自然言語処理 137－3
（2000．6．1）

統計的な係り受け解析結果を用いた対訳表現抽出について

山本 薫，松本 裕治
奈良先端科学技術大学院大学 情報科学研究科
{kaoru-ya,matsu}@is.aist-nara.ac.jp

近年、対訳コーパスが普及し、文レベルの対応や連語を含む語レベルの対応をとる手法が活発に提案された。一方、中間に位置する文節レベルの対応についての研究は、あまりされていない。機械翻訳では、目的語に応じた動詞の訳し分けなど、語の依存関係まで考慮する場合が多い。このため、文節レベルでの対応は重要な課題である。

本稿では、統計的係り受け関係結果を用いた文節レベルの対訳表現抽出の手法を提案する。統計的な係り受け解析は、規則主導型と違い、入力文を制限することがない。完璧な解析結果は望めないが、部分的に有用な結果が多く含まれているものと思われる。我々は、言語族が違う二言語間対応でも依存関係は保たれるという予測のもと、係り受け関係を使って候補パターンを生成し、日英の候補パターン間の類似度を計算し、対応の高いものから対訳表現として抽出した。

実験は、約1,3000対訳文を使った。人手で評価したところ77%の適合率が出て、部分的な依存関係を保持した対訳表現が抽出された。

[キーワード] 対訳表現，機械翻訳，知識獲得

Translation Pattern Acqusition using Dependency Structures

YAMAMOTO Kaoru, MATSUMOTO Yuji
Graduate School of Information Science, Nara Institute of Science and Technology
{kaoru-ya,matsu}@is.aist-nara.ac.jp

This paper describes a method to find translation patterns from parallel corpora using dependency structure. Our approach is based on the assumption that the word ordering may not necessarily coincide between two languages, but the dependency structure between words will be preserved in many cases. We use statistical dependency parsers to get candidate dependency relations between base phrases in a sentence. Our method is tested with approx. 13000 Japanese–English sentence pairs and achieved 77% precision. Translation patters corresponding at phrase-level are extracted successfully.

[Keyword] translation patterns, machine translation, lexical knowledge acquisition

1 はじめに

近年、電子化された対訳コーパスをもとに文や連語を含む単語の対応をとる手法が提案されている。これらの手法は、(1) 形態素解析や NP-recognizer で得られる、それぞれの言語の語順や語源 (cognate) の対応を主な手かがりとしている、(2) 統計的な類似尺度を使って対訳表現を推定する、という特徴がある。[4] [9]

我々は、対訳表現抽出において、(1) で挙げた手がかりでは不十分と考える。なぜなら、日本語と英語のように異なる言語族の場合、語源を手がかりにしにくい[1]。語順は、名詞句などの狭い範囲に限定すれば、修飾関係が閉じているので手かがりとして有効である。しかし、日英対応においては、基本的な文の構造が異なるため、離散的な対訳表現は抽出できない。

一方、統計的手法を用いた構文解析技術が向上しつつある。従来の規則主導型の構文解析と比べ、統計的手法は、煩雑なチューニングがいらず、長文や複文の解析も扱える。精度を犠牲にする欠点はあるが、ある確率で確信できる答を出力する利点がある。現状では、一文の完全な構文解析は困難である。しかし、有効な部分解析結果は多く含まれている。

そこで、本研究では、二言語間対応の多くの場合において、語の依存関係は保たれることに着目した。そして、部分的な係り受け関係を利用することにより、(部分) 構造を考慮した対訳表現が抽出できるのではないかと考え、対訳コーパスからの候補パターン生成に部分的係り受け関係を利用した。

提案する手法は、次の 2 つの段階から構成される。

1. 日英対訳コーパスに複数回出現し、かつ、係り受け関係がある候補パターンを生成する。
2. 生成された候補パターン間で類似度計算を行い、対応関係の強いものから順に抽出する。

本稿の構成は、以下の通りである。2 章で、統計的な係り受け解析とそれを利用した候補パターン生成について述べる。3 章で、対訳パターンの抽出アルゴリズムを紹介する。4 章で、実験結果を報告し、考察する。5 章で、関連研究との違いを説明し、6 章でまとめる。

[1] カタカナ表記された外来語を語源とみなす場合もできるが、今の形態素解析では、考慮されていない。

2 係り受けを使った候補パターン

2.1 統計的係り受け解析

係り受け解析は、依存文法をもとにしている。我々は、藤尾らの統計的係り受けモデルを用いた [7]。このモデルは、文節の主辞や係り関係等の文節属性に基づいている。全体の流れは、文節を規則で区切り、係り受け関係を文節の主辞および関係語の共起確率で推定している。このとき、係り受けは非交差であり、文末以外の文節は必ず一つの係り先をもつという制約がある。

文節区切り規則は品詞および語の列を正規表現で用意した。日本語の文節は、一般的に用いられている 1 つ以上の内容語と 1 つ以上の機能語を基本単位とした。英語は「文節」という単位がないため、次のような方針で区切られたものを文節と見なした。

- BaseNP(再帰的に NP を含まない NP)。
- 前置詞と BaseNP が連結しているもの。
- 助動詞や完了形などを主動詞とまとめた動詞的表現。
- 時間や日付表現。

我々は、上記の統計的係り受けモデルを基に実装された日本語の係り受け解析器 jdep と英語の係り受け解析器 edep を利用した。jdep は、EDR コーパスを用いて係り受け関係を推定しており、86%の精度がでていると報告されている [7]。一方、edep は、PennTree Bank コーパスを用いて学習をした。edep は実験的に作成されたもので、客観的な評価はされていない。

係り受け解析においては、ある一文のみに注目して、各文節の唯一の係り先を決定するのは困難な場合がある。藤尾らの統計的な係り受け解析モデルでは、精度を犠牲にして再現率をあげること (以下、冗長解析と呼ぶ) が可能である。そこで、本研究では、統計的に最良と思われる係り受け関係のみを考慮した「統計最良モデル」と統計的に曖昧と推測される係り受け関係も含めて考慮した「統計曖昧モデル」を用意した。

「統計曖昧モデル」では、統計的に推定された係り受け関係の信頼度をもとに、冗長に解析させ、複数の係り先を許す。部分解析において、尤らしい係り受け関係は多く出現することになり、その類似度

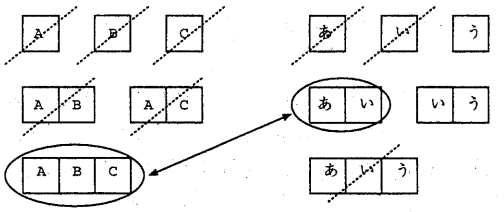

図 1: 対訳アラインメント: 対訳ペアは丸で囲む。対訳ペア確定後、対象外となる候補パターンには斜線を引く。

は上がるものと思われる。したがって、係り受け関係での曖昧性を吸収しつつ、尤らしい係り受け関係を保った対訳ペアが抽出されることを期待する。

2.2 候補パターン生成

係り受け関係を利用した候補パターンは、文の部分依存木に相当する。一方、語順を利用して n-gram 的に連結した候補パターンは文の部分文字列に相当する。

ここでは、候補パターンの長さとは、候補パターン生成に使った文節の数を指す。一般に、長さ n の候補パターンは、長さ $n-1 (n > 1)$ 以下の候補パターンから生成される。元になった候補パターンを親パターンと呼び、生成されたものを子パターンと呼ぶ。

候補パターンとその親および子パターンの関係は、対訳アラインメントの際に使う。例えば、対訳文から英語の候補パターン {A, B, C, AB, AC, ABC} と日本語の候補パターン { あ, い, う, あい, いう, あいう } が生成され、<ABC, あい>が対訳ペアとして確定されたとする。(図 1 を参照。) このとき、"ABC"と重なりあう親子パターンは、"あい"とは対訳ペアとならないであろうと推測される。同様に、"あい"と重なりあう親子パターンも"ABC" とは対訳ペアにならないと推測される。このヒューリスティックを対訳ペア推定に使うため、候補パターン生成と同時に親および子パターンも求める。

係り受け関係を利用した候補パターン生成手順は、以下の通りである。

1. 何回以上コーパスに出現したものを対象にするか (min) と候補パターンの長さ (n) を決める。
2. 各文について、以下の処理を行う。
   (a) 候補パターンを生成する。
      i. 長さ n の候補パターンを係り受け関係を有する長さ $(n-1)$ 以下の候補パターンの組み合わせで作成する。係り元パターンはそのままだが、受け先パターンの関係語は削除する。
      ii. 長さ 1 の候補パターンを作成する。この関係語は削除する。
   (b) 候補パターンの集合から、それぞれの親パターンと子パターンを特定する。<候補パターン, 親子パターン集合>の対を格納する。
3. min 回以上出現した候補パターンとその親子パターンを集計する。

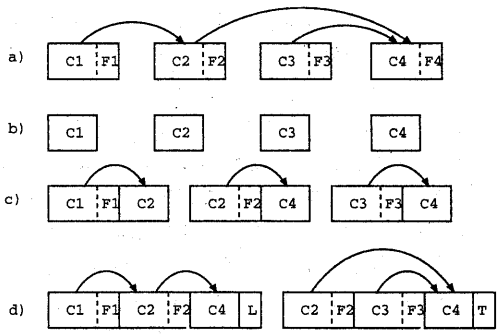

図 2: 候補パターン: a) 係り受け解析結果, b) 長さ 1 の候補パターン, c) 長さ 2 の候補パターン, d) 長さ 3 の候補パターン

図 2 に、長さ 1,2,3 の候補パターン生成過程を示す。a) は、文の係り受け解析結果を表す。四角が文節区切りで、矢印が係り受け関係を示す。文節の内容語と関係語を点線で区切り、Cx は文節の内容語で、Fx は文節の関係語である。この文から生成される候補パターンを b),c),d) に示す。長さ 3 以上の候補パターンでは、異なる部分依存木 (DAG) が生成されるので、ラベルを付与する。(長さ 3 の場合、2 種類の DAG が可能で、それぞれ L,T とラベル付けしている。) 英語の場合、文の後方から前方の文節に係る場合もあるため、候補パターンは必ずしも語順通りにならない。

統計的な係り受け解析モデルでは、各係り受け関係に対して確率が付与されている。しかし、候補パ

ターン生成では、確率値に関係なく、ある信頼度以上で出力された係り受け関係をすべて利用した。

本手法では、対応する単位が文節レベルになっている。このため、任意長の自立語の文字列を対象とした北村らの手法[9]と比べ、data sparseness を起こしやすい。そこで、(1) 各文節を原形に戻す「原形モデル」、(2) 係り受け関係の推定に用いた文節の主辞に縮退する「主辞モデル」を用意した。

「原形モデル」では、"director's meeting"と"an important meeting"は、文節の中心語は同じ"meeting"でも修飾部が違う ("director's"と"important") ため、異なる候補パターンとして扱う。一方、「主辞モデル」では、両文節は、"meeting"に縮退するため、同じ候補パターンとして扱う。これにより、候補パターンの出現回数の底上げ効果を狙うと共に、係り受け関係が活かされる候補パターン (例えば "meeting" と "be held") が抽出されやすいという効果を期待する。

また、ある文節が直後の文節に係っていると仮定すれば、文節を連結した候補パターンも生成できる。これを「文節 n-gram モデル」として用意した。北村らの手法では、文節の区切り境界を意識しないため、不適切な候補パターンが生成されやすい。それに比べ、「文節 n-gram モデル」では、意味のある区切れ境界を利用していると期待できる。

候補パターン生成では、出現回数の多い順に、候補パターンファイルと親子パターンファイルを作成する。これらは、英語と日本語にそれぞれ独立に用意し、次節で説明する対訳表現抽出アルゴリズムの入力となる。

3 対訳表現抽出

3.1 対訳ペアの推定方法

対訳ペアの推定は、日英の候補リストの全組み合わせの類似度計算を行い、対応関係の強いペアから順に抽出する。

類似度は、北村ら[9]で提案された、重み付き Dice 係数で計算する。

$$sim(\langle p_j, p_e \rangle) = (log_2 f_{je}) \frac{2 f_{je}}{f_j + f_e} \qquad (1)$$

p_jは日本語の候補パターン、p_eは英語の候補パターン、f_jは日本語コーパスのp_jの出現回数、f_eは英語コーパスのp_eの出現回数、f_{je}はp_jとp_eの同時出現回数を示す。

候補パターンは、コーパスに出現した場所 (文番号) 情報を持つ。対訳抽出アルゴリズムでは、出現回数の多いものから順に対訳候補の対象を段階的に拡大する。p_eとp_jが対訳ペアとして確定されたら、p_e、p_j、p_e又はp_jと重なりあう候補パターンからp_eとp_jが同時に出現した場所 (対訳文番号) を削除する。

一度対訳ペアとして確定されたp_eとp_jの同時出現場所を次の繰り返しから数え上げない。このため、p_eとp_jの出現回数が (ゆえに、類似度も) 低下し、他の対訳表現が新たに抽出される。また、前節で説明した対訳アラインメントにより、p_eとp_jの同時出現対訳文で生成されたp_eとp_jと重なりあう候補パターンも、次の繰り返しから数え上げの対象外とする。

北村らのアルゴリズムでは、毎回、対訳ペアとして登録されているものを除いた出現回数を数え上げ、類似度を計算している。しかし、出現回数が更新されるのは、対訳ペアとして確定した候補パターン、および、それらと重なりあう候補パターンに限る。その他の候補パターンの類似度は影響をうけない。

我々のアルゴリズムでは、不必要な類似度の再計算を省くために次の改良をした。タイムスタンプを導入し、ある候補パターンp_eの類似度最大ペアp_jは、p_eとp_jの出現場所を更新した後に決定されているという特性を使う。つまり、候補パターンp_eの類似度最大ペアp_jのタイムスタンプ (ts_best) がp_eの出現場所更新のタイムスタンプ (ts_update) より後であれば、類似度結果は有効である。この場合、ts_best 以降に出現場所が更新された候補パターンp_j'に対してのみ類似度を計算し、ts_best に計算されたp_eと類似度最大ペアp_jの類似度と比較しながら、類似度最大ペアを見つける。

これを内部に更新リストをもって実現する。対訳ペアとして確定した後、出現場所が更新された候補パターンを順にリストの先頭へ追加する。p_eにおいて最大の類似度をもつp_jを探すときは、相手言語の更新リストの先頭からp_j' の ts_update とp_e の ts_best を比較し、暫定的なp_eの最大類似度相手p_jと ts_best 以降に更新されたp_j' を比較する。このように類似度計算をタイムスタンプの差分だけでお

こなうことにより、処理時間の高速化を図った。

3.2 対訳抽出アルゴリズム

1. 入力：日英の候補パターンファイルと親子パターンファイル、アルゴリズムの各ステージにおける出現回数の閾値 th、最低出現回数 min
2. 閾値 th 以上出現した候補パターン p_j と p_e について以下の処理をする。
   (a) p_j において最大の類似度をもつ p_e' 探す。
   (b) p_e において最大の類似度をもつ p_j' を探す。
   (c) p_j と p_j'、p_e と p_e' が同じで、かつ、sim($\langle p_j, p_e \rangle$) が $\alpha log_2(th)$ 以上であれば、p_j と p_e を対訳表現として登録する。
   (d) p_j と p_e が登録された場合、p_j と p_e と重なりあう候補パターンからの出現場所から p_j と p_e の同時出現場所を引く。
3. 対訳表現抽出数が β 以下であれば、出現回数の閾値 th を下げる。閾値 th で繰り返し 2 の処理をする。出現回数の閾値 th が最低出現回数 min 未満になったら、終了する。
4. 出力：対訳表現ファイル

αは、[0,1] の任意の値でよい。今回の実験では 1 に固定した。重み付き Dice 係数においては、一番厳しい条件に設定したことを意味する。βも 1 以上の値であればよい。今回の実験では 10 にした。

4 結果と考察

4.1 結果

実験は、あらかじめ対訳文の対応がつけられた、日経ビジネスレター例文集 (13577 文)[6] を使った。前処理に必要な形態素解析は茶筌 [5] を使い、係り受け解析は jdep, edep を利用した。

閾値 th は 100 からはじめ、2 まで下げた。以下に述べる各実験結果では、閾値の段階別に対訳ペアが抽出された数「合計」、そのうちの正解の数「正解」、「精度」、「累積精度」を記す。評価は人手で行い、対訳コーパスの出現形に復元した後、そのまま翻訳辞書に登録できるものを正解としている。「精度」は、各閾値 th における正解数と抽出数の比率で、「累積精度」は、閾値 th 以上の正解数と抽出数の比率である。

閾値	正解	合計	精度	累積精度
100	0	0	–	–
50	0	0	–	–
25	6	6	100.00	100.00
12	7	7	100.00	95.00
10	6	7	85.71	95.83
9	4	4	100.00	92.30
8	13	13	100.00	97.29
7	10	13	76.92	92.00
6	19	20	95.00	92.85
5	29	29	100.00	94.94
4	67	72	93.05	94.15
3	150	164	91.46	82.93
2	678	934	72.59	**77.93**

表 1: 統計最良モデル

はじめに、各文節を原形に戻し、係り受け解析の最良候補のみを使った「統計最良モデル」と係り受け解析の曖昧性を許した「統計曖昧モデル」の実験をした。結果を表 1、表 2 に示す。「統計曖昧モデル」における冗長解析は Ratio Next 型 [7] とし、信頼度 0.5 以上の係り受け関係にあるものをすべて同じ重さで考慮した。

表 3 に、「統計最良モデル」で抽出された正解例を示す。+ は、文節区切りを示し、_ は形態素区切りを示す。対訳ペアは、コーパスの出現形に復元してある。

表 4 に、「統計最良モデル」で抽出された半正解例を示す。どちらか一方の候補パターンの一文節の削除によって正解に変換できる抽出ペアを半正解とした。削除されるべき文節を () で囲む。

次に、統計的に最良と思われる係り受け解析結果のみを考慮して、各文節を主辞に縮退した「主辞モデル」を実験してみた。結果を表 5 に示す。data sparseness にどのくらい有効かどうか調査するのが目的である。

最後に、英語と日本語の片方を「文節 n-gram モデル」にし、もう片方を「統計最良モデル」にして、実験を行った。結果を表 6、表 7 に示す。対訳表現抽出において、文節区切り情報か係り受け関係のどちらがより効果的な働きをしているのかを調べるのが目的である。

英語	日本語	類似度
apply+for_the_position	職_に+応募_いたす	2.2157
thank+you+in_advance	前もって+お願い+申し上げる	1.6000
be+enclosed+a_copy	1_部_同封_いたす	1.0566
be_writing+to_let+know	書状_をもって+お知らせ_いたす	1.0566
upcoming_borard+of_director_s´_meeting	次回_の+取締役_会	1.0000
will_have+to_cancel	中止_せ_ざる_を+得_なく+なる	1.0000
have+high_hope	大いに+期待_する	1.0000
business+is_expanded	商売_は+発展_する	1.0000
we+have_learned+from_your_fax	貴_ファックス_で+知る	1.0000
leaving+in+about_ten_days	約_1_0_日_後+出発	1.0000
get+you+in_close_business_relationship	緊密_な+取引_関係_を+築く	1.0000
we+are_inquiring+regarding	に関し+お尋ね_いたす	1.0000
pay+special_attention	特別_の+注意_を+払う	1.0000

表 3: 統計最良モデルで抽出された正解例。+は文節区切りを示し、_は形態素区切りを示す。

英語	日本語	類似度
(have_been_pleased)+to_serve+as_thier_main_banker	主力_銀行_と+なる	1.0000
hotel_new_ohtani	ホテル_ニューオータニ_で+(開催_する)	1.0000
assets_position+(in_good_shape)	資産_状態	1.0000
(have_been_placed)+into_our_file	私ども_の+ファイル	1.0000
(put)+one_month_limit	1_ヶ月_の+期限	1.0000
past_tuesday	火曜日_に+(亡くなら_れる)	1.0000

表 4: 統計最良モデルで抽出された半正解例。削除されるべき文節を () で囲む。

閾値	正解	合計	精度	累積精度
100	0	0	–	–
50	0	0	–	–
25	6	6	100.00	100.00
12	7	7	100.00	100.00
10	6	7	85.71	95.00
9	4	4	100.00	95.83
8	13	13	100.00	97.29
7	11	13	84.61	94.00
6	18	19	94.73	94.20
5	29	29	100.00	95.91
4	68	73	93.15	94.73
3	118	126	93.65	94.27
2	688	1227	56.07	**63.51**

表 2: 統計曖昧モデル

閾値	正解	合計	精度	累積精度
100	0	0	–	–
50	2	2	100.00	100.00
25	12	12	100.00	100.00
12	52	52	100.00	100.00
10	17	17	100.00	100.00
9	20	21	95.23	99.03
8	21	21	100.00	99.20
7	27	28	96.42	98.69
6	40	41	97.56	98.45
5	50	50	100.00	98.77
4	128	133	96.24	97.87
3	193	199	96.98	97.56
2	601	790	76.07	**85.13**

表 5: 主辞モデル

4.2 考察

表 3 にみるように、係り受けを使った手法においても、訳し分け ("thank you/ありがとう" と "thank you in advance/前もってお願い申し上げます") ができた。日本語の主語省略 ("I must object/反対します") や、統語構造が違うため逐語訳で困難なもの ("be writing to let know/書状をもってお知らせする) も抽出できた。

本手法の特徴は、対訳ペアを機能語を含めて抽出できる点にある。しかし、長さ 1 の候補パターンの場合、文節の機能語を省くため、"(at) least/少なくとも"のような前置詞を含めた慣用表現の抽出が

閾値	正解	合計	精度	累積精度
100	0	0	–	–
50	0	0	–	–
25	6	6	100.00	100.00
12	7	7	100.00	100.00
10	6	7	85.71	95.00
9	4	4	100.00	95.83
8	13	13	100.00	97.29
7	9	11	81.18	93.75
6	18	19	94.73	94.02
5	29	30	96.66	94.84
4	69	74	93.24	94.15
3	118	124	95.16	94.57
2	695	969	71.72	**77.05**

表 6: 英語:文節 n-gram モデル 日本語:統計最良モデル

閾値	正解	合計	精度	累積精度
100	0	0	–	–
50	0	0	–	–
25	6	6	100.00	100.00
12	7	7	100.00	100.00
10	6	7	85.71	95.00
9	4	4	100.00	95.83
8	13	13	100.00	97.29
7	11	13	84.61	94.00
6	18	19	94.73	94.20
5	29	29	100.00	95.91
4	66	72	91.66	94.11
3	116	120	96.66	95.17
2	679	978	69.42	**75.31**

表 7: 英語:統計最良モデル 日本語:文節 n-gram モデル

できないという問題がある。

どのモデルも、閾値 3 までは、精度は良いものの、抽出数が少ない。抽出される対訳ペアの大半が、長さ 1 の候補パターン同士の対訳ペアである。逆に、閾値 2 まで下げると、精度が突然落ちる。この段階にきて、長さ 2 以上の候補パターンが対象になり、表 4 に見られる半正解、もしくは、部分対訳になっているもの[2]が多く抽出されるからと考えられる。

部分対訳が多く抽出される現象は、(1) データ表記の揺れ、(2) 係り受け解析間違い、の 2 点に起因すると考えられる。(1) は、日本語の表記方法が多様なために起きる。"put a one-month limit"では、"一ヶ月の期限をつける"と"一ヶ月の期限を付ける"に対応していた。(2) は、解析精度が向上すれば改善される。

「統計曖昧モデル」と「統計最良モデル」を比較すると、後者モデルのほうが、精度が良い。これは、曖昧性を許した解析にしても、尤らしい係り受け関係が多く出現するため類似度が高くなる、という効果が得られなかったためと考えられる。英語の候補パターン数で比較すると、「統計最良モデル」は 14705 個、「統計曖昧モデル」は 34234 個であった。つまり、曖昧性を許したことにより 19529 個の候補パターンが新たに生成された。更に、増加分のうち、14063 個 (72%) が 2 回しか出現していない。曖昧性を許すことにより、出現回数の底上げ効果を狙ったが、実際は、候補パターンの種類が増加し、data sparseness になった。

また、両モデルの正解総数 (「統計最良モデル」が 989 個、「統計曖昧モデル」が 968 個) のうち、904 個が同じである。このことから、曖昧性を許して増加した候補パターンの大半は対訳ペアにならなかったことがうかがえる。以上の理由から、「統計曖昧モデル」の精度が悪くなったと推測する。

「主辞モデル」と「統計最良モデル」を比較すると、前者モデルのほうが精度がよい。これは、主辞に縮退することにより抽象化をされたため、抽出数も少ないが、data sparseness への解消につながったとも考えられる。しかし、主辞モデルでは、抽出ペアの大半が長さ 1 の候補パターン同士であった。係り受け関係を活かした対訳パターンが抽出されなかったことをふまえると、あまり有効な抽象化とは言いがたい。

「文節 n-gram モデル」と「統計最良モデル」を混合させて対訳抽出を行ったモデルでは、精度においては、大差がみられなかった。英語を n-gram にしても係り受けモデルと差がないのは、英語は語順制約が厳しいからだと推測される。一方、日本語は、比率は少ないが、語順より係り受け関係のほうが有用であるのが、実験結果から、うかがえる。

5 関連研究

語順や語源を用いた対訳表現方法は数多く発表されている。たとえば、Smadja [4] らはコロケーション抽出方法を提案した。英語のコロケーション候補をあらかじめ作成し、対応するフランス語のコロケーションを抽出した。北村ら [9] は、ある出現回数以上の任意の文字列を英語と日本語それぞれに集

[2]正解に変換するために、抽出ペアの 2 文節以上の削除が必要なもの。片方言語に限らない。

めたものを候補パターン集合とし、重み付き Dice 係数を使って、対訳表現を推定している。Haruno ら [1] は、ワードソーティングであらかじめ有意な単語列を抽出し、対応付けを行っている。この手法では、単語レベルの対応をボトムアップに対応付けながら、離散的な対訳表現も抽出している。

我々の手法は、[4] と違い、双方向に候補パターンを生成する。また、[1] と違い、あらかじめの単語レベルでの対応を必要とせず、一段階で離散的な対訳表現が抽出できる。さらに、[9] や [1] と違い、係り受け解析結果を利用しているため、文の部分構造を意識した対訳表現が抽出できる。

依存構造を用いた対訳表現の抽出も発表されている。たとえば、Matsumoto ら [2]、北村ら [8]、Meyers ら [3] は、文を解析して、構造照合を行うことによって、翻訳規則を抽出している。

彼らの手法は、ルールベースで記述した文法を基に対訳文をそれぞれ解析し、依存構造によって明らかになる文全体の構造照合を行っている。一方、我々の手法は、統計ベースの係り受けパーザーを利用しており、かつ、依存木の部分照合を目的としている。これは、(複雑な) 文の完全な解析が難しいため、また、部分照合を目指すことにより、頑健さと被覆率を向上できると思われる。

6 まとめ

本稿では、統計的な係り受け解析結果を利用して、対訳コーパスから対訳表現を抽出する方法を提案した。人手で評価した結果、精度は 77%で、部分 (依存) 構造を考慮した対訳表現が抽出された。

係り受けでの曖昧性を吸収した対訳表現抽出を目指したが、英語の係り受け解析の精度が不安定なため、期待したほどの結果は得られなかった。係り受け解析精度が向上すれば、この問題は改善されると思われる。

data sparseness 解消のために、各文節を (1) 原形に戻す、(2) 主辞に縮退するモデルを試した。しかし、目的語に応じた動詞の訳し分け等に役立つ、係り受け関係が活かされる対訳表現の抽出にはつながらなかった。今後、抽象化に工夫がいるものと思われる。

今後は、対訳表現抽出において、data sparseness に効果的な抽象化の方法や、抽出された対訳表現を部分翻訳に活かしていく手法を考えたい。

参考文献

[1] Masahiko Haruno, Satoru Ikehara, and Takefumi Yamazaki. Learning bilingual collocations by word-level sorting. In *COLING-96: The 16th International Conference on Computational Linguistics*, pp. 525–530, 1996.

[2] Yuji Matsumoto, Hiroyuki Ishimoto, and Takehito Utsuro. Structural matching of parallel texts. In *ACL-93: 31st Annual Meeting of the Association for Computational Linguistics*, pp. 23–30, 1993.

[3] Adam Meyers, Roman Yangarber, and Ralph Grishman. Alignment of shared forests for bilingual corpora. In *COLING-96: The 16th International Conference on Computational Linguistics*, Vol. 1, pp. 460–465, 1996.

[4] Frank Smadja, Kathleen R. McKeown, and Vasileios Hatzivassiloglou. Translating collocations for billingual lexicons: A statistical approach. In *Computational Linguistics*, Vol. 22(1), pp. 1–38, 1996.

[5] 松本裕治, 北内啓, 山下達雄, 平野善隆, 松田寛, 浅原正幸. 日本語形態素解析システム『茶筌』 version 2.0 使用説明書 第二版. NAIST Technical Report, NAIST-IS-TR99012, 奈良先端科学技術大学院大学, 1999.

[6] 田久保浩平, 橋本光憲. 英文ビジネスレター文例大辞典. 日本経済新聞社, 1995.

[7] 藤尾正和, 松本裕治. 語の共起確率に基づく係り受け解析とその評価. 情報処理学会論文誌 Vol.40 No.20, pp. 4201–4211, 1999.

[8] 北村美穂子, 松本裕治. 対訳コーパスを利用した翻訳規則の自動獲得. 情報処理学会論文誌 Vol.37 No.6, pp. 1030–1040, 1996.

[9] 北村美穂子, 松本裕治. 対訳コーパスを利用した対訳表現の自動抽出. 情報処理学会論文誌 Vol.38 No.4, pp. 727–736, 1997.