Working Memory Networks:
Augmenting Memory Networks with a Relational Reasoning Module

Juan Pavez, Héctor Allende, Héctor Allende-Cid
Reasoning for Question Answering

Reasoning is **crucial** for building systems that can dialogue with humans in natural language.

**Reasoning**: The process of forming conclusions, judgments, or inferences from facts or premises.

**Examples**:

- **Inferential Reasoning**: Premise 1, Premise 2 -> Conclusion
  - John is in the kitchen, John has the ball -> The ball is in the kitchen

- **Relational Reasoning**: Reason about relations between entities and their properties (Santoro et al.)

- **Causal Reasoning, Logical Reasoning, ...**
bAbI Dataset (Weston et al., 2015)

- One of the earliest datasets to measure the reasoning abilities of ML systems.

- **Synthetic.** Not NLP.

- Easy to evaluate different reasoning capabilities.

- **Noiseless tasks:** Separates reasoning analysis from natural language understanding.

- A thorough analysis can be found in (Lee et al., 2016)

<table>
<thead>
<tr>
<th>Category 2: Two Supporting Facts.</th>
</tr>
</thead>
<tbody>
<tr>
<td>01: Mary went to the kitchen.</td>
</tr>
<tr>
<td>02: Sandra journeyed to the office</td>
</tr>
<tr>
<td>03: Mary got the football there.</td>
</tr>
<tr>
<td>04: Mary travelled to the garden.</td>
</tr>
<tr>
<td>05: <em>Where is the football?</em> garden 3 4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category 4: Path Finding.</th>
</tr>
</thead>
<tbody>
<tr>
<td>01: The bedroom is south of the hallway.</td>
</tr>
<tr>
<td>02: The bathroom is east of the office.</td>
</tr>
<tr>
<td>03: The kitchen is west of the garden.</td>
</tr>
<tr>
<td>04: The garden is south of the office.</td>
</tr>
<tr>
<td>05: The office is south of the bedroom.</td>
</tr>
<tr>
<td>05: <em>How do you go from the garden to the bedroom?</em> n,n 4 5</td>
</tr>
</tbody>
</table>
**bAbI Dataset** (Weston et al., 2015)

- One of the earliest datasets to measure the reasoning abilities of ML systems.
- **Synthetic.** Not NLP.
- Easy to evaluate different reasoning capabilities.
- **Noiseless tasks:** Separates reasoning analysis from natural language understanding.
- A thorough analysis can be found in (Lee et al., 2016)

### Category 2: Two Supporting Facts.

| 01: Mary went to the kitchen. |
| 02: Sandra journeyed to the office |
| 03: Mary got the football there. |
| 04: Mary travelled to the garden. |
| 05: Where is the football? garden 3 4 |

**Has(Mary, Football), Is(Mary, Garden) → Is(Football, Garden)**

### Category 4: Path Finding.

| 01: The bedroom is south of the hallway. |
| 02: The bathroom is east of the office. |
| 03: The kitchen is west of the garden. |
| 04: The garden is south of the office. |
| 05: The office is south of the bedroom. |
| 05: How do you go from the garden to the bedroom?? n,n 4 5 |

---

Has(Mary, Football), Is(Mary, Garden) → Is(Football, Garden)
bAbI Dataset (Weston et al., 2015)

- One of the earliest datasets to measure the reasoning abilities of ML systems.
- **Synthetic.** Not NLP.
- Easy to evaluate different **reasoning capabilities.**
- **Noiseless tasks:** Separates reasoning analysis from natural language understanding.
- A thorough analysis can be found in (Lee et al., 2016)

### Category 2: Two Supporting Facts.

| 01: Mary went to the kitchen.  |
| 02: Sandra journeyed to the office |
| 03: Mary got the football there. |
| 04: Mary travelled to the garden. |
| 05: Where is the football? garden 3 4 |

### Category 4: Path Finding.

| 01: The bedroom is south of the hallway. |
| 02: The bathroom is east of the office. |
| 03: The kitchen is west of the garden. |
| 04: The garden is south of the office. |
| 05: The office is south of the bedroom. |
| 05: How do you go from the garden to the bedroom?? n,n 4 5 |

\[
S(\text{Garden, Office}), S(\text{Office, Bedroom}) \land N = S^{-1} \rightarrow N, N
\]
Memory Augmented Neural Networks

Memory Networks (Weston et al. 2014, Sukhbaatar et al. 2015)
Process a set of inputs and store them in memory. Then, at each hop, an important part of the memory is retrieved and used to retrieve more memories. Finally, the last retrieved memory is used to compute the answer.

\[
m_1 = w_{daniel} + w_{went} + \ldots
\]

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

\[
\mathbf{u} = w_{where} + w_{is} + \ldots
\]

q: Where is the football.

m_1
Memory Augmented Neural Networks

Memory Networks (Weston et al. 2014, Sukhbaatar et al. 2015)
Process a set of inputs and store them in memory. Then, at each hop, an important part of the memory is retrieved and used to retrieve more memories. Finally, the last retrieved memory is used to compute the answer.

\[ \alpha_i = \text{Softmax}(u^T m_i) \]
\[ o_1 = \sum_i \alpha_i m_i \]

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

q: Where is the football.
Memory Augmented Neural Networks

Memory Networks (Weston et al. 2014, Sukhbaatar et al. 2015)
Process a set of inputs and store them in memory. Then, at each hop, an important part of the memory is retrieved and used to retrieve more memories. Finally, the last retrieved memory is used to compute the answer.

\[
\alpha_i = \text{Softmax}(o_1^T m_i)
\]

\[
o_2 = \sum_{i} \alpha_i m_i
\]

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

q: Where is the football.
Memory Augmented Neural Networks

Memory Networks (Weston et al. 2014, Sukhbaatar et al. 2015)
Process a set of inputs and store them in memory. Then, at each hop, an important part of the memory is retrieved and used to retrieve more memories. Finally, the last retrieved memory is used to compute the answer.

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

q: Where is the football.

\[ a = \text{Softmax}(Wo_2) \]
Memory Augmented Neural Networks

Memory Networks (Weston et al. 2014, Sukhbaatar et al. 2015)
Process a set of inputs and store them in memory. Then, at each hop, an important part of memory is retrieved and used to retrieve more memories. Finally, the last retrieved memory is used to compute the answer.

\[ L(y, \hat{y}) = - \sum_i y_i \ln(\hat{y}_i) \]
Memory Augmented Neural Networks

Memory Networks (Weston et al. 2014, Sukhbaatar et al. 2015)

Some weaknesses:

- The attention mechanism is simple
- The attention mechanism relies on embeddings.
  - It may be nice to separate embedding learning from attention learning (modularization, reusability).
- The answer computation is too simple, it only uses one retrieved memory. Hard to see how can produce more complex reasoning based on memories.
Relational Neural Networks

Relation Networks (Santoro et al. 2017)
Neural Network with an inductive bias to learn pairwise relations of the input objects and their properties. A type of Graph Neural Networks.

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

q: Where is the football.
Relational Neural Networks

Relation Networks (Santoro et al. 2017)
Neural Network with an inductive bias to learn pairwise relations of the input objects and their properties. A type of Graph Neural Networks.

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

q: Where is the football.

memories

01
02
03
04

memories pairs with question

m1
m2
m3
m4

...
Relational Neural Networks

Relation Networks (Santoro et al. 2017)
Neural Network with an inductive bias to learn pairwise relations of the input objects and their properties. A type of Graph Neural Networks.

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

q: Where is the football.

\[
o_{i,j} = g_\theta([m_i; m_j; u])
\]
Relational Neural Networks

Relation Networks (Santoro et al. 2017)
Neural Network with an inductive bias to learn pairwise relations of the input objects and their properties. A type of Graph Neural Networks.

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

q: Where is the football.

\[ a = f_\phi \left( \sum_{i,j} o_{i,j} \right) \]
Relational Neural Networks

Relation Networks (Santoro et al. 2017)
Neural Network with an inductive bias to learn pairwise relations of the input objects and their properties. A type of Graph Neural Networks.

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

q: Where is the football.

\[ L(y, \hat{y}) = -\sum_i y_i \ln(\hat{y}_i) \]
Relational Neural Networks

Relation Networks (Santoro et al. 2017)

Some weaknesses:

- The model needs to process $N^2$ pairs where N is the number of memories.
  - 500 memories would require 250k backward and forward computations!
- Can not filter out unuseful objects that can produce **spurious relations**.
Working Memory Networks

**Working Memory Network** (Pavez et al., 2018)
A Memory Network model with a new working memory buffer and relational reasoning module. Produces state-of-the-art results in reasoning tasks. Inspired by the Multi-component model of working memory.

Short-term Memory Module

\[ m_1 = GRU([W_{\text{daniel}}; W_{\text{went}}; \ldots]) \]

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

Attention Module

\[ u = GRU([W_{\text{where}}; W_{\text{is}}; \ldots]) \]

q: Where is the football.

Reasoning Module
Working Memory Networks

Working Memory Network (Pavez et al., 2018)
A Memory Network model with a new working memory buffer and relational reasoning module. Produces state-of-the-art results in reasoning tasks. Inspired by the Multi-component model of working memory.

Multi-head attention (Vaswani et al. 2017)

\[ m_i^l = W_m^l m_i \]
\[ \alpha_i^l = \text{Softmax}(u^T m_i^l / \sqrt{d}) \]
\[ h_i = \sum_j \alpha_j^l m_j^l \]
\[ o_1 = [h_1^l; h_2^l; \ldots ] W_o \]
Working Memory Networks

A Memory Network model with a new working memory buffer and relational reasoning module. Produces state-of-the-art results in reasoning tasks. Inspired by the Multi-component model of working memory.

Multi-head attention (Vaswani et al. 2017)

\[ m_i^l = W^l_m m_i \]
\[ \alpha_i^l = \text{Softmax}(((f_i(o_1))^T m_i^l / \sqrt{d})) \]
\[ h_l = \sum_j \alpha_i^l m_j^l \]
\[ o_1 = [h_1; h_2; \ldots]W_o \]
01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

c: Where is the football.

q: Where is the football.
q: Where is the football.

\[ a = f_\phi \left( \sum_{i,j} g_\theta ([o_i; o_j; u]) \right) \]
Short-term Memory Module

01: Daniel went to the bathroom.
02: Sandra journeyed to the office.
03: Mary got the football there.
04: Mary travelled to the garden

Attention Module

\[ L(y, \hat{y}) = -\sum_i y_i \ln(\hat{y}_i) \]

Reasoning Module

q: Where is the football.
Results - Jointly trained bAbI-10k.

• Results on **jointly trained** bAbI-10k: Train a single model on all tasks simultaneously.

• Note that **EntNet** (Henaff et al.) solves all tasks in the **per-task version**: A single model for each task.
Results - Jointly trained bAbI-10k.

- Results on **jointly trained** bAbI-10k: Train a single model on all tasks simultaneously.

- Note that **EntNet** (Henaff et al.) solves all tasks in the **per-task version**: A single model for each task.

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemNN</td>
<td>92.5</td>
<td>Sukhbaatar et al.</td>
</tr>
<tr>
<td>MemNN-S</td>
<td>96.8</td>
<td>Sukhbaatar et al.</td>
</tr>
<tr>
<td>RN</td>
<td>?</td>
<td>Santoro et al.</td>
</tr>
<tr>
<td>SDNC</td>
<td>97</td>
<td>(Rae et al.)</td>
</tr>
<tr>
<td>WMemNN</td>
<td>18</td>
<td>(Pavez et al.)</td>
</tr>
<tr>
<td>WMemNN*</td>
<td>18</td>
<td>(Pavez et al.)</td>
</tr>
</tbody>
</table>
Results - Jointly trained bAbI-10k.

- Results on **jointly trained** bAbI-10k: Train a single model on all tasks simultaneously.

- Note that **EntNet** (Henaff et al.) solves all tasks in the **per-task version**: A single model for each task.
Results - Jointly trained bAbI-10k.

- Results on **jointly trained** bAbI-10k: Train a single model on all tasks simultaneously.

- Note that **EntNet** (Henaff et al.) solves all tasks in the **per-task version**: A single model for each task.

Accuracy

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy (Sukhbaatar et al.)</th>
<th>Accuracy (Santoro et al.)</th>
<th>Accuracy (Rae et al.)</th>
<th>Accuracy (Pavez et al.)</th>
<th>Accuracy (Pavez et al.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemNN</td>
<td>92.5</td>
<td>18</td>
<td>18</td>
<td>19</td>
<td>99.7</td>
</tr>
<tr>
<td>MemNN-S</td>
<td>96.8</td>
<td>?</td>
<td>97.2</td>
<td>99.6</td>
<td>99.7</td>
</tr>
<tr>
<td>RN</td>
<td>14</td>
<td>18</td>
<td>19</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>SDNC</td>
<td>?</td>
<td>97.2</td>
<td>?</td>
<td>99.6</td>
<td>99.7</td>
</tr>
<tr>
<td>WMemNN</td>
<td>92.5</td>
<td>?</td>
<td>99.6</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>WMemNN*</td>
<td>96.8</td>
<td>?</td>
<td>19</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>
Results - Jointly trained bAbI-10k.

- Results on **jointly trained** bAbI-10k: Train a single model on all tasks simultaneously.

- Note that **EntNet** (Henaff et al.) solves all tasks in the **per-task version**: A single model for each task.

Accuracy

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemNN (Sukhbaatar et al.)</td>
<td>92.5</td>
</tr>
<tr>
<td>MemNN-S (Sukhbaatar et al.)</td>
<td>96.8</td>
</tr>
<tr>
<td>RN (Santoro et al.)</td>
<td>?</td>
</tr>
<tr>
<td>SDNC (Rae et al.)</td>
<td>97.2</td>
</tr>
<tr>
<td>WMemNN (Pavez et al.)</td>
<td>99.6</td>
</tr>
<tr>
<td>WMemNN* (Pavez et al.)</td>
<td>99.7</td>
</tr>
</tbody>
</table>
## Ablations

<table>
<thead>
<tr>
<th>2 supporting facts</th>
<th>3 supporting facts</th>
<th>counting</th>
<th>basic induction</th>
<th>size reasoning</th>
<th>positional reasoning</th>
<th>path finding</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemNN (Sukhbaatar et al.)</td>
<td>99.0</td>
<td>93.2</td>
<td>94.4</td>
<td>99.2</td>
<td>92.0</td>
<td>59.2</td>
</tr>
<tr>
<td>MemNN(S) (Sukhbaatar et al.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RN (Santoro et al.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WMemNN (no multi-head)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WMemNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **complex attention patterns**
- **multiple relations**
## Ablations

<table>
<thead>
<tr>
<th></th>
<th>2 supporting facts</th>
<th>3 supporting facts</th>
<th>counting</th>
<th>basic induction</th>
<th>size reasoning</th>
<th>positional reasoning</th>
<th>path finding</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemNN</td>
<td>99.0</td>
<td>93.2</td>
<td>94.4</td>
<td>99.2</td>
<td>92.0</td>
<td>59.2</td>
<td>25.3</td>
</tr>
<tr>
<td>MemNN(S)</td>
<td>100.0</td>
<td>99.7</td>
<td>96.7</td>
<td>100.0</td>
<td>97.9</td>
<td>73.4</td>
<td>69.1</td>
</tr>
<tr>
<td>RN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WMemNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WMemNN (no multi-head)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **complex attention patterns**
- **multiple relations**
### Ablations

<table>
<thead>
<tr>
<th></th>
<th>2 supporting facts</th>
<th>3 supporting facts</th>
<th>counting</th>
<th>basic induction</th>
<th>size reasoning</th>
<th>positional reasoning</th>
<th>path finding</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MemNN</strong> (Sukhbaatar et al.)</td>
<td>99.0</td>
<td>93.2</td>
<td>94.4</td>
<td>99.2</td>
<td>92.0</td>
<td>59.2</td>
<td>25.3</td>
</tr>
<tr>
<td><strong>MemNN(S)</strong> (Sukhbaatar et al.)</td>
<td>100.0</td>
<td>99.7</td>
<td>96.7</td>
<td>100.0</td>
<td>97.9</td>
<td>73.4</td>
<td>69.1</td>
</tr>
<tr>
<td><strong>RN</strong> (Santoro et al.)</td>
<td>91.9</td>
<td>83.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>WMemNN</strong> (no multi-head)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>WMemNN</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**complex attention patterns**

**multiple relations**
## Ablations

<table>
<thead>
<tr>
<th></th>
<th>2 supporting facts</th>
<th>3 supporting facts</th>
<th>counting</th>
<th>basic induction</th>
<th>size reasoning</th>
<th>positional reasoning</th>
<th>path finding</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MemNN</strong> (Sukhbaatar et al.)</td>
<td>99.0</td>
<td>93.2</td>
<td>94.4</td>
<td>99.2</td>
<td>92.0</td>
<td>59.2</td>
<td>25.3</td>
</tr>
<tr>
<td><strong>MemNN(S)</strong> (Sukhbaatar et al.)</td>
<td>100.0</td>
<td>99.7</td>
<td>96.7</td>
<td>100.0</td>
<td>97.9</td>
<td>73.4</td>
<td>69.1</td>
</tr>
<tr>
<td><strong>RN</strong> (Santoro et al.)</td>
<td>91.9</td>
<td>83.5</td>
<td>97.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>WMemNN</strong> (no multi-head)</td>
<td>98.6</td>
<td>90.3</td>
<td>98.8</td>
<td>50.2</td>
<td>99.9</td>
<td>99.7</td>
<td>97.2</td>
</tr>
<tr>
<td><strong>WMemNN</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Ablations

### MemNN (Sukhbaatar et al.)
- 2 supporting facts: 99.0
- 3 supporting facts: 93.2
- Counting: 94.4
- Basic induction: 99.2
- Size reasoning: 92.0
- Positional reasoning: 59.2
- Path finding: 25.3

### MemNN(S) (Sukhbaatar et al.)
- 2 supporting facts: 100.0
- 3 supporting facts: 99.7
- Counting: 96.7
- Basic induction: 100.0
- Size reasoning: 97.9
- Positional reasoning: 73.4
- Path finding: 69.1

### RN (Santoro et al.)
- 2 supporting facts: 91.9
- 3 supporting facts: 83.5
- Counting: 97.9

### WMemNN (no multi-head)
- 2 supporting facts: 98.6
- 3 supporting facts: 90.3
- Counting: 98.8
- Basic induction: 50.2
- Size reasoning: 99.9
- Positional reasoning: 99.7
- Path finding: 97.2

### WMemNN
- 2 supporting facts: 99.3
- 3 supporting facts: 94.7
- Counting: 99.5
- Basic induction: 99.7
- Size reasoning: 99.6
- Positional reasoning: 99.9
- Path finding: 100.0

### Additional notes:
- Supporting facts and counting have a strong impact on performance.
- Basic induction and size reasoning are crucial for higher scores.
- Positional reasoning and path finding are less critical but still important.
Time comparison

- Time comparisons for a forward and backward pass for a single batch of size 32.
- For 30 memories there is a speedup of almost 20x.

![Graph showing wall time comparison between Relation Network and W-MemNN](image-url)
Conclusions

- We presented the **Working Memory Neural Network**, a Memory Network model augmented with a new **working memory buffer** and **relational reasoning module**.

- It retains the relational reasoning capabilities of the relation network while **reducing its computation times** considerably.

- We hope that this contribution may help applying the relation network in **larger problems**.
Conclusions

• It is a very general **framework**. We argue that it should include:

Embedding + Short-term storage
Conclusions

- It is a very general **framework**. We argue that it should include:

  - Embedding + Short-term storage
  - Attentional controller + Working memory buffer
Conclusions

- It is a very general **framework**. We argue that it should include:

  - Embedding + Short-term storage
  - Attentional controller + Working memory buffer
  - Reasoning module
Conclusions

• It is a very general **framework**. We argue that it should include:

- Embedding + Short-term storage
- Attentional controller + Working memory buffer
- Reasoning module

There is exactly one black triangle not touching any edge

**NLVR:** dev/test 65.6/65.8
Conclusions

• It is a very general **framework**. We argue that it should include:

- Embedding + Short-term storage
  - CNN
  - GRU
  - There is exactly one black triangle not touching any edge

- Attentional controller + Working memory buffer
  - Multi-head attention
  - biGRU
  - Scaled Dot-product Attention

- Reasoning module
  - Relational Reasoning Module
  - Attention Sum Reasoning Module

---

16. Why, what are YOUR shoes done with?
17. said the Gryphon.
18. I mean, what makes them so shiny?
19. Alice looked down at them, and considered a little before she gave her answer.
20. They are done with blacking, I believe.
21. Boots and shoes under the sea." the ______ went in a deep voice, are done with a whiting.
Conclusions

- It is a very general **framework**. We argue that it should include:

1. **Embedding + Short-term storage**
   - CNN
   - GRU
   - There is exactly one black triangle not touching any edge

2. **Attentional controller + Working memory buffer**
   - Multi-head attention

3. **Reasoning module**
   - BiGRU
   - Scaled Dot-product Attention

4. **Relational Reasoning Module**

---

16. Why, what are YOUR shoes done with?
17. said the Gryphon.
18. I mean, what makes them so shiny?
19. Alice looked down at them, and considered a little before she gave her answer.
20. They are done with blacking, I believe.

q. Boots and shoes under the sea. ’ the ______ went in a deep voice, are done with a whiting.
Conclusions

- It is a very general framework. We argue that it should include:

  - Embedding + Short-term storage
  - Attentional controller + Working memory buffer
  - Reasoning module

  ![Diagram with elements labeled](image)

---

16. Why, what are YOUR shoes done with?
17. said the Gryphon.
18. I mean, what makes them so shiny?
19. Alice looked down at them, and considered a little before she gave her answer.
20. They are done with blacking, I believe.

---

```
def input_module(x, u, adjacent=None, use_lstm=False, seq_lstm=True):...
```

```
def attention_module(memories, u, use_softmax=True, MLP=None):...
```

```
def reasoning_module(working_buffer):
  ... Implementation of the relational reasoning module ...
```

---

**NLVR:**

- dev/test: 65.6/65.8
Thanks!

juan.pavezs@alumnos.usm.cl

Code: https://github.com/jgpavez/Working-Memory-Networks

@juanpavez