Attention-based Model
External Memory

Input → DNN/RNN → output

Reading Head Controller

Reading Head

Machine’s Memory

Ref:
Reading Comprehension

Query → DNN/RNN → answer

Reading Head Controller

Semantic Analysis

Each sentence becomes a vector.
Sentence to vector can be jointly trained.

$\sum_{n=1}^{N} \alpha_n x^n$

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, Rob Fergus, “End-To-End Memory Networks”, NIPS, 2015
$\sum_{n=1}^{N} \alpha_n h^n$
Memory Network

Document

Compute attention

Extract information

Compute attention

Extract information

\[ \sum \rightarrow \text{DNN} \]

\[ q \rightarrow \sum \rightarrow \text{DNN} \rightarrow a \]
Multiple-hop


The position of reading head:

<table>
<thead>
<tr>
<th>Story (16: basic induction)</th>
<th>Support</th>
<th>Hop 1</th>
<th>Hop 2</th>
<th>Hop 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brian is a frog.</td>
<td>yes</td>
<td>0.00</td>
<td>0.98</td>
<td>0.00</td>
</tr>
<tr>
<td>Lily is gray.</td>
<td></td>
<td>0.07</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Brian is yellow.</td>
<td>yes</td>
<td>0.07</td>
<td>0.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Julius is green.</td>
<td></td>
<td>0.06</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Greg is a frog.</td>
<td>yes</td>
<td>0.76</td>
<td>0.02</td>
<td>0.00</td>
</tr>
</tbody>
</table>

What color is Greg? Answer: yellow Prediction: yellow

Keras has example:
https://github.com/fchollet/keras/blob/master/examples/babi_memnn.py
Multiple-hop

**ReasoNet**
https://arxiv.org/abs/1609.05284
Bi-directional Attention Flow
Fully-Aware Fusion Network

Fully-Aware Self-Boosted Fusion

Context Understanding

Question Understanding

The above can be used to capture long range info.

Fully-Aware Multi-level Fusion

Context Understanding

Question Understanding

High-level Concept

Low-level Concept

Input Vector

Context

Question

BilSTM

Concat w/ Distant C Infor.

7-Level Fully-Aware Attention

Concat w/ Multi-level Q Infor.

3-Level Fully-Aware Attention
Visual Question Answering

What is the mustache made of?

source: http://visualqa.org/
Visual Question Answering

Query → DNN/RNN → answer

Reading Head Controller

CNN

A vector for each region
Visual Question Answering

Visual Question Answering

External Memory v2

Input → DNN/RNN → output

Reading Head Controller → Writing Head

Writing Head Controller → Reading Head

……

Machine’s Memory

Neural Turing Machine
Neural Turing Machine

\[ r^0 = \sum \hat{\alpha}_0^i m_0^i \]

Retrieval process
Neural Turing Machine

\[ r^0 = \sum \hat{\alpha}_0^i m_0^i \]

\[ \alpha_1^i = \cos(m_0^i, k^1) \]
Neural Turing Machine

\[ m_i^1 = m_i^0 - \alpha_i^1 \odot m_i^0 + \alpha_i^1 a^1 \] (element-wise)

\[ k^1 \]

\[ e^1 \]

\[ a^1 \]

0 ~ 1
Neural Turing Machine

\[ h^0 \rightarrow f \rightarrow h^1 \rightarrow f \rightarrow y^1 \]

\[ x^1 \rightarrow r^0 \rightarrow y^2 \]

\[ x^2 \rightarrow r^1 \]

\[ \hat{\alpha}_0^1, \hat{\alpha}_0^2, \hat{\alpha}_0^3, \hat{\alpha}_0^4 \]

\[ m_0^1, m_0^2, m_0^3, m_0^4 \]

\[ \hat{\alpha}_1^1, \hat{\alpha}_1^2, \hat{\alpha}_1^3, \hat{\alpha}_1^4 \]

\[ m_1^1, m_1^2, m_1^3, m_1^4 \]

\[ \hat{\alpha}_2^1, \hat{\alpha}_2^2, \hat{\alpha}_2^3, \hat{\alpha}_2^4 \]

\[ m_2^1, m_2^2, m_2^3, m_2^4 \]
Wei-Jen Ko, Bo-Hsiang Tseng, Hung-yi Lee, "Recurrent Neural Network based Language Modeling with Controllable External Memory", ICASSP, 2017
Stack RNN

Armand Joulin, Tomas Mikolov, Inferring Algorithmic Patterns with Stack-Augmented Recurrent Nets, arXiv Pre-Print, 2015