文法誤り訂正モデルの横断評価

三田 雅人^{†,††}・水本 智也^{†††,†}・金子 正弘^{††††,†}・
永田 亮^{†††††,†}・乾 健太郎^{†††}

文法誤り訂正の既存研究の多くはこれまで主に CoNLL-2014 評価データセットを用いた単一コーパス上で文法誤り訂正モデルを評価してきた。しかし、書き手の習熟度やエッセイのトピックなど様々なバリエーションのある入力文が想定される文法誤り訂正タスクにおいて、タスクの難易度は各条件下によって異なるため、単一コーパスによる評価では不十分であると考えられる。そこで本研究では、文法誤り訂正の評価の方法論として、単一コーパス評価は不十分であるという仮説に基づきコーパス横断評価の必要性について調査を行う。具体的には、4 種類の手法 (LSTM, CNN, Transformer, SMT) を 6 種類のコーパス (CoNLL-2014, CoNLL-2013, FCE, JFLEG, KJ, BEA-2019) で評価し、各コーパス間でモデル順位にばらつきが生じるかについて検証を行った。評価実験の結果、モデル順位は各コーパスによって大きく変動したため、既存の単一コーパス評価では不十分であることがわかった。また、横断評価はメタ評価方法としてだけでなく、実応用を見据えた場合においても有用であると考えられる。そこで、横断評価の有用性のケーススタディとして、文法誤り訂正の入力に想定される代表的な条件の一つである、書き手の習熟度を評価セグメントとした場合の横断評価について調査を行った。その結果、書き手の習熟度が初級レベルと上級レベル間ではモデルの性能評価に関して大きな乖離があることがわかった。

キーワード：文法誤り訂正、評価尺度、学習支援、深層学習

Cross-Sectional Evaluation of Grammatical Error Correction Models

MASATO MITA^{†,††}, TOMOYA MIZUMOTO^{†††,†}, MASAHIRO KANEKO^{††††,†},
RYO NAGATA^{†††††,†} and KENTARO INUI^{†††}

Grammatical error correction (GEC) systems have been typically evaluated on the single corpus: the CoNLL-2014 benchmark. However, evaluation remains incomplete because the task difficulty should vary depending on test corpus properties including

[†] 理化学研究所, RIKEN

^{††} 東北大学, Tohoku University

^{†††} フューチャー株式会社, Future Corporation

^{††††} 東京都立大学, Tokyo Metropolitan University

^{†††††} 甲南大学, Konan University

the proficiency levels of the writers and essay topics. This study explores the necessity of cross-corpora evaluation for GEC systems based on the hypothesis that a single corpus evaluation is insufficient for evaluating GEC systems. Specifically, we evaluated performance of four GEC models (based on LSTM, CNN, Transformer and SMT) against six corpora (CoNLL-2013, CoNLL-2014, FCE, JFLEG, KJ and BEA-2019). Evaluation results revealed that model rankings vary considerably depending on the corpora, indicating that single-corpus evaluation is insufficient for GEC models. Moreover, cross-sectional evaluation is useful not only as a meta-evaluation method but also for practical applications. As a case study of the usefulness of cross-sectional evaluation, we investigated the cross-sectional evaluation of one of the typical conditions for input of grammatical error correction: writer's proficiency level as a unit of evaluation. The results showed that there was a large divergence in the evaluation between the beginner-intermediate and advanced levels of writer's proficiency.

Key Words: *Grammatical Error Correction, Evaluation Metric, Computer Assisted Language Learning, Deep Learning*

1 はじめに

文法誤り訂正 (Grammatical Error Correction, GEC) は, テキスト中に含まれる様々な種類の誤りを自動的に訂正するタスクである. 文法誤り訂正技術は作文支援や言語学習者支援など幅広い応用が期待されることから, 自然言語処理における主要な応用タスクの一つとなっている. 特に, 英語を対象とした文法誤り訂正タスクでは, CoNLL-2014 データセット (以下, CoNLL-2014) (Ng et al. 2014) を用いて文法誤り訂正モデルの評価を行うことで盛んに研究開発が進んできた (Junczys-Dowmunt and Grundkiewicz 2016; Chollampatt and Ng 2018; Lichtarge et al. 2019; Zhao et al. 2019; Kiyono et al. 2019).

一方で, 文法誤り訂正モデルの性能評価に関してはタスクの性質上不十分な可能性が高い. 文法誤り訂正タスクの入力には書き手の習熟度・母語・エッセイトピックなどの条件によって様々なバリエーションのある文が想定され, タスクの難易度は各条件下によって異なる. しかし, 既存研究の多くは1つまたは2つのコーパスを用いた限定的な条件でのみ評価を行う傾向がある (Mizumoto and Matsumoto 2016; Ji et al. 2017; Junczys-Dowmunt et al. 2018; Lichtarge et al. 2019). 実際, Grammarly¹や Ginger²など文法誤り訂正技術の実サービスでは, 特定の条件ではなく任意の条件を入力として想定する場合が多い. 仮に, 難易度に依存して異なるコーパス間でモデルの評価にばらつきが生じた場合, 単一コーパスによる評価ではある特定の条件でのみ成立する限定的な議論しか行うことができないため, 評価の方法論として不十分である.

¹ <https://www.grammarly.com/>

² <http://www.getginger.jp/>

そこで本研究では、文法誤り訂正の評価の方法論として、既存の単一コーパスによる評価は不十分であるという仮説に基づき、コーパス横断評価の必要性について調査する。ここでコーパス横断評価とは、単一の評価尺度と複数のコーパスを用いて、モデルを多角的に評価することと本研究では定義する。上記仮説を検証するにあたり、4種類の手法 (LSTM, CNN, Transformer, SMT) を6種類のコーパス (CoNLL-2014, CoNLL-2013, FCE, JFLEG, KJ, BEA-2019) を用いて評価実験を行った結果、コーパス毎にモデル順位が大幅に変動した。そのため、タスクの性質上、文法誤り訂正モデルは複数のコーパスを用いて横断的に評価を行うことが重要である。また、横断評価は実応用を見据えた場合においても有用であると考えられる。そこで、横断評価の有用性を検証するために、文法誤り訂正の入力に想定される代表的な条件の一つである書き手の習熟度を評価セグメントとした場合の横断評価を行った。その結果、書き手の習熟度が初中級レベルと上級レベル間ではモデルの性能評価に関して大きな乖離があることがわかった。本研究の貢献は次の3点である。

- 初めて文法誤り訂正モデルのコーパス横断評価の必要性について調査した。
- 既存の単一コーパス評価では文法誤り訂正モデルの評価として信頼性に欠ける可能性があることを示した。
- 書き手の習熟度が初中級レベルと上級レベル間ではモデルの性能評価に関して大きな乖離があることを示した。

2 関連研究

表1に、既存研究が使用した評価データセットおよび評価尺度の概要を示す。1節でも述べた通り、既存手法の多くは単一コーパスのみを用いて評価を行っている (Susanto et al. 2014; Felice and Briscoe 2015; Junczys-Dowmunt and Grundkiewicz 2016; Ji et al. 2017)。最近では、文法誤り訂正モデルの文法性および流暢性を評価するために提案された JFLEG データセット (以下, JFLEG) (Napoles et al. 2017) や BEA-2019 Shared Task で提供された BEA-2019 データセット (以下, BEA-2019)³の登場により、CoNLL-2014に加えて、JFLEG や BEA-2019 を評価データセットとして併用する研究も増えてきている (Junczys-Dowmunt et al. 2018; Grundkiewicz and Junczys-Dowmunt 2018; Lichtarge et al. 2019; Zhao et al. 2019; Kiyono et al. 2019; Kaneko et al. 2020)。しかし、表1から、それらの多くは評価データ毎に異なる評価尺度を独立的に用いて評価を行っているため、コーパス横断的にモデルの評価は行えていないことがわかる。具体的には、CoNLL-2014では M^2 (Dahlmeier and Ng 2012) を用いて算出した $F_{0.5}$ で評価を行い、JFLEGではGLEU (Napoles et al. 2015, 2016a) を用いて評価を行う傾向にある (Junczys-Dowmunt et al.

³ データセットの詳細を付録Aに示す。

既存研究	CoNLL-2014			JFLEG			BEA-2019		
	M^2	GLEU	ER.	M^2	GLEU	ER.	M^2	GLEU	ER.
(Susanto et al. 2014)	✓	—	—	—	—	—	—	—	—
(Felice and Briscoe 2015)	✓	—	—	—	—	—	—	—	—
(Bryant and Ng 2015)	✓	—	—	—	—	—	—	—	—
(Rozovskaya and Roth 2016)	✓	—	—	—	—	—	—	—	—
(Junczys-Dowmunt and Grundkiewicz 2016)	✓	—	—	—	—	—	—	—	—
(Ji et al. 2017)	✓	—	—	—	—	—	—	—	—
(Yannakoudakis et al. 2017)	✓	✓	—	✓	✓	—	—	—	—
(Junczys-Dowmunt et al. 2018)	✓	—	—	✓	—	—	—	—	—
(Grundkiewicz and Junczys-Dowmunt 2018)	✓	—	—	✓	—	—	—	—	—
(Lichtarge et al. 2019)	✓	—	—	✓	—	—	—	—	—
(Kiyono et al. 2019)	✓	—	✓	—	✓	—	—	—	✓
(Kaneko et al. 2020)	✓	—	—	—	✓	—	—	—	✓

表 1 既存研究が使用した評価データセットおよび評価尺度. ER. は ERRANT を示している.

2018; Kiyono et al. 2019; Kaneko et al. 2020). 限定的ではあるが横断評価を行っている数少ない既存研究として, (Yannakoudakis et al. 2017) が存在する. しかし, そこでは横断評価の必要性および評価の方法論に関する議論はされていないため, その後から現在に至るまで未だに単一評価が主流の傾向にある. コーパス横断評価の目的は, 入力の変えた場合における各モデルの挙動を評価することであるが, 評価セット毎に独立した評価尺度を用いると, モデルの挙動差が入力に起因するものか, 評価尺度に起因するものかの判別が困難になる. このように, 文法誤り訂正の研究コミュニティ全体として, 利用可能なデータが増えるにしたがって複数のコーパスを使ってモデルの性能評価を行う試みは増えてはいるが, その一方で, 実質的には単一的な評価に留まっている傾向にある. また, 入力に様々なバリエーションのある文を想定するタスクの性質自体がモデルの性能評価にどのくらい影響を与えるかについても明らかにされていない.

他分野をみると, 例えば構文解析分野では以前は Penn Treebank (Marcus et al. 1993) 上における解析精度を向上させることに重点を置いていた. その後, Penn Treebank に加えて, Ontonotes (Hovy et al. 2009) や Google Web Treebank (Petrov and McDonald 2012) を含む複数のコーパスを用いて評価を行うようになり, 評価手法の頑健性が大幅に改善されている. 我々は文法誤り訂正分野においても構文解析分野と同様の状況が起きていると考える. すなわち, 現在の文法誤り訂正の評価は CoNLL-2014 に依存しているため, 他のコーパスを用いて横断的な評価を行うことで, 評価手法の信頼性が向上することが期待される.

3 調査方針

本研究の目的は、文法誤り訂正の評価の方法論として、単一コーパス評価は不十分であるという仮説に基づき、コーパス横断評価の必要性について調査することである。調査方針として、各コーパス間でモデルの性能順位にばらつきが起こるかを検証する。なぜなら、各コーパスによってモデル順位が大きく変動した場合、単一コーパス上での評価はある特定の条件下でのみ成立することとなり、多様なバリエーションのある文が入力に想定されるタスクの性質上、評価の方法論として不十分であるからである。ここで、理想的には入力に想定されるあらゆる条件を網羅することが好ましいが、リソースの制限があるため全ての条件を網羅するのは困難である。そこで本研究では、可能な限り多くのコーパスを用意することで、代表的な条件である「書き手の習熟度」、「エッセイトピック」、「書き手の母語」について多様性の担保を図る。

3.1 評価用コーパス

コーパス横断評価の必要性を検証するにあたり、本研究では既存研究の多くで使用されている CoNLL-2014 に加えて、CoNLL-2013, FCE, JFLEG, KJ, BEA-2019 の合計 6 つのコーパスを用いる。以下に、評価実験に用いた各評価用コーパスの詳細を説明する。

CoNLL-2014 (Ng et al. 2014): CoNLL-2014 Shared Task の公式データセットであり、既存研究で最も一般的に使用されている評価用データセットである。シンガポール国立大学の学生が書いたエッセイを基に作られている。シンガポールは英語が公用語のため、書き手の習熟度は比較的高いとされている。なお、シンガポールは多民族国家であるため、母語は一概に定義できない。また、エッセイのトピック数は 2 つのみである。

CoNLL-2013 (Ng et al. 2013): CoNLL-2013 Shared Task の公式データセットである。CoNLL-2014 と同様に書き手の習熟度は比較的高い。また、エッセイトピック数は 2 つのみである。

Cambridge ESOL First Certificate in English (FCE) (Yannakoudakis et al. 2011): ケンブリッジ英検の試験答案を基に作られた学習者コーパスである。様々な国籍の英語学習者が書いたエッセイであるため、学習者の習熟度や母語は多様性がある。また、ケンブリッジ英検の FCE 試験は中級から上級レベルと比較的幅広い受験者を対象としていることから、書き手の習熟度も中・上級レベルであると考えられる。なお、エッセイのトピック数は 10 である。

JHU FLuency-Extended GUG corpus (JFLEG) (Napoles et al. 2017): 様々な国籍の非母語話者が書いた英語答案を基に作られた学習者コーパスである。書き手の習熟度、母語、エッセイトピックともに多様性を担保するよう設計されている。また、文法的になるだけでなく流暢な文になるように訂正されていることも本コーパスの特徴の一つである。そのため、最近の研究では、モデルがどれだけ流暢性のある訂正ができているかを評価するためのベンチマークデータセットとして使用されることが多い。

Konan-JIEM learner corpus (KJ) (Nagata et al. 2011): 甲南大学と教育測定研究所 (JIEM) が共同で収集, アノテーションを行ったコーパスであり, 日本人英語学習者の 233 エッセイから成る. 書き手の習熟度に関しては, 公開情報がないため定義することは難しい. エッセイのトピック数は 10 である. なお, 他のコーパスと異なり, 綴り誤りの情報と文法誤りの情報が別でアノテーションされていることも特徴の一つである.

BEA-2019 Shared Task Official Development Set (BEA-2019): BEA-2019 Shared Task で公式に提供されたデータセットである. 訓練・開発・評価用のデータセットが提供されており⁴, 開発・評価用データセットは W&I+LOCNESS (Yannakoudakis et al. 2018; Granger 1998) のサブセットが使用されている. W&I+LOCNESS コーパスには, メタ情報として, 言語能力を評価する国際指標である CEFR に対応する書き手の習熟度の情報が付与されているのが特徴の一つである. なお, BEA-2019 Shared Task では CodaLab⁵ 上にて評価が行われていたため, 公式評価用データセットの参照訂正文は公開されていない. そのため, 本評価実験では公式開発用データセットを評価データセットとして使用する.

各評価用コーパスの特徴を整理したものを表 2 に示す.

なお, 単語編集率は以下の式により算出した.

$$\text{単語編集率} = \frac{\sum_{i=1}^N d(\mathbf{X}_i, \mathbf{Y}_i)}{\sum_{i=1}^N \mathbf{T}_i}$$

ここで, $\mathbf{T}_i$ は各入力文における総単語数, $d(\mathbf{X}_i, \mathbf{Y}_i)$ は入力文 $\mathbf{X}_i$ と出力文 $\mathbf{Y}_i$ 間の単語に基づくレーベンシュタイン距離をそれぞれ示している. 各コーパスを概観すると, (1) 現在主流である CoNLL-2014 は他のコーパスと比較して書き手の習熟度・母語・トピックに偏りがあること, (2) 一方, 本評価実験で選定した 6 つのコーパスを用いることで習熟度・母語・トピックの 3 つの条件において一定の多様性を確保できていることがわかる.

なお, 綴り誤りは厳密には文法誤りには含まないが,

コーパス	文数	参照訂正文数	単語編集率 (%)	習熟度	トピック数	母語
CoNLL-2014	1,312	2	12.35	上級	2	N/A
CoNLL-2013	1,381	1	14.85	上級	2	N/A
FCE	32,222	1	12.01	中・上級	10	多様
JFLEG	747	4	20.86	多様	多様	多様
KJ	3,081	1	16.01	N/A	10	日本語
BEA-2019	4,384	1	10.61	多様	50	多様

表 2 各評価コーパスの概要. 公開情報がなく判断が難しいものは N/A としている.

⁴ <https://www.cl.cam.ac.uk/research/nl/bea2019st/>

⁵ <https://competitions.codalab.org/competitions/20228>

- 既存データセットの多くはそれらを区別せずにまとめてアノテーションしている
- また、そのようなデータを用いて訓練した機械翻訳モデルに基づく手法も、それらを区別して訂正できない

という理由から、本評価実験においても従来の慣習に従い綴り誤りを含めて評価を行う。そのため、綴り誤りと文法誤りが別でアノテーションされている KJ に関しては、統合処理したものを評価実験に使用する。

3.2 訓練用コーパス

NAIST Lang-8 Learner Corpora (Lang-8) (Mizumoto et al. 2011; Tajiri et al. 2012): 奈良先端科学技術大学院大学の研究チームが開発した、語学学習のための SNS Lang-8 の添削ログからクローリングして作られたデータである。データサイズはおよそ 200 万文対で、一般利用可能な文法誤り訂正用のデータとしては最大規模のものである。書き手である学習者の母語としては日本語が多いが、他の言語を母語とする学習者の英文も数多く含まれている。

NUS Corpus of Learner English (NUCLE) (Dahlmeier et al. 2013): シンガポール国立大学の研究チームが開発したデータセットであり、シンガポール国立大学の学生が書いたエッセイを基に作られている。データサイズはおよそ 6 万文対で、Lang-8 には劣るが、専門家によってアノテーションされているため、訂正の品質はよいとされている。エラータグも人手で付与されており、CoNLL-2013, CoNLL-2014 の両 Shared Task の公式訓練データセットとして採用された。

各訓練用コーパスの特徴を整理したものを表 3 に示す。NUCLE は、CoNLL-2014 および CoNLL-2013 と書き手の属性が同じことから、習熟度および母語の観点で非常にドメインが似ているといえる。また、Lang-8 は日本語を母語とする学習者の書いた英文を多く含んでいるため、書き手の母語という観点で KJ とドメインが似ているといえる。

3.3 訂正モデル

文法誤り訂正モデルの選定については、(1) 現在主流であり、かつ複数の誤りを訂正可能な機械翻訳モデルに基づくモデルであること、(2) CoNLL-2014 上において各モデルの性能は同程度であること、の 2 つの要件を設けた。1 つ目の要件に関しては、本研究は、現在の文法性誤り訂

コーパス	文数	参照訂正文数	単語編集率 (%)	習熟度	トピック数	母語
NUCLE	57,151	1	6.13	上級	多様	N/A
Lang-8	2,167,655	1	31.58	多様	多様	多様

表 3 各訓練用コーパスの概要。公開情報がなく判断が難しいものは N/A としている。

正の評価の方法論に対する問題提起を行うことを目的としているため、古典的なモデルではなく、現在主流、かつ一般的な設定のモデルを用意することが望ましいからである。2つ目の要件に関しては、コーパス横断評価の必要性に関する着想として、入力に様々なバリエーションのある文を想定するタスクの性質がモデルの評価に対して無視できない影響を与えるのではないかと、という考えがあるからである。基準となる現在主流の CoNLL-2014 上において各モデルの性能差に大差がある場合、実験結果に対する解釈を行う上でタスクの性質に起因した影響なのかどうかの判断が困難になるため、2つ目の要件を設けた。

以下に、評価実験に用いた各モデルの詳細について説明する。

LSTM: エンコーダは双方向 LSTM で構成され、デコーダと注意機構は (Luong et al. 2015) の seq2seq モデルに準拠している。なお、単語埋め込みの事前学習には English Wikipedia を用いており、ハイパーパラメータは (Chollampatt and Ng 2018) と同じものを使用した。

CNN: (Chollampatt and Ng 2018) の実装⁶を用いた。なお、単語埋め込みの事前学習には English Wikipedia を用いており、ハイパーパラメータは (Chollampatt and Ng 2018) と同じものを使用した。

Transformer: Transformer は (Vaswani et al. 2017) によって提案された自己注意型モデルである。本研究では、seq2seq モデルのツール `fairseq` の実装⁷を用いた。なお、単語埋め込みの事前学習には English Wikipedia を用いており、ハイパーパラメータは (Vaswani et al. 2017) と同じものを使用した。

SMT: (Junczys-Dowmunt and Grundkiewicz 2016) によって提案されたモデルを採用した。モデル訓練時に評価尺度である $F_{0.5}$ が高くなるようにパラメータのチューニングを行うことで大幅な性能向上が報告されている (Junczys-Dowmunt and Grundkiewicz 2016)。具体的には、(Junczys-Dowmunt and Grundkiewicz 2016) の実装⁸を用いた。なお、(Junczys-Dowmunt and Grundkiewicz 2016) は、言語モデルの訓練に Common Crawl (Buck et al. 2014) と English Wikipedia を用いていたが、本評価実験では NMT に基づく各モデルと使用するデータの設定を可能な限り等しくするため、言語モデルの訓練に English Wikipedia のみを、翻訳モデルの訓練に NUCLE および Lang-8 を使用した。

各モデル共通の実験設定は次の通りである。訓練データとして、Lang-8 および NUCLE の2つのパブリックなデータセットを統合したものをを用いた。訓練データセットの前処理および分割は (Chollampatt and Ng 2018) に従い、最終的に開発データセットとして NUCLE のサブセット 5,448 文対、訓練データとして NUCLE の残り 15,974 文対および Lang-8 1,283,529 文対の合計 1,299,503 文対を使用した。

⁶ <https://github.com/nusnlp/mlconvgec2018>

⁷ <https://github.com/pytorch/fairseq>

⁸ <https://github.com/grammatical/baselines-emnlp2016>

3.4 評価尺度

評価尺度には、これまで文法誤り訂正タスクで最もよく利用されている M^2 および GLEU を用いる。

M^2 (Dahlmeier and Ng 2012): M^2 によって算出された $F_{0.5}$ 値が CoNLL-2013, 2014 Shared Task の公式採用されて以降、文法誤り訂正タスクの評価尺度として最も一般的に用いられている (表 1 参照)。 M^2 が登場する以前の文法誤り訂正の評価では、訂正モデルが行った編集操作がどのくらい参照訂正文の編集と一致しているかを F 値で評価していた (Dale and Kilgariff 2011; Dale et al. 2012)。しかし、長いフレーズの編集が必要な場合などに訂正モデルを過小評価してしまうという問題があった。 M^2 は “edit lattice” を用いることにより、モデルが行った編集操作を正解と最大一致するように同定することでこの問題を解決する。

ここで、 M^2 を計算するツールとして、 M^2 Scorer (Dahlmeier and Ng 2012)⁹ と ERRANT (Bryant et al. 2017; Felice et al. 2016)¹⁰ の 2 つが存在する。 M^2 Scorer では、誤りタイプ別の評価においては recall しか算出できず、エラー分析が難しいという問題があった。一方で、ERRANT では入力文とモデル出力の平行データから、モデル出力に対して誤りのタイプを約 96% という高い精度で自動付与することでこの問題を解決する。これにより、誤りタイプ別の訂正性能を評価することができ、モデルがどの誤りタイプに強い・弱いかについてエラー分析可能となる。また、どこからどこまで 1 つの編集と見なすかという “edit extraction” の方法が M^2 Scorer と ERRANT とでは異なるため、算出される $F_{0.5}$ 値も異なる。本研究では、評価ツールの違いが横断評価にどのように影響するかについて分析するために、 M^2 Scorer を用いて算出した $F_{0.5}$ 値 (以降、 M^2 と表記する) と ERRANT を用いて算出した $F_{0.5}$ 値 (以降、ERRANT と表記する) の両方を用いる。

GLEU (Napoles et al. 2015, 2016a): 機械翻訳タスクで最も一般的な評価尺度である BLEU (Papineni et al. 2002) を文法誤り訂正用に改良した評価尺度である。

GLEU は、モデル出力文 (H) と参照訂正文 (R) で一致する n -gram 数から、入力文 (S) に現れるが参照文に現れない n -gram 数を減算することによって計算される。形式的には次式で表される。

$$GLEU+ = BP \cdot \exp\left(\sum_{n=1}^4 \frac{1}{n} \log(p_n')\right) \quad (1)$$

$$p_n' = \frac{N(H, R) - [N(H, S) - N(H, S, R)]}{N(H)} \quad (2)$$

ただし、 $N(A, B, C, \dots)$ は集合間での n -gram 重なり数を表し、BP は BLEU と同様の brave

⁹ <https://github.com/nusnlp/m2scorer>

¹⁰ <https://github.com/chrisjbryant/errant>

penalty¹¹を表す。M²を含む、これまでに提案された文法誤り訂正の評価尺度の中では最も人手評価との相関が高いことが知られている (Napoles et al. 2016b)。

実験結果として報告するのは4つのランダムシードを用いて訓練したモデル性能の平均値である

4 コーパス横断評価

図1にM²を用いて算出したF_{0.5}に基づき順位付けされた各モデルの性能を示す。この図から、モデルの性能順位が各コーパスによって大きく変動していることが確認できる。これまで多くの研究が評価に用いていたCoNLL-2014では、Transformerが48.62ポイントと最も高い性能を出している。一方で、FCEやKJでは順位が全体的に大きく変動しており、Transformerは3位に転じている。順位が逆転した根本原因については、書き手の習熟度や母語、エッセイトピックなど様々な要因が複合的に関係していると予想されるため一概に言えない。しかし本評価実験の結果から、CoNLL-2014上の性能値に基づく議論は特定の条件下でしか成立しない可能性があり、CoNLL-2014での結果を不用意に一般化してしまうと誤った結論を導く危険性があることが示唆される。さらに、図1から、モデルの性能はコーパスによって大きく異なることが確認できる。例えば、Transformerの性能は、JFLEGでは60.06ポイントであるのに対し、CoNLL-2013では36.20ポイントと大きく下回っている。このように、コーパスによって性能は大きく変動するため、見た目の性能値の解釈には十分に注意する必要がある。実際のモデル出力を表4および表5に示す。Transformerを用いることで正しい訂正ができる場合(表4)もあれば、逆にTransformerでは正しい訂正に失敗し、LSTMやCNNを用いることで正しい訂正ができていた場合(表5)があることがわかる。

次に、図2にGLEUに基づき順位付けされた各モデルの性能を示す。GLEUに基づく評価においてもM²を用いたときと同様にコーパスによってモデル順位が変動することが確認できる。一方で、M²を用いたときよりもTransformerが安定的に性能が高い傾向になっていることがわ

図1 M²を用いて算出したF_{0.5}に基づき順位付けされたモデル性能。

¹¹ brave penalty は入力文に対して出力文が短い場合に n-gram 適合率を減点する項である。

入力文	<i>Hence , some *seen it as being considerate in keeping the genetic risk of getting the disease *in confidential .</i>
参照訂正文	<i>Hence , some see it as being considerate in keeping the genetic risk of getting the disease [DEL] confidential .</i>
Transformer (1)	<i>Hence , some see it as being considerate in keeping the genetic risk of getting the disease [DEL] confidential .</i>
LSTM (2)	<i>Hence , some *seen it as being considerate in keeping the genetic risk of getting the disease in *confidentiality .</i>
CNN (3)	<i>Hence , some *seen it as being considerate in keeping the genetic risk of getting the disease in *confidentiality .</i>

表 4 CoNLL-2014 におけるモデル出力例. 括弧の中の数値は CoNLL-2014 におけるモデル順位を示している. また, **[DEL]** は削除操作を示している.

入力文	<i>*In that day , the time I left school was about eleven p.m .</i>
参照訂正文	<i>On that day , the time I left school was about eleven p.m .</i>
LSTM (1)	<i>On that day , the time I left school was about eleven p.m .</i>
CNN (2)	<i>On that day , the time I left school was about eleven p.m .</i>
Transformer (3)	<i>*[DEL] That day , the time I left school was about eleven p.m .</i>

表 5 KJ におけるモデル出力例. 括弧の中の数値は KJ におけるモデル順位を示している. また, **[DEL]** は削除操作を示している.

図 2 GLEU に基づき順位付けされたモデル性能.

かる. GLEU は, 基本的に入力文・モデル出力・参照訂正文の 3 つにおける n-gram 一致率を基に計算されるが, 入力文において変更すべき箇所が変更できていなかった n-gram に対してペナルティを与えている (Napoles et al. 2016a). そのため, Recall が高いモデルを高く評価する傾向にあると解釈ができる.

最後に, 図 3 に ERRANT を用いて算出した $F_{0.5}$ に基づき順位付けされた各モデルの性能を示す. ERRANT に基づき順位付けされたモデル性能をみると, どのコーパスにおいても LSTM が最も性能が高くなっていることが確認できる. 同じ $F_{0.5}$ に基づいた評価にも関わらず, 評

図 3 ERRANT による $F_{0.5}$ に基づき順位付けされたモデル性能.

入力文	Genetic risk refers *more to your chance of inheriting a disorder or disease .
モデル	Genetic risk refers to the possibility of inheriting a disorder or disease .
参照訂正文	Genetic risk refers to your chance of inheriting a disorder or disease .
モデルの修正 (M^2)	[more → DEL (削除)], [your chance → the possibility]
モデルの修正 (ERRANT)	[more → DEL (削除)], [your → the], [chance → possibility]
正解の修正	[more → DEL (削除)]
M^2 の評価	TP/FP/FN: 1/1/0
ERRANT の評価	TP/FP/FN: 1/2/0

表 6 M^2 Scorer と ERRANT の評価方法の違い. 表中の TP/FP/FN はそれぞれ, True positive/False positive/False negative を示している.

価尺度が異なるだけで傾向が大きく異なるのは着目に値する¹². これらの傾向の違いは両評価尺度の計算方法の違いに起因すると考えられる. 表 6 は CoNLL-2014 におけるモデル出力に対する M^2 および ERRANT で評価したときの実例である. ERRANT では, 連続するスパンで訂正が発生した場合, 一単語区切りで評価を行っていることがわかる ([your → the], [chance → possibility]). 一方, M^2 では, 連続するスパンで訂正が発生した場合, 一定のスパン区切りで評価を行っていることがわかる ([your chance → the possibility]). 以上のことから, ERRANT は M^2 と比較して FP が増えやすい傾向にあるといえる. 実際, CoNLL-2014 における Transformer を評価した場合, M^2 による評価では FP は合計 713 個カウントされていたが, ERRANT による評価では合計 900 個カウントされていることを確認した.

このように, 全く同じデータセットおよびモデルを用いたにも関わらず, 評価尺度によってモデルの性能順位の傾向が大幅に異なることがわかる. 言い換えると, どのようなモデルを得たいか, あるいは, モデルの何を評価したいかといった目的意識によって適切な評価尺度を設計することが重要であると言える.

¹² ERRANT を用いれば単一コーパス評価でよいのかについては, 6 節に示すケーススタディと合わせて議論する.

5 分析

5.1 多様性が確保された単一コーパス評価では十分か？

4 節の評価実験により、既存の単一コーパス評価は不十分であることを示したが、多様性が確保されたコーパスを用いた場合においても同様のことがいえるか、という疑問が残る。表 2 に示す通り、FCE・JFLEG・BEA-2019 は多様性が確保されたコーパスであるといえる。ここで、もし多様性が確保された単一コーパス評価で十分であれば、モデルの順位変動は起きないことが期待される。しかし、結果をみると、これらの多様性が確保されたコーパス間で、大幅に順位変動が起きていることがわかる（表 1）。このことは、多様性の確保されたコーパスの設計自体が困難であることを示唆している。

5.2 順位変動はなぜ起きたか？

分析 1. 誤りタイプ別性能評価 (CoNLL-2014)： 図 4 に、ERRANT を用いて CoNLL-2014 における主要な誤りタイプ別のモデル性能 $F_{0.5}$ を示す¹³。コーパス横断評価の結果（図 1）では、CoNLL-2014 は他と比べて Transformer が最も高い性能を達成していた。しかし、図 4 から、モデルによって誤りタイプ別性能が異なることが確認でき、モデル毎に訂正傾向が異なることがわかった。

分析 2. 誤りタイプ別性能評価（コーパス横断）： 次に、CoNLL-2014 だけでなく、他の全てのコーパスにおける誤りタイプ別性能について分析する。表 7 は、各コーパスにおいて誤りタイプ別に最も性能が高かったモデルを示している。この表から、各コーパスによって誤りタイプ別性能にばらつきが起きていることがわかる。図 4 に示した CoNLL-2014 上での分析では、例えば動詞の時制誤りに関しては Transformer が他と比べたときに群を抜いて性能が高かった。一方で、コーパス横断的にみても、JFLEG と KJ ではそれぞれ LSTM、CNN のほうが性能が高いことが確認できる。これは、コーパス毎に書き手の習熟度や母語などの影響によって誤り傾向が異なるからだと考えられる。

4 節で述べた通り、コーパス毎にモデル順位が変動した根本原因については、書き手の習熟度や母語、エッセイトピックなど様々な要因が複合的に関係していると予想されるため一概に言えない。しかし、分析 1 および分析 2 の結果から、モデル順位変動は、訂正傾向が異なるモデルと誤り傾向の異なるコーパスの組み合わせで生じるということがわかった。

¹³ CoNLL-2014 における頻度の高い上位 6 つの誤りタイプ。

図 4 CoNLL-2014 における各モデルの誤りタイプ別性能 (F_{0.5}).

	CoNLL-2014	CoNLL-2013	FCE	JFELG	KJ	BEA-2019
限定詞	LSTM	LSTM	LSTM	SMT	CNN	LSTM
前置詞	SMT	Transformer	SMT	Transformer	LSTM	Transformer
句読点	Transformer	Transformer	Transformer	SMT	LSTM	SMT
動詞	LSTM	CNN	SMT	LSTM	LSTM	Transformer
名詞の単複	LSTM	Transformer	CNN	LSTM	CNN	LSTM
動詞の時制	Transformer	Transformer	Transformer	LSTM	CNN	Transformer

表 7 各コーパスにおける誤りタイプ別に最も性能が高かったモデル.

6 横断評価の有用性

本研究では、メタ評価方法論として、文法誤り訂正モデルをコーパス横断的に評価することの重要性を示してきた。本節では、より実応用を考えた場合の横断評価の有用性について議論する。実応用では、対象とするユーザーや、着目したいモデルの評価観点がある程度絞られることが考えられる。例えば、文法誤り訂正システムを学習者のライティング支援として学校などの教育現場で用いることを考えた場合、書き手の母語や習熟度はある程度絞られており、また、それらの情報をあらかじめ知っているという設定が多いといえる。その際、評価セグメントは多様な観点が混ざり合ったコーパスというセグメントである必然性はなく、着目したい観点到に絞った評価セグメントを用いた横断評価を行うことで、より精緻にモデルの挙動を調査することが可能となる。そこで本節では、横断評価の有用性のケーススタディとして、文法誤り訂正モデルの入力として想定される代表的なバリエーションの一つである、**書き手の習熟度**を評価セグメントに横断評価を行うことで、書き手の習熟度を変えた場合モデルはどのような挙動をするかについて調査する。

6.1 習熟度別データセット

習熟度別に横断評価を行うために BEA-2019 を使用する。3.1 節で述べた通り、BEA-2019 はメタ情報として、CEFR に対応した書き手の習熟度情報が付与されている。具体的には、BEA-2019 に含まれる各文には CEFR で定められている 3 段階のレベル（習熟度が低い順に A・B・C レベル）および母語話者 (N) の合計 4 レベルの習熟度情報が付与されており、習熟度別の評価が可能となる¹⁴。BEA-2019 の習熟度別データの特徴を整理したものを表 8 に示す。書き手の習熟度が上がるにつれて、文が長くなり（**平均文長の増加**）、コーパス中に含まれる誤りの個数が減少し（**単語編集率の低下**）、使用する語彙サイズが大きくなる（**異なり語数の増加**）、といった傾向があることがわかる。

BEA-2019	文数	平均文長	単語編集率 (%)	異なり語数
A	1,037	18.0	19.6	2,810
B	1,290	18.4	13.2	3,351
C	1,069	20.1	6.6	3,522
N	988	23.4	4.9	3,824

表 8 BEA-2019 における習熟度別評価データセットの概要。

¹⁴ 厳密には、習熟度レベル間でトピックが異なり、文長に偏りがあるなど、習熟度レベル以外の要素も含まれる。

6.2 習熟度横断評価

表 9 に, M^2 を用いて算出した Precision, Recall, $F_{0.5}$ をそれぞれ示す. コーパス横断評価と同様に, モデルの性能順位が各習熟度によって大きく変動していることが確認できる. 具体的には, 書き手の習熟度が初中級レベル (A, B) に対しては Transformer が他と比べて高い性能を達成している. 一方で, 習熟度が上級レベル (C, N) に対しては SMT が最も高い性能となり, 習熟度間で評価に大きな乖離があることがわかった. また, GLEU を用いて評価した場合も同じような傾向があることが確認できる (表 10). 最近では NMT に基づく手法が主流となっているため (Chollampatt and Ng 2018; Lichtarge et al. 2019; Zhao et al. 2019; Kiyono et al. 2019; Kaneko et al. 2020), SMT に基づくモデルの性能が他の全ての NMT に基づくモデルの性能よりも高い結果となったのは驚くべき結果と言える.

習熟度の高い C および N レベルで SMT の性能が高い結果となった要因は, 単語編集率と誤りの質という観点から説明できる. 書き手の習熟度が高い文の特徴として, 単語編集率が低い, すなわち修正すべき誤りの割合が少ない点がある (表 8 参照). また, C や N レベルにおいて, SMT は他と比べて Precision が高く Recall が低い傾向にあると言える (表 9 参照). したがって, 修正すべき誤りの割合が低い C や N レベルの書き手の文においては, Precision の高いモデルのほうが高く評価されている. 一方で, 書き手の習熟度が低い A および B レベルの文においては, Recall の高いモデルのほうが高く評価される傾向にある.

訂正モデル	A			B			C			N		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$
Transformer	49.9	26.3	42.3	42.5	23.5	36.6	26.1	19.8	24.6	18.7	14.9	17.8
LSTM	57.8	18.3	40.4	50.3	14.4	33.5	32.9	10.6	23.2	27.9	8.9	19.6
CNN	54.2	19.7	40.1	47.0	15.9	33.8	31.4	12.8	24.3	25.7	9.4	19.1
SMT	47.8	12.9	31.1	46.5	12.3	29.9	35.7	12.6	26.1	25.7	10.2	19.7

表 9 BEA-2019 習熟度別データセットにおいて, M^2 を用いて算出した Precision (P), Recall (R), $F_{0.5}$: 太字の値は各評価データセットにおける最高精度を示している.

訂正モデル	A	B	C	N
Transformer	50.97	61.66	76.02	79.42
LSTM	47.66	59.88	77.36	82.08
CNN	47.72	59.51	76.97	81.47
SMT	44.57	58.41	77.70	81.21

表 10 BEA-2019 習熟度別データセットにおける GLEU スコア: 太字の値は各評価データセットにおける最高精度を示している.

訂正モデル	A			B			C			N		
	P	R	F _{0.5}	P	R	F _{0.5}	P	R	F _{0.5}	P	R	F _{0.5}
Transformer	40.1	24.2	35.4	34.4	21.6	30.7	20.4	18.5	20.0	14.8	14.3	14.7
LSTM	52.0	17.7	37.4	43.9	13.6	30.4	29.1	10.5	21.4	24.5	8.7	17.9
CNN	45.4	18.6	35.3	39.7	14.9	29.8	29.1	12.4	21.4	22.1	9.1	17.2
SMT	44.3	12.8	29.7	43.2	12.1	28.5	33.5	12.3	25.0	23.2	10.0	18.4

表 11 BEA-2019 習熟度別データセットにおいて, ERRANT を用いて算出した Precision (P), Recall (R), F_{0.5}: 太字の値は各評価データセットにおける最高精度を示している.

次に, 表 11 に ERRANT を用いて算出した Precision, Recall, F_{0.5} を示す. 表 11 から, (1) モデルの性能順位が各習熟度によって大きく変動し, (2) 書き手の習熟度が高い文においては SMT が高く評価される, といった M² を用いたときと一貫性のある結果となった. また, コーパス横断評価 (図 3) とは異なり, 習熟度横断評価では順位が大きく変動した. そのため, 少なくとも ERRANT を用いれば単一コーパス評価でも十分というわけでないことがわかる.

これまで文法誤り訂正分野で用いられてきた CoNLL-2014 や JFLEG をはじめ, 全ての既存評価コーパスでは, 概ね書き手の習熟度が初中級レベルのうえでモデルの良し悪しを議論してきた. しかし, 上級レベルの書き手を対象に文法誤り訂正モデルを用いて作文支援などを行う場合は, これまで本分野が蓄積してきた知見をそのまま適用するには注意が必要であることが習熟度横断評価を通して明らかになった.

7 おわりに

本研究では, 多様なバリエーションのある入力文を想定した文法誤り訂正の評価の方法論として, 既存の単一コーパス評価では不十分であるという仮説を基に, 横断コーパス評価の必要性について調査した. 具体的には, 4 種類の訂正モデルを 6 種類のコーパス上で横断評価を行い, 各コーパス間でモデルの性能順位にばらつきが生じるかについて調査を行った. その結果, コーパス毎にモデル順位が大幅に変動したため, 既存の単一コーパス評価では不十分であるということがわかった. さらに, 文法誤り訂正の応用を見据えて, 入力に想定される代表的な条件の一つとして書き手の習熟度を評価セグメントとした横断評価をケーススタディ的に行った. その結果, 書き手の習熟度が初中級レベルと上級レベル間でモデルの性能評価に大きな乖離があることがわかった.

文法誤り訂正の方向性として, 目的指向な方向性と汎用指向な方向性の二通り考えられる. 目的指向な方向性は, 特定の習熟度レベルのユーザーのみ対象など, ある特定の条件下で誤りを

徹底的に訂正できればよいという立場である。一方で、汎用指向な方向性とは、多様な文が入力文として入ってきた場合でも頑健に訂正したいといった立場である。目的指向な方向性においては、本研究で提案するような、モデルの入力に想定される条件に応じた評価セグメントにて横断評価を行うことで、より精緻な評価が実現可能となる。しかし、汎用指向な方向性の立場に立った場合、訂正モデルを開発するユーザー視点において、可能であれば単一コーパス上でモデルの頑健性を評価できる方が好ましいと考えられる。そのため、今後の研究として、頑健性のある新しい評価尺度も同時に検討していく必要がある。

謝 辞

本論文の査読にあたり、著者の不十分な記述などに対してご意見・ご指摘をくださった査読者の方々へ感謝します。本論文の内容の一部は、言語処理学会第25回年次大会(三田 他 2019) および The 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Mita et al. 2019) で発表したものです。

参考文献

- Bryant, C., Felice, M., and Briscoe, T. (2017). “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.
- Bryant, C. and Ng, H. T. (2015). “How Far are We from Fully Automatic High Quality Grammatical Error Correction?” In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015)*, pp. 697–707.
- Buck, C., Heafield, K., and van Ooyen, B. (2014). “N-gram Counts and Language Models from the Common Crawl.” In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC’14)*, pp. 3579–3584.
- Chollampatt, S. and Ng, H. T. (2018). “A Multilayer Convolutional Encoder-Decoder Neural Network for Grammatical Error Correction.” In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI 2018)*, pp. 5755–5762.
- Dahlmeier, D. and Ng, H. T. (2012). “Better Evaluation for Grammatical Error Correction.” In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2012)*, pp. 568–572.

- Dahlmeier, D., Ng, H. T., and Wu, S. M. (2013). “Building a Large Annotated Corpus of Learner English: The NUS Corpus of Learner English.” In *Proceedings of the 8th Workshop on Building Educational Applications Using NLP (BEA 2013)*, pp. 22–31.
- Dale, R., Anisimoff, I., and Narroway, G. (2012). “HOO 2012: A Report on the Preposition and Determiner Error Correction Shared Task.” In *Proceedings of the 7th Workshop on Building Educational Applications Using NLP*, pp. 54–62. Association for Computational Linguistics.
- Dale, R. and Kilgariff, A. (2011). “Helping Our Own: The HOO 2011 Pilot Shared Task.” In *Proceedings of the Generation Challenges Session at the 13th European Workshop on Natural Language Generation*, pp. 242–249. Association for Computational Linguistics.
- Felice, M. and Briscoe, T. (2015). “Towards a Standard Evaluation Method for Grammatical Error Detection and Correction.” In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2015)*, pp. 578–587.
- Felice, M., Bryant, C., and Briscoe, T. (2016). “Automatic Extraction of Learner Errors in ESL Sentences Using Linguistically Enhanced Alignments.” In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016): Technical Papers*, pp. 825–835.
- Granger, S. (1998). “The Computer Learner Corpus: A Versatile New Source of Data for SLA Research.” In Granger, S. (Ed.), *Learner English on Computer (LEC 1998)*, pp. 3–18.
- Grundkiewicz, R. and Juncys-Dowmunt, M. (2018). “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 (NAACL 2018)*, pp. 284–290.
- Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2009). “OntoNotes: The 90% Solution.” In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2009), Companion Volume: Tutorial Abstracts*, pp. 57–60.
- Ji, J., Wang, Q., Toutanova, K., Gong, Y., Truong, S., and Gao, J. (2017). “A Nested Attention Neural Hybrid Model for Grammatical Error Correction.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pp. 753–762.
- Juncys-Dowmunt, M. and Grundkiewicz, R. (2016). “Phrase-based Machine Translation is State-of-the-Art for Automatic Grammatical Error Correction.” In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pp. 1546–1556.
- Juncys-Dowmunt, M., Grundkiewicz, R., Guha, S., and Heafield, K. (2018). “Approaching

- Neural Grammatical Error Correction as a Low-Resource Machine Translation Task.” In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2018)*, pp. 595–606.
- Kaneko, M., Mita, M., Kiyono, S., Suzuki, J., and Inui, K. (2020). “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.
- Kiyono, S., Suzuki, J., Mita, M., Mizumoto, T., and Inui, K. (2019). “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.
- Lichtarge, J., Alberti, C., Kumar, S., Shazeer, N., Parmar, N., and Tong, S. (2019). “Corpora Generation for Grammatical Error Correction.” In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2019)*, pp. 3291–3301.
- Luong, T., Pham, H., and Manning, C. D. (2015). “Effective Approaches to Attention-based Neural Machine Translation.” In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pp. 1412–1421.
- Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). “Building a Large Annotated Corpus of English: The Penn Treebank.” *Computational Linguistics*, **19** (2), pp. 313–330.
- 三田雅人, 水本智也, 金子正弘, 永田亮, 乾健太郎 (2019). 文法誤り訂正のコーパス横断評価: 単一コーパス評価で十分か? 言語処理学会第 25 回年次大会発表論文集. [M. Mita et al. (2019). Bunpo Ayamari Teisei no Kopasu Odan Hyoka: Tannitsu Kopasu Hyoka de Jubun ka? Proceedings of the 25th Annual Meeting of the Association for Natural Language Processing.].
- Mita, M., Mizumoto, T., Kaneko, M., Nagata, R., and Inui, K. (2019). “Cross-Corpora Evaluation and Analysis of Grammatical Error Correction Models — Is Single-Corpus Evaluation Enough?” 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. 1309–1314, Minneapolis, Minnesota. Association for Computational Linguistics.
- Mizumoto, T., Komachi, M., Nagata, M., and Matsumoto, Y. (2011). “Mining Revision Log of Language Learning SNS for Automated Japanese Error Correction of Second Language Learners.” In *Proceedings of the 5th International Joint Conference on Natural Language*

- Processing (IJCNLP 2011)*, pp. 147–155.
- Mizumoto, T. and Matsumoto, Y. (2016). “Discriminative Reranking for Grammatical Error Correction with Statistical Machine Translation.” In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016)*, pp. 1133–1138.
- Nagata, R., Whittaker, E., and Sheinman, V. (2011). “Creating a Manually Error-tagged and Shallow-parsed Corpus.” In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011)*, pp. 1210–1219.
- Napoles, C., Sakaguchi, K., Post, M., and Tetreault, J. (2015). “Ground Truth for Grammatical Error Correction Metrics.” In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015)*, pp. 588–593.
- Napoles, C., Sakaguchi, K., Post, M., and Tetreault, J. (2016a). “GLEU Without Tuning.” *arXiv preprint arXiv:1605.02592*.
- Napoles, C., Sakaguchi, K., and Tetreault, J. (2016b). “There’s No Comparison: Reference-less Evaluation Metrics in Grammatical Error Correction.” In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2109–2115. Association for Computational Linguistics.
- Napoles, C., Sakaguchi, K., and Tetreault, J. (2017). “JFLEG: A Fluency Corpus and Benchmark for Grammatical Error Correction.” In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2017)*, pp. 229–234.
- Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H., and Bryant, C. (2014). “The CoNLL-2014 Shared Task on Grammatical Error Correction.” In *Proceedings of the 18th Conference on Computational Natural Language Learning (CoNLL 2014): Shared Task*, pp. 1–14.
- Ng, H. T., Wu, S. M., Wu, Y., Hadiwinoto, C., and Tetreault, J. (2013). “The CoNLL-2013 Shared Task on Grammatical Error Correction.” In *Proceedings of the 17th Conference on Computational Natural Language Learning (CoNLL 2013): Shared Task*, pp. 1–12.
- Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). “BLEU: A Method for Automatic Evaluation of Machine Translation.” In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318.
- Petrov, S. and McDonald, R. (2012). “Overview of the 2012 Shared Task on Parsing the Web.” In *Proceedings of the 1st Workshop on Syntactic Analysis of Non-Canonical Language (SANCL*

2012).

- Rozovskaya, A. and Roth, D. (2016). “Grammatical Error Correction: Machine Translation and Classifiers.” In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pp. 2205–2215.
- Susanto, R. H., Phandi, P., and Ng, H. T. (2014). “System Combination for Grammatical Error Correction.” In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 951–962.
- Tajiri, T., Komachi, M., and Matsumoto, Y. (2012). “Tense and Aspect Error Correction for ESL Learners Using Global Context.” In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012)*, pp. 198–202.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). “Attention Is All You Need.” In *Advances in Neural Information Processing Systems 31 (NIPS 2017)*, pp. 5998–6008.
- Yannakoudakis, H., Andersen, Ø. E., Geranpayeh, A., Briscoe, T., and Nicholls, D. (2018). “Developing an Automated Writing Placement System for ESL Learners.” *Applied Measurement in Education*, pp. 251–267.
- Yannakoudakis, H., Briscoe, T., and Medlock, B. (2011). “A New Dataset and Method for Automatically Grading ESOL Texts.” In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011)*, pp. 180–189.
- Yannakoudakis, H., Rei, M., Andersen, Ø. E., and Yuan, Z. (2017). “Neural Sequence-Labeling Models for Grammatical Error Correction.” In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pp. 2795–2806.
- Zhao, W., Wang, L., Shen, K., Jia, R., and Liu, J. (2019). “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)*, pp. 156–165.

付録

A The BEA-2019 公式データセット

BEA-2019 Shared Task では、参加者に以下のデータセットを公式データセットとして提供した：Lang-8 (Mizumoto et al. 2011; Tajiri et al. 2012), the National University of Singapore Corpus

of Learner English (NUCLE) (Dahlmeier et al. 2013), the First Certificate in English corpus (Yannakoudakis et al. 2011), W&I+LOCNESS (Yannakoudakis et al. 2018; Granger 1998). 公式データセットは <https://www.cl.cam.ac.uk/research/nl/bea2019st/> にて一般利用可能である.

略歴

三田 雅人：理化学研究所革新知能統合研究センター 自然言語理解チーム テクニカルスタッフ. 2016 年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了. 日本マイクロソフト株式会社のエンジニアを経て, 2018 年より現職. 同年, 東北大学情報科学研究科博士後期課程に進学. 文法誤り訂正を中心とした自然言語処理による言語学習/教育支援に関心がある. 言語処理学会, ACL 各会員.

水本 智也：2015 年奈良先端科学技術大学院大学情報科学研究科博士後期課程修了. 博士 (工学). 東北大学情報科学研究科特任助教, 理化学研究所 AIP センター特別研究員を経て 2019 年よりフューチャー株式会社シニアコンサルタント, 理化学研究所 AIP センター客員研究員.

金子 正弘：2016 年北見工業大学工学部システム工学科卒業. 同年, 東京都立大学システムデザイン研究科博士前期課程に進学. 2018 年博士前期課程修了. 同年, 東京都立大学システムデザイン研究科博士後期課程に進学.

永田 亮：甲南大学知能情報学部准教授. 2005 年三重大学大学院博士後期課程修了. 兵庫教育大学助手, 甲南大学知能情報学部講師を経て, 2012 年より現職. 言語のモデル化, 文書の自動添削, 解説文の自動生成などの研究に従事. 電子情報通信学会, 言語処理学会, ACL 各会員.

乾 健太郎：東北大学大学院情報科学研究科教授. 1995 年東京工業大学大学院情報理工学研究科博士課程修了. 同大学助手, 九州工業大学助教授, 奈良先端科学技術大学院大学助教授を経て, 2010 年より現職. 2016 年より理化学研究所 AIP センター自然言語理解チームリーダー兼任. 情報処理学会論文誌編集委員長, 同会自然言語処理研究会主査, 言語処理学会論文誌編集委員長等を歴任, 2020 年より言語処理学会副会長.

(2020 年 8 月 1 日 受付)

(2020 年 11 月 8 日 再受付)

(2020 年 12 月 17 日 採録)