

根拠を明示するニューラル文法誤り訂正

小山田創哲^{†,††,†††} 兼村 厚範^{††} 石井 信^{†††,††††}

† 株式会社リクルートテクノロジーズ 〒100-6640 東京都千代田区丸の内 1-9-2

†† 産業技術総合研究所 〒135-0064 東京都江東区青海 2-3-26

††† 京都大学 大学院情報学研究科 〒606-8501 京都府京都市左京区吉田本町 36-1

†††† ATR 脳情報通信総合研究所 〒619-0288 京都府相楽郡精華町光台 2-2-2

E-mail: †koyamada-s@sys.i.kyoto-u.ac.jp, ††atsu-kan@aist.go.jp, †††ishii@i.kyoto-u.ac.jp

あらまし 非ネイティブ話者の語学学習においては、機械による文法誤り訂正によって学習効率を改善することが期待されている。近年隆盛のニューラルネットワークは、文法誤り訂正タスクでも用いられているが、その誤り訂正の根拠が必ずしも自明でないという欠点がある。もし言語学習者に誤りの種類を提示することができれば、学習効率が改善すると期待される。そこで我々は、誤りの種類も陽に推定することで、誤りを訂正すると同時にその誤りの種類を提示するニューラル文法誤り訂正モデルを提案する。具体的には、ネットワーク中に誤りの種類を示すノードを追加し、その出力を教師有りで学習する。提案モデルを CoNLL2014 Shared Task データセットにおいて評価した結果、誤り訂正の性能が見劣りせず、かつ誤りの種類も一定の精度で推定できることが分かった。

キーワード 自然言語処理, 文法誤り訂正, リカレントニューラルネットワーク

1. はじめに

文章を通して意図を伝えるためには、文が正しい文法にのって構成されていなければならないが、第二言語において正しい文法での作文はしばしば困難であり、正しい文法での作文を効率的に習得するためには、適切なガイドが必要である。習得ガイドの一つとして、学習者が実際に書いた文章に対し、適切なフィードバックが与えられることが重要だとされている [2, 11]。しかし、適切なフィードバックは現状、その言語を使いこなせる人間（第一言語とするか第二言語習得者として既に熟練した人間）による校閲が必要であり、ヒューマンコストがボトルネックとなる。このボトルネックを解決するため、機械による自動での文法誤り訂正 (grammatical error correction; GEC) 技術の開発が期待されている。

文法誤り訂正の主要なアプローチの一つは統計的機械翻訳 (statistical machine translation; SMT) に基づくものである。文法誤り訂正を、誤りのある文から、正しい文への翻訳だと考えれば、SMT の手法は文法誤り訂正でも活用が可能である。実際、代表的なベンチマークデータセットである CoNLL-2014 Shared Task データセット [6] において、現在の state-of-the-art としてのベンチマークは SMT を使った手法により達成されている [4]。

SMT の分野では近年、ニューラルネットワークを用いた encoder-decoder 型のモデルが成功を収めている [9, 1]。その自然な流れとして、文法誤り訂正のモデルとしてもこのニューラルネットワークを用いた手法 (ニューラル文法誤り訂正モデル) が提案され始めている [14]。Xie らの研究 [14] は、最初のニューラル文法誤り訂正モデルであり、キャラクタベースのモデルにより CoNLL-2014 Shared Task データセットにおいて高い性

能を示しており、文法誤り訂正の分野においてもニューラルな手法が有望であることを示している。なお、state-of-the-art は非ニューラルな手法であり [4]、ニューラルな手法は改善の余地が大きいと期待される。

これまでのニューラル文法誤り訂正は、SMT ベースの文法誤り訂正手法として有望である一方で、単に修正案となる文を出力するだけに留まり、誤り訂正の理由 (根拠) が必ずしも自明でなく “ブラックボックス” 性が高いという欠点がある。これは次の 2 つの観点で問題がある。

- (1) (信頼性) モデルの提示した修正案が正解であったとしても、正しく学習した結果としてその修正案を提示したのか、それとも偶然の結果なのかを判断する根拠に乏しい。
- (2) (学習者支援) 第二言語学習者にとってのフィードバックとして、単なる正しい修正案だけでは乏しい。

モデルの学習が正しく行われているかどうかは、通常何かしらの指標 (accuracy, precision, recall, F -score 等) を用いてテストデータで定量的に確認されるが、定量的な指標を盲目的に信頼することの問題点も指摘されている [8]。また、第二言語学習者がより効率的に言語を習得できるように、詳細かつ示唆に富むフィードバックを出力するためには、正しい修正案の提示だけに留まるのではなく、人間による添削においてコメントがしばしば添付されるのと同じように、文法誤り訂正の理由 (根拠) も、機械によって明示されるのが望ましい。

そこで本研究では、根拠とする誤り訂正の理由も隠れ変数として同時に推定し、出力するニューラル文法誤り訂正モデルを提案する。これにより、モデルが何を根拠としてその訂正をしたのかの方がより明らかとなり、また、第二言語学習者には、誤り訂正の結果だけでなく、その理由 (根拠) をフィードバックすることが可能となる。より詳細かつ信頼性のあるフィードバッ

クを与えることで、第二言語学習者の能力開発に寄与することが期待できる。

我々の提案手法の利点は次の通りである。

- (1) 誤り理由も同時に推定することで、モデルの提示する修正理由を理解することができ、信頼性が高く、第二言語学習者にとってより理解しやすい。
- (2) これまで個別のモデルで対処されてきた誤り発見 (error detection) と誤り訂正 (error correction) を同時に一つのモデルで学習が可能である。
- (3) データセット中に誤り理由の教師データが少なくても、半教師有り学習が可能である。

2. 関連手法

この節では、提案手法の背景となる関連手法について説明を行う。

2.1 Encoder-Decoder モデル

本研究の提案手法は、encoder-decoder に基づくものである。SMT の分野では、encoder-decoder モデルの他手法に対する有効性が確認されている [9, 1]。Encoder-decoder モデルは、再帰ニューラルネットワーク (recurrent neural network; RNN) を利用し、入力された系列データを別の系列データに写像するモデルである。SMT の他にも、同じく系列から系列への写像を学習するタスク、例えば対話への応用 [12] や本論文で対象とする文法誤り訂正タスクへの適用 [14] が報告されている。

Encoder-decoder モデルは、ベクトルの系列 $x = (x_1, \dots, x_T)$ を入力として、ベクトルの系列 $y = (y_1, \dots, y_{T'})$ を出力する写像を学習する:

$$p(y|x) = \prod_{t'=1}^{T'} p(y_{t'} | x, y_1, \dots, y_{t'-1}) \quad (1)$$

この写像は、encoder と decoder という二つの系列モデルを組み合わせることで実装される。

Encoder は、入力ベクトルの系列 $x = (x_1, \dots, x_T)$ から中間表現ベクトルの系列 $h = (h_1, \dots, h_T)$ を生成する:

$$p(h|x) = \prod_{t=1}^T p(h_t | x_1, \dots, x_t, h_{t-1}) \quad (2)$$

より具体的には、RNN f により、 $h_t = f(h_{t-1}, x_t)$ を逐次的に $t = 1, \dots, T$ において計算し (ただし、 $h_0 = 0$)、これを $p(h_t | x_1, \dots, x_t, h_{t-1})$ からの出力とする。 L 個の RNN を階層的に用いる場合には $h_t^l = f(h_{t-1}^l, h_t^{l-1})$ ($l \in \{0, \dots, L\}$) とする。ただし、 $h_t^0 = x_t$ 、 $h_t = h_t^L$ であるとする。

Decoder は encoder によって得られた中間表現ベクトルの系列 $h = (h_1, \dots, h_T)$ から、より decoder が扱いやすいように生成されたコンテキスト $c_{t'} = c_{t'}(h_1, \dots, h_T)$ を用いて、出力のベクトル系列 $y = (y_1, \dots, y_{T'})$ を生成する:

$$p(y|h) = \prod_{t'=1}^{T'} p(y_{t'} | y_1, \dots, y_{t'-1}, c_{t'}) \quad (3)$$

Decoder は、RNN f' と、隠れ表現から出力を予測する関数 g によって実装される。RNN f' は、 $h_{t'}^l = f'(y_{t'-1}, h_{t'-1}^l, c_{t'})$ によって隠れ状態 $h_{t'}^l$ を $t' = 1, \dots, T'$ に対し逐次的に出力し、 g

は $y_{t'} = g(h_{t'}^l, c_{t'})$ を $t' = 1, \dots, T'$ に対し逐次的に出力することでベクトルの系列 $y = (y_1, \dots, y_{T'})$ を予測する。 g の具体的な実装としては、例えば単純な多層ニューラルネットワークが用いられる。Decoder を多層にする場合にも、encoder のときと同じく、 $h_{t'}^l = f'(h_{t'-1}^l, h_{t'}^l)$ (ただし、 $h_{t'}^0 = y_{t'}$ 、 $h_{t'}^l = h_{t'}^L$) とする。Encoder-decoder モデルを提案した [9] では、よりシンプルに、 $c_{t'}$ は t' に依存しない値 h_T によって $c_{t'} = c = h_T$ と定義された。また、 $h_{t'}^l = f'(y_{t'}, h_{t'-1}^l)$ ($h_0^l = c$) と計算された。一方、[1] では、 $c_{t'}$ を以下で説明する attention メカニズムを用いて求めることで、 $x = (x_1, \dots, x_T)$ の情報を固定長のベクトル c に制限することなく、 t' に依存して活用することを可能にした。

Attention メカニズムでは、コンテキスト $c_{t'} = c_{t'}(h)$ の定義を、 $y_{t'}$ の予測において x_t (h_t) がどれだけ貢献すべきかを表すスカラー値 $\alpha_{t,t'}$ を導入し、 h_t の $\alpha_{t,t'}$ による重み付き平均とした:

$$c_{t'} = \sum_{t=1}^T \alpha_{t,t'} h_t \quad (4)$$

$\alpha_{t,t'}$ は次のような alignment モデル a によって定められる:

$$\alpha_{t,t'} = \frac{\exp(e_{t,t'})}{\sum_{t=1}^T \exp(e_{t,t'})} \quad (5)$$

$$e_{t,t'} = a(h_t, h_{t'-1}') \quad (6)$$

ここで a の実装としては、 $e_{t,t'}$ を多層ニューラルネットワークによって直接求めるもの、パラメータ行列 W によって $h_t^T W h_{t'-1}'$ と求めるもの、 h_t と $h_{t'-1}'$ を同じ距離空間に写してその内積 $\phi(h_t)^T \phi'(h_{t'-1}')$ によって定義するものなどがある。

2.2 複数の Attention を学習するモデル

重み $\alpha_{t,t'}$ は、 $(x_t, y_{t'})$ に対する関連度の指標であるが、タスクによっては関連度の自然な定義が一意に定まるとは限らず、むしろ、複数の指標を考える方が自然な場合がある。そうした場合、複数の attention を学習するようモデルを拡張でき、本研究でもこれを採用する。

Semantic matching の文脈でも、複数個 ($M (> 1)$) の attention を学習するモデルが提案されている [13]。[13] では、 M 個の attention $\alpha_{t,t'}^m$ ($m \in \{1, \dots, M\}$) から semantic matching において活用すべき k ($< M$) 個を選択して利用している。具体的には、 $\alpha_{t,t'}^m$ ($t \in \{1, \dots, T\}, t' \in \{1, \dots, T'\}, m \in \{1, \dots, M\}$) の値の大きい順に k 個利用する (k -max pooling) ことで、複数の attention 候補から実際に $(x_t, y_{t'})$ の組の関連度の高い k 個を選び出している。

2.3 Multiple Kernel Learning

本研究の提案手法は、multiple kernel learning に着想を得て、複数 attention を統合して利用する。Multiple kernel learning では、事前に用意した複数のカーネルから、どのカーネル (あるいは、どのカーネルの組み合わせ) を予測に用いるかを、データから学習する [3]。Attention のなす行列 ($\alpha_{t,t'}$) は必ずしも正定値ではない (正方行列であるとも限らない) のでカーネルが定めるグラム行列とみなすことはできず、直接 multiple kernel learning の問題になるわけではないが、同様の手法を複数の

attention を用いた予測にも応用可能である。

M 個のカーネル $\{K^1, \dots, K^M\}$ から、各サンプル $(x_t, x_{t'})$ に対して適切なカーネル $K_{t,t'} = K(x_t, x_{t'})$ を求める手法はデータ依存 (data-dependent) な手法として定義されている [3]。データ依存な multiple kernel learning のうち、もっともシンプルなもの、次のようにカーネルの重み付き線形和をその予測に用いるものである：

$$K_{t,t'} = \sum_{m=1}^M z_{t,t'}^m K(x_t, x_{t'})^m \quad (7)$$

ここで、 $z_{t,t'}^m$ は各 $(x_t, x_{t'})$ に対し、 m 番目のカーネルがどれだけ予測に寄与するかを定める値であり、各 t, t' に対し、 $z_{t,t'}^m \in [0, 1]$ で、 $\sum_{m=1}^M z_{t,t'}^m = 1$ である。

2.4 生成モデルにおける半教師有り学習

本研究では、誤り理由を隠れ変数とした生成モデルを提案する。しかし、誤り理由の教師データは相対的に取得コストが高いため、サンプルサイズが小さいのが一般的である。そこで、誤り理由の教師データのサンプルサイズが小さい場合にも適用できるように、半教師有り学習が行えることが重要である。

生成モデルにおける自然な半教師有り学習の提案と、そのテキストデータにおける応用は [7] で行われている。隠れ変数を z 、予測する変数を y 、学習するパラメータベクトルを θ とする。次の同時対数尤度を考える：

$$\log p(y, z|\theta) = \log \sum_z p(y|z, \theta)p(z|\theta) \quad (8)$$

z の教師データが利用できる場合には、同時対数尤度 $\log p(y, z|\theta)$ を直接計算し、パラメータを更新する。一方、 z の教師データが利用できない場合は、右辺の log-sum 形式の尤度を最大化するように、例えば EM アルゴリズムによって z の予測と θ の更新を交互に行う。

2.5 ニューラル文法誤り訂正

Attention メカニズムを実装した encoder-decoder モデルを用いた文法誤り訂正のモデルは既に [14] で提案されている。

[14] は、モデル内の変数を、よく用いられる単語単位ではなくキャラクタ単位とした。その理由として、次の 2 点が挙げられている [14]：

- (1) 入力系列データは必ずしもスペルチェックが行われていないため。
- (2) 顔文字や URL といった、vocabulary に含まれない out-of-vocabulary (OOV) なデータに対応するため。

この研究のモデルが、通常の encoder-decoder モデルと異なる点は、RNN として GRU を用いて、双方向かつピラミッド構造な encoder を実装している点である。Encoder の系列の長さを、階層が上の GRU ユニットになるにつれ短くしていくことで、長い系列のサンプルにも対応する狙いがある。

[14] は CoNLL2014 Shared Task のデータセットにおいて、M2 スコア ($F_{0.5}$ スコア) を用いて評価を行い、論文公開時点での state-of-the-art を達成した。

2.6 誤り発見と誤り訂正

我々の提案するモデルは、これまで個別に対応されることが多かった、誤り発見と、誤り訂正を同時に学習する機構になっ

ている。

例えば、[14] ではニューラル文法誤り訂正モデルとは別個に誤り発見識別器 (error classifier) を学習し、ニューラル文法誤り訂正モデルの修正を実際に適用するかどうかの判断に利用した。

3. 提案手法

文法誤りの理由を陽に推定することで、“ブラックボックス”でないニューラル文法誤り訂正モデルを提案する。提案する encoder-decoder モデルは、[14] と同じくキャラクタベースとし、attention メカニズムを実装している。提案手法の概略を図 1 に示す。

3.1 誤り訂正理由に応じた Attention

t' 番目の文字の誤り修正する際、修正前の文における何番目の文字に注目して直すべきかは、その誤りの理由に依存する。よって、 $(x_t, y_{t'})$ の関連度合いの指標には、ありうる誤り理由の個数 (M とする) に応じた M 個の attention ($\alpha_{t,t'}^1, \dots, \alpha_{t,t'}^M$) を考える。なお、ここでは便宜的に、 M 個の誤り理由の中には、“誤っていない” というクラスも含まれるものとする。これによって、先行研究 [14] ではニューラル文法誤り訂正モデルとは別に学習をしていたエラー識別も同時に学習が出来るようになる。

3.2 サンプルに応じた適切な誤り理由

$(x_t, y_{t'})$ に対する適切な attention を、先に述べた誤り理由に応じた attention の重み付き線形和によって求める：

$$\alpha_{t,t'} = \sum_{m=1}^M z_{t,t'}^m \alpha_{t,t'}^m \quad (9)$$

ここで、 $z_{t,t'}^m$ は文法誤りの理由を表すパラメータで、 $z_{t,t'}^m \in [0, 1]$ かつ $\sum_{m=1}^M z_{t,t'}^m = 1$ とする。これによって、 $\alpha_{t,t'}$ は、適切な誤り理由 $z_{t,t'}^m$ が与えられていれば、それに応じた attention $\alpha_{t,t'}^m$ を活用することができる。この形は 2.3 節で定義した multiple kernel learning のデータ依存なカーネルの定義の仕方に対応している。

なお、本研究では出力ベクトルの系列 $y = (y_1, \dots, y_{T'})$ の各 $y_{t'}$ に対して誤り理由を付与し、 t' だけに依存した $z_{t,t'}^m = z_{t'}^m$ を考えるものとする。

Attention に対する $z_{t'} = (z_{t'}^1, \dots, z_{t'}^M)$ も、他変数と同様に、入力系列と出力系列から推定される。これによって、

- (1) 推定時においても (推定された) 誤り理由に応じた適切な attention の活用が可能となり、
- (2) 文法誤り訂正時に、学習者に対して推定された誤り理由のフィードバックが可能となる。

$z_{t'}$ は、ニューラルネットワークモデル内の変数として追加するため、その計算や学習も他既存変数と同様に導出できる。まず、 h_t と $h_{t'}$ を結合した特徴量から、多層ニューラルネットワーク g' により、 $(h_t, h_{t'})$ を入力として、 t' 番目の文字に対してはどの誤り理由が適切かを予測する：

$$u'_{t,t'} = g'(h_t, h_{t'}) \quad (10)$$

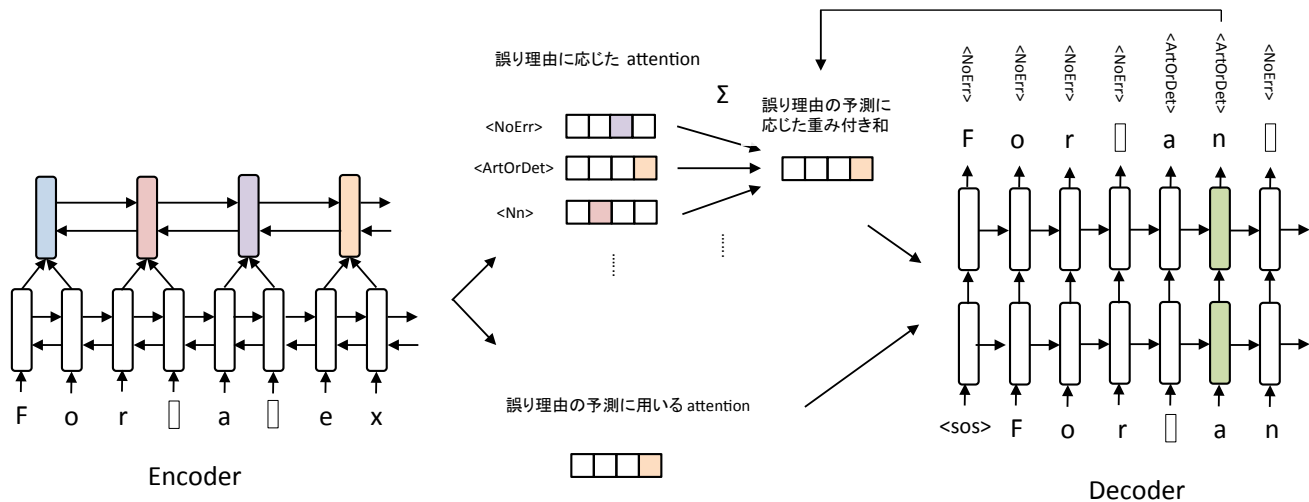


図 1 誤り理由に適応的なアテンション: 誤り理由に応じた複数のアテンションを学習し, 誤り理由に応じて適応的に選択することで適切なアテンションを生成する.

3.3 生成モデルとしての解釈

$z_{t,t'}^m \in [0, 1]$ かつ $\sum_{m=1}^M z_{t,t'}^m = 1$ という条件から, 式 (9) は, 誤り理由である隠れ変数 $z_{t,t'}$ を用いた生成モデルとして解釈し直すことが可能である. これにより, 後に説明するように [7] で提案された生成モデルにおける自然な半教師有り学習のアイデアを本手法にも適用することが可能となる. このモデルにおいて学習される全てのパラメータをまとめて θ とする. 式 (9) は次のように書き直せる:

$$p(\alpha_{t,t'}, z_{t,t'} | \theta) = \sum_z p(\alpha_{t,t'} | z_{t,t'}, \theta) p(z_{t,t'} | \theta) \quad (11)$$

(ここでは簡単のため, 入力ベクトル等の依存関係は省略してある.)

3.4 半教師有り学習に対応した学習則

学習は通常の encoder-decoder モデルと同じく誤差逆伝搬法によって行う. 提案モデルは誤り訂正理由を隠れ変数 $z_{t'}$ とした生成モデルであり, 2.4 節で述べた手法 [7] と同じアイデアを使って $z_{t'}$ について半教師有り学習をすることが出来る.

$z_{t'}$ の教師データが得られる場合, 目的関数は二つの誤差項の和からなる:

- (1) $z_{t'}$ を教師データで置き換えた encoder-decoder モデルから計算される誤差 (最終出力は g から)
 - (2) $z_{t'}$ を教師データとして, $z_{t'}$ を予測している多層ニューラルネットワーク g' から出力される値から計算される誤差
- これらについて誤差逆伝搬法によってパラメータを更新することにより, attention の選択と, それによる誤り訂正の両方が同時に学習される.

$z_{t'}$ の教師データが得られない場合, $z_{t'}$ をブートストラップして目的関数を計算することになる. まず, 固定された θ について $z_{t'}$ を予測し, その $z_{t'}$ に基づいて encoder-decoder モデルの出力を計算することで目的関数の値を計算する. その後, 誤差逆伝搬法に基づいて θ を更新する.

4. 数値実験の設定

4.1 データセット

モデルの学習と評価に用いたデータセットは CoNLL-2014 データセット [6], CoNLL-2013 データセット [5], Lang-8 データセット [10] の 3 つである.

先行研究のベンチマーク [4] との比較のため, CoNLL-2014 データセット [6] のテストデータにおいて誤り訂正と誤り理由推定の評価を行った. モデルの完全な教師有り学習には, 誤り文, 修正文のペア (x, y) 及び, 誤り理由を示す系列 z が必要であるが, 今回, それぞれ CoNLL-2014 データセットの誤り文, 修正文, 及び人手でラベリングされた誤りカテゴリを用いた. 今回, 誤りカテゴリのうち, トレーニングデータにおける誤りの数が 1500 以上ある 9 クラスと “誤りなし” を示す “NoErr” 及びその他の誤りを示す “OtherErr” の計 11 カテゴリを誤り推定の学習の対象とし学習を行った.

ニューラル文法誤り訂正のモデルの学習には, CoNLL-2014 データセットのトレーニングデータのサンプルサイズは十分であるとは言えないため, [14] と同じく, Lang-8 データセット [10] の誤り文, 修正文のペアもトレーニングデータとして利用した. 一方で, Lang-8 データセットには, CoNLL-2014 データセットと同じ誤りカテゴリのラベルは付けられていないため, これは 2.4 節で示した半教師有り学習の枠組みで学習を行った. 即ち, 現在のパラメータにおいて誤り理由の推定を行い, その推定結果に基づいて重み付き平均された attention を用いて予測を行い誤差を計算した.

CoNLL-2013 データセット [5] データセットは, 先行研究に倣い, テストデータをバリデーション用のデータセットとして利用した.

4.2 モデル及び学習の詳細

今回利用したモデルは, 3 層の GRU のユニットを用いて, そ

れぞれ隠れ層のノード数は 400 とした．本研究で用いたモデル及び最適化アルゴリズムは，提案部分である誤り理由に応じた attention の箇所以外 (Seq-to-Seq 部分) は [14] と同じ構造・アルゴリズム・パラメータを採用している．この部分についての詳細は [14] を参照されたい．

4.3 デコーディング

デコーディングにおいては，今回提案したモデル (NLC-ET モデル) 以外に，[14] で提案されたモデル (NLC モデル) 及び [14] と同じく言語モデル (LM モデル) によるスコアリングをデコーディングに活用を行った．スコアは LM モデルのスコアと割合 $\lambda \in [0, 1]$ で混合した．これらのハイパーパラメータ λ は共に検証用 (validation) データセットでの結果に基づいて決定をした (4.4 節)．

4.4 ハイパーパラメータの調整

Lang-8 と CoNLL2014 Shared Task のトレーニングデータで学習をした提案モデルに対して，precision 調整用のパラメータ η を $\eta \in \{0.0, 1.0, 3.0, 5.0, 10.0, 30.0, 100.0\}$ で検証用データセットにおいてグリッドサーチして M^2 スコアが最も高くなるものを求めた．この η は，高くなるほど“誤りなし”と誤り理由だと推定する確率が高くなるパラメータである．

また，言語モデルとのスコアの割合を決める λ も同様に $\lambda \in \{0.0, 0.1, \dots, 0.9, 1.0\}$ から，検証用データセットにおいてグリッドサーチして求めた．

5. 数値実験の結果

本節では，4. 節のデータセットとモデルに基づいて，本研究の貢献について検証した結果について述べる．5.1 節では先行研究の NLC モデル [14] と提案手法である NLC-ET モデルの精度比較を行い，NLC-ET モデルが先行研究のモデルと比べ遜色ない性能を持つことを示す．また， η を検証用データセットにおいて調整することで，誤り訂正においてより precision を重視したデコーディングが可能になることを示す．また，5.2 節では提案手法が誤り理由をどの程度正しく推定出来ているかを示す．

5.1 誤り理由の推定有りの文法誤り訂正の精度の評価

CoNLL2014 Shared Task のテストデータにおいて，提案手法の文法誤り訂正の精度を評価し，[14] のモデルと比較を行った．ここで，パラメータ η はで述べたように検証用データセットでのスコアから定めたもの ($\eta = 10.0$) と，調整をしていないもの ($\eta = 0.0$) を使っている．

結果は表 1 に示す通りである． M^2 スコアにおいて，先行研究のモデルと比べて遜色ないスコアが得られていることが確認できる．また，ハイパーパラメータ η を適切に設定することで誤った訂正の数を減らし，precision と M^2 スコアを改善できることがわかる．

なお，先行研究 [14] で示しているスコアはここで示しているスコアより高いものになっている．これは [14] では人工的な誤り生成をしたデータを学習に用いたり，誤り訂正を適用するかどうかを判定するモデルを別途学習し活用しているためだと考えられる．

表 1 Precision/Recall/ $F_{0.5}$ (M^2) スコアの比較

Model	Precision	Recall	M^2
NLC	42.23	15.36	31.29
NLC-ET ($\eta = 0.0$)	40.59	15.88	30.96
NLC-ET ($\eta = 10.0$)	42.91	15.80	31.95

5.2 誤り理由の推定精度の評価

提案手法において，Lang-8 と CoNLL2014 Shared Task のトレーニングデータを半教師有りで学習したモデルに対し，誤り理由の推定の精度の評価を行った．誤り理由は各クラス (理由) 毎にサンプルサイズが大きく異なるため，誤り理由の推定の精度は各クラス毎の AUC を計算し，それらのマクロ平均を取った．結果は表 2 に示す通りである．なお，ここでは誤り理由の正答ラベルを用意するため，誤り理由の推定において条件付ける文はテストデータセット中における真の訂正文を用いている．

6. おわりに

本研究では，ニューラル文法誤り訂正において，その訂正の理由 (根拠) が必ずしも自明でないという問題点に対し，その誤り理由も同時に隠れ変数として推定し，提示することが出来るモデルを提案した．また，我々の提案したモデルが既存のニューラル文法誤り訂正モデルに対して次の利点を持つことを示した:

- (1) モデルの提示する修正理由を推定することができる．
- (2) 誤り発見と誤り訂正を同時に一つのモデルで学習が可能で，precision/recall の調整に利用できる．
- (3) また，誤り理由の教師データが少ないデータセットに対しても，半教師有り学習が可能である．

今後の課題としては，文法誤り訂正の分野においては，そもそも誤り文と修正文のコーパスのサイズがそもそも十分なサイズと言えないため，これについても半教師有りに対応可能なニューラル文法誤り訂正モデルの開発や，現在考えている誤り理由は，人手でのラベル付けに依存するもののため，これをデータドリブンに教師なしで求める手法の開発などが挙げられる．

文 献

- [1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2015.
- [2] D. R. Ferris. Does error feedback help student writers? New evidence on the short-and long-term effects of written error correction. In K. Hyland and F. Hyland, editors, *Feedback in Second Language Writing: Contexts and Issues*, chapter 5, pages 81–104. Cambridge University Press, 2006.

表 2 正しいの訂正文で条件付けた場合の誤り理由推定の各クラス毎の AUC とそれらのマクロ平均

	NoErr	Vt	SVA	ArtOrDet	Nn	Prep	Wci	Wform	Mec	Rloc-	OtherErr	マクロ平均
AUC	0.74	0.81	0.77	0.85	0.74	0.85	0.92	0.74	0.68	0.84	0.84	0.79

- [3] M. Gönen and E. Alpaydm. Multiple kernel learning algorithms. *Journal of Machine Learning Research*, 12 (Jul):2211–2268, 2011.
- [4] M. Junczys-Dowmunt and R. Grundkiewicz. Phrase-based machine translation is state-of-the-art for automatic grammatical error correction. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.
- [5] H. T. Ng, S. M. Wu, Y. Wu, C. Hadiwinoto, and J. Tetreault. The CoNLL-2013 shared task on grammatical error correction. In *Proceedings of Conference on Computational Natural Language Learning: Shared Task (CoNLL)*, 2013.
- [6] H. T. Ng, S. M. Wu, T. Briscoe, C. Hadiwinoto, R. H. Susanto, and C. Bryant. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of Conference on Computational Natural Language Learning: Shared Task (CoNLL)*, 2014.
- [7] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2–3):103–134, 2000.
- [8] M. T. Ribeiro, S. Singh, and C. Guestrin. “Why Should I Trust You?”: Explaining the predictions of any classifier. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2016.
- [9] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.
- [10] T. Tajiri, M. Komachi, and Y. Matsumoto. Tense and aspect error correction for ESL learners using global context. In *Proceedings of Association for Computational Linguistics: Short Papers (ACL)*, 2012.
- [11] C. G. Van Beuningen, N. H. De Jong, and F. Kuiken. Evidence on the effectiveness of comprehensive error correction in second language writing. *Language Learning*, 62(1):1–41, 2012.
- [12] O. Vinyals and Q. Le. A neural conversational model. In *Proceedings of International Conference on Machine Learning (ICML)*, 2015.
- [13] S. Wan, Y. Lan, J. Guo, J. Xu, L. Pang, and X. Cheng. A deep architecture for semantic matching with multiple positional sentence representations. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2016.
- [14] Z. Xie, A. Avati, N. Arivazhagan, D. Jurafsky, and A. Y. Ng. Neural language correction with character-based attention. *arXiv:1603.09727 [cs.CL]*, 2016.