

論文の内容の要旨

Rule-driven Parsing Strategies for Robust Processing of Extra-grammatical Input in HPSG and Similar Unification-based Formalisms

(HPSG 等の単一化文法における文法外入力の頑健な処理のための
規則駆動型構文解析戦略)

シュタイナー ローランド
STEINER Roland

ほとんどの応用ソフトウェアにおける現在のユーザーインターフェースは、特に訓練されていない一般のユーザーには非常に複雑である。情報システムにより簡便にアクセスするためには、自然言語（日本語、英語などの人が話す言葉）による入出力が望ましい。しかしながら、自然言語の解析そのものは単純ではなく、特にユーザーが自由に発話した文を解析するのは非常に困難である。

より詳細な解析を出力する構文解析に取り組む現在の文法開発プロジェクトの多くは、素性構造を用いて単語、句の制約を記述した単一化のフレームワークを用いている。

そのような最先端の文法枠組の一つに HPSG がある。しかしながら、HPSG や他の単一化ベースの文法は基礎部として織り成す方法論の厳格さのため解析が却って困難となっている。つまり、文中のわずかな統語論的、意味論的間違い（言い間違い、書き間違いなど）により単一化は失敗してしまい、各々の文法に完全に追従する文に対してしか解析できない。人によって発話されたそのままの文にはそのような間違いが付随しているが、そのほとんどは例外的というよりも規則性を持っている。与えられた発話文に対する解析失敗の原因は雑音であったり、不明瞭な発音であったり、または慌ててタイピングをしたためであったり、一時的挿入文であったり、文法的間違いであったりいろいろな原因が考えられる。このような問題を取り扱うためには解析過程のうちそれぞれのステージにおける様々な手法が考えられるが、我々は純粋に文法的間違いに論点を絞り、HPSG や他の単一化文法において主眼視されている意味論的間違いにもある程度に言及する。

単一化ベースのシステムの頑健性を増す方法としては基本的に二つある。一つは、文法により規定される制約を緩めることである。しかしながら、いくつかの理由によりそれは望ましくない。解析過程で得られる情報が減り、探索空間を大きく広げ、たとえ適格文が与えられたとしても非常に冗長な曖昧性を生み出してしまう。もう一つの方法は、文中に出現する間違いに対し修正をしたり、許したりすることである。

文法の受容範囲を超える解析を許す方法の一つは、"単一化強制"である。単一化強制を用いることにより、システムは全ての句、単語の組み合わせを計算し、文法的衝突（例えば、人称の不一致など）が生じる場合でも解析される。探索と修正を繰り返して解析する方法では、そのような文法的衝突も解析され正しい解析結果が得られる。

しかしながら、そのような方法では、解析結果はあまりに一般的な解析結果（ありとあらゆる可

能性を考慮した解析結果) になってしまい却って情報量が少なくなり、また、高効率なシステムを実装するのは難しい。また、一般的に言って柔軟性がない。

これらの理由から、我々は、文法外入力を取り扱うためのユーザー定義可能なルールベースの機構を提案する。ルールベースのアプローチは、そのルールを拡張することでカバレッジを拡張することができる一方、ルールに記述された場合だけが取り扱われるため、おかしい結果はほとんどあるいは全く生成されないという利点がある。パーズングシステムが文法外であると判断した入力に基づいた(部分的な)結果を分類するために、我々はペナルティを使う。ここで、ペナルティの大きさは文法侵害の大きさを示す。このアプローチによって、システムは、ありそうもない入力の解釈を考慮する前に、低いペナルティが与えられた最も可能性の高いパーズ解析を最初に追求することができる。

この論文では、我々はまずメタ単一化の概念を導入する。メタ単一化とは、ルールで統御された単一化プロセスで、特定の状況の下、素性構造上の特定の位置についての単一化過程を "修正し"、通常の単一化から逸脱するような単一化結果を生成することをユーザーに許す。これを元にして、我々は単一化ベースのシステムにおける頑健性のための2つの主要な機構 M-ルールと拡張 ID スキーマを説明する。M-ルールは、2ステップからなるアプローチで、ユーザー定義によって型階層を拡張することで、単一化が失敗するような特定の状況下においても単一化が成功するようにするものである。この型階層の拡張は、型階層の任意の点を与えると一意に定まるもので、事前に拡張することで実行時のシステムに影響を与えない。我々はこの拡張を生み出す正確なアルゴリズムとその特性を詳述する。適度に拡張された型階層によって、ユーザー定義の M-ルールは、以前であれば衝突を起こすような単一化において、衝突の原因となる素性構造の場所を見つけだして修復する、すなわち、正しい結果に置き換える。これらを通常の単一化に関連づけるために、我々は実装の詳細と基本的枠組みの拡張について述べるだけでなく、M-ルールとメタ単一化の理論的特性についても考察する。

もう一つのアプローチ、つまり、拡張 ID スキーマでは単一化操作を行う前に文法的衝突を発見し、直接正しい結果に変換する。HPSG 文法枠組では単一化の対象となる二つの構文木構成要素は常に ID スキーマにより支配されるため、ID スキーマを拡張することが理想的である。名前が示すように、拡張 ID スキーマは文法的例外を直接検出し、取り扱いを行う。スキーマの変化は ID スキーマをより細かな部分に分割することにより定義される。この細かな部分を素性構造構築ブロック (FSBB) と呼ぶ。文法的例外を取り扱うためにユーザーは単一化を妨げる FSBB を見つけ取り除き、そして、意味のある出力を得るために他の FSBB を追加することになる。各々の例外を記述するルールは前もって定義された種類の現象に対してしか作用しないので、ユーザーはこのような現象を唯一に発見できる FSBB を追加できる。このように制約を追加することにより、探索空間を自在に制限することができる。

拡張 ID スキーマの他の利点は、複数の例外を同時に扱えるように例外の記述を合成することができるということにある。そのような場合、全ての例外定義から ID スキーマへの修正は同時に行われる。この方法の理論的特性は非単調性を避けられるということである。

M-ルールと同様に我々は拡張 ID スキーマの実装の詳細、拡張の可能性、および実験について報告する。