Combining strategies efficiently: high-quality decisions from conflicting advice

Koolen, W.M.

Publication date
2011

Prelude

I want to build intelligent computer systems. Like every computer programmer, I experience great joy and pride when my creation successfully performs a complex task. After half a century of Moore's law[1] we do have the hardware necessary for intelligent behaviour[2]. However it seems that more complicated tasks require more complicated programs. This problem is actually so severe that a lot of intelligent tasks are completely beyond the abilities of current systems. We simply have no clue how to program the desired behaviour. We've hit the so-called *software bottleneck*.

The solution is relatively simple, at least in principle (it is the one encountered in Nature). Instead of trying to *design* a complete system for the desired behaviour, we build a flexible system that can *learn*, and then *train* it to perform the desired task. The design and analysis of such systems is called *machine learning*.

I particularly like the modular approach of building systems that can be immediately put into production untrained. While executing its task, the system continuously learns by receiving (virtual) reward for desired behaviour and (virtual) punishment for *faux pas*. This feedback allows the system to converge to the desired behaviour quickly. The study of this class of systems is called *online learning*. This thesis is the result of four years of doctoral research in online learning.

[1] G. E. Moore noted that the transistor count of integrated circuits doubled every two years.

[2] For example, in November 2009, IBM Research presented its cortical simulator, which runs on supercomputer hardware, and whose simulations exceed the size of the cat cerebral cortex [7]. Earlier milestones include rat-scale and mouse-scale cortices.