日本語の大規模な基盤モデルに対する LoRA チューニング

王 昊[1]　中町 礼文[2]　佐藤 敏紀[2]
[1] 早稲田大学　[2] LINE 株式会社
conan1024hao@akane.waseda.jp
{akifumi.nakamachi, toshinori.sato}@linecorp.com

概要

本研究では, 日本語の大規模基盤モデルを用いて, テキスト分類・生成タスクにおける LoRA チューニング [1] を検証した. 具体的には, XLSum[2] (要約), JNLI[3] (含意関係認識), JCommonsenseQA[3] (常識推論) の三つのタスクにて, LoRA チューニングとファインチューニングを行い, チューニングしたモデルの精度や必要なパラメータ数や学習時間などの比較を行った. 比較実験により, ファインチューニングと比較して, LoRA チューニングは計算時間やメモリ使用量を大幅に低減でき, 推論の精度がファインチューニングと同等以上であることを確認した.

1　はじめに

自然言語処理の諸課題において, 大規模なテキストデータを用いた事前訓練による基盤モデルに基づく手法が広く存在している. 特に, GPT-3[4] をはじめとする超大規模な基盤モデルの追加訓練を伴わない Prompting に基づく手法が, 対話システム [5] などで高い精度を達成している. 一方で, 超大規模な基盤モデルによる Prompting に基づく手法では, Prompt の複雑な前処理や, 事後フィルタリングを用いた再生成を伴う出力制御などに膨大な計算コストが必要になる. また, 比較的に小規模な基盤モデルでも追加訓練によって, 特定タスクにおいて超大規模な基盤モデルと匹敵する精度が出せる場合がある. さらに, 超大規模な基盤モデルでも追加訓練によって, 特定タスクにて Prompting に基づく手法より高い精度を達成できる場合がある. 計算コストや精度の改善に向けて, 基盤モデルにおける軽量な追加訓練 [6, 7, 8] がある. 本研究では, GPT-3 と同様な形式の日本語の基盤モデルを用いて, LoRA チューニング [1] と通常のファインチューニングの比較検証を行い, メモリ使用量や訓練に必要な計算時間などの計算コストの評価や予測性能を評価した. 実験により, LoRA チューニングはファインチューニングと同程度の性能を大幅に低い計算コストで達成できることを確認した.

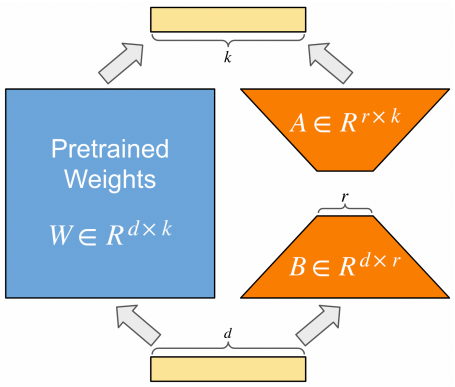

図 1: LoRA のアーキテクチャ

2　LoRA チューニング

パラレルコーパス $\mathbf{Z} = \{(x_i, y_i)\}_{i=1 \cdots N}$ を用いた基盤モデル $P_\Phi(y|x)$ の追加訓練のデファクトスタンダードな手法として, 全てのパラメータ Φ を, 式 1 のように最適化するファインチューニングがある. ファインチューニングでは, 事前訓練と同様に基盤モデルのパラメータ Φ 全てを更新するため, モデルのサイズと比例して計算コストが増大する. LoRA チューニングでは, 式 2 のように, 小規模なパラメータ $\Delta\Phi(\Theta)$ を導入し, 基盤モデルのパラメータ Φ_0 を固定し $\Phi_0 + \Delta\Phi(\Theta)$ とし, $\Delta\Phi(\Theta)$ のみを更新する. 特に, 図 1 で示すように, 基盤モデルのパラメータ Φ_0 のうち, パラメータ $W_0 \in \mathbb{R}^{d \times k}$ を持ち, 入力 z から 密ベクトル h を出力する線形層 $h = W_0 z$ に対して, 低ランク行列 $A \in \mathbb{R}^{r \times k}, B \in \mathbb{R}^{d \times r}, r \ll \min(d, k)$ を用いて $h = W_0 z + BAz$ とし, B, A のみを最適化する.

$$\max_{\Phi} \sum_{(x,y) \in \mathbf{Z}} \sum_{y_i}^{|y|} \log \left(P_\Phi(y_t | x, y < t) \right) \tag{1}$$

$$\max_{\Theta} \sum_{(x,y) \in \mathbf{Z}} \sum_{y_i}^{|y|} \log \left(P_{\Phi_0 + \Delta\Phi(\Theta))}(y_t | x, y < t) \right) \tag{2}$$

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 1: データセットの統計量

Task	Train	Validation	Test	Input Format
XLSum	7,113	889	889	{text}[SEP]{summary}
JNLI	20,073	1,217	1,217	{text1}[SEP]{text2}[SEP]{label}
JCommonsenseQA	8,939	560	559	{question}[SEP]{choice0}[SEP]{choice1}[SEP] {choice2}[SEP]{choice3}[SEP]{choice4}[SEP]{label}

超大規模な基盤モデルの W_0 のランク d は非常に大きく, 例えば 1750 億のパラメータを持つ 175B GPT-3 は $d = 12,288$ であるのに対し, r は 1 でも 2 でも実行できる. 英語における先行研究による 175B GPT-3 を用いた検証では, GPU メモリを三分の一, 学習可能なパラメータ数を一万分の一に削減しつつ, ファインチューニングと同等以上の精度を達成している. GPT-3 の他に, $\text{RoBERTa}_{\text{base}}$[9], $\text{DeBERTa}_{\text{xxl}}$[10], $\text{GPT-2}_{\text{medium}}$[11] による検証でも, ファインチューニングと同等以上の精度であった.

他手法との比較として, Adapter チューニング [7] では, Transformer[12] の各部分に Adapter 層が追加されることにより, 推論時の計算コストが LoRA チューニングよりも増加する. また, Prefix チューニング [8] では, 入力の一部を接頭辞に分ける必要があるため入力長が短くなるが, LoRA チューニングでは, 入力長を元のモデルのままに保つことができる.

3 実験

本実験では, LINE が作成している日本語の GPT のうち, 10 億個のパラメータを持つ 1B モデルと 67 億個のパラメータを持つ 6.7B モデルを使用し, XLSum[2], JNLI[3], JCommonsenseQA[3] のタスクで, LoRA チューニングとファインチューニングの比較検証を行った.

3.1 データセット

基盤モデルの言語理解の性能評価として, 含意関係認識タスクの JNLI や, 常識推論タスクの JCommonSenseQA の評価を行った. また, 自然言語生成の生成を測るため, 要約ベンチマークデータセットの XLSum のうち, 日本語のサンプルのみを抽出し評価を行なった.

3.1.1 JNLI

日本語の言語理解ベンチマークデータセットの JGLUE[3] に含まれる含意関係認識タスクである. 含意関係認識タスクでは, text1, text2 の 2 つの文の間に含意関係があるかを含意/中立/矛盾の 3 段階で判定する. 比較実験では, モデルに対して文のペアをモデルに与えて, 含意/中立/矛盾のいずれかの label のテキストを生成させる. 評価は Accuracy を用いる.

3.1.2 JCommonsenseQA

日本語の言語理解ベンチマークデータセットの JGLUE[3] に含まれる常識推論能力を評価するタスクである. 質問文 (question) と 5 つの選択肢のテキスト (choice0 – 4) のを与え, 正解のテキスト (label) を出力する. ラベルのインデックスを直接出力する BERT[13] などと異なり, ラベルテキストを生成する形式の本実験では, 選択肢にない回答を出力する場合が存在する. そこで本実験では, モデルに対して質問文と選択肢を与え, 生成されたテキストに対しての Exact Match (出力文字列と正解文字列が完全一致した出力を返した割合) を評価指標として用いる.

3.1.3 XLSum

XLSum は, BBC の記事の本文や要約文から作成された 44 言語を含むニュース要約タスクであり, 本文から要約文を生成する. 日本語の GPT の評価として, 本研究では XLSum の日本語部分のみを用いる. 自動評価の指標として ROUGE[14] を用いる.

3.1.4 データセットの分割

JNLI と JCommonsenseQA は Train データと Validation データのみが公開されており Test データが公開されていない. そこで, 本研究で ハイパーパラメータの探索などのための擬似的な Validation データとして元の Validation データの半分を用い, 残り半分を比較検証の際の Test データとして用いた. それぞれのタスクのサンプル数と, モデルへの入力フォーマットは, 表 1 に示す.

3.2 モデル

本研究では GPT-3 と同様な形式の基盤モデルを使用し比較実験を行う. モデルは LINE が独自に構築した JPLM コーパスで事前学習を行った. 1B モデルの Layer 数は 24, Hidden Dimension は 2,048, Attention

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 2: 各タスクの評価結果

Task	Method	Trainable Params (M)	GPU Memories (MB)	sec/iter (sec)	Metrics
JNLI	1B-FT	1,317	2,635	0.100	0.732
	1B-LoRA	6	24	0.123	0.731
	6.7B-FT	6,666	483,652	9.040	0.927
	6.7B-LoRA	16	54	0.120	0.935
	RoBERTa-large	336	-	-	0.924
JCommonsenseQA	1B-FT	1,317	2,635	0.089	0.641
	1B-LoRA	6	24	0.151	0.012
	6.7B-FT	6,666	483,652	9.031	0.933
	6.7B-LoRA	16	54	0.108	0.935
	RoBERTa-large	336	-	-	0.901
XLSum	1B-FT	1,317	2,635	0.472	0.417/0.154/0.290
	1B-LoRA	6	24	2.315	0.153/0.034/0.134
	6.7B-FT	6,666	483,652	9.060	0.397/0.139/0.274
	6.7B-LoRA	16	54	4.271	0.457/0.213/0.340

※ XLSum の評価指標は ROUGE-1/ROUGE-2/ROUGE-l.

Head は 16 で, 6.7B モデルの Layer 数は 32, Hidden Dimension は 4,096, Attention Head は 32 である. 両モデルの Max Position Embedding は 2,048 である.

3.3 実験設定

本実験では, 1B モデルと 6.7B モデルそれぞれに対し, LoRA チューニング (LoRA) とファインチューニング (FT) を行い, 予測性能や学習時間, メモリ使用量などの比較した. LoRA, FT それぞれの実験に用いたハイパーパラメータの探索範囲を表 3 に示す. LoRA Weight は, Self-Attention の内部の query, key, value の 3 つの線形層と, 出力層 (output) の重み W_q, W_k, W_v, W_o への LoRA の適用の組み合わせの設定を表す. LoRA r は, LoRA における低ランク行列の r である. また, 特に JNLI や JCommonsenseQA において, ラベルを直接生成できる BERT 系モデルと異なり, モデルが存在しない解答のテキストを生成する場合があるため, そのような場合は, 解答失敗として取り扱った. 本実験では NVIDIA A100 (80GB) を用いた. また, 6.7B FT については, モデルのサイズの都合上, 8 枚の NVIDIA A100 (80GB) を用いている.

表 3: 比較実験に用いたハイパーパラメータ

Hyper Parameter	LoRA	FT
Batch Size	8	{8, 16}
Learning Rate	{1e-5, 2e-5, 5e-5, 2e-4}	{1e-5, 2e-5, 5e-5, 2e-4}
Epoch	{2, 3, 4}	{2, 3, 4, 5, 10}
LoRA Weight	$\{W_q + W_v$, $W_q + W_k + W_v + W_o\}$	-
LoRA r	{4,8,16}	-

3.4 実験結果

表 2 に示した結果より, 本研究でのハイパーパラメータの探索範囲内では, 1B-LoRA は JNLI 以外のタスクにおいて, 適切な推論を行えなかった. 特に, JCommonsenseQA では, 1B-FT はほぼ全ての入力に対して解答候補のいずれかを生成したが, 1B-LoRA では解答候補が出力されることはなかった. 以上より 1B-LoRA は, タスクの難易度やハイパーパラメータの探索範囲に敏感であると考えられる.

また, 6.7B FT について, XLSum をのぞいて, 6.7B LoRA とほぼ同程度の精度であったが, GPU メモリの使用量や訓練の計算時間が膨大であった. 特に, 6.7B LoRA や 1B の実験では 1 枚の GPU 上でモデルを訓練できるが, 6.7B FT は 8 枚の GPU を用いてのみ訓練が可能だった.

6.7B FT や 1B FT などと比較して 6.7B-LoRA は全てのタスクにおいて, 大幅に小規模なメモリと計算時間で高い性能であった. 実験結果より, 6.7B-LoRA は, ハイパーパラメータの探索を広く行うことで, 小規模なモデルより高い性能を得られた.

また, BERT 系モデルとの比較の参考として, RoBERTa-large の評価結果を引用している.[1)] よって, 直接的な比較は行えないが, BERT 系のモデルと同等以上の予測性能であることを確認できた.

[1)] https://github.com/yahoojapan/JGLUE. JGLUE の評価では, Validation データ全てを用いて評価を行っているため, 本研究の他の実験と評価データが異なる.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 4: XLSum の生成例

Method	Text
Input	ワシントン州キング郡のアウトブレイクはひとつの介護施設に集中している 死者 6 人のうち 5 人は、シアトルのあるキング郡に住んでいた。 この地域ではこれまでに 18 人の感染が確認されており、今後の増加が懸念されている。 同州で最初に確認された死者 2 人について調べていた専門家によると、新型ウイルスは州内で数週間前から拡散し、これまでに最大 1500 人が感染している可能性があると指摘した。 ＜関連記事＞ ワシントン州のキャシー・ローファイ保健担当官は、感染はキング郡とスノホミッシュ郡に限定されているものの、ウイルスは「活発に」流行していると述べた。 また、他地域でも感染している可能性もあると指摘した。 キング郡で確認された患者 14 人のうち 8 人と死者 4 人が、特定の介護施設に関連している。 亡くなったほとんどは高齢者か、基礎疾患のある人だったという。 同郡当局は感染者の隔離施設として、ホテルを購入する方針。 シアトルでは休校が相次いでいる。 シアトルおよびキング郡公衆衛生庁のジェフ・ドゥチン博士は「事態を深刻に捉えている」とした上で、、現時点では広範囲での休校措置や大規模集会の中止などは考えていないと述べた。 その一方、感染者の数は今後も増えるだろうとの見方を示した。 これまでに 100 人近くが感染 アメリカではこの週末に感染者の数が急増し、懸念が広がっている。 これまでにアメリカ全土で 91 人の感染が確認されたが、このうち一部は感染多発地域に渡航していた。 一方で、アメリカ国内で感染したとみられる事例もある。 ワシントン州、カリフォルニア州、オレゴン州など西海岸の州当局は、感染地域への渡航歴や感染者との接触がない人からもウイルスが検出されたことに、懸念を示している。 その他のアメリカでの状況は以下の通り。 [SEP] 米ワシントン州で 2 日、新型コロナウイルスによる感染症（COVID-19）による新たな死者が 4 人報告された。 アメリカでの死者は 6 人となった。 ワシントン州は週末に非常事態宣言を発令している。
1B-FT	新型コロナウイルスのためアメリカ全土で死者が 4 人増え、最大 324 人になったことが 13 日、明らかになった。
1B-LoRA	米フロリダ州で初めて確認された。
6.7B-FT	新型コロナウイルスによる国内初の死者が 12 日、ワシントン州で確認された。同州保健当局は、今後も死者数は増え続けるとの見通しを示した。
6.7B-LoRA	米ワシントン州で新型コロナウイルスの感染者が急増している問題で、同州当局は 13 日、新たに 4 人が死亡したと発表した。

表 5: JNLI の生成例

Method	Text
Input	文 1: キリンが、木の中から首を出しています。 [SEP] 文 2: キリンが木々のあいだから顔を出しています。 [SEP] 含意
1B-FT	含意
1B-LoRA	中意
6.7B-FT	含意
6.7B-LoRA	含意

表 6: JCommonsenseQA の生成例

Method	Text
Input	質問: 田んぼが広がる風景を何という?[SEP]1. 畑 [SEP]2. 海 [SEP]3. 田園 [SEP]4. 地方 [SEP]5. 牧場 [SEP] 田園
1B-FT	田園
1B-LoRA	森
6.7B-FT	田園
6.7B-LoRA	田園

3.5 ハイパーパラメータ探索

Hu ら [1] は, LoRA の学習率として 2e-4 を用いているため, 本実験も 2e-4 を探索範囲に含めた. ハイパーパラメータの探索の結果, 大規模なモデルのチューニングは, 小規模なモデルに比べてハイパーパラメータに敏感であることが確認された. 特に, LoRA チューニングの性能は, 学習率などのハイパーパラメータだけでなく, LoRA を適用する箇所や r などのハイパーパラメータも強い影響があり, 多くのハイパーパラメータでは学習が適切に行えなかった. LoRA の Attention の重みタイプと低ランク分解行列の次元 r については, JNLI と JCommonsenseQA の両タスクにおいて, 一番コストが小さい組み合わせ (Weight Type = $W_q + W_v$, Adapter Dim = 4) が最適となっている一方, XLSum において一番コストが高い組み合わせ (Weight Type = $W_q + W_k + W_v + W_o$, Adapter Dim = 16) が最適であった. LoRA チューニングは, タスクに応じて, 最適な低ランク行列が大きく異なると考えられる. また, 6.7B FT でも, 1B FT と比較してより広範な探索が必要であり, 超大規模な基盤モデルのチューニングは, 効率的なハイパーパラメータの探索が重要であることが確認された.

4 まとめ

本研究では, 日本語の大規模な基盤モデルを用いて, 含意関係認識, 常識推論, テキスト要約の 3 つのタスクを用いて ファインチューニングと LoRA チューニングの比較を行った. 実験結果より, 日本語の大規模な基盤モデルのチューニングにおいて, LoRA チューニングがメモリ消費量や計算時間の観点で効率的であることを確認した.

謝辞

本研究は LINE 株式会社でのインターン成果である.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

参考文献

[1] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. In **International Conference on Learning Representations**.

[2] Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages. In **Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021**, pp. 4693–4703, 2021.

[3] Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. JGLUE: Japanese General Language Understanding Evaluation. In **Proceedings of the 13th Language Resources and Evaluation Conference**, pp. 2957–2966, 2022.

[4] 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**, pp. 1877–1901, 2020.

[5] 山崎天, 川本稔己, 大萩雅也, 水本智也, 小林滉河, 吉川克正, 佐藤敏紀. ペルソナ一貫性の考慮と知識ベースを統合した HyperCLOVA を用いたマルチモーダル雑談対話システム. 第 96 回 言語・音声理解と対話処理研究会 (第 13 回対話システムシンポジウム), pp. 113–118, 2022.

[6] Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. BitFit: Simple Parameter-efficient Fine-tuning for Transformer-based Masked Language-models. In **Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics**, pp. 1–9, 2022.

[7] Ruidan He, Linlin Liu, Hai Ye, Qingyu Tan, Bosheng Ding, Liying Cheng, Jiawei Low, Lidong Bing, and Luo Si. On the Effectiveness of Adapter-based Tuning for Pretrained Language Model Adaptation. In **Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing**, pp. 2208–2222, 2021.

[8] Xiang Lisa Li and Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In **Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing**, pp. 4582–4597, 2021.

[9] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. **arXiv preprint arXiv:1907.11692**, 2019.

[10] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Chen Weizhu. In **International Conference on Learning Representations**.

[11] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In **Advances in Neural Information Processing Systems**, 2017.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 4171–4186, 2019.

[14] Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In **Text Summarization Branches Out**, pp. 74–81, 2004.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).