Sequence Modeling: Recurrent and Recursive Networks

Markus Dumke

27th January 2016

Contents

Why RNN's?

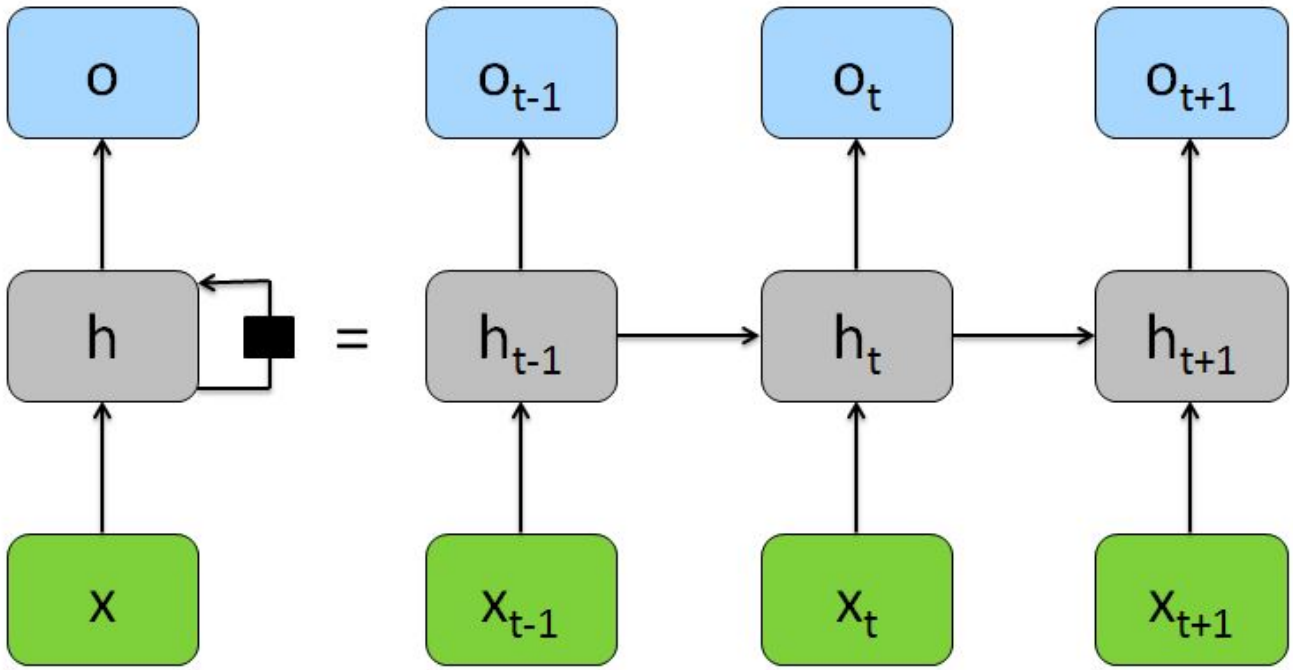
- sequential data
- outputs depend on all previous inputs (no independence)
- long-term dependencies
- memory

Applications

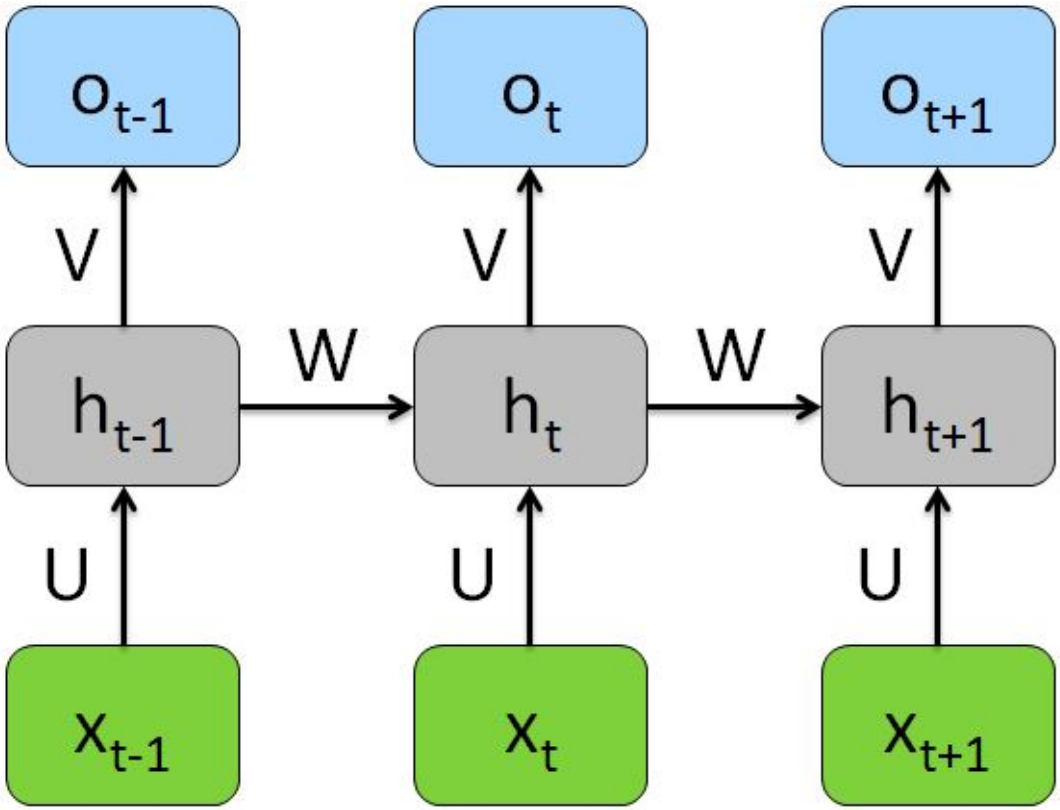
Natural Language Processing

- machine translation
- character- or word-level language model
- text summary or labels
- sentiment analysis
- image captioning
- handwriting recognition and generation
- speech recognition and generation
- time series data
- ...

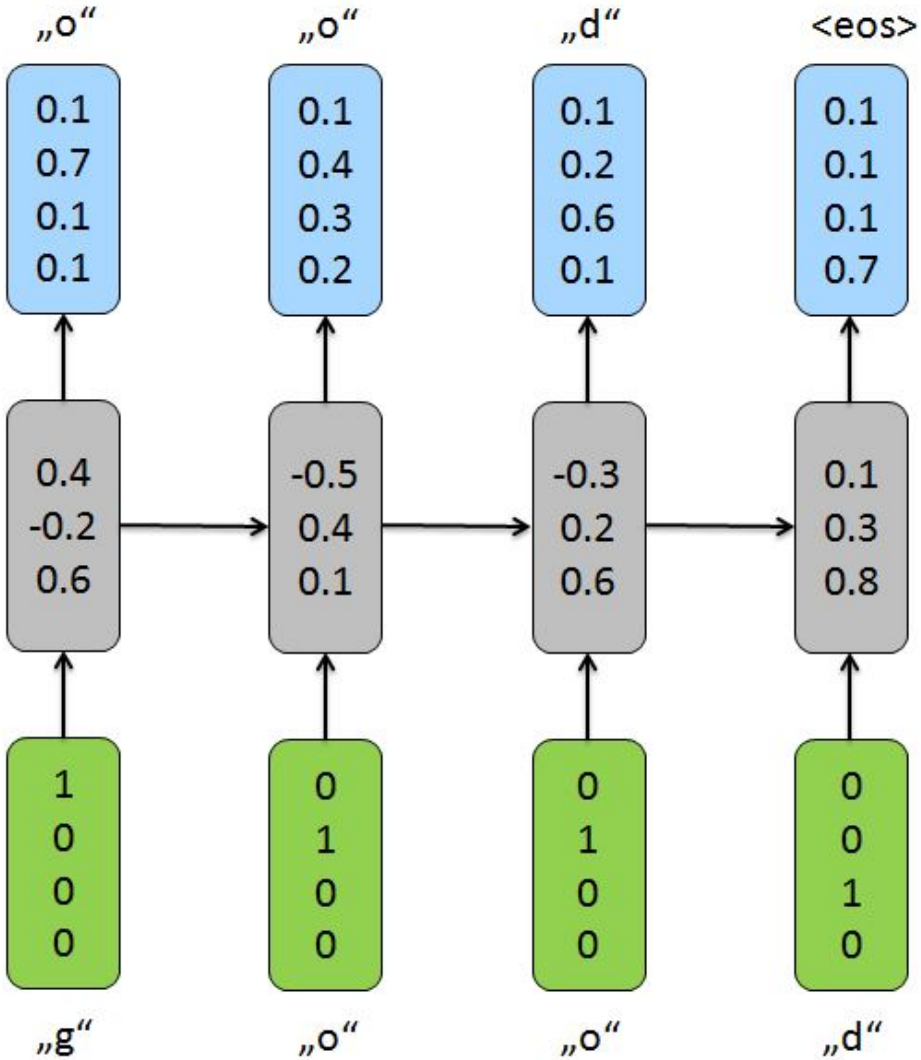
Recurrent Neural Network



Recurrent Neural Network



Recurrent Neural Network



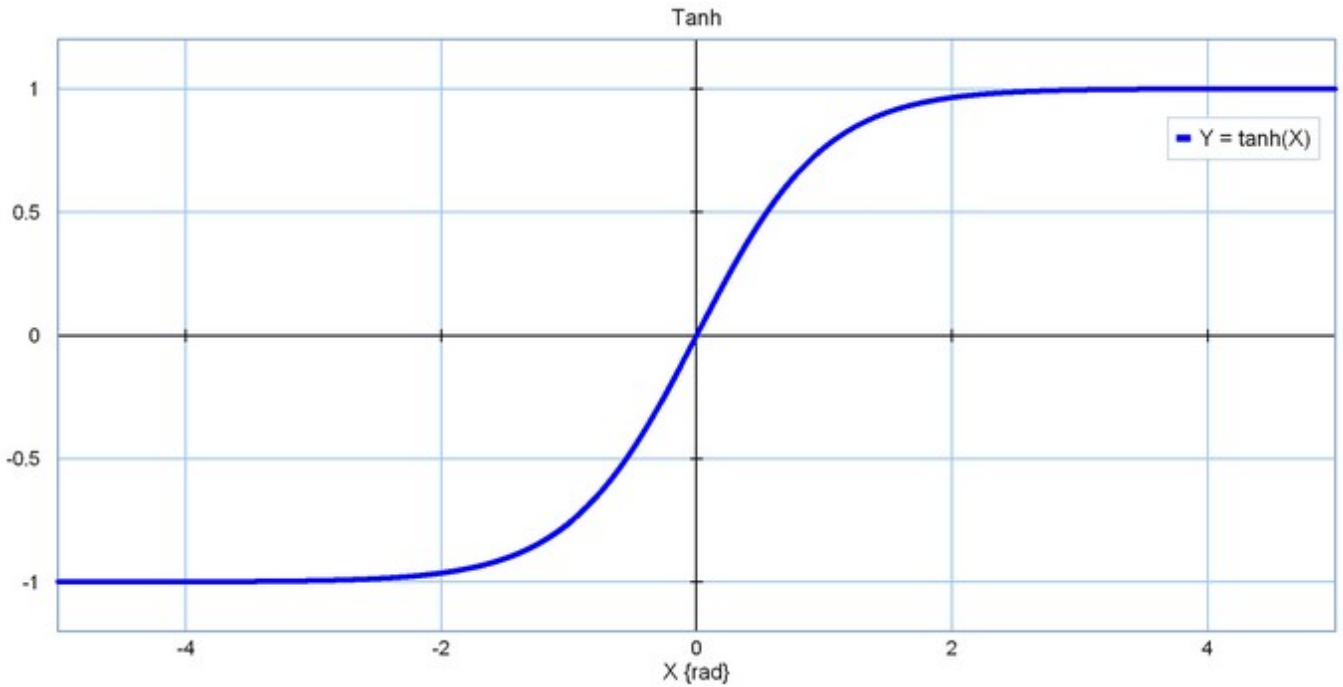
Recurrent Neural Network

for t = 1 to τ:

$$a^{(t)} = b + Wh^{(t-1)} + Ux^{(t)}$$
$$h^{(t)} = f(a^{(t)})$$
$$o^{(t)} = c + Vh^{(t)}$$
$$\hat{y}^{(t)} = softmax(o^{(t)})$$

Which activation function?

$$f(x) = tanh(x) = \frac{sinh(x)}{cosh(x)} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

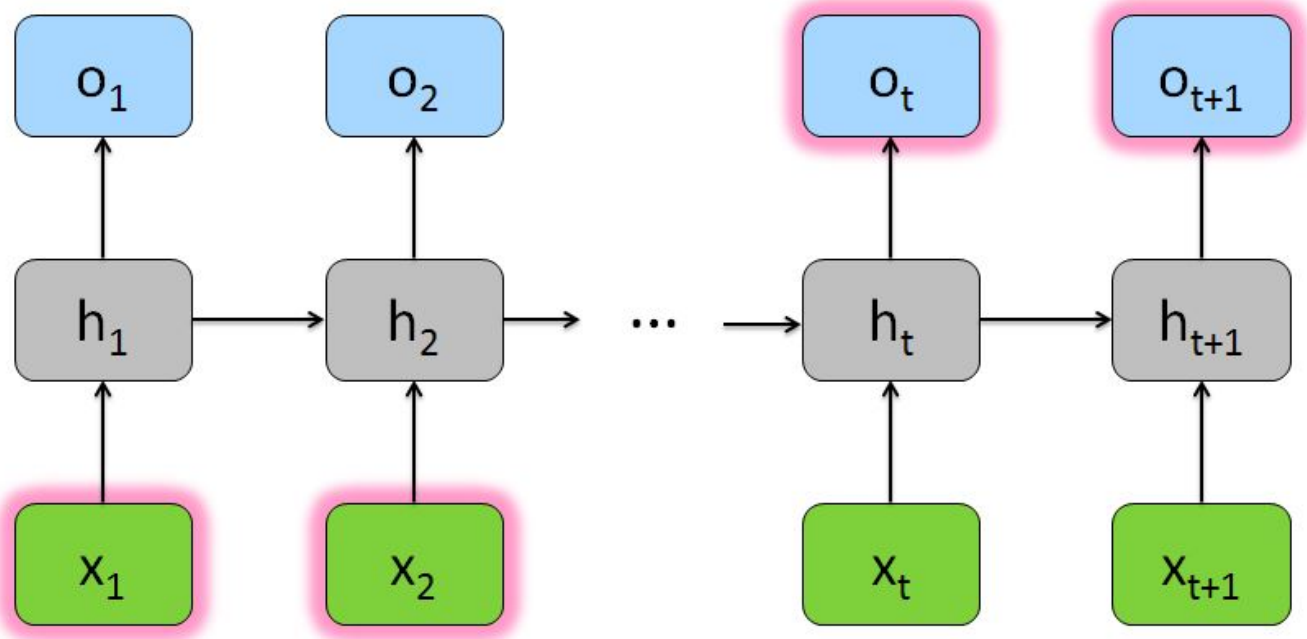


http://www.20sim.com/webhelp/language_reference_functions_tanh.php

Optimization

- Forward Propagation, compute loss
- Backward Propagation through time (BPTT), compute gradients
- Stochastic Gradient Descent (Minibatch)

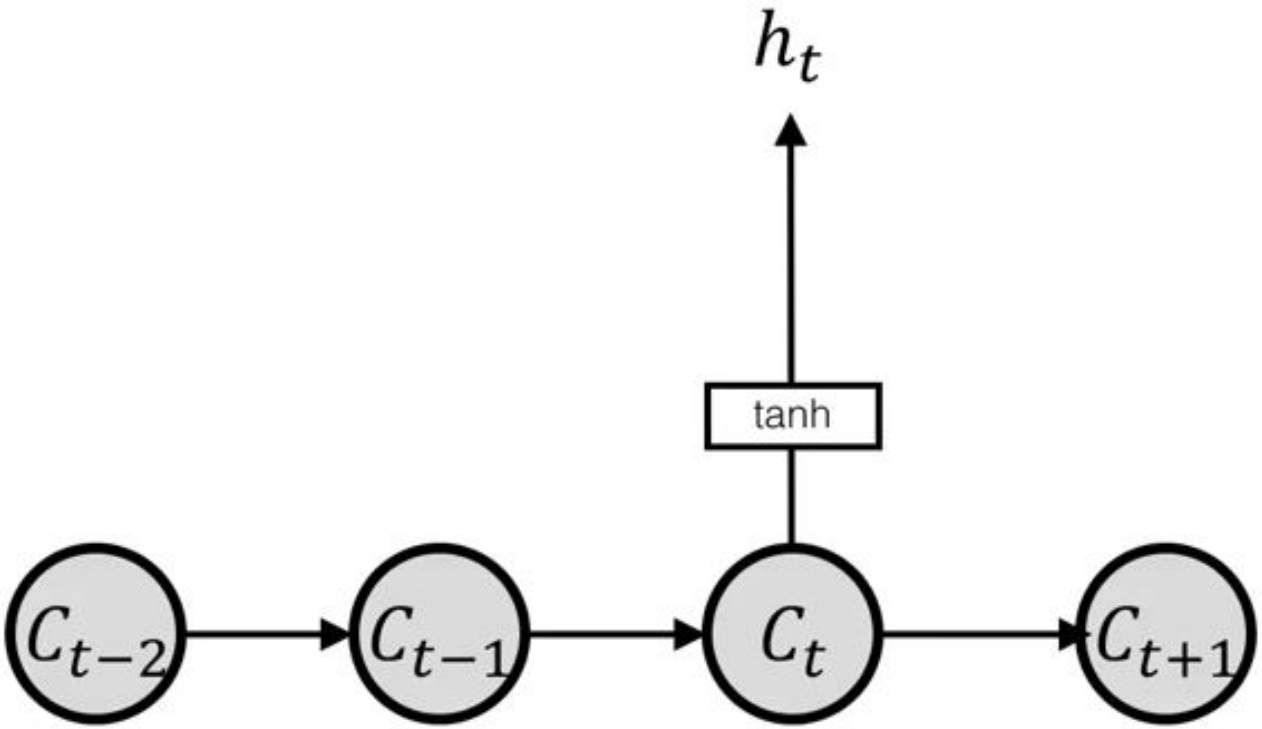
Vanishing (and Exploding) Gradient Problem



How to deal with vanishing gradients?

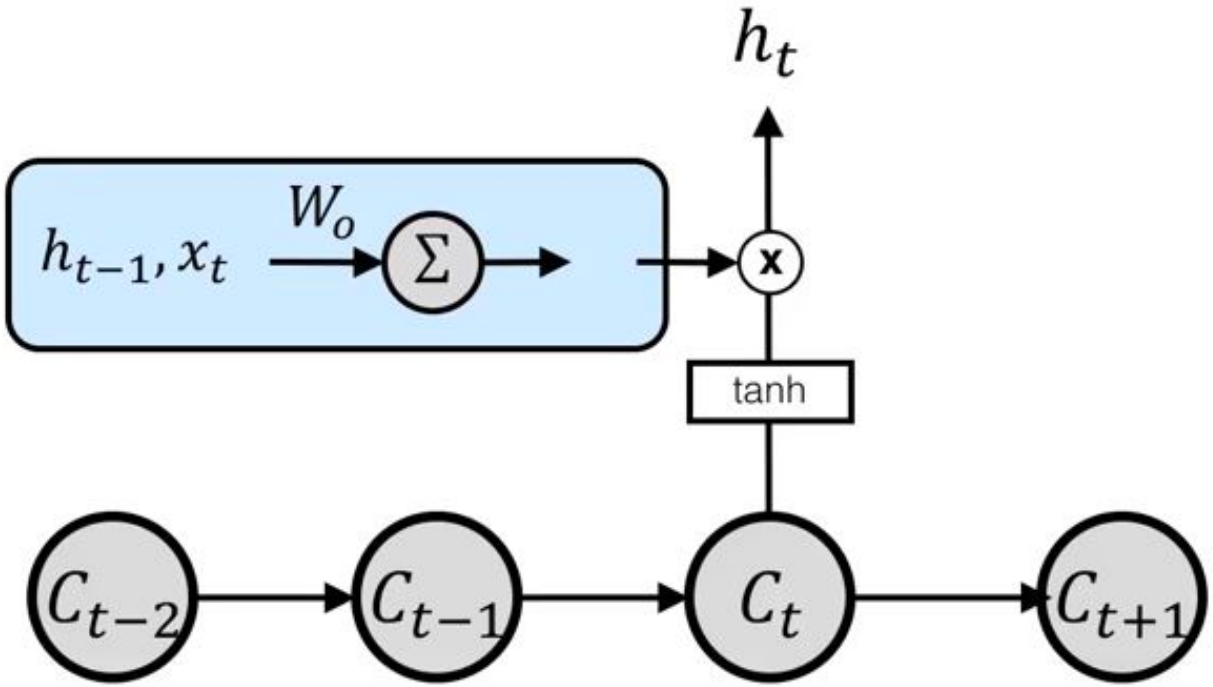
- Gradient Clipping
- Regularization
- Leaky Units
- different time scales
- **LSTM**, GRU and variants

LSTM



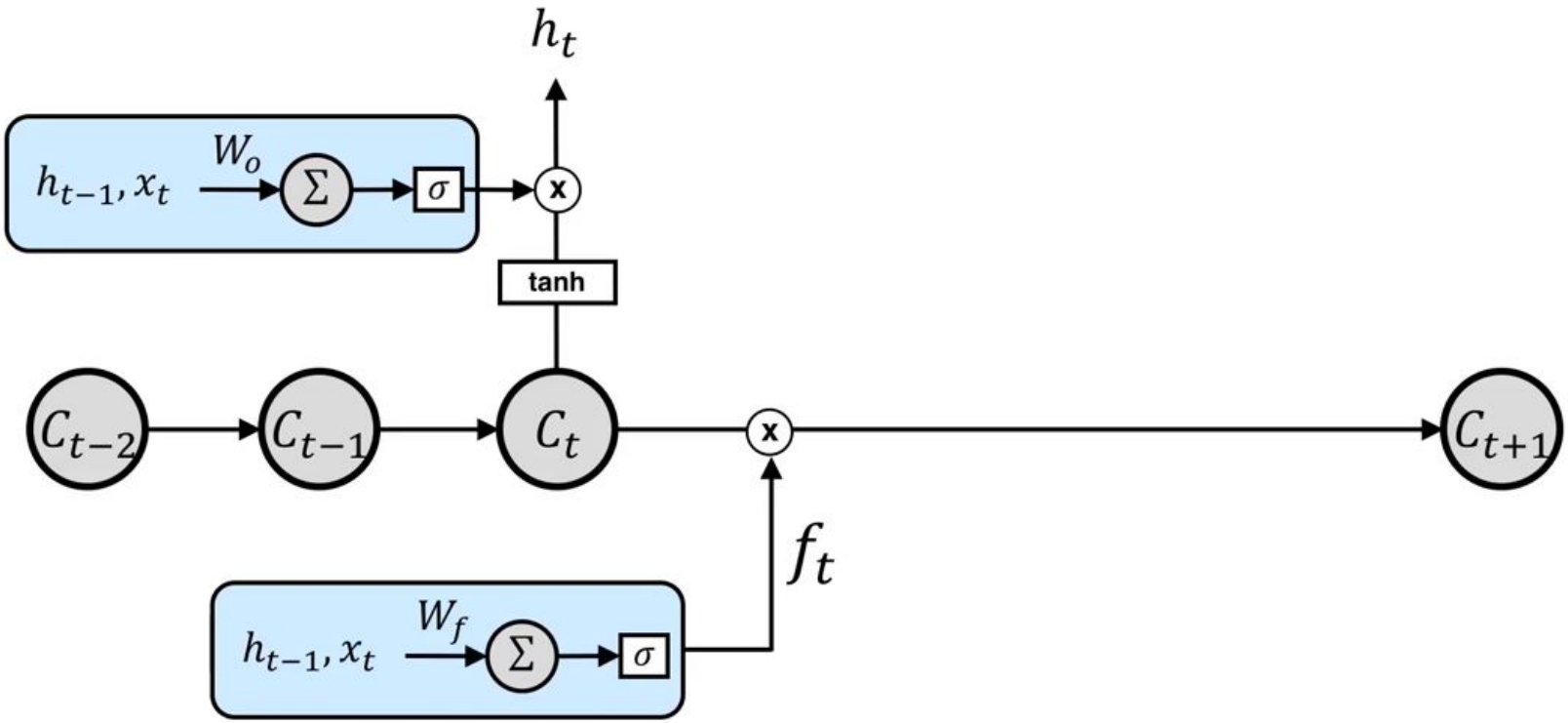
https://www.nervanasys.com/recurrent-neural-networks

LSTM



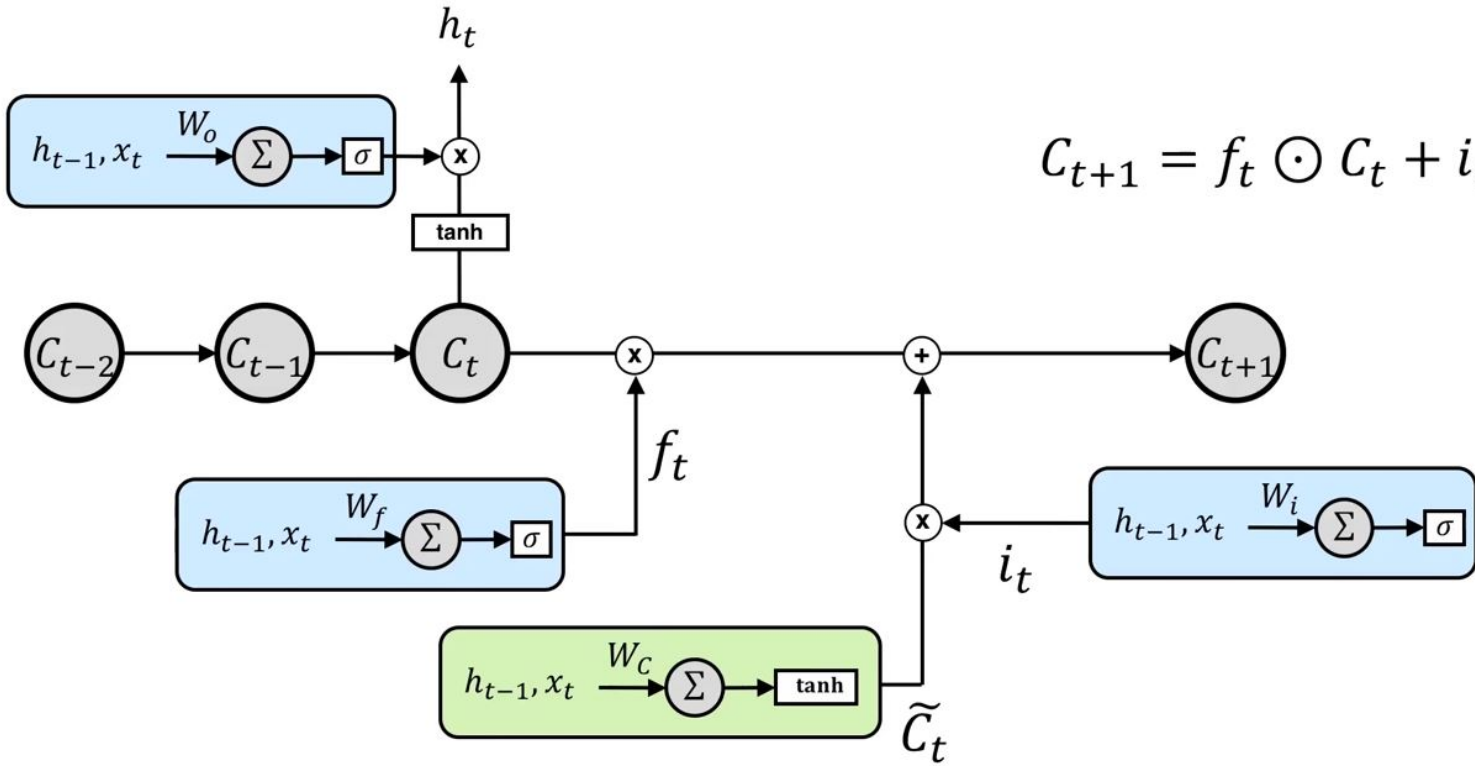
https://www.nervanasys.com/recurrent-neural-networks

LSTM



https://www.nervanasys.com/recurrent-neural-networks

LSTM



$$C_{t+1} = f_t \odot C_t + i_t \odot \tilde{C}_t$$

https://www.nervanasys.com/recurrent-neural-networks

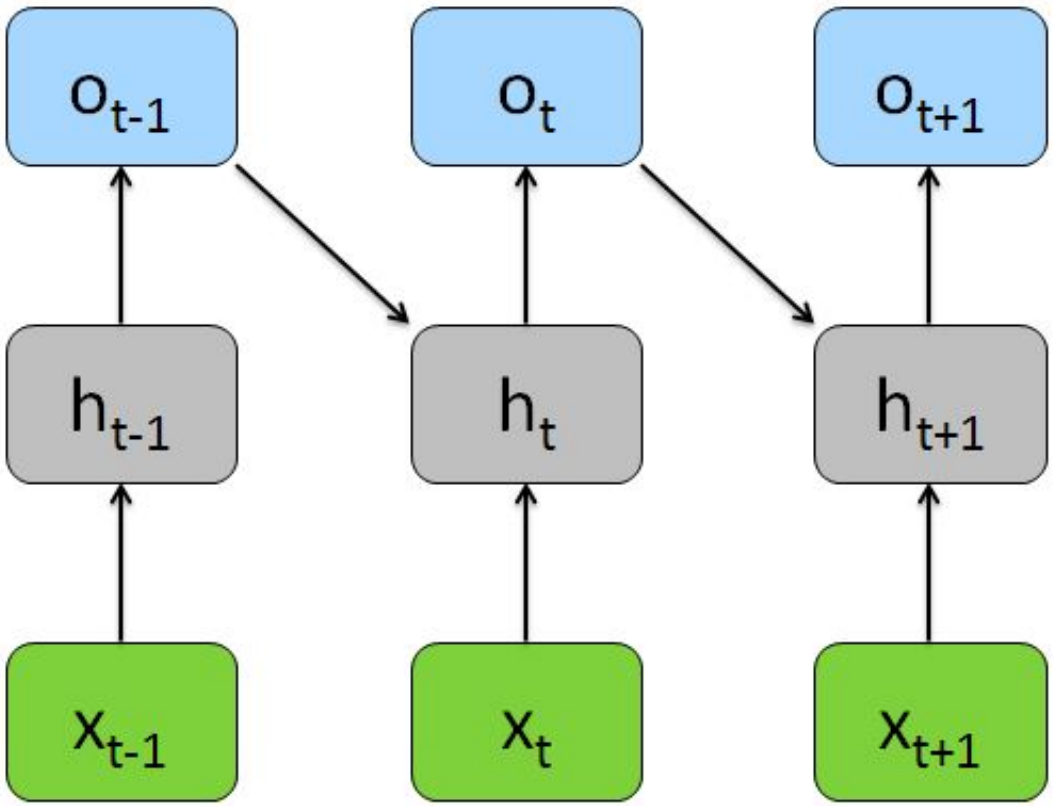
Sampling from an RNN

- sample from conditional distribution at each time step
- how to generate sequence length?
- special end symbol
- Bernoulli random variable
- integer value τ

Language Modeling

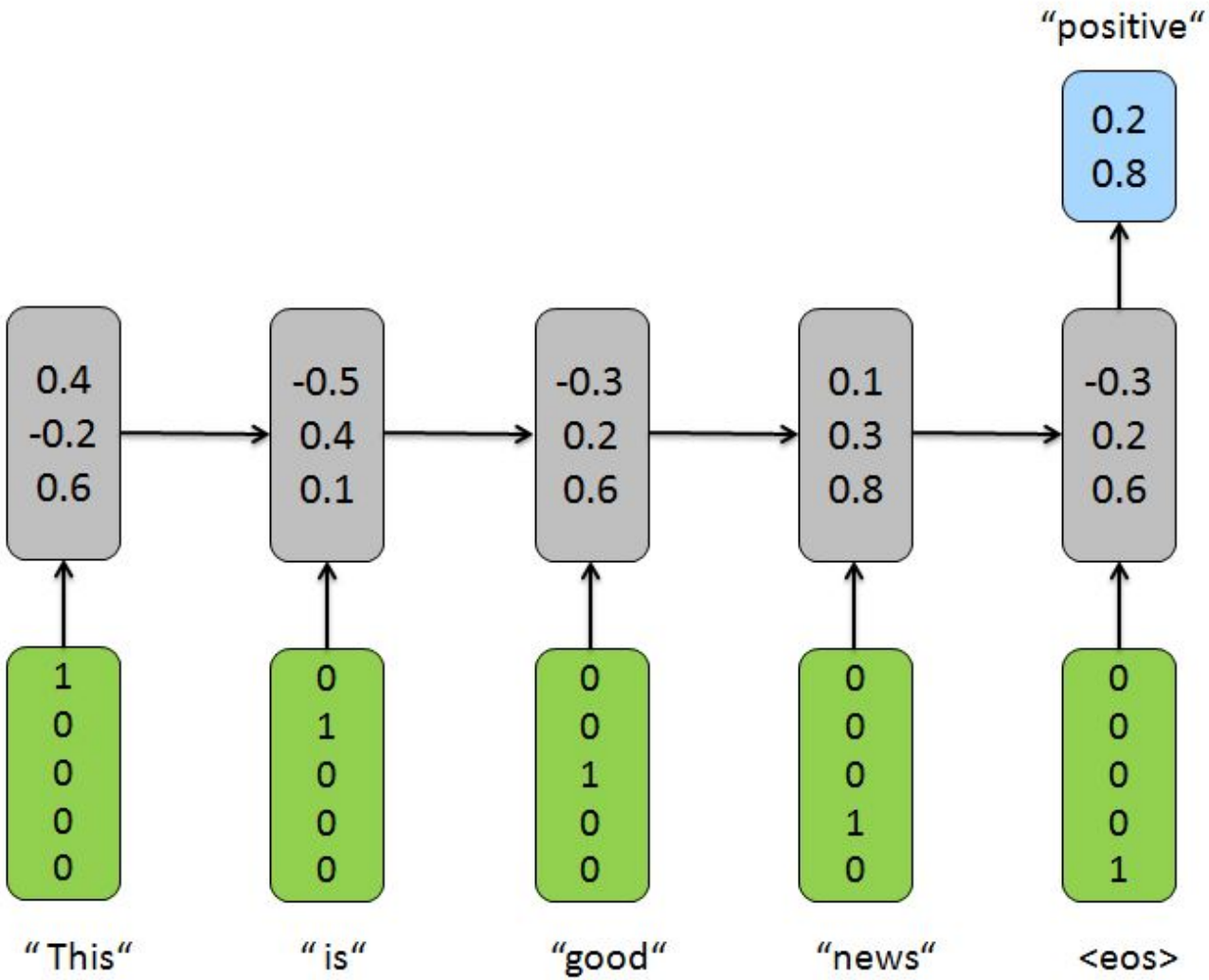
- Output: Probability distribution over words given previous words
- $P(y1, ..., yn) = \prod_{i=1}^{n} P(yi|y1, ..., y_{i-1})$
- scoring candidates
- word-level or character-level possible
- Input: word/character encoded as one-hot vector

output-to-hidden RNN

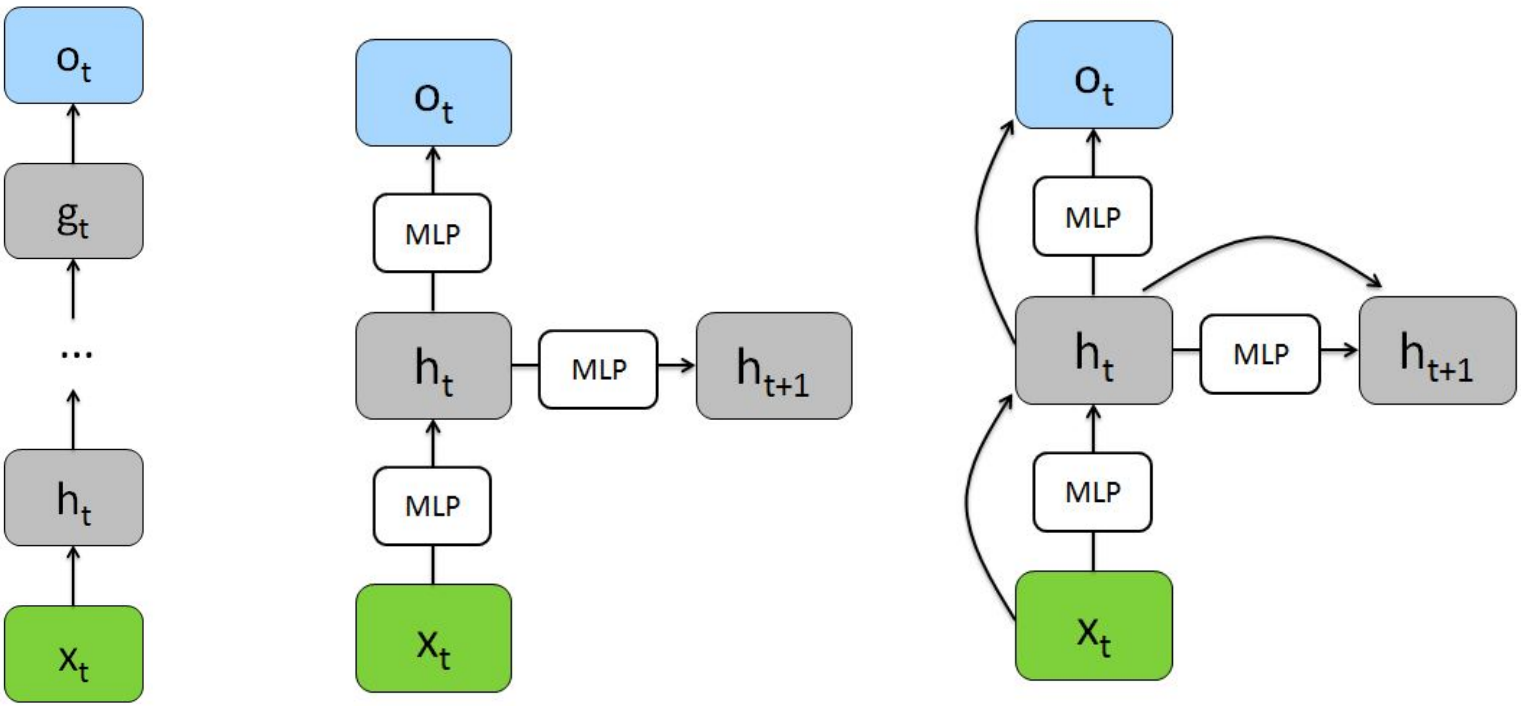


Teacher Forcing

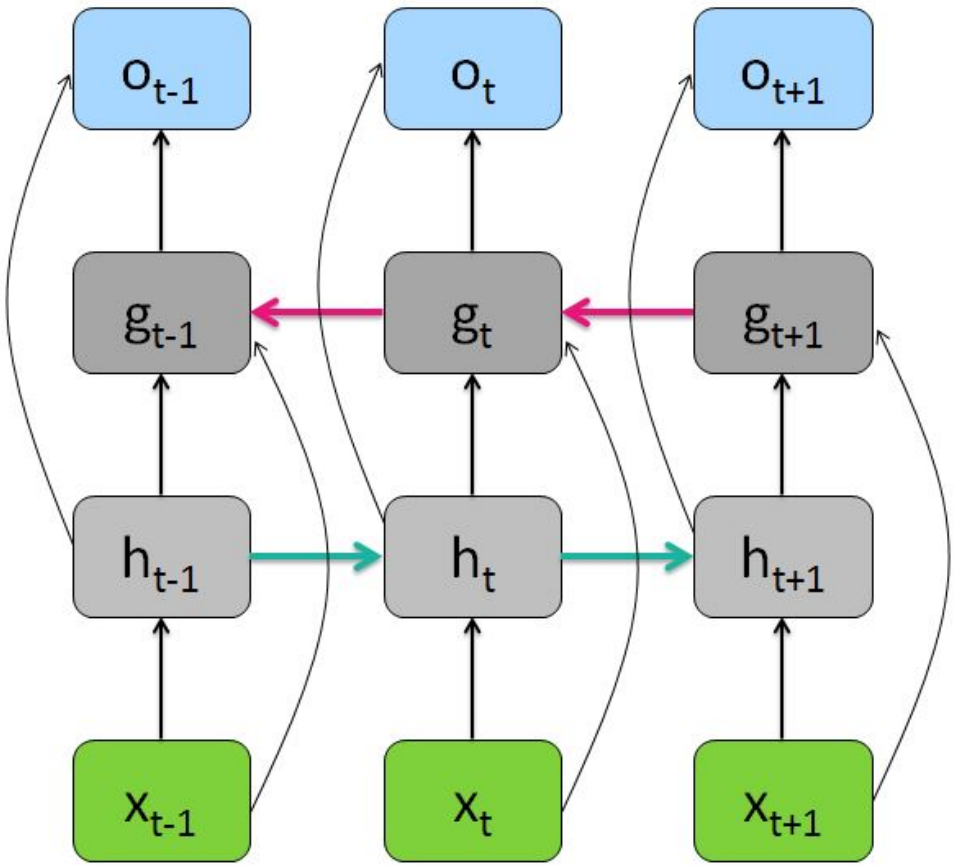
One-output RNN



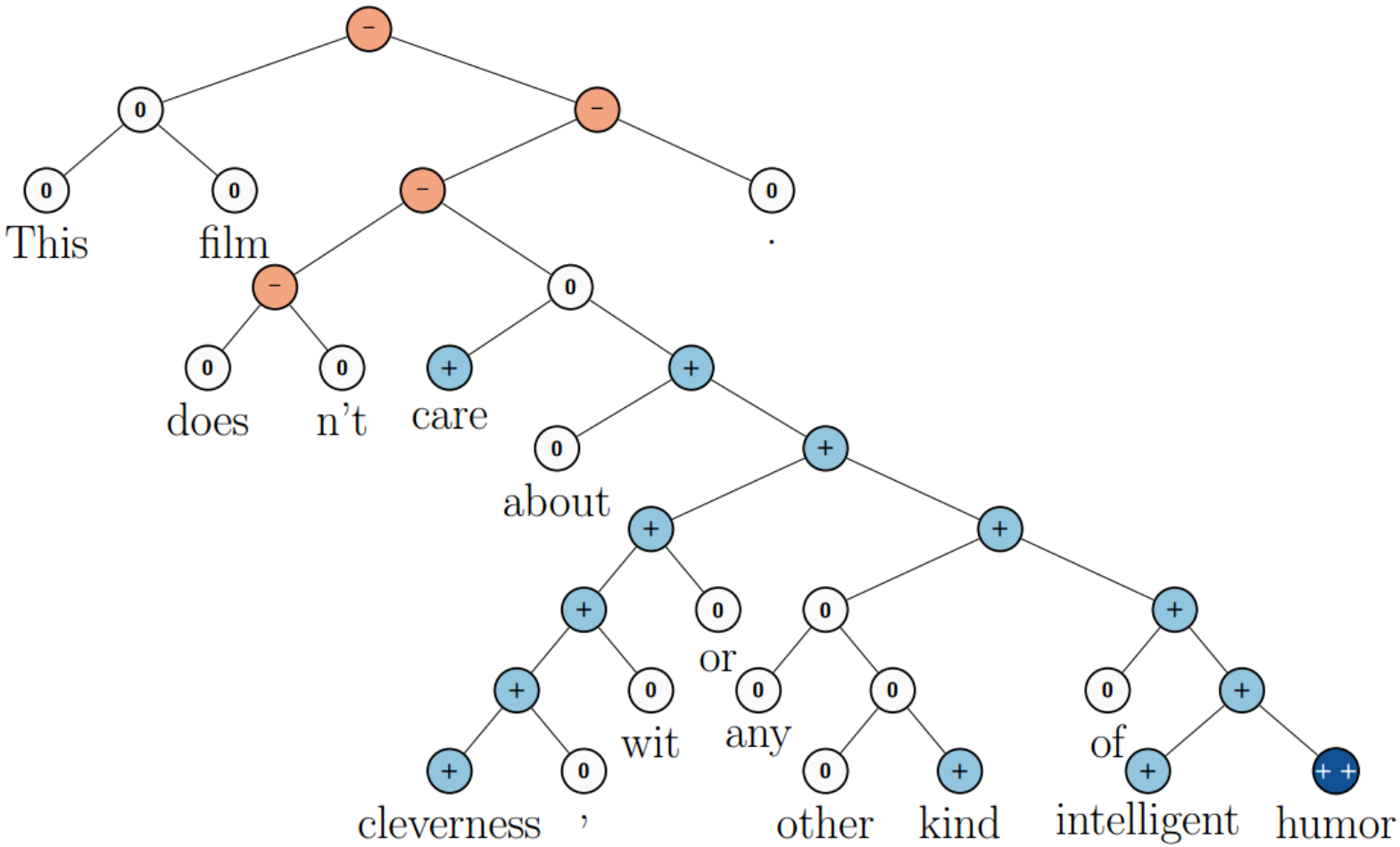
Deep RNNs



Bidirectional RNN

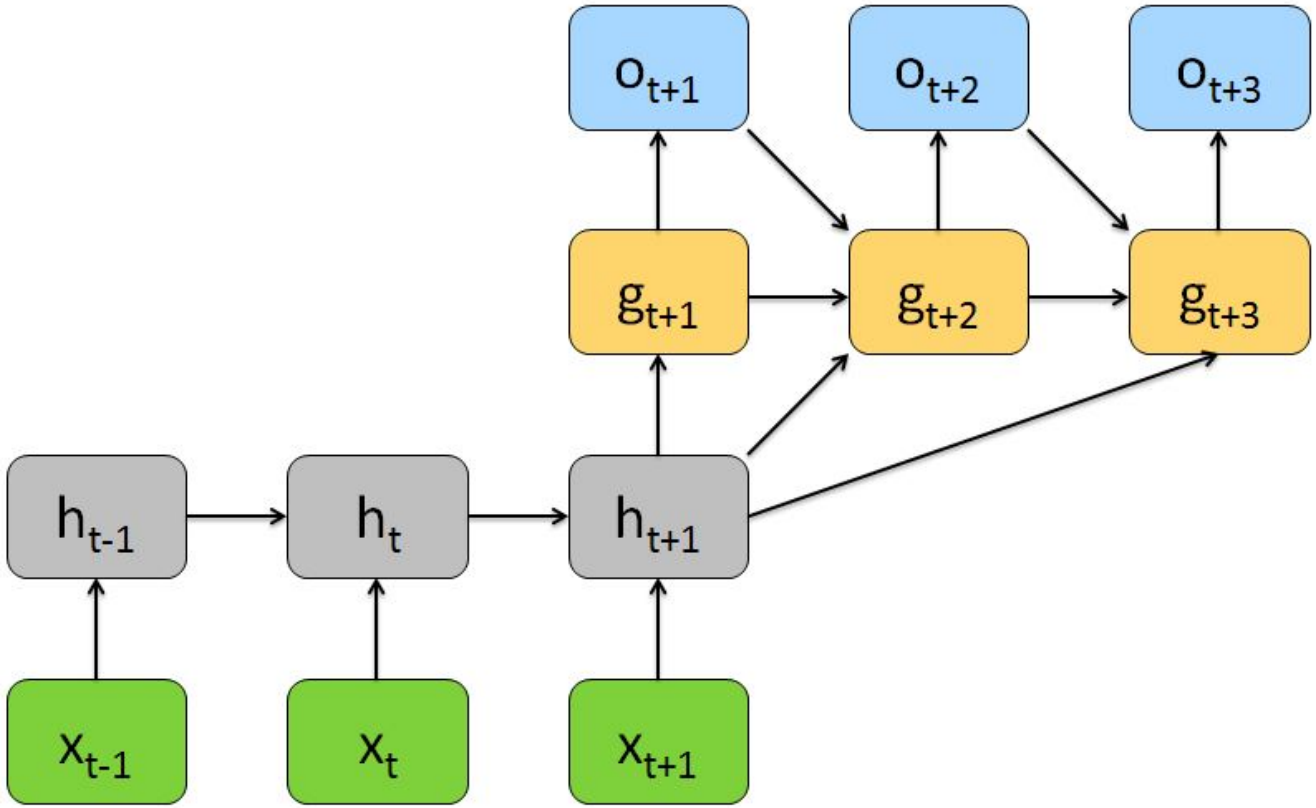


Recursive Neural Network

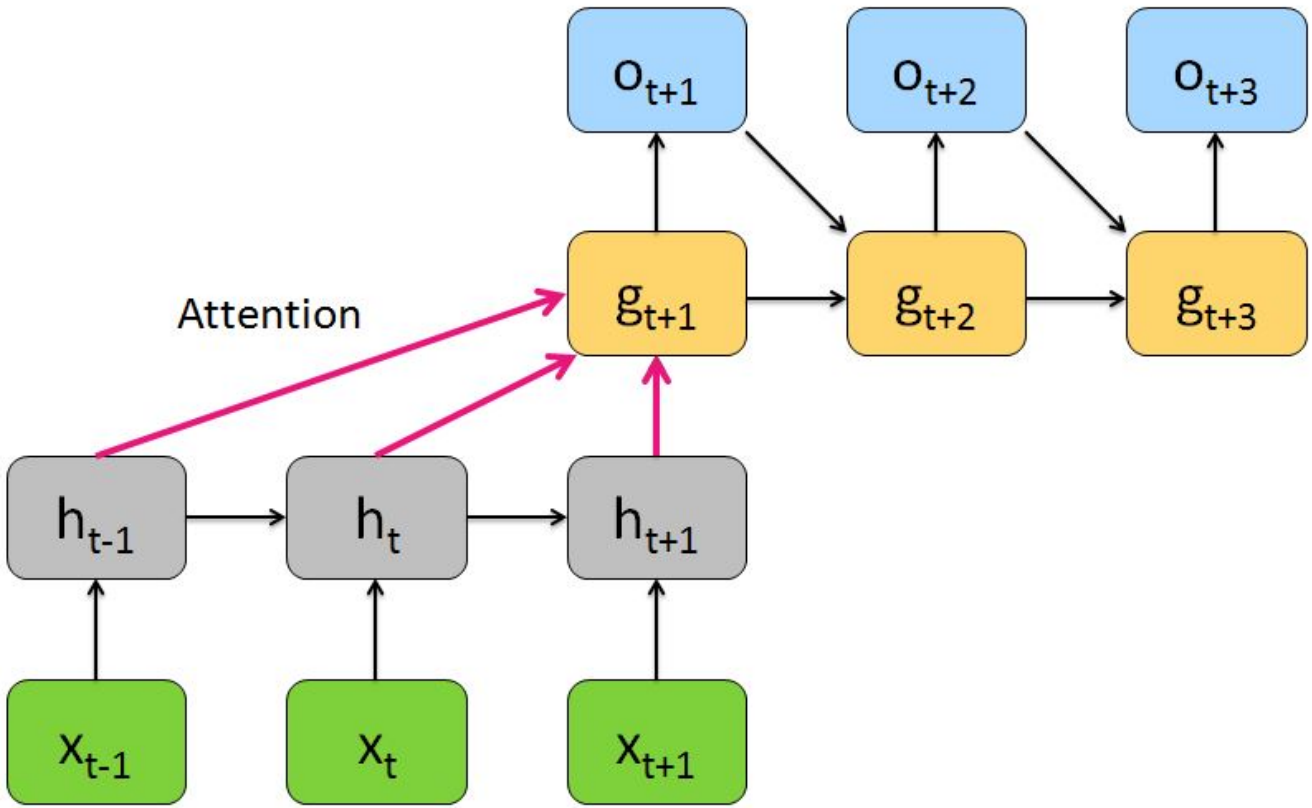


Socher et al. (2013)

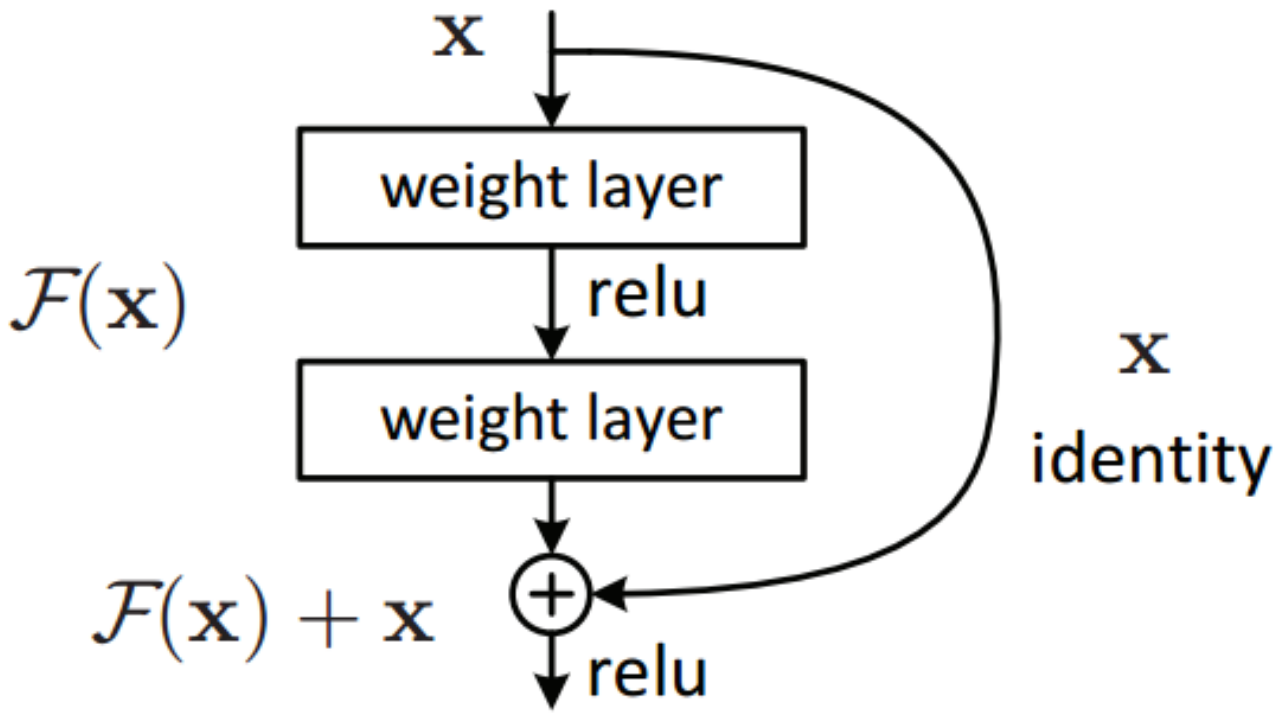
Encoder-Decoder Architecture



Attention

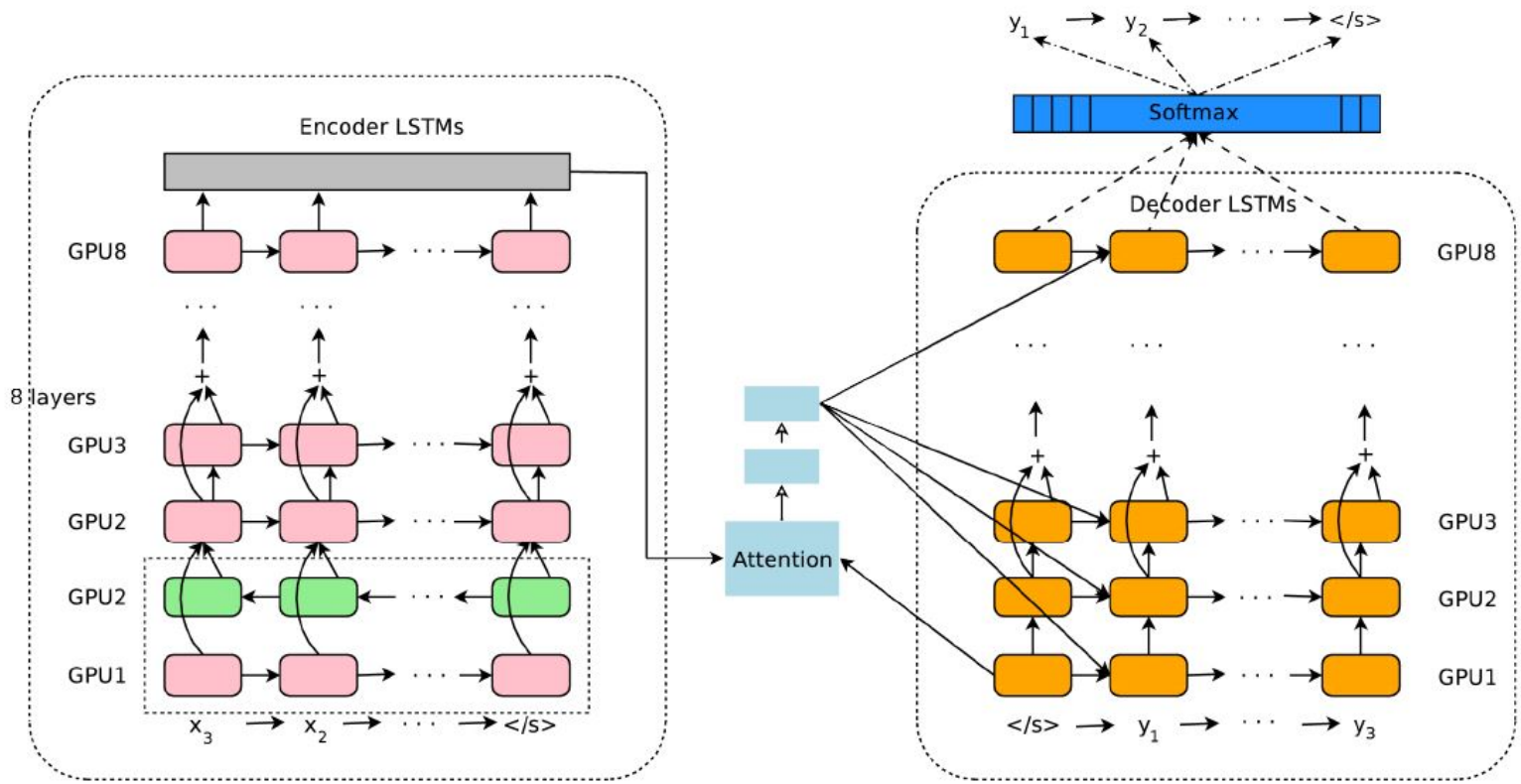


Res-Net



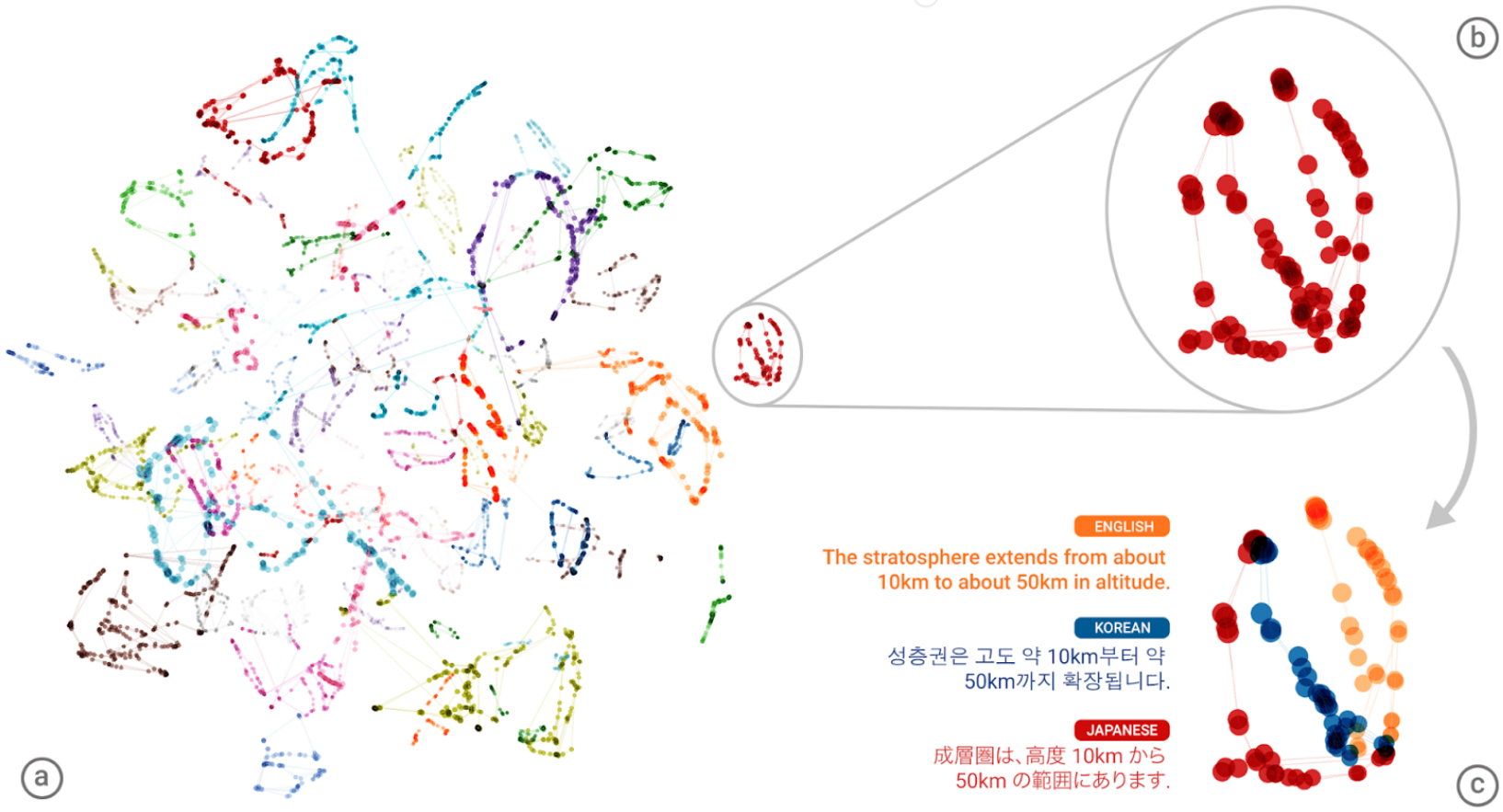
He, Zhang, Ren & Sun (2015)

Google's Neural Machine Translation System



Wu et al. (2016): Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation

Language Embeddings



Google Research Blog

Bibliography