deeplearning.ai

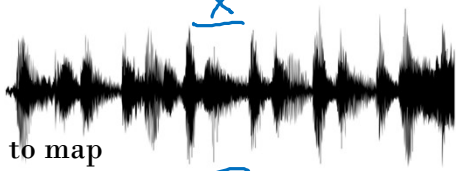
Recurrent Neural Networks

rrn have revulutinized speech recognition, natural language processing and other areas, here we learn how to build these models

Why sequence models?

Examples of sequence data

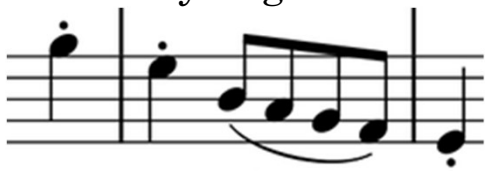
Speech recognition



"The quick brown fox jumped over the lazy dog."

you are given an audio clip imput x and asked to map it to a transcript y, both input and output are sequence data

Music generation



this is another example with sequence data, only output is a sequence, input can be empty set or single integer

Sentiment classification

"There is nothing to like in this movie."

DNA sequence analysis

AGCCCCTGTGAGGAACTAG

AGCCCCTGTGAGGAACTAG

this part of the dna sequence correspond to a proteine

Machine translation

Voulez-vous chanter avec moi?

Do you want to sing with me?

Plasmid sequence AP4 Oligo

Video activity recognition

Running

you are given a sequence of frames and you are asked to recognize the activity

Name entity recognition

Yesterday, Harry Potter met Hermione Granger.

Yesterday, Harry Potter met Hermione Granger.

you are given a sentence and you are asked to identify the people

Andrew Ng

so its supervised learning and there are different type of sequences
in some examples x and y have same lenght, sometime only one is a sequence.

deeplearning.ai

Recurrent Neural Networks

thats start by defining a notation to define these sequence models

Notation

Motivating example

Now we starting to work in NLP. How to reppresent words, how should Harry really be

NLP

say you want to input a sequence like x and we want y to tell where are peolples names. this problem is called Named Entety Recognition, and its used by search engines. Some also tell where name starts and where it ends.

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

$x^{<1>}$ $x^{<2>}$ $x^{<3>}$ ... $x^{<t>}$ ... $x^{<9>}$

these index the words

$T_x = 9$

y: 1 1 0 1 1 0 0 0 0

$y^{<1>}$ $y^{<2>}$ $y^{<3>}$ ... $y^{<9>}$

same in indexing output

$T_y = 9$

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

that i there indicates the training example

$T_x^{(i)} = 9$ 15

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

$T_y^{(i)}$

these indicate the lenght of the input and output of the x and y of the example i

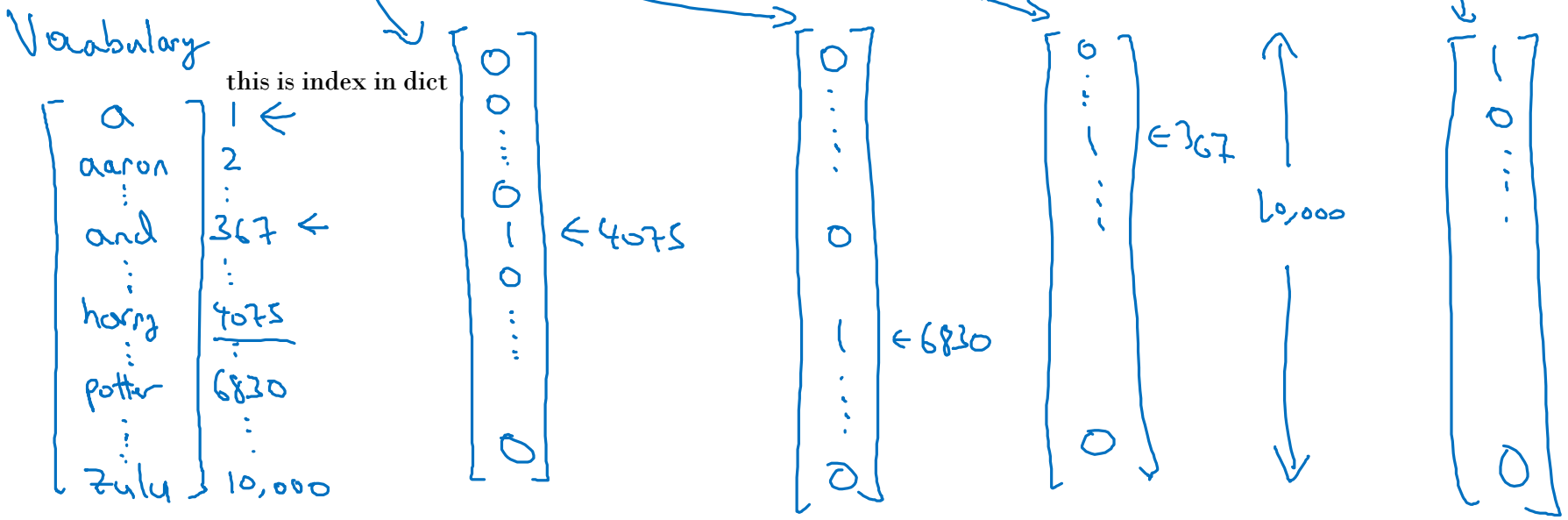
Andrew Ng

how to reppresent the words in a sentence, you take a vocabulary

Representing words

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

$x^{<1>}$ $x^{<2>}$ $x^{<3>}$ ... $x^{<7>}$ $x^{<9>}$



we using 10000 words dictionary

this for words not in dictionary

One-hot

then we use one hot encoding to reppresent Harry which will be all o vector exept a 1 in index where we find harry in dict eache of this vectors will be of size 10000 just like dict

Andrew Ng

Representing words

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

$x^{<1>}$ $x^{<2>}$ $x^{<3>}$ … $x^{<9>}$

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

Andrew Ng

Now we see how to learn the mapping from x to y with the neural network

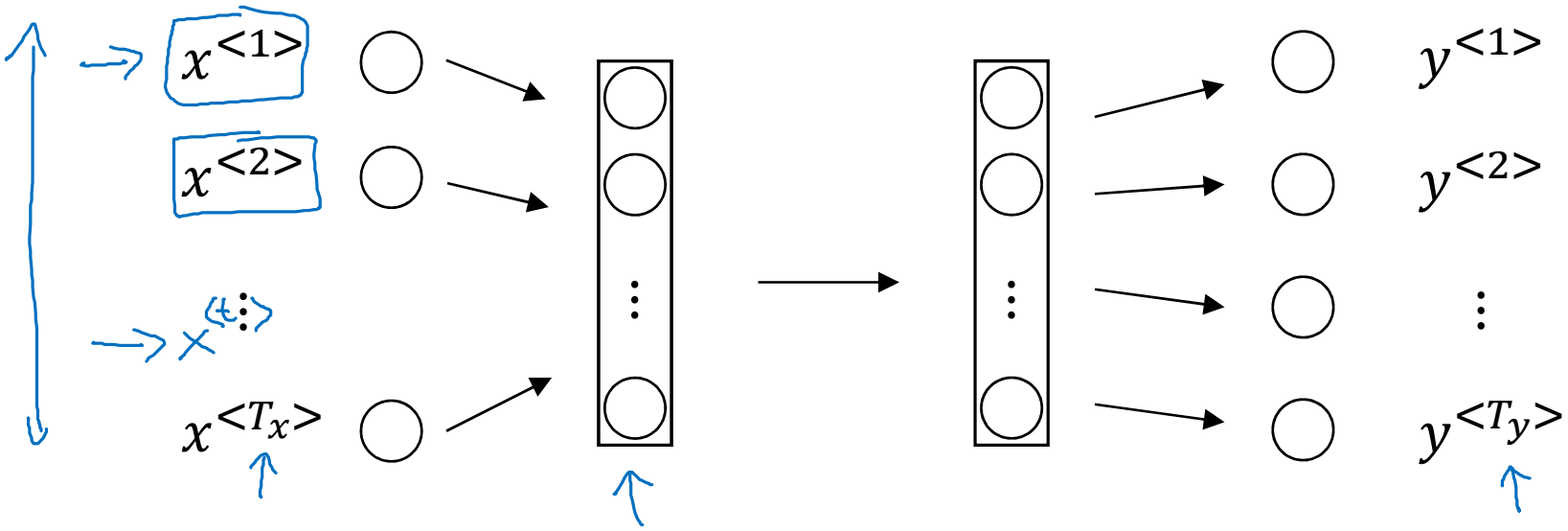
deeplearning.ai

Recurrent Neural Networks

Recurrent Neural Network Model

feeding the 9 inputs to a NN and then get those 9 y its not a good idea for the problems below

Why not a standard network?



Problems:

becouse u have examples with different sentences that contain different amount of words

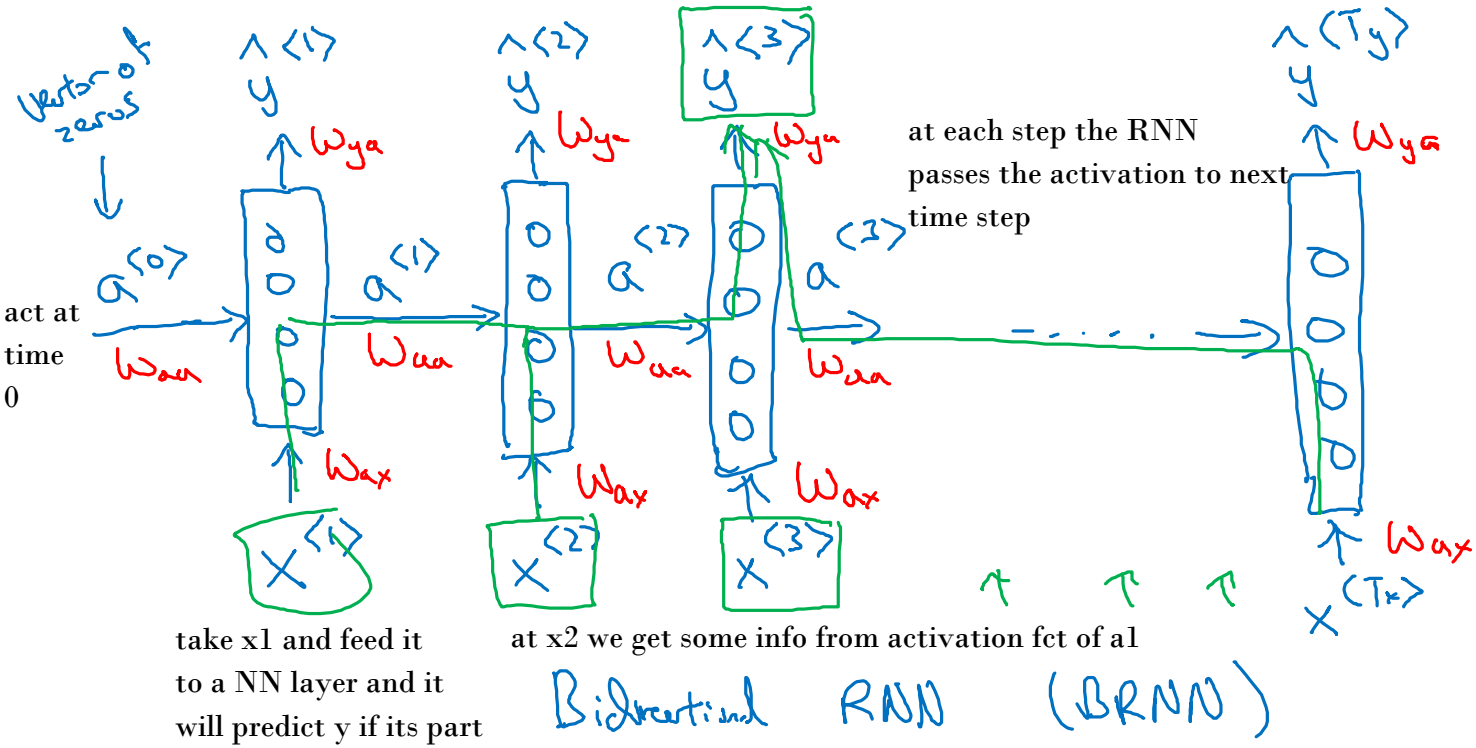
- Inputs, outputs can be different lengths in different examples.
- Doesn't share features learned across different positions of text.

Andrew Ng

A RNN does not have these two dissatvantages

What is a recurrent NN? thats build one up

Recurrent Neural Networks



act at time 0

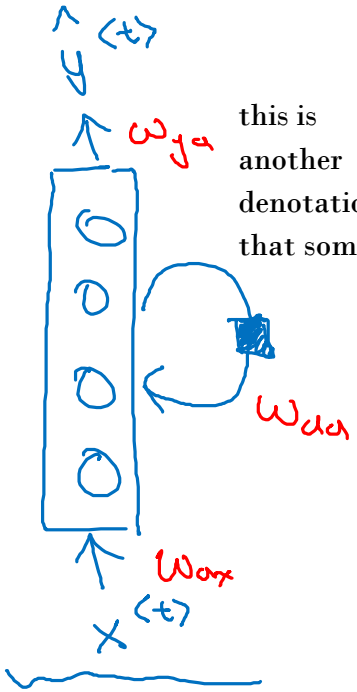
take x1 and feed it to a NN layer and it will predict y if its part of person name or not

at x2 we get some info from activation fct of a1

at each step the RNN passes the activation to next time step

Bidirectional RNN (BRNN)

$T_x = T_y$ in this case they are equal, architecture change a bit if not equal



this is another denotation that some use

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

He said, “Teddy bears are on sale!”

one weakness is that it uses info only about previews words not next ones

you cant tell whether Teddy is a name by just looking at first 3 words. We will address this when we talk about bidirectional recurrent NN

the parameters Wax are shared. we describe in dettail next slide

Andrew Ng

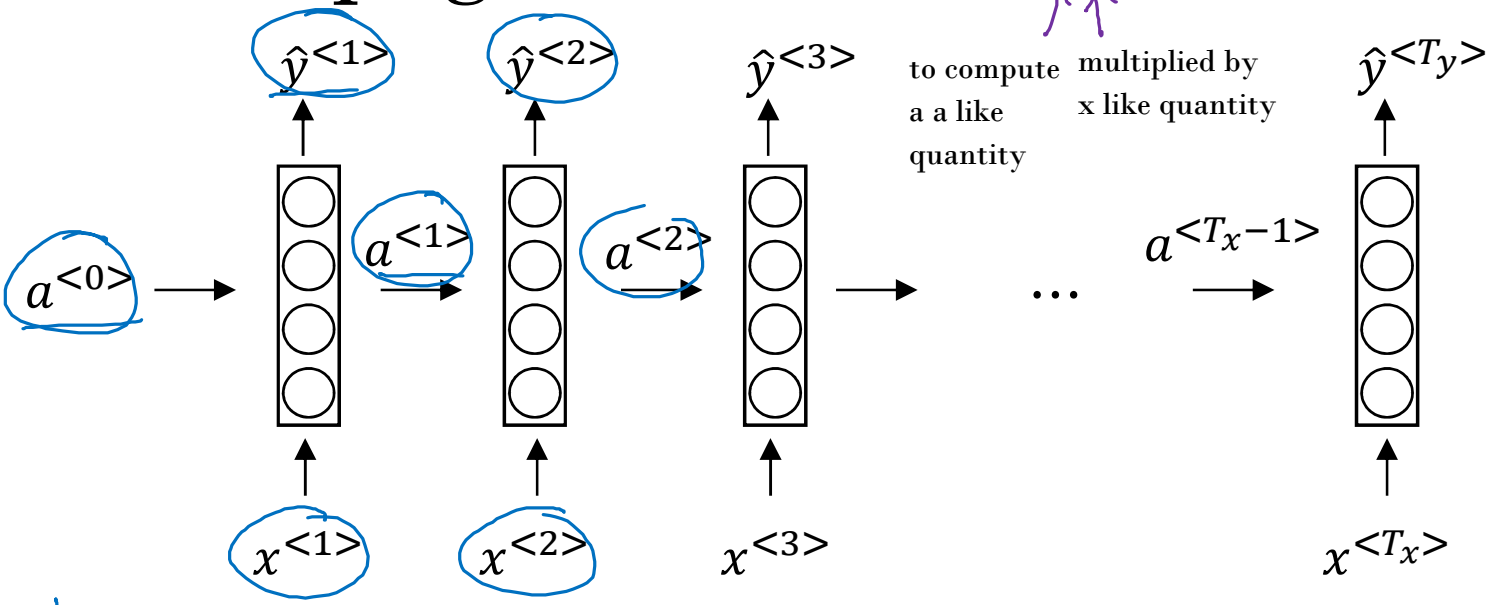
thats see what are the calculations that the RNN does

Forward Propagation

$a \leftarrow W_{ax} x^{\langle 1 \rangle}$

to compute a a like quantity

multiplied by x like quantity



$a^{\langle 0 \rangle} = \vec{0}$

start with that vector of all 0

$a^{\langle 1 \rangle} = g_1(W_{aa} a^{\langle 0 \rangle} + W_{ax} x^{\langle 1 \rangle} + b_a)$ ← tanh / ReLU

$\hat{y}^{\langle 1 \rangle} = g_2(W_{ya} a^{\langle 1 \rangle} + b_y)$ ← sigmoid

depends on type of output u have

$$a^{\langle t \rangle} = g(W_{aa} a^{\langle t-1 \rangle} + W_{ax} x^{\langle t \rangle} + b_a)$$
$$\hat{y}^{\langle t \rangle} = g(W_{ya} a^{\langle t \rangle} + b_y)$$

this is more generally

Andrew Ng

Simplified RNN notation

Here you have the semplified notation

we writ this simpler way

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

(100,100) 100 (100,10,000) 10,000

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

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

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

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

100 × (100, 10100)

100, 10000

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

100, 10000, 10100

$$\begin{bmatrix} W_{aa} ; W_{ax} \end{bmatrix} \begin{bmatrix} a^{<t-1>} \\ x^{<t>} \end{bmatrix} = W_{aa}a^{<t-1>} + W_{ax}x^{<t>}$$

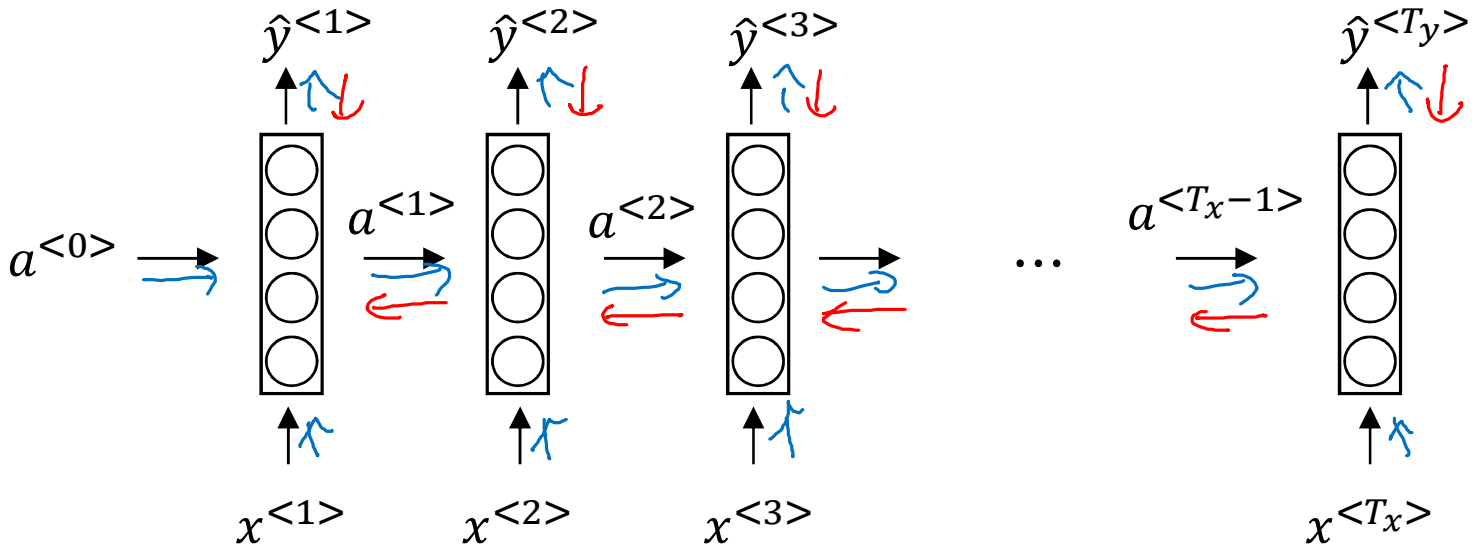
Andrew Ng

deeplearning.ai

Recurrent Neural Networks

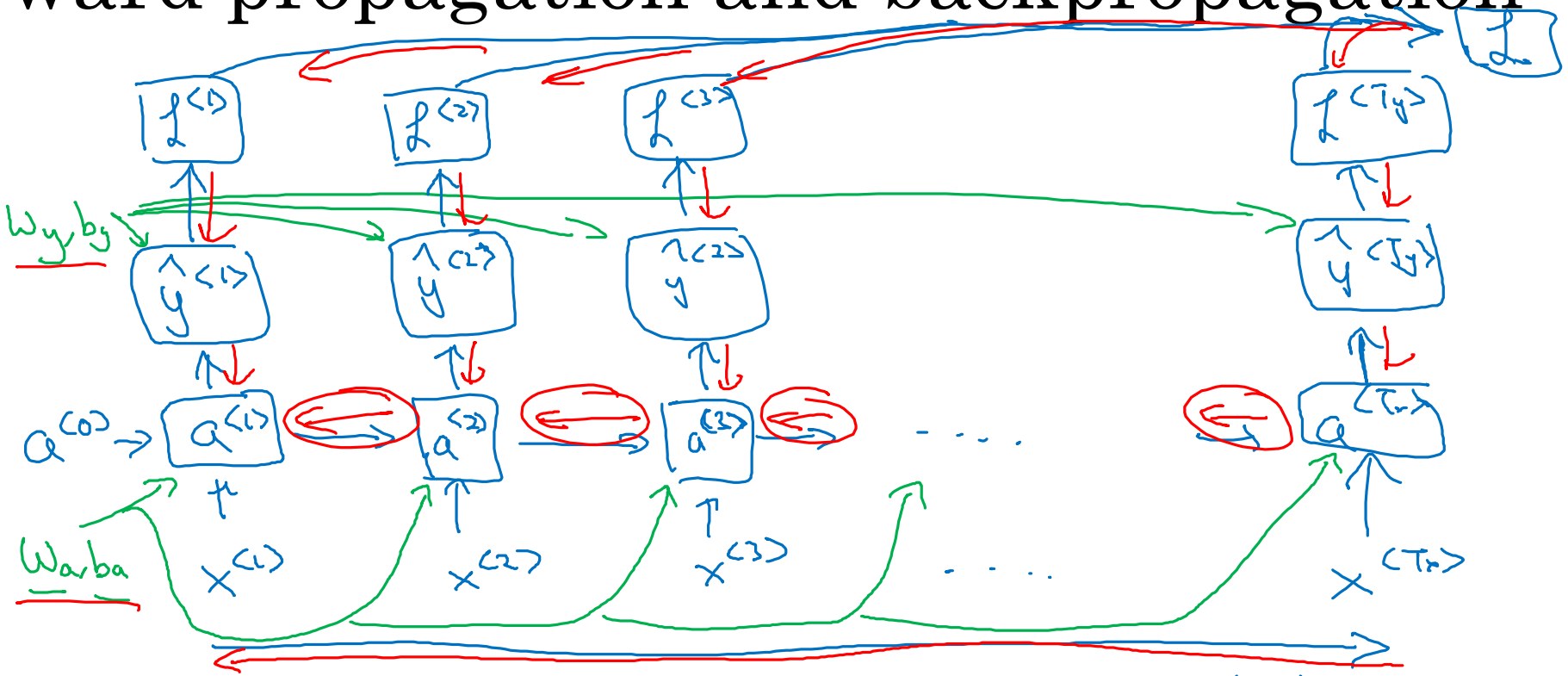
Backpropagation through time

Forward propagation and backpropagation



Andrew Ng

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

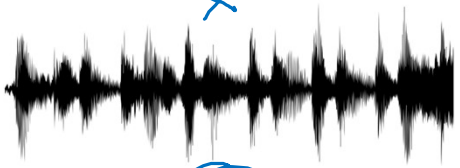
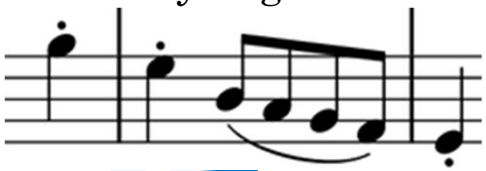
not allways these two are equal

Tx Ty

Examples of sequence data

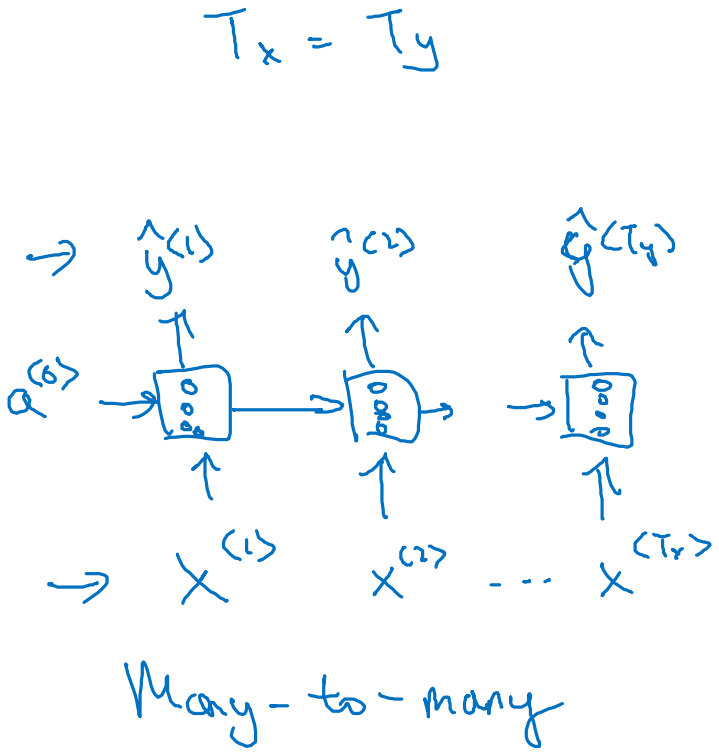
x

y

Speech recognition		"The quick brown fox jumped over the lazy dog."
Music generation	∅	
Sentiment classification	"There is nothing to like in this movie."	
DNA sequence analysis	AGCCCCTGTGAGGAACTAG	AGCCCCTGTGAGGAACTAG
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

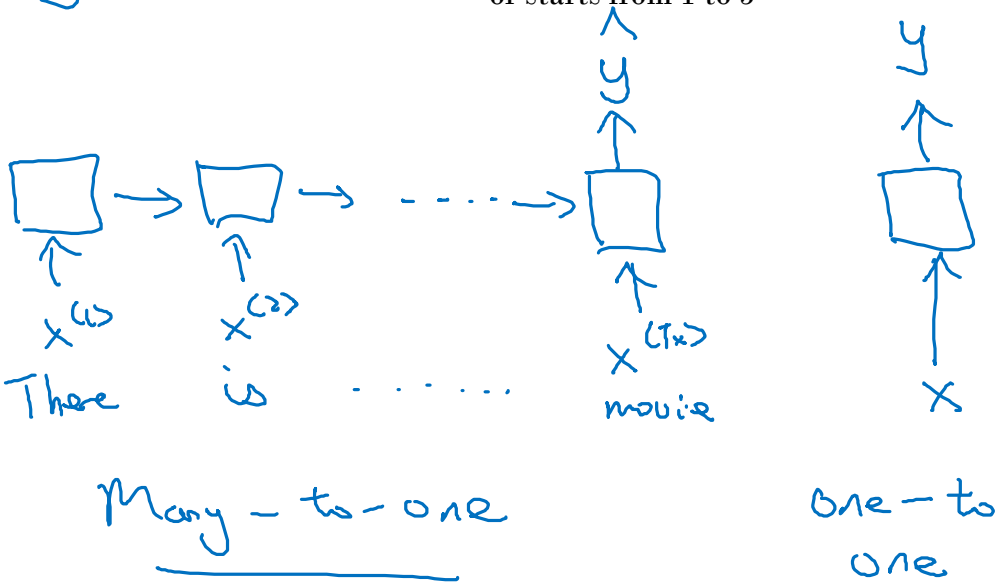


many inputs many outputs

Sentiment classification

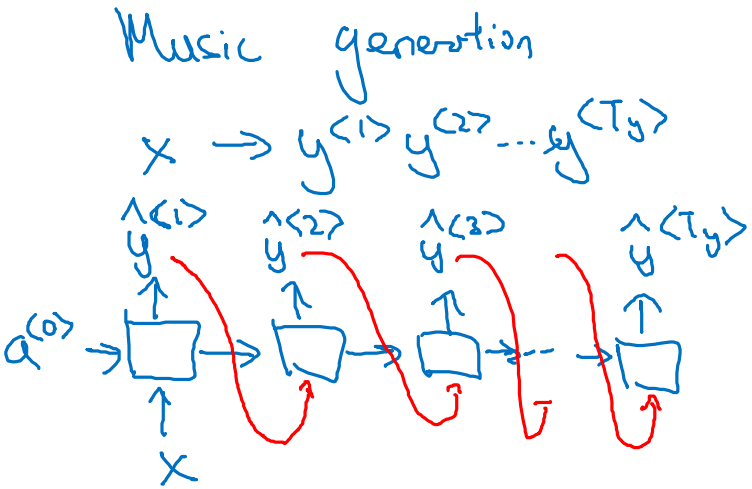
x = text — this it the text input, the review

y = 0/1 ... 1-5 — review of a movie positive/negative or starts from 1 to 5



thi is NN that Andrew Ng we did in first two courses

Examples of RNN architectures

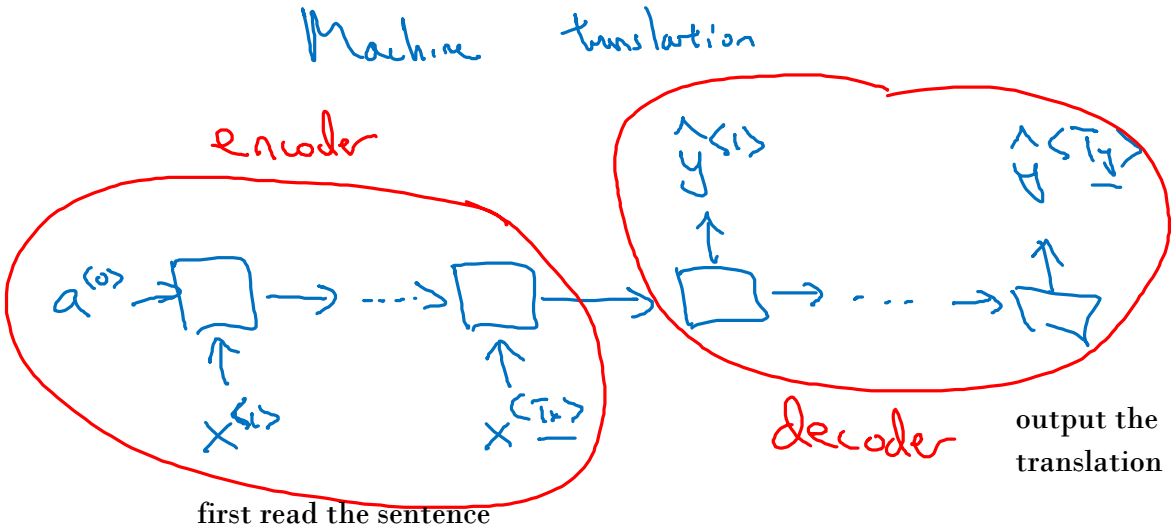


One-to-many

$x = \phi$

an example of one to many is music generation,
and we can implement thi in the exercises. Input can
be an integer identifying the gener of music u want, or u can have
no input just input of 0

Machine translation



first read the sentence

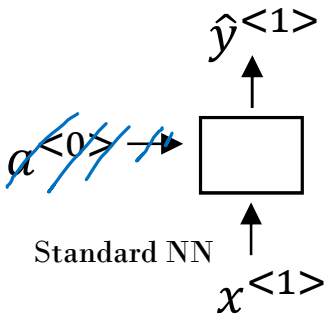
output the translation

Many-to-many

when input length and output length are different.

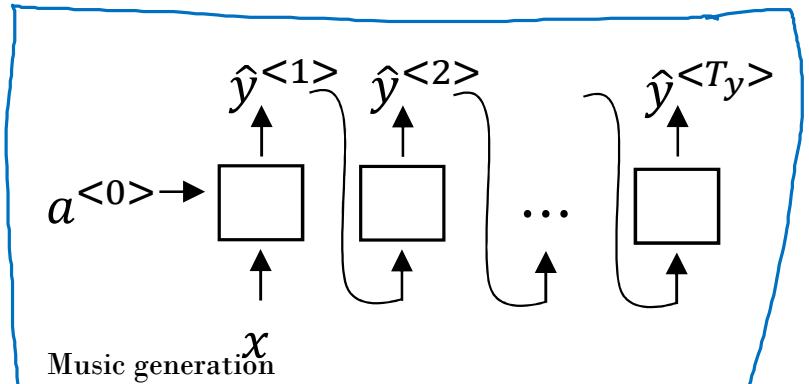
Andrew Ng

Summary of RNN types



Standard NN

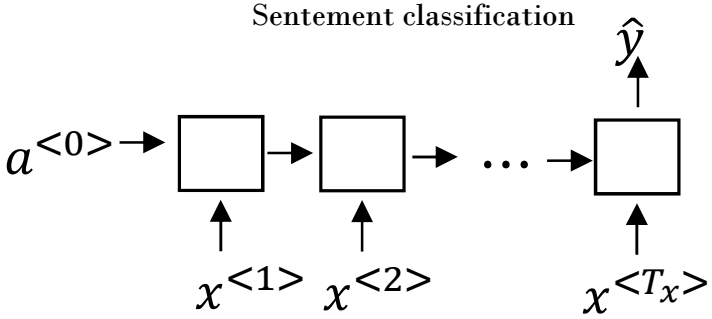
One to one



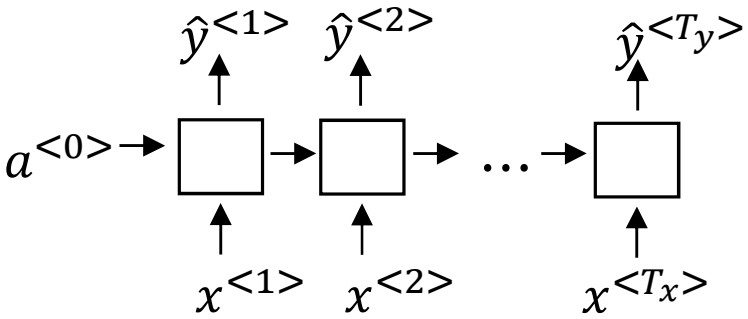
Music generation

One to many

Sentement classification

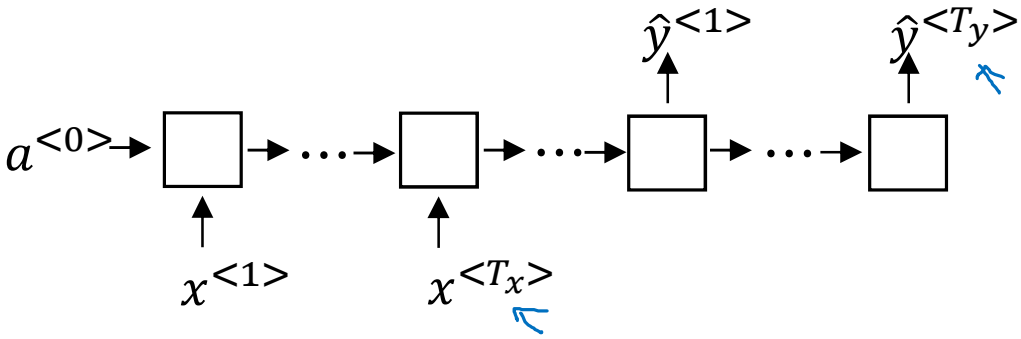


Many to one



Many to many

$T_x = T_y$



Many to many

Machine translation

Andrew Ng

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.

These two sound very similar but a speech recognit would hear the second one.

$P(\text{The apple and pair salad}) = 3.2 \times 10^{-13}$

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

A probability model tells the chances of these two models.

this is much more likely

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

$P(y^{\langle 1 \rangle}, y^{\langle 2 \rangle}, \ldots, y^{\langle T_y \rangle})$

What a language model does is tells the probability of that sentence.

Andrew Ng

Language modelling with an RNN

Training set: large corpus of english text.

Tokenize

Cats average 15 hours of sleep a day. <EOS>

$y^{<1>}$ $y^{<2>}$ $y^{<3>}$ ... $y^{<8>}$ $y^{<9>}$

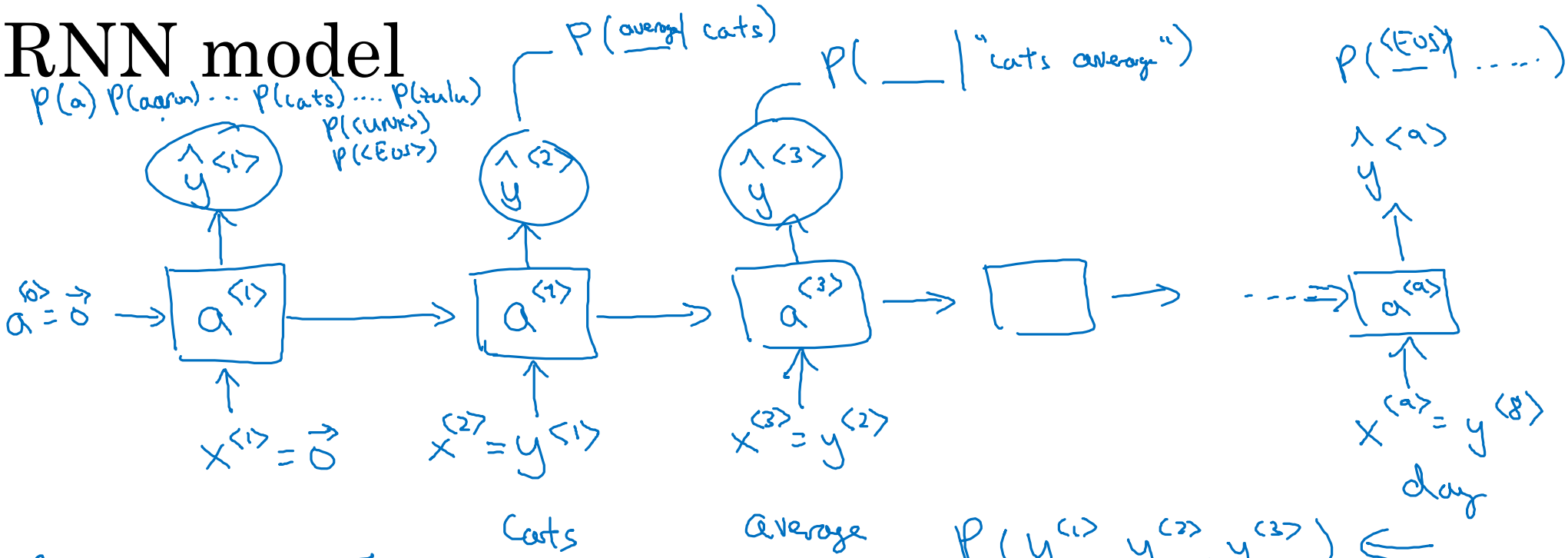
$x^{<t>} = y^{<t-1>}$

The Egyptian ~~Mau~~ is a bread of cat. <EOS>

<UNK>

10,000

RNN model



Cats average 15 hours of sleep a day. <EOS>

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

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

$$P(y^{<1>}, y^{<2>}, y^{<3>}) = P(y^{<1>}) \, P(y^{<2>} \mid y^{<1>}) \, P(y^{<3>} \mid y^{<1>}, y^{<2>})$$

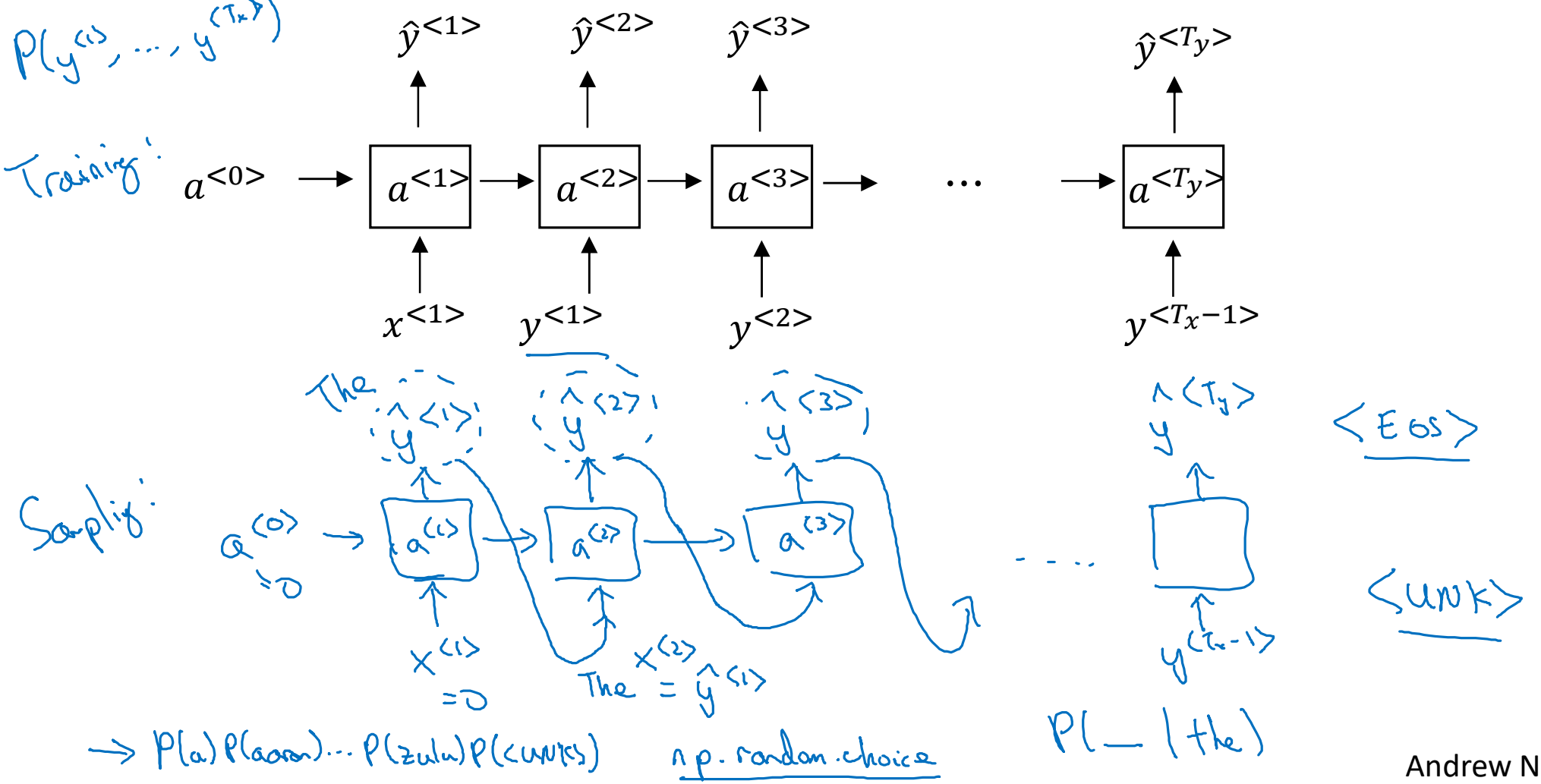
Andrew Ng

deeplearning.ai

Recurrent Neural Networks

Sampling novel sequences

Sampling a sequence from a trained RNN



Andrew Ng

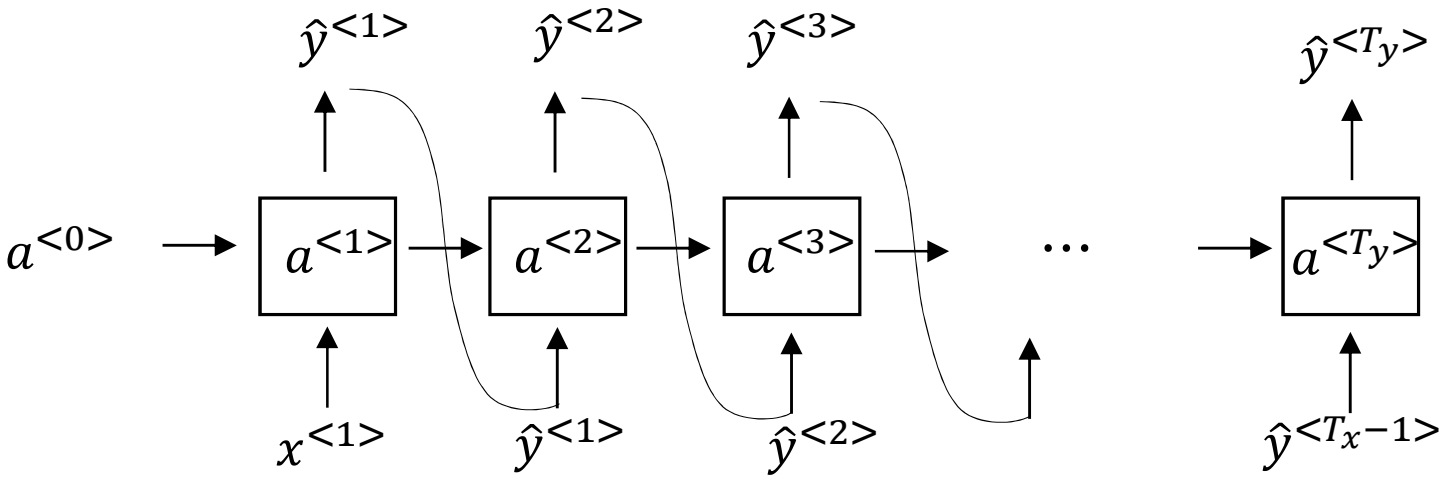
Character-level language model

→ Vocabulary = [a, aaron, …, zulu, <UNK>] ←

→ Vocabulary = [a, b, c, … , z, ␣, ., ,, ;, 0, …, 9, A, …, Z]

$y^{<1>}$ $y^{<2>}$ $y^{<3>}$ $y^{<4>}$

Cat average … Mau



Andrew Ng

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 besser be my love to me see sabl's.

For whose are ruse of mine eyes heaves.

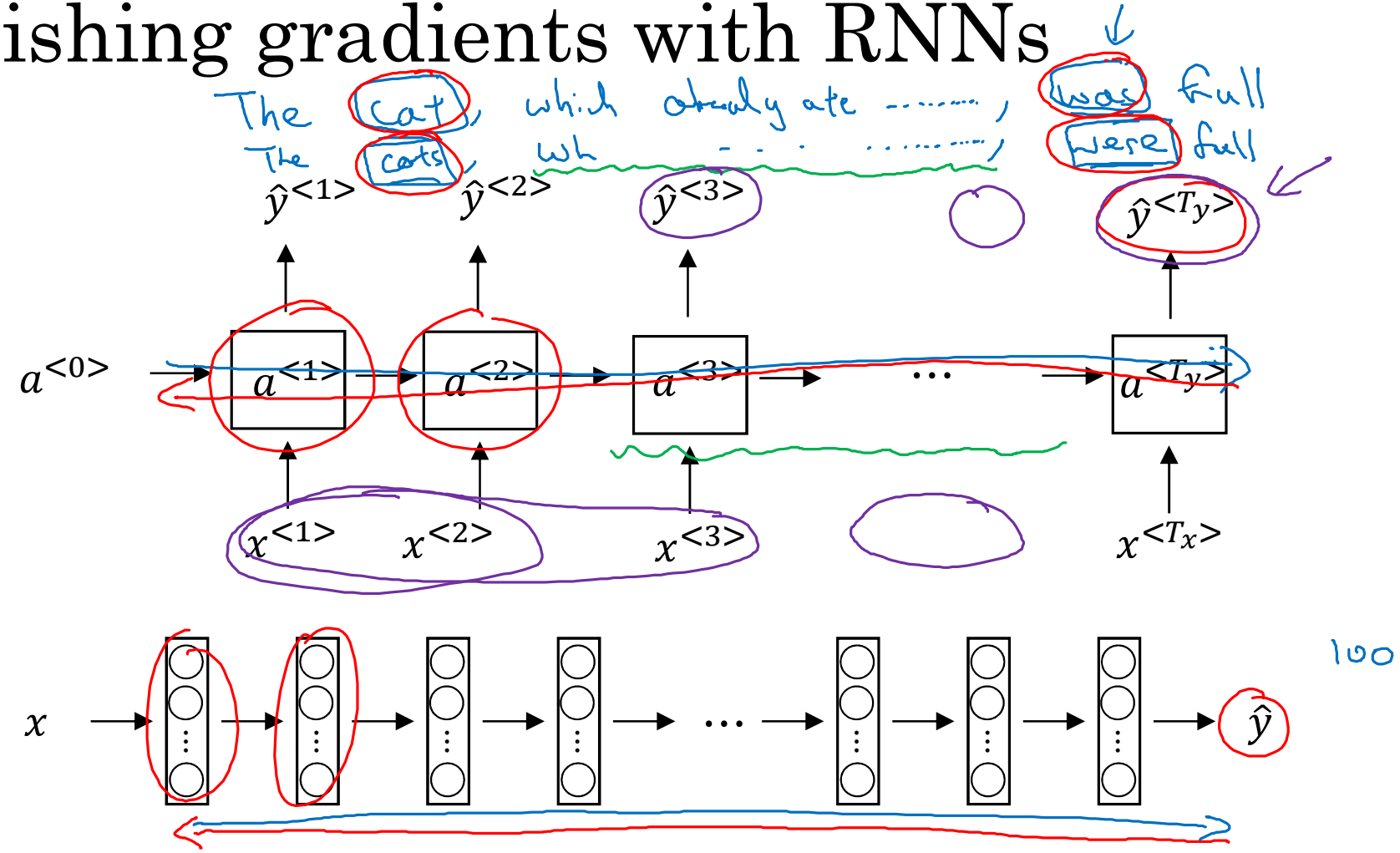
Andrew Ng

deeplearning.ai

Recurrent Neural Networks

Vanishing gradients with RNNs

Vanishing gradients with RNNs



Exploding gradients.

NaN

Gradient clipping

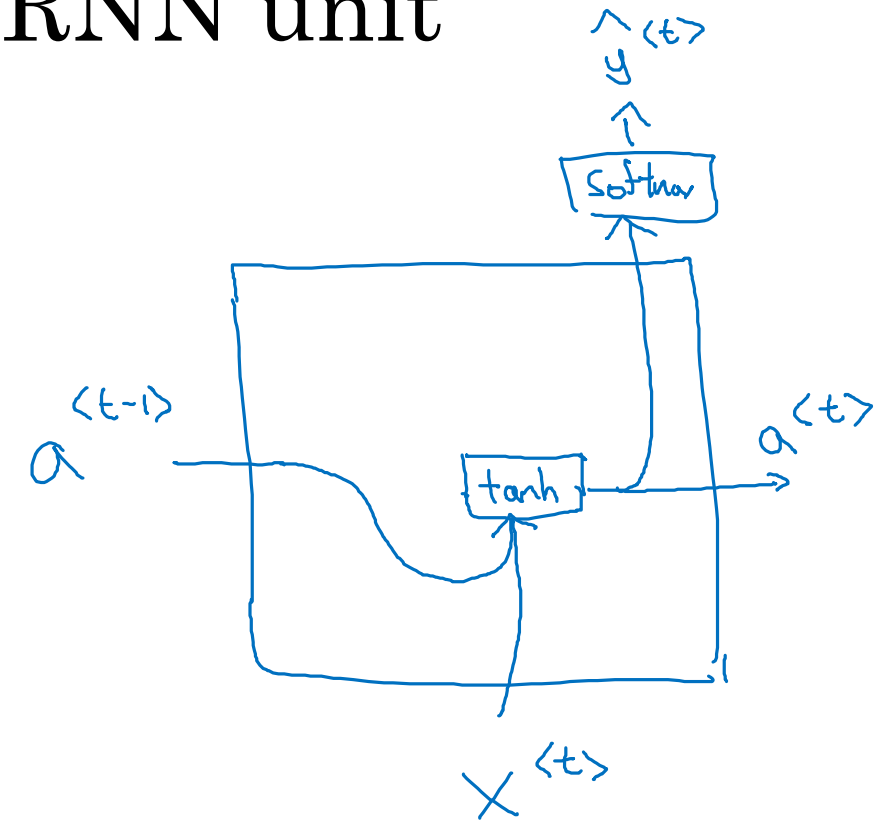
Andrew Ng

deeplearning.ai

Recurrent Neural Networks

Gated Recurrent Unit (GRU)

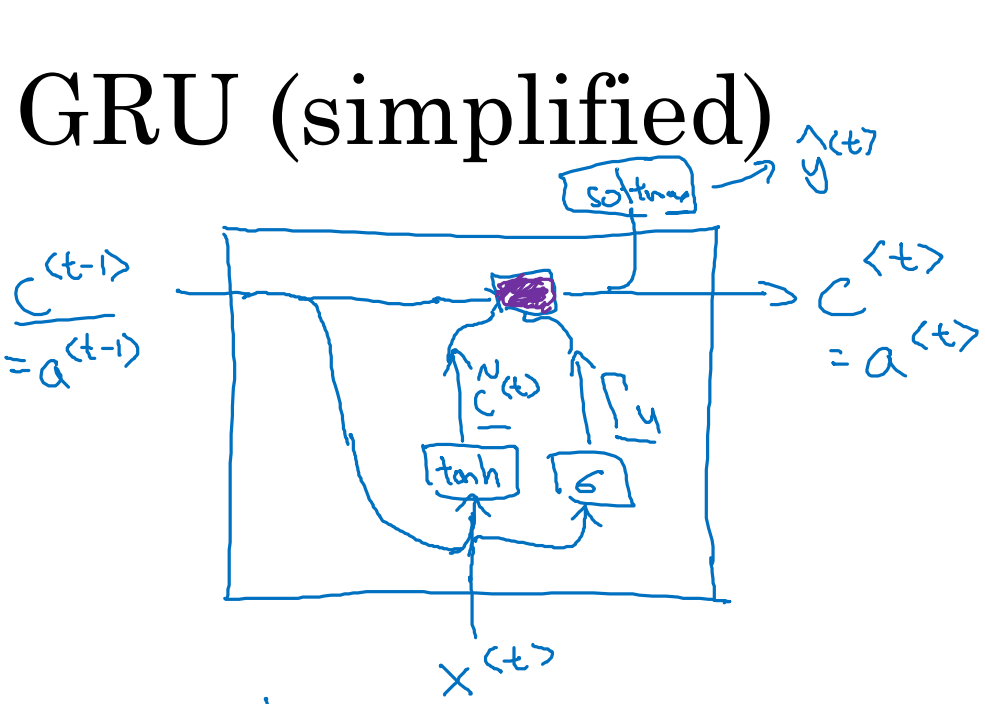
RNN unit

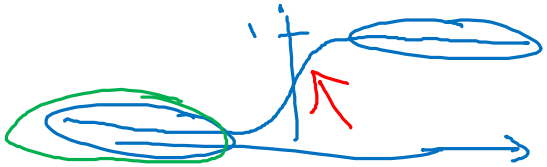


$$a^{<t>} = g(W_a[a^{<t-1>}, x^{<t>}] + b_a)$$

tanh

GRU (simplified)





c = memory cell

$c^{\langle t \rangle} = a^{\langle t \rangle}$

$\tilde{c}^{\langle t \rangle} = \tanh\left(W_c \left[c^{\langle t-1 \rangle}, x^{\langle t \rangle}\right] + b_c\right)$

$\Gamma_u = \sigma\left(W_u \left[c^{\langle t-1 \rangle}, x^{\langle t \rangle}\right] + b_u\right)$

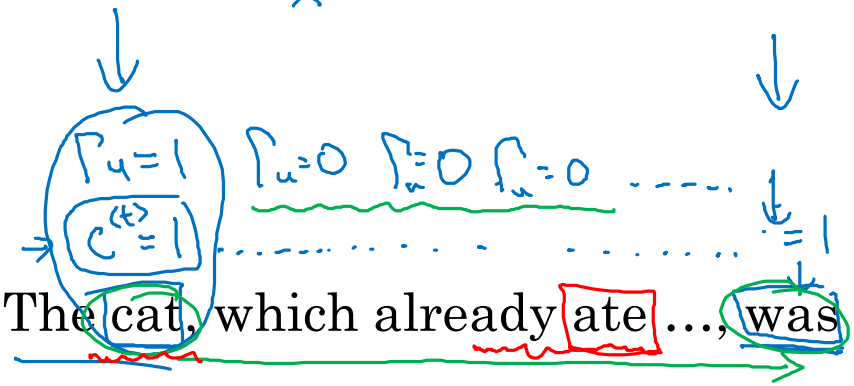
"update"

$c^{\langle t \rangle} = \Gamma_u * \tilde{c}^{\langle t \rangle} + (1 - \Gamma_u) * c^{\langle t-1 \rangle}$

element-wise

Gate

$\Gamma_u = 0.0 \ldots 001$



The cat, which already ate …, was full.

[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

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

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

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

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

LSTM

The cat, which ate already, was full.

deeplearning.ai

Recurrent Neural Networks

LSTM (long short term memory) unit

GRU and LSTM

GRU

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

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

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

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

$$a^{<t>} = c^{<t>}$$

Γ_f

LSTM

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

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

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

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

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

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

[Hochreiter & Schmidhuber 1997. Long short-term memory]

LSTM units

GRU

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

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

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

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

$$a^{<t>} = c^{<t>}$$

LSTM

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

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

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

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

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

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

[Hochreiter & Schmidhuber 1997. Long short-term memory]

LSTM in pictures

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

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

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

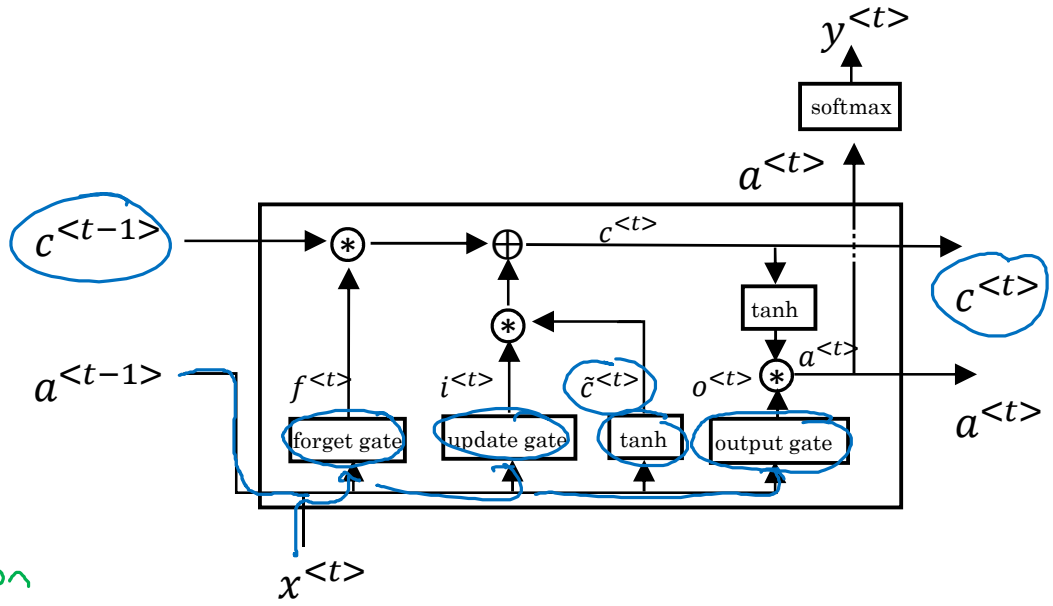
$$\Gamma_o = \sigma(W_o[a^{<t-1>}, x^{<t>}] + b_o)$$

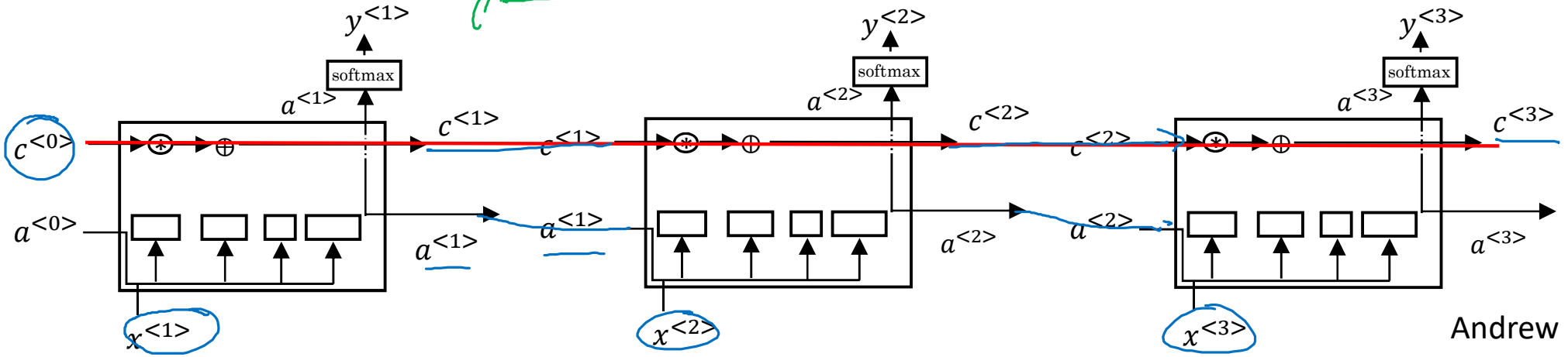
$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + \Gamma_f * c^{<t-1>}$$

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

$c^{<t-1>}$

peephole connection





Andrew Ng

deeplearning.ai

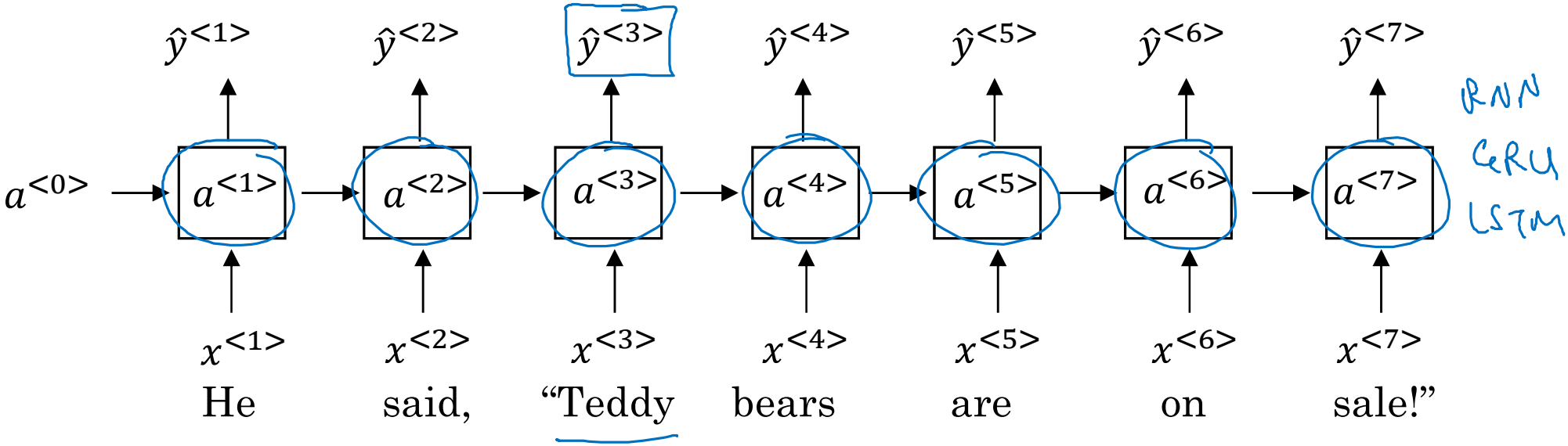
Recurrent Neural Networks

Bidirectional RNN

Getting information from the future

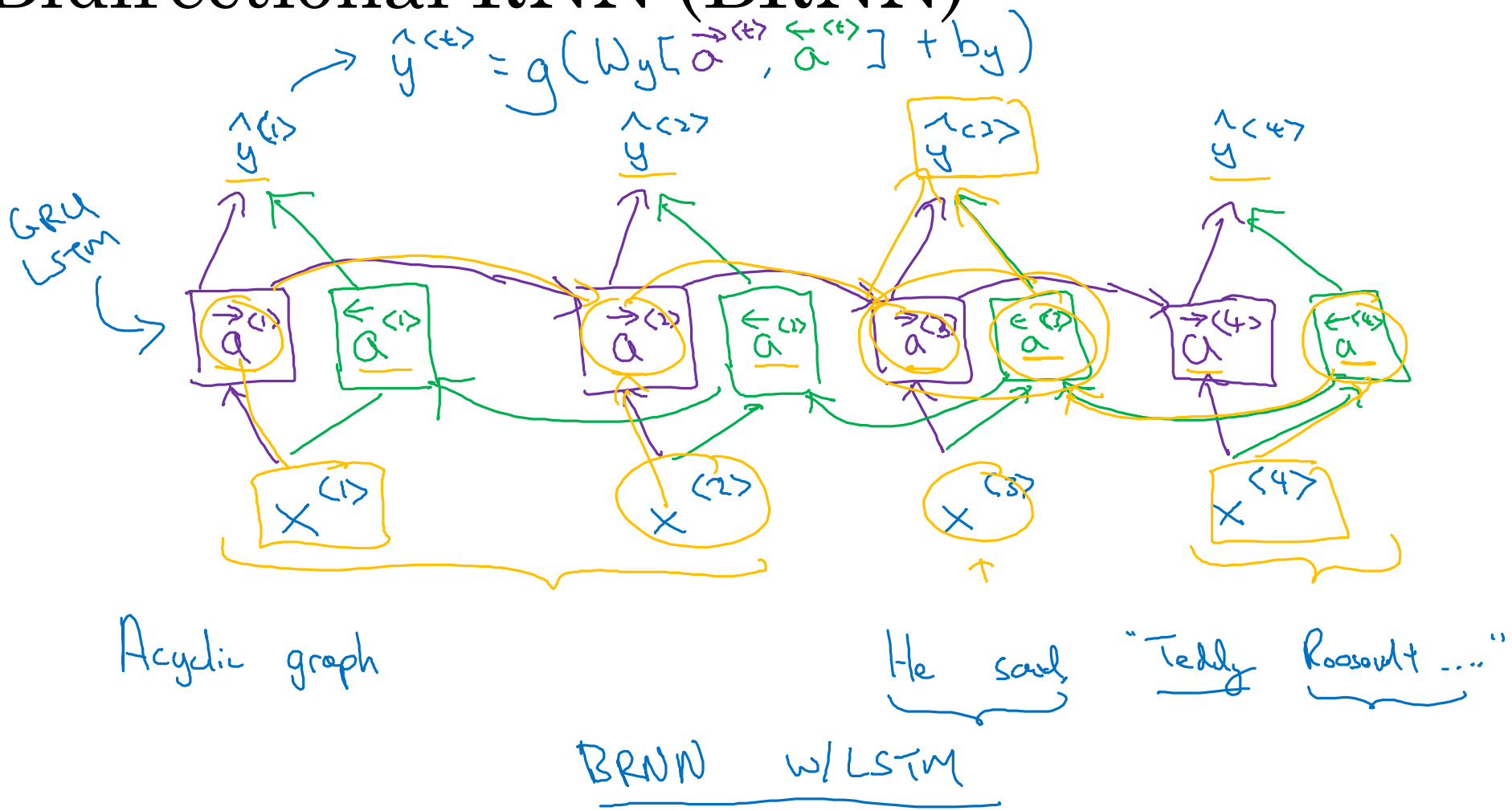
He said, “Teddy bears are on sale!”

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



Andrew Ng

Bidirectional RNN (BRNN)



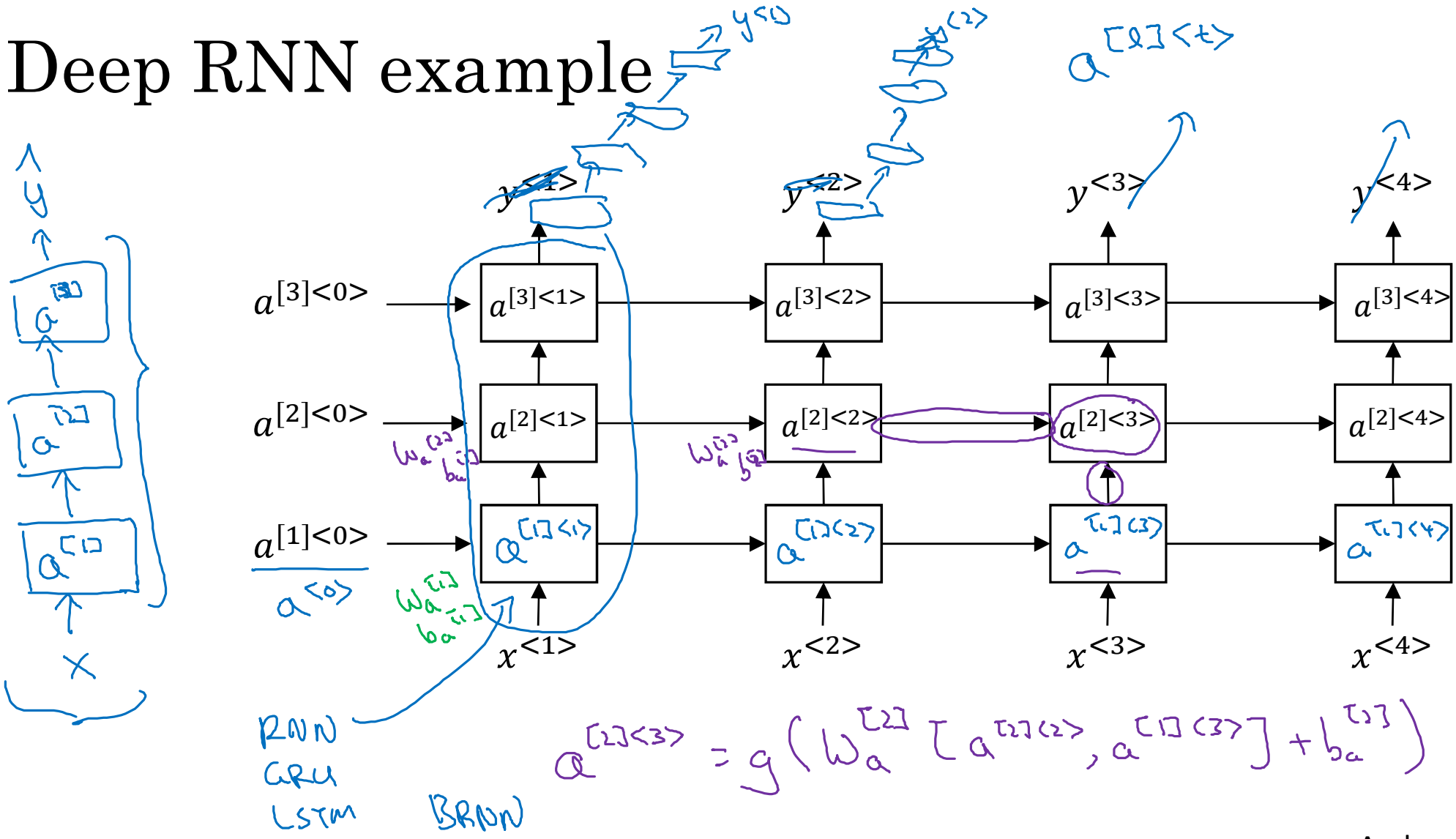
Andrew Ng

deeplearning.ai

Recurrent Neural Networks

Deep RNNs

Deep RNN example



Andrew Ng