Developing Computational Morphology for Low- and Middle-Density Languages

Kemal Oflazer

Carnegie Mellon University - Qatar
Education City, PO Box 24866, Doha, Qatar**
ko@cs.cmu.edu

Words in languages contain pieces of syntactic and semantic information encoded in various ways. Many language processing tasks need to either extract and process the information encoded in the words or need to synthesize words from available semantic and syntactic information. Computational morphology aims at developing formalisms and algorithms for the computational analysis and synthesis of word forms for use in language processing applications. Applications such as spelling checking and correction, stemming in document indexing etc., also rely on techniques in computational morphology especially for languages with rich morphology.

Morphological analysis is the process of decomposing words into their constituents. Individual constituents of a word can be used to determine the necessary information about the word as a whole and how it needs to interpreted in the given context. Such information may range from basic part-of-speech information assigned from a fixed inventory of tags to structural information consisting of the relationships between components of the word further annotated with various features and their values. *Morphological generation* synthesizes words by making sure that the components making up a word are combined properly and their interactions are properly handled.

This tutorial will present an overview of the techiques for developing morphological processors that can be used in part-of-speech tagging, syntactic parsing, text–to–speech, spelling checking and correction, document indexing and retrieval, and many other language processing applications.

This tutorial cover the following main topics:

- An overview of aspects of morphology and how morphology encodes information in various languages
- An overview of computational morphology and how it relates to other tasks in natural language processing,
- A review of the mathematical background: finite state recognizers, regular languages and expressions, finite state transducers, regular relations, combining transducers, extended regular expression operators, etc.
- Fundamental components of morphological processors: lexicons, morphophonological and morphographemic rules,
- Finite state approaches to computational morphology,

** On long term leave from, Faculty of Engineering and Natural Sciences, Sabancı University, Istanbul, 34956, Turkey

- Parallel constraints approach: Two-level mophology,
- Serial transductions: Cascaded rules,
- Issues in engineering wide-coverage morphological analyzers

Depending on the time, the tutorial may cover certain interesting morphological phenomena in certain languages can be computationally handled, and provide an demonstration of various aspects of *xfst*, the Xerox Finite State Tool, for building morphological processors.