Journal of Automata, Languages and Combinatorics **10** (2005) 5/6, 655–669

LEARNING THE MORPHOLOGICAL FEATURES OF A LARGE SET OF WORDS [1]

ABOLFAZL FATHOLAHZADEH
Supélec – Campus de Metz
2, rue Édouard Belin, 57078 Metz, France, and
Institute for Advanced Studies in Basic Sciences (IASBS)
Zanjan 45195, Iran
e-mail: abolfazl.fatholahzadeh@supelec.fr

ABSTRACT

This paper presents a new computational method for learning the morphological features (LMF) of a large input language. LMF makes use of our recent plan based on both automata theory and machine learning for mapping the input language onto a set of values, consisting of associating a finite-state automaton accepting the input language with a decision tree representing the output values. The advantages of this plan is that it leads to more compact representations than transducers, that decision trees can easily be synthesized by machine learning techniques. To improve yet these advantages, we introduce the integration of the default logic in LMF allowing to deal with regular and irregular strings of used language. By regularities, we mean repeated suffixes in strings. Thanks to hierarchical sub-languages formed by the default logic, it turns out that our extension is more beneficial for compact representation and fast lookup of a large set of strings. Experiments done on the two large datasets demonstrate the effectiveness of our method from the time and space requirements of the computation.

Keywords: Automata theory, machine learning, default logic, morphological features

1. Introduction

The morphological features (i. e., mode, tense, person and gender) are supposed to be the important ingredients of the lexicons which are widely used in the process of determining for a word (e. g., "livre") its output values (e. g., Verb+IND-PRES-1-SING, Verb+IND-PRES-3-SING, Verb+IMP-PRES-3-SING, Noun+MASC-SING and Noun+FEM-SING).

An obvious solution to such a task is to store the words along with their associated output values in a large-scale dictionary. But in this case two major problems have to be solved: *fast lookup* and *compact representation*. Two efficient methods can achieve fast lookup by determination and compact representation by minimization. The first

[1] Full version of a submission presented at the *Prague Stringology Conference* (Czech Technical University in Prague, Czech Republic, September 22–24, 2003).