日本語に特化した60億パラメータ規模のGPTモデルの構築と評価

麻場直喜　梅沢知紀　川村晋太郎
株式会社リコー
{naoki.asaba, tomoki.umezawa, shintaro.kawamura}@jp.ricoh.com

概要

GPT-3をはじめとするTransformerベースの大規模な事前学習済み言語モデルは，様々な下流タスクを高精度に解けることが報告されている．一方でこれらの大規模言語モデルの多くは英語を対象言語としており，日本語を対象言語とした検証はまだ少ない．本稿では，60億パラメータの日本語GPTモデル(Japanese-GPT-6B)の事前学習を行い，日本語言語理解ベンチマークJGLUEの質問応答タスク2種を用いて，in-context learningによるfew-shot性能評価を行った結果について報告する．

1 はじめに

近年，Transformer[1]をベースアーキテクチャとしたニューラル言語モデルの大規模化が活発に進められている．特にTransformerのdecoderのみを用いた言語モデルで，重みパラメータ数を大規模化かつ学習データ量を大規模化して事前学習を行ったGPT-3[2]やPaLM[3]などのモデルが，様々な下流タスクを高精度に解けることが報告されている．本稿では，これらdecoderに特化した大規模な言語モデルを，BERT[4]に代表されるencoderに特化したマスク言語モデルや，T5[5]に代表される系列変換モデルと区別する意味で，大規模生成系言語モデルと呼ぶこととする．なお，大規模か否かを分ける境界の定説はないが，本稿では重みパラメータ数が概ね数十億（十億＝Billion，以後‘B’と表記）以上のものを大規模と想定している．

大規模生成系言語モデルは，プロンプトと呼ばれるタスク説明文と少数の例文を与えるのみで，重みパラメータを更新することなく様々な言語タスクを高精度に解くことができる．この手法はin-context learning[2]やprompt-based learning[6]と呼ばれ，汎用性と高精度を両立した新パラダイムといえる．このパラダイムシフトはビジネス応用において重要であり，大規模生成系言語モデルへの入力プロンプトを最適化するプロンプトエンジニアリングや，出力文に対する後処理技術を含めた使いこなし方を確立できれば，様々なユースケースに対する素早い実用化が期待できる．

しかし，既存の大規模生成系言語モデルの多くは英語を対象としており，日本語を対象とした検証はまだ少ない．言語資源の観点からも，公開テキストデータ，公開事前学習済みモデル，公開ベンチマークデータセットなどはいずれも英語と比較して日本語では少ない．例えば英語モデルにおいては，公開の事前学習済みモデルであるGPT-J-6B[7]は6Bパラメータ，OPT[8]は175Bパラメータ，非公開の事前学習済みモデルであるPaLMは540Bパラメータである．一方で日本語モデルにおいては，公開の事前学習済みモデルとしては株式会社ABEJAの2.7Bパラメータ（gpt-neox-japanese-2.7b），非公開の事前学習済みモデルとしては同じく株式会社ABEJAの13Bパラメータが最大と思われる[9]．

本研究では，日本語に特化した大規模生成系言語モデルの実用化を目指して，6Bパラメータの日本語GPTモデル(Japanese-GPT-6B)を開発している．本稿では，日本語言語資源を用いた大規模生成系言語モデルの事前学習及び性能評価事例の共有を目的とし，日本語170Bトークンで事前学習して評価を行った結果について報告する．評価では日本語言語理解ベンチマークJGLUE[10]の質問応答タスク2種を用いて，in-context learningによるfew-shot学習性能を評価した．結果はGPT-J-6Bと比較して優位であり，日本語特化の効果を確認した．

2 モデル構築

2.1 モデル設計

本稿で報告するJapanese-GPT-6BはTransformerのdecoderに特化した自己回帰モデルであり，GPT-3

のアーキテクチャを踏襲している．重みパラメータ数は GPT-J-6B 同等の 6B に設定した．表 1 に主なモデルネットワーク仕様を示す．

トークナイザは，2.2 節で述べる日本語学習データの一部を用いて SentencePiece[i][11]のユニグラム言語モデルを学習して構築した．語彙数は約 5 万とし，SentencePiece の byte-fallback を有効にすることで未知語の発生を防いでいる．

2.2 学習データ

Japanese-GPT-6B の事前学習には以下 4 つの公開データの日本語版を用いた．

- Wikipedia：インターネット百科事典
- CC100，OSCAR，mC4：Web 上のテキストをスクレイピングした Common Crawl コーパスを言語分類及びクレンジングしたデータ

日本語版の概算データ容量は Wikipedia 7GB，CC100 70GB，OSCAR 100GB，mC4 800GB である．これらの学習データに対して，追加のクレンジング処理と，2.1 節で述べたトークナイザを用いたトークン化処理を行った．また，一部のデータを事前学習におけるバリデーション用の開発データとして分割し，最終的に学習データ量は合計 170B トークンとした．

2.3 学習環境

Japanese-GPT-6B の事前学習では 80 個の NVIDIA A100 40GB GPU を用いて分散学習を行った．学習を実行するプラットフォームは Amazon SageMaker を用いた．NVIDIA DGX A100 320GB のインスタンスに搭載された 8 個の A100 GPU にテンソルパラレルでモデルをロードし，そのインスタンスを 10 台用いてデータパラレルで並列化した．

学習コードは Amazon Web Services, Inc.が公開している分散学習向け GPT2 学習コードサンプル[ii]をベースとした．ディープラーニングのフレームワークは PyTorch である．

2.4 ハイパーパラメータ

表 2 に，Japanese-GPT-6B の事前学習における主なハイパーパラメータを示す．

表 1 Japanese-GPT-6B の主なネットワーク仕様

パラメータ	値
max_context_width	2,048
num_layers	28
hidden_width	4,096
num_head	16
vocab_size	50,272

表 2 Japanese-GPT-6B の事前学習における主なハイパーパラメータ

パラメータ	値
batch_size (tokens)	1M
num_epochs	1
learning_rate	1.2×10^{-4}

バッチサイズは GPT-J-6B と同様に 1M トークンとした．

学習コストの制約から，学習データ 170B トークンを一通り学習した時点で事前学習を終了した．すなわちエポック数を 1 とし，学習ステップ数は 170k ステップとなった．

学習率の最大値は GPT-3 6.7B モデルと同様に 1.2×10^{-4} とした．但し，GPT-3 6.7B モデルのバッチサイズが 2M トークンであることを考慮すると Japanese-GPT-6B はその 1/2 であるため，ノイズスケールは 2 倍となることに注意が必要である．

上記ハイパーパラメータにて事前学習 1 エポックの所要時間は 14 日であった．

3 評価

2 節で述べた事前学習を行った Japanese-GPT-6B に対して，下流タスク性能の評価を行った．主な比較対象は公開の大規模生成系言語モデルである GPT-J-6B とした．表 3 に各モデルの仕様を示す．Japanese-GPT-6B と GPT-J-6B のモデルサイズは同等で，主な違いは日本語特化の有無であり，日本語に特化することの効果を測る．

3.1 評価方法

本評価においては事前学習済みモデルのファインチューニングは行わず，プロンプトとしてタスク説明文と少数の例文を与える in-context learning による few-shot 学習でタスクを解くこととする．

[i] https://github.com/google/sentencepiece

[ii] https://github.com/aws/amazon-sagemaker-examples/tree/main/training/distributed_training/pytorch/model_parallel/gpt2

表 3 評価対象の大規模生成系言語モデルの仕様．Japanese-GPT-6B の学習データとトークナイザは日本語に特化しており，学習データの末尾の“-ja”は各データの日本語版であることを示す．

モデル	パラメータ数	学習データ	学習データ量	トークナイザ
GPT-J-6B	6B	Pile	402B tokens	Byte-level BPE
Japanese-GPT-6B (ours)	6B	Wikipedia-ja CC100-ja OSCAR-ja mC4-ja	170B tokens	SentencePiece

評価用のタスク及びデータセットとしては，日本語の生成系タスクのベンチマークで公開されているものはないため，生成系タスクに近いタスクとして日本語言語理解ベンチマーク JGLUE[iii]から下記 2 種の質問応答タスクを採用する．

- JCommonsenceQA
- JSQuAD

使用した JGLUE のバージョンは v1.1.0 である．テストデータは非公開であるため，開発データセットと同量のデータ（JCommonsenceQA は question 数が同量，JSQuAD は title 数が同量）を学習データセットからランダム分割してテストデータセットとして用いた．残りの学習データセットから few-shot 用の例文を抽出してプロンプトを作成した．プロンプトのチューニングには開発データセットを用いた．

表 4 に，各評価タスクにおけるプロンプト及び生成した回答文の例を示す．各タスクに対する回答は文生成によって行う．回答文生成には Transformers[iv]を用いた．常に確率最上位のトークンを生成する貪欲法にて文生成を行った．

JCommonsenceQA　常識的知識を問う質問文に対して，選択肢として与えられた 5 択から最も適切な回答を 1 つ選択するタスクである．JGLUE データセットの選択肢には 1 から 5 の選択肢番号が付与されているが本評価ではその選択肢番号は用いず，選択肢の各文字列を生成させて Exact Match で正否を判定して Accuracy を算出した．生成文が選択肢の文字列のいずれにも合致しない場合は不正解とした．Exact Match は JCommonsenceQA ベンチマークを解く条件としては厳しい制約であるが，大規模生成系言語モデルの実用に向けては直接的に回答を生成してほしいという期待からそのように設定している．プロンプトとして与える例文の数は 3 例とした．

JSQuAD　与えられたコンテキストに関する質問に対して，回答をコンテキストから抽出するタスクである．評価指標は Exact Match と F1 であり，F1 は文字単位で算出した．回答は文生成で行うため，生成文はコンテキストから抽出されるとは限らないが，評価指標算出においては回答がコンテキストから抽出されているか否かは不問とした．プロンプトとして与える例文の数は 2 例とした．テストデータ 4,777 件のうち 6 件で GPT-J-6B の最大トークン長 2,048 トークンを超えたため，その 6 件はいずれのモデルのテストデータからも除外した．

3.2 評価結果

表 5 に，本評価タスクにおける評価結果を示す．本評価の 2 つの質問応答タスクのいずれも，我々の Japanese-GPT-6B が GPT-J-6B と比較して性能が高い結果であった．これは日本語の質問応答タスクに対して，日本語に特化したモデルであることによる効果と考えられる．

表 6 に，JCommonsenceQA の評価において生成文が選択肢のいずれかに合致した割合（選択肢合致率）と，合致しなかった具体的な生成例を示す．いずれのモデルも選択肢合致率は約 99%であり，約 1%は選択肢と合致しなかった．しかし選択肢と合致しなかった場合でも意味としては合致しているものが多く見られた．これらの結果は JCommonsenceQA のような形式のタスクを in-context learning による few-shot 学習で正しく解くには，さらなる大規模化が望ましいことを示唆している可能性がある．これについてはさらなる検証が必要であり，今後の課題とする．

[iii] https://github.com/yahoojapan/JGLUE
[iv] https://github.com/huggingface/transformers

表 4 各評価タスクにおけるプロンプト及び生成した回答文の例．プロンプトを**太字**以外のテキストで，生成した回答文を**太字**のテキストで示す．プロンプトはタスク説明文，例文，テスト用問題文で構成される．

JCommonsenceQA	JSQuAD
[問題]に対する[答え]を[選択肢]の中から選んでください。 [問題]:会社で一番偉い人はだれ? [選択肢]:[社長，部長，人事部，課長，エントリーシート] [答え]:社長 [問題]:顔についていてものを食べるところは? [選択肢]:[鼻，目，言葉，口，電話] [答え]:口 [問題]:町より大きくて県より小さいものは何? [選択肢]:[村，役場，市，郡，町内] [答え]:市 [問題]:目標や手段や態度を一つに絞り、終始それで押し通そうとすること。また、そのさまを何という? [選択肢]:[剣道，なぎなた，牡丹槍，一本槍，管槍] [答え]:**一本槍**	[題名]と[問題]から[質問]に対する[答え]を抜き出しなさい [題名]:第一次世界大戦 [問題]:第一次世界大戦 [SEP] アメリカでは参戦から 6 週間の間、募兵者の人数が 7 万 3 千人と目標の 100 万人を大きく下回ったため、政府は徴兵を決定した。アメリカの徴兵は 1917 年に開始され、一部の農村部を除いて受け入れられた。 [質問]:参戦から 6 週間後のアメリカの募兵者の人数は? [答え]:7 万 3 千人 [題名]:大分市 [問題]:大分市 [SEP] 大分市(おおいたし)は、大分県の中部に位置する市。大分県の県庁所在地で、中核市に指定されている。 [質問]:大分市(おおいたし)は、何県の中部に位置する市? [答え]:大分県 [題名]:フィレンツェ [問題]:フィレンツェ [SEP] 市街中心部は「フィレンツェ歴史地区」としてユネスコの世界遺産に登録されている。1986 年には欧州文化首都に選ばれた。 [質問]:欧州文化首都に選ばれた年は? [答え]:**1986 年**

表 5 JGLUE データセットを用いた in-context learning による few-shot 性能評価結果

モデル	JCommonsenceQA	JSQuAD	
	Accuracy	EM	F1
GPT-J-6B	24.7	37.2	53.0
Japanese-GPT-6B (ours)	37.4	57.0	72.6

表 6 JCommonsenceQA データセットを用いた in-context learning による few-shot 性能評価における，選択肢合致率（テストデータ 1,119 件のうち，生成文が選択肢のいずれかに合致した件数の割合）と，合致しなかった生成例．タスク説明文や例文は省略し，テスト用問題文と生成した回答文（**太字**）のみを記載している．

モデル	選択肢合致率	合致しなかった生成例
GPT-J-6B	98.9%	[問題]:子どもたちは平日毎日どこに行っている？ [選択肢]:[努力する，公園，勉学に励む，学校に行く，会社] [答え]:**学校**
Japanese-GPT-6B (ours)	98.5%	[問題]:会津市がある県は? [選択肢]:[青森県，福島，秋田県，沖縄県，愛知県] [答え]:**福島県**

4 おわりに

本稿では，6B パラメータの日本語 GPT モデル (Japanese-GPT-6B)の事前学習を行い，公開データセット JGLUE の質問応答タスク 2 種を用いて in-context learning による few-shot 性能評価を行った結果について報告した．評価結果より，我々の Japanese-GPT-6B は GPT-J-6B と比較して質問応答性能が優位であり，日本語特化の効果を確認した．

今後の展望としては，JGLUE の他のタスクでの in-context learning による few-shot 性能評価と，同じく JGLUE データセットを用いてファインチューニングした場合の性能評価を行う予定である．さらには生成系言語モデルの本来の用途である言語生成タスク性能の評価と，生成文に対する後処理技術の検討を行い，大規模生成系言語モデルのビジネス応用を目指す．

謝辞

本研究のモデル構築にあたり，Amazon Web Services, Inc.の Amazon Machine Learning Solutions Lab による支援を頂きました．感謝いたします．

参考文献

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in Neural Information Processing Systems, 2017.

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In Advances in Neural Information Processing Systems, 2020.

[3] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. PaLM: Scaling Language Modeling with Pathways. arXiv preprint arXiv:2204.02311, 2022.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pretraining of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

[5] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683, 2019.

[6] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv preprint arXiv:2107.13586, 2021.

[7] Wang Ben and Komatsuzaki Aran. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax, 2021.

[8] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open Pre-trained Transformer Language Models. arXiv preprint arXiv:2205.01068, 2022.

[9] ABEJA で作った大規模 GPT モデルとその道のり . https://tech-blog.abeja.asia/entry/abeja-gpt-project-202207, 2022.

[10] 栗原健太郎, 河原大輔, 柴田知秀. JGLUE: 日本語言語理解ベンチマーク. 言語処理学会第 28 回年次大会, 2022.

[11] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. arXiv preprint arXiv:1808.06226, 2018.