# Hybrid neuro-symbolic reasoning

https://neuralreasoning.github.io/

Presented by Vuong Le

# The two main approaches in Image QA

- Neuro-symbolic reasoning
  - Parse the question into a “program” of small logical inference steps
  - Learn the inference steps as *neural modules*
  - Use and reuse the modules for different programs
  - \+ Explicit and interpretable
  - \+ Close to human’s logical inference
  - \+ **Strongly support generalization**
  - \- Brittle, cannot recover from mistakes
  - \- Struggling with nuances of language and visual context
  - \- *Leon Bottou: Reasoning needs not to be logical inferences*
- **Compositional reasoning**

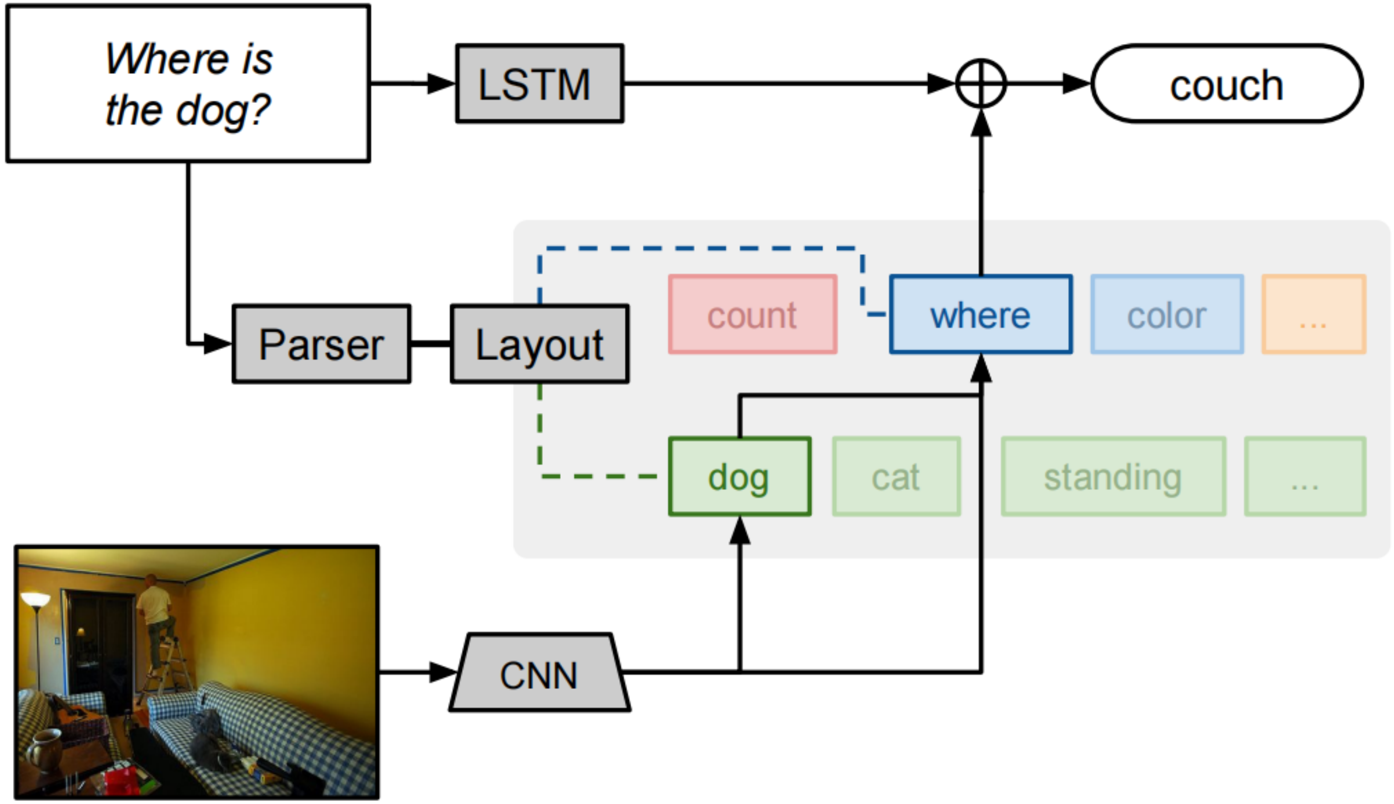
*what color is the vase?*

```
classify[color](
  attend[vase])
```

green (green)

# Neural Module Networks

- NLP parser to build program
- The layout consists of modules which are learnable sub-networks
- Use attention as key compositional operator

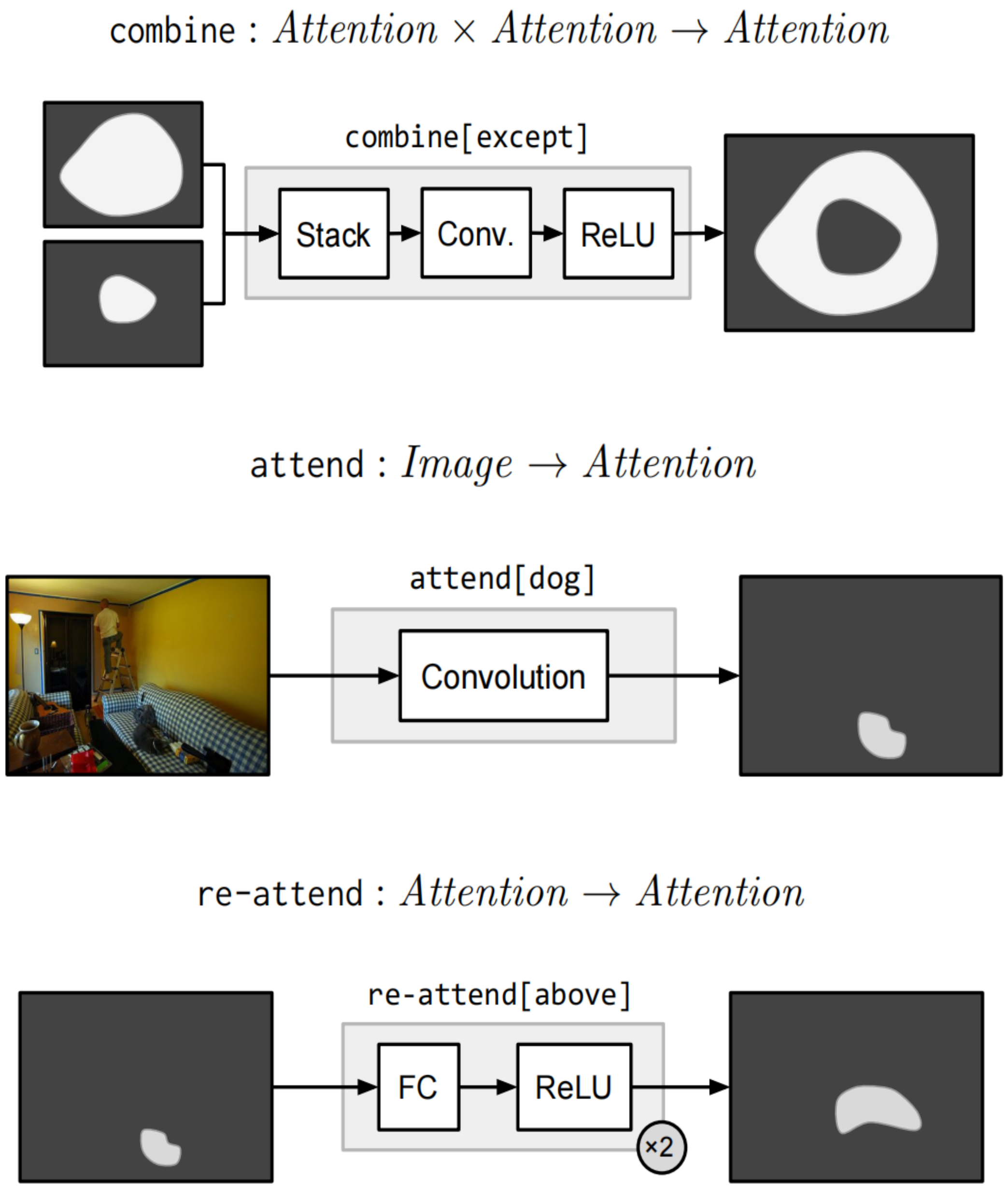





Neural Module Networks, Andreas et.al CVPR 16

# Modules

- attend[c] has weights distinct for each c to produce a heatmap
- re-attend[c] is MLP mapping from one attention to another.
- combine[c] merges two attentions
- into a single attention.



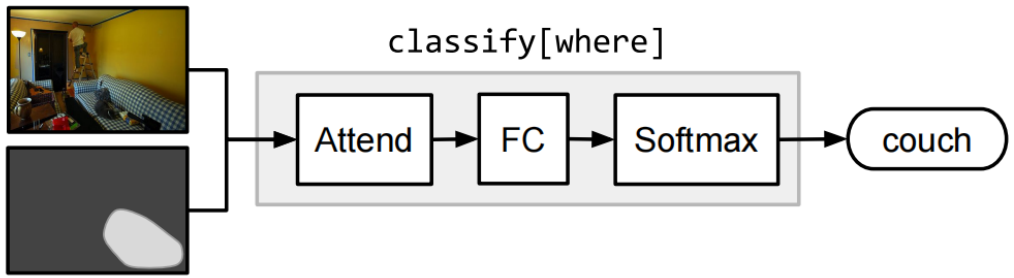



Neural Module Networks, Andreas et.al CVPR 16

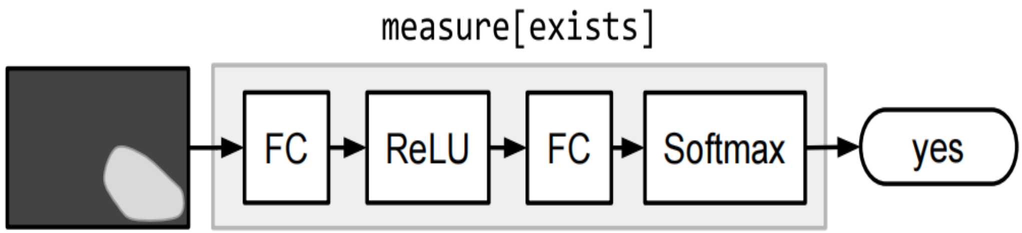
# Modules

- classify[c] takes an attention and the input image and maps them to a distribution over labels.
- measure[c] takes an attention alone and maps it to a distribution over count labels

$\texttt{classify} : \mathit{Image} \times \mathit{Attention} \to \mathit{Label}$



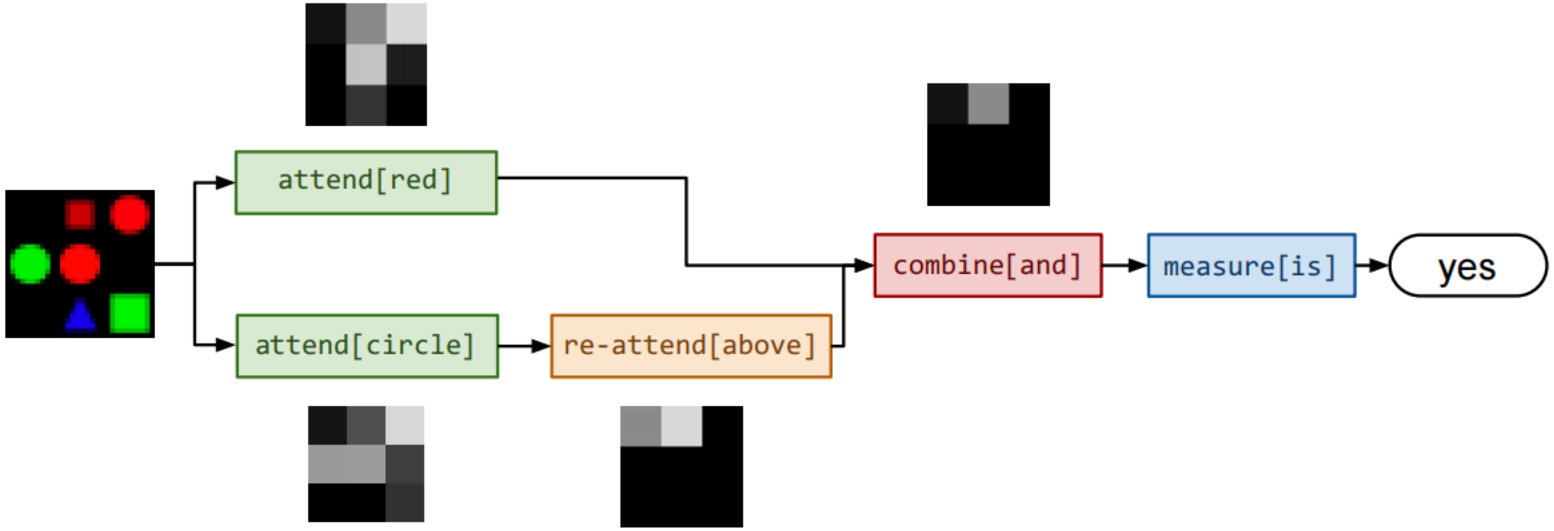

$\texttt{measure} : \mathit{Attention} \to \mathit{Label}$



# Parsing

- Stanford parser: create grammatical dependency tree
- Forming the layout
  - Leaves become attend modules
  - Internal nodes become re-atten or combine
  - Root nodes become classify or measure depend on the question type

# Neural Module Networks – example

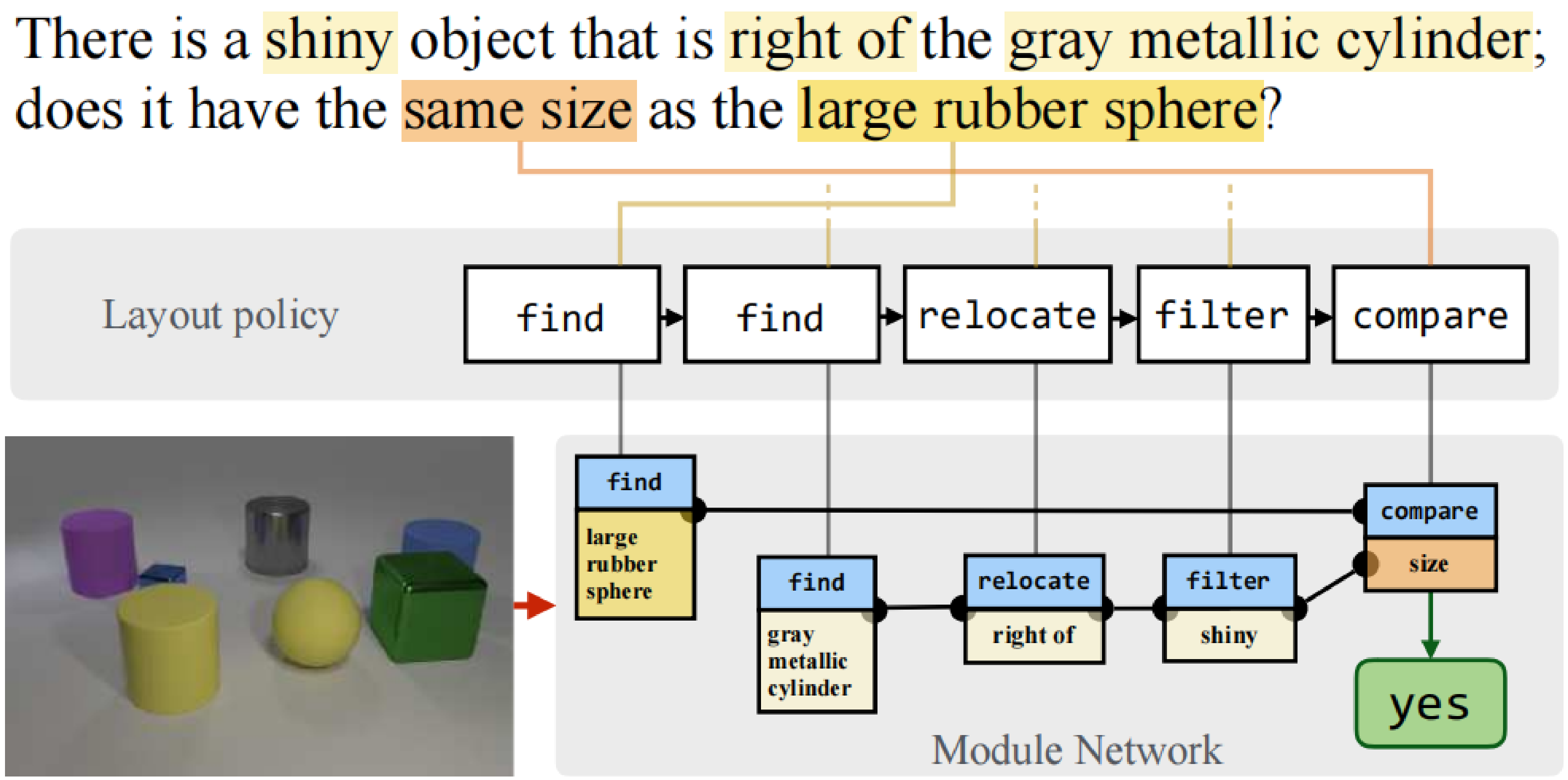



*Is there a red shape above a circle?*

→ Relying on an off-the-shelf parser. What if it makes a mistake?
Can the two steps be connected?

# End-to-End Module Networks

- Construct the program internally
- The two parts are jointly learnable

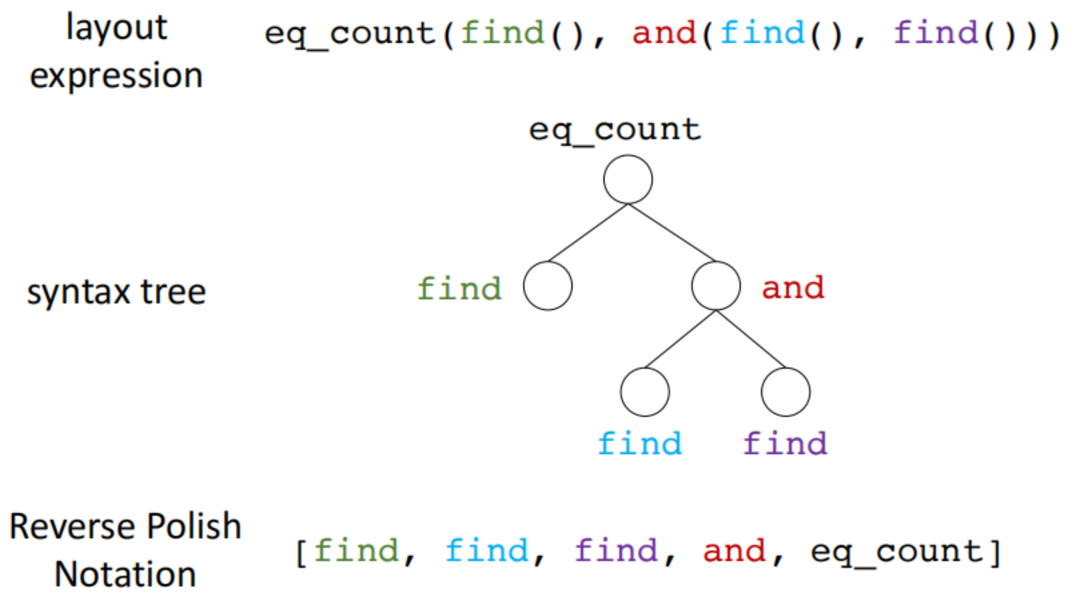





End-to-End Module Networks , Hu et.al., ICCV 17

# Layout policy

- A layout can be linearized into a sequence
- Then a layout prediction turns into seq-2-seq problem
- And can be done by an RNN encoder-decoder arch.



# End-to-End Module Nets

- Layout policy $p(l|q;\theta)$
- QA loss according to such policy $\tilde{L}(\theta, l; q, I)$
- End-to-end loss $L(\theta) = E_{l\sim p(l|q;\theta)}[\tilde{L}(\theta, l; q, I)]$

  - This loss is not fully differentiable as $l$ is discrete

  →Policy gradient for non-diff parts, estimated through MC sampling

  - Still a very hard problem as the two parts are more or less independent.

  → Direct supervision of $p(l|q;\theta)$ using some expert policy

# Combine the two main reasoning approaches

- Neuro-symbolic reasoning vs Compositional reasoning
  - \+ Explicit and interpretable
  - \+ Close to human's logical inference
  - \+ Strongly support generalization
  - \- Brittle, cannot recover from mistakes
  - \- Struggling with nuances of language and visual context

→Can we combine the two?

  →Process questions into a series of symbolic instructions

  →Use the instructions for guide the compositional reasoning process

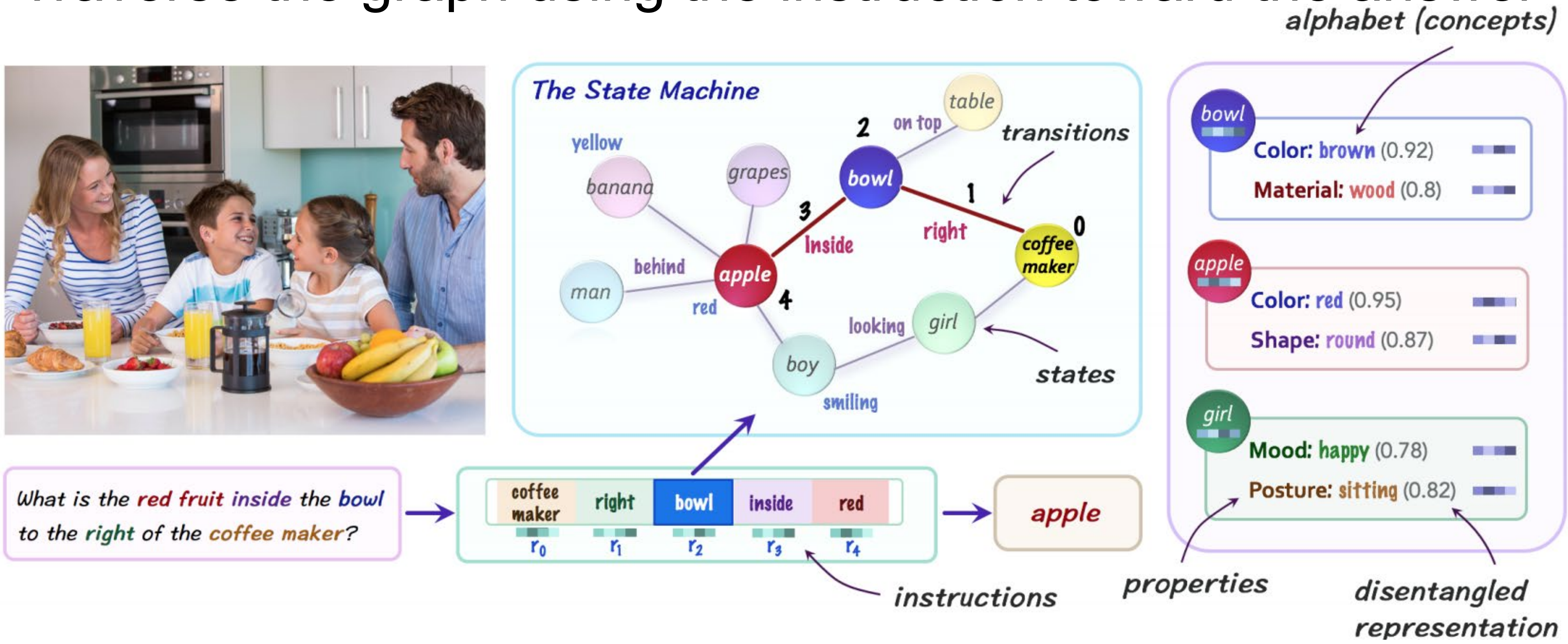
*what color is the vase?*

```
classify[color](
  attend[vase])
```

green (green)

# Neural State Machine

- Generate a scene graph from image
- Translate question into a series of instructions
- Traverse the graph using the instruction toward the answer




Neural State Machine, Hudson and Manning, NeurIPS 2019

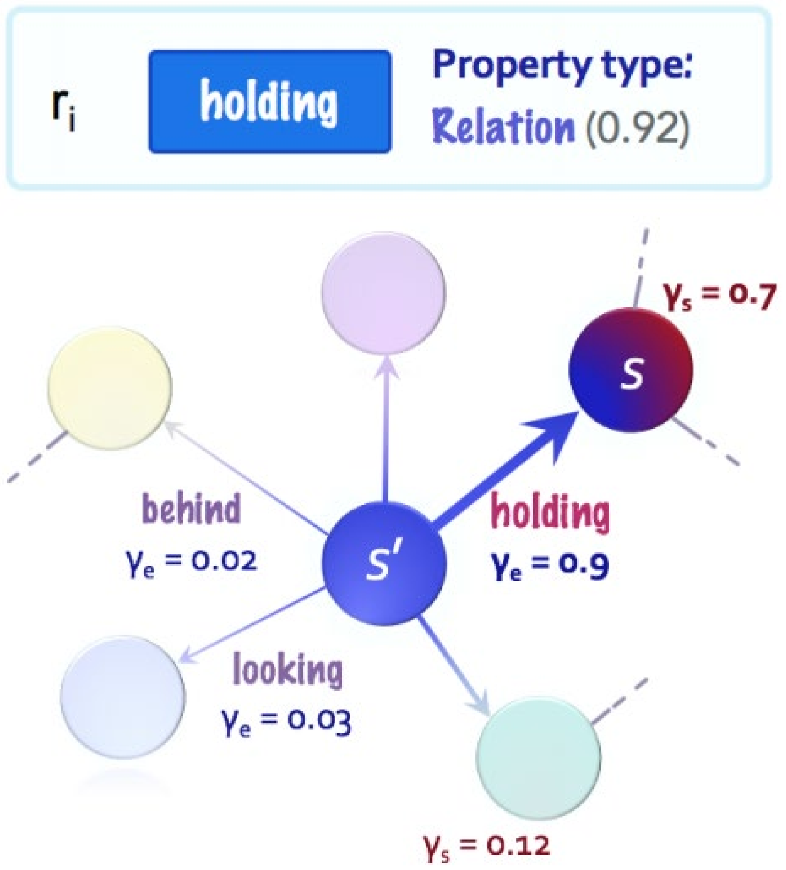
# Neural State Machine $(C, S, E, \{r_i\}_{i=0}^{N}, p_0, \delta)$

- $C$: Concepts: *obj identity, attributes, relation*
- $S$: States: *objs detected in image*
- $E$: Transition edges between the states: *relations of objs*
- $r_i$ a sequence of instructions: *encoded from the question*
- $p_0 : S \rightarrow [0, 1]$ distribution of the initial state.
- $\delta_{S,E}$ : $pi \times ri \rightarrow pi{+}1$ a state transition function
  - a neural module that at each step i
  - considers the distribution pi over the states as well as an input instruction ri
  - redistribute the probability along the edges, yielding an updated state distribution pi+1.

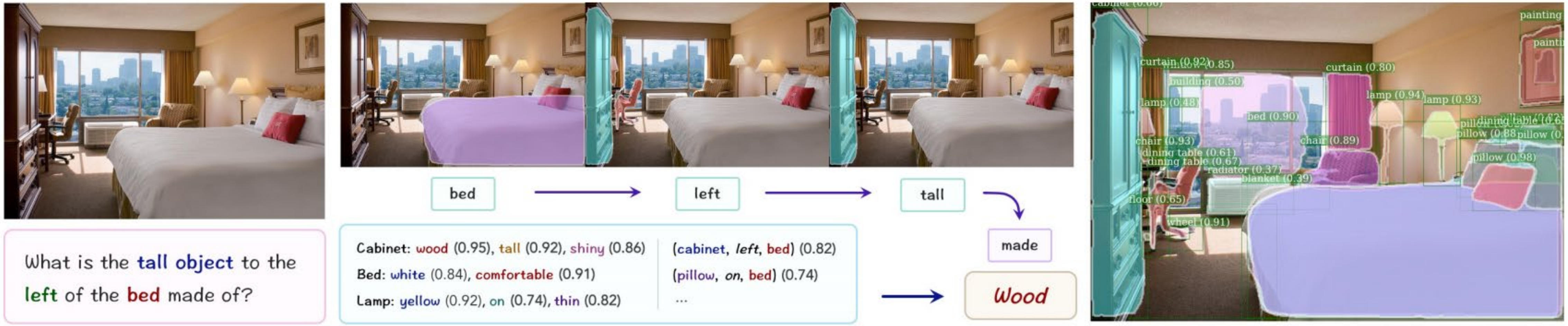
# State transition

Attention is being shifted from one node to its neighbor along the most relevant edge.

- Explicit reasoning ✓
- Multi-step information refinement ✓
- Dynamic structure reasoning ✗



# NSM in action




→Is the sequential order of reasoning necessarily the (inverse) order of the words in question?

→Is the reasoning state transitions only attention shifting?

→The gap between symbolic and compositional reasoning is still there