出力と文脈の自己相互情報量に基づく文脈翻訳

杉山 普　　吉永 直樹
東京大学大学院 情報理工学系研究科　東京大学 生産技術研究所
{sugi, ynaga}@tkl.iis.u-tokyo.ac.jp

1 はじめに

翻訳タスクでは入力された翻訳対象の文（注目文）が同じであったとしても，その文が置かれた前後の文脈によって適切となる訳文が異なる場合がある．標準的なニューラル機械翻訳（NMT）は注目文のみから得られる情報に基づいて訳を決定する単文翻訳モデルであるため，文脈に即して適切な訳文を出力することが原理的にできない．そこで近年は注目文に付随する文脈情報を考慮する文脈翻訳モデルが研究されている．

既存の NMT ベースの文脈翻訳モデルの多くは注目文と文脈文（注目文の前後の数文）を入力し，注目文の訳文を出力する End-to-End のモデルである．現状の文脈翻訳モデル [1, 2, 3] は学習時に連続する文について対応が取れた対訳コーパスが必要であるが，そのような対訳コーパスが存在するドメインは極めて少ない．また，文脈翻訳は単文翻訳と比較して入出力の空間が大きく，より長距離の依存関係を捉えることが求められるため学習難易度が高く，学習データが十分になければ単文翻訳に対する精度面での優位性が明確には現れない [4]．結果として，多くのドメインで高精度な文脈翻訳モデルが学習できない状況にある．

そこで本研究では，単文翻訳モデルが確率モデルとして様々な文脈に対応した訳文を出力する能力を有する点に着目し，目的言語の単言語コーパスで学習した言語モデルによって翻訳モデルの出力に文脈を考慮した修正を加えることで文脈翻訳を実現する手法を提案する．本手法は既存の文脈翻訳モデルと異なり，単文翻訳モデルと文脈解釈を担う言語モデルを独立に学習するため文脈付きの対訳コーパスは必要ない．

実験では，英仏及び日英翻訳モデルを IWSLT2017 コーパスを用いて学習し，これを BookCorpus 及び OpenSubtitles2018 コーパスから学習した英語及び仏語の言語モデルとそれぞれ組み合わせて提案手法を実装し，評価した．文脈翻訳テストによる評価を通して，提案手法が標準的な End-to-end 文脈翻訳モデルと同等以上の性能を示すことを確認した．

2 事前知識

本節では，事前知識として提案手法で介入を行う単文翻訳モデルのデコードについて説明する．入力系列 $x = [x_1, ..., x_{N_x}]$ 及び出力系列 $y = [y_1, ..., y_{N_y}]$ が与えられたとき，x を翻訳することで y が得られる条件付き確率 $p(y|x)$ は以下のように Left-to-Right の生成に対応する条件付き確率の積に分解できる．

$$p(y|x) = \prod_{t=1}^{N_y} p(y_t|y_{<t}, x) \tag{1}$$

NMT ではこの条件付き確率をニューラルネットワークにより表現された関数 TM() でモデル化する．

$$p_{\mathrm{TM}}(y_t|y_{<t}, x) = \mathrm{TM}(y_t, y_{<t}, x) \tag{2}$$

デコードではモデルの翻訳スコアを最大化する系列

$$\hat{y} = \arg\max_y \prod_{t=1}^{N_y} \mathrm{TM}(y_t, y_{<t}, x) \tag{3}$$

を，ビームサーチによる探索で近似的に求める．

先に述べたように，単文翻訳モデルは文脈を考慮しないため，入力文が置かれた文脈に応じて常に正しい訳文を得ることはできない．一方で単文翻訳モデルの学習には多様な文脈中に現れる対訳文対が用いられるため，出力される $\hat{y}$ 以外の訳候補 y に対しても，y が x の適切な訳となるような文脈があれば比較的高い翻訳スコアを与えることが経験的に知られている．本研究ではこの性質に着目し，単文翻訳モデルの判断と文を越える文脈を考慮した言語モデルから得られる文脈情報を組み合わせることによる文脈翻訳を検討する．

3 提案手法

本節では，目的言語側の文脈文と出力文の自己相互情報量（PMI）に基づき文脈を考慮するデコード手法を提案する．複数の文からなる原言語の文書中の文 x に対応する訳文 y を求めることを考える．ただし，x の直前に位置する文章 $c^{(x)}$ は与えられており，さらに $c^{(x)}$ の訳文 $c^{(y)}$ は既に得られているとする．よってデコード時に目的言語側の文脈情報として $c^{(y)}$ を利用できる．以下では簡単のため $c^{(y)}$ を単に c と書く．

文脈 c を考慮した翻訳は，条件付き確率 $p(y|x, c)$ の最大化として定式化できる．

$$\begin{aligned}
\hat{y} &= \arg\max_y \log p(y|x, c) \\
&= \arg\max_y \log \frac{p(y, c|x)}{p(c|x)} \\
&= \arg\max_y \log p(y, c|x) \\
&= \arg\max_y \log p(c|y, x) p(y|x)
\end{aligned} \tag{4}$$

ここで，x と y が近い意味を持つと仮定し，$p(c|y,x) \simeq p(c|y)$ と近似する．このとき，

$$\hat{y} \simeq \arg\max_y \log p(c|y)p(y|x)$$
$$= \arg\max_y \log \frac{p(c,y)}{p(c)p(y)} p(y|x) \quad (5)$$

より，$\hat{y}$ を求めることは c と y の共起度合いを表す自己相互情報量 $\mathrm{PMI}(c,y) = \log p(c,y)/p(c)p(y)$ と，翻訳確率 $\log p(y|x)$ の和（以下合成翻訳スコア）の最大化と等価である．式 5 をもとに $\hat{y}$ を求める手法として以下の 3 方式を検討する．

3.1 リランキング

ビーム幅 N のビームサーチにより N ベストのリストを生成し，それらの中で PMI または合成翻訳スコアを最大化する文を最終出力とする（前者を言語モデルリランキング，後者を合成スコアリランキングと呼ぶ）．$p(y|x)$ は翻訳モデル，$p(y)$ 及び $p(c,y)$ は言語モデルにより計算できる．

この手法ではビームサーチで得られる N ベストリストの多様性の低さが問題となる．例えば「まだ寝ているみたいです」という日本語の文を英語に訳すとき，ゼロ代名詞の訳となり得る *he/she/they/it* などを含む文が N ベストリストで網羅されていることが望ましいが，ビームサーチでは構文の違いやカンマの有無などのバリエーションの網羅が優先され，単語のバリエーションは探索中に枝刈りで消失する可能性がある．

3.2 合成ビームサーチ

本節では，文脈翻訳で解消すべき曖昧性（例えば 3.1 節の例における *he/she/they/it*）が発生した位置で直ちに文脈との PMI を考慮したトークンの取捨選択をするビームサーチベースの手法を検討する．

式 5 は以下のように Left-to-Right の条件付き確率（トークン単位の合成翻訳スコア）の積の形に書ける．

$$\hat{y} \simeq \arg\max_y \sum_{t=1}^{N_y} \log \frac{p(y_t|c,y_{<t})p(y_t|y_{<t},x)}{p(y_t|y_{<t})} \quad (6)$$

デコード時に $p(y_t|y_{<t},x)$ を翻訳モデルで $p(y_t|c,y_{<t})$ 及び $p(y_t|y_{y<t})$ を目的言語の言語モデルでそれぞれ計算することで，文レベル NMT のデコードと同様にビームサーチが適用可能である．

具体的にはビーム幅を B，語彙を $V = \{w_1, ..., w_{|V|}\}$ としたとき，位置 t で採用する経路リストは位置 $t-1$ 時点での経路リスト $Y_{<t} = [y_{<t}^{(1)}, ..., y_{<t}^{(B)}]$ をもとに次のように決定する．位置 t における経路の候補は $Y_{<t} \times V = \{(y_{<t}, y_t) | y_{<t} \in Y_{<t} \wedge y_t \in V\}$ である．この中で以下の BeamScore を最大にする上位 B 個をもって位置 t での経路リストとする．

$$\mathrm{BeamScore}(y_{<t}, y_t)$$
$$= \sum_{t'=1}^{t} [\mathrm{PMI}(y_{t'}, c; y_{<t'}) + \log p(y_{t'}|y_{<t'})] \quad (7)$$

ただし，$\mathrm{PMI}(y_t, c; y_{<t}) = \log p(y_t|c, y_{<t})/p(y_t|y_{<t})$ は位置 t のトークン単位の PMI を表す．

合成ビームサーチでは翻訳確率 $p(y_t|y_{<t},x)$ が低いトークンに高い PMI が与えられることでトークン単位の合成翻訳スコアが高くなる可能性がある．翻訳確率が低すぎるということは式 5 の近似が前提としている x と y が意味的に近いという仮定が満たされておらず，得られた合成翻訳スコアに正当性がないことを意味する．このような場合，不適切なトークンが枝刈りされずに採用され翻訳誤りが生じる可能性がある．

3.3 遅延合成ビームサーチ

合成ビームサーチで問題となる，翻訳としての信頼性が十分でないトークンに高い PMI が与えられてしまう可能性を減らすため，翻訳としての信頼度の高いトークンにのみ PMI を計算する．具体的には合成ビームサーチで位置 t における経路スコアを与える式 7 を以下のように遅延パラメータ d を含む形に変更する．

$$\mathrm{BeamScore}_d(y_{<t}, y_t)$$
$$= \sum_{t'=1}^{t-d} \mathrm{PMI}(y_{t'}, c; y_{<t'}) + \sum_{t'=1}^{t} \log p(y_{t'}|y_{<t'}) \quad (8)$$

これにより，位置 t における探索で PMI が計算されるのは，$t-d$ 以前の位置で採用され，その後現在まで d ステップ以上に渡って枝刈りされずに経路リストに残留し続けたトークンに限定される．遅延合成ビームサーチは $d = 0$ のとき合成ビームサーチと等価になり，$d \geq$ (文長) のとき通常のビームサーチと等価になる．

4 実験

提案手法の有効性を検証するため英仏及び日英翻訳で評価を行った．評価には BLEU [5] 及び文脈翻訳テスト [2, 4] を用いた．

4.1 実験設定

コーパス 単文翻訳モデルの訓練/開発/評価データとして対訳コーパス IWSLT2017 [6] の日英，英仏部分を用いた．英語及び仏語の言語モデルの訓練/開発データとして，単言語コーパス BookCorpus [7] 及び OpenSubtitles2018 [8] をそれぞれ用いた．表 1 に各コーパスの情報を示す．翻訳の評価は IWSLT2017 のテストセット test2010 を用いた．

	文数	平均文長
IWSLT17 ja-en	225k/871/1549	ja:40.2, en:12.8
IWSLT17 en-fr	234k/890/1568	en:17.2, fr:17.9
BookCorpus	72M/10k	24.1
OpenSubs2018	104M/10k	5.85

表 1: 実験で用いたコーパスの統計値．IWSLT2017 の文数は訓練/開発/評価データに対応．単言語コーパスの文数は訓練/開発データに対応．文長は，英語及び仏語は単語数，日本語は文字数でカウント．

	英→仏		日→英	
文脈	生成	oracle	生成	oracle
リランキング				
（言語モデル）	35.33	35.39	**11.01**	**11.25**
（合成翻訳スコア）	**36.82**	**36.85**	10.68	10.87
合成ビームサーチ	35.91	36.14	10.17	10.82
遅延合成ビームサーチ	36.15	36.44	10.45	10.95
単文翻訳	35.65		10.44	

表 2: BLEU による評価．全ての手法でビーム幅は 8．遅延合成ビームサーチの遅延は $d = 4$ とした．モデル生成文脈は文脈としてモデルが 1 文前の翻訳として出力した文を用いることを表す．oracle は文脈として 1 文前の参照訳（正解文脈）を用いることを表す．

日本語文は mecab-ipadic (ver. 2.7.0) を辞書として MeCab (ver. 0.996)[1]により分かち書きした．英語文と仏語文は Moses toolkit v4.0[2]を用いて句読点の正規化とトークン化を行った．さらに SentencePiece (ver. 0.1.81)[3]により，全ての文を Unigram モデルでサブワード化した．翻訳の原言語のサブワードの語彙は対訳コーパスの原言語の訓練データから学習した．目的言語のサブワードの語彙は単言語コーパスの訓練データから学習した．

言語モデルの学習に用いる BookCorpus 及び OpenSubtitles2018 は共に IWSLT2017 とはドメインが異なっている．そこでドメイン乖離の影響を低減するため，単言語コーパスで学習した言語モデルは IWSLT2017 の目的言語側訓練/開発データで Fine-tuning した．

翻訳モデル　単文翻訳モデルとして Transformer [9] を利用した．位置情報は相対位置表現 [10] により入力した．その他のモデルの構成は原論文 [9] に倣った．訓練時のバッチサイズはバッチ内のトークン数が 8192 程度になるよう動的に決定した．

言語モデル　言語モデルとして Transformer のデコーダ（エンコーダへアテンションをするブロックは除去）を利用した．単文翻訳モデルと同様に位置情報は相対位置表現により与えた．層は 12 層，埋め込み及びアテンションの次元は 768，アテンションヘッド数は 12，全結合層の活性化関数として ReLU（次元は 2048）とした．バッチを作る際は訓練データ中の連続する 512 トークンを 1 文とし，バッチサイズは 64 とした．

[1] https://taku910.github.io/mecab/
[2] http://www.statmt.org/moses/
[3] https://github.com/google/sentencepiece

	Coref. 1	Coref. 2	Coherence
言語モデル	93.0	93.0	87.0
合成翻訳スコア	78.0	70.0	62.0
Bawden+ 2018 [2]	77.0	68.0	57.0

表 3:　英仏の文脈翻訳テストの正解率．Coref. 1，Coref. 2 及び Coherence はそれぞれ 100 問で構成される．Coref. 1 と Coref. 2 では文をまたいだ共参照関係に基づく代名詞の性・数の曖昧性解消が要求される．Coherence では目的言語側文脈を考慮して一貫性を維持するような語彙選択が要求される．参考のため [2] にて報告された最良のモデル（原言語・目的言語の文脈を考慮し End-to-end で翻訳）の正解率を記載した．

	スコア
言語モデル	94
合成翻訳スコア	82
Sugiyama+ 2019 [4] （原言語・目的言語の文脈を利用して学習）	79

表 4: 日英の文脈翻訳テストの正解率．テストは 100 問で構成され，ゼロ代名詞の解決が要求される．

4.2　実験結果

BLEU による評価　単文翻訳及び提案手法を BLEU で評価した結果を表 2 に示す．文脈翻訳の評価では，BLEU スコアは参考程度に留まるが，日英の合成ビームサーチと英仏の言語モデルリランキングを除き，単文翻訳に対して一定の性能向上が得られていることが確認できる．

文脈翻訳テストによる評価　文脈を考慮することによる翻訳の改善を BLEU で適切に評価することは難しいため，文脈レベル NMT の評価に特化した文脈翻訳テストが Bawden らにより提案されている [2]．文脈翻訳テストは複数の 2 択問題からなる．各問題は原言語注目文 x，原言語/目的言語の文脈文 $c^{(x)}, c^{(y)}$，及び 2 つの翻訳候補文 y_1, y_2 からなる．文脈を考慮しなければ y_1, y_2 は共に x の訳文として妥当であるが，$c^{(x)}, c^{(y)}$ を考慮することで正解がどちらか一方に定まる．文脈考慮型 NMT はより高い翻訳スコアを与える訳文

$$\hat{y} = \arg\max_{y \in \{y_1, y_2\}} \mathrm{Score}(y|x, c^{(x)}, c^{(y)}) \tag{9}$$

を回答し，合計の正答率により評価される．文脈翻訳テストは文脈を考慮せずに回答した場合の正答率が 50% となるように設計されている．本研究では前述の英仏文脈翻訳テスト [2] と永田ら [11] の手順を参考に我々が作成した日英文脈翻訳テスト [4][4]により提案手

[4] https://github.com/sugi-a/discomt2019

法が文に与えるスコアの妥当性を検証した．

英仏，日英文脈翻訳テストの結果を表 3，表 4 にそれぞれ示す．表中で合成翻訳スコアに対応する正解率は式 9 中の翻訳スコアとして提案手法の合成翻訳スコア $\mathrm{PMI}(c^{(y)}, y) + \log p(y|x)$ を用いることで訳文を選択した場合の正解率である．一方，言語モデルに対応する正解率は翻訳スコアとして $\mathrm{PMI}(c^{(y)}, y)$ を用いた場合，すなわち単文翻訳モデルを使用せず言語モデルのみで正解翻訳を判断した場合の正解率である．PMI を用いた場合の正解率の高さから文脈情報として PMI が有効であることが分かる．一方単文翻訳モデルと組み合わせた場合は単文翻訳モデルのバイアスにより正解率が大きく下がるが，既存の文脈翻訳モデルより高い正解率を達成している．

5 関連研究

文脈翻訳に単言語コーパスを利用する手法としては目的言語の単言語コーパスを逆翻訳して疑似対訳データを構築し学習に用いるデータ拡張が高い効果を示すことが確認されている [4]．データ拡張では文脈を捉える性能だけでなく文レベルの翻訳性能も大きく向上する．本研究の提案手法は文脈を取り入れる手段とはなるものの，基本的な翻訳の品質は単文翻訳モデルの性能に依存するため，データ拡張と組み合わせることで一層の精度向上が期待できる．

単文翻訳モデルの出力に，翻訳モデルとは独立に学習されたモジュールで文脈を考慮した修正を加える手法として，Voita ら [12] は目的言語側の単言語コーパスを逆翻訳しさらにそれを順翻訳することで，元文書と，概ね内容は同じだが文脈の乱れた文書の対を獲得し，それを用いて文脈の乱れた文書を文脈の整った文書に変換する系列変換モデルを学習する手法を提案した．この手法では文レベル翻訳器が出力した単一の系列を元に，修正用モジュールが文脈の整った真の系列を推定することになる．これに対し，本研究の提案手法では文レベル翻訳器の出力として語彙空間上の確率分布や N ベストリストといった，翻訳の曖昧性を情報として含むような量を用いる．

また，独立に学習された翻訳モデルと言語モデルが出力するトークン単位の確率値を組み合わせてデコードする関連手法として Shallow fusion [13] がある．Shallow fusion は単文翻訳において出力文 y の流暢さを補強するため，翻訳モデルの出力する翻訳スコア $\log p(y|x)$ に言語モデルが出力する生成確率 $\log p(y)$ を足したものを合成翻訳スコアとしてデコードを行う．一方本研究における合成翻訳スコアは文脈 $c^{(y)}$ で条件づけられた y の生成確率 $\log p(y|c^{(y)})$ から y 単独での生成確率 $\log p(y)$ を差し引いた PMI により y と文脈の共起度合いを合成翻訳スコアに取り入れている点で Shallow fusion とは理論的背景が異なっており，文脈文が空文であったとしても 2 つの手法は一致しない．

6 おわりに

本稿では単文翻訳モデルのデコード時に，目的言語側の文脈と訳文候補の自己相互情報量を言語モデルによって計算し翻訳スコアを修正する文脈翻訳を提案した．文脈翻訳テストの正解率は提案手法の PMI 及び合成翻訳スコアの文脈情報としての有効性を支持したものの，合成翻訳スコアの最大化を忠実に追及する合成ビームサーチや遅延合成ビームサーチが BLEU では低評価となる傾向が見られることから，文脈を捕捉する性能と一般的な翻訳品質を両立するような，より良いデコードアルゴリズム及び実際に生成された訳文に基づく文脈翻訳の性能評価手法の必要性が示唆された．合成翻訳スコアの実践的な使用法として，代名詞など特に曖昧性の発生しやすいトークンの組み合わせを事前知識として保持しておき，それらに関する曖昧性が発生した場合にのみ提案手法を適用するなどのアプローチも考えられる．

謝辞

この研究の一部は，2019 年度国立情報学研究所 CRIS 委託研究，及び JST，CREST，JPMJCR19A4 の支援を受けています．

参考文献

[1] J. Tiedemann, Y. Scherrer. Neural machine translation with extended context. In *DiscoMT*, 2017.

[2] R. Bawden *et al.* Evaluating discourse phenomena in neural machine translation. In *NAACL*, 2018.

[3] S. Maruf, G. Haffari. Document context neural machine translation with memory networks. In *ACL*, 2018.

[4] A. Sugiyama, N. Yoshinaga. Data augmentation using back-translation for context-aware neural machine translation. In *DiscoMT*, 2019.

[5] K. Papineni *et al.* Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002.

[6] M. Cettolo, C. Girardi, M. Federico. Wit3: Web inventory of transcribed and translated talks. In *EAMT*.

[7] Y. Zhu *et al.* Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *ICCV*, 2015.

[8] P. Lison, J. Tiedemann, M. Kouylekov. OpenSubtitles2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In *LREC*, 2018.

[9] A. Vaswani *et al.* Attention is all you need. In *NIPS*, 2017.

[10] P. Shaw, J. Uszkoreit, A. Vaswani. Self-attention with relative position representations. In *ACL*, 2018.

[11] 永田昌明, 森下睦. 日本語から英語への文脈翻訳テストの提案. 言語処理学会第 25 回年次大会 発表論文集, pp. 1–4, 2019.

[12] E. Voita, R. Sennrich, I. Titov. Context-aware monolingual repair for neural machine translation. In *EMNLP-IJCNLP*, 2019.

[13] C. Gulcehre *et al.* On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*, 2015.