

# 英作文誤り訂正における複数の手法の利用に関する考察

水本 智也<sup>1,a)</sup> 林部 祐太<sup>1,b)</sup> 坂口 慶祐<sup>1,c)</sup> 小町 守<sup>1,d)</sup> 松本 裕治<sup>1,e)</sup>

**概要:** 英語学習者の書く作文には様々な種類の文法誤りが含まれている。文法誤りの中にはヒューリスティックスを用いたルールで訂正できるものもあれば、長距離の依存関係や選択選好を考慮した統計的なモデルを用いないと訂正が難しいものもある。英語学習者の文法誤りの自動訂正に取り組んだ先行研究では、訂正する誤りの種類を数種類に限定して取り組んできたが、全ての種類の誤りを対象にした場合、1つのモデルだけで十分に対処できるかは未だ分かっていない。そこで、本稿では全ての誤りを対象に、誤りの種類に応じて異なる誤り訂正システムを構築して誤り訂正を行ない、誤り訂正システムの適用の仕方による誤り訂正の結果について分析・考察を行なう。

## 1. はじめに

英語学習者の書く作文には多くの種類の誤りが含まれている。表1は日本人大学生の書いた作文から作った学習者コーパスであるKJコーパス<sup>\*1</sup>の誤りを分類したものである<sup>\*2</sup>。最も頻出する誤りは冠詞に関する誤りであり、そのあと前置詞、名詞の語彙選択の誤りと続いている。文法誤りの特徴として、動詞の一致のように単純なルールでほとんどの誤りを訂正できる誤りもある一方、前置詞のように述語と名詞との関係や長距離の依存関係を参照しないと訂正できない誤りもある。

英語学習者の犯す誤りに対して、これまで多くの誤り訂正に関する研究が行なわれている。これらのほとんどは誤りの種類を1種類もしくは数種類に限定して誤りの訂正を試みている。例えば、Rozovskaya and Roth [30]は前置詞の誤り、Liuら [24]は動詞選択に関する誤り、Tajiriら [32]は動詞の時制、Lee and Seneff [23]は動詞の語形に関する誤り(動詞の一致、動詞の時制)、Dahlmeier and Ng [7]は前置詞と冠詞の誤り、Park and Levy [29]はスペリング誤り、冠詞、前置詞、語形(動詞の一致、動詞の時制)を対象として訂正を行なった。これらの研究は1つの誤り訂正シ

表1 KJコーパスにおける誤りの分布

種別	割合(%)
冠詞	23
前置詞	13
名詞の語彙選択	10
名詞の単複	7
代名詞	6
動詞の時制	6
動詞の語彙選択	5
動詞の人称・数の不一致に関する誤り	4
その他	26

テムで誤り訂正を行っているが、冒頭で述べたように誤りの種類によって有効な訂正の手がかりが異なるため、必ずしも最適な訂正ができていないとは限らない。

そこで本稿では、全ての種類の誤りを対象に、誤りの種類に応じて異なる誤り訂正システムを構築して誤り訂正を行ない、誤り訂正システムの適用の仕方による誤り訂正の結果について分析、考察を行なう。具体的には、各誤り訂正システムの出力結果を統合して最終出力を得る場合と、1つの誤り訂正システムで誤りを訂正してから異なる誤り訂正システムで訂正を行なう場合とで得られる結果がどのように変わるかについて分析を行なう。

## 2. 関連研究

学習者の誤りは種類に応じて訂正の難しさや有効な素性が異なるため、簡単なヒューリスティックルールと言語モデル、誤りの識別モデルなどの複数のモデルを組み合わせた手法が典型的に用いられている [22]。そして近年、複数のシステムの出力を組み合わせるために、機械学習におけるメタ学習の枠組みを用いた研究も行われている。

<sup>1</sup> 奈良先端科学技術大学院大学  
Nara Institute of Science and Technology

a) tomoya-m@is.naist.jp

b) yuta-h@is.naist.jp

c) keisuke-sa@is.naist.jp

d) komachi@is.naist.jp

e) matsu@is.naist.jp

<sup>\*1</sup> <http://gsk.or.jp/catalog/GSK2011-B/catalog.html>

<sup>\*2</sup> KJコーパスではスペリング誤りはアノテートの対象から除外されている。

たとえば, Gamon [12] は英語学習者の前置詞と冠詞の誤り検出・訂正タスクに対し, 言語モデルの出すスコアと誤りモデルの出す誤りクラスの確率を入力として受け取るメタ分類器を構築し, 大規模なネイティブコーパスを効率的に用いる手法を提案した. 本稿は, 前置詞と冠詞だけでなく全ての誤りを対象にしており, また, 言語モデルや誤りモデルの構築にネイティブコーパスではなくウェブデータから作成した大規模英語学習者コーパスを用いている.

また, Seo ら [31] は英語学習者の冠詞誤り訂正タスクに対し, 複数のコーパスから作成した分類器を作成し, それぞれの分類器の出力するスコアを入力としたメタ分類器を構築する手法を提案している. 本稿も複数のコーパスから誤り訂正システムを構築する点は共通しているが, 全ての種類の誤りを対象にしており, メタ学習器を用いて訂正モデルを学習するのではなく, さまざまなシステム統合方法を検討している点が異なる.

恐らく我々の研究にもっとも近いのは, Ehsan ら [11] が提案した統計的機械翻訳を用いたペルシャ語の文法誤り訂正システムである. 彼らはフレーズベースの統計的機械翻訳を用いることによって誤り訂正の再現率が低下する問題に対処するため, ルールベースの誤り訂正と組み合わせる手法を提案した. 彼らの手法では統計的機械翻訳と誤り訂正ルール両方が訂正候補を出力し, いずれかで訂正された結果が正しければ正解だとして評価を行なっているが, この oracle を用いた評価は人間に複数の訂正候補を見せて選択させることが前提となっており, システム出力をどのように組み合わせるのが最適かに関しては検討していない. 一方, 本稿ではシステムの出力結果を統合する問題について比較し, どのような順番でシステムを適用・結果を統合すれば最適な結果が得られるかの比較を行なっている.

### 3. 複数の誤り訂正システムを用いた誤り訂正

本稿では, 訂正する誤りに応じて複数の手法を用いて誤り訂正を行ない, 複数の手法の適用の仕方について議論を行なう. 誤り訂正に用いる手法と対象とする誤りは以下の通りである.

- フレーズベース統計的機械翻訳 - 全ての誤り
- 最大エントロピーモデル - 前置詞誤り
- ルールベース - 動詞の一致誤り

#### 3.1 訂正方法による誤りの分類

学習者の誤りは訂正の方法によって, 次の3通りに分類できる.

**置換誤り (replacement error)** は, 学習者が誤って本来使うべき単語とは, 異なる単語を使った誤りを指す. 例えば, 例 (1) の *with* である.

I went there *with*<sub>by</sub> bus. (1)

**挿入誤り (insertion error)** は, 本来その単語が不要であるのにも関わらず, 学習者が余分に書いてしまった誤りを指す. 例えば, 例 (2) の “about” である.\*3

We discussed *about*<sub>NONE</sub> the topic. (2)

**削除誤り (deletion error)** は, 単語が必須であるのにも関わらず, 学習者が単語を書いていない誤りを指す. 例えば, 例 (3) の “NONE” である.

This is the place to relax *NONE*<sub>in</sub>. (3)

本稿で扱う誤り訂正システムのうち, 誤りを限定しない訂正システムは全ての誤りを対象とし, 前置詞誤り訂正システムは置換誤りと挿入誤りを対象とし\*4, 動詞の一致誤り訂正システムは置換誤りを対象とする.

#### 3.2 全ての誤り訂正システム

本稿で用いる1つ目の誤り訂正システムはフレーズベースの統計的機械翻訳による, 誤りを限定しない誤り訂正システムであり, 置換誤り, 挿入誤り, 削除誤りを全て扱うことができる. フレーズベースの統計的機械翻訳を用いた文法誤り訂正には Brockett ら [4], Mizumoto ら [25] が行なったものがある. Brockett らは, フレーズベースの統計的機械翻訳を用いて英語の誤り訂正を行なったが, 訂正を行なう対象は名詞の加算・不加算に関する誤りのみであった. Mizumoto らは, 誤りを限定せず全ての誤りを対象として誤り訂正を行なったが, 英語ではなく日本語を対象としていた.

本稿では, フレーズベース統計的機械翻訳 [20] を用いて英語を対象として誤りを限定することなく誤り訂正を行なう.

$$\hat{e} = \arg \max_e P(e|f) = \arg \max_e \sum_{m=1}^M \lambda_m h_m(e, f) \quad (1)$$

式 (1) は対数線形モデルを使った統計的機械翻訳の式である [27]. ここで  $e$  はターゲット側 (訂正後の文) であり,  $f$  がソース側 (学習者の書いた訂正前の文) である.  $h_m(e, f)$  は  $M$  個の素性関数であり,  $\lambda_m$  が各素性関数に対する重みである. ソース側の文  $f$  に対して, 素性関数の重み付き線形和を最大化するターゲット側の文  $e$  を探せばいいことを意味している. 素性関数には, 翻訳モデルや言語モデルなどが用いられる. 翻訳モデルは一般に  $P(f|e)$  という条件付き確率の形で表される. この翻訳モデル  $P(f|e)$  はフレーズ間の対訳確率に分解して定義される. 言語モデルは一般に  $P(e)$  という確率の形で表され,  $n$ -gram 言語モデルが広

\*3 “NONE” は, そこに語が無いことを示す.

\*4 置換誤りと挿入誤り訂正問題は, 学習者が用いた前置詞に対して, 本来の前置詞を推測する多クラス分類として扱うことができる. しかしながら, 削除誤りは文中のいたるところに現れる可能性があり, 他の2つの誤り種類とは性質が異なるため, 本稿では置換誤りと挿入誤りの訂正にのみ取り組む.

く用いられている。また、翻訳モデルは添削前後の文で1対1対応のとれた添削コーパスから学習し、言語モデルはターゲット側言語から学習することができる。

### 3.3 前置詞誤り訂正システム

前置詞を適切に用いるのは英語学習者にとって難しい。そのため前置詞の誤用を自動的に検出・訂正する研究が活発になってきている。例えば、

- 人手で記述したルールに基づく方法 [10]
- コーパスにおける頻度を利用する方法 [16] [14]
- 機械学習を用いる方法 [18] [8] [13]

が提案されている。最近では、機械学習を用いる方法が多く提案されており、性能が良いことが報告されているため、本稿でも機械学習を用いたシステムを構築した。

英語には多くの前置詞がある。例えば、[1]では、50以上の前置詞が挙げられている。本システムでは、12種類の前置詞: *of, in, for, to, by, with, at, on, from, as, about, since* に絞って、誤り訂正を行う。これらの前置詞は、全前置詞の用法のうち、概ね91%をカバーするとの報告がある [6]。

前置詞の誤り訂正には、最大エントロピー法モデル [2] を多クラス分類器として用いた。素性は [33][9] で挙げられているものを用いた。次に用いた素性の一覧を示す。

- 表層素性
  - 周辺2単語の表層形・品詞・WordNetの意味クラス(全40種類)
  - 先行する動詞句・名詞句の主辞
  - 後続する動詞句・名詞句の主辞
- 構文素性
  - 前置詞の主辞・補語の表層形・品詞
  - 前置詞と主辞・補語の関係
  - 前置詞の構文木上での親と親の親のノード名
  - 前置詞の構文木上での親の子のノード名
- 頻度・意味素性
  - 着目している前置詞を *prep* に置換し、左  $i$  単語の列  $L$  と、右  $j$  単語の列  $R$  を用い ( $i+j \in \{3,4,5\}$ )、単語列  $LprepR$  を Web N-gram コーパスで検索したときの頻度  $f_{prep,i}$
  - 訂正前置詞候補集合  $T$  における、前置詞 *prep* の頻度の割合  $p_{prep,i} = \frac{f_{prep,i}}{\sum_{k \in T} f_{k,i}}$

### 3.4 動詞の一致誤り訂正システム

動詞の一致とは、文中において主語の人称、数、格、時制とそれに呼応する動詞の形が一致しなければならない、という文法規則である。例えば英語では、主語が三人称単数形現在時制の場合、一般的な規則動詞は語尾が *-s* あるいは *-es* へと変化した形に活用し (例: *He plays ..., She studies ...*)、不規則動詞に関しても主語の特性にあわせて活用する (例: *I am, You are, He is, I have, She has* etc.)。

英文誤り訂正において、動詞の一致誤りの検出および訂正は他の文法訂正に比べて歴史が長く、1980年代にIBMで始まった一連の研究 [15], [19] や、2000年代以降では Educational Testing Service (ETS) が開発した *Assessment of Lexical Knowledge (ALEK)* [21] や、*Criterion* [5], Microsoft Research による *ESL Assistant* [13] などが挙げられる。

先に述べたように動詞の一致は厳格な規則に従っているため、いずれのシステムにおいても動詞の一致誤り検出・訂正はヒューリスティクスに基づくルールベースで行われている。ただし、その精度に関する実験結果などは明らかにされていない。動詞の一致に関する精度を示している研究としては Lee and Seneff [23] が挙げられるが、彼らの手法は動詞の一致を含む種々の動詞誤り (受動態や進行形などにおける動詞の変化) を検出・訂正することを目的とし、ルールベースではなく構文解析結果によるテンプレートマッチと言語モデルによるフィルタリングを用いている。実験の結果、NICT JLE コーパス [17] に対する動詞の一致の誤り訂正・検出精度は、Precision (検出) = 0.847, Precision (訂正) = 0.714, Recall = 0.315, F 値 (検出) = 0.459, F 値 (訂正) = 0.437 であった。

今回我々が行う動詞の一致誤り検出・訂正は、係り受け解析の結果から主語および動詞を抽出し、人手で作成したルールに従って誤りを検出・訂正するというものである。品詞解析と係り受け解析には、Stanford Parser<sup>\*5</sup>を用いるが、Stanford Dependency では一致させるべき主語と動詞が直接的に取得できないケースがあるため、表2のような追加処理を行い、適切な主語と動詞のペアを取得する。取得した主語と動詞のペアに対して、それらが一致しているかどうかを人称、数、格、時制をもとに一般的な英文法のルールに則って判定する。

例えば “For example, very strange things \*was / were happened somewhere.” という文<sup>\*6</sup>に対する処理は以下のようになる。

1. 入力文を Stanford Parser で解析し、依存関係を取得する。
2. 1.の結果から、動詞 (happened) に対して、things が受動態の主語 (nsubjpass), was が受動態助動詞 (aux-pass) として係ることがわかるので、受動態の場合の追加規則 (表2) を適用する。
3. 追加規則から一致させる対象となる主語と動詞が things と was となる。
4. 主語 (things) と動詞 (was) の人称、数が一致していないので、主語に合わせて動詞を was → were へ訂正する。

予備実験として KJ コーパスを用いて動詞の一致誤りの

<sup>\*5</sup> <http://nlp.stanford.edu/software/lex-parser.shtml>

<sup>\*6</sup> 例文は KJ コーパスから抽出した。

表2 動詞の一致誤り訂正システムにおける追加処理一覧

追加処理する項目	追加処理の詳細	例文
受動態	受動態の場合は過去分詞形となる動詞ではなく、その動詞に係る受動態助動詞 (auxpass) を一致の対象とする。	I <i>*were / was</i> impressed ...
関係詞	関係詞 (WDT) が主語と解析された場合は、その先行詞 (関係詞に先行する名詞) を一致対象の主語とする。	... power that <i>*make / makes</i> people happy.
並列句構造	主語が並列句構造であるかどうかは、主語の後ろに <i>and</i> もしくは <i>;</i> が来ているかどうかで判定する。並列句構造であると判定された場合、主語は複数形であるとみなす。	There <i>*is / are</i> a farm and many cows.
動名詞	主語が <i>-ing</i> で終わる場合、元々の品詞に関わらず、単数名詞とみなす。	Reading books <i>is</i> very good ...
数量を表す名詞	主語が <i>lot, many, more, most</i> と解析された場合は、元々の品詞に関わらず複数名詞とみなす。	There <i>*was / were</i> a lot of people.
主語の誤り	主語が単数形で、前3単語以内に数詞がある場合は主語を複数名詞とみなす。	There were two little <i>*garden / gardens</i> ...

検出精度を求めた結果、本システムの精度は Precision = 0.556, Recall = 0.723, F 値 = 0.628 であった。

### 3.5 複数の誤り訂正システムの出力の統合方法

複数の誤り訂正システムの出力を統合するために、本稿では3つの方法を用いて比較を行なった。1つ目は単純に各誤り訂正システムの出力結果をマージし、いずれかのシステムが訂正を行った場合に訂正する方法である。ここでは、統計的機械翻訳による全ての誤り訂正システムと他の訂正システムの訂正結果が異なる場合は全ての誤り訂正システムの訂正結果を優先する。マージの例を挙げると次のようになる。

学習者の作文: He are not in interested to gardening.  
 全ての誤り訂正: He are not ~~in~~<sub>NONE</sub> interested at garden-  
 ing.  
 前置詞訂正: He are not ~~in~~<sub>NONE</sub> interested in gardening.  
 動詞の一致訂正: He is not in interested to gardening.  
 マージ: He is not ~~in~~<sub>NONE</sub> interested at gardening.

学習者の文の“interested to”に対して、全ての誤り訂正システムは“interested at”と訂正を行ない、前置詞訂正システムは“interested in”に訂正している。このような場合は全ての誤り訂正システムの結果を優先して、“interested at”を最終的な出力とする。

2つ目と3つ目の統合方法はパイプライン方式で誤り訂正システムを連続して用いて誤り訂正を行なう。2つ目は、前置詞、動詞の一致の誤りの訂正を行なってから、全ての誤り訂正を行なう方法である。3つ目は、2つ目とは反対に全ての誤り訂正を行なってから、前置詞、動詞の一致の誤りの訂正を行なう。なお、2つ目、3つ目の方法で2回目の訂正を行なう際に、モデルの再学習は行なわない。

## 4. 複数の手法による誤り訂正の適用に関する実験

3.5節で示した複数の誤り訂正システムの出力の統合方法の違いによって誤り訂正の結果がどう変わるのかを調べるために実験を行なった。フレーズベース統計的機械翻訳を

用いた全ての誤り訂正システムでは、Moses 2010-08-13<sup>\*7</sup>をデコーダ、GIZA++ 1.0.5<sup>\*8</sup>をアライメントのツールとして利用した。フレーズ抽出は grow-diag-final-and [28] を用いた。言語モデルは 3-gram を用いた。機械翻訳で利用されている最適化手法である MERT [26] は今回使用しなかった。前置詞誤り訂正に用いる最大エントロピーモデルのツールとして、Maximum Entropy Modeling Toolkit<sup>\*9</sup>をデフォルトパラメータで使用した。素性の抽出には、Stanford parser 2.0.2 の品詞解析・係り受け解析の結果を用いた。また、Web N-gram の計算には、Google N-gram [3] と SSGNC 0.4.6.<sup>\*10</sup>を用いた。

### 4.1 実験に使用したデータ

統計的機械翻訳による誤り訂正システムの学習には、クローリングを行ない獲得した2010年12月までの言語学習 SNS Lang-8<sup>\*11</sup>のデータを用いた。Lang-8 は学習者同士で相互に作文を添削しあうサイトであり、学習者の書いた文とその文に対してネイティブが添削を行なった文が対になった大規模なデータを手に入れることができる [25], [32]。

統計的機械翻訳による誤り訂正システムの翻訳モデル、言語モデルのトレーニングデータには、Lang-8 の日本人英語学習者の書いた作文とその添削後の文を用いた。509,116 文対の作文があったが、文が大きく添削されている場合は機械翻訳でアライメントがとりにくく精度を下げる要因となる。そこで、実験には動的計画法でアライメントをとり、単語の挿入数、削除数ともに5以下のものだけに限定し、391,699 文対を使用した。

前置詞誤り訂正システムの学習には、タグが付けられたコーパスが必要であったため、KJ コーパスを使用した。使用した KJ コーパスは 170 エッセイ、2411 文である。学習は前置詞以外の誤りも含めたまま行なった。

<sup>\*7</sup> <http://http://www.statmt.org/moses/>

<sup>\*8</sup> <http://code.google.com/p/giza-pp/>

<sup>\*9</sup> <https://github.com/lzhang10/maxent>

<sup>\*10</sup> <http://code.google.com/p/ssgnc/>

<sup>\*11</sup> <http://lang-8.com>

テストデータには KJ コーパスを用いた。前置詞誤りシステムの学習にも KJ コーパスを使用するため、KJ コーパス全体で 5-fold cross validation を行ない、トレーニングデータとテストデータに分割して実験を行なった。

#### 4.2 評価尺度

評価尺度として、単語単位による再現率、適合率および F 値を用いた。再現率、適合率、F 値は以下のように定義する。

$$\text{再現率} = \frac{tp}{tp + fn}, \quad \text{適合率} = \frac{tp}{tp + fp}, \quad (2)$$

$$F \text{ 値} = \frac{2 \times \text{再現率} \times \text{適合率}}{\text{再現率} + \text{適合率}} \quad (3)$$

$tp$  (true positive) はシステムが訂正を行ない正解だった箇所、 $fp$  (false positive) はシステムが訂正を行なったが訂正する必要がなかった箇所もしくは訂正が必要だったがシステムが訂正候補を間違えた箇所、 $fn$  (false negative) はシステムは訂正を行なわなかったが訂正が必要だった箇所である。また、F 値は再現率と適合率の調和平均である。

#### 4.3 実験結果

全ての訂正を ALL、前置詞の訂正を PREP、動詞の一致の訂正を VAGR と表記する。表 3 に各誤り訂正システムの出力の実験結果とマージを行なった出力の実験結果を示す。各誤り訂正システムの出力結果を見ると、ALL、VAGR の適合率に比べ PREP の適合率が低い値となった\*12。マージを行なうことで再現率を上げることに成功しており、F 値は全ての出力結果をマージした場合が最も高かった。

表 4 に前置詞、動詞の一致の誤り訂正を行なったあと、その出力に対して全ての誤りを訂正した結果を示す。比較のために、全ての訂正を行なったあと、もう一度全ての訂正を行なった結果も示し、“ALL → ALL” で表記する。“ALL → ALL” でも再現率、F 値は ALL の場合よりも上がった。しかし、ALL を 2 度繰り返すよりも、PREP、VAGR といった別の手法で訂正してから ALL で訂正を行なう方が再現率、F 値は高くなった。

表 5 に全ての誤り訂正を行なったあと、その出力に対して前置詞、動詞の一致の誤りの訂正を行なった結果を示す。“ALL → PREP + VAGR” は“ALL → PREP”と“ALL → VAGR”の出力結果をマージした結果を示す。“ALL → ALL”よりも ALL で訂正したあとに PREP、VAGR の訂正を行なったほうが再現率、F 値は高くなった。

### 5. 考察

4 節で行なった実験の結果を見ると、全ての結果を比べた際一番 F 値が高かったのは各誤り訂正システムの出力結

\*12 全ての誤りを対象として評価を行なっているため、再現率に関しては PREP、VAGR の値は低くなる。

表 3 各訂正システムの出力をマージした場合の結果

	再現率	適合率	F 値
ALL	0.171	0.331	0.226
PREP	0.014	0.224	0.026
VAGR	0.014	<b>0.426</b>	0.027
ALL + PREP	0.181	0.319	0.231
ALL + VAGR	0.182	0.325	0.234
ALL + PREP + VAGR	<b>0.192</b>	0.316	<b>0.239</b>

表 4 前置詞、動詞の一致誤りを訂正したあと全ての誤り訂正を行なった結果

	再現率	適合率	F 値
ALL	0.171	0.331	0.226
ALL → ALL	0.174	0.330	0.228
PREP → ALL	0.179	0.317	0.229
VAGR → ALL	0.179	<b>0.335</b>	0.233
PREP + VAGR → ALL	<b>0.187</b>	0.321	<b>0.236</b>

表 5 全ての誤り訂正を行なった後、前置詞、動詞の一致の誤りを訂正した結果

	再現率	適合率	F 値
ALL	0.171	<b>0.331</b>	0.226
ALL → ALL	0.174	0.330	0.228
ALL → PREP	0.181	0.317	0.230
ALL → VAGR	0.178	0.328	0.231
ALL → PREP + VAGR	<b>0.188</b>	0.315	<b>0.236</b>

果をマージした“ALL + PREP + VAGR”であった。これは、ALL で訂正が難しい前置詞、動詞の一致の誤りが訂正され再現率が上がったためである。前置詞、動詞の一致を訂正したあとに全ての誤り訂正する場合、全て訂正を行なったから前置詞、動詞の一致を訂正した場合も F 値は向上したが、出力結果をマージした場合には及ばなかった。

“ALL + PREP + VAGR”と“PREP + VAGR → ALL”の 2 つを比べると、再現率は“ALL + PREP + VAGR”の方が高い一方で、適合率は“PREP + VAGR → ALL”の方が高くなっている。この 2 つの実際の出力を分析した結果、各誤り訂正システムの出力結果の false positive が影響していることがわかった。“PREP + VAGR → ALL”で再現率が低くなっている理由は、“ALL”単体では訂正できたものが“PREP”と“VAGR”で誤った単語に変換したことにより、その周辺で“ALL”によって訂正できなくなる箇所 (false negative) が増えたためである。一方で、“PREP + VAGR → ALL”で適合率が高くなっているのは、“PREP”や“VAGR”で元から正しい単語 (true negative) を誤ったものに変換してしまった場合でも、“ALL”で訂正することで元の正しい単語に訂正し直すことができたためである。

表 6 に例を示す。1 つ目の例は前置詞訂正が誤ってしまったため、ALL でも訂正できなくなった例である。この

表6 “ALL + PREP + VAGR” と “PREP + VAGR → ALL” の出力結果例

学習者の文	then he tried to across the road where i was driving.
正解	then he tried to cross the road where i was driving.
ALL	then he tried to cross the road, where i was driving.
PREP	then he tried to <sub>NONE</sub> across the road where i was driving.
ALL + PREP + VAGR	then he tried to <sub>NONE</sub> cross the road, where i was driving.
PREP + VAGR → ALL	then he tried to <sub>NONE</sub> across the road, where i was driving.
学習者の文	trees grow up too.
正解	trees grow up too.
ALL	trees grow up too.
VAGR	trees grows up too.
ALL + PREP + VAGR	trees grows up too.
PREP + VAGR → ALL	trees grow up too.

例では、“tried to across”と学習者が書いているのに対して、正解は“tried to cross”である。しかしながら、PREPで訂正を行なった結果“tried across”と、“to”を削除してしまっている。その結果、ALLで訂正を行なっても“across”を“cross”に訂正できなかった。2つ目と3つ目の例はPREP、VAGRの訂正を行なって元から正しかった単語を誤った単語に変換してしまったが、ALLによって正しい元の単語に戻すことができた例である。2つ目では“trees grow”と学習者が正しく書いているのに対して、VAGRでは“trees grows”と訂正しているが、ALLの訂正を行なうことにより“trees grow”と元の正しい単語に戻している。

出力結果をマージする場合でも異なる手法を用いて連続で訂正を行なう場合でも基本的にALLの結果に比べて、再現率が上がって、適合率が下がる傾向があった。ひとつだけ例外があり、“VAGR → ALL”の場合のみ再現率、適合率ともにALLよりも高い値となった。これは、ALLで訂正できないものをVAGRで訂正できるため再現率が上がり、VAGR自体適合率が高いのに加え表6の2つ目のような例をALLにより訂正できるため、適合率がALLやマージした場合よりも高くなったと考えられる。このことから、パイプラインで2つ以上のシステムを繰り返して用いる場合は、1つ目に用いる手法がルールベースのような適合率が高い手法であれば、最終的な出力でも適合率、再現率が高くなり効果があると言える。また、今回の実験ではパイプラインで2つ目のシステムでモデルの再学習を行わなかったが、1度誤り訂正を行なうことで誤りの分布が変わっていることが予想される。そのため、1度誤り訂正を行なった後、再学習を行なうことで性能の改善ができると考えられる。

## 6. おわりに

本稿では英語学習者の全ての文法誤り訂正タスクに取り組んだ。これまでの研究では、単一のシステムを用いて誤

りの種類を数種類に限定した訂正タスクで評価していた。しかし、誤りの種類によってはルールベースの手法が効果的であったり、大規模データを用いた統計的な手法が効果的であったりするため、単一のシステムで全ての誤りに対応することには限界があった。

そこで、我々は全ての誤りを対象に、誤りの種類に応じた訂正システムを複数構築し、それらの出力を統合する手法について考察した。複数の手法を組み合わせることで異なった種類の誤りに対応できるようになり、単一のシステムで誤り訂正を行なう場合と比較して、再現率が向上することを確認した。また、各システムの出力結果を統合して最終出力を得る場合と、単一のシステムで誤りを訂正してから異なる誤り訂正システムで訂正を行なう場合とを比較し、前者が最もF値が高いことを示した。さらに、複数のシステムを段階的に適用する場合、ルールベースのような適合率の高いシステムを前処理として実行することで、再現率と適合率の両方を同時に向上させることができることも分かった。

今回は、全ての誤り・前置詞・動詞の一致に対する3種類の訂正システムを用いたが、今後は冠詞誤りや動詞の時制などの他の誤りの訂正システムも加えて実験を行ないたい。また、パイプラインで誤り訂正システムを用いる際にモデルの再学習を行なった場合の違いについても実験を行いたいと考えている。

## 参考文献

- [1] 安藤貞雄：現代英文法講義，開拓社（2005）。
- [2] Berger, A. L., Pietra, V. J. D. and Pietra, S. A. D.: A Maximum Entropy Approach to Natural Language Processing, *Computational Linguistics*, Vol. 22, No. 1, pp. 39–71 (1996).
- [3] Brants, T. and Franz, A.: Web 1T 5-gram corpus version 1.1, *Linguistic Data Consortium* (2006).
- [4] Brockett, C., Dolan, W. B. and Gamon, M.: Correcting ESL Errors Using Phrasal SMT Techniques, *Proceedings of COLING-ACL*, pp. 249–256 (2006).

- [5] Burstein, J., Chodorow, M. and Leacock, C.: Automated Essay Evaluation: the Criterion Online Writing Service, *AI Mag.*, Vol. 25, No. 3, pp. 27–36 (2004).
- [6] Chodorow, M., Gamon, M. and Tetreault, J.: The Utility of Article and Preposition Error Correction Systems for English Language Learners: Feedback and Assessment, *Language Testing*, Vol. 27, No. 3, pp. 419–436 (2010).
- [7] Dahlmeier, D. and Ng, H. T.: Grammatical Error Correction with Alternating Structure Optimization, *Proceedings of ACL-HLT*, pp. 915–923 (2011).
- [8] De Felice, R. and Pulman, S. G.: Automatically Acquiring Models of Preposition Use, *Proceedings of the 4th ACL-SIGSEM Workshop on Prepositions*, pp. 45–50 (2007).
- [9] De Felice, R. and Pulman, S. G.: A Classifier-Based Approach to Preposition and Determiner Error Correction in L2 English, *Proceedings of COLING*, pp. 169–176 (2008).
- [10] Eeg-olofsson, J. and Knutsson, O.: Automatic Grammar Checking for Second Language Learners - the Use of Prepositions, *In NoDaRiDa* (2003).
- [11] Ehsan, N. and Faili, H.: Statistical Machine Translation as a Grammar Checker for Persian Language, *Proceedings of the Sixth International Multi-Conference on Computing in the Global Information Technology*, pp. 20–26 (2011).
- [12] Gamon, M.: Using Mostly Native Data to Correct Errors in Learners' Writing: A Meta-Classifier Approach, *Proceedings of HLT-NAACL*, pp. 163–171 (2010).
- [13] Gamon, M., Gao, J., Brockett, C. and Klementiev, A.: Using Contextual Speller Techniques and Language Modeling for ESL Error Correction, *Proceedings of IJCNLP*, pp. 449–456 (2008).
- [14] Gamon, M., Leacock, C., Brockett, C., Gao, J., Belenko, D. and Klementiev, A.: Using Statistical Techniques and Web Search to Correct ESL Errors, *Journal of the Computer Assisted Language Instruction Consortium*, Vol. 26, No. 3, pp. 491–511 (2009).
- [15] Heidorn, G. E., Jensen, K., Miller, L. A., Byrd, R. J. and Chodorow, M. S.: The EPISTLE text-critiquing system, *IBM Syst. J.*, Vol. 21, No. 3, pp. 305–326 (1982).
- [16] Hermet, M. and Désilets, A.: Using First and Second Language Models to Correct Preposition Errors in Second Language Authoring, *Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 64–72 (2008).
- [17] Izumi, E., Uchimoto, K. and Isahara, H.: The Overview of the SST Speech Corpus of Japanese Learner English and Evaluation Through the Experiment on Automatic Detection of Learners' Errors, *Proceedings of LREC* (2004).
- [18] Izumi, E., Uchimoto, K., Saiga, T., Supnithi, T. and Isahara, H.: Automatic Error Detection in the Japanese Learners' English Spoken Data, *Proceedings of ACL*, pp. 145–148 (2003).
- [19] Jensen, K., Heidorn, G. E., Miller, L. A. and Ravin, Y.: Parse Fitting and Prose Fixing: Getting a Hold on Ill-Formedness, *Comput. Linguist.*, Vol. 9, No. 3-4, pp. 147–160 (1983).
- [20] Koehn, P., Och, F. J. and Marcu, D.: Statistical Phrase-Based Translation, *Proceedings of HLT-NAACL*, pp. 48–54 (2003).
- [21] Leacock, C. and Chodorow, M.: Automated grammatical error detection, *Automated Essay Scoring: A Cross-Disciplinary Perspective* (Sherrnis, M. and Burstein, J., eds.), Lawrence Erlbaum Associates Mahwah, NJ, pp. 195–207 (2003).
- [22] Leacock, C., Chodorow, M., Gamon, M. and Tetreault, J.: *Automated Grammatical Error Detection for Language Learners*, Morgan & Claypool (2010).
- [23] Lee, J. and Seneff, S.: Correcting Misuse of Verb Forms, *Proceedings of ACL-HLT*, pp. 174–182 (2008).
- [24] Liu, X., Han, B. and Zhou, M.: Correcting Verb Selection Errors for ESL with the Perceptron, *Proceedings of CICLing*, pp. 411–423 (2011).
- [25] Mizumoto, T., Komachi, M., Nagata, M. and Matsumoto, Y.: Mining Revision Log of Language Learning SNS for Automated Japanese Error Correction of Second Language Learners, *Proceedings of IJCNLP*, pp. 147–155 (2011).
- [26] Och, F. J.: Minimum Error Rate Training in Statistical Machine Translation, *Proceedings of ACL*, pp. 160–167 (2003).
- [27] Och, F. J. and Ney, H.: Discriminative Training and Maximum Entropy Models for Statistical Machine Translation, *Proceedings of ACL*, pp. 295–302 (2002).
- [28] Och, F. J. and Ney, H.: A Systematic Comparison of Various Statistical Alignment Models, *Computational Linguistics*, Vol. 29, No. 1, pp. 19–51 (2003).
- [29] Park, Y. A. and Levy, R.: Automated Whole Sentence Grammar Correction Using a Noisy Channel Model, *Proceedings of ACL*, pp. 934–944 (2011).
- [30] Rozovskaya, A. and Roth, D.: Algorithm Selection and Model Adaptation for ESL Correction Tasks, *Proceedings of ACL*, pp. 924–933 (2011).
- [31] Seo, H., Lee, J., Kim, S., Lee, K., Kang, S. and Lee, G. G.: A Meta Learning Approach to Grammatical Error Correction, *Proceedings of ACL*, pp. 328–332 (2012).
- [32] Tajiri, T., Komachi, M. and Matsumoto, Y.: Tense and Aspect Error Correction for ESL Learners Using Global Context, *Proceedings of ACL*, pp. 198–202 (2012).
- [33] Tetreault, J., Foster, J. and Chodorow, M.: Using Parse Features for Preposition Selection and Error Detection, *Proceedings of ACL*, pp. 353–358 (2010).