機械翻訳のための係り受け構造に基づくトランスフォーマーモデルの研究

Studies on Dependency-Based Transformer Models for Machine Translation

愛媛大学大学院理工学研究科教授
二宮　崇

2001 年東京大学大学院理学系研究科情報科学専攻博士課程修了。博士（理学）。2017 年より愛媛大学大学院理工学研究科教授。自然言語処理の研究に従事。

愛媛大学大学院理工学研究科電子情報工学専攻
表　悠太郎

2019 年愛媛大学工学部情報工学科卒業。現在、愛媛大学大学院理工学研究科電子情報工学専攻博士前期課程在学中。自然言語処理の研究に従事。

愛媛大学大学院理工学研究科電子情報工学専攻
出口　祥之

2019 年愛媛大学工学部情報工学科卒業。現在、愛媛大学大学院理工学研究科電子情報工学専攻博士前期課程在学中。自然言語処理の研究に従事。

愛媛大学大学院理工学研究科助教
田村　晃裕

2013 年東京工業大学大学院総合理工学研究科博士課程修了。博士（工学）。2017 年より愛媛大学大学院理工学研究科助教。自然言語処理の研究に従事。

1 はじめに

機械翻訳は自然言語処理の初期から盛んに研究され、様々な手法が提案されてきているが、近年では、ニューラルネットワークを用いた機械翻訳（ニューラル機械翻訳）が高い精度と自然な翻訳を実現することから盛んに研究されている。特に、同一文内の単語間の関係を捉える自己アテンション（Self Attention）という構造を用いたトランスフォーマー（Transformer）[1] に基づくモデルが最も高い精度を実現することから現在大きく注目されている。従来のニューラル機械翻訳は、大きく分けると、各言語の中間表現を求める計算機構として、畳み込みニューラルネットワーク（Convolutional Neural Network; CNN）を用いるニューラル機械翻訳 [2] と再帰型ニューラルネットワーク（Recurrent Neural Network; RNN）を用いるニューラル機械翻訳 [3][4] に分けられていたが、トランスフォーマーはいずれのモデルとも異なり、各言語の中間表現を計算するために、原言語文（翻訳元言語の文）や目的言語文（翻訳先言語の文）のすべての単語の組み合わせに対する

アテンション（自己アテンション）を計算して中間表現を求める点を大きな特徴とする。トランスフォーマーでは、語順や前後関係など各単語の文中における位置に関する情報は、位置エンコーディング（Positional Encoding）を用いて各単語の埋め込み表現に付随させている。

トランスフォーマーや自己アテンションを改善するための手法がいくつか提案されており、Shaw ら[5]は、単語の絶対的な位置情報に加えて、文中における単語間の相対的な位置関係情報を自己アテンションにおいて考慮することでトランスフォーマーの精度改善を行っている。Strubell ら[6]は、意味役割付与（Semantic Role Labeling）において、構文情報（係り受け構造）を用いて自己アテンションの重み付けを学習するマルチタスク学習や、構文情報を直接自己アテンションに用いる手法を提案している。これまで、統計的機械翻訳やニューラル機械翻訳では、原言語文や目的言語文、あるいはその両方の構文情報（句構造や係り受け構造など）を活用することで翻訳精度が改善されることが知られているため[7][8][9][10]、トランスフォーマーにおいても構文情報を用いることで精度が改善されることが期待される。しかしながら、これまで構文情報を陽に活用した機械翻訳のためのトランスフォーマーモデルはまだ提案されていない。

本稿は、愛媛大学で行っている係り受け構造を用いたトランスフォーマーモデルの研究を 2 つ紹介する。一つは、原言語側の係り受け構造の情報をトランスフォーマーの相対的位置表現に用いた新しいニューラル機械翻訳[11]である。この研究は、Shaw ら[5]が用いている相対的位置関係に注目し、語順に対する相対的位置関係だけを用いるのではなく、原言語文を係り受け解析し、得られた係り受け構造における単語間の相対的位置関係を埋め込んだベクトルを単語埋め込みベクトルに付随させることを行う。もう一つの研究は、意味役割付与で用いられる Strubell ら[6]の手法を機械翻訳に応用し、係り受け構造に基づく自己アテンションと機械翻訳のモデルを同時に学習するマルチタスク学習を行う研究[12]である。構文解析器により得られる係り受け構造を自己アテンションの正解データと捉え、自己アテンションの出力と係り受け構造の差分を小さくする制約を目的関数の一つとして導入することで、マルチタスク学習を実現する。

2 トランスフォーマー[1]

トランスフォーマー[1]は、自己アテンションという構造を持ったエンコーダとデコーダから構成されるニューラル機械翻訳モデルである。トランスフォーマーの概要図を図 1 に示す。トランスフォーマーは、エンコーダレイヤとデコーダレイヤがそれぞれ複数層スタックされたエンコーダ・デコーダ構造を持つ。エンコーダでは、入力された原言語文から中間表現を獲得する処理が行われる。デコーダでは、中間表現から目的言語文を予測し、1 単語ずつ逐次的に出力する処理が行われる。

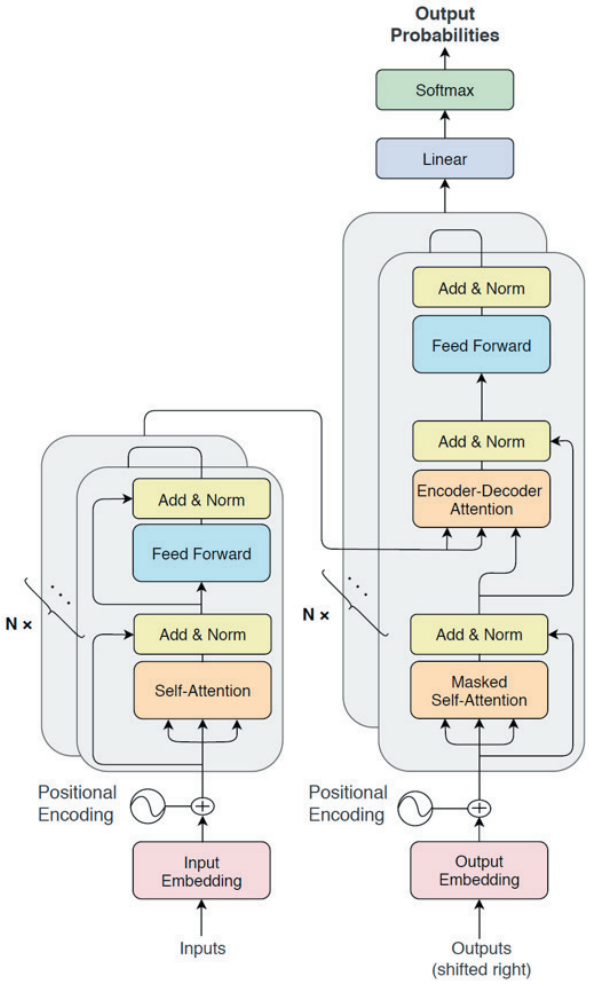


図 1　トランスフォーマーの概要図

トランスフォーマーのエンコーダとデコーダでは、まず、埋込み層で入力単語列（エンコーダ側は原言語文の単語列、デコーダ側は目的言語文の単語列）を埋込み表現を表す行列に変換する。その後、トランスフォーマーは RNN に基づくニューラル機械翻訳とは異なり再帰的

な構造を持たないため、位置エンコーディングにより単語の系列情報を付与する。具体的には、入力単語列の埋込み表現行列に対して、各単語の文における絶対的な位置情報をエンコードした行列 PE を加える。PE の各成分は異なる周波数の sin、cos 関数を用いて次式により算出したものである。

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

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

ここで、d は入力単語の埋込み次元、pos は単語の位置、i は各成分の次元を表す。単語埋込み表現行列に PE を加えたものが、第 1 層目のエンコーダレイヤやデコーダレイヤの入力となる。

エンコーダレイヤは、下位のサブレイヤから順に、原言語文中の単語間の関係を捉える自己アテンション、位置ごとのフィードフォーワードネットワーク（FFN）の2つのサブレイヤで構成されている。デコーダレイヤは、下位のサブレイヤから順に、目的言語文中の単語間の関係を捉えるマスキング付き自己アテンション、原言語文の単語と目的言語文の単語間の関係を捉えるアテンション（言語間アテンション）、位置ごとの FFN の 3 つのサブレイヤで構成されている。

各サブレイヤ間では、残差接続 [13] を行った後にレイヤ正規化 [14] が適用される。レイヤ正規化を適用する関数を LayerNorm、下位のサブレイヤからの出力を x、現在のサブレイヤの処理を行う関数を SubLayer とすると、LayerNorm（x+ SubLayer(x)）がサブレイヤの出力となる。

自己アテンションと言語間アテンションはマルチヘッドアテンションを用いて実現されている。マルチヘッドを用いることは、複数のアテンション機構を持つことに相当し、マルチヘッドの数を h としたとき、入力を線形変換により d 次元ベクトルを d/h 次元に縮退させ、d/h 次元のベクトルを各ヘッドに渡す。アテンション機構への入力列を $x_1,\cdots,x_n$ としたとき、各ヘッドは次の $z_1,\cdots,z_n$ を計算する。

$$z_i = \Sigma_j \alpha_{ij} x_j W^V$$

$$\alpha_{ij} = \exp(e_{ij}) / \Sigma_k \exp(e_{ik})$$

$$e_{ij} = (x_i W^Q)(x_j W^K)^T / (d / h)^{0.5}$$

W^Q、W^K、W^V はそれぞれヘッド毎に定義される重み行列 である。各ヘッドの出力は連結され、重み行列 W^O で線形写像する機構がマルチヘッドアテンションである。

デコーダの自己アテンションでは、デコーダの内部状態系列を用いて計算を行う。ただし、推論時には、予測する単語より後で生成される単語を知ることはできない。そのため、デコーダの自己アテンションでは、予測する単語とそれより後方に位置する単語の間の関係性を考慮しないようにマスクしたマスキング付き自己アテンションを用いる。

デコーダは、最終のデコーダレイヤの出力 $h_1,\cdots,h_n$ に対して線形変換を施し、その後ソフトマックス関数を施すことによって、目的言語の各単語の生成確率分布を得る。

3 係り受け構造に基づく相対的位置表現を用いたトランスフォーマーモデル

本節では、最初に Shaw ら [5] の相対的な位置表現を用いたトランスフォーマーについて説明し、続いて我々の提案手法である係り受け構造に対する相対的位置表現を用いたトランスフォーマーモデルについて説明する。

3.1 相対的位置表現を用いたトランスフォーマー [5]

Shaw ら [5] は、2 単語間の相対的な位置関係をトランスフォーマーエンコーダおよびデコーダ内の自己アテンションで捉える手法を提案した。Shaw らの手法では、入力文中の各単語の中間表現 x_i、x_j 間の関係は重みベクトルベクトル $a^V_{(j-i)}$、$a^K_{(j-i)}$ で表現する。そして、サブレイヤの出力に単語間の相対的位置情報を付加して次の層への入力とする。具体的には、次式を用いて自己アテンションの出力系列 $z_1,\cdots,z_n$ を求める。

$$z_i = \Sigma_j \alpha_{ij} (x_j W^V + a^V_{(j-i)})$$

また、自己アテンション計算過程の e_{ij} も単語間の相対的位置情報を考慮するため、次式を用いる。

$$e_{ij} = (x_i W^Q)(x_j W^K + a^K_{(j-i)})^T / (d / h)^{0.5}$$

Shaw らは、単語間が一定距離以上離れると離れ具

合の影響は少ないと仮定し、相対的位置の距離の最大値を定数 k と定め、それより離れた相対的位置 j − i は k としている。

3.2　提案手法

本節では、最初に係り受け関係について説明し、続いて提案手法となる係り受け構造に対する相対的位置表現を用いたトランスフォーマーモデルについて説明する。

係り受け関係とは、単語間の「修飾」「被修飾」の関係のことであり、方向性を持つ。一例として、“My father bought a red car. ” という文の係り受け構造を図 2 に示す。隣接する 2 単語の関係は上側の単語が被修飾語（主辞）を表し、下側の単語が修飾語を表す。つまり、単語 A が単語 B を修飾するとき、単語 A は単語 B の子ノードになる。係り受け構造は単語をノードとする全域木になっている。

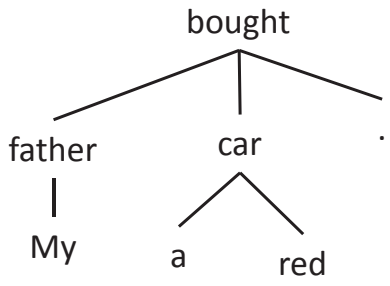


図 2　係り受け構造の例

提案手法では、Shaw ら [5] に倣い、原言語文の係り受け構造における相対的位置ラベルを埋込みベクトルで表現し、原言語文中の 2 単語間の係り受け構造における相対的位置の情報をトランスフォーマーエンコーダ内の自己アテンションに導入する。原言語文中の 2 単語 w_i、w_j に対して、係り受け構造における相対的位置関係を表す位置ラベル dep(i, j) を次の通り与える。

単語 w_i と単語 w_j に対応する係り受け構造のノードをそれぞれ n_i と n_j とすると、dep(i, j) = depth(n_j) − depth(n_i) とする。ここで、depth(n) はノード n の深さを表す。例えば、図 2 において、“My”（= w_1）を基準にした “bought”（= w_3）との位置関係を表す位置ラベルは dep(1, 3) = 0 − 2 = − 2 である。この定義より、係り受け構造においてある単語の親方向は負、子方向は正の値で相対的位置関係が表される。

図 2 の係り受け構造における 2 単語間の相対的位置ラベルを表 1 に示す。表 1 では、行が単語 w_i、列が単語 w_j に対応している。

表 1　係り受け構造に対する相対的位置の例

	My	father	bought	a	red	car	.
My	0	− 1	− 2	0	0	− 1	− 1
father	1	0	− 1	1	1	0	0
bought	2	1	0	2	2	1	1
a	0	− 1	− 2	0	0	− 1	− 1
red	0	− 1	− 2	0	0	− 1	− 1
car	1	0	− 1	1	1	0	0
.	1	0	− 1	1	1	0	0

相対的位置ラベルをもとに、原言語文の各単語の中間表現 x_i、x_j 間の係り受け構造に対する相対的位置関係をベクトル $b^V_{dep(i,j)}$、$b^K_{dep(i,j)}$ で表現し、次式を用いて自己アテンションの出力系列 $z_1,\cdots,z_n$ を求める。

$$z_i = \Sigma_j \alpha_{ij} (x_j W^V + b^V_{dep(i,j)})$$

また、自己アテンション計算過程の e_{ij} も単語間の相対的位置情報を考慮するため、次式を用いる。

$$e_{ij} = (x_i W^Q)(x_j W^K + b^K_{dep(i,j)})^T / (d / h)^{0.5}$$

係り受け構造における相対的位置関係においても、一定以上距離が大きいと離れ具合の影響は少なくなると仮定し、最大距離を定数 k に制限する。

上述の手法を提案手法 1 とする。提案手法 1 に加えて、Shaw ら [5] の提案する文内における相対的位置表現の両方の情報を考慮する手法を提案手法 2 とする。具体的には、$a^V_{(j-i)}$、$b^V_{dep(i,j)}$ と $a^K_{(j-i)}$、$b^K_{dep(i,j)}$ をそれぞれ加算したベクトル c^V_{ij}、c^K_{ij} を 2 単語間の相対的位置情報として用いる。つまり、提案手法 2 は次式を用いて自己アテンションの出力系列 $z_1,\cdots,z_n$ を求める。

$$c^K_{ij} = a^K_{(j-i)} + b^K_{dep(i,j)}$$
$$c^V_{ij} = a^V_{(j-i)} + b^V_{dep(i,j)}$$
$$z_i = \Sigma_j \alpha_{ij} (x_j W^V + c^V_{ij})$$
$$e_{ij} = (x_i W^Q)(x_j W^K + c^K_{ij})^T / (d / h)^{0.5}$$

3.3 実験

本実験では、科学技術論文の概要から作成された対訳文集合である ASPEC（Asian Scientific Paper Excerpt Corpus）[15] を用いて日英翻訳を行った。英語文の単語分割は Moses[16] を用い、日本語文の単語分割は KyTea[17] を用いて行った。また、EDA を用いて係り受け解析を行った。モデルの学習では、英語文・日本語文ともに文長 50 単語以下の 1,341,417 対訳文対を使用した。検証データとして 1,790 文対、テストデータとして 1,812 文対を用いた。

実験では、2 種類の提案手法を、従来の絶対的位置表現を考慮するトランスフォーマー（$\text{Transformer}_{\text{abs}}$）[1] と文中の相対的位置表現を考慮する Transformer（$\text{Transformer}_{\text{rel}}$）[5] と比較する。

表 2　実験結果（日→英）

	BLEU(%)
$\text{Transformer}_{\text{abs}}$ [1]	25.91
$\text{Transformer}_{\text{rel}}$ [5]	26.72
提案手法 1	26.10
提案手法 2	27.22

実験結果を表 2 に示す。なお、翻訳性能の評価指標は BLEU を用いた。表 2 より、提案手法 1 と $\text{Transformer}_{\text{abs}}$ の性能差は 0.19 ポイントであり同等の性能であったが、提案手法 2 は $\text{Transformer}_{\text{abs}}$ と比較して BLEU（%）が 1.31 ポイント上回り、$\text{Transformer}_{\text{rel}}$ と比較して 0.50 ポイント上回った。これらの結果より、日英翻訳タスクにおいては、原言語文の係り受け構造を相対的位置表現で考慮することでトランスフォーマーモデルの翻訳精度を改善できることが分かった。

3.5 まとめ

本研究では、トランスフォーマーにおいて原言語文の係り受け構造を活用するため、原言語の係り受け構造における単語間の相対的位置関係をトランスフォーマーエンコーダの自己アテンションの中の相対的位置表現に導入する手法を提案した。提案手法は係り受け情報を単語埋め込みベクトルに付随させるだけなので、トランスフォーマー全体の仕組みや目的関数を変更する必要がなく、その他のトランスフォーマーの拡張モデルに適応しやすく、拡張性が高い。ASPEC データ [15] を用いた評価実験を通じて、日英翻訳タスクにおいては、原言語文の係り受け構造に対する相対的位置表現を考慮することでトランスフォーマーモデルの精度改善を達成できることを確認した。

4 係り受け構造に基づく自己アテンションを用いたトランスフォーマーモデル

意味役割付与で用いられる Strubell ら [6] の手法を機械翻訳に応用し、係り受け構造に基づく自己アテンションの重み付けと機械翻訳のモデルを同時に学習するマルチタスク学習を行う。

4.1 提案手法

Strubell ら [6] は、意味役割付与のタスクにおいて、係り受け構造に基づく自己アテンションの重み付けと機械翻訳のモデルを同時に学習するマルチタスク学習を提案している。本研究は、彼らの手法に基づき、機械翻訳のためのトランスフォーマーエンコーダとデコーダの両方に彼らの手法を適用し、さらにサブワードに対応することを行う。

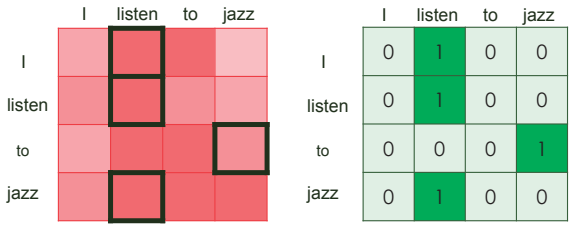
トランスフォーマーの自己アテンション α_{ij} の計算過程において e_{ij} は 2 単語の埋め込み表現に対する内積で計算されていたが、内積の計算の代わりに次式で表されるバイアフィン計算を行い、2 単語間の関係を直接学習する。

$$\alpha_{ij} = \exp(e_{ij}) / \Sigma_k \exp(e_{ik})$$
$$e_{ij} = (x_i W^Q) U (x_j W^K)^T / (d / h)^{0.5}$$

U は 2 単語間の関係を表す重み行列となる。

次に、係り受け解析器により得られる係り受け構造をアテンションの教師データとして捉え、係り受け解析の学習を行う。係り受け構造に対し、単語 w_i が w_j に係る場合、$d_{ij} = 1$ とし、それ以外の場合は $d_{ij} = 0$ とする。例えば、図 3 の係り受け構造に対して、d_{ij} は図 4(b) のような行列となる。

図３　係り受け構造

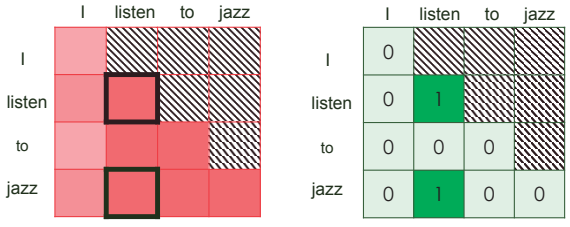


(a) 自己アテンションα_{ij}　(b) 係り受け構造 d_{ij}

図４　自己アテンションと係り受け構造

図４(a) は自己アテンションα_{ij}の大きさを色の濃さで表現したヒートマップであり、図４(a) のヒートマップが図４(b) の係り受け構造に近づくように自己アテンションの学習が行われる。

デコード時には、トランスフォーマーデコーダにおいて、未来に生成される単語の自己アテンションを計算することができないため、ヒートマップおよび係り受け構造のマスキング処理を行う。図５はデコーダのマスキング処理を表す。斜線で表された要素がマスクされており、マスクされた要素の自己アテンションの計算および係り受け構造との差分の計算は行わない。



(a) 自己アテンションα_{ij}　(b) 係り受け構造 d_{ij}

図５　デコーダのマスキング処理

現在の多くの NMT はサブワード列 [19] を入力とするサブワード化が行われているが、係り受け解析は単語単位で行われるため、提案手法に対して直接サブワード化を適用することはできない。本研究では、サブワード化に対応するため、係り受け関係にある単語に対し、(1) 他の単語からの係り先は一番左端のサブワードとし、(2) 単語中のサブワードの係り先は右隣りのサブワードとし、(3) 他の単語に係る係り元は右端のサブワードとすることで対応した。図６は“listen”が“li@@”、“s@@”、“ten”というサブワードに分割された場合の係り受け構造の例を示している。図左側が単語単位の係り受け構造を表し、図右側がサブワード単位の係り受け構造を示している。

図６　係り受け構造のサブワード化

提案手法の学習は、機械翻訳の学習、エンコーダにおける係り受け構造の学習、デコーダにおける係り受け構造の学習に対する目的関数を線形補間したマルチタスク学習により行う。

4.3　実験

本実験では、科学技術論文の概要から作成された対訳文集合である ASPEC（Asian Scientific Paper Excerpt Corpus）[15] を用いて日英翻訳を行った。英語文の単語分割は Moses [16] を用い、係り受け解析は Stanford CoreNLP [18] を用いて行った。日本語文は KyTea [17] を用いて単語分割を行い、EDA を用いて係り受け解析を行った。サブワード化には BPE [19] を用いた。モデルの学習では、サブワード長 250 以下の 1,198,149 対訳文対を使用した。検証データとして 1,790 文対、テストデータとして 1,812 文対を用いた。

表３　実験結果

	BLEU(%)
Transformer[1]	27.29
提案手法	28.29

実験結果を表３に示す。翻訳性能の評価指標には BLEU を用いた。表３より、提案手法を用いることにより BLEU（%）が 1.0 ポイント上昇していることがわかる。この結果より、係り受け構造に対する自己アテンションの学習を行うことで、トランスフォーマーモデルの翻訳精度を改善できることが分かった。

4.4　まとめ

本研究では、原言語および目的言語における係り受け構造を自己アテンションの学習に用いることでトランス

フォーマーモデルを改善する手法を提案した。提案手法は、係り受け構造を自己アテンションの正解とみなすことにより、機械翻訳の学習と、エンコーダの自己アテンションの学習と、デコーダの自己アテンションの学習を同時に学習するマルチタスク学習を行う。デコーダに適用する際にはマスキングを施すことにより、学習時とテスト時の不整合を解消する。サブワード化に対応するため、単語単位の係り受けをサブワード単位の係り受けに変換する手法を提案した。提案手法は、学習時には係り受け構造を解析する係り受け解析器が必要となるが、係り受け解析も同時に学習されるため、テスト時には係り受け解析器を必要としない。ASPEC データ [15] を用いた日英翻訳の評価実験を通じて、提案手法によりトランスフォーマーモデルの精度改善を達成できることを確認した。

謝辞

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。ここに謝意を表する。

参考文献

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. U. Kaiser, and I. Polosukhin. (2017) Attention is all you need. Advances in Neural Information Processing Systems 30, pp. 5998—6008.

[2] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin. (2017) Convolutional sequence to sequence learning. In Proc. of ICML 2017, pp. 1243—1252.

[3] I. Sutskever, O. Vinyals, and Q. V. Le. (2014) Sequence to sequence learning with neural networks. Advances in Neural Information Processing Systems 27, pp. 3104—3112.

[4] T. Luong, H. Pham, and C. D. Manning. (2015) Effective approaches to attention-based neural machine translation. In Proc. of EMNLP 2015, pp. 1412—1421.

[5] P. Shaw, J. Uszkoreit, and A. Vaswani. (2018) Self-attention with relative position representations. In Proc. of NAACL 2018 Short Papers, pp. 464—468.

[6] E. Strubell, P. Verga, D. Andor, D. Weiss, and A. McCallum. (2018) Linguistically-informed self-attention for semantic role labeling. In Proc. of EMNLP 2018, pp. 5027—5038.

[7] Y. Ding and M. Palmer. (2005) Machine translation using probabilistic synchronous dependency insertion grammars. In Proc. of ACL 2005, pp. 541—548.

[8] K. Chen, R. Wang, M. Utiyama, L. Liu, A. Tamura, E. Sumita, and T. Zhao. (2017) Neural machine translation with source dependency representation. In Proc. of EMNLP 2017, pp. 2846—2852.

[9] A. Eriguchi, Y. Tsuruoka, and K. Cho. (2017) Learning to parse and translate improves neural machine translation. In Proc. of ACL 2017 Short Papers, pp. 72—78.

[10] S. Wu, D. Zhang, Z. Zhang, N. Yang, M.

Li, and M. Zhou. (2018) Dependency-to-dependency neural machine translation. IEEE/ACM Trans. Audio, Speech and Lang, Vol. 26, No. 11, pp. 2132—2141.

[11] Y. Omote, A. Tamura, and T. Ninomiya. (2019) Dependency-Based Relative Positional Encoding for Transformer NMT. In Proc. of RANLP 2019.

[12] H. Deguchi, A. Tamura, and T. Ninomiya. (2019) Dependency-Based Self-Attention for Transofrmer NMT. In Proc. of RANLP 2019.

[13] K. He, X. Zhang, S. Ren, and J. Sun. (2016) Deep residual learning for image recognition. In Proc. of CVPR 2016, pp. 770—778.

[14] J. L. Ba, J. R. Kiros, and G. E. Hinton. (2016) Layer normalization. arXiv preprint arXiv:1607.06450.

[15] T. Nakazawa, M. Yaguchi, K. Uchimoto, M. Utiyama, E. Sumita, S. Kurohashi, and H. Isahara. (2016) ASPEC: Asian scientific paper excerpt corpus. In Proc. of LREC 2016, pp. 2204—2208.

[16] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. (2007) Moses: open source toolkit for statistical machine translation, In Proc. of ACL on Interactive Poster and Demonstration Sessions, pp. 177—180.

[17] G. Neubig, Y. Nakata, and S. Mori. (2011) Pointwise prediction for robust, adaptable Japanese morphological analysis. In Proc. of ACL 2011, pp. 529—533.

[18] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. (2014) The Stanford CoreNLP natural language processing toolkit. In Proc. of ACL System Demonstrations, pp. 55—60.

[19] R. Sennrich, B. Haddow, and A. Birch. (2016) Neural machine translation of rare words with subword units. In Proc. of ACL 2016, pp. 311—318.