文法誤り訂正のための疑似誤り生成による ラベルなしコーパスの利用

澤井 裕一郎　　進藤 裕之　　松本 裕治
奈良先端科学技術大学院大学　情報科学研究科
{sawai.yuichiro.sn0, shindo, matsu}@is.naist.jp

1 はじめに

文法誤り訂正は，文法的な誤りを含む可能性のある文を，元の意味を保ちつつ文法的に正しい文に変換するタスクである．この文法誤り訂正タスクに関して，大別して，分類器ベースの手法 [8] と，機械翻訳ベースの手法 [4] が提案されてきた．本稿では，文法誤り訂正を系列ラベリングとみなし，双方向リカレントニューラルネットワークによる分類器ベースの手法を用いて，訂正対象を冠詞と前置詞のみに限定した文法誤り訂正タスクに取り組む．分類器ベースの手法の利点の一つとして，予測スコアに対して閾値を設けることにより，適合率と再現率のバランスを容易に調整できることが挙げられる．実際，文法誤り訂正のコンペティションである CoNLL-2014 Shared Task の評価指標には，再現率よりも適合率を重視する $F_{0.5}$ 値が用いられている．また，双方向リカレントニューラルネットワークを用いることにより，表層形の異なる単語間でも分散表現を通じて素性共有ができること，入力文全体を考慮した文法誤り訂正が行えることが期待できる．

文法誤り訂正においては，多種多様な語の用例情報が必要であるため，大規模ラベルなしコーパスから学習した言語モデルが高精度化に寄与することが知られている [4]．しかし，本稿で用いる系列ラベリングモデルにおいては，大規模ラベルなしコーパスを利用する方法は自明ではない．そこで，本稿では，ラベルなしコーパスに対して人工的に疑似誤りを導入し，それを擬似的な訓練データとして用いることで分類器を学習する手法を検証する．

疑似誤りの生成方法としては，訂正前-訂正後文ペアからなるパラレルコーパスから得た混同行列に基づく手法が従来用いられてきた [8]．この手法では，混同行列が定める確率に従って，ラベルなしコーパス内の各単語をランダムに置換・削除・挿入して疑似誤りを生成する．しかし，この手法では前後の文脈を考慮して疑似誤りを生成することが困難である．特に，冠詞・前置詞の挿入誤りの分布は前後の文脈に大きく影響されるため，これらの自然な疑似誤りを作り出すことは難しいと考えられる．そこで，本稿では，訂正前-訂正後文ペアのパラレルコーパスを逆方向に用いて，文法的に正しい文から文法的に誤った文を生成する疑似誤り生成モデルを学習する．そして，このモデルを用いて生成した訓練データから誤り訂正モデルを学習し，その性能を検証する．

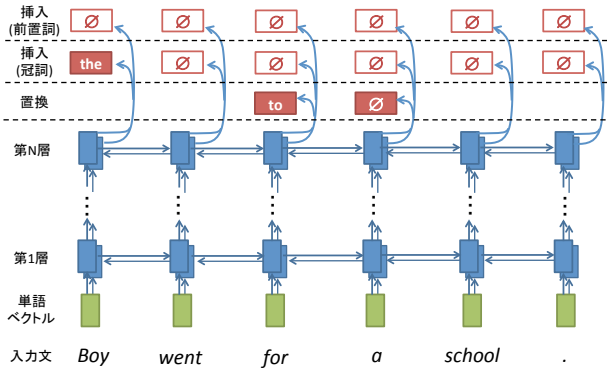

図 1: 誤り訂正ラベル付けモデルの概略図（入力文”Boy went for a school.”を，ラベル付けにより文”The boy went to school.”に訂正する例．赤に白抜きのラベルは入力文を改変する訂正ラベルを表す.）

本稿では，まず，文法誤りの事例の多くの割合を占める冠詞・前置詞を訂正対象として，混同行列を用いて疑似誤りを生成する手法と，パラレルコーパスから学習したモデルにより疑似誤りを生成する手法の比較実験を行う．また，訂正対象を冠詞に限定した上で，より大規模なラベルなしコーパスを用いて，混同行列による擬似誤り生成手法により学習した文法誤り訂正モデルの評価実験を行う．

2 双方向リカレントニューラルネットワークによる誤り訂正モデル

本稿では，文法誤り訂正のタスクを，入力文中の各トークンに対して置換（削除）・挿入のラベル付けを行う系列ラベリング問題として捉える．系列ラベリングモデルとしては，双方向の多層リカレントニューラルネットワークを用いる．モデルの概要を図 1 に示す．

まず，モデルは入力として語彙数次元の one-hot ベクトルとして表現されたトークン列 $\{\boldsymbol{x}_t\}_{t=1}^{T}$ を受け取る．各トークン $\boldsymbol{x}_t$ はパラメータ行列 $\boldsymbol{W}_{emb}$ によって単語ベクトル $\boldsymbol{h}_{0,t}$ に変換される．

$$\boldsymbol{h}_{0,t} = \boldsymbol{W}_{emb}\boldsymbol{x}_t$$

得られた単語ベクトルの系列を，双方向の多層リカレントニューラルネットワークにより組み合わせる．隠れ状態ベクトルの計算には，リカレントニューラルネットワークの一種である Long Short-Term Memory (LSTM) [3] を用いる．第 n 層の時刻 t における順方向隠れ状態ベクトル $\boldsymbol{h}^f_{n,t}$ は，同じ層の前の時刻 t における隠れ状態ベクトル $\boldsymbol{h}^f_{n,t-1}$ と下層の時刻 t における隠れ状態ベクトル $\boldsymbol{h}^f_{n-1,t}$ を，関数 f_{LSTM} により組み合わせることで計算される．

$$\boldsymbol{h}_{n,t} = f_{LSTM}(\boldsymbol{h}_{n,t-1}, \boldsymbol{h}_{n-1,t})$$

第 n 層の時刻 t における順方向隠れ状態ベクトル $\boldsymbol{h}^b_{n,t}$ も同様にして計算される．

最終層の各時刻 t の隠れ状態ベクトル $\boldsymbol{h}^f_{N,t}$，$\boldsymbol{h}^b_{N,t}$ から，対応する入力トークン $\boldsymbol{x}_t$ の誤り訂正ラベルを予測する．誤り訂正ラベルの予測は，標準的な多クラス分類問題として定式化する．冠詞候補集合を $S_{art} = \{$a,the$\}$（冠詞”an”は”a”に正規化），前置詞候補集合を $S_{prep} = \{$in, of, on, for, to, at, about, with, from, by, into, during, as$\}$ とする．$\boldsymbol{x}_t$ が冠詞である場合，置換ラベルの候補集合 $S_{art} \cup \{\varnothing\}$ のうち 1 つを予測する．ただし，ラベル $\varnothing$ はトークン $\boldsymbol{x}_t$ を削除することを意味する．同様に，$\boldsymbol{x}_t$ が前置詞である場合，置換ラベルの候補集合 $S_{prep} \cup \{\varnothing\}$ のうち 1 つを予測する．$\boldsymbol{x}_t$ が冠詞でも前置詞でもない場合は，置換ラベルの予測は行わない．また，全ての時刻 t において，トークン $\boldsymbol{x}_t$ の直前に挿入するべき冠詞，前置詞を，それぞれ $S_{art} \cup \{\varnothing\}$，$S_{prep} \cup \{\varnothing\}$ の中から 1 つずつ予測する．ただし，ラベル $\varnothing$ は何も挿入しないことを意味する．

トークン $\boldsymbol{x}_t$ に対するラベル y への置換スコア $s^y_{t,SUB}$，挿入スコア $s^y_{t,INS}$ は，パラメータ $\boldsymbol{w}^y_{SUB}$，$\boldsymbol{w}^y_{INS}$ により，それぞれ以下の式に基づき計算される．ただし，$\oplus$ はベクトルの直和である．

$$\begin{aligned} s^y_{t,SUB} &= \boldsymbol{w}^y_{SUB}(\boldsymbol{h}^f_{N,t} \oplus \boldsymbol{h}^b_{N,t}) \\ s^y_{t,INS} &= \boldsymbol{w}^y_{INS}(\boldsymbol{h}^f_{N,t} \oplus \boldsymbol{h}^b_{N,t}) \end{aligned} \quad (1)$$

予測時には，訂正の起こりやすさを調整するため，無訂正ラベル（置換の場合は入力と同じ単語，挿入の場合は $\varnothing$）以外のラベルに対し，スコアにバイアス θ_{SUB}，θ_{INS} を足し合わせる．これらのバイアスは，ハイパーパラメータとして，開発データの評価値が最大となるように調整される．バイアスを加えた後のスコアが最大のラベルが予測ラベルとして出力される．

モデルの各パラメータは，訓練データ中の各予測事例に対する交差エントロピー誤差関数の和を，確率的勾配法で最小化することにより最適化する．

2.1 疑似誤りの生成手法

本稿では，ラベルなしコーパスに対して疑似誤りを生成し，これを擬似的な訓練データとして用いることで，誤り訂正モデルを学習する．本節では，疑似誤りの生成に必要な誤りの分布情報を訂正前-訂正後文ペアのパラレルコーパスから得る手法を述べる．

2.1.1 混同行列に基づく疑似誤りの生成手法

パラレルコーパスから得た混同行列が定める確率 $p(w_{\text{訂正前}}|w_{\text{訂正後}})$ に基づき，文中の各トークンに対しランダムに疑似誤りを生成する．例えば，節 3.2 の実験 2 で用いた冠詞の混同行列は，表 1 の通りである．ただし，”訂正後”が $\varnothing$ であるとは，疑似的な挿入誤りを生成することを意味し，”訂正前”が $\varnothing$ であるとは，疑似的な削除誤りを生成することを意味する．

混同行列をそのまま用いた場合の誤りの生成割合は非常に小さく，これが問題になる場合がある．先行研究では，誤りの生成確率を増加させる Error Inflation 法が有効であることが示唆されている [8]．Error Inflation 法では，疑似誤りを生成しない確率（$\varnothing \to \varnothing$，$a \to a$，等）に 1 以下の Inflation 係数を掛け合わせ，残りの確率質量を他の候補に比例配分することにより，疑似誤りを生成する確率を増加させる．本稿の実験でも，Inflation 係数を変化させることによって，誤りの生成確率を増減させる．

表 1: 冠詞の混同行列の例

訂正後＼訂正前	$\varnothing$	a	the
$\varnothing$	0.974	0.004	0.022
a	0.035	0.956	0.010
the	0.040	0.002	0.958

2.1.2 擬似誤り生成モデルによる手法

混同行列に基づく疑似誤りの生成手法では，前後の文脈によらず元の単語のみに依存する確率で疑似誤りが生成される．しかし，文脈を考慮して疑似誤りを生成した方が，より自然な疑似誤りを生成できると考えられる．例えば，冠詞が名詞句の直前以外に誤って挿入されることは稀である．また，前置詞は動詞の直後に誤って挿入されることが多い．

そこで，パラレルコーパスを逆方向に用いて誤り訂正モデルを学習することにより，文脈を考慮しつつ文法的に正しい文から文法的に誤った文を生成するモデルを学習する．疑似誤り生成時には，式 1 により定められるスコアをソフトマックス関数により正規化して得られる確率値に基づき，疑似誤りをランダムに生成する．また，混同行列における Error Inflation 法と同様，疑似誤りの生成割合を調整するために，スコアに足し合わせるバイアス θ を変化させて実験を行う．

3 実験

3.1 実験 1: 疑似誤り生成手法の比較

疑似誤り生成手法として，混同行列を用いる方法と，学習した疑似誤り生成モデルを用いる方法の比較実験を行う．冠詞と前置詞の訂正を対象として，比較的小規模なラベルなしコーパスに対して疑似誤りを生成し，誤り訂正モデルの訓練データの作成と，学習された誤り訂正モデルの評価を行う．

表 2: ラベルあり，ラベルなしコーパスと大きさ

コーパス		文数
ラベルあり	Lang-8	1.03 M
	NUCLE	57.15 K
	CoNLL-2013 Test	1.38 K
	CoNLL-2014 Test	1.31 K
ラベルなし	CommonCrawl	59.13 G
	Wikipedia	213.08 M
	Gigaword (nyt_eng)	80.53 M

3.1.1 実験設定

実験に関係するコーパスとそれぞれの大きさを表 2 にまとめる．混同行列と疑似誤り生成モデルの学習に必要なパラレルコーパスとして，Lang-8 コーパス [6] を用いた．疑似誤りを生成する対象のラベルなしコーパスとしては，Annotated English Gigaword [7] の New York Times (nyt_eng) セクションを用いた．実験 1 では，さらにその一部（100 万文）をラベルなしコーパスとして使用した．

実際の誤りの分布を得るために，パラレルコーパスに対して編集距離に基づくアラインメントをとり，訂正後の文中の各トークンに対して，訂正前の文を生成するために必要な置換・挿入ラベルを付けた．この際，冠詞，前置詞以外の誤りタイプを含む等の理由でラベル付けに失敗した文（全体の 42.4%）は使用しなかった．こうして得られた置換・挿入ラベルから，混同行列，疑似誤り生成モデルを学習した．混同行列を作る際，冠詞，前置詞の挿入誤り生成箇所は全ての単語の直前を候補とした．混同行列の Inflation 係数としては {1.0,0.8,0.6} を使用した．．疑似誤り生成モデルには，以下に記述する誤り訂正モデルと同じアーキテクチャ，学習方法を用いた．疑似誤り置換・挿入のスコアに足すバイアス θ として {+0.0, +3.0, +6.0, +9.0} を使用した．(大きい方がより多くの疑似誤りを生成する)

誤り訂正モデルのアーキテクチャとしては，語彙数 65,536，単語ベクトル・隠れ層の次元数 512 次元，4 層の双方向リカレントニューラルネットワークを用いた．モデルのパラメータは Adam [5] によりミニバッチ（サイズは 512 文）を利用した最適化を行った．学習中，勾配ベクトルの値が [-1.0, 1.0] の範囲に収まるようにクリッピングを行った．学習は 20 エポック行い，バイアス θ とモデルのエポック数は開発データ（CoNLL-2013 評価データ）での評価値が最大になるように選んだ．

誤り訂正モデルの評価には，CoNLL-2014 評価データを用いた．CoNLL-2014 の評価方法に倣い，システムによる訂正の集合の $F_{0.5}$ 値（適合率を重視）により評価した．訂正の集合は，CoNLL-2014 Shared Task の公式評価スクリプトである MaxMatch scorer[1] の出力により得た．ただし，評価対象は冠詞・前置詞の訂正のみに限定した．そのために，訂正の対象が冠詞，前置詞，その他のいずれであるかはルールにより判別した．

3.1.2 実験結果

表 3，表 4 にそれぞれ，混同行列により擬似誤りを生成した場合，疑似誤り生成モデルを使用した場合の結果を示す．評価データでは，混合行列による疑似誤り生成手法に比べて，モデルによる擬似誤り生成を行った場合の方が全体の訂正性能が高かった．特に，前置詞の訂正の性能が高いことがわかった．また，疑似誤り生成割合を増加させる（Inflation 係数を減少させる，バイアス θ を増加させる）ことによる性能の向上は両手法で見られた．開発データと評価データで一部傾向が異なるが，これは，冠詞・前置詞誤りの個数，挿入・削除・置換誤りの分布が両データで大きく異なることが原因であると考えられる．

3.2 実験 2: 大規模ラベルなしデータの利用

実験 1 に加えて，ラベルなしデータの量を増加させた場合の評価実験を行った．ただし，今回はデータの前処理にかかる時間の制約上，訂正の対象は冠詞に限定し，混同行列により擬似誤りを生成する手法のみで評価実験を行った．

3.2.1 実験設定

疑似誤りを生成する対象のラベルなしデータとして，Annotated Gigaword Corpus (nyt_eng) 全体を用いる．混同行列の作成には，NUCLE corpus [2] を用いた．33.3%の文は置換・挿入タグ付けに失敗したため，使用しなかった．また，実験 2 では，混同行列において挿入誤りの生成箇所を，名詞句の直前に限定した．そのために，句構造解析器の出力を元に名詞句の位置を同定し，混同行列の作成と疑似誤りの生成を行った．モデルのパラメータの学習，ハイパーパラメータの調整は実験 1 と同じ手順で行った．評価は冠詞の訂正のみを対象として行った．

3.2.2 実験結果

実験結果を表 5 に示す．Inflation 係数を 1.0 から少し減少させた際に最も良いスコアを達成した．比較対象として，現在 CoNLL-2014 Shared Task で最高精度を達成しているシステム（機械翻訳ベース）[4] の冠詞誤り訂正性能を表 6 に示す．冠詞誤り訂正に限定すると，同程度のスコアを達成している．なお，[4] はパラレルコーパスとして Lang-8 コーパス，大規模ラベルなしコーパスとして Common Crawl，Wikipedia コーパスを利用している．それに対して，我々の手法では，混同行列を得るために NUCLE コーパスの一部を利用し，ラベルなしコーパスとして Annotated Gigaword Corpus を用いており，[4] の設定に比べて小規模のコーパスのみを使用している．

表 3: 混同行列による擬似誤り生成を行った場合の結果

Inflation 係数	評価データ			開発データ		
	$F_{0.5}$ (冠詞)	$F_{0.5}$ (前置詞)	$F_{0.5}$ (全体)	$F_{0.5}$ (冠詞)	$F_{0.5}$ (前置詞)	$F_{0.5}$ (全体)
1.0	0.261	**0.096**	0.238	0.275	**0.110**	0.254
0.8	**0.287**	0.000	**0.266**	0.286	0.069	**0.331**
0.6	0.265	0.048	0.250	**0.326**	0.015	0.308

表 4: モデルによる擬似誤り生成を行った場合の結果

θ	評価データ			開発データ		
	$F_{0.5}$ (冠詞)	$F_{0.5}$ (前置詞)	$F_{0.5}$ (全体)	$F_{0.5}$ (冠詞)	$F_{0.5}$ (前置詞)	$F_{0.5}$ (全体)
+0.0	0.200	0.000	0.186	0.276	0.016	0.255
+3.0	0.256	0.114	0.234	**0.314**	0.059	**0.283**
+6.0	**0.290**	**0.188**	**0.275**	0.305	**0.110**	0.282
+9.0	0.239	0.080	0.225	0.260	0.070	0.245

表 5: 冠詞誤り訂正実験の結果

Infl.	評価データ			開発データ		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$
1.0	0.44	0.58	0.46	0.50	0.46	0.49
0.8	**0.49**	0.64	**0.51**	**0.53**	0.45	**0.51**
0.6	0.45	**0.69**	0.48	0.51	**0.52**	**0.51**

表 6: CoNLL-2014 Shared Task の State-of-the-art システムの冠詞誤り訂正性能

P	R	$F_{0.5}$
0.55	0.42	0.52

4 おわりに

本稿では，大規模ラベルなしコーパスを利用するための疑似誤りの生成手法として，混同行列を用いる手法と，パラレルコーパスから疑似誤り生成モデルを学習する手法の比較実験を行った．評価データにおいては，疑似誤り生成モデルを用いた手法の方が性能が高く，特に前置詞訂正の性能が高いことがわかった．今後，より大規模なラベルなしコーパスを用いて，比較実験を行う予定である．

また，混同行列を用いて生成した疑似誤りデータを利用することにより，冠詞誤り訂正に関しては，現在の State-of-the-art のシステムと同程度の性能を得られることがわかった．今後，同様の枠組みを利用して，名詞，動詞などの他の誤りタイプの訂正を行えるように拡張する予定である．

参考文献

[1] 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, 2012.

[2] Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. Building a large annotated corpus of learner english: The nus corpus of learner english. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 22–31, 2013.

[3] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory, 1997.

[4] Marcin Junczys-Dowmunt and Roman Grundkiewicz. Phrase-based machine translation is state-of-the-art for automatic grammatical error correction. In *Proceedings of EMNLP 2016*.

[5] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[6] Tomoya Mizumoto, Yuta Hayashibe, Mamoru Komachi, Masaaki Nagata, and Matsumoto Yuji. The effect of learner corpus size in grammatical error correction of esl writings. In *Proceedings of COLING 2012*, pp. 863–872, 2012.

[7] Courtney Napoles, Matthew Gormley, and Benjamin Van Durme. Annotated gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pp. 95–100, 2012.

[8] Alla Rozovskaya and Dan Roth. Building a state-of-the-art grammatical error correction system. *Transactions of the Association for Computational Linguistics*, pp. 419–434, 2014.