Learning in System F

JOEY VELEZ-GINORIO, Massachusetts Institute of Technology, U.S.A.

NADA AMIN, Harvard University, U.S.A.

Program synthesis, type inhabitance, inductive programming, and theorem proving. Different names for the same problem: learning programs from data. Sometimes the programs are proofs, sometimes they're terms. Sometimes data are examples, and sometimes they're types. Yet the aim is the same. We want to construct a program which satisfies some data. We want to learn a program.

What might a programming language look like, if its programs could also be learned? We give it data, and it learns a program from it. This work shows that System F yields a simple approach for learning from types and examples. Beyond simplicity, System F gives us a guarantee on the soundness and completeness of learning. We learn correct programs, and can learn all observationally distinct programs in System F. Unlike previous works, we don't restrict what examples can be. As a result, we show for the first time how to learn arbitrary higher-order programs in System F from types and examples.

Additional Key Words and Phrases: Program Synthesis, Type Theory, Inductive Programming

Authors' addresses: Joey Velez-Ginorio, Brain & Cognitive Sciences, Massachusetts Institute of Technology, 43 Vassar St., Cambridge, Massachusetts, 02139, U.S.A., joeyv@mit.edu; Nada Amin, John A. Paulson School of Engineering and Applied Sciences, Harvard University, 29 Oxford St., Cambridge, Massachusetts, 02138, U.S.A., namin@seas.harvard.edu.