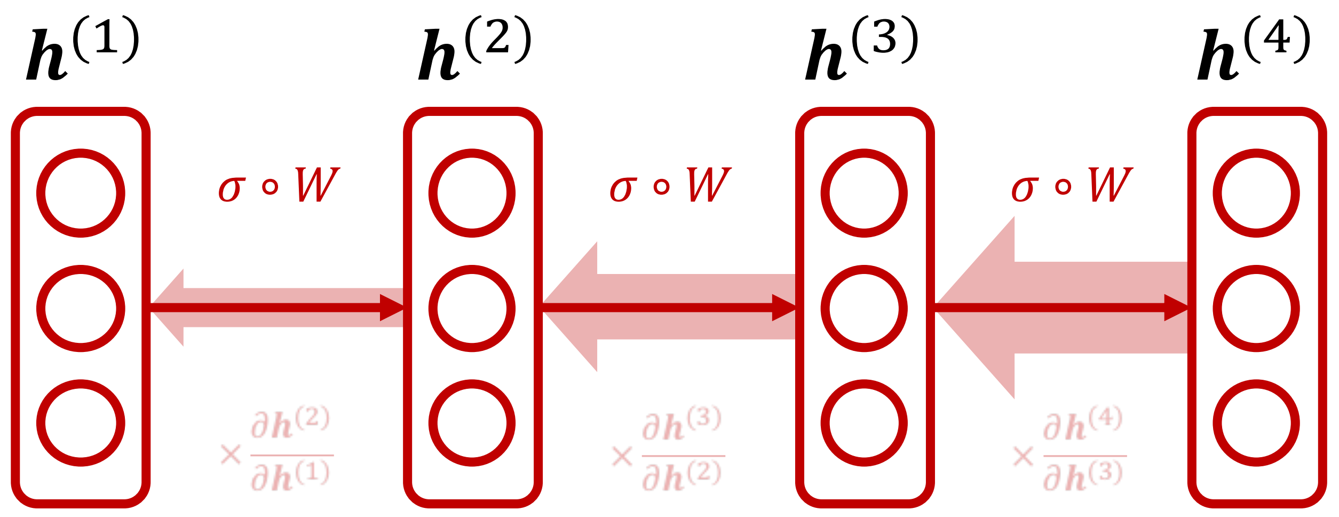
uRNN: An Approach to Bounded Gradients

David Kewei Lin
linkewei@stanford.edu

Jensen Jinhui Wang
wangjh97@stanford.edu

CS 224N (Win 2019) Final Project
Mentor: Michael Hahn

Problem


$h^{(1)}$
$h^{(2)}$
$h^{(3)}$
$h^{(4)}$
$\sigma \circ W$
$\sigma \circ W$
$\sigma \circ W$
$\times \frac{\partial h^{(2)}}{\partial h^{(1)}}$
$\times \frac{\partial h^{(3)}}{\partial h^{(2)}}$
$\times \frac{\partial h^{(4)}}{\partial h^{(3)}}$

Vanilla RNNs tend to face **the problem of vanishing or exploding gradients**: iterative applications of the weight matrix and the activation function cause the gradient to expand or shrink exponentially in the number of time steps:

$$\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{h}_k} = \prod_{t \geq i > k} W^T \operatorname{diag}(\sigma'(\boldsymbol{h}_{i-1}))$$

Sensitive to eigenvalues [1]

Typically ≤ 1

Bounds

Main Theorem. If W is a $(h \times h)$ diagonalizable matrix whose largest and smallest eigenvalues are $\overline{\lambda}$ and $\underline{\lambda}$ respectively, and suppose that σ' is bounded between $[\beta, \gamma]$, then

$$(\underline{\lambda}\beta)^{t-k} \|I_h\| \leq \left\| \frac{\partial h_t}{\partial h_k} \right\| \leq (\overline{\lambda}\gamma)^{t-k} \|I_h\|$$

where I_h is the identity matrix.

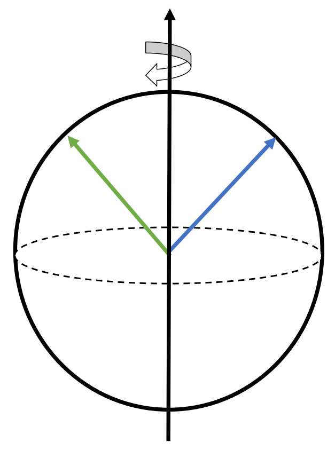
	$\underline{\lambda}\beta < 1$	$\underline{\lambda}\beta = 1$	$\underline{\lambda}\beta > 1$
$\overline{\lambda}\gamma < 1$	**Vanishes**	-	-
$\overline{\lambda}\gamma = 1$	May vanish	**Constant**	-
$\overline{\lambda}\gamma > 1$	May vanish or explode	May explode	**Explodes**

The choice of $\beta, \gamma, \overline{\lambda}, \underline{\lambda}$ determines the behavior of gradients across timesteps.

Ideally, we may select $\sigma(x) = |x|$ and W to be an orthogonal matrix, then the gradients are forced to have constant norm.

Approach

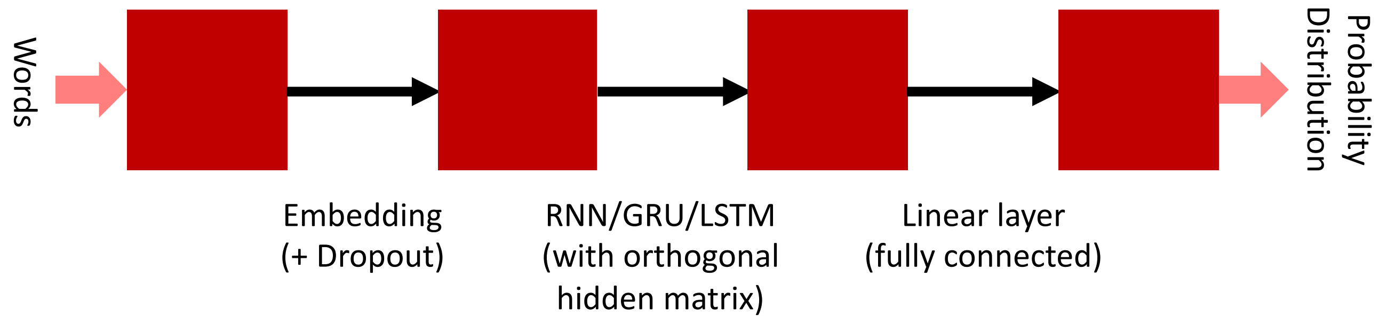
The idea of constraining eigenvalues suggests thinking about W as a ***rotation*** in space. [2] terms this a ***unitary RNN (uRNN)*** as the weight W is unitary (or in the case with real entries, orthogonal).



W is kept orthogonal via a modified update rule (on the right) based on [3] and [4].

$$W \leftarrow \exp(-\alpha g) W$$
$$g = \nabla_W J \, W^T - W (\nabla_W J)^T$$

Overall architecture:



Toy Experiments

Task 1 – First term recall

5, 7, 9, 53, 99, ...

(Ans: 5)

Acc. (%) \ Input length	20	50	100
RNN	10.22	10.56	0.936
uRNN	99.98	100.00	100.00

Task 2 – k-th highest term

5, 7, 9, 53, 99, 98, 2, 100, ...

(Ans: 98)

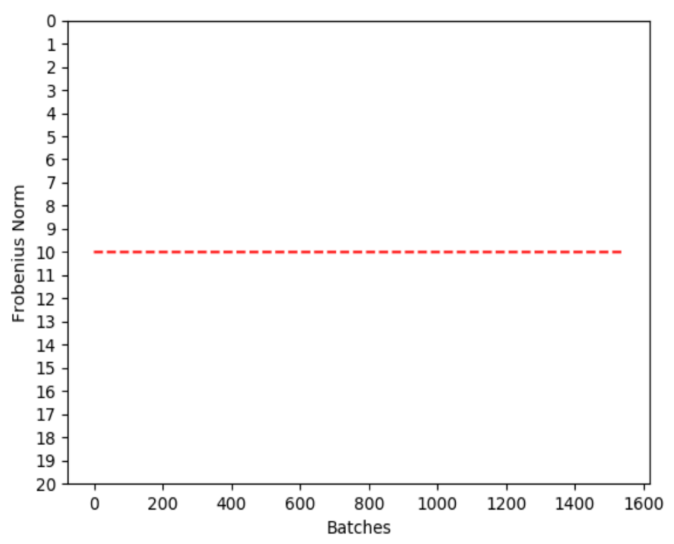
Acc. (%) \ Input length	100	200	400
RNN	38.12	34.15	15.23
uRNN	**92.01**	**80.45**	**56.73**

The uRNN outperforms the vanilla RNN on both toy experiments, especially on longer sequence lengths. With more timesteps, the gradients on the RNN should start to vanish, making it harder to train.

Language Modelling

The task of language modelling involves predicting the next word given a sequence of previous words.

- Conducted on Penn Treebank Dataset.
- Successfully bounded gradient norms.
- Example sentences suggest "long-term" memory.
- 2ms / 1 – 10% slower per batch despite "costly" matrix exponential operation



"this talk would the most important guidelines by now for our benefit capital which makes about N N of the key machines related a rate for customer democrats morning while the number of shared ones also with best united states now be counted following midnight night."

Ppl \ Activation	ReLU	|x|	Leaky ReLU
RNN	754.65 / 705.28	784.53 / 773.91	**756.48 / 712.88**
uRNN	**720.88 / 666.93**	**717.12 / 613.89**	869.21 / 810.21

Ppl \ BPTT	35	150
GRU	**100.80 / 97.25**	108.33 / 105.34
uGRU	101.76/ 98.05	**105.05 / 102.07**

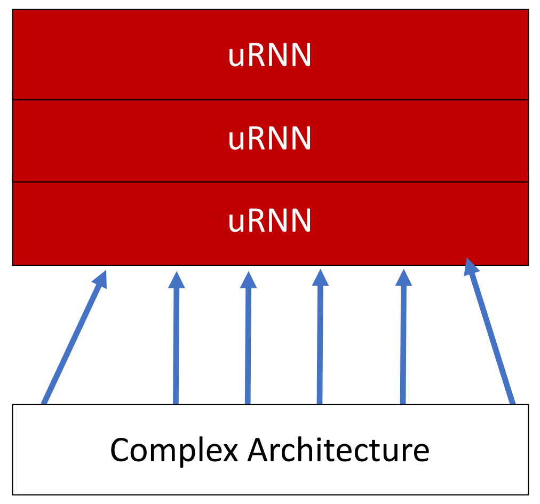
Discussion

Advantages

- Great at contextual tasks which require "long-term" memory
- Concrete bound on gradients
- Potentially better at training networks before the recurrent layer

Disadvantages

- Negligibly slower (especially for larger batch sizes)



Possible architecture for future work

References:
[1] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. Understanding the exploding gradient problem. CoRR, abs/1211.5063, 2012. URL http://arxiv.org/abs/1211.5063.
[2] Martín Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. CoRR, abs/1511.06464, 2015. URL http://arxiv.org/abs/1511.06464.
[3] Scott Wisdom, Thomas Powers, John R. Hershey, Jonathan Le Roux, and Les E. Atlas. Full-capacity unitary recurrent neural networks. CoRR , abs/1611.00035, 2016. URL http://arxiv.org/abs/1611.00035.
[4] Mark D. Plumbley. Lie group methods for optimization with orthogonality constraints. Independent Component Analysis and Blind Signal Separation Lecture Notes in Computer Science, page 1245–1252, 2004. doi: 10.1007/978-3-540-30110-3_157.

Acknowledgement:
We would like to thank Michael Hahn for his guidance, advice and various suggestions throughout the project.