統計的係り受け結果を用いた対訳表現抽出

山本 薫[†] 松本裕治[†]

近年，対訳コーパスが普及し，文レベルの対応や語レベルの対応をとる手法が活発に提案されてきた．一方，中間に位置する局所的な文節係り受けレベルの対応についての研究は，あまり行われていない．機械翻訳では，主語，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して翻訳規則を記述する場合が多い．このため，文節係り受けレベルでの対応を自動的に獲得する技術は，重要な課題である．本稿では，文対応された対訳コーパスから統計的係り受け解析結果を用いて文節間の依存構造が反映された対訳表現を抽出する手法を提案する．従来の研究では，言語の品詞，語順，語源（cognate）を手がかりとして，語や連語レベルの平面的な対訳表現を抽出にとどまることが多かった．しかし，本稿で提案する手法は，近年，精度向上が目覚ましい統計的係り受け解析技術を利用することにより，文節（部分依存木）レベルの構造的な対訳表現を抽出できる．また，日英対応のように，文の基本構造が異なる 2 言語間でも文節の依存関係は保たれるため，文節の依存関係が，対訳表現抽出において，有用な手がかりであることを示す．

Translation Pattern Acquisition Using Dependency Structures

Kaoru Yamamoto[†] and Yuji Matsumoto[†]

This paper proposes a method to extract phrase–level translation patterns from parallel corpora using dependency structures. Previous approaches use part of speech, word ordering, and/or cognate information as linguistic clues. Inevitably, the extracted translation patterns are of flat form. However, our proposed approach uses statistical dependency analyzers to induce probable dependency relationship between phrases, thereby extracting translation patterns of structured form. Moreover, we argue that dependency relationship serves as effective linguistic clues in the task of translation pattern acquisition, since such dependency tends to be preserved even if the two languages have different syntactic structure.

1. はじめに

1990 年代に入り，電子化された対訳コーパスが大量に入手可能になり，統計的手法を機械翻訳の諸問題に応用する手法が提案された[2]．具体的な題材として，文アラインメント[3] や単語対応の手法[7] が取り上げられ，実用的なレベルにまで到達している技術もある．

これに比べ，中間に位置する，局所的な文節係り受けレベルの対応をとる有効な手法はあまり提案されていない．主語，目的語に応じた動詞の訳し分けや慣用表現などは単語対応のみで翻訳できない．このような背景から，ひとまとまりに翻訳される最小単位での対訳表現を自動的に獲得することは，重要な課題といえる．

今まで，2 言語間の単語以上の対応をとる試みはされている．これらの手法は，主に，形態素解析や NP-recognizer で得られる，それぞれの言語の品詞，語順，語源（cognate）を手がかりとしていた[10),13)]．文の構造が考慮されていないため，必然的に，抽出された対訳表現は平面的な名詞句対応にとどまる．

筆者らは，文節（部分依存木）レベルの構造的な対訳表現抽出を目標としており，従来の手がかりでは不十分と考える．理由は次のとおりである．日本語と英語のように言語族が異なる場合，語源を共有しない☆．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，日英対応のように基本的な文の構造が異なる言語間から，語順を手がかりとして，名詞句以上の文構造を反映した対訳表現を抽出するのは困難である．

近年，統計的手法を用いた構文解析技術が向上しつつある[5),6),15),17)]．従来の規則主導型の構文解析と比べ，統計的手法は，人手による煩雑な文法規則保守の

† 奈良先端科学技術科学大学院大学
Nara Institute of Science and Technology

☆ カタカナ表記された外来語を語源と見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．

手間が省け，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，統計的手法で一文の完全な構文解析は困難であり，今後も統計的手法の枠組みで大幅な精度向上は難しいと考えられている[4]．しかし，筆者らは，ある程度解析精度が劣っていても，有効な部分解析結果は多く含まれており，これらを活用することにより，有用な対訳表現が抽出できると考える．

そこで，本研究では，1) 統計的な係り受け関係を用いて文節間の依存構造が反映された対訳表現を抽出する方法を提案し，2) 文節の依存関係が，従来の品詞，語順，語源（cognate）に代わる手がかりとしてどの程度有用かを考察する．文節の依存関係を採用する理由は，2 言語間対応の多くの場合において，表層の語順は必ずしも一致しないが，文節の依存関係は保たれるからである．具体的な手法は，以下のとおりである．まず，文の係り受け関係を利用して，文の部分依存木を候補パターンとして作成する．次に，それらの共起頻度をもとに類似度を計算し，類似度が高い候補パターン対から順に抽出する．

以下，2 章で，統計的な係り受け解析の概要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，対訳表現抽出方法について述べ，4 章で，実験結果を考察する．5 章で，関連研究との違いを説明し，6 章でまとめる．

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[9] に基づく解析で，文節とその関係に着目する枠組みである．依存木は，木構造で表現され，係り元文節から受け先文節へ矢印で示される．

統計的係り受け解析は，まず，文を適切な単位（ここでは文節単位）に区切り，次に，単位の間の係り受け関係を単位の属性（素性）から統計的に推定する処理になる．このとき，係り受け関係は非交差，非循環であり，ルート文節☆以外は必ず 1 つの係り先を持つという制約を持たせている．

文節区切りは，品詞および語の列を正規表現で規定した．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，(1) BaseNP（再帰的に NP を含まない NP），(2) 前置詞と BaseNP が連結しているもの，(3) 助動詞や完了形などを主動詞とまとめた動詞的表現，(4) 時間や日付表現で区切ったものを文節相当と見なした．

係り受け関係の学習には，語の共起確率に基づく統計的係り受けモデルを用いた[17]．このモデルは，文節の主辞，関係名，品詞，句読点の有無といった文節素性や 2 文節間の距離と方向，句読点数といった文節間素性をもとに統計をとる．

日本語の係り受け解析器 jdep は，EDR コーパスを用いて係り受け関係を推定しており，精度は約 86%と報告されている[17]．英語の係り受け解析器 edep は，Penn Treebank を用いて係り受け関係を推定している．ただし，実験システムで正式な精度は報告されていない．

2.2 候補パターン生成

候補パターンは，統計的に解析された文の係り受け関係を利用して作成する．係り受け解析においては，ある 1 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であるという問題がある．統計的係り受け解析の文献では，完全な係り受け解析精度の向上を求めるという立場で議論される．本研究の目的からすると，文節の係り先が曖昧な場合はある確信度以上の係り先を冗長に出力する，つまり，解析の適合率を犠牲にして再現率を向上するほうが好ましい．

そこで，候補パターンを生成する際の依存構造の有用性について調べるため，次の 3 つのモデルを用意した．

- 「統計最良モデル」統計的に最良と思われる係り受け関係のみを考慮．
- 「統計曖昧モデル」統計的に曖昧と推測される係り受け関係も含めて考慮．
- 「文節 n-gram モデル」ある文節が直前の文節に係っていると仮定．

係り受け関係を利用した「統計最良モデル」と「統計曖昧モデル」より生成された候補パターンは，文の部分依存木に相当する．一方，文節を隣接文節に限って連結した「文節 n-gram モデル」より生成した候補パターンは，文の部分文字列に相当する．これは，比較のために作成したモデルである．

「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．どの程度の冗長性を持たせるかは 4.1 節で記述する．コーパス全体の候補パターンの集合に，もっともらしい係り受け関係が多く出現することになり，その類似度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，もっともらしい係り受け関係を保った対訳表現が抽出されることを

☆ 日本語では文末の文節，英語では主節の動詞を含む文節を指す．

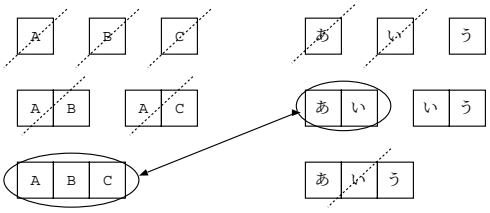


図 1 対訳アラインメント
Fig. 1 Alignment heuristic.

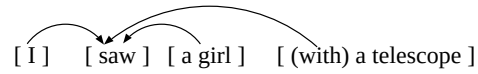
期待している．

以下，候補パターン生成の具体的な手順について述べる．候補パターンの長さとは，翻訳規則の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 $n-1$ $(n>1)$ の候補パターンと長さ 1 の候補パターンの組合せから生成される．もと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．たとえば，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 {A, B, C, AB, AC} 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 { あ, い, あいう } 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（以下のステップ e），候補パターン生成と同時に親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，以下のとおりである．

(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) 形態素解析し，品詞タグを付与する．
  (b) 文節区切り規則をもとに，文節まとめあげをする．
  (c) 係り受け解析をし，文節の依存関係を推定する．
  (d) 候補パターンを生成する．
    (i) 長さ 1 の候補パターンを作成する．
    (ii) 長さ n の候補パターンを係り受け関係を有する長さ $(n-1)$ の候補パターンと長さ 1 の候補パターンの組合せで作成する．係り元パターンはそのままだが，受け先パターンの機能語は削除する．
  (e) 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を，出現回数が多い順に，もし同じなら候補パターンの長さが長い順に，出力する．

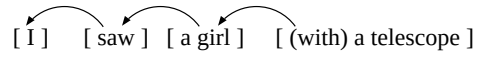


size 1) {I, saw, girl, telescope}
size 2) {I_saw, girl_saw, with-telescope_saw}
size 3) {I_girl_saw<T>,
I_with-telescope_saw<T>,
girl_with-telescope_saw<T>}

図 2 統計最良モデル
Fig. 2 Best-one model.

size 1) {I, saw, girl, telescope}
size 2) {I_saw, girl_saw, with-telescope_saw,
with-telescope_girl}
size 3) {I_girl_saw<T>,
I_with-telescope_saw<T>,
girl_with-telescope_saw<T>,
with-telescope_girl_saw<L>}

図 3 統計曖昧モデル
Fig. 3 Ambiguous model.



size 1) {I, saw, girl, telescope}
size 2) {saw_I, girl_saw, with-telescope_girl }
size 3) {girl_saw_I<L>,
with-telescope_girl_saw<L>}

図 4 文節 n–gram モデル
Fig. 4 Adjacent model.

図 2，**図 3**，**図 4** に，それぞれ「統計最良モデル」，「統計曖昧モデル」，「文節 n–gram モデル」から生成された候補パターン例を示す．経験的に，受け側文節の機能語は，翻訳規則に貢献しない場合が多いため，受け側文節の機能語を削除しておく．また，長さ 3 以上の候補パターンでは，異なる部分依存木が生成され

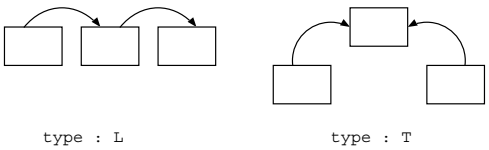


図 5 長さ 3 の候補パターン
Fig. 5 Candidate patterns of length 3.

るので，ラベルを付与する（長さ 3 の場合，2 種類の依存木が可能で，それぞれ L，T とラベル付けしている．**図 5** を参照）．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おり，これらの確率を候補パターン生成の際に考慮することが可能である．しかし，今回の実験では，簡単化のため，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．

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，次章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 類似度の計算方法

2 言語間の候補パターンの対応関係の強さを示す尺度として候補パターン間の類似度を定義する．これは，候補パターンが各言語で独立に出現する回数と，対訳文に同時に出現する回数で求められる．

筆者らは，北村らで提案された，重み付き Dice 係数を類似度として採用した[20]．

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

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．

先行研究では相互情報量などの類似度が広く利用されている☆が，重み付き Dice 係数を採用した理由は，以下のとおりである．第 1 に，相互情報量では，出現回数が低い場合に正しく評価されないという問題点と翻訳の方向の偏りを扱えない問題点が報告されている[16]．第 2 に，標準の Dice 係数は，出現回数の大小にかかわらず，独立出現回数と同時出現回数の相対比で類似度が決まる．北村らは，Dice 係数を同時出現回数で重みを付けることにより，候補パターンの相関関係と出現回数の両方を考慮した類似度を提案した[20]．今回の実験は，出現回数が少ない対訳表現も抽出したいため，出現回数も反映した重み付き Dice 係数を類似度として利用した．

☆ Matsumoto らが各類似度の特徴を比較している[12]．

3.2 対訳抽出アルゴリズム

本手法の対訳抽出アルゴリズムは，基本的に北村らの枠組みをベースにしている[20]．つまり，上記の類似度を 2 言語の候補パターン集合の全組合せに対して計算し，対応関係の強いペアから順に抽出する．

アルゴリズムは，出現回数を閾値とし，出現回数の多いものから順に類似度を計算する．出現回数を段階的に下げることにより，対訳表現候補対象を拡大する．アルゴリズム途中で，一度対訳表現として確定された p_e と p_j の同時出現場所を次の繰返しから数え上げない．このため，アルゴリズムが進むにつれ，p_e と p_j の出現回数が低下し，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，次の繰返しから数え上げの対象外とする．

対訳抽出アルゴリズムの手順は，以下のとおりである．

(1) 入力：日英の候補パターンファイル，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 を決める．
(2) 閾値 th より多く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持つ p_e' を探す．
  (b) p_e において最大の類似度を持つ p_j' を探す．
  (c) p_j と p_j'，p_e と p_e' が同じで，かつ，
  $$sim(\langle p_j, p_e \rangle) > \alpha \log_2(th)$$
  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回数から p_j と p_e の同時出現回数を引く．
(3) 対訳表現抽出数が規定値になれば，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，β は 1 以上の値であれば

表 1 前処理ツール
Table 1 Pre-processing tool.

前処理	ツール	備考
形態素解析 (英)	ChaSen2.0	精度 96%
形態素解析 (日)	ChaSen2.0	精度 98%
文節まとめあげ (英)	MatCha-1.0	rule-based
文節まとめあげ (日)	Unit	rule-based
係り受け解析 (英)	edep	—
係り受け解析 (日)	jdep	精度 85–87%

よい.

4. 実験結果

4.1 設定

実験は，あらかじめ文レベルで対応付けられた対訳コーパス，日経ビジネスレター例文集[18]（13,000 文）を使った．利用した前処理ツールと備考を**表 1** に整理する．

パラメータは次のように設定した.「統計最良モデル」においては，統計的に尤もらしい依存木を利用している．一方,「統計曖昧モデル」では，各文節の複数の係り受け関係をすべて考慮する．複数の係り先は次のように求める．各文節の係り受け関係を確率順に並べ，

$$\frac{\texttt{prob(kth - ranked dependency)}}{\texttt{prob((k + 1)th ranked dependency)}} \geq \texttt{0.5}$$

を満たす係り受け関係を出力する．いずれのモデルでも，候補パターン生成の際の候補パターンは長さ 1, 2, 3 までに限定した．

対訳抽出アルゴリズムの閾値は，初期値は 100 とし，最終値を 2 に設定した．α は 1 にし，一番厳しい類似度上限条件に設定した．閾値は，100 から 10 までは半分ずつ 10 以下は 1 ずつ減るという推移をとらせた．すべてのパラメータ設定は，実験を通じて決定した．

4.2 結果

表 2, **表 3**, **表 4** に，それぞれ「統計最良モデル」,「統計曖昧モデル」,「文節 n–gram モデル」の実験結果を示す.「抽出数」は閾値の段階別に対訳表現が抽出された数を,「正解」はそのうちの正解の数を示す．評価は人手で行い，正解の基準は，対訳表現をコーパスに出現した形に復元したあと，そのまま辞書に登録できるかどうかで判断する．半正解の基準は，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かどうかで判断する.「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 より上の正解数と抽出数の比率である．

各モデルの特徴をさらに詳しく調査するために，閾値を > 2 から ≥ 2 まで拡大してみた．その結果は，表中では，*で示している．

表 5 に「統計最良モデル」で抽出された対訳表現の正解例を示す．+ は文節境界を示し，_ は形態素境界

表 2 精度の推移：「統計最良モデル」
Table 2 Precision: best-one model.

閾値	正解	半正解	抽出数	精度	累積精度
25	6	0	6	100.00	100.00
12	7	0	7	100.00	95.00
10	6	1	7	85.71	95.83
9	4	0	4	100.00	92.30
8	13	0	13	100.00	97.29
7	10	2	13	76.92	92.00
6	19	1	20	95.00	92.85
5	29	0	29	100.00	94.94
4	67	4	72	93.05	94.15
3	150	4	164	91.46	92.83
2	414	4	461	89.80	91.08
(*2	264	66	474	55.69	77.93)
total	**725**	**20**	**796**	—	**91.08**
(*total	989	82	1269	—	77.93)

表 3 精度の推移：「統計曖昧モデル」
Table 3 Precision: ambiguous model.

閾値	正解	半正解	抽出数	精度	累積精度
25	6	0	6	100.00	100.00
12	7	0	7	100.00	100.00
10	6	1	7	85.71	95.00
9	4	0	4	100.00	95.83
8	13	0	13	100.00	97.29
7	11	2	13	84.61	94.00
6	18	1	19	94.73	94.20
5	29	0	29	100.00	95.91
4	68	5	73	93.15	94.73
3	118	3	126	93.65	94.27
2	432	4	468	91.50	93.07
(*2	256	132	759	33.72	63.51)
total	**712**	**16**	**765**	—	**93.07**
(*total	968	148	1524	—	63.51)

表 4 精度の推移：「文節 n–gram モデル」
Table 4 Precision: adjacent model.

閾値	正解	半正解	抽出数	精度	累積精度
25	6	0	6	100.00	100.00
12	7	0	7	100.00	100.00
10	6	1	7	85.71	95.00
9	4	0	4	100.00	95.83
8	13	0	13	100.00	97.29
7	10	2	13	84.61	92.00
6	18	1	19	94.73	92.75
5	29	0	29	100.00	94.89
4	68	4	73	93.15	94.15
3	114	3	126	93.65	92.59
2	419	4	484	86.57	88.86
(*2	280	76	496	56.45	76.27)
total	**694**	**15**	**781**	—	**88.86**
(*total	974	91	1277	—	76.27)

表 5 「統計最良モデル」で抽出された正解例
Table 5 Best-one model: random sample of correct translation pairs.

英語	日本語	類似度
thank+you	ありがとう	4.7037
consultations+include	協議_に_は+含める	2.3219
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
not+hesitate+to_contact	遠慮なく+ご連絡	1.6000
I+must_object	反対_いたす	1.1887
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
applications+include	用途_に_は+ある	1.0000
upcoming_board+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 6 「統計最良モデル」で抽出された半正解例
Table 6 Best-one model: random sample of near-correct translation pairs.

英語	日本語	類似度
(have_been_pleased)+to_serve+as_their_main_banker	主力_銀行_と+なる	1.0000
[be_held]+at_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
[passed]+on_past_tuesday	火曜日_に+亡くなら_れる	1.0000

を示す．表の対訳表現はコーパスに出現した形に復元したことに注意されたい．

表 6 に「統計最良モデル」で抽出された対訳ペアの半正解例を示す．正解基準に満たすために，追加されるべき文節を [ ] で，削除されるべき文節を () で囲む．

4.3 考　　察

はじめに，対訳表現抽出において，係り受け関係の有用性を検証するために,「統計最良モデル」と「文節 n–gram モデル」を比較する．表 2 と表 4 を見ると，「統計最良モデル」の方が「文節 n–gram モデル」より高い精度が出ている．結果内容をさらに調べると，閾値が高い場合は，ほぼ同じ対訳表現が抽出されている．これは，学習した係り受け解析器が，2 文節間の距離が近いほど係りやすくなることに起因する．このため，多くの係り受け関係が隣接文節の距離 3（文節 tri–gram）以内の範囲で収まっていると考えられる．実験データの「統計最良モデル」と「文節 n–gram モデル」の重なり度合いは，英語の場合，候補パターン総数 14,705 個のうち 9,438 個（63.55%）で，日本語の場合，候補パターン総数 11,566 個のうち 6,625 個（57.27%）であり，かなり高い割合を占めている．

一方，閾値が 3 まで下がると,「文節 n–gram モデル」では生成されない対訳表現が「統計最良モデル」では抽出しはじめた．たとえば，抽出された対訳表現 ⟨not to hesitate to contact, 遠慮なくご連絡⟩ の場合，コーパスには「遠慮なくご連絡」,「遠慮なく私にご連絡」,「遠慮なく折り返し当方にご連絡」のように，部分的に変形して出現している．「文節 n–gram モデル」では，これらすべて違う候補パターンとして集計される反面,「統計最良モデル」では，同じ [遠慮なく+ご連絡] という係り受け関係で集計される．このため，出現回数の底上げ効果が得られ，結果として，対訳表現として抽出される．このような部分的に変形したものは，比較的出現回数が低いところに集中している．以上の観察から，出現回数が低い対訳表現の抽出もできるという点において，係り受け関係のほうが表層的な文節の順番より有用な手がかりであることがいえる．

次に，対訳表現抽出において，係り受け関係の曖昧性の影響度を検証するために,「統計最良モデル」と「統計曖昧モデル」を比較する．表 2 と表 3 を見ると，閾値 > 2 の場合,「統計曖昧モデル」の方が「統計最良モデル」より高い精度が出ている．これは，係

表 7 被覆率
Table 7 Coverage rate.

モデル	英語	日本語	平均
統計最良	19.12	19.59	19.13
統計曖昧	19.57	19.95	19.76
文節 n–gram	18.69	19.20	18.40

り受け解析器の精度がまだ十分でないことを示唆している．現状では，曖昧な係り受け関係は複数許し，候補パターンを生成するほうが結果がいい．しかし，係り受け解析の精度が向上すれば，必然的に正しい係り受け関係で生成された候補パターンが多くできることになり,「統計最良モデル」の結果が改善されると予測する．

しかし，閾値を 2 にすると,「統計最良モデル」と「統計曖昧モデル」の正解数はさほど変化しないが，抽出数が約 2 倍となる．さらに，英語の候補パターン数で比較すると,「統計最良モデル」は 14,705 個に対し,「統計曖昧モデル」は 34,234 個も生成された．曖昧性を許すことで 19,529 個の候補パターンが新たに生成され，そのうち 14,032（72%）が 2 回しか出現していない．全体の傾向として，閾値が高いときは，係り受けの曖昧性を吸収し類似度の底上げ効果が得られているが，閾値が低いときは，候補パターンが大量に増加しノイズの原因になったと考えられる．

表 7 に，日英それぞれのコーパスにおける，各モデルで得られた正解パターンの被覆率を示す．今回の実験では，取り出された対訳表現の数には大差なく，いずれのモデルも約 20%で，被覆率の差はあまり見られなかった．わずかではあるが,「統計曖昧モデル」の被覆率が他のモデルより高いという結果が得られた．

まだ，完全な係り受け解析結果があると仮定した場合，どのくらいの精度と被覆率を見込めるかという疑問が残る．今回の実験では，利用した対訳コーパス 13,000 文に対して，正解係り受けデータを準備するのは現実的ではないと判断しため，この点に関する評価ができない．しかし，完全な係り受け解析を得られなくても，再現率重視の冗長な係り受け解析結果から，比較的精度の高い対訳表現が抽出ができたことにより，本手法の有効性を示せたと考える．

今度は，抽出された対訳表現に関して考察する．表 5 にみるように，本手法においても，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．また，日本語の主語省略 ⟨I must object, 反対いたす⟩ や文の構造が違うため逐語訳が困難なもの ⟨be writing to let know, 書状をもってお知らせする⟩ も，抽出できた．

しかし，抽出に失敗した例もいくつかある．閾値が低くなると，⟨my deepest sympathy on the death, 太田首相のご逝去⟩ のように部分的に対応は正しいが，対訳表現としては間違ったものがあった．この現象は，対訳抽出アルゴリズムが貪欲的であるため，処理過程で間違った対訳表現が最初に抽出されると，その誤りは伝搬されるという性質に起因するものと考えられる．閾値が低くなると，同じ対訳文から生成されている候補パターンどうしで類似度が計算される可能性が高い．類似度を計算した順番によって半正解の対訳表現が抽出されることがある．

上記の例では，同じ対訳文から候補パターン “my deepest sympathy on the death of Prime Minister Ohta”, “on the death of Prime Minister Ohta”,「太田首相のご逝去」が 2 個ずつ生成されていた☆．対訳抽出アルゴリズムは，同じ類似度の場合，長い候補パターンを選ぶので ⟨my deepest sympathy on the death, 太田首相のご逝去⟩ が抽出され，対訳アラインメントにより，“on the death of Prime Minister Ohta” と「太田首相のご逝去」は，アルゴリズムの次の繰返しで数え上げの対象から外れる．このため，⟨on the death of Prime Minister Ohta, 太田首相のご逝去⟩ は抽出されなくなる．抽出アルゴリズムを EM 的な繰返し法にすれば，このような誤り伝搬はある程度回避できると思われる．今回の実験では，計算時間とのトレードオフを考え，EM 的な繰返し法は採用しなかった．

次に，データ表記の揺れのために候補パターンとして生成されなかったものがあった．たとえば，英語コーパスに “put a one-month limit” は 2 回出現したが，日本語コーパスでは “1 カ月の期限をつける” と “1 カ月の期限を付ける” に訳されており，対応する日本語の候補パターンは生成されなかった．このような現象は，英語では，stemming 処理に，日本語では，漢字–ひらがな–カタカナ使いに見られた．形態素解析レベルでこれらの異表記語の同定ができるようになれば，この問題は解決できる．

最後に，離散対訳表現（係り受け関係では長さ 3 以内だが，表層文では文節間距離が離れているパターン）が抽出できない場合もあった．係り受け解析の精

☆ 対訳文は，⟨On behalf of your nation, please accept my deepest sympathy on the death of Prime Minister Ohta., 太田首相のご逝去に際し，日本の皆様（のため）に心よりお悔やみ申し上げます.⟩ と ⟨On behalf of your nation, please accept my deepest sympathy on the death of Prime Minister Ohta. , 貴国の皆様に，太田首相のご逝去に対する，私の深い哀悼の意をお受け下さい.⟩

度が比較的良い,「主格–述部」型の対訳表現（たとえば〈consultations include, 協議は含める〉「協議は」と「含める」の間に「何を」のような挿入句が存在する場合）は抽出できた．しかし，係り受け解析に曖昧性が高くなる「目的格–述部」型の対訳表現（たとえば〈be held on DATE, DATE に開催する〉“be held” と “on DATE” の間に “at PLACE” のような挿入句が存在する場合）型の対訳表現は抽出できなかった．

一般的に離散対訳表現の抽出は難しい問題であり，今回の実験では完全には解決できなかった．係り受け解析間違いが主な失敗した理由としてあげられる．一般的に，同じ表層文字列で構成される離散対訳表現の出現回数は少なく，部分的に変形して出現することが多い．また，実世界のデータでは，副詞句など挿入箇所が比較的自由な節は，多く出現するものと考えられる．このような部分変形を確実に吸収するには，今後，より精度の高い係り受け解析がますます必要となる．

5. 関 連 研 究

先行研究では，語順を用いた連語レベルの対訳表現抽出も発表されている．文献 16) では，コロケーション抽出方法を提案している．ある一定のウィンドウサイズ内に共起する語の分布をもとに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 1 語ずつ延長しながら抽出する．文献 20) 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，対訳表現を抽出している．文献 8) 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，対応付けを行う．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する．

筆者らの手法は，文献 16) の方法と違い，双方向に候補パターンを生成する．また，文献 8) の方法と違い，あらかじめの単語レベルでの対応を必要とせず，1 段階で離散的な対訳表現が抽出できる．さらに，文献 20) の方法と違い，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．文献 11)，14)，19) では，文を解析して，構造照合を行うことによって，翻訳規則を抽出する．これらの手法は，規則主導型で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．また，文献 1) は，対訳コーパスから確率的ヘッドトランスデューサを用いて，同期依存木を自動学習する手法を提案している．語の類似度として ϕ measure を利用しており，対訳文から部分文字列で構成される依存木を抽出している．

筆者らの手法は，文献 11)，14)，19) の各方法と違い，統計的係り受け解析器を利用しており，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，手法の頑健さを向上できると思われる．さらに，文献 1) の方法と違い，筆者らの手法は，文の依存構造のみに着目しており，対訳コーパス全体から出現回数を集計することにより依存木どうしの類似性を測っている．

6. お わ り に

本稿では，文対応された対訳コーパスから，統計的係り受け解析結果を用いて，文節間の依存構造が反映された対訳表現を抽出する手法を提案した．従来の研究では，言語の品詞，語順，語源（cognate）を手がかりとして，語や連語レベルの平面的な対訳表現の抽出にとどまることが多かった．しかし，本稿で提案した手法は，近年，精度向上が目覚ましい統計的係り受け解析技術を利用することにより，文節（部分依存木）レベルの構造的な対訳表現を抽出できた．また，日英対応のように，文の基本構造が異なる 2 言語間でも文の依存関係は保たれるため，文節の依存関係が，対訳表現抽出において，有用な手がかりであることを示した．

実験を通じて，データ表記の揺れのほかに，出現回数が低い対訳表現抽出の精度が比較的悪い，離散的な対訳表現が抽出されない場合もある，という課題も残った．しかし，より精度の高い係り受け解析が実現されれば，これらの問題も解決されるものと思われる．

提案した手法により，2 言語の単文のあらゆる文型の依存構造の対応付けを扱えるようになった．今後は，係り受け解析が困難と思われる連接接続表現などに現れる複文構造を 2 言語間で対応させる手法を提案したいと考えている．

謝辞 日経ビジネスレター例文集の研究利用許諾をいただいた日本経済新聞社に感謝の意を表する．本研究に関して有益な助言をいただいた沖電気工業の北村美穂子氏と，ツールを提供していただいた日立中央研究所の藤尾正和氏に感謝する．

参 考 文 献

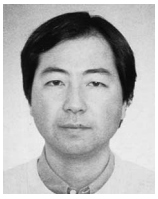
1) Alshawi, H., Bangalore, S. and Douglas, S.: Learning Dependency Translation Models as Collections of Finite-State Head Transduc-

ers, *Computational Linguistics*, Vol.26, No.1, pp.45–60 (2000).
2) Brown, P., Pietra, J.C., Pietra, S., Jelinek, V.F., Lafferty, J.D., Mercer, R.L. and Roossin, P.S.: A Statistical Approach to Machine Translation, *Computational Linguistics*, Vol.16, pp.79–85 (1990).
3) Brown, P., Lai, J. and Mercer, R.: Aligning Sentences in Parallel Corpora, *ACL-29: 29th Annual Meeting of the Association for Computational Linguistics*, pp.169–176 (1991).
4) Charniak, E.: Statistical Parsing with a Context-free Grammar and Word Statistics, *AAAI-97: Proc. 14th National Conference on Artificial Intelligence*, pp.598–603 (1997).
5) Charniak, E.: A Maximum-Entropy-Inspired Parser, *NAACL-2000: 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pp.132–139 (2000).
6) Collins, M.: Three Generative, Lexicalised Models for Statistical Parsing, *ACL-97: 35th Annual Meeting of the Association for Computational Linguistics*, pp.16–23 (1997).
7) Dagan, I., Church, K. and Gale, W.: Robust Bilingual Word Alignment for Machine Aided Translation, *Proc. Workshop on Very Large Corpora*, pp.1–8 (1992).
8) Haruno, M., Ikehara, S. and Yamazaki, T.: Learning Bilingual Collocations by Word-Level Sorting, *COLING-96: 16th International Conference on Computational Linguistics*, pp.525–530 (1996).
9) Hudson, R.: *Word Grammar*, Blackwell (1984).
10) Kupiec, J.: An Algorithm for Finding Noun Phrase Correspondences in Bilingual Corpora, *ACL-93: 31st Annual Meeting of the Association for Computational Linguistics*, pp.23–30 (1993).
11) Matsumoto, Y., Ishimoto, H. and Utsuro, T.: Structural Matching of Parallel Texts, *ACL-93: 31st Annual Meeting of the Association for Computational Linguistics*, pp.23–30 (1993).
12) Matsumoto, Y. and Utsuro, T.: Lexical Knowledge Acquisition, *Handbook of Natural Language Processing, Part II*, Dale, R., Moisl, H. and Somers, H.(Eds.), pp.563–610, Marcel Dekker (2000).
13) Melamed, I.: Automatic Evaluation and Uniform Filter Cascades for Inducing N-Best Translation Lexicons, *Proc. 3rd Workshop on Very Large Corpora*, pp.184–198 (1995).
14) Meyers, A., Yangarber, R. and Grishman, R.: Alignment of Shared Forests for Bilingual Corpora, *COLING-96: The 16th International Conference on Computational Linguistics*, Vol.1, pp.460–465 (1996).
15) Ratnaparkhi, A.: A Linear Observed Time Statistical Parser Based on Maximum Entropy Models, *Proc. 2nd Conf. on Emperical Methods in Natural Language Processing*, pp.1–10 (1997).
16) Smadja, F., McKeown, K. and Hatzuvassiloglou, V.: Translating Collocations for Billingual Lexicons: A Statistical Approach, *Computational Linguistics*, Vol.22, No.1, pp.1–38 (1996).
17) 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.12, pp.4201–4212 (1999).
18) 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).
19) 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.78–88 (1996).
20) 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).

(平成 12 年 11 月 6 日受付)
(平成 13 年 6 月 19 日採録)

山本 薫

1973 年生．1995 年ウェールズ大学カーディフカレッジにて BSc. Computer Science を取得．1996 年ロンドン大学インペリアルカレッジにて MSc. Foundations of Advanced Information Technology を取得．同年（財）九州システム情報技術研究所研究助手．1999 年奈良先端科学技術大学院大学博士後期課程入学．自然言語処理，特に機械翻訳に興味を持つ．



松本 裕治（正会員）

1955 年生．1979 年京都大学大学院工学研究科修士課程修了．同年電子技術総合研究所入所．1984 年英国インペリアルカレッジ客員研究員．1985 年（財）新世代コンピュータ技術開発機構に出向．京都大学助教授を経て，1993 年奈良先端科学技術大学院大学教授，現在に至る．工学博士．専門は自然言語処理．