Hierarchical Multiscale Recurrent Neural Networks

Junyoung Chung, Sungjin Ahn, Yoshua Bengio
Presented by Arvid Frydenlund

February 23, 2018

The problem with stacked RNNs

- **Issue:** Temporal data often has structure at different time scales

  i.e. characters → words → phrases → sentences

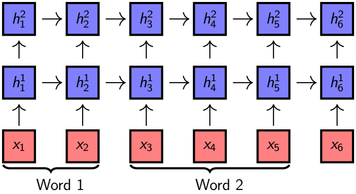
- Want an RNN to make efficient use of that hierarchical structure
- Stacked RNNs need to
  1. work on lowest common scale

     i.e. needs to step all layers every character
  2. work on set cyclic scales

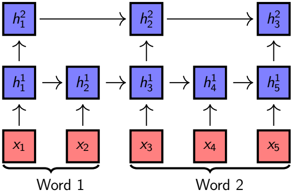
     i.e. a clock work RNN stepping layers according to hyperparameters
  3. be given boundary information about the time scales as input

     i.e. knowing that words are separated by white space
- **Solution:** Make dynamic decisions on when to step layers

Stacked and Clockwork RNNs

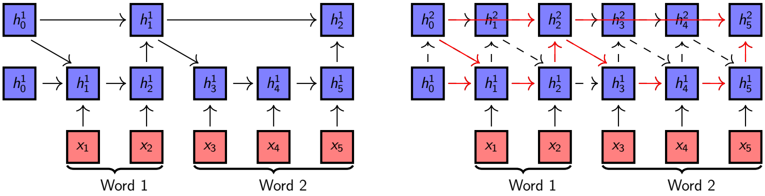

h_1^2 h_2^2 h_3^2 h_4^2 h_5^2 h_6^2
h_1^1 h_2^1 h_3^1 h_4^1 h_5^1 h_6^1
x_1 x_2 x_3 x_4 x_5 x_6
Word 1
Word 2

(a) Stacked RNN


h_1^2 h_2^2 h_3^2
h_1^1 h_2^1 h_3^1 h_4^1 h_5^1
x_1 x_2 x_3 x_4 x_5
Word 1
Word 2

(b) Clockwork RNN

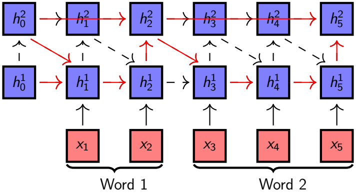
Boundary-aware and Hierarchical Multiscale RNNs



(a) Boundary-aware RNN

(b) Hierarchical Multiscale RNN

Hierarchical Multiscale RNNs



- Boundary detection at every layer
  - $z_t^l = \max(0, \min(1, \frac{ax+1}{2})) > 0.5$, a is a slope hyper-parameter
- Operations (Simplified)
  1. Update: Takes in new input and updates hidden state, if boundary is detected
  2. Copy: Carries over whole hidden state without change, if no boundary is detected
  3. Flush: Pushes current hidden state to next layer then does a (hard) reset of the hidden state, if boundary is detected

Pros and Cons of Hierarchical Multiscale RNNs

- **Pros:**
  1. Computational Efficiency since upper layers require less updates
  2. Less updates means better information transfer across network and less vanishing gradients
  3. Better resource allocation since we can make upper layers higher dimensional
  4. Possibly using learned hierarchal information for down stream tasks
- **Con:** Discrete choices means that the network is no longer differentiable
  - Use 'Straight-through' estimator
  - Use thresholded hard sigmoid during forward pass and ignore threshold during backward pass
  - Anneal slope, a, to train from a softer function to a sharper one

Language Modelling Experiments

PTB	**LayerNorm HyperNetworks**	1.23
	HM-LSTM (No annealing)	1.25
	HM-LSTM (annealing)	1.24
Hutter	**decomp8**	1.28
	HM-LSTM (annealing)	1.32
Text8	**HM-LSTM (annealing)**	1.29
	BatchNorm-LSTM	1.36

Table: Bits-per-character for character-level language modelling. HM-LSTM is our model. Then SOTA bolded.

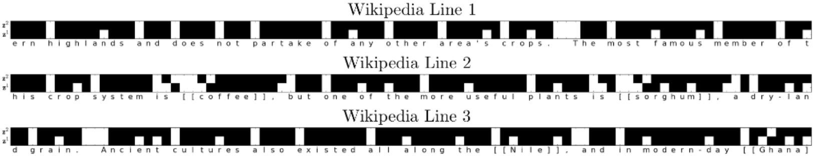


Figure: Detected boundaries in white.

Thanks