E-031

日本語事前学習言語モデルにおける 語彙の直接的操作を用いたドメイン適応の試みが 下流タスクの精度に与える影響の評価

The Effect Evaluation of Direct manipulations to Vocabulary in Japanese Pretrained Language Models

浜 直史[1)] 安井 雅彦[1)] 森 靖英[1)] 和久井 一則[2)]
Naofumi Hama Masahiko Yasui Yasuhide Mori Katsunori Wakui

1 はじめに

近年の機械学習を利用した自然言語処理技術の発展の要因の 1 つが，学習過程を事前学習言語モデル (PLM: pretrained language model) の作成とタスク依存の Fine-Tuning へ分離可能になったこと [1, 2] である。

一般に，大量のコーパスを用意し大規模なモデルを作成することで，多くのタスクで高精度を達成する高性能な事前学習言語モデルが作成できると考えられている。また，実際に自然言語処理モデルを運用するドメインと同一ドメインからコーパスを収集するようにすると，そのドメイン特有の語彙や統語規則などに従って学習を行うことができるため，より高性能なモデルを作成できる。しかし現実には，このような特定ドメインについて事前学習言語モデルを作成できるほどの量のコーパスを収集することは困難である。そこで，近年公開数が増えてきている，日常的に用いられる文章からなるコーパスを用いて作成されたドメインに特化していない事前学習言語モデルを用い，これに Fine-Tuning を施して精度の高いモデルを作成する技術が必要とされる。

この Fine-Tuning の際に行われるドメイン適応を促進させる目的で，最も単純に考えられるのが，事前学習言語モデルの語彙から当該ドメインに現れないであろう語を削除し，当該ドメインに頻出する固有表現などで上書きする手法である。事前学習言語モデルの語彙とは，埋込ベクトルに変換されるトークンの一覧のことである。

日本語を処理するモデルで採用される語彙作成手法の 1 つである SentencePiece[3] では，連続する複数文字の組み合わせ (サブワード) のうちコーパス内で出現する頻度の高いものから順に (byte-pair encoding[4])，事前学習言語モデルごとに予め設定された数までトークンとして採用される。このように語彙をデータドリブンに作成することで，単語単位などで語彙を手動で作成するのに比べコーパス内に未知語を残さないという利点がある。一方で今回の問題設定のように，汎用の事前学習言語モデルを作成するためのコーパスと Fine-Tuning するための限定ドメイン少量コーパスとのように，コーパスが作成されたドメインが大きく違う場合にはこの利点は限定的となる。つまり，当該ドメインでのみ頻出する固有表現が汎用の事前学習言語モデルの語彙に登録されていないためにサブワードに分割されて処理される一方，当該特定ドメインに出現する確率が低いトークンが登録され実質的な語彙数が狭められるなどの非効率が発生しているとも直感的に考えられるためである。

以下本稿では，このような事前学習言語モデルの語彙の直接的な操作が，Fine-Tuning 後のタスクの精度に与える影響を系統的に評価し，一般には本操作が Fine-Tuning の際のドメイン適応に寄与しないことを示す。

1） (株) 日立製作所 Hitachi, Ltd.
2） (株) 日立産業制御ソリューションズ
Hitachi Industry & Control Solutions, Ltd.

2 関連研究

事前学習言語モデルのドメイン適応に関して，教師なし学習を用いた手法のサーベイ論文として [5] がある。

特に [6] では，下流タスクで用いられるコーパスから自己教師学習の訓練データを作成し事前学習言語モデルに追加の学習を行うことで，当該コーパスのドメインやタスクへの適応を行う試みについて，ドメインごとに定量的な評価が行われている。

また，得られる特定ドメインのコーパス量が制限されている場合に，それらを upsampling した上で汎用ドメインのコーパスと合わせて事前学習を行うことで，日本語の事前学習言語モデルの当該ドメイン内での性能を上げる試みに [7] がある。

3 調査手法

3.1 語彙を操作する事前学習言語モデル

語彙を操作して影響を評価する対象の，汎用ドメインのコーパスで自己教師学習を行った事前学習言語モデルとして，東北大学が作成，公開したモデル [8](以下東北大 BERT) を採用した。このモデルは日本語の Wikipedia 記事をコーパスとして自己教師学習を行い，語彙数は 32,000 である。

また，特定ドメインのコーパスで自己教師学習がなされた事前学習言語モデルとして，東京大学が作成，公開したモデルである UTH-BERT-BASE-128[9](以下 UTH-BERT) についても，同様に語彙を操作した影響を下流タスクの精度で評価する。このモデルは，電子カルテをコーパスとした医療ドメインに適用されるもので，語彙数は 25,000 である。

3.2 評価する下流タスク

Fine-Tuning を行う対象のタスクとして，医療ドメインへの適応を擬するため，NTCIR-13 MedWeb[10] を採用する。これは，作成された日本語のツイートテキストについて，8 つの病気または症状 (Influenza, Diarrhea, Hayfever, Cough, Headache, Fever, Runnynose, Cold) のそれぞれのカラムに関する陽性または陰性のアノテーションが付された，8 マルチラベル 2 値分類タスクである。これらは，学習データ 1,920 発言，テストデータ 640 発言から構成されている。

以下ではこのマルチラベルタスクを同時に解くために，前述の事前学習言語モデルが文頭に付す [CLS] についての埋込ベクトルから，各カラムの陽性尤度を出力する総結合層を並列に 8 層追加し，教師あり学習を行うものを下流タスクとする。

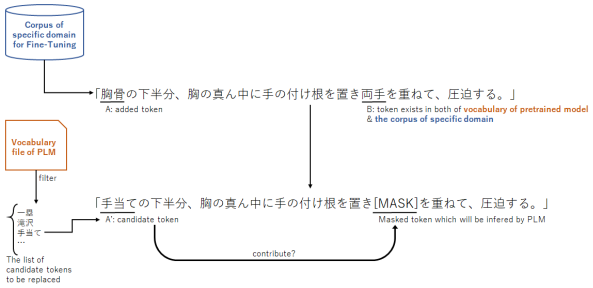

図 1　不要語特定のために作成するクエリの模式図

3.3　語彙の操作手法

下流タスクのドメインに頻出する固有表現 (以下追加語) を選択し，事前学習言語モデルの所与の語彙のうち下流タスクに現れないであろうトークン (以下不要語) を上書きする。

3.3.1　追加語の選択

上述の NTCIR-13 MedWeb 学習データに形態素解析を施し名詞を抽出する。それらのうち語彙に含まれないもので，出現頻度の高い語を追加語として選択する。出現頻度の閾値としては 3 以上および 6 以上を採用しそれぞれで評価する。ここで形態素解析には MeCab[11] を用い，ユーザ辞書として NEologd[12, 13, 14] および万病辞書 [15] を用いている。

3.3.2　不要語の選択

語彙のうち NTCIR-13 MedWeb 学習データでの出現頻度が 0 回の語であって，またその語単体で形態素解析をした際に名詞とされる語を候補とし，以下の 2 つの方式でそれぞれ不要語を選択し評価する。

- ランダムに選択する
- attention を用いたクエリを作成し，その結果から選択する。追加語の中から特に頻出の数語を A，NTCIR-13 Med 学習 Web データと語彙との両方に含まれる語を B とする。ここで不要語として特定したいのは，事前学習言語モデルの語彙にありながら B と文脈的関係を持たないトークンである。そこで A と B をともに含む文を MedWeb13 訓練データから取得し，このテキストから B をマスクする一方，事前学習言語モデルの語彙にあり，かつ NTCIR-13 MedWeb 学習データには含まれない，入替対象の候補となる語 A' を A の位置と入れ替え，クエリ (以下不要語選択用クエリ) を生成する。この問題文を事前学習言語モデルに入力しマスクされた B を推論させる際の，attention の値の低い A' を不要語とする。本方式の模式図を 1 に示す。

3.4　追加自己教師学習

語彙を操作したのち，NTCIR-13 MedWeb 学習データから自己教師学習用データを作成し，事前学習言語モデルに追加的に学習を施す。これを以下追加学習とする。この追加学習のハイパーパラメータは，多くを事前学習言語モデルの作成時 [8] と同じくし，学習率のみ 5e-6 へと調整する。また学習のステップ数も複数の候補でそれぞれ下流タスクの Fine-Tuning を行った精度を評価する。

4　評価結果

4.1　語彙の操作

3.3.1 で設定した条件で，出現頻度の閾値を 6 以上とした場合，43 語が追加語として選択される。うち，NTCIR-13 MedWeb 学習データ内での出現頻度が高かった 3 語は「咳が止まらない」「鼻水止まらない」「花粉症対策」である。これらは一般的には 1 語として扱われないが，症状などの表現が収集される万病辞書では 1 語としてそれぞれ登録されている。

出現頻度の閾値を 3 以上とした場合には，これらに加え計 91 語が追加語として選択される。

4.2　下流タスク精度

事前学習言語モデルの語彙を操作したうえで追加学習を行い，Fine-Tuning を施すことで，NTCIR-13 MedWeb のマルチラベル 2 値分類タスク用のモデルが得られる。この分類精度を使って，事前学習言語モデルの語彙への操作が下流タスクの精度に与える影響を評価する。

なお，以下で示す精度の値は，NTCIR-13 MedWeb 学習データの 8 カラム全てで分類結果が正答したデータを計数したものとする。また，下流タスクの学習について，ハイパーパラメータは，語彙を操作しない状態の各事前学習モデルに Fine-Tuning を施して作成した分類モデル (以下 baseline) に対し，学習データについての 5-fold validation に係る検証データでの正答率を基準に，簡単にハイパーパラメータサーチを行って得た値を採用する。このように得たハイパーパラメータを用いて，語彙を操作した事前学習言語モデルに Fine-Tuning を施して作成する分類モデルについても，同じく 5-fold validation に係る検証データでの正答率を基準に early stoppoing を行うという試行を，ランダムシードを変えながら 3 度ずつ行い，計 15 モデルに亘る正答率や各カラムの分類に関する F1 値の平均値で評価している。

4.2.1　不要語の選択方式と下流タスク精度

3.3.2 で示した 2 つの不要語選択方式による語彙の操作を行い，それぞれ追加学習を施し下流タスクの Fine-Tuning および精度の評価を行った。ここでは，事前学習言語モデルを東北大 BERT とし，追加語は出現頻度の閾値を 6 以上とした 43 語としている。

追加学習のステップごとの実験結果について，正答率と各カラムの F1 値とを 1 に示す。いずれも数値は，上段が前述の 15 モデルに係る平均値であり，下段が (不偏標準偏差) である。また，結果内で最も高い数値をそれぞれ太字で表記する。

このように，追加学習を行うことによって分類精度は若干低下するが，今回確認した範囲では有意な低下を見せるものではなかった。F1 値もカラムごとに傾向の大きな違いは認められず，特定の追加語を語彙に加えることによる効果も見られていない。また，この傾向は不要語の選択方式によっては変わらないことが分かった。

4.2.2　追加する語の数と下流タスク精度

次に，追加語の選択基準を定める出現頻度の閾値を変更し，追加する語の数と下流タスク精度との関係を評価する。4.1 で触れたように，出現頻度が 6 以上の 43 語を追加語とする場合 (4.2.1 に示したものと同じ) および出現頻度が 3 以上の 91 語を追加語とする場合，語彙を操作せずに追加学習のみを行った場合の下流タスク精度を表 2 に示す。なお，これらでの不要語の選択方式は，不要語選択用クエリを用いている。

表 1　不要語の選択方式と下流タスク精度

選択方式	Steps	Total acc.	Influenza	Diarrhea	Hayfever	Cough	Headache	Fever	Runnynose	Cold
Baseline		**82.083**	0.694	0.888	**0.879**	**0.919**	0.930	**0.793**	**0.902**	**0.893**
		(1.258)	(0.038)	(0.017)	(0.016)	(0.009)	(0.019)	(0.018)	(0.009)	(0.013)
Random	10k	80.573	0.689	0.882	0.862	0.900	0.930	0.779	0.885	0.879
		(1.863)	(0.048)	(0.024)	(0.021)	(0.021)	(0.018)	(0.023)	(0.022)	(0.012)
	20k	80.500	0.681	0.881	0.863	0.905	0.927	0.780	0.885	0.881
		(1.414)	(0.044)	(0.020)	(0.015)	(0.018)	(0.018)	(0.026)	(0.016)	(0.014)
	30k	80.760	0.695	0.887	0.858	0.908	0.928	0.778	0.887	0.884
		(1.624)	(0.036)	(0.013)	(0.030)	(0.018)	(0.016)	(0.024)	(0.014)	(0.016)
Querying	10k	80.115	**0.707**	0.889	0.862	0.912	0.910	0.777	0.879	0.879
		(1.502)	(0.037)	(0.015)	(0.021)	(0.010)	(0.022)	(0.025)	(0.015)	(0.016)
	20k	80.104	0.699	0.889	0.865	0.900	0.920	0.770	0.877	0.879
		(1.932)	(0.036)	(0.020)	(0.025)	(0.019)	(0.017)	(0.019)	(0.021)	(0.026)
	30k	80.844	0.696	**0.890**	0.864	0.912	**0.933**	0.774	0.884	0.879
		(1.650)	(0.044)	(0.023)	(0.024)	(0.011)	(0.016)	(0.033)	(0.017)	(0.018)

このように，追加する語の数を変更しても，全体の分類精度や各カラムでの F1 値に有意な差はなかった。

4.2.3　事前学習言語モデルの違いと下流タスク精度

語彙の操作を行う対象の事前学習言語モデルに，UTH-BERT を用いた場合の結果を表 3 に示す。

追加語の選択基準として，3.3.1 で設定した条件に出現頻度が 6 以上とすると，31 語が追加語として選択される。「咳が止まらない」「鼻水止まらない」「花粉症対策」など，東北大 BERT と共通の追加語も含まれる一方，汎用ドメイン向けに作られていた東北大 BERT の語彙に含まれていた「ネパール」「科学」などの語が，UTH-BERT の語彙に含まれないが NTCIR-13 MedWeb 学習データ内で頻出であるとして選択されることが特徴となる。

これらを語彙に加え，下流タスクの分類精度を評価した結果，および語彙を操作せずに追加学習のみを施した結果が表 3 となる。ただし，これらでの不要語の選択方式は，不要語選択用クエリを用いたものである。

東北大 BERT に操作を行った場合よりいずれも高い精度を示しているが，各実験結果の間に有意な差は認められていない。ただし，いずれも有意ではないものの，語彙の操作の有無で比較すると，語彙を操作しない場合に高い精度が出ていることが多く，その差は東北大 BERT に操作を行った場合より大きいことが観察される。

5　おわりに

本報告では，事前学習言語モデルの語彙を直接的に操作した際の，Fine-Tuning 後のタスクの精度に与える影響をいくつかの観点から評価し，いずれの条件の元でも精度に有意な差をもたらさないことを確認した。これによって，事前学習言語モデルをドメイン適応させる際に，語彙を直接操作する試みは，下流タスクの精度に一般には寄与しないことが分かった。

参考文献

[1] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, Vol. 30, pp. 5998–6008, 2017.

[3] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 66–71, 2018.

[4] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, 2016.

[5] Alan Ramponi and Barbara Plank. Neural unsupervised domain adaptation in nlp—a survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6838–6855, 2020.

[6] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. Don’t stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8342–8360, 2020.

[7] Shoya Wada, Toshihiro Takeda, Shiro Manabe, Shozo Konishi, Jun Kamohara, and Yasushi Matsumura. A pre-training technique to localize medical bert and enhance biobert. *arXiv preprint arXiv:2005.07202*, 2020.

[8] Tohoku University Inui-Suzuki Laboratory. Github - cl-tohoku/bert-japanese at v1.0.

[9] Yoshimasa Kawazoe, Daisaku Shibata, Emiko Shinohara, Eiji Aramaki, and Kazuhiko Ohe. A clinical specific bert developed with huge size of japanese clinical narrative. *medRxiv*, 2020.

[10] Shoko Wakamiya, Mizuki Morita, Yoshinobu Kano, Tomoko Ohkuma, and Eiji Aramaki. Overview of the ntcir-13: Medweb task. In *In Proceedings of the 13th NTCIR Conference on Evaluation of Information Access Technologies (NTCIR-13)*, pp. 40–49, 2017.

[11] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to japanese morphological analysis. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pp. 230–237, 2004.

[12] 佐藤敏紀, 橋本泰一, 奥村学. 単語分かち書き辞書 mecab-ipadic-neologd の実装と情報検索における効果的な使用方法の検討. 言語処理学会第 23 回年次大会 (NLP2017), pp. NLP2017–B6–1. 言語処理学会, 2017.

[13] 佐藤敏紀, 橋本泰一, 奥村学. 単語分かち書き用辞書生成システム neologd の運用 — 文書分類を例にして—. 自然言語処理研究会研究報告, pp. NL–229–15. 情報処理学会, 2016.

[14] Sato Toshinori. Neologism dictionary based on the language resources on the web for mecab, 2015.

[15] 2019 Social Computing Laboratory NAIST. Mednlp 医療言語処理グループ 万病辞書.

表 2　追加する語の数と下流タスク精度

#追加語	Steps	Total acc.	Influenza	Diarrhea	Hayfever	Cough	Headache	Fever	Runnynose	Cold
Baseline		**82.083**	0.694	0.888	0.879	**0.919**	0.930	0.793	**0.902**	0.893
		(1.258)	(0.038)	(0.017)	(0.016)	(0.009)	(0.019)	(0.018)	(0.009)	(0.013)
0	100	81.250	0.671	**0.896**	0.857	0.913	0.924	0.790	0.893	0.889
		(0.595)	(0.058)	(0.016)	(0.016)	(0.006)	(0.019)	(0.024)	(0.012)	(0.018)
	1k	81.656	0.682	0.894	0.859	0.916	0.932	0.794	0.891	0.886
		(1.371)	(0.042)	(0.010)	(0.028)	(0.013)	(0.020)	(0.022)	(0.010)	(0.016)
	10k	81.802	0.685	0.890	**0.882**	0.913	0.929	0.796	0.897	**0.896**
		(0.961)	(0.046)	(0.016)	(0.018)	(0.017)	(0.019)	(0.011)	(0.009)	(0.016)
	20k	81.708	0.692	0.886	0.865	0.905	0.937	**0.802**	0.895	0.889
		(1.361)	(0.056)	(0.021)	(0.021)	(0.015)	(0.014)	(0.017)	(0.013)	(0.014)
	30k	81.615	0.686	0.894	0.868	0.909	0.931	0.798	0.890	0.886
		(1.335)	(0.058)	(0.015)	(0.017)	(0.017)	(0.017)	(0.021)	(0.013)	(0.016)
43	100	80.500	0.705	0.887	0.864	0.914	0.915	0.779	0.880	0.881
		(1.845)	(0.029)	(0.015)	(0.029)	(0.013)	(0.014)	(0.029)	(0.023)	(0.013)
	1k	79.813	0.686	0.885	0.857	0.907	0.915	0.764	0.872	0.873
		(2.009)	(0.039)	(0.021)	(0.021)	(0.024)	(0.022)	(0.028)	(0.021)	(0.023)
	10k	80.115	**0.707**	0.889	0.862	0.912	0.910	0.777	0.879	0.879
		(1.502)	(0.037)	(0.015)	(0.021)	(0.010)	(0.022)	(0.025)	(0.015)	(0.016)
	20k	80.104	0.699	0.889	0.865	0.900	0.920	0.770	0.877	0.879
		(1.932)	(0.036)	(0.020)	(0.025)	(0.019)	(0.017)	(0.019)	(0.021)	(0.026)
	30k	80.844	0.696	0.890	0.864	0.912	0.933	0.774	0.884	0.879
		(1.650)	(0.044)	(0.023)	(0.024)	(0.011)	(0.016)	(0.033)	(0.017)	(0.018)
91	100	79.958	0.690	0.874	0.852	0.903	0.923	0.787	0.880	0.871
		(1.075)	(0.028)	(0.022)	(0.018)	(0.014)	(0.017)	(0.018)	(0.015)	(0.018)
	1k	79.833	0.694	0.867	0.850	0.896	0.927	0.789	0.881	0.881
		(1.675)	(0.048)	(0.022)	(0.021)	(0.018)	(0.019)	(0.028)	(0.013)	(0.016)
	10k	81.125	0.706	0.877	0.860	0.906	0.933	0.795	0.884	0.888
		(1.384)	(0.036)	(0.026)	(0.015)	(0.013)	(0.021)	(0.018)	(0.018)	(0.020)
	20k	80.729	0.689	0.881	0.859	0.894	0.929	0.796	0.886	0.889
		(1.440)	(0.041)	(0.019)	(0.024)	(0.020)	(0.025)	(0.020)	(0.017)	(0.011)
	30k	80.385	0.696	0.870	0.861	0.902	**0.938**	0.788	0.884	0.877
		(1.337)	(0.053)	(0.021)	(0.018)	(0.016)	(0.013)	(0.020)	(0.015)	(0.019)

表 3　UTH-BERT に語彙操作を施した場合の下流タスク精度

#追加語	Steps	Total acc.	Influenza	Diarrhea	Hayfever	Cough	Headache	Fever	Runnynose	Cold
Baseline		84.552	0.685	**0.926**	0.873	0.948	0.941	0.837	0.909	0.894
		(1.368)	(0.038)	(0.019)	(0.028)	(0.009)	(0.018)	(0.024)	(0.018)	(0.015)
0	10k	84.896	**0.718**	0.923	0.881	0.950	**0.945**	**0.845**	0.914	0.893
		(1.230)	(0.035)	(0.026)	(0.013)	(0.011)	(0.020)	(0.018)	(0.013)	(0.015)
	20k	**84.938**	0.717	0.922	**0.882**	0.951	0.943	0.844	**0.917**	0.895
		(1.056)	(0.052)	(0.018)	(0.010)	(0.011)	(0.012)	(0.023)	(0.010)	(0.016)
	30k	84.917	0.705	0.923	0.881	**0.952**	0.940	0.840	**0.917**	**0.900**
		(1.296)	(0.037)	(0.022)	(0.029)	(0.006)	(0.014)	(0.014)	(0.014)	(0.020)
31	10k	83.135	0.695	0.920	0.862	0.942	0.938	0.839	0.886	0.866
		(1.533)	(0.045)	(0.011)	(0.024)	(0.011)	(0.015)	(0.026)	(0.023)	(0.021)
	20k	82.344	0.695	0.910	0.859	0.940	0.936	0.836	0.875	0.861
		(1.210)	(0.054)	(0.014)	(0.030)	(0.008)	(0.022)	(0.028)	(0.027)	(0.035)
	30k	83.083	0.683	0.918	0.866	0.946	0.940	0.831	0.881	0.867
		(1.156)	(0.046)	(0.012)	(0.022)	(0.010)	(0.016)	(0.028)	(0.015)	(0.020)