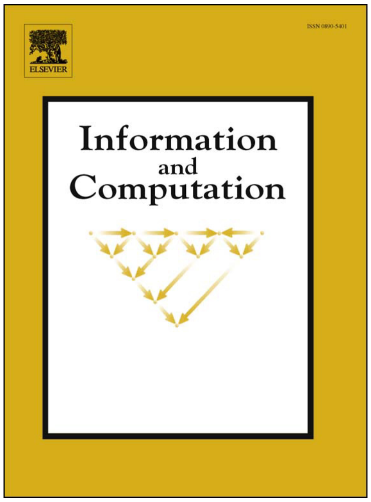

Distinguishing Pattern Languages with Membership Examples

Ziyuan Gao, Zeinab Mazadi, Regan Meloche, Hans Ulrich Simon, Sandra Zilles

PII: S0890-5401(17)30141-4
DOI: http://dx.doi.org/10.1016/j.ic.2017.08.003
Reference: YINCO 4302

To appear in: *Information and Computation*

Received date: 25 April 2016
Revised date: 21 February 2017

Please cite this article in press as: Z. Gao et al., Distinguishing Pattern Languages with Membership Examples, *Inf. Comput.* (2017), http://dx.doi.org/10.1016/j.ic.2017.08.003

Distinguishing Pattern Languages with Membership Examples

Ziyuan Gao[a,*], Zeinab Mazadi[a], Regan Meloche[a], Hans Ulrich Simon[b], Sandra Zilles[a]

[a] *Department of Computer Science, University of Regina, Regina, SK, Canada S4S 0A2*
[b] *Fakultät für Mathematik, Ruhr-Universität Bochum, D-44780 Bochum, Germany*

Abstract

This article determines two learning-theoretic combinatorial parameters, the teaching dimension and the recursive teaching dimension, for various families of pattern languages over alphabets of varying size. Our results and formal proofs are of relevance to recent studies in computational learning theory as well as in formal language theory. This is an expanded and corrected version of an earlier paper [Mazadi, Z., Gao, Z., Zilles, S: Distinguishing pattern languages with membership examples. In: LATA (2014) 528–540].

Keywords. pattern languages; regular pattern languages; computational learning theory; teaching; teaching dimension; recursive teaching dimension

1. Introduction

A pattern is a non-empty finite string of variable symbols and constant symbols, the latter being chosen from a fixed alphabet Σ. Each pattern generates a formal language, which contains all words obtained by replacing the variables in the pattern with non-empty words over Σ [1]. Since their introduction by Angluin in 1980 [1], pattern languages have continually served as interesting objects of study both in computational learning theory and in formal language theory. A large variety of very recent studies include for example the following.

(1) In computational learning theory: learning extensions of pattern languages [11, 12] and novel models of learning pattern languages [10, 11].

(2) In formal language theory: embedding pattern languages into the Chomsky hierarchy [14, 21], the complexity of the membership problem for pattern languages [7**?** , 11, 20], and decision problems on comparing pattern languages [9].

Pattern languages have also found applications in bioinformatics [2], for text editing in automatic program synthesis [18], in database theory [3], and generally in pattern matching [4].

[*] Corresponding author

Email addresses: gao257@cs.uregina.ca (Ziyuan Gao), mazadi2z@cs.uregina.ca (Zeinab Mazadi), reganmeloche@gmail.com (Regan Meloche), hans.simon@rub.de (Hans Ulrich Simon), zilles@cs.uregina.ca (Sandra Zilles)