

T5 を用いた古文から現代文への翻訳

白井久生¹ 古宮嘉那子²東京農工大学工学部知能情報システム工学科¹東京農工大学大学院生物システム応用科学府²s190260u@st.go.tuat.ac.jp¹kkomiya@go.tuat.ac.jp²

概要

本稿では、日本語の事前学習済み T5 モデルを用いて古文から現代文への翻訳を行う。fine-tuning には、奈良時代から江戸時代までの文献の、古文と現代語訳の平行データを利用した。また複数の青空文庫の作品を利用し、逆翻訳により疑似平行データを作成し、これを fine-tuning の学習データとして追加的に利用することで、翻訳の性能に効果があるかを検証した。実験の結果、正規の平行データのみで行った場合は BLEU 値が 27.73 となり、統計的機械翻訳に迫る結果となった。また逆翻訳による疑似平行データも用いた場合は BLEU 値が 23.81 となり、古文の翻訳に逆翻訳による疑似平行データの利用は有効でないことが分かった。

1 はじめに

本稿では T5 を用いて、日本語の古文から現代文への翻訳を行う。日本語の古文から現代文への翻訳の先行研究には、統計的機械翻訳 [1] とニューラル機械翻訳 [2] による研究があるが、このうちのニューラル機械翻訳による研究は LSTM(Long Short Term Memory) を用いており、統計的機械翻訳の結果を超えることはできなかった。ニューラル機械翻訳は小規模なコーパスではよい結果が得られないとされており [3]、古文と現代文の平行コーパスは小規模であることが、ニューラル機械翻訳を用いた先行研究の性能の低さの主因であると考えられる。

一方で、深層学習は LSTM の後にも大きな進歩を見せており、2018 年に GPT[4] や BERT[5] が公開され、これらを受けて T5[6] が 2019 年に発表された。これらのモデルは大規模なコーパスを用いて事前学習を行っているため、言語的知識を有していると考えられ、これを Fine-tuning することでデータが少ないタスクでも性能の高いモデルが作成可能になった。

本稿ではこの事前学習済みモデルである T5 を用いて、古文から現代文への翻訳を行う。古文と現代文の平行コーパスは小規模であるため、LSTM を用いた研究においては十分な結果が得られなかったが、事前学習済みモデルを用いることでデータの少なさを補完できる可能性があると考えられる。本稿では、日本語の古文から現代文への機械翻訳において、T5 を用いることにより統計的機械翻訳に迫る結果を得られることを示す。

2 関連研究

古文の現代語への機械翻訳の例は、星野ら [1] の統計的機械翻訳と、高久ら [2] の LSTM, Transformer で行った例がある。星野らは古文と現代文の 86,684 文のコーパスを段落ごとに区切ったものを用いて、統計的機械翻訳を行った。高久らは LSTM を用いて、同じコーパスを現代から古代に向けて遡りながら学習を行い、ニューラル機械翻訳を行った。結果は、統計的機械翻訳では 28.02, LSTM では 19.95 であった。

T5 を用いた翻訳の先行研究には Emezue ら [7] の例がある。これは、mt5 を用いてアフリカの言語であるコサ語、ヨルバ語、イボ語などを英語やフランス語に翻訳したものである。これらのアフリカの言語はデータ数が少なく、最も多いコサ語で 158,660 件であり、英語との平行コーパスは 137,000 件ほどであった。コサ語の英語翻訳の BLEU 値は 30.25 であった。

3 T5 を用いた古文から現代文への翻訳

T5 とは、Transformer[8] をベースにして、C4 というコーパスを用いて事前学習を行ったモデルである。この C4 はウェブクロールしたデータを整形したものであり、このように高品質なデータを利用したことと事前学習手法、ファインチューニング手法の調整によって T5 は 2019 年当時 26 種のタスクで

最高性能を達成した。

本稿では日本語データで作成された T5 である `sonoisa/t5-base-japanese` を用いて、古文の現代文への翻訳を行った。Fine-tuning の際は、古文を学習データ、現代文を教師データとして学習を行い、古文を入力して現代文を出力するようにモデルを作成する。

4 データ

本稿では星野ら [1] が抽出したパラレルコーパスを用いる。このコーパスは古文作品とそれに対して人手で付与した現代語翻訳文が対になっている。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。このコーパスは全 86,684 文対であり、中には一つの古文に対して複数の現代語翻訳が対応しているものも含まれる。データの分割割合は、高久ら [2] にならない、(学習: 開発: テスト)=(82,591:2,000:2,093) 文対に分割をした。分割の偏りを避ける為に分割の際にはランダムに分割を行った。コーパスの単語分割には MeCab v0.996¹⁾ を使用し、古文側には中古和文 UniDic v1.3²⁾、現代文側には UniDic v2.3.0³⁾ を辞書として利用した。

5 実験

本稿では、T5 モデルを用いて古文から現代文への翻訳を行う。4 節で示したパラレルデータの古文を入力とし、現代文を出力として学習データをシステムに与え、Fine-tuning を行う。評価の際は学習したモデルに古文データを与え、出力した現代文翻訳データと正解となる現代文データで BLEU 値を算出した。T5 のモデルは `transformers` ライブラリの `sonoisa/t5-base-japanese`⁴⁾ を利用し、Automodel で指定した。また、Tokenizer には Autotokenizer で `sonoisa/t5-base-japanese` を指定した。BLEU 値の算出には `evaluate` ライブラリの `sacrebleu`⁵⁾ [9] を利用した。

1) <https://taku910.github.io/mecab/>

2) <https://clrd.ninjal.ac.jp/unidic/>

3) <https://clrd.ninjal.ac.jp/unidic/>

4) <https://huggingface.co/sonoisa/t5-base-japanese>

5) <https://huggingface.co/spaces/evaluate-metric/sacrebleu>

5.1 古文データを用いた実験

古文と現代文のパラレルデータを用いて T5 の fine-tuning を行った。この実験において調整を行ったパラメータは、学習率とエポック数、repetition penalty である。まず学習率とエポック数の組み合わせについて実験を行ってこれらを決定し、次に学習率と repetition penalty の組み合わせについて実験を行って repetition penalty を選定した。

学習率, エポック数 学習率とエポック数の設定は、repetition penalty を固定し、最も BLEU 値が高くなる組み合わせを探索した。この際、学習率は対数スケールでおおよそ均等になるように設定した。表 1 に組み合わせの候補を示す。なお、repetition penalty は 5 とした。

表 1 エポック数と学習率の組み合わせ

エポック数	学習率	エポック数	学習率
1	0.00001	5	0.0002
	0.0001		0.0003
	0.00015		0.0004
	0.0002		0.0005
	0.0003		0.0007
	0.0004		0.00089
	0.0005		0.00089
	0.0007		0.001
	0.00089		0.0015
	0.001		0.002
10	0.0015	0.0003	
	0.002	0.0004	
	0.003		
	0.01		

repetition penalty repetition penalty は、生成の際の繰り返しに対してペナルティを与えるパラメータである [10]。エポック数と学習率を固定し、repetition penalty について grid search を行った。学習率の候補は 0.0001, 0.0002, 0.0003, 0.0004 とし、repetition penalty の候補は 1 から 10 までの 1 刻み及び 1.1 から 2.5 まで 0.1 刻みとした。また、これら二つの実験から、学習率、エポック数、repetition penalty を決定し、最終的な実験を行った。

5.2 逆翻訳データを用いた実験

5.1 節の小学館コーパスから抽出した正規のパラレルデータのみによる学習の実験に加えて、逆翻訳データを学習データに加えた実験を行った。逆翻訳データの作成には、古文から現代文への翻訳同様、`sonoisa/japanese-t5-base` を用いた。パラメータは、予

備実験の結果、学習率を 0.0002, エポック数を 5, repetition penalty を 1.5 に設定した。予備実験の結果は付録の表 8 に示した。学習データを現代文、正解データを古文として現代文からの古文生成モデルを作成し、これに青空文庫から取得した複数の著者による複数の書籍データを入力として与え、古文訳を出力した。このデータは句点区切りで、全 522,801 件であった。この出力した古文と現代文をペアとし、疑似パラレルデータとして新たに古文から現代文への翻訳システムの学習を行った。この際、学習率を 0.0002, repetition penalty を 1.5 とした。

また、逆翻訳による疑似パラレルデータをパラレルデータと併用するにあたり、データを与える順番と正規のパラレルデータを与える回数を以下のように変えて実験した。

学習方法 1) すべての逆翻訳データの次にすべての正規のパラレルデータを与え、1 エポック回す。

学習方法 2) 逆翻訳データを n 分割し、分割したファイルの途中にすべての正規のパラレルデータを与えて学習する。結果として、逆翻訳データは 1 エポック、正規のパラレルデータは n エポック回す計算となる。

学習方法 2 は、正規のパラレルデータを途中途中に挟んで学習することで、翻訳エラーを減らすことを期待して考案したものである。なお、ライブラリの仕様上データは 2 万件の倍数ずつ入力するようになっているため、逆翻訳データを n 分割する際は等分に分割はせず、まず 2 万件の倍数ずつ入力し最後に残りのデータを入力する形で、全件を入力した。そのため、分割数 n と分割したファイルのデータ件数は表 2 のようになる。

表 2 逆翻訳の分割数 n と、分割後のファイルごとの疑似データ数

学習手法名	分割数 n	疑似データ数
学習手法 2-1	4	16 万件
学習手法 2-2	7	8 万件
学習手法 2-3	13	4 万件
学習手法 2-4	27	2 万件

例えば学習方法 2-1 では、まず逆翻訳データのうち 16 万件を与え、次にすべての正規のパラレルデータを与える。これを三回繰り返す。ここまでで逆翻訳データのうち 48 万件を利用している。逆翻訳データは 522,801 件なので、最後に残りの 42,801

件の逆翻訳データとすべての正規のパラレルデータを与えて学習する。結果的に、逆翻訳データは 1 エポック、正規のパラレルデータは 4 エポック回した計算となる。

6 結果

6.1 古文データを用いた実験の結果

各エポック数でもっとも BLEU 値が高かった際の BLEU 値をその際の学習率とともに表 3 に示す。その他を含むすべての結果は付録の表 7 に示す。

表 3 各エポック数の最良の BLEU 値

エポック数	学習率	BLEU 値
1	0.0007	25.18
5	0.0003	27.50
10	0.0002	27.58

学習率と repetition penalty の実験における最高の BLEU 値は表 4 のようになった。すべての結果は付録の表 9 に示す。また、その時のグラフを付録の図 1, 図 2 に示す。

表 4 各学習率の最良の BLEU 値

学習率	repetition penalty	BLEU 値
0.0001	1.4	22.82
0.0002	1.5	24.68
0.0003	1.5	25.41
0.0004	1.4	25.61

エポック数 10, 学習率 0.0002 のパラメータの際に BLEU 値が最高であることが分かる。また表 4 の結果から学習率が 0.0002 の際の最高の repetition penalty は 1.5 であることが分かる。そのため、エポック数を 10, 学習率を 0.0002, repetition penalty を 1.5 とした最終的な学習を行ったところ、BLEU 値は 27.73 となった。その際の生成例を表 5 に示す。

表 5 最終実験の翻訳例

入力	而るに、既に講の終る日に成て、道俗男女員不知ず参り集たり。
システムの出力	そのうち、いつしか講の終る日になり、僧俗・男女が数知れず集ってきた。
正解	さて、いよいよ講の終る日になると、僧俗・男女を問わず、数知れぬほどの人々が参詣してきた。

6.2 逆翻訳データを用いた実験の結果

現代文から古文を生成するモデルの BLEU 値は、27.03 であった。また、それぞれのデータの順序と与える回数を変えた学習方法について、結果を表 6 に示す。

表 6 逆翻訳データを用いた際のそれぞれの学習方法の最良の BLEU 値

学習方法	BLEU 値
1	23.81
2-1	19.60
2-2	19.05
2-3	18.35
2-4	18.11

7 考察

本稿の最良の BLEU 値は 27.73 となった。LSTM の BLEU 値 [2] は 19.95 であるので、これを 7.78 上回っている。このことから、日本語大規模データを用いて事前学習した T5 を用いることによって、データ数の少ない古文の現代文への翻訳に効果があるということが分かる。統計的機械翻訳の BLEU 値 [1] は 28.02 であり、残念ながら本研究の結果はこれに 0.29 ポイント及ばなかった。しかし、この差は僅差であり、統計的機械翻訳に迫る結果となった。

逆翻訳データを用いた実験の BLEU 値は、すべて古文データのみを用いた実験の最良の BLEU 値を上回ってはなかった。逆翻訳で生成したデータをチェックしたところ、古文らしいものではなかったため、学習の際にノイズとして働いてしまった可能性があると考えている。また、学習方法 2 は学習方法 1 よりも BLEU 値が低くなった。また、分割数を多くすると BLEU 値が下がる結果となった。このことから、逆翻訳データの間には正規の平行データを挟んで学習する手法は有効でないことが分かった。

さらに BLEU 値を上げる為に、エポック数などのパラメータのさらなる調整も考えられるが、本稿で行った調整よりさらに細かいパラメータの調整で BLEU 値が大きく上がることは考え難い。また、データ数を増やすことは、翻訳のプロの人手によるデータの作成に大きな手間がかかる。ただし、日本語の T5 の large サイズのモデルは筆者らの知る限り公開されていないが、さらに大きな事前学習モデルが公開されれば、統計的機械翻訳の結果を上回る可

能性があると考えられる。

8 まとめ

本稿では、事前学習済み日本語 T5 モデルを用いて古文から現代文への翻訳を行った。LSTM を利用した結果を 7.78 上回る結果となり、事前学習モデルである T5 を用いることで平行データの少なさを補完できることを示した。またこの結果は、関連研究のうちで最も高い BLEU 値を出した統計的機械翻訳のシステム [1] に迫る BLEU 値である。また、青空文庫から逆翻訳データを作成して、これを学習データに追加する実験を行ったが、この手法は古文から現代文への翻訳には有効でないことが分かった。

謝辞

本研究はJSPS科研費17KK0002, 18K11421, 22K12145の助成を受けたものです。また, 国立国語共同研究プロジェクト「開かれた共同構築環境による通時コーパスの拡張」「多様な語義資源を統合した研究活用基盤の共創」「アノテーションデータを用いた実証的計算心理言語学」の成果です。また, 明星大学の横野光先生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] 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.
- [2] 高久雅史, 平澤寅庄, 小町守, 古宮嘉那子. 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. 言語処理学会, 第26回年次大会, 2020.
- [3] Philipp Koehn and Rebecca Knowles. Six challenges for neural machine translation. In **Proceedings of the First Workshop on Neural Machine Translation**, pp. 28–39, Vancouver, August 2017. Association for Computational Linguistics.
- [4] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving Language Understanding by Generative Pre-Training.
- [5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019.
- [6] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. **J. Mach. Learn. Res.**, Vol. 21, No. 140, pp. 1–67, 2020.
- [7] Chris C. Emezue and Bonaventure F. P. Dossou. MMTAfrica: Multilingual Machine Translation for African Languages, April 2022.
- [8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need, December 2017.
- [9] Matt Post. A call for clarity in reporting BLEU scores. In **Proceedings of the Third Conference on Machine Translation: Research Papers**, pp. 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics.
- [10] Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. CTRL: A Conditional Transformer Language Model for Controllable Generation, September 2019.

表7 各学習率とエポック数での BLEU 値
(横軸: エポック数, 縦軸: 学習率),repetition penalty:5.0

	1	5	10
0.00001	16.24		
0.000089			27.12
0.0001	22.35		27.23
0.00015	23.41		
0.0002	24.27	27.48	27.57
0.0003	24.41	27.50	27.10
0.0004	24.07	27.23	24.07
0.0005	24.81	27.12	
0.0007	25.18	26.54	
0.00089	24.50	25.85	27.10
0.001	24.37		
0.0015	22.15		
0.002	17.12		
0.003	1.88		
0.01	1.01		

表8 逆翻訳モデル作成時の BLEU 値
(横軸: エポック数, 縦軸: 学習率),repetition penalty:5.0

	1	5
0.0001	20.51	25.92
0.0002	22.30	27.03
0.0003	24.83	26.65
0.0004	23.90	25.60
0.0005	22.90	25.87
0.0007	23.53	25.00

表9 repetition penalty ごとの BLEU 値
(縦軸:repetition penalty 横軸: 学習率 エポック数:1)

	0.0001	0.0002	0.0003	0.0004
1	22.72	24.63	25.24	25.55
2	22.90	24.63	25.32	25.43
3	22.80	24.41	25.23	25.27
4	22.57	24.23	24.90	25.00
5	22.46	24.12	24.76	24.94
6	22.35	24.04	24.64	24.77
7	22.22	23.97	24.61	24.67
8	22.08	23.78	24.52	24.58
9	21.98	23.70	24.40	24.44
10	21.9	23.57	24.24	24.44

表10 repetition penalty ごとの BLEU 値
(縦軸:repetition penalty 横軸: 学習率 エポック数:1)

	0.0001	0.0002	0.0003	0.0004
1	22.72	24.63	25.24	25.55
1.1	22.75	24.64	25.32	25.57
1.2	22.81	24.64	25.36	25.57
1.3	22.81	24.59	25.31	25.58
1.4	22.82	24.65	25.37	25.61
1.5	22.80	24.68	25.41	25.56
1.6	22.84	24.68	25.30	25.52
1.7	22.89	24.67	25.29	25.47
1.8	22.92	24.65	25.34	25.43
1.9	22.88	24.64	25.32	25.44
2	22.90	24.63	25.32	25.43
2.1	22.88	24.57	25.30	25.42
2.2	22.88	24.60	25.28	25.41
2.3	22.80	24.57	25.31	25.43
2.4	22.91	24.53	25.31	25.43
2.5	22.87	24.51	25.28	25.42

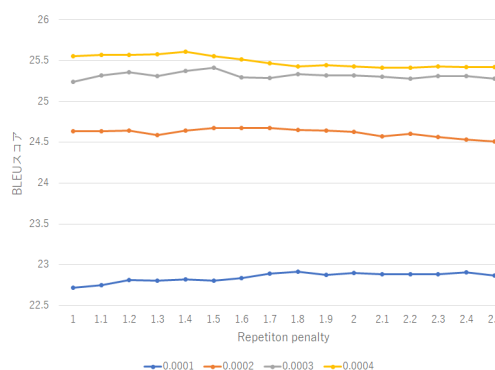


図1 学習率ごとの repetition penalty の変化による BLEU 値変化 (0.1 刻み)

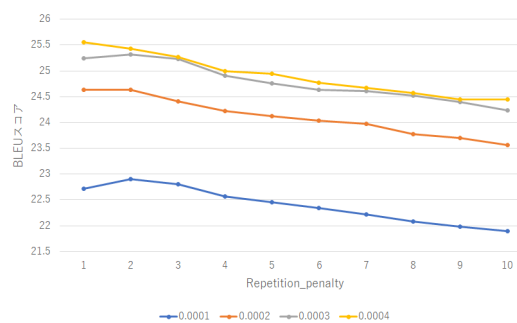


図2 学習率ごとの repetition penalty の変化による BLEU 値変化 (1 刻み)