

90 Overview of the Core Language Engine

H. Alshawi (SRI International, 英国)

発表要旨

Core Language Engine (CLE) はドメインに依存しない、自然言語 (英語) の文章からその文の意味の正規表現にトランスレートするためのものである。この正規表現は推論を支援することが可能である。データベース管理システムや、医学診断のエキスパート・システムとのインターフェースのような相互作用的な助言システム (advisor system) の重要な要素として設計された。CLE の重要な貢献は、構文的にも意味的にも、広範囲にわたる。しかも、拡張可能な英語の構造をカバーすることを目的としている点である。そこでは、ユーザがシステムの単語を拡張できるような相互作用的な機能が用意されている。CLE は様々な段階の言語的なプロセスとの十分なインターフェースを持つモジュール的な構造を持っている。ユニフィケーションがルール適用の機構として、又、形態素解析、パーシング、インタプリテーション、フィルタ選択の各部分で情報を渡すための機構として使われている。従って、これらの各部分のルールは宣言的に表現できる。ローカルな曖昧性を表す簡潔 (packing) な表現が構文にも意味にも系統的に適応されているので、計算効率を落とさずにモジュール演算を行うことができる。

質疑応答

質問：私の貴方が話した曖昧性の処理のコメントについて興味があります。具体的に言いますと、曖昧さ、又は選言を atomic feature レベルにも feature matrix レベルにも簡潔 (packing) な記述方法で記述できるとのことですね。複雑さが問題となる典型的な場合は、このような選言、又は、曖昧性を持つ構造に、相互作用の可能性がある場合ですが、貴方はこれのために何か工夫をしていますか。

回答：実際は、ユニフィケーションが直接に選言の構造に適用されるわけではありません。むしろ、packing がどのように働くかという、表現には二つのソースの参照が使われています。一つは、カテゴリが入力文の要素の構文か意味をどのくらいカバーしているかを表し、もう一つは、そのカテゴリが解析中、他にどのような候補があるのかを表します。ここで行わなければならないことは、全てのカテゴリをその選言において含むようなカテゴリが唯一存在するための包含検査です。パーシングの時にこの pack 表現では、feature 間の包含されるもの同士のチェックを行わなければならない。しかし、私が先程述べたように、我々は選言を許さない Prolog のユニフィケーションを使っていますが、ユーザーの構文ルール (syntactic rule) からの表現をコンパイルします。例えば、「単数の第三人称ではない」を表す feature の場合、このような情報は feature の値にはなりません。これは、有限のドメインの場合ではユニフィケーションにマッチできるような形にコンパイルできますが、無限のドメインの場合では不可能だからです。