日本語文法誤り訂正における誤り傾向を考慮した擬似誤り生成

小川耀一朗　山本和英

長岡技術科学大学
{ogawa,yamamoto}@jnlp.org

1 はじめに

日本に在留する外国人は令和元年6月末時点で約282万人と過去最高となり、今後外国人労働者の受け入れが広がりさらに増加することが予想される。学習者作文の自動誤り訂正システムは語学教師の負担軽減や学習者の言語習得支援に活用することができるが、日本語での文法誤り訂正の研究は統計的機械翻訳 (SMT) を用いた手法が提案された 2013 年以来ほとんど行われていない。

本研究ではニューラル機械翻訳器にコピー機構を組み込んだモデルを用いて日本語学習者作文の訂正を行う。訓練データには学習者作文とその訂正文がペアになったコーパスを使用するが、その量は機械翻訳タスクに用いられるコーパスと比べて非常に少なく、十分な性能を引き出すことができない。そこで我々は大規模で容易に入手可能な日本語コーパスから擬似的に誤りを生成することで訓練データを拡張する。英語の文法誤り訂正で提案されている擬似誤り生成手法に加え、日本語特有の誤り傾向を考慮した擬似誤り生成手法を提案し比較を行った。NAIST 誤用コーパスでの性能評価において、提案手法は先行研究である SMT を用いた手法よりも M^2 スコアが 5.71 ポイント上回った。

2 関連研究

水本らは語学学習 SNS である Lang-8 から学習者作文とその添削文を収集して構築した大規模な学習者コーパスと、統計的機械翻訳 (SMT) を用いて日本語学習者作文の誤り訂正を行った [1]。機械翻訳モデルを用いた手法は大規模な学習者コーパスを必要とするため、Lang-8 コーパスは重要な役割を持つ。しかし、このコーパスを用いた日本語誤り訂正の研究はそれ以来行われていない。

英語作文の文法誤り訂正では機械翻訳手法、特にニューラル機械翻訳 (NMT) を用いた手法が主流となっている。Chollampatt らは多層畳み込みニューラルネットワークを用いた手法を提案し、従来の SMT を用いた手法を上回った [2]。Junczys-Dowmunt らは文法誤り訂正に資源の限られた機械翻訳タスクでの技術を適用し、また NMT の一つである Transformer[3] を用いた手法を提案した [4]。これ以降多くの研究で Transformer が用いられるようになった。

NMT は大規模な訓練データを必要とするが、機械翻訳タスクで使用されている対訳コーパスの量に比べ学習者コーパスの量は少ない。学習者コーパス不足を補うようにして、正しい文から擬似的に誤り文を生成して訓練データを拡張する手法が提案されるようになった。Xie らは学習者コーパスの訂正文から誤り文を生成するような逆翻訳モデルを訓練させ、そのモデルに正しい文を入力することで誤りを含む文を生成した [5]。Zhao らは正しい文に対して置換・削除・挿入・入れ替えの操作をランダムに行うといった、シンプルな擬似誤り生成手法を提案し、Transformer にコピー機構 [6] を組み込んだモデルを事前学習させることで高い性能を示した [7]。Grundkiewicz らはより現実的な擬似誤りを生成するために、confusion set と呼ばれる品詞が同じであったりスペルが似ていたりして混同されやすい単語の集合を定義し、置換する単語を confusion set から選択するようにした [8]。我々はこれらの手法に加え、日本語特有の誤り傾向を考慮した擬似誤り生成手法を提案し、NMT の事前学習に用いる。

3 ニューラル機械翻訳モデル

本研究では、Zhao らが提案したコピー機構を組み込んだ Transformer(TransformerCopy) を用いて誤り訂正を行う。Transformer はエンコーダとデコーダで構成されている。エンコーダは複数ヘッドの自己注意機構と位置毎の全結合層を含むブロックを L 個スタックした構造となっており、入力文から文脈を考慮した潜在表現を生成する。デコーダはエンコーダのブロックである自己注意機構と位置毎の全結合層に加え、エンコーダの潜在表現に向けた注意機構を持つ。

コピー機構は入力系列からトークンをコピーする機能を持つ。最終的な確率分布 P_t は式 (1) のように、生成確率分布 P_t^{gen} とコピー確率分布 P_t^{copy} の組み合わせで表される。$\alpha_t^{copy} \in [0,1]$ は生成とコピーのバランスを制御する役割を持つ。

$$p_t(w) = (1 - \alpha_t^{copy}) * p_t^{gen}(w) + \alpha_t^{copy} * p_t^{copy}(w) \quad (1)$$

コピー確率分布はデコーダの潜在表現 h^{trg} とエンコーダの潜在表現 H^{src} の間の注意の計算と同様に計

Copyright(C) 2020 The Association for Natural Language Processing.
All Rights Reserved.

表 1: 擬似誤り文

Original	少年スコットの夢は、イギリス海軍の提督司令官だった。
BackTrans	ほとんど、少年スコットの夢はイギリス海軍の提督司葬官だった。
DirectNoise	少スコのは、のギリ海の軍督提司令官だっ。
DirectNoise(ja)	少年スコットで夢なんて、イギリス海軍の提督司令官だった。
Original	その犠牲は余りにも大きい。
BackTrans	じゃあ、その遂籠は余りにも大きい。
DirectNoise	、の牲はに要も大。い
DirectNoise(ja)	その犠牲は余にも大きい

算される。

$$q_t, K, V = h^{trg}W_q^T, H^{src}W_k^T, H^{src}W_v^T \quad (2)$$

$$A_t = q_t^T K \quad (3)$$

$$P_t^{copy}(w) = softmax(A_t) \quad (4)$$

q_t, K, V は注意の計算に必要な query, key, value のことである。コピー機構の潜在表現をバランス変数 α_t^{copy} として用いる。

$$\alpha_t^{copy} = sigmoid(W^T \sum(A_t^T \cdot V)) \quad (5)$$

4 擬似誤り生成手法

本研究では BackTrans、DirectNoise、DirectNoise(ja) の 3 つの手法で擬似誤り文を生成し、モデルの事前学習に用いた。表 1 は生成された擬似誤り文の例である。

4.1 BackTrans

BackTrans は Xie らの提案手法で、学習者コーパスの添削文から誤り文を生成するように学習された逆翻訳モデルに正しい文を入力し、擬似誤り文を獲得する方法である。多様な出力を得るために、ビームサーチ時の各候補のスコアに $r\beta_{random}$ のノイズを加える。r は [0, 1] のランダム値で、β_{random} は 6 に設定した。

4.2 DirectNoise

DirectNoise は正しい文に対して直接ノイズを加える手法であり、本研究では Zhao らの手法を用いる。入力文の各トークンに対して以下の 4 つの操作を行う。

置換 10%の確率でコーパスからランダムに選択した単語で置換する。

削除 10%の確率で削除する。

挿入 10%の確率で後ろにコーパスからランダムに選択した単語を挿入する。

入れ替え そのトークンの位置に標準偏差 0.5 の正規分布の確率値を加えたスコアを割り振る。最終的にトークンのスコアで語順をソートすることで語順入れ替えを行う。

4.3 DirectNoise(ja)

DirectNoise の操作は現実的ではないノイズを発生させてしまう。そこで以下の 3 つの日本語特有の誤り傾向を DirectNoise の操作に取り入れる。

- 助詞の誤りが頻出する。
- 送り仮名の不足が生じる。
- 文節内の語順誤りはあるが、文節の順番の違いは多くの場合に文法誤りにならない。

置換 助詞は 10%、助詞以外は 5%の確率で置換する。置換する単語は 70%の確率で助詞セットから、30%の確率でコーパスからランダムに選択する。

削除 助詞は 10%、助詞以外は 5%の確率で削除する。また、送り仮名がある単語は 50%の確率で送り仮名の 1 文字目を削除する。

挿入 5%の確率で後ろに単語を挿入する。挿入する単語は 70%の確率で助詞セットから、30%の確率でコーパスからランダムに選択する。

入れ替え DirectNoise の入れ替え操作を文節ごとに行う。文節の入れ替えは行わない。

5 日本語学習者作文の誤り訂正実験

5.1 実験データ

訓練データには学習者コーパス Lang-8 から日本語の作文を抽出し、さらに日本語と記号以外の文字 (アルファベットなど) を含む文を除外した 1,627,963 文対を使用し、うち 5,000 文を開発データに使用した。擬似誤り生成の元となる日本語コーパスとして現代日本語書き言葉均衡コーパス (BCCWJ) の 5,883,005 文を使用した。評価データには NAIST 誤用コーパス [9] を使用した。これは国立国語研究所による収集された「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 (作文対訳 DB)」に誤用タグを付与した

Copyright(C) 2020 The Association for Natural Language Processing.
All Rights Reserved.

表 2: NAIST 誤用コーパスでの誤り訂正結果

モデル	Precision(%)	Recall(%)	$F_{0.5}$(%)	GLEU(%)
Transformer	38.7	5.97	18.5	48.1
TransformerCopy	**40.3**	6.80	20.3	50.0
TransformerCopy+BackTrans	35.4	11.0	24.5	51.6
TransformerCopy+DirectNoise	34.2	12.2	25.1	51.5
TransformerCopy+DirectNoise(ja)	36.2	12.2	**26.0**	**51.9**
水本ら (SMT)[1]	23.6	**13.0**	20.3	-

ものである。このコーパスから 6,672 文を取り出し、誤用文と添削文に分けて使用した。

5.2 実験設定

日本語は英語とは異なり単語で区切られておらず、何らかの文を単位で分割する必要がある。日本語学習者の作文では漢字を避けて平仮名を多用する傾向があるため、従来の形態素解析器で単語分割を行っても正しく分割されない。そこで入力文を文字単位に分割することでこの問題に対応する。Lang-8 の文字種数である 6,288 をモデルの語彙サイズとした。

本研究で用いる TransformerCopy のパラメータは、エンコーダ及びデコーダは 6 層、ヘッド数は 8、潜在表現は 512 次元、フィルタサイズは 4096、ドロップアウトは 0.2、推論時のビームサイズは 12 に設定した。最適化手法には Nesterovs Accelerated Gradient を用い、学習率は最大値が 0.004 から 0.001 に減衰する Cyclical Learning Rate[10] に設定した。また、変更されるべき単語の損失を加重する edit-weighted MLE objective[4] を用いた。事前学習時は $\Lambda = 3$ に、学習者コーパス学習時は $\Lambda = 1.2$ に設定した。BackTrans のモデルには TransformerCopy と同じパラメータに設定した Transformer を使用した。

5.3 評価尺度

評価尺度には MaxMatch(M^2)[11] 及び GLEU[12] を用いた。M^2 スコアは文字単位で適合率 (Precision)、再現率 (Recall) を計算し、Precision を重視した $F_{0.5}$ で評価する。GLEU スコアは出力文を単語単位に分割し直し、出力文と正解文の単語 4-gram 一致率から計算する。単語分割には MeCab と UniDic 辞書を用いた。

5.4 実験結果

表 2 に誤り訂正実験の結果を示す。Transformer にコピー機構を組み込むことで $F_{0.5}$ が 1.8 ポイント向上した。さらに擬似誤り生成により得たデータでモデルを事前学習し、そのパラメータを初期値として学習者コーパスを訓練することで、BackTrans 手法では $F_{0.5}$ が 4.2 ポイント、DirectNoise 手法では 4.8 ポイント向上した。提案手法である DirectNoise(ja) 手法では元の手法である DirectNoise より Precision が 2 ポイント向上し、事前学習なしより $F_{0.5}$ が 5.7 ポイント向上した。GLEU スコアも最も高くなっている。また提案手法は先行研究と比べ、Recall が 0.8 ポイント低いが Precision は 12.6 ポイント高く、$F_{0.5}$ は 5.7 ポイント上回った。

6 考察

6.1 擬似誤りコーパスによる事前学習

事前学習しないモデルは Precision が高い一方で Recall が低い。M^2 スコアの計算ログからモデルの編集回数を集計すると、Transformer は 2,919 回だったのに対し TransformerCopy+DirectNoise(ja) は 5,813 回と、約 2 倍も編集を行っていた。つまり事前学習なしのモデルは訂正に消極的であり、自信のある訂正のみ行っている。それに対し擬似誤りコーパスによる事前学習を行うことで Recall が高まる。NAIST 誤用コーパスの 1 文当たりの誤りタグ数の平均は 2.75 であり、学習者コーパスではほとんどがコピーである一方、擬似誤り生成ではそれより多くの操作が行われるため、訂正することをより学習しているモデルであると考えられる。

6.2 擬似誤り文の比較

表 1 に生成した擬似誤り文の例を挙げた。BackTrans の擬似誤り文は原文から言い換えや付け足しが行われた流暢な文になっており、文法誤りの割合は少ない。一方 DirectNoise は文字単位でランダムに操作が行われるため非常に非文法的な文が生成される。提案手法である DirectNoise(ja) は、原文の文法を保持しつつも部分的な誤りが含まれており、BackTrans と DirectNoise の間に位置すると言える。

6.3 出力例

表 3 に提案手法での出力例を示す。赤文字が誤り箇所、青文字が訂正箇所を表している。提案手法は日本語学習者作文に頻出する助詞誤りに対して正しく訂正

Copyright(C) 2020 The Association for Natural Language Processing.
All Rights Reserved.

表 3: 出力例

Input	だから、たばこの吸う人がたくさんいる。
Output	だから、たばこを吸う人がたくさんいる。
Reference	だから、たばこを吸う人がたくさんいる。
Input	人間の健康ようにたばこを吸わなくてください。
Output	人間の健康のためにたばこを吸わないでください。
Reference	人間の健康のためにたばこを吸わないでください。
Input	個人の権利はもし他人の権利を悪く影響したら、禁止られるべきです。
Output	個人の権利はもし他人の権利に悪く影響したら、禁止されるべきです。
Reference	個人の権利はもし他人の権利に悪く影響したら、禁止されるべきです。
Input	でも、人はその一人ばかりこの地球ですむのはない、
Output	でも、人はその一人だけこの地球に住むことはない、
Reference	でも、人は一人だけでこの地球にすんでいるのではありません。

することができている。4 つ目の例は正しい出力ができなかった例である。入力文と比べたら文法的な文に訂正されているものの、添削文のように多くの編集を行うことまではできていない。一部の誤りを訂正する場合と文の大部分を訂正する場合の識別が必要であると考える。

7 まとめ

本研究ではニューラル機械翻訳器にコピー機構を組み込んだモデルを用いて日本語学習者作文の文法誤り訂正を行い、また擬似誤り生成による訓練データ拡張により性能が向上することを確認した。日本語特有の誤り傾向を考慮した擬似誤り生成手法での性能が最も高く、先行研究よりも高い性能を示した。どのような擬似誤りが性能に寄与するのか、更に詳しい分析・検証が必要である。

NAIST 誤用コーパスでの評価においてまだまだ性能が低い。誤り訂正において誤り箇所以外は変更を加える必要はないが、生成モデルの性質上、誤り箇所以外で原文とは異なるフレーズを出力してしまう。日本語は活用の多い動詞が文末に現れる特徴があるが、特に文末の余計な言い換えによって不正解となるケースが多かった。本研究ではコピー機構を用いて不要な言い換えを抑制しようと試みたが、さらに強い制約が必要であると考える。

最後に、本研究で用いた誤り訂正モデルをウェブで試すことができるように、日本語文法誤りチェッカーを公開した。`http://www.jnlp.org/SNOW/S24`

謝辞

本研究は、平成 27-31 年科学研究費補助金 基盤 (B) 課題番号 15H03216 の助成を受けています。

参考文献

[1] 水本智也, 小町守, 永田昌明, 松本裕治. 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. 人工知能学会論文誌, Vol. 28, No. 5, pp. 420–432, 2013.

[2] Shamil Chollampatt and Hwee Tou Ng. A Multilayer Convolutional Encoder-Decoder Neural Network for Grammatical Error Correction. In *AAAI*, 2018.

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pp. 5998–6008. Curran Associates, Inc., 2017.

[4] Marcin Junczys-Dowmunt, Roman Grundkiewicz, Shubha Guha, and Kenneth Heafield. Approaching Neural Grammatical Error Correction as a Low-Resource Machine Translation Task. In *NACCL-HLT*, pp. 595–606, 2018.

[5] Ziang Xie, Guillaume Genthial, Stanley Xie, Andrew Ng, and Dan Jurafsky. Noising and Denoising Natural Language: Diverse Backtranslation for Grammar Correction. In *NACCL-HLT*, pp. 619–628, 2018.

[6] Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. Incorporating Copying Mechanism in Sequence-to-Sequence Learning. In *ACL*, pp. 1631–1640, 2016.

[7] Wei Zhao, Liang Wang, Kewei Shen, Ruoyu Jia, and Jingming Liu. Improving Grammatical Error Correction via Pre-Training a Copy-Augmented Architecture with Unlabeled Data. In *NACCL-HLT*, pp. 156–165, 2019.

[8] Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. Neural Grammatical Error Correction Systems with Unsupervised Pre-training on Synthetic Data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 252–263, 2019.

[9] 大山浩美, 小町守, 松本裕治. 日本語学習者の作文における誤用タイプの階層的アノテーションに基づく機械学習による自動分類. 自然言語処理, Vol. 23, No. 2, pp. 195–225, 2016.

[10] Leslie N. Smith. Cyclical Learning Rates for Training Neural Networks. In *Applications of Computer Vision (WACV)*, pp. 464–472, 2017.

[11] Daniel Dahlmeier and Hwee Tou Ng. Better Evaluation for Grammatical Error Correction. In *NAACL*, pp. 568–572, 2012.

[12] Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. Ground Truth for Grammatical Error Correction Metrics. In *ACL-IJCNLP*, pp. 588–593, 2015.

Copyright(C) 2020 The Association for Natural Language Processing.
All Rights Reserved.