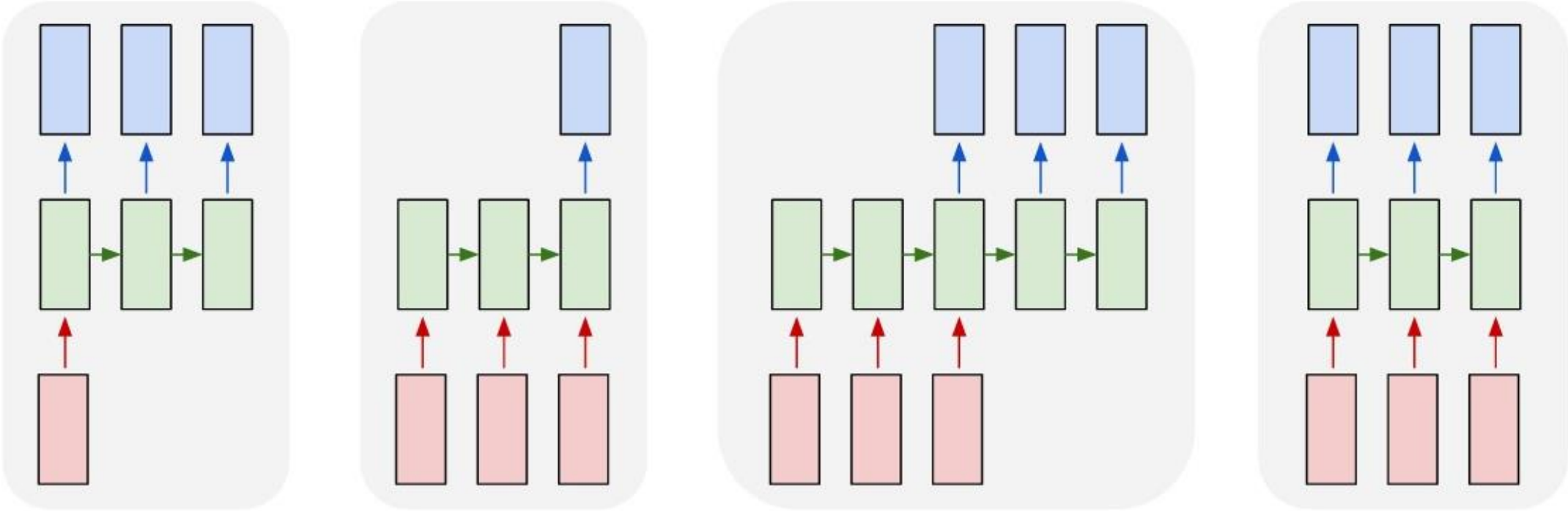
Recurrent Neural Networks

Xiaolong Wang

This Class: Recurrent Neural Networks



Slides partially from: http://cs231n.stanford.edu/ https://slazebni.cs.illinois.edu/fall20/

This Class: Recurrent Neural Networks

- The Basic RNN
- LSTM
- Application in language and vision tasks

The Basic RNN

Sequential prediction tasks

- So far, we focused mainly on prediction problems with fixed-size inputs and outputs

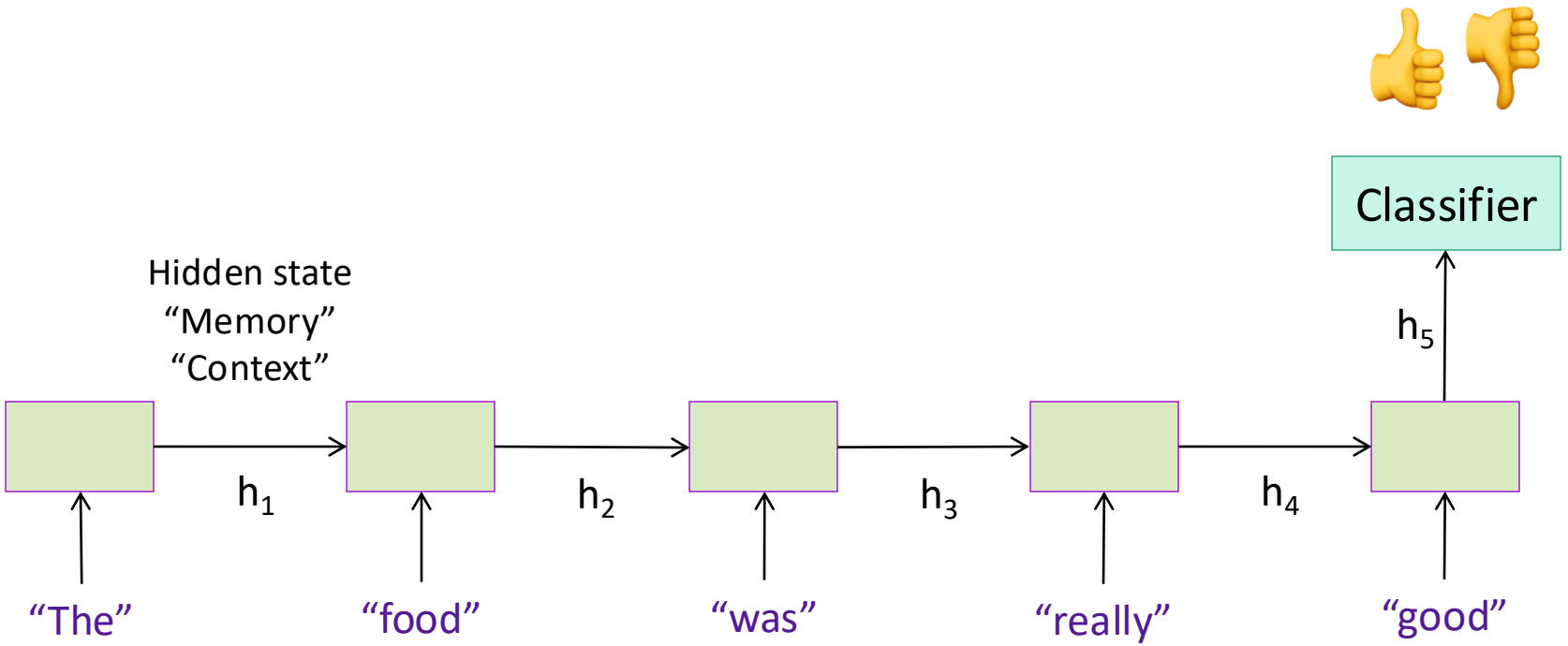
- But what if the input and/or output is a variable-length sequence?

Task 1: Sentiment classification

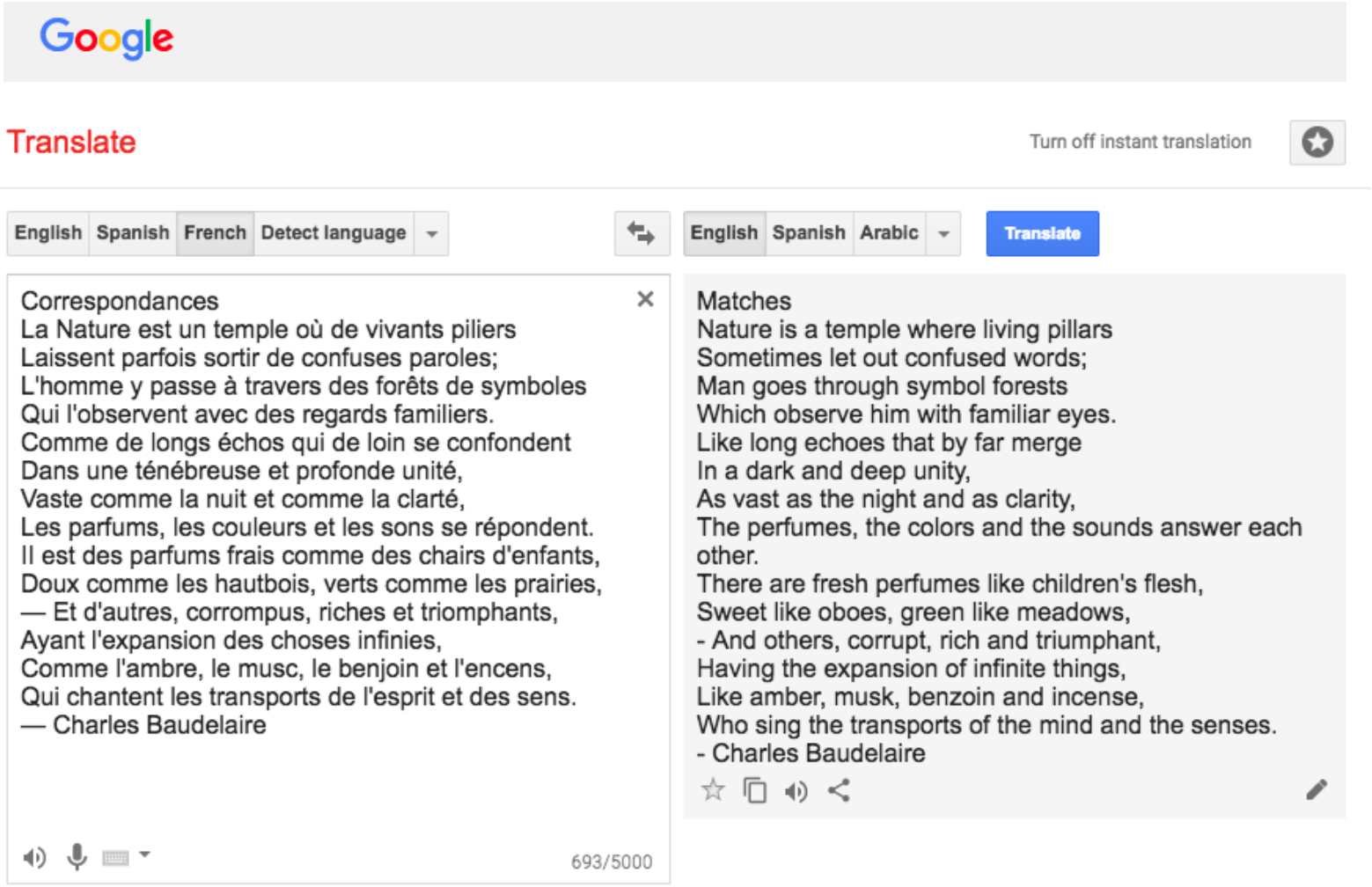
- Goal: classify a text sequence (e.g., restaurant, movie or product review, Tweet) as having positive or negative sentiment
  - “The food was really good”
  - “The vacuum cleaner broke within two weeks”
  - “The movie had slow parts, but overall was worth watching”

Task 1: Sentiment classification

Recurrent model:

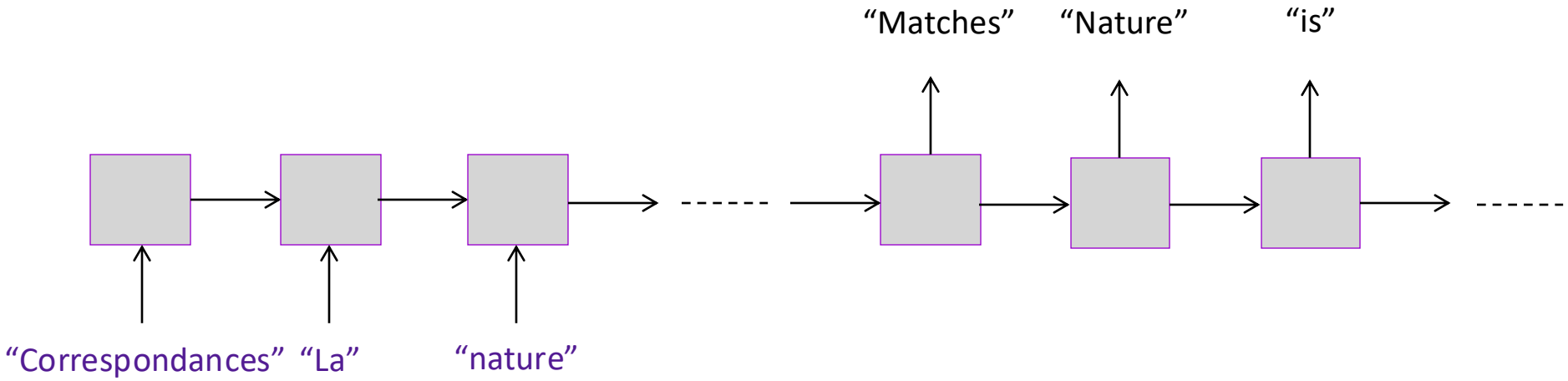


Task 2: Machine translation



https://translate.google.com/

Task 2: Machine translation



Task 3: Image caption generation

A cat sitting on a suitcase on the floor

A cat is sitting on a tree branch

A dog is running in the grass with a frisbee

A white teddy bear sitting in the grass

Two people walking on the beach with surfboards

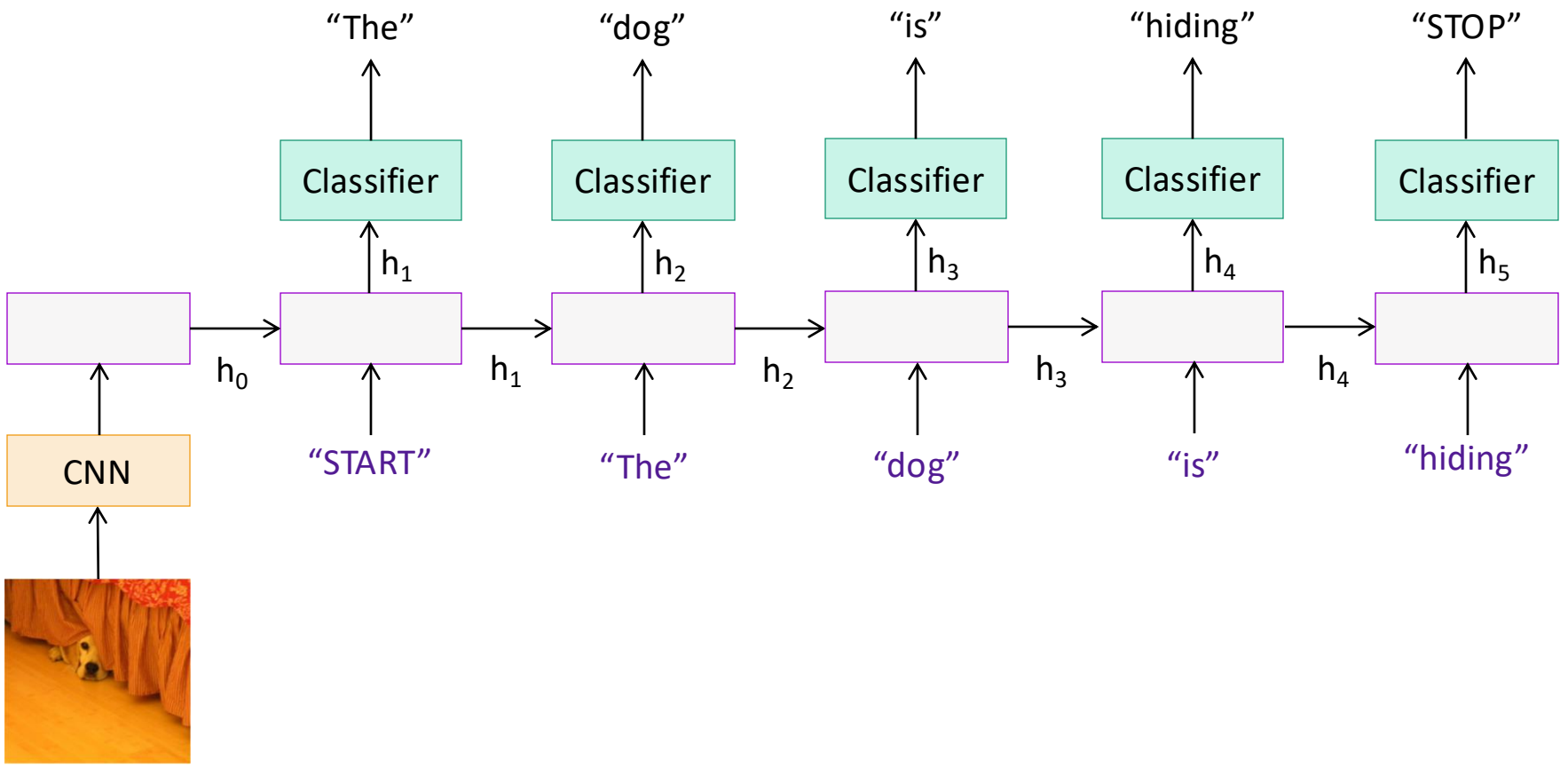
A tennis player in action on the court

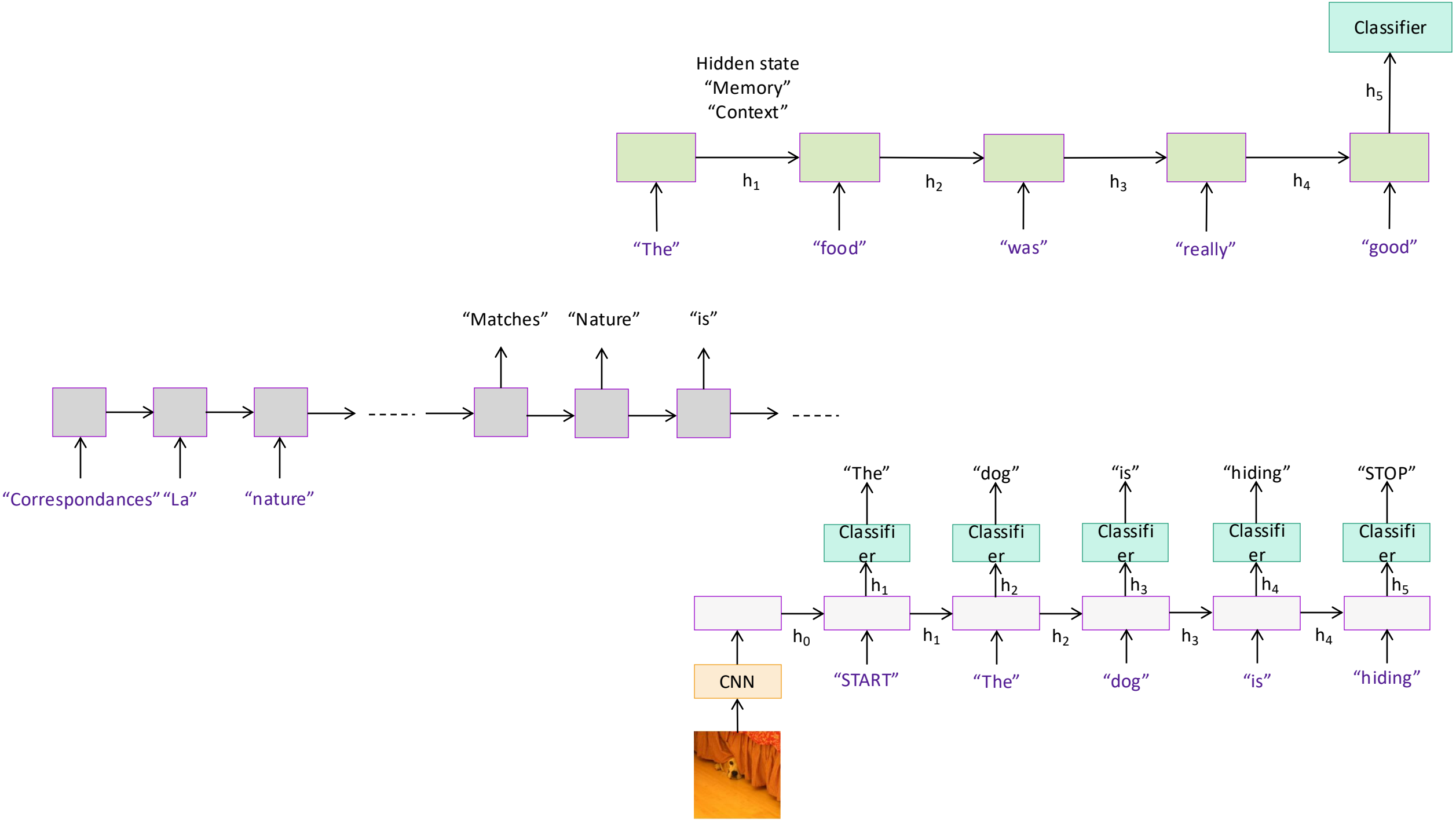
Two giraffes standing in a grassy field

A man riding a dirt bike on a dirt track

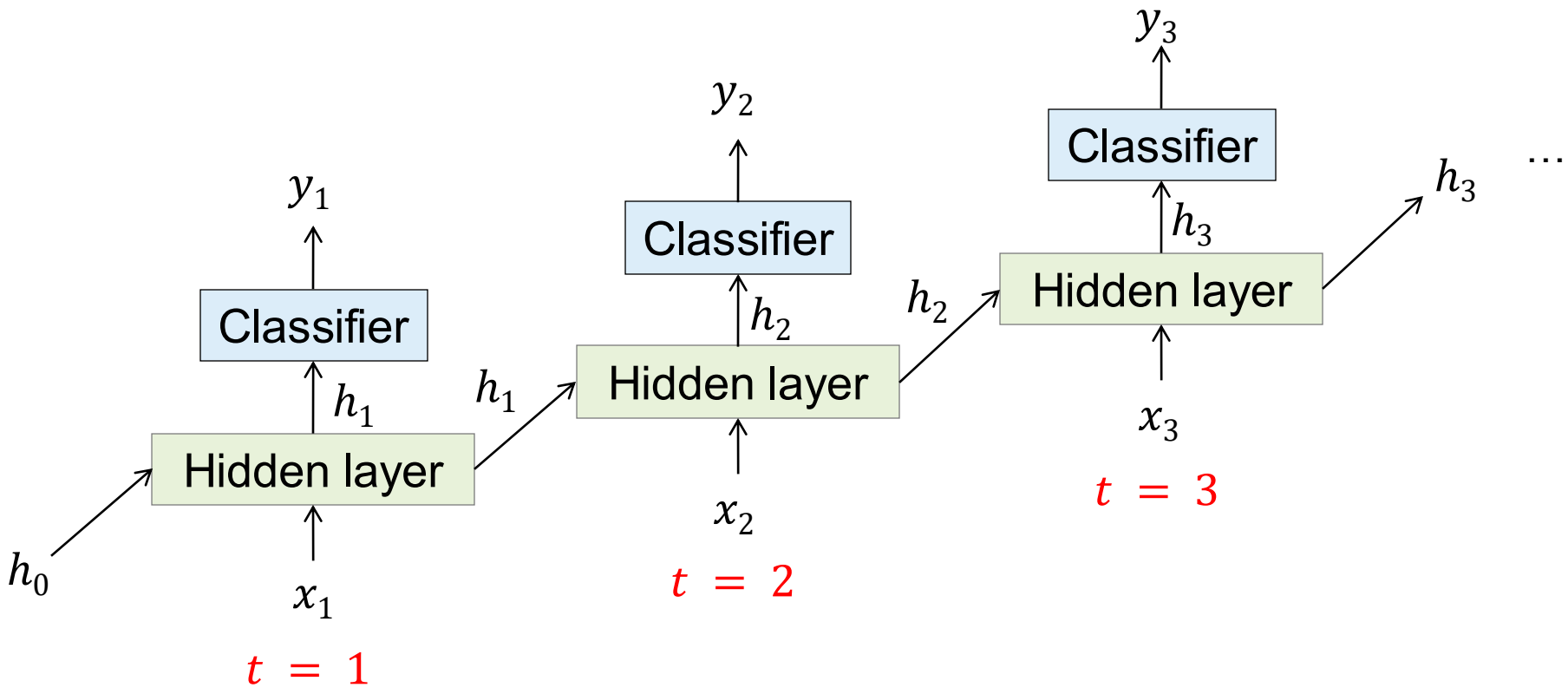
Source: J. Johnson

Task 3: Image caption generation

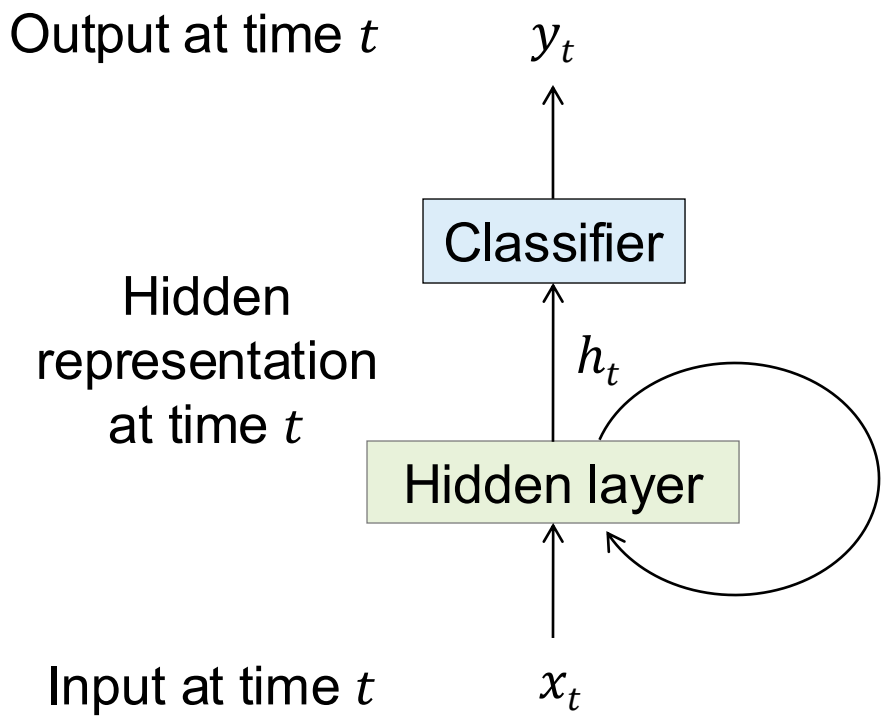



Classifier
Hidden state
"Memory"
"Context"
h_5
h_1
h_2
h_3
h_4
"The"
"food"
"was"
"really"
"good"
"Matches"
"Nature"
"is"
"Correspondances"
"La"
"nature"
"The"
"dog"
"is"
"hiding"
"STOP"
Classifier
h_0
CNN
"START"
"The"
"dog"
"is"
"hiding"

Recurrent unit



Recurrent unit



Recurrence:

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

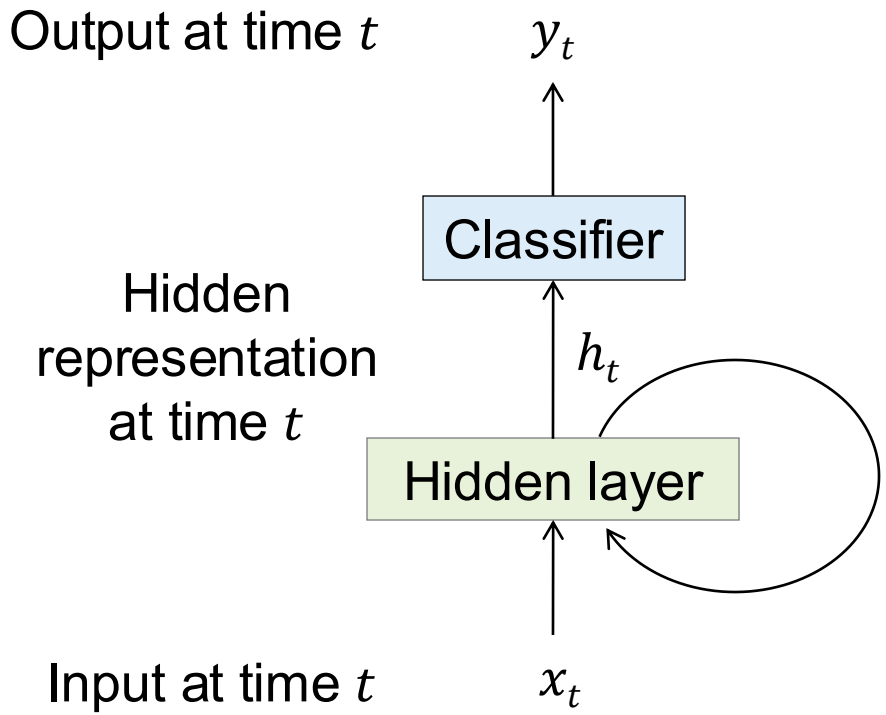
new state

function of W

input at time t

old state

Recurrent unit



Recurrence:

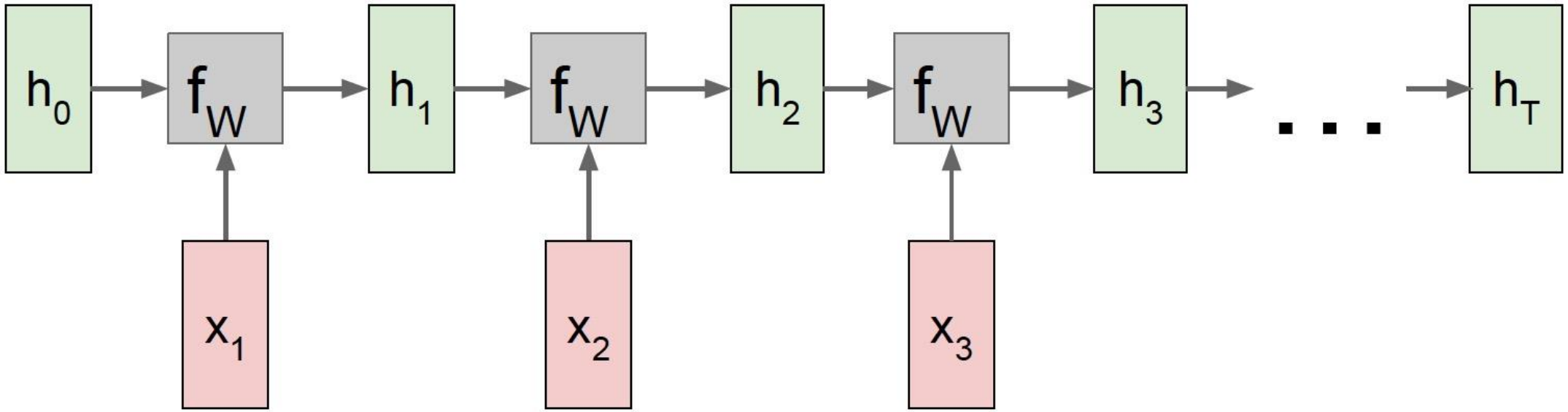
$$h_t = f_W(x_t, h_{t-1})$$

new state | function of W | input at time t | old state

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

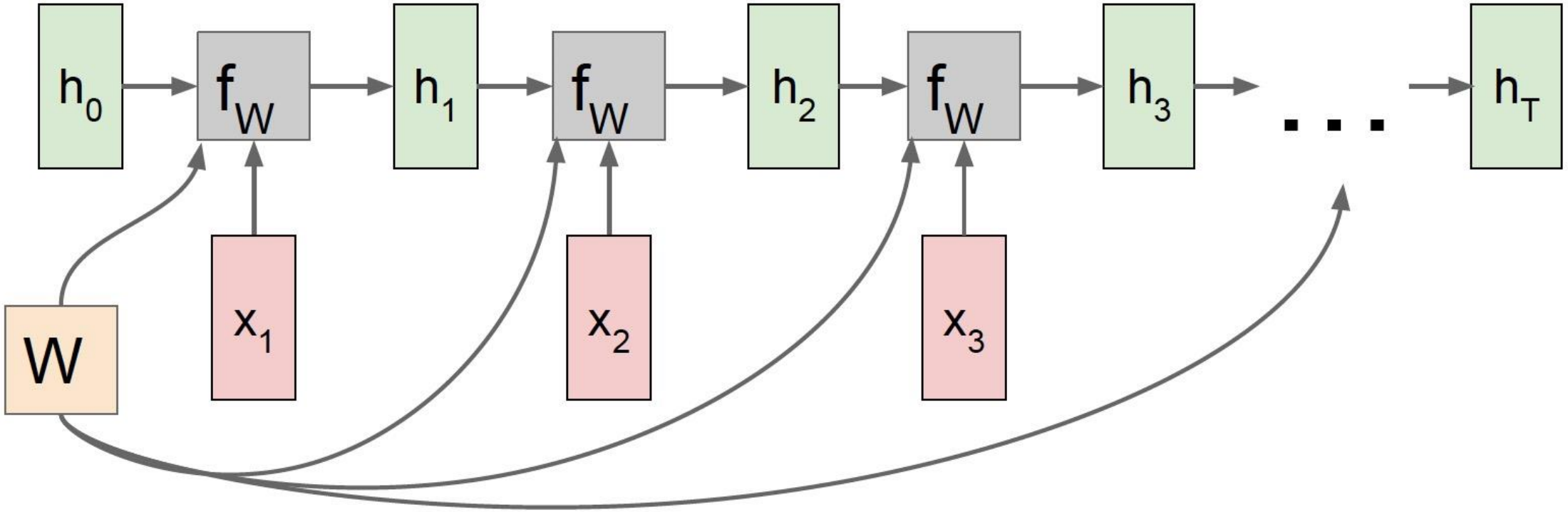
$$y_t = W_{hy}h_t$$

RNN: Computational Graph

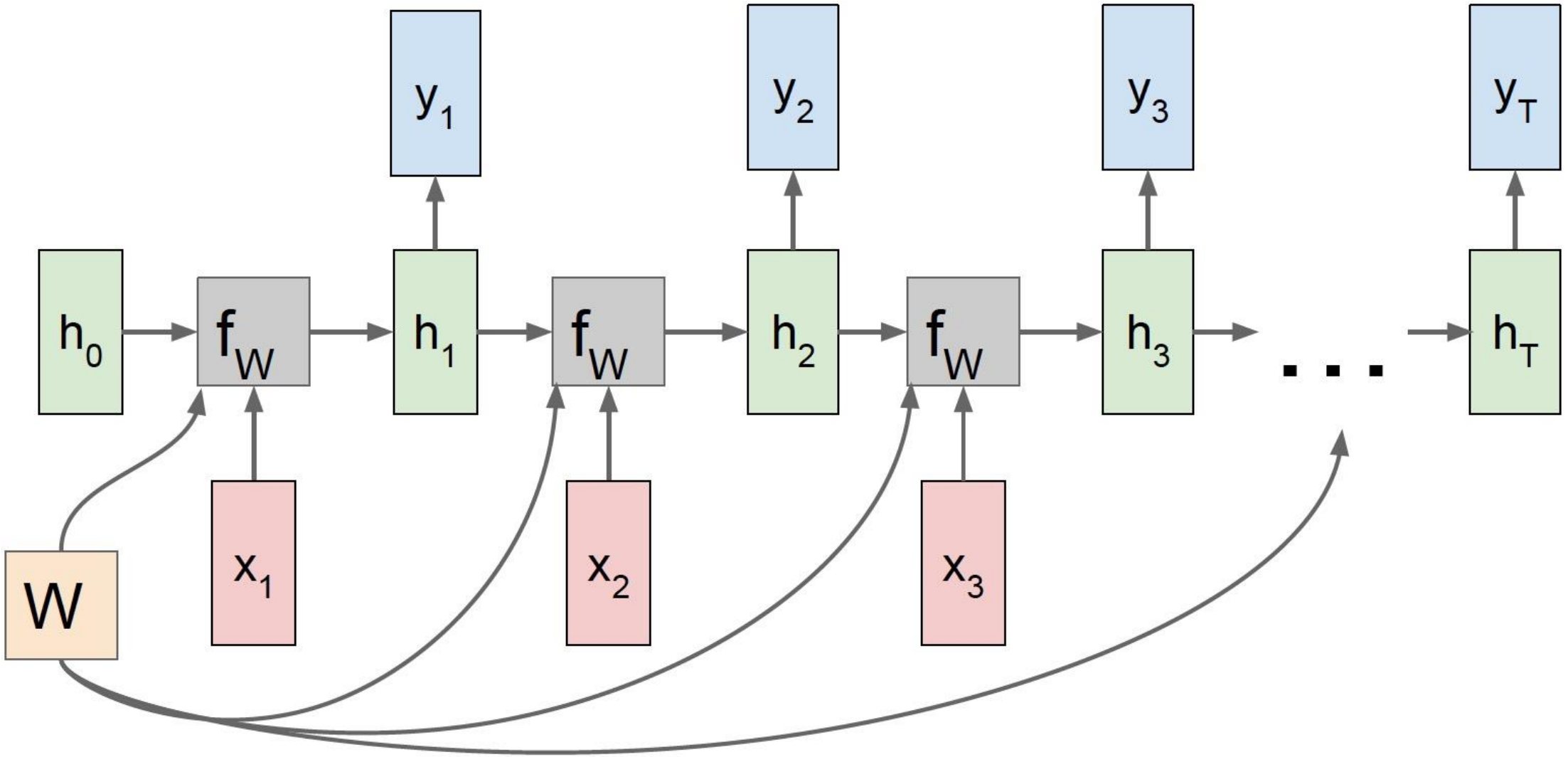


RNN: Computational Graph

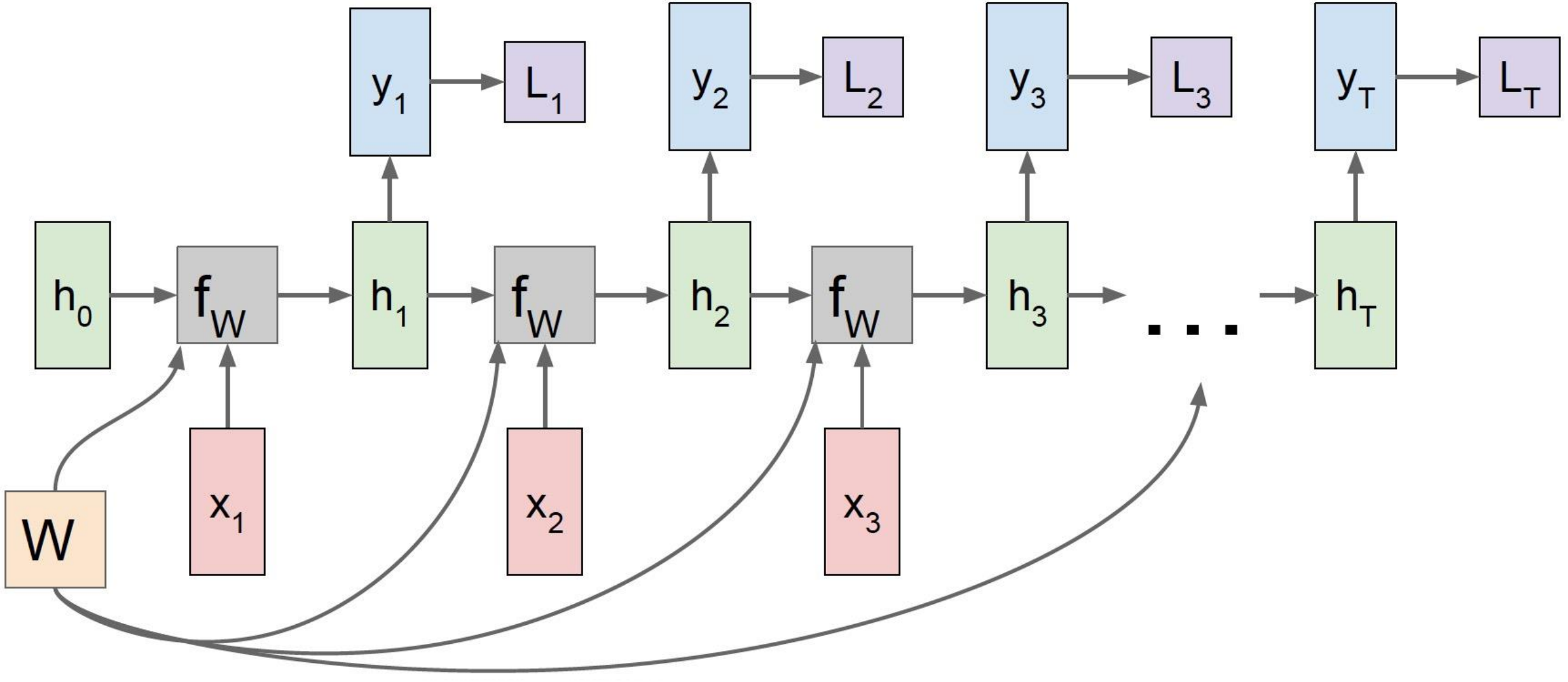
Re-use the same weight in every time step



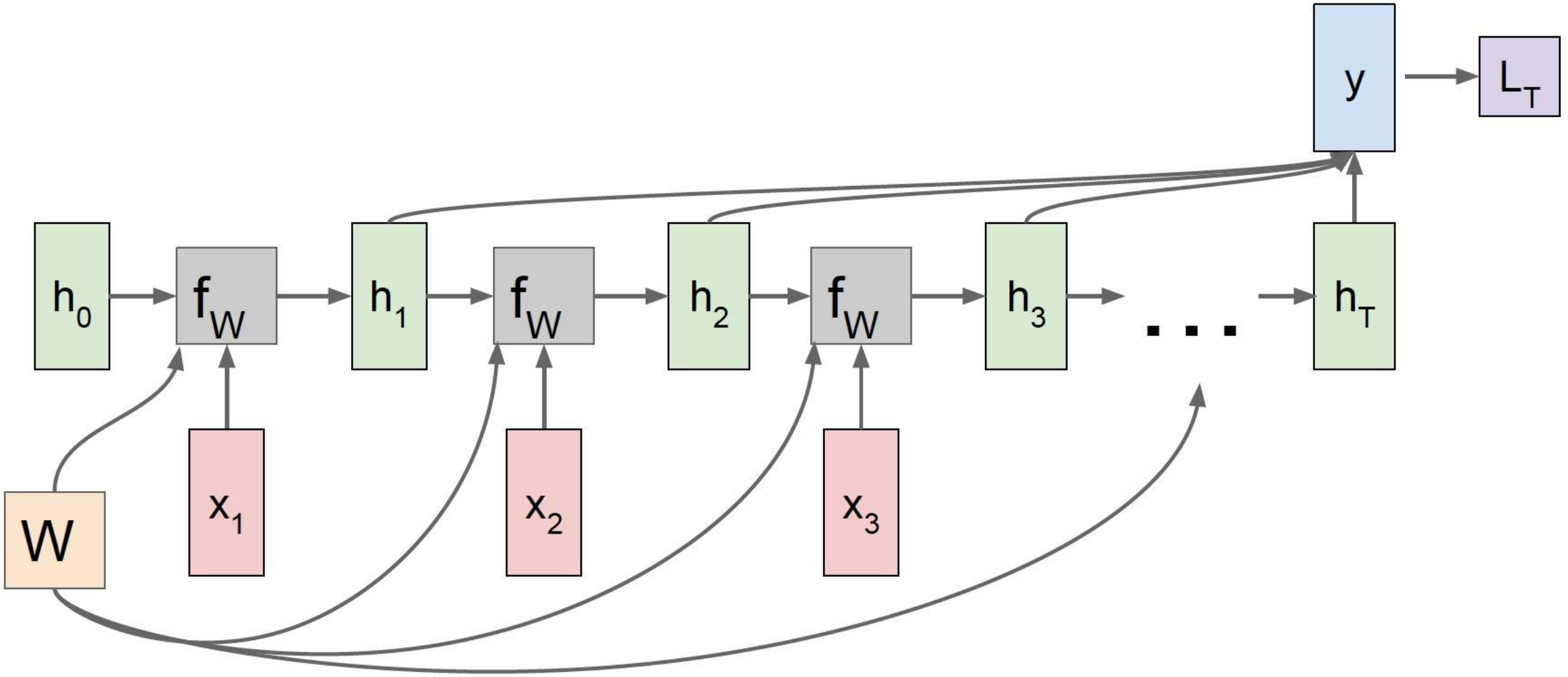
RNN: Computational Graph



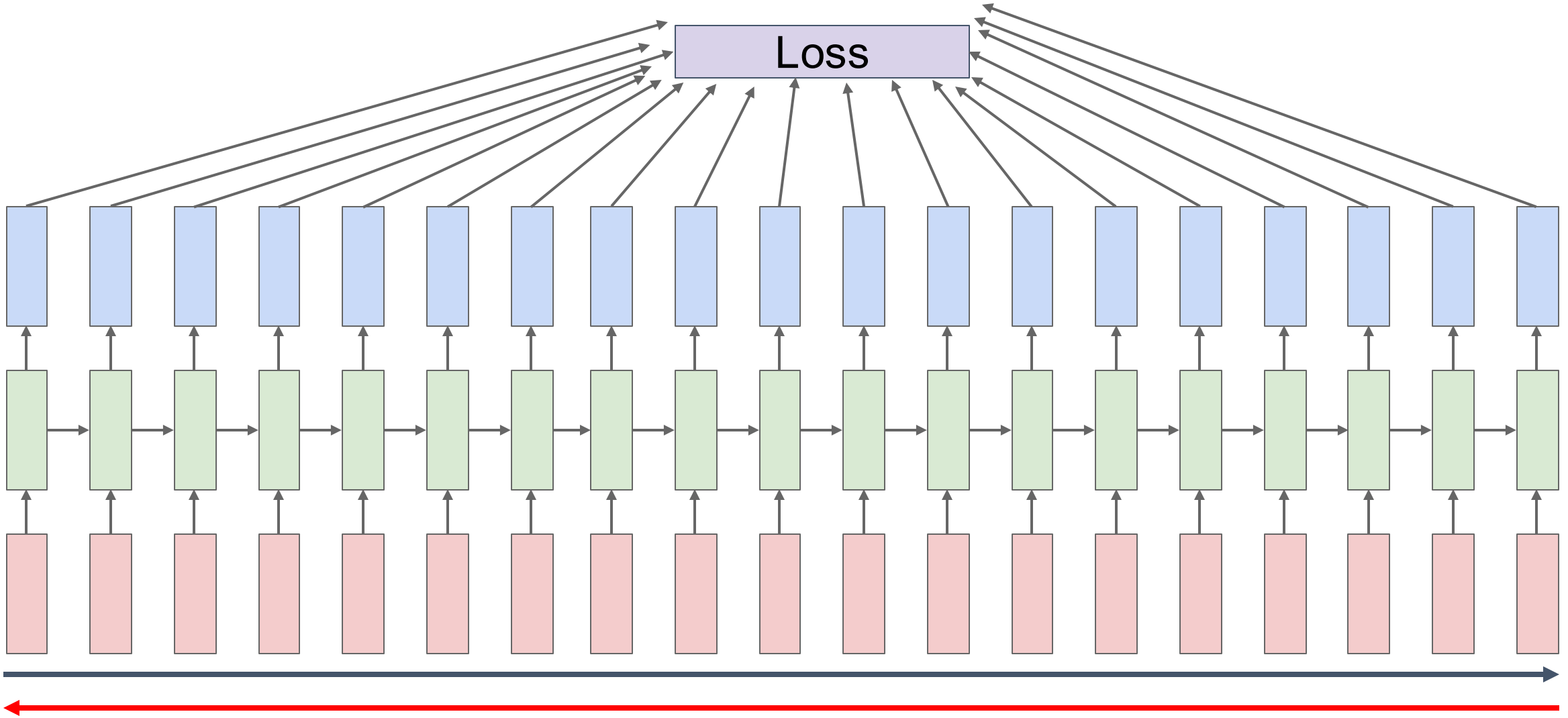
RNN: Computational Graph



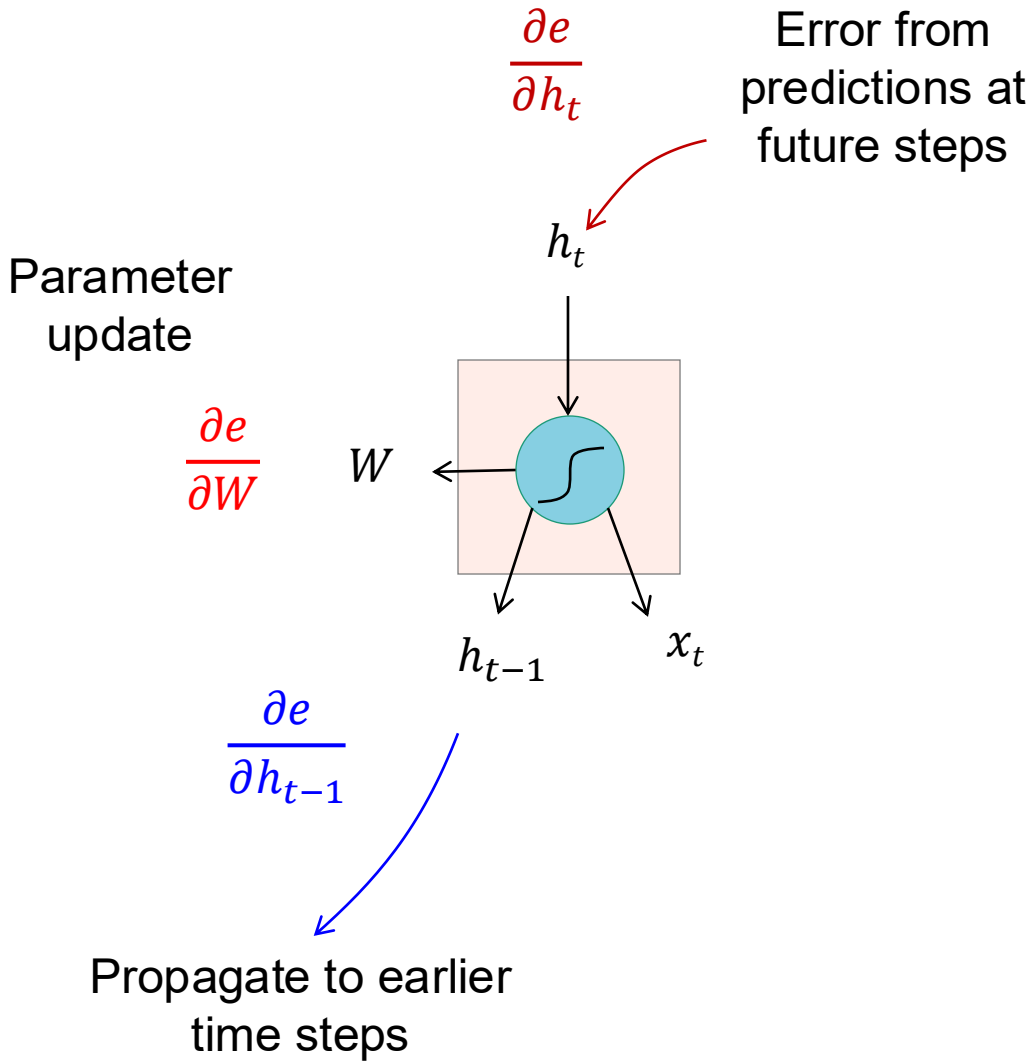
RNN: Computational Graph



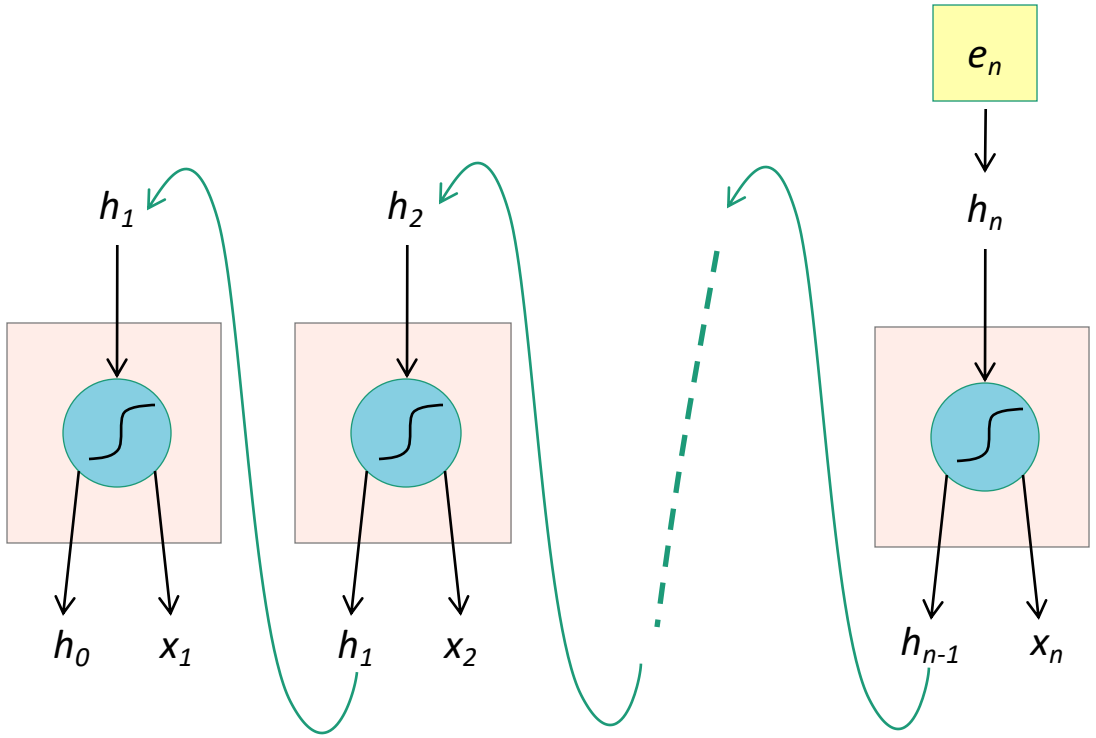
Backpropagation through time



Backpropagation through time (BPTT)



Backpropagation through time (BPTT)



Char-RNN

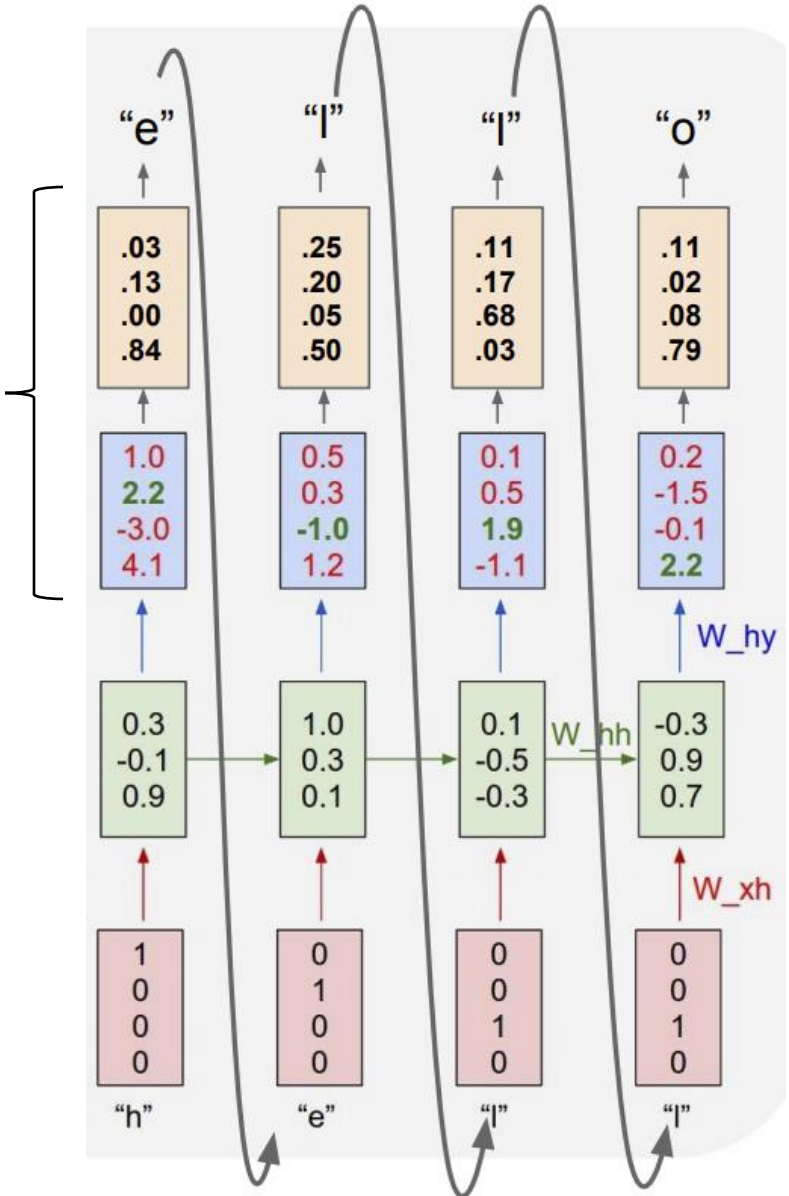
Output symbol y_i

Output layer
(linear + softmax)

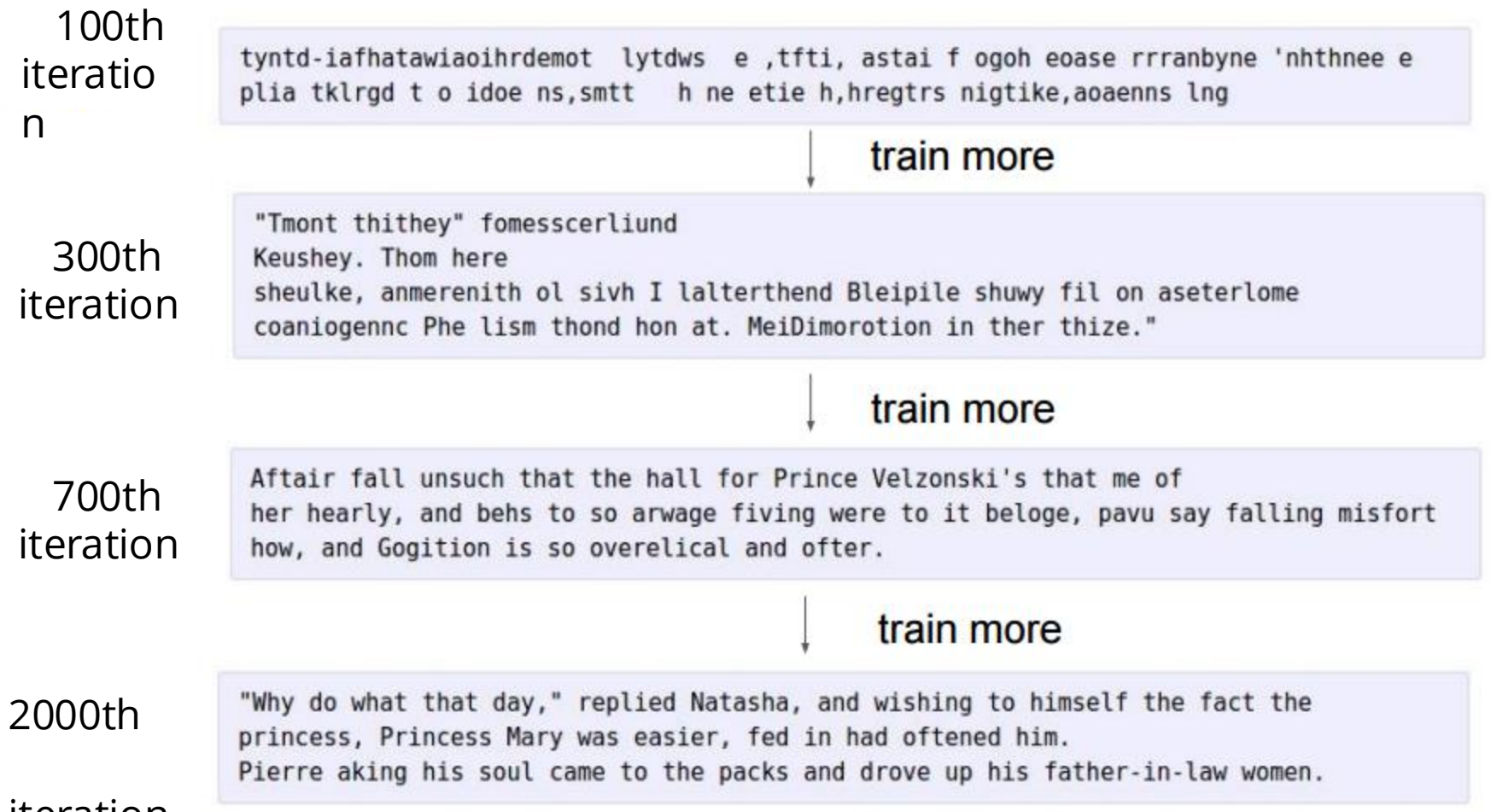
Hidden state h_i

One-hot encoding x_i

Input symbol

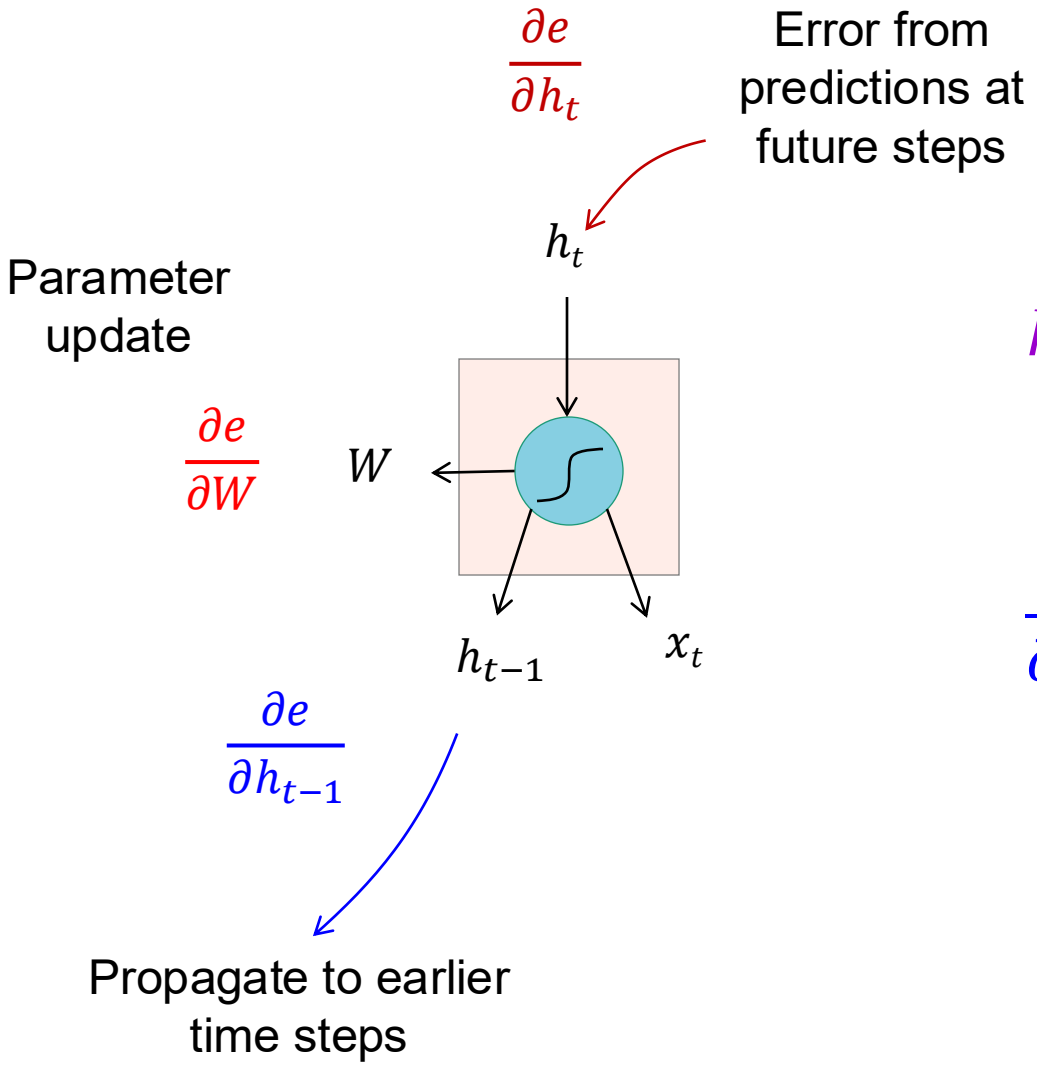


Char-RNN


100th iteration
tyntd-iafhatawiaoihrdemot lytdws e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e
plia tklrgd t o idoe ns,smtt h ne etie h,hregtrs nigtike,aoaenns lng
train more
300th iteration
"Tmont thithey" fomesscerliund
Keushey. Thom here
sheulke, anmerenith ol sivh I lalterthend Bleipile shuwy fil on aseterlome
coaniogennc Phe lism thond hon at. MeiDimorotion in ther thize."
train more
700th iteration
Aftair fall unsuch that the hall for Prince Velzonski's that me of
her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort
how, and Gogition is so overelical and ofter.
train more
2000th iteration
"Why do what that day," replied Natasha, and wishing to himself the fact the
princess, Princess Mary was easier, fed in had oftened him.
Pierre aking his soul came to the packs and drove up his father-in-law women.

Long short-term memory (LSTM)

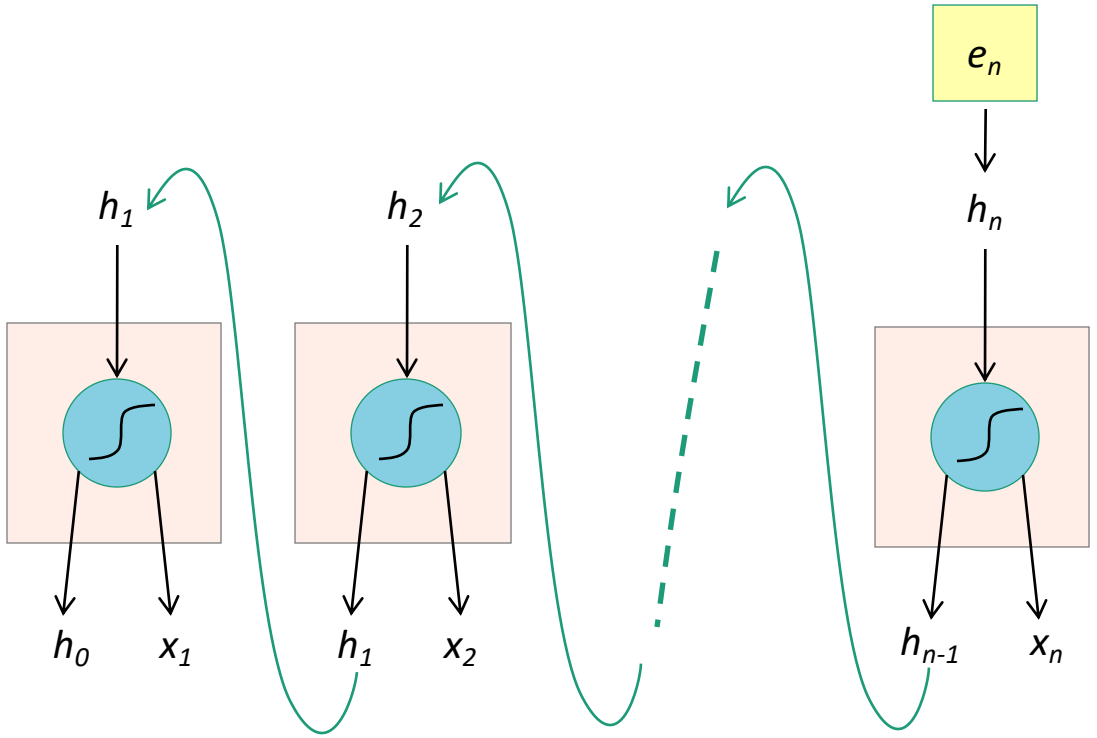
Vanishing Gradients in RNNs



$$h_t = \tanh(W_x x_t + W_h h_{t-1})$$

$$\frac{\partial e}{\partial h_{t-1}} = W_h^T \left(1 - \tanh^2(W_x x_t + W_h h_{t-1})\right) \odot \frac{\partial e}{\partial h_t}$$

Vanishing Gradients in RNNs



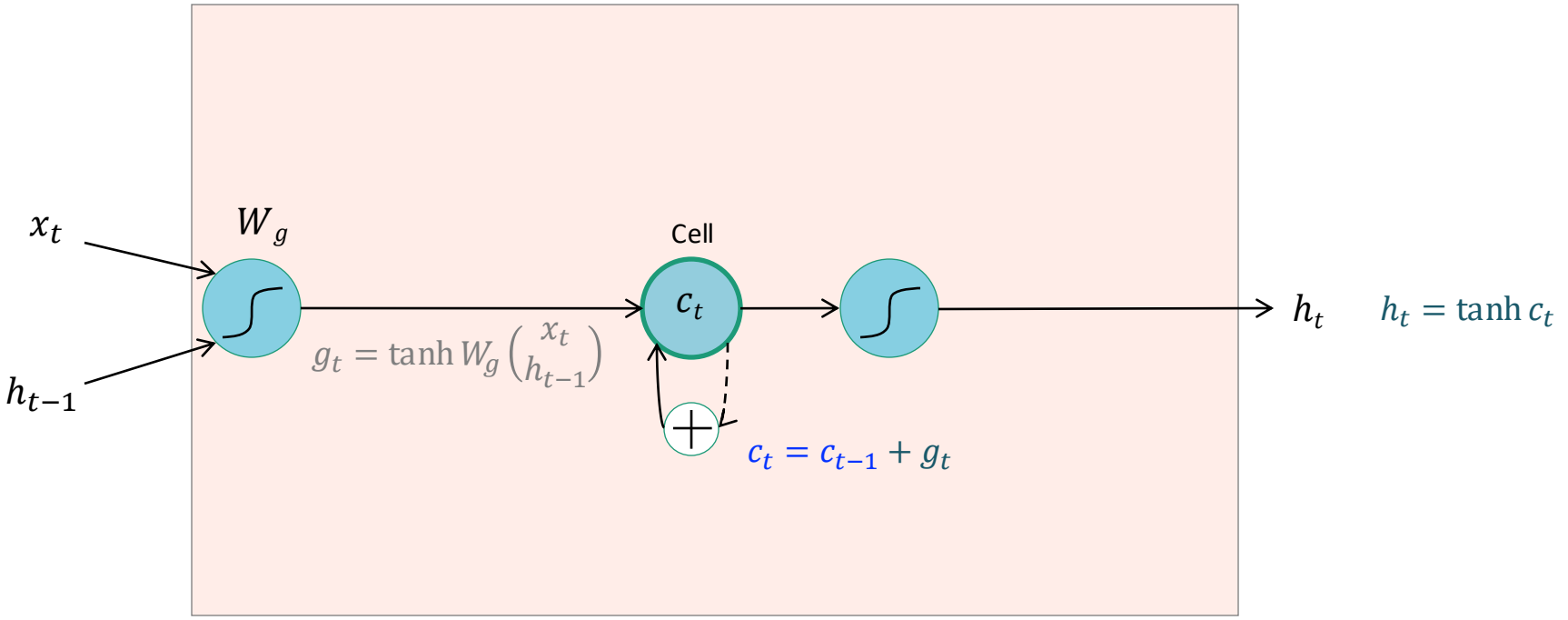
$$\frac{\partial e}{\partial h_{t-1}} = W_h^T\left(1 - \tanh^2(W_x x_t + W_h h_{t-1})\right) \odot \frac{\partial e}{\partial h_t}$$

Computing gradient for h_0
involves many multiplications by W_h^T
(and rescalings between 0 and 1)

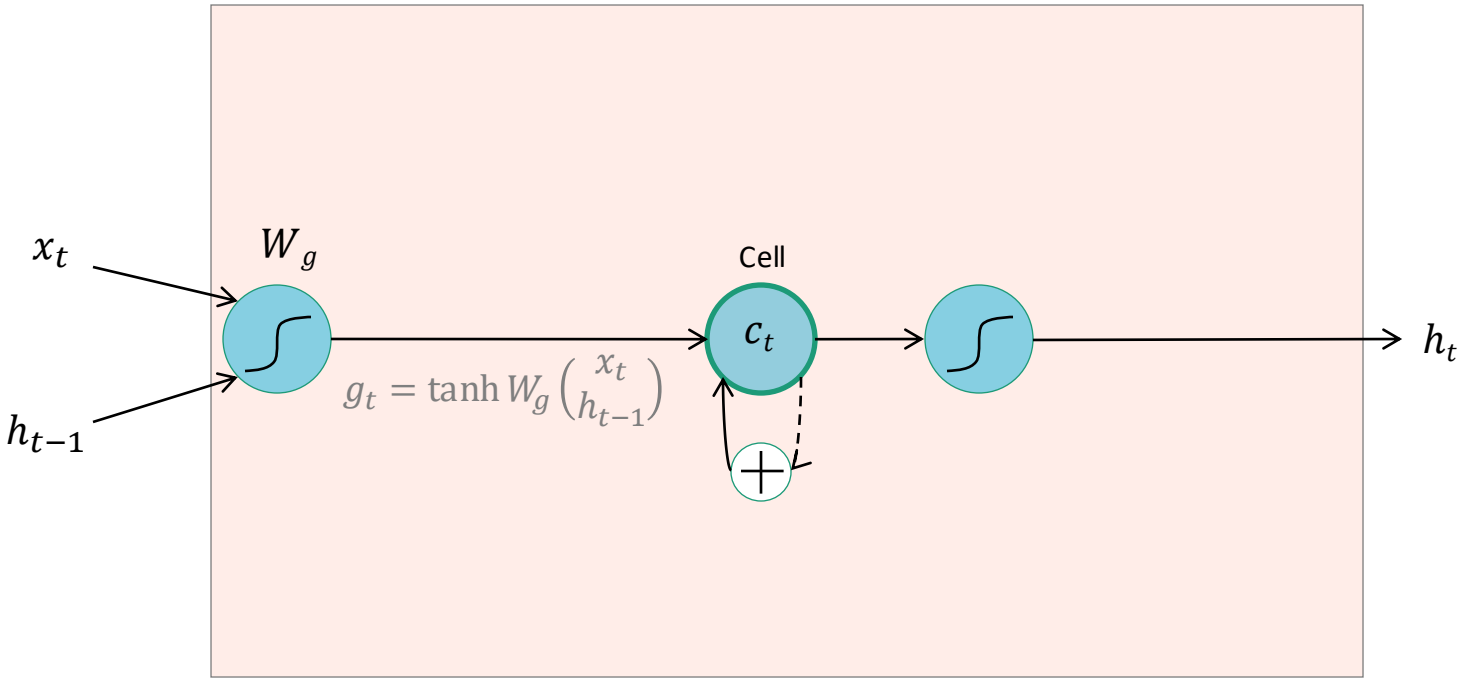
Gradients will *vanish* if largest
singular value of W_h is less than 1
and *explode* if it's greater than 1

Long short-term memory (LSTM) cell

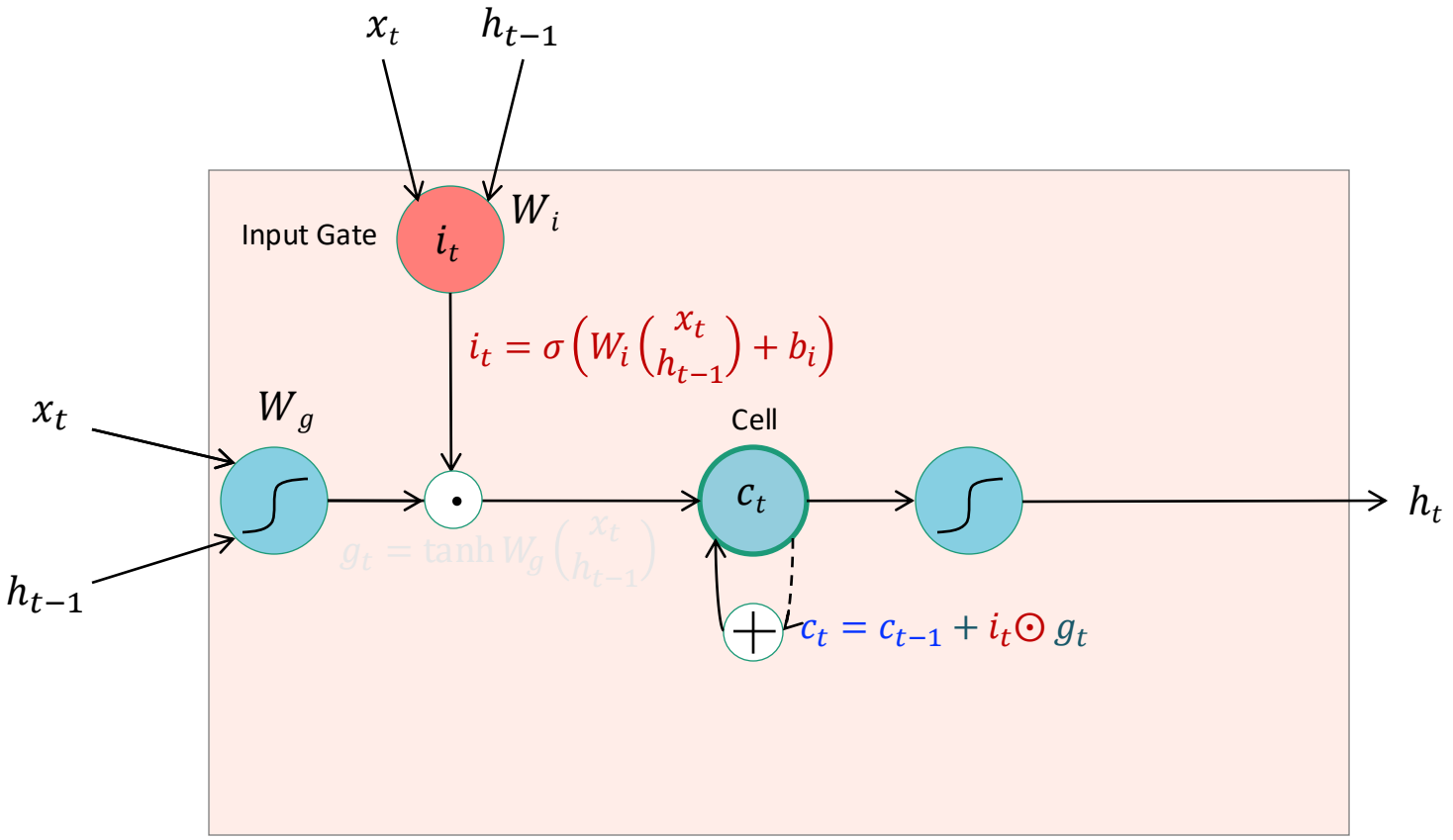
- Add a *memory cell* that is not subject to matrix multiplication or squishing, thereby avoiding gradient decay



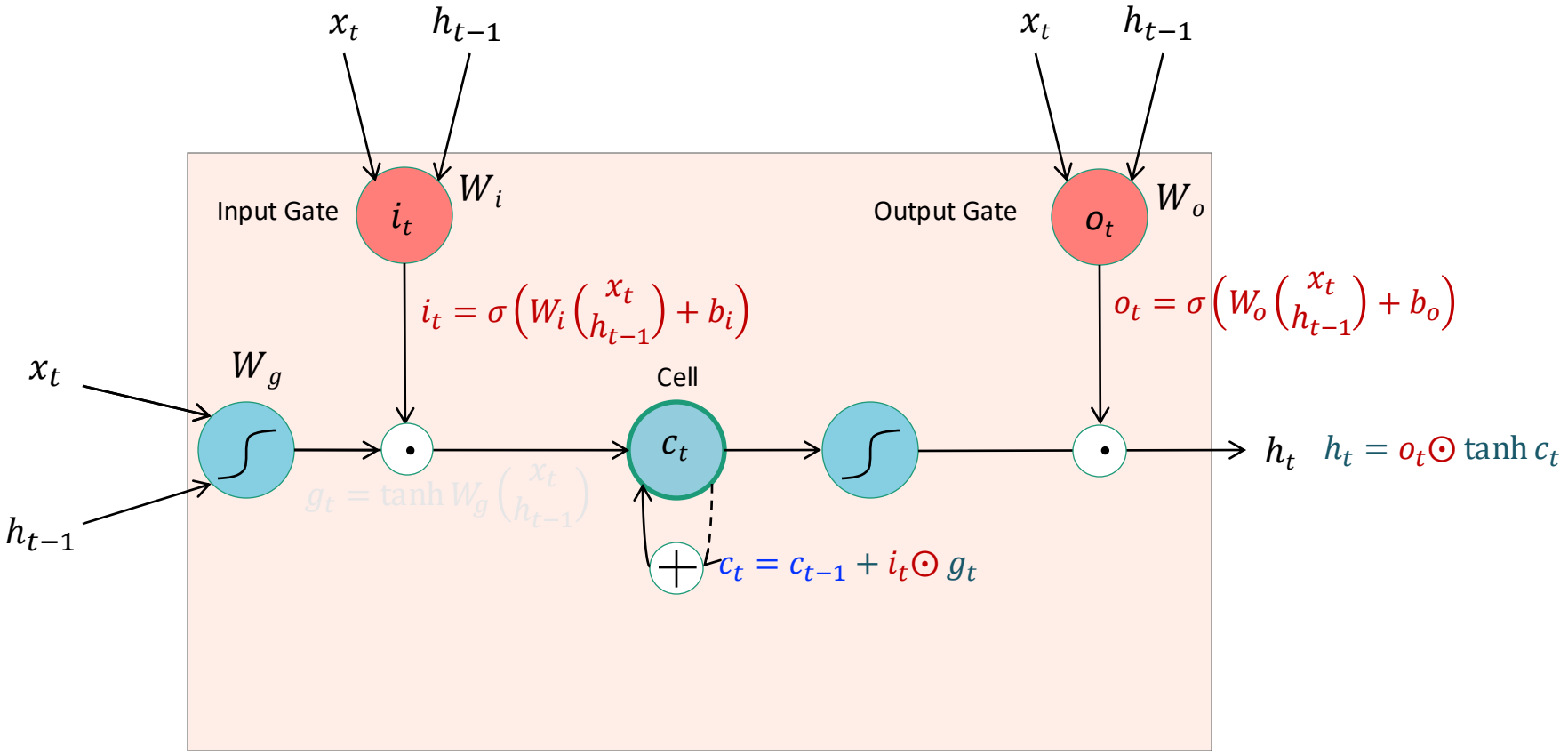
Long short-term memory (LSTM) cell



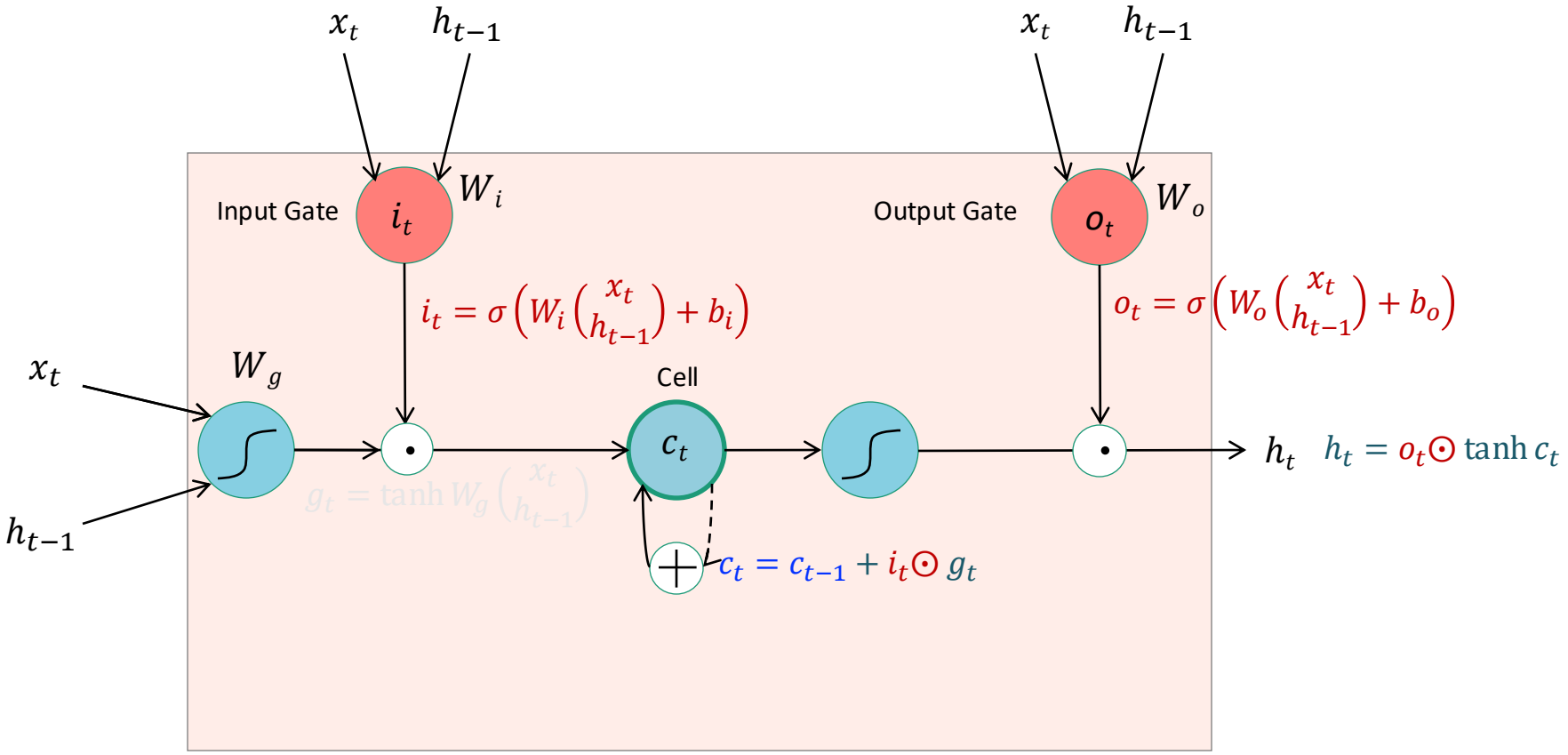
Long short-term memory (LSTM) cell



Long short-term memory (LSTM) cell

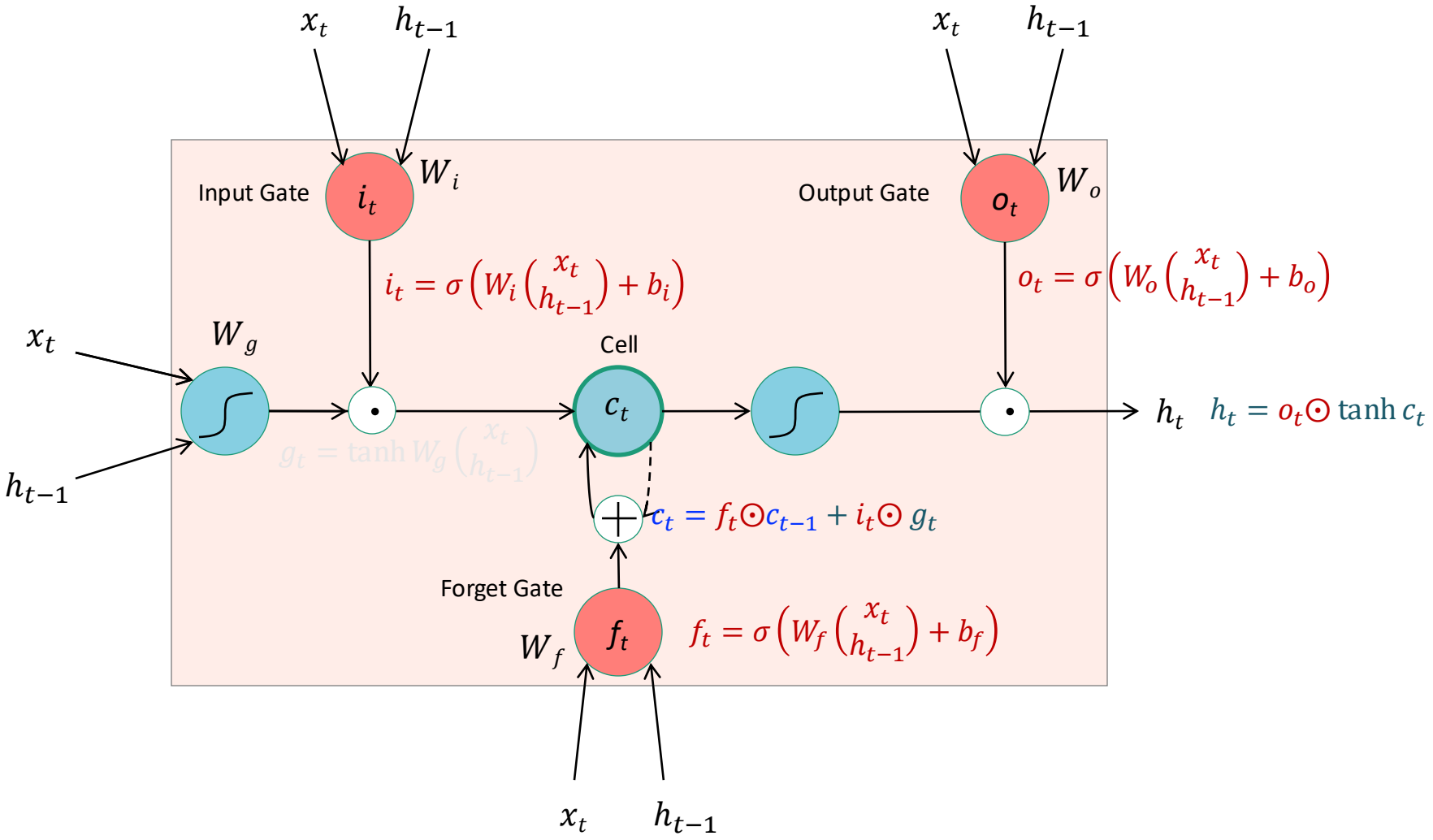


Long short-term memory (LSTM) cell



The gradients back-propagated from c_t to c_{t-1} are maintained. Thus we can do learning for *long-term*

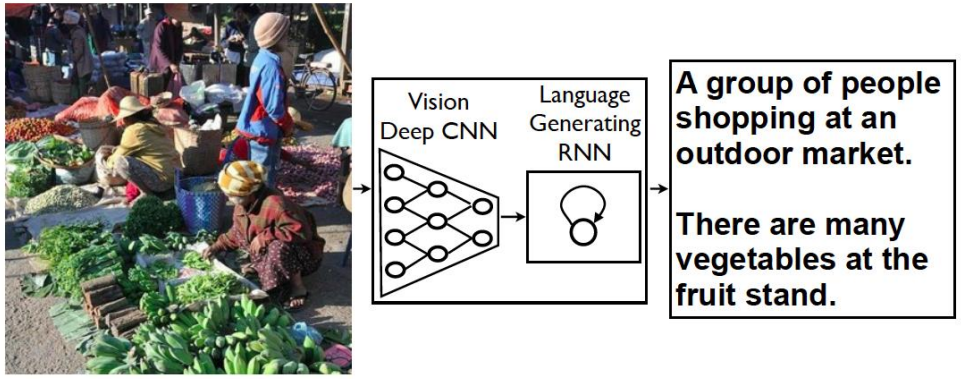
Adding a forget gate for *short-term*



The gradients back-propagated from c_t to c_{t-1} are adjusted by f_t .

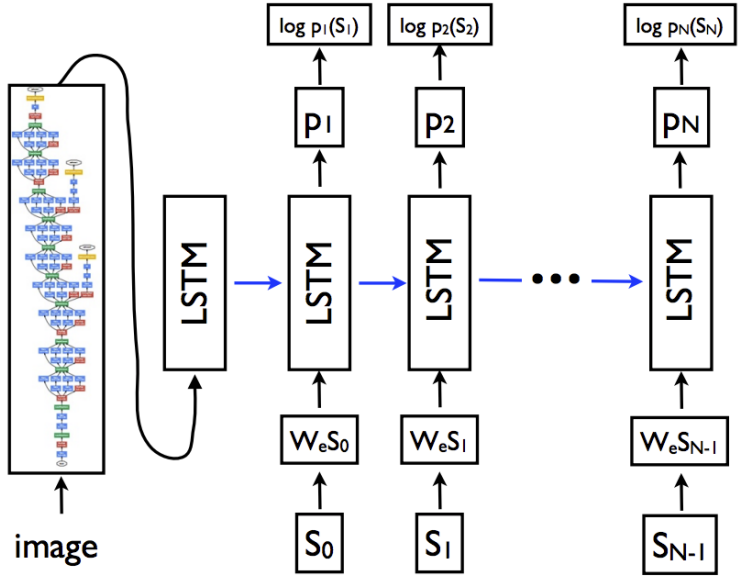
Application in language and vision tasks

Image caption generation



Training time

- Maximize likelihood of reference captions



Log-likelihood of next reference word

Softmax probability over next word

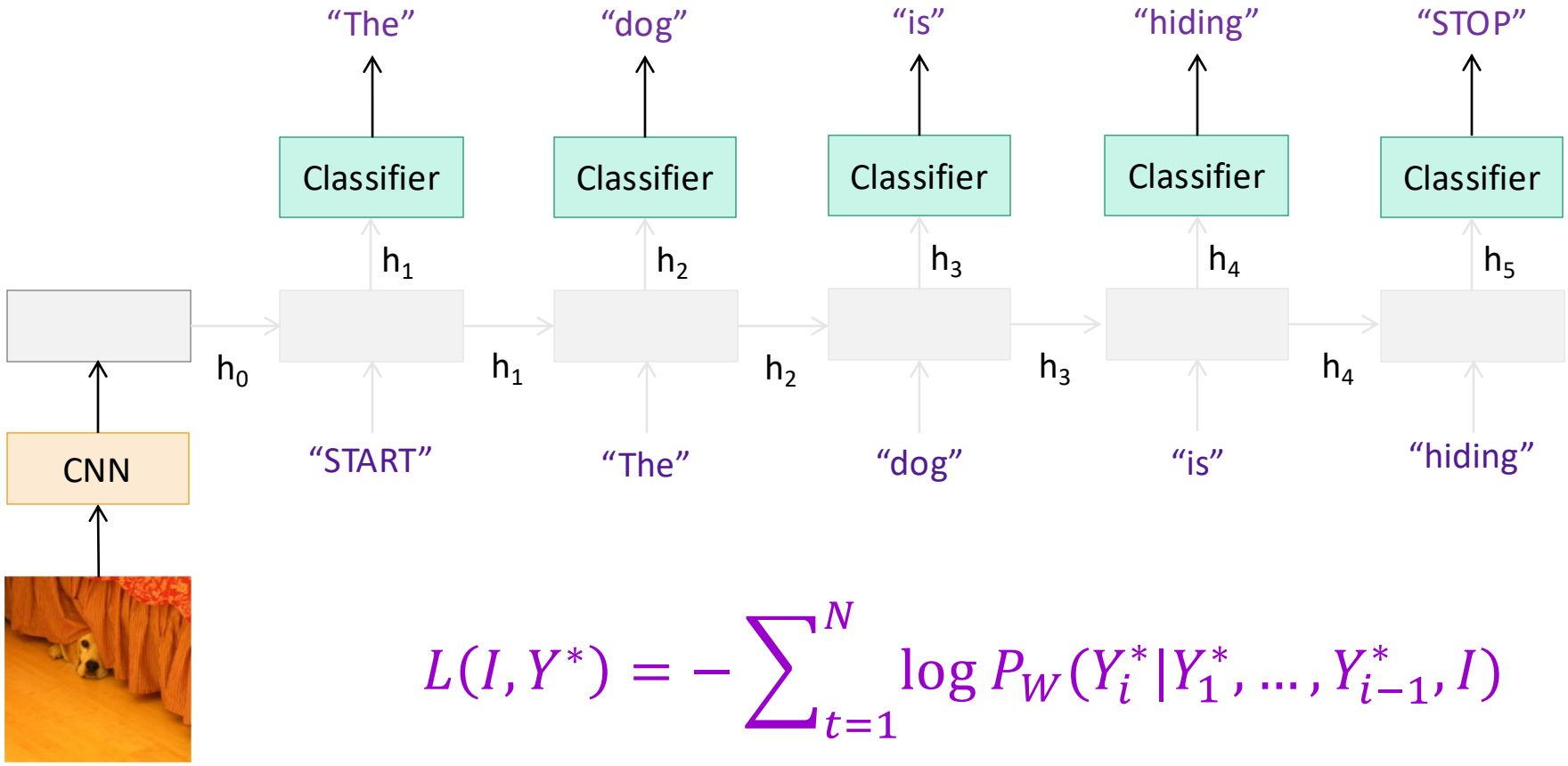
Word embedding

Words of reference caption (one-hot encoding)

Vinyals et al. 2015

Image caption generation

- Minimize negative log-likelihood of the ground truth caption $Y^* = (Y_1^*, \ldots, Y_N^*)$ given image I:



$$L(I, Y^*) = -\sum_{t=1}^{N} \log P_W(Y_i^* | Y_1^*, \ldots, Y_{i-1}^*, I)$$

Image caption generation

Visual Question Answering (VQA)

Q: What endangered animal is featured on the truck?

A: **A bald eagle.**
A: A sparrow.
A: A humming bird.
A: A raven.

Q: Where will the driver go if turning right?

A: **Onto 24 ¾ Rd.**
A: Onto 25 ¾ Rd.
A: Onto 23 ¾ Rd.
A: Onto Main Street.

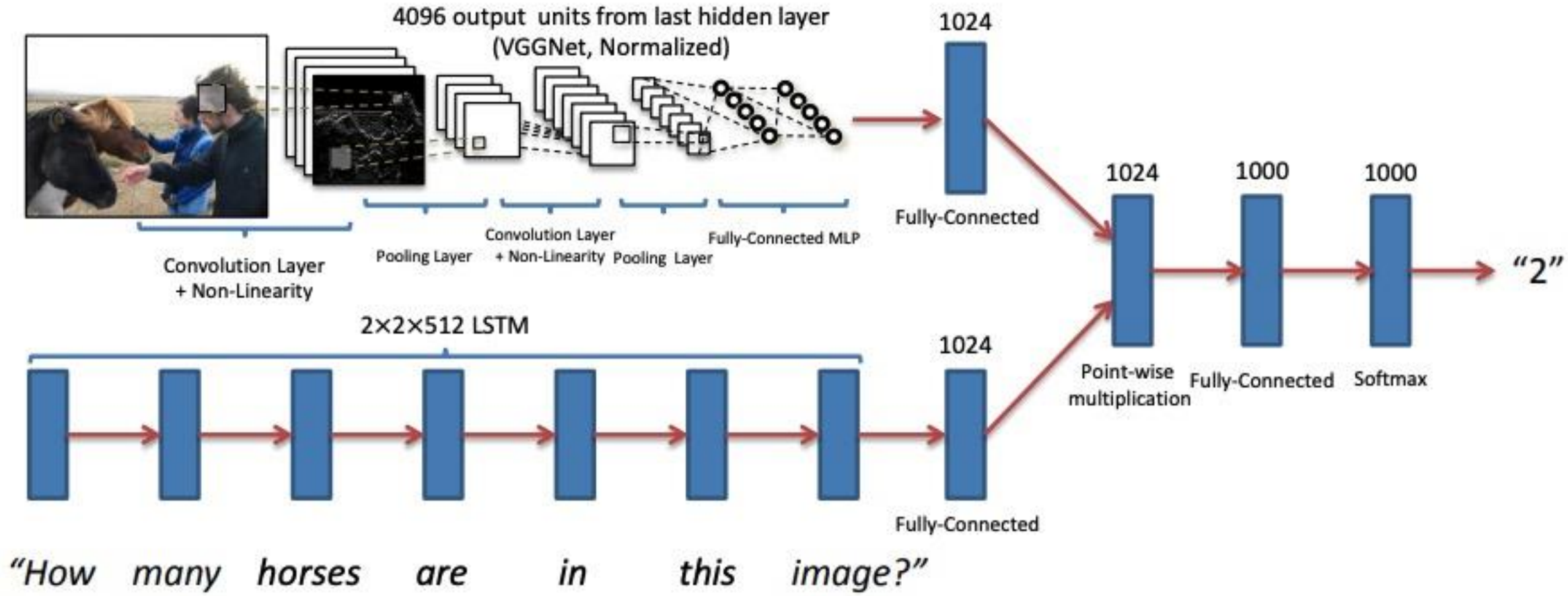
Q: When was the picture taken?

A: **During a wedding.**
A: During a bar mitzvah.
A: During a funeral.
A: During a Sunday church service

Q: Who is under the umbrella?

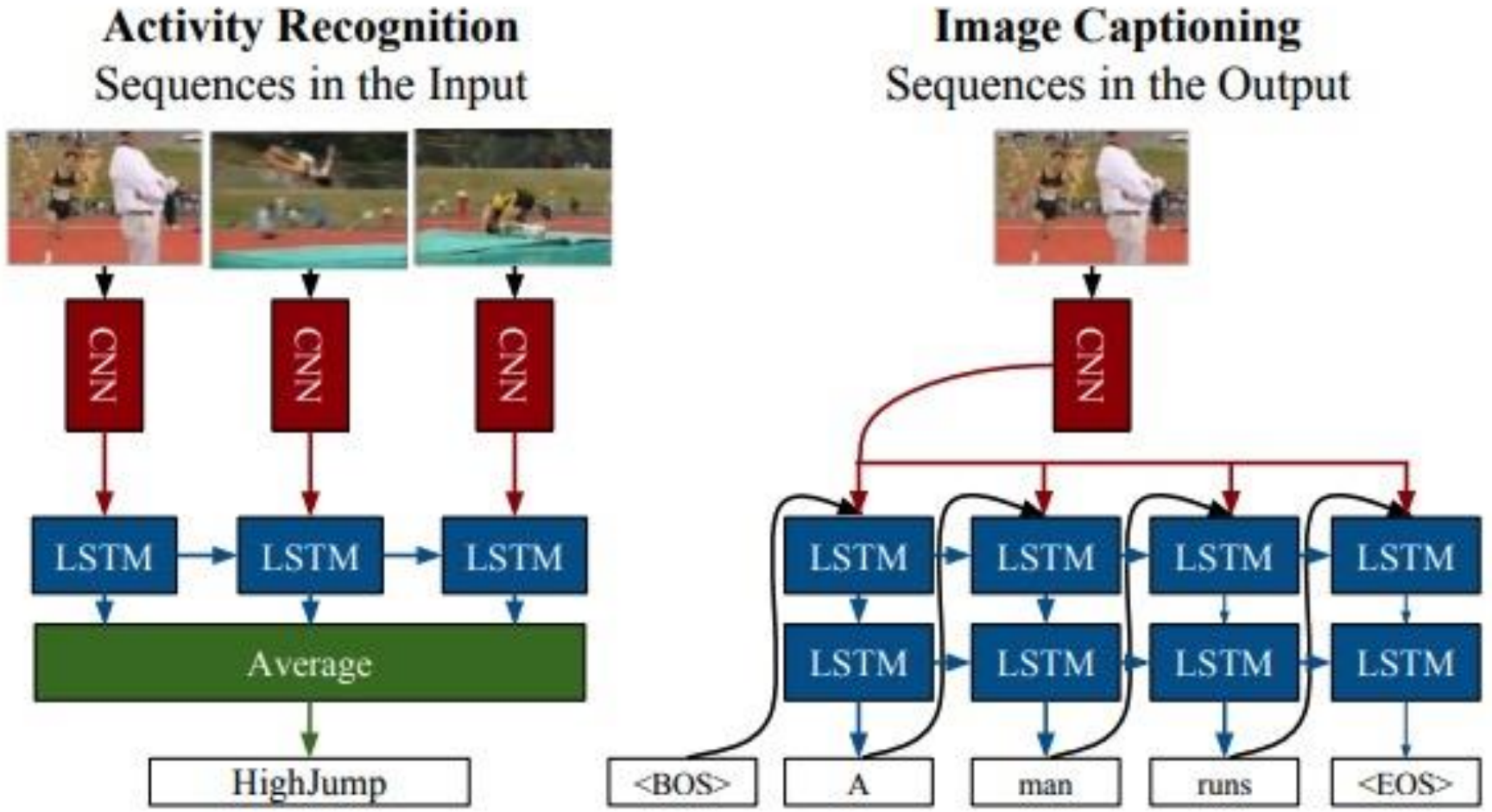
A: **Two women.**
A: A child.
A: An old man.
A: A husband and a wife.

Visual Question Answering (VQA)



Agrawal et al. 2015

Long-term Recurrent Convolutional Networks



Donahue et al. 2016

Summary

- The Basic RNN
- LSTM
- Application in language and vision tasks