



Title	事前並び替え手法による機械翻訳性能の向上
Author(s)	瓦, 祐希
Citation	大阪大学, 2021, 博士論文
Version Type	VoR
URL	https://doi.org/10.18910/85272
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

事前並び替え手法による機械翻訳性能の向上

提出先 大阪大学大学院情報科学研究科

提出年月 2021 年 4 月

瓦 祐 希

本研究に関する研究業績

論文誌

1. 瓦祐希, Chenhui Chu, 荒瀬由紀. 統計的機械翻訳のための Recursive Neural Network による事前並び替えと分析. 言語処理学会論文誌 自然言語処理, Vol. 26, No. 1, pp. 155–178, 2019 年 3 月.
2. Yuki Kawara, Chenhui Chu, Yuki Arase. Preordering Encoding for Transformer. IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP), Vol. 29, pp. 644–655, December 2020.

国際会議

1. Yuki Kawara, Chenhui Chu, Yuki Arase. Recursive Neural Network Based Pre-ordering for English-to-Japanese Machine Translation. in Proceedings of the Annual Meeting of the Association for Computational Linguistics, Student Research Workshop (ACL-SRW), pp. 21–27, July 2018.
2. Yuki Kawara, Yuto Takebayashi, Chenhui Chu, Yuki Arase. Osaka University MT Systems for WAT 2018: Rewarding, Preordering, and Domain Adaptation. in Proceedings of the Workshop on Asia Translation (WAT), pp. 1120–1126, December 2018.

国内会議

1. 瓦祐希, Chenhui Chu, 荒瀬由紀. Recursive Neural Network を用いた事前並び替えによる英日機械翻訳. 言語処理学会年次大会, pp. 33–36, 2018 年 3 月.

2. 瓦祐希, Chenhui Chu, 荒瀬由紀. ニューラル機械翻訳における事前並び替えの影響分析. 言語処理学会年次大会, pp. 1455–1458, 2019 年 3 月.
3. 瓦祐希, Chenhui Chu, 荒瀬由紀. 事前並び替え位置表現を用いた Transformer による日英機械翻訳. 言語処理学会年次大会, pp. 465–468, 2020 年 3 月.
4. 瓦祐希, Chenhui Chu, 荒瀬由紀. Non-Autoregressive Translation モデルにおける事前並び替え適用手法の検討. 言語処理学会年次大会, pp. 425–429, 2021 年 3 月.

本論文の梗概

国際交流が盛んになるにつれて外国語を母語に、また母語を外国語に変換する翻訳へのニーズが高まっている。翻訳者による翻訳は信頼性、精度がともに高いが、翻訳者の数は限られており、一般ユーザが気軽に使用することは出来ない。そこで、コンピュータにより自動で翻訳を行う機械翻訳への注目が高まっている。

特に日本語、英語間のように語順が大きく異なる言語対の機械翻訳においては、語順の差異が障壁となる。この問題に対して、翻訳器に入力する前に原言語文を目的言語文の語順に並び替える事前並び替え手法が提案されている。事前並び替えによって長距離の並び替えを効果的かつ効率的に行え、翻訳精度の向上につながる。そこで本論文では、代表的な機械翻訳手法に対して事前並び替えを使用し、語順が大きく異なる言語対の機械翻訳における翻訳精度の向上に取り組む。本論文は(1)統計的機械翻訳のための Recursive Neural Network (RvNN) による事前並び替え手法の開発、(2) Transformer のための事前並び替え位置表現の開発、(3) 非自己回帰ニューラル機械翻訳における事前並び替えの効果の検証の三つの研究を実施した。

(1) では RvNN を使用した事前並び替え手法を提案する。当時主流であったフレーズベース統計的機械翻訳において事前並び替えが語順の問題の解決に有効な手段であることが知られていた。しかし、当時最高精度を達成していた既存の事前並び替え手法では人手での特徴量の設計が必要であった。提案手法は構文木に基づく RvNN によって事前並び替えの学習を行うことで、人手での特徴量の設計に依存しない手法を提案する。また RvNN はボトムアップに並び替えの計算を行うため、並び替えの際に重要である部分木を考慮した並び替えが実現出来る。翻訳評価実験では、言語構造の大きく異なる英日対において先行研究である人手での特徴量設計が必要な既存の事前並び替え手法と同等の翻訳精度を達成したことを確認した。

(2) では Transformer のための事前並び替え位置表現手法を提案する。2016 年からフレーズベース機械翻訳に代わり自己回帰ニューラル機械翻訳が主流となつた。自己回帰ニューラル機械翻訳の一つである Recurrent Neural Network ベースの機械翻訳モデルでは事前並び替えのインデックスを使用することで翻訳精度が

向上することが報告されているが、現在機械翻訳のスタンダードなモデルである Transformer において事前並び替えが有効に活用出来るかは定かではなかった。提案手法である事前並び替え位置表現は、事前並び替えのインデックスに基づいたベクトル表現であり、Transformer において事前並び替えを活用した初めての事例である。特に各言語から英語方向への翻訳において事前並び替えの精度が高い場合、提案手法によって翻訳性能が向上することを示した。また目的言語が英語の翻訳においては、事前並び替えの情報を使用することで中程度の長さ以上の文において事前並び替え位置表現が有効に働くことがわかった。

(3) では非自己回帰ニューラル機械翻訳モデルにおける事前並び替えの利用を検討する。2018 年に自己回帰ニューラル機械翻訳において翻訳文の出力の際に単語数に比例した時間がかかるという問題の解決のために非自己回帰ニューラル機械翻訳が提案された。非自己回帰ニューラル機械翻訳では原言語文の語順のベクトル表現に基づいて翻訳文を出力するが、言語構造が大きく異なる場合に翻訳精度が大きく低下してしまう。そこで非自己回帰ニューラル機械翻訳モデルにおいて事前並び替えの利用の検討を行い、事前並び替えの情報が翻訳精度に与える影響を調査した。翻訳評価実験の結果、非自己回帰ニューラル機械翻訳では並び替えた文を直接入力として使用しても翻訳精度が向上することが確認出来た。また、特に 20 単語以上の文において事前並びを使用することで翻訳精度が向上することが分かった。

目次

第 1 章	序論	1
1.1	研究背景	1
1.1.1	機械翻訳需要の高まり	1
1.1.2	機械翻訳モデルの変遷	2
1.2	機械翻訳の課題	5
1.3	本研究の位置づけと貢献	8
1.4	本論文の構成	9
第 2 章	統計的機械翻訳のための RvNN による事前並び替え	11
2.1	はじめに	11
2.2	関連研究	12
2.3	前提知識	15
2.3.1	フレーズベース統計的機械翻訳	16
2.3.2	注意機構付き RNN モデル	17
2.4	提案手法	19
2.4.1	単語アライメント	19
2.4.2	正解ラベルの付与	20
2.4.3	事前並び替えモデル	21
2.5	翻訳評価実験	24
2.5.1	実験設定	24
2.5.2	評価指標	26
2.5.3	実験結果	28
2.6	分析	34
2.6.1	機械学習手法の効果	34
2.6.2	翻訳例の分析	34
2.7	おわりに	35
第 3 章	Transformer のための事前並び替えのベクトル表現	41

3.1	はじめに	41
3.2	関連研究	42
3.3	提案手法	43
3.3.1	前提知識: Transformer	43
3.3.2	前提知識: 絶対的位置表現と相対的位置表現	46
3.3.3	事前並び替え位置表現	48
3.4	翻訳評価実験	49
3.4.1	実験設定	50
3.4.2	実験結果	52
3.5	分析	55
3.5.1	事前並び替え位置表現による翻訳精度の上限	56
3.5.2	事前並び替えの精度と翻訳精度の関係	58
3.5.3	訳抜けと重複訳に対する効果	59
3.5.4	文長と翻訳の関係	62
3.6	おわりに	65
第 4 章	非自己回帰ニューラル機械翻訳における事前並び替えの効果の検証	69
4.1	はじめに	69
4.2	LaNMT における事前並び替えの適用	70
4.2.1	前提知識: LaNMT	70
4.2.2	事前並び替えの適用	72
4.3	翻訳評価実験	74
4.3.1	実験設定	74
4.3.2	実験結果	75
4.4	分析	76
4.4.1	文長と翻訳精度の関係	76
4.4.2	Knowledge Distillation に対する影響	77
4.4.3	Byte Pair Encoding による翻訳精度の変化	79
4.4.4	翻訳例	80
4.5	おわりに	83
第 5 章	結論	85
	謝辞	87

参考文献	89
------	-------	----

第1章 序論

1.1 研究背景

1.1.1 機械翻訳需要の高まり

国際交流が盛んになるに伴い、翻訳へのニーズが高まっている。翻訳者による翻訳は信頼性、精度がともに高いが、翻訳者の数は少なくコストがかかるため、一般的なユーザが気軽に利用することは出来ない。そこで、コンピュータによって自動で翻訳を行う機械翻訳技術への注目が高まっている。図 1.1 に、現在性能が一番いいと言われている、DeepL GmbH によって開発された DeepL¹⁾ による機械翻訳の一例を示す。この例では日本語から英語方向を指定し、翻訳したい日本語文を入力することで英文が出力されている。このようにブラウザ上で簡単に翻訳を行えるツールが提供されていたり、NICT による VoiceTra²⁾ のように音声の翻訳を行えるアプリケーションが提供されていたりする。また公開時点では二言語間の翻訳にしか対応していなかったグーグル翻訳は、2020 年時点で 100 言語を超える言語の翻訳に対応している。さらに使用しているユーザ数も数百人から数百万人へと増加しており、機械翻訳技術に対する注目は徐々に高まってきている。³⁾⁴⁾国内においても、オリンピックや 2025 年に大阪で開催予定の万国博覧会によって外国人の流入が増加することが予想されるため、機械翻訳に対する需要が高くなっている。機械翻訳技術がより発達することで母国語以外の言語を使用する相手とのコミュニケーションを行うことがより容易になり、円滑なやりとりが可能となる。

1) <https://www.deepl.com/home>

2) <https://voicetra.nict.go.jp/>

3) <https://blog.google/products/translate/ten-years-of-google-translate/>

4) <https://ai.googleblog.com/2020/06/recent-advances-in-google-translate.html>

Original text (Japanese):

世界人権宣言では「すべての人間は、生まれながらにして自由であり、かつ、尊厳と権利について平等である」とうたわれている。これは、すべての人が基本的人権を享有し、法の下に平等であると定めている日本国憲法と共通の理念である。しかしながら、今日もなお、人種、民族、信条、性別、社会的身分、障害があることなどにより人権が侵害されている現実があり、また、社会状況の変化等により、人権にかかわる新たな課題も生じてきている。

Translated text (English):

The Universal Declaration of Human Rights states that "All human beings are born free and equal in dignity and rights. This is a principle shared by the Japanese Constitution, which stipulates that all people enjoy fundamental human rights and are equal under the law. However, the reality is that human rights are still violated today due to race, ethnicity, creed, gender, social status, disability, etc. In addition, new issues related to human rights have arisen due to changes in social conditions."

図 1.1 DeepL による機械翻訳の例 (日本語の例文は「吹田市人権尊重の社会をめざす条例」より引用)

1.1.2 機械翻訳モデルの変遷

機械翻訳の始まり 機械翻訳は 1947 年に Warren Weaver から Norbert Wiener に送られた手紙を発端としている [69]。Weaver はその手紙の中で翻訳を暗号解読の問題として捉えており、さらに二年後には機械翻訳のアイデアを覚書として記述している。その覚書の中で、情報理論 [81] に基づいた暗号解読や統計的手法、さらに言語学的な知識も取り入れて機械翻訳の可能性について論じており、それを契機に大学や研究所などで機械翻訳の研究が盛んになった。1954 年には Georgetown 大学と IBM が共同で開発したロシア語から英語への機械翻訳のデモンストレーションが発表される [38] など、1950、60 年代に機械翻訳の研究が盛り上がったがその翻訳精度は実用には程遠く、1966 年の ALPAC (Automatic Language Processing Advisory Committee) によるレポート [16] では “there is no immediate or predictable prospect of useful machine translation” と報告され、その結果 10 年以上機械翻訳の研究が停滞した。

原言語文とそれに紐づいた目的言語文による対訳コーパスなどの言語資源が充実し始めたのに伴い、1980 年代からデータに基づく翻訳手法が提案されるようになった。図 1.2 に示すように、データに基づく機械翻訳では対訳コーパスから機械翻訳のモデルを学習し、そのモデルに基づいて翻訳を行う。1984 年には Nagao [64] が用例に基づく機械翻訳を提案した。用例に基づく機械翻訳は事前に翻訳の知識や確率モデルなどを仮定せず、対訳コーパスから自動で用例を獲得し、翻訳文の生成の際は入力文と用例との編集距離のような類似度に基づいて翻訳を決定する手法で

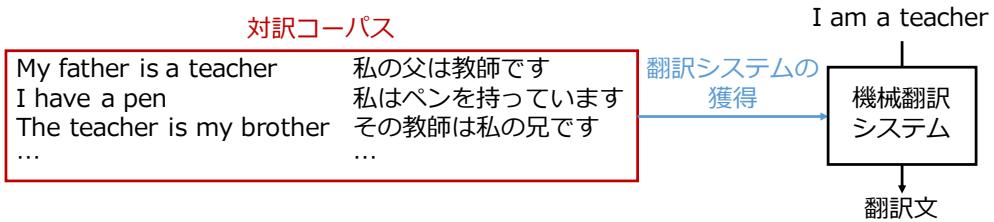


図 1.2 データに基づく機械翻訳の概要図

ある。しかし、入力文と類似した用例が見つかるることは多くないため、複数の用例から部分的な訳を取り出し、それらを組み合わせることによって翻訳文を決定する。

統計的機械翻訳 用例に基づく機械翻訳では対訳コーパスから用例を取得し、入力文との類似度に基づいて翻訳を決定するが、その類似度の計算はシステムの開発者に委ねられているという問題点がある。そこで 1990 年に対訳コーパスから統計的なモデルを獲得する統計的機械翻訳が Brown et al. [7] によって提案された。用例に基づく機械翻訳では対訳コーパスから自動的に翻訳対を抽出し、入力文に近い対訳のペアから翻訳を決定するが、統計的機械翻訳では翻訳の正しさだけではなく目的言語文の流暢さも同時に考慮して翻訳文を決定する。2003 年には Koehn et al. [50] によってフレーズベース統計的機械翻訳が提案された。その後も階層的フレーズベース統計的機械翻訳手法 [13] や構文木に基づいた統計的機械翻訳手法 [35, 56] が提案され、2006 年には機械翻訳の国際的なコンペティションである Workshop on statistical machine translation (WMT) が開催される [49] など、統計的機械翻訳は 20 年ほど機械翻訳のベースラインとなった。

自己回帰ニューラル機械翻訳 2006 年に Hinton et al. [32] によって発表されたニューラルネットワークによる次元削減の研究を皮切りに、ニューラルネットワークを使用した機械学習の研究が盛んになり、Recurrent Neural Network (RNN) による言語モデルがそれまでの統計的な言語モデルの精度を上回る [61] など、自然言語処理の分野においてもニューラルネットワークを使用した手法が提案され始めた。2014 年には Sutskever et al. [90] によって、RNN ベースの自己回帰ニューラル機械翻訳手法が提案された。自己回帰ニューラル機械翻訳は翻訳文の出力の際に以前に出力した単語に基づいて次の単語を予測するモデルである。単純な RNN ベースの自己回帰ニューラル機械翻訳モデルは長距離の依存関係を考慮することが難し

く、長文の翻訳精度が低い傾向があった。そこで 2015 年には Bahdanau et al. [2] や Luong et al. [57] によって注意機構を使用した RNN ベースの自己回帰ニューラル機械翻訳モデル(注意機構付き RNN モデル)が提案された。このモデルはエンコーダのどの単語に注目するかを計算することで長文の翻訳精度が低くなる問題の解決を図っている。2016 年には WMT においても自己回帰ニューラル機械翻訳が統計的機械翻訳の精度を上回り [4, 79]、ニューラルネットワークを使用したモデルが機械翻訳のスタンダードなモデルとなっていました。

これらのモデルは RNN をベースとしているため、原言語文の入力、翻訳文の出力とともに文長に比例した時間がかかるという問題点があった。入力文のエンコーダにかかる時間を削減する手法として、Gehring et al. [22] によって Convolutional Neural Network (CNN) を使用した自己回帰ニューラル機械翻訳モデルが提案された。しかし CNN では完全な並列化は出来ておらず、ある単語のベクトル表現を計算する際に他の位置の単語のベクトル表現を使用するためには位置の差に比例した計算時間がかかるという問題点が残っていた。ベクトル表現の計算の完全な並列化に向けて、Vaswani et al. [95] は 2017 年に自己注意機構を使用した自己回帰ニューラル機械翻訳モデルである Transformer を提案し、このモデルは現在の機械翻訳のベースラインとなっている。しかし Transformer でも推論の際に以前の単語に基づいて次の単語を予測するために単語の出力を並列に行うことが出来ず、翻訳文の出力にその単語数に比例した時間がかかるてしまう。

非自己回帰ニューラル機械翻訳 翻訳文の出力の際に文長に比例した時間がかかるという問題を解決するために、Gu et al. [28] は翻訳文全体を一度に出力する非自己回帰ニューラル機械翻訳手法を提案した。非自己回帰ニューラル機械翻訳は自己回帰ニューラル機械翻訳と比較して翻訳にかかる時間を大幅に短縮することに成功した。しかし前から順番に単語を出力する自己回帰ニューラル機械翻訳モデルとは異なり、翻訳文の単語を一度に全て出力するため、同じ単語の繰り返し(重複訳)や、参照訳と比較して単語が欠如する(訳抜け)という問題が頻繁に起こる。Lee et al. [53] や Gu et al. [29] は出力した文を再度デコーダの入力として使用し繰り返し翻訳文を出力することで翻訳精度が低下してしまう問題の解決を図っている。Ghazvininejad et al. [24] らは一度出力した翻訳文に対して、どの単語の翻訳を変えるべきかのマスクをかけて再度翻訳を行う手法を提案しており、Wang et al. [98] は n -gram のチャンクごとにまとめて前から順に出力する手法を提案し周囲の単語

が考慮出来ない問題を解決している。また Wei et al. [102] らは非自己回帰ニューラル機械翻訳モデルのベクトル表現を自己回帰ニューラル機械翻訳モデルのベクトル表現に近づくように学習する手法を提案している。Wang et al. [101] は目的言語文の単語のベクトル表現の類似度とその単語に対応したデコーダ内のベクトル表現の類似度が近づくように学習させ、さらにデコーダのベクトル表現から原言語文を復元するタスクを設定することで重複訳や訳抜けの問題の解決を図っている。一方、Kaiser et al. [41] や Ma et al. [58]、Shu et al. [83] は潜在変数を使用した非自己回帰ニューラル機械翻訳モデルを提案している。これらの非自己回帰ニューラル機械翻訳モデルによる翻訳精度は年々上昇しており、データセットによっては自己回帰ニューラル機械翻訳モデルを上回る翻訳精度を達成している。

現在の機械翻訳 現在の機械翻訳の分野では、自己回帰ニューラル機械翻訳モデルの一つである Transformer をベースラインとして研究がなされている。非自己回帰ニューラル機械翻訳は近年注目が集まって盛んに研究がなされている。非自己回帰ニューラル機械翻訳の翻訳精度は、データセットによっては自己回帰ニューラル機械翻訳を超えるものも存在しているが、平均的にはまだ比肩しうるものではなく、現在では自己回帰ニューラル機械翻訳、非自己回帰ニューラル機械翻訳の両方が主に研究されている手法である。

1.2 機械翻訳の課題

1.1.2 節で議論した機械翻訳技術には、原言語文の単語を目的言語のどの単語に翻訳すべきかを決定する語彙選択の問題や、対訳データが少ない言語対において翻訳精度が低下してしまう低資源機械翻訳の問題、新しく使用され始めた単語の翻訳が存在しない新語翻訳の問題、原言語と目的言語の構造が大きく異なることによって翻訳精度が低下する語順の問題など解決すべき問題が多く存在している。本研究では語順の問題を取り組む。

語順の問題は、どの順番で原言語文の単語の翻訳を行うべきかを決定する問題である。例えば英語から日本語への翻訳を行う際に、“My father is a teacher.” という文をそのままの語順で翻訳を行ってしまうと正しい日本語文の語順とならないため、原言語文をどの順で翻訳していくべきかを考慮する必要がある。特に語順の問題は、英日対のように言語構造が大きく異なる言語間の翻訳において無視出来ない

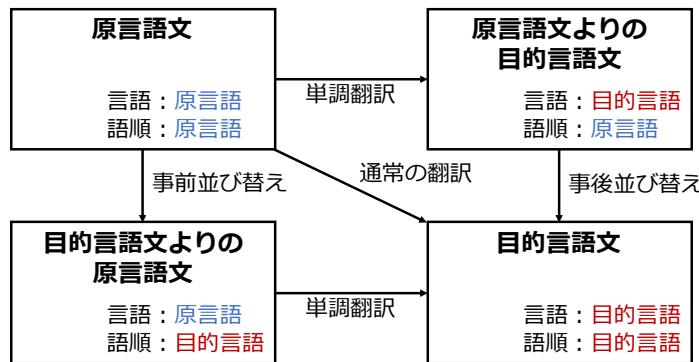


図 1.3 事前並び替えと事後並び替えの処理手順 (図は [86] より引用)

問題であり、本論文ではこの語順の問題の解決に取り組む。

翻訳と同時に語順を考慮する手法 これまでにさまざまな語順の問題に対する解決手法が提案されており、その一つに翻訳を行う際に同時に語順を考慮する手法 [27, 65, 92, 106] がある。例えば Tillmann [92] は直前に出力したフレーズに対して、翻訳するべき原言語文の次のフレーズがその直前か直後に出力されるか、それとも隣接していないかを予測することで語順の問題の解決を図っている。この手法では前のフレーズと比較して前後どちらに出力すべきかの予測は出来るが、その位置までは考慮することが出来ない。また Green et al. [27] は前のフレーズに対して次に出力するフレーズのインデックスの距離をクラスとして分類する手法を提案している。しかしこの手法ではすべての距離をクラスとして考慮することは出来ない。これらの手法は、英仏対のように語順が近い言語間では局所的な語順を考慮することで精度の高い翻訳が実現出来るが、英日対のように語順が大きく異なる言語間では局所的な語順の考慮だけでは語順の問題を解決出来ないため翻訳精度が低く、また翻訳時に同時に長距離の語順を考慮しようとすると計算量が爆発してしまうといった問題点がある。

翻訳とは独立して語順を考慮する手法 翻訳とは独立して並び替えを行う手法として「事後並び替え」と「事前並び替え」と呼ばれる手法が提案されている(図 1.3)。事後並び替えは原言語文をそのままの語順で翻訳を行う単調翻訳を行った後に、原言語文の語順で翻訳された翻訳文を目的言語文の語順に並び替える手法である。特に日英翻訳における事後並び替え [30, 86, 88] では、SVO 型言語の文を SOV 型言語の文の語順に近づける head finalization [37] を使用して原言語文の語順と同じ目的言語文を作成する。その後、作成した対訳コーパスを使用して翻訳器を学習

し原言語文の語順の翻訳文を出力した後、その翻訳文を目的言語文の語順に並び替えることで最終的な翻訳文を決定している。

事前並び替えは翻訳器に原言語文を入力する前にあらかじめ原言語文を目的言語文の語順に近づくように並び替える手法であり、特に長距離の並び替えを効果的かつ効率的に行える [39, 43, 66]。事前並び替えにおいて、考えうる原言語文の語順全てから最適な語順を探索しようとすると原言語文の単語数の階乗に比例した時間が必要となり、計算量が爆発してしまう。そのため、これまでにさまざまな手法による効率的な事前並び替え手法が提案されてきた。

Isozaki et al. [37] や Gojun and Fraser [25] は構文木に対して手動で定めたルールを適用することで事前並び替えを実現している。しかし、ある言語間におけるルールを定めるためには原言語文および目的言語文の知識が必要であり、また全ての言語間のルールを定めることは非常に難しい。そこで対訳コーパスから自動で事前並び替えを学習する手法も提案されている。Zhang et al. [108] や Crego and Habash [17] は n -gram からなるチャンクに対して、並び替えルールを対訳コーパスから得る手法を提案している。

また、木構造を使用することで部分フレーズの並び替えをその部分フレーズが対応したノードを並び替える問題へと帰着でき、長距離の部分フレーズの並び替えが容易に実行出来るという利点があるため、木構造を使用して自動で事前並び替えを学習する手法も多く提案されている。特に 2 分木の構文木に限定することで、子ノードを並び替えるかどうかの二値分類として問題を設定出来る。Hoshino et al. [34] は、2 分木に対して順位相関係数であるケンドールの τ [45] が最大となるように二値分類の分類器を使用して各ノードでの並び替えを行っている。DeNero and Uszkoreit [20] や Neubig et al. [68]、Nakagawa [66] は Bracketing Transduction Grammar (BTG)に基づいて構文解析をしながら事前並び替えを同時に行う手法を提案しており、特に Nakagawa の手法はフレーズベース統計的機械翻訳において最高性能を達成している手法の一つである。これらの手法は構文木を構築しながら並び替えも予測するため構文解析を事前に行う必要がなく、構文解析器の精度に依ることなく並び替えを行える。しかしその学習のために素性テンプレートを人手で設計する必要がある。

近年では、素性テンプレートの設計を必要としないニューラルネットワークに基づいた手法も提案されている。de Gispert et al. [19] は Feed-Forward Neural

Network (FFNN) を用いた 2 分木での並び替えを提案している。Botha et al. [6] も FFNN を用いた並び替えを提案しているが、木構造を使用せずに並び替えを行っている。Miceli-Barone and Attardi [60] は RNN を用いて係り受けのノードを辿ることで並び替えを行う手法を提案している。彼らは、単語を出力する “EMIT”、親ノードへ移動する “UP”、 j 番目のノードへ移動する “DOWN $_j$ ” の三つの動作を定義し、RNN で動作を予測することで並び替えを行う。Kanouchi et al. [42] は統計的機械翻訳の翻訳モデルにより抽出するフレーズペアについて、Recursive Neural Network (RvNN) を用いて並び替えラベルを推定することで、翻訳システムの内部で翻訳と同時にフレーズの並び替えを行う。

1.3 本研究の位置づけと貢献

1.2 節で議論したように、機械翻訳の語順の問題に対して事前並び替え手法を使用することで翻訳精度が向上する。特に、BTG に基づく事前並び替え手法は最高性能を達成した手法であるが、人手での特徴量の設計が必要であった。それに対して、第 2 章で提案する RvNN を使用した 2 分木に基づく事前並び替え [43, 111] はニューラルネットワークを使用した手法であり、人手での特徴量の設計が不要である。また、構文木を使用することで長距離の並び替えを容易に行うことが出来る。さらにボトムアップに計算し並び替えを行うため、並び替えの際に重要な部分木を考慮した並び替えが実現出来る。フレーズベース統計的機械翻訳による翻訳評価実験において、言語間の大きく異なる英日間において提案手法による翻訳精度の向上を示す。

自己回帰ニューラル機械翻訳ではフレーズベース統計的機械翻訳とは異なり、事前並び替えを直接使用して訓練および翻訳を行うと翻訳精度が低下することが知られている [21, 43]。そのため、語順の情報を自己回帰ニューラル機械翻訳モデルにおいて活用する手法が提案されてきた [11, 12, 21, 107, 109]。特に Zhao ら [109] は注意機構付き RNN モデルにおいて事前並び替えのインデックスのベクトル表現を加えることで翻訳精度が向上したことを報告している。しかし現在の機械翻訳モデルのスタンダードである Transformer における事前並び替えの効果は定かではない。第 3 章で提案する事前並び替え位置表現 [44] は Transformer において事前並び替えの情報を活用した初めての事例である。翻訳評価実験では事前並び替えのイ

ンデックスのベクトル表現を使用することで翻訳精度が向上することを示す。

非自己回帰ニューラル機械翻訳ではエンコーダの出力をそのままの順序でデコーダの入力に使用するため、特に英日対のように大きく語順が異なる言語対では翻訳精度が大きく低下してしまう。Ran et al. [77] は非自己回帰ニューラル機械翻訳モデルにおいて並び替えられた潜在変数を予測することで翻訳精度が向上することを示しているが事前並び替えは使用していない。そこで第 4 章では非自己回帰ニューラル機械翻訳における事前並び替えの利用方法の検証を行い、事前並び替えが非自己回帰ニューラル機械翻訳における翻訳精度に与える影響を調査し分析を行う。三つの検証手法による翻訳評価実験を行い、非自己回帰ニューラル機械翻訳では事前並び替えによって翻訳精度が向上することを示す。

1.4 本論文の構成

本論文は 5 章で構成されている。第 2 章ではフレーズベース統計的機械翻訳における人手での特微量設計が不要な RvNN を使用した事前並び替え手法を提案する。まず事前並び替えのための正解ラベル付与を行う手法について説明し、RvNN の構築を説明する。提案手法を適用し翻訳評価実験を行った結果、人手での特微量設計が必要な先行手法である BTG に基づく事前並び替え手法と同等の翻訳精度を達成したことを述べる。また、構文解析器に基づく事前並び替え手法の先行研究である Hoshino et al. の手法と比較し、提案手法の有効性を確認する。さらに、事前並び替えの精度がフレーズベース統計的機械翻訳に与える影響について分析を行う。

第 3 章では自己回帰ニューラル機械翻訳モデルの一つである Transformer における事前並び替え位置表現を使用した翻訳手法を提案する。提案手法である事前並び替え位置表現とその適用方法について説明し、翻訳評価実験を行うことで提案手法の有効性を確認する。また、事前並び替えの精度が Transformer の翻訳精度に与える影響について分析を行う。さらに提案手法が翻訳に与える影響について詳細に調査を行うため、訳抜けと重複訳に与える影響および文長と翻訳精度の関係について詳細な分析を行う。

第 4 章では非自己回帰ニューラル機械翻訳モデルにおける事前並び替えの検証を行う。非自己回帰ニューラル機械翻訳モデルではデコーダの入力としてエンコー

ダの出力をそのままの順序で使用するため、語順の相違を考慮した翻訳が出来ない。そこで事前並び替えが非自己回帰ニューラル機械翻訳における翻訳精度に与える影響を確かめるため、事前並び替えを適用した文をそのまま入力として使用する方法、エンコーダの出力を並び替える方法、エンコーダの入力時に事前並び替えの情報を使用する三つの方法を翻訳評価実験を通して評価する。また、事前並び替えが翻訳精度に与える影響について詳細な調査を行うため、文長と翻訳精度の関係について分析する。第5章では以上の成果についてまとめ、今後の展望について議論する。

第2章 統計的機械翻訳のための RvNN による事前並び替え

2.1 はじめに

フレーズベース統計的機械翻訳モデル [50] は、フレーズを翻訳単位として機械翻訳を行う手法である。この手法ではあるフレーズを翻訳した後、次に翻訳するフレーズを前後数フレーズのうちから選ぶため、英語とフランス語のように語順が似ている言語対や短い文においては高品質な翻訳を行えることが知られている。しかし、英語と日本語のように語順が大きく異なる言語対では、前後数フレーズを考慮するだけでは次に翻訳するべきフレーズを正しく選択することは難しいため、翻訳精度が低い。このような語順の問題に対し、Nakagawa の提案した BTG に基づく事前並び替え手法 [66] は最高性能を達成しているが、並び替えの学習のために人手による素性テンプレートの設計が必要である。

そこで、本章では統計的機械翻訳のための RvNN [85] を用いた事前並び替え手法を提案する。ニューラルネットワークによる学習の特徴として、人手による素性テンプレートの設計が不要であり、訓練データから直接素性ベクトルを学習出来るという利点がある。また、RvNN は木構造の再帰的ニューラルネットワークであり、事前並び替えに適用することで長距離の並び替えが容易に行える。提案手法では与えられた構文木にしたがって RvNN を構築し、葉ノードからボトムアップに計算を行っていくことで、各節ノードにおいて、並び替えに対して重要であると考えられる部分木の単語や品詞・構文タグを考慮した並び替えを行う。

現在ではニューラル機械翻訳が主流となっているが、統計的機械翻訳をベースにすることで、事前並び替えのような中間プロセスに注目した手法の性能が翻訳全体に与える影響について明らかに出来る利点がある。また統計的機械翻訳のようにホワイトボックス的なアプローチは、商用翻訳においてシステムの修正やアップデートが容易であるという利点もある。さらに統計的機械翻訳とニューラル機械翻訳を組み合わせることで性能を向上するモデルが先行研究 [100] により提案されており、

統計的機械翻訳の性能を向上させることは有益である。

本章は以下のように構成されている。2.2 節では、関連研究について説明を行う。2.3 節では本章で使用する機械翻訳手法であるフレーズベース統計的機械翻訳と注意機構付き RNN モデルについて説明する。2.4 節では提案手法である RvNN を使用した事前並び替え手法における正解ラベルの付与の手法と、事前並び替えモデルの構造及び構築方法を説明する。2.5 節では英日、英仏、英中の言語対を使用した翻訳評価実験について述べ、2.6 節で実験の分析を行う。2.7 節で本章のまとめと今後の課題を検討する。

2.2 関連研究

本節では、提案手法と関連の深い事前並び替えに関する既存研究について詳細に議論する。Collins et al. [15] や Gojun and Fraser [25]、Wang et al. [97] は並び替えのルールを定め、そのルールにしたがって事前並び替えを行っている。Xu et al. [104] や Isozaki et al. [37] は木構造に対して並び替えのルールを定め、SVO 型言語である英語から SOV 型言語への翻訳における事前並び替えを行っている。これらの手法のように、ある言語対について並び替えのルールを定めるためには原言語と目的言語についての知識が必要である。また、全ての言語対において並び替えのルールを定めることは難しい。単純に特定の言語をピボットとする手法では、原言語の語順からピボットとする言語の語順に並び替えるルール、ピボットとする言語の語順から目的言語の語順に並び替えるルールが必要であるが、並び替え後の原言語文の単語やフレーズがピボットとする言語の単語やフレーズとの部分に対応するかを特定しなければ並び替え後の原言語文をピボットとする言語と同じルールで並び替えることが出来ないため、ルールに基づいた並び替えでは適用可能な言語対が限られてしまう。

そこで、対訳コーパスから自動で並び替えを学習する手法も提案されている。Zhang et al. [108] や Crego and Habash [17] は n -gram からなるチャンクに対して、並び替えルールを対訳コーパスから得る手法を提案している。Crego and Mariño [18] や Tromble and Eisner [93] は品詞タグを用いた事前並び替え手法を提案している。また Visweswariah et al. [96] は単語をノードとしたグラフを作成し、巡回セールスマントリップルとして並び替えの問題を定式化している。

木構造を用いることで長距離の部分フレーズの並び替えが容易に行えるという利点があるため、木構造を用いる手法も多く提案されている。Xia and McCord [103] や Genzel [23] は木構造から並び替えパターンを抽出し、これを原言語文に適用することで並び替えを行っている。機械学習を用いて並び替えを学習する手法も提案されている。Li et al. [55] は構文木での各ノードにおいて最大エントロピーモデルを用いて、子ノードが三つ以内のノードに限定して学習および並び替えを行うモデルを提案している。Lerner and Petrov [54] は依存木に対して、子ノードが七つ以内のノードに対して並び替えを行う手法を提案している。Yang et al. [105] は並び替えを子ノードの順序を求める順序問題とし、Ranking-SVM [40] を使用して子ノードの順序を求ることで並び替えを行っている。

木構造を用いて各ノードにおける子ノードの順序を決定するようなモデルでは、子ノードの数が多くなるにつれて並び替え候補が爆発的に増加するという問題がある。そこで、木構造を2分木に限定することで、各ノードにおいて子ノードを並び替えるか否かという二値分類の問題として定義出来る。Jehl et al. [39] は2分木に対して、単語アライメントの交差が少なくなるようにロジスティック回帰モデルを用いて並び替えを学習する手法を提案している。単語アライメントは原言語文の単語と目的言語文の単語の対応関係を表しており、原言語文の単語のあるペアを考えたとき、対応している単語の目的言語文内の順序が原言語文の単語の順序と異なる場合に単語アライメントが交差していると定義している。Hoshino et al. [34] は、2分木に対して順位相関係数であるケンドールの τ [45] が最大となるように二値分類の分類器を用いて各ノードでの並び替えを行っている。DeNero and Uszkoreit [20] は構文解析をしつつ同時に並び替えも学習する手法を提案している。Neubig et al. [68] は BTGに基づいて構文木の構築および並び替えを行う手法を提案しているが、計算量が多く時間がかかるという問題があった。本研究で比較を行う BTG に基づく事前並び替え手法 [66] は計算量の問題を解決した手法であり、翻訳において本論文執筆時点で最高性能を達成している。この手法は入力として与えられた文を構文解析しながら同時に並び替える手法であり、latent variable perceptron [89] によって単語、単語クラスタ、品詞タグ、親ノードの並び替えラベルから素性テンプレートに従って選択された素性に従い、並び替え後の単語アライメントの交差がなくなるようにモデルのパラメータを訓練する。単語クラスタは各単語に対して周辺単語を元にクラスタリングを行ったものを指す。素性はモデルの

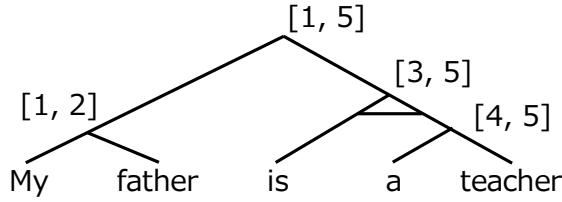


図 2.1 BTG に基づく並び替えモデルによる “My father is a teacher” の構文木の例 (横線はそのノードを入れ替えることを表す)

訓練を行う際に注目する値のことを指す。素性テンプレートは、「現在のノードが動詞で、かつ同じスペインに属している一番左の単語が名詞か」のように与えられた情報のどの部分に注目するか、どのように組み合わせるかが記述された表であり、人手によってどの情報を組み合わせるかが設計される。BTG による構文木の各ノードは入力として与えられた文のある一つのスペインに対応している。二単語以上のスペインに対応したノードは二つの子ノードとそれら子ノードを並び替えるかどうかのラベルを持ち、一単語のスペインに対応したノードは終端ノードとしてそのスペインに対応した単語に紐づけられる。図 2.1 に BTG による構文木の例を示す。 $[l, r]$ は文の l 番目から r 番目までの単語を覆う節ノードであることを表している。BTG に基づく並び替え手法では各節ノードに対して *Inverted (I)* か *Straight (S)* のどちらのラベルを付与するかをトップダウンで決定していく。*Inverted (I)* ラベルは子ノードの順序を入れ替えることを表しており、*Straight (S)* は子ノードの順序を維持することを表している。各節ノードではまずそのノードが覆うスペインのどこで二つに分割するかを決定し、その後どちらのラベルを付与するかを決定する。すべてのノードにおいてラベルが付与された後、各ノードに付与されたラベルに従ってノードを並び替えていくことで事前並び替えを実現する。推論時は以下の式に従って構文解析およびラベルを決定する。

$$\hat{\xi} = \arg \max_{\xi \in \Xi(\mathbf{x})} \sum_{\eta \in \text{Nodes}(\xi)} \Lambda \cdot \Phi(\eta)$$

$$\mathbf{x}' = \text{Proj}(\hat{\xi})$$

$\Phi(\eta)$ は BTG による構文木のノード η の素性ベクトルを返す関数であり、 Λ は各素性に対する重みを表すベクトルである。素性ベクトルは各次元がそれぞれ別の素性に対応したベクトルであり、素性が存在しているなら 1 を、そうでなければ 0

となるベクトルである。例えば素性として、与えられた単語が「名詞かどうか」、「動詞かどうか」を考える。この時、「素性」という単語は名詞であるが動詞ではないため、素性ベクトルは $[0, 1]$ として表される。素性をどのように選択するかを表す素性テンプレートはモデルの設計時に人手で設計される必要があるが、言語やドメインによっては新たに素性テンプレートを設計しなければならない。本研究では 2.4 節で説明するように、ニューラルネットワークを使用することで最適な情報の組み合わせ方が決定される点で BTG に基づく事前並び替え手法と異なる。 $\exists(x)$ は BTG によって到達可能な文 x の構文木の集合であり、 $Nodes(\xi)$ は構文木 ξ のノードの集合、 $Proj(\xi)$ は BTG の構文木 ξ から並び替えられた文 x' を生成する関数である。

近年では、素性テンプレートの設計を必要としないニューラルネットワークに基づいた手法も提案されている。de Gispert et al. [19] は FFNN を用いた 2 分木での並び替えを提案している。Botha et al. [6] も FFNN を用いた並び替えを提案しているが、木構造を用いずに並び替えを行っている。Miceli-Barone and Attardi [60] は RNN を用いて依存木のノードを辿ることで並び替え手法を提案している。彼らは、単語を出力する “EMIT”、親ノードへ移動する “UP”、 j 番目のノードへ移動する “DOWN $_j$ ” の三つの動作を定義し、RNN で動作を予測することで並び替えを行う。Kanouchi et al. [42] は統計的機械翻訳の翻訳モデルにより抽出するフレーズペアについて、RvNN を用いて並び替えラベルを推定することで、翻訳システムの内部で翻訳と同時にフレーズの並び替えを行う。提案手法と同様 RvNN を使用しているが、1.2 節で議論したように翻訳と同時に並び替えを考慮する手法は長距離の並び替えを考慮しようとすると計算量が爆発してしまう。提案手法は翻訳システムとは独立して事前並び替えを行う点で異なり、構文木を使用することで長距離の並び替えがより簡単に行えるという利点がある。また提案手法では原言語の構文木に対してボトムアップに構築される RvNN を用いて 2 種類のラベルの予測を行うことで、部分木全体を考慮した並び替えを行える。

2.3 前提知識

本節では事前並び替えを適用する機械翻訳手法について説明する。2.3.1 節では本章で事前並び替えを適用する機械翻訳手法であるフレーズベース統計的機械翻訳

について説明し、2.3.2 節では事前並び替えを適用する機械翻訳手法である注意機構付き RNN モデルについて説明する。

2.3.1 フレーズベース統計的機械翻訳

本節では、事前並び替えを適用する機械翻訳手法として使用する、統計的機械翻訳の一つであるフレーズベース統計的機械翻訳モデル [50] について説明する。統計的機械翻訳は雑音のある通信路モデル [81] に基づいて \mathbf{x} に対する最適な翻訳文 $\hat{\mathbf{y}}$ を決定する手法である。

$$\begin{aligned}\hat{\mathbf{y}} &= \arg \max_{\mathbf{y} \in \mathcal{E}(\mathbf{x})} Pr(\mathbf{y}|\mathbf{x}) \\ &= \arg \max_{\mathbf{y} \in \mathcal{E}(\mathbf{x})} Pr(\mathbf{x}|\mathbf{y})Pr(\mathbf{y})\end{aligned}\quad (2.1)$$

ここで $\mathcal{E}(\mathbf{x})$ は原言語文 \mathbf{x} の可能な翻訳文の集合を表す。 $Pr(\mathbf{x}|\mathbf{y})$ は翻訳モデルと呼ばれ翻訳の確からしさを表しており、 $Pr(\mathbf{y})$ は言語モデルと呼ばれ文の確からしさを表している。機械翻訳において一般的に使用される言語モデルは n -gram¹⁾ 言語モデルと呼ばれるものである。 n -gram 言語モデルは各単語の確率をその前に出現した $n - 1$ 個の単語の条件付き確率として計算するモデルであり、式 (2.2) のように計算される。

$$\begin{aligned}Pr(\mathbf{y}) &= \prod_{i=1}^m p_{ml}(\mathbf{y}_i | \mathbf{y}_{i-n+1}^{i-1}) \\ p_{ml}(\mathbf{y}_i | \mathbf{y}_{i-n+1}^{i-1}) &= \frac{C(\mathbf{y}_{i-n+1}^i)}{C(\mathbf{y}_{i-n+1}^{i-1})}\end{aligned}\quad (2.2)$$

ここで \mathbf{y}_i^j は文 \mathbf{y} の i 番目から j 番目の単語列を表す。 $C(\mathbf{y}_i^j)$ は単語列 \mathbf{y}_i^j がデータ中で出現する回数を表す。

フレーズベース統計的機械翻訳モデルは単語の代わりに、連続した単語列であるフレーズというまとまりを基本単位として翻訳を行う手法である。フレーズベース統計的機械翻訳モデルでは以下の翻訳モデルの確率を最大化する。

1) n -gram は n 個の連続した単語の連なりを表す。

$$\begin{aligned}
Pr(\mathbf{x}|\mathbf{y}) &= \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}} \sum_{\phi,\alpha} Pr(\mathbf{x},\phi,\alpha|\mathbf{y}) \\
&\approx \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}} \sum_{\phi,\alpha} p_d(\mathbf{x},\alpha|\phi) p_\phi(\phi|\mathbf{y}) \\
&\approx \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}} \sum_{\phi,\alpha} p_d(\mathbf{x},\alpha|\phi) \prod_{k=1}^L p_\phi(\bar{\mathbf{x}}(\phi_k)|\bar{\mathbf{y}}(\phi_k))
\end{aligned}$$

ここで \mathcal{D} は原言語文と目的言語文のペアの集合を表す。 $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ はそれぞれ原言語側と目的言語側のフレーズを表し、 $p_\phi(\bar{\mathbf{x}}(\phi_k)|\bar{\mathbf{y}}(\phi_k))$ は k 個目のフレーズの、フレーズ単位の翻訳モデルを表す。 ϕ は原言語文と目的言語文の対訳フレーズ、 α はフレーズ単位のアライメントである。 $p_\phi(\phi|\mathbf{y})$ はフレーズ翻訳モデルと呼ばれ、フレーズへの分割と対訳フレーズの生成を担当する。 $p_d(\mathbf{x},\alpha|\phi)$ はフレーズ歪みモデルと呼ばれ、フレーズの並び順を決定し \mathbf{x} を生成する。推論時は式 (2.1) を最大化するような翻訳文が生成される。推論時は可能な全てのフレーズペアを列挙すると計算量が爆発してしまうため、フレーズベース統計的機械翻訳では歪み制約と呼ばれる、最後に翻訳を決定した原言語文のフレーズの最後の単語の位置と次に翻訳すべき原言語文のフレーズの最初の位置の距離の差に上限を設けることで計算量爆発を回避している。

2.3.2 注意機構付き RNN モデル

本節では、事前並び替えを適用する機械翻訳手法として使用する、注意機構付き RNN モデルについて説明する。Luong et al. [57] によって提案された注意機構付き RNN モデルは、Long Short-Term Memory (LSTM) [33] を使用してベクトル表現を計算しそのベクトル表現に基づいて翻訳文の出力を行う。図 2.2 に注意機構付き RNN モデルの全体図を載せる。この例では原言語文 “My father is a teacher” から翻訳文「私の父親は教師です」を出力する例である。²⁾

注意機構付き RNN モデルはハイパーパラメータ L によって決められた回数 RNN をスタッツしたモデルであり (図 2.2 は $L = 2$ の例)、第 l 層における k 番目の単語のベクトル表現 $\mathbf{h}_k^l \in \mathbb{R}^{d_h}$ (d_h はベクトル表現の次元数) は式 (2.3) に従って計算される。

2) “<bos>” と “<eos>” はそれぞれ文の先頭と末尾を表すタグである。

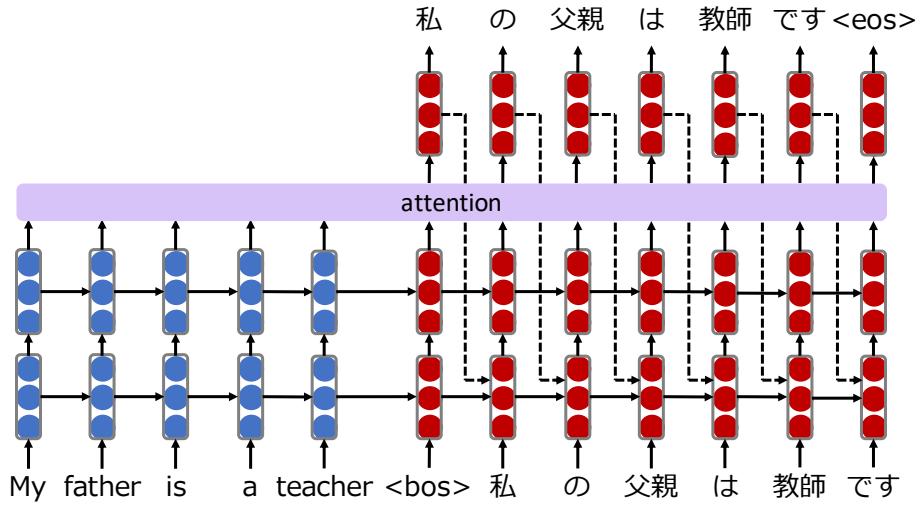


図 2.2 注意機構付き RNN モデルの全体図

$$\mathbf{h}_k^l = f(\mathbf{h}_{k-1}^l, \mathbf{h}_k^{l-1}) \quad (2.3)$$

f は非線形関数であり、Luong et al. [57] と同様に本論文でも LSTM を使用した。また 0 番目の層は存在しないため、 \mathbf{h}_k^0 として k 番目の単語のベクトル表現を使用する。 j 番目の単語 y_j はデコーダの最終的なベクトル表現 $\tilde{\mathbf{h}}_j \in \mathbb{R}^{d_h}$ に基づいて決定される。 $\tilde{\mathbf{h}}_j$ はデコーダの最後の層のベクトル表現 $\bar{\mathbf{h}}_j^L$ と文脈ベクトル $\mathbf{c}_j \in \mathbb{R}^{d_h}$ から式 (2.4) に従って計算される。

$$\begin{aligned} \tilde{\mathbf{h}}_j &= \tanh(W_c \cdot \text{Concat}(\mathbf{c}_j, \bar{\mathbf{h}}_j^L)) \\ \mathbf{c}_j &= \sum_{i=1}^n a_{ij} \hat{\mathbf{h}}_i^L \\ a_{ij} &= \frac{\exp((\hat{\mathbf{h}}_i^L)^T W_a \bar{\mathbf{h}}_j^L)}{\sum_{k=1}^n \exp((\hat{\mathbf{h}}_k^L)^T W_a \bar{\mathbf{h}}_j^L)} \end{aligned} \quad (2.4)$$

$\hat{\mathbf{h}}_i^l \in \mathbb{R}^{d_h}$ はエンコーダの第 l 層における i 番目の単語のベクトル表現、 $\text{Concat}(\cdot)$ は受け取ったベクトルをその順に結合したベクトルを返す関数である。 $W_c \in \mathbb{R}^{d_h \times 2d_h}, W_a \in \mathbb{R}^{d_h \times d_h}$ は重み行列、 a_{ij} はデコーダの j 番目に対するエンコーダの i 番目の重みである。このようなあるベクトル表現への重みは attention と呼ばれる。この attention によって次の単語の予測時にエンコーダのどの単語のベク

トル表現に注目するかを考慮してベクトル表現の計算を行い、そのベクトル表現に基づいて次に出力すべき単語を決定する。

2.4 提案手法

本節では提案手法である RvNN による事前並び替え手法を説明する。まず本手法で使用する単語アライメントについて説明を行う。その後、節ノードにおける正解ラベル付与のアルゴリズムについて述べ、続いて RvNN を用いた事前並び替えモデルについて述べる。

2.4.1 単語アライメント

単語アライメントとは対訳文における単語単位の対応関係を表した集合である。図 2.3 に英語から日本語への単語アライメントの例を示す。対訳文において愚直に原言語文と目的言語文の単語のペアを考慮すると、全てのペア $n \times m$ 通りに対して対応づけるかどうかの $2^{n \times m}$ 通りの候補を探索する必要がある。そこで IBM モデル [8] では原言語文側から目的言語文側への対応は最大一単語という制約を加えることで計算量を大幅に削減しており、単語アライメントは原言語文の各単語に対応する目的言語文の単語の位置として表現することが出来る。例として図 2.3 における英語から日本語方向への単語アライメントは、それぞれの英語文の単語に対応づけられている日本語文の単語の位置を並べた $\mathbf{a} = (1, 3, 7, 6, 5, 9)$ として表される。ここで “My” は「私」と「の」に対応づけられているが、原言語文側から目的言語文側への対応は最大一単語という制約によって「私」のみへのアライメントとなっている。また、アライメントが存在しない場合は 0 番目に NULL の単語が存在すると仮定し 0 へと対応づけられる。原言語文の単語 \mathbf{x}_j の単語アライメント \mathbf{a}_j のアライメントモデル p_a は以下のように表される。

$$\begin{aligned} Pr(\mathbf{a}_j | \mathbf{x}_1^{j-1}, \mathbf{a}_1^{j-1}, \mathbf{y}) &\approx p_a(\mathbf{a}_j | \mathbf{a}_{j-}, \mathbf{x}_1^{j-1}, \mathbf{y}) \\ j_- &= \arg \max_{j'} j' < j \wedge \mathbf{a}_{j'} \neq 0 \end{aligned} \tag{2.5}$$

$\mathbf{x}_1^{j-1}, \mathbf{a}_1^{j-1}$ はそれぞれ原言語文の最初から $j - 1$ 番目までの単語と単語アライメントを表す。アライメントモデルによって単語アライメントは前から順に決定さ

My									
parents			■						
live						■■■			
in					■				
London				■					■
.	私	の	両親	は	ロンドン	に	住んで	いる	。

図 2.3 “My parents live in London.” と「私の両親はロンドンに住んでいる。」の単語アライメントの例

れる。

2.4.2 正解ラベルの付与

事前並び替えは翻訳器を通した後の文で評価を行うため、翻訳に最適な並び替えを行った文を人手で得ることは難しい。そこで Nakagawa は単語アライメントに基づいて並び替えラベルを決定し正解ラベルとして付与している。提案手法でも同様に、単語アライメントに基づいて並び替えの正解ラベルを付与する。2 分木である句構文木の各節ノードにおいて子ノードの順序を入れ替えるかどうかのラベル付けをケンドールの τ に基づいて行い、並び替えの訓練データを作成する。Algorithm 1 に正解ラベル付与の擬似コードを示す。入力は 2 分木のノード n と単語アライメント a である。ノードは左右の子ノードへのリンク (left および right) と、並び替えを示すラベル (label) を保持する。 $a[n]$ は、葉ノードにおける単語のアライメントの目的言語におけるインデックスを表す。並び替えのラベルは関数 $KendallTau(a_l, a_r)$ によってケンドールの τ の値を計算し、その結果に基づいて決定する。各節ノードにおいて、子ノードを並び替えた際にケンドールの τ が大きくなる場合は並び替えを行う “Inverted” ラベルを、小さくなるまたは変わらない場合はそのままの順序を維持する “Straight” ラベルを付与する。これをボトムアップで行うことで、与えられた構文木においてケンドールの τ が最大となるように正解ラベルが付与される。

Algorithm 1 正解ラベル付与の擬似コード

Input: 2文木のノード n 、単語アライメント a

Output: ラベルが付与された2分木

```
1: function LABELING( $n, a$ )
2:   if  $n \neq \text{NULL}$  then
3:     if  $n.\text{left} \neq \text{NULL}$  and  $n.\text{right} \neq \text{NULL}$  then
4:        $N_l, a_l \leftarrow \text{LABELING}(n.\text{left}, a)$ 
5:        $N_r, a_r \leftarrow \text{LABELING}(n.\text{right}, a)$ 
6:       if KendallTau( $a_l, a_r$ ) < KendallTau( $a_r, a_l$ ) then
7:          $N \leftarrow \{\text{"left": } N_l, \text{"right": } N_r, \text{"label": "Inverted"}\}$ 
8:          $Align \leftarrow a_r \cup a_l$ 
9:       else
10:         $N \leftarrow \{\text{"left": } N_l, \text{"right": } N_r, \text{"label": "Straight"}\}$ 
11:         $Align \leftarrow a_l \cup a_r$ 
12:      end if
13:      return  $N, Align$ 
14:    else
15:      return  $n, a[n]$ 
16:    end if
17:  end if
18: end function
```

2.4.3 事前並び替えモデル

RvNN は木構造型のニューラルネットワークである [85]。RvNN における各ノードは図 2.4 に示す構造を持ち、これを再帰的に結合することで、図 2.5 に示すような木構造型のニューラルネットワークを構築する。

提案手法では句構文木にしたがって RvNN を構築し、各節ノードにおいて Algorithm 1 で付与した正解ラベルを予測する学習を行う。Algorithm 2 に予測したラベルを用いた並び替えの擬似コードを示す。入力は Algorithm 1 によって正解ラベルが付与された2分木のノード n であり、左右の子ノードへのリンク (left および right) と並び替えの予測結果を保持する label を持つ。len(\cdot) は要素の数を計算する。葉ノードはさらに自身の単語 word とそのベクトル表現 e を保持する。 S_l 、 S_r は並び替え後の単語列である。節ノードにおいては、左と右の子ノードから、それぞれのベクトル表現 v_l 、 v_r を入力とし、関数 $\text{RvNN}(v_l, v_r)$ によって自身のベクト

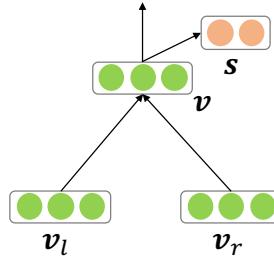


図 2.4 RvNN の基本となる構造

Algorithm 2 RvNN による並び替えの擬似コード

Input: 正解ラベルが付与された 2 分木のノード n
Output: 並び替えを適用した文、ノードのベクトル表現

```

1: function PREORDERING( $n$ )
2:   if  $\text{len}(n.\text{children}) = 2$  then
3:      $S_l, \mathbf{v}_l \leftarrow \text{PREORDERING}(n.\text{left})$ 
4:      $S_r, \mathbf{v}_r \leftarrow \text{PREORDERING}(n.\text{right})$ 
5:      $\text{Label}, \mathbf{v} \leftarrow \text{RvNN}(\mathbf{v}_l, \mathbf{v}_r)$ 
6:     if  $\text{Label} = \text{"Straight"}$  then
7:       return  $S_l + S_r, \mathbf{v}$ 
8:     else
9:       return  $S_r + S_l, \mathbf{v}$ 
10:    end if
11:  else
12:    return  $n.\text{word}, n.e$ 
13:  end if
14: end function

```

ルの計算とラベルの予測を行う。 $\text{RvNN}(\mathbf{v}_l, \mathbf{v}_r)$ はベクトル表現 $\mathbf{v}_l, \mathbf{v}_r$ を受け取り、予測されたラベル Label と節ノードのベクトル \mathbf{v} を返す関数である。左の子ノードと右の子ノードのベクトル表現を用いてラベルを予測することで、部分木を考慮しつつ並び替えを行うかどうかを決定出来る。

図 2.5 に “My parents live in London” という文に対して RvNN を用いた並び替えの例を示す。例えば “live in London” のフレーズに対応したノードにおいて、式 (2.6) に従い “live” と “in London” の子ノードを考慮してベクトルを計算する。

$$\mathbf{v} = f(\text{Concat}(\mathbf{v}_l, \mathbf{v}_r)W + \mathbf{b}) \quad (2.6)$$

$$\mathbf{s} = \mathbf{v}W_s + \mathbf{b}_s \quad (2.7)$$

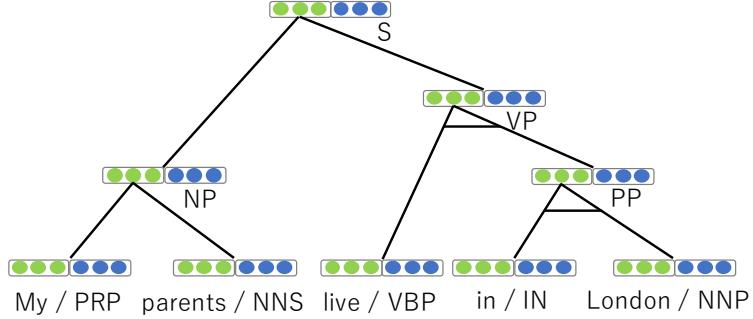


図 2.5 “My parents live in London” の RvNN による並び替え (横線が引いてある節ノードは “Inverted”)。緑色はノードのベクトル表現を表し、青色は品詞・構文タグのベクトル表現を表す。

f は ReLU 関数、 $W \in \mathbb{R}^{\lambda \times 2\lambda}$ は重み行列、 $\mathbf{v}_l, \mathbf{v}_r \in \mathbb{R}^\lambda$ はそれぞれ左、右の子ノードのベクトル、 $W_s \in \mathbb{R}^{2 \times \lambda}$ は出力層における重み行列、 $\mathbf{b} \in \mathbb{R}^\lambda$ 、 $\mathbf{b}_s \in \mathbb{R}^2$ はバイアス項を表す (λ は隠れ層の次元数を表す)。 $s \in R^2$ は各ラベルに対する重みのベクトルであり、式 (2.8) に示すソフトマックス関数に入力することで “Straight” および “Inverted” ラベルの確率を計算する。

$$p_i = \frac{\exp(s_i)}{\sum_{m=1}^{|s|} \exp(s_m)} \quad (2.8)$$

$|\cdot|$ はベクトルの次元数を表し、ここでは $|s| = 2$ である。

葉ノードでは、単語ベクトルを入力とし、式 (2.10) によってベクトル表現を得る。

$$\mathbf{e} = \mathbf{x}W_x \quad (2.9)$$

$$\mathbf{v}_e = f(\mathbf{e}W_e + \mathbf{b}_e) \quad (2.10)$$

ここで $\mathbf{x} \in \mathbb{R}^V$ は入力単語を表す one-hot ベクトル、 $W_x \in \mathbb{R}^{V \times \lambda}$ は単語ベクトル表現を表す行列 (V は語彙数を表す)、 $W_e \in \mathbb{R}^{\lambda \times \lambda}$ は重み行列、 $\mathbf{b}_e \in \mathbb{R}^\lambda$ はバイアス項である。ロス関数は式 (2.11) で定義される交差エントロピーを用いる。

$$L(\theta) = -\frac{1}{K} \sum_{k=1}^K \sum_{n \in \mathcal{T}} \log p(l_k^n; \theta) \quad (2.11)$$

θ はモデルのパラメータ、 n は構文木 \mathcal{T} のノードであり、 K はミニバッチのサイズ、 l_k^n はミニバッチの k 番目の構文木の n 番目のノードのラベルを表す。

本研究では、各ノードにおける品詞もしくは構文タグを考慮する手法も提案する。これらを考慮する際は、式 (2.6) に代わり式 (2.12) を使用する。

$$\mathbf{v}_t = f([\mathbf{v}_l; \mathbf{v}_r; \mathbf{e}_t]W_t + \mathbf{b}_t) \quad (2.12)$$

$\mathbf{e}_t \in \mathbb{R}^\lambda$ は品詞・構文タグの情報を表現するベクトルで、各ノードの品詞または構文タグを表す one-hot ベクトルを入力とし、式 (2.9) と同様に計算する。 $W_t \in \mathbb{R}^{3\lambda \times \lambda}$ は重み行列、 $\mathbf{b}_t \in \mathbb{R}^\lambda$ はバイアス項である。式 2.10 および式 (2.12) における重み行列 W_e, W_t の値が訓練時に更新されることで入力として与えられた情報からどの情報をどの程度の重みで使用するかが決定されるため、人手での素性テンプレート設計の必要がなくなる。

2.5 翻訳評価実験

本節では事前並び替えを用いた翻訳評価実験について述べる。まず初めに実験設定について説明し、次に実験結果を示す。実験ではまず提案手法と BTG に基づく事前並び替えの精度を評価し、それから事前並び替えを使用した翻訳の評価を行うためにフレーズベース統計的機械翻訳による翻訳精度の評価を行う。それぞれの機械翻訳モデルが出力した翻訳文の評価値の統計的有意差を検証するため、ブートストラップによる検定 [48] を行う。

2.5.1 実験設定

コーパスの前処理 英日、英仏、英中対において原言語文の事前並び替えを行い、その上で機械翻訳システムを訓練し、翻訳精度を評価する。英日翻訳は ASPEC コーパス [67] を使用した。ASPEC コーパスは、科学技術論文の概要から Utiyama and Isahara [94] の提案した文間の類似度に基づいて対応づけられている約 300 万文からなる対訳コーパスである。本研究では上位 50 万文対から 10 万文対をサンプリングして事前並び替えの訓練データとした。英仏翻訳は Common Crawl コーパス [5] を使用し、英中翻訳は IWSLT コーパス [9] を使用した。英仏、英中とともに、訓練データから無作為にサンプリングした 10 万文を並び替えの訓練データとした。翻訳の訓練データは全ての言語対において、先行研究 [57, 66] と同様翻訳

表 2.1 翻訳の学習に用いたデータの統計量 (文対)

	訓練データ	開発データ	テストデータ
英日 (ASPEC)	1,805,583	1,790	1,812
英仏 (Common Crawl)	2,652,425	3,000	3,003
英中 (IWSLT15)	198,718	887	1,261

器の学習には原言語文、目的言語文とともに 50 単語以下で、文対の単語数の比は Moses の前処理スクリプト³⁾のデフォルト値である 9 以下の条件を満たす文対を用いた。表 2.1 にそれぞれの言語対の実験で使用した文対数を載せる⁴⁾。

英語文は Stanford CoreNLP⁵⁾[59] で単語分割と品詞タグ付けを、Enju⁶⁾[62] で構文解析を行った。日本語文は MeCab⁷⁾[52] で形態素解析を、CkyLark⁸⁾[73] で構文解析を行った。フランス語文は Moses に付属しているスクリプト⁹⁾で単語分割を行い、Berkeley Parser¹⁰⁾[75] で構文解析を行った。中国語文は KyotoMorph¹¹⁾を用いて単語分割を行い、Berkeley Parser で構文解析を行った。KyotoMorph の訓練には CTB version 5 (CTB5) と SCTB [14] を用いた。並び替えの学習における単語アライメントは MGIZA¹²⁾[72] を使用し、IBM Model1 と hidden Markov model [8] をそれぞれ 3 回繰り返して原言語文から目的言語文方向とその逆方向のアライメントをそれぞれ計算した。この時の単語クラスタリングのサイズは、先行研究 [66] にしたがって 256 とし、クラスタリングは mkcls¹³⁾[70] を使用した。両方向のアライメントの積集合をとったものを最終的な単語アライメントとした。

3) <https://github.com/moses-smt/mosesdecoder/blob/master/scripts/training/clean-corpus-n.perl>

4) ASPEC コーパスはノイズが多く、全てを使用して訓練した場合の翻訳精度よりも上位 200 万文を使用して訓練した場合の翻訳精度の方が高い。そのため ASPEC を使用した研究の多くは上位 100 万文から 200 万文を使用しているものが多く、本研究でも類似度の高い順から上位 200 万文に対して前処理を行ったものを翻訳器の訓練に使用した

5) <http://stanfordnlp.github.io/CoreNLP/>

6) <https://github.com/mynlp/enju>

7) <http://taku910.github.io/mecab/>

8) <https://github.com/odashi/CkyLark>

9) <https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl>

10) <https://github.com/slavpetrov/berkeleyparser>

11) <https://bitbucket.org/msmoshen/kyotomorph-beta>

12) <https://github.com/moses-smt/mgiza>

13) <https://github.com/clab/mkcls>

事前並び替え手法の訓練 提案手法である RvNN は Chainer¹⁴⁾を用いて実装し、語彙は頻度が高いものから 5 万語を用いた。最適化には Adam [47] に重み減衰 (0.0001) および GradientClipping (5) を適用して行った。バッチサイズは 500 文とした。開発データにおけるロス値が最小となったエポック (英日:2、英中・英仏:5) のモデルを使用し、事前並び替えを行う。Nakagawa [66] の BTG に基づく事前並び替え手法を比較対象とし、公開されている実装¹⁵⁾を用いた。訓練には提案手法と同一の前処理を行った 10 万文対の対訳データを用い事前並び替えを行った。

機械翻訳モデルの訓練 統計的機械翻訳器として Moses¹⁶⁾のフレーズベース統計的機械翻訳 (PBSMT) を使用した。訓練データの目的言語文を用いて KenLM¹⁷⁾ [31] を使用し 5-gram 言語モデルを訓練した。フレーズ歪みモデルは Linear モデル [50] を用いた。ハイパーパラメータのチューニングは開発データを用いて MERT [71] で 3 回行った。それぞれの設定でテストデータの翻訳を評価した評価値の平均を最終的な評価値とする。また、注意機構付き RNN モデル (AttRNN) として OpenNMT-py¹⁸⁾を使用した。語彙は原言語、目的言語とともに頻度の上位 5 万語を用い、単語ベクトルの次元数は 500、隠れ層のベクトルの次元数は 500 とした。デフォルトの設定に従い、エンコーダ、デコーダともに 2 層の LSTM を用いた。バッチサイズは 64 文とし、13 エポックの学習を行った。

2.5.2 評価指標

本節では並び替えの評価指標として使用したケンドールの τ [45]、翻訳の評価指標として使用した BiLingual Evaluation Understudy (BLEU 値) [74] と Rank-based Intuitive Bilingual Evaluation Score (RIBES 値) [36] について説明する。

ケンドールの τ ケンドールの τ は式 (2.13) で表される、順位相関係数の一つである。

14) <http://chainer.org/>

15) <http://github.com/google/topdown-btg-preordering>

16) <http://www.statmt.org/moses/>

17) <http://github.com/kpu/kenlm>

18) <https://github.com/OpenNMT/OpenNMT-py>

$$\tau(\mathbf{a}) = \frac{4 \sum_{i=1}^{n-1} \sum_{j=i+1}^n \delta(a_i < a_j)}{n(n-1)} - 1, \quad (2.13)$$

$$\delta(x) = \begin{cases} 1 & (x \text{ is true}), \\ 0 & (\text{otherwise}) \end{cases}$$

ここで \mathbf{a} は長さ n の数列である。数列 \mathbf{a} の要素が完全に昇順に並んでいる場合、 $\tau(\mathbf{a})$ は 1 を、完全に降順であれば -1 をとり、それ以外であれば $-1 < \tau(\mathbf{a}) < 1$ となる。式 (2.13) は \mathbf{a} に含まれる数値のペア a_i と a_j ($i < j$) が昇順 ($a_i < a_j$) になっている割合を -1 から 1 の間に正規化したものであり、この値が大きいほど昇順に並んでいるペアの割合が多い。事前並び替えの評価を行う際は、各対訳文で並び替えた文の単語に対応した目的言語文の単語のインデックスをその順に並べた数列に対してケンドールの τ を計算し、その平均値をケンドールの τ として評価を行う。¹⁹⁾

BLEU BLEU 値は機械翻訳において標準的に使用されている翻訳精度の評価指標であるため、本研究でも使用する。BLEU 値は翻訳文の適合率に基づいた評価指標であり、式 (2.14) のように参照訳 R と翻訳文 E の n -gram ごとの適合率の幾何平均として計算される。本論文では一般的に使用されている、式 (2.14) に示すような 4-gram までの適合率で BLEU 値を計算する。

$$\text{BLEU}(R, E) = \prod_{n=1}^4 \left(\frac{\text{M}_n(R, E)}{\text{C}_n(E)} \right)^{\frac{1}{4}} \cdot \text{BP}(R, E) \quad (2.14)$$

$$\text{BP}(R, E) = \min \left\{ 1, \exp \left(1 - \frac{|R|}{|E|} \right) \right\}$$

ここで $\text{C}_n(E)$ は E 中の n -gram の数を表し、 $\text{M}_n(R, E)$ は R と E で重複する n -gram の数を表す。 $\text{BP}(R, E)$ は簡潔ペナルティと呼ばれる値であり、翻訳文 E が短い場合に適合率が高くなることによって BLEU 値が不当に高くなる問題を回避するために用いられる。

RIBES RIBES 値 [36] は適合率に加えて語順も考慮した評価指標であり、特に言語構造の大きく異なる言語間の翻訳に対する評価指標として提案された。RIBES

19) 他の順位相関係数としてスピアマンの順位相関係数が使用されることもあるが、スピアマンの順位相関係数は順位の差の絶対値を基に評価を行う。今回は「並び替えが出来ているか」に注目したいため、順位の差ではなく交差の数に基づいて順位相関係数を計算するケンドールの τ を使用した。

は式 (2.15) のように単語の 1-gram の適合率とその順位相関係数によって計算される。

$$\text{RIBES}(R, E) = \tau(\text{h}(R, E)) \cdot \left(\frac{|\text{h}(R, E)|}{|E|} \right)^\alpha \cdot \text{BP}(R, E)^\beta \quad (2.15)$$

ここで $\text{h}(R, E)$ は翻訳文の単語とアライメントされた参照訳の単語のインデックスを順に並べた数列であり、アライメントの数が 1-gram のマッチした数と一致する。 α , β はハイパーパラメータであり、本論文では通して、著者実装によるデフォルト値であり、人手評価との相関が一番高かった値として報告されている $\alpha = 0.25$ 、 $\beta = 0.10$ を使用し評価を行った。

2.5.3 実験結果

事前並び替え精度の評価指標としてケンドールの τ を、翻訳精度の評価指標として BLEU 値、翻訳と語順の評価指標として RIBES 値を使用する。表には全て 100 倍したものを記載している。

提案手法における品詞・構文タグの効果 RvNN における品詞・構文タグおよび単語ベクトル、節ノードのベクトルの次元数が事前並び替えおよび PBSMT の翻訳精度に与える影響を検証するため、英日対において ASPEC コーパスの上位 50 万文対を用いて実験を行った。表 2.2 に、単語のみを入力とした場合と品詞・構文タグを付与した場合の開発セットにおける BLEU 値を示す。機械翻訳のコンペティションの一つである WAT2017 のベースラインシステム²⁰⁾と同様に並び替えなしのものは歪み制約は 20 とし、BTG と RvNN で並び替えを行った場合の歪み制約は 0 とした。品詞・構文タグがない場合、ベクトルの次元数が 100 の時に比べて、200 の時は BLEU 値が低下しているが、500 の時は向上している。品詞・構文タグを用いた場合、ベクトルの次元数が 100 の時に比べて、200 の時は BLEU 値が向上したが、500 の時は低下している。また、品詞・構文タグを用いないベクトル次元数が 500 の時と、品詞・構文タグを用いたベクトル次元数が 200 の時を比較すると、有意差は見られなかった。以上の結果に基づき、以降の実験では隠れ層のベクトルの次元数をより少ない 200 とし、品詞・構文タグを用いて事前並び替えを行う。

20) <http://lotus.kuee.kyoto-u.ac.jp/WAT/WAT2017/baseline/baselineSystemPhrase.html>

表 2.2 ベクトルの次元数と品詞・構文タグの有無による開発セットにおける BLEU 値の変化 (英日 ASPEC コーパスの 50 万文対で学習)

ノードの次元	100	200	500
並び替えなし		24.37	
品詞・構文タグなし	28.41	28.18	28.79
品詞・構文タグあり	28.55	28.88	28.32

事前並び替えと翻訳精度の関係 事前並び替えの性能および事前並び替えが翻訳結果に与える影響を調査するため、事前並び替えの性能と翻訳精度の関連を分析する。入力文の理想的な事前並び替えが行えると、原言語と目的言語の語順が等しくなる。つまり、並び替えた入力文と参照翻訳の語順が等しくなり、翻訳タスクは逐語翻訳に近づくと考えられる。そのため、並び替えた入力文と参照翻訳の語順の近さを評価するケンドールの τ と、翻訳結果と参照翻訳の語順を評価する RIBES 値には相関があると期待される。また逐語翻訳に近づくことで翻訳タスク自体が簡単になり、BLEU 値も向上すると期待出来る。表 2.3 に、開発セットにおける並び替え前後の入力文それぞれと参照翻訳文間のケンドールの τ と、PBSMT による翻訳結果の BLEU 値、RIBES 値を示す。BTG の行は BTG に基づく事前並び替えによる結果を表し、RvNN の行は RvNN を使用した事前並び替えによる結果を表す。英日対において、RvNN は並び替えなしに比べてケンドールの τ が 27.07 ポイント向上しており、英語、日本語文での語順の一致率を大きく向上出来ている。BLEU 値、RIBES 値もそれぞれ 3.68 ポイント、8.27 ポイント向上している。日英対においても並び替えなしと比較して RvNN による事前並び替えでケンドールの τ が 10.83 ポイント向上し、BLEU 値、RIBES 値もそれぞれ 2.27 ポイント、8.23 ポイント向上している。つまり RvNN では、語順の一致率を大きく向上出来た英日、日英対では、BLEU 値および RIBES 値を改善出来ていることが分かる。一方、仏英、中英対では並び替えなしに比べて RvNN のケンドールの τ がそれぞれ 1.08 ポイント、0.26 ポイント向上したが、BLEU 値、RIBES 値に有意な変化はみられなかった。このことから、事前並び替えにより語順の一致率を高めることが出来れば、翻訳精度に大きく貢献出来るが、ケンドールの τ の小規模な改善が翻訳精度に与える影響は限定的であることが分かる。

表 2.3 開発データにおける PBSMT のケンドールの τ と BLEU 値および RIBES 値 (最も性能の高いものと有意差がないもの ($p < 0.05$) を太字で表す)

		τ	BLEU	RIBES
英日	並び替えなし	37.56	25.23	67.44
	BTG	77.93	30.32	77.58
	RvNN	64.63	28.91	75.71
日英	並び替えなし	39.10	15.45	61.44
	BTG	46.98	19.38	69.22
	RvNN	49.93	17.72	69.67
英仏	並び替えなし	89.03	23.42	78.29
	BTG	83.70	25.27	79.25
	RvNN	88.93	23.93	78.16
仏英	並び替えなし	90.27	26.29	80.87
	BTG	91.70	27.56	81.28
	RvNN	91.35	26.58	80.92
英中	並び替えなし	76.21	7.90	58.83
	BTG	78.63	7.80	59.08
	RvNN	78.11	7.66	58.65
中英	並び替えなし	80.51	8.88	63.78
	BTG	82.16	7.97	64.02
	RvNN	80.77	7.69	61.25

一方で、BTG に基づく事前並び替えでは仏英対においてケンドールの τ を 1.43 ポイント向上しており、また BLEU 値、RIBES 値がそれぞれ 1.27 ポイント、0.41 ポイント、中英対においてもケンドールの τ を 1.65 ポイント改善し、また RIBES 値が 0.24 ポイント向上している。提案手法では中英対において BLEU、RIBES を向上出来なかった理由として、構文解析エラーの影響が考えられる。BTG に基づく事前並び替えでは事前並び替えに適した木構造を構築しながら並び替えを行う。一方、RvNN では構文解析器が出力する構文木に基づいて事前並び替えを行うため、構文解析器の精度が RvNN による並び替えの精度に影響する。本実験で用いた構文解析器の精度は、中国語で 77% [10] と報告されており、英語の場合の 91%

[62] より大幅に低い。中英翻訳のため構文解析した中国語文のうち、開発データから一文単位の BLEU 値が低下した 50 件を観察した結果、構文解析エラーと単語アライメントの質が低いことによる複合的な要因により事前並び替えに失敗していることが明らかとなった。構文解析に失敗した文は 13 文あり、そのうち名詞句の解析誤りが 9 件、動詞句の解析誤りが 6 件あった。これらの構文解析エラーにより、事前並び替えに失敗したものは 6 件あった。このうち構文解析エラーによりどのような並び替えを行ってもケンドールの τ を向上出来ないものが 3 件、単語アライメントそのものが出来ておらず、どのような並び替えを行ってもケンドールの τ を向上出来ないものが 2 件あった。これらは中英翻訳の実験に用いた対訳コーパス (IWSLT2015) は TED²¹⁾ より収集された口語体の文であるため構文解析が難しく、またコーパスサイズが小さいことから単語アライメントも困難なためと考えられる。

PBSMT および AttRNN による翻訳精度 PBSMT、AttRNN により翻訳を行った結果を表 2.4 に示す。BTG に基づく事前並び替えと RvNN を使用した事前並び替えを用いた PBSMT における歪み制約は 0 とした。並び替えなしのものは英日、日英では WAT2017 のベースラインシステムの設定に従い、歪み制約を 20 とした。英中、中英、英仏、仏英対では歪み制約を Moses のデフォルト値である 6 とした²²⁾。

PBSMT で翻訳を行った場合、英日方向の翻訳では並び替えなしに比べて RvNN を使用した事前並び替えと BTG に基づく事前並び替えの両方で BLEU 値がそれぞれ 4.62 ポイント、4.97 ポイント有意に向上した。また RIBES 値も事前並び替えなしに比べて、RvNN を使用した事前並び替えおよび BTG に基づく事前並び替えでそれぞれ 8.77 ポイント、9.58 ポイント有意に向上した。これらの結果から、英日翻訳において事前並び替えを行うことで翻訳精度が大きく向上していることが分かる。RvNN を使用した事前並び替えと BTG に基づく事前並び替えでは、BLEU 値および RIBES 値において統計的有意差は認められなかった (p 値はそれぞれ $p = 0.068$ 、 $p = 0.226$ であった)。このことから提案手法では、素性テンプレートの設計を必要とすることなく、本論文執筆時点で事前並び替え手法の最善の手法であ

21) <https://wit3.fbk.eu/2015-01>

22) Goto et al. [26] は中英翻訳において歪み制約を 10、20、30、 ∞ にして翻訳を行い、歪み制約が 10 の時に BLEU 値が一番高かったと報告している。そのため、英中、中英翻訳において、歪み制約を 6、10、20 に設定し翻訳実験を行った。その結果、開発データにおいて BLEU 値が一番高かった歪み制約 6 を選択した。

表 2.4 テストセットにおける BLEU 値および RIBES 値の評価結果 (最も性能の高いものと有意差がないもの ($p < 0.05$) を太字で表す)

		PBSMT		AttRNN	
		BLEU	RIBES	BLEU	RIBES
英日	並び替えなし	24.54	67.62	32.68	81.68
	BTG	29.51	77.20	28.91	79.58
	RvNN	29.16	76.39	29.01	79.63
英仏	並び替えなし	26.39	81.60	28.20	81.66
	BTG	27.85	82.05	27.86	81.30
	RvNN	26.64	81.56	28.85	81.84
英中	並び替えなし	10.16	68.71	12.44	71.86
	BTG	10.65	69.65	11.83	71.39
	RvNN	10.27	68.72	11.98	71.61
日英	並び替えなし	15.31	61.71	24.85	74.89
	BTG	19.14	69.98	20.11	71.07
	RvNN	17.30	69.36	19.92	71.74
仏英	並び替えなし	27.45	83.23	29.06	83.41
	BTG	28.32	83.64	28.37	82.97
	RvNN	27.18	83.16	28.61	83.18
中英	並び替えなし	11.94	70.88	15.49	72.52
	BTG	12.25	71.45	14.61	71.78
	RvNN	11.24	70.09	16.11	73.09

る BTG に基づく事前並び替えと同等の翻訳性能を達成していることが分かる。

英仏、英中方向における PBSMT を用いた翻訳結果の BLEU 値および RIBES 値は、BTG に基づく事前並び替えでは有意に向上したが、RvNN を使用した事前並び替えでは並び替えなしと同程度となった。一方で、RvNN を使用した事前並び替え、BTG に基づく事前並び替え、並び替えなしの三者について、英仏・英中翻訳における RIBES の値に有意差はなかった。これは英中、英仏の言語対では元々長距離の語順変換が不要であるため、事前並び替えの効果が限定的だったことを示唆していると考える。

また日英方向では、PBSMT を使用した翻訳結果では、並び替えなしと比較して RvNN を使用した事前並び替えを行うことで、BLEU 値が有意に 1.99 ポイント向上している。しかし BTG に基づく事前並び替えでの並び替えによる BLEU 値の向上には及ばない結果となった。RIBES 値は RvNN を使用した事前並び替えと BTG に基づく事前並び替えでそれぞれ 7.65 ポイント、8.27 ポイントの向上を達成している。仏英、中英方向において PBSMT による翻訳では、BTG に基づく事前並び替えでは BLEU 値が並び替えなしの場合に比べ有意に向上了しているが、RvNN を使用した事前並び替えでは、並び替えなしの場合よりも低下している。これは構文木の精度が影響したものと考えられる。

AttRNN では、RvNN による事前並び替えを行うと、英日、英中、日英、仏英方向の翻訳において BLEU 値、RIBES 値が低下した。しかし、英仏方向では BLEU 値が 0.65 ポイント、RIBES 値が 0.18 ポイント、中英方向では BLEU 値が 0.62 ポイント、RIBES 値が 0.57 ポイント向上した。BTG に基づく事前並び替えでは、AttRNN による翻訳の結果、並び替えなしの場合に比べすべての言語対で BLEU 値、RIBES 値が低下する結果となった。これは Sudoh et al. [87] の英中翻訳における実験結果と共通の現象であり、原因の一つとして、事前並び替えにより言語の構造が崩れてしまうことが考えられる。しかし英仏、中英対においては、RvNN を使用した事前並び替えの結果 AttRNN での翻訳精度が向上しており、自己回帰ニューラル機械翻訳おいても事前並び替えによる効果が発揮される場合があることを示している。

AttRNN による翻訳における RIBES 値について、英仏、仏英翻訳において BTG に基づく事前並び替えを行った PBSMT の性能が事前並び替えなしの AttRNN の性能をわずかに上回っているが、これらの結果には統計的有意差はなかった（それぞれ $p = 0.329$ 、 $p = 0.323$ ）。先行研究でも示されている通り、本実験においても事前並び替えの有無に関わらず、AttRNN が PBSMT を上回る結果となっている。

表 2.5 機械学習手法の違いによる並び替えの PBSMT による英日対での翻訳評価 (最も性能の高いものと有意差がないもの ($p < 0.05$) を太字で表す)

	BLEU	RIBES
並び替えなし	24.54	67.62
Hoshino et al.	28.58	75.27
RvNN	29.16	76.39

2.6 分析

2.6.1 機械学習手法の効果

提案手法では構文木に基づく機械学習により事前並び替えを行うが、構文木を用いる効果および機械学習手法の効果を分けて検証するため、提案手法と同様に機械学習を用いて構文木の各ノードで並び替えを行う手法である Hoshino et al. [34] と英日対で比較実験を行った。Hoshino et al. の手法は機械学習手法として Support Vector Machine を使用したものであり、著者らによる実装を使用した。

表 2.5 にその結果を示す。Hoshino et al. の手法と比較して、RvNN を使用した事前並び替えによる翻訳では BLEU 値が有意に 0.58 ポイント高い結果となった。この結果より、単語ベクトルおよび品詞・構文タグベクトルを考慮し RvNN による事前並び替えを行うことで、構文情報をよりとらえた並び替えを実現出来ることが分かる。

2.6.2 翻訳例の分析

表 2.6 に英日対において事前並び替えに成功した例、およびその PBSMT、AttRNN を使用した翻訳結果を示す。原文と参照訳では語順が大きく異なっているが、並び替えを行うことで語順が近づいていることが分かる。PBSMT による翻訳例では、並び替えなしの文と比べ、並び替えを行った文は意味が通るような訳文となっている。特に、動詞である “causes” が並び替えを行うことで文末に移動し、参照訳の「ひきおこす」と同様の意味を表す「原因となる」と翻訳されており、並び替えなしの翻訳である「部品である」と比べて正しい翻訳結果となってい

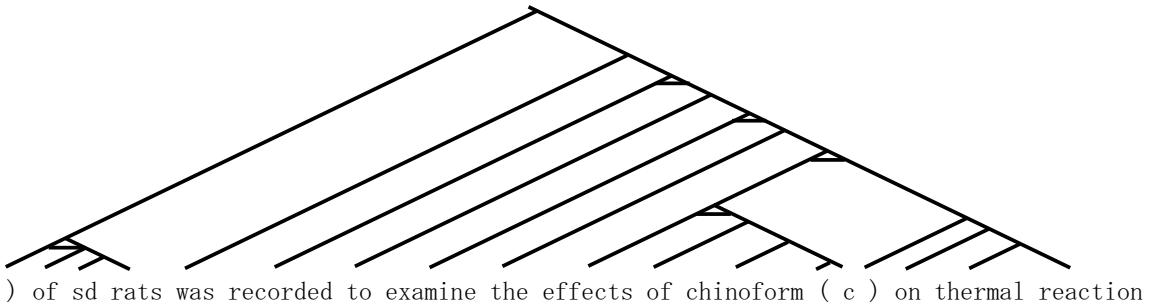


図 2.6 失敗した構文解析結果の一部分 (横線が引いてある節ノードは “Inverted” を示す)

る。AttRNN による翻訳例では PBSMT による翻訳例と比較してより流暢な翻訳となっているが、低頻度語である “economizer” が未知後を表す ⟨unk⟩ へと翻訳されている。

表 2.7 に英日対において事前並び替えに失敗した例およびその PBSMT、AttRNN による翻訳結果を示す。RvNN を使用した事前並び替えの例では、元々括弧外にあった単語列が、並び替えの結果括弧の中に入っていたり、左括弧と右括弧の順番が反対になっている。これは構文解析の結果、図 2.6 に示すように “(c)” というフレーズが誤って二つの句に分断されており、その誤りが事前並び替えに影響してしまったためである。BTG に基づく事前並び替えでは構文解析と並び替えを同時にを行うため、このような構文解析誤りの影響を受けない。

表 2.8、2.9 に、英仏対および英中対での並び替えおよび翻訳結果を示す。これらの言語対では語順が似ているため、事前並び替えを行っても語順はほとんど変化せず、実際に並び替えなしの文と BTG、RvNN によって並び替えられた文も、それほど変化しておらず、英仏対においては参照訳とほとんど同じ文が生成されている。

2.7 おわりに

本章では統計的機械翻訳のための素性テンプレートの設計を必要としない RvNN を使用した事前並び替え手法を提案した。英日、英仏、英中言語対を用いた評価実験の結果、提案手法は英日統計的機械翻訳において、人手で設計した素性テンプレートに基づく事前並び替え手法の本論文執筆時点で最善の手法 [66] と同等の翻訳性能を達成している。しかし、提案手法では外部の構文解析器を必要とするため、構文解析の誤りが事前並び替えの精度に影響する。よって、構文解析とノード

表 2.6 英日対における並び替えの成功例とその翻訳例

並び替え例	
原言語文	sulfur content causes sulfate corrosion of low temperature parts such as the fuel economizer and air preheater .
BTG	sulfur content the fuel economizer and air preheater as such of low temperature parts sulfate corrosion causes .
RvNN	sulfur content of as the fuel economizer and air preheater such low temperature parts sulfate corrosion causes .
参照訳	
硫黄分は節炭器や空気予熱器などの低温部で硫酸腐食をひきおこす。	
PBSMTによる翻訳例	
並び替えなし	硫酸塩の硫黄含有量が燃料及び空気予熱器エコノマイザの低温腐食などの部品である。
BTG	硫黄含有量が燃料エコノマイザ、空気予熱器等の低温部の硫酸腐食の原因となる。
RvNN	硫黄含有量の燃料は、エコノマイザ、空気予熱器等の低温部の硫酸腐食の原因となる。
AttRNNによる翻訳例	
並び替えなし	硫黄含有量は、燃料<unk>や空気予熱器などの低温部品の硫酸塩腐食を引き起こす。
BTG	硫黄含有量は、燃料予熱器や空気予熱器などの低温部品の硫酸腐食を引き起こす。
RvNN	硫黄含有量は燃料<unk>や空気予熱器などの低温部品の硫酸腐食を引き起こす。

表 2.7 英日対における並び替えの失敗例とその翻訳例

並び替え例

原言語文	single neural discharge of the cutaneous pain receptor (r) of sd rats was recorded to examine the effects of chinoform (c) on thermal reaction .
BTG	of sd rats of the cutaneous pain receptor (r) single neural discharge was recorded the of thermal reaction on chinoform (c) effects examine to .
RvNN	of the cutaneous pain receptor (r single neural discharge of sd rats) was to the) on thermal reaction of chinoform (c effects examine recorded .

参照訳

S D ラットの皮膚痛覚受容器（R）の単一神経放電を記録し、熱反応に対するキノホルム（C）の作用を検討した。

PBSMT による翻訳例

並び替えなし	单一神経放電に及ぼす影響を調べるため、キノホルムの S D ラットの皮膚痛覚受容体（R）の（C）について熱反応を記録した。
BTG	の S D ラットの皮膚痛覚受容体（R）の単一神経放電を記録の熱反応のキノホルム（C）の効果を検討した。
RvNN	の皮膚痛覚受容体（R の単一神経放電の S D ラット）は、）について熱反応の chinoform（C の効果を検討した。

AttRNN による翻訳例

並び替えなし	S D ラットの皮膚痛覚受容体（R）の単一神経放電を記録し、熱反応に対するキノホルム（C）の影響を検討した。
BTG	S D ラットの皮膚痛覚受容体（R）の単一神経放電を記録し、キノホルム（C）の熱反応に対する影響を調べた。
RvNN	ラットの皮膚痛覚受容体（R）の单神経放電 S D ラットの熱反応に対する影響を検討した。

表 2.8 英仏対における並び替えの例とその翻訳例

並び替え例	
原言語文	the u.s. senate approved a \$ 90-million pilot project last year that would have involved about 10,000 cars .
BTG	the senate u.s. approved a project pilot \$ 90-million year last that would have involved about 10,000 cars .
RvNN	the u.s. senate approved a project pilot \$ 90-million last year that would have involved about 10,000 cars .
参照訳	
le sénat américain a approuvé un projet pilote de 90 m \$ l' année dernière qui aurait porté sur environ 10000 voitures .	
PBSMTによる翻訳例	
並び替えなし	le sénat américain a approuvé un projet pilote 90-million \$ l' année dernière qui aurait participé environ 10000 voitures .
BTG	le sénat américain a approuvé un projet pilote \$ 90-million l' année dernière qui aurait participé environ 10000 voitures .
RvNN	le sénat américain a approuvé un projet pilote 90-million \$ l' année dernière qui aurait participé environ 10000 voitures .
AttRNNによる翻訳例	
並び替えなし	le sénat américain a approuvé un projet pilote de <unk> \$ l' an dernier qui aurait impliqué environ 10000 voitures .
BTG	le sénat américain a approuvé un projet pilote de <unk> \$ l' an dernier qui aurait impliqué environ 10000 voitures .
RvNN	le sénat américain a approuvé un projet pilote de <unk> \$ l' an dernier qui aurait impliqué environ 10000 voitures .

表 2.9 英中対における並び替えの例とその翻訳例

並び替え例

原言語文	i worked for an italian ngo , and every single project that we set up in africa failed .
BTG	i worked for an italian ngo , and that we set up in africa every single project failed .
RvNN	i worked for an italian ngo , and every that we set up in africa single project failed .

参照訳

我在一家意大利非政府组织工作。我们在非洲的每一个项目，都失败了。

PBSMT による翻訳例

並び替えなし	我的工作是为意大利的非政府组织，每一个项目，我们设立在非洲的失败了。
BTG	我做了一个意大利的非政府组织，和我们设立在非洲的每个项目都失败了。
RvNN	我的工作是为意大利公司的非政府组织，与每一个我们设立在非洲的一个项目都失败了。

AttRNN による翻訳例

並び替えなし	我为一个意大利非政府组织工作，每个我们在非洲建立的项目都失败了。
BTG	我为意大利非政府组织工作，我们建立了非洲的每一个项目。
RvNN	我为一个意大利非政府组织工作，我们在非洲建立了一个项目。

の並び替えを同時に行うことにより構文解析のエラーを考慮した並び替えモデルの構築が今後の課題の一つとして挙げられる。

先行研究 [21, 87]において、事前並び替えを行った文対で注意機構付き RNN モデルを訓練すると、翻訳精度が低下することが報告されている。しかし提案手法を用いた場合、英仏、中英対では注意機構付き RNN モデルにおいて BLEU 値が有意に向上しており、事前並び替えが自己回帰ニューラル機械翻訳モデルに貢献する可能性が示された。

第3章 Transformer のための事前並び替えのベクトル表現

3.1 はじめに

統計的機械翻訳において、原言語文と目的言語文の語順の相違は翻訳精度に大きく影響する [30, 66, 92]。この問題を解決する手法として、第2章では RvNN による事前並び替えを提案し、語順が大きく異なる英日言語対におけるフレーズベース統計的機械翻訳の翻訳精度を大幅に向上させることができることを示した。近年の機械翻訳における標準的な手法であるニューラル機械翻訳では Zhao et al. [109] が並び替えた後の文における位置に対応したベクトル表現を利用して翻訳精度を向上させた。しかし、RNN ベースの機械翻訳モデルより高い性能を持つ Transformer [95]において事前並び替えが有効に働くかどうかは依然として不明である。

Vaswani et al. によって提案された Transformer は、RNN ベースの機械翻訳モデルの翻訳精度を大きく上回った。しかし、Transformer は各単語のベクトル表現を独立に計算するため単語の語順を考慮することが出来ないという問題がある。そのため、Vaswani et al. は絶対的位置表現と呼ばれる位置のベクトル表現を単語のベクトル表現に加えてからモデルへの入力としている。また Shaw et al. は絶対的位置表現の代わりに相対的位置表現という、Transformer の各層で単語の相対位置を計算して各単語のベクトル表現に足し合わせる手法を提案した [82]。これらの位置ベクトル表現によって、原言語文と目的言語文のそれぞれの語順を Transformer で考慮出来るようになった。しかし、原言語、目的言語両方の文の語順を同時に考慮することは出来ない。

そこで本章では、Transformerにおいて原言語文と目的言語文の両方の語順を考慮するための事前並び替え位置表現を提案する。事前並び替え後の位置を使用したベクトル表現を単語のベクトル表現に加えることで、attention と呼ばれる重みを計算する際に語順の相違を考慮することが可能になり、事前並び替えの情報を考慮した単語のベクトル表現を計算することが可能になる。これにより、Transformer

は原言語文と目的言語文の語順を考慮した翻訳文を出力することが出来ると考えられる。

本章は以下のように構成されている。3.2 節では本章に関連する先行研究について議論する。3.3.1, 3.3.2 節では提案手法を適用するにあたって必要な背景知識について説明し、3.3.3 節では提案手法である事前並び替え位置表現について説明する。3.4 節では提案手法の有効性を確かめるため英日、英独、英チェコ、英露対を使用した翻訳評価実験とその結果について述べ、3.5 節で実験の結果から提案手法が翻訳制度に与える影響について詳細な分析を行う。3.6 節で本章のまとめと今後の課題を検討する。

3.2 関連研究

注意機構付き RNN モデルにおける語順の活用 Zhang et al. [107] は 注意機構付き RNN モデルにおいて、統計的機械翻訳における歪みモデルを attention の計算に活用する手法を提案している。この歪みモデルは原言語文の単語に対して、目的言語文中の位置を紐づける役割をしており、歪みモデルに基づいて attention をシフトし次の単語予測のための重みの計算を行っており、翻訳精度を大きく向上させた。この結果は目的言語文中の単語の位置が翻訳精度に貢献することを表している。Murthy et al. [63] は低資源の注意機構付き RNN における転移学習において、並び替えを活用する手法を提案している。この手法では、まず量が豊富な言語対の原言語文を、低資源な言語対の原言語文の語順に近づくように並び替えを行い、並び替えた文を使用して学習を行う。その後、低資源な言語対で転移学習を行う。この手法によって、並び替えをしなかったものと比べて翻訳精度が大幅に向上した。この結果は多言語翻訳においても語順の情報が大幅に翻訳精度を向上させることを示唆している。

位置のベクトル表現の活用 Du and Way [21] は、Sennrich and Haddow の提案した特徴量を使用した注意機構付き RNN モデル [78] において事前並び替えの情報を特徴量として使用することで翻訳精度を向上させている。Zhao et al. [109] は、注意機構付き RNN モデルにおいて事前並び替えのインデックスをベクトル表現として加えることで翻訳精度が向上したことを報告している。Chen et al. [11, 12] は Transformer モデルにおいて、語順に基づいたベクトル表現の学習手法を提案して

いる。エンコーダ側で原言語文の語順のベクトル表現を、デコーダ側で目的言語文の語順のベクトル表現を学習しており、原言語文、目的言語文両方の語順を使用して計算を行っている。しかし、事前並び替えは使用しておらず、Transformerにおいて事前並び替えを活用した研究は存在していない。

3.3 提案手法

3.3.1 前提知識: Transformer

Transformer は Vaswani et al. [95] によって提案されたモデルであり、自己注意機構を使用して単語のベクトル表現を計算し、そのベクトル表現に基づいて翻訳文の出力を行う。自己注意機構は、入力として受け取った文の単語ベクトル表現それぞれに対して attention を計算する仕組みである。図 3.1 にモデルの全体図を載せる。Transformer は原言語文を入力として受け取るエンコーダと、それまでに予測した単語列から次の単語の予測を行うデコーダの二つの部分からなり、式 (3.1) で表される損失関数 $\mathcal{L}(\mathbf{x}, \mathbf{y})$ の最小化を目的として訓練される。

$$\begin{aligned}\mathcal{L}(\mathbf{x}, \mathbf{y}) &= -\log Pr(\mathbf{y}|\mathbf{x}) \\ &= -\sum_{i=1}^m \log Pr(\mathbf{y}_i | \mathbf{y}_{<i}, \mathbf{x})\end{aligned}\tag{3.1}$$

ここで $\mathbf{x} = (x_1, x_2, \dots, x_n)$ は長さ n の原言語文、 $\mathbf{y} = (y_1, y_2, \dots, y_m)$ は長さ m の目的言語文を表す。

(1) エンコーダ エンコーダでは原言語文を入力として受け取り、入力文の単語数と同じ数のベクトル表現の列へと変換する。各単語を単語に対応付けられたベクトル表現に変換するモデルと、自己注意機構を使用してベクトル表現の計算を行う Multi-head attention および FFNN を一つの層とし、複数の層をスタックしたモデルからなる。Multi-head attention と FFNN で計算された後のベクトル表現は、Residual connection によりそれぞれの入力のベクトル表現と足し合わされた後に Layer Normalization [1] によって正規化される。

(2) デコーダ デコーダは $i - 1$ 番目までに出力した単語とエンコーダで計算されたベクトル表現から i 番目の単語を予測する。各単語を単語に対応付けられた

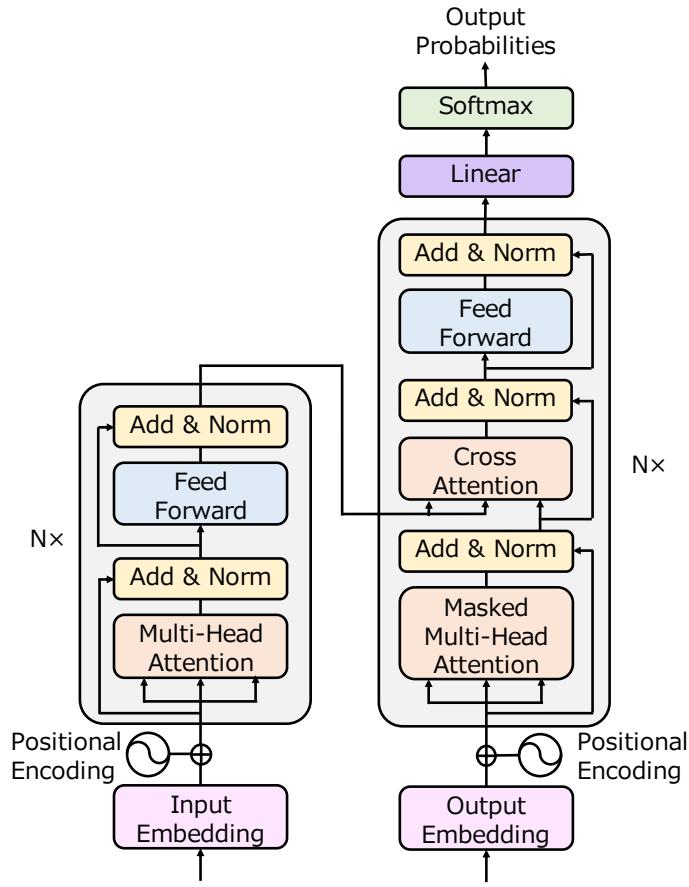


図 3.1 Transformer モデルの全体図 ([95] より引用、一部表記を改変)

ベクトル表現に変換するモデルと、Masked multi-head attention、Cross-attention、FFNN を 1 つの層として複数の層をスタックしたモデルからなる。自己注意機構を使用してベクトル表現の計算を行う Masked multi-head attention、Cross-attention および FFNN で計算された後のベクトル表現は、Residual connection によりそれぞれの入力のベクトル表現と足し合わされた後に Layer Normalization によって正規化される。その後 FFNN を通して計算されたベクトル表現から softmax 関数によって各単語の確率を計算し i 番目の単語を決定する。

(3) 自己注意機構 エンコーダ、デコーダとともに各層において自己注意機構を複数個使用する Multi-head attention によって各単語のベクトル表現を計算する。図 3.2 に Multi-head attention のモデル図を載せる。Multi-head attention は、以下の式に従って head と呼ばれる部分で自己注意機構を使用してベクトル表現を計算した後に各 head のベクトル表現を結合する。

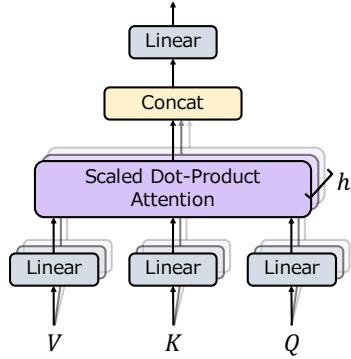


図 3.2 Multi-head attention のモデル図 ([95] より引用)

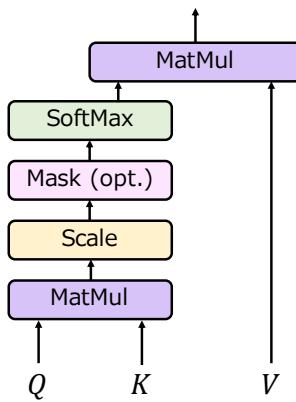


図 3.3 Scaled dot-product attention のモデル図 ([95] より引用)

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h)W^O$$

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

ここで $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d_k \times d_m}$ はそれぞれ重み行列、 d_m はベクトル表現の次元数である。それぞれの head ではこの重み行列によって各単語のベクトル表現の線形変換を行ってから attention の計算を行い新たなベクトル表現の計算を行う。このように複数の head によってベクトル表現の計算を行うことで、同じ単語に対して様々なベクトル表現を計算することが出来るため翻訳精度が向上する [95]。

本論文における自己注意機構では Scaled dot-product attention と呼ばれるものを使用した。図 3.3 に Scaled dot-product attention のモデル図を載せる。Scaled dot-product attention は入力である行列 $Q = (q_1, q_2, \dots, q_J)$, $K = (k_1, k_2, \dots, k_J)$, $V = (v_1, v_2, \dots, v_J)$, $q_i, k_i, v_i \in \mathbb{R}^{d_k}$ から以下の式に従って新たなベクトル表現を計算

する。

$$\mathbf{z}_i = \sum_{j=1}^n \alpha_{ij} (\mathbf{v}_j W^V + \mathbf{b}^V) \quad (3.2)$$

$$\begin{aligned} \alpha_{ij} &= \frac{\exp(\mathbf{s}_{ij})}{\sum_{k=1}^n \exp(\mathbf{s}_{ik})}, \\ \mathbf{s}_{ij} &= \frac{(\mathbf{q}_i W^Q + \mathbf{b}^Q)(\mathbf{k}_j W^K + \mathbf{b}^K)^T}{\sqrt{d_z}}, \end{aligned} \quad (3.3)$$

ここで J は原言語文または目的言語文の単語数であり、 d_k はベクトル表現の次元数である。 $W^V, W^V, W^Q \in \mathbb{R}^{d_e \times d_z}$ はベクトル表現を変換するための重み行列であり、 $\mathbf{b}^V, \mathbf{b}^V, \mathbf{b}^Q \in \mathbb{R}^{d_z}$ はバイアスである。 α_{ij} は i 番目のベクトル表現を計算する際の、 j 番目のベクトル表現に対する attention の重みである。

Cross-attention では V, K としてエンコーダで計算されたベクトル表現、 Q としてその前の層でのベクトル表現を受け取って、Multi-head attention と同様に複数の head でベクトル表現を計算する。その後、各 head で計算されたベクトル表現を結合する。

Feed-Forward では式 (3.4) で表される関数を通してベクトル表現の計算を行う。

$$\text{FFNN}(a) = \text{ReLU}(0, aW_1 + b_1)W_2 + b_2 \quad (3.4)$$

ここで $W_1 \in \mathbb{R}^{d_m \times d_{ff}}, W_2 \in \mathbb{R}^{d_{ff} \times d_m}$ は重み行列、 $b_1 \in \mathbb{R}^{d_{ff}}, b_2 \in \mathbb{R}^{d_m}$ はバイアス項である。

3.3.2 前提知識：絶対的位置表現と相対的位置表現

Vaswani et al. [95] はサイン関数とコサイン関数によって計算される絶対的位置表現で各単語の絶対的な位置の情報を使用している。絶対的位置表現は以下の式で計算される。

$$PE(i, 2q) = \sin(i/10000^{2q/d_z}), \quad (3.5)$$

$$PE(i, 2q + 1) = \cos(i/10000^{2q/d_z}), \quad (3.6)$$

i は単語のインデックスであり、0 から始まる位置で表される。 q は絶対的位置表現のベクトルの次元の何番目かを表すインデックスであり、0 から始まる位置で表

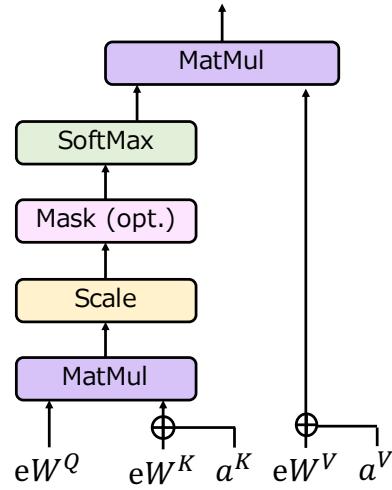


図 3.4 相対的位置表現を使用した Scaled dot-production attention

される。このベクトル表現を単語のベクトル表現に足し合わせることで単語の位置を表している。

Shaw et al. [82] は、絶対的な位置のベクトル表現の計算手法の代わりとして、単語の相対的な位置に基づいた相対的位置表現を使用した自己注意機構を提案している。相対的位置表現 a_{ij}^K と a_{ij}^V は以下の式で計算される。

$$a_{ij}^K = \text{rel}_{\text{clip}(j-i,k)} E_a^K, \quad (3.7)$$

$$a_{ij}^V = \text{rel}_{\text{clip}(j-i,k)} E_a^V, \quad (3.8)$$

$$\text{clip}(x, k) = \max(-k, \min(k, x)),$$

$\text{clip}(\cdot, k)$ で相対的な位置を計算する。 k は考慮する最大の距離であり、前後 k 単語(自身を含めた周辺の $2k + 1$ 単語)の位置をラベルとすることでそれぞれの単語の相対的位置表現の計算を行う。 $\text{rel}_i \in \mathbb{R}^{2k+1}$ は、対応した位置が 1 でそれ以外は 0 である one-hot ベクトルである。 E_a^K と $E_a^V \in \mathbb{R}^{(2k+1) \times d_z}$ は、各行が一つの位置のベクトル表現に対応した行列であり、訓練を通して学習される。

図 3.4 に相対的位置表現を用いた Scaled dot-product-attention のモデル図を示す。 a_{ij}^K と a_{ij}^V は式 (3.2) と (3.3) において、式 (3.9) と (3.10) に従って足し合わされる。

$$\mathbf{z}_i = \sum_{j=1}^n \alpha_{ij} (\mathbf{e}_j W^V + \mathbf{b}^V + \mathbf{a}_{ij}^V), \quad (3.9)$$

$$\mathbf{s}_{ij} = \frac{(\mathbf{e}_i W^Q + \mathbf{b}^Q)(\mathbf{e}_j W^K + \mathbf{b}^K + \mathbf{a}_{ij}^K)^T}{\sqrt{d_z}}. \quad (3.10)$$

相対的位置表現を attention の計算の際にそれぞれ単語のベクトル表現に足し合わせることで、 \mathbf{z}_i は各単語の位置を考慮したベクトル表現となる。

3.3.3 事前並び替え位置表現

3.3.2 節で説明した絶対的位置表現や相対的位置表現は、エンコーダで原言語文の単語の位置を考慮することは出来るが、原言語文と目的言語文の語順の相違を考慮することが出来ない。そこで、本節では事前並び替えによって得られた原言語文の単語の並び替え後のインデックスを使用した事前並び替え位置表現を提案する。事前並び替えを使用した絶対的位置表現では、並び替えた後のインデックスを使用して各単語の事前並び替え位置表現を計算する。その後、元々のインデックスでの絶対的位置表現と並び替えた後のインデックスによる事前並び替え位置表現の両方を単語のベクトル表現に足し合わせることで事前並び替えの語順を考慮したベクトル表現が計出来る。

事前並び替えを用いた相対的位置表現は式 (3.7) と (3.8) において事前並び替え位置表現を足し合わせて計算される。図 3.5 に英語の文 “I like the pen that my father bought yesterday.” に対して事前並び替え位置表現を計算する例を示す。この英語文は、事前並び替えを適用することで図の下側の “I my father yesterday bought that the pen like” へと並び替えられる。並び替える前の文の、並び替えた後の文のインデックスは $(0, 8, 6, 7, 5, 1, 2, 4, 3)$ となる。ここで例として “bought” に対するクリップされた相対位置を考えると、それぞれ $(-4, 4, 2, 3, 1, -3, -2, 0, -1)$ となる。例えば、“pen” の並び替え後の文のインデックスは 7 であり、相対位置は $7 - 4 = 3$ となる。また、“yesterday” の並び替え後の文のインデックスは 3 であり、その相対位置は $3 - 4 = 1$ となる。

相対位置表現に対する事前並び替え位置表現 \mathbf{r}_{ij}^K と \mathbf{r}_{ij}^V は以下の式に従って計算される。

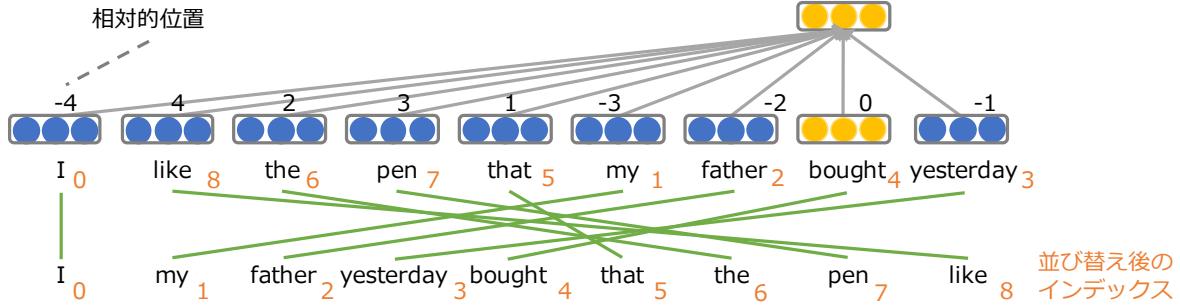


図 3.5 “bought” に対して相対位置の計算を行う例

$$\begin{aligned} \mathbf{r}_{ij}^K &= \text{rel}_{\text{RCD}(i,j,k)} E_r^K, \\ \mathbf{r}_{ij}^V &= \text{rel}_{\text{RCD}(i,j,k)} E_r^V, \\ \text{RCD}(i, j, k) &= \text{clip}(\text{rpos}(j) - \text{rpos}(i), k) \end{aligned}$$

ここで $\text{rpos}(k)$ は k 番目の単語の、並び替えた後のインデックスを返す関数である。 $E_r^K, E_r^V \in \mathbb{R}^{(2k+1) \times d_z}$ は各行が一つの位置のベクトル表現に対応した行列であり、 $\mathbf{r}_{ij}^K, \mathbf{r}_{ij}^V \in \mathbb{R}^{d_z}$ は並び替えた後のインデックスを使用して計算された相対位置のベクトル表現である。これによって並び替えを考慮したベクトル表現を計算することが出来る。

$$\begin{aligned} \mathbf{z}_i &= \sum_{j=1}^n \alpha_{ij} (\mathbf{e}_j W^V + \mathbf{b}^V + \mathbf{a}_{ij}^V + \mathbf{r}_{ij}^V), \\ \mathbf{s}_{ij} &= \frac{(\mathbf{e}_i W^Q + \mathbf{b}^Q)(\mathbf{e}_j W^K + \mathbf{b}^K + \mathbf{a}_{ij}^K + \mathbf{r}_{ij}^K)^T}{\sqrt{d_z}}. \end{aligned}$$

\mathbf{r}_{ij}^K と \mathbf{r}_{ij}^V を計算した後、式 (3.9) と (3.10) において事前並び替え位置表現を足し合わせることでベクトル表現 \mathbf{z}_i を計算する。これにより、 \mathbf{z}_i は原言語文と目的言語文の両方の語順を考慮したベクトル表現となる。

3.4 翻訳評価実験

本節では翻訳実験とその結果を述べる。3.4.1 節では実験設定について述べ、3.4.2 節では実験結果を述べる。詳細な分析については 3.5 節で行う。

表 3.1 翻訳の学習に用いたデータの統計量 (文対)

	訓練データ	開発データ	テストデータ
英日	1,814,494	1,790	1,812
英独	2,227,354	3,003	2,169
英チェコ	745,782	3,003	2,656
英露	817,256	3,003	2,818

3.4.1 実験設定

コーパスと前処理 提案手法の有効性を確認するため、言語構造の大きく異なる英日対で翻訳実験を行った。さらに、ドメインの違いによる翻訳への影響を避けるため、ニュースコーパスが提供されている英独対、英チェコ対、英露対で翻訳実験を行った。英日対では ASPEC コーパス [67]、英独対、英露対では Common Crawl Corpus¹⁾、英チェコ対では Common Crawl Corpus と CzEng 1.0²⁾[3] を使用した。前処理として、原言語文もしくは目的言語文が 50 単語以下で、文対の単語数の比が Moses の前処理スクリプトのデフォルト値である 9 以下の条件を満たす文対を使用した。表 3.1 にそれぞれの言語対の実験で使用した文対数を載せる。

英語文の単語分割および品詞タグ付けは Stanford Core NLP を使用し、構文解析には Enju を使用した。日本語文の単語分割には Juman³⁾ を使用し、構文解析には CkyLark を使用した。ドイツ語文の単語分割には Moses に付属の tokenizer⁴⁾ を使用し、構文解析には Berkeley parser⁵⁾ を使用した。チェコ語とロシア語の文の単語分割および品詞タグ付けは Stanza⁶⁾[76] を使用した。⁷⁾

事前並び替えモデルの訓練 本実験では事前並び替えモデルとして BTG に基づく事前並び替えと RvNN を使用した事前並び替えを使用した。単語のアライメ

1) <http://www.statmt.org/wmt15/translation-task.html>

2) <http://ufal.mff.cuni.cz/czeng/czeng10>

3) <http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN>

4) <https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl>

5) <https://github.com/slavpetrov/berkeleyparser>

6) <https://stanfordnlp.github.io/stanza/>

7) チェコ語とロシア語の構文解析器が存在しないため、RvNN によるチェコ英翻訳、露英翻訳の実験は行わなかった。

表 3.2 英日翻訳における、クリッピングの距離 k を変化させた時の開発データの BLEU 値

	$k = 1$	$k = 2$	$k = 4$	$k = 8$
ベースライン	33.36	33.62	34.15	33.53
+BTG (相対的位置表現)	33.91	34.36	34.64	34.72

ントの計算には MGIZA を用いた。BTG に基づく事前並び替えは、Nakagawa によって実装されたコードを使用した。訓練は訓練データから無作為にサンプリングした 10 万文を用いて 20 イテレーション学習させた。単語クラスタのサイズは 256 とした。

RvNN を使用した事前並び替えは BTG に基づく事前並び替えと同じ 10 万文の文で 5 エポック学習を行った。語彙は頻度が高いものから 5 万語とし、ミニバッチのサイズは 500 とした。単語ベクトル表現と品詞タグベクトル表現の次元数は 200 とした。

Transformer の訓練 Transformer の訓練では全て以下に示すハイパーパラメータを使用した。語彙は頻度の多いものから 5 万単語を使用した。エンコーダ、デコーダはともに 6 層であり、 d_e と d_z は 512、head の数は 8、相対的位置表現と事前並び替え位置表現のクリッピングの距離はともに 4 とした。最適化手法として Adam [47] を用い、学習率の初期値は 0.001 とした。学習率は 5 万回のイテレーション後、1 万イテレーションごとに減衰させていった。学習は 25 万イテレーションを行い、開発データでパープレキシティが一番低いイテレーションでのモデルを使用してテストデータの翻訳を行った。

提案手法は OpenNMT-py⁸⁾ の Transformer 上に実装した。絶対的位置表現と相対的位置表現を両方とも使用したものをベースラインとした。Shaw et al. [82] は絶対的位置表現と相対的位置表現を両方使用しても翻訳精度は向上しなかったと報告している。しかし、予備実験において英日翻訳、英独翻訳で実験した際には両方使用することで翻訳精度が向上した。そのため、本実験では両方使用したものをベースラインとした。

クリッピングの距離 k による BLEU 値の変化を検証するため、英日翻訳において開発データを使用して実験を行った。表 3.2 に翻訳実験の結果を示す。 k が 4 以下の時は、ベースライン、BTG による相対的位置表現に対する事前並び替え

8) <https://github.com/OpenNMT/OpenNMT-py>

表 3.3 英語から各言語への翻訳実験における、ケンドールの τ 、BLEU 値、および RIBES 値のテストデータの数値 (表中の太字は $p < 0.05$ で一番スコアの高いモデルと統計的な有意差がなかったものであり、 \uparrow と \downarrow はそれぞれベースラインのスコアと比較して $p < 0.05$ で統計的に有意に向上したもの、および有意に低下したもの。“+BTG” と “+RvNN” はそれぞれ BTG と RvNN を使用した事前並び替え位置表現を指す)

		英日			英独		
		τ	BLEU	RIBES	τ	BLEU	RIBES
ベースライン		27.46	35.53	83.68	75.79	13.41	74.45
相対的位置表現	+BTG	59.35 \uparrow	34.50 \downarrow	83.35	72.92 \downarrow	15.25\uparrow	75.44
	+RvNN	44.15 \uparrow	35.05 \downarrow	83.42	75.77	15.23\uparrow	75.30
絶対的位置表現	+BTG	59.35 \uparrow	34.41 \downarrow	83.30	72.92 \downarrow	15.60\uparrow	75.73
	+RvNN	44.15 \uparrow	34.56 \downarrow	83.23	75.77	15.33\uparrow	75.09

		英チェコ			英露		
		τ	BLEU	RIBES	τ	BLEU	RIBES
ベースライン		75.42	10.79	70.29	79.28	11.80	70.87
相対的位置表現	+BTG	74.40 \downarrow	10.87	70.60	77.61 \downarrow	11.80	71.49
	+RvNN	75.19	9.23 \downarrow	68.29 \downarrow	79.28	13.06\uparrow	72.51\uparrow
絶対的位置表現	+BTG	74.40 \downarrow	10.80	70.18	77.61 \downarrow	11.79	70.99
	+RvNN	75.19	9.24 \downarrow	67.80 \downarrow	79.28	13.28\uparrow	72.67\uparrow

位置表現とともに BLEU 値が増加している。しかし、 k が 8 の時は、ベースラインの BLEU 値は低下する。一方で、BTG による相対的位置表現に対する事前並び替え位置表現による BLEU 値は k が 8 の時も増加している。しかし、 k が 4 と 8 の BLEU 値の差は無視出来るほど小さい。そのため、本実験では、全てのモデルにおいて $k = 4$ とした。

3.4.2 実験結果

翻訳文は BLEU 値と RIBES 値によって評価した。実験結果がパラメータの初期値のランダム性に依存してしまうのを避けるため、各実験は 3 回ずつを行い、その平均の値を最終的な評価値とした。

表 3.4 各言語から英語への翻訳実験における、ケンドールの τ 、BLEU 値、および RIBES 値のテストデータの数値 (表中の太字は $p < 0.05$ で一番スコアの高いモデルと統計的な有意差がなかったものであり、 \uparrow と \downarrow はそれぞれベースラインのスコアと比較して $p < 0.05$ で統計的に有意に向上したもの、および有意に低下したものである。“+BTG” と “+RvNN” はそれぞれ BTG と RvNN を使用した事前並び替え位置表現を適用したものを指す)

		日英			独英		
		τ	BLEU	RIBES	τ	BLEU	RIBES
ベースライン		33.74	23.94	76.06	75.07	17.08	77.83
相対的位置表現	+BTG	49.82 \uparrow	25.28\uparrow	76.52	73.47	16.49 \downarrow	77.50
	+RvNN	35.04 \uparrow	25.06\uparrow	76.52	74.81	16.49 \downarrow	77.43
絶対的位置表現	+BTG	49.82 \uparrow	24.95\uparrow	76.59	73.47	16.76 \downarrow	77.86
	+RvNN	35.04 \uparrow	24.67 \uparrow	76.28	74.81	16.70 \downarrow	77.61

		チェコ英			露英		
		τ	BLEU	RIBES	τ	BLEU	RIBES
ベースライン		76.84	16.88	75.28	79.12	15.16	74.64
相対的位置表現	+BTG	78.16 \uparrow	17.03\uparrow	75.36	78.24 \downarrow	14.86 \downarrow	74.29
	+RvNN	-	-	-	-	-	-
絶対的位置表現	+BTG	78.16 \uparrow	16.94	75.18	78.24 \downarrow	15.17	74.53
	+RvNN	-	-	-	-	-	-

英語から各言語への翻訳結果 表 3.3 に英語から各言語への翻訳実験における BLEU 値と RIBES 値を示す。ベースラインと比較すると、英独翻訳では BTG による事前並び替え位置表現 (表 3.3 中の “BTG” の行) と RvNN による事前並び替え位置表現 (表 3.3 中の “RvNN” の行) によって BLEU 値が、相対的位置表現においてそれぞれ 1.84 ポイント、1.82 ポイント、絶対的位置表現においてそれぞれ 2.19 ポイント、1.92 ポイント向上し、RIBES 値は相対的位置表現においてそれぞれ 0.99 ポイント、0.85 ポイント、絶対的位置表現においてそれぞれ 1.28 ポイント、0.64 ポイント向上した。英露翻訳では BTG に基づく事前並び替えでは翻訳精度は向上しなかったが、RvNN を使用した事前並び替えによって相対的位置表現、絶対的位置表現それぞれで BLEU 値が 1.26 ポイント、1.48 ポイント、RIBES 値が 1.64 ポイント、1.80 ポイント向上した。

しかし、英日翻訳においては、RvNN を使用した事前並び替えと BTG に基づく事前並び替えによる BLEU 値が相対的位置表現によってそれぞれ 1.03 ポイント、0.48 ポイント、絶対的位置表現によってそれぞれ 1.12 ポイント、0.97 ポイント低下した。また RIBES 値は相対的位置表現によってそれぞれ 0.33 ポイント、0.26 ポイント、絶対的位置表現によってそれぞれ 0.38 ポイント、0.45 ポイント低下した。英チェコ翻訳では、RvNN を使用した事前並び替えによる相対的位置表現と絶対的位置表現によって BLEU 値が 1.56 ポイント、1.55 ポイント、RIBES 値が 2.00 ポイント、2.29 ポイント低下した。

各言語から英語への翻訳結果 表 3.4 に各言語から英語方向への翻訳実験における BLEU 値と RIBES 値を示す。日英翻訳では、BTG による事前並び替え位置表現(表 3.4 中の “BTG” の行)と RvNN による事前並び替え位置表現(表 3.4 中の “RvNN” の行)によって BLEU 値が、相対的位置表現によって 1.34 ポイント、1.12 ポイント向上し、絶対的位置表現によって 1.01 ポイント、0.73 ポイント向上した。チェコ英翻訳では、BTG に基づく事前並び替えを使用した相対的位置表現によって 0.15 ポイント、絶対的位置表現によって 0.06 ポイント向上した。

しかし、独英翻訳においては BTG に基づく事前並び替えと RvNN による BLEU 値が相対的位置表現によって 0.59 ポイント、0.59 ポイント、絶対的位置表現によって 0.32 ポイント、0.38 ポイント低下した。RIBES 値も BTG に基づく事前並び替えと RvNN を使用した事前並び替えによる相対的位置表現によって 0.33 ポイント、0.40 ポイント低下し、RvNN を使用した事前並び替えによる絶対的位置表現によって 0.22 ポイント低下した。露英翻訳では、BTG に基づく事前並び替えを使用した相対的位置表現で BLEU 値が 0.30 ポイント、RIBES 値が 0.35 ポイント低下した。一方で、BTG に基づく事前並び替えを使用した絶対的位置表現では BLEU 値が 0.01 ポイント向上し、RIBES 値は 0.09 ポイント低下した。

事前並び替え位置表現を使用しない場合の翻訳結果 事前並び替えを行って得られた文のみを使用した場合の翻訳精度への影響を確かめるために、事前並び替え位置表現を使用せず並び替えた文を直接 Transformer へ入力して翻訳実験を行った。表 3.5 に事前並び替えによって得られた文を用いた際の英日対における実験結果を載せる(表中の “BTG” と “RvNN” はそれぞれ BTG に基づく事前並び替え使用したもの、RvNN を使用した事前並び替えを使用したものと指す)。この結果から、事前並び替えを行った文を直接使用すると翻訳精度が低下することが分かった。こ

表 3.5 英日対における並び替えを行った文のみを使用した際の BLEU 値と RIBES 値

	英日		日英	
	BLEU	RIBES	BLEU	RIBES
ベースライン	35.53	83.68	23.94	76.06
BTG	31.20	81.15	22.11	73.51
RvNN	32.18	81.83	20.99	73.88

の結果は Du and Way [21]、Chen et al. [11]、Wang et al. [99] が報告している「事前並び替えを適用した文で直接翻訳の訓練を行うと翻訳精度が低下する」という結果と一致しており、先行研究においてもベースラインと比較して BLEU 値が 1 から 3 ポイント、RIBES 値が 2 から 3 ポイント程度低下している。この結果から、並び替えを使用する際は並び替える前の文の位置が重要であることが分かる。

3.5 分析

事前並び替え位置表現がどういう点で翻訳精度を向上させたのかを調査するため、以下の四つの観点から詳細な分析を行った。

- Q1** 事前並び替え位置表現によって向上出来る翻訳精度の上限はいくらか。
(3.5.1 節)
- Q2** 事前並び替え位置表現の精度はどの程度翻訳精度に影響を与えるか。 (3.5.2 節)
- Q3** 事前並び替え位置表現が訳抜けと重複訳の問題に対してどう影響するか。⁹⁾
(3.5.3 節)
- Q4** 文長によって翻訳精度はどのように変化するか。 (3.5.4 節)

絶対的位置表現に対する事前並び替え位置表現と相対的位置表現に対する事前並び替え位置表現の翻訳結果の傾向は似ているため、この後では相対的位置表現に対する事前並び替え位置表現について分析を行った。

表 3.6 テストデータでの BLEU 値と RIBES 値（「+オラクル」はオラクルな事前並び替えによる事前並び替え位置表現を使用した結果を表す）

		BLEU	RIBES
英日	ベースライン	35.53	83.68
	+オラクル	44.84	89.85
英独	ベースライン	13.41	74.45
	+オラクル	20.67	79.88
英チェコ	ベースライン	10.79	70.29
	+オラクル	14.37	74.84
英露	ベースライン	11.80	70.87
	+オラクル	14.88	74.76
日英	ベースライン	23.94	76.06
	+オラクル	36.45	87.91
独英	ベースライン	17.08	77.83
	+オラクル	21.88	81.54
チェコ英	ベースライン	16.88	75.28
	+オラクル	20.82	79.00
露英	ベースライン	15.16	74.64
	+オラクル	18.58	77.19

3.5.1 事前並び替え位置表現による翻訳精度の上限

まず事前並び替え位置表現による翻訳精度の向上の上限値がどの程度かを調査するために、オラクルな並び替えを用いて翻訳実験を行った。訓練データ、開発データ、テストデータ全てを使用して MGIZA によって単語アライメントを計算した後、Neubig et al. [68] と同様に単語アライメントに基づいて原言語文の単語を目的言語文の語順となるように並び替えたものをオラクルな並び替えとした。

表 3.6 にオラクルな事前並び替えを用いた結果を示す。全ての言語対において、オラクルな事前並び替えを使用することでベースラインよりも翻訳精度が大幅に向

9) 訳抜けは翻訳モデルの出力した文の意味が参照訳と比較して欠如している問題、重複訳は翻訳モデルの出力した文が同じフレーズを何度も繰り返している問題を指す。

表 3.7 ベースラインとオラクルな事前並び替えを使用した翻訳文における 1、2、3、

		4-gram ごとの適合率			
		1-gram	2-gram	3-gram	4-gram
英日	ベースライン	71.0	47.5	33.3	23.8
	オラクル	75.4 (+4.4)	55.9 (+8.4)	42.7 (+9.4)	33.0 (+9.2)
英独	ベースライン	40.9	18.3	9.6	5.4
	オラクル	47.0 (+6.1)	25.9 (+7.6)	15.7 (+6.1)	9.8 (+4.4)
英チェコ	ベースライン	36.6	14.8	7.1	3.6
	オラクル	38.9 (+2.3)	18.7 (+3.9)	10.2 (+3.1)	5.6 (+2.0)
英露	ベースライン	35.8	15.7	8.1	4.3
	オラクル	38.6 (+2.8)	19.0 (+3.3)	10.7 (+2.6)	6.1 (+1.8)
日英	ベースライン	60.4	31.6	18.6	11.3
	オラクル	69.6 (+9.2)	46.2 (+14.6)	32.0 (+13.4)	22.5 (+11.2)
独英	ベースライン	46.4	22.3	12.0	6.6
	オラクル	50.5 (+4.1)	27.7 (+5.4)	16.4 (+4.4)	10.0 (+3.4)
チェコ英	ベースライン	48.6	22.1	11.8	6.4
	オラクル	52.2 (+3.6)	27.4 (+5.3)	16.2 (+4.4)	9.8 (+3.4)
露英	ベースライン	45.4	20.2	10.4	5.5
	オラクル	48.5 (+3.1)	24.1 (+3.9)	13.4 (+3.0)	7.7 (+2.2)

上している。この結果は目的言語文の語順の情報が翻訳精度に大きな影響を与えることを示している。英語を原言語とする翻訳での BLEU 値は、英日翻訳で 9.31 ポイント、英独翻訳で 7.26 ポイント、英チェコ翻訳で 3.58 ポイント、英露翻訳で 3.08 ポイント向上した。英語を目的言語とする翻訳での BLEU 値は、日英翻訳で 12.51 ポイント、独英翻訳で 4.80 ポイント、チェコ英翻訳で 3.94 ポイント、露英翻訳で 3.42 ポイント向上した。この結果から、今回使用したコーパスでは日英翻訳においてより事前並び替え位置表現の効果があることが分かる。

翻訳精度の向上についてさらに分析を行うために、ベースラインとオラクルな事前並び替えを使用した翻訳の n -gram 適合率の評価を分析した。表 3.7 にテストデータでの 1、2、3、4-gram ごとの精度を載せる。各 n -gram の適合率において、オラクルな並び替えを使用した結果がベースラインを上回っている。この結果は、

事前並び替えの情報が単語レベルの翻訳 (1-gram) だけでなくフレーズレベルの翻訳 (2 から 4-gram) にとっても有効であることを示している。

3.5.2 事前並び替えの精度と翻訳精度の関係

事前並び替えの精度と翻訳精度の関係を調査するため、オラクルな並び替えを正解データとして事前並び替えの性能を評価した。事前並び替えの評価にはケンドールの τ を使用した。

表 3.3 に英語から各言語への翻訳における結果を載せる。英日翻訳ではケンドールの τ が、BTG に基づく事前並び替えによって 31.89、RvNN を使用した事前並び替えによって 16.69 向上した。一方で、BLEU 値は BTG に基づく事前並び替えで 1.03 ポイント、RvNN を使用した事前並び替えで 0.48 ポイント低下した。英独翻訳では、ケンドールの τ は BTG に基づく事前並び替えによって 2.87 ポイント低下し、RvNN を使用した事前並び替えでは 0.02 ポイント低下した一方で、BLEU 値は、相対的位置表現に対する事前並び替え位置表現を用いることで BTG に基づく事前並び替えで 1.84 ポイント、RvNN を使用した事前並び替えで 1.82 ポイント、絶対的位置表現に対する事前並び替え位置表現を用いることで BTG に基づく事前並び替えで 2.19 ポイント、RvNN を使用した事前並び替えで 1.92 ポイント向上した。英チェコ翻訳では、RvNN を使用した事前並び替えではケンドールの τ は 0.23 ポイント低下し、BLEU 値は相対的位置表現に対する事前並び替え位置表現で 1.56 ポイント、絶対的位置表現に対する事前並び替え位置表現で 1.55 ポイント低下した。英露翻訳では、RvNN を使用した事前並び替えではケンドールの τ は変化しなかった一方で、BLEU 値は相対的位置表現に対する事前並び替え位置表現で 1.26 ポイント、絶対的位置表現に対する事前並び替え位置表現で 1.48 ポイント向上した。これに対して様々な分析 (ケンドールの τ ごとの BLEU 値や、並び替えた後と元々の単語の距離とそれぞれの BLEU 値、元々の文と並び替えた文のケンドールの τ の差と BLEU 値など) を行ったが、英語から各言語への翻訳では明確な傾向は見られなかった。

表 3.4 に各言語から英語への翻訳における結果を載せる。日英翻訳ではケンドールの τ が、BTG に基づく事前並び替えによって 16.08、RvNN を使用した事前並び替えによって 1.30 ポイント向上し、チェコ英翻訳では、BTG に基づく事前並び替えによってケンドールの τ が 1.32 ポイント向上した。一方で独英翻訳では、BTG

表 3.8 英語から各言語への翻訳における、文ごとの単語の削除の回数と挿入の平均回数 (\uparrow と \downarrow はそれぞれベースラインと $p < 0.05$ で有意に向上したものと低下したものを表す。
“+BTG”、“+RvNN”、“+オラクル”はそれぞれ提案手法を BTG による並び替え、RvNN による並び替え、オラクルな事前並び替えで訓練したものを指す)

	英日		英独		英チェコ		英露	
	訳抜け	重複訳	訳抜け	重複訳	訳抜け	重複訳	訳抜け	重複訳
ベースライン	1.15	4.52	2.12	1.94	1.28	1.52	1.45	1.79
+BTG	1.17	4.80	2.07	1.98	1.21	1.52	1.35	1.95 \downarrow
+RvNN	1.16	4.58	2.29	2.01	1.19	1.75 \downarrow	1.54	1.73
+オラクル	0.77 \uparrow	3.58 \uparrow	1.48 \uparrow	2.45 \downarrow	0.89 \uparrow	1.69 \downarrow	1.03 \uparrow	2.18 \downarrow

表 3.9 各言語から英語への翻訳における、文ごとの単語の削除と挿入の平均回数 (\uparrow と \downarrow はそれぞれベースラインと $p < 0.05$ で有意に向上したものと低下したものを表す。
“+BTG”、“+RvNN”、“+オラクル”はそれぞれ提案手法を BTG による並び替え、RvNN による並び替え、オラクルな事前並び替えで訓練したものを指す)

	日英		独英		チェコ英		露英	
	訳抜け	重複訳	訳抜け	重複訳	訳抜け	重複訳	訳抜け	重複訳
ベースライン	1.81	3.45	1.09	3.15	1.21	2.41	1.42	3.40
+BTG	1.68	3.55	1.26	3.30	1.26	2.34	1.33	3.56
+RvNN	1.66 \uparrow	3.66	1.22	3.22	-	-	-	-
+オラクル	0.68 \uparrow	3.07 \uparrow	0.85 \uparrow	3.52 \downarrow	0.91 \uparrow	2.69 \downarrow	1.38	3.66 \downarrow

に基づく事前並び替えによってケンドールの τ が 1.60 ポイント低下し、RvNN を使用した事前並び替えで 0.26 ポイント低下した。露英翻訳では BTG に基づく事前並び替えによってケンドールの τ が 0.88 ポイント低下した。特に、独英、チェコ英、露英翻訳はニュースのコーパスであり、ケンドールの τ と BLEU 値に相関が見られる。このことから、各言語から英語方向への翻訳においては事前並び替え位置表現は翻訳精度の向上に有効であることが示唆される。

3.5.3 訳抜けと重複訳に対する効果

翻訳の訓練に事前並び替え位置表現を使用することで目的言語文の語順が分かり、そのため訳抜けが減少すると推測した。この仮説を確かめるために、Takebayashi et al. [91] と同様に重複訳と訳抜けの自動評価を行った。ここでは、Translation Edit Rate (TER) [84] における「挿入」と「削除」操作をそれぞれ訳抜けと重複訳として評価を行った。

表 3.8 と 3.9 に文ごとの操作回数の平均を示す。全ての結果について、ブートス

表 3.10 日英翻訳における並び替えとその翻訳例

並び替え例	
原言語文	対策としては導電性のテーブルマットとフロアマット、リストストラップを挙げ、人体の静電容量も表で示した。
BTG	挙げを対策 テーブルマットととしては導電性のフロアマット、リストストラップ、容量電人体の静も示しで表た。
RvNN	として対策はマットとフロアテーブル性の電マット、リストストラップを導挙げ、容量電人体の静もで表した。

参照訳

as a countermeasure, **electroconductive table mat and floor mat and list strap are listed**, and the static charged capacity of human body is shown in table.

翻訳例

ベースライン	antistatic capacity of human body is also shown in a table.
+BTG	antistatic table mat and floor mat of electroconductive are mentioned as countermeasure, and electrostatic capacity of human body is also shown in the table.
+RvNN	as countermeasure, this paper mentions table mat and floor mat of the conductivity, and <unk>, and shows the electrostatic capacity of the human body in the table.

トラップ法 [48] を用いて有意差検定を行った。ベースラインと比較すると、日英翻訳において RvNN による事前並び替えを用いることで訳抜けが有意に減少したことが分かる。絶対的位置表現と相対的位置表現は、原言語文の単語の位置は分かれるがそれに対応した目的言語文の単語の位置は分からぬ。そのため、並び替えを考慮した翻訳が出来ない。一方で、事前並び替え位置表現は事前並び替えモデルによって推測された目的言語文の語順を使用することが出来る。そのため、特に日英翻訳においては、目的言語文の訳抜けを考慮しつつ翻訳を行うことが出来たと考えられる。

また全ての翻訳において、オラクルな並び替えを用いることで訳抜けが減少し

表 3.11 日英翻訳における並び替えとその翻訳例

並び替え例	
原言語文	調査が潮間帯に限られたため得られた種類数は少なかった。
BTG	調査が数は種類たられ少なかつ得れたため潮間帯に限らた。
RvNN	得られ少なかつたはた種類数ため潮間帯にが調査限られた。

参照訳

A few species were obtained, **since the survey was limited** to the intertidal zone.

翻訳例

ベースライン	The number of species obtained in the intertidal zone was small.
+BTG	The number of species obtained was small because the investigation was limited to intertidal zone.
+RvNN	The number of species obtained was small because the survey was limited to the intertidal zone.

た。同時に、英独対、英チェコ対、英露対において、ベースラインと比較して重複訳が増加した。この結果より、オラクルな並び替えによる事前並び替え位置表現は訳抜けを減少させるが、不必要的単語を生成してしまう傾向にあることがわかる。

表 3.10、3.11、3.12、3.13 に日英翻訳における翻訳例を示す。表 3.10 の例では、ベースラインによる翻訳文は “electroconductive table mat and floor mat and list strap are listed” というフレーズが欠落しており、原言語文の情報が翻訳文では抜けている。一方で、事前並び替え位置表現を使用することで “electroconductive table mat and floor mat and list strap are listed” というフレーズが出力されている (RvNN を使用したモデルでは “this paper mentions table mat and floor mat of the conductivity” というフレーズが出力されており、BTG を使用したモデルでは “antistatic table mat and floor mat of electroconductive are mentioned as countermeasure” というフレーズが出力されている)。表 3.11 の例においても、ベースラインによる翻訳文では “since the survey was limited” というフレーズが出力されておらず原言語文の情報が欠落しているが、事前並び替え位置表現を使用することで原言語文の情報を損なわずに翻訳文が出力出来ている。

表 3.12 日英翻訳における並び替えとその翻訳例

並び替え例	
原言語文	抗生素を投与したが、突然のショック状態となり、心・呼吸停止となり死亡した。
BTG	抗生 剤 を 投与 し し が た ショッ ク の 状態 と 突然， な り 停止 と な り 死亡 呼吸 ・ 心 ， た。
RvNN	剤 を 抗生 投与 し た が， の 突然 ショッ ク 状態 と な り， 心 ・ 呼吸 停止 と な り 死亡 し た。

参照訳

Though antibiotics were administered, sudden shock state and
cardiac and respiratory arrest were caused and the patient was dead.

翻訳例

ベースライン	Though antibiotics were administered, he suddenly fell into shock state, and died of heart and respiratory arrest.
+BTG	Though antibiotics were administered, he suddenly fell into shock state and died of heart and respiratory arrest.
+RvNN	Although antibiotics were administered, he suddenly fell into shock state and died.

表 3.12 の例では、ベースラインによる翻訳文は参照訳とほとんど同じである。しかし、RvNN による事前並び替え位置表現では “cardiac and respiratory arrest were caused.” というフレーズに当たる部分が出力されていない。表 3.13 の例では、ベースラインの翻訳文、事前並び替え位置表現の翻訳文とともに原言語文の情報が欠落した文となっている。これは、前述したように、事前並び替えが完璧でないことに起因すると考えられる。よって、今後の課題の一つとして事前並び替えの性能向上が挙げられる。

3.5.4 文長と翻訳の関係

原言語文が長くなると翻訳品質が低下すること知られており、並び替えにおいても同じ現象が起こると考えられる。そのため、原言語文の長さと並び替え及び翻訳品質についてさらに分析を行った。図 3.6、3.7、3.8、3.9 に日英、チェコ英、英独、

表 3.13 日英翻訳における並び替えとその翻訳例

並び替え例	
原言語文	走査型プローブ顕微鏡（SPM）は走査型トンネル顕微鏡（STM）が原型であるが、試料-深針間の様々な相互作用を用いた SPM が研究・開発されてきた。
BTG	走査型プローブ顕微鏡（SPM）は様々な相互作用を用いた SPM がの研究間・開発されて針で、試料-探あるがき走査型トンネル顕微鏡原型が）STM（た。
RvNN	走査型プローブ顕微鏡（SPM）は走査型顕微鏡（STM）が原型あるが、試料-探針た用いを様々な相互作用の SPM が間トンネル研究・開発されてきた。

参照訳

Scanning probe microscopes (SPM) are based on scanning tunnel microscopes (STM), and SPMs using various interactions between samples and probes have been studied and developed.

翻訳例

ベースライン	Scanning probe microscopy (SPM) is a prototype of scanning tunneling microscopy (STM).
+BTG	Scanning probe microscopy (SPM) is a prototype of scanning tunneling microscopy (STM).
+RvNN	Scanning probe microscope (SPM) is a prototype, but SPM using various interactions between sample and probe has been studied and developed.

英露翻訳における原言語文の長さとケンドールの τ および BLEU 値の平均のグラフを載せる。これらは提案手法においてベースラインよりも翻訳精度が向上したものである。

日英翻訳では、原言語文の長さが長くなるほどケンドールの τ が減少していく。特に 60 単語以上の文において、BTG に基づく事前並び替えと RvNN を使用した事前並び替えによるケンドールの τ は並び替えなしの文よりも低くなっている。しかし、20 単語以上の文において、BLEU 値はベースラインよりも高い。また、

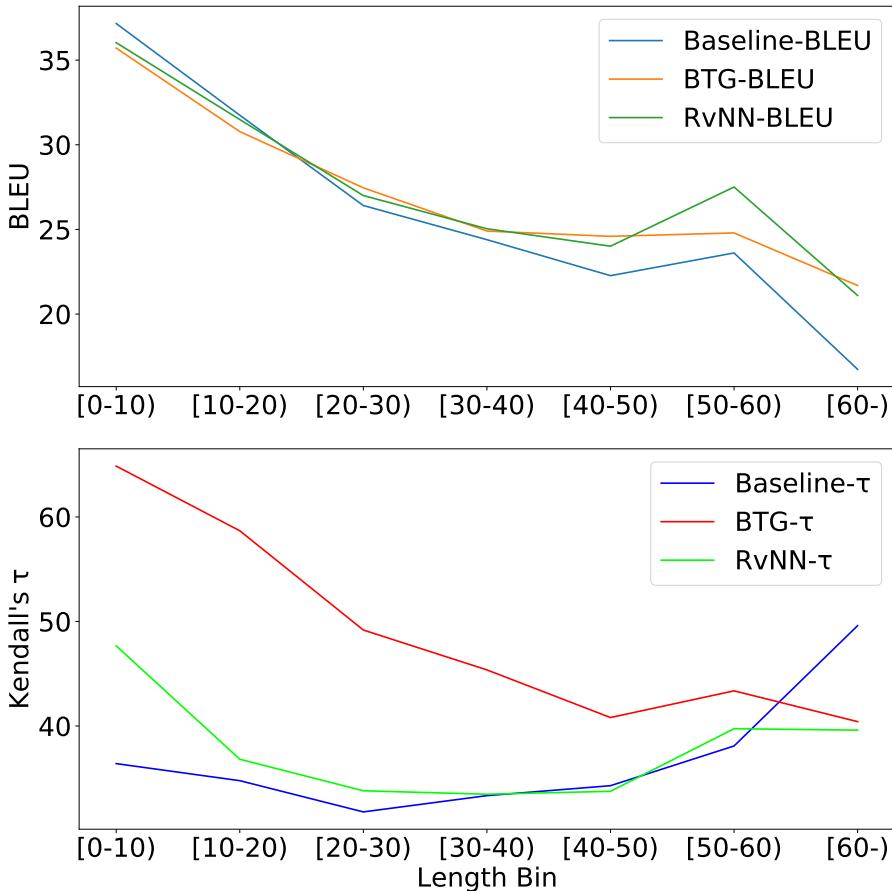


図 3.6 日英翻訳における文長ごとの BLEU 値とケンドールの τ

チェコ英翻訳では、ケンドールの τ は 20 単語以上の文においてベースラインよりも高く、30 単語以上の文において BLEU 値も高い。この結果から、目的言語が英語の翻訳においては中程度の長さ以上の文において事前並び替え位置表現が有効に働くことがわかる。

英独翻訳においては、BTG に基づく事前並び替えと RvNN を使用した事前並び替え位置表現を使用すると常にベースラインよりも低くなっている。しかし、長文において BLEU 値はベースラインよりも高い。英露翻訳では BTG に基づく事前並び替えも RvNN を使用した事前並び替えもケンドールの τ はベースラインより低い。しかし、RvNN を使用した事前並び替えによる BLEU 値は 10 単語以上の文でベースラインよりも高い。また BTG に基づく事前並び替えによる BLEU 値も 50 単語以上の文でベースラインよりも高くなっている。目的言語が英語ではない

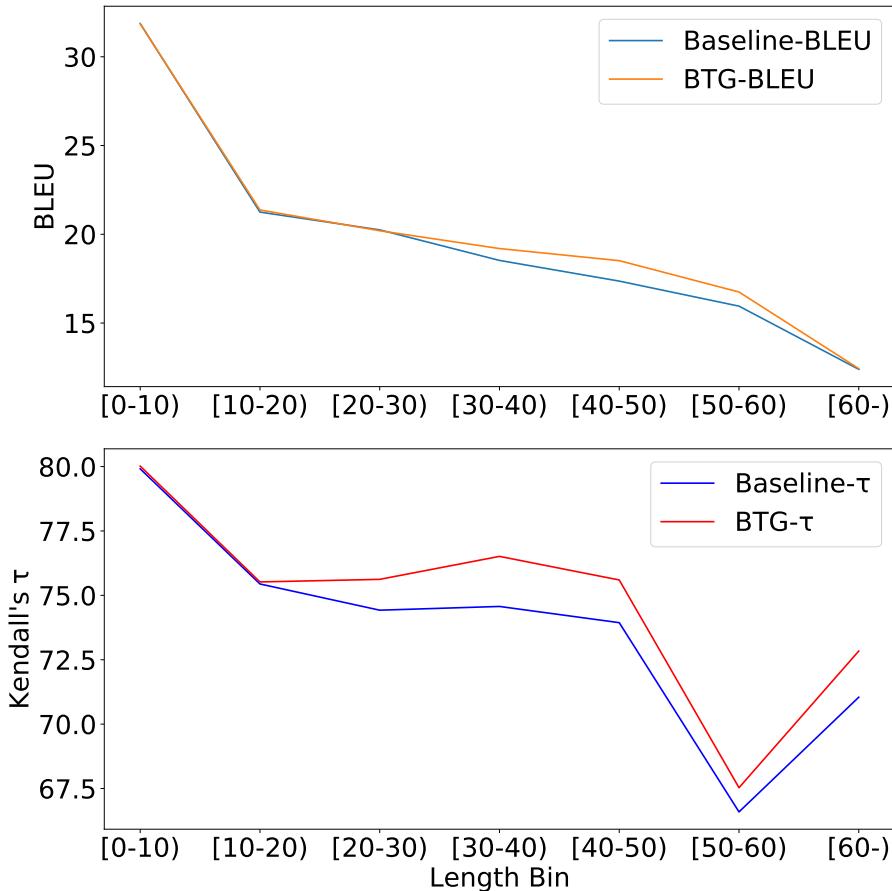


図 3.7 チェコ英翻訳における文長ごとの BLEU 値とケンドールの τ

ことによってケンドールの τ がその言語に大きく影響してしまうため、この結果は日英翻訳やチェコ英翻訳と直接比較できるものではない。今後の課題として、目的言語が英語でない場合の事前並び替え位置表現についてより詳細な調査を行う必要があると考えている。

3.6 おわりに

本章では、Transformer モデルにおいて並び替えの情報を利用するための事前並び替え位置表現を提案した。この手法によって、Transformer モデルは翻訳時に原言語文と目的言語文の両方の語順を考慮することが出来る。実験結果では、Transformer モデルによる日英、チェコ英、英独、英露翻訳において提案手法が翻

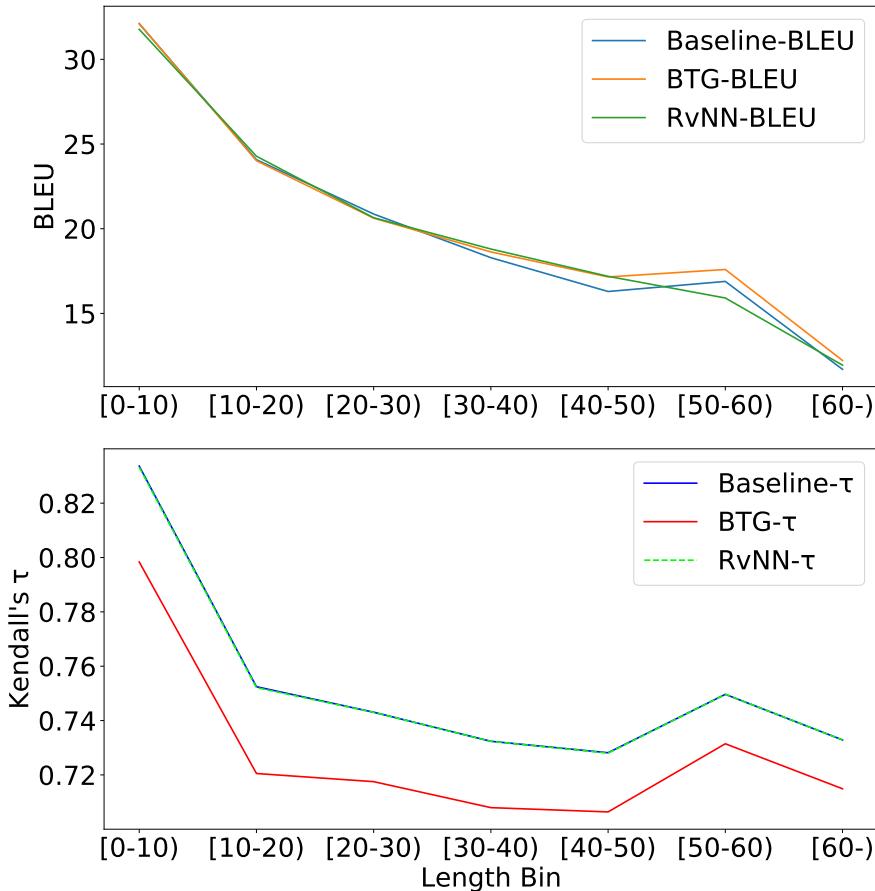


図 3.8 英独翻訳における文長ごとの BLEU 値とケンドールの τ

訳精度を向上させることを確認出来た。また、提案手法では英語方向への翻訳において事前並び替えの精度向上が翻訳精度の向上に寄与することがわかった。さらに、英語方向への翻訳において中程度以上の長さの文で翻訳精度が向上しており、提案手法によって中程度以上の距離の依存関係を捉えられていることがわかった。この手法は著者の知る限り Transformer モデルにおいて並び替えを翻訳に活用した初めての成功例である。

今後の課題として、オラクルな事前並び替えと比較すると事前並び替えモデルの並び替えのケンドールの τ が低いため、事前並び替えモデルの精度向上が挙げられる。また、今回は事前並び替えモデルと Transformer モデルは完全に分離されているために事前並び替えモデルの誤りが Transformer モデルに影響してしまうといったデメリットがある。そのため、事前並び替えモデルと Transformer モデルの

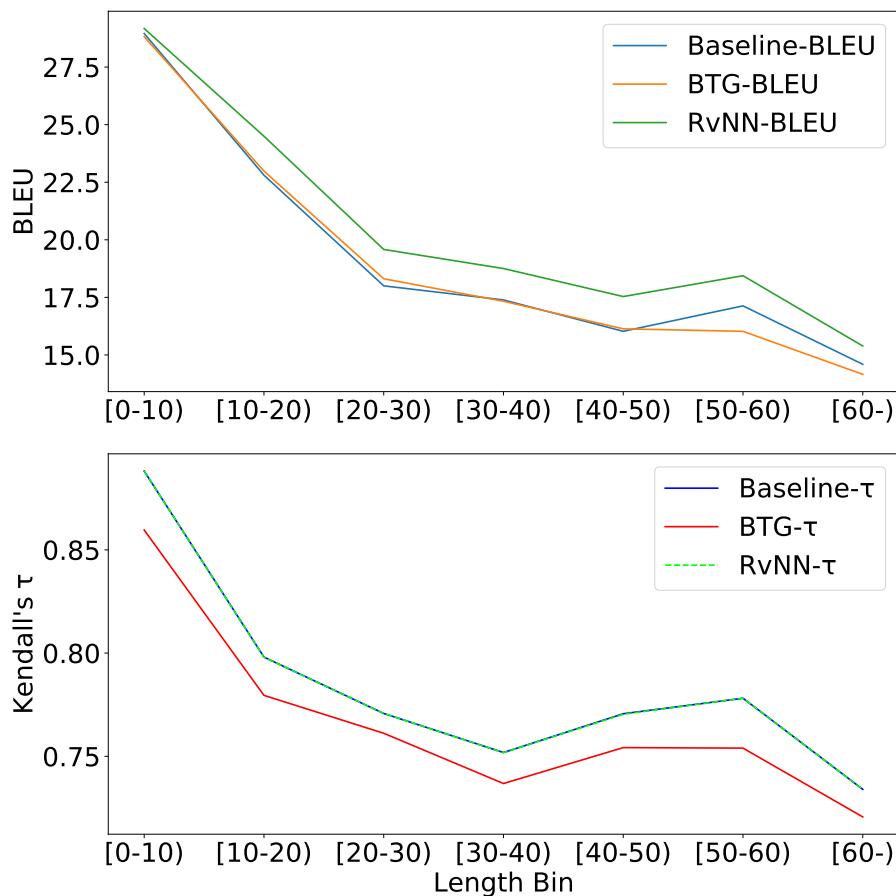


図 3.9 英露翻訳における文長ごとの BLEU 値とケンドールの τ

統合が課題として挙げられる。

第4章 非自己回帰ニューラル機械翻訳における事前並び替えの効果の検証

4.1 はじめに

第3章では自己回帰ニューラル機械翻訳モデルである Transformer に対して、事前並び替え位置表現を適用することで翻訳精度が改善することを示した。近年では、推論時に翻訳文の全単語を一度に出力する非自己回帰ニューラル機械翻訳が提案され [28]、盛んに研究されている。非自己回帰ニューラル機械翻訳のモデルの一つとして、Shu et al. [83] は潜在変数モデルによる非自己回帰ニューラル機械翻訳 (LaNMT) を提案した。目的言語文の不確かさを低次元の潜在変数によってことでモデリングし、その潜在変数に基づいて翻訳文の出力をを行うことでより精度の高い翻訳を可能にしており、最高性能を達成した翻訳手法の一つである。しかし、エンコーダの出力をそのままの順序で使用してデコーダの入力として用いているため、原言語文と目的言語文の語順の相違を考慮出来ない。Ran et al. [77] は Transformer により原言語文の語順を並び替えた隠れ変数を非自己回帰ニューラル機械翻訳のデコーダに入力することで翻訳精度が向上すると報告している。しかし本手法を LaNMT に直接適用することは出来ず、また Transformer による事前並び替えは、自己回帰ニューラル機械翻訳と同様、文の長さに比例した時間がかかるという課題がある。非自己回帰ニューラル機械翻訳において原言語文と目的言語文の語順の相違が考慮出来ないという問題に対し、事前並び替え手法を使用することで、語順を考慮することが出来ると考えられる。一方で自己回帰ニューラル機械翻訳では、事前並び替えを行った文をそのまま入力として使用すると、事前並び替えを行わなかった文を入力として使用した場合と比較して翻訳精度が低下することが報告されており [21, 43, 44]、非自己回帰ニューラル機械翻訳における事前並び替え手法の適用方法は明らかでない。そこで本章では、非自己回帰ニューラル機械翻訳モデルの一つである LaNMT における事前並び替えの効果的な適用手法の検討を行う。

本章は以下のように構成されている。4.2 節では事前並び替えの利用を検討する

非自己回帰ニューラル機械翻訳について説明し、事前並び替えの適用手法について述べる。4.3 節では英日間における翻訳評価実験について述べ、4.4 節では実験結果のより詳細な分析を行う。また、Knowledge distillation を使用した場合、Byte pair encoding を使用した場合の事前並び替えの効果についても実験を行い分析する。4.5 節で本章のまとめと今後の課題を検討する。

4.2 LaNMT における事前並び替えの適用

4.2.1 前提知識: LaNMT

本節では、本論文で使用する潜在変数を使用した非自己回帰ニューラル機械翻訳手法の一つである Latent-variable non-autoregressive translation (LaNMT) モデルについて説明する。図 4.1 に LaNMT の全体図を載せる。LaNMT は大きく三つのサブモデルからなる。一つ目は \mathbf{x} から潜在変数 \mathbf{z} の事前確率分布を予測するモデル、二つ目は \mathbf{x} と \mathbf{y} から \mathbf{z} の事後確率分布を予測するモデル、三つ目は \mathbf{z} から翻訳文の単語数 l_y と \mathbf{y} を予測するモデルである。

LaNMT は式 (4.1) で表される、対数周辺尤度 $\log p(\mathbf{y}|\mathbf{x}) = \log \int p(\mathbf{y}|\mathbf{x}, \mathbf{z})p(\mathbf{z}|\mathbf{x})$ の変分下限を最大化するように学習を行う。

$$\begin{aligned} & \mathbb{E}_{\mathbf{z} \sim q_\phi} \left[\sum_{i=1}^m \log p_\theta(\mathbf{y}_i | \mathbf{x}, \mathbf{z}, l_y) + \log p_\theta(l_y | \mathbf{z}) \right] \\ & - \sum_{j=1}^n \text{KL}[q_\phi(\mathbf{z}_j | \mathbf{x}, \mathbf{y}) || p_\omega(\mathbf{z}_j | \mathbf{x})] \end{aligned} \quad (4.1)$$

ここで $\text{KL}[P||Q]$ は確率分布 P と Q の KL ダイバージェンスを表す。 θ, ϕ, ω はそれぞれデコーダ、潜在変数の事前確率分布 q_ϕ のモデル、潜在変数の事後確率分布 p_ω のモデルのパラメータを指す。式 (4.1) に従い、LaNMT は \mathbf{z} が与えられた時の l_y の条件付き確率、および $\mathbf{x}, \mathbf{z}, l_y$ が与えられた時の \mathbf{y} の条件付き確率の期待値を最大化しつつ、 \mathbf{z} の事前分布と事後分布の KL ダイバージェンスが小さくなるように訓練される。これにより、訓練時は \mathbf{x}, \mathbf{y} から \mathbf{z} が計算できるが、推論時は \mathbf{x} のみを使用して \mathbf{z} を計算しなければならないために \mathbf{z} の計算をうまく行うことが出来なくなる問題を解決している。推論時はまず \mathbf{x} から \mathbf{z} の計算を行い、それをデ

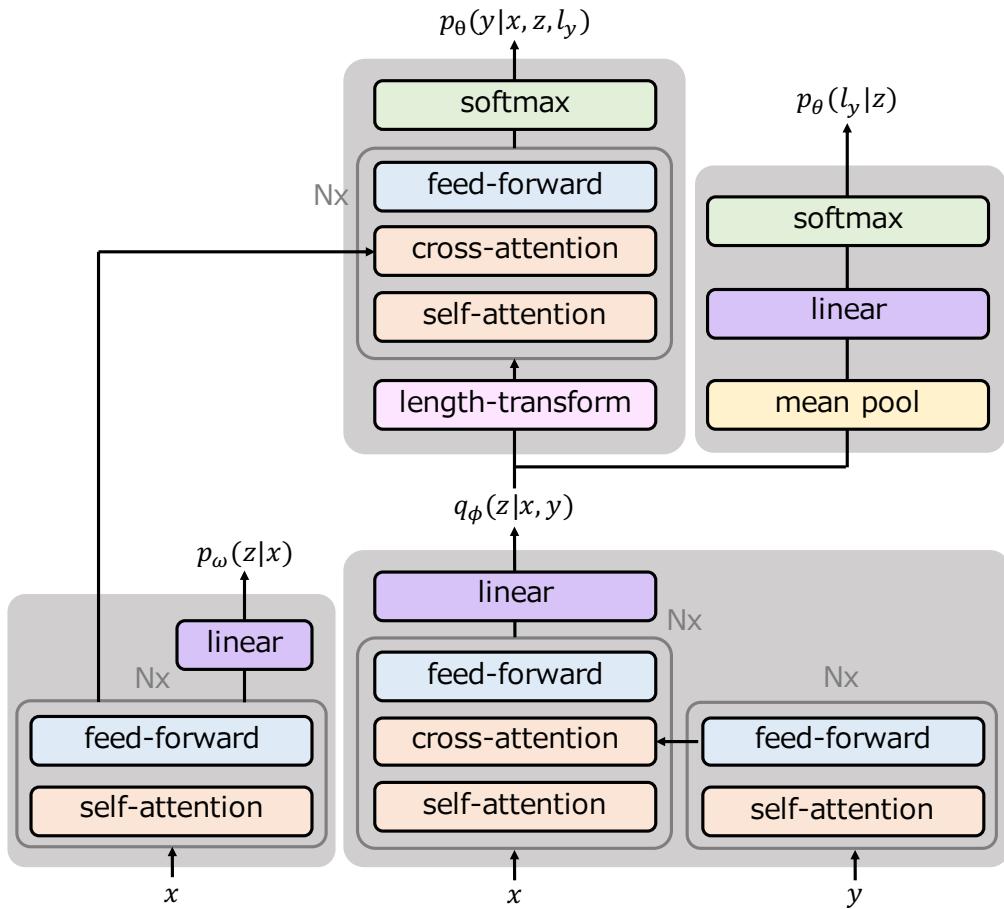


図 4.1 LaNMT のモデル図 (Shu et al. [83] より引用)

コーダの入力として使用して翻訳文 y' の各単語を決定する。それから出力した y' と x を使用して z の再計算を行い、新たな翻訳文の出力を行う。これを一定回数繰り返し、最後の出力を最終的な翻訳文とする。

原言語文の長さと目的言語文の長さは異なることがあるため、デコーダの入力にエンコーダの出力をそのまま使用することは出来ない。そのため、エンコーダの出力を変換してデコーダの入力を得る。LaNMT では length-transform という機構によってエンコーダの出力を変換し、デコーダの入力を生成している。図 4.2 に例を示す。エンコーダの出力 z からデコーダの入力 z' は、以下の式に従って各位置を基準に計算された重みを使用し z の重み付き和によって計算される。

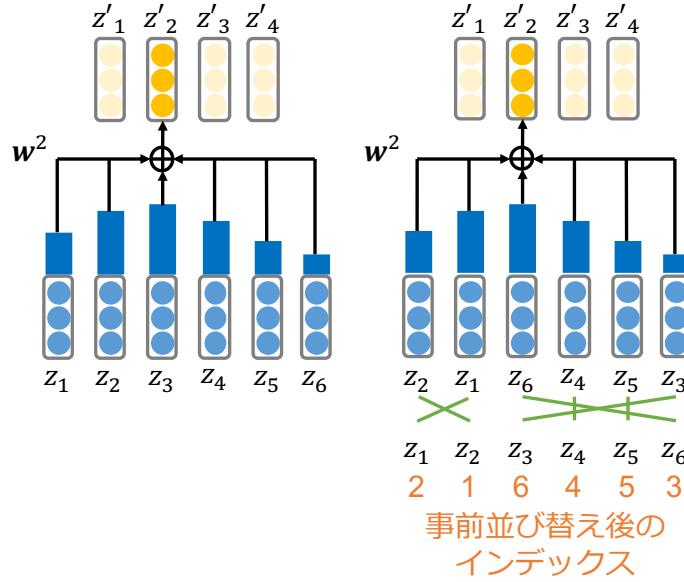


図 4.2 Length-transform の例 (ここではエンコーダの出力は 6 単語、デコーダへの入力は 4 単語としており、2 単語目への入力の重みの例を表している)

$$\mathbf{z}'_j = \sum_{k=1}^{|x|} w_k^j \mathbf{z}_k \quad (4.2)$$

$$w_k^j = \frac{\exp(a_k^j)}{\sum_{k'=1}^{|x|} \exp(a_{k'}^j)}$$

$$a_k^j = -\frac{1}{2\sigma^2} (k - \frac{|x|}{l_y} j)^2 \quad (4.3)$$

j はデコーダの入力のインデックス、 k はエンコーダの出力のインデックス、 a_k^j はデコーダの入力に対するエンコーダの出力のアテンション、 w_k^j はデコーダの入力に対するエンコーダの出力の重みを表す。各 \mathbf{z}' に対する重みは式 (4.3) によってそれぞれの位置を中心とした正規分布によって計算される。

4.2.2 事前並び替えの適用

本研究では、(1) 事前並び替えを行った原言語文をエンコーダに入力するシンプルな手法、(2) 潜在変数 z を並び替える手法、(3) 第 3 章で提案した事前並び替えの位置表現を足し合わせる手法によって事前並び替えの利用の検証を行う。以降では (1)、(2) および (3) の手法について説明する。

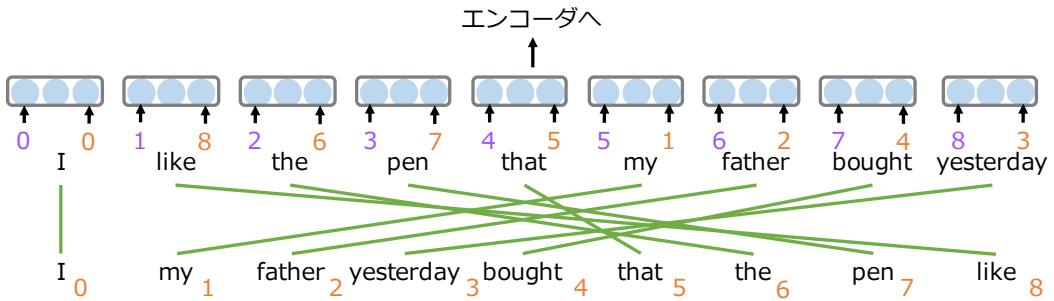


図 4.3 事前並び替え位置表現を足し合わせる手法の概要図

(1) 事前並び替えを入力として使用する手法 事前並び替えを使用するシンプルな手法として、事前並び替えを行った原言語文 $x' = (x_{\text{rpos}^{-1}(1)}, x_{\text{rpos}^{-1}(2)}, \dots, x_{\text{rpos}^{-1}(n)})$ ($\text{rpos}(k)$ は第 3 章と同様に k 番目の単語の並び替えた後のインデックスを返す関数を指す) をそのまま入力として使用する。非自己回帰ニューラル機械翻訳ではエンコーダの出力をそのままの順序で使用するため、事前並び替えをそのまま使用すると逐次翻訳のように翻訳を行えるため翻訳精度が向上すると考えられる。

(2) 潜在変数を並び替える手法 LaNMT では原言語文の潜在変数 z の順番をそのまま使用しているが、そのままで語順の相違を考慮することが出来ない。そこで潜在変数 z の変換の際に以下の式 (4.4) に示すように並び替えた後のインデックスを使用することで語順の相違を考慮した翻訳が可能になると期待される。

$$z'_j = \sum_{k=1}^{|x|} w_k^j z_{\text{rpos}(k)} \quad (4.4)$$

例を図 4.2 の右に示す。オレンジの数字が並び替え後のインデックスを表す。事前並び替えによって予測されたインデックスに基づいて z_k を並び替えてから重み付き和を計算することで、対応した目的言語文の位置に近い $z_{\text{rpos}(k)}$ の重みが大きくなり、並び替えた後の文における周辺単語をより考慮した z' が計算出来る。

(3) 事前並び替え位置表現を足し合わせる手法 ここでは事前並び替えを利用するため、事前並び替え位置表現を足し合わせる手法を設計する。図 4.3 に概要図を示す。単語の左上の紫色の数字は元の文でのインデックスを表し、単語の右上の橙色の数字は事前並び替え後のインデックスを表す。事前並び替えのインデックスによる事前並び替え位置表現は式 (3.5)、(3.6) において、元の文のインデックスの代

わりに以下のように事前並び替え後のインデックスを使用して計算される。

$$PE(\text{rpos}(k), 2q) = \sin(\text{rpos}(k)/10000^{2q/d_m}), \quad (4.5)$$

$$PE(\text{rpos}(k), 2q + 1) = \cos(\text{rpos}(k)/10000^{2q/d_m}), \quad (4.6)$$

第 3 章では Transformerにおいて事前並び替え位置表現を足し合わせることで翻訳精度の向上を達成したが、非自己回帰ニューラル機械翻訳モデルにおける効果は定かではない。非自己回帰ニューラル機械翻訳モデルにおいて、図 4.3 に示すように単語のベクトル表現に元の文の位置表現と事前並び替え位置表現を足し合わせてエンコーダの入力として、エンコーダで語順を考慮した潜在変数の計算が出来ると期待される。

4.3 翻訳評価実験

4.3.1 実験設定

ASPEC コーパス [67] を使用して英日翻訳実験を行った。ASPEC コーパスに含まれている訓練データは 300 万文対、開発データは 1,790 文対、テストデータは 1,812 文対である。前処理として、原言語文もしくは目的言語文が 50 単語以下で、文対の単語数の比が Moses の前処理スクリプトのデフォルト値である 9 以下の条件を満たす文対である約 180 万文対を使用した。英語文の単語分割および品詞タグ付けには Stanza を使用した。日本語の形態素解析は Juman で行った。

事前並び替え手法として BTG に基づく事前並び替え [66] を使用した。事前並び替えモデルは、訓練データのうち上位 50 万文からランダムにサンプリングした 10 万文を使用して 20 イテレーションの訓練を行った。単語クラスタのサイズは 256 に設定し、単語アライメントは MGIZA を使用して計算した。

自己回帰ニューラル機械翻訳モデルのベースラインとして第 3 章と同じ Transformer モデルを使用した。非自己回帰ニューラル機械翻訳モデルのベースラインは Shu et al. の公開モデルである LaNMT¹⁾ を使用し、事前並び替えの適用手法 (2) および (3) は LaNMT 上に実装した。LaNMT のベクトル表現の次元数、層

1) <https://github.com/zomux/lanmt>

表 4.1 英日翻訳における翻訳結果 (それぞれのモデルのベースラインと $p < 0.05$ で有意差があり、精度が向上したものを↑で、精度が低下したものを↓で示す)

		BLEU	RIBES
自己回帰 ニューラル 機械翻訳	事前並び替えなし	35.53	83.68
	BTG	31.20↓	81.15↓
	オラクル	44.84↑	89.85↑
非自己回帰 ニューラル 機械翻訳	事前並び替えなし	22.14	79.42
	BTG	24.01↑	77.92↓
	オラクル	32.39↑	86.26↑
	潜在変数の並び替え (BTG)	13.49↓	78.21↓
	潜在変数の並び替え (オラクル)	15.71↓	81.55↑
	事前並び替え位置表現 (BTG)	15.49↓	78.18↓
	事前並び替え位置表現 (オラクル)	21.05↓	83.75↑

数はエンコーダ、デコーダとともに自己回帰ニューラル機械翻訳モデルと同様の設定とし、潜在変数のベクトルは Shu et al. に従って 8 次元とした。訓練は 5 万ステップ行った。また Shu et al. に同様に、初めに翻訳文を出力した後に、再度その翻訳文を入力として新たに出力した翻訳文を最終的な翻訳文とした。翻訳結果において同じ単語が二回以上繰り返されている場合は二回目以降を除去した。

4.3.2 実験結果

本研究では翻訳精度の評価指標として BLEU 値、また語順の評価指標として RIBES 値を使用した。それぞれの評価値の統計的有意差を検証するため、ブートストラップによる検定 [48] を行った。

実験結果を表 4.1 に示す。ここでオラクルは第 3 章と同様に単語アライメントの計算をした後、アライメントの交差がなくなるように並び替えたものを指す。自己回帰ニューラル機械翻訳モデルでは BTG に基づく事前並び替えによって得られた文をそのまま使用すると、事前並び替えなしと比較して BLEU 値および RIBES 値がそれぞれ 4.33 ポイント、2.53 ポイント低下した。一方、非自己回帰ニューラル機械翻訳モデルでは BTG に基づく事前並び替えによって得られた文をそのまま使用すると、事前並び替えなしと比較して BLEU 値が 1.87 ポイント向上した。また、

オラクルな並び替えを使用して学習を行った翻訳実験では、自己回帰ニューラル機械翻訳モデルにおいても非自己回帰ニューラル機械翻訳モデルにおいても事前並び替えなしと比較して大幅に翻訳精度が向上した(自己回帰ニューラル機械翻訳モデルで BLEU 値が 9.31 ポイント、RIBES 値が 6.17 ポイント、非自己回帰ニューラル機械翻訳モデルで BLEU 値が 10.25 ポイント、RIBES 値が 6.84 ポイント)。これは、翻訳の学習において事前並び替えの情報が有用であることを示唆している。

潜在変数を並び替える方法では、BTG に基づく事前並び替えの順に並び替える方法においてもオラクルな事前並び替えの語順に並び替える方法においても、並び替えなしの場合より BLEU 値が低下した(BTG に基づく事前並び替えの順に並び替えた場合で -8.65 ポイント、オラクルな順に並び替えた場合で -6.43 ポイント)。また、事前並び替え位置表現を使用した場合でも BLEU 値は低下した(BTG に基づく事前並び替えの順を使用した場合で -6.65 ポイント、オラクルな語順を使用した場合で -1.09 ポイント)。翻訳精度が低下した原因として、潜在変数を並び替えることで潜在変数同士の依存関係が崩れてしまったためであると考えられる。また、位置表現を足し合わせる手法では、 z の変換において原言語文の語順をそのまま使用しているため、並び替え後のインデックスが離れている潜在変数の重みが大きくなる場合があるためであると考えられる。

4.4 分析

4.4.1 文長と翻訳精度の関係

非自己回帰ニューラル機械翻訳モデルではエンコーダの出力をそのままの順序でデコーダの入力へと変換するため、長距離の単語の依存構造を捉えることが出来ずより翻訳精度が低下すると考えられる。図 4.4 に原言語文の文長とその翻訳精度を示す。事前並び替えなしの場合と BTG に基づく事前並び替えによって並び替えた文を使用した場合のグラフを比較すると、15 単語以上の文において事前並び替えなしの場合よりも BTG に基づく事前並び替えを使用した文の翻訳精度が向上していることがわかる。このことから、事前並び替えを使用することで長距離の依存関係をうまく捉えた翻訳が出来ていることが分かる。また 60 単語未満の文において潜在変数を並び替える手法よりも事前並び替え位置表現を使用する手法のほうが、

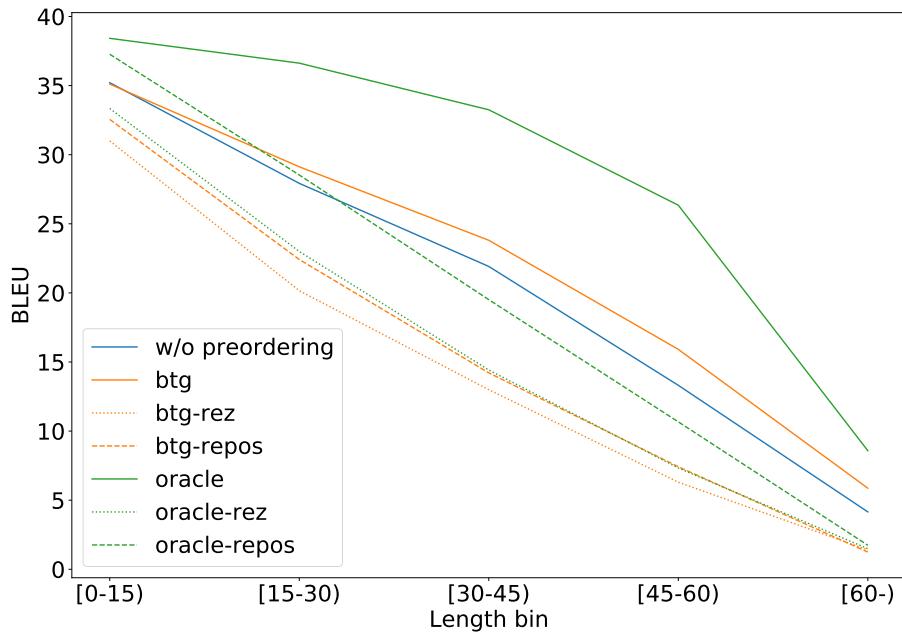


図 4.4 非自己回帰ニューラル機械翻訳モデルにおける文長と翻訳精度の関係 (“rez” は潜在変数を並び替える手法、“repos” は事前並び替え位置表現を使用する手法を指す)

BTG に基づく事前並び替えを使用した場合、オラクルな並び替えを使用した場合どちらにおいても BLEU 値が高いが、60 単語以上の文ではほとんど同じ値となっている。このことから提案手法だけでは長距離の依存関係を考慮することは難しく、事前並び替えの情報を有効に活用する手法の提案が今後の課題として挙げられる。

また、事前並び替えなし、BTG に基づく事前並び替え、オラクルな並び替え、潜在変数を並び替える手法、事前並び替え位置表現を使用する手法での全ての翻訳においてより長い文になるにつれて翻訳精度が低下している。このことから非自己回帰ニューラル機械翻訳モデルでは長文の依存関係を捉えた翻訳はまだ難しく、うまく長距離の依存関係を捉えた翻訳手法の検討が今後の課題として挙げられる。

4.4.2 Knowledge Distillation に対する影響

非自己回帰ニューラル機械翻訳モデルの訓練では、訓練データの参照訳をそのまま使用するのではなく、自己回帰ニューラル機械翻訳モデルで翻訳した出力を使用した訓練を行う Knowledge Distillation [46] によって翻訳性能が大幅に向上去ることが知られている [28, 83]。表 4.2 に Knowledge Distillation を適用し訓練を行った

表 4.2 非自己回帰ニューラル機械翻訳モデルにおける Knowledge Distillation を使用した英日翻訳結果 (それぞれのモデルのベースラインと $p < 0.05$ で有意差があり、精度が向上したものと ↓ で示す。KD の列で ✓ が付いている行は Knowledge distillation を適用し翻訳を行ったものを指す)

	KD	BLEU	RIBES
事前並び替えなし		22.14	79.42
	✓	30.15	82.32
BTG		24.01 [↑]	77.92 [↓]
	✓	27.11	79.49
オラクル		32.39 [↑]	86.26 [↑]
	✓	37.46	88.04

結果を示す。ここで、BTG に基づく事前並び替えによって得られた文で自己回帰ニューラル機械翻訳モデルの訓練を行うと表 4.1 の実験結果のように事前並び替えなしの文を使用した場合よりも精度が低下し、結果として非自己回帰ニューラル機械翻訳モデルの訓練結果も低下すると考えられるため、Knowledge Distillation は全てオラクルな事前並び替えを行った文によって自己回帰ニューラル機械翻訳モデルを訓練し翻訳したものを使っている。先行研究と同様に、非自己回帰ニューラル機械翻訳モデルに対して Knowledge Distillation で訓練を行うことでそのままの対訳コーパスを使用する場合と比較して翻訳精度が向上した(事前並び替えなしで BLEU 値が 8.01 ポイント、RIBES 値が 2.90 ポイント、BTG に基づく事前並び替えを適用した文で BLEU 値が 3.10 ポイント、RIBES 値が 1.57 ポイント、オラクルな事前並び替えを使用した文で BLEU 値が 5.07 ポイント、RIBES 値が 1.78 ポイント)。Knowledge distillation を適用することによって原言語文に対する参照訳のばらつきが減ることが知られている [110]。例えば英日翻訳において、“Thank you”に対して「ありがとうございます」と「どうもありがとうございます」という二つの対訳文が訓練データに含まれている場合、自己回帰ニューラル機械翻訳モデルでは以前に出力した単語に基づいて翻訳を行うため「ありがとうございます」を最初に出力した場合は続けて「ございます」と出力出来る。自己回帰ニューラル機械翻訳モデルの出力を使用することで同じ原言語文に対する翻訳が一つに定まり参照訳のばらつきが減るため、翻訳精度が向上したと考えられる。

表 4.3 非自己回帰ニューラル機械翻訳モデルにおける BPE を使用した英日翻訳結果
(BPE の列で ✓ が付いている行は BPE を適用したもの、KD の列で ✓ が付いている行は
Knowledge Distillation を適用したものと指す)

	BPE	KD	BLEU	RIBES
事前並び替えなし			22.14	79.42
	✓		28.66	81.31
BTG			24.01	77.92
	✓		24.78	78.05
オラクル			28.82	79.99
	✓		32.39	86.26
オラクル			31.08	86.12
	✓	✓	39.01	88.09

4.4.3 Byte Pair Encoding による翻訳精度の変化

近年のニューラル機械翻訳では、単語をそのまま使用するのではなく subword [51, 80] と呼ばれる部分文字列単位に単語を分割してニューラル機械翻訳を訓練する手法もとられている。そこで本節では Byte Pair Encoding (BPE) [80] を非自己回帰ニューラル機械翻訳モデルで使用した際の翻訳への影響を調査した。BPE ツールとして fastBPE²⁾ を使用し、BTG に基づく事前並び替え、オラクルな事前並び替えを使用する文については並び替えを行った後に BPE を適用した。

表 4.3 に翻訳実験結果を示す。BPE を適用すると事前並び替えなしでは BLEU 値が 6.52 ポイント、RIBES 値が 1.89 ポイント向上し、BTG に基づく事前並び替えを使用したものでは BLEU 値が 0.77 ポイント、RIBES 値が 0.13 ポイント向上したが、オラクルな事前並び替えを使用したものでは BLEU 値が 1.31 ポイント、RIBES 値が 0.14 ポイント低下した。今回は単語単位で単語アライメントの計算を行った後にアライメントの交差がなくなるように並び替えを行ってから BPE を適用して部分文字列に分割した。しかし部分文字列単位でのアライメントを計算した場合にアライメントの交差が増えてしまう場合があり、特にオラクルの事前並び替

2) <https://github.com/glample/fastBPE>

表 4.4 英日翻訳における翻訳成功例
並び替え例

原言語文	groupware was introduced as work environment for activating group activity .
BTG	group activity activating for work environment as groupware introduced was .
参照訳	
グループ活動を活性化するための作業環境としてグループウェアを導入した。	
翻訳例	
事前並び替えなし (後処理なし)	活性化活動の作業環境としてグループを紹介した。 活性化活動の作業環境としてグループグループを紹介した。
BTG (後処理なし)	グループ活動を活性化する作業環境としてグループウェアを紹介した。 グループ活動を活性化する作業環境としてグループウェアを紹介した。
潜在変数の 並び替え (後処理なし)	活性化活性をとしてグループを紹介した。 活性化活性活性をとしてとしてとしてグループグループグループを紹介した。
事前並び替え 位置表現 (後処理なし)	グループ活動化として作業ウェアを紹介した。 グループ活動化として作業作業作業作業作業ウェアを紹介した。

えを使用したものでは翻訳精度が下がってしまったと考えられる。また BPE を適用し Knowledge distillation を使用した文による翻訳結果では、事前並び替えなし、BTG に基づく事前並び替えを行った文、オラクルな事前並び替えを行った文全てにおいて Knowledge distillation を使用しなかったものと比較して翻訳精度が向上した。これは第 4.4.2 節と同様の結果であり、事前並び替えを行い BPE を適用した文で学習を行った非自己回帰ニューラル機械翻訳モデルにおいても Knowledge distillation によって翻訳精度が向上することが分かった。

4.4.4 翻訳例

表 4.4、4.5、4.6 に各翻訳モデルの出力例を表す。「後処理なし」の行は各モデルの翻訳文において二回以上繰り返されている単語を削除していない文を指す。表 4.4 の例では、事前並び替えなしの翻訳文において「グループウェア」が出力され

表 4.5 英日翻訳における翻訳失敗例
並び替え例

原言語文	next , the change of hydrogen bond networks which was a basis of the motion of the water was explained .
BTG	next , water the of motion of the a which hydrogen bond networks of the change was basis explained was .
参照訳	
次に，水の運動の基本である水素結合ネットワークの変化を説明した。	
翻訳例	
事前並び替えなし (後処理なし)	次いで，水の運動の基礎である水素結合ネットワークの変化を説明した。 次いで，水の運動運動の基礎である水素結合ネットワークの変化を説明した。
BTG (後処理なし)	次に，水の運動である水素結合ネットワークの変化を基に説明した。 次に，水の運動運動である水素結合ネットワークの変化を基に説明した。
潜在変数の 並び替え (後処理なし)	次に，水の基礎である水素ネットワークの変化を説明した。 次に，水ののの基礎である水素水素水素ネットワークのの変化を説明した。
事前並び替え 位置表現 (後処理なし)	次に，水の基本である水素結合の変化を説明した。 次に，水のの基本である水素水素結合結合の変化を説明した。

ていない。「グループウェア」は参照訳では文の後ろに配置されているが原言語文では“group”が文の後ろ側にあるために語順の相違を考慮出来ずこのような翻訳が出力されたと考えられる。BTGに基づく事前並び替えでは“groupware”が文の後ろへと並び替えられ、その結果 BTGに基づく事前並び替えを使用した翻訳文では「グループウェア」が出力されている。表 4.5 の例は BTGに基づく事前並び替えを使用した翻訳の失敗例である。ここでは参照訳の「基本」に対応した“basis”が後ろに並び替えられてしまったことで、「変化を基に説明した」という間違った翻訳が出力されている。表 4.6 の例では非自己回帰ニューラル機械翻訳において同じ翻訳が繰り返された例である。事前並び替えなしの場合の翻訳文では「肝」と「肝臓」、「流から」が二回繰り返されており、BTGに基づく事前並び替えによって得

表 4.6 英日翻訳における重複訳例
並び替え例

原言語文	the liver is the organ in which the portal blood flow from the digestive system organ flows first , and it is the organ in which metastasis frequency of the digestive cancer is the highest .
BTG	liver the is first the the digestive system organ flows from portal blood flow in which the organ , and digestive cancer the of the metastasis frequency which organ in the highest is is it .

参照訳

肝臓は消化器系臓器からの門脈血流が最初に流入する臓器であり、消化器癌の転移頻度が最も高い臓器である。

翻訳例

事前並び替えなし (後処理なし)	肝 肝臓は消化器系の流から門脈流から最初臓器であり、消化器癌の転移頻度最も臓器である。 肝 肝臓は消化器系の流から門脈流から最初最初臓器であり、消化器癌の転移頻度頻度最も最も臓器臓器臓器である。
BTG (後処理なし)	肝臓は消化器臓器からの門脈血流である臓器であり、消化器癌の転移頻度が最も臓器最も臓器である。 肝臓は消化器臓器からの門脈血流流である臓器であり、消化器癌の転移頻度が最も臓器臓器臓器最も臓器である。
潜在変数の並び替え (後処理なし)	肝 は消化系から血臓器であり消化器癌転移頻度臓器である。 肝 はは消化消化系系からからから血血血血臓器臓器でありあり消化消化器癌転移頻度頻度臓器臓器臓器臓器である。
事前並び替え位置表現 (後処理なし)	肝臓は、消化管臓器から門脈からする肝系で、消化転移の頻度である。 肝臓は、消化管臓器からから門門脈からからからする肝系でで、消化消化転移転移転移転移転移の頻度である。

られた文を使用した翻訳では「最も臓器」というフレーズが二回繰り返されている。このように、非自己回帰ニューラル機械翻訳では自己回帰ニューラル機械翻訳と異なり以前に出力した翻訳を考慮することが難しいため、同じフレーズを繰り返してしまう重複訳が多くなる傾向にある。

表 4.7 に各翻訳モデルの出力した文の 1 単語あたりの平均繰り返し回数を示す。事前並び替えなしと比較すると BTG による翻訳文では 1 単語あたりの平均繰り返し回数が増加しており、事前並び替えを使用することで重複訳が多くなる傾向にあ

表 4.7 非自己回帰ニューラル機械翻訳モデルにおける 1 単語あたりの平均繰り返し回数

事前並び替えなし	BTG	潜在変数の 並び替え	事前並び替え 位置表現
0.1093	0.1571	0.3659	0.3243

ることがわかる。また潜在変数を並び替える手法、事前並び替え位置表現を使用する手法では 1 単語あたりの平均繰り返し回数がベースラインの約 3 倍に増加している。実際に表 4.6 の翻訳例において、潜在変数を並び替える手法では「は」、「消化」、「系」、「から」、「血」、「臓器」、「あり」、「頻度」、「臓器」という単語が繰り返され、事前並び替え位置表現を使用する手法では「から」、「門」、「で」、「消化」、「転移」という単語が繰り返し出力されている。検討手法によって非自己回帰ニューラル機械翻訳モデルにおける重複訳の問題が顕著に現れており、この原因の調査は今後の課題とする。

4.5 おわりに

非自己回帰ニューラル機械翻訳では原言語文の語順をそのまま使用して翻訳文を出力するため、語順の相違を考慮することが出来ない。そのため、英日対のように語順が大きく異なる言語対では翻訳精度が低下する。そこで本章では非自己回帰ニューラル機械翻訳手法の一つである LaNMT モデルにおける事前並び替えの適用手法について調査を行った。非自己回帰ニューラル機械翻訳では自己回帰ニューラル機械翻訳の場合と異なり、事前並び替えを行った文をそのまま入力することで翻訳精度が向上することが明らかとなった。一方、潜在変数を並び替える手法や事前並び替え位置表現を足し合わせる手法では翻訳精度が低下し、とくに重複訳が多く見られる傾向にあることが判明した。今後の課題として、語順を予測するモデルを組み込んで潜在変数に語順の情報を取り入れるなど、よりうまく語順の情報を活用した非自己回帰ニューラル機械翻訳の検討が挙げられる。

第 5 章 結論

機械翻訳では、特に英語と日本語のような語順の大きく異なる言語対において翻訳精度が低下するという問題がある。本論文では機械翻訳における語順の問題に着目し、フレーズベース統計的機械翻訳における事前並び替え手法の提案、自己回帰ニューラル機械翻訳における事前並び替え位置表現の提案、非自己回帰ニューラル機械翻訳における事前並び替えの効果の検証を行った。

第 2 章のフレーズベース統計的機械翻訳における事前並び替えでは、ニューラルネットワークである RvNN を使用した人手での特徴量設計が不要な事前並び替え手法を提案した。この手法は先行研究であり人手での特徴量設計に依存する BTG と同等の翻訳性能を達成し、特に英日対のような語順が大きく異なる言語対で翻訳精度が向上したことを実験で確認した。

統計的機械翻訳の後継となるニューラル機械翻訳において、注意機構付き RNN モデルでは並び替え後のインデックスのベクトル表現を使用し翻訳精度が向上したことが報告されていた。しかし現在の機械翻訳でスタンダードである Transformer において事前並び替えが有効に働くかは定かではなかった。そこで第 3 章では Transformer 対して、事前並び替えによって得られたインデックスのベクトル表現を使用する手法を提案した。この手法によって原言語文と目的言語文の両方の語順を考慮した翻訳を行うことが可能になる。翻訳評価実験では、特に各言語から英語方向への翻訳において、事前並び替えが精度よく行えた場合に翻訳精度が向上することが確認出来た。また、中程度の長さ以上の文において提案手法によって翻訳精度が向上していることが分かった。

近年では翻訳にかかる時間を削減することを目的とし、翻訳文を一度に出力する非自己回帰ニューラル機械翻訳が盛んに研究されている。しかし、非自己回帰ニューラル機械翻訳では原言語文のベクトル表現をそのままの順番で使用するため、語順の相違を考慮することが出来ない。そこで第 4 章では非自己回帰ニューラル機械翻訳における事前並び替えの効果の検証を行った。その結果、非自己回帰ニューラル機械翻訳では事前並び替えをそのまま使用しても翻訳精度が向上するこ

とを確認した。一方で、潜在変数を並び替える手法や事前並び替え位置表現を使用する方法では翻訳精度が低下することが明らかとなった。

上述の通り本論文では、機械翻訳における語順の問題の解消を目指し、事前並び替え手法の開発と、フレーズベース統計的機械翻訳モデル、自己回帰・非自己回帰ニューラル機械翻訳モデルに対する事前並び替えの適用に取り組んだ。第1章でも触れたように、機械翻訳において語順の問題は特に英日対のように語順の大きく異なる言語間において障壁となっている。そのため語順の問題の解決のためにこれまで多くの手法が提案されてきた。本論文で提案した手法によって語順を考慮した翻訳が可能になり翻訳精度が向上したが、RvNN を使用した並び替えは外部の構文解析器を必要とする。そこで今後の課題として、構文解析器を使用しない並び替え手法の開発が挙げられる。また、Transformer に対する事前並び替え位置表現は翻訳精度を向上したが、事前並び替え器と Transformer が完全に切り離されているために翻訳精度が事前並び替えの誤りに影響を受けてしまうという問題がある。そのため、事前並び替えモデルと翻訳モデルを統合したモデルのように語順の情報をモデル内部で活用する手法の検討が今後の課題である。

謝辞

まず初めに、始終研究の相談に対して有益な助言を頂き、熱心な指導をして頂いた大阪大学大学院情報科学研究科の荒瀬由紀先生、京都大学大学院情報学研究科のChenhui Chu 先生、大阪大学大学院情報科学研究科の鬼塚真先生、佐々木勇和先生に深く感謝いたします。また、大阪大学大学院情報科学研究科の原隆浩先生、データビリティフロンティア機構の春本要先生には本論文の内容に関し、貴重なご助言を頂きましたことを深く感謝いたします。また、本研究の内容をご審議頂きましたマルチメディア工学専攻の藤原融先生、松下康之先生、下條真司先生、前川卓也先生、安永憲司先生、大倉史夫先生、伊達進先生、小島一秀先生、木戸善之先生、天方大地先生、矢内直人先生、山藤浩明先生に感謝いたします。また、研究の相談に乗っていただき、的確なアドバイスをして頂いたNTTコミュニケーション科学基礎研究所協創情報研究部の永田昌明様に深く感謝いたします。さらに、2.6.1節の比較実験においてソースコードを提供していただいた株式会社みらい翻訳の星野翔様にも心より感謝いたします。また、同窓生や後輩などの研究室のメンバーには常に刺激的な議論を頂き、また精神的にも支えられました。ありがとうございます。また、辛い時に愚痴を聞いてもらったり困った時に相談に乗ってもらった友人たちには非常に感謝しています。博士後期課程まで進学することを許して頂き、様々な相談に乗って頂いた両親、妹たちには特に感謝しています。家族の理解がなければ大阪大学大学院に入学し、ここまで進むことは出来ませんでした。本当にありがとうございます。最後に、本研究を進める上で様々な面で支えて頂いた全ての方に感謝の意を表します。

参考文献

- [1] Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. In *Proceedings of the International Conference on Neural Information Processing Systems (NIPS)*, December 2016.
- [2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference for Learning Representations (ICLR)*, December 2015.
- [3] Ondřej Bojar, Zdeněk Žabokrtský, Ondřej Dušek, Petra Galuščáková, Martin Majliš, David Mareček, Jiří Maršík, Michal Novák, Martin Popel, and Aleš Tamchyna. The Joy of Parallelism with CzEng 1.0. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pp. 3921–3928, May 2012.
- [4] Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. Findings of the 2016 conference on machine translation. In *Proceedings of the Conference on Machine Translation (WMT)*, pp. 131–198, August 2016.
- [5] Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. Findings of the 2015 workshop on statistical machine translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation (WMT)*, pp. 1–46, September 2015.
- [6] Jan A. Botha, Emily Pitler, Ji Ma, Anton Bakalov, Alex Salcianu, David Weiss,

- Ryan McDonald, and Slav Petrov. Natural language processing with small feed-forward networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2879–2885, September 2017.
- [7] Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, Vol. 16, No. 2, pp. 79–85, 1990.
- [8] Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, Vol. 19, No. 2, pp. 263–311, 1993.
- [9] Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, Roldano Catttoni, and Marcello Federico. The iwslt 2015 evaluation campaign. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, December 2015.
- [10] Wanxiang Che, Valentin Spitkovsky, and Ting Liu. A comparison of chinese parsers for stanford dependencies. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 11–16, July 2012.
- [11] Kehai Chen, Rui Wang, Masao Utiyama, and Eiichiro Sumita. Neural machine translation with reordering embeddings. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1787–1799, July 2019.
- [12] Kehai Chen, Rui Wang, Masao Utiyama, and Eiichiro Sumita. Recurrent positional embedding for neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1361–1367, November 2019.
- [13] David Chiang. Hierarchical phrase-based translation. *Computational Linguistics*, Vol. 33, No. 2, pp. 201–228, 2007.
- [14] Chenhui Chu, Toshiaki Nakazawa, Daisuke Kawahara, and Sadao Kurohashi. SCTB: A chinese treebank in scientific domain. In *Proceedings of the Workshop on Asian Language Resources (ALR)*, pp. 59–67, December 2016.

- [15] Michael Collins, Philipp Koehn, and Ivona Kucerova. Clause restructuring for statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 531–540, 2005.
- [16] Automatic Language Processing Advisory Committee. Language and machines: Computers in translation and linguistics. *National Academy of Sciences, National Research Council*, 1966.
- [17] Josep M. Crego and Nizar Habash. Using shallow syntax information to improve word alignment and reordering for SMT. In *Proceedings of the Workshop on Statistical Machine Translation (WMT)*, pp. 53–61, June 2008.
- [18] Josep Maria Crego and José B Mariño. Improving statistical MT by coupling reordering and decoding. *Machine Translation*, Vol. 20, No. 3, pp. 199–215, September 2006.
- [19] Adrià de Gispert, Gonzalo Iglesias, and Bill Byrne. Fast and accurate preordering for SMT using neural networks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 1012–1017, May–June 2015.
- [20] John DeNero and Jakob Uszkoreit. Inducing sentence structure from parallel corpora for reordering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 193–203, July 2011.
- [21] Jinhua Du and Andy Way. Pre-reordering for neural machine translation: Helpful or harmful? *The Prague Bulletin of Mathematical Linguistics*, Vol. 108, pp. 171–182, June 2017.
- [22] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional sequence to sequence learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1243–1252, August 2017.
- [23] Dmitriy Genzel. Automatically learning source-side reordering rules for large scale machine translation. In *Proceedings of the International Conference on Computational Linguistics (COLING)*, pp. 376–384, August 2010.
- [24] Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. Mask-predict: Parallel decoding of conditional masked language models. In *Proceedings*

of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 6112–6121, November 2019.

- [25] Anita Gojun and Alexander Fraser. Determining the placement of german verbs in English-to-German SMT. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pp. 726–735, April 2012.
- [26] Isao Goto, Masao Utiyama, Eiichiro Sumita, Akihiro Tamura, and Sadao Kurohashi. Distortion model considering rich context for statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 155–165, August 2013.
- [27] Spence Green, Michel Galley, and Christopher D. Manning. Improved models of distortion cost for statistical machine translation. In *Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 867–875, June 2010.
- [28] Jiatao Gu, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. Non-autoregressive neural machine translation. In *International Conference on Learning Representations (ICLR)*, April–May 2018.
- [29] Jiatao Gu, Changhan Wang, and Junbo Zhao. Levenshtein transformer. In *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 32, 2019.
- [30] Katsuhiko Hayashi, Katsuhito Sudoh, Hajime Tsukada, Jun Suzuki, and Masaaki Nagata. Shift-reduce word reordering for machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1382–1386, October 2013.
- [31] Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 690–696, August 2013.
- [32] Geoffrey E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, Vol. 313, No. 5786, pp. 504–507, 2006.
- [33] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural*

Computation, Vol. 9, No. 8, pp. 1735–1780, 1997.

- [34] Sho Hoshino, Yusuke Miyao, Katsuhito Sudoh, Katsuhiro Hayashi, and Masaaki Nagata. Discriminative preordering meets Kendall’s τ maximization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pp. 139–144, July 2015.
- [35] Liang Huang, Kevin Knight, and Aravind Joshi. Statistical syntax-directed translation with extended domain locality. In *Conference of the Association for Machine Translation in the Americas (AMTA)*, pp. 66–73, August 2006.
- [36] Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 944–952, October 2010.
- [37] Hideki Isozaki, Katsuhito Sudoh, Hajime Tsukada, and Kevin Duh. Head finalization: A simple reordering rule for SOV languages. In *Proceedings of the Workshop on Statistical Machine Translation and MetricsMATR*, pp. 244–251, July 2010.
- [38] Hutchins J. *Encyclopedia of Language & Linguistics*, chapter Machine Translation: History, pp. 375–383. Elsevier, 2 edition, 2006.
- [39] Laura Jehl, Adrià de Gispert, Mark Hopkins, and William Byrne. Source-side preordering for translation using logistic regression and depth-first branch-and-bound search. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pp. 239–248, 2014.
- [40] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 133–142, New York, USA, 2002.
- [41] Lukasz Kaiser, Samy Bengio, Aurko Roy, Ashish Vaswani, Niki Parmar, Jakob Uszkoreit, and Noam Shazeer. Fast decoding in sequence models using discrete latent variables. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 2390–2399, July 2018.
- [42] Shin Kanouchi, Katsuhito Sudoh, and Mamoru Komachi. Neural reordering

- model considering phrase translation and word alignment for phrase-based translation. In *Proceedings of the Workshop on Asian Translation (WAT)*, pp. 94–103, December 2016.
- [43] Yuki Kawara, Chenhui Chu, and Yuki Arase. Recursive neural network based preordering for english-to-japanese machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), Student Research Workshop*, pp. 21–27, July 2018.
 - [44] Yuki Kawara, Chenhui Chu, and Yuki Arase. Preordering encoding on transformer for translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, Vol. 29, pp. 644–655, 2020.
 - [45] M. G. Kendall. A new measure of rank correlation. *Biometrika*, Vol. 30, No. 1/2, pp. 81–93, 1938.
 - [46] Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1317–1327, November 2016.
 - [47] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference for Learning Representations (ICLR)*, December 2015.
 - [48] Philipp Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 388–395, July 2004.
 - [49] Philipp Koehn and Christof Monz. Manual and automatic evaluation of machine translation between european languages. In *Proceedings on the Workshop on Statistical Machine Translation (WMT)*, pp. 102–121, June 2006.
 - [50] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL-HLT)*, pp. 48–54, May–June 2003.
 - [51] Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 66–75, July 2018.

- [52] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to Japanese morphological analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 230–237, July 2004.
- [53] Jason Lee, Elman Mansimov, and Kyunghyun Cho. Deterministic non-autoregressive neural sequence modeling by iterative refinement. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1173–1182, October–November 2018.
- [54] Uri Lerner and Slav Petrov. Source-side classifier preordering for machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 513–523, October 2013.
- [55] Chi-Ho Li, Minghui Li, Dongdong Zhang, Mu Li, Ming Zhou, and Yi Guan. A probabilistic approach to syntax-based reordering for statistical machine translation. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL)*, pp. 720–727, June 2007.
- [56] Yang Liu, Qun Liu, and Shouxun Lin. Tree-to-string alignment template for statistical machine translation. In *Proceedings of the International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics (COLING-ACL)*, pp. 609–616, July 2006.
- [57] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1412–1421, September 2015.
- [58] Xuezhe Ma, Chunting Zhou, Xian Li, Graham Neubig, and Eduard Hovy. FlowSeq: Non-autoregressive conditional sequence generation with generative flow. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, November 2019.
- [59] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of Annual Meeting of the Association for Com-*

- putational Linguistics (ACL)*, pp. 55–60, June 2014.
- [60] Antonio Valerio Miceli Barone and Giuseppe Attardi. Non-projective dependency-based pre-reordering with recurrent neural network for machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pp. 846–856, July 2015.
 - [61] Tomas Mikolov, Martin Karafiat, Lukas Burget, Jan Cernocky, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1045–1048, September 2010.
 - [62] Yusuke Miyao and Jun’ichi Tsujii. Feature forest models for probabilistic HPSG parsing. *Computational Linguistics*, Vol. 34, No. 1, pp. 35–80, March 2008.
 - [63] Rudra Murthy, Anoop Kunchukuttan, and Pushpak Bhattacharyya. Addressing word-order divergence in multilingual neural machine translation for extremely low resource languages. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 3868–3873, June 2019.
 - [64] Makoto Nagao. A framework of a mechanical translation between japanese and english by analogy principle. In *Proceedings of the International NATO Symposium on Artificial and Human Intelligence*, p. 173–180, 1984.
 - [65] Masaaki Nagata, Kuniko Saito, Kazuhide Yamamoto, and Kazuteru Ohashi. A clustered global phrase reordering model for statistical machine translation. In *Proceedings of the International Conference on Computational Linguistics and the Annual Meeting of the Association for Computational Linguistics (COLING-ACL)*, pp. 713–720, July 2006.
 - [66] Tetsuji Nakagawa. Efficient top-down BTG parsing for machine translation preordering. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pp. 208–218, July 2015.
 - [67] 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 International Conference on Language Resources and Evaluation (LREC)*, pp. 2204–2208, May 2016.
- [68] Graham Neubig, Taro Watanabe, and Shinsuke Mori. Inducing a discriminative parser to optimize machine translation reordering. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 843–853, July 2012.
 - [69] Sergei Nirenburg, Harold L. Somers, and Yorick A. Wilks. *Readings in Machine Translation*, chapter 1, pp. 13–17. MIT Press, 2003.
 - [70] Franz Josef Och. An efficient method for determining bilingual word classes. In *Conference of the European Chapter of the Association for Computational Linguistics*, pp. 71–76, June 1999.
 - [71] Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 160–167, July 2003.
 - [72] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, Vol. 29, No. 1, pp. 19–51, 2003.
 - [73] Yusuke Oda, Graham Neubig, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura. Ckylark: A more robust PCFG-LA parser. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pp. 41–45, June 2015.
 - [74] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 311–318, July 2002.
 - [75] Slav Petrov and Dan Klein. Improved inference for unlexicalized parsing. In *Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 404–411, April 2007.
 - [76] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many human

- languages. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, July 2020.
- [77] Qiu Ran, Yankai Lin, Peng Li, and Jie Zhou. Guiding non-autoregressive neural machine translation decoding with reordering information. *Association for the Advancemnt of Artificial Intelligence (AAAI)*, February 2021.
 - [78] Rico Sennrich and Barry Haddow. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation (WMT)*, pp. 83–91, August 2016.
 - [79] Rico Sennrich, Barry Haddow, and Alexandra Birch. Edinburgh neural machine translation systems for WMT 16. In *Proceedings of the Conference on Machine Translation (WMT)*, pp. 371–376, August 2016.
 - [80] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1715–1725, August 2016.
 - [81] Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, Vol. 27, No. 3, pp. 379–423,623–656, 1948.
 - [82] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 464–468, June 2018.
 - [83] Raphael Shu, Jason Lee, Hideki Nakayama, and Kyunghyun Cho. Latent-variable non-autoregressive neural machine translation with deterministic inference using a delta posterior. *Association for the Advancemnt of Artificial Intelligence (AAAI)*, February 2020.
 - [84] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas (AMTA)*, August 2006.
 - [85] Richard Socher, Cliff Chiung-Yu Lin, Andrew Y. Ng, and Christopher D. Manning. Parsing natural scenes and natural language with recursive neural networks. In *Proceedings of the on International Conference on Machine Learning*

- (*ICML*), pp. 129–136, June–July 2011.
- [86] Katsuhiko Sudoh, Xianchao Wu, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. Post-ordering in statistical machine translation. *Proceedings of the Machine Translation Summit*, pp. 316–323, September 2011.
 - [87] Katsuhito Sudoh and Masaaki Nagata. Chinese-to-japanese patent machine translation based on syntactic pre-ordering for WAT 2016. In *Proceedings of the Workshop on Asian Translation (WAT)*, pp. 211–215, December 2016.
 - [88] Katsuhito Sudoh, Xianchao Wu, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. Syntax-based post-ordering for efficient japanese-to-english translation. *ACM Transactions on Asian Language Information Processing*, Vol. 12, No. 3, August 2013.
 - [89] Xu Sun, Takuya Matsuzaki, Daisuke Okanohara, Jun’ichi Tsujii. Latent variable perceptron algorithm for structured classification. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1236–1242, July 2009.
 - [90] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Proceedings of the International Conference on Neural Information Processing Systems (NIPS)*, pp. 3104–3112, December 2014.
 - [91] Yuto Takebayashi, Chu Chenhui, Yuki Arase, and Masaaki Nagata. Word rewarding for adequate neural machine translation. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, pp. 14–22, October 2018.
 - [92] Christoph Tillmann. A unigram orientation model for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 101–104, May 2004.
 - [93] Roy Tromble and Jason Eisner. Learning linear ordering problems for better translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1007–1016, August 2009.
 - [94] Masao Utiyama and Hitoshi Isahara. A Japanese-English patent parallel corpus. In *Proceedings of the Machine Translation Summit*, pp. 475–482, September

2007.

- [95] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the International Conference on Neural Information Processing Systems (NIPS)*, pp. 5998–6008. December 2017.
- [96] Karthik Visweswariah, Rajakrishnan Rajkumar, Ankur Gandhe, Ananthakrishnan Ramanathan, and Jiri Navratil. A word reordering model for improved machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 486–496, July 2011.
- [97] Chao Wang, Michael Collins, and Philipp Koehn. Chinese syntactic reordering for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 737–745, June 2007.
- [98] Chunqi Wang, Ji Zhang, and Haiqing Chen. Semi-autoregressive neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 479–488, Octover–November 2018.
- [99] Rui Wang, Chenchen Ding, Masao Utiyama, and Eiichiro Sumita. English-myanmar nmt and smt with pre-ordering: Nict’s machine translation systems at wat-2018. In *Proceedings of the Pacific Asia Conference on Language, Information and Computation (PACLIC): Workshop on Asian Translation (WAT)*, December 2018.
- [100] Xing Wang, Zhaopeng Tu, Deyi Xiong, and Min Zhang. Translating phrases in neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1421–1431, September 2017.
- [101] Yiren Wang, Fei Tian, Di He, Tao Qin, ChengXiang Zhai, and Tie-Yan Liu. Non-autoregressive machine translation with auxiliary regularization. *Association for the Advancemnt of Artificial Intelligence (AAAI)*, January–February 2019.
- [102] Bingzhen Wei, Mingxuan Wang, Hao Zhou, Junyang Lin, and Xu Sun. Imitation learning for non-autoregressive neural machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp.

1304–1312, July 2019.

- [103] Fei Xia and Michael McCord. Improving a statistical MT system with automatically learned rewrite patterns. In *Proceedings of the International Conference on Computational Linguistics (COLING)*, pp. 508–514, August 2004.
- [104] Peng Xu, Jaeho Kang, Michael Ringgaard, and Franz Och. Using a dependency parser to improve SMT for subject-object-verb languages. In *Proceedings of the Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 245–253, June 2009.
- [105] Nan Yang, Mu Li, Dongdong Zhang, and Nenghai Yu. A ranking-based approach to word reordering for statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 912–920, July 2012.
- [106] Richard Zens and Hermann Ney. Discriminative reordering models for statistical machine translation. In *Proceedings on the Workshop on Statistical Machine Translation (WMT)*, pp. 55–63, June 2006.
- [107] Jinchao Zhang, Mingxuan Wang, Qun Liu, and Jie Zhou. Incorporating word reordering knowledge into attention-based neural machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1524–1534, July 2017.
- [108] Yuqi Zhang, Richard Zens, and Hermann Ney. Chunk-level reordering of source language sentences with automatically learned rules for statistical machine translation. In *Human Language Technologies: The Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 1–8, April 2007.
- [109] Yang Zhao, Jiajun Zhang, and Chengqing Zong. Exploiting pre-ordering for neural machine translation. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, May 2018.
- [110] Chunting Zhou, Jiatao Gu, and Graham Neubig. Understanding knowledge distillation in non-autoregressive machine translation. In *International Conference on Learning Representations (ICLR)*, April 2020.

- [111] 瓦祐希, Chenhui Chu, 荒瀬由紀. 統計的機械翻訳のための recursive neural network による事前並び替えと分析. 自然言語処理, Vol. 26, No. 1, pp. 155–178, 2019.