機械学習を用いた日本語機能表現のチャンキングおよび係り受け解析

注連隆夫[†1] 土屋雅稔[†2] 松吉俊[†1,†3]
宇津呂武仁[†4] 佐藤理史[†3]

1. はじめに

機能表現とは,「にあたって」や「をめぐって」のように，2つ以上の語から構成され，全体として1つの機能的な意味をもつ表現である．一方，この機能表現に対して，それと同一表記をとり，内容的な意味をもつ表現が存在することがある．例えば，文（1）と文（2）には,「にあたって」という表記の表現が共通して現れている．

(1) 出発する にあたって、荷物をチェックした。
(2) ボールは、壁 にあたって 跳ね返った。

文（1）では，下線部はひとかたまりとなって,「機会が来たのに当面して」という機能的な意味で用いられている．それに対して，文（2）では，下線部に含まれている動詞「あたる」は，動詞「あたる」本来の内容的な意味で用いられている．これらの表現においては，機能的に用いられている場合と，内容的に用いられている場合とを識別する必要がある．以下，文（1），文（2）の下線部のように，表記のみに基づいて判断すると，機能的に用いられている可能性がある部分を**機能表現候補**と呼ぶ．

ここで，日本語複合辞用例データベース[1](以下，**用例データベース**と呼ぶ) は，機能表現の機械処理を研究するための基礎データを提供することを目的として設計・編纂されたデータベースである．この用例データベースは，現代語複合辞用例集[2] に収録されている 125 種類の複合辞および，その異形 (合計 337 種類の機能表現) を対象として，機能表現候補と一致する表記のリストと，個々の機能表現候補に対して最大 50 個の用例を毎日新聞 (1995 年) から収集したものを収録している．そして，各機能表現候補が文中において果たしている働きが，機能的用法・内容的用法のいずれであるかを人手で付与している☆1．

用例データベースにおいて，機能的用法か内容的用法かの判別が必要な表現は 111 表現存在する．111 表現に対する，既存の解析系の扱いを調べてみた．形態素解析器 JUMAN ☆2 と構文解析器 KNP ☆3 の組み合わせによって，機能的な意味で用いられている場合と内容的な意味で用いられている場合とが識別される可能性がある表現は 111 表現中 43 表現である．また，形態素解析器 ChaSen ☆4 と構文解析器 CaboCha[3] の組合わせを用いた場合には，識別される可能性がある表現は 111 表現中 40 表現である．このような現状を改善するには，機能表現候補の用法を正しく識別する検出器と検出器によって検出される機能表現を考慮した係り受け解析器が必要である．そこで本稿では，SVM を用いたチャンキングによって機能表現検出器を実現し[4]，その機能表現検出器と工藤らの SVM を用いた統計的係り受け解析手法[3] を利用して構築した機能表現を考慮した係り受け解析器を使用して，機能表現を考慮した係り受け解析を実現する．

2. 機能表現検出器

本稿で使用している機能表現検出器は，SVM を用いたチャンキングによって実現している．具体的には SVM を用いたチャンカー YamCha ☆5 を利用して，形態素解析器 ChaSen による形態素解析結果を入力とする機能表現検出器を実装した．チャンカーの学習には，形態素素性，チャンク素性，チャンク文脈素性を利用した．形態素素性には，形態素解析器 ChaSen の形態素解析結果を採用した．チャンク素性には，機能表現候補を構成している形態素の数と，機能表現候補中における形態素の相対位置の情報の 2 つの組を採用した．チャンク文脈素性には，機能表現候補の前後 2 つずつの形態素に付与された，形態素素性とチャンク素性を採用した．上記の素性により学習を行い実装された機能表現検出器は，5 節の評価実験において F 値で約 93.5 という高い性能を示した．本稿では，この機能表現検出器を利用する．

3. SVM を用いた統計的係り受け解析

本稿で提案している機能表現を考慮した係り受け解析は，工藤らの SVM を用いた統計的係り受け解析器 CaboCha[3] を利用している．本節では，工藤らの手法の学習・解析アルゴリズム，学習・解析に使用する素性について述べる．

3.1 学習・解析アルゴリズム

工藤らの手法は，入力文 B に対する，条件付き確率 P(D|B) を最大にする係り受けパターン列 D を求める従

†1 京都大学大学院 情報学研究科
†2 豊橋技術科学大学 情報メディア基盤センター
†3 名古屋大学大学院 工学研究科
†4 筑波大学大学院 システム情報工学研究科

☆1 実際には，機能的用法については,「現代語複合辞用例集」の用法であることを表すラベル F，接続詞的用法であることを表すラベル A，その他の機能的用法であることを表すラベル M という細分類がされており，一方，内容的用法については，機能表現候補は，用法判定単位として不適切であることを表すラベル B，機能表現候補の読みが，判定対象の機能表現の読みと一致しないことを表すラベル Y，内容的用法であることを表す C という細分類がされている．本稿では，ラベル F，A，M を機能的用法として統合し，ラベル C，B，Y を内容的用法として統合して扱っている．

☆2 http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman/
☆3 http://nlp.kuee.kyoto-u.ac.jp/nl-resource/knp/
☆4 http://chasen.naist.jp/hiki/ChaSen/
☆5 http://chasen.org/~taku/software/yamcha/

表 1　係り受け解析において用いる素性

静的素性	係り元/係り先の文節	主辞見出し，主辞品詞，主辞書品詞細分類，主辞活用，主辞活用形，語形見出し，語形品詞，語形品詞細分類，語形活用，語形活用形，括弧有無，句読点の有無，文節の位置 (文頭，文末)
	文節間	距離 (1,2-5,6 以上)，助詞，括弧，句読点の有無
動的素性	係り先に係る文節の静的素性	
	係り元に係る文節の静的素性	
	係り元が係る文節の静的素性	

来の手法と異なり，チャンキングを段階的適用することによって係り受け解析を実現している．ここで登場した入力文 B とは，あらかじめ文節にまとめられ，属性付けされた文節列 $b_1, b_2, ..., b_m$ を表しており，係り受けパターン列 D とは，$Dep(1), Dep(2), ..., Dep(m-1)$ を表している．ただし，$Dep(i)$ は，文節 b_i の係り先文節番号を示す．実際には，以下のようなアルゴリズムを使用して，段階的にチャンキングを行っている．

(1) 入力文節すべてに対し，係り受けが未定という意味の O タグを付与する．
(2) 文末の文節を除く O タグが付与された文節に対し，直後の文節に係るか推定．係る場合は D タグを付与．後から 2 番目の文節は無条件に D タグを付与．
(3) O タグの直後にあるすべての D タグおよびその文節を削除する．
(4) 残った文節が一つ (文末の文節) の場合は終了，それ以外は 2. に戻る．

このアルゴリズムにおける係り受け関係の同定には，SVM を用いている．この場合，従来手法では，訓練データ中の全ての 2 文節の候補を学習事例として抽出していた．しかし，このような抽出方法では，学習データを不必要に多くしてしまい，学習の効率が悪い．それに対して，工藤らの手法では，学習も解析時と同じアルゴリズムを採用することにより，学習で使われる文節組のセットを，隣り合う文節組のみとしている．これにより，負例を不必要に増えるのを防ぐことができている．

3.2　学習・解析に使用する素性

SVM の学習・解析に使用する素性は，**表 1** に示す通りである． 静的素性とは，文節の作成時に決定される素性を示しており，動的素性とは，係り関係そのものを素性としたものである．また，主辞とは文節内で品詞が特殊，助詞，接尾辞となるものを除き，文節末に一番近い形態素を指し，語形とは文節内で品詞が特殊となるものを除き，文節末に一番近い形態素のことを指す．

4.　機能表現を考慮した係り受け解析

本稿で提案している機能表現を考慮した係り受け解析の流れは，**図 1** の通りである．まず，ChaSen によって形態素解析を行う．次に，形態素解析結果に対して，機能表現検出器を用いて，機能表現検出を行う．その際，検出された機能表現は，構成している形態素列を連結し，一つの形態素として出力される．最後に，その出力結果に対して，機能表現を考慮した係り受け解析器を用いて，係り受け解析を行う．

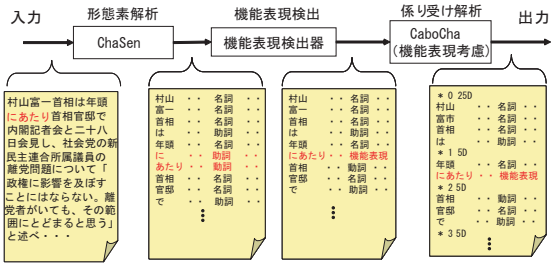


図 1　機能表現を考慮した係り受け解析

機能表現を考慮した係り受け解析器の学習において，形態素を連結して作られた機能表現に対して，新たに品詞名を付与する必要がある．用例データベースによると，機能表現は，接続詞相当の働きをするもの (接続詞型) と助詞相当の働きをするもの (助詞型)，助動詞相当の働きをするもの (助動詞型) に分類することができる．さらに，助詞型の機能表現は，接続助詞相当のもの (接続辞類)，格助詞相当のもの (連用辞類)，連体助詞相当のもの (連体辞類) に細分類することができる．そこで，本稿では，**表 2** のような品詞体系を採用した．そして，現代語複合辞用例集[2] に掲載されている各機能表現と品詞分類との対応に基づいて，機能表現への品詞の付与を行った．特に，接続詞型になる可能性のある機能表現については，文頭に出現した場合は接続詞型とし，文頭以外の場合は助詞型とした．

本稿では，SVM を用いた統計的係り受け解析手法の学習・解析ツールとして CaboCha を利用して機能表現を考慮した係り受け解析器を実現している．具体的には，CaboCha の係り受け解析における訓練データを機能表現を考慮したものに変換するという方法を用いている．機能表現を考慮した係り受け解析の訓練データを作成するために必要な情報は二つある．一つは，対象文における係り受け解析の訓練データ．もう一つは，対象文における機能表現の情報．この二つの情報を用いて，**図 2** の流れで訓練データを作成し，学習を行っている．

図 2 の訓練データ作成モジュールでは，訓練データは，末尾の文節から順番に以下のアルゴリズムに従って作成している．

1. 機能表現を構成している形態素列を連結する．
2. 連結する形態素列が複数の文節にまたがっている場合，文節の連結も行う．連結後の文節の係り先は，連

表 2　機能表現の品詞体系

機能表現の分類		付与する品詞
接続詞型		接続詞-機能表現
助詞型	接続辞類	助詞-接続助詞-機能表現
	連用辞類	助詞-格助詞-機能表現
	連体辞類	助詞-連体化-機能表現
助動詞型		助動詞-機能表現

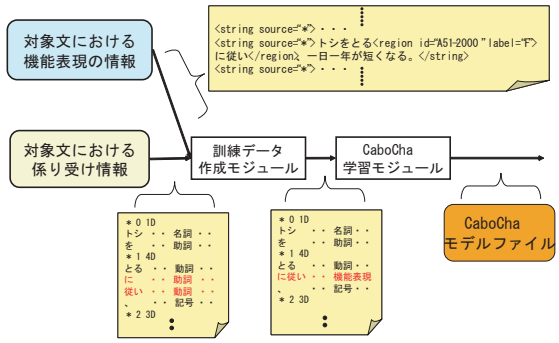


図 2　機能表現を考慮した係り受け解析器の学習の流れ

結文節中の末尾の文節の係り先を採用する．

3a.　助詞・助動詞型の機能表現の場合で，連結した文節の先頭形態素が，機能表現の場合は，直前の文節に連結する．連結後の文節の係り先は，連結文節中の末尾の文節の係り先を採用する．

3b.　接続詞型の機能表現の場合で，一文節が機能表現のみで構成されない場合は，機能表現のみで一文節を構成するように文節を分解する．

4.　文節の連結，分解に伴う文節 ID，係り先の変化を反映させる．

機能表現を考慮した係り受け解析の学習と機能表現を考慮しない係り受け解析の学習における学習に使用する素性の変化を**図 3**，**図 4** の文における「して」,「保護者として」という文節と「参加した」という文節の係り受け関係の学習・解析に使用する素性について見てみる．まず，図 4 においては，文節の区切りが機能表現を考慮したものになっている．そして，それによって注目する文節が図 3 では,「して」という文節なのに対し，図 4 では「保護者として」となる．

その変化によって，実際に学習・解析に使用する素性も，**表 3** のように機能表現を考慮したものに変化する．具体的には，係り元の文節が「して」から「保護者として」と変化することによって，係り元の主辞が「し」から「保護」に，係り元の語形が「て」から「として」に変化している．また，着目している係り元に係る文節も「保護者と」から「甥の」に変化している．このように学習・解析に使用する素性を機能表現を考慮したものにすることによって，機能表現を考慮した係り受け解析が実現できると考えられる．

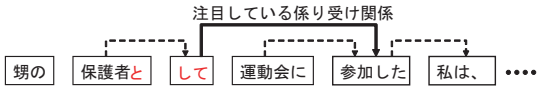


図 3　係り受け解析例 (機能表現考慮せず)

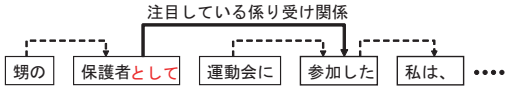


図 4　係り受け解析例 (機能表現考慮)

表 3　係り受け解析において用いる素性の例

			機能表現を考慮しない	機能表現を考慮する
静的素性	係り元	主辞見出し	し	保護
		主辞品詞	動詞	名詞
		主辞品詞細分類	自立	サ変接続
		主辞活用	サ変・スル	*
		主辞活用形	連用形	*
		語形見出し	て	として
		語形品詞	助詞	助詞
		語形品詞細分類 1	接続助詞	格助詞
		語形品詞細分類 2	*	機能表現
		括弧の有無	0	0
		句読点の有無	0	0
		文節の位置（文頭）	0	0
		文節の位置（文末）	0	0
	係り先	主辞見出し	た	た
		主辞品詞	助動詞	助動詞
		主辞活用	特殊・タ	特殊・タ
		主辞活用形	基本形	基本形
		語形見出し	た	た
		語形品詞	助動詞	助動詞
		語形活用	特殊・タ	特殊・タ
		語形活用形	基本形	基本形
		括弧の有無	0	0
		句読点の有無	0	0
		文節の位置（文頭）	0	0
		文節の位置（文末）	0	0
	文節間	距離	2 以上 5 以下	2 以上 5 以下
		助詞	に	に
		括弧の有無	0	0
		句読点の有無	0	0
動的素性	着目している係り先に係る文節	語形見出し	に	に
	着目している係り先に係る文節	語形見出し	と	の
	着目している係り先が係る文節	主辞品詞	名詞	名詞

表 4　係り受け解析器用データセットの各統計量

用法			全文数
機能的用法	内容的用法	計	
4675	817	5492	38400

5.　実験と考察

本稿で提案する係り受け解析器の学習および解析を行った．本稿では，判定が必要な 111 表現のなかでも，新聞記事においても，機能的用法と内容的用法の両方が一定の割合で出現する 32 表現を対象とする (実際には，そのような表現は 60 表現程度存在するが，データ整備の都合上，本稿の範囲では 32 表現のみを対象とした)．実験で使われた機能表現検出器は，2 節で説明したものを使用した．この際，素性は，形態素素性，チャンク素性，チャンク文脈素性を使用した．

5.1　データセット

係り受け解析器の学習データとしては，京都テキストコーパス[5] を利用する．ここで，オリジナルの京都テキストコーパスには，機能表現の情報は付与されていないので，まず，京都テキストコーパス 38,400 文に存在する全ての機能表現に対して，用法ラベルを付与した．

5.2　評価尺度

実験結果を評価する際の尺度には，以下の式で表される係り先精度，係り元精度を用いた．ただし，FE 文節とは，機能表現を含む文節を表している．

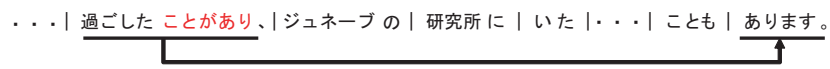
$$\text{係り先精度} = \frac{\text{係り先を正しく同定できた FE 文節数}}{\text{機能表現候補数}}$$

$$\text{係り元精度} = \frac{\text{係り元を正しく同定できた FE 文節数}}{\text{機能表現候補数}}$$

5.3　評価結果

本稿で提案している機能表現を考慮した係り受け解析

(a) ベースラインによる失敗例



(b) 提案手法による成功例

図 5 係り先同定の改善例

(a) ベースラインによる失敗例

(b) 提案手法による成功例

図 6 係り元同定の改善例

表 5 係り受け解析の評価結果 (%)

		係り先精度	係り元精度
ベースライン	CaboCha(機能表現抜き)	91.3	81.4
	CaboCha(オリジナル)	91.4	82.6
提案手法	検出器出力使用	91.9	82.8
	正解用法ラベル使用	92.2	83.4

器と各ベースラインの精度を**表 5** に示す．評価方法としては，京都テキストコーパスを訓練・評価データとする 10 分割交差検定を行った．表 5 中の「CaboCha(機能表現抜き)」は，ipadic 辞書に連語として登録されている機能表現の内，評価対象の機能表現にあたるものを機能表現を構成している形態素に分解し，学習し直している．「CaboCha(オリジナル)」は，他のモデルと同一の訓練データセットを用いて学習を行ったものである．また，機能表現を考慮した係り受け解析では，機能表現用法ラベルとして，2 節で述べた検出器により出力された結果を用いた場合，および，人手で付与した正解用法ラベルを用いた場合の二通りを評価した．提案手法は，高頻度な表現において，ベースラインと同等かそれ以上の性能を達成しており，また，その他の表現に対する性能も含めて，総体として，ベースラインを上回る性能を達成している．特に，CaboCha(機能表現抜き) との比較においては，係り元精度については，有意水準 8%で上回っている☆．

係り先の推定が改善された表現に注目すると，その表現を構成している形態素列を独立した形態素として扱うのではなく，一つの機能表現として検出していることが効果的に働いていると考えられる．例えば，「ことがある」を連用活用した「ことがあり」の場合，構成要素である形態素列を独立に扱うと．**図 5** (a) のように構成要素の一つである動詞「ある」の連用形「あり」が付近の動詞と並立に誤って係ってしまうことがある．それに対して，「ことがあり」を機能表現として扱った場合，図 5 (b) のように正しく係り先を推定できる．

一方，係り元の推定が改善された表現に注目した場合も，その表現を構成している形態素列を独立に扱うのではなく，一つの機能表現として検出していることが効果的に働いていると考えられる．例えば，「に応じて」の場合，構成要素である形態素列を独立に扱うと，**図 6** (a) のような例文において，「限度に」という文節が動詞を含む文節に係りやすいという特徴をもっているので，誤って「応じて」という文節に係ってしまう．それに対して，「に応じて」を機能表現として扱った場合，図 6 (b) のように，「限度に」の係り先を正しく推定することができる．

☆ この評価においては，手違いのため，京都テキストコーパスの約 7 割程度のデータのみを用いた評価となっており，このため，現状では，十分な有意水準を達成できていない．

6. おわりに

本稿では，形態素を単位とするチャンク同定問題として機能表現検出タスクを定式化し，機械学習を利用して機能表現の検出を実現し，さらに，機能表現を考慮した係り受け解析を実現した．

参 考 文 献

1) 土屋雅稔, 宇津呂武仁, 松吉俊, 佐藤理史, 中川聖一: 日本語複合辞用例データベースの作成と分析, 情報処理学会論文誌, Vol. 47, No. 6, pp. 1728–1741 (2006).
2) 国立国語研究所: 現代語複合辞用例集 (2001).
3) 工藤拓, 松本裕治: チャンキングの段階適用による係り受け解析, 情報処理学会論文誌, Vol. 43, No. 6, pp. 1834–1842 (2002).
4) 土屋雅稔, 注連隆夫, 高木俊宏, 内元清貴, 松吉俊, 宇津呂武仁, 佐藤理史, 中川聖一: 機械学習を用いた日本語機能表現のチャンキング, 自然言語処理, Vol. 14, No. 1 (2007). (掲載予定).
5) 黒橋禎夫, 長尾眞: 京都大学テキストコーパス・プロジェクト, 言語処理学会第 3 回年次大会発表論文集, pp. 115–118 (1997).