逆翻訳によるニューラル機械翻訳の最適化

松村 雪桜　　佐藤 貴之　　小町 守
首都大学東京
{matsumura-yukio, sato-takayuki}@ed.tmu.ac.jp, komachi@tmu.ac.jp

1 はじめに

近年，機械翻訳タスクの中でも，アテンション機構を用いたニューラル機械翻訳 [1] が盛んに研究されている．従来のニューラル機械翻訳は，エンコーダ・デコーダを用いて，原言語文を固定長ベクトルに変換し，その固定長ベクトルから目的言語文を出力する [2]．しかし，1 文を 1 つのベクトルに変換するため，長文をうまく翻訳できない，原言語文のどの単語に注目して翻訳を行うか考慮することができない，という問題点があった．アテンションニューラル機械翻訳では，エンコーダ・デコーダにアテンション機構を加えることにより，エンコーダの各隠れ層の重みを考慮しながら出力単語を予測することができる．アテンション機構を用いることで妥当性が高くなり，ニューラル機械翻訳の精度は向上した．しかしながら，ニューラル機械翻訳には依然として，翻訳時にいくつかの単語が翻訳されず消失してしまう，あるいは不必要な単語が出現したり繰り返されてしまうといった現象がたびたび起きる [8] という問題がある．

また，ニューラル機械翻訳モデルの最適化にはクロスエントロピーが用いられており，翻訳精度を直接最大化していない．Shen ら [7] は，クロスエントロピーを用いたニューラル機械翻訳モデルの最適化が適切な最適化ではない可能性があることを指摘し，翻訳精度を直接最大化するようにニューラル機械翻訳モデルを最適化することで，翻訳精度は向上した．しかしながら，一般的に翻訳指標として用いられる BLEU は，n-gram 適合率に基づき精度を評価し，文長が短いほど低くなる指標であり，精度が向上していたとしても不必要な単語の繰り返しが起きてしまう可能性がある．

そこで本研究では，出力した目的言語文を原言語文に逆翻訳することで，不必要な単語の繰り返しや消失を防ぎつつ，ニューラル機械翻訳モデルを最適化する枠組みを導入した．提案手法では，事前に従来のアテンションニューラル機械翻訳と同様に順方向の翻訳の学習を行った後，デコーダの隠れ層を直接新たなアテンション機構に入力して原言語文に逆翻訳できるように新たなデコーダで学習する．

日英翻訳の実験を行ったところ，従来のアテンションニューラル機械翻訳に比べて，Asian Scientific Paper Excerpt Corpus (ASPEC) では BLEU が 0.43 ポイント，NII Testbeds and Community for Information access Research (NTCIR) では 1.00 ポイント高くなった．また，定性的にも翻訳時における単語の消失や不必要な単語の出現，繰り返しを抑えるといった有用性が示された．

2 関連研究

Shen ら [7] は，翻訳精度を用いてニューラル機械翻訳モデルを最適化するために，従来のクロスエントロピーを用いた目的関数に翻訳精度を考慮する項を追加した．本研究でも，目的関数に新たな項を追加することでニューラル機械翻訳モデルを最適化しているが，翻訳精度ではなく，逆翻訳のクロスエントロピーを考慮している点で異なる．

Niehues ら [6] は，原言語文を統計的機械翻訳によって事前に翻訳し，その出力を原言語文と合わせてニューラル機械翻訳に入力することによって，ニューラル機械翻訳モデルを最適化した．統計的機械翻訳の出力を入力として追加した．また，Mi ら [5]，Feng ら [3] は，原言語文のどの単語をすでに翻訳したかを考慮するための分散表現によるカバレッジベクトルを導入した．これらの研究は，翻訳時における原言語文の情報の消失を抑えることで，単語の消失や不必要な単語の出現，繰り返しというニューラル機械翻訳特有の問題を改善した．本研究では，翻訳時に単語の消失や繰り返しが生じていた場合に正しく逆翻訳できないことを利用し，逆翻訳を用いた最適化を行うことで，これらの問題を解決した．

Meng ら [4] は，アテンション機構をエンコーダ側の隠れ層だけではなくデコーダ側の隠れ層の重みも考慮するものに改良した．また，Feng ら [3] は，新しいアテンションを求める際にこれまでのアテンションを考慮するものに改良した．これらの研究は，アテンション機構の改良により，間接的に単語の消失や繰り返しの削減につながっているが，本研究ではアテンション機構の改良ではなく目的関数の改善を行った．

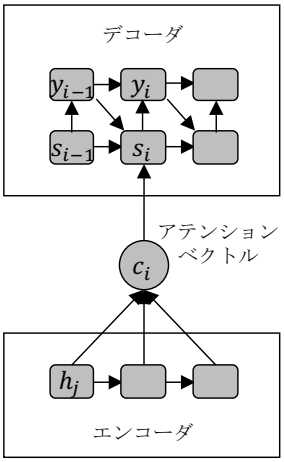

図 1: アテンションニューラル機械翻訳

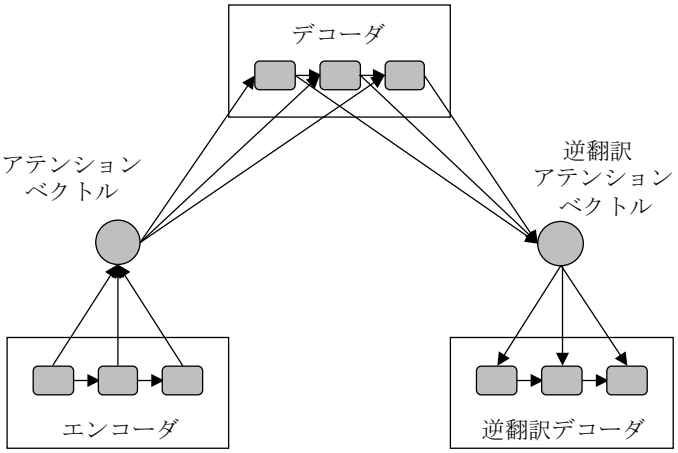

図 2: 提案手法：逆翻訳による最適化

3 アテンションニューラル機械翻訳

ここで，Bahdanau ら [1] が提案したアテンションニューラル機械翻訳モデルについて説明する．

入力された原言語文 ($\boldsymbol{x} = [x_1, x_2, \cdots, x_{|\boldsymbol{x}|}]$) は，リカレントニューラルネットワークを用いたエンコーダで固定長ベクトルに変換される．ステップ t でのエンコーダの隠れ層 h_t は，両方向のリカレントニューラルネットワークを用いて，

$$h_t = [\overrightarrow{h_t}^\top : \overleftarrow{h_t}^\top]^\top \tag{1}$$

と表される．ここで，$\overrightarrow{h_t}$ および $\overleftarrow{h_t}$ は，それぞれ非線形関数 r および r' を用いて，

$$\overrightarrow{h_t} = r(x_t, h_{t-1}), \quad \overleftarrow{h_t} = r'(x_t, h_{t+1}) \tag{2}$$

と計算される． 各隠れ層 $(h_1, h_2, \cdots, h_{|\boldsymbol{x}|})$ は，非線形関数 q を用いることで，

$$v = q([h_1, h_2, \cdots, h_{|\boldsymbol{x}|}]) \tag{3}$$

として固定長ベクトル v に変換される．

エンコーダで変換した固定長ベクトル v は，エンコーダと同様にリカレントニューラルネットワークを用いたデコーダで目的言語文 ($\boldsymbol{y} = [y_1, y_2, \cdots, y_{|\boldsymbol{y}|}]$) へと変換される．$i$ 番目の出力単語の条件付き確率は，非線形関数 f を用いて，

$$p(\hat{y}_i|\boldsymbol{y}_{<i}, \boldsymbol{x}) = f(s_i, y_{i-1}, c_i) \tag{4}$$

と計算され，ステップ i でのデコーダの隠れ層 s_i は，非線形関数 g を用いて，

$$s_i = g(s_{i-1}, y_{i-1}, c_i) \tag{5}$$

として，1 ステップ前の隠れ層 s_{i-1} と単語 y_{i-1}，およびアテンションベクトル c_i を用いて計算される．

アテンションベクトル c_i は，エンコーダの各隠れ層 h_j の重み付き和であり，

$$c_i = \sum_{j=1}^{|x|} \alpha_{ij} h_j \tag{6}$$

で表される．上式における重み α_{ij} は，ソフトマックス関数を用いて全体の和が 1 となるよう正規化される確率分布であり，

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{|x|} \exp(e_{ik})} \tag{7}$$

$$e_{ij} = v_a^\top \tanh(W_a s_{i-1} + U_a h_j) \tag{8}$$

として計算される．ここで，v_a は重みベクトル，W_a，U_a はそれぞれ重み行列である．

なお，式中で用いられる非線形関数には tanh，ReLU（Rectified Linear Unit）などが用いられる．

4 逆翻訳による最適化

本研究では，デコーダの隠れ層を直接新たなアテンション機構に入力し，原言語文に逆翻訳できるように新たなデコーダ（**逆翻訳デコーダ**）で学習する．

順方向のデコーダと同様に，アテンション機構を使用しながら，リカレントニューラルネットワークを用いた逆翻訳デコーダで原言語文 ($\boldsymbol{x}$) へと逆翻訳する．i 番目の出力単語の条件付き確率は，非線形関数 f' を用いて，

$$p(\hat{x}_i|\boldsymbol{x}_{<i}, \hat{\boldsymbol{y}}) = f'(s'_i, x_{i-1}, c'_i) \tag{9}$$

と計算され，ステップ i での逆翻訳デコーダの隠れ層 s'_i は，非線形関数 g' を用いて，

$$s'_i = g'(s'_{i-1}, x_{i-1}, c'_i) \tag{10}$$

として，1 ステップ前の隠れ層 s'_{i-1} と逆翻訳デコーダの単語 x_{i-1}，およびアテンションベクトル c'_i を用いて計算される．

表 1: 対訳コーパスの文数

	ASPEC	NTCIR
学習用	827,503	1,169,201
開発用	1,790	2,741
評価用	1,812	2,300

アテンションベクトル c'_i は，順方向のデコーダの各隠れ層 s_j の重み付き和であり，

$$c'_i = \sum_{j=1}^{|y|} \alpha'_{ij} s_j \tag{11}$$

で表される．上式における重み α'_{ij} は，ソフトマックス関数を用いて全体の和が 1 となるよう正規化される確率分布であり，

$$\alpha'_{ij} = \frac{\exp(e'_{ij})}{\sum_{k=1}^{|y|} \exp(e'_{ik})} \tag{12}$$

$$e'_{ij} = v'_a{}^{\top} \tanh(W'_a s'_{i-1} + U'_a s_j) \tag{13}$$

として計算される．ここで，v'_a は重みベクトル，W'_a，U'_a はそれぞれ重み行列である．

なお，提案モデルの目的関数は，

$$\begin{aligned}\mathcal{L}(\theta) = \frac{1}{N}\sum_{n=1}^{N}\Big\{&\sum_{i=1}^{|y|}\log p(\hat{y}_i^{(n)}|\boldsymbol{y}_{<i}^{(n)}, \boldsymbol{x}^{(n)}, \theta) \\ &+ \sum_{i=1}^{|x|}\log p(\hat{x}_i^{(n)}|\boldsymbol{x}_{<i}^{(n)}, \hat{\boldsymbol{y}}^{(n)}, \theta)\Big\}\end{aligned} \tag{14}$$

となる．ここで，N は学習データ数，θ はモデルにおける全てのパラメータとする．

5 実験

5.1 コーパス

実験に使用したコーパスは，ASPEC および NTCIR-10 の日英コーパスである．ただし，ASPEC に関しては学習用データ約 300 万文のうち，文アライメントの類似度上位 100 万文を用いた．

日本語の単語分割には形態素解析器 MeCab（バージョン 0.996，IPADIC）を用い，英語の単語分割には Moses の Tokenizer を用いた．原言語および目的言語の学習用データから 1 文あたり 40 単語を超える文対を削除したところ，コーパスの文数は表 1 のようになった．なお，学習用データを用いて作成したモデルを開発用データで評価し，最も精度の高いモデルに評価用データを用いた．

5.2 モデル

実験には，ベースラインとしてアテンションニューラル機械翻訳 [1] を参考に実装したモデル (Attention-based Neural Machine Translation; ANMT)[1]，提案手法として逆翻訳によるアテンションニューラル機械翻訳最適化モデル (Back Translate Optimization for Attention-based Neural Machine Translation; BTO-ANMT) を用いた．提案手法では，ベースラインと同様の順方向の翻訳を事前に学習，開発用データで評価して最も精度の高いモデルを選択した後，式 (14) に従って両方向の翻訳を学習し，評価は順方向のみで行った．式 (14) による最適化で BLEU が向上しない場合は，従来のモデルが使用される．

[1] https://github.com/tmu-nlp/NMT2016

表 2: 日英翻訳実験結果

コーパス	手法	BLEU	p 値
ASPEC	ANMT	21.05	-
	BTO-ANMT	21.48	0.04
NTCIR	ANMT	29.12	-
	BTO-ANMT	30.12	0.00

リカレントニューラルネットワークには LSTM を用い，語彙数 30,000，埋め込み層の次元数 512，隠れ層の次元数 512，バッチサイズ 128 のハイパーパラメータに設定した．提案手法でも同様のハイパーパラメータに設定したが，メモリの都合上バッチサイズは 64 に設定して実験を行った．なお，各パラメータの最適化手法には Adagrad（初期学習率 0.01）を用いた．

5.3 結果

実験結果を翻訳指標 BLEU で評価，ブートストラップを用いて 1,000 回有意差検定を行い p 値を測定し，その値を表 2 に示した．実験の結果，ベースラインと比較して，提案手法の BLEU の値が，ASPEC では 0.43 ポイント，NTCIR では 1.00 ポイント高くなった．いずれの結果も統計的に有意であった（$p < 0.05$）．

6 考察

日英翻訳における各モデルの出力例を表 3 に示した．例 1 では，ANMT において “as shown” が消失してしまっているが，BTO-ANMT では近い “as shown in the drawing” が出力されている ．また例 2 では，ANMT において “array” が 4 回出力されてしまっているが，BTO-ANMT では繰り返されることなく，より参照訳に近い文を出力している．しかしながら例 3 では，逆に ANMT において正しく出力されていた “is satisfied” が，BTO-ANMT では消失してしまっている．

ここで，各コーパスおよびモデルにおける単語の出現回数の比較を表 4 に示した．文ごとに単語の出現回数を測定し，参照訳に含まれている単語の場合は参照訳より出現回数が多かった単語の数を (i) に，参照訳に含まれていない単語の場合は文中に 2 回以上出現する単語の数を (ii) に示した．ただし，これらの単語に

表 3: 日英翻訳における各モデルの出力例

例 1：消失の改善	
入力	ダイ 23 は 、 図示 の よう に ダイ 支持 部 29 により 支持 さ れ て いる 。
ANMT	the die 23 is supported by a die support 29 .
BTO-ANMT	the die 23 is supported by a die support 29 *as shown in the drawing* .
参照訳	the die 23 is supported by a die support part 29 *as shown* .
例 2：繰り返しの改善	
入力	入射 光 と 電気 信号 の 間 の 相関 検出 器 を 2 次元 に 配列 する 新しい 形式 の イメージ センサ を 提案 し た 。
ANMT	a new type of image sensor *array array* is proposed which is a *array* of *array* of the correlation between the incident light and the electrical signal .
BTO-ANMT	we propose a new type image sensor which is *arrayed* in two-dimensional correlation *array* between the incident light and the electric signal .
参照訳	this paper proposes the new image sensor in which the correlation detectors between incident light and electric signal are two - dimensionally *arranged* .
例 3：悪化例	
入力	W 1 = 150 nm を 満たし て いる 。
ANMT	W1 = 150 nm *is satisfied* .
BTO-ANMT	W1 = 150 nm .
参照訳	therefore , W1 = 150 nm *is satisfied* .

表 4: 各コーパス，モデルにおける単語出現回数比較

コーパス	手法	(i)	(ii)	(iii)
ASPEC	ANMT	1,222	683	1,377
	BTO-ANMT	1,208	664	1,222
NTCIR	ANMT	2,514	1,095	1,782
	BTO-ANMT	2,214	1,022	1,476

未知語は含まれていない．未知語を意味する unk トークンの全体での出力個数は (iii) に示した．どの場合でも ANMT と比較して BTO-ANMT の同一単語出現回数が少なくなっており，単語の繰り返しは減少していると考えられる．

このように，悪化してしまった例もあるものの，全体的に単語の消失や不必要な繰り返しは減少し，参照訳により近い文を出力していることが確認できた．

7 おわりに

本研究では，逆翻訳によるアテンションニューラル機械翻訳モデルの最適化を提案した．加えて，日英翻訳の実験を通して，既存のアテンションニューラル機械翻訳と性能を比較した．実験の結果，既存のアテンションニューラル機械翻訳に比べて BLEU が有意に向上し，翻訳時における単語の消失や不必要な単語の出現，繰り返しを抑えるという観点からも提案手法の有用性が示された．

参考文献

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. *ICLR*, pages 1–15, 2015.

[2] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *EMNLP*, pages 1724–1734, 2014.

[3] Shi Feng, Shujie Liu, Nan Yang, Mu Li, and Ming Zhou. Improving Attention Modeling with Implicit Distortion and Fertility for Machine Translation. *COLING*, pages 3082–3092, 2016.

[4] Fandong Meng, Zhengdong Lu, Hang Li, and Qun Liu. Interactive Attention for Neural Machine Translation. *COLING*, pages 2174–2185, 2016.

[5] Haitao Mi, Baskaran Sankaran, Zhiguo Wang, and Abe Ittycheriah. Coverage Embedding Models for Neural Machine Translation. *EMNLP*, pages 955–960, 2016.

[6] Jan Niehues, Eunah Cho, Thanh-Le Ha, and Alex Waibel. Pre-Translation for Neural Machine Translation. *COLING*, pages 1828–1836, 2016.

[7] Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Minimum Risk Training for Neural Machine Translation. *ACL*, pages 1683–1692, 2016.

[8] Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Modeling Coverage for Neural Machine Translation. *ACL*, pages 76–85, 2016.