Machine learning is the key to machine intelligence just as human learning is the key to human intelligence.

Richard Forsyth

Chapter 2

Preliminaries

In this chapter we give a short overview of those research fields forming the basis of this thesis. First, we consider the various aspects of learning, the organization of a learning system including an environment in which the learning takes place, and the nature of the learning mechanisms. This is followed by a review of the major paradigms of *machine learning* with *evolutionary algorithms* in the focus. Finally, some background about *formal languages* is given and the basic terminology of *attribute grammars* is introduced.

2.1 Machine learning

Machine learning (ML) is an interdisciplinary subfield of *artificial intelligence* that deals with modeling and realizing the cognitive process of learning. It is well-known that a great number of factors and conditions may influence the learning phenomena. Many of them are still not identified explicitly even though researchers in neurobiology discover more and more about the detailed structure of the brain thanks to todays' modern techniques.

Therefore, one cannot simulate or model the process of learning the way it is probably done in the brain of a human or an animal. One can only try to make computers imitate this behavior since, as Natarajan in [57] points out:

> "Whether or not they [computers] think like humans is not of great concern to us, as long as they appear to do so when observed from outside."

The ability to learn is considered as a distinctive characteristic of intelligence and has been extensively studied by philosophers and cognitive scientists for over a century. Moreover, ever since the first computer was invented the attempt to make them learn has been playing an important role in artificial intelligence.

The early phase of machine learning is characterized by the development of general learning mechanisms. The first approaches to letter recognition, game playing, the foundations of neuronal networks, and the first model of the perceptron date back to this time.

This beginning euphoria, however, was cooled down by the analysis of Minsky and Papert on the limitation of the perceptron model in 1969 [51]. Parallel to this disappointing result, in the mid '60s the AI researchers realized the importance of an a priori knowledge and mainly focused on understanding the role of this knowledge with regards to intelligent behavior. The main research concentrated on knowledge representation and on the construction of expert systems. Although only marginal attention was paid to the issues of learning, the roots of some methods, elementary nowaday, such as incorporating heuristics and general domain-independent methods go back to this period.

The renaissance of machine learning began with the 1980s. The renewed interest and extensive research led to a firm methodological basis with systematic experimentation on common databases and precise theoretical analysis. The work on language acquisition and concept induction, which began in the years of stagnation in the '70s, was continued and extended by new research topics like: evolutionary algorithms; learning methods concerning problem solving, planning, and control; and the revived neural networks.

Machine learning is the most attractive research field of artificial intelligence in which the introduction of a great variety of new hybrid learning methods and the issues of applying machine learning on real-world problems come more and more to the front. Table 2.1 offers a short summary of the disciplines contributing to the evolution of machine learning.

In [42] four basic motivations of machine learning are discussed:

- *Psychological*, *cognitive scientific*: the learning algorithms are developed to model certain specific learning behavior in order to better understand the mechanisms that form the basis of human learning.

- *Empirical*: the aim is to discover general principles that relate the characteristics of learning algorithms and of the domain in which they operate to the learning behavior. The standard approach involves, firstly, running experiments that vary either the algorithm or the domain, and then, secondly, observing the impact of these manipulations on the learning process.

- *Mathematical*: A typical approach involves defining a learning problem, conjecturing that it can or cannot be solved with a reasonable number of training cases, and then proving that the conjecture holds under very general conditions (cf. *Probable Approximately Correct Learning* [34], Gold's *learning in limit* [24]). Other research focuses on the examination of alternative reasoning methods in the learning course apart from inductive inference, such as deduction and abduction.

Table 2.1: Research fields influencing machine learning

Artificial Intelligence	Learning symbolic representation of concepts. Machine learning as a search problem. Learning as an approach to improving problem solving. Using prior knowledge together with training data to guide learning.
Bayesian Methods	Computing the probability of a hypothesis or the probable value of a feature. Stochastic/probabilistic learning.
Biology	Search strategies motivated by the nature: evolutionary algorithms. Neuro-biological studies motivating artificial neuronal network models of learning.
Computational Complexity Theory	Theoretical bounds on the inherent complexity of different learning tasks, measured in terms of the computational efforts, number of examples, number of mistakes, etc. required in order to learn.
Information Theory	Heuristic measures of entropy and information content, minimum description length approaches. Optimal codes and their relationship to optimal training sequences for encoding the hypothesis. Mathematical/theoretical learning.
Pattern Matching	The origin of the notion "overfitting" and the approaches like feature-selection or nearest-neighbor classification.
Philosophy	Occam's razor: suggesting the simplest hypothesis is the best. Reasoning methods. Analysis of the justification for generalizing beyond observed data.
Psychology	The *power law of practice* which states that over a very broad range of learning problems, people's response time improves with practice.
Statistics	Characterization of errors (e.g. variance) that occur when estimating the accuracy of a hypothesis based on a limited sample of data. Confidence intervals, statistical tests. Hypothesis evaluation.

- *Application-oriented*: The primary aim is to apply machine learning to real-world problems: ML holds the potential for automating the process of knowledge acquisition since it can transform training data into knowledge.

  The typical steps involve a developer formulating an interesting problem in terms of machine learning: s/he must design a representation for training cases and learned knowledge, collect the training data, use machine learning to generate a knowledge-base, and work with other users to create the resulting knowledge-based system.

The main thread, however, focuses on design, understanding and evaluation of *learning algorithms*.

2.1.1 What does 'learning' mean?

In daily life one may refer to one or another situation where 'learning' takes place by saying:

```
I'm learning literature.
I'm learning to swim.
I'm learning to program effectively in C++.
```

The definition taken from the Merriam-Webster Dictionary comprehends the several aspects of learning by defining *learning* as:

> To gain knowledge or understanding of or skill in by study, instruction, or experience.

This definition clearly marks the main points of learning, its adaption to computers is nevertheless very complicated. In literature one can find several approaches to formalize the notion of *machine learning*, emphasizing one or another aspect of it. Let us review the most important ones.

In [77] Simon proposes the following definition:

> "Learning denotes *changes in the system* that are adaptive in the sense that they enable the system to do the *same task* or *tasks drawn from the same population more effectively* the next time."

Simon sees the essence of learning in accomplishing *changes in the system* for the sake of improved performance in time. This approach is often criticized for two reasons: some phenomena, which are viewed as learning, do not satisfy this definition, while some typically non-learning phenomena are covered by it. For instance, the fine-tuning on system parameters that leads to more effective results would be considered as learning by Simon.

Michie considers the achieved improvement of some kind of performance and reorganization of the gained knowledge as the basis of learning. He argues that it could be useful for the user of the system even if the performance will not improve. In [49] he writes:

> "A learning system uses sample data to *generate an update basis* for *improved [performance]* on subsequent data from the *same source* and *express* the new basis in *intelligible symbolic form*."

Both the improved performance and the experience in learning are also acknowledged by Langley in [42]. Besides she emphasizes the impact of the environment on the learning process as well:

> "Learning is the *improvement of performance* in some *environment* through the acquisition of the *knowledge* resulting from *experience* in that environment."

Clearly, it is rather complicated to precisely define the notion of learning for computers involving every aspect which might be relevant for the design of a learning system. The previous approaches emphasize important characteristics of learning, yet they are too ambiguous for a design of a learning system. In contrast to them, Mitchell gives a more formal definition of learning in [52]:

Definition 2.1 *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 measured by P at tasks in T improves with experience E.*

He underlines T, E and P to be the elementary components of a *well-defined learning problem.*

1. The *class of tasks T*: It is the domain of the learning activity. The learning system aims to solve these tasks by attempting to achieve certain improvements.

2. The *experience E*: It involves special observations, *instances* about a certain task of T, whose solutions are already known at the time of learning.

3. The *performance measure P*: During the learning process a learning system may aim to achieve various kinds of improvements like:

    - providing more accurate solutions;
    - covering a wider range of tasks of T;
    - obtaining the solutions more efficiently, thus improvement in speed;
    - simplifying the encoded results; etc.

The last goal presumes that a simplification of stored knowledge is valuable for its own sake (cf. Michie's approach). However, the first two criteria—accuracy of solutions and range of applicability—usually have the highest priority. Various goals are to achieve via a learning process, and these objectives strongly influence the choice or the definition of an appropriate performance measure.

In addition to the three components T, E, and P, a fourth has to be mentioned: the *background knowledge B*. It includes special auxiliary information about the class of tasks that may be useful during the learning process. Unfortunately, such information is not always available (or it cannot be precisely expressed in a suitable form that can be submitted to the learning system). Thus background knowledge is not a standard, but an optional component of the learning task.

2.1.2 The learning framework

In this section we describe a typical learning scenario and sketch the structure of a typical learning system.

In machine learning, the course of learning, just as in nature, is a multi-stage process including several revision phases. These steps can be divided into two main phases: the *training phase* followed by the *test phase*. Figure 2.1 depicts the interactions among the major components of a typical learning system (indicated as dark boxes) during the training phase.

The initial input of the system consists of the *training instances* and the *target output* associated to these instances. The *training instances* are certain observations: situations related to the class of tasks T, i.e. to the *problem domain*. The set of *target outputs* or *target values* represents the outcomes in those situations. Each target output o_i is defined by an *Ideal system* or an *Expert* of the problem domain who has carried out the target task on a training instance x_i. In practice, the training instances and target values are usually available in the form of $< x_i, o_i >$ pairs called *training examples*.

It is a fundamental characteristic of the learning problem that the learning system has no information about the way these results have been produced by the *Expert*. Hence the primary aim of the learning process is to infer *hypotheses* that lead to the same or almost the same output with regards to the training instances as the (target) output delivered by the *Ideal system*. In

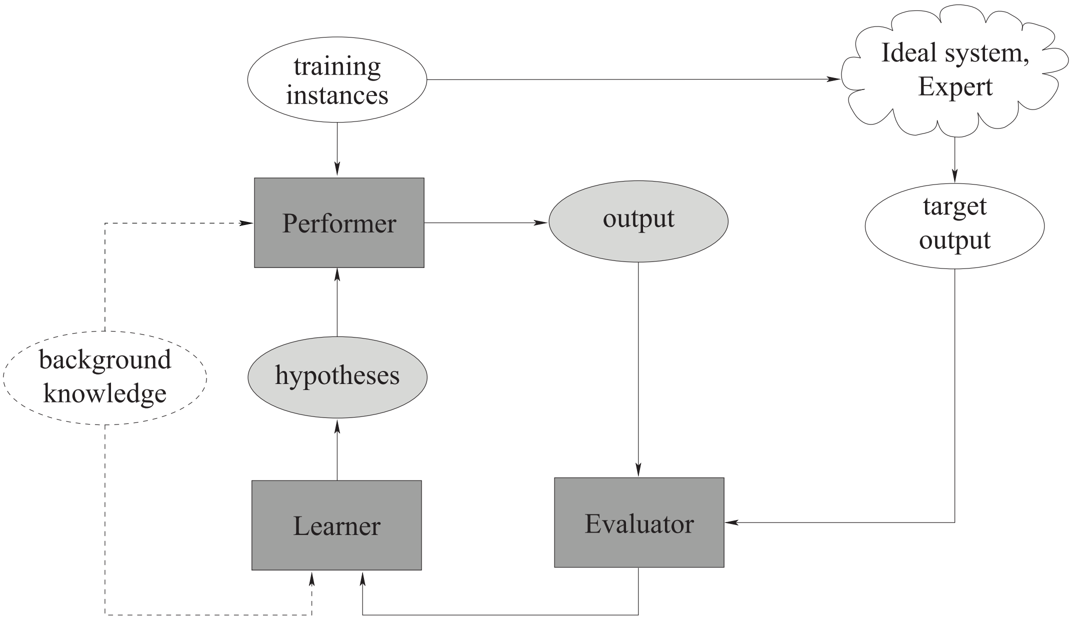

Figure 2.1: The workflow of a typical learning system

some cases, the construction of *hypotheses* may be supported by the *background knowledge* (marked by dashed lines in Figure 2.1).

According to their functionality three main modules of the system can be distinguished:

1. The *Learner* module is the heart of the system: it encapsulates the learning algorithm. It either infers new *hypotheses* for associating the input descriptions of the *training instances* with the *target outputs*, or it attempts to amend the already existing *hypotheses* according to the analysis made by the *Evaluator*.

2. The *Performer* (or *Interpreter*) is the part of the system that carries out the target task on the *training instances*. The *hypotheses* (*candidate solutions*) which encode the system's current level of expertise, are employed by the *Performer* as guides or solving strategies in creating the set of *output*.

3. The *Evaluator* module compares the *output* produced by the *Performer* to the *target output*. In other words the *Evaluator* measures the performance of the system on the *training instances* against the performance of the *Ideal system*. In general, this evaluation focuses on accuracy, i.e. on the correctness of the *output*, but the performance measure embedded in the *Evaluator* may specify other factors or components to examine as well. When the performance of the system achieves the desired level or the performance cannot be improved further, the training phase comes to its end.

   If further corrections or fine-tuning of the *hypotheses* are required, then the *Evaluator* passes feedback to the *Learner* module and so the system enters a new revision or training step.

The essential difference of the *test phase* compared to the *training phase* is that the *Learner* module is inactive: the goal here is to gain additional confidence about the performance of the system on examples, *test examples*, which so far have been unseen, i.e. which were not considered during the construction of the *hypotheses*. Clearly, in order to achieve an objective view about the quality of the results, the *test examples* must be of the same sort and must have the same distribution as the training examples. This elementary requirement is typically ensured by dividing the initial set of examples into two parts: one is used for the training of the system while the other is kept for testing.

It may occur that although the *hypotheses* exactly yield the expected *target output* on the *training instances*, they perform badly on the test examples. In the literature on machine learning this problem is termed as *overfitting*. Several methods are proposed for its avoidance: the usage of a more sensible performance measure during the training; the correction or post-pruning of the *hypotheses* by means of the test results.