継続事前学習による日本語に強い大規模言語モデルの構築

藤井 一喜 [1] 中村 泰士 [1] Mengsay Loem [1] 飯田 大貴 [1,2] 大井 聖也 [1]
服部 翔 [1] 平井 翔太 [1] 水木 栄 [1,3] 横田 理央 [1] 岡崎 直観 [1]
[1] 東京工業大学情報理工学院 [2] 株式会社レトリバ [3] 株式会社ホットリンク

概要

本研究では，Llama 2 をベースに日本語の大規模ウェブコーパスで継続事前学習を行い，日本語能力を強化した大規模言語モデル Swallow を構築した．実験から，7B, 13B, 70B のいずれのモデル規模においても，継続事前学習が大規模言語モデルの日本語能力を引き上げ，高い性能を達成することが分かった．また，継続事前学習の学習データ量の増加に伴い，日本語の性能が向上すること（学習のスケール性）を確認した．構築した Swallow モデルを公開し，コミュニティでの後続研究・活用を期待している．

1 はじめに

OpenAI の ChatGPT や Google の Bard などの大規模言語モデル（LLM）は，人間に近い言語理解能力と生成能力，さまざまな分野への適用可能性を示し，大きな注目を集めた．日本でも LLM 開発の機運が高まり，2023 年は日本語に強い LLM（以降，日本語の LLM と呼ぶ）の開発・発表が盛んに行われた．ところが，日本語の LLM 開発において日本語以外の言語資源，特に英語の言語資源の活用について，未だ知見が少ないのが現状である．

英語は事実上の世界共通言語であることから，世の中に存在する言語資源の中で英語の割合が突出している．Common Crawl が発表している統計情報によると，英語のウェブページは日本語のウェブページの約 9 倍の量であると推定される[1)]．また，論文の採択・引用されやすさ等の理由も加わり，良質な注釈付きコーパスやデータセットは英語をターゲットとして開発されやすい傾向がある．このため，日本語を含む多くの非英語言語では，高品質なコーパスやデータセットが不足しがちである．

英語の言語資源を活用した日本語の LLM 構築に関する知見を得るには，日本語と英語の言語データで LLM の事前学習を行うのが自然であろう．ところが，これには膨大な計算資源が必要であり，計算資源が限られた状況では取り組みにくい．そこで，本研究では英語の LLM の能力や知識を日本語に転移することを狙い，英語の LLM からの継続事前学習（continual pre-training）に取り組む．

継続事前学習，或いは追加事前学習（further pre-training）は，事前学習済みの言語モデルを下流タスクでファインチューニングする前に，当該タスクのテキストで事前学習タスクを追加的に実施し，ドメイン適応を行う手法であった [1, 2, 3]．オープンで高性能な英語の LLM が登場してからは，継続事前学習を行い，LLM を他のタスクや言語に適応をさせる試みが増えている [4, 5, 6, 7, 8]．継続事前学習で構築された日本語の LLM も公開されているが[2)3)]，継続事前学習の効果について網羅的な調査は行われていない．例えば，継続事前学習に用いる日本語データの量とモデルの性能の関係や，サイズの異なるモデルにおける効果の検証，フルスクラッチで学習したモデルとの比較は不十分である．

本研究では，Llama 2 7B, 13B, 70B に日本語の継続事前学習を行う．主な貢献は以下の通りである．

- すべてのサイズにおいて，継続事前学習による日本語能力の向上および有用性を確認した．
- フルスクラッチで学習した日本語 LLM よりも高い性能を効率よく発揮することを実証した．
- 日本語の学習データ量の増加に応じた性能向上（学習のスケール性）を示した．
- 本研究で構築した日本語 LLM である Swallow 7B, 13B, 70B を HuggingFace 上で公開した [4)]．
- Swallow 70B は llm-jp-eval において，日本国内で開発されたモデルの中で最高性能を達成した（2023 年 12 月現在）．

1） Statistics of Common Crawl Monthly Archives: https://commoncrawl.github.io/cc-crawl-statistics

2） ELYZA-japanese-Llama-2-7b: https://note.com/elyza/n/na405acaca130

3） Japanese Stable LM Beta: https://ja.stability.ai/blog/japanese-stable-lm-beta

4） https://tokyotech-llm.github.io/swallow-llama

2 手法

2.1 モデルのアーキテクチャ

表 1 に Swallow モデルのハイパーパラメータを示す．継続事前学習ではベースモデルからアーキテクチャを変更できないので，Swallow モデルは Llama 2 と同じ Transformer のデコーダを採用し，分散表現サイズ，注意ヘッド数，層数，文脈長は Llama 2 から変更していない．継続事前学習時には学習データを連結し，ちょうど 4096 個のトークンの系列長になるように学習データ分割し，パディングトークンは用いなかった．Llama 2 34B, 70B モデルに Grouped-Query Attention (GQA) が導入されているため，Swallow 70B でも GQA を採用した．また，重み減衰（weight decay）に 0.1，勾配クリッピング（gradient clipping）に 1.0 を使用した．さらに，計算効率の向上と省メモリ化のため，Flash Attention 2 [9] を採用した．

バッチサイズ Llama 2 では，グローバルバッチサイズが 4M トークンであり，事前学習時と同様のバッチサイズを設定するために，Swallow ではすべてのモデルサイズで 1024 のバッチサイズを使用した．

オプティマイザー AdamW [10] を採用した．ハイパーパラメータには $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 1.0 \times 10^{-8}$ を使用した．

学習率のスケジューリング コサイン波形による減衰（cosine learning rate scheduler）を利用し，学習率は 1,000 ウォームアップ・ステップで最大値に達し，最終的にはその 1/30 に減衰するように設定した．

2.2 モデルの分散並列学習

70B のモデルを学習するために必要なメモリは 1 枚の GPU メモリを超えるため，データ並列とモデル並列を併用した分散並列学習を採用した．

2.2.1 学習環境

学習には産業総合研究所の AI 橋渡しクラウド (ABCI) を利用した．混合精度（`bfloat16`）を採用し NVIDIA A100 ノードを複数台使用し，分散並列学習を行った．各ノードは NVIDIA A100 40GB GPU を 8 基を搭載し，ノード間は InfiniBand HDR にて接続されている．

2.2.2 分散学習手法

効率的に学習を行うために，データ並列（data parallelism），テンソル並列化（tensor parallelism），パイプライン並列化（pipeline parallelism）を統合した 3D 並列化（3D parallelism）を採用し，高い計算効率と効率的なメモリ利用を目指した．学習ライブラリには，Megatron-LM[5] を採用した．表 2 に各モデルサイズにおける分散学習の設定を示した[6]．他にも，以下に挙げる工夫を取り入れた．

効率的なメモリ消費 Megatron-LM の Distributed Optimizer を用いて，オプティマイザの状態変数（optimizer state）をデータ並列プロセス間に分散配置し，冗長性を排除することで必要なメモリ使用量を削減した．

トポロジーを考慮した 3D マッピング Transformer ブロックはパイプライン並列により複数の GPU に分散配置され，さらにテンソル並列により層内のパラメータが分散配置される．この際，Megatron-LM [11] で提案されているように，通信を多く必要とする分散手法のワーカー（テンソル並列）はノード内に配置した．また，データ並列の勾配平均化のための通信を考慮し，データ並列ワーカーも可能な限りノード内に配置した．パイプライン並列は他の並列化手法と比較して通信量が少ないため，パイプラインステージはノード間で配置した．

2.3 継続事前学習データ

継続事前学習に用いたコーパスは，日本語は Swallow コーパス [12] および日本語 Wikipedia[7]，英語は RefinedWeb [13] および The Pile [14] である．これらのコーパスから約 100B トークンをサンプリングし，継続事前学習に用いた．サンプリングは，RefinedWeb の英語テキストが 5%，The Pile の arXiv 論文テキスト（英語）が 5%とし，残りの 90%は日本語テキストが占めるように構成した。日本語テキ

5） https://github.com/NVIDIA/Megatron-LM

6） 7B, 13B モデルについては，実験途中で使用ノード数の変更せざるを得ない状況が発生したため，データ並列数が 2 倍もしくは半分になっている期間がある．これは本論文で説明する実験以外にも同時に進めた実験があるためで，限られた計算資源・期間で有望そうな設定を優先してモデル構築を進めたからである．

7） https://dumps.wikimedia.org/other/cirrussearch/ 2023 年 3 月 20 日付のダンプを使用．

表 1　Swallow モデルのアーキテクチャとハイパーパラメータ

パラメータ数	分散表現のサイズ	注意ヘッド数	層数	文脈長	GQA	トークン数	バッチ数	学習率
7B	4096	32	32	4096	無	約 100B	1024	1.0×10^{-4}
13B	5120	40	40	4096	無	約 100B	1024	1.0×10^{-4}
70B	8192	64	80	4096	有	約 100B	1024	5.0×10^{-5}

表 2　分散学習の設定．DP，TP，PP，SP はそれぞれ，データ並列 (Data Parallelism)，テンソル並列 (Tensor Parallelism)，パイプライン並列 (Pipeline Parallelism)，シークエンス並列 (Sequence Parallelism) を表す．

パラメータ数	DP	TP	PP	SP	Distributed Optimizer
7B	16	2	2	✓	✓
13B	8	2	4	✓	✓
70B	4	8	8	✓	✓

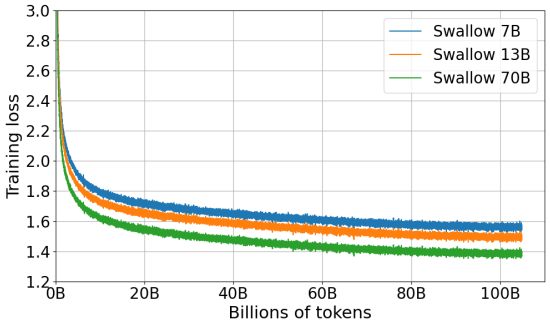

図 1　Swallow 7B，13B，70B の学習曲線（学習したトークン数と損失値）

ストは，約 1.6B トークンを日本語 Wikipedia から，残りを Swallow コーパスから抽出した．

2.4　トークン化

Llama 2 はバイト対符号化（BPE: byte-pair encoding）でトークン化を行うため，Swallow もそれを踏襲した．ただし，日本語の文字やサブワードを Llama 2 の語彙に追加し，43,176 語からなる語彙でトークン化を行った[8]．語彙拡張の詳細については，別の論文を参照されたい [15]．

2.5　学習の経過

Swallow モデルの学習曲線を図 1 に示す．Swallow 7B，13B，70B の学習にはそれぞれ，約 5.0×10^{21} FLOPs，約 9.4×10^{21} FLOPs，約 5.0×10^{22} FLOPs の計算量を要した．

3　評価

3.1　評価データセット

日本語の評価ベンチマークとして，llm-jp-eval [16] (v1.0.0) および JP Language Model Evaluation Harness[9] (commit #9b42d41)，機械翻訳は Language Model Evaluation Harness [17] (v0.3.0) を使用した．llm-jp-eval は，多値選択式質問応答（JCommonsenseQA [18]），自由記述式質問応答（JEMHopQA [19]，NIILC [20]），機械読解（JSQuAD [18]）を 4 ショット推論で評価した．なお，自然言語推論については言語モデルが予測するラベルが偏る傾向があり，偶然に正解ラベルの偏りと一致する場合にスコアが高くなるため，今回は評価対象から除外した．JP Language Model Evaluation Harness は，自動要約（XL-Sum [21]）を 1 ショット推論，算術推論（MGSM [22]）を 4 ショット推論で評価する．Language Model Evaluation Harness は，日英・英日機械翻訳（WMT 2020 Japanese↔English [23]）を 4 ショット推論で評価する．

3.2　汎用言語モデルの評価結果

表 3 に Swallow モデルおよびそのベースである Llama 2，そして日本国内で事前学習および継続事前学習を施した主要な日本語 LLM の評価結果を示す（モデルの出典は付録の表 4 参照）．なお，これらのモデルは指示チューニングを行っておらず，LLM の「素」の能力を評価している．

ベースである Llama 2 と比較すると，Swallow 7B は 7.1 ポイント，13B は 6.6 ポイント，70B は 7.0 ポイントの平均スコアの向上を達成した．特に，Swallow 70B は国内で構築された LLM の中で最も高い性能を示した．また，国内でフルスクラッチから学習された LLM（calm-7b, llm-jp-13b-v1.0, PLaMo-13B）と比較すると，Swallow の平均スコアは 8.4〜17.4 ポイント高く，継続事前学習の有用性が示された．

ただ，海外に目を向けると，フルスクラッチか

8）分散学習の効率化とテンソル並列の最適化のため，語彙数は 8 の倍数に調整した

9）https://github.com/Stability-AI/lm-evaluation-harness

表 3　日本語タスクにおける評価結果．VE (Vocabulary Expansion) は語彙拡張を表す．また，CT (Continual Pre-Training) は継続事前学習を表す．「日」は日本語による継続事前学習を，「日英」は日本語と英語による継続事前学習を表す．

モデル / 種類	MC	QA		RC	SUM	MATH	MT (WMT20)		平均
	JCom	JEMHop	NIILC	JSQuAD	XL-Sum	MGSM	En-Ja	Ja-En	
calm2-7b	0.2198	0.5047	0.5066	0.7799	0.0233	0.0600	0.2345	0.1499	0.3098
Japanese Stable LM Beta 7B (CT, 日英)	0.3610	0.4478	0.4432	0.8318	0.2195	0.0720	0.1946	0.1226	0.3366
ELYZA-japanese-Llama-2-7b (CT, 日)	0.5791	0.4703	0.4019	0.8226	0.1312	0.0600	0.1795	0.1289	0.3467
youri-7b (CT, 日英)	0.4620	0.4776	0.4999	0.8506	0.1957	0.0640	0.2671	**0.1971**	0.3768
Mistral v0.1 7B	0.7301	0.4245	0.2722	0.8563	0.2006	0.1760	0.1405	0.1733	0.3717
japanese-stablelm-base-gamma-7b (CT, 日英)	0.7364	0.4643	0.5568	**0.8910**	**0.2293**	0.1680	0.2390	0.1561	**0.4301**
Qwen-7B	**0.7712**	0.4234	0.2376	0.8594	0.1371	**0.2160**	0.1689	0.1801	0.3742
nekomata-7b (CT, 日英)	0.7417	0.4928	0.5022	0.8707	0.1676	0.1240	**0.2673**	0.1815	0.4185
Llama-2-7b	0.3852	0.4240	0.3410	0.7917	0.1905	0.0760	0.1783	0.1738	0.3201
Swallow-7b (CT, VE, 日英)	0.4808	**0.5078**	**0.5968**	0.8573	0.1830	0.1240	0.2510	0.1511	0.3940
llm-jp-13b-v1.0	0.2261	0.4790	0.3857	0.7744	0.1082	0.0240	0.1955	0.1185	0.2889
PLaMo-13B	0.2270	0.5189	0.4137	0.7621	0.1025	0.0360	0.1582	0.1196	0.2923
ELYZA-japanese-Llama-2-13 (CT, 日)	0.7399	0.4263	0.4684	0.8718	0.1407	0.0320	0.2198	0.1490	0.3810
Qwen-14B	0.8829	0.4243	0.3220	0.8980	0.1851	**0.3880**	0.2223	0.2224	0.4431
nekomata-14b (CT, 日英)	**0.9169**	**0.5775**	0.6112	**0.9149**	0.2126	0.3560	**0.2985**	**0.2312**	**0.5149**
Llama-2-13b	0.6997	0.4415	0.4170	0.8533	0.2139	0.1320	0.2146	0.1982	0.3963
Swallow-13b (CT, VE, 日英)	0.7837	0.5063	**0.6398**	0.9005	**0.2168**	0.2040	0.2720	0.1771	0.4625
Japanese Stable LM Beta 70B (CT, 日英)	0.9115	0.4925	0.6042	**0.9192**	**0.2573**	0.4160	0.2765	0.2335	0.5138
Llama-2-70b	0.8686	0.4656	0.5256	0.9080	0.2361	0.3560	0.2643	**0.2398**	0.4830
Swallow-70b (CT, VE, 日英)	**0.9348**	**0.6290**	**0.6960**	0.9176	0.2266	**0.4840**	**0.3043**	0.2298	**0.5528**

ら学習された LLM でも Llama 2 よりも高い日本語性能を示すもの（Mistral v0.1, Qwen-7B, Qwen-14B）が存在する．これらの LLM から継続学習されたモデル（japanese-stablelm-base-gamma-7b, nekomata-7b, nekomata-14b）は Swallow よりも高い平均スコアを示すことから，ベースモデルの性能差が反映されたと考えられる．いずれにしても，表 3 の結果から継続事前学習で日本語の能力を強化した LLM を構築できることが明らかになった．

3.3　学習トークンに対するスケール性

図 2 に，継続事前学習の学習データ量（トークン数）と日本語ベンチマークの平均スコアの関係を示した．この実験では，Swallow 7B, 13B,70B の継続事前学習のデータ量を約 20B トークンずつ増やしたときの平均スコアを計測した（学習トークン数が 0B の時はベースモデルである Llama 2 のスコアを表す）．この図から，継続事前学習の日本語のデータ量の増加に伴い，平均スコアが単調に増加する傾向が読み取れる．学習の初期段階である 20B の時の改善幅が最も大きく，その後改善幅が小さくなる傾向があるが，それでも学習データ量の増加に伴い性能は伸び続けていることから，約 100B トークンの継続事前学習でも性能向上が飽和したとは言えない．100B トークン以上で継続事前学習を行った場合の性能については，今後検証したいと考えている．

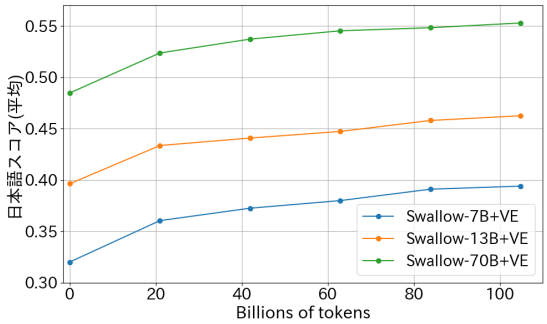

図 2　日本語タスクにおける継続事前学習のスケール性

4　結論と今後の展望

本研究では，Llama 2 に日本語データで継続事前学習を行い，日本語を強化した LLM である Swallow を構築した．評価実験から，継続事前学習の有効性および学習データ量に対するスケール性を示された．実験では，評価データに特化したハックを行わないように細心の注意を払ったが，今後は下流タスクでの性能を高める方策や指示に従う LLM の構築方法も模索したい．また，異なるベースモデルからの継続事前学習を行い，より性能の高いモデルの開発を目指したい．これらの実験を通して，日本語 LLM をフルスクラッチから学習した場合でも高い性能を発揮するための知見を蓄積し，国産 LLM の開発に貢献したいと考えている．

謝辞

国立研究開発法人産業技術総合研究所が構築・運用する AI 橋渡しクラウド（ABCI: AI Bridging Cloud Infrastructure）の「大規模言語モデル構築支援プログラム」の支援を受けた．学習に関して貴重なアドバイスを提供してくださった Sakana AI 秋葉 拓哉氏に感謝する. 学習した LLM の評価実験では, LLM-jp（LLM 勉強会）で開発されているデータや公開されている知見を活用しました. 本研究は, JST, CREST, JPMJCR2112 の支援を受けた.

参考文献

[1] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. **Bioinformatics**, Vol. 36, No. 4, pp. 1234–1240, September 2019.

[2] Iz Beltagy, Kyle Lo, and Arman Cohan. SciBERT: A pretrained language model for scientific text. In **Proceedings of EMNLP-IJCNLP**, pp. 3615–3620, 2019.

[3] Chul Sung, Tejas Dhamecha, Swarnadeep Saha, Tengfei Ma, Vinay Reddy, and Rishi Arora. Pre-training BERT on domain resources for short answer grading. In **Proceedings of EMNLP-IJCNLP**, pp. 6071–6075, 2019.

[4] Kshitij Gupta, Benjamin Thérien, Adam Ibrahim, Mats L Richter, Quentin Anthony, Eugene Belilovsky, Irina Rish, and Timothée Lesort. Continual Pre-Training of Large Language Models: How to (re) warm your model? In **Proceedings of NeurIPS**, 2023.

[5] Yiming Cui, Ziqing Yang, and Xin Yao. Efficient and effective text encoding for Chinese LLaMA and Alpaca. arXiv:2304.08177, 2023.

[6] Ramon Pires, Hugo Abonizio, Thales Sales Almeida, and Rodrigo Nogueira. Sabiá: Portuguese large language models. arXiv:2304.07880, 2023.

[7] Wenhao Zhu, Yunzhe Lv, Qingxiu Dong, Fei Yuan, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. Extrapolating large language models to non-English by aligning languages. arXiv:2308.04948, 2023.

[8] Jun Zhao, Zhihao Zhang, Qi Zhang, Tao Gui, and Xuanjing Huang. Llama beyond english: An empirical study on language capability transfer. arXiv:2401.01055, 2024.

[9] Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. arXiv:2307.08691, 2023.

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv:1412.6980, 2014.

[11] Deepak Narayanan, Mohammad Shoeybi, Jared Casper, Patrick LeGresley, Mostofa Patwary, Vijay Anand Korthikanti, Dmitri Vainbrand, Prethvi Kashinkunti, Julie Bernauer, Bryan Catanzaro, Amar Phanishayee, and Matei Zaharia. Efficient large-scale language model training on GPU clusters using Megatron-LM. In **Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis**, pp. 1–15, 2021.

[12] 岡崎直観, 服部翔, 平井翔太, 飯田大貴, 大井 聖也藤井 一喜, 中村泰士, Mengsay Loem, 横田理央, 水木栄. Swallow コーパス: 日本語大規模ウェブコーパス. 言語処理学会第 30 回年次大会 (NLP2024), 2024.

[13] Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data, and web data only. arXiv:2306.01116, 2023.

[14] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: an 800GB dataset of diverse text for language modeling. arXiv:2101.00027, 2020.

[15] 飯田大貴, 水木栄, 藤井一喜, 中村泰士, Mengsay Loem, 服部翔, 平井翔太, 大井聖也, 横田理央, 岡崎直観. 大規模言語モデルの日本語能力の効率的な強化：継続事前学習における語彙拡張と対訳コーパスの活用. 言語処理学会第 30 回年次大会 (NLP2024), 2024.

[16] Namgi Han, 植田暢大, 大嶽匡俊, 勝又智, 鎌田啓輔, 清丸寛一, 児玉貴志, 菅原朔, Bowen Chen, 松田寛, 宮尾祐介, 村脇有吾, 劉弘毅. llm-jp-eval: 日本語大規模言語モデルの自動評価ツール. 言語処理学会第 30 回年次大会 (NLP2024), 2024.

[17] Leo Gao, Jonathan Tow, Stella Biderman, Charles Lovering, Jason Phang, Anish Thite, Fazz, Niklas Muennighoff, Thomas Wang, sdtblck, tttyuntian, researcher2, Zdeněk Kasner, Khalid Almubarak, Jeffrey Hsu, Pawan Sasanka Ammanamanchi, Dirk Groeneveld, Eric Tang, Charles Foster, kkawamu1, xagi dev, uyhcire, Andy Zou, Ben Wang, Jordan Clive, igor0, Kevin Wang, Nicholas Kross, Fabrizio Milo, and silentv0x. EleutherAI/lm-evaluation-harness: v0.3.0, December 2022.

[18] Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. JGLUE: Japanese general language understanding evaluation. In **Proceedings of LREC**, pp. 2957–2966, 2022.

[19] 石井愛, 井之上直也, 関根聡. 根拠を説明可能な質問応答システムのための日本語マルチホップ QA データセット構築. 言語処理学会第 29 回年次大会 (NLP2023), pp. 2088–2093, 2023.

[20] 関根聡. 百科事典を対象とした質問応答システムの開発. 言語処理学会第 9 回年次大会 (NLP2003), pp. 637–640, 2003.

[21] Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. XL-Sum: Large-scale multilingual abstractive summarization for 44 languages. In **Findings of ACL-IJCNLP**, pp. 4693–4703, 2021.

[22] Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. Language models are multilingual chain-of-thought reasoners. In **Proceedings of ICLR**, 2023.

[23] Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri. Findings of the 2020 Conference on Machine Translation (WMT20). In **Proceedings of WMT**, pp. 1–55, 2020.

表 4　評価したモデルとその配布元 URL

モデル	URL
calm2-7b	https://huggingface.co/cyberagent/calm2-7b
Japanese Stable LM Beta 7B+CT(日英)	https://huggingface.co/stabilityai/japanese-stablelm-base-beta-7b
ELYZA-japanese-Llama-2-7b+CT(日)	https://huggingface.co/elyza/ELYZA-japanese-Llama-2-7b
youri-7b+CT(日英)	https://huggingface.co/rinna/youri-7b
Mistral v0.1 (7B)	https://huggingface.co/mistralai/Mistral-7B-v0.1
japanese-stablelm-base-gamma-7b+CT(日英)	https://huggingface.co/stabilityai/japanese-stablelm-base-gamma-7b
Qwen-7B	https://huggingface.co/Qwen/Qwen-7B
nekomata-7b+CT(日英)	https://huggingface.co/rinna/nekomata-7b
Llama-2-7b	https://huggingface.co/meta-llama/Llama-2-7b-hf
Swallow-7b+VE++CT(日英)	https://huggingface.co/tokyotech-llm/Swallow-7b-hf
llm-jp-13b-v1.0	https://huggingface.co/llm-jp/llm-jp-13b-v1.0
PLaMo-13B	https://huggingface.co/pfnet/plamo-13b
ELYZA-japanese-Llama-2-13+CT(日)	https://huggingface.co/elyza/ELYZA-japanese-Llama-2-13b
Qwen-14B	https://huggingface.co/Qwen/Qwen-14B
nekomata-14b+CT(日英)	https://huggingface.co/rinna/nekomata-14b
Llama-2-13b	https://huggingface.co/meta-llama/Llama-2-13b-hf
Swallow-13b+VE+CT(日英)	https://huggingface.co/tokyotech-llm/Swallow-13b-hf
Japanese Stable LM Beta 70B+CT(日英)	https://huggingface.co/stabilityai/japanese-stablelm-base-beta-70b
Llama-2-70b	https://huggingface.co/meta-llama/Llama-2-70b-hf
Swallow-70b+VE+CT(日英)	https://huggingface.co/tokyotech-llm/Swallow-70b-hf

A　評価したモデルの出典

表 4 に、実験に用いたモデルの名前と配布元 URL の対応を掲載した。