Chapter 2.3
Machine Learning for Agents and Multi-Agent Systems

Daniel Kudenko
University of York, UK

Dimitar Kazakov
University of York, UK

Eduardo Alonso
City University, UK

ABSTRACT

In order to be truly autonomous, agents need the ability to learn from and adapt to the environment and other agents. This chapter introduces key concepts of machine learning and how they apply to agent and multi-agent systems. Rather than present a comprehensive survey, we discuss a number of issues that we believe are important in the design of learning agents and multi-agent systems. Specifically, we focus on the challenges involved in adapting (originally disembodied) machine learning techniques to situated agents, the relationship between learning and communication, learning to collaborate and compete, learning of roles, evolution and natural selection, and distributed learning. In the second part of the chapter, we focus on some practicalities and present two case studies.

INTRODUCTION

Intelligence implies a certain degree of autonomy, which in turn, requires the ability to make independent decisions. Truly intelligent agents have to be provided with the appropriate tools to make such decisions. In most dynamic domains, a designer cannot possibly foresee all situations that an agent might encounter, and therefore, the agent needs the ability to learn from and adapt to new environments. This is especially valid for multi-agent systems, where complexity increases with the number of agents acting in the environment.

For these reasons, machine learning is an important technology to be considered by designers of intelligent agents and multi-agent systems.

The goal of this chapter is not to present a comprehensive review of the research on learning agents (see Sen & Weiss, 1999, for that purpose) but rather to discuss important issues and give the reader some practical advice in designing learning agents.

The organization of the chapter is as follows. In the following section, the differences between pure machine learning and that performed by (single) learning agents are discussed. We start with the introduction of basic machine learning concepts, followed by examples of machine learning techniques that have been applied to learning agents, such as Q-learning, explanation-based learning, and inductive logic programming. In the third section, we discuss several issues surrounding multi-agent learning, namely, the relationship between learning and communication; learning to collaborate and compete; the learning of roles, evolution, and natural selection; and distributed inductive learning. Following this discussion, we focus on some practicalities and present two case studies. We finish the chapter with conclusions and further work.

FROM MACHINE LEARNING TO LEARNING AGENTS

In this section, we discuss the nature of machine learning (ML), its integration into agents, and the parallels between machine learning systems and learning agents. We start with a basic introduction to machine learning.

While most of the fundamental ML concepts introduced below are commonly associated with *supervised learning (SL)* (i.e., the generalization from annotated examples provided by a teacher), they are equally relevant for *reinforcement learning (RL)*, where an agent learns through the feedback (i.e., reinforcement) from the environment in each entered state. To date, most attention in agent learning has been reserved for RL techniques such as Q-learning (see below), due to its suitability to situated agents. Nevertheless, we see SL and RL as strongly related approaches. In the case of RL, the environment could be seen as the teacher, and the generalization process would be over states (which correspond to the examples). In fact, SL methods can be directly applied in a RL setting (see, e.g., the use of neural networks in TD learning (Tesauro, 1992).

Note that we mostly exclude a third class of ML techniques, *unsupervised learning* (or *learning by discovery*), from our discussion, because there are only few research results in this area to date.

Introduction to Machine Learning

A computer program is said to learn *from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E* (Mitchell, 1997). Machine learning can be seen as the task of taking a set of observations represented in a given *object* (or *data*) *language* and representing (the information in) that set in another language called the *concept* (or *hypothesis*) *language.* A side effect of this step can be the ability to deal with unseen observations. As an example, one can consider an object language that consists of pairs of coordinates (`x,y`). A number of observations are provided, and each is labeled as a positive or negative example of the *target concept*, i.e., the concept to be learned. Let the hypothesis language define a concept as an ellipse,[1] such that a point (`x,y`) would represent a positive example of the concept if it is inside that ellipse or a negative one otherwise. In the example in Figure 1, there are infinitely many such ellipses, each of which would satisfy the usual requirements for *completeness* and