日英ニューラル機械翻訳におけるアテンションを用いた 未知語置き換えの手法

伊部 早紀[†]・松田 源立[†]・山口 和紀[†]

ニューラル機械翻訳では、従来の統計的機械翻訳に比べ文法的に流暢な文が生成されるが、出力結果に未知語が含まれることがしばしば指摘される。この問題に対処する方法としては、学習コーパス中の低頻度語を分割したり、未知語に位置情報を付け加えるなどの方法があるが、どれも日英翻訳では効果が低い。そこで本論文では、アテンションから構成した単語アライメント表を用いて出力文中の未知語と対応する入力文中の単語を見つけ、その単語を翻訳した単語で未知語を置き換えることで未知語をなくす手法を提案する。本論文の有効性を示すために ASPEC, NTCIR-10 の 2 種類のコーパスを用いて実験を行った結果、本論文で提案する単語アライメント表の構成法を用いると、未知語を全く発生させず、かつ、BLEU 値を向上させることができた。

キーワード：機械翻訳, ニューラルネットワーク, 未知語, アテンション

Replacement of Unknown Words Using an Attention Model in Japanese to English Neural Machine Translation

SAKI IBE[†], YOSHITATSU MATSUDA[†] and KAZUNORI YAMAGUCHI[†]

It is well known that machine translation using recurrent neural networks often composes fluent sentences but may include many unknown words. Although there have been many works to address the unknown word problem, they are ineffective in Japanese to English translation. In this study, we propose a hybrid method that makes an alignment table using an attention weight matrix, detects input words that are aligned with each unknown words, and finally replaces those unknown words with the translated words using a statistical machine translation method. We evaluate our approach by using two corpora: ASPEC and NTCIR-10. The results showed that the proposed method generated no unknown words and improved the BLEU (BiLingual Evaluation Understudy) score.

Key Words: *Machine Translation, Neural Networks, Unknown Words, Attention Mechanism*

[†] 東京大学, The University of Tokyo

1 はじめに

ニューラル機械翻訳 (Bahdanau, Cho, and Bengio 2014; Sutskever, Vinyals, and Le 2014; Cho, van Merriënboer, Gulcehre, Bahdanau, Bougares, Schwenk, and Bengio 2014) は、ソース言語を数値ベクトルによる分散表現で表し、それをニューラルネットワークを用いて変換して求めた数値ベクトルからターゲット言語の単語列を求めることで翻訳を行う手法である。従来の統計的機械翻訳 (Koehn, Och, and Marcu 2003) では対訳コーパスから求めた変換規則の確率を用いてソース言語の単語やフレーズをターゲット言語への単語やフレーズに変換していたため、フレーズ同士の長い区間でのつながりが十分に反映されていなかった。これに対して、ニューラル機械翻訳では、リカレントニューラルネットワークおよび LSTM (Long Short Term Memory) の利用により、長い区間での単語のつながりが考慮されている。そのため、ニューラル機械翻訳を用いると従来の統計的機械翻訳と比べて流暢な文を生成できるが、訳抜けや繰り返しがあることや、出力結果に未知語 (UNK) が含まれる (Luong, Sutskever, Le, Vinyals, and Zaremba 2015; Jean, Cho, Memisevic, and Bengio 2015) という問題が指摘されている。未知語が含まれる問題に対処するこれまでに提案されている主な手法には以下のものがある。

まず、コーパスに前処理を行う方法としては、コーパス中の未知語をすべて未知語トークンに置き換え、位置情報を付け加えて学習を行う PosUNK (Luong et al. 2015) がある。通常の手法では、語彙制限のために未知語が生じた場合は、ソース言語およびターゲット言語内の未知語を一律に特殊な未知語トークン UNK で置き換える。一方 PosUNK においては、ソース言語内の未知語はすべて UNK に置き換えるのは同じだが、ターゲット言語の未知語は位置情報を利用して区別する。具体的には、ソース言語を $f_1, \dots, f_n$ とし、ターゲット言語を $e_1, \dots, e_m$ として、ソース言語に未知語 f_i が存在したとする。 f_i に対応する未知語 e_j がターゲット言語内に存在した場合は、相対位置 $d = j - i$ を利用して、 e_j を位置情報付き未知語トークン PosUNK_d に置き換える。ただし、ソース言語に対応する未知語を持たないターゲット言語の未知語は、空集合 ϕ に対応する PosUNK_ϕ に置き換える。これにより学習した翻訳機は、未知語を一律に UNK としてではなく、 PosUNK_d として相対位置付きで推定するため、 d を利用して対応するソース言語内の単語を推定することができる。しかしながら、この手法を日英の言語対で実験を行ったところ、効果が低かった (4.3 節参照)。この手法は、ターゲット言語とソース言語において、単語間の相対位置は同一であると仮定しているため、日英のような文法構造が大幅に異なる言語間では適用が困難であるためと考えられる。

また、コーパス中の単語を分割して全体の単語の種類を少なくする BPE (Byte Pair Encoding) (Sennrich, Haddow, and Birch 2016) がある。BPE は頻度の低い単語を複数の文字列に分割することで、単語の頻度を増やし学習しやすくする方法である。この手法も日英の言語対で実験を行ったところ、効果が低かった (4.3 節参照)。ヨーロッパ言語などの文字の種類が少なく単

語の成り立ちが類似する言語間に比べ、日本語と英語では単語の構成が大きく異なり、日本語は文字の種類が多いことが原因だと考えられる。ニューラル機械翻訳のモデルを変更する方法としては、アテンションの計算で単語翻訳確率を考慮する方法 (Arthur, Neubig, and Nakamura 2016), coverage を導入する方法 (Tu, Lu, Liu, Liu, and Li 2016; Tu, Liu, Lu, Liu, and Li 2017), 統計的機械翻訳で作成したフレーズテーブルを組み入れる方法 (Stahlberg, Hasler, Waite, and Byrne 2016; Khayrallah, Kumar, Duh, Post, and Koehn 2017; Zhang, Utiyama, Sumita, Neubig, and Nakamura 2018), 入力文中の単語が既知語の場合はそのまま処理し、未知語の場合は単語を文字に分解して処理する方法 (Luong and Manning 2016) などがあるが、いずれもニューラルネットワークの性能を改善させることが主眼であり、未知語そのものを完全に消去することを目的としていない。ニューラル機械翻訳の出力結果の単語列を統計的機械翻訳を用いて並べ替える手法 (Skadina and Rozis 2016) もあるが、この手法では単語の辞書を作るためだけにニューラル機械翻訳を使っており、最終的な翻訳は統計的機械翻訳で行っている。よってニューラル機械翻訳の利点である単語間の長区間でのつながりを考慮した流暢性が失われてしまう。以上のように従来の手法の多くは、未知語を減少させることはできているが、日英翻訳では翻訳精度の向上が期待できない。

そこで本論文では、ニューラルネットワークのモデルや探索方法を変更することなく未知語を減少させ、かつ翻訳精度を向上させる手法を提案する。そのために、アテンションに基づいたニューラル機械翻訳をソース言語に対して適用することで生成されたアテンションを利用する。アテンションは翻訳におけるソース言語の単語とターゲット言語の単語の対応を数値化したもの (Bahdanau et al. 2014) で、統計的機械翻訳における2言語間の単語の対応を表す単語アライメント表 (Koehn et al. 2003) と類似したものである。(Hashimoto, Eriguchi, and Tsuruoka 2016) 及び (Freitag and Al-Onaizan 2016) では、この性質を利用し、アテンションを元に対応する未知語を推定する手法を提案している。ともに未知語が対応する単語はアテンションが最も高い値を持つ単語だとしている。しかしながら、類似しているとはいえ、アテンションは単語アライメント表そのものではなく、数値の大小と実際のアライメントが一致しない場合も多い。また、言語学的な性質を満たしているとも限らない。本研究では、隣接関係等の、単語間の対応関係の言語学的性質に関するヒューリスティックを利用して、アテンションから単語アライメント表を推定する手法を提案する。そして、その単語アライメント表を利用して未知語を置き換える。これによりニューラル翻訳の利点を生かしたアテンションと言語学的な性質の双方を組み合わせた未知語問題の解決を行う。本論文はアテンションを用いた未知語解決に関する論文 (伊部, 松田, 山口 2018) の内容を発展させたものである。ASPEC と NTCIR-10 の2つのコーパスに提案手法を適用したところ、未知語を完全に除くことができ、BLEU 値も上昇させることができた。

2 背景

本章では、ニューラル機械翻訳の手法と統計的機械翻訳の手法の中から本論文で使用する手法の説明を行う。

2.1 フレーズに基づく統計的機械翻訳

本論文では、未知語を置き換える単語を探す時にフレーズベース機械翻訳 (Koehn et al. 2003) で生成されるフレーズテーブルを使うため、フレーズベース機械翻訳について説明する。フレーズベース機械翻訳は、統計的手法を用いて翻訳規則を学習し、翻訳を行う手法である。手順を以下に示す。

- (1) 対訳コーパスにアライメントモデルを適用し、単語アライメント表を作成する。単語アライメント表の計算では同時に単語翻訳確率 $t(e|f)$ も計算される。単語アライメント表をソース言語からターゲット言語、またその逆に対して作成し、grow-diag-final-and (gdfa) などのヒューリスティックを用いて2つの表を重ね合わせたものを最終的な単語アライメント表とする。
- (2) 作成した単語アライメント表とソース言語・ターゲット言語をもとに、ソース言語における単語列がターゲット言語におけるどの単語列に翻訳されるかを表す翻訳規則と、フレーズ翻訳確率 $P(e|f)$ を学習する。規則の抽出は (Och and Ney 2004) などの方法がある。
- (3) 翻訳規則を利用して出力候補文をいくつか生成する。各候補文に対し翻訳規則の確率と言語モデルを用いて翻訳確率を計算し、翻訳確率が最も高い候補文を出力結果とする。

2.2 アテンションに基づくニューラル機械翻訳

アテンションに基づくニューラル機械翻訳 (Bahdanau et al. 2014) は、アテンションを用いて翻訳するソース言語の各単語に異なる重みを与えるニューラル機械翻訳である。

入力文 $\mathbf{f} = (f_1, f_2, \dots, f_J)$ とその分散表現 $\mathbf{x} = (x_1, x_2, \dots, x_J)$ 、出力文 $\mathbf{e} = (e_1, e_2, \dots, e_I)$ とその分散表現 $\mathbf{y} = (y_1, y_2, \dots, y_I)$ のペアの条件付き確率を最大化するように学習する。

$$p(\mathbf{e}|\mathbf{f}) = \prod_{i=1}^I p(e_i|e_1, \dots, e_{i-1}, \mathbf{f}) \quad (1)$$

e_i の生成確率は以下で与えられる。

$$p(e_i|e_1, \dots, e_{i-1}, \mathbf{x}) = g(y_{i-1}, s_i, c_i) \quad (2)$$

s_i は i 番目の出力単語列の隠れ状態であり、以下の式 (3) で計算される。

$$s_i = f(s_{i-1}, y_{i-1}, c_i) \quad (3)$$

c_i は出力文の i 番目の単語の文脈ベクトルといい, 以下の式 (4) で計算される.

$$c_i = \sum_{j=1}^J \alpha_{ij} h_j \quad (4)$$

α_{ij} はアテンション値といい, y_i に関連しているのが x_j である確率である. 式 (5) と式 (6) に従って計算する.

$$\alpha_{ij} = \frac{\exp e_{ij}}{\sum_{k=1}^J \exp e_{ik}} \quad (5)$$

$$e_{ij} = a(s_{i-1}, h_j) \quad (6)$$

h_j は入力文の j 番目の単語の文脈ベクトルで, 式 (7) に従って計算する¹.

$$h_j = h(x_j, h_{j-1}) \quad (7)$$

以上の $a(), f(), g(), h()$ は入力変数の重み付き線形和を何らかの非線形関数 (通常は $\tanh$ 関数) で変換したものである. 具体的には, 入力変数群を $v_1, \dots, v_n$ とすると, $a(v_1, \dots, v_n) = \tanh(\sum_i w_i v_i + c)$ と表される. w_i は各変数の重みであり, c は切片である. 全ての w_i および c がパラメータとして, 損失関数を最小化するために最適化される. 学習では, 以下の式 (8) で定義されるクロスエントロピーを損失関数として用いる.

$$L = -\sum_j z_j \log z'_j \quad (8)$$

z_j はコーパスにおける y_j に対応する単語 e_j の出現確率, z'_j は学習したパラメータを用いて翻訳を行った時に y_j に対応する単語 e_j が出力される確率である. 学習後の翻訳では式 (1) を最大化する $\mathbf{e}$ をビームサーチなどで求める.

3 手法

本章ではニューラル機械翻訳を用いてモデル学習し翻訳を行った後, アテンションから単語アライメント表を作成し, 作成した単語アライメント表を用いて未知語を適切な単語に置き換える手法を説明する. 3.1 節では, 単語アライメント表の作成の方法として, 従来手法である intersection と gdfa を説明し, 提案手法である gdfa-f (gdfa fill) を導入する. 3.2 節では, 生成した単語アライメント表にしたがって未知語に対応する単語を決定する方法を説明する.

¹ 双方向 RNN の場合は入力文を逆向きに計算した h'_j を計算し, h_j と h'_j を連結したベクトルを使うが, ここでは説明は省略する.

3.1 単語アライメント表の作成

単語アライメント表の作成アルゴリズムの **intersection** と **gdfa** は (Koehn et al. 2003) で提案されたもので, **gdfa-f** は **gdfa** を改良した提案手法である. ある i, j に対する a_{ij} や単語アライメント表の要素をセルと呼ぶ.

intersection ソース言語とターゲット言語どちらから見てもアテンション値が最も高いセルを対応付ける. 各セルの値 b_{ij} は以下の式に従って計算する.

$$b_{ij} = \begin{cases} 1 & \text{if } i = \arg \max_{i'} a_{i'j} \text{ and } j = \arg \max_{j'} a_{ij'} \\ 0 & \text{otherwise} \end{cases}$$

gdfa **gdfa** では, **intersection** により作成した $b_{ij} = 1$ となるセルに隣接するセル群をまず候補として抽出する. 最初は全ての候補セルの値は 0 である. ある候補セルのアテンション値が, 他のソース単語に対応するアテンション値と比べて大きい場合, そのセルの値を 1 とする. 言い換えれば, そのセルのアテンション値が, その列内で最大値をとる場合, セルの値を 1 とする. また, ある候補セル内のアテンション値が, 他のターゲット単語に対応するアテンション値と比べて大きい場合 (すなわち, 行内で最大値をとる場合) も, そのセルの値を 1 とする (厳密な定義は以下の式を参照のこと). これはソース言語の 1 つの単語がターゲット言語において複数の単語に対応している場合, それらは隣り合うことが多いので, すでに単語アライメント表に入っている単語と隣り合う単語のみを表に追加することを許すという考え方による. **gdfa** による単語アライメント表 b'_{ij} は, $b_{p,q}$ の (上下左右の) 隣接セルの中で 1 をとるセルの個数を返す関数

$$neighbor(b_{pq}) = b_{(p-1)q} + b_{(p+1)q} + b_{p(q-1)} + b_{p(q+1)}$$

を使って以下のように計算する.

$$b'_{ij} = \begin{cases} 1 & \text{if } (b_{ij} = 1) \text{ or } \left(neighbor(b_{ij}) \geq 1 \text{ and } \left(i = \arg \max_{i'} a_{i'j} \text{ or } j = \arg \max_{j'} a_{ij'} \right) \right) \\ 0 & \text{otherwise} \end{cases}$$

gdfa-f **gdfa** において作成した行列 b'_{ij} をもとに, ソース言語のそれぞれの単語に対して, 対応付けされているターゲット言語の単語がなければ, アテンション値が最も高いセルに対応付ける. ターゲット言語のそれぞれの単語に対しても同様の操作を行う. **gdfa-f** による単語アライメント表 b''_{ij} は以下の式に従い計算する.

$$b''_{ij} = \begin{cases} 1 & \text{if } b'_{ij} = 1 \text{ or } (j = \arg \max_{j' \in \mathcal{J}} a_{ij'}) \text{ and } i \in \mathcal{I} \text{ or } (i = \arg \max_{i' \in \mathcal{I}} a_{i'j} \text{ and } j \in \mathcal{J}) \\ 0 & \text{otherwise} \end{cases}$$

図 1 intersection, gdfa, gdfa-f の例

ここで $\mathcal{I}$, $\mathcal{J}$ はそれぞれ以下の式で定義される.

$$\mathcal{I} = \{i \mid \sum_j b'_{ij} = 0\} \text{ and } \mathcal{J} = \{j \mid \sum_i b'_{ij} = 0\}$$

アテンションから intersection, gdfa, gdfa-f を求めた例を図 1 に示す. 図中の各セルはそれぞれ $b_{ij}, b'_{ij}, b''_{ij}$ の値を表し, 白は 0, 灰色は 1 とした. gdfa-f では, 出力文中の各単語に対して, 少なくとも一つの入力文中の単語が必ず対応する. 言い換えれば, gdfa-f により, 出力文中の全ての未知語に対して, 既知の入力文中の単語を必ず割り当てることができる.

3.2 未知語の置き換え

生成した単語アライメント表を用いて, ニューラル機械翻訳の出力文中の未知語 e_i を単語に置き換える方法について説明する. まず, e_i に対応している入力文中の単語列 $\mathbf{f}_i$ を $\mathbf{f}_i = \{f_j \mid b_{ij} = 1\}$ とする. 次に, $\mathbf{f}_i$ の翻訳となる単語列を決定して未知語 e_i と置き換える. 本論文では $\mathbf{f}_i$ の翻訳となる単語列の決定方法として IBM と ChangePhrase (Koehn et al. 2003) と Dict を用いた.

IBM 単語アライメント表の主な作成法は (Hashimoto et al. 2016; Arthur et al. 2016) などを用いられている IBM モデルが挙げられる. ここでは IBM モデル 4 を用いた. 対訳コーパスに IBM モデル 4 を適用したときの単語翻訳確率 $t(e|f)$ を使い, $\mathbf{f}_i = (f_{i_1}, \dots, f_{i_n})$ の要素それぞれにおいて最も確率の高い $e_{best} = \arg \max_e t(e|\mathbf{f}_i)$ を選択する手法である.

ChangePhrase f_i の翻訳を統計的機械翻訳で作成したフレーズテーブルから参照し、コーパスから計算したフレーズ翻訳確率 $P(e|f) = \frac{c(e,f)}{c(f)}$ が最も高いフレーズ $e_{best} = \arg \max_e P(e|f_i)$ を選択する手法である。 $c(f)$ はコーパス中のフレーズ f の出現回数、 $c(e, f)$ はフレーズ e と f の同時出現回数である。フレーズで置き換えるため、未知語を複数単語と置き換えることもある。

Dict 外部の辞書から f_j の各単語の訳を検索した単語を選択する手法である。

4 実験

4.1 実験データ・方法

対訳コーパスとしては ASPEC(Nakazawa, Yaguchi, Uchimoto, Utiyama, Sumita, Kurohashi, and Isahara 2016), NTCIR-10 の patentMT(Goto, Chow, Lu, Sumita, and Tsou 2013)²を用いた。コーパスのサイズと単語数を表1に示す。モデル学習およびデコーダには nematus³を用い、日英および英日の翻訳を行った。隠れ層の数は1,000, 単語ベクトルの次元数は512, RNNはGRU, 学習アルゴリズムはAdam, 学習率は0.0001, バッチサイズは40, dropoutは未使用で学習を行った。英語の構文解析にはStanford Parser⁴を、日本語のトークン化にはKyTea⁵を用いた。IBMモデル4の実装としてはGIZA++⁶を用いた。フレーズテーブルの抽出はmosesdecoder⁷を用いた。未知語の置き換えの手法のDictでは外部辞書としてEDict(Breen 2000)を用いた。コーパスの前処理としては、(Neubig 2013)を参考に、単語数51以上の文は学習データから取り除いた。

ASPECで語彙数を1万から5万まで1万ずつ変化させた時の翻訳結果のBLEU値を計算し

表1 各コーパスのサイズと全語彙数

		train		dev	test
		文	語彙	文	文
ASPEC	日本語	908.1 K	162.3 K	1.8 K	1.8 K
	英語		326.8 K		
NTCIR-10	日本語	3.2 M	146.5 K	2.0 K	0.9 K
	英語		265.0 K		

train は学習に、dev はパラメータチューニングに、test はテストに用いた。

² 日英特許機械翻訳タスクデータ中の NTCIR-7 および NTCIR-8 を合わせたものを用いた。

³ <https://github.com/EdinburghNLP/nematus>

⁴ <https://stanfordnlp.github.io/CoreNLP/>

⁵ <http://www.phontron.com/kytea/index-ja.html>

⁶ <https://github.com/moses-smt/giza-pp>

⁷ <https://github.com/moses-smt/mosesdecoder>

表 2 ASPEC において語彙数を変化させたときの日英の翻訳結果の BLEU 値

語彙数	1 万	2 万	3 万	4 万	5 万
BLEU	22.97	24.06	24.61	24.97	24.74

太字は最も高い値.

表 3 語彙数を制限した時の未知語の割合

	日本語 (%)	英語 (%)
ASPEC	0.85	2.48
NTCIR-10	0.21	0.68

た結果を表 2 に示す. この結果から, 最も BLEU 値が高かった語彙数 4 万を実験で用いた. パラメータを統一するために NTCIR-10 でも同じパラメータで実験を行った. 語彙数を 4 万に制限した時の (延べ) 全単語数における未知語の割合を表 3 に示す.

次にニューラル機械翻訳のコーパスからの学習における適切な更新回数を調べるために, 更新回数 1 万から 15 万まで 1 万ごとに損失関数の値を計算した. その結果 ASPEC は 10 万回, NTCIR-10 は 12 万回以降, 損失関数の値が変化しなくなったため, これらを更新回数として設定した.

4.2 評価指標

翻訳精度の評価指標には BLEU (Papineni, Roukos, Ward, and Zhu 2002), METEOR (Banerjee and Lavie 2005) の 2 つを用いた. また METEOR は日本語の評価には対応していないため日英翻訳の評価のみに用いた.

4.3 実験結果と考察

コーパスとして ASPEC を用いた翻訳の結果を表 4 に示す. baseline はアテンションベースのニューラル機械翻訳のシステムである nematus のデフォルトの設定で学習したモデルである. BPE は (Sennrich et al. 2016), PosUNK は (Luong et al. 2015) を再実装して実験した結果である. 語彙数はそれぞれ 4 万とした. 語彙数を変更した実験も行ったが, BLEU 値や未知語の割合に大きな変化はなかったため, ここでは提案手法と同様の 4 万とした. 学習の更新回数については, 提案手法同様に損失関数の値が最も小さくなった時の BLEU 値を採用した. intersection+ChangePhrase 以下が提案手法を実装して実験した結果である. これは baseline である nematus で求めたアテンションから intersection, gdfa, gdfa-f のいずれかを用いて単語アライメント表を作成し, ChangePhrase, IBM, Dict のいずれかを用いて未知語に置き換える単語列を決めたものである. 単語アライメント表の作成方法と未知語に置き換える単語列の決め

方の 3×3 通りの全ての組合せを実行した.

表 4 より, 未知語を置き換える手法としては ChangePhrase より IBM の方が良い結果が出ている. ChangePhrase では未知語が複数連続している場合にそれらをまとめてフレーズとするため, そのフレーズの訳が見つからなければ置き換えることができないが, IBM では 1 単語ずつ未知語を置き換えているので, 学習コーパス中にある単語であれば必ず置き換えることができるので, それが原因として考えられる.

単語アライメントを求める方法としては, gdfa-f と IBM を組み合わせるとすべての未知語を別の単語で置き換えることができています. また BLEU 値もよく使われる手法である intersection と比べて大差がなく, 同等の結果となっている. 一方, 既存手法である BPE と PosUNK では, 未知語の数を大幅に減らすことはできているが, 翻訳性能そのものはむしろ悪化している. コーパスとして NTCIR-10 を用いて同じ実験を行った結果を表 5 に示す. NTCIR-10 では語彙数を 4 万に制限した場合に未知語の数自体が少なく BLEU 値などに差が少ないが, ASPEC と同様に gdfa-f+IBM を使うとすべての未知語を別の単語で置き換えることができています. また BLEU 値も intersection と比べて大差がなく, 同等の結果となっている.

表 4 を見ると METEOR では gdfa-f+IBM が最も高い数値を出している. METEOR は BLEU と違い同義語に高いスコアを与える翻訳の評価指標であるため, これは gdfa-f を適用することで意味の近い単語を選択できているということを示唆する.

次に, gdfa-f+IBM において, パラメータ更新回数と BLEU 値および UNK の出現率の関係を

表 4 ASPEC コーパスでの翻訳結果の精度比較結果

ASPEC	日→英			英→日	
	BLEU (%)	METEOR (%)	UNK (%)	BLEU (%)	UNK (%)
baseline	24.97	31.50	2.72	34.58	2.07
BPE	21.47	28.34	0.00	29.54	0.00
PosUNK	24.25	31.63	0.07	34.47	0.17
+intersection+ChangePhrase	25.34	31.97	1.10	34.64	0.57
+intersection+IBM	25.56	32.06	0.42	34.63	0.50
+intersection+Dict	25.05	31.71	2.00	34.58	0.81
+gdfa+ChangePhrase	25.37	32.03	1.00	34.67	0.36
+gdfa+IBM	25.50	32.14	0.14	34.65	0.23
+gdfa+Dict	25.04	31.72	1.96	34.58	0.99
+gdfa-f+Changephrase	25.37	32.06	0.89	34.68	0.16
+gdfa-f+IBM	25.51	32.16	0.00	34.66	0.00
+gdfa-f+Dict	25.05	31.73	1.91	34.59	1.10

太字は最も高い (UNK の場合は最も低い) 値.

グラフに表したものが図2および図3である. ここからパラメータ更新回数が少ない場合でも gdfa-f+IBM を適用することで未知語をなくし, BLEU 値も若干あげることができることがわかる.

表5 NTCIR-10 コーパスでの翻訳結果の精度比較結果

NTCIR-10	日→英			英→日	
	BLEU (%)	METEOR (%)	UNK (%)	BLEU (%)	UNK (%)
baseline	35.31	33.88	1.59	36.73	0.37
BPE	32.95	31.12	0.00	33.26	0.00
PosUNK	34.88	33.52	0.09	36.39	0.03
+intersection+ChangePhrase	35.33	33.92	1.12	36.76	0.20
+intersection+IBM	35.37	33.99	0.21	36.83	0.05
+intersection+Dict	35.33	33.91	1.16	36.74	0.28
+gdfa+ChangePhrase	35.31	31.88	1.08	36.77	0.18
+gdfa+IBM	35.31	34.02	0.10	36.83	0.03
+gdfa+Dict	35.32	33.91	1.20	36.74	0.27
+gdfa-f+changePhrase	35.33	33.90	1.01	36.77	0.21
+gdfa-f+IBM	35.31	34.03	0.00	36.83	0.00
+gdfa-f+Dict	35.32	33.91	1.17	36.73	0.26

太字は最も高い (UNK の場合は最も低い) 値.

図2 パラメータ更新回数と BLEU 値の関係 (日英)

図 3 パラメータ更新回数と BLEU 値の関係 (英日)

5 おわりに

本論文では、ニューラル機械翻訳により生成されたアテンションをもとに、単語の重複がないように単語アライメント表を作成することで、出力結果における未知語が入力文のどの単語に対応しているかを判別し、統計的機械翻訳でのモデルを用いて未知語を適切な単語に置き換える手法を提案した。その結果、提案手法の gdffa-f を用いて単語アライメント表を作ることによって未知語をなくすことができ、BLEU 値なども向上させることができることがわかった。今後はより大きな外部辞書を活用するなどしてさらに翻訳精度を高めていきたい。また、本論文で提案した手法はターゲット言語における 1 つの未知語が単語もしくはフレーズに対応しているという仮定に基づいているが、未知語とその周辺の単語またはフレーズが一つの単語に対応している、すなわち置き換えた単語と周辺の単語が重複する状況も考えられる。そのような場合も考慮した手法を検討していきたい。

参考文献

- Arthur, P., Neubig, G., and Nakamura, S. (2016). “Incorporating Discrete Translation Lexicons into Neural Machine Translation.” In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1557–1567.
- Bahdanau, D., Cho, K., and Bengio, Y. (2014). “Neural Machine Translation by Jointly Learning to Align and Translate.” *arXiv preprint arXiv:1409.0473*.
- Banerjee, S. and Lavie, A. (2005). “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments.” In *Proceedings of ACL Workshop on In-*

- trinsic and Extrinsic Evaluation Measures for Machine Translation And/or Summarization*, Vol. 29, pp. 65–72.
- Breen, J. (2000). “A www Japanese Dictionary.” *Japanese Studies*, **20** (3), pp. 313–317.
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation.” In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, Doha, Qatar. Association for Computational Linguistics.
- Freitag, M. and Al-Onaizan, Y. (2016). “Fast Domain Adaptation for Neural Machine Translation.” *arXiv preprint arXiv:1612.06897*.
- Goto, I., Chow, K.-P., Lu, B., Sumita, E., and Tsou, B. K. (2013). “Overview of the Patent Machine Translation Task at the NTCIR-10 Workshop.” In *NTCIR*.
- Hashimoto, K., Eriguchi, A., and Tsuruoka, Y. (2016). “Domain Adaptation and Attention-based Unknown Word Replacement in Chinese-to-Japanese Neural Machine Translation.” In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pp. 75–83.
- 伊部早紀, 松田源立, 山口和紀 (2018). 日英ニューラル機械翻訳における未知語への対応. 言語処理学会第 24 年次大会発表論文集, pp. 1046–1049.
- Jean, S., Cho, K., Memisevic, R., and Bengio, Y. (2015). “On Using Very Large Target Vocabulary for Neural Machine Translation.” In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Vol. 1, pp. 1–10.
- Khayrallah, H., Kumar, G., Duh, K., Post, M., and Koehn, P. (2017). “Neural Lattice Search for Domain Adaptation in Machine Translation.” In *Proceedings of the 8th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Vol. 2, pp. 20–25.
- Koehn, P., Och, F. J., and Marcu, D. (2003). “Statistical Phrase-based Translation.” In *NAACL HLT-Volume 1*, pp. 48–54. ACL.
- Luong, M.-T. and Manning, C. D. (2016). “Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models.” In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1, pp. 1054–1063.
- Luong, T., Sutskever, I., Le, Q., Vinyals, O., and Zaremba, W. (2015). “Addressing the Rare Word Problem in Neural Machine Translation.” In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Vol. 1, pp. 11–19.
- Nakazawa, T., Yaguchi, M., Uchimoto, K., Utiyama, M., Sumita, E., Kurohashi, S., and Isahara,

- H. (2016). “ASPEC: Asian Scientific Paper Excerpt Corpus.” In Calzolari, N., Choukri, K., Declerck, T., Grobelnik, M., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., and Piperidis, S. (Eds.), *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2016)*, pp. 2204–2208, Portorož, Slovenia. ELRA.
- Neubig, G. (2013). “Travatar: A Forest-to-String Machine Translation Engine based on Tree Transducers.” In *ACL System Demonstrations*, pp. 91–96.
- Och, F. J. and Ney, H. (2004). “The Alignment Template Approach to Statistical Machine Translation.” *Computational Linguistics*, **30** (4), pp. 417–449.
- Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). “BLEU: A Method for Automatic Evaluation of Machine Translation.” In *ACL*, pp. 311–318. ACL.
- Sennrich, R., Haddow, B., and Birch, A. (2016). “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)*, Vol. 1, pp. 1715–1725.
- Skadina, I. and Rozis, R. (2016). “Towards Hybrid Neural Machine Translation for English-Latvian.” In *Human Language Technologies–The Baltic Perspective: Proceedings of the Seventh International Conference Baltic HLT 2016*, Vol. 289, p. 84. IOS Press.
- Stahlberg, F., Hasler, E., Waite, A., and Byrne, B. (2016). “Syntactically Guided Neural Machine Translation.” In *The 54th Annual Meeting of the Association for Computational Linguistics*, p. 299.
- Sutskever, I., Vinyals, O., and Le, Q. V. (2014). “Sequence to Sequence Learning with Neural Networks.” In *Advances in Neural Information Processing Systems*, pp. 3104–3112.
- Tu, Z., Liu, Y., Lu, Z., Liu, X., and Li, H. (2017). “Context Gates for Neural Machine Translation.” *Transactions of the Association for Computational Linguistics*, **5**, pp. 87–99.
- Tu, Z., Lu, Z., Liu, Y., Liu, X., and Li, H. (2016). “Modeling Coverage for Neural Machine Translation.” In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1, pp. 76–85.
- Zhang, J., Utiyama, M., Sumita, E., Neubig, G., and Nakamura, S. (2018). “Guiding Neural Machine Translation with Retrieved Translation Pieces.” In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, Vol. 1, pp. 1325–1335.

略歴

伊部 早紀：2017年東京大学教養学部学際科学科卒業。東京大学大学院総合文化研究科広域システム科学専攻修士課程在学中。言語処理に関する研究開発

に従事.

松田 源立: 1997 年東京大学教養学部卒業. 2002 年東京大学大学院総合文化研究科博士課程修了. 博士 (学術). 学振特別研究員 (PD), 青山学院大学理工学部助教を経て, 2013 年より東京大学大学院総合文化研究科学術研究員.

山口 和紀: 1979 年東京大学理学部数学科卒業. 1985 年理学博士 (東京大学). 1992 年東京大学教養学部助教授. 2007 年より東京大学大学院総合文化研究科教授. コンピュータのためのモデリング全般に興味を持つ.

(2018 年 5 月 1 日 受付)

(2018 年 7 月 30 日 再受付)

(2018 年 9 月 6 日 採録)