複数の述語間関係を考慮した日本語述語項構造解析

大村 舞　　進藤 裕之　　松本 裕治
奈良先端科学技術大学院大学 情報科学研究科
{omura.mai.oz5, shindo, matsu}@is.naist.jp

1 はじめに

本稿では，複数の述語を考慮した日本語述語項構造解析の手法を提案する．既存研究において，述語と直接係り関係にある項は 8 割から 9 割と高い解析精度を達成している一方，述語と直接係り関係にない (文内係り受けなし) 項は，ガ格でも 5 割程度の精度となっている [10, 12]．特にガ格は文内係り受けなしの項が 4 割を占めており [1]，これらの解析精度が全体の解析精度に大きく影響する．本稿では，1 文単位の解析に焦点を絞り，特に文内係り受けなしの項の解析精度向上に取り組む．

項が文内係り受け無しの関係になる原因のひとつとして，複数の述語が存在するために，一方の述語と統語的に直接係り関係にないことが挙げられる．このため，複数の述語項関係を同時に考慮することによって，さらなる解析精度向上が期待できる．これまでの研究では，1 つの述語を単位として文を解析するモデルが多く，文中の複数の述語，あるいは複数の述語項構造を考慮して解析する手法はあまり多くない．そこで本稿では，複数の述語あるいは複数の述語項を同時に考慮して解析するモデルを提案する．提案モデルでは特に文内係り受けなしの項の精度が 6 割以上向上することが分かった．

2 関連研究

日本語の述語項構造解析モデルとしては，述語に対して尤もらしい項を選択するモデル [4]，述語の語義曖昧性と同時に解くモデル [9]，文中の項同士を比較して述語に対して尤もらしい項を選出するトーナメントモデル [2, 12] などが存在する．日本語の場合，述語に対する項が統語的に直接関係でない，あるいは文外に存在することも多い．そこで，大規模な新聞コーパスから得た統計値をモデルに取り入れたり [4, 12]，格フレームの情報を用いたりする [6, 11] など，外部知識を利用した研究も多い．しかし，これらのモデルは項候補あるいは述語と項単位で解析が行われており，文中の述語同士の関係を考慮したモデルではない．

複数の述語間を考慮している日本語述語項構造解析モデルとして吉川らの Markov Logic を用いたモデル [10] がある．Markov Logic は一階述語論理と Markov Network を組み合わせたモデルで，一階述語論理に対して重みを統計的に学習することにより，緩い制約を設けることができる．しかし吉川らのモデルは計算量の問題から，そのまま文間へ拡張することが困難であるとされている．吉川らのモデルは文内係り受けなしのガ格で高い精度を達成することが分かっており，文全体の解を最適化することが有効であることが分かる．

本研究では，リランキングモデル [1] をベースにした述語項構造解析モデルを提案する．リランキングモデルはインスタンスの N-Best 解を出力し，N-Best 解の中で尤もらしいインスタンスを選出するモデルである．一度ベースのモデルによって解析したのち，解析結果を用いて再度尤もらしいインスタンスを選択するため，解析結果やインスタンス全体に対する大域的な素性を入れることができるというメリットがある．リランキングモデルを用いて述語項構造解析に取り組んだ研究として Toutanova ら [8] の研究が挙げられる．しかし Toutanova らのモデルはひとつの述語項をインスタンスとしてモデルを構築しており，複数の述語や文全体の解を最適化したものではない．本研究では，1 つの文に対してラティス構造を構築し，N-Best 解を求めることにより，複数の述語項を同時に考慮して全体の述語項構造解析を行うモデルを提案する．

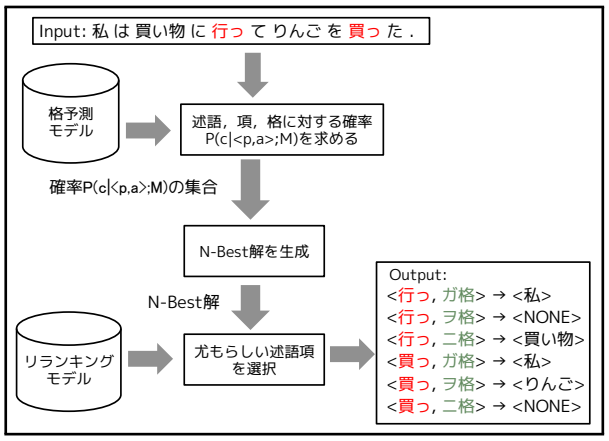

図 1: 提案モデルの流れの概略．

3 提案モデル

本稿では，複数の述語関係を考慮したモデルを提案する．図 1 に提案モデルの流れを示す．このモデルは 1 文単位で解析を行う．まず，1 文に対して，格予測モデルで述語に対する項の格らしさの確率を計算する．

[1] NAIST テキストコーパス (ver1.5) 中のガ格のうち，ゼロ項 (文内係り受けなし) は 105646 個中 47876 個存在した．

そして，求めた確率を基に文の述語項構造の N-Best 解を得る．最後に出力した N-Best 解の中で尤もらしい述語項構造を出力する．はじめに，ベースモデルかつ格予測モデルとなる松林らのモデルについて 3.1 節で概略を述べる．その次に複数の述語の関係を捉える新たな素性について 3.2 節で説明する．最後に提案モデルのフレームワークとなるリランキングモデルについて 3.3 節で述べる．

3.1 格予測モデル (ベースモデル)

ベースモデルとして，松林らの格予測モデル [11] を用いる．松林らのモデルは，以下の様な手順でモデルを構築し解析を行う．

1. 与えられた文から述語と項候補を抽出する．述語は NAIST テキストコーパス [3] のアノテーションから抽出する．項候補は松林らのモデルにならい，特定の品詞から抽出する．
2. 抽出した述語と項に対して格 (ガ格，ヲ格，ニ格，NONE) を多値分類として予測するモデル M を構築する．NONE は与えられた述語と項は格関係が無いことを表す特別な格とする．そして構築したモデルによって述語と項に対する格を予測する．この際，モデルの予測スコア [2] も同時に出力する．
3. 述語に対する文内候補から，ガ格，ヲ格，ニ格それぞれの格で最もスコアの高い項を選ぶ．予めそれぞれの格で閾値を設け，その閾値より項のスコアが高ければ項として認定する．

各閾値は開発データで最も F 値が高くなるように最適化したものを用いる．松林らは SVM でモデルを構築していたのに対し，本研究では最大エントロピーモデルで構築する．ベースモデルの素性も松林らのモデルに準ずる．この素性は $f(x)$ と表現する [3]．この格予測モデルを用いた手法は，述語単位でスコアが最大となるように出力するため，述語間の関係は考慮していない．次節で説明する素性やリランキングモデルを用いることで複数の述語を考慮したモデルに拡張する．

3.2 複数の述語関係を考慮した素性

本稿では，文中の複数の述語に関する大域的な素性をモデルに取り入れることで述語項構造解析の精度向上に取り組む．この新たに追加する素性は $f_m(x)$ と表現する．

飯田ら [2] は統語的なパターンを取り入れたモデルを提案しており，統語情報が精度向上に貢献することを示している．本稿でもこのパターンを参考にし，係り受けパスをベースにした素性を提案する．違いとしては，飯田らは比較対象としている項同士間のパスを用いた素性をモデルに取り入れているのに対し，本稿ではある述語と対象としている述語間のパスを用い，ノードはある程度具体化したものを用いる．

[2] 今回は構築したモデルによって出力される述語 p と項 a に対する c の確率 $P(c|\langle p, a\rangle; M)$ を用いる

[3] この x は，述語，項，格の三組を表現している

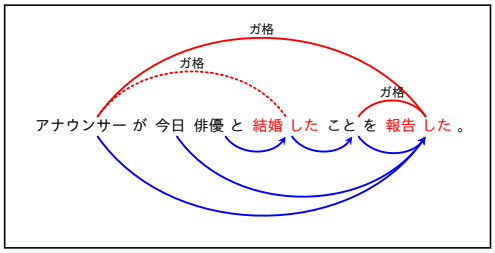

図 2: 「アナウンサーは今日結婚したことを報告した.」の述語項構造解析の例．赤文字は述語，青線は係り受けパス，赤線は述語項関係，点線は文内係り受けなしの述語項関係を示している．

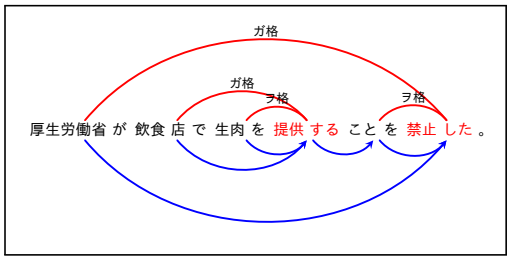

図 3: 「厚生労働省が飲食店で生肉を提供することを禁止した.」の述語項構造解析の例．

文内係り受けなし関係の項が発生する原因として，文中に複数の述語が存在していることが 1 つ挙げられる．例えば，図 2 の「アナウンサー」と「結婚した」は統語的には文内係り受けなし関係であるが,「ガ格」の述語項関係を持っている．一方「アナウンサー」は「報告した」と直接係り関係であり，同様に「ガ格」の述語項関係を持っている．二つの述語「結婚した」と「報告した」の間には「ことを」という節を挟んだ係り受けパスが存在する．ここから「○○したことを○○した」というパターンは項を共有する可能性があることが分かる．しかし，図 3 の場合,「提供する」と「禁止する」では図 2 と同じパターンであるにも関わらず,「厚生労働省」という項を共有していない．これは「禁止する」という述語が「提供する」のガ格と共通しにくいという傾向から共有しないことが推定できる．このように，統語的な情報のみではなく，述語自身の特徴も考慮したパスが述語項構造解析に有効であると考える．

対象としている項 a と述語 p について，上記の傾向を利用して，以下の様な手順で係り受けパスを抽出する．

1. 対象としている項と直接係り関係を持つ述語があるか確認する．この直接係り関係の述語を p' とする．
2. 述語 p', p の間の係り受けパスを抽出する．抽出した係り受けパスのそれぞれのノードを以下の形で変換したそれぞれを最終的なパスとする．
   - ノード節の主辞の品詞
   - ノード節の主辞と機能語の品詞
   - ノード節の主辞のクラスタ
   - ノード節の主辞と機能語のクラスタ

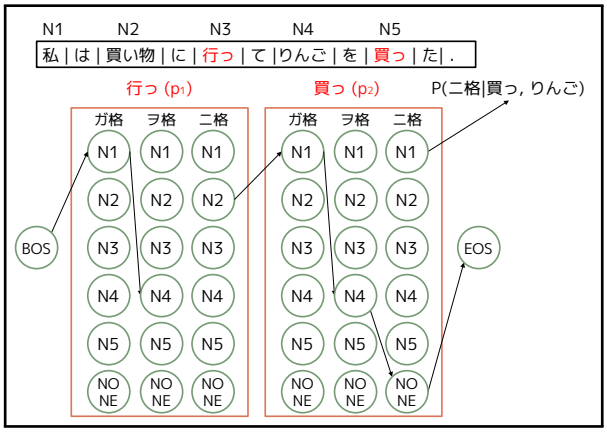

図 4: N-Best 解を生成するための 1 文の表現のイメージ

ノードに品詞を用いたのは，パスをある程度抽象化しつつも統語的な情報を取り入れるためである．クラスタはその単語の特徴は前後の単語によって特徴付けられると仮定し，ベクトル表現によってクラスタリングされたものを指す．ベクトルの生成には，word2vec[4] を用いる．データには毎日新聞 95 年分を利用した．そして，word2vec から得たベクトルを K-means アルゴリズムによってクラスタリングする．抽出したパスを素性 $f_m(x)$ をモデルに組み込むことで精度向上が見込めるかを確認する．

3.3 リランキングモデル

3.3 節ではリランキングモデルを基にした述語項構造解析について説明する．リランキングモデルでは，解に対してスコアが高い順に生成した N-Best 解を生成し，N-Best 解の中から尤もらしい解を選択する．本稿では文中に存在する述語に対する述語項構造の集合を解とし「文の述語項構造」と呼ぶ．

3.3.1 リランキングモデルの入力と N-Best 解

N-Best 解をモデル化するために，文 s の述語項構造を定式化する．s に含まれる述語の集合を P，s に含まれる項候補の集合を A ($NONE \in A$) とする．述語に対する格関係の集合を $C = \{$ ガ格, ヲ格, ニ格 $\}$ とする．文 s に対する述語項構造 x はそれぞれの述語とそれぞれ格関係 $P \times C$ に対して当てはまる項を列挙したリストで表現する．例えば，図 4 の文の述語項構造 x は

$$x = (\text{私}, \text{NONE}, \text{買い物}, \text{私}, \text{りんご}, \text{NONE})$$

のように表現される．そして文 s の述語項構造 x のスコアは

$$\text{score}(x) = \prod_{p \in P} \prod_{c \in C} \prod_{a \in x} P(c|\langle p, a\rangle; M) \qquad (1)$$

と表現する．$P(c|\langle p, a\rangle; M)$ は 3.1 節で構築したモデル M で出力される確率値である．項が NONE の場合は 3.1 節で用いた閾値を確率として用いる．

式 (1) を基にして N-Best 解を出力するために，図 4 のようなラティス構造を構築してモデルに与える．各ノードが述語 p と格 c に対する項 a を表現しており，確率 $P(c|\langle p, a\rangle; M)$ を持っているものとする．つまり BOS から EOS までの 1 つのパスがひとつの述語項構造に相当する．実際に N-Best 解を求める手法として，永田らの N-Best 探索 [5] を用いた．N-Best 探索によって述語項構造の N-Best 解を出力したのち，リランキングモデルによって尤もらしい述語項構造 $\hat{x}$ を選択する．

3.3.2 リランキングモデルの学習

前節で得た文 s の述語項構造の N-Best 解を X と表現する．このときリランキングモデルは式 (2) によって尤もらしい文の述語項構造 $\hat{x}$ を選択する．

$$\hat{x} = \max_{x \in X} w \cdot F(x) \qquad (2)$$

$F(x)$ は素性関数，w は重みを表わす．つまり式中の $w \cdot F(x)$ が最も高くなる文の述語項構造 $\hat{x}$ を選択する．重み w は平均化パーセプトロンを用いて学習し，N-Best 解の中でも正解の述語項構造に最も項の一致率が高い述語項構造で更新する．

格予測モデルで用いた素性ベクトルを連結したものを素性関数 $F(x)$ した．素性関数 $F(x)$ には，ベースの素性 $f(x)$ に加え，文中の述語同士の共起関係，述語と格のペアの共起関係，述語項同士の共起関係も素性として加えた．これらの加えた素性によって，1 つの述語項のみではなく，複数の述語項の共起関係を考慮して解を最適化することが目的である．項が NONE の場合，ベースの素性 $f(x)$ に相当するものが無いため，述語の表層，述語の原形，そして京都大学格フレーム[5] から述語と格に対する頻度情報を素性として加えた．具体的には述語に対して，それぞれの格関係を持つ項の頻度を計算し，その頻度の割合で発火させたものを用いる．

4 実験

4.1 実験設定

評価データとして，述語項構造のアノテーションデータである NAIST テキストコーパス[6] [3](ver 1.5) を用いた．平ら [7] と同様に，NAIST テキストコーパスの 1 月 1 日～11 日までのニュース記事と 1～8 月の社説記事を訓練，1 月 12 日～13 日のニュース記事と 9 月の社説記事を開発，評価の 3 つにコーパスを分け，実験を行った．松林ら [11] と同様に，システムが出力した項の位置とラベルが NTC に項として示されている共参照クラスタ内の形態素いずれかと一致すれば正解，しなければ不正解とし，適合率，再現率，F 値を求めることで評価を行った．本稿では，直接係り関係，文内係り無しの項のみ評価し，同一文節，事態性名詞の項は評価しない．

[4] https://code.google.com/p/word2vec/

[5] http://www.gsk.or.jp/catalog/gsk2008-b/

[6] https://sites.google.com/site/naisttextcorpus/

表 1: 実験結果．各数値は F 値を示している．∗ が付いたものはマクネマー検定によりベースラインと比較して $p < 0.01$ で有意差があったことを示している．

		直接係り	文内係無	合計
ガ格	Base	0.878	0.444	0.781
	Base + $f_m(x)$	∗0.882	∗0.452	∗0.785
	Reranking	∗0.924	∗0.633	∗0.857
	Reranking + $f_m(x)$	∗**0.927**	∗**0.649**	∗**0.863**
ヲ格	Base	0.928	0.301	0.892
	Base + $f_m(x)$	∗0.933	∗0.304	∗0.896
	Reranking	0.925	∗0.402	0.894
	Reranking + $f_m(x)$	∗**0.934**	∗**0.420**	∗**0.903**
ニ格	Base	0.629	0.297	0.610
	Base + $f_m(x)$	∗0.633	∗0.295	∗0.613
	Reranking	∗**0.747**	∗**0.381**	∗**0.723**
	Reranking + $f_m(x)$	∗0.741	∗0.369	∗0.718

今回の実験では，いずれの設定も閾値はそれぞれガ格 0.3，ヲ格 0.4，ニ格 0.1 として求めた．リランキングモデルの N-Best 解は $n = 10$，重み w の学習のイテレーション回数は 15 に固定した．リランキングモデルの重みを学習するための N-Best 解は訓練データを 10 分割し，それぞれ 9 個を格予測モデルの訓練，1 個を出力というように 10 回繰り返して生成した N-Best 解を用いた．評価データの N-Best 解はすべての訓練データを用いて学習した格予測モデルによって生成した．

4.2 実験結果と考察

表 1 に実験結果を示す．Base はベースモデル (格予測モデル) のみ，Reranking はリランキングモデル，$+f_m(x)$ は 3.2 節で説明した素性 $f_m(x)$ をベースモデルに加えたことを示す．表 1 に示すとおり，素性 $f_m(x)$，リランキングモデルはいずれの項のタイプでもベースモデルより精度が向上することが分かった．リランキングのみと素性 $f_m(x)$ を加えたものを比較すると，ガ格とヲ格は述語間の係り受けパスを素性として加えた方が精度向上することが分かった．

提案モデルでも解析できていないものとして,「いじめにどう (学校ガ) 取り組んだかについて、各学校に詳細な報告を〜」の「取り組んだ」のガ格が埋まっていないのように，格を埋めていない解がいくつか見受けられた．提案モデルでは，それぞれの述語に対して項が実際に埋まっているかどうかを評価することができる．そのため，項が埋まっているか否かを情報として取り入れることで精度向上できるのではないかと考えられる．また，リランキングモデルの重みの更新評価方法などによって結果に違いがでる可能性がある．今後はこのように述語項の状態を考慮した素性を取り入れる，評価方法を工夫するなどして，モデルの精度を向上させたいと考えている．

5 おわりに

本稿では，リランキングモデルを基にした複数の述語を考慮した述語項構造解析モデルを提案した．複数の述語，述語項の解を全体的に最適化するリランキングモデルを用いることで述語項構造解析の精度，とくに文内項の精度が向上することが分かった．

今後は 1 つの文のみではなく，複数の文を対象とした文外項を対象に拡張したモデルを構築する．その際，文外では統語的な情報を取り入れることが難しくなるため，既存の研究で用いられたように外部資源を用いて述語同士の関係を考慮した共起関係などを取り入れることを検討したい．

参考文献

[1] M. Collins and T. Koo. Discriminative reranking for natural language parsing. *Computational Linguistics*, 31(1):25–70, 2005.

[2] R. Iida, K. Inui, and Y. Matsumoto. Zero-anaphora resolution by learning rich syntactic pattern features. *ACM Transactions on Asian Language Information Processing (TALIP)*, 6(4):1:1–1:22, 2007.

[3] R. Iida, M. Komachi, K. Inui, and Y. Matsumoto. Annotating a japanese text corpus with predicate-argument and coreference relations. In *Proceedings of the Linguistic Annotation Workshop (LAW '07)*, pp. 132–139, 2007.

[4] K. Imamura, K. Saito, and T. Izumi. Discriminative approach to predicate-argument structure analysis with zero-anaphora resolution. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing Conference (IJCNLP 2009)*, pp. 85–88, 2009.

[5] M. Nagata. A japanese morphological analysis method using a statistical language model and an n-best search algorithm. *Information Processing Society of Japan (IPSJ)*, 40(9):3420–3431, 1999. (In Japanese).

[6] R. Sasano, D. Kawahara, and S. Kurohashi. A fully-lexicalized probabilistic model for japanese zero anaphora resolution. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008)*, Vol. 1, pp. 769–776. Association for Computational Linguistics, 2008.

[7] H. Taira, S. Fujita, and M. Nagata. A japanese predicate argument structure analysis using decision lists. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '08)*, pp. 523–532, 2008.

[8] K. Toutanova, A. Haghighi, and C. D. Manning. A global joint model for semantic role labeling. *Computational Linguistics*, 34(2):161–191, June 2008.

[9] Y. Watanabe, M. Asahara, and Y. Matsumoto. A structured model for joint learning of argument roles and predicate senses. In *Proceedings of The 48th Annual Meeting of the ACL (ACL 2010)*, pp. 98–102, 2010.

[10] 吉川, 浅原, 松本. Markov logic による日本語述語項構造解析. 自然言語処理, 20(2):251–271, 2013.

[11] 松林, 乾. 統計的日本語述語項構造解析のための素性設計再考. 言語処理学会第 21 回年次大会予稿集 (ANLP 2014), pp. 360–363, 2014.

[12] 林部, 小町, 松本. 述語と項の位置関係ごとの候補比較による日本語述語項構造解析. 自然言語処理, 21(1):3–25, 2014.