# 日本語文法誤り訂正における最適な分割単位の調査

水本 智也 [1,2]
[1] フューチャー株式会社　[2] 理化学研究所
t.mizumoto.yb@future.co.jp

## 1 はじめに

英語の文法誤り訂正の研究が盛んに行なわれるようになり，その性能を競う Shared Task も 2013 年，2014 年，2019 年と開催された [1, 7, 8]．Shared Task で使用された共通の評価セット上において文法誤り訂正手法の研究開発が行なわれており，年々その訂正性能は向上している．

英語の文法誤り訂正の研究が盛んに行なわれている一方で，日本語を対象とした研究はほとんど行なわれていない．日本語を対象したものでは，助詞を対象とした研究 [9, 12] や，英語のように誤りタイプを限定しないで訂正する研究 [13, 14] がある．また，英語の文法誤り訂正では，機械翻訳タスクで使われている Neural Machine Translation (NMT) の手法が主流として使われているが，日本語の文法誤り訂正では 2019 年に藤本ら [14] が NMT ベースの訂正器を使ったのみである．また，日本語文法誤り訂正の性能は英語を対象としたもの研究と比べると高いとは言えない．

日本語の文法誤り訂正が難しい理由の一つは，誤りを含んだ文の単語分割が難しいことである．英語の文法誤り訂正の場合は単語が空白で区切られているため分割する必要はなく，誤りがあったとしてもそのまま処理することが可能である．しかしながら，日本語を対象とした場合は文を単語もしくは何らかの文字列の単位に分割して処理する必要がある．学習者の書いた文のように単語分割したい文に誤りが含まれていると形態素解析器が単語分割に失敗することがある．以下の例は学習者が書いた文を訂正したものとそれを形態素解析器で単語分割したものであり，適切に単語分割ができている．

(1) a. でもじょうずじゃありません
　　b. でも じょうず じゃ あり ませ ん

一方，学習者の書いた原文とその単語分割したものは以下のようになり，形態素解析器で解析すると単語分割に失敗してしまう．

(2) a. でもじょずじゃりません
　　b. でも じ ょずじゃりません

このように単語分割が失敗している場合は，分割された文字列が適切な意味のあるかたまりとなっておらず訂正が困難となる．この問題に対処するために水本らは，SMT による文法誤り訂正で文字分割をベースとして日本語誤り訂正を行なう方法を提案した [13]．

自然言語処理の分割単位の観点から見ると，深層学習を用いた手法，特に応用タスクの手法では，サブワードと呼ばれる分割単位が用いられる．サブワードは単語を分割することで，低頻度語の問題に対処することを可能としている．Sennrich ら [10] は Byte Pair Encoding (BPE) を NMT に適応し，その後 NMT での一般的な分割方法として使われている．英語の文法誤り訂正においても NMT の手法を使ったシステムでは，BPE が使用されている．また，日本語のような分かち書きされていない言語に対して，単語分割することなく文から直接高頻度語をひとかたまりとして分割する手法として Sentencepiece が提案されている [4]．

水本らによって，日本語の誤り訂正で SMT による文法誤り訂正での分割単位の研究は行なわれ分析も行なわれた一方で，NMT を使った訂正における最適な分割単位が何かはわかっていない．すでに述べたように，英語を対象とした場合と異なり，日本語を対象とした場合は誤った文を何らかの文字列の単位に分割して訂正する必要がある．そのため日本語文法誤り訂正において，分割単位が訂正性能に与える影響は英語の場合に比べて大きいと考える．そこで本研究では，日本語文法誤り訂正において分割単位が訂正性能に与える影響を調べる．また，実際の訂正結果の違いから考察を行なう．

## 2 分割単位

本研究で実験に使用する分割単位について説明する．1節で挙げた分割単位を元に以下の 5 つの基本単位を用いる．

**1. 単語分割 (WORD 分割)**　自然言語処理において一般的な分割単位でありベースラインとして用いる．文法誤

**表1:** 学習者の書いた誤り文 “いい天気てすね” と訂正文 “いい天気ですね” に対する分割例

| 分割方法 | 誤り文 | 訂正文 |
|---|---|---|
| WORD 分割 | いい 天気 て すね 。 | いい 天気 です ね 。 |
| CHAR 分割 | い い 天 気 て す ね 。 | い い 天 気 で す ね 。 |
| SP 分割 | __ いい天気 て す ね 。 | __ いい天気 ですね 。 |
| WORD+BPE 分割 | いい 天気 て す@@ ね 。 | いい 天気 です ね 。 |
| WORD+SP 分割 | __いい __天気 __て __すね __。 | __いい __天気 __です __ね __。 |

り訂正においては，誤っている文を解析する必要があるため単語分割に失敗し，訂正に悪影響を与える可能性が高い．

**2. 文字分割 (CHAR 分割)** 単語よりも小さい単位である文字単位に分割したものである．単語分割の誤りの影響を受けなくなる一方で単語の持つ情報も失われるデメリットも存在する．

**3. Sentencepiece (SP 分割)** 単語分割を行なうことなく，文から直接分割を行ない，出現頻度の高い文字列をひとかたまりとして分割し，低頻度語はより短い単位，文字などに分割される．この手法は，誤りがあり出現頻度の低い文字列は文字単位に分割され，誤りでない部分で頻度の高いは文字列は長い単位で分割されるため文法誤り訂正にも効果の高い分割方法だと考える．

**4. 単語分割 +BPE (WORD+BPE 分割)** 英語の文法誤り訂正で用いられる設定であり，本研究でも比較対象とする．日本語を対象とした場合はまず 1 の単語分割を行なった後に，BPE によって分割を行なう．最初に単語分割をしているため単語分割の失敗の影響を受ける可能性はあるが，低頻度語はより小さい単位として分割されるため，単語分割で訂正する場合よりも頑健に訂正可能だと考える．

**5. 単語分割 +Sentencepiece (WORD+SP 分割)** 単語分割 +BPE と同様に，単語分割した後に Sentencepiece で分割を行なう．BPE を使った場合と同じく単語分割の失敗の影響を受ける可能せがあり，同じような結果が得られると考えられる．

表1に各分割方法で分割した際の例を示す．どの分割方法でも訂正後の文に関しては妥当的な分割になっている．学習者の書いた誤り文に関しては，“てすね” の部分を単語分割では “て” と “すね” に分割しており，分割として不自然である．また，単語分割から BPE や Sentencepiece で分割を行なった場合，BPE では文字列 “すね” を “す @@” と “ね” に分割しているが，Sentencepiece では “すね” のままと少し違いが見られる．Sentencepiece 単独による分割を見ると，“てすね” の文字列を “て” と “す” と “ね” のように文字単位で分割されており，低頻度な語に関しては文字分割で訂正する場合と同じような効果が期待できる．

別の比較対象として，学習者の書いた誤りを含む文（**誤り文**）と訂正後の文（**訂正文**）で異なる分割単位を使う手法を用いる．これは水本らが提案した誤り文は文字分割，訂正文は単語分割にして SMT で訂正する手法 [13] を NMT に応用したものになる．この手法では，誤り文は文字分割になっているため単語分割の影響を受けず，訂正文は単語単位になっているため単語同士の繋がりを考慮できるため高い訂正性能が期待できる．本研究では 1. 誤り文は文字分割，訂正文は単語分割にしたもの (**CHAR-WORD 分割**) と 2. 誤り文は文字分割，訂正文は Sentencepiece で分割したもの (**CHAR-SP 分割**) の 2 つを比較対象として追加する．

## 3 実験

2節で説明した分割単位が，日本語の文法誤り訂正においてどのような影響があるかを検証するために実験を行なう．実験に用いる分割単位としては，表1にある 5 つの分割単位と，誤り文と訂正文で異なる分割単位を使う 2 つの手法の合計 7 つの手法で比較を行なう．

### 3.1 データと評価方法

訂正システムの学習には水本ら [13] が相互添削型 SNS から作成した Lang-8 Learner Corpora[*1] を使用し，コーパスの前処理は水本らの方法に従い行なった．また，英語の文法誤り訂正の先行研究に従い，訂正が行なわれていない文対に関しては学習用データから除外した [2]．その結果，学習用データは 1,169,604 文対となった．

開発用，評価用のデータとして「日本語学習者による日本語作文と，その母語訳との対訳データベース」を使用した[*2]．データベースから抽出した文対が 4,952 文対あり，開発用に 2,452 文対，評価用に 2,500 文対を使用した．

[*1] https://sites.google.com/site/naistlang8corpora/
[*2] http://db3.ninjal.ac.jp/contr-db/

**表2:** BPE，Sentencepiece で語彙サイズを変えた場合の各分割方法を使ったシステムの訂正性能

| 分割方法 | 語彙サイズ | 文字 GLEU |
|---|---|---|
| SP 分割 | 4,000 | **62.89** |
| | 8,000 | 62.78 |
| | 16,000 | 62.88 |
| WORD+BPE 分割 | 4,000 | 62.18 |
| | 8,000 | 62.38 |
| | 16,000 | **62.48** |
| WORD+SP 分割 | 4,000 | 62.68 |
| | 8,000 | 62.72 |
| | 16,000 | **62.84** |

訂正性能の評価には GLEU [5, 6] を使用した．GLEU は機械翻訳の評価で使用されている評価尺度 BLEU を改良して作られており，基本的には N-gram の一致率で評価される．文法誤り訂正の場合は，システムの訂正と正解とを比較するだけでなく，元の学習者の文とも比較する必要があり，そこが BLEU との大きな違いである．日本語を対象として評価する際には，訂正時と同様に，N-gram 一致を見る際の単位（分割単位）が問題となる．本研究では，文字単位での N-gram 一致率を見る文字単位の GLEU（**文字 GLEU**）と単語単位での N-gram 一致率を見る単語単位での GLEU（**単語 GLEU**）の両方の結果を報告する．

### 3.2 訂正手法

本研究の実験では訂正手法として，Transformer [11] を用いる．英語の文法誤り訂正では，Transformer を用いた手法が高い性能を達成している [3]．Transformer の実装は Fairseq[*3] を利用し，ハイパーパラメータ設定は，一般的なものを用いた．入力 Embedding の事前学習は行なっていない．また，報告する全ての値は，シードを変えて 4 回学習しその 4 つのモデルの訂正性能を平均したものである．

WORD 分割での分割には MeCab を使用し，辞書として IPA 辞書を使用した．Sentencepiece と BPE では語彙サイズを指定して，分割を学習することができる．開発用データを用いて，予備実験として SP 分割，WORD+BPE 分割，WORD+SP 分割で語彙サイズ {4,000, 8,000, 16,000} と変更した場合の性能を調査した．表2に文字 GLEU での評価結果を示す．SP 分割以外の結果を見ると，語彙サイズが 16,000 の時が最も高くなっており，語彙サイズが大きくなるにつれ性能が上がっている．SP 分割の場合は，語彙サイズが 4,000 が最も GLEU 値が高くなっているが，語彙サイズが 16,000 の場合もほぼ同程度であるため，評価データでの以降の実験結果は語彙サイズ 16,000 の時の値を報告する．

[*3] https://github.com/pytorch/fairseq

**表3:** 分割単位別の訂正性能

| 分割方法 | 文字 GLEU | 単語 GLEU |
|---|---|---|
| 原文 | 58.69 | 46.08 |
| WORD 分割 | 57.96 | 49.07 |
| CHAR 分割 | 60.82 | 48.96 |
| SP 分割 | 61.70 | 50.17 |
| WORD+BPE 分割 | 61.59 | 50.10 |
| WORD+SP 分割 | 61.85 | 50.44 |
| CHAR-WORD 分割 | 60.17 | 50.27 |
| CHAR-SP 分割 | **61.92** | **50.56** |

**表4:** “ハリラヤの特別な食物はとで もたくさんが ある。” に対する訂正例．変化のあった箇所を下線で示す．

| 分割方法 | システム訂正文 |
|---|---|
| WORD 分割 | ガクト の特別な食物はとでもたくさん＿ある。 |
| CHAR 分割 | ハリラヤの特別な食物はとてもたくさん＿ある。 |
| SP 分割 | ハリラヤの特別な食物はとても＿ある。 |

### 3.3 実験結果

分割単位を変えた場合の誤り訂正の結果を表3に示す．GLEU は原文，すなわち訂正をせずにそのまま出力した場合でも評価できるため，原文を出力とした場合の結果についても比較対象として示す．原文をそのまま出力した場合でも，文字 GLEU で見ると，WORD 分割よりも高い値となっている．WORD 分割では，長い文字列を訂正し誤ったものに変えてしまうことで文字 GLEU での減点が大きくなっていると考える．

従来の日本語文法誤り訂正でも使われてきた WORD 分割と CHAR 分割を比較すると，文字 GLEU では CHAR 分割の方が値が高い一方で，単語 GLEU では WORD 分割が 0.09 高い結果となった．これは上で挙げた WORD 分割の問題に加え，文字単位で訂正を行なった場合，誤っている単語を完璧に直せているわけではなく単語の一部だけ訂正していることを示唆している．

SP 分割の結果を見ると，WORD 分割や CHAR 分割よりも文字 GLEU，単語 GLEU の両方において高い値となった．これは Sentencepiece の分割が学習者の誤りを含む文，その訂正後の文に対して妥当な分割を行なっているからだと考える．WORD+BPE 分割と WORD+SP 分割は少しの差ではあるが，WORD+SP 分割の方が高い結果となった．

**表5:** 誤り文“トラックを運転中寝むくなったりして、事故が起こりやすいです。”に対する各分割方法による分割例と訂正例．正しい文は“トラックを運転中眠くなったりして、事故を起こしやすい。”であり，後半部分はどのシステムも訂正できていない．

| 分割方法 | 誤り文 | システム訂正文 |
|---|---|---|
| SP 分割 | __ トラック を 運転 中 寝 む くなった りして 、 事故 が 起こ り やすいです 。 | __ トラック を 運転 中 寝 む くなった りして 、 事故 が 起こ り やすいです 。 |
| WORD+SP 分割 | __ トラック __を __運転 __中 __寝 __む く __ なっ __たり __し __て __、 __事故 __が __起こ り __やすい __です __。 | __ トラック __を __運転 __中 __に __眠 っ __ たり __し __て __、 __事故 __が __起こ り __ やすい __です __。 |
| CHAR-SP 分割 | トラックを運転中寝むくなったりして、事故が起こりやすいです。 | __ トラック を 運転 中に 眠 くなった りして 、 事故 が 起こ り やすいです 。 |

学習者の文を文字分割し，訂正後の文を WORD 分割もしくは SP 分割にした CHAR-WORD 分割や CHAR-SP 分割の結果を見ると，どちらも入出力の分割が同じ場合と比べて高い．これは学習者の文側の分割の失敗が日本語の文法誤り訂正に影響を与えているということである．特に CHAR-WORD 分割は入出力共に単語分割である WORD 分割の場合に比べて文字 GLEU 値が 2.21，単語 GLEU が 1.2 高い．これは WORD 分割で，学習者の文に対する単語分割の失敗が訂正に影響していると言える．

## 4 考察

本節では，1. 基本的な分割単位である WORD 分割，CHAR 分割，SP 分割を使ったシステムの実際の訂正例と，2. Sentencepiece を使った分割方法 3 つでの違いを実際の訂正例を見ながら考察する．前者の 3 つの分割単位の違いによる比較するための例を表4に示す．WORD 分割はハリラヤが学習データには出現せず未知語なため別の単語に置き換えられているのに対し，他の 2 つでは分割された時点でハリラヤが文字単位に分割されているため，未知語とはならないため誤って異なる単語に変えることはなかった．この例では文字分割のみが正しく訂正できており，“とでも”を“とても”，“たくさんが”の“が”を削除するといった訂正が正しくできた．

表5に Sentencepiece を分割に使った 3 つのシステムの実際分割と訂正例を示す．分割例を見ると，SP 分割では“寝むくなったりして”が“寝 む くなった りして”と 4 単語に分割されており WORD+SP 分割の 7 単語，CHAR-SP 分割の 9 単語と比べて分割が少ない．この例を見ると，SP 分割の 4 単語への分割の仕方でも問題ないように思えるが，学習データ中での分割が，CHAR 分割に比べると一貫性に欠けており，訂正に影響が出ていると考えられる．実際システム訂正文を見ると，SP 分割では“寝むくなったりして”をそのまま出力しており，WORD+SP 分割では一部訂正，CHAR-SP 分割では“眠くなったり”と正しく訂正できている．

## 5 おわりに

日本語の文法誤り訂正タスクにおける文字列の分割単位が訂正性能に与える影響について調査した．従来の日本語文法誤り訂正に使われていた単語，文字に加えて，NMT で使用される BPE や Sentencepiece による分割，原言語・目的言語側で分割を変えた方法により実験を行なった．その結果，従来の単語や文字分割よりも BPE や Sentencepiece による分割が有効であることが明らかとなった．

## 参考文献


[1] Christopher Bryant et al. “The BEA-2019 Shared Task on Grammatical Error Correction”. In: *Proc. of BEA*. 2019, pp. 52–75.

[2] Shamil Chollampatt and Hwee Tou Ng. “A Multilayer Convolutional Encoder-Decoder Neural Network for Grammatical Error Correction”. In: *Proc. of AAAI*. 2018, pp. 5755–5762.

[3] Shun Kiyono et al. “An Empirical Study of Incorporating Pseudo Data into Grammatical Error Correction”. In: *Proc. of EMNLP*. 2019, pp. 1236–1242.

[4] Taku Kudo and John Richardson. “SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing”. In: *Proc. of EMNLP*. 2018, pp. 66–71.

[5] Courtney Napoles et al. “GLEU Without Tuning”. In: *arXiv* (2016). eprint: 1605.02592.

[6] Courtney Napoles et al. “Ground Truth for Grammatical Error Correction Metrics”. In: *Proc. of ACL*. 2015, pp. 588–593.

[7] Hwee Tou Ng et al. “The CoNLL-2013 Shared Task on Grammatical Error Correction”. In: *Proc. of CoNLL Shared Task*. 2013, pp. 1–12.

[8] Hwee Tou Ng et al. “The CoNLL-2014 Shared Task on Grammatical Error Correction”. In: *Proc. of CoNLL Shared Task*. 2014, pp. 1–14.

[9] Youichiro Ogawa and Kazuhide Yamamoto. “Japanese Particle Error Correction employing Classification Model”. In: *Proc. of IALP*. 2019, pp. 23–28.

[10] Rico Sennrich, Barry Haddow, and Alexandra Birch. “Neural Machine Translation of Rare Words with Subword Units”. In: *Proc. of ACL*. 2016, pp. 1715–1725.

[11] Ashish Vaswani et al. “Attention Is All You Need”. In: *Proc. of NIPS*. 2017, pp. 5998–6008.

[12] 今村賢治 et al. “小規模誤りデータからの日本語学習者作文の助詞誤り訂正”. In: 自然言語処理 19.5 (2012), pp. 381–400.

[13] 水本智也 et al. “日本語学習者の作文自動誤り訂正のための語学学習 SNS の添削ログからの知識獲得”. In: 人工知能学会論文誌 28.5 (2013), pp. 420–432.

[14] 藤本恭子 et al. “日本語作文の自動誤り訂正における統計的機械翻訳とニューラル機械翻訳の性能評価”. In: 言語処理学会第 *25* 回年次大会発表論文集. 2019, pp. 414–417.