LiteLong: Resource-Efficient Long-Context Data Synthesis for LLMs

Junlong Jia[1,5,7], Xing Wu[2,3†], Chaochen Gao[2], Ziyang Chen[2], Zijia Lin[4], Zhongzhi Li[3], Weinong Wang[3], Haotian Xu[3], Donghui Jin[1,7], Debing Zhang[3], Binghui Guo[1,5,6,7†]

[1]School of Artificial Intelligence, Beihang University
[2]Institute of Information Engineering, Chinese Academy of Sciences
[3]Xiaohongshu Inc
[4]Tsinghua University
[5]Zhongguancun Laboratory, Beijing
[6]Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing, Beijing
[7]LMIB, NLSDE, Beihang University, Beijing

Abstract

High-quality long-context data is essential for training large language models (LLMs) capable of processing extensive documents, yet existing synthesis approaches using relevance-based aggregation face challenges of computational efficiency. We present LiteLong, a resource-efficient method for synthesizing long-context data through structured topic organization and multi-agent debate. Our approach leverages the BISAC book classification system to provide a comprehensive hierarchical topic organization, and then employs a debate mechanism with multiple LLMs to generate diverse, high-quality topics within this structure. For each topic, we use lightweight BM25 retrieval to obtain relevant documents and concatenate them into 128K-token training samples. Experiments on HELMET and Ruler benchmarks demonstrate that LiteLong achieves competitive long-context performance and can seamlessly integrate with other long-dependency enhancement methods. LiteLong makes high-quality long-context data synthesis more accessible by reducing both computational and data engineering costs, facilitating further research in long-context language training.

Code/Appendix —
https://github.com/jiajunlong-buaa/LiteLong

Introduction

Large language models (LLMs) have gained widespread attention due to their robust capabilities. Recently, LLM context windows have expanded significantly (Peng et al. 2024; Young et al. 2024; Team 2024; rednote hilab 2025). The Llama series illustrates this trend, evolving from 4K tokens in Llama 2 (Touvron et al. 2023) to 128K in Llama 3.1 (Dubey et al. 2024). This expanded capacity enables LLMs to address complex tasks like document summarization (Wu et al. 2023b), question answering on books (De Cao, Aziz, and Titov 2018), and Code Planning (Bairi et al. 2024). Current approaches for modeling long-range dependencies typically continue training LLMs with documents reaching the target length. However, high-quality long documents remain scarce across most domains, creating a significant challenge as target context lengths increase (Gao et al. 2025a).

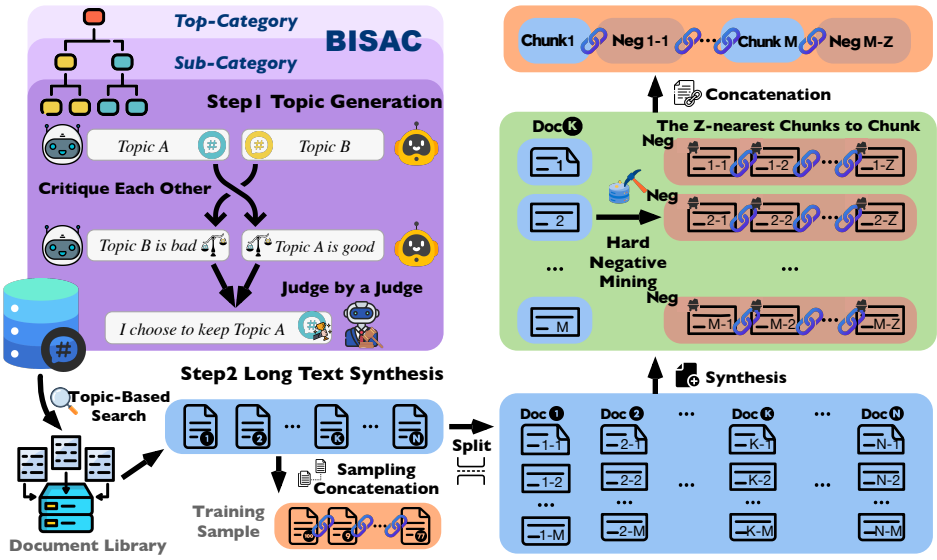
Existing methods for long-context data synthesis are mainly relevance-based (Guu et al. 2020; Levine et al. 2022; Shi et al. 2024; Gao et al. 2025a), which aggregate semantically relevant documents to form long-range dependencies. For example, ICLM (Shi et al. 2024) constructs a semantic document graph via retrieval and indexing, and proposes a traveling salesman problem-based sorting algorithm to maximize contextual similarity while ensuring that each document is included only once. Similarly, Quest (Gao et al. 2025a) uses a generative model to predict potential queries for each document, then groups documents with similar query patterns. While these methods improve document relevance, they face two main limitations: (1) they require either generating embeddings or constructing queries over massive corpora, both of which demand considerable computational resources—the GPU cost incurred during the data synthesis process; and (2) they lack a clear framework to ensure diversity coverage in the generated content. This raises the question of *whether we can achieve effective long-context data synthesis while maintaining resource efficiency*.

We draw inspiration from Cosmopedia V2 (Ben Allal et al. 2024), which adopts the Book Industry Standards and Communications (BISAC) (Martínez-Ávila 2016) book classification—a comprehensive subject categorization system that offers better coverage and diversity than categories constructed through unsupervised clustering. The success of adopting BISAC provides a promising direction for creating large-scale long-context data without incurring high clustering computational costs as previous works (Shi et al. 2024; Gao et al. 2025a).

We start from the BISAC classification standard and use LLMs to generate diverse topics for each second-level category. We then employ lightweight BM25 (Robertson, Zaragoza et al. 2009) retrieval to obtain topic-relevant documents and concatenate them into long-context training data of the target length. This process is highly efficient, thus termed **LiteLong**, because the BISAC book classification system contains only a few thousand categories, minimizing GPU computation required for topic generation.

We further implement a multi-agent debate (Du et al.

[0†]Corresponding author.



(A) LiteLong-Base (B) LiteLong+NextLong

Figure 1: Overview of the LiteLong framework. The left subfigure illustrates the LiteLong method, focusing on topic synthesis using BISAC categories. The right subfigure shows the integration of LiteLong with the long-dependency enhancement method, NExtLong(Gao et al. 2025b).

2023; Kenton et al. 2024) mechanism to promote the diversity and quality of generated topics. This mechanism employs two separate Debate LLMs to independently generate topic candidates, then has them mutually critique each other's topics based on multiple criteria, such as topic relevance and diversity. Finally, the Judge LLM aggregates the critiques and adjudicates the best subset of topics for each secondary category. This mechanism has two main advantages: (1) the competitive generation process involves two independent LLMs, which can produce more diverse topics and avoid the limitations of a single LLM due to its training data; (2) the Judge LLM can reject weakly relevant topics, thereby improving the topic quality.

The incorporation of the multi-agent debate mechanism does not significantly increase LiteLong's resource demands. Specifically, the debate is conducted at the topic level, rather than across full documents. Since the number of BISAC subcategories is only a few thousand, and each category only generates a limited number of topic candidates, the total number of LLM inference is relatively small. This contrasts sharply with relevance-based approaches that require expensive document embeddings or dense clustering over billions of tokens.

Through extensive experiments and evaluations on the HELMET (Yen et al. 2025) and RULER (Hsieh et al. 2024) benchmarks, we demonstrate that LiteLong achieves competitive performance in a resource-efficient setting. Furthermore, we find that LiteLong can seamlessly integrate with recently proposed long-dependency enhancement method NExtLong (Gao et al. 2025b), further improving the model's long-context capabilities. We also provide ablation studies investigating the key components of LiteLong. Overall, LiteLong provides a resource-efficient approach to long-context data synthesis, substantially reducing resource consumption during the data construction process, facilitating further research in long-context language modeling and contributing to the democratization of these capabilities.

Our contributions are as follows:

- We present LiteLong, a resource-efficient method for synthesizing long-context data through structured topic organization and multi-agent debate mechanism.
- We demonstrate that LiteLong achieves competitive long-context performance and can be seamlessly integrated with recently proposed long-dependency enhancement method.
- We provide an in-depth analysis of the design choices in LiteLong to validate the effectiveness of each component.

Related Work

Long-Context Data Synthesis

The development of effective long-context language models is hindered by the scarcity of high-quality long-document training data. Several approaches are proposed to address this challenge. The simplest approach, known as Random Concatenation (Standard) (Ouyang et al. 2022), involves randomly joining shorter documents to create longer training samples (Xiong et al. 2023). While computationally efficient, this method often produces incoherent transitions between documents, limiting the model's ability to develop robust long-range understanding. Similarity-based methods, such as KNN (Jiang et al. 2023), aggregate documents based on content similarity, improving semantic coherence but

potentially limiting exposure to diverse content patterns. Query-centric approaches, like Quest (Gao et al. 2025a), use a generative model to predict potential queries for each document, then group documents with similar query patterns. While effective, this approach introduces significant computational overhead for query generation. Document transformation methods, such as Untie the Knots (UtK) (Tian et al. 2024) and NExtLong (Gao et al. 2025b), manipulate document structure by chunking, shuffling, and incorporating negative examples to encourage the model to develop better document navigation capabilities, focusing primarily on improving attention mechanisms rather than semantic organization. LiteLong differentiates itself by leveraging an external topic framework to guide document organization, avoiding both the incoherence of random concatenation and the computational expense of query generation.

Multi-Agent Collaboration in Content Generation

Recent research explores the potential of multi-agent systems for content generation and evaluation, highlighting several key approaches. Debate-based methods, such as those used in systems like Generative Agents (Park et al. 2023) and Debating AI (Irving, Christiano, and Amodei 2018), utilize structured debates between multiple AI agents to enhance reasoning and content quality, showing particular promise in domains requiring complex evaluation criteria. Role-based collaboration methods, which employ specialized agent roles such as debater, critic, and editor, demonstrate improvements in content quality and diversity (Du et al. 2023; Wu et al. 2023a). Additionally, LLM agents are applied to synthetic data generation tasks, including instruction tuning data (Xu et al. 2024) and reasoning examples (Wang et al. 2022). Our approach extends these ideas to the domain of long-context data synthesis, employing a novel debate mechanism between competing LLMs to ensure both diversity and quality in generated topics.

Method

In this section, we describe LiteLong, our approach to efficient long-context data synthesis through multi-agent debate. We first introduce the BISAC (Martínez-Ávila 2016) category system, and then detail the two main components of our method: (1) multi-agent debate topic generation, and (2) document retrieval and aggregation. Finally, we describe how LiteLong can be seamlessly integrated with NExtLong to further enhance long-range dependency modeling.

Preliminary: BISAC Categories

The BISAC classification system is a comprehensive, hierarchical structure consisting of 51 primary categories and approximately 4,500 subcategories, covering nearly all domains of human knowledge. Widely used in the book industry, this system categorizes books by subject, facilitating easier discovery and organization. The BISAC system offers several advantages: (1) **Comprehensive Coverage**: It spans a wide range of subjects, ensuring representation of nearly all areas of human knowledge, making it a versatile tool for organizing diverse content. (2) **Hierarchical Structure**: The system is organized hierarchically, allowing for both broad and specific categorization, which aids in efficiently navigating from general topics to more specific subtopics. (3) **Expert-Developed and Regularly Updated**: Developed by industry experts, the categories are regularly updated to reflect new trends and knowledge areas, ensuring their relevance and accuracy. These advantages make BISAC our chosen basis for organizing long-context data synthesis.

Algorithm 1: Efficient Long-Context Data Synthesis via Multi-Agent Debate

Notation:
  $\mathcal{T}_{\text{total}}$: All candidate topics generated by both Debate models
  $\mathcal{T}_{\text{reject}}$: Low-quality topics filtered out by the Judge
  $\mathcal{T}_{\text{final}}$: Final set of high-quality topics used for document retrieval
Initialize $\mathcal{T}_{\text{total}}, \mathcal{T}_{\text{reject}}, \mathcal{T}_{\text{final}} \leftarrow \emptyset$
for each BISAC subcategory $\mathcal{S}$ **do**
  $\mathcal{T}_1 \leftarrow \text{Debate}_1$ generates topics
  $\mathcal{T}_2 \leftarrow \text{Debate}_2$ generates topics
  Each Debate model critiques the other's outputs
  $\mathcal{T}_{\mathcal{S}} \leftarrow$ Judge filters low-quality topics
  $\mathcal{T}_{\text{total}} \leftarrow \mathcal{T}_{\text{total}} \cup \mathcal{T}_1 \cup \mathcal{T}_2$
  $\mathcal{T}_{\text{reject}} \leftarrow \mathcal{T}_{\text{reject}} \cup \mathcal{T}_{\mathcal{S}}$
end for
$\mathcal{T}_{\text{final}} \leftarrow \mathcal{T}_{\text{total}} \setminus \mathcal{T}_{\text{reject}}$
for all $t \in \mathcal{T}_{\text{final}}$ **do**
  $\mathcal{D}_t \leftarrow$ Top 256 retrieved docs via BM25
  $\mathcal{S}_{\text{final}} \leftarrow \mathcal{S}_{\text{final}} \cup$ Aggregate($\mathcal{D}_t$)
end for

Multi-Agent Debate Topic Generation

To ensure that the generated topics are both diverse and high-quality, we employ a novel multi-agent debate mechanism that combines generative diversity with discriminative filtering. This approach leverages the complementary strengths of multiple language models: two specialized Debate LLMs independently generate candidate topics, while a Judge LLM identifies and filters out those that are redundant or low-quality. The Judge LLM is typically chosen to be slightly weaker than the Debate LLMs—not only because detecting flaws is generally easier than generating new content (Kenton et al. 2024), but also to reduce inference costs during the judging phase. This design effectively balances adequacy and efficiency.

For each BISAC subcategory, the two Debate LLMs independently generate candidate topics—each accompanied by a brief explanation—based on their interpretation of the subcategory. They then critique each other's outputs using multiple criteria, including relevance, semantic diversity, complementarity, and overall topical quality, and provide persuasive justifications for their evaluations. The Judge LLM reviews all topics and critiques, flags low-quality or overlapping entries, and constructs the final topic set by removing the rejected topics from the combined outputs of the two Debate models. This process fosters greater topical diversity

and conceptual richness than single-model generation, while maintaining content quality through lightweight evaluation.

Document Retrieval and Aggregation

Once the final collection of topics T_{final} is generated, we use each topic to retrieve and aggregate relevant documents into long-context training samples. For each topic in our collection, we use a lightweight BM25 retrieval method, via Manticore Search[1], to retrieve the top 256 relevant documents from the pretraining corpus. These documents are then aggregated with some strategies to create a sample reaching the target length. The aggregation strategies can be simply random shuffling, or recently proposed long-dependency enhancement methods (Tian et al. 2024; Gao et al. 2025b). The above process is formulated in Algorithm 1.

Combining with NExtLong

To further enhance the long-context capabilities of LiteLong, we integrate it with the recently proposed NExtLong method (Gao et al. 2025b). Initially, we randomly select a document from the retrieved set and apply NExtLong's chunking strategy to this document. The document is divided into multiple meta-chunks, each undergoing hard negative mining to identify challenging negative samples. These negatives are then concatenated with the meta-chunks to form an extended document. In the subsequent step, the model is trained on this synthesized long document, focusing on modeling long-range dependencies by distinguishing the meta-chunks across a wide range of hard negatives. This integration allows LiteLong to leverage NExtLong's strengths in modeling long-range dependencies, further improving the model's ability to handle extensive documents.

Moreover, the combination of LiteLong and NExtLong enhances resource efficiency. While NExtLong typically operates directly on the pretraining corpus, such as FineWeb-Edu (Lozhkov et al. 2024) and Cosmopedia V2 (Ben Allal et al. 2024), requiring considerable resources to build large vector retrieval databases, LiteLong + NExtLong only necessitates building a vector retrieval database for the retrieved samples. This is less than one-fifth the size of the original corpus, greatly reducing resource demands.

Experiments

In this section, we first describe our experimental setup, then present our main results and comparisons with existing methods. Finally, we compare the resource consumption of LiteLong with other methods.

Experimental Setup

Datasets We utilized two primary datasets for our experiments: FineWeb-Edu (Lozhkov et al. 2024) and Cosmopedia V2 (Ben Allal et al. 2024). **FineWeb-Edu** is a curated subset of web content that focuses on educational resources, scholarly articles, and instructional content. **Cosmopedia V2** serves as a comprehensive knowledge corpus that spans multiple domains, with a strong emphasis on factual accuracy and depth.

[1]https://manticoresearch.com/

Multi-agent LLMs For the Debate LLMs, we use Qwen2.5-7B (Yang et al. 2024) and Mixtral-8x7B-v0.1 (Jiang et al. 2024) as the default models, and for the Judge LLM, we use a smaller model, Gemma3-1B (Team et al. 2025).This configuration reflects our efficiency-oriented design: each model in the system is selected to be as lightweight as possible while still sufficiently capable for its respective role.

Baseline Methods We compare LiteLong with the following baseline methods: *Standard* (Ouyang et al. 2022) (random concatenation of documents to reach 128K context length), *KNN* (Guu et al. 2020; Levine et al. 2022)(implementation of the K-Nearest Neighbors approach from Quest (Gao et al. 2025a)), *ICLM* (Shi et al. 2024) (implementation of the In-Context Learning approach from ICLM (Shi et al. 2024)), and *Quest* (implementation of the query-centric approach from Quest (Gao et al. 2025a)). We also combine LiteLong with a recently proposed long-dependency enhancement method, *NExtLong*, with the implementation from NExtLong (Gao et al. 2025b). Baseline method details are shown in Appendix G. All methods synthesize approximately 4 billion tokens of training data for fair comparison.

Training Hyperparameters We fine-tune LLaMA-3-8B as our base model using each of the datasets described above. Training is performed with a learning rate of 4e-5 (using cosine decay), a batch size of 32 samples (128K tokens each), and 1,000 training steps. We apply a weight decay of 0.1 and use bfloat16 precision. For combined methods such as LiteLong+NExtLong, we first apply the LiteLong process to generate topically coherent document sets, followed by the respective transformation (e.g., chunking in NExtLong).

Evaluation Benchmarks We conduct evaluations on the HELMET (Yen et al. 2025) and RULER (Hsieh et al. 2024) benchmarks across five context lengths (8K, 16K, 32K, 64K, and 128K), which are specifically designed to assess long-context understanding across multiple dimensions such as recall, retrieval, reasoning, and comprehension.

Main Results

Table 1 presents the results on the HELMET benchmark (Yen et al. 2025), RULER benchmark (Hsieh et al. 2024) and the resource consumption comparison for different methods.

Experimental Results Analysis LiteLong achieves the highest average score among all baseline methods, reaching 61.90 across long-context metrics. This represents a substantial improvement of 6.65 points over Quest (55.25), the second best baseline. LiteLong demonstrates particularly strong performance on the Recall task (83.23) and the RULER task (83.88), highlighting the effectiveness of our topic-based organization in improving the model's ability to retain and utilize relevant information from extended contexts.

When combined with long-dependency enhancement methods, LiteLong+NExtLong achieves performance exceeding NExtLong alone (63.04 vs. 62.58 average score), with LiteLong+NExtLong showing particular strengths in Rerank (33.68) and RULER (83.73) metrics. These results

Method	Long-Context Metrics							GPU Hours	
	Recall	RAG	ICL	Rerank	LongQA	RULER	AVG	Embed (H)	Gen (H)
Comparison with Baseline Methods									
Standard	62.33	58.67	71.24	19.18	28.99	76.68	52.85	0	0
KNN	64.24	56.00	60.28	18.77	32.27	74.30	50.97	617	0
ICLM	64.04	54.48	72.36	14.04	28.17	69.14	50.37	617	0
Quest	69.13	57.47	72.08	22.35	**33.82**	76.63	55.25	0	806
LiteLong	**83.23**	**60.43**	**80.12**	**30.73**	33.01	**83.88**	**61.90**	0	6
Combined with Long-dependency Enhancement Method									
NExtLong	82.56	**60.91**	**81.76**	31.47	**37.30**	81.50	62.58	928	0
LiteLong+NExtLong	**82.93**	60.81	80.12	**33.68**	36.97	**83.73**	**63.04**	170	6

Table 1: Results on the HELMET and RULER benchmarks, along with GPU resource consumption. The evaluation results are averaged across context lengths of 8K, 16K, 32K, 64K, and 128K. GPU hours are reported separately for document embedding and topic generation.

suggest that LiteLong's topic-based organization complements techniques focused on attention patterns or negative examples by addressing different aspects of long-context modeling.

Resource Consumption Comparison LiteLong demonstrates exceptional efficiency advantages over competing methods. While Quest demands considerable computational resources (806 GPU hours for generation), LiteLong delivers superior performance with significantly reduced requirements (only 6 GPU hours for generation). This efficiency advantage is further demonstrated in the combined setting: while NExtLong alone requires 928 GPU hours for embedding, integrating LiteLong reduces the overall cost to just 176 GPU hours (170 hours for embedding and 6 hours for generation). This improvement stems from the fact that NExtLong typically constructs retrieval indices over the entire corpus (e.g., FineWeb-Edu and Cosmopedia V2), whereas LiteLong+NExtLong only requires indexing the much smaller set of documents already retrieved and organized by LiteLong—less than one-fifth the size of the original corpus—thereby substantially reducing embedding and indexing costs. With a substantial reduction in total resource consumption, LiteLong offers a practical and efficient solution for long-context modeling, achieving strong performance without the heavy computational overhead of existing methods.

Ablation Studies

In this section, we conduct in-depth ablation studies to analyze the impact of different components and design choices in our approach. Unless otherwise specified, all experiments are conducted with a 128k context length for training, using a mixed dataset composed of FineWeb-Edu (Lozhkov et al. 2024) and Cosmopedia V2 (Ben Allal et al. 2024).

Effectiveness of BISAC Categories

To explore the effectiveness of BISAC categories, we utilize the leading model GPT-4o to synthesize a category system and compare it with the BISAC standard. As shown in Table 2, data synthesized using the BISAC system achieves the highest average scores on the HELMET (Yen et al. 2025) and RULER (Hsieh et al. 2024) benchmarks, reaching an average score of 61.90—surpassing the category system automatically generated by large models. These results suggest that leveraging authoritative, structured external knowledge systems (such as BISAC) can lead to more comprehensive and diverse topic coverage in long-context data synthesis, thereby enhancing the long-text understanding capabilities of downstream models. We believe that as LLMs continue to evolve, automatic classification systems will become increasingly accurate and comprehensive, potentially surpassing BISAC in the future.

Approach	Average
GPT-4o Categories	59.94
BISAC Categories	**61.90**

Table 2: BISAC categories achieve better performance than automatically generated categories, evaluated on the HELMET and RULER benchmarks.

Method	Average
w/o Multi-agent Debate	56.33
w/ Multi-agent Debate	**58.59**

Table 3: Multi-agent debate improves performance. The debate mechanism achieves an average improvement of 2.26 points, evaluated on the HELMET and RULER benchmarks at 128K context length.

Effectiveness of Multi-agent Debate Mechanism

We investigate the impact of our multi-agent debate mechanism by comparing topic generation with and without the debate process at 128K context length. As shown in Table 3, incorporating the debate mechanism yields improvements across task categories. The model with debate achieves an average score of 58.59, compared to 56.33 without debate, representing a gain of 2.26 points. These results confirm

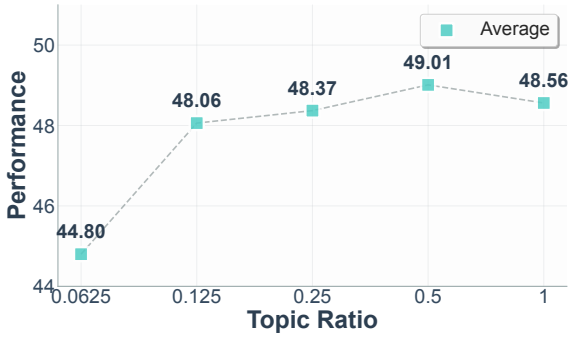


Figure 2: Performance across topic count configurations on the HELMET and RULER benchmarks. The 0.5× configuration (18,720 topics) achieves the highest average score (49.01), with performance degrading as topic count decreases further.

Data Source	Average
Mix Data	**61.90**
Cosmopedia V2 Only	52.32
FineWeb-Edu Only	61.29

Table 4: Impact of data sources on HELMET performance and RULER performance.

that the competitive generation process between two independent LLMs produces more diverse topics and the Judge LLM effectively filters out weakly relevant topics, thereby enhancing overall data quality and downstream model performance.

Impact of Topic Scaling

To understand how the number of topics affects performance, we vary the number of topics in LiteLong while keeping the total token count constant at 2B. As shown in Figure 2, model performance first improves with an increasing number of topics, but declines once the topic count becomes too high. This trend suggests that it is not merely the number of topics, but the diversity and quality of the topic set that play a crucial role in long-context understanding. Excessive topic granularity may lead to redundancy or reduced per-topic depth, ultimately hindering performance.

Impact of Data Source Selection

We evaluate the impact of using different data sources for document retrieval. As shown in Table 4, the mixed dataset (FineWeb-Edu + Cosmopedia V2) achieves the highest average performance (61.90) across the HELMET and RULER benchmarks, outperforming both individual sources. FineWeb-Edu alone yields strong results (61.29), indicating that its diverse writing styles and rich educational content are well-suited for long-context understanding. In contrast, Cosmopedia V2 alone results in a substantially lower score (52.32), suggesting that while curated content may offer high quality, it lacks the variability needed to generalize across complex, long-context tasks. These findings highlight that combining diverse data sources enhances topic coverage and improves downstream performance by providing both breadth and depth in training data.

Retention Strategy	Average Score
Filter-Reject	**61.90**
Keep-Accept	61.34
Keep-Fixed-K-Accept	61.54

Table 5: Comparison of topic retention strategies, all based on decisions from the Judge model.

Impact of Topic Retention Strategies in Multi-agent Debate

We investigate how different topic retention strategies, guided by the Judge model, influence the final performance. All strategies begin with the union of candidate topics generated by the Debate models:

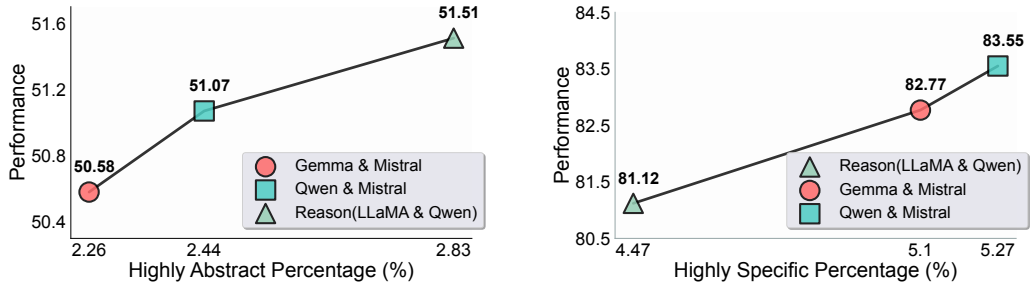
- **Filter-Reject**: Remove only those topics explicitly marked as low-quality by the Judge. (Default strategy)
- **Keep-Accept**: Retain all topics the Judge deems acceptable, without imposing a count limit.
- **Keep-Fixed-K-Accept**: Allow the Debate model to directly select a fixed number (e.g., $K = 10$) of high-quality topics.

As shown in Table 5, the most effective strategy is **Filter-Reject**. This approach consistently outperforms strategies where the Judge model either adaptively selects or selects a fixed number of high-quality topics. Specifically, the **Keep-Fixed-K-Accept** strategy yields slightly lower performance, while the **Keep-Accept** strategy performs the worst—even falling below the baseline without multi-agent debate. These findings suggest that the Judge model is more effective in identifying low-quality topics than at reliably selecting the optimal ones. Moreover, enforcing a fixed topic count (as in Keep-Fixed-K-Accept) may restrict diversity and coverage. In contrast, the Filter-Reject strategy preserves flexibility in topic selection while eliminating weak content, striking a better balance between quality and coverage.

Impact of LLM Combinations in Multi-agent Debate

We investigate how different combinations of LLMs in the multi-agent debate framework influence the characteristics of generated topics. We combine various topic generation models (Debate LLMs) with a fixed Judge model (Gemma3-1B), applying a Filter-Reject topic selection strategy. We quantify the abstraction level of topics by computing their average hierarchical depth in WordNet (Miller 1995), classifying topics with a depth less than 3 as highly abstract, and those with a depth greater than 9 as highly specific. Finally, we analyze how these abstraction levels impact downstream performance on different task types.

Different LLM combinations naturally lead to different distributions of topic abstraction. As shown in Figure 3(a), topic sets with a greater proportion of abstract topics yield



(a) (b)

Figure 3: Impact of Topic Abstraction on Different Task Types. (a) The proportion of highly abstract topics positively correlates with performance in reasoning-oriented tasks (RAG, ICL, Re-rank, LongQA). (b) The proportion of highly specific topics contributes more to performance in memory-intensive tasks (Recall, RULER). **Model combinations:** *Gemma & Mistral* refers to Debate LLMs Gemma3-12B and Mixtral-8x7B-v0.1; *Qwen & Mistral* uses Qwen2.5-7B and Mixtral-8x7B-v0.1; *Reason (LLaMA & Qwen)* (DeepSeek-AI 2025) uses Reason-LLaMA-8B and Reason-Qwen2.5-7B (DeepSeek-AI 2025).

Model	Short Tasks Avg
Base model	62.05
LiteLong	61.92

Table 6: Comparison of model performance on short-context tasks, showing that LiteLong maintains comparable performance to the base model. Detailed results for each individual dataset can be found in Appendix B.

Model	Overall	GPU hours
SOTA	30.4	928
+ LiteLong	**30.6**	**176**

Table 7: LiteLong performs strongly after supervised fine-tuning while maintaining its resource-efficient merit, as evaluated on LongBench v2 benchmarks. Result details are presented in Appendix B.

better performance on reasoning-oriented tasks such as Retrieval-Augmented Generation (RAG), In-Context Learning (ICL), Re-ranking, and LongQA. These tasks benefit from broader conceptual coverage and generalization.

In contrast, Figure 3(b) shows that task performance on memory-intensive evaluations, such as Recall and RULER, is more strongly influenced by specific, narrowly defined topics. Such topics better match the need for detailed content anchoring and information recall.

These findings suggest that the abstraction profile of topic sets plays a critical role in shaping downstream task performance, as it is driven by the LLMs used during generation. Selecting or mixing LLMs with appropriate abstraction tendencies can thus serve as a tool to tailor training data for different long-context task categories.

LiteLong Maintains Performance on Short Tasks

We also examine whether the LiteLong approach compromises performance on short-context tasks. As shown in Table 6, LiteLong achieves a score of 61.92 on short-context benchmarks, closely matching the performance of the base model of 62.05. This marginal difference demonstrates that LiteLong introduces minimal interference with short-context capabilities. In other words, the integration of long-context data, which is carefully synthesized through topic-driven document aggregation, does not hinder the model's proficiency in conventional short-sequence NLP tasks. These results confirm that LiteLong effectively preserves short-context generalization while simultaneously strengthening long-context understanding.

LiteLong Performs Strongly After Supervised Finetuning

Finally, we investigate the potential of LiteLong after supervised fine-tuning and evaluate on the LongBench v2 benchmarks (Bai et al. 2024). For fair comparison, we follow the supervised fine-tuning procedure from the most recent SOTA method NExtLong (Gao et al. 2025b). As shown in Table 7, combined with LiteLong, the model achieves a competitive score of 30.6, on par with NExtLong's 30.4, while requiring only 19% of the GPU resources used by NExtLong. The results indicate that LiteLong's resource efficiency advantage does not come at the expense of downstream performance, demonstrating the enormous potential of LiteLong's applications.

Conclusion

This paper introduces LiteLong, a resource-efficient method for synthesizing high-quality long-context training data. Traditional Random Concatenation methods are computationally efficient but lack coherence, while Similarity-Based and Query-Centric approaches improve coherence at the cost of computational overhead. LiteLong employs a BISAC-guided structure and a multi-agent debate mechanism to generate diverse topics and retrieves documents via lightweight BM25 to build 128K-token sequences. Experiments on the HELMET and RULER benchmarks show that LiteLong delivers competitive performance while keeping computational costs low. Future work may explore other modalities and incorporate more diverse retrieval strategies.

Acknowledgments

This work is supported by the National Key R&D Program of China (Grant No. 2022ZD0117802).

References

Bai, Y.; Lv, X.; Zhang, J.; Lyu, H.; Tang, J.; Huang, Z.; Du, Z.; Liu, X.; Zeng, A.; Hou, L.; Dong, Y.; Tang, J.; and Li, J. 2024. LongBench: A Bilingual, Multitask Benchmark for Long Context Understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 3119–3137. Bangkok, Thailand: Association for Computational Linguistics.

Bairi, R.; Sonwane, A.; Kanade, A.; C, V. D.; Iyer, A.; Parthasarathy, S.; Rajamani, S.; Ashok, B.; and Shet, S. 2024. Codeplan: Repository-level coding using llms and planning. *Proceedings of the ACM on Software Engineering*, 1(FSE): 675–698.

Ben Allal, L.; Lozhkov, A.; Penedo, G.; Wolf, T.; and von Werra, L. 2024. SmolLM-Corpus.

De Cao, N.; Aziz, W.; and Titov, I. 2018. Question answering by reasoning across documents with graph convolutional networks. *arXiv preprint arXiv:1808.09920*.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948.

Du, Y.; Li, S.; Torralba, A.; Tenenbaum, J. B.; and Mordatch, I. 2023. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning*.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The llama 3 herd of models. *arXiv e-prints*, arXiv–2407.

Gao, C.; W, X.; Fu, Q.; and Hu, S. 2025a. Quest: Query-centric Data Synthesis Approach for Long-context Scaling of Large Language Model. In *The Thirteenth International Conference on Learning Representations*.

Gao, C.; Wu, X.; Lin, Z.; Zhang, D.; and Hu, S. 2025b. Nextlong: Toward effective long-context training without long documents. *arXiv preprint arXiv:2501.12766*.

Guu, K.; Lee, K.; Tung, Z.; Pasupat, P.; and Chang, M. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, 3929–3938. PMLR.

Hsieh, C.-P.; Sun, S.; Kriman, S.; Acharya, S.; Rekesh, D.; Jia, F.; and Ginsburg, B. 2024. RULER: What's the Real Context Size of Your Long-Context Language Models? In *First Conference on Language Modeling*.

Irving, G.; Christiano, P.; and Amodei, D. 2018. AI safety via debate. *arXiv preprint arXiv:1805.00899*.

Jiang, A. Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Hanna, E. B.; Bressand, F.; et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Jiang, H.; Wu, Q.; Luo, X.; Li, D.; Lin, C.-Y.; Yang, Y.; and Qiu, L. 2023. Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression. *arXiv preprint arXiv:2310.06839*.

Kenton, Z.; Siegel, N. Y.; Kramar, J.; Brown-Cohen, J.; Albanie, S.; Bulian, J.; Agarwal, R.; Lindner, D.; Tang, Y.; Goodman, N.; and Shah, R. 2024. On scalable oversight with weak LLMs judging strong LLMs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Levine, Y.; Wies, N.; Jannai, D.; Navon, D.; Hoshen, Y.; and Shashua, A. 2022. The Inductive Bias of In-Context Learning: Rethinking Pretraining Example Design. In *International Conference on Learning Representations*.

Lozhkov, A.; Ben Allal, L.; von Werra, L.; and Wolf, T. 2024. FineWeb-EDU. https://huggingface.co/datasets/HuggingFaceFW/fineweb-edu. Accessed: 2025-04-13.

Martínez-Ávila, D. 2016. BISAC: Book Industry Standards and Communications. *Knowledge Organization*, 43(8).

Miller, G. A. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11): 39–41.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Gray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Advances in Neural Information Processing Systems*.

Park, J. S.; O'Brien, J.; Cai, C. J.; Morris, M. R.; Liang, P.; and Bernstein, M. S. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, 1–22.

Peng, B.; Quesnelle, J.; Fan, H.; and Shippole, E. 2024. YaRN: Efficient Context Window Extension of Large Language Models. In *The Twelfth International Conference on Learning Representations*.

rednote hilab. 2025. dots.llm1 Technical Report. *arXiv preprint arXiv:TBD*.

Robertson, S.; Zaragoza, H.; et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4): 333–389.

Shi, W.; Min, S.; Lomeli, M.; Zhou, C.; Li, M.; Lin, X. V.; Smith, N. A.; Zettlemoyer, L.; tau Yih, W.; and Lewis, M. 2024. In-Context Pretraining: Language Modeling Beyond Document Boundaries. In *The Twelfth International Conference on Learning Representations*.

Team, G.; Kamath, A.; Ferret, J.; Pathak, S.; Vieillard, N.; Merhej, R.; Perrin, S.; Matejovicova, T.; Ramé, A.; Rivière, M.; et al. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Team, Q. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Tian, J.; Zheng, D.; Cheng, Y.; Wang, R.; Zhang, C.; and Zhang, D. 2024. Untie the knots: An efficient data augmentation strategy for long-context pre-training in language models. *arXiv preprint arXiv:2409.04774*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, Y.; Kordi, Y.; Mishra, S.; Liu, A.; Smith, N. A.; Khashabi, D.; and Hajishirzi, H. 2022. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Wu, S.; Iyer, R.; Wang, Y.; Bonilla, S.; Zhou, L.; Du, Y.; and Xu, W. 2023a. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*.

Wu, Y.; Iso, H.; Pezeshkpour, P.; Bhutani, N.; and Hruschka, E. 2023b. Less is more for long document summary evaluation by llms. *arXiv preprint arXiv:2309.07382*.

Xiong, W.; Liu, J.; Molybog, I.; Zhang, H.; Bhargava, P.; Hou, R.; Martin, L.; Rungta, R.; Sankararaman, K. A.; Oguz, B.; et al. 2023. Effective long-context scaling of foundation models. *arXiv preprint arXiv:2309.16039*.

Xu, C.; Sun, Q.; Zheng, K.; Geng, X.; Zhao, P.; Feng, J.; Tao, C.; Lin, Q.; and Jiang, D. 2024. WizardLM: Empowering Large Pre-Trained Language Models to Follow Complex Instructions. In *The Twelfth International Conference on Learning Representations*.

Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; and et al. 2024. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*.

Yen, H.; Gao, T.; Hou, M.; Ding, K.; Fleischer, D.; Izsak, P.; Wasserblat, M.; and Chen, D. 2025. HELMET: How to Evaluate Long-context Models Effectively and Thoroughly. In *The Thirteenth International Conference on Learning Representations*.

Young, A.; Chen, B.; Li, C.; Huang, C.; Zhang, G.; Zhang, G.; Wang, G.; Li, H.; Zhu, J.; Chen, J.; et al. 2024. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*.