Sequences in Caffe

Jeff Donahue
CVPR Caffe Tutorial
June 6, 2015

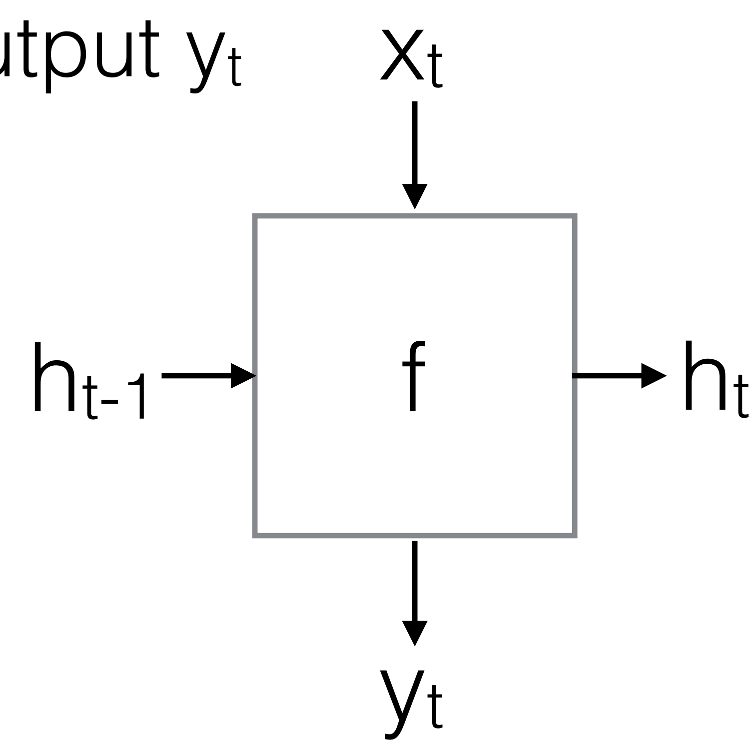
Sequence Learning

- Instances of the form $\mathbf{x} = <x_1, x_2, x_3, \ldots, x_T>$
- Variable sequence length T
- Learn a transition function f with parameters W:
- f should update hidden state h_t and output y_t

$h_0 := 0$

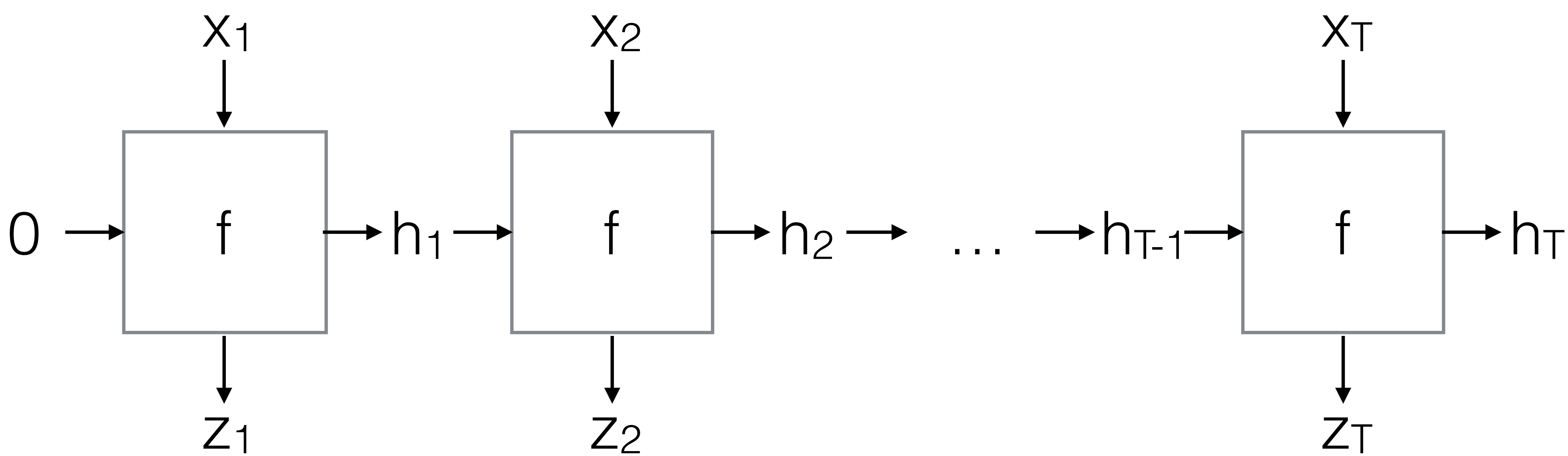
for $t = 1, 2, 3, \ldots, T$:

$<y_t, h_t> = f_W(x_t, h_{t-1})$



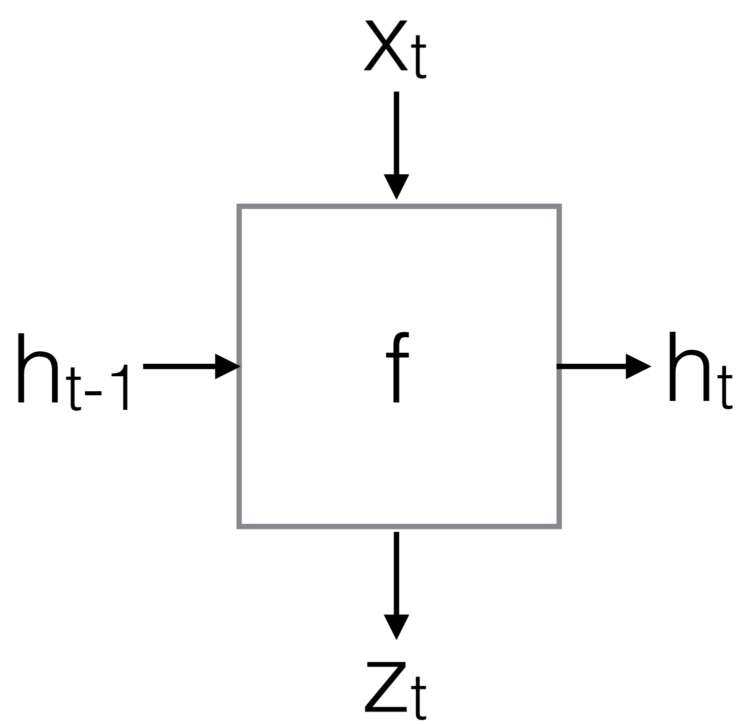
Sequence Learning

Equivalent to a T-layer deep network, unrolled in time



Sequence Learning

- What should the transition function f be?



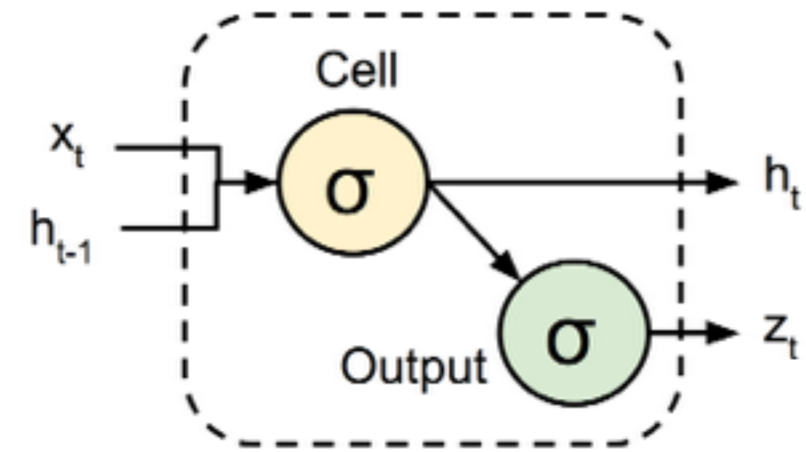
- At a minimum, we want something **non-linear** and **differentiable**

Sequence Learning

- A "vanilla" RNN:

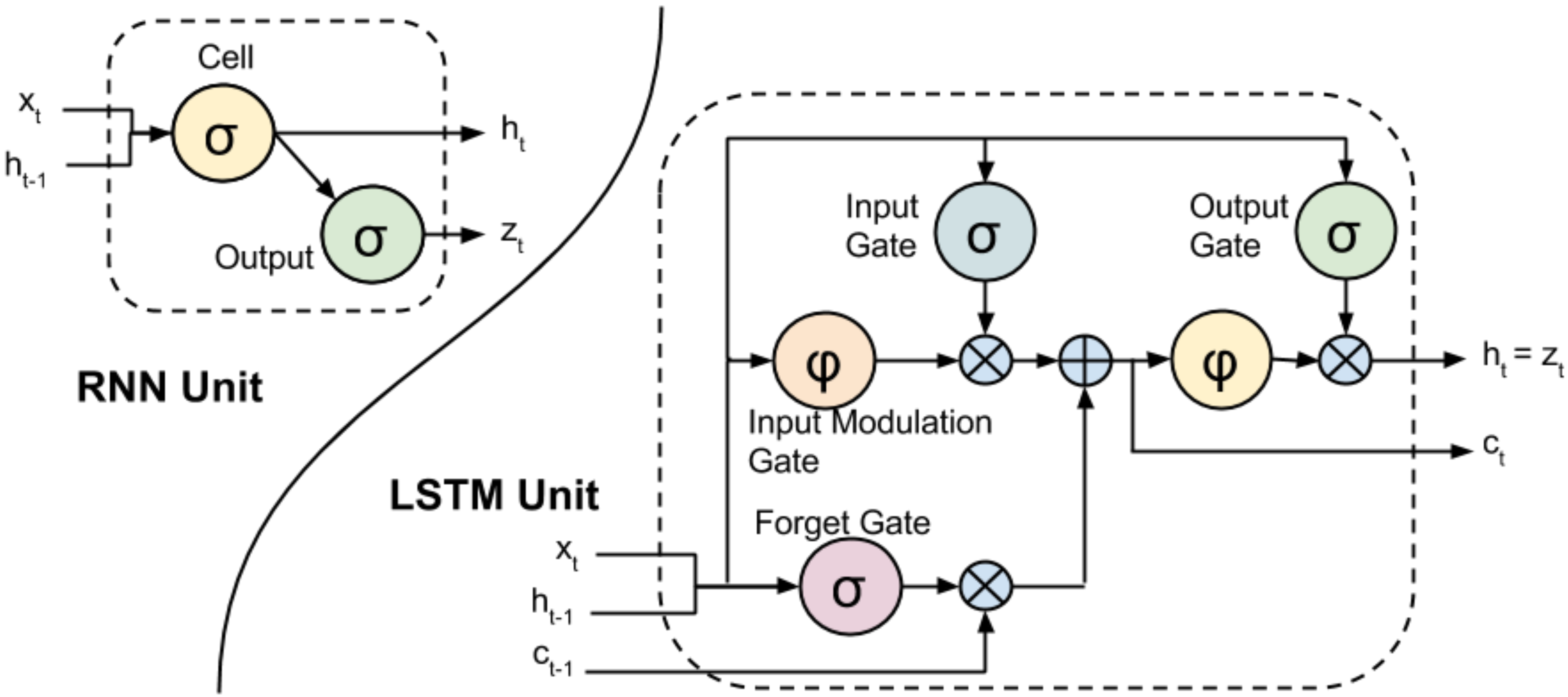
$$h_t = \sigma(W_{hx}x_t + W_{hh}h_{t-1} + b_h)$$

$$z_t = \sigma(W_{hz}h_t + b_z)$$



- Problems
  - Difficult to train — vanishing/exploding gradients
  - Unable to "select" inputs, hidden state, outputs

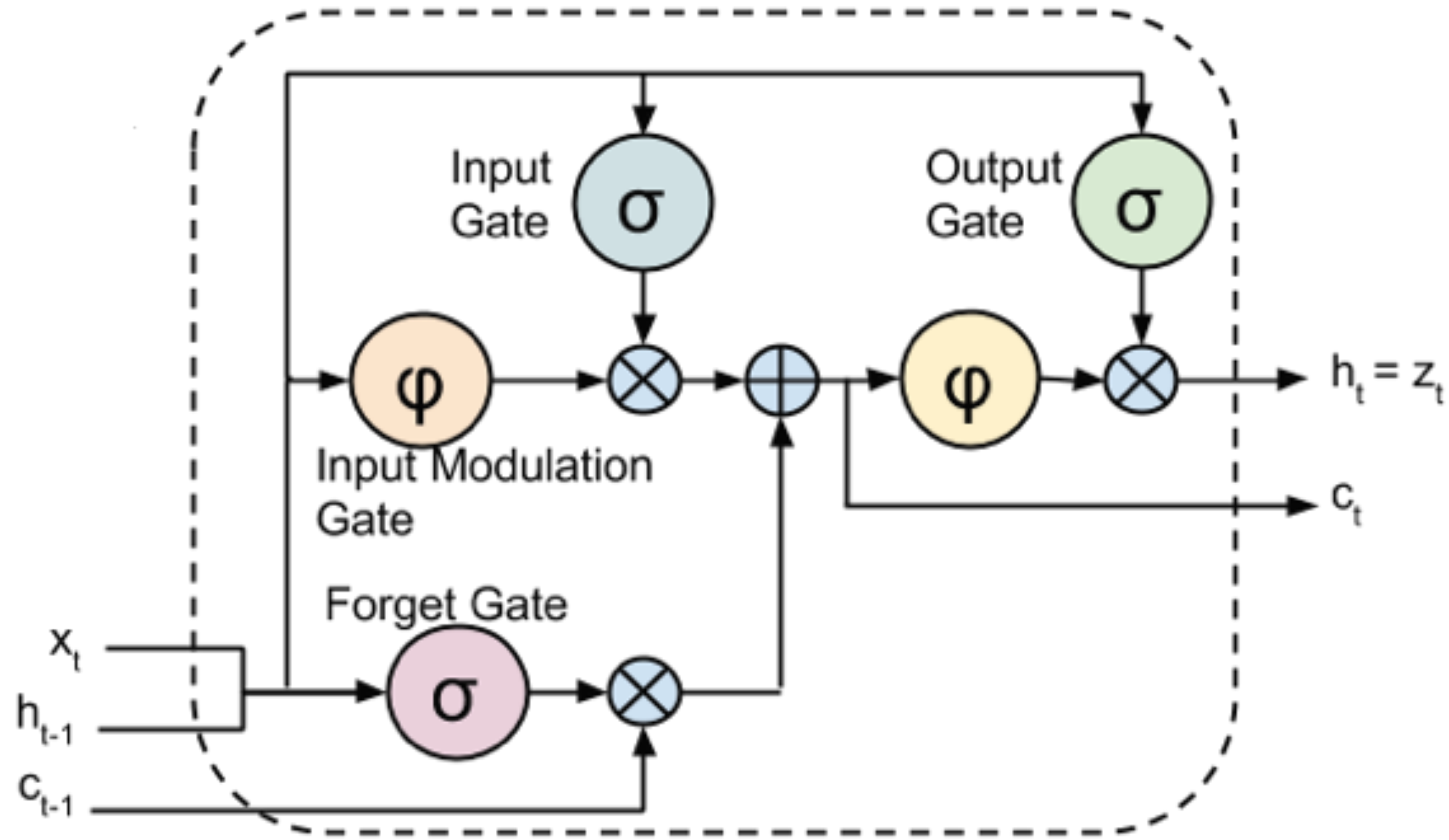
Sequence Learning



Long Short-Term Memory (LSTM)
Proposed by Hochreiter and Schmidhuber, 1997

Sequence Learning

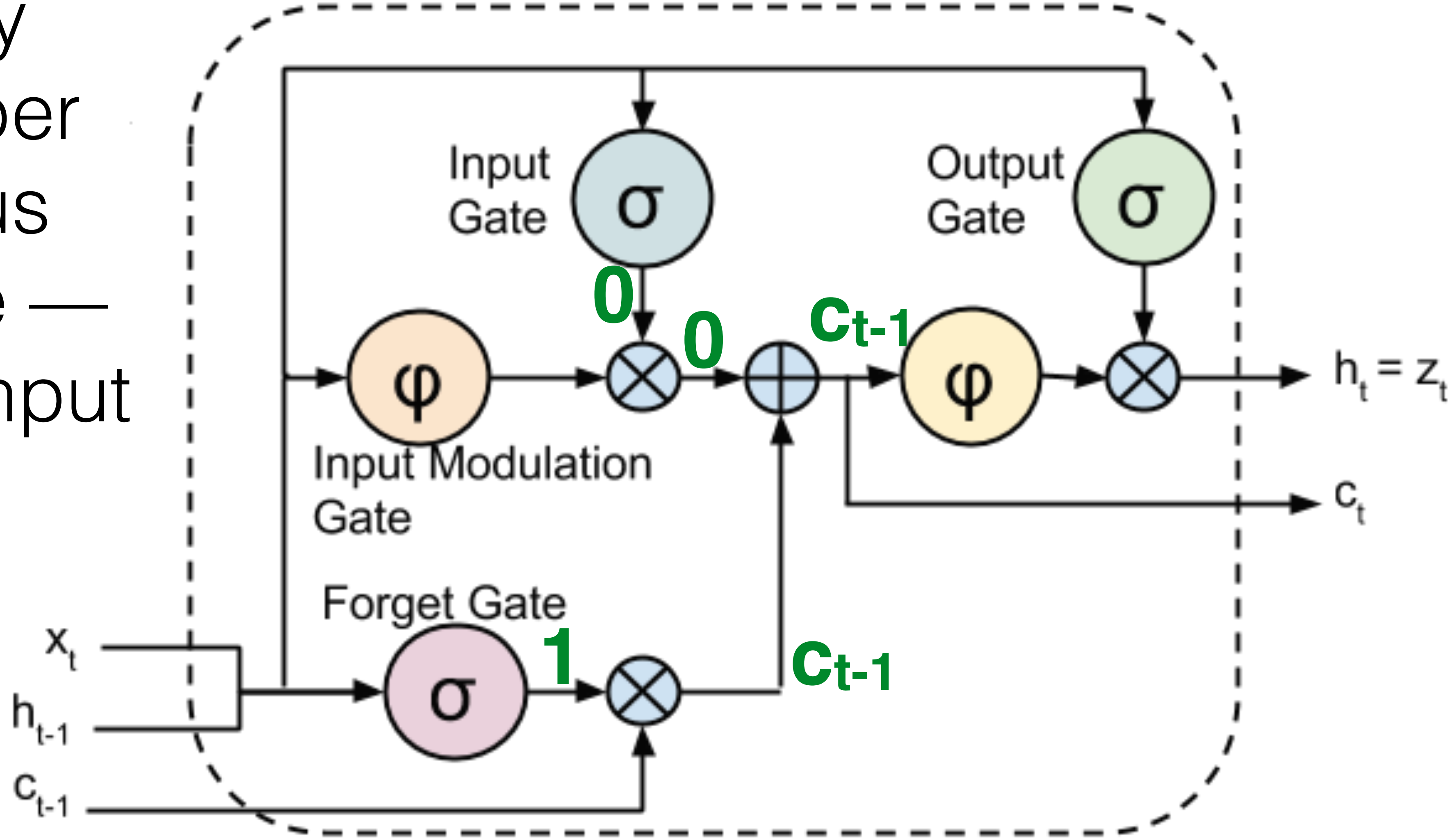
- Allows long-term dependencies to be learned
- Effective for
  - speech recognition
  - handwriting recognition
  - translation
  - parsing



LSTM
(Hochreiter & Schmidhuber, 1997)

Sequence Learning

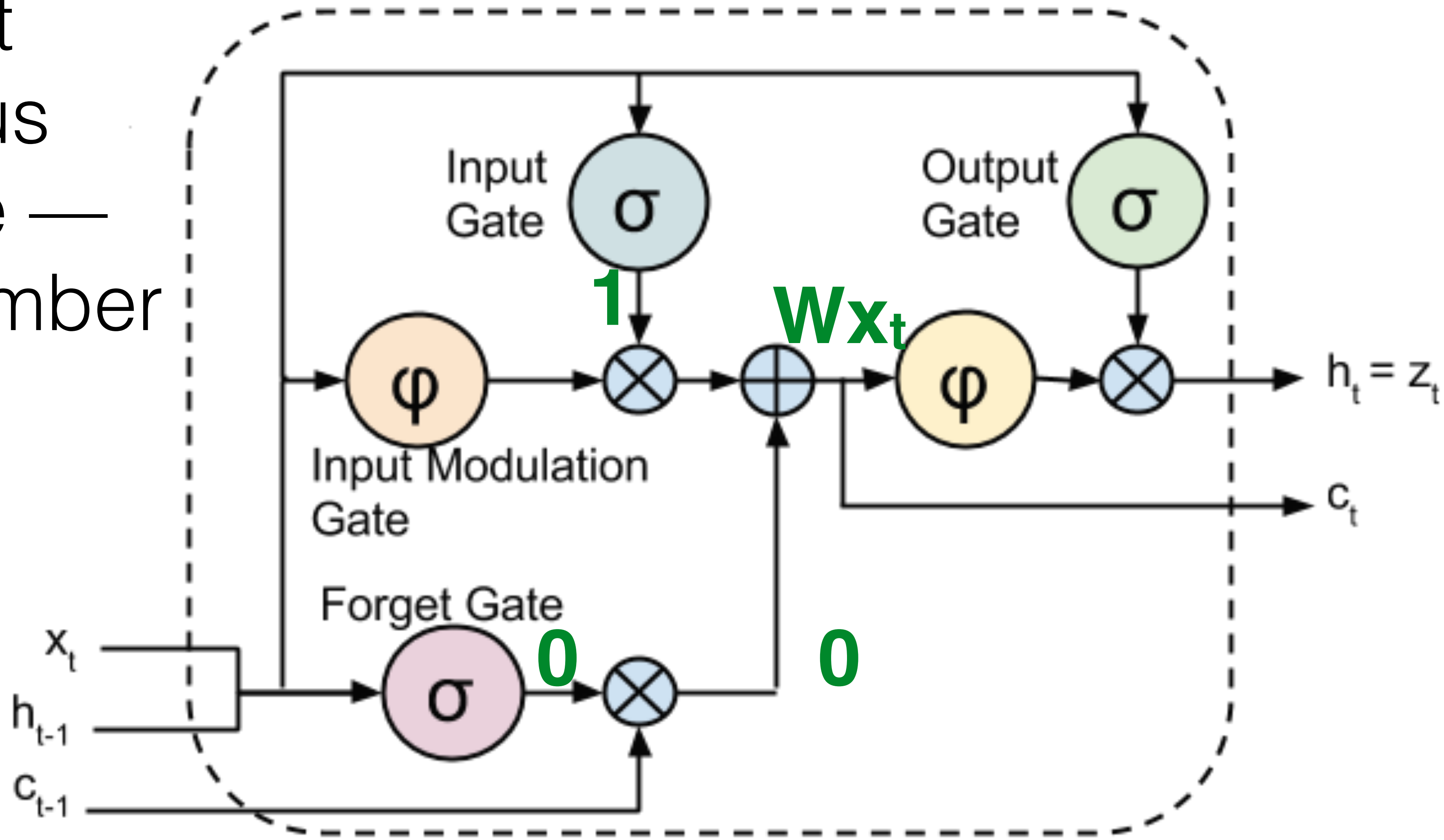
Exactly remember previous cell state — discard input



LSTM
(Hochreiter & Schmidhuber, 1997)

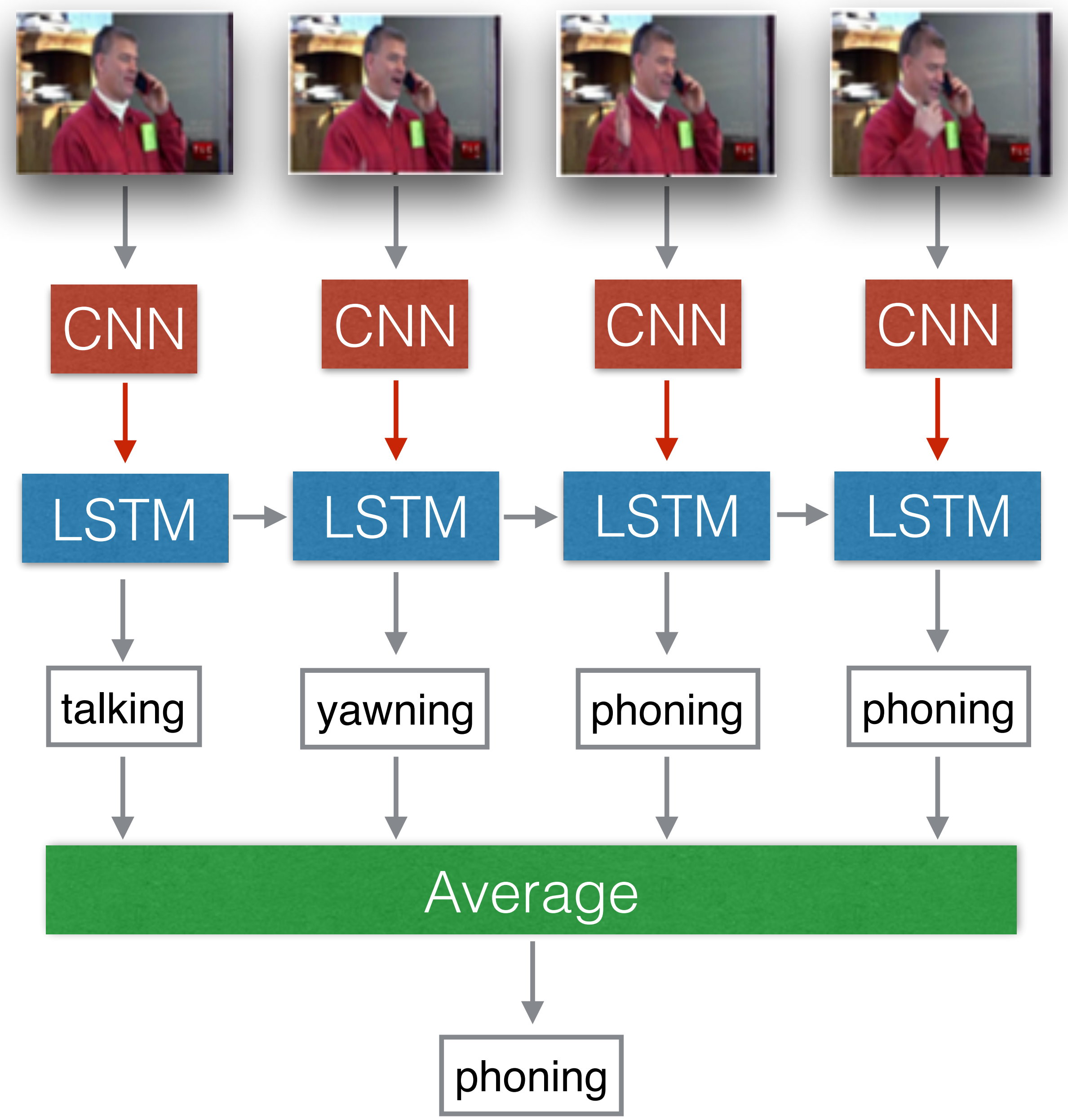
Sequence Learning

Forget previous cell state — only remember input



LSTM
(Hochreiter & Schmidhuber, 1997)

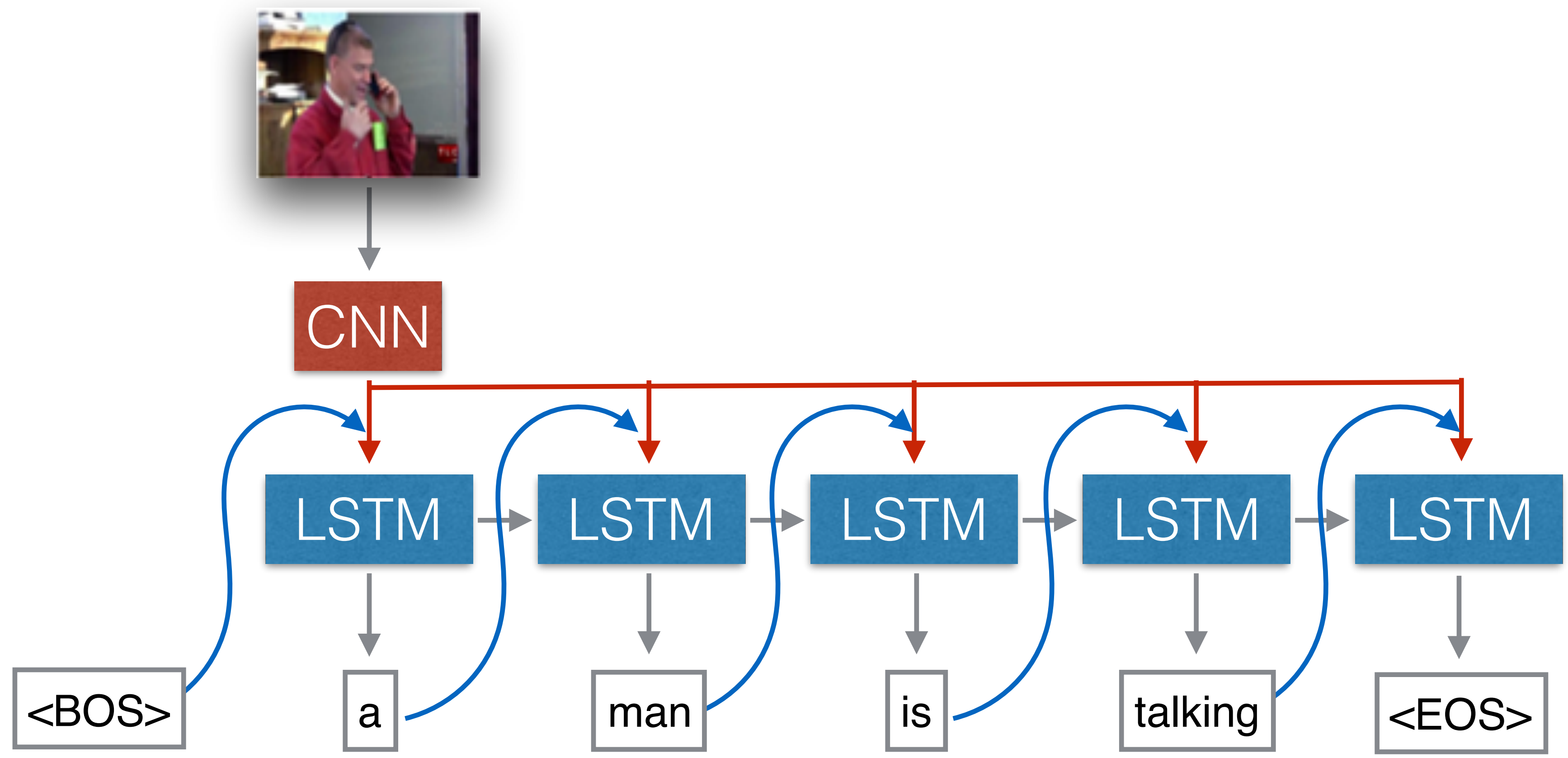
Activity Recognition



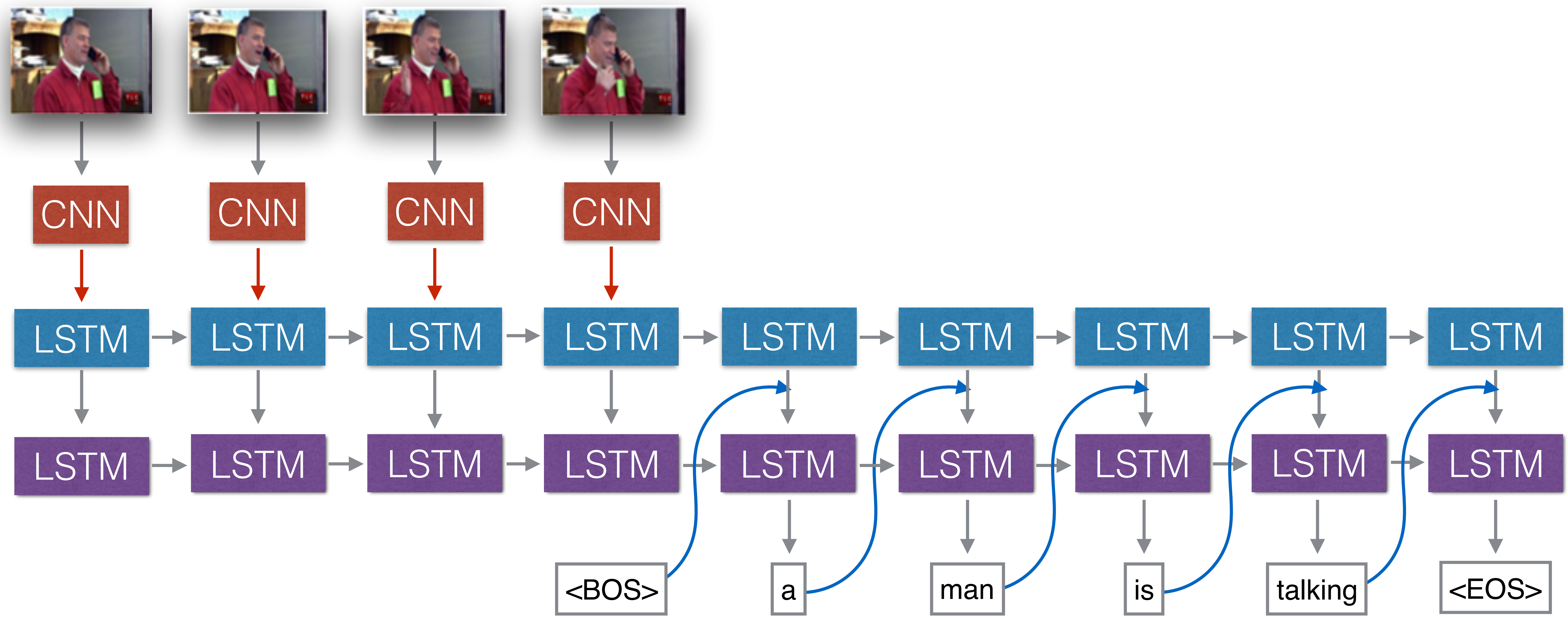
sequential
input

Image Description

sequential
output



Video Description



sequential input & output

Sequence learning features now available in Caffe.
Check out PR #2033
“Unrolled recurrent layers (RNN, LSTM)”

Training Sequence Models

- At training time, want the model to predict the next time step given all previous time steps: $p(w_{t+1} \mid w_{1:t})$
- Example: *A bee buzzes.*

	input	output
0	<BOS>	a
1	a	bee
2	bee	buzzes
3	buzzes	<EOS>

Sequence Input Format

- First input: “cont” (continuation) indicators (T x N)
- Second input: data (T x N x D)

N = 2, T = 6

batch 1

a	dog	fetches	<EOS>	the	bee
0	1	1	1	0	1
cat	in	a	hat	<EOS>	a
0	1	1	1	1	0

batch 2…

buzzes	<EOS>	a
1	1	0
tree	falls	<EOS>
1	1	1

Sequence Input Format

- Inference is exact over infinite batches
- Backpropagation approximate — truncated at batch boundaries

N = 2, T = 6

batch 1

a	dog	fetches	<EOS>	the	bee
0	1	1	1	0	1
cat	in	a	hat	<EOS>	a
0	1	1	1	1	0

batch 2…

buzzes	<EOS>	a
1	1	0
tree	falls	<EOS>
1	1	1

Sequence Input Format

- Words are usually represented as one-hot vectors

vocabulary with 8000 words

a	0
the	1
am	2
.	.
.	.
cat	3999
bee	4000
dog	4001
.	.
.	.
run	7999

8000D vector; all 0s except index of word

"bee"

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ . \\ . \\ 0 \\ 1 \\ 0 \\ . \\ . \\ 0 \end{bmatrix}$$

Sequence Input Format

- EmbedLayer projects one-hot vector

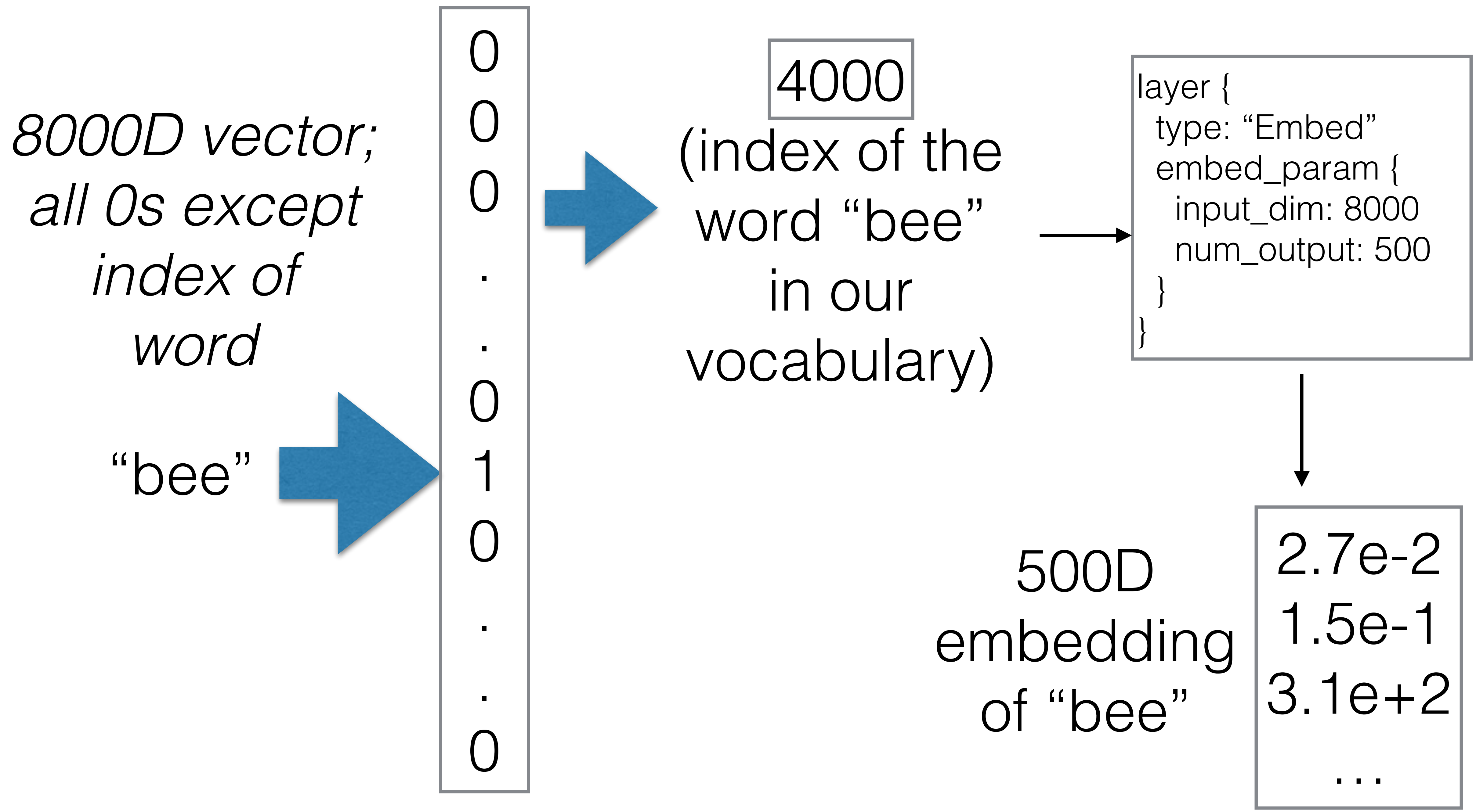


Image Description

A female tennis player in action on the court.

A group of young men playing a game of soccer.

A man riding a wave on top of a surfboard.

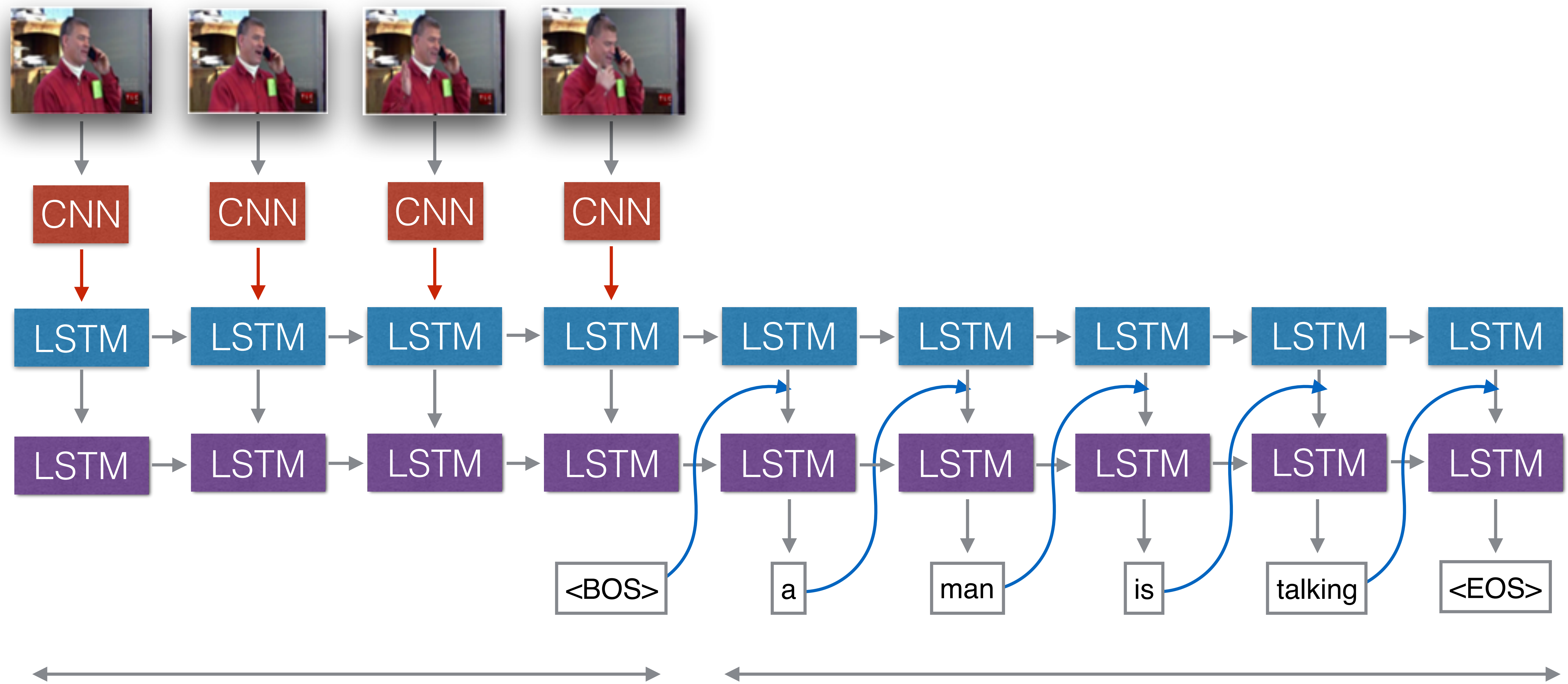
Image Description

A black and white cat is sitting on a chair.

A large clock mounted to the side of a building.

A bunch of fruit that are sitting on a table.

Video Description



N timesteps: watch video M timesteps: produce caption

Venugopalan et al., "Sequence to Sequence -- Video to Text," 2015.
http://arxiv.org/abs/1505.00487