Machine Learning for Compilers

Chris Cummins, University of Edinburgh
c.cummins@ed.ac.uk

July 4, 2018

The meteoric rise of machine learning is promising enormous changes to how we write software, spearheaded by sub fields such as deep learning. Recent work in applying language modeling to program code indicates that it may be possible to build models which fully capture the semantics of programming languages. Such models could drastically reduce the cost of development for compilers and compiler tools. Potential applications of such models include learned source-to-source code translation, automatic generation of tests to maximize code coverage, and automated generation of algorithms from high level specifications. I believe that innovations in machine learning techniques, combined with increased compute and data availability, have the potential to revolutionize the way software is developed in the coming years.