Deep Learning course

Session 12 – Recurrent Neural Networks (GRU & LSTM)

E. Francisco Roman-Rangel
edgar.roman@alumni.epfl.ch

CInC-UAEM. Cuernavaca, Mexico. October 5$^{\text{th}}$, 2018.

Outline

Vanilla RNNs

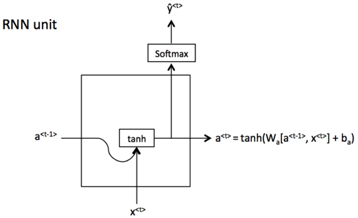
- Sequential data.
- State-of-the-art.
- Limitation: Vanishing/Exploiting gradient.
- Alternative: GRU and LSTM.

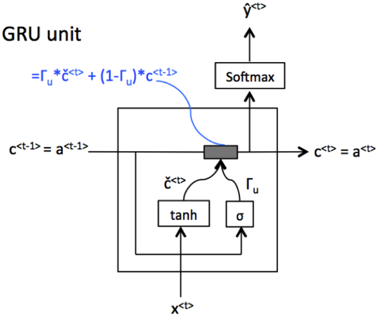
Outline

Gated Recurrent Unit (GRU)

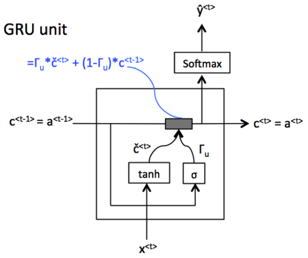
[Cho et al., 2014. "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation"].

An RNN with internal gates for memory-like functionality.





Gated Recurrent Unit (GRU)



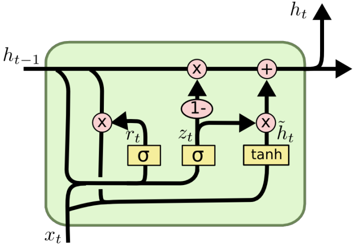
- $\mathbf{C}_t = \mathbf{a}_t$: memory cell.
- Γ: gate.

- $\tilde{\mathbf{C}}_t = \tanh(\mathbf{w}^{(C)}[\mathbf{C}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(C)})$
- $\Gamma_t = \sigma(\mathbf{w}^{(u)}[\mathbf{C}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(u)})$
- $\mathbf{C}_t = \Gamma_t \times \tilde{\mathbf{C}}_t + (1 - \Gamma_t) \times \mathbf{C}_{t-1}$

Γ is a sigmoid: probability of updating (or keeping memory).
Γ is of the same dimension as $\mathbf{x}$.
Element-wise multiplication.

Full GRU

Weight relevance of previous memory.



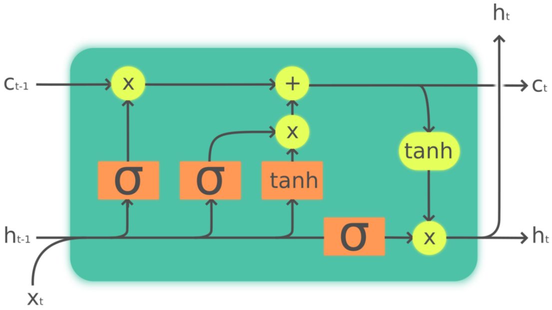
- $\tilde{\mathbf{C}}_t = \tanh(\mathbf{w}^{(C)}[\Gamma_t^{(r)} \times \mathbf{C}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(C)})$
- $\Gamma_t^{(r)} = \sigma(\mathbf{w}^{(r)}[\mathbf{C}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(r)})$
- $\Gamma_t^{(u)} = \sigma(\mathbf{w}^{(u)}[\mathbf{C}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(u)})$
- $\mathbf{C}_t = \Gamma_t^{(u)} \times \tilde{\mathbf{C}}_t + (1 - \Gamma_t^{(u)}) \times \mathbf{C}_{t-1}$

Outline

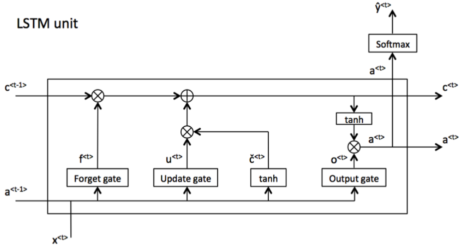
Long Short-Term Memory (LSTM)

[Hochreiter & Schmidhuber, 1997. “Long short-term memory”].

A type of RNN even more powerful than the GRU.
Contains: memory cell, input gate, output gate, and forget gate.



LSTM



- $\tilde{\mathbf{C}}_t = \tanh(\mathbf{w}^{(C)}[\mathbf{a}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(C)})$
- $\Gamma_t^{(u)} = \sigma(\mathbf{w}^{(u)}[\mathbf{a}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(u)})$
- $\Gamma_t^{(f)} = \sigma(\mathbf{w}^{(f)}[\mathbf{a}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(f)})$
- $\Gamma_t^{(o)} = \sigma(\mathbf{w}^{(o)}[\mathbf{a}_{t-1}, \mathbf{x}_t] + \mathbf{b}^{(o)})$
- $\mathbf{C}_t = \Gamma_t^{(u)} \times \tilde{\mathbf{C}}_t + \Gamma_t^{(f)} \times \mathbf{C}_{t-1}$
- $\mathbf{a}_t = \Gamma_t^{(o)} \times \mathbf{C}_t$

Comparison

- GRU: older, simpler.
- LSTM: More complex to understand, more powerful.

To know more

- Cho et al., 2014. "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation".
- Chung et al., 2014. "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling".
- https://www.youtube.com/watch?v=wSabaLGEegM
- Hochreiter & Schmidhuber, 1997. "Long short-term memory".
- Gers et al., 2000. "Learning to Forget: Continual Prediction with LSTM".
- https://www.youtube.com/watch?v=fdY10i0MAQc
- https://jhui.github.io/2017/03/15/RNN-LSTM-GRU/

Thank you.

Q&A