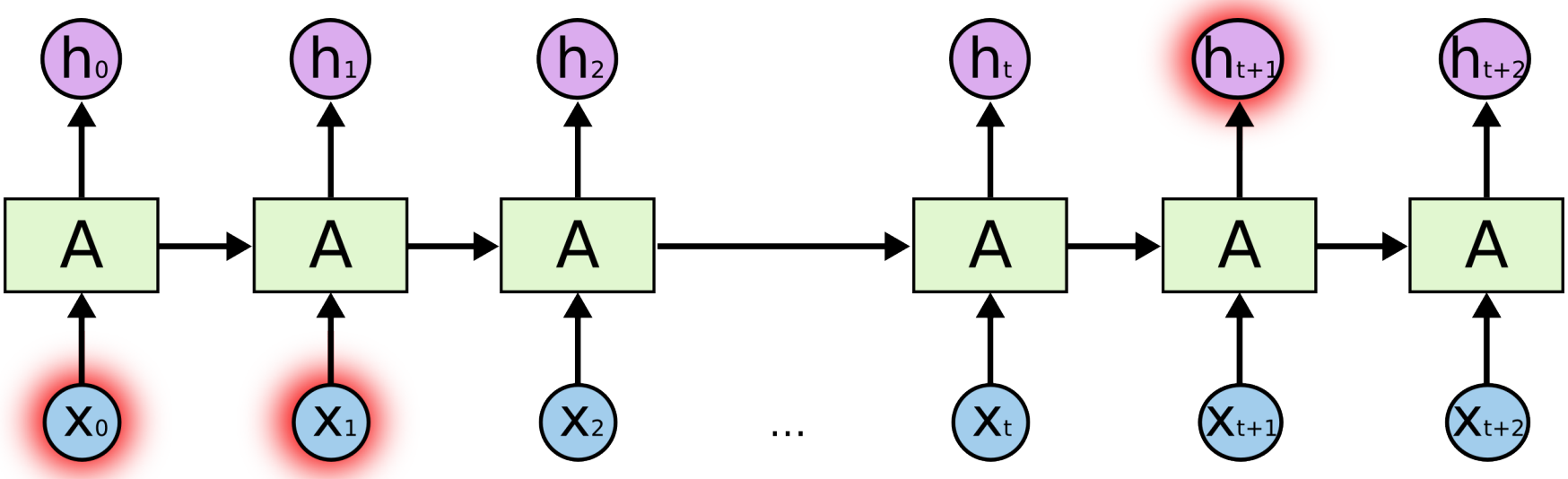
Deep Learning

Vazgen Mikayelyan

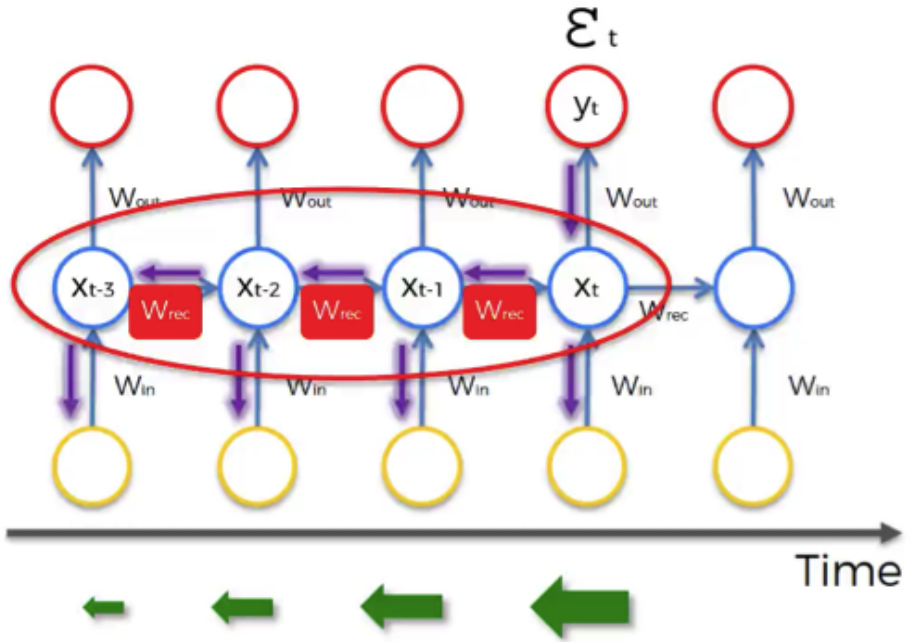
December 1, 2020

FAST DISCOVERING THE FUTURE

Problem of Long Term Dependencies



The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

W_{rec} ~ small ⇨ Vanishing

W_{rec} ~ large ⇨ Exploding

Formula Source: Razvan Pascanu et al. (2013)

Deep Learning A-Z

© SuperDataScience

Outline

Simple RNN

Simple RNN

GRU

GRU

LSTM

LSTM

Additional state

LSTM

LSTM

LSTM

LSTM

LSTM

Why to use tanh?

Outline

BiRNN

Deep RNNs

Outline

Basic Idea

Encode each word in the sentence into a vector using RNNs.

Basic Idea

Encode each word in the sentence into a vector using RNNs.

When decoding, perform a convex combination of these vectors, weighted by attention weights .

Basic Idea

Encode each word in the sentence into a vector using RNNs.

When decoding, perform a convex combination of these vectors, weighted by "attention weights".

Use this combination in picking the next word.

Attention Model