日本語インストラクションデータを用いた対話可能な日本語大規模言語モデルのLoRAチューニング

鈴木　雅弘[†,††] 平野　正徳[†]　坂地　泰紀[†]

† 東京大学大学院工学系研究科　〒113–8656 東京都文京区本郷 7–3–1
E-mail: †b2019msuzuki@socsim.org, research@mhirano.jp, sakaji@sys.t.u-tokyo.ac.jp

あらまし　本研究では，日本語インストラクションデータを用い，日本語と英語のそれぞれをベースにした大規模言語モデル (LLM) に対して LoRA チューニングを行った．チューニングしたモデルに対し定量と定性による両面から評価を行い，日本語インストラクションデータによるチューニングの効果を確認した．また幅広いインストラクションデータや実際のモデルが出力した文字列による評価の必要性など，日本語における大規模言語モデルや言語資源における課題を明らかにした．

キーワード　大規模言語モデル，日本語，インストラクションチューニング

LoRA Tuning Conversational Japanese Large Language Models using Japanese Instruction Dataset

Masahiro SUZUKI[†,††], Masanori HIRANO[†], and Hiroki SAKAJI[†]

† School of Engineering, The University of Tokyo　7–3–1 Hongo, Bunkyo, Tokyo, 113–8656 Japan
E-mail: †b2019msuzuki@socsim.org, research@mhirano.jp, sakaji@sys.t.u-tokyo.ac.jp

Abstract　In this study, we performed LoRA tuning on large language models (LLM) based on both Japanese and English using Japanese instruction tuning and evaluated these models from both quantitative and qualitative perspectives. As a result of the evaluation, the effectiveness of tuning with Japanese instruction data was confirmed. Furthermore, we clarified the challenges in large-scale language models and language resources in Japanese, such as the need for evaluation using a wide range of instruction data and the actual output strings of the models.

Key words　Large Language Model (LLM), Japanese, Instruction Tuning

1. はじめに

大規模言語モデル (LLM) は，近年，著しい性能向上と汎化が進んでいる．特に，Transformer [1] ベースの大規模言語モデルである BERT [2] や RoBERTa [3]，GPT シリーズ [4]〜[6] を始めとして様々なモデルが事前学習由来の高い性能を発揮している．また 2022 年以降，モデルサイズをよりスケーリングさせより高い性能を示すモデルとして，OPT [7]，GPT-NeoX-20B [8]，UL2 [9]，PaLM [10]，BLOOM [11]，Pythia [12]，LLaMA [13] など非常に多くのモデルが登場している．

言語モデルが乱立している一方で，多様な指示の入力への対応にはまだまだ難しさがある．そんな中，事前学習を行ったこれらの LLM の能力を特定の目的に応じて更に適応させるための取り組みがなされている．LLM を対話形式で活用するためのアプローチとしてインストラクションチューニング (Instruction tuning) [14] がある．インストラクションチューニングは，自然言語で様々なタスクを解かせることで，未知の課題の性能を向上させる学習方法である．これは特定のタスクで学習し，そのタスクでの推論精度を向上させる Finetuning とは異なる．公開されているモデルでは，Dolly [15] は Pythia に，Vicuna [16] や Alpaca [17] は LLaMA に対してインストラクションチューニングを施したモデルとなっている．しかしながら，これらのモデルでは，英語以外の言語への対応は不完全である．上述の Dolly や Alpaca, Vicuna でのインストラクションチューニングを行うデータセットは英語のみであり，英語以外の言語でこれらのモデルのメリットを享受することは難しい．

本研究では，日本語ベース，英語ベースの大規模言語モデルに対してそれぞれインストラクションチューニングを行う．さらに，構築したモデルに対し，日本語のデータセットでの評価を行う．構築した学習済みモデルや本研究で用いた実装はオー

(注††): 責任著者

プンソースとして公開している．

- 学習や評価のための実装: https://github.com/retarfi/jallm
- チューニング済みモデル (stormy 10 epochs)：https://huggingface.co/izumi-lab/stormy-7b-10ep
- チューニング済みモデル (LLaMA 7B 5 epochs): https://huggingface.co/izumi-lab/llama-7b-japanese-lora-v0-5ep

詳細については，後述する．

2. モデルのインストラクト LoRA チューニング

本研究では，公開されている大規模言語モデル (LLM) に対し，日本語のデータセット [18] を用いチューニングを行う．

2.1 使用モデル

日本語ベースのモデルと英語ベースのモデルの 2 つを用いる．日本語ベースのモデルとして，CyberAgent 社が公開している OpenCALM-7B(注2) (以下 CALM) を用いる．CALM は日本語の Wikipedia と Common Crawl を GPT-NeoX [19] のアーキテクチャで事前学習した 70 億のパラメータを持つモデルとなっている．英語ベースのモデルとして，Meta が公開している(注3) LLaMA [13] の 7B モデル (以下 LLaMA 7B) を用いる．

2.2 使用データセット

日本語のインストラクションチューニングのためのデータとして llm-japanese-dataset v0 [18] を用いる．日本語ベースの CALM と英語ベースの LLaMA 7B のそれぞれの性質の違いから，CALM に対しては当該データセットの一部を，LLaMA 7B に対しては当該データセットの全てを学習データとして用いる．

llm-japanese-dataset v0 には約 840 万件の instruction データが含まれるものの，そのうち 75 % 以上を占める 6,581,044 件が翻訳データをベースに構築されたものである．これは，LLaMA を含む多くの LLM が英語において良いパフォーマンスを示していることを念頭に，英語と日本語の言語をリンクすることで英語で学習されている知識を日本語でも引き出すことを目的としている．一方で CALM の事前学習で用いられているコーパスは日本語部分であるため，英語と日本語の言語のリンクを目的としたこのデータセットの必要性は相対的に低い．そこで，CALM には llm-japanese-dataset v0 のうち翻訳タスクを除いた 1,809,964 個のデータを用いて学習を行う．

CALM の学習時に使用するフォーマットは下記のとおりである．

入力がある場合のフォーマット

以下はタスクを説明する指示とさらなる文脈を適用する入力の組み合わせです。要求を適切に満たすような返答を書いてください。

指示:
{Instruction}

入力:
{Input}

返答:
{Resoponse}

入力が無い場合のフォーマット

以下はタスクを説明する指示です。要求を適切に満たすような返答を書いてください。

指示:
{Instruction}

返答:
{Resoponse}

LLaMA 7B に対しては LLaMA 13B モデルをチューニングした文献 [18] と同様に，当該データセットの全てを用いて学習を行う．入力フォーマットについても文献 [18] と同じものを用いる．

2.3 LoRA チューニング

パラメータ数の多い LLM モデルは，事前学習に限らずファインチューニングにおいても GPU リソースを必要とする．本研究では，精度を大きく下げないまま大規模モデルをファインチューニングするための手法として LoRA [20] を用いる．LoRA では，LLM のパラメータの差分のみを小規模パラメータで更新する．実験の際に使用した主なパラメータを表 1 に示す．比較のため，LLaMA 13B をチューニングしたモデル [18] についても併記している．

実装には PEFT [21], DeepSpeed ZeRO 2 [22] を用いた．コードは https://github.com/retarfi/jallm にて公開している．

3. 構築したモデルの評価

チューニングしたモデルに対し，定量評価と定性評価を行う．定量評価では，2 つの観点から評価実験を行う．1 つ目はドメイン外の Question Answering データに対する Perplexity である．2 つ目は JNLI と MARC-ja によるテキスト分類タスクでの，選択肢の尤度から求める Accuracy である．定性評価ではいくつかのプロンプトに対する出力を定性的に評価する．生成の温度パラメータは 0.0，繰り返しのペナルティ [23] は CALM と stormy で 1.05，Instruct LLaMA 7B と LLaMA 7B で 1.0 とする．入力に使用するプロンプトは文献 [18] と同一のものを使用する．

3.1 Perplexity

Perplexity [24] は負の対数尤度の平均の指数表現として定義される．データセット内の単語が正しく出力される確率が高い

(注2)：https://huggingface.co/cyberagent/open-calm-7b
(注3)：厳密には，当初はオープンソースではなかったが，一定のライセンス下で提供されるようになった

表 1　LoRA チューニングのパラメータ.

パラメータ	stormy	Instruct LLaMA 7B	Instruct LLaMA 13B [18]
ベースモデル	CALM-7B	LLaMA-7B	LLaMA-13B
学習率	3e-4	3e-4	3e-4
入力長	300	256	256
バッチサイズ	128	128	130
データ数	約 140 万	約 840 万	約 840 万
エポック数	10	5	1
LoRA の r	4	4	4
LoRA の α	16	16	16
LoRA の Dropout 率	0.05	0.05	0.05
チューニングパラメータ	query_key_value	q_proj, v_proj	q_proj, v_proj

表 2　評価実験の結果．* は評価データセットに LoRA チューニングの入力長 (stormy は 300, Instruct LLaMA 7B と Instruct LLaMA 13B は 256) を超えたデータが存在したことを示す．† は評価データセットにモデルの最大入力長 (CALM ベース，LLaMA ベースともに 2,048) を超えたデータが存在したことを示す．タスクごとに最も性能が高い箇所を太字で示している．

	VQA	JNLI			MARC-ja		
Model	Perplexity	1-shot	2-shot	3-shot	1-shot	2-shot	3-shot
stormy (Instruct CALM)	**29.9**	**0.459**	**0.508**	0.475	0.468	0.828*	0.784*
CALM(注4)	246.6	0.294	0.331	0.314	0.781	0.836	**0.856**
Instruct LLaMA 7B	68.5	0.398*	0.454*	**0.479***†	0.795*	0.829*	0.847*
LLaMA 7B [13]	1,499	0.171	0.273	0.303†	0.839	0.848	0.852
Instruct LLaMA 13B [18]	38.8	0.302*	0.302*	0.302*†	**0.859***	**0.855***	0.855*
LLaMA 13B [13]	971.5	0.316	0.281	0.263†	0.855	**0.855**	0.855

ほど低い値となる．トークン化された配列 $X = (x_0, x_1, \cdots, x_t)$ があるとき，X の Perplexity は式 (1) によって表される．

$$\text{Perplexity}(X) = \exp\left\{-\frac{1}{t}\sum_{i}^{t}\log p_\theta(x_i|x_{<i})\right\} \qquad (1)$$

ここで $\log p_\theta(x_i|x<i)$ は先行するトークン $x_{<i}$ を条件とする i 番目のトークンの対数尤度である．

本研究では，言語モデルのチューニングで使用した llm-japanese-dataset v0 に含まれない Japanese Visual Question Answering (VQA) dataset [25] を用いて Perplexity の計測を行う．本 VQA データセットは提示された画像を見て行う質問応答タスクであるものの，正解となる応答文を予測する確率が高いモデルはより自然なモデルであると推察される．VQA データセットから抽出された 793,664 件の質問応答をプロンプト形式に変換して入力する．以下に入力の例を示す．

日本語ベースモデルでの VQA による入出力の例

以下の質問に答える返答を書いてください。

質問:
飛行機の機体は何色ですか

返答:
白色

なお，LLaMA ベースのモデルはシステムメッセージに英語を用いている．そのため，文献 [18] に従い上記の例は以下のように変更する．

英語ベースのモデルでの VQA による入出力の例

Write a response to answer the following question.

Question:
飛行機の機体は何色ですか

Response:
白色

Perplexity の計算はモデルへの入力では行わず，応答に対してのみ適用する．つまり，上記の例では出力が「白色」にあたるトークンの箇所についてのみ Perplexity を算出する．

3.2　JNLI・MARC-ja

もう 1 つは JGLUE [26] に含まれる JNLI と MARC-ja による評価である．JNLI は前提文が仮説文の文ペアに対し示す関係を entailment (含意)，contradiction (矛盾)，neutral (中立) の 3 つから選ぶタスクである．MARC-ja は商品レビューに対し「ポジティブ」と「ネガティブ」の 2 つから選択するタスクで，Multilingual Amazon Reviews Corpus (MARC) [27] の日本語部分を用いて構成される．JGLUE にはこれらの他に，常識を問

う JCommonsenseQA や抜き出しタスクである JSQuAD も含むが，言語モデルのチューニングで使用した llm-japanese-dataset v0 にこれらのデータが含まれているため，評価タスクとしては不適切と判断し除外した．

実験の実装には，Stability-AI/lm-evaluation-harness [28] の日本語評価用のブランチ(注5)を用いる．モデルに入力するプロンプトのバージョンについて，stormy では 0.2，それ以外の CALM，Instruct LLaMA 13B，LLaMA 13B では 0.3 を用いる．詳細なプロンプトは付録 1. 節に記載する．

入力プロンプトに対し，タスクのそれぞれの選択肢を出力する尤度を比較し，最も高いものをモデルの出力とする．つまり，JNLI では entailment, contradiction, neutral の 3 つ，MARC-ja では「ポジティブ」と「ネガティブ」の 2 つが選択肢であり，モデルはこれらの選択肢を出力する尤度が最も高いものを出力とする．そのため選択肢以外の出力が考慮されることはない．入力中で例を 1 つ，2 つまたは 3 つ示す 1-shot，2-shot，3-shot についてそれぞれ評価を行う．

4. 結果と考察

4.1 定量評価

評価実験の結果を表 2 に示す．VQA を用いた Perplexity の評価では，CALM，LLaMA 7B，LLaMA 13B の全てのモデルにおいて，インストラクションデータを用いたチューニングにより Perplexity が下がり性能が向上した．特に LLaMA ベースのモデルでの Perplexity の改善が顕著で，英語という日本語以外のモデルに対しても翻訳データを含むインストラクションデータを用いて学習を行うことで，日本語とのリンクが生まれ性能が向上したと考えられる．6 つのモデルの中で最も Perplexity が高く性能が悪かったのは LLaMA 7B であった．英語ベースのモデルかつ LLaMA 13B よりもモデルのパラメータ数が少ないことによるものであると考えられる．一方最も Perplexity が低く良い性能を発揮したのは stormy となった．日本語をモデルとした CALM に対して更にインストラクションチューニングを行うことで性能が向上したと考えられる．チューニングを行うベースとなった CALM，LLaMA 7B，LLaMA 13B を比較すると，日本語ベースの CALM が最も性能が高かった．

JNLI による評価では，1-shot，2-shot，3-shot の全てにおいて stormy の精度が最も高かった．llm-japanese-dataset v0 には含意関係認識に相当するデータセットは含まないものの，文献 [14] と同様，様々なタスクを解かせることで性能が向上したと考えられる．CALM と LLaMA 7B にインストラクションチューニングを行った stormy と Instruct LLaMA 7B はそれぞれ性能が向上したことから，Perplexity の考察と同様にインストラクションチューニングの効果が示された．一方で LLaMA 13B でのインストラクションチューニングの効果は比較的小さくなった．これは Instruct LLaMA 13B でのインストラクションチューニングが 1 epoch しか行われなかったことによると考えられる．より多くの学習を行うことで性能が向上する可能性がある．

MARC-ja による評価では 1-shot，2-shot，3-shot の全てでインストラクションチューニングによる性能向上がない，または性能が悪化する結果となった．このような現象は文献 [14], [29] でも報告されている．FLAN [14] のように，インストラクションデータとして様々なタスクを幅広く採用することで改善されると考えられる．chABSA-dataset(注6)など，日本語で組み込みうるセンチメントに関連したデータセットも存在しており，MARC-ja のセンチメントタスクを学習しうることから，このようなデータセットを追加で学習することで精度の低下を抑えられる可能性がある．また，LLaMA ベースのモデルで性能が向上しなかった理由としては，本研究でのインストラクションチューニングの入力長が関係している可能性がある．LLaMA ベースのモデル自体は 2,048 トークンまで入力可能であり，事前学習もこの長さで行われているが，本研究では入力長を 256 トークンに制限している．そのため，長いトークンが入力となるデータではインストラクションチューニングの効果が発揮できなかった可能性がある．入力長を伸ばしたインストラクションチューニングは今後の課題である．

本研究で評価を実施したタスクは，ある特定の出力に対してモデルがどれくらい出力しうるかについて評価しており，実際のモデルの出力は考慮していない．より精緻な比較のために，実際の出力を用いた評価を行うことが今後の課題と考えられる．また，タスクの正誤だけでなく人間の好みの観点からの評価 [30], [31] を行うことも将来的に必要になるであろう．

4.2 定性評価

以下に，構築したモデルの応答を示す．なお，生成の温度パラメータは 0.0，繰り返しのペナルティ [23] は CALM と stormy で 1.05，Instruct LLaMA 7B と LLaMA 7B で 1.0 とする．入力に使用するプロンプトは文献 [18] と同一のものを使用している．Instruct LLaMA 13B と LLaMA 13B の応答については，文献 [18] を参照されたい．

応答の比較 (1)

入力:
あなたは誰ですか？
出力 (stormy):
私は、私の妻であり、私は彼女を愛しています。
出力 (CALM):
私は、Nick M. Kelemel (ニック・ケレムエル) です。私は、コンピュータサイエンスと数学の学位を取得しています。そして、現在、研究者として働いています。
出力 (Instruct LLaMA 7B):
あなたは何をしていますか？
出力 (LLaMA 7B):
あなたは誰ですか？

(注5)：https://github.com/Stability-AI/lm-evaluation-harness/tree/jp-stable

(注6)：https://github.com/chakki-works/chABSA-dataset

応答の比較 (2)

入力:
東京から大阪の行き方を教えてください。
出力 (stormy):
飛行機で東京から出発します。大阪に到着したら、新幹線に乗り換えて名古屋に行きます。名古屋からは近鉄特急に乗ります。
出力 (CALM):
電車と新幹線、どちらがいいですか?
*「電車」は、JR の電車や私鉄の電車など、あらゆる交通機関を指しています。
*「新幹線」は、東京 新大阪間を走る東海道・山陽新幹線をさします。
*「飛行機」は、羽田 伊丹間を指すのが一般的です。
*「バス」は、東京都内を運行する都営バスをさします。
*「車」は、自家用車だけでなく、タクシーやレンタカーで移動する場合も含みます。
*「自転車」は、電動アシスト付自転車も含まれます。
*「徒歩」「歩き」「ジョギング」「散歩」などは除きます。
出力 (Instruct LLaMA 7B):
大阪から東京の行き方を教えてください。
出力 (LLaMA 7B):
東京から大阪の行き方を教えてください。

応答の比較 (3)

入力:
東京大学はどこにありますか？
出力 (stormy):
東京大学は、東京都文京区本郷にある国立大学です。
出力 (CALM):
東大は、東京都文京区本郷 7 丁目 3-1 にあります。東大の緯度経度は 35.676044, 135.869422 です。

##回答: その住所に行くには何を使いますか? (地図や Google Map など)

##説明: その住所へ行くためにどの交通機関を使ったらいいですか?

##回答: 電車とバスはどちらを使ったらよいですか?

##説明: JR 線を使うなら「山手線」で降りてください。JR 線は『東京』駅に行きます。

##回答: 東京駅からどこに行けばいいですか?

##説明: 「東京駅」から乗るのは、「東海道新幹線」「東北・上越・北陸新幹線」そして「特急ひたち」です。

##回答: 常磐線でいくのと、新幹線でいくのではどちらが速くて安いですか?

##説明: 新幹線の方が早いです。ただ、値段も違います。

##回答: お金がないですが、どうしたらいいですか?

##説明: まずはお金を用意してください。それから、クレジットカードがあれば使えます。

##回答: どうやって払えばいいですか?

##説明: カード払いできます。VISA カード
出力 (Instruct LLaMA 7B):
東京大学は東京都にあります。
出力 (LLaMA 7B):
東京大学はどこにありますか？

応答の比較 (4)

入力:
電話が鳴りました。どうするべきですか？
出力 (stormy):
あなたは電話を取るべきですか?それとも保留にするべきですか?
出力 (CALM):
保留ボタンを押して、電話に出るか、あるいは留守番電話サービスに電話してください。着信音が聞こえたら、10 秒以内に答えを返信してください。もし、応答がなかったら、もう一度かけてみてください。それでもだめだったら、次の手順に進んでください。
* 「0」を押して通話を終了します。
* 呼び出し音の後で「*」を押します。これで通話が可能です。
出力 (Instruct LLaMA 7B):
電話が鳴りました。どうするべきですか？
出力 (LLaMA 7B):
電話が鳴りました。どうするべきですか？

応答の比較 (5)

入力:
京都の三大祭りと言えば何？
出力 (stormy):
葵祭、祇園祭、時代祭
出力 (CALM):
祇園祭、葵祭、時代祭の 3 つです。京都三大祭りとも言われます。

#N/A
#N/A
(筆者注: 以降「 #N/A」が続く．)
出力 (Instruct LLaMA 7B):
京都の三大祭りと言えば何？
出力 (LLaMA 7B):
これは、京都の三大祭りと言えば何？

CALM からインストラクションチューニングを行った stormy は，一方で Instruct LLaMA 7B と LLaMA 7B の差分はわずかで，応答の比較 (3) の精度が若干向上した程度である．文献 [18] において LLaMA 13B の 1 epoch チューニングでの改善が少なかった理由として，学習量が少ないことが挙げられていた．しかし LLaMA 7B に 5 epoch のチューニングを行った Instruct LLaMA 7B でも大きな改善が見られなかったことから，モデルサイズや特に日本語についての事前学習の量や質など，インストラクションチューニングだけでは大きな改善が見込めない可能性がある．日本語単体で事前学習を行うのが良いのか，BLOOM [11] のように英語を始めとした他言語も含めて学習を行うのが良いかなど，英語以外で高い性能を示すための手法の調査が今後の課題であろう．

5. ま と め

本研究では，日本語インストラクションデータを用い日本語と英語のそれぞれで事前学習された大規模言語モデルに対して LoRA チューニングを行った．チューニングを行ったモデルに対し定量・定性の両面から評価を行った．日本語インストラクションデータでのチューニングによって定量評価での性能が向上することを確認した．また，今回のインストラクションデータによるチューニングと評価を通じて，日本語の大規模言語モデルの構築における課題も明らかとなった．

謝　　辞

本研究は JSPS 科研費 JP21K12010 および JST さきがけ JPMJPR2267 の助成を受けたものです．

文　　献

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin, “Attention Is All You Need,” Advances in Neural Information Processing Systems, vol.30, pp.5999–6009, 2017.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, pp.4171–4186, Association for Computational Linguistics, 2019.

[3] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, and P.G. Allen, “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” 2019. https://arxiv.org/abs/1907.11692

[4] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving Language Understanding by Generative Pre-Training,” 2018. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

[5] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language Models are Unsupervised Multitask Learners,” 2019. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[6] T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, “Language Models are Few-Shot Learners,” Advances in Neural Information Processing Systems, vol.33, pp.1877–1901, 2020.

[7] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X.V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P.S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer, “OPT: Open Pre-trained Transformer Language Models,” 2022. https://arxiv.org/abs/2205.01068

[8] S. Black, S. Biderman, E. Hallahan, Q. Anthony, L. Gao, L. Golding, H. He, C. Leahy, K. McDonell, J. Phang, M. Pieler, U.S. Prashanth, S. Purohit, L. Reynolds, J. Tow, B. Wang, and S. Weinbach, “GPT-NeoX-20B: An open-source autoregressive language model,” Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models, pp.95–136, Association for Computational Linguistics, 2022. https://aclanthology.org/2022.bigscience-1.9

[9] Y. Tay, M. Dehghani, V.Q. Tran, X. Garcia, J. Wei, X. Wang, H.W. Chung, D. Bahri, T. Schuster, S. Zheng, D. Zhou, N. Houlsby, and D. Metzler, “UL2: Unifying Language Learning Paradigms,” The Eleventh International Conference on Learning Representations, pp.••–••, 2023. https://openreview.net/forum?id=6ruVLB727MC

[10] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H.W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A.M. Dai, T.S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, “PaLM: Scaling Language Modeling with Pathways,” 2022. https://arxiv.org/abs/2204.02311

[11] T.L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A.S. Luccioni, F. Yvon, M. Gallé, et al., “BLOOM: A 176B-Parameter Open-Access Multilingual Language Model,” 2022. https://arxiv.org/abs/2211.05100

[12] S. Biderman, H. Schoelkopf, Q. Anthony, H. Bradley, K. O’Brien, E. Hallahan, M.A. Khan, S. Purohit, U.S. Prashanth, E. Raff, A. Skowron, L. Sutawika, and O. van derWal, “Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling,” 2023. https://arxiv.org/abs/2304.01373

[13] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., “LLaMA: Open and Efficient Foundation Language Models,” 2023. https://arxiv.org/abs/2302.13971

[14] J. Wei, M. Bosma, V. Zhao, K. Guu, A.W. Yu, B. Lester, N. Du, A.M. Dai, and Q.V. Le, “Finetuned language models are zero-shot learners,” International Conference on Learning Representations, pp.••–••, 2022. https://openreview.net/forum?id=gEZrGCozdqR

[15] Databricks, “Dolly,” https://github.com/databrickslabs/dolly, 2023.

[16] Vicuna, “Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90% ChatGPT Quality,” https://vicuna.lmsys.org/, 2023.

[17] R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, and T.B. Hashimoto, “Stanford Alpaca: An Instruction-following LLaMA model,” https://github.com/tatsu-lab/stanford_alpaca, 2023.

[18] 平野正徳，鈴木雅弘，坂地泰紀，“llm-japanese-dataset v0: 大規

模言語モデルのための日本語チャットデータセット構築," 2023. https://doi.org/10.51094/jxiv.383
[19] S. Black, S. Biderman, E. Hallahan, Q. Anthony, L. Gao, L. Golding, H. He, C. Leahy, K. McDonell, J. Phang, M. Pieler, U.S. Prashanth, S. Purohit, L. Reynolds, J. Tow, B. Wang, and S. Weinbach, "GPT-NeoX-20B: An open-source autoregressive language model," Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models, pp.95–136, Association for Computational Linguistics, 2022. https://aclanthology.org/2022.bigscience-1.9
[20] E.J. Hu, yelongshen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-Rank Adaptation of Large Language Models," International Conference on Learning Representations, pp.1–13, 2022. https://arxiv.org/abs/2106.09685
[21] S. Mangrulkar, S. Gugger, L. Debut, Y. Belkada, and S. Paul, "PEFT: State-of-the-art Parameter-Efficient Fine-Tuning methods," https://github.com/huggingface/peft, 2022.
[22] S. Rajbhandari, J. Rasley, O. Ruwase, and Y. He, "ZeRO: Memory Optimizations toward Training Trillion Parameter Models," SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, pp.1–16, 2020.
[23] N.S. Keskar, B. McCann, L.R. Varshney, C. Xiong, and R. Socher, "CTRL: A conditional transformer language model for controllable generation," 2019. http://arxiv.org/abs/1909.05858
[24] F. Jelinek, R.L. Mercer, L.R. Bahl, and J.K. Baker, "Perplexity—a measure of the difficulty of speech recognition tasks," The Journal of the Acoustical Society of America, vol.62, no.S1, pp.S63–S63, 1977.
[25] N. Shimizu, N. Rong, and T. Miyazaki, "Visual question answering dataset for bilingual image understanding: A study of cross-lingual transfer using attention maps," Proceedings of the 27th International Conference on Computational Linguistics, pp.1918–1928, Association for Computational Linguistics, 2018. http://aclweb.org/anthology/C18-1163
[26] K. Kurihara, D. Kawahara, and T. Shibata, "JGLUE: Japanese General Language Understanding Evaluation," Proceedings of the Thirteenth Language Resources and Evaluation Conference, pp.2957–2966, 2022.
[27] P. Keung, Y. Lu, G. Szarvas, and N.A. Smith, "The multilingual amazon reviews corpus," Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pp.••–••, 2020.
[28] L. Gao, J. Tow, S. Biderman, S. Black, A. DiPofi, C. Foster, L. Golding, J. Hsu, K. McDonell, N. Muennighoff, J. Phang, L. Reynolds, E. Tang, A. Thite, B. Wang, K. Wang, and A. Zou, "A framework for few-shot language model evaluation," 2021. https://doi.org/10.5281/zenodo.5371628
[29] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Gray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," Advances in Neural Information Processing Systems, eds. by A.H. Oh, A. Agarwal, D. Belgrave, and K. Cho, pp.••–••, 2022. https://openreview.net/forum?id=TG8KACxEON
[30] B. Peng, C. Li, P. He, M. Galley, and J. Gao, "Instruction Tuning with GPT-4," 2023. https://arxiv.org/abs/2304.03277
[31] C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, A. Efrat, P. Yu, L. Yu, S. Zhang, G. Ghosh, M. Lewis, L. Zettlemoyer, and O. Levy, "LIMA: Less Is More for Alignment," 2023. https://arxiv.org/abs/2305.11206

付　　録

1. JNLI と MARC-ja で用いたプロンプト

JNLI で使用する 1-shot プロンプトの例 (v0.2)

前提と仮説の関係を entailment、contradiction、neutral の中から回答してください。

制約:
- 前提から仮説が、論理的知識や常識的知識を用いて導出可能である場合は entailment と出力
- 前提と仮説が両立しえない場合は contradiction と出力
- そのいずれでもない場合は neutral と出力

前提: 草地でフリスビーを取ろうと、二人の女性がジャンプしています。
仮説: 二人の女性がドーナツの載ったトレーを持っています。
関係:entailment

前提: 子供が 2 人いて、ミキサーの横に、バナナとキュウイが置いてあります。
仮説: ミキサーが置かれたテーブルにスポイトを持った子供たちがいます。
関係:

JNLI で使用する 1-shot プロンプトの例 (v0.3)

以下は、タスクを説明する指示と、文脈のある入力の組み合わせです。要求を適切に満たす応答を書きなさい。

指示:
与えられた前提と仮説の関係を回答してください。

出力は以下から選択してください：
entailment
contradiction
neural

入力:
前提：草地でフリスビーを取ろうと、二人の女性がジャンプしています。
仮説：女性がフリスビーを取ろうとしています。

応答:
entailment

(注6)：同一プロンプト中に同じ指示が繰り返されているものの，実装の参照元 (https://github.com/Stability-AI/lm-evaluation-harness/tree/jp-stable) のテンプレートをそのまま利用している．

指示:
与えられた前提と仮説の関係を回答してください。(注6)

出力は以下から選択してください：
entailment
contradiction
neural

入力:
前提：子供が 2 人いて、ミキサーの横に、バナナとキュウイが置いてあります。
仮説：ミキサーが置かれたテーブルにスポイトを持った子供たちがいます。

応答:

MARC-ja で使用する 1-shot プロンプトの例 (v0.2)

製品レビューを negative か positive のいずれかのセンチメントに分類してください。出力は小文字化してください。

製品レビュー: 私はカントリーが好きで当初 CD を購入していいなと思ったのです、映画はそれなりのストーリー、まずまずです
センチメント:positive

製品レビュー: 最後まで、楽しめました。個人的にはダンスのシーンがもっと見たかったですね。是非舞台化してほしいと思います。
センチメント:

MARC-ja で使用する 1-shot プロンプトの例 (v0.3)

以下は、タスクを説明する指示と、文脈のある入力の組み合わせです。要求を適切に満たす応答を書きなさい。

指示:
以下の製品レビューを、ポジティブまたはネガティブの感情クラスのいずれかに分類してください。

入力:
私はカントリーが好きで当初 CD を購入していいなと思ったのです、映画はそれなりのストーリー、まずまずです

応答:
ポジティブ

指示:
以下の製品レビューを、ポジティブまたはネガティブの感情クラスのいずれかに分類してください。(注6)

入力:
最後まで、楽しめました。個人的にはダンスのシーンがもっと見たかったですね。是非舞台化してほしいと思います。

応答: