Inferring and executing programs for Visual Reasoning

Justin Johnson, Bharath Hariharan, Laurens Maaten, Judy Hoffman,
Li Fei-Fei, C.Lawrence Zitnick, Ross Girshick
Stanford University, Facebook Research
International Conference on Computer Vision (ICCV 2017)

CPSC 532L presentation

Presented by:
Gursimran Singh
Borna Ghotbi
{msimar,bgotbi}@cs.ubc.ca
Visual question answering

- Generalizes well to new kinds of questions
  - who is wearing spectacles; how many people?

Q: How many spheres are the left of the big sphere and the same color as the small rubber cylinder?

Johnson, Justin, et al. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. (CVPR), 2017
Compositional visual reasoning

Q: How many spheres are the left of the **big sphere** and the same color as the small rubber cylinder?

Johnson, Justin, et al. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. (CVPR), 2017
Q: How many spheres are the left of the big sphere and the same color as the small rubber cylinder?
Q: How many spheres are the left of the big sphere and the same color as the small rubber cylinder?

Johnson, Justin, et al. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. (CVPR), 2017
Q: How many spheres are the left of the big sphere and the same color as the small rubber cylinder?
Compositional visual reasoning

Q: How many spheres are the left of the big sphere and the same color as the small rubber cylinder? A: 1

Johnson, Justin, et al. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. (CVPR), 2017
Standard VQA?

Q: How many spheres are the left of the big sphere and the same color as the small rubber cylinder?

LIMITATIONS

● Can’t model complex questions
Standard VQA?

Q: How many spheres are the **left** of the big sphere and the same color as the small rubber cylinder?

Q: How many spheres are the **right** of the big sphere and the same color as the small rubber cylinder?

LIMITATIONS

- Can’t model complex questions
- Lacks composition
**Standard VQA?**

**Q:** How many spheres are the **left** of the big sphere and the same color as the small rubber cylinder?

**Q:** How many spheres are the **right** of the big sphere and the same color as the small rubber cylinder?

**LIMITATIONS**
- Can’t model complex questions
- Lacks composition

Decompose the network into multiple modules
Standard VQA?

Q: How many spheres are the left of the big sphere and the same color as the small rubber cylinder?

Q: How many spheres are the right of the big sphere and the same color as the small rubber cylinder?

Q: How many objects are either red cylinders or metal objects?

LIMITATIONS

- Can’t model complex questions
- Lacks composition
- Uses same structure
Standard VQA?

Q: How many spheres are the **left** of the big sphere and the same color as the small rubber cylinder?

Q: How many spheres are the **right** of the big sphere and the same color as the small rubber cylinder?

Q: How many objects are either red cylinders or metal objects?

**LIMITATIONS**

- Can’t model complex questions
- Lacks composition
- Uses same structure

**Solution**

- Use composition and structure

Use separate networks for each question
Instead: consider a compositional model

Q: How many spheres are the left of the big sphere and the same color as the small rubber cylinder?

Q: How many spheres are the right of the big sphere and the same color as the small rubber cylinder?

Q: Is the big sphere the same material as the thing on the right of the cube?

Attributes identification
Counting objects
Comparisons
Spatial relationships
Logical operations

Common operations

Network architecture corresponding to the third question
Overview of approach

Graphics take from -> https://www.youtube.com/watch?v=3pCLma2FqSk
Overview of approach

Graphics take from -> https://www.youtube.com/watch?v=3pCLma2FqSk
Module networks

NLP Semantic Parser

Program Generator

Program

greater than

count

filter
shape=cube

cube

filter

count

color=yellow

yellow

scene

scene

Execution Engine

Answer: Yes

Graphics take from -> https://www.youtube.com/watch?v=3pCLma2FqSk
Module networks

NLP Semantic Parser

Program Generator

Question

Are
there
more
cubes
than
yellow
things

Answer: Yes

Execution Engine

Module Inventory

Classifier

greater than

count

count

filter shape= cube

cube

filter color= yellow

CNN

Module Network,
Andreas et al,
CVPR 2016

Graphics take from -> https://www.youtube.com/watch?v=3pCLma2FqSk
Modules recap

Attention

\[ \text{attend} : \text{Image} \rightarrow \text{Attention} \]

Re-attention

\[ \text{re-attend} : \text{Attention} \rightarrow \text{Attention} \]

Classification

\[ \text{classify} : \text{Image} \times \text{Attention} \rightarrow \text{Label} \]

Measurement

\[ \text{measure} : \text{Attention} \rightarrow \text{Label} \]
Module networks - limitations

Uses some pre-trained parser

NLP Semantic Parser

Answer: Yes

Execution Engine

Module Inventory

CNN

Module Network, Andreas et al, CVPR 2016

Graphics take from -> https://www.youtube.com/watch?v=3pCLma2FqSk
Inferring and executing programs

Trained end-end!!!
Inferring and executing programs

Graphics take from -> https://www.youtube.com/watch?v=3pCLma2FqSk
Execution engine
Modules architectures

<table>
<thead>
<tr>
<th>Layer</th>
<th>Output size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input image</td>
<td>$3 \times 224 \times 224$</td>
</tr>
<tr>
<td>ResNet-101 [14] conv4_6</td>
<td>$1024 \times 14 \times 14$</td>
</tr>
<tr>
<td>Conv($3 \times 3$, 1024 $\rightarrow$ 128)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>ReLU</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>Conv($3 \times 3$, 128 $\rightarrow$ 128)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>ReLU</td>
<td>$128 \times 14 \times 14$</td>
</tr>
</tbody>
</table>

- **a)** Visual feature extraction

<table>
<thead>
<tr>
<th>Index</th>
<th>Layer</th>
<th>Output size</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Previous module output</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(2)</td>
<td>Conv($3 \times 3$, 128 $\rightarrow$ 128)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(3)</td>
<td>ReLU</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(4)</td>
<td>Conv($3 \times 3$, 128 $\rightarrow$ 128)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(5)</td>
<td>Residual: Add (1) and (4)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(6)</td>
<td>ReLU</td>
<td>$128 \times 14 \times 14$</td>
</tr>
</tbody>
</table>

- **b.1)** Unary modules

<table>
<thead>
<tr>
<th>Index</th>
<th>Layer</th>
<th>Output size</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Previous module output</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(2)</td>
<td>Conv($3 \times 3$, 128 $\rightarrow$ 128)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(3)</td>
<td>ReLU</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(4)</td>
<td>Conv($3 \times 3$, 128 $\rightarrow$ 128)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(5)</td>
<td>Residual: Add (1) and (4)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
</tbody>
</table>

- **b.2)** Binary modules

<table>
<thead>
<tr>
<th>Index</th>
<th>Layer</th>
<th>Output size</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Previous module output</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(2)</td>
<td>Conv($3 \times 3$, 128 $\rightarrow$ 128)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(3)</td>
<td>ReLU</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(4)</td>
<td>Conv($3 \times 3$, 128 $\rightarrow$ 128)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>(5)</td>
<td>Residual: Add (1) and (4)</td>
<td>$128 \times 14 \times 14$</td>
</tr>
</tbody>
</table>

- **d)** Classifier

<table>
<thead>
<tr>
<th>Layer</th>
<th>Output size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final module output</td>
<td>$128 \times 14 \times 14$</td>
</tr>
<tr>
<td>Conv($1 \times 1$, 128 $\rightarrow$ 512)</td>
<td>$512 \times 14 \times 14$</td>
</tr>
<tr>
<td>ReLU</td>
<td>$512 \times 14 \times 14$</td>
</tr>
<tr>
<td>MaxPool($2 \times 2$, stride 2)</td>
<td>$512 \times 7 \times 7$</td>
</tr>
<tr>
<td>FullyConnected($512 \cdot 7 \cdot 7$ $\rightarrow$ 1024)</td>
<td>$1024$</td>
</tr>
<tr>
<td>ReLU</td>
<td>$1024$</td>
</tr>
<tr>
<td>FullyConnected($1024 \rightarrow</td>
<td>A</td>
</tr>
</tbody>
</table>
What do the modules learn?

Q: What shape is the... purple thing? blue thing? red thing right of the blue thing? red thing left of the blue thing?
A: cube sphere sphere cube

Q: How many cyan things are... right of the gray cube? left of the small cube? right of the gray cube and left of the small cube? right of the gray cube or left of the small cube?
A: 3 2 1 4

Figure 3. Visualizations of the norm of the gradient of the sum of the predicted answer scores with respect to the final feature map. From left to right, each question adds a module to the program; the new module is underlined in the question. The visualizations illustrate which objects the model attends to when performing the reasoning steps for question answering. Images are from the validation set.
Training

- Train Program Generator
- Freeze Program Generator, Train Execution Engine
- Finetune
Clever dataset

A training set of 70,000 images and 699,989 questions

● A validation set of 15,000 images and 149,991 questions

● A test set of 15,000 images and 14,988 questions

● Answers for all train and val questions

● Scene graph annotations for train and val images giving ground-truth locations, attributes, and relationships for objects

● Objects can be cubes, cylinders and spheres.
Experiments: Baselines

![Bar chart showing CLEVR overall accuracy for different models. Q-type mode: 42, LSTM: 47, CNN+LSTM: 54, CNN+LSTM+SA+MLP: 73, Human: 93, Ours-strong (700K prog.): 97.][1]

---

[1]: Image of the bar chart.
Experiments: Strongly and semi-supervised learning

![CLEVR Overall Accuracy Graph]

- Q-type mode: 42
- LSTM: 47
- CNN + LSTM: 54
- CNN+LSTM + SA+MLP: 73
- Human: 93
- Ours-strong (700K prog.): 97
- Ours-semi (18K prog.): 95
- Ours-semi (9K prog.): 89
Experiments

Generalizing to new attribute combinations

**Compositional Generalization Test (CoGenT)**

This data was used in Section 4.7 of the paper to study the ability of models to recognize novel combinations of attributes at test-time. The data is generated in two different conditions:

**Condition A**
- Cubes are **gray, blue, brown, or yellow**
- Cylinders are **red, green, purple, or cyan**
- Spheres can have any color

**Condition B**
- Cubes are **red, green, purple, or cyan**
- Cylinders are **gray, blue, brown, or yellow**
- Spheres can have any color

<table>
<thead>
<tr>
<th>Method</th>
<th>Train A A</th>
<th>Train A B</th>
<th>Finetune B A</th>
<th>Finetune B B</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>55.2</td>
<td>50.9</td>
<td>51.5</td>
<td>54.9</td>
</tr>
<tr>
<td>CNN+LSTM</td>
<td>63.7</td>
<td>57.0</td>
<td>58.3</td>
<td>61.1</td>
</tr>
<tr>
<td>CNN+LSTM+SA+MLP</td>
<td>80.3</td>
<td>68.7</td>
<td>75.7</td>
<td>75.8</td>
</tr>
<tr>
<td>Ours (18K prog.)</td>
<td><strong>96.6</strong></td>
<td><strong>73.7</strong></td>
<td><strong>76.1</strong></td>
<td><strong>92.7</strong></td>
</tr>
</tbody>
</table>

Accuracy over time for different methods.
Experiments

Generalizing to new question types

**Short**: all questions which their questions family has a mean program length less than 16

**Long**: otherwise

<table>
<thead>
<tr>
<th>Method</th>
<th>Train Short Short</th>
<th>Train Short Long</th>
<th>Finetune Both Short</th>
<th>Finetune Both Long</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>46.4</td>
<td>48.6</td>
<td>46.5</td>
<td>49.9</td>
</tr>
<tr>
<td>CNN+LSTM</td>
<td>54.0</td>
<td>52.8</td>
<td>54.3</td>
<td>54.2</td>
</tr>
<tr>
<td>CNN+LSTM+SA+MLP</td>
<td>74.2</td>
<td>64.3</td>
<td>74.2</td>
<td>67.8</td>
</tr>
<tr>
<td>Ours (25K prog.)</td>
<td><strong>95.9</strong></td>
<td>55.3</td>
<td><strong>95.6</strong></td>
<td><strong>77.8</strong></td>
</tr>
</tbody>
</table>

Table 2. Question answering accuracy on short and long CLEVR questions. **Left columns**: Models trained only on short questions; our model uses 25K ground-truth short programs. **Right columns**: Models trained on both short and long questions. Our model is trained on short questions then finetuned on the entire dataset; no ground-truth programs are used during finetuning.
Experiments

The CLEVR-Humans Dataset

- Use of questions that are hard to answer for a “smart robot”
- Filtered questions by asking three workers to answer them and removing those that a majority of workers answers incorrectly.
- About 17000 training questions and 7000 validation and test questions on CLEVR images.
Experiments

Human-composed questions

<table>
<thead>
<tr>
<th>Method</th>
<th>Train CLEVR</th>
<th>Train CLEVR, finetune human</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>27.5</td>
<td>36.5</td>
</tr>
<tr>
<td>CNN+LSTM</td>
<td>37.7</td>
<td>43.2</td>
</tr>
<tr>
<td>CNN+LSTM+SA+MLP</td>
<td>50.4</td>
<td>57.6</td>
</tr>
<tr>
<td>Ours (18K prog.)</td>
<td><strong>54.0</strong></td>
<td><strong>66.6</strong></td>
</tr>
</tbody>
</table>

Table 3. Question answering accuracy on the CLEVR-Humans test set of four models after training on just the CLEVR dataset (**left**) and after finetuning on the CLEVR-Humans dataset (**right**).
Results

Q: Is there a blue box in the items? A: yes

Predicted Program:
exist
filter.shape[cube]
filter.color[blue]
scene

Predicted Answer: ✓ yes

Q: What shape object is farthest right? A: cylinder

Predicted Program:
query.shape
unique
relate[right]
unique
filter.shape[cylinder]
filter.color[blue]
scene

Predicted Answer: ✓ cylinder

Q: Are all the balls small? A: no

Predicted Program:
equal.size
query.size
unique
filter.shape[sphere]
scene
query.size
unique
filter.shape[sphere]
filter.size[small]
scene

Predicted Answer: ✓ no

Q: Is the green block to the right of the yellow sphere? A: yes

Predicted Program:
exist
filter.shape[cube]
filter.color[green]
same.material
unique
filter.shape[sphere]
filter.color[yellow]
scene

Predicted Answer: ✓ yes

Q: Two items share a color, a material, and a shape; what is the size of the rightmost of those items? A: large

Predicted Program:
count
filter.shape[cube]
same.material
unique
filter.shape[cylinder]
scene

Predicted Answer: x 0
Other approaches

Andreas et al. ICCV 2017

Santoro et al. arXiv 2017
Strengths and weaknesses

Strengths

● Novel idea of using compositional reasoning to answer complex questions
● Train program generator on questions using LSTMs
● Training the whole network end to end

Weaknesses

● Not enough results on real images!
● More complex questions may not work properly
Future works/ possible improvements

Ideas taken from paper

● Adding ternary operations (if/else/then) and loops (for, do) to answer questions like “What color is the object with a unique shape?”.
● Control-flow operators could be incorporated into the framework
● Learning programs with limited supervision

Our ideas

● Using treeRNNs to synthesize programs
● Testing the whole framework on real images
Conclusion

● This method outperforms previous baselines.
● Neural module networks are a more natural way to reproduce reasoning step.
● More flexibility in the composition of the neural module network as modules have generic architectures.
References

- Inferring and Executing Programs for Visual Reasoning
- CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning
- Talk - https://www.youtube.com/watch?v=3pCLma2FqSk
- Learning to Reason: End-to-End Module Networks for Visual Question Answering
- https://github.com/facebookresearch/clevr-iep
- Deep Compositional Question Answering with Neural Module Networks
Thanks!
Visual question answering

- But does not really understand the question; same answer for
  - who is wearing hat? who is wearing?; wearing?
Standard VQA?

Q: How many spheres are the **left** of the big sphere and the same color as the small rubber cylinder?  

Q: How many spheres are the **right** of the big sphere and the same color as the small rubber cylinder?

Decompose the network!!