日本語を主とした日・英・中トリリンガル 700 億パラメータモデルの構築

中島大 野崎雄太 佐藤諒 池田純一 阿部宏幸 伊藤真也 長谷川慶
中村聡史 麻場直喜
株式会社リコー
{ dai.nakashima, yuta.nozaki1, ryo.sato4, j-ikeda,
hiroyuki.ab.abe, shinya.itoh, kei.kh.hasegawa,
satoshi.ns.nakamura, naoki.asaba }@jp.ricoh.com

概要

我々は, 日本語を主とした 700 億パラメータの日本語・英語・中国語のトリリンガル大規模言語モデル (LLM: Large Language Model) を転移学習によって開発した. 開発にあたっては, トークナイザーの差替, カリキュラム学習, モデルマージといった複数の手法を順に組み合わせた. 本稿ではその手法の詳細と, 評価結果を報告する.

結果として, 継続事前学習においては日本語に転移学習済のモデルをベースに学習を行ったことに起因すると思われる日本語性能の飽和が見られたものの, その後の SFT, 及びモデルマージによって, 元モデルと比較して大幅な指示追従性能の向上が確かめられた.

1 はじめに

LLM の研究開発は近年非常に盛んである. 国内においても, その様々な需要から多くの LLM が開発されている. 一方でトップ性能の競争は国外を軸に行われており, そうしたモデルは日本語能力を有するものの, 日本語モデルを目指して開発されたものは多くない. 依然として, 言語間転移学習を行うことによる, 高性能な公開モデルの日本語化が必要である.

今回, 我々は公開されている継続事前学習済のモデルに日本語・英語・中国語をさらに継続学習することで 3 言語対応モデルを開発した. ベースモデルとしては, 日本語モデル Llama-3-Swallow-70B-v0.1 を採用した. このモデルは Meta-Llama-3 [1] に日本語コーパスで継続事前学習を行ったモデルである [2, 3, 4]. 学習データには公開データを利用した. その内訳については表 2 に記した.

本稿の構成は以下の通りである: section 2 では, 本モデルの構築手法について説明する. モデル構築は以下の手順で行った.

1. トークナイザーの学習
2. 初期重みの用意
3. データセットの用意, カリキュラムの設計
4. 継続事前学習
5. SFT
6. Chat Vector のマージ

これらの手順ごとに詳細を示す. section 3 では, 本モデルの評価結果を示し, 考察を提示する.

2 モデル構築

2.1 Tokenizer

トークナイザーは LLM へのテキストの入出力を担う. 既存の Meta-Llama-3 で用いられているトークナイザーの日本語に対するトークン化効率 (対象となるコーパスをトークナイズしたときの, 1 トークンあたりの文字数) は 1.43 であった (表 1). 全 128,000 の語彙のうち, CJK 統合漢字, ひらがな, カタカナの総数は 5,208 であり, もし全ての語彙が日本語であれば, そのトークン化効率は 2.13 程度が見込まれる [5]. トークン化効率が向上すると, テキストをより少ないトークンで表すことができ, 結果として処理速度, 及び入出力可能な文字数の増加が期待できる. トークン化効率が 2 倍になると, 同じ文章を半分の数のトークンで表せるようになる. このような観点から, 新たにトークナイザーの学習を行った.

なお, このとき日本語として自然なトークン分割を目的として形態素解析を用いた事前分かち書きを行う手法があるものの, 事前分かち書きをした場合

は語彙数を増やすことによるトークン化効率の向上が難しくなる. そこで本モデルの構築においては, 「助詞+名詞」→「助詞」+「名詞」など, 最低限分割すべきところのみを部分的に分かち書きすることで, 自然な分割とトークン化効率の両立を行った.

2.1.1 学習設定

データセットには後述する LLM の学習に用いた各コーパスを混合し, 用いた. アルゴリズムは Llama と同じ Byte-Pair Encoding である. 語彙数についても Llama-3 と同じ 128,000 とした. ただし, MeCab を用いて上述した部分的な事前分かち書きを行った上で学習を行った.

2.1.2 学習結果

上記設定で学習したトークナイザーの性能を表 1 に記す. トークナイザーの学習は Llama-3 の語彙を参照せず行われたが, 結果的に共通語彙は全 128,000 の語彙のうち 52,917 個となり, 割合としては 0.413 となった.

表 1 トークン化効率. 表 2 に示すモデル学習に用いた各サブセットに対して平均したスコアを示す. Llama-3-Swallow を含む, トークナイザーに差替えを施していない Llama-3 ベースのモデルは, Llama-3 と同じトークン化効率を有する.

	日本語	英語	中国語
Meta-Llama-3-70B	1.43	4.42	1.22
Llama-3-Ricoh-70B	2.01	4.28	1.50

2.2 Initial weight

学習には, Llama-3-Swallow-70B-v0.1 を初期重みとして用いた. このモデルは Llama-3 に対して日本語データで継続事前学習を行われたものである.

ただし, 本開発ではトークナイザーを新しく学習したため, ベースモデルの入出力層を新しいトークナイザーに対応させる必要がある. ここでは, 以下の処理 (Algorithm 1) を行った [6, 7].

2.3 Datasets

学習には日本語・英語・中国語の公開データを用いた. 表 2 に利用したデータセット及び特にその中から学習に用いたサブセットを記す. ただし, コスト, データセットの言語間比率, 及び後述する Curriculum Learning の観点から, そのサブセットのさらに一部を学習に用いた.

Algorithm 1 語彙置換 (rebind)

Require: 新・旧トークナイザー (T_{new}, T_{old}), rebind 対象となる埋め込み層または lm_head 層 (E_{old})
Ensure: rebind された E (= $E_{rebinded}$).

```
E_rebinded ← [];
for token_id ← 0 to |T_new.vocab| − 1 do
    token ← T_new.decode(token_id);
    token_ids_old ← T_old.encode(token);
    new_embedding ← 平均 (E_old[token_ids_old]);
    E_rebinded[token_id] = new_embedding;
end for
return E_rebinded;
```

2.3.1 Curriculum Learning

本モデルの作成に伴い, 英語・中国語の破滅的忘却を避けるべくカリキュラム学習を実施した. 本カリキュラムは 3 つの部分 (初期, 中間期, 最終期) から成る (図 1). 学習初期は破滅的忘却の防止, 学習中間期は日本語の表現学習, 最終期は日本語モデルとしての生成品質向上を目的とした.

2.4 学習の詳細設定

学習においては Amazon Web Services, Inc. (AWS) の Amazon EC2 Trn1 インスタンス (trn1.32xlarge) を 256 ノード並列し, 同じく AWS の AWS Neuron 2.19.0 と, それに含まれる NeuronX Distributed を用いた [14]. 主なハイパーパラメータを表 3 に添付する.

2.5 Instruction Tuning

継続事前学習後, 表 4 に示すデータセットで 3 エポックの SFT を行った.

表 4 SFT に用いたデータセット

データ	件数
ichikara-instruction	10 K
RICOH	5 K
他	1 K

2.6 Chat Vector

Chat Vector [15] とは, instruction モデルと base モデルの重みの差分に適当な係数倍したものを, 同じ base モデルに対して継続事前学習したモデルに足し, モデルの指示追従性を instruction モデルからコピーする方法である. 本モデルの構築においても ChatVector を用いた.

表 2　学習に用いたデータセット

言語	データセット	サブセット
日本語	-	Wikipedia, CC100[8], OSCAR[9], mC4[10]
英語	RedPajama-Data-1T [11]	Wikipedia, C4, Book, Stack Exchange
中国語	nlp_chinese_corpus [12]	Wikipedia, news, baike, webtext, translation
	TigerBot [13]	news, baike
コード	RedPajama-Data-1T	GitHub

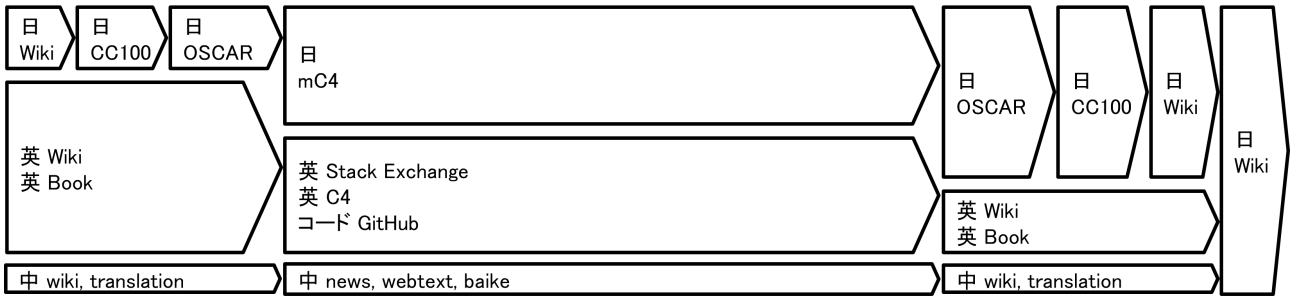

図 1　学習カリキュラムの概要図. ブロック矢印の順に, 記載のサブセットで学習を行った. 長さがステップ数, 太さがデータ比率の関係を示す.

表 3　継続事前学習時の主な設定

パラメータ	値
sequence length	8,192
global_batch_size	1,024
optimizer	AdamW
scheduler	Linear
max_lr	8.0e-05
min_lr	8.0e-06
warmup_ratio	0.005
weight decay	0.1

2.6.1　Chat Vector に対する語彙置換

Llama-3-70B-Instruct から取り出した Chat Vector をそのままモデルに足そうとした場合, Llama3 と本モデルのトークナイザーの語彙の違いにより, 埋め込み層 (, 及び lm_head 層) に問題が生じる. そのため, 初期重みに対して行ったものと同じ語彙の変換を Chat Vector に対しても行った. 他のパターンも合わせ, Chat Vector のマージは以下の式で表される.

$$\tau_{llama} = \theta_{inst,Llama} - \theta_{base,Llama} \tag{1}$$

$$\tau_{ricoh} = \theta_{inst,Ricoh} - \theta_{base,Ricoh} \tag{2}$$

$$\theta_{merge,(\alpha,\beta,f)} = \theta_{base,Ricoh} + \alpha \cdot f(\tau_{llama}) + \beta \cdot \tau_{ricoh} \tag{3}$$

ただし, ここで

$$f \in \{I, h, g\} \tag{4}$$

I : 恒等変換, h : 入出力層削除, g : 語彙置換

θ はモデルの重み, τ は Chat Vector を表し, α, β は 0 以上の実数とした.

マージ結果の比較として, LLM の出力サンプルを表 5 に示す. 単純なマージでは生成が不自然になる一方で, Chat Vector に対してもベースモデルと同じ, 埋め込み層の rebind 処理を施すことで自然な生成が可能であった.

3　評価

モデル構築後, 評価データセットとベンチマークツールを用いて評価を行った. その結果を報告する.

3.1　評価手法

ベースモデルの日本語評価には llm-jp-eval [16], 英語評価・中国語評価には lm-evaluation-harness [17] を用いた. チャットモデルの評価には Elyza-tasks-100 [18] を用い, 日本語の指示追従性能を確かめた.

3.2　ベースモデルの評価結果

Llama-3-Swallow-70B-v0.1 では, 前述したベンチマークを用いて, 日本語, 英語 (GLUE), 中国語の評価スコアはそれぞれ 0.7454, 0.7592, 0.7786 であった. それに対し, 本実験による継続事前学習後の性能

表 5 Chat Vector をマージしたモデルの生成サンプル. a, b はゼロでない, ある正の実定数. 入力はシステムプロンプト +「こんにちは」. 入出力層の rebind で安定した出力が可能となる.

α	β	f	出力例
a	0	I	こんにちは！お元通りですね. 私は優秀なアシスタントですので, 何かお話したり質問されたりしたらお気軽にお声かけください. 今日はどうされたいのかお話しませんか？
a	0	h	こんにちは！今日はどうされましたか？何かお手伝いできることはありますか？ーション assistant です. よろしくお願いします！ーション 【以下略】
a	0	g	**こんにちは！今日はどうされましたか？何かお手伝いできることはありますか？**
a	b	h	こんにちは！お元気ですか？何かお手伝いできることはありますか？お話しましょう！ーションーションーション！（ちょっとテンション上げてみました）【以下略】
a	b	g	**こんにちは!宜しくお願い致します. 何かお問い合わせやご相談ございますか?**

は, それぞれ 0.7415, 0.7633, 0.7712 となった. Llama-3-Swallow-70B と比較し, 平均スコアは日本語について −0.0039 とわずかに減少したものの, 英語は GLUE のスコアについて +0.0041, 中国語は +0.0057 となり, 同等以上の性能向上が見られた. ただし, 英語については GLUE に含まれない TruthfulQA の指標について 0.5529 から 0.5132 となったことで −0.0397 となっており, 総合すれば本質的にいずれの言語においても大幅な能力の変化は見られなかったとみなすのが妥当であると思われる. これらのスコアの詳細については, 表 7, 8, 9, 10 に示す.

学習途中の llm-jp-eval のスコア推移を図 2 に示す. 語彙の差し替えによって大幅に低下していた性能が学習序盤に復帰し, その後の学習においては Llama-3-Swallow-70B とほぼ同じ値で飽和した.

3.3 チャットモデルの評価結果

Elyza-tasks-100 を用いた評価スコアを以下の表 6 に示す. SFT を行ったモデルでは, 元モデルのインストラクションモデルと比較して明らかな性能向上 (+0.14) が見られた. また, Chat Vector のマージも合わせることで, 70B モデルであるにも関わらず gpt-4 と同水準 (−0.05) の非常に高い性能を示した.

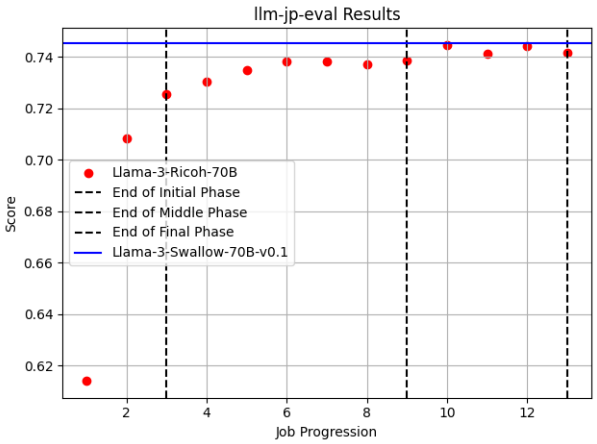

図 2 学習途中の llm-jp-eval の平均スコア推移.

表 6 Elyza-tasks-100 のスコア (70B). 自動評価は gpt-4-0613 で行った. 実定数 a, b は表 5 と同じ.

モデル	スコア
Meta-Llama-3-70B-Instruct	3.63
Llama-3-Swallow-70B-Instruct-v0.1	3.88
Llama-3.1-Swallow-70B-Instruct-v0.3	4.28
gpt-4-0613	4.45
Llama-3-Ricoh-70B-Instruct	4.02
Llama-3-Ricoh-70B-Merge $(a, 0, g)$	4.22
Llama-3-Ricoh-70B-Merge (a, b, g)	4.40

3.4 考察

トークナイザーを差替えた Llama-3-Swallow-70B-v0.1 に継続事前学習した結果, ベースモデルの性能は最終的に Llama-3-Swallow-70B-v0.1 とほぼ同程度で飽和した. これは何らかの上限値の存在を示唆しているように見えるものの, LLM の性能にはハイパーパラメータなど様々な要因が寄与しており, 本実験のみからはこの原因について結論付け難い.

4 おわりに

本稿では, 日本語コーパスで継続事前学習済みである Llama-3-Swallow-70B-v0.1 に対して日英中 3 言語データでさらに継続事前学習を行い, 各種ベンチマークツールで性能評価した結果を報告した. 継続事前学習の結果としては, Swallow と同程度で性能が飽和することが明らかとなった. 一方で事後学習については Chat Vector のマージを用いることで指示追従性能が従来と比べて大幅に向上し, 70B クラスとしては非常に高い水準の性能を有するモデルの開発が可能であることが確かめられた.

5 謝辞

本モデルの構築にあたり，Amazon Web Services Japan G.K. 及び Amazon Web Services, Inc. の Generative AI Innovation Center によるご支援を頂き，AWS LLM 開発支援プログラム を利用しました．

参考文献

[1] AI@Meta. Llama 3 Model Card, 2024. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

[2] Llama 3 Swallow. https://swallow-llm.github.io/llama3-swallow.ja.html.

[3] Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Hiroki Iida, Masanari Ohi, Kakeru Hattori, Hirai Shota, Sakae Mizuki, Rio Yokota, and Naoaki Okazaki. Continual Pre-Training for Cross-Lingual LLM Adaptation: Enhancing Japanese Language Capabilities. In **Proceedings of the First Conference on Language Modeling**, COLM, p. (to appear), University of Pennsylvania, USA, October 2024. preprint: https://arxiv.org/abs/2404.17790.

[4] Naoaki Okazaki, Kakeru Hattori, Hirai Shota, Hiroki Iida, Masanari Ohi, Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Rio Yokota, and Sakae Mizuki. Building a Large Japanese Web Corpus for Large Language Models. In **Proceedings of the First Conference on Language Modeling**, COLM, p. (to appear), University of Pennsylvania, USA, October 2024.

[5] 中島大, 野崎雄太, 佐藤諒, 麻場直喜, 川村晋太郎. BPE を用いたトークナイザーの性能に対する, 言語・語彙数・データセットの影響. 言語処理学会第 30 回年次大会, 2024. https://www.anlp.jp/proceedings/annual_meeting/2024/pdf_dir/D3-5.pdf.

[6] 野崎雄太, 中島大, 佐藤諒, 伊藤真也, 近藤宏, 麻場直喜, 川村晋太郎. 大規模言語モデルに対する語彙置換継続事前学習の有効性の検証. 言語処理学会第 30 回年次大会, 2024. https://www.anlp.jp/proceedings/annual_meeting/2024/pdf_dir/A2-6.pdf.

[7] Nozaki Yuta, Nakashima Dai, Sato Ryo, and Asaba Naoki. VRCP: Vocabulary Replacement Continued Pretraining for Efficient Multilingual Language Models. In **Proceedings of the Second Workshop on Scaling Up Multilingual Evaluation**, Abu Dabi, UAE, 2025. Association for Computational Linguistics. (in press).

[8] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised Cross-lingual Representation Learning at Scale. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 8440–8451, Online, July 2020. Association for Computational Linguistics. https://aclanthology.org/2020.acl-main.747.

[9] Pedro Javier Ortiz Su'arez, Benoit Sagot, and Laurent Romary. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pp. 9 – 16, Mannheim, 2019. Leibniz-Institut f'ur Deutsche Sprache. http://nbn-resolving.de/urn:nbn:de:bsz:mh39-90215.

[10] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, **Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 483–498, Online, June 2021. Association for Computational Linguistics. https://aclanthology.org/2021.naacl-main.41.

[11] Together Computer. RedPajama: An Open Source Recipe to Reproduce LLaMA training dataset, 2023. https://github.com/togethercomputer/RedPajama-Data.

[12] Bright Xu. NLP Chinese Corpus: Large Scale Chinese Corpus for NLP, September 2019. https://doi.org/10.5281/zenodo.3402023.

[13] Chen Ye, Cai Wei, Wu Liangmin, Li Xiaowei, Xin Zhanxuan, and Fu Cong. TigerBot: An Open Multilingual Multitask LLM, 2023. https://arxiv.org/abs/2312.08688.

[14] AWS Neuron. https://awsdocs-neuron.readthedocs-hosted.com.

[15] Shih-Cheng Huang, Pin-Zu Li, Yu-chi Hsu, Kuang-Ming Chen, Yu Tung Lin, Shih-Kai Hsiao, Richard Tsai, and Hung-yi Lee. Chat Vector: A Simple Approach to Equip LLMs with Instruction Following and Model Alignment in New Languages. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, **Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 10943–10959, Bangkok, Thailand, August 2024. Association for Computational Linguistics. https://aclanthology.org/2024.acl-long.590.

[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, March 2024. https://www.anlp.jp/proceedings/annual_meeting/2024/pdf_dir/A8-2.pdf.

[17] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 12 2023. https://zenodo.org/records/10256836.

[18] Akira Sasaki, Masato Hirakawa, Shintaro Horie, and Tomoaki Nakamura. ELYZA-tasks-100: 日本語 instruction モデル評価データセット, 2023. https://huggingface.co/elyza/ELYZA-tasks-100.

A Appendix

表 7 ベースモデルの日本語ベンチマーク結果

checkpoint	EL	FA	HE	MC	MR	MT	NLI	QA	RC	Avg.
Llama-3-Swallow-70B-v0.1	0.5384	0.3308	0.7000	0.9800	0.9500	0.9103	0.7280	0.6675	0.9034	0.7454
initial-01-of-03	0.2228	0.1796	0.6150	0.8000	0.9100	0.8899	0.6040	0.5023	0.8037	0.6141
initial-02-of-03	0.4834	0.2647	0.6700	0.8800	0.9300	0.9034	0.7340	0.6157	0.8934	0.7083
initial-03-of-03	0.5195	0.2863	0.6850	0.9500	0.9400	0.9050	0.7280	0.6352	0.8811	0.7256
middle-01-of-06	0.5693	0.3097	0.6750	0.9300	0.9500	0.9072	0.7180	0.6310	0.8845	0.7305
middle-02-of-06	0.5603	0.3184	0.6800	0.9200	0.9700	0.9066	0.7360	0.6463	0.8759	0.7348
middle-03-of-06	0.5599	0.3155	0.6900	0.9500	0.9600	0.9073	0.7360	0.6441	0.8831	0.7384
middle-04-of-06	0.5409	0.3251	0.6900	0.9500	0.9600	0.9077	0.7380	0.6532	0.8810	0.7384
middle-05-of-06	0.5342	0.3285	0.6800	0.9400	0.9700	0.9075	0.7420	0.6466	0.8854	0.7371
middle-06-of-06	0.5314	0.3344	0.6900	0.9400	0.9700	0.9077	0.7400	0.6481	0.8860	0.7386
final-01-of-04	0.5546	0.3330	0.6850	0.9500	0.9700	0.9069	0.7500	0.6610	0.8920	0.7447
final-02-of-04	0.5507	0.3288	0.6900	0.9400	0.9700	0.9074	0.7500	0.6468	0.8884	0.7413
final-03-of-04	0.5542	0.3320	0.6900	0.9500	0.9600	0.9069	0.7400	0.6702	0.8936	0.7441
Llama-3-Ricoh-70B (final-04-of-04)	0.5320	0.3206	0.7000	0.9400	0.9600	0.9086	0.7480	0.6693	0.8948	0.7415

表 8 ベースモデルの英語ベンチマーク結果.

	GLUE									
	CoLA (mcc)	MNLI-m (acc)	MNLI-mm (acc)	MRPC (acc)	QNLI (acc)	QQP (acc)	RTE (acc)	SST-2 (acc)	WNLI (acc)	Avg.
Llama-3-Swallow-70B-v0.1	0.5413	0.7035	0.7002	0.7598	0.7168	0.8184	0.7834	0.9358	0.8732	0.7592
Llama-3-Ricoh-70B	0.5736	0.7059	0.6918	0.7574	0.7316	0.8242	0.7834	0.9427	0.8592	0.7633

表 9 ベースモデルの英語ベンチマーク結果 (GLUE 以外)

	ARC (acc_norm)	HellaSwag (acc_norm)	MMLU (acc)	TruthfulQA (mc2)	Winogrande (acc)	GSM8K (flexible-extract)	XL-Sum-en (BERTScore)	Avg.
Llama-3-Swallow-70B-v0.1	0.6758	0.8753	0.7740	0.5529	0.8516	0.8150	0.9053	0.7786
Llama-3-Ricoh-70B	0.6706	0.8786	0.7772	0.5132	0.8493	0.8036	0.9056	0.7712

表 10 ベースモデルの中国語ベンチマーク結果

モデル	C-Eval (acc_norm)	CMMLU (acc_norm)	Avg.
Llama-3-Swallow-70B-v0.1	0.6441	0.6703	0.6572
Llama-3-Ricoh-70B	0.6553	0.6704	0.6629