Recurrent Neural Networks

Introduction and overview of the latest advances and trends in the field

Roberto Souza
Assistant Professor
Electrical and Computer Engineering
Schulich School of Engineering

March 2021

UNIVERSITY OF CALGARY

Outline

- Learning Goals
- Motivation
- Recurrent Neural Networks (RNNs)
  - Basic architecture
  - Long short-term memory (LSTM)
  - Gated Recurrent Unit
- Summary

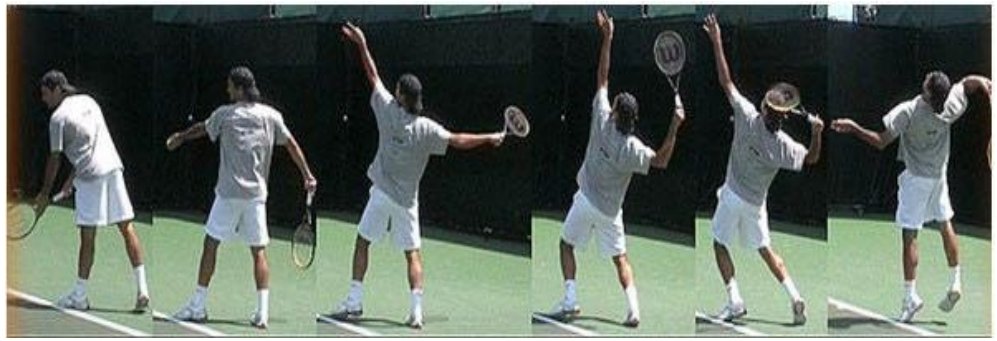
Learning Goals

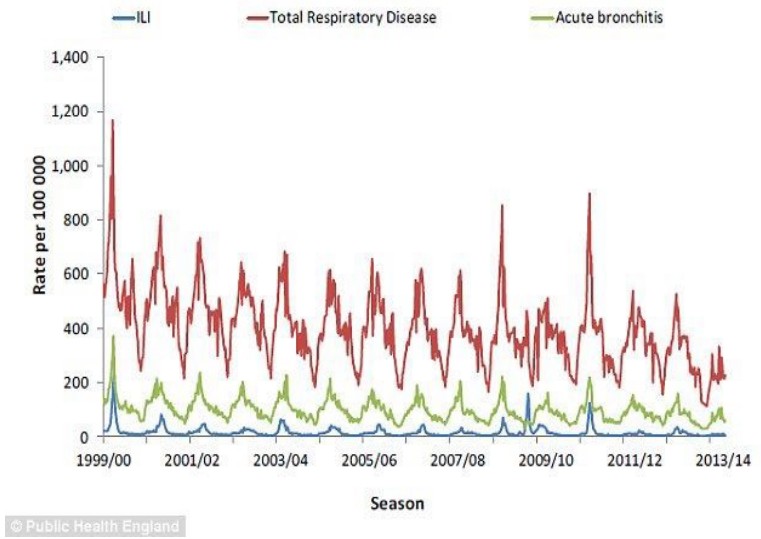
- Learn the intuition behind RNNs
- Get familiar with the most common types of RNNs (LSTM, GRU)

Motivation - Data is often sequential in nature

Steph Curry releases the ball and as it moves you know it is going to be 3 points to Golden State…

Motivation - Data is often sequential in nature





UNIVERSITY OF CALGARY

Introduction

- Building models of sequential data is important: automatic speech recognition, machine translation, natural language, …

- Recurrent nets (RNN) are a simple and general framework for this type of tasks

Introduction

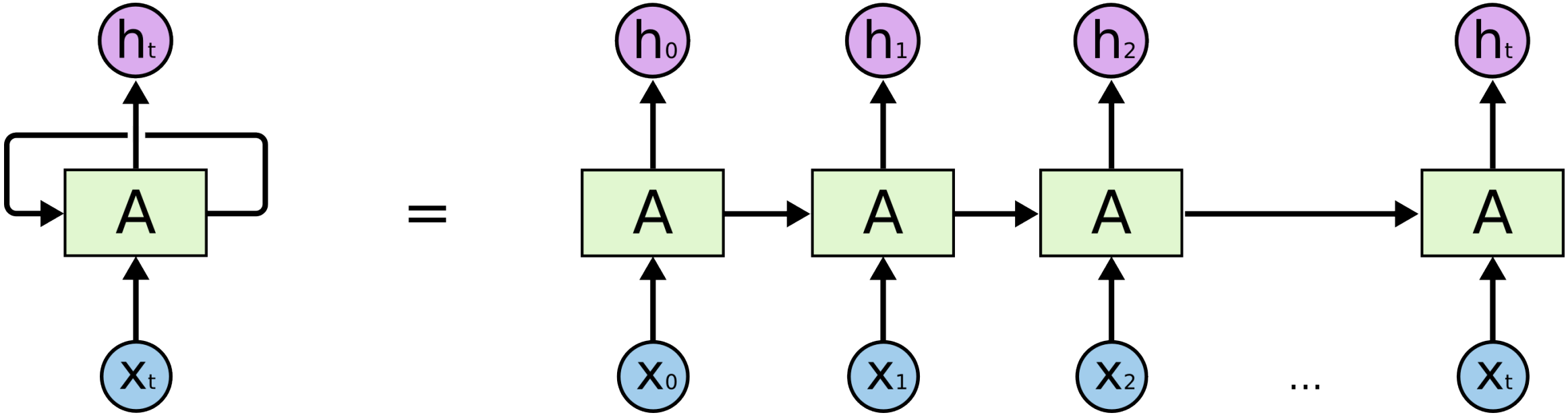
A B C A B C A B _

- What symbol comes next?
- What is its probability?

Yesterday it was Sunday, so today it must be _

- How to predict the next word?
- What is this good for?

Simple RNN



Major shortcomings

- Handling of complex non-linear interactions
- Difficulties using BPTT to capture long-term dependencies exploding gradients
- Vanishing gradients

Training: Backpropagation through time (BPTT)

- How to train the recurrent nets?
- The output value does depend on the state of the hidden layer, which depends on all previous states of the hidden layer (and thus, all previous inputs)
- Recurrent net can be seen as a (very deep) feedforward net with shared weights

Training: Backpropagation through time (BPTT)

- How to train the recurrent nets?
- The output value does depend on the state of the hidden layer, which depends on all previous states of the hidden layer (and thus, all previous inputs)
- Recurrent net can be seen as a (very deep) feedforward net with shared weights

Vanishing gradients

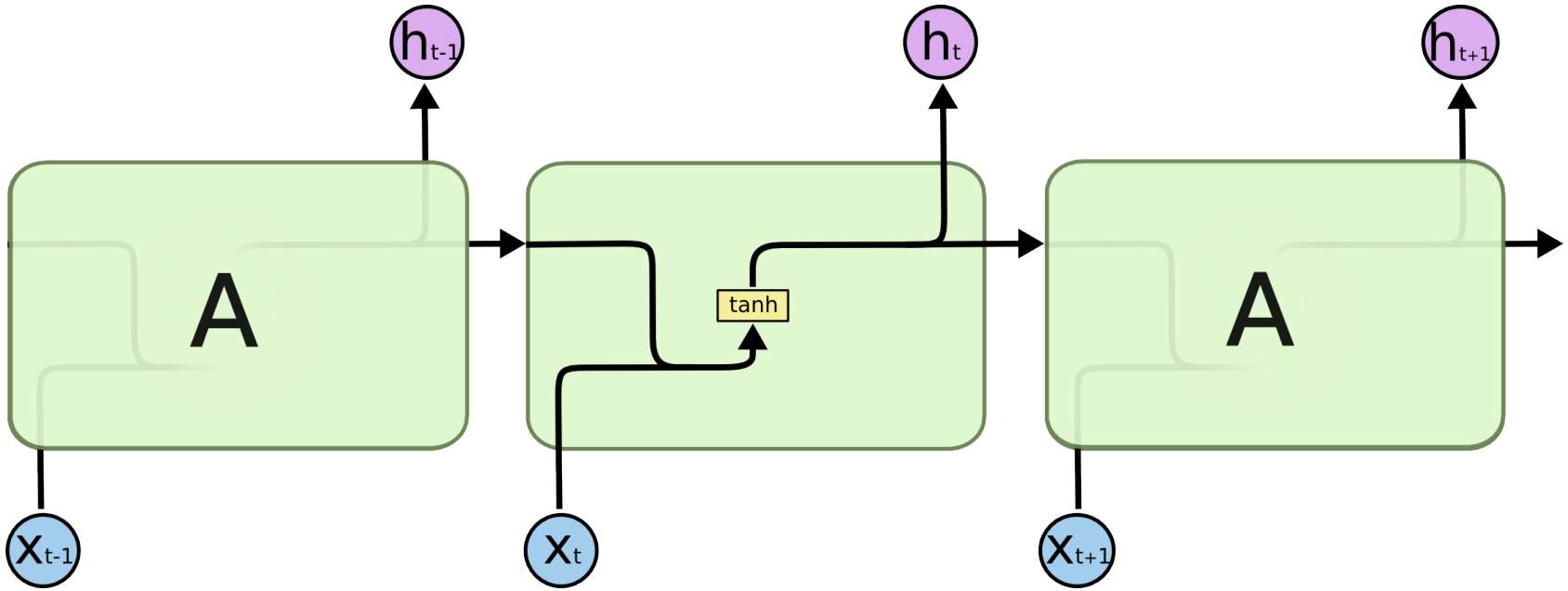
- As we propagate the gradients back in time, usually their magnitude quickly decreases: this is called “vanishing gradient problem”

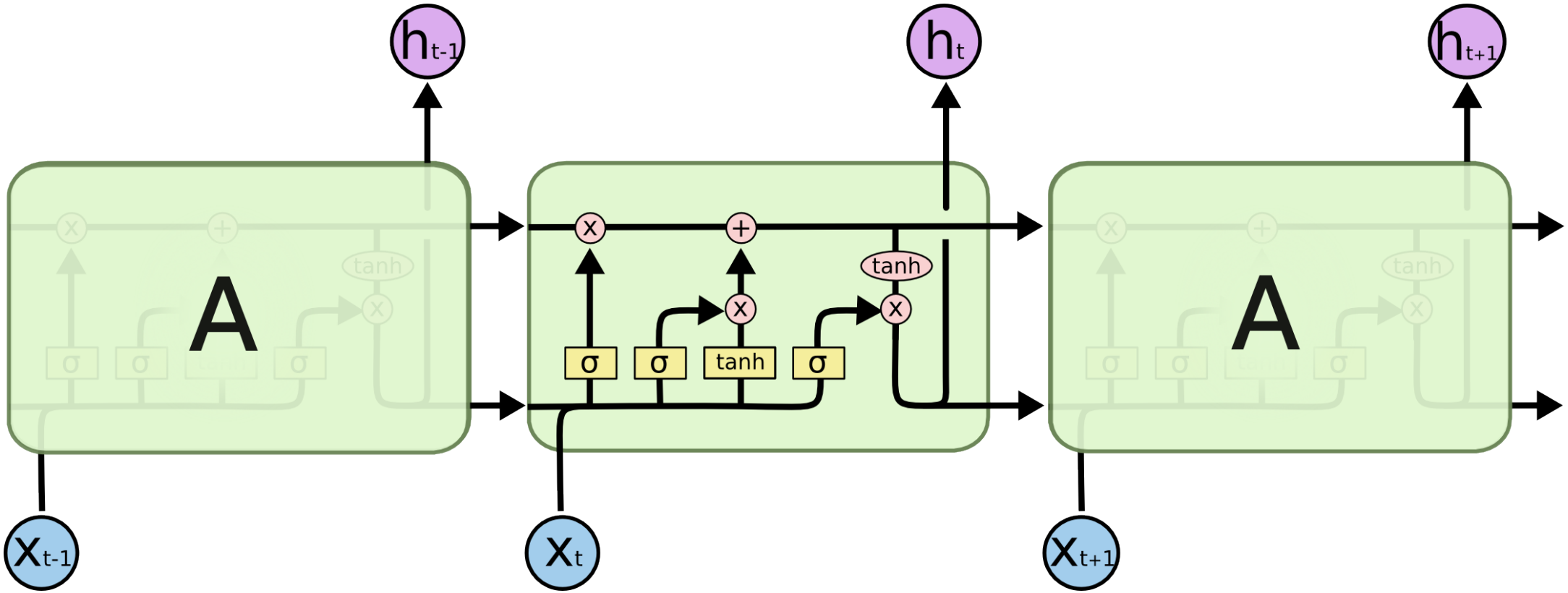
- In practice this means that learning long term dependencies in data is difficult for simple RNN architecture

- Special RNN architectures address this problem:
  - *Exponential trace memory* (Jordan 1987, Mozer 1989)
  - *Long Short-term Memory* (Hochreiter & Schmidhuber, 1997))
  - will be described in the second part of this lecture

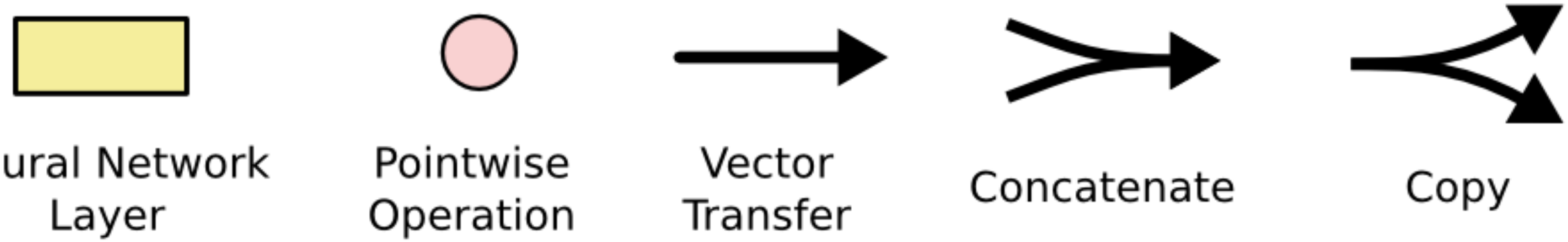
Exploding gradients

- Sometimes, the gradients start to increase exponentially during backpropagation through the recurrent weights
- Happens rarely, but the effect can be catastrophic: huge gradients will lead to big change of weights, and thus destroy what has been learned so far
- One of the main reasons why RNNs were supposed to be unstable
- Simple solution (first published in RNNLM toolkit in 2010): clip or normalize values of the gradients to avoid huge changes of weights





Summary



Thank you!