CS4060 Knowledge Representation & Automata

Introduction

www.scss.tcd.ie/Tim.Fernando/KRA

2017-18

CS4404 Machine Learning
CS4LL5 Unsupervised ML for NLP (since 2014)
CS4060 Knowledge Representation & Automata

Introduction

www.scss.tcd.ie/Tim.Fernando/KRA

2017-18
CS4404 Machine Learning
CS4LL5 Unsupervised ML for NLP (since 2014)

2014-2017
CS4061 Artificial Intelligence (Michaelmas)
CS4062 Machine Learning for NLP (Hilary)
CS4060 Knowledge Representation & Automata

Introduction

www.scss.tcd.ie/Tim.Fernando/KRA

2017-18
CS4404 Machine Learning
CS4LL5 Unsupervised ML for NLP (since 2014)

2014-2017
CS4061 Artificial Intelligence (Michaelmas)
CS4062 Machine Learning for NLP (Hilary)

1999-2015
CS4LL4ab Artificial Intelligence (year-long)
- logical orientation (Prolog, over LISP)
  Logicomix, by Doxiadis & Papadimitriou
Can machines think? (Turing 1950)

Turing test: can C tell A from B?

From Wikipedia, (Juan Alberto Sánchez Margallo)

Intelligence operationalized: subject to testing
Can machines think? (Turing 1950)

**Turing test:** can C tell A from B?

From Wikipedia, (Juan Alberto Sánchez Margallo)

Intelligence operationalized: subject to testing

... cheating?
ELIZA (Weizenbaum, 1964-66) & artful deception

- use pattern matching and substitution to fake understanding

**ELIZA effect**: humans are inclined to see computers as humans
e.g. when ATM says “thank you”
ELIZA (Weizenbaum, 1964-66) & artful deception

- use pattern matching and substitution to fake understanding

**ELIZA effect**: humans are inclined to see computers as humans
  e.g. when ATM says “thank you”

An AI problem is **AI-complete** if any AI problem is mechanically reducible to it (at least as hard as any other).

E.g. Natural Language Understanding

*The town councilors refused to give the demonstrators a permit because they feared violence.*

*Who feared violence?*  
T. Winograd
ELIZA (Weizenbaum, 1964-66) & artful deception

- use pattern matching and substitution to fake understanding

**ELIZA effect**: humans are inclined to see computers as humans
e.g. when ATM says “thank you”

An AI problem is **AI-complete** if any AI problem is mechanically
reducible to it (at least as hard as any other).

E.g. Natural Language Understanding

*The town councilors refused to give the demonstrators a permit because they advocated violence.*

*Who advocated violence?*  

T. Winograd
ELIZA (Weizenbaum, 1964-66) & artful deception

- use pattern matching and substitution to fake understanding

**ELIZA effect**: humans are inclined to see computers as humans
e.g. when ATM says “thank you”

An AI problem is **AI-complete** if any AI problem is mechanically reducible to it (at least as hard as any other).

E.g. Natural Language Understanding

*The town councilors refused to give the demonstrators a permit because they advocated violence.*

Who advocated violence?  

---

**Caution**: Programs may appear to work better than they do

**Siri rage** (Urban dictionary):

*When you get enraged because Siri just doesn't get it.*
Locating intelligence (black box)

Poole & Mackworth

Intelligence: \((\text{abilities, goals, ...}, \text{experience}) \mapsto \text{action}\)
Locating intelligence (black box)

Intelligence: (abilities, goals, ..., experience) $\mapsto$ action

Turing test: what to say $\rightsquigarrow$ what to do

Poole & Mackworth
Between agent and environment

<table>
<thead>
<tr>
<th>agent</th>
<th>environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>program</td>
<td>data</td>
</tr>
<tr>
<td>Cognitive Revolution</td>
<td>Big Data</td>
</tr>
<tr>
<td>hard-wired</td>
<td>experienced</td>
</tr>
<tr>
<td>rationalist</td>
<td>empiricist</td>
</tr>
<tr>
<td>nativist</td>
<td>behaviorist</td>
</tr>
<tr>
<td>innate</td>
<td>tabula rasa</td>
</tr>
<tr>
<td>nature</td>
<td>nurture</td>
</tr>
</tbody>
</table>

Turing machine & specialized automaton

Learning (from environment)
trial & error: “data as oil”
## Between agent and environment

<table>
<thead>
<tr>
<th>agent</th>
<th>environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>program</td>
<td>data</td>
</tr>
<tr>
<td>Cognitive Revolution</td>
<td>Big Data</td>
</tr>
<tr>
<td>hard-wired</td>
<td>experienced</td>
</tr>
<tr>
<td>rationalist</td>
<td>empiricist</td>
</tr>
<tr>
<td>nativist</td>
<td>behaviorist</td>
</tr>
<tr>
<td>innate</td>
<td>tabula rasa</td>
</tr>
<tr>
<td>nature</td>
<td>nurture</td>
</tr>
</tbody>
</table>

Turing machine &
specialized automaton

Learning (from environment)
trial & error: “data as oil”

Moving target: changing agent & environment
e.g. change in state
So having stated the motivation for working on AI and the challenges, how should we actually make progress?

Given a complex real-world task, at the end of the day, we need to write some code (and possibly build some hardware too). But there is a huge chasm between the real-world task and code.

A useful paradigm for solving complex tasks is to break them up into two stages. The first stage is modeling, whereby messy real-world tasks are converted into clean formal tasks called models. The second stage is algorithms, where we find efficient ways to solve these formal tasks.

Algorithms (example)

Formal task:
- **Input**: list $L = [x_1, \ldots, x_n]$ and a function $f: X \rightarrow \mathbb{R}$
- **Output**: $k$ highest-scoring elements

Example ($k = 2$):

$L$: A B C D

$f$:

3 2

7 1

Two algorithms:
- Scan through to find largest, scan through again to find the second largest, etc.
- Sort $L$ based on $f$, return first $k$ elements

Let's start with something that you're probably familiar with: algorithms. When you study algorithms, you are generally given a well-defined formal task, something specified with mathematical precision, and your goal is to solve the task. A solution either solves the formal task or it doesn't, and in general, there are many possible solutions with different computational trade-offs.

As an example, suppose you wanted to find the $k$ largest elements in a list of $L = [x_1, \ldots, x_n]$ according to given a scoring function $f$ that maps each element into a real-valued score.

Solving a formal task involves coming up with increasingly more efficient algorithms for solving the task.

unstructured information $\rightsquigarrow$ actionable knowledge

Demis Hassabis

www.theguardian.com/technology/2016/feb/16/demis-hassabis-artificial-intelligence-deepmind-alphago
Levels of intelligence

Search problems
Markov decision processes
Adversarial games
Constraint satisfaction problems
Bayesian networks

Reflex States Variables Logic
"Low-level intelligence" "High-level intelligence"

Machine learning

https://web.stanford.edu/class/cs221/ (Autumn 2017)
Levels of intelligence

Search problems
Markov decision processes
Adversarial games

Constraint satisfaction problems
Bayesian networks

Reflex
States
Variables
Logic

"Low-level intelligence"
"High-level intelligence"

Machine learning

https://web.stanford.edu/class/cs221/ (Autumn 2017)

A Pendulum Swung Too Far, Kenneth Church,
Linguistic Issues in Language Technology 2 (4), 2011

CS4060 focuses on the right side, logic
► what we can say (& can’t): knowledge representation
► what we can do (& can’t): automata
Resources (links in www.scss.tcd.ie/Tim.Fernando/KRA)

- *AI: Foundations of Computational Agents*
  Poole & Mackworth, 2nd edition (2017)

- *Knowledge Representation & Reasoning*
  Brachman & Levesque

- *Elements of Finite Model Theory*, L. Libkin

- *Algorithmic Model Theory*, E. Grädel

- Predication via Finite-State Methods, ESSLLI 2017 course