LongMagpie: A Self-synthesis Method for Generating Large-scale Long-context Instructions

Chaochen Gao[1,2], **Xing Wu**[1,2,3] , **Zijia Lin**[4], **Debing Zhang**[3], **Songlin Hu**[1,2]
[1]Institute of Information Engineering, Chinese Academy of Sciences
[2]School of Cyber Security, University of Chinese Academy of Sciences
[3]Xiaohongshu Inc, [4]Tsinghua University
{gaochaochen,wuxing,husonglin}@iie.ac.cn
dengyang@xiaohongshu.com, linzijia@tsinghua.edu.cn

Abstract

High-quality long-context instruction data is essential for aligning long-context large language models (LLMs). Despite the public release of models like Qwen and Llama, their long-context instruction data remains proprietary. Human annotation is costly and challenging, while template-based synthesis methods limit scale, diversity, and quality. We introduce LongMagpie, a self-synthesis framework that automatically generates large-scale long-context instruction data. Our key insight is that aligned long-context LLMs, when presented with a document followed by special tokens preceding a user turn, auto-regressively generate contextually relevant queries. By harvesting these document-query pairs and the model's responses, LongMagpie produces high-quality instructions without human effort. Experiments on HELMET, RULER, and Longbench v2 demonstrate that LongMagpie achieves leading performance on long-context tasks while maintaining competitive performance on short-context tasks, establishing it as a simple and effective approach for open, diverse, and scalable long-context instruction data synthesis. Our code is available in `https://github.com/caskcsg/longcontext/tree/main/LongMagpie`.

1 Introduction

Large Language Models (LLMs) have demonstrated impressive capabilities across a wide range of tasks, with recent advancements significantly extending their context lengths (Liu et al., 2024a; Achiam et al., 2023; Google DeepMind, 2025). The ability to process long documents is essential for complex applications such as Longbook QA (Caciularu et al., 2023), document summarization (Wu et al., 2023), and code planning (Bairi et al., 2023). However, fine-tuning LLMs to leverage long contexts requires access to high-quality long-context instruction data (Chen et al., 2023b; Bai et al., 2024a). While the model weights of several open-source LLMs, such as Qwen (Yang et al., 2024) and Llama (Grattafiori et al., 2024), have been made publicly available, the corresponding instruction datasets for long-context training remain proprietary. This closed-data paradigm poses a substantial barrier to the advancement of open-source long-context models.

Existing methods for creating open-source instruction data face substantial limitations when extended to long contexts. (1) Human labor costs are prohibitively high for creating diverse, high-quality long-context instruction data. The annotation difficulty is substantially greater than for short-context data, requiring individuals to read documents spanning thousands of tokens before formulating instructions—a demonstrably challenging task. (2) Existing synthetic approaches, often relying on predefined templates (Sanh et al., 2021) or seed questions (Wang et al., 2022a), do not guarantee the diversity needed for effective long-context instruction. While existing projects (Li et al., 2025;
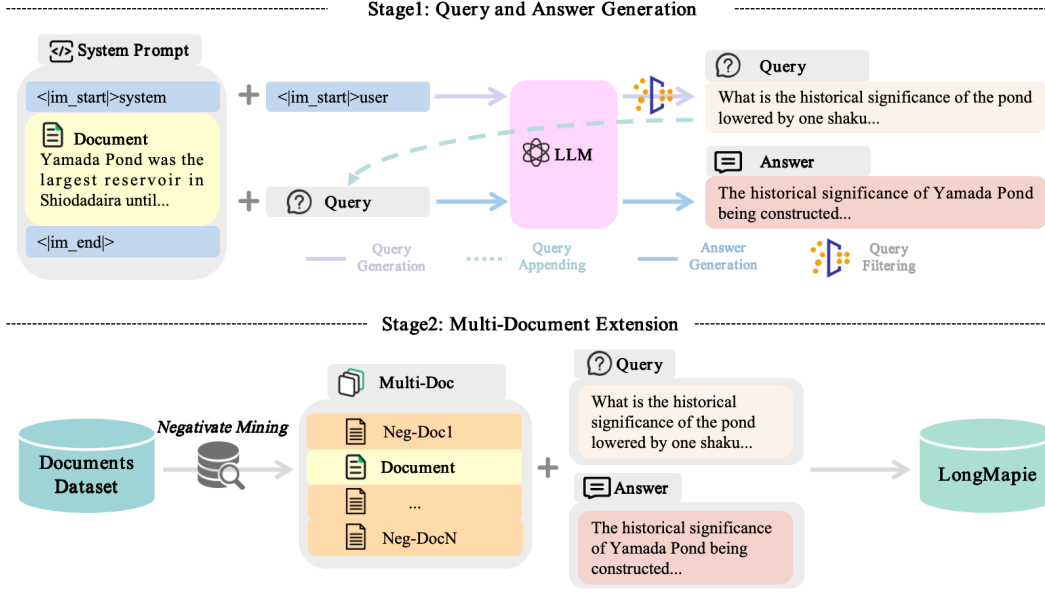
Figure 1: LongMagpie pipeline overview. Stage one: a document serves as a system prompt, a special user token triggers query generation, followed by the model response. Stage two: combines the query-response pair with the source document and sampled documents from the corpus to create challenging multi-document long-instruction data.

Xu et al., 2024b; Bai et al., 2024a) attempt to broaden seed data diversity, creating large-scale long-context instructions with high quality and diversity remains an expensive and time-consuming process.

A recently proposed self-synthesis method, Magpie (Xu et al., 2024c), has gained widespread attention for eliminating the need for seed instructions and prompt engineering required by previous approaches (Wang et al., 2022a; Li et al., 2025; Xu et al., 2024b; Bai et al., 2024a). It creates alignment data by prompting aligned LLMs with only special tokens preceding a user turn, leveraging their auto-regressive nature. Inspired by Magpie, we introduce **LongMagpie**, a self-synthesis method for generating large-scale long-context instruction data without human annotation or complex prompting. A key observation is that long-context understanding often involves document-based question answering, such as RAG or long document QA. Thus instruction-tuned LLMs such as Qwen (Yang et al., 2024) and Llama (Grattafiori et al., 2024) internalize patterns of document-query relationships during their long-context instruction training. Thus, when aligned models are presented with only a document, followed by the special tokens that typically precede a user query, they auto-regressively generate contextually relevant queries about that document. By leveraging this behavior, we can automatically create high-quality instruction-response datasets for long-context training without explicit prompting or manual intervention.

This approach offers advantages: it scales efficiently to generate diverse, high-quality long-context instructions without labor costs or complex prompt engineering; produces naturally varied queries that probe different aspects of documents; and eliminates complex pipeline components required by previous methods. Furthermore, we extend LongMagpie beyond single documents to multi-document contexts, creating more challenging scenarios that require distinguishing relevant information such as RAG Gao et al. (2023). This multi-document extension enhances the model's ability to handle complex real-world applications that frequently involve reasoning across multiple information sources while providing a natural way to increase context length and task difficulty without additional computational overhead.

To further balance the long-context and short-context capabilities, we introduce the p-Mix strategy, which addresses the performance degradation on short-context tasks when models are predominantly trained on long-context instructions. This strategy employs a probabilistic mixing approach that begins by prepending a short-context instruction to each training sequence, followed by a dynamic

sequence constructed through probabilistic sampling. Specifically, with probability P_L, a long-context instruction (generated by LongMagpie) is appended; otherwise, with probability $1 - P_L$, another short-context instruction is selected. This process continues iteratively until approaching the maximum sequence length L_{max}. p-Mix effectively prevents the model from overfitting to long-context patterns while maintaining strong performance across diverse task scenarios.

Through extensive evaluation on HELMET (Yen et al., 2024), RULER (Hsieh et al., 2024), and Longbench v2 (Bai et al., 2024b) benchmarks, we demonstrate that models trained on LongMagpie-generated data achieve leading performance. When incorporated with p-Mix, our approach maintains competitive performance on short-context tasks. We conduct detailed analytical experiments on the LongMagpie method to explain its effectiveness. The positive experimental results demonstrate that LongMagpie represents a meaningful step toward democratizing long-context capabilities for LLMs, making high-quality long-context instruction accessible to the broader research community.

Our main contributions are:

- We introduce a novel self-synthesis approach for generating high-quality long-context instruction data that leverages the auto-regressive nature of aligned LLMs, eliminating the need for human annotation or predefined examples.
- We propose the p-Mix technique, a probabilistic mixing strategy that effectively balances the model's performance on both long-context and short-context tasks.
- We conduct extensive evaluations demonstrating that models trained on LongMagpie-generated data achieve leading results on long-context benchmarks compared to existing methods.
- We provide in-depth analyses revealing the key factors contributing to LongMagpie's effectiveness, including query diversity and quality.

2 Method

This section introduces LongMagpie, our method for synthesizing long-context instruction data. We first describe the key insight of our approach, followed by the detailed pipeline of LongMagpie and p-Mix strategy for balancing long-context and short-context capabilities.

2.1 Key Insight: Auto-Regressive Document-Query Generation

The foundation of LongMagpie is a key observation about aligned long-context LLMs: when provided with a document followed by tokens that typically precede a user query (without the query itself), these models generate contextually relevant queries about that document. This behavior stems from the fact that long-context understanding often involves document-based question answering tasks such as RAG and long document QA. During instruction tuning, models like Qwen and Llama internalize document-query relationship patterns, enabling them to auto-regressively predict meaningful questions when presented with document-only contexts. This capability allows us to synthesize diverse, high-quality instruction data without human annotation, predefined templates, or seed questions.

Formally, for an aligned LLM $\mathcal{M}$ with vocabulary $\mathcal{V}$, we define the document-query generation process as follows: given a document $D = \{d_1, d_2, ..., d_n\} \in \mathcal{V}^n$ and pre-query template $T_{pre} = \{t_1, t_2, ..., t_m\} \in \mathcal{V}^m$ (containing tokens indicating a user or query role, e.g., `<|im_start|>user`), we provide input $X = D \oplus T_{pre}$, where $\oplus$ denotes sequence concatenation. The model then generates a sequence $Q = \{q_1, q_2, ..., q_k\} \in \mathcal{V}^k$ representing a query related to document D. This process can be described as:

$$p_{\mathcal{M}}(Q \mid D, T_{\text{pre}}) = \prod_{i=1}^{k} p_{\mathcal{M}}(q_i \mid D, T_{\text{pre}}, q_{<i}),\qquad(1)$$

This approach differs fundamentally from traditional prompt engineering or instruction-following, as we are not explicitly instructing the model to generate a query about the document. Instead, we leverage the model's learned patterns of document-query relationships that emerge from its instruction training.

2.2 LongMagpie Pipeline

The LongMagpie pipeline consists of two main steps: (1) query and answer generation, and (2) extension to a multi-document setting.

2.2.1 Query and Answer Generation

Document Preparation   We collect diverse documents from various domains and lengths, primarily using curated resources like Fineweb. These documents span domains including science, history, literature, and technical topics, with an average length of approximately 1.6k tokens in our primary dataset. This provides a range of context lengths while focusing on truly long-context scenarios.

Query Generation   For each document D, we construct an input sequence $X = D \oplus T_{pre}$, where T_{pre} contains tokens preceding a user query in the model's instruction template. For example, the tokens for Llama-3-Instruct model are `<|start_header_id|>user` and for Qwen-2.5-Instruct are `<|im_start|>user`. We pass X to the aligned LLM and sample a completion Q until an end-of-template token is generated or a maximum length is reached. This completion represents a contextually relevant query. By generating multiple queries per document with different sampling parameters, we create diverse document-query pairs that naturally vary in complexity.

Response Generation   For each document-query pair (D, Q), we construct a standard instruction prompt by combining the document, query, and tokens that precede an assistant response (e.g., `<|eot_id|><|start_header_id|>assistant<|end_header_id|>` for Llama-3-Instruct). We then generate a response R, forming a complete instruction triplet (D, Q, R) for long-context training. If the same model is used for both query and response generation, these steps can be consolidated without manual intervention.

Query Filtering   In query generation, we observed that LLMs occasionally continue the input document rather than generate queries, particularly when the model size is small. To ensure the quality of the generated queries, we applied two filtering strategies: (1) **Rule-based filtering**: we retain queries that end with a question mark as a simple heuristic to identify interrogative sentences; (2) **Length-based filtering**: we discard generated texts longer than 1.5k characters, as they are typically descriptive passages rather than valid queries.

2.2.2 Multi-Document Extension

To enhance task diversity and real-world applicability, we extend LongMagpie to multi-document settings. Many tasks require reasoning over several related documents rather than a single one. Our approach involves:

- Obtaining x documents $\{D_1, \ldots, D_x\}$ as negative documents via random sampling, where x is drawn uniformly from 0 to n (with $n = 0$ reducing to the standard single-document QA setting). .
- Concatenating documents using a special separator token (e.g., `<|doc_sep|>`) to form $D_{\text{multi}} = D_1 \oplus \texttt{<|doc\_sep|>} \oplus \cdots \oplus D_x$.
- Generating queries and responses as in the single-document pipeline, producing triples (D_{multi}, Q, R) requiring cross-document reasoning.

2.3 p-Mix: Balancing Long-Context and Short-Context Capabilities

Fine-tuning predominantly on long-context data degrades performance on short-instruction tasks (Bai et al., 2024a; Xu et al., 2024b). To balance these capabilities, we introduce p-Mix, a novel instruction data hybridization strategy. The core idea is twofold. First, to emulate the typical non-contextual start of general tasks, we sample a short-context instruction at the beginning of each training sequence. Second, we append subsequent data segments probabilistically to construct a mixed-context sequence up to length L_{max}. With probability P_L, a long-context instruction (generated by LongMagpie) is chosen; otherwise, with probability $1 - P_L$, another short-context sample is chosen. This process repeats until approaching the target sequence length, ensuring each instance starts with a short, context-free instruction followed by a dynamically mixed sequence of long and short segments. This prepares the model for diverse real-world scenarios. The procedure is formalized in Algorithm 1, and we conduct an ablation study of the parameters related to p-Mix in Appendix A.3.

3   Experiments

In this section, we describe our experimental setup, present our main results, and analyze the factors that contribute to LongMagpie's performance.

3.1   Experimental Setup

Dataset Generation   Using the LongMagpie pipeline described in Section 1, we generate a long-context instruction dataset using Qwen2.5-70B-Instruct, with documents sampled from FineWeb-Edu (Lozhkov et al., 2024). FineWeb-Edu is a subset of the FineWeb dataset, comprising 1.3 trillion tokens extracted from educational web content.

Compared Datasets   We compare LongMagpie-generated data against several widely used instruction datasets. These include datasets specifically designed for long contexts and standard short-context datasets adapted for long-context fine-tuning based on ProLong (Gao et al., 2024b).

- **Long Instruction Datasets** We compare with two long-context datasets: **ChatQA** (Xu et al., 2024b) combines multiple data sources, including LongAlpaca12k (Chen et al., 2023b) and GPT-4 samples from Open Orca (Lian et al., 2023), containing 1.5 million synthetic instructions. In this work, we refer to ChatQA2 as ChatQA by default; **LongAlign** (Bai et al., 2024a) generates questions and answers for long documents by prompting LLMs.
- **Short Instruction Datasets** Following findings that concatenated short instructions benefit long-context capabilities (Gao et al., 2024b), we include: **Tulu** (Lambert et al., 2024), an open-source collection based on Llama 3.1; **Magpie** (Xu et al., 2024c), a self-synthesis method using template prefixes; and **UltraChat** (Ding et al., 2023), comprising 1.5 million multi-turn dialogues. We concatenate samples from these datasets to reach the target context length during fine-tuning.

3.1.1   Model Training

We select `Llama-3-8B-NExtLong-512K-Base` (Gao et al., 2025) as our base model, which has undergone extensive long-context continued pre-training. The batch size is 4M tokens for 250 steps, a total of 1B tokens for baseline datasets and LongMagpie. The same training configuration is applied across all datasets to ensure a fair comparison. Further details are provided in Appendix A.1.

3.1.2   Evaluation Benchmarks

Long-context Evaluation   We evaluate our models on three comprehensive long-context benchmarks. These benchmarks provide a holistic assessment of models' abilities to utilize long contexts effectively across different tasks and complexity levels.

- **HELMET** (Yen et al., 2024) evaluates long-context models across diverse application-centered tasks with context lengths up to 128k tokens, using model-based evaluation that prioritizes complex tasks for better real-world performance prediction.
- **RULER** (Hsieh et al., 2024) provides fine-grained evaluation of long-context reasoning with synthetic tasks that offer flexible control over sequence length and complexity to identify performance bottlenecks beyond simple retrieval.
- **LongBench-v2** (Bai et al., 2024b), an upgrade to LongBench (Bai et al., 2023), assesses extremely long-context understanding (8k to 2M words) through 503 expert-validated questions across six categories, revealing a need for improved ultra-long reasoning capabilities.

Short-context Evaluation   To further evaluate the model's ability to follow short instructions, we select 7 widely-used short-context datasets: HellaSwag (Hel.) (Zellers et al., 2019), Lambada_OpenAI (Lam.) (Paperno et al., 2016), ARC-Challenge (AR-C.) (Clark et al., 2018), ARC-Easy (AR-E.), PIQA (Bisk et al., 2020), WinoGrande (Win.) (Sakaguchi et al., 2021), and Logiqa (Log.) (Liu et al., 2020).

Table 1: Main experimental results comparing LongMagpie with other methods on long-context and short-context benchmarks. Best scores in each column are bolded. LongAVG is the average of HELMET, RULER, and Longbench v2, ShortAVG is the average of different short-context tasks.

Dataset	Long Evaluation				Short Evaluation
	HELMET	RULER	Longbench v2	LongAVG	ShortAVG
Short Instruction Data					
Tulu	61.93	87.92	28.4	59.42	63.90
Magpie	60.18	87.06	31.4	59.55	63.32
UltraChat	60.55	83.85	30.4	58.27	**64.43**
Long Instruction Data					
ChatQA	60.23	89.82	30.8	60.28	63.58
LongAlign	57.79	86.08	24.5	56.12	60.97
LongMagpie	**62.10**	**91.17**	**34.4**	**62.56**	62.37
***p*-Mix: Long + Short Instruction Data**					
ChatQA + UltraChat	60.80	87.42	31.4	59.87	64.38
LongAlign + UltraChat	60.98	89.49	30.6	60.36	64.17
LongMagpie + UltraChat	**62.11**	**89.70**	33	**61.60**	64.10

3.2 Main Results

As shown in Table 1, models trained solely on **LongMagpie** data already set a leading performance on long-context evaluation, topping HELMET (62.10), RULER (91.17), LongBench-v2 (34.4) and the LongAVG score (62.56) within the *Long Instruction Data* group. The performance gains are substantial compared to existing long-context instruction datasets: LongMagpie outperforms ChatQA by +1.87 on HELMET, +1.35 on RULER, and +3.6 on LongBench-v2, yielding a +2.28 improvement on LongAVG. The gap is even more pronounced when compared with LongAlign, where LongMagpie delivers gains of +4.31 on HELMET, +5.09 on RULER, and +9.9 on LongBench-v2, resulting in a remarkable +6.44 improvement on LongAVG. The strong performance of LongMagpie on long-context tasks demonstrates the effectiveness of our self-synthesis approach for generating high-quality long-context instruction data without human annotation or seed examples.

Among the models trained with *p*-Mix strategy, which mixes LongMagpie with other short-instruction datasets, **LongMagpie + UltraChat** achieves the best or tied-best scores on HELMET (62.11), RULER (89.70) and LongAVG (61.60) among *all* mixed datasets. It also retains a competitive Short-AVG accuracy (64.10), only 0.33 below the overall best, confirming that 1) The long-context signals produced by our self-synthesis method are highly complementary to existing short-instruction data, and 2) The probabilistic mixing schedule effectively balances these two instruction regimes, yielding models that are robust across both ultra-long reasoning and everyday short-instruction scenarios. These results highlight the practical value of *p*-Mix: it preserves the strength of LongMagpie on long-context tasks while simultaneously mitigating the typical performance drop on short-context benchmarks. We provide further analysis to demonstrate the advantages of *p*-Mix compared to alternative mixing strategies in Section 4.2.

4 Ablation Studies

This section first analyzes the key configurations that influence LongMagpie's performance, then evaluates the quality and diversity of its generated queries, and finally assesses the its resource efficiency.

4.1 Impact of Different Multi-Document Settings

To increase instruction difficulty and further enhance the model's ability to capture long-range dependencies, we introduce a multi-document setting. With a certain probability, the document associated with a generated query-answer pair is mixed with x randomly sampled documents from the corpus, where x is drawn uniformly from 0 to n (with $n = 0$ reducing to the standard single-

Table 2: Detailed results for the impact of the maximum number of documents (n) in a user prompt.

n	HELMET	RULER	Longbench v2	LongAVG	ShortAVG
0	60.13	89.04	31.4	60.19	**63.20**
5	61.42	89.91	31.4	60.91	61.98
10	**62.10**	**91.17**	**34.4**	**62.56**	<u>62.37</u>
20	61.75	<u>91.08</u>	<u>32.8</u>	<u>61.88</u>	62.04
40	<u>62.08</u>	90.77	31.0	61.28	<u>62.37</u>
80	61.15	90.65	31.0	60.93	62.13

Table 3: p-Mix better balances the performance of long-context and short-context than other mixing strategies.

Strategy	HELMET	RULER	Longbench v2	LongAVG	ShortAVG
No Mix	62.10	**91.17**	**34.4**	**62.56**	62.37
Sequential Mix	61.60	88.85	31.8	60.75	61.89
Simple Mix	61.84	89.65	31.2	60.90	64.04
p-Mix (Ours)	**62.11**	89.70	33.0	61.60	**64.10**

document QA setting). Table 2 provides the detailed performance scores for different values of n in the multi-document setting, corresponding to the trends shown in Appendix A.2. We observe that the multi-document strategy significantly improves performance on long-context tasks (from 60.19 to 62.56). As the value of n increases, the performance on long-context tasks improves and degrades, with the best performance observed when $n = 10$. We hypothesize that this trend is due to an excessive number of documents increasing the task difficulty beyond the model's learning capacity, thereby leading to a drop in performance.

4.2   Impact of Different Mixing Strategy

To investigate the effectiveness of the p-Mix Strategy, we compare p-Mix with three alternative mixing approaches: (1) No Mix: training solely on LongMagpie data without short-context SFT datasets; (2) Sequential Mix: first training on short-context data (UltraChat) then fine-tuning on long-context data (LongMagpie), similar to (Ding et al., 2023); (3) Simple Mix: directly combining and shuffling long and short data in a single training stage, similar to approaches used with LongAlign (Bai et al., 2024a); and (4) p-Mix (Ours): our proposed strategy from Algorithm 1 that pre-pends short instructions and probabilistically mixes segments. As Table 3 demonstrates, alternative strategies struggle to balance long-context and short-context performance compared to our p-Mix approach. In contrast, our p-Mix strategy demonstrates a superior balance: it achieves a competitive LongAVG of 61.60 (notably better than sequential and simple mixing, and only a slight trade-off compared to no mixing) while attaining the best ShortAVG score of 64.10. This highlights the efficacy of the p-Mix approach in maintaining strong long-context reasoning abilities while significantly bolstering performance on short, non-contextual tasks. More details can be found in Appendix A.3.

4.3   Impact of Different Data Size

To investigate the impact of data volume on model performance, we train our models using two different sizes of LongMagpie-generated data: 190k and 450k samples. As shown in Table 4, scaling up the training data from 190k to 450k samples leads to consistent improvements across all long-context evaluation benchmarks. Specifically, we observe gains of +0.81 on HELMET, +0.52 on RULER, and +1.8 on Longbench v2, resulting in a +1.05 improvement in the overall LongAVG metric. This demonstrates that increasing the volume of high-quality long-context instruction data significantly enhances the model's ability to comprehend and reason over extended contexts.

4.4   Impact of Different Source Model Size

To assess the impact of different models on data synthesis, we use LongMagpie to generate two 450k long-context instructions respectively by the Qwen-2.5-7B model and the Qwen-2.5-70B model. As shown in Table 5, using the larger 70B model improves LongAVG performance (59.61 $\rightarrow$ 62.56),

Table 4: Increasing the volume of training data improves performance on long-context benchmarks.

Source Model	Data Volume	HELMET	RULER	Longbench v2	LongAVG	ShortAVG
Qwen-2.5-70B	190k	61.29	90.65	32.6	61.51	62.30
Qwen-2.5-70B	450k	**62.10**	**91.17**	**34.4**	**62.56**	**62.37**

Table 5: Using the larger source model improves performance on long-context benchmarks..

Source Model	Data Volume	HELMET	RULER	Longbench v2	LongAVG	ShortAVG
Qwen-2.5-7B	450k	59.28	86.95	32.6	59.61	62.18
Qwen-2.5-70B	450k	**62.10**	**91.17**	**34.4**	**62.56**	**62.37**

(a) Reward model scores for different datasets.

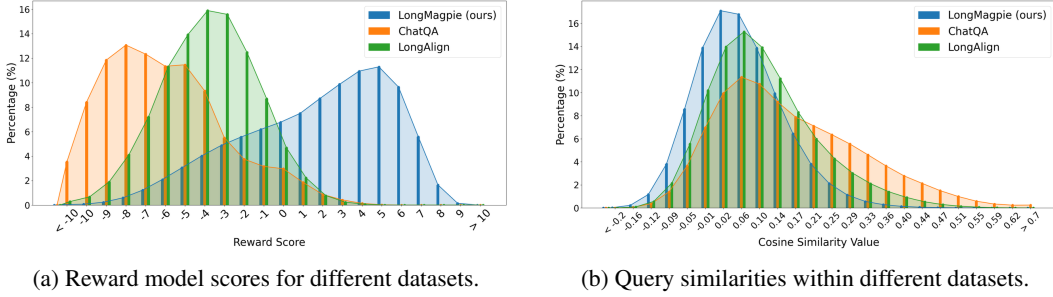(b) Query similarities within different datasets.

Figure 2: Analysis of LongMagpie-generated data quality and diversity. (a) higher reward model scores indicates higher quality. (b) lower pairwise query similarity indicates better diversity.
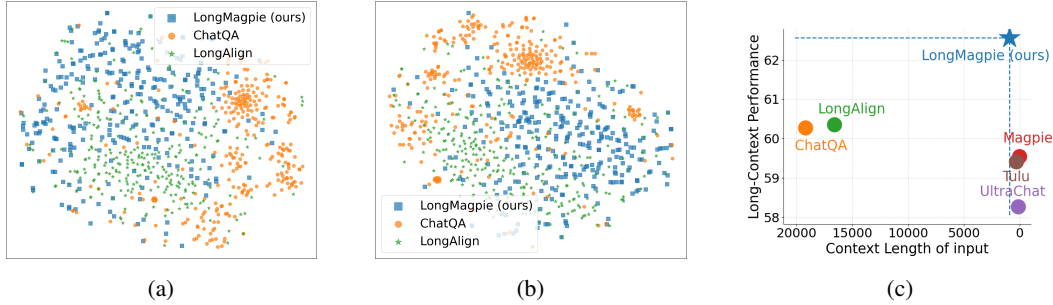

(a)

(b)

(c)

Figure 3: Visualizations of LongMagpie characteristics: (a,b) t-SNE visualizations of query embeddings from different datasets showing LongMagpie's dispersed distribution indicating diversity; (c) Long-context performance vs. token consumption showing LongMagpie's superior performance.

and shows similar performance on ShortAVG. This superior performance likely stems from larger models' enhanced ability to model long-context capabilities (Xiong et al., 2023), which translates to better results when applied to the LongMagpie method.

4.5 Analysis of of LongMagpie Queries

Higher Quality of LongMagpie Queries   We use the Reward Model FsfairX-Llama3-RM-v0.1 (Dong et al., 2024) to score three long-context fine-tuning datasets. As shown in Figure 2a, the x-axis represents the scores given by the reward model, and the y-axis represents the proportion of data within each dataset corresponding to that score. The overall data quality of LongMagpie is significantly higher than that of ChatQA and LongAlign.

Better Diversity of LongMagpie Queries   To investigate the diversity of different datasets, we sampled 300 queries from each dataset, inferred their embeddings using the jina-embeddings-v3 (Sturua et al., 2024) model, and visualized their distribution using t-SNE (Van der Maaten and Hinton,

2008), as shown in Figure 3. It can be observed that LongMagpie's distribution is more dispersed, reflecting its better diversity.

Furthermore, we repeated the following experiment 30 times: sampled queries from each dataset, calculated the pairwise similarity between the sampled queries within each dataset, and aggregated the distributions of all similarities, as shown in Figure 2b. It can be seen that LongMagpie queries generally exhibit lower similarity among themselves, which also reflects their good diversity.

4.6 Sample Efficiency of LongMagpie

We analyze the sample efficiency of various long-context instruction synthesis methods by quantifying the average token processing requirements during instruction synthesis. As illustrated in Figure 3c, LongMagpie exhibits exceptional sample efficiency, achieving superior long-context performance while processing substantially fewer tokens per instruction (averaging 1.6K tokens)[1]. This efficiency stands in stark contrast to methods like ChatQA and LongAlign, which consume 10-13× more tokens per instruction during synthesis yet produce inferior performance outcomes. LongMagpie's remarkable sample efficiency facilitates greater scalability and diversity.

5 Related Work

5.1 Long-Context Data Synthesis

Existing approaches to synthesizing long-context data can be divided into two categories.

Continuation-Oriented Methods   Approaches in this category generate long-context data by concatenating shorter documents. Early methods (Roziere et al., 2023; Chen et al., 2023b) used random sampling and concatenation, but failed to maintain meaningful long-range dependencies. Later approaches preserved semantic coherence through document clustering (Guu et al., 2020) or nearest-neighbor retrieval (Shi et al., 2023). Quest (Gao et al., 2024a) balances relevance and diversity using keyword matching. NExtLong (Gao et al., 2025) decomposes a document into multiple meta-chunks and extends the context by interleaving hard negative distractors retrieved from pretraining corpora. However, these methods focus on pre-training rather than instruction tuning. In contrast, LongMagpie directly generates instruction-following data with the model's auto-regressive capabilities.

Instruction-Oriented Methods   There exist many approaches to generate long-context instruction data (Zhu et al., 2025b; Wang et al., 2024; Köksal et al., 2024; Sun et al., 2025). Representative works include WildLong (Li et al., 2025) uses templates and seed questions, LongAlign (Bai et al., 2024a) employs Self-Instruct with packing strategies but requires curated examples, ChatQA (Liu et al., 2024b) blends QA datasets with conversational QA, ChatQA 2 (Xu et al., 2024b) packs documents into 32-128K token contexts, LOGO (Tang et al., 2024) adapts self-synthesis for long-context alignment, and GATEAU (Si et al., 2024) focuses on valuable instruction selection. These methods obtain high-quality data through complex pipelines. In contrast, LongMagpie eliminates seed questions, and complex pipelines by leveraging aligned LLMs' ability to generate contextually relevant queries when provided only with documents.

5.2 Synthesis Methods for Short-Context Instruction Data

Recent studies scale synthesis across various dimensions: Unnatural Instructions (Honovich et al., 2022) yields diverse instructions through paraphrasing; WizardLM (Xu et al., 2023a) uses evolutionary strategies for challenging variants; GLAN (Li et al., 2024a) eliminates templates by generating tasks from taxonomies; BARE (Zhu et al., 2025a) improves factual correctness; and Humpback (Li et al., 2024b) performs instruction back-translation. Domain-specific approaches like MetaMath (Yu et al., 2024) generate specialized content. Magpie (Xu et al., 2024c) demonstrates aligned LLMs can autoregressively generate diverse instructions without human annotation or seed examples. Motivated

---

[1]Our multi-document extension approach enables arbitrary context length extension without incurring additional computational overhead.

by Magpie, LongMagpie extends this paradigm to long-context settings by leveraging document-query relationship patterns from instruction tuning, enabling diverse long-context instruction data without specialized prompting.

6 Conclusion

This paper introduces LongMagpie, a self-synthesis method that automatically generates large-scale long-context instruction data without human annotation or seed examples. Extensive experiments on HELMET, RULER, and Longbench v2 demonstrate that models trained on LongMagpie data achieve leading performance on long-context tasks while maintaining competitive short-context capabilities when combined with our proposed p-Mix strategy. This work establishes LongMagpie as an effective approach for democratizing long-context capabilities.

7 Limitations

First, LongMagpie unavoidably inherits biases from the source instruction-tuned LLMs, which future work should detect and mitigate. Second, the current implementation of LongMagpie inadequately covers tasks requiring long-form outputs, as it primarily focuses on document-query relationships rather than extended reasoning or generation. Future research should expand support for diverse output formats and complex analytical tasks.

References

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

Mohammad Gheshlaghi Azar, Kevin Swersky, Daniel Tarlow, Satinder Sundararajan, Mohammad Hashemi, Danilo Jimenez Rezende, Stephan Gouws, Wenjie Zhao, Yunhao Li, Edward Gibson, et al. A general theoretical paradigm to understand learning from human preferences. *arXiv preprint arXiv:2310.12036*, 2023.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. Longbench: A bilingual, multitask benchmark for long context understanding. *arXiv preprint arXiv:2308.14508*, 2023.

Yushi Bai, Xin Lv, Jiajie Zhang, Yuze He, Ji Qi, Lei Hou, Jie Tang, Yuxiao Dong, and Juanzi Li. Longalign: A recipe for long context alignment of large language models. *arXiv preprint arXiv:2401.18058*, 2024a.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. *arXiv preprint arXiv:2412.15204*, 2024b.

Ramakrishna Bairi, Atharv Sonwane, Aditya Kanade, VageeshD C, Arun Iyer, Suresh Parthasarathy, Sriram Rajamani, B. Ashok, and Shashank Shet. Codeplan: Repository-level coding using llms and planning. Sep 2023.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.

Avi Caciularu, MatthewE. Peters, Jacob Goldberger, Ido Dagan, and Arman Cohan. Peek across: Improving multi-document modeling via cross-document question-answering. May 2023.

Chi Han Chen, Sang Michael Liu, Tianyi Xiao, Zhilin Xu, Rameswar Panda, Xin Eric Xu, Yikang Ma, Kevin Xie Lin, Soheil Feizi, and Pradip Mittal. Extending context window of large language models via positional interpolation. In *Proceedings of the 40th International Conference on Machine Learning*, 2023a.

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. Longlora: Efficient fine-tuning of long-context large language models. *arXiv preprint arXiv:2309.12307*, 2023b.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Sheng Si, Yun Liu, Zhiyuan Zhang, Yu Wu, Chao Li, et al. Ultrachat: A large-scale auto-generated data for diverse conversations with large language models. *arXiv preprint arXiv:2305.14233*, 2023.

Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. Rlhf workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*, 2024.

Yury Fu, Peter Levin, Nikos Casas, Orhan Firat, and Rohan Anil. Data engineering for scaling language models to 128k context. *arXiv preprint arXiv:2402.10171*, 2024.

Chaochen Gao, Xing Wu, Qi Fu, and Songlin Hu. Quest: Query-centric data synthesis approach for long-context scaling of large language model. *arXiv preprint arXiv:2405.19846*, 2024a.

Chaochen Gao, Xing Wu, Zijia Lin, Debing Zhang, and Songlin Hu. Nextlong: Toward effective long-context training without long documents. *arXiv preprint arXiv:2501.12766*, 2025.

Tianyu Gao, Alexander Wettig, Howard Yen, and Danqi Chen. How to train long-context language models (effectively). 2024b. URL https://api.semanticscholar.org/CorpusID: 273098476.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2:1, 2023.

Google DeepMind. Gemini model updates: March 2025, March 2025. Accessed on May 8, 2025.

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

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Retrieval augmented language model pre-training. *arXiv preprint arXiv:2002.08909*, 2020.

Chi Han, Qifan Wang, Wenhan Wang, Yun Bai, Eric P Xing, Xiang Ren, and Fan Yang. Lm-infinite: Simple on-the-fly length generalization for large language models. *arXiv preprint arXiv:2308.16137*, 2023.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. Unnatural instructions: Tuning language models with (almost) no human labor. *arXiv preprint arXiv:2212.09689*, 2022.

Tianyu Hsieh, Gupta Otkrist, Jeff Wu, Devamanyu Lin, Yuntian Li, Yue Tian, Yann LeCun, and Wenhan Xiong. Ruler: Discrimination-aware long-context benchmarking. *arXiv preprint arXiv:2405.17781*, 2024.

Cheng Jin, Wenli Zhang, Aniruddha Li, Yang Zhang, Xiang Yuan, Wenjie Liu, Bin Zhao, Shiyu Chang, Yongfeng Zhang, and Yueting Zhuang. Llm self-extension: Teaching large language models to use tools by generating canonical api calls. *arXiv preprint arXiv:2403.08233*, 2024.

Abdullatif Köksal, Timo Schick, Anna Korhonen, and Hinrich Schütze. Longform: Optimizing instruction tuning for long text generation with corpus extraction. In *Findings of EMNLP 2024*, 2024.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. T\" ulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Haoran Li, Qingxiu Dong, Zhengyang Tang, Chaojun Wang, Xingxing Zhang, Haoyang Huang, Shaohan Huang, Zeqiang Huang, Dongdong Zhang, Yuxian Gu, Xin Cheng, Xun Wang, Si-Qing Chen, Li Dong, Wei Lu, Zhifang Sui, Benyou Wang, Wai Lam, and Furu Wei. Synthetic data (almost) from scratch: Generalized instruction tuning for language models. *arXiv preprint arXiv:2402.13064*, 2024a.

Jiaxi Li, Xingxing Zhang, Xun Wang, Xiaolong Huang, Li Dong, Liang Wang, Si-Qing Chen, Wei Lu, and Furu Wei. Wildlong: Synthesizing realistic long-context instruction data at scale. *arXiv preprint arXiv:2502.16684*, 2025.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Omer Levy, Luke Zettlemoyer, Jason Weston, and Mike Lewis. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*, 2023.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Omer Levy, Luke Zettlemoyer, Jason Weston, and Mike Lewis. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*, 2024b.

Wing Lian, Bleys Goodson, Eugene Pentland, Austin Cook, Chanvichet Vong, and "Teknium". Openorca: An open dataset of gpt augmented flan reasoning traces. `https://https://huggingface.co/datasets/Open-Orca/OpenOrca`, 2023.

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

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. *arXiv preprint arXiv:2007.08124*, 2020.

Zihan Liu, Wei Ping, Rajarshi Roy, Peng Xu, Chankyu Lee, Mohammad Shoeybi, and Bryan Catanzaro. Chatqa: Building gpt-4 level conversational qa models. *CoRR*, 2024b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Anton Lozhkov, Loubna Ben Allal, Leandro von Werra, and Thomas Wolf. Fineweb-edu, May 2024. URL `https://huggingface.co/datasets/HuggingFaceFW/fineweb-edu`.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The lambada dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031*, 2016.

Bowen Peng, Ye Zheng, Tie Xiao, Guangtao Liu, Siyuan Wang, Yutao Yang, and Aston Zhang. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*, 2023.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE, 2020.

Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Liu, Denis Lebowitz, Piero Molino Ferrer, Tom Cochrane, et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021.

Weijia Shi, Sewon Min, Maria Lomeli, Chunting Zhou, Margaret Li, Gergely Szilvasy, Rich James, Xi Victoria Lin, Noah A Smith, Luke Zettlemoyer, et al. In-context pretraining: Language modeling beyond document boundaries. *arXiv preprint arXiv:2310.10638*, 2023.

Shuzheng Si, Haozhe Zhao, Gang Chen, Yunshui Li, Kangyang Luo, Chuancheng Lv, Kaikai An, Fanchao Qi, Baobao Chang, and Maosong Sun. Gateau: Selecting influential sample for long context alignment. *arXiv preprint arXiv:2410.15633*, 2024.

Saba Sturua, Isabelle Mohr, Mohammad Kalim Akram, Michael Günther, Bo Wang, Markus Krimmel, Feng Wang, Georgios Mastrapas, Andreas Koukounas, Andreas Koukounas, Nan Wang, and Han Xiao. jina-embeddings-v3: Multilingual embeddings with task lora, 2024. URL `https://arxiv.org/abs/2409.10173`.

Rui Sun, Zhiwei Sun, Yang Li, Yi Ren, and Wei Bi. Efficient training of ultra-long context large language models. *arXiv preprint arXiv:2504.06214*, 2025.

Zecheng Tang, Zechen Sun, Juntao Li, Qiaoming Zhu, and Min Zhang. Logo–long context alignment via efficient preference optimization. *arXiv preprint arXiv:2410.18533*, 2024.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpaca: A strong, replicable instruction-following model. *arXiv preprint arXiv:2303.16199*, 2023.

Llama team at Meta. Llama 3: Third-generation open foundation language models. *arXiv preprint arXiv:2405.08456*, 2024.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Liang Wang, Nan Yang, Xingxing Zhang, Xiaolong Huang, and Furu Wei. Bootstrap your own context length. *arXiv preprint arXiv:2412.18860*, 2024.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*, 2022a.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*, 2022b. URL `https://arxiv.org/abs/2212.10560`.

Yunshu Wu, Hayate Iso, Pouya Pezeshkpour, Nikita Bhutani, and Estevam Hruschka. Less is more for long document summary evaluation by llms. *arXiv preprint arXiv:2309.07382*, 2023.

Guangxuan Xiao, Han Tian, Ji Lin, Xiang Zhang, Hanling Cheng, Kai Huang, Chuang Gan, Luowei Zhou, Jordan Boyd-Graber, and Ser-Nam Wang. Efficient streaming language models with attention sinks. In *Proceedings of the 41st International Conference on Machine Learning*, 2023.

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

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*, 2023a.

Peng Xu, Wei Ping, Xianchao Wu, Chejian Xu, Zihan Liu, Mohammad Shoeybi, and Bryan Catanzaro. Chatqa 2: Bridging the gap to proprietary llms in long context and rag capabilities. *arXiv preprint arXiv:2407.14482*, 2024a.

Peng Xu, Wei Ping, Xianchao Wu, Chejian Xu, Zihan Liu, Mohammad Shoeybi, and Bryan Catanzaro. Chatqa 2: Bridging the gap to proprietary llms in long context and rag capabilities. *arXiv preprint arXiv:2407.14482*, 2024b.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing. In *International Conference on Learning Representations*, 2023b.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing. *arXiv preprint arXiv:2406.08464*, 2024c.

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

Tse-Yu Yen, Tsu-Jui Cheng, Paul Pu Liang, Xiang Dong, Tianyu Zhao, Wenhan Liu, Wenhao Wang, Min Peng, Oleksiy Shliazhko, Li Zhang, et al. Helmet: A hierarchical efficient benchmark for long context evaluation with modular units and comprehensive taxonomy. *arXiv preprint arXiv:2405.18696*, 2024.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. In *ICLR 2024*, 2024.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

Alan Zhu, Parth Asawa, Jared Q. Davis, Lingjiao Chen, Boris Hanin, Ion Stoica, Joseph E. Gonzalez, and Matei Zaharia. Bare: Combining base and instruction-tuned language models for better synthetic data generation. *arXiv preprint arXiv:2502.01697*, 2025a.

Dawei Zhu, Zhihong Xia, Fan Ye, Yuchao Qin, Haoran Zhou, Haoyu Chen, Rongpeng Ren, Yankai Li, Zheng Zhang, Rui Fu, et al. Pose: Efficient context window extension of llms via positional skip-wise training. *arXiv preprint arXiv:2309.10400*, 2023.

Wenhao Zhu, Pinzhen Chen, Hanxu Hu, Shujian Huang, Fei Yuan, Jiajun Chen, and Alexandra Birch. Effective data synthesis for long-context instruction tuning. *arXiv preprint arXiv:2502.15592*, 2025b.

A Detailed Experimental Results

A.1 Training Config

We employ the AdamW (Loshchilov and Hutter, 2017) optimizer with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.95$. Following ProLong (Gao et al., 2024b), we concatenate samples up to 64K sequence length and apply the document masking technique to prevent interactions between independent sequences. Additionally, we utilize FlashAttention-2 (Dao, 2023) and ZeRO (Rajbhandari et al., 2020) to optimize memory usage and accelerate training. The detailed training config is shown in Table 6.

Table 6: Model Training Configuration.

	training setting
Initial Model	Llama-3-8B-NExtLong-512K-Base
rotary-emb-base	128,000,000
β_1	0.9
β_2	0.95
lr	$2e^{-5}$
precision	bfloat16
gradient-clipping	1.0
weight-decay	0.1
lr-decay-style	cosine
train-iters	250
seq-length	65536
GPU-type	H100
GPU-numbers	8
training-time	10h

A.2 Impact of Multi-Document Setting

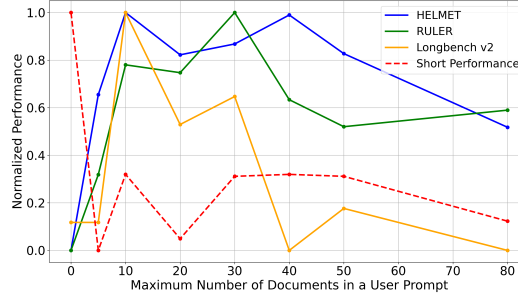Figure 4 illustrates the performance variation under different multi-document configurations.

Figure 4: Impact of the multi-document setting on model performance. As the number of documents increases, the performance on long-context tasks improves and then decreases.

A.3 Ablation Study on *p*-Mix Strategy Parameters

To further understand the behavior of the *p*-Mix strategy, we conducted an ablation study on its key parameters: the number of initial short-context samples pre-pended (N_S), and the probability (P_L) of selecting a long-context sample during the probabilistic mixing phase (see Algorithm 1). The results, presented in Table 7, showcase how different configurations impact overall performance on both long and short tasks evaluation benchmarks. These experiments were conducted with $n = 10$ for the multi-document context length parameter.

Table 7: Detailed ablation results for different parameter settings of the p-Mix strategy. N_S is the number of pre-pended short tasks. P_L is the long-context selection probability.

N_S	P_L	HELMET	RULER	Longbench	LongAVG	ShortAVG
0	0.2	61.38	88.52	29.60	59.83	64.17
	0.4	61.84	89.65	31.20	60.90	64.04
	0.6	61.64	90.51	31.00	61.05	63.92
	0.8	61.48	90.54	30.40	60.81	63.41
1	0.2	61.62	88.05	31.60	60.42	**64.39**
	0.4	62.11	89.70	33.00	**61.60**	**64.10**
	0.6	61.74	90.58	29.80	60.71	63.71
	0.8	61.45	90.66	28.80	60.30	63.33
5	0.2	61.41	88.12	29.80	59.78	64.16
	0.4	61.70	88.67	31.20	60.52	64.13
	0.6	61.90	90.07	30.00	60.66	63.97
	0.8	61.34	90.53	31.00	60.96	63.68
30	0.2	61.17	85.67	31.80	59.55	**64.41**
	0.4	60.77	85.30	30.00	58.69	64.25
	0.6	60.67	86.09	30.80	59.19	**64.39**
	0.8	60.60	84.42	30.00	58.34	64.21

Algorithm 1 Hybrid SFT Data Construction with short-context Pre-pending and Probabilistic Mixing

---

1: **procedure** CONSTRUCTHYBRIDSAMPLE($D_S, D_L, P_L, L_{max}, sep$)
2:   Initialize $S_{concat} \leftarrow$ empty sequence   ▷ D_S: set of short-context SFT samples, D_L: set of long-context SFT samples   ▷ P_L: probability of selecting a long-context sample, L_{max}: max sequence length   ▷ sep: separator token/sequence between samples
3:   $s_0 \leftarrow$ RandomSample(D_S)
4:   $S_{concat} \leftarrow$ FormatSample(s_0)
5:   $current\_length \leftarrow$ Length(S_{concat})
6:   **while** $current\_length < L_{max}$ **do**
7:     $rand \leftarrow$ RandomReal($0, 1$)
8:     **if** $rand < P_L$ **then**   ▷ Select long-context sample with probability P_L
9:       $l_{next} \leftarrow$ RandomSample(D_L)
10:       $formatted\_l_{next} \leftarrow$ FormatSample(l_{next})
11:       **if** $current\_length +$ Length(sep) $+$ Length($formatted\_l_{next}$) $\leq L_{max}$ **then**
12:         $S_{concat} \leftarrow S_{concat} \oplus sep \oplus formatted\_l_{next}$
13:         $current\_length \leftarrow$ Length(S_{concat})
14:       **else**
15:         **break**   ▷ Next sample exceeds L_{max}
16:       **end if**
17:     **else**   ▷ Select short-context sample with probability $1 - P_L$
18:       $s_{next} \leftarrow$ RandomSample(D_S)
19:       $formatted\_s_{next} \leftarrow$ FormatSample(s_{next})
20:       **if** $current\_length +$ Length(sep) $+$ Length($formatted\_s_{next}$) $\leq L_{max}$ **then**
21:         $S_{concat} \leftarrow S_{concat} \oplus sep \oplus formatted\_s_{next}$
22:         $current\_length \leftarrow$ Length(S_{concat})
23:       **else**
24:         **break**   ▷ Next sample exceeds L_{max}
25:       **end if**
26:     **end if**
27:   **end while**
28:   **return** S_{concat}
29: **end procedure**