文法誤り訂正モデルは訂正に必要な文法を学習しているか

三田雅人
理化学研究所　東北大学
masato.mita@riken.jp

谷中瞳
理化学研究所
hitomi.yanaka@riken.jp

1 はじめに

文法誤り訂正（GEC）は，テキストに含まれる文法誤りを自動的に訂正するタスクである．本タスクでは，その構造的な類似性から文法誤りが含まれる文から誤りが含まれない文への機械翻訳とみなして解くアプローチが主流である．そのため，GEC のためのニューラルエンコーダデコーダモデル（EncDec）が数多く提案されており，現在では Transformer [1] に基づくモデルが標準的になっている [2, 3, 4]．最近では，学習データ不足の問題を解決するために擬似データの活用が盛んに行われており，モデルの性能向上が報告されている [5, 6]．

一方で，GEC の実応用を考えたとき，性能だけでなくモデルサイズや推論速度といったモデルの軽量化も重要な観点であるが，現行の擬似データを活用して性能向上を図る方法は，この観点で課題が残る．例えば，Kiyono ら [6] の報告によると，訂正性能を測る標準的な評価指標である $F_{0.5}$ においてたった 2 ポイント性能を向上させるために，約 6,000 万文対もの擬似データを追加で増やす必要があることがわかった．現行の機械翻訳としてのモデル化や擬似データの活用による性能向上の背景には，大量のパラメータを用いた**訂正パターンの丸暗記**という要因が挙げられるが，ここで，**全ての誤りに対してパターンを丸暗記させる必要があるのか**，といった問いが考えられる．なぜなら，主語と動詞の一致誤りのような文法規則に基づく一部の文法誤りに関しては，人間は文法規則さえ学習できていれば必ずしも個々のパターンを丸暗記する必要はないからである．また，モデルは大量のパターンから訂正に必要な文法知識を学習していることを期待されるが，文法知識をどの程度汎化できているかは自明ではない．もし仮に，文法規則に従って訂正できる文法誤りに対して汎化できていないのであれば，ありとあらゆる語彙と構文の組み合わせからなる訂正パターンを丸暗記する必要があることになり，より軽量な

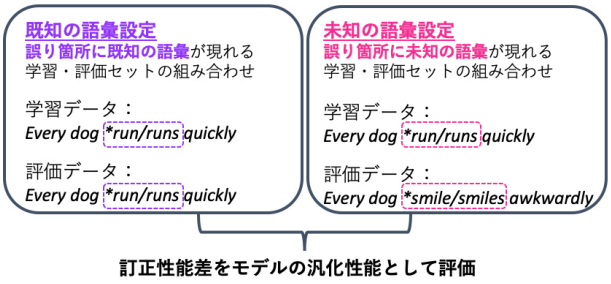


図 1　GEC モデルの汎化性能評価の概要．

GEC モデルを実現させるためには訂正に必要な文法知識をルールとして教える手法を組み合わせるといった改善が必要となる．

そこで本研究では，GEC モデルの軽量化に向けて，モデルが訂正パターンを単に丸暗記しているのではなく，訂正に必要な文法知識を汎化できているかについて分析する手法を提案する．また本研究では，（i）文脈自由文法（CFG）を用いて語彙と構文を制御しながら自動構築した人工データ，（ii）既存の学習者データを用いて語彙を制御しながら自動構築した実データという 2 種類のデータセットを用いて評価を行い，入力文の語彙や構文の複雑さと汎化性能との関係について分析を行う．

2 関連研究

言語モデルが自然言語の文法性を獲得しているかについて分析する先行研究としては，英語の主語と動詞の数の一致に着目した分析 [7, 8] がある．この分析では，言語モデルが *The farmer *smile/smiles* という単語列に対し，正しい活用形である *smiles* に誤りの活用形である *smile* よりも高い確率を付与できるかを評価する．これらの先行研究では言語モデルが単純な統語現象から長距離依存関係を含むような複雑な統語現象を対象として，文法的に正しい文と誤っている文をどの程度区別できるかに焦点を当てている．これに対して本研究では，誤り訂正で重要な誤りタイプを対象として，言語モデルと翻訳モデルから構成される GEC モデルの汎化性能を分析す

表 1　自動構築した人工データおよび実データの例.

誤りタイプ	人工データ	実データ
VERB:SVA	*Every white dog* ***run/runs** quickly*	*My mother and father* ***is/are** really an affectionate couple*
VERB:FORM	*Some white dogs* ***running/ran** quickly*	*I am interested in* ***work/working** with you*
WO	***White every/Every white** dog ran quickly*	*I've never seen it* ***before like this/like this before***
MORPH	*Some white dogs ran* ***quick/quickly***	*We have a good* ***relation/relationship** , she is my main friend*
NOUN:NUM	*Every* ***dogs/dog** ran*	*You know that I love action* ***film/films** like this*

る．先行研究が言語モデルの文法能力のみを分析対象としているのに対し，本研究では誤り検出だけでなく翻訳モデルによる誤り訂正においても汎化性能を評価することで実践的なモデルに求められる文法能力を評価できる．また，先行研究の多くは語彙や文法をコントロールした人工データを用いて評価を行っているのに対して，本研究では人工データと実際の学習者データの両方でモデルの評価を行い，人工データと実データの結果について比較分析を行う．

3　提案手法

本研究では，現行の機械翻訳に基づく GEC モデルが訂正に必要な文法知識を汎化できているか，より具体的には，**未知の語彙に対する汎化**に焦点を当てて調査する．なお，本研究では未知の語彙を含む文法誤りを検出できることと訂正できることを合わせて**文法知識の汎化**と定義する．図 1 は，主語と動詞の一致を例とした提案手法の概要図である．この提案手法では，学習・評価セットの誤り箇所に現れる語彙を制御して自動構築したデータを用いてモデルの評価を行う．具体的には，誤り箇所に既知の語彙が現れる学習・評価セットで評価したときの性能（**既知の語彙設定**）と，誤り箇所に未知の語彙が現れる学習・評価セットで評価した時の性能（**未知の語彙設定**）を比較することで，未知の語彙を含む文法誤りに対するモデルの汎化性能を評価する．

ここで，モデルが（i）語彙を知らなかったため訂正できなかったのか，（ii）文法知識の汎化に失敗したため訂正できなかったのかを切り分けて検証を行うために，出現語彙自体は学習データと評価データで共通となるように制御する．例えば，図 1 の未知の語彙設定の例においては，*Every dog *run quickly* → *Every dog runs quickly* という文対と *Every dog smiles awkwardly* → *Every dog smiles awkwardly* という（無編集な）文対は学習データに出現し，*Every dog *smile awkwardly* → *Every dog smiles awkwardly* という文対は出現しないように制御した学習データを用いてモデルを学習させる．このとき，仮にモデルが *Every dog *smile/smiles awkwardly* という入力文に対し，正しい活用形である *smiles* と訂正できなかった場合は，主語と動詞の一致に関する文法知識を汎化できていないということになる．

本研究では，Bryant ら [9] が定義する誤りタイプの中から，文法規則に基づく誤りである，主語と動詞の一致誤り（VERB:SVA）・動詞の態誤り（VERB:FORM）・語順誤り（WO）・形態素誤り（MORPH）・名詞の単複誤り（NOUN:NUM）の計 5 種類の誤りタイプを対象に調査する．

3.1　人工データを用いた検証

語彙と構文を制御した状況で評価を行うため，CFG の生成規則から自動構築したデータで DNN の推論における汎化性能を分析する手法 [10] を応用して人工データを自動構築し，分析を行う．分析対象である 5 種類の誤りタイプについて，誤りタイプごとに文法的に誤りを含む生成規則と正しい生成規則の 2 種類の生成規則を設計（設計した CFG 生成規則の一部を付録 A に示す）し，これらの生成規則から元文と訂正文をそれぞれ生成する．

VERB:SVA の人工データの例として，文法的に誤りを含む生成規則 $S \rightarrow NP_{pl}\ VP_{sg}$ から元文 *dogs smiles*，正しい生成規則 $S \rightarrow NP_{sg}\ VP_{sg}$ から訂正文 *dog smiles* を生成できる．使用する語彙は生成文が自然となるように一般名詞，自動詞，他動詞，形容詞，副詞各 15 種類を選定し，誤り箇所以外の語彙は共通にした．データサイズは CFG で生成される文を調整することで調整でき，本論文では各誤りタイプを十分に学習できる量の学習データとして，各誤りタイプ 5 万文対のデータを自動構築した．

3.2　実データを用いた検証

3.1 節では，未知の語彙に対する汎化性能にのみ焦点を当てて分析を行うため，CFG を用いて語彙と構文を制御しながら自動構築した人工データを用いた検証について述べた．しかし，CFG に基づいて構

表 2　未知の語彙に対する汎化性能.

データセット		VERB:SVA	VERB:FORM	WO	MORPH	NOUN:NUM
人工データ	既知の語彙設定	99.61	99.17	99.09	98.44	97.47
	未知の語彙設定	46.05	56.93	84.00	29.35	65.55
	Δ	-53.56	-42.24	-15.09	-69.09	-31.92
実データ	既知の語彙設定	87.84	86.36	74.89	87.77	83.75
	未知の語彙設定	6.28	6.28	9.25	3.83	12.49
	Δ	-81.56	-80.08	-65.64	-83.94	-71.26

築した人工データは，GEC の実際の入力として想定される実データとは質が大きく異なる．具体的には，人工データでは語彙や構文が制限された単純な文のみで構成されるが，実データでは，語彙および構文は単純なものから複雑なものまで多種多様である（表 1）．そこで本研究では，より実践的な設定におけるモデルの汎化性能を分析するために実データを用いた検証も併せて行う．

まず，既存の学習者データセットに対して，ERRANT [9] を用いて誤りタイプラベルおよび訂正パターンの自動アノテーションを行う．ここで，本研究では学習者データセットとして，BEA Shared Task で配布された学習データおよび開発データを結合した約 200 万文対のデータ使用した [1]．次に，一文につき一つの訂正パターンになるように誤りタイプおよび訂正パターンを保持しながらデータを分割する [2]．未知の語彙設定は，保持した訂正パターンをもとにデータ全体でソートし，重複のあるものを学習データに，重複のないものを評価データに分類することで構築する．上記手順で誤りタイプごとにサンプリングを行いデータセットを構築した結果，VERB:SVA は 25,889 文対，VERB:FORM は 41,592 文対，WO は 18,779 文対，MORPH は 26,345 文対，NOUN:NUM は 68,002 文対となった．

4　実験

4.1　実験設定

3 節で構築した人工データおよび実データを用いて評価実験を行う．実験に使用した学習・開発・評価セットの詳細は付録 B を参照せよ．評価尺度として，ERRANT [9] によって算出された $F_{0.5}$ を用いる．GEC モデルは，1 節で述べたように，現在では Transformer に基づいた EncDec モデルが標準的になっているため，本評価実験でもこれを採用する．具体的には，seq2seq モデルのツールキット `fairseq` [11] における “Transformer（big）” を使用した [3]．

4.2　実験結果

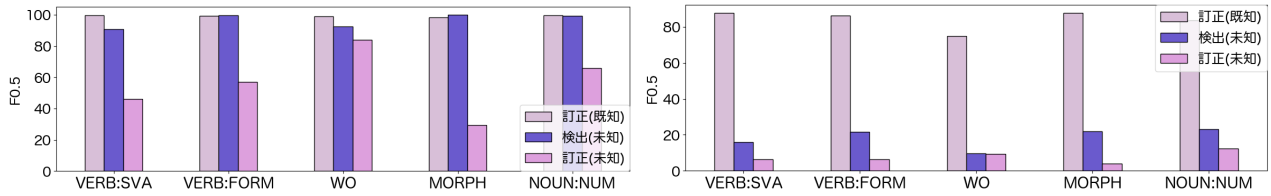
表 2 に評価結果を示す．人工データを用いた検証をみると，WO は除き，モデルは既知の語彙設定に比べて未知の語彙設定で訂正性能が大幅に下がっていることが確認できる．WO が未知の語彙に対して相対的に汎化性能が高かった要因の一つとして，訂正タスクとしての複雑さが考えられる．具体的には，WO は単語の位置がわかれば修正できるが，他の誤りは単語の表層形の違いや特定の単語間の依存関係をみて修正する必要があるため，訂正タスクとして複雑さが増している．

一方で，実データを用いた検証においては，WO を含め全ての誤りが訂正に必要な文法知識を汎化できていないことがわかる．人工データと実データの結果の傾向が異なったことについては，5 節で分析する．以上の結果より，人工データで使用したような比較的単純な語順誤りを除いて，モデルは大量のパターンから訂正に必要な文法知識をほとんど汎化できていないことがわかった．

5　分析・考察

エンコーダ vs. デコーダ　文法知識の汎化に失敗したとき，誤りを検出するのに失敗したのか（**エンコーダ起因の問題**），誤りを検出はしているが正しい単語の復元に失敗したのか（**デコーダ起因の問題**）を分析するために，未知の語彙に対する誤り訂正性能と検出性能の比較を行う．図 2 に，人工データおよび実データを用いた検証における，検出性能と訂正性能の比較を示す．ここで，検出性能の評価は訂正性能と同様に ERRANT を用いて行った．人工データを用いた検証結果（図 2a）では，どの誤り

1）　https://www.cl.cam.ac.uk/research/nl/bea2019st/
2）　一文中に複数の誤りや異なる誤りタイプが含まれる場合も，対象とする誤り以外は訂正をしないことに注意されたい．
3）　ハイパーパラメータの詳細は付録 C を参照せよ．



(a) 人工データ　　(b) 実データ

図 2　検出性能と訂正性能の比較.

表 3　実データにおけるノイズの有無による訂正性能比較（未知の語彙設定）.

	ノイズなし	ノイズあり
VERB:SVA	9.95	5.78
VERB:FORM	12.33	5.47
WO	7.89	9.35
MORPH	6.32	3.90
NOUN:NUM	24.16	12.49

タイプも誤りの検出には成功していることがわかる．一方で，実データを用いた検証結果（図 2b）では，誤り検出の時点で全ての誤りタイプについて大幅に性能が低下していることがわかる．

人工データ vs. 実データ　図 2 の実験結果から，現行の GEC モデルは人工データでは少なくとも誤り検出をある程度汎化できているが，実データでは誤り検出の時点で汎化に失敗していることがわかった．実データにおいてうまく汎化できなかった要因としては大きく，**（i）語彙や構文の多様さ**，**（ii）文中の誤りの複雑さ**の 2 点が考えられる．（ii）については，実データには一文中に複数かつ様々な誤りを含む場合があるため，周囲の誤りが対象の誤りを検出・訂正する際に悪影響を与えてしまった可能性がある．そこで，文中の誤りの複雑さが訂正に与える影響を調査するために，未知の語彙設定における実データを一文中に対象の誤りのみ含まれるデータと対象の誤り以外も含むデータに分け，それぞれにおける訂正性能を評価した．ここで，本論文では一文中に対象の誤りタイプのみ含まれるデータをノイズなし，複数の誤りタイプの誤りを含むデータをノイズありと呼ぶ．表 3 にその結果を示す．この結果から，WO はノイズの有無に関わらず性能が一定であるが，他の誤りはノイズに対して少なからず影響を受けていることがわかる．しかし，ノイズなしの設定においても，既知の語彙設定における訂正性能（表 2 参照）と比較すると大幅に性能が低いことを踏まえると，実データで汎化できなかった要因としては，語彙や構文の多様さが支配的であると考えられる．

言語モデル vs. 翻訳モデル　言語モデルと翻訳モデルから構成される GEC モデルにおいて，表 2 の評価実験は，翻訳モデルが各誤りタイプにおいて訂正性能の向上にどのくらい寄与しているか，に関する一種のアブレーション実験と捉えることが可能である．なぜなら，未知の語彙設定では，正しい言語モデル（*Every dog smiles awkwardly*）はモデルに明示的に教えているが，正しい翻訳モデル（**smile/smiles*）は明示的にはモデルに教えていない設定であるからである．したがって，WO は言語モデルの情報だけでも訂正が可能なのに対して，他の誤りは翻訳モデルも訂正性能の向上に重要な要素であることがわかる．この結果は言語モデルが語順には強いという報告 [12] と整合性のある結果である．以上の結果から，GEC のような実践的なモデルの汎化性能を測るためには言語モデル（検出）の汎化性能だけ見ていては不十分であり，翻訳モデル（訂正）の汎化性能の両方を見る必要性があることが示唆される．

6　おわりに

本研究では人工データと実データの 2 種類のデータを用いて，未知の語彙を含む誤りに対するモデルの性能を評価することで，モデルが訂正に必要な文法知識を汎化できているかについて分析を行った．実験の結果，現行の GEC モデルは人工データのような簡単な設定においては誤りの検出をある程度汎化できている一方で，訂正はほとんど汎化できていないことが示唆された．また，多様な語彙や構文を含む実データでは，誤りの検出も訂正も汎化できていないことが示唆された．今後の展望として，実データではなぜ検出の時点で汎化できていないのか，分析を進める．

謝辞．本研究の一部は理研・産総研「チャレンジ研究」(FS 研究)，JSPS 科研費 JP20K19868 の助成を受けたものである．

参考文献

[1] 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 31 (NIPS 2017)*, pp. 5998–6008, 2017.

[2] Roman Grundkiewicz and Marcin Junczys-Dowmunt. Near Human-Level Performance in Grammatical Error Correction with Hybrid Machine Translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2018)*, pp. 284–290, 2018.

[3] Wei Zhao, Liang Wang, Kewei Shen, Ruoyu Jia, and Jingming Liu. Improving Grammatical Error Correction via Pre-Training a Copy-Augmented Architecture with Unlabeled Data. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2019)*, 2019.

[4] Masahiro Kaneko, Masato Mita, Shun Kiyono, Jun Suzuki, and Kentaro Inui. Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 4248–4254, 2020.

[5] Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2019)*, pp. 252–263, 2019.

[6] Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. An empirical study of incorporating pseudo data into grammatical error correction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019)*, pp. 1236–1242, 2019.

[7] Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies. *Transactions of the Association for Computational Linguistics (TACL)*, Vol. 4, pp. 521–535, 2016.

[8] Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. Colorless Green Recurrent Networks Dream Hierarchically. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2018)*, pp. 1195–1205, 2018.

[9] Christopher Bryant, Mariano Felice, and Ted Briscoe. Automatic Annotation and Evaluation of Error Types for Grammatical Error Correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pp. 793–805, 2017.

[10] Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, and Kentaro Inui. Do Neural Models Learn Systematicity of Monotonicity Inference in Natural Language? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 6105–6117, 2020.

[11] 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 (NAACL 2019)*, 2019.

[12] Richard Futrell and Roger P. Levy. Do RNNs Learn Human-like Abstract Word Order Preferences? In *Proceedings of the Society for Computation in Linguistics (SCiL 2019)*, pp. 50–59, 2019.

[13] Diederik Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*, 2015.

[14] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, pp. 2818–2826, 2016.

A 人工データの構築に用いた CFG 規則

表 4 人工データ構築に用いた CFG 規則（一部）．各誤りタイプの誤りありの生成規則を-誤りタイプで示す．

	生成規則	
S	→	NP VP
$S_{\text{-SVA}}$	→	NP_{sg} VP_{pl} | NP_{pl} VP_{sg}
VP	→	IV | IV ADV | TV NP
$VP_{\text{-FORM}}$	→	IV_{ing} | IV_{ing} ADV | TV_{ing} NP
$VP_{\text{-MORPH}}$	→	IV ADJ
NP	→	Q N | Q ADJ N
$NP_{\text{-WO}}$	→	ADJ Q N
$NP_{\text{-NUM}}$	→	Q_{sg} N_{pl} | Q_{pl} N_{sg} | Q_{sg} ADJ N_{pl} | Q_{pl} ADJ N_{sg}
	語彙項目	
Q	→	{*a, every, no, some, many*}
N	→	{*dog, rabbit, cat, bear, tiger*}
IV	→	{*run, walk, come, dance, leave*}
TV	→	{*kicked, hit, cleaned, touched, accepted*}
ADJ	→	{*white, gray, big, small, large, old*}
ADV	→	{*quickly, slowly, gracefully, seriously, happily*}

B 実験に使用したデータセットの詳細

表 5 人工データにおける分割（訓練/開発/評価）の詳細．

	VERB:SVA	VERB:FORM	WO	MORPH	NOUN:NUM
既知	50,000 / 2,000 / 18,562	50,000 / 2,000 / 10,125	50,000 / 2,000 / 8,438	50,000 / 2,000 / 10,125	50,000 / 2,000 / 8,438
未知	50,000 / 2,000 / 13,749	50,000 / 2,000 / 7,500	50,000 / 2,000 / 6,250	50,000 / 2,000 / 7,500	50,000 / 2,000 / 6,250

表 6 実データにおける分割（訓練/開発/評価）の詳細．

	VERB:SVA	VERB:FORM	WO	MORPH	NOUN:NUM
既知	23,889 / 2,000 / 2,000	39,592 / 2,000 / 2,000	16,779 / 2,000 / 2,000	24,345 / 2,000 / 2,000	66,002 / 2,000 / 2,000
未知	23,889 / 2,000 / 633	34905 / 2,000 / 2000	16,779 / 2,000 / 9,199	24,345 / 2,000 / 5227	66,002 / 2,000 / 3,111

C GEC モデルのハイパーパラメータ

表 7 GEC モデルに使用したハイパーパラメータの一覧．

項目	設定値
モデルアーキテクチャ	Transformer [1]
最適化手法	Adam [13]
学習率	文献 [1] の 5.3 節と同じ
エポック数	30（人工データ），150（実データ）
ドロップアウト	0.3
ロス関数	Label smoothed cross entropy [14]
ビーム幅	5