修　　士　　論　　文

英日翻訳における
seq2seq と Transformer の
スワップモデルを利用した比較

A Comparison of Seq2seq and Transformer on En-Ja Translation Using Their Swap Models

東京大学大学院 情報理工学系研究科
電子情報学専攻

氏　名　　48-176427 根石 将人

指導教員　　吉永 直樹 准教授

提　出　日　　平成 31 年 1 月 31 日

概 要

技術の発達により人や情報が激しく行き交う中で，より多くの人が，より多くの情報にアクセスするためには，言葉の壁は大きな問題である．機械に言葉を翻訳させる機械翻訳は，これに対する解決方法の一つで，言語知識の無い人でも，その言語を介した情報を扱うことを可能にする．

深層学習を利用したニューラル機械翻訳（NMT）は，従来の句に基づく統計的機械翻訳を上回る翻訳精度を実現し，現在，多くの機械翻訳システムに採用されている．しかしながら，実用的には，翻訳支援用途において翻訳者を必須としていたり，近年普及してきたスマートフォンのアプリの用途においても学習データ量の不足が問題となっているなど，未だ発展途上の技術である．

NMT の登場以降，そのモデル構造についての研究が進められ，現在までに様々な構造を持つ NMT モデルが提案されている．しかしながら，多様化した NMT モデル間の比較は，標準的な評価指標に基づく翻訳精度による数値評価に留まっており，また，その翻訳精度の向上幅も減少傾向にある．実用的な機械翻訳の実現のためには更なる高精度化が必要であり，NMT モデル間の構造の差異がもたらす影響について分析することは，今後のモデル改善のために重要な課題である．

本研究では，NMT モデルの改善を目的として，NMT 登場初期から広く用いられている再帰型ニューラルネットワーク に基づく NMT モデルと，現在主流となりつつある自己注意機構に基づく NMT モデルという 2 種類の NMT モデルに注目し，比較実験を通して，これらの構造の差異による影響を明らかにする．また，比較実験においては，2 種類の NMT モデルが共にエンコーダとデコーダの 2 つの構成素から成り立つことを利用し，エンコーダとデコーダを交換したモデル（スワップモデル）も比較対象として加えることで，より詳細な比較分析を行う．

謝辞

修士課程の２年間に渡って，常に親身で的確な指導を頂いた吉永直樹准教授に深く感謝致します．これ以上ない研究室環境を提供して下さった喜連川優教授に心から尊敬と感謝を申し上げます．適切で有益な助言と共に私の研究を支えて下さった豊田正史教授に厚く感謝申し上げます．日頃研究室でお世話になりました先生方，秘書の皆様に深く感謝申し上げます．肩を並べて研究に取り組み，同じ時を過ごした先輩，同期，後輩の皆様に感謝申し上げます．出会いから今まで，継続的に親交を深めてきた友人達に感謝申し上げます．最後に，今日この日まで辛抱強く，暖かく見守って下さっている父，母，姉に感謝の意を表します．　　2019 年 1 月 31 日

目 次

図 目 次

表 目 次

第1章　はじめに

1.1　機械翻訳の需要と普及

旧約聖書のバベルの塔の話に記されるように，言葉の壁は，古来からコミュニケーションにおける大きな障害として認識されてきた．技術の発達により，人の移動，更には情報の移動が激しさを増す中，対面での人とのコミュニケーションだけでなく，インターネット上のテキストデータなどの活用において，情報獲得及び情報発信における言葉の壁の解決，つまり，知識のない言語を通した情報理解の重要性が高まっている．

機械翻訳は，言葉の壁の解決方法の一つで，機械に言葉を翻訳させることで，言語知識の無い人でも，その言語で書かれた情報を理解可能にするという手法である．機械翻訳のアイディアは，1947 年に Warren Weaver と Nobert Wiener との手紙のやり取りの中で世界で初めて議論され，それから 50 年以上が経過した今，機械翻訳は手法を進化させながら，実世界での実用段階に達しつつある．

深層学習を利用したニューラル機械翻訳 (Neural Machine Translation; NMT) は，現在最も広く用いられている機械翻訳の手法である．2014 年に Cho ら [1], Sutskever ら [2] が提案して以来, それまで最良であった句構造に基づく統計的機械翻訳 (Phrase-Based Statistical Machine Translation; PBSMT) を上回る性能を実現し，現在，多くの機械翻訳システムに採用されている．しかしながら，そのような機械翻訳の手法の進化による翻訳性能の向上がありつつも，専門の翻訳家による翻訳ほどの高品質な翻訳は実現が難しく，また，場合によっては言語知識が無い人でも気づくような誤りをする場合などもあり，機械翻訳は，実用的には未だに発展途上であるといえる．

機械翻訳の実用例としては，まず，翻訳支援が挙げられる．翻訳支援において機械翻訳による翻訳は，翻訳者が後編集（ポストエディット）を行うための草稿として用いられる．ポストエディットにおける機械翻訳は，あくまで翻訳者の支援に留まっており，逆に捉えれば，機械翻訳による翻訳が，実用には未だ不十分な精度であることを示している．

また，翻訳者の支援に留まらず，Web 上で無料で利用可能な Google 翻訳[1]や Bing Microsoft Translator [2] は現在広く一般に利用されており，近年のスマートフォンの普及により，機械翻訳アプリの開発及び普及も進んでいる．2018 年の平昌オリンピックでは，通訳・翻訳アプリ「Genie Talk」が大会の公式アプリとして採用され[3]，また日本でも，情報通信研究機構 (National Institute of Information and Communications Technology; NICT) が音声翻訳アプリ「VoiceTra [4]」を無料で公開しており，2020 年の東京オリンピックに向けて，NICT は総務省と共同で，機械翻訳の更なる高精度化のために大規模な翻訳データの収集を始めている[5]．国の行政機関による翻訳データの収集は，機械翻訳が実用段階に至っている，もしくは至りつつあることを示すと同時に，依然，高精度化のための研究開発の余地があることも示している．

1.2 本研究の目的と得られた知見

2014 年の NMT の登場以降，NMT モデルの構造についての研究が進められ，現在までに様々な構造を持つ NMT モデルが提案されている．PBSMT と NMT の比較研究は多く，翻訳精度を始め，殆どの観点において PBSMT に対する NMT の優位性が確認されている一方で，多様化した NMT モデル間の比較は，標準的な評価指標 (BLEU スコア [3]) に基づく翻訳精度の数値評価に留まっており，またその翻訳精度の向上幅も減少傾向にある．しかしながら，1.1 で述べたように，実用的な機械

[1] https://translate.google.com/
[2] https://www.bing.com/translator
[3] https://en.yna.co.kr/view/AEN20180129007000320
[4] http://voicetra.nict.go.jp/
[5] http://www.soumu.go.jp/menu_news/s-news/01tsushin03_02000220.html

翻訳の実現のためには更なる高精度化が求められており，翻訳精度においておよそ拮抗する NMT モデル間の構造の差異がもたらす影響について分析することは，今後のモデル改善のために重要な課題である．

そこで，本研究では，今後のモデル改善による翻訳精度の向上を目的として，NMT 登場初期から広く用いられている再帰型ニューラルネットワーク (Recurrent Neural Network; RNN) に基づく NMT モデル（本研究ではこれに Luong ら [4] のモデルを採用し，以降では特にこれを seq2seq と呼ぶ）と，現在主流となりつつある自己注意機構に基づく NMT モデル (Transformer) [5] という 2 種類の NMT モデルに注目し，比較実験を通して，これらの構造の差異による影響を明らかにする．また，比較実験においては，Chen ら [6] の研究を参考に，seq2seq と Transformer が共にエンコーダとデコーダの 2 つの構成素から成り立つことを利用し，エンコーダとデコーダを交換した 2 種類のモデル（スワップモデル）も比較対象として加え，エンコーダ・デコーダレベルでの詳細な比較分析を行う．

実験では，seq2seq，Transformer，そしてそれらの 2 つのスワップモデルの合計 4 種類の NMT モデルを対象に，3 種類のデータセットによる英日翻訳タスクを通して，比較及び分析を行う．具体的に，以下に挙げる観点について実験を行う．

1. 自動評価による翻訳結果の比較
2. 翻訳結果に基づく比較
3. 学習データ量と精度の関係
4. ドメイン外データセットでの精度
5. 翻訳原文の文長と精度の関係

以上の実験から得られた知見を以下にまとめる．なお，数字は，上記の実験と対応していない．

1. エンコーダとデコーダの組み合わせにより，モデルの機能（得意とする問題）は変わる．

2. Transformer のエンコーダ・デコーダの影響力は強く，2 つのスワップモデルは共に，翻訳結果及び学習データ量に対する翻訳性能において，seq2seq よりも Transformer に類似する．

3. Transformer のエンコーダ・デコーダどちらかのみを NMT モデルに含めることで学習が効率化され，少ない学習データでの翻訳精度が向上する．

4. ビームサーチはデコーダと関連が強く，Transformer のデコーダはビームサイズを大きくした時に精度の低下が少ない．

5. ドメイン外の翻訳についてははどのモデルにも優位性はなく，モデル構造以外でのアプローチが必要である．

6. 長文の翻訳ではエンコーダの影響が大きく，特に学習データが少ない場合は seq2seq のエンコーダが優れ，学習データが十分にある場合は Transformer のエンコーダが優れる．

7. 出力文長の制御ではデコーダの影響が大きく，seq2seq のエンコーダは，不適切な文を出力しつつも，Transformer のデコーダよりも適切な長さの文を出力する．

1.3 本論文の構成

以降の本論文の構成は以下の通りである．

第 2 章 本研究と関連する，NMT と SMT の比較研究，多様な構造の NMT モデルの研究，そして NMT モデルの分析研究について述べる．

第 3 章 本研究の前提となる基礎知識について述べる．

第 4 章 本研究の分析が基づく seq2seq と Transformer の構造の違い，また比較に導入したスワップモデルについて述べる．

第 5 章　本研究の主な目的である seq2seq と Tranformer，およびそれらのスワップモデルの比較のための実験および分析を行う．

第 6 章　全体のまとめと今後の課題について述べる．

第2章　関連研究

本章では，1.2 で触れた本研究を位置づける関連研究について，その概要を述べる．

2.1　NMT と SMT の比較

NMT の登場以来，従来主流であった統計的機械翻訳 (Statistical Machine Translation; SMT) は NMT との比較対象となっている．

特に NMT と SMT の違いを分析した研究としては，Bentivogli ら [7]，Toral ら [8]，Koehn ら [9] の研究が挙げられる．

Bentivogli ら [7] は，NMT [10] と 3 種類の SMT（句と構文に基づくモデル [11]，階層的な句に基づくモデル [12]，句に基づくモデル [13] ）を対象に，文長毎や文書単位での翻訳精度の比較に加え，形態，語彙，語順に関しての誤り分析を行い，全てにおいて NMT が優れていることを示した．また同時に，NMT について，やや長文に弱い傾向があること，そして深い言語理解が必須な語彙の語順的な誤りがあることを示した．

Toral ら [8] は，9 種類の原言語･目的言語のペアに対して WMT16 (First Conference on Machine Translation) に提出された中で最も精度が高い NMT と PBSMT を用いて比較を行い，NMT と PBSMT の翻訳結果が大きく異なることを始め，NMT が PBSMT に比べ，より流暢な翻訳を出力すること，より語順の変更が多いこと，そして，語尾変化や語順の誤りがより少ないことを示した．また，Bentivogli らとやや異なる分析結果として，NMT が，ある文長を境としてそれを超える長文の翻訳において PBSMT に劣ること，また語彙的な誤りについては PBSMT と大きな違いはないことを示した．

Koehn ら [9] は，NMT [14] と PBSMT [15] の比較を行い，NMT について，学習データのドメイン外の翻訳において性能が落ちること，学習データが少ない場合に PBSMT に劣ること，一部の語尾変化由来の低頻度語等に弱いこと，長文に弱いこと，ビームサーチにおいて小さいビームサイズでしか精度が向上しないことを示した．

その他にも，対訳データの量が限られている英語とアイルランド語の翻訳における SMT と NMT の比較をした研究 [16] や様々なドメイン間での比較をした研究 [17] なども挙げられる．

これらの研究によって，SMT に対する NMT の優位性がおよそ決定的なものと示されると同時に，SMT との比較において発見された NMT が解決できていない問題も示されている．

2.2 NMT モデルの構造の多様化

最初に登場した NMT モデルは，再帰型ニューラルネットワーク (Recurrent Neural Network; RNN) に基づく NMT モデルである．当初はエンコーダとデコーダの 2 つの RNN から構成される単純な構造 [1, 2] であったが，後に，デコーダの内部にエンコーダの出力を利用するための注意機構 [4, 14] が提案され，旧来の PBSMT を上回る翻訳精度を実現した．2.1 で述べた NMT と SMT の比較研究において，後者の注意機構付きの RNN に基づく NMT モデルが NMT として採用されている．（seq2seq という表現は，現時点ではこの注意機構付きの RNN に基づく NMT モデルを指す表現であるが，1.2 で述べたように，本研究では，この中の Luong ら [4] のモデルを比較実験に採用するため，以降では特にこれを seq2seq と呼ぶ．）

言語の特徴をより上手く捉えること等を目的として，注意機構付きの RNN に基づく NMT モデルの構造に変更を加える研究は非常に多く，エンコーダに改変を加えた研究 [18–20]，そしてデコーダに改変を加えた研究 [18, 21–24] などに大分することができる．

RNN には系列処理に前の時刻の計算結果を必要とするという処理速度上のボトルネックがある．Gehring ら [25] はこれに注目し，RNN の代わりに並列処理による

高速化が可能な畳み込みニューラルネットワーク (Convolutional Neural Network; CNN) を用いた NMT モデル (ConvS2S) を提案した．ConvS2S はその狙い通り，Wu ら [26] による大規模な注意機構付きの RNN に基づく NMT モデルと比較して，推論時においておよそ 10 倍以上の高速化を達成しただけでなく，モデルがもつ階層構造により，階層的に単語間関係を捉えることでより高い翻訳精度を達成した．

Vaswani ら [5] は，RNN や CNN に基づくこれまでの NMT モデルを複雑だとみなし，注意機構のみに基づくシンプルな NMT モデル (Transformer) を提案した．Transformer は RNN や CNN の代わりとして自己注意層を利用しており，CNN と同じく並列処理による高速化が可能である上に，CNN のカーネルサイズによる参照先の制限は無いため，各単語間を直接的に処理することで ConvS2S を更に上回る翻訳精度を達成した．

Chen ら [6] は，さらなる翻訳精度の向上を目指し，これまでの NMT を組み合わせたモデル (RNMT+) を提案した．RNMT+は実質的には seq2seq と Transformer を組み合わせたモデルになっており，RNN を内部に持つことから処理速度の高速化が難しいが，学習時にモデルを並列に用意することで Transformer 並の処理速度を実現し，また，翻訳精度についても，これまでのモデルを上回った．

2.3 NMT モデルの分析と比較

上記のように，現在，NMT モデルの構造の多様化が進んでいるが，しかしながら一方では，それらのモデル間の比較は自動評価指標 (BLEU スコア) に基づく数値評価に留まっており，また，精度の向上幅も減少傾向にある．そのような状況の中で，NMT モデルを分析，また NMT モデル同士の比較をする研究が行われている．

Ding ら [27] は，注意機構付きの RNN に基づく NMT モデル [14] を対象として，可視化を通した NMT モデルの解釈を行い，この，モデルの注意機構だけでは原文と翻訳文の単語の対応を捉えきれないことや，RNN の層によって扱う情報が異なること示した．一方で，Lakew ら [28] は RNN に基づく NMT モデル [2] と Transformer の 2 つのモデルについて，通常の 1 言語対翻訳だけでなく，多言語翻訳，さらに多

言語翻訳において目的の言語対の対訳データを与えないゼロショット翻訳の各タスクにおいて比較し，翻訳精度だけでなく，詳細な誤り分析においても Transformer が優れていることを示している．

第3章　基礎知識

3.1　ニューラル機械翻訳モデル

本節では，Luong らによる注意機構付きの RNN に基づく NMT モデルである seq2seq [4] と，自己注意機構に基づく NMT モデルである Transformer [5] について，それぞれのエンコーダとデコーダの構造及び両モデルの違いについて述べる．なお本研究では，RNN に基づく NMT モデルには，Bahdanau らのモデルと Luong らのモデルの 2 つを比較する事前実験を行い，より精度の高かった Luong らのモデルを採用した．

以降の説明の参考として，一部簡略化した seq2seq と Transformer のエンコーダとデコーダについてのモデル構造の概要図を図 3.1 に示す．

3.1.1　Seq2seq

エンコーダは埋め込み層と RNN 層から構成され, 原言語のトークン列 $\mathbf{x} = (x_1, ..., x_M)$ を入力とし，エンコーダの各時刻の隠れ状態 $\mathbf{h} = (h_1, ..., h_M)$ 及び，RNN の最終隠れ状態 $h_{final} = h_M$ を出力する．各時刻の隠れ状態 h_i は一時刻前の出力 h_{i-1} と現時刻の入力 x_i から，以下の式で計算される．

$$h_i = \mathrm{RNN}(h_{i-1}, \mathrm{Emb}(x_i)) \tag{3.1}$$

ここで，RNN は RNN 層，Emb は埋め込み層である．

RNN としては，その一種である LSTM [29] や GRU [30] 等が用いられる．エンコーダには順方向の RNN と逆方向の RNN を組み合わせた双方向 RNN が使われる

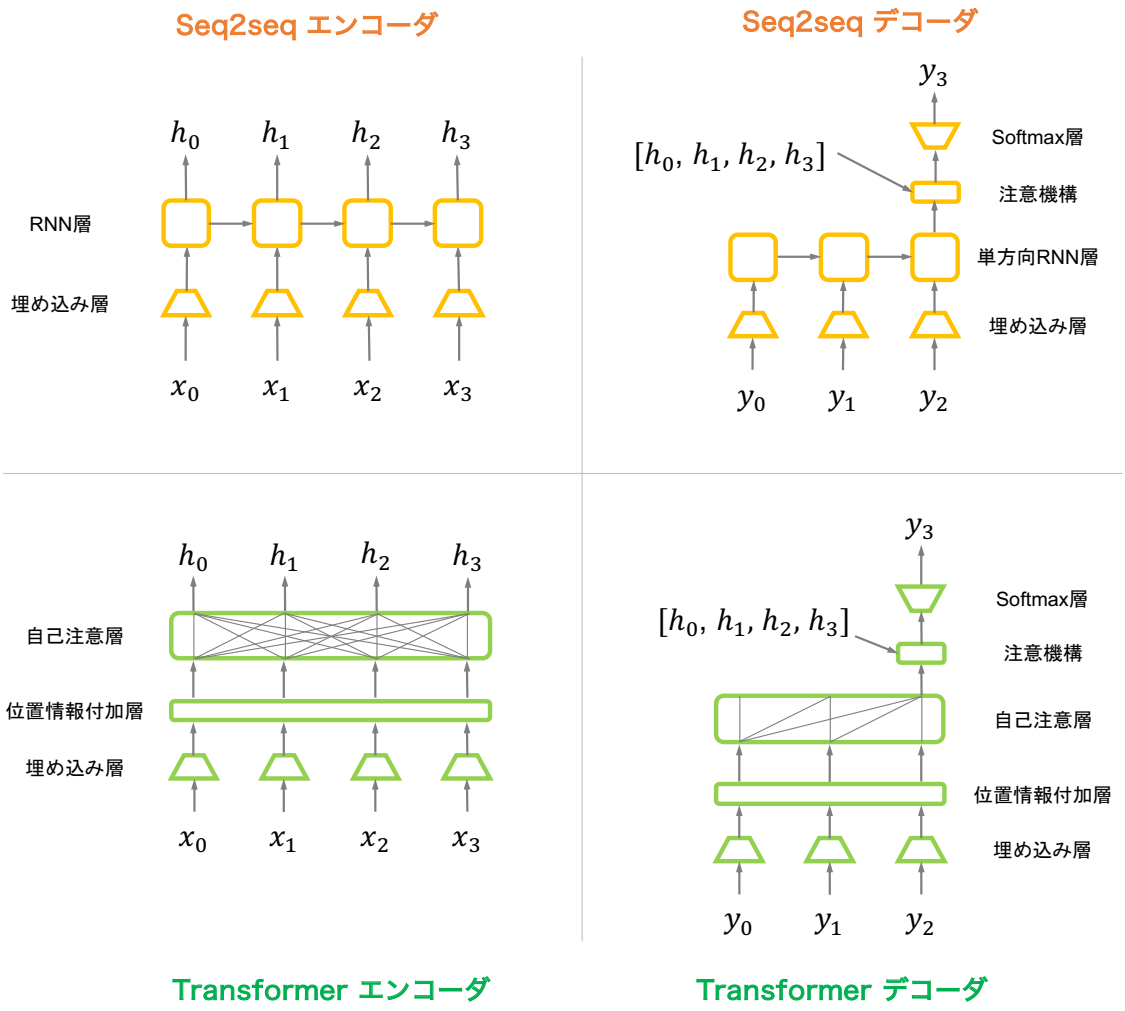

図 3.1: seq2seq と Transformer のモデル構造の概要図

ことが多く，また，RNN を複数重ねた多層 RNN 層が使われることが多い．この場合，RNN の最終隠れ状態 h_{final} は，各層及び双方向の全ての RNN の最終状態を連結したものを用い，またエンコーダの各時刻の隠れ状態 $\mathbf{h}$ としては，最終層の隠れ状態のみがデコーダへと渡される．なお，図 3.1 では h_{final} は省略してある．

デコーダは，エンコーダの各時刻の隠れ状態 $\mathbf{h}$ 及び，RNN の最終隠れ状態 h_{final} を入力とし，目的言語のトークン列 $\mathbf{y} = (y_1, y_2, ..., y_N)$ を出力する．デコーダの RNN

の隠れ状態はエンコーダの RNN の最終隠れ状態 h_{final} を用いて初期化し，各出力トークン y_i の確率分布は以下の式で計算される．

$$h_i = \mathrm{RNN}(h_{i-1}, [\mathrm{Emb}(y_{i-1}); \tilde{h}_{i-1}]) \tag{3.2}$$

$$c_i = \mathrm{Attention}(h_i, \mathbf{h}) \tag{3.3}$$

$$\tilde{h}_i = \tanh(W_c[c_i; h_i]) \tag{3.4}$$

$$p(y_i|y_{1:i-1}, \mathbf{h}) = \mathrm{softmax}(W_s\tilde{h}_i + b_s) \tag{3.5}$$

ここで，文脈ベクトル c_i はエンコーダの隠れ状態の重み付け和で，Attention$(\cdot)$ はその重みを決める関数である．$\tilde{h}_i$ は Input-feeding と呼ばれる機構のためのもので，デコーダの RNN の隠れ状態に加えて，Attention$(\cdot)$ 以降の計算結果も次の時刻へ渡す役割を持つ．

3.1.2 Transformer

Vaswani ら [5] が提案した Transformer は，エンコーダ，デコーダともに，埋め込み層と位置情報を付加する層を先頭に一層だけ持ち，以降は 6 層のサブレイヤーを組み合わせた層を，直列に複数重ねた構造となっている．

エンコーダは原言語のトークン列 $\mathbf{x} = (x_1, x_2, ..., x_M)$ を入力とし，同系列長のエンコーダの出力 $\mathbf{s} = (s_1, s_2, ..., s_M)$ を出力する．エンコーダの一つの層は，自己注意層とフィードフォワード層の 2 つのサブレイヤーで構成され，また各サブレイヤーの直後には，その層への入力を出力と足し合わせる Residual Connection 層，及び層の出力を正規化する Layer Normalizaion 層が付随する．（以降この 2 つの連続する層をまとめて RC-LN 層と呼ぶ．）自己注意層は，一つのベクトルをいくつかのベクトルへ分解して別々に処理を行う Multi-head という仕組みを採用している．以上

の演算は以下の式で表される．

$$\mathbf{x}_{pos} = \mathrm{PosEnc}(\mathrm{Emb}(\mathbf{x})) \tag{3.6}$$

$$\mathbf{x}_{self\_attn} = \mathrm{MultiHeadAttention}(\mathbf{x}_{pos}, \mathbf{x}_{pos}) \tag{3.7}$$

$$\mathbf{x}_{sa\_norm} = \mathrm{LayerNorm}(\mathbf{x}_{self\_attn} + \mathbf{x}_{pos}) \tag{3.8}$$

$$\mathbf{x}_{ff} = W_2 \max(0, W_1\mathbf{x}_{norm1} + b_1) + b_2 \tag{3.9}$$

$$\mathbf{x}_{ff\_norm} = \mathrm{LayerNorm}(\mathbf{x}_{ff} + \mathbf{x}_{norm1}) \tag{3.10}$$

ここで，PosEnc(·) は位置情報の付加層，MultiHeadAttenion(·) は Multi-head の仕組みを持つ注意層，LayerNorm(·) は Layer Normalizaion 層である．式（3.7）から式（3.10）までが直列に層を重ねる際の一層に相当し，最終層の $\mathbf{x}_{ff\_norm}$ がエンコーダの出力 $\mathbf{s}$ となる．

デコーダは，エンコーダの出力 $\mathbf{s}$ とその時点で出力済みの目的言語のトークン列 $y_{1:i-1}$ を入力とし，一時刻シフトしたトークン列 $y_{2:i}$ を出力する．デコーダの一つの層は，自己注意層とフィードフォワード層の間に，エンコーダの出力への注意層を追加した 3 つのサブレイヤーで構成され，エンコーダと同様に各サブレイヤーの直後には，RC-LN 層が付随する．ただし，デコーダの自己注意層は，入力系列に対して，過去のみに注意を張るようマスクがされ，エンコーダとはやや異なる仕組みになっている．また，出力トークンの確率分布の計算には，seq2seq と同様に，Softmax 層を使用する．エンコーダには存在しないエンコーダの出力への注意層とその直後の RC-LN 層，及び Softmax 層は次の式で計算される．

$$\mathbf{y}_{attn} = \mathrm{MultiHeadAttention}(\mathbf{y}_{sa\_norm}, \mathbf{s}) \tag{3.11}$$

$$\mathbf{y}_{at\_norm} = \mathrm{LayerNorm}(\mathbf{y}_{attn} + \mathbf{y}_{sa\_norm}) \tag{3.12}$$

$$\mathbf{p}(y_{2:i}|y_{1:i-1}, \mathbf{s}) = \mathrm{softmax}(W_s\mathbf{y}_{final} + b_s) \tag{3.13}$$

ここで，$\mathbf{y}_{sa\_norm}$ は自己注意層に付随する RC-LN 層の出力で，$\mathbf{y}_{at\_norm}$ は続くフィードフォワード層及び付随する RC-LN 層の入力となる．また，$\mathbf{y}_{final}$ は最終層の $\mathbf{y}_{ff\_norm}$ である．

第4章　本研究の分析の観点

4.1　構造の違い

seq2seq と Transformer の違いの一つは，入力系列に対する位置情報の扱いである．seq2seq は RNN を用いているため，系列データを逐次的に RNN に入力することにより，相対的な位置関係を扱っている．一方で，Transformer は位置情報の付加層を独立に設けており，絶対的な位置に基づいて定数を入力に加算している．この点において，相対位置を扱う seq2seq の方が，絶対位置を扱う Transformer に比べ，汎用的な性能を持つことが予想される．そのため，学習データが少ない長文の翻訳においては，seq2seq が優れることが予想される．

注意機構の構造も大きく異なる．seq2seq には自己注意機構は存在せず，入力系列データにおける前後の情報の処理は，RNN の隠れ状態の遷移にのみ依存し，入力系列の各要素間の距離が，そのまま処理における距離となる．また，デコーダからエンコーダの出力への注意機構は，一時刻の処理に付き，一度きりかつ一つの文脈ベクトルのみを扱う．それに対して Transformer は，自己注意機構によりエンコーダについては入力全体，デコーダについては過去の出力系列全体へのアクセスが可能であり，系列の要素間の距離が非常に近い．また，デコーダからエンコーダの出力への注意機構は一時刻あたり，デコーダの層と同じ数だけエンコーダの出力を参照する．これらの点においては，seq2seq は扱える情報が局所的に縛られている一方で，Transformer は縛りのない大域的なアクセスが可能である優位性があり，長文において離れた単語間の関係を捉えることや，また，深層学習の学習（パラメタの更新）における誤差伝搬において有利であると考えられる．

4.2 Seq2seq-Transformer スワップモデル

本研究では，seq2seq と Transformer についてより詳細な分析を行うために，Chen ら [6] が導入した互いにエンコーダとデコーダを交換した 2 つのスワップモデルを比較対象として含める．

Chen ら [6] は NMT モデルの性能向上を目的として，NMT モデルを組み合わせるハイブリッドモデルの一つとして異なるモデルのエンコーダとデコーダを組み合わせたモデルを導入した．それに対して．本研究では，エンコーダとデコーダを交換することで，モデル間の違いをエンコーダ，デコーダレベルで分析することを目的とした．

seq2seq のエンコーダと Transformer のデコーダを組み合わせたモデルは，エンコーダの出力 $\mathbf{h}$ と RNN の最終隠れ状態 h_{final} のうち，前者のみをデコーダの入力とする．

また，Transformer のエンコーダと seq2seq のデコーダを組み合わせたモデルは，Transformer のエンコーダの出力 $\mathbf{h}$ をデコーダの入力の一つとするが，この場合，デコーダの RNN の隠れ状態の初期化を行う入力が存在しない．しかしながら，5.2.1 の予備実験で示すように，seq2seq のデコーダは RNN の隠れ状態を全て 0 で初期化しても，エンコーダの最終隠れ状態で初期化を行った場合と同等の性能を発揮する．よって，このスワップモデルではデコーダの RNN の隠れ状態は全て 0 で初期化した．

第5章　実験

本章では，seq2seq と Transformer，及び互いにエンコーダとデコーダを交換した 2 つのスワップモデルの合計 4 つのモデルに対して，英日翻訳タスクを通した比較実験を行う．なお，以降では便宜的に，エンコーダとデコーダの弁別に基づいて，seq2seq モデルを SS，seq2seq モデルのエンコーダと Transformer モデルのデコーダを組み合わせたスワップモデルを ST，Transformer モデルのエンコーダと seq2seq モデルのデコーダを組み合わせたスワップモデルを TS，Transformer モデルを TT と表記する．

行う実験を目的と共に以下にまとめる．

1. **自動評価による翻訳結果の比較**
   各モデルの基本的な翻訳精度を確認するために，機械翻訳タスクの評価で最も標準的な自動評価手法を用いた翻訳結果の評価を行う．

2. **翻訳結果に基づく比較**
   モデル間の類似度を測るために，各モデルの翻訳結果同士の類似度を複数の評価手法を用いて評価する．

3. **学習データ量と翻訳精度の関係**
   学習データ量と翻訳精度の関係を明らかにするために，4 種類のモデル全てに対して学習データ量を変化させた学習を行い，評価する．

4. **ビームサーチにおける各モデルの精度の変化の比較**
   ビームサーチによるデコードを行った場合の各モデルの違いを明らかにするために，ビームサーチにおける精度の変化を比較する．

5. ドメイン外データセットでの精度
   モデルの違いと学習データのドメイン外の翻訳精度の関係を明らかにするために，各モデルに対して，学習データと異なるデータセットでの翻訳精度の評価を行う．

6. 翻訳原文の文長と翻訳精度の関係
   翻訳文の長さと翻訳精度の関係を明らかにするために，各モデルの翻訳結果を，原言語文の長さで分割して評価を行う．

7. 学習データの文長を制御した場合の翻訳精度の比較
   学習データの文長と翻訳精度の関係を明らかにするために，学習データを文長に基づいて 3 つに分割して各モデルの学習を行い，その翻訳精度を比較する．

5.1 実験設定

5.1.1 NMT モデル

実装には，PyTorch[1] (ver. 0.4.1) を用いた．seq2seq については，Hashimoto ら [31] を参考に，エンコーダには 3 層の双方向 LSMT を，デコーダには 1 層の単方向 LSTM を用い，各 LSTM の隠れ状態は 512 次元とした．Transformer については，Vaswani ら [5] の base model を採用し，エンコーダ，デコーダは共に 6 層とし，モデルサイズは 512 次元，フィードフォワード層は 2048 次元とした．

上記以外のパラメータは全て共通とし，埋め込み層は 512 次元，dropout の確率は 0.2 とした．最適化手法には Adam [32] を用い，初期学習率を 0.0001 とした．また，パラメータ更新時の勾配のノルムに制限を設け，最大値を 3.0 とした．ミニバッチサイズは基本的に 128 とし（データセットにより，一部異なる．5.1.2 参照），400,000 ステップの学習を行った．以降の実験では，400,000 ステップの学習の中で，10,000 ステップ毎に開発データを用いて自動評価手法である BLEU [3] による評価を行い，

[1] https://pytorch.org/

最も良かったステップのモデルを使用した．なお，Transformer について Vaswani ら [5] が採用している学習中の学習率の変更，及び正解データを調整するラベルスムージング [33] などの翻訳精度をさらに向上させる手法は，条件を揃えるために本研究では使用していない．

5.1.2 データセット

本研究では，英日翻訳タスクによる比較実験を行うにあたって，Asian Scientific Paper Excerpt Corpus (ASPEC) [34]，京都フリー翻訳タスク (KFTT) [35]，Japanese-English Subtitle Corpus (JESC) [36]，の 3 つのデータセットを用いた．以降ではそれぞれのデータセットの概要と，行った前処理の詳細を述べる．前処理後の学習データ数も含めた各データセットの内訳を表 5.1 にまとめる．

表 5.1: データセットの内訳

	学習データ数（オリジナル）	学習データ数（前処理後）	開発データ数	テストデータ数
ASPEC	3,008,500	1,314,495	1,790	1,812
KFTT	440,288	426,531	1,166	1,160
JESC	3,237,376	3,156,692	2,000	2,001

5.1.2.1 Asian Scientific Paper Excerpt Corpus

Asian Scientific Paper Excerpt Corpus (ASPEC) [34] は，科学技術論文の概要から収集した対訳文からなる大規模な対訳コーパスであり，3,008,500 文対の学習データ，1,790 文対の開発データ，1,812 文対のテストデータから構成される．

テキストデータの前処理は，次のように行った．まず，英語のデータは，Moses[2] (ver. 2.2.1) [15] のスクリプトを用いてトークン化及び Truecasing を行い，日本語の

[2] http://www.statmt.org/moses/

データは KyTea[3] (ver. 0.4.2) [37] を用いてトークン化を行った．以上の処理をしたデータに対して，学習データの対訳文のペアについて，どちらかの言語において文長 50 を超えるものを除外した．この結果，学習データは 1,783,817 文対となった．

続いて，SentencePiece[4] [38] を用いたサブワードへのトークン化を，両言語のデータを結合して行った．この時，分割アルゴリズムは unigram を選択し，語彙数は 16,000（両言語共有）とした．また，橋本ら [31] の前処理を参考に，学習データは先頭から 1,500,000 文対までを用い，サブワード化による文長の増加を考慮して，再び，どちらかの言語において文長 50 を超えるものを除外した．この結果，最終的な学習データは 1,314,495 文対となった．

5.1.2.2 京都フリー翻訳タスク

京都フリー翻訳タスク（KFTT） [35] は，京都関連の Wikipedia 記事を利用した対訳コーパスからなる翻訳タスクで，440,288 文対の学習データ，1,166 文対の開発データ，1,160 文対テストデータから構成される．

テキストデータの前処理は，ASPEC と同様に，まず，英語のデータは Moses を用いてトークン化及び Truecasing を行い，日本語のデータは KyTea を用いてトークン化を行った．ASPEC とは異なり，ここで文長による学習データの除外は行わず，以上の処理をしたデータをに対して，分割アルゴリズムは unigram，語彙数は 16,000（両言語共有）として，SentencePiece によるサブワードへのトークン化を，両言語のデータを結合して行った．

最後に，文長による学習データの除外を行うにあたって，文長制限を 50 に設定した場合，学習データが大幅に減少してしまうため，KFTT については，文長制限を 100 とし，同時に学習時のミニバッチサイズを 64 とした．このため，3 つのデータセットの内，KFTT のみミニバッチサイズが異なる．以上の処理により，最終的な学習データは 426,531 文対となった．

[3] http://www.phontron.com/kytea/
[4] https://github.com/google/sentencepiece

5.1.2.3 Japanese-English Subtitle Corpus

Japanese-English Subtitle Corpus (JESC) [36] は，インターネット上からクロールした映画と TV 番組の字幕データを元に作られた大規模な対訳コーパスで，他の 2 つのコーパスと比べて，口語表現が含まれている特徴がある．JESC は，3,237,376 文対の学習データ，2,000 文対の開発データ，2,001 文対のテストデータから構成される．

前処理はほぼ KFTT と同様である．英語のデータは Moses を用いてトークン化及び Truecasing を行い，日本語のデータは KyTea を用いてトークン化を行った．以上の処理をしたデータに対して，分割アルゴリズムは unigram，語彙数は 16,000（両言語共有）として，SentencePiece によるサブワードへのトークン化を，両言語のデータを結合して行った．

最後に，ASPEC と同様に，どちらかの言語において文長 50 を超えるものを除外した．この結果，学習データは 3,156,692 文対となった．

5.2 実験

5.2.1 自動評価による翻訳精度の比較

4 種類のモデルの各データセットにおける翻訳精度を確認するために，3 つのデータセットに対してそれぞれ NMT モデルの学習を行い，各モデルの翻訳精度を，自動評価手法である BLEU [3] と RIBES [39] を用いて評価した．この時，SentencePiece によるサブワード化されている出力に対して，サブワードマーカー及び単語区切りは削除し，再度の KyTea による分割（以降この処理を KyTea による再分割と呼ぶ）の後に評価を行った．なお，ASPEC データセットに対してのみ，予備実験として，SS モデルにおいてデコーダの RNN の初期隠れ状態を 0 で初期化したモデル，SS* を追加した．データセット毎の結果とそれぞれの BLEU スコアに対するブートストラップ検定の結果を，表 5.2〜5.7 に示す．

表 5.2: ASPEC における各モデルの BLEU スコアと RIBES スコア

	SS	ST	TS	TT	SS*
BLEU	*37.02*	**39.19**	37.74	38.65	36.94
RIBES	*82.01*	**83.33**	82.51	83.31	82.17

表 5.3: ASPEC における各モデルの BLEU スコアに対するブートストラップ検定の結果（サンプル数 : 10000）（p-value について">>": 0.01 以下，">": 0.05 以下，" ": 0.05 以上）

	SS	ST	TS	TT
SS				
ST	>>			
TS	>	<<		
TT	>>	<	>>	

まず，ASPEC を用いた予備実験であるが，表 5.2 に示すように，SS と SS*のスコアはおよそ同程度であり，また BLEU と RIBES においてそれぞれ優劣が逆転している．このことから，SS において，デコーダの RNN の初期隠れ状態を 0 で初期化しても，エンコーダの最終隠れ状態で初期化した場合と同等の性能を発揮することが分かる．

Vaswani ら [5] や Chen ら [6] によって，SS より TT が翻訳精度において上回ることが示されているが，ASPEC，KFTT についてはそれが再確認された．しかしながら，JESC における BLEU スコアでは SS が TT を上回る結果となった．RIBES スコアにおいては逆転しているため，JESC においては SS と TT は同等の翻訳精度といえる．

また，ST は ASPEC，JESC において最も良く，KFTT においても TT に次ぐ 2 番目に良いスコアであり，本研究で比較する 4 種類のモデルの中で最も優れた翻訳精度を出している．TT も ASPEC，KFTT においては高い精度を出してはいるが，JESC においては SS と同程度で，ST に劣る精度である．

表 5.4: KFTT における各モデルの BLEU スコアと RIBES スコア

	SS	ST	TS	TT
BLEU	*28.39*	30.20	29.08	**31.04**
RIBES	*75.93*	78.56	77.11	**78.75**

表 5.5: KFTT における各モデルの BLEU スコアに対するブートストラップ検定の結果（サンプル数 : 10000）（p-value について”>>”: 0.01 以下，”>”: 0.05 以下，” ”: 0.05 以上）

	SS	ST	TS	TT
SS				
ST	>>			
TS	>	<<		
TT	>>	<	>>	

JESC には口語表現が含まれており，逐語翻訳の問題としてはやや特異であり，実際ブートストラップ検定により，有意差が無いモデル同士のほうが多いことが示されているため，その他の 2 つのデータセットの結果から TT が SS に対して翻訳精度において優れるといえる．また，ST の翻訳精度が高く，これにより seq2seq のエンコーダについては，デコーダとの組み合わせにより Transformer のエンコーダより優れる場合があることが示された．

また，ASPEC データセットについて，4 種類のモデルを組み合わせたアンサンブルモデル [40] による翻訳も行った．その結果を表 5.8 に示す．

2 つのモデルのアンサンブルでは，(ST, TT) のモデルの組み合わせが最もスコアが高く，(SS, TS) が最も低い．これは，それぞれが単一モデルのスコアでの上位 2 つ同士，下位 2 つ同士の組み合わせである．3 つのモデルのアンサンブルについては，単一モデルでの下位 2 つのどちらかを含めない (SS, ST, TT)，(ST, TS, TT) が最もスコアが高く最もスコアが高い ST を含めない (SS, TS, TT) の組み合わせが最もスコアが低い．しかしながら，TT を含めない (SS, ST, TS) もそれらに比類する

表 5.6: JESC における各モデルの BLEU スコアと RIBES スコア

	SS	ST	TS	TT
BLEU	16.60	**17.24**	*16.14*	16.36
RIBES	52.62	**54.03**	*52.38*	53.53

表 5.7: JESC における各モデルの BLEU スコアに対するブートストラップ検定の結果（サンプル数：10000）（p-value について">>": 0.01 以下，">": 0.05 以下，" ": 0.05 以上）

	SS	ST	TS	TT
SS				
ST	$\sim$			
TS	$\sim$	$<<$		
TT	$\sim$	$<$	$\sim$	

スコアを出しており，単純に最良モデルである ST を含めるか否かが影響していると考えられる．最後に，全てを用いたアンサンブルモデルであるが，3 つのモデルのアンサンブルモデルと同程度のスコアであり，アンサンブルするモデル数を増やせば必ずしもスコアが向上するとは限らないことが分かる．

Imamura ら [41] は，小規模なデータセットにおいて，最大 16 個までの同一構造のモデルのアンサンブルを行い，その効果が飽和しないかったことを報告している．本実験では，大規模なデータセットを用い，最大 4 つまでの，構造が異なり，翻訳精度も異なるモデルのアンサンブルを行ったという違いがあるため，先行研究との比較は難しい．全てのモデルのアンサンブルはやや翻訳精度が減少する結果とはなったが，モデルをアンサンブルする際には，単一で精度の高いモデルを含めることが特に効果的であること，また，精度に差があるモデル同士のアンサンブルもおおむね精度向上に貢献することが明らかになった．

表 5.8: ASPEC における 4 種類のモデルを組み合わせたアンサンブルモデルの BLEU スコアと RIBES スコア

SS	ST	TS	TT	BLEU	RIBES
✓	✓			40.05	83.61
✓		✓		*39.23*	*83.22*
✓			✓	39.44	83.42
	✓	✓		40.16	83.69
	✓		✓	**40.19**	**84.00**
		✓	✓	39.35	83.52
✓	✓	✓		40.54	83.71
✓	✓		✓	**40.72**	84.01
✓		✓	✓	*39.83*	*83.61*
	✓	✓	✓	40.47	**84.02**
✓	✓	✓	✓	40.65	83.95

5.2.2 翻訳結果に基づく比較

本実験では，各モデルの類似度を，各モデルの翻訳結果の文の相互類似度を通して比較する．4 つのモデルの相互類似度を考えると，直感的には，エンコーダもしくはデコーダを共有する，SS と ST，SS と TS，また ST と TT，TS と TT の 4 つのペアが類似し，逆にモデルの構成素を一切共有しない SS と TT，及び ST と TS が最も類似しないと予想される．

翻訳文の相互類似度の評価には，BLEU，文字単位の編集距離，トークン単位の編集距離，平均単語ベクトルによる文ベクトルの距離，単語ベクトルに基づく Word Mover's Distance [42] の 5 つの自動評価手法を用いる．なお，生のモデルの出力を評価するために，KyTea による再分割は行わず，SentencePiece によるトークンのままで比較を行った．また，単語ベクトルは，gensim (ver. 3.4.0) の word2vec をデフォルト設定（窓幅 5 の CBOW [43]）で使用し，日本語の学習データを用いて学習を行った．

全てのデータセットにおいて，文字単位の編集距離とトークン単位の編集距離におけるモデル間の類似度の順位は同じであったため，文字単位の編集距離を除く 4 つの評価手法の結果を，データセット毎に，表 5.9〜5.14 に示す．

表 5.9: ASPEC における　翻訳文間の BLEU スコア（左下），及びトークン単位編集距離の平均（右上）

	REF	SS	ST	TS	TT
REF		18.9	18.0	18.8	18.1
SS	37.61		12.5	*13.1*	12.6
ST	39.87	57.86		11.9	**10.6**
TS	38.23	*56.48*	59.23		10.7
TT	39.20	57.17	**63.54**	62.22	

表 5.10: ASPEC における　翻訳文間の文ベクトル距離の平均（左下），及び Word Mover's Distance の平均（右上）

	REF	SS	ST	TS	TT
REF		9.77	9.30	9.64	9.34
SS	2.93		6.36	*6.71*	6.39
ST	2.85	2.15		6.12	**5.41**
TS	2.93	*2.24*	2.12		5.55
TT	2.85	2.16	**1.92**	1.96	

表 5.11: KFTT における　翻訳文間の BLEU スコア（左下），及びトークン単位編集距離の平均（右上）

	REF	SS	ST	TS	TT
REF		19.51	18.17	19.42	18.20
SS	30.88		14.96	*16.16*	15.20
ST	32.96	42.93		14.71	**12.75**
TS	31.06	*40.79*	43.67		13.96
TT	33.80	42.89	**49.89**	46.84	

表 5.12: KFTT における　翻訳文間の文ベクトル距離の平均（左下），及び Word Mover's Distance の平均（右上）

	REF	SS	ST	TS	TT
REF		7.32	6.92	7.23	6.80
SS	2.82		5.81	*6.12*	5.78
ST	2.66	2.35		5.54	**4.96**
TS	2.75	*2.44*	2.22		5.24
TT	2.62	2.32	**2.00**	2.11	

表 5.13: JESC における　翻訳文間の BLEU スコア（左下），及びトークン単位編集距離の平均（右上）

	REF	SS	ST	TS	TT
REF		8.12	8.18	8.17	8.30
SS	17.07		5.39	5.31	*5.44*
ST	17.99	35.96		5.10	5.03
TS	16.38	*35.94*	38.15		**4.91**
TT	16.81	36.21	**41.38**	40.54	

ASPEC については 5 つの評価手法全てにおいて，2 モデル間の類似度の順位が同一となった．KFTT については，1 位から 4 位までは全ての評価手法で同順位となったが，5 位と 6 位については，BLEU，文字単位の編集距離，トークン単位の編集距離の 3 つの評価手法と，文ベクトルの距離，Word Mover's Distance の 2 つの評価手法で，SS と TT，SS と TS，が入れ替わる結果となった．また，JESC では，各モデルのテストデータでの BLEU スコア及び RIBES スコアが低いために各評価手法における類似度の差が小さく，類似度の順位もやや異なる結果となった．

4 つのモデルの相互類似度について，エンコーダもしくはデコーダを共有する SS と ST，SS と TS，また ST と TT，TS と TT の 4 つのペアが類似し，逆にモデルの構成素を一切共有しない SS と TT，及び ST と TS が最も類似しないと予想をしたが，以上で示された結果はやや異なっている．構成素を共有する ST と TT，また TS と TT については直感通り翻訳結果の類似度が高い結果となってはいるが，一方で，同

表 5.14: JESC における　翻訳文間の文ベクトル距離の平均（左下），及び Word Mover's Distance の平均（右上）

	REF	SS	ST	TS	TT
REF		11.33	11.23	11.49	11.33
SS	5.31		7.31	*7.39*	7.36
ST	5.30	3.62		6.91	**6.51**
TS	5.41	3.67	3.45		6.55
TT	5.33	*3.68*	**3.26**	3.30	

じく構成素を共有する SS と ST，及び SS と TS は類似度が低く，その内 SS と TS については，一切構成素を共有しない SS と TT よりも低い場合が非常に多い．また，SS はその他の 3 つのモデル全てと類似度が低い傾向があり，一方でその他の 3 つのモデル同士はそれぞれ類似度が高い傾向がある．

参考までに，ASPEC データセットにおいて，SS とその他の 3 つのモデルで翻訳に違いが見える例を表 5.15 に示す．この例では，原文の最後の "... and the network structure" の 等位接続詞 "and" が何と接続するかが問題である．正解は"hydrogen bond"であり，直前の"the water molecules"は不正解である．それぞれのモデルの翻訳結果を見ると，SS のみが"the water molecules"と接続してしまっており，"水分子とネットワーク構造"と誤って翻訳している．

以上の結果から，エンコーダ・デコーダの共有が必ずしもモデルの出力の相互類似度に寄与しないこと，言い換えれば，個別のエンコーダ・デコーダがモデルの出力に直接的には影響しないことが分かった．このことから，エンコーダ及びデコーダは，それぞれ独立して普遍的な機能を保持している訳ではなく，組み合わせにより，その機能を変えていると考えられる．

5.2.3　学習データ量に対する翻訳精度の比較

本実験では，学習データ量と翻訳精度の関係における各モデルの比較を行う．Transformer の注意機構によるネットワーク内の大域的なアクセスが学習を効率化

表 5.15: ASPEC における，SS が他の 3 つのモデルと異なる翻訳をしている例

原文	to begin with , various properties of water were explained on hydrogen bond in which the force works among the water molecules and the network structure .
参照訳	水の性質の多様性について, まず, 水分子同士の間に働く力である水素結合と, そのネットワーク構造について解説した.
SS	まず, 水分子とネットワーク構造の間に働く水素結合について, 水の諸性質を解説した.
ST	まず, 水分子間に力が働く水素結合とネットワーク構造について, 水の諸特性を説明した.
TS	まず, 水の種々の性質について, まず水分子間の力が作用する水素結合とネットワーク構造について解説した.
TT	まず, 水分子間で力が働く水素結合とネットワーク構造について, 水の諸特性を説明した..

させることで，少ない学習データにおいてもより高い精度が期待できると予想する．

データセットは ASPEC のみを対象として，学習に用いる学習データ量について，1,500,000 文対（1,314,495）を全て用いる場合のみでなく，先頭から 15,000 文対（14,888），50,000 文対（48,955），150,000 文対（143,444），500,000 文対（459,813）の場合を加えて，合計 5 つの場合に対して各モデルの学習を行い，その翻訳精度を比較する．なお，各文対数は SentencePiece トークンでの文長を 50 で制限する前であり，制限後の実際に学習に使用される文対数を括弧内に示した．

翻訳精度の評価には自動評価手法である BLEU を用い，また，モデルの出力である翻訳文に対しては，KyTea による再分割をしてから評価を行った．結果を図 5.1 に示す．

全ての学習データ量の場合に渡って，SS が最も翻訳精度が低く，また，他の 3 つのモデルと異なる振る舞いをしていることが分かる．Koehn ら [9] によって，学習データ量が少ない場合では，SS が PBSMT に劣ることが示されているが，本実験の結果から，学習データ量が少ない場合に，SS はその他の 3 つの NMT モデルにも劣ることが示された．また，モデル間の類似度について，5.2.2 の実験においても，SS

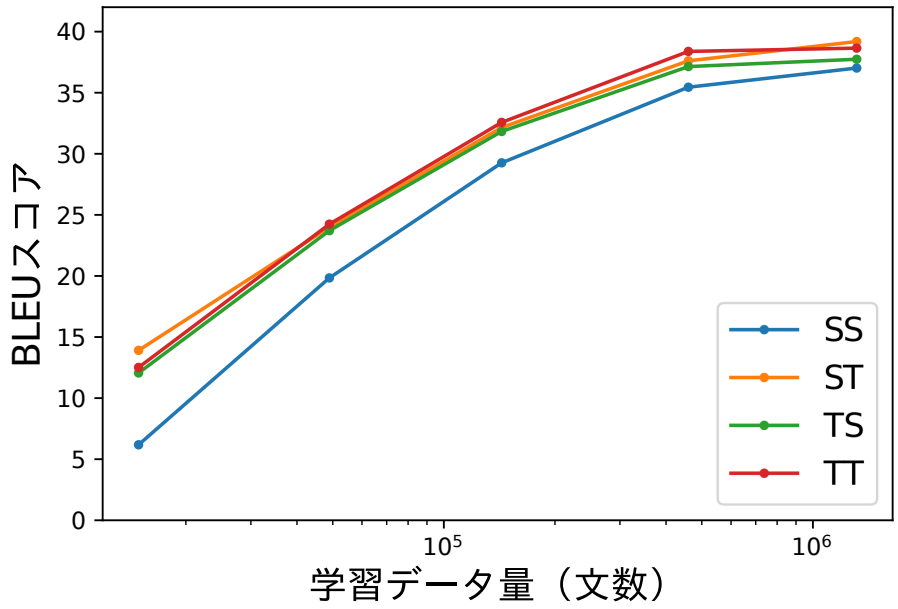

図 5.1: 学習データ量に対する各モデルの BLEU スコア

が他の 3 つのモデルと異なり，ST，TS，TT が互いに類似した結果を考慮すると，エンコーダ及びデコーダが，組み合わせによりその機能を変えていることがこの結果からも考えられる．

モデルの構造由来の違いとして，Transformer のエンコーダ，デコーダの注意機構の仕組みの影響が考えれられる．Transformer のエンコーダ，デコーダの注意機構はネットワーク内に大域的なアクセスを可能にしており，学習のための誤差伝搬に有利であると考えれられる．結果と照らし合わせると，Transformer のエンコーダ，デコーダのどちらかしらがモデルに含まれる場合には十分な学習の効率化がされていると考えられる．結果としては，SS のみが効率的な学習を行えず，4 種類のモデルの中で唯一異なる振る舞いを見せていると考えられる．

5.2.4 ビームサーチにおける各モデルの精度の変化の比較

ビームサーチによるデコードを行った場合の各モデルの違いを明らかにするために，ビームサーチにおける精度の変化を比較する．ビームサーチは基本的にエンコーダには関係がなく，デコーダのみに依存するため，4 種類のモデルの内，それぞれデコーダを共有するモデル同士が似た性質をもつと考えられる．

ASPEC データセットで学習したモデルに対して，ビームサーチによる翻訳を行う．開発データで最適なビーム幅を探索した後，そのビーム幅を用いてテストデータでの評価を行う．ビームサーチのアルゴリズムには Wu ら [26] のものを，ハイパーパラメータ α=1 として用いた．なお，seq2seq と Transformer のデコーダの構造の違いにより，Transformer のデコーダにそのまま応用することが出来なかった coverage penalty は seq2seq においても使用しなかった．開発データによる各モデルのビーム幅の探索結果を図 5.2 に示す．

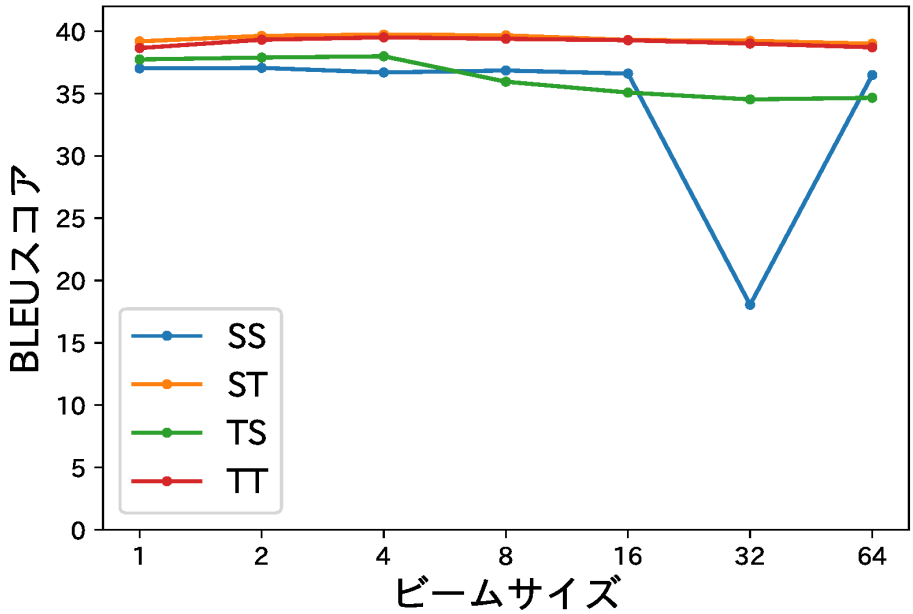

図 5.2: 開発データにおける各モデルのビーム幅による BLEU スコアの変化

SS がビーム幅 32 において不規則な挙動を示したが，その点を除けば，おおむね

デコーダを共有する同士がビーム幅の変化に対してほぼ同じスコアの推移をしている．また，Transformer のデコーダはビームサイズを大きくした時の精度の低下が少ないことが確認できる．この結果を利用してビームサーチを用いてテストデータにおける評価をした結果を表 5.16 に示す．全てのモデルにおいて翻訳精度の向上が確認できるが，SS のみスコアの増加量が小さい結果となった．

表 5.16: ASPEC における各モデルへのビームサーチの結果

	BLEU スコア	Δ	ビーム幅
SS	37.05	+0.03	8
ST	39.68	+0.49	4
TS	38.00	+0.26	4
TT	39.02	+0.37	4

5.2.5 ドメイン外データセットでの翻訳精度

モデルの違いがドメイン外データセットでの翻訳精度へ影響するかを調べるために，3 つのデータセットにおいて学習を行った 4 種類のモデルを，ドメイン外である，学習データと異なるデータセットを用いて評価を行った．モデルの構造由来の位置情報の扱いの違いや，注意機構によるネットワーク内のアクセスの違いは特に影響しないと予想される．

前処理の違いにより，各データセットごとに異なる語彙を使用しているため，テストデータを，モデルが学習したデータセットに対応させる必要がある．そのため，学習データのデータセットと同一の前処理を，それぞれのテストデータについて行った．具体的には，Moses のスクリプトによる Truecasing と，SentencePiece によるサブワードへのトークン化を，学習データのデータセットで使用した学習済みモデルを用いて，テストデータに対して改めて前処理を行った．また，KyTea による再分割を行った上で，BLEU を用いて翻訳精度を評価した結果を学習データのデータセット毎に，それぞれ表 5.17，表 5.18，表 5.19 に示す．

表 5.17: ASPEC データセットで学習したモデルの各データセットにおける BLEU スコア

	ASPEC			
	SS	ST	TS	TT
ASPEC	*37.02*	**39.19**	37.74	38.65
KFTT	7.08	**8.09**	*5.02*	7.99
JESC	1.81	**2.09**	*0.90*	2.43

表 5.18: KFTT データセットで学習したモデルの各データセットにおける BLEU スコア

	KFTT			
	SS	ST	TS	TT
ASPEC	*8.11*	9.75	8.48	**10.26**
KFTT	*28.39*	30.20	29.08	**31.04**
JESC	2.32	3.39	*2.29*	**3.70**

ドメインの違いの影響は大きく，どのモデルも，ドメイン外のデータセットでの翻訳精度は，学習データのデータセットでの翻訳精度から大幅に下がっている．また，JESC データセットで学習したモデルの KFTT での評価を除き，基本的にはどの場合においても，元のデータセットで最高精度のモデルが，ドメイン外でも最高精度を出している．しかしながら，特定のモデルがドメイン外において翻訳精度の傾向の大幅な変化は認められず，NMT モデルの構造はドメイン外での翻訳精度に影響しないことが分かる．

本実験のドメイン外での評価おける低い翻訳精度の原因の一つとして，サブワードを使用している事による，語彙の違いが考えられる．そこで，学習データのデータセットとテストデータの組み合わせにおける，学習データのデータセットの語彙へと分割したテストデータの平均文長を表 5.20 に示す．この表から，JESC のデータセットに合わせてテストデータの分割を行うと，他のデータセットに合わせた場

表 5.19: JESC データセットで学習したモデルの各データセットにおける BLEU スコア

	JESC			
	SS	ST	TS	TT
ASPEC	4.25	**4.68**	*3.11*	3.43
KFTT	**3.75**	3.68	3.20	*2.48*
JESC	16.60	**17.24**	*16.14*	16.36

合に比べ，文長が長い，つまりトークン数が多いことが分かる．同一の文についてトークン数が多いということは，サブワードへのトークン化において，単語がより細かく分割されているということである．JESC のテストデータの文を，各データセットに合わせて分割した例を表 5.21 に示す．JESC データセットに合わせた分割では”you”を除く全ての単語が細かく分割されていることが分かる．これは，JESC データセットが大規模であり，かつ映画および TV の字幕というドメインにより，多彩な語彙が含まれていることが原因と考えれる．文字数が圧倒的に多い日本語と，アルファベットのみと文字数が限られる英語を連結してサブワード化を行ったことで，日本語に多くの語彙数が割かれ，英語はより細かく分割され少ない語彙数に留まった結果である．

表 5.20: 各学習データのデータセットに合わせて語彙分割した場合の各テストデータの平均文長（行：学習データのデータセット，列：テストデータのデータセット）

	ASPEC	KFTT	JESC
ASPEC	30.69	39.22	12.69
KFTT	42.40	32.01	11.75
JESC	**80.82**	**71.99**	**20.90**

一方で ASPEC と KFTT は語彙の分割においてそれほど大きな差はなく，語彙の分割が適切になされたとしても，ドメイン外データセットの翻訳の問題は解決され

表 5.21: JESC のテストデータの一文を各データセットに合わせて分割した例（”_” は SentencePiece による半角スペースを表す特殊記号)

ASPEC	_/y/ou/_want/_my/_opinion/_/?
KFTT	_you/_want/_my/_opinion/_?
JESC	_you/_w/an/t/_m/y/_/op/in/i/on/_?

ないことが分かる．以上の議論と合わせて，ドメイン外翻訳の問題は NMT モデルの構造の変化でも，語彙の対応でも解決できない事がわかる．人間に置き換えて考えてみても，知識の無いドメインの文は，翻訳はおろか理解すら難しい．この問題は，機械翻訳の問題の一つである未知語や低頻度語の問題と重なるものであり，本研究で注目しているモデル構造とは異なるアプローチの解決方法が求められる．

5.2.6 翻訳文の文長に基づく比較

翻訳文の長さと翻訳精度の関係を明らかにするために，3 つのデータセット全てに対して，テストデータの翻訳文を，翻訳前の原言語文の文長に基づいて分割し，文長毎に BLEU スコアを計算した．文長は SentenecPiece のトークン数で測り，またこれとの対応を保つために，KyTea による再分割は行っていない．

Transformer は絶対的な位置情報を扱っており，学習データにない長い文の扱いにおいて不利であると考えられる一方で，遠い単語間の関係を捉えることができるという利点もある．本実験では，これらのどちらの影響がより大きいか明らかにする．

まず参考資料として，各データセットの学習データおよびテストデータの原言語文の文長毎の割合を図 5.3，5.4 に示し，そして文長毎の BLEU スコアを図 5.5〜5.7 に示す．

まず ASPEC について，4 つのモデル全てにおいて，文長 50〜59 から文長 60〜にかけて，スコアの急な下降が見られる．また，SS 以外の 3 つのモデルについては，文長 40〜49 から文長 50〜59 にかけてもスコアが低下している．これらについては，学習時に文長制限を 50 としていることが主な原因と考えられる．また，4 つのモデ

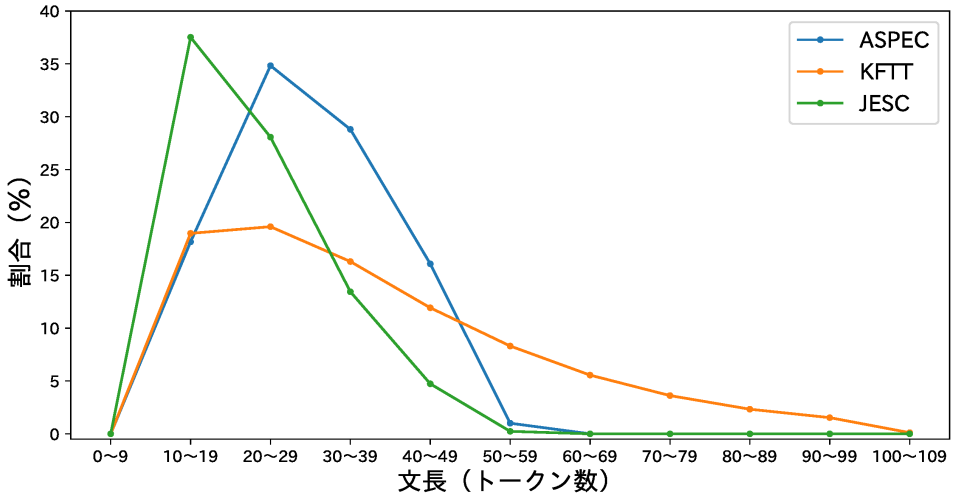

図 5.3: 各データセットの学習データ（前処理後）の原言語文の文長毎の割合

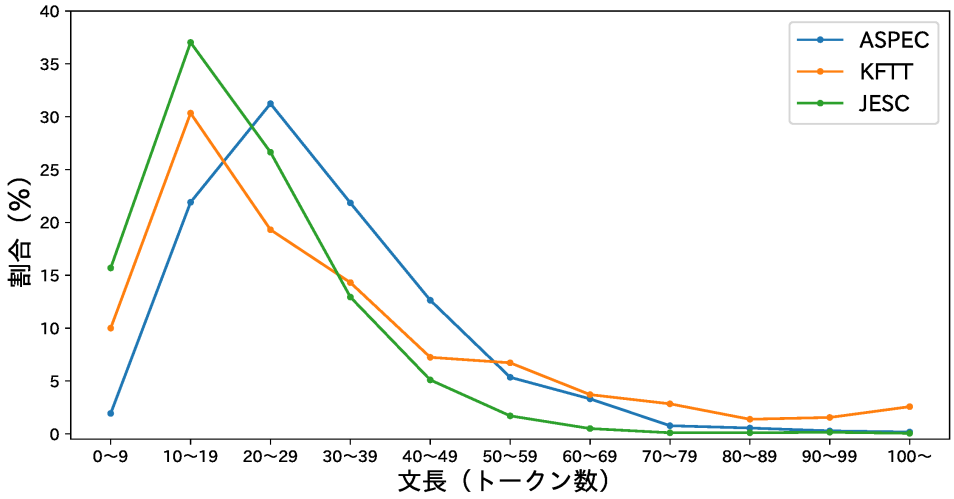

図 5.4: 各データセットのテストデータの原言語文の文長毎の割合

ル間で下降の様子の比較をすると，SS と ST は緩やかで，TS と TT は急である．これらはそれぞれ共通のエンコーダを用いていることから，学習データ以上の文長を

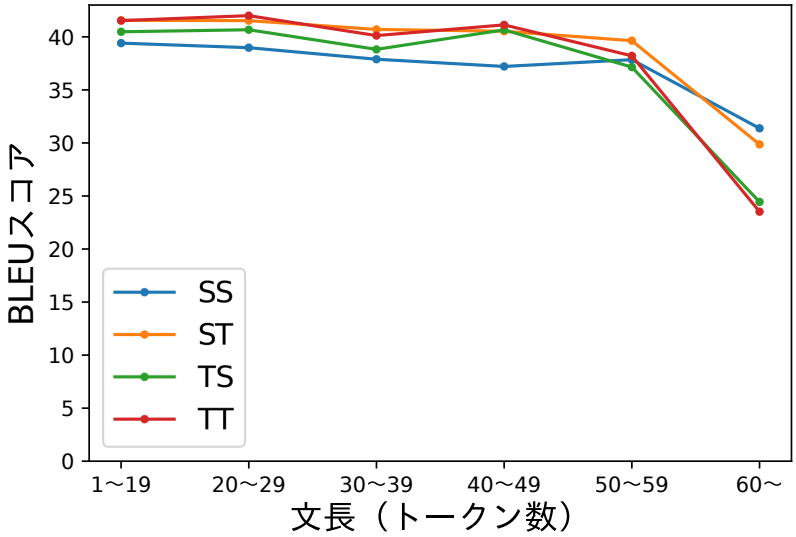

図 5.5: ASPEC における原言語文の文長毎の BLEU スコア

もつ文の翻訳については，seq2seq のエンコーダが，Transformer のエンコーダより優れていることが分かる．やはり，学習データを超える範囲の位置情報については，RNN による相対的な位置の扱いが，絶対的な位置情報の付加手法に比べて優れていると考えられる．離れた単語間の扱いよりも，位置情報の学習のほうがより重要であることが分かる．

KFTT については，文長 60〜69 から文長 70〜にかけて，SS 以外の 3 つのモデルについてスコアの下降が見られ，特に TT が最も急な下降をしている．KFTT は，ASPEC や JESC と異なり，文長制限を 100 としているため，学習データより長い文長のテストデータが非常に少なく，そのため，この結果を元に学習データ以上の文長をもつ文の翻訳についての議論をすることは出来ない．しかしながら，この図の結果から，学習データ以下の文長の翻訳においては，全ての文長に対して ST，TT が SS，TS よりも優れている事が分かる．

JESC については，異なる文長に渡って，各モデルの優劣が大きく入れ替わる結果となっている．5.2.2 でも述べたように，JESC においては，各モデルのテストデー

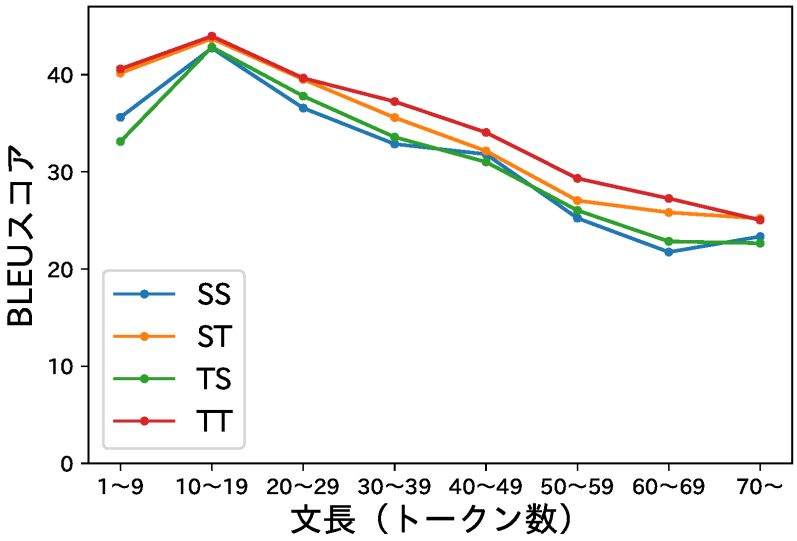

図 5.6: KFTT における原言語文の文長毎の BLEU スコア

タでの BLEU スコア及び RIBES スコアが低く，データセット自体の問題が難しく，どのモデルも十分に機能していないことが考えられる．しかし一方で，学習データ以上の文長をもつ文の翻訳については，文長 40〜49 から文長 50〜にかけて，TS，TT，SS，ST の順にスコアの下降が穏やかになっており，ASPEC と同様に，seq2seq のエンコーダが，Transformer のエンコーダより優れる傾向を示している．

5.2.7 学習データの文長を制御した場合の翻訳精度

学習データの文長と翻訳精度の関係を明らかにするために，学習データを文長に基づいてデータセット 3 つに分割して各モデルの学習を行い，その翻訳精度を比較する．学習データ以上の文長については seq2seq のエンコーダが優れることが示されたが，ここでは，学習データがある場合の seq2seq と Transformer のエンコーダ同士を比較する．

データセットには ASPEC のみを用い，文長順に並び替えた学習データを，トーク

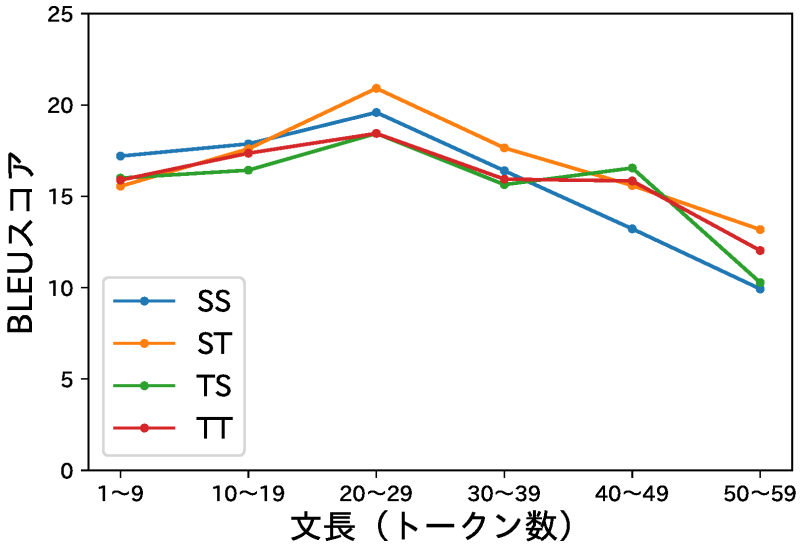

図 5.7: JESC における原言語文の文長毎の BLEU スコア

ン数に基づいて 3 分割した．分割した学習データを文長に基づき，それぞれ short，middle，long と呼ぶ．分割した学習データの内訳を表 5.22 に示す．これらの 3 つの学習データを用いて，それぞれ 4 種類のモデルを学習し，5.2.6 と同様に，それぞれテストデータの原言語の文長毎の翻訳精度を評価した．学習データ毎の結果を表 5.8，表 5.9，表 5.10，に示す．

表 5.22: ASPEC の学習データを文長毎に 3 等分した学習データの内訳

	最小文長	最大文長	文対数	単語数
short	2	27	623,520	12,327,165
middle	27	36	394,949	12,327,168
long	36	50	296,026	12,327,117

どの学習データにおいても，その学習データがカバーする文長の区間にでは，TT が ST に勝り，TS が SS に勝っている．これにより，学習データが存在するのであれば，Transformer のエンコーダが優れていることが分かる．

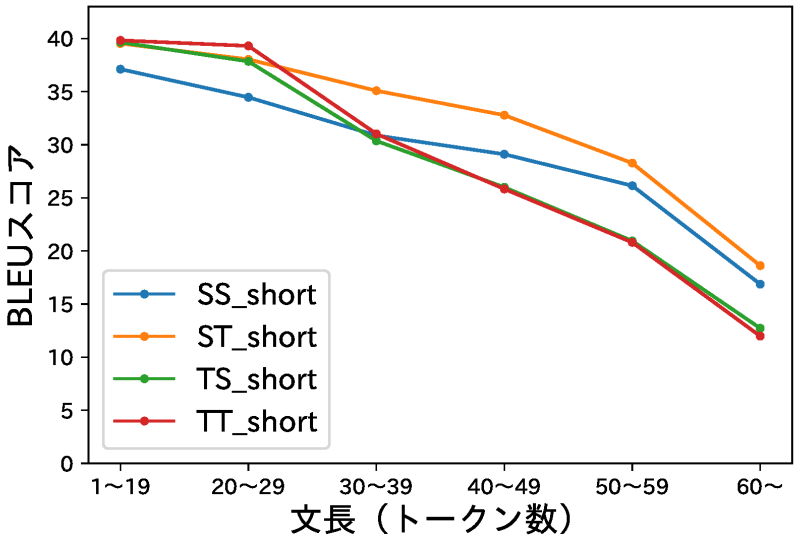

図 5.8: short 学習データにおける各モデルの原言語文の文長毎の BLEU スコア

一方，short と middle では，学習データがカバーしない長文において TS と TT が顕著にスコアが低いことが確認できる．さらに，予想外の結果として，long の文長 1〜19 から 20〜29 にかけて，TS と TT が大きくスコアの変動をしており，学習データが存在しない範囲では，長文だけでなく，短文においても，Transformer のエンコーダが seq2seq のエンコーダに劣ることが示された．

この傾向は，モデル毎にまとめた結果でさらにはっきりと確認できる．3 つの学習データを全て用いた場合の all を加え，結果をモデル毎に図 5.11〜5.14 に示す．

これらの図の文長 1〜19 における short と long の差に注目すると，TS と TT における差が明らかに大きく，SS と ST における差は小さいことが分かる．学習データに存在しない長文を扱うにあたっては，Transformer の絶対的な位置情報の扱いによって，位置情報を学習しなければならないことが問題となると予想していたが，短文については，長文を扱う中でその位置情報を学習できるはずである．これについては，新たな仮説として，NMT モデルが文長についてのバイアスを学習していることが考えられる．つまり，学習データが長文ばかりであると，どのような入力

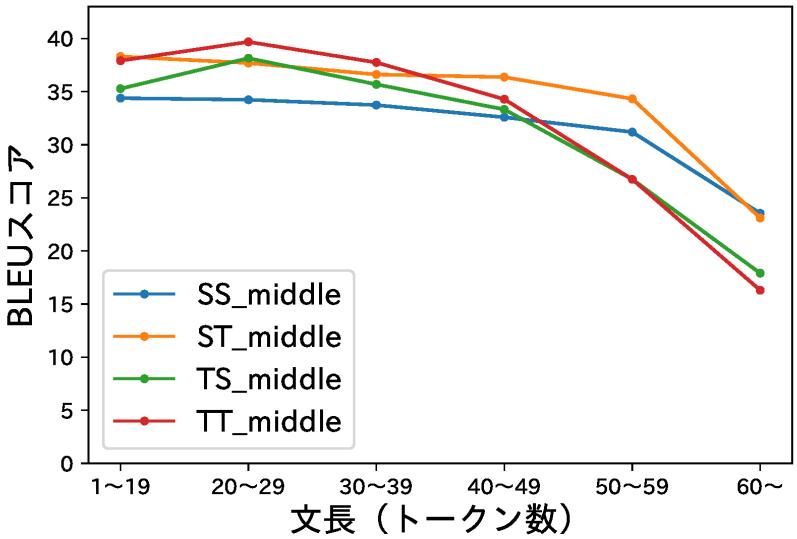

図 5.9: middle 学習データにおける各モデルの原言語文の文長毎の BLEU スコア

に対しても長文を出力してしまうのではないかということである．short 学習データと，long 学習データについて，文長毎に各モデルの参照訳との符号付き文長差の平均を図 5.15 と図 5.16 に示す．これらの図を見比べると，文長 1〜19 において，long 学習データでは TS と TT が参照訳より長く文を出力していることが分かる．

また同時に，short 学習データにおいて，ST と TT が長文の翻訳時に参照役より短く文を生成していることがわかる．これは，学習データにない長文の翻訳で ST が，特に seq2seq のエンコーダの貢献により優れるという結果に反する．翻訳文の例を SentencePiece トークンでの文長と共に表 5.23 に示す．この例から，seq2seq のデコーダが適切な文長を生成しようとする一方で，既に出力した単語を繰り返し，不適切な文を出力していることが分かる．適切な文長を達成しても，翻訳精度は上がらない．そのため，学習データにない長文の翻訳での seq2seq のエンコーダの貢献は否定されない．ただし一方で，seq2seq のデコーダが文長については適切に扱えていることも示された．これは RNN による相対位置の影響により，学習データ以上の文長も扱うことができているということに他ならない．

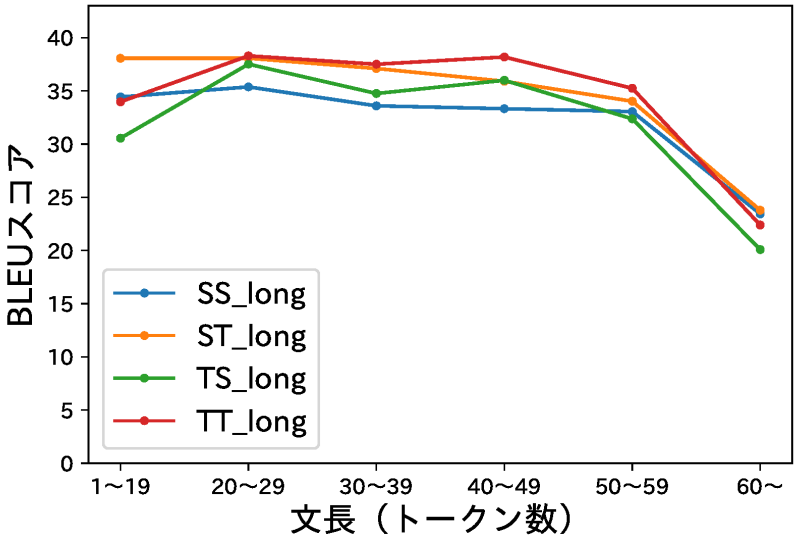

図 5.10: long 学習データにおける各モデルの原言語文の文長毎の BLEU スコア

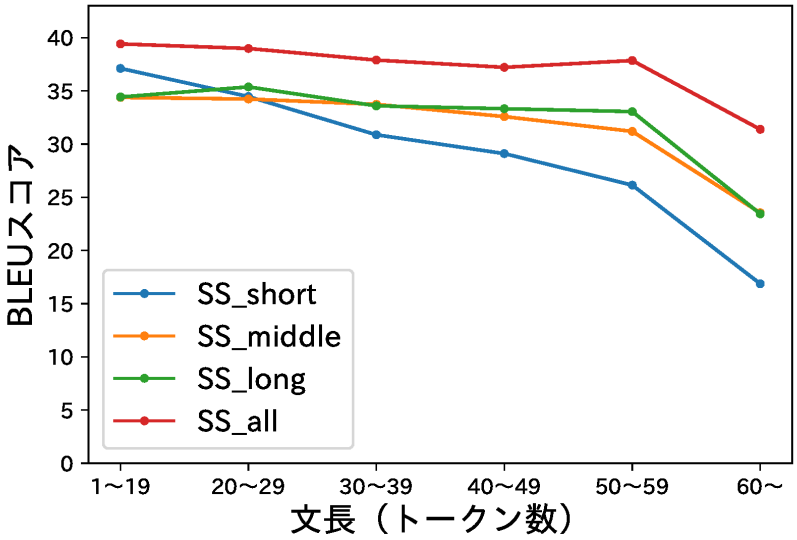

図 5.11: SS の各学習データにおける原言語文の文長毎の BLEU スコア

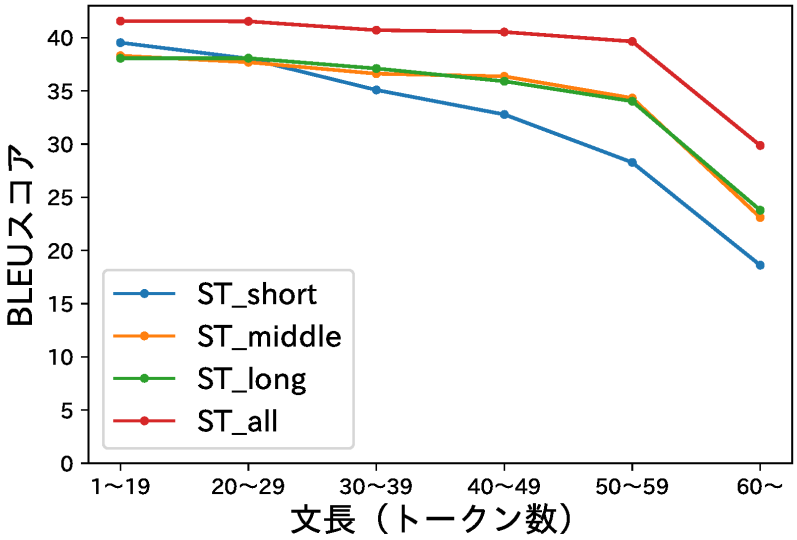

図 5.12: ST の各学習データにおける原言語文の文長毎の BLEU スコア

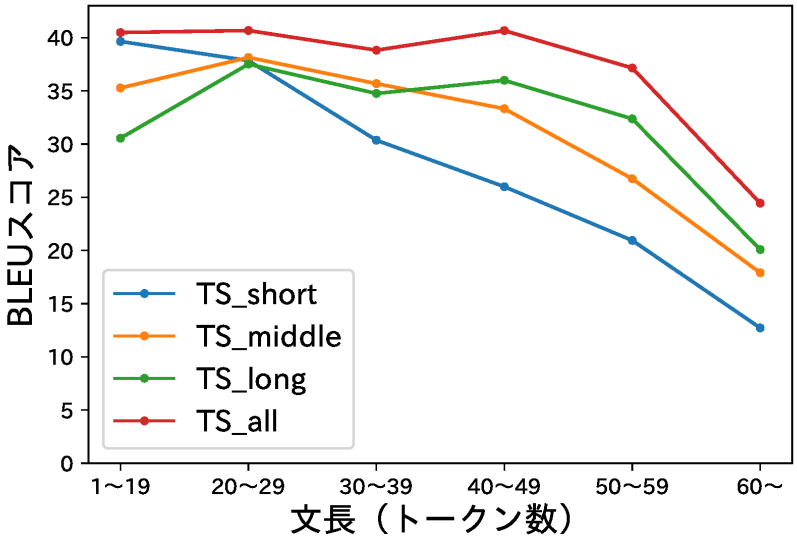

図 5.13: TS の各学習データにおける原言語文の文長毎の BLEU スコア

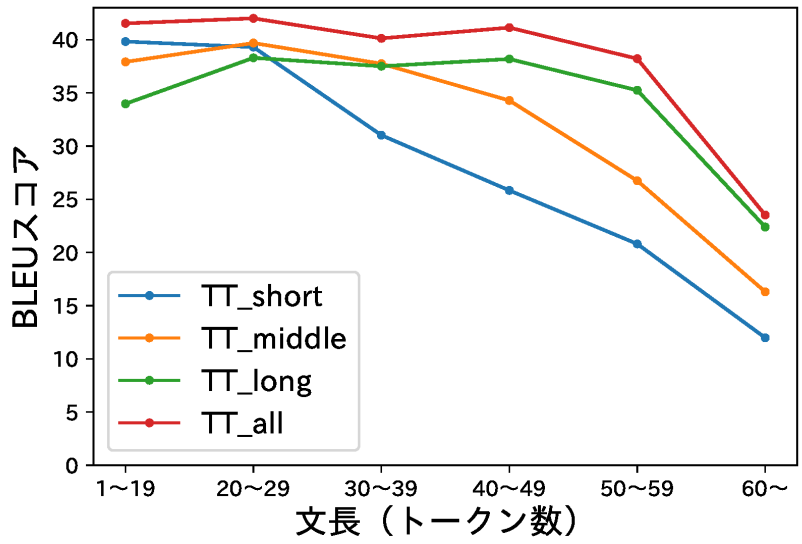

図 5.14: TT の各学習データにおける原言語文の文長毎の BLEU スコア

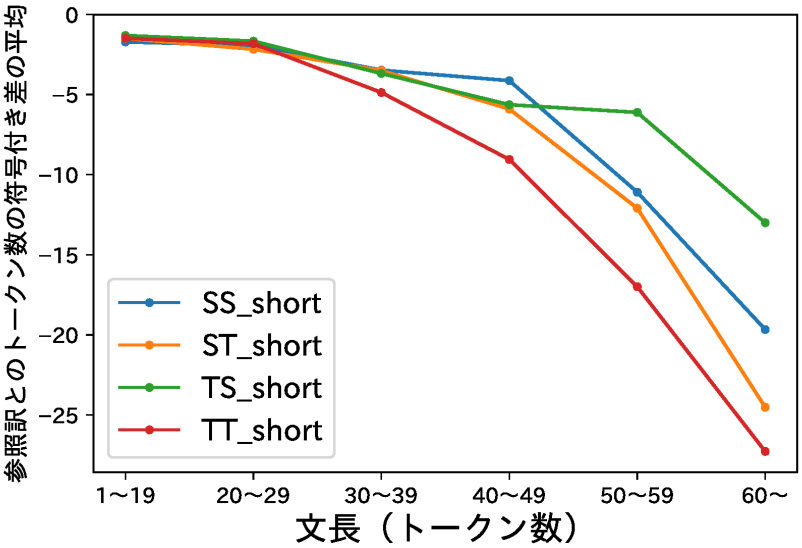

図 5.15: short 学習データにおける各モデルの文長毎の参照訳との符号付き文長差の平均

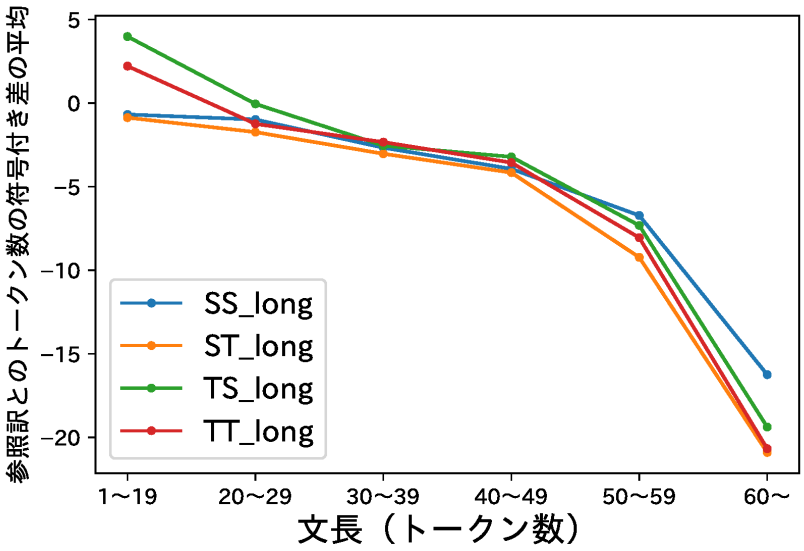

図 5.16: long 学習データにおける各モデルの文長毎の参照訳との符号付き文長差の平均

表 5.23: short 学習データにおける各モデルの長文の翻訳例

	文長	文
原文	58	two methods were adopted : a method to thrust rubber and pipe together to the axial direction with a pair of cylinders on the left and right , and the other method to thrust rubber and pipe individually to the axial direction with two pairs of cylinders on the left and right .
参照訳	51	左右一対のシリンダーでゴムと管を一緒に軸方向に押込む方法と, 左右二対のシリンダーでゴムと管を別々に軸方向に押込む方法を採用した..
SS	64	実験は 2 通りの方法で, 左右には軸方向に一対の円筒を持つように, 他の方法で左右の 2 対の円筒を押さえ, 他の方法では軸方向に管を置き, 他の方法では左右のものである.
ST	30	2 つの方法を用いて, ゴムと管を左右の 2 対の円柱で軸方向に対して推力した.
TS	69	左右の直管と直円の左右の直管と, 左右の左右の直管に対して, 左右の左右の対角を左右する方法と, 左右の左右の左右の左右の対角に対して左右の左右の方向に対して左右のない管を持つ..
TT	41	左右の円柱対と左右の対を持つ管軸方向に伸縮ゴムと管をそれぞれ左右の円柱に対してそれぞれ独立に曲げる方法を採用した..

第6章　おわりに

本研究では，seq2seq，Transformer，そしてそれらの 2 つのスワップモデルの合計 4 種類の NMT モデルを対象に，3 種類のデータセットによる英日翻訳タスクを通して，比較実験及び分析を行った．比較においては NMT モデルの構造に注目し，エンコーダデコーダレベルの詳細な分析を行った．

実験によって得られた知見は以下の通りである．

1. エンコーダとデコーダの組み合わせにより，モデルの機能（得意とする問題）は変わる．

2. Transformer のエンコーダ・デコーダの影響力は強く，2 つのスワップモデルは共に，翻訳結果及び学習データ量に対する翻訳性能において，seq2seq よりも Transformer に類似する．

3. Transformer のエンコーダ・デコーダどちらかのみを NMT モデルに含めることで学習が効率化され，少ない学習データでの翻訳精度が向上する．

4. ビームサーチはデコーダと関連が強く，Transformer のデコーダはビームサイズを大きくした時に精度の低下が少ない．

5. ドメイン外の翻訳についてははどのモデルにも優位性はなく，モデル構造以外でのアプローチが必要である．

6. 長文の翻訳ではエンコーダの影響が大きく，特に学習データが少ない場合は seq2seq のエンコーダが優れ，学習データが十分にある場合は Transformer のエンコーダが優れる．

7. 出力文長の制御ではデコーダの影響が大きく，seq2seq のエンコーダは，不適切な文を出力しつつも，Transformer のデコーダよりも適切な長さの文を出力する．

殆どの NMT の研究において，NMT モデルのエンコーダとデコーダはセットとして扱われ，分離されることが無い中で，本研究は，スワップモデルを比較対象に加えることで，NMT モデルの中では翻訳精度に劣ると示されている seq2seq について，そのエンコーダの価値を再発見した．また，学習データ外の長さの文に対しては，Transformer の絶対的な位置情報の扱いは適しておらず，RNN による相対的な位置情報の扱いが有効であることを確認した．

今後の課題としては，本研究で比較した 4 種類の NMT モデルの中で最も翻訳精度に優れる seq2seq のエンコーダと Transformer のデコーダを組み合わせたスワップモデルを元に，絶対的な位置情報を扱うデコーダを，相対的な位置を扱う様に変更を加えることが考えられる．Transformer に相対位置を付け加えたモデルは Peter ら [44] によって既に提案されている．しかしこの提案手法は RNN を用いているものではないため，RNN 以外の方法での相対位置の扱いが今回の分析によって分かった，学習データ外の文長に対応しているかの分析が必要である．

ドメイン外の翻訳の問題は，本研究の着眼点である NMT モデルの構造からは議論することが出来なかったが，機械翻訳において避けることの出来ない重要な課題の一つである．今回の分析では語彙の面での分析を試みたが，その他のアプローチの模索が必要である．￥

参考文献

[1] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 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, October 2014.

[2] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NIPS) 27*, pp. 3104–3112. 2014.

[3] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 311–318, 2002.

[4] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1412–1421, 2015.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

[6] Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. The best of both worlds: Combining recent advances in neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 76–86. Association for Computational Linguistics, 2018.

[7] Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. Neural versus phrase-based machine translation quality: a case study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 257–267. Association for Computational Linguistics, 2016.

[8] Antonio Toral and Víctor M. Sánchez-Cartagena. A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 1063–1073. Association for Computational Linguistics, 2017.

[9] Philipp Koehn and Rebecca Knowles. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pp. 28–39. Association for Computational Linguistics, 2017.

[10] Minh-Thang Luong and Christopher D. Manning. Stanford neural machine translation systems for spoken language domain. In *International Workshop on Spoken Language Translation*, 2015.

[11] Matthias Huck and Alexandra Birch. Explorer the edinburgh machine translation systems for iwslt 2015. In *International Workshop on Spoken Language Translation*, 2015.

[12] Laura Jehl, Patrick Simianer, Julian Hitschler, and Stefan Riezler. The heidelberg university english-german translation system for iwslt 2015. 2015.

[13] Thanh-Le Ha, Jan Niehues, Eunah Cho, Mohammed Mediani, and Alex Waibel. The kit translation systems for iwslt 2013. 2015.

[14] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of the Third International Conference on Learning Representations (ICLR)*, 2015.

[15] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL), Demo and Poster Sessions*, pp. 177–180, 2007.

[16] Meghan Dowling, Teresa Lynn, Alberto Poncelas, and Andy Way. Smt versus nmt: Preliminary comparisons for irish. In *Proceedings of the AMTA 2018 Workshop on Technologies for MT of Low Resource Languages (LoResMT 2018)*, pp. 12–20. Association for Machine Translation in the Americas, 2018.

[17] M. Amin Farajian, Marco Turchi, Matteo Negri, Nicola Bertoldi, and Marcello Federico. Neural vs. phrase-based machine translation in a multi-domain scenario. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 280–284. Association for Computational Linguistics, 2017.

[18] Huadong Chen, Shujian Huang, David Chiang, and Jiajun Chen. Improved neural machine translation with a syntax-aware encoder and decoder. In *Proceedings of the 55th Annual Meeting of the Association for Computational Lin-*

guistics (Volume 1: Long Papers), pp. 1936–1945. Association for Computational Linguistics, 2017.

[19] Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. Tree-to-sequence attentional neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 823–833. Association for Computational Linguistics, 2016.

[20] Junhui Li, Deyi Xiong, Zhaopeng Tu, Muhua Zhu, Min Zhang, and Guodong Zhou. Modeling source syntax for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 688–697. Association for Computational Linguistics, 2017.

[21] Roee Aharoni and Yoav Goldberg. Towards string-to-tree neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 132–140. Association for Computational Linguistics, 2017.

[22] Akiko Eriguchi, Yoshimasa Tsuruoka, and Kyunghyun Cho. 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. Association for Computational Linguistics, 2017.

[23] Shonosuke Ishiwatari, Jingtao Yao, Shujie Liu, Mu Li, Ming Zhou, Naoki Yoshinaga, Masaru Kitsuregawa, and Weijia Jia. Chunk-based decoder for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1901–1912. Association for Computational Linguistics, 2017.

[24] Shuangzhi Wu, Dongdong Zhang, Nan Yang, Mu Li, and Ming Zhou. Sequence-to-dependency neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 698–707. Association for Computational Linguistics, 2017.

[25] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional sequence to sequence learning. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, Vol. 70 of *Proceedings of Machine Learning Research*, pp. 1243–1252, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

[26] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, Vol. abs/1609.08144, , 2016.

[27] Yanzhuo Ding, Yang Liu, Huanbo Luan, and Maosong Sun. Visualizing and understanding neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1150–1159. Association for Computational Linguistics, 2017.

[28] Surafel Melaku Lakew, Mauro Cettolo, and Marcello Federico. A comparison of transformer and recurrent neural networks on multilingual neural machine translation. In *Proceedings of the 27th International Conference on Compu-*

tational Linguistics, pp. 641–652. Association for Computational Linguistics, 2018.

[29] Felix Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. *Neural computation*, Vol. 12, pp. 2451–71, 10 2000.

[30] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 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. Association for Computational Linguistics, 2014.

[31] Kazuma Hashimoto and Yoshimasa Tsuruoka. Neural Machine Translation with Source-Side Latent Graph Parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 125–135, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[32] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the third International Conference on Learning Representations (ICLR)*, 2015.

[33] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 2818–2826, 2016.

[34] Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. ASPEC: Asian scientific

paper excerpt corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pp. 2204–2208, 2016.

[35] Graham Neubig. The Kyoto free translation task. http://www.phontron.com/kftt, 2011.

[36] R. Pryzant, Y. Chung, D. Jurafsky, and D. Britz. JESC: Japanese-English Subtitle Corpus. *ArXiv e-prints*, October 2017.

[37] Graham Neubig, Yosuke Nakata, and Shinsuke Mori. Pointwise prediction for robust, adaptable Japanese morphological analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT), Short Papers*, pp. 529–533, 2011.

[38] T. Kudo and J. Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *EMNLP, System Demonstrations*, pp. 66–71, 2018.

[39] Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pp. 944–952. Association for Computational Linguistics, 2010.

[40] L. K. Hansen and P. Salamon. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 12, No. 10, pp. 993–1001, October 1990.

[41] Kenji Imamura and Eiichiro Sumita. Ensemble and reranking: Using multiple models in the nict-2 neural machine translation system at wat2017. In *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, pp. 127–134. Asian Federation of Natural Language Processing, 2017.

[42] Matt J. Kusner, Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger. From word embeddings to document distances. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pp. 957–966. JMLR.org, 2015.

[43] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proceedings of the First International Conference on Learning Representations (ICLR)*, 2013.

[44] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 464–468. Association for Computational Linguistics, 2018.

発表文献

査読なし国際会議

1. Masato Neishi, Jin Sakuma, Satoshi Tohda, Shonosuke Ishiwatari, Naoki Yoshinaga, Masashi Toyoda, A Bag of Useful Tricks for Practical Neural Machine Translation: Embedding Layer Initialization and Large Batch Size. The 4th Workshop on Asian Translation (WAT 2017), Taipei, 2017.

査読なし国内会議

1. 根石将人，吉永直樹，英日翻訳タスクにおけるスワップモデルを通した seq2seq と Transformer の比較．言語処理学会第 25 年次大会（NLP 2019），名古屋，2019．（発表予定）

研究会

1. 根石将人，吉永直樹，高性能なオートエンコーダとのマルチタスク学習を利用したニューラル機械翻訳．NLP 若手の会 (YANS) 第 13 回シンポジウム，岡山，2018．

2. 根石将人，佐久間 仁，遠田 哲史，石渡 祥之佑，吉永 直樹，豊田 正史，ニューラル機械翻訳における埋め込み層の教師なし事前学習．第 233 回 自然言語処理研究会（NL233），沖縄，2017．