Meta-Learning with Memory-Augmented Neural Networks
Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, Timothy Lillicrap
ICML 2016

Reviewed by: Jack Lanchantin

1Department of Computer Science, University of Virginia
https://qdata.github.io/deep2Read/
Meta-learning

- Scenario in which an agent learns at two levels
  - Rapid learning occurs within a task, for example, when learning to accurately classify within a particular dataset.
  - This learning is guided by knowledge accrued more gradually across tasks, which captures the way in which task structure varies across target domains.
- Given its two-tiered organization, meta-learning is often described as “learning to learn.”
META-LEARNING TASK METHODOLOGY

- Usually we try to choose parameters $\theta$ to minimize loss $\mathcal{L}$ across dataset $D$.

- In meta-learning, we choose parameters to reduce the expected loss across a distribution of datasets $p(D)$:

$$
\theta^* = \arg\min_{\theta} \mathbb{E}_{D \sim p(D)}[\mathcal{L}(D; \theta)]
$$
Setup for This Paper

- Dataset $D = \{d_t\}_{t=1}^T = \{(x_t, y_t)\}_{t=1}^T$
- At each timestep $t$, the network receives input $x_t$ as well as the label of the previous example, $y_{t-1}$:
  - $(x_1, \text{null}), (x_2, y_1), \ldots, (x_T, y_{T-1})$
- Labels, classes, and samples are shuffled in each training “episode”.

![Diagram showing class prediction, shuffle, and episode structure](image)
Memory-Augmented Neural Nets (MANNs)

- Learns to hold samples in memory until the correct labels are shown, after which they can be bound and stored for later use.
MANN Reading

Given input $x_t$, controller (LSTM) produces key $k_t$.

$M_t$ is addressed using cosine similarity:

$$K(k_t, M_t(i)) = \frac{k_t \cdot M_t(i)}{||k_t|| \cdot ||c_t(i)||},$$

which is used to produce read-weight vector $w^r_t$:

$$w^r_t(i) \leftarrow \frac{\exp(K(k_t, M_t(i)))}{\sum_j K(k_t, M_t(j))}.$$

A certain memory $r_t$ is read using this read-weight vector:

$$r_t \leftarrow \sum_i w^r_t(i) M_t(i).$$
LRUA: Content-based writer that writes memories to either the least used or most recently used memory location.

Usage weights:

\[ w^u_t \leftarrow \gamma w^u_{t-1} + w^r_t + w^w_t \]

Least used weight:

\[ w^{lu}_t(i) = \begin{cases} 
0, & \text{if } w^u_t(i) > m(w^u_t, n) \\
1, & \text{if } w^u_t(i) \leq m(w^u_t, n) 
\end{cases} \]

\( m(v, n) = n^{th} \text{ smallest element of vector } v \)
Write weight $w_t^w$:

$$w_t^w \leftarrow \sigma(\alpha) w_{t-1}^r + (1 - \sigma(\alpha)) w_{t-1}^{lu}$$

Writing to $M$

$$M_t(i) \leftarrow M_{t-1}(i) + w_t^w(i) k_t, \forall i$$
Omniglot Experiment Results

LSTM

MANN

Graphs showing the percent correct over episodes for 1st, 2nd, 5th, and 10th instance for both LSTM and MANN models.
Omniglot Experiment Results

*Table 1.* Test-set classification accuracies for humans compared to machine algorithms trained on the Omniglot dataset, using one-hot encodings of labels and five classes presented per episode.

<table>
<thead>
<tr>
<th>Model</th>
<th>1\textsuperscript{st}</th>
<th>2\textsuperscript{nd}</th>
<th>3\textsuperscript{rd}</th>
<th>4\textsuperscript{th}</th>
<th>5\textsuperscript{th}</th>
<th>10\textsuperscript{th}</th>
</tr>
</thead>
<tbody>
<tr>
<td>HUMAN</td>
<td>34.5</td>
<td>57.3</td>
<td>70.1</td>
<td>71.8</td>
<td>81.4</td>
<td>92.4</td>
</tr>
<tr>
<td>FEEDFORWARD</td>
<td>24.4</td>
<td>19.6</td>
<td>21.1</td>
<td>19.9</td>
<td>22.8</td>
<td>19.5</td>
</tr>
<tr>
<td>LSTM</td>
<td>24.4</td>
<td>49.5</td>
<td>55.3</td>
<td>61.0</td>
<td>63.6</td>
<td>62.5</td>
</tr>
<tr>
<td>MANN</td>
<td>36.4</td>
<td>82.8</td>
<td>91.0</td>
<td>92.6</td>
<td>94.9</td>
<td>98.1</td>
</tr>
</tbody>
</table>