- Today: Models and beginning search
- Chapters 3 & 4 are assigned
- How to read the text
  – Skim the assignment
  – Read interesting/important parts
  – Listen to lectures; ask questions
  – Read (again?) the interesting / important parts
  – Ask questions at the beginning of class and at office hours

Unified view of AI as Model-based Reasoning

- What is reasoning?
  - Making a decision
  - Drawing a conclusion
  - Choosing an action
  - Developing an interpretation

- What is a “model?”
  - A stand-in for the real thing
  - Must be mathematically precise:
    quantifiable computational properties
  - NB in logic there is a different technical usage

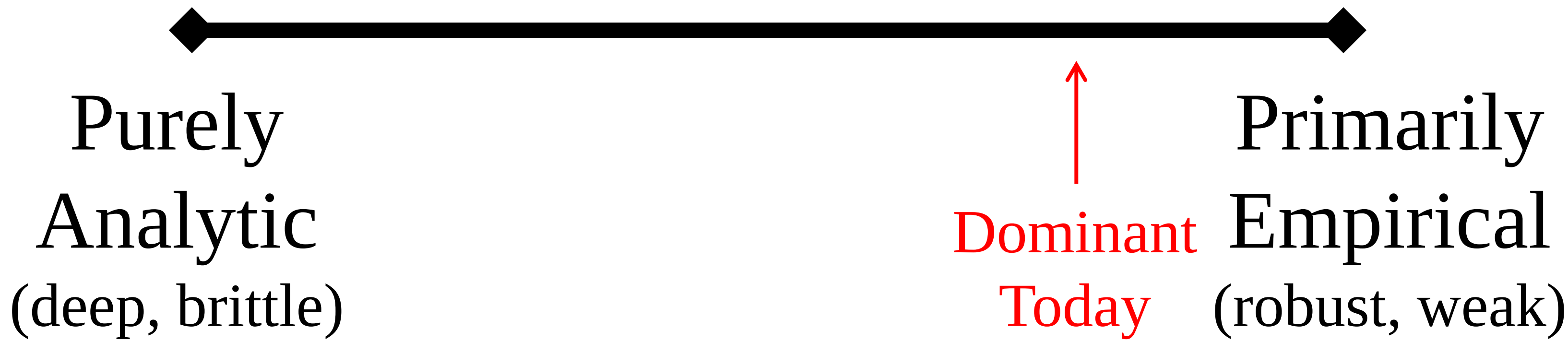
Inference with Computational Models

- Inference:
  explicit new justified beliefs
- Model design
  - Driven by task
  - As simple as possible
- Classification
  - From input features
  - Infer a class label
- Problem Solving
  - Decisions/actions (often a sequence)
  - Achieve a goal
- In a moment, the search example of cryptarithmetic

Designing a Model

- Distinctions to notice
  - Real world is infinitely subtle and complex
  - Ideally notice all and only necessary distinctions (leave out almost everything)
  - States / attributes / features
- Dynamics – change
  - Operators / successor states / transition function (R & N)
  - When: *preconditions*
  - How: *effects*
  - Often *change = world change* but not always
- Inference
  - change in explicit beliefs about world
  - True?
  - Useful?

AI Models



TODAY's AI

- Models are often empirically driven and statistically sophisticated
- Machine Learning plays an important role
- Weak prior knowledge / analytic commitments
- Relies on empirical evidence

Analytic / Emprical

- Models can be purely analytic
- Models cannot be purely empirical
- For us “analytic” means *purely* analytic;
  “empirical” means not purely analytic
- Empirical models always have some analytic component;
  they use some observations of the world

Why Use Empirical?

- Necessity:
  - The world's dynamics may be intrinsically stochastic
  - We do not fully understand its dynamics
- By Choice:
  - World dynamics are uncomfortably complex
  - Simpler model is sufficient for our purposes

Models
to predict coin flip outcome

- Factory produces 2-headed coins

  Analytic

- Coins are either 2-headed or 2-tailed

  Empirical w/ strong analytic component

  unknown model characteristics estimated from data

- Coins are somewhat biased

  Empirical, weaker analytic component

  quasi stochastic world

Examples of Current AI Systems
w/ Empirical Models employing Machine Learning

- Vision
  - Object identification
  - Object localization
  - • • •
- Natural language processing (NLP)
  - Syntactic parsing
  - Semantic role labeling
  - • • •

Vision

Object Identification / Localization

- CalTech 4
- Subset of CalTech 101 (now CalTech 256)
- Serious problems but still employed / influential
- Many (several 10s to many hundreds) of (standardized) training images
- 101 categories

Training Images

Faces
98%

Motorbikes
99%

Airplanes
88%

Cars
99%

What is this?

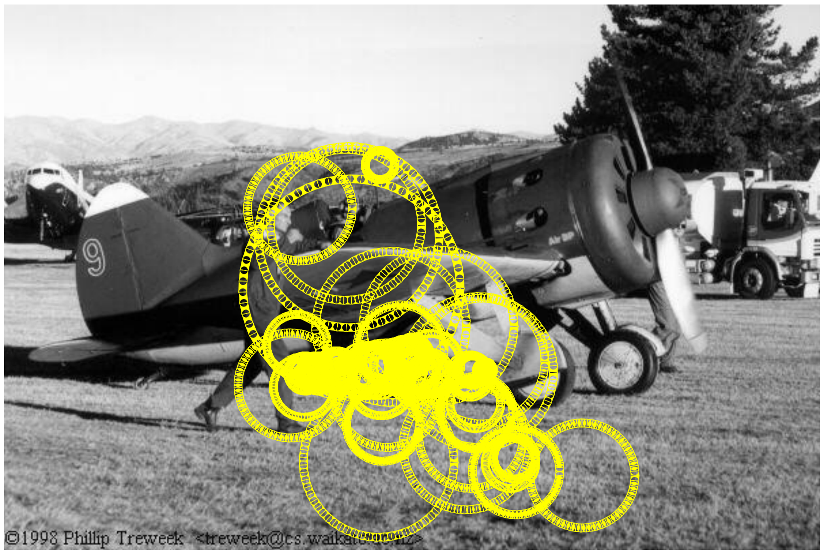
"It's a car"

What are these?

"It's a face"

"It's a motorbike"

What is this?



"It's a face"

Accuracy is high, but…

simple tasks
fragile decisions
non-human mistakes

John drove his car to buy groceries.

John drove his sister to buy groceries.

John drove his car to commit suicide.

John drove his sister to commit suicide.

Is Common Sense Necessary?

Common sense $\rightarrow$ Adequate analytic model

What is the alternative?   An empirical model

Statistical NLP

- Quite successful (dominant today)
- Little linguistic knowledge
- Minimal world knowledge
- Large Annotated Corpus
- Find / Appreciate statistically significant patterns
- Interpret new S accordingly

John drove his car to buy groceries

			Parse Tree
John	driver [A0]	buyer [A0]	**(S1 (S (NP (NNP John))**
drove	V: drive		**(VP (VBD drove)**
his	thing in motion [A1]		**(NP (PRP$ his)**
car			**(NN car)**
to			**(S (VP (TO to)**
buy		V: buy	**(VP (VB buy)**
groceries		thing bought [A1]	**(NP (NNS groceries)))))))))**

John drove his sister to buy groceries

			Parse Tree
John	driver [A0]		(S1 (S (NP (NNP John))
drove	V: drive		(VP (VBD drove)
his	thing in motion [A1]	buyer [A0]	(S (NP (PRP$ his)
sister			(NN sister))
to			(VP (TO to)
buy		V: buy	(VP (VB buy)
groceries		thing bought [A1]	(NP (NNS groceries)))))))

John drove his car to commit suicide

			Parse Tree
John	driver [A0]	causal agent [A0]	(S1 (S (NP (NNP John))
drove	V: drive		(VP (VBD drove)
his	thing in motion [A1]		(NP (PRP$ his)
car			(NN car)
to			(S (VP (TO to)
commit		V: commit	(VP (VB commit)
suicide		entity committed [A1]	(NP (NN suicide))))))))

John drove his sister to commit suicide

Word	drive	commit	Parse Tree
John	driver [A0]		(S1 (S (NP (NNP John))
drove	V: drive		(VP (VBD drove)
his	thing in motion [A1]	causal agent [A0]	(S (NP (PRP$ his)
sister			(NN sister))
to			(VP (TO to)
commit		V: commit	(VP (VB commit)
suicide		entity committed [A1]	(NP (NN suicide))))))))

Last night I shot an elephant in my pajamas

		Parse Tree
Last	temporal [AM-TMP]	(S1 (S (NP (JJ Last)
night		(NN night))
I	shooter [A0]	(NP (PRP I))
shot	V: shoot	(VP (VBD shot)
an	corpse [A1]	(NP (DT an)
elephant		(NN elephant))
in		(PP (IN in)
my		(NP (PRP$ my)
pajamas		(NNS pajamas))))))

Statistical / Probabilistic Model

- Tuned with a (large) training set
  - Corpus of many annotated sentences, e.g., "John shot Bill in the leg"
  - Images with labels (& perhaps regions / parts)
- Learned restrictions / preferences are primarily syntactic
- Little prior meaning or semantic structure

- Non-human-intuitive behavior / performance
- Often low confidence (even when correct)
- BUT…Analytic models in these tasks usually perform much worse

Search for Problem Solving

- Analytic models
  - Capture the dynamics of the world
  - Very simple
  - Design them ourselves (programming)
- Defines a space of possibilities
- Rely on search to find interesting sequences

Search for Problem Solving

Analytic Models

(more standard than textbook)

N.B.: Lectures supercede textbook!

• World State (state)	attributes
• Initial State	single or distribution
• Operators (actions)	mechanism of change
• Action (path step, action)	instantiated operator
• Preconditions (transition	of operators
• Effects function)	of operators
• Goal (goal test)	partial state
• Cost model	action $\rightarrow \Re$

EXAMPLES

Romanian Road Trips (text)

Cryptarithmetic

```
  send
+ more
 money
```

```
forty
  ten
+ ten
sixty
```

Find a letter/digit substitution that forms a natural & correct arithmetic expression

Analytic or Empirical Model?

Analytic Model

More intelligence / less search

More search / less intelligence

→

Computational Model for Search

- Representation scheme for states
- Define operators
- Generate successor states
  - evaluate preconditions
  - apply operators
- Select which unexamined state to examine next (expand)
- Recognize goal achievement

Cryptarithmetic Operators

- What will they do?
- How many do we need?

State Representation for Cryptarithmetic

- Our state representation must support the operators

State Representation for Cryptarithmetic

One possibility (of many):

((unassigned letters)
 (unassigned numbers)
 (assignments))

((d e m n o r s y)
 (0 1 2 3 4 5 6 7 8 9)
 ())

Operator Definition

Crypt((A,B,C)):

PC: A not empty, B not empty

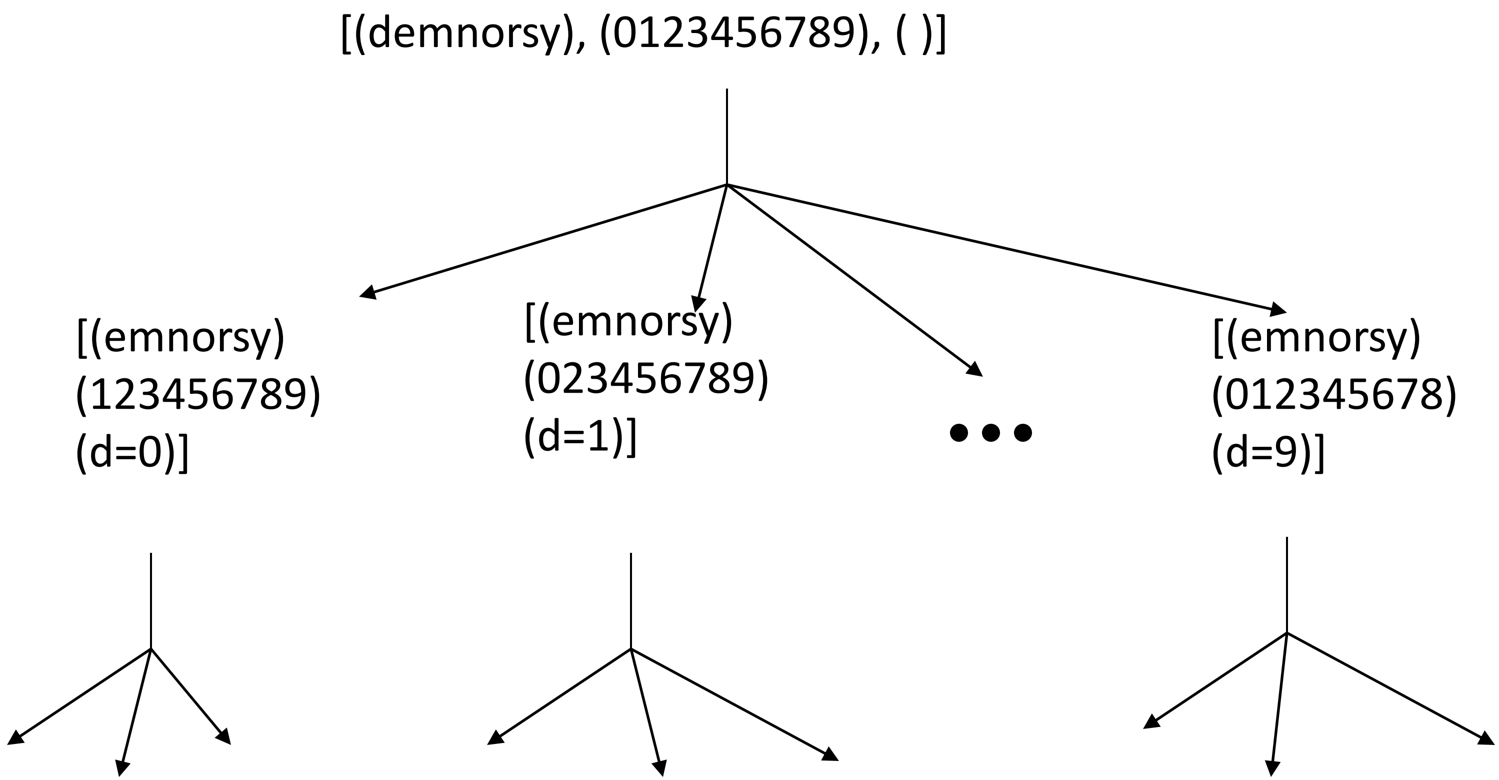
Effects:

$a \leftarrow \text{First}(A)$

For each $b \in B$

Collect (Remove(A,a), Remove(B,b), $C \cup (a=b)$)

Search Tree



Why a tree?

Do all searches result in trees?

Goal Test for Cryptarithmetic

The list (unassigned letters) is empty

The arithmetic is correct

No leading zeros

Model for Problem Solving with Search

- Search tree is defined by Initial State and Operators
  - Mathematical notion of “defined” – not algorithmic!
  - Model must support an Algorithm
- In code, build / represent only what is needed
  - Do NOT generate the entire search tree
  - Do NOT save everything generated
  - Generate states incrementally
  - Forget anything not needed for future

EXPAND(Node N) finds children

```
EXPAND(Node N):
For every Operator O
    For every way of satisfying PC(O) in N
        Collect distinct Effects(O) applied to N;
Return the collection;
```

Constructing operators can be much trickier than Russell & Norvig would have us believe...

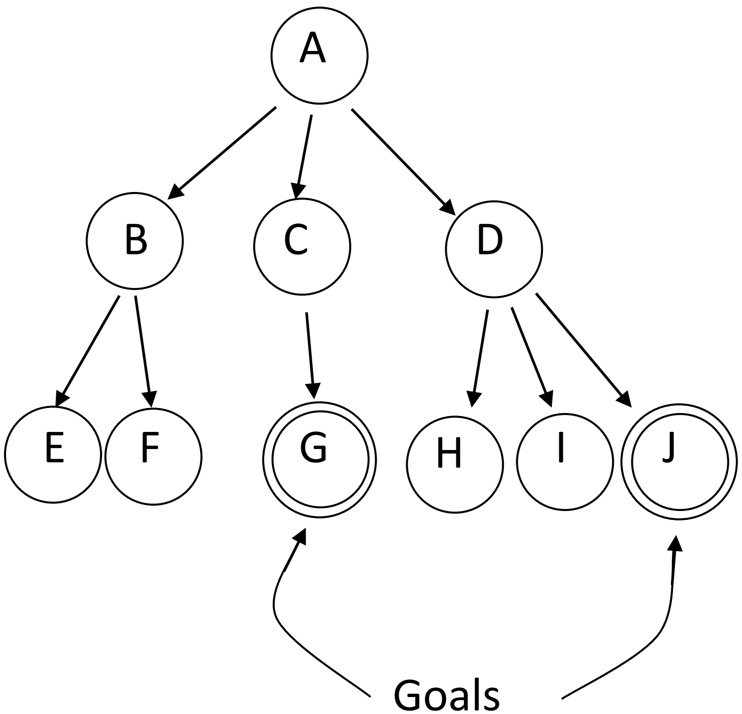
Generic Search Function

```
SEARCH(Problem P, Queuing Function QF):
  local:    n    /* current node */
            q    /* nodes to explore */
  q ← singleton of Initial_State(P);
  Loop:
    if q = () return failure;
    n ← Pop(q);
    if n Solves P return n;
    q ← QF(q, Expand(n));
  end
```

Depth First: `QF(old, new): Append(new, old);`

Breadth First: `QF(old, new): Append(old, new);`

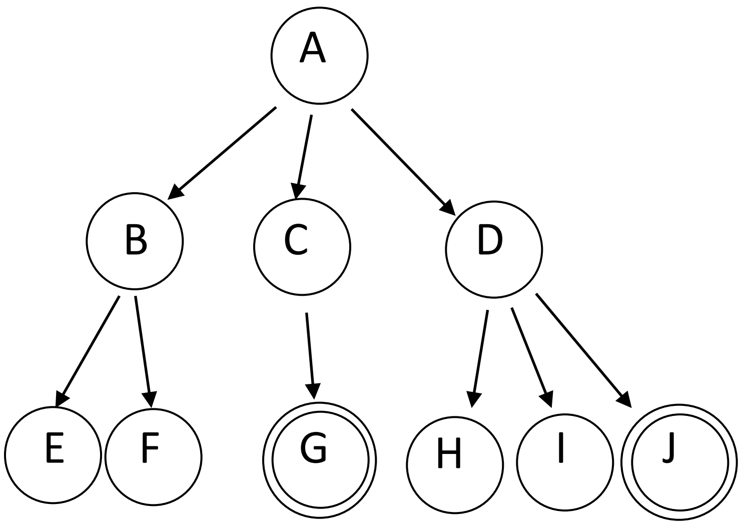
Sample Search Tree



Depth First

Breadth First

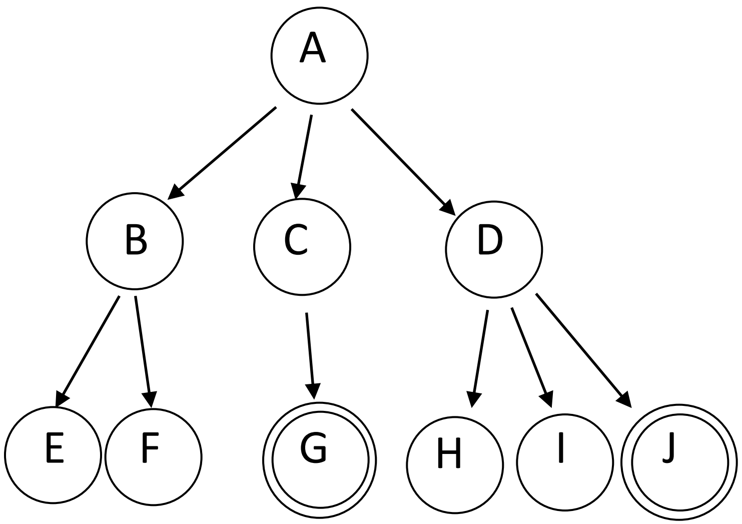
Sample Search Tree



Depth First

n	q (between iterations)
-	(A)
A	(B C D)
B	(E F C D)
E	(F C D)
F	(C D)
C	(G D)
G	(D)

Sample Search Tree



Breadth First

n	q
-	(A)
A	(B C D)
B	(C D E F)
C	(D E F G)
D	(E F G H I J)
E	(F G H I J)

•••