トーナメントモデルを用いた日本語係り受け解析

岩立将和 †　　浅原正幸 †　　松本裕治 †

† 奈良先端科学技術大学院大学 情報科学研究科
email: {masakazu-i,masayu-a,matsu}@is.naist.jp

1　はじめに

決定的に解析を進めていく日本語係り受け解析アルゴリズムは、二文節 (係り元文節と係り先候補文節の組) が係る確率を用いたアルゴリズムに比べて時間計算量が少ないだけでなく、解析精度に関しても同等以上であることが知られている。

工藤ら [1] は、英語の構文解析で用いられているチャンキングの段階適用 (cascaded chunking) を日本語係り受け解析に適用した決定的アルゴリズムを提案している。このアルゴリズムは、係り元と係り先の距離が短い係り関係を優先して係けるアルゴリズムといえる。また、颯々野 [2] は Shift-Reduce 法に基づいた、係り先がより文頭に近く係り距離がより短い係り関係を優先して係ける決定的アルゴリズムを提案している。

これらのアルゴリズムでは、ある二文節が係るか否かを判定する際にその二文節の周辺の情報以外を利用するのが難しく、係り先候補間の相対的な係りやすさを考慮できないという問題がある。そこで我々は飯田ら [3] の提案したトーナメントモデルを係り受け解析に適用することで上記の問題を緩和した決定的日本語係り受け解析アルゴリズムを提案する。相対的な係りやすさを考慮した解析手法としては、工藤ら [4] の提案した確率モデルがあるが、我々の提案手法はさらにモデルとして候補間の相対的な距離を考慮できるため、より高精度の解析を行うことができると考えられる。

2　トーナメントモデル

トーナメントモデルは、飯田ら [3] が照応解析における最尤先行詞候補の選択に用いたモデルである。このモデルでは、ある照応詞の先行詞候補のうち二つを提示し、そのどちらがより先行詞らしいかを SVM などの二値分類器を用いて判断するという勝ち抜き戦を行っていくことで、最尤先行詞候補を選択する。

このトーナメントモデルを日本語係り受け解析に適用することを考えると、日本語には、文末の文節を除く各文節が右側（文末方向）にただ一つの係り先文節を持つという制約があるので、図 1 のように、係り元文節の右側に位置する複数の係り先候補から最尤候補を選択するという形になる。日本語の解析においては、係り関係が交差しないという制約（非交差制約）を仮定して解析を行うので、係り元文節の右側に位置する文節であっても、その文節に係けると非交差制約に反するような文節は係り先候補とは考えない。

図 1 の例では、「彼は」の係り先文節候補は三つあるのでまず「本を」と「読まない」を戦わせ、次に勝者となった「読まない」と「人だ。」を戦わせる。そして最終的に勝ち残った「人だ。」が「彼は」の最尤係り先候補に選ばれる。

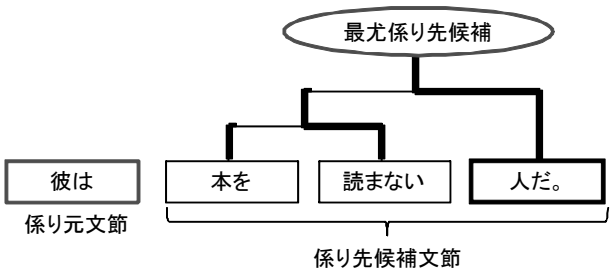

図 1: 日本語係り受け解析におけるトーナメントモデル

トーナメントモデルの一つの見方としては、全ての候補を考慮する解析手法を二値分類器を用いて実装したものであるといえる。だがそれだけでなく、二つの候補を対等と考えず、「係り元文節に近い方の候補、遠い方の候補」と区別することで、「係り元文節に近い候補ほど係りやすい」など、様々な傾向を学習できると考えられる。

従来の決定的解析アルゴリズム [1] [2] とトーナメントモデルの違いとして、候補間の相対的な位置の情報をアルゴリズムで表現するか、近い遠いの枠組みで表現するかという点があげられる。従来法では訓練事例生成と解析を同一のアルゴリズムとすることで、つまり依存構造木を特定の順序で構築することで、ほぼ同じ素性ベクトルを持つ訓練事例間でクラスラベルが異なるような矛盾した訓練事例を生じにくくしている。このような戦略をとると当然、全ての組み合わせを考慮するようなモデルと比べて訓練事例数は少なくなるが、訓練事例の質を高く保つことを重視していると考えられる。これに対してトーナメントモデルでは、近い方の候補と遠い方の候補という枠組みで学習しているため、訓練事例が両候補の依存構造木における相対的な位置の情報を保持する。従来法では矛盾していた事例群も、この情報があるために矛盾のない有用な事例群とすることができる場合があると考えられる。

また、工藤ら [4] は、ある係り元文節に対する各係り先候補の係りやすさの強さを数直線上に並べ、その大小で最尤候補を選択している。つまり、候補間の係りやすさの強さに推移律が成り立つという仮定を置いている。これに対してトーナメントモデルでは、各二候補間について、どちらの候補がより係り先としてふさわしいかを学習するので、推移律は仮定しておらず工藤らの手法より仮定が緩い。現実には二つ以上の候補の関係によって決まることは珍しくなく、仮定が緩いトーナメントモデ

ルの方がよい近似ができると考えられる。

3 提案アルゴリズム

トーナメントモデルを用いて訓練事例生成と解析を行う方法を述べる。従来法では両者に同一のアルゴリズムを用いているが、提案手法では別々のアルゴリズムを用いる。訓練事例生成アルゴリズムと解析アルゴリズムのどちらも、時間計算量の上限は入力文の文節数に対して $O(n^2)$ である。

3.1 訓練事例生成アルゴリズム

図 2 に示すように、各係り元文節について、正解係り先文節と他の全ての係り先候補との組について訓練事例を生成する。訓練事例生成の際には非交差制約は考慮せず、係り元文節より右側に位置するすべての文節を係り先候補とみなす。

```
// N: 入力文の文節数
// true_head[j]: 文節 j の訓練データでの係り先
// gen(j,i1,i2,LEFT): 文節 j が i1 に係るという
//                          訓練事例を生成する
// gen(j,i1,i2,RIGHT): 文節 j が i2 に係るという
//                          訓練事例を生成する
for j = 1 to N-1 do
  h = true_head[j];
  for i = 2 to h-1 do gen(j,i,h,RIGHT);
  for i = h+1 to N do gen(j,h,i,LEFT);
end-for;
```

図 2: 訓練事例生成アルゴリズム

3.2 解析アルゴリズム

トーナメントモデルでは訓練事例生成と解析に別々のアルゴリズムを用いるために、モデルが解析アルゴリズムを規定するということがない。そのため、解析順としてはいくつか考えられるが、ここでは文末の文節から文頭に向かって係り先文節を同定していき、ある文節の係り先の同定は文末方向に行うアルゴリズムについて説明する。この解析順が最もアルゴリズムが単純になる。

アルゴリズムを図 3 に示す。この解析順の場合、係り元より右側は完全に解析済みなので、最も近い候補である係り元の右隣の文節から始めて係り関係をたどっていけば、非交差制約に違反しない候補のみをたどることができる。実際、head[] は各文節の係り先文節番号が格納される配列であるが、解析中には非交差制約に違反しない係り先候補をつないだ線形リストとしての役割も果たしている。

```
// N: 入力文の文節数
// head[]: 文節 j の係り先
// classify(j,i1,i2): 文節 j が i1 と i2 の
//      どちらに係りそうかを SVM で分類し、
//      前者なら LEFT、後者なら RIGHT を返す

head[] = {2,3,...,N-1,N,EOS};

for j = N-1 downto 1 do
  h = j+1;
  i = head[h];

  while i != EOS do
    if classify(j,h,i)==RIGHT then h = i;
    i = head[i];
  end-while;

  head[j] = h;
end-for;
```

図 3: 解析アルゴリズム

4 実験

4.1 実験設定

トーナメントモデルおよび従来法を実装し、京大コーパス Version 4.0 を使用して、京大コーパス Version 2 と同じ条件で係り受け正解率と文正解率を評価した。具体的には、1 月 1 日から 1 月 8 日までの記事 (7587 文) を訓練データとし、1 月 9 日の記事 (1213 文) をテストデータとした。ただし、1 月 9 日分だけでなく、1 月 10 日分 (1479 文) や 1 月 15 日分 (1179 文) をテストデータとした実験も行った。なお、括弧内は長さが二文節以上の文の数である。また、係り受け正解率は文末の一文節 (係り先を持たない) を除いて評価した。

二値分類器としては TinySVM を使用し、三次の多項式カーネルを用いた。誤分類のコストは 1 とした。なお、実験は全て Dual Core Xeon 3.0GHz x 2 の Linux 上で行った。

4.2 使用した素性

以下、係り元文節と係り先候補文節をそれぞれ係り元、候補と略記する。なお、トーナメントモデルでは二つの候補を同時に検討するので、候補に関する素性は、近い方の候補と遠い方の候補に関してそれぞれ作成する。また、文節の「情報」とは、その文節の主辞と語形それぞれの表層形・品詞・品詞細分類・活用形および句読点・開き括弧・閉じ括弧の有無、および文頭の文節か否か・文末の文節か否かの素性のこととする。主辞とは文節の形態素のうち品詞が特殊・助詞・接尾辞以外の形態素のうち最も右側のもの、語形とは品詞が特殊以外の形態素のうち最も右側のものである。

表 1: 各テストデータに対する係り受け正解率/文正解率 [%]

方式	素性セット	1 月 9 日分	1 月 10 日分	1 月 15 日分
トーナメント	標準	89.89/49.63	89.63/48.34	89.40/49.70
	標準+追加+格助詞	**90.09/49.71**	**90.11/49.02**	**90.35/52.59**
颯々野 [2]	標準	88.18/45.92	88.80/44.76	88.03/47.24
	標準+追加+格助詞	89.22/47.90	89.79/47.87	89.55/49.79
工藤ら [1]	標準	88.17/45.92	88.80/44.76	88.00/47.24
	標準+追加+格助詞	89.22/47.90	89.80/47.94	89.53/49.79

標準素性と追加素性は、颯々野 [2] の用いたものを参考にした素性である。標準素性とは、係り元と候補の情報および、係り元と候補の間の距離 (1、2-5、6 以上)、句読点・開き括弧と閉じ括弧の有無、すべての助詞である。追加素性とは、係り元と候補のすべての格助詞、候補文節の最左形態素の情報および、候補の右隣の文節の表層形 (すべての形態素の表層形の連結) である。

格助詞素性とは、候補に係っている全ての文節の格助詞の表層形および、係り元文節と候補に係っている文節に共通して含まれる格助詞があるか (yes/no) と共通して含まれる格助詞の表層形である。ただし、「候補に係っている文節」とは、係り元より右側にあるものに限る。これはアルゴリズムの性質上係り元より左の係り関係を考慮できないためである。この素性は、「ある候補にヲ格がすでに係っているなら、さらに別のヲ格が係ることはない」というような制約を学習させることを意図している。

4.3 解析精度

各方式・素性セットについての精度に関する実験結果を表 1 に示す。

京大コーパス Version 2 の 1 月 9 日分の記事をテストデータとした実験のうちこれまで報告されているものの中で最も高い精度としては、颯々野 [2] が標準素性、追加素性相当の素性に加えて、係り元の左隣の文節や候補の両隣の文節の情報、並列句情報を素性として用いたときの係り受け正解率 89.56%であるが、トーナメントモデルの 1 月 9 日分に対する精度はこれを上回っている。

また、標準素性のみを用いた場合と標準+追加+格助詞素性の場合の精度向上幅に注目すると、トーナメントモデルは従来法より向上幅が小さい。この理由の一つとしては、元の精度が高いほど大幅な精度向上をしにくいということが考えられる。ただ、標準素性がほぼ係り元と注目している候補の情報のみからなる素性であることを考えると、従来法よりもトーナメントモデルの方がアルゴリズムとして周辺の情報を考慮できているということを示しているとも解釈できる。

なおこの結果からは、同一の素性を用いた場合、颯々野 [2] と工藤ら [1] の方式の精度がほぼ同じとなることがわかる。ただ、両方式は解析順が異なり、ゆえに利用できる動的素性 (解析済みの係り関係に関する素性) も異なることから、それぞれに有効な動的素性を入れていって到達できる最高精度には差が生まれると思われる。

4.4 解析速度および訓練事例の規模

素性セットは標準＋追加＋格助詞とし、1 月 9 日の記事をテストデータとした場合の解析時間等について実験した。その結果を表 2 に示す。訓練データは 1 月 1 日から 1 月 8 日までの記事である。

表 2: 各方式における解析ステップ数、解析時間 [秒]、訓練事例数、素性次元数

方式	Step 数	時間	事例数	素性数
トーナメント	26396	371	374579	56165
颯々野 [2]	15641	80	94669	37183
工藤ら [1]	18922	99	112759	37183

トーナメントモデルと工藤ら [1] の方式の時間計算量は $O(n^2)$、颯々野 [2] の方式は $O(n)$ であるから、解析ステップ数 (SVM classify の呼び出し回数) は颯々野の方式が最も少なくなっている。トーナメントモデルのステップ数は颯々野の方式の約 1.7 倍となっている。また、解析時間は 4 倍以上かかっているが、これはトーナメントモデルは訓練事例数・素性次元数ともに颯々野の方式よりも多く、SVM のモデルが大きくなるからである。

4.5 決定的訓練事例生成

既存の決定的アルゴリズムは、訓練事例と解析に同一のアルゴリズムを用いることで状況を限定し、高精度を達成していると考えられる。そこで、トーナメントモデルにおいてもこのように決定的に訓練事例を生成することを考える。

トーナメントモデルにおける決定的訓練事例生成とは、単に非交差制約を考慮して訓練事例を生成することを指すものとする。解析時と同じようにトーナメントの試合の流れをシミュレートするわけではない。

素性セットは標準＋追加＋格助詞とし、通常の生成法（全事例生成）と決定的生成法を比較した結果を表 3 に示す。なお、訓練事例数は全事例で 374579、決定的で 153503 と、決定的生成をすることで半数以下になった。

結果によると、単純に訓練事例数の多い全事例生成の方が解析精度が高いことから、トーナメントモデルの訓練事例は精度低下につながるような矛盾する訓練事例を

表 3: 訓練事例生成法による精度差

方式	1 月 9 日分	1 月 10 日分	1 月 15 日分
全事例	90.09/49.71	90.11/49.02	90.35/52.59
決定的	89.65/48.56	89.90/48.55	90.07/51.57

あまり含まず、比較的事例の質が良いと推測される。ただ、全事例生成は決定的生成に比べて訓練事例数が 2 倍なのに対して、精度の向上は 0.2-0.3%と若干小さいようにも思える。したがってこの結果は、決定的生成によって訓練事例の質はある程度上がったものの、訓練事例数の差を覆すほどではなかったと解釈できる。

4.6 解析アルゴリズムの選択

トーナメントモデルではモデルが解析アルゴリズムを規定しないので、図 1 のようなパラマストーナメントに限らず他の方式を採用することもできる。

ここでは、図 1 や 3.2 節で説明したような文末方向にパラマストーナメントを行う方式と、文頭方向にパラマストーナメントを行う方式について、素性セットは標準＋追加＋格助詞として実験した。その結果を表 4 に示した。両アルゴリズムには大差はないように思える。

表 4: 解析アルゴリズムの選択による精度差

方式	1 月 9 日分	1 月 10 日分	1 月 15 日分
文末へ	90.09/49.71	90.11/49.02	90.35/52.59
文頭へ	90.11/49.63	90.19/49.43	90.42/52.67

トーナメントモデルの解析アルゴリズムは、全ての候補の総当たり戦を行う代わりに一部の試合を行って最尤候補を選択するアルゴリズムであるから、正解と紛らわしい候補があるときに、正しい候補を選択するための決め手となる試合を確実に行うことが重要である。もし、どれが決め手となる試合なのかがあらかじめある程度分かれば、前処理としてトーナメントの組み方をヒューリスティックに決めるという方法も考えられる。

なお、文頭の文節から係り先を同定していくなど、大幅に解析アルゴリズムを変更することも考えられるが、その際には、格助詞素性のような動的素性は同定順依存の素性であること、SVM の呼び出し回数が $O(n^2)$ であっても非交差制約を管理する部分のコードが $O(n^2)$ で書けるとは限らないことに注意する必要がある。

5 今後の課題

さらなる精度向上のためには、並列句と共起の情報の導入が有効であると考えている。颯々野 [2] によると、並列句の情報を並列句素性として入れることで精度が向上したということなので、並列句の情報を素性として、あるいは別の形で入れることを検討したい。また、Web 等の大量のラベルなしデータから共起情報を獲得し利用したい。係り受け解析における共起というのは、共起の強さ=係りやすさと考えると素性というより訓練データに相当するように思えるが、どのように利用するかは難しい問題である。係り元との共起の強さが正解候補と同程度ならば紛らわしい候補であると考えるなら、前述したようなトーナメントの組み方の決定に利用できるかもしれない。

日本語以外の言語への対応も検討したい課題である。日本語では必ず係り先が係り元の右側に位置しているが、左右どちらに位置することもありうる言語の場合、トーナメントモデルをそのまま使用することはできない。もちろん、左側に位置する候補同士の係りやすさと右側に位置する候補同士の係りやすさを別々に学習するようなことは現時点でのトーナメントモデルでも可能であるが、左側の候補と右側の候補の間の相対的な係りやすさをどう表現するかは工夫が必要である。

6 おわりに

本稿では、トーナメントモデルを用いた日本語係り受け解析手法を提案した。

トーナメントモデルにおいては、二つの候補を対等に扱うのではなく係り元に近い方の候補、遠い方の候補と区別することで候補間の相対的な位置を考慮した解析が行える。また、ある係り元文節に対する候補間の相対的な係りやすさの表現法として二つの候補の組についてどちらがより正解らしいかを学習するため、候補間の相対的な係りやすさに推移律を要求しない、より現実に則した解析が可能である。これらの特長があるため、訓練事例の質が比較的高いと考えられる。

解析精度は従来法を上回るものであり、解析速度の問題はあるものの、同時に複数の候補について検討することでより高精度の解析ができる可能性があることを示している。

参考文献

[1] 工藤拓, 松本裕治. チャンキングの段階適用による日本語係り受け解析. 情報処理学会論文誌, Vol. 43, No. 6, pp. 1834–1842, 2002.

[2] 颯々野学. 日本語係り受け解析の線形時間アルゴリズム. 自然言語処理, Vol. 14, No. 1, pp. 3–18, 2007.

[3] 飯田龍, 高村大也, 乾健太郎, 松本裕治. 機械学習によるゼロ代名詞同定の一方法. 情報処理学会研究報告 2003-NL-154, pp. 161–168, 2003.

[4] 工藤拓, 松本裕治. 相対的な係りやすさを考慮した日本語係り受け解析モデル. 情報処理学会研究報告 2004-NL-162, pp. 205–212, 2004.