事前学習済みLlama2モデルを活用した言語間転移日英モデルの作成

佐藤諒　麻場直喜　野崎雄太　中島大　近藤宏　川村晋太郎
株式会社リコー デジタル戦略部 デジタル技術開発センター　言語AI開発室
{ryo.sato4, naoki.asaba, yuta.nozaki1, dai.nakashima, hiroshi.xx.kondoh,
shintaro.kawamura}@jp.ricoh.com

概要

近年英語を中心に事前学習済み大規模言語モデルが多く公開されてきた。本研究ではそれらのモデルの中でも、ここ最近で高スコアを出したMeta社のLlama2 13B Chatモデルを再利用し、できるだけ性能の高い日本語モデルを作成することを試みる。特に言語間転移、カリキュラムラーニングの知見を投入し、なるべく事前学習の量を減らした継続学習を行い、日本語を優先した日英モデルの作成の検証結果を報告する。

1 はじめに

近年中程度の大規模言語モデル(10B前後)が多く作成されオープンなモデルとして公開されている。一方、日本語では学習データ、学習環境の制限もあり、性能の高い大規模モデルを作るのは以前困難な課題である。そのため、既にある性能の高い英語モデルを再利用し、日本語の継続学習をすることは十分に価値がある。

今回我々も可能な限りオリジナルモデルのアーキテクチャを変えずに既存英語モデルで性能の高いものを利用し、日本語モデルとしての性能の高いモデルへ変化させる試みを行う。本研究では英語データが日本語データよりも多く公開されている状況に合わせて、最初3:1の割合で英語：日本語データで学習し、その後1:3の割合で学習する転移学習を主とする効果を確認する。また学習の成功率を上げるために良質なデータから広範なデータへといった学習順序も含めて計画し、カリキュラムラーニングの要素ありで学習を進める。この二つの効果を確かめるために設定を理想形から外したモデルの作成、比較も行う。すなわち、両言語のデータを同比率で訓練した場合と割合を変えた場合の比較実験、データ順序を考慮する場合としない場合との比較実験、元々の英語モデルの重みを再活用する場合としない場合の比較実験の結果を公開する。

2 セットアップ

2.1 スタートモデルの準備

継続学習をスタートするにあたって、まずは学習開始モデルである既存のLlama2(Meta社 Llama 2 13B Chat [1])モデルを理解することが必要である。素のLlama2モデルは英語を中心に学習されており、英語90%に対して日本語はわずか0.1%である[2]。Tokenizerについてもそれを受けており、英語語彙がほとんどを占めている。実際に計測してみると、800語程度(CJK, HIRAGANA, KATAKANAの計)であり、全体の32000語彙と比べると少量である。このままで継続学習すると、日本語の文章分解能自体が下がり、学習効率が低下する可能性が高い。そこでtokenizerの改良にも取り組む。

最初、日英両言語に対応できるtokenizerを作るために、日英でwikipediaコーパスを用いて学習する。手法はSentencePiece[3]で、BPE構築行い作成する。この学習により、(LATINを含む)英語が13841語彙、日本語(CJK, HIRAGANA, KATAKANA)が17033語彙の結果になった。ここで語彙数合計数が32000に満たないのはその他の文字を含むためである。ここで文章分解能(Length per Token)を計算すると、32000語彙全て日本語語彙の場合と比べて、84.88%の長さであり、32000語彙全て英語語彙の場合と比べると88.92%の長さであり、同じ文章に対してのtoken分解数は微増の結果となった。これはtokenizerの語彙数と文章分解能の関係がlogで近似できるからである[4]。この日英対応のtokenizerを本体の重みと結合する。Tokenizerは学習をし直したことにより、元のtokenizerと一致する語彙と、一致しない語彙を持つ。一致する語彙に関してはembeddingの重みと合

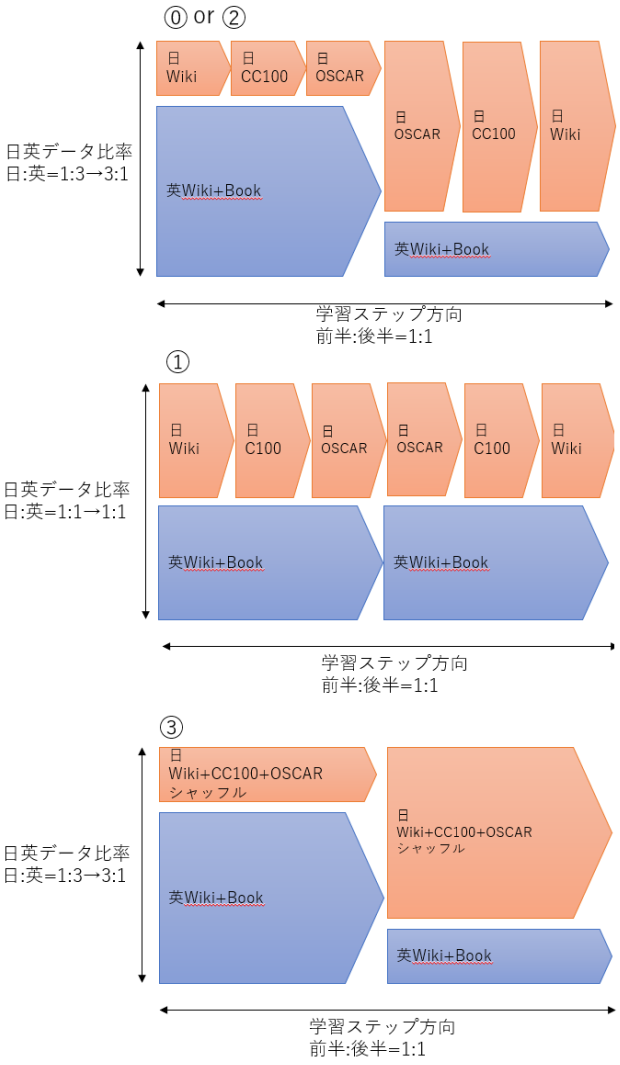
This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

うように語彙 ID の順番を合わせる(並び替える)。また、一致しない語彙に関しては ID を合わせないで適当な embedding の重みに結合する。こうすることでなるべく大量の英語データで学習された Llama2 モデルの重みを再活用することができる。ここでは詳しく触れられなかったが、他の tokenizer と embedding の結合方法は[5]で触れる。

2.2 学習データの準備

英語から日本語に強いモデルに継続学習で改造するために、コーパスは日英どちらも用意する。しかしながら、オープンのデータでは日本語データよりも英語データのほうが多いという事情があるため、日本語データに合わせて必要な英語データの数を決める。バイリンガルモデルを学習する際には主言

図 1 各実験モデルのデータ比率



語:サブ言語のデータ比率

を 2:1 で学習すると良いという先行研究がある[4]。今回は日本語データが不足している背景と、より緩急を付けた割合で学習して違いを見るために敢えて 3:1 の割合で学習を行う。日本語データで用いるのは公開データの Wiki(0.7B token), C100(12.5B token), OSCAR(21.0B token)である。ここで B は 10 億の略である。このとき、データの品質は Wiki, C100, OSCAR の順で低下していきよりデータ量の多い、広範なデータとなる。なるべく品質の高いデータから学習したほうがよいカリキュラムラーニング[6]の文脈ではこの順番で行う。一方で英語データのほうでは比較的品質の高い Wiki+Book のデータを一貫して用意し、日本語とトータルで同量分確保する。

2.3 比較学習モデル設定

これまで述べてきたそれぞれの観点で比較が可能なように 4 つのモデルを作成して検討する(図 1)。

実験⓪番のモデル:

・事前学習済みの Llama 2 13B Chat モデルの重みを利用する。Tokenizer は 2.1 節で説明した通りの設定とし、他は元の Llama2 のアーキテクチャをそのまま利用する。

・英:日=3:1 のデータ割合で最初学習し、その後、英:日=1:3 のデータ割合で学習する。こうすることで最初に英語の出力本位だったのが、徐々に日本語本位になることが予想され、また英日データが緩急を付けた割合で学習が進むことから英語で獲得した知識の忘却を防ぎつつ、英語から日本語に強いモデルへの転移を期待できる。

・日本語データは前半に wiki, CC100, OSCAR の順で学習するというカリキュラムラーニングの文脈でいう品質の高い順に相当する。また後半は逆に OSCAR, CC100, wiki の順で学習し、最後のデータに出力が引きずられやすいことを考慮して、品質の高い wiki データで締めくくる。以上の三点を抑える。

実験①番のモデル:

比較用に⓪の設定の中で英:日=3:1→1:3 の割合で学習するところを 1:1 で学習するように変える。

(言語間の特性を無視する場合)

実験②番のモデル:

⓪の設定の中で Llama 2 13B Chat モデルの重みを利用せずにスクラッチから学習をスタートする。

(継続学習をしない場合)

実験③番のモデル:

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

⓪の設定の中で wiki, CC100, OSCAR の順番にせず、シャッフルして学習する。
(カリキュラムラーニングを無視する。)

これらの 4 つのモデルを比較し、特に学習結果に効く方法を確認し、また一部で途中のデータ区切りごとにチェックポイントの保存を行ったため、チェックポイントごとの評価を行うことで、言語間転移の様子を観測できると期待できる。

2.4 学習環境と詳細設定

本実験の学習環境は Amazon EC2 Trn1 インスタンス (trn1.32xlarge) 64 ノードを使用した。学習フレームワークは AWS Neuron Reference for NeMo Megatron であり、Llama 2-13b の学習コードサンプル をベースに改造し、本実験の継続学習、カリキュラムラーニングが可能な状態にして使用した。AWS Neuron SDK のバージョンは 2.14 である。ハイパーパラメータとして、バッチサイズは Llama 2 [2]と同様に 4M トークンであり、学習率は Loss が比較的安定しているラインの 8.0×10-5 にした．カリキュラム学習における学習データのつなぎ時には学習率ウォームアップを実施して学習の安定化を図った.

3 実験結果と考察

3.1 各実験のモデル精度評価

2 節で行った各学習モデルに対して下流タスクの性能評価を行った。性能評価に関しては公開評価データセットとベンチマークツールを用いた自動評価を行った。具体的には日本語版は stabilityAI 社の lm-evaluation-harness [7]、英語版は EleutherAI 社の lm-evaluation-harness [8]を用いた。

lm-evaluation-harness の設定は few_shot 数を一貫して 3 に固定して行った。英語の lm-evaluation-harness は MNLI (Acc(%)), QNLI(Acc スコア(%)),のタスクについて行う。これらの推論タスクの結果を見て、学習が進んだとしても英語での推論能力が残存するか確認する。一方、日本語の lm-evaluation-harness は JCommonsenseQA, JNLI, MARC-ja, JSQuAD, Jacket_V2, MGSM, JCoLA, JAQuAD という複数のタスクについて行う。それぞれのタスクの設定は表 1 に記載した。

実験⓪~③のモデル対して評価スコアの平均を計算した結果が表 2 である。

表 1 lm-evaluation-harness (日) でのタスク設定

Task	Version	Prompt	評価スコア
JcommonsenseQA	1,1	0.1	Acc(%)
JNLI	1.3	0.1	Acc
Marc_ja	1.1	0.3	Acc
JSQuAD	1.1	0.1	F1
Jacket_v2	0.2	0.1	F1
MGSM	1.0	0.3	Acc
JCoLA	0.2	0.3	Balanced_acc
JAQuAD	0.1	0.1	F1

表 2 lm-evaluation-harness 英日ベンチマーク結果

モデル	日平均スコア	英推論平均スコア
Llama2	63.6	53.5
実験⓪	70.0	54.8
実験①	69.1	52.8
実験②	29.1	41.8
実験③	68.2	52.3

結果的に最も精度が良くなるようセットアップした実験⓪のモデルが良いスコアを出した。またモデルの重みにおいてスクラッチから学習が開始されている実験②のモデルに関しては英語で十分に事前学習された Llama2 の性能を活用できない手法であるため、他の実験モデルと比べ、大きく精度を落とした。またこれは見方を変えると、圧倒的に英語事前学習の多い元の Llama2 モデルの重みを再利用することで、少量の数十 B 程度の継続学習で高英語性能から高日本語性能への言語間転移をしたということができる。

わずかな差であるが、実験①との比較により、前半に英語の割合を多くして学習し、後半で日本語を多く学習したほうが日英を均等に学習するよりも効率的に英語の知識転移を行えることが分かった。また、実験③との比較により、日本語だけでも高品質なデータから学習することは精度の改善に寄与することが分かった。これは日本語に関して、元の Llama2 モデルでは性能が高くないことから、新たに継続学習で入る日本語に対応するために品質に拘ったデータから加える必要があることを示唆している。

さらに英語のスコアを見ると、こちらも日本語同様に実験⓪の結果が最も良いスコアを出した。それだけではなく、実験①、実験③のスコア順も変化していない。2.1 節で説明した Tokenizer の重み利用の方法より、英語主体の元の Llama2 の embedding か

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

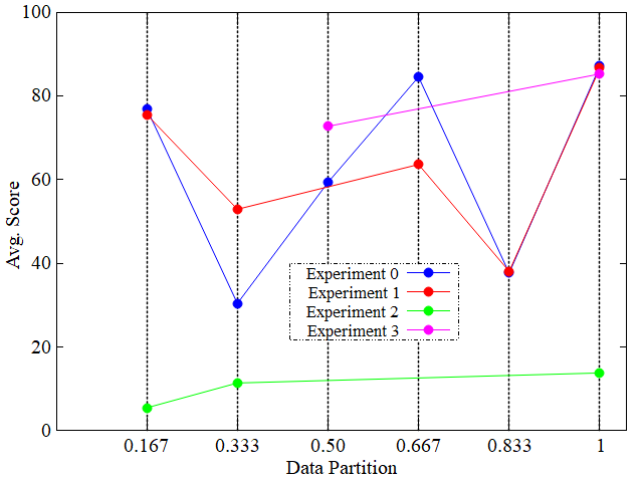
ら始まる重みを活用出来てはいるが、一部の重みはそうでないため、今回の継続学習で英語部も多少再学習が始まっていることになる。このときに、英語データの学習総量は実験⓪、実験③で変わらないが、日本語データの品質順に影響を受けて英語推論能力が実験⓪と実験③で変化する言語間波及が起きていると考えられる。

3.2 チェックポイントにおける評価

3.1 節では各実験の大きな単位で結果を評価したが、日本語の性能がどう変化しているか見るため、途中チェックポイントでも同様に精度を評価する。lm-evaluation-harness の中でも特に知識獲得+読解力を調べるため JCommonsenseQA, JSQuAD, Jacket_V2, JAQuAD のタスクを測定した。タスク設定は表 1 と同様である。平均スコアで比較した結果が図 2 である。横軸は学習データの切り替え時の刻みであり、学習 step 数ではない。具体的には実験⓪のデータ区分をもとに決めており、前半日本語 wiki, CC100, OSCAR データの学習終わりをそれぞれ 1/6, 1/3, 1/2 とし、後半日本語 OSCAR, CC100, wiki データの学習終わりをそれぞれ 2/3、5/6、1 としている。

図 2 を見ると、最終的に精度が高いのは 3.1 節でも説明したように実験⓪のモデルであるが、学習前半途中のチェックポイント(0.333)では、日英 1:1 の割合で学習した実験①のモデルが高い精度を出している。これは実験⓪で学習の前半に英語の学習が優先されるような比率で行なっているため、日本語の性能が前半は上がり切っていない学習途中の様相を呈しているからである。しかしながら、基礎的な日本語能力を獲得し、後半に日本語の学習データが上昇した領域では両者の精度が逆転している。これは前半では英語が易しいデータに、日本語が難しいデータに相当するとも考えることができる。この傾向は日本語の学習データを難易度によらずシャッフルしている実験③の結果が実験①同様のなだらかな上がり幅をしているのを見れば関連を推測することができる。また、実験⓪、実験①共に品質の高い wiki のデータ学習時が高いスコアを出しており、途中のデータでデータ量確保のために品質を下げて学習データを増やした場合、最終的に品質の高いデータで学習し終える必要性も示している。

図 2 各チェックポイントの日本語タスク性能



4 おわりに

本稿では，オープンな大規模言語モデルである Llama 2 13B Chat に対して日英コーパスデータを用いて語彙置換継続事前学習を行い，言語間転移について調査した結果を報告した。Tokenizer を改良し、なるべく元のモデルの重みを再利用しながらカリキュラムラーニングを行うことで、ある程度の英語能を残したまま、性能の高い日本語モデルを作成することに成功した。また本稿の結果は Ricoh モデルの学習途中のものであり、ここからさらに継続学習した結果は[9]の発表を参照されたい。今後の展望としては、データセットをコーパス種や言語での大きな区分でなく、もっと小さいセクションでカリキュラムラーニングの手法を活用することである。またここまで実験した日英 2 言語に中国語を混ぜた三言語に拡張し、さらに大規模な 70B パラメーターのモデル作成にも取り組む予定である。

謝辞

本研究のモデル構築、開発実験にあたり，アマゾン ウェブ サービス ジャパン合同会社の AWS LLM 開発支援プログラムにより多大な助力を受けました。感謝申し上げます。

参考文献

[1] “meta-llama/Llama-2-13b-chat · Hugging Face.” Accessed: Jan. 11, 2024. [Online]. Available: https://huggingface.co/meta-llama/Llama-2-13b-chat

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

[2] Hugo Touvron, *et al.*, Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv preprint arXiv: 2307.09288, 2023.
[3] 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.
[4] 中島大, 野崎雄太, 麻場直喜, 佐藤諒, 川村晋太郎, BPE を用いたトークナイザーの性能に対する、 言語・語彙数・データセットの影響. 言語処理学会第 30 回年次大会, 2024.
[5] 野崎雄太, 中島大, 佐藤諒, 伊藤真也, 近藤宏, 麻場直喜, 川村晋太郎. 大規模言語モデルに対する語彙置換追加事前学習の有効性の検証. 言語処理学会第 30 回年次大会, 2024.
[6] Yoshua Bengio, *et al.,* Curriculum learning. ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning, June, 2009. Pages 41–48.
[7] JP Language Model Evaluation Harness https://github.com/Stability-AI/lm-evaluation-harness
[8] Language Model Evaluation Harness https://github.com/EleutherAI/lm-evaluation-harness
[9] 麻場直喜, 野崎雄太, 中島大, 佐藤諒, 池田純一 伊藤真也, 近藤宏, 小川武士, 坂井昭一朗, 川村晋太郎, 語彙置換継続事前学習による日英バイリンガルモデルの構築と評価. 言語処理学会第 30 回年次大会, 2024.

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