

1. Finite-state Language Processing

Shuly Wintner, University of Haifa

<http://cs.haifa.ac.il/~shuly>

Finite-state technology is becoming an invaluable tool for various levels of language processing. It is the computational means of choice for describing the phonology, lexicon and morphology of natural languages, but is used more and more for other purposes as well, including (shallow) parsing, word-level translation, named entity recognition, etc.

The tutorial will provide an introduction to the technology and its many applications in natural language processing. It starts with the very basics of finite-state devices and regular expressions and concludes with a sketch of how to design and implement a large-scale project. Several examples of real applications illustrate the formal material.

1.1. Tutorial Outline

1. Finite-state automata (FSA)
2. Regular expressions
3. Operations on automata
4. Applications of FSA in NLP
 1. Storing lexicons
5. Regular relations
6. Finite-state transducers (FSTs)
7. Properties of FSTs
8. Applications of FSTs in NLP
 1. Morphological analysis
 2. Part of speech tagging
 3. Translation dictionaries
9. Extended regular expression languages
10. Replace rules and composition
11. Applications
 1. Markup
 2. Morphological analysis and generation
 3. Shallow parsing
12. Available tools

1.2. Target Audience

This tutorial is designed for computer scientists and linguistics alike. Acquaintance with basic formal language theory and knowledge of some programming language will be useful, but not mandatory.

Shuly Wintner is an assistant professor in the Department of Computer Science at the University of Haifa, Israel. His research involves adaptation of computer science techniques and paradigms to computational linguistics, with an emphasis on formal grammars and finite-state devices.