

# 係り受け構造に基づく Attention の制約を用いた Transformer ニューラル機械翻訳

出口 祥之<sup>†</sup>・田村 晃裕<sup>†,††</sup>・二宮 崇<sup>†</sup>

本論文では、ニューラル機械翻訳 (NMT) の性能を改善するため、原言語側と目的言語側両方の係り受け関係を捉える dependency-based self-attention を組み込んだ新しい Transformer NMT モデルを提案する。Dependency-based self-attention は linguistically-informed self-attention (LISA) から着想を得ており、各単語のアテンションが係り先の単語となるように、Transformer の self-attention を係り受け関係に基づいた制約を与えて学習させる。LISA は、意味役割付与の性能を改善するため、Transformer エンコーダ向けに設計されているが、本論文では LISA を Transformer NMT に拡張する。具体的には、デコーダ側の dependency-based self-attention を訓練する際に後方の単語に係る係り受け関係をマスクすることで、訓練時と推論時との不整合を解消する。さらに、我々の提案する dependency-based self-attention は、byte pair encoding のようなサブワード単位での演算が可能である。Asian Scientific Paper Excerpt Corpus (ASPEC) の日英・英日翻訳タスクの評価実験により、Dependency-based self-attention を組み込んだ提案モデルは、従来の Transformer NMT モデルと比較して、それぞれ 1.04 ポイント・0.30 ポイント BLEU スコアが上昇することを示す。

キーワード：ニューラル機械翻訳、係り受け構造、Transformer, Self-Attention

## Dependency-Based Self-Attention for Transformer Neural Machine Translation

HIROYUKI DEGUCHI<sup>†</sup>, AKIHIRO TAMURA<sup>†,††</sup> and TAKASHI NINOMIYA<sup>†</sup>

This paper proposes a new Transformer neural machine translation (NMT) model that incorporates dependency relations into self-attention on both the source and target sides, “dependency-based self-attention”. The dependency-based self-attention is trained to attend to the modifier for each token under constraints based on dependency relations. This was inspired by linguistically-informed self-attention (LISA). LISA was originally designed for the Transformer encoder for semantic role labeling. However, this paper extends LISA to the Transformer NMT by masking future information on words in the decoder-side dependency-based self-attention. Further, our dependency-based self-attention operates with sub-word units created by byte pair

---

<sup>†</sup> 愛媛大学大学院理工学研究科, Graduate School of Science and Engineering, Ehime University

<sup>††</sup> 現在, 同志社大学, Presently with Doshisha University

encoding. In the experiments, our model achieved an increase of 1.04 and 0.30 points in the BLEU over the baseline model, respectively, on the Asian Scientific Paper Excerpt Corpus Japanese-to-English and English-to-Japanese translation tasks.

**Key Words:** *Neural Machine Translation, Dependency Structure, Transformer, Self-Attention*

## 1 はじめに

近年、自然言語処理の多くのタスクにおいて、ニューラルネットワークが活用されている。機械翻訳の分野においてもその有効性が示されており、その中でも、Transformer (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, and Polosukhin 2017) というモデルがリカレントニューラルネットワーク (RNN) を用いたモデルや畳み込みニューラルネットワーク (CNN) を用いたモデルの翻訳性能を上回り、注目を浴びている。これまでに、統計的機械翻訳やニューラル機械翻訳では原言語や目的言語の文構造を考慮することで翻訳性能が改善されており (Lin 2004; Chen, Wang, Utiyama, Liu, Tamura, Sumita, and Zhao 2017; Eriguchi, Tsuruoka, and Cho 2017), Transformer NMT においても文構造の有用性が示されている (Wu, Zhang, Zhang, Yang, Li, and Zhou 2018). そこで本研究では、Transformer モデルで係り受け構造を考慮することで翻訳性能の改善を試みる。

Transformer の特徴の一つである self-attention は文内における単語間の関連の強さを考慮することができ、機械翻訳のみならず、言語モデルの獲得や意味役割付与など、様々なタスクにおいて精度の向上に寄与してきた。Strubell らは意味役割付与の性能を向上させるため、Transformer エンコーダの self-attention で文の係り受け構造を捉える、linguistically-informed self-attention (LISA) と呼ばれるモデルを提案している (Strubell, Verga, Andor, Weiss, and McCallum 2018). LISA では、multi-head self-attention のうちの 1 つのヘッドを、各単語が係り先の単語を指すように、係り受け関係に基づいた制約を与えて学習させている。

本研究は、Transformer エンコーダとデコーダの self-attention で、それぞれ、原言語の文と目的言語の文の係り受け構造を捉える Transformer NMT モデルを提案する。以降、この係り受け関係を捉える self-attention を dependency-based self-attention と呼ぶ。具体的には、NMT モデルの訓練時に、エンコーダとデコーダの self-attention の一部を、各単語が係り先の単語を指すように、原言語の文や目的言語の文の係り受け関係に基づいた制約を与えて学習させる。そして、推論時には制約を与えて学習した self-attention が文の係り受け関係を捉えながら翻訳する。ただし、推論時には目的言語文が明らかでないため、LISA の手法を直接 Transformer NMT モデルのデコーダに適用することはできない。そこで、提案の dependency-based self-attention では、まだ予測していない単語に対してアテンションを向けないように、デコーダ側の self-attention を学習する際は、自身の単語より後方に係る係り受け関係にマスクをかけた制約を用いる。

また, 近年のニューラル機械翻訳モデルの多くは, 文を単語列ではなくサブワード列として扱うことで低頻度語の翻訳に対応している (Sennrich, Haddow, and Birch 2016). そこで, 本研究では, dependency-based self-attention を byte pair encoding (BPE) などによるサブワード列に対しても適用できるように拡張する.

Asian Scientific Paper Excerpt Corpus (ASPEC) データ (Nakazawa, Yaguchi, Uchimoto, Utiyama, Sumita, Kurohashi, and Isahara 2016) を用いた日英・英日翻訳の評価実験において, 提案の Transformer モデルと従来の係り受け構造を考慮しない Transformer モデルを比較し, dependency-based self-attention を組み込むことで BLEU がそれぞれ 1.04 ポイント・0.30 ポイント向上することを確認した. また, 実験では, 原言語側の dependency-based self-attention と目的言語側の dependency-based self-attention のそれぞれの有効性と BPE に拡張したときの有効性も確認した.

本稿の構成は以下の通りである. まず 2 章で提案手法が前提とする Transformer モデルについて説明したのち, 3 章で提案手法である dependency-based self-attention を示す. 4 章で提案手法を組み込んだモデルの翻訳性能を評価することで手法の有効性を示す. 5 章で提案手法の拡張である subword dependency-based self-attention の有効性, 提案手法の dependency-based self-attention が捉える係り受け構造, 従来モデルと提案モデルの翻訳文の違い, 提案モデルの設定に関する比較実験をそれぞれ示す. 6 章で関連研究の文構造を考慮したニューラル機械翻訳モデルについて議論し, 7 章で本稿のまとめとする.

## 2 Transformer モデル

本節では, 提案モデルの基礎となる Transformer NMT モデル (Vaswani et al. 2017) を説明する. Transformer NMT モデルの概要を図 1 に示す.

Transformer NMT モデルは, 入力された原言語の単語列  $X(= x_1, x_2, \dots, x_{n_{enc}})^T$  をエンコードする Transformer エンコーダ (以下, エンコーダ) と目的言語の単語列  $Y(= y_1, y_2, \dots, y_{n_{dec}})^T$  を生成する Transformer デコーダ (以下, デコーダ) を組み合わせたエンコーダ・デコーダモデルである. エンコーダとデコーダはそれぞれ, 図 1 のようにエンコーダレイヤを  $J_{enc}$  レイヤ, デコーダレイヤを  $J_{dec}$  レイヤずつ積み重ねたものである.

Transformer は RNN のような回帰構造を持たないため, 入力文の各単語に対して並列に処理することができる. 入力単語の位置情報は位置エンコーディング (positional encoding) により考慮する. 具体的には, 単語埋め込みの次元数を  $d$  とすると, 位置エンコーディングの行列  $PE$  は

$$PE_{(pos, 2i)} = \sin(pos/10000^{2i/d}), \quad (1)$$

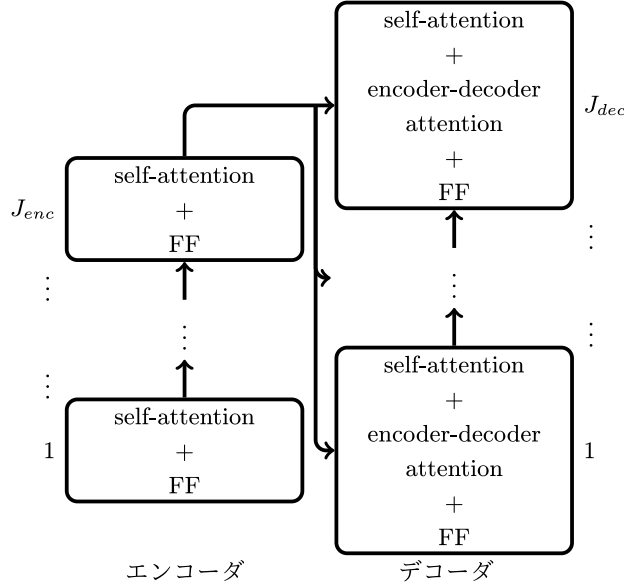


図 1 Transformer モデルの概要

$$PE_{(pos, 2i+1)} = \cos(pos/10000^{2i/d}), \quad (2)$$

と表される. なお,  $pos$  は単語の位置,  $i$  は埋め込み成分の次元である. そして, 式 (1), (2) によって計算された位置エンコーディングの行列  $PE$  を単語の埋め込み行列に加算したものを, エンコーダ及びデコーダの入力とする.

エンコーダの第  $j$  層目の出力  $S_{enc}^{(j)}$  は self-attention ( $SelfAttn()$ ) と位置毎の全結合フィードフォワードネットワーク ( $FFN()$ ) により以下の通り計算される.

$$H_{enc}^{(j)} = LN(S_{enc}^{(j-1)} + SelfAttn(S_{enc}^{(j-1)})), \quad (3)$$

$$S_{enc}^{(j)} = LN(H_{enc}^{(j)} + FFN(H_{enc}^{(j)})). \quad (4)$$

ここで,  $S_{enc}^{(0)}$  はエンコーダへの入力,  $H_{enc}^{(j)}$  は第  $j$  層目のエンコーダの self-attention の出力,  $LN()$  は Layer Normalization (Lei Ba, Kiros, and Hinton 2016) であり,  $FFN()$  は間に ReLU 関数を挟んだ 2 層の全結合層から成るネットワークである.

デコーダの第  $j$  層目の出力  $S_{dec}^{(j)}$  は self-attention と encoder-decoder attention ( $EncDecAttn()$ ), 位置毎のフィードフォワードネットワークによって以下の通り計算される.

$$H_{dec}^{(j)} = LN(S_{dec}^{(j-1)} + SelfAttn(S_{dec}^{(j-1)})), \quad (5)$$

$$G_{dec}^{(j)} = LN(H_{dec}^{(j)} + EncDecAttn(H_{dec}^{(j)}, S_{enc}^{(J_{enc})})), \quad (6)$$

$$S_{dec}^{(j)} = LN(G_{dec}^{(j)} + FFN(G_{dec}^{(j)})), \quad (7)$$

ここで,  $S_{dec}^{(0)}$  はデコーダへの入力,  $H_{dec}^{(j)}$  は第  $j$  層目のデコーダの self-attention の出力,  $S_{enc}^{(J_{enc})}$  はエンコーダの最終層の出力,  $G_{dec}^{(j)}$  は第  $j$  層目のデコーダの encoder-decoder attention の出力である. デコーダの最終層の出力  $S_{dec}^{(J_{dec})}$  は  $|\mathcal{V}|$  次元の行列に線形変換される. ここで,  $\mathcal{V}$  は出力の語彙集合であり,  $|\mathcal{V}|$  は出力の語彙数である. 最後に,  $|\mathcal{V}|$  次元行列に softmax 関数を適用することで  $P(Y|X)$  が計算され, これに基づいて出力列  $Y$  が生成される. モデルの学習はラベル平滑化交差エントロピー (Szegedy, Vanhoucke, Ioffe, Shlens, and Wojna 2016) による  $\mathcal{L}_{token}$  を最小化することで実現される.

$$\mathcal{L}_{token} = - \sum_{i=1}^{n_{dec}} \left( (1 - \epsilon) \log P(y_i | X, y_{<i}) + \epsilon \sum_{v \in \mathcal{V}} \log P(v | X, y_{<i}) \right) \quad (8)$$

ただし,  $\epsilon \geq 0$  は正解ラベルに対する信頼度をコントロールするラベル平滑化のハイパーパラメータである.

self-attention は同一文中の単語間 (原言語文中の単語間あるいは目的言語文中の単語間) の関連の強さを計算し, encoder-decoder attention は原言語文中の単語と目的言語文中の単語間の関連の強さを計算する. self-attention や encoder-decoder attention は multi-head attention によって実現される. multi-head attention では, 単語の埋め込み空間を  $n_{head}$  個の  $d_{head} = d/n_{head}$  次元部分空間に射影し, それぞれの部分空間で attention を計算する. 具体的には, self-attention ( $SelfAttn()$ ) の第  $j$  層目は, 直前の層の出力  $S^{(j-1)} \in \mathbb{R}^{n \times d}$  (ただし,  $n$  はエンコーダあるいはデコーダの入力系列長) を,  $W_h^{Q(j)} \in \mathbb{R}^{d \times d_{head}}$ ,  $W_h^{K(j)} \in \mathbb{R}^{d \times d_{head}}$ ,  $W_h^{V(j)} \in \mathbb{R}^{d \times d_{head}}$  を用いて  $d_{head}$  次元の部分空間  $Q_h^{(j)}$ ,  $K_h^{(j)}$ ,  $V_h^{(j)}$  に射影する (なお,  $1 \leq h \leq n_{head}$ ). デコーダで用いられる encoder-decoder attention ( $EncDecAttn()$ ) では, 直前のデコーダレイヤの出力を部分空間  $Q_h^{(j)}$  に, エンコーダの最終層の出力を部分空間  $K_h^{(j)}$ ,  $V_h^{(j)}$  に射影する. 射影後, 次の式によって各部分空間で単語間の関連の強さを表す行列を算出する.

$$A_h^{(j)} = \text{softmax}(d_{head}^{-0.5} Q_h^{(j)} K_h^{(j)\top}) \quad (9)$$

この  $A_h^{(j)}$  に対して  $V_h^{(j)}$  を掛け合わせることで, 単語間の関連の強さを重みとする荷重和による表現  $M_h^{(j)}$  を得ることができる.

$$M_h^{(j)} = A_h^{(j)} V_h^{(j)} \quad (10)$$

最後に, 各部分空間の  $M_{1,2,\dots,n_{head}}^{(j)}$  を結合させて, 単語の埋め込み次元に線形変換する.

$$M^{(j)} = W^{M^{(j)}} [M_1^{(j)}; M_2^{(j)}; \dots; M_{n_{head}}^{(j)}] \quad (11)$$

ただし,  $W^{M^{(j)}} \in \mathbb{R}^{d \times d}$  はパラメータ行列である. self-attention の  $M_h^{(j)}$  は原言語文あるいは目的言語文内の全ての単語間の関連の強さを含んでおり, encoder-decoder attention の  $M_h^{(j)}$  は原言語文の単語と目的言語文の単語の間の全ての関連の強さを含んでいる. また, デコーダの self-attention は, 推論時に予測していない単語に関する関連を求めないように未来の単語情報をマスクして訓練する. 具体的には, 後方の単語に対して attention が向けられないよう,  $Q_h^{(j)}$  の各単語から見た後方の単語を表す  $K_h^{(j)}$  の各成分を  $-\infty$  に置換してから  $\text{softmax}$  を計算し,  $A_h^{(j)}$  を求める.

### 3 提案手法

提案モデルの概要を図2に示す. 本稿で提案するモデルは, 係り受け関係を捉える self-attention (dependency-based self-attention) を Transformer NMT モデルのエンコーダとデコーダに組み込む. 具体的には,  $p_{enc}$  層目のエンコーダの multi-head self-attention と  $p_{dec}$  層目のデコーダ

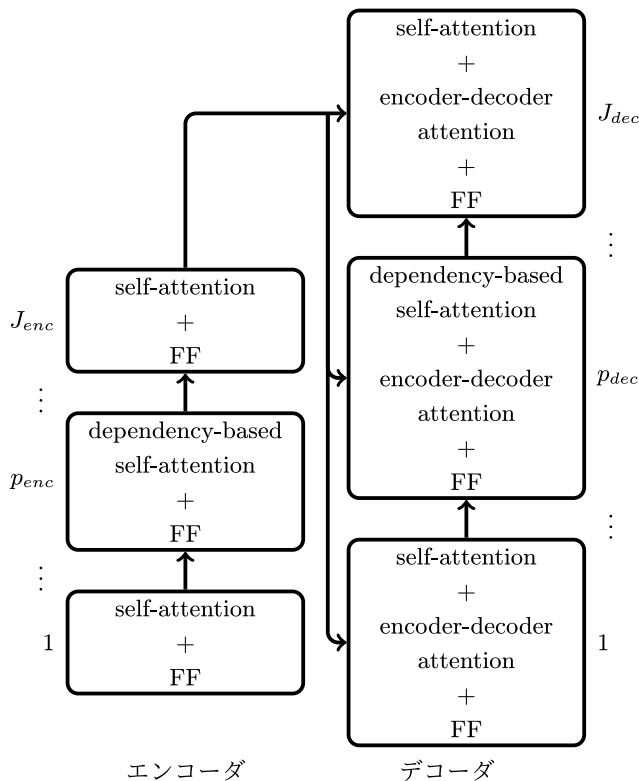


図2 提案モデルの概要

の multi-head self-attention のそれぞれにおいて, 1 つのヘッドで原言語文または目的言語文の係り受け関係を捉えることで, 原言語側および目的言語側の係り受け構造を考慮した翻訳を行う. 係り受け解析は, LISA 同様, Dozat らの deep bi-affine parser (Dozat and Manning 2016) を用いる. dependency-based self-attention と LISA との違いは次の 2 点にまとめられる: (1) dependency-based self-attention はデコーダでは未来の単語に関する係り受け関係をマスクして学習する, (2) dependency-based self-attention は単語単位だけではなく byte pair encoding などによって作られるサブワード単位にも適用可能である.

### 3.1 Dependency-Based Self-Attention

dependency-based self-attention は, エンコーダまたはデコーダの  $p$  層目<sup>1</sup>の multi-head self-attention を拡張することにより, 係り受け関係を解析する. 以下では,  $p$  層目の multi-head self-attention について説明する. まず, 直前の層の出力である行列  $S^{(p-1)} \in \mathbb{R}^{n \times d}$  から以下の式により  $Q_{parse}$ ,  $K_{parse}$ ,  $V_{parse}$  を得る.

$$Q_{parse} = S^{(p-1)} W^{Q_{parse}}, \quad (12)$$

$$K_{parse} = S^{(p-1)} W^{K_{parse}}, \quad (13)$$

$$V_{parse} = S^{(p-1)} W^{V_{parse}}. \quad (14)$$

ここで,  $W^{Q_{parse}}$ ,  $W^{K_{parse}}$ ,  $W^{V_{parse}}$  は, それぞれ,  $d \times d_{head}$  次元のパラメータ行列である. 次に, 2 単語間の係り受け関係を示す行列  $A_{parse}$  を, 次のように bi-affine 変換によって求める.

$$A_{parse} = \text{softmax}(Q_{parse} U^{(1)} K_{parse}^\top + Q_{parse} U^{(2)}), \quad (15)$$

ここで,  $U^{(1)} \in \mathbb{R}^{d_{head} \times d_{head}}$  と  $U^{(2)} = \overbrace{(\mathbf{u} \dots \mathbf{u})}^n$ ,  $\mathbf{u} \in \mathbb{R}^{d_{head}}$  はパラメータ行列である. また,  $A_{parse}$  の要素  $A_{parse}[t, q]$  は, 以下の通り, 単語  $q$  が単語  $t$  の係り先である確率を示している.

$$P(q = \text{head}(t) \mid X) = A_{parse}[t, q]. \quad (16)$$

ここで,  $X$  は原言語文もしくは目的言語文である. また, 係り先のないトークン  $t_{ROOT}$  は自分自身を指すように以下のように定義する.

$$t_{ROOT} = \text{head}(t_{ROOT}). \quad (17)$$

次に, 以下のように  $A_{parse}$  と  $V_{parse}$  を掛け合わせることで単語間の係り受け関係の強さを重みとする荷重和による表現を得る.

<sup>1</sup>  $p$  は, エンコーダの場合は  $p_{enc}$ , デコーダの場合は  $p_{dec}$  である.

$$M_{parse} = A_{parse} V_{parse}. \quad (18)$$

あとは, multi-head self-attention の 1 つのヘッドを  $M_{parse}$  とする以外は 2 節で説明した通常の multi-head attention と同様である. つまり,  $M_{parse}$  と  $n_{head} - 1$  個の通常のヘッド ( $M_{2, \dots, n_{head}}^{(p)}$ ) を結合し, 行列  $M^{(p)}$  に線形変換する.

$$M^{(p)} = W^{M^{(p)}} [M_{parse}; M_2^{(p)}; \dots; M_{n_{head}}^{(p)}], \quad (19)$$

ここで,  $W^{M^{(p)}} \in \mathbb{R}^{d \times d}$  はパラメータ行列である. このように, dependency-based self-attention は係り受け関係を第  $p$  層の multi-head attention の 1 つのヘッド  $M_{parse}$  で捉える.

### 3.2 目的関数

提案モデルは次の目的関数  $\mathcal{L}$  を最小化することにより, 翻訳と係り受け解析を同時に学習する.

$$\mathcal{L} = \mathcal{L}_{token} + \lambda_{enc} \mathcal{L}_{encparse} + \lambda_{dec} \mathcal{L}_{decparse}, \quad (20)$$

ここで,  $\mathcal{L}_{token}$  は翻訳に関する損失であり,  $\mathcal{L}_{encparse}$  と  $\mathcal{L}_{decparse}$  はそれぞれエンコーダ側とデコーダ側の係り受け解析に関する損失を表している. また,  $\lambda_{enc} > 0$  と  $\lambda_{dec} > 0$  はハイパーパラメータである.  $\mathcal{L}_{token}$  は通常の Transformer モデル同様, ラベル平滑化交差エントロピーによって計算し,  $\mathcal{L}_{encparse}$  と  $\mathcal{L}_{decparse}$  は交差エントロピーによって次式のように計算する.

$$\mathcal{L}_{parse} = - \sum_{t,q} \hat{A}_{parse}[t,q] \times \log(A_{parse}[t,q]) \quad (21)$$

ただし,  $\mathcal{L}_{parse}$  は  $\mathcal{L}_{encparse}$  または  $\mathcal{L}_{decparse}$  である. また,  $\hat{A}_{parse}$  は  $A_{parse}$  の教師データであり, 各トークン  $t$  に対して係り先  $head(t)$  を指す要素のみが 1 ( $\hat{A}_{parse}[t, head(t)] = 1$ ), それ以外の要素は 0 の行列である.

デコーダ側の係り受け解析に関する損失を算出する際は, 自身より後方の単語への係り受け関係はマスクする. 推論時には, デコーダ側の self-attention は予測済みの単語 (自身より前方の単語) に対するアテンションしか捉えないため, 未来の単語へのアテンションを訓練しないようにするためである. デコーダ側の dependency-based self-attention の教師データの例を図 3 に示す. 図 3(a) は対訳データ中の目的言語文の係り受け構造の例であり<sup>2</sup>, 図 3(b) は図 3(a) に対するデコーダ内の dependency-based self-attention の教師データである. 図 3(a) では, 教師データでマスクされる対象となる後方の単語に対する係り受け関係を点線で示している. 図 3(b) では, 黒色のセルが係り受け関係にある要素を表しており, 点線によって囲まれたセルは

<sup>2</sup> 本稿では, 係り受け関係を係り元から係り先に向けた矢印で表現する. 例えば図 3(a) では, “He” の係り先は “knows” (“knows” =  $head(“He”)$ ) である.

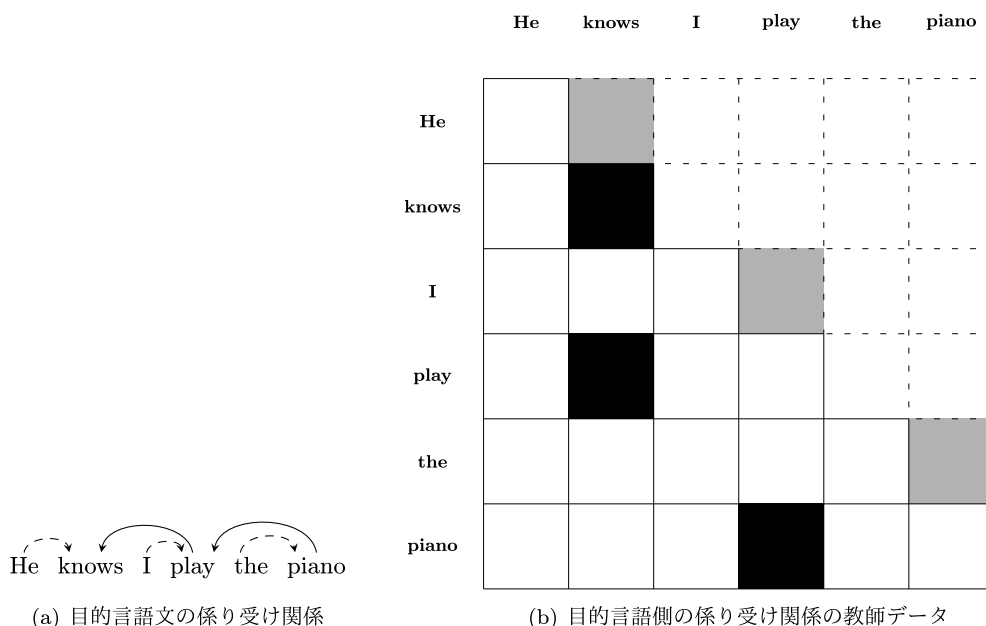


図 3 デコーダ側のマスク付き dependency-based self-attention

マスクされた要素であることを表しており、灰色のセルは係り受け関係にあった要素がマスクされたことを表している。図 3(b) の通り、“knows” が “He” の係り先であることを示す要素など、後方の単語に対する係り受け関係がマスクされている。

### 3.3 Subword Dependency-Based Self-Attention

近年、ニューラル機械翻訳は文を単語列ではなく、BPE (Sennrich et al. 2016) などによって分割されたサブワード列として扱うことにより、翻訳の性能を向上させている。そのため、本研究では、dependency-based self-attention をサブワード列に対しても適用できるように拡張する。まず、単語単位の係り受け構造をサブワード単位の係り受け構造に変換し、続いて変換したサブワード単位の係り受け構造に対して dependency-based self-attention を適用する。サブワード単位の係り受け構造の変換については左から右へと係るよう、次の通りを行う。1 つの単語が複数のサブワードに分割される場合、右端のサブワードの係り先 (head) は分割元の単語の係り先とし、右端以外の各サブワードの係り先は隣接する右側のサブワードとする。また、係り先の単語がサブワードに分割される場合は、分割されたサブワードの中で左端のサブワードを係り先とする。以降、サブワード列に拡張した dependency-based self-attention を subword dependency-based self-attention と呼ぶ。

サブワード単位の係り受け関係の例を図 4 に示す。なお、“@@” はサブワードの分割を表した



図 4 サブワード単位の係り受け関係

シンボルである。この例では，“listen”という単語が3つのサブワード，“li@@”，“s@@”，“ten”に分割されている。このとき，“listen”が係り先となっている場合は左端のサブワード“li@@”が係り先になる。また，“listen”を構成するサブワードの係り先は、右端のサブワード“ten”は分割元の単語“listen”の係り先、それ以外のサブワードは隣接する右側のサブワードとなる。具体的には，“li@@”の係り先は“s@@”，“s@@”の係り先は“ten”となる。

## 4 実験

### 4.1 実験設定

本実験では、提案モデルの有効性を確認するため、提案モデルと係り受け構造が組み込まれていない従来の Transformer NMT モデルの翻訳性能を比較する。Transformer NMT モデルには Transformer base モデル (Vaswani et al. 2017) を使用し、エンコーダとデコーダはそれぞれ 6 層積み重ねた。

翻訳性能の評価には Asian Scientific Paper Excerpt Corpus (ASPEC) (Nakazawa et al. 2016) の日英・英日翻訳タスクを用いた。訓練データには ASPEC 日英対訳コーパスの `train-1.txt` と `train-2.txt` から上位 150 万文対を抽出して用いた。データの前処理は、日本語文の単語分割に *KyTea* (Neubig, Nakata, and Mori 2011) を用いた以外は WAT の baseline system の構築方法<sup>3</sup>に従った。サブワード分割には BPE を用い、語彙数は 100,000 トークンとした。また、サブワード分割した文のトークン数が原言語側と目的言語側ともに 250 以下であり、なおかつ対訳文対のトークン数の比が 1.5 以内である 1,198,149 文対を用いて訓練した。実験データの対訳文対数を表 1 に示す。

訓練時に用いるエンコーダとデコーダそれぞれの dependency-based self-attention の係り受け構造の教師データには、予め訓練データの各文を係り受け解析器によって解析したものを使用した。日本語側の係り受け解析には EDA<sup>4</sup> を用い、英語側の係り受け解析には Stanford CoreNLP<sup>5</sup> を用いた。

従来モデルと提案モデルの学習には Adam (Kingma and Ba 2014) を用い、学習率などのハ

<sup>3</sup> <http://lotus.kuee.kyoto-u.ac.jp/WAT/WAT2018/baseline/dataPreparationJE.html>

<sup>4</sup> <http://www.ar.media.kyoto-u.ac.jp/tool/EDA>

<sup>5</sup> <https://nlp.stanford.edu/software/stanford-dependencies.html>

イパーパラメータの設定は Vaswani ら (Vaswani et al. 2017) に従った. dependency-based self-attention を組み込むエンコーダレイヤ  $p_{enc}$ , デコーダレイヤ  $p_{dec}$  は, それぞれ  $p_{enc} = 4$ ,  $p_{dec} = 4$  とした. 目的関数の  $\mathcal{L}_{token}$  の算出に用いるラベル平滑化交差エントロピーの  $\epsilon$  (Szegedy et al. 2016) は 0.1 とし,  $\lambda_{enc}$  と  $\lambda_{dec}$  はどちらも 1.0 とした. ミニバッチの大きさは 224 文とし, エポック数は 20 とした. 毎エポックの終わりに開発データを用いて翻訳文を生成し, BLEU スコアの最も高いモデルによって評価した. 評価時はビームサーチを用いて翻訳し, ビーム幅は 4, length penalty は  $\alpha = 0.6$  (Wu, Schuster, Chen, Le, Norouzi, Macherey, Krikun, Cao, Gao, Macherey, et al. 2016) とした.

## 4.2 実験結果

表 2 に実験結果を示す. 翻訳性能は BLEU (Papineni, Roukos, Ward, and Zhu 2002) によって評価した. 表 2 において, “DBSA” は提案手法の subword dependency-based self-attention を表している. 実験結果より, 提案モデルである “Transformer + DBSA(Enc) + DBSA(Dec)” がベースラインモデルである “Transformer” より高い翻訳性能であることが分かり, 提案手法の有効性が確認できる. また, 表 2 より, 日英翻訳においてはエンコーダ側, デコーダ側それぞれの dependency-based self-attention が翻訳性能の改善に寄与しており, それらを組み合わせることでさらに性能が改善できることが分かる. 英日翻訳においては, 目的言語側の日本語が後ろの単語に係る文構造を持っているため, デコーダ側のみに制約を付与した場合は翻訳性能にほぼ変化が無かったが, エンコーダ側に制約を付与したモデルは, エンコーダ側単体・デコーダ側の制約との組み合わせを問わず, どちらも性能が改善していることが確認できる.

表 1 実験データの対訳文対数

|       | 対訳文対      |
|-------|-----------|
| 訓練データ | 1,198,149 |
| 開発データ | 1,790     |
| 評価データ | 1,812     |

表 2 ASPEC 日英・英日翻訳の実験結果 (単位: BLEU (%))

| モデル                                 | 日英           | 英日           |
|-------------------------------------|--------------|--------------|
| Transformer                         | 26.92        | 40.55        |
| Transformer + DBSA(Enc)             | 27.70        | 40.70        |
| Transformer + DBSA(Dec)             | 27.54        | 40.53        |
| Transformer + DBSA(Enc) + DBSA(Dec) | <b>27.96</b> | <b>40.85</b> |

表 3 サブワードの有効性 (単位: BLEU (%))

| モデル                                 | BPE | 日英           | 英日           |
|-------------------------------------|-----|--------------|--------------|
| Transformer                         | w/o | 25.11        | 38.38        |
| Transformer                         | w/  | 26.92        | 40.55        |
| Transformer + DBSA(Enc) + DBSA(Dec) | w/o | 25.33        | 39.16        |
| Transformer + DBSA(Enc) + DBSA(Dec) | w/  | <b>27.96</b> | <b>40.85</b> |

## 5 考察

### 5.1 Dependency-Based Self-Attention のサブワード単位への拡張の有効性

dependency-based self-attention のサブワード単位への拡張の有効性を調べるため, BPE を用いずに各文を単語列として扱った場合の性能比較を行う. BPE を用いないモデルでは, 訓練データに出現する頻度が 5 未満の単語を “<unk>” というトークンに置換した. 実験結果を表 3 に示す. 実験結果より, BPE を用いることで従来モデルと提案モデル両方とも翻訳性能が向上することが分かる. この結果から, dependency-based self-attention のサブワードへの拡張は有用であることが確認できる. また, 表 3 より, BPE を用いない場合においても, 提案モデルの方が従来モデルよりも翻訳性能が良いことが分かる. これより, 提案モデルはサブワード系列だけではなく単語系列に対しても有効であることが分かる.

### 5.2 Dependency-Based Self-Attention が捉える係り受け構造の評価

dependency-based self-attention が係り受け関係を学習しているか確認するため, 評価データに対する各単語の係り先を予測し, 評価する. 具体的には, 評価データを入力としたときの  $A_{parse}$  に対して, 各単語  $t$  の係り先の予測結果  $head_{pred}(t)$  を  $\text{argmax}$  により以下のように得る.

$$head_{pred}(t) = \text{argmax}(A_{parse}[t]) \quad (22)$$

正解データは, 評価データに対して係り受け解析器を用いた出力結果を用い, 訓練時に与える教師データと同じ方法により作成する. モデルの係り先予測結果と正解データの一致率を以下の式により求め, 評価する.

$$\frac{1}{T} \sum_t^T \delta_{head(t), head_{pred}(t)} \quad (23)$$

ただし,  $T$  は系列長,  $\delta_{i,j}$  は  $i=j$  のとき 1,  $i \neq j$  のとき 0 となるクロネッカーのデルタ,  $head(t)$  は単語  $t$  に対する係り先の正解データである.

評価データには評価時と同じ ASPEC の評価データを用いた. なお, 目的言語側についてはモデルの出力文と参照訳が異なり, 係り先の予測結果を評価できないため, 原言語側の  $A_{parse}$  に

についてのみ評価を行った。係り先予測に用いるモデルは原言語・目的言語両側に対して subword dependency-based self-attention を組み込んだモデルを用い、係り先の正解データはサブワード単位で作成した。実験結果であるモデルの係り先予測結果と正解データとの一致率は、日英翻訳実験において日本語の係り受け一致率が **98.57%**、英日翻訳実験において英語の係り受け一致率が **93.05%** となった。これらの結果より、モデルが係り受け関係を正しく捉えていることが確認できる。

### 5.3 従来モデルと提案モデルの翻訳文の違い

dependency-based self-attention を組み込むことで具体的にどのような翻訳が良くなったのかを調べる。比較には、従来モデルがサブワード単位の Transformer モデル、提案モデルが原言語・目的言語両側に対して subword dependency-based self-attention を組み込んだ Transformer + DBSA(Enc) + DBSA(Dec) モデルを用いた。

従来モデルと提案モデルの翻訳結果の違いを表 4 に示す。ただし、表 4 に示した例文は翻訳結果の差異が特徴的な部分を抽出しており、文中の太字は従来モデルによる翻訳文と提案モデルによる翻訳文の係り受け関係が異なる箇所を示した。また、原言語文の太字部分を係り受け解析した結果を図 5 に示す。

表 4 の原言語文の太字中の「without neurological abnormality」という項は、図 5 の通り、「without」の直前の項「21 infant cases」に係っていることが分かる。従来モデルの翻訳文は「神経学的異常のない」という項が「無症候性脳梗塞患者 20 名」に対して係っており、原言語文の係り受け関係とは異なる誤った係り受け関係になっているが、提案モデルはこの係り受け関係

表 4 従来モデルと提案モデルの翻訳結果の違い（英日翻訳）

|                |                                                                                                                                                       |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| 原言語文           | In 20 asymptomatic cerebral infarction patients, 10 normal persons, and <b>21 infant cases without neurological abnormality</b> in fiscal 2000, ..... |
| 翻訳文<br>(従来モデル) | 2000年度に <b>神経学的異常のない</b> 無症候性脳梗塞患者20名、正常者10名、小児21名を .....                                                                                             |
| 翻訳文<br>(提案モデル) | 2000年度の無症候性脳梗塞患者20例、正常者10例、 <b>神経学的異常のない</b> 小児21例に .....                                                                                             |
| 正解文            | 2000年度は、無症候性脳梗塞患者20名、正常人10名、 <b>神経学的異常を有して</b> いない小児21例に .....                                                                                        |

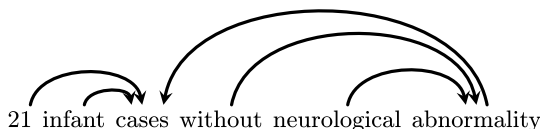


図 5 表 4 の原言語文の太字部分の係り受け解析結果

を正しく捉えて翻訳しており、「小児 2 1 例」に対して係った文となっていることを確認した。

## 5.4 提案モデルの設定に関する比較実験

本節では、提案モデルにおける、(1) サブワードの係り先、(2) 制約を与えるヘッド数、(3) bi-affine 変換の有効性 について、それぞれ比較実験を行う。なお、全ての比較実験には原言語側と目的言語側の両側に対して subword dependency-based self-attention を組み込んだ提案モデルを用いる。

### 5.4.1 サブワードの係り先方向の比較実験

subword dependency-based self-attention におけるサブワードの係り先の設定について比較実験を行った。提案モデルでは、単語がサブワード単位に分割されたとき、単語内の各サブワードの係り先はそれぞれ隣接する右のサブワードへ係るように設定した。本実験では、単語内の各サブワードの係り先を隣接する左のサブワードへ係るように設定したモデルを用いて比較した。実験結果を表 5 に示す。なお、表中の DBSA-L2R は本稿で提案した左から右へと係る構造を用いたモデル、DBSA-R2L は比較対象の右から左へと係る構造を用いたモデルを示す。表 5 より、単語がサブワード単位に分割されたときの係り先は、各サブワードが隣接する右に向かって、左から右へと係る構造を用いたほうが性能改善に寄与できることを確認した。

### 5.4.2 制約を与えるヘッド数の比較実験

提案モデルは、multi-head self-attention のヘッドから選択した単一のヘッドにのみ制約を与えて学習する。本節では、制約を与えるヘッドを単一ヘッドとマルチヘッド全体で比較し、翻訳性能の比較を行った。実験結果を表 6 に示す。表 6 より、制約を与えるヘッドはマルチヘッド全体よりも単一ヘッドに設定したほうが性能改善に寄与できることを確認した。

表 5 サブワードの係り先の方向の比較実験 (BLEU (%))

| モデル      | 日英           | 英日           |
|----------|--------------|--------------|
| DBSA-L2R | <b>27.96</b> | <b>40.85</b> |
| DBSA-R2L | 27.40        | 40.64        |

表 6 制約を与えるヘッド数の比較実験 (BLEU (%))

| 制約を与える head 数 | 日英           | 英日           |
|---------------|--------------|--------------|
| 1 (単一ヘッド)     | <b>27.96</b> | <b>40.85</b> |
| 8 (マルチヘッド全体)  | 27.05        | 40.52        |

表 7 提案モデルにおける bi-affine 変換の有無と翻訳性能の差 (BLEU (%))

| bi-affine 変換 | 日英           | 英日           |
|--------------|--------------|--------------|
| w/           | <b>27.96</b> | <b>40.85</b> |
| w/o          | 27.22        | 40.65        |

表 8 提案モデルにおける bi-affine 変換の有無と翻訳結果の違い (日英翻訳)

|                        |                                                                                                          |
|------------------------|----------------------------------------------------------------------------------------------------------|
| 原言語文                   | 肥満 と NASH の 関連 及び NASH と HCC の 関連 について 述べた。                                                              |
| 翻訳文<br>(w/o bi-affine) | the relation between NASH and obesity <b>and</b> NASH and relation between NASH and HCC were described . |
| 翻訳文<br>(w/ bi-affine)  | relation between obesity and NASH and the relation between NASH and HCC were described .                 |
| 正解文                    | the relation of obesity and NASH , and relation of NASH and HCC was described .                          |

### 5.4.3 bi-affine 変換の有無に関する比較実験

提案モデルでは係り受け関係を捉える self-attention で bi-affine 変換を用いているが, bi-affine 変換を用いず式 (9) で得られる確率分布行列に制約を与えたモデルとの比較実験を行った. 実験結果を表 7 に示す. 表 7 より, bi-affine 変換を用いたほうが性能改善に寄与できることを確認した.

また, BLEU 値による評価だけでなく, bi-affine 変換の有無による翻訳結果の違いを調べた. bi-affine 変換の有無による翻訳結果の違いを表 8 に示す. 翻訳結果は文中における差異が特徴的な部分のみを抽出した. 表中の太字は単語間の関係性を誤って捉えていると考えられる箇所である. 表 8 より, bi-affine 変換を用いることで複雑な係り受け関係を捉えることが可能となり, 単語間の関係を正しく捉えられるようになることが確認できる.

## 6 関連研究

これまで, ニューラル機械翻訳は係り受け構造や句構造を考慮することにより, 翻訳性能が改善されてきた. Chen ら (Chen et al. 2017) は, RNN ベースのニューラル機械翻訳モデルのエンコーダの中で, 原言語文の係り受けの親子関係の情報を CNN で畳み込んだベクトルを用いる “SDRNMT” を提案している. Eriguchi ら (Eriguchi et al. 2017) は, RNN ベースのニューラル機械翻訳モデルのデコーダに RNN ベースの係り受け解析モデルである RNNG (Dyer, Kuncoro, Ballesteros, and Smith 2016) を組み込んだ “NMT+RNNG” を提案している.

Wu ら (Wu et al. 2018) は, RNN ベースのニューラル機械翻訳モデルや Transformer NMT モデルにおいて, エンコーダで原言語文の係り受け木を線形化した系列情報を符号化し, デコーダで目的言語文と共にその係り受け木を予測する機構を組み合わせた手法を提案している. Wu ら

の手法では Transformer のエンコーダとデコーダに係り受け関係を考慮できるが、Transformer NMT のモデル自体は改良されていない。一方で、本研究では、Transformer NMT モデルの self-attention を改良し、self-attention の中で係り受け関係を考慮する点が異なる。また、Wu らの手法では、推論時に係り受け構造を考慮するためには、原言語の文を既存の係り受け解析器で解析する必要がある。一方で、本研究の提案手法は、推論時には、訓練時に原言語文の係り受け関係を捉えるように学習した self-attention を用いて推論を行うため、推論時は係り受け解析器が必要ない点も異なる。

Currey ら (Currey and Heafield 2019) は、Transformer のエンコーダに原言語文の句構造を組み込んだモデルを提案している。Currey らの手法では句構造を用いているが、我々の手法では係り受け構造を用いている点が異なる。また、Currey らは原言語側の文構造のみを考慮しているが、本研究の提案手法では原言語と目的言語の両側の文構造を考慮している点も異なる。

Omote ら (Omote, Tamura, and Ninomiya 2019) は、Transformer のエンコーダの self-attention で原言語文の 2 単語の係り受け木における相対的な位置関係を捉えるモデルを提案している。Omote らの手法では、原言語側の係り受け構造のみを考慮しているが、本研究の提案手法では目的言語側の係り受け構造も考慮している。また、Omote らの手法は、Wu ら (Wu et al. 2018) の手法同様、推論時に係り受け解析器が必要な点も本研究とは異なる。

## 7 まとめ

本稿では、原言語側と目的言語側両方の係り受け関係を捉える self-attention (dependency-based self-attention) を組み込んだ新たな Transformer NMT モデルを提案した。デコーダ側の dependency-based self-attention を訓練する際は、訓練時と推論時で不整合が生じないようにするため、自身より後方の係り受け関係をマスクした教師データを用いた。さらに、dependency-based self-attention をサブワード系列にも適用できるように拡張した。実験結果より、提案モデルを用いることで、ASPEC 日英・英日翻訳タスクにおいて、ベースラインの Transformer モデルと比較して BLEU スコアがそれぞれ 1.04 ポイント・0.30 ポイント上昇したことが確認できた。また、評価実験を通じて、エンコーダとデコーダそれぞれの dependency-based self-attention の有効性と dependency-based self-attention のサブワード系列への拡張の有効性も確認した。今後は、他の言語対でも提案モデルの有効性も確認したい。

## 謝 辞

本論文は国際会議 International Conference Recent Advances in Natural Language Processing 2019 (RANLP2019) で発表した論文 (Deguchi, Tamura, and Ninomiya 2019) に基づいて日本語

で書き直し, 説明を追加したものである.

本研究成果は, 国立研究開発法人情報通信研究機構の委託研究により得られたものである. また, 本研究の一部は JSPS 科研費 18K18110 の助成を受けたものである. ここに謝意を表する.

また, 本稿の英文校正を担当していただいたエナゴ ([www.enago.jp](http://www.enago.jp)) に謝意を表する.

## 参考文献

- Chen, K., Wang, R., Utiyama, M., Liu, L., Tamura, A., Sumita, E., and Zhao, T. (2017). “Neural Machine Translation with Source Dependency Representation.” In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2846–2852.
- Currey, A. and Heafield, K. (2019). “Incorporating Source Syntax into Transformer-Based Neural Machine Translation.” In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pp. 24–33.
- Deguchi, H., Tamura, A., and Ninomiya, T. (2019). “Dependency-Based Self-Attention for Transformer NMT.” In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2019 (RANLP 2019)*, pp. 239–246.
- Dozat, T. and Manning, C. D. (2016). “Deep Biaffine Attention for Neural Dependency Parsing.” *arXiv preprint arXiv:1611.01734*.
- Dyer, C., Kuncoro, A., Ballesteros, M., and Smith, N. A. (2016). “Recurrent Neural Network Grammars.” In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 199–209.
- Eriguchi, A., Tsuruoka, Y., and Cho, K. (2017). “Learning to Parse and Translate Improves Neural Machine Translation.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 72–78.
- Kingma, D. P. and Ba, J. (2014). “Adam: A Method for Stochastic Optimization.” *arXiv preprint arXiv:1412.6980*.
- Lei Ba, J., Kiros, J. R., and Hinton, G. E. (2016). “Layer Normalization.” *arXiv preprint arXiv:1607.06450*.
- Lin, D. (2004). “A Path-based Transfer Model for Machine Translation.” In *Proceedings of the 20th International Conference on Computational Linguistics*, pp. 625–630.
- Nakazawa, T., Yaguchi, M., Uchimoto, K., Utiyama, M., Sumita, E., Kurohashi, S., and Isahara, H. (2016). “ASPEC: Asian Scientific Paper Excerpt Corpus.” In *Proceedings of LREC 2016*, pp. 2204–2208.
- Neubig, G., Nakata, Y., and Mori, S. (2011). “Pointwise Prediction for Robust, Adaptable

- Japanese Morphological Analysis.” In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 529–533.
- Omote, Y., Tamura, A., and Ninomiya, T. (2019). “Dependency-Based Relative Positional Encoding for Transformer NMT.” In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2019 (RANLP 2019)*, pp. 854–861.
- Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). “BLEU: A Method for Automatic Evaluation of Machine Translation.” In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 311–318.
- 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)*, pp. 1715–1725, Berlin, Germany.
- Strubell, E., Verga, P., Andor, D., Weiss, D., and McCallum, A. (2018). “Linguistically-Informed Self-Attention for Semantic Role Labeling.” In *Proceedings of EMNLP 2018*, pp. 5027–5038.
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). “Rethinking the Inception Architecture for Computer Vision.” In *The IEEE Conference on CVPR*, pp. 2818–2826.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). “Attention is All you Need.” In *Advances in Neural Information Processing Systems 30*, pp. 5998–6008.
- Wu, S., Zhang, D., Zhang, Z., Yang, N., Li, M., and Zhou, M. (2018). “Dependency-to-Dependency Neural Machine Translation.” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **26** (11), pp. 2132–2141.
- Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation.” *arXiv preprint arXiv:1609.08144*.

## 略歴

出口 祥之：2019年愛媛大学工学部情報工学科卒業。2019年より同大学院理工学研究科博士前期課程に在学。

田村 晃裕：2005年東京工業大学工学部情報工学科卒業。2007年同大学院総合理工学研究科修士課程修了。2013年同大学院総合理工学研究科博士課程修了。日本電気株式会社、国立研究開発法人情報通信研究機構にて研究員として務めた後、2017年より愛媛大学大学院理工学研究科助教、2020年より同志社大学理工学部准教授となり、現在に至る。博士（工学）。言語処理学会、情報処理学会、人工知能学会、ACL各会員。

**二宮 崇**：1996 年東京大学理学部情報科学科卒業．1998 年同大学大学院理学系研究科修士課程修了．2001 年同大学大学院理学系研究科博士課程修了．同年より科学技術振興事業団研究員．2006 年より東京大学情報基盤センター講師．2010 年より愛媛大学大学院理工学研究科准教授，2017 年同教授．博士（理学）．言語処理学会，アジア太平洋機械翻訳協会，情報処理学会，人工知能学会，電子情報通信学会，日本データベース学会，ACL 各会員．

(2019 年 12 月 30 日 受付)

(2020 年 4 月 8 日 再受付)

(2020 年 5 月 11 日 採録)