Sequences, Datalog and Transducers

Anthony Bonner
Department of Computer Science
University of Toronto — Canada M5S 1A1
`bonner@db.toronto.edu`

Giansalvatore Mecca
D.I.F.A. — Università della Basilicata
Via della Tecnica, 3 — 85100 Potenza, Italy
`mecca@dis.uniroma1.it`

Abstract

This paper develops a query language for sequence databases, such as genome databases and text databases. The language, called *Sequence Datalog*, extends classical Datalog with interpreted function symbols for manipulating sequences. It has both a clear operational and declarative semantics, based on a new notion called the *extended active domain* of a database. The extended domain contains all the sequences in the database and all their subsequences. This idea leads to a clear distinction between safe and unsafe recursion over sequences: safe recursion stays inside the extended active domain, while unsafe recursion does not. By carefully limiting the amount of unsafe recursion, the paper develops a safe and expressive subset of Sequence Datalog. As part of the development, a new type of transducer is introduced, called a *generalized sequence transducer*. Unsafe recursion is allowed only within these generalized transducers. Generalized transducers extend ordinary transducers by allowing them to invoke other transducers as "subroutines." Generalized transducers can be implemented in Sequence Datalog in a straightforward way. Moreover, their introduction into the language leads to simple conditions that guarantee safety and finiteness. This paper develops two such conditions. The first condition expresses exactly the class of PTIME sequence functions; and the second expresses exactly the class of elementary sequence functions.

Appears in *Journal of Computer and System Sciences (JCSS)*, 57(3): 234-259, December 1998. Special issue on the Fourteenth ACM Symposium on Principles of Database Systems (PODS'95).