文法誤り訂正の自動評価のための 原文・参照文・訂正文間の N-gram F-score

古山翔太 [1,2]　永田亮 [2,3]　高村大也 [2]　岡崎直観 [1,2]
[1] 東京工業大学　[2] 産業技術総合研究所　[3] 甲南大学
shota.koyama@nlp.c.titech.ac.jp　nagata-nlp2024@ml.hyogo-u.ac.jp
takamura.hiroya@aist.go.jp　okazaki@c.titech.ac.jp

概要

本稿では n-gram の頻度に基づき F 値を計算する文法誤り訂正のための自動評価尺度 GREEN を提案する。GREEN は従来の手法よりも人手評価と高い相関を示す。また、計算量が小さく高速に計算ができる。さらに、人手評価で評価が低いシステムの出力に対して従来手法が高い評価を与える場合でも、GREEN は人手評価と近い評価が可能である。

1　はじめに

文法誤り訂正 (GEC) は、文書中の誤りを訂正するタスクで、文書校正 [1, 2, 3] や言語学習支援 [4, 5, 6] への応用が期待されている。GEC では、原文、参照文、訂正文に対し評価を行う自動評価尺度の導入により人手評価を介さない研究開発が可能となった。

GEC の自動評価尺度には、文間のアライメントに基づき訂正の正誤を数える手法と、n-gram の頻度に基づく手法がある。前者のうち F 値を求める M^2 [7] と ERRANT [8] は、原文から無変更の訂正文よりも誤訂正によって改悪された訂正文に高い評価を与えてしまう [9]。前者のうち正解率を求める I-measure [9] は、改悪された訂正文に負の評価を与えることができるが、人手評価との相関が低い。また、アライメントによる手法は計算量が大きい。後者に分類される GLEU [10] は、計算量は小さいが、人手評価で評価が低いシステムの出力を不当に高く評価することがある。

以上の問題を解決するため、本稿では n-gram の頻度に基づき F 値を計算する新しい GEC 自動評価尺度 GREEN を提案する。GREEN は、原文から参照文、原文から訂正文への訂正で生じた n-gram の出現頻度の差分を比較し、F 値を計算する。GREEN は従来手法よりも人手評価と高い相関を示すことを確認した。また、文間のアライメントを計算しないため M^2 や ERRANT よりも計算量が小さく、高速である。さらに、人手評価で入力より悪化したと評価されるシステムの出力に M^2 や GLEU が不当に高いスコアを与える場合においても、GREEN では人手評価と近い評価が行えることを確認した。

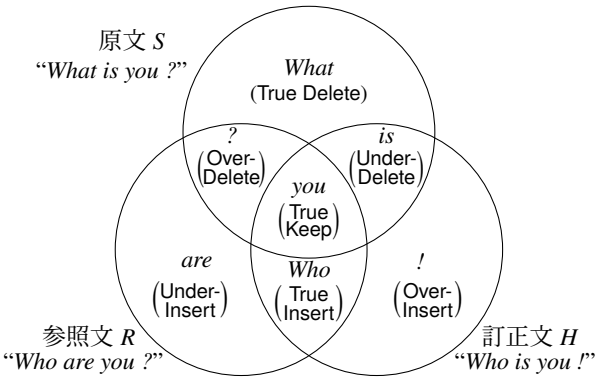

図 1　単語 1-gram の出現の差を表現するベン図

2　提案手法

最初に、1 つの原文に対して 1 つの参照文だけを用いる場合の提案手法 GREEN の説明を行う。複数の参照文を用いる場合は本節の最後で説明する。

GREEN は、原文 S、参照文 R、訂正文 H を n-gram[1] の多重集合として扱う。例えば、文 A = “$a\ a$” は、$\{a, a, a\ a\}$ として扱う。図 1 は、ある S, R, H での各文の単語 1-gram の出現をベン図で表現している[2]。GREEN は、このベン図の各領域に含まれる n-gram を削除や挿入などの異なる種類の訂正とみなし、スコアを計算する。n-gram の多重集合間の訂正は、削除、挿入、保持のいずれかに分類できる。例えば、

1）　要素の重複を許す集合。本文の例は a が重複して現れる。
2）　簡単のため長さ 2 以上の n-gram は省略した。

領域	$S \to R$	$S \to H$	分類	例
True Delete (TD)	削除	削除	TP	What
True Insert (TI)	挿入	挿入		Who
True Keep (TK)	保持	保持		you
Over-Delete (OD)	保持	削除	FP	?
Over-Insert (OI)	なし	挿入		!
Under-Delete (UD)	削除	保持	FN	is
Under-Insert (UI)	挿入	なし		are

表 1 図 1 の各領域の説明。訂正前後で n-gram が挿入、削除、保持されたかで分類できる。訂正前後のどちらにも出現しない場合は「なし」となる。

多重集合 $\{a, c\}$ から $\{b, c\}$ への訂正では、個数が減っている a が削除、増えている b が挿入、変化していない c が保持となる[3)]。図 1 の各領域の特徴は、表 1 のようにまとめられる。例えば、図の “*What*” を含む領域は、$S \to R$ で削除され、かつ、$S \to H$ でも正しく削除される n-gram を含む。この領域を、True Delete (TD) と呼ぶ。同様に、“*Who*” の領域は、$S \to R, S \to H$ でともに挿入される n-gram を含む True Insert (TI) と呼び、“*you*” の領域は $S \to R, S \to H$ でともに保持される n-gram を含む True Keep (TK) と呼ぶ。TD、TI、TK は、$S \to R$ と $S \to H$ で訂正の種類が一致しており、True Positive (TP) である。“*?*”、“*!*” の領域は、$S \to R$ では削除、挿入がされないが、$S \to H$ では過剰に削除、挿入がされている n-gram を含み、それぞれ、Over-Delete (OD)、Over-Insert (OI) と呼ぶ。OD、OI の要素は $S \to H$ で間違って削除、挿入されているため、 False Positive (FP) である。“*is*”、“*are*” の領域は、$S \to H$ で本来 $S \to R$ と同様に削除、挿入が行われるべきだった n-gram を含み、それぞれ、Under-Delete (UD)、Under-Insert (UI) と呼ぶ。UD、UI の要素は $S \to H$ で削除、挿入されるべきだったため、False Negative (FN) である。

次に、各領域に含まれる n-gram の数を多重集合の演算によって求める方法を説明する。多重集合 A の要素 x の重複度 $m_A(x)$ は、x が A に現れる回数を表す。例えば、$A = \{a, a, a\ a\}$ で、$m_A(a) = 2$ である。本稿では、各領域に含まれる n-gram を数えるための演算として、多重集合の積 ($\cap$)、和 ($\cup$)、差 ($\setminus$) を用いる。多重集合 A と B に対する各演算は、A または B の任意の要素 x の重複度に関して以下のように定義される。

$$m_{A\cap B}(x) = \min(m_A(x), m_B(x)),$$
$$m_{A\cup B}(x) = \max(m_A(x), m_B(x)),$$
$$m_{A\setminus B}(x) = \max(m_A(x) - m_B(x), 0).$$

ゆえに、図 1 のベン図の 7 領域に含まれる n-gram x の数は、以下のように表される。

$$\begin{aligned}
\mathrm{TD}(x) &= m_{S\cap\overline{R}\cap\overline{H}}(x) = m_{S\setminus(R\cup H)}(x) \\
&= \max\{m_S(x) - \max(m_R(x), m_H(x)), 0\}, \\
\mathrm{TI}(x) &= m_{\overline{S}\cap R\cap H}(x) = m_{(R\cap H)\setminus S}(x) \\
&= \max\{\min(m_R(x), m_H(x)) - m_S(x), 0\}, \\
\mathrm{TK}(x) &= m_{S\cap R\cap H}(x) = \min(m_S(x), m_R(x), m_H(x)), \\
\mathrm{OD}(x) &= m_{S\cap R\cap\overline{H}}(x) = m_{(S\cap R)\setminus H}(x) \\
&= \max\{\min(m_S(x), m_R(x)) - m_H(x), 0\}, \\
\mathrm{OI}(x) &= m_{\overline{S}\cap\overline{R}\cap H}(x) = m_{H\setminus(S\cup R)}(x) \\
&= \max\{m_H(x) - \max(m_S(x), m_R(x)), 0\}, \\
\mathrm{UD}(x) &= m_{S\cap\overline{R}\cap H}(x) = m_{(S\cap H)\setminus R}(x) \\
&= \max\{\min(m_S(x), m_H(x)) - m_R(x), 0\}, \\
\mathrm{UI}(x) &= m_{\overline{S}\cap R\cap\overline{H}}(x) = m_{R\setminus(S\cup H)}(x) \\
&= \max\{m_R(x) - \max(m_S(x), m_H(x)), 0\}.
\end{aligned}$$

さらに、ある特定の長さ n の n-gram における $S \to H$ の訂正の TP、FP、FN は以下のように表される。

$$\mathrm{TP}_{n,S,R,H} = \sum_{\forall n\text{-gram } x} (\mathrm{TI}(x) + \mathrm{TD}(x) + \mathrm{TK}(x)),$$
$$\mathrm{FP}_{n,S,R,H} = \sum_{\forall n\text{-gram } x} (\mathrm{OI}(x) + \mathrm{OD}(x)),$$
$$\mathrm{FN}_{n,S,R,H} = \sum_{\forall n\text{-gram } x} (\mathrm{UI}(x) + \mathrm{UD}(x)).$$

GREEN は、コーパス単位で TP、FP、FN を集計し、F 値を求める。$\mathbb{S} = (S_1, \ldots, S_D), \mathbb{R} = (R_1, \ldots, R_D), \mathbb{H} = (H_1, \ldots, H_D)$ は、D 個の原文、参照文、訂正文の組を表す。GREEN は、1 から、評価に用いる n-gram の最大長 N までの n-gram の長さで、適合率 (precision)、再現率 (recall) を計算し、GLEU と同様にその幾何平均を適合率、再現率とする。

$$\mathrm{precision}(\mathbb{S}, \mathbb{R}, \mathbb{H}) = \left(\prod_{n=1}^{N} \frac{\sum_{i=1}^{D} \mathrm{TP}_{n,S_i,R_i,H_i}}{\sum_{i=1}^{D} (\mathrm{TP}_{n,S_i,R_i,H_i} + \mathrm{FP}_{n,S_i,R_i,H_i})} \right)^{\frac{1}{N}},$$
$$\mathrm{recall}(\mathbb{S}, \mathbb{R}, \mathbb{H})$$

3) アライメントを取らないため置換操作は現れない。置換は削除と挿入を組み合わせたものに対応する。

$$= \left(\prod_{n=1}^{N} \frac{\sum_{i=1}^{D} \mathrm{TP}_{n,S_i,R_i,H_i}}{\sum_{i=1}^{D} \left(\mathrm{TP}_{n,S_i,R_i,H_i} + \mathrm{FN}_{n,S_i,R_i,H_i} \right)} \right)^{\frac{1}{N}}.$$

最後に F_β 値を計算する。

$$F_\beta(\mathbb{S}, \mathbb{R}, \mathbb{H}) = \frac{(1+\beta^2)\mathrm{precision}(\mathbb{S}, \mathbb{R}, \mathbb{H})\mathrm{recall}(\mathbb{S}, \mathbb{R}, \mathbb{H})}{\beta^2 \mathrm{precision}(\mathbb{S}, \mathbb{R}, \mathbb{H}) + \mathrm{recall}(\mathbb{S}, \mathbb{R}, \mathbb{H})}.$$

本稿では、この F_β 値を GREEN_β と呼ぶ。

複数の参照文を用いる場合は、M^2 や ERRANT と同様に、次のように計算を行う。i 番目の原文 S_i に対して、m 個の参照文 $R_{i_1}, \ldots, R_{i_m}$ がある時、この中から、訂正文 H_i に対して、文単位の GREEN_β を最大にする参照文 $\hat{R}_i$ を選ぶ。

$$\hat{R}_i = \underset{R \in \{R_{i_1}, \ldots, R_{i_m}\}}{\mathrm{argmax}} \mathrm{GREEN}_\beta((S_i), (R), (H_i)). \quad (1)$$

式 1 で得られた D 個の参照文 $\hat{\mathbb{R}} = \{\hat{R}_1, \ldots \hat{R}_D\}$ を用いて $\mathrm{GREEN}_\beta(\mathbb{S}, \hat{\mathbb{R}}, \mathbb{H})$ を計算し、評価に用いる。

3 実験

自動評価値と人手評価値の Pearson の相関係数 r および Spearman の順位相関係数 ρ を計測し、GREEN の有効性を検証する。人手評価値の計算には GEC での有効性が認められている Expected Wins 法 [11] を用いる。n-gram の最大長 N は、単語単位の分割の場合は BLEU [12] や GLEU に従い $N = 4$、文字単位は chrF [13] に従い $N = 6$ とする。分割単位の違いは、“wordGREEN” (単語単位) や “charGREEN” (文字単位) と表記し区別する。

比較対象の GEC の評価尺度は以下の通りである。

M^2 [7]: 人手でアライメントが付与された参照文の編集に訂正文の編集が最大限アライメントされるよう探索し、任意の訂正文に対して自動的に原文とのアライメントを求め、F_β 値を計算する。公平な速度比較のため公式実装より高速な実装を利用した[4]。

I-measure [9]: 原文、参照文、訂正文間の多重アライメントを求め、原文から参照文、原文から訂正文へのアライメントを比較し、その正誤に対して重み付き正解率を計算する。M^2 は改悪された訂正文と変更がない訂正文を区別できないが、I-measure は変更なしの文に 0、改悪された文に負のスコアを与えられる。公式実装[5]を用い、高速に実行するため `-nomix` オプションを指定し実行した。

GLEU [10]: BLEU の分子に罰則項を追加することで人手評価値と正の相関を示すように設計された。本稿では訂正版 [14] の式に基づき計算する。

ERRANT [8]: M^2 と同様に F_β 値を計算するが、編集を Damerau-Levenshtein 距離とルールによる区間検出で抽出する。そのため、人手でアライメントを付与した参照文を必要としない。公式実装の `v2.0.0` を用いた[6]。

GREEN、M^2、ERRANT での評価を行う際は、β として、各データセット上で最適な β_* を推定して用いる。β_* の値として、β を 0.00 から 5.00 まで 0.01 ずつ変え、テストデータを 10 分割し、その内の 9 個の分割を用いて人手評価との相関を最大にする β を 10 通り求め、その平均の値を β_* として用いた[7]。

評価データセットとして CoNLL および GMEG を用いる。CoNLL は、CoNLL-2014 Shared Task [5] の評価データセットである。原文は 1312 文ある。参照データは 2 つあり、Shared Task の参加システムの訂正データは 12 ある。原文と 12 システムの訂正文の 13 文対に人手で順位が付与されている [11]。GMEG は、FCE、Wiki、Yahoo の 3 データセットからなる [15]。原文は、それぞれ、1936、1981、1999 文ある。参照データは 4 つあり、訂正データは 6 つある。ドメインはデータセットごとに異なり、FCE は学習者のエッセイ、Wiki は Wikipedia の記事、Yahoo はウェブ上の投稿となっている。原文と 6 システムの訂正文の 7 文対に人手評価が付与されている。

3.1 人手評価との相関

CoNLL、GMEG データセットでの人手評価との相関を表 2 に示す。CoNLL と GMEG のすべてのデータセットで charGREEN が人手評価と最も高い相関を示した。charGREEN は、wordGREEN より良い性能を示す。charGREEN は、綴り誤りなどに対する文字単位の訂正と、単語単位の訂正を異なる規模で評価でき、より人手評価と近い評価が行えると考えられる。M^2 と ERRANT は、CoNLL、FCE では GLEU、I-measure と同等かより高い性能を示すが、Wiki、Yahoo ではより低い性能を示す。これは、学習者コーパスのドメイン以外では M^2 と ERRANT が低い性能を示す可能性を示唆する。I-mesarue は、CoNLL では負の相関を示すが、GMEG では比較的良い性能を示し、人手評価においてデータセットの

4) https://github.com/craggy-otake/m2scorer_python3_fast
5) https://github.com/mfelice/imeasure
6) https://github.com/chrisjbryant/errant
7) 付録 A.1 でこの方法の経緯および留意点を説明した。

	CoNLL			FCE		Wiki		Yahoo		
	r	ρ	時間	r	ρ	r	ρ	r	ρ	計算量
$M^2_{\beta_*}$	0.697	0.725	5.03	0.972	0.964	0.544	0.464	0.738	0.857	$O(k^2)$
β_*	0.17	0.15		0.91	0.73	0.01	1.79	0.08	0.05	
I-measure	-0.250	-0.385	197.29	0.904	0.929	0.892	0.714	0.817	0.714	$O(k^3)$
wordGLEU	0.696	0.445	1.95	0.961	0.964	0.684	0.429	0.938	0.750	$O(k)$
charGLEU	0.606	0.593	3.86	0.992	0.964	0.964	0.821	0.883	0.893	$O(k)$
ERRANT$_{\beta_*}$	0.714	0.742	33.60	0.972	0.964	0.604	0.464	0.781	0.750	$O(k^2)$
β_*	0.20	0.19		0.90	0.82	0.00	3.84	0.01	0.45	
wordGREEN$_{\beta_*}$	0.790	0.709	**0.50**	0.989	0.964	0.951	0.714	0.927	**0.964**	$O(k)$
β_*	1.67	2.51		1.14	1.58	1.12	1.04	2.16	2.13	
charGREEN$_{\beta_*}$	**0.803**	**0.824**	1.59	**0.993**	**1.000**	**0.974**	**0.893**	**0.973**	**0.964**	$O(k)$
β_*	1.85	1.96		1.69	1.51	1.56	1.92	2.63	2.43	

表 2 CoNLL、GMEG データセットでの自動評価尺度と人手評価との Pearson の相関係数 r および Spearman の相関係数 ρ、並びに、各手法の平均事例での時間計算量を示す。CoNLL は、AMU の評価の 10 回の平均実行時間 (秒) を付した。

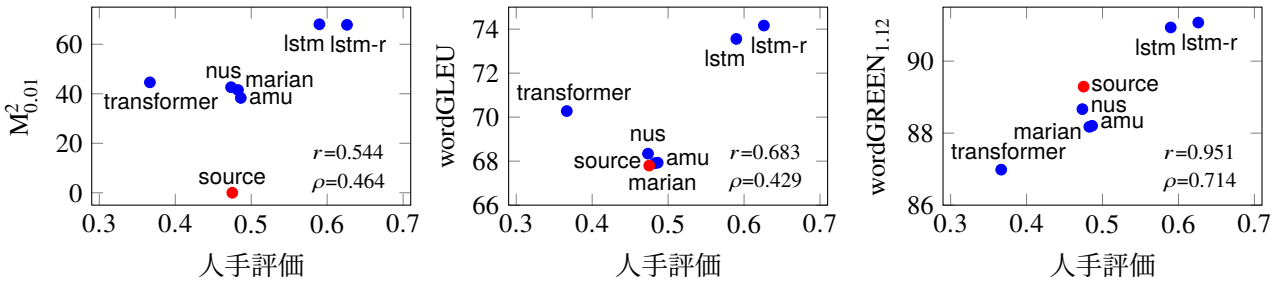

図 2 Wiki での $M^2_{0.01}$、wordGLEU、$wordGREEN_{1.12}$ と人手評価の相関

違いに頑健ではないと言える。

3.2 計算時間と計算量

原文、参照文、訂正文のトークン数が k 以下であるときの平均時間計算量と、CoNLL データセットでの実際の実行時間を表 2 に示す[8]。n-gram の種類数はトークン数 k を超えないため、GREEN と GLEU の計算量は $O(k)$ である。GLEU は、複数の参照文を用いる場合にサンプリングを行うため、計算時間が GREEN の 2 倍以上に長くなる。M^2 と ERRANT は、編集距離の計算に $O(k^2)$ の動的計画法を用いる。3 文間の多重アライメントを求める I-measure の計算量は $O(k^3)$ である。以上より、GREEN は従来の手法よりも計算量が小さく、高速と言える。

3.3 人手評価の値が低いシステムの評価

図 2 に M^2、wordGLEU、wordGREEN と人手評価の値を図示する。β は表 2 の β_* を用いた。M^2 は原文 (source) にスコア 0、人手評価値が原文よりも低い transformer の訂正文に不当に良いスコアを与えている。GLEU も transformer に不当に良いスコアを与えている[9]。このため、これらの手法は Wiki データセットで比較的悪い相関を示す。一方、GREEN は source を、人手評価値が近い amu, marian, nus とほぼ同等に評価しており、また、transformer にも低いスコアを与えられている。GREEN は原文や人手評価値の低いシステムの出力に対して妥当な評価が可能であり、人手評価と高い相関を示す。

4 おわりに

本稿では、GEC のための新しい自動評価尺度 GREEN を提案した。GREEN は、原文、参照文、訂正文を n-gram の多重集合として扱うことで、F 値を計算する。CoNLL、GMEG データセットでの実験において、GREEN は従来手法よりも人手評価との高い相関を示した。また、M^2、ERRANT、I-measure と異なり、文間のアライメントを用いないため計算量が小さい。さらに、M^2 や GLEU で人手評価値の低い訂正文に高い評価を与えてしまう事例でも、GREEN は人手評価に近い評価が行える。

8) Intel® Core™ i9-9900K 上で実行した。

9) 付録 A.3 でこの理由を説明した。

謝辞

本研究は国立研究開発法人新エネルギー・産業技術総合開発機構（ＮＥＤＯ）の助成事業（JPNP20006）並びに JSPS 科研費 JP23KJ0930 の助成を受けたものである。

参考文献

[1] Robert Dale and Adam Kilgarriff. Helping our own: The HOO 2011 pilot shared task. In **Proceedings of the 13th European Workshop on Natural Language Generation**, pp. 242–249, Nancy, France, September 2011. Association for Computational Linguistics.

[2] Robert Dale, Ilya Anisimoff, and George Narroway. HOO 2012: A report on the preposition and determiner error correction shared task. In **Proceedings of the Seventh Workshop on Building Educational Applications Using NLP**, pp. 54–62, Montréal, Canada, June 2012. Association for Computational Linguistics.

[3] Vidas Daudaravicius, Rafael E. Banchs, Elena Volodina, and Courtney Napoles. A report on the automatic evaluation of scientific writing shared task. In **Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications**, pp. 53–62, San Diego, CA, June 2016. Association for Computational Linguistics.

[4] Hwee Tou Ng, Siew Mei Wu, Yuanbin Wu, Christian Hadiwinoto, and Joel Tetreault. The CoNLL-2013 shared task on grammatical error correction. In **Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task**, pp. 1–12, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[5] 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 **Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task**, pp. 1–14, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[6] Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. The BEA-2019 shared task on grammatical error correction. In **Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications**, pp. 52–75, Florence, Italy, August 2019. Association for Computational Linguistics.

[7] Daniel Dahlmeier and Hwee Tou Ng. Better evaluation for grammatical error correction. In **Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 568–572, Montréal, Canada, June 2012. Association for Computational Linguistics.

[8] 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 (Volume 1: Long Papers)**, pp. 793–805, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[9] Mariano Felice and Ted Briscoe. 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**, pp. 578–587, Denver, Colorado, May–June 2015. Association for Computational Linguistics.

[10] Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 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 (Volume 2: Short Papers)**, pp. 588–593, Beijing, China, July 2015. Association for Computational Linguistics.

[11] Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Edward Gillian. Human evaluation of grammatical error correction systems. In **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**, pp. 461–470, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[12] 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.

[13] Maja Popović. chrF: character n-gram F-score for automatic MT evaluation. In **Proceedings of the Tenth Workshop on Statistical Machine Translation**, pp. 392–395, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[14] Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. GLEU without tuning. arXiv:1605.02592, 2016.

[15] Courtney Napoles, Maria Nădejde, and Joel Tetreault. Enabling robust grammatical error correction in new domains: Data sets, metrics, and analyses. **Transactions of the Association for Computational Linguistics**, Vol. 7, pp. 551–566, 2019.

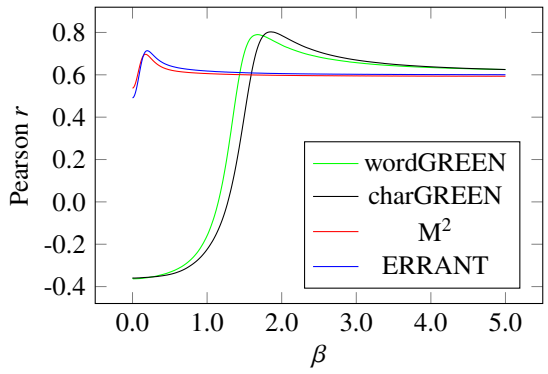

図 3　β を変えた時の CoNLL での Pearson の相関係数 r

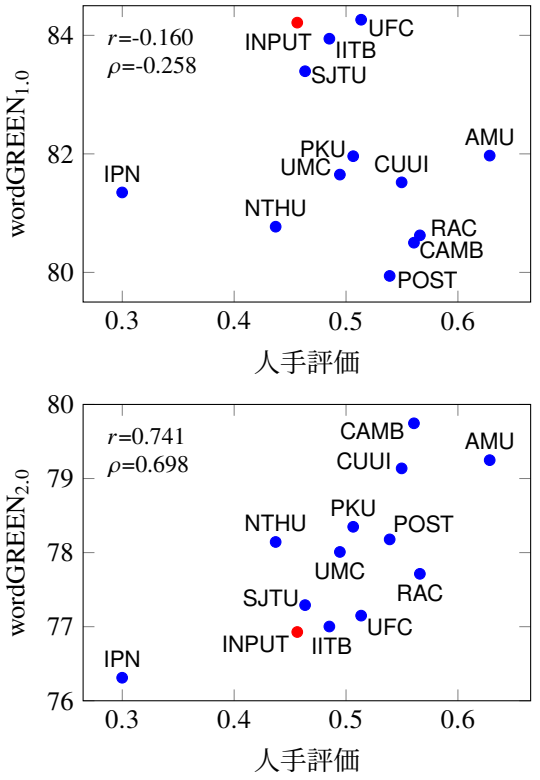

図 4　CoNLL での $\text{wordGREEN}_{1.0}$、$\text{wordGREEN}_{2.0}$ と人手評価の相関

A　付録

A.1　β_* の推定方法に関して

β_* の推定は、β_* を推定するための検証データを用意するか、交差検証を行うべきである。ところが、10 個の異なる β_* で求めた F_{β_*} 値の平均値は妥当性に乏しいため、本稿では評価データの部分データで相関を最大にするような β の平均 β_* を求めてから、F_{β_*} 値を計算した。しかし、これは β をパラメータとして持たない他の手法との比較においてフェアではないという点に留意が必要である。最適

$$
\begin{aligned}
p_n &= \frac{\sum_{\forall n\text{-gram } x \in R\cap H} m_{R\cap H}(x) - \sum_{\forall n\text{-gram } x \in S\cap H} \max\{0, m_{S\cap H}(x) - m_{R\cap H}(x)\}}{\sum_{\forall n\text{-gram } x \in H} m_H(x)} \\
&= \frac{\sum_{\forall n\text{-gram } x \in R\cap H} m_{R\cap H}(x) - \sum_{\forall n\text{-gram } x \in S\cap H} \max\{0, \min(m_S(x), m_H(x)) - \min(m_R(x), m_H(x))\}}{\sum_{\forall n\text{-gram } x \in H} m_H(x)} \\
&= \frac{\sum_{\forall n\text{-gram } x \in R\cap H} m_{R\cap H}(x) - \sum_{\forall n\text{-gram } x \in S\cap H} \max\{0, \min(m_S(x), m_H(x)) - m_R(x)\}}{\sum_{\forall n\text{-gram } x \in H} m_H(x)} \\
&= \frac{\sum_{\forall n\text{-gram } x} m_{R\cap H}(x) - \sum_{\forall n\text{-gram } x} m_{(S\cap H)\setminus R}(x)}{\sum_{\forall n\text{-gram } x} m_H(x)} = \frac{\sum_{\forall n\text{-gram } x} \mathsf{TK}(x) + \mathsf{TI}(x) - \mathsf{UD}(x)}{\sum_{\forall n\text{-gram } x} \mathsf{TK}(x) + \mathsf{TI}(x) + \mathsf{OI}(x) + \mathsf{UD}(x)}
\end{aligned}
$$

図 5　GLEU の再解釈のための式変形

な β において高い相関を示すか比較するという点では、M^2、ERRANT とはフェアな比較になっている。

A.2　最適な β に関する分析

図 3 は、CoNLL データセット上で β の値を変えた時の各評価尺度での Pearson の相関係数 r の変化を示す。GREEN は、適切な β を設定すれば M^2 や ERRANT より性能が良くなるが、β を小さく設定しすぎると性能が低下してしまう。この原因を調べるために、図 4 に、$\beta = 1.0, 2.0$ での wordGREEN と人手評価のスコアを示す。IITB、INPUT、SJTU、UFC では、$\text{wordGREEN}_{1.0}$ が不当に高い。INPUT は原文であり訂正がない。また、IITB、SJTU、UFC は、全出力の中で原文からの訂正が最も少ない 3 出力である。これらの出力では適合率が高くなるため、β が小さくなると wordGREEN が不当に高くなる。$\beta = 2.0$ など、β を高く設定するとより正確な評価が可能となる。$\text{wordGREEN}_{2.0}$ は、積極的に正しい訂正を行うシステム (AMU) に高いスコアを与え、過度に訂正が控えめなシステム (IITB) や誤った訂正が多いシステム (IPN) に低いスコアを与えられる。

A.3　GLEU の再解釈

GLEU は、人手評価と正の相関を示すように BLEU に罰則項を導入した手法である [14]。罰則項の効果を調べるため、図 5 に GLEU に対する本稿の n-gram の分類を用いた式変形を示す。この式から、GLEU の罰則項は UD そのものであることがわかる。UD は GREEN では FN であるため、GLEU は FN を考慮できているが、FP は罰則として考慮できていないと言える。図 2 に示すように、wordGLEU は FP を多く生成する transformer を不当に高く評価してしまう。