

# MARS-Bench: A Multi-turn Athletic Real-world Scenario Benchmark for Dialogue Evaluation

Anonymous ACL submission

## Abstract

Large Language Models (LLMs), e.g. ChatGPT, have been widely adopted in real-world dialogue applications. However, LLMs’ robustness, especially in handling long complex dialogue sessions, including frequent motivation transfer, sophisticated cross-turn dependency, is criticized all along. Nevertheless, no existing benchmarks can fully reflect these weaknesses. We present **MARS-Bench**, a **Multi-turn Athletic Real-world Scenario Dialogue Benchmark**, designed to remedy the gap. MARS-Bench is constructed from play-by-play text commentary so to feature realistic dialogues specifically designed to evaluate three critical aspects of multi-turn conversations: ultra multi-turn, interactive multi-turn, and cross-turn tasks. Extensive experiments on MARS-Bench also reveal that closed-source LLMs significantly outperform open-source alternatives, explicit reasoning significantly boosts LLMs’ robustness on handling long complex dialogue sessions, and LLMs indeed face significant challenge when handling motivation transfer and sophisticated cross-turn dependency. Moreover, we provide mechanistic interpretability on how attention sinks due to special tokens lead to LLMs’ performance degradation when handling long complex dialogue sessions based on attention visualization experiment in Qwen2.5-7B-Instruction.

## 1 Introduction

Large Language Models (LLMs) have made remarkable advances, enabling fluent interactions with users, even on sessions with more than 30 turns, messy information and unnatural motivation transfer. However, LLMs’ robustness has long been criticized when handling users’ shifts between information-seeking questions, reasoning tasks, and creative content generation in same dialogue sessions without clear task boundaries, while no existing evaluations fully reflect the weakness.



Data Format

Time	Play	
4:49	Ivica Zubac defensive rebound	47 39
4:34	Ivica Zubac makes 2-foot dunk (James Harden assists)	49 39
4:22	Devin Booker misses 26-foot three point pullup jump shot	49 39
4:20	Bogdan Bogdanovic defensive rebound	49 39
...		
3:27	Ivica Zubac makes alley oop dunk shot (James Harden assists)	51 41
3:07	Nick Richards makes pullup jump shot (Devin Booker assists)	51 43
3:01	James Harden makes driving layup	53 43
...		
1:03	Ivica Zubac makes 5-foot two point shot (James Harden assists)	60 46
...		
35.1	James Harden makes 28-foot three point step back jumpshot	63 46
1.2	Ivica Zubac makes 7-foot two point shot (James Harden assists)	65 46

Team's Players		Player Name	Score	Assists	.....
Phoenix Suns	Devin Booker	0	1		
	Ivica Zubac	8	0		
	James Harden	3	4		
	Nick Richards	2	0		
LA Clippers	Ivica Zubac	8	0		
	James Harden	3	4		
	Bogdan Bogdanovic	2	0		
	...	...	...		

Figure 1: **Overview of the Data Format.** Each sample represents a single game and includes: (1) play-by-play records; (2) team rosters; and (3) player statistics. The first two are used as model input, and the third for answer verification.

We point out that LLMs must (i) retrieve evidence dispersed across distant, sometimes dozens-of-turn-old, utterances, and (ii) reason jointly over these fragments while adapting to frequent task switches to handle the aforementioned scenarios. As these conversational scenarios grow increasingly complex, there is a clear need for robust evaluation protocols that can systematically assess LLMs’ ability to understand, reason, and respond coherently across conversational turns.

However, many benchmarks focus on short conversations, provide the full dialogue history upfront rather than revealing it turn by turn, and rarely test reasoning over information scattered across distant turns (Bai et al., 2024; Zheng et al., 2023; Kwan et al., 2024; Sun et al., 2024; Sirdeshmukh et al., 2025). Agent-based benchmarks (Liu et al., 2024; Ye et al., 2025) explore complex task settings, but often lack alignment with real-world dialogue scenarios, making it difficult to assess how models adapt in natural. These limitations highlight the lack of comprehensive benchmarks for interactive

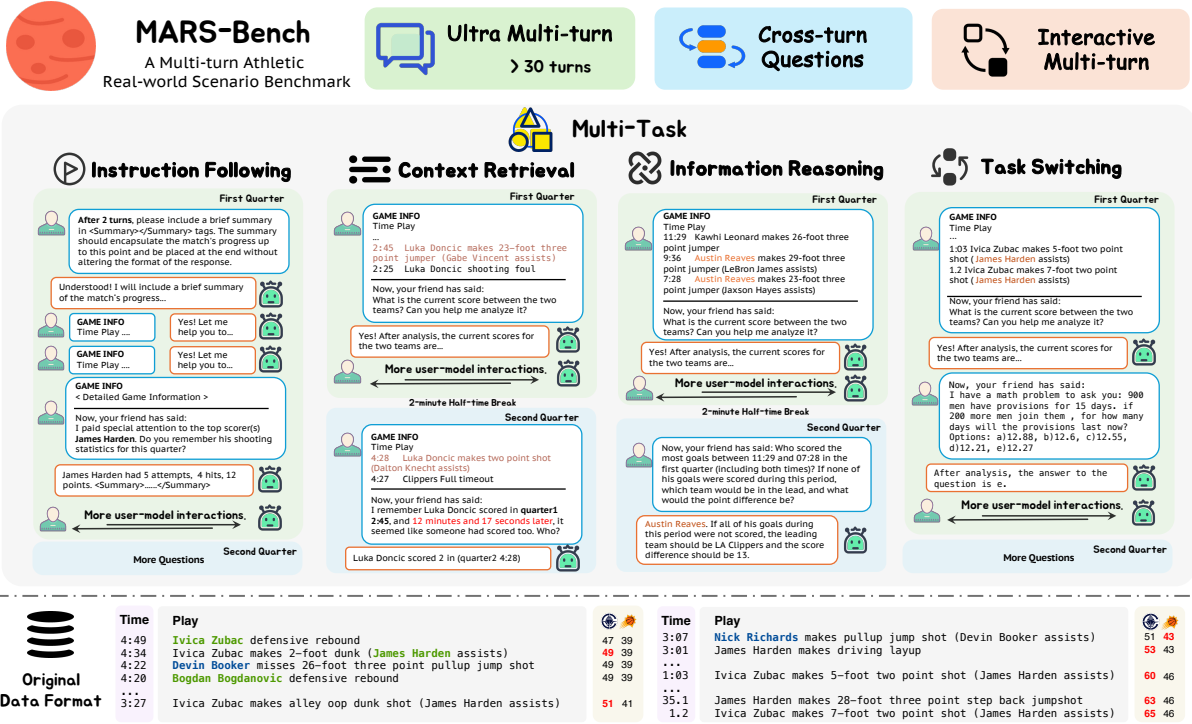


Figure 2: **Overview of MARS-Bench.** MARS-Bench is constructed from real-world play-by-play sports data and supports ultra multi-turn (UMT), cross-turn tasks (CTT), and interactive (IMT) dialogue scenarios. It comprises four core task categories—Instruction Following, Context Retrieval, Information Reasoning, and Task Switching—each illustrated with representative dialogue examples. The bottom section shows the structured game data format.

multi-turn (IMT) dialogue, cross-turn tasks (CTT), and ultra multi-turn (UMT) scenarios.

To address gaps in existing research, we propose **MARS-Bench**, a multi-turn dialogue benchmark constructed from real-world play-by-play sports data. MARS-Bench emphasizes three key features: **Ultra Multi-turn Dialogues** with over 30 turns per instance, capturing instruction shifts and contextual evolution; **Cross-turn Questions** that require reasoning over non-adjacent information; and **Interactive Multi-turn Generation**, where LLMs must respond at every turn, reflecting realistic user interactions and frequent task switches. Built on top of these settings, MARS-Bench defines four core tasks: instruction following, context retrieval, information reasoning, and task switching. These tasks jointly enable comprehensive evaluation of multi-turn and multi-task dialogue capabilities. Extensive experiments on state-of-the-art LLMs with MARS-Bench reveal that:

- **Closed-Source LLMs Leaders:** Closed-source models, e.g. Claude-3.7-Sonnet-Thinking, substantially outperform open-source alternatives, particularly in tasks requiring deep contextual understanding and multi-turn reasoning.

- **LLMs Benefit from Explicit Reasoning:** Models employing explicit reasoning mechanisms (*System 2*) consistently achieve higher accuracy and stability, whereas heuristic-driven (*System 1*) approaches falter with increased task complexity.

- **LLMs Struggle with Multi-Turn Dialogue:** Models face notable difficulties in instruction following, retaining context across multiple turns, and managing cumulative errors from incremental predictions, highlighting critical bottlenecks in long-range contextual memory and structured inference processes.

## 2 MARS-Bench: Design and Construction

This section is organized as follows: Data Collection and Processing (2.1), Task Categories (2.2), and Benchmark Statistics (2.3).

### 2.1 Data Collection and Processing

Building on the play-by-play textual game data, we construct a three-stage data pipeline comprising data collection, question construction, and manual verification, as illustrated in Figure 3.

**Data Collection** We collect English play-by-play records and post-game statistics for the NBA and

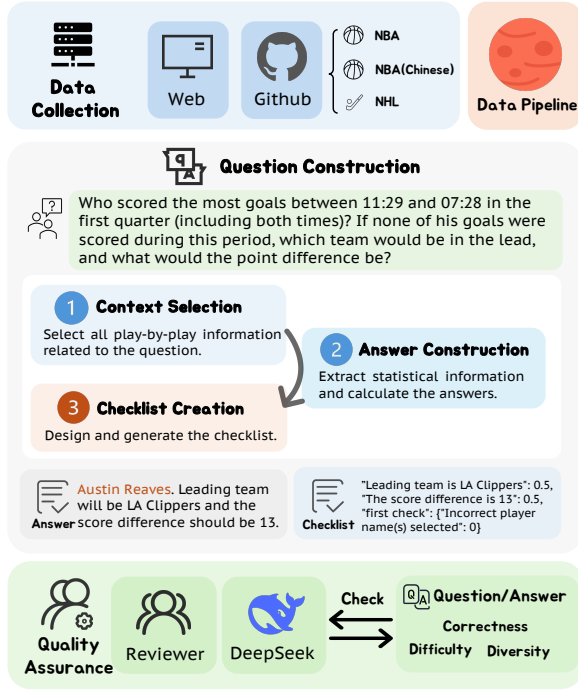


Figure 3: **Overview of the Data Construction Pipeline.** The pipeline comprises three main stages: (1) **Data Collection**, in which sports data is gathered from online sources; (2) **Question Construction**, involving the generation of (Question, Answer, Checklist) triples through context selection, answer formulation, and checklist creation; (3) **Quality Assurance**, where each (Question, Answer, Checklist) triple is reviewed for correctness, difficulty, and diversity.

NHL from ESPN, and supplement them with Chinese NBA data from Hupu, a major sports community in China, to increase both linguistic and domain diversity. The raw data is standardized into structured formats for multi-turn dialogue modeling, with each sample represent a complete sports game. Figure 1 illustrates the structure of the collected data.

**Question Construction** Each question is manually designed according to a specific task type. Relevant information is extracted from the associated play-by-play context entries using regular expressions, which is then used to generate the corresponding answer and verification checklist automatically. These components serve as the basis for evaluating model responses. This stage yields a set of (*question*, *answer*, *checklist*) triplets.

**Quality Assurance** All generated (*Question*, *Answer*, *Checklist*) triplets are reviewed by both human annotators and a LLM, specifically DeepSeek-V3-0324, to ensure their correctness,

difficulty, and diversity.

## 2.2 Task Categories

MARS-Bench defines four core task categories: **Instruction Following**, **Context Retrieval**, **Information Reasoning**, and **Task Switching**. Instruction Following (IF) and Context Retrieval (CR) assess a model’s ability to handle cross-turn queries in multi-turn scenarios, while Information Reasoning (IR) and Task Switching (TS) focus on interactive reasoning. These categories evaluate diverse reasoning types and dynamic dialogue behaviors, including tracking evolving instructions, retrieving distant context, aggregating scattered information, and adapting to abrupt task shifts.

Each game is divided into periods (e.g., NBA quarters, NHL periods), with each period split into five score-tracking turns that form a multi-turn dialogue. Category-specific questions are inserted at appropriate points based on the four task categories. Models must respond incrementally and maintain coherence throughout.

Figure 2 provides abstract illustrations of the task categories, while Table 1 outlines their descriptions, subtask distributions, and instance counts per period. Full task examples and question placements are detailed in Appendix E.

## 2.3 Benchmark Statistics

MARS-Bench consists of 120 games, with an average of 33.42 dialogue turns per game. Each of the four task categories comprises 30 games, evenly distributed across three sports domains: 15 NBA (English), 10 NBA (Chinese), and 5 NHL(English). Figure 4 visualizes the distribution of tasks and domains, where the outer ring indicates task categories and the inner ring reflects domain composition.

Table 2 presents a comparison between MARS-Bench and existing multi-turn dialogue benchmarks. In contrast to previous datasets, MARS-Bench offers substantially longer dialogues, incorporates real user–model interactions, and covers a more diverse set of tasks spanning multiple languages and domains.

## 3 Experiments

### 3.1 Experiment Setup

**Prompting Setting** All models are evaluated under a zero-shot prompting strategy. Task instructions and dialogue scenarios are specified using

Table 1: **Task categories and subtask types in MARS-Bench**, along with the number of subtasks generated per game segment. Each segment corresponds to a natural period in sports games—three periods in NHL and four quarters in NBA (excluding overtime).

Task Type	Description	Sub-task	Sub-task Description	Questions per Period
Instruction Following	Follow turn-specific instructions with format constraints.	Fixed-format Single-turn Response	Follow the format specified for the current dialogue turn.	1
		Turn-conditioned Prompted Formatting	Adapt the response format according to system instructions at each turn.	8
		Turn-conditioned Inferred Formatting	Adjust the response format based on instructions inferred from prior dialogue turns.	1
Context Retrieval	Locate and retrieve factual information from previous dialogue turns.	Anchored Event Retrieval	Given a time anchor and interval, retrieve a specific event.	2
		Interval-based Event Retrieval	Given a start and end time, retrieve events of a specific type.	1
Information Reasoning	Aggregate and reason over distributed contextual information.	Current Score Tracking	Provide the current score for both teams.	1 (last period)
		Score Lead Fluctuation Detection	Identify the number and timing of score lead changes between the two teams within a given time period.	1
		Player Performance Impact Analysis	Given a time span, analyze how a change in a player’s performance affected the game situation.	2
Task Switching	Handle abrupt interleaving of unrelated queries.	In-context Reasoning Query	Ask questions related to the match.	3
		Out-of-context Math Query	Ask unrelated mathematical questions from MathQA (Amini et al., 2019).	3

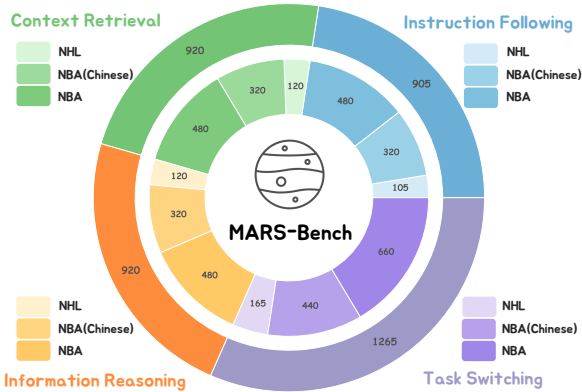


Figure 4: **Statistics of MARS-Bench**. The outer ring shows the distribution of the four task categories, and the inner ring indicates the corresponding sports domains: NBA (English), NBA (Chinese), and NHL. Numbers on the chart represent the total number of model interaction turns per task.

carefully designed prompt templates tailored to each task category, as detailed in Appendix E.

**Evaluation Metrics** We adopt the LLM-as-a-judge approach to evaluate model outputs. To implement this, and following the procedure described in 2.1, we construct a corresponding checklist that specifies key assessment criteria. The judge model then assigns scores to the predicted responses based on these checklists. The prompt tem-

Table 2: Comparison of MARS-Bench with Other Multi-turn Dialogue Benchmarks

Benchmark	Real Interaction	Cross-turn	Multi-task
MT-Bench	✗	✗	✗
MT-Bench-101	✗	✓	✓
MultiChallenge	✗	✓	✓
MARS-Bench(Ours)	✓	✓	✓

Benchmark	Avg. Turns	Total Queries	Language
MT-Bench	1–2	80	English
MT-Bench-101	3	4208	English
MultiChallenge	5	1365	English
MARS-Bench(Ours)	33	4010	English, Chinese

plate used for evaluation is provided in Appendix D. In our experiments, we use DeepSeek-v3-0324 as the judge model, and this evaluation configuration is consistently applied to all experiments, including those presented in the Discussion section.

### 3.2 Experiment Results

Table 3 presents the evaluation results of various representative models on MARS-Bench. Based on these results, we summarize the following key observations.

**LLMs Struggle in Complex Multi-turn Dialogues.** Even top models achieve around 70 points, with performance decreasing as dialogue



Table 3: **Performance of different models on MARS-Bench.** The benchmark includes four task categories: Instruction Following (IF), Context Retrieval (CR), Information Reasoning (IR), and Task Switching (TS). Shaded cells indicate the best performance, **bold** indicates the second-best, and underlined the third-best. For more details about the scores of math questions, please refer to the Appendix B.

Model	Reasoning	Open Source	Overall	IF	CR	IR	TS
Gemini-2.5-Pro (Google, 2025b)	✓	✗	72.44	65.08	87.06	70.92	<b>66.72</b>
Claude-3.7-Sonnet-Thinking (Anthropic, 2025)	✓	✗	<b>62.29</b>	43.28	71.51	<b>66.98</b>	67.38
o1-1217 (OpenAI, 2024b)	✓	✗	59.62	53.09	64.48	<u>62.63</u>	58.28
Gemini-2.5-Flash (Google, 2025a)	✓	✗	59.22	45.96	<b>77.76</b>	52.93	<u>60.23</u>
GPT-4.5-Preview (OpenAI, 2025b)	✗	✗	53.33	<u>55.52</u>	66.65	50.43	40.74
Doubao-1.5-Pro-Thinking (ByteDance, 2025b)	✓	✗	52.62	51.99	55.64	52.17	50.69
Grok3 (xAI, 2025)	✓	✗	51.21	<b>61.19</b>	<u>73.91</u>	33.89	35.87
o4-mini-0416 (OpenAI, 2025d)	✓	✗	47.13	47.48	61.26	39.74	40.03
DeepSeek-R1 (DeepSeek-AI et al., a)	✓	✓	45.42	53.04	49.23	40.01	39.40
Claude-3.5-Sonnet (Anthropic, 2024)	✗	✗	43.17	44.45	52.09	39.03	37.09
o3-mini-high (OpenAI, 2025c)	✓	✗	42.15	53.16	50.68	32.58	32.18
Claude-3.7-Sonnet (Anthropic, 2025)	✗	✗	41.21	34.13	59.77	36.80	34.15
o3-mini-medium (OpenAI, 2025c)	✓	✗	39.25	52.17	42.66	32.31	29.84
Doubao-1.5-Pro-32k (ByteDance, 2025a)	✗	✗	38.88	42.80	46.81	33.63	32.28
DeepSeek-V3-0324 (DeepSeek-AI et al., b)	✓	✓	37.31	45.34	46.18	27.70	30.02
GPT-4o-1120 (OpenAI, 2024a)	✗	✗	35.83	39.28	31.26	36.69	36.12
Gemini-2.0-Flash (Google, 2024)	✗	✗	35.61	48.56	39.24	26.71	27.92
Qwen3-235B (Qwen Team, 2025c)	✓	✓	34.88	42.47	39.42	28.03	29.59
DeepSeek-V3-1226 (DeepSeek-AI et al., c)	✓	✓	33.16	37.23	37.08	28.71	29.63
GPT-4.1-mini-0414 (OpenAI, 2025a)	✗	✗	31.23	40.23	30.17	26.39	28.13
Qwen2.5-Max (Qwen Team, 2025a)	✗	✗	30.41	39.77	31.90	26.76	23.22
Qwen2.5-72B-Instruct (Qwen Team, 2025b)	✗	✓	29.21	38.38	30.41	24.06	23.97
GLM-Z1-Air (BigModel, 2025)	✓	✓	25.84	35.75	22.49	24.36	20.76

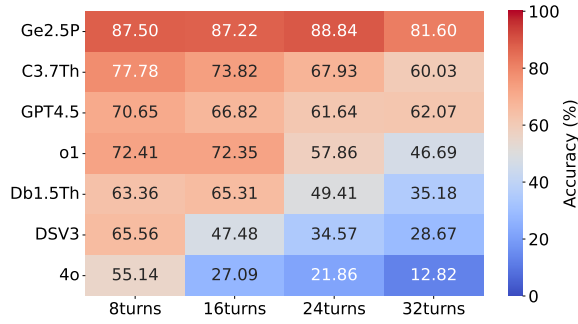
turns increase, shown in 4.1, highlighting limitations in handling extended multi-turn interactions. Lower scores on IF, IR, and particularly TS tasks further underscore deficiencies in cross-turn context management and interactive scenarios.

**Closed-Source Models Lead in Complex Multi-Turn Scenarios.** In challenging multi-turn dialogue tasks, closed-source models consistently outperform open-source counterparts. For example, Google’s Gemini-2.5-Pro achieves a 72.44 overall on MARS-Bench under complex contextual and reasoning requirements, while the top open-source DeepSeek-R1 reaches just 45.42. Open-source models—though flexible—often lack the scale and targeted optimization needed to excel in intricate information reasoning and task-switching.

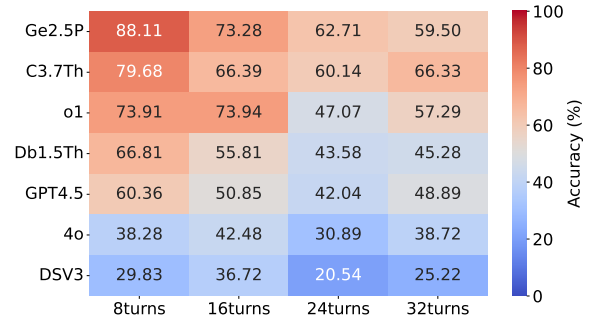
**Reasoning Models demonstrate greater performance.** Models equipped with chain-of-thought reasoning tend to engage more deliberate, System 2-style inference and decision-making processes.

As a result, they exhibit higher consistency and correctness across multi-turn dialogue tasks. In contrast, models that rely on System 1-style heuristic generation are more susceptible to variations in task complexity and context, leading to comparatively weaker overall performance. For example, DeepSeek-R1 achieves an overall MARS-Bench score of 45.42, outperforming DeepSeek-V3 (37.31) by 8.11 points, and even scores over 12 points higher on information reasoning (40.01 vs. 27.70).

**Models perform worse on the Instruction Following task.** Both reasoning-enhanced and standard models demonstrate relatively poor performance on the instruction following task. Analysis reveals that current models struggle to track turn-level structures as required by system prompts. In particular, they often fail to produce the correct output in the specified dialogue turn, suggesting limitations in their ability to align generation be-



(a) Accuracy trends in the Context Retrieval task



(b) Accuracy trends in the Information Reasoning task

Figure 5: **Accuracy degradation in CR and IR tasks with increasing input turns.** Subfigures (a) and (b) show results for the Context Retrieval (CR) and Information Reasoning (IR) tasks, respectively. Db1.5Th refers to Doubao-1.5-Pro-Thinking, Ge2.5P to Gemini-2.5-Pro, and C3.7Th to Claude-3.7-Sonnet-Thinking. See Appendix A for the full list of model names and abbreviations.

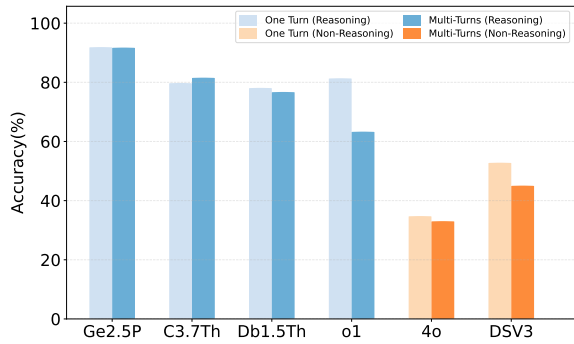


Figure 6: **Ablation on input format: multi-turn vs. single-turn.** Most models perform worse with multi-turn inputs, suggesting that input fragmentation hinders reasoning. Top models like Gemini-2.5-Pro and Claude-3.7-Sonnet-Thinking remain robust. Additional results are provided in Appendix C.

havior with round-dependent instructions.

## 4 Discussion

We organize our discussion around the following research questions: (1) whether increasing the number of dialogue turns harms model performance; (2) whether cross-turn context hinders reasoning; and (3) whether interactive multi-turn generation degrades LLMs performance.

### 4.1 LLMs Struggle with More Turns

#### Increasing Turns Leads to Lower Accuracy.

As an additional analysis setting, we break down the main experimental results by interaction rounds to examine how performance evolves over turns. As shown in Figure 5a and Figure 5b, model accuracy in context retrieval and information reasoning tasks tends to decline in the later stages of multi-turn interactions.

**Special Tokens Consume Attention in Multi-turn Contexts.** Motivated by the observed degradation in later turns, we further investigate whether the number of interaction rounds, independent of information content, contributes to the performance drop. We conduct an ablation study using identical play-by-play records presented in two formats: a 20-turn dialogue and a single-turn concatenation. As shown in Figure 6, most models perform worse in the multi-turn setting, except for top models like Gemini-2.5-Pro.

To further illustrate this effect, we use a mechanistic interpretability approach to visualize attention in Qwen2.5-7B-Instruct, as shown in Figure 15 (Appendix F). The multi-turn format introduces more special tokens, which absorb a notable portion of attention (e.g., “<lim\_end>”), reducing attention efficiency and contributing to performance degradation.

### 4.2 LLMs Fail Cross-Turn Context

**Distributed Information Reduces Focus on Relevant Context** We split an equal amount of play-by-play game records into different turn lengths ( $x = 1, 10, 20$ ) and evaluated models on quarter-level statistical questions. As shown in Figure 8a, accuracy generally declines with more turns, except for Gemini-2.5-Pro, which maintains or slightly improves its performance.

To gain further insight into the model’s attention patterns, we visualize attention in Qwen2.5-7B-Instruct. Figure 7 shows how the model’s attention to key information changes across settings, revealing that longer dialogues impair its ability to attend effectively to relevant content.

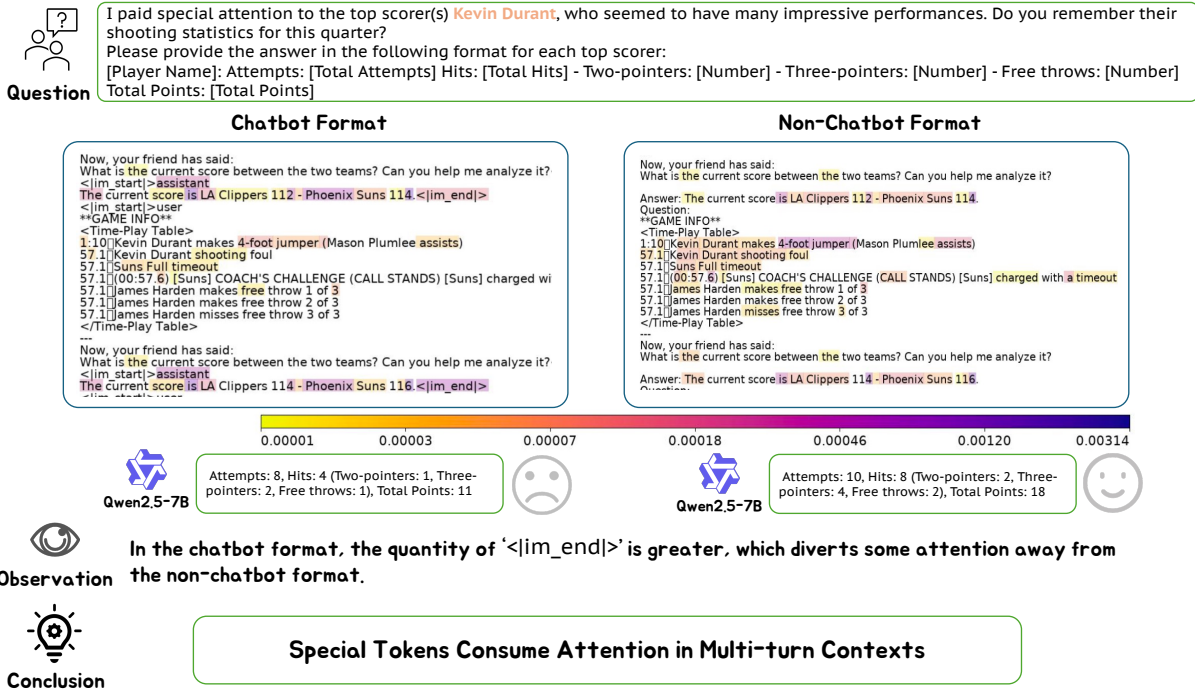


Figure 7: **Attention visualization in Qwen2.5-7B-Instruct.** Attention to key content drops from 0.000714 (1 turn) to 0.000274 (20 turns), suggesting degraded focus in longer dialogues, indicating a 2.6× decay in attention to key content.

## Distant Context Weakens Retrieval Accuracy

We evaluate models’ long-range recall ability by asking questions about the first quarter after each subsequent quarter. As shown in Figure 8b, accuracy declines as the retrieval distance increases, with substantial variation across models. While Gemini-2.5-Pro maintains consistent accuracy across settings, most models (e.g., DeepSeek-V3-0324) exhibit substantial performance degradation with increasing retrieval distance.

### 4.3 LLMs Underperform in Interaction

**Error Accumulation Degrades Interactive Performance** We partition the play-by-play records of each quarter into turn settings of varying lengths ( $x = 1, 10, 20$ ), where the model predicts the current score at each turn. The total information remains fixed, but more turns reduce per-turn complexity. As shown in Figure 9, most models improve with more interaction turns, but o3-mini-high and o3-mini-medium exhibit inconsistent trends. To better understand these inconsistencies and how performance evolves over turns, we conduct a turn-level analysis under the 20-turn setting. As illustrated in Figure 10, strong models maintain stable accuracy across turns, whereas models with solid reasoning abil-

ity, such as o3-mini-high and o3-mini-medium, are hindered by early errors that accumulate and degrade overall performance.

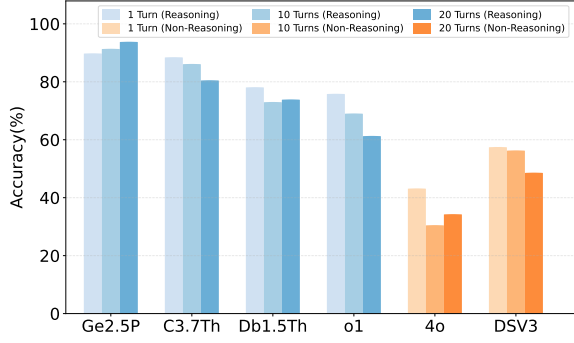
## 5 Conclusion

We introduce MARS-Bench, a benchmark constructed from real-world play-by-play sports data to evaluate LLMs in complex multi-turn dialogue settings. It defines four task types: instruction following, context retrieval, information reasoning, and task switching, enabling systematic and fine-grained assessment of long-context dialogue capabilities. Experimental results suggest that models employing explicit reasoning strategies tend to perform more consistently, although all models exhibit persistent challenges in instruction alignment and context retention. Further analysis indicates that dialogue depth, input structure, and the accumulation of errors across turns can significantly affect model performance. MARS-Bench provides a realistic and focused benchmark for advancing research on multi-turn dialogue understanding.

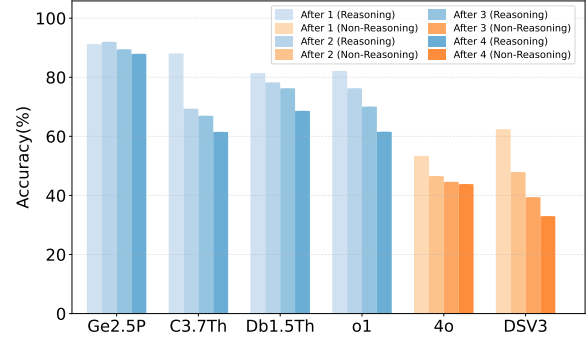
## 6 Related Work

### Multi-turn Dialogue Evaluation Benchmarks

Multi-turn dialogue capability is a key research area for large language models. Early benchmarks



(a) Performance under different turn lengths with the same content ( $x = 1, 10, 20$ ).



(b) Performance under increasing retrieval distances.

Figure 8: **Ablation study on cross-turn context.** (a) Splitting identical content across more dialogue turns ( $x = 1, 10, 20$ ) results in reduced performance. (b) Accuracy decreases as models are required to recall first-quarter information after each section (“After 1” indicates the question is posed immediately following Q1).

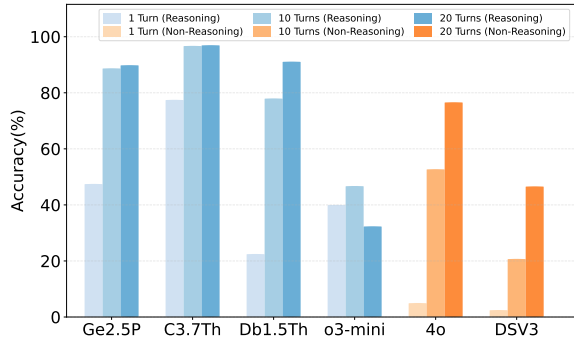


Figure 9: **Ablation on Interaction Turns: Effect on Overall Accuracy.** Model performance on score updates under different interaction turns (1, 10, 20). Most models exhibit significant improvement with increasing turns.

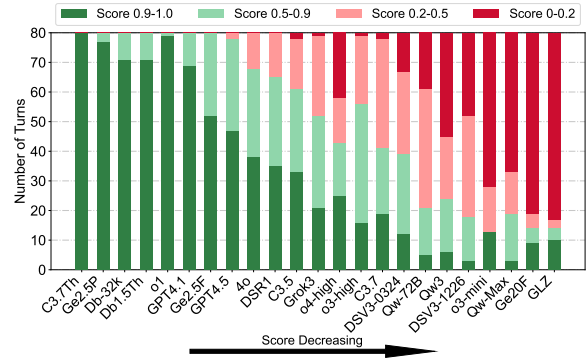


Figure 10: **Ablation on Interaction Turns: Error Accumulation Degrades Interactive Performance.** In multi-turn settings, early errors by the model often accumulate and negatively impact final performance.

such as MT-Bench (Zheng et al., 2023) and MT-Eval (Kwan et al., 2024) primarily focused on short-turn dialogues and basic instruction following. As model capabilities have improved, these evaluations have become less effective at distinguishing model performance. MT-Bench++ (Sun et al., 2024) expanded the dialogue length to eight turns, while MultiChallenge (Sirdeshmukh et al., 2025) introduced various task forms with five-turn dialogues. MINT (Wang et al., 2024) added user feedback and tool usage, increasing the interaction complexity. Despite progress in task coverage and interactivity, most benchmarks still rely on synthetic data and static dialogue settings, limiting their ability to reflect the dynamic evolution of context across turns. There remains a lack of systematic evaluation for key capabilities such as cross-turn reasoning and information tracking.

**Agent-Centric Evaluation of Interactive and Ultra-Long Multi-Turn Dialogues** Recent studies have shifted toward longer and more complex evaluation frameworks to better simulate real-world interactions. OpenAI introduced AlreadySaidThat and TrackTheState (OpenAI, 2023) to assess historical consistency and cross-turn reasoning. LongMemEval (Wu et al., 2024) tests long-term information retention, while LTM-Benchmark (Castillo-Bolado et al., 2024) evaluates task-switching in long-form dialogues. AgentBench (Liu et al., 2024) and RealWebAssist (Ye et al., 2025) expand evaluations to complex environments like database and web tasks. However, a unified framework that covers Interactive Multi-turn (IMT), Cross-turn Tasks (CTT), and Ultra Multi-turn (UMT) dialogues, addressing long-range dependency, task switching, and real-data interaction, is still lacking.



## Limitations

While MARS-Bench offers a structured and realistic setting for evaluating multi-turn dialogue, several limitations remain:

- **Domain specificity:** The focus on sports scenarios may limit generalizability to open-domain or everyday dialogues.
- **Modality constraints:** The benchmark is limited to text and does not include multimodal inputs such as vision or speech.
- **Evaluation method:** Checklist-based automatic scoring with LLM judges may miss subtle issues in coherence, style, or pragmatics.

Future work may explore broader domains, multimodal inputs, and human evaluation to improve generalizability and coverage.

## References

- Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. In *NAACL-HLT (1)*, pages 2357–2367. Association for Computational Linguistics.
- Anthropic. 2024. Claude-3.5-Sonnet: A large language model. <https://www.anthropic.com/news/claude-3-5-sonnet>. Accessed: May 2025.
- Anthropic. 2025. Claude-3.7-Sonnet: A large language model. <https://www.anthropic.com/news/claude-3-7-sonnet>. Accessed: May 2025.
- 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 *ACL (1)*, pages 7421–7454. Association for Computational Linguistics.
- BigModel. 2025. GLM-Z1-air: A large language model. <https://open.bigmodel.cn/dev/api/Reasoning-models/glm-z1>. Accessed: May 2025.
- ByteDance. 2025a. Doubao-1.5-Pro-32k: A large language model. <https://www.volcengine.com/docs/82379/1554678>. Accessed: May 2025.
- ByteDance. 2025b. Doubao-1.5-Pro-Thinking: A large language model. <https://www.volcengine.com/docs/82379/1536428>. Accessed: May 2025.
- David Castillo-Bolado, Joseph Davidson, Finlay Gray, and Marek Rosa. 2024. Beyond prompts: Dynamic conversational benchmarking of large language models. In *NeurIPS*.

- DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. a. *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*. Accessed: May 2025.
- DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. b. *DeepSeek-V3 Technical Report*. Released: March 2025; Accessed: May 2025.
- DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. c. *DeepSeek-V3 Technical Report*. Released: December 2024; Accessed: May 2025.
- Google. 2024. Gemini-2.0-Flash: A large language model. <https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-0-flash>. Accessed: May 2025.
- Google. 2025a. Gemini-2.5-Flash: A large language model. <https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-flash>. Accessed: May 2025.
- Google. 2025b. Gemini-2.5-Pro(preview 05-06): A large language model. <https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-pro>. Accessed: May 2025.
- 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 *EMNLP*, pages 20153–20177. Association for Computational Linguistics.
- Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, and 3 others. 2024. Agentbench: Evaluating llms as agents. In *ICLR*. OpenReview.net.
- OpenAI. 2023. Openai evals: A framework for evaluating llms. <https://github.com/openai/evals>. Accessed: 2025-05-20.
- OpenAI. 2024a. GPT-4o-1120: A large language model. <https://platform.openai.com/docs/models/gpt-4o>. Accessed May: 2025.
- OpenAI. 2024b. o1-1217: A large language model. <https://platform.openai.com/docs/models/o1>. Accessed: May 2025.

OpenAI. 2025a. GPT-4.1-mini-0414: A large language model. <https://platform.openai.com/docs/models/gpt-4.1-mini>. Accessed: May 2025.

OpenAI. 2025b. GPT-4.5-Preview: A large language model. <https://platform.openai.com/docs/models/gpt-4.5-preview>. Accessed: May 2025.

OpenAI. 2025c. o3-mini: A large language model. <https://platform.openai.com/docs/models/o3-mini>. Accessed: May 2025.

OpenAI. 2025d. o4-mini: A large language model. <https://platform.openai.com/docs/models/o4-mini>. Accessed: May 2025.

Qwen Team. 2025a. Qwen-Max: A large language model. <https://huggingface.co/spaces/Qwen/Qwen2.5-Max-Demo>. Accessed: May 2025.

Qwen Team. 2025b. Qwen2.5-72B-Instruct: A large language model. <https://huggingface.co/Qwen/Qwen2.5-72B-Instruct>. Accessed: May 2025.

Qwen Team. 2025c. Qwen3-235B: A large language model. Accessed: May 2025.

Ved Sirdeshmukh, Kaustubh Deshpande, Johannes Mols, Lifeng Jin, Ed-Yeremai Cardona, Dean Lee, Jeremy Kritiz, Willow Primack, Summer Yue, and Chen Xing. 2025. Multichallenge: A realistic multi-turn conversation evaluation benchmark challenging to frontier llms. *CoRR*, abs/2501.17399.

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 *ACL (1)*, pages 9729–9750. Association for Computational Linguistics.

Xingyao Wang, Zihan Wang, Jiateng Liu, Yangyi Chen, Lifan Yuan, Hao Peng, and Heng Ji. 2024. MINT: evaluating llms in multi-turn interaction with tools and language feedback. In *ICLR*. OpenReview.net.

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

xAI. 2025. Grok3: A large language model. <https://docs.x.ai/docs/models#models-and-pricing>. Accessed: May 2025.

Suyu Ye, Haojun Shi, Darren Shih, Hyokun Yun, Tanya Roosta, and Tianmin Shu. 2025. Realwebassist: A benchmark for long-horizon web assistance with real-world users. *arXiv preprint arXiv:2504.10445*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhaghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *NeurIPS*.

## A Model Abbreviations and Full Names

A complete mapping between model abbreviations and their full names is presented in Table 4.

Table 4: List of model abbreviations and their corresponding full names.

Abbreviation	Full Model Name
4o	GPT-4o-1120
C3.5	Claude-3.5-Sonnet
C3.7	Claude-3.7-Sonnet
C3.7Th	Claude-3.7-Sonnet-Thinking
Db-32k	Doubao-1.5-Pro-32k
Db1.5Th	Doubao-1.5-Pro-Thinking
DSR1	DeepSeek-R1
DSV3	DeepSeek-V3-0324
DSV3-1226	DeepSeek-V3-1226
Ge20F	Gemini-2.0-Flash
Ge2.5F	Gemini-2.5-Flash
Ge2.5P	Gemini-2.5-Pro
GLZ	GLM-Z1-Air
GPT-4.5	GPT-4.5-Preview
GPT4.1	GPT-4.1-mini-0414
Grok3	Grok3
o1	o1-1217
o3-high	o3-mini-high
o3-mini	o3-mini-medium
o4-high	o4-mini-0416
Qw-72B	Qwen2.5-72B-Instruct
Qw3	Qwen3-235B
Qw-Max	Qwen2.5-Max

## B Additional Experiment Data

In the TS task, models handle both context-relevant sports tasks and unrelated math problems. While the Table 3 presents only the performance on sports tasks, the full results including math problems are shown in Table 5.

## C Detailed Ablation Study Results

This section presents detailed data for each ablation study, along with visualizations similar to those shown in the Section 4. Specific ablation results for different models are provided in Table 6.

### C.1 Ablation on Input Format

In the Section 4.1, we analyzed the impact of input formats on dialogue performance using several representative models. Here, we provide bar charts

Table 5: Performance of different models on various task types. Additional results are provided for unrelated subtasks (e.g., mathematics) within the Task Switching (TS) category.

Model	Overall	IF	CR	IR	TS	
					Sports Games	Other Tasks
Gemini-2.5-Pro (Google, 2025b)	72.44	65.08	87.06	70.92	<b>66.72</b>	89.57
Claude-3.7-Sonnet-Thinking (Anthropic, 2025)	<b>62.29</b>	43.28	71.51	<b>66.98</b>	67.38	86.96
o1-1217 (OpenAI, 2024b)	59.62	53.09	64.48	<u>62.63</u>	58.28	<u>87.55</u>
Gemini-2.5-Flash (Google, 2025a)	59.22	45.96	<b>77.76</b>	52.93	<u>60.23</u>	87.54
GPT-4.5-Preview (OpenAI, 2025b)	53.33	<u>55.52</u>	66.65	50.43	40.74	65.80
Doubao-1.5-Pro-Thinking (ByteDance, 2025b)	52.62	51.99	55.64	52.17	50.69	<b>88.12</b>
Grok3 (xAI, 2025)	51.21	<b>61.19</b>	<u>73.91</u>	33.89	35.87	57.97
o4-mini-0416 (OpenAI, 2025d)	47.13	47.48	61.26	39.74	40.03	79.71
DeepSeek-R1 (DeepSeek-AI et al., a)	45.42	53.04	49.23	40.01	39.40	85.80
Claude-3.5-Sonnet (Anthropic, 2024)	43.17	44.45	52.09	39.03	37.09	87.25
o3-mini-high (OpenAI, 2025c)	42.15	53.16	50.68	32.58	32.18	87.54
Claude-3.7-Sonnet (Anthropic, 2025)	41.21	34.13	59.77	36.80	34.15	87.25
o3-mini-medium (OpenAI, 2025c)	39.25	52.17	42.66	32.31	29.84	87.54
Doubao-1.5-Pro-32k (ByteDance, 2025a)	38.88	42.80	46.81	33.63	32.28	72.46
DeepSeek-V3-0324 (DeepSeek-AI et al., b)	37.31	45.34	46.18	27.70	30.02	85.80
GPT-4o-1120 (OpenAI, 2024a)	35.83	39.28	31.26	36.69	36.12	39.71
Gemini-2.0-Flash (Google, 2024)	35.61	48.56	39.24	26.71	27.92	68.41
Qwen3-235B (Qwen Team, 2025c)	34.88	42.47	39.42	28.03	29.59	79.71
DeepSeek-V3-1226 (DeepSeek-AI et al., c)	33.16	37.23	37.08	28.71	29.63	73.91
GPT-4.1-mini-0414 (OpenAI, 2025a)	31.23	40.23	30.17	26.39	28.13	75.36
Qwen2.5-Max (Qwen Team, 2025a)	30.41	39.77	31.90	26.76	23.22	68.41
Qwen2.5-72B-Instruct (Qwen Team, 2025b)	29.21	38.38	30.41	24.06	23.97	76.81
GLM-Z1-Air (BigModel, 2025)	25.84	35.75	22.49	24.36	20.76	86.38

for the experimental results of all models (see Figure 11).

## C.2 Ablation on Cross-turn Context

In Section 4.2, we analyzed the impact of cross-turn context on model performance from two dimensions: first, different turn lengths with the same content, and second, the retrieval distances between the queried information and the current turn. Representative experimental results of selected models were presented. In this section, we provide detailed visualization results for all models: the trend of model performance with respect to the turn of information retrieval is shown in Figure 12, while the trend of model performance relative to the information distance is illustrated in Figure 13.

## C.3 Ablation on Interaction Turns

In Section 4.3, we analyzed the impact of interaction turns on model performance. Here, we provide detailed Figure 14 illustrating the results.

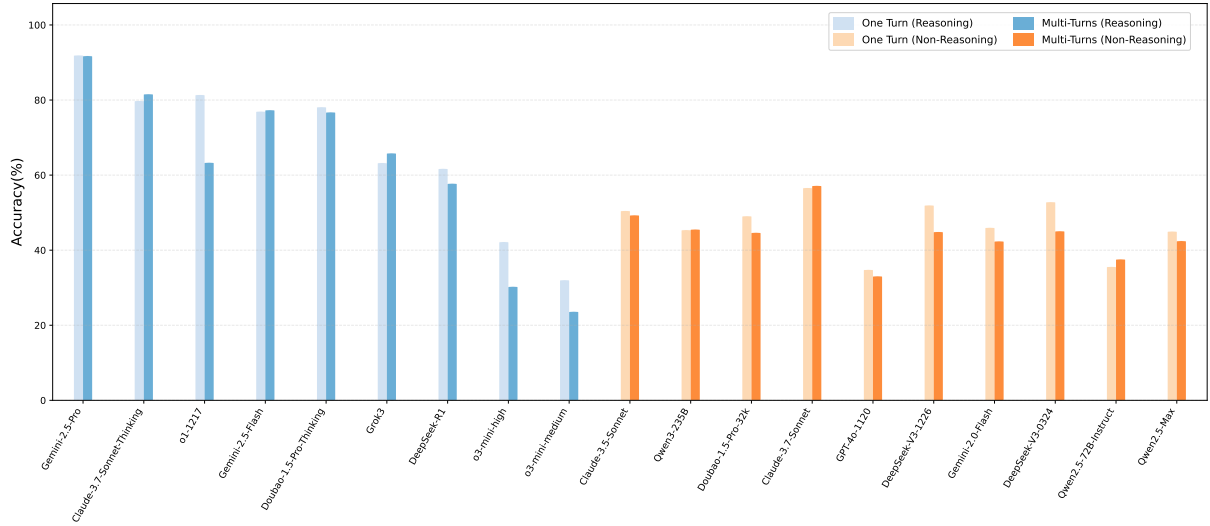


Figure 11: Detailed Ablation Results on Input Format

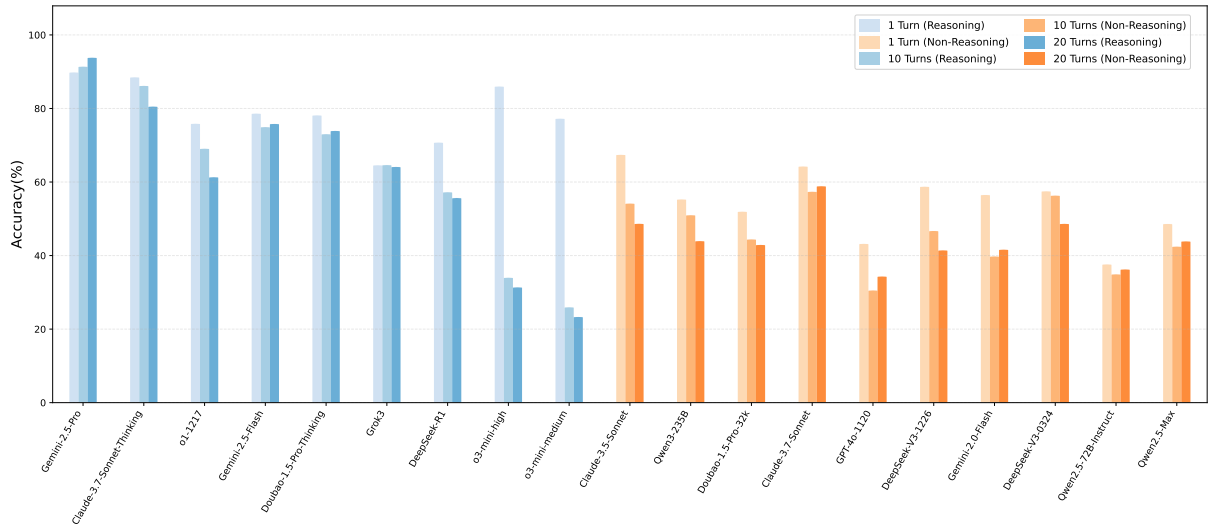


Figure 12: Detailed Results of Ablation on Turn Lengths with Identical Content

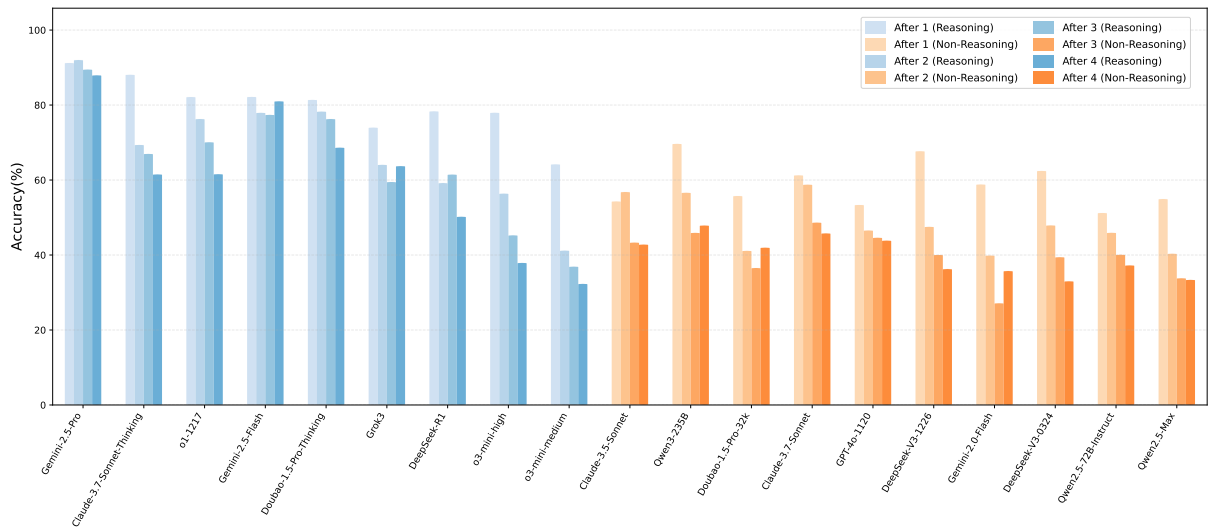


Figure 13: Detailed Ablation Results on Information Retrieval Distance



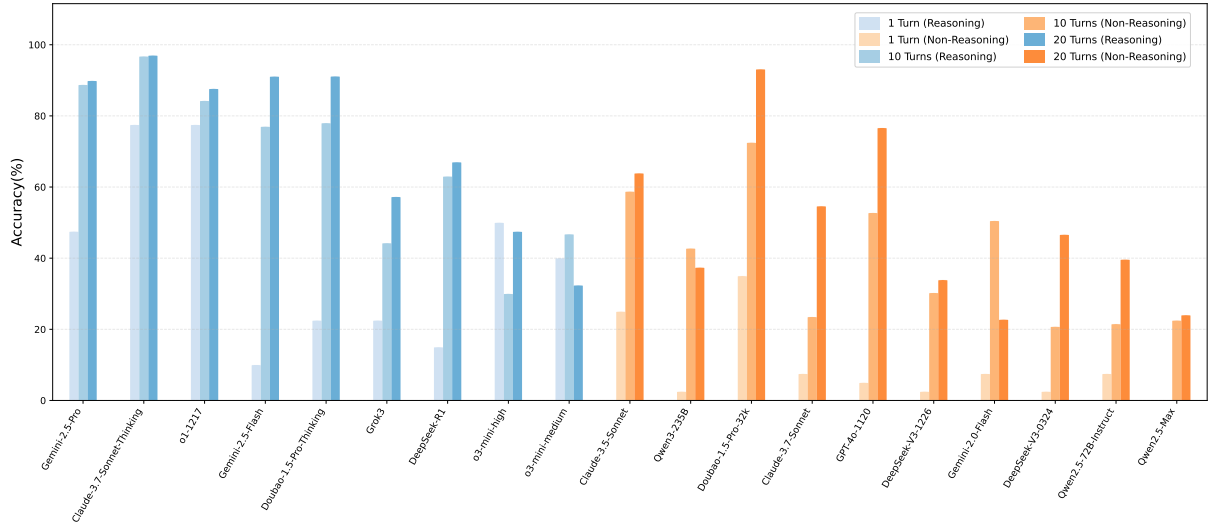


Figure 14: Detailed Ablation Results on Interaction Turns

Table 6: Detailed Ablation Data for Different Models.

Model	Ablation Interaction Turns			Ablation Turn Length			Ablation Input Format		Ablation Retrieval Distance			
	1 turn	10 turns	20 turns	1 turn	10 turns	20 turns	Long Text	Multi-turn	After 1 turn	After 2 turns	After 3 turns	After 4 turns
Gemini-2.5-Pro	47.50	88.75	89.87	89.83	91.40	93.83	91.92	91.75	91.25	92.03	89.50	87.96
Claude-3.7-Sonnet-Thinking	77.50	96.75	97.00	88.50	86.17	80.54	79.78	81.58	88.11	69.39	67.00	61.53
o1-1217	77.50	84.25	87.63	75.87	69.08	61.33	81.38	63.33	82.17	76.30	70.11	61.59
Gemini-2.5-Flash	10.00	77.00	91.09	78.63	74.96	75.82	76.96	77.33	82.20	77.96	77.41	81.03
Doubao-1.5-Pro-Thinking	22.50	78.00	91.13	78.15	73.04	73.92	78.13	76.75	81.41	78.29	76.30	68.67
Grok3	22.50	44.25	57.25	64.58	64.63	64.13	63.25	65.83	74.02	64.09	59.50	63.74
DeepSeek-R1	15.00	63.00	67.00	70.75	57.25	55.67	61.71	57.75	78.36	59.22	61.49	50.24
o3-mini-high	50.00	30.00	47.48	86.00	34.00	31.38	42.20	30.29	77.98	56.41	45.28	37.92
o3-mini-medium	40.00	46.75	32.38	77.25	25.96	23.33	32.04	23.63	64.22	41.21	36.93	32.33
Claude-3.5-Sonnet	25.00	58.75	63.88	67.42	54.17	48.67	50.48	49.32	54.33	56.83	43.35	42.82
Qwen3-235B	2.50	42.75	37.38	55.29	51.00	43.96	45.38	45.54	69.68	56.64	45.94	47.90
Claude-3.7-Sonnet	7.50	23.50	54.63	64.25	57.38	58.88	56.60	57.17	61.29	58.80	48.68	45.80
GPT-4o-1120	5.00	52.75	76.63	43.21	30.54	34.33	34.79	33.07	53.38	46.58	44.64	43.87
Doubao-1.5-Pro-32k	35.00	72.50	93.13	51.96	44.42	42.92	49.08	44.67	55.77	41.14	36.56	41.99
DeepSeek-V3-1226	2.50	30.25	33.88	58.75	46.71	41.44	51.96	44.88	67.72	47.55	40.05	36.28
Gemini-2.0-Flash	7.50	50.50	22.75	56.50	39.79	41.63	46.00	42.38	58.85	39.89	27.16	35.75
DeepSeek-V3-0324	2.50	20.75	46.62	57.50	56.33	48.65	52.83	45.08	62.46	47.93	39.45	33.02
Qwen2.5-72B-Instruct	7.50	21.50	39.63	37.62	34.92	36.25	35.58	37.58	51.24	45.96	40.10	37.26
Qwen2.5-Max	0.00	22.50	24.00	48.62	42.46	43.88	45.00	42.46	54.96	40.38	33.83	33.40

## D Metrics

This section primarily describes the judge model, including its system prompt as well as input examples.

### System Prompt for the Judge Model

You are a meticulous Grader tasked with evaluating the factual accuracy and completeness of a large language model's 'prediction' regarding a sports game. Your evaluation must be based *strictly* on the provided 'checklist'. The 'reference\_answer' is provided as an ideal response example for context but scoring relies solely on the checklist items.

### Evaluation Process: 1. **Analyze Checklist:** Carefully examine the 'checklist'. It contains specific facts or pieces of information ("fact") expected in the prediction, each associated with points ('points'). It might also contain a 'first\_check' section with overriding conditions.

2. **Compare Prediction to Checklist:** For each item "fact": points in the main checklist: Determine if the specific 'fact' is accurately present in the 'prediction'. Reasonable paraphrasing is acceptable if the core meaning and data are identical to the fact stated in the checklist item.

If the fact is present and correct in the prediction, award the corresponding 'points'.

3. **Calculate Initial Score:** Sum the points awarded for all correctly matched checklist items.

4. **Apply 'first\_check' Overrides (If Applicable):** Examine the 'first\_check' section of the checklist, if present. This section typically contains key-value pairs like {"Reason for Zero Score": 0}.

Evaluate if any of the listed "Reasons for Zero Score" accurately describe a fundamental flaw present in the 'prediction'. A common example is if the prediction significantly misunderstands the core subject of the query (e.g., providing stats for the wrong player or team entirely when a specific one was asked about).

If **any** condition listed in 'first\_check' is determined to be true based on your assessment of the 'prediction', the **final score must be 0**. This overrides any points accumulated from the main checklist items.

5. **Determine Final Score:** The final score is the summed points from the main checklist (Step 3), potentially overridden to 0 if a 'first\_check' condition was met (Step 4). The score will be a float between 0.0 and 1.0.

### Output Format: Provide your evaluation in the following format:

[Scoring analysis]: - Briefly explain which checklist items were found correctly in the 'prediction' and which were missing or incorrect, referencing the awarded points.

- If a 'first\_check' condition was met and resulted in a score of 0, **clearly state which specific condition** (the key from the 'first\_check' dictionary, e.g., "Player name incorrect") was triggered and briefly explain why it applies to the prediction.

- Keep the analysis concise (around 100-150 words).

- Conclude with: "In conclusion, the prediction should receive x points" (where x is the final float score).

[Score]: x points

[JSON]: ```json

```
{
  "answer_score": [[score]]
}
```

```

—  
**\*Example Scenario (Illustrative):\***

**\*Checklist:** ```json

```
{
  "checklist": [
    "Player A scored 10 points": 0.4,
    "Player A had 3 assists": 0.3,
    "Player B had 5 rebounds": 0.3
  ],
  "first_check": [
    "Player name incorrect": 0
  ]
}
```

### An Input Example for the Judge Model

<Prediction>

Here's the detailed breakdown of lead changes in the first quarter:

In the First quarter, the score order changed **\*2 times\***. The specific time and the leading team are as follows:

1. **At 10:05**, the Suns' score (2 points) had surpassed the Spurs' score (0 points) after Kevin Durant's 14-foot pullup jump shot. **(This was Phoenix's first and only lead until late in the quarter)**

2. **At 5:46**, the Spurs' score (9 points) had surpassed the Suns' score (7 points) after Jeremy Sochan's driving dunk (assisted by Chris Paul). **(San Antonio then maintained the lead for the rest of the quarter, though Phoenix briefly tied it at 16-16 before falling behind again)**

**Note:** The Suns briefly tied the game at 16-16 (4:12) but did not retake the lead, so this does not count as a lead change. The only two clear lead changes occurred at the times listed above.

</Prediction>

<Reference Answer>

In the First quarter, the score order changed 1 times. The specific time and the leading team are as follows:

11:30: San Antonio Spurs's score had surpassed Phoenix Suns's

</Reference Answer>

<Checklist>

{"The score order changed 1 times": 0.2, "At 11:30, San Antonio Spurs's score had surpassed Phoenix Suns's": 0.8}

</Checklist>

## E Task Categories

This section provides the system prompts corresponding to four distinct task types, and illustrative examples of questions, answers, and checklists for each associated sub-task.

### E.1 System Prompt

The following is a basic system prompt example that can be directly applied to the Context Retrieval (CR), Information Reasoning (IR), and Task Switching (TS) tasks. The Instruction Following (IF) task, however, introduces additional requirements built upon this base prompt.

#### System Prompt for CR, IR and TS Tasks

You have made plans with friends to watch a sports match, but your home TV is broken. You can only follow the match via text live streaming on your phone. During the match, you will chat based on the text updates. As a knowledgeable friend, you must:

- Accurately track match progress based on the text live stream;
- Naturally engage with your friend's comments and provide accurate answers;
- Ensure all quoted match data is absolutely correct;
- Only infer from objective live text information, no irrelevant content;
- Ensure your responses relate to your friend's comments, whether you're explaining, analyzing, or casually chatting, and always focus on the current match.

Your main goal is to match your friend's conversation rhythm and create an engaging, reliable dialogue experience.

Match: Phoenix Suns vs San Antonio Spurs

Team info: Phoenix Suns: Kevin Durant, Royce O'Neale, Nick Richards, Tyus Jones, Devin Booker, Ryan Dunn, Mason Plumlee, Bol Bol, Bradley Beal, Grayson Allen San Antonio Spurs: Harrison Barnes, Bismack Biyombo, Chris Paul, De'Aaron Fox, Devin Vassell, Julian Champagnie, Jeremy Sochan, Keldon Johnson, Stephon Castle

#### System Prompt in Instruction Following

ou have made plans with friends to watch a sports match, but your home TV is broken. You can only follow the match via text live streaming on your phone. During the match, you will chat based on the text updates. As a knowledgeable friend, you must:

- Accurately track match progress based on the text live stream;
- Naturally engage with your friend's comments and provide accurate answers;
- Ensure all quoted match data is absolutely correct;
- Only infer from objective live text information, no irrelevant content;
- Ensure your responses relate to your friend's comments, whether you're explaining, analyzing, or casually chatting,

and always focus on the current match.

**Each reply must:**

- Be in English and less than 100 words; - Be divided into three paragraphs, each ending with the same rhyme sound, and only three paragraphs are included, excluding the following tags.
- Begin with two parts on the same line:
- First, identify your friend's intent and print '[Update Score]' or '[Other Questions]';
- Second, insert a tag according to this cycle:
- Insert '[Tags A]' on the 1st, 5th, 9th... replies;
- Insert '[Tags B]' on the 3rd, 7th, 11th... replies;
- No tag on the 2nd, 4th, 6th, 8th... replies.

Your main goal is to match your friend's conversation rhythm and create an engaging, reliable dialogue experience.

— Match: Phoenix Suns vs San Antonio Spurs

Team info: Phoenix Suns: Kevin Durant, Royce O'Neale, Nick Richards, Tyus Jones, Devin Booker, Ryan Dunn, Mason Plumlee, Bol Bol, Bradley Beal, Grayson Allen San Antonio Spurs: Harrison Barnes, Bismack Biyombo, Chris Paul, De'Aaron Fox, Devin Vassell, Julian Champagnie, Jeremy Sochan, Keldon Johnson, Stephon Castle

—

## E.2 Score Tracking Task

To simulate a realistic multi-turn dialogue, we define the core interaction between the user and the model as a text-based live broadcast excerpt from a game, provided by the user. The model is subsequently required to extract and report the score information of the teams. An illustrative example is provided below:

### Score Tracking Question

Question:

**GAME INFO**

<Time-Play Table>

12:00\tNick Richards vs. Bismack Biyombo (Tyus Jones gains possession)

11:47\tKevin Durant misses 16-foot jumper

11:44\tBismack Biyombo defensive rebound

11:30\tDe'Aaron Fox makes 22-foot three point jumper (Chris Paul assists)

11:08\tTyus Jones misses 22-foot three point jumper

**other play-by-play records.....**

9:32\tRoyce O'Neale defensive rebound

</Time-Play Table>

Now, your friend has said:

What is the current score between the two teams? Can you help me analyze it?"

Answer:

"The current score is Phoenix Suns 2 - San Antonio Spurs 8."

Checklist:

```
{  
  "Phoenix Suns's score is 2": 0.5,  
  "San Antonio Spurs's score is 8": 0.5  
}
```

## E.3 Instruction Following

In the IF (Instruction Following) task, formatting requirements are derived from both the system prompt and the user's in-dialogue instructions. At each turn, model responses are assessed for adherence to these requirements. Checklist scores are calculated based on the number of formatting constraints in the current question, with points distributed as evenly as possible across different sources. Each response is assigned a maximum score of 1.0.



### E.3.1 Fixed-format Single-turn Response

In the Fixed-format Single-turn Response sub-task, the model is required to follow turn-specific instructions, which may include constraints on rhythm, formatting, or other stylistic aspects provided within the dialogue.

#### Fixed-format Single-turn Response

Question:

**GAME INFO**

<Time-Play Table>

3:40\tJulian Champagnie defensive rebound

3:36\tDevin Vassell misses running pullup jump shot

3:33\tDevin Vassell offensive rebound

3:29\tKeldon Johnson makes 23-foot three point jumper (Julian Champagnie assists)

3:14\tGrayson Allen bad pass (Stephon Castle steals)

**other play-by-play records.....**

1:51\tSuns defensive team rebound

</Time-Play Table>

Now, your friend has said:

I think Phoenix Suns is awesome, I want to write a sentence as support, can you help me write one? Write a sentence no longer than 20 words. Starting from the fifth word, the first letters of consecutive words must be sequentially composed of "Team name initial" plus "KING", and the order must match each letter, with no interruptions in between. If the team name is "NewYork Yankees", the first letters of consecutive words must be "NYKING".

**Two turns after this one**, after your normal answer, add a short summary in <Summary></Summary> tags. The summary should cover the match progress so far, under 50 words, placed at the end without affecting the answer's flow or format.

Answer: "Open answer, please evaluate according to the checklist."

Checklist: {

"Adheres to English less than 100 words": 0.1,

"Adheres to three paragraphs, each ending with the same rhyme sound (excluding tags)": 0.1,

"Adheres to beginning with correct bracket ([Other Questions])": 0.05,

"Excludes tags as expected (No tag expected)": 0.25,

"Generated sentence starts the required letter sequence ('PSKING') from the 5th word": 0.25,

"Generated sentence uses consecutive words to form the complete required letter sequence ('PSKING')": 0.25

}

### E.3.2 Turn-conditioned Prompted Formatting

Turn-conditioned Prompted Formatting is a sub-task where the response format is specified by the system prompt and remains consistent throughout the entire dialogue. To evaluate whether the model can distinguish between different dialogue turns, we assign different formatting requirements to different turns. The specific formatting instructions can be found in System Prompt E.2. Moreover, in this sub-task, simpler formatting instructions (e.g., enclosing the response within a <Question> tag) are allocated lower scores, while more complex requirements (e.g., using distinct tags for different turns) are assigned higher weights to reflect their increased difficulty.

#### Turn-conditioned Prompted Formatting

Question:

**GAME INFO**

<Time-Play Table>

12:00\tNick Richards vs. Bismack Biyombo (Tyus Jones gains possession)

11:47\tKevin Durant misses 16-foot jumper

11:44\tBismack Biyombo defensive rebound

11:30\tDe' Aaron Fox makes 22-foot three point jumper (Chris Paul assists)

11:08\tTyus Jones misses 22-foot three point jumper

11:05\tNick Richards offensive rebound  
 11:01\tRoyce O'Neale misses 24-foot three point jumper  
 10:58\tNick Richards offensive rebound  
 10:56\tNick Richards misses dunk  
 10:56\tDevin Vassell defensive rebound  
 10:48\tChris Paul makes 15-foot pullup jump shot  
 10:37\tKevin Durant misses 25-foot three point jumper  
 10:37\tDe'Aaron Fox defensive rebound  
 10:24\tHarrison Barnes makes 22-foot three point jumper (Chris Paul assists)  
 10:08\tDevin Booker misses 13-foot pullup jump shot  
 10:08\tSuns offensive team rebound  
 10:05\tKevin Durant makes 14-foot pullup jump shot  
 9:48\tDe'Aaron Fox misses driving floating jump shot  
 9:46\tNick Richards defensive rebound  
 9:41\tDevin Booker misses two point shot  
 9:39\tChris Paul defensive rebound  
 9:36\tHarrison Barnes misses 23-foot three point shot  
 9:32\tRoyce O'Neale defensive rebound  
 </Time-Play Table>

Now, your friend has said:

What is the current score between the two teams? Can you help me analyze it?

Answer: "Open answer, please evaluate according to the checklist"

Checklist: {

"Adheres to English less than 100 words": 0.2,

"Adheres to three paragraphs, each ending with the same rhyme sound (excluding tags)": 0.2,

"Adheres to beginning with correct bracket ([Other Questions])": 0.1,

"Includes correct tag ([Tags A] expected)": 0.5

}

### E.3.3 Turn-conditioned Inferred Formatting

Turn-conditioned Inferred Formatting focuses on evaluating whether the model complies with turn-specific formatting instructions that span multiple dialogue turns. Specifically, in one turn, the model may be instructed to append a summary of the game—wrapped in a <Summary> tag and limited to fewer than 50 words—two turns later, after the regular response. The checklist assigns individual scores to each of these requirements. Detailed instructions are provided within the multi-turn dialogues (see Question Case E.2), and examples of the questions and corresponding checklists are shown below:

#### Turn-conditioned Inferred Formatting

Question:

**GAME INFO**

<Time-Play Table>

End of Quarter / Half-time break

</Time-Play Table>

Now, your friend has said:

Wow, the First quarter was really exciting! Both teams performed very well, and the game was very intense. I paid special attention to the top scorer(s) Stephon Castle, De'Aaron Fox, who seemed to have many impressive performances. Do you remember their/his shooting statistics for this quarter?

Please provide the answer in the following format for each top scorer:

[Player Name]: Attempts: [Total Attempts]

Hits: [Total Hits]

- Two-pointers: [Number]

- Three-pointers: [Number]

- Free throws: [Number]

Total Points: [Total Points]

Question: "Open answer, please evaluate according to the checklist"

```

Checklist: {
  "Adheres to English less than 100 words": 0.1,
  "Adheres to three paragraphs, each ending with the same rhyme sound (excluding tags)": 0.1,
  "Adheres to beginning with correct bracket ([Other Questions])": 0.05,
  "Excludes tags as expected (No tag expected)": 0.25,
  "Includes <Summary> tags (Gate condition)": 0.0,
  "Summary content is less than 50 words": 0.25,
  "Summary content accurately reflects game history up to this point": 0.25
}

```

## E.4 Context Retrieval

The system prompt for CR task can be found in Prompt D.2.

### E.4.1 Anchored Event Retrieval

Anchored Event Retrieval requires the model to identify a specific event (e.g., a player's score) given a reference timestamp and a time interval. The evaluation assigns 0.5 points for correctly identifying the time and 0.5 points for retrieving the correct event. No points are awarded if the wrong player is identified. Each game period includes 2 such questions.

#### Anchored Event Retrieval

##### Question:

##### GAME INFO

<Time-Play Table>

End of Quarter / Half-time break

</Time-Play Table>

Now, your friend has said:

Wow, the Second quarter was really exciting! Both teams performed very well, and the game was very intense.

I remember LeBron James scored in (quarter2 9:27), and 8.0 minutes and 38.9 seconds later, it seemed like someone had scored too.

at what time did they/he score each time, and how many points did they/he score each time?

Please list all the scoring information that meets the requirements

**\*\*Note:\*\*** - Set all NBA quarter breaks to 2 minutes.

- If some players scored in the same time, please list all the scoring information.

Please provide the answer in the following format for each top scorer:

Player1 scored [Points] in [Time]

Player2 scored [Points] in [Time]

for example:

Player1 scored 2 points in (quarter2 10:00)

Player2 scored 3 points in (quarter3 12:00)

...

**Answer:** "Kris Dunn scored 3 in (quarter2 48.1)",

##### Checklist:

```

{
  "Event 1: Kris Dunn scored 3": 0.5,
  "Event 1: Kris Dunn scored in (quarter2 48.1)": 0.5,
  "first_check": {
    "Incorrect player name(s) selected": 0
  }
}

```

### E.4.2 Interval-based Event Retrieval

Interval-based Event Retrieval requires the model to retrieve specific events occurring within a given time interval. The evaluation distributes the total score evenly across all target events within the specified interval. Each game period includes 1 such question.

#### Interval-based Event Retrieval

**Question:**

**GAME INFO**

<Time-Play Table>

End of Quarter / Half-time break

</Time-Play Table>

Now, your friend has said:

One of my friends was watching this NBA game too, but left to answer a phone call at (quarter1 4:29) and didn't return until (quarter1 2:45). Which goals did he miss? Please list the players who scored, when they scored, and how many points they got(including (quarter1 4:29) and (quarter1 2:45)).

Please provide the answer in the following format:

Player1 Name scored [Points] in [Time] Player2 Name scored [Points] in [Time]

for example:

Player1 Name scored 2 points in (quarter2 10:00)

Player2 Name scored 3 points in (quarter3 12:00)

...

**Answer:** "Devin Vassell scored 2 in (quarter1 4:29)

Nick Richards scored 2 in (quarter1 4:12)

Nick Richards scored 1 in (quarter1 4:12)

Keldon Johnson scored 3 in (quarter1 3:29)

Stephon Castle scored 1 in (quarter1 3:12)

Stephon Castle scored 1 in (quarter1 3:12)

Stephon Castle scored 2 in (quarter1 2:45)"

**Checklist:** {

"Event 1: Devin Vassell scored 2 in (quarter1 4:29)": 0.14285714285714285,

"Event 2: Nick Richards scored 2 in (quarter1 4:12)": 0.14285714285714285,

"Event 3: Nick Richards scored 1 in (quarter1 4:12)": 0.14285714285714285,

"Event 4: Keldon Johnson scored 3 in (quarter1 3:29)": 0.14285714285714285,

"Event 5: Stephon Castle scored 1 in (quarter1 3:12)": 0.14285714285714285,

"Event 6: Stephon Castle scored 1 in (quarter1 3:12)": 0.14285714285714285,

"Event 7: Stephon Castle scored 2 in (quarter1 2:45)": 0.14285714285714285

}

## E.5 Information Reasoning

The system prompt for IR task can be found in Prompt [D.2](#).

### E.5.1 Current Score Tracking

Current Score Tracking shares the same question design as in the Multi-turn scenarios in the Question [E.1](#), but the question is accessed only at the end of the game.

### E.5.2 Score Lead Fluctuation Detection

Score Lead Fluctuation Detection requires the model to identify the number of lead changes and the corresponding details during a specified game phase. The score is assigned as 0.2 for the correct count, and the remaining 0.8 is evenly distributed across the descriptions of each lead change. If there is at least one lead change in the period, one such question is included ; otherwise, none is included.

#### Score Lead Fluctuation Detection

**Question:**

**GAME INFO**

<Time-Play Table>

End of Quarter / Half-time break

</Time-Play Table>

Now, your friend has said:



In the First quarter, how many times did one team take the lead after previously being behind ? At what exact times did these lead changes occur, and which team became the new leader?

**\*\*Note:\*\*** A tie does not count as changing the order. If a team that is lagging continuously goes from tying to leading, information on the time of the comeback needs to be recorded.

Please answer using the following format:

In the First quarter, the score order changed [Number] times. The specific time and the leading team are as follows: At [Time1], [Team1]'s score had surpassed [Team2]'s score, ...

**Answer:** "In the First quarter, the score order changed 1 times. The specific time and the leading team are as follows:  
11:30: San Antonio Spurs's score had surpassed Phoenix Suns's  
"

**Checklist:** { "The score order changed 1 times": 0.2,  
"At 11:30, San Antonio Spurs's score had surpassed Phoenix Suns's": 0.8  
}

### E.5.3 Player Performance Impact Analysis

Player Performance Impact Analysis requires the model to analyze the impact of a specified player's on-court performance on a given segment of the game. For example, in the NBA game, the model is tasked with assessing how missed free throws or shots by the player affect the overall score during that segment. The scoring is evenly distributed across answers for each hypothetical scenario. If there is at least one lead change in the period, 2 such questions are included ; otherwise, one is included.

#### Player Performance Impact Analysis

**Question:**

**GAME INFO**

<Time-Play Table>

End of Quarter / Half-time break

</Time-Play Table>

Now, your friend has said:

Wow, the Second quarter was really exciting! Both teams performed very well, and the game was very intense. Who has scored the most goals from (1 quarter 11:29) to (2 quarter 9:42)(including (1 quarter 11:29) and (2 quarter 9:42))?  
If he has not made any free throws during this period, which team should be leading now and by how many points?  
What if all of his goals during this period were not scored?

**\*\*Note:\*\*** if multiple players have the same score, please select the player whose name comes first in alphabetical order  
Please provide the answer in the following format for the scorer:

The top scorer is [Player Name]. If he has not made any free throws during this period, the leading team should be [Team Name] and the score difference should be [Score Difference].

If all of his goals during this period were not scored, the leading team should be [Team Name] and the score difference should be [Score Difference].

**Answer:** "The top scorer is LeBron James.

If he has not made any free throws during this period, the leading team should be Los Angeles Lakers and the score difference should be 1.

If all of his goals during this period were not scored, the leading team should be LA Clippers and the score difference should be 8."

**Checklist:** {  
"Without free throws scenario: leading team is Los Angeles Lakers": 0.25,  
"Without all goals scenario: leading team is LA Clippers": 0.25,  
"Without free throws scenario: The score difference is 1": 0.25,  
"Without all goals scenario: The score difference is 8": 0.25,  
"first\_check": {  
"Incorrect player name(s) selected": 0  
}  
}

### E.6 Task Switch

The system prompt for TS task can be found in Prompt D.2.

### E.6.1 In-context Reasoning Query

The In-context Reasoning Query in TS task are same as IR's Question. See Appendix E.5.

### E.6.2 Out-of-context Math Query

Out-of-context Math Query mixes in math questions from MathQA within the dialogue. An example format is shown below. Each game period includes 3 such questions.

#### Example Question of Task Switch

Question:

**GAME INFO**

<Time-Play Table>

The competition is ongoing, users are chatting.

</Time-Play Table>

Now, your friend has said:

I have a computer science problem to ask you. Please choose the correct answer from the following options and reply with the letter of the option directly:

Another term for out-of-distribution detection is?

A. precision-recall detection B. underfitting detection

C. bias-variance tradeoff detection

D. regularization detection

E. one-class detection

F. overfitting detection

G. cross-validation detection

H. background detection

I. train-test mismatch robustness

J. anomaly detection

K. outlier detection

L. Both anomaly detection and outlier detection are correct

Answer:

"L"

Checklist: {

"Correct answer is L": 1.0

}

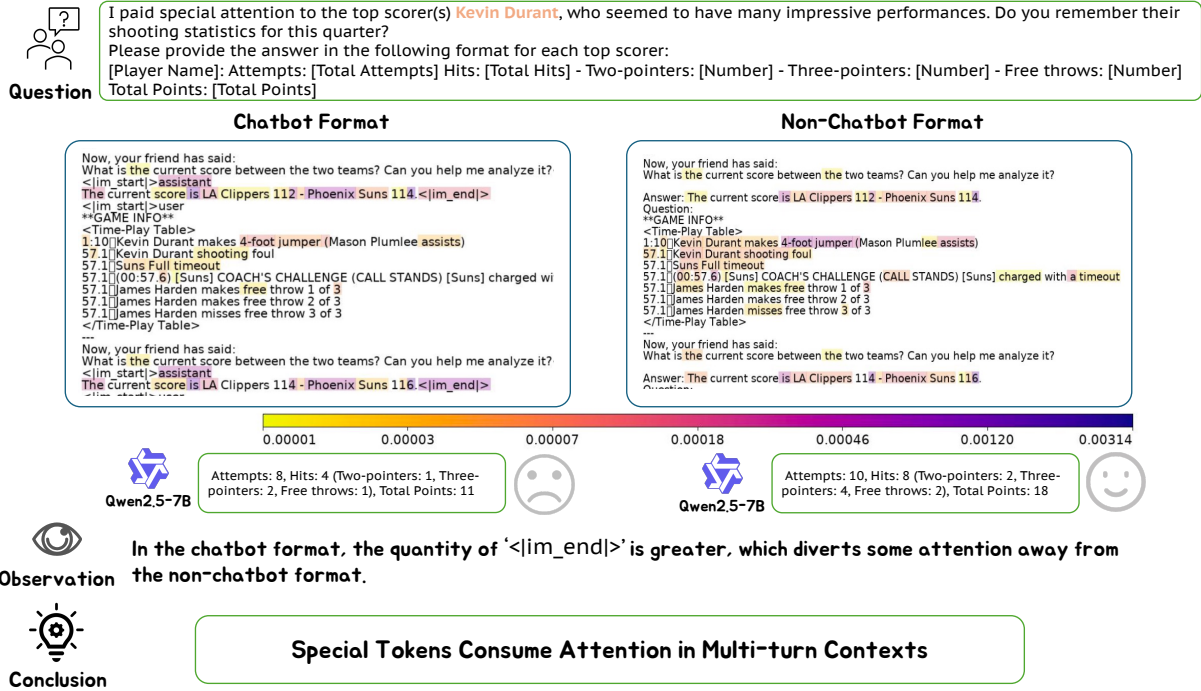


Figure 15: **Attention visualization in Qwen2.5-7B-Instruct.** Multi-turn inputs introduce more special tokens, which absorb a substantial portion of attention (e.g., “<|im\_end|>”), potentially reducing attention efficiency.

## F Attention Visual Analysis

To better understand how the model allocates attention under different input formats, we visualize the attention distributions of Qwen2.5-7B-Instruct in both chatbot and non-chatbot settings. As shown in Figure 15, multi-turn inputs introduce a larger number of special tokens, such as “<|im\_end|>”, which absorb a disproportionate amount of attention. This suggests a potential reduction in attention efficiency caused by input formatting, particularly in long-context, instruction-following scenarios.