格助詞と冠詞に着目した日英tree-to-string翻訳における単語アライメント表の改良

伊部 早紀　　松田 源立　　山口 和紀
東京大学
{ibe, matsuda, yamaguch}@graco.c.u-tokyo.ac.jp

1 はじめに

ツリーベース機械翻訳 [14] は2000年代に提案された翻訳手法であり，原言語あるいは目的言語の構文木を用いた確率に基づく翻訳を行う．従来のフレーズベース機械翻訳 [5] は構文木を使わずに単語列の意味対応と並べ替えを行う手法であり，語順が大きく異なる日英間の並べ替えに上手く対処できなかったが，この翻訳方式では構文木を用いることで言語の文法構造を捉え，語順の並べ替えに対処することができた．

ツリーベース機械翻訳を行うにあたっては，構文解析だけでなく，異言語間の構文木の対応を決定する単語アライメント表の作成が重要となる．従来，単語アライメントの計算には IBM モデル [2] および HMM モデル [13] を実装した GIZA++[9] や，統語的情報を用いて教師あり学習を行う Nile[12] などが用いられてきたが，これらは日英間に関して改善の余地がある．

そこで本研究では，日英 tree-to-string 翻訳 [3] において，日本語の格助詞および英語の冠詞に着目して単語アライメント表の誤りを適切に修正することで翻訳モデルの改善および翻訳精度の向上をはかった．

2 tree-to-string翻訳

tree-to-string 翻訳 [3] は，原言語における構文木と目的言語における単語列からなる翻訳規則を統計的手法を用いて学習し，翻訳を行う手法である．手順を以下に示す．

第一に，対訳コーパスに GIZA++[9] などを用い，単語アライメント表を作成する．これを原言語から目的言語，またその逆に対して作成し，grow-diag-final-and（gdfa）などのヒューリスティックを用いて 2 つの表を重ね合わせたものを最終的な単語アライメント表とする．

第二に，作成した単語アライメント表と原言語の構文木をもとに，翻訳規則とその確率を学習する．規則の抽出は GHKM アルゴリズム [4] に従う．

最後に，翻訳規則を利用して出力候補文をいくつか生成する．各候補文に対し翻訳規則の確率と言語モデルを用いて翻訳確率を計算し，翻訳確率が最も高い候補文を出力結果とする．

3 提案手法

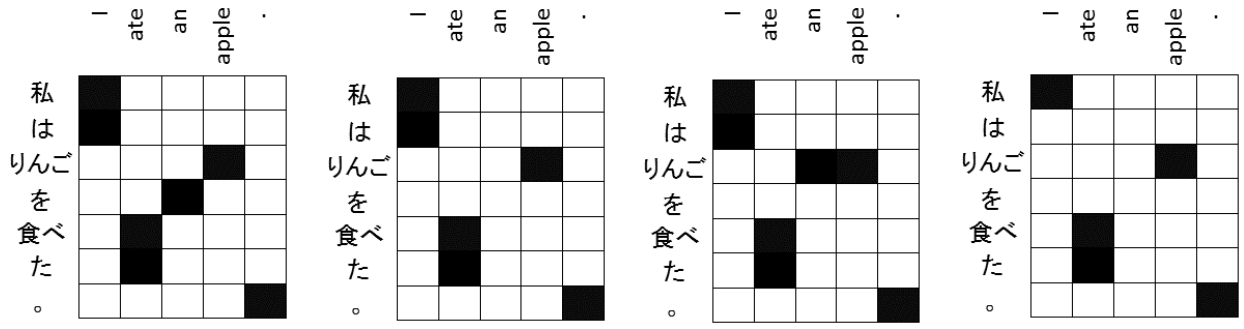
tree-to-string 翻訳において，翻訳規則の抽出は原言語の構文木と単語アライメント表に基づいて行われる．このため，単語アライメント表の精度が低いと正しい翻訳規則が抽出できない．そこで，単語アライメント表の誤りが多い格助詞および冠詞に相当する単語のアライメント位置を修正し，翻訳結果の精度向上を図る．

具体的には以下の 5 手法を提案する．

1. ErmA（remove Articles）
   英語の冠詞が，どの日本語の単語にもアライメントされないようにした．図 1(a) の単語アライメント表に対して ErmA を行った結果を図 1(b) に示す．

2. EcrA（correct Articles）
   英語の冠詞について，“直後の名詞” のアライメント先と同じ場所にアライメントさせるように修正した．この時，冠詞の判定には英語の構文木を用いる．直後の名詞が複数ある場合は，冠詞から始まる名詞節のうち最も後ろにある名詞の単語を “直後の名詞” とした．図 1(a) の単語アライメント表に対して EcrA を行った例を図 1(c) に示す．

3. JrmC（remove Cases）
   日本語の格助詞のうち，英語に対応する単語がな



(a) gdfa **(b)** gdfa+ErmA **(c)** gdfa+EcrA **(d)** gdfa+JrmC

図 1: 各手法による単語アライメント表の違い

表 1: コーパスのサイズと単語数．train は学習に，dev はパラメータチューニングに，test はテストに用いた．

	train		dev	test
	文	単語	文	文
日本語	908.1K	162.3K	1.8K	1.8K
英語		326.8K		

い主格の「が」「は」，目的格の「を」について，どの単語にもアライメントされないように修正した．図 1(a) のアライメントに対して JrmC を行った場合，図 1(d) のように，格助詞「は」「を」のアライメント点が外れる．

4. ErmA+JrmC
   ErmA を行った後，JrmC を行う．

5. EcrA+JrmC
   EcrA を行った後，JrmC を行う．

4 実験

4.1 実験データ・方法

対訳コーパスとしては ASPEC[7] を用いた．コーパスのサイズと単語数を表 1 に示す．英語の構文解析には Stanford Parser[6] を，日本語のトークン化には KyTea を，日本語の構文解析には Ckylark[10] を用いた．前処理として [8] を参考に，単語数 51 以上の文は学習データから取り除いた．デコーダは Travatar[8] を用い，日英および英日の翻訳を行った．gdfa により作成された単語アライメント表を用いて学習したモデルをベースラインとし，提案手法による修正を加えた単語アライメント表で学習したモデルと翻訳精度を比較した．

4.2 評価指標

翻訳精度の評価指標には BLEU[11]，RIBES[15]，METEOR[1] の 3 つを用いる．ただし BLEU はコーパス単位および文単位の双方で測り，文単位の評価値についてはベースラインと比較して平均値が上昇したものに関して片側検定における p 値を求めた．また METEOR は日本語の評価には対応していないため日英翻訳の評価のみとなる．

4.3 実験結果と考察

実験結果を表 2 に示す．表 2 より，日英翻訳においてはコーパス単位の BLEU は EcrA+JrmC の場合が最大で+0.50，RIBES は ErmA の場合に最大で+0.17，METEOR は ErmA+JrmC の場合に最大で+0.13 の上昇を確認できた．また表 3 より，英日翻訳においてはコーパス単位の BLEU は EcrA+JrmC の場合が最大で+0.42，RIBES は JrmC の場合に最大で+0.26 の上昇を確認できた．テスト用のデータが少ないため p 値は統計的に明確に差があると言えるほど小さくないが，傾向としては単語アライメント表を正しく修正することで翻訳精度の向上がみられることが分かった．また抽出できた翻訳規則数については，BLEU 値の向上に比例して増加していることが分かる．

次に，抽出された翻訳規則の変化の具体例を表 3 に示す．学習データ中のある対訳文について，ベースラインで抽出できた翻訳規則表 3(a) と，ベースラインに EcrA+JrmC の修正をかけたアライメントで抽出できた翻訳規則表 3(b) を比較した．(a) では抽出できなかった規則が (b) で現れており，また (b) の方が，

表 2: 翻訳結果の精度評価と，学習時に抽出できた翻訳規則数．ベースラインである gdfa での単語アライメントに修正を施しモデルの学習を行った結果．太字は gdfa と比較して上昇した値，その中で最も高い値に下線を施した．

日→英	BLEU		RIBES	METEOR	翻訳規則数
	コーパス単位	文単位			
gdfa	20.79	14.75	69.67	30.62	18.9M
+ErmA	**20.97**	**14.98** (p=0.34)	**69.84**	**30.73**	20.8M
+EcrA	**21.06**	**14.98** (p=0.34)	**69.71**	30.62	21.3M
+JrmC	**20.82**	14.65	**69.82**	30.49	20.5M
+ErmA +JrmC	**20.94**	**15.05** (p=0.30)	69.47	**30.75**	21.8M
+EcrA +JrmC	**21.29**	**15.13** (p=0.25)	69.45	**30.63**	22.3M

英→日	BLEU		RIBES	METEOR	翻訳規則数
	コーパス単位	文単位			
gdfa	32.50	28.89	77.24	-	21.9M
+ErmA	**32.57**	28.80	77.22	-	22.5M
+EcrA	31.64	28.15	77.04	-	22.5M
+JrmC	**32.66**	**29.23** (p=0.31)	**77.50**	-	23.4M
+ErmA +JrmC	**32.62**	**29.07** (p=0.40)	**77.47**	-	23.6M
+EcrA +JrmC	**32.92**	**29.33** (p=0.26)	**77.47**	-	24.2M

より一般的で多くの文に適用可能な翻訳規則が抽出できていることが分かる．このように，単語アライメント表を正しく修正することで抽出できる翻訳規則がより一般的になったため，未知のテストデータに対して適用可能な規則が増え，翻訳精度が向上したと考えられる．

5 つの提案手法の中では，最も BLEU 値が高く，かつ翻訳規則数が多い EcrA+JrmC が最も有効な手法と考えられる．

5 おわりに

アライメント表を作成する IBM モデルや HMM モデルでは，言語間で語順などが大きく異なる日本語－英語においては単語の対応が取りづらく，しばしば意味が対応していない単語のペアにアライメントされることがある．本研究では，日本語の格助詞と英語の冠詞に着目し，目的言語において意味が対応する単語が存在しない場合はアライメントを外すことで，翻訳モデルの改善が見られ，翻訳精度の向上につながることを確認した．

今回は単語アライメント表の改良で翻訳精度の向上を図り，翻訳規則数の増加が性能の向上につながることが示唆されたが，実際には提案手法を用いても翻訳結果が改善されなかった例も多く存在した．原因としては，学習データ中の対訳文の意味の対応が取れていないものが多いこと，日本語の構文解析に失敗していること，翻訳結果の意味が正しくても評価指標の値が低いことなどが挙げられる．これらに対する改善策として，コーパス中のノイズの除去，日本語の構文解析器の精度向上，翻訳自動評価指標の精度向上などが挙げられる．

また，全ての評価指標において，日英翻訳よりも英日翻訳の方が高い値が得られた．原言語の構文木を用いる tree-to-string 翻訳では構文木の正確さが翻訳規則を定める一つの要素となっており，日英間の翻訳に関しては，精度の高い構文解析器が存在する英語を原言語とした方が良い翻訳結果が出ることが示唆された．今後は目的言語の構文木を用いる string-to-tree 翻訳においても同様の実験を行い，結果を比較検討していきたい．

表 3: 日本語→英語の翻訳規則の例

英	the postoperative scoring points for evaluation in the hip joint were improved .
日	術後 の 股 関節 評価 点 は 改善 さ れ た 。
(a)	gdfa で抽出された翻訳規則の例 名詞 p (助詞の p (名詞 (“術後”) 助詞の (“の”)) 名詞 p (x_0:名詞 p 名詞 p (x_1:名詞 名詞 (“点”)))) → the postoperative scoring points for x_1 in the x_0
(b)	gdfa+EcrA+JrmC で抽出された翻訳規則の例 名詞 p (x_0:助詞の p x_1:名詞 p) → x_0 x_1 名詞 p (x_0:名詞 p x_1:名詞 p) → x_1 x_0

参考文献

[1] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proc. ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72, 2005.

[2] P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[3] M. Galley, J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang, and I. Thayer. Scalable inference and training of context-rich syntactic translation models. In *ACL*, pages 961–968, 2006.

[4] M. Galley, M. Hopkins, K. Knight, and D. Marcu. What's in a translation rule. Technical report, DTIC Document, 2004.

[5] P. Koehn, F. J. Och, and D. Marcu. Statistical phrase-based translation. In *NAACL HLT-Volume 1*, pages 48–54, 2003.

[6] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *ACL System Demonstrations*, pages 55–60, 2014.

[7] T. Nakazawa, M. Yaguchi, K. Uchimoto, M. Utiyama, E. Sumita, S. Kurohashi, and H. Isahara. ASPEC: Asian scientific paper excerpt corpus. In *Proc. of the Ninth International Conference on Language Resources and Evaluation (LREC 2016)*, 2016.

[8] G. Neubig. Travatar: A forest-to-string machine translation engine based on tree transducers. In *ACL System Demonstrations*, pages 91–96, 2013.

[9] F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *CL*, 29(1):19–51, 2003.

[10] Y. Oda, G. Neubig, S. Sakti, T. Toda, and S. Nakamura. Ckylark: A more robust pcfg-la parser. In *NAACL-HLT*, pages 41–45, 2015.

[11] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. BLEU: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.

[12] J. Riesa and D. Marcu. Hierarchical search for word alignment. In *ACL*, pages 157–166, 2010.

[13] S. Vogel, H. Ney, and C. Tillmann. HMM-based word alignment in statistical translation. In *Proc. 16th conference on CL-Volume 2*, pages 836–841. ACL, 1996.

[14] K. Yamada and K. Knight. A syntax-based statistical translation model. In *ACL*, pages 523–530, 2001.

[15] 平尾努, 磯崎秀樹, K. Duh, 須藤克仁, 塚田元, and 永田昌明. RIBES ：順位相関に基づく翻訳の自動評価法. **言語処理学会第 *17* 年次大会発表論文集**, pages 1111–1114, 2011.