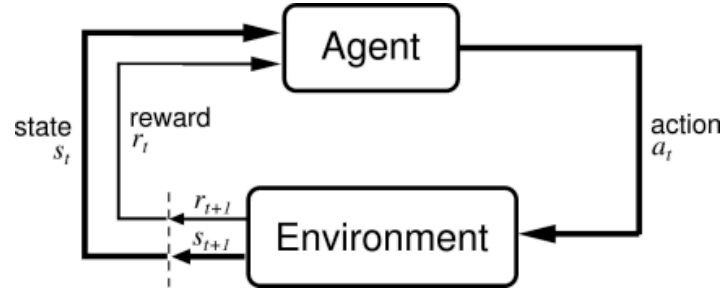
Interpretation and connections

Reinforcement learning

- states defined via features
- the agent is a classifier
- rewards?



(https://webdocs.cs.ualberta.ca/~sutton/book/ebook/node28.html)

Inverse reinforcement learning

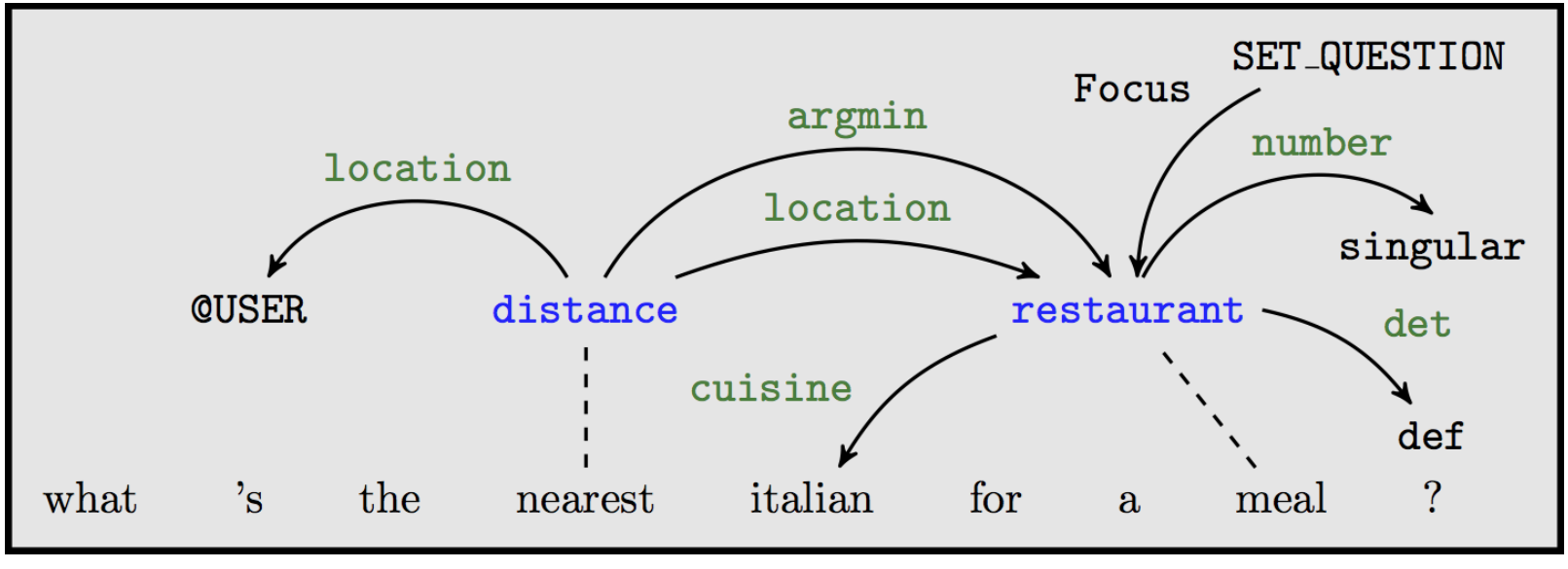
- we have the expert policy (inferred from the gold standard in the training data)
- we infer the per-action reward function (rollin/ out)

Replacing the expert policy in LoLS with a random (sub-optimal) one is RL (Chang et al., 2015 (https://arxiv.org/pdf/1502.02206.pdf))

Semi/Unsupervised learning

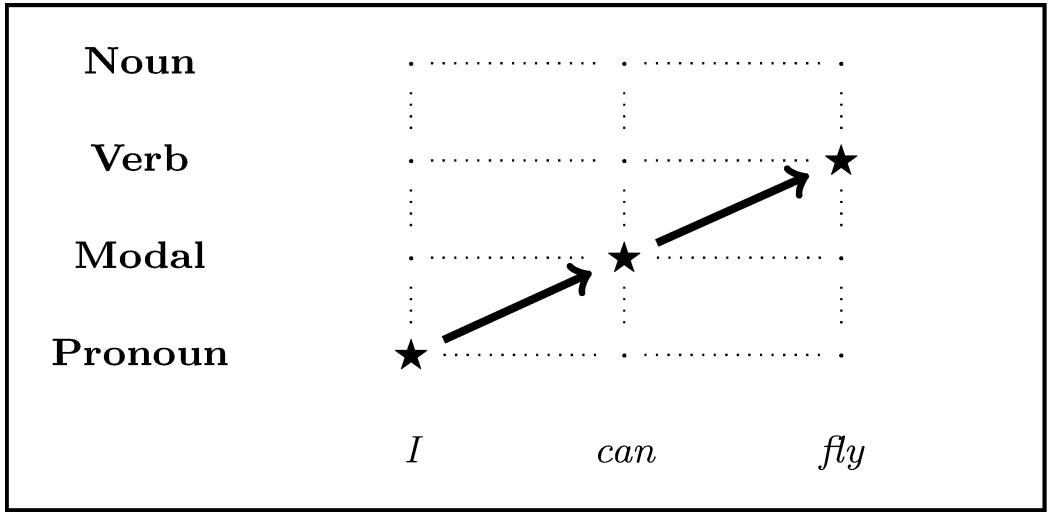
Learning with non-decompos able loss functions means

- no need to know the correct actions
- learn to predict them in order to minimize the loss



UNSEARN (Daumé III, 2009 (http://www.umiacs.umd.edu/~hal/docs/daume09unsearn.pdf)): Predict the structured output so that you can predict the input from it (auto-encoder!)

Negative data sampling



- Expert action sequence → positive example
- All other action sequences → negative examples
- Using all negative examples inefficient

Imitation learning: generate negative samples around the expert

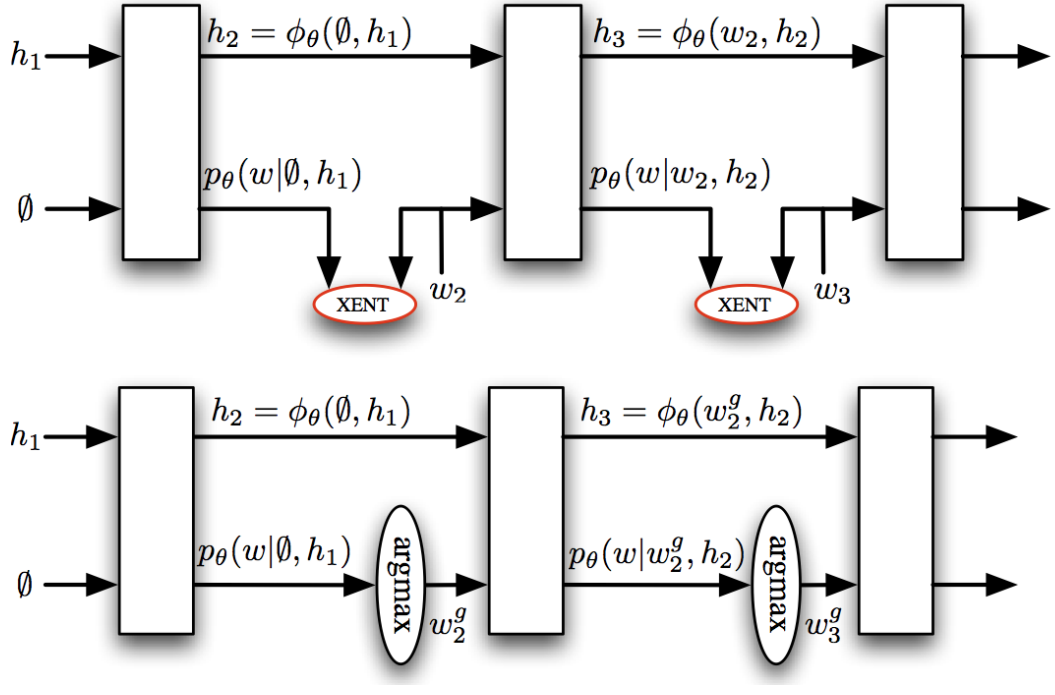
A form of **Adversarial training** (Ho and Ermon, 2016 (https://arxiv.org/abs/1606.03476))

Coaching

If the optimal action is difficult
to predict, the coach teaches
a good one that is easier
(He et al., 2012 (https://papers.nips.cc/paper/4545-imitation-learning-by-coaching.pdf))

(https://commons.wikimedia.org/wiki/File:US_Navy_091206-N-2013O-023_Sam_Givens,_a_player_for_the_Harlem_Ambassadors_basketball_team,_demonstrate

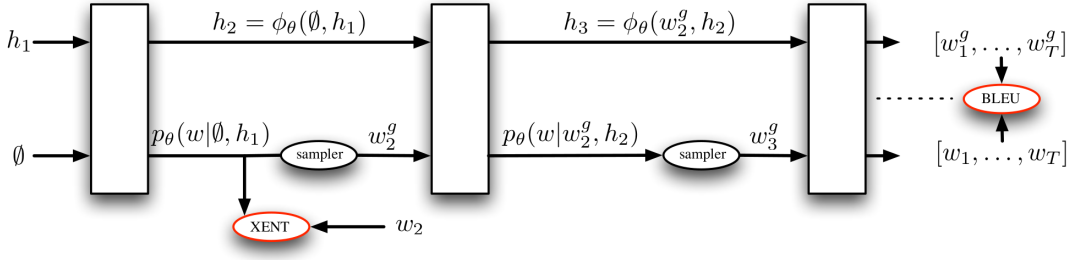
What about Recurrent Neural Networks?



They face similar problems:

- trained at the word rather than sentence level
- assume previous predictions are correct

Imitation learning and RNNs



- DAgger mixed rollins, similar to scheduled sampling (Bengio et al., 2015 (http://arxiv.org/abs/1506.03099))
- no rollouts, learn a regressor to estimate action costs
- end-to-end back propagation through the sequence
- MIXER (Ranzato et al., 2016 (https://arxiv.org/abs/1511.06732)): Mix REINFORCE-ment learning with imitation: we have the expert policy!

Summary so far

- basic concepts
  - loss function decomposability
  - expert policy
- imitation learning
  - rollin/outs
  - DAgger algorithm
  - DAgger with rollouts and LoLS
- connections and interpretations

After the break

- Applications:
  - dependency parsing
  - natural language generation
  - semantic parsing
- Practical advice
  - making things faster
  - debugging

Break!