CMPT 983

Grounded Natural Language Understanding

February 25, 2021

Semantic Parsing

Today

- Semantic parsing for language grounding

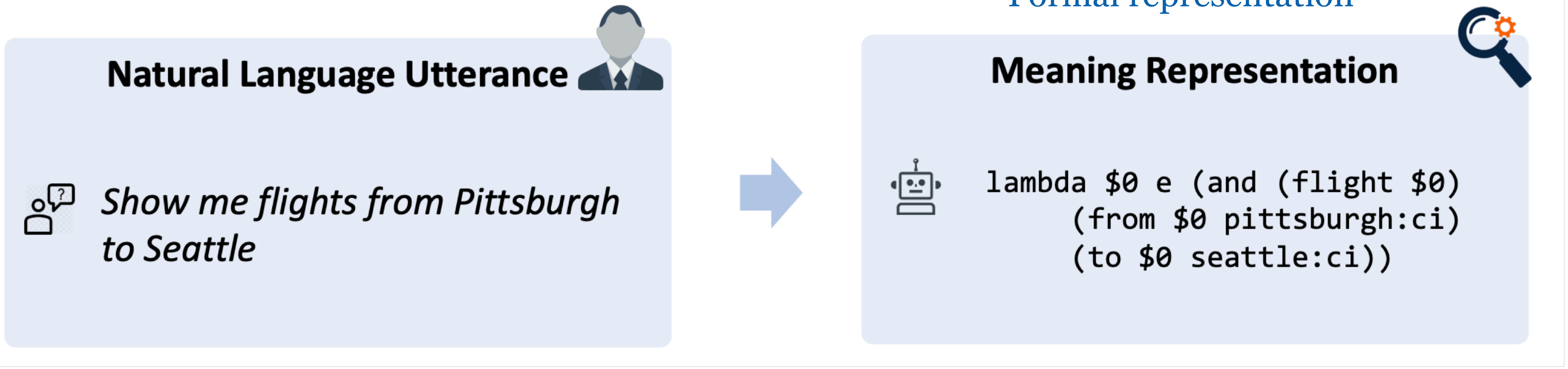
- What is semantic parsing?

- Semantic parsing for VQA

What is semantic parsing?

Semantic parsing

Logical form
Formal representation



Interpretable by a machine!

(figure credit: CMU CS 11-747, Pengcheng Yin)

Meaning representations

Machine-executable Meaning Representations

Show me flights from Pittsburgh to Seattle

```
lambda $0 e (and (flight $0)
     (from $0 pittsburgh:ci)
     (to $0 seattle:ci))
```

`Lambda Calculus Logical Form`

Lambda Calculus

Python, SQL, ...

Meaning Representations For Semantic Annotation

Arithmetic expressions

Lambda calculus

Computer Programs:

SQL / Python / DSLs

Abstract Meaning Representation (AMR),

Combinatory Categorical Grammar (CCG)

(figure credit: CMU CS 11-747, Pengcheng Yin)

Semantic parsing components and terminology

Utterance

User's Natural Language Query

Show me flights from Pittsburgh to Seattle

Parser

Parsing to Meaning Representation

```
lambda $0 e (and (flight $0)
     (from $0 pittsburgh:ci)
     (to $0 seattle:ci))
```

Program

Executor

Execute Programs against KBs

Freebase

WIKIDATA

Denotation

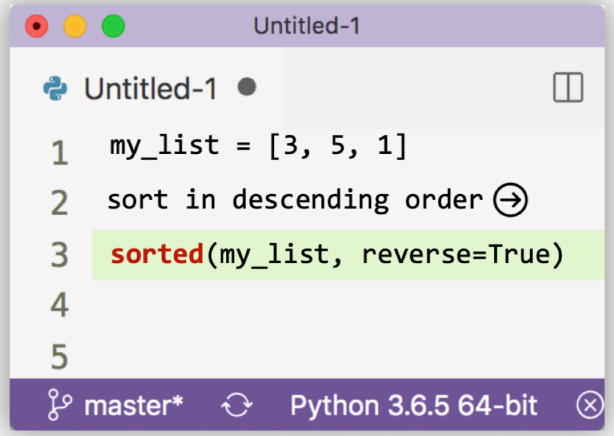
Execution Results (Answer)

```
1. Alaska Air 119
2. American 3544 -> Alaska 1101
3. …
```

(figure credit: CMU CS 11-747, Pengcheng Yin)

Applications

NLP Tasks
Question Answering
Applications
Natural language interfaces
Dialogue agents
Robots



Virtual Assistants

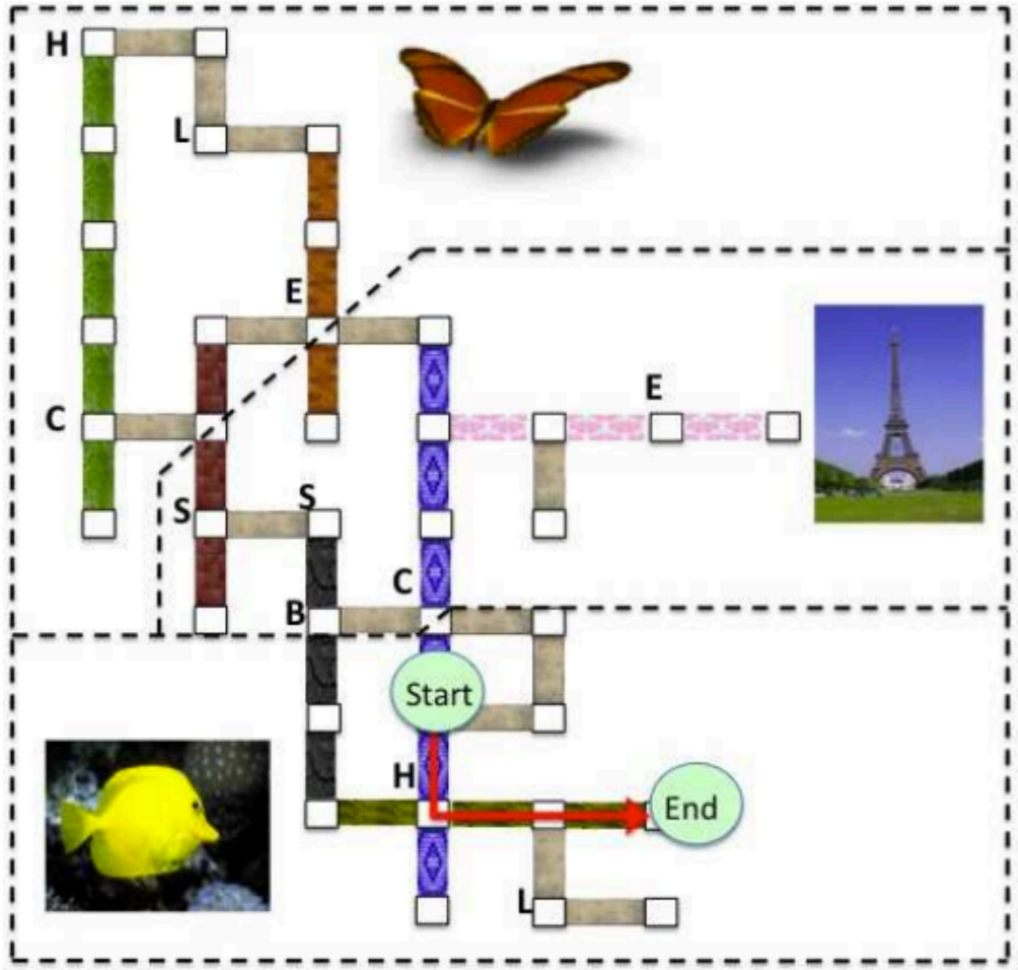
- *Set an alarm at 7 AM*
- *Remind me for the meeting at 5pm*
- *Play Jay Chou's latest album*

Natural Language Programming

- *Sort my_list in descending order*
- *Copy my_file to home folder*
- *Dump my_dict as a csv file output.csv*

(figure credit: CMU CS 11-747, Pengcheng Yin)

Semantic parsing for instruction following



Instruction: "Place your back against the wall of the 'T' intersection. Turn left. Go forward along the pink-flowered carpet hall two segments to the intersection with the brick hall. This intersection contains a hatrack. Turn left. Go forward three segments to an intersection with a bare concrete hall, passing a lamp. This is Position 5."

Parse:
Turn ( ),
Verify (back: WALL),
Turn (LEFT),
Travel ( ),
Verify (side: BRICK HALLWAY),
Turn (LEFT),
Travel (steps: 3),
Verify (side: CONCRETE HALLWAY)

Tiny amount of data, pipelined system

	Original	Single-sentence
# instructions	706	3236
Vocabulary size	660	629
Avg. # sentences	5.0 (2.8)	1.0 (0)
Avg. # words	37.6 (21.1)	7.8 (5.1)
Avg. # actions	10.4 (5.7)	2.1 (2.4)

Learning to Interpret Natural Language Navigation Instructions from Observations, Chen and Mooney, AAAI 2011

Training semantic parsers

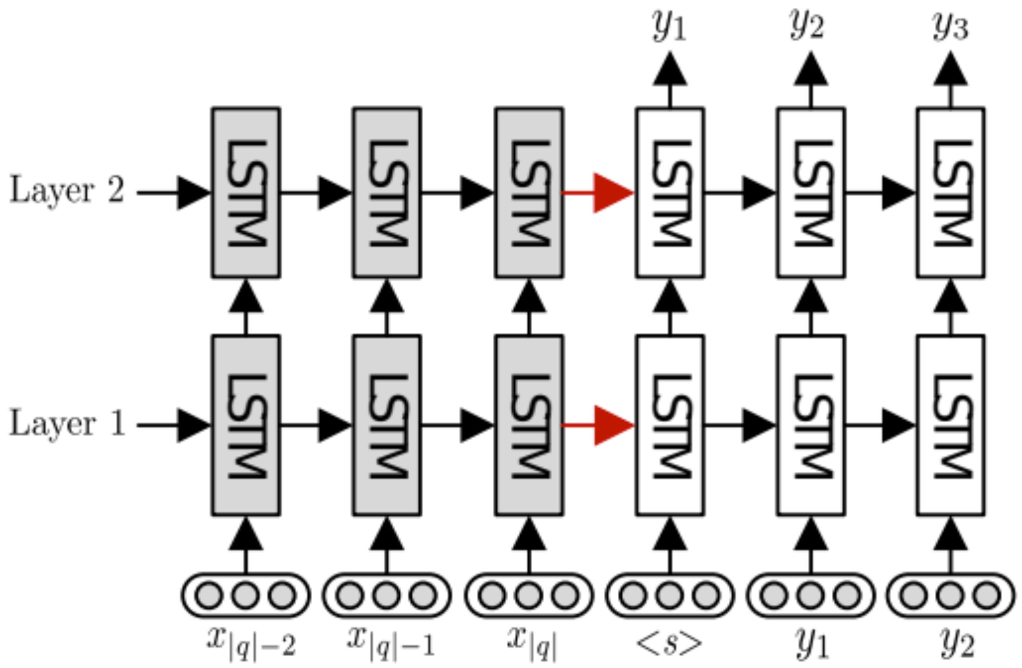
- Supervised learning
  - Training data of (utterance, program) pairs
  - Use general supervised structured prediction methods
    - similar methods as for constituency parsing and dependency parsing

- Weakly supervised learning
  - Training data of (utterance, denotation) pairs
  - Hypothesize programs, execute them and check if the denotation matches

Semantic parsing as seq2seq

- Treat the target meaning representation as a sequence of surface tokens
- Reduce the (structured prediction) task as another sequence-to-sequence learning problem

Usually with attention and copy mechanism

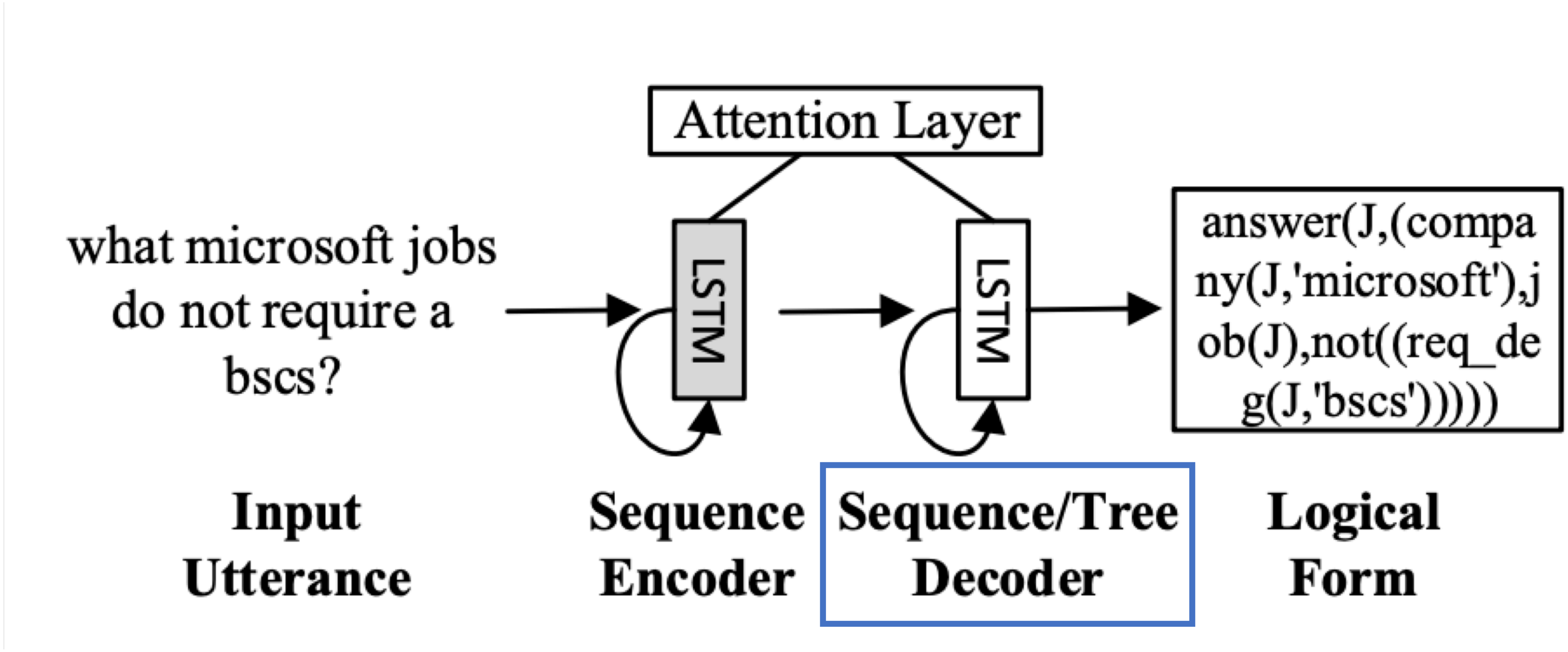


Warning: Output may not be valid!

Also used for structured parsing in general (Vinyals et al. 2014, Vaswani et al. 2017)

Language to Logical Form with Neural Attention, Dong and Lapata, ACL 2016

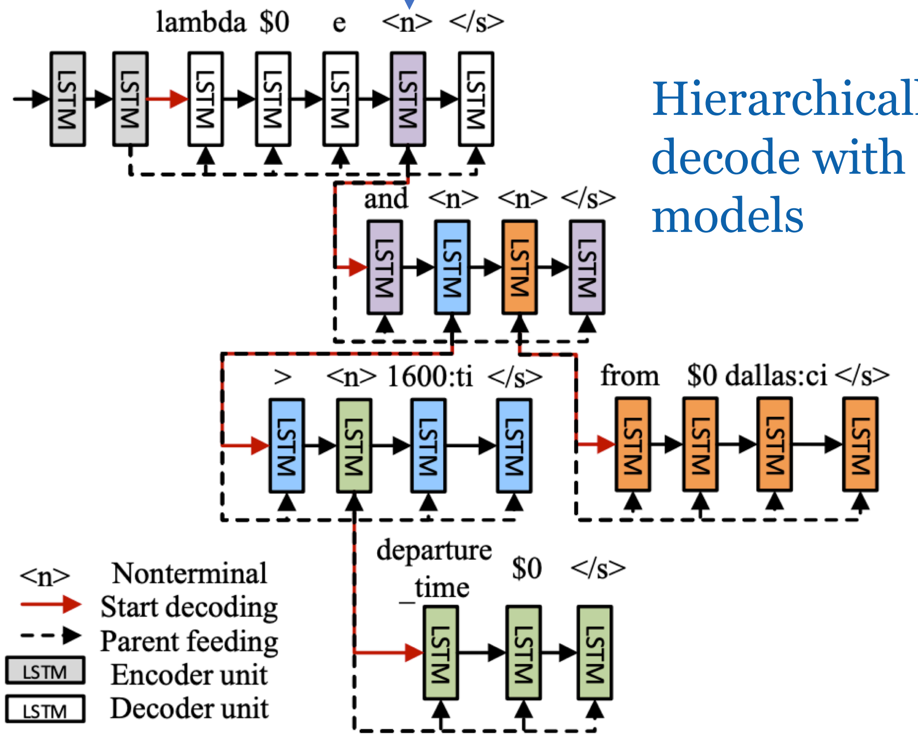
Structured decoding



Language to Logical Form with Neural Attention, Dong and Lapata, ACL 2016

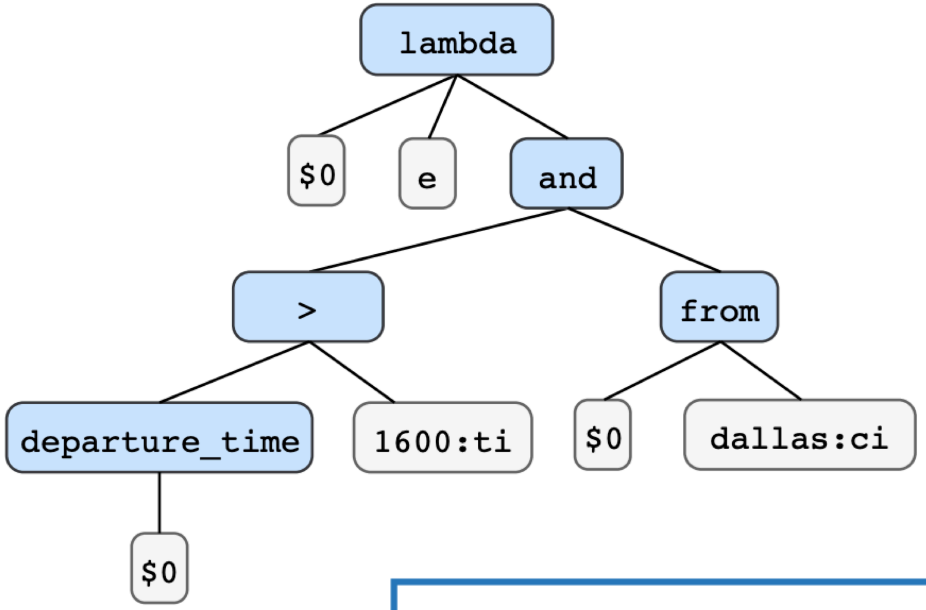
Structured decoding

Special nonterminal symbol



Hierarchically decode with seq2seq models

Show me flight from Dallas departing after 16:00



	GEO	ATIS
seq2seq	84.6	84.2
seq2tree	87.1	84.6

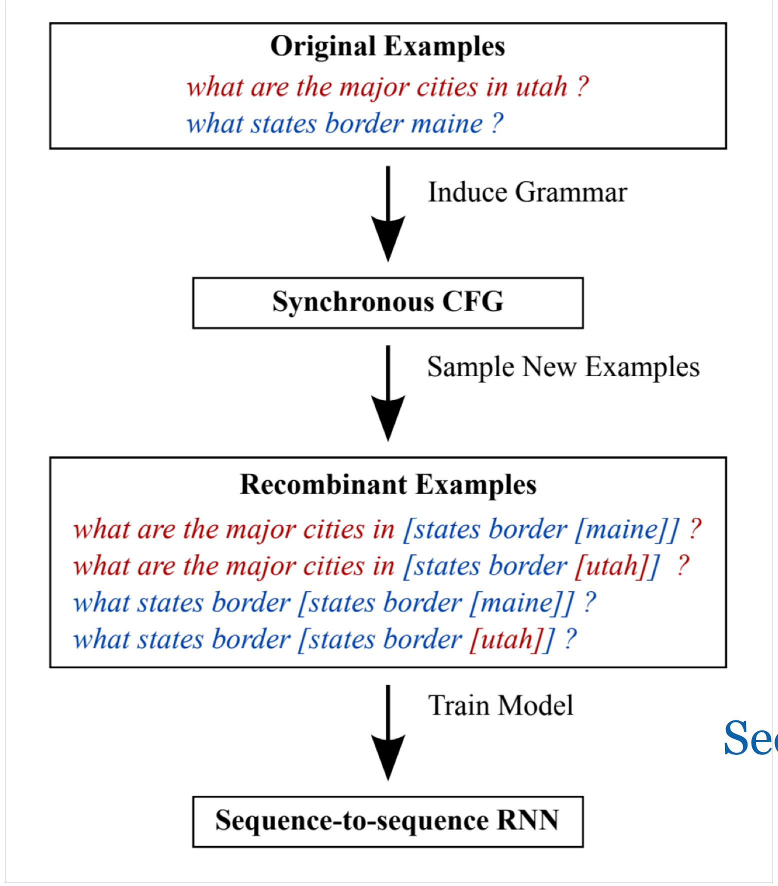
Training semantic parsers

These kind of training data is expensive and hard to obtain

- Supervised learning
  - Training data of (utterance, program) pairs
  - Use general supervised structured prediction methods
    - similar methods as for constituency parsing and dependency parsing
  - Data augmentation: try to generate more training data
- Weakly supervised learning
  - Training data of (utterance, denotation) pairs
  - Hypothesize programs, execute them and check if the denotation matches

Data augmentation

- Generate training data using a grammar



GEO: 880 examples (600 train, 280 test)
JOBS: 610 examples (500 train, 140 test)
ATIS: 5410 examples (4480 train, 480 dev, 450 test)

	GEO	ATIS
no copy	74.6	69.9
with copy	85.0	76.3
with data recomb	89.3	83.3

Seq2seq model with attention + copy mechanism

Data Recombination for Neural Semantic Parsing, Jia and Liang, ACL 2016

Weakly supervised semantic parsing

Weakly Supervised Semantic Parsing

What is the most populous city in United States?

City	Country	Population	GDP
New York	USA	8.62M	1275B
Hong Kong	China	7.39M	341.4B
Tokyo	Japan	9.27M	1800B
London	UK	8.78M	650B
Los Angeles	USA	4.00M	941B

```
Answer: New York
```

Hypothesized Programs

```
City.OrderBy(Population)
     .First() => Result: Tokyo
```

```
City.Filter(Country=='USA')
     .OrderBy(Population)
     .First() => Result: New York
```

```
City.Filter(Country=='USA')
     .OrderBy(GDP)
     .First() => Result: New York
```

(figure credit: CMU CS 11-747, Pengcheng Yin)

Weakly supervised semantic parsing

Hypothesized Programs

```
City.OrderBy(Population)
    .First() => Result: Tokyo
```

```
City.Filter(Country=='USA')
    .OrderBy(Population)
    .First() => Result: New York
```

```
City.Filter(Country=='USA')
    .OrderBy(GDP)
    .First() => Result: New York
```

Large Search Space

Exponentially large search space w.r.t. the size of programs

Very Sparse Rewards

Only very few programs are actually correct

Spurious Programs

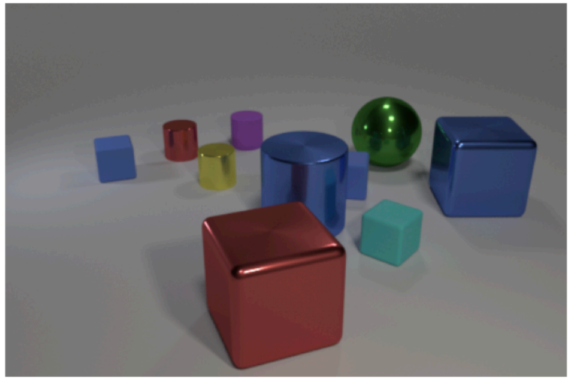
Spurious programs could also hit the correct answer, leading to noisy reward signals.

(figure credit: CMU CS 11-747, Pengcheng Yin)

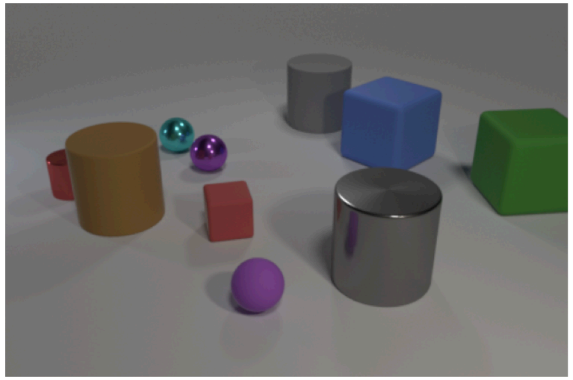
Semantic parsing for VQA

Last time: MAC/NSM on CLEVR/GQA

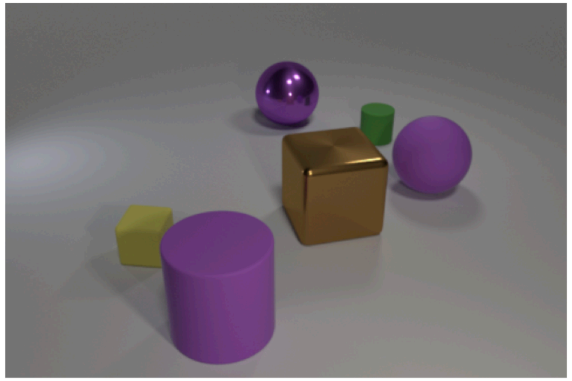
- Constructed by building functional programs converted to natural language
- Small space of shapes, attributes, and relations



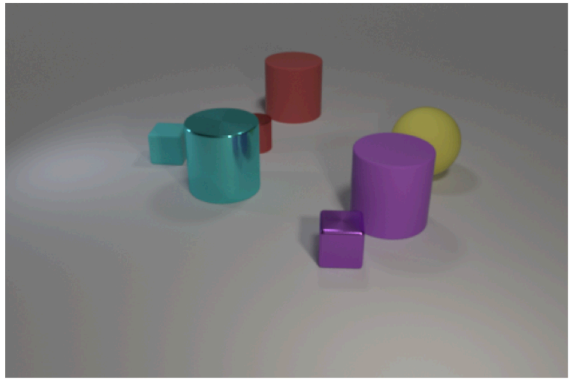
Q: What shape is the object reflected in the blue cylinder?
A: cube



Q: What number of cylinders share the same color?
A: 2



Q: How many objects are not purple and not metallic?
A: 2

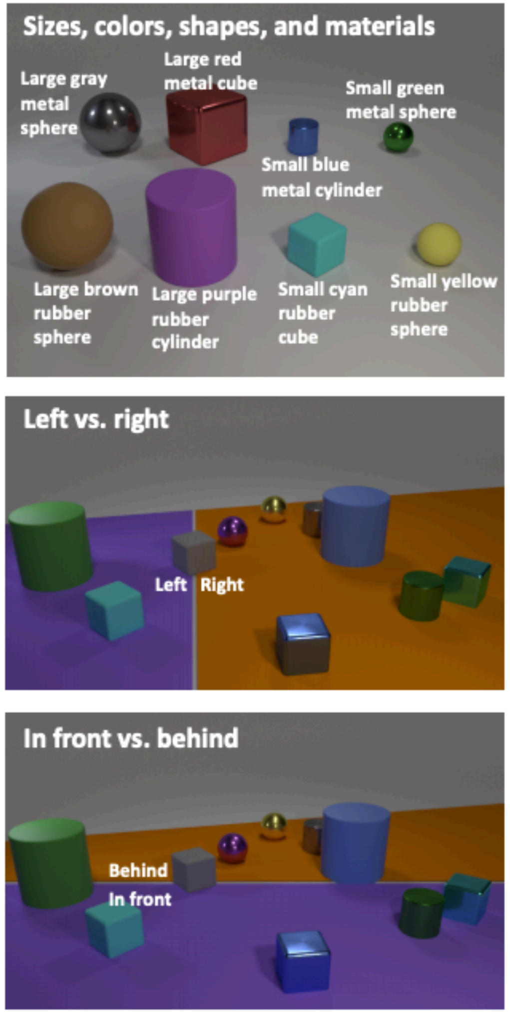


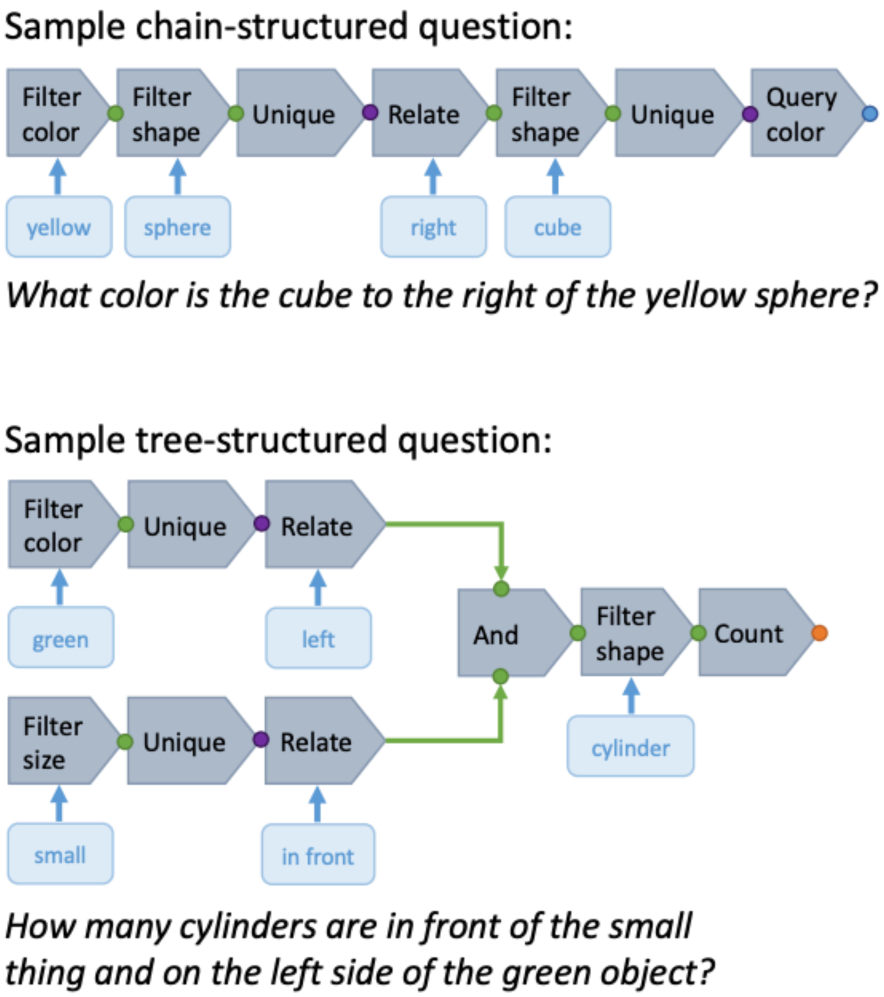
Q: What color is the object partially blocked by the purple cylinder?
A: yellow

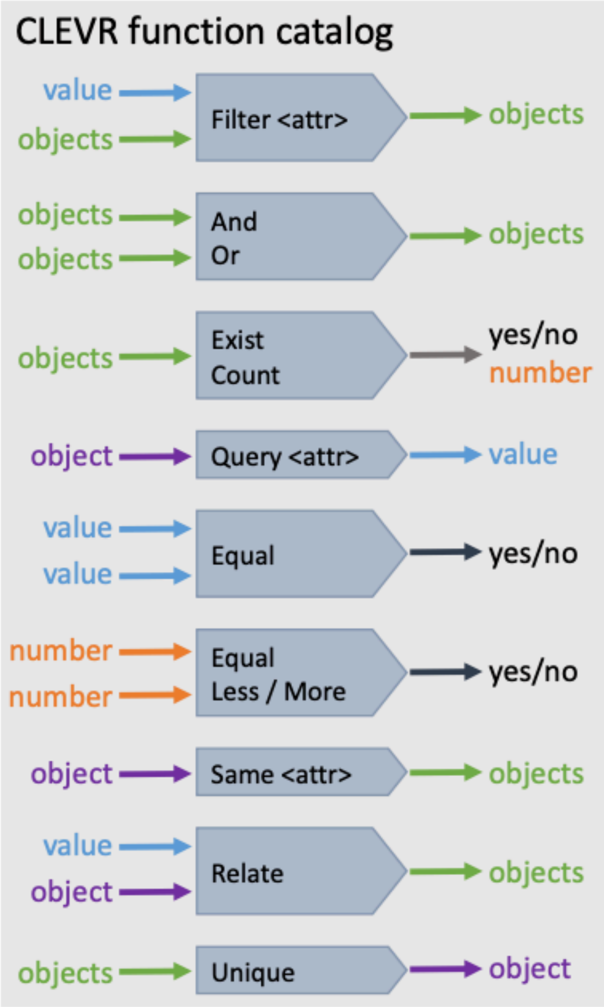
A closer look at CLEVR

Shape and attributes

Programs: formed from composable modules



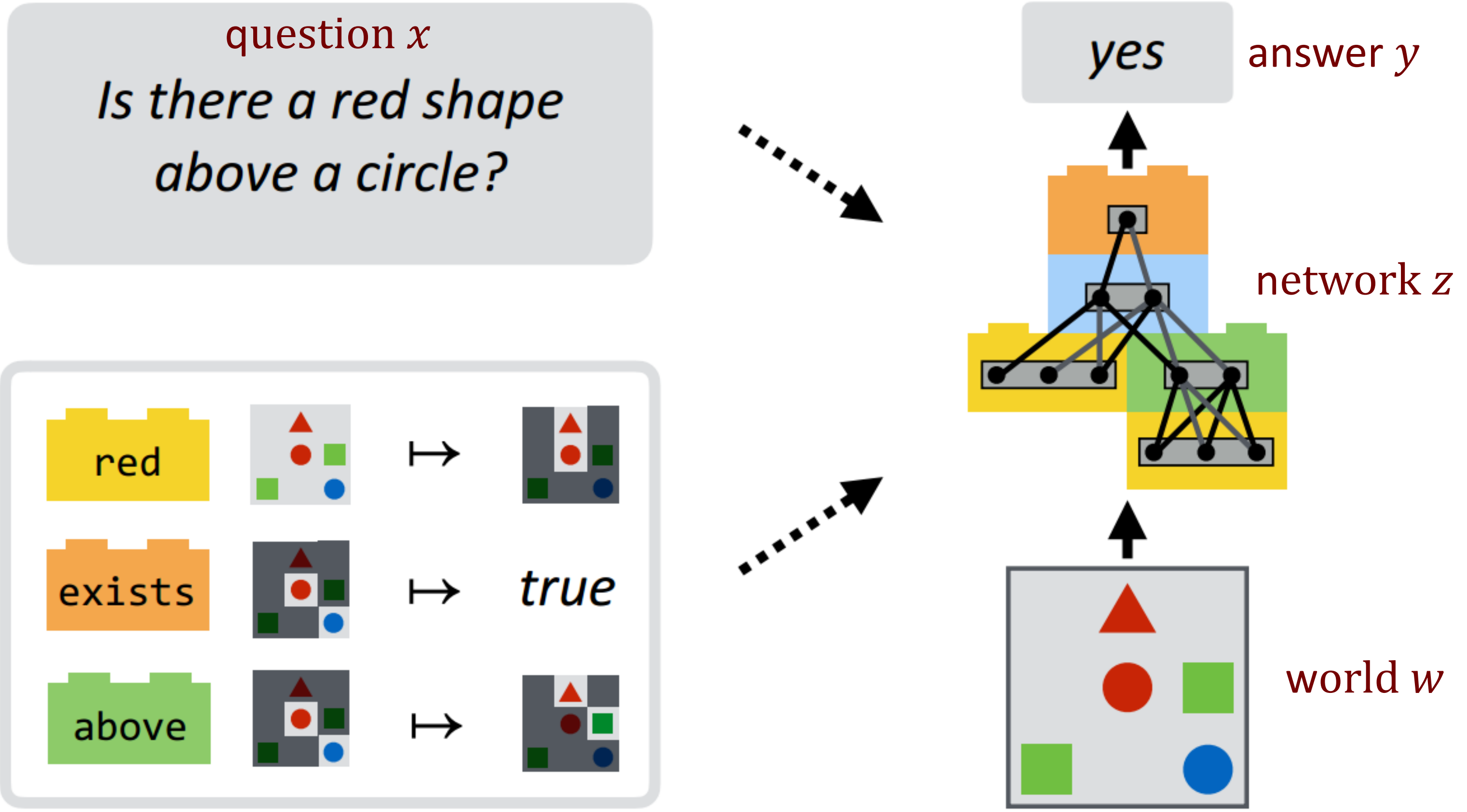




Relations

Generated language

Neural module networks



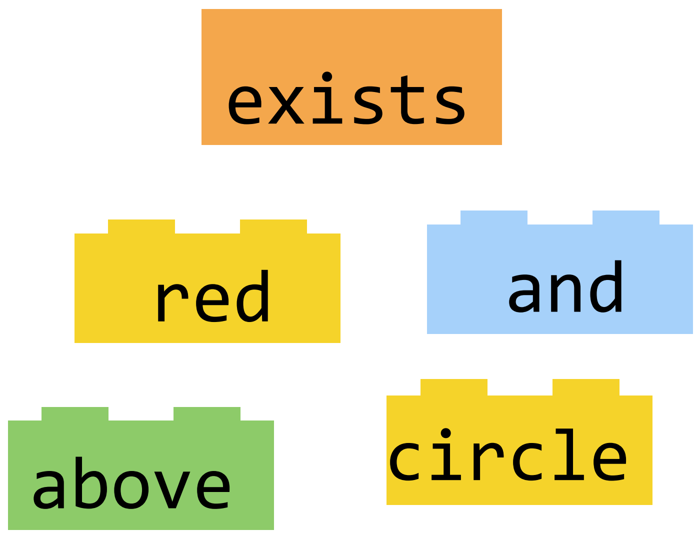
Neural module networks, Andreas et al, CVPR 2016

Learning to compose neural networks for question answering, Andreas et al, NAACL 2016

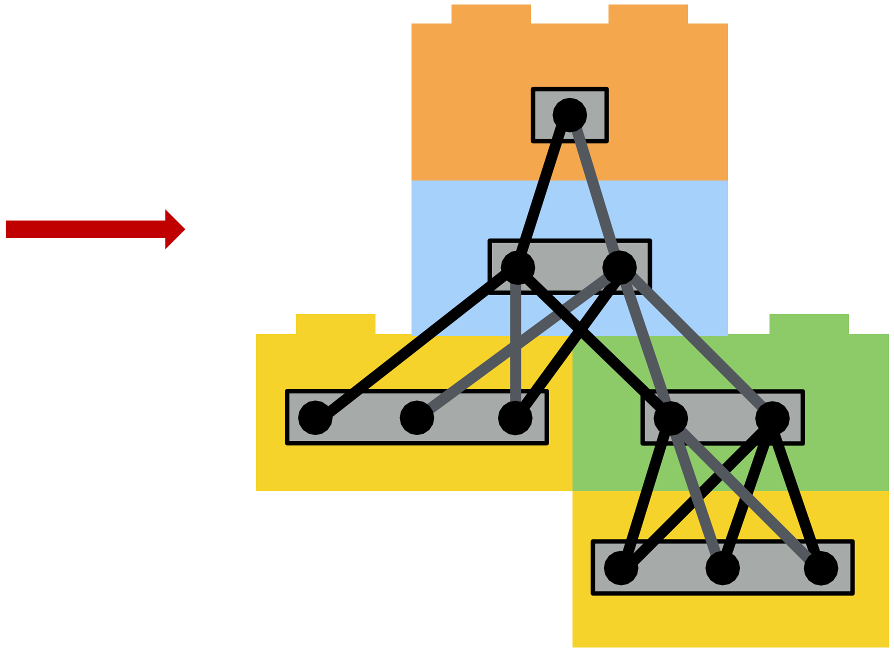
Neural module networks

- Neural networks as little lego blocks (modules) that can be composed together to form a program to execute

Types of modules are prespecified



lego brick = function



Neural module networks, Andreas et al, CVPR 2016

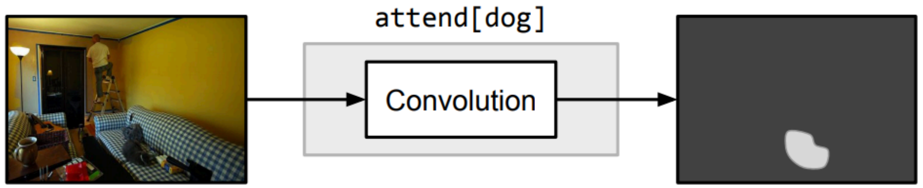
Learning to compose neural networks for question answering, Andreas et al, NAACL 2016

Types of neural modules

Modules are instantiated with different weights

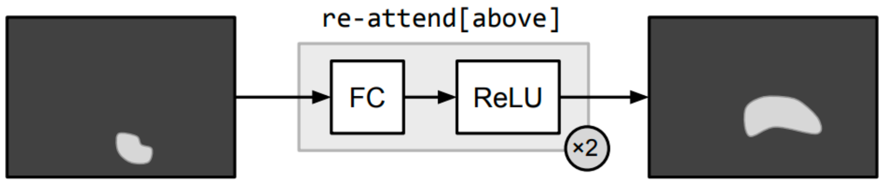
Find

$\texttt{attend} : \mathit{Image} \to \mathit{Attention}$



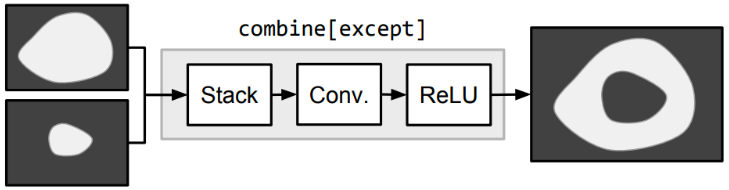
Relate / Transform

$\texttt{re-attend} : \mathit{Attention} \to \mathit{Attention}$



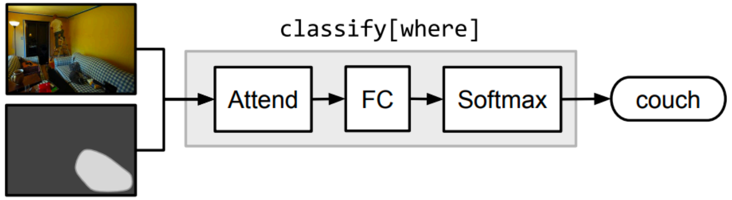
And

$\texttt{combine} : \mathit{Attention} \times \mathit{Attention} \to \mathit{Attention}$



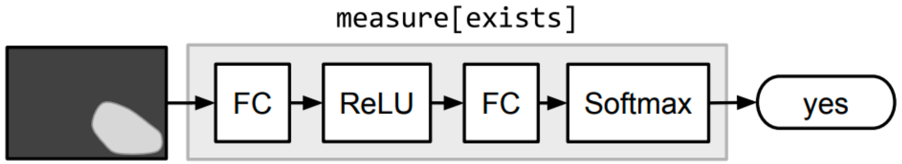
Describe / Classify

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

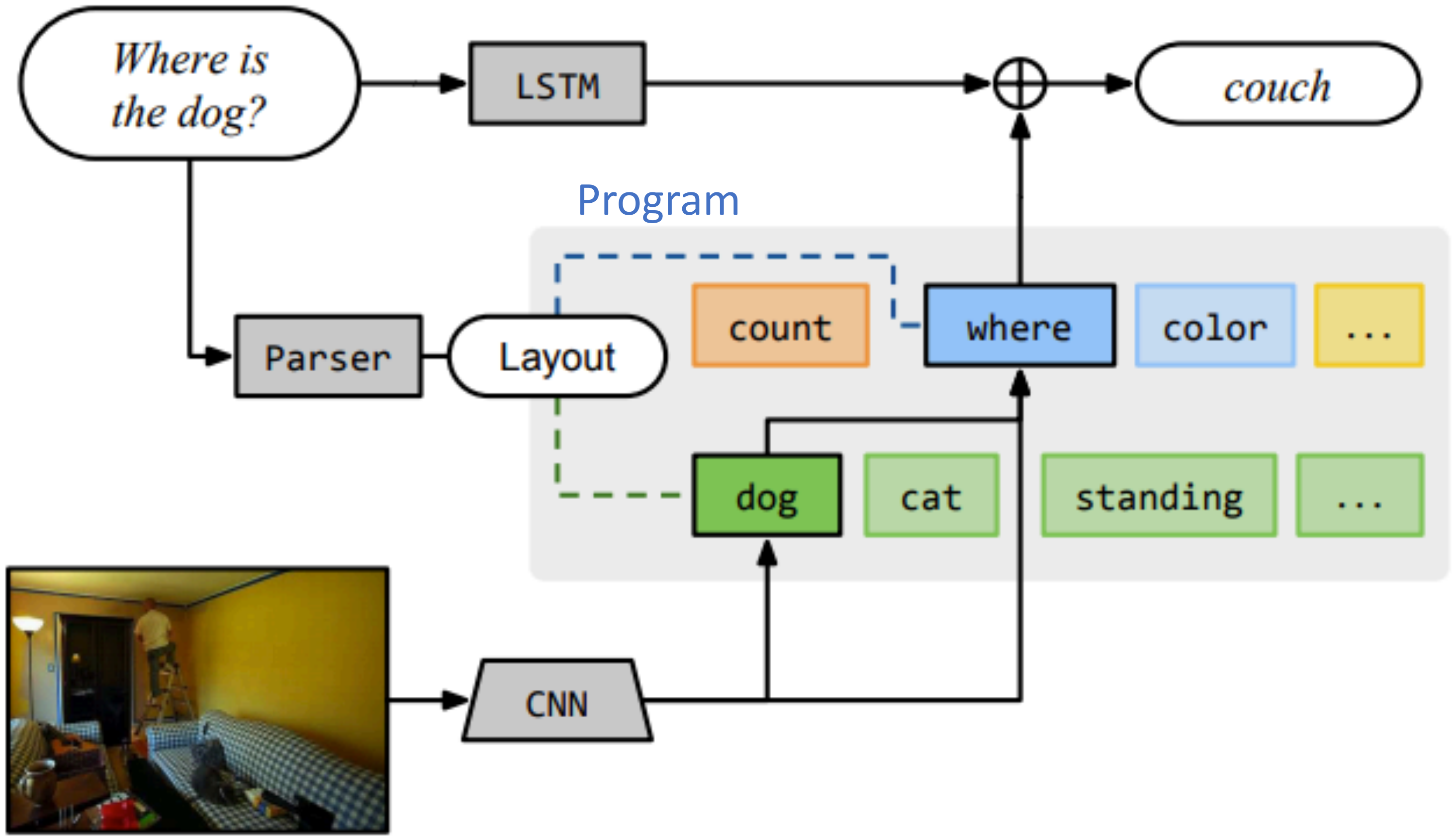


Exists / Count

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



Neural module networks



Neural module networks, Andreas et al, CVPR 2016

Neural module networks

Parameters θ

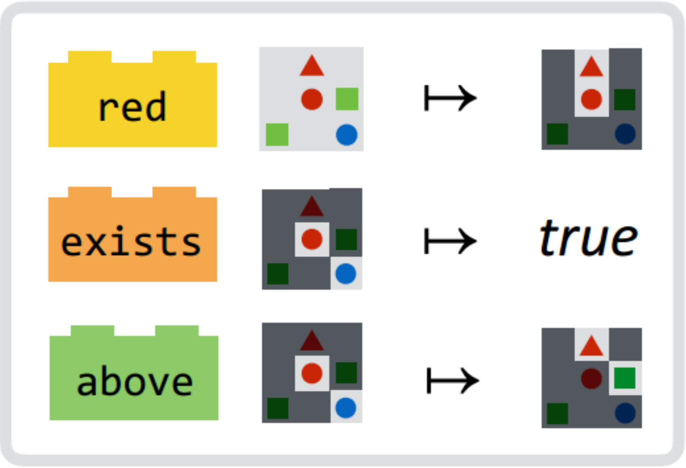
Uses a separate dependency parser to extract relations between words

Layout is heuristically generated from parse

Modules are trained

question x

Is there a red shape above a circle?



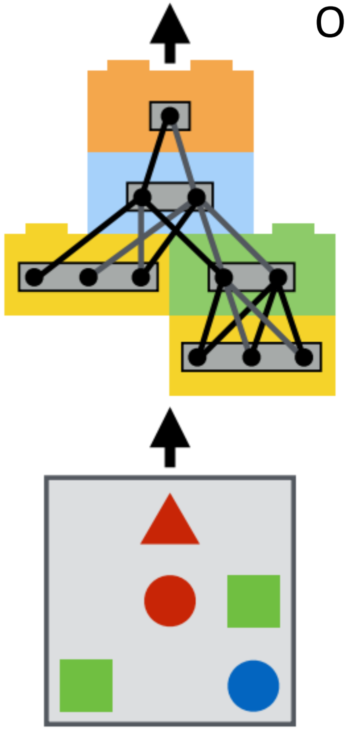
Execution model: $p_z(y|w;\theta_e)$

answer y

yes

Given a network layout z, input world w, what is the answer y?

Operate on continuous values



network z

Layout model:
Given a question x, what network layout z to use?

world w

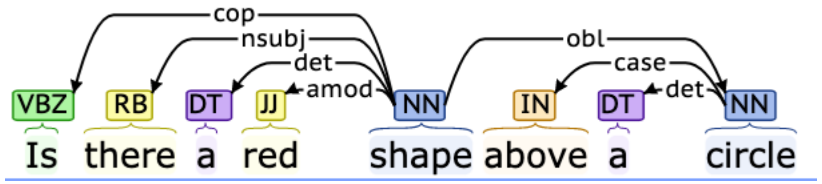
Neural module networks, Andreas et al, CVPR 2016

Learning to compose neural networks for question answering, Andreas et al, NAACL 2016

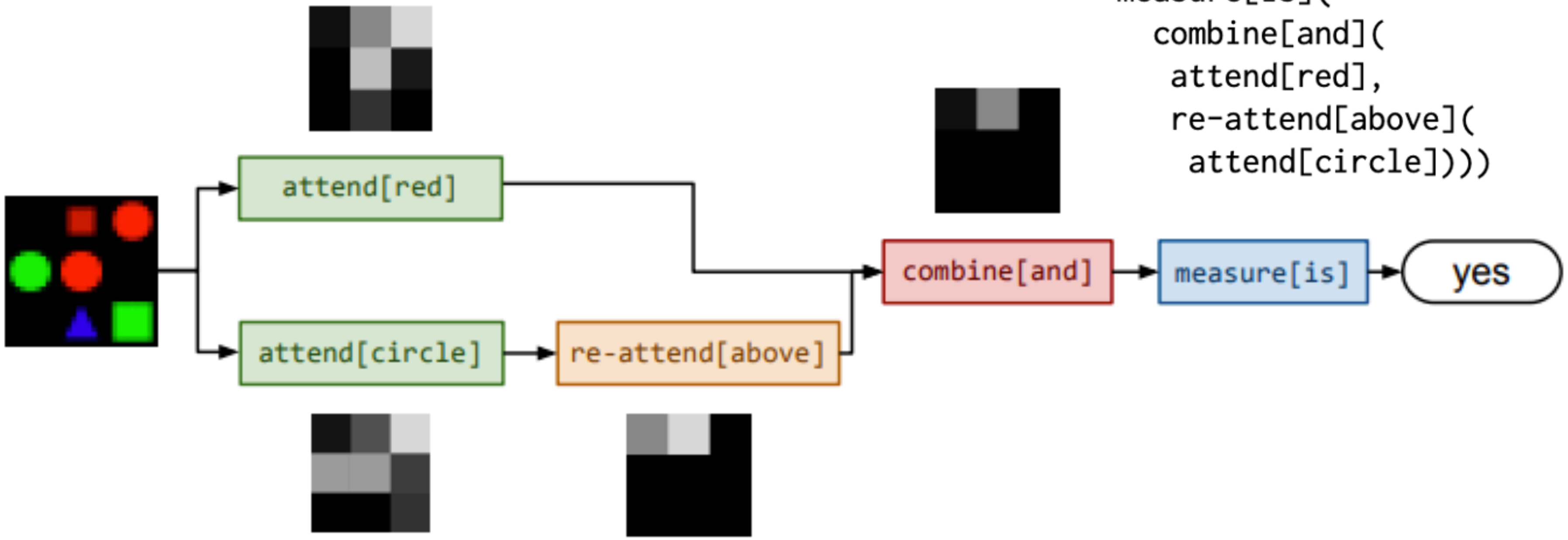
Example

- *Is there a red shape above a circle?*

Dependency Parse



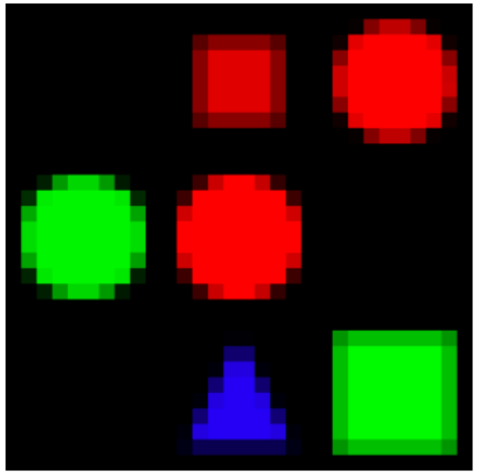

```
measure[is](
  combine[and](
   attend[red],
   re-attend[above](
    attend[circle])))
```



Leaves are attend modules

Internal nodes are re-attend or combine modules

Root is measure or classify modules

 *what is the color of the horse?*	 *what color is the vase?*	 *is the bus full of passengers?*	 *is there a red shape above a circle?*

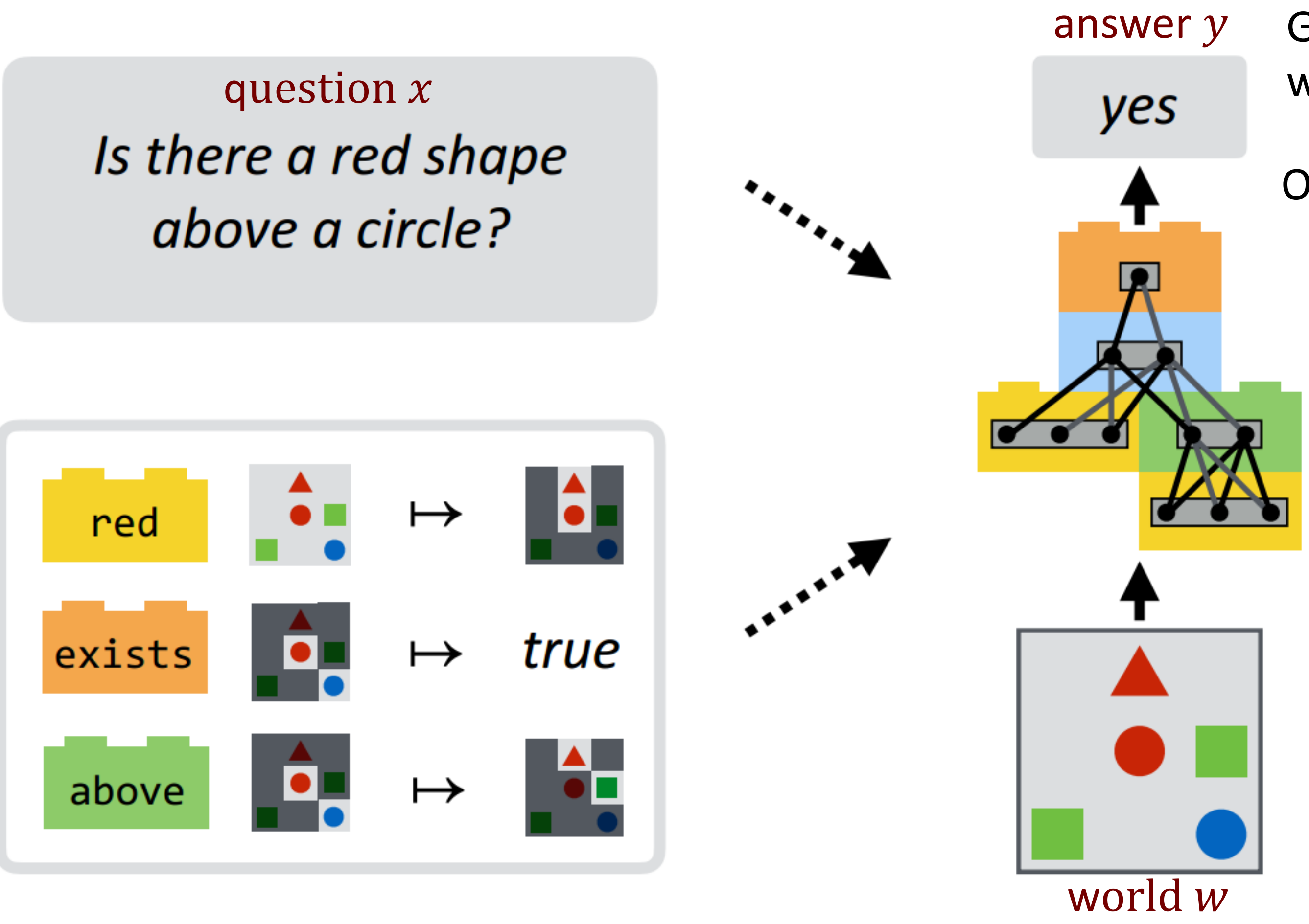
`classify[color](`
`  attend[horse])`
	`classify[color](`
	`  attend[vase])`
		`measure[is](`
		`  combine[and](`
		`    attend[bus],`
		`    attend[full])`
			`measure[is](`
			`  combine[and](`
			`   attend[red],`
			`   re-attend[above](`
			`    attend[circle])))`

brown (brown)	green (green)	yes (yes)	no (no)

Neural module networks

Parameters θ

Separate dependency parser is used to generate candidate layouts



Execution model: $p_z(y|w;\theta_e)$

Given a network layout z, input world w, what is the answer y?

Operate on continuous values

Layout model: $p(z|x;\theta_\ell)$

Given a question x, what network layout z to use?

Learn to score layouts

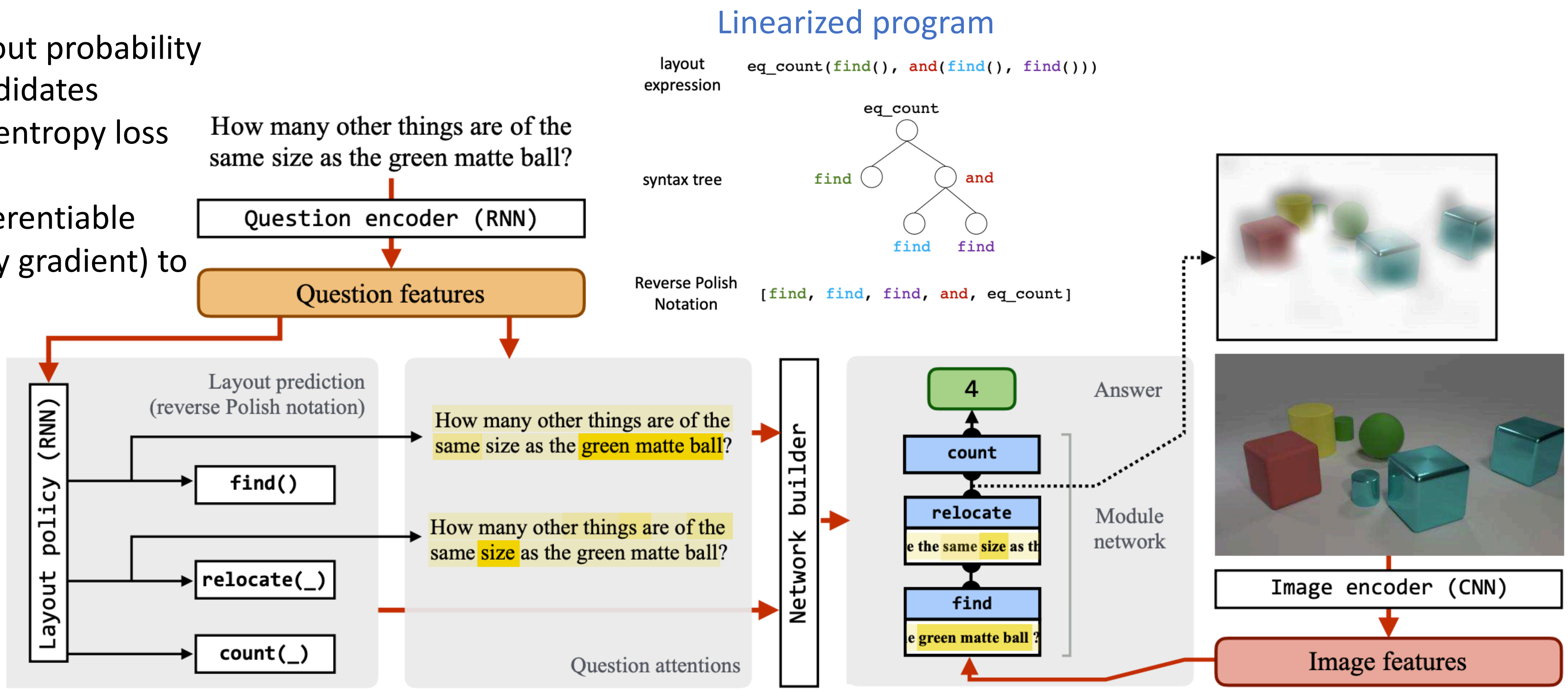
Modules are trained

Neural module networks, Andreas et al, CVPR 2016

Learning to compose neural networks for question answering, Andreas et al, NAACL 2016

End-to-End Module Networks (N2NMN)

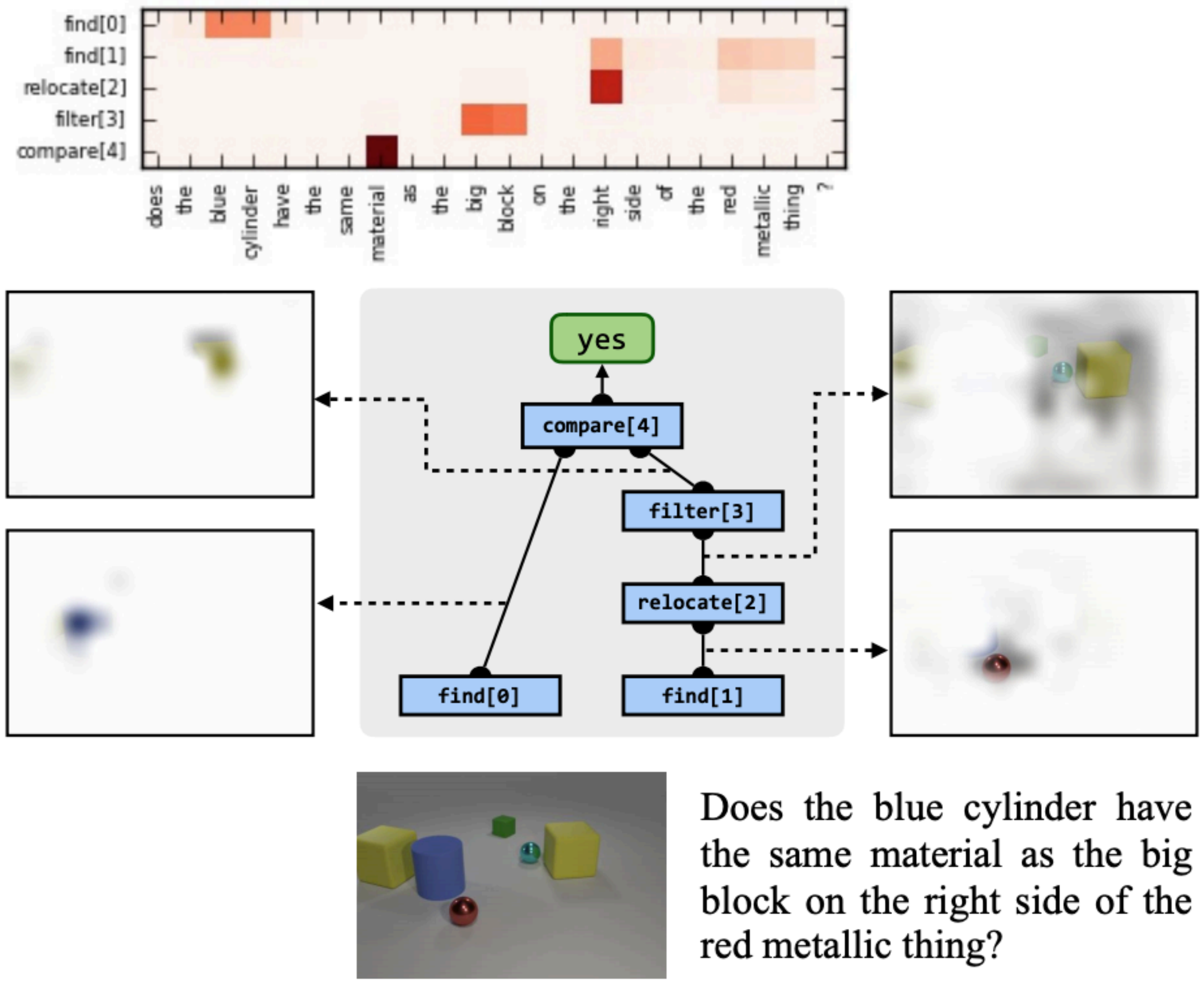
- Modeled layout probability
- Sampled candidates
- Loss is cross-entropy loss over answers
- Not fully differentiable
- Use RL (policy gradient) to train



Learn to generate program directly!

Learning to Reason: End-to-End Module Networks for Visual Question Answering, Hu et al, ICCV 2017

End-to-End Module Networks (N2NMN



Learning to Reason: End-to-End Module Networks for Visual Question Answering, Hu et al, ICCV 2017

Inferring and Executing Programs for Visual Reasoning

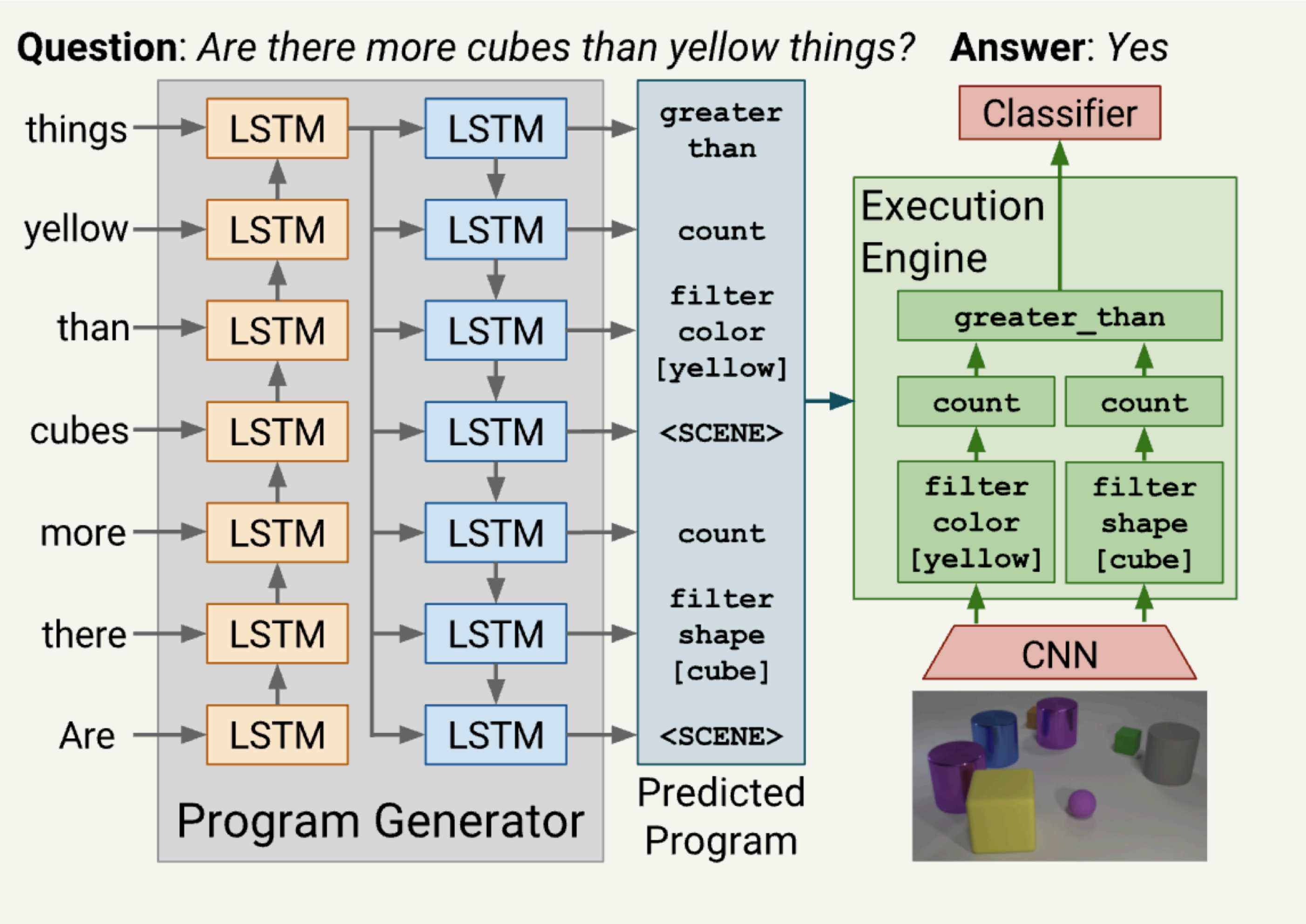
- Program generator

text → program

- Execution engine

program + image → answer

- Both neural networks
- Can be trained end-to-end in a supervised manner



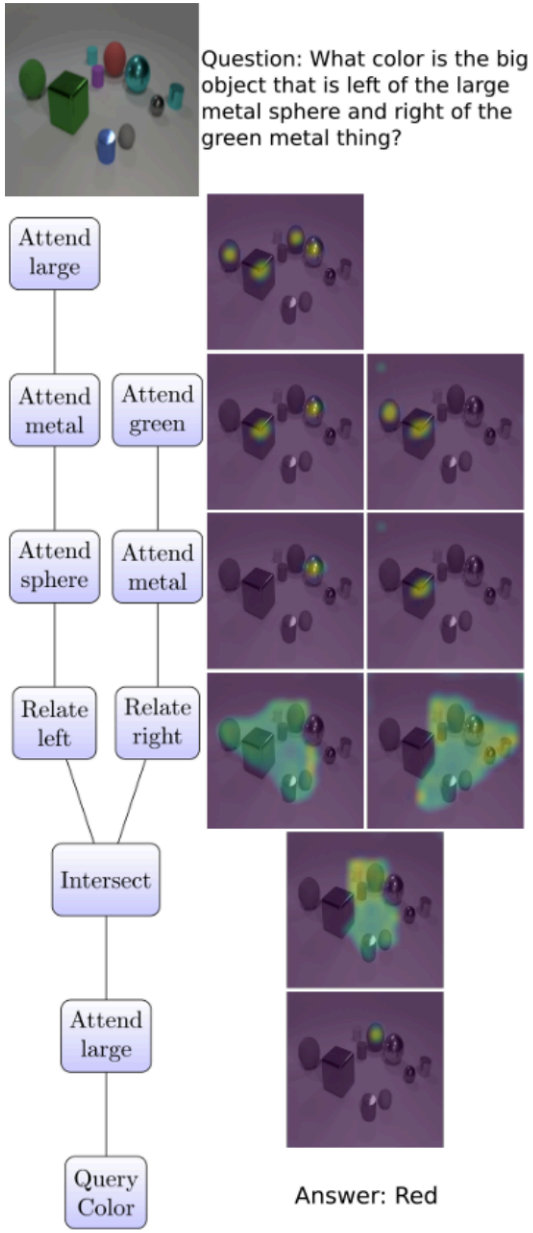
(Referred to by other work as IEP or PG+EE)

Inferring and Executing Programs for Visual Reasoning, Johnson et al, ICCV 2017

Combining NMN + IEP

- Main idea: NMN (attention) + PG (supervised training)
- Some additional improvements
  - Original Image features (stem) is retained
  - Increased spatial resolution

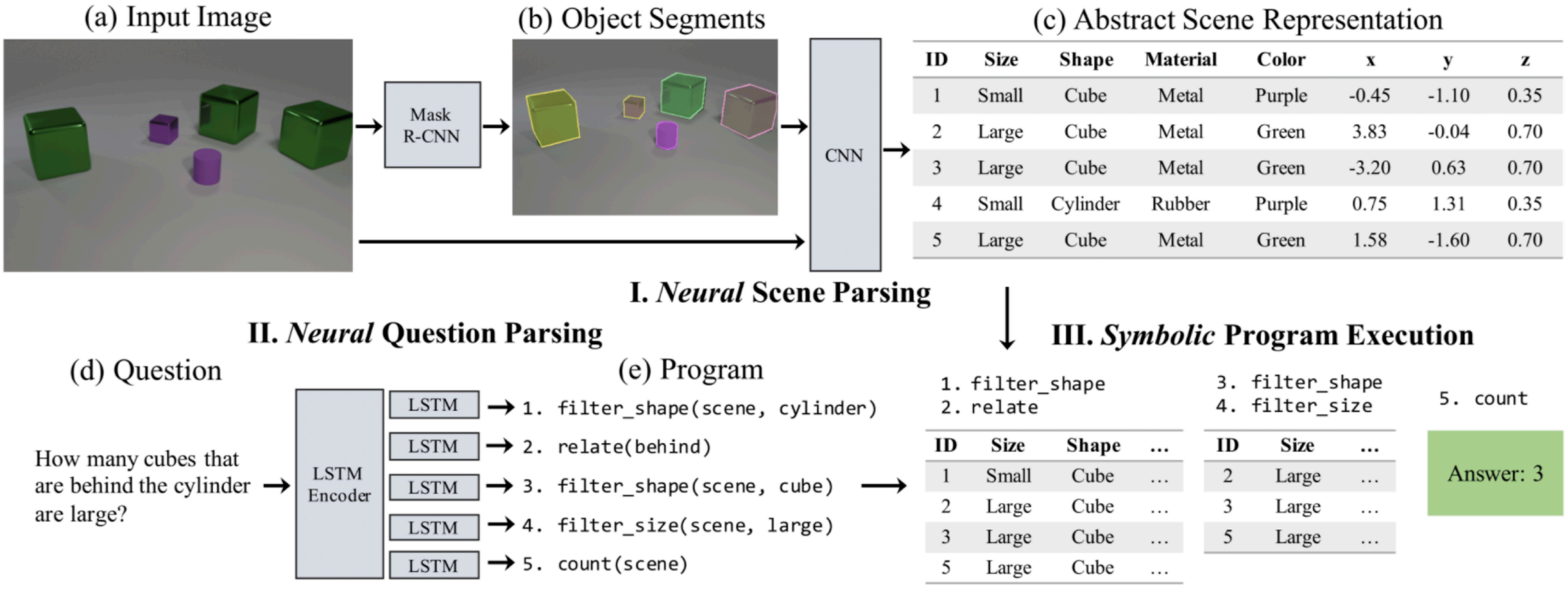
Module Type	Operation	Language Analogue
`Attention`	Attention × Stem → Attention	Which things are [property]?
`Query`	Attention × Stem → Encoding	What [property] is x?
`Relate`	Attention × Stem → Attention	Left of, right of, in front, behind
`Same`	Attention × Stem → Attention	Which things are the same [property] as x?
`Comparison`	Encoding × Encoding → Encoding	Are x and y the same [property]?
`And`	Attention × Attention → Attention	Left of x *and* right of y
`Or`	Attention × Attention → Attention	Left of x *or* right of y



Transparency by Design: Closing the Gap Between Performance and Interpretability in Visual Reasoning, Mascharka et al, CVPR 2018

Neural Symbolic VQA

Scene parser



Trained using REINFORCE

Collection of
Python functions

Neural-Symbolic VQA: Disentangling Reasoning from Vision and Language Understanding, Yi et al, NeurIPS 2018

Comparison of models (CLEVR, synthetic)

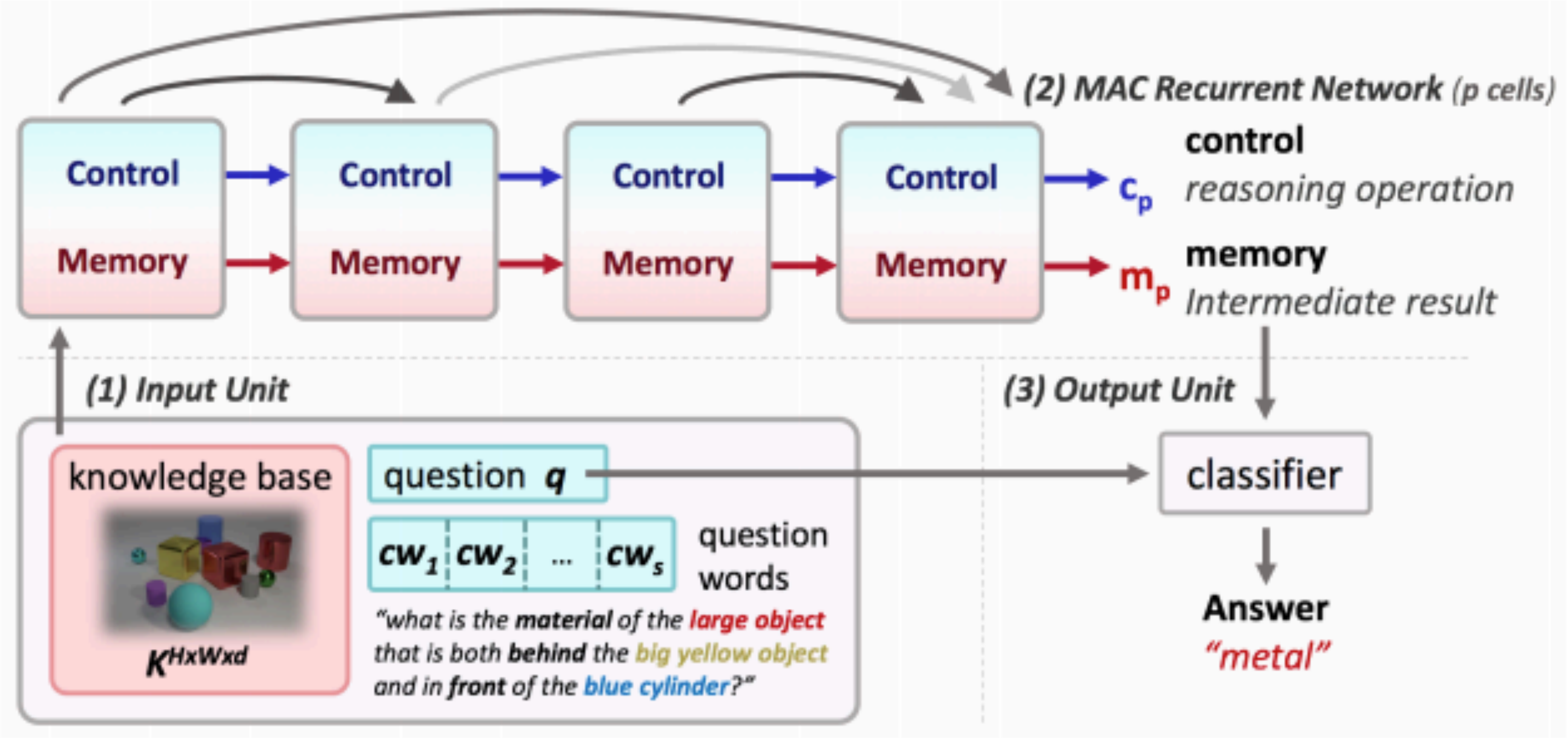
Methods	Count	Exist	Compare Number	Compare Attribute	Query Attribute	Overall
Humans [Johnson et al., 2017b]	86.7	96.6	86.4	96.0	95.0	92.6
CNN+LSTM+SAN [Johnson et al., 2017b]	59.7	77.9	75.1	70.8	80.9	73.2
N2NMN* [Hu et al., 2017]	68.5	85.7	84.9	88.7	90.0	83.7
Dependency Tree [Cao et al., 2018]	81.4	94.2	81.6	97.1	90.5	89.3
CNN+LSTM+RN [Santoro et al., 2017]	90.1	97.8	93.6	97.1	97.9	95.5
IEP* [Johnson et al., 2017b]	92.7	97.1	98.7	98.9	98.1	96.9
CNN+GRU+FiLM [Perez et al., 2018]	94.5	99.2	93.8	99.0	99.2	97.6
DDRprog* [Suarez et al., 2018]	96.5	98.8	98.4	99.0	99.1	98.3
MAC [Hudson and Manning, 2018]	97.1	99.5	99.1	99.5	99.5	98.9
TbD+reg+hres* [Mascharka et al., 2018]	97.6	99.2	99.4	99.6	99.5	99.1
NS-VQA (ours, 90 programs)	64.5	87.4	53.7	77.4	79.7	74.4
NS-VQA (ours, 180 programs)	85.0	92.9	83.4	90.6	92.2	89.5
NS-VQA (ours, 270 programs)	**99.7**	**99.9**	**99.9**	**99.8**	**99.8**	**99.8**

*trained with all program annotations (700K)

Neural-Symbolic VQA: Disentangling Reasoning from Vision and Language Understanding, Yi et al, NeurIPS 2018

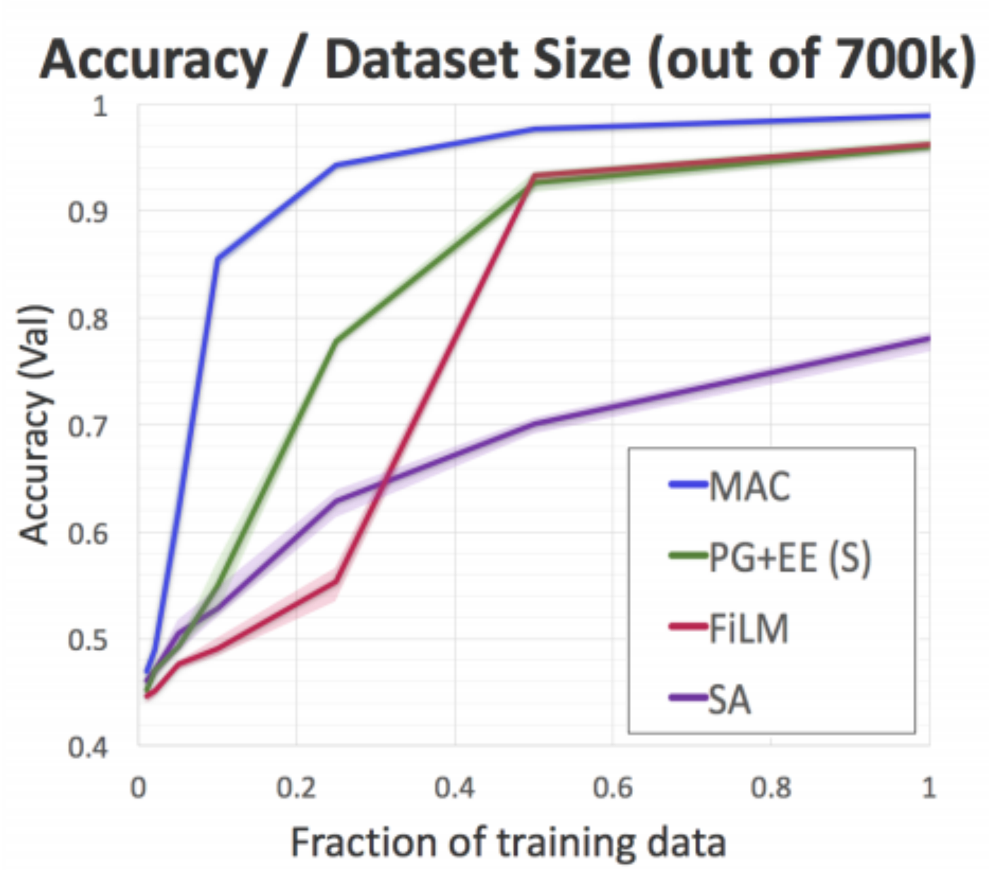
Last time: MAC (Memory, Attention, Control)

- Recurrent network with cell with read/write/control

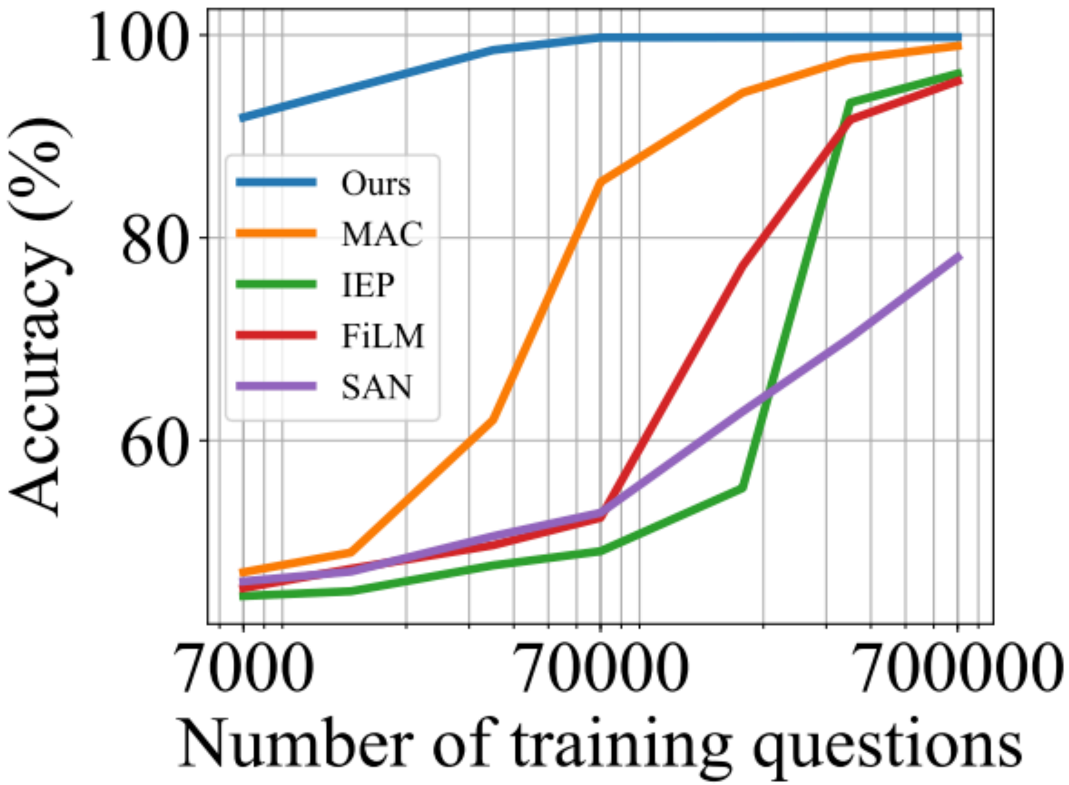


Compositional Attention Networks for Machine Reasoning, Hudson and Manning, ICLR 2018

Comparison of models (CLEVR, synthetic)



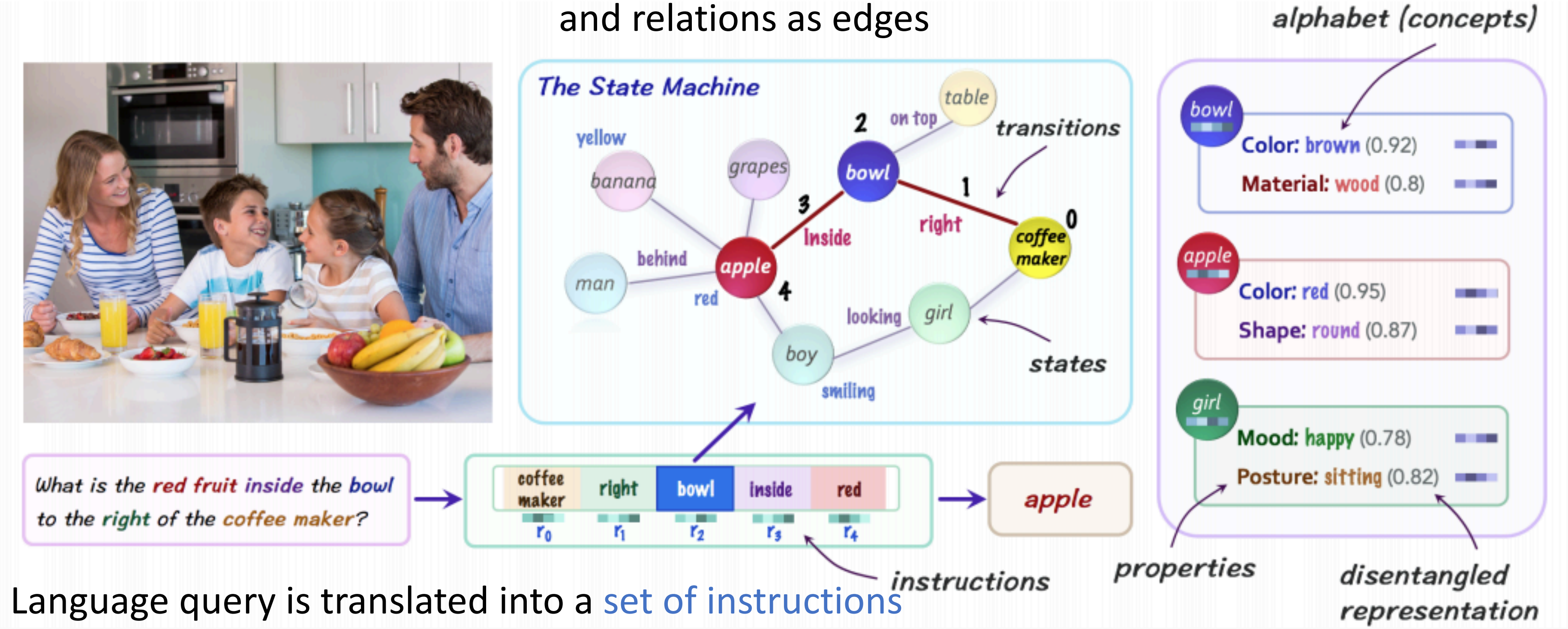
MAC [Hudson and Manning, ICLR 2018]



NS-VQA [Yi et al, NeurIPS 2018]

Last time: MAC/NSM on CLEVR/GQA

Scene graph with objects as nodes and relations as edges



Language query is translated into a set of instructions represented as vectors

Learned concept embeddings

Executing the query = going through the instructions step by step
At each timestep shift attention over the graph.
At the end, there is final state from which the answer is computed

Learning by Abstraction: The Neural State Machine, Hudson and Manning, NeurIPS 2019

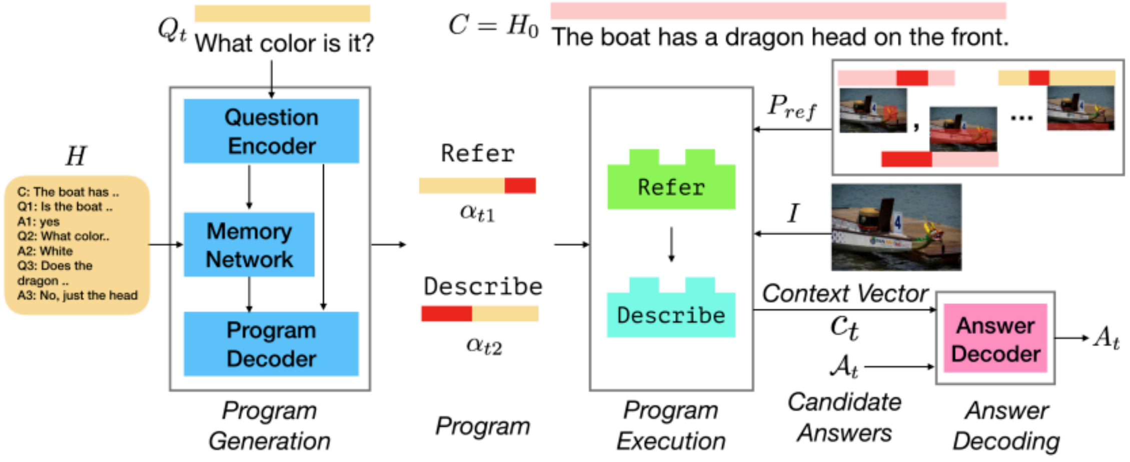
Semantic parsing vs MAC/NSM

- Neuro-symbolic models
  - Combines neural and symbolic (discrete symbols) representations
- MAC/NSM: Neural “computers” executing instructions
  - Instructions were also represented as embeddings
  - They are not ”symbolic” (converted into sequences of discrete symbols, i.e. programs)

- Are neuro-symbolic models the missing piece to general AI?

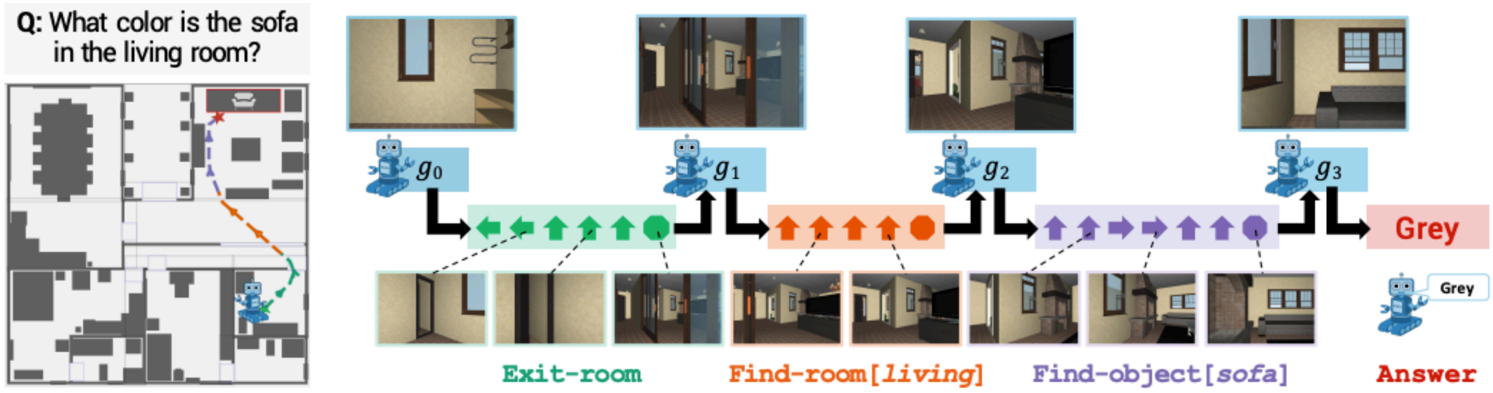
NMN for more complex VQA

- VQA with dialog and coreference



Visual coreference resolution in visual dialogue using neural module networks, Kottur et al, ECCV 2018

- Embodied QA



Neural modular control for embodied question answering, Das et al, CoRL 2018

Next time

- Paper presentations (3/1)
  - Learning to compose neural networks for question answering (Carolina)
  - Neural-Symbolic VQA: Disentangling Reasoning from Vision and Language Understanding (discussion)
- Thursday (3/4): Speaker listener models