Introduction to AI

Michael J. Watts

http://mike.watts.net.nz

Lecture Outline

- Artificial / Computational intelligence
- CI models

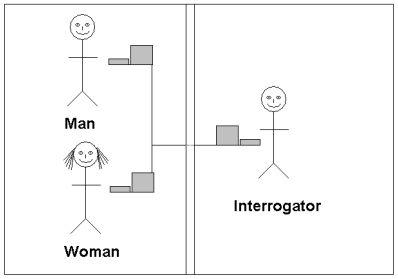
Computational Intelligence

- What's it all about?
  - The Turing test
  - An alternative definition

The Turing Test

- Proposed by Alan Turing in 1950
- Deals with the question: "Can machines think?"
- Turing thought this question absurd
- How to define 'think'?
- Dealt with it with a thought experiment

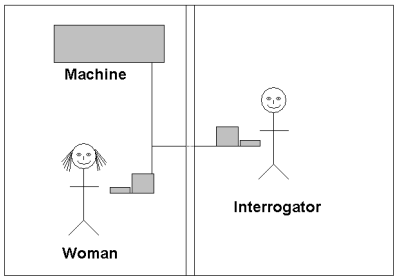
The Turing Test



The Turing Test

- Man's objective is to convince interrogator that he is a woman
- Woman's objective is to help the interrogator
- After a while, the interrogator must decide which is which
- Now, replace the man with a computer

The Turing Test



The Turing Test

- Will the interrogator decide wrongly as often as before?
- In other words, can the machine convince the interrogator that it is human as often as a man can pass as a woman?

The Turing Test

- "The Turing Test is no more a test for intelligence than it is a test for femininity… A man doesn't become a woman because he can fool you into thinking that he's a woman. By the same token, a machine doesn't become…an intelligent machine, just because it can fool you into thinking that it's thinking"
  - David B. Fogel, *Blondie24: Playing at the Edge of AI* pg 11

An Alternative Definition

- "Intelligence is the capability of a decision-making system to adapt its behavior to meet its goals in a range of environments"
  - David B. Fogel, *Blondie24: Playing at the Edge of AI* pg 14

Computational Intelligence

- Artificial Intelligence is now called Computational Intelligence
- Why the change?
  - Fashion?
  - Politics?
  - More accurate?

CI Models

- We will be studying three general paradigms of CI
  - Rule based and fuzzy systems (FS)
  - Artificial Neural Networks (ANN)
  - Evolutionary Computation (EC)

CI Models

- Why cover different models?
- No Free Lunch Theorem
  - If there is a problem A on which the algorithm performs well, there will be a problem B on which the algorithm performs poorly
  - Nothing is good at everything

Rule-Based Systems

- Systems that make decisions based on rules
- Used when the rules can be stated
- Crisp rules
  - When the numbers dealt with are always exact
  - Can be a pain in the neck to program, though
- Fuzzy rules
  - deals with inexact concepts – 'bigger', 'smaller', 'faster'
  - easier to state rules
  - Optimisation can be difficult

Rule-Based Systems

- First, define the rules
- Easier said than done
- Are the rules consistent?
- Are the rules complete?
  - Cover all possibilities
- Fuzzy systems
  - define the fuzzy membership functions

Artificial Neural Networks

- Based on models of the brain
- Consist of network of interconnected subunits
  - Neurons
- Used when the rules are not known
- ANN are learning structures
  - Don't need to be told the answer to the problem

Artificial Neural Networks

- Many kinds in existence
- We will be covering only three
  - Perceptrons
  - Multi-layer Perceptrons (MLP)
  - Kohonen Self-Organising Maps (SOM)

Artificial Neural Networks

- Which networks are used depends on the application
- perceptrons useful for simple problems
  - linear separability
- MLPs handle problems perceptrons cannot
  - Non linearly separable

Artificial Neural Networks

- Perceptrons and MLPs both use supervised learning
  - Must know the target values the network is learning
- SOMs are unsupervised
  - capture clusters in the data
  - vector quantisers

Artificial Neural Networks

- Care must be taken with the data used to train the network
  - It is easy to badly train an ANN
- Other circumstances where ANN don't function well

Evolutionary Algorithms

- Based on the mechanisms of natural selection and biological evolution
- Evaluates fitness of (initially) random solutions
- More fit individuals produce more offspring
- Search algorithms
- Used when brute-force (exhaustive) search is not feasible
- Useful for multi-parameter optimisation

Evolutionary Algorithms

- Several kinds exist
  - Genetic Algorithms (GA)
  - Evolution Strategies (ES)
  - Evolutionary Programming (EP)
  - Genetic Programming (GP)
- Require the following characteristics
  - representable
  - Fitness (objective) function

Representation

- Must be able to represent the problem in the algorithm
- some means of encoding candidate solutions

Evaluation

- some means of rating candidates must exist
  - binary ratings are no good
    - right/wrong
  - fitness function must be objective
  - fitness function must separate good candidates from bad candidates
  - Most problem dependent component of EA

Conclusions

- Computational intelligence is hard to define
- Oldest attempt is the Turing Test, but not very accurate
- Many different kinds of CI about
- Deal with 3 in this course
  - Rule based systems
  - Neural Networks
  - Evolutionary Algorithms