Subtractive gating improves generalization in working memory tasks

Milton Llera Montero (m.lleramontero@bristol.ac.uk)
School of Psychological Science and
Computational Neuroscience Unit, SCEEM
University of Bristol
Bristol, United Kingdom

Gaurav Malhotra (gaurav.malhotra@bristol.ac.uk)
School of Psychological Science, University of Bristol
Bristol, United Kingdom

Jeff Bowers[1] (j.bowers@bristol.ac.uk)
School of Psychological Science, University of Bristol
Bristol, United Kingdom

Rui Ponte Costa[1] (rui.costa@bristol.ac.uk)
Computational Neuroscience Unit, SCEEM University of Bristol
Bristol, United Kingdom

Abstract

It is largely unclear how the brain learns to generalize to new situations. Although deep learning models offer great promise as potential models of the brain, they break down when tested on novel conditions not present in their training datasets. One of the most successful models in machine learning are gated-recurrent neural networks. Because of its working memory properties here we refer to these networks as working memory networks (WMN). We compare WMNs with a biologically motivated variant of these networks. In contrast to the multiplicative gating used by WMNs, this new variant operates via subtracting gating (subWMN). We tested these two models in a range of working memory tasks: orientation recall with distractors, orientation recall with update/addition and distractors, and a more challenging task: sequence recognition based on the machine learning handwritten digits dataset. We evaluated the generalization properties of these two networks in working memory tasks by measuring how well they copped with three working memory loads: memory maintenance over time, making memories distractor-resistant and memory updating. Across these tests subWMNs perform better and more robustly than WMNs. These results suggests that the brain may rely on subtractive gating for improved generalization in working memory tasks.

Keywords: Gating; Working memory; Recurrent neural networks; Cortical Circuits

Introduction

Deep Learning models loosely based on neuroscientific principles are generating increased interest in fields like psychology and neuroscience as a way of modelling various cognitive and neuroscientific phenomena (Hassabis et al. (2017)).

The mechanisms underlying humans ability to generalize to new situations have remained largely unclear and far outperforms current machine learning neural networks (Lake et al. (2017)).

Current machine learning recurrent networks rely heavily on the use of gating to solve different tasks, such as used in state-of-the-art models of language. In neuroscience there has been a long standing debate on whether the brain relies on subtractive or divisive gating (Doiron et al. (2001); Mejias et al. (2013); El Boustani and Sur (2014); Seybold et al. (2015)). Recently, we proposed a mapping between gated recurrent neural networks (RNNs) and cortical microcircuits observed across the brain (Costa et al. (2017)) where the key difference is that networks operate with subtractive rather than multiplicative gates as found in vanilla machine learning gated-RNNs (Fig. 1).

In this study we compare these two forms of gating (subtractive and multiplicative) in three working memory tasks: (i) orientation recall task with distractors (Manohar et al. (2019)), (ii) orientation recall with addition/update and distractors (Fallon et al. (2018)), and (iii) a more challenging sequence recognition task. We studied the working memory generalization by testing the working memory load in three ways that were not presented during training: (i) memory maintenance (by adding more distractors), (ii) distractor-resistance (by increasing the strength of the distractors) and (iii) memory updating (using the recall with addition task).

Results

Our preliminary findings show that using subtractive instead of multiplicative gating in memory units can have important effects on the way models learn and perform on test data and extrapolate to new settings for different working memory problems. These results can be summarized into two groups: the effects of different gating mechanisms in the training convergence of the networks, and the performance they achieve in testing or extrapolation datasets. Below we discuss the different tasks used and the respective results.

[1] Co-senior authors.

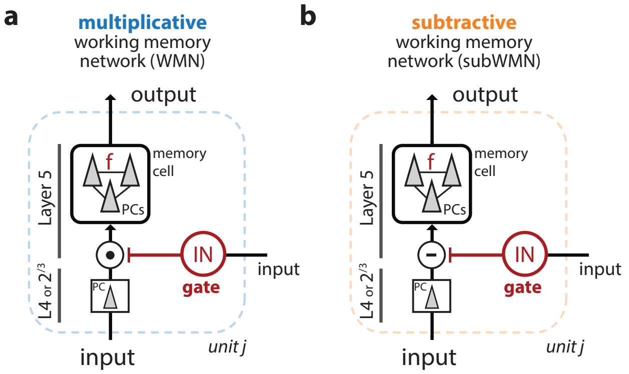

Figure 1: **Simplified illustration of the working memory units.** (**a**) Artificial working memory unit (WU) as used in machine learning, which relies on multiplicative gating (illustrated with element-wise multiplication – filled circle). (**b**) Working memory cortical unit (WCU), which relies on subtractive gating (illustrated with a minus). In both models input arrives at pyramidal cells (PC), which we propose to be located in layer-4 or layer 2/3 in the cortex. This neuron then projects to a memory cell in layer-5 of the cortex, implemented by a population of recurrently connected pyramidal cells recurrent neural network. This in turn sends output to other units or to the output layer. Importantly, the flow of information is controlled by a gating unit or interneuron (IN; red), which can be multiplicative (a) or subtractive (b). See Costa et al. (2017) for a more detailed description of these models.

Single item working memory task

We started out by testing both models with a simple working memory task, which is based on recent orientation working memory encoding tasks (Manohar et al. (2019); Fallon et al. (2018)). The network receives both a cue signal $\{0, 1\}$ and an orientation encoded in the following range $[0..1]$. The task of the network is to remember the cued orientation and ignore distractors until it is asked to recall the original orientation (Figure 2a).

For this task all models were trained on sequences of 20 time steps. Each model was comprised of 20 hidden units receiving two-element vectors containing the orientation and a mask representing the cue. The models were all implemented in Python using PyTorch and trained for 1000 epochs using the ADAM (Kingma and Ba (2014)) optimizer with a learning rate of 0.01. In each epoch 100 instances were sampled at random from the given range and fed into the model in batches of 10. The hidden units project into a single output unit and the error is measured using the mean squared error between this output and the target. Each model was randomly initialized 3 times with different seeds.

During training, we found that WMN converged quickly to a solution, while subWMN required exposure to more training examples to achieve similar results (Figure 2b). Interestingly, this is in contrast with our results in more complex problems (see next sections). However, both models eventually converged to similar solutions.

We hypothesize that subtractive gating has a regularizing effect on the error landscape, enabling the networks to find better solutions. To test this we varied the testing conditions independently along two parameter: the number of distractors and their magnitude. The results shown in Figures 2c and 2d demonstrate that indeed, subtractive gating allowed the networks to generalize better. However, the performance in both networks decreases substantially. When increasing the number of distractors, subWMNs exhibited asymptotic behaviour in their errors as the number of distractors was increased. Whereas when increasing the magnitude of the distractors the performance in both models decreased substantially, but less so in subWMNs. This is expected given that we are going of the range of distractors with which the network was trained. Not only that, but because of the loss function used, incorrect recall of cued stimuli will incur in larger errors.

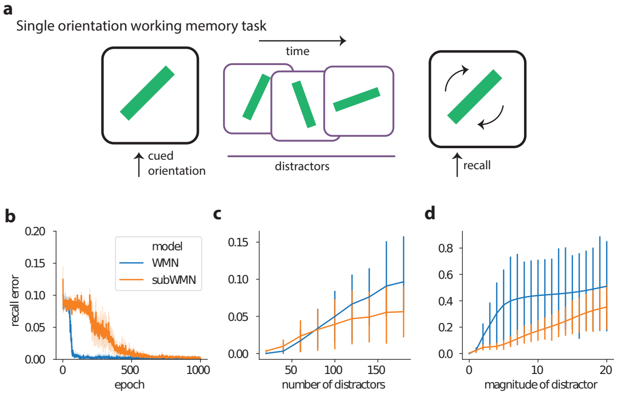

Figure 2: **Subtractive gating generalizes better to distractors in working memory task.** (**a**) Schematic of the task. The network needs to learn to recall one orientation, while ignoring a number of distractors. Note that the network is trained and tested with a large number of orientations. (**b**) Learning curve for both models. (**c**) Memory maintanence test with an increasing number of distractors for both models. (**d**) Resistance to distractor strength for both models.

Adding working memory task

This is a more challenging task in which we studied working memory update (related to Fallon et al. (2018)). Compared to the previous task the network is cued a second time and its task is to update the old orientation by adding it to the new one (i.e. if the old orientation was 0.3, and the new 0.5, the final orientation should be 0.8). As before the network needs to also ignore distractors. For this task, the networks were trained on sequences of length 50 and used 50 hidden units. The starting learning rate was also lowered to 0.0001 and decayed by a factor of 0.1 after 10 epochs of no improvements until reaching a minimum of 1e-8. As in the previous task, we use 3 random initializations for each model and the data generator.

While in the previous task training was faster with WMN, this is no longer the case in this harder setting. As we can see in results shown in Figure 3b, both networks converge at similar rates, but now a characteristic irregular behaviour can be observed in the learning curve of WMN, briefly having much worse performance before improving again until converging to a solution.

As in the previous task, generalization performance in Figure 3c and 3d under similar variations show that subWMN not only incurred in lower error, but also less variance in their performance, which suggests that the solutions found by this class of models is more robust. Interestingly, the difference between the two models in this, harder task, is more evident than in the previous task.

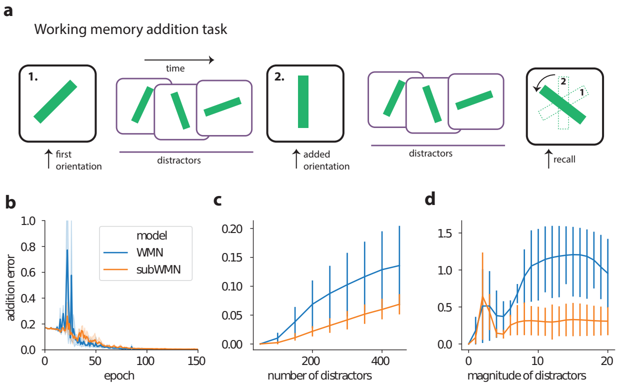

Figure 3: **Subtractive gating generalizes better to distractors in working memory task with addition.** (**a**) Schematic of the task. Two orientations (1 and 2) are cued and should be added for recall, while ignoring a number of distractors. (**b**) Learning curve for both models. (**c**) Memory maintanence test with an increasing number of distractors for both models. (**d**) Resistance to distractor strenght for both models.

Temporal handwritten digit working memory task

Most real life working memory tasks that we are faced with require us to process complex temporal inputs. We tested the networks with a problem – recognizing handwritten digits – that is derived from the machine learning community and is a more challenging test of the memory properties of these networks. We trained the networks to solve a version of this MNIST dataset where the pixels are presented as a single sequences of 784 steps (i.e. showing only one pixel at a time). While this may not be a natural stimuli, it still serves as a good test of the ability of the models to integrate complex evidence along large time spans.

For this task, the capacity of the hidden layer was increased to 100, and instead of projecting its hidden state into to a single output unit, they where linearly projected into 10 softmax units in order to compute a probability for each of the 10 digits. We used a standard cross entropy loss function between the correct label and the output probability given by the network.

As observed in Figure 4 the irregular behaviour of the validation curve in WMN is even more pronounced. The same learning curve for subWMN, is not only smoother but tends to converge to lower values of the cross entropy score. This has important consequences for the accuracy of the models, but also for their confidence: models with lower cross entropy are more confidence in their decisions, even if they achieve the same classification accuracy.

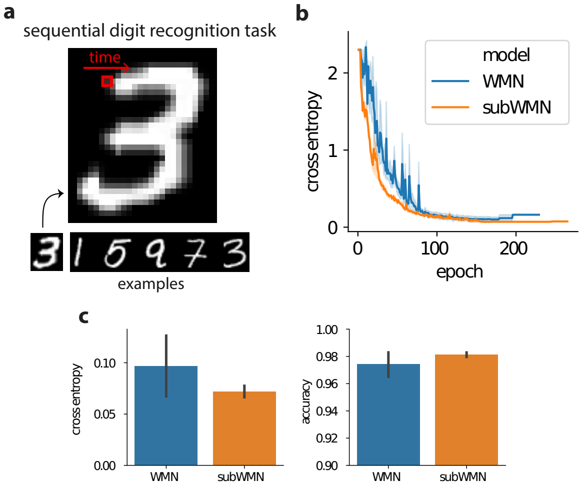

Figure 4: **Temporal handwritten digit working memory task.** (**a**) This task tested the memory properties of the network, as they are trained to recognise a sequence of pixels (i.e. they only receive as input one pixel at the time as represented by the small red box). The network is trained to recognise 10 different digits (some examples are given in bottom inset). (**b**) Learning curve (based on cross entropy) of the two models over. (**c**) Final cross entropy (left) and accuracy (right). Both models perform well, but subWMNs have lower variance, suggesting that they are a more robust model.

Discussion

Here we have provided some initial evidence suggesting that subtractive gated RNNs have better generalisation properties in working memory tasks. We should highlight that the generalization tests done were never seen during training. Importantly, these results appear to also apply to more challenging tasks where the network needs to remember relatively complex temporal sequences.

It is unclear why subtractive gating should lead to better generalisation than multiplicative gating. However, we hypothesize that subtractive gating has a regularizing effect, enabling the networks to find more robust solutions. In future work we aim to study both the cause of this effect and its robustness.

In addition, while the networks studied here show some form of improved generalisation, they are still far from what humans can do. It would be interesting to investigate what other biological constraints may further improve generalisation in these tasks. On the other hand, humans and other animals do not always behave flawlessly. Therefore, it would be important to relate our findings to similar experimental observations from the working memory literature (Manohar et al. (2019)).

As next steps we plan to study the representation learned by the two models. Given that such a relatively simple change in gating mode has an effect on the generalisation properties of the network we also expect this to be reflected in the representation profile of the network.

Overall, our results suggest that recurrent networks with subtractive gating provide better models of working memory.

References

Costa, R. P., Assael, I. A., Shillingford, B., de Freitas, N., & Vogels, T. (2017). Cortical microcircuits as gated-recurrent neural networks. , 272–283.

Doiron, B., Longtin, A., Berman, N., & Maler, L. (2001, January). Subtractive and divisive inhibition: effect of voltage-dependent inhibitory conductances and noise. *Neural Computation*, *13*(1), 227–248.

El Boustani, S., & Sur, M. (2014). Response-dependent dynamics of cell-specific inhibition in cortical networks in vivo. *Nature Communications*, *5*, 5689.

Fallon, S. J., Mattiesing, R. M., Dolfen, N., Manohar, S. G., & Husain, M. (2018, October). Ignoring versus updating in working memory reveal differential roles of attention and feature binding. *Cortex*, *107*, 50–63.

Hassabis, D., Kumaran, D., Summerfield, C., & Botvinick, M. (2017, July). Neuroscience-Inspired Artificial Intelligence. *Neuron*, *95*(2), 245–258.

Kingma, D., & Ba, J. (2014, December). Adam: A Method for Stochastic Optimization.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, *40*.

Manohar, S. G., Zokaei, N., Fallon, S. J., Vogels, T. P., & Husain, M. (2019, June). Neural mechanisms of attending to items in working memory. *Neuroscience & Biobehavioral Reviews*, *101*, 1–12.

Mejias, J. F., Kappen, H. J., Longtin, A., & Torres, J. J. (2013). Short-term synaptic plasticity and heterogeneity in neural systems. , *1510*, 185.

Seybold, B. A., Phillips, E. A. K., Schreiner, C. E., & Hasenstaub, A. R. (2015, September). Inhibitory Actions Unified by Network Integration. *Neuron*, *87*(6), 1181–1192.