Recurrent Neural Network with Adaptive Gating Timescales Mechanisms for Language and Action Learning

Libo Zhao[1](✉) and Junpei Zhong[2]

[1] S.M.Wu School of Intelligent Engineering, South China University of Technology, Guangzhou, China
wilbzhao@mail.scut.edu.cn

[2] Department of Rehabilitation Sciences, The Hong Kong Polytechnic University, Hong Kong, China
joni.zhong@polyu.edu.hk

Abstract. Inspired by the neurons' differences in membrane time-scales, the multiple timescale recurrent neural network model (MTRNN) adopts the hierarchical architecture with increasing time-scales from bottom to top layers. Based on this idea, the recent adaptive and continuous time recurrent neural networks (ACTRNN) and the gated adaptive continuous time recurrent neural network (GACTRNN) develop the novel learning mechanism on the time-scales. In this paper, we test the performance of GACTRNN using the dataset obtained from a real-world humanoid robot's object manipulation experiment. By using trainable timescale parameters with the gating mechanism, it can be observed that the GACTRNN can better learn the temporal characteristics of the sequences. Besides, to eliminate the effects of parameters' overgrowing with a large data-set, we improve the GACTRNN model and propose the MATRNN model. In this model, the *sigmoid* function is used instead of *exponential* function. We compare the performances of the CTRNN, GACTRNN and MATRNN models, and find that the GACTRNN and MATRNN models perform better than the CTRNN model with the large-scale dataset. By visualizing the timescales adapting in the training process, we also qualitatively show that the MATRNN model performs better than the GACTRNN model in terms of stability with the dataset.

Keywords: Recurrent neural network · Adaptive timescale · Membrane time constants · Developmental robotics

1 Introduction

Neurons process information with adaptively converting complex entities (e.g. information comes from motor actions to semantic meanings) into smaller elements in the sequential sequences. Such elements, when learnt in the neural populations, can be further reused to form new complex entities or to understand the sensory information when needed. The idea underlying this process

T. Mantoro et al. (Eds.): ICONIP 2021, CCIS 1517, pp. 405–413, 2021.
https://doi.org/10.1007/978-3-030-92310-5_47

was proposed by Arbib called "schema theory" [1], and based on the above capability, a brain system can divide any complex information entity into primitives and reuse them is broadly defined as functional hierarchy [2,5,6,11]. The functional hierarchy in the large-scale brain networks allows the human brain to cope with sequential processes with different levels of granularity of sensorimotor sequences. However, how this mechanism could be implemented in artificial neural system is not clear. Furthermore, another related challenge is that how can the brain system determine the levels of granularity of such primitives in the sequences, in order to make the encoding in the most efficient way.

The Recurrent Neural Network (RNN) models may be useful to model the recurrent and feedback connections which contribute a lot of the connections in the hierarchical brain [20]. The Long Short-Term Memory (LSTM) [12] and the Gated Recurrent Unit (GRU) [3] are two of the most common RNN variants with gating mechanisms in the state-of-the-art machine learning community. Both of them use particular architectural constraints called gating mechanisms which can alleviate the vanishing gradient problem of original RNNs and help the recurrent networks to learn both the short-term dependencies and long-term dependencies of the data. Nevertheless, with the lack of modeling the critical mechanisms of the neural dynamics in the biological brain, the above RNN variants are not sufficient to build biologically inspired neural systems. To mimic the mechanism of functional hierarchy in neural system at the level of neural dynamics, multiple timescale recurrent neural network (MTRNN) [19] model was proposed. Adopting the concept of time-scale, it can capture the information of sequential data on different timescales by manually-set time-scales. Besides, inspired by the idea that time constants in the brain are subjects to change during development [8], Heinrich et al. proposed the adaptive and continuous time recurrent neural networks (ACTRNN) [10], and its improved version, the gated and adaptive continuous time recurrent neural networks (GACTRNN) [9].

In the original paper of GACTRNN, the proposed model has shown its potential to learn the temporal characteristics on multiple timescales from sequential data, but the paper did not involve an experiment with larger data sets that show distinct and highly complex multi-timescale dependencies such as sensorimotor sequences from robot behavioral data. On the other hand, interaction of a humanoid robot with a physical embodiment is important if we would like to use the neural network to model certain cognitive functions [15,18]. In this paper, we will fill such a gap using a set of multi-modal data containing sensorimotor sequences of a robotic manipulation experiment as training data. We use CTRNN model as the baseline, compare the performance of the CTRNN model, the GACTRNN model and the MATRNN model. The adaptive process of the timescales is also visualized to investigate what the models learn from the sequential data with complex temporal characteristics. By this way, we prove that the MATRNN model performs better than the GACTRNN model in terms of timescale adaptation in more challenging cases.

2 Models

2.1 Continuous Time Recurrent Neural Network Model

Continuous time recurrent neural network (CTRNN) is the main component of the MTRNN model. It is derived from the leaky integrate-and-fire model and thus from a simplification of the Hodgkin-Huxley model. Afterwards, the network architecture was initially developed by Hopfield and Tank [13] as a non-linear graded-response neural network. Later Doya and Yoshizawa [4] developed an adaptive neural oscillator based on previous work. The transformation and activation functions for the CTRNN units are as below:

$$\mathbf{z}_t = \left(1 - \frac{1}{\boldsymbol{\tau_t}}\right)\mathbf{z}_{t-1} + \frac{1}{\boldsymbol{\tau_t}}\left(\mathbf{W}\mathbf{x} + \mathbf{V}\mathbf{y}_{t-1} + \mathbf{b}\right) \tag{1}$$

$$\mathbf{y}_t = f\left(\mathbf{z}_t\right) \tag{2}$$

where $\mathbf{x}$ is the input, $\mathbf{z}_{t-1}$ is the previous internal state, $\mathbf{W}$ and $\mathbf{V}$ are the weights, $\mathbf{b}$ is the bias and $f(\cdot)$ is the activation function. The timescale parameter $\boldsymbol{\tau_t}$ expresses the leakage within a certain time, which is biologically plausible. When the value of $\boldsymbol{\tau_t}$ is large, the activation will change slowly. Conversely the activation will change quickly when the value of $\boldsymbol{\tau_t}$ is small [19].

2.2 Gated Adaptive Continuous Time Recurrent Neural Network Model

Compared with the CTRNN model, in the gated adaptive continuous time recurrent neural network (GACTRNN), the timescales are no longer hyperparameters that are determined by means of empiricism. Instead they are trainable parameter determined by the learning mechanism. The adaptive and gating timescales mechanisms of the GACTRNN model are implemented by the computation of the timescale parameter as follows:

$$\boldsymbol{\tau}_t = 1 + \exp\left(\mathbf{H}\mathbf{x} + \mathbf{G}\mathbf{y}_{t-1} + \mathbf{a} + \boldsymbol{\tau}_0\right) \tag{3}$$

where $\mathbf{x}$ and $\mathbf{y}$ are defined the same as the CTRNN, $\boldsymbol{\tau}_t$ is the timescale parameter at time t, $\mathbf{H}$ and $\mathbf{G}$ are the weights which simulates the gating on the neuron's leakage characteristic, $\mathbf{a}$ is the bias, $\boldsymbol{\tau}_0$ is a vector that can be defined as initial values of the timescales.

2.3 Multiple Adaptive Timescale Recurrent Neural Network Model

It can be noticed that the value range of the timescale parameter in the GACTRNN model is $[1, +\infty]$. The infinity timescale is not we expect in most cases. Therefore, in this paper, we propose a novel model called the multiple adaptive timescale recurrent neuron network (MATRNN), to eliminate this problem by changing the activation function so that the timescale parameter can hierarchically adapt and will not obtain any unexpected values during the training process. The computation for the timescale parameter of the MATRNN model is as follows:

$$\boldsymbol{\tau}_t = 1 + \boldsymbol{\tau}_0 \otimes sigmoid\left(\mathbf{Hx} + \mathbf{Gy}_{t-1} + \mathbf{a}\right) \tag{4}$$

Where $\otimes$ indicates the element-wise multiplication and the other parameters are similar to the GACTRNN model.

3 Experiment

3.1 Experiment Dataset

The dataset we use is from the object manipulation experiment on the iCub robot [14], recorded by Zhong et al. [21]. The iCub robot is a child sized humanoid robot, built as an open humanoid platform for cognitive and neuroscience research [17]. This training data includes two parts of information: semantic commands and sensorimotor information. The semantic command part is composed of two discrete words: 9 actions and 9 objects are used to form 81 possible combinations, which are represented by discrete values, as shown in Table 1. The sensorimotor information part of training data, which includes object location (neck and eyes) and torso joints of the iCub robot, is the recording of a complete movement sequence of the corresponding semantic command.

Table 1. Look-up table of verbs and nouns for the data sets: the instructor showed the robot with different combinations of the 9 actions and 9 objects

Actions	Slide left	Slide right	Torch	Reach	Push	Pull	Point	Grasp	Lift
Verb values	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
Objects	Tractor	Hammer	Ball	Bus	Modi	Car	Cup	Cubes	Spiky
Noun values	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8

3.2 Case 1: Basic Mode

The purpose of this case is to enable the models to learn specific semantic commands and corresponding action sequences through training process. The input data and ground truth of our experiment are both training data.

To get a fair comparison, the same architecture with the same parameters are used for the three models. We selected the hyperparameters that perform well in the model after a brief hyperparameter search experiment. In our experiments, we defined networks with one hidden layer, which consists of $(43, 30, 20, 10)$ neurons (4 modules) where recurrent connections are only connected to neighboring modules (adjacent connectivity [9]). All hidden neurons were densely (fully) connected to the input and activated with a tanh function. Besides, our experiment has the initial timescales of $(1, 35, 70, 125)$, and a learning rate of 0.001 using Adam optimization algorithm for training over $5,000$ epochs. During training, the loss was calculated by the Mean Squared Error (MSE).

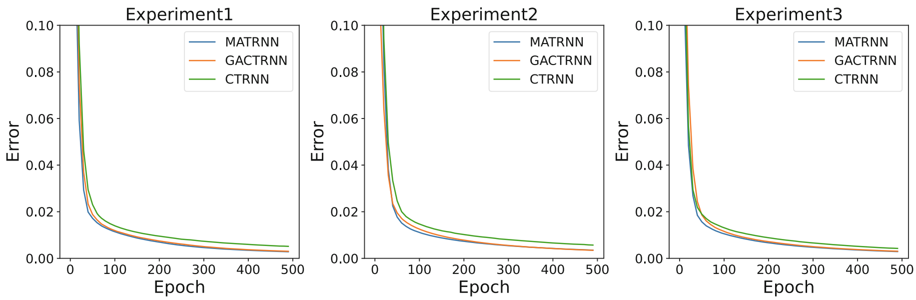

Fig. 1. Training curves of the three models. The three training curves are from three experiments with the same hyperparameters. In order to better show the details, we only select the data of the first 500 epochs for illustration.

Table 2. Performance of the three models

	Experiment1	Experiment2	Experiment3	Average
CTRNN	3.12E−04	3.49E−04	2.72E−0.4	3.11E−04
GACTRNN	1.78E−04	1.81E−04	1.60E−04	1.73E−04
MATRNN	1.76E−04	1.93E−04	1.71E−04	1.80E−04

To eliminate the effect of randomness, three experiments were done with the same parameters. The performance of these three models is shown in Table 2, and the training curves of the models are illustrated in Fig. 1. In the above experiments on the multi-modal data, the GACTRNN and MATRNN model perform a bit better in terms of error, although each model generates a low error. Next, the adaptive process of the timescale parameters is visualized to explore more information during the training process, as shown in Fig. 2 and Fig. 3. We can see that compared with the CTRNN model using pre-determined timescales, the timescale parameters in GACTRNN and MATRNN are constantly changing with the increasing of epoch and finally have a tendency to converge to a certain distribution. So in this basic mode, the adaptation of the timescale parameters in the GACTRNN and MATRNN models performs well, which has the potential to simulate the adaptation process of the timescale in the neuron.

3.3 Case 2: Trajectory Generation Mode

In the basic mode, the mixed signal, which includes the output of the previous step (y_{t-1}) and the original input (x_t), is used for training of the model. While in the trajectory generation mode, we explored the situation where the input of each step only includes the output of the previous step (y_{t-1}). This is a mode of automatically generating trajectories, which is more challenging for the models, in which a neural system may exhibit its instability due to small error. For the generation model, we expect that after training, the models can generate corresponding sensorimotor sequences based on the semantic commands and

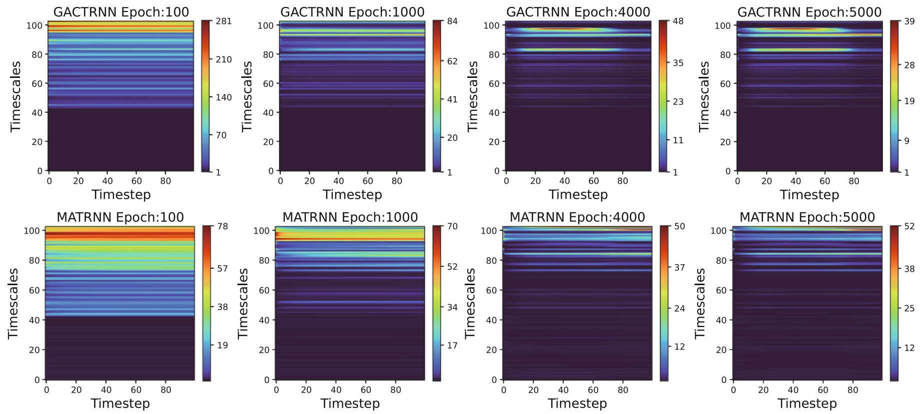

Fig. 2. Distribution diagram of the timescale parameters of a certain action sequence in GACTRNN and MATRNN. The figure shows the timescale parameters distribution of four epochs during a training process of a certain action sequence.

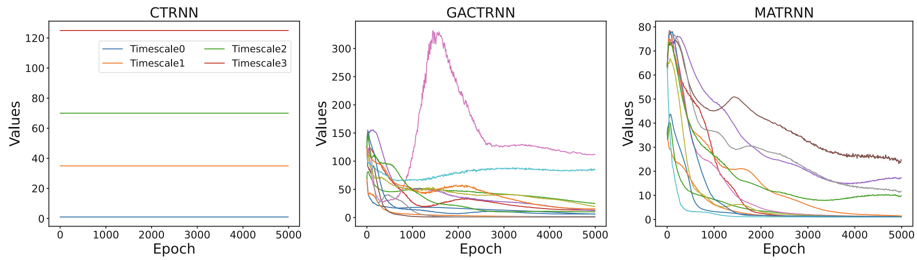

Fig. 3. Adaptive process of part of the timescale parameters of the three models.

information of object location and torso joints at the first timestep and the initial condition.

To realize the generation mode, the computation for activation $\mathbf{y}$ in the new mode is as below:

$$\mathbf{z}_t = \left(1 - \frac{1}{\boldsymbol{\tau}_t}\right)\mathbf{z}_{t-1} + \frac{1}{\boldsymbol{\tau}_t}\left(\mathbf{V}\mathbf{y}_{t-1} + \mathbf{b}\right) \tag{5}$$

$$\mathbf{y}_t = f\left(\mathbf{z}_t\right) \tag{6}$$

$$For\ GACTRNN : \boldsymbol{\tau}_t = 1 + \exp\left(\mathbf{G}\mathbf{y}_{t-1} + \mathbf{a} + \boldsymbol{\tau}_0\right) \tag{7}$$

$$For\ MATRNN : \boldsymbol{\tau}_t = 1 + \boldsymbol{\tau}_0 \otimes sigmoid\left(\mathbf{G}\mathbf{y}_{t-1} + \mathbf{a}\right) \tag{8}$$

In addition, by visualizing the timescale parameters, we investigated the adaptive process of timescales in these two models. The hyperparameters of the model were set to be the same as in the Case 1. The results are shown in Table 3, Fig. 4, and Fig. 5. From the perspective of the error curve shown in Table 3 and Fig. 4, the performances in the training of two models look similar. However, the adaptive process of timescale parameters of these two models is quite different. From Fig. 5 we can see that in the later stage of the training process of this challenging task, only the timescale parameters of a small area in GACTRNN achieve abnormally large values, while the distribution of timescale parameters in MATRNN is more normal and capable of simulating the adaptivity in neuron's timescales. Therefore, in this challenging trajectory generation mode, as a cognitive neural network model, the MATRNN model performs better than the GACTRNN model in terms of timescales' stability.

Table 3. Performance of the two models

	Experiment1	Experiment2	Experiment3	Average
GACTRNN	3.25E−03	4.04E−03	4.05E−03	3.78E−03
MATRNN	4.46E−03	3.52E−03	3.26E−03	3.75E−03

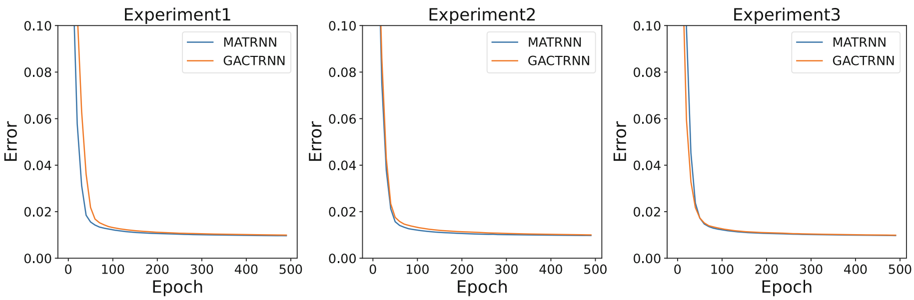

Fig. 4. Training curves of the two models. The three training curves are from three experiments with the same hyperparameters.

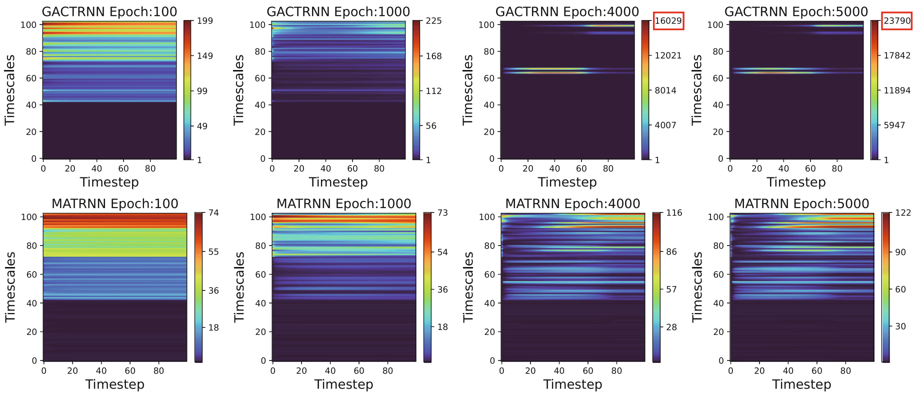

Fig. 5. Adaptive process of the timescale parameters in GACTRNN and MATRNN. As shown in the red-boxes, timescale parameters of a small area in GACTRNN achieve abnormally large values. (Color figure online)

4 Conclusion and Future Work

In this study, using the large-scale sensorimotor dataset based on a humanoid robot, we compared the performance of CTRNN, GACTRNN, and MATRNN, and visualized adaptive process of the timescale parameters. Through the experiment, the following conclusions can be drawn:

1. The GACTRNN and MATRNN models perform a little better than the CTRNN model in our basic mode experiment.
2. During the training process, the timescale parameters in GACTRNN and MATRNN are constantly adapting and have a tendency to converge to a certain distribution which can simulate the adaptivity in neuron's timescales.
3. In the challenging trajectory generation mode, as a cognitive neural network model, the MATRNN model performs better than the GACTRNN model in terms of timescale adaptation.

Neurons' timescales seem to be quite dynamic in prefrontal cortex while neurons in other regions seem quite stable over a short period of time [7,16]. Using the MATRNN model or GACTRNN model, we have interest to apply them in computational psychiatry where the neural dysfunctional mechanisms related to abnormal timescales can be simulated. The artificial model can also make the computational psychiatry embodiment be feasible.

Acknowledgments. This work is partially supported by PolyU Start-up Grant (ZVUY-P0035417).

References

1. Arbib, M.A., et al.: Neural Organization: Structure, Function, and Dynamics. MIT Press, Cambridge (1998)
2. Boemio, A., Fromm, S., Braun, A., Poeppel, D.: Hierarchical and asymmetric temporal sensitivity in human auditory cortices. Nat. Neurosci. **8**(3), 389–395 (2005)
3. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
4. Doya, K., Yoshizawa, S.: Adaptive neural oscillator using continuous-time backpropagation learning. Neural Netw. **2**(5), 375–385 (1989)
5. Felleman, D.J., Van Essen, D.C.: Distributed hierarchical processing in the primate cerebral cortex. Cereb. Cortex (New York, NY: 1991) **1**(1), 1–47 (1991)
6. Fuster, J.M.: The prefrontal cortex-an update: time is of the essence. Neuron **30**(2), 319–333 (2001)
7. Gao, R., van den Brink, R.L., Pfeffer, T., Voytek, B.: Neuronal timescales are functionally dynamic and shaped by cortical microarchitecture. Elife **9**, e61277 (2020)
8. He, B.J.: Scale-free brain activity: past, present, and future. Trends Cogn. Sci. **18**(9), 480–487 (2014)
9. Heinrich, S., Alpay, T., Nagai, Y.: Learning timescales in gated and adaptive continuous time recurrent neural networks. In: 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 2662–2667. IEEE (2020)

10. Heinrich, S., Alpay, T., Wermter, S.: Adaptive and variational continuous time recurrent neural networks. In: 2018 Joint IEEE 8th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), pp. 13–18. IEEE (2018)
11. Hilgetag, C.C., O'Neill, M.A., Young, M.P.: Hierarchical organization of macaque and cat cortical sensory systems explored with a novel network processor. Philos. Trans. R. Soc. Lond. Ser. B: Biol. Sci. **355**(1393), 71–89 (2000)
12. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
13. Hopfield, J.J., Tank, D.W.: Computing with neural circuits: a model. Science **233**(4764), 625–633 (1986)
14. Metta, G., Sandini, G., Vernon, D., Natale, L., Nori, F.: The iCub humanoid robot: an open platform for research in embodied cognition. In: Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems, pp. 50–56. Association for Computing Machinery (2008)
15. Pfeifer, R., Lungarella, M., Iida, F.: Self-organization, embodiment, and biologically inspired robotics. Science **318**, 1088–1093 (2007)
16. Spitmaan, M., Seo, H., Lee, D., Soltani, A.: Multiple timescales of neural dynamics and integration of task-relevant signals across cortex. Proc. Natl. Acad. Sci. **117**(36), 22522–22531 (2020)
17. Tsagarakis, N.G., et al.: iCub: the design and realization of an open humanoid platform for cognitive and neuroscience research. Adv. Robot. **21**(10), 1151–1175 (2007)
18. Varela, F.J., Thompson, E., Rosch, E.: The Embodied Mind, Revised Edition: Cognitive Science and Human Experience. MIT Press, Cambridge (2017)
19. Yamashita, Y., Tani, J.: Emergence of functional hierarchy in a multiple timescale neural network model: a humanoid robot experiment. PLoS Comput. Biol. **4**(11), e1000220 (2008)
20. Zhong, J.: Artificial neural models for feedback pathways for sensorimotor integration. Ph.D. thesis, Staats-und Universitätsbibliothek Hamburg Carl von Ossietzky (2015)
21. Zhong, J., Peniak, M., Tani, J., Ogata, T., Cangelosi, A.: Sensorimotor input as a language generalisation tool: a neurorobotics model for generation and generalisation of noun-verb combinations with sensorimotor inputs. Auton. Robot. **43**(5), 1271–1290 (2019)