

1 B-4

パーキング制御における 多視点ヒューリスティックスの取扱い

山端 潔 亀井 真一郎 村木 一至

NEC (株) C&C情報研究所

1. はじめに

曖昧性の解消は、自然言語の解析における最大の問題の一つである。最近では、入力に対する頑健性や曖昧性解消知識記述の柔軟性などの点から、解析結果に優先度を付与し相互比較により解を選択する、選好に基づいたアプローチが主流になりつつある[1, 2, 3, 4]。優先度付与には、原因を異にするさまざまな要因を考慮する必要がある。多数の要因にたいし評価手続きを与え、整合性を管理するのは大変難しい作業である。そのため、優先度計算が、自然で理解しやすい形で記述できることは、曖昧性解消の質を高めるうえで本質的である。

本稿では、構文木の選択のための優先度について、まず、単一の数値として構文木に付与しこれを単位に構文木上を伝播させる方式の問題点を示す。次に、優先度を各評価要因に対する評価値を要素とするベクトルとして表現し、これを単位に構文木上を伝播させる表現方法を提案し、これがこの問題点を自然な形で解決することを示す。さらに、この方式が、優先度間の整合性の管理、新規の評価要因の導入、文法の分割開発など実際のな面においても利点を持つことを論じる。

2. 優先度付与における問題

構文木への優先度付与の一般的な方式として、方式1:

- 優先度は各ノードに与える。構文木の優先度はそのルートノードの優先度である。
- 親ノードの優先度は、すべての子ノードの優先度と、親ノードを作成する際に用いられた文法規則に由来する数値(文法規則の優先度、組合せの優先度など)の和とする。

というもの[3, 4]がある。しかし、この方法には、

1. 優先度が、自分自身の評価と、親ノードの優先度の一部としての評価の二つの意味をもつが、これらが常に同一であるとは限らない
2. ひとつの要因に対する評価を行う手続きと、各要因の評価を総合する手続きが一体であり、要因間の整合性を保つ際に困難を招く

という問題がある。以下、これらについて例で説明する。

評価のキャンセル/再評価

関係節を含む文

I know the man she was talking to.

A Treatment of Multiple Preference Heuristics in Natural Language Analysis
YAMABANA Kiyoshi, KAMEI Shin'ichiro and MURAKI Kazunori
C&C Information Technology Research Laboratories, NEC

を、素性構造で補強された文脈自由文法で解析する場合を考える。関係節中に埋め込まれる文は、ギャップの存在を除き、統語的には一般の文と同様に振舞う。文法記述と管理の手間を考えると、ギャップのある/なしや、関係節中の文か独立した文か、の区別に関わりなく、同じ文法規則群で扱うことが望ましい。この場合、部分構造

she was talking to

には、単独の文として現れた時と同じ優先度が与えられる。一般に、ギャップを持つ構造が単独で現れた場合、容認性が低い場合、ギャップのない構造解釈よりも優先度が低くなければならない。そこで、上記の部分構造は低い優先度を持つことになる。しかし、解析が進み先行詞“the man”と結びついて

the man she was talking to

とまとまると、これは自立した一般の名詞句であり、優先度は決して低くないはずである。前述の方式1では、優先度は“the man”に由来する数値、“she was talking to”に由来する数値、および文法規則に由来する数値の和であるため、この優先度の差は文法規則中で高い優先度を与えて埋めるしかない。しかし、この文法規則自身がの文法規則と比べて特に高い優先度を持つ理由はない。下部構造での具体的な評価値に合わせて、下部構造における評価値が「結果として相殺されるように」優先度値を決めているのである。

この優先度のジャンプは、ギャップを持つ文では、自分自身の優先度(自身が単独で現れた時の優先度)と、関係節の一部としての優先度が異なることに起因する。ギャップを持つ文に対する人間の容認度は低いので前者は低くなければならないが、関係節の一部として見た時ギャップの存在は評価の対象にはならない。言い替えると、後者は前者の評価から「ギャップの存在」に由来するマイナス評価を除く、すなわちキャンセルすることにより計算される。

このように、ある時点で付与された優先度がキャンセルあるいは再評価される現象は、ギャップの存在のみならず、評価単位として本来一まとまりである構造が複数の規則により構造化・評価される場合には必ず起こる現象である。他の例として、日本語の名詞と助詞の整合性がある。助詞「で」は、「包丁で」は容認性が高いが「田中さんで」は容認性が低い(助動詞解釈が優先される)、のように、直前の名詞の意味により標準的な整合性が算出できる。しかし、後置される動詞が「で」格を意味的な必須格とし、名詞の意味に対する独自の制約を持っている場合(「田中さんで終わりだ」)、この整合性は動詞の情報を使って再評価されなければならない。助詞「で」を機能別に細分し別エッジとして扱うことも考えられるが、解析

の曖昧性をできるだけ減らそうとすると、このような再評価が必要になる。

評価要因間の整合性の管理

前述の方式1では、一つの評価項目に対する具体的な評価値を定める際に、その項目内での妥当性のみならず、他の評価要因との間の相互作用をもあらかじめ考えておく必要がある。特に、文法を扱う現象毎に分割して複数の人間で開発する場合、個々の評価要因を記述する段階で全体の整合性をとる必要があることは、文法記述を繁雑にし、評価要因間の整合性の維持を困難にする。

3. 提案する優先度表現の枠組

以上の考察にしたがい、望まれる優先度表現の原則をまとめてみる。

1. キャンセルや再評価など評価の柔軟な記述が可能である。
2. 個々の要因に対する評価を行う手続きと、要因を総合して全体の優先度を算出する手続きが分離されている。

この原則に則り、次の優先度表現の枠組を提案する。

方式2：

1. 各ノードに、要因毎の評価値を要素とするベクトル（ベクトルスコア）を付与し、該ノードをルートとする構文木の優先度情報を表す。
2. 文法規則適用によって新たな構文木を作る際には、ベクトルスコアの要素を単位として伝播を行う。すなわち、子供の構文木のベクトルスコアを要素毎に加算し、必要なら文法規則の指示により手を加えた後、親ノードのベクトルスコアとする。
3. ベクトルスコアとは別に、単一の数値であるトータルスコアを各ノードに付与し、構文木の選択に用いる。トータルスコアは、同じノードのベクトルスコアから構文木毎に算出する。これは親の構文木に原則として伝播しない。

要約すれば、優先度付与と構文木上の伝播の単位を単一の数値ではなく、評価項目からなるベクトルとすること、選択のための優先度の数値は、構文木作成の度にこのベクトルから総合評価用の関数により毎回計算すること、の二点を特徴とする。ベクトルスコアの要素の決定は、以下の原則にしたがう。

1. 構文木上の伝播の様子が異なれば分割する。
2. 総合評価への寄与の仕方が異なれば分割する。
3. 対応する言語現象が異なれば分割する。

4. 考察

この枠組では、構文木の上を親へと伝播されるのは、総合後のトータルスコアではなく、個々の要因に対する評価点をそのまま保持したベクトルスコアである。他の項目の評価値と分離されたまま構文木上を伝播するため、一つの項目に対する評価を変更することが容易である。例えば、例えば、ギャップをもつ文と先行詞をまとめる規則における項目「ギャップの存在」に対する評価のキャンセルは、対応するベクトルの要素を（例えば0点に）クリアすればよい。下部構造で与えられた具体的な

点数をこの文法規則が知っている必要はなく、単に規則の適用により評価項目「ギャップの存在」がキャンセルされることだけを知っていればよいことに注意して欲しい。再評価についても同様である。

さらに、各構文木に対し、その構文木における評価項目の各々に対する評価結果が付随している。しかも、これらベクトルスコアの各要素は、言語現象を単位に分割されており、その意味するところが明瞭である。従って、トータルスコアにどのような要因がどう寄与したかを容易に理解できる。また、個別の要因に対するスコアリングの手続きと、それらを総合して優先度を算出する手続きとが分離されている。このため、評価要因を総合してトータルスコアを算出する手続きの記述と、評価要因間の整合性の管理が容易になる。

新たな評価要因の導入も容易になる。この枠組によれば、評価要因の導入を、個別の評価要因を導入する第一の段階と、それを総合する第二の段階の二つのステップに分けて行うことができる。第一の段階では、ベクトルスコア中に対応する要素を設けて、関連する文法規則に評価法を記述する。第二の段階で、新しいベクトルスコアの内容と実際の解析結果とを見比べて、トータルスコア算出関数に手を入れる。評価要因間の相互作用はこの関数一箇所に集約されており、第一の段階であらかじめ評価要因間の相互作用を考慮する必要がない。

ルールの分割開発が容易になるのも利点の一つである。一般に、一つの評価要因は一つの文法現象に対応しており、文法記述者ひとりの担当範囲に入る。要因間の相互作用は総合関数として分離されており、要因内の評価は他の要因と独立に、ルールを分割したまま記述できる。全体の整合性は、ルールをあわせる段階で初めて意識すればよい。

ただし、この枠組は、ベクトルスコア保持のために大きな記憶領域を必要とし、これが問題となる可能性がある。また、トータルスコア算出時には、下部構造で既に行った計算を繰り返して行うことが多く、これも問題点の一つである。スコアの細分化はこれらの点を十分考慮して行う必要がある。

5. むすび

優先度の表現について考察し、評価のキャンセルや再評価が柔軟かつ自然に記述できることが望ましいこと、従来の一般的な表現ではこれが難しいことを述べた。そして、優先度の柔軟な記述のために、ベクトルとして表現して構文木上をこれを単位に伝播させる方式を提案した。さらに、評価要因の間の整合性の管理が容易になる、評価要因の新規導入や文法規則の分割開発が容易である等、実際的な利点をも持つことを論じた。

参考文献

- [1] Fass, Dan, and Yorick Wilks. "Preference Semantics, Ill-Formedness, and Metaphor", *Computational Linguistics*, vol.9, pp.178-187 (1983)
- [2] 谷 幹也、市山 俊治、村木 一至: "大局的な係受け構造情報による局所化並列解析モデル", 人工知能学会全国大会(第4回) 論文集 pp.349-350(1990).
- [3] 隅田 英一郎、丸山 直子: "拡張 CFG を用いた日本語構文解析", 情処研報 NL63-3 (1987).
- [4] Hobbs, Jerry. R, and Bear, John. "Two Principles of Phrase Preference." In *Proceedings of COLING-90*, vol.3, pp.162-167 (1990).