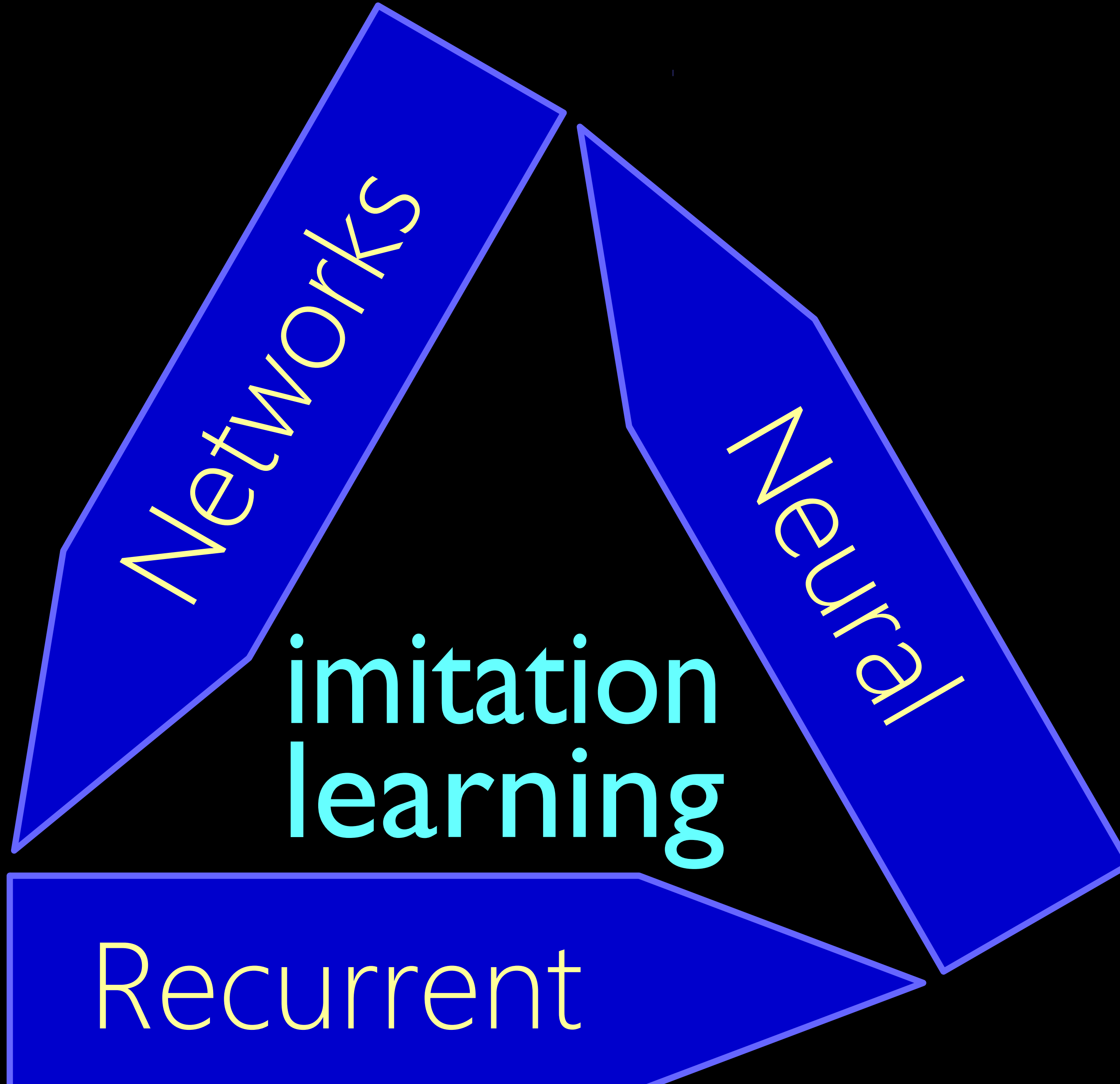
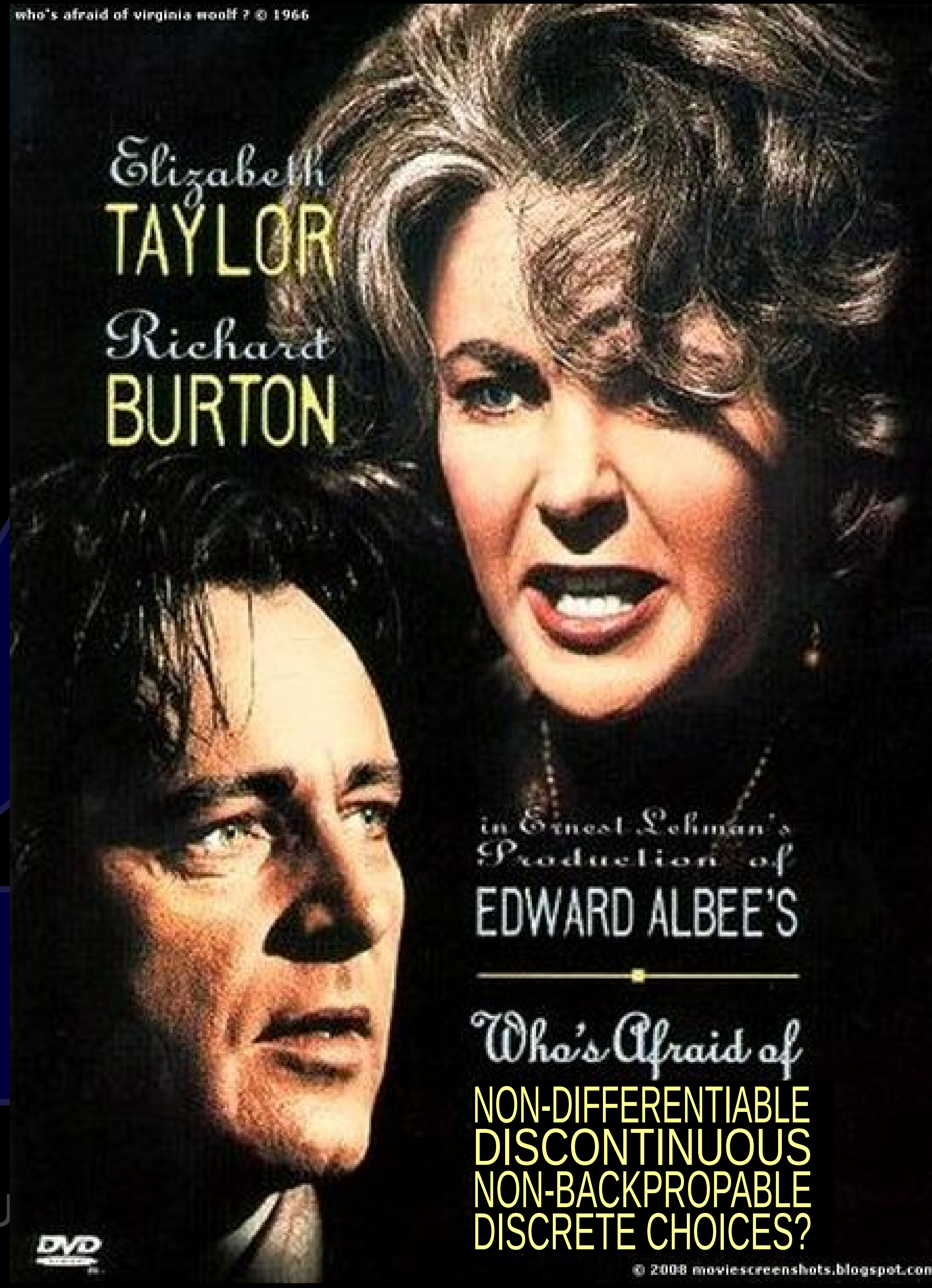


Hal Daumé III | University of Maryland | me@hal3.name | @haldaume3


who's afraid of virginia woolf ? © 1966
Elizabeth
TAYLOR
Richard
BURTON
in Ernest Lehman's
Production of
EDWARD ALBEE'S
Who's Afraid of
NON-DIFFERENTIABLE
DISCONTINUOUS
NON-BACKPROPABLE
DISCRETE CHOICES?
DVD
© 2008 moviescreenshots.blogspot.com

Examples of ~~structured~~ joint prediction

Sequence labeling

```
x = the monster ate the sandwich
y = Dt     Nn     Vb  Dt     Nn

x = Yesterday I traveled to Lille
y =     -     PER   -       -  LOC
```

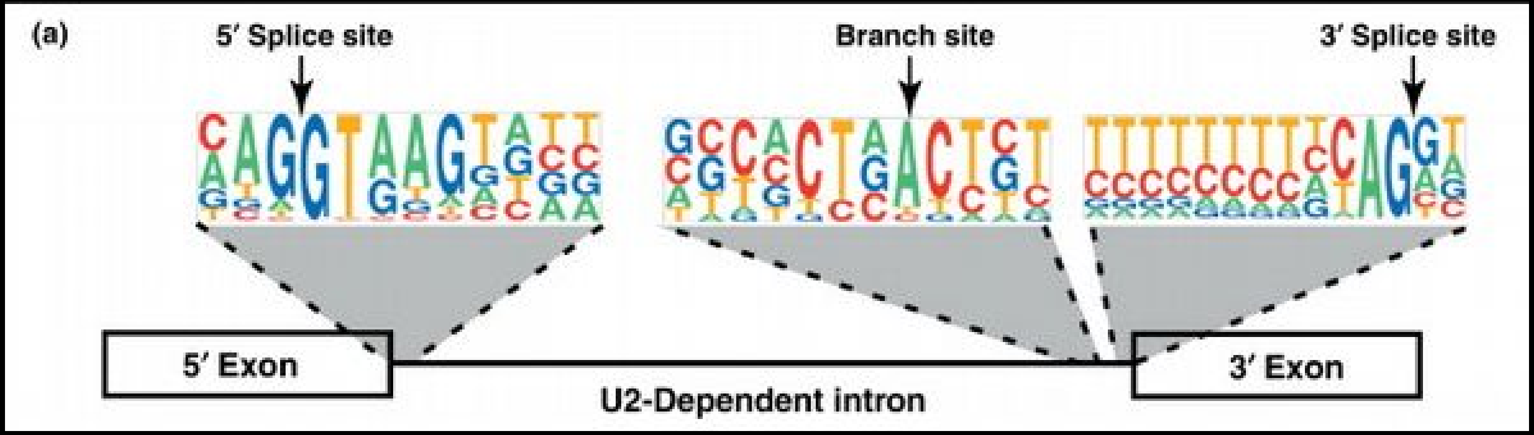
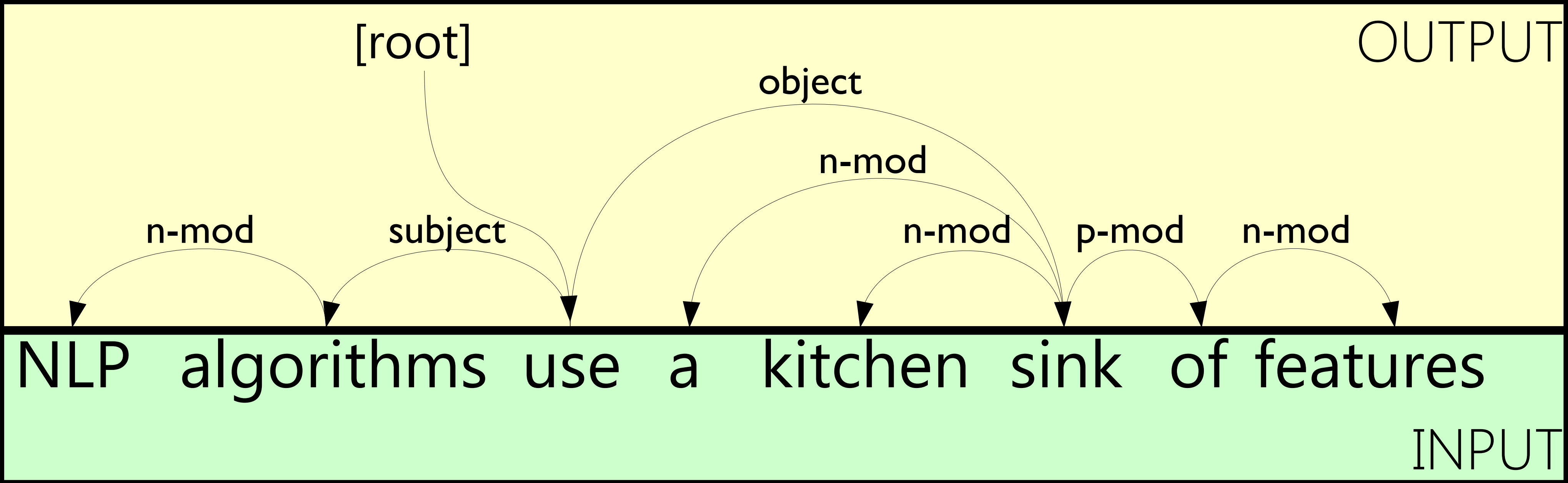


image credit: Richard Padgett

Natural language parsing



Segmentation

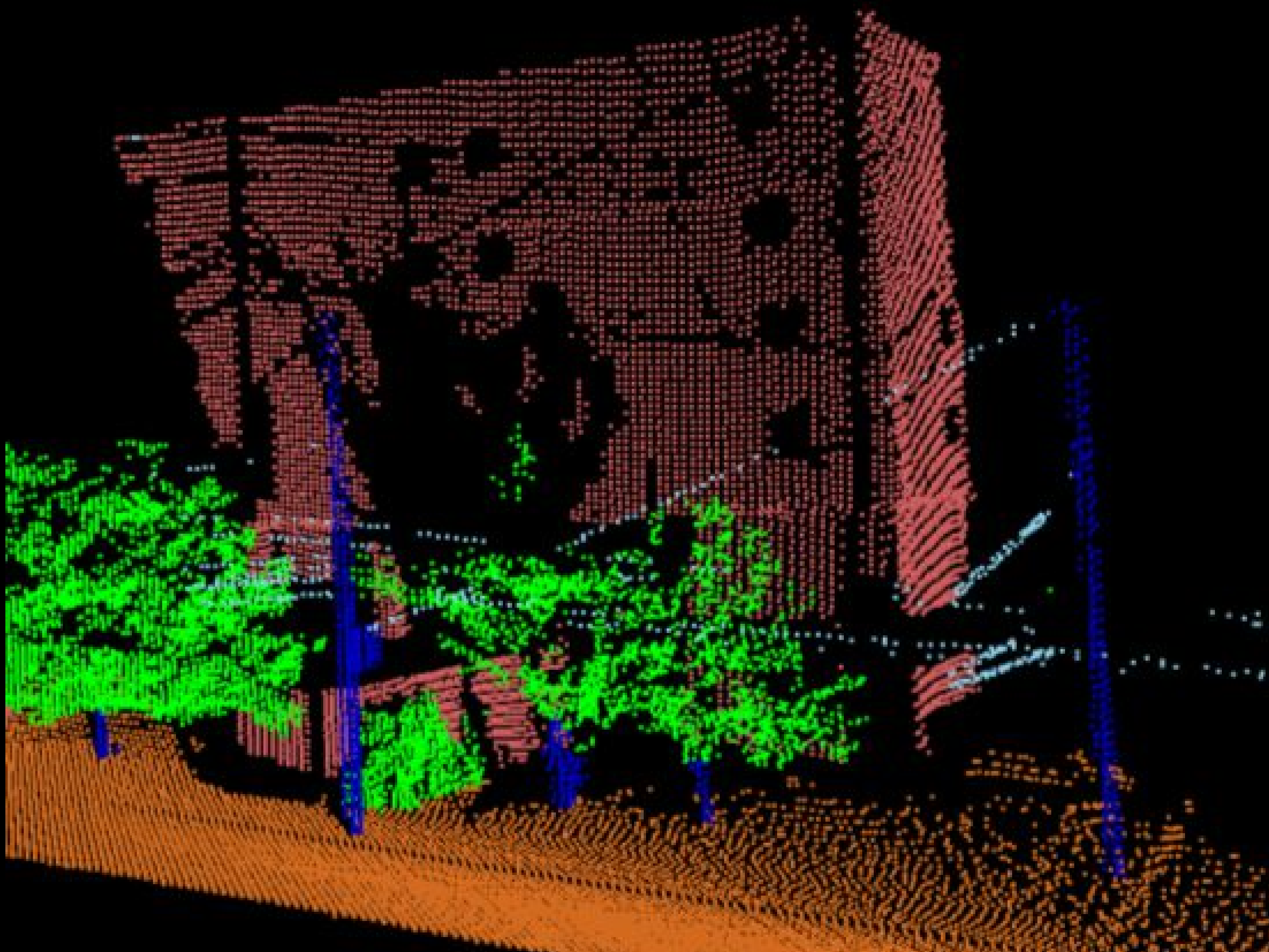


image credit: Daniel Muñoz

Simultaneous (machine) interpretation

Nuremburg Trials

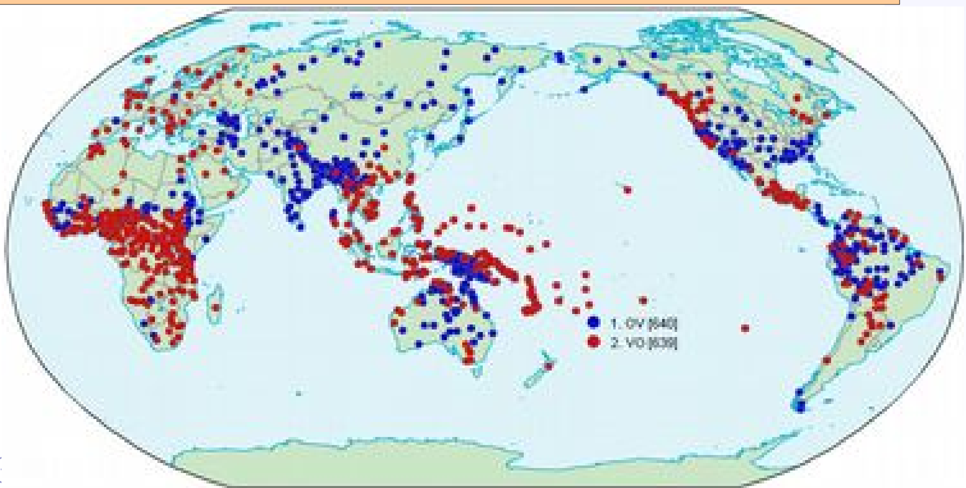
- Dozens of defendants
- Judges from four nations (three languages)
- Status quo: speak, then translate
- After Nuremberg, simultaneous translations became the norm
- Long wait → bad conversation

Why simultaneous interpretation is hard

- Human languages have vastly different word orders
  - About half are OV, the other half are VO
  - This comes with a lot more baggage than just verb-final

Running (German/English) Example:

Ich	bin	mit	dem	Zug	nach	Ulm	gefahren
I	am	with the		train	to	Ulm	traveled
I	*(……waiting……)*						traveled by train to Ulm



Model for interpretation decisions

- We have a set of actions (predict / translate)
  - Wait
  - Predict clause-verb
  - Predict next word
  - Commit ("speak")
- In a changing environment (state)
  - The words we've seen so far
  - Our models' internal predictions
- With well-defined notions of:
  - Reward (or loss) at the end
  - Optimal action at training time

Example of interpretation trajectory

Big Challenges:

No supervision about when to "wait"

Complicated loss/reward functions

I	…ug	nach	Ulm	gefahren	
I	…ne	train	to	Ulm	traveled
I	… *waiting……)*				traveled by train to Ulm

Back to the original problem...

- How to optimize a discrete, joint loss?

- Input: $x \in X$
- Truth: $y \in Y(x)$
- Outputs: $Y(x)$
- Predicted: $\hat{y} \in Y(x)$
- Loss: $\mathrm{loss}(y, \hat{y})$
- Data: $(x,y) \sim D$

I	can	can	a	can
Pro	Md	Vb	Dt	Nn
Pro	Md	Md	Dt	Vb
Pro	Md	Md	Dt	Nn
Pro	Md	Nn	Dt	Md
Pro	Md	Nn	Dt	Vb
Pro	Md	Nn	Dt	Nn
Pro	Md	Vb	Dt	Md
Pro	Md	Vb	Dt	Vb

Back to the original problem...

- How to optimize a discrete, joint loss?

- Input: $x \in X$
- Truth: $y \in Y(x)$
- Outputs: $Y(x)$
- Predicted: $\hat{y} \in Y(x)$
- Loss: $\text{loss}(y, \hat{y})$
- Data: $(x,y) \sim D$

Goal:

find $h \in H$

such that $h(x) \in Y(x)$

minimizing

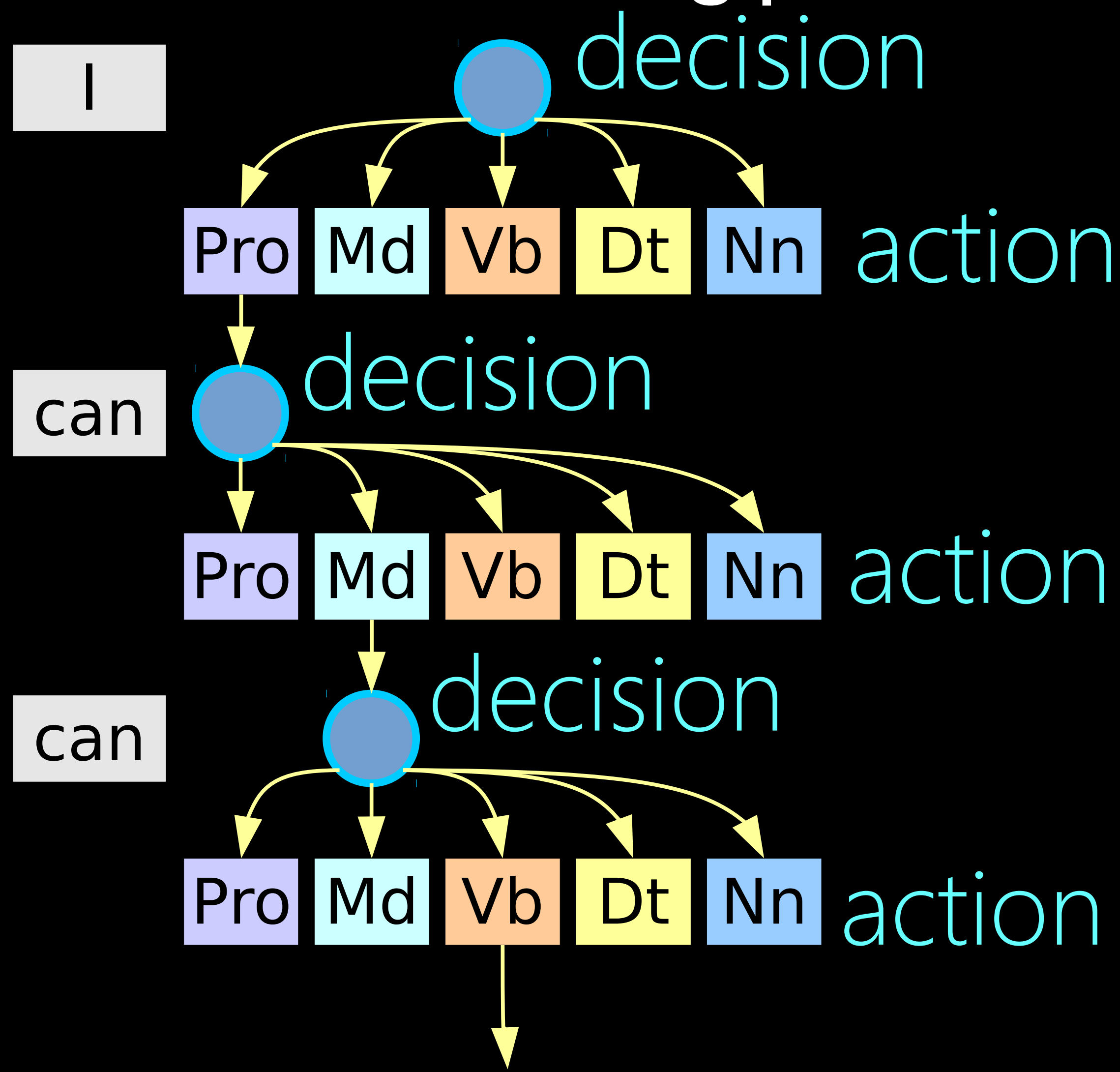
$$E_{(x,y)\sim D} \left[\text{loss}(y, h(x)) \right]$$

based on N samples

$(x_n, y_n) \sim D$

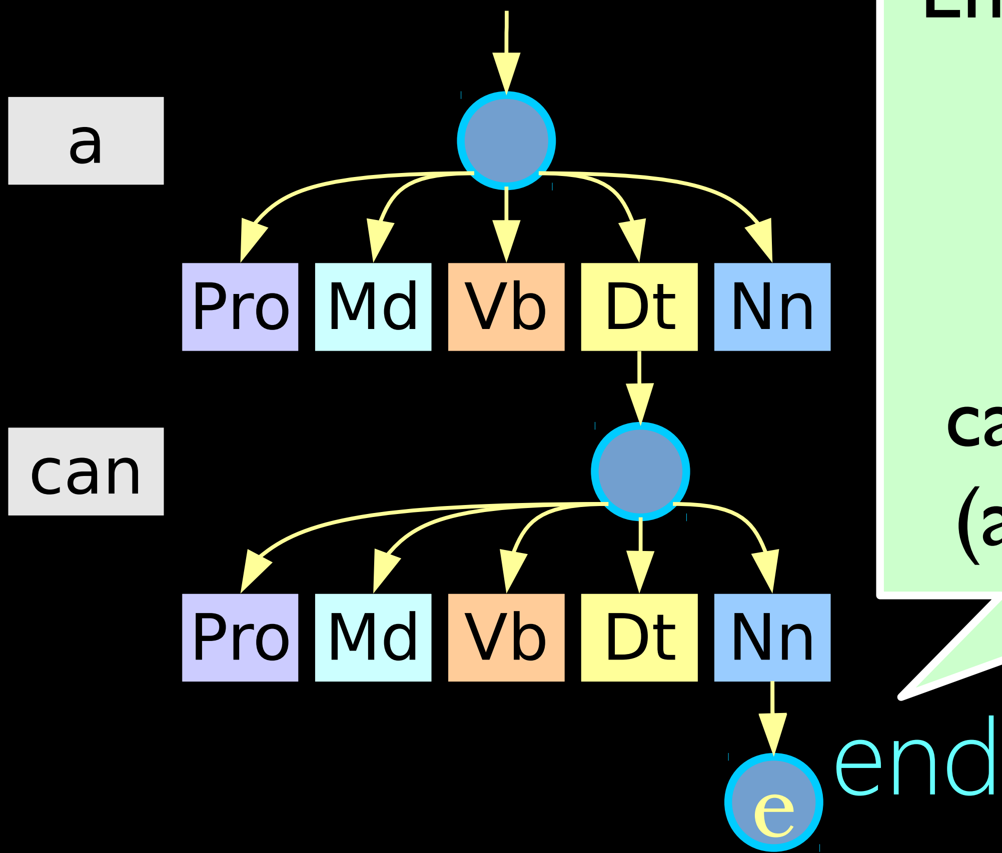
Search spaces

- When $\mathbf{y}$ decomposes in an ordered manner, a sequential decision making process emerges



Search spaces

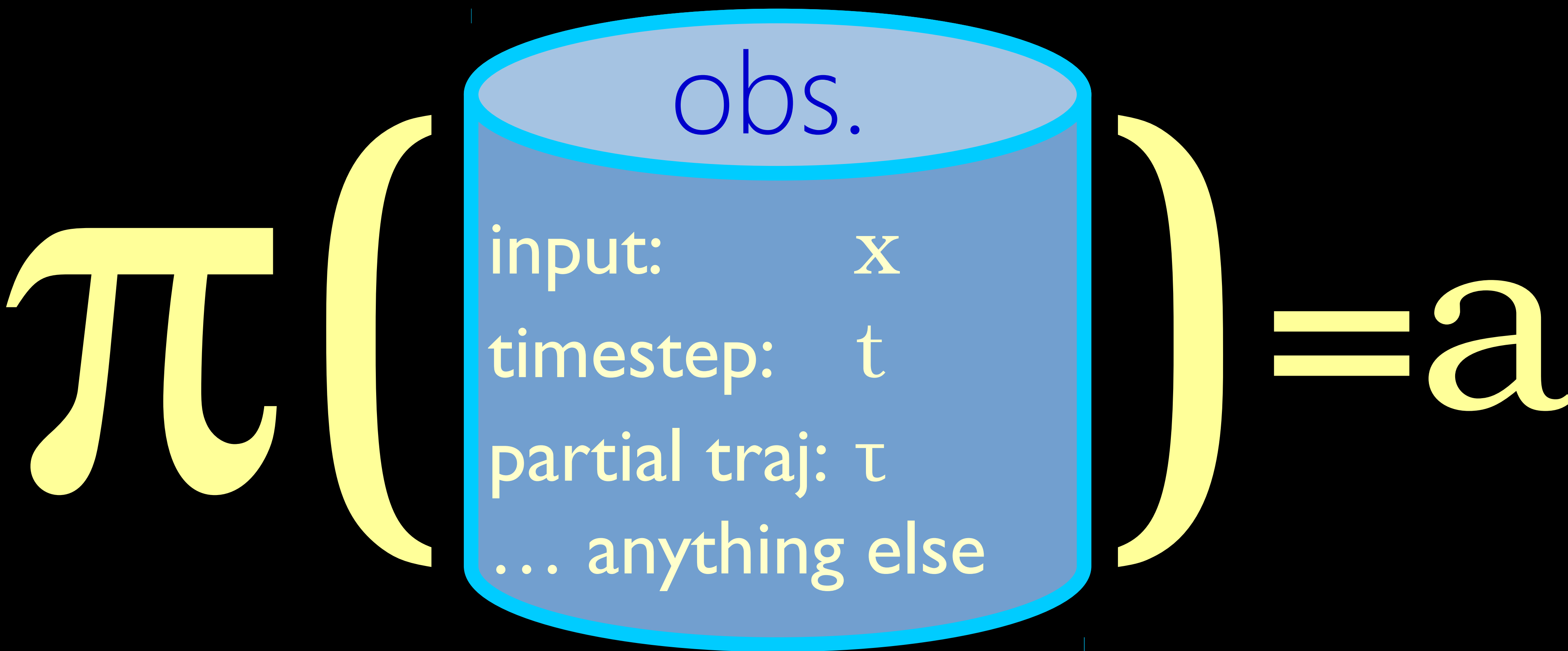
- When y decomposes in an ordered manner, a sequential decision making process emerges



Encodes an output
$\hat{y} = \hat{y}(e)$
from which
$\mathrm{loss}(y, \hat{y})$
can be computed
(at training time)

Policies

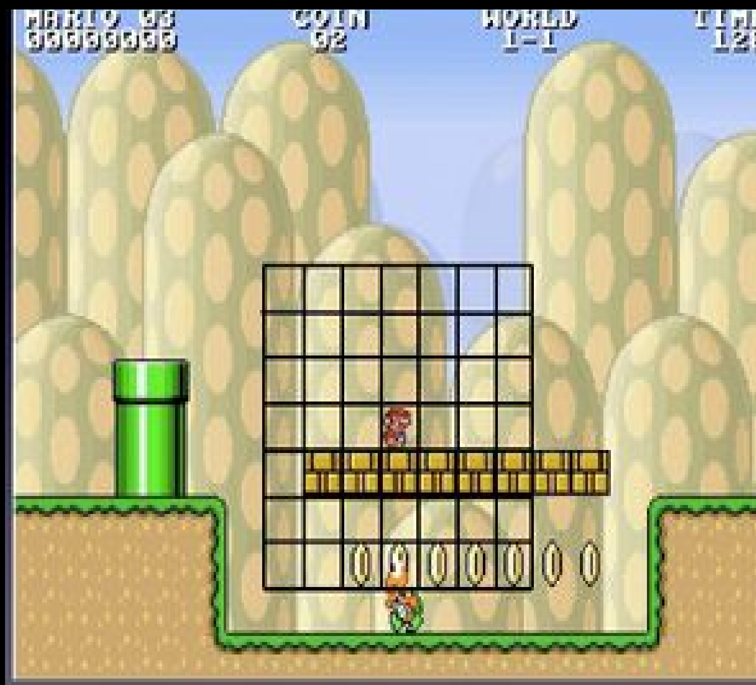
- A policy maps observations to actions



An analogy from playing Mario

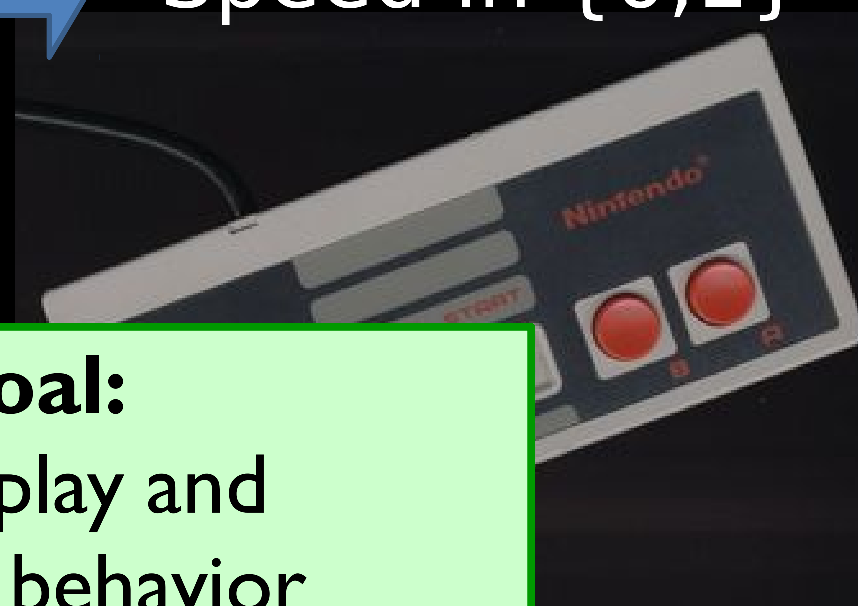
From Mario AI competition 2009

Input:



Output:

Jump in {0,1}
Right in {0,1}
Left in {0,1}
Speed in {0,1}



High level goal:
Watch an expert play and
learn to mimic her behavior

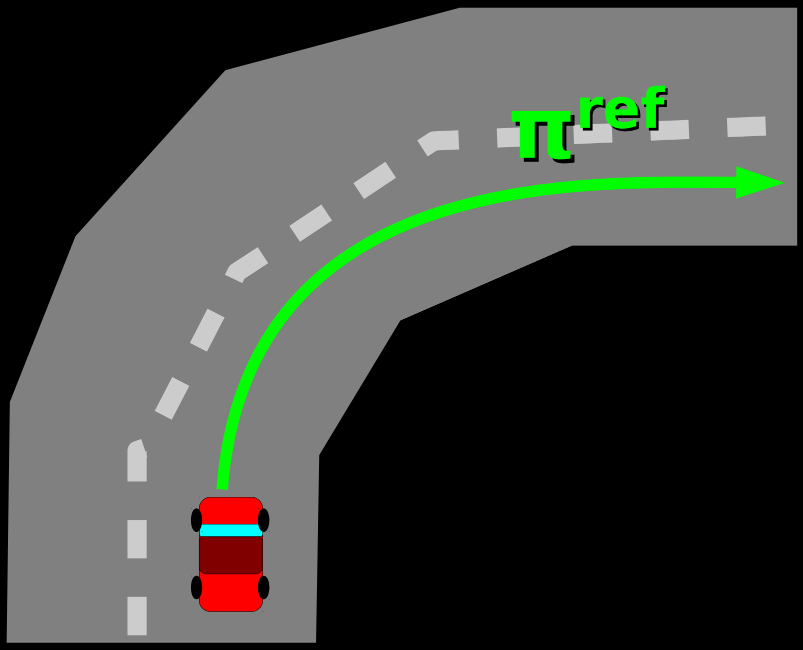
Training (expert)

Video credit: Stéphane Ross, Geoff Gordon and Drew Bagnell

Warm-up: Supervised learning

1. Collect trajectories from expert π^{ref}
2. Store as dataset $\mathbf{D} = \{ (o, \pi^{ref}(o,y)) \mid o \sim \pi^{ref} \}$
3. Train classifier π on $\mathbf{D}$

- **Let π play the game!**



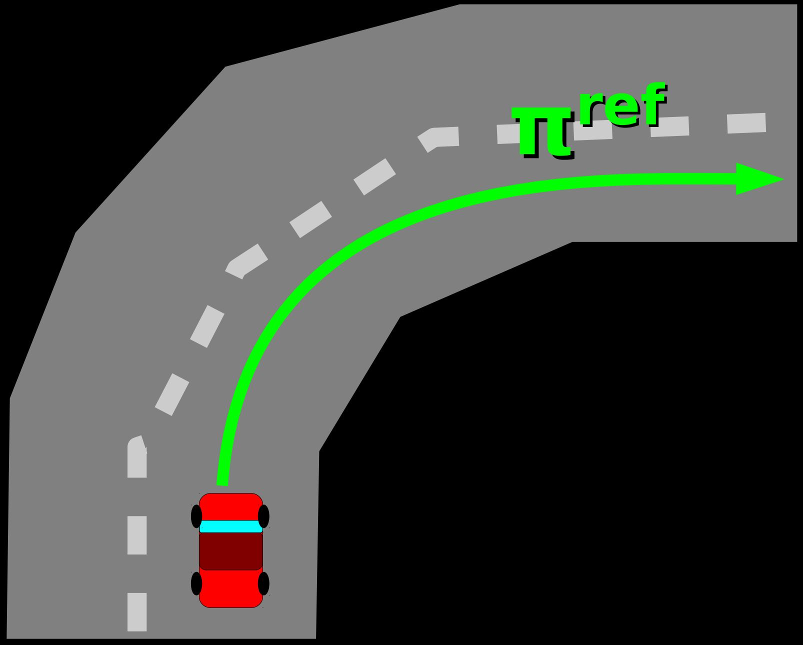
Test-time execution (sup. learning)

Video credit: Stéphane Ross, Geoff Gordon and Drew Bagnell

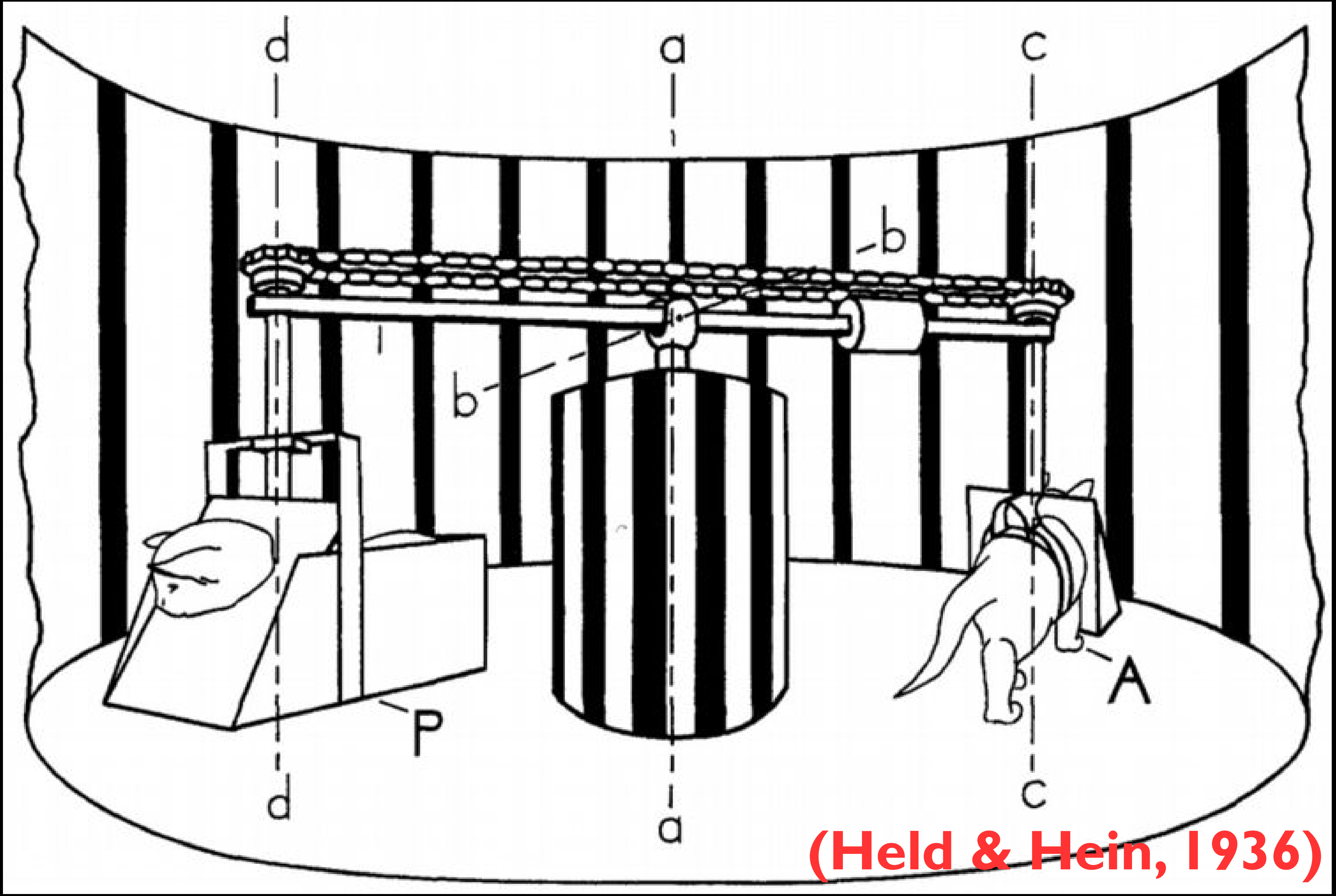
What's the (biggest) failure mode?

The expert never gets stuck next to pipes

⇒ Classifier doesn't learn to recover!



Kittens, revisited.



(Held & Hein, 1936)

Warm-up II: Imitation learning

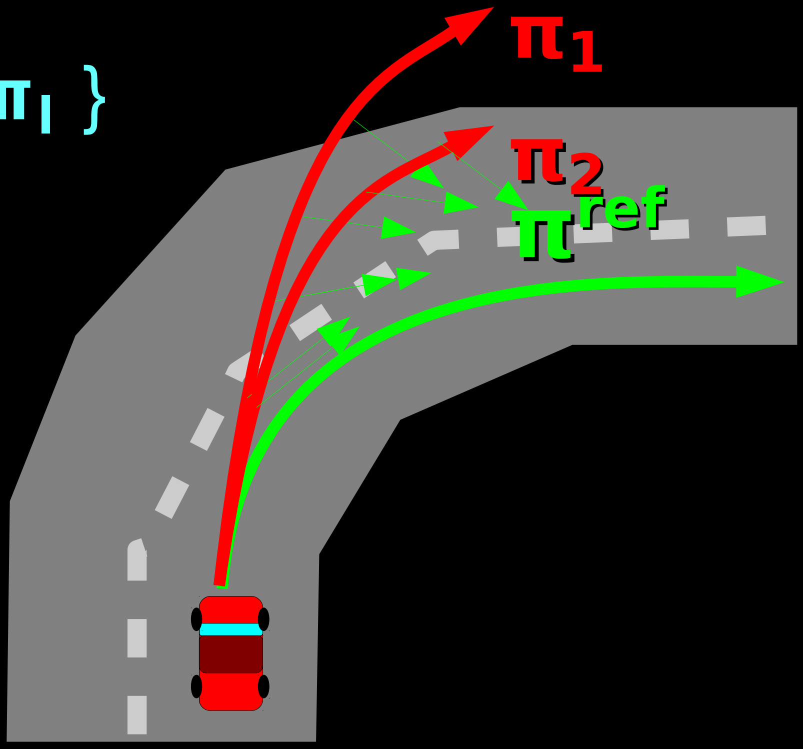
1. Collect trajectories from expert $\boldsymbol{\pi}^{\mathrm{ref}}$
2. Dataset $\mathbf{D_0} = \{\ (\ o, \boldsymbol{\pi}^{\mathrm{ref}}(o,y)\ )\ |\ o \sim \boldsymbol{\pi}^{\mathrm{ref}}\ \}$
3. Train $\boldsymbol{\pi_1}$ on $\mathbf{D_0}$
4. Collect new trajectories from $\boldsymbol{\pi_1}$
   - But let the *expert* steer!
5. Dataset $\mathbf{D_1} = \{\ (\ o, \boldsymbol{\pi}^{\mathrm{ref}}(o,y)\ )\ |\ o \sim \boldsymbol{\pi_1}\ \}$
6. Train $\boldsymbol{\pi_2}$ on $\mathbf{D_0} \cup \mathbf{D_1}$

- In general:
  - $\mathbf{D_n} = \{\ (\ o, \boldsymbol{\pi}^{\mathrm{ref}}(o,y)\ )\ |\ o \sim \boldsymbol{\pi_n}\ \}$
  - Train $\boldsymbol{\pi_{n+1}}$ on $\bigcup_{i \le n} \mathbf{D_i}$

If $N = T \log T$,

$$\mathbf{L(\boldsymbol{\pi}_n) < T\,\epsilon_N + O(1)}$$

for some n



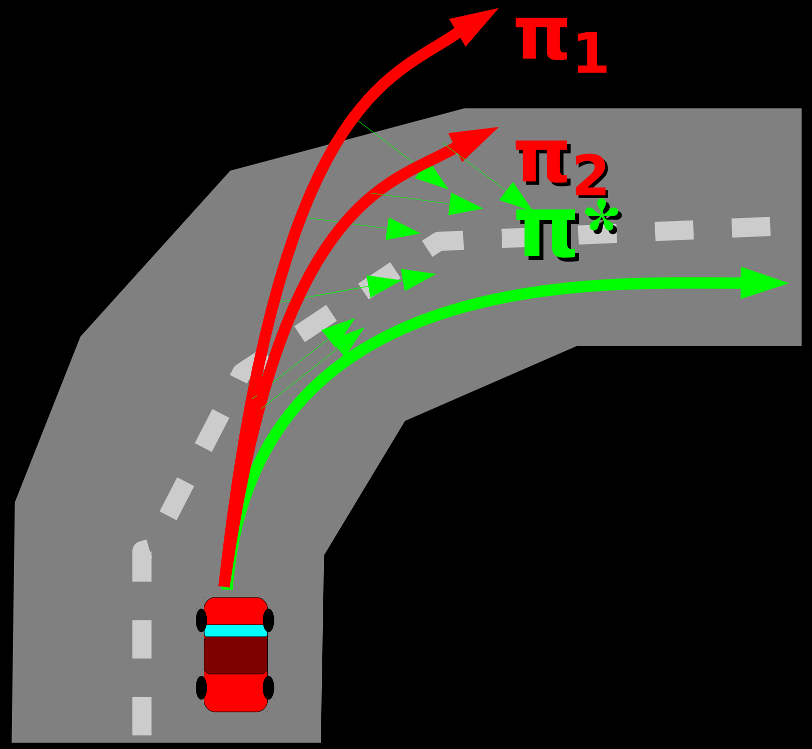
Test-time execution (DAgger)

Video credit: Stéphane Ross, Geoff Gordon and Drew Bagnell

What's the biggest failure mode?

Classifier only sees *right* versus *not-right*

- No notion of *better* or *worse*
- No *partial credit*
- Must have a single *target* answer

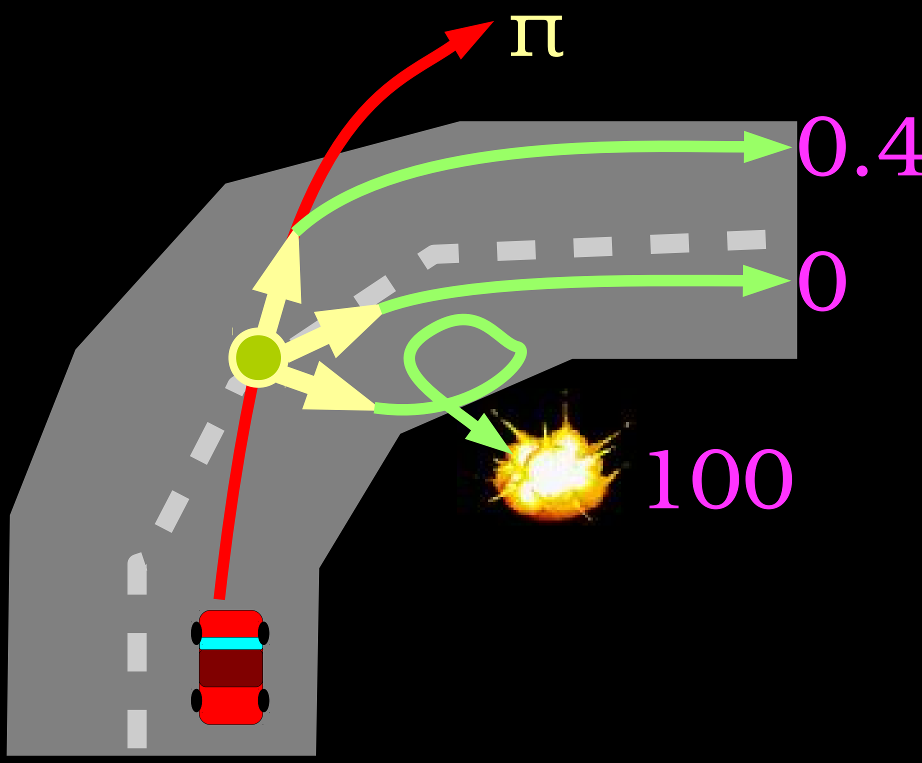


Learning to search: LOLS

1. Let learned policy π drive for t timesteps to obs. o
2. For each possible action a:
   - Take action a, and let expert π^{ref} drive the rest
   - Record the overall loss, c_a
3. Update π based on example:

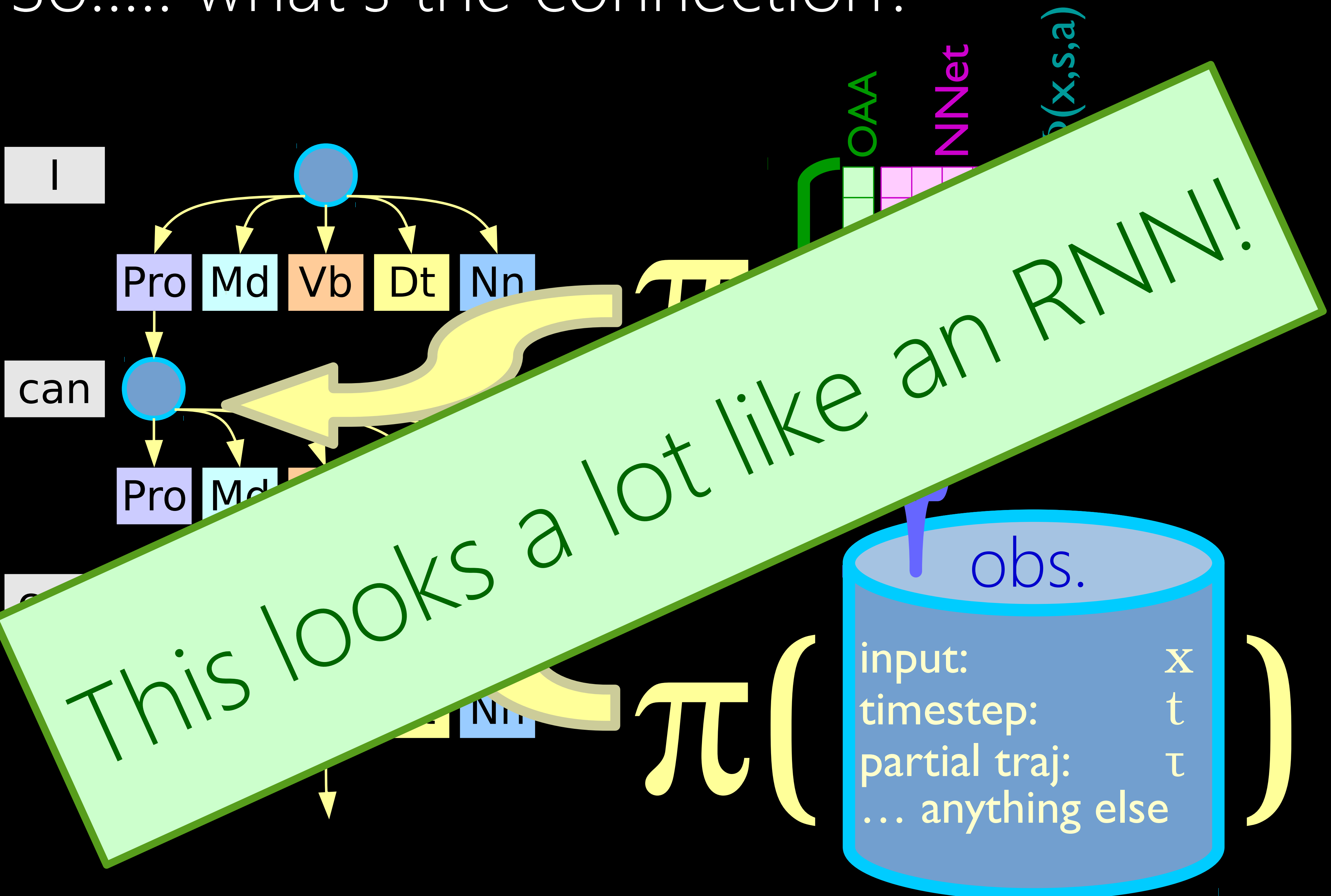
   $$(o, \langle c_1, c_2, \ldots, c_K \rangle)$$

4. Goto (1)



Side note: can also be run in "bandit" mode w/ sampling

So..... what's the connection?

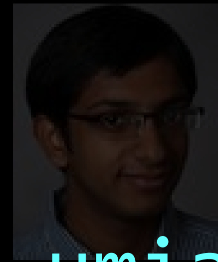


This looks a lot like an RNN!

Two quick results

- If you *don't* backprop through time:
  - POS tagging: no change
  - Named entity recognition: marginal improvement
  - Dependency parsing: 1% gain over strong baseline

ICPR '10 EMNLP'13 ICC'15
CVPR '11 EMNLP'14 Fusion'15
EMNLP'12 NIPS '14 EMNLP'15
NIPS '12 SLT '14 + more



Alekh Agarwal

Kai-Wei Chang

Akshay Krishnamurthy

John Langford

I am on the job market!
umiacs.umd.edu/~hhe

He He

- Simultaneous machine interpretation is a super fun problem and you should work on it!
- Not being able to backprop something isn't always the end of the world – you're not stuck with RL!
- RNN+LOLS mashup appears promising!

Thanks! Questions?