

Overview of artificial intelligence

Experts in AI would say that AI is a domain in itself, which is neither reducible to computer science nor biology, for instance.

Some experts would probably say that AI pretends to *solve complex problems the way humans do*.

Some other AI experts would disagree and say that the goal of AI is to solve so hard problems that, somehow, *programs may seem intelligent*.

Overview/A computer scientist's point of view

My presentation of AI is not based on history or philosophy or cognitive sciences or biology. I will only try to give some **intuitions** and some **techniques**, as algorithms and heuristics, that are useful in practice.

So what is AI, then?

From our point of view, it is a collection of techniques for solving or getting partial or suboptimal solutions to complex or intractable problems.

Plan

I. Overview

- A computer scientist's point of view
- **Applications**
- Symbolic approaches
- Connectionist approaches

Overview/Applications

One reason why some problems are hard to solve is that they may come from the interaction between the physical world and the computer, considered as a discrete model of the world.

For instance, how can the computer cope with shape or **face recognition** in signal processing (or **speech recognition**) and improve its success rate given successful and failed samples?

In this case, part of the difficulty lies in the passage from continuity (physical phenomenon) to discontinuity (logical abstraction).

Overview/Applications (cont)

Some problems are difficult because their computational complexity is too high (i.e. their solution requires too much computation to be realistic) and/or because the data is incomplete, hence the need for good approximations.

For instance, find the **shortest path** from one town to another, given the whole map, or play a **maze game**, without the whole map.

Overview/Applications (cont)

Some problems are difficult because they involve **automatic logical reasoning**.

For instance, is it possible to assist a physician in establishing a diagnosis or help to pinpoint a car failure?

Or, in security analysis: how can we prove this safety theorem about this train without pilot?

Overview/Applications (cont)

Another approach is more ideological, which consists in modeling some biological process, without any specific problem in mind, and then find the fruitful problems.

For instance, some techniques come from a model inspired by the brain, called **abstract neural networks**, or from genetics, called **genetic algorithms**.

These techniques are nevertheless applied, with various extents of success, to unexpected different kinds of problems.

Plan

I. Overview

- A computer scientist's point of view
- Applications
- **Symbolic approaches**
- Connectionist approaches

Overview/Symbolic approaches

Symbolic approaches are based on formal logic reasoning.

The rationale is to model objects of the problem, as well as their relationships (in time or space), by **axioms** and **logical rules**.

- An axiom is a fact, something which is true by definition.
- A logical rule is an implication, like $A \Rightarrow B$, allowing to go from axioms to **theorems**.
- A theorem is a formula that can be deduced by logical rules from the axioms.

Overview/Symbolic approaches (cont)

A software assisting the task of proving theorems is called a **proof assistant** or a **knowledge-based system** (or **expert system**), depending on the abstraction level.

Proof assistants are used by computer theorists in the framework of very expressive logics, in order to prove for instance security property for embedded systems (i.e. autonomous vehicles like rockets, space robots, subway or plane without pilot, car ignition systems etc.).

Expert systems are more similar to a database of domain-specific informations and logic rules (simpler than within proof assistants) which allow queries to be formulated. Such systems are often used as an interactive tool to help an expert diagnose a failure. The **programming language Prolog** is most famous for such applications.

Plan

I. Overview

- A computer scientist's point of view
- Applications
- Symbolic approaches
- **Connectionist approaches**

Overview/Connectionist approaches

Connectionist approaches are based on the modeling of the brain as an **abstract neural network** whose properties are inspired by the electrical behaviour of biological neurons.

This approach tries to capture the transition between the continuity of physical phenomenon (electric waves) and discrete logical units (bits) by imitating brain-like structures.

An interesting question (not answered here) is how information pass from one model to another.

Overview/Connectionist approaches (cont)

Following the brain metaphor, the abstract neural networks are usually programmed through **learning** (or **supervised training**), i.e. by submitting inputs that must be accepted and inputs that must be rejected and, each time, adjusting the network parameters (called **weights**) to fit *all* the given examples (accept some and reject others).

Next, the network is expected to extrapolate accordingly to its learning phase to unknown examples. One application is **pattern recognition** (like bit-map character classification in noisy environments).