教師なし文法誤り訂正

勝又 智　　小町 守

首都大学東京
katsumata-satoru@ed.tmu.ac.jp, komachi@tmu.ac.jp

1 はじめに

近年，文法誤り訂正（GEC）の研究が盛んである．GEC は，入力が文法的に誤っている文（原言語側），出力がこの誤りを訂正した文（目的言語側）であり，この入出力の系列変換を行うタスクである．そのため，状況設定の類似性から，機械翻訳（MT）で用いられる手法を GEC に適用した研究が数多く存在している．最近では，大量の整形されたコーパスと擬似的に作成したコーパスを用いることで，高い精度が得られることが知られている．

GEC において一般に使用することが可能なデータとして，言語学習者の相互添削サイトである Lang-8 から抽出したものが知られている．学習者の文とそれを訂正した文として対訳関係になっており，対訳コーパスとして使用可能である．しかし，この学習者の文に対する訂正文が必ずしも完璧な訂正文かどうかは保証されていない．きちんとした母語話者に学習者の文を訂正してもらう，というのがコーパスの整形方法として一番良いと考えられるが，その場合はコーパス作成のコストが莫大にかかってしまう．

本研究では，主に MT で研究されている教師なし手法を GEC に取り入れて，学習者の文と訂正文の対訳関係を崩した状態での性能を調査する．つまり，訂正側の文が学習者の誤り文と対応している必要があるのかどうかを調べた．GEC では学習者の文は Lang-8 などの SNS に投稿されていくため，時間とともに増えていき，対応する訂正文を作成するのは多大なコストを費やす．この手法を発展させていくと，将来的には多くの母語話者の単言語データを目的言語側として使用できるので，このようなコストをかけずに済む．

Lang-8 から抽出したデータの対訳関係を崩した状況下での誤り訂正の実験の結果，$F_{0.5}$ で 22.40 ポイント，GLEU スコアで 43.14 ポイントを達成した．

2 関連研究

2.1 教師なし機械翻訳

近年，ニューラル機械翻訳（NMT），統計的機械翻訳（SMT）それぞれにおいて，教師なし手法の研究が盛んである．特に，Artetxe ら [2] による SMT を用いた教師なし手法（USMT）は，まず言語横断的な embedding を獲得，この embedding を元にフレーズテーブルを作成する．次にこのフレーズテーブルと，単言語データから学習した言語モデルを既存の SMT の枠組みに導入したものを初期モデルとする．そしてこの初期モデルを元に，単言語データから擬似データを作成して SMT モデルを洗練していく．また Lample ら [8] や Marie and Fujita [9] は，USMT の最終的なモデルから作成した擬似データを，UNMT の最終モデルに追加する，または NMT モデルの初期化に使用する有効性についても報告している．

本研究では，この教師なし翻訳手法を GEC に適用した．具体的には Artetxe らの USMT 手法を用いた[1]．

2.2 教師あり文法誤り訂正

GEC の分野では大規模コーパスを用いた教師あり NMT 手法の研究が盛んである．現在の最高精度を報告している Ge ら [5] は，純粋な対訳データだけでも合計 5.4M 文対のコーパスを使用し，教師あり NMT の手法を用いている．彼らが使用しているデータには，英語教師による訂正や Lang-8 利用者による訂正といった様々な粒度の訂正が含まれている．

しかしながら，Lang-8 の訂正の不十分さや，他の言語の GEC について考えた際に，質の高い訂正が行われたデータを用意することは本来難しい．本研究では Lang-8 から抽出した対訳データ [11] の対訳関係を崩したものを学習データとすることで，対訳関係の必要性を調査し，訂正が不十分であるデータから GEC が可能なのかを調べる．

教師あり NMT だけでなく，教師あり SMT を GEC に用いた研究も多く報告されている．これらの研究は SMT の代表的ツールである Moses [7] を，翻訳で使用される枠組みをほぼそのまま GEC に適用している．

SMT を用いた MT と GEC の違いとしてチューニングに関する点が挙げられる．Susanto ら [15] は GEC のチューニングに MERT [14] を使用して，MT の評

[1] 事前実験で UNMT を用いた手法を用いたが有効な結果は得られなかった．

アルゴリズム 1 教師なし文法誤り訂正

Require: 原言語側言語モデル LM_s
Require: 目的言語側言語モデル LM_t
Require: 原言語側コーパス C_s
Require: 目的言語側コーパス C_t
Require: 繰り返し数 N
Ensure: 原言語→目的言語フレーズテーブル $P^{(N)}_{s \to t}$

1: $W^{emb}_s \leftarrow \text{TRAIN}(C_s)$
2: $W^{emb}_t \leftarrow \text{TRAIN}(C_t)$
3: $W^{cross\_emb}_s, W^{cross\_emb}_t \leftarrow \text{MAPPING}(W^{emb}_s, W^{emb}_t)$
4: $P^{(0)}_{t \to s} \leftarrow \text{INITIALIZE}(W^{cross\_emb}_s, W^{cross\_emb}_t)$
5: **for** $iter = 1, \ldots, N$ **do**
6: $\quad \text{pseudo\_data}_s \leftarrow \text{DECODE}(P^{(iter-1)}_{t \to s}, LM_s, C_t)$
7: $\quad P^{(iter)}_{s \to t} \leftarrow \text{TRAIN}(\text{pseudo\_data}_s, C_t)$
8: $\quad \text{pseudo\_data}_t \leftarrow \text{DECODE}(P^{(iter)}_{s \to t}, LM_t, C_s)$
9: $\quad P^{(iter)}_{t \to s} \leftarrow \text{TRAIN}(\text{pseudo\_data}_t, C_s)$
10: **end for**

価尺度である BLEU スコアに対して最適化している．一方で，Junczys-Dowmunt and Grundkiewicz [6] は，BLEU スコアに対して最適化した場合，原言語側と目的言語側の表層が一致するように出力するようになり，消極的な訂正になると主張し，$F_{0.5}$ スコアに対して最適化した．

本研究では後述する教師なし設定におけるチューニングの問題により，Moses のデフォルト設定を使用している．つまり，語順に関する素性と GEC 特有の素性は使用せず，MT のデフォルトの重みを用い，再チューニングは行っていない．

3 教師なし文法誤り訂正

アルゴリズム 1 に本研究における教師なし GEC 手法の疑似コードを記述する．大部分は Artetxe らの USMT 手法を元にしている．

言語横断的な n-gram embedding の作成 原言語側，目的言語側のそれぞれから n-gram embedding を作成する．具体的にはそれぞれの単言語データにおいて頻度の高い unigram，bigram，trigram[2]に対して skip-gram [10] の枠組みを用いて単言語 embedding を作成する．その後，作成した個々の単言語 embedding を共有言語横断空間へマッピングする．このマッピングには Artetxe ら [1] の手法を使用し，教師なしでマッピングを行っている．この embedding の次元数は 300 とした．

フレーズテーブルの作成 作成した言語横断的な n-gram embedding からフレーズテーブルを作成する．具体的には句翻訳モデルと語彙翻訳モデルを作成する．

[2]それぞれ学習データ内の頻度順に 200K，400K，400K を使用．

原言語側の句 $\bar{e}$ に対する目的言語側の句 $\bar{f}$ の句翻訳モデル $\phi(\bar{f}|\bar{e})$ は，ある原言語側の句に対して，共有言語横断空間内の 100 近傍の目的言語の句を候補とした．句翻訳モデルのスコアはある原言語側の句と目的言語側の句のコサイン類似度を正規化したものを使用している．具体的には次式の通りである．

$$\phi(\bar{f}|\bar{e}) = \frac{\cos(\bar{e}, \bar{f})/\tau}{\sum_{\bar{f}'} \cos(\bar{e}, \bar{f}')/\tau}$$

$\bar{f}'$ は目的言語側の句集合の各要素を表しており，τ は予測の信頼度を制御する温度パラメーターである．[3]目的言語側の句に対する原言語側の句の句翻訳モデル $\phi(\bar{e}|\bar{f})$ も同様である．

原言語側の句 $\bar{e}$ に対する目的言語側の句 $\bar{f}$ の語彙翻訳モデル $\text{lex}(\bar{f}|\bar{e})$ は，原言語側の句内の各単語に対して，目的言語側の句内で翻訳確率が最も高い単語を対応した単語とする．つまり語彙翻訳モデルのスコアは原言語側の句内の各単語に対応する翻訳確率の積を用いる．具体的には次式の通り．

$$\text{lex}(\bar{f}|\bar{e}) = \prod_i \max\left(\epsilon, \max_j \phi\left(\bar{f}_i|\bar{e}_j\right)\right)$$

ϵ は対応する単語が存在しない場合のための定数項である．本研究では Artetxe らと同様に 0.001 とした．目的言語側の句に対する原言語側の句の語彙翻訳モデル $\text{lex}(\bar{e}|\bar{f})$ も同様である．

逆翻訳による SMT モデルの学習 上記で作成したフレーズテーブルは trigram までしか考慮されておらず，句翻訳モデル，語彙翻訳モデルそれぞれのスコアも言語横断的な embedding を元に推定したものである．

そのため，逆翻訳の機構を利用することでフレーズテーブルの更新を行う．具体的な流れはアルゴリズム 1 の 5-10 行目に対応している．初期フレーズテーブル $P^{(0)}_{t \to s}$ と言語モデル LM_s を用いて，目的言語側の単言語コーパスに対する原言語側の擬似データを作成する．この原言語側が擬似データとなっている擬似対訳データを用いて SMT を学習，$P^{(1)}_{s \to t}$ を作成する．この $P^{(1)}_{s \to t}$ を用いて，原言語側の単言語データを翻訳，目的言語側が擬似的になっている擬似対訳データを作成する．この擬似データを用いて $P^{(1)}_{t \to s}$ を学習する．この操作を予め決めた繰り返し数 N だけ実行する．

教師なしチューニングに関する手法 我々の手法と Artetxe らの手法の大きな違いは SMT のチューニングの有無である．彼らは逆翻訳の枠組みを利用して，開発データとして擬似対訳データを作成し，この開発

[3]τ は Artetxe らと同様に，ある embedding に対して逆側の最近傍の句の embedding と，元の embedding の句翻訳確率が最大になるように推定する．

データに対して MERT を用いて BLEU に最適化している．Marie and Fujita は，Artetxe らのチューニング手法は最初に Moses のデフォルトの重みを使用しており，この重みは欧米言語の翻訳タスクに対して調整されているため，翻訳タスクでの教師なしチューニングとして良いのか疑問を投げかけている．

本研究が取り組む問題は MT ではなく GEC であるため，デフォルトの重みを使用しても教師なし設定のままである．また，GEC において，教師あり設定ならば $F_{0.5}$ スコアに対して最適化することが有効であることは知られているが，教師なし設定の場合，$F_{0.5}$ を測定するための開発データ作成が困難となる．BLEU スコアに対するチューニングを考えると，2.2 節で述べた消極的な訂正になりやすいという悪影響が存在し，この特性が教師なし設定で与える影響を考えると分析が困難となる．本研究はできる限り教師なしの設定で実験を行い，簡単のためチューニングの影響をできるだけ考えないよう，重みをデフォルトの値に固定して実験を行う．

4 教師なし文法誤り訂正実験

4.1 実験設定

本研究では使用したデータとして Mizumoto ら [11] の Lang-8 から抽出したデータを用いた．前処理として，Chollampatt and Ng [3] と同様の処理[4]を行った．最終的に 1,282,789 文対の学習データをそれぞれシャッフルし，対訳コーパスではない設定で実験を行った．評価には CoNLL-14 の評価データ（1,312 文）と JFLEG の評価データ（747 文）をそれぞれ用いた．

逆翻訳時の SMT の学習には Moses を用いた．単語アラインメントには FastAlign[5]を，言語モデルの学習には KenLM[6]を使用し，5-gram 言語モデルを推定した．文長制限として，擬似データの文対は [3, 80] 単語の文長を満たすものを使用した．繰り返し数 N は 3 とした．

本研究では比較として，対訳状況を崩す前の Lang-8 のデータを用いて，NMT，SMT による教師あり GEC もそれぞれ行った．教師あり NMT は Ge らと同様に畳み込みネットワークを用いたモデルを使用しており，パラメータ設定は彼らと同一である．NMT のモデル選択に CoNLL-13 [13] のデータを開発データとして使用した．教師あり SMT に関しては，チューニングはしておらず，SMT の設定は USMT の逆翻訳のものと同一である．

[4]tokenizer には NLTK（https://www.nltk.org/）を用いた．
[5]https://github.com/clab/fast_align
[6]https://kheafield.com/code/kenlm

表 1: 文法誤り訂正実験の $F_{0.5}$ および GLEU スコア

		CoNLL-14			JFLEG
	iter	P	R	$F_{0.5}$	GLEU
訂正なし	-	-	-	-	40.54
教師あり NMT	-	**55.72**	25.89	**45.29**	**51.62**
教師あり SMT	-	23.15	13.27	20.15	44.57
USMT	0	17.04	**29.38**	18.60	7.75
	1	14.73	20.38	15.60	36.12
	2	37.00	8.69	**22.40**	**43.14**
	3	**39.19**	7.81	21.72	42.89

評価尺度として CoNLL-14 のデータには $F_{0.5}$[4] を，JFLEG のデータには GLEU スコア [12] を使用した．

4.2 実験結果

誤り訂正実験の結果を表 1 に示す．CoNLL-14 では iter = 0 で $F_{0.5}$ が 18.60 ポイントを達成しており，教師あり SMT と比べて 1.55 ポイント低いことがわかる．また，iter = 2 のとき，$F_{0.5}$ は教師あり SMT と比べて 2.25 ポイント向上していることがわかる．一方で教師あり NMT と比較すると 22.89 ポイント低いことがわかる．

JFLEG については，USMT の中で最も高いのは iter = 2 のときで，教師あり SMT と比べて 1.43 ポイント低いことがわかる．教師あり NMT と比較すると 8.48 ポイント低いことがわかる．

5 考察

表 1 より，USMT では iter が大きくなると Precision が大きくなり，Recall が減少していることがわかる．つまり，逆翻訳によってモデルを再学習するにつれて，訂正が消極的になっていると考えられる．iter が大きくなるにつれて消極的な訂正を行っている例を表 2 に示す．誤り文に対して iter = 0, 1 は多くの訂正を提案しているが，文意を変えてしまったり，文法的に誤ったものを出力するものが多いことがわかる．一方で iter = 2, 3 では 1ヶ所のみ文法的に正しい訂正を行っている．また，実際に訂正回数を数えたものを表 3 に示す．iter = 1 から iter = 2 でモデルの訂正数が 2,914 件も減少していることがわかる．Artetxe らや Lample らは iter を増加させることで MT の精度が向上することを報告している．GEC でも iter を増加させることで精度は向上するが，代わりに消極的な訂正を行うようになると考えられる．

表 3 から教師あり SMT では削除操作の方が挿入操作より多いが，USMT では iter = 1, 2, 3 のときに挿入操作の方が削除操作より多いことがわかる．つまり，逆翻訳を行いフレーズテーブルを更新することで，削

表 2: CoNLL-14 における様々な iter での USMT の出力例

誤り文	It is better to let the other party know the fact than after the baby is born and certain type of genetic disease is found.
iter 0	When you want to kill people know now than before the stroller and certain types of societal disease corrected.
iter 1	It is better to let the other party. The fact than after the baby is born, and **a** certain type of genetic cancer was found.
iter 2, 3	It is better to let the other party know the fact than after the baby is born and **a** certain type of genetic disease is found.
正解	It is better to let the other party know the **facts rather than wait until** after the baby is born and **a** certain type of genetic disease is found.

表 3: CoNLL-14 について，実際の訂正数の分析

	iter	置換	削除	挿入	合計
教師あり SMT	-	588	520	188	1,296
	0	3,819	695	64	4,578
	1	2,304	336	774	3,414
	2	245	59	196	500
USMT	3	200	51	170	421

除操作を行わず挿入操作を行うようになっている．これは原言語から目的言語へのフレーズテーブルを更新する際に使用する擬似データの影響だと考えられる．このときのフレーズテーブルの更新には原言語側，つまり誤り側が擬似データとなっている．この擬似誤りを作成する際に USMT は単語を挿入することで誤りを生成するのではなく，単語を削除することにより誤りを生成していると考えられる．実際に $P^{(0)}_{t \to s}$ と原言語側の言語モデル LM_s で生成した擬似データの文長は，平均 5.33 単語だけ擬似誤り側が短くなっていることを確認した．生成された擬似誤りデータ内には単語を挿入することで文法的に正しい文になるものが多く存在し，作成したフレーズテーブルは挿入操作を行うようになっていると考えられる．

6 おわりに

本研究では，文法誤り訂正の分野において，学習データが対訳関係にある必要があるのか，崩した場合にはどの程度の精度になるのかを調査した．手法として，統計的機械翻訳を用いた教師なし手法を用いた．

小規模な訂正に関しては $F_{0.5}$ が 22.40 ポイント，大規模な訂正に関しては GLEU スコアが 43.14 ポイントという結果となり，統計的機械翻訳では対訳関係を残した場合と比較して $F_{0.5}$ が 2.25 ポイント向上した．また，逆翻訳を用いて SMT を更新していくことで，より消極的で，置換操作と挿入操作を中心とした訂正を行うようになっていくことを示した．

現在，文法誤り訂正は**英語**学習者支援の目的で盛んである．しかし，データとして Lang-8 には部分的に訂正されたデータが**他の言語**でも存在している．本研究の実験設定は英語以外の言語に対しても適用可能であると考えられるので，英語以外の語学学習者支援として発展させたい．

参考文献

[1] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proc. of ACL*, pp. 789–798, 2018.

[2] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Unsupervised statistical machine translation. In *Proc. of EMNLP*, pp. 3632–3642, 2018.

[3] Shamil Chollampatt and Hwee Tou Ng. A multilayer convolutional encoder-decoder neural network for grammatical error correction. In *Proc. of AAAI*, pp. 5755–5762, 2018.

[4] Daniel Dahlmeier and Hwee Tou Ng. Better evaluation for grammatical error correction. In *Proc. of NAACL-HLT*, pp. 568–572, 2012.

[5] Tao Ge, Furu Wei, and Ming Zhou. Reaching human-level performance in automatic grammatical error correction: An empirical study. *arXiv preprint arXiv:1807.01270*, 2018.

[6] Marcin Junczys-Dowmunt and Roman Grundkiewicz. The AMU system in the CoNLL-2014 shared task: Grammatical error correction by data-intensive and feature-rich statistical machine translation. In *Proc. of CoNLL*, pp. 25–33, 2014.

[7] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proc. of ACL*, pp. 177–180, 2007.

[8] Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. Phrase-based & neural unsupervised machine translation. In *Proc. of EMNLP*, pp. 5039–5049, 2018.

[9] Benjamin Marie and Atsushi Fujita. Unsupervised neural machine translation initialized by unsupervised statistical machine translation. *arXiv preprint arXiv:1810.12703*, 2018.

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *ICLR Workshop*, 2013.

[11] Tomoya Mizumoto, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. Mining revision log of language learning SNS for automated Japanese error correction of second language learners. In *Proc. of IJCNLP*, pp. 147–155, 2011.

[12] Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. Ground truth for grammatical error correction metrics. In *Proc. of ACL-IJCNLP*, pp. 588–593, 2015.

[13] Hwee Tou Ng, Siew Mei Wu, Yuanbin Wu, Christian Hadiwinoto, and Joel Tetreault. The CoNLL-2013 shared task on grammatical error correction. In *Proc. of CoNLL*, pp. 1–12, 2013.

[14] Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proc. of ACL*, pp. 160–167, 2003.

[15] Raymond Hendy Susanto, Peter Phandi, and Hwee Tou Ng. System combination for grammatical error correction. In *Proc. of EMNLP*, pp. 951–962, 2014.