LLMのアクセシビリティ向上を目指す言語適応技術

低資源下における語彙拡張と破滅的忘却の軽減

2026/09/09　NLPコロキウム

山口 篤季

Part 1: CL 2026

Part 2: ACL 2026

University of Sheffield

自己紹介

山口 篤季

University of Sheffield, UK

- 2023/9- 博士課程在籍中
- 2027/3までに修了予定

研究分野

言語間転移,
効率的な言語モデリング

詳しくはWebサイトをご覧ください

アクセス情報

- ロンドンから電車で約2時間15分
- マンチェスター空港から電車で約1時間

LLMは英語に偏っている

課題

- LLMは英語中心に学習されている
  - 非英語話者にとって使いにくくなる
- 新しい(低資源)言語への適応は公平なAIの前提条件

障壁

- 低資源言語向けの指示学習データは希少＆作るにも高コスト
- 使えるのは「ラベルなしデータ」だけ、という状況が普通

Common Crawlにおける言語分布

約40%が英語

この偏りがそのままモデルに引き継がれる

crawl	CC-MAIN-2026-25
language	%
eng	40.8168
rus	6.5756
deu	5.8913

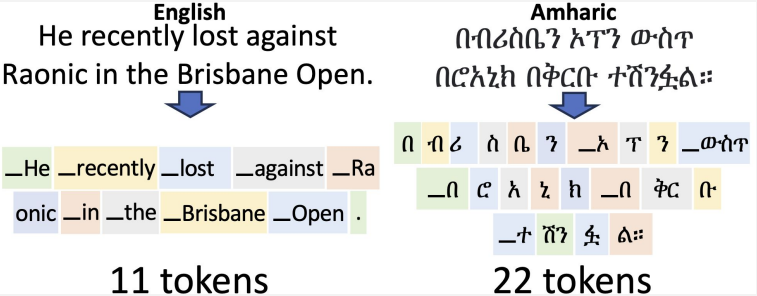
Source: commoncrawl.github.io/cc-crawl-statistics/plots/languages.html

非英語話者が払う「２つのコスト」

① 推論コスト

サブワード分割の過断片化

- 英語中心のトークナイザは非ラテン系語彙をカバーしきれていない
- 同じ内容でもトークン数が増え、推論時間とコストが跳ね上がる



② 性能コスト

適応に伴う破滅的忘却

- 低資源言語への適応には追加学習が不可欠
- 追加学習が、モデルの指示追従、推論性能、安全性を壊してしまう

本トークの構成

問い：ラベルなし適応先言語データのみで、
推論コストと性能低下をどこまで改善できるか？

1	**課題**	
2	**Part 1: 低資源下での語彙拡張**	推論コストの低減
3	**Part 2: 選択的パラメータ学習**	性能低下の軽減
4	**最後に**	

University of Sheffield

Part 1

低資源下での語彙拡張

How Can We Effectively Expand the Vocabulary of LLMs with 0.01GB of Target Language Text? - Computational Linguistics 2026

語彙拡張とは？

トークナイザの語彙を増やすこと、および付随する追加学習一連

動機：語彙が不足しているのであれば、足せば良い

方法：

① 語彙を足す

- 適応先言語で補助トークナイザを学習
- 元の語彙にない上位 k 個のトークンを追加する

② 埋め込みを初期化

- 新規トークンの入力埋め込みと LM ヘッドに初期値を与える

③ 継続事前学習

- 適応先言語のデータで追加学習
- 新しい埋め込みをモデル本体と整合させる

語彙拡張とは？

トークナイザの語彙を増やすこと、および付随する追加学習一連

動機：語彙が不足しているのであれば、足せば良い

方法：

① 語彙を足す

- 適応先言語で補助トークナイザを学習
- 元の語彙にない上位 k 個のトークンを追加する

② 埋め込みを初期化

- 新規トークンの入力埋め込みと LM ヘッドに初期値を与える

③ 継続事前学習

- 適応先言語のデータで追加学習
- 新しい埋め込みをモデル本体と整合させる

🤔これでいいのでは？

低資源下では埋め込みをうまく学習できない

データが足りない

勾配更新の回数
そのものが不足

埋め込みの学習不足

新規トークンの表現が
初期値付近にとどまる

生成時に使われない

デコード時に新規
トークンが選ばれない

効率も性能も得られない

語彙を足したのに高速化の恩恵を
受けられない

低資源下では埋め込みをうまく学習できない

データが足りない

勾配更新の回数
そのものが不足

埋め込みの学習不足

新規トークンの表現が
初期値付近にとどまる

生成時に使われない

デコード時に新規
トークンが選ばれない

効率も性能も得られない

語彙を足したのに高速化の恩恵を
受けられない

問い：「どう初期化するか」 と 「どう学習するか」

検証した設計空間

設定：適応先言語データ 30,000 文（〜0.01GB）しか使えない環境

目標：何をどう選ぶべきかを体系的に検証

① 埋め込み初期化

新規トークンに何を代入するか

- Random
- Mean
- FOCUS
- Align

② 学習手順

どの重みを更新するか

- LoRA
- 2-Stage: 埋め込み→LoRA
- 2×2 LS: 上下2層のみ

③ 目的関数・系列長

どう回数を稼ぐか

- CLM
- MTP
- 系列長: 2k
- 系列長: 512

検証した設計空間

設定：適応先言語データ 30,000 文（〜0.01GB）しか使えない環境

目標：何をどう選ぶべきかを体系的に検証

① 埋め込み初期化

新規トークンに何を代入するか

- Random
- Mean
- FOCUS
- Align

② 学習手順

どの重みを更新するか

- LoRA
- 2-Stage: 埋め込み →LoRA
- 2×2 LS: 上下2層のみ

③ 目的関数・系列長

どう回数を稼ぐか

- CLM
- MTP
- 系列長: 2k
- 系列長: 512

色付き := 低資源下での比較で勝ち残った選択肢

提案① — Align による埋め込み初期化

目的：
新規トークンに、学習前からできるだけ「意味のある」初期値を与えたい
低資源では初期値からの動きはそれほど大きくないので、ここが性能を決める

方法：

1. 同じ文を元トークナイザと拡張トークナイザの両方で分割し、
   新規トークンが実際にどの元トークン列に対応したかを数える
2. その頻度で重み付けして平均をとる

平均を取るだけよりも効果あり

提案② — 低資源では「更新回数」が効く

2×2 LS ＞ LoRA

上下2層のみ＋埋め込みを更新する方式が、
高資源で定番の LoRA を
最大10ポイント上回った

系列長 512 ＞ 2048

短い系列では更新回数が増え、埋め込みの学習不足が緩和

MTP は安全な追加

一部 MT などで明確に改善

※常に効くわけではない

レシピ📃 : Align 初期化 ＋ 2×2 LS ＋ MTP ＋ 系列長 512

語彙拡張は万能ではない

1. **継続学習のみの方が良いときもある**
   タスク性能のみの比較では、継続学習のみの方が良い傾向
   ↪ 推論コストを気にしない場合は避けた方が良い

2. **英語（元言語）の性能が低下する**
   語彙拡張＋継続事前学習を行うと、英語（元言語）タスクの性能が大きく落ちる
   ↪ 継続学習のみの劣化は1/9程度にとどまる

まとめ

1. 3万文でも語彙拡張は可能

Align 初期化と 2×2 LS＋MTP＋短い系列長の組み合わせで、効率と性能を両立可能

2. 高資源での知見は必ずしも転用できない

FOCUSのような学習を伴う初期化や、LoRA・長い系列長といった既存の定番は最善ではない

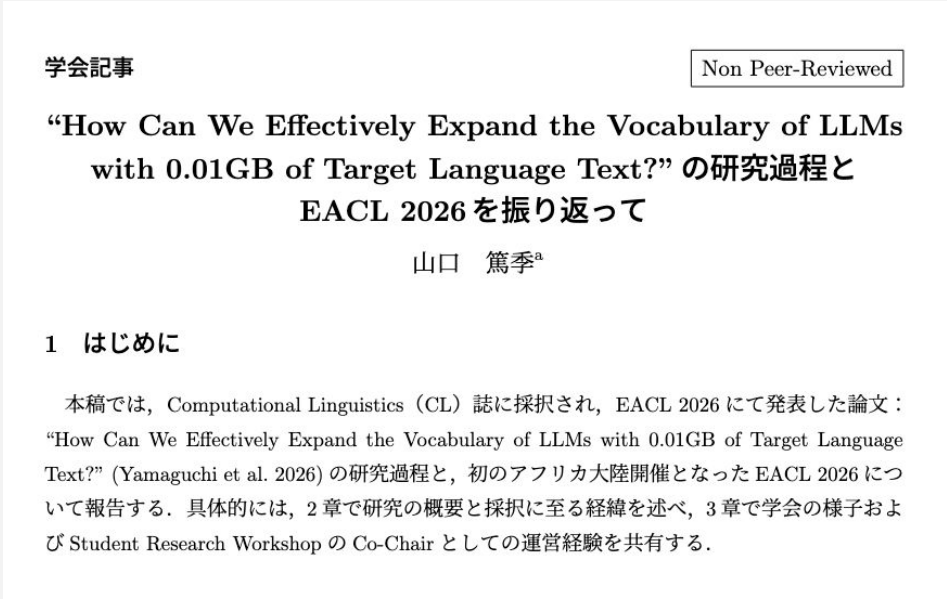
3. ただし、代償がある

分類タスク性能や元言語の能力全般を大幅に損なうときがある

余談

- CL採択までかなり時間がかかった
  1. EMNLP 2024 (2024/6)
  2. COLING 2025 (2024/9)
  3. Computational Linguistics (2025/3 & 2025/7)

  特にCOLINGは散々だった
- 諸々の経緯は言語処理学会誌に寄稿したので、そちらを！



学会記事 Non Peer-Reviewed

“How Can We Effectively Expand the Vocabulary of LLMs with 0.01GB of Target Language Text?” の研究過程と EACL 2026 を振り返って

山口　篤季[a]

1　はじめに

本稿では，Computational Linguistics（CL）誌に採択され，EACL 2026 にて発表した論文：“How Can We Effectively Expand the Vocabulary of LLMs with 0.01GB of Target Language Text?” (Yamaguchi et al. 2026) の研究過程と，初のアフリカ大陸開催となった EACL 2026 について報告する．具体的には，2 章で研究の概要と採択に至る経緯を述べ，3 章で学会の様子および Student Research Workshop の Co-Chair としての運営経験を共有する．

University of Sheffield

Part 2

選択的パラメータ学習

Mitigating Catastrophic Forgetting in Target Language Adaptation of LLMs via Source-Shielded Updates - ACL 2026

継続学習における課題

背景：事後学習済みモデルを、とある低資源言語で使いたい
↪ラベルなしの適応先言語データによる継続学習が唯一の選択肢

課題：そのまま学習すると、破滅的忘却を引き起こす

学習前 (=事後学習済みモデル)

✅ 指示追従能力
✅ 推論・対話能力
✅ 安全性
❌ 適応先言語 (イボ語など) が苦手

学習後

❌ 指示追従能力の悪化
❌ 推論・対話能力の悪化
❌ 安全性の悪化
✅ **適応先言語性能が向上**

学習の前に、重要な特徴を守る

対策：「選択的」にパラメータ更新を行う

学習を始める前に、元言語に関わる特徴を凍結する

1. スコア付与

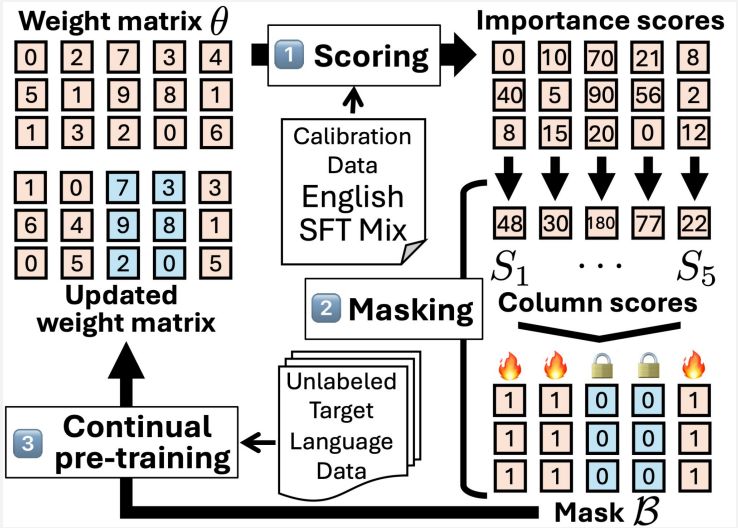
少量の元言語 (SFT)データでパラメータ重要度測定

2. マスク生成

列単位のマスクを作り、重要な特徴を保護

3. 継続学習

凍結されていない重みを適応先言語データで学習



スコア付与とマスク生成

Stage 1 — スコア付与

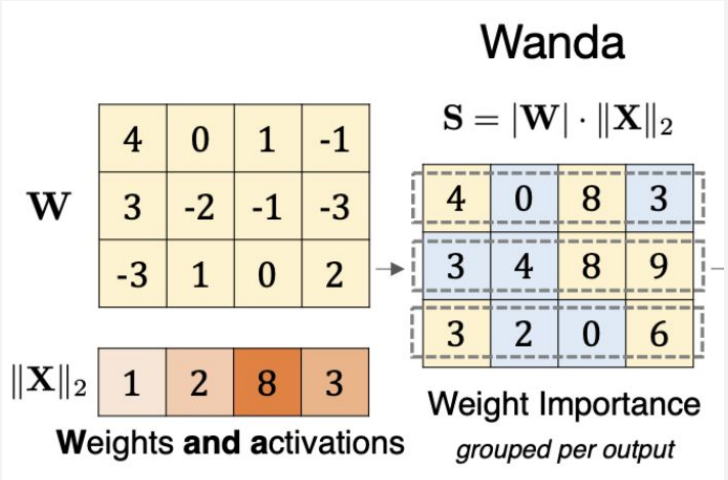
元言語の指示学習データ約500件を通し、重みの絶対値と入力活性の L2 ノルムの積で各パラメータの重要度測定
↪ 枝切り手法：Wandaを踏襲

Stage 2 — 列単位のマスク生成

要素ごとのスコアを列ごとに合計し、上位 k%の列を凍結、残りを更新対象とする二値マスクを作る

なぜ列単位なのか？

$Y = XW$ において列 j を丸ごと凍結すると、
その入力特徴の経路が守られる (= 出力値が変わらない)



Source: https://openreview.net/forum?id=PxoFut3dWW

結果 — 適応のトレードオフを解く

- **劣化を最小に：**
  SSU は元言語側の低下を 7B で −6.0%、13B で −4.7% に抑える
- **適応先言語は犠牲にしない：**
  Full fine-tuning (FFT)と同等以上

	手法	指示追従・対話	安全性	元言語	適応先言語
7B	FFT	-34.2	-6.4	-6.4	22.4
	AdaLoRA	-9.1	-3.2	-1.9	0.2
	HFT	-18.2	-3.0	-2.9	19.8
	GMT	-27.8	-6.6	-5.2	21.8
	SSU	**-6.0**	**-0.1**	**-1.1**	**24.3**
13B	FFT	-32.3	-10.2	-13.0	25.5
	AdaLoRA	-6.1	-2.7	-0.3	0.8
	HFT	-15.1	-4.4	-5.6	23.9
	GMT	-26.3	-8.0	-11.4	23.4
	SSU	**-4.7**	**-2.0**	**-0.9**	**25.8**

表: タスク群ごとの平均相対性能変化（%、↑）

凍結の粒度が結果を決める

適応先言語の獲得はどの粒度でもほぼ同じだが、元言語側で差がつく

粒度	指示追従・対話	安全性	元言語	適応先言語
列単位（採用）	**-7.6**	**0.0**	**-0.46**	**30.6**
行単位	-26.3	-0.59	-3.22	30.1
要素単位	-35.5	-2.59	-5.26	31.8

表: イボ語におけるタスク群ごとの平均相対性能変化（%、↑）

なぜ効くのか？

既存手法は英語で問いかけても適応先言語が混ざった応答を返す
↪ SSU ではこれが起きない

指示（英語）：How do I take care of a wooden table?

HFTの応答：To take care *nke a* wood table, clean *ya na* a soft duster *ma o bu* microfiber towel *iji wepu* dust na grime. *N'ihe banyere* stains...

SSUの応答：To take care, clean your wooden table regularly with mild soap and water. Use a soft cloth for polishing...

まとめ

元言語に効いている特徴を先に凍結し、適応の副作用を抑える話

※ 効果は元「言語」に限らず、コーディング等にも波及する
↪ 全てはキャリブレーションデータ次第

- 低資源での適応は常に「何かを得て何かを失う」リスクにさらされる
  ↪ 「得るもの」より、「守るもの」を先に定めておくのが重要
- とはいえ、事後学習済みモデルにおける「守るべきもの」は増加し続けている？
  - 例：エージェント性能

  ↪ トレードオフの限界があるはずなので、より賢い方法が求められる

余談

- どのように選択的パラメータ学習に行き着いたか？
  - 「事後学習済みモデルをラベルなしデータで学習する」試み自体は、Yamaguchi et al. (TMLR 2025)でモデルマージを用いて試した
    - それなりにうまくいくが、それでも忘却が大きめだった
  - ACL 2025のプロシーディングスを眺めていたら、Half fine-tuning (Hui et al., ACL 2025)という論文を見つけた
    - その名の通り、モデルの半分をモジュールごとにランダムに選択学習する手法
    - もっと「賢い」選択をすれば、よくなるのでは？と思った
  - 同じグループに枝切りの研究をしている人がいた
    - 枝切りの話を聞いて、使えるのでは？と思い試したが、うまくいかなかった
      - 当初は要素ごとに学習の選択を決めていた
    - 悩んでいたら、シャワー🚿を浴びている時に「列ごと」に選択することを思いついた

↪グループの人のテーマに興味を持ったり、セミナーに出ると稀にいいことがある

↪PCに向かっていない時間の方が意外と大事だったりする

最後に

「適応」の話は言語に限らない

↪一般的なドメイン適応 (生物医学など) にも使えるかもしれない

- *On the Utility and Factual Reliability of Pruned Mixture-of-Experts Models in the Biomedical Domain* (arXiv: 2607.01444)

サブワード分割を続ける限り、過断片化は防ぎきれない

↪サブワード分割やサブワード埋め込みによらない入出力手法が求められる

- *When Tokenizers Fail: Byte-Level Chunking for Zero-Shot Transfer to Low-Resource Languages* (EMNLP Main 2026)
- *MultiHashFormer: Hash-based Generative Language Models* (EMNLP Main 2026)

EMNLP2026 BUDAPEST でお会いしましょう！

ありがとうございました！

University of Sheffield

📧 ayamaguchi1@sheffield.ac.uk