教師なし同期的句構造を用いた機械翻訳

原田慎太朗　渡辺太郎
奈良先端科学技術大学院大学 先端科学技術研究科
{harada.shintaro.hk4, taro}@is.naist.jp

1　はじめに

近年、ニューラルネットワークを用いた機械翻訳モデルは、高い翻訳性能を達成している．しかし、例えば仏英や独英などの構文的に近い言語間と比較して、例えば日英や韓英などの構文的に遠い言語間の翻訳性能はあまり高くない [1]．これに対し、統計的機械翻訳では原言語文と目的言語文間の同期文法を考慮することで翻訳性能を改善してきた [2]．同期文法は二言語間の複雑な関係を表現し、統語論的な句構造を取り入れることで、より言語学的に正確な翻訳を可能にする [3]．

これをニューラル機械翻訳に適用することで、構文的に遠い言語間の翻訳性能の向上が期待できる．しかし、構文情報を人手で与えるのは非常にコストがかかり、多言語に適用するは難しい．この問題に対処するため、構文距離 [4] を利用して自己注意機構から教師なしに句構造を導出する方法と、導出した句構造を同期するための方法を提案する．提案手法と先行研究の翻訳性能を BLEU[5] を用いて比較したところ、IWSLT14 独英翻訳では 0.54 ポイント の向上、ASPEC 日英翻訳では 0.51 ポイント の向上を確認した．句構造解析の性能については Unlabeled F 値 で評価し、同期的句構造が構文解析に対しても有用であることを示した．また、同期的句構造が単語アライメントを正しく修正することを確認した．

2　関連研究

機械翻訳において様々なニューラルネットワークモデルが提案されているが、翻訳精度および学習効率の面から自己注意機構をベースとした Transformer[6] がよく用いられる．構文情報を機械翻訳に用いる既存研究 [7] では、人手またはツールで作成された構文情報をモデルに組み込むことで、高い翻訳性能を報告している．しかし、テキストに対して構文情報を付与したデータを作成するには非常にコストがかかるため多言語への適用が難しい．また、機械翻訳における Out-Of-Vocablary 問題への対処のために、サブワードが必要である [8]．しかし、構文解析ツールがサブワードに対応しているとは限らないため、構文情報を機械翻訳モデルに組み込むことは容易ではない．

近年、言語モデリング用いて間接的に句構造解析を学習する手法が提案されている [4, 9, 10]．これらで報告されている構文構造は、教師なしにも関わらず人間のアノテーションに近い．また、言語モデリングの応用である機械翻訳からでも翻訳性能を劣化させずに高品質な構文木が導出できることが報告されており、機械翻訳が言語モデリングよりも構文情報の導出に優れていることを示している [11]．しかし、単言語の構造しか考慮していないため、教師なしに導出した構文情報は必ずしも機械翻訳の精度向上に貢献しない．

3　教師なし句構造に対する同期制約

本稿では、教師なしに導出した句構造構造を明示的に利用する機械翻訳手法を提案する．提案内容は、(1) 構文距離 [4] を利用して自己注意機構 [6] から句構造を導出する方法、(2) 導出した原言語と目的言語の句構造情報を同期させる方法の 2 つである．

3.1　教師なし句構造導出

句構文構造の表現方法として構文距離 [4] がある．構文距離は隣接する単語間の句切れの程度を示す実数値であり、先行研究では RNN 言語モデルの注意機構を用いて構文距離を導出している．本稿では、このアイデアを機械翻訳において高い性能を達成する Transformer の自己注意機構 [6] に応用することで教師なしに句構造を導出する．Transformer は複数の自己注意機構層から構成されており、自己注意の重みは以下の式で表される．

$$a_i^t = \mathrm{softmax}(\frac{q_i k_t^\top}{\sqrt{\delta_{head}}}) \quad (1)$$

ここで、$q = hW_q, k = hW_k$ であり、h はある隠れ層のベクトル、W_q と W_k はパラメータである．また、$\delta_{head} = \delta_h / N_{head}$ であり、δ_h は隠れ層のベクトルの次元数、N_{head} は隠れ層のベクトルの分割数である．t は現在注目している単語位置、i はその他の単語位置を示す．i 番目の単語の構文距離は以下の式で表される．

$$d_i = \mathrm{ReLU}(W \begin{bmatrix} k_{i-N} \\ k_{i-N+1} \\ \cdots \\ k_i \end{bmatrix} + b) \tag{2}$$

ここで、W, b はパラメータ、k は自己注意機構内の潜在表現、N は構文距離を計算する際に参照する過去の単語数を示す．次に句構造をモデル化するために構文距離を以下の式で確率値に変換する．

$$\alpha_j^t = \frac{\mathrm{hardtanh}((d_t - d_j) \cdot \tau) + 1}{2} \tag{3}$$

ここで、$\mathrm{hardtanh}(x) = \max(-1, \min(1, x))$、$\tau$ は構文距離の差に対する感度を制御する温度パラメータである．句構造のモデル化は以下の式で表現される．

$$g_i^t = P(l_t \leq i) = \prod_{j=i+1}^{t-1} \alpha_j^t \tag{4}$$

ここで、l_t は句構造の切れ目の位置を示す変数である．句構造モデルを自己注意機構に組み込むことで、同階層内だけで自己注意が係るよう、以下の式で制約をかける．

$$\tilde{a}_i^t = \frac{g_i^t \cdot a_i^t}{\sum_i g_i^t \cdot a_i^t} \tag{5}$$

しかし、句構造を自己注意機構に組み込むだけでは局所的な制約が強すぎるため、大局的な情報を参照できないという問題がある、そこで、下層で参照した情報を上層でも参照できるよう、以下の式で上層に上がるにつれて局所的制約を緩和させる．

$$\hat{g}_i^{t,(l)} = g_i^{t,(l-1)} + (1 - g_i^{t,(l-1)}) \cdot g_i^{t,(l)} \tag{6}$$

3.2 句構造に対する同期制約

原言語側と目的言語側の自己注意の整合性を保つための制約として同期注意制約 [12] がある．先行研究では、機械翻訳に同期注意制約を課すことで翻訳性能を向上させている．この手法は同期文法から着想を得ており、同期文法は構文的に遠い言語間の統計的機械翻訳において有用であることが知られている [2]．本研究では句構造を表現する構文距離を同期させる事により、構文的に遠い言語間でのニューラル機械翻訳の向上が期待できると考えた．この同期制約は原言語側と目的言語側の構文距離を最小二乗誤差 (Mean Squared Error; MSE) を用いて実現でき、以下の式で表される．

$$\mathcal{L}_{sync} = \frac{1}{L} \sum_{l}^{L} \sum_{i} \left(d_i^{(l)} - \tilde{d}_i^{(l)} \right)^2 \tag{7}$$

ここで、$d^{(l)}$ は目的言語側 l 層目の構文距離であり、$\tilde{d}^{(l)}$ は原言語側 l 層目の構文距離 $s^{(l)}$ を目的言語側 l 層目の構文距離 $d^{(l)}$ へと写像したものである．L は提案手法である句構造導出層の数である．

構文構造における重要な要素は、構文距離から導出される階層的位置関係である．しかし、MSE では正確な距離当て問題に帰着してしまうため、階層的位置関係が無視され、モデルに過剰なペナルティを与えてしまう．そこで、階層的位置関係を考慮する Rank 誤差 [13] を利用する．構文距離に適用すると以下の式で表される．

$$\mathcal{L}_{sync} = \frac{1}{L} \sum_{l}^{L} \sum_{i,j>i} \mathrm{ReLU} \left(1 - \mathrm{sign}(d_i^{(l)} - d_j^{(l)})(\tilde{d}_i^{(l)} - \tilde{d}_j^{(l)}) \right) \tag{8}$$

ここで、$\mathrm{sing}(x)$ は符号関数である．写像した目的言語側 l 層目の構文距離 $\tilde{d}^{(l)}$ は以下の式より求める．

$$\tilde{d}^{(l)} = C^{(l)} s^{(l)} \tag{9}$$

ただし、$C \in \mathbb{R}^{m \times n}$ は原言語と目的言語の隠れ表現の関連度を表す言語間注意であり、n は原言語側の入力長、m は目的言語側の入力長である．言語間注意 C は、以下の式で計算される．

$$C^{(l)} = \mathrm{softmax}(\frac{Q^{(l)} K^{(l)\top}}{\sqrt{\delta_{head}}}) \tag{10}$$

ここで、$Q = (q_1, q_2, ..., q_m) \in \mathbb{R}^{m \times \delta_{head}}$ と $K = (k_1, k_2, ..., k_n) \in \mathbb{R}^{n \times \delta_{head}}$ はそれぞれ、原言語側と目的言語側の隠れ表現である．以上より、全体的な目的関数は次の通りである．

$$\mathcal{L} = \mathcal{L}_{trans} + \lambda \mathcal{L}_{sync} \tag{11}$$

ここで、$\mathcal{L}_{trans}$ は機械翻訳における目的関数である．また、λ は $\mathcal{L}_{sync}$ を考慮する度合いを制御するハイパーパラメータである．

4 実験

4.1 モデル

本稿で用いる全てのモデルは Seq2Seq フレームワークである Fairseq[14] を用いて実装した．ベース

ラインには Fairseq で提供されている Transformer を用いた．機械翻訳における同期的句構造の有効性を確認するため、先行研究である同期注意制約モデル [12] とも比較する．

4.2 データセット

翻訳性能の評価実験は，IWSLT14 独英翻訳タスク，ASPEC 日英翻訳を用いた．IWSLT14 独英翻訳の実験データは、英語、独語ともに Mosesdecoder を用いて単語分割した後、結合した訓練データから学習した BPE (Byte Pair Encoding)[15] によりサブワード単位に分割した．BPE の語彙数は 37,000 とした．ASPEC 日英翻訳の実験データは、WAT のベースラインシステムの構築方法[1] を参考に，英語には Mosesdecoder、日本語には KyTea を用いた．訓練データは train-1.txt と train-2.txt から上位 150 万を抽出して用いた．単語分割した後、結合した訓練データから学習した BPE によりサブワード単位に分割した．BPE の語彙数は 16,000 とした．

構文解析の評価実験は、IWSLT14 独英翻訳を用いた．BPE によるサブワード単位への分割は行わない．ベースラインとなる構文木は Stanford Parser[2] を用いて作成した．

4.3 実装詳細

モデルのハイパーパラメータにおいて、IWSLT14 は Fairseq の transformer_iwslt_de_en、ASPEC は同期注意制約 [12] を参考にした．また、同期的句構造の導出には、$N = 5$、$\tau = 0.1$、$\lambda = 0.05$ を用いた．翻訳文の生成にはビーム探索を用い、ビーム幅は 4、文長正則化パラメータは 0.6 とした．句構造の導出にはデコーダ 2 層までの平均構文距離を用いた．しかし、機械翻訳モデルごとに生成される翻訳文は異なるため、正解文をデコーダ側に直接入力することで、他のモデルと比較可能な構文木を獲得する．単語アライメントの導出には、FastAlign[16] と F 値を比較して最も良い 4 層目の言語間注意を用いた．

4.4 機械翻訳の結果

表 1 に実験結果を示す．モデルの翻訳性能は BLEU を用いて評価した．A.sync は同期注意制約 [12]、P. は句構造、P.sync は同期的句構造を用いたモデルを示す．最も高い BLEU スコアは太字で示す．表 1 より、提案手法の同期的句構造を加味して Transformer を訓練することにより，IWSLT14 の独英翻訳実験では 0.54 ポイント、ASPEC の日英翻訳では 0.51 ポイント の上昇が確認できた．また、同期注意制約のモデルと比較しても、全ての翻訳実験で BLEU が向上しており、機械翻訳における同期的句構造の有効性が確認できる．特に Rank 損失による句構造の同期制約を用いたモデルが一番高い BLEU を達成しており、同期的句構造の学習にも階層的位置関係が重要であることが確認できる．しかし、日英翻訳において MSE 損失を用いた場合、BLEU がベースラインよりも低下することが確認できた．理由としては、日本語と英語の句構造が大きく異なり、句構造を全く同じ距離で同期することが難しいためと考えられる．

	IWSLT14 De→En	ASPEC Ja→En
	BLEU	BLEU
Transformer	34.52	29.69
w/ A.sync	34.64 (+0.12)	29.78 (+0.09)
w/ P.	34.79 (+0.27)	29.92 (+0.21)
w/ P.sync MSE	34.93 (+0.41)	29.65 (-0.04)
w/ P.sync Rank	**35.06 (+0.54)**	**30.20 (+0.51)**

表 1 機械翻訳性能の比較

	IWSLT14 De→En	
	UF1	BLEU
PRPN (Htut et al., 2019)	56.1	30.2
Transformer	–	30.86
w/ P.	14.60	31.25
w/ P.sync MSE	14.60	31.12
w/ P.sync Rank	20.39	31.24

表 2 構文解析性能の比較

4.5 句構造解析の結果

表 2 に実験結果を示す．句構造解析性能は評価スクリプトである Evalb[3] を用いて評価した．表 2 より、提案手法の結果が先行研究で報告されている結果を大きく下回った．これは多層モデルである Transformer と構文距離の相性によるものであり、句構造導出層を下層に限定する必要性を示唆している．また、構文解析性能は向上しているにも関わらず BLEU は向上していないことから、構文情報は必ずしも機械翻訳性能に貢献しないことが分かる．

1） https://lotus.kuee.kyoto-u.ac.jp/WAT/WAT2019/baseline/dataPreparationJE.html
2） http://nlp.stanford.edu/software/stanford-corenlp-full-2018-02-27.zip
3） https://nlp.cs.nyu.edu/evalb/

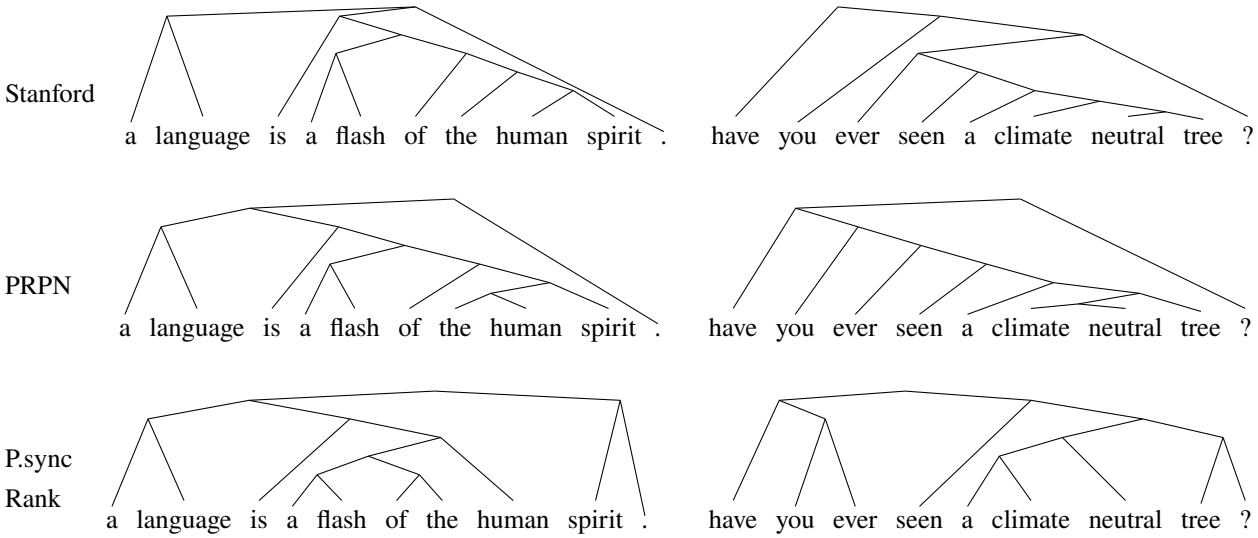

図 1　句構造構文木の比較

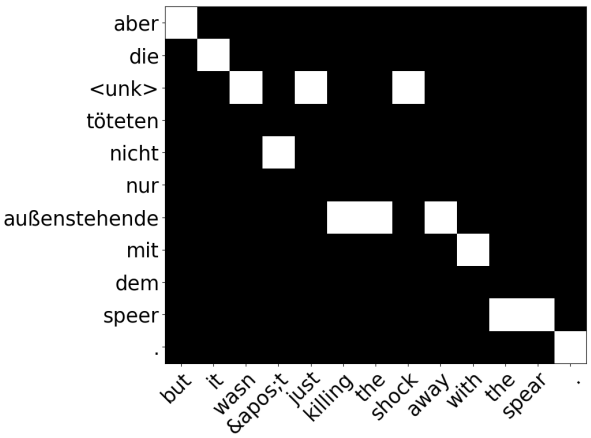

図 2　Transformer による単語アライメント

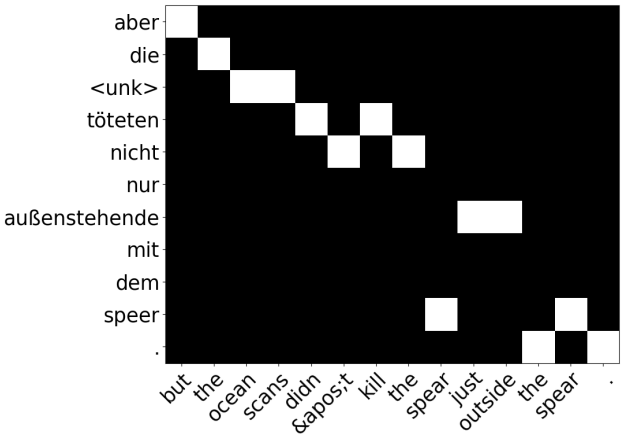

図 3　同期的句構造による単語アライメント

図 1 に Stanford Parser、先行研究および提案手法で得られた句構造の一部を示す．図 1 から分かるように、Stanford Parser と先行研究の構文木はある程度似ているが、提案手法で得られた構文木とは大きく異なる．しかし、どの構文木もある程度の名詞句を一塊りとして捉えているという特徴が読み取れる．これは、同期的句構造も意味をある程度塊として捉えることを示している．

4.6　単語アライメントの定性評価

それぞれ図 2 と 3 に、Transformer と同期的句構造を用いた Transformer の単語アライメントを示す．ただし、先行研究 [17] のように双方向モデルでアライメントを導出していないことに注意されたい．図 2 と 3 を比較すると、Transformer は”töte” を正しく”kill” にアライメントが取れているが、提案手法では正しくアライメントが取れている．また、提案手法では、”außenstehende” が正しく”outside” と訳され、その周辺にアライメントが取れている．これは、同期的句構造が単語アライメントに対しても有効であることを示している．

5　おわりに

本稿では、教師なしに句構造を導出する方法、および導出した句構造文法を同期することで機械翻訳性能を向上する手法を提案した．評価実験の結果、提案手法は機械翻訳を通じて適切な句構造情報を導出して同期させることで機械翻訳の性能および説明性を向上できることを示した．今後は、単語アライメントを定量的に評価することで注意機構と構文情報の関係性をより詳細に分析し、機械翻訳の性能および説明性のさらなる向上を目指す．また、他の言語間における提案手法の有効性を調査する．

参考文献

[1]Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, Vol. 5, pp. 339–351, 2017.

[2]David Chiang. Hierarchical phrase-based translation. *Computational Linguistics*, Vol. 33, No. 2, pp. 201–228, 2007.

[3]Fai Wong, Dong-Cheng Hu, Yu-Hang Mao, Ming-Chui Dong, and Yi-Ping Li. Machine translation based on constraint-based synchronous grammar. In *Second International Joint Conference on Natural Language Processing: Full Papers*, 2005.

[4]Yikang Shen, Zhouhan Lin, Chin-Wei Huang, and Aaron Courville. Neural language modeling by jointly learning syntax and lexicon. *In Proceedings of the International Conference on Learning Representations*, 2017.

[5]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, Philadelphia, Pennsylvania, USA, 2002. Association for Computational Linguistics.

[6]Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[7]Hiroyuki Deguchi, Akihiro Tamura, and Takashi Ninomiya. Dependency-based self-attention for transformer NMT. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pp. 239–246, Varna, Bulgaria, 2019. INCOMA Ltd.

[8]Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, Berlin, Germany, 2016. Association for Computational Linguistics.

[9]Yikang Shen, Shawn Tan, Alessandro Sordoni, and Aaron Courville. Ordered neurons: Integrating tree structures into recurrent neural networks. In International Conference on Learning Representations, 2019.

[10]Yau-Shian Wang, Hung-Yi Lee, and Yun-Nung Chen. Tree transformer: Integrating tree structures into self-attention. *arXiv preprint arXiv:1909.06639*, 2019.

[11]Phu Mon Htut, Kyunghyun Cho, and Samuel R Bowman. Inducing constituency trees through neural machine translation. arXiv preprint arXiv:1909.10056, 2019.

[12]二宮 崇出口 祥之. 同期注意制約を与えた transformer によるニューラル機械翻訳. 言語処理学会 第 26 回年次大会 発表論文集, pp. 1459–1462, 2020.

[13]Yikang Shen, Zhouhan Lin, Athul Paul Jacob, Alessandro Sordoni, Aaron Courville, and Yoshua Bengio. Straight to the tree: Constituency parsing with neural syntactic distance. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1171–1180, Melbourne, Australia, 2018. Association for Computational Linguistics.

[14]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 the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pp. 48–53, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.

[15]Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, Berlin, Germany, 2016. Association for Computational Linguistics.

[16]Chris Dyer, Victor Chahuneau, and Noah A. Smith. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 644–648, Atlanta, Georgia, 2013. Association for Computational Linguistics.

[17]Sarthak Garg, Stephan Peitz, Udhyakumar Nallasamy, and Matthias Paulik. Jointly learning to align and translate with transformer models. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Hong Kong, 2019.