双方向学習と再現学習を統合したニューラル機械翻訳

森下 睦[1]， 鈴木 潤[2]， 永田 昌明[1]
[1] 日本電信電話株式会社 NTT コミュニケーション科学基礎研究所, [2] 東北大学
{morishita.makoto, nagata.masaaki}@lab.ntt.co.jp
jun.suzuki@ecei.tohoku.ac.jp

1 はじめに

ニューラル機械翻訳 (NMT) により機械翻訳文の流暢性は大幅に向上したが，時折原文の意味が一部欠落し忠実に訳出できないという問題が指摘されている [13]．この問題を軽減するために，通常の訳文に基づく誤差に加えて，復号化器の隠れ層を基に原文を再現した際の誤差を用いてマルチタスク学習を行う手法が提案された [12]．本手法により NMT はより原文に忠実に訳文を生成できるようになるが，原文再現に伴うモデルの拡張が必要なためパラメータ数が大きくなってしまう問題がある．

また，近年機械学習の枠組みで機械翻訳のように対をなすタスク (原言語 ↔ 目的言語) を同時に学習することで相乗効果的に両方向のタスクの精度向上を狙う双方向学習 (Dual Learning) と呼ばれる手法が提案されている [4]．本研究では特に符号化器および復号化器を共有しつつ双方向の同時学習を行うモデルレベル双方向学習 (Model-Level Dual Learning) に着目し，これに原文再現に伴う誤差を併用して学習を行うよう拡張することでさらなる精度向上を狙う．実験を通して，提案法により通常の双方向学習と比較してパラメータ数を増加させること無く統計的有意に BLEU スコアが向上することを確認した．

2 関連研究

2.1 ニューラル機械翻訳

初期の NMT モデルは主に再帰的ニューラルネットワーク (RNN) を用いて構築されていた [11, 1, 6]．近年では，学習時間の短縮と性能向上を目的として，RNN を用いず，注意機構 (Attention) とフィードフォワード層のみを用いた NMT モデルが提案された [14]．本モデルは一般に Transformer モデルと呼ばれており，短時間の学習で既存の RNN ベースモデルを上回る翻訳精度を達成できることが知られている．本研究では，

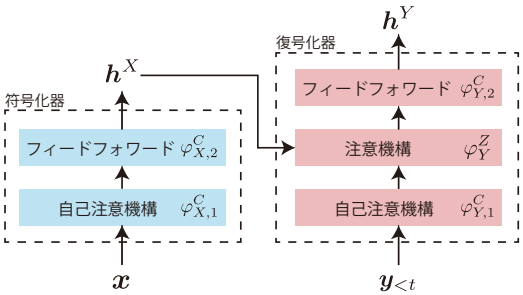

図 1: Transformer モデルの概要図

以降 Transformer モデルを用いて NMT モデルを構築することを考える．

Transformer モデルの概要を図 1 に示す．Transformer モデルは主に (1) 自己注意 (Self-Attention) 層 $\varphi^C_{\cdot,1}(\cdot)$，(2) 符号化器側への注意層 $\varphi^Z_{\cdot}(\cdot)$，(3) フィードフォワード層 $\varphi^C_{\cdot,2}(\cdot)$ の 3 つからなっており以下のように定式化される[1]．

$$\tilde{\boldsymbol{x}} = \varphi^C_{X,1}(\boldsymbol{x}),\ \boldsymbol{h}^X = \varphi^C_{X,2}(\tilde{\boldsymbol{x}}) \tag{1}$$

$$\tilde{\boldsymbol{y}}_{<t} = \varphi^C_{Y,1}(\boldsymbol{y}_{<t}),\ \boldsymbol{h}^Y_t = \varphi^C_{Y,2}(\tilde{\boldsymbol{y}}_{<t} + \varphi^Z_Y(\tilde{\boldsymbol{y}}_{<t}, \boldsymbol{h}^X)) \tag{2}$$

2.2 再現学習モデル

NMT が生成する訳文は流暢ではあるものの，時折原文の一部を訳出しないなど忠実さを欠く問題が指摘されている [13]．これは，統計翻訳 (SMT) では自然に備わっていた，原言語文の情報すべてが翻訳されたことを保証する機構が，NMT モデルでは扱うのが難しく，うまく保証できないことに起因すると考えられる．この問題を解決するために，Tu らは再現学習モデル (Reconstruction Model) を提案した [12]．本モデルでは復号化器の隠れ層を基に原言語文を再現する再現器 (Reconstructor) を導入し，通常の目的言語文に依存する誤差に加え，原言語の再現に伴う誤差 (原文再現誤差) も併用するマルチタスク学習としてモデルを

[1] 符号化器側への注意層は復号化器側にのみ存在する．

学習する．これにより，復号化器の隠れ層に原文のすべての情報を含むようになり，翻訳文もより原文に忠実になることが期待される．

再現器は 2.1 節の復号化器と同様に以下のように定式化される．

$$\tilde{\boldsymbol{r}}_{<t} = \varphi^C_{R,1}(\boldsymbol{x}_{<t}),\ \boldsymbol{h}^R_t = \varphi^C_{R,2}(\tilde{\boldsymbol{r}}_{<t} + \varphi^Z_R(\tilde{\boldsymbol{r}}_{<t}, \boldsymbol{h}^Y)) \tag{3}$$

なお，通常再現器を用いる際は事前に符号化器-復号化器のみを用いて学習を行い，その後再現器を追加してさらに学習を行う[2]．

本モデルの欠点として，通常の NMT モデルに加えて再現器を別途学習する必要があるため，全体のモデルパラメータ数は通常の NMT より大きくなる点が挙げられる．

2.3 双方向学習

近年，様々な対称性を持ったタスク (e.g. 音声 ↔ テキスト変換，日本語 ↔ 英語翻訳) を学習する際に，双方向のモデルを同時に学習することで相乗効果的に双方向それぞれのモデルの性能向上を狙った手法が提案されている [4]．これを一般に双方向学習 (Dual Learning) と呼ぶ．特に機械翻訳は，原言語 → 目的言語および目的言語 → 原言語の双方向モデルの対称性が高いこともあり，双方向学習の題材として頻繁に取り上げられる．

通常の双方向学習ではモデルのパラメータ自体は双方向独立に保持しており，双方向のモデルの出力を利用して学習を行う．一方，Xia らは双方向のモデルの一部を共有することで，双方向学習を実現する手法を提案した [15]．これをモデルレベル双方向学習 (Model-Level Dual Learning) と呼ぶ．特に NMT においては，復号化器の注意機構を除けば符号化器と復号化器はほぼ同じ構造をしているという点に着目し，言語ごとに符号化器，復号化器のパラメータを共有することでこれを実現した．具体的には，式 (2) の $\varphi^C_{Y,1}(\cdot)$ および $\varphi^C_{Y,2}(\cdot)$ を符号化器として，式 (1) の $\varphi^C_{X,1}(\cdot)$ および $\varphi^C_{X,2}(\cdot)$ を復号化器として使用することで，以下の式のように $\boldsymbol{y} \to \boldsymbol{x}$ 方向の翻訳を同時に学習する．

$$\tilde{\boldsymbol{y}} = \varphi^C_{Y,1}(\boldsymbol{y}),\ \boldsymbol{h}^Y = \varphi^C_{Y,2}(\tilde{\boldsymbol{y}}) \tag{4}$$

$$\tilde{\boldsymbol{x}}_{<t} = \varphi^C_{X,1}(\boldsymbol{x}_{<t}),\ \boldsymbol{h}^X_t = \varphi^C_{X,2}(\tilde{\boldsymbol{x}}_{<t} + \varphi^Z_X(\tilde{\boldsymbol{x}}_{<t}, \boldsymbol{h}^Y)) \tag{5}$$

[2]事前学習を行わずに最初からマルチタスク学習として学習する場合，事前学習を行う場合と比べて精度が落ちることが報告されている [7].

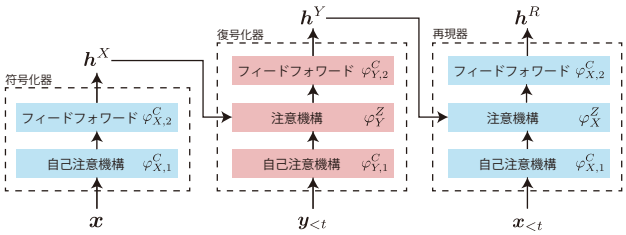

図 2: 提案手法の概要図

本手法で双方向を学習する場合，通常の $\boldsymbol{x} \to \boldsymbol{y}$ のみを学習するモデルと比べてパラメータの増分は $\varphi^Z_X(\cdot)$ に関連する部分だけに留まるため，モデル圧縮としての効果も得られる．本手法の学習時は $P(\boldsymbol{y}|\boldsymbol{x};\theta)$ および $P(\boldsymbol{x}|\boldsymbol{y};\theta)$ の誤差を基にマルチタスク学習を行いモデルパラメータを更新する．

3 提案法

再現学習モデルは翻訳精度の向上が期待できるものの，パラメータの増加が主なデメリットとして上げられる．一方，モデルレベル双方向学習では符号化器と復号化器のパラメータを共有し，両方向を一つのモデルでマルチタスク学習として学習することで翻訳精度の向上およびモデル圧縮の効果がある．

本研究では，これら 2 つの先行研究を統合しモデルレベル双方向学習におけるマルチタスク学習のタスクの一つとして再現学習を導入する．モデルレベル双方向学習は復号化器が符号化器の構造が類似しているという点に着目しこれらのパラメータを共有をした．さらに本研究では，再現器と復号化器も類似していることからこれらのパラメータも共有することを考える．これにより，新たなモデルのパラメータを増やすことなく再現器による原文再現誤差を学習に用いることができる．またマルチタスク学習として全体を学習することで，より双方向の精度が向上することが期待できる．

提案法の概要図を図 2 に示す．図 2 では $\boldsymbol{x} \to \boldsymbol{y}$ 方向への翻訳の例を示しているが，実際は逆方向についても同様の手順で同時に学習を行う．再現器を導入することにより，新たな目的関数は以下の式のようになる．

$$\begin{aligned}\mathcal{L}(\theta) = \arg\max_{\theta} \sum_{n=1}^{N} \Big\{ & \\ \log P(\boldsymbol{y}^{(n)}|\boldsymbol{x}^{(n)};\theta) + \log P(\boldsymbol{x}^{(n)}|\boldsymbol{h}^{Y(n)};\theta) & \\ + \log P(\boldsymbol{x}^{(n)}|\boldsymbol{y}^{(n)};\theta) + \log P(\boldsymbol{y}^{(n)}|\boldsymbol{h}^{X(n)};\theta) \Big\} & \end{aligned} \tag{6}$$

ここで，学習に用いる対訳コーパスは全体で N 文あり

	de-en	fr-en
Train	189,318	208,323
Dev	888	890
Test	3,998	3,455

表 1: 本実験で用いたデータセットに含まれる文数

$\boldsymbol{x}^{(n)}$, $\boldsymbol{y}^{(n)}$ はそれぞれ n 番目の原言語文，目的言語文を示す．また，$\boldsymbol{h}^{Y(n)}$, $\boldsymbol{h}^{X(n)}$ はそれぞれ $\boldsymbol{x}^{(n)} \rightarrow \boldsymbol{y}^{(n)}$, $\boldsymbol{y}^{(n)} \rightarrow \boldsymbol{x}^{(n)}$ 翻訳時の復号化器の隠れ層を示す．

4 実験

提案手法の有効性を実験を通して検証する．本研究では (1) 提案法により翻訳精度の向上は見られるか，(2) モデルのパラメータ数はベースラインと比較してどの程度削減できるか，以上の 2 点を検証する．

4.1 実験設定

データセット　本研究では IWSLT シェアードタスクのデータ[3][2] を基に英語 (en)↔ ドイツ語 (de)，英語 (en)↔ フランス語 (fr) の翻訳実験を行った．NMT の学習には IWSLT2016 の学習用セットを使用した．また，開発用セットとして dev2010, 評価用セットとして tst2012, tst2013, tst2014 を結合したデータを使用した．各データセットの文数を表 1 に示す．

各データセットは Moses Tokenizer[4]により単語分割し，学習用セットについては 50 トークン以上となった文を取り除いた．その後，Sennrich らが提供しているスクリプト[5]を使用してサブワード単位に分割した [10]．この際，BPE のマージ回数は 16,000 回に設定した．

モデルおよびハイパーパラメータ　ベースライン及び提案法は fairseq[6] [3] を基に，これを改良する形で実装した．符号化器および復号化器は 6 層とし，各隠れ層は 512 次元，フィードフォワード層の隠れ層は 1,024 次元とした．Multi-Head Attention の Head 数は 4 とし，各層間のドロップアウト確率は 0.3 に設定した．単語埋め込み層および出力層は Three-way-weight-tying [9] によりパラメータを共有した[7]．最適化手法には Adam を使用し，各ミニバッチはおよそ 4,000 トークンを含むように設定した．最初は再現器をつけずに 150,000 ステップ学習し，その後再現器を追加し合計 250,000 ステップ学習した．学習中は 1,000 ステップごとにモデルを保存し，デコーディング時は最後に出力された 16 モデルのパラメータを平均したモデルを使用した．

デコーディング時はビーム幅 4 のビームサーチを行った．この際，各仮説のスコアを以下の式により補正しリランキングすることで，出力文の文長を調整した．

$$\bar{s}_y = \frac{s_y}{|y|^\alpha} \tag{7}$$

ここで，s_y は各仮説のスコア，$|y|$ は仮説の文長を表す．α は文長の傾向を制御するハイパーパラメータであり，$\alpha < 1.0$ のとき短い文のスコアを高く，$\alpha > 1.0$ のとき長い文のスコアを高く評価するようになる．本実験では，$\alpha = 0.6$ とした．

評価および検定　翻訳精度は BLEU [8] を用いて自動評価した．また，ブートストラップ・リサンプリング法 [5] により統計的有意差の検定を行った．BLEU スコアの計算および統計的有意差の検定には Moses ツールキット付属のスクリプト[8][9]を利用した．

4.2 実験結果

4.2.1 翻訳精度

表 2 にベースラインおよび提案法の BLEU スコアを示す．片方向学習 (ベースライン) では，再現器を導入することによりすべての言語対で 1%水準で統計的有意に BLEU スコアが向上していることがわかる．また，双方向学習を行うことにより，通常の片方向学習を大きく上回る精度が達成できた．さらに，提案法により双方向学習に再現器に基づく原文再現誤差を追加することで本研究で比較した実験設定では最も高い翻訳精度を達成することが確認できた．また，提案法は再現器なしの双方向学習時の翻訳精度と比較して en-fr, fr-en で統計的有意に高い翻訳精度であることを確認した．

4.2.2 モデルパラメータ数

表 3 にベースラインおよび提案法のモデルパラメータ数を示す．

[3] https://wit3.fbk.eu/

[4] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl

[5] https://github.com/rsennrich/subword-nmt

[6] https://github.com/pytorch/fairseq/

[7] 双方向学習時は，双方向の単語埋め込み層および出力層の 4 層すべてで同一のパラメータを使用した．

[8] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

[9] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/analysis/bootstrap-hypothesis-difference-significance.pl

	en-de	de-en	en-fr	fr-en
片方向学習 (ベースライン)	25.57	30.77	39.79	37.74
+ 再現器	28.04‡	33.72‡	41.99‡	40.62‡
双方向学習	29.38	34.78	42.86	42.14
+ 再現器 (提案法)	**29.55**	**34.83**	**43.46**‡	**42.42**†

表 2: テストセットにおける BLEU スコア．表中太字は本実験で最も高い BLEU スコアが得られたモデルを示す．再現器なしの結果と比べて，† は 5%水準で，‡ は 1%水準で統計的に有意な差があることを示す．

	パラメータ数
片方向学習 (ベースライン)	92.9M
+ 再現器	130.7M
双方向学習	52.7M
+ 再現器 (提案法)	52.7M

表 3: 英語 ↔ ドイツ語両方向を学習した際のモデルパラメータ数．提案法により双方向学習時は再現器を追加した場合もパラメータ数の増加はない．

通常の NMT では翻訳方向ごとにモデルを独立して学習する必要があり，両方向の翻訳を行おうとする場合，必要なモデルパラメータ数は大きくなってしまう．一方，双方向学習では，両側のモデルパラメータを共有し一つのモデルとして学習するため，片方向学習に比べてパラメータ数が削減できている．また片方向学習に再現器を追加する場合再現器用のパラメータを新たに追加する必要があるのに対し，提案法では新たにパラメータを増やす必要はない．

ゆえに，提案法により再現器ありの片方向学習モデルに比べてモデルパラメータ数を 59.7%削減しつつ，ベースラインを上回る翻訳精度を達成できていることがわかった．

5 おわりに

本研究ではモデルレベル双方向学習に追加する新たなタスクの一つとして再現学習を導入し，モデルのパラメータ数を増加させずに原文再現誤差を加味して NMT を学習する手法を提案した．実験により，ベースラインモデルと比較して提案法は最大 59.7%パラメータ数を削減しつつ既存手法を有意に上回る BLEU スコアを得られることが示せた．今後の課題として，提案法をベースラインと同等のより大きなモデルパラメータで学習した場合，更なる性能向上が達成できるかを検証したい．

参考文献

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In Proceedings of ICLR, 2015.

[2] Mauro Cettolo, Christian Girardi, and Marcello Federico. WIT3: web inventory of transcribed and translated talks. In Proceedings of EAMT, pp. 261–268, 2012.

[3] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann Dauphin. Convolutional sequence to sequence learning. In Proceedings of ICML, pp. 1243–1252, 2017.

[4] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. Dual learning for machine translation. In Proceedings of NIPS, pp. 820–828, 2016.

[5] Philipp Koehn. Statistical significance tests for machine translation evaluation. In Proceedings of EMNLP, pp. 388–395, 2004.

[6] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In Proceedings of EMNLP, pp. 1412–1421, 2015.

[7] Yukio Matsumura, Takayuki Sato, and Mamoru Komachi. English-Japanese neural machine translation with encoder-decoder-reconstructor. arXiv preprint arXiv:1706.08198, 2017.

[8] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In Proceedings of ACL, pp. 311–318, 2002.

[9] Ofir Press and Lior Wolf. Using the output embedding to improve language models. In Proceedings of EACL, pp. 157–163, 2017.

[10] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In Proceedings of ACL, pp. 1715–1725, 2016.

[11] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In Proceedings of NIPS, pp. 3104–3112, 2014.

[12] Zhaopeng Tu, Yang Liu, Lifeng Shang, Xiaohua Liu, and Hang Li. Neural machine translation with reconstruction. In Proceedings of AAAI, pp. 3097–3103, 2017.

[13] Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Modeling coverage for neural machine translation. In Proceedings of ACL, pp. 76–85, 2016.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Proceedings of NIPS, pp. 6000–6010, 2017.

[15] Yingce Xia, Xu Tan, Fei Tian, Tao Qin, Nenghai Yu, and Tie-Yan Liu. Model-level dual learning. In Proceedings of ICML, pp. 5383–5392, 2018.