Differentiable Neural Computers with Memory Demon

Ari Azarafrooz
ari.azarafrooz@gmail.com

Abstract

A Differentiable Neural Computer (DNC) Graves et al. [2016] is a neural network with an external memory which allows for iterative content modification via read, write and delete operations. We show that information theoretic properties of the memory contents play an important role in the performance of such architectures. We introduce a novel concept of *memory demon* [1] to DNC architectures which modifies the memory contents implicitly via additive input encoding. The goal of the memory demon is to maximize the expected sum of mutual information of the consecutive external memory contents.

Github codes https://github.com/azarafrooz/dnc-with-demon

1 Introduction

A Differentiable Neural Computer (DNC) Graves et al. [2016] is a neural network coupled to an external memory matrix $\boldsymbol{M} \in \mathbb{R}^{N \times W}$ where W is the word/cell length and N is the number of cells and independent of the number of training parameters. Previous researches have shown external memory matrix provide proper architectural bias for solving algorithmic and structured tasks.

The neural network in DNC is referred to as the 'controller'. It interacts with the external world via input $\boldsymbol{x}_t \in \mathbb{R}$ and outputs $\boldsymbol{y}$. But it also interacts with the memory matrix at each time step $\boldsymbol{M}_t$ using read and write heads to read from $\boldsymbol{M}_t$ and to arrive at the next memory matrix $\boldsymbol{M}_{t+1}$.

One main mechanism to allow for the interaction between heads and the memory is content-based addressing. In this mechanism, a key vector is emitted by the controller as an approximation to a part of the stored data which is then compared to memory $\boldsymbol{M}$ to yield the exact value. The comparison metric is a cosine similarity score that determines weightings that should be used by the read headers or by the write header to modify the memory content.

Keep memories interesting A technique common between Memory Champions is to assign interesting imageries to the subjects of memorization, also known as the memory palace Foer [2012]. For example, a certain number is an interesting event or picture. Interesting imageries seem to make the memorization/recall easier. Keeping memory associations interesting, seems to be an efficient strategy to reduce the interference of current and past information and avoids the memory contents to be overwritten or misinterpreted. Inspired by such a technique and in the context of DNC, we setup a Reinforcement Learning (RL) framework where a RL agent is equipped with a mutual information-based reward and an encoding action. Mutual information-based reward $I(\boldsymbol{M}_t; \boldsymbol{M}_{t+1}) = H(\boldsymbol{M}_t) + H(\boldsymbol{M}_{t+1}) - H(\boldsymbol{M}_t, \boldsymbol{M}_{t+1})$ serves as a proxy to measure "interestingness". The intuition is that, if the DNC dynamics are "too simple", then I will be small because both entropy terms $H(\boldsymbol{M}_t)$ and $H(\boldsymbol{M}_{t+1})$ are small. On the other hand, if the DNC dynamics are "too random", then I will be small because $H(\boldsymbol{M}_t, \boldsymbol{M}_{t+1}) \approx H(\boldsymbol{M}_t) + H(\boldsymbol{M}_{t+1})$. As a

[1] The name is inspired by the concept of Maxwell's Demon who decreases the *entropy* of gas in a box by letting all the high-velocity molecules accumulate on one side and all the low-velocity ones on the other.

NeurIPS 2022 Workshop On Memory in Artificial and Real Intelligence.

result, "interesting" non-randomness will exist only in the intermediate regime with high value of $I(\boldsymbol{M}_t; \boldsymbol{M}_{t+1})$. The RL agent action is to assign to each input embedding a proper encoding, akin to assigning a certain imagery for each subject of memorization.

2 DNC with Memory demon

At each time step, demon modifies samples via continuous embedding $\mathbf{a}_t \sim \pi(\boldsymbol{s}_t)$ which gets added to the input data $\boldsymbol{x}_t$ where $\boldsymbol{s}_t = (\boldsymbol{x}_t, \boldsymbol{M}_t)$. As a result, the input of the controller is $\boldsymbol{x}_t + \mathbf{a}_t$ which leads to the writing the next memory content $\boldsymbol{M}_{t+1}$. One can view $\mathbf{a}_t$ as interestingness encoding (rather than a positional encoding.) This iterative process is visualized in figure 1.

Demon then computes the mutual information of the consecutive memory contents $I(\boldsymbol{M}_t; \boldsymbol{M}_{t+1})$ as its reward and updates its policy π in order maximize the expected sum of rewards $\sum_t \mathbb{E}_{(\boldsymbol{s}_t, \mathbf{a}_t) \sim \rho_\pi} I(\boldsymbol{M}_t; \boldsymbol{M}_{t+1})$.

In order to estimate mutual information of consecutive memory contents, we utilize Belghazi et al. [2018]. RL agent then uses this measure to guide its policy encoding actions using PPO Schulman et al. [2017].

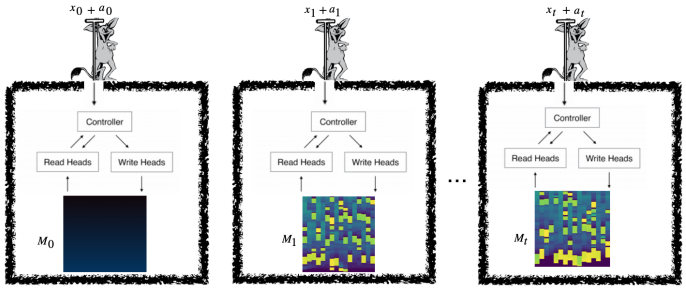

Figure 1: Memory Demon encodes the observations with additive encodings $\mathbf{a}_t$ and receives the mutual information of the consecutive memory contents $I(\boldsymbol{M}_t; \boldsymbol{M}_{t+1})$ as reward.

3 Results

We refer to our architecture as Demon-DNC. The most important observation was that we found out that Demon-DNC works best when combined with *(key) masking techniques* of Csordás and Schmidhuber [2019]. One known issue with content-based memory addressing is that the entire key and cell values are being used to produce the similarity score. This can flatten the memory address distribution when the unknown part of the cell values are more significant. The dynamic and high variance nature of RL agent policies seem to exacerbate this situation which in turn makes the role of masking more important.

To measure the efficacy of the Demon-DNC, we utilize the 3 following common tasks in the Csordás and Schmidhuber [2019]. Please refer to the description of such tasks to Csordás and Schmidhuber [2019] for a more comprehensive description.

3.1 Associated-recall and Copy task

Fig. 6 shows that Demo-DNC with masking is more efficient than the rest in both tasks. Although the improvement (in terms of the convergence) seems to be more significant for the Associated-recall task than for the copy task. Moreover, it shows the drops in the mutual information loss measures are consistent with drop in the mean test errors. Networks that have lower mean test errors have also lower mutual information loss, confirming our main assertion on the role of information theoretic properties of the memory contents.

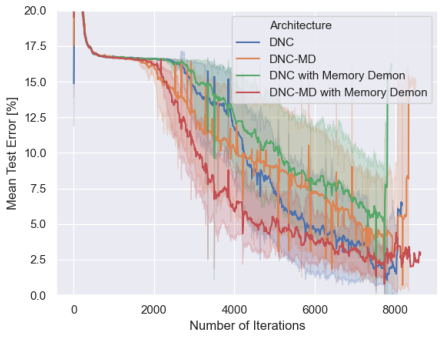

Figure 2: Effect of memory demon on the convergence: Associative Recall

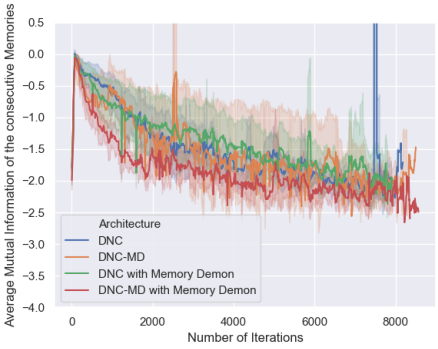

Figure 3: Effect of memory demon on the mutual information of consecutive memories: Associative Recall

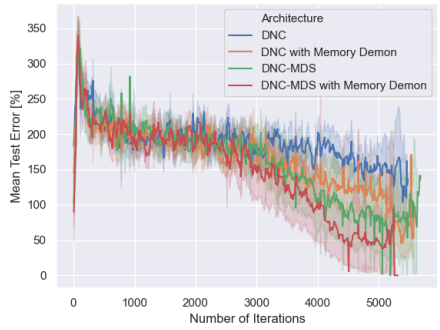

Figure 4: Effect of memory demon on the convergence: Repeat-Copy

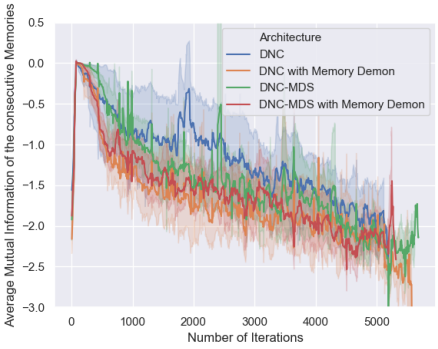

Figure 5: Effect of memory demon on the mutual information of consecutive memories: Repeat-Copy

Figure 6: Associative Recall and Repeat-Copy experiment over 10 seeds.

3.2 bAbi-10 task

We also conducted the bAbI experiments and the results are reported in Table 1. As it can be seen, given the same number of training steps, not only Demon-DNC outperforms the best DNC network in mean, it also outperform it in 9 tasks.

4 Shortcomming and future work

4.1 Comparing with some of the state of art

We also experimented on bAbi-1K. Since we found no available numbers on previous DNC-networks, we only compare it with UT Dehghani et al. [2019] in Table 2 and Table 3. It can be seen that after 0.5M iterations, UT outperforms Demon-DNC. However, the full potential of external memories might have not been explored fully. The idea of optimizing the information theoretic properties of memory contents (with or without RL agents) for example might inspire other fresh ideas for networks with external memory.

4.2 Scalability

One attractive properties of DNC is that number of *parameters* is independent of the number of memory cells N. However, this is not true for the information estimator and RL networks. We are exploring the feasibility of integrating our framework with Sparse DNC Rae et al. [2016] to address this issue.

Table 1: 10K-bAbI error rates of different models after 0.5M iterations of training [%]

Task	DNC	DNC-MDS	DNC-DS	DNC-MS	DNC-MD	DNC-MD-Demon
1	2.5 ± 4.4	0.4 ± 1.2	0.7 ± 1.6	0.0 ± 0.1	$\mathbf{0.0 \pm 0.0}$	$\mathbf{0.0 \pm 0.0}$
2	29.0 ± 19.4	8.6 ± 10.1	18.6 ± 15.1	7.8 ± 5.9	6.9 ± 4.7	$\mathbf{1.54 \pm 0.57}$
3	32.3 ± 14.7	10.8 ± 9.5	16.9 ± 13.0	$\mathbf{7.9 \pm 7.8}$	12.4 ± 5.1	8.34 ± 0.81
4	0.8 ± 1.5	0.8 ± 1.5	6.4 ± 10.0	0.8 ± 1.0	$\mathbf{0.1 \pm 0.2}$	0.2 ± 0.3
5	1.5 ± 0.6	1.6 ± 1.0	1.3 ± 0.5	1.7 ± 1.1	1.3 ± 0.7	$\mathbf{0.97 \pm 0.17}$
6	5.2 ± 6.8	1.1 ± 2.1	2.4 ± 3.8	$\mathbf{0.0 \pm 0.1}$	0.1 ± 0.1	0.1 ± 0.1
7	8.8 ± 5.8	3.4 ± 2.3	7.6 ± 5.1	2.5 ± 2.0	3.0 ± 5.0	$\mathbf{1.24 \pm 1.04}$
8	11.6 ± 9.4	4.6 ± 4.5	10.9 ± 7.9	$\mathbf{1.8 \pm 1.6}$	2.5 ± 2.1	1.91 ± 1.19
9	4.5 ± 5.8	0.8 ± 1.9	2.0 ± 3.3	$\mathbf{0.1 \pm 0.2}$	0.1 ± 0.2	$\mathbf{0.06 \pm 0.06}$
10	9.1 ± 11.5	2.6 ± 3.9	4.1 ± 5.9	0.6 ± 0.6	0.5 ± 0.5	$\mathbf{0.33 \pm 0.27}$
11	11.6 ± 9.4	0.1 ± 0.1	0.1 ± 0.2	$\mathbf{0.0 \pm 0.0}$	$\mathbf{0.0 \pm 0.0}$	$\mathbf{0.0 \pm 0.0}$
12	1.1 ± 0.8	$\mathbf{0.2 \pm 0.2}$	0.5 ± 0.4	0.3 ± 0.4	0.2 ± 0.2	$\mathbf{0.03 \pm 0.07}$
13	1.1 ± 0.8	$\mathbf{0.1 \pm 0.1}$	0.2 ± 0.2	0.2 ± 0.2	$\mathbf{0.1 \pm 0.1}$	0.16 ± 0.34
14	24.8 ± 22.5	8.0 ± 13.1	20.0 ± 19.4	1.8 ± 0.9	2.0 ± 1.6	$\mathbf{1.97 \pm 0.73}$
15	40.8 ± 1.4	26.3 ± 20.7	42.1 ± 6.3	33.0 ± 15.1	23.6 ± 18.6	$\mathbf{6.46 \pm 12.32}$
16	$\mathbf{53.1 \pm 1.2}$	54.5 ± 1.8	53.5 ± 1.4	53.2 ± 2.3	53.9 ± 1.2	53.28 ± 2.18
17	$\mathbf{37.8 \pm 2.5}$	39.9 ± 3.2	40.1 ± 2.0	41.2 ± 3.0	39.8 ± 1.2	39.14 ± 1.89
18	7.0 ± 3.0	6.3 ± 4.1	9.4 ± 0.9	3.3 ± 2.2	2.0 ± 2.6	$\mathbf{1.67 \pm 1.64}$
19	67.6 ± 8.6	48.6 ± 32.8	67.6 ± 7.9	48.1 ± 26.7	40.7 ± 34.9	$\mathbf{14.08 \pm 2.87}$
20	$\mathbf{0.0 \pm 0.0}$	0.9 ± 0.9	1.5 ± 1.0	5.3 ± 12.5	0.1 ± 0.1	0.06 ± 0.06
mean	16.9 ± 5.2	11.0 ± 3.8	15.3 ± 3.5	10.5 ± 1.9	9.5 ± 1.6	$\mathbf{6.58 \pm 0.73}$

Table 2: 1K-bAbI error rates of different models

Model	Error
DNC-MD-Demon	9.98 $\mathbf{(7/20)}$
Universal Transformer (UT)	$\mathbf{8.50}$ $(8/20)$

Table 3: 1K-bAbI error rates of different models across tasks

Task	UT with dynamic halting	DNC-MD-Demon
1	$\mathbf{0.0}$	$\mathbf{0.0}$
2	$\mathbf{0.5}$	27.34
3	$\mathbf{5.4}$	24.12
4	$\mathbf{0.0}$	2.2
5	$\mathbf{0.5}$	0.8
6	$\mathbf{0.5}$	0.7
7	$\mathbf{3.2}$	7.54
8	$\mathbf{1.6}$	2.16
9	$\mathbf{0.2}$	0.3
10	$\mathbf{0.4}$	1.21
11	$\mathbf{0.1}$	$\mathbf{0.1}$
12	$\mathbf{0.0}$	$\mathbf{0.0}$
13	0.6	$\mathbf{0.2}$
14	3.8	$\mathbf{0.6}$
15	5.9	$\mathbf{0.0}$
16	$\mathbf{15.4}$	53.75
17	42.9	$\mathbf{28.63}$
18	$\mathbf{4.1}$	5.62
19	68.2	$\mathbf{44.24}$
20	2.4	$\mathbf{0.0}$

References

Ishmael Belghazi, Sai Rajeswar, A. Baratin, R. Devon Hjelm, and Aaron C. Courville. Mine: Mutual information neural estimation. *ArXiv*, abs/1801.04062, 2018.

R. Csordás and J. Schmidhuber. Improving differentiable neural computers through memory masking, de-allocation, and link distribution sharpness control. *ArXiv*, abs/1904.10278, 2019.

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Lukasz Kaiser. Universal transformers. *ArXiv*, abs/1807.03819, 2019.

Joshua Foer. *Moonwalking with Einstein: The Art and Science of Remembering Everything*. 2012.

Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, AdriàPuigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen King, Christopher Summerfield, Phil Blunsom, Koray Kavukcuoglu, and Demis Hassabis. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626): 471–476, 2016.

Jack W. Rae, Jonathan J. Hunt, Ivo Danihelka, Tim Harley, A. Senior, Greg Wayne, A. Graves, and T. Lillicrap. Scaling memory-augmented neural networks with sparse reads and writes. In *NIPS*, 2016.

John Schulman, F. Wolski, Prafulla Dhariwal, A. Radford, and Oleg Klimov. Proximal policy optimization algorithms. *ArXiv*, abs/1707.06347, 2017.