

426 A Limitations

427 We acknowledge several shortcomings in our benchmark construction, which we summarize as
428 follows:

429 **Limited coverage of real-world scenarios and task design** Due to considerations in question design
430 and the need for reliable benchmark evaluation, our scope is limited to four representative domains
431 with a selected subset of key long-dependency tasks. However, many other real-world long-text
432 scenarios—such as extended debates, policy analyses, and scientific literature reviews—remain
433 unexplored and could provide valuable evaluation challenges.

434 **Limited length distribution** Although the overall context length in the benchmark is relatively evenly
435 distributed between 16K and 2M tokens, there are notable differences in the average context lengths
436 across different tasks. This may introduce task-specific biases when evaluating model performance
437 across different context lengths. Meanwhile, when assessing the effectiveness of our benchmark
438 through cross-task comparisons of model performance, variations in context length can also act as a
439 confounding factor.

440 B Instructions

441 This appendix outlines the prompt instructions and input formats used across the 10 tasks in **LooGLE**
442 **v2**. These prompts are carefully designed to guide models to produce outputs in a consistent format,
443 enabling more robust and reliable evaluation.

444 B.1 Law

445 B.1.1 Legal article extraction

Instruction: You are a senior expert in American law. You will now receive a text with two sections:

1. Section <MASKED TARGET CASE> contains an American legal case with several masked placeholders (e.g., <MASK_1>, <MASK_2>). These placeholders represent targeted parts in the case that need to be filled in, while other placeholders (<HIDDEN_INFO>) are included to prevent information leakage.
2. Section <RELATED LAW> provides related legal documents (e.g., <LAW_1>, <LAW_2>, etc.) that may be used as references to fill in those placeholders.

```
<text>
{context}
</text>
```

Your task is to analyze the MASKED TARGET CASE and the RELATED LAW, and determine which legal documents (LAW_x) best fills the specific placeholder mentioned in the following question:

```
<question>
{question}
</question>
```

You must select only one best answer. Please select the correct option and only response a single sentence with the format as follows:

"The correct answer is (LAW_x)"

446

447 **B.1.2 Legal case retrieval**

Instruction: You are a senior expert in American law. You will now receive a text with two sections:

1. Section <MASKED TARGET CASE> contains an American legal case with several masked placeholders (e.g., <MASK_1>, <MASK_2>). These placeholders represent missing parts in the case that need to be filled in.
2. Section <RELATED CASE> provides related legal cases (e.g., <CASE_1>, <CASE_2>, etc.) that may be used as references to fill in those placeholders.

```
<text>
{context}
</text>
```

Your task is to analyze the MASKED TARGET CASE and the RELATED CASE, and determine which related legal case (CASE_x) best completes the specific masked placeholder mentioned in the following question:

```
<question>
{question}
</question>
```

You must select only one best answer. Please select the correct option and only respond with a single sentence in the following format:

"The correct answer is (CASE_x)"

448

449 **B.2 Finance**

450 All tasks in the Finance domain share a common instruction prompt. Specific requirements for output
451 format and task details are embedded in each individual question. See Appendix C.2 for examples.

Instruction: You are a senior financial analyst. You will be provided with certain fiscal year's annual reports for one or more companies. In the provided financial reports, any data enclosed in parentheses () within tables is to be interpreted as a negative value. The text below contains all the 10-K annual reports in the format <FILE: filename> followed by the file content.

```
<text>
{context}
</text>
```

Your task is to analyze the financial information and answer the following question:

```
<question>
{question}
</question>
```

452

453 B.3 Game

454 B.3.1 Crafter

Instruction: Now, I'm going to present you a textual description of a Crafter game and a question, please read the text and answer the question. Crafter is a survival game in which a traveler explores a jungle, gathering materials and crafting tools to survive. The text I provide shows the traveler's trajectory in this game.

The text is as follows:

```
<text>
{context}
</text>
```

After reading the text, please answer the following question:

```
<question>
{question}
</question>
```

The options are as follows:

```
<option>
{options}
</option>
```

Please select the correct option and only response a single sentence with the format as follows:

"The correct answer is (insert answer here)."

455

456 B.3.2 Counter-strike 2 (CS2)

Instruction (User Behavior Analysis): Now, I'm going to present you a textual description of a Counter-Strike-2 (CS2) game and a question. Please read the text carefully and answer the question. CS2 is a tactical first-person shooter game featuring two teams, each consisting of five players. In the first half of the match, one team plays the role of the Terrorists ([T]), and the other plays the role of the Counter-Terrorists ([CT]). After the first 13 rounds, the two teams switch sides. The goal of the Terrorists is to plant the bomb in a designated area, while the Counter-Terrorists aim to prevent the bomb from being planted or to defuse it after it has been planted. The text I provide shows the gameplay process, including player positions, shooting, utility usage, and bomb plant/defuse actions in this game.

The text is as follows:

```
<text>
{context}
</text>
```

After reading the text, please answer the following question:

```
<question>
{question}
</question>
```

The options are as follows:

```
<option>
{options}
</option>
```

Please select the correct option and only response a single sentence with the format as follows:

"The correct answer is (insert answer here)."

457

Instruction (Environment Understanding): Now, I'm going to present you a textual description of a Counter-Strike-2 (CS2) game and a question. Please read the text carefully and answer the question. CS2 is.....(related rules).....After the first 13 rounds, the two teams switch sides. The goal of the Terrorists is to plant the bomb in a designated area, while the Counter-Terrorists aim to prevent the bomb from being planted or to defuse it after it has been planted. In a CS2 match, there are two bomb planting sites (A and B). In each round, the [T] side may plant a bomb at one of the sites. The text I provide shows the gameplay process, including player positions, shooting, utility usage, and bomb plant/defuse actions in this game.

<text>
{context}
</text>

After reading the text, please answer the following question:

<question>
{question}
</question>

The options are as follows:

<option>
{options}
</option>

Please select the correct option and only respond with a single sentence in the following format:

"The correct answer is (insert answer here)."

458

Instruction (Rule Understanding): Now, I'm going to present you a textual description of a Counter-Strike-2 (CS2) game and a question. Please read the text carefully and answer the question. CS2 is.....(related rules).....After the first 13 rounds, the two teams switch sides. The goal of the Terrorists is to plant the bomb in a designated area, while the Counter-Terrorists aim to prevent the bomb from being planted or to defuse it after it has been planted. The text I provide shows the gameplay process, including player positions, shooting, utility usage, and bomb plant/defuse actions in this game. In the game, at the end of each round, there will be an announcement "The round has ended."

<text>
{context}
</text>

There are five possible victory conditions for each round:

- T side wins when all CT players are eliminated
- T side wins when the bomb is planted and either the countdown or round time expires
- CT side wins when all T players are eliminated
- CT side wins by successfully defusing the bomb
- CT side wins when the round time expires and the bomb was not planted.

After reading the text, please answer the following question:

<question>
{question}
</question>

The options are as follows:

<option>
{options}
</option>

Please select the correct option and only respond with a single sentence in the following format:

"The correct answer is (insert answer here)."

459

460 **B.4 Code**

461 **B.4.1 Call graph analysis**

Instruction: You are a senior Python expert. You will now receive a full source code of a Python project. The text below contains all the source code in the format '<File>: relative_path\n', followed by the content of each file.

```
<text>
{context}
</text>
```

Your task is to analyze the call graph of the project and answer the following multiple choice question.

```
<question>
{question}
</question>
```

The options are as follows:

```
<option>
{options}
</option>
```

There is only one correct option. You must ensure that your answer is one of the given option letters.

Please select the correct option and only response a single sentence with the format as follows:
"The correct answer is (insert answer here)"

462

463 **B.4.2 Version control**

Instruction: You are a senior Python expert. You are given two full versions of a codebase, marked by <PREV_VERSION> and <CURR_VERSION>, representing the state before and after a commit.

Each version contains multiple files, and each file begins with a line in the format <File>: relative_path, followed by the file content.

```
<text>
{context}
</text>
```

Your task is to answer the following question:

```
<question>
{question}
</question>
```

Please respond with a list of the relative paths of the changed files, exactly as shown after each <File>: in the context.

Please begin your answer with:

"The changed files are:[insert relative file paths here]"

464

465 **C Examples in each domain from LooGLE v2**

466 To facilitate a better understanding of LooGLE v2's domain-specific task design, we present repre-
467 sentative questions from each of the 10 sub-tasks in the following section.

468 **C.1 Law**

Task: Legal Case Retrieval

Question: Choose the best related case(CASE_x) to be cited at the placeholder marked as <MASK_1>.

Options: []

Answer: CASE_4

Evidence:

original text: 'Porter v. Nussle, 534 U.S. 516, 520, 122 S.Ct. 983, 152 L.Ed.2d 12 (2002)'

469

Task: Legal Article Extraction

Question: Choose the best related law (LAW_x) to be cited at the placeholder marked as <MASK_1>.

Options: []

Answer: LAW_3

Evidence:

original text: '42 U.S.C. 1983'

470

471 **C.2 Finance**

Task: Metric Calculation

Question: Based on the annual reports of ELI LILLY AND COMPANY, what was ELI LILLY AND COMPANY's QuickRatio for the FY2020? Please format your response as: "The correct answer is X.XX" (ratio, rounded to 2 decimal places e.g. 1.25). Formula: QuickRatio = (CurrentAssets - InventoryNet) / CurrentLiabilities

Options: []

Answer: 1.08

Evidence:

CurrentAssets (2020): 17,462,100,000.00

InventoryNet (2020): 3,980,300,000.00

CurrentLiabilities (2020): 12,481,600,000.00

QuickRatio (2020): 1.08

472

Task: Trend Analysis

Question: Between FY2020 and FY2024, in which year did COSTCO WHOLESALE CORP /NEW's CapitalExpenditure have the largest absolute year-over-year change? Please format your response as: "The correct answer is XXXX-XXXX" (year pair e.g. 2000-2001).

Options: []

Answer: 2020-2021

Evidence:

2020-2021: +27.69%

2021-2022: +8.44%

2022-2023: +11.10%

2023-2024: +8.95%

473

Task: Cross-Company Comparison

Question: Among the following companies: 1. ELI LILLY AND COMPANY, 2. Exxon Mobil Corporation, 3. Intuitive Surgical, Inc., 4. COCA COLA CO, 5. Abbott Laboratories, which company had the highest WorkingCapital in FY2023? Please answer using the number in front of the company name. Please format your response as: "The correct answer is X" (plain integer number e.g. 2). Formula: $\text{WorkingCapital} = \text{CurrentAssets} - \text{CurrentLiabilities}$

Options: []

Answer: 2

Evidence:

WorkingCapital (2023):

Abbott Laboratories: 8,829,000,000.0

COCA COLA CO: 3,161,000,000.0

ELI LILLY AND COMPANY: -1,566,200,000.0

Exxon Mobil Corporation: 31,293,000,000.0

Intuitive Surgical, Inc.: 6,229,300,000.0

474

475 **C.3 Game**

Task: User Behavior Analysis

Question: Usually, the player who planted the bomb the most times is the one responsible for planting the bomb. Based on the description, please determine which of the following players was responsible for planting the bomb during the match?

Options:

A. PKL

B. lukiz

C. matios

D. xureba

Answer: A

Evidence:

Line 6497: [T]lukiz has planted the bomb

Line 7175: [T]PKL has planted the bomb

Line 8507: [T]PKL has planted the bomb

Line 10093: [T]PKL has planted the bomb

Line 16856: [T]matios has planted the bomb

476

Task: Environment Understanding

Question: If the traveler attempts to move in a certain direction but his or her position does not change in the next step, it means that the intended destination is blocked by an obstacle. Please determine which of the following positions contains an obstacle.

Options:

A. [28, 29]

B. [41, 1]

C. [30, 32]

D. [26, 1]

Answer: C

Evidence:

[28, 29] : The traveler is now at the position <[28, 29]>.

[41, 1] : The traveler is now at the position <[41, 1]>.

[30, 32] : The traveler is now at the position <[30, 33]>...In step 22, the traveler moved left...The traveler is now at the position <[30, 33]>.

[26, 1] : Not mentioned

477

Task: Environment Understanding

Question: Please determine which map is the same as the reference map.

Options:

- A. Map option 1
- B. Map option 2
- C. Map option 3
- D. Map option 4

Answer: A

Evidence:

(inferred from context)

- Reference map: train
- Map 1: train
- Map 2: ancient
- Map 3: nuke
- Map 4: anubis

478

Task: Rule Understanding

Question: Which round has the same victory condition as round 3.

Options:

- A. Round 1
- B. Round 13
- C. Round 9
- D. Round 20

Answer: C

Evidence:

Line 4361: [CT]naitte took down player [T]xureba with a M4A1. ... The round has ended.
Side CT won!

Line 8003: [CT]naitte took down player [T]detr0it with a M4A1. ... The round has ended.
Side CT won!

479

Task: Rule Understanding

Question: In the context, some steps describe actions as “the traveler did <PLACEHOLDER>”. The possible choices for the placeholder are picking, placing, crafting, or other activities. However, we do not know the exact action the traveler performed. You can infer it by comparing the information of the traveler before and after the action. Please identify at which step the traveler did the same thing as at step 13.

Options:

- A. Step 160
- B. Step 131
- C. Step 75
- D. Step 182

Answer: C

Evidence:

After doing this action, the traveler has 3 pieces of water, 5 pieces of wood
In step 13, the traveler did <PLACEHOLDER>

After doing this action, the traveler has 2 pieces of water, 6 pieces of wood

After doing this action, the traveler has 5 pieces of water, 8 pieces of wood

In step 75, the traveler did <PLACEHOLDER>

After doing this action, the traveler has 4 pieces of water, 9 pieces of wood

480

Task: Call Graph Analysis

Question: Which of the following function pairs has a call stack depth of **more than 2**? (e.g., if the call trace for the function pair (A, C) is $A \rightarrow B \rightarrow C$, the call stack depth is 3.)

Options:

- A. (src/core/video_planner.py::VideoPlanner._generate_scene_implementation_single, src/rag/rag_integration.py::RAGIntegration._generate_rag_queries_narration)
- B. (evaluate.py::process_theorem, mllm_tools/utils.py::_prepare_text_image_inputs)
- C. (src/core/video_planner.py::VideoPlanner.generate_scene_outline, src/rag/rag_integration.py::RAGIntegration.set_relevant_plugins)
- D. (src/rag/rag_integration.py::RAGIntegration._generate_rag_queries_storyboard, mllm_tools/utils.py::_prepare_text_inputs)

Answer: B**Evidence:**

- A: src/core/video_planner.py::VideoPlanner._generate_scene_implementation_single
→ src/rag/rag_integration.py::RAGIntegration._generate_rag_queries_narration
- B: evaluate.py::process_theorem
→ eval_suite/image_utils.py::evaluate_sampled_images
→ mllm_tools/utils.py::_prepare_text_image_inputs
- C: src/core/video_planner.py::VideoPlanner.generate_scene_outline
→ src/rag/rag_integration.py::RAGIntegration.set_relevant_plugins
- D: src/rag/rag_integration.py::RAGIntegration._generate_rag_queries_storyboard
→ mllm_tools/utils.py::_prepare_text_inputs

482

Task: Version Control

Question: Which files have been changed (added, deleted, or modified) between the two versions?

Options: []**Answer:**

'src/transformers/modeling_utils.py', 'src/transformers/models/blip_2/modeling_blip_2.py', 'tests/models/instructblip/test_modeling_instructblip.py'

Evidence:

diff -git a/src/transformers/modeling_utils.py b/src/transformers/modeling_utils.py
(too long, omitted...)

483

484 **D Finance metrics**

485 This part presents the base and derivative financial metrics used in the design of our finance-related
486 tasks. These metrics are selected and extended based on the set proposed in (Islam et al., 2023), with
487 refinements to include additional commonly used financial indicators. A detailed list of the metrics
488 used is shown in Table 4.

489 **E Text conversion templates and examples for game domain**

490 In this section, we present some examples of templates used for generating textual description
491 documents from game-source data. These templates are designed to accurately convey gameplay
492 information while closely resembling natural language. The raw data obtained from the source is
493 structured, consisting of multiple records, each corresponding to a specific type of event (referred
494 to as 'Event' below). We convert these structured information into textual descriptions based on
495 predefined templates (referred to as 'Template' below) associated with each event type. Detailed
496 method for generating textual descriptions is detailed in Appendix F.

Table 4: Financial metrics used in the finance task

Base Metrics		Derivative Metrics	
Revenue	Gross Profit	Gross Margin (%)	Net Profit Margin (%)
Operating Expenses	Net Income	Return on Assets (%)	Asset Turnover Ratio
Interest Expense	Income Tax Expense / Benefit	Fixed Asset Turnover Ratio	Inventory Turnover Ratio
Cost of Goods Sold	Depreciation and Amortization	Accounts Receivable Turnover	Current Ratio
Operating Income / Loss	Total Assets	Quick Ratio	Operating Cash Flow Ratio
Current Assets	Current Liabilities	Free Cash Flow Growth Rate (%)	Cash Flow Margin (%)
Long-Term Debt	Inventory (Net)	Cash-to-Revenue Ratio	COGS Margin (%)
Accounts Receivable (Net)	Property, Plant and Equipment (Net)	D&A Margin (%)	Operating Income Margin (%)
Operating Cash Flow	Investing Cash Flow	CapEx as % of Revenue	Net Working Capital
Capital Expenditure	Cash and Cash Equivalents	Free Cash Flow	EBITDA
Marketable Securities	Accounts Payable	Working Capital	
Dividends Paid	Shareholders' Equity		
Total Liabilities			

497 E.1 Counter-strike 2

Event: Killing

Template: Oops! Player <killer> took down player <victim> with a <weapon>. It seems the game is so easy for player <killer>. It was/wasn't a headshot! It was/wasn't a wallbang!

498

Event: End of a round

Template: The round has ended. Side <winner> won! The score is now T: <t_score> - CT: <ct_score>.

499

Event: Chat message

Template: Player <sender> sent an message. The content of the message is '<text>'

500

Event: Player Hurt

Template: Oops! Player <attacker> dealt <damage> points of damage to player <victim> with a <weapon>. Player <victim> now has <hp_after> points of health left.

501

Event: Bomb Plant

Template: <thrower> has planted the bomb at <bomb_site>. The clock is ticking!

502

Event: Bomb Defuse

Template: Fantastic! <thrower> successfully defused the bomb at <bomb_site>. Crisis averted!

503

Event: Frame Snapshot

Template: Let's take a look at the conditions and the position of some players. Player <name> of team <team> now has <health> points of health left. The player's remaining money is <money>. And now we discover that the player walked into <room>. / And the player is still at <room>.

504

505 E.2 Crafter

Event: Action Execution

Template: In step <step>, the traveler <action>.

Event: Position Update

Template: The traveler is now at the position <[x, y]>.

Event: Inventory Change

Template: After doing this action, the traveler earns <n pieces of resource, ...>. The traveler's health is now <hp> points.

Event: Tool Crafting

Template: The traveler crafted a <tool name>.

Event: Achievement Unlocked

Template: The traveler achieved <achievement name>.

511 F Task annotation details for game

512 In this section, we present the detailed process of generating natural language textual descriptions and
513 automatically annotating questions for the game domain. All questions in this domain are generated
514 through automated procedures. We describe the structure of the original structured data, the process
515 by which this data is transformed into natural language, the format and characteristics of the resulting
516 textual descriptions, the construction of questions based on key information extracted from the text
517 using predefined *Text Conversion Templates* (see Appendix C.3), and the procedure for extracting
518 supporting evidence and identifying the correct answers.

519 F.1 Counter-strike 2

520 In the tasks related to *Counter-Strike 2*, each original document we downloaded corresponds to a
521 replay file of a single match, stored in a binary format. These files chronologically log in-game
522 events and contain comprehensive gameplay traces, including player location data, various event
523 types (e.g., kills, bomb plants, defusals, damage), player-specific information (e.g., health, weapon,
524 armor), as well as match scores and metadata. We extracted key information from these binary
525 original documents using the CS2 Demo Parser¹¹, converting the data into a structured JSON format.
526 The resulting JSON files preserve the chronological order of events, with each entry corresponding
527 to a single in-game event. However, we selectively retain only those events that are critical to
528 the game's win conditions and indicative of player performance—precisely the aspects we aim
529 to evaluate LLMs on for understanding. Additionally, we convert player location data from raw
530 coordinates to corresponding room names on the game map. Subsequently, each event tuple in the
531 JSON file is transformed into one or more natural language textual descriptions using predefined
532 sentence templates (see Appendix E). These descriptions collectively form the documents used in
533 our benchmark. The question formats for this task are shown in Appendix C.3. The procedures for
534 question generation and evidence annotation are described as follows.

¹¹<https://github.com/markus-wa/demoinfocs-golang/>

F.1.1 Environment understanding

In our benchmark, each document filename contains the name of the map on which the corresponding match was played. We use this as the ground truth to construct the task. Specifically, we select two documents played on the same map—one is used as the reference match mentioned in the question stem, and the other as the correct option. Additionally, we randomly select three other documents played on different maps to serve as incorrect options. The five selected documents are then concatenated to form the context for the question.

F.1.2 User behavior analysis

This task includes three distinct types of questions. In the following, we describe the generation method for each question type, as well as the corresponding answer and evidence annotation strategies.

For bomb-related tasks, we design three types of questions, each with a distinct generation and annotation strategy.

For questions targeting bomb planting, we extract all occurrences of the sentence pattern “[T]XXXXXX has planted the bomb at X. The clock is ticking!”, where XXXXXX denotes the player’s name. We then count the number of bomb plants performed by each player. If the difference between the most and least frequent bomb planter is less than 3, the document is discarded. Otherwise, the player with the highest number of bomb plants is selected as the correct answer, while the three players with the lowest frequencies are used as incorrect options.

For questions focusing on bomb defusal, we extract sentences of the form “[CT]XXXXXX successfully defused the bomb at X. Crisis averted!”, again identifying player names from the text. We compute the number of defusals per player. If the difference between the most and least frequent defuser is less than 2, we discard the document. If not, the player with the most defusals is chosen as the correct answer, and three players with the fewest defusals are chosen as distractors.

For questions related to bomb plant site preference, we again extract the sentence “[T]XXXXXX has planted the bomb at X. The clock is ticking!”, but this time focus on the bomb site X. We count how often each site is used within a document. If the absolute difference in frequency between the two sites is less than 3, the document is excluded. Otherwise, the more frequently used site is set as the correct answer.

F.1.3 Rule understanding

From each document, we extract one of the following five sentence patterns at the end of each round to identify the round’s win condition:

1. “Fantastic! [X]XXXXXX successfully defused the bomb at X. Crisis averted!
The round has ended. Side CT won! The score is now T: X - CT: X.”
2. “Oops! Player [X]XXXXXX took down player [X]XXXXXX with a XXXX. It seems the game is so easy for player [X]XXXXXX. It wasn’t a headshot. It wasn’t a wallbang.
The round has ended. Side CT won! The score is now T: X - CT: X.”
3. “The round has ended. Side CT won! The score is now T: X - CT: X.” (but not the above two types)
4. “Oops! Player [X]XXXXXX took down player [X]XXXXXX with a XXXX. It seems the game is so easy for player [X]XXXXXX. It wasn’t a headshot. It wasn’t a wallbang.
The round has ended. Side T won! The score is now T: X - CT: X.”
5. “The round has ended. Side T won! The score is now T: X - CT: X.” (but not the fourth type)

Each of these patterns corresponds to a specific type of win condition. For every document, we construct a list that records the win condition of each round in sequential order based on the extracted sentence patterns.

To construct the question, we select two rounds that are sufficiently far apart and share the same win condition. One of these rounds serves as the reference round in the question stem, and the other as the correct option. We then randomly select three additional rounds from the beginning, middle, and end of the list, each with a different win condition, to serve as incorrect options.

584 F.2 Crafter

585 In the tasks related to *Crafter*, each original document we downloaded contains a snapshot of every
586 step in the gameplay. These documents are in JSON format, where each entry records the information
587 of a single step, including the action taken by the player, the player’s position, various player attributes
588 (e.g., food, water, energy, materials, and the amount of different tools), and whether the player has
589 completed each achievement up to that point. We first organize this information into a simplified
590 form by computing the stepwise deltas in tool quantities and achievement completion status. Then,
591 we convert the structured data into natural language textual descriptions using predefined sentence
592 templates. These descriptions collectively constitute the documents used in our benchmark. The
593 question formats for this task are shown in Appendix C.3. The procedures for question generation
594 and evidence annotation are described as follows.

595 F.2.1 Environment understanding

596 From each document, we extract all instances of the sentence pattern: “... The traveler is now at the
597 position [X, Y] ... In step 38, the traveler moved <A DIRECTION>.”
598 The traveler is now at the position [X, Y] ...” where the two position coordinates [X, Y] are identical.
599 Such patterns indicate that the traveler attempted to move one step in the specified direction but
600 remained in the same position, suggesting that there is an obstacle in the intended direction of
601 movement. We traverse the entire document to collect all such patterns and infer the corresponding
602 blocked grid positions, forming a list of obstacle locations. From this list, we randomly select one
603 position as the correct answer. Then, we choose three additional positions that are not in the list as
604 incorrect options.

605 F.2.2 User behavior analysis

606 From each document, we extract all instances of the sentence pattern: “In step X, the traveler did
607 nothing.” Each occurrence of this sentence indicates that the player was sleeping during step X. We
608 collect all such step numbers into a list. We then search this list for any continuous interval of 50
609 steps during which the player slept without interruption. From these 50-step sleeping intervals, we
610 extract all possible subintervals of 10 consecutive sleeping steps, forming a new list of candidate
611 intervals. A certain 10-step sleeping interval is randomly selected from this list as the correct option.
612 Three additional 10-step intervals that do not belong to this list are sampled as incorrect options.

613 F.2.3 Rule understanding

614 From each document, we extract all instances of the sentence pattern: “In step X, the traveler did
615 <PLACEHOLDER>.” All such step numbers X are collected into a list of unknown actions. For each
616 step in this list, we further extract the textual descriptions from both the current step and the previous
617 step in the format: “... the traveler earns 9 pieces of food, 9 pieces of drink, 8 pieces of energy, 1
618 piece of sapling, 1 piece of wood. The traveler’s health is now 9 points. In step 40, the traveler did
619 something ... After doing this action, the traveler earns 9 pieces of food, 9 pieces of drink, 8 pieces
620 of energy, 1 piece of sapling, 2 pieces of wood. The traveler’s health is now 9 points.” We compute
621 the difference in resources and health before and after each unknown action and store the results as
622 a dictionary in the format {step: change vector}. We then search this dictionary for two steps with
623 identical change vectors. One is selected as the reference step in the question stem and the other as
624 the correct option. To generate incorrect options, we select three additional steps from the beginning,
625 middle, and end of the dictionary whose change vectors differ from the correct one. These steps are
626 used as incorrect options.

627 G Implementation details

628 G.1 Model configurations and hyperparameters

629 We evaluate our benchmark on 10 representative language models, comprising 6 locally deployed
630 models and 4 API-based models. The locally deployed models include Qwen2.5-7B-Instruct-1M
631 (Yang et al., 2025), LLaMA-3.1-8B-Instruct (Dubey et al., 2024), Mistral-7B-Instruct-v0.2 (Jiang
632 et al., 2023), GLM-4-9B-Chat-hf (Zeng et al., 2024), Phi-3-Medium-128K-Instruct (Abdin et al.,

2024), and Yarn-Mistral-7b-128k (Peng et al.). The API-based models include GPT-4.1(Achiam et al., 2023), GPT-o3-mini, DeepSeek-V3(Liu et al., 2024a) and DeepDeek-R1(Guo et al., 2025). These models span a wide range of context window sizes(from 32K to 1M tokens) and parameter scales, providing a comprehensive testbed for evaluating long-context understanding. All models are evaluated using a decoding temperature of 0.1, top- p of 1.0, and a generation limit of 512 tokens (max_new_tokens=512).

639 G.2 Hardwares and resource

640 All evaluations are conducted on four A100 GPUs, each equipped with 80 GB of memory. During
641 evaluation, we first serve the model locally using the vLLM engine (Kwon et al., 2023), and then query
642 the model responses via the OpenAI-compatible client interface.

643 G.3 Evaluation Metrics For Version Control Task

644 To evaluate model performance on the VERSION CONTROL task, we use the Jaccard Similarity (Jac-
645 card, 1901) as the primary metric. This task involves identifying the set of files or paths that have
646 changed between two commits. Let A denote the set of predicted changed paths and B denote the
647 ground-truth set. The Jaccard Similarity is defined as:

$$\text{Jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

648 This metric captures the overlap between the predicted and actual changed files, balancing both
649 precision and recall. A score of 1.0 indicates a perfect match, while 0.0 indicates no overlap. We
650 report the average Jaccard score across all evaluated samples.

651 H Detailed results of retrieve based methods

652 In the evaluation of retrieve-based methods, we employ two base models—LLAMA-3.1-8B-
653 INSTRUCT and GLM-4-9B-CHAT—to assess whether retrieval augmentation can improve models’
654 long-dependency reasoning capabilities. A subset of 645 questions is selected for testing, with the
655 distribution of question types closely matching that of the full benchmark to ensure representative-
656 ness. For each test instance, the context is divided into 512-token chunks. Both the chunks and the
657 corresponding question are tokenized using the LONGCITE-GLM4-9B tokenizer (Zhang et al., 2024a).
658 We then compute the semantic similarity between the question and each chunk, selecting the top- k
659 most relevant chunks as the retrieved context. Experiments are conducted with $k = 128, 64, 32, 16,$
660 $8,$ and 4 to evaluate performance under varying retrieval granularity.

661 Table 5 presents the results of the retrieval-based methods, with ‘w/o’ indicating the baseline without
662 retrieval augmentation. We observe that retrieval augmentation does not lead to consistent per-
663 formance gains for either LLAMA-3.1-8B-INSTRUCT or GLM-4-9B-CHAT. This suggests that
664 LOOGLE v2 is well-aligned with the characteristics of long-dependency reasoning tasks, where
665 answering typically requires reasoning over dispersed, non-localized information rather than relying
666 on a few highly relevant chunks. In fact, performance often degrades as fewer top- k chunks are used,
667 highlighting the limitations of local retrieval in capturing global dependencies. However, we note
668 an exception in domain-specific tasks such as *Finance*, where certain questions focus on locating
669 specific indicators. In such cases, a small number of retrieved chunks may suffice to provide the
670 necessary information, and moderate performance gains are observed when using retrieval-based
671 methods.

672 I Detailed results with chain-of-thoughts

673 To better evaluate the varying demands of long-dependency reasoning across different tasks in
674 LOOGLE v2, we incorporate chain-of-thought (CoT) prompting to encourage explicit reasoning in
675 model outputs. Explicit reasoning in the model’s responses significantly improves its performance in
676 our long-context reasoning tests.

Table 5: Evaluation of Retrieval-augmented methods on **LooGLE v2**

	Law		Finance			Game			Code		Avg
Top-k	Legal Article Extraction	Legal Case Retrieval	Metric Calculation	Trend Analysis	Cross-Company Comparison	Environmental Understanding	User Behavior Analysis	Rule Understanding	Call Graph Analysis	Version Control	Average
Llama-3.1-8B-Instruct											
4	1.61	5.62	8.82	9.09	5.97	16.95	20.00	20.37	23.20	3.13	12.26
8	4.84	8.99	11.76	12.12	10.45	22.03	23.64	29.63	25.60	5.95	16.12
16	8.06	10.11	26.47	15.15	11.94	18.64	30.91	18.52	23.20	6.46	16.64
32	9.68	17.98	44.12	15.15	22.39	22.03	32.73	25.93	16.00	8.12	19.76
64	9.68	15.73	70.59	42.42	25.37	27.12	30.91	29.63	20.80	11.70	24.47
128	8.06	15.73	64.71	36.36	22.39	23.73	34.55	33.33	25.60	12.35	24.69
w/o	25.81	19.10	61.76	36.36	17.91	22.42	25.45	22.22	30.40	18.39	26.25
GLM-4-9b-chat											
4	1.61	7.87	2.94	0.00	0.00	10.17	12.73	22.22	27.20	2.51	10.80
8	3.23	14.61	8.82	12.12	5.97	11.86	9.09	25.93	27.20	5.91	13.95
16	4.84	16.85	23.53	21.21	13.43	11.86	23.64	31.48	30.40	6.80	18.85
32	8.06	26.97	38.24	18.18	14.93	6.78	21.82	22.22	24.80	7.03	18.87
64	6.45	28.09	47.06	30.30	25.37	15.25	27.27	22.22	25.60	10.52	22.80
128	14.52	25.84	35.29	39.39	22.39	11.86	25.45	18.52	23.20	12.87	21.80
w/o	27.42	37.08	32.35	48.48	17.91	15.25	16.36	29.63	27.20	17.65	26.17

Following the setup proposed by Bai et al. (2024b), we adopt a two-step chain-of-thought (CoT) prompting strategy. In the first step, the model is prompted to generate a detailed reasoning process (Wei et al., 2022). To achieve this, we modify the original prompt by removing the output formatting instructions at the end and appending the phrase “Let’s think step by step.” In the second step, the generated chain of thought is appended to the prompt, and the model is queried again to produce the final answer. This CoT setup is applied to selected tasks and models in LOOGLE v2 to evaluate its effectiveness in enhancing long-range dependency reasoning.

We evaluated chain-of-thought (CoT) prompting on four models across various tasks, with results shown in Table 6. The results are averaged over multiple trials to mitigate variability and enhance robustness. Overall, CoT does not consistently improve performance on the benchmark but benefits specific tasks that require focused, step-by-step reasoning, such as finance. Notably, Qwen2.5-7B-Instruct-1M and LLaMA-3.1-8B-Instruct show substantial gains on metric calculation and cross-company comparison with CoT, while GLM-4-9B-Chat’s performance remains stable. Mistral-7B-Instruct-v0.2 exhibits a decline with CoT, likely influenced by its low baseline performance.

Table 6: Comparative results of chain-of-thought prompting effects on model performance

		Law		Finance			Game			Code		Avg
Model	Chain of Thoughts	Legal Article Extraction	Legal Case Retrieval	Metric Calculation	Trend Analysis	Cross-Company Comparison	Environmental Understanding	User Behavior Analysis	Rule Understanding	Call Graph Analysis	Version Control	Average
Locally-Deployed Models												
LLaMA-3.1-8B-Instruct	w/o	17.28	20.60	65.00	33.00	21.00	17.61	15.84	19.39	28.99	21.12	24.16
	w/	28.14	25.22	69.67	36.00	29.00	21.02	29.07	28.08	23.49	18.9	27.94
Qwen2.5-7B-Instruct-1M	w/o	22.58	16.85	84.00	31.00	27.00	24.57	59.26	24.39	23.14	18.27	28.97
	w/	32.80	35.96	82.00	33.00	26.40	14.29	46.30	16.46	21.58	12.09	28.91
GLM-4-9b-chat	w/o	28.49	37.08	41.00	43.00	18.50	17.61	23.17	12.73	26.06	19.12	25.81
	w/	26.70	36.96	61.67	30.00	21.17	18.18	25.81	13.74	25.97	18.07	26.54
Mistral-7B-v0.2-Instruct	w/o	6.81	5.62	1.00	15.00	8.50	5.68	7.92	22.42	23.94	9.59	11.88
	w/	3.23	5.37	5.00	19.33	10.83	0.95	5.49	18.99	10.99	8.07	8.57