Sequential Data Processing (RNNs, LSTMs)

Machine Learning Decal

Hosted by Machine Learning at Berkeley

Agenda

Motivation

Recurrent Neural Networks (RNNs)

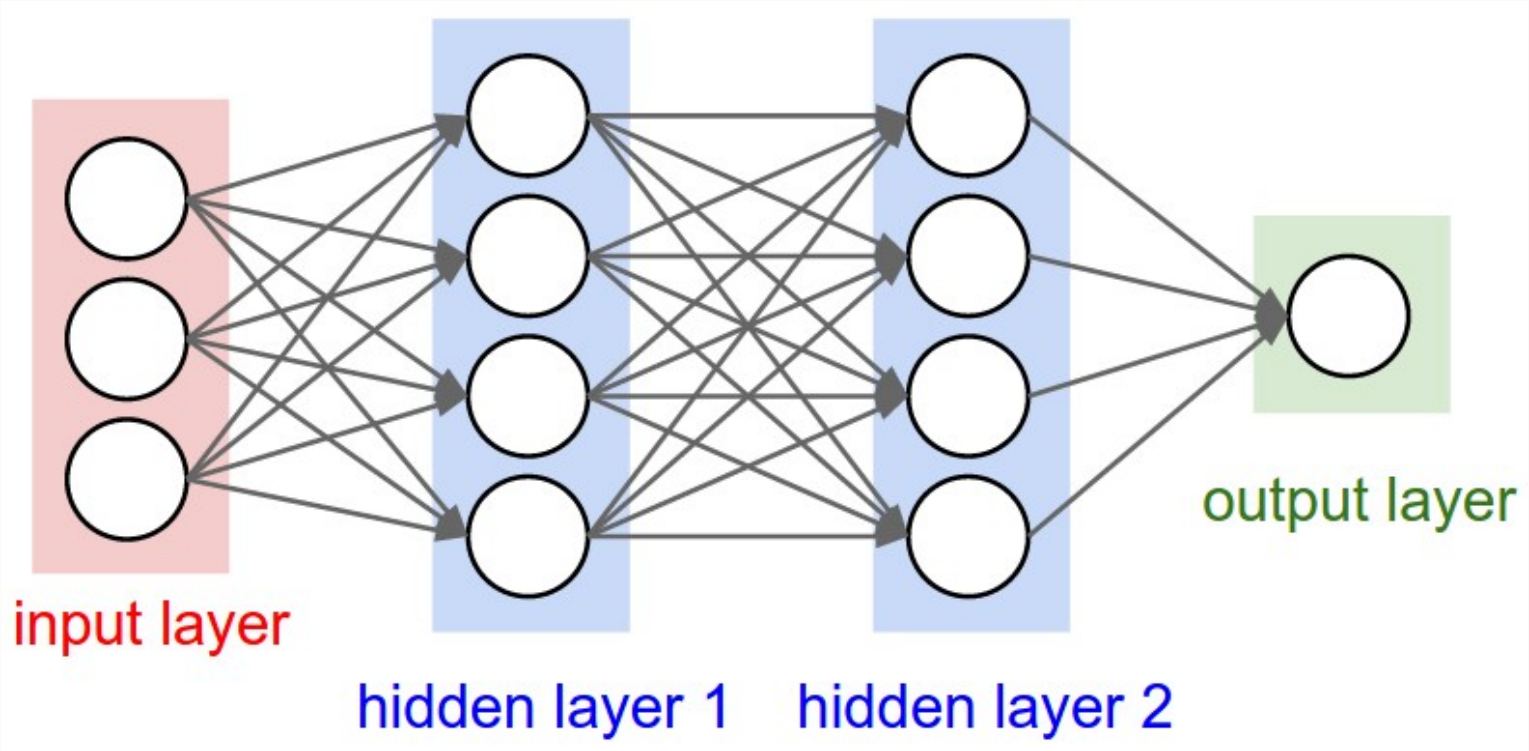
RNN Demo

Long Short-Term Memory Networks (LSTMs)

LSTM Demo

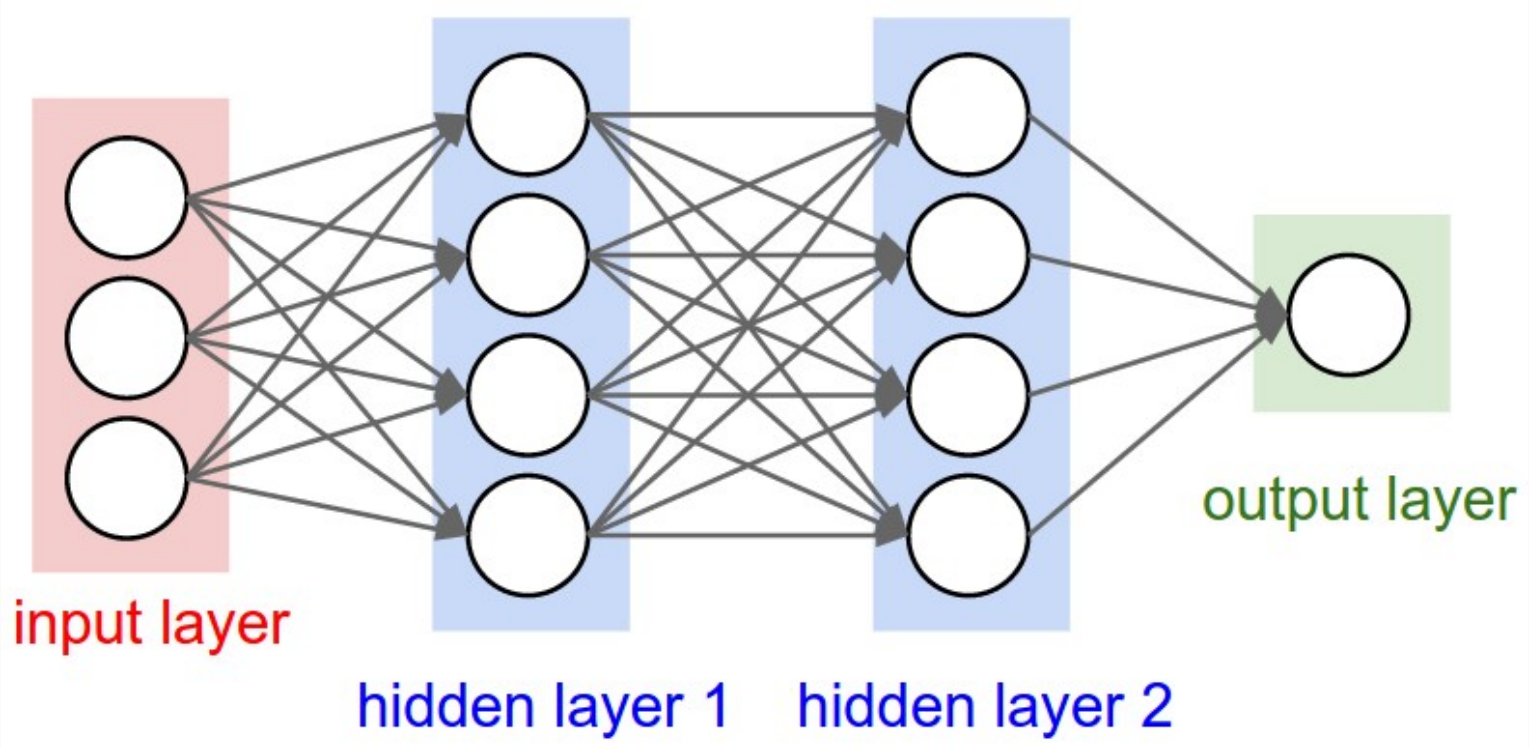
Motivation

Review: Fully-Connected (Dense) Networks



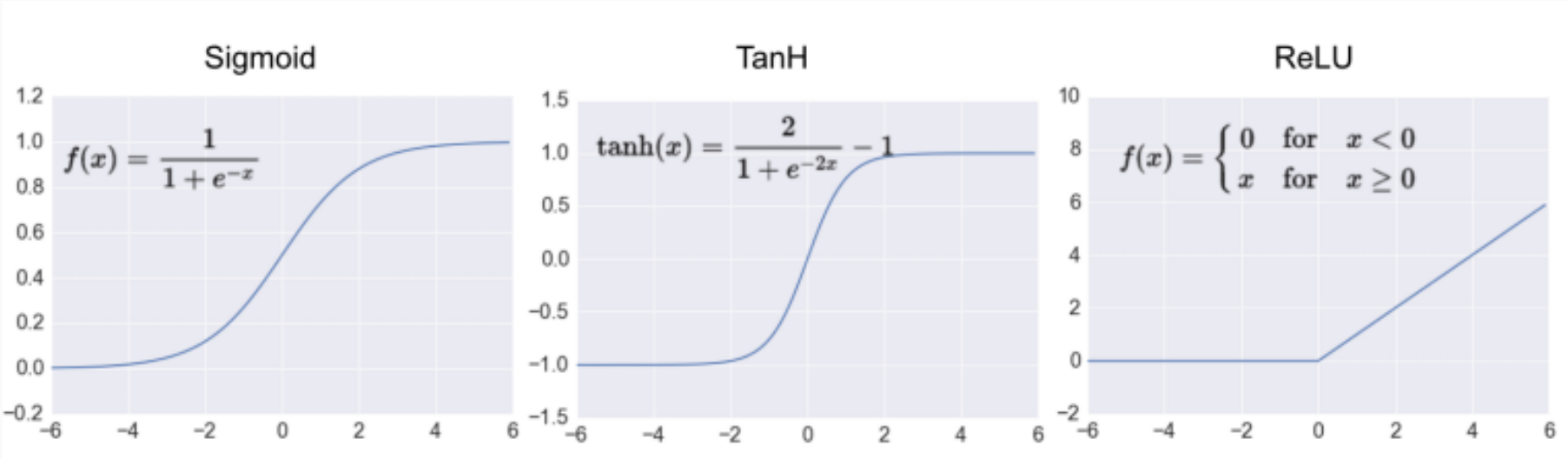
$$a_i^t = \sigma(w_1 a_1^{t-1} + w_2 a_2^{t-1} + \cdots + w_k a_k^{t-1} + b_t)$$

Review: Fully-Connected (Dense) Networks



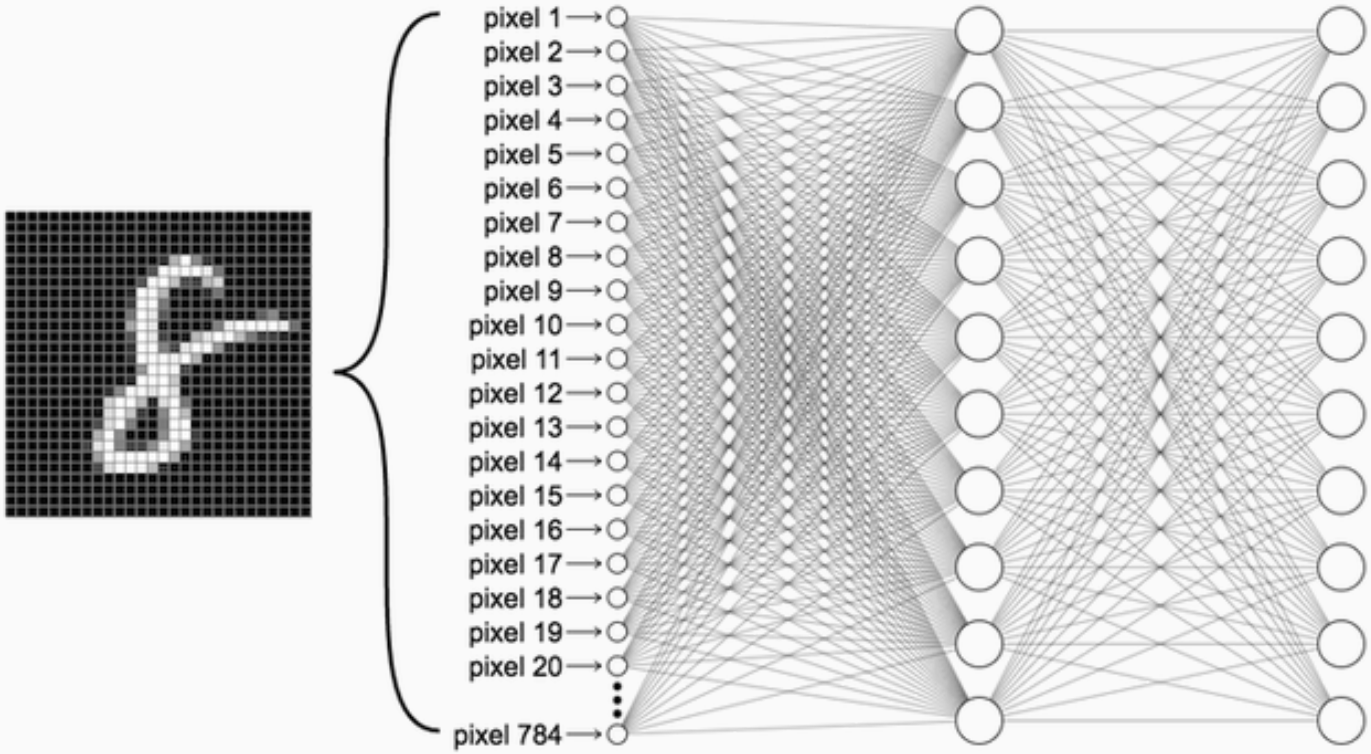
$$a^{(t)} = \sigma\left(W^{(t)}a^{(t-1)} + b^{(t)}\right)$$

Review: Fully-Connected (Dense) Networks



Examples of activation functions.

Review: Fully-Connected (Dense) Networks



Review: Fully-Connected (Dense) Networks

Remember: Dense neural networks are *universal function approximators*, meaning that any function from $\mathbb{R}^n$ to $\mathbb{R}^m$ can be represented *arbitrarily well* by some neural network.

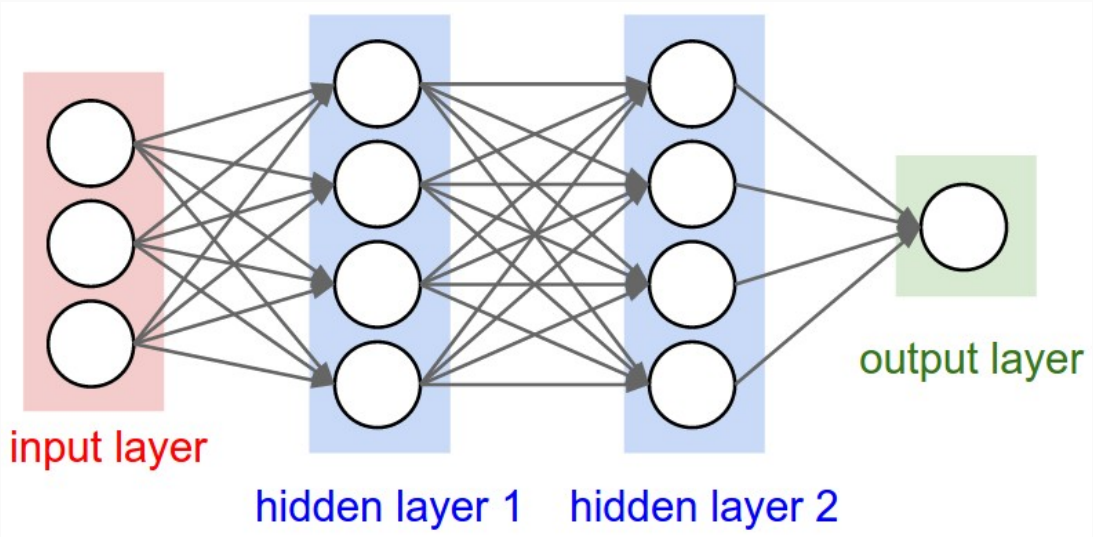
Review: Fully-Connected (Dense) Networks

What types of problems can dense neural networks *not* solve, then?

Motivation: Shortcomings of Dense Neural Networks

Neural networks cannot solve time-series problems!

- No notion of 'order' of the inputs.
- No support for variable-length sequences.



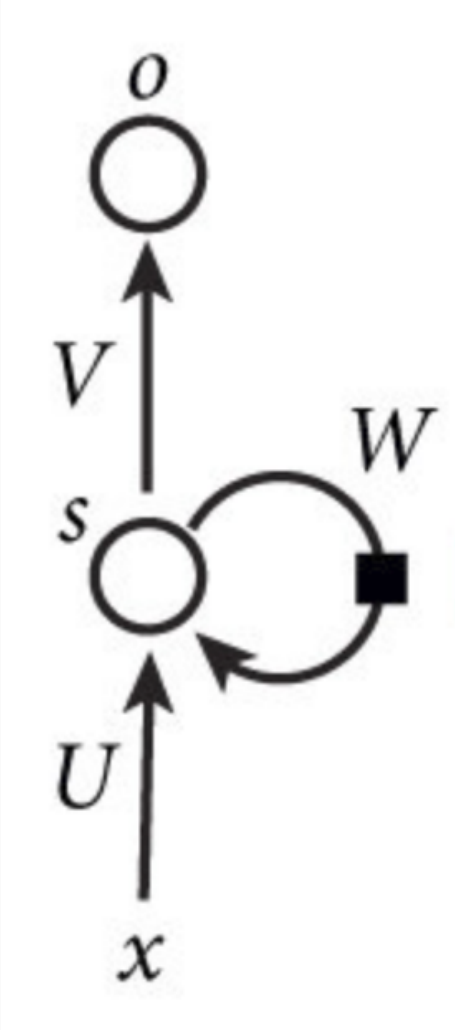
Applications: Videos

Applications: Videos

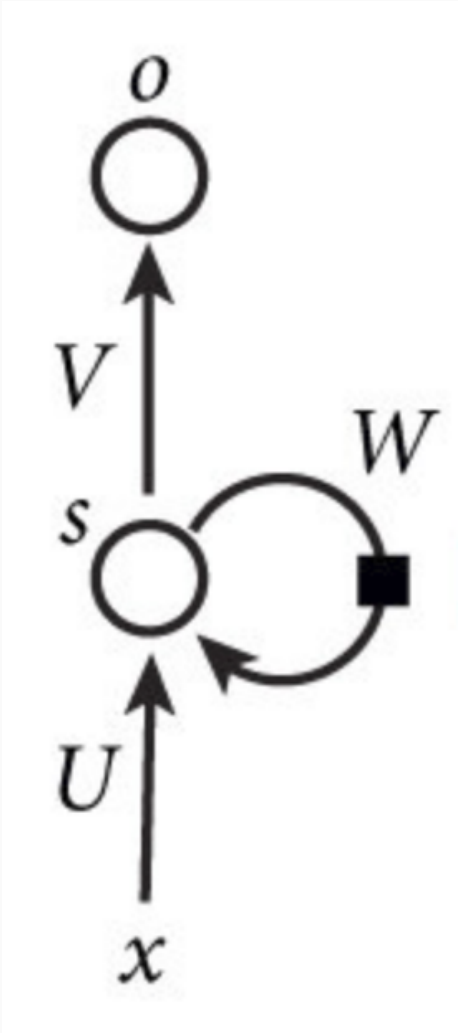
Applications: Language

Recurrent Neural Networks (RNNs)

RNNs: Internal States



RNNs: Update Equations



$$h^{(t)} = \tanh\left(b + Wh^{(t-1)} + Ux^{(t)}\right)$$

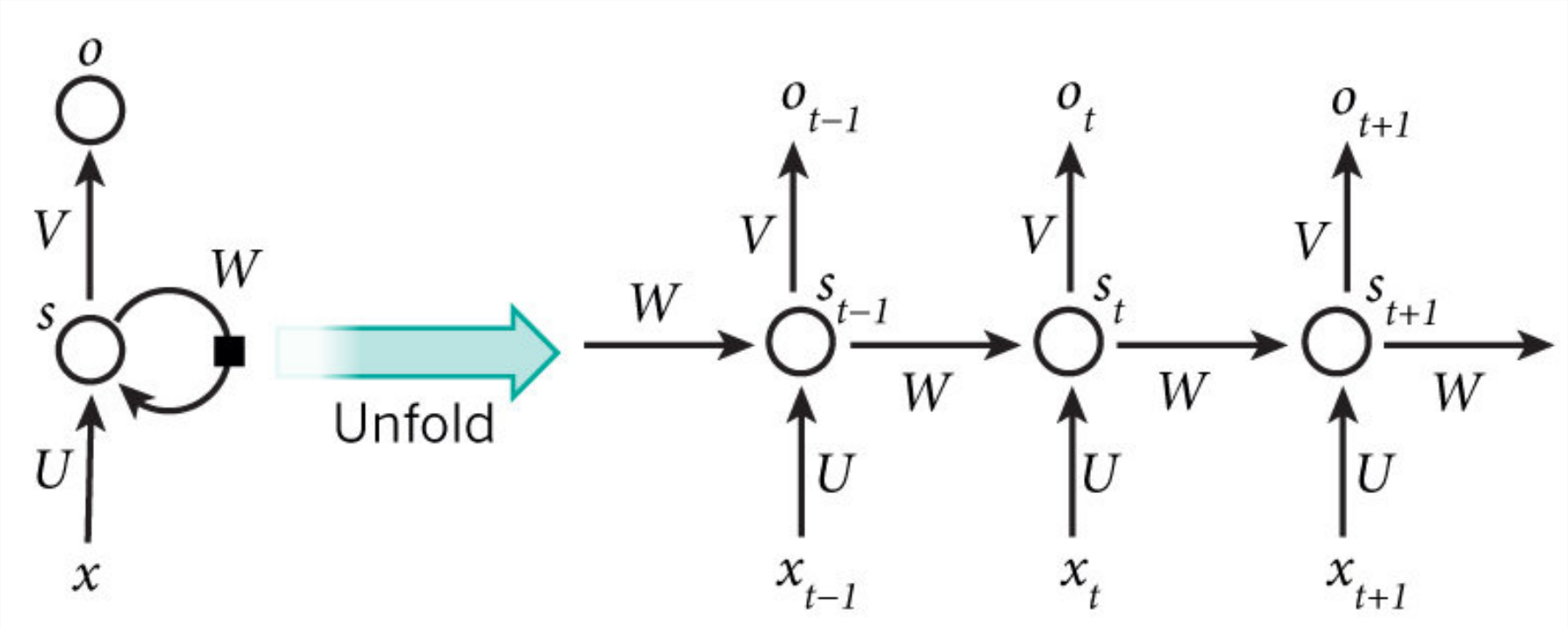
$$o^{(t)} = c + Vh^{(t)}$$

RNNs: Backprop?

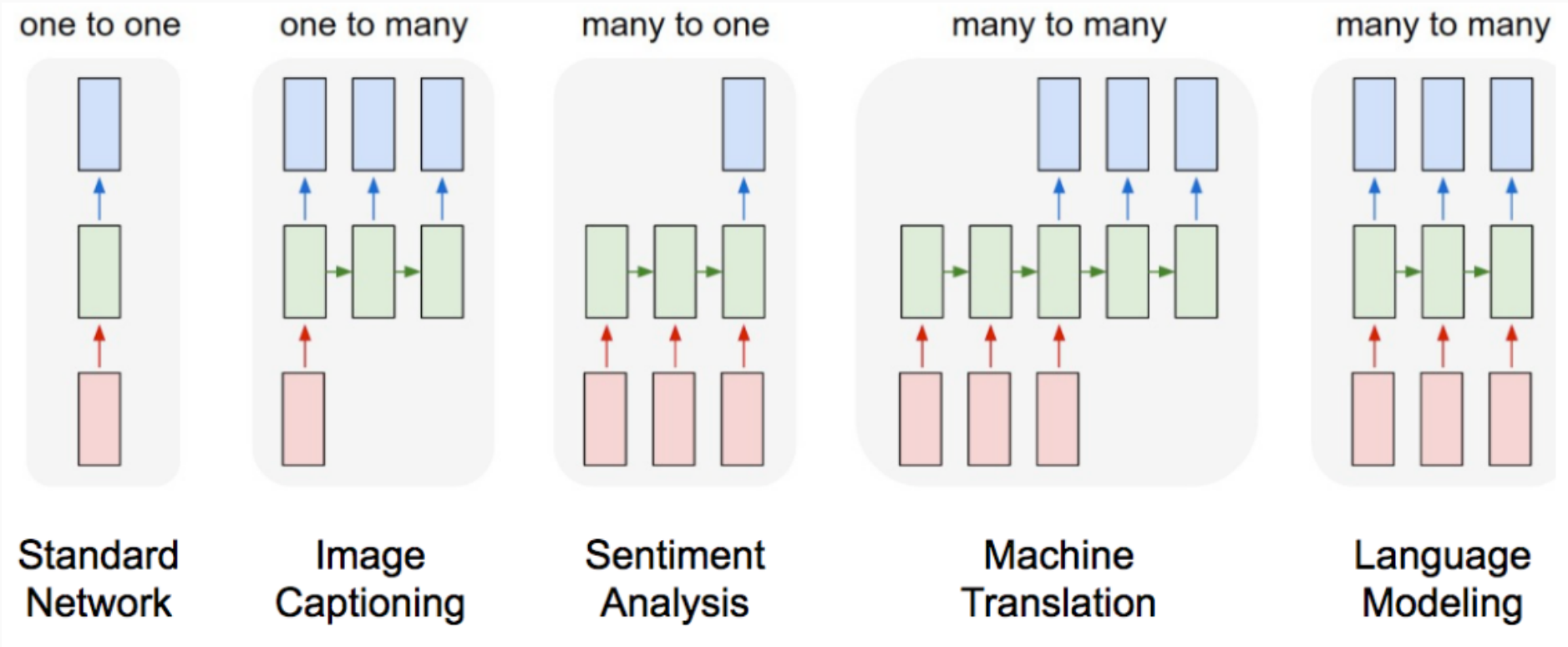
Remember that backpropagation requires no cycles in the computation graph (i.e., a DAG).

Does backprop work for networks with recurrence?

RNNs: Unrolling the Graph



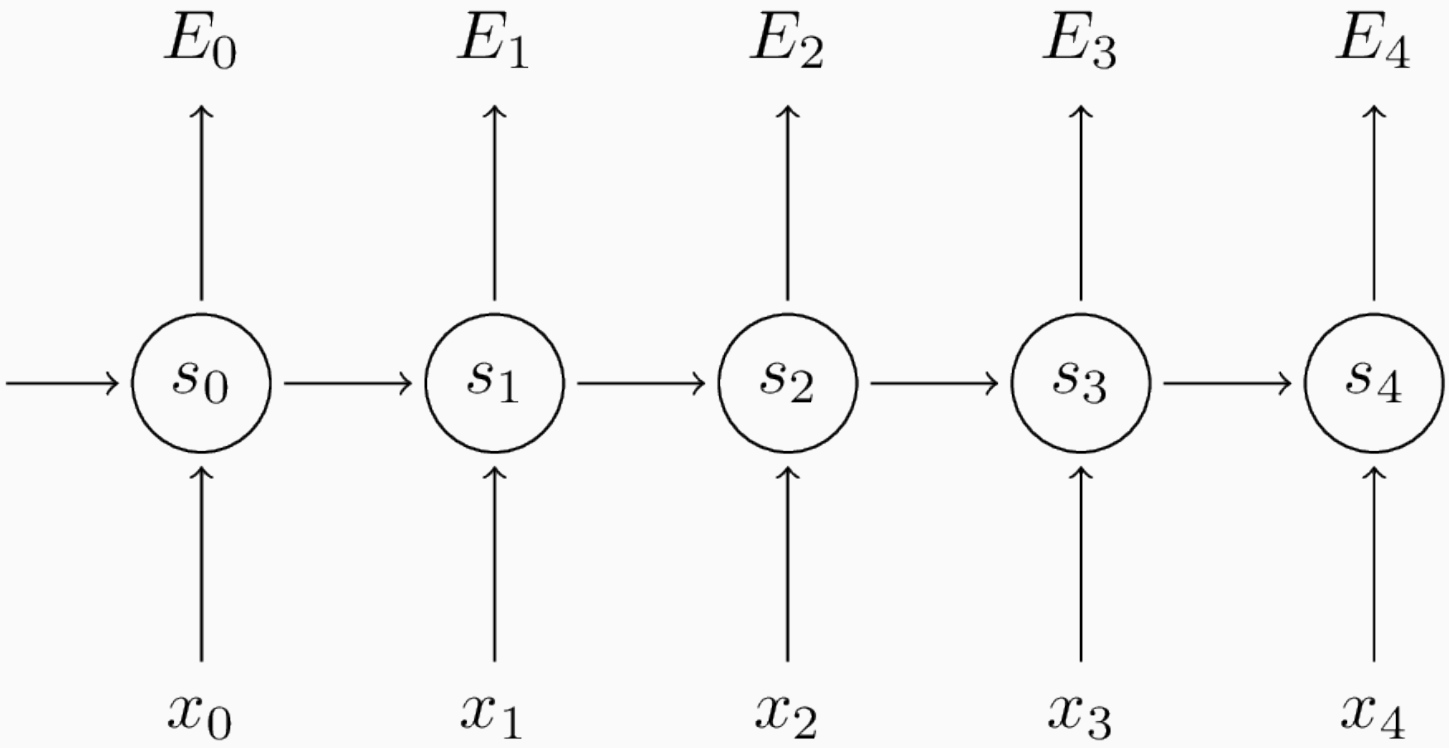
RNNs: Uses


one to one
one to many
many to one
many to many
many to many
Standard Network
Image Captioning
Sentiment Analysis
Machine Translation
Language Modeling

RNNs: Sequence Modeling

Sequence modeling is the task of predicting the "next item" given the current token.
(e.g., Bitcoin price prediction, language modeling).

RNNs: Considerations for Sequence Modeling

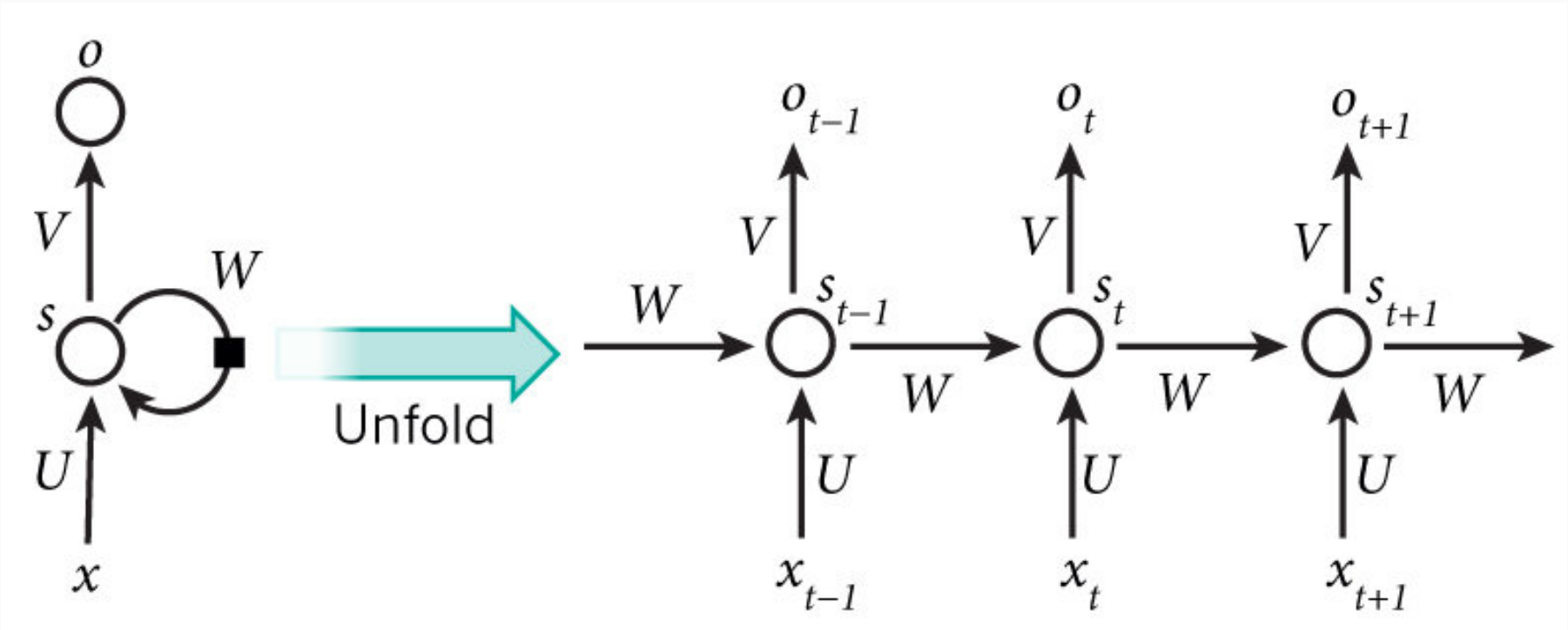


Teacher Forcing: Use the predicted value or the ground-truth value for input?

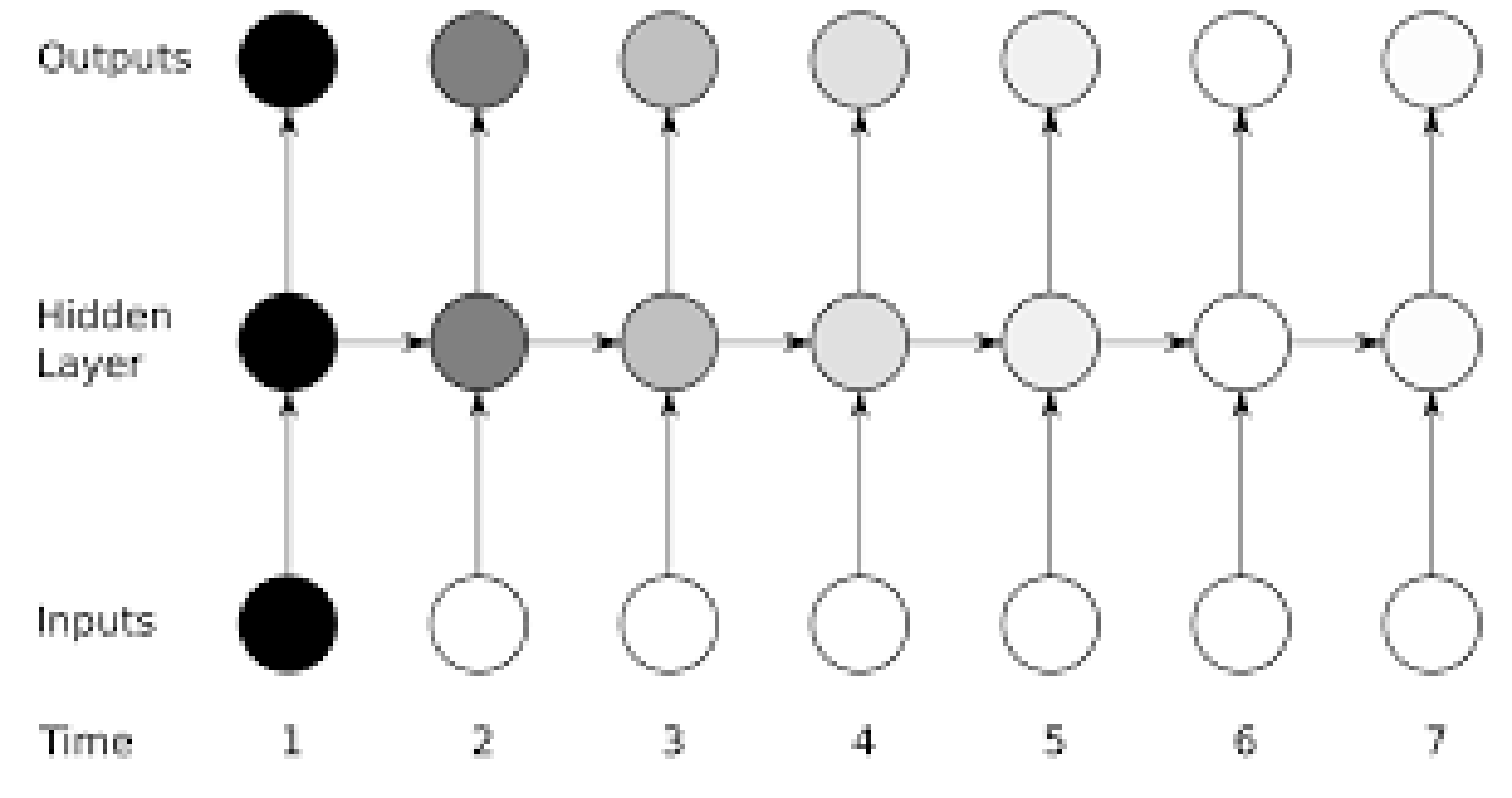
RNN Demo

Long Short-Term Memory Networks (LSTMs)

LSTMs: Vanishing Gradient Problem

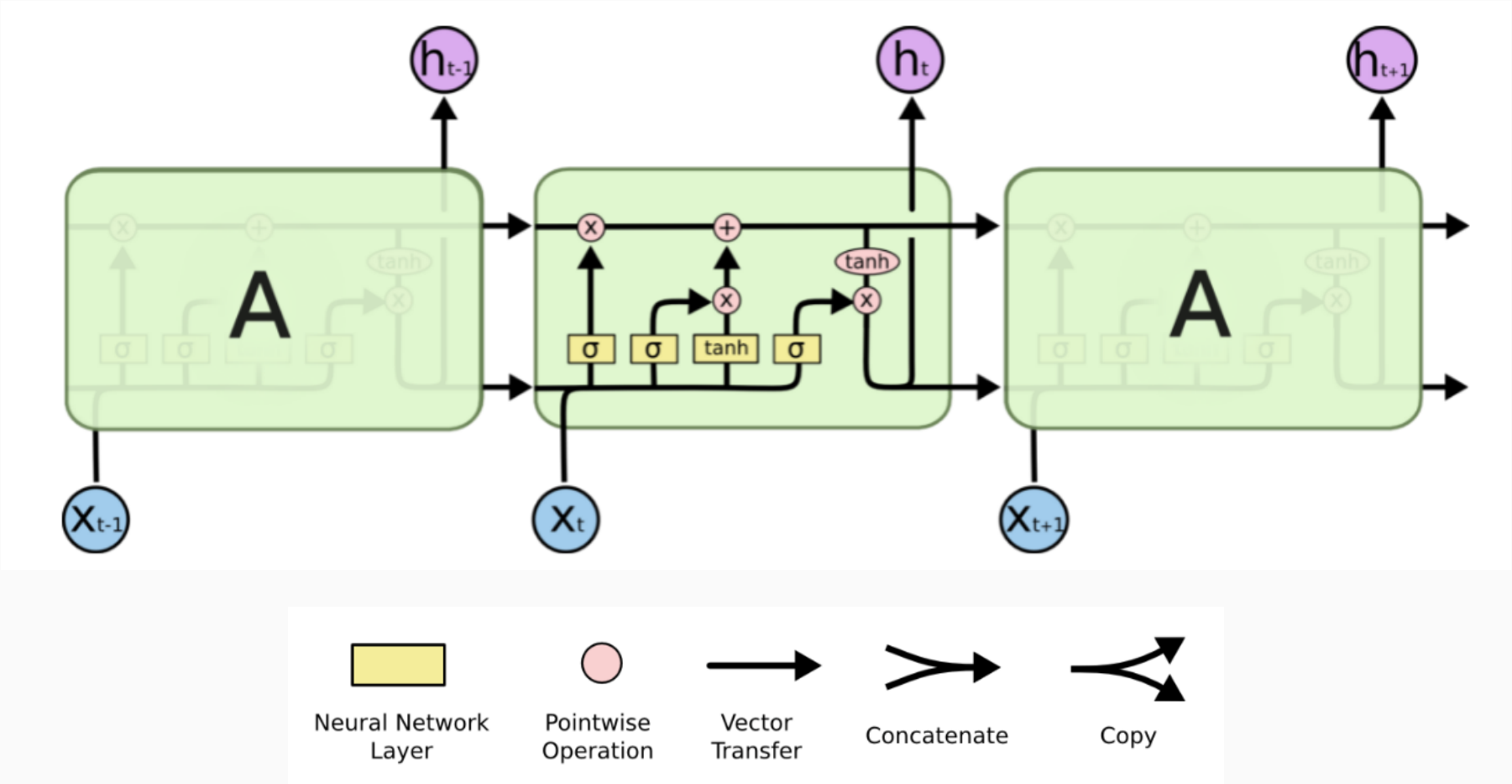


LSTMs: Vanishing Gradient Problem



As we change the input weight matrix U, signal from earlier timesteps exponentially decays through time.

LSTMs: A Solution to Vanishing Gradient



LSTMs: A Solution to Vanishing Gradient

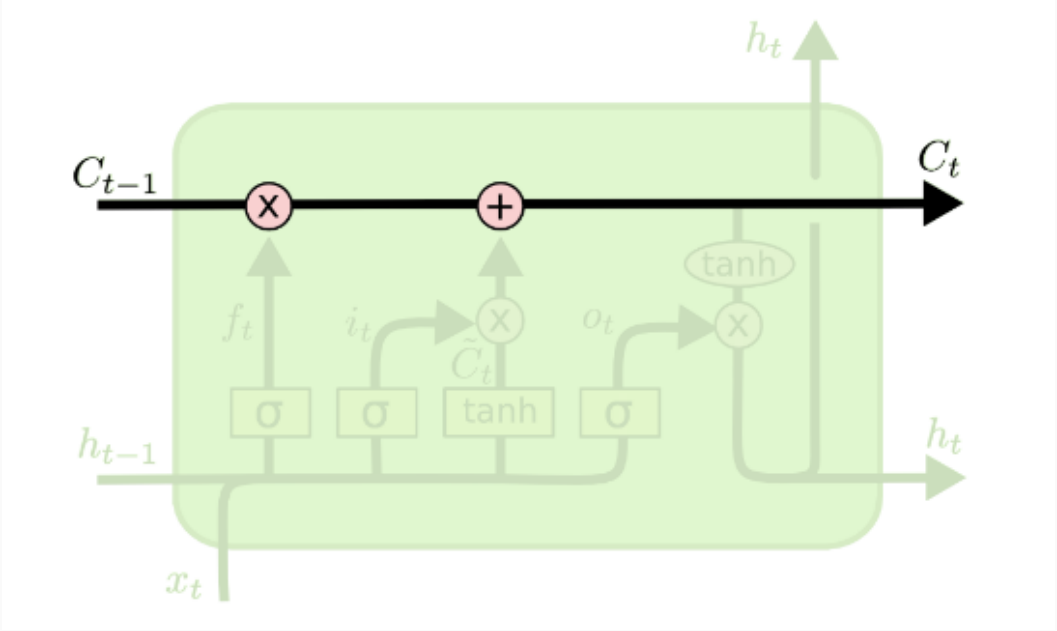
$$
\begin{aligned}
f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\
i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\
o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\
c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\
h_t &= o_t \circ \sigma_h(c_t)
\end{aligned}
$$

LSTMs: A Solution to Vanishing Gradient

LSTMs contain four parts:

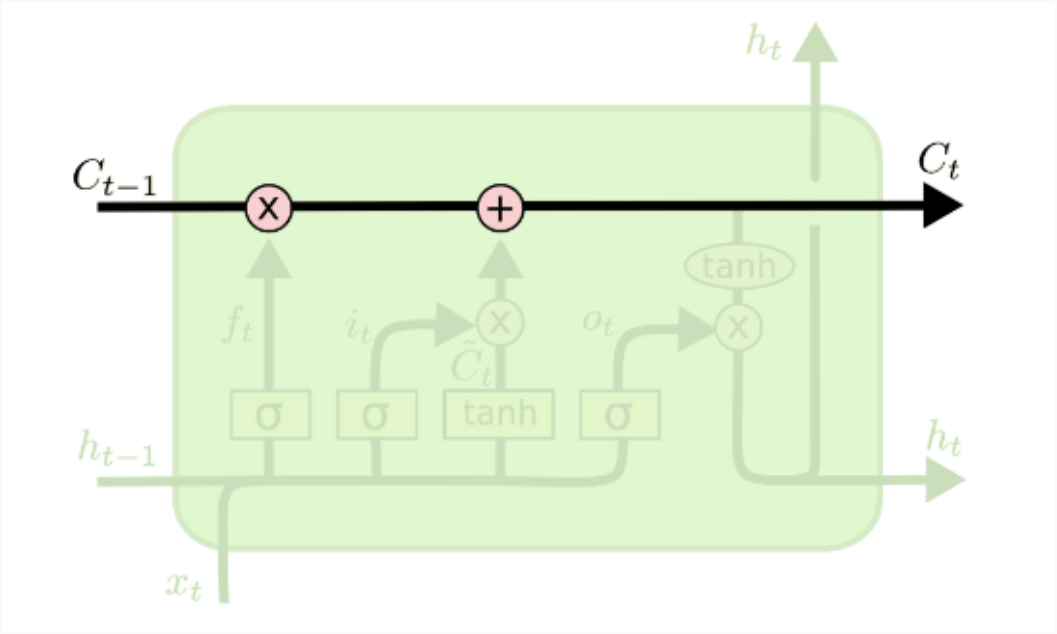
- A cell state.
- A forget gate.
- An input gate.
- An output gate.

LSTMs: What's different?



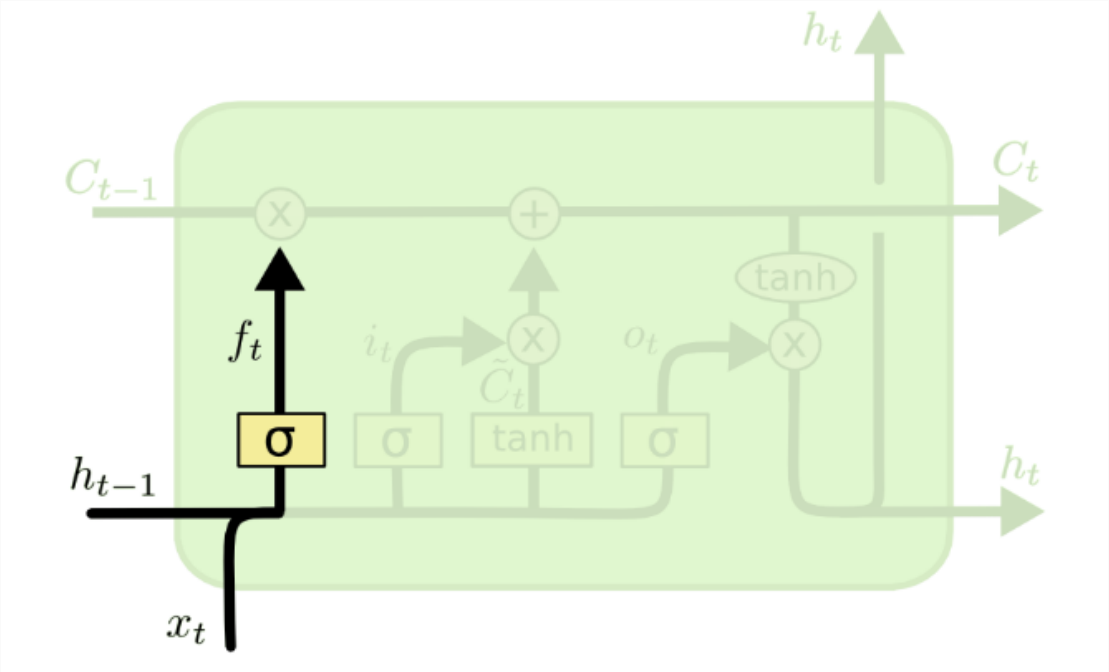
Why does the LSTM avoid the vanishing gradient issue?

LSTMs: What's different?



A single, continuous 'information highway' across the entire calculation that does not have any matrix multiplications!

LSTMs: Understanding the Gates



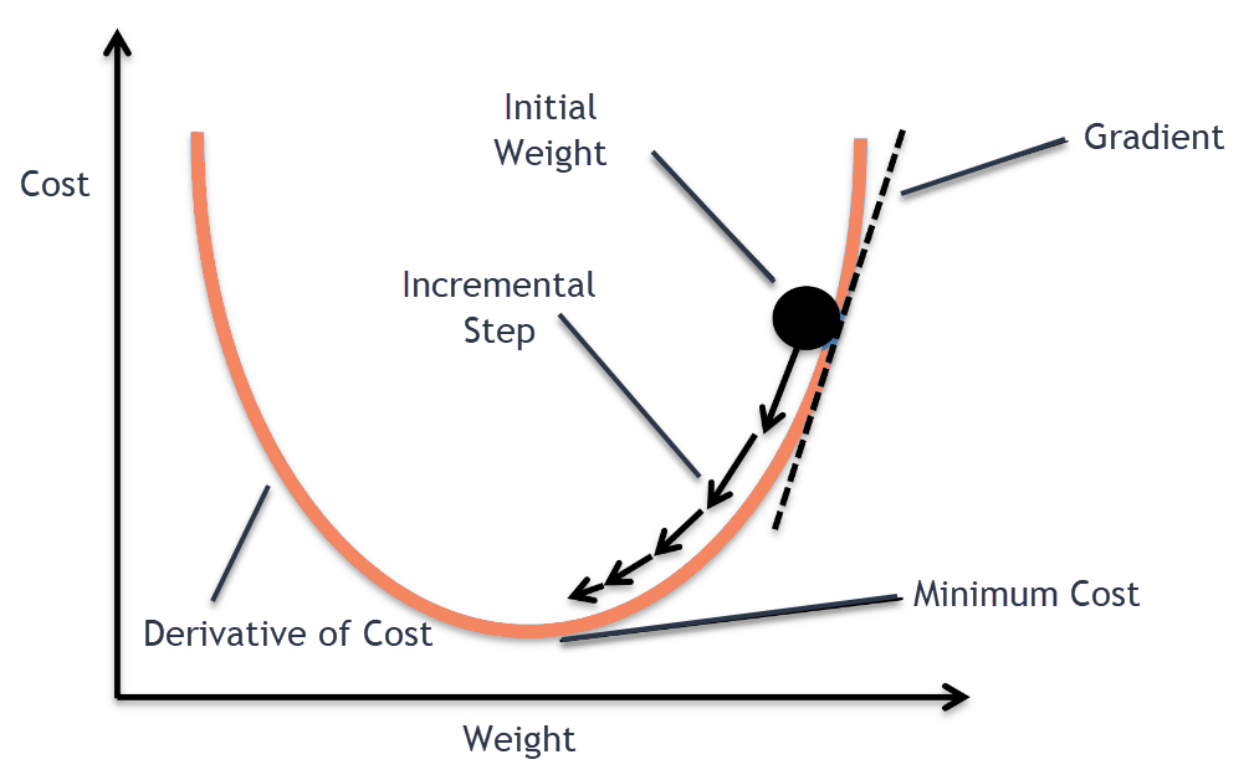
$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f)$$

The Forget Gate

LSTMs: Optimization Methods

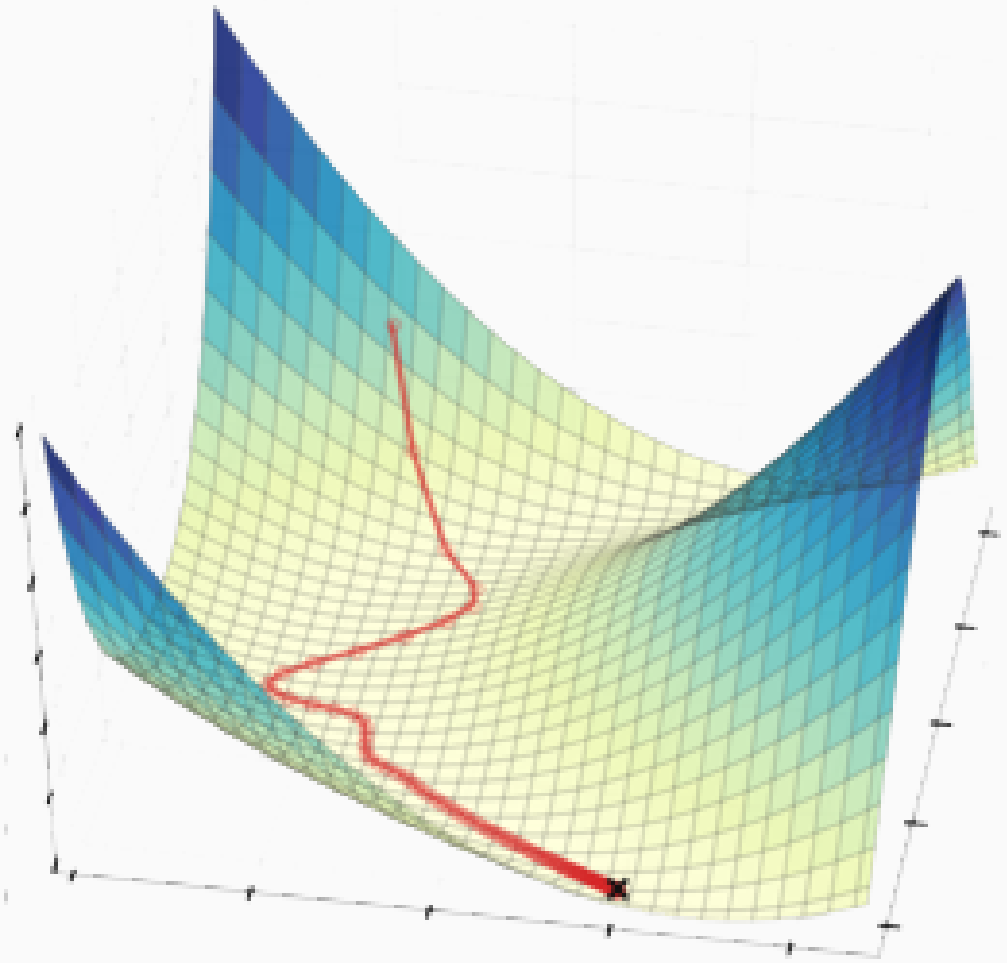
LSTMs can be difficult to train because they map infinite-dimensional spaces to infinite-dimensional spaces.

LSTMs: Optimization Methods



Vanilla Gradient Descent

LSTMs: Optimization Methods



Vanilla Gradient Descent

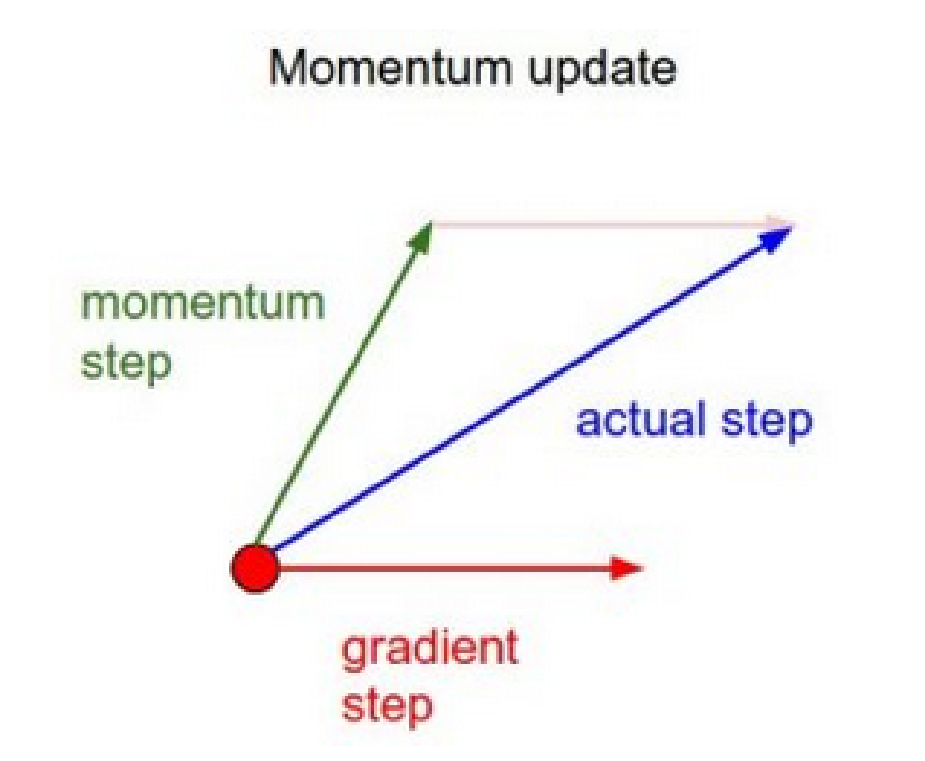
LSTMs: Optimization Methods

Basic gradient descent only considers the current location, and is thus sensitive to sharp changes in the loss surface.

LSTMs: Optimization Methods

Solution: Gradient Descent with Momentum!

LSTMs: Optimization Methods



$$z^{(t)} = \beta z^{(t-1)} + (1-\beta)\nabla_\theta C(\theta; X, y)$$

$$w^{(t)} = w^{(t-1)} - \alpha z^{(t)}$$

LSTMs: Optimization Methods

Benefits of SGD with Momentum:

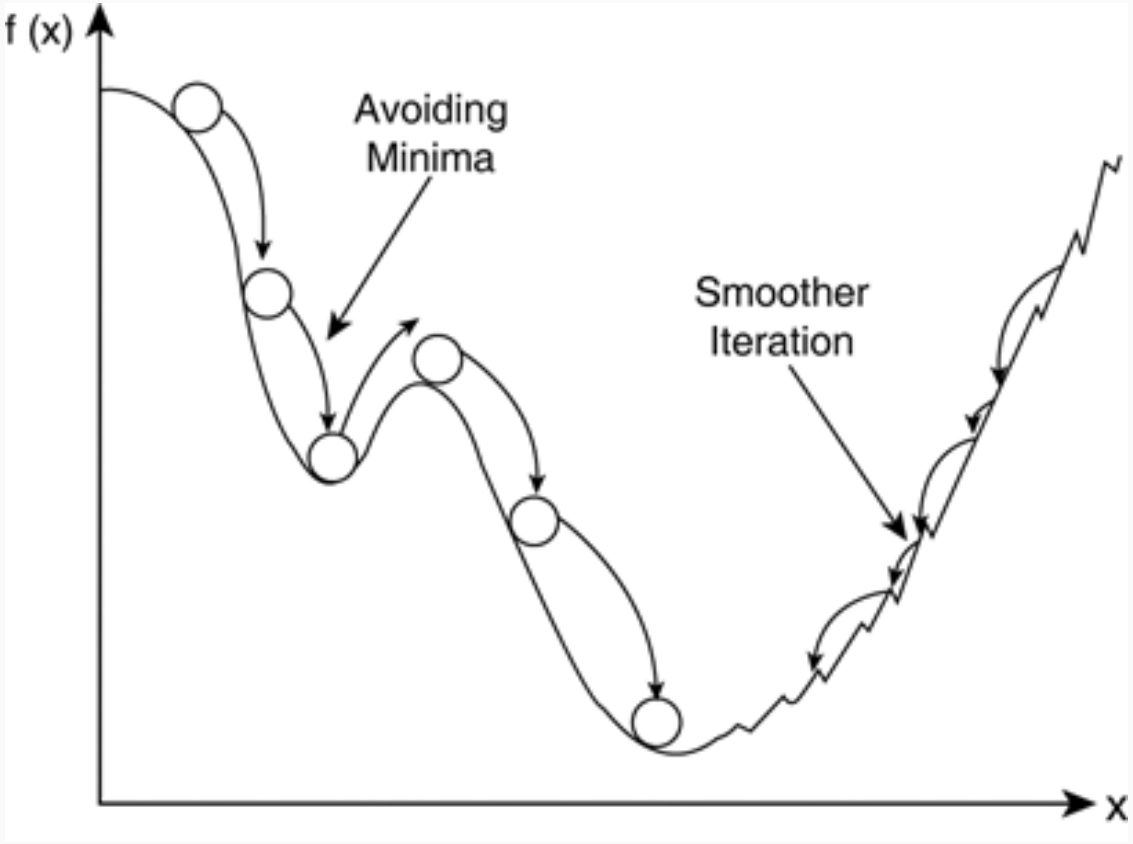
- Smoother iteration to reduce sensitivity to steep loss surfaces.
- ???

LSTMs: Optimization Methods

Benefits of SGD with Momentum:

- Smoother iteration to reduce sensitivity to steep loss surfaces.
- Ability to jump out of local minima.

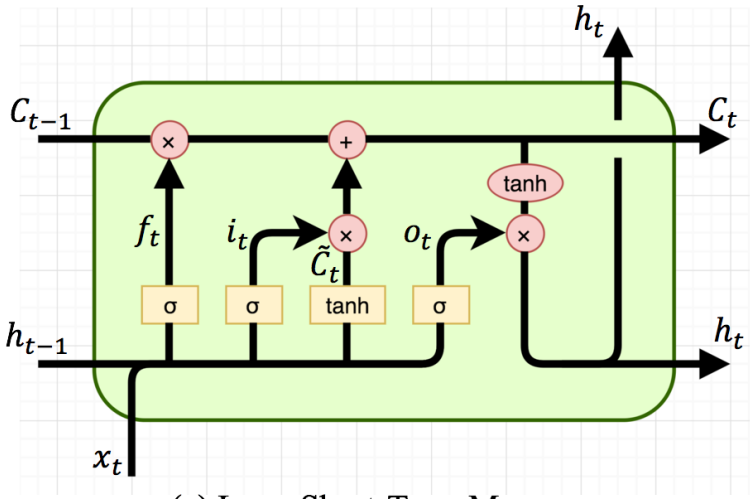
LSTMs: Optimization Methods



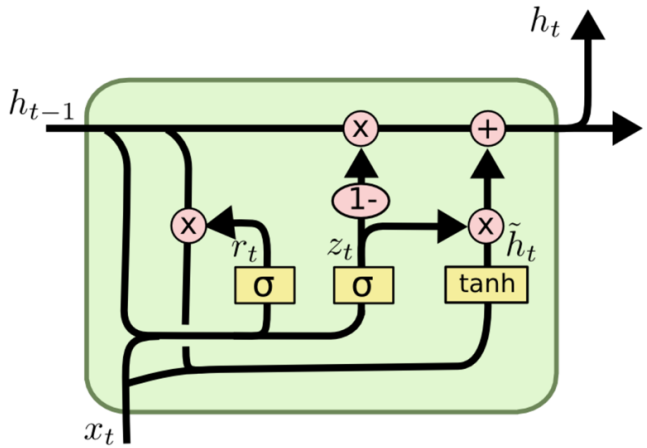
LSTMs: Optimization Methods

Variants include RMSProp, Adagrad, Adadelta, Nesterov Momentum, Adam, etc.

GRUs: An Extension of LSTMs



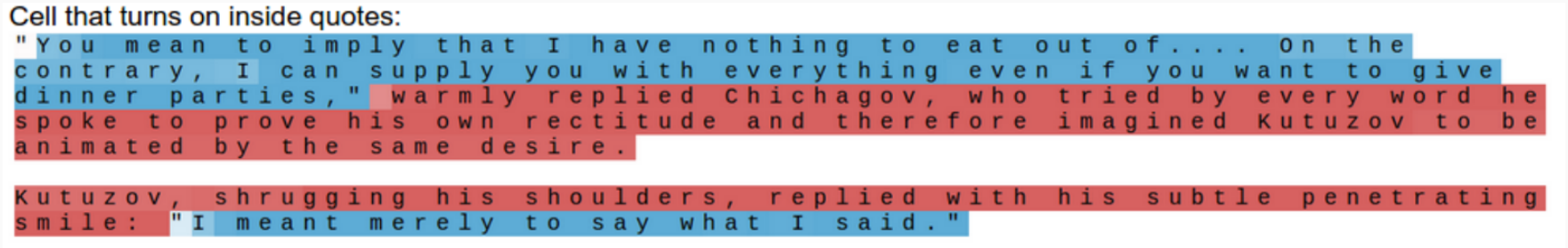
(a) Long Short-Term Memory



(b) Gated Recurrent Unit

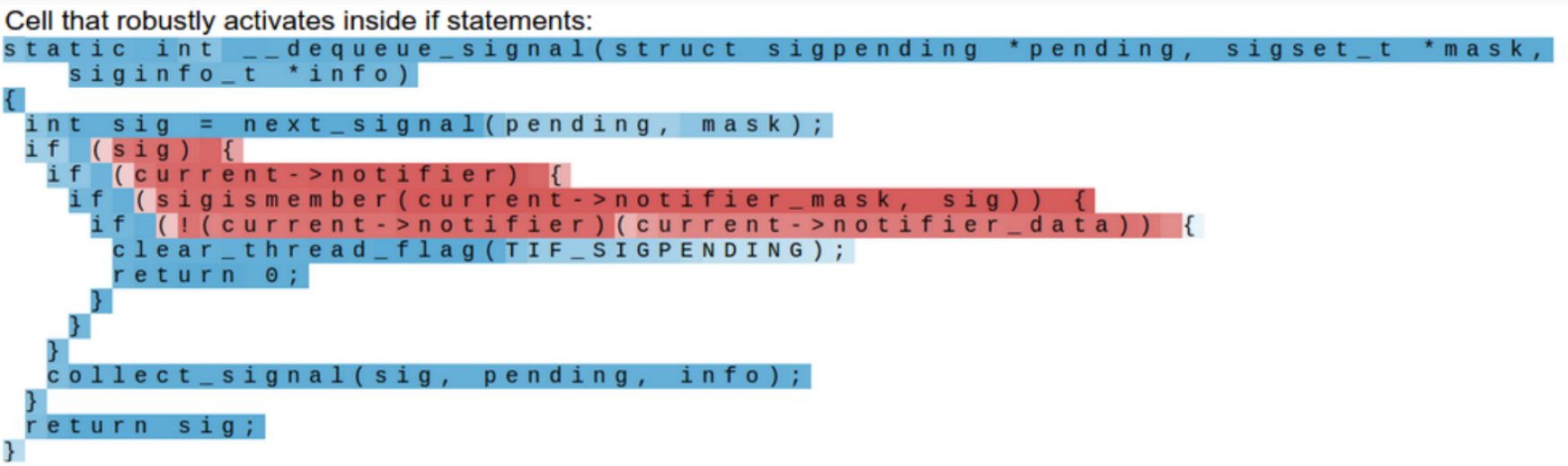
LSTM Demo

Analyzing LSTMs

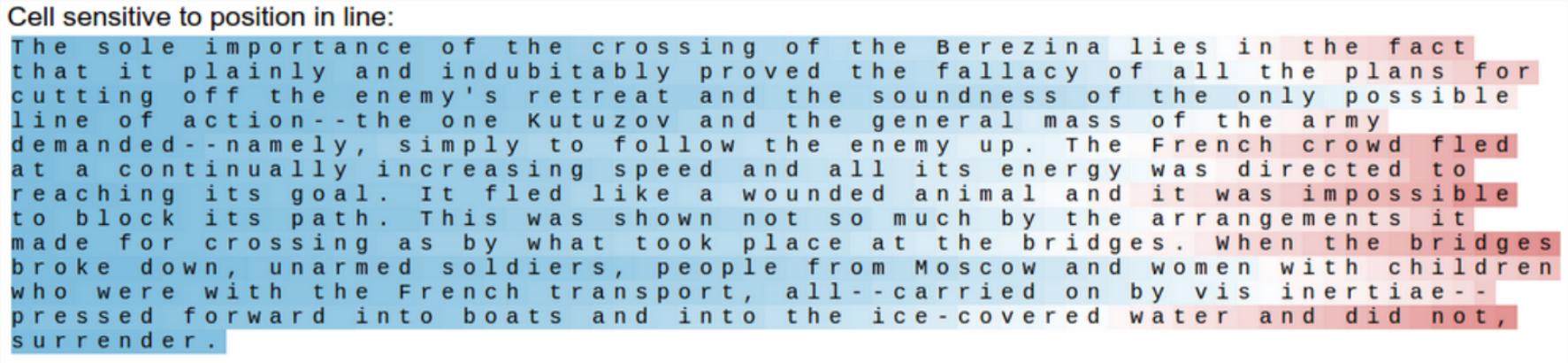


LSTMs keep track of interesting information!

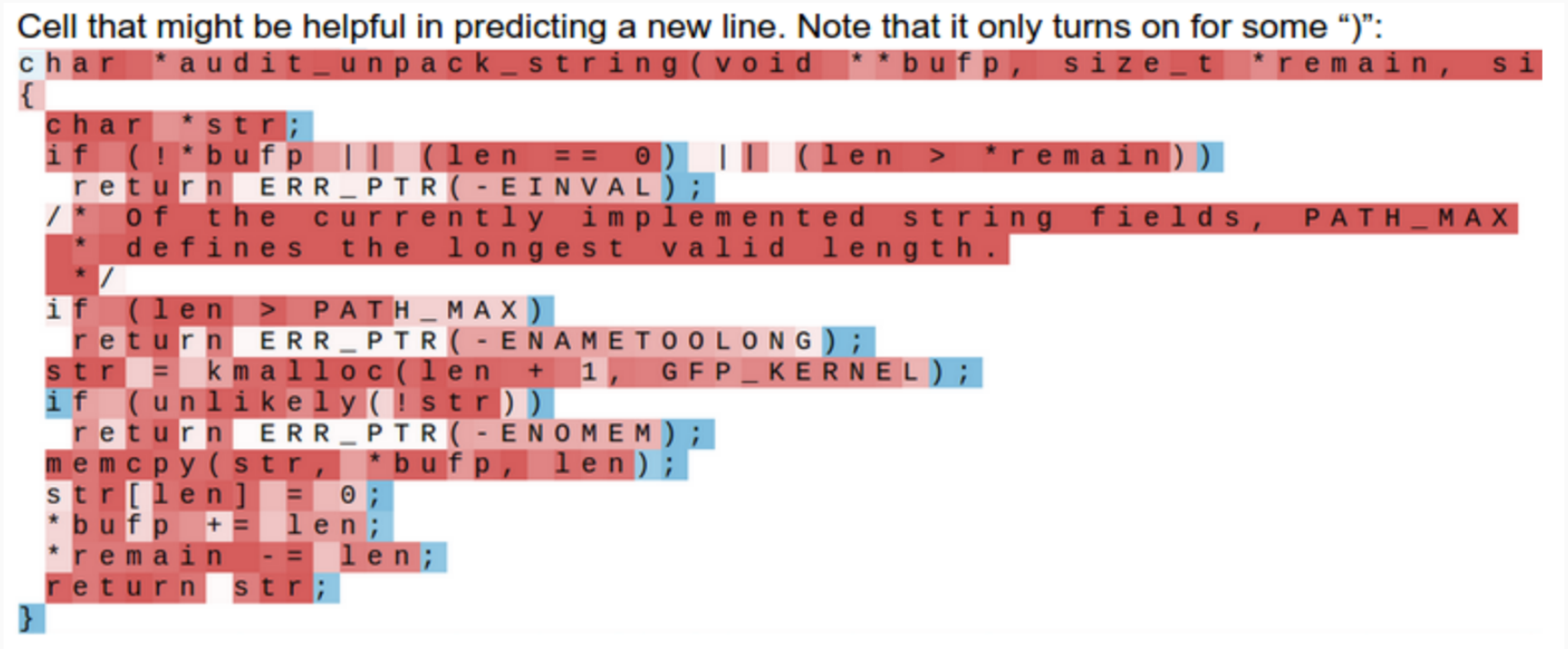
Analyzing LSTMs



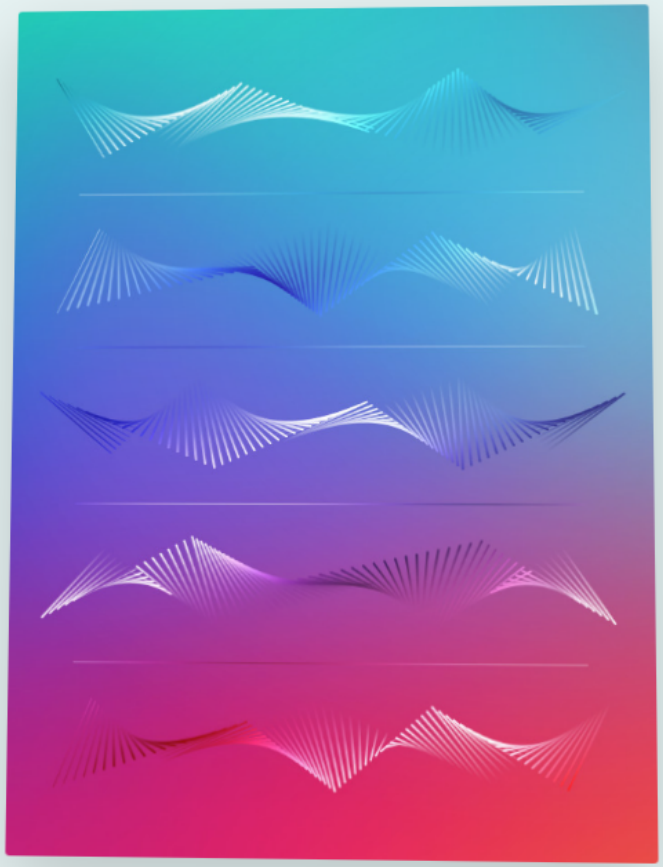
Analyzing LSTMs



Analyzing LSTMs



Analyzing LSTMs



APRIL 6, 2017

Unsupervised Sentiment Neuron

We've developed an unsupervised system which learns an excellent representation of sentiment, despite being trained only to predict the next character in the text of Amazon reviews.

A linear model using this representation achieves state-of-the-art sentiment analysis accuracy on a small but extensively-studied dataset, the Stanford Sentiment Treebank (we get 91.8% accuracy versus the previous best of 90.2%), and can match the performance of previous supervised systems using 30-100x fewer labeled examples. Our representation also contains a distinct "sentiment neuron" which contains almost all of the sentiment signal.

VIEW ON GITHUB

VIEW ON ARXIV

READ MORE