Transformer に基づく英日翻訳器からの単語アラインメント抽出手法の比較

古澤智博　　松崎拓也
東京理科大学　理学部第一部　応用数学科
1417092@ed.tus.ac.jp　　matuzaki@rs.tus.ac.jp

1　はじめに

単語アラインメントとは、対訳文ペアにおいて、意味の上で対応している単語を繋いだものである。単語アラインメントの一例を図 1 に示す。エンコーダ・デコーダモデルに基づくニューラル機械翻訳では、モデルの内部に単語アラインメントを明示的に持たない方式が一般的である。近年、ニューラル機械翻訳と同時に単語アラインメントを付与する手法が、盛んに研究されている [1, 2]。この理由として、翻訳モデルのブラックボックス化が起き、翻訳過程の可視化が望まれている点 [3]、また応用面では、強調表示などの原文に対するテキスト修飾を訳文の対応する部分に付加する目的 [4] などがあげられる。特に、翻訳モデルとして現在広く使われている Transformer モデルについて、単語アラインメント抽出手法の提案が複数なされている [2, 4, 5, 6]。

本研究では、これまで英仏翻訳、英独翻訳などに対して評価結果が報告されている Transformer に基づく単語アラインメント抽出手法を、英日翻訳に適用し、得られる結果を分析する。実験の結果から、

- Saliency に基づく Transformer からの単語アラインメント抽出 [6] では記号類に誤りが多い
- 相対的に、GIZA++ では前置詞 → 助詞ペアに誤りが多い

という観察が得られた。

2　Transformer に基づく単語アラインメント抽出手法

Transformer [7] は、ニューラルネットワークによるエンコーダ・デコーダモデルの一種である。エンコーダの最初の処理として、単語 ID が埋め込みベクトルに変換される。その後、複数の自己注意機構を経たのち、出力ボキャブラリの全単語についての確率分布が出力される。以上の過程は、全体として微分可能である。

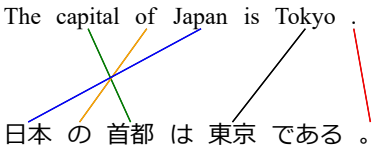

図 1　単語アラインメントの一例

Require: I = 入力長, J = 出力長, $[M_{ij}]$ = Saliency の行列
Ensure: $[X_{ij}]$ = 単語アラインメントの行列
  $X_{ij} \leftarrow$ **false**　$(i = 1,\ldots,I,\quad j = 1,\ldots,J)$
  for $i = 1,\ldots,I$ **do**
    $t \leftarrow \max(M_{i1},\ldots,M_{iJ})$
    for $j = 1\ldots J$ **do**
      if $M_{ij} > t\alpha$ **then**
        $X_{ij} \leftarrow$ **true**
  for $j = 1,\ldots,J$ **do**
    $t \leftarrow \max(M_{1j},\ldots M_{Ij})$
    for $i = 1,\ldots,I$ **do**
      if $M_{ij} < t\beta$ **then**
        $X_{ij} \leftarrow$ **false**

図 2　単語アラインメントを取得するアルゴリズム

Ding ら [6] は、Transformer の内部機構が微分可能である点を利用し、入力の各単語が出力の各単語に与える影響を微分係数の大きさとして数値化し、単語アラインメントを取得する手法を提案した。具体的には、まず入力の単語埋め込みの各成分について、デコーダが出力する各単語の確率の勾配の大きさを計算する。次に、ある出力単語の選択における各入力単語の寄与（Saliency）を表す量を、先ほど得た成分についての勾配をもとに計算する。以上により、（入力単語数）×（出力単語数）の Saliency の行列を得る。Ding らは、Saliency を表す尺度として、勾配ベクトルのノルムを用いているが、この尺度の選択には検討の余地があるため、後述する実験によって英日翻訳に適した尺度を比較検討した。また、Ding らの実装に含まれる SmoothGrad [8] も加えて実装した。SmoothGrad は、入力を多数複製してノイズを加えたものそれぞれについて勾配を調べる

表 1　Saliency の尺度を変えたときの精度（F_1）

	SmoothGrad 有り	SmoothGrad 無し
成分の絶対値の平均	0.549	0.539
成分のフロベニウスノルム	0.511	0.536
成分の絶対値の最大値	0.365	0.497
成分の平均の絶対値	0.283	0.297

表 2　言語対別の AER の違い

	GIZA++	Saliency
英日	0.435	0.449
英独 [6]	0.231	0.430
英仏 [6]	0.098	0.259
英羅 [6]	0.322	0.414

表 3　品詞別の precision/recall/F_1 の違い（KFTT）

英品詞 → 日品詞	GIZA++			Saliency		
	Prec	Rec	F_1	Prec	Rec	F_1
固有名詞 → 名詞	0.772	0.791	0.781	0.745	0.818	0.779
名詞 → 名詞	0.779	0.672	0.721	0.783	0.779	0.781
句読点 → 補助記号	0.790	0.651	0.714	0.476	0.245	0.324
前置詞 → 助詞	0.482	0.106	0.174	0.632	0.444	0.522
動詞 → 動詞	0.688	0.453	0.546	0.739	0.550	0.631
動詞 → 名詞	0.669	0.605	0.636	0.706	0.688	0.697
動詞 → 助詞	0.177	0.278	0.217	0.145	0.414	0.215

表 4　α と β を変化させたときの精度（F_1）

		出力側 (β)			
		0.35	0.40	0.45	0.50
入力側 (α)	0.80	0.524	0.532	0.533	0.535
	0.85	0.539	0.543	0.545	0.542
	0.90	0.546	0.551	0.549	0.547
	0.95	0.550	0.550	0.549	0.543

ことで、精度を向上させる手法である。

Saliency の行列から単語アラインメントを取得する手法として、Ding らは各出力単語に対して Saliency が最大の入力単語を選んだ後に grow-diagonal heuristic [9] により補正する方法を用いている。一方、本論文では図 2 に示す方法を用いた。アルゴリズム中のパラメータ α および β は、後述する実験によって最適値を探索した。

図 2 に示すアルゴリズムの要約は、次の通りである。まず、入力側の各単語から見て、相対的に高い Saliency を持つすべての出力単語との間にエッジを張る。次に、出力側の各単語から見て、相対的に低い Saliency を持つすべての入力単語との間のエッジを（あれば）削除する。最後に残ったエッジの集まりを単語アラインメントとする。

3　実験に使用する英日データ

訓練・評価データとする英日の対訳コーパスとして、京都フリー翻訳タスクデータ（KFTT）[10] と Asian Scientific Paper Excerpt Corpus（ASPEC）[11] を使用した。KFTT は Wikipedia の京都に関係する記事から作成された英日対訳コーパスである。ASPEC は学術論文の要旨から作成された英日対訳コーパスである。KFTT は約 44 万文、ASPEC は約 300 万文の対訳文から成る。KFTT の評価データ（1235 文）には、人手で単語アラインメントが付与されている。

4　実験設定

Transformer の実装として、fairseq [12] に含まれるものを用い、これを改造することで実験を行った。KFTT の正解の単語アラインメントと比較する際は、Transformer がコーパス内の目的言語文そのものを出力したとみなし、コーパス内の対訳文に対して単語アラインメントを付与した。ASPEC を用いた実験では、正解の単語アラインメントデータが存在しないため、入力文に対してモデルが出力した目的言語文との間の単語アラインメントを抽出し、分析した。また、Transformer の訓練と単語アラインメントの抽出は、KFTT と ASPEC それぞれで独立に行った。

トークナイズには、SentencePiece [13] を用いた。原言語・目的言語で共通の語彙集合を用い、語彙サイズは、16000 とした。語彙集合の生成は、KFTT と ASPEC それぞれで独立に行った。

SmoothGrad の分散とサンプル数は、評価データ上でチューニングを行い、それぞれ 1.0 と 50 とした。

KFTT について、品詞別の単語アラインメントの精度を測定する際、品詞の自動判定に英文には

表 5　誤って付いた単語アラインメントの品詞別分類

		出力側 和文							
		単語一部	形容詞	名詞	コピュラ	動詞	機能語	記号類	数詞
入力側 英文	単語一部	1	1	5	0	0	7	10	0
	形容詞	1	0	5	0	2	0	6	0
	名詞	11	0	13	1	1	6	11	0
	コピュラ	1	0	0	0	0	5	15	0
	動詞	1	0	6	0	0	3	11	1
	機能語	3	1	16	1	1	7	22	7
	記号類	1	1	7	0	0	4	15	1
	数詞	0	0	1	0	0	0	0	0

表 6　付くべきところが付いていない単語アラインメントの原語（英文）品詞別分類

単語一部	形容詞	名詞	コピュラ	動詞	機能語	記号類	数詞
10	6	40	0	9	13	101	2

spaCy[1)]を、和文には KyTea[2)]を用いた。Transformer から抽出した単語アラインメントは、SentencePiece によるサブトークンの間のアラインメントとして得られる。これを KFTT の単語単位の正解アラインメントと比較する際、KFTT の各単語対と一文字ずつでも重なるサブトークン対にアラインメントが付いていれば、その単語対にはアラインメントが付いているとみなした。

また、比較対象として、代表的な統計的単語アラインメント抽出手法である GIZA++ [14] を用いた。GIZA++ による単語アラインメントの推定には、KFTT に含まれる対訳文約 44 万文のみを用いた。

5　実験項目

5.1　α と β に関する実験

KFTT に対して、図 2 の手続きのパラメータ α および β の値を変化させて、Saliency による単語アラインメントを出力した。ここでは、Saliency の尺度として、成分の絶対値の平均を用いた。

表 4 に、そのときの単語アラインメント精度（F_1 スコア）を示す。結果から $\alpha = 0.9$、$\beta = 0.4$ の時が最も精度が高いことが分かる。以降の実験ではこの値を用いた。

5.2　Saliency を表す尺度に関する実験

Saliency を表す尺度については、Transformer から自動微分により各入力単語に対し得られる 512 次元の勾配ベクトルを、SmoothGrad の複製により 512 × 50 の勾配行列にしたものに対して、成分の絶対値の平均、成分のフロベニウスノルム、成分の絶対値の最大値、成分の平均の絶対値の 4 つを比較した。成分の平均の絶対値は、Ding らの実験で使用された基準と実質的に同等である。

表 1 に、Saliency の尺度を変えたときの KFTT での単語アラインメント精度の比較を示す。SmoothGrad の有無にかかわらず、成分の絶対値の平均が最も精度が高いことが分かる。また、SmoothGrad を用いた場合、成分の絶対値の平均以外については精度が低下していることが分かる。以降の実験では成分の絶対値の平均を Saliency の尺度として用い、SmoothGrad を併用した。

5.3　単語アラインメント精度

GIZA++ 及び Saliency による単語アラインメントを、KFTT の人手による正解アラインメントと比較し、品詞別に precision/recall/F_1 を計算したものを表 3 に示す。ただし、得られた単語アラインメントが多く付いた上位の品詞対についてのみ示す。また、表中の「句読点」は、括弧などを含む広義の句読点である。表より、GIZA++ は前置詞 → 助詞の精度が相対的に低く、Saliency に基づく手法では句読点 → 補助記号の精度が相対的に低いことがわかる。

また、Ding らの実験結果との比較を、表 2 に示す。表中の数値は Alignment Error Rate (AER) であり、値が小さいほど精度が高いことを表す。Ding らの実験結果と同様に、GIZA++ の精度が Saliency に基づく手法を上回っていることが分かる。

5.4　ASPEC を用いた実験

ASPEC に対して、Saliency による単語アラインメントを出力し、英文の文字数が 80 から 90 のもの 69 文、および 270 から 290 のもの 62 文の計 131 文に対して単語アラインメントの誤りの分析を行った。

1）https://spacy.io
2）http://www.phontron.com/kytea/index-ja.html

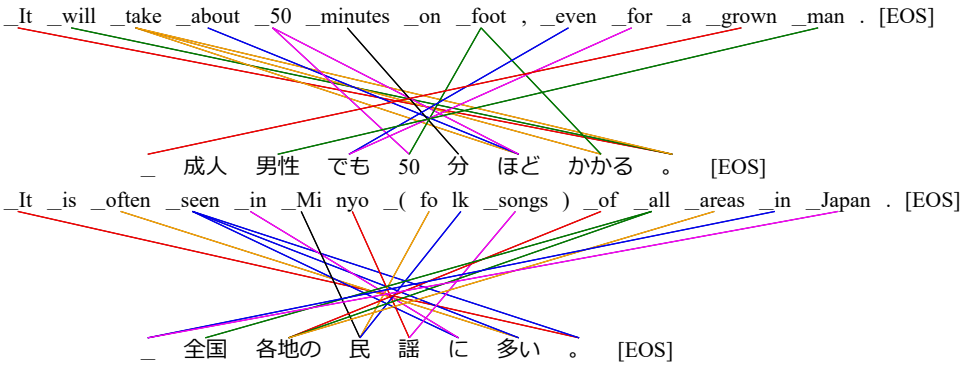

図 3　Saliency に基づく手法からの単語アラインメントの例

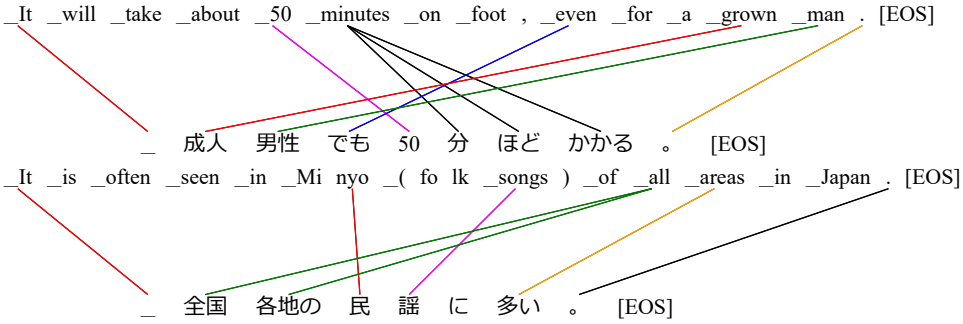

図 4　GIZA++ による単語アラインメントの例

表 5 に誤った単語アラインメントの数を、表 6 に本来付くべきところが付いていない単語アラインメントの数を、品詞別に集計したものを示す。表中の「単語一部」とは、SentencePiece によって形態素より細かく分割されたサブワードを意味する。KFTT を用いた実験結果と同様、Saliency に基づく手法は記号類に弱いことが分かる。また、出力側の和文の記号類に対して誤ったアラインメントが付きやすいことが分かる。

6　考察

Saliency に基づく単語アラインメント抽出手法について、Ding らの英独、英仏、英羅翻訳での実験では、SmoothGrad の使用によって AER を半分近く減少させていた。しかし、本研究では、SmoothGrad による効果は大きくなかった。

また、品詞別の評価において、Saliency による単語アラインメントでは、記号類に多く誤りが発生することが確認された。特に、文末のピリオドと句点の対にアラインメントが付かない例が多く見られた。その例を、図 3 に示す。これは、出力側の文末の句点に対する確率は、Transformer において入力側の文末のピリオドに強く依存しているわけではないことを示している。一方、GIZA++ の出力では、前置詞 → 助詞のアラインメントが付かない誤りが相対的に多く確認された。その例を、図 4 に示す。図中の上の例では、「about」→「ほど」にアラインメントが付いていないことが分かる。これは、英語の前置詞と日本語の助詞の間には、単純な一対一ではない意味的な対応があり、GIZA++ の統計モデルではこの関係を適切に学習できなかったためだと考えられる。同一のデータで訓練した Transformer からはより正確な前置詞 → 助詞のアラインメントが得られていることは、Transformer の能力の一端を示していると言える。

7　おわりに

本研究では、これまで英仏翻訳、英独翻訳などに対して評価結果が報告されている Transformer に基づく単語アラインメント抽出手法を英日翻訳に適用し、得られる結果を分析した。品詞別の単語アラインメント精度を比較することで、全体としての精度の比較では分からない手法別の結果の違いが確認できた。今後の課題として、他の単語アラインメント抽出手法も含めた精度の比較および日英翻訳の場合の評価が挙げられる。

参考文献

[1] Xintong Li, Guanlin Li, Lemao Liu, Max Meng, and Shuming Shi. On the word alignment from neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1293–1303, 2019.

[2] Thomas Zenkel, Joern Wuebker, and John DeNero. End-to-end neural word alignment outperforms GIZA++. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1605–1617, 2020.

[3] Yanzhuo Ding, Yang Liu, Huanbo Luan, and Maosong Sun. Visualizing and understanding neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1150–1159, 2017.

[4] Thomas Zenkel, Joern Wuebker, and John DeNero. Adding interpretable attention to neural translation models improves word alignment. *CoRR*, Vol. abs/1901.11359, , 2019.

[5] Yun Chen, Yang Liu, Guanhua Chen, Xin Jiang, and Qun Liu. Accurate word alignment induction from neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 566–576, 2020.

[6] Shuoyang Ding, Hainan Xu, and Philipp Koehn. Saliency-driven word alignment interpretation for neural machine translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pp. 1–12, 2019.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, Vol. 30, pp. 5998–6008, 2017.

[8] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *CoRR*, Vol. abs/1706.03825, , 2017.

[9] Philipp Koehn, A. Axelrod, Alexandra Birch, Chris Callison-Burch, M. Osborne, and David Talbot. Edinburgh system description for the 2005 iwslt speech translation evaluation. In *IWSLT*, 2005.

[10] Graham Neubig. The Kyoto free translation task. http://www.phontron.com/kftt, 2011.

[11] Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. ASPEC: Asian scientific paper excerpt corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2016)*, pp. 2204–2208, 2016.

[12] 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.

[13] Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 66–75, 2018.

[14] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, Vol. 29, No. 1, pp. 19–51, 2003.