日本語ロングコンテキスト処理能力評価ベンチマークの開発

玉腰 勇司 [1*] 中山 功太 [2] 児玉 貴志 [2] 宮尾 祐介 [1,2]
[1] 東京大学 [2]NII LLMC
uotstudent2001@g.ecc.u-tokyo.ac.jp {nakayama,tkodama}@nii.ac.jp
yusuke@is.s.u-tokyo.ac.jp

概要

大規模言語モデル（LLM）のロングコンテキスト処理能力を評価するためのベンチマークが数多く提案され，その知見はモデル改良に活用されてきた．一方で，日本語においてはそのようなベンチマークが未整備であるという課題がある．本研究では，JEMHop および NIILC データセットを基盤として，日本語における LLM のロングコンテキスト処理能力を評価する初のベンチマークを構築し，複数の代表的な LLM を用いて評価を行った．その結果，モデル間でロングコンテキスト耐性に差が見られること，および情報の位置によって性能が変動する傾向が日本語環境においても確認された．

1 はじめに

LLM の開発は盛んに行われているが，新しいモデルが開発されるにつれ，より多くの入力を受け付けられるようになっている．これは，長文要約や，複数の文章にまたがった質問応答といった需要に答えるためである．実際，OpenAI が 2023 年にリリースした GPT-4[8] の最大コンテキスト長は 8k だったが，2025 年にリリースされた GPT-5[10] は 200k である．

LLM のロングコンテキスト処理能力を評価すべく，様々なベンチマークが構築されている．ここで得られた知見がより良いモデル開発につながっている．一方，日本語において，それを評価するようなデータセットはない．

本研究においては，JEMHop[6]，NIILC[11] データセットを用いて，日本語で LLM のロングコンテキスト処理能力を評価する初のベンチマークを作成し，複数の LLM を評価した．

2 関連研究

LLM のロングコンテキスト処理能力は，与えられた文章から質問に解答する能力，文章を要約する能力など多岐にわたり，それぞれの能力を評価するベンチマークが求められる．それらを網羅的に評価するベンチマークを構築した研究としては，ZeroScroll[12]， LongBench[4], L-Eval[1] がある．

一方，既存のベンチマークによってロングコンテキスト処理能力が正しく測れているかは疑念視されている．Bai らは，ロングコンテキスト処理能力の評価のために作成されたベンチマークの多くは，文章中から単語を抜き出すという簡単なタスクであることや，タスク設定が非現実的であること，評価指標に信頼性がないことを指摘し，より良いベンチマークとして，LongBench v2[3] を提案している．他にも，同様の問題意識を持って作成されたベンチマークとして，HELMET[15] がある．

ところで，情報の長さだけでなく，その位置も重要である．Liu らと Xu らの研究によって，LLM は，コンテキスト中の情報のうち，文頭と文末の情報を重視し，中間にある情報を取りこぼすことが指摘されている [7][13]．そして，Shengnan らはその現象は学習データが含むバイアスによるものだと仮説をたて，この課題を解決するためのトレーニング方法を提案している [2].

このように，LLM を評価することで得られた知見がより良いモデル開発に活かされている．LLM のロングコンテキスト処理能力について正しく評価し，理解するためのベンチマークが整備されることで，そこから得られた知見がより良い LLM の開発に繋がっていくと考えられる．一方，日本語ではそのようなベンチマークが整備されていない．

* 本研究は NII LLMC 在籍時の成果です．

3 データ作成

3.1 既存のデータセットのロングコンテキスト化

3.1.1 JEMHop

JEMHop は，説明可能な QA システムの開発のために作成された，回答導出ステップ情報付きの日本語のマルチホップ QA データセットである [6]．データセットは，質問の ID，質問の種類，質問，解答，解答に至る推論，根拠となる Wikipedia の ID，質問の時間依存性の有無からなる．

JEMHop データセットに紐づけられている Wikipedia 記事を質問と結合することで，ロングコンテキストデータセットを作成した．Wikipedia は JEMHop で提供されている，CirrusDump の 2021 年 8 月 23 日のデータを用いた[1)]．本研究では，各 Wikipedia 記事に付与されている ID を利用して，3.2 節で処理した Wikipedia 記事を各質問に紐づけた．

付録 A 図 2 に作成されたデータセットの一例，付録 B 表 3 に，質問とコンテキストを合算したプロンプトのトークン長分布を添付する．

3.1.2 NIILC

NIILC は，質問応答システムの研究のために作成されたデータセットである [11]．データセットは，質問の ID，質問，単一あるいは複数の解答の他に大量のメタデータを含む．

NIILC データセットを用いて，トークン長と正例の位置を指定できるロングコンテキストデータセットを作成した．まず，使用しない質問の削除を行った．説明を要求する質問については，解答が文章となるため削除した．また，解答は数やその性質によって，唯一，時間的唯一，複数，曖昧の分類がされている．今回は，分類が唯一の質問だけを対象とし，それ以外のタグを持つもの，タグが付与されていないものの，複数の回答が存在するものは除外した．

次に，質問とそれに対応した Wikipedia 記事が紐づいたデータセットを作成した．Wikipedia は CirrusDump の 2025 年 9 月 15 日のデータを用いた[2)]．NIILC の質問の一部は，解答の根拠となる Wikipedia 記事の見出し情報が付与されている．これを利用し Wikpedia 記事を紐付けた．この時，見出しが変更されているため紐付けができない質問があったが，その質問は取り除いた．

最後に，トークン長と正例の位置を指定してデータセットを作成できるようにした．トークン長を指定すると，その質問のコンテキスト（正例）と，いくつかの，他の質問のコンテキスト（負例）を合わせることで，一つのコンテキストが作成される．合計のトークン長が指定したトークン長を超えてしまう場合は，追加する負例の一つの文章を途中で切ることでトークン長を調整する．作成時，正例のトークン長が指定の長さを超える場合はその質問は取り除かれる．トークン長の指定は正例と負例の合計に対して行うため，質問文によって LLM に与えられるトークン長はより長くなる．そのため，その分を考慮し，256 トークン差し引いた値を指定することとする．また，正例の挿入位置により 4 つの評価設定を導入する．挿入位置は，コンテキストの先頭，中央，末尾，ランダムがあり，対応した位置に正例を挿入する．質問は，コンテキストの直後に結合する．

付録 A 図 3 に作成されたデータセットの一例．付録 B 表 4 に，各トークン長において作成されたデータセットの平均トークン長とその質問数を示す．

3.2 Wikipedia 前処理

本研究では，日本語 Wikipedia CirrusDump を入力として記事本文テキストを構築した．ノイズとなるナビゲーション系テンプレートやメタデータ表を除去する一方で，infobox や本文中の表など意味的情報を含む要素は保持する設計とした．

前処理後，各記事から連続した本文テキストを生成し，記事ごとに ID，タイトル，本文テキストを保存した．前処理の具体的な手順および除去対象の詳細は付録 D に示す．

4 実験

以降，K=1,024 とし，k=1,000 と区別する．

4.1 モデル

本研究では，評価対象のモデルとして，gpt-oss-20b[9]，Llama-3 系列の Meta-Llama-3.1-8B-Instruct，Llama-3.2-3B-Instruct，Llama-3.3-70B-Instruct [5]，および Qwen3 系列の Qwen3-30B-A3B-Instruct-2507，Qwen3-30B-A3B-Thinking-2507，Qwen3-Next-80B-

1） https://github.com/aiishii/JEMHopQA
2） https://dumps.wikimedia.org/other/cirrussearch/

表 1 各 LLM における，JEMHop のトークン長と正解率 (各列は前列のトークン長上限以上)

タイプ	モデル	トークン長				
		コンテキストなし	8K	16K	32K	64K
Instruct	Llama-3.2-3B-Instruct	6.8	0.0	0.4	0.0	0.0
Instruct	Llama-3.3-70B-Instruct	35.2	75.4	65.7	56.9	38.0
Instruct	Meta-Llama-3.1-8B-Instruct	22.1	24.6	11.1	1.8	0.0
Instruct	Qwen3-30B-A3B-Instruct-2507	35.0	66.7	60.2	59.8	56.9
Instruct	Qwen3-Next-80B-A3B-Instruct	41.2	54.2	46.8	48.9	50.4
Thinking	gpt-oss-20b	2.1	5.7	6.0	9.7	12.9
Thinking	Qwen3-30B-A3B-Thinking-2507	39.0	77.5	80.1	77.6	74.4
Thinking	Qwen3-Next-80B-A3B-Thinking	41.0	51.7	55.4	53.2	57.3

表 2 各 LLM における，NIILC のトークン長と正解率

タイプ	モデル	トークン長						
		コンテキストなし	8K	16K	32K	64K	96K	120K
Instruct	Llama-3.2-3B-Instruct	7.5	4.3	1.1	0.0	0.0	0.0	0.0
Instruct	Llama-3.3-70B-Instruct	50.9	64.7	64.8	62.5	54.5	20.4	0.7
Instruct	Meta-Llama-3.1-8B-Instruct	30.0	42.6	26.1	14.2	1.3	0.7	0.6
Instruct	Qwen3-30B-A3B-Instruct-2507	35.7	61.2	57.1	58.0	57.3	57.2	57.7
Instruct	Qwen3-Next-80B-A3B-Instruct	45.6	58.1	57.9	58.2	58.7	58.3	60.3
Thinking	gpt-oss-20b	1.3	18.5	17.6	22.5	22.5	19.0	17.8
Thinking	Qwen3-30B-A3B-Thinking-2507	24.5	59.3	57.6	59.8	58.9	59.6	56.6
Thinking	Qwen3-Next-80B-A3B-Thinking	36.4	51.6	52.5	53.5	54.8	54.6	54.4

A3B-Instruct，Qwen3-Next-80B-A3B-Thinking [14] を用いた．gpt-oss-20b，Llama3 のコンテキスト長は 128k トークン，Qwen3 は 256K トークンである．

4.2 評価方法

LLM に<Answer></Answer>タグをつけて解答を出力させた．プロンプトは付録 C を参照．タグの間の解答を抽出し，それと，データセットの模範解答を正規化して完全一致で比較した．正規化の方法は，二つのデータセットで異なる．NIILC については，数字と小数点を半角にそろえる正規化を行う．JEMHop については，末尾の句点を落とし，前後の空白を削り，日本語のはい／いいえや，大小さまざまな YES/NO を YES or NO に統一する．

4.3 トークン長と正答率の関係

LLM が与えられたコンテキストを活用できているか調べるため，コンテキストのトークン長を変更して，正答率を評価した．

JEMHop JEMHop データセットは，付録 B 表 3 のようなトークン長分布を持つ．よって，質問をトークン長が 0 以上 8K 未満，8K 以上 16K 未満，16K 以上 32K 未満，32K 以上 64K 未満の場合で分割し，正答率を評価した．加えて，そもそも LLM が Wikipedia を学習しておりコンテキストが正答率の向上に寄与しない可能性があるため，コンテキストを与えない場合でも正答率を評価した．モデルの最大トークン長は 128K トークンとし，生成時の温度パラメータは全てのモデルで 0.0 とした．

NIILC NIILC データセットを利用して，所望のトークン長のロングコンテキストデータセットを作成できる．今回，トークン長を 8K，16K，32K，64K，96K，120K に設定して，正答率を評価した．加えて，コンテキストを与えない場合でも正答率を評価した．正例の位置はランダムとした．モデルの最大トークン長を 128K トークンとし，生成時の温度パラメータは全てのモデルで 0.0 とした．

JEMHop，NIILC に対する結果はそれぞれ表 1，2 となった．

どちらのデータセットでも，Llama-3.2-3B-Instruct モデルを除いて，コンテキストなしと 8K トークンの結果を比較すると，8K の方が正答率が高い．これは，質問に関連する情報を LLM が参照していることを意味する．一方で，トークン長をさらに伸ばしていくと，Llama-3 モデルで正答率が低下する傾向が見られた．実際の生成文章を確認すると，付録 E の表 5 のように，トークン長が長い条件ではモデルが指示どおりに <Answer>…</Answer> 形式で出力できないケースが増加しており，たとえ内容的には正答に近い応答を生成していても，フォーマット崩れによって不正解として扱われる事例が多かった．すなわち，本タスクにおける正答率は，「コンテキストを与えることによる情報量のメリット」と，「長文コンテキスト下で全体の文脈を把握する能力」という二つの要因のバランスで決まっていると解釈できる．Llama-3.2-3B-Instruct において，コンテキストを与えることでかえって正答率が低下したのも，このトレードオフによるものだと考えられる．また，トークン長の増加に伴う性能劣化の度合いは一様ではない．大規模モデルや長コンテキスト対応モデルほど，長いコンテキストでも正答率の低下が緩やかであり，モデル規模および最大コンテキスト長が，情報量と出力制御のトレードオフをどこまで許容できるかを規定している可能性が示唆される．加えて，thinking モデルである gpt-oss-20b は，最大コンテキスト長が 128k にも関わらず，大きな性能の低下が発生しなかったという点で，長い文脈を適宜参照しながら解答を生成していることがわかる．

4.4 正例の位置と正答率の関係

長文コンテキストを扱う LLM の実運用においては，複数の参考情報の一部のみが正解に必要となる状況が一般的である．このとき，モデルには情報の位置に依らず適切な参照を行う能力が求められる．既存研究では，文中における情報位置が正答率に影響を与えることが示されているが [7][13]，多くは英語データを対象としており，日本語データについては十分に検証されていない．本実験では，最大 120K トークンまでのコンテキストにおいて，NIILC データセットを用いて，正例の位置と正答率の関係を評価する．Llama-3.3-70B-Instruct モデルを対象にし，トークン長を 8K，16K，32K，64K，96K，120K に設定した上で，正例の位置を先頭，中央，末尾に設定して正答率を評価した．

結果を図 1 に示す．先頭, 中央では正答率が変わらなかった一方，32K，64K，96K では，末尾において，他二つの位置より正答率が高かった．

既存研究においては，解答の根拠となる文章が文中にある場合は，文頭，あるいは文末に置いた場合に比べて正答率が低いが，このデータセットにおいてはそれは起こらなかった [7][13]．しかし，正例が文末にあるとき正答率が向上しており，正例の位置が確かに正答率に影響を与えていることがわかる．そしてその影響は，与えられるコンテキストのトークン長が大きいほど大きくなると考えられる．トークン長が 64K トークンより大きい時には，64K の時と比較して差分が小さくなっているが，これは，付録 E の表 5 のように，96K トークン以上では回答率が大きく下がっているためである．

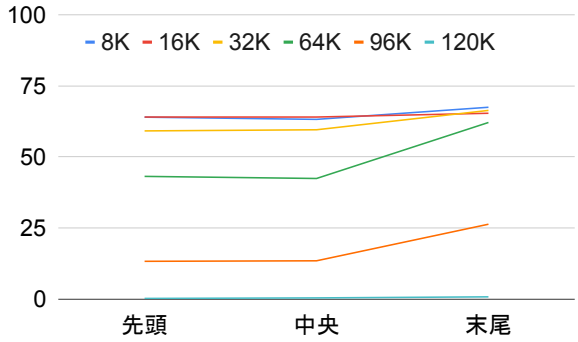

図 1 各トークン長における正例の位置と正答率

5 結論

本研究では，LLM のロングコンテキスト処理能力を評価できる日本語のベンチマークが存在しない課題を解決すべく，JEMHop データセット，NIILC データセットを活用することでベンチマークを作成した．そして，様々な LLM を評価した．

結果，長いコンテキストを与えると，それが質問の根拠となる情報を含んでいても，正答率が低下する場合があり，その長さに対する影響度合いはモデル間で異なること，また，コンテキスト全体における情報の位置も重要となることが，日本語環境でも確認された．

今回作成されたデータセットは解答を単語とする QA に限られており，今後の展望として，文章を解答とするものや，要約などのタスクをカバーする，網羅的なベンチマークの作成が考えられる．

謝辞

本研究成果の一部は，データ活用社会創成プラットフォーム mdx を利用して得られたものです．

References

[1]Chenxin An et al. “L-Eval: Instituting Standardized Evaluation for Long Context Language Models”. In: **Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 14388–14411. DOI: 10.18653/v1/2024.acl-long.776. URL: https://aclanthology.org/2024.acl-long.776/.

[2]Shengnan An et al. “Make your LLM fully utilize the context”. In: **Proceedings of the 38th International Conference on Neural Information Processing Systems**. NIPS ’24. Vancouver, BC, Canada: Curran Associates Inc., 2024. ISBN: 9798331314385.

[3]Yushi Bai et al. “LongBench v2: Towards Deeper Understanding and Reasoning on Realistic Long-context Multitasks”. In: **arXiv preprint arXiv:2412.15204** (2024).

[4]Yushi Bai et al. “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)**. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 3119–3137. DOI: 10.18653/v1/2024.acl-long.172. URL: https://aclanthology.org/2024.acl-long.172/.

[5]Aaron Grattafiori et al. **The Llama 3 Herd of Models**. 2024. arXiv: 2407.21783 [cs.AI]. URL: https://arxiv.org/abs/2407.21783.

[6]Ai Ishii et al. “JEMHopQA: Dataset for Japanese Explainable Multi-Hop Question Answering”. In: **Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)**. Ed. by Nicoletta Calzolari et al. Torino, Italia: ELRA and ICCL, May 2024, pp. 9515–9525. URL: https://aclanthology.org/2024.lrec-main.831/.

[7]Nelson F. Liu et al. “Lost in the Middle: How Language Models Use Long Contexts”. In: **Transactions of the Association for Computational Linguistics** 12 (2024), pp. 157–173. DOI: 10.1162/tacl_a_00638. URL: https://aclanthology.org/2024.tacl-1.9/.

[8]OpenAI. **GPT-4**. https://openai.com/index/gpt-4-research/. 2023.

[9]OpenAI. **Introducing gpt-oss**. https://openai.com/index/introducing-gpt-oss/. 2025.

[10]OpenAI. **Introducing GPT-5 for developers**. https://openai.com/index/introducing-gpt-5-for-developers/. 2025.

[11]Satoshi Sekine. “Development of a question answering system focused on an encyclopedia”. In: **Proceedings of the 9th Annual Meeting of the Association for Natural Language Processing**. (in Japanese). 2003.

[12]Uri Shaham et al. “ZeroSCROLLS: A Zero-Shot Benchmark for Long Text Understanding”. In: **Findings of the Association for Computational Linguistics: EMNLP 2023**. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 7977–7989. DOI: 10.18653/v1/2023.findings-emnlp.536. URL: https://aclanthology.org/2023.findings-emnlp.536/.

[13]Peng Xu et al. “Retrieval meets Long Context Large Language Models”. In: **The Twelfth International Conference on Learning Representations**. 2024. URL: https://openreview.net/forum?id=xw5nxFWMlo.

[14]An Yang et al. **Qwen3 Technical Report**. 2025. arXiv: 2505.09388 [cs.CL]. URL: https://arxiv.org/abs/2505.09388.

[15]Howard Yen et al. “HELMET: How to Evaluate Long-Context Language Models Effectively and Thoroughly”. In: **International Conference on Learning Representations (ICLR)**. 2025.

A　データセット

作成されたデータセットの一例を図 2,3 に示す．

Question: 『仮面ライダー電王』と『あまちゃん』、放送回数が多いのはどちらでしょう？
Context: {{半保護}}{{Pathnav|仮面ライダーシリーズ... 中略 ... 毎週日曜 8:00 - 8:30（JST）に全 49 話が放映された... 中略 ... 番組名 = あまちゃん... 中略 ... 2013 年 4 月 1 日 - 2013 年 9 月 28 日| 放送回数 1 = 156 ... 中略 ...
Answer: あまちゃん

図 2　JEMHop から作成されたロングコンテキストデータセットの一例．

Question: 初めてノート型パソコンを作ったメーカーは？
Context: {{Otheruses|球技のソフトボール ... 中略 ... ノートパソコン... 中略 ... 同年 7 月に東芝から発売された DynaBook（現・dynabook）J-3100SS は、19 万 8,000 円という価格で衝撃を与えた。... 中略 ...
Answer: 東芝

図 3　NIILC から作成されたロングコンテキストデータセットの一例．トークナイザーとして Llama-3.3-70B-Instruct を使用し，トークン長は 64K，正例の位置はランダム

B　作成されたデータセットのトークン長分布

表 3 に，JEMHop の質問とコンテキストを合算したプロンプトのトークン長分布を添付する．表 4 に，NIILC から作成されたデータセットの平均トークン長とその質問数を示す．トークナイザーは Llama-3.3-70B-Instruct である．

表 3　JEMHop から作成されたロングコンテキストデータセットのトークン長ごとの質問数

範囲	平均	質問数
0 以上 8K 未満	5,453	126
8K 以上 16K 未満	12,388	262
16K 以上 32K 未満	23,485	332
32K 以上 64K 未満	46,048	258
64K 以上	96,293	81

C　使用したプロンプト

JEMHop で使用したプロンプトは以下である．
以下は与えられた文章です。

表 4　NIILC から作成されたロングコンテキストデータセットのトークン長ごとの質問数

設定値	平均	質問数
8K	8,014	258
16K	16,206	375
32K	32,590	499
64K	65,358	538
96K	98,126	544
120K	122,702	544

`{context}`
回答は答えのみを出力し、`<Answer></Answer>`タグで囲んでください。
質問：`{question}` 回答：

NIILC で使用したプロンプトは以下である．
与えられた文章を読んで質問に答えてください。
文章：`{context}`
回答は答えのみを出力し、`<Answer></Answer>`タグで囲んでください。
質問：`{question}`

D　Wikipedia 処理詳細

各記事について source_text を取得し，ナビゲーションテンプレート，スタブ通知などのノイズ的テンプレートを正規表現により削除した．また，クラス属性に navbox，sidebar，metadata を含む表を除去しつつ，infobox や通常の表は保持した．

次に，引用用タグやスクリプトを含む各種 HTML タグおよびコメントを削除した．行単位で，箇条書きマークの単純化，強調記号 (”, ”’)，外部リンク表記などを除去した．

また，`[[...]]` 形式の内部リンクについては，表示文字列のみを本文中に残す処理を行った．

E　トークン長と回答率

各トークン長，モデルにおいて，解答が抽出できた質問の割合（回答率）は表 5 の通り．

表 5　各モデルにおけるトークン長と回答率

トークン長	Llama-3.2-3B-Instruct	Llama-3.3-70B-Instruct	Qwen3-Next-80B-A3B-Thinking
コンテキストなし	51.3	99.8	100.0
8K	5.4	96.9	100.0
16K	2.9	97.6	99.5
32K	0.0	94.8	99.4
64K	0.0	81.2	99.4
96K	0.0	31.4	99.4
120K	0.0	1.1	98.9