Gated Orthogonal Recurrent Units: On Learning to Forget

Li Jing, Caglar Gulcehre*, John Peurifoy, Yichen Shen, Max Tegmark, Marin Soljačić, Yoshua Bengio*

* equal contribution

Massachusetts Institute of Technology

MILA

Contribution

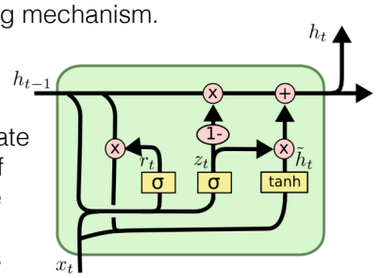
We present a new recurrent neural network architecture called Gated Orthogonal Recurrent Unit (GORU) that combines Gated mechanism and orthogonal approach, and is able to forget and learn long-term dependency at the same time in sequential tasks.

Background

Gating Mechanism: Forgetting

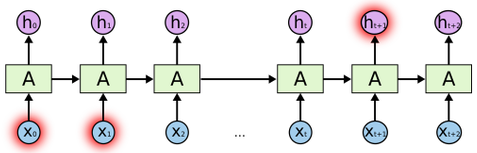
LSTM and GRU were proposed to learn long/short term dependency in sequential data. The short term part is successfully solved by gating mechanism.

Forgetting ability becomes particularly critical when the dimensionality of the RNN state is smaller than the product of the sequence length with the input dimension; i.e., when some form of compression is necessary.



Long term dependency: Orthogonal Matrices

However, long-term problem still prevents LSTM/GRU to successfully solve real world tasks mainly because of gradient vanishing and explosion problem.



$$\frac{\partial C}{\partial \mathbf{h}^{(t)}} = \frac{\partial C}{\partial \mathbf{h}^{(T)}}\frac{\partial \mathbf{h}^{(T)}}{\partial \mathbf{h}^{(t)}} = \frac{\partial C}{\partial \mathbf{h}^{(T)}}\prod_{k=t}^{T-1}\frac{\partial \mathbf{h}^{(k+1)}}{\partial \mathbf{h}^{(k)}} = \frac{\partial C}{\partial \mathbf{h}^{(T)}}\prod_{k=t}^{T-1}\mathbf{D}^{(k)}\mathbf{W},$$

By constraining evolutionary matrix to be unitary/orthogonal. [1,2] have successfully solved gradient vanishing/explosion problem.

Reference & Code

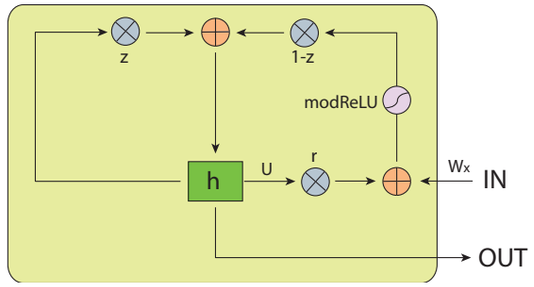
[1] M. Arjovsky, A. Shah, and Y. Bengio, "Unitary Evolution Recurrent Neural Networks," ICML 2016

[2] L. Jing, Y. Shen et al. "Tunable efficient unitary neural networks (EUNN) and their application to RNNS." ICML 2017

https://github.com/jingli9111/GORU-tensorflow

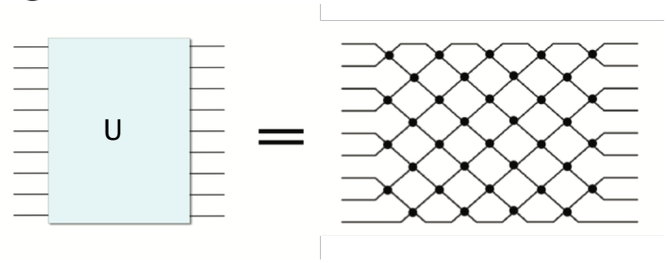
Model

Gated Orthogonal Recurrent Unit Architecture



$$\begin{aligned}
\mathbf{h}_t &= \mathbf{z}_t \odot \mathbf{h}_{t-1} + (1-\mathbf{z}_t)\odot \mathrm{modReLU}(\mathbf{W}_x\mathbf{x}_t + \\
&\quad \mathbf{r}_t \odot (\mathbf{U}\mathbf{h}_{t-1}) + \mathbf{b}_h), \\
\mathbf{z}_t &= \mathrm{sigmoid}(\mathbf{W}_z\mathbf{h}_{t-1} + \mathbf{W}_{z,x}\mathbf{x}_t + \mathbf{b}_z), \\
\mathbf{r}_t &= \mathrm{sigmoid}(\mathbf{W}_r\mathbf{h}_{t-1} + \mathbf{W}_{r,x}\mathbf{x}_t + \mathbf{b}_r),
\end{aligned}$$

Orthogonal Matrix Parametrization



We follow orthogonal matrix parametrization method described in [2]. In this parametrization method, orthogonal matrices are constructed by sequences of 2-by-2 rotational matrices.

Compared to Gated Recurrent Unit (GRU),

1. the matrix before the reset gate is restrained to be orthogonal.
2. activation is changed to **modReLU**

$$\mathrm{modReLU}(z_i, b_i) \equiv \frac{z_i}{|z_i|}\mathrm{ReLU}(|z_i| + b_i)$$

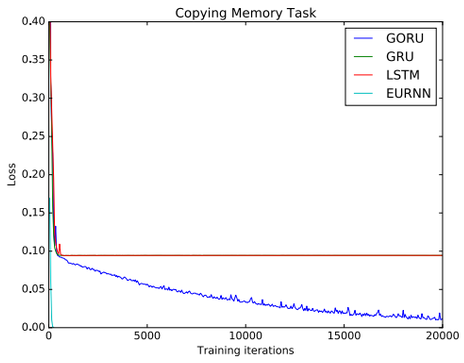
Experiment

We compare our model to LSTM/GRU basic unitary RNN in varieties of tasks.

Copying Memory Task

Long-term dependency

The copying task is a synthetic task that is commonly used to test the network's ability to remember information seen T time steps earlier.



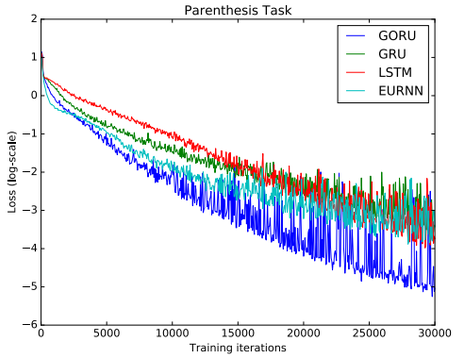
GORU solves copying memory problem which LSTM/GRU get stack to baseline.

Parenthesis Task

Forgetting

The parenthesis task(Foerster et al. 2016) requires the RNN model to count the number of each type of unmatched parentheses at each time step, given that there are 10 types of parentheses.

GORU significantly outperforms LSTM, GRU and basic Unitary RNN.



Other Tasks

We did other tasks including:

1. Denoise Task;
2. Algorithmic Learning;
3. Question answering;
4. Character-level Language Modeling
5. Speech Prediction task.

Each task is either able to test *forgetting* ability or to test *long-term dependency.*

Only Gated Orthogonal Recurrent Unit has the ability to forget and learn long-term dependency at the same time. As a result, it is the only model succeed in all these tasks.

AAAI workshop @ New Orlean, February 2 2018