模倣学習による大規模言語モデルの指示チューニング

Youmi Ma[1] 水木 栄[1,2] 藤井 一喜[1,2] 中村 泰士[1,2]
大井 聖也[1,2] 島田 比奈理[1] 塩谷 泰平[1] 齋藤 幸史郎[1] 前田 航希[1]
服部 翔[1,2] 岡本 拓己[1] 石田 茂樹[1] 横田 理央[1,2,3] 高村 大也[2] 岡崎 直観[1,2,3]
[1] 東京科学大学 [2] 産業技術総合研究所 [3] NII LLMC
{ma.y@, okazaki@, swallow@nlp.}comp.isct.ac.jp

概要

指示チューニングを効率的に行う手段として，高性能な大規模言語モデル（LLM）の挙動を模倣する手法が注目を集めている．既存研究では GPT-4 を模倣した学習が主流であるが，ライセンスの制限が厳しいうえ，汎用的な知見が得られにくい．本稿では複数のオープンな LLM を模倣先とし，模倣学習の有効性を検証する．実験結果により，Llama-3.1-Swallow-8B-v0.1 に Gemma-2-27B-IT の模倣学習をさせることで，13B 以下のモデルの中でトップクラスの性能を達成した．また，性能は高いものの模倣学習の効果が限定的な LLM や，模倣学習で習得しにくい能力の存在を明らかにした．

1 はじめに

事前学習された大規模言語モデル（Large Language Model; LLM）を指示と応答の対で追加学習し，様々な指示に応答できるようにすることを指示チューニングと呼ぶ [1]．指示チューニングは汎用的な能力を示す LLM の構築に欠かせないが，どのような指示や応答で学習すると指示チューニングの効果が高まるのか自明でないうえ，応答作成に要するスキルが高いため，人手で応答を作成するには多大な時間と労力が必要になる．そこで，LLM を活用して指示と応答の対 [2, 3, 4, 5, 6]，もしくは応答のみを自動生成し，学習データの構築コストを低減する手法が提案された [7, 8]．このような合成データで指示チューニングをすることは，生成元のモデル（教師モデル）の挙動を別のモデル（生徒モデル）に模倣させているため，**模倣学習**と呼ばれる [9]．

既存研究では，最高性能を示す OpenAI 社の LLM を教師モデルとしているが，これには二つの懸念点がある．まず，このモデルはプロプラエタリであるため，ライセンスの制約が厳しく，構築した合成データやモデルの商用利用が難しい．また，単一の教師モデルに頼ってしまうと汎用的な知見が得られにくく，模倣先の違いによる模倣学習への影響を分析できない．最近ではオープンな LLM の進歩が著しく，プロプラエタリな LLM との性能差が縮小してきたことから[1)]，オープンな LLM を模倣先とすることについて，研究および応用の両面において関心が高まりつつある．

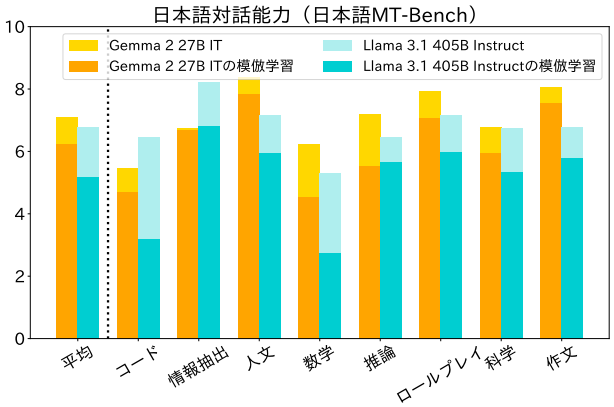


図 1: 日本語 MT-Bench [11] で評価した教師モデルと模倣学習後の生徒モデルの日本語対話能力．生徒モデルは Llama-3.1-Swallow-8B-v0.1 である．

本研究では，オープンな LLM による模倣学習の有効性を調べる．日本語の LLM 開発において指示チューニングデータが圧倒的に不足していることから，対象言語を日本語とする．人間と LLM の実対話を収集したデータセットである LMSYS-Chat-1M [10] の指示を和訳し，複数の応答を教師モデルに生成させる．さらに，生成された応答を LLM で自動採点し，厳選する．

オープンな LLM の中で，トップクラスの対話性能を示す Gemma-2-27B-IT [12] と Llama-3.1-405B-Instruct [13] で応答を合成した．合成したデータを

1) https://lmarena.ai/?leaderboard

Llama 3.1-Swallow-8B-v0.1 [14, 15, 16] の指示チューニングに用いた結果，Gemma-2 の模倣学習はパラメータ数が 13B 以下のモデルの中でトップクラスの性能を達成した．ところが，図 1 に示すように，Llama-3.1 の模倣学習は Gemma-2 ほど有効ではなかった．これは，Gemma-2 が得意とする能力が模倣学習により継承できたのに対し，Llama-3.1 が得意とする能力はあまり継承できなかったためと考えられる．分析の結果，模倣学習により習得できる能力には限界があり，知識や問題解決力を含めてモデルの能力を強化するためには，別の手法が必要であることが示唆された．本研究で合成したデータセットは公開済み[2]で，指示チューニングされたモデルの最新版は Llama-3.1-Swallow-8B/70B-Instruct-v0.3[3] として公開している．

2 手法

本研究では，教師モデルを用いてデータセットを合成し，生徒モデルを指示チューニングする．本稿では，教師モデルを $\hat{\pi}$，生徒モデルを π で表す．

2.1 データセットの合成

データセットの合成は LMSYS-Chat-1M を起点とする．LMSYS-Chat-1M は Chatbot Arena [17] などのプラットフォームから人間と LLM の対話履歴を計 100 万件収集したデータセットである．本稿では，LMSYS-Chat-1M の指示を一部和訳し，模倣先の LLM に応答を生成させる．なお，LMSYS-Chat-1M にはマルチターンの対話が含まれるが，本研究では 1 ターン目の指示のみを合成に用いる．1 ターン目の指示の和訳からなる集合 $\mathcal{I}$ を得てから，それぞれの指示 $I_k \in \mathcal{I}$ に対し，日本語応答 R_k を生成する．

指示は DeepL[4]で機械翻訳した．安全性に懸念がある指示は翻訳対象から外し，残った 732,392 件を和訳した．その後，重複または空の指示を除去し，残った 453,889 件を応答生成に用いた．

次に，教師モデルを用いて指示に対する応答を生成した．このとき，既存研究に倣い，**棄却サンプリング**を実施した [13]．具体的には，和訳した指示 $I_k \in \mathcal{I} (1 \leq k \leq N)$ につき，最大 n 件（$n = 6$）の応答 $R_{k,1}, R_{k,2}, \ldots, R_{k,n}$ を生成した後，指示 I_k と対に

2） https://huggingface.co/datasets/tokyotech-llm/lmsys-chat-1m-synth
3） https://swallow-llm.github.io/llama3.1-swallow.ja.html
4） https://www.deepl.com/en/translator

表 1: 合成したデータセットと既存データセットの比較．Gemma-2 は Gemma-2-27B-IT，Llama-3.1 は Llama-3.1-405B-Instruct である．

	教師モデル	言語	レコード件数
Alpaca [3]	GPT-3	英	52,002
Baize [18]	GPT-3.5	英	210,311
UltraChat [4]	GPT-{3.5,4}	英, 中	1,468,352
Evol-Instruct [5]	GPT-3.5	英	70,000
Ja-Self-Instruct [6]	GPT-4	日	52,002
LMSYS-Chat-Synth	Gemma-2	日	451,450
LMSYS-Chat-Synth	Llama-3.1	日	453,802

したうえで，教師モデルに自動採点させた．採点結果に基づき，点数が最も高かった応答 $\hat{R}_k$ のみ残し，そのほかは棄却した（式 1）．なお，同点の場合はランダムに応答を一つ選んだ．

$$\hat{R}_k := R_{k,\hat{i}}, \quad \hat{i} = \arg\max_{i \in \{1,\ldots,n\}} f_{\hat{\pi}}(I_k, R_{k,i}). \tag{1}$$

ここで $f_{\hat{\pi}}(\cdot) \mapsto [1, 10]$ は教師モデルによる自動採点結果を表す．自動採点に用いるプロンプトの詳細を付録 A に示す．また，棄却サンプリングの有効性を検証する実験の詳細を付録 D で報告する．

このようにして，教師モデル $\hat{\pi}$ 毎にデータセット $\mathcal{D}_{\hat{\pi}} = \{(I_1, \hat{R}_1), \ldots, (I_N, \hat{R}_N)\}$ を合成した．表 1 に，既存の合成データセットと本研究の合成データ LMSYS-Chat-Synth の統計情報を示した．

2.2 指示チューニング

LMSYS-Chat-Synth を用いて，生徒モデルに指示チューニングを施す．ここでは，教師ありファインチューニング (Supervised Fine-Tuning; SFT) を行う．具体的には，以下の損失関数 $\mathcal{L}$ を最小化するように，モデル π のすべてのパラメータを更新する．

$$\mathcal{L}_{\hat{\pi}} = -\sum_{(I_k, \hat{R}_k) \in \mathcal{D}_{\hat{\pi}}} \log \pi(\hat{R}_k | I_k). \tag{2}$$

ここで $\pi(y|x)$ は文字列 x が与えられたとき，生徒モデル π が文字列 y に対して計算した尤度である．

3 実験

3.1 実験設定

教師モデル オープンな LLM のうち，トップレベルの性能を有する Gemma-2-27B-IT と Llama-3.1-405B-Instruct を採用した．Gemma-2-27B-IT の推論は，vLLM [19] を用いて一枚の NVIDIA H100 SXM5 で行った．Llama-3.1-405B-Instruct の推論は，DeepInfra 社提供の API（FP8 量子化版）を用いた．

表 2: JMT-Bench の全カテゴリ平均スコア．Gemma-2 は Gemma-2-27B-IT，Llama-3.1 は Llama-3.1-405B-Instruct である．実験対象となるベースモデルの模倣先を記載する．Llama-3.1-Swallow-8B-Instruct-v0.1 は LMSYS-Chat-Synth-Llama，Llama-3.1-Swallow-8B/70b-Instruct-v0.3 は LMSYS-Chat-Synth-Gemma を包含したデータでチューニングされている．

モデル名	模倣先	スコア
GPT-4o (gpt-4o-2024-05-13)	–	**7.79**
GPT-4 (gpt-4-0613)	–	7.53
Gemma-2-27B-IT	–	7.10
Llama-3.1-405B-Instruct (FP8)	–	6.78
GPT-3.5 (gpt-3.5-turbo-0125)	–	6.66
Llama-3.1-8B	Gemma-2	**5.64**
Llama-3.1-8B	Llama-3.1	5.05
Llama-3.1-8B-Instruct	–	4.65
llm-jp-3-13b	Gemma-2	**5.38**
llm-jp-3-13b	Llama-3.1	4.74
llm-jp-3-13b-instruct	–	5.18
Llama-3.1-Swallow-8B-v0.1	Gemma-2	**6.25**
Llama-3.1-Swallow-8B-v0.1	Llama-3.1	5.18
Llama-3.1-Swallow-8B-Instruct-v0.1	–	5.33
Llama-3.1-Swallow-8B-Instruct-v0.3	–	6.42
Llama-3.1-Swallow-70B-Instruct-v0.3	–	7.12

ただし，Llama-3.1-405B-Instruct の生成結果の自動採点は，同社の Llama-3.1-70B-Instruct API で行った[5]．

生徒モデル　特定のモデルに依存した分析にならないように，生徒モデルとして Llama-3.1-8B [13]，llm-jp-3-13b [20]，Llama-3.1-Swallow-8B-v0.1 [16] の三つを用いた．それぞれ，英語を主要言語とした事前学習モデル，日本語を中心に事前学習したモデル，英語を中心に事前学習したモデルを日本語で継続事前学習したモデルである．

学習と評価　指示チューニング後の生徒モデルの性能を MT-Bench の日本語版（以降 JMT-Bench と呼ぶ）で評価した．JMT-Bench は 80 件の高品質なマルチターン対話データからなるベンチマークであり，8 つのカテゴリを含む．評価は LLM-as-a-Judge による 10 段階自動採点で行われ，審判として gpt-4-1106-preview[6] を利用した．同じ指示に対して応答を 5 回生成させ，その自動採点結果の平均点を最終スコアとした．学習の詳細を付録 B に示す．

3.2　実験結果

指示チューニング済みの生徒モデルを JMT-Bench で評価した結果を表 2 に示す．参考のため，教師モデル，生徒モデルの公開済み指示チューニング版，及び GPT-3.5 と GPT-4 [21] の性能を併記した．より多くのモデルを網羅した評価結果は，Swallow LLM の評価ウェブサイト[7] を参照されたい．

表 2 から，**オープンな LLM を模倣した学習は有効である**ことが確認できた．Gemma-2-27B-IT の模倣学習を施すことで，すべての生徒モデルは開発元から公開されている指示チューニング版を上回る性能を達成した．特に，Llama-3.1-Swallow-8B-v0.1 の Gemma-2-27B-IT 模倣学習版は，高い JMT-Bench のスコア（6.25）を示した．このスコアは，パラメータ数が 13B 以下のモデルの中で Gemma-2-9B-IT の 6.75 に続いて二番手であり，トップクラスである．なお，LMSYS-Chat-Synth から 2 ターン目の指示と応答を追加で生成し，さらに Magpie [22] という手法に基づいて合成したデータセット[8] を学習に併用したのが Llama-3.1-Swallow-8B/70B-Instruct-v0.3 である．Llama-3.1-Swallow-8B-Instruct-v0.3 は GPT-3.5 に，Llama-3.1-Swallow-70B-Instruct-v0.3 は GPT-4 に近い性能を達成できたことは，Gemma-2-27B-IT を模倣する手法の有効性の高さを改めて示している．

また，**Gemma の模倣学習は Llama より有効である**ことが確認できた．Gemma-2-27B-IT と Llama-3.1-405B-Instruct のスコアの差は僅か 0.32 であるが，それらを模倣した生徒モデルのスコアの差は大きい．具体的には，Llama-3.1-8B では 0.59，llm-jp-13b-v3 では 0.64，Llama-3.1-Swallow-8B-v0.1 では 1.07 と差が付いている．これにより，高性能なモデルを模倣しても，対話能力がその性能に比例して強化されるとは限らないことが分かる．また，模倣学習により対話能力が継承されやすいモデルと，そうではないモデルの存在を示唆していると考えられる．§4 では，模倣学習の効果に差が生じる原因を探求する．

3.3　GPT-4 を模倣した学習との比較

ここでは，プロプラエタリな LLM を模倣した指示チューニングを行い，得られたモデルの性能をオープンな LLM を模倣したモデルと比較する．

実験は指示と応答の対を GPT-4 で生成した日本語指示チューニングデータセットである Ja-Self-Instruct [6] に基づいて行った．§ 3.2 と同じように，指示を揃えて，応答を異なる LLM から生成した．具体的には，Ja-Self-Instruct の指示毎に，オープン

5）　70B モデルと 405B モデルによる自動採点の品質に大差はないことを事前に検証した．
6）　既存のリーダーボードでは他の審判を採用していることがあるため，その場合は本稿の報告値と直接比較できない．
7）　https://swallow-llm.github.io/evaluation/
8）　https://huggingface.co/datasets/tokyotech-llm/swallow-magpie-ultra-v0.1

表 3: JMT-Bench の全カテゴリ平均スコア．模倣学習には Ja-Self-Instruct を用いた．

ベースモデル	模倣先	スコア
Llama-3.1-8B	gpt-4-0613	3.85
Llama-3.1-8B	Gemma-2-27B-IT	**5.16**
llm-jp-3-13b	gpt-4-0613	**3.71**
llm-jp-3-13b	Gemma-2-27B-IT	3.37
Llama-3.1-Swallow-8B-v0.1	gpt-4-0613	4.53
Llama-3.1-Swallow-8B-v0.1	Gemma-2-27B-IT	**5.74**

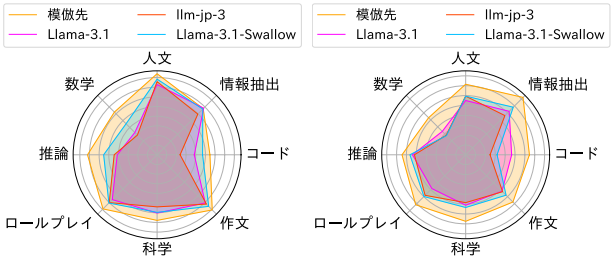
な LLM を用いて応答を生成し，元の指示と対にしたデータセットを合成した．模倣先の LLM として，模倣学習の有効性が高かった Gemma-2-27B-IT を採用した．新たに合成したデータセット（Gemma-2 の模倣学習）と Ja-Self-Instruct（GPT-4 の模倣学習）で指示チューニングを行った結果を表 3 に示す．

実験結果から，Llama-3.1-8B および Llama-3.1-Swallow-8B-v0.1 では，GPT-4 を模倣するよりも Gemma-2 を模倣する方がスコアの伸びが大きかった．llm-jp-3-13b では，GPT-4 の模倣学習の性能が Gemma-2 の模倣学習を上回ったが，僅差であった．これにより，オープンな LLM に対する模倣学習でも，プロプラエタリな LLM に対する模倣学習と同水準，もしくはそれ以上の性能を達成できることが示された．さらに表 2 の結果と比較すると，Gemma-2 を模倣先とした場合，Ja-Self-Instruct よりも LMSYS-Chat-Synth を指示チューニングに用いた方が，生徒モデルの性能が高かった．学習事例数（表 1）も含めて，LMSYS-Chat-Synth の指示集合は Ja-Self-Instruct のものより有効であると考えられる．

4 分析

模倣学習により教師モデルの対話能力がどのように継承されたのかを精査する．教師モデルと模倣学習後の生徒モデルの性能を JMT-Bench のカテゴリ毎に比較し，可視化した結果を図 2 に示す．スペースの都合上，モデル名を一部省略した．データセットとして LMSYS-Chat-Synth を用いるが，Ja-Self-Instruct を用いた場合の結果を付録 C に示す．

図 2a と 2b を見比べると，Gemma-2 の模倣学習はカテゴリ別でも総じて対話能力がよく継承されていることが分かる．特に Gemma-2-27B-IT が得意とする人文と作文の能力は，どの生徒モデルにも順調に継承された．一方，Llama-3.1-405B-Instruct が得意とする情報抽出とコードの能力はうまく継承されなかった．JMT-Bench の平均に差が生じたのは，このためであると考えられる．



(a) Gemma-2-27B-IT　(b) Llama-3.1-405B-Instruct

図 2: 教師モデルと模倣学習後の生徒モデルの JMT-Bench カテゴリ別の性能

また，どの教師モデルを用いても，数学，推論とコードの対話能力が継承されにくく，特に数学能力の継承が最も困難であった．これは，強い LLM が生成した応答を模倣した学習を行っても，生徒の論理的な思考力は改善しにくいことを示唆する．既存研究でも，模倣学習は表層的なスタイルの模倣でしかなく，真にモデルの問題解決力を強化するものではないと指摘しており [9]，本研究で得られた知見と整合している．

人文カテゴリでは，どの教師モデルを用いても，Llama-3.1-Swallow-8B-v0.1 は Llama-3.1-8B より高い性能を示した．前者は後者の日本語知識強化版であり，ベースモデルの地頭の良さは模倣学習を通しても打ち消されずに現れた．模倣学習でモデルの問題解決力を高めることは難しいという知見も踏まえ，知識や問題解決力も含めてモデルの対話能力を真に向上させるためには，ベースモデルの改善や，別のチューニング手法が必要であろう．

5 おわりに

本稿では，二つのオープンかつ高性能な LLM を教師として用い，模倣学習に基づいた LLM の指示チューニングの有効性を検証した．具体的には，Gemma-2-27B-IT と Llama-3.1-405B-Instruct を用いてデータセットを合成し，三つの LLM で指示チューニングを行った．実験から，Gemma-2-27B-IT を模倣することで，Llama-3.1-Swallow-8B-v0.1 の対話性能を同規模の LLM のトップレベルまで強化できた．また，性能は高いが模倣先としての有効性は限定的なモデルが存在する，数学やコードなど模倣学習で強化しにくい能力がある等の知見が得られた．

今後の課題として，日本語以外の言語での実験や，知識や問題解決力を含めたモデルの対話能力をさらに向上させる手法の探求が挙げられる．

謝辞

本研究は，産総研政策予算プロジェクト「フィジカル領域の生成 AI 基盤モデルに関する研究開発」，国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の「次世代人工知能・ロボットの中核となるインテグレート技術開発」プロジェクト（JPNP18002）の「熟練者観点に基づき，設計リスク評価業務における判断支援を行う人工知能適用技術の開発」，文部科学省補助事業「生成 AI モデルの透明性・信頼性の確保に向けた研究開発拠点形成」，その他の支援によって実施されました．本研究は，ABCI の大規模生成 AI 研究開発支援プログラム，および東京科学大学のスーパーコンピュータ TSUBAME4.0 を利用して実施した．

参考文献

[1] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In **Tenth International Conference on Learning Representations (ICLR)**, 2022.

[2] 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. In **61st Annual Meeting of the Association for Computational Linguistics (ACL)**, 2023.

[3] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford Alpaca: An instruction-following LLaMA model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[4] Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In **2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, 2023.

[5] Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. WizardLM: Empowering large pre-trained language models to follow complex instructions. In **Twelfth International Conference on Learning Representations (ICLR)**, 2024.

[6] Yikun Sun, Zhen Wan, Nobuhiro Ueda, Sakiko Yahata, Fei Cheng, Chenhui Chu, and Sadao Kurohashi. Rapidly developing high-quality instruction data and evaluation benchmark for large language models with minimal human effort: A case study on japanese. In **2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)**, 2024.

[7] Wei-Lin Chiang, et al. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality. https://lmsys.org/blog/2023-03-30-vicuna/, 2023.

[8] Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. Wildchat: 1m chatGPT interaction logs in the wild. In **Twelfth International Conference on Learning Representations (ICLR)**, 2024.

[9] Arnav Gudibande, Eric Wallace, Charlie Victor Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. The false promise of imitating proprietary language models. In **Twelfth International Conference on Learning Representations (ICLR)**, 2024.

[10] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric Xing, Joseph E. Gonzalez, Ion Stoica, and Hao Zhang. LMSYS-chat-1m: A large-scale real-world LLM conversation dataset. In **Twelfth International Conference on Learning Representations (ICLR)**, 2024.

[11] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In **Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track (NeurIPS)**, 2023.

[12] Gemma Team, et al. Gemma 2: Improving open language models at a practical size. arXiv:2408.00118, 2024.

[13] Aaron Grattafiori, et al. The Llama 3 herd of models. arXiv:2407.21783, 2024.

[14] 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 **First Conference on Language Modeling (COLM)**, 2024.

[15] 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 **First Conference on Language Modeling (COLM)**, 2024.

[16] 服部翔, 岡崎直観, 水木栄, 藤井一喜, 中村泰士, 大井聖也, 塩谷泰平, 齋藤幸史郎, Youmi Ma, 前田航希, 岡本拓己, 石田茂樹, 横田理央, 高村大也. Swallow コーパス v2: 教育的な日本語ウェブコーパスの構築. 言語処理学会第 31 回年次大会 (NLP2025), 2025.

[17] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios N. Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael I. Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: an open platform for evaluating LLMs by human preference. In **41st International Conference on Machine Learning (ICML)**, 2024.

[18] Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. In **2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, 2023.

[19] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In **ACM SIGOPS 29th Symposium on Operating Systems Principles**, 2023.

[20] Akiko Aizawa, et al. LLM-jp: A cross-organizational project for the research and development of fully open Japanese LLMs. arXiv:2407.03963, 2024.

[21] OpenAI, et al. GPT-4 technical report. ArXiv:2306.02707, 2024.

[22] 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:2406.08464, 2024.

[23] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. ZeRO: memory optimizations toward training trillion parameter models. In **International Conference for High Performance Computing, Networking, Storage and Analysis**, 2020.

[24] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In **Ninth International Conference on Learning Representations (ICLR)**, 2019.

[Instruction]
Please act as an impartial judge and evaluate the quality of the response provided by an AI assistant to the user question displayed below. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of the response. However, if the prompt asks for a simple "yes" or "no" answer, do not deduct points for lacking depth, creativity, or detail. Instead, deduct scores if the assistant provides more information than needed when a simple response was requested. Begin your evaluation by providing a short explanation. Be as objective as possible. The assistant is expected to respond in Japanese. Responses in languages other than Japanese will incur score deductions, unless the response language is specifically requested in the question. Also, deduct scores for responses in Japanese but with repetition, low fluency, or responses that alternate between multiple languages. Failure to use Japanese at all will result in the lowest score. However, using Japanese is not mandatory when: (1) providing only Python scripts or calculation results or (2) the language to use in the response is otherwise-specified. Additionally, your explanation of judgement should be in Japanese. After providing your explanation, you must rate the response on a scale of 1 to 10 by strictly following this format: "[[rating]]", for example: "Rating: [[5]]".

[Question]
{question}

[The Start of Assistant's Answer]
{answer}
[The End of Assistant's Answer]

図 3: 棄却サンプリングに用いた採点プロンプト

A　自動採点プロンプト

棄却サンプリングを行うため，生成した応答を採点する必要がある．ここでは，自動採点に用いるプロンプトの詳細を図 3 に示す．

本稿で用いる自動採点プロンプトは Nejumi リーダボードの JMT-Bench 評価に用いるもの[9]に基づいている．また，冗長性ペナルティや日本語の流暢性ペナルティを図の赤色の箇所の通りに追加した．

B　学習の詳細

学習スクリプトの実装は llm-jp により公開されたレポジトリを参考した[10]．学習は全て 4 枚の NVIDIA H100 SXM5 で行い，DeepSpeed ZeRO [23] を用いて 24 時間以内で終了した．

ハイパーパラメータの調整は，Llama-3.1-405B-Instruct を教師モデル，Llama-3.1-Swallow-8B-v0.1 を生徒モデルとした設定で探索し，残りはそれを流用した．探索した結果，バッチサイズを 512，エポック数を 2 にした．

オプティマイザとして AdamW [24] を用い，β_1 を 0.9, β_2 を 0.95 とした．

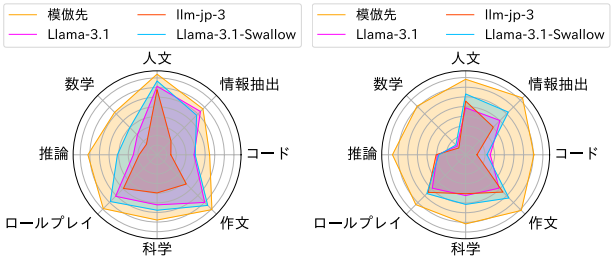
学習率スケジューラとして，コサイン波形による減衰（cosine learning rate scheduler）を用いた．学習の最初の 10%では線形にウォームアップを行い，最大学習率である 2.5e-5 に到達した後，コサイン波形による減衰を適用し，学習の最後には最小学習率である 2.5e-6 に到達するように調整した．

C　分析：Ja-Self-Instruct

Ja-Self-Instruct に基づた模倣学習を行った後，カテゴリ別の性能を比較した結果を図 4 に示す．

9）https://wandb.ai/wandb-japan/llm-leaderboard/reports/Nejumi-LLM-Neo--Vmlldzo2MTkyMTU0

10）https://github.com/llm-jp/llm-jp-sft



(a) Gemma-2-27B-IT　(b) gpt-4-0613

図 4: 教師モデルと模倣学習後の生徒モデルの JMT-Bench カテゴリ別の性能

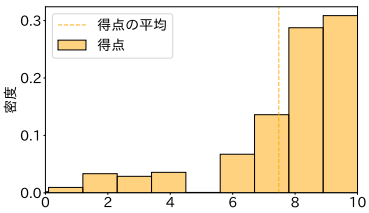


図 5: Gemma-2-27B-IT が生成した応答の得点分布

図 4a と図 2a を見比べると，特に Gemma-2 の模倣学習において全カテゴリの能力継承度合いが悪く，LMSYS-Chat-Synth の優位性が分かる．また，図 4a で Gemma が得意とする人文と作文の能力はうまく継承でき，図 4b で GPT-4 の数学，推論，コードの能力が継承できなかったことから，§ 4 で得られた知見を再度確認できた．最後に，GPT-4 を模倣した学習の効果は全カテゴリにおいて低かったことが判明した．データセットの品質による影響もあるが，プロプラエタリな LLM を模倣することは必ずしも最良戦略とは限らないことが示唆された．

D　棄却サンプリングの有効性検証

式 1 に示すように，本稿では同じ指示に対して複数の応答を生成し，最高得点の応答のみ選出した．ここでは，応答の厳選の有効性を検証するため，(1) ランダムに応答を選出する，(2) 最低得点の応答を選出する，の二つの戦略を追加で検証する．

得られたデータセットで指示チューニングを行い評価した結果，(1) では 6.23，(2) では 6.02 のスコアが得られ，ランダムに選ばれた応答の模倣と最高得点の応答の模倣の効果が同程度だった．また，最低得点の応答を模倣すると，乱択の場合より性能が若干低下した．

既存研究では棄却サンプリングがデータの品質を向上する手段として有用であると報告されている [13]．しかし，本稿ではその効果は予想より薄かったため，自動採点結果の分布を調べた．図 5 のように，得点が 6 以上に集中していることが判明した．したがって，棄却サンプリングの効果が薄いのは，生成した応答は品質が高く，乱択でも十分に高品質な応答が得られているためであると考えられる．