The Virtue of Reward: Performance, Reinforcement and Discovery in Case-Based Reasoning

Derek Bridge

Department of Computer Science,
University College, Cork
Ireland
d.bridge@cs.ucc.ie

Agents commonly reason and act over extended periods of time. In some environments, for an agent to solve even a single problem requires many decisions and actions. Consider a robot or animat situated in a real or virtual world, acting to achieve some distant goal; or an agent that controls a sequential process such as a factory production line; or a conversational diagnostic system or recommender system. Equally, over its life time, a long-lived agent will make many decisions and take many actions, even if each problem-solving episode requires just one decision and one action. In spam detection, for example, each incoming email requires a single classification decision before it moves to its designated folder; but continuous operation requires numerous decisions and actions.

Reasoning and acting over time is challenging. A learner's experiences may prove unrepresentative of subsequent problems; a changing environment can render useless the system's knowledge. A system that tries to solve hard combinatorial problems, for example, may find, through exploration in the space of solutions, that earlier training examples are suboptimal. Concept drift in spam detection is another example: spammers send new kinds of unwanted email or find new ways of disguising spam as ham. Agents must be highly adaptive if, over time, they are to attain and maintain high standards of, for example, accuracy, coverage and efficiency.

To address these challenges in case-based agents, I have been drawing ideas from another field, that of *classifier systems*. Classifier systems, first proposed by John Holland, are rule-based systems. They comprise a performance component, a reinforcement component and a discovery component. The performance component chooses the agent's actions. The other two components enable classifier systems to exhibit two kinds of plasticity, parametric plasticity and structural plasticity. The reinforcement component uses feedback from the environment to update rule quality parameters. The discovery component uses genetic operators and other techniques to propose new rules, which may displace existing rules.

I will describe my attempts to build a case-based counterpart to Stewart Wilson's XCS, which is one of the most popular, modern classifier systems. I will describe each of its three components. In discussing the reinforcement component, I will offer reflections on the relationship between Case-Based Reasoning and reinforcement learning. In discussing the discovery component, I will offer reflections on automatic case discovery and case base maintenance.