通時的な領域適応を行った単語分散表現を利用した古文から現代文へのニューラル機械翻訳

高久雅史[1]　平澤寅庄[2]　小町守[2]　古宮嘉那子[1]

[1] 茨城大学　[2] 首都大学東京

16t4027n@vc.ibaraki.ac.jp, hirasawa-tosho@ed.tmu.ac.jp,
komachi@tmu.ac.jp, kanako.komiya.nlp@vc.ibaraki.ac.jp

1　はじめに

本稿ではニューラル機械翻訳 (NMT) を用いて古文から現代文への翻訳を行う。近年では深層学習と機械翻訳を合わせた NMT が盛んに研究されている。しかし、古文についての NMT 研究は行われていない。NMT は流暢な出力を出すことが知られていることから、現代文への翻訳の流暢さが上がることが期待される。しかし、一般に NMT は大規模パラレルコーパスを用いてモデル全体を学習させることで実現されるため、小規模パラレルコーパスでの学習では高い翻訳精度のモデルを得ることは難しい [2] と言われている。古文現代文のパラレルコーパスも同様に小規模であり、NMT には適さないと考えられる。

一方、十分な規模のパラレルコーパスがない言語対においては、翻訳モデルの性能を向上させるために、大規模単言語コーパスで事前学習された単語分散表現を使用してモデルを初期化する手法がある [5]。古文においても、同様の手法で翻訳モデルの改善が見込める。

良質な単語分散表現を獲得するためには、規模の大きい学習用コーパスで訓練することが望ましい。そのため、古文に比べより大規模なコーパスから学習された現代語の単語分散表現を利用したい。しかしその一方で、現代文の単語分散表現を翻訳モデルの古文単語の単語埋め込みにそのまま使用すると、ドメインが異なっているため、翻訳性能が落ちることが予想される。もう一つの問題として、古文現代文のパラレルコーパスは、異なる時代の作品データを含む通時コーパスであるため、時代によって単語の意味や形（表層形）が変化するという言語変化の問題がある。

そこで本研究では、現代文の単言語コーパスで学習された単語分散表現に対し、古文の単言語コーパスを使用して時代順に領域適応した後、翻訳モデルを初期化する手法を提案する。提案手法と比較するため、古文の単言語コーパス全部をまとめて、一度のみ fine-tuning した単語分散表現を使用したモデルでも実験を行った。実験の結果から提案手法が Baseline の BLEU を上回ったが、近代以降では徐々にスコアの低下がみられた。一方で、通時適応を行った翻訳モデルでは、特定の時代にしか現れない語彙を訳出することができ、翻訳品質の改善が見られた。

2　関連研究

星野ら [7] は古文と現代文の段落単位パラレルコーパスから、文分割のためのルールベースのスコア関数を用いて、文単位のパラレルコーパスを得る手法を提案した。それによって得られたパラレルコーパスでの統計的機械翻訳 (SMT) を行い、提案手法による文分割の有効性を示した。古文から現代文への SMT は星野らによって行われたが NMT での研究はされていないため、本研究は初の試みと言える。

言語処理において単語分散表現を用いた研究は盛んに行われている。ニューラル言語処理タスクでも単語分散表現は利用されるが、NMT タスクにおいてはあまり利用されない。これは大規模パラレルコーパスで訓練を行う場合、翻訳モデル自身が適切な単語分散表現を学習するためである。しかし、小規模パラレルコーパスにおける翻訳では、単語分散表現でモデルを初期化することで性能の改善が見込める。Qi ら [5] は小規模なパラレルコーパスしか持たない言語対で NMT モデルを訓練する際に、事前学習済みの単語分散表現を適用することで、翻訳精度が向上することを示した。

単語分散表現の fine-tuning については、柳沼ら [8] の研究が挙げられる。柳沼らは事前学習済みの大規模な単語分散表現とターゲットコーパスとの領域シフト

の問題に対して、ターゲットコーパスが小規模である場合に fine-tuning を行うことで、単語分散表現の質が向上することを語義曖昧性解消タスクにおいて示した。

また、単語分散表現の通時適応については、Kim ら [1] が通時コーパスを用いて単語分散表現を年ごとにさかのぼる形で学習させる手法を行った。学習によって得られた単語分散表現は、通時コーパスにおける言語変化の検出を可能とした。Kim らの研究から、通時適応させた単語分散表現は、時代による単語の意味変化を捉えることが可能であると考えられる。

3 古文コーパスを用いた単語分散表現の通時的な領域適応

本研究では、翻訳モデルの初期化に用いる単語分散表現を、現代文コーパスからスタートして古文単言語コーパスで新しい時代から古い時代にかけて時代順に fine-tuning する手法を提案する。時代を徐々に戻すのは、適応元の単語分散表現と適応先の単語分散表現との意味のずれが小さくなるようするためである。

単語分散表現の fine-tuning には、柳沼ら [8] の手法を用いる。事前学習された単語分散表現の時代順 fine-tuning には、近代（江戸以降）・室町・鎌倉・平安の 4 つの時代に分けた古文コーパスを利用する。単語分散表現の通時適応の流れは以下のようになる。

(1) 事前学習済み単語分散表現を
　近代のコーパスで fine-tuning する
(2) (1) を室町のコーパスで fine-tuning する
(3) (2) を鎌倉のコーパスで fine-tuning する
(4) (3) を平安のコーパスで fine-tuning する

この手続きで得られた (1)、(3)、(4)[1]の単語分散表現を翻訳モデルの単語埋め込み層の初期化に利用する。また、単語埋め込み層のパラメータは固定せず、翻訳モデルの学習によって更新が行われる。

4 実験

4.1 モデル

実験では LSTM をベースとした Attention 付き Encoder-Decoder モデルを使用した。実装にはオープンソースのニューラル機械翻訳ツールである OpenNMT[2]を利用した。ネットワーク構成は、中間層には 2 層の単方向 LSTM を、Attention 層には Global Attention [3] を利用する。Encoder-Decoder ともに、単語ベクトルサイズを 200、中間層の次元数を 512 とした。最適化アルゴリズムには Adam を使用し、学習率は 0.001 にした。学習の際にはモデルが扱う語彙を 20,000 に絞り、それ以外の未知語については`<unk>`トークンとして処理した。

単語分散表現の初期化は、翻訳モデルの単語埋め込み層の重みの初期値として利用することで実現できる。提案手法では、古文コーパスで通時適応させた事前学習済みの単語分散表現を、翻訳モデル Encoder の単語埋め込み層の重みの初期値に利用した。一方で翻訳モデル Decoder には、事前学習済みの単語分散表現を前処理せずにそのまま初期値として利用した。

モデルの評価には BLEU [4] を用いた。手法ごとに異なるシード値を与え、50,000 step の学習を行う。5,000 step ごとに開発データでの検証を行い、BLEU の値が最高となるモデルでテストした。異なるシードでの学習を 3 回行い、BLEU の平均スコアをモデル全体のスコアとした。

提案手法と比較するため、古文単言語コーパスをまとめて一度のみ fine-tuning を行った際の単語分散表現を利用するモデルでも実験を行った。

4.2 データセット

翻訳では、星野ら [7] が抽出したパラレルコーパスを利用した。このコーパスは、近代、鎌倉、平安の 3 時代の作品データで構成される通時コーパスであり、時代の内訳は（近代:鎌倉:平安）＝（4,577:30,075:52,032）文対である。元の 86,684 文対のパラレルコーパス（表 1）の分割割合は、先行研究のテストサイズに合わせて (学習:開発:テスト)=(82,591:2,093:2,093) の分割をした。分割の偏りをなくすため、時代の割合に合わせてランダムサンプリングで実験データセットを作成した。テストセットの用例数は表 2 となった。コーパスの単語分割には MeCab v0.996[3]を使用し、古文側には中古和文 UniDic v1.3[4]、現代文側には UniDic v2.3.0[4]を辞書として利用した。学習時のモデルへの入力・出力文の長さは 1 文あたり 100 語に制限した。

[1]翻訳に用いた対訳データが近代、鎌倉、平安の 3 時代の作品データで構成されていたため

[2]https://github.com/OpenNMT/OpenNMT

[3]https://taku910.github.io/mecab/

[4]https://unidic.ninjal.ac.jp/

	総文数	語彙サイズ	トークン数
古文	86,684	49,200	2,774,745
現代文		45,690	3,611,783

表 1: 古文現代文パラレルコーパス

時代	用例数
近代テストセット	123 文
鎌倉テストセット	739 文
平安テストセット	1231 文

表 2: テストセットの時代別の用例数

fine-tuning に用いるコーパスには、国語研究所より提供を受けた小学館新編日本古典文学全集より抽出した古文単言語コーパス約 13 万文（表 3）を用いる。古文コーパスの時代分けは、JapanKnowledge の新編日本古典文学全集タイトル一覧[5]を参照した。

4.3 単語分散表現

事前学習済みの単語分散表現には、大規模コーパス NWJC（国語研日本語ウェブコーパス）で学習された、新納ら [6] の nwjc2vec をベースとして利用した。約 258 億語からなるコーパスの NWJC で学習されており、形態素解析の辞書には UniDic が利用されている。nwjc2vec の fine-tuning には柳沼らの手法を、イテレーション 5、単語ベクトルサイズ 200、window サイズ 5 の設定で用いた。

5 実験結果

表 4 にテストデータ全体でのモデル別の BLEU と、提案手法における時代別テストデータでの BLEU を示す。まずモデル別の結果を見る。“Baseline” は事前学習された単語分散表現を使わず、古文・現代文パラレルコーパスのみで学習した NMT の性能である。また、“SMT” [7] は星野らが提案した古文現代文 SMT の結果の引用である。また、“nwjc2vec” は fine-tuning を用いない nwjc2vec を初期化に利用した結果を指す。

[5] https://japanknowledge.com/contents/koten/title.html

	総文数	語彙サイズ	トークン数
近代	22,485	25,584	544,293
室町	12,640	14,931	386,101
鎌倉	35,020	29,062	933,190
平安	59,744	29,520	1,543,102
計	129,889	55,332	3,406,686

表 3: 古文単言語コーパス

実験の結果から、提案手法では Baseline より BLEU が改善されたが、時代を遡るごとに BLEU の低下がみられた。また、(1)、(3)、(4) で得られた 3 つのモデルにでのアンサンブル翻訳[6]と、全古文コーパスで fine-tuning された単語分散表現を利用したモデルで精度の高かった上位 3 つを利用したアンサンブル翻訳も行った。提案手法のアンサンブル翻訳では Baseline から 2 ポイント以上 BLEU の改善がみられた。

次に時代別での結果を見る。表 4 の時代別の BLEU は、近代テストセットでは (1) 近代の単語分散表現を利用した際が最もよかった。鎌倉テストセットは (4) 近代→室町→鎌倉→平安の単語分散表現を利用したときが最もよく、(3) 近代→室町→鎌倉を上回った。平安テストセットは (1) 近代の分散表現を利用したときが最もよく、次に (3) 近代→室町→鎌倉の単語分散表現を利用したときで、(4) 近代→室町→鎌倉→平安の単語分散表現を利用したときが最も悪かった。

6 考察

提案手法モデルでの BLEU の大幅な向上は見られなかったが、翻訳結果にはいくつかの改善が見られた。時代順の通時適応による効果としては、主に表 5 のような語彙の訳出改善である。例えば例 (1) では “やまとうた” という “和歌” と訳せる単語を江戸までの通時適応では訳出できなかったが、それより前の時代では訳出が行えたことから、nwjc2vec が領域適応できたと考えられる。また領域適応を行った 2 つのアンサンブル結果について、調査を行った所、提案手法での翻訳結果では 4,548 異なり語、全古文コーパスでの fine-tuning では 4,543 異なり語の訳出が行われた。提案手法の方が僅かではあるが、多くの語彙を訳出して

[6] 複数のモデルで同時に予測を行った結果を組み合わせて、確率的に出力を決定する方法

手法	BLEU
SMT [7]	28.02
Baseline	19.22
nwjc2vec	19.16
全古文コーパス	19.24
+アンサンブル	20.94
(1) 近代	**19.43**
(3) 近代→室町→鎌倉	19.33
(4) 近代→室町→鎌倉→平安	19.29
+アンサンブル	21.59

時代別モデル	BLEU		
	近代	鎌倉	平安
(1) 近代	5.24	25.16	19.53
(3) 近代→室町→鎌倉	4.09	25.65	19.43
(4) 近代→室町→鎌倉→平安	3.59	25.76	19.40

表 4: テストデータ全体での BLEU：表上部、時代別テストデータでの BLEU：表下部

いると言える。実際の例としては、(2) の例文で確認できた。“騒がしき” という語彙を全古文コーパスでの fine-tuning では誤訳しているが、通時適応での結果では “騒がしい” と正しく訳出できたため、通時適応による効果が見られたと考えられる。

次に、時代別のテストコーパスでの結果から、時代順の通時適応の効果について考える。表 4 の結果より、「テストセットと同時代までさかのぼって fine-tuning した単語分散表現で初期化して学習した翻訳モデルがいつも優れている」という仮説は正しくないことが分かった。今回利用した柳沼らの fine-tuning の手法は、新しく追加学習に利用したコーパスにこれまでに学習していたコーパス中になかった単語が閾値以上現れた場合、新たな語彙として単語分散表現を作成しているため、時代をさかのぼればさかのぼるほど、分散表現のエントリ数は増える仕様となっている。そのため、時代をさかのぼるほど、未知語が減るはずであるが、近代テストセットと平安テストセットに関しては、未知語が最も多いはずの (1) 近代の単語分散表現を利用した際が最もよい結果となっている。ただし、1 位と 3 位の BLEU の差は近代テストセットにおいて 1.65 であり、鎌倉テストセットにおいて 0.6 であるのに対し、平安テストセットについては 0.13 であるため、平安テストセットにおいては、「どのモデルを使ってもそれほど違いはない」という結果であると言える。

7 おわりに

本研究では、古文現代文による初の NMT の試みと、単語分散表現の利用時に段階的な fine-tuning をした単語分散表現の初期化手法について提案した。実験の結果、通時適応した単語分散表現の利用によって、時代固有の語の訳出を可能としたり、他のモデルよりも多くの語彙を訳出できたことが確認された。一方で BLEU 自体は大きな向上は見られず、提案手法によって次第に下がる結果となった。今後の研究としては、翻訳対象の時代に合わせた通時適応の効果の調査が必要である。特に、ある時代までで fine-tuning した単語分散表現をその時代のパラレルコーパスの翻訳に利用して、どれくらいモデルが改善されるかを明らかにする必要がある。また、nwjc2vec ではない単語分散表現をベースとしたり、時代順の通時適応ではなく、時代別の通時適応などの手法についても調べる。

謝辞

本研究の一部は国立国語研究所の共同研究プロジェクト「通時コーパスの構築と日本語史研究の新展開」の研究成果を報告したものです。また、富士通研究所の横野光様には、先行研究のデータをいただきました。御礼申し上げます。

参考文献

[1] Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. Temporal analysis of language through neural language models. In *ACL*, 2014.

[2] Philipp Koehn and Rebecca Knowles. Six challenges for neural machine translation. In *ACL*, 2017.

[3] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, 2015.

[4] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *ACL*, 2002.

[5] Ye Qi, Devendra Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. When and why are pre-trained word embeddings useful for neural machine translation? In *NAACL-HLT*, 2018.

[6] 新納浩幸, 浅原正幸, 古宮嘉那子, 佐々木稔. nwjc2vec: 国語研日本語ウェブコーパスから構築した単語の分散表現データ. 自然言語処理, Vol. 24, No. 5, 2017.

(a) 鎌倉のコーパスまでの通時適応で翻訳が改善された例	
入力文:	やまとうた の 道 、 浅き に 似 て 深く 、
参照文:	和歌 の 道 は 、 浅い よう で い て じつ は 深く 、
Baseline:	<unk> の 道 は 、 浅い の に 似 て 深く 、
近代:	<unk> の 道 、 浅い の と 同様 に 、深く て 、
近代→室町→鎌倉:	和歌 の 道 は 、 浅い 時代 に 似 て 深く 、
(b) 提案手法のアンサンブル翻訳で翻訳が改善された例	
入力文:	大饗 に 劣ら ず 、 あまり 騒がしき まで なん 集ひ たまひ ける 。
参照文:	大 饗 の とき に 劣ら ない ほど 、 あまり に 騒がしい まで 大勢 お 集まり に なる の だっ た 。
Baseline:	大 饗 に も 劣ら ず 、 あまり に も あわただしい くらい に お 集まり に なっ た 。
全古文:	大 饗 に 負け ず 、 あんまり 暑い まで 集まっ て お いで に なっ た 。
通時適応:	大 饗 に 劣ら ず 、 あまり に 騒がしい まで 集まっ て お いで に なっ た 。

表 5: 翻訳のサンプル

[7] 星野翔, 宮尾祐介, 大橋駿介, 相澤彰子, 横野光. 対照コーパスを用いた古文の現代語機械翻訳. 言語処理学会第 20 回年次大会, 2014.

[8] 柳沼大輝, 古宮嘉那子, 新納浩幸ほか. 分散表現のファインチューニングによる語義曖昧性解消の領域適応. 研究報告自然言語処理 (NL), Vol. 2018, No. 1, pp. 1–5, 2018.