ニューラル機械翻訳における Iterative Back-Translation を利用したコンパラブルコーパスの活用

山本 優紀　秋葉 友良　塚田 元
豊橋技術科学大学
{yamamoto.yuki.pr, akiba.tomoyoshi.tk, tsukada.hajime.hl}@tut.jp

概要

ニューラル機械翻訳 (NMT) の学習に用いる対訳コーパスの構築法として, 文書単位で対応付けられた 2 つの言語のコーパス (コンパラブルコーパス) から、対応付けられる文ペアを自動的に抽出する手法が広く採用されている. しかし, 文単位で意味が対応するものは少なく, 多くの文は抽出されず捨てられてしまう. 本研究では、対訳コーパスとして抽出されなかった文を含めて, コンパラブルコーパス全体を NMT の学習に活用する手法を提案する. 評価実験により, コンパラブルコーパスでデータ拡張を行うことや, コンパラブル性の利用, Iterative Back-Translation の活用によって翻訳モデルの性能が向上することを確認した.

1　はじめに

機械翻訳の分野では, 深層学習の発達により, ニューラルネットワークを用いるニューラル機械翻訳 (Neural Machine Translation:NMT) が, 従来手法の統計的機械翻訳よりも高い性能を示しており, 様々な研究が行われている. NMT では, ニューラルネットワークで構築した翻訳モデルを, 翻訳元の言語 (原言語) の文と, その訳の言語 (目的言語) の文のペアにした対訳コーパスを用いて学習を行う. NMT は, 対訳コーパスから翻訳に関わる様々な知識を学習するため, 対訳コーパスの質や量が NMT モデルの翻訳性能に大きく影響する. しかし, 大規模な対訳コーパスを人手で作成することは困難という問題点がある.

この問題の解決策として, 既存の日本語と英語の翻訳テキストから対訳コーパスを構築する手法が提案されている.[1] これは, 新聞などの文書単位で対応付けつけられた 2 つの言語コーパス (コンパラブルコーパス) から, 対応付けられる文ペアを自動的に抽出することで対訳コーパスを構築する方法である. しかし, コンパラブルコーパスの中で文単位で意味が対応するものは少なく, 多くの文は抽出されずに捨てられてしまう. 実際, 本論文で使用した PatentMT の調査では 1 つの文書から平均約 27.1 % の文しか抽出されていなかった.

本研究では, 対訳コーパスとして抽出されなかった文を含めて, コンパラブルコーパス全体を NMT の学習に活用する手法を提案する. データ拡張手法として, 逆翻訳 (Back-Translation:BT)[2] や, その拡張手法である Iterative Back-Translation (IBT)[3][4][5] を利用することで, より効果的なデータ拡張手法を探す. さらに, 上記の手法をコンパラブルコーパスのコンパラブル性を活用して行い, その効果を調べる.

2　提案手法

2.1　コンパラブルコーパスの再現

本研究では, 対訳コーパスの抽出元であるコンパラブルコーパスを翻訳モデル学習に活用することを目的とする. しかし, 実験で用いる NTCIR-10 PatentMT[6] のコンパラブルコーパスを直接入手することができなかったため, 以下の方法で対訳コーパスからコンパラブルコーパスを再現した.

1. $C = \{\}$ と初期化する.
2. 対訳コーパス P の各文ペア $(x, y) \in P$ について以下を繰り返す.
   2.1 x と y の抽出元の文書である D_x と D_y を特定する.
   2.2 特定した D_x と D_y を文書ペア (D_x, D_y) とし, C に $C \leftarrow C \bigcup \{(D_x, D_y)\}$ と追加する.

最終的にコンパラブルコーパス $C = \bigcup_{(x,y)\in P}\{(D_x, D_y)\}$ が得られる.

2.2 データ拡張手法

節 2.1 で構築したコンパラブルコーパスを利用して, データ拡張を行う. 本研究では, 4 つの手法でデータ拡張実験を行い, 比較を行うことで, より効果的なコンパラブルコーパスの活用方法を模索する.

2.2.1 Back-Translation

逆翻訳手法 (Back-Translation:BT) は, Sennrich ら [2] の提案した手法である. BT の流れを図 1 に示す. 図 1 では, 言語 X から言語 Y への翻訳モデルの構築を考えている. はじめに, 対訳コーパスを利用して $Y \rightarrow X$ 方向の翻訳モデル $Model_{Y \rightarrow X}0$ を作成する. 次に, このモデルを用いて, 単言語コーパス $C_Y mono$ からサンプリングして得たサブセット $\hat{C}_Y mono$ を逆翻訳し, 翻訳結果 $\hat{C}'_X mono$ を得る. 翻訳結果と元の単言語コーパスを組み合わせて疑似対訳コーパス $(\hat{C}'_X mono, \hat{C}_Y mono)$ を構築する. 構築した疑似対訳コーパスと対訳コーパスを混合し, 言語 X から言語 Y の翻訳モデル $Model_{X \rightarrow Y}1$ を学習する. 以上が BT の流れである. 本研究では, 構築したコンパラブルコーパス $C = \bigcup_{(x,y) \in P}\{(D_x, D_y)\}$ の Y 言語側 $C_Y = \bigcup_{(x,y) \in P}\{D_y\}$ を単言語コーパスとすることで BT を利用する.

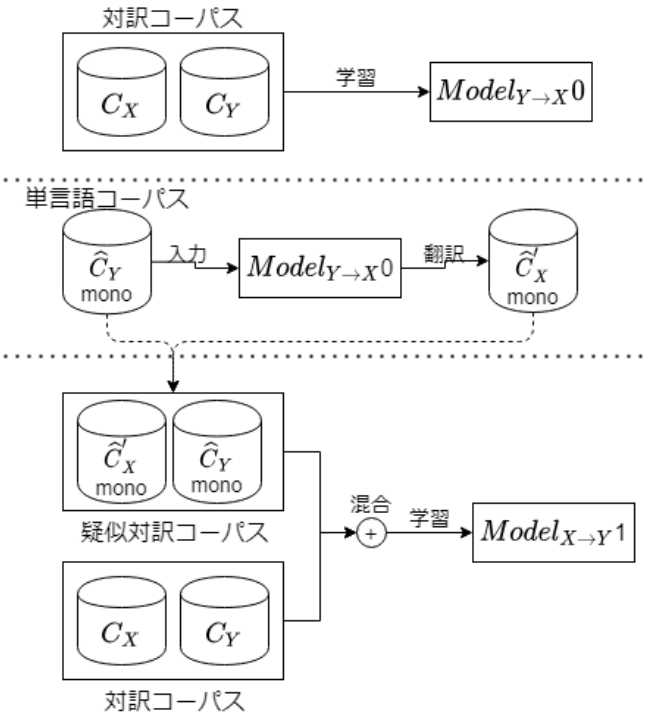

図 1 Back Translation

2.2.2 Iterative Back-Translation

Iterative Back-Translation(IBT) は, 原言語の単言語コーパスと目的言語の単言語コーパスを用いて, BT を双方向かつ反復的に繰り返す手法である. IBT の流れを図 2 に示す. 図では, 言語 X と言語 Y における IBT の流れを示している. IBT は以下のようにしてモデルを学習する.

1. 対訳コーパスを用いて, $X \rightarrow Y$, $Y \rightarrow X$ の各方向の翻訳モデル $Model_{X \rightarrow Y}0$, $Model_{Y \rightarrow X}0$ を学習し, $i \leftarrow 0$ に初期化する.
2. 以下の手順で $Model_{X \rightarrow Y}i$ を更新する.
   2.1 $Model_{Y \rightarrow X}i$ で単言語コーパス $C_Y mono$ からサンプリングして得たサブセット $\hat{C}_Y mono$ を翻訳し, 疑似対訳コーパス $(\hat{C}'_X mono, \hat{C}_Y mono)$ を得る.
   2.2 疑似対訳コーパス $(\hat{C}'_X mono, \hat{C}_Y mono)$ と対訳コーパス (C_X, C_Y) を結合し, $Model_{X \rightarrow Y}i$ を fine-tuning し, $Model_{X \rightarrow Y}(i+1)$ を学習する.
3. ステップ 2 と同様に $Model_{Y \rightarrow X}i$ を更新する.
4. $i \leftarrow i+1$ としてステップ 2 に戻る.

本研究では, BT と同じように, 構築したコンパラブルコーパスを, 単言語コーパスとすることで IBT を利用する.

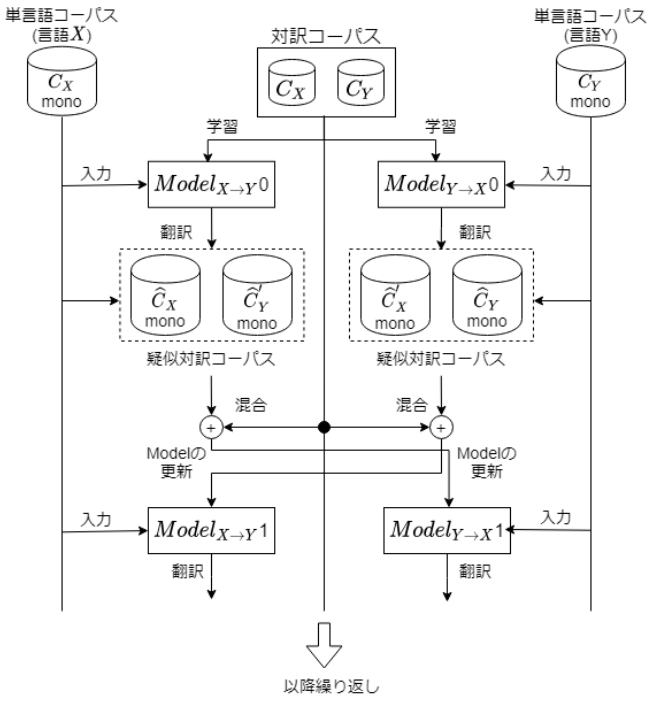

図 2 Iterative Back-Translation

表 1 実験に使用したコーパスサイズ

コーパス	文数
PatentMT(訓練データ)	3,186,254
PatentMT(開発データ)	2,000
PatentMT(テストデータ)	2,300
コンパラブルコーパス (英語)	9,685,920
コンパラブルコーパス (日本語)	9,593,513

表 2 コンパラブルコーパスの効果確認実験の結果

Model	学習データのサイズ (対訳+単言語)	BLEU(dev/test)	
		英日	日英
ベースライン	318k	42.05/44.99	40.29/44.18
コンパラブル	318k + 318k	43.04/46.30	40.23/44.92
	318k + 960k	43.92/47.13	40.35/44.53
NTCIR-10 のベストモデル	—	—/41.82	—/33.08

2.2.3 コンパラブル性を利用した IBT

コンパラブル性を利用した IBT では, 構築したコンパラブルコーパスが文書単位で対応付けられていることを利用して, IBT に利用する両言語の単言語コーパスをコンパラブルになるように選択する方法である. 具体的には, IBT のステップ 2.1 および 3.1 で単言語コーパスから $\hat{C}_X mono$ および $\hat{C}_Y mono$ をサンプリングする際, $\hat{C}_X mono$ と $\hat{C}_Y mono$ が互いにコンパラブルになるように選ぶ. すなわち, 指定されたサンプリングサイズを満たすように最小限のコンパラブルコーパスのサブセット $C_{sub} = \{(D_X, D_Y)\} \subset C$ をサンプリングして, $\hat{C}_X mono \subseteq \cup_{(D_X,D_Y)\in C_{sub}}\{D_X\}$ および $\hat{C}_Y mono \subseteq \cup_{(D_X,D_Y)\in C_{sub}}\{D_Y\}$ のように単言語コーパスを選択する.

3 評価実験

3.1 データセット

本研究では, 使用する大規模なコーパスとして特許機械翻訳テストコレクションである NTCIR10 PatentMT[6] を使用した. PatentMT は特許文書から文を抽出することで構築されている対訳コーパスである. PatentMT の対訳コーパスから, 2.1 節の方法でコンパラブルコーパスを構築した. このとき, 数式を含む文や長い文を除いた. 使用した対訳コーパスと構築したコンパラブルコーパスのサイズを表 1 に示す.

また, PatentMT の対訳コーパスと構築したコンパラブルコーパスの関係を調査した. コンパラブルコーパスの全文書は 66,414 文書である. このうちの 20,485 文書は, 文書内の 10%以下の文しか対訳コーパスとして抽出されていないことがわかった. また, 構築したコンパラブルコーパスを利用することで, 約 67%の文を新しく学習に使用することができることがわかった.

3.2 データセットの前処理

前処理として英語文, 日本語文ともに NFKC 正規化を行った. また, 英語文は Moses[7] に付属するトークナイザーと truecaser でトークナイズ大文字小文字の表記を統一した. 学習前の事前処理として, SentencePiece[8] で語彙サイズを 16,000 でサブワード化を行った.

3.3 ニューラル機械翻訳のパラメータ

NMT システムには Fairseq[9] の Transformer を使用した. エンコーダー及びデコーダは Transformer を 6 層とした. 学習率は 5e-4 とし, Warmup は 4000 ステップ, dropout は 0.1 としている. 損失関数は, ラベル平滑化クロスエントロピーを使用した. 最適化関数は Adam を利用し, パラメータである β_1 を 0.9, β_2 を 0.98 に設定した.

3.4 コンパラブルコーパスの効果

今回構築したコンパラブルコーパスの効果を確認するための実験を行った. PatentMT の対訳コーパスのみで学習した翻訳モデルと, コンパラブルコーパスを利用してデータ拡張を行った翻訳モデルを比較する.

ベースラインは, PatentMT の対訳コーパスのみで学習したものを利用した. コンパラブルコーパスを利用した翻訳モデルは, ベースラインに加え, 全てのコンパラブルコーパスを利用したものと, 対訳コーパスと同サイズである 3,186,254 文をコンパラブルコーパスから抽出したものの 2 つで実験を行った. ベースラインを利用してそれぞれ BT を行い, データ拡張して学習を行った. ベースラインは 20epoch, コンパラブルコーパスを利用した翻訳モデルはどちらも 10epoch の学習を行った. 評価尺度は BLEU[10] を用いる. また, NTCIR-10 のベスト翻訳モデルとも比較を行った.

コンパラブルコーパスの効果確認の実験結果を表

表 3 翻訳モデルの BLEU

学習データ	学習データのサイズ (対訳＋単言語)	英日翻訳モデルの BLEU(dev/test)			日英翻訳モデルの BLEU(dev/test)		
		BT	IBT	IBT(comp)	BT	IBT	IBT(comp)
対訳のみ	10k	17.66/18.91	17.66/18.91	17.66/18.91	21.96/23.18	21.96/23.18	21.96/23.18
対訳 ＋ コンパラブルコーパス	10k + 10k	20.71/21.99	20.71/21.99	23.04/24.65	21.31/22.95	21.31/22.95	23.09/24.73
	10k + 20k	22.32/23.84	21.46/22.27	23.85/25.75	**23.65/25.51**	23.77/25.75	25.04/26.90
	10k + 30k	21.23/23.57	20.78/21.48	23.85/25.31	23.45/24.89	25.03/26.95	26.01/28.09
	10k + 40k	22.53/22.71	**23.56/24.52**	23.31/23.46	22.75/23.94	24.96/26.71	**26.43/28.12**
	10k + 50k	**24.51/24.13**	22.42/23.35	**24.37/25.51**	22.75/24.27	**25.05/26.92**	25.40/27.21

2 に示す. なお, 表 2 のサイズは, 左が対訳コーパスの使用文数, 右が単言語コーパスの使用文数となっている.

コンパラブルコーパスを利用した 2 つの結果がベースラインを上回ったことから, これまで利用されていなかったコンパラブルコーパスを活用することの有効性を示している. また, NTCIR-10 のベスト翻訳モデルと BLEU を比較すると, BLEU を大きく上回っており, 本実験で作成された翻訳モデルは十分な性能があるといえる.

3.5 データ拡張手法の比較

節 2.2 で説明した BT, IBT, コンパラブル性を利用した IBT の 3 つの手法で実験を行い, データ拡張手法の比較を行った. データ拡張は学習データのサイズが少ないほど効果が見られるため, 学習に使用するデータ数を減らして実験を行った. ベースラインは対訳コーパスを 10 万文使用して学習を行った. 提案手法である 3 つのデータ拡張手法では, ベースラインに加え, 10 万文ずつコンパラブルコーパスからサンプリングし, データ拡張を行い, モデルを更新した. モデルの更新後, 新たに 10 万文をコンパラブルコーパスからサンプリングし, 対訳コーパスと混合してデータ拡張を行う. これを繰り返すことで, モデルの更新を進める. モデルの更新は 3 手法とも 5 回行った. 比較は, 開発データで最も高い BLEU スコアのモデルで比較を行った.

データ拡張手法の比較を行うために, BT, IBT, コンパラブル性を利用した IBT の 3 つの手法を行った. 実験の翻訳モデルの学習結果を, 表 3 に示す. なお, 表 3 の学習データサイズは, 左が対訳コーパスの使用文数, 右が単言語コーパスの使用文数となっている. なお, 太字になっている BLEU スコアが, 開発データで最も高い BLEU を示した Model である.

英日方向における各手法の BLEU を比較すると, コンパラブル性を利用した IBT が最も性能が高く, 続いて IBT の性能が高い. 日英方向における各手法の BLEU を比較すると, 英日と同じく, コンパラブル性を利用した IBT が最も性能が高く, 続いて IBT の性能が高い. IBT は, BT と比較して, BLEU が高いことが確認できる. コンパラブル性を利用した IBT は, コンパラブル性を利用していない BT や IBT と比較して, BLEU が高いことが確認できる.

4 結論

対訳コーパスをとして抽出されなかった文を含めたコンパラブルコーパスを利用してデータ拡張を行うことで, 翻訳モデルの性能が向上し, これまで利用されていなかったコンパラブルコーパスを活用することの有効性を確認した. また, コンパラブルコーパスの活用方法として, IBT を利用することの有効性と, 利用する単言語コーパスにコンパラブル性を持たせることの効果を確認することができた.

謝辞

本研究は JSPS 科研費 18H01062 の助成を受けた.

参考文献

[1] 内山将夫. 対訳データの効率的な構築方法. 情報通信研究機構季報 Vol.58, pp. 37–43, 2012.

[2] Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. **In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 86–96, 2016.

[3] Vu Cong Duy Hoang, Phiilpp Koehn, Gholamreza Haffari, and Trevor Cohn. Iterative back-translation for neural machine translation. **In Proceedings of the 2nd Workshop on Neural Machine Translation and Generation**, pp. 18–24, 2018.

[4] Zhirui Zhang, Shujie Liu, Mu Li, Ming Zhou, and Enhong Chen. Joint training for neural machine translation models with monolingual data. In **Proceedings of the AAAI Conference on Artificial Intelligence**, pp. 555–562, 2018.

[5] 森田知熙, 秋葉友良, 塚田元. 双方向の逆翻訳を利用したニューラル機械翻訳の教師なし適応の検討. 情報処理学会研究報告 2018-NL-238 (第 5 回自然言語処理シンポジウム), pp. 1–5, 2018.

[6] Isao Goto, Ka Po Chow, Bin Lu, Eiichiro Sumita, and Benjamin K. Tsou. Overview of the patent machine translation task at the NTCIR-10 workshop. **Proceedings of the 10th NTCIR Conference**, pp. 260–286, 2013.

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

[8] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. **In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**, pp. 66–71, 2018.

[9] 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 **Proceedings of NAACL-HLT 2019: Demonstrations**, 2019.

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