

日本語の作文および 対話性能向上のための 合成データ生成と大規模言語 モデル「Tanuki」の構築

東京科学大学 物質理工学院 材料系

畠山 歓

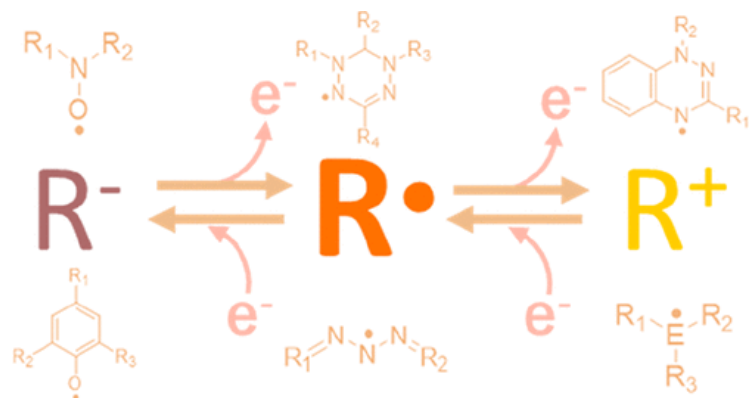
大規模言語モデルの学習にかかる成果は、NEDO(国立研究開発法人新エネルギー・産業技術総合開発機構)の助成事業「ポスト5G情報通信システム
基盤強化研究開発事業」(JPNP20017)の結果得られたものです。
開発に関わったすべての皆様に感謝申し上げます。

アウトライン

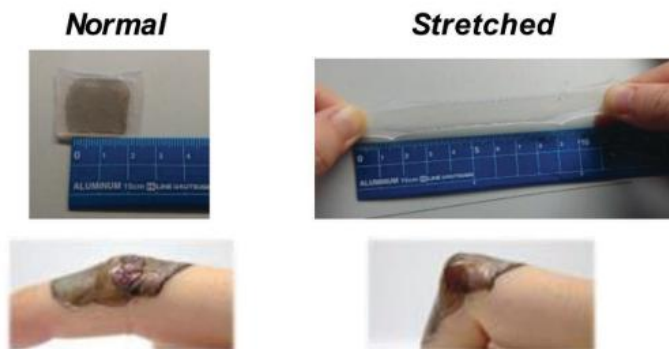
- 自己紹介
- モデル開発の指針と特色
 - 合成データの生成と活用
 - 対話試験によるモデル評価

自己紹介

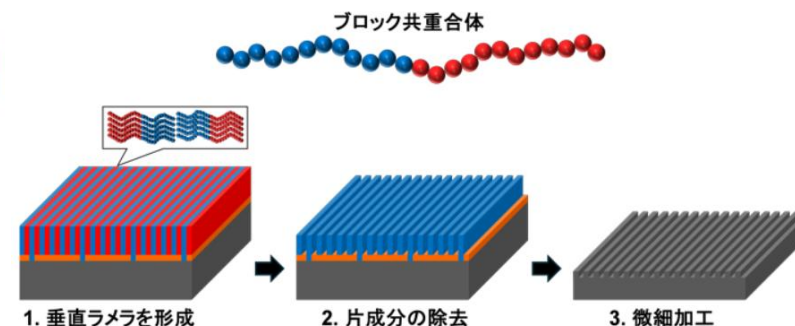
- ・ 高分子合成の研究室で活動中(13年目)
- ・ インフォマティクスの活用研究は7年目



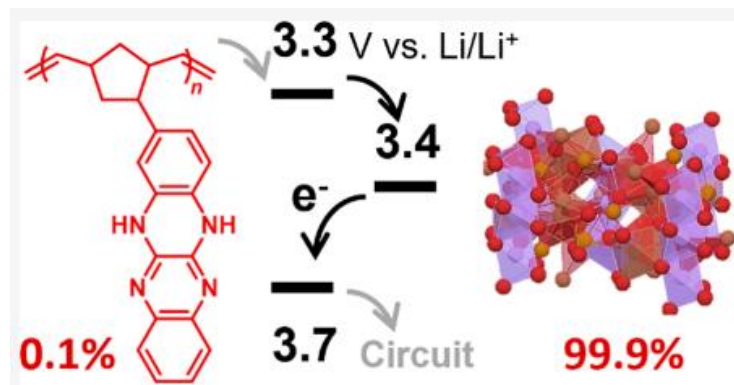
有機材料の電気化学
(Chem. Rev. 2023, 主著)



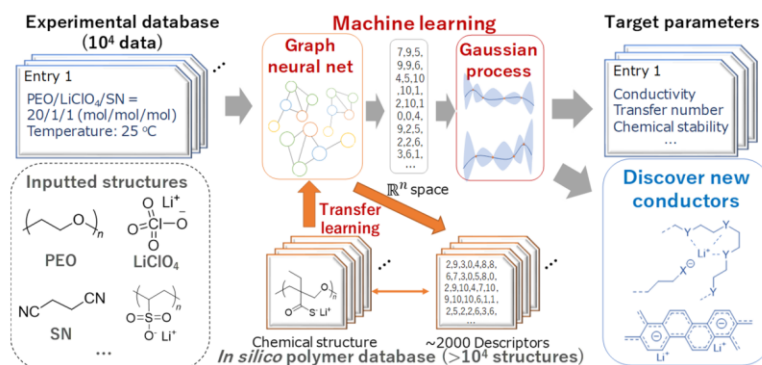
ゴムのように伸びる電池
(Small 2019, 主著)



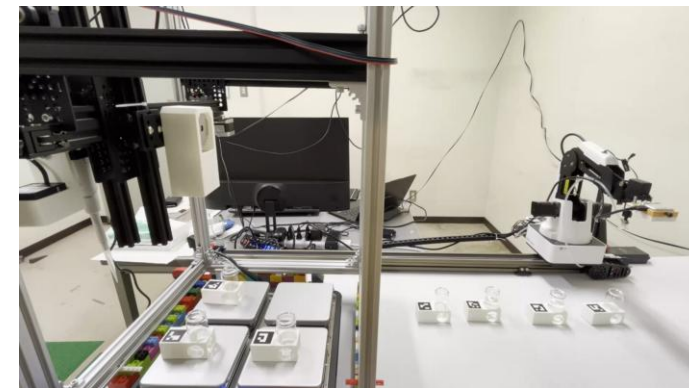
半導体エッチング材料
(Nat. Commun. 2024, 共著)



リチウムイオン電池材料
(ACS Energy Lett. 2020, 主著)



機械学習を使った材料探索
(J. Am. Chem. Soc. 2020, 主著)



ロボット実験
(Polymer J. 2024, 主著)

2024年前半の LLM開発の潮流(?)

LLM開発の流儀 ≡ 教育論？

- ○○をやったら上手になりました、というエピソード型の研究報告が目立つ
- しかし学習法や学習対象がバラバラなので...

LLM開発は黒魔術の世界...？

- ○○のすごい子育て
- ○○を引き出す○○戦略
- ○○思考法
- ○○は○○が9割
- ○○が使っている学力アップ方法
- 超効率○○の絶対やるべき勉強法

最近は合成データ(distill)ブーム

Models 5,276 Full-text search ↑↓ Sort: Trending

deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B Text Generation • Updated 15 days ago • 1.09M • 917	NexaAIDev/DeepSeek-R1-Distill-Qwen-1.5B-NexaQuant Updated 4 days ago • 4.86k • 80
NexaAIDev/DeepSeek-R1-Distill-Llama-8B-NexaQuant Updated 4 days ago • 4.4k • 80	deepseek-ai/DeepSeek-R1-Distill-Qwen-32B Text Generation • Updated 15 days ago • 1.04M • 1.16k
perplexity-ai/r1-1776-distill-llama-70b Updated 3 days ago • 425 • 39	deepseek-ai/DeepSeek-R1-Distill-Llama-8B Text Generation • Updated 15 days ago • 943k • 578
deepseek-ai/DeepSeek-R1-Distill-Qwen-7B Text Generation • Updated 15 days ago • 874k • 480	deepseek-ai/DeepSeek-R1-Distill-Qwen-14B Text Generation • Updated 15 days ago • 586k • 419
deepseek-ai/DeepSeek-R1-Distill-Llama-70B Text Generation • Updated 15 days ago • 472k • 581	bartowski/DeepSeek-R1-Distill-Qwen-32B-GGUF Text Generation • Updated Jan 23 • 2.25M • 237
huihui-ai/DeepSeek-R1-Distill-Qwen-32B-abliterated Text Generation • Updated 8 days ago • 5.66k • 179	bartowski/DeepSeek-R1-Distill-Qwen-32B-abliterated-... Text Generation • Updated 29 days ago • 389k • 92
huihui-ai/DeepSeek-R1-Distill-Qwen-14B-abliterated-... Text Generation • Updated 8 days ago • 5.94k • 113	mradermacher/DeepSeek-R1-Distill-Qwen-14B-Uncensore... Updated 30 days ago • 185k • 84
unsloth/r1-1776-distill-llama-70b-GGUF Text Generation • Updated 1 day ago • 858 • 9	unsloth/DeepSeek-R1-Distill-Qwen-32B-GGUF Updated 30 days ago • 628k • 117
rinna/deepseek-r1-distill-qwen2.5-bakeneko-32b-gguf Text Generation • Updated 10 days ago • 8.49k • 14	lmstudio-community/DeepSeek-R1-Distill-Qwen-7B-GGUF Text Generation • Updated Jan 21 • 947k • 66

2025/2/24時点の
HuggingFaceの様子.
Distillで5276件のモデルがヒット
しかし1年前は...

2024年 上半期

Preprintの初稿はMay 2023
(arXiv:2305.17493)

コンピューターサイエンス：生成AIのデータで訓練された AIモデルが崩壊する可能性

Nature

2024年7月25日

AIが生成したデータセットを次世代の機械学習モデルの学習に使用すると、その出力が汚染される可能性があることを報告する論文が、*Nature*に掲載される。この研究は、数世代以内にオリジナルのコンテンツが無関係のナンセンスなものに置き換えられてしまうことを示しており、AIモデルの学習に信頼性の高いデータを使用することの重要性を示している。

nature

AI models collapse when trained on recursively generated data
Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson & Yarin Gal
Nature volume 631, pages 755–759 (2024)

<https://www.natureasia.com/ja-jp/nature/pr-highlights/14983>

現実との違い

- 人間がモデルや合成データの品質をチェックする
 - 基本的には良いものだけを残す
 - Perplexityが上がったら開発は中止する
- 実際はより大きなモデルサイズで開発
 - 論文: 0.1 B (人間よりも低品質?)
 - 現実: larger (場合によっては人間よりも高品質)

独自の大規模言語モデルを模倣することの虚偽の約束

May 2023
(arXiv:2305.15717)

arXiv > cs > arXiv:2305.15717

Search.

Help

Computer Science > Computation and Language

[Submitted on 25 May 2023]

The False Promise of Imitating Proprietary LLMs

Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, Dawn Song

An emerging method to cheaply improve a weaker language model is to finetune it on outputs from a stronger model, such as a proprietary system like ChatGPT (e.g., Alpaca, Self-Instruct, and others). This approach looks to cheaply imitate the proprietary model's capabilities using a weaker open-source model. In this work, we critically analyze this approach. We first finetune a series of LMs that imitate ChatGPT using varying base model sizes (1.5B–13B), data sources, and imitation data amounts (0.3M–150M tokens). We then evaluate the models using crowd raters and canonical NLP benchmarks. Initially, we were surprised by the output quality of our imitation models -- they appear far better at following instructions, and crowd workers rate their outputs as competitive with ChatGPT. However, when conducting more targeted automatic evaluations, we find that imitation models close little to none of the gap from the base LM to ChatGPT on tasks that are not heavily supported in the imitation data. We show that these performance discrepancies may slip past human raters because imitation models are adept at mimicking ChatGPT's style but not its factuality. Overall, we conclude that model imitation is a false promise: there exists a substantial capabilities gap between open and closed LMs that, with current methods, can only be bridged using an unwieldy amount of imitation data or by using more capable base LMs. In turn, we argue that the highest leverage action for improving open-source models is to tackle the difficult challenge of developing better base LMs, rather than taking the shortcut of imitating proprietary systems.

この研究では、さまざまなモデルサイズ(1.5B~13B)、データソース、模倣データ量(0.3M~150Mトークン)を使用して、ChatGPTを模倣する一連の言語モデルを微調整しました。その結果、模倣モデルは指示に従う能力や出力のスタイルにおいてChatGPTに近づくものの、事実性や問題解決能力などの面では依然として大きな差があることが明らかになりました。このことから、著者らは、オープンソースのモデルを改善するためには、独自モデルの模倣ではなく、基盤となるモデル自体の能力向上に取り組むべきだと結論づけています。

現実との違い

- **合成データの品質とモデルの基礎性能が低めだった**
 - 論文投稿は23年5月 (初代Llama & ChatGPTで検証)
- **合成データの使用量が必ずしも多くなかった (0.3-150 M token)**
 - 最近では事前学習で>> 1 B tokenを入れる事例も普通
 - Imitating(模倣)も徹底的に行えば、それなりに身につく
 - e.g., Microsoftによる事前学習段階での指示学習の活用 (arXiv:2406.14491, 2024)
- **推論周りのalignmentの有用性がわかってきた**
 - 最近だとDeepSeek V3 to R1事後学習、そしてdistillの有用性などが有名

事後学習は高品質な1000件程度で良い

Computer Science > Computation and Language

[Submitted on 18 May 2023]

LIMA: Less Is More for Alignment

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, Omer Levy

Large language models are trained in two stages: (1) unsupervised pretraining from raw text, to learn general-purpose representations, and (2) large scale instruction tuning and reinforcement learning, to better align to end tasks and user preferences. We measure the relative importance of these two stages by training LIMA, a 65B parameter LLaMa language model fine-tuned with the standard supervised loss on only 1,000 carefully curated prompts and responses, without any reinforcement learning or human preference modeling. LIMA demonstrates remarkably strong performance, learning to follow specific response formats from only a handful of examples in the training data, including complex queries that range from planning trip itineraries to speculating about alternate history. Moreover, the model tends to generalize well to unseen tasks that did not appear in the training data. In a controlled human study, responses from LIMA are either equivalent or strictly preferred to GPT-4 in 43% of cases; this statistic is as high as 58% when compared to Bard and 65% versus DaVinci003, which was trained with human feedback. Taken together, these results strongly suggest that almost all knowledge in large language models is learned during pretraining, and only limited instruction tuning data is necessary to teach models to produce high quality output.

現実との違い

- **ベースに使うモデルで性能は大きく変わる**
 - 成否は事前学習の品質に大きく依存する
 - 国産LLMを1kのデータでSFTしても、まともなモデルは得られなかった(体験談)
- **解きたいタスクで性能は大きく変わる**
 - もやっと作文をさせたいのか、形式を遵守した出力をしたいのか
- **事後データの平均品質が大きく上がっている**
 - 2023年5月という、LLM的には「はるか昔」の報告

一連の論文が示唆する学習指針

- 人間の書いた文章を重用する
 - Webテキストを多量に学習させる (書籍は入手が困難なので。)
- 事後学習は控えめにする
 - 高品質データを頑張って手作業で作る

ネット上の日本語データをたくさん集めて、高品質な事後学習データを1000件ほど作れば、ChatGPT的なものを作れるようになる...?

(しかしよく見ると、論文の投稿時期は23年の中頃。つまり実際に研究が行われたのは22年末ー23年初旬頃であり、24年とは情勢が大きく異なる。なので、これらを真に受けてしまうと...)

Tanukiの構築

GENIAC第一期 東大 松尾・岩澤研究室での採択テーマ

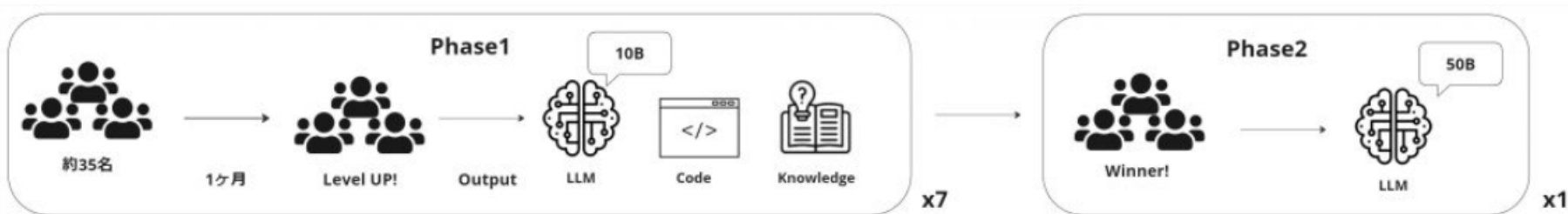
LLM開発者を増やし、日本からイノベーションを生み出す



2つのフェーズから構成されており、

Phase1では7チームが10Bクラスのモデルを開発するコンペティションを実施。

Phase2では、Phase1の優勝チームが50Bクラスのモデル開発に挑戦します。



開発メンバー(phase2)

開発メンバー

畠山 歓 [リーダー]、asaoka_tadashi、Atsushi Saito、Chattso-GPT、Chihiro Arata、Chihiro HIGUCHI、Daichi Kohmoto、Esty、Hideaki Hayashi、hiroaki shioya、Issei Fujimoto、Jie Zeng、Jinsei Shiraishi、K. Nishizawa、Kazutaka Nishimae、Kunihiro Watanabe、masaki okamura、Minami Someya、Mさん、Nishi、Nishijima、p1atdev、Rumi Nakagawa、Ryota Mitsuhashi、Susumu Ota、takagi、Toshio Nishida、y_morinaga、Yuki Namiuchi、Yukie Kawano、永原恒治、加藤純、河越 淳、岩田 兼太朗、菊池満帆、熊田匡仁、江國翔太、佐野敏幸、山口 裕輝、西井康隆、川村 正春、片上 舜、堀江吏将、林寛太 (Kanta Hayashi)

**予備検討:
Webコーパス+事後学習1k件では
良いモデルは得られなかった***

*発表者の検討に基づく結果 (GENIACの前に実施)

Llm-jp-13b-v1.0の事後学習

*日英合わせて約300 B tokenのWebコーパスを
事前学習したモデル

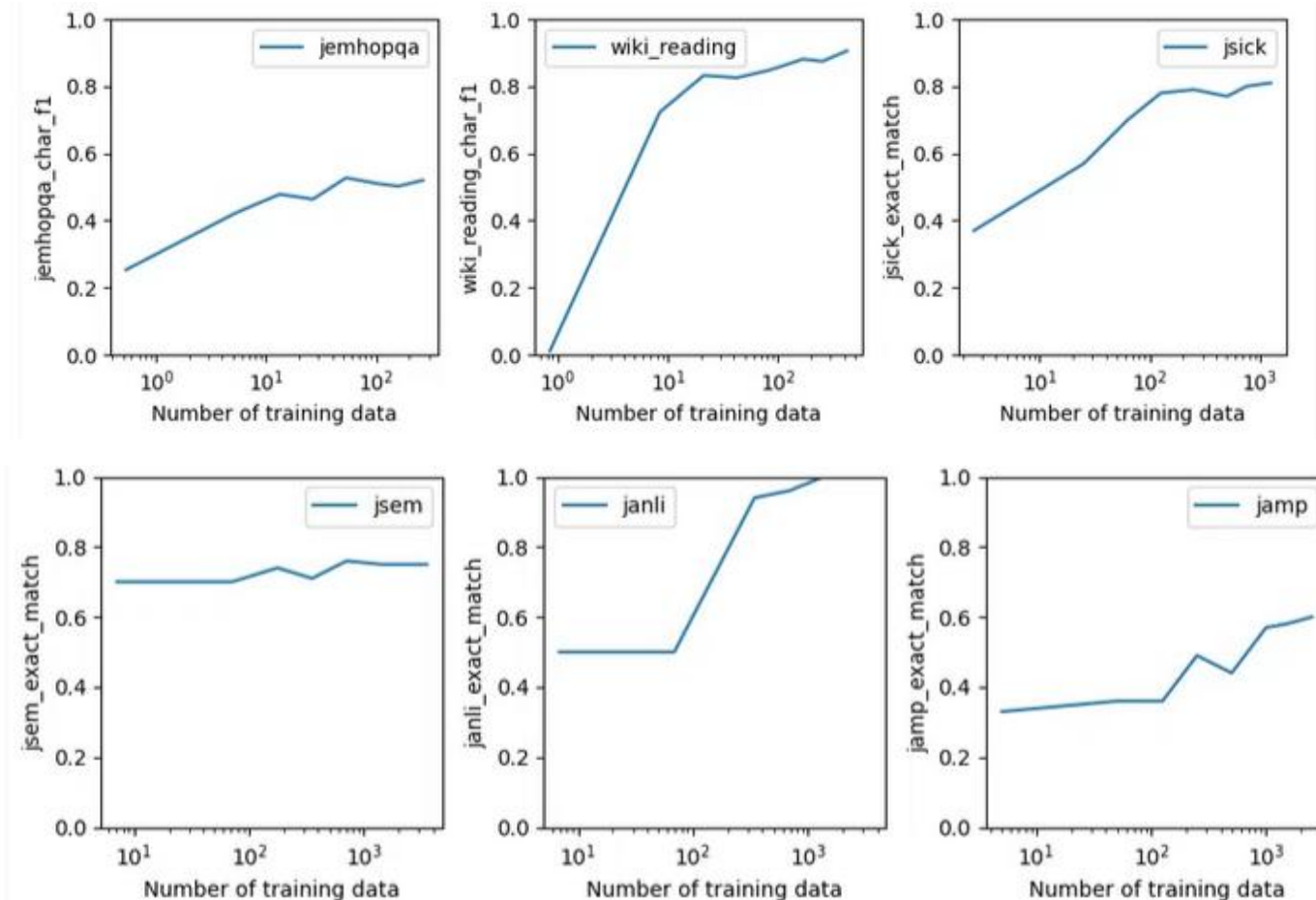
- 種々の事後学習を施してJaster/JLUEベンチマークで評価

B	C	D	E	F	G
主要db	model by	lr	epoch	AVG_jglue	AVG_jaster
ベースモデル	llmjp	-	-	0.0545	0.0675
Dolly	hatakeyama	2.00E-05	2	0.2318	0.1349
Ichikara	hatakeyama	2.00E-05	2	0.0255	0.044
Oasst	hatakeyama	2.00E-05	2	0.0893	0.1229
DPO	llmjp	-	-	0.1564	0.0851
Dolly-ichikara-Oasst	llmjp	-	-	0.165	0.1207
LoRA	llmjp	-	-	0.1142	0.1507
jaster + oasst+dolly	llmjp	-	-	0.9113	0.4688
dolly-oasst	llmjp	-	-	0.1569	0.1551
jaster	llmjp	-	-	0.9134	0.4748
Ichikara	hatakeyama	2.00E-05	3	0.0777	0.0453

高品質な指示学習データセットIchikaraで数千件程度、指示学習したが、Jasterのスコアはほぼゼロ

予備検討: ベンチマークのdevデータの学習

- llm-jp/llm-jp-13b-v1.0にベンチマークのdevデータを数百件程度、学習させた条件でスコアが上昇
- Devデータを学習させない条件では、出力形式の遵守そのものに失敗



個人的に立てた仮説

- Webコーパス + 少量の事後学習で良いモデルを得るのは難しい
- ユーザーの指示を遵守した出力を得るには、類似タスクをあらかじめ数百件程度は学習しておく必要がある
- アーキテクチャやトークナイザは大同小異なので、おそらくは(事後)学習データが最も重要

前半戦 (Phase1: 4-5月ごろ)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R		
1	Megatron-DeepSpeedで10bクラスのモデル構築を行おうとした際の取り組みメモ(phase1での記録)																			
2			node	gpu	global_batch	micro_batch	size (b)	zero stage	pp	mp	micro_batch	activation_checkpoint	view小	view大	TFLOPs	ワット消費電	rope	備考1	備考2	
3			4x23	2	0	32	global_batch: 10a		1	1		FALSE	64	80	72.70		切れた			
4			4x23	2	0	32	global_batch: 10a		0	1	1	FALSE	メモリオーバー	メモリオーバー			切れた			
5			4x23	2	0	16	global_batch: 10a		0	1	1	FALSE	メモリオーバー	メモリオーバー			切れた			
6			4x23	2	0	32	global_batch: 10a		1	1	2	FALSE	71	71	104		切れた			
7			4x23	2	0	32	global_batch: 10a		1	2	2	FALSE	62	62	124.36		切れた			
8			4x23	2	0	32	global_batch: 10b		1	4	4	FALSE	66	66			切れた		assert	
9			4x23	2	0	32	global_batch: 10b		1	2	2	FALSE	66	66	116.12	200-450で変動	切れた			
10			4x23	2	0	32	global_batch: 10b		1	6	6	FALSE					切れた		batch sizeで割れ残し1つ	
11			4x23	2	0	32	global_batch: 10b		1	2	1	FALSE	60	60	116	200-400くらい	切れた			
12			4x23	2	0	32	global_batch: 10b		1	4	1	FALSE	55	55	77.88	150-400	切れた			
13			4x23	1	0	64	global_batch: 2.7	1	1	1	FALSE	80	80	107.23	490-600	切れた	1 nodeの値、異設備			
14			4x23	1	0	64	global_batch: 2.7	1	2	2	FALSE					切れた	正常			
15			4x23	1	0	16	global_batch: 2.7	1	1	2	FALSE				80	切れた	flash attention v2をoff			
16			4x23	2	0	32	global_batch: 7	1	1	1	FALSE				76	切れた				
17			4x23	2	0	32	global_batch: 7	1	2	1	FALSE	46	46	112	200-300	切れた				
18			4x23	2	0	48	global_batch: 7	1	2	1	FALSE	59	59	132	200-400	切れた				
19	zombie vramで1		4x23	2	0	64	global_batch: 7	1	2	1	FALSE	69	69	142	200-500	off	cuda out of memory(1			
20	zombie vramで1		4x23	2	0	80	global_batch: 7	1	2	1	FALSE	70	70	200-500	off	47.24メモリオーバー				
21	zombie vramで1		4x24	3	0	96	global_batch: 7	1	3	1	FALSE	65	67	70	110-500	off	gpuが一瞬、リバースで			
22	zombie vramで1		4x24	3	0	96	global_batch: 7	1	6	1	FALSE					off	cuda out of memory			
23			4x24	3	0	96	global_batch: 7	1	3	3	FALSE					off	24が32の倍値じゃなく14			
24			4x24	3	0	72	global_batch: 7	1	3	3	FALSE					off	24が32の倍値じゃなく14			
25	zombie vramで1		4x24	3	0	96	global_batch: 7	1	3	1	FALSE				70	110-500	on	cuda out of memory		
26	zombie vramで1		4x24	3	0	96	global_batch: 7	1	3	1	FALSE				70	110-500	off	cuda out of memory		
27	zombie vramで1		4x24	3	0	72	global_batch: 7	1	3	1	FALSE						off	cuda out of memory		
28	zombie vramで1		4x24	3	0	72	global_batch: 7	1	1	1	FALSE						off	12.8から1.12でバースが		
29			4x24	3	0	96	global_batch: 7	1	3	1	FALSE	65	66	102	130-500	off				
30			4x24	3	0	96	global_batch: 7	1	3	2	FALSE					off	cuda allocate error			
31			4x24	3	0	96	global_batch: 7	1	6	1	FALSE				50	off				
32			4x24	3	0	48	global_batch: 13	1	3	1	FALSE	69	70	107	10-500	off				
33			4x24	3	0	96	global_batch: 7	1	3	1	TRUE	37	38	110	110-680	off				
34			4x24	3	0	192	global_batch: 7	1	3	1	TRUE	43	44	116	110-700	off				
35			4x24	3	0	768	global_batch: 7	1	3	1	TRUE	over	over			off				
36			4x24	3	0	384	global_batch: 7	1	3	1	TRUE	over	over			off				
37			4x24	3	0	288	global_batch: 7	1	3	1	TRUE	over	over			off				
38			4x24	3	0	240	global_batch: 7	1	3	1	TRUE	over	over			off				
39			4x24	1	0	80	global_batch: 7	1	1	1	TRUE	over	over			off				
40			4x24	1	0	64	global_batch: 7	1	1	1	TRUE	over	over			off				
41			4x24	1	0	32	global_batch: 7	1	1	1	TRUE	over	over			off				
42			4x24	1	0	32	global_batch: 7	1	1	1	FALSE	70	80	240	400-650	off				
43			4x24	1	0	32	global_batch: 13		1	1	FALSE	over	over	80	244	490-680	off			
44			4x24	1	0	32	global_batch: 10b		1	1	FALSE	over	over			off				
45			4x24	1	0	24	global_batch: 10b		1	1	FALSE	over	over			off				
46			4x24	1	0	16	global_batch: 10b		1	1	FALSE	over	over	80	80	180.72	500-600	off		
47			4x24	1	0	32	global_batch: 7	1	1	1	FALSE		70	80	244	490-680	off	→normalization memom →attention_dropout 0 →hidden_dropout 0		
48			4x20	3	0	1024	1	13	1	3	1	FALSE	over	over			off			
49			4x20	3	0	512	1	13	1	3	1	FALSE			142.13		off			
50			4x20	2	0	512	1	13	1	2	1	FALSE	74	70	320	650	off			
51			4x20	2	0	512	2	13	1	2	1	FALSE	50	70	334		off	メモリ消費量が50-80		
52			4x20	2	0	512	3	13	1	2	1	FALSE					off	512を3で割れ残し1		
53			4x20	2	0	512	4	13	1	2	1	FALSE	over	over			off			
54			4x20	2	0	256	1	13	0	2	1	FALSE	over	over			off			
55			4x20	2	0	128	1	13	0	2	1	FALSE	over	over			off			
56			4x20	2	0	16	1	13	0	2	1	FALSE	over	over			off			
57			4x20	2	0	512	2	13	1	4	1	FALSE	60	75	334	600	off			
58			4x20	2	0	512	2	13	1	2	2	FALSE	60	64	270		off			
59	※・C側の調整		4x20	1	0	1024	1	7	1	1	1	FALSE	over	over			off			
60			4x20	2	0	1024	2	7	1	2	1	FALSE	54	54	306		off			
61			4x20	2	0	1536	1	7	1	2	1	FALSE	over	over						
62			4x20	2	0	1280	1	7	1	2	1	FALSE	45	45	270					
63			4x20	2	0	1280	2	7	1	2	1	FALSE	54	54	310					
64			4x20	2	0	1536	2	7	1	2	1	FALSE	54	54	311					
65			4x20	2	0	2048	2	7	1	2	1	FALSE	54	54	314					
66			4x20	2	0	1024	2	7	1	6	1	FALSE	32	44	264					
67			4x20	3	0	1536	3	7	1	3	1	FALSE	59	60	308					
68			4x20	3	0	1536	3	13	1	3	1	FALSE	over	over						
69			4x20	3	0	1536	3	13	1	6	1	FALSE	70	70	240	650-710		23day		
70			4x20	3	0	512	3	13	1	6	1	FALSE					error (refill L1)			
71			4x27	3	0	768	3	13	1	6	1	FALSE			324		24day			
72			4x27	3	0	1536	3	13	1	6	1	FALSE					error (refill L1)			
73			4x27	3	0	1536	6	13	1	6	1	FALSE	over	over			スタックする			
74			4x27	3	0	1536	3	13	1	3	1	FALSE	over	over						
75			4x27	3	0	1536	3	13	1	1	1	FALSE					AssertionError: global batch size (1536) is not divisible by micro batch size			
76			4x20	3	0	1536	3	13	1	6	1	TRUE	35	40	250	650-710				
77			4x20	3	0	1536	6	13	1	6	1	TRUE	42	47	220		28日計かる、logcatは溜			
78			4x20	3	0	1536	3	13	1	12	1	TRUE					スタックする			
79	4k llama		4x20	3	0	1536	6 147		3	1	1	FALSE	60	65	401					
80	4k llama		4x20	3	0	1536	6 147		3	1	2	FALSE	77	79	423					
81	4k llama		4x20	3	0	1536	6 147		6	1	2	TRUE	32	35	430					
82	4k llama		4x20	3	0	2072	6 147		3	1	2	FALSE	73	75	270			以下1/4メモークラスタ		
83	4k llama		4x27	3	0	2072	6 147		6	1	2	FALSE	73	75	272					
84			4x28	2	0	1536	4 llama-13		4	1	2	FALSE			430					
85			4x28	3	0	1536	6 llama-13		6	1	3	FALSE	71	79	422	600-700		16日で終わる模様		
86			4x20	3	0	1536	3	6	1	3	1		70%増		460					
87			4x20	3	0	1536	3	7	1	6	1	recompute-activations			375		14.4day			
88			4x20	3	0	1536	3	7	1											

ハイパラの最適化 (50条件超)
計算速度(FLOPs)を上げるた
めの様々な因子を検討

データ収集加工: 事前学習

H100 x8の3ノードで学習を実施(1ヶ月程度: 初めての取り組みのため、データ処理etcで色々と苦戦)

- **日本語データ (約100 b tokens)**
 - 種々の日本語コーパス + CommonCrawl (数snapshot分)
 - ルールベース&確率&機械学習フィルタを組み合わせ、**クリーニング**を実施
 - 広告文など、類似性の高い&クオリティの低いデータを相当数(数十%?)、除外
 - 限られたリソースで最大限の学習効率を出すことが目的
- **英語データ (約100 b tokens)**
 - 英語論文+英語Wiki+コードデータ
 - 一般的なWebサイトの文章は未学習
 - 日本語モデルにおいて、学習の重要度は低いと判断*
 - 限られたリソースを踏まえ、選択と集中を実施

事後学習の工夫

- 指示データを100万件以上、自動生成して学習 (by Mixtral 8x22b)
 - 可能な限り多くのケースを想定しながら、人類との対話訓練を大量に行う戦略
 - データの例 (特定のベンチマークは想定せず、様々なケースを想定した汎用性あるQ&Aを生成)
 - マルチターン会話
 - 既存の指示データセットの質問に対する回答の再作成 (人手の回答よりも高品質なものが多かった印象)
 - 既存の質問からの類題の生成
 - 特定のキーワードからの問題生成
 - 既存の文章を元にしたQ&Aを生成

自動生成されたQ&Aの例 (正否判定が、専門家にも難しくなっているレベル)

Q. 有機化学において、「極性転換」(きょくせいてんかん)はどのようなものでしょうか？また、それが1,4-ジケトンを経合成する際に重要となる理由は何でしょうか。具体的な例を交えて説明してください。

A. 有機化学における極性変換(きょくせいへんかん)とは、あるシントーンに対応する官能基を化学的な方法で変換して、そのシントーンとは逆の電荷を持つシントーンにすることです。これは特に逆合成(逆反応)の際に重要な概念となります。例えば、1,4-ジケトン($R-CO-CH_2-CH_2-CO-R'$)を経合成すると、3-カルボニルカルボカチオン($R-CO-CH_2-C^+H_2$)とアシルアニオン($-CO-R'$)に分解されます。この際、3-カルボニルカルボカチオンは α, β -不飽和カルボニル化合物があるためシントーンとして考えられますが、アシルアニオンについては、求電子性を持つ官能基であることから元のシントーンと矛盾しています。そこで、一旦カルボニル基を保護し、その後保護基によって極性変換を行う必要があるため、極性変換が重要となります。具体的には、シアノ基を付加させシアノヒドリンエーテルに変換したり、1,3-プロパンジチオールを付加させてジチアンへと変換することで、求核性を持つ官能基を元のカルボニル基に戻すことが可能となります。

- 一問一答形式を想定した独自のデータセットも軽く生成

コンペの結果

ランキング	モデル名 (チーム名)	総合得点 (*1)	Nejumi Leaderboard Neo			追加評価指標		
			合計 (*2)	一問一答	長文生成	合計 (*3)	一問一答	長文生成
1	Team たぬき (hatakeyama)	0.42	0.42	0.37	0.46	0.42	0.37	0.48
2	Team ビジネス (kawagoshi)	0.39	0.38	0.40	0.35	0.40	0.36	0.45
3	Team 天元突破 (ozaki)	0.35	0.35	0.38	0.31	0.35	0.28	0.41
4	Team 甲 (hajjima)	0.29	0.27	0.18	0.36	0.31	0.27	0.35
5	Team Kuma (kumagai)	0.26	0.21	0.23	0.19	0.31	0.31	0.32
6	Team JINIAC (nakamura)	0.18	0.11	0.09	0.13	0.25	0.24	0.26
7	Team Zoo (sannai)	0.10	0.09	0.07	0.10	0.11	0.01	0.21

https://weblab.t.u-tokyo.ac.jp/geniac_llm/

一問一答: Jaster系の出力形式が厳格なベンチマーク
長文生成: Japanese MT-Bench, Elyza tasks100など

国内外モデルとの比較

発表当時としては、Japanese MT-Benchでそれなりに高い性能を実現



	学習トークン数 (b)	JMT-Bench
Tanuki-8b (ours)	280	4.9-5.0 (提出版は4.6)* *コンペ後のDPOなどの追加検討で 性能が向上 (by Hayashiさんなど)
Gemma-7b	6000	4.8
Qwen-7b	2400	5.3
Mistral-7b	(>2000との噂)	5.3 (継続学習モデルRakuten AIのスコア)
Llama2-13b	2400	5.05 (継続学習モデルElyzaのスコア)
Llama2-70b	2400	5.03 (継続学習モデルStableLMのスコア)
Calm-7b	1300	4.4
Weblab-10b	600	2.1

Phase1での感想

- 事後学習はかなり重要である
- LLMは初見の問題には、ほぼ対応できない
 - しかも学習効率が低く、何種類も類題を解かないと性能が上がらない
 - なので多量の演習問題を解かせるのが合理的なアプローチとなる
- 合成データの活用は重要である
 - 過去の論文の主張ではなく、手元の実験結果を信じた方が良さそう
 - 論文は数ヶ月以上、遅れて公開されるので、最新のトレンドは追えてない可能性
 - 24年の上旬頃から、高品質なオープンモデルが多く出現 (過去とは状況が大きく異なる)

後半戦 (Phase2: 6-8月ごろ)

開発指針

- 性能向上に注力
 - とりあえず良いものを無心に作って、その後に何が良かったかを考えるエンジニア的な発想で開発した
- 対話・作文性能に力点
 - Japanese MT-Benchで最高性能を目指す
- 一問一答(Jaster)系は保留
 - 様々なベンチマークを想定した演習問題を多量に作るのが最短ルートだが、そこまでの意義があるかは不明だったため

余談: 幻のDeepSeekモデル

- Phase2の開始時、Llama or DeepSeek V2のどちらのアーキテクチャを選ぶかで、チーム内で意見が割れた。
- 最終的には、学習可能なトークン数、実績、エコシステムの充実度でLlama系アーキテクチャを選定

参考: ハイパラの最適化

Megatron-DeepSpeedで500クラスのモデル構築を行おうとした際の取り組みメモ																								
run	エラーメッセージ	node	gpu	num_workers	global batch	microbatch	zero stage	pp	tp	sp	num_key_value	num layers	hidden size	th_hidden_size	attention head	adam-opt	activation_checkpoint	size (b)	mem	TFLOPs	GPU power %			
0	0/1: 1/1に成功した out of mem	10	0	128	1536	1	1	1	1	1	0	56	0164	21504	48	1.00E-05	FALSE	28.0	over	70	233			
1	0/2: 1/1に成功した global batch 8	10	0	128	768	1	1	1	1	1	0	56	0164	21504	48	1.00E-05	FALSE	28.0						
2	0/2: adam-opt 1/1 adam-opt 8	10	0	128	768	1	1	1	1	1	0	56	0164	21504	48	1.00E-07	FALSE	28.0						
3	0/2: num7+Context out of mem	10	0	128	768	2	1	1	1	1	0	56	0164	21504	48	1.00E-05	FALSE	28.0	over					
4	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	56	0164	21504	48	1.00E-05	FALSE	28.0			237			
5	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	56	0164	21504	48	1.00E-05	FALSE	28.0			230			
6	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	56	0164	21504	48	1.00E-05	FALSE	28.0			170			
7	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37			300			
8	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	70	0164	21576	48	1.00E-05	FALSE	37						
9	0/2: num7+Context out of mem	10	0	128	768	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
10	0/2: num7+Context out of mem	10	0	128	384	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
11	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	80	0164	21504	48	1.00E-05	FALSE	37						
12	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37	70		200			
13	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37	73		200			
14	0/2: num7+Context out of mem	10	0	128	768	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37	over					
15	0/2: num7+Context out of mem	10	0	128	768	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37	77		177			
16	0/2: num7+Context out of mem	10	0	128	1536	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37	77		175			
17	0/2: num7+Context out of mem	10	0	128	1536	8	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
18	0/2: num7+Context out of mem	10	0	128	1536	4	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
19	0/2: num7+Context out of mem	10	0	128	2304	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37	70		180			
20	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	TRUE	37	53		270			
21	0/2: num7+Context out of mem	10	0	128	768	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	TRUE	37	56		254			
22	0/2: num7+Context out of mem	10	0	128	2304	3	1	1	1	1	0	70	0164	21504	48	1.00E-05	TRUE	37	60		333			
23	0/2: num7+Context out of mem	10	0	128	2304	3	1	1	1	1	0	70	0164	21576	48	1.00E-05	TRUE	41	63		306			
24	0/2: num7+Context out of mem	10	0	128	2304	3	1	1	1	1	0	80	0164	21504	48	1.00E-05	TRUE	43	66		331			
25	0/2: num7+Context out of mem	10	0	128	2304	3	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	73		330			
26	0/2: num7+Context out of mem	10	0	128	2304	6	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	80		303			
27	0/2: num7+Context out of mem	10	0	128	2304	3	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46						
28	0/2: num7+Context out of mem	10	0	128	2304	3	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46						
29	0/2: num7+Context out of mem	10	0	128	2304	3	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	80		310			
30	0/2: num7+Context out of mem	10	0	128	2304	4	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	70		320			
31	0/2: num7+Context out of mem	10	0	128	2304	2	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	66		337			
32	0/2: num7+Context out of mem	10	0	128	2304	1	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	63		336			
33	0/2: num7+Context out of mem	10	0	128	2304	1	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46						
34	0/2: num7+Context out of mem	10	0	128	2304	1	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46						
35	0/2: num7+Context out of mem	10	0	128	768	1	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	214					
36	0/2: num7+Context out of mem	10	0	128	4096	1	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	205					
37	0/2: num7+Context out of mem	10	0	128	8216	1	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	63		364			
38	0/2: num7+Context out of mem	10	0	128	16432	1	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	63		382			
39	0/2: num7+Context out of mem	10	0	128	36864	1	1	1	1	1	0	85	0164	21504	48	1.00E-05	TRUE	46	63		380			
40	0/2: num7+Context out of mem	10	0	128	36864	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	TRUE	37	54		368			
41	0/2: num7+Context out of mem	10	0	128	8216	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	TRUE	37						
42	0/2: num7+Context out of mem	10	0	128	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	TRUE	37						
43	0/2: num7+Context out of mem	10	0	128	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	TRUE	37						
44	0/2: num7+Context out of mem	10	0	128	1152	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	TRUE	37						
45	0/2: num7+Context out of mem	10	0	128	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
46	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
47	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
48	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
49	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
50	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
51	0/2: num7+Context out of mem	10	0	300	1152	2	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
52	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
53	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
54	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
55	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
56	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
57	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
58	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
59	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
60	0/2: num7+Context out of mem	10	0	300	1152	1	1	1	1	1	0	70	0164	21504	48	1.00E-05	FALSE	37						
61	0/2: num7+Context out of mem	10	0																					

余談: 幻の38Bモデル

- 始めは38Bモデルを開発していたが、loss spikeが多発したため、中止した (FP8の利用、Rope thetaが大きすぎた、運が悪かったetcが要因?)



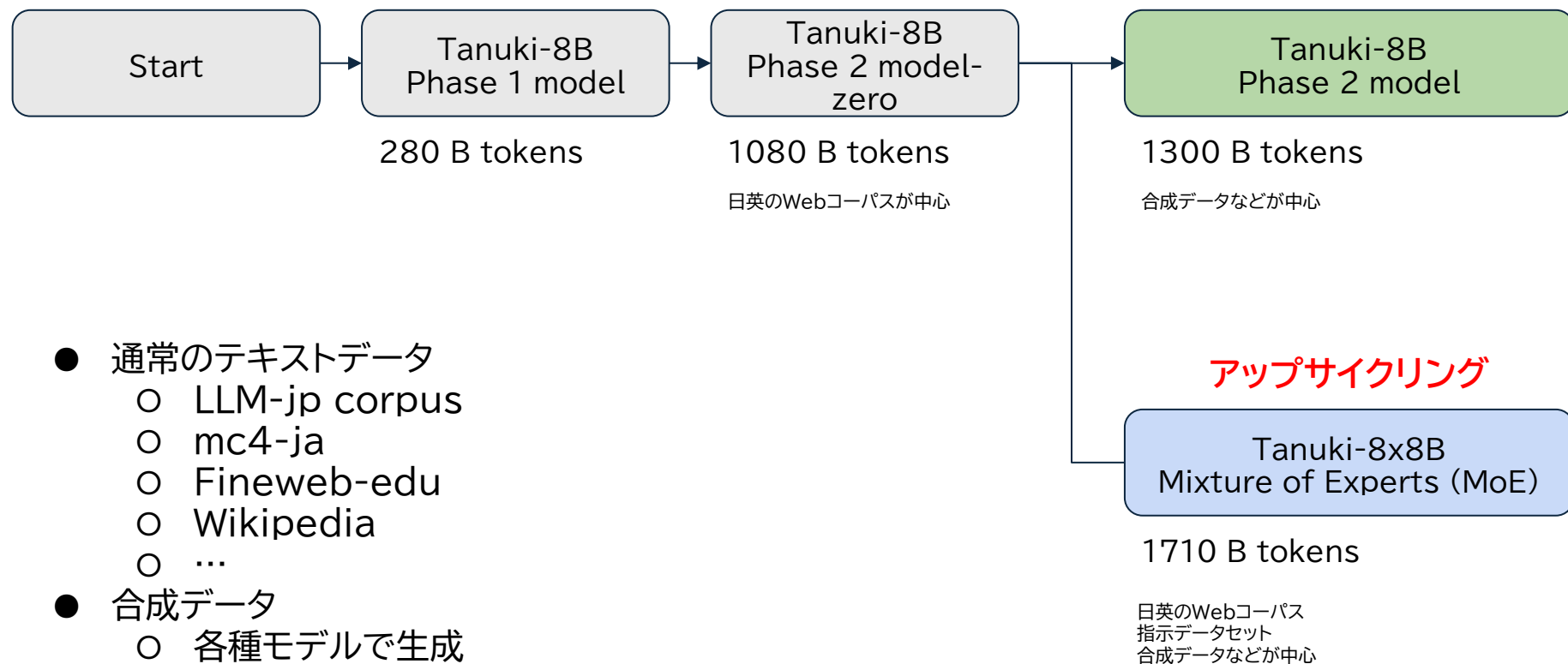
詳細は以下の記事などを参照

<https://zenn.dev/matsuolab/articles/60ba12079b99dd>

選択肢

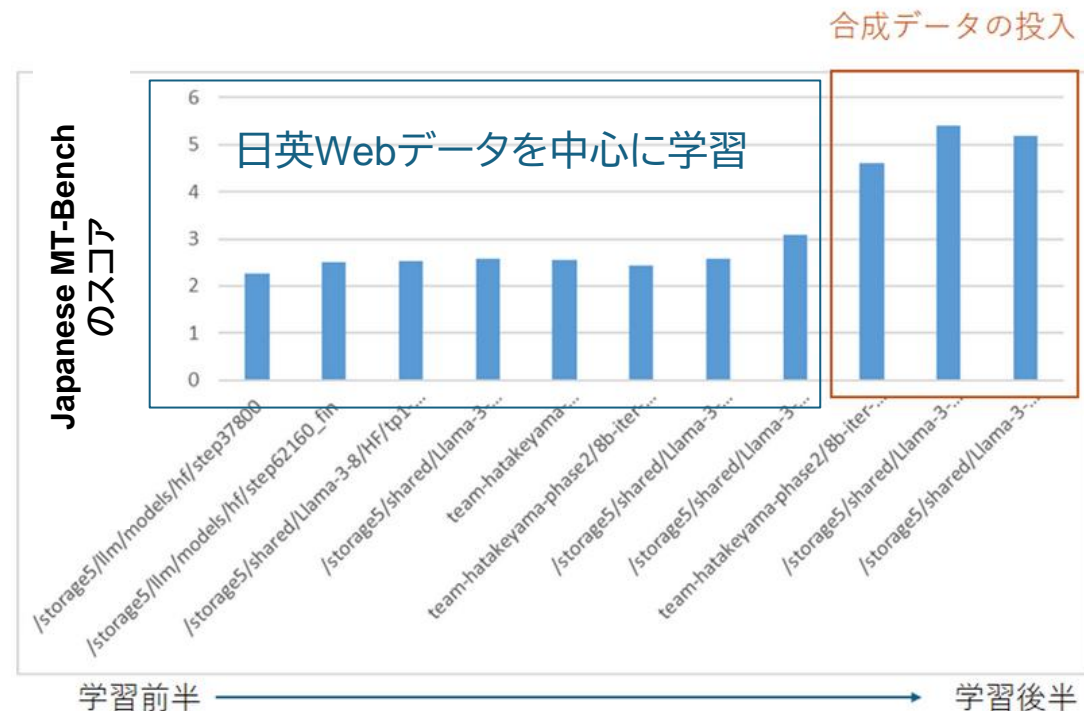
- 38Bモデルを続行する
 - 1-2 weekほどloss spike対策に取り組んだが、見込みが薄かった
 - 精神的にも辛い
- 新たなモデルをゼロから構築する
 - しかし学習させるトークン数がどうしても少なくなる
- Up cyclingする (chosen)
 - Phase 1で作ったモデルに継続学習を行い、その後に8x8Bに拡張する

モデル構築の流れ



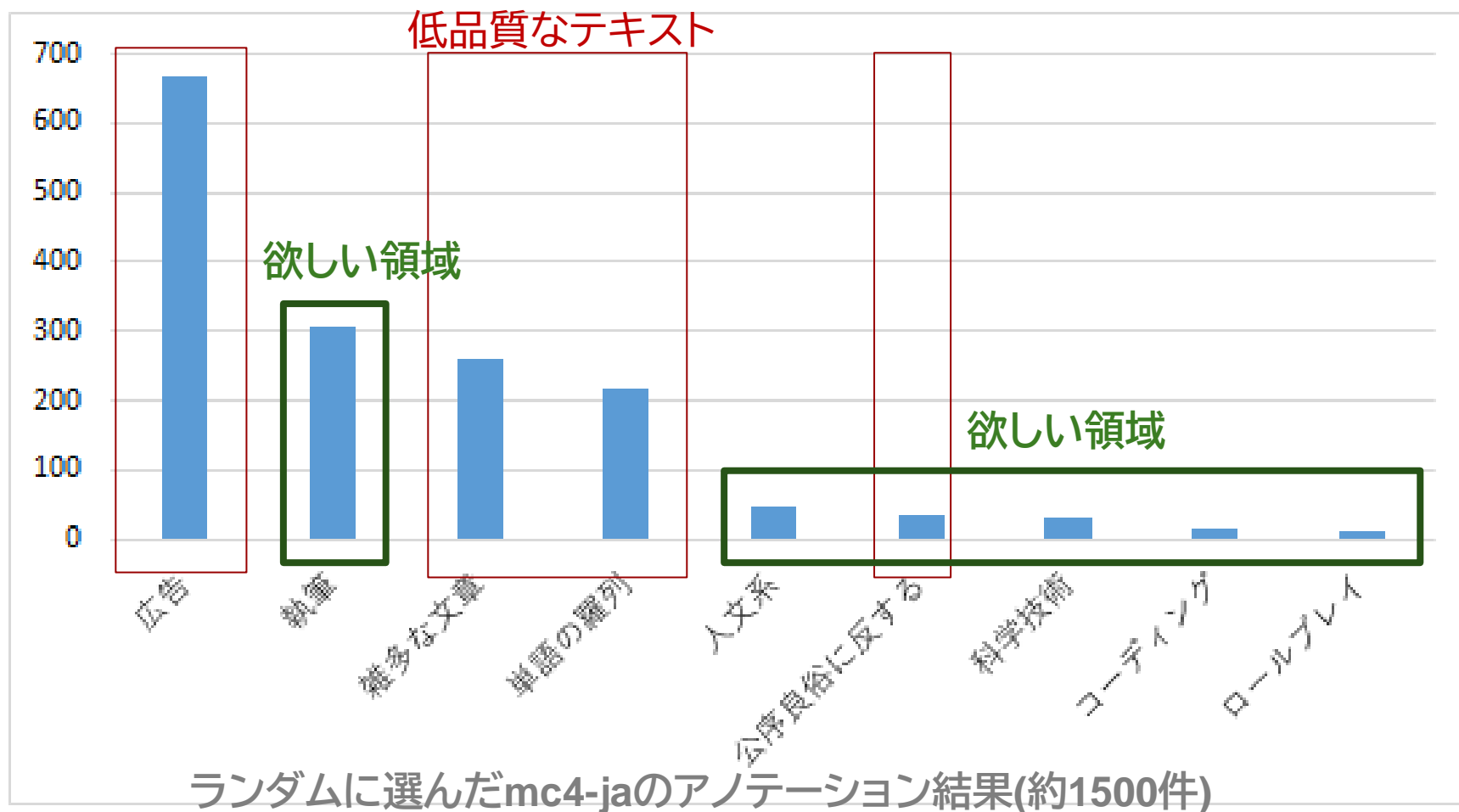
データ準備・学習・評価サイクル

- 事前学習と同時並行で合成データを生成
- データ準備・学習・評価のサイクルを数十回以上(?), 回しながら軌道修正
 - 主な流れ
 - 1. Webデータを学習
 - 作文・対話性能が弱点
 - 2. 合成データを学習
 - 顕著に性能が向上
 - 数学・論理・コーディングが弱点
 - 3. 特化型の合成データを学習
 - 苦手分野に特化して学習



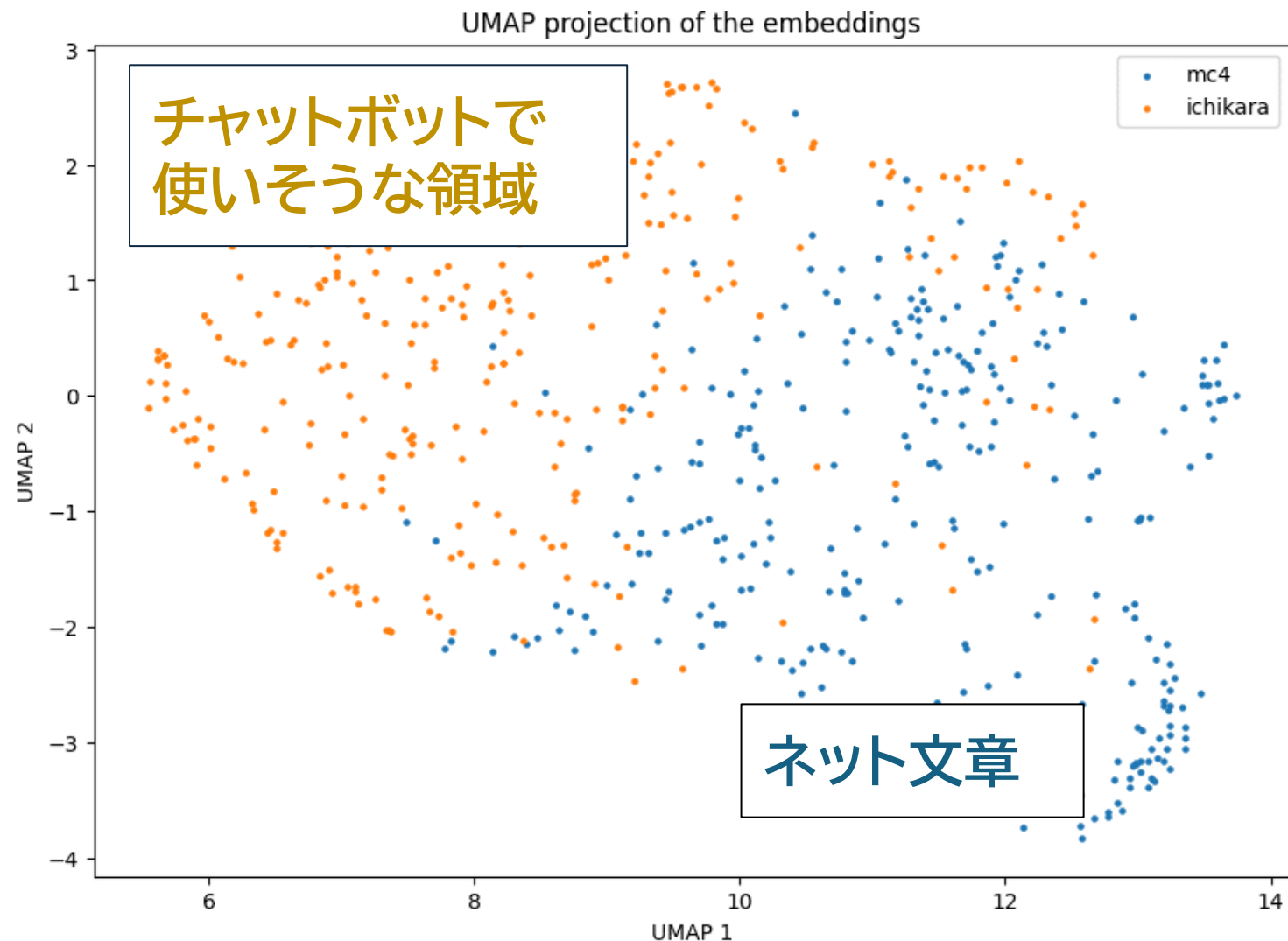
Webデータの課題

- Webサイトの大半は低品質で、教育的な内容は少ない
- 教育的、対話関連の日本語データが極端に不足 → データ合成で補填



Webデータ vs. 対話データ

- mc4-jaと対話データ (ichikara)をembed & umapで2次元転写
 - 詳細はこちら
- Webデータと対話データの重複が少ない
- 対話はWebデータの「外挿領域」に近い？



事前学習に用いた合成データ (>230 B tokens)

dataset	モデル	概要	tokens (b)
1SyntheticText	Phi-3	Creative Commons系テキストの再生成	52
2logical-wizardlm-7b	WizardLM-7b	論理・コード・数学系の合成指示データ(英語)	29
3CommonCrawl-RAG-QA-Calm3-22b-chat	Calm3	CommonCrawl系テキストの再生成	29
4phase2-calm-generated-texts-commoncrawl	Calm3	CommonCrawl系テキストの再生成	25
5SyntheticTextCC	Phi-3	Creative Commons系テキストの再生成	22
6SyntheticTextWikiTranslate	Phi-3	Creative Commons系テキストの再生成	16
7SyntheticTextCCUncleaned	Phi-3	Creative Commons系テキストの再生成	10
80723-calm3-22b-random-genre-inst-sft-multiturn-clean-tsub	Calm3	合成指示データ	7.9
90719-calm3-22b-random-genre-inst-sft-multiturn-tsub	Calm3	合成指示データ	7.5
100804calm3-logical-multiturn-pretrain	Calm3	論理・コード・数学系の合成指示データ	7.1
11logical-wizardlm-7b-ja-0805	Calm3	論理系データのCalm3による日本語訳	5.6
12logical-wizardlm-7b-ja	Calm3	論理系データのCalm3による日本語訳	4.2
130717-calm3-22b-random-genre-inst-sft-tsub	Calm3	合成指示データ	2.8
14logical-wizardlm-7b-ja-0730	Calm3	論理系データのCalm3による日本語訳	2.8
150722-calm3-22b-random-genre-inst-sft-multiturn-tsub	Calm3	合成指示データ	2.4
16SyntheticTextOpenMathInstruct	Phi-3	OpenMathInstructの再生成	2.0
17logical-wizardlm-7b-ja-0731	Calm3	論理系データのCalm3による日本語訳	1.5
180804ramdom-to-fixed-multiturn-Calm3-pretrain-tsub	Calm3	合成指示データ	0.91
19logicaltext-wizardlm8x22b-Ja	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.69
20CreativeCommons-RAG-QA-Mixtral8x22b	Mixtral8x22b	Creative Commons系テキストの再生成	0.56
21wizardlm8x22b-logical-math-coding-sft	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.51
22Synthetic-JP-EN-Coding-Dataset-453k	各種	コードの合成データ	0.42
23wizardlm8x22b-logical-math-coding-sft_additional	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.39
24logicaltext-wizardlm8x22b	wizardlm8x22b	論理・コード・数学系の合成テキスト	0.32
25OrcaJaMixtral8x22b	Mixtral8x22b	Orcaの再生成	0.27
26wizardlm8x22b-logical-math-coding-sft-ja	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.19
27multiturn-Calm3-manual	Calm3	タスクを半固定した指示データ	0.19
28LogicalDatasetsByMixtral8x22b	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.064
29wizardlm8x22b-logical-math-coding-sft_additional-ja	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.060
30OpenMathInstruct-ja-phi3	Phi-3	OpenMathInstructの再生成	0.055
31AutoMultiTurnByCalm3-22B	Calm3	合成指示データ	0.050
32AutoMultiTurnByMixtral8x22b	wizardlm8x22b	合成指示データ	0.049
33AutoWikiQA	Mixtral8x22b	Creative Commons系テキストからの指示データ	0.031
34ramdom-to-fixed-multiturn-Calm3	Calm3	タスクを半固定した指示データ	0.026
35logicaltext-wizardlm8x22b-api	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.019
360717-calm3-22b-random-genre-inst-sft-tsub-part	Calm3	合成指示データ	0.007
37ChatbotArenaJaMixtral8x22b	wizardlm8x22b	ChatbotArenaの回答を自動生成	0.005
38databricks-dolly-15k-ja-regen-nemotron	Nemotron340b	dolly-jaの回答を自動生成	0.002

合成データの生成

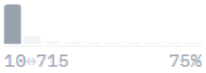
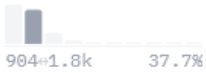
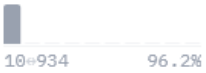

A. 既存テキストの再活用

- ランダムに選んだCommonCrawlの文章を教科書・対話形式などで再生成
 - トピックの多様性の確保
 - 事実に基づく作文 (幻覚の抑制)
 - ノイズ表現の自動削除 ([送料無料] のような表現のLLMによる自動削除)
- 自動英訳もついでに生成
 - 「まともな日本語」to「そこそこの自動英訳」を多量に生成し、知識転移を図る

B. ランダムに選んだキーワードに基づく指示データの生成

- 学習指導要領に基づく単元などを指定
- JSON形式での出力など、抑えておきたい指示なども手作業でセット

生成した合成データの例

q1 string · lengths	a1 string · lengths	q2 string · lengths	a2 string · lengths
			
ある動物園では、ウサギとカメが一緒に暮らしています。ウサギは毎日3匹のエサを食べ、カメは毎日2匹のエサを食べています。ある日、エサの在庫が減ってしまったので動物たちは新しいエサを買いに行くことになりました。しかし、予算には限りがあり、合計で80匹しか買えません。このとき、どのようにエサを分けると良いでしょうか？また、その理由も考えてください。	Aさん「もし全員が均等にエサをもらえるなら、一日に必要なエサの量を考えれば、ウサギは一日あたり7匹、カメは一日あたり5.67匹が必要になるから、単純に考えるとウサギの方が多くなるよね。」 Bさん「ただし、現実的にはカメの方が必要なエサの量が少ないからといって無視するわけにはいかないよ。ウサギが速いペースで動くとしても、カメも同じように重要な存在だからね。」 Cさん「そうだね、でもエサの在庫管理を考えるとウサギの必要量が多いのは事実だし、その分カメに配分を増やすのは難しいかもしれない。じゃあ、どうすればバランスを取れるかな？」 Dさん「一度、全体の目標を設定して考えるのが良いんじゃないか。『全員で合計100匹分の食事を確保』といった明確な数値目標があれば調整しやすいかも。」 Eさん「『全員で合計100匹分の食事を確保』、という目標はいいね！それなら、それぞれの動物に対してどれだけの割合で分配するかを決めることができるはずだ。具体的に、例えばウサギには40%の配分を、カメには残りの60%の配分をするというのはどうだろう？」 Fさん「『全体で100匹分の食事を確保』を目標に設定する方法は賛成だけど、その場合の具体的な割合についてももう少し考え直す必要があると思うんだ。ウサギとカメの活動時間や体力を考慮しないと不公平だと思うんだけど...」 Gさん「確かに活動時間や体力の違いは重要だね。例えば、ウサギは夜行性でよく動くけど、カメはあまり動かないから長時間のエサの消費が見込まれる場合もあるしね。」	1. エサの分割が事前には同意されていたとしても、個々のウサギやカメの健康状態や体重の変化によって最適な栄養摂取量は変わる可能性があるため、定期的な見直しが必要です。具体的にはどのくらいの頻度で評価を行い、必要に応じて配分を修正する方法を検討するべきですか？ 2. もしもエサの価格が高騰した場合、予算内での必要量を賄うことが難しくなります。このような場合に備えて、他の安価な代替品や地元での自給可能な食料源を活用するなどの緊急対応策を考えていますか？	エコツーリズムの計画において、動物たちの健康管理は非常に重要な要素です。そしてそれに伴うエサの管理も同様に大切です。以下に具体的な回答を示します。 1. 評価および配分の見直し頻度： - 各ウサギまたはカメの個体差により、健康状態や成長速度は異なりますので、少なくとも月に一度の評価を行うことをお勧めします。特に初期段階の成長期においては、毎週の評価が望ましいかもしれません。これにより、各個体のニーズを適切に把握し、その都度適切な量の餌を提供することで、健康に保つことができます。 2. 価格高騰時の対応策： - 価格が上昇した場合には、以下のような対策を考えるべきです。 - *食材の多様化*：高騰している特定の飼料ではなく、代用となる安価な植物性食品や地元で入手できる生野菜などを取り入れます。例えば、野菜や果物のくずを再利用することも考えられます。 - *供給元の変更*：一時的に異なるサプライヤーから購入するか、地域の農家や市場から直接調達を試みることでコストを抑えることが可能です。また、在庫の調整や効率的な使用方法を考慮しつつ、無駄を最小限に抑えることも有効です。 - *保存技術の活用*：乾燥・冷凍などの技術を用いて長期保存が可能な方法を探ることで、急な価格変動にも対応しやすくなります。 また、長期的な視点で見た際に、持続可能な農業技術を養う機会として捉えることも可能であり、コミュニティ全体での共同作業を通じて安定した食料供給を目指すことが有益でしょう。 これらの対策を講じることにより、動物たちの適正な栄養を確保しながら費用対効果の高い管理が実現できると信じています。ご参考になれば幸いです。

今から見返すと、
まだまだ品質には
向上の余地あり

指示

ランダムに
問題生成

対話風の
解答を生成

追加質問を
生成

レポート風の
解答を生成

合成データの学習

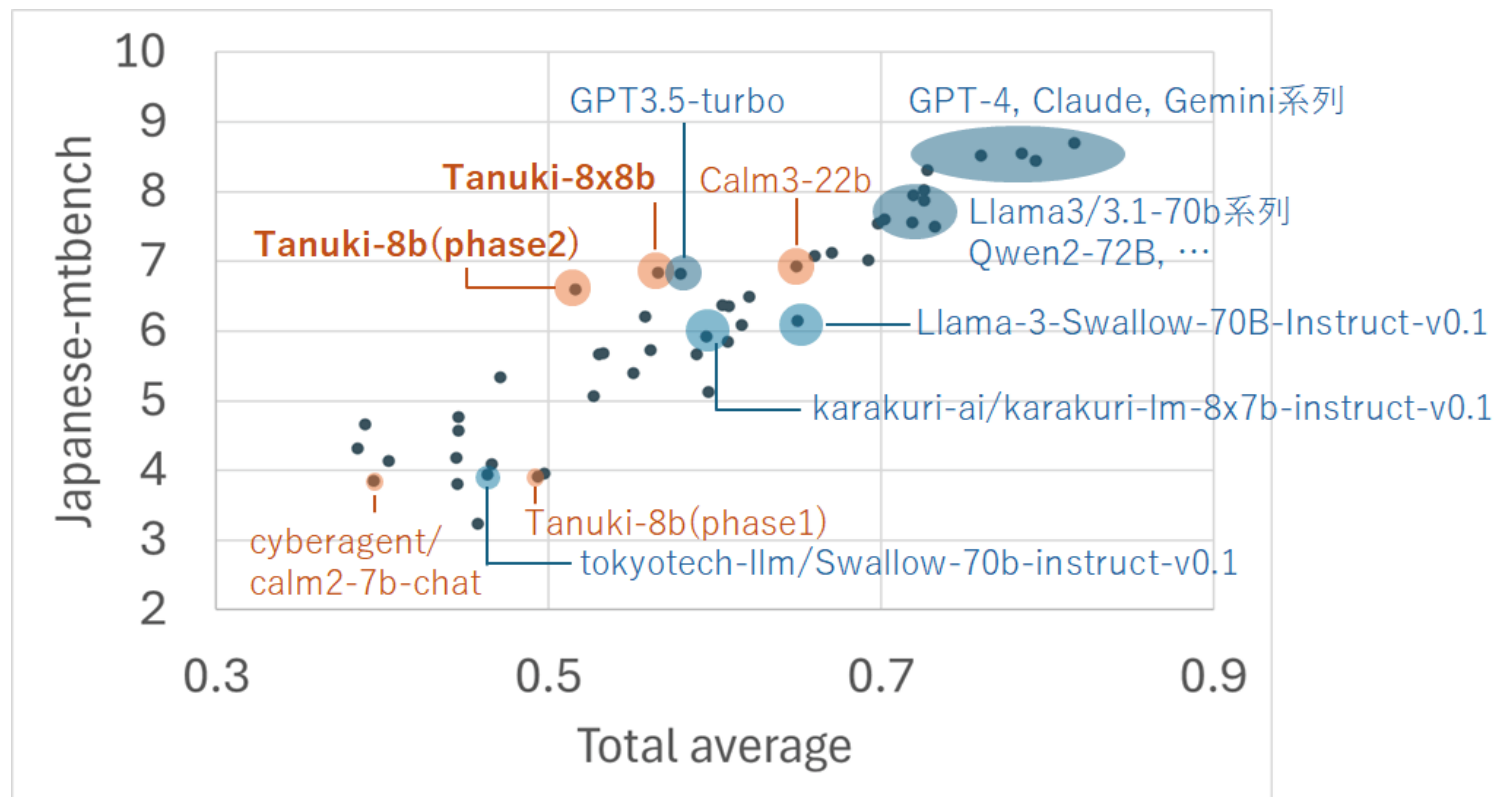
- データ量が多いので、通常の事前学習の一環(Megatron)で学習
 - 指示データは以下の形式に変換して学習
 - user:
 - assistant:
 - user:
 - assistant: ...
- 事前学習の最終盤&事後学習は、合成データのみを使用
 - 最後の数十Bトークン程度
- 事後学習は文字通りの微調整
 - SFT/DPOをLoRAで実施
 - 事前学習でも使ったデータをピックアップして学習
 - 乱数調整のイメージ
 - 計算コストを下げて試行錯誤の回数を増加

モデル評価

Japanese MT-Bench, Jaster系, 実戦形式でのテストの評価の関係性

Nejumi Leaderboard3における位置づけ

- フルスクラッチの国産モデルとしては、Japanese MT-Benchで最高クラス (24年8月時点)
- Jaster系のスコアは低め (考察は後述)



モデルの実践的な対話性能をブラインドテストで評価

- ChatBot Arena形式
- 第一弾: 開発メンバーを中心に評価 (24年8月中旬): 1800件
- 第二弾: 公開版を仮運用 (24年10月ー12月): 500件ほど

🗡 Chatbot Arena 日本語版(α版)

🔥 使い方

- 質問をすると、ランダムに選ばれた2つのモデルが回答します。
- どちらのモデルがより良い回答をするかを選択してください。
- リーダーボードはこちら: [here](#).

👉 Chat now!

The screenshot displays the Chatbot Arena interface. At the top, a search bar says "Expand to see the descriptions of 20 models". Below it, two chat windows are shown side-by-side for comparison.

Model A (Left):

h100を借りるのにどれくらいかかると思います

ご質問ありがとうございます。H100を借りる費用は、レンタル会社やレンタル期間によって異なります。

一般的なレンタル費用は、1日あたり10,000円から20,000円程度です。ただし、レンタル会社によっては、キャンペーンや割引を実施している場合もありますので、事前に問い合わせしてみてください。

また、レンタル期間が長くなればなるほど、1日あたりのレンタル費用は安くなる傾向にあります。

Model B (Right):

書籍や映画などのメディアコンテンツであれば、オンラインストリーミングサービスや書店での購入が考えられます。

2. レンタルサービス:

- もし「h100」がレンタル可能な商品やサービスである場合、レンタルショップやオンラインレンタルサービスを利用することができます。この場合、レンタル期間や料金体系はサービス提供者によって異なります。

3. イベントチケットや体験型サービス:

- コンサートやイベントのチケットであれば、公式ウェブサイトやチケット販売サイトで購入手続きを行います。
- 体験型サービスやアクティビティの場合、予約が必要な場合が多く、その場合の予約方法や料金も確認が必要です。

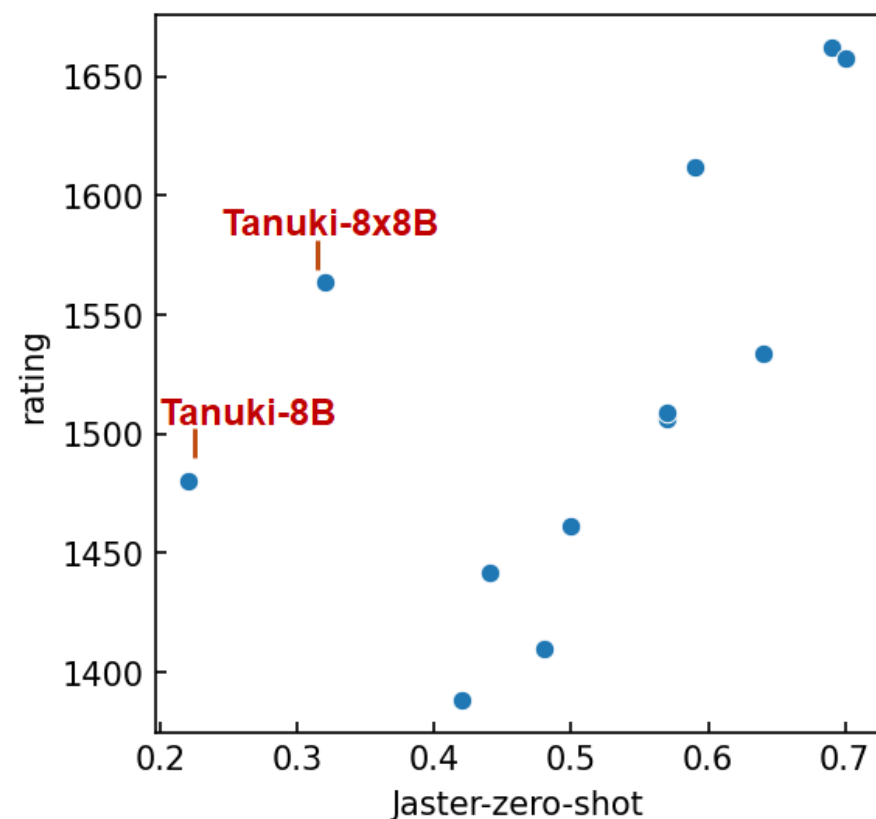
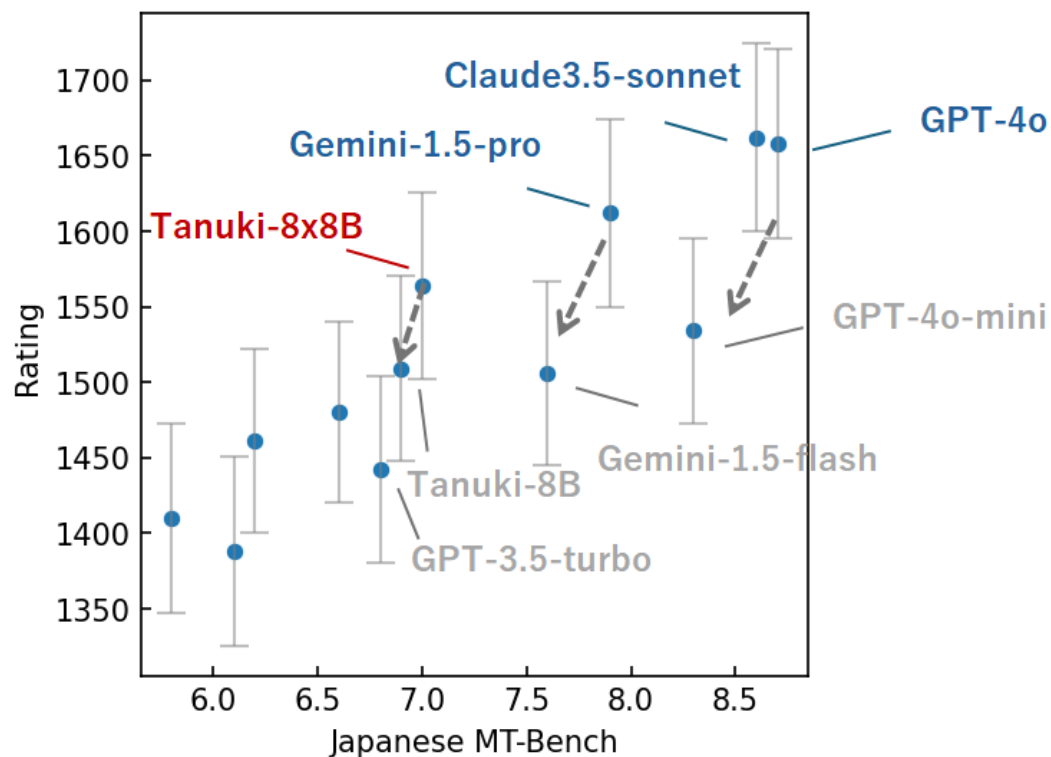
ブラインドテストの結果

	モデル名	モデル重 みの公開	レーティング	勝率	対戦回数	Japanese MT-Bench平均点	Nejumi leader board 3総合点
OpenAI	GPT-4o-2024-05-13	非公開	1178	0.57	297	8.6	0.78
Anthropic	Claude-3.5-sonnet-20240620	非公開	1173	0.56	272	8.7	0.82
Google	Gemini-1.5-pro	非公開	1128	0.50	208	7.9	0.73
構築したモデル	Tanuki-8x8B-dpo-v1.0	オープン	1099	0.41	382	7.0	0.57
OpenAI	GPT-4o-mini-2024-07-18	非公開	1086	0.41	303	8.3	0.72
Google	Gemini-1.5-flash	非公開	1070	0.41	318	7.6	0.70
サイバーエージェント	Calm3-22b-chat	オープン	1067	0.36	392	6.9	0.65
Preferred Elements	PLAMO-100B	非公開	1002	0.29	297	(報告無し)	(報告無し)
構築したモデル	Tanuki-8B-dpo-v1.0	オープン	1000	0.28	379	6.6	0.52
Karakuri	Karakuri-lm-8x7b-chat-v0.1	オープン	948	0.21	329	5.8	0.60
Elyza	Llama-3-ELYZA-JP-8B	オープン	945	0.17	282	6.1	0.62
Swallow	Llama-3-Swallow-70B-Instruct-v0.1	オープン	906	0.16	348	6.2	0.65
OpenAI	GPT-3.5-turbo	非公開	893	0.17	291	6.8	0.58

Tanukiの課題

対話性能とベンチマーク性能の相関

- Japanese MT-Benchと対話性能は良い相関を示した
- Jaster系と対話性能はTanukiを除いて良い相関を示した



対話・作文に特化したTanukiは定型回答が苦手

Tanuki-8Bの出力

出力指定

以下に、あるタスクを説明する指示があり、それに付随する入力が更なる文脈を提供しています。リクエストを適切に完了するための回答を記述してください。

質問と回答の選択肢を入力として受け取り、選択肢から回答を選択してください。なお、回答は選択肢の番号(例:0)とするものとします。回答となる数値をint型で返し、他には何も含めないことを厳守してください。

質問: スキー場などにある、輸送機器の一つで、高所低所に張られたケーブルにアームが付き、そこにむき出しの椅子を据えたものをなんというか？

選択肢: 0.北海道大学, 1.ポプラ並木, 2.ベンチ, 3.パイプ椅子, 4.リフト

回答: 3 (誤り)

自由回答 (正解)

 Chatbot

質問: スキー場などにある、輸送機器の一つで、高所低所に張られたケーブルにアームが付き、そこにむき出しの椅子を据えたものをなんというか？

選択肢: 0.北海道大学, 1.ポプラ並木, 2.ベンチ, 3.パイプ椅子, 4.リフト

スキー場などにある輸送機器で、高所低所に張られたケーブルにアームが付き、そのアームにむき出しの椅子が据えられているものを指す言葉は「リフト」です。リフトはスキーやスノーボードを運ぶために使用され、特に山岳地帯での移動を容易にするために設置されています。したがって、正しい選択肢は4の「リフト」です。北海道大学やポプラ並木、ベンチ、パイプ椅子といった他の選択肢は、この文脈には当てはまりません。

記憶定着の難しさ

Tanukiが「のび太の特技」を間違えるのはなぜか

- 野比のび太の3大特技:「射撃」、「あやとり」、「昼寝」([wikipedia](#))
- 「昼寝」を誤答

のび太の特技は？



のび太の特技についてお話しします。のび太は、ドラえもんの登場人物であり、その特技は物語の中でさまざまな形で描かれています。以下にいくつかの代表的な特技を紹介します。

1. 射撃の腕前:

のび太は射撃の名手として知られています。特に、ドラえものの道具「ショックガン」や「空気砲」を使って、敵を撃退するシーンが多く見られます。

2. あやとり:

のび太はあやとりを非常に得意としています。彼は複雑なあやとりのパターンを短時間で作り上げることができ、その技術はしばしば仲間たちを驚かせます。

3. あやとりの知識:

のび太はあやとりに関する豊富な知識を持っており、新しいパターンや技をすぐに習得することができます。これは彼の学習能力の高さを示しています。

4. 料理の腕前:

のび太は料理も得意で、特に簡単な料理や即席の食事を作るのが上手です。彼の料理はしばしば仲間たちに提供され、その味は好評です。

5. 音楽の才能:

のび太はピアノを弾くのが得意で、特に即興演奏が得意です。彼は音楽を通じて感情を表現することができ、その演奏は感動を呼びます。

mc-4jaを全文解析

- 10 B程度のモデルが何かをきちんと覚えるには、事象毎に1000件ほどの異なるテキストが必要(?)
 - [Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws](#)
 - のび太の特技を覚える上で、数が全く足りていなかった
- 知識の網羅的な取得のためには、計画的な合成データの生成と学習が必要？

mc4-jaをキーワードで全文検索した結果

条件	件数	
全体	87425304	
のび太	76785	
のび太ー特技	1876	
のび太ー特技ー射撃	356	
のび太ー特技ー射撃(±70文字以内)	136	※100件以上はコピペ文章
のび太ー特技ーあやとり	287	
のび太ー特技ーあやとり(±70文字以内)	151	※100件以上はコピペ文章
のび太ー特技ー昼寝	212	
のび太ー特技ー昼寝(±70文字以内)	14	

数学・論理・コードは難しい

Japanese MT-Benchでスコアが伸びにくかった領域

Tanukiは50 B token以上の特化データを学習させたが、中学数学で躓いた

dataset	モデル	概要	tokens (b)
logical-wizardlm-7b	WizardLM-7b	論理・コード・数学系の合成指示データ(英語)	29
0804calm3-logical-multiturn-pretrain	Calm3	論理・コード・数学系の合成指示データ	7.1
logical-wizardlm-7b-ja-0805	Calm3	論理系データのCalm3による日本語訳	5.6
logical-wizardlm-7b-ja	Calm3	論理系データのCalm3による日本語訳	4.2
logical-wizardlm-7b-ja-0730	Calm3	論理系データのCalm3による日本語訳	2.8
SyntheticTextOpenMathInstruct	Phi-3	OpenMathInstructの再生成	2.0
logical-wizardlm-7b-ja-0731	Calm3	論理系データのCalm3による日本語訳	1.5
logicaltext-wizardlm8x22b-Ja	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.69
wizardlm8x22b-logical-math-coding-sft	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.51
Synthetic-JP-EN-Coding-Dataset-453k	各種	コードの合成データ	0.42
wizardlm8x22b-logical-math-coding-sft_additional	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.39
logicaltext-wizardlm8x22b	wizardlm8x22b	論理・コード・数学系の合成テキスト	0.32
wizardlm8x22b-logical-math-coding-sft-ja	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.19
LogicalDatasetsByMixtral8x22b	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.064
wizardlm8x22b-logical-math-coding-sft_additional-ja	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.060
OpenMathInstruct-ja-phi3	Phi-3	OpenMathInstructの再生成	0.055
logicaltext-wizardlm8x22b-api	wizardlm8x22b	論理・コード・数学系の合成指示データ	0.019

中学レベルの数学の問題を解かせる

(詳細は関連Slideを参照)

問1: 抽象概念 (石橋 一昂「モデル化の視点からみた中学生の確率の意味理解に関する考察」教育数学研究 2020, 26, 73. より一部改変)

1の目が出る確率が $\frac{1}{6}$ であるサイコロがあります。

このサイコロの性質について、正しいものを選びなさい。

1. 5回投げて、1の目が1回もで買ったとすれば、次に投げると必ず1の目が出る
2. 6回投げると、そのうち1回は必ず1の目が出る
3. 6回投げると、1から6までの目が必ず1回出る
4. 30回投げると、そのうち1の目は必ず5回出る
5. 3000回投げると、1の目はおよそ500回出る

問2: 組み合わせの列挙 (Japanese MT-Benchより改変)

サイコロを2つ振って、目の合計が3以上になる確率

問2a: 組み合わせの列挙 (ゴリ押しでの解法の強制)

サイコロを2つ振って、目の合計が3以上になる確率について、ありうる組み合わせをすべて列挙しながら求めよ。

問2b: 組み合わせの列挙 (余事象の利用の強制)

サイコロを2つ振って、目の合計が3以上になる確率を計算しなさい。

回答には以下の過程を踏みなさい。

1. すべての組み合わせを掛け算で推定する。
2. 目が2になる組み合わせを求める
3. 1の組み合わせから2の組み合わせを引く

回答結果

本当に中学数学の問題が解けるモデルは、
フロンティアモデル+αに限られる

Arenaの 暫定順位	備考	model	問1	問2	問2での 余事象の 利用	問2a	問2b
1	フロンティア	claude-3-5-sonnet-latest	1	1	1	1	1
2	フロンティア	gemini-1.5-pro-latest	1	1	1	1	1
3	準フロンティア	gpt-4o-mini-2024-07-18	1	1	1	0	1
4	フロンティア	gpt-4o-latest	1	1	1	1	1
5	準フロンティア	mistral-large-latest	1	0	0	0	1
6	準フロンティア	deepseek-chat	1	1	1	1	1
7	準フロンティア	gpt-4-turbo-2024-04-09	1	1	1	1	1
8	オープン(海外)	Qwen/Qwen2.5-72B-Instruct	1	1	1	1	1
9	オープン(海外)	nvidia/Llama-3.1-Nemotron-70B-Instruct	1	1	1	0	1
10	準フロンティア	gemini-1.5-flash-latest	1	1	1	0	1
11	オープン(海外)	google/gemma-2-27b-it	1	1	1	0	1
12	オープン(海外)	meta-llama/Meta-Llama-3.1-405B-Instruct	1	0	0	0	1
13	国産	Tanuki-8x8B-dpo-v1.0-AWQ	0	0	0	0	0
14	準国産	Llama-3.1-Swallow-8B-Instruct-v0.1-Q8_0	0	0	0	0	0
15	国産	cyberagent/calm3-22b-chat-BitsAndBytes	0	0	0	0	0
16	オープン(海外)	command-r-plus	1	0	0	0	0
17	国産	Tanuki-8B-dpo-v1.0-BitsAndBytes	0	0	0	0	0
18	オープン(海外)	NousResearch/Hermes-3-Llama-3.1-405B	1	1	0	1	0
19	準国産	Llama-3.1-Swallow-70B-Instruct-v0.1-BitsAndBytes	1	0	0	0	1
20	国産	llm-jp-3-13b-instruct-Q8_0.gguf	0	0	0	0	0
21	オープン(海外)	ibm-granite/granite-3.0-8b-instruct-fp8	0	0	0	0	0

1: 正解
0: 不正解

中学数学におけるモデルの分類

- **高精度のモデル群** (フロンティア3社、DeepSeek、Qwen-72B)
 - 余事象の概念を使わず、愚直に全ての組み合わせを数え上げてでも正答
 - ただしQwenは一定確率でミスをするケースも多かった
- **回答テクニックを習得した中精度のモデル群** (GPT-4o-mini, Gemini-1.5-flash, Llama-3.1-Nemotron-70b, Gemma-27b)
 - 計算量を削減するテクニックを習得済み(余事象の活用)
 - 愚直に組み合わせを数え上げる解法だと、途中で幻覚が生じて誤答
- **テクニックを知らない中精度のモデル群** (mistral-large, llama-3.1-400B, Swallow-70B)
 - 余事象のテクニックを教えれば正答
- **低精度のモデル群** (**Tanuki-8B**, **8x8B**, Swallow-8B, Calm3, Comman-r, llm-jp, granite)
 - 余事象のテクニックを教えても誤答

まとめ

- Tanukiの開発において、特に対話・作文ドメインでのデータ不足を補うため、合成データを多量に生成して学習に活用した。
- 合成データの活用は世界的なトレンドになっており、モデルの高性能化には必要不可欠な模様である。
- Tanukiの理科系能力はまだ中学レベルであり、フロンティアモデル(e.g., 熟考系)との差は非常に大きい。
- モデルの能力を底上げするための基礎学理&種々のノウハウの蓄積は継続して必要である。