構文ラベル予測による同時ニューラル機械翻訳

加納保昌 [1]　須藤克仁 [1,2]　中村哲 [1,2]
[1] 奈良先端科学技術大学院
[2] 理化学研究所 革新知能統合研究センター AIP
{kano.yasumasa.kw4,sudoh,s-nakamura}@is.naist.jp

概要

同時翻訳は、話し手が話し終わる前に翻訳を開始するタスクであり、いつ翻訳を開始するかが重要である。しかし、英語と日本語のように語順が異なる言語ペアのニューラル機械翻訳では、そのタイミング判断は未だ難しい。そこで、構文ラベル予測モデルと、そのラベルを利用した簡単な判断ルールを提案する。これらにより、既存手法では難しかった翻訳の構文や出力タイミングの制御もしやすくなった。英日同時通訳の実験で、提案手法は精度と遅延のトレードオフにおいて、ベースラインを上回った。

1　はじめに

原言語文	I **bought a pen**.
順送り訳	私は 買った **ペンを**.
フルセンテンス翻訳	私は **ペンを** 買った.

表 1　英語 (SVO) から日本語 (SOV) への翻訳

同時翻訳は、話し手が話し終わる前に翻訳を始めるタスクである。文の後半を見ずに翻訳を始めなければいけないので、文全体を見て行う通常の翻訳 (フルセンテンス翻訳) より難しい。同時翻訳では、いつ翻訳を始めるのかというタイミングを決めることが重要である。そのタイミングの遅延と翻訳精度にはトレードオフがあり、できるだけ遅延を抑える必要がある。

よく使われる同時機械翻訳モデルの一つに wait-k [1] がある。これは、原言語文の k トークンが読み込まれるのを待ったのちに、翻訳を開始する。しかし、このような単純な手法では日英などの、長距離の単語の並べ替えが必要となる言語対の同時翻訳は難しい。

表 1 に示されている通り、単純に前から訳すだけの順送り翻訳は不自然になりがちである。“bought a pen” という部分は、“ペンを買った” というように訳される方が自然である。ここでは、“bought” の後の目的語を待つ必要がある。この例文では、“I” が並べ替えを必要としない最後の単語であり、ここが自然に翻訳をすることができるセグメントの境である。

このような並べ替えの問題を解決するため、部分入力文の次にくる構文ラベルの予測とそれに基づくシンプルなルールによってセグメンテーションを行い、チャンクベースで翻訳をする手法を提案する。

これによって、既存手法では困難だった翻訳出力の構文とタイミングの制御をしやすくした。そして、英日同時翻訳の実験の結果、ベースラインを精度と遅延のトレードオフにおいて上回った。

2　関連研究

ニューラル同時機械翻訳においては、wait-k よりも翻訳タイミングを柔軟に決められるように、遅延をロス関数に組み込む手法が提案されてきた [2, 3, 4] 。しかし、学習済みモデルを用いて最適なセグメント境界を BERT [5] による予測モデルで見つけ、チャンクごとに翻訳していく Meaningful Unit (MU)[6] という手法も提案され、これらを上回った。

統計的機械翻訳においては、構文情報を用いる手法として、小田ら [7] が、各セグメントの右辺と左辺の構文要素ラベルを予測することで、不完全な文を漸進的に構文解析する手法を提案した。彼らは、予測された構成要素とセグメント内の単語を連結し、その結果を tree2string の機械翻訳に入力した。そして、翻訳結果のどこに構成要素が現れるかによって、さらに入力トークンを待つか、翻訳を出力するかを決めた。本研究はこの手法をニューラル機械翻訳に展開し、よりシンプルにしたものである。

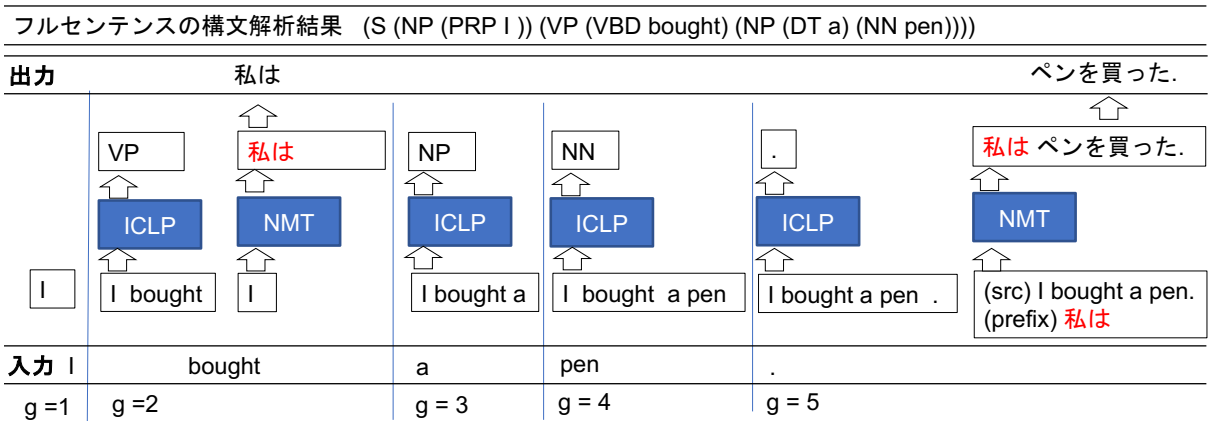

図 1 ICLP から予測されたラベルとルールに基づいて境界が検出されると、NMT（翻訳モデル）は翻訳を開始する。前の翻訳（図では赤色）は次の翻訳のプレフィックスとして使われる。図では簡単のため EOS は省略されている。

セグメンテーション	You	/ can	save	time	by	/ doing	this	.
構文ラベル		VP	VP	NP	PP	S	NP	.
構文木	(S (NP (PRP You)) (VP (MD can) (VP (VB save) (NP (NN time)) (PP (IN by) (S (VP (VBG doing) (NP (DT this))))))) (. .))							

表 2 最小セグメント長が 1 の ICLP の結果例。

3 提案手法

図 1 は提案手法の全体図である。まず、文のプレフィックスから次の構文要素ラベルを予測する。そして、そのラベルを用いたルールによってそこがセグメント境界だと判断された場合には、そのチャンクを翻訳する。

3.1 チャンクベースの同時翻訳

標準的なニューラル機械翻訳は次の式で表される。

$$p_{full}(Y|X) = \prod_{t=1}^{|Y|} P(y_t|X, y_{<t}), \quad (1)$$

$X = x_1, x_2, ..., x_n$ は n トークンからなる入力文で、$Y = y_1, y_2, ..., y_m$ は m トークンからなる予測された翻訳文である。

チャンクベースの同時翻訳でも、上記のモデルを通常の対訳文ペアで学習する。しかし、推論時には以下の処理を行う。

チャンク系列 $\boldsymbol{X}^{i-1} = X_1, X_2, ..., X_{i-1}$ をチャンク系列 $\widetilde{\boldsymbol{Y}}^{i-1} = \widetilde{Y}_1, \widetilde{Y}_2, ..., \widetilde{Y}_{i-1}$ に翻訳したとする。次に X_i から $\widetilde{Y}_i$ へと翻訳する際には、過去の全ての X_1^i を含めて、頭から翻訳し直すが、$\widetilde{\boldsymbol{Y}}^{i-1}$ に forced decoding を適用する。プレフィックス $\widetilde{\boldsymbol{Y}}^i$ の確率は以下の式で表され、第一項は (1) 式のフルセンテンスの翻訳と同様に計算される。

$$p_{prefix}(\widetilde{\boldsymbol{Y}}^i|\boldsymbol{X}^i) = p_{full}(\widetilde{\boldsymbol{Y}}^{i-1}|\boldsymbol{X}^i) \times p_{chunk}(\widetilde{Y}_i|\boldsymbol{X}^i, \widetilde{\boldsymbol{Y}}^{i-1}). \quad (2)$$

第二項は、$\widetilde{Y}_i = y_1^i, y_2^i, ..., y_{|\widetilde{Y}_i|}^i$ として、以下のように分解される。

$$p_{chunk}(\widetilde{Y}_i|\boldsymbol{X}^i, \widetilde{\boldsymbol{Y}}^{i-1}) = \prod_{t=1}^{|\widetilde{Y}_i|} P(y_t^i|\boldsymbol{X}^i, \widetilde{\boldsymbol{Y}}^{i-1}, y_{<t}^i). \quad (3)$$

チャンクベースのデコーディングは、チャンクごとに Encoder をリフレッシュする。チャンクは複数単語で構成されることが多いため、新しい単語が入ってくるたびに Encoder をリフレッシュする漸進的な Transformer[1] よりも効率的である。

3.2 構文ラベル予測

現在の時間ステップにおいて、文のプレフィックスの後に来る構文構成ラベルを予測する。この処理を ICLP (Incremental Constituent Label Prediction) と呼ぶ。ここで、次の構文ラベルとは、木の先行順において文のプレフィックスの次に来る構成要素であると定義する。しかし、ある統語要素の次には通常多数の統語要素が出現し得るため、このままでは予測が難しい。そこで提案手法では、一つ先の単語を見て、その単語から始まる構文要素のラベルを予測することとする。入力系列 $W = [w_1, w_2, ..., w_{|W|}]$ において、部分単語系列 w_i から構文ラベル c_i を予測す

る式は以下で表される。

$$c_i = \underset{c' \in C}{\operatorname{argmax}}\, p(c'|w_{\leq i}), \tag{4}$$

C は構文ラベル集合である。

3.3 セグメンテーションルール

表 2 はセグメンテーションの例を示す。我々は一つの基本ルールと二つの補助ルールを提案する。

- S と VP の直前で区切る。
- 前のラベルが S または VP であれば区切らない。
- チャンクが最小セグメント長より小さければ区切らない。

SVO 型から SOV 型へと翻訳する際には、原言語の主語と動詞の間では問題なく区切れるが、動詞と目的語の間では並べ替えが発生するため、区切るべきではない。よって第一のルールとして、VP の前で区切ることとした。Table 3 のように、“(S (VP …))” の形で現れることも多いので、S もルールに追加した。

表 2 の “can save” のように、VP が連続する場合は、“can” と “save” で分けるのは適切ではないため、2 つ目のルールを設けた。

また、精度と遅延をコントロールするため、三つ目のルールとして、最小セグメント長を推論時のハイパーパラメータとして与えた。

4 実験

提案手法の遅延と精度のトレードオフ、そして翻訳タイミングの制御を既存手法と比べるため、英日同時翻訳の実験を行った。

4.1 データと前処理

英日対訳データとして WMT2020 の約 1790 万文で学習し、IWSLT2017 の約 22.3 万文でファインチューニングを行った。IWSLT の dev2010、tst2011、tst2012、tst2013 からなる 5312 文のペアを開発データに使用し、IWSLT の dev2021 の 1442 文でモデルを評価した。

英語は Moses [8] で、日本語は MeCab [9] で単語分割した。BPE でサブワード分割し、語彙サイズは英日共有で 1.6 万とした。ICLP は Penn Treebank 3 を用いて学習し、その 1%を開発データとした。そして、NAIST-NTT TED Talk Treebank で評価した。

4.2 モデル設定

全ての翻訳モデルは fairseq [10] の Transformer-base [11] をベースに実装され、IWSLT2021 のベースモデルの設定に従った [12, 13]。wait-k では k を、MU ではセグメント境界である確率の閾値を、固定長区切りではセグメント長を、ICLP では最小セグメント長を変えることによって遅延を制御した。値の範囲などの詳細は付録に記述する。

ICLP は 2 種類実装し、比較した。一つは 2 層の単方向 LSTM [14] に全結合層を加えたもので、エンベディングと隠れ層は 512 次元とし、入力は Moses でトークナイズし、BPE[15] で語彙 16K のサブワードに分割した。もう一つは、BERT を用いたもので、Huggingface transformers[16] の `bert-base-uncased` を用いて、それに付随するトークナイザを用いた。これらに、文のプレフィックスを入力し、構文ラベルを予測した。

一つ先の単語を見ることで精度の向上が見られ、LSTM と BERT の差は小さかった。詳細は付録参照、

4.3 評価

simuleval [17] を用い、精度は BLEU [18] で、遅延の大きさは Average Lagging (AL) [1] で測った。AL は以下の式で表される。

$$AL_g(X, Y) = \frac{1}{\tau_g(|X|)} \sum_{t=1}^{\tau_g(|X|)} g(t) - \frac{t-1}{\gamma}. \tag{5}$$

$g(t)$ は、非減少関数で、t 番目の目的言語単語を出力するために読む原言語の単語数を示す。$\tau_g(|X|)$ は原言語文全体を読み終えた時のデコードステップである。最後の原言語トークンを読み込んだ直後に予測される $\tau_g(|X|)$ 番目のターゲットトークンまでの遅延をカウントする。γ は $|Y|/|X|$ である。

4.4 結果

図 2 は遅延パラメータを変化させたときの BLEU と AL の関係を示す。提案手法は、ベースラインの左上にあり、遅延と精度のトレードオフで上回った。VP に S を加えることによって、境界の数が増え、点が左側に移動し、遅延を抑えた。wait-k のように単純なモデルとその他では大きな差があった。その一方、固定長区切りのチャンクデコーディングは単純だが、wait-k よりも精度が提案手法に近

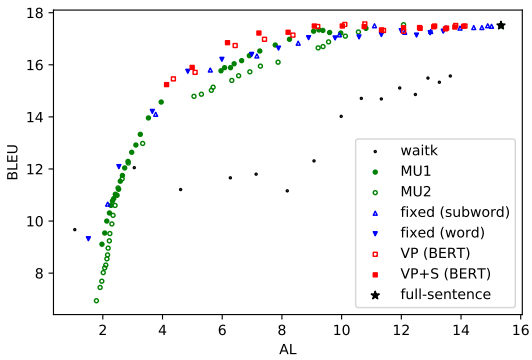

図 2　BLEU と AL の散布図。 MU1 と MU2 はそれぞれ境界予測に 1 つまたは 2 つ先の単語を用いた MU。fixed はワブワード、または単語単位で固定長に区切って翻訳したモデル。

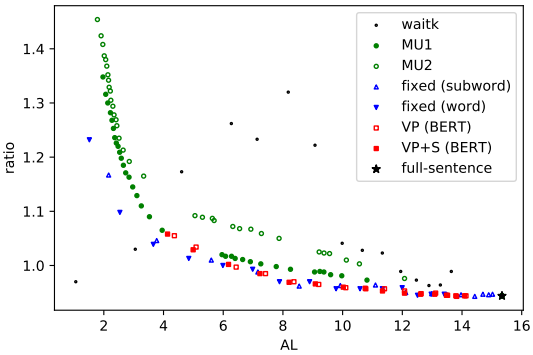

図 3　Length Ratio と AL の散布図。

かった。

5　分析

5.1　セグメント長分布

図 3 に機械翻訳と参照訳の長さ比である length ratio と AL の関係を示す。Wait-k は遅延ごとにモデルを学習するので不安定であるが、それ以外のモデルに関しては、Ratio は AL が小さく遅延が小さいほど、大きくなっている。AL が小さいということは、平均セグメント長が短いということであり、その場合には、想定された長さより長いセグメント翻訳が出力されているということがわかる。

図 4,5 は AL が 7.2 に近い時のテストセットにおける原言語のセグメント長分布を示す。長さは各セグメントに含まれるサブワード数である。ICLP に比べると、MU は、より広い分布を持ち、多くのセグメントが 2 トークンから構成されている。このような短いセグメントは、周囲のコンテクストが使えず、上記のように長めに翻訳されることが原因で、MU の BLEU の低下につながったと考えられる。

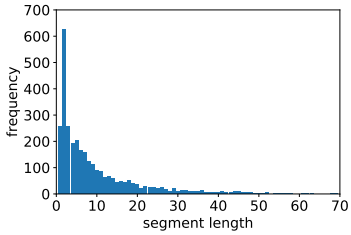

図 4　1 つ先の単語を利用した MU のセグメント長分布 (AL=7.26, BLEU=16.53)

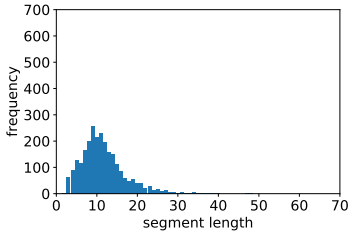

図 5　ICLP のセグメント長分布 (AL=7.23, BLEU=17.22)

5.2　遅延制御

図 4,5 で示されているように、提案手法の方が分布の広がりが狭く、より翻訳タイミングの間隔が安定している。

最小セグメント長というハイパーパラメータによってどのようなタイミングで翻訳がされるかということが予測しやすくなることに加え、構文ラベルのルールによって、出力がどのような構文になるかもある程度制御できるようになる。本稿で提案したルールでは、フルセンテンスの翻訳に近づけることを目指した。しかし、それよりも遅延を小さくしたい場合でも、NP の前でも区切るというルールを加えるだけで、表 1 のような順送りだが内容を理解できる翻訳を出力することもできる。実際の出力例を付録に記載する。

6　おわりに

本稿では、構文ラベル予測モデルとそれに基づくルールを使った同時翻訳を提案した。翻訳出力の制御もしやすくなり、ベースラインを精度と遅延のトレードオフで上回った。今後は、このようなルールを自動で発見して、他の言語対にも適用することが期待される。

謝辞

本研究の一部は JSPS 科研費 JP21H05054 と JP21H03500 の助成を受けたものである。

参考文献

[1] Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, Hua Wu, and Haifeng Wang. STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework. In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**, pp. 3025–3036, Florence, Italy, July 2019. Association for Computational Linguistics.

[2] Colin Raffel, Minh-Thang Luong, Peter J Liu, Ron J Weiss, and Douglas Eck. Online and lineartime attention by enforcing monotonic alignments. In **Proceedings of the 34th International Conference on Machine Learning-Volume 70**, pp. 2837–2846, 2017.

[3] Naveen Arivazhagan, Colin Cherry, Wolfgang Macherey, Chung-Cheng Chiu, Semih Yavuz, Ruoming Pang, Wei Li, and Colin Raffel. Monotonic infinite lookback attention for simultaneous machine translation. In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**, pp. 1313–1323, Florence, Italy, July 2019. Association for Computational Linguistics.

[4] Xutai Ma, Juan Pino, James Cross, Liezl Puzon, and Jiatao Gu. Monotonic multihead attention. In **ICLR 2020**, 2020.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[6] Ruiqing Zhang, Chuanqiang Zhang, Zhongjun He, Hua Wu, and Haifeng Wang. Learning adaptive segmentation policy for simultaneous translation. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 2280–2289, Online, November 2020. Association for Computational Linguistics.

[7] Yusuke Oda, Graham Neubig, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura. Syntax-based simultaneous translation through prediction of unseen syntactic constituents. In **Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)**, pp. 198–207, Beijing, China, July 2015. Association for Computational Linguistics.

[8] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In **Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions**, pp. 177–180, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[9] Taku Kudo. Mecab : Yet another part-of-speech and morphological analyzer., 2005. http://mecab.sourceforge.net/.

[10] 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, June 2019. Association for Computational Linguistics.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszko-reit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and andIllia Polosukhin. Attention is all you need. **CoRR**, p. Vol.abs/1706.03762, 2017.

[12] How to reproduce the result of wmt14 en-de on transformer base model?, 2018. https://github.com/pytorch/fairseq/issues/346.

[13] An example of english to japaneses simultaneous translation system, 2021. https://github.com/pytorch/fairseq/blob/master/examples/simultaneous_translation/docs/enja-waitk.md.

[14] Sepp Hochreiter, J ¨ urgenSchmidhuber. **Long short-term memory**. Neural Computation, 1997.

[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, August 2016. Association for Computational Linguistics.

[16] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**, pp. 38–45, Online, October 2020. Association for Computational Linguistics.

[17] Changhan Wang Jiatao Gu Juan Pino Xutai Ma, Mohammad Javad Dousti. Simuleval: An evaluation toolkit for simultaneous translation. In **Proceedings of the EMNLP**, 2020.

[18] 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, July 2002. Association for Computational Linguistics.

A　付録

A.1　翻訳モデルのハイパーパラメータ

wait-k

k の範囲は、[2, 4, 6,..., 30]

MU

ハイパーパラメータ p はセグメント境界である確率の閾値を表す。p の範囲は、p are [0.5, 0.1, 0.15,..., 0.95], [0.99, 0.991, 0.992,..., 0.999], and [0.9991, 0.9992,..., 0.9999]。境界予測の際には、1 つ、または 2 つ先のトークンを利用した。

Fixed-size Segmentation

これは入力系列を固定長で区切ってチャンクデコーディングを行う。セグメントサイズを表すハイパーパラメータ f の範囲は、単語単位では、[2, 4, 6,..., 30] 、サブワード単位では、 [4, 8, 12,..., 60] とした。

ICLP

ハイパーパラメータは、セグメントを構成する最小単語数 m で、その範囲は、 [1, 2, 3, ..., 29].。

Wait-k は k トークン待った後は、一つの単語の入力後、必ず一つの単語を出力しなければならないため、greedy decoding を行った。それ以外のモデルは、beam search をビーム幅 4 で行った。

A.2　ルールの違いによる実際の出力例

原文	I /like / delicious food .
ラベル	/ VP / NP / NN / .
翻訳	私は / 好きです / 美味しい食べ物.

表 3　VP+NP で区切った結果

原文	I / like delicious food .
ラベル	/ VP / NP / NN / .
翻訳	私は / 美味しい食べ物が好きです.

表 4　VP のみで区切った結果

A.3　ICLP モデルの比較

Label	Precision	Recall	F1
NP	0.90	0.94	0.92
VP	0.89	0.97	0.93
NN	0.95	0.97	0.96
,	0.98	1.00	0.99
PP	0.85	0.93	0.89
S	0.87	0.52	0.65

表 5　1 つ先の単語を用いた BERT ベースの ICLP

Label	Precision	Recall	F1
NP	0.85	0.89	0.87
VP	0.91	0.94	0.92
NN	0.93	0.92	0.92
,	0.98	1.00	0.99
PP	0.78	0.94	0.86
S	0.84	0.52	0.64

表 6　1 つ先の単語を用いた LSTM ベースの ICLP

Label	Precision	Recall	F1
NP	0.62	0.85	0.72
VP	0.75	0.80	0.78
NN	0.60	0.78	0.68
,	0.41	0.34	0.37
PP	0.50	0.47	0.48
S	0.77	0.62	0.69

表 7　先の単語を用いない BERT ベースの ICLP

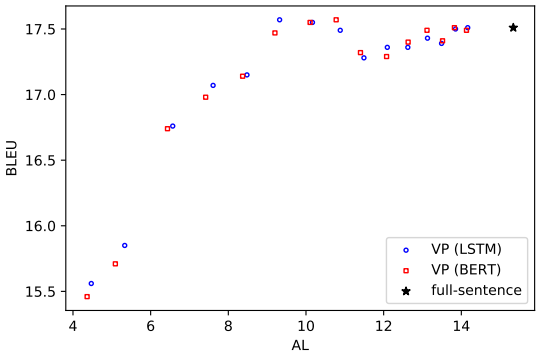

図 6　LSTM ベースと BERT ベースの ICLP の英日同時翻訳比較