翻訳文の部分構造を制約とした機械翻訳

帖佐 克己 [1,2]　上垣外 英剛 [2]　渡辺 太郎 [2]
[1]NTT コミュニケーション科学基礎研究所　[2] 奈良先端科学技術大学院大学
katsuki.chousa@ntt.com　{kamigaito.h, taro}@is.naist.jp

概要

語彙制約付き機械翻訳は，指定された語句を訳語として含む文を生成するという制約の下で機械翻訳を行うタスクである．制約として指定する単位を語句からテキストの構造へ拡張することで，機械翻訳結果への操作性が向上することが期待できるが，これまで構造を制約としたニューラル機械翻訳は取り組まれてこなかった．そこで，本論文では，従来の語彙制約付き機械翻訳手法を拡張し，目的言語側の部分構造を制約とした機械翻訳を行う手法を検討する．

1　はじめに

語彙制約付き機械翻訳は，指定された語句を訳語として含む文を生成するという制約の下で機械翻訳を行うタスクである [1, 2, 3, 4]．訳語を指定することで，特許や法務等での翻訳で重要とされる，文書内での訳語の一貫性を担保できる．また，後編集のように，人間が修正の指示を与えながら翻訳を行う，インタラクティブな翻訳にも語彙制約付き機械翻訳は応用可能である．このタスクは近年活発に取り組まれており，制約を満たした上で高品質な翻訳文の生成が可能になりつつある．

制約として指定する単位を語句からテキストの構造へ拡張することで，機械翻訳の出力に対する操作性がより向上することが期待できる．テキストはその背後に構造を持ち，その構造によってテキスト中の語句間の関係や文型が表現される．そのため，構文構造や談話構造を用いて，その部分構造を制約とすることで，語彙制約では実現できなかったフレーズ間の関係や文型に対しても操作可能になる．例えば，句構造や依存構造などの構文構造の部分木を制約とすることで態を選んだ訳出が可能になり，日英などの同時通訳で問題となる語順の異なりを解決することができる．また，談話構造を利用することで，特許翻訳などで重要とされる句の並列関係の一貫性が翻訳前後で保たれていることを担保できる．

しかし，構造を制約としたニューラル機械翻訳に関してはこれまで取り組まれていない．また，従来の語彙制約付き機械翻訳手法 [2, 3] が構造を制約とした機械翻訳に適用可能であるかどうかも明らかではない．

本論文では，語彙制約付き機械翻訳手法を拡張し，目的言語側の部分構造を制約とした機械翻訳を行う手法を検討する．構造制約を与えるには翻訳文とその構造が必要であるため，その 2 つの情報をまとめた構造付き翻訳文を 1 つの S 式として表し，原言語文からその S 式を直接生成する．また，構造制約も，葉ノードとして語彙が含まれる S 式の形で与える．このとき，構造制約を含む構造付き翻訳文を探索するために構造制約を考慮する翻訳モデルや構造制約付きデコーディングを行い，構造制約付き機械翻訳を実現する．ASPEC [5] を用いた日英翻訳での実験を行い，構造付き翻訳文を出力する翻訳モデルと制約付きデコーディングを組み合わせる事によって，制約を完全に満たす翻訳文を精度を落とすことなく生成できることが確認できた．また，構造制約を考慮する翻訳モデルが構造制約の充足率が 100%に近い状態で高精度な翻訳文を生成できることも示した．

2　関連研究

構造を制約としたニューラル機械翻訳に関してはこれまで取り組まれていないが，語彙制約付き機械翻訳に取り組んだ研究はこれまでにいくつか提案されている．それらの研究は制約を完全に満たすことを保証するかどうかによって 2 種類に大別され，制約を完全に満たすことを保証する手法はハード制約付き手法，そうでない手法をソフト制約付き手法と呼ばれる [3]．

ハード制約付き手法では，主にデコーディングの過程で全ての指定語句を含む系列を探索する方法が用いられる．Post ら [2] はビームサーチの遷移や状

態の持ち方を拡張することで制約付きデコーディングを実現している．Lu ら [6] は，トークンの先読みを行い，将来的な制約の充足率に基づく探索による制約付きデコーディング手法を提案している．これらの手法は制約を満たすことを保証できるが，従来の翻訳器と比べて探索に大きい計算コストを必要とし，入力によっては翻訳精度が低下してしまう．

一方で，ソフト制約付き手法では，主に翻訳モデルへの入力を工夫する方法が採用されている．LeCA [3] は，原文の末尾に語彙制約を結合した系列をモデルに入力するというシンプルな手法で，一定の制約の充足率を達成している．他にも，大規模言語モデル（LLM）のプロンプトとして対訳となるフレーズのペアに関する指示を与えて翻訳を行う手法も提案されている [7]．このような手法はハード制約の手法に比べて高速に動作するが，いくつかの指定語句が出力されない可能性がある．また，LLM を用いた構造制約付き翻訳に関しては，自然言語で厳密に翻訳文の構造に関する指示を与えるのは難しいことや，LLM で構造付き翻訳文を生成する方法が明らかではないという課題がある．

また，この両方の手法を組み合わせることによって，ハード制約を満たした上で，ハード制約単体よりも少ない計算コストで，高い翻訳精度を実現できることも報告されている [4]．

3 提案手法

3.1 定式化

構造制約付き機械翻訳は，I トークンの原文 $X = (x_1, \cdots, x_I)$ と翻訳先言語の部分構造の集合 $C = (C_1, \cdots, C_K)$ が構造制約として与えられたときに，その部分構造を含む J トークンの翻訳文 $Y = (y_1, \cdots, y_J)$ を生成するタスクである．このとき，制約として与えられたリスト中の順序と，翻訳文中の部分構造の出現順序は必ずしも一致しない．また，語彙制約は翻訳文中にそのまま含まれていたのに対して，本タスクでは構造制約がそのまま翻訳文に含まれない．そのため，生成された翻訳文が制約を満たしているかどうかを判定するには翻訳文の構造を別途獲得する必要がある．さらに，従来の翻訳モデルでは生成過程において翻訳文の構造にアクセスできないため，既存の制約付きデコーディング手法を構造制約に適用することが難しい．

翻訳文とその構造を得るために，その構造を S 式として線形化して 1 つの文字列で表した，L トークンの構造付き翻訳文 $Y^{str} = (y_1^{str}, \cdots, y_L^{str})$ を直接生成する手法 [8, 9] を利用する．また，各構造制約 C_k は，Y_{str} と同じ種類の構造による，連続した 1 つの S 式として与えられる．そして，制約 C を満たす系列のうちから以下の条件付き確率を最大化する系列を探索することで，構造制約付き機械翻訳を実現する．

$$p(Y^{str} \mid X, C) = \prod_{l}^{L} p(y_l^{str} \mid y_{<l}^{str}, X, C). \quad (1)$$

このとき，与えることができる構造制約 C_k は翻訳文の構造の部分木に制限されるが，これらの構造制約が構造付き翻訳文 Y_{str} の部分文字列として現れるようになる．また，生成過程において翻訳文の構造を参照することができるため，語彙制約付き機械翻訳と同様の枠組みで構造制約付き機械翻訳を実現できる．

3.2 構造制約付き翻訳モデル

生成過程で翻訳文の構造を利用するために，原文から構造付き翻訳文を生成する，**構造付き翻訳モデル**を作成する．翻訳モデルの学習に用いられる対訳コーパスには一般的に翻訳文の構造情報が付与されていないため，既存のパーサを用いて翻訳文の構造情報を獲得し，S 式として表される構造付き翻訳文を作成する．この構造付き翻訳文と原文のペアを用いて，従来の翻訳モデルと同様の枠組みでモデルを学習することで，構造付き翻訳モデルを作成する．

さらに，翻訳モデル中で構造制約を考慮して翻訳を行う**構造制約付き翻訳モデル**を作成するために，LeCA [3] と同様に，原文と制約を結合して入力系列を拡張する手法を用いる．この入力系列は，区切りとなるシンボルである ⟨sep⟩ で原文 X と制約 C 中の各部分構造 C_k を連結することにより，以下のように作成する．

$$[X, \langle \text{sep} \rangle, C_1, \langle \text{sep} \rangle, C_2, \cdots, C_K, \langle \text{eos} \rangle] \quad (2)$$

ここで，⟨eos⟩ は文の終端を表すシンボルである．

モデルの学習時にも式（2）の入力系列は用いられるが，既存の対訳コーパスには構造制約の情報が付与されていない．そこで，正解の翻訳文から構造制約を作成し，構造制約の擬似的な正解データとして利用する．この構造制約は，構造付き翻訳文から構造同士の重なりが無いように K 個の部分構造をランダムに抽出することで作成する．このとき，制

表 1　開発用・テスト用セットの構造制約の統計情報

平均 (標準偏差)	1 制約あたりのトークン数	1 文あたりの制約数
開発	21.31 (14.52)	1.52 (0.65)
テスト	21.76 (14.64)	1.51 (0.65)

約の個数 K はハイパーパラメータとして与えられる確率分布に従って，ランダムに決定する．

3.3　構造制約付きデコーディング

本タスクでは与えられた制約を必ず満たす翻訳文を生成する必要があるが，前述の翻訳モデル単体では生成した翻訳文が制約を満たすことを保証することができない．そこで，構造付き翻訳モデルのデコーディング時に，語彙制約付きデコーディング（lexically constrained decoding; LCD）を用いる [2]．LCD は翻訳文の探索のためにビームサーチを行う際に，制約を考慮して翻訳文の候補を選択を行う手法であり，これにより制約を満たした翻訳文が得られることを保証できる．この手法は元々は語彙制約のための制約付きデコーディングとして提案されたものだが，3.1 節の通り，構造制約を構造付き翻訳文の部分文字列とすることで，語彙制約付き機械翻訳と同様の枠組みで構造制約を扱うことができるため，この手法を構造制約のための制約付きデコーディング手法として利用することができる．

4　実験

提案手法の有効性を評価するため，日英翻訳を対象として，句構造を構造制約とした構造制約付き機械翻訳の制度評価を行った．

4.1　データセット

学習および評価に使用する対訳コーパスとして ASPEC [5] を使用した．ASPEC の学習データのうち，最後の 100 万文対に関してはノイズが多く含まれていることから，最初の 200 万文対のみを学習データとして使用した．テキストの構造情報には句構造を使用した．`SuPar` [1] の CRF モデル [10] を用いて構文解析を行い，S 式で表される構造付き翻訳文を作成した．`SentencePiece` [11] を用いてサブワードへの分割を行った．語彙は日本語と英語で共有とし，サイズは 32,000 とした．また，学習時に過度に長い文を用いるのを避けるために，`Moses` の `clean-corpus-n.perl` [2] によって文長が 512 を超えるものを学習データから削除した．

学習データの構造制約の作成に際して，各制約の最大トークン数は 150，最小の木の深さは 2 とした．また，1 文あたりの構造制約の最大個数は 3 とし，各文に対する制約の個数 K は以下の分布 $p(K)$ に従ってサンプリングを行い決定した．

$$p(K) = \begin{cases} 0.3 & (K = 0) \\ 0.7/3 & (1 \leq K \leq 3) \end{cases} \tag{3}$$

開発およびテスト用データセットの構造制約についても，学習データと同様の方法で作成した．表 1 に，開発およびテスト用データセットの構造制約の統計情報を示す．

4.2　翻訳モデル

本実験では以下の 3 つの翻訳モデルを作成し，評価に用いた．

ベースライン　原文から翻訳文を生成する，一般的な翻訳モデル

構造付き翻訳モデル　原文から構造付き翻訳文を生成する翻訳モデル

構造制約付き翻訳モデル　原文と構造制約から構造付き翻訳文を生成する翻訳モデル

ベースラインには Transformer（big） [12] を用いた．制約付きデコーディングを用いる際には通常よりも大きいビームサイズを使用する必要があるため，本実験ではビームサイズを 20 とした．その他のベースラインの設定およびハイパーパラメータの詳細については付録の表 3 に示す．その他のモデルについても，特に明記していない場合はベースラインに準じた設定およびハイパーパラメータを使用している．モデルの実装には `fairseq` [13] を用いた．

4.3　評価尺度

構造制約付き機械翻訳の評価は，語彙制約付き機械翻訳における観点を基に，翻訳文の翻訳精度および構造制約の充足率という観点に基づいて行った．

翻訳精度の評価尺度には自動評価尺度としてデファクトスタンダードな手法である BLEU を使用し，その計算には `sacrebleu` [14] を用いた．本実験で用いたモデルには構造付き翻訳文を出力するものもあるが，それらの翻訳精度の評価に際しては，出

1）https://github.com/yzhangcs/parser

2）https://github.com/moses-smt/mosesdecoder/blob/master/scripts/training/clean-corpus-n.perl

表 2 各手法の翻訳精度および制約の充足率．太字は各尺度で最も高いスコアを示す．

	BLEU	Term%	Sent%
ベースライン	28.5	-	-
構造付き翻訳モデル	27.7	19.54	10.76
＋制約付きデコーディング	28.7	**99.73**	**99.67**
構造制約付き翻訳モデル	**44.8**	98.53	97.90
＋制約付きデコーディング	34.5	98.61	98.12

力から構造情報を取り除いて翻訳文のみを抽出して計算を行った．

構造制約の充足率を測る尺度には，語彙制約付き機械翻訳でも用いられている Term%と Sent%の 2 つの尺度を用いた [4]．Term%は構造制約のうち正しく生成された構造制約の割合，Sent%は構造制約をすべて満たす翻訳文の割合として定義される．

4.4 実験結果

各手法の翻訳文の翻訳精度および構造制約の充足率を表 2 に示す．

まず，構造付き翻訳モデルの翻訳精度が，ベースラインと比べて BLEU が 0.8 ポイントの減少と，大きく低下していないことがわかる．構造付き翻訳モデルは S 式で表される構造付き翻訳文を生成することで翻訳とその構文解析を同時に行う．そのため，ベースラインのような翻訳だけを行うタスクよりも難しく，翻訳精度が低くなってしまう可能性があった．この懸念に対して，結果より，構造付き翻訳文を生成する方法でもベースラインと同程度の精度で翻訳が行えることが確認できた．

次に，構造付き翻訳モデルに制約付きデコーディングを組み合わせた結果に着目すると，構造付き翻訳モデル単体と比べて翻訳精度が BLEU で 1 ポイント改善しており，さらに構造制約の充足率もほぼ 100%を達成していることが確認できる．このとき，ハード制約を満たすはずの制約付きデコーディングを適用しているにもかかわらず Term%と Sent%が 100%になっていないが，これはトークナイズ時の文字の正規化などが表記ゆれが発生しているためであり，その点に対応すると両方の充足率は 100%を達成している．また，WAT の語彙制約付き翻訳データセット [1] での語彙制約の平均単語数が 6.6〜7.4 単語であるのに対して，今回作成した構造制約のトークン数は平均 21 トークン前後と，語彙制約が S 式で与えられることによりトークン数が増加している．このような長いトークンからなる構造制約に対しても，制約付きデコーディングを用いることで，制約を満たす翻訳文を精度を落とすことなく探索できることが確認できた．

また，構造制約付き翻訳モデルの結果に着目すると，ベースラインに比べて BLEU が+16.3 ポイントと，翻訳精度が大幅に改善していることが確認できる．さらに，構造制約の充足率についても，制約付きデコーディングを使用していないにもかかわらず Term%と Sent%の両方においてほとんど 100%に近いスコアを達成しており，ほとんどの構造制約は満たされている．この高い充足率の理由としては，構造制約のトークン数が非常に大きいということが考えられる．構造制約のトークン数が増加するということは，構造制約付き翻訳モデルの入力から出力にそのままコピーするトークンの数が増えるということになるため，モデルの学習過程において入力をコピーするようにモデルが強く学習した結果，非常に高い充足率に繋がった可能性が考えられる．

最後に，構造制約付き翻訳モデルに制約付きデコーディングを組み合わせた際の結果に着目すると，構造制約付き翻訳モデル単体と比べて制約の充足率は改善していることがわかる．一方で，翻訳精度については，ベースラインや構造付き翻訳モデルを用いた手法に比べると改善しているが，構造制約付き翻訳モデル単体と比べると減少している．これは，制約付きデコーディングによって構造制約を考慮した翻訳文の候補の選択が行われるために，一般的なビームサーチと比べて十分な探索が行えずに翻訳精度が減少しているのではないかと考えられる．

5 まとめ

本論文では語彙制約付き機械翻訳を拡張し，目的言語側の部分構造を制約とした構造制約付き機械翻訳タスクを提案した．さらに，その構造制約付き機械翻訳を行う手法として，翻訳文とその構造を同時に出力する構造付き翻訳モデルやそのモデルを拡張して構造制約も考慮した構造制約付き翻訳モデル，構造制約付きデコーディングを提案した．ASPEC を用いた日英翻訳での実験を行い，構造付き翻訳モデルと制約付きデコーディングを組み合わせる事によって，制約を完全に満たす翻訳文を精度を落とすことなく生成できることが確認できた．また，構造制約付き翻訳モデルが構造制約の充足率が 100%に近い状態で高精度な翻訳文を生成できることも示した．

参考文献

[1] Toshiaki Nakazawa, Hideki Nakayama, Chenchen Ding, Raj Dabre, Shohei Higashiyama, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, Shantipriya Parida, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, Yusuke Oda, and Sadao Kurohashi. Overview of the 8th workshop on Asian translation. In Toshiaki Nakazawa, Hideki Nakayama, Isao Goto, Hideya Mino, Chenchen Ding, Raj Dabre, Anoop Kunchukuttan, Shohei Higashiyama, Hiroshi Manabe, Win Pa Pa, Shantipriya Parida, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, Yusuke Oda, Katsuhito Sudoh, Sadao Kurohashi, and Pushpak Bhattacharyya, editors, **Proceedings of the 8th Workshop on Asian Translation (WAT2021)**, pp. 1–45, Online, August 2021. Association for Computational Linguistics.

[2] Matt Post and David Vilar. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, **Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)**, pp. 1314–1324, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[3] Guanhua Chen, Yun Chen, Yong Wang, and Victor O.K. Li. Lexical-constraint-aware neural machine translation via data augmentation. In **Proceedings of IJCAI 2020: Main track**, pp. 3587–3593, 7 2020.

[4] Katsuki Chousa and Makoto Morishita. Input augmentation improves constrained beam search for neural machine translation: NTT at WAT 2021. In **Proceedings of the 8th Workshop on Asian Translation (WAT2021)**, pp. 53–61, Online, August 2021. Association for Computational Linguistics.

[5] Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. ASPEC: Asian scientific paper excerpt corpus. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, **Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)**, pp. 2204–2208, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA).

[6] Ximing Lu, Sean Welleck, Peter West, Liwei Jiang, Jungo Kasai, Daniel Khashabi, Ronan Le Bras, Lianhui Qin, Youngjae Yu, Rowan Zellers, Noah A. Smith, and Yejin Choi. NeuroLogic a*esque decoding: Constrained text generation with lookahead heuristics. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, **Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 780–799, Seattle, United States, July 2022. Association for Computational Linguistics.

[7] Yasmin Moslem, Rejwanul Haque, John D. Kelleher, and Andy Way. Adaptive machine translation with large language models. In Mary Nurminen, Judith Brenner, Maarit Koponen, Sirkku Latomaa, Mikhail Mikhailov, Frederike Schierl, Tharindu Ranasinghe, Eva Vanmassenhove, Sergi Alvarez Vidal, Nora Aranberri, Mara Nunziatini, Carla Parra Escartín, Mikel Forcada, Maja Popovic, Carolina Scarton, and Helena Moniz, editors, **Proceedings of the 24th Annual Conference of the European Association for Machine Translation**, pp. 227–237, Tampere, Finland, June 2023. European Association for Machine Translation.

[8] Oriol Vinyals, L ukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. Grammar as a foreign language. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 28. Curran Associates, Inc., 2015.

[9] Roee Aharoni and Yoav Goldberg. Towards string-to-tree neural machine translation. In Regina Barzilay and Min-Yen Kan, editors, **Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 132–140, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[10] Yu Zhang, Houquan Zhou, and Zhenghua Li. Fast and accurate neural CRF constituency parsing. In **Proceedings of IJCAI**, pp. 4046–4053, 2020.

[11] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Eduardo Blanco and Wei Lu, editors, **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**, pp. 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 30. Curran Associates, Inc., 2017.

[13] 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 Waleed Ammar, Annie Louis, and Nasrin Mostafazadeh, editors, **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.

[14] Matt Post. A call for clarity in reporting BLEU scores. In **Proceedings of the Third Conference on Machine Translation: Research Papers**, pp. 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics.

A　ベースラインの設定

ベースラインに用いた翻訳モデルの設定及びハイパーパラメータを表 3 に示す.

表 3　ベースラインのモデルの設定およびハイパーパラメータ

アーキテクチャ	Transformer(big) [12]
Tied-embeddings	Encoder/Decoder の Embedding と Decoder の最終層の重みを共有
Optimizer	Adam ($\beta_1 = 0.9, \beta_2 = 0.98, \epsilon = 1e-8$)
学習率の更新方法	Inverse square root decay
Warmup	4,000 ステップ
学習率の最大値	0.001
Dropout	0.3
Gradient Clipping	1.0
Label Smoothing	0.1
ミニバッチサイズ	512,000 トークン
更新回数	30,000 ステップ
ビームサイズ	20