高再現率な文法誤り訂正システムの実現に向けて

松本悠太 [1]　清野舜 [2,1]　乾健太郎 [1,2]
[1] 東北大学　[2] 理化学研究所
yuta.matsumoto.q8@dc.tohoku.ac.jp,　shun.kiyono@riken.jp,
inui@ecei.tohoku.ac.jp

1　はじめに

文法誤り訂正 (Grammatical Error Correction; GEC) は文法的な誤りを含んだ文を入力とし、正しい文を出力とすることで誤りを訂正するタスクである．一般的に GEC システムの評価にあたっては、参照文との比較により各編集について正解・不正解を計算し、適合率、再現率と $F_{0.5}$ 値を求める [1, 2].

GEC システムの多くには適合率が再現率よりも高いという傾向が見られる．例えば、2019 年に開催された GEC の Shared Task (BEA-2019)[3] の優勝システム [4] では、適合率は 72.28%だったのに対し、再現率は 60.12%であった． また、Shared Task においてシステムの優劣を $F_{0.5}$ で比較していることからも、GEC が分野として適合率の高さを重要視していることが分かる．

一方で、適合率を度外視してでも高い再現率を担保したい状況も考えられる．例えば、英語の母語話者が GEC システムを自分で書いた英文の校正に使用する場面を考える．母語話者としては、全ての文法誤りがシステムによって訂正される（つまり、再現率が 100%である）ことが望ましいはずである．このとき、各訂正の良し悪しはシステムのユーザが判断すれば良いため、必ずしも全ての訂正が正しい必要はない．

我々の最終的な目的は高再現率な GEC システムの実現である．しかしながら、我々の知る限り高再現率を主目的とした研究は存在せず、この目標への取り組み方も明らかではない．そのため、本研究では次の 2 つの戦略のもと，高再現率システムの構築に向けて解決すべき問題の性質を明らかにする．**(1) 各種文法誤りの種類のうち、前置詞の誤りに限定して取り組む．**前置詞の誤りは、名詞や動詞の誤りよりも種類が少ないため、GEC システムの制御や分析が容易になると考えられる．**(2) 再現率向上に寄与するとされる複数の既存手法を組み合わせて、再現率の変化を分析する．**具体的には、疑似データ、リランキングと反復復号化という 3 つの手法を使う．既存手法の効果を分析することで、将来的に新手法の効果をより精緻に分析することが可能になる．

分析では、一連の手法における最高値を知るためにモデルのオラクル出力を獲得し、その再現率を計算した．その後、訂正できなかった誤りの性質を理解するためにオラクル出力を人手評価によって分析した．その結果、参照文で正解とされている編集が必ずしも正しいとは言えない可能性が示唆された．

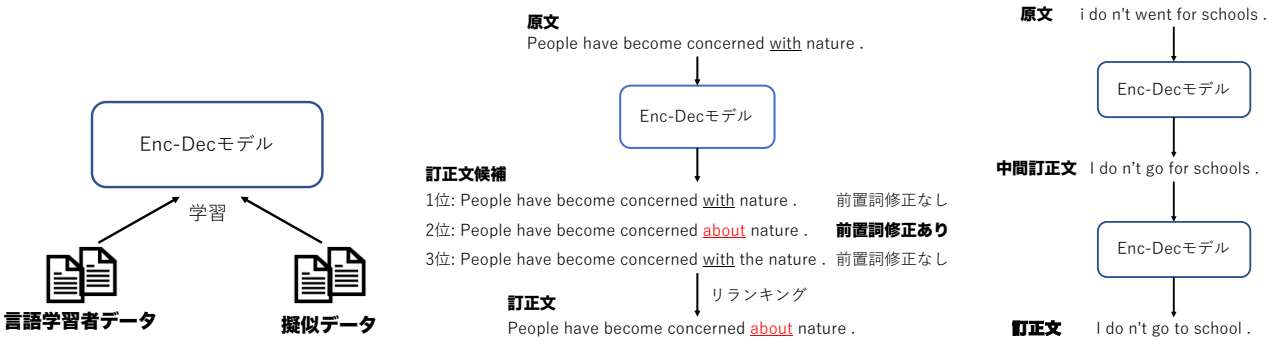
2　再現率を向上させるための手法

本節では、前置詞の再現率向上に有効だと考えられる 3 つの既存手法：(1) 擬似データ (2) リランキング (3) 反復復号化について述べる．Encoder-Decoder (Enc-Dec) モデル [5] を用いた本研究の概観を図 1 に示す．

2.1　擬似データ

GEC の訓練データは言語学習者の書いた英文とその訂正文からなるパラレルコーパスである．ここで、前置詞の用法のうち誤りが含まれているのは全体の 1 割に過ぎず、ほとんどの場合では正しく用いられていることが知られている [6]．このようなデータを用いて訓練をおこなう場合、モデルは前置詞をできるだけ訂正しないような傾向を学習してしまい、結果として、モデルの再現率は低くなってしまうと考えられる．

本研究では、前置詞の誤りを大量に含んだ疑似データを訓練に取り入れることによって、再現率の向上をねらう（図 1(a)）．GEC モデルの訓練においては、元の訓練データに加えて擬似データを用いることが一般的であり [4, 7, 8]、本研究もその枠組みに従う．具体的には、単一言語コーパスに対して前置詞誤りを付与することで、前置詞誤りを含んだ擬似データを生成する．以下にその過程を述べる．



(a) 疑似データを用いた学習　(b) リランキング　(c) 反復復号化

図 1　高再現率 GEC システムの概観

まず Takahashi ら [8] に従い、コーパス中に含まれる前置詞の頻度から、最もよく使われる前置詞 10 個の前置詞セット [9, 8] を作成する．次に、単一言語コーパスの文中の各単語に対して確率 0.1 で前置詞セット中の単語をランダムに挿入する．また、対象の単語が前置詞セットに含まれている場合，誤りを確率 p で付与する．誤りを付与する場合は、(1) 削除、(2) 他の前置詞との置換、(3) 前後の単語との入れ替え をそれぞれ 0.1, 0.8, 0.1 の確率で行う．この確率分布は Grundkiewicz ら [4] を参考にした．

ここで、操作 (2) の置換については、前置詞セットではなく、別に用意した前置詞混乱セット [10] を用いることで、学習者の各前置詞に関する誤り傾向を擬似データ中に取り入れる．具体的には、前置詞 w_i の前置詞混乱セットは、学習者データ中で w_i に多く訂正された上位 4 単語の前置詞 w_j の集合とした．例えば、単語 for は to・of・in・on から高頻度で訂正されるため、for の前置詞混乱セットは {to, of, in, on} となる[1)]．

2.2　リランキング

Enc-Dec モデルのデコード過程では、幅 n のビームサーチによって n 個の候補文が出力される．このとき、通常は Enc-Dec が付与したスコアの最も大きい候補文を選択するが、原文からの前置詞の編集を多く含むような文を選択することができれば、再現率を向上させられるはずである．この直観のもと、前置詞の編集数をスコアとして取り入れたリランキングを提案する（図 1(b)）．これは、他の Enc-Dec モデルや言語モデルを用いたリランキング手法 [7, 11] の亜種であるとみなせる．

リランキングでは、以下のように新しいスコア（NewScore $\in \mathbb{R}$）を計算する．

$$\text{NewScore} = \text{BeamScore} + \alpha * \text{NumOfPrepEdit} \quad (1)$$

ここで、BeamScore $\in \mathbb{R}$ と NumOfPrepEdit $\in \mathbb{N}$ はそれぞれ Enc-Dec が付与したスコアと訂正前後の文間における前置詞の編集数である．前置詞の編集は ERRANT[1] によって検出した．また、重み $\alpha \in \mathbb{R}$ はハイパーパラメータである[2)]．

2.3　反復復号化

反復復号化は、デコード時に得られた文を再びモデルでデコードする手法である（図 1(c)）．これにより、複数の文法誤りを含む文に対して、モデルは段階的に訂正をおこなうことができる．GEC においては複数の既存研究 [12, 13, 14] が反復復号化を採用し、再現率の向上を報告している．そのため、本研究でも手法の一つとしてこれを採用する．

3　実験

3.1　実験設定

データセット　訓練データと検証データには BEA-2019 Shared Task[3] で配布された物を使用する．以降はそれぞれ BEA-train、BEA-valid と言及する．評価データには CoNLL-2014 で使用されたテストデータ (CoNLL-2014)[15] を用いる．擬似データは Gigaword[3)] から生成した．また、これらのデータは全て subword-nmt[4)] によって BPE 化した．各データセットの統計量を表 1 に示す．

性能評価　モデルの性能評価には、BEA-valid と CoNLL-2014 を用いる．ERRANT[1] を用いて誤りタ

1)　前置詞混乱セットの詳細や実際に生成された擬似データについては付録 A を参照されたい．

2)　実際のリランキング例は、付録 B を参照されたい．

3)　https://catalog.ldc.upenn.edu/LDC2003T05

4)　https://github.com/rsennrich/subword-nmt

表 1　各データセットの詳細な統計量

データセット	文 (ペア) の数	参照文の数	データの用途
BEA-train	561,430	1	訓練
BEA-valid	4,384	1	検証
CoNLL-2014	1,312	2	評価
Gigaword	500,000	-	-

表 2　擬似データを加えた時の前置詞誤りと全体に対する性能．太字は最も高再現率だったスコアを表す．

	前置詞性能		全体の性能
前置詞誤り生成確率	適合率	再現率	$F_{0.5}$
N/A(ベースライン)	57.37	38.92	48.18
0.5	43.06	44.82	45.88
0.7	41.43	44.28	46.02
0.9	39.89	44.37	45.50
1.0	40.37	**44.87**	45.76

イプごとに適合率、再現率、$F_{0.5}$ 値を計算した．

モデル　Enc-Dec モデルとして Transformer (big)[5] を用いた．また、最適化には Adafactor[16] を用いた．全てのモデルは、Kiyono ら [7] の作成した事前学習済みモデルを初期値として用いて学習した．この事前学習済みモデルを BEA-train のみで学習させたものをベースラインモデルとする[5)]．ビームサーチの幅は 20 とした．

3.2　実験 1：擬似データ

本節では前置詞誤りの擬似データを生成する際の前置詞誤り付与確率を変化させたときの性能を比較する．誤り付与確率 p を $p = \{0.5, 0.7, 0.9, 1.0\}$ と変化させた時の性能は表 2 のようになった．

表 2 から、擬似データを用いることで前置詞における再現率が向上したとわかる．一方で、再現率の向上幅は誤り付与確率を変化させてもほぼ変わらない．以降の実験 2、3 では、最も高再現率となった $p = 1.0$ の擬似データを含むデータで学習したモデル（$p = 1.0$ の擬似データモデル）を使用する．

3.3　実験 2：リランキング

ベースラインモデルと $p = 1.0$ の擬似データモデルに対して、ビームサーチで得られた候補 20 文について前置詞編集数の重み α を $\alpha = \{0, 0.1, 0.2, 0.3, 0.5, 0.7\}$ と変化させた場合の性能を図 2 に示す．図から、どちらのモデルにおいても、前置詞の編集数を用いたリランキングによって再現率が高くなるように制御できることがわかる．

5)　これは、Kiyono らの PRETLARGE+SSE モデル [7] に相当する．

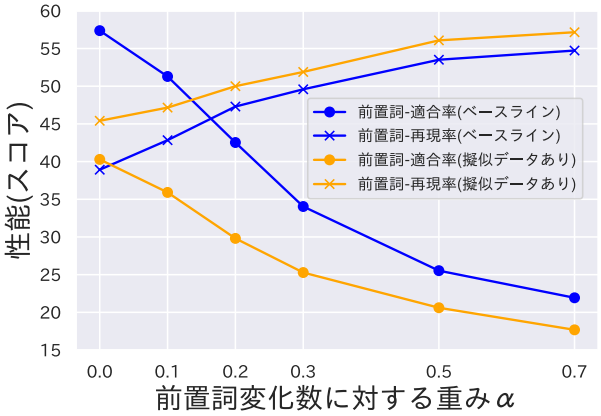


図 2　ベースラインモデル、$p = 1.0$ の擬似データモデルにおける前置詞編集数によるリランキングの効果

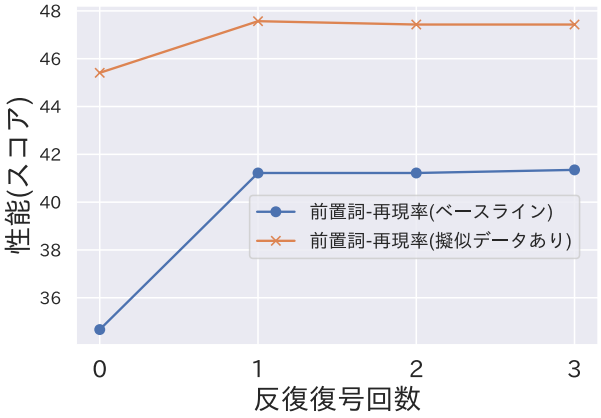


図 3　ベースラインモデル、$p = 1.0$ の擬似データモデルにおける反復復号化の効果

一方で、重みを大きくするとビームサーチのスコアの影響が小さくなり、適合率は下がる．

3.4　実験 3：反復復号化

ベースラインモデルと $p = 1.0$ の擬似データモデルで反復復号化をそれぞれ 3 回行った際の性能変化を調べた結果を図 3 に示す．図から、反復復号化は 1 回目は効果があるものの、2 回目以降はほぼ効果がないことが分かる．これは先行研究 [12, 14] と一致する結果である．

3.5　実験 4：各手法の組み合わせ

手法 3 つを全て組み合わせて実験をおこなうことで、本研究の枠組みで達成可能な再現率の最高値を調べた．また、$p = 1.0$ の疑似データモデルを用いて、前置詞再現率のオラクル値を計算した．これは、モデルの出力する候補 20 文のうち、前置詞の再現率が最も高くなるような文を選ぶことによって

表 3 各実験を組み合わせた際の前置詞誤りに対する性能．太字は最も高い再現率のスコアを表す．

手法	BEA-valid		CoNLL-2014	
	適合率	再現率	適合率	再現率
ベースライン	57.37	38.92	71.36	52.01
擬似データ ($p = 1.0$)	40.37	44.87	58.71	56.16
リランキング ($\alpha = 0.5$)	25.53	53.51	37.28	68.87
反復復号化 (1 回)	55.86	41.22	66.67	53.24
擬似データ ($p = 1.0$)+リランキング ($\alpha = 0.5$)	20.61	56.08	32.76	**70.81**
擬似データ ($p = 1.0$)+リランキング ($\alpha = 0.5$)+反復復号化 (1 回)	23.01	**56.35**	33.88	69.23
疑似データ ($p = 1.0$) から計算したオラクル値	49.57	62.97	-	-

表 4 人手評価のスコアに基づくモデル出力文と参照文の比較

人手評価のスコア	数	割合 (%)
参照文>モデル出力文	16	59.2
モデル出力文>参照文	5	18.6
同点	6	22.2

計算した．

その結果を表 3 に示す．表 3 から、それぞれの手法が相補的に再現率向上に寄与することがわかる．特に、全ての手法を組み合わせることで、BEA-valid における前置詞誤りの再現率はベースラインの 38.92 から 56.35 まで向上した．一方で、再現率のオラクル値（62.97）とは 6 ポイント以上の差があることから、リランキングや反復復号化手法の改善が必要であることが示唆される．また、CoNLL-2014 の場合は反復復号化を行わず、擬似データ手法とリランキング手法を組み合わせた場合が最も高い再現率となった．

4 分析

第 3.5 節で再現率のオラクル値（62.97）を求めたが、我々の究極的な目標である再現率 100%のためには 40 ポイント弱の上積みが必要であるとわかる．本節では、この内訳を分析することで、さらなる再現率向上のための手がかりを得ることを試みる．具体的には、モデルが訂正できなかった前置詞誤りのうち、真にモデルが訂正するべきだと思われる誤りの割合を人手評価を用いて明らかにする．

モデルの出力した訂正文候補のうち、オラクル値の計算に用いた文をオラクル文と呼ぶ．モデルが BEA-valid の訂正で出力したオラクル文では、前置詞誤り 588 例が正しく訂正されていたが、243 例が未訂正、もしくは正しく訂正できなかった．未訂正、もしくは正しく訂正できなかった誤り 243 例のうちランダムに 30 例を抽出し、2 人の英語母語話者に評価を依頼した．具体的には、訂正前の原文とそれに対応するオラクル文、参照文を示し、それぞれの文における**前置詞の**修正を「訂正が文を悪化させている」「訂正前と変わらない」「完璧ではないが改善されている」「完璧な訂正である」の 4 段階スコア、または「自分ではこの結果を判断できない」という答えで評価してもらった．なお、長文や原文と参照文の編集距離が極端に大きい例は分析の対象から除外した．これは、対象の**前置詞の**訂正の評価にあたって、前置詞以外の訂正が評価者にとってノイズになると考えたからである．

30 例のうち「自分ではこの結果を判断できない」という答えであった例を除いた 27 例において、それぞれの前置詞の編集に対するスコアを比較した結果を表 4 に示す．表 4 から、“不正解” とされていた前置詞の編集の約 4 割は間違いとは言い切れないことがわかった．“不正解” とされていた前置詞の編集の約 40%が参照文と同等以上の評価を得たことを考慮すると、現在の検証セットの信頼性に疑問が生まれる[6)]．

5 おわりに

本研究では、高再現率な誤り訂正システムの実現を目的として、再現率向上に有効だと思われる 3 つの手法を実験した．その結果、前置詞の再現率についてベースラインから 17 ポイント強の性能向上を達成したが、オラクル値との比較から、リランキング等の手法に改善が必要であるとわかった．また、モデルが訂正できなかった誤りの人手分析から、不正解だとされている訂正が必ずしも間違いではない可能性が示唆された．今後は、前置詞以外を対象としたシステムの構築のほか、クラウドソーシングを用いたより大規模な人手評価に取り組みたい．

6) モデルの出力文の評価値が、参照文の評価値と同じまたはそれ以上であった例を付録 C に示した．

参考文献

[1]Christopher Bryant, Mariano Felice, and Ted Briscoe. Automatic Annotation and Evaluation of Error Types for Grammatical Error Correction. In *ACL*, pp. 793–805, 2017.

[2]Daniel Dahlmeier and Hwee Tou Ng. Better Evaluation for Grammatical Error Correction. In *NAACL*, pp. 568–572, 2012.

[3]Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. The BEA-2019 Shared Task on Grammatical Error Correction. In *BEA*, pp. 52–75, 2019.

[4]Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. Neural Grammatical Error Correction Systems with Unsupervised Pre-training on Synthetic Data. In *BEA*, pp. 252–263, 2019.

[5]Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *NIPS*, pp. 5998–6008, 2017.

[6]Alla Rozovskaya, Mark Sammons, and Dan Roth. The UI System in the HOO 2012 Shared Task on Error Correction. In *BEA*, pp. 272–280, 2012.

[7]Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. An Empirical Study of Incorporating Pseudo Data into Grammatical Error Correction. In *EMNLP-IJCNLP*, pp. 1236–1242, 2019.

[8]Yujin Takahashi, Satoru Katsumata, and Mamoru Komachi. Grammatical Error Correction Using Pseudo Learner Corpus Considering Learner's Error Tendency. In *ACL*, pp. 27–32, 2020.

[9]Christopher Bryant and Ted Briscoe. Language Model Based Grammatical Error Correction without Annotated Training Data. In *BEA*, pp. 247–253, 2018.

[10]Alla Rozovskaya and Dan Roth. Generating Confusion Sets for Context-Sensitive Error Correction. In *EMNLP*, pp. 961–970, 2010.

[11]Shamil Chollampatt, Weiqi Wang, and Hwee Tou Ng. Cross-sentence grammatical error correction. In *ACL*, pp. 435–445, 2019.

[12]Jared Lichtarge, Chris Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. Corpora Generation for Grammatical Error Correction. In *NAACL*, pp. 3291–3301, 2019.

[13]Tao Ge, Furu Wei, and Ming Zhou. Fluency boost learning and inference for neural grammatical error correction. In *ACL*, pp. 1055–1065, 2018.

[14]浅野広樹, 鈴木潤, 水本智也, 乾健太郎. 文法誤り訂正における反復訂正の効果検証. 言語処理学会第 25 回年次大会予稿集, pp. 578–581, 2019.

[15]Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. The CoNLL-2014 Shared Task on Grammatical Error Correction. In *CoNLL*, pp. 1–14, 2014.

[16]Noam Shazeer and Mitchell Stern. Adafactor: Adaptive Learning Rates with Sublinear Memory Cost. In *ICML*, pp. 4596–4604, 2018.

A　擬似データ

前置詞誤り生成の際に今回使用した混乱セットを表 5 に示す．

表 5　前置詞誤りの擬似データを生成する際に使用する前置詞混乱セット．元コーパス中の for という単語を置換する場合、to・of・in・on のいずれかに置換される．

元コーパス中の前置詞	前置詞混乱セット
for	to of in on
in	on at of to
on	in at to of
of	in for to about
at	in to of on for
with	by in to of for
about	of for in on
from	of in by at for
to	for in with of
by	with from of in

表 5 を用いて実際に生成された擬似データの例を表 6 に示す．

B　リランキング

リランキングの例を表 7 に示す．

C　人手評価の例

モデルの出力文の評価値が、参照文の評価値と同じまたはそれ以上であった例を表 8 に示す．

D　謝辞

本研究を行うにあたり、多くの方々のご協力、ご指導を賜りました．様々な有益なアドバイス、議論をしてくださった東北大学乾・鈴木研究室の皆様へ感謝いたします．特に、英文の人手評価に協力してくださった Paul Reisert さん、Keshav Singh さんへ、記して深く感謝いたします．

表 6　Gigaword コーパスから生成された前置詞誤りを含む擬似データの例

原文	疑似誤りを含む文
It was a different position **for** us .	It was a different **of** position **of** us .
The euro stood **at** 1.1988 dollars .	The euro stood 1.1988 dollars .

表 7　重み $\alpha = 0.3$、候補文 3 文におけるリランキングの例 この場合、リランキング前は 2 位だった文が with を about に編集しているためスコアが追加され、リランキングによって 1 位の文となる．

原文	
People have also become concerned with nature .	
スコア	訂正文
リランキング前	
-0.1759156	People have also become concerned with nature .
-0.4630002	People have also become concerned about nature .
-0.6949797	People have also become concerned with nature ,
リランキング後	
-0.1630002	People have also become concerned about nature .
-0.1759156	People have also become concerned with nature .
-0.6949797	People have also become concerned with nature ,

表 8　人手評価でモデル出力文の評価値が参照文の評価値と同程度以上であった文の例

文の種類	入/出力文
原文	Furthermore , elevation of the water level on the river helps people to realize the possibility of expanding the area of fertile lands .
モデル出力文	Furthermore , the elevation of the water level on the river helps people to realize the possibility of expanding the area of fertile lands .
参照文	Furthermore , the rise in the water level on the river helps people to realize the possibility of expanding the area of fertile lands .