

# Structure-Originated Reasoning Data Improves Long-Context Reasoning Ability of Large Language Models

Quyet V. Do, Thinh Pham, Nguyen Nguyen, Sha Li, Pratibha Zunjare, Tu Vu

Virginia Tech, VA 24061, USA

{quyetdo, thinhphp, stephenallstar24, shal, pratibhaz, tuvu}@vt.edu

## Abstract

We study a QA curation pipeline for improving long-context complex reasoning in large language models (LLMs). Our approach,  $\pi^2$ , constructs high-quality reasoning data through rigorous QA curation: 1) extracting and expanding tables from Wikipedia, 2) from the collected tables together with relevant metadata, generating complex reasoning questions whose answers are automatically determined and validated through dual-path code execution, 3) finally, back-translating chain-of-thoughts solutions grounded in realistic context. Supervised fine-tuning with GPT-OSS-20B and QWEN3-4B-INSTRUCT-2507 on  $\pi^2$  yields consistent improvements across four long-context reasoning benchmarks and our alike  $\pi^2$ -Bench, with average absolute accuracy gains of +6.25% and +3.37% respectively. Through deeper analyses, we observe that reasoning style contributes little, while faithful reasoning patterns discovered by back translation and grounded realistic long context, as  $\pi^2$  is designed for, are crucial for the improvement. Our code, data, and models are fully open-source at <https://github.com/vtpss/pi-squared>.

## 1 Introduction

*"In all chaos there is a cosmos, in all disorder a secret order."* - Carl Jung

Large language models (LLMs) have made rapid progress in handling long contexts, with recent systems supporting hundreds of thousands to millions of tokens (OpenAI et al., 2025; Yang et al., 2025a; OpenAI, 2026; Anthropic, 2026; Team, 2026). This capability enables applications that require integrating information from multiple documents, such as open-domain question answering, document analysis, and web-scale search. However, improved context capacity does not translate to reasoning capacity, as current flagship models still struggle on complex reasoning over realistic long contexts (Zhu et al., 2024; Bai et al., 2025; Chen et al., 2025b; Pham et al., 2026; Opsahl-Ong et al., 2026).

A key bottleneck lies in the lack of verifiable supervision for complex reasoning over *realistic long context in open-domain tasks*. Hugging Face (2025) and Guha et al. (2026) represent a line of works that provide complex reasoning data with high-quality supervision, however, data domains are limited to natural science or coding. In another line of works, Yang et al. (2025b); Wang et al. (2025) and Xie et al. (2026) ensure the verifiability of open-domain reasoning data, yet the contexts are synthetic or limited to tables, not realistic long documents. Meanwhile, others were built on real open-domain documents and validated via code execution on concept graphs parsed from documents (Yang et al., 2018; Trivedi et al., 2022; Chen et al., 2025a; Shen et al., 2025; Sun et al., 2026). Nonetheless, their complexity is limited to *low branching-factor* reasoning patterns. This creates a gap: reliable supervision on complex reasoning is common in structured domains, but remains largely absent for realistic long contexts in open-domain tasks.

We observe that structured representations, such as tables, is a natural medium that bridges complex reasoning and real long-context open-domain data, in which the data construction and verification are otherwise difficult. Tables represent entities, attributes, and relations across sources, which can be losslessly altered by realistic documents. On the other hand, they remain compatible with operations such as filtering, aggregation, and sorting, which enables easy formation of *high branching-factor* reasoning tasks and automatic verification of answers. For example (Figure 2), consider the table "Player

of the Match awards" in the "2026 Six Nations Championship" Wikipedia page, where each row links to the player's page. After augmenting this table with a new attribute `INTERNATIONAL_CAPS` retrieved from linked pages, we can generate a complex reasoning question: In the 2026 Six Nations Championship, which player has the highest number of international caps among Player of the Match winners in Round 4? Answering this question involves filtering by match round and selecting the maximum over caps (e.g., filtering rows where `ROUND = 4`, then sorting by `INTERNATIONAL_CAPS` and selecting the top entry), which can be verified exactly through an executable query. Also, when the Wikipedia page and linked pages are provided instead of the augmented table, all the evidence is losslessly covered and distributed across multiple documents.

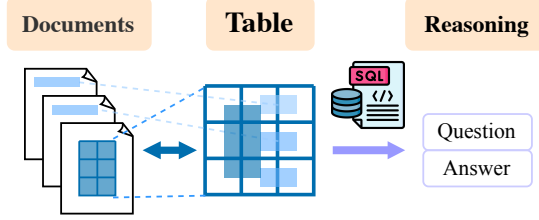


Figure 1: Tables bridge realistic documents and complex reasoning in open-domain tasks. They represent data across sources, while being compatible with programmatic operations.

Based on this observation, we propose  $\pi^2$  (Pipeline for Generating Structure-based Question Answering and Reasoning Data over Long Documents). The pipeline has three steps: 1) Collects tables from webpages and expands them with information retrieved from linked documents if conditions are met; 2) For each collected table, generate a complex question together with an executable SQL query by an LLM and verify the SQL-derived answer with an independent LLM-generated Python solution; 3) Finally, back-translates reasoning traces grounded in real free-form context from the verified answer. The generated data trains models to perform complex reasoning over realistic long documents in open-domain tasks.

We show that training on data generated with  $\pi^2$  consistently improves long-context complex reasoning performance across multiple benchmarks, including in-domain LongSeal (Pham et al., 2026), OfficeQA (Opsahl-Ong et al., 2026), and our held-out evaluation set  $\pi^2$ -Bench; as well as out-of-domain LongBench-v2 (Bai et al., 2025) and Oolong-Synth (Bertsch et al., 2025); with average gains of +6.25% and +3.37% for GPT-OSS-20B and QWEN3-4B-INSTRUCT-2507. Compared to the base models, finetuned models exhibit a stronger ability to reason over extended contexts, though it comes with a cost of overthinking. Through deeper analyses, we observe that reasoning style contributes little, while faithful reasoning patterns discovered by back translation and grounded realistic long context are crucial for the improvement. Finally, by the direct comparison with a previous long-context reasoning dataset, we show that  $\pi^2$  certainly fills the gap between previous works and real-world demands.

Our contributions are as follows:

- We propose  $\pi^2$ , a QA curation pipeline that leverages tables as structured intermediate representations to generate verifiable, long-context, complex reasoning data across domains (§2).
- We curate a dataset, also named  $\pi^2$ , with a held-out benchmark,  $\pi^2$ -Bench, for training and evaluating long-context complex reasoning in realistic settings (§2.4, §2.5).
- We show consistent improvements across multiple long-context reasoning benchmarks by finetuning models on  $\pi^2$ , and that the improvements are attributed to faithful reasoning patterns discovered by back translation and grounded realistic long context (§3).

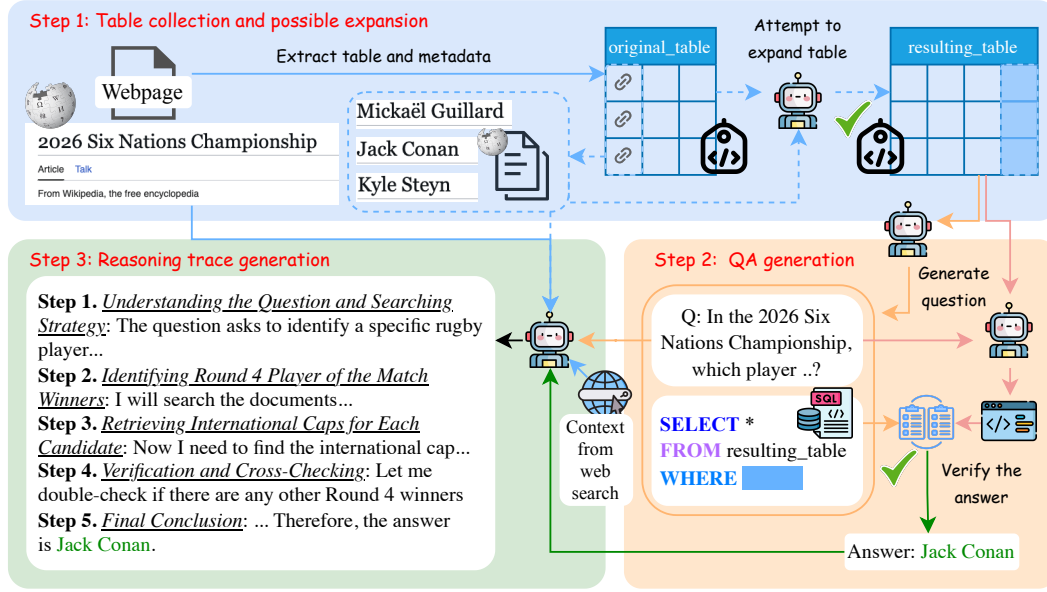


Figure 2:  $\pi^2$  curation pipeline. It first 1) collects tables from webpages and expands them with information retrieved from linked documents if conditions are met (indicated by dashed arrows and objects); then 2) from each collected table, generate a complex question together with an executable SQL query by an LLM and verify the SQL-derived answer with an independent LLM-generated Python solution, and finally 3) back-translates reasoning traces grounded in real free-form context from the verified answer.

## 2 Curation of $\pi^2$

### 2.1 Table Collection and Expansion

We start with a diverse set of high-view and recent Wikipedia pages.<sup>1</sup> For each page, we extract tables and their associated metadata (e.g., caption, section, hatnote) by parsing the underlying HTML form. Indeed, an extracted table represents itself as evidence in one document, while it is not certain if the table desirably represents evidence from multiple documents. To actualize this property, we attempt to expand each extracted table with up to 3 new, contextually relevant columns whose content does not exist in the original Wikipedia source. In case a column in the extracted table contains reference links for all entries, we use the infoboxes of those linked Wikipedia pages as external knowledge, then employ an LLM for the expansion. Post-expansion verification, including exact matching and LLM-as-a-judge, is executed to guarantee the information integrity and relevance of new columns.

### 2.2 QA Generation

For each resulting table (original one if the expansion failed) and its associated metadata, we prompt an LLM to generate a realistic complex question in natural language alongside its corresponding executable SQL query. The question is designed to cover many columns of the table. If the table is expanded, the question should cover new columns as conditions or the target of QA. That resembles tasks which requires information aggregation from multiple real documents. Meanwhile, the question is expected to have a deterministic, unique, and short answer, as well as to be *multi-hop reasoning with high branching factor*, i.e., to require multi-step linking, filtering, sorting, and/or numerical operations on attributes of multiple entities. The SQL query is then executed on the table, giving the candidate ground-truth answer.

<sup>1</sup>Accessed via the pymediawiki Python package

| Question                                                                                                                                                             | Answer         | Evidence                                                          |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------|-------------------------------------------------------------------|
| Among the television roles held by Debbie Reynolds in her career, how many distinct character or guest roles did she perform specifically on shows broadcast by NBC? | 3              | Debbie Reynolds; Aloha Paradise; Alice (American TV series); etc. |
| In the 2023 Cricket World Cup, among bowlers who use a Right-arm fast style, which player took the most wickets?                                                     | Mohammed Shami | 2023 Cricket World Cup; Mohammed Shami; etc.                      |
| Based on the 2022 cobalt production and reserves data reported by the USGS, which country had the highest ratio of reserves to annual production?                    | Australia      | Cobalt                                                            |

Table 1: **Samples in  $\pi^2$** . The "Evidence" column shows the evidence Wikipedia page(s). The sample is originated from a table in the first listed page. Our samples cover diverse topics and complexity levels.

We control the question’s quality and the ground-truth answer’s correctness as follows. First, we prompt an LLM to judge the complexity, self-containedness,<sup>2</sup> and naturalness of the question in a Likert-5 scale. If the judge provides a low rating among three ratings, which indicates issues such as being overly simple, lack of clarity, or grammatical errors; the generation process is triggered to re-run (at most three times). Second, we validate the SQL-derived answer with another independent computational path. We prompt an LLM to generate an independent Python-implemented solution of the question with the same tabular data as input. A generated QA sample is only retained if the answers from the SQL query and the Python script reach a consensus, determined by an LLM judge. More details of this step are shown in [Appendix A](#).

### 2.3 Reasoning Trace Generation

For each validated question-answer pair, we first gather a corpus of evidence documents and possibly hard negative ones, which resemble real-world context. This corpus includes the Wikipedia pages used for table curation and ten relevant articles retrieved via automated web search with the question. To make the context manageable for long-context LLMs, we aggregate these sources and compact them to a maximum of 96K tokens, resulting in the *compacted context*. We use BM25+ ([Lv & Zhai, 2011](#)) to rank the relevance to the question of all context passages, then iteratively discard the least relevant passage until the context length decreases to at most 96K. As location of evidence in Wikipedia pages is known, we ensure that all evidence, remains in the compacted context.

Finally, we prompt an LLM to articulate the *reasoning steps* required to solve the question, given the compacted context and the verified answer. We note that the original extracted table is likely used as evidence in the reasoning traces, yet, retrieval steps from real context is required. For cases where the table is expanded, there must be further steps for evidence aggregation from multiple documents.

### 2.4 $\pi^2$

We started with 6,614 Wikipedia pages, collected 2,834 qualified tables and 1,172 validated QA samples with reasoning traces. We demonstrate 3 samples of  $\pi^2$  in [Table 1](#). They cover diverse topics, question types, and complexity levels. Among the validated ones, 369 samples are based on expanded tables. For table expansion and QA curation, we used QWEN3.5-35B-A3B-FP8. For reasoning traces generation, as we observed different reasoning "styles" of models from different families, we adaptively select the teacher model according to the target student model. We select GPT-OSS-120B/QWEN3.5-35B-A3B-FP8 if the student model is GPT-OSS-20B/QWEN3.4B-INSTRUCT-2507, respectively. Overall,  $\pi^2$  samples have context length ranging from 3.4K to 96K tokens with a median of 50.8K tokens, and reasoning trace length

<sup>2</sup>Self-containedness means "providing all necessary parameters, definitions, and background information within itself, requiring no external links, data, references, or prior knowledge to be understood"

|                        | In-domain    |              |                | Out-of-domain |              | Avg.         |
|------------------------|--------------|--------------|----------------|---------------|--------------|--------------|
|                        | LongSeal     | OfficeQA     | $\pi^2$ -Bench | LongBenchV2   | Oolong       |              |
| GEMINI 3.1 FLASH-LITE  | <b>64.96</b> | <b>57.14</b> | <b>75.29</b>   | 58.14         | <b>42.38</b> | <b>59.58</b> |
| GEMINI-2.5-PRO         | 59.84        | 53.37        | 72.32          | 56.3          | 36.56        | 55.68        |
| QWEN3.5-35B-A3B-FP8    | 58.5         | 55.74        | 76.97          | <b>58.22</b>  | 39.76        | 57.84        |
| GPT-OSS-120B (low)     | 42.18        | 33.88        | 75             | 51.06         | 38.08        | 48.04        |
| QWEN3-4B-INSTRUCT-2507 | 29.14        | <b>15.12</b> | 54.68          | 36.09         | 30.51        | 33.11        |
| $\pi^2$ -4B            | <b>37.01</b> | 14.53        | <b>55.35</b>   | <b>41.12</b>  | <b>34.41</b> | <b>36.48</b> |
| GPT-OSS-20B (low)      | 34.38        | 22.38        | 60.86          | 38.76         | 27.96        | 36.87        |
| $\pi^2$ -20B (low)     | <b>39.69</b> | <b>30.40</b> | <b>68.01</b>   | <b>44.52</b>  | <b>32.98</b> | <b>43.12</b> |
| GPT-OSS-20B (high)     | 51.12        | 32.75        | 70.82          | 47.42         | 36.65        | 47.75        |
| $\pi^2$ -20B (high)    | <b>61.09</b> | <b>41.91</b> | <b>73.93</b>   | <b>50.73</b>  | <b>37.59</b> | <b>53.05</b> |

Table 2:  $\pi^2$  significantly improves long-context reasoning ability of QWEN3-4B-INSTRUCT-2507 and GPT-OSS-20B. We evaluate our models and baselines on in-domain benchmarks LongSeal, OfficeQA, and our  $\pi^2$ -Bench; as well as out-of-domain benchmarks LongBenchV2 and Oolong. For overall comparison, we report the weight-less average performance of models across benchmarks. **Bold-faced numbers** indicate the best among the group. We confirm that the average improvements brought by SFT on  $\pi^2$  are significant at  $\alpha = 0.01$  under t-test (i.e., with 99% confidence level).

ranging from 500 to 10K tokens with a median of 1,480 tokens.<sup>3</sup> 88.4% of the questions requires at least 3 reasoning hops. More details on  $\pi^2$  are shown in Appendix A.

## 2.5 $\pi^2$ -Bench and Quality of $\pi^2$

To provide an additional measurement of long-context reasoning capabilities and to qualitatively assess the quality of our curated dataset, we held out and revised a challenging subset of  $\pi^2$  to serve as a benchmark  $\pi^2$ -Bench. In each sample of  $\pi^2$ -Bench, models will be given the *compacted context* and the question, and prompted to predict a short answer. Similar to previous works (Pham et al., 2026; Chen et al., 2025b; Zhu et al., 2024), we used an LLM grader (with backbone GPT-OSS-120B) to judge the answer correctness. The grader template is provided in Appendix F. **For convenient notation,  $\pi^2$  is referred to the training set of the whole  $\pi^2$  dataset by default, which is disjoint from the introduced  $\pi^2$ -Bench.**

**Sample selection and quality control of  $\pi^2$ -Bench:** We deliberately prioritize difficulty and quality. In particular, we randomly selected 150 and 100 samples respectively from two pools: (1) samples originating from expanded tables with complexity ratings of 4 or 5, and (2) remaining samples with the highest complexity rating of 5. Each selected sample was reviewed by us to verify question quality and answer correctness against the Internet, including reference Wikipedia pages. That means, *QA samples in  $\pi^2$ -Bench are grounded to real-world information*. During this review process, 22 samples were discarded due to being difficult to revise without significant reconstruction. Among the remaining 228 samples, 209 required no revision, as the generated questions were already well-formed and complex as expected. The other 19 samples required human intervention to either refine the question or correct the answer. Among 228 human-validated samples, we randomly select 50 samples for the development set. Remained 178 samples form the test set.

**Notes on  $\pi^2$ 's quality:** A high yield was observed in this review process, where 209 out of 250 selected samples (83%) required no revision. It shows the *effectiveness of our data pipeline for the intended verifiable, long-context, complex QA*.

<sup>3</sup>With respect to the o200k\_base encoding

### 3 Experiments

#### 3.1 Methods and Baselines

We perform supervised finetuning (SFT) with LoRA (Hu et al., 2022) on GPT-OSS-20B and QWEN3-4B-INSTRUCT-2507 with the 922 training samples of  $\pi^2$  to obtain our models  $\pi^2$ -20B and  $\pi^2$ -4B, using hyperparameters outlined in Appendix B. The SFT process for each model took 1.5 hours on 4 NVIDIA H200 GPUs with PyTorch DDP.<sup>4</sup> We report results with two reasoning effort configurations for  $\pi^2$ -20B (low and high), to assess the benefits of finetuning under different configurations.

We compare our finetuned models against the following baselines. **Base models:** The original GPT-OSS-20B and QWEN3-4B-INSTRUCT-2507 without any finetuning on  $\pi^2$ . **Larger open-source models:** GPT-OSS-120B (low reasoning effort) and QWEN3.5-35B-A3B-FP8, which represent a larger model scale and are models used for generating reasoning traces. **Proprietary models:** GEMINI-2.5-PRO and GEMINI 3.1 FLASH-LITE,<sup>5</sup> leading commercial LLMs.

For open-source models, we run evaluations using vLLM (Kwon et al., 2023). **We report the average performance of the base models and our finetuned models across three runs.** Other evaluations, which serves as upper bounds, are run *once*. We use models’ default (i.e., recommended) hyper-parameters to maximize their performance.

#### 3.2 Benchmarks and Metrics

We establish a comprehensive evaluation framework to assess our models and baselines across different dimensions of long-context reasoning. We consider both in-domain and out-of-distribution evaluations to understand the benefits of  $\pi^2$  and the generalization capabilities of our models.

**In-domain evaluation.** Our in-domain suite contains benchmarks that have a large portion concerning realistic long-context QA tasks, with evidence grounded in tables in the context. Along with our  $\pi^2$ -Bench’s test set, the suite includes LongSeal and OfficeQA. **LongSeal** (Pham et al., 2026) measures document-grounded QA in adversarial, noisy contexts. Here, each question’s context contains the ground-truth document and 12 deceptive distractors. We checked the overlap between reference pages of LongSeal and  $\pi^2$ , and confirmed no contamination. Meanwhile, **OfficeQA** (Opsahl-Ong et al., 2026) straightforwardly evaluates end-to-end analytical reasoning on historical U.S. Treasury Bulletin documents. We follow the “LLM with Oracle Parsed PDF Page(s)” setup.

**Out-of-domain evaluation.** We evaluate our models on two benchmarks that differ in style or domain from our training data to probe distributional generalization. **LongBenchV2** (Bai et al., 2025) is a large-scale multiple-choice benchmark for reasoning over massive realistic contexts, covering various topics such as codebase QA, conversational, legal, financial reasoning, etc. Its “Medium” split (215 samples, context lengths 32k–128k) is used for evaluation. On the other hand, **Oolong** (Bertsch et al., 2025) is an analytical reasoning benchmark requiring atomic text analysis and aggregation across long contexts. 400 Oolong-Synth test samples with 64k-token contexts are considered.

**Metrics.** We follow each benchmark’s *recommended metrics*. Notably, we use the provided LLM grader (with backbone GPT-OSS-120B) to report the accuracy on  $\pi^2$ -Bench and LongSeal, while for others, we report scores from the **deterministic** graders. Because our focus is complex reasoning over long context rather than ultra-long-context tasks, inputs whose context exceeds 66% of a 128K-token window are compacted with BM25+ in the same way as described in §2.3.

<sup>4</sup>Done with unsloth and torchrun

<sup>5</sup>Access via the [official Gemini API](#)

---

**Question:** According to the WHO, in which year were the highest number of measles cases recorded worldwide?

**Correct Answer:** 1980

**Context:** (Omitted due to its length. A snapshot of evidence is shown in Figure 5)

---

**GPT-OSS-20B (low):**

From the WHO data presented in the documents, the peak in worldwide measles cases occurred in 2019, with a total of approximately 869,770 reported cases globally. This is higher than any preceding or following year reported in the sources (e.g., 2018 had 343,806 cases, and 2024 had 359,521 cases). Therefore, according to WHO statistics, 2019 was the year with the highest number of measles cases worldwide.

---

**$\pi^2$ -20B (low):**

The WHO's own global measles surveillance data (published in Document 1) lists the total number of reported measles cases worldwide by year:

1980 – 3,852,242 cases, 1990 – 1,325,074 cases, 2000 – 853,479 cases

2010 – 343,806 cases, 2019 – 869,770 cases, 2020 – 159,073 cases

Among these figures, the largest reported number is the 3,852,242 cases recorded for the year 1980. No other year shown in the official WHO data exceeds this count, and [31 words omitted for brevity]. Therefore, according to the WHO's own records, the highest number of measles cases worldwide occurred in 1980.

---

Table 3:  $\pi^2$  equips models with stronger long-context retrieval and reasoning. We show the response of GPT-OSS-20B and  $\pi^2$ -20B (with low reasoning effort) on a sample of LongSeal. The base model fail to reason and retrieve the correct piece of evidence, as it mistakes the phrase "climbed to 869,770 in 2019" as the signal for the correct answer, thus missing out other evidence. On the other hand,  $\pi^2$ -20B successfully identifies all pieces of evidence and reason thoroughly through them to derive the correct answer. That is a core capability targeted by our data curation pipeline.

### 3.3 Results

Table 2 presents our main results, demonstrating that  $\pi^2$  significantly improves the long-context reasoning capability of base models across benchmarks. Below are our key findings.

**Consistent improvements across models and benchmarks.** Supervised finetuning on  $\pi^2$  yields improvement for models in different ways.  $\pi^2$ -20B significantly outperforms the base GPT-OSS-20B on the in-domain benchmarks (LongSeal, OfficeQA, and  $\pi^2$ -Bench), with improvements of 5.31, 8.02, and 7.15 points under low reasoning effort, and even larger gains of 9.97, 9.16, and 3.11 points under high reasoning effort. Regarding the out-of-domain benchmarks, the improvement is, though less pronounced, still remarkable. While  $\pi^2$ -20B (high) slightly improves on Oolong and LongBenchV2,  $\pi^2$ -20B (low) has a 5.03-point improvement on Oolong and a 5.77-point improvement on LongBenchV2. Meanwhile,  $\pi^2$ -4B also shows a large improvement on LongSeal but not on OfficeQA and  $\pi^2$ -Bench among in-domain benchmarks. We attribute this phenomenon to the fact that QWEN3-4B-INSTRUCT-2507 is a smaller model and has a weaker reasoning capability than GPT-OSS-20B, thus benefiting less from the reasoning patterns in  $\pi^2$ . This phenomenon has been observed in previous works Li et al. (2025); Ye et al. (2025). Nonetheless,  $\pi^2$ -4B interestingly outperforms the base model 5.03 and 3.90 points on LongBenchV2 and Oolong, respectively. Although each model absorbs a different skill set from  $\pi^2$ , the overall improvements are consistent, indicating that the benefits of  $\pi^2$  are not limited to a specific data distribution or pre-trained model.

**Narrowing down the gap with proprietary and larger models.** While proprietary models GEMINI 3.1 FLASH-LITE, GEMINI-2.5-PRO, and larger open-source models such as QWEN3.5-35B-A3B-FP8 maintain an overall lead,  $\pi^2$ -20B (high) substantially narrows the gap. Notably, on complex reasoning benchmarks such as LongSeal (61.09 vs. 58.50 for QWEN3.5-35B-A3B-FP8) and OfficeQA (41.91 vs. 55.74 for QWEN3.5-35B-A3B-FP8), the gap is largely reduced, demonstrating that high-quality training data can partially compensate for model scale.

---

**Question:** Who holds the all-time record at the Grammys for the most wins in the album of the year category?

**Correct Answer:** Serban Ghenea

**Context:** (Only show the most important evidence. Omitted remaining text for brevity.)

[In Document 1: "Grammy Award for Album of the Year" Wikipedia page, as a document in the context] ... [Serban Ghenea](#) (as engineer/mixer) is the most frequent winner in this category with five awards. Taylor Swift (as performer); John Hanes (as engineer/mixer); and Tom Coyne and Randy Merrill (as mastering engineers) have won the award four times ...

---

**GPT-OSS-20B (low):**

From the provided context ([Document 1](#), [Document 5](#), [Document 7](#), [Document 9](#)) we see that:

- Frank Sinatra, Paul Simon, and Stevie Wonder each have three wins.
- Taylor Swift has won the award four times: for *\*Fearless\** (2010), *\*1989\** (2016), *\*Folklore\** (2021), and *\*Midnights\** (2024).

Thus, [Taylor Swift](#) holds the all-time record for the most Album of the Year wins.

---

**$\pi^2$ -20B (low):**

The answer can be found in the "Achievements" section of **Document 1** (Wikipedia – Grammy Award for Album of the Year). In that table the entry for *\*Taylor Swift\** reads: Taylor Swift | 4 (2010, 2016, 2021, 2024) | *Fearless* (2008), 1989 (2014), *Folklore* (2020), *Midnights* (2022) | 7 (2010, 2014, 2016, 2021, 2022, 2024, 2025). The table also shows *\*Frank Sinatra\** with 3 wins and other artists with up to 3, confirming that **"no one has more than four wins"**.

As a cross-check, **BusinessInsider** article ([Document 2](#)) states: "Taylor Swift... has won album of the year four times: *Fearless* (2010), 1989 (2016), *Folklore* (2021) and most recently *Midnights* (2024)." (54 words omitted for brevity) ... The conclusion is that **Taylor Swift** holds the all-time record for the most Album of the Year wins at the Grammys."

---

Table 4:  $\pi^2$  **does not help models overcome reporting bias**. We show the response of GPT-OSS-20B and  $\pi^2$ -20B (with low reasoning effort) on another sample of LongSeal. Both models are inclined to Taylor Swift due to the **reporting bias**, thus **fail to reason** that the question does not target a specific role in music production.

**Analyses.** We present two samples from the LongSeal benchmark in [Table 3](#) and [Table 4](#) to qualitatively illustrate the impact of  $\pi^2$  on model’s reasoning capability.  $\pi^2$ -20B demonstrates skills for comprehensive evidence retrieval and complex reasoning, thus not making a mistake like the base model. That is a core capability targeted by our data curation pipeline. Nonetheless, as not primarily designed for,  $\pi^2$  does not help models overcome reporting bias. Furthermore, the stronger reasoning capability of our finetuned models is accompanied by a cost of overthinking, as the output length of  $\pi^2$ -20B is significantly longer than the base model. We provide more analysis on this phenomenon in [Appendix C.7](#).

## 4 Ablation Study

To further understand the contribution of  $\pi^2$  and its individual components, we conduct ablation studies on GPT-OSS-20B (low reasoning effort).

### 4.1 Reasoning Style versus Faithful Reasoning Patterns

We investigate whether forward sampling (with or w/o rejection) could serve as an effective alternative to backward translation for reasoning trace generation. Also, we re-evaluate the base model with a carefully-designed prompt (adapted from the prompt used for reasoning trace generation), to examine the contribution of the reasoning style encoded in  $\pi^2$ .

Specifically, for forward sampling baselines, we prompt GPT-OSS-120B to generate reasoning traces for the 922 training samples. For the one w/o rejection, we directly finetune GPT-OSS-20B on these sampled reasoning traces regardless of their predicted answers’ correctness. For the one with rejection, we compare the predicted answer against the ground-truth answer using the grader template of  $\pi^2$ -Bench. Out of the 922 samples, 673 has reasoning

|                              | LongSeal     | OfficeQA     | $\pi^2$ -Bench | LongBenchV2  | Oolong       | Avg.         |
|------------------------------|--------------|--------------|----------------|--------------|--------------|--------------|
| base                         | 34.38        | 22.38        | 60.86          | 38.76        | 27.96        | 36.87        |
| Back Translation ( $\pi^2$ ) | <b>39.69</b> | 30.40        | <b>68.01</b>   | <b>44.52</b> | 32.98        | <b>43.12</b> |
| Sampling w/ Rejection        | 35.60        | 21.28        | 60.25          | 41.46        | 25.91        | 36.90        |
| Sampling w/o Rejection       | 35.83        | 19.42        | 63.43          | 43.66        | 29.37        | 38.34        |
| Prompting                    | 35.04        | 20.00        | 63.48          | 42.79        | 26.57        | 37.58        |
| QA only                      | 18.55        | 14.29        | 54.49          | 36.28        | 27.46        | 30.21        |
| Table as context             | 34.25        | 20.61        | 62.36          | 37.67        | 23.74        | 35.73        |
| Expanded tables              | 35.43        | <b>30.53</b> | 65.71          | 44.34        | <b>37.17</b> | 42.64        |
| LongMIT                      | 28.46        | 22.22        | 55.93          | 37.85        | 22.55        | 33.40        |

Table 5: **Ablation study** on different SFT approaches with GPT-OSS-20B, and a comparison to a previous long-context reasoning dataset LongMIT. The results show that *faithful reasoning patterns grounded on realistic long context are crucial elements of  $\pi^2$* . Also,  $\pi^2$  certainly fills the gap between previous long-context reasoning datasets and real-world demands.

traces passed the verification and retained for SFT. We finetune GPT-OSS-20B under the same configuration as our main experiment.

As shown in Table 5, SFT on sampling, without or with rejection, yields substantially worse results than SFT with backward-translated traces, as the average performance drops to 38.34/36.90 from 43.12. We attribute the performance degradations respectively to *data correctness* and the *selective nature of rejection sampling*. While there is no doubt that data correctness hinders the effective learning, discarding traces that produce incorrect answers also limit the learning to easier datapoints in the population. In contrast, *backward translation* generates reasoning traces for all samples, including challenging ones, by conditioning on the given answer, thereby *exposing the model to a broader spectrum of reasoning patterns*. This hypothesis is partially validated by the fact that, SFT on sampling w/o rejection perform slightly better than SFT on sampling with rejection, as the former is exposed to more challenging samples.

Meanwhile, there is no significant difference between the "Prompting" baseline and the original baseline. It suggests that the reasoning style does not have a significant contribution. In another words, faithful reasoning patterns learned from  $\pi^2$  must be necessary factors for improvement.

## 4.2 Contribution of Table Expansion and Context Realisticness

We further ablate the table expansion, QA generation, and context components in  $\pi^2$  to study their contributions. We compare GPT-OSS-20B and  $\pi^2$ -20B with the following baselines (with the same SFT setup to  $\pi^2$ -20B):

- *Expanded tables*: As we notice that the proportion of samples with expanded tables in  $\pi^2$  is modest, we generate more data and filter for samples with expanded tables only, until we obtain 1000 samples.
- *QA only*: For each sample, we only use the question and the answer for finetuning. The long context and reasoning trace are not included.
- *Table as context*: For each sample, we alter the free-form context by its table representation for reasoning trace generation and finetuning.

Results are shown in Table 5. The *Expanded tables* baseline consistently performs better than the base model and is competitive to  $\pi^2$ -20B on some benchmarks. It suggests that though valuable, having more samples with expanded tables is not a must. We note that this result is not trivial, as samples with expanded tables are more challenging than samples with non-expanded tables and we expect GPT-OSS-20B can learn more from these samples than  $\pi^2$ .

On the other hand, the *QA only* baseline performs significantly worse than the base model across all benchmarks. It is surprising, as this finetuning means to give knowledge advan-

tage over the base model while not affecting the model’s reasoning behavior. Likewise, the *Table as context* baseline slightly degrade from the base model. These results highlights that faithful reasoning grounded on realistic long context is the crucial element of  $\pi^2$ .

### 4.3 Comparison to Previous Long-Context Reasoning Datasets

We compare  $\pi^2$  with an open-source reasoning dataset. We choose LongMIT (Chen et al., 2025a) because of its accessibility and its focus on multi-hop question answering grounded in real documents. We finetune GPT-OSS-20B on the 800 training samples of LongMIT with the same configuration as our main experiment, and evaluate on the same benchmarks. As shown in Table 5, *SFT on LongMIT yields worse results than both the base model and  $\pi^2$ -20B on all benchmarks*. This suggests that while LongMIT provides some benefits for long-context reasoning, it does not fully capture the complexity and diversity of reasoning patterns required for realistic applications, which  $\pi^2$  is designed to address.

## 5 Related Work

Despite rising demand for better long-context reasoning in LLM, the availability of high-quality training data for such capabilities remains limited. Recent efforts address this gap through various approaches.

A line of works, including HotPotQA (Yang et al., 2018), MuSiQue (Trivedi et al., 2022), LongMIT (Chen et al., 2025a), CGMIS (Sun et al., 2026), and QwenLong (Shen et al., 2025), focus on generating multi-hop question-answer pairs grounded in real documents. They used pre-defined reasoning patterns to structure the question answering samples from real documents, or parse the real documents into concept graphs, then generate question-answer pairs with verifiable reasoning traces. However, the reasoning complexity of these datasets is limited by low branching factors of the reasoning trees, which restricts the diversity of reasoning paths and the breadth of inference required.

In contrast, IN2 (An et al., 2024), LongFaith (Yang et al., 2025b), and LoongRL (Wang et al., 2025) attempted to generate long-context reasoning data by synthesizing long contexts where evidence is inserted at different positions. TableLong (Xie et al., 2026) instead worked directly with tables, generating complex reasoning questions and training models directly in structured table data.

Meanwhile, OpenR1 (Hugging Face, 2025) and OpenThoughts (Guha et al., 2026) provided high-quality reasoning data in Math, Coding, and Science by collecting questions from various sources, generating answers with LLMs, and filtering with meticulous schema. These approaches enable the generation of complex reasoning data with high branching factors, but the long contexts are mostly synthetic or limited to structured, easy-to-verify domains such as table question answering, natural science, and coding.

*Our approach,  $\pi^2$ , fills a missing piece by generating verifiable complex reasoning data with high branching factors, grounded in realistic long contexts in open-domain tasks, and guided by common data structures.* This allows us to systematically train LLMs to perform complex reasoning over realistic long contexts, improving their core reasoning capabilities.

## 6 Conclusion

We presented  $\pi^2$ , a structure-based data curation pipeline that leverages Wikipedia tables to generate high-quality long-context reasoning data. Through dual-path verification and reasoning trace back-translation, our pipeline produces diverse complex questions with reliably correct answers over realistic long context. Supervised finetuning on  $\pi^2$  yields consistent improvements, with  $\pi^2$ -20B and  $\pi^2$ -4B achieving +6.25% and +3.37% average gains respectively over the base models. Faithful reasoning patterns discovered by back translation and grounded realistic long context play a crucial role for the improvements.

## Ethics Statement

We present a data curation method for improving reasoning capabilities in LLMs. We identify the following ethical considerations.

**Data source and privacy.** Our dataset is constructed from publicly available Wikipedia content and open-source LLMs. No private or personally identifiable information is included in our data pipeline or evaluation benchmarks. All used data and tools are openly accessible.

**Biases in synthetic data generation.** The reasoning traces and expanded table columns are synthetically generated by LLMs. While we implement verification steps to ensure factual correctness, synthetic data may inadvertently propagate biases present in the underlying model or source content. Wikipedia itself reflects historical and cultural biases that may be inherited in our dataset.

## Acknowledgments

As described in the main paper, we use LLMs as a part of our data pipeline and our benchmark’s evaluation. AI assistants (e.g., ChatGPT) are used to accelerate the coding, rephrasing, and grammar checking of our paper.

## References

- Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, and Jian-Guang Lou. Make your llm fully utilize the context. *ArXiv*, abs/2404.16811, 2024. URL <https://api.semanticscholar.org/CorpusId:269362524>.
- Anthropic. System card: Claude opus 4.6, 2026. URL <https://www-cdn.anthropic.com/14e4fb01875d2a69f646fa5e574dea2b1c0ff7b5.pdf>. [Accessed 30-03-2026].
- Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, et al. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3639–3664, 2025. URL <https://aclanthology.org/2025.acl-long.183/>.
- Amanda Bertsch, Adithya Pratapa, Teruko Mitamura, Graham Neubig, and Matthew R Gormley. Oolong: Evaluating long context reasoning and aggregation capabilities. *arXiv preprint arXiv:2511.02817*, 2025. URL <https://arxiv.org/abs/2511.02817>.
- Zhi Chen, Qiguang Chen, Libo Qin, Qipeng Guo, Haijun Lv, Yicheng Zou, Hang Yan, Kai Chen, and Dahua Lin. What are the essential factors in crafting effective long context multi-hop instruction datasets? insights and best practices. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27129–27151, Vienna, Austria, July 2025a. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1316. URL <https://aclanthology.org/2025.acl-long.1316/>.
- Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, et al. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent. *arXiv preprint arXiv:2508.06600*, 2025b. URL <https://arxiv.org/abs/2508.06600>.
- Etash Kumar Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, et al. Openthoughts: Data recipes for reasoning models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL <https://openreview.net/forum?id=7xjoTuaNmN>.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL <https://openreview.net/forum?id=nZeVKeeFYf9>.
- Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL <https://github.com/huggingface/open-r1>.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023. URL <https://dl.acm.org/doi/10.1145/3600006.3613165>.
- Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. Small models struggle to learn from strong reasoners. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 25366–25394, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1301. URL <https://aclanthology.org/2025.findings-acl.1301/>.
- Yuanhua Lv and ChengXiang Zhai. Lower-bounding term frequency normalization. In *Proceedings of the 20th ACM international conference on Information and knowledge management, CIKM '11*, pp. 7–16. ACM, October 2011. doi: 10.1145/2063576.2063584. URL <http://dx.doi.org/10.1145/2063576.2063584>.

- John X. Morris. gpt-oss-20b-base, 2025. URL <https://x.com/jxmnop/status/1955436075947847843>.
- Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candes, and Tatsunori Hashimoto. s1: Simple test-time scaling. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20275–20321, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1025. URL <https://aclanthology.org/2025.emnlp-main.1025/>.
- OpenAI. Gpt-5.4 thinking system card, 2026. URL <https://deploymentsafety.openai.com/gpt-5-4-thinking/gpt-5-4-thinking.pdf>. [Accessed 30-03-2026].
- OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, et al. gpt-oss-120b and gpt-oss-20b Model Card, 2025. URL <https://arxiv.org/abs/2508.10925>.
- Krista Opsahl-Ong, Arnav Singhvi, Jasmine Collins, Ivan Zhou, Cindy Wang, Ashutosh Baheti, Owen Oertell, Jacob Portes, Sam Havens, Erich Elsen, et al. Officeqa pro: An enterprise benchmark for end-to-end grounded reasoning. *arXiv preprint arXiv:2603.08655*, 2026. URL <https://arxiv.org/abs/2603.08655>.
- Thinh Pham, Nguyen Nguyen, Pratibha Zunjare, Weiyuan Chen, Yu-Min Tseng, and Tu Vu. Sealqa: Raising the bar for reasoning in search-augmented language models. In *ICLR*, 2026. URL <https://openreview.net/forum?id=zWb7ueH16c>.
- Weizhou Shen, Ziyi Yang, Chenliang Li, Zhiyuan Lu, Miao Peng, Huashan Sun, Yingcheng Shi, Shengyi Liao, Shaopeng Lai, Bo Zhang, Dayiheng Liu, Fei Huang, Jingren Zhou, and Ming Yan. Qwenlong-l1.5: Post-training recipe for long-context reasoning and memory management. *ArXiv*, abs/2512.12967, 2025. URL <https://api.semanticscholar.org/CorpusId:283897395>.
- Zechen Sun, Zecheng Tang, Juntao Li, Wenpeng Hu, Wenliang Chen, Zhunchen Luo, and Qiaoming Zhu. Cgmis: Concept-graph based multi-hop instructions synthesis for enhancing long-context reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(39):33153–33161, March 2026. ISSN 2159-5399. doi: 10.1609/aaai.v40i39.40599. URL <http://dx.doi.org/10.1609/aaai.v40i39.40599>.
- The Gemini Team. Gemini 3.1 pro: A smarter model for your most complex tasks, 2026. URL <https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-3-1-pro/>. [Accessed 30-03-2026].
- Harsh Trivedi, Niranjana Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl\_a\_00475. URL <https://aclanthology.org/2022.tacl-1.31/>.
- Siyuan Wang, Gaokai Zhang, L. Zhang, Ning Shang, Fan Yang, Dongyao Chen, and Mao Yang. Loongrl: Reinforcement learning for advanced reasoning over long contexts. *ArXiv*, abs/2510.19363, 2025. URL <https://api.semanticscholar.org/CorpusId:282272075>.
- Huaibing Xie, Guoliang Zhao, Yang Liu, Shihan Dou, Siming Huang, Yan-Yun Xiao, Shaolei Wang, Yiting Liu, Chengui Zhang, Shao-Hua Liu, and Pluto Zhou. Probing how scalable table data enhances general long-context reasoning. In *unknown*, 2026. URL <https://arxiv.org/abs/2603.21719>.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a. URL <https://arxiv.org/abs/2505.09388>.

- Cehao Yang, Xueyuan Lin, Chengjin Xu, Xuhui Jiang, Shengjie Ma, Aofan Liu, Hui Xiong, and Jian Guo. Longfaith: Enhancing long-context reasoning in llms with faithful synthetic data. In *Annual Meeting of the Association for Computational Linguistics*, 2025b. URL <https://arxiv.org/abs/2502.12583>.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, Brussels, Belgium, October–November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL <https://aclanthology.org/D18-1259/>.
- Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*, 2025. URL <https://openreview.net/forum?id=T2TZ0RY4Zk>.
- Alex L Zhang, Tim Kraska, and Omar Khattab. Recursive language models. *arXiv preprint arXiv:2512.24601*, 2025. URL <https://arxiv.org/abs/2512.24601>.
- Andrew Zhu, Alyssa Hwang, Liam Dugan, and Chris Callison-Burch. Fanoutqa: A multi-hop, multi-document question answering benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 18–37, 2024. URL <https://aclanthology.org/2024.acl-short.2/>.

| Topics                         | %    |
|--------------------------------|------|
| Reference and Event Pages      | 40.1 |
| Culture, Religion and Mind     | 16.4 |
| Science and Technology         | 12.4 |
| History, Politics and Conflict | 12.2 |
| Geography and Transport        | 8.8  |
| Entertainment and Media        | 4.5  |
| Law, Business and Institutions | 2.3  |
| Sports                         | 1.3  |
| Nature and Environment         | 1.1  |
| Education and Biography        | 0.8  |
| Total                          | 100  |

Table 6: Distribution of Wikipedia pages’ topics

| LLM-judge Rating   | 3   | 4   | 5    |
|--------------------|-----|-----|------|
| Complexity         | 194 | 762 | 216  |
| Self-containedness | 0   | 123 | 1049 |
| Naturalness        | 0   | 354 | 838  |

Table 7: Distribution of LLM-judge automatic ratings of all 1,172 QA samples in  $\pi^2$ . There are 68 training samples and 38 evaluation samples with all ratings of 5.

## A $\pi^2$ ’s Curation and Statistics

As stated, we choose Wikipedia as the source for table collection. That ensures high data quality, diversity, realism, and factual grounding. Our dataset covers a variety of topics, with a large portion belonging to events (such as "2026 Winter Olympics"), as shown in Table 6. In the table collection step, we discard too small or too large tables (i.e., with fewer than 5 rows or 2 non-index columns, or more than 30 rows or 6 columns) to avoid simplicity or over-complication. In the step of QA generation in our data pipeline, a question is only kept if the automatic ratings for complexity, self-containedness, and naturalness are at least 3, 4, 4 respectively. Again, we elaborate the meaning of self-containedness as "providing all necessary parameters, definitions, and background information within itself, requiring no external links, data, references, or prior knowledge to be understood". Table 7 shows the distribution of automatic ratings of 1,172 validated QA samples in the whole  $\pi^2$  dataset. For the automated web searching to form questions’ contexts, we use LangSearch.<sup>6</sup>

To facilitate further understanding of our dataset, we show a sample reasoning trace generated by QWEN3.5-35B-A3B-FP8 for the question Among the destinations where Uruguayan citizens can pick up pre-approved visas upon arrival, which country mandates that a sponsor must submit the visa application at least 48 hours prior to the traveler’s arrival? with the correct answer Eritrea and the main reference Wikipedia page Visa requirements for Uruguayan citizens.

### Sample Reasoning Trace

#### \* Step 1: Understanding the Question and Planning Strategy

The question asks to identify a specific country from a list of destinations where Uruguayan citizens can pick up "pre-approved visas upon arrival." The distinguishing criterion is that for this specific country, a **\*\*sponsor\*\*** must submit the visa application at least **\*\*48 hours prior\*\*** to the traveler’s arrival. My strategy is to scan the provided documents for a section specifically

<sup>6</sup><https://langsearch.com/>

discussing "Pre-approved visas" or "Visas pick-up on arrival," identify the list of countries mentioned there, and then check the specific conditions for each country to find the one matching the 48-hour sponsor requirement.

\* Step 2: Locating Relevant Information in Documents

I need to find a document that lists countries allowing pre-approved visas to be picked up on arrival.

- Scanning **Document 1** (Live in Uruguay): This document details entry procedures, visas, and residency types for entering Uruguay, but it does not list countries where **Uruguayans** can pick up visas. It focuses on incoming travel to Uruguay.

- Scanning **Document 3** (ETIAS for Uruguayan Citizens): This discusses ETIAS requirements for Europe, not general pre-approved visas for other countries.

- Scanning **Document 4** (Wikipedia - Visa requirements for Uruguayan citizens): I see a section titled "Pre-approved visas pick-up". Let's examine this section closely.

- Scanning **Document 5** (Visa Requirements for Uruguayan Citizens): This document categorizes countries by visa type (Not Required, On Arrival, eVisa) but does not have a specific "Pre-approved visas pick-up" section with detailed conditions like the Wikipedia page.

- Scanning **Document 6** (Administrative entry restrictions): This appears to be a duplicate or very similar version of Document 4. It also contains a "Pre-approved visas pick-up" section.

Let's focus on **Document 4** (and confirmed by **Document 6**) as they explicitly contain the section header "Pre-approved visas pick-up".

\* Step 3: Extracting Candidates from the "Pre-approved visas pick-up" Section

In **Document 4**, under the section "**Section: Pre-approved visas pick-up**", the text states: "Pre-approved visas can be picked up on arrival in the following countries instead in embassy or consulate."

The table lists the following countries:

1. Bhutan
2. Cameroon
3. Eritrea
4. Liberia
5. Nigeria
6. Sudan
7. Turkmenistan

Now I must check the specific condition for each of these candidates to see which one requires a sponsor to apply 48 hours in advance.

\* Step 4: Evaluating Candidates Against the Criteria

Let's examine the conditions listed for each country in the table found in **Document 4** (and verified in **Document 6**):

- **Bhutan**: The condition is "For a maximum stay of 15 days if the application was submitted at least 2 and a half months before arrival and if the clearance was obtained."

- **Analysis**: This mentions a time frame (2.5 months), but it refers to the applicant submitting the application, not necessarily a sponsor, and the timeframe is different (months vs hours). This does not match the "48 hours prior" criterion.

- **Cameroon**: The condition is "Must hold approval from the General Delegate of Security."

- **Analysis**: No mention of a sponsor or a 48-hour deadline.

- **Eritrea**: The condition is "Must have a sponsor who must submit an application at least 48 hours before arrival."

- **Analysis**: This matches all criteria perfectly.
    - Requires a **sponsor**.

- The sponsor must **submit the application**.
- The deadline is **at least 48 hours before arrival**.

- **Liberia**: The condition is "Available only if arriving from a country without a diplomatic mission of Liberia and if a sponsor obtained an approval."

- **Analysis**: Mentions a sponsor, but no specific 48-hour submission deadline is listed.

- **Nigeria**: The condition is "Holders of a visa application who have a Nigerian company taking responsibility for them."

- **Analysis**: Mentions a company (similar to a sponsor), but no specific 48-hour timeline is mentioned.

- **Sudan**: The condition is "Holders of an entry permit issued by the Ministry of Interior."

- **Analysis**: No mention of a sponsor or 48-hour rule.

- **Turkmenistan**: The condition is "Holders of an invitation letter of the local company that was approved by the Ministry of Foreign Affairs."

- **Analysis**: Mentions a company/invitation, but no 48-hour deadline.

\* Step 5: Cross-Verification  
Let's verify this finding using **Document 6** ("Administrative entry restrictions"), which contains the same table.  
Under the section "**Section**: Pre-approved visas pick-up":

| Country                                                                                              | Conditions                                                                           |
|------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| ---                                                                                                  | ---                                                                                  |
|  | Must have a sponsor who must submit an application at least 48 hours before arrival. |

The information is consistent across both documents. The specific phrasing "Must have a sponsor who must submit an application at least 48 hours before arrival" is unique to Eritrea in this list.

\* Step 6: Final Conclusion  
Based on the analysis of the "Pre-approved visas pick-up" section in Document 4 and Document 6, the only country that mandates a sponsor to submit the application at least 48 hours prior to arrival is Eritrea.

\* Step 7: Answer Formulation  
The correct answer is Eritrea.

## B Training Details

We finetune two base models on our  $\pi^2$  training data: GPT-OSS-20B (a 20B parameter model using MXFP4 quantization) and QWEN3-4B-INSTRUCT-2507 (a 4B parameter instruction-tuned model). Both models are loaded via Unsloth for memory efficiency.

We use LoRA (Hu et al., 2022) with rank  $r = 8$  and target modules  $\{q, k, v, o\}$  for attention layers, with alpha set to  $2r$ . Training is performed with a total batch size of 4 and a learning rate of  $2e-5$  with linear warmup for 5 steps followed by linear decay. The batch size per device and the gradient accumulation depend on the availability of hardware. We train for 200 steps, approximately 1 epoch over our 922 training samples. The maximum sequence length is set to 128K tokens to accommodate our long reasoning traces. We use the AdamW optimizer with weight decay of 0.001 and train in mixed precision (bfloat16 for QWEN3-4B-INSTRUCT-2507, 4-bit quantization for GPT-OSS-20B). The training takes approximately 2.5 hours on 4 NVIDIA H200 GPUs using PyTorch DDP. Figure 3 shows the training dynamics of GPT-OSS-20B on our dataset.

We format training samples using chat templates with special token delimiters. For GPT-OSS-20B models, we use reasoning effort tokens (low or high) to control the thinking budget; for

Figure 3: Training dynamics of GPT-OSS-20B on  $\pi^2$ .

|                                          | LongSeal     | OfficeQA     | $\pi^2$ -Bench | LongBenchV2  | Oolong       | Avg.         |
|------------------------------------------|--------------|--------------|----------------|--------------|--------------|--------------|
| <i>GPT-OSS-20B Low reasoning effort</i>  |              |              |                |              |              |              |
| base                                     | 34.65        | 21.63        | 62.92          | 39.53        | 31.03        | 37.95        |
| $\pi^2$ -20B                             | 38.19        | 26.53        | <b>67.05</b>   | 46.05        | <b>33.31</b> | 42.23        |
| SFT on $\pi^2$ -100                      | <b>40.94</b> | 18.75        | 63.48          | 47.57        | 25.72        | 39.29        |
| Self-distillation                        | 37.8         | <b>28.89</b> | 66.29          | <b>48.45</b> | 30.34        | <b>42.35</b> |
| <i>GPT-OSS-20B High reasoning effort</i> |              |              |                |              |              |              |
| base                                     | 52.17        | 37.84        | 72.57          | 46.07        | 35.86        | 48.9         |
| $\pi^2$ -20B                             | <b>64</b>    | <b>46.58</b> | <b>77.27</b>   | 46.32        | 31.95        | <b>53.22</b> |
| SFT on $\pi^2$ -100                      | 58.01        | 41.94        | 74.21          | <b>51.7</b>  | 32.9         | 51.75        |
| Self-distillation                        | 55.56        | 29.33        | 72.12          | 44.38        | 35.9         | 47.46        |

Table 8: **Ablation study** with GPT-OSS-20B (both low and high reasoning efforts) on a minimal  $\pi^2$ -100 training set and self-distillation. **Bold-faced numbers** indicates the best among the group.  $\pi^2$ -100 brings improvements to both reasoning effort configuration, though the improvements are less than those from the full  $\pi^2$  training set. Meanwhile, self-distillation shares similar improvements with our original setup for low reasoning effort configuration, but does not hold for high reasoning effort configuration.

QWEN3-4B-INSTRUCT-2507 models, we use the standard `<|im_start|>` format. We compute loss only on the assistant response (reasoning traces and final answer), not on the user prompt or system message.

Note that for GPT-OSS-20B, evaluation uses the same reasoning effort configuration as training. That is,  $\pi^2$ -20B with high reasoning effort was trained with high reasoning effort and evaluated with high reasoning effort; similarly for low. This ensures consistency between the learned reasoning patterns and the evaluation setup.

## C Additional Experiments and Analyses

Beyond the main experimental results presented in §3, we conducted several additional studies to better understand the properties of our finetuned models and the effectiveness of  $\pi^2$ . These supplementary experiments are summarized in Table 8 and Table 9.

### C.1 Efficiency of High-Quality Data.

We investigate whether a small set of our highest-quality samples can boost model performance. Specifically, we finetune GPT-OSS-20B 25 steps on just 100 samples with the sum of quality ratings at least 14 over 15, selected from the  $\pi^2$ 's training set. As shown in Table 8, even with this minimal subset ( $\pi^2$ -100), the finetuned model achieves noticeable improvements over the base one: +1.34 points average with low reasoning effort (39.29 vs. 37.95) and +2.85 points with high reasoning effort (51.75 vs. 48.90). While these gains

|                                         | LongSeal | OfficeQA | $\pi^2$ -Bench | LongBenchV2 | Oolong | Avg.  |
|-----------------------------------------|----------|----------|----------------|-------------|--------|-------|
| GPT-5.4                                 | n/a      | n/a      | 81.46          | n/a         | n/a    | n/a   |
| QWEN3-4B-INSTRUCT-2507                  |          |          |                |             |        |       |
| base                                    | 33.07    | 14.88    | 56.18          | 37.56       | 29.67  | 34.27 |
| <b>SFT, lr=2e-5 (ours)</b>              | 40.94    | 13.58    | 55.62          | 39.25       | 35.48  | 36.97 |
| SFT, lr=2e-6                            | 29.13    | 13.47    | n/a            | 37.85       | 30.82  | n/a   |
| SFT, lr=2e-4                            | 28.93    | 53.09    | n/a            | 30.52       | 29.6   | n/a   |
| SFT noisy context                       | 36.22    | 16.73    | 54.49          | 36.45       | 33.28  | 35.43 |
| RLM*                                    | 28.85    | 28.57    | 53.71          | 28.57       | 48.95  | 37.73 |
| SFT + RLM*                              | 34.54    | 21.43    | 49.38          | 32.54       | 40.51  | 35.68 |
| Prompting                               | 30.71    | 15.98    | 54.49          | 32.82       | 29.91  | 33.88 |
| GPT-OSS-20B <i>Low reasoning effort</i> |          |          |                |             |        |       |
| base                                    | 34.65    | 21.63    | 62.92          | 39.53       | 31.03  | 37.95 |
| <b>SFT, lr=2e-5 (ours)</b>              | 38.19    | 26.53    | 67.05          | 46.05       | 33.31  | 42.23 |
| SFT, lr=2e-6                            | 40.16    | 22.04    | n/a            | 39.07       | 28.22  | n/a   |
| SFT, lr=2e-4                            | 54.97    | n/a      | n/a            | 38.78       | n/a    | n/a   |
| SFT noisy context                       | 40.55    | 24.18    | 65.73          | 44.19       | 36.11  | 42.15 |
| RLM*                                    | 45.28    | 31.82    | 63.48          | 34.04       | 41.55  | 43.23 |
| SFT + RLM*                              | 49.21    | 31.82    | 64.2           | 29.29       | 38.55  | 42.61 |
| Prompting                               | 35.04    | 20.00    | 63.48          | 42.79       | 26.57  | 37.58 |

Table 9: **Additional experiments.** Due to the prohibitively expensive cost of GPT-5.4, we only run experiment on our  $\pi^2$ -Bench. **Rows respectively represent:** (1) *base*: the base model, (2) *SFT, lr=2e-5 (ours)*: our finetuned model with default setup, (3) *SFT, lr=2e-6*: our finetuned model with a smaller learning rate 2e-6, (4) *SFT, lr=2e-4*: our finetuned model with a larger learning rate 2e-4, (5) *SFT noisy context*: our finetuned model with data which the context is naively prepared, (6) *RLM*: Recursive Language Models, (7) *SFT + RLM*: the combination of our SFT and RLM, where we first apply RLM on top of our finetuned model for inference, and (8) *Prompting*: the base model but evaluated with a dedicated prompt adapted from the reasoning trace generation prompt. **n/a** entries indicate that the model’s performance could not be reliably measured due to excessive inference time, or unconsidered evaluation after hyperparameters probing. *The results suggest that our SFT with the default setup (lr=2e-5) still yields the best performance*, while other variants of SFT and RLM show mixed results. (\*) We note that the result of Recursive Language Models (RLM) is more than 50% incomplete due to the context overflow problems of the `rlms==0.1.1` package that we encountered. Thus, although its results are competitive, the comparison between its and ours is not apple-to-apple.

are smaller than those achieved with the full 922-sample training set, they demonstrate that our curation pipeline produces high-quality data where even a small subset provides meaningful supervision. This validates the efficiency of  $\pi^2$ , as reasonable improvements can be achieved with minimal computational cost.

## C.2 Effective Self-Distillation with Curated QA Data.

A feature of  $\pi^2$  is that the reasoning traces are generated by large teacher models (GPT-OSS-120B and QWEN3.5-35B-A3B-FP8). Here, we investigate whether smaller models can benefit from their own generated traces through self-distillation. We use the base GPT-OSS-20B to generate reasoning traces for QA samples and finetune it on these self-generated traces. As shown in Table 8, self-distillation yields impressive results with low reasoning effort: the average performance improves from 37.95 (base) to 42.35 (+4.40), competitive with our

original SFT setup (42.23). This demonstrates *the value of our core QA curation* that facilitates effective self-distillation. However, with high reasoning effort, self-distillation shows no improvement performance (47.46 vs. 48.9). It indicates that the thinking patterns scaffolded by GPT-OSS-20B itself are not new to the model with high reasoning effort, unlike those scaffolded by the larger GPT-OSS-120B. Our results suggest the importance of using stronger teacher models for effective learning in reasoning-intensive models.

### C.3 Effect of Learning Rate

We investigate the impact of learning rate (lr) on model performance during supervised finetuning. In addition to our default learning rate of  $2e-5$ , we experiment with both smaller ( $2e-6$ ) and larger ( $2e-4$ ) learning rates. As shown in Table 9, the smaller learning rate of  $2e-6$  possibly results in underfitting, where the model fails to learn sufficient reasoning patterns from  $\pi^2$  and only performs comparably to or marginally better than the base model.

Conversely, the larger learning rate of  $2e-4$  causes training instability, even when setting the weight decay of 0.01 and 0.1. We hypothesize that this case is akin to the *alignment reversal* phenomenon described in Morris (2025), where the finetuned model’s weights diverge substantially from the base model, leading to unexpected generation behaviors. In particular, the model produces excessively long outputs that frequently cause API timeout errors during evaluation, making reliable performance measurement impossible. These findings underscore the importance of careful learning rate selection for long-context reasoning finetuning, and we chose  $lr=2e-5$  as our default after initial hyperparameter probing.

### C.4 Effect of Context Cleanliness

As mentioned in §2, we implement a strict validity check to ensure that all evidence remains present within the final context for reasoning trace generation. Here, we study possible benefit of the reversely defined noisy context setup, where we do not explicitly keep evidence passages during the compaction. That mimics noisy real-world QA problem, as some explicit evidences to be missing in the final context, and the model must resort to alternative sources of information scattered across the context or embedded in model weights. The results, labeled as "SFT noisy context" in Table 9, suggests that using the clean context setup is still preferred, though the observed gaps are not statistically significant.

### C.5 Validation of $\pi^2$ -Bench

Our held-out benchmark  $\pi^2$ -Bench serves both as an evaluation tool and a validation of our data curation pipeline’s quality. The GPT-5.4 performance confirms that the curated questions are solvable and still having room for improvement. After finetuning base models on  $\pi^2$ , performance on  $\pi^2$ -Bench improves from 62.92 to 67.05 for GPT-OSS-20B, demonstrating that the reasoning patterns learned from  $\pi^2$  transfer effectively to our curated benchmark.

### C.6 Comparison with Recursive Language Models

Recursive Language Models (RLM) (Zhang et al., 2025) represent an orthogonal approach to handling long-context reasoning through recursive context processing. We compare our SFT approach with RLM and investigate potential synergies between the two methods. For RLM experiments, we use the publicly available `rlms==0.1.1` package and apply it to both base models and our finetuned models during inference. We note that the result of RLM is more than 50% incomplete, due to a context overflow problem of the current `rlms==0.1.1` package that we encountered. Thus, the comparison between RLM and our models are not apple-to-apple.

As shown in Table 9, RLM applied to base GPT-OSS-20B achieves an average of 43.23, which is competitive with our SFT approach (42.23). Notably, RLM shows particular strength on Oolong (41.55 vs. 33.31 for SFT), suggesting that recursive processing is effective for

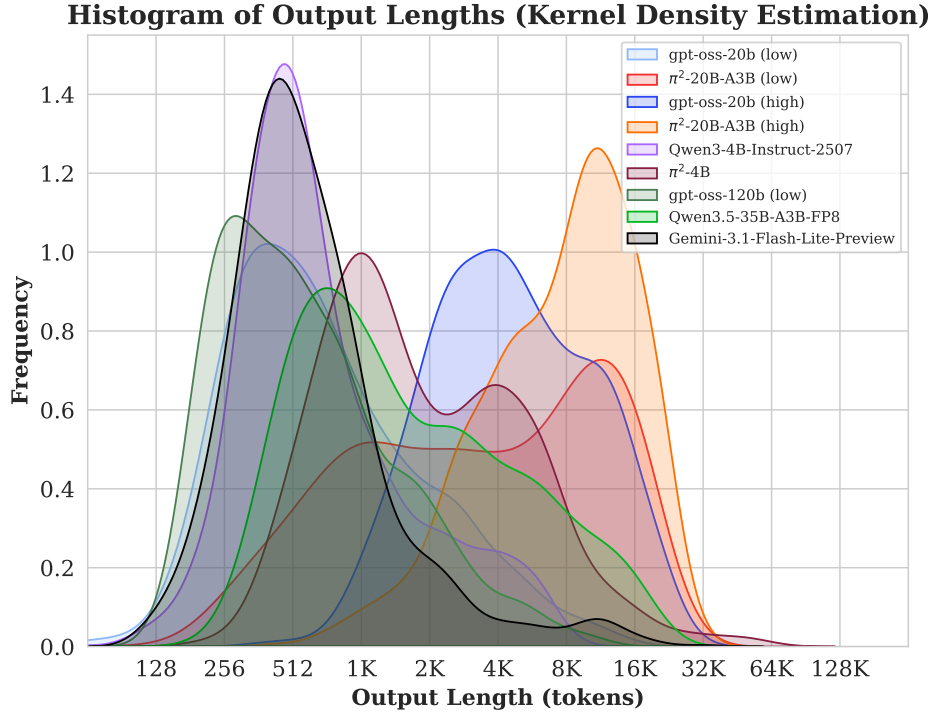


Figure 4: Comparing output length (in tokens) over all benchmarks of our models, base models, larger open-source models, and leading commercial GEMINI 3.1 FLASH-LITE. Here, the token count includes thinking tokens. Our models generate relatively more tokens than the base models (so-called *overthinking tax*). Interestingly, our  $\pi^2$ -4B shares a similar output length distribution as its teacher model QWEN3.5-35B-A3B-FP8, suggesting successful reasoning skill transfer. Meanwhile, GEMINI 3.1 FLASH-LITE desirably generate the least tokens among considered models and have the best performance. That is what we ultimately aim for.

distributional reasoning tasks. When we apply RLM on top of our SFT models ("SFT + RLM"), the results are mixed, as some benchmarks show improvement while others degrade. That is explainable, as our finetuned model may have different instructional behaviors from the base one that affects its use of RLM pipeline. Overall, our findings suggest that RLM and SFT on  $\pi^2$  address long-context reasoning through different mechanisms, and a naive combination does not yield consistent synergistic benefits.

### C.7 Output Length Analysis

Given that our training data contains relatively long reasoning traces (median 1,480 tokens), we analyze whether finetuning induces changes in output behavior. Figure 4 compares the output token distributions (including thinking tokens) across base models, our finetuned models, larger open-source models, and commercial baselines.

The analysis reveals that our finetuned models consistently generate longer outputs than their base counterparts. For GPT-OSS-20B, the output length increases substantially after SFT, especially for low reasoning effort. On the other hand,  $\pi^2$ -4B exhibits an output length distribution remarkably similar to its teacher model QWEN3.5-35B-A3B-FP8, suggesting that our data pipeline successfully transfers reasoning style from larger to smaller models. Meanwhile, GEMINI 3.1 FLASH-LITE desirably generate the least tokens among considered models and have the best performance. That is what we ultimately aim for.

| FT                                                    | LR   | Weight Decay | Steps | Avg.  |
|-------------------------------------------------------|------|--------------|-------|-------|
| <i>Probing with GPT-OSS-20B High reasoning effort</i> |      |              |       |       |
| SFT                                                   | 2e-4 | 0.1          | 200   | n/a   |
| SFT                                                   | 2e-5 | 0.01         | 200   | n/a   |
| SFT                                                   | 5e-6 | 0.01         | 200   | 53.5  |
| <i>Scaling with GPT-OSS-20B High reasoning effort</i> |      |              |       |       |
| SFT                                                   | 5e-6 | 0.01         | 500   | n/a   |
| SFT                                                   | 5e-6 | 0.01         | 1000  | n/a   |
| <i>Trials with QWEN3-4B-INSTRUCT-2507</i>             |      |              |       |       |
| SFT                                                   | 2e-4 | 0.1          | 500   | 30.72 |
| SFT                                                   | 2e-5 | 0.01         | 500   | 36.71 |
| GRPO                                                  | 2e-5 | 0.01         | 500   | 31.68 |

Table 10: **Attempts at Larger Improvements.** Additional experiments probing learning rate, scaling up training steps with GPT-OSS-20B, and trials with QWEN3-4B-INSTRUCT-2507. Column "FT", "LR", "Avg." respectively denotes the finetuning method, learning rate, and average performance of each experiment. The *n/a* entries indicate that the model’s performance could not be reliably measured due to excessive inference time. Overall, our attempts in scaling up the training time with more data did not yield further improvements.

While longer outputs are not inherently desirable, they reflect the complex reasoning patterns learned from our training data, as shown in ???. We note that in some samples, fine-tuned models exhibit overthinking or over-verbose, which represents an area for future improvement in balancing thoroughness and conciseness.

### C.8 Statistical Significance of Results

Following the approach of Muennighoff et al. (2025), we conduct a statistical significance t-test using 10,000 bootstrap samples drawn from all benchmark results. The improvements achieved by our finetuned models over base models are statistically significant at  $\alpha = 0.01$  for both GPT-OSS-20B (under both reasoning effort configurations) and QWEN3-4B-INSTRUCT-2507, providing strong evidence that the observed performance gains are genuine and not attributable to random variation.

## D Attempts at Larger Improvements

Because our data pipeline is fully open source and thus our dataset can be scaled up naturally, following Guha et al. (2026), we attempted at larger improvements through scaling up the training data. We have further processed 30000+ pages from Wikipedia, having totally 80000+ qualified tables and 14000+ validated QA samples with reasoning traces. In this section, we mostly work with GPT-OSS-20B (high reasoning effort), as it is relatively the best base model we finetuned in this paper.

**Probing hyperparameters:** We observed significant increase in completion tokens count after finetuning, even excessively so that the generation are not able to complete (as SFT with learning rate  $2e-4$ ). We attributed the problem to weight divergence of the finetuned model and the base model. Thus, we first optimized learning rate and weight decay to avoid that problem in longer finetuning. We fixed the optimizer as AdamW with gradient clipping at 1.0, and the learning rate scheduler as linear warmup for 5 steps followed by linear decay. We also fixed the total number of finetuning steps to 1500, stopping step at 200, global batch size at 4, and LoRA rank at 8. Results are shown in Table 10. Among probed learning rate and weight decay pairs, we find that only the learning rate of  $5e-6$  and weight decay of 0.01 lead to an overall improvement which is aligned with our main experiments (53.5 vs. 53.22) while having the inference time reasonable. Thus, we use that pair for scaling up the training.

**Scaling up training steps:** However, further increasing the training steps to 500 or 1000 with the chosen setup again leads to excessive inference time. That is likely because the model weights diverge more from the base model as training goes on, which leads to more verbose generation.

**Trials with QWEN3-4B-INSTRUCT-2507:** We also attempted at improving QWEN3-4B-INSTRUCT-2507 with more training steps and with GRPO. For GRPO, we adapt  $\pi^2$ -Bench’s LLM Grader prompt to a rule-based function to compute reward of rollouts. While we did not encounter excessive inference time problem with QWEN3-4B-INSTRUCT-2507, the results are not better than the default SFT setup in our main experiments. For the GRPO-finetuned model, we observe a stable training reward from the start to the end of the training process.

**Possible explanation:** While we will further investigate the excessive inference time problem of GPT-OSS-20B in future work, we would like to comment on the result of QWEN3-4B-INSTRUCT-2507. The above results suggest that simply scaling up the training time with more data does not necessarily lead to larger improvements. We hypothesize that the difficulty and task diversity of the training data are also key factors in driving performance gains. Our dataset is limited in the question type (by design) and limited in difficulty (as the evaluation on  $\pi^2$ -Bench showed). Thus, we believe that our future work should focus on improving the difficulty and diversity of training samples.

## E Limitations

While our work demonstrates strong empirical results, several limitations should be acknowledged.

**Context length scope.** Our study focuses on long-context reasoning within the 128K-token context range, which covers common context window limits but does not address extremely long contexts (e.g., million-token scale). While our pipeline can theoretically generate data for longer contexts, we leave exploration of this regime for future work.

**Quality and yield.** Despite our dual-path verification and quality filtering, approximately 17% of initially generated samples required human revision in our benchmark curation. While this demonstrates the effectiveness of our pipeline, it also indicates room for improvement in automatic quality control.

**Evaluation setup.** We evaluate models on long-context reasoning tasks in context-provided QA setup. Performance on other long-context capabilities (e.g., retrieval accuracy, long-context free-form generation) is not directly assessed.

## F Prompts

We show the prompts used in our experiments in this section.

### Table expansion

You are a creative and grounded brainstorming partner for enriching tables in Wikipedia pages.

You will be given:

- a table in HTML format,
- some metadata about the table,
- summaries of the Wikipedia pages referenced by a column of the table.

Your task is to enrich the table by adding one or several (up to 3) new columns that capture common attributes from the Wikipedia summaries which are not already semantically reflected in the existing columns of the table.

```

**Action plan**:
- Identify one or several common attributes from the Wikipedia summaries. Each attribute must be:
  - relevant to the data in the table and provide additional insights to the readers,
  - **not already present or partially reflected in the existing columns of the table**,
  - not in conflict with the existing data in the table (e.g., statistics from different years compared to similar existing attributes).
- Extract the relevant data/values from the summaries to populate the new columns.

- Values must be short pieces of content (e.g., a number, a phrase, a several-word string), directly retrieved from the summaries, no extra notes, and should not require additional interpretation or analysis.
- Return the enriched table in HTML format, ensuring that the new columns are seamlessly integrated with the existing columns.
- **Provide a list of notes on the enriched table** about any cautions or assumptions made during the enrichment process, for examples, in the follow cases:
  - Statistics in a new column are from different years, so they may not be directly comparable. You should provide such a note with specific year information, e.g., "- Attribute X: row 1's value is from 2020, while row 2's value is from 2022."
  - In case there are typos in the original table, you are allowed to modify these typos. Don't need to report the incident. Typos cases include but are not limited to:
    - Misspelled words -> Fix the typos (e.g., "Spanyol" -> "Espanyol")
    - Inconsistent (and not sensical) statistics -> Resolve the inconsistent or the typos (e.g., in a numeric column, values are in different units, or one uses `,` as thousand separator while others use `.` , etc.)

NOTE:
- Existing column names may not fully reflect the content of the column, so you should also consider the actual data in the columns when determining whether an attribute is already present.
- There are some common taxonomies of attributes that are often present in Wikipedia summaries, such as:
  - Person: teams/companies/organizations, awards, career milestones, etc.
  - Organization: industry, headquarters location, revenue, number of employees, etc.
  - Event: date, location, participants, outcome, etc.
  - Geographical entity: population, area, GDP, neighboring entities, etc.
  - Knowledge: year, field, key contributions, inventors, etc.

Now, enrich the table based on the above instructions and the following input.

[[ ## table ## ]]
{table}

[[ ## metadata ## ]]
{metadata}

[[ ## column_name ## ]]
{column_name}

[[ ## summaries ## ]]
{summaries}

```

## QA generation

You are an expert designed for advanced data analysis.

You are given a table in Markdown format, a column name, and some context about the table, which reflect real information from the Internet.

Your task is to generate **a complex, self-contained, and natural question** about the data along with a SQL implementation to answer it.

**Strict Requirements:**

1. **Question Design:**

1.1. Generate ONE single-focus and concise question.

- DON'T concatenate multiple sub-questions with "and" or "which... and which".
- Split multi-part questions into separate questions. For example, instead of "which X and which Y", ask only "which X".
- Instead of "which X and which Y when X", just ask "When X, then which Y?" to maintain the complexity and multi-hop reasoning, but avoid asking two separate questions in one.
- DON'T: "Which team has the highest average score and which team has the lowest average score?", "What is the year when the Lansing fare per mile reached its highest recorded value, and what was the difference in fare per mile between Lansing and Grand Rapids in the year?"
- DO: "Which team has the highest average score?", "Which team has the lowest average score?", "In the year when the Lansing fare per mile reached its highest recorded value, what was the difference in fare per mile between Lansing and Grand Rapids?"

1.2. **Complexity:** Ensure the question probes deeply into the data's nuances.

- It should involve multi-hop reasoning and information aggregation, by embedding multiple constraints or conditions in the question.
- Example: "in the year when X reached its maximum", "for the team with the lowest average score in X", "among the teams with average score above 80", etc.
- It can involving (conditional) **counting or information aggregation** (e.g., sum, max, min, average, etc.) to increase the complexity.
- Nonetheless, the question is still as meaningful and natural as being asked by real data analysts.

1.3. **Self-containedness:** The question should be fully self-contained, providing all necessary context and details within the question itself, without requiring external information or references to the table or metadata.

- Utilize specific information in the table and metadata to design the question. Assuming that the question will be given without any context, and one can still fully understand it.
- For example, given a table with a section name "Number of athletes by National Olympic Committee". phrase "ranked 31 to 45" should be more specific to explain which ranking is referred to, such as "ranked 31 to 45 in the number of athletes sent to the Olympics".
- Must have phrases about time, location, or taxonomy etc. to narrow down the scope of the question.
- Example: "In the year when X reached its highest value", "In competition X, among the teams with average score above 80", "for the team with the lowest average score in X", etc.
- The wording **should NOT** refer to the table or any metadata items, as real users do NOT have access to these information when being asked.
- AVOID phrases like "which column", "in the column of", "the value in column X", or, "as listed in (somewhere)", "section", etc.
- AVOID words like 'listed', 'list', 'table', 'column', 'row', 'field', 'section', 'caption', 'metadata', etc., which implicitly or explicitly refer to the table or metadata.

1.4. **\*\*Naturalness\*\***: The question should be phrased in a natural and conversational manner, as if it were being asked by a real data analyst or researcher.

- Use simple, clear terminology. Avoid jargon, use layman-term descriptions instead.
- The wording must be fluent and coherent, with proper grammar and syntax. It should read like a question that a human would naturally ask when analyzing data.

1.5. If the input ``condition_col`` is not "No condition column", the generated question MUST include a clear condition or constraint that references that column.

- The condition should be derived from the values present in that column (inspect the column contents) and may be:
  - a filter on categorical values (e.g., "for [the object of question] with [condition\_col] is 'Value'"),
  - a numeric comparison (e.g., "consider [the object of question] with [condition\_col] > 1000", "regarding ... among in the top 10% of [condition\_col]"),
  - a temporal or extrema-based condition (e.g., "in the year when [condition\_col] reached its maximum", "for the month with the lowest [condition\_col]"),
  - a ranking-based constraint (e.g., "among the top 3 entries by [condition\_col]").

1.6. Answer:

- The question must have a deterministic, single answer.
- Ensure the question is answerable using only data present in the table.
- Keywords of the question that will be used to determine the answer should be present in the table and not cover multiple values in the corresponding column.
  - E.g., "Existing" vs "Existing with temporary stands" should be treated as different keywords, and the question should be specific to one of them, such as "for the existing stadiums" or "for the stadiums with temporary stands".
  - For the ease of post evaluation, the answer must be a specific value (a **\*\*short\*\*** string, a number) or a small set of information (a list or an entity-value dictionary with no more than 3 elements or key-value pairs).
  - Example: `{{'team_home': 'Poli Ejido', 'team_away': 'Espanyol'}}`, "Espanyol", 83.6, ["Espanyol", "Poli Ejido"], etc.

1.7. **\*\*Excellent, Good and Bad Examples\*\***:

- EXCELLENT questions:
  - Based on the 2014 Pew Research Center survey data comparing the ethnic composition of Latter-day Saints in the U.S. to the general U.S. population in 2020, which ethnicity had the largest positive percentage point difference between its representation among Latter-day Saints and its representation in the general U.S. population?
  - Among the regional groupings for which both the International Monetary Fund and the World Bank provided data, which region showed the largest absolute difference between its 2025 IMF forecast and its 2024 World Bank estimate?
  - In the 2025-26 NCAA Division I women's basketball season, among the recorded upsets where a non-Division I team defeated a Division I opponent, which game featured the smallest point differential between the winner and the loser?
  - In the study of standard reduction potentials for aqueous iodine species, which chemical couple exhibits a potential of exactly 0 volts in basic conditions while simultaneously having the highest recorded potential among all such couples in acidic conditions?
  - Based on the 2022 cobalt production and reserves data reported by the USGS, what is the ratio of reserves to annual production of the group of all explicitly reported Asian countries (excluding aggregate categories like 'Other countries' or 'World total')?
- GOOD questions:
  - What was the amount spent in millions of nominal dollars by the highest spending U.S. Federal Department in the fiscal year of 1955?

- Who holds the all-time record at the Grammys for the most wins in the album of the year category?
- What is the current age of the oldest person to sail solo across the Pacific Ocean?
- How many NBA players have scored 60 or more points in a regular season game since 2023?
- Of the countries with a head of state assuming office prior to 2000, which are the five with the largest GDP per capita?
- How many current and former Real Madrid players are ranked in the top 10 of the 2025 Forbes list of the world's highest-paid athletes?
- BAD questions (and reason why each is bad):
  - Among the listed cities, which city has the highest population while being located in a state whose 2019 HDI is greater than 0.73? --> The phrase "Among the listed cities" is not clear enough to make the question fully self-contained.
  - Among the venues owned by AC Milan & Inter Milan for the 2026 Winter Olympics (as listed under a table in 'Milan cluster' section), which venue has the highest seating capacity? --> The note inside the parentheses explicitly refers to the table and section, which is not allowed.
  - Which venue in the Valtellina cluster has the highest total spectator capacity when the capacities of all its assigned events are summed together? --> Not fully self-contained.
  - What is the spectator capacity of the Verona Olympic Arena, the venue for the closing ceremony of the 2026 Winter Olympics? --> Too simple.

## 2. \*\*SQL Implementation:\*\*

- Write a SQL query assuming the table is loaded as `df` and the engine is SQLite.
- Ensure the query is syntactically correct and optimized for performance.
- Always wrap column names with quotes, as some column names may contain spaces or special characters.
- Ensure the query returns a single definitive value or a small result set.
- The query must produce a non-empty result.

Now, generate a question and its SQL implementation based on the above instructions and the following input.

```
[[ ## table ## ]]  
{table}
```

```
[[ ## column_name ## ]]  
{column_name}
```

```
[[ ## metadata ## ]]  
{metadata}
```

---

## Question quality checking

You are an expert in Q&A.

You are given a question, which is expected to be complex, self-contained, and natural.

Your task is to briefly discuss the question in the following criteria, then on a scale from 1 to 5 (1 is worst, 5 is best), rate the question on each criterion:

- Complexity: The question involves multi-hop reasoning and information aggregation.
- Self-containedness: The question provides all necessary parameters, definitions, and background information within itself, requiring no external links, data, references, or prior knowledge to be **understood**. Using the question, people

can expect to search for all necessary information to answer it from the Internet.

- Naturalness: The question is as natural as being asked by real human.

Now, evaluate the question based on the above criteria and provide a brief explanation for each rating.

```
[[ ## question ## ]]  
{question}
```

---

### Python solution generation

You are an expert in table question answering and problem solving with Python.

You are given a table in Markdown format, some metadata about the table, and a complex question about the table.

Your task is to generate a Python implementation to answer the question. You must name the final answer variable `final_answer_python`.

The answer is expected to be a specific value (a short string, a number) or a small set of information (a list or an entity-value dictionary with no more than 3 elements or key-value pairs). Please only pack the final answer in `final_answer_python` variable, and avoid putting any narration in `final_answer_python` (e.g., "the answer is ...", "the result is ...", etc.).

**Guidelines:**

- Use `pandas` to solve the problem, **assuming the table is already loaded into a DataFrame named `df`**.
- Skip the data loading step.
- **Break complex operations into small steps.** Each step should compute one logical operation.
- Assign each intermediate result to a named variable (e.g., `step1_filtered`, `step2_grouped`).
- **Do not reuse the same variable name for different steps.**
- If the change is in a row or column of an existing variable, create a new variable to store the updated version (e.g., `step1_filtered` -> `step2_filtered`).
- Each step should have a unique variable name and should reflect the meaning of the data in the variable.
- Display the intermediate results and final result by printing them.
- Comment your code to explain each step clearly.
- **Use the provided context about the table to narrate your solution.**
- Name the final answer variable `final_answer_python`.

Now, generate a Python solution based on the above instructions and the following input.

```
[[ ## table ## ]]  
{table}  
  
[[ ## metadata ## ]]  
{metadata}  
  
[[ ## question ## ]]  
{question}
```

You are an expert in data analysis.

You are given two results (tables, values, or dictionaries) from answering the same question about a dataset.  
Your task is to compare if the content/meaning of the two results are essentially the same.

**\*\*Metadata is provided\*\*** to help you understand what the question was asking and what the results represent.

**\*\*Comparison Guidelines:\*\***

- The two results may have different formatting (e.g., table vs. dictionary, different column order, different indexing)
- They may have slight numeric differences due to floating-point precision (e.g., 83.6 vs 83.60000000000001)
- As long as the semantic content of results regarding the question are the same, consider them equivalent
- Focus on whether both results answer the question correctly

**\*\*Examples of equivalent results:\*\***

- "Espanyol" vs. {"team\_home": "Poli Ejido", "team\_away": "Espanyol", "team\_advances": "Espanyol"} if the question is "Which team advances to the next round?" and both results indicate Espanyol advances
- {"party": "Democratic", "avg": 83.6} vs {"party": "Democratic", "avg": 83.60000000000001}
- A single-row table vs a dictionary with the same key-value pairs
- 248Cm vs. {}^{248}Cm vs. 248Cm (i.e., {}^{248}Cm): all of them represent the same chemical element isotope, just different in the way of writing.

Give a binary answer (True or False) and a short explanation for your judgment.

Now compare the two results based on the above instructions and the following input.

```
[[ ## result1 ## ]]  
{result1}
```

```
[[ ## result2 ## ]]  
{result2}
```

## Reasoning trace generation (Back Translation)

You are given a question, a set of context documents/chunks, and the correct answer.  
Your task is to generate a detailed, step-by-step reasoning trace that explains how to arrive at the answer by analyzing the context.

**## Input Format**

```
### **Context:**  
{context}
```

```
### **Question:**  
{question}
```

```
### **Correct Answer:**  
{answer}
```

**## Output Format Requirements**

Generate a natural-language reasoning trace following these structural patterns:

### 1. Step Numbering

- Use `\* Step X:` format for major steps
- Steps should progress logically from understanding the problem to arriving at the answer

### 2. Planning Steps (Initial Steps)

- Start by analyzing what the question is asking
- Identify what information needs to be extracted from the context
- Determine the search strategy (e.g., "find all candidates for each criterion")
- Example: "The question asks about X that fits Y criteria, so we need to find all candidates..."

### 3. Extraction & Navigation Steps

- Systematically review relevant documents/chunks
- Document your search process (e.g., "let's search for keyword X: found in documents A, B, C")
- For each document examined, state whether it contains relevant information
- Use phrases like:
  - "Let's find..."
  - "Document X says..."
  - "It fits/does not fit because..."
  - "Let's check..."
  - "Move to document..."

### 4. Iterative Filtering

- For multi-criteria questions, evaluate candidates against each criterion
- Narrow down the candidate list as you verify criteria
- Explicitly state when a candidate is eliminated and why
- State when a candidate passes a criterion

### 5. Evidence Citation

- Always cite specific documents/chunks when stating facts
- Quote relevant passages when they directly support conclusions
- Use formats like:
  - "Document X states: '...'"
  - "According to document X,..."

### 6. Self-Verification

- Include steps that cross-check findings
- Verify consistency across multiple sources when available
- Address potential contradictions explicitly

### 7. Reasoning Steps

- Perform virtual operations (filtering, sorting, comparing) explicitly
- When calculating, show the calculation
- When comparing, state the comparison clearly
- Build confidence through multiple verified facts before final answer

### 8. Conclusion

- The final step should clearly state the answer
- Briefly summarize the key evidence that leads to it

## Style Guidelines

1. **Natural and exploratory**: Write as if discovering the answer in real-time
2. **Self-contained**: Each step should be understandable without re-reading previous steps
3. **Critical evaluation**: Don't accept information at face value; evaluate relevance
4. **Document-focused**: Ground all conclusions in the provided context

5. **\*\*Conversational but precise and concise\*\***: Use natural language while maintaining factual accuracy and efficiency

#### ## Example Structure

---

- \* Step 1: [Understanding the question and planning strategy]
  - \* Step 2: [Search for candidates matching criterion A - examining documents]
  - \* Step 3: [Search for candidates matching criterion B - narrowing down]
  - \* Step 4: [Cross-referencing and verification]
  - \* Step 5: [Conclusion with answer]
- 

#### ## Important Notes

- The reasoning trace should be answer-agnostic in style (don't reveal you know the answer upfront)
- Show the discovery process as it would happen when solving the problem
- Include dead-ends and corrections when relevant (e.g., "Wait, this document is from 2018, so it may be outdated")
- If multiple interpretations exist, explore them before concluding

Now generate the reasoning trace for the given input.

England and Wales from 2007 to 2017 death occurred between two and three cases out of 10,000.<sup>[88]</sup> In children one to three cases out of every 1,000 die in the United States (0.1–0.2%).<sup>[89]</sup> In populations with high levels of malnutrition and a lack of adequate healthcare, mortality can be as high as 10%.<sup>[7][5]</sup> In cases with complications, the rate may rise to 20–30%. In 2012, the number of deaths due to measles was 78% lower than in 2000 due to increased rates of immunization among UN member states.<sup>[18]</sup> Between 2000 and 2016, global cases decreased by 84%; by 2019 cases had increased to a total of 870,000, the highest since 1996.<sup>[41]</sup>

| Reported cases <sup>[90]</sup> |           |           |         |         |         |
|--------------------------------|-----------|-----------|---------|---------|---------|
| WHO-Region                     | 1980      | 1990      | 2000    | 2010    | 2020    |
| African Region                 | 1,240,993 | 481,204   | 520,102 | 199,174 | 115,369 |
| Region of the Americas         | 257,790   | 218,579   | 1,754   | 247     | 9,996   |
| Eastern Mediterranean Region   | 341,624   | 59,058    | 38,592  | 10,072  | 6,769   |
| European Region                | 492,660   | 185,818   | 37,421  | 30,625  | 10,945  |
| Southeast Asia Region          | 199,535   | 224,925   | 78,558  | 54,228  | 9,389   |
| Western Pacific Region         | 1,319,640 | 155,490   | 177,052 | 49,460  | 6,605   |
| Worldwide                      | 3,852,242 | 1,325,074 | 853,479 | 343,806 | 159,073 |

[Two irrelevant paragraphs omitted for brevity]

Omitted Cases reported in the first three months of 2019 were 300% higher than in the first three months of 2018, with outbreaks in every region of the world, even in countries with high overall vaccination coverage where it spread among clusters of unvaccinated people.<sup>[94]</sup> The numbers of reported cases as of mid-November is over 413,000 globally, with an additional 250,000 cases in DRC (as reported through their national system), similar to the increasing trends of infection reported in the earlier months of 2019, compared to 2018.<sup>[9]</sup> In 2019, the total number of cases worldwide climbed to 869,770. The number of cases reported for 2020 is lower compare to 2019.<sup>[95]</sup> According to the WHO, the COVID-19 pandemic hindered vaccination campaigns in at least 68 countries, including in countries that were experiencing outbreaks, which caused increased risk of additional cases.<sup>[95][96]</sup>

Figure 5: The evidence as a part of the context of the question "According to the WHO, in which year were the highest number of measles cases recorded worldwide?" in the LongSeal benchmark. Retrieved from the Wikipedia page titled "Measles" on May 20, 2026. Given the context, the answer of the question is 1980. Green-highlighted text notes the evidence that leads to the correct answer, while the yellow-highlighted text notes the evidence that may lead to a misunderstanding and thus an incorrect answer.