

# Recurrent Neural Network

Group 6

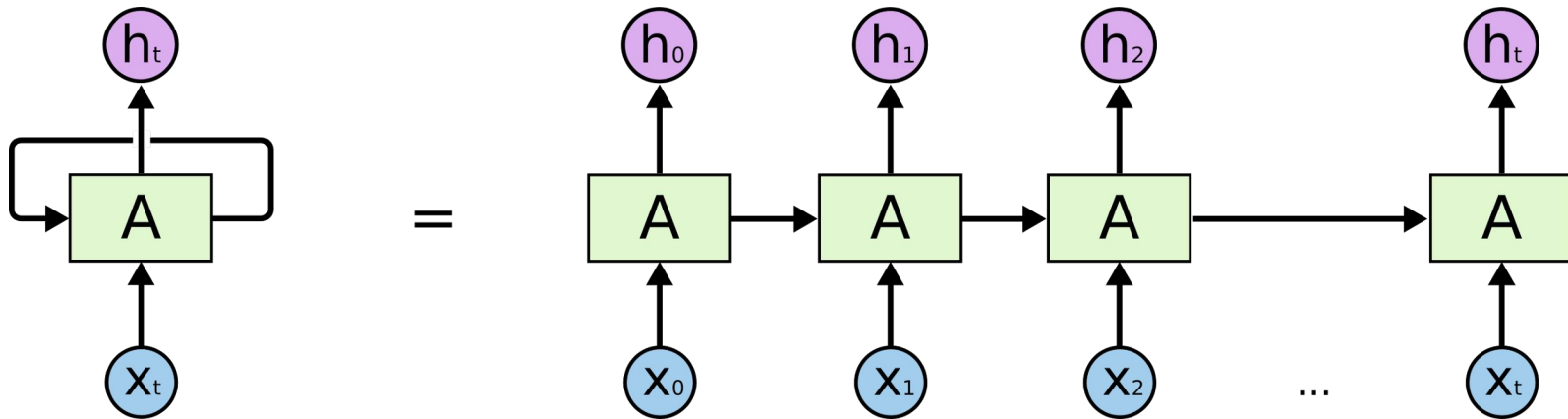
# Introduction

- A recurrent neural network (RNN) is a class of artificial neural network where connections between units form a **directed cycle**.
- This creates an internal state of the network which allows it to exhibit **dynamic temporal behavior**.
- Unlike feedforward neural networks, RNNs can use their **internal memory** to process arbitrary sequences of inputs.
- This makes them applicable to tasks such as unsegmented connected handwriting recognition or speech recognition.

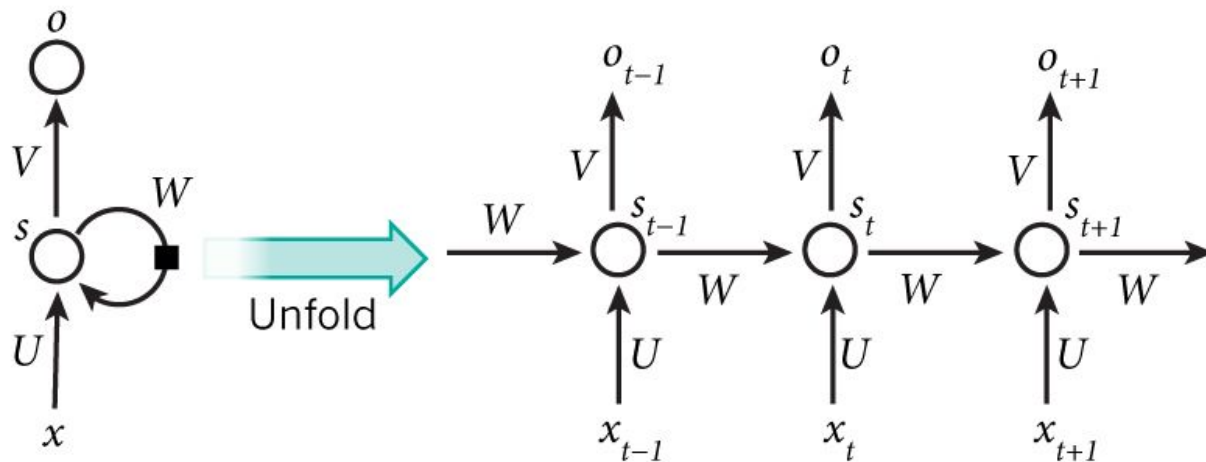
# Example

*'I grew up in France... I speak fluent \_\_\_\_\_'*

# RNN



# Vanilla RNN

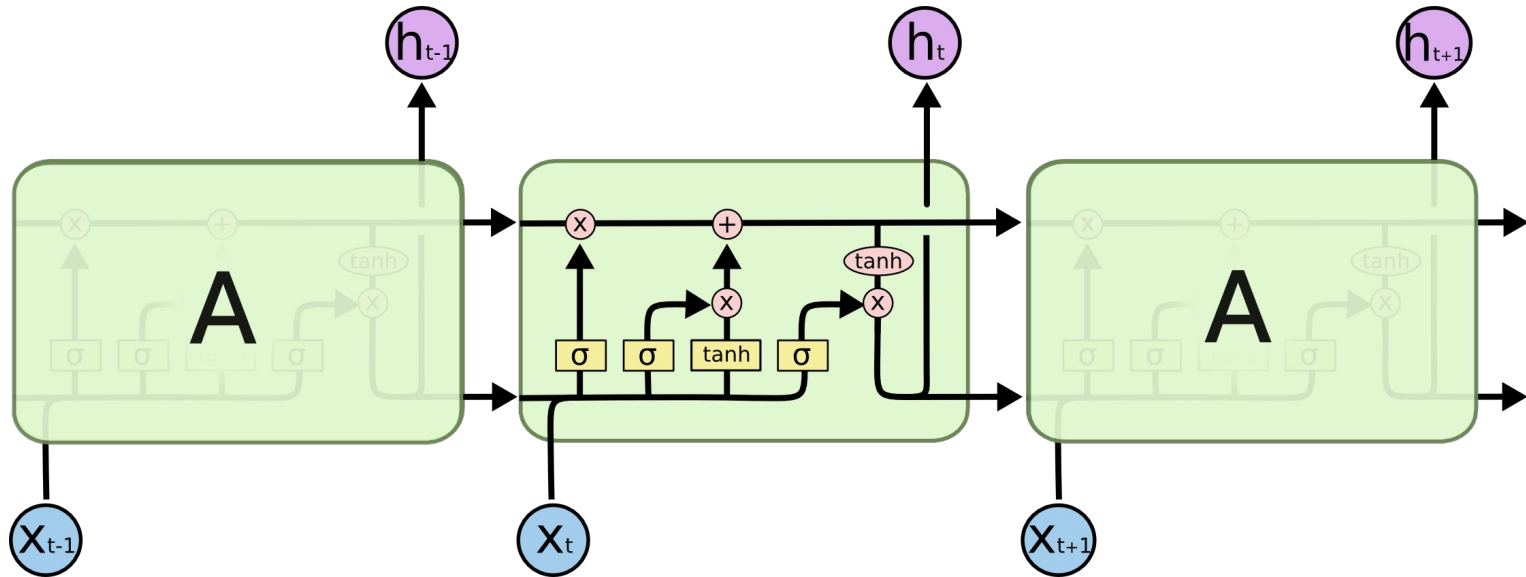


$$s_t = f(Ux_t + Ws_{t-1})$$
$$o_t = \text{softmax}(Vs_t)$$

# LSTM

Long *Short Term* Memory

# LSTM



# Implementing RNNs

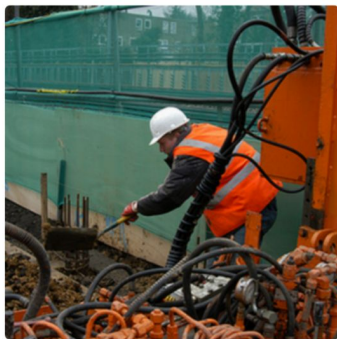
<http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-2-implementing-a-language-model-rnn-with-python-numpy-and-theano/>

- Initialize  $U, V, W$  to some random value
- Forward propagation using the equations of RNN (Save intermediate steps)
- Train with SGD and Backpropagation through Time (BPTT)





"man in black shirt is playing guitar."



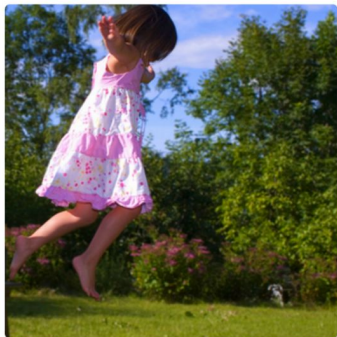
"construction worker in orange safety vest is working on road."



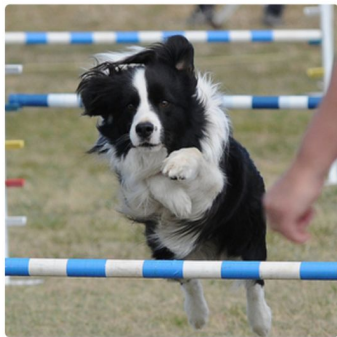
"two young girls are playing with lego toy."



"boy is doing backflip on wakeboard."



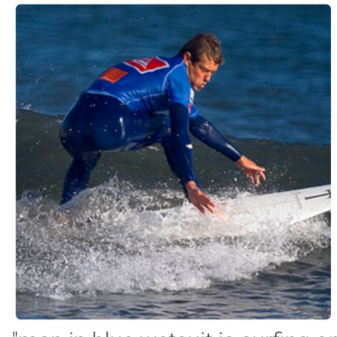
"girl in pink dress is jumping in air."



"black and white dog jumps over bar."



"young girl in pink shirt is swinging on swing."



"man in blue wetsuit is surfing on wave."

# Issues

Most RNNs used to have scaling issues. In particular, RNNs could not be easily trained for large numbers of neuron units nor for large numbers of inputs units. Recent advances in training techniques have greatly increased the capabilities of recurrent neural networks. Successful training has been mostly in time series problems with few inputs and in chemical process control.

# References

- <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- <http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/>
- <http://karpathy.github.io/2015/05/21/rnn-effectiveness/>
- <https://github.com/dennybritz/rnn-tutorial-rnnlm/>
- <https://www.tensorflow.org/versions/r0.7/tutorials/recurrent/index.html>
- <http://nikhilbuduma.com/2015/01/11/a-deep-dive-into-recurrent-neural-networks/>
- [https://en.wikipedia.org/wiki/Recurrent\\_neural\\_network](https://en.wikipedia.org/wiki/Recurrent_neural_network)

# Architectures

Fully recurrent network

Recursive neural networks

Hopfield network

Elman networks and Jordan networks

Echo state network

Neural history compressor

Long short term memory

Bi-directional RNN

Continuous-time RNN

# Architectures

Hierarchical RNN

Recurrent multilayer perceptron

Second order RRN

Multiple timescales recurrent neural network (MTRNN) model

Pollack's sequential cascaded networks

Neural Turing Machines

Neural network pushdown automata

Bidirectional associative memory

# Various RNN Architectures

<http://jmlr.org/proceedings/papers/v37/jozefowicz15.pdf>

<http://arxiv.org/pdf/1503.04069.pdf>

# Training

Gradient Descent

Global Optimization Methods