決定リストを用いた形容詞の修飾先の決定

白井 清昭　橋本 泰一　西館 耕介　徳永 健伸　田中 穂積

東京工業大学 大学院情報理工学研究科

{kshirai,taiichi,nishidy,take,tanaka}@cl.cs.titech.ac.jp

1 はじめに

1.1 背景

本研究では,「N_1 の Adj N_2」(但し，N_i は名詞，Adj は形容詞) という構文における Adj の修飾先を決定する手法を提案する．例文を以下に示す．

1. 信濃 の 美しい 川
2. 水面 の 美しい 川

例文 1 では,「美しい」は後の名詞「川」を修飾し，例文 2 では,「美しい」は前の名詞「水面」を修飾する．このように,「N_1 の Adj N_2」という構文において，Adj の修飾先は曖昧である．Adj の修飾先を正確に決定しなければならない場面は多い．例えば，機械翻訳で例文 1 と 2 を訳し分けることを考えれば，Adj の修飾先を決定することの意義は明らかであろう．

1.2 関連研究

田中らは,「N_1 の Adj N_2」という構文の Adj の修飾先は 2 つの名詞 N_1 と N_2 の意味的依存関係と関連があると述べ，Adj の修飾先を決定する「名詞関係依存の原則」を提案した [7]．その概略は以下の通りである．

原則 I もし，N_1+の+N_2 が成立すれば，Adj は N_2 を修飾する．

原則 II もし，N_2+の+N_1 が成立すれば，Adj は N_1 を修飾する．

原則 III 原則 I と原則 II が共に成立する場合には，Adj と N_1, N_2 との間の結合可能性をさらに調べる．

名詞関係依存の原則を 1.1 項の例文 1,2 に適用してみよう．例文 1 では「信濃の川」が文として成立するので，原則 I より Adj の修飾先は N_2(川) となる．一方，例文 2 では「川の水面」が文として成立するので，原則 II より Adj の修飾先は N_1(水面) となる．また，田中らは，名詞関係依存の原則は「N_1 の Adj N_2」という構文だけでなく,「N_1 の $AdjV$ N_2」(但し，$AdjV$ は形容動詞) という構文にも同様に適用できるとしている．

橋本らは，名詞関係依存の原則に基づき，コーパスから得られる統計情報を用いて Adj の修飾先を決定する手法を提案した [2]．その手法を簡単に紹介する．まず,「名詞+の+名詞」というパタンの共起頻度をコーパスから獲得する．次に,「N_1+の+N_2」と「N_2+の+N_1」の共起頻度を比較し，その大きい方が成立すると判断し，原則 I または II により Adj の修飾先を決定する．橋本らの手法では，N_1, N_2 のみから Adj の修飾先を決定し，Adj そのものを考慮していない．つまり，名詞関係依存の原則の原則 III については考慮していない．しかし，Adj の修飾先の決定に Adj そのものの情報を無視するのは望ましくない．むしろ N_1 や N_2 の情報と組み合わせて用いるべきである．

菊池らは，約 4400 個の実例文を人手で分析し,「N_1 の $Adj(AdjV)$ N_2」という構文の Adj または $AdjV$ の修飾先を決定する 7 つの規則を提案した [3]．これら 7 つの規則は，Adj とは独立に N_1, N_2 のみで決定する規則，Adj と N_1, Adj と N_2 の関係で決定する規則，Adj のみで決定する規則に分類され，橋本らが考慮しなかった Adj の情報も考慮されている．しかし，大量の実例文を調査してはいるものの，これらの規則は人手で作成されている．そのため，規則が対象とする形容詞の数は 21 と少ない．Adj の修飾先を決定する規則を網羅的に人手で記述することには限界があるので，これらの規則はコーパスなどから自動的に学習できることが望ましい．

1.3 目的

本研究では,「N_1 の Adj N_2」という構文内の形容詞の修飾先を決定するシステムをコーパスから自動的に学習することを目的とする．学習アルゴリズムとしては決定リスト [4] を用いる．決定リストは，自然言語処理分野においても，アクセント記号復元 [10]，語義曖昧性解消 [11]，スペルミス検出 [6]，固有表現抽出 [5, 9]，文節の係り受け解析 [8] などに応用され，比較的良い成果を挙げている．

本研究のもうひとつの目的は，形容詞の修飾先の決定に有効な素性を調査することである．特に，1.2 項で述べた名詞関係依存の原則は，Adj の修飾先を決定する際，Adj そのものよりも先に前後の名詞を参照するという点で興味深い．したがって，前後の名詞のみを用いた判定がどの程度有効であるかを実験的に検証したい．学習アルゴリズムとして決定リストを用いる理由の一つは，決定リストは優先順位付きの規則の集合であるため，学習結果を容易に考察できる点にある．

表 1: 規則のテンプレート

規則 ($C_i \rightarrow d_i$)			タイプ
Adj=x	$\rightarrow$	$\{N_1,N_2\}$	A
N_1=x	$\rightarrow$	$\{N_1,N_2\}$	N1
N_2=x	$\rightarrow$	$\{N_1,N_2\}$	N2
Adj=x & N_1=y	$\rightarrow$	$\{N_1,N_2\}$	A+N1
Adj=x & N_2=y	$\rightarrow$	$\{N_1,N_2\}$	A+N2
N_1=x & N_2=y	$\rightarrow$	$\{N_1,N_2\}$	N1+N2

2 決定リストの学習

本節では，形容詞の修飾先を決定する決定リストの学習アルゴリズムについて述べる．なお，ここで述べるアルゴリズムは，2.2.2 で述べる手法を除いて，Yarowsky の手法 [11] とほぼ同じである．

2.1 規則の候補の生成

訓練データは「N_1 の Adj N_2」という構文の集合である．これは，コーパスから品詞のパタンマッチにより自動的に抽出する．但し，名詞の連続は 1 つの複合名詞とみなして N_1 または N_2 にマッチさせる．また，各構文における Adj の正しい修飾先も与えられているものとする．

次に，訓練データから規則の候補を抽出する．規則のテンプレートとして，表 1 に示す 6 種類を用いた．表 1 における規則 $C_i \rightarrow d_i$ は，条件 C_i を満たすとき，Adj の修飾先を d_i に決定することを表わす．また,「タイプ」は，条件 C_i で参照される単語の種類による規則の分類を表わす．

例えば,「話題:満載 の 若い 挑戦:者」という例文からは，以下に示す 6 つの規則の候補を生成する．

- Adj=若い $\rightarrow N_2$
- Adj=若い & N_1=満載 $\rightarrow N_2$
- N_1=満載 $\rightarrow N_2$
- Adj=若い & N_2=者 $\rightarrow N_2$
- N_2=者 $\rightarrow N_2$
- N_1=満載 & N_2=者 $\rightarrow N_2$

「話題:満載」や「挑戦:者」は，名詞連続を 1 つにまとめた複合名詞を表わす．規則の候補数の増大を妨げるために，N_1 や N_2 を参照する規則の候補を生成する際には，複合名詞全体を参照せず，複合名詞の一番最後に現われる名詞のみを参照する．

また，式 (1) のデフォルト規則を導入する．

$$true \quad \rightarrow \quad N_2 \tag{1}$$

これは入力文に対して常に適用される規則である．3.1 項で述べる訓練データでは，Adj が N_2 を修飾する構文が N_1 を修飾する構文よりも多かったため，デフォルト規則の修飾先も N_2 とした．

2.2 決定リスト作成

訓練データから生成された規則に対して，これらの規則の適用順序を決め，決定リストを作成する．規則の適用順序の決め方は 2 つある．ひとつは規則の尤度に基づく手法，もうひとつは名詞関係依存の原則に基づく手法である．

2.2.1 尤度に基づく決定リスト

規則 $r_i(C_i \rightarrow d_i)$ の尤度 $L(r_i)$ を式 (2) のように定義する．

$$L(r_i) = \log \frac{P(d_i \mid C_i)}{P(\overline{d_i} \mid C_i)} \tag{2}$$

ここで，$P(d_i \mid C_i)$ は，規則の条件 C_i が成立するときに Adj の修飾先が d_i となる確率であり，$P(\overline{d_i} \mid C_i)$ はその排反事象の確率である．これは式 (3) で推定する．

$$P(d_i \mid C_i) = \frac{O(C_i, d_i) + \alpha}{O(C_i) + \alpha} \tag{3}$$

式 (3) において，$O(C_i)$ は条件 C_i を満たす構文の出現頻度であり，$O(C_i, d_i)$ は条件 C_i を満たしかつ Adj の修飾先が d_i である構文の出現頻度である．また，α は平滑化のためのパラメタである．本手法では $\alpha = 0.5$ とする．式 (2),(3) が示すように，条件 C_i を満たす構文のうち，Adj の修飾先が d_i である構文の割合が大きければ大きいほど，$L(r_i)$ の値も大きくなる．すなわち，$L(r_i)$ は，規則 r_i の確からしさを表わす尺度と考えられる．したがって，規則の候補を $L(r_i)$ の大きい順に並べて規則の適用順序を決定する．

さらに，決定リストに含まれる規則の数を削減する．まず，尤度が式 (1) のデフォルト規則よりも小さい規則は決定リストから除去する．つまり，決定リストの一番最後にある規則はデフォルト規則である．また，冗長な規則も決定リストから除去する．冗長な規則とは，決定リストの順に規則を適用した場合，決して使われることのない規則を指す．例えば，2 つの規則 r_a, r_b が以下の順序で並んでいるとする．

r_a: Adj=若い $\rightarrow N_1$

r_b: Adj=若い & N_2=者 $\rightarrow N_2$

このとき，r_b を適用できる入力文は，必ず r_a も適用可能であるため，r_b は決して使われることのない冗長な規則である．

このように作成された決定リストを dl-L とする．

2.2.2 名詞関係依存の原則に基づく決定リスト

1.2 項で述べた名詞関係依存の原則を本研究で提案する決定リストの枠組に当てはめると，まず最初にタイプが N1+N2 の規則を適用し (原則 I,II)，次に A+N1,

A+N2 の規則を適用する (原則 III) ことに相当する．そこで，規則のタイプによって，以下の順序で規則を並べた決定リスト dl-T を作成した．

$$\text{N1+N2} > \text{A+N2} > \text{A+N1} > \text{N1} > \text{A} > \text{N2}$$

同じタイプの規則の順序は，式 (2) の尤度 $L(r_i)$ によって決定した．また，デフォルト規則より尤度の低い規則は決定リストから除去した．なお，dl-L では冗長な規則を削除したが，dl-T では単語を 2 つ参照する規則が必ず先に適用されるので，冗長な規則は存在しない．

2.3 ブートストラップ法

これまで述べた手法では，訓練データとして *Adj* の正しい修飾先が付与された例文を使用する．しかし，正しい修飾先を人手で付与するコストを考えると，大量の訓練データを確保することは難しい．そこで，正解が付与された例文と正解未付与の例文の両方から決定リストを学習する．その手順は以下の通りである．

1. 正解付きの例文から決定リスト dl_1 を学習する．
2. 以下の操作を繰り返す．
   (a) dl_i を用いて，正解未付与の例文の *Adj* の修飾先を決める．
   (b) 正解付きの例文と，dl_i によって *Adj* の修飾先を決めた例文から，決定リスト dl_{i+1} を再学習する．
3. 2(a) の手続きで，正解未付与の例文の *Adj* の修飾先に変化がなければ，学習を終了する．

3 評価実験

3.1 学習

毎日新聞 1991 年から 1995 年の新聞記事に対して品詞タグを自動的に付与した RWC コーパス [1] から，「N_1 の *Adj* N_2」という構文を 31,541 個取り出した．このうち，479 文を評価データ，3,634 文を正解が付与された訓練データ，27,428 文を正解未付与の訓練データとして使用した．評価データと正解付き訓練データについては，*Adj* の正しい修飾先を人手で付与した．次に，訓練データから，2 つの決定リスト dl-L(2.2.1 参照) と dl-T(2.2.2 参照) を学習した．正解未付与の訓練データを用いた学習では，dl-L については 5 回，dl-T については 2 回で，正解未付与の例文に与える *Adj* の修飾先に変化がなくなった．

3.2 実験結果と考察

表 2 は，決定リストを評価データに適用したときの正解率と規則適用率である．正解率は，決定リストによって *Adj* の修飾先を正しく決定できた構文の割合，

表 2: 実験結果

(a) 正解率	BL	dl-L	dl-T
正解付与データのみ	52.6%	94.77%	94.35%
正解未付与データも使用	52.6%	94.78%	93.95%

(b) 規則適用率	BL	dl-L	dl-T
正解付与データのみ	100%	99.79%	99.79%
正解未付与データも使用	100%	100%	100%

規則適用率は，デフォルト規則以外の規則によって修飾先を決定できた構文の割合である．BL(ベースライン) は，*Adj* の修飾先を常に N_2 とする手法を表わす．dl-L, dl-T ともに正解率は約 94%となり，ベースラインを大きく上回った．また，規則適用率はほぼ 100%であった．一方，正解付与データのみから学習した決定リストと，正解未付与データに対して反復学習した決定リストを比較すると，ほとんど差は見られなかった．

この結果を先行研究と比較する．正解率は，橋本らの手法 [2] の 92%を上回るが，菊池らの手法 [3] の 97%には及ばなかった．一方，規則適用率は，これら 2 つの手法をはるかに上回る．橋本らの手法では，前後の名詞の組のみを素性として用いたため，修飾先を決定できる構文の割合は 61%程度である．また，菊池らの手法が 21 語の形容詞のみを対象としたのに対し，dl-L 中のタイプ A の規則で参照される形容詞の異なり数は 536 であり，これらの形容詞を含む構文については修飾先を必ず決定できる．したがって，学習された決定リストは，様々な形容詞を含む構文に対応できるという意味で適用範囲が広い．

図 1 の棒グラフは，正解未付与の訓練データも用いて学習した dl-L, dl-T において，学習された規則の異なり数を規則のタイプ別に示したものである．また，図 1 の折れ線グラフは，決定リストを評価データに適用した際に，実際に使用された規則ののべ数を表わす．一方，図 2 は，規則のタイプ別にみた正解率である．

まず，dl-L について考察する．dl-L においては，タイプ N1,N2 の規則が多く学習されたことがわかる．これは，単語を 1 つ参照する規則の尤度が単語を 2 つ参照する規則の尤度よりも全般的に大きく，単語を 2 つ参照する規則の多くが冗長な規則として削除されたためである．実際，尤度による順位付けの上位 100 位までの規則を調べてみると，条件部で単語を 1 つ参照する規則 (タイプ A,N1,N2) の割合は 81%であった．ちなみに，冗長な規則を削除する前の dl-L の規則数は，dl-T の規則数と一致する．

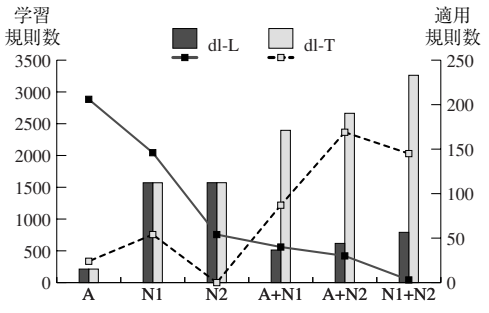

図 1: 規則数

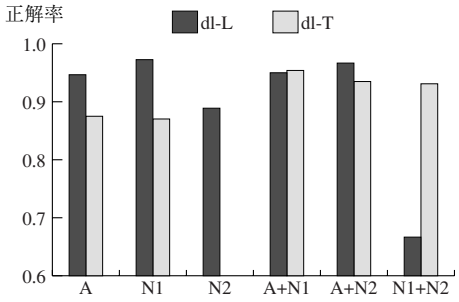

図 2: 規則のタイプ毎の正解率

田中らが提案した名詞関係依存の原則は，タイプ N1+N2 の規則が優先的に学習されることを予想している．また，筆者らは，形容詞と前後の名詞の意味的関係が形容詞の修飾先の決定に重要な役割を果し，タイプ A+N1 や A+N2 の規則が優先的に学習されると予想した．しかし，これらの予想とは逆に，単語を 1 つ参照する規則が優先的に学習される結果となった．実際に適用された回数を調べてみても，単語を 2 つ参照する規則に比べて，単語を 1 つ参照する規則が多く使われている．特に，タイプ A の規則は，学習された規則の異なり数が 6 つのタイプの中で最も少ないのにも関わらず，実際に適用された回数が一番多い．しかも，図 2 に示した通り，タイプ A の規則の正解率は 95%と高い．つまり，形容詞の修飾先は，周囲の単語に関係なく，形容詞のみだけでもかなり正確に決定できると言える．

次に，dl-T について考察する．dl-T は，名詞関係依存の原則にしたがって規則の適用順序を決めた決定リストである．その正解率は dl-L とほぼ等しい．また，タイプ N1+N2 の規則も多く使われ，その正解率は約 93%である．このことから，dl-L で学習された規則の適用順序とは異なるものの，名詞関係依存の原則にしたがって形容詞の修飾先を決定することも有効であることが実験的に確かめられた．

4 おわりに

決定リストを学習する際，名詞の異なり数は形容詞に比べて圧倒的に多いので，規則の尤度の計算に必要な訓練データが十分に得られないことが多い．したがって，今後は，意味クラスを用いて名詞を抽象化し，決定リストを学習することを考えている．名詞の意味クラスの候補が複数あるときに，これらを全て規則に展開するナイーブな手法を試したが，意味クラスを導入する前と比べて正解率は下がった．これは，訓練文においては正しくない意味クラスからも規則の候補を生成することが原因であり，決定リスト作成の際にも名詞の多義性解消を行う必要があると筆者らは考えている．決定リストの作成に意味クラスを有効に利用する方法を今後検討していきたい．

参考文献

[1] Koiti Hasida et al. The RWC text databases. In *Proceedings of the LREC*, pp. 457–462, 1998.

[2] 橋本泰一, 白井清昭, 徳永健伸, 田中穂積. 統計的手法に基づく形容詞または形容動詞の修飾先の決定. 情報処理学会情報処理学会自然言語処理研究会, Vol. 2000, No. 65, pp. 87–94, 2000.

[3] 菊池浩三, 伊東幸宏. 連体形イ・ナ形容詞に先行する格助詞句の係りに関するルールの抽出. 自然言語処理, Vol. 6, No. 3, pp. 75–99, 1999.

[4] Ronald L. Rivest. Learning decision lists. *Machine Learning*, Vol. 2, pp. 229–246, 1987.

[5] 颯々野学, 宇津呂武仁. 統計的日本語固有表現抽出における固有表現まとめ上げ手法とその評価. 情報処理学会自然言語処理研究会, Vol. 2000, No. 86, pp. 1–8, 2000.

[6] Hiroyuki Shinnou. Detection of Japanese homophone errors by a decision list including a written word as a default evidence. In *Proceedings of the EACL*, pp. 180–187, 1999.

[7] 田中穂積, 荻野孝野. 形容詞もしくは形容動詞の修飾先の名詞を決める原則について. 計量国語学, Vol. 12, No. 5, pp. 191–203, 1980.

[8] 梅村洋之, 原田義久, 清水司, 杉本軍司. 音声合成におけるポーズ制御のための決定リストを用いた局所係り受け解析. 自然言語処理, Vol. 7, No. 5, pp. 51–70, 2000.

[9] 宇津呂武仁, 颯々野学. ブートストラップによる低人手コスト日本語固有表現抽出. 情報処理学会情報処理学会自然言語処理研究会, Vol. 2000, No. 86, pp. 9–16, 2000.

[10] David Yarowsky. Decision lists for lexical ambiguity resolution: Application to accent restoration in spanish and french. In *Proceedings of the ACL*, pp. 88–95, 1994.

[11] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the ACL*, pp. 189–196, 1995.