



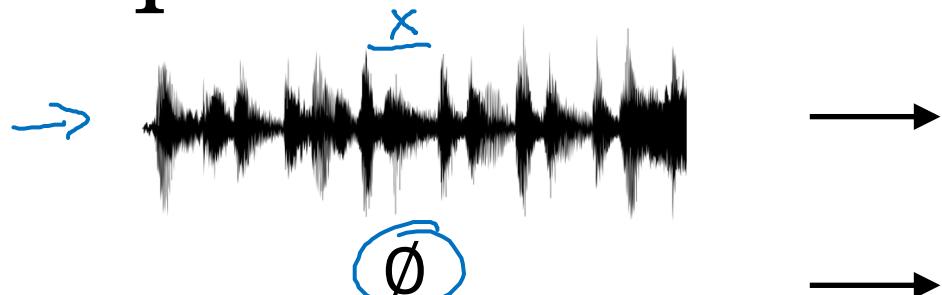
deeplearning.ai

Recurrent Neural Networks

Why sequence models?

Examples of sequence data

Speech recognition



y
“The quick brown fox jumped
over the lazy dog.”

Music generation



Sentiment classification

“There is nothing to like
in this movie.”



DNA sequence analysis → AGCCCCTGTGAGGAAC TAG



AGCCCCTGTGAGGAAC TAG

Machine translation

Voulez-vous chanter avec
moi?



Do you want to sing with
me?

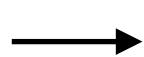
Video activity recognition



Running

Name entity recognition

Yesterday, Harry Potter
met Hermione Granger.



Yesterday, Harry Potter
met Hermione Granger.

Andrew Ng



deeplearning.ai

Recurrent Neural Networks

Notation

Motivating example

NLP

x: Harry Potter) and Hermione Granger invented a new spell.

$\rightarrow \underline{x}^{<1>} x^{<2>} x^{<3>} \dots x^{<t>} \dots x^{<9>}$

$$T_x = 9$$

$\rightarrow y:$

| | 0 | | 0 0 0 0
 $y^{<1>} y^{<2>} y^{<3>} \dots y^{<9>}$

$$T_y = 9$$

$x^{(i)<t>}$

$y^{(i)<t>}$

$$T_x^{(i)} = 9$$

15

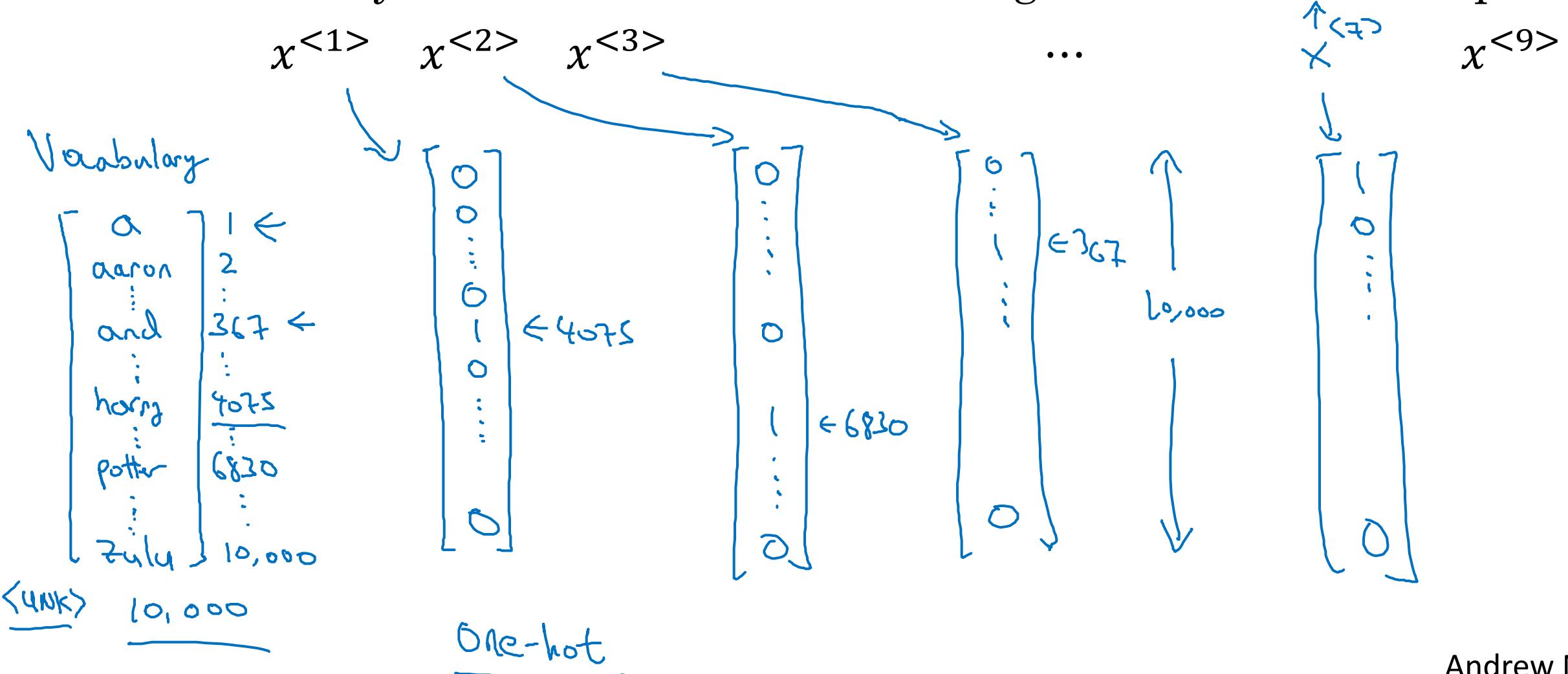
$$T_y^{(i)}$$

Representing words

$$x^{(t)} \quad (x, y)$$

$x \rightarrow y$

x: Harry Potter and Hermione Granger invented a new spell.



Representing words

x: Harry Potter and Hermione Granger invented a new spell.

$x^{<1>} \quad x^{<2>} \quad x^{<3>} \quad \dots \quad x^{<9>}$

And = 367
Invented = 4700
A = 1
New = 5976
Spell = 8376
Harry = 4075
Potter = 6830
Hermione = 4200
Gran... = 4000

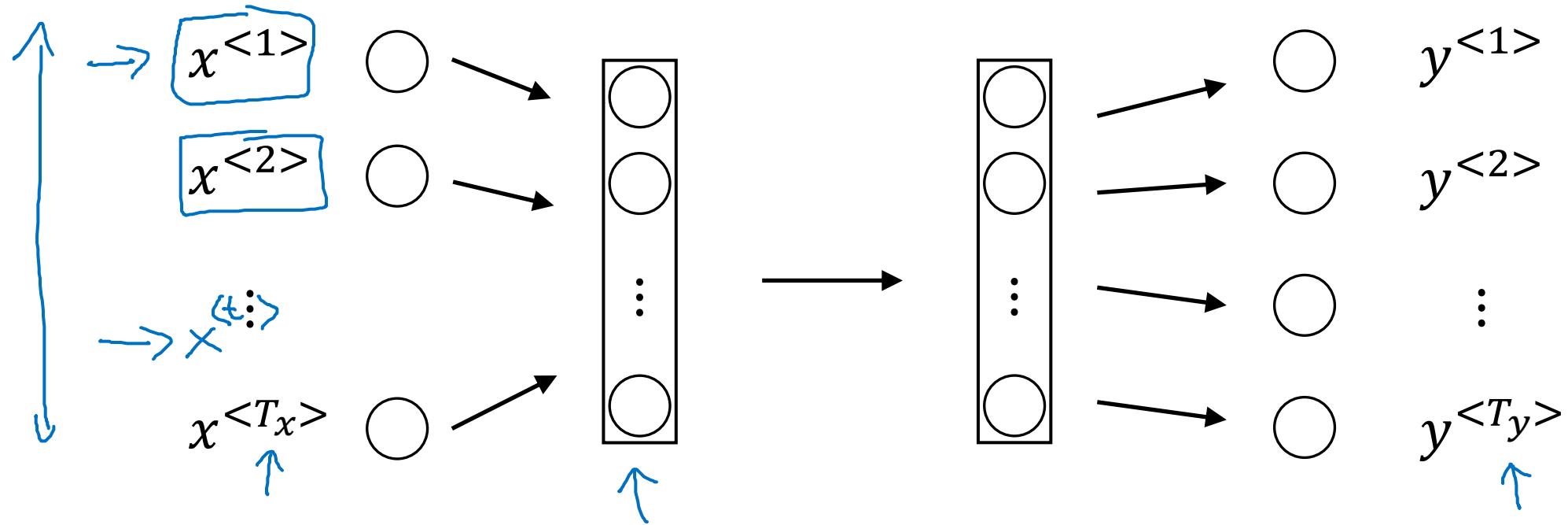


deeplearning.ai

Recurrent Neural Networks

Recurrent Neural Network Model

Why not a standard network?

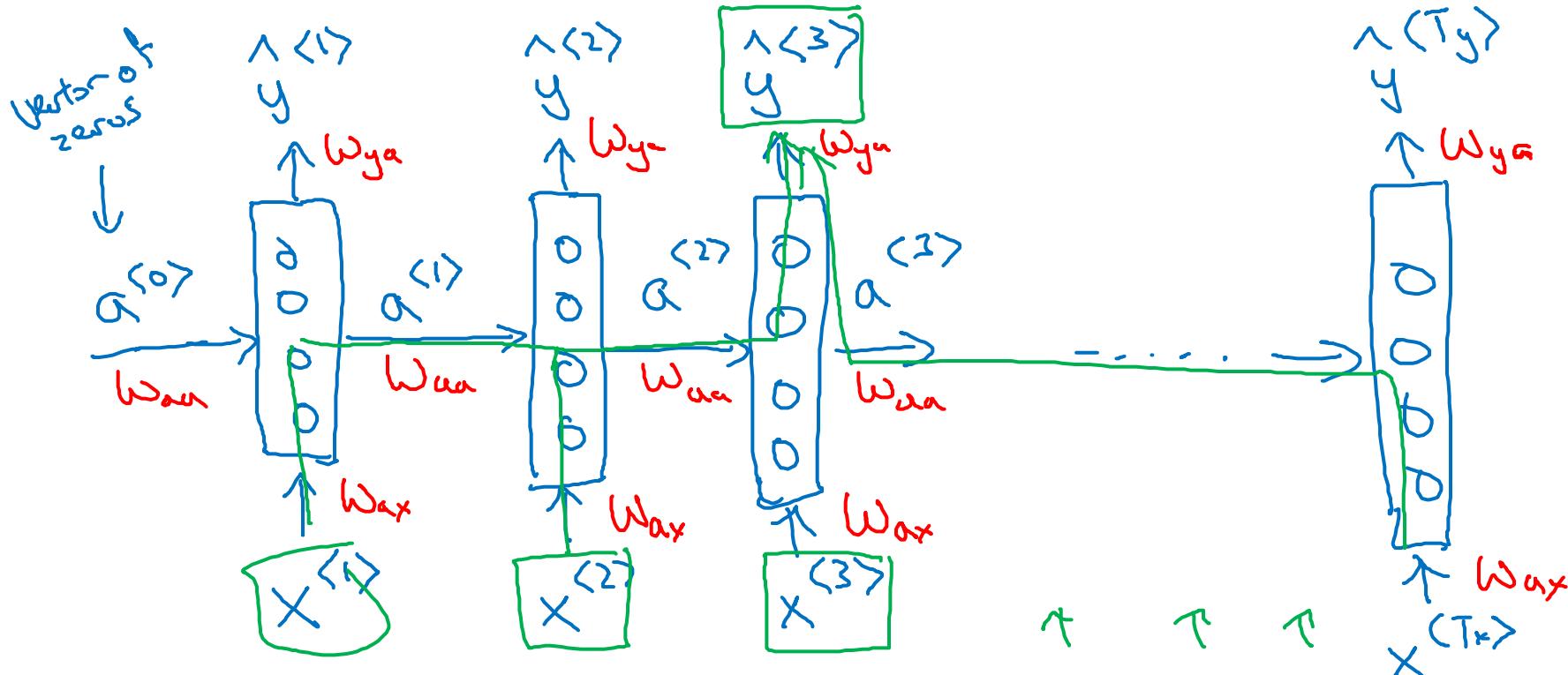


Problems:

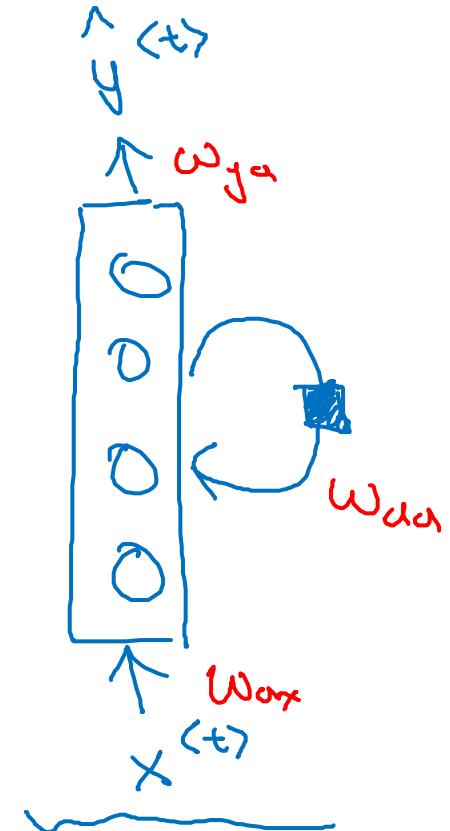
- - Inputs, outputs can be different lengths in different examples.
- - Doesn't share features learned across different positions of text.

Recurrent Neural Networks

$$\overline{T}_x = \overline{T}_y$$



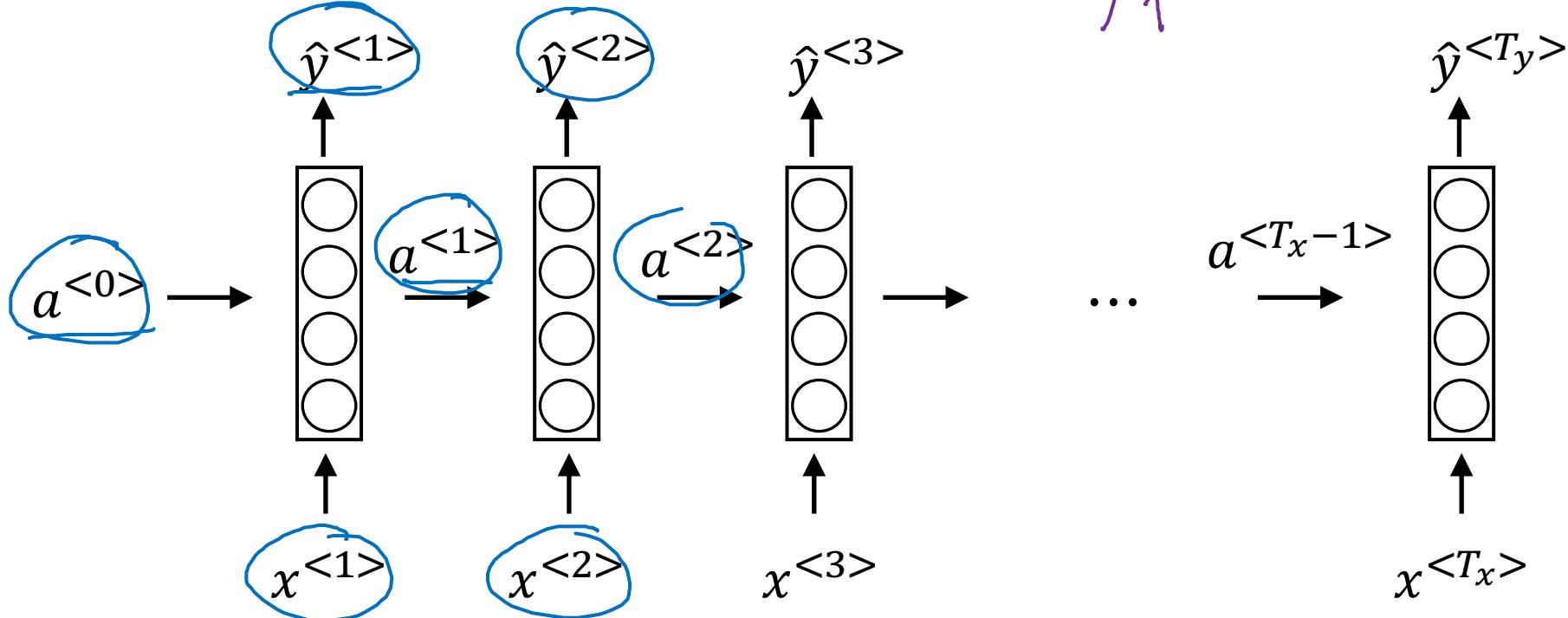
Bidirectional RNN (BRNN)



He said, "Teddy Roosevelt was a great President."

He said, "Teddy bears are on sale!"

Forward Propagation



$$a^{<0>} = \vec{0}.$$

$$\underline{a^{<t>} = g_1(W_{aa} a^{<t-1>} + \underline{W_{ax}} x^{<t>} + b_a)} \leftarrow \underline{\tanh} \text{ / ReLU}$$

$$\underline{\hat{y}^{<t>} = g_2(W_{ya} \underline{a^{<t>}} + b_y)} \leftarrow \text{Sigmoid}$$

$$a^{<t>} = g(W_{aa} a^{<t-1>} + W_{ax} x^{<t>} + b_a)$$

$$\hat{y}^{<t>} = g(W_{ya} a^{<t>} + b_y)$$

Simplified RNN notation

$$a^{(t)} = g(W_{aa}a^{(t-1)} + W_{ax}x^{(t)} + b_a)$$

Dimensions:
 W_{aa} : $(100, 100)$
 W_{ax} : $(100, 10,000)$

$$\hat{y}^{(t)} = g(W_{ya}a^{(t)} + b_y)$$

$$y^{(t)} = g(W_y a^{(t)} + b_y)$$

\uparrow \uparrow \uparrow

$$a^{(t)} = g(W_a [a^{(t-1)}, x^{(t)}] + b_a)$$

$$\begin{bmatrix} W_{aa} & W_{ax} \end{bmatrix} = W_a$$

W_{aa} : $(100, 100)$
 W_{ax} : $(100, 10,000)$

$$[a^{(t-1)}, x^{(t)}] = \begin{bmatrix} a^{(t-1)} \\ x^{(t)} \end{bmatrix}$$

$a^{(t-1)}$: 100
 $x^{(t)}$: $10,000$

$$[W_{aa}; W_{ax}] \begin{bmatrix} a^{(t-1)} \\ x^{(t)} \end{bmatrix} = W_{aa}a^{(t-1)} + W_{ax}x^{(t)}$$

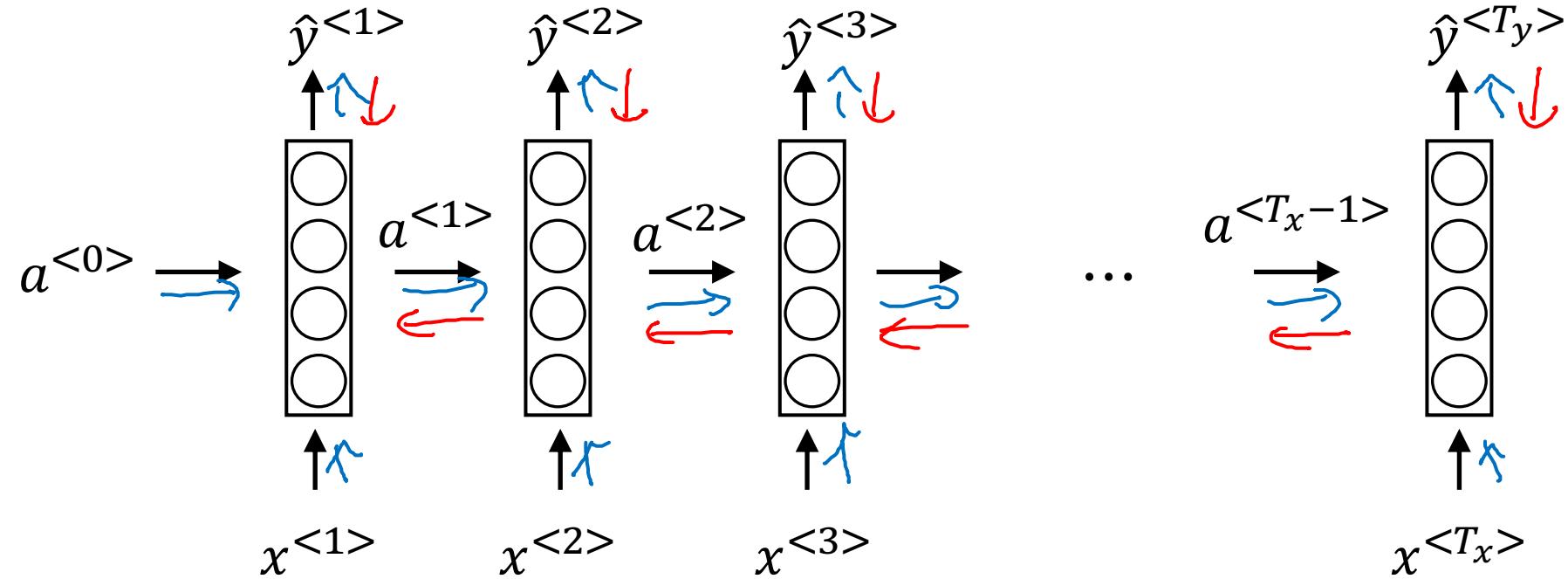


deeplearning.ai

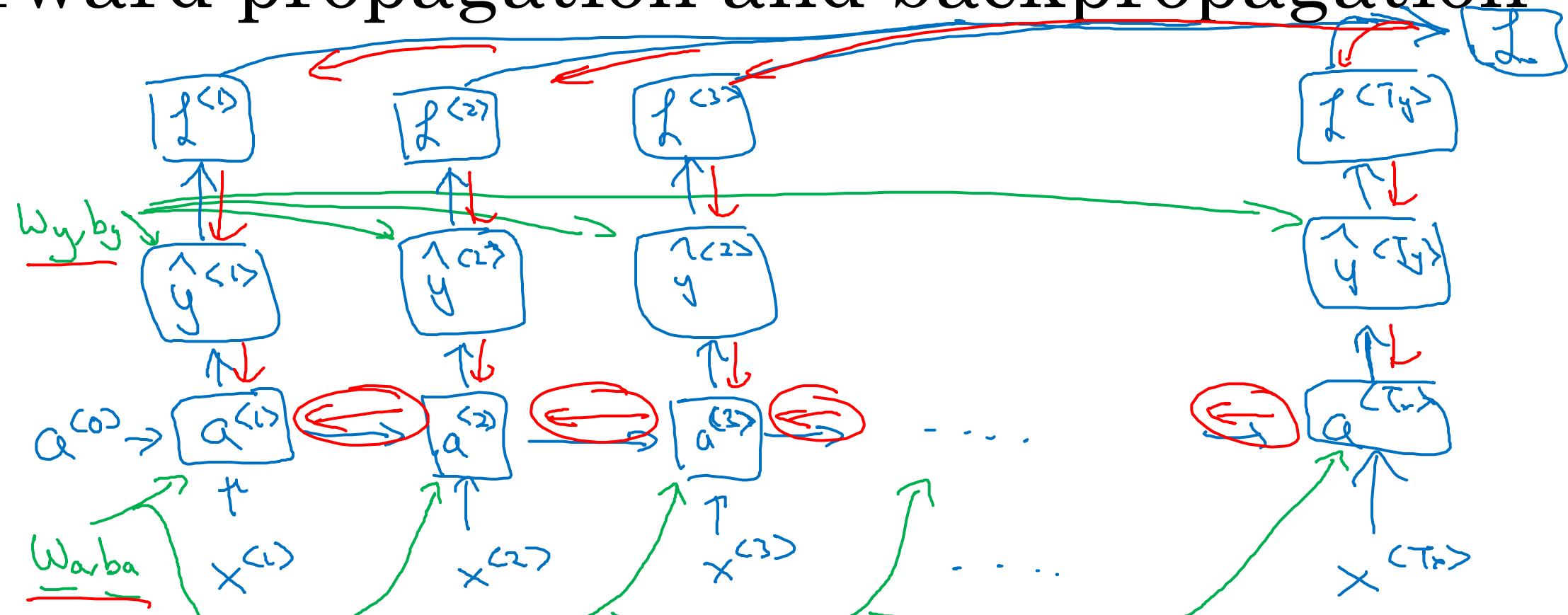
Recurrent Neural Networks

Backpropagation through time

Forward propagation and backpropagation



Forward propagation and backpropagation



$$\mathcal{L}^{(t)}(\hat{y}^{(t)}, y^{(t)}) = -y^{(t)} \log \hat{y}^{(t)} - (1-y^{(t)}) \log (1-\hat{y}^{(t)})$$

$$\mathcal{L}(\hat{y}, y) = \sum_{t=1}^{T_y} \mathcal{L}^{(t)}(\hat{y}^{(t)}, y^{(t)})$$

Backpropagation through time



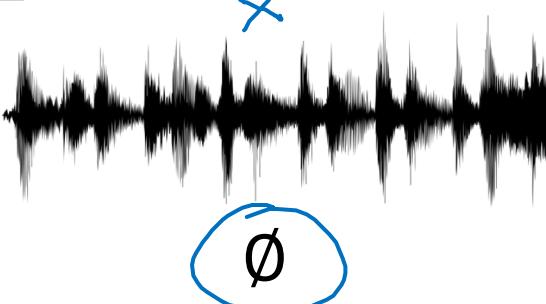
deeplearning.ai

Recurrent Neural Networks

Different types of RNNs

Examples of sequence data

Speech recognition



$$\begin{matrix} T_x \\ \xrightarrow{\hspace{1cm}} \\ T_y \end{matrix}$$

A blue letter y is positioned above the output sequence.

“The quick brown fox jumped over the lazy dog.”

Music generation



Sentiment classification

“There is nothing to like
in this movie.”



DNA sequence analysis

AGCCCCTGTGAGGAAC TAG



AG~~CCCCTGTGAGGAAC~~ TAG

Machine translation

Voulez-vous chanter avec
moi?



Do you want to sing with
me?

Video activity recognition



Running

Name entity recognition

Yesterday, Harry Potter
met Hermione Granger.

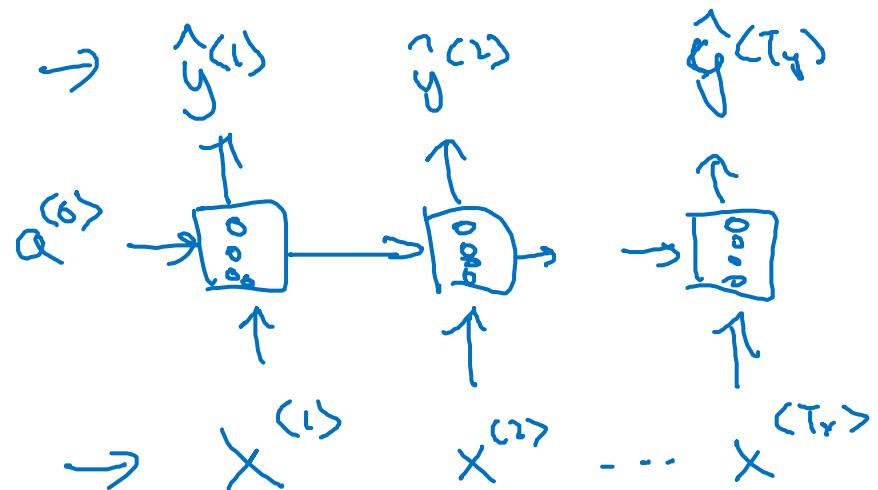


Yesterday, **Harry Potter**
met **Hermione Granger**.

Andrew Ng

Examples of RNN architectures

$$T_x = T_y$$

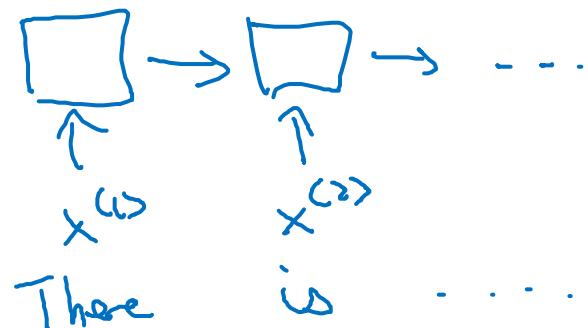


Many-to-many

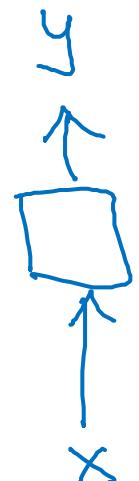
Sentiment classification

$x = \text{text}$

$y = 0/1 \quad 1 \dots 5$

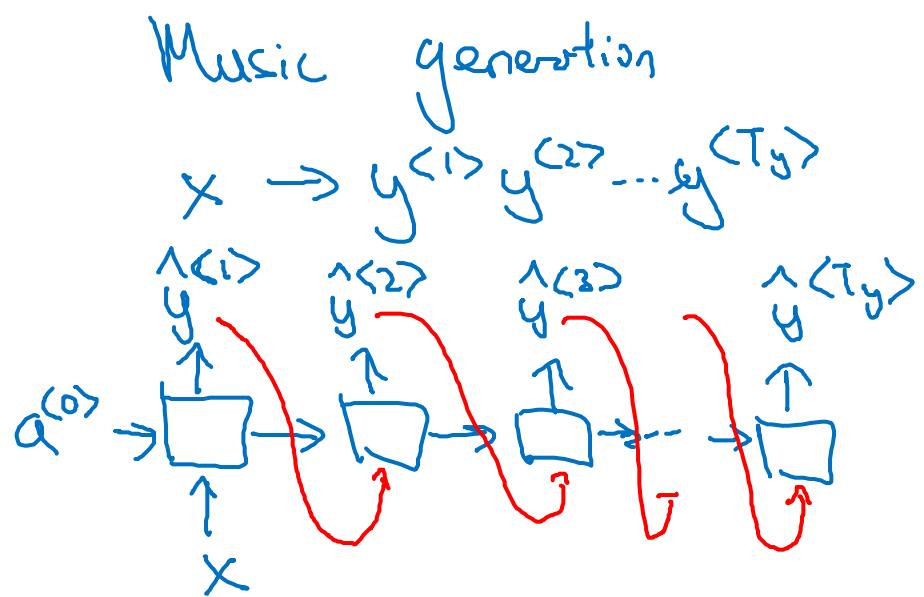


Many-to-one



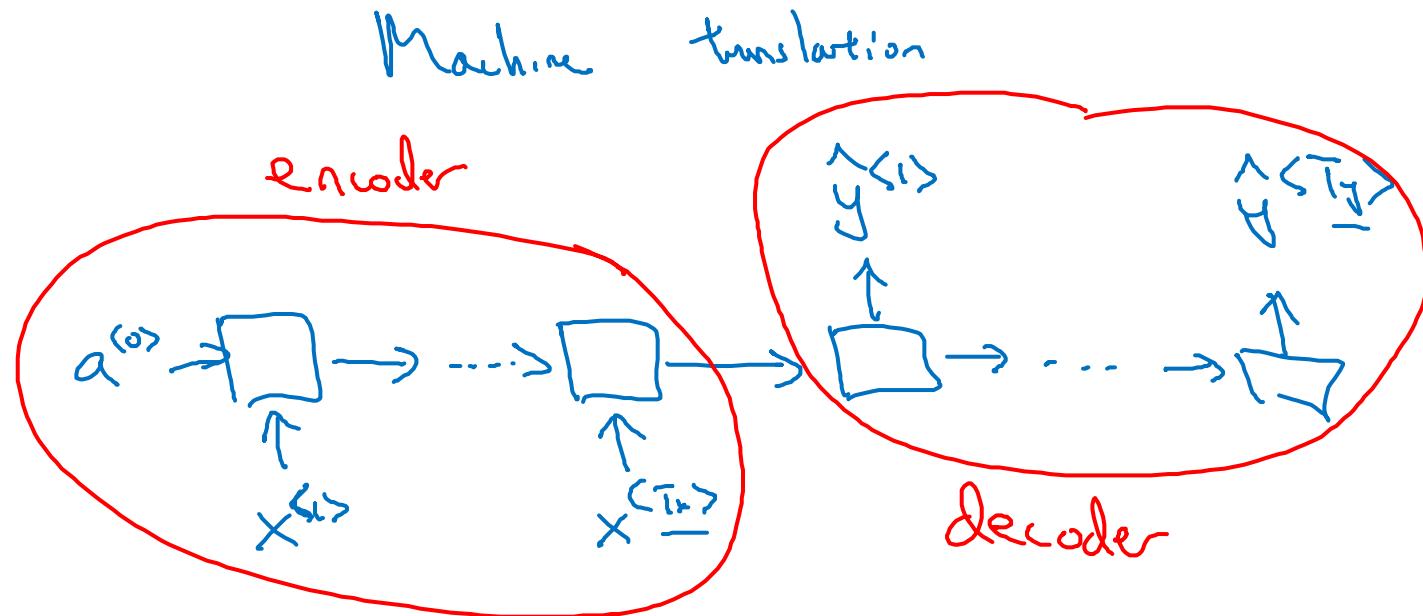
One-to-one

Examples of RNN architectures



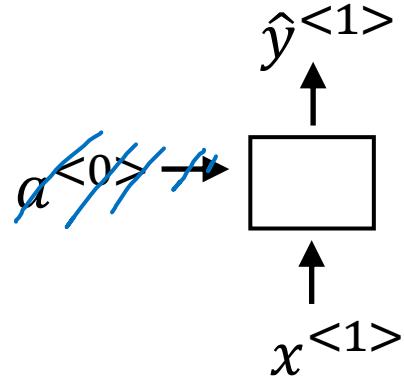
One-to-many

$$x = \phi$$

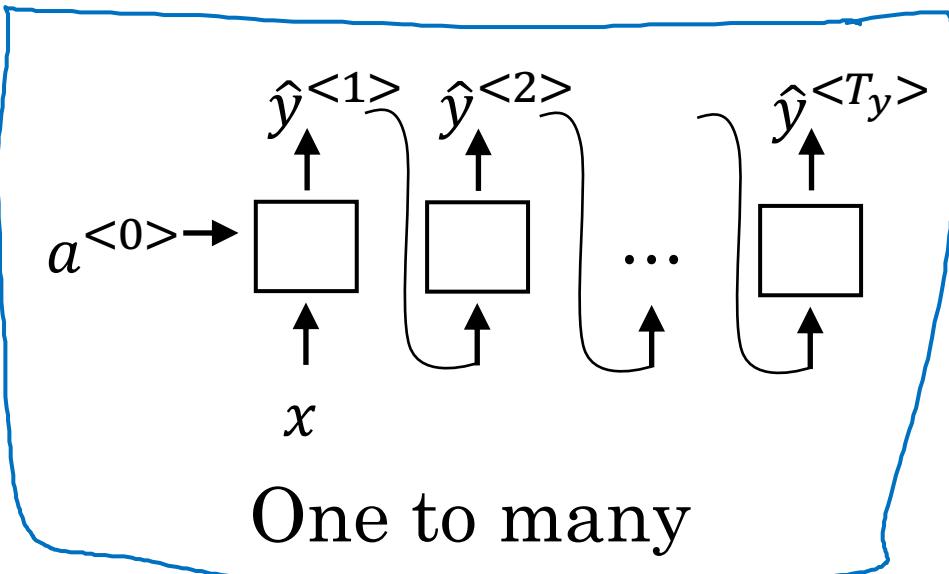


Many - to - many

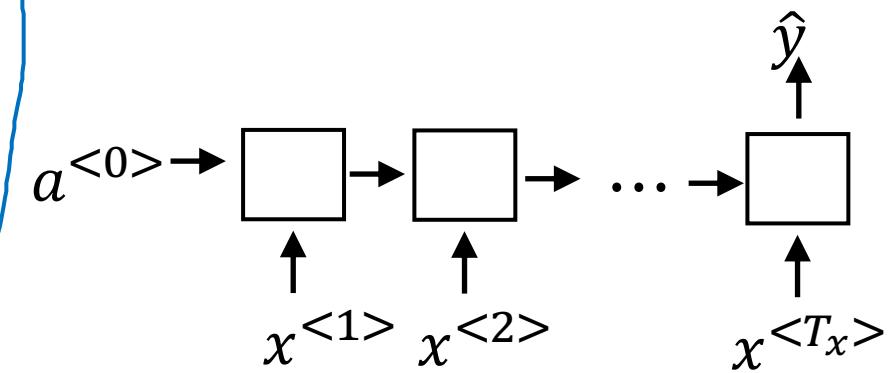
Summary of RNN types



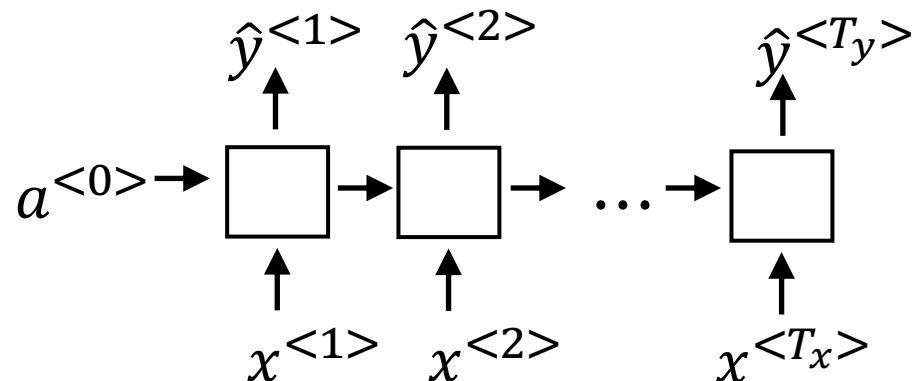
One to one



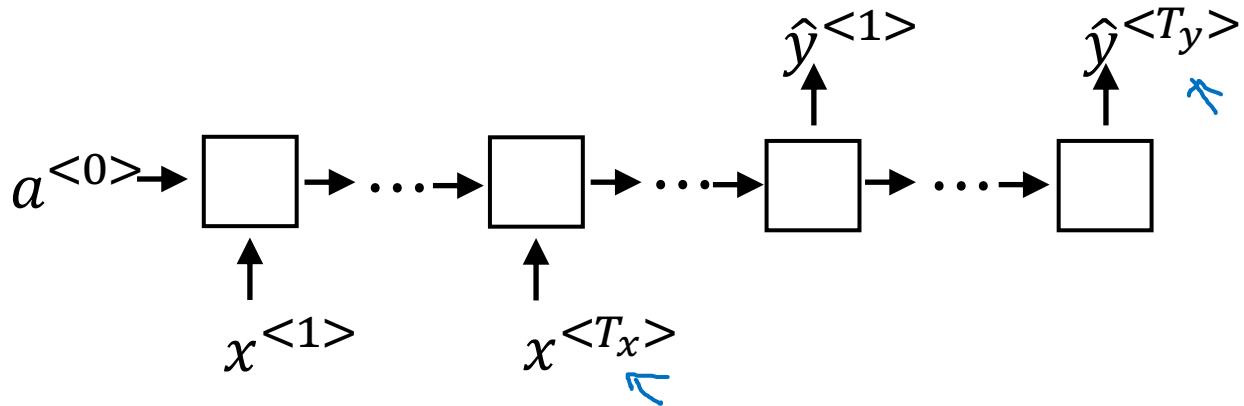
One to many



Many to one



Many to many
 $T_x = T_y$



Many to many



deeplearning.ai

Recurrent Neural Networks

Language model and
sequence generation

What is language modelling?

Speech recognition

The apple and pair salad.

→ The apple and pear salad.

$$P(\text{The apple and pair salad}) = 3.2 \times 10^{-3}$$

$$P(\text{The apple and pear salad}) = 5.7 \times 10^{-10}$$

$$P(\text{Sentence}) = ?$$

$$P(y^{(1)}, y^{(2)}, \dots, y^{(T)})$$

Language modelling with an RNN

Training set: large corpus of english text.

Tokenize

Cats average 15 hours of sleep a day. $\downarrow <\text{EOS}>$

$y^{<1>}$ $y^{<2>}$ $y^{(3)}$

$x^{<t>} = y^{<t-1>}$

...

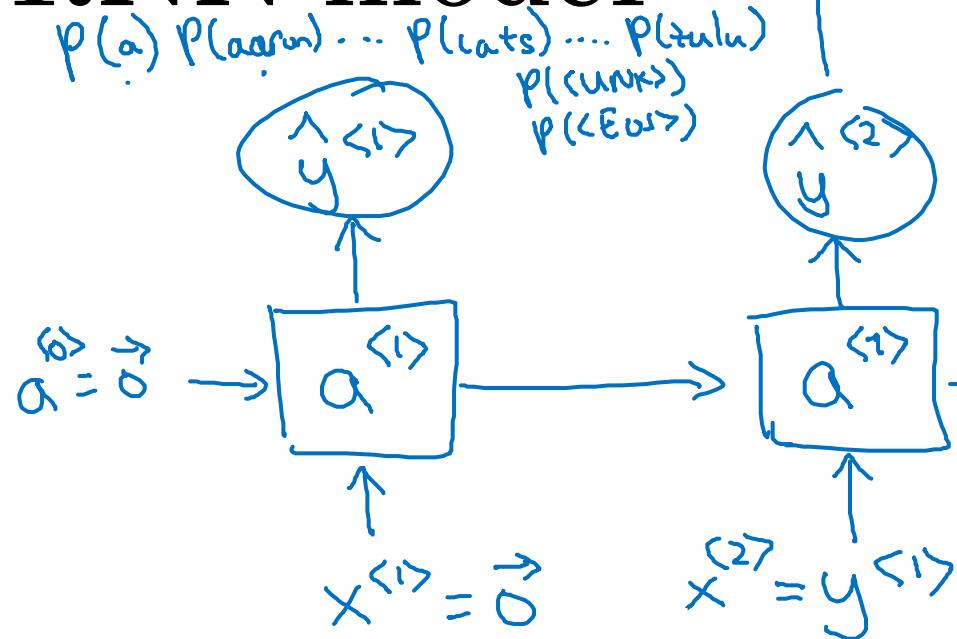
$y^{(8)}$ $y^{(9)}$

The Egyptian ~~Mau~~ is a bread of cat. $<\text{EOS}>$

$<\text{UNK}>$

10,000

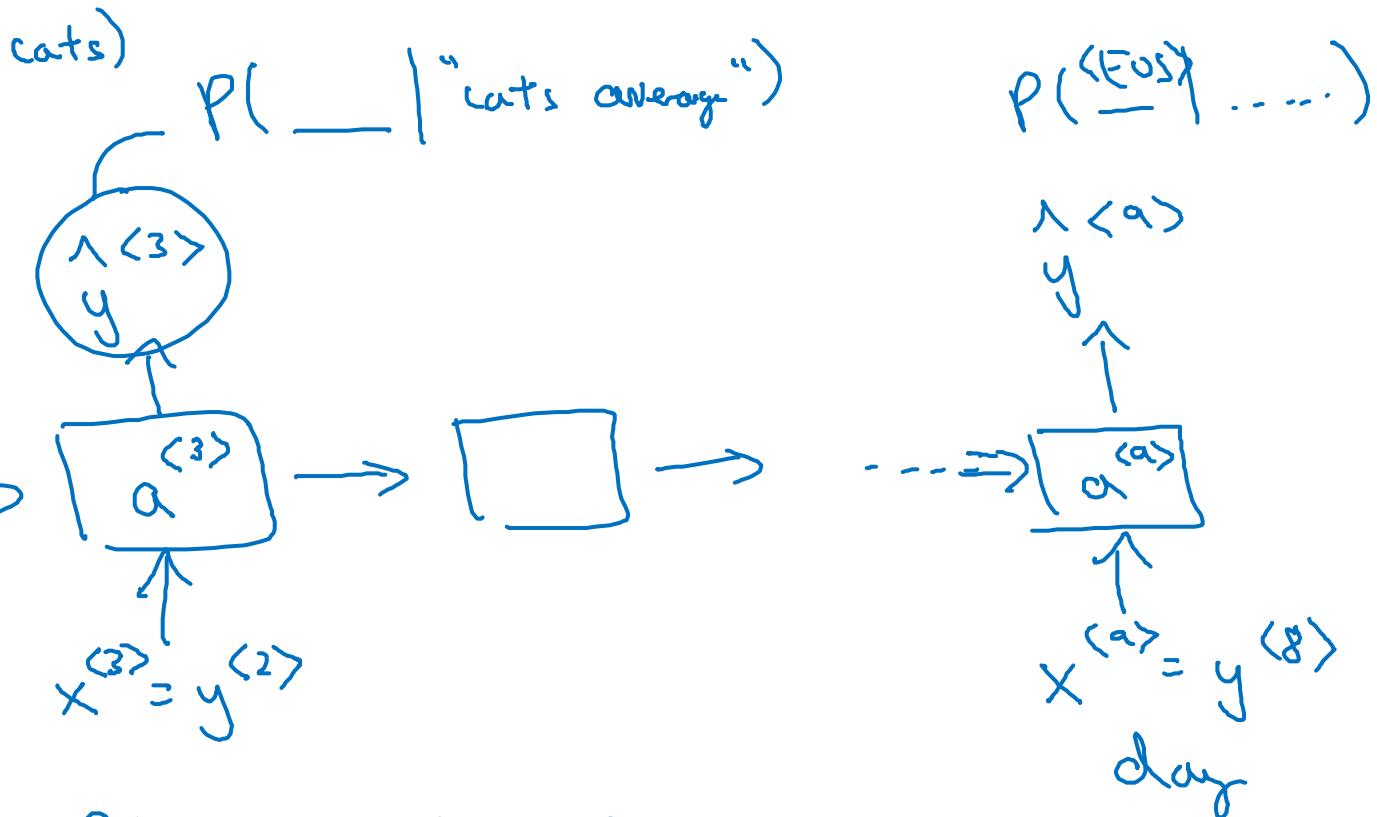
RNN model



Cats
 → Cats average 15 hours of sleep a day. $\langle \text{EOS} \rangle$

$$\mathcal{L}(\hat{y}^{(t)}, y^{(t)}) = - \sum_i y_i^{(t)} \log \hat{y}_i^{(t)}$$

$$\mathcal{L} = \sum_t \mathcal{L}^{(t)}(\hat{y}^{(t)}, y^{(t)})$$



$$P(y^{(1)}, y^{(2)}, y^{(3)}) \leftarrow \\ = \frac{P(y^{(1)}) P(y^{(2)} | y^{(1)})}{P(y^{(3)} | y^{(1)}, y^{(2)})}$$

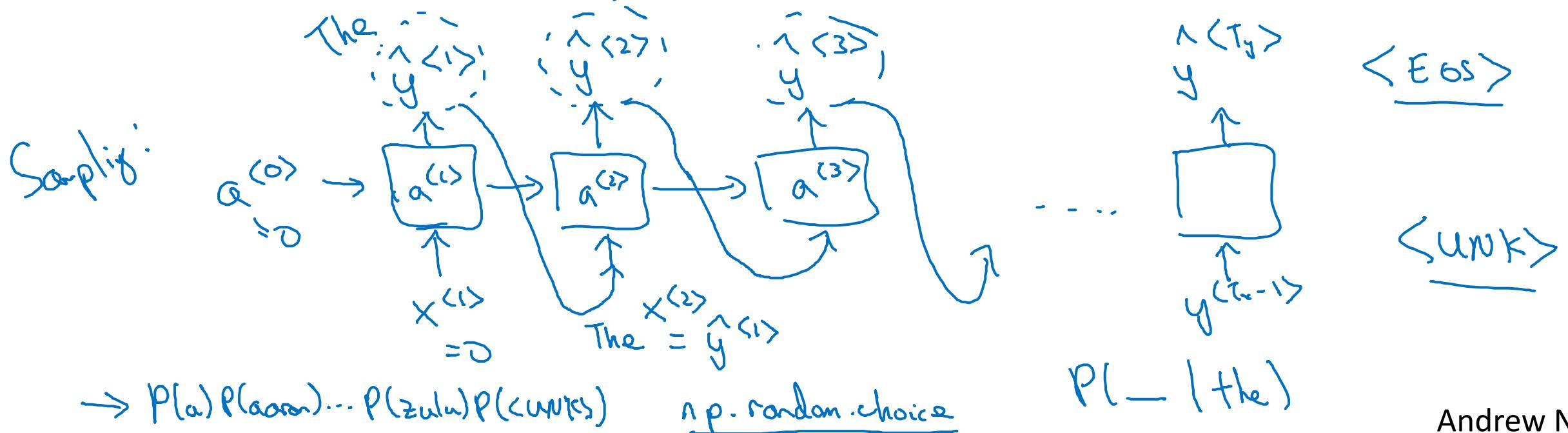
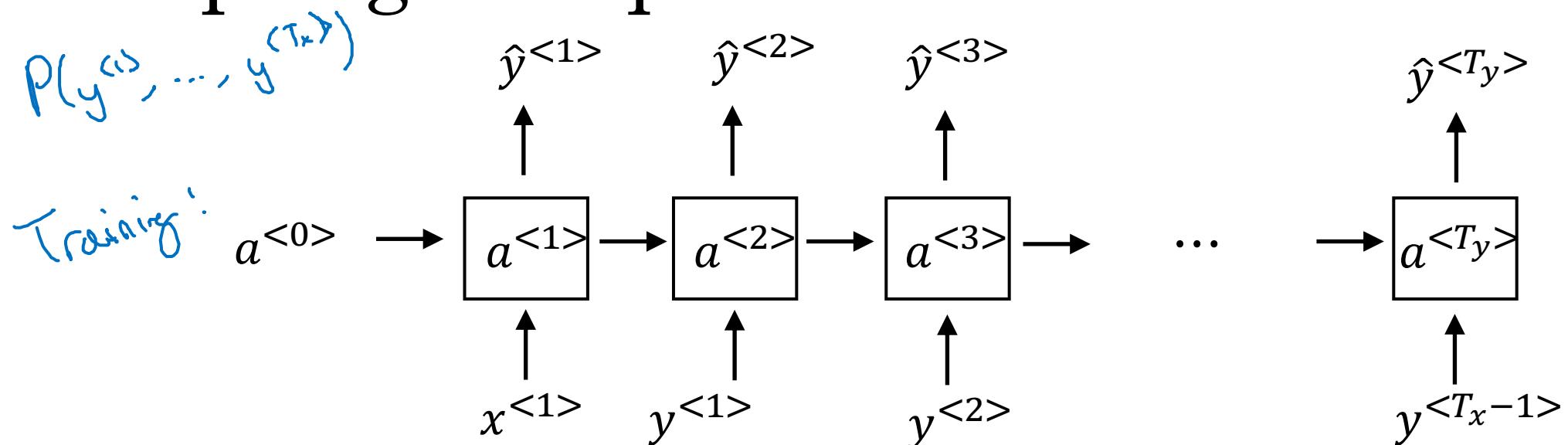


deeplearning.ai

Recurrent Neural Networks

Sampling novel
sequences

Sampling a sequence from a trained RNN



Character-level language model

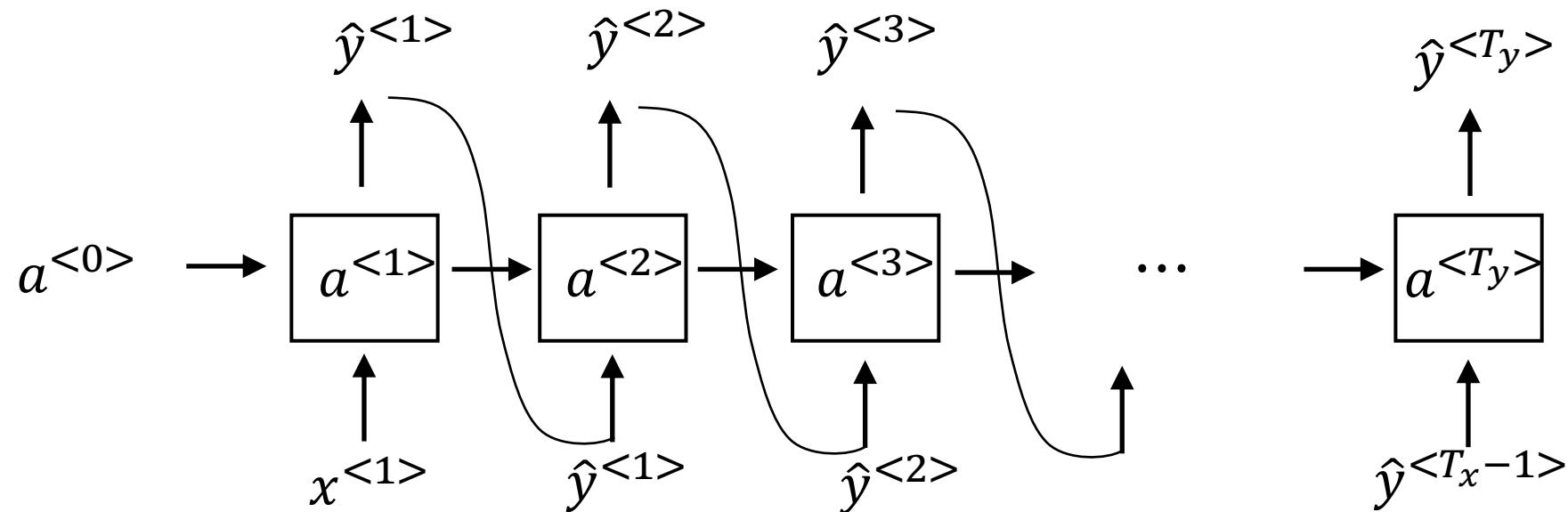
→ Vocabulary = [a, aaron, ..., zulu, <UNK>] ↪

$\rightarrow \text{Vocabulary} = [a, b, c, \dots, z, \cup, \circ, \rightarrow, ;, 0, \dots, 9, A, \dots, Z]$

$$y^{(0)} = y^{(1)} = y^{(2)} = y^{(3)}$$

Cat average
↑ ↑ ↑ ↑ . . .

May



Sequence generation

News

President enrique peña nieto, announced
sench's sulk former coming football langston
paring.

“I was not at all surprised,” said hich langston.

“Concussion epidemic”, to be examined. ←

The gray football the told some and this has on
the uefa icon, should money as.

Shakespeare

The mortal moon hath her eclipse in love.
And subject of this thou art another this fold.
When lesser be my love to me see sabl's.
For whose are ruse of mine eyes heaves.

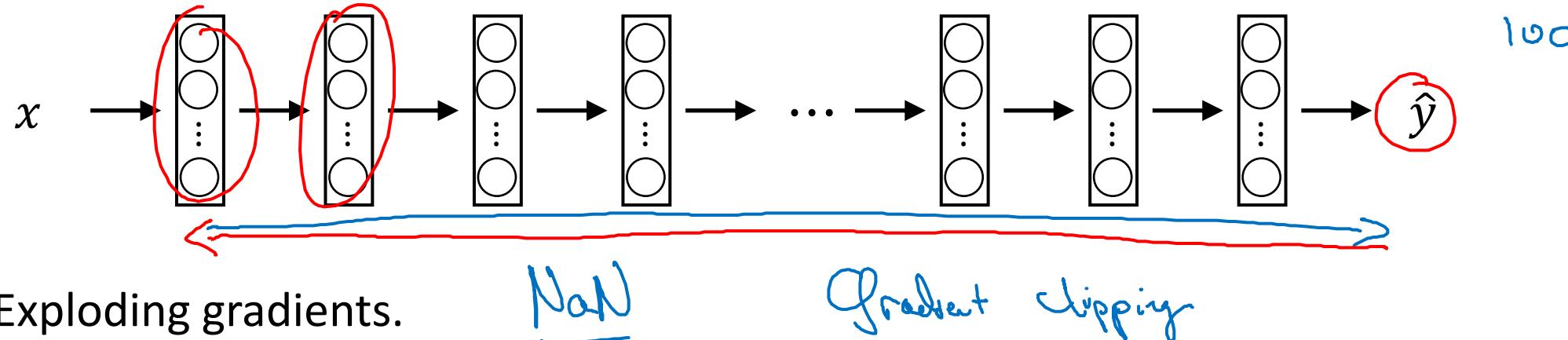
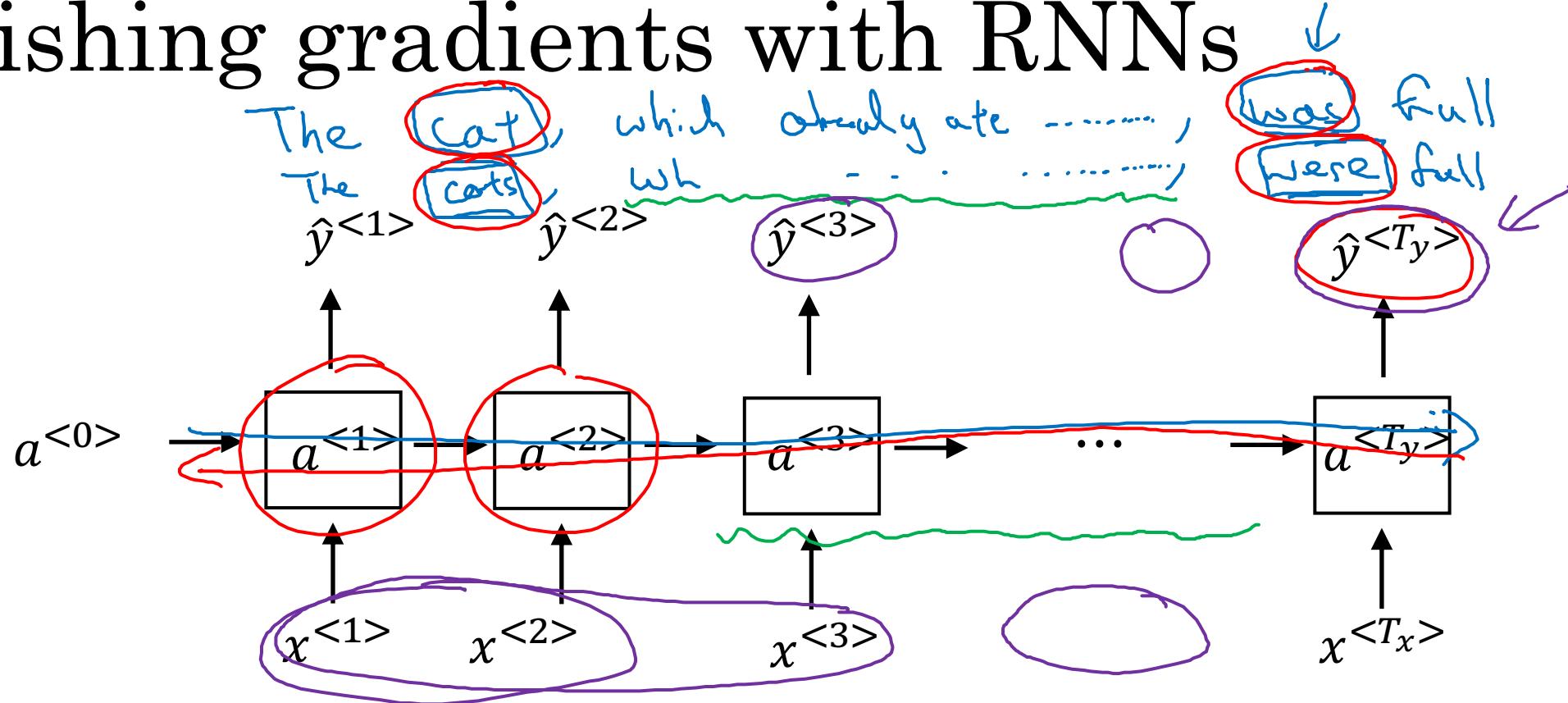


deeplearning.ai

Recurrent Neural Networks

Vanishing gradients with RNNs

Vanishing gradients with RNNs



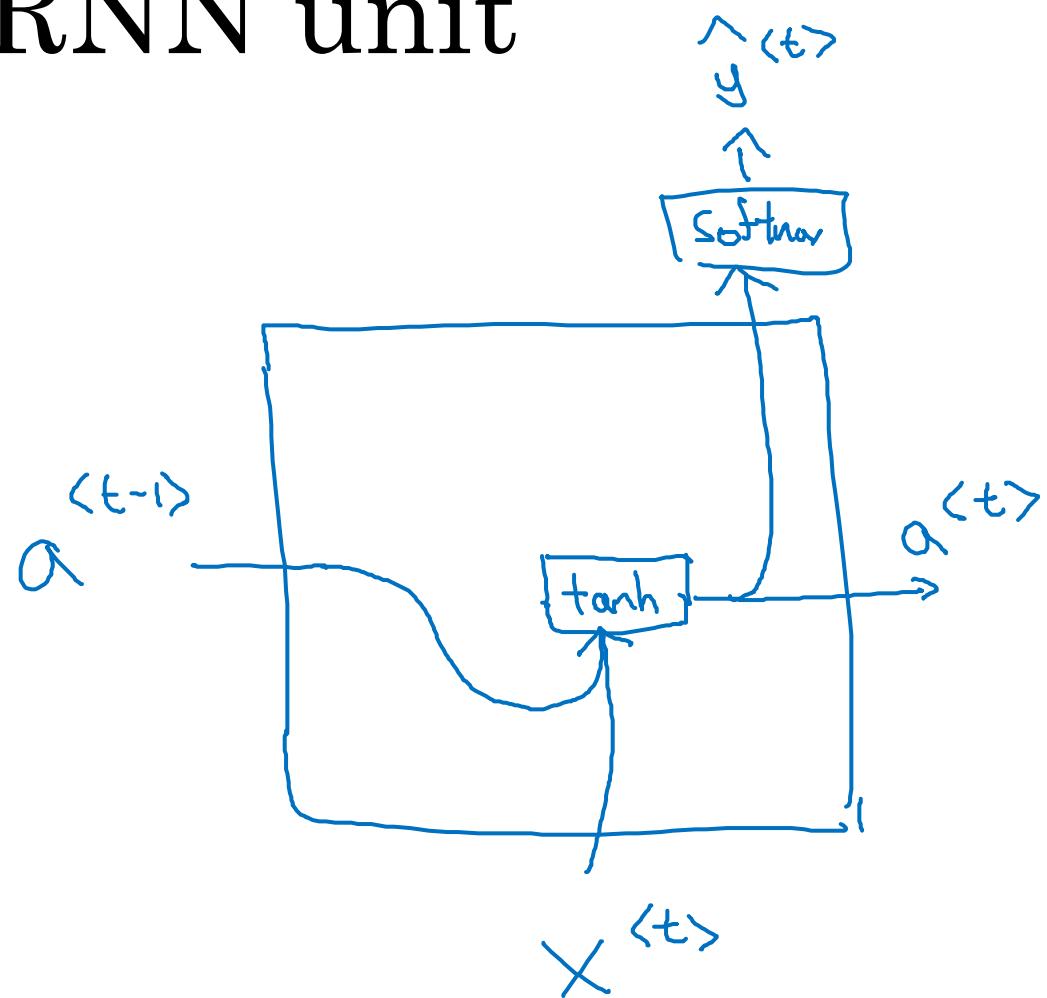


deeplearning.ai

Recurrent Neural Networks

Gated Recurrent Unit (GRU)

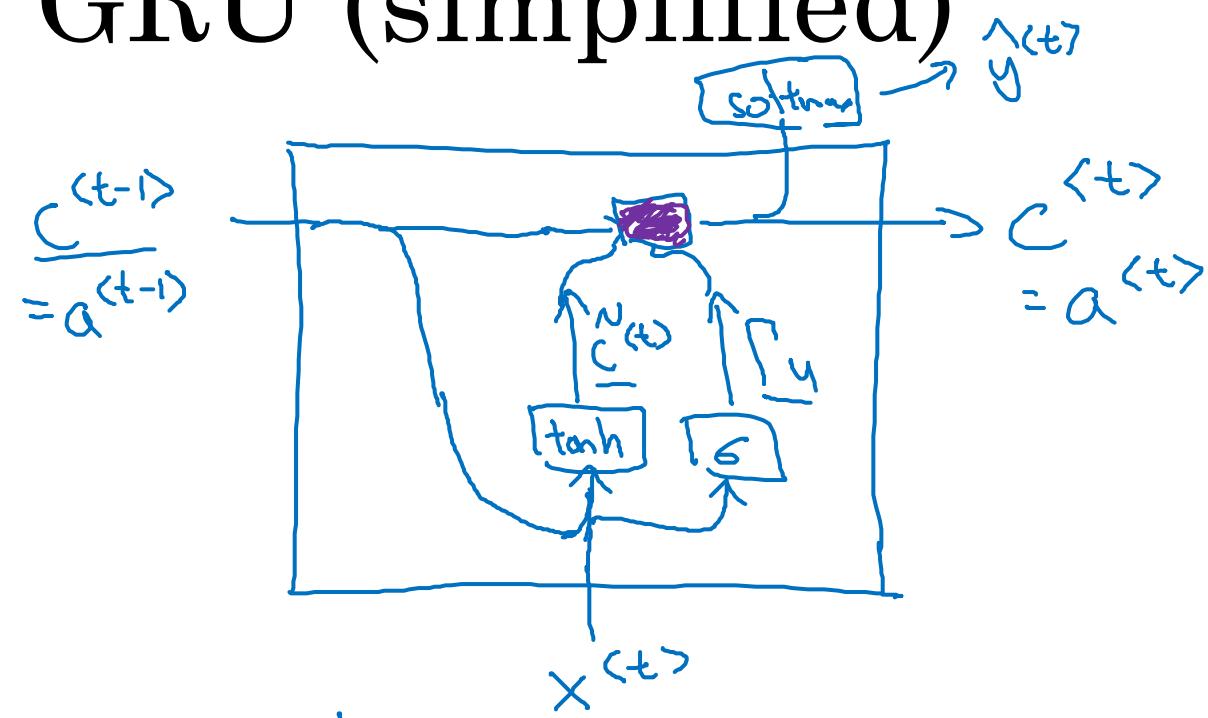
RNN unit



$$\underline{a}^{(t)} = g(W_a[\underline{a}^{(t-1)}, \underline{x}^{(t)}] + b_a)$$

Below the equation, there is a wavy line representing the hidden state $a^{(t)}$. A blue bracket under the wavy line indicates its dimensionality, and a blue arrow points upwards from the center of the bracket towards the equation, labeled "tanh".

GRU (simplified)



The diagram illustrates the internal structure of an LSTM cell. At the top, a green oval represents the cell state, and a blue oval represents the hidden state. Below, the cell is labeled "C = memory cell". The cell state $C^{<t>}$ is shown as a red box with an arrow pointing to it. The hidden state $\underline{a}^{<t>}$ is also shown as a red box. The cell's output is $N^{<t>} = \tanh(C^{<t>})$, and the hidden state is $y^{<t>} = \sigma(C^{<t>})$. The cell state is updated via element-wise multiplication (\times) with the previous cell state $C^{<t-1>}$ using a forget gate f_u (green box). A red arrow labeled "update" points to the multiplication operation. The forget gate calculation is $f_u = \sigma(W_u [C^{<t-1>}, x^{<t>}] + b_u)$. The input $x^{<t>}$ is shown as a blue box. The final cell state $C^{<t>}$ is calculated as $C^{<t>} = f_u \times N^{<t>} + (1-f_u) \times C^{<t-1>}$. A red arrow labeled "element-wise" points to the multiplication of f_u and $N^{<t>}$. A value $f_u = 0.000001$ is written next to the forget gate. At the bottom, a grid of arrows shows the flow of information from the previous hidden state $\underline{a}^{<t-1>}$ through the cell state C to produce the current hidden state $\underline{a}^{<t>}$.

[Cho et al., 2014. On the properties of neural machine translation: Encoder-decoder approaches] ↵

[Chung et al., 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling] ↵

Andrew Ng

Full GRU

$$\tilde{c}^{<t>} = \tanh(W_c[\tilde{c}_r^{<t-1>}, x^{<t>}] + b_c)$$

$$u \quad \Gamma_u = \sigma(W_u[c^{<t-1>}, x^{<t>}] + b_u)$$

$$r \quad \Gamma_r = \sigma(W_r[c^{<t-1>}, x^{<t>}] + b_r)$$

LSTM

$$h \quad c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + (1 - \Gamma_u) * c^{<t-1>}$$

The cat, which ate already, was full.



deeplearning.ai

Recurrent Neural Networks

LSTM (long short
term memory) unit

GRU and LSTM

GRU

$$\tilde{c}^{<t>} = \tanh(W_c[\Gamma_r * \underline{c}^{<t-1>}, x^{<t>}] + b_c)$$

$$\underline{\Gamma_u} = \sigma(W_u[c^{<t-1>}, x^{<t>}] + b_u)$$

$$\underline{\Gamma_r} = \sigma(W_r[c^{<t-1>}, x^{<t>}] + b_r)$$

$$c^{<t>} = \underline{\Gamma_u} * \tilde{c}^{<t>} + (1 - \underline{\Gamma_u}) * c^{<t-1>} \quad \text{(output)}$$

Γ_u

$$a^{<t>} = c^{<t>} \quad \Gamma_f$$

LSTM

$$\tilde{c}^{<t>} = \tanh(W_c[a^{<t-1>}, x^{<t>}] + b_c)$$

$$\underline{\Gamma_u} = \sigma(W_u[a^{<t-1>}, x^{<t>}] + b_u) \quad \text{(update)}$$

$$\underline{\Gamma_f} = \sigma(W_f[a^{<t-1>}, x^{<t>}] + b_f) \quad \text{(forget)}$$

$$\underline{\Gamma_o} = \sigma(W_o[a^{<t-1>}, x^{<t>}] + b_o) \quad \text{(output)}$$

$$c^{<t>} = \underline{\Gamma_u} * \tilde{c}^{<t>} + \underline{\Gamma_f} * \underline{c}^{<t-1>}$$

$$a^{<t>} = \underline{\Gamma_o} * c^{<t>}$$

LSTM in pictures

$$\tilde{c}^{<t>} = \tanh(W_c[a^{<t-1>}, x^{<t>}] + b_c)$$

$$\Gamma_u = \sigma(W_u[a^{<t-1>}, x^{<t>}] + b_u)$$

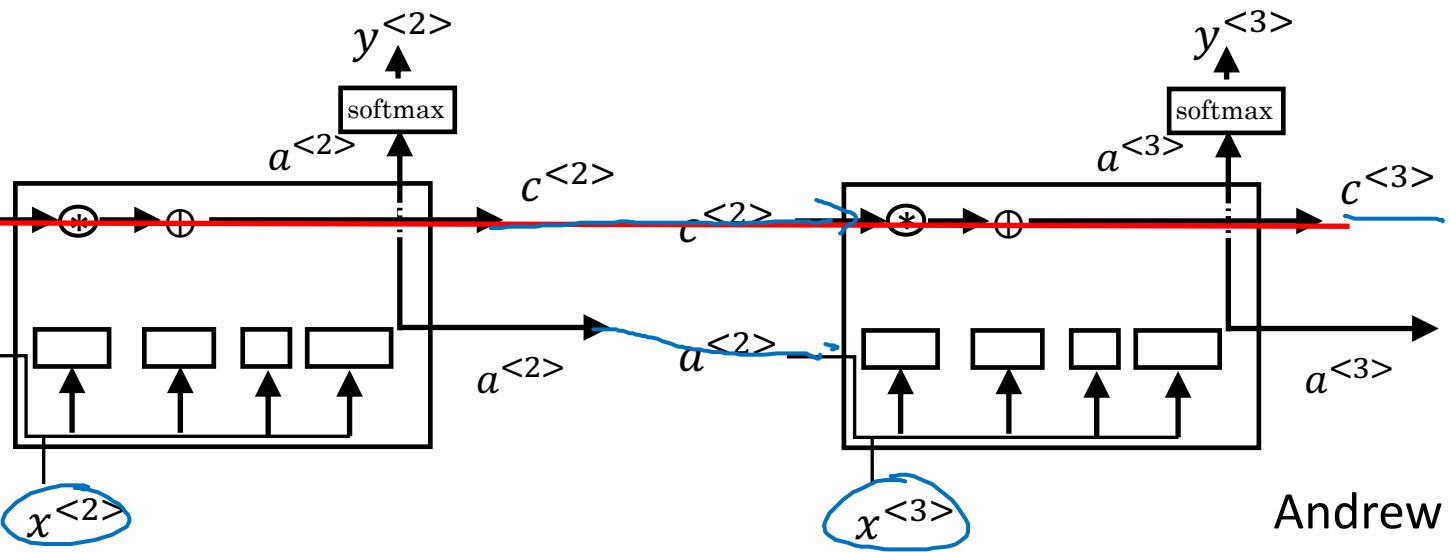
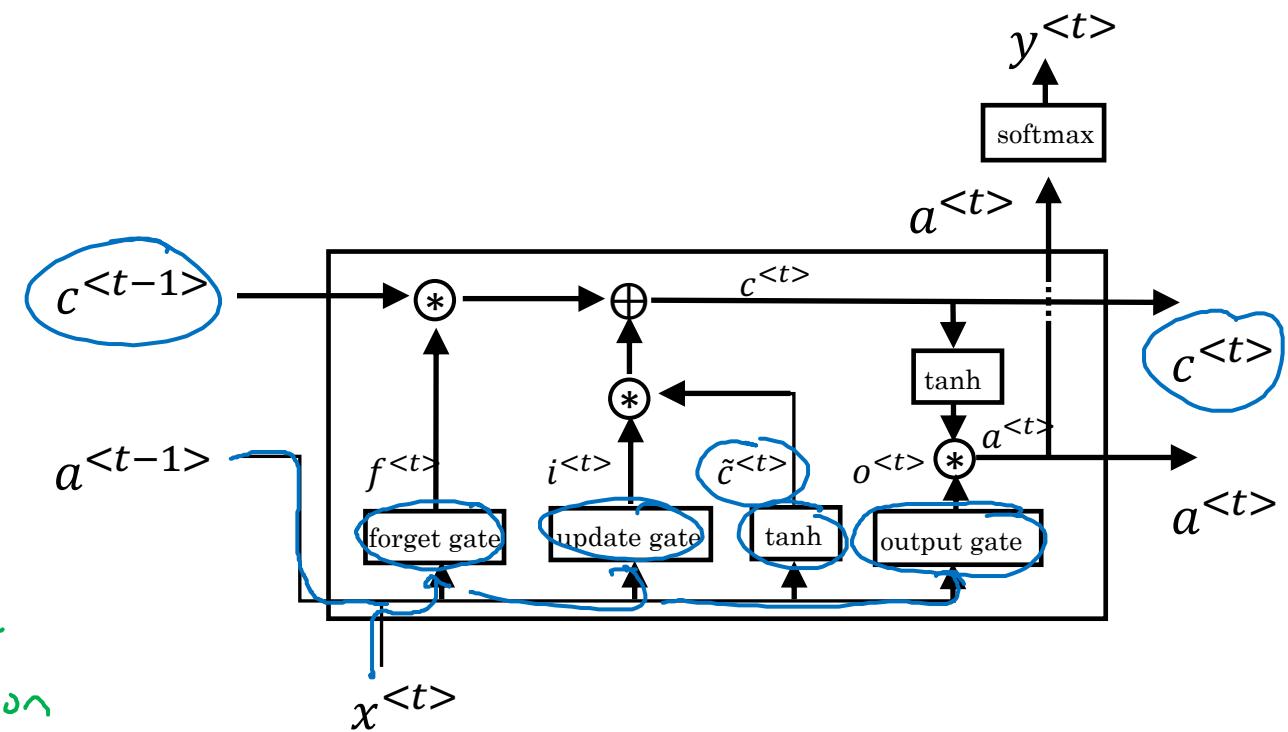
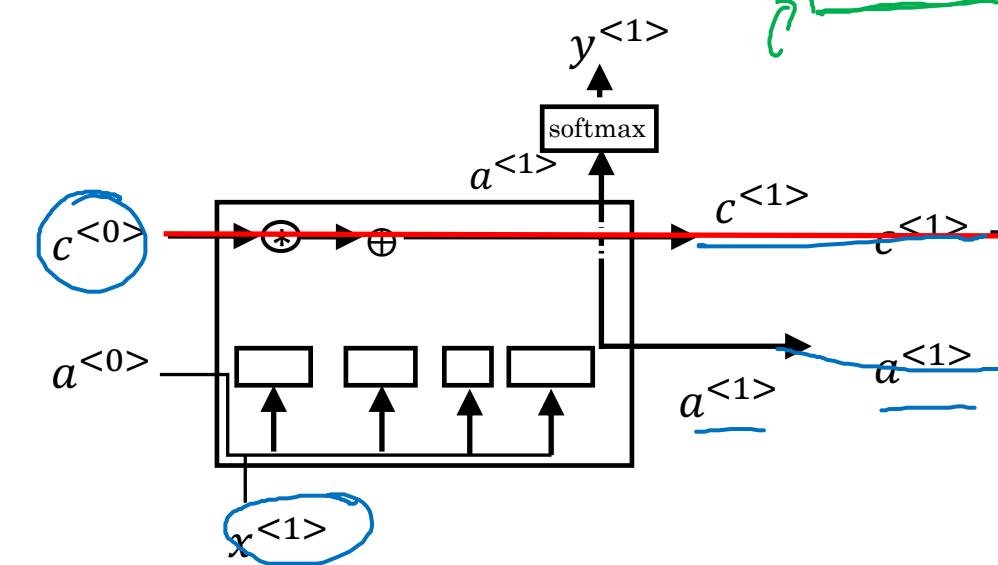
$$\Gamma_f = \sigma(W_f[a^{<t-1>}, x^{<t>}] + b_f)$$

$$\Gamma_o = \sigma(W_o[a^{<t-1>}, x^{<t>}] + b_o)$$

$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + \Gamma_f * c^{<t-1>}$$

$$a^{<t>} = \Gamma_o * c^{<t>}$$

peephole connection



Andrew Ng



deeplearning.ai

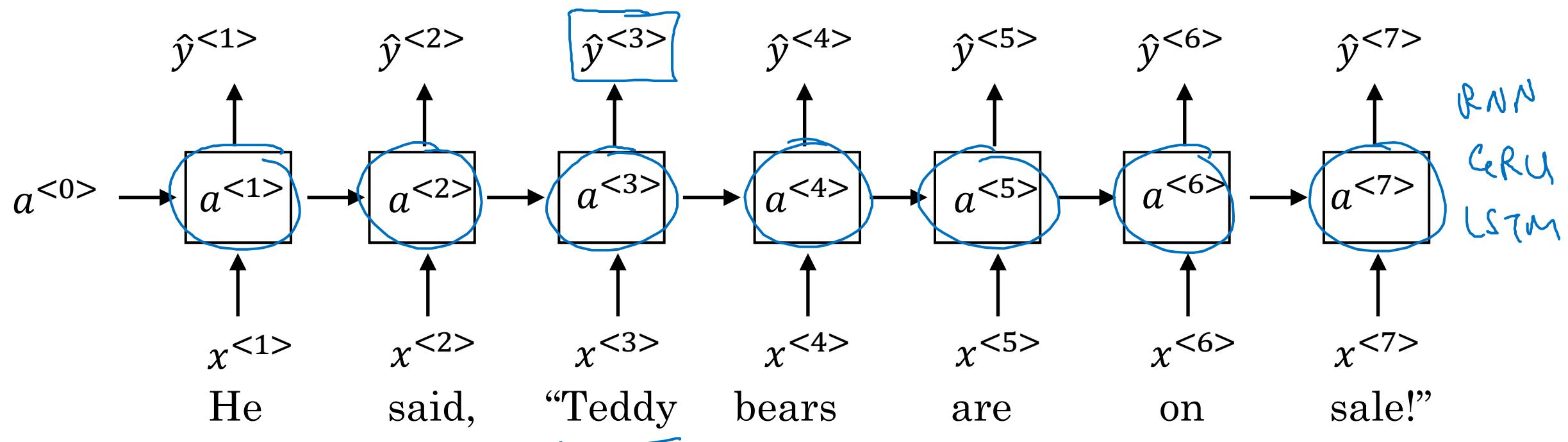
Recurrent Neural Networks

Bidirectional RNN

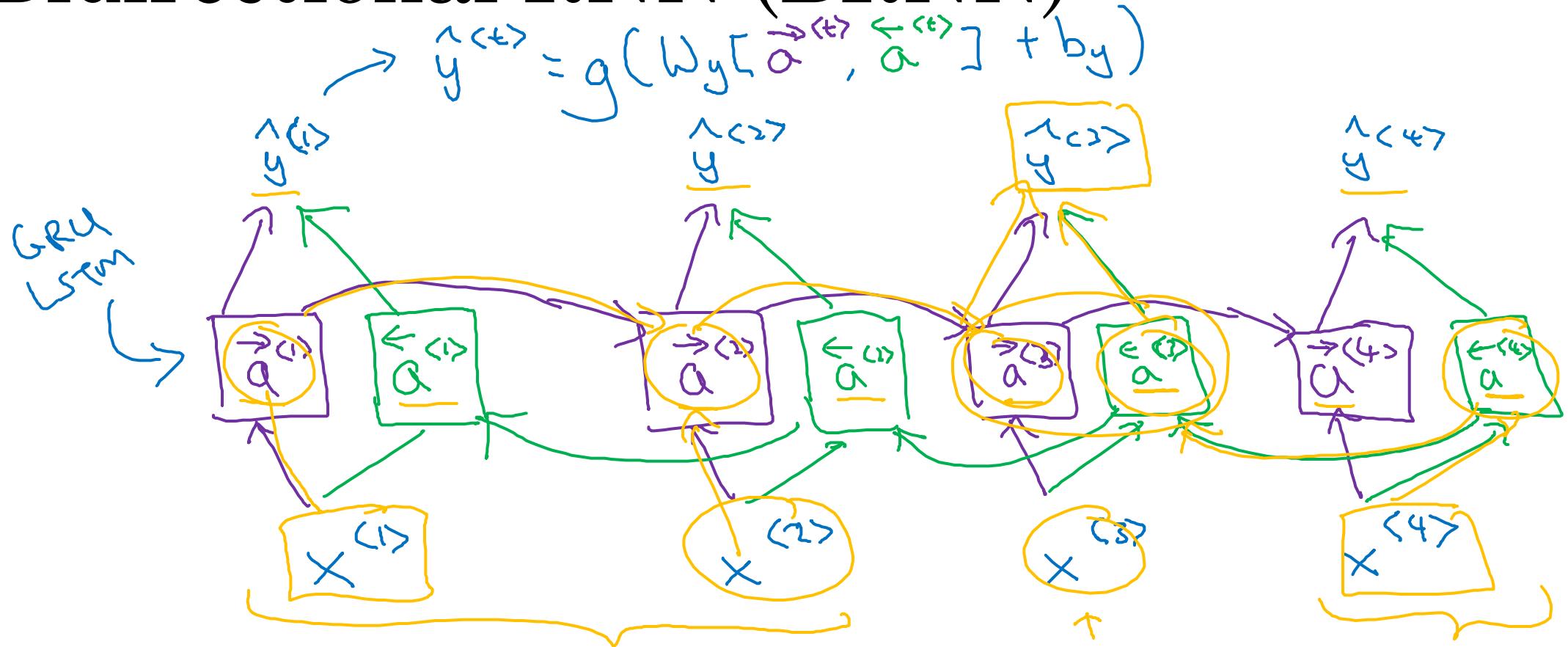
Getting information from the future

He said, “Teddy bears are on sale!”

He said, “Teddy Roosevelt was a great President!”



Bidirectional RNN (BRNN)



Acyclic graph

BRNN w/ LSTM

He said,

"Teddy Roosevelt ..."



deeplearning.ai

Recurrent Neural Networks

Deep RNNs

Deep RNN example

