未入力文節との構文的関係を考慮する漸進的な係り受け解析

村田 匡輝[†]　　大野 誠寛[‡]　　松原 茂樹[††]

[†]豊田工業高等専門学校 情報工学科　　[‡]名古屋大学 情報基盤センター

[††]名古屋大学 大学院情報科学研究科

murata@toyota-ct.ac.jp

1　はじめに

同時通訳や字幕生成，対話システムなどの音声言語アプリケーションでは，入力と同時的に処理を行うことが求められる．このようなアプリケーションにおいて構文的情報を利用するためには，音声入力途中の段階で随時，構文構造を生成できる必要がある．

本論文では，未入力文節との係り受け関係を明示する漸進的な係り受け解析手法を提案する．提案手法は，文節が入力されるたびに解析を実行し，係り先が入力されていない文節に対しては，係り先が未入力であることを同定する．また，係り先が未入力である文節が複数あるときは，それらの係り先が同一か否かを同定する．

本研究では，著者らがこれまでに開発している漸進的係り受け解析手法 [2] を拡張することにより提案手法を実現する．日本語講演データを用いた解析実験の結果，係り先が正しく同定できているかの判定において正解率 71.22% を達成し，未入力文節との係り受け関係が明示された係り受け構造を精度よく生成できることを確認した．

2　出力する係り受け構造

文の入力に追従して処理するアプリケーションでは，音声の入力途中でも随時，構文的情報を獲得できることが望ましい．これらのアプリケーションでは，例えば日本語音声の場合には，

- どの文節がどの文節に係るか

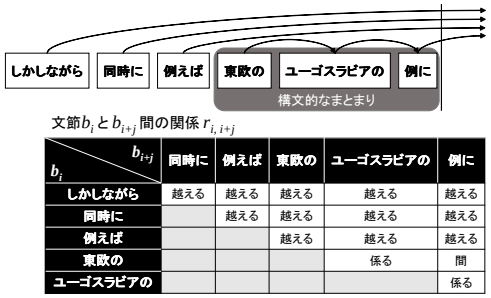
といった文節間の構文的関係に加え，

- ある文節列の係り受けが閉じているか

といった構文的なまとまりに関する情報も有用である．一例として，字幕生成における読みやすい改行位置の決定において，構文的まとまりに関する情報が重要な手がかりとなることが示されている [1]．

著者らはこれまでに漸進的な係り受け解析手法 (以下，**従来手法**) を提案している [2]．この手法は，文節が入力されるごとに解析を実行し，係り先が入力されていない文節に対して，その係り先は未入力であることを明示した係り受け構造を出力する．未入力文節との構文的関係から既に入力されている文節列内の構文的なまとまりを捉えることができる．

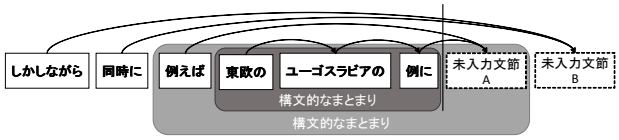
図 1 に文，



文節b_iとb_{i+j}間の関係$r_{i,\,i+j}$

b_i \ b_{i+j}	同時に	例えば	東欧の	ユーゴスラビアの	例に
しかしながら	越える	越える	越える	越える	越える
同時に		越える	越える	越える	越える
例えば			越える	越える	越える
東欧の				係る	間
ユーゴスラビアの					係る

図 1: 従来手法が出力する係り受け構造



文節b_iとb_{i+j}間の関係$r_{i,\,i+j}$

b_i \ b_{i+j}	同時に	例えば	東欧の	ユーゴスラビアの	例に
しかしながら	同一	異なる	異なる	異なる	異なる
同時に		異なる	異なる	異なる	異なる
例えば			異なる	異なる	同一
東欧の				係る	間
ユーゴスラビアの					係る

図 2: 提案手法が出力する係り受け構造

- しかしながら同時に例えば東欧のユーゴスラビアの例に見られますように地域的な紛争というものはむしろ頻発してきたということが言えると思います

の文節「例に」まで入力された段階で従来手法が出力する係り受け構造を示す．従来手法では，着目している 2 文節 (前文節と後文節) 間の係り受け関係を，

- 前文節が後文節に「**係る**」関係
- 前文節と後文節の間に存在する文節について前文節がその文節を越えて後文節に係る (「**越える**」関係)
- 後文節より文末側に存在する文節について前文節がその文節との間にある後文節に係る (「**間**」の関係)

という 3 種類のいずれかで表現される，前文節とそれ以降の文節との関係の集合として定義している．

図 1 に示した係り受け構造では，文節「しかしながら」,「同時に」,「例えば」の係り先が未入力であることが明示されており，それにより，既入力文節内の「東欧のユーゴスラビアの例に」が構文的なまとまりを構成することが分かる．

一方，係り先が未入力である文節が複数存在したとき，それぞれの文節は別々の未入力文節に係ることもあれば，同一であることもある．各文節の係り先が同一か否かを同定できれば，構文的なまとまりをより詳細に捉えることが可能となる．図 2 に，文節「しかしながら」と「同時に」の係り先が同一 (未入力文節 A) で，「例えば」と「例に」の係り先が同一 (未入力文節 B) であり，両係り先は異なることを明示する係り受け構造を示す．このような係り受け構造を同定することができれば,「例えば東欧のユーゴスラビアの例に」と未入力文節 A からなる文節列が構文的なまとまりを構成し,「東欧のユーゴスラビアの例に」というまとまりはその中に含まれることが分かる．このように，未入力文節に係る文節の係り先が他の文節の係り先と同一か否かということが分かれば，従来手法では捉えられなかった構文的なまとまりを捉えることが可能となる．

本手法は従来手法を拡張することで実現する．「ある文節が未入力文節に係る」という関係は，ある文節と全ての既入力文節とが「越える」関係になることを意味する．本研究では,「越える」の関係となる 2 文節の係り先が同一となるか否かを表現するために，従来手法における「越える」の関係を，

- 越える・同一の文節に係る（「**同一**」の関係）
- 越える・異なる文節に係る（「**異なる**」関係）

の 2 種類の関係に細分化する．これらの関係を用いることで，図 2 のように，ある文節の係り先が未入力であり，かつ，他の文節の係り先と同一か否かを示す係り受け構造を表現することができる．

以下では，本研究において出力する係り受け構造を定式化する．文節列 $b_1 \cdots b_n$ からなる文 S を解析する場合に，文節 $b_x(1 \leq x \leq n)$ が入力された時点で本手法が出力する係り受け構造 D_x を，各文節 $b_i(i = 1, \cdots, x-1)$ を係り元の文節とする係り受け関係 d_i の順序付き集合 $\{d_1, \cdots, d_{x-1}\}$ で表す．係り受け関係は，$d_i = \{r_{i,i+1}, \cdots, r_{i,x}\}(1 \leq i \leq x-1)$ で表す．ここで，$r_{i,i+j}$ は，文節 b_i と文節 b_{i+j} の間の関係を表すフラグであり，以下のように「同一」,「異なる」,「係る」,「間」を表す 0, 1, 2, 3 の 4 値をとるものとする．

$$r_{i,i+j} = \left.\begin{cases} 0 & (1 \leq j < dep(i), j = dep(i) - dep(i+j)) \\ 1 & (1 \leq j < dep(i), j \neq dep(i) - dep(i+j)) \\ 2 & (j = dep(i)) \\ 3 & (dep(i) < j \leq n - i) \end{cases}\right\}$$

ここで，文節 b_i の係り先の文節が $b_l(i < l \leq n)$ のとき，$dep(i) = l - i$ と定義する．

3 漸進的係り受け解析手法

従来手法は，文節列 $b_1 \cdots b_n$ からなる文 S を解析する際，文節列 $B_x = b_1, \cdots, b_x(1 \leq x \leq n)$ までが入力された段階で，B_x の係り受け構造が D_x となる確率 $P(D_x|B_x)$ が最大となるものを出力する．2 文節間の関係が「係る」となる確率,「越える」となる確率,「間」となる確率の計 3 種類の確率を用いて，着目している 2 文節間の係り受け確率を計算する．

一方，本研究では,「越える」の関係を「同一」と「異なる」の 2 種類の関係に細分化するため，合計 4 種類の確率を用いて計算する．ある文節の係り先が未入力である場合，その係り先文節の情報がなく，確率を推定することは一般には難しい．しかし，上記の 4 種類の確率を用いれば，既入力文節についてのみの計算を行うことによって，本研究が目的とする係り受け構造を求めることができる．

3.1 確率モデル

本手法では，文節列 $b_1 \cdots b_n$ からなる文 S について，文節列 $B_x = b_1, \cdots, b_x(1 \leq x \leq n)$ まで入力された時点で，確率 $P(D_x|B_x)$ が最大となる係り受け構造 D_x を出力する．$P(D_x|B_x)$ は以下のように計算する．

$$\begin{aligned} &P(D_x|B_x)^2 \\ &= \prod_{i=1}^{x-1} P(d_i|B_x) = \prod_{i=1}^{x-1} P(r_{i,i+1}, \cdots, r_{i,x}|B_x) \\ &= \prod_{i=1}^{x-1} \Bigg(\prod_{j \in \left\{ y \mid \substack{y = dep(i) - dep(i+y), \\ 1 \leq y < dep(i)} \right\}} P(r_{i,i+j} = 0|B_x) \\ &\qquad \times \prod_{j \in \left\{ y \mid \substack{y \neq dep(i) - dep(i+y), \\ 1 \leq y < dep(i)} \right\}} P(r_{i,i+j} = 1|B_x) \\ &\qquad \times P(r_{i,i+dep(i)} = 2|B_x) \\ &\qquad \times \prod_{j=dep(i)+1}^{x-i} P(r_{i,i+j} = 3|B_x) \Bigg) \end{aligned}$$

2 文節間の関係（「同一」,「異なる」,「係る」,「間」）の確率 $P(r_{i,i+j}|B_x)$ は，最大エントロピー法によって学習し推定した値を用いる．

$P(D_x|B_x)$ を最大とする係り受け構造 D_x を求める方法として，文末から文頭に向けて解析し，組み合わせの数を減らしながら一文全体の係り受けを決定する手法 [3] を用いる．解の探索にはビームサーチを用いる．

3.2 素性

最大エントロピー法による確率の推定には，従来手法の素性 (文献 [4] 参照) と同様のものを使用した．

さらに本研究の目的である，係り先が未入力である二つの文節の係り先が同一か否かを同定するために，格フレームから得られる情報を素性として用いる．格フレームでは，述語とその述語に関係する格要素を，述語の用法ごとに記述している．格フレームを使用すると，例えば，入力済みの二つの文節が格要素であり，両文節が一つの格フレームに含まれるならば，それらの文節が未だ入力されていない述語と係り受け関係をもつ可能性を見い出せる．つまり，既入力文節の情報と格フレー

ムの情報を用いることで，既入力文節と未入力文節との関係を捉えることができる．

格フレームには，京都大学格フレーム [5] を使用する．前文節と後文節が「同一」の関係になること，及び,「間」の関係となることを捉えるために，以下の 2 種類の素性を使用する．

- 前文節が格要素であり，後文節が格要素，または後文節の主辞が動詞である場合，両文節が一つの格フレームに含まれるか否か
- 前文節が格要素である場合，前文節と後文節に挟まれた文節の中に，主辞が動詞で，前文節とともに一つの格フレームに含まれる文節が存在するか否か

4 実験

本手法の有効性を確認するために，日本語講演データを用いて係り受け解析実験を行った．

4.1 実験概要

実験データとして，同時通訳データベース [6] に収録されている日本語講演音声の書き起こしデータを使用した．全てのデータに，形態素情報，文節境界情報，節境界情報，係り受け情報が人手で付与されている [1]．

実験は全 16 講演を用いた交差検定により実施した．すなわち，1 講演をテストデータとし，残りの 15 講演を学習データとして係り受け解析を実行した．ただし，従来手法の評価データと合わせるために，16 講演のうち 2 講演を評価データから取り除き，残りの 14 講演 (1,714 文，20,707 文節) に対する実験結果に基づいて評価した．なお，係り受け解析の入力として，形態素情報，文節境界情報，節境界情報は，人手で付与されたものを利用した．また，最大エントロピー法のツールとしては文献 [7] のものを利用した．オプションは，学習アルゴリズムにおける繰り返しを 1000 に設定し，それ以外はデフォルトのまま使用した．また，ビームサーチにおけるビーム幅は 3 とした．

4.2 評価

漸進的係り受け解析の精度を以下の方法で評価した．まず，正解率として，以下の評価指標を導入した．

$$\text{正解率} = \frac{\sum_i^N \sum_{j=1}^{n_i} DepNum(Match(D_j^i, G_j^i))}{\sum_i^N \sum_{j=1}^{n_i} DepNum(D_j^i)}$$

ここで，D_j^i と G_j^i はそれぞれ，文 $S_i(1 \leq i \leq N)$ の解析において，文節 $b_j(1 \leq j \leq n_i)$ が入力された時点で出力する係り受け構造と，正解の係り受け構造を示す．$DepNum()$ は係り受け関係の集合を入力とし，その中に含まれる係り受け関係の数を返す関数である．

$Match()$ は二つの係り受け関係の集合を入力とし，一致する係り受け関係の集合を返す関数である．本手法は，係り先が未入力である文節について，その係り先が他の文節と同一か否かという情報を出力するものの，係り先文節は具体的には決まらない．よって，正解と出力結果の係り先が一致するかを単純には判定できない．

本評価では，出力結果の係り受け構造から擬似的な係り先文節を用意し，一致する係り受け関係の数が最も多くなるように正解と出力の係り先文節を対応付け，一致した係り受け関係を返す．係り先が既入力の場合は，正解との比較において正しく係り先文節が同定できている係り受け関係を返す．

さらに，係り先文節が未入力の場合と既入力の場合に分け，それぞれの係り受け解析の再現率，適合率を測定した．係り先が未入力の場合の再現率，適合率はそれぞれ以下のように計算する．

$$\text{再現率} = \frac{\sum_i^N \sum_{j=1}^{n_i} DepNum(Over(Match(D_j^i, G_j^i)))}{\sum_i^N \sum_{j=1}^{n_i} DepNum(Over(G_j^i))}$$

$$\text{適合率} = \frac{\sum_i^N \sum_{j=1}^{n_i} DepNum(Over(Match(D_j^i, G_j^i)))}{\sum_i^N \sum_{j=1}^{n_i} DepNum(Over(D_j^i))}$$

ここで，$Over()$ は係り受け関係の集合を入力とし，その中で係り先が未入力である係り受け関係の集合を返す関数である．係り先が既入力の場合は，$Over()$ 関数を $NonOver()$ 関数に置き換えて計算する．

図 3 に 6 文節からなる入力文の解析結果の評価例を示す．係り先が未入力である文節は，正解データでは文節 1,2 の二つであり，$DepNum(Over(G_6^i)) = 2$，出力結果では文節 1,2,4 の三つであるため，$DepNum(Over(D_6^i)) = 3$ となる．出力結果では，文節 1,2 と文節 4 の係り先が異なっていることから，擬似的な係り先を二つ用意する．このとき，正解と出力における係り受け関係の一致数が最も多くなるのは，文節 7 と文節 B を対応付けたときであり，$DepNum(Over(Match(D_6^i, G_6^i))) = 2$ となる．

一方，係り先が既入力である文節は，正解では文節 3,4,5 の三つであり，$DepNum(NonOver(G_6^i)) = 3$，出力結果では文節 3,5 の二つであり，$DepNum(NonOver(D_6^i)) = 2$ である．文節 3 と 5 の係り先が正解と出力結果で一致しているため，$DepNum(NonOver(Match(D_6^i, G_6^i))) = 2$ となる．ま

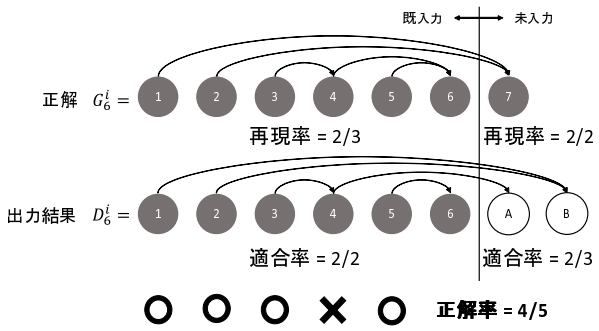


図 3: 係り受け解析精度の評価例

表 1: 実験結果

係り先	再現率	適合率	F 値
未入力	52.62% (17,043/32,389)	59.48% (17,043/28,651)	55.84%
既入力	75.60% (103,863/137,376)	73.60% (103,863/141,114)	74.59%

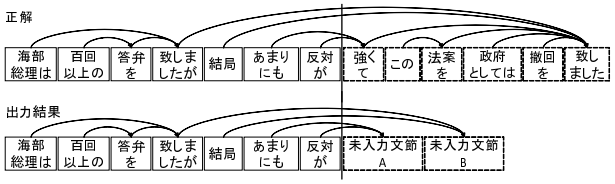


図 4: 係り受け解析結果

た，全体の一致数は $DepNum(Match(D_6^i, G_6^i)) = 4$ となる．

4.3 実験結果

本手法による正解率は 71.22% (120,906/169,765) であった．参考として，従来手法における評価指標で本手法の係り受け解析の精度を評価し，比較した．従来手法の評価では，4.2 節の $Match()$ 関数において，ある文節の係り先が未入力である場合，係り先が未入力文節であると同定できていれば係り受け関係が一致していると判定する．本手法，従来手法の正解率はそれぞれ 74.02% (125,660/169,765) と 73.97% (125,589/169,765) [1]であった．本手法は従来手法よりも精度が向上し，かつ，未入力文節との係り受け関係において，係り先が同一となるか否かの同定を実現できている．

次に，係り先が未入力である場合と既入力である場合に分けて再現率，適合率を測定した結果を表 1 に示す．係り先が未入力である場合の値は，既入力である場合と比べて低い．係り先が未入力である場合は，その係り先文節の情報を取得できないため，係り先が同一であるか否かの判定が難しくなるためであると考えられる．

文「海部総理は百回以上の答弁を致しましたが結局あまりにも反対が強くてこの法案を政府としては撤回を致しました」の正解の係り受け構造と，文節「反対が」まで入力された段階での本手法の出力結果を図 4 に示す．文節「致しましたが」「結局」,「あまりにも」の係り先がいずれも未入力であり,「致しましたが」と「結局」,「あまりにも」と「反対が」の係り先がそれぞれ同一でかつ，両係り先文節は異なることを正しく同定している．

5 おわりに

本論文では，未入力文節との係り受け関係を同定する漸進的な係り受け解析手法を提案した．本手法は，文節が入力されるごとに係り受け解析を実行し，ある文節の係り先が未だ入力されていない場合は，係り先が未入力文節であることを，また，二つの文節の係り先が未入力であった場合，それらの係り先が同一となるか否かを同定する．日本語講演データを用いた解析実験の結果，係り先の同定において，正解率 71.22% を達成した．逐次的な改行挿入手法 [8] に本手法を導入し，その効果を検証することは今後の課題である．

謝辞 本研究は一部，科研費基盤研究（B）(No. 22300051)，ならびに，科研費若手研究（B）(No. 25730134) により実施した．

[1] 従来手法では格フレームに関する素性を用いていない．

参考文献

[1] 村田匡輝, 大野誠寛, 松原茂樹. 読みやすい字幕生成のための講演テキストへの改行挿入. 電子情報通信学会論文誌, Vol. J92-D, No. 9, pp. 1621–1631, 2009.

[2] Tomohiro Ohno and Shigeki Matsubara. Dependency structure for incremental parsing of Japanese and its application. In *Proceedings of the 13th International Conference on Parsing Technologies (IWPT2013)*, pp. 91–97, 2013.

[3] 関根聡, 内元清貴, 井佐原均. 文末から解析する統計的係り受け解析アルゴリズム. 自然言語処理, Vol. 6, No. 3, pp. 59–73, 1999.

[4] Tomohiro Ohno, Shigeki Matsubara, Hideki Kashioka, Takehiko Maruyama, Hideki Tanaka, and Yasuyoshi Inagaki. Dependency parsing of Japanese monologue using clause boundaries. *Language Resources and Evaluation*, Vol. 40, No. 3-4, pp. 263–279, 2007.

[5] Daisuke Kawahara and Sadao Kurohashi. Case frame compilation from the web using high-performance computing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC2006)*, pp. 1344–1347, 2006.

[6] Shigeki Matsubara, Akira Takagi, Nobuo Kawaguchi, and Yasuyoshi Inagaki. Bilingual spoken monologue corpus for simultaneous machine interpretation research. In *Proceedings of the 3rd Language Resources and Evaluation Conference (LREC2002)*, pp. 153–159, 2002.

[7] Zhang Le. Maximum entropy modeling toolkit for python and c++. `http://homepages.inf.ed.ac.uk/lzhang10/maxent_toolkit.html`. [Online; accessed 18-Dec.-2013].

[8] 大野誠寛, 村田匡輝, 松原茂樹. 講演のリアルタイム字幕生成のための逐次的な改行挿入. 電気学会論文誌, Vol. 133-C, No. 2, pp. 418–426, 2013.