A Complex (Hungarian) Parser as an Embedded System

Balázs Kis, Mátyás Naszódi, Gábor Prószéky

MorphoLogic
{kis,naszodim,proszeky}@morphologic.hu

This paper presents the development, new features and applications of the HumorESK parser being developed since 1995.

HumorESK does not implement a fundamentally new parsing algorithm. It performs parsing in a bottom-up fashion; individual symbols are assigned simplified feature structures. While building the parse forest, it checks and propagates values in these feature structures. The logic of HumorESK operation closely resembles that of the PATR-II grammar formalism (Shieber, Uszkoreit, Pereira, Robinson, Tyson 1983).

The novelty in the HumorESK implementation is the method of applying rules and the grammar description itself. Rules are represented as finite sets of underspecified patterns, so that the system can use a lexicon similar to those in morphological analyzers. Each pattern consists of underspecified components (symbols); some symbols are described only by the label of the morphosyntactic class; others have the lexical form (the lemma) or even the surface form specified.

The author of the grammar is free to decide on the diversity of the morphosyntactic labels and the level of utilization of the feature structures. The architecture of HumorESK even enables to match rules without respect to the constituents' surface order – this feature is still under development. In the latter case, symbols can be derived to form a non-projective syntactic structure.

The HumorESK architecture has been designed to enable the parser to easily integrate with various applications as an embedded component. The program is entirely data-driven, which means that it makes no assumptions on the particular language, the grammar and the depth of parsing. This makes it possible to parse entire sentences or multiple-sentence structures to a great depth; this could drive detailed content extraction applications (the authors are able to present such an application as well). With other grammars, HumorESK can parse partial structures to a small depth, even perform shallow parsing; this provides for NP chunking or generalized collocation search. Parsing depth and scope can be different with the same grammar as well: grammars can be split into multiple levels, and the configuration can specify the last level HumorESK will execute, so parsing can be stopped before it reaches the topmost symbols.

It is an important feature of HumorESK that the patterns representing the rules can be assigned transformations. Thus, with appropriate formulation of the rules, the system can 'translate' the text into a logical structure in parse time.

HumorESK is a real-time system; parsing time can be limited; parsing data can be retrieved even when, according to the parsing logic, the process is not complete for a given segment.

In the presentation, the authors will briefly describe applications and projects where the HumorESK parser was utilized.