, and 5 represents perfect alignment with complex scenario requirements.

Confusion Factor (CF): To further evaluate the datasets, we introduced the Confusion Factor (CF), which quantifies the proportion of dialogues in each dataset that scored 4 or higher, indicating they were mistakenly perceived as real-world interactions. The CF is calculated as follows:

$$\text{CF} = \frac{\text{Number of dialogues with average score} \geq 4}{\text{Total number of dialogues}},$$

By comparing the CF values of our StructFlowBench dataset with those of others, we can assess whether our dataset outperforms the others in terms of alignment with complex scenarios.

Results and Discussion: The results are presented as a heatmap, as shown in Figure 3. StructFlowBench achieves the highest scores across all three evaluation dimensions, leading with a confusion factor of 0.83. MT-Bench-101, with its comprehensive dialogue generation process and rigorous human proofreading, also produces high-quality dialogues and ranks closely behind with strong scores. In contrast, the WILDCHAT real multi-turn dialogue dataset, containing one million dialogues, exhibits generally low quality. Although we performed preliminary filtering on the WILDCHAT data, such as considering prompt length and dialogue content, the extracted dialogues still failed to meet the ideal quality standards As a result, WILDCHAT performed the worst across the three evaluation dimensions for data-driven simulated scenarios.

4.4 Human Verification

We extracted 30 dialogues from the output of Qwen2-7B-Instruct and invited domain experts to conduct a comprehensive and detailed evaluation of the results. The experts rated the outputs using a binary scoring system. The results showed that the Kappa coefficient between GPT-4o's evaluations and those of the experts was approximately 0.75. This indicates that utilizing advanced LLMs, like GPT-4o, to assess the quality of outputs from other models is a reliable approach, effectively reducing both subjective bias and the time costs associated with relying solely on human evaluation.

5 Conclusion

In this work, we address key limitations in current multi-turn instruction-following research by introducing StructFlowBench, a novel benchmark designed to capture the structural intricacies of complex dialogue scenarios. By incorporating a dual-constraint evaluation system and a six-category structural flow taxonomy, we provide a more comprehensive framework for assessing the logical coherence, goal clarity, and transition naturalness of multi-turn dialogues. Our evaluations of 13 representative LLMs reveal critical insights into the structural processing capabilities of both closed-source and open-source models, offering valuable guidance for future advancements in instruction-following systems. Through StructFlowBench, we lay the foundation for more robust, realistic, and contextually aware dialogue systems.

Limitations

Currently, the structural flow in StructFlowBench is designed with a single linear relationship to facilitate analysis and data generation. For instance, if the third turn dialogue serves as both a recall structure to the first turn and a follow-up structure to the second turn, the current approach retains only the recall relationship while disregarding other structural dependencies. This simplification may limit the comprehensive modeling of hierarchical dialogue structures. Future work should extend the structural flow framework to simultaneously capture multiple coexisting dialogue relationships, thereby providing a more holistic representation of multi-turn dialogue complexity.

Ethics Statement

This study utilizes GPT-4o to generate multi-turn dialogue data and annotate constraints, with manual review to filter out inappropriate content. However, unintended biases in GPT-4o's generation process, as well as potential oversight during human review, may result in residual errors or biases in the dataset. While we have made every effort to ensure data quality and mitigate these issues, completely eliminating them remains challenging. Additionally, since this dataset is publicly available, there is a risk of misuse for model training, which may compromise the validity of our benchmark. Therefore, we encourage the research community to exercise caution when using this dataset and to complement it with other evaluation methods to ensure comprehensive and fair model assessment.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

AI Anthropic. 2024. Claude 3.5 sonnet model card addendum. *Claude-3.5 Model Card*, 3:1–8.

Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, and Wanli Ouyang. 2024. MT-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7421–7454, Bangkok, Thailand. Association for Computational Linguistics.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45.

Haodong Duan, Jueqi Wei, Chonghua Wang, Hongwei Liu, Yixiao Fang, Songyang Zhang, Dahua Lin, and Kai Chen. 2024. BotChat: Evaluating LLMs' capabilities of having multi-turn dialogues. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3184–3200, Mexico City, Mexico. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Qianyu He, Jie Zeng, Wenhao Huang, Lina Chen, Jin Xiao, Qianxi He, Xunzhe Zhou, Jiaqing Liang, and Yanghua Xiao. 2024a. Can large language models understand real-world complex instructions? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18188–18196.

Yun He, Di Jin, Chaoqi Wang, Chloe Bi, Karishma Mandyam, Hejia Zhang, Chen Zhu, Ning Li, Tengyu Xu, Hongjiang Lv, et al. 2024b. Multi-if: Benchmarking llms on multi-turn and multilingual instructions following. *arXiv preprint arXiv:2410.15553*.

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

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. 2024. Follow-Bench: A multi-level fine-grained constraints following benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4667–4688, Bangkok, Thailand. Association for Computational Linguistics.

Wai-Chung Kwan, Xingshan Zeng, Yuxin Jiang, Yufei Wang, Liangyou Li, Lifeng Shang, Xin Jiang, Qun Liu, and Kam-Fai Wong. 2024. MT-eval: A multi-turn capabilities evaluation benchmark for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 20153–20177, Miami, Florida, USA. Association for Computational Linguistics.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. 2024. Infobench: Evaluating instruction following ability in large language models. *arXiv preprint arXiv:2401.03601*.

Yuchong Sun, Che Liu, Kun Zhou, Jinwen Huang, Ruihua Song, Xin Zhao, Fuzheng Zhang, Di Zhang, and Kun Gai. 2024. Parrot: Enhancing multi-turn instruction following for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9729–9750, Bangkok, Thailand. Association for Computational Linguistics.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Xingyao Wang, Zihan Wang, Jiateng Liu, Yangyi Chen, Lifan Yuan, Hao Peng, and Heng Ji. 2023. Mint: Evaluating llms in multi-turn interaction with tools and language feedback. *arXiv preprint arXiv:2309.10691*.

Bosi Wen, Pei Ke, Xiaotao Gu, Lindong Wu, Hao Huang, Jinfeng Zhou, Wenchuang Li, Binxin Hu, Wendy Gao, Jiaxin Xu, et al. 2024. Benchmarking complex instruction-following with multiple constraints composition. *arXiv preprint arXiv:2407.03978*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Guoyin Wang, Heng Li, Jiangcheng Zhu, Jianqun Chen, et al. 2024. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*.

Tao Zhang, Yanjun Shen, Wenjing Luo, Yan Zhang, Hao Liang, Fan Yang, Mingan Lin, Yujing Qiao, Weipeng Chen, Bin Cui, et al. 2024. Cfbench: A comprehensive constraints-following benchmark for llms. *arXiv preprint arXiv:2408.01122*.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. *arXiv preprint arXiv:2405.01470*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric P Xing, et al. 2023a. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. *arXiv preprint arXiv:2309.11998*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023b. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

A Details of Topics and Tasks

- **Topic:** Our dataset is generated across a diverse range of 22 topics, including health, history, science, technology, digital media, automotive, astronomy, geography, lifestyle, literature, physics, finance, stocks, law, humanities, entertainment, music, fashion, art, environment, psychology, and a mixed category that incorporates multiple topics. This broad coverage ensures that our data spans multiple domains, capturing a wide array of fields and areas of interest.

- **Task:** StructFlowBench comprises seven NLP tasks and one mixed-category task, with their exact distribution detailed in Table 3.

Category	#Dialogues
Fact-based Questions	25
Open-ended Questions	20
Practical Writing	26
Creative Writing	21
Professional Writing	21
Casual Chat	15
Task-oriented Role Play	17
Mixture	10
Total	155

Table 3: Task distribution of STRUCTFLOWBENCH dataset.

B Details of Constraints

The distribution of all constraints is detailed in Table 4, with the definitions of intra-turn constraints as follows:

Content Constraint: The response must strictly focus on the specified content scope and avoid any deviation from the topic.

Keyword/Element Constraint: The response must include specific words or elements as required.

Style Constraint: The response must be generated in a specific writing style, such as formal, humorous, poetic, etc.

Basic Format Constraint: The output must adhere to a specified basic format, such as JSON, XML, CSV, Table, Markdown, etc.

Quantity Format Constraint: The response must meet a precise requirement for the number of characters, words, sentences, or paragraphs as specified.

Template Format Constraint: The response must follow a predefined template structure, such as starting with a specific phrase, ending with a certain statement, or using a custom template provided by the user.

Situation Constraint: The response must be tailored to a given scenario or perspective, such as responding from a specific identity or context.

Inverse Constraint: The response must deliberately exclude or avoid certain constraints, such as not containing a specific keyword, not involving a particular element, or not using a certain language style.

C Detailed Results Categorized by Intra-turn Constraints and Task Types

Table 5 presents the intra-turn constraints performance of various models on StructFlowBench, while Table 6 illustrates the task-categorized performance. Additionally, Figure 4 provides a radar chart comparing both perspectives.

D Details of Prompts

Figure 5 to Figure 8 respectively illustrate the intermediate dialogue plan generation template, complete dialogue generation prompt template, constraint extraction prompt template, and GPT-4o evaluation prompt template used in our study.

E Case of Data

Table 7 presents a sample case from StructFlow-Bench.

F Details of Models

All the details about the evaluated models are provided in Table 8.

Follow-up	Refinement	Expansion	Summary	Recall	C1	C2	C3	C4	C5	C6	C7	C8
95	32	156	63	118	505	153	140	105	175	98	83	52

Table 4: The constraints distribution of StructFlowBench. *Follow-up*, *Refinement*, *Expansion*, *Summary*, *Recall* denote the structural constraints. The designations C1 - C8 denote the Constraint types of *Content Constraint, Keyword/Element Constraint, Style Constraint, Basic Format Constraint, Quantity Format Constraint, Template Format Constraint, Situation Constraint, Inverse Constraint*

Model Name	Inverse Constraint	Keyword/Element Constraint	Style Constraint	Situation Constraint	Basic Format Constraint	Quantity Format Constraint	Template Format Constraint	Content Constraint
Deepseek-v3	**1.0**	**1.0**	**1.0**	**1.0**	**0.99**	**1.0**	**0.99**	**1.0**
Gemini-1.5-Pro	**1.0**	0.99	0.99	**1.0**	**0.99**	0.99	**0.99**	0.99
GPT-4o	**1.0**	**1.0**	**1.0**	**1.0**	**0.99**	0.98	**0.99**	**1.0**
Claude-3.5-Sonnet	0.98	0.97	0.99	**1.0**	0.95	0.99	0.94	0.97
GLM-4-9B-Chat	0.98	0.98	0.99	0.96	0.97	0.95	0.95	0.99
Qwen2.5-14B-Instruct	0.96	0.99	0.99	0.95	0.9	0.93	0.92	0.97
Qwen2.5-7B-Instruct	0.96	0.97	0.99	0.99	0.95	0.91	0.88	0.96
Deepseek-R1-Distill-Qwen-7B	0.9	0.89	0.91	0.84	0.82	0.7	0.8	0.83
DeepSeek-R1-Distill-Llama-8B	0.88	0.95	0.9	0.9	0.9	0.84	0.84	0.88
Llama-3.1-Instruct-8B	0.98	0.87	0.92	0.94	0.73	0.79	0.7	0.88
Phi-3.5-mini-instruct	0.94	0.93	0.96	0.96	0.82	0.81	0.8	0.9
Yi-6B-Chat	0.83	0.92	0.91	0.9	0.87	0.65	0.91	0.9
Mistral-7B-Instruct-v0.3	0.88	0.82	0.84	0.9	0.65	0.59	0.56	0.8

Table 5: The intra-turn constraints performance of various models on StructFlowBench.

Model Name	Fact-based Questions	Open-ended Questions	Professional Writing	Practical Writing	Creative Writing	Casual Chat	Task-oriented Role-playing	Mixture
Deepseek-v3	**0.93**	0.96	**0.99**	**0.96**	0.97	**0.98**	0.95	0.97
Gemini-1.5-Pro	0.91	**0.97**	0.96	0.91	**0.98**	0.96	0.95	0.97
GPT-4o	0.92	0.96	0.96	0.95	0.97	0.94	0.92	**0.98**
Claude-3.5-Sonnet	**0.93**	0.95	0.97	0.88	0.94	0.92	**0.97**	0.95
GLM-4-9B-Chat	0.89	0.93	0.96	0.92	0.94	0.95	0.93	0.97
Qwen2.5-14B-Instruct	0.9	0.94	0.93	0.9	0.94	0.91	0.91	0.93
Qwen2.5-7B-Instruct	0.9	0.92	0.89	0.91	0.93	0.93	0.94	0.95
Deepseek-R1-Distill-Qwen-7B	0.77	0.85	0.86	0.82	0.74	0.79	0.8	0.77
DeepSeek-R1-Distill-Llama-8B	0.79	0.9	0.9	0.87	0.86	0.88	0.86	0.83
Llama-3.1-Instruct-8B	0.81	0.88	0.8	0.83	0.84	0.76	0.88	0.88
Phi-3.5-mini-instruct	0.86	0.88	0.86	0.84	0.94	0.86	0.86	0.86
Yi-6B-Chat	0.84	0.9	0.87	0.82	0.82	0.77	0.86	0.8
Mistral-7B-Instruct-v0.3	0.71	0.82	0.72	0.76	0.75	0.73	0.79	0.78

Table 6: Task-categorized performance of various models on StructFlowBench.

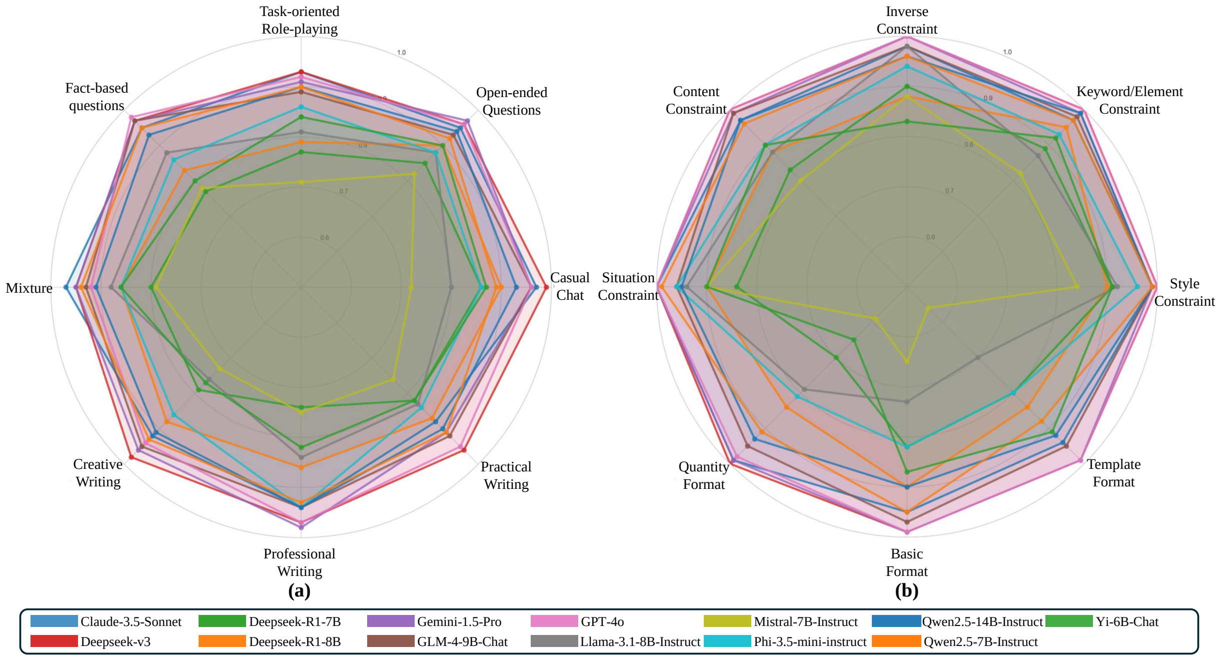


Figure 4: The radar chart of intra-turn-constraint-categorized performance (a) and task-categorized performance (b).

Intermediate Dialogue Plan Generation Template

[Task Description]

Objective:

As a real user, generate appropriate simple multi-round dialogue user prompts based on the given dialogue structure in [Dialogue Structure Template]

Steps to Construct Simple User Prompts based on the given dialogue structure:

1.**Read and Understand the [Background Knowledge] and [Dialogue Structure Template] carefully**

-**Think**:What is the relation between each turn of dialogue?

2.**Set User Purpose**

- Dialogue Topic: {topic}
- Dialogue Type: {task}

-**Consider**: Given the specified dialogue topic and type, reflect on what the purpose of the user engaging in this multi-turn dialogue might be?

-**Action**: Define the overarching purpose of the user engaging in this multi-turn dialogue based on the specified dialogue topic, type. Identify what specific goals the user aims to achieve through this dialogue.

3.**Generate summarized user prompts**

-**Think**:How the user can progressively ask questions through a dialogue process similar to the provided dialogue structure template? What requests would users ask in each turn of dialogue?

-**Action**:Generate the detailed summarized user prompts in each turn of dialogue based on the dialogue structure template. Ensure that the generated summarized prompts are naturally reasonable within the multi-turn dialogue, making the entire conversation coherent and smooth, aligning with the process of a real user engaging in dialogue.

Deliverable: Provide fully constructed summarized prompts following the designated [Output Format] without including extra analysis or commentary.

[Background Knowledge]

Definitions and Scenarios of the Five Basic Structures:

- Follow-up: (Definition)
- Recall: (Definition)
- Expansion: (Definition)
- Summary: (Definition)
- Refinement: (Definition)

[Dialogue Structure Template]

{structure}

[Output Format]

```
{
  "conv_purpose":"<str:The summary of the user's purpose for this multi-turn conversation>"
  "summarized prompts": [
    {
      "name": "c1",
      "description": "<str:detailed summarized user's prompt,clearly reflecting the relationship given in structure template>",
      "explanation": "<str:explain how the summrized prompt follow definition of the given dialogue structure relation in this round>"
    },
    ...
  ]
}
```

Figure 5: Intermediate Dialogue Plan Generation Template

Complete Dialogue Generation Prompt Template

[Task Description]
Main Objective:
Expand the provided summarized user prompts in [Seed Summarized Prompts] into detailed, realistic user prompts with various types of constraints. Ensure these expansions align with the summrized prompts and feel natural, reflecting genuine user inquiries.
Requirements for Constructing Realistic and Constraint-Integrated User Prompts:
- Establish a conversation background that aligns with the user's conversation purpose:{conv_purpose}
- Integrate relevant and reasonable constraints from the [Constraint Guideline] from real human user's needs, embedding these constraints seamlessly into prompts while keeping the conversation flow natural and clear.
- Make sure ever intended constraints are expressed in the user prompt and accurately presented according to their use methods and definition in [Constraint Guideline].
- Adjust the communication style of your expanded prompts to match specified user characteristics:{user_characteristic}
- Answer the user prompt of the current round as a LLM assistant, providing responses that reflect the above requirements.
Deliverable: Provide fully constructed conversation following the designated [Output Format] without including extra analysis or commentary.

[Seed Summarized Prompts]
{smmarized_conv}

[Constraint Guideline]
Constraints are those requests or limiations included in user prompts for guiding LLM provide a better response. Please understand them carefully:
- Inverse Constraint:(Definition)
...
- Keyword/Element Constraint:(Definition)
- Style Constraint:(Definition)

[Output Format]

~~~
```json
{
   "whole_conv":[
      {
         "name":"c1",
         "user prompt":"<str:real user prompt>",
         "assistant answer":"<str:answer to the user prompt as a LLM assistant>"
      },
      ...
   ]
}
```
~~~

Figure 6: Complete Dialogue Generation Prompt Template

Constraint Extraction Prompt Template

~~~
[Task Description]
You are a professional atomic constraint extractor. Your task is to extract as many atomic constraint expressions as possible from the given [user prompt] which is sampled from a multi-round conversation between user an a LLM assistant.
Definition of atomic constraint expression: The smallest unit of description or constraint for the required task within the instruction.
Refer to the list of atomic constraint types and their definitions provided in the [Constraint Extraction Guideline].
Identify both the type of each constraint and its corresponding content from the [user prompt].
Ensure that all constraints are correctly categorized and expressed as questions.

You can refer to these examples:
# Example 1 #
# Example 2 #

**Deliverable**: Provide fully constructed conversation following the designated [Output Format] without including extra analysis or commentary.

[Constraint Extract Guideline]
Constraints are those atomic requests or limiations included in user prompts for guiding LLM provide a better response
- Inverse Constraint:(Definition)
...
- Keyword/Element Constraint:(Definition)
- Style Constraint:(Definition)

[user prompt]
{user_prompt}

[Output Format]
```json
{{
    "constraints":[
        {{
            "type":"<str:constraint type name in [Constraint Extract Guideline]>",
            "content":"<str:the content of the specific constraint included in the user prompt, express as a question>",
            "explanation":"<str:explain why the constraint is classified as the current type.>"
        }},
        ...
    ]
}}
```
~~~

Figure 7: Constraint Extraction Prompt Template

GPT-4o Evaluation Prompt Template

~~~
[Task Description]
You are an exceedingly meticulous and fair judge. Your task is to rigorously evaluate whether the [Current Round LLM Response] strictly adheres to every detail specified in the [Current Round User Prompt], using the provided [Check List] as your guide.
- [Conversation History] provides context from previous rounds of the dialogue.
- [Current Round User Prompt] represents the latest instruction given by the user in the dialogue; each aspect of this prompt must be addressed with exactness and thoroughness.
- [Current Round LLM Response] is the response generated by the language model in accordance with the user's prompt; it must meet all explicit and implicit requirements without exception.
- [Check List] contains specific questions that assess whether the [Current Round LLM Response] meets each detailed requirement outlined in the [Current Round User Prompt]; each item must be scrutinized meticulously.

For each item in the [Check List], answer with 'Yes' if the criterion is met beyond doubt, or 'No' if there is any deviation, ambiguity, or omission. Provide a clear and concise explanation for your judgment, highlighting how the response does or does not meet the criteria. Justify your answer with reference to both the [Current Round User Prompt] and relevant parts of the [Conversation History].

**Deliverable**: Provide judgement following the designated [Output Format] without including extra analysis or commentary.

[Conversation History]
{golden_context}

[Current Round User Prompt]
{cur_user_prompt}

[Current Round LLM Response]
{cur_llm_response}

[Check List]
{check_list}

[Output Format]
```json
{{
    "judge result":[
        {{
            "judgement":"<str:only 'Yes' or 'No', indicating whether the constraint was followed.>",
            "reason":"<str:Provide an explanation for your judgment basis, i.e., the reasoning behind determining whether the constraint was followed>"
        }},
        ...
    ]
}}
```
~~~

Figure 8: GPT-4o Evaluation Prompt Template

User purpose	The user aims to develop a financial plan for a fictional character by interacting with the assistant as a financial advisor.The user wants to learn about different music genres and styles to enhance their personal music knowledge and broaden their music listening experience.
Structure	"source": "c1","target": "c2","relation": "follow-up" "source": "c1","target": "c3","relation": "recall" "source": "c3","target": "c4","relation": "unrelatedness" "source": "c4","target": "c5","relation": "refinement"
Summarized Prompts	"c1" : "The user asks the assistant, role-playing as a financial advisor, to provide a general strategy for a young professional who wants to start saving for retirement." ... "c5": "The user modify the detail level in last round's prompt to request a deeper dive into the unique instruments used in each genre for better understanding of their sounds."
Complete Dialogue	"name": "c1", "user prompt": "Imagine I am a young professional entering the workforce. As my financial advisor, could you...", "assistant answer": "Certainly! Here's a comprehensive strategy for..." ... "name": "c5", "user prompt": "In order to delve deeper into the musical intricacies ... Please format the response as a table and ..." "assistant answer": "Certainly! Here is a detailed examination of the unique instruments associated with each genre in a table format:..."
Check Lists	"name":"c1" "Situation Constraint":"Is the response given from the perspective of a financial advisor?" "Keyword/Element Constraint":"Does the response include specific keywords such as... ?" ... "name":"c5" "Basic Format Constraint":"Is the response formatted as a table?" "Refinement Constraint":"Is the c5 conversation a refinement of c4 conversation?"

Table 7: An example of synthetic data.

Model		**Model Link**
GPT	GPT-4o	https://platform.openai.com/docs/models#gpt-4o
Claude	Claude-3.5-Sonnet	https://docs.anthropic.com/en/docs/about-claude/models
Gemini	Gemini-1.5-Pro	https://ai.google.dev/gemini-api/docs/models/gemini?hl=en#gemini-1.5-pro
Deepseek	DeepSeek-v3 DeepSeek-R1-Distill-Qwen-7B DeepSeek-R1-Distill-Llama-8B	https://huggingface.co/deepseek-ai/DeepSeek-V3 https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-7B https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-8B
Qwen	Qwen2.5-14B-Instruct Qwen2.5-7B-Instruct	https://huggingface.co/Qwen/Qwen2.5-14B-Instruct https://huggingface.co/Qwen/Qwen2.5-7B-Instruct
GLM	GLM-4-9B-Chat	https://huggingface.co/THUDM/glm-4-9b-chat
Yi	Yi-6B-Chat	https://huggingface.co/01-ai/Yi-6B-Chat
LLAMA	Llama-3.1-8B-Instruct	https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct
Mistral	Mistral-7B-Instruct-v0.3	https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3
Phi	Phi-3.5-mini-instruct	https://huggingface.co/microsoft/Phi-3.5-mini-instruct

Table 8: Model Links.