言い換えによる機械翻訳のドメイン不適合の緩和

惟高 日向 [1]　梶原 智之 [2]　藤田 篤 [3]　二宮 崇 [2]
[1] 愛媛大学工学部　[2] 愛媛大学大学院理工学研究科　[3] 情報通信研究機構
{koretaka@ai., kajiwara@, ninomiya@}cs.ehime-u.ac.jp　atsushi.fujita@nict.go.jp

概要

本研究では，対象ドメインのデータを用いずに，入力文のドメインと機械翻訳モデルの訓練ドメインの不一致を緩和する手法を提案する．対象ドメインが未知であってもより高品質な翻訳文を得るために，提案手法では入力文に対する複数の言い換えを生成し，各々を翻訳した後にリランキングを行う．日英の機械翻訳における評価実験の結果，未知ドメインにおける翻訳品質の改善を確認できた．

1　はじめに

ニューラル機械翻訳 [1–4] の研究の進展に伴い，DeepL[1] や TexTra[2] など，オンライン機械翻訳サービスの利用も広がりつつある．ただし，機械翻訳の品質は訓練データに依存するため，訓練データと大きく異なる特性を持つ入力文に対しては，翻訳品質が低下する恐れがある．例えばドメインの不一致に対しては，訓練済みの機械翻訳モデルを対象ドメインの対訳コーパスを用いて転移学習するドメイン適応 [5] がよく用いられている．しかし，様々なドメインへの適用を考える場合，対象ドメインごとの転移学習に要する時間や訓練済みモデルの管理などのコストの課題，少資源のドメインにおいて対訳コーパスを用意できないという課題がある．そのため，オンライン機械翻訳サービスのように，多様な入力が想定され対象ドメインを限定できない状況では，転移学習などの既存のドメイン適応の手法を採用することは容易ではない．

本研究では，対象ドメインを限定せずに入力文と機械翻訳モデルの間のドメインの不一致を緩和する手法を提案する．提案手法は，既存のドメイン適応のようにモデルを調整するのではなく，機械翻訳モデルは変更せずに入力文を編集する．具体的には，入力文の前編集により多様な言い換えを生成し，各々を翻訳した中から後処理として最良の翻訳文を選択する．これらの多様な言い換えの中には，ドメインの不一致を緩和するなど，所与の機械翻訳モデルにより適した表現が含まれる可能性に期待している．本手法は，機械翻訳モデルを再訓練する必要がないだけでなく，対象ドメインのデータを用意する必要もないという利点を持つ．日英の機械翻訳における評価実験の結果，機械翻訳モデルの訓練に使用されていない 2 つのドメインの評価用データに対して，提案手法による翻訳品質の改善を確認できた．

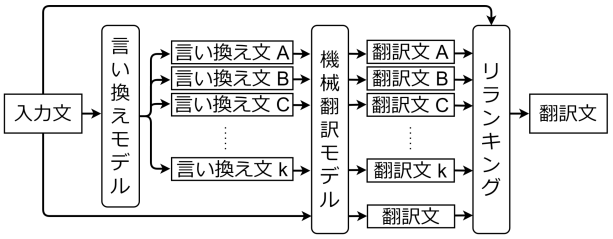

図 1　提案手法の概要

2　関連研究

文書要約 [6] や情報抽出 [7] などの様々な自然言語処理タスクにおいて，入力文の前編集による性能改善が報告されている．機械翻訳においても，手動および自動の前編集が研究されている．機械翻訳のための手動前編集の研究 [8, 9] では，翻訳品質の改善のために効果的な編集事例について分析が行われている．機械翻訳のための自動前編集の研究 [10–12] では，入力文の語句や構造を平易化することで翻訳品質を改善している．前者 [8, 9] は，多数の言い換えおよび翻訳候補を比較することで翻訳品質を改善しているが，人手による大きなコストが必要となる．一方で後者 [10–12] は，自動生成されたひとつの言い換えを翻訳するため，低コストではあるが必ずしも翻訳品質が改善されるとは限らない．本研究では両者の利点を組み合わせ，多数の言い換え文の自動生成によって低コストで翻訳品質を改善する．

1）https://www.deepl.com/translator
2）https://mt-auto-minhon-mlt.ucri.jgn-x.jp/

表 1　言い換えおよびそれに伴う翻訳文の変化の例

	入力文	翻訳文
原文	奈良時代における貴重な作例である。	This is a valuable example of the Nara period.
単語単位の言い換え	奈良時代において貴重な作例である。	It is a valuable example of this work in the Nara period.
文単位の言い換え	奈良時代の貴重な作例。	A valuable example of the Nara period.

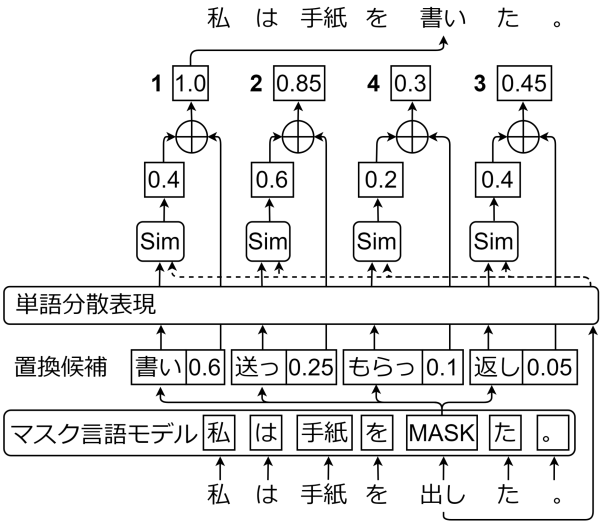

図 2　単語単位の言い換え生成器

3　提案手法

提案手法の概要を図 1 に示す．本研究では，言い換え生成とリランキングを組み合わせることで，入力文のドメインと機械翻訳の訓練ドメインの間のギャップを埋め，翻訳品質を改善する．

提案手法では，まず入力文から複数の言い換え文を生成する．その後，入力文およびこれらの言い換え文を，訓練済み翻訳器を用いてそれぞれ翻訳する．最後に，入力文を考慮して翻訳文候補をリランキングし，翻訳品質の高い候補文を出力する．

3.1　言い換え生成

本ステップでは，任意の機械翻訳モデルがより適切な翻訳候補を生成できるように，所与の入力文に対する多様な言い換えを得る．ある機械翻訳モデルにとっての翻訳しやすい表現および翻訳しにくい表現を事前に知ることは困難だが，生成した複数の言い換えの中に，対象の機械翻訳モデルにとって翻訳しやすい表現が含まれることを期待する．本研究では，単語単位（図 2）および文単位（図 3）の 2 種類の言い換え手法を比較する．各手法の言い換えおよびそれに伴う翻訳文の変化の例を表 1 に示す．

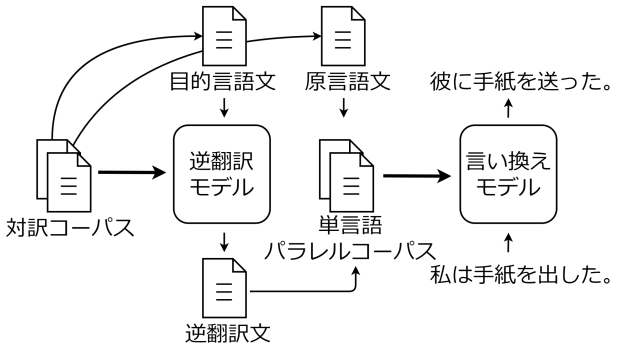

図 3　文単位の言い換え生成器

単語単位の言い換え　単語単位の言い換えでは，BERT [13] などのマスク言語モデルと fastText [14] などの単語分散表現を用いて，局所的な言い換え表現を得る．ここでは，入力文を $|X|$ 個の単語からなる $X = x_1, ..., x_{|X|}$ という単語系列で表す．まず，i 番目の単語をマスクした入力文をマスク言語モデルに入力し，単語穴埋め確率に従って上位 10 件の置換単語の候補を生成する．次に，元の単語 x_i と置換候補の単語の間の意味的類似度を，単語分散表現の余弦類似度を用いて推定する．そして，$(|X| \times 10)$ 種類の置換候補を，マスク言語モデルの単語穴埋め確率および単語分散表現の余弦類似度の和でリランキングし，上位 k 件の置換候補を選択する．最後に，各単語の置換を適用し，k 件の言い換え文を生成する．

文単位の言い換え　文単位の言い換えでは，Transformer [4] などの系列変換モデルを訓練して，大域的な言い換え表現を得る．まず，対象の機械翻訳モデルを訓練した対訳コーパスを用いて，目的言語から原言語への機械翻訳（逆翻訳）モデルを訓練する．これを用いて対訳コーパスの目的言語文を翻訳し，逆翻訳文と原言語文の対からなる単言語パラレルコーパスを自動生成する．そして，この単言語パラレルコーパスを用いて，逆翻訳文から原言語文を生成する言い換えモデル[3)]を訓練する．この言い換えモデルを用いてビーム幅 k のビーム探索を行い，入力文に対する k 件の言い換え文を生成する．

3)　原言語文から逆翻訳文を生成するという逆方向の言い換え生成も検討したが，性能が悪いため採用しなかった．

表 2　教師ありリランキングに用いた素性の一覧（言い換え素性は，単語単位または文単位の片方のみを用いる）

素性	説明
順方向翻訳	順方向の機械翻訳モデルによる入力文から翻訳候補への forced decode 確率
逆方向翻訳	逆方向の機械翻訳モデルによる翻訳候補から入力文への forced decode 確率
言語モデル	対訳コーパスの目的言語側で訓練した言語モデルによる翻訳候補の言語モデル確率
文長	入力文と翻訳候補の文長の差分およびその絶対値
言い換え（単語）	マスク言語モデルの単語穴埋め確率と単語分散表現の余弦類似度の和
言い換え（文）	言い換えモデルによる入力文から言い換え文への forced decode 確率

3.2　リランキング

本ステップでは，入力文の言い換えを介して得られた複数の翻訳候補の中から，高品質な翻訳文を選択する．本研究では，教師なしおよび教師ありの 2 種類のリランキング手法を比較する．

教師なしリランキング　順方向および逆方向の機械翻訳モデルを用意し，入力文と翻訳候補の間で双方向に forced decoding を行う．そして，それぞれの対数尤度の平均値によってリランキングを行う．

教師ありリランキング　複数の素性および K-best Batch MIRA [15] のアルゴリズムを用いて，検証用データ全体に対する BLEU [16] を最大化するリランキングモデルを訓練する．素性には，翻訳候補のリランキングに関する先行研究 [17] でも採用されている forced decode 確率・言語モデルスコア・文長のリランキング素性に加えて，3.1 節で構築した言い換え生成器に基づく素性を用いる．本研究で使用するリランキング素性の一覧を表 2 に示す．

4　評価実験

提案手法の有用性を確認するために，3 つのドメインにおける日英機械翻訳の実験を行った．

4.1　実験設定

データ　日英の機械翻訳モデル（順方向および逆方向）を訓練するために，日英の言語対における最大規模の対訳コーパスである JParaCrawl[4] [18] を用いた．本実験では，JParaCrawl (v3.0) から無作為に抽出した 1,000 万文対を訓練用に，別の 2,000 文対を検証用に，それぞれ使用した．評価用には，未知ドメインのデータとして学術論文ドメインである ASPEC [19] の評価用データ 1,812 文対およびニュースドメインである WMT20 [20] の評価用データ 993 文対，既知ドメインのデータとして JParaCrawl から無作為に抽出した訓練・検証用データとは異なる 2,000 文対を用いた．

日本語における文単位の言い換えモデルを訓練するために，機械翻訳モデルの訓練に使用した 1,000 万文対の対訳コーパスを使用した．逆翻訳モデルを用いて目的言語文を翻訳し，得られた原言語文と対訳コーパス中の原言語文を対にした単言語パラレルコーパスを，言い換えモデルの訓練に使用した．検証用も同様に，機械翻訳モデルの検証用データ 2,000 文対を使用して構築した．

前処理として，日本語文には Mecab[5] (IPADIC) [21]，英語には Moses Tokenizer[6] を用い，単語分割を行った．その後，日本語文および英語文の両方に対して語彙サイズ 32,000 のサブワード分割[7] [22] を適用した．最後に，訓練用データのうち 100 トークンを超える長文を含む文対を除外した．

モデル　機械翻訳モデルおよび文単位の言い換えモデルは，fairseq ツールキット[8] [23] を用いて Transformer モデル [4] を訓練した．モデルの構造は Vaswani ら [4] に倣い，6 層 8 ヘッド 512 次元とした．訓練は，バッチサイズを 70,000 トークンに設定し，最適化手法には Adam [24] を使用して行った．検証用データにおける交差エントロピー損失を 1,500 ステップごとに評価し，この損失が 10 回低下しなかった時点で訓練を終了した．機械翻訳の推論時には，ビーム幅 5 の 1 ベスト出力を用いた．

単語単位の言い換えモデルは，マスク言語モデルと単語分散表現を用いて構築した．マスク言語モデルには東北大 BERT[9] [13] を，単語分散表現には fastText の日本語モデル[10] [14] を，それぞれ用いた．

4）https://www.kecl.ntt.co.jp/icl/lirg/jparacrawl/

5）https://taku910.github.io/mecab/

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

7）https://github.com/glample/fastBPE

8）https://github.com/facebookresearch/fairseq

9）https://huggingface.co/cl-tohoku/bert-base-japanese-whole-word-masking

10）https://fasttext.cc/docs/en/crawl-vectors.html

表 3 BLEU による評価結果（k は翻訳候補の数，*は最上段との統計的有意差（$p < 0.05$），太字は段ごとの最高値）

		ASPEC			WMT20			JParaCrawl		
言い換え	リランキング	$k=5$	$k=10$	$k=20$	$k=5$	$k=10$	$k=20$	$k=5$	$k=10$	$k=20$
-	-	19.9	20.0	20.3	19.7	19.7	19.8	35.3	35.2	35.3
-	教師なし	20.3*	20.4*	20.6*	19.8	**20.2***	19.9	35.3	35.4	**35.6***
単語単位	教師なし	20.6*	20.7*	**21.1***	**19.9**	20.1*	**20.2***	**35.5***	**35.5***	**35.6***
文単位	教師なし	**20.7***	**20.8***	21.0*	**19.9**	20.0	20.1	35.4	35.3	35.4
-	教師あり	20.2*	20.2	20.5	19.6	19.6	19.9	**35.9***	**36.2***	**36.6***
単語単位	教師あり	**20.7***	**20.8***	**20.8***	**20.2***	**20.1***	**20.4***	35.7*	35.6*	35.8*
文単位	教師あり	20.6*	20.4*	20.3	20.1*	20.0	20.1	35.7*	35.7*	35.7*

教師ありリランキングモデルは，Moses [25] の kbmira [15] を用いて構築した．リランキング素性として用いる言語モデルは，機械翻訳モデルの訓練データにおける目的言語側から 4-gram 言語モデルを Moses の lmplz [26] を用いて訓練した．

比較手法　提案手法では，入力文および k 種類の言い換え文の合計 $(k+1)$ 文をそれぞれ翻訳し，それらの翻訳候補をリランキングした．言い換えを行わない比較手法では，ビーム幅 $(k+1)$ のビーム探索を用いて入力文から $(k+1)$ 文の翻訳候補を出力し，それらをリランキングした．

4.2 実験結果

表 3 に，各手法の出力に対して SacreBLEU[11] [27] を用いて得た BLEU スコア [16] を示す．統計的有意差検定には，SacreBLEU に実装されている Paired Bootstrap Resampling [28] を用いた．

まず，評価用データのドメインに注目する．提案手法では未知ドメインである ASPEC および WMT20 において，言い換えを行わない比較手法よりも大きく翻訳品質を改善できた．一方で，既知ドメインである JParaCrawl に対しては，特に教師ありリランキングにおいて，提案手法は言い換えを行わない比較手法ほど品質を改善できなかった．この結果から，既知ドメインでは入力文と機械翻訳モデルの間にドメインの不一致が起こらないため前編集の必要がない，または，既知ドメインにおける教師ありリランキングが有用である，ということが示唆される．

次に，言い換え生成の手法に注目する．未知ドメインにおける教師ありリランキングにおいては単語単位の言い換えが一貫して最高性能を示した．未知ドメインにおける教師なしリランキングにおいては言い換え生成手法による優劣は見られないが，既知ドメインにける教師なしリランキングにおいては単語単位の言い換えが最も高い性能を示した．全体的には，文単位の言い換えよりも単語単位の言い換えの方が効果的であったと言える．

最後に，言い換えの数 k に注目する．教師なしリランキングにおいては k の増加につれて翻訳品質が向上した．一方で，教師ありリランキングにおいては k の大小と翻訳品質の関連は見られなかった．

4.3 言い換えの人手評価

各ドメインの評価用データから 20 文ずつの入力文を無作為に抽出し，単語単位および文単位で $k=5$ の言い換えを生成し，同義性および流暢性を人手評価した．単語単位で 74%，文単位で 86% が同義であり，単語単位で 78%，文単位で 85% が流暢であった．どちらの観点からも，文単位の言い換えの方が高品質であった．前節の実験結果をふまえると，入力文の高品質な言い換えが必ずしも翻訳品質の改善に寄与するとは限らないことが示唆される．

5 おわりに

本研究では，言い換え生成とリランキングを組み合わせることで，入力文のドメインと機械翻訳の訓練ドメインの不一致を緩和し，翻訳品質を改善した．特に，単語単位の言い換え生成と教師ありリランキングの組み合わせによって，機械翻訳モデルの訓練に使用されていない 2 つのドメインにおいて，一貫して最も大きく翻訳品質を改善できた．今後は，本手法をブラックボックス機械翻訳にも適用できるように，リランキングの素性を再検討する．

11） https://github.com/mjpost/sacrebleu

謝辞

本研究は JST（ACT-X，課題番号: JPMJAX1907）および国立研究開発法人情報通信研究機構の委託研究（課題番号：225）による助成を受けたものです．

参考文献

[1] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to Sequence Learning with Neural Networks. In **Proc. of NIPS**, pp. 3104–3112, 2014.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In **Proc. of ICLR**, 2015.

[3] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective Approaches to Attention-based Neural Machine Translation. In **Proc. of EMNLP**, pp. 1412–1421, 2015.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In **Proc. of NIPS**, pp. 5998–6008, 2017.

[5] Chenhui Chu and Rui Wang. A Survey of Domain Adaptation for Neural Machine Translation. In **Proc. of COLING**, pp. 1304–1319, 2018.

[6] Advaith Siddharthan, Ani Nenkova, and Kathleen McKeown. Syntactic Simplification for Improving Content Selection in Multi-Document Summarization. In **Proc. of COLING**, pp. 896–902, 2004.

[7] Richard J. Evans. Comparing Methods for the Syntactic Simplification of Sentences in Information Extraction. **Literary and Linguistic Computing**, Vol. 26, No. 4, pp. 371–388, 2011.

[8] Rei Miyata and Atsushi Fujita. Dissecting Human Pre-Editing toward Better Use of Off-the-Shelf Machine Translation Systems. In **Proc. of EAMT**, pp. 54–59, 2017.

[9] Rei Miyata and Atsushi Fujita. Understanding Pre-Editing for Black-Box Neural Machine Translation. In **Proc. of EACL**, pp. 1539–1550, 2021.

[10] Sanja Štajner and Maja Popovic. Can Text Simplification Help Machine Translation? In **Proc. of EAMT**, pp. 230–242, 2016.

[11] Sanja Štajner and Maja Popović. Improving Machine Translation of English Relative Clauses with Automatic Text Simplification. In **Proc. of ATA**, pp. 39–48, 2018.

[12] Sneha Mehta, Bahareh Azarnoush, Boris Chen, Avneesh Saluja, Vinith Misra, Ballav Bihani, and Ritwik Kumar. Simplify-Then-Translate: Automatic Preprocessing for Black-Box Translation. In **Proc. of AAAI**, pp. 8488–8495, 2020.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In **Proc. of NAACL**, pp. 4171–4186, 2019.

[14] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching Word Vectors with Subword Information. **Transactions of the Association for Computational Linguistics**, Vol. 5, pp. 135–146, 2017.

[15] Colin Cherry and George Foster. Batch Tuning Strategies for Statistical Machine Translation. In **Proc. of NAACL**, pp. 427–436, 2012.

[16] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In **Proc. of ACL**, pp. 311–318, 2002.

[17] Benjamin Marie and Atsushi Fujita. A Smorgasbord of Features to Combine Phrase-Based and Neural Machine Translation. In **Proc. of AMTA**, pp. 111–124, 2018.

[18] Makoto Morishita, Jun Suzuki, and Masaaki Nagata. JParaCrawl: A Large Scale Web-Based English-Japanese Parallel Corpus. In **Proc. of LREC**, pp. 3603–3609, 2020.

[19] Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. ASPEC: Asian Scientific Paper Excerpt Corpus. In **Proc. of LREC**, pp. 2204–2208, 2016.

[20] Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri. Findings of the 2020 Conference on Machine Translation. In **Proc. of WMT**, pp. 1–55, 2020.

[21] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying Conditional Random Fields to Japanese Morphological Analysis. In **Proc. of EMNLP**, pp. 230–237, 2004.

[22] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units. In **Proc. of ACL**, pp. 1715–1725, 2016.

[23] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A Fast, Extensible Toolkit for Sequence Modeling. In **Proc. of NAACL**, pp. 48–53, 2019.

[24] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In **Proc. of ICLR**, 2015.

[25] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open Source Toolkit for Statistical Machine Translation. In **Proc. of ACL**, pp. 177–180, 2007.

[26] Kenneth Heafield. KenLM: Faster and Smaller Language Model Queries. In **Proc. of WMT**, pp. 187–197, 2011.

[27] Matt Post. A Call for Clarity in Reporting BLEU Scores. In **Proc. of WMT**, pp. 186–191, 2018.

[28] Philipp Koehn. Statistical Significance Tests for Machine Translation Evaluation. In **Proc. of EMNLP**, pp. 388–395, 2004.