単語単位の日本語係り受け解析

Daniel FLANNERY[1]　宮尾 祐介[2]　Graham NEUBIG[1]　森 信介[1]
[1] 京都大学 情報学研究科　[2] 国立情報学研究所

1 はじめに

係り受け解析は自然言語処理の基盤技術の 1 つであり、機械翻訳や情報抽出といった様々な応用がある。統計的手法に基づいた係り受け解析器は、一般分野のテキストにおける精度を徐々に上げている。一方で、一般分野のコーパスから学習された係り受け解析器が専門分野のテキストに対して精度が低下することも知られており、対策としては専門分野のコーパスにアノテーションし、学習データを用意することが挙げられる。

本研究は、学習データのアノテーション労力の軽減に着目し、点予測に基づく最大全域木による係り受け解析器を提案する。新しい係り受け解析手法と効率の良いアノテーション法を組み合わせることで分野適応のコスト削減を実現する。必要な部分のみをアノテーションする部分的アノテーションを使用するにより、精度向上につながる係り受けにアノテーション作業を集中することができる。

従来の最大全域木による係り受け解析器 [6] は、単語間の係り受けを推定する際に周辺の係り受け情報を用いる。しかし、この手法では部分的アノテーションコーパスを学習に使用できない。これに対して提案手法は、係り受け木のエッジスコアを個別に計算できることを仮定し、点予測を用いる。周辺の係り受け情報を参照しないことで、部分的アノテーションからの学習が可能になるという利点がある。また、実装が簡便になり、係り受け解析速度が向上するという長所もある。一方で、精度低下の懸念があるが、その問題は起こらないことを実験的に示す。

我々の係り受け解析のもう 1 つの特徴は、単位を文節ではなく、単語とすることである。日本語においては、文節を単位とする係り受け解析が一般的である。しかしながら、単語を単位とすることが他の主要言語では主流である。また、単語を単位とすることで、複合語内の構造や統語的複合動詞の各動詞の格要素など、文節単位の係り受けでは表せない情報を表すことができるという利点がある。また、機械翻訳や情報抽出などのタスクでは、単位として単語が適している場合が多く、単語単位の日本語係り受け解析器の需要がある。

このような背景から、単語を単位とする係り受けコーパスを構築し、点予測に基づく係り受け解析器を学習した。本論文では、単語を単位とする係り受け解析器について説明し、部分的アノテーションによる効率的な分野適応と、従来の文節単位の係り受け解析器との比較実験について述べる。

2 係り受け解析

本研究では、近年の係り受け解析の定式化 [2] を用いる。入力として単語列 $\boldsymbol{w} = \langle w_1, w_2, \ldots, w_n \rangle$ が与えられ、これに対して係り受け木 $\boldsymbol{d} = \langle d_1, d_2, \ldots, d_n \rangle$ を特定する。ここで $d_i \equiv j$ は単語 w_i の係り先が w_j であることを示す。文の係り受け木は根付き木であり、ある単語 w_i に対して $d_i = 0$ とする。

2.1 点予測による係り受け解析器

本論文は McDonald らのモデル [6] を採用している。全ての辺（単語間の係り受け）d_i にエッジスコア $\sigma(\langle i, d_i \rangle, \boldsymbol{w})$ を割り当て、エッジスコアの総和が最大となる係り受け木 $\hat{\boldsymbol{d}}$ を入力文に対する全ての可能な最大全域木の中から探索する。

$$\hat{\boldsymbol{d}} = \operatorname*{\mathbf{argmax}}_{\boldsymbol{d} \in D} \sum_{i=1}^{n} \sigma(\langle i, d_i \rangle, \boldsymbol{w}) \tag{1}$$

文中の全ての単語間の係り受けに $\sigma(\langle i, d_i \rangle, \boldsymbol{w})$ を割り当てておけば Chu-Liu/Edmonds 法などの最大全域木アルゴリズムで $\hat{\boldsymbol{d}}$ が算出できる。

提案手法では、エッジスコアの合計を文全体で最大化する McDonald ら [6] の手法と異なり、点予測によるスコアの推定を行う。すなわち、文中のそれぞれの w_i に対して $\sigma(\langle i, d_i \rangle, \boldsymbol{w})$ を個別に多クラス分類問題として推定する。

エッジスコアの推定には様々な機械学習手法が利用可能であるが、ここでは係り先の単語を同定する確率 $\sigma(\langle i, d_i \rangle, \boldsymbol{w}) = \log p(d_i = j | w)$ として定義する。これは以下のように対数線形モデル [1] で学習する。ある単語 w_i の係り先 d_i が w_j である確率を文脈情報 $x = \langle \boldsymbol{w}, \boldsymbol{t}, i \rangle$ から計算する。ここで、$\boldsymbol{t} = \langle t_1, t_2, \ldots, t_n \rangle$ は $\boldsymbol{w}$ の品詞タグ列であり、条件付き確率 $p(j|x)$ は下記の式となる。

$$p(j | x, \boldsymbol{\theta}) = \frac{\exp\left(\boldsymbol{\theta} \cdot \boldsymbol{\phi}(x, j)\right)}{\sum_{j' \in \mathcal{J}} \exp\left(\boldsymbol{\theta} \cdot \boldsymbol{\phi}(x, j')\right)} \tag{2}$$

素性ベクトル $\boldsymbol{\phi} = \langle \phi_1, \phi_2, \ldots, \phi_m \rangle$ は 2 単語の対 (x, j) に対する素性から計算される実数値からなる。また、$\boldsymbol{\theta} = \langle \theta_1, \theta_2, \ldots, \theta_m \rangle$ は各素性に対する重みベク

i	1	2	3	4	5	6	7	8	9	10
w_i	小遣い	を	全部	使	っ	て	しま	っ	た	。
t_i	名詞	助詞	名詞	動詞	語尾	助詞	動詞	語尾	助動詞	補助記号
F d_i	2	4	4	5	6	7	8	9	10	0
P d_i	–	4	–	–	–	–	–	–	–	–

図 1: 各アノテーション法 : **F** フルアノテーション、**P** 部分的アノテーション

トルである。単語係り受けが付与された文から $\boldsymbol{\theta}$ を学習する。確率 $p(d_i = j|w)$ は i、j と入力 $\boldsymbol{w}$、$\boldsymbol{t}$ のみで求められるため、各単語 w_i に対して、個別に計算できる。パラメータ推定時に $\hat{\boldsymbol{d}}$ を算出しなくてもよいため、学習のときは最大全域木アルゴリズムを利用しない。

2.2 最大全域木探索

本論文での係り受け解析の対象は、日本語書き言葉としている。日本語は主辞後置型言語であるので、解探索においては $d_n = 0$ および $d_i > i$ $(\forall i < n)$ を仮定する。この制約により、解探索は、一般の最大全域木アルゴリズムよりも簡単になる。すなわち、文中の各単語に対して最大スコアを持つ係り先を選択すれば良い。こうすれば係り受け関係が閉路を作ることはないため、Chu-Liu/Edmonds 法のような再帰的処理は不要である。この制約は英語や日本語話し言葉に当てはまらないが、これらを解析対象とする場合は、通常の最大全域木アルゴリズムを用いればよい。

なお、提案手法の係り受け解析は、多くの既存の日本語係り受け解析と異なり、交差する係り受けを扱うことができえる。

3 アノテーション戦略

アノテーション作業のコストがアノテーションの数に概ね比例すると仮定すると、アノテーション可能な箇所のうち精度向上につながる箇所を選択することが望ましい。アノテーション作業に伴う高いコストがこの手法の動機である。

3.1 部分的アノテーション

単語係り受けにおけるフルアノテーションとは、文中の全ての単語に対し係り先を付与することである。一方、部分的アノテーションは、文中の一部の単語にのみ係り先を付与することである。この方法では精度向上に貢献しないと推測される係り受けや、確信の持てない係り受けをアノテーションしないため、より効率的であると考えられる。各アノテーション法の例を図 1 に示す。

3.2 部分的アノテーションからの学習

2.1 節で述べたように、提案手法では各単語 w_i に対するエッジスコア $\sigma(\langle i, d_i\rangle, \boldsymbol{w})$ を個別に推定する。従って $\sigma(\langle i, d_i\rangle, \boldsymbol{w})$ を推定するのに w_i に対する正解の係り先のみが必要であり、それ以外の単語の係り受け情報は不要である。このため、部分的アノテーションコーパスから素性の重みベクトル $\boldsymbol{\theta}$ を学習することができる。学習データにある 2 単語 w_i、w_j が係り受け関係にあるという正解のアノテーションがあれば、$d_i = j$ を正例、$j' \neq j$ を満たす $d_i = j'$ を負例と見なすことで分類器を学習する。

日本語係り受け解析の場合、各係り受け関係 $d_i = j$ が $j > i$ を満たすため、$j' \neq j$ と $j' > i$ の条件を満たす $d_i = j'$ が負例となる。例えば、図 1 が示す部分的アノテーションから w_2「を」に対して $d_2 = 4$ を正例、$d_2 = 3, 5, \ldots, 9, 10$ を負例として用いる。

表 2: EHJ-test に対する係り受け精度

手法	係り受け精度
Malt	96.63%
MST	96.67%
PW MST	96.83%

4 評価

この節では、提案手法を用いた係り受け解析の実験とその結果について述べる。

4.1 実験設定

表 1 は実験に用いたコーパスの諸元である。全てのコーパスに単語境界とその単語間の係り受け関係を人手で付与した。各単語の品詞推定には KyTea[7] を使用した。一般分野コーパス EHJ-train、EHJ-test は両方とも日常会話のための辞書の例文 [3] からなる。分野適応実験では、日経新聞の記事からなる NKN-train と NKN-test を用いた。NKN-train を除き、全てのコーパスはフルアノテーションコーパスである。

4.2 従来法との比較

提案手法 (PW MST) の係り受け精度を評価するために、最大全域木系の McDonald ら [6] の手法 (MST) と Nivre ら [8] が提案した shift-reduce 法による手法 (Malt) との比較実験を行った。この二つの手法は提案手法と異なり、学習データに部分的アノテーションコーパスを利用できないため、フルアノテーションコーパスである EHJ-train を使った。テストコーパスに EHJ-test を用いた。実験結果を表 2 に示す。係り受け精度とは、全ての単語に対して、係り先を正しく同定できたものの割合である。Malt と MST の精度は

表 1: コーパスの諸元

ID	種類	用途	分数	単語数	文字数
EHJ-train	辞書例文	学習	11,700	145,925	197,941
EHJ-test	辞書例文	評価	1,300	16,348	22,207
NKN-train	新聞記事	PA プール	9,023	263,427	398,570
NKN-test	新聞記事	評価	1,002	29,038	43,695

NKN-train は部分的アノテーション (PA) のプールとして用いた。

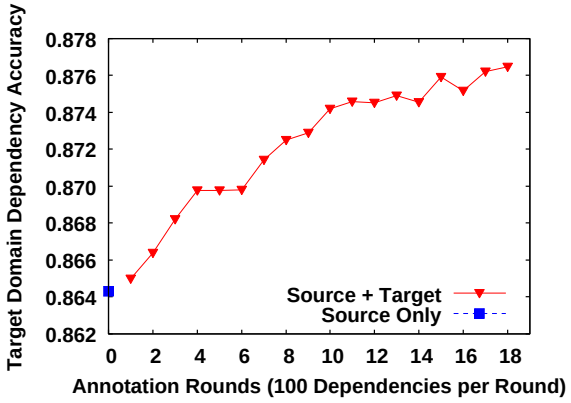

図 2: NKN-test に対する係り受け精度

ほぼ同等となったが、PW MST の精度がこれをわずかに上回った[1]。この結果から、周辺の係り受け情報を参照しないことによる精度低下はなく、柔軟に言語資源が活用できるという利点を実現している。

また、この実験で各手法の学習時間と解析速度も測った（表 3 参照）。その結果、MST は学習・解析ともに Malt より遅いことを確認した。提案手法は同じ最大全域木系の MST より学習時間が短く、shift-reduce 法に基づく Malt とほぼ同等の解析速度を実現していることがわかる。能動学習で分野適応を行う場合コーパスにアノテーションする度にモデルの再学習を行う必要があるため、提案手法の学習時間が短い点は重要である。

理論的には、提案手法の学習時間はアノテーションされた係り受けの数に比例する。2.2 節で述べたように、提案手法では学習時に最大全域木アルゴリズムを利用しない。MST と PW MST の学習時間の差はこの要因に起因すると考えられる。

4.3 分野適応

提案手法の長所の 1 つは、部分的アノテーションコーパスからの学習が可能な点である。この長所を確認するために、適応分野の部分的アノテーションコーパスを用いた分野適応実験を行った。一般分野コーパス EHJ-train に加え、適応分野コーパス NKN-train に 100 箇所のアノテーションする毎に係り受け解析器を学習し、NKN-test の精度を測った[2]。分野適応においては、名詞の直後の助詞の係り先が特徴的であると考え、ある名詞と助詞の連接 (例: 決算/名詞-を/助詞) が 10 回出現する毎に助詞の係り先を 1 回アノテーションすることとした。

図 2 に結果を示す。一般分野のコーパスのみを学習に用いた場合 (Source Only)、適応分野では精度が大きく低下することが分かる。しかし、適応分野の部分的アノテーション (Source+Target) も用いることで、精度が向上することが分かる。適応分野コーパスに対する 1,000 箇所のアノテーションで、約 1%の精度向上が確認された。このように、部分的アノテーションコーパスから学習できることは、分野適応において非常に有用であると考えられる。

4.4 文節単位の係り受け解析器との比較

日本語係り受け解析においては文節単位が多く利用されており、工藤ら [4] が提案したチャンキングの段階適用による係り受け解析手法は高精度を実現している。この手法をベースラインとして、提案手法の文節係り受けの精度を比較する実験を行った。

まず、この 2 つの手法を比較するのに提案手法が出力する単語係り受けを文節係り受けに変換する必要がある。この変換処理は以下の 2 段階からなる。

1. 単語を文節にまとめる。
2. 単語係り受けから文節係り受けを同定する。

まず、処理 1 は、文節係り受けの研究が行っているように、文節は 1 個以上の自立語と 0 個以上の付属語からなるとの原則に従い、表記や品詞を参照するルールにより行なった。図 1 に示す単語間の係り受け関係にこの変換処理をかけることで得た文節間の係り受けを図 3 に示す。

次に、処理 2 であるが、ある文節中の単語の係り先を先頭から見ていき、それががその文節の外になっている場合に、その係り先の単語を含む文節を注目文節の係り先とする。ただし、括弧などの例外がいくつかあり、ルールを記述することで対処した[3]。

チャンキングの段階適用による手法 (CC) として、工藤ら [4] の実装である「CaboCha/南瓜」[4]を用いた。まず、学習コーパス (EHJ-train) を上述の方法で文節係り受けに変換し、文節単位の係り受け解析モデルを学習した[5]。次に、文節列に変換したテストコーパ

[1] ただしこの差は有意ではない。
[2] 最大、1,800 箇所の単語間の係り受けをアノテーションした。
[3] 単語係り受けの基準では、開き括弧は対応する閉じ括弧に係る。
[4] `http://code.google.com/p/cabocha/` より入手可能 (2011 年 12 月現在)。
[5] 単語分割や品詞推定の影響を排除するため、CaboCha の単語分割や品詞推定の機能を利用しなかった。

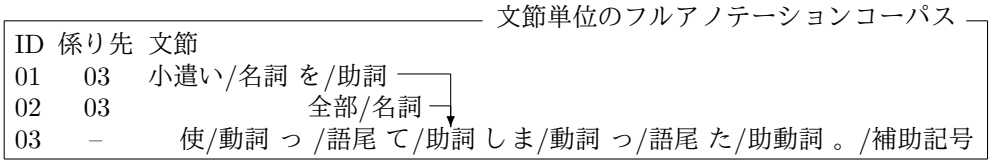

図 3: 単語係り受けを文節係り受けに変換する例

表 3: 学習時間と係り受け解析速度

手法	学習時間	解析速度
Malt	14[秒]	1.3[ミリ秒/文]
MST	1901[秒]	32.7[ミリ秒/文]
PW MST	125[秒]	2.8[ミリ秒/文]

学習は 3.33GHz プロセッサー、12GB のメモリーを搭載したマシン上で行った。

表 4: EHJ-test に対する文節係り受け精度

手法	係り受け精度
チャンキングの段階適用 (CC)	92.59%
点予測 + 最大全域木 (PW MST)	94.41%

ス (EHJ-test) を入力とし、文節間の係り受けを出力した。

提案手法 (PW MST) の場合は、学習コーパス (EHJ-train) から単語単位の係り受けのモデルを学習し (4.2 節と同じ)、EHJ-test の単語列を入力として単語関の係り受けを出力し、それを上述の方法で文節係り受けに変換した。

それぞれの結果は、文節係り受けに変換した EHJ-test と比較し、文節単位の係り受け精度を計算した。

両手法の文節係り受け精度を表 4 に示す。係り受け精度とは、文中の最後の文節を除く全ての文節に対して、その係り先の文節を正しく同定できたものの割合である。表から、PW MST は CC を上回っていることが分かる。数値の差は有意 ($p < 0.01$) であったが、PW MST が CC よりも優れいてると一概には言えないであろう。というのも、CC は複数階層からなる詳細な品詞体系を利用するように設計されていると考えられるが、この実験では品詞大分類のみを用いている。品詞細分類がないため、素性からの情報が乏しくなり CC の精度が下がったという可能性が否定できない。BCCWJ[5] には品詞細分類も付与されているが、本実験では利用しなかった。これは、実際の係り受け解析の分野適応では、形態素解析の分野適応も必要であるが、適応分野の知識に加えて品詞細分類を熟知している作業者を確保するのが困難であるとの経験からである。もう 1 つの可能性として、提案手法が単語係り受けを学習しており、結果として学習事例が多くなっているという要因が考えられる。

しかしながら、単語単位の係り受けの学習コーパスを所与とすれば、提案手法は文節単位の結果として既存手法と同等以上であるといえる。部分的アノテーションコーパスの利用可能性により分野適応が容易である点や、文節係り受けが表せない詳細な文構造を解明できる点を考慮に入れると、提案手法は非常に有効であると言える。

5 おわりに

本論文では、最大全域木のエッジスコアを点予測で計算する係り受け解析モデルを提案した。提案手法は、部分的アノテーションコーパスからの学習が可能となり、柔軟に言語資源を活用できる。評価実験では、提案手法が従来の最大全域木による係り受け解析手法とほぼ同等の精度となった。また、文節単位の係り受け解析器と比べた実験でも高い係り受け精度となることを確認した。

参考文献

[1] A. L. Berger, V. J. Della Pietra, and S. A. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

[2] S. Buchholz and E. Marsi. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pp. 149–164, 2006.

[3] D. Keene, H. Hatori, H. Yamada, and S. Irabu. *Japanese-English Sentence Equivalents (in Japanese)*. Asahi Press, Electronic book edition, 1992.

[4] T. Kudo and Y. Matsumoto. Japanese dependency analysis using cascaded chunking. In *Proceedings of the Sixth Conference on Computational Natural Language Learning*, Vol. 25, pp. 1–7, 2002.

[5] K. Maekawa. Balanced corpus of contemporary written Japanese. In *Proceedings of the 6th Workshop on Asian Language Resources*, pp. 101–102, 2008.

[6] R. McDonald, F. Pereira, K. Ribarov, and J. Hajič. Non-projective dependency parsing using spanning tree algorithms. In *Proc. of the 2005 EMNLP*, pp. 523–530, 2005.

[7] G. Neubig, Y. Nakata, and S. Mori. Pointwise prediction for robust, adaptable Japanese morphological analysis. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies Short Paper Track*, 2011.

[8] J. Nivre, J. Hall, and J. Nilsson. Maltparser: A data-driven parser-generator for dependency parsing. In *Proc. of the LREC06*, 2006.