

# 教師なし学習を用いた近代文語体と現代口語体の 相互翻訳の検討

藤井千香子<sup>†1</sup> 竹本有紀<sup>†1</sup> 石川由羽<sup>†2</sup> 高田雅美<sup>†1</sup> 城和貴<sup>†1</sup>

**概要**：近代文語体は時代によって文体が異なり，その文語体で書かれた文章は専門知識無しでは容易に読むことができない．本稿では教師なし学習を用いた機械翻訳で，近代文語体と現代口語体の相互翻訳に向け，実験を行う．単言語データや対訳データを用いた教師なし学習で翻訳精度向上を図り，その有用性を検証する．

## Examination of Mutual Translation between Modern Literary Style and Modern Colloquial Style using Unsupervised Learning

CHIKAKO FUJII<sup>†1</sup> YUKI TAKEMOTO<sup>†1</sup>  
YU ISHIKAWA<sup>†2</sup> MASAMI TAKATA<sup>†1</sup> KAZUKI JOE<sup>†1</sup>

### 1. はじめに

近代とは，明治から昭和初期にあたる時代であり，この時代に刊行された書籍はインターネット上に多く公開されている．近代文語体は，近代の間だけであっても，言文一致運動や戦後に行われた国語の大改革によって，文体が変化している．近代文語体で書かれた資料は，専門的な知識なしでは容易に読むことができない．また，資料は膨大に存在しているため，手動で翻訳するにはコストがかかる．そのため，機械で自動的に翻訳することが求められている．

本稿では，深層学習による教師なしニューラル機械翻訳[1]を用いて，近代文語体と現代口語体の相互翻訳を行う．機械翻訳の方式は，ルールベース機械翻訳[2]，統計的機械翻訳[3]の2つがメインであったが，現在はニューラル機械翻訳が主流となっている．はじめに，ルールベース機械翻訳とは，あらかじめ文法のようなルールを設定し，構文を解釈して翻訳を行う方法である．この方法は最初にルールの設定をする必要があり，時間と手間がかかる．つづいて，統計的機械翻訳とは，大量の対訳データを登録し，統計的に翻訳する方法である．近代文語体と現代口語体のテキスト化された対訳データはとても少なく，入手が困難である．そのため，この方法は近代文語体と現代口語体の翻訳に最適ではないと考える．最後に，ニューラル機械翻訳とは，ディープラーニングを利用する方法である．主な実例として，Google 翻訳[4]がこのニューラル機械翻訳を利用している．これは，人間が頭で言葉を理解することと同じように，機械が前後の文を参考に文脈や意味を考え，直訳ではない文意に合う翻訳を学習する．そのため，ほか2つの方法よ

りも流暢な訳文が生成される．ニューラル機械翻訳は，学習に使うデータの量は多い方が良い結果が出るが，少ない場合であっても十分理解できる翻訳が可能である．よって，ルール設定の必要がないことや，膨大な量の対訳データが用意できない場合でも品質の良い訳文が生成できるニューラル機械翻訳を，翻訳方法として採用する．

ニューラル機械翻訳において Sequence to Sequence モデル[5]という有名なモデルがある．これは，シーケンスと呼ばれる文章のペアを学習することで，片方のシーケンスから，それに対応するシーケンスを出力するモデルである．しかし，このモデルは長い文章への対応が困難であるので，Attention Mechanism[6]という手法が提案されている．これは，入力語と出力語の対応関係を学習して，翻訳文を出力する際，対応する入力単語を引き出すことで，長い文章の翻訳精度を向上させることが可能である．しかし，Sequence to Sequence モデルのような対訳データを使う教師あり学習には課題がある．それは，比較的マイナーな言語に対応できないことである．マイナーな言語の対訳データは少なく，少量のデータを学習させるとしても，翻訳精度には限界がある．よって，対訳コーパスを必要としない，完全な教師なし学習での翻訳精度の向上が求められている．そこで，対訳関係のない単一言語データによって，どの程度まで翻訳ができるか明らかにするため，教師なしのニューラル機械翻訳を用いて実験を行う．

本稿では，2章で翻訳手法である教師なしニューラル機械翻訳について述べ，3章では，2章で紹介した手法を用いて実験を行い，考察を述べる．

<sup>†1</sup> 奈良女子大学 Nara Women's University  
<sup>†2</sup> 滋賀大学 Shiga University

## 2. 翻訳手法

本稿に用いる翻訳手法は、教師なし学習を用いたニューラル機械翻訳である。2種類の単言語コーパスにより翻訳モデルを構築する。本章では、翻訳モデルを使って学習を始める前に必要な翻訳単語のベクトル化や、クロスリンガル単語埋め込みの生成について述べる。

### 2.1 翻訳モデルの生成手順

近代文語体と現代口語体の相互翻訳を行う翻訳モデルの生成は、以下のような手順で実施する。

- 手順Ⅰ. データ収集
- 手順Ⅱ. データの分かち書き
- 手順Ⅲ. 単語のベクトル化
- 手順Ⅳ. クロスリンガル単語埋め込みを計算
- 手順Ⅴ. モデルを用いて学習
- 手順Ⅵ. 翻訳モデルの評価

手順Ⅰではデータ収集を行う。近代文語体のデータは、青空文庫[7]から集める。青空文庫とは、著作権が消滅した作品や著者が許諾した作品をインターネット上で公開している電子図書館のことである。この青空文庫から、近代に活躍した作家の作品を集め、学習データに使用する。現代口語体のデータは同じく青空文庫から1945年以降に刊行された書籍の文章や、日本語版 Wikipedia[8]の文章を集める。Wikipediaとは、ウィキメディア財団が運営しているインターネット百科事典である。日本語版 Wikipediaの文章は、形態素解析器の辞書や、AIに知識を与えるためのデータとして活用されており、データベース・ダンプ[9]での提供が行われている。このデータベースから文章データを収集する。また、対訳データでの実験を行うとき、近代に刊行された邦字新聞を利用する。邦字新聞とは、日本以外の世界各国で発刊されている日本語新聞のことである。

手順Ⅱではデータの分かち書きを行う。形態素解析を行い、分かち書きにすることで機械処理を容易にする。形態素解析エンジンには、Mecab[10]を用いる。Mecabの辞書として、近代文語体のデータには近代文語 UniDic[11]、現代口語体のデータには UniDic[11]を用いる。

手順Ⅲでは単語のベクトル化を行う。翻訳モデル生成には、2種類の単言語コーパスとクロスリンガル単語埋め込みのデータが必要である。各言語の単語ベクトルを同一空間上に写像し、意味が似ていると思われる単語ベクトル同士の値を近似する。クロスリンガルな単語埋め込みとは、ベクトルの値を近似して、計算し直されたベクトルのことを指す。このクロスリンガル単語埋め込みは、各単語ベクトルから計算するため、手順Ⅲで単語ベクトルを生成する。使用するツールは、fastText[12]である。fastTextとは、

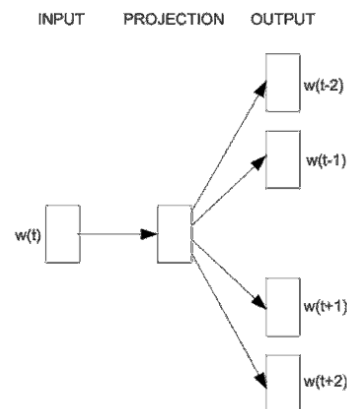


図1 skip-gram モデル

Facebookが開発した自然言語処理の学習を行うツールである。単語のベクトル化の詳細は2.2節で述べる。

手順Ⅳではクロスリンガル単語埋め込みを計算する。使用するツールは Artetxe らが開発した vecmap[13]である。クロスリンガル単語埋め込みの計算方法は2.3節で述べる。

手順Ⅴではモデルを用いて学習を行う。使用するモデルは Artetxe らが開発した教師なしニューラル機械翻訳のモデル[1]である。学習に用いるデータは、手順Ⅱで作成する2種類の言語の分かち書きデータと、手順Ⅳで作成する各言語のクロスリンガル単語埋め込みである。数万回の学習を行って、モデルの改良を重ねる。

手順Ⅵで翻訳モデルを評価する。評価尺度は BLEU[14]を用いる。この BLEU の値が大きいほど、翻訳精度が高いことを示す。

### 2.2 単語のベクトル化

単語は、機械で処理する時、必ず数値として表現される。その数値表現の1つにベクトルがある。単語をベクトル化するモデルは大きく分けて2つ存在する。それは、CBOW(Continuous Bag-of-Word)モデル[15]と skip-gram モデル[16]である。この2つのモデルを比較すると、単語ベクトルの精度において、skip-gram モデルの方が良い結果が得られることが多い。特に、データが大きくなるにつれて、低頻出の単語の対応や、類推問題の性能の点で skip-gram モデルが優れている。したがって、本稿では、skip-gram モデルを用いて単語のベクトル化を行う。

skip-gram モデルは、入力された単語をもとに周辺語を出力する。構造は、入力層、隠れ層、出力層から成るシンプルなニューラルネットワークである。その概略図を図1に示す。ある単語を入力データ、その周辺語を教師データにして、単語の重みを学習する。また、skip-gram モデルは式1のように定式化ができる。

$$\max_N \frac{1}{N} \prod_{n=1}^N \prod_{-m \leq l \leq m, l \neq 0} P(w^{(n+l)} | w^n) \quad (1)$$

$w^n$  を今注目する単語として、周囲の  $m$  個の単語について出現する条件付き確率を計算する。この条件付き確率を文全体を通して最大化する。N は文の長さを表す。また、式 1 中の確率は式 2 で表す。

$$p(w_o | w_q) = \frac{\exp(u_o^N v_q)}{\sum_{r \in V} \exp(u_r^N v_q)} \quad (2)$$

ここで、語彙のリストを  $V$ 、注目する任意の単語を  $w_q$ 、周辺の単語を  $w_o$  とする。また、 $w_q$  を表すベクトルを  $v_q$ 、 $w_o$  を表すベクトルを  $u_o$  と表す。これは、 $w_q$  の周囲に  $w_o$  が出現する確率をソフトマックス関数を用いて表したものである。式 2 の分母の式から分かるように、全ての語彙の出現確率を求める必要があるため、計算量が非常に大きい。よって、skip-gram では Negative Sampling[16] という計算を高速化する手法が一緒に用いられている。

### 2.3 クロスリンガル単語埋め込みの生成

クロスリンガルな単語埋め込みとは、単一言語コーパスを使用して異なる言語ごとに分散表現を個別に訓練し、線形変換によって共有空間にマッピングすることで得られる分散表現のことである。教師あり学習で辞書を必要とする手法がほとんどであるが、本稿では、教師なし学習で埋め込みを計算する手法[17]を利用する。

$X$  と  $Z$  を 2 つの言語の単語埋め込み行列として、 $\alpha$  番目の単語埋め込みをそれぞれ  $X_\alpha$ 、 $Z_\alpha$  と表す。マッピングされた埋め込み  $X A_X$  と  $Z A_Z$  が同じ言語空間にあるように、線形変換行列  $A_X$  と  $A_Z$  を学習することを目標としている。また、目的言語の  $\beta$  番目の単語が、原言語の  $\alpha$  番目である単語の翻訳の場合、 $D_{\alpha\beta} = 1$  である両言語間の辞書  $D$  の構築を目指す。

初めに、各言語の単語ベクトルを正規化し、多次元尺度法(Multi-Dimensional Scaling)によってデータを中心化する。そして、再度、単語ベクトルを正規化する。次に、初期解を生成して、2 つの言語の語彙を対応させる最初の辞書を構築する。元の単語埋め込み  $X$  と  $Z$  はそれぞれ異なるため、対応する類似行列  $M_X = X X^T$  と  $M_Z = Z Z^T$  を考える。そして、 $M_X$  と  $M_Z$  について各行の値を並べ替える。結果として、ソートされた  $\text{sorted}(M_X)$  と  $\text{sorted}(M_Z)$  の行列が得られる。 $\text{sorted}(M_X)$  の行に最近傍検索を適用し、対応する翻訳を検索できるようになる。初期解を生成後、解が最適になるように学習を行なっていく。学習は収束するまで、以下の式 3 と式 4 による 2 つの計算を繰り返す。

$$\text{argmax} \sum_{\alpha} \sum_{\beta} D_{\alpha\beta} ((X_{\alpha} A_X) \cdot (Z_{\beta} A_Z)) \quad (3)$$

式 3 で現在の辞書  $D$  を最大化する直行マッピングを計算する。また、式 4 で最近傍検索を行う。

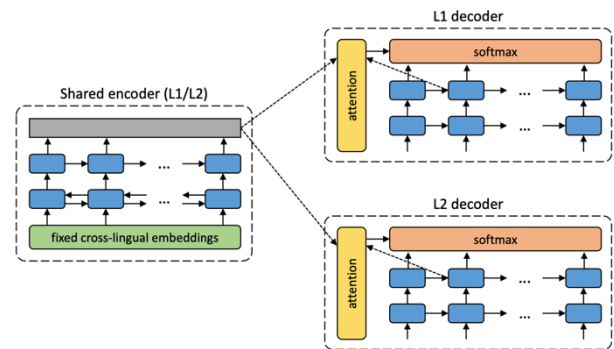


図 2 教師なしニューラル機械翻訳モデル

$$D_{\alpha\beta} = 1 \text{ if } \beta = \text{argmax}_k (X_{\alpha} A_X) \cdot (Z_k A_Z) \quad (4)$$

最後に、それぞれの相互相関にしたがって、再度、重み付けをする。

### 2.4 教師なしニューラル機械翻訳

教師なしニューラル機械翻訳[1]では、対訳データを用いず、2 種類の単一言語コーパスにより翻訳モデルを構築する。ノイズ除去と逆翻訳による擬似対訳コーパスを利用して学習を行う手法が提案されており、完全な教師なし学習が実現している。翻訳モデルの概略図を図 2 に示す。図 2 から、翻訳モデルが共有エンコーダと言語別デコーダから構成されていることがわかる。エンコーダで 2 層の双方向 RNN を使用し、デコーダでは別の 2 層の RNN を使用する。全ての RNN は 600 の隠れユニットを持つ GRU(Gated Recurrent Unit)[18]セルを用いている。また、長い文章にも対応できるように、Attention Mechanism[6]も用いられている。

提案モデルには、一般的なニューラル機械翻訳モデルと異なる箇所が 3 つ存在する。1 つ目は、両方向の翻訳を処理する点である。例えば、通常、フランス語から英語、英語からフランス語のような特定の翻訳方向で翻訳モデルは構築される。しかし、今回の手法では、フランス語と英語の両方向を処理する。2 つ目は、エンコーダを共有している点である。これは、入力データの言語に依存しない表現を生成することを目的としており、デコーダはそれぞれ対応する言語に変換する必要がある。3 つ目は、事前学習済みのクロスリンガル単語埋め込みをエンコーダで使用する点である。多くの機械翻訳モデルは、埋め込みをランダムに初期化し、学習時に更新しているが、教師なしニューラル機械翻訳モデルのエンコーダは、学習済みのクロスリンガル単語埋め込みを使用している。エンコーダでは言語に依存しない単語レベルの表現を構成し、大きなフレーズの表現を構築する学習が行われる。

次の 2 つの手法を利用して、教師なしで学習が可能となっている。1 つ目は、学習にノイズ除去を用いる手法である。この手法は、エンコーダが言語に依存しない方法で入

力語の構成を学習するために、入力文にランダムノイズを導入する。翻訳モデルは、ランダムノイズが入った入力文を、元の文と同じになるように再構築し、ノイズを除去することで学習を行う。そのため、連続する単語間でランダムにシャッフルすることにより、入力文の単語の順序を変更する。このように、正しい語順を回復できるように、言語の内部構造を学習する。また、入力文の語順に依存しすぎないようにすることで、言語間で実際の語順の相違が明らかになる。2つ目は、学習に逆翻訳を用いる手法である。この手法は、学習途中の翻訳モデルを使って文章をもう1つの言語に翻訳し、その翻訳した文章を元の言語に再び翻訳することで擬似対訳文を生成する。この擬似対訳文から元の文章を予測するように、学習が行われる。学習が進んで翻訳モデルが改善されると、逆翻訳によって、より良い擬似対訳文が生成され、次の学習で翻訳モデルがさらに改善される。

### 3. 実験と考察

本稿では、近代文語体と現代口語体の相互翻訳に向け、教師なしニューラル機械翻訳モデルを用いて、相互翻訳を試みる。翻訳の実験環境やその結果と考察について、本章で述べる。

#### 3.1 実験環境

本稿では、学習回数、学習データ数、バッチサイズ、最適化関数、学習データ比率を変更して、翻訳精度の比較を行う。翻訳モデルのパラメータについて、学習率の初期値は0.0002、RNNの層数はエンコーダ、デコーダともに2層、層のユニット数は600である。学習データは第2章で述べるように、近代文語体、現代口語体はそれぞれ青空文庫、日本語版 Wikipedia から収集したものと、邦字新聞を使用する。また、翻訳モデルの評価に利用する近代文語体と現代口語体の対訳データは、実験データに応じて3種類使用することを表1に示す。表1の①と②の評価データは近代語訳が1892年に刊行された森鷗外の文学作品「即興詩人」[19]、現代語訳が1960年に刊行された神西清の文学作品「即興詩人」[20]を用いる。表1の③と⑤の近代文語体学習データは、近代の中でも比較的翻訳が容易とされている、およそ大正後期から昭和初期の文体を集めたものである。そのため、評価データには、1927年に刊行された芥川龍之介の文学作品「浅草公園」[21]を用いる。現代語訳した文章は、ネット上に存在していないため、送り仮名やカナ表記は人為的に変更を加えたものとしている。最後に、④は学習データが対訳の邦字新聞であるため、学習用とは別に評価用に割り当てているデータを用いる。翻訳精度の評価尺度には、BLEU[14]を用いる。BLEUとは機械翻訳と参照訳がどの程度似ているかを表す翻訳尺度であり、0~1の実数で表

表1 学習データ数とその評価データ

	近代文語	現代口語	評価データ
①	20000 文	20000 文	即興詩人
②	300000 文	300000 文	即興詩人
③	700000 文	700000 文	浅草公園
④	28000 文	28000 文	邦字新聞
⑤	700000 文	200000 文	浅草公園

表2 バッチサイズ変更による BLEU の変化

バッチサイズ	近代文語から 現代口語	現代口語から 近代文語
4	0.16	0.07
8	0.32	0.05
16	0.12	0.11
32	0.19	0.09
50	0.13	0.07
64	0.13	0.07

現される。本稿では、その実数を100倍した値をBLEUスコアと述べる。この値が大きいほど、良い翻訳文であると判断される。表1の①の時、使用する計算機はIntel®Core™2DuoCPU E8400@3.00GHz×2である。OSはUbuntu18.04、使用アプリケーション、ライブラリのバージョンはPython3.7.3, PyTorch1.4.0を用いる。それ以外の時、使用する計算機は、Intel®Core™i6-9700 CPU@3.00GHz×8、GPUは、NVIDIA GeForce RTX2080Tiである。OSはUbuntu20.04.3 LTS、使用アプリケーション、ライブラリのバージョンはPython3.8.10, PyTorch1.7.1を用いる。

#### 3.2 実験結果と考察

様々な条件下で実験を行い、翻訳精度の変化を明らかにする。学習回数や学習データなどの実験条件について、文章のはじめに明記する。

まず、表1の①の学習データ数を用いて、最適化関数はAdam[22]、学習回数を10000回とした時の、バッチサイズ変更によるBLEUの変化を表2に示す。表2の実験結果から、最高BLEUスコアはバッチサイズが8の際、0.32ポイントである。近代文語から現代口語に翻訳する場合、バッチサイズ8が最適であり、現代口語から近代文語に翻訳する場合、バッチサイズ16が最適であることがわかる。次に、最適化関数変更によるBLEUの変化を表3に示す。学習データ数は表1の②であり、学習回数は300000回である。最適化関数には、Adam[22]、Adagrad[23]、RMSprop[24]、確率的勾配降下法[25]、Adadelta[26]を用いる。最適化関数がAdamの際、最高でBLEUスコア0.30ポイントを獲得していることがわかる。その次に、0.03ポイントの差でAdagradが良い結果となった。

表3 最適化関数変更による BLEU の変化

最適化関数	BLEU
Adam	0.30
Adagrad	0.27
RMSprop	0.23
確率的勾配降下法(SGD)	0.16
Adadelta	0.07

表4 各条件で実験した時の BLEU

	近代文語から 現代口語	現代口語から 近代文語
1対1	0.89	1.67
1対2	0.30	0.95
2対1	1.86	2.08
邦字新聞	7.04	6.62
邦字新聞 (改行あり)	9.66	9.18
現代文学	4.03	3.34

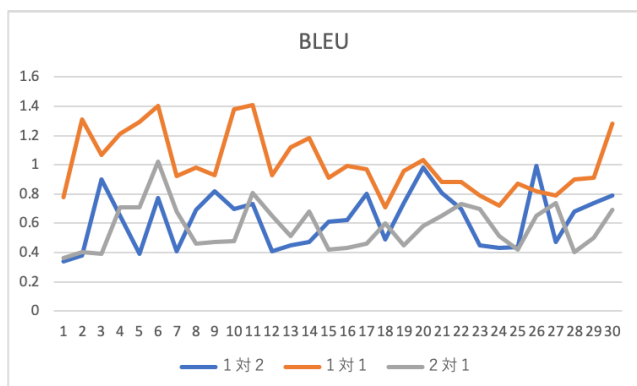


図3 学習データ比率を変更した時の BLEU の変化

近代文語と現代口語の学習データ比率を変えた時の BLEU の変化を図3に示す。図3の縦軸は BLEU スコア、横軸は学習回数 10000 回から 300000 回までを表す。バッチサイズ 32, 最適化関数は Adam, 学習回数 300000 回固定で、1対1の場合、学習データ数は表1の③である。近代文語データ数は 700000 文に固定して、現代口語データ数を変動させる。1対2の場合、現代口語データを 1400000 文に変更し、2対1の場合、現代口語データを 350000 文に変更して、実験を行う。図3の橙色のグラフ線を見ると、1対1の場合が他の対数比で実験する時よりも、全体的に翻訳精度が高いことがわかる。また、300000 回学習後、テストデータで BLEU スコアを算出する際の数値を表4に示す。学習データ比率 2対1 で近代文語から現代口語に翻訳する時、最高 BLEU スコア 1.86 ポイントを獲得している。また、同じ比率で現代口語から近代文語に翻訳する時、最高 BLEU スコア 2.08 ポイントを獲得している。ゆえに、近

代文語体と現代口語体の相互翻訳に最適な学習データ比率は 2対1 であることがわかる。

次に、近代に刊行された邦字新聞の対訳データを使って実験を行う。バッチサイズ 32, 最適化関数は Adam, 学習回数は 300000 回に設定している。その結果をテストデータの改行がある場合とない場合の 2 パターンに分けて、表4に示す。2種類の単一言語データで実験していた場合と比べて、BLEU スコアがおよそ 6~7 ポイント上昇していることがわかる。初めに、テストデータの翻訳を行う時、比較的長い文章の翻訳ができていないことが顕著に現れているため、BLEU スコアは 7.04 と 6.62 ポイントである。よって、テストデータの中で長い文章がある際、読点の箇所を改行を追加し、テストデータの改変を行う。改行を追加する際の BLEU スコアは 9.66 と 9.18 ポイントになり、およそ 2~3 ポイント上昇している。そのときの翻訳例を表5に示す。1行目は、近代文語から現代口語へ翻訳する場合の入力文と出力文と参照訳である。「申し込まれたしと」の翻訳結果が参照訳と一致しており、現代口語への翻訳が成功しているといえる。2行目は、現代口語から近代文語へ翻訳する場合の入力文と出力文と参照訳である。翻訳結果が参照訳と全文一致しており、近代文語への翻訳が成功しているといえる。

表1の⑤の学習データ数で、近代文語データに近代書籍 700000 文、現代口語データに日本語 Wikipedia ではなく、現代の文学作品 200000 文を使用し、実験を行う。バッチサイズ 32, 最適化関数 Adam, 学習回数は 300000 回に設定している。その結果を表4に示す。BLEU スコア 4.03 と 3.34 ポイントを獲得しており、対訳データで実験する時よりも劣るが、現代口語データに日本語 Wikipedia を使用している時よりも、優れた結果が出ている。また、その時の翻訳例を表6に示す。文中の<OOV>は未知語を表している。1行目は、近代文語から現代口語へ翻訳する時の入力文と出力文と参照訳である。「当る」の翻訳結果が参照訳と一致しており、現代口語への翻訳が成功しているといえる。この送り仮名の付き方は、大正から昭和初期の文体によく見られる特徴である。2行目は、現代口語から近代文語へ翻訳した時の入力文と出力文と参照訳である。「あたる」の翻訳結果が参照訳と比べると、漢字表記は異なっているが、近代文語らしく翻訳がされているといえる。

#### 4. まとめ

本稿では、教師なし学習を用いて近代文語体と現代口語体の相互翻訳を目指し、翻訳実験を行う。近代文語体と現代口語体の対訳データを獲得することが困難であるため、対訳関係のないデータを使用し、教師なしニューラル機械翻訳を構築する。バッチサイズ, 最適化関数, 学習データ比率を変更して実験を行い、翻訳精度を比較する。また、

表5 学習データが邦字新聞の時のテストデータ入出力結果

	入力文	出力文(参照訳)
近代文語から 現代口語	山口 兩 夫人 まで 申し込ま れ た し と	サクラメント 魚 店 まで 申し込ん で いた だ き た い の こ と である 。 (山口 兩 夫人 まで 申し込ん で いた だ き た い の こ と である 。)
現代口語から 近代文語	プログラム は 左 の 通 り である 。	プログラム は 如 左 (プログラム は 如 左)

表6 学習データが現代文学作品の時のテストデータ入出力結果

	入力文	出力文(参照訳)
近代文語から 現代口語	その うち に 息 の 当 る せ い か 、 顔 だけ ぼんやり と 曇 っ て し ま う 。	それ に 言葉 の あたる 全部 か 、 無 電 組 だけ <OOV> し ま う 。 (その うち に 息 の あたる せ い か 、 顔 だけ ぼんやり と 曇 っ て し ま う 。)
現代口語から 近代文語	それ から この 人形 に あたる コルク の 弾 丸 。	その とき 私 の なか に 当 る 犯人 。

学習データには2種類の単一言語データだけではなく、邦字新聞の対訳データや、現代口語データに現代文学作品の文章データを使用し、翻訳結果を比較する。最高 BLEU スコアは、邦字新聞のデータを用いて実験を行う時であり、近代文語から現代口語に翻訳する場合の BLEU スコアは 9.66 ポイントである。現代の文学作品を学習データとする時、データ数が少ないため、BLEU スコアは 4.03 ポイントである。データ数を増やすことが可能であれば、翻訳精度の向上が期待できる。本稿では、近代文語体の中でも比較的翻訳が容易である、大正から昭和初期の文体に着目し、データを集めて実験を行う。今後は、明治初期や明治後期の文体にも対応できるような翻訳モデルの構築を検討する。

## 参考文献

- [1] Mikel Artetxe, Gorka Labaka, Eneko Agirre, Kyunghyun Cho, Unsupervised Neural Machine Translation, ICLR 2018
- [2] Sholom M. Weiss et. al. Rule-based Machine Learning Methods for Functional Prediction. Journal of Artificial Intelligence Research 3.1995.383-403p.
- [3] Phillip Koehn et. al. Statistical phrase-based translation. NAACL HLT vol.1. 2003.48-54p.
- [4] Google 翻訳(online)  
<https://translate.google.com/about/>(参照 : 2021/11/10)
- [5] Sutskever et al., Sequence to Sequence Learning with Neural Networks, NIPS 2014
- [6] Dzmitry Bahdanau, KyungHyun Cho Yoshua Ben-gio. NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE. ICLR.2015
- [7] 青空文庫(online)  
<https://www.aozora.gr.jp/>(参照:2021/11/10)
- [8] ウィキペディア日本語版(online)  
<https://ja.wikipedia.org/wiki/ウィキペディア日本語版>(参照: 2021/11/10)
- [9] Wikipedia:データベースダウンロード(online)  
<https://ja.wikipedia.org/wiki/Wikipedia:データベースダウンロード>(参照:2021/11/10)
- [10] Mecab(online)  
<http://taku910.github.io/mecab/>(参照:2021/11/10)
- [11] UniDic(online)  
<https://unidic.ninjal.ac.jp/>(参照:2021/11/10)
- [12] fastText(online)  
<https://fasttext.cc/>(参照:2021/11/10)
- [13] vecmap(online)  
<https://github.com/artetxem/vecmap>(参照:2021/11/10)
- [14] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu. BLEU: a method for automatic evaluation of machine translation. In ACL, 2002.
- [15] T. Mikolov, K. Chen, G. Corrado, J. Dean. Efficient Estimation of Word Representation in Vector Space. In ICLR, 12 pages, 2013.
- [16] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In Advances in NIPS, pp. 3111-3119, 2013.
- [17] Mikel Artetxe, Gorka Labaka, Eneko Agirre, A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings, In ACL, 2018
- [18] YN Dauphin et.al. Language Modeling with Gated Convolutional Networks. arXiv preprint arXiv:1612.08083.2016.
- [19] アンデルセン, 森鷗外. 即興詩人. 筑摩書房. 1892.
- [20] アンデルセン, 神西清. 即興詩人. 角川文庫. 1960.
- [21] 芥川竜之介. 浅草公園. ちくま文庫. 1927
- [22] DP Kingma, Jimmy Lei Ba. Adam: A Method for Stochastic Optimization. ICLR.2015
- [23] John Duchi, Elad Hazan, Yoram Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. MLR12.2011.2121-2159p.
- [24] Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 4:2, 2012
- [25] L'eon Bottou. Online learning and stochastic approximations. Online learning in neural networks, Vol. 17, No. 9, p. 142, 1998
- [26] Matthew D Zeiler. Adadelta: an adaptive learning rate method. arXiv preprint arXiv:1212.5701,2012.