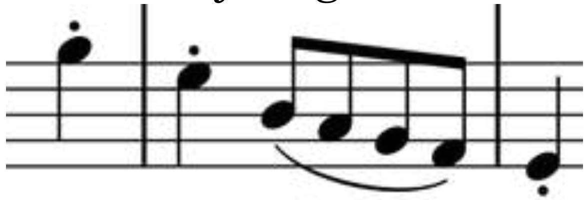
deeplearning.ai

Recurrent Neural Networks

Why sequence models?

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.

deeplearning.ai

Recurrent Neural Networks

Notation

Motivating example

NLP

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

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

$T_x = 9$

y: 1 1 0 1 1 0 0 0 0

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

$T_y = 9$

$X^{(i)<t>}$ $T_x^{(i)} = 9$ 15

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

Andrew Ng

Representing words

$x^{<t>}$ (x, y)

$x \rightarrow y$

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

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

Vocabulary

$$\begin{bmatrix} \text{a} \\ \text{aaron} \\ \vdots \\ \text{and} \\ \vdots \\ \text{harry} \\ \vdots \\ \text{potter} \\ \vdots \\ \text{zulu} \end{bmatrix} \begin{matrix} 1 \leftarrow \\ 2 \\ \vdots \\ 367 \leftarrow \\ \vdots \\ \underline{4075} \\ \vdots \\ 6830 \\ \vdots \\ 10{,}000 \end{matrix}$$

<UNK> $\underline{10{,}000}$

$$\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \leftarrow 4075 \\ \\ \\ \\ \end{matrix} \quad \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} \\ \\ \\ \leftarrow 6830 \\ \\ \\ \end{matrix} \quad \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} \\ \\ \leftarrow 367 \\ \\ \\ \end{matrix} \quad \updownarrow 10{,}000 \quad \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

One-hot

Andrew Ng

Representing words

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

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

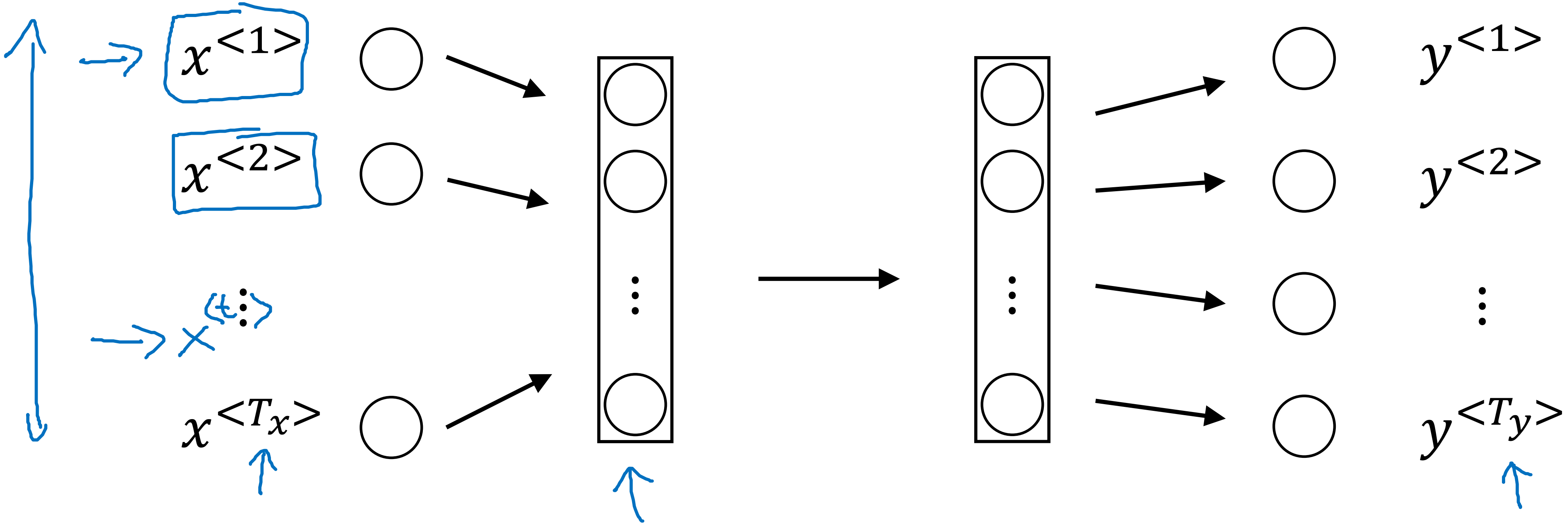
And = 367
Invented = 4700
A = 1
New = 5976
Spell = 8376
Harry = 4075
Potter = 6830
Hermione = 4200
Gran… = 4000

deeplearning.ai

Recurrent Neural Networks

Recurrent Neural Network Model

Why not a standard network?



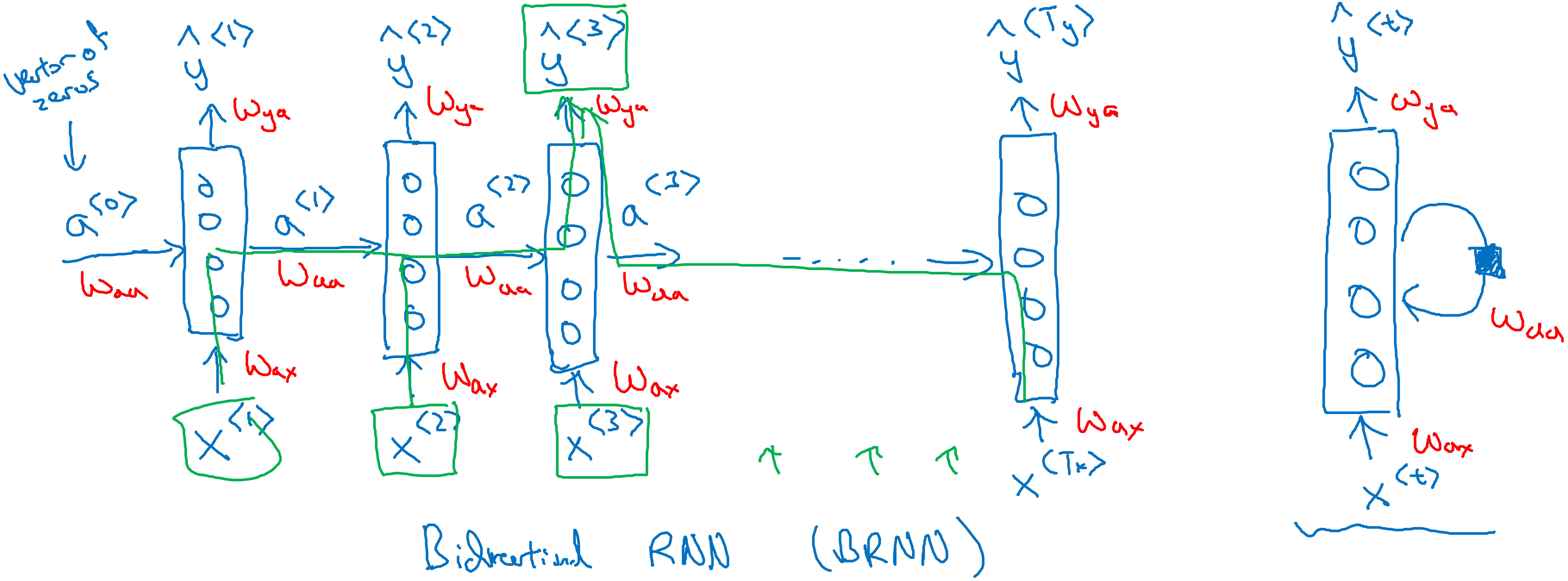
Problems:

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

Andrew Ng

Recurrent Neural Networks

$T_x = T_y$



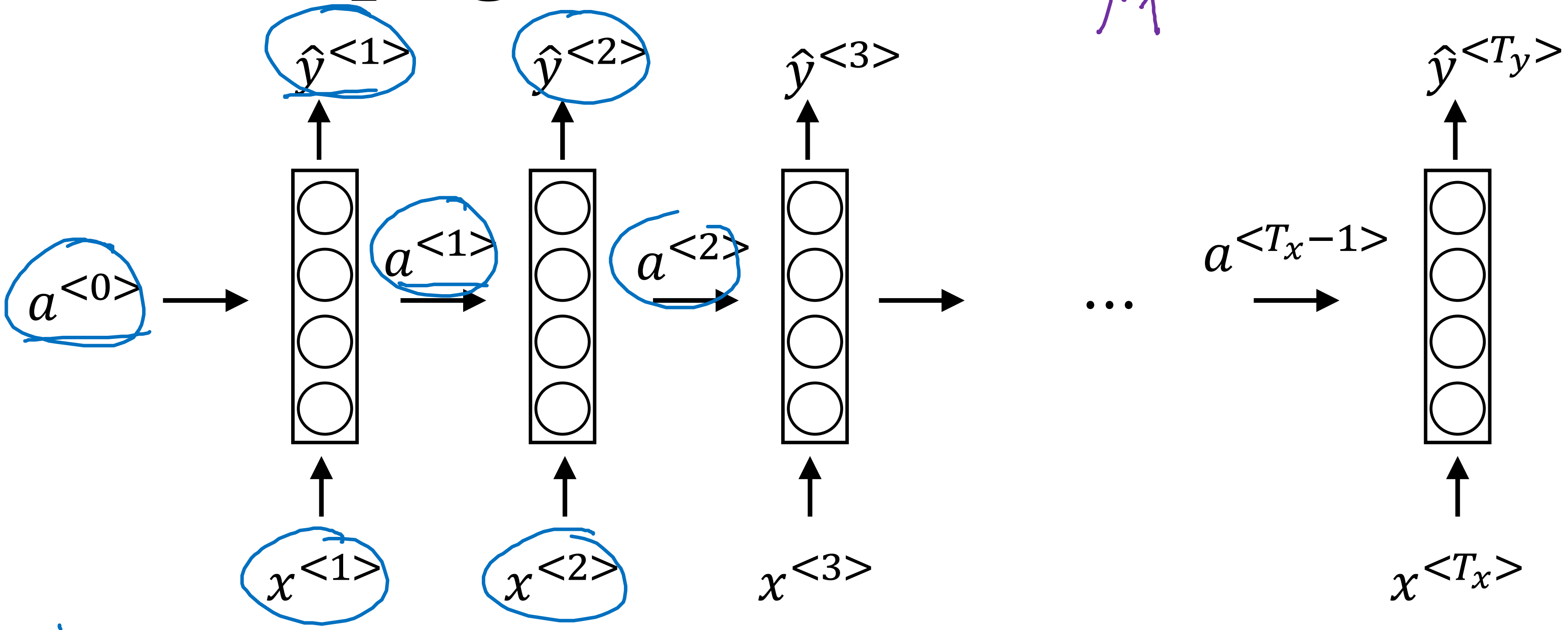
He said, “Teddy Roosevelt was a great President.”

He said, “Teddy bears are on sale!”

Andrew Ng

Forward Propagation

$a \leftarrow W_{ax} x^{<1>}$



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

$a^{<1>} = g_1(W_{aa} a^{<0>} + W_{ax} x^{<1>} + b_a) \leftarrow$ tanh / ReLU

$\hat{y}^{<1>} = g_2(W_{ya} a^{<1>} + b_y) \leftarrow$ sigmoid

$$\boxed{\begin{aligned} a^{<t>} &= g(W_{aa} a^{<t-1>} + W_{ax} x^{<t>} + b_a) \\ \hat{y}^{<t>} &= g(W_{ya} a^{<t>} + b_y) \end{aligned}}$$

Andrew Ng

Simplified RNN notation

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

W_{aa}: (100,100), $a^{<t-1>}$: 100; W_{ax}: (100,10,000), $x^{<t>}$: 10,000

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

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

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

$$\left[\, W_{aa} \,\vdots\, W_{ax} \,\right] = W_a$$

100 rows; W_{aa}: 100 columns, W_{ax}: 10 000 columns; W_a: (100, 10100)

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

$a^{<t-1>}$: 100, $x^{<t>}$: 10000, total: 10100

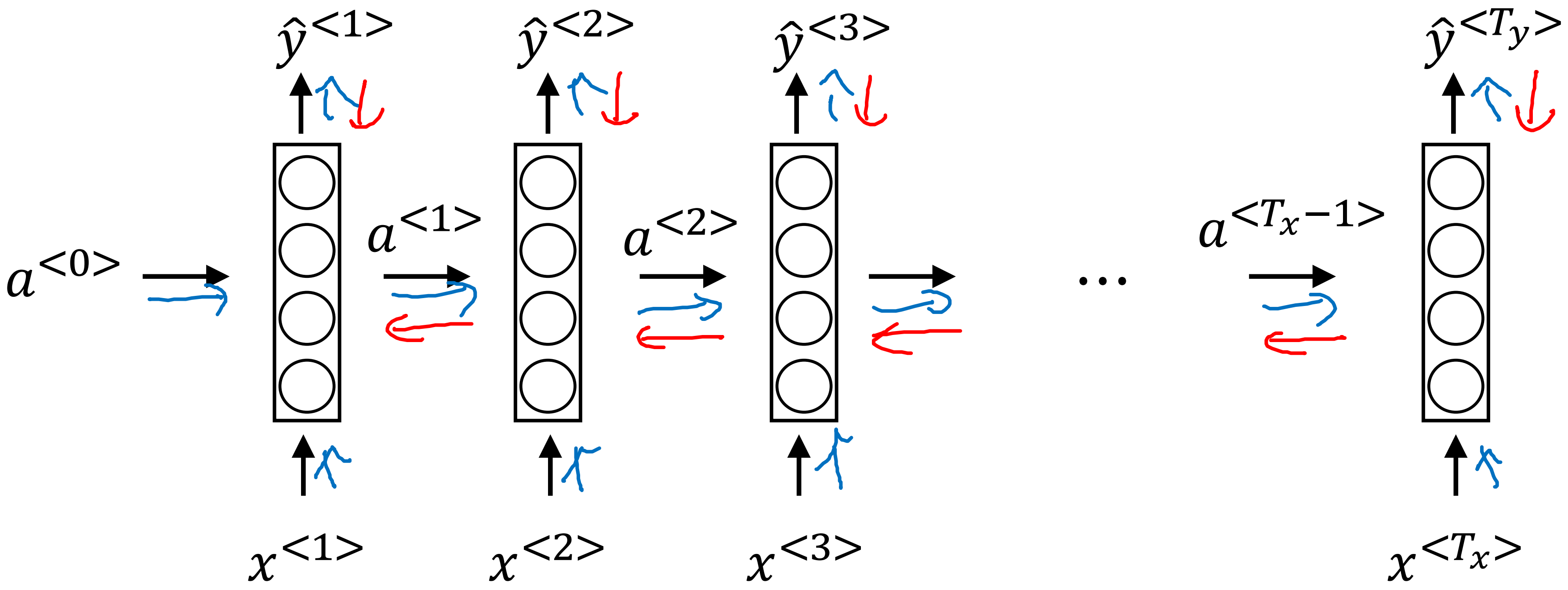
$$\left[W_{aa} \,\vdots\, W_{ax} \right] \begin{bmatrix} a^{<t-1>} \\ x^{<t>} \end{bmatrix} = W_{aa}a^{<t-1>} + W_{ax}x^{<t>}$$

deeplearning.ai

Recurrent Neural Networks

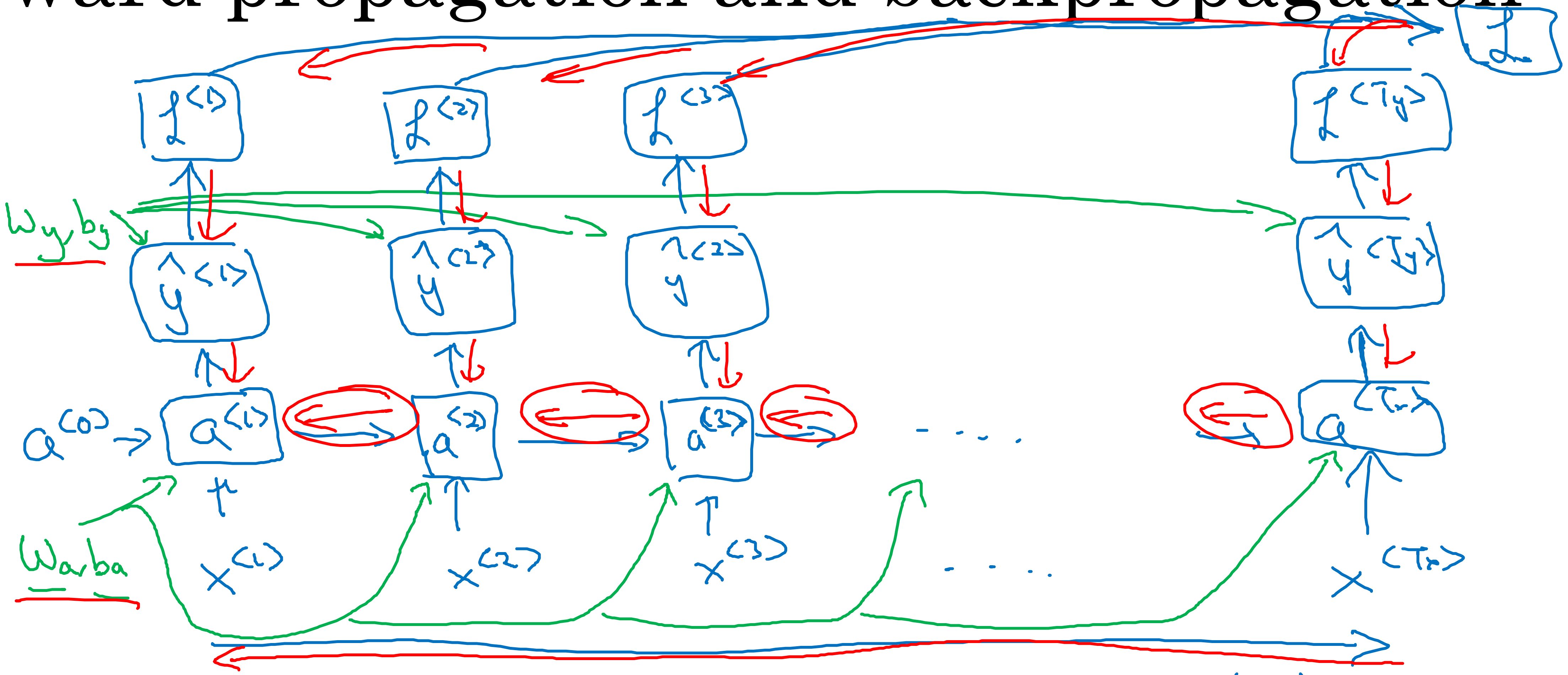
Backpropagation through time

Forward propagation and backpropagation



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

Andrew Ng

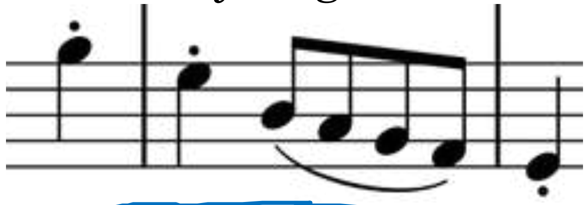
deeplearning.ai

Recurrent Neural Networks

Different types of RNNs

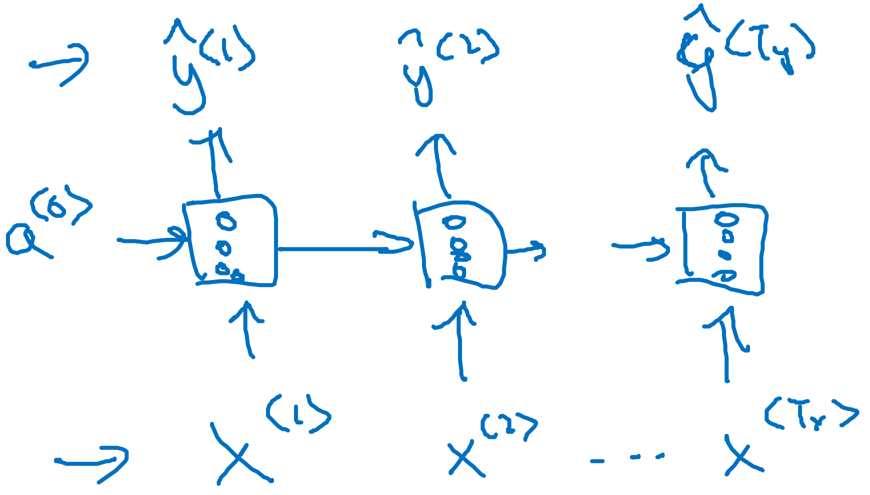
Examples of sequence data

T_x T_y

	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.

Examples of RNN architectures

$T_x = T_y$

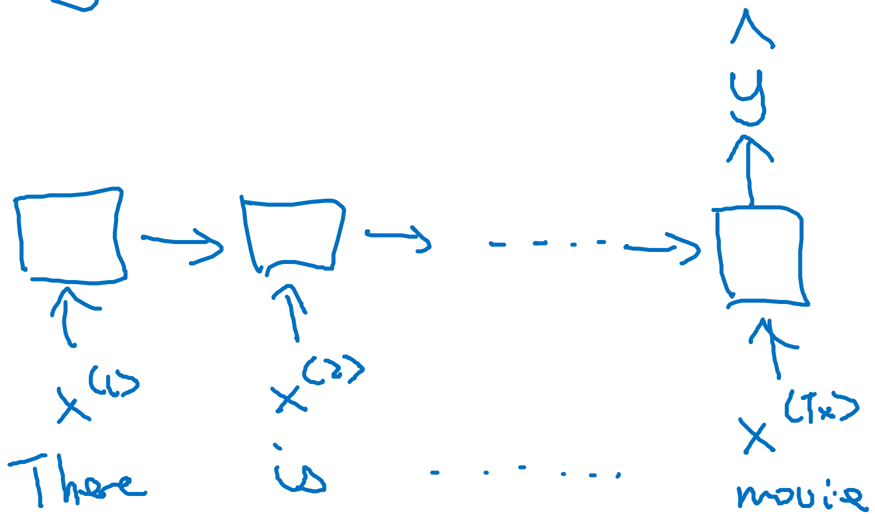


Many-to-many

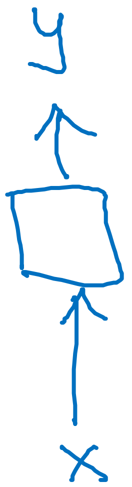
Sentiment classification

x = text

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



Many-to-one



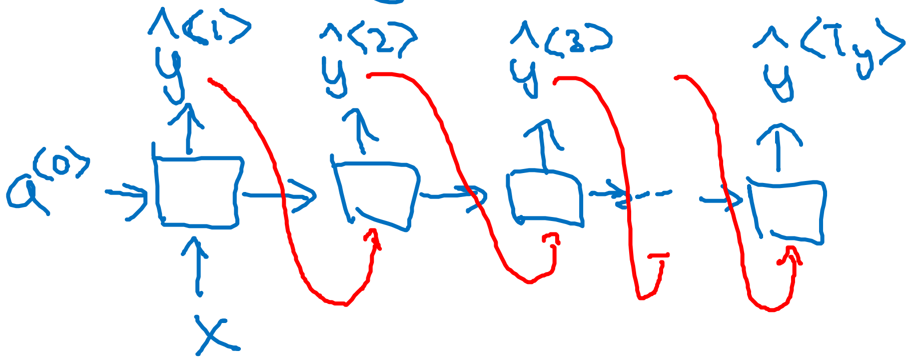
one-to-one

Andrew Ng

Examples of RNN architectures

Music generation

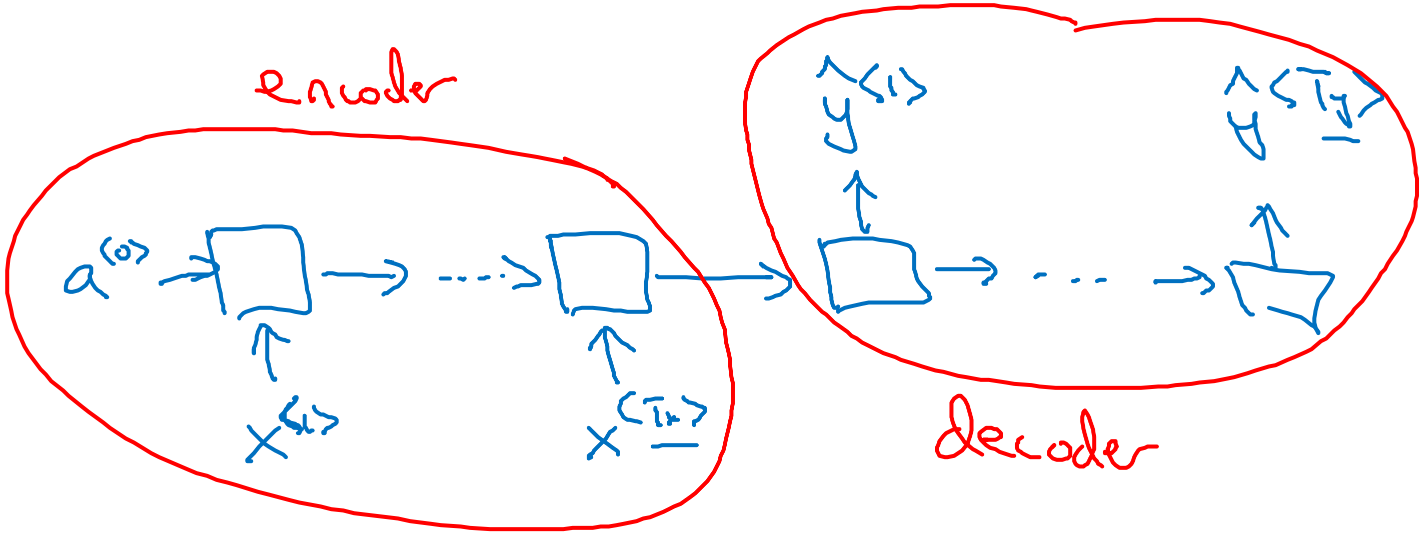
$x \rightarrow y^{<1>}\ y^{<2>} \dots y^{<T_y>}$



One-to-many

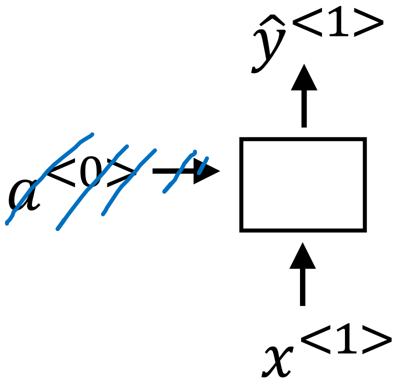
$x = \emptyset$

Machine translation

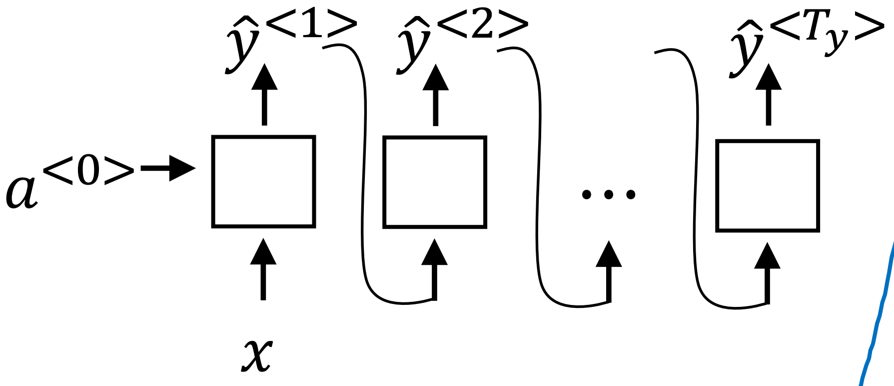


Many-to-many

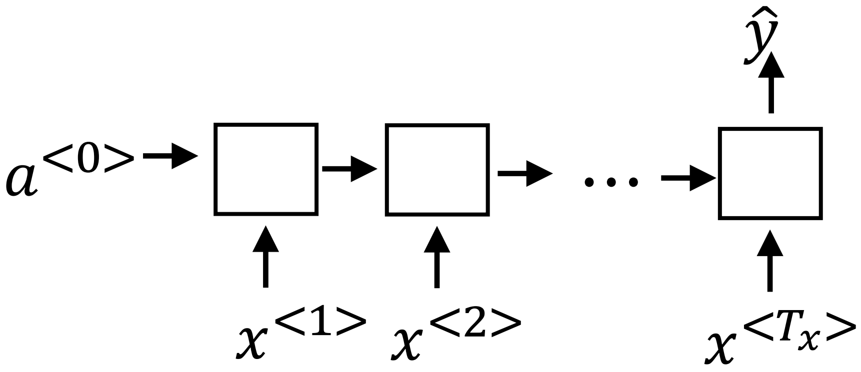
Summary of RNN types



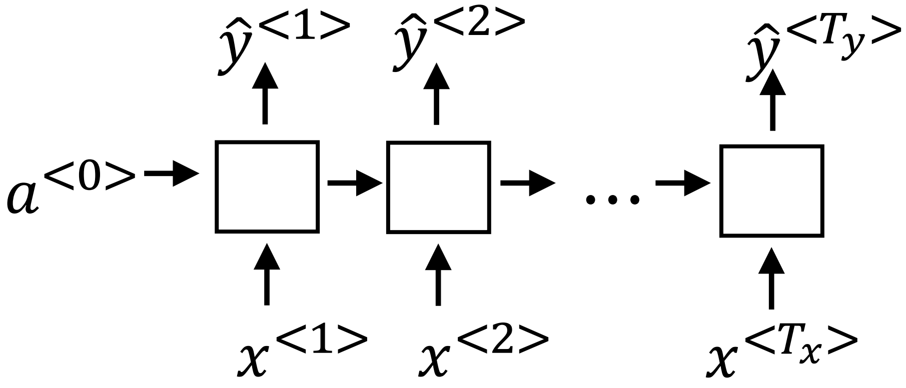
One to one



One to many

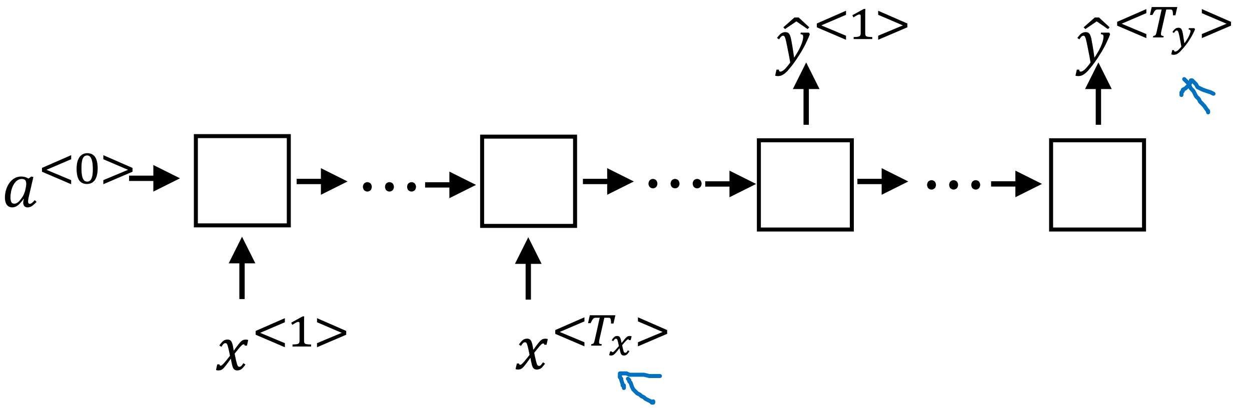


Many to one



Many to many

$T_x = T_y$



Many to many

Andrew Ng

Recurrent Neural Networks

deeplearning.ai

Language model and sequence generation

What is language modelling?

Speech recognition

The apple and pair salad.

→ The apple and pear salad.

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

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

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

$P(y^{<1>}, y^{<2>}, \ldots, y^{<T_y>})$

Andrew Ng

Language modelling with an RNN

Training set: large corpus of english text.

Tokenize

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

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

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

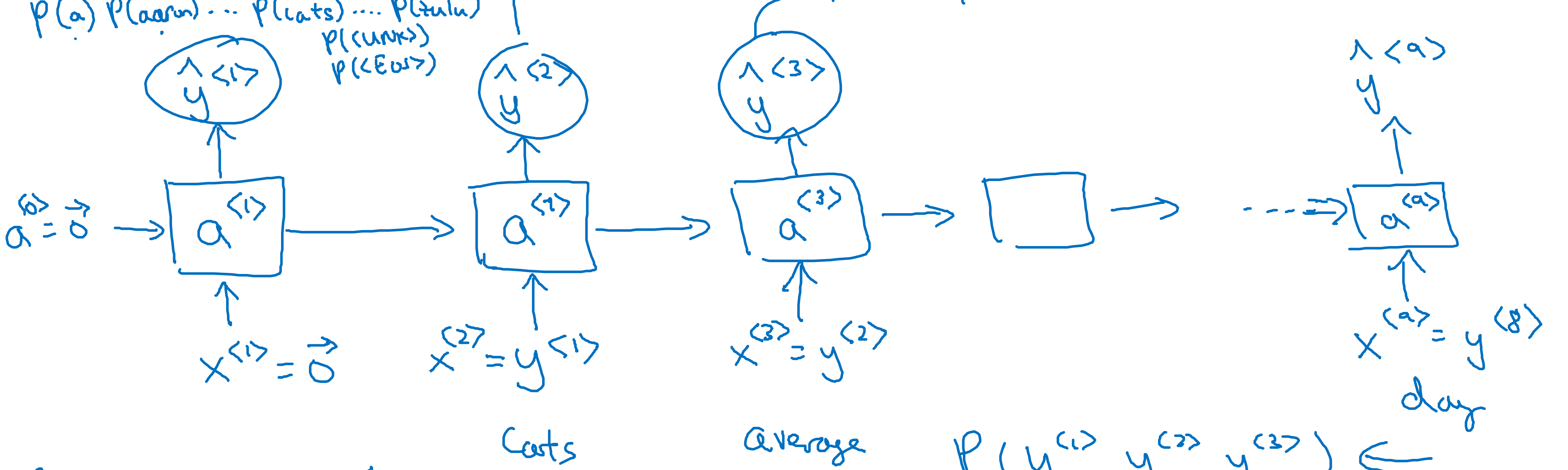
The Egyptian ~~Mau~~ is a bread of cat. <EOS>

<UNK>

10,000

Andrew Ng

RNN model



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

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

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

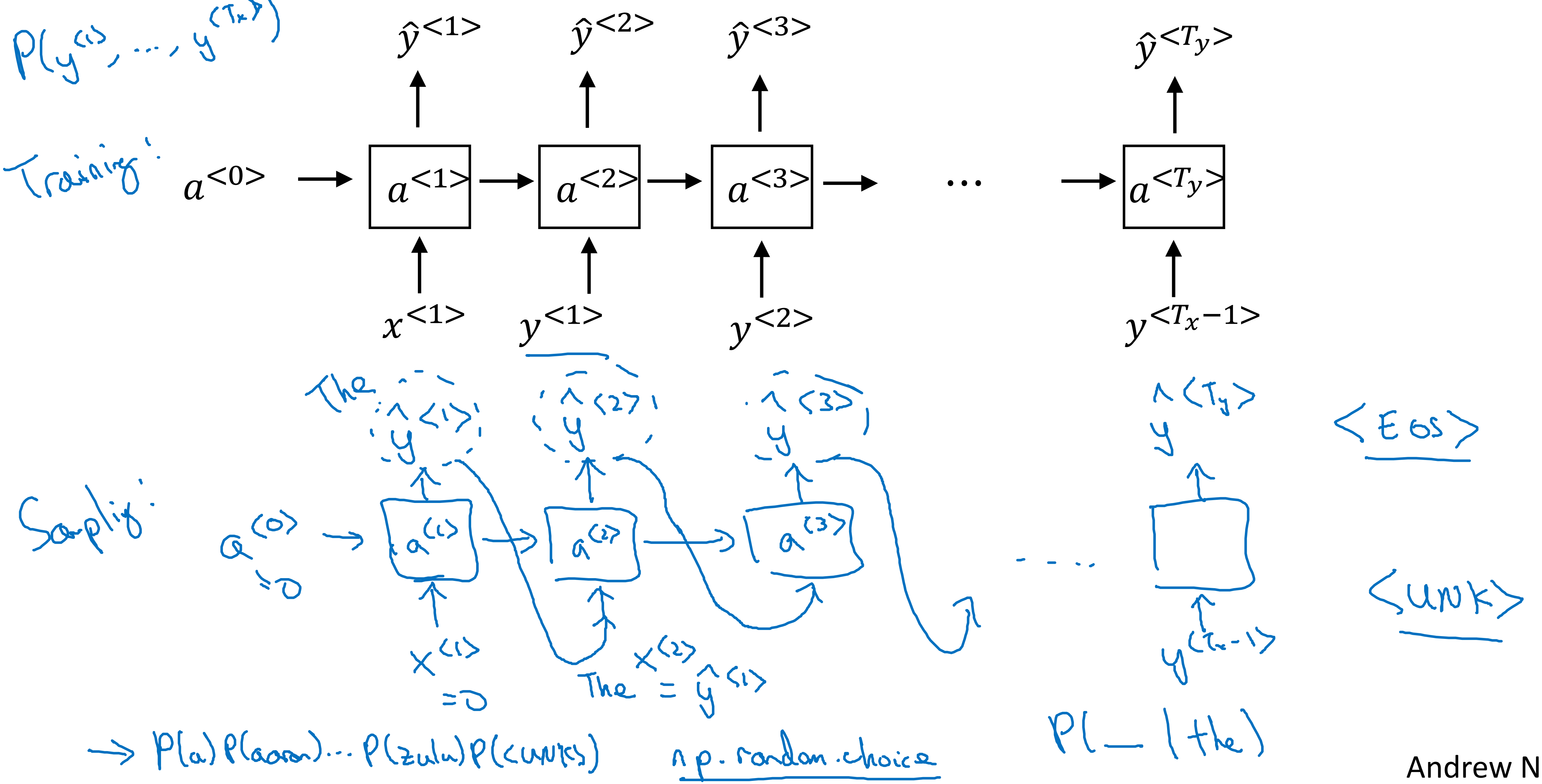
$$\mathcal{L} = \sum_t \mathcal{L}^{<t>}(\hat{y}^{<t>}, y^{<t>})$$

deeplearning.ai

Recurrent Neural Networks

Sampling novel sequences

Sampling a sequence from a trained RNN

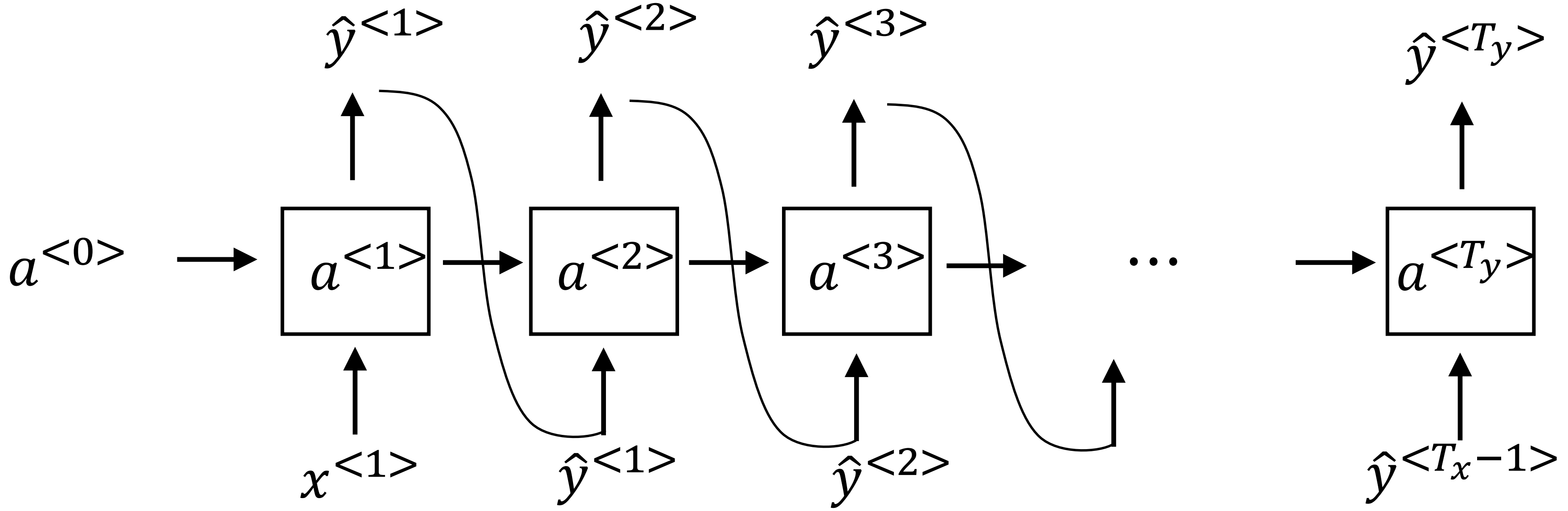


Character-level language model

→ 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.

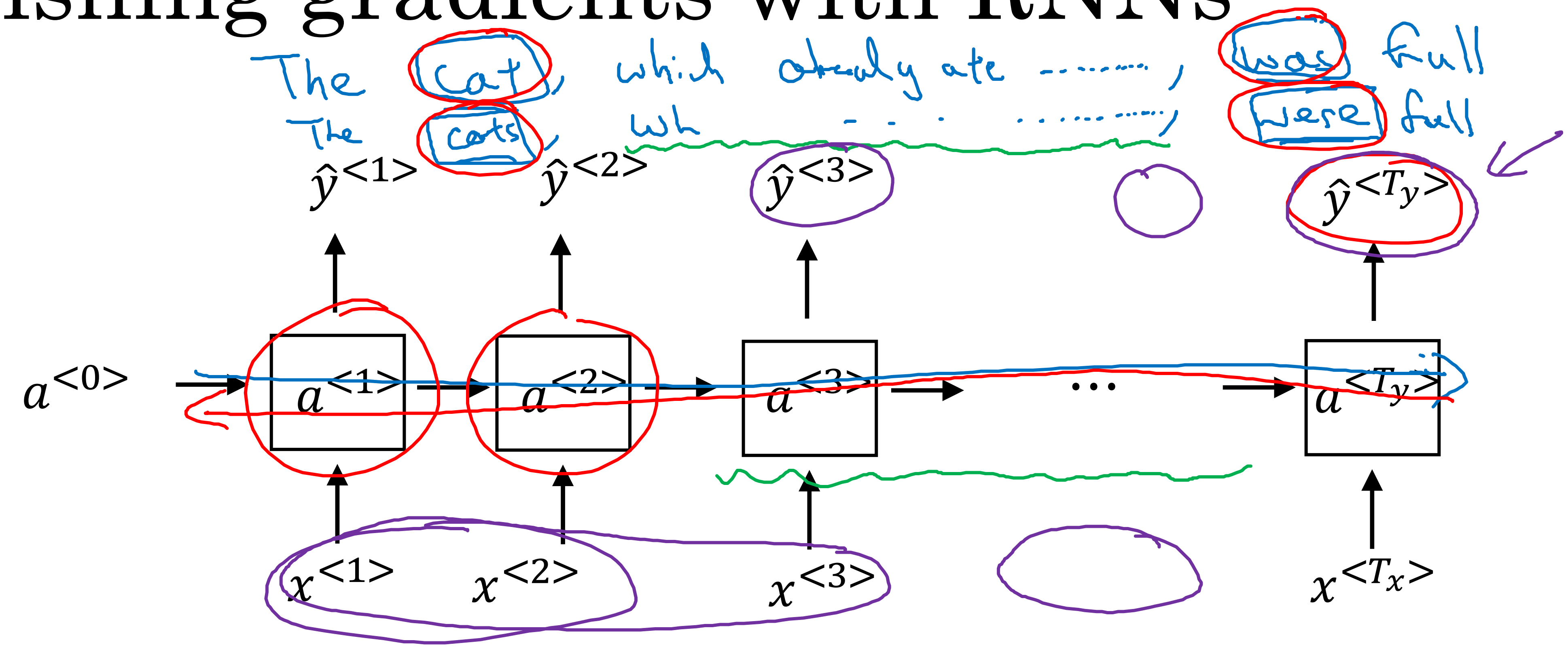
For whose are ruse of mine eyes heaves.

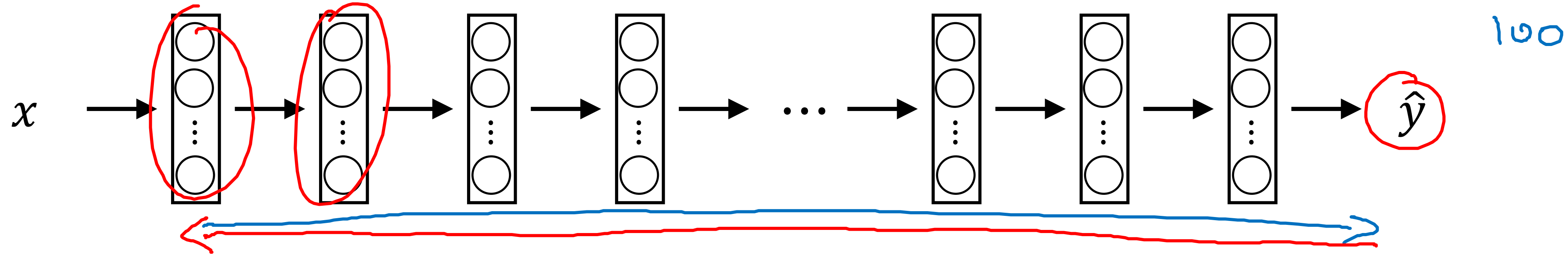
Recurrent Neural Networks

Vanishing gradients with RNNs

deeplearning.ai

Vanishing gradients with RNNs





Exploding gradients. NaN Gradient clipping

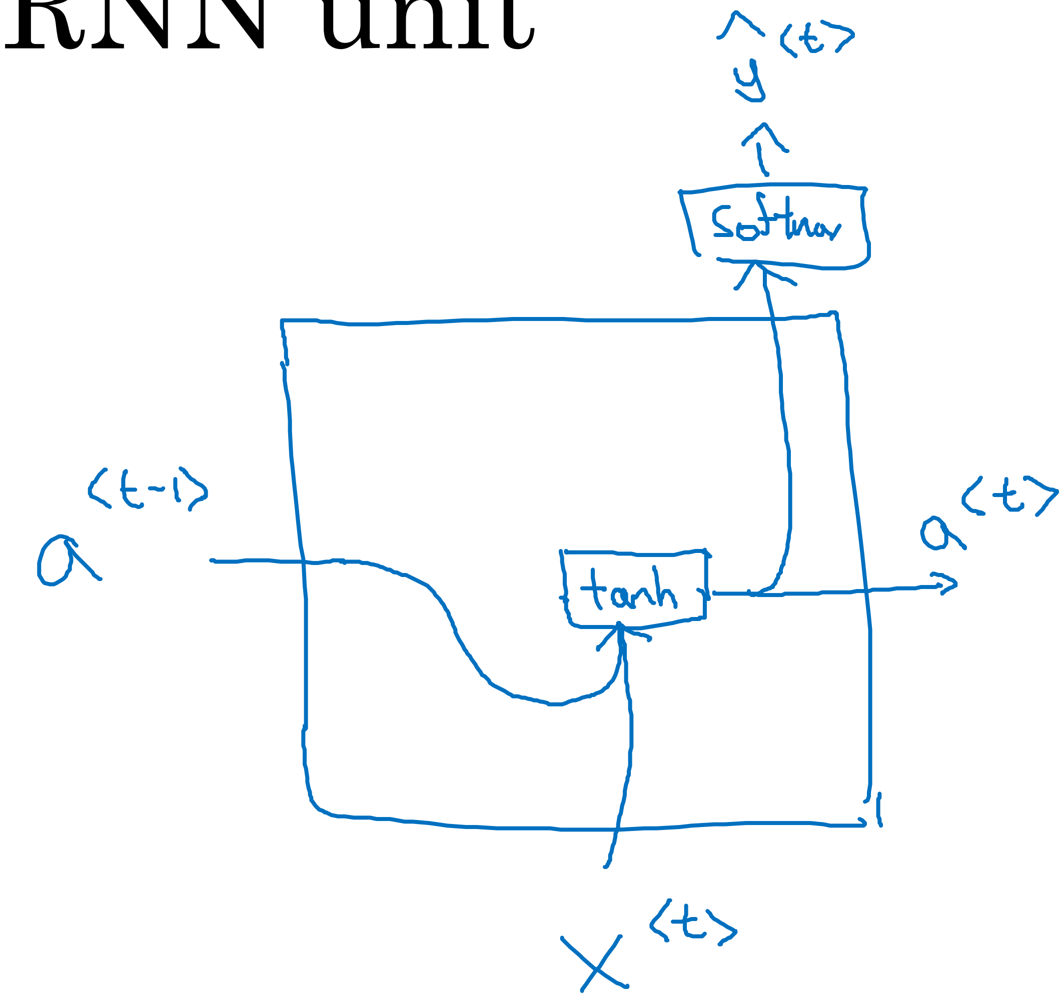
Andrew Ng

deeplearning.ai

Recurrent Neural Networks

Gated Recurrent Unit (GRU)

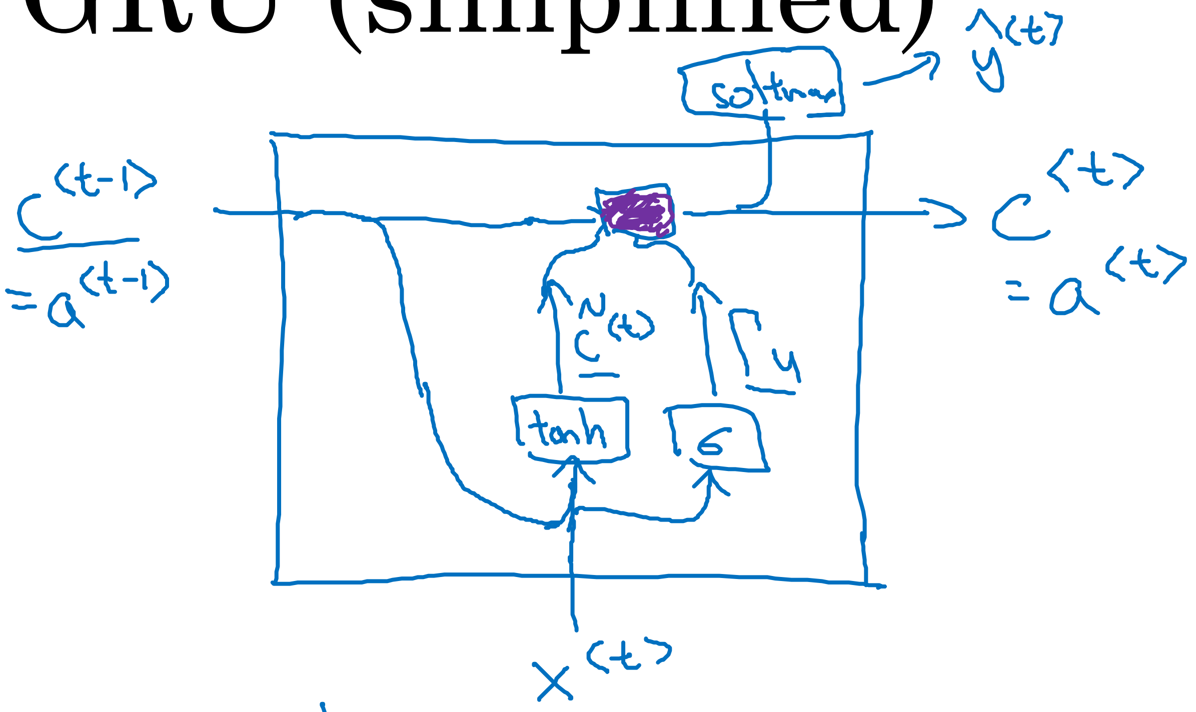
RNN unit

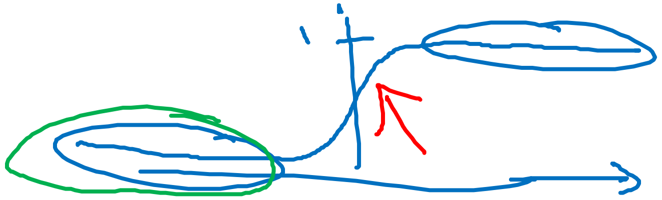


tanh

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

GRU (simplified)





c = memory cell

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

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

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

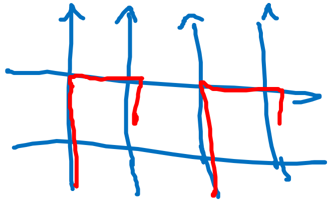
"update"

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

$\Gamma_u = 1$

element-wise

$\Gamma_u = 0.000001$



Gate

$\Gamma_u = 1$ $\quad \Gamma_u = 0 \quad \Gamma_u = 0 \quad \Gamma_u = 0 \; \ldots$

$c^{<t>} = 1 \; \ldots \ldots \; = 1$

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]

Full GRU

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

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.

Recurrent Neural Networks

deeplearning.ai

LSTM (long short term memory) unit

GRU and LSTM

GRU

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

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

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

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

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

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

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

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

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

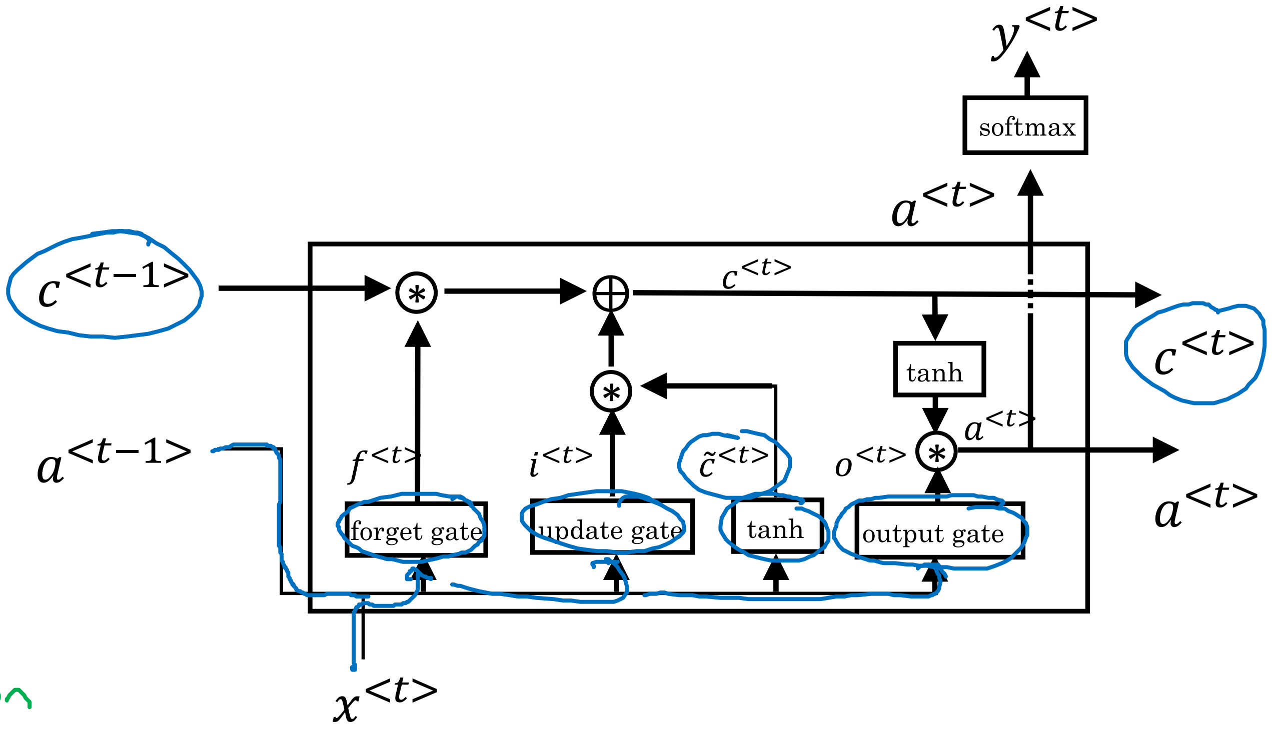
$$\Gamma_o = \sigma(W_o[\, a^{<t-1>}, x^{<t>}] + b_o)$$

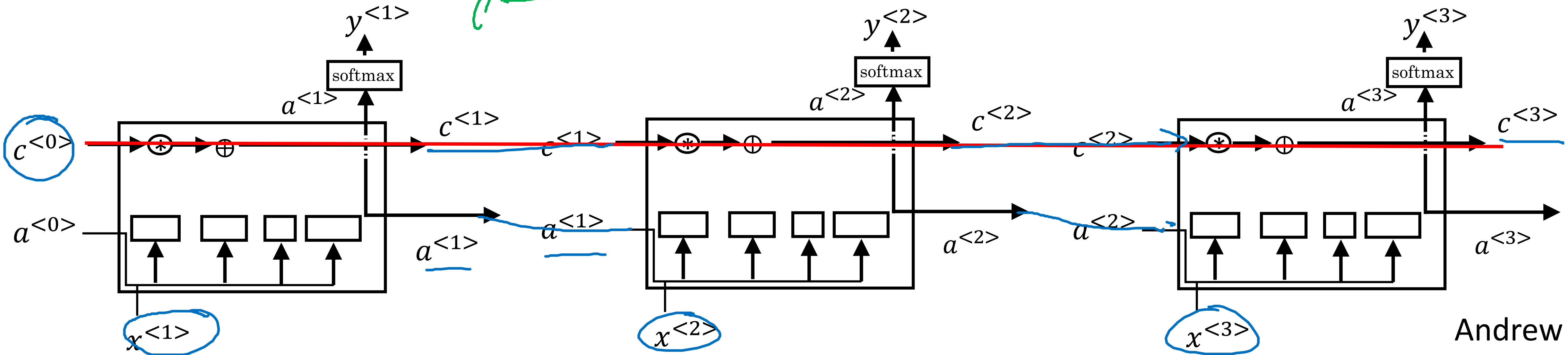
$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + \Gamma_f * c^{<t-1>}$$

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

$\boxed{c^{<t-1>}}$

peephole connection





Andrew Ng

Recurrent Neural Networks

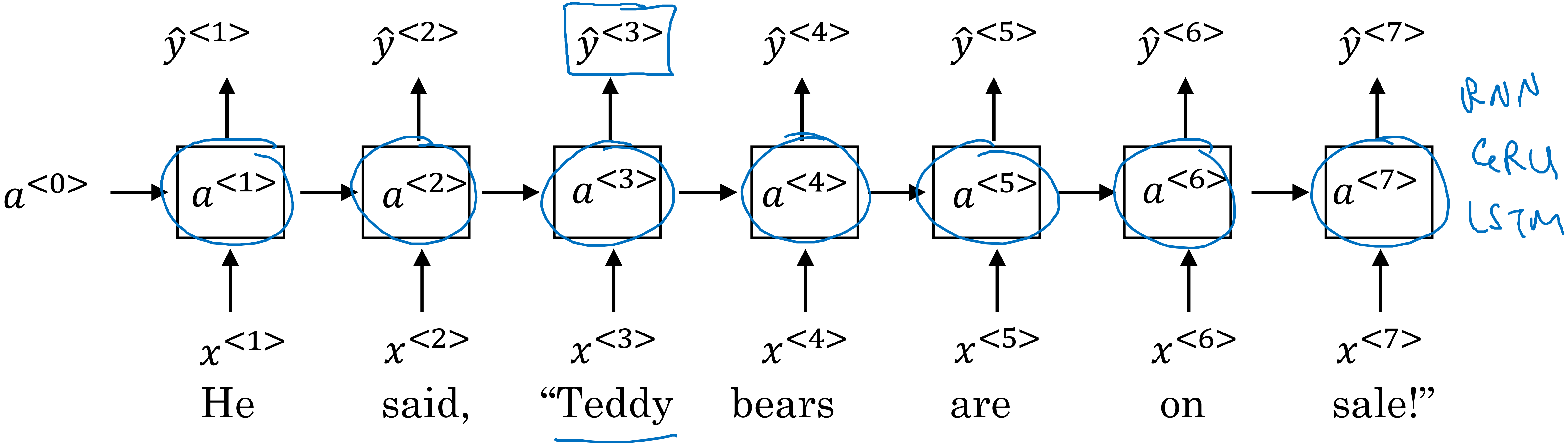
Bidirectional RNN

deeplearning.ai

Getting information from the future

He said, "Teddy bears are on sale!"

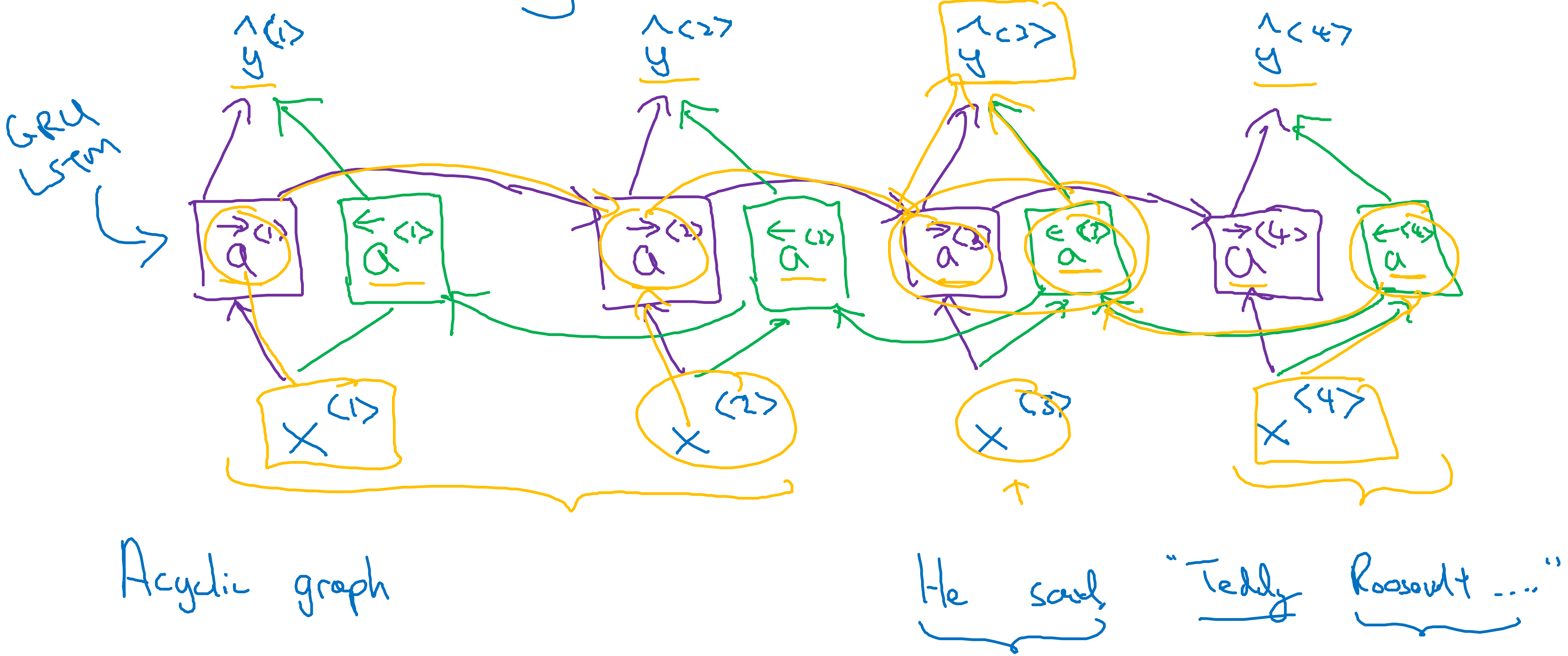
He said, "Teddy Roosevelt was a great President!"



Andrew Ng

Bidirectional RNN (BRNN)

$$\hat{y}^{<t>} = g(W_y[\overrightarrow{a}^{<t>}, \overleftarrow{a}^{<t>}] + b_y)$$



BRNN w/ LSTM

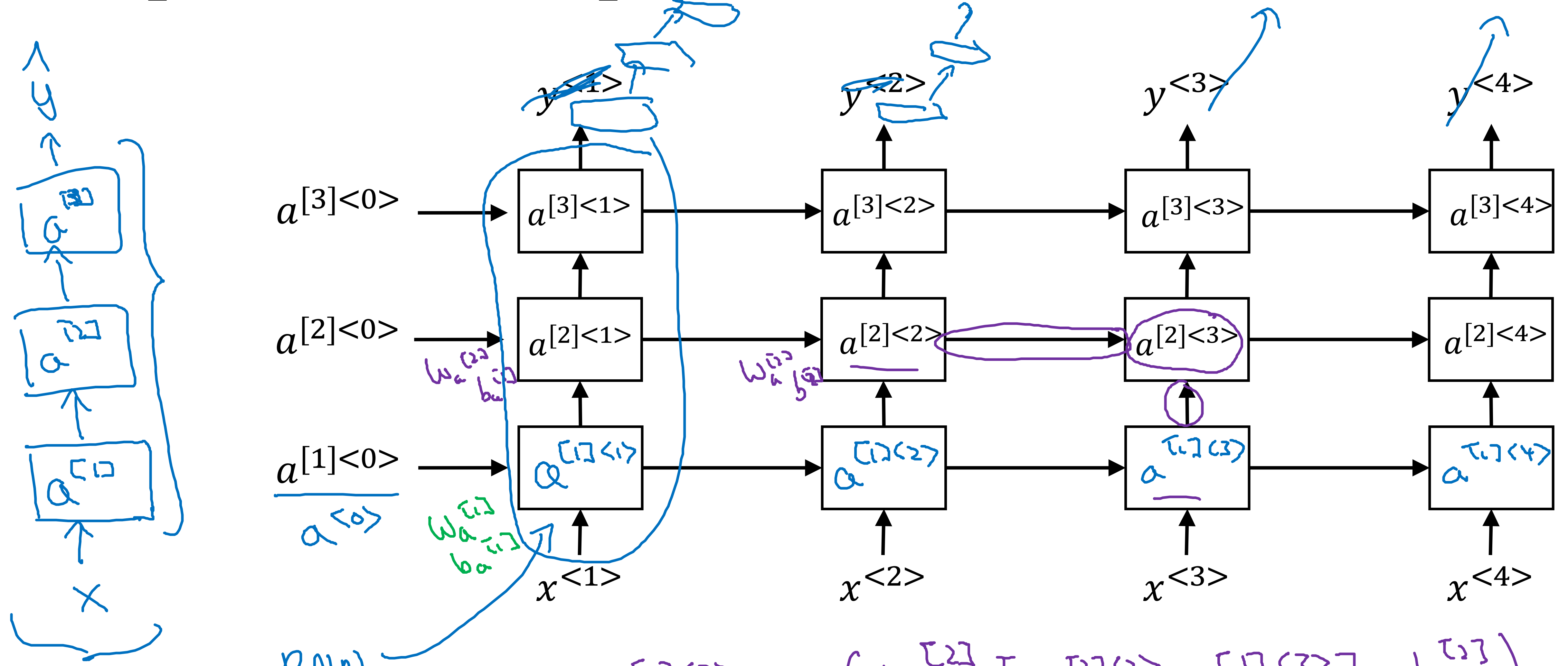
Andrew Ng

deeplearning.ai

Recurrent Neural Networks

Deep RNNs

Deep RNN example



RNN
GRU
LSTM

BRNN

$$a^{[2]<3>} = g\left(W_a^{[2]}\left[a^{[2]<2>}, a^{[1]<3>}\right] + b_a^{[2]}\right)$$

NLP and Word Embeddings

deeplearning.ai

Word representation

Word representation

V = [a, aaron, …, zulu, <UNK>]

$|V| = 10,000$

1-hot representation

$$\begin{array}{cccccc}
\text{Man} & \text{Woman} & \text{King} & \text{Queen} & \text{Apple} & \text{Orange} \\
(5391) & (9853) & (4914) & (7157) & (456) & (6257) \\
\begin{bmatrix}0\\0\\0\\0\\\vdots\\1\\\vdots\\0\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\1\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\\vdots\\1\\\vdots\\0\\0\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\1\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\\vdots\\1\\\vdots\\0\\0\\0\\0\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\1\\\vdots\\0\end{bmatrix} \\
O_{5391} & O_{9853} & & & &
\end{array}$$

I want a glass of orange juice.

I want a glass of apple ? .

Andrew Ng

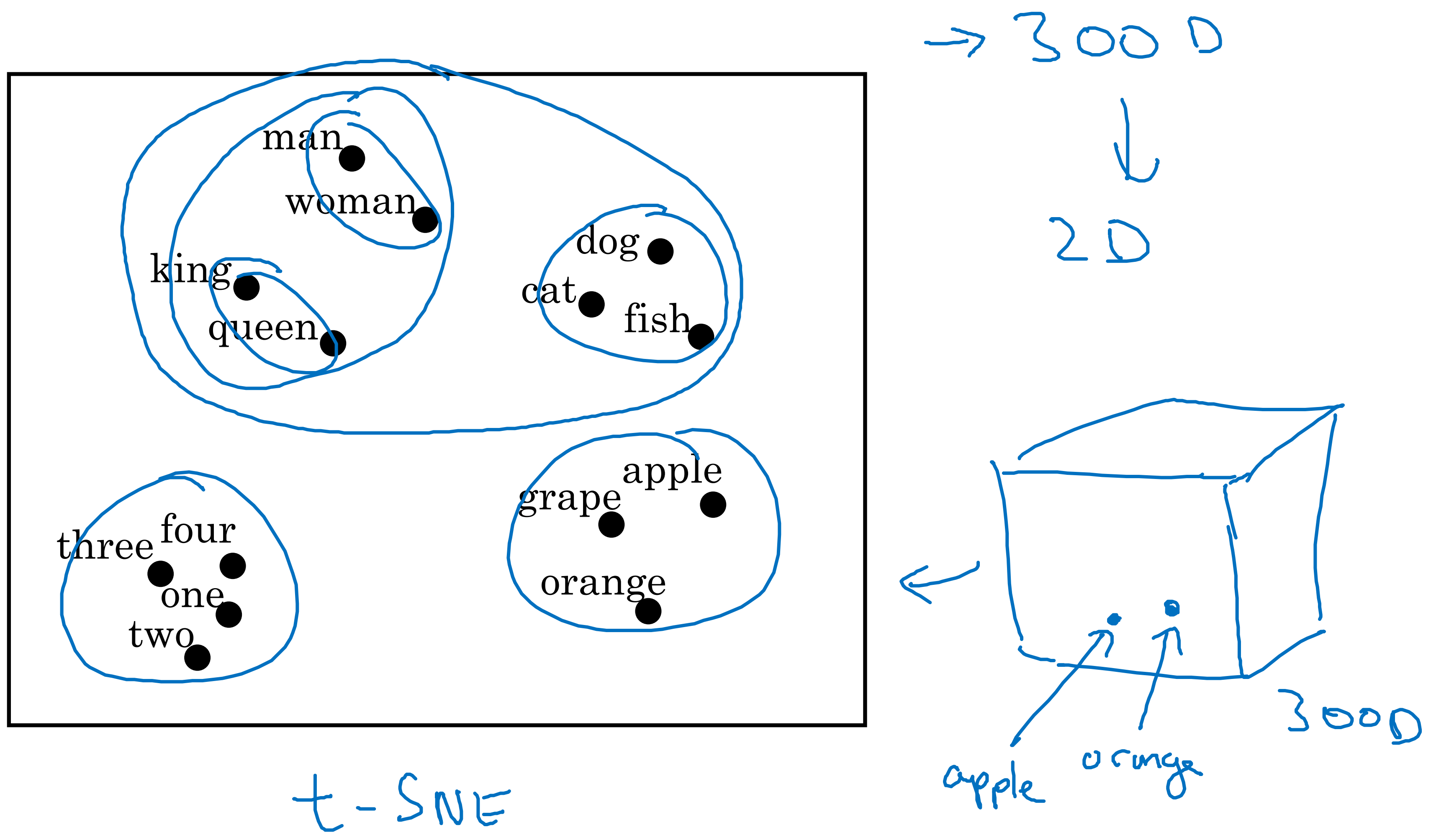
Featurized representation: word embedding

300	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	-1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03	0.02	0.7	0.69	0.03	-0.02
Food	0.04	0.01	0.02	0.01	0.95	0.97
⋮	⋮	⋮				
size						
cost						
alive						
verb						
	e_{5391}	e_{9853}				

I want a glass of orange juice.

I want a glass of apple juice.

Visualizing word embeddings



[van der Maaten and Hinton., 2008. Visualizing data using t-SNE]

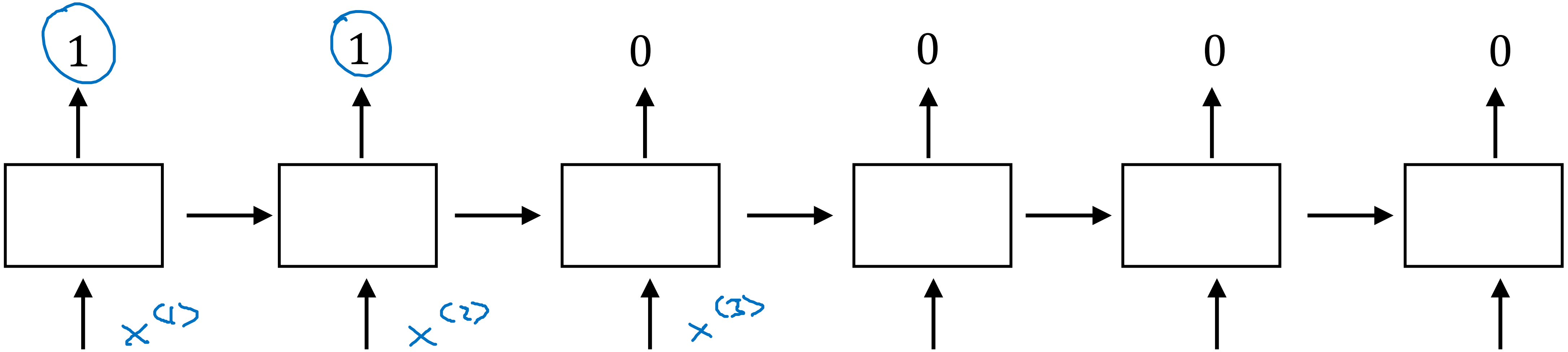
Andrew Ng

deeplearning.ai

NLP and Word Embeddings

Using word embeddings

Named entity recognition example



Sally Johnson is an orange farmer

Robert Lin is ~~an~~ ~~apple farmer~~

a durian cultivator.

→ 1B words – 100B words

→ 100k words

BRNN

Andrew Ng

Transfer learning and word embeddings

A

1. Learn word embeddings from large text corpus. (1-100B words)

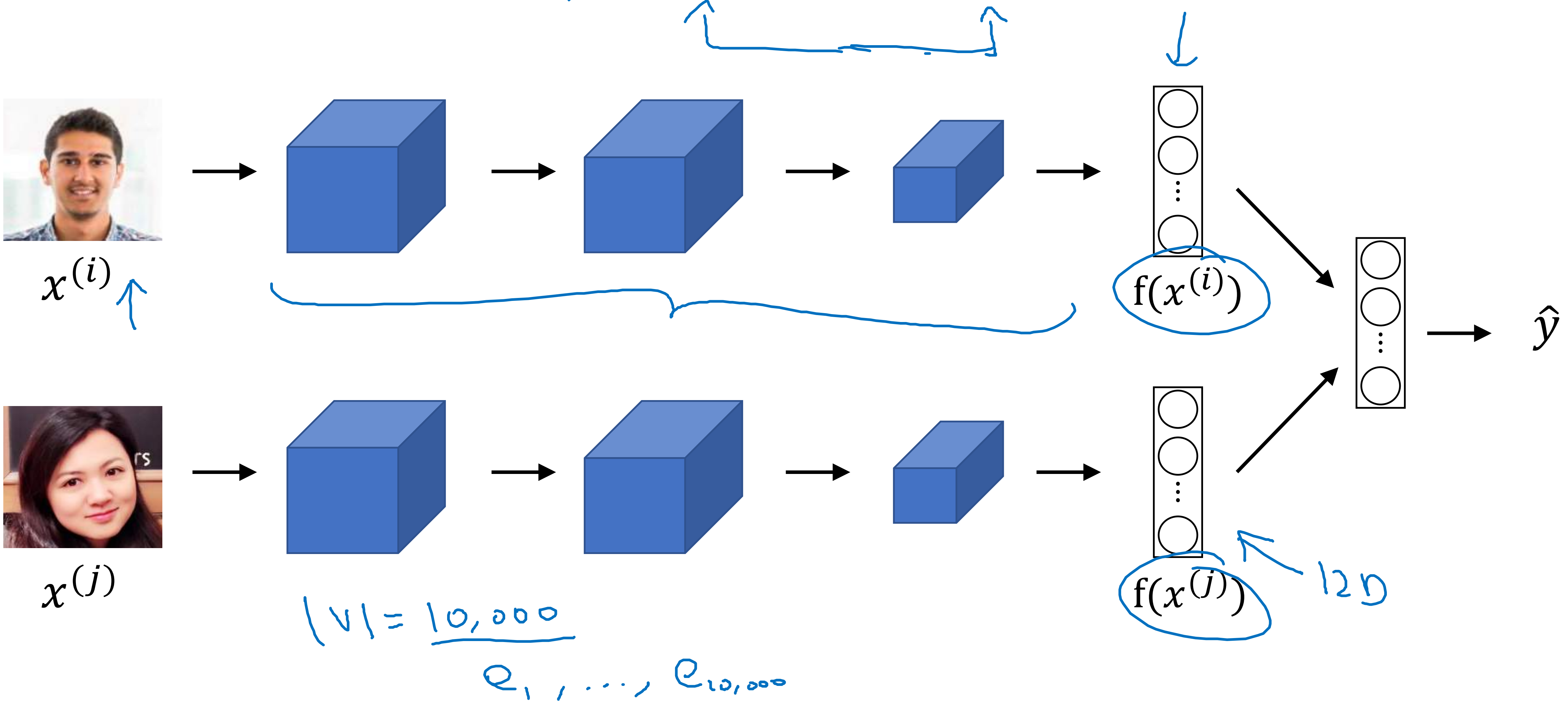
   (Or download pre-trained embedding online.)

B

2. Transfer embedding to new task with smaller training set. (say, 100k words) → 10,000 → 300

3. Optional: Continue to finetune the word embeddings with new data.

Relation to face encoding (embedding)



[Taigman et. al., 2014. DeepFace: Closing the gap to human level performance]

deeplearning.ai

NLP and Word Embeddings

Properties of word embeddings

Analogies

	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	−1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03	0.02	0.70	0.69	0.03	-0.02
Food	0.09	0.01	0.02	0.01	0.95	0.97

e_{5391}

e_{man} $\qquad e_{woman}$

Man → Woman as King →? Queen

$$e_{man} - e_{woman} \approx e_{king} - e_{?}$$

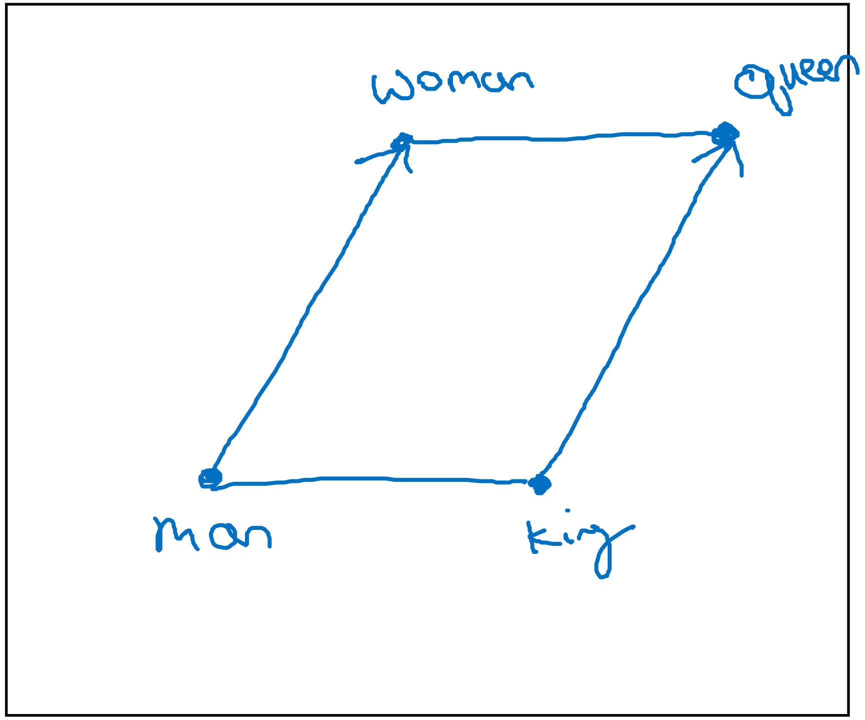
$e_{?}$ ← queen

$$e_{man} - e_{woman} \approx \begin{bmatrix} -2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$e_{king} - e_{queen} \approx \begin{bmatrix} -2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

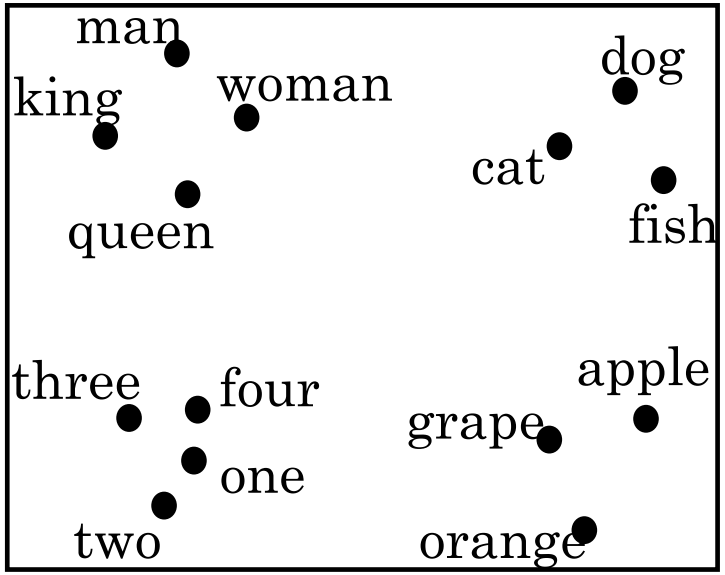
[Mikolov et. al., 2013, Linguistic regularities in continuous space word representations]

Analogies using word vectors



300 D

300 → 2D



t-SNE

$$e_{man} - e_{woman} \approx e_{king} - \cancel{e_?}\ e_w$$

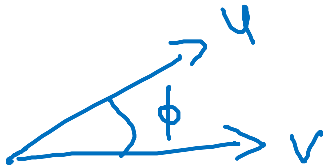
Find word w: $\arg\max_w \boxed{sim(e_w, e_{king} - e_{man} + e_{woman})}$

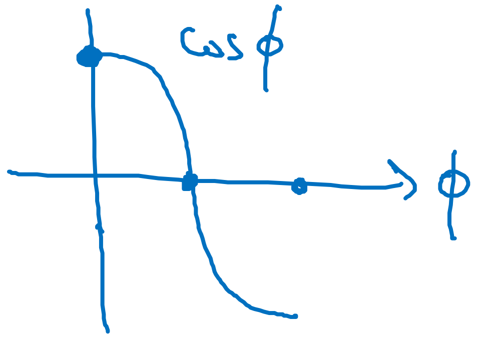
30 - 75%

Cosine similarity

$$\boxed{sim(e_w, e_{king} - e_{man} + e_{woman})}$$

$$\underline{sim(u,v)} = \frac{u^T v}{\|u\|_2 \|v\|_2}$$





$$\|u - v\|^2$$

Man:Woman as Boy:Girl

Ottawa:Canada as Nairobi:Kenya

Big:Bigger as Tall:Taller

Yen:Japan as Ruble:Russia

NLP and Word Embeddings

deeplearning.ai

Embedding matrix

Embedding matrix

$E \cdot O_{6257} = e_{6257}$

(300, 10K) (10K,1)

(300,1)

$\rightarrow E \cdot O_j = e_j$

= embedding for word j

In practice, use specialized function to look up an embedding.

→ Embedding

deeplearning.ai

NLP and Word Embeddings

Learning word embeddings

Neural language model

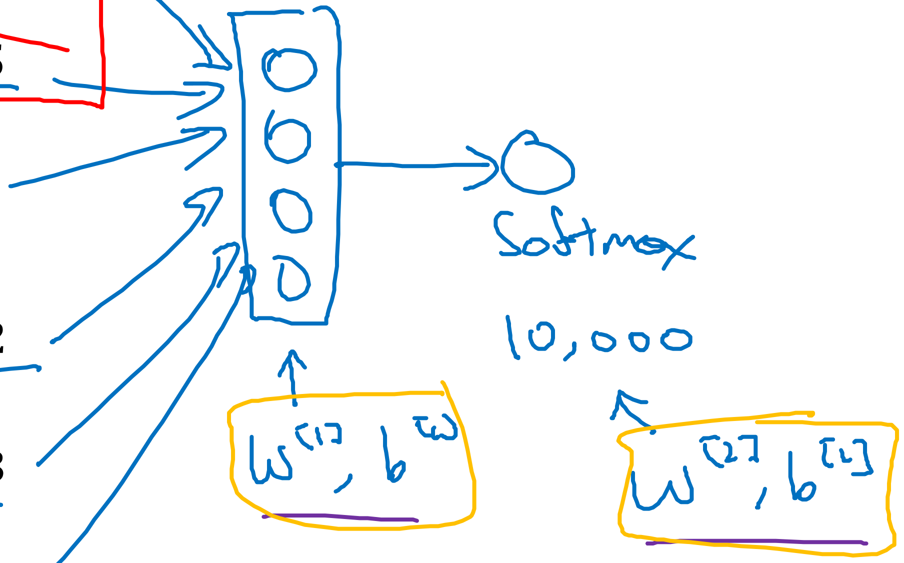
4

I	want	a	glass	of	orange	juice .
4343	9665	1	3852	6163	6257	

$e_{4343} = E\, o_{4343}$

apple juice .

I	o_{4343}	$\longrightarrow$	E	$\longrightarrow$	e_{4343}
want	o_{9665}	$\longrightarrow$	E	$\longrightarrow$	e_{9665}
a	o_{1}	$\longrightarrow$	E	$\longrightarrow$	e_{1}
glass	o_{3852}	$\longrightarrow$	E	$\longrightarrow$	e_{3852}
of	o_{6163}	$\longrightarrow$	E	$\longrightarrow$	e_{6163}
orange	o_{6257}	$\longrightarrow$	E	$\longrightarrow$	e_{6257}



~~1800~~ 1200

[Bengio et. al., 2003, A neural probabilistic language model]

Other context/target pairs

I want a glass of orange juice to go along with my cereal.

context

target

Context: Last 4 words.

- 4 words on left & right
- Last 1 word
- Nearby 1 word

a glass of orange ___?___ to go along with

orange ___?___

glass ___?___

skip gram

deeplearning.ai

NLP and Word Embeddings

Word2Vec

Skip-grams

I want a glass of orange juice to go along with my cereal.

Context	Target
orange	juice
orange	glass
orange	my

[Mikolov et. al., 2013. Efficient estimation of word representations in vector space.]

Andrew Ng

Model

Vocab size = 10,000k

$x \longrightarrow y$

Context c ("orange") 6257 $\longrightarrow$ Target t ("juice") 4834

$$o_c \longrightarrow E \longrightarrow e_c \longrightarrow \underset{\text{softmax}}{O} \longrightarrow \hat{y}$$

$$e_c = E o_c$$

Softmax: $$p(t|c) = \frac{e^{\theta_t^T e_c}}{\sum_{j=1}^{10,000} e^{\theta_j^T e_c}}$$

θ_t = parameter associated with output t

$$\rightarrow \mathcal{L}(\hat{y}, y) = -\sum_{i=1}^{10,000} y_i \log \hat{y}_i$$

$$y = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \leftarrow 4834$$

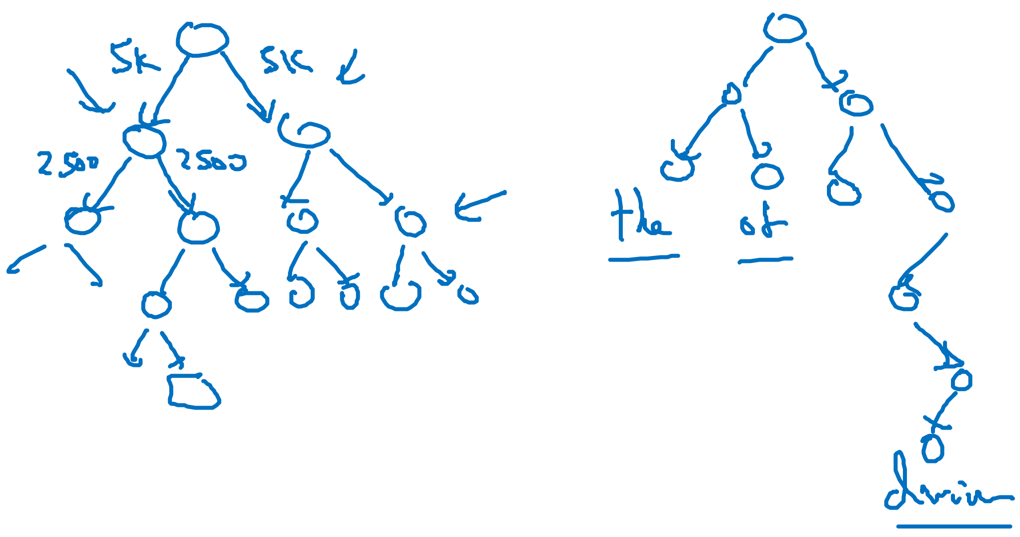
Andrew Ng

Problems with softmax classification

Hierarchical softmax.

$$p(t|c) = \frac{e^{\theta_t^T e_c}}{\boxed{\sum_{j=1}^{10,000} e^{\theta_j^T e_c}}}$$

$\log |V|$



How to sample the context c?

t

→ the, of, a, and, to, ...

$c \rightarrow t$

→ orange, apple, durian

$P(c)$

durian

NLP and Word Embeddings

Negative sampling

deeplearning.ai

Defining a new learning problem

I want a glass of orange juice to go along with my cereal.

x → y

Context	word	target?
orange	juice	1
orange	king	0
orange	book	0
orange	the	0
orange	of	0

k

k = 5-20 smaller datasets

k = 2-5 larger dataset

[Mikolov et. al., 2013. Distributed representation of words and phrases and their compositionality]

Model

Softmax: $p(t|c) = \dfrac{e^{\theta_t^T e_c}}{\sum_{j=1}^{10{,}000} e^{\theta_j^T e_c}}$

10,000-way softmax

$P(y=1 \mid c, t) = \sigma(\theta_t^T e_c)$

context	word	target?
orange	juice	1
orange	king	0
orange	book	0
orange	the	0
orange	of	0

x, y

c, t, y

k

Orange 6257

$O_{6257} \rightarrow E \rightarrow e_{6257}$

juice?

king

10,000

10,000 binary classification problem

k+1

Andrew Ng

Selecting negative examples

$\overbrace{\text{context} \quad \text{word}}^{x}$		$\overbrace{\text{target?}}^{y}$
context	word	target?
orange	juice	1
orange	king	0
orange	book	0
orange	the	0
orange	of	0

t

the , of, and, ...

$$P(w_i) = \frac{f(w_i)^{3/4}}{\sum_{j=1}^{10{,}000} f(w_j)^{3/4}}$$

$$\frac{1}{|V|}$$

Andrew Ng

deeplearning.ai

NLP and Word Embeddings

GloVe word vectors

GloVe (global vectors for word representation)

I want a glass of orange juice to go along with my cereal.

c, t

X_{ij} = # times i appears in context of j.

$$\underset{c\ \ t}{X_{ij}} = \#\text{times } \underset{t}{i} \text{ appears in context of } \underset{c}{j}$$

$$\underbrace{X_{ij}}_{} = X_{ji}$$

[Pennington et. al., 2014. GloVe: Global vectors for word representation]

Andrew Ng

Model

$$\text{minimize} \sum_{i=1}^{10,000} \sum_{j=1}^{10,000} \underbrace{f(X_{ij})}_{\text{weighting term}} \left(\theta_i^T e_j + b_i + b_j' - \log X_{ij}\right)^2$$

$t \quad c$

"$\theta_t^T e_c$"

$\log X_{ij} \to 0$?

$f(X_{ij}) = 0$ if $X_{ij} = 0$. "$0 \log 0$" $= 0$

this, is, of, a, ...

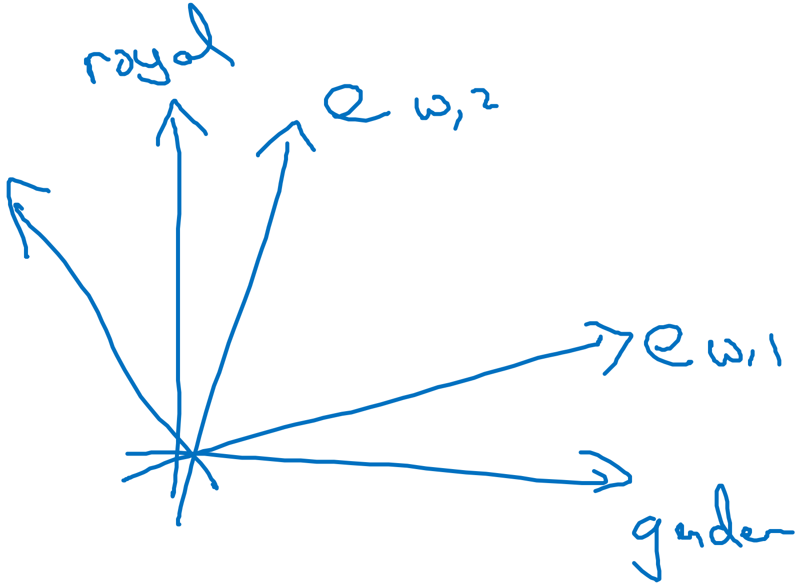
durian

θ_i, e_j are symmetric

$$e_w^{(final)} = \frac{e_w + \theta_w}{2}$$

A note on the featurization view of word embeddings

	Man (5391)	Woman (9853)	King (4914)	Queen (7157)
Gender	−1	1	-0.95	0.97
Royal	0.01	0.02	0.93	0.95
Age	0.03	0.02	0.70	0.69
Food	0.09	0.01	0.02	0.01



$$\text{minimize} \sum_{i=1}^{10,000} \sum_{j=1}^{10,000} f(X_{ij}) \left(\theta_i^T e_j + b_i - b_j' - \log X_{ij} \right)^2$$

$$(A\theta_i)^T (A^{-T} e_j) = \theta_i^T \cancel{A^T} \cancel{A^{-T}} e_j$$

Andrew Ng

deeplearning.ai

NLP and Word Embeddings

Sentiment classification

Sentiment classification problem

$x \longrightarrow y$

The dessert is excellent. ★★★★☆

Service was quite slow. ★★☆☆☆

Good for a quick meal, but nothing special. ★★★☆☆

Completely lacking in good taste, good service, and good ambience. ★☆☆☆☆

10,000 → 100,000 words

Simple sentiment classification model

The dessert is excellent ★★★★☆

8928 2468 4694 3180

The $o_{8928} \longrightarrow E \longrightarrow e_{8928}$

desert $o_{2468} \longrightarrow E \longrightarrow e_{2468}$

is $o_{4694} \longrightarrow E \longrightarrow e_{4694}$

excellent $o_{3180} \longrightarrow E \longrightarrow e_{3180}$

$\underbrace{e_{3180}}_{300D}$

Avg

300D

$\rightarrow O \rightarrow \hat{y}$

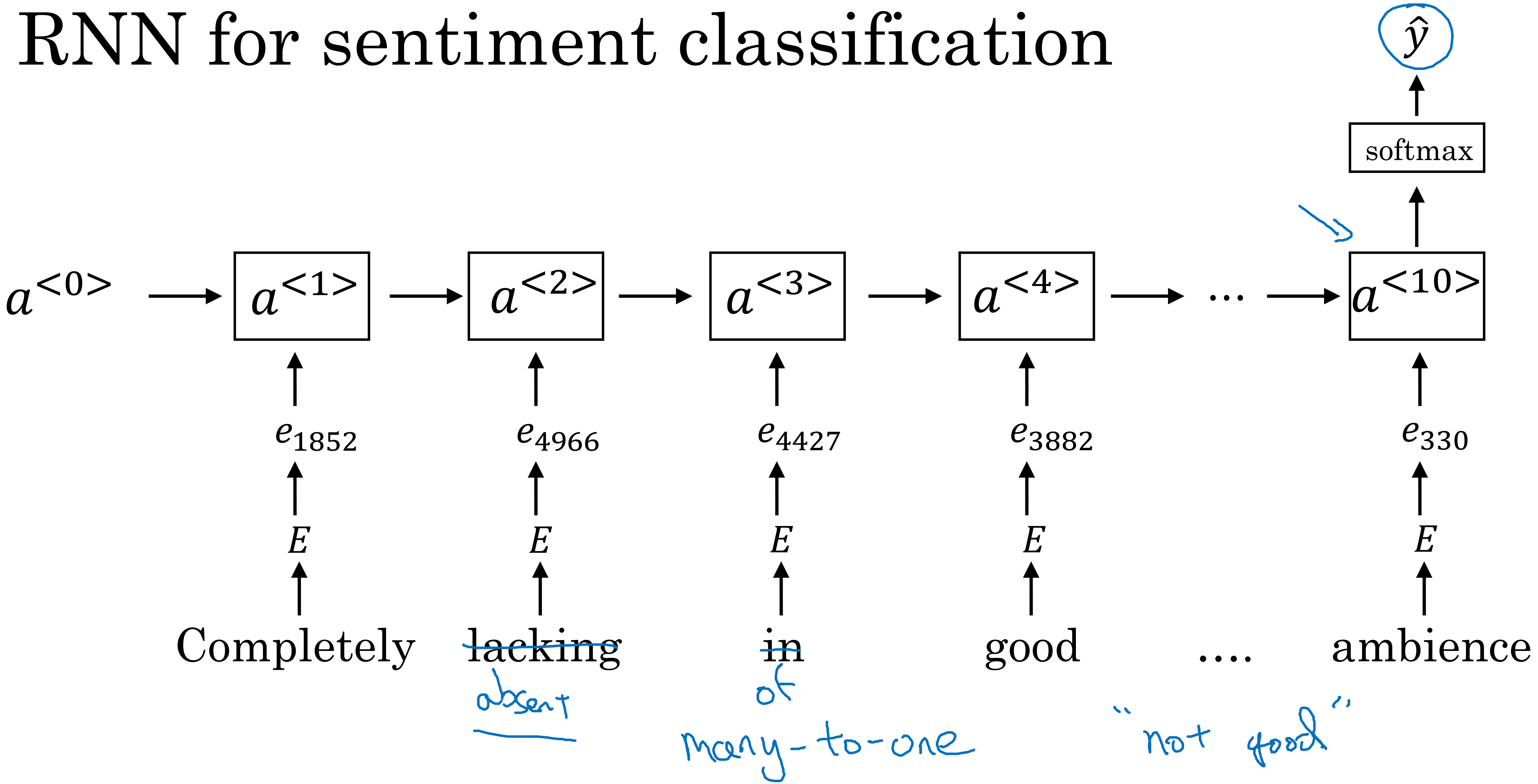
softmax

1-5

100B words

"Completely lacking in good taste, good service, and good ambience."

RNN for sentiment classification



deeplearning.ai

NLP and Word Embeddings

Debiasing word embeddings

The problem of bias in word embeddings

Man:Woman as King:Queen

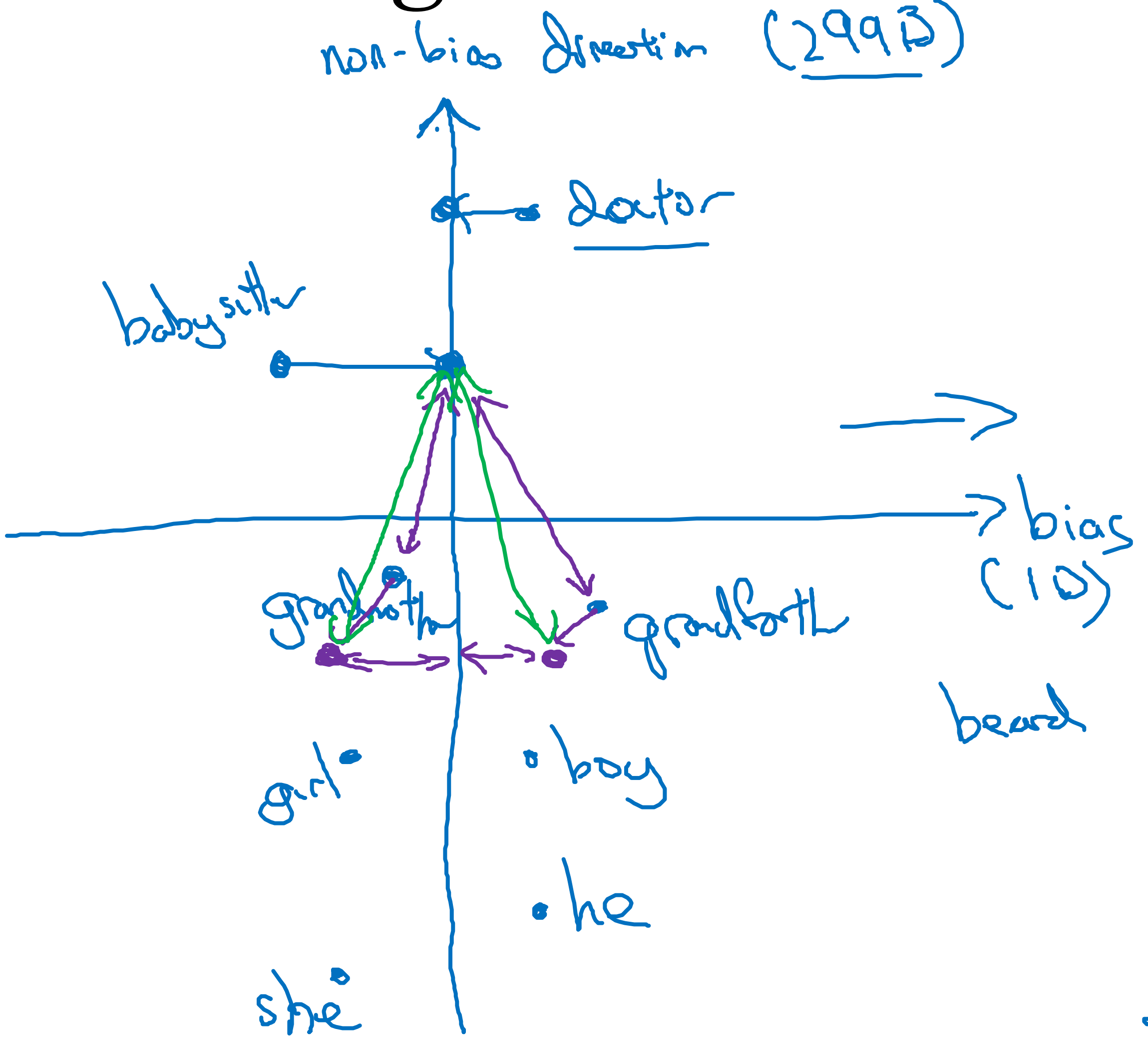
Man:Computer_Programmer as Woman: Homemaker ✗

Father:Doctor as Mother: Nurse ✗

Word embeddings can reflect gender, ethnicity, age, sexual orientation, and other biases of the text used to train the model.

[Bolukbasi et. al., 2016. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings]

Addressing bias in word embeddings



1. Identify bias direction.

$e_{he} - e_{she}$

$e_{male} - e_{female}$

$\vdots$

average

2. Neutralize: For every word that is not definitional, project to get rid of bias.

3. Equalize pairs.

grandmother – grandfather

girl – boy

[Bolukbasi et. al., 2016. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings]

Andrew Ng

deeplearning.ai

Sequence to sequence models

Basic models

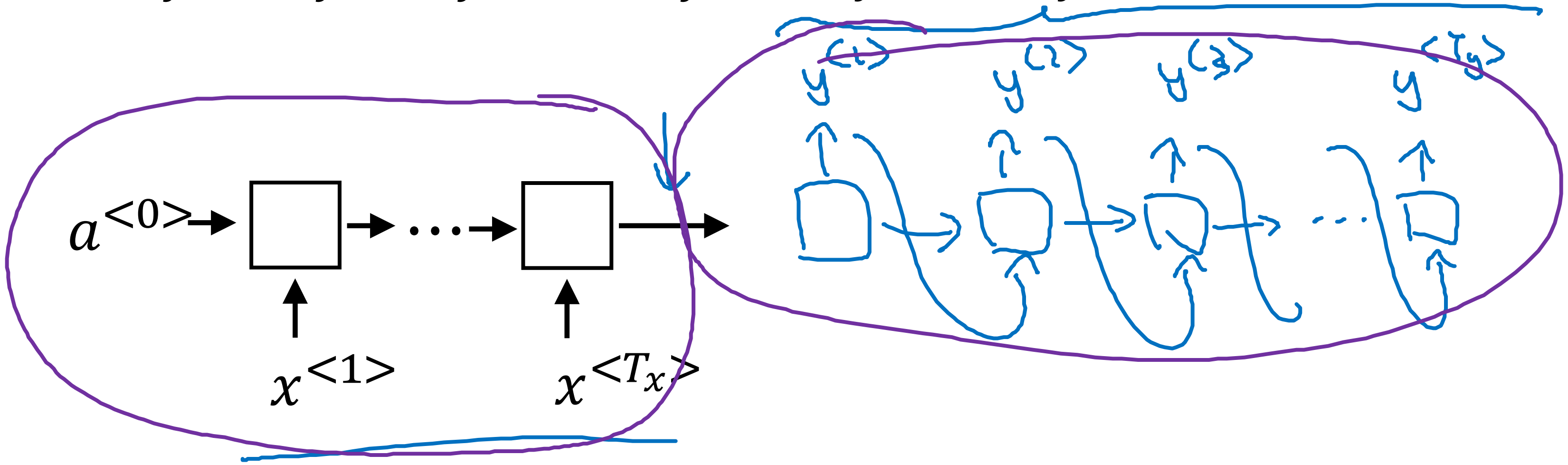
Sequence to sequence model

$x^{<1>}$ $x^{<2>}$ $x^{<3>}$ $x^{<4>}$ $x^{<5>}$

Jane visite l'Afrique en septembre

$\longrightarrow$ Jane is visiting Africa in September.

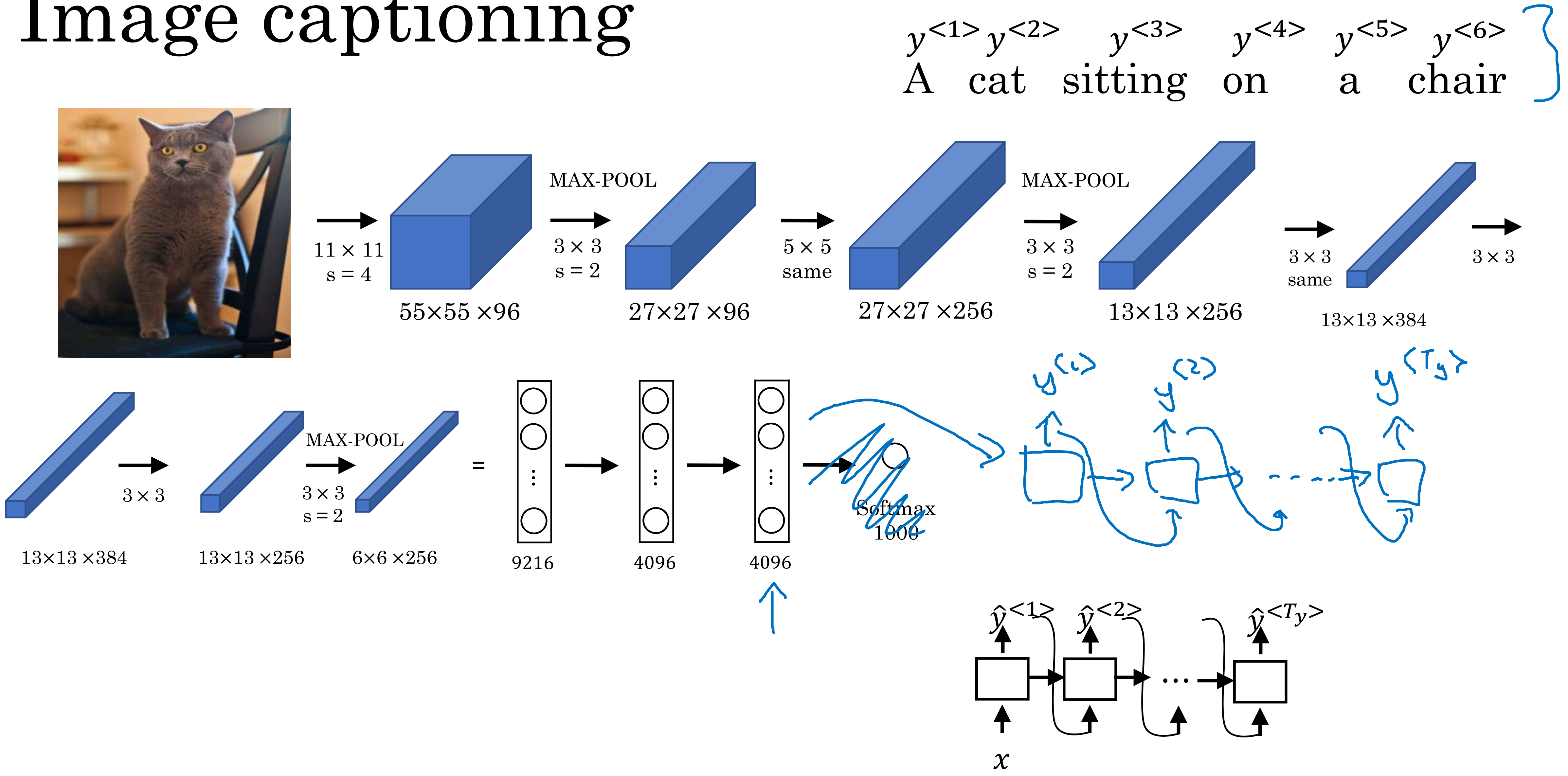
$y^{<1>}$ $y^{<2>}$ $y^{<3>}$ $y^{<4>}$ $y^{<5>}$ $y^{<6>}$



[Sutskever et al., 2014. Sequence to sequence learning with neural networks]

[Cho et al., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation]

Image captioning



[Mao et. al., 2014. Deep captioning with multimodal recurrent neural networks]
[Vinyals et. al., 2014. Show and tell: Neural image caption generator]
[Karpathy and Li, 2015. Deep visual-semantic alignments for generating image descriptions]

deeplearning.ai

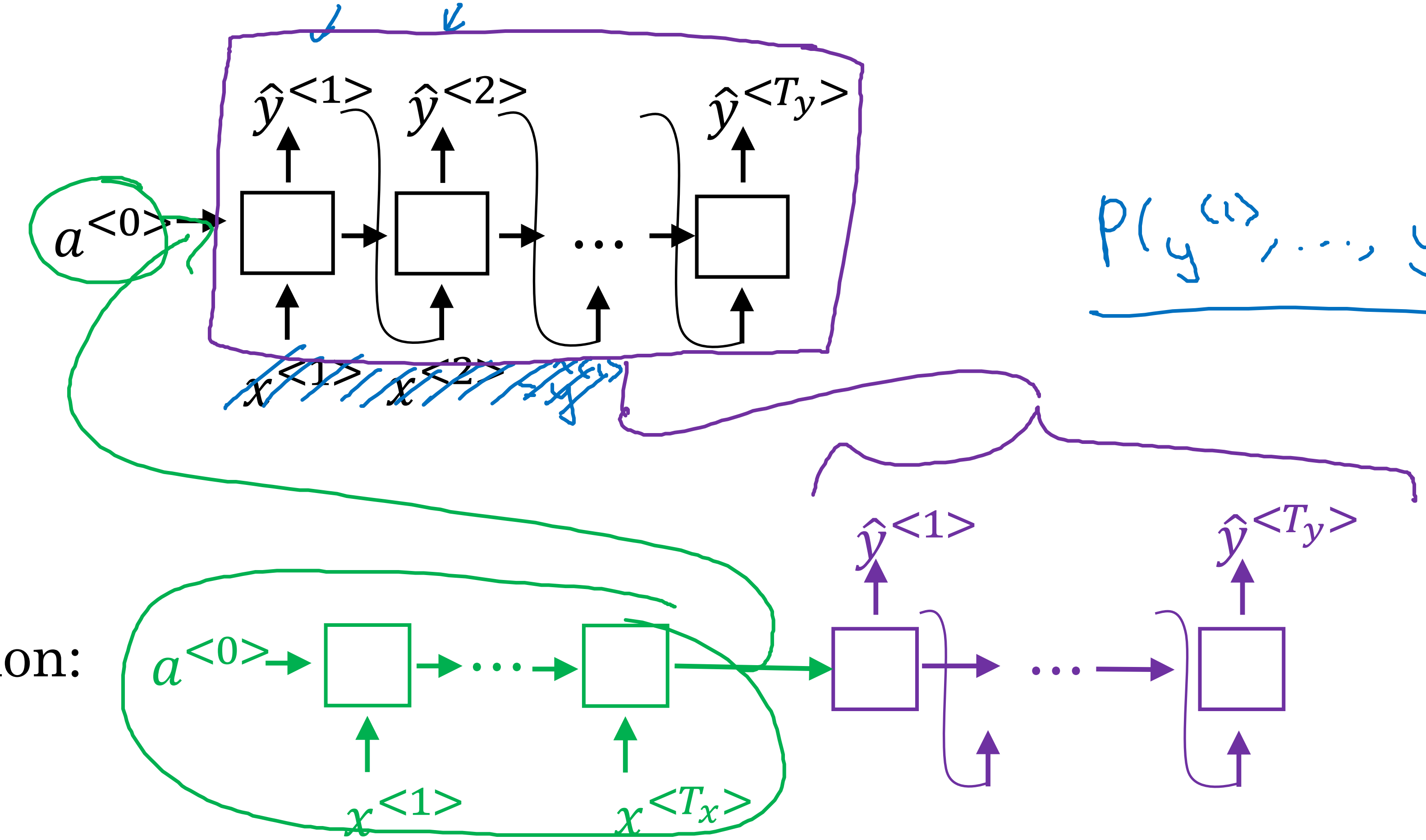
Sequence to sequence models

Picking the most likely sentence

Machine translation as building a conditional language model

Language model:

Machine translation:



$P(y^{<1>}, \ldots, y^{<T_y>})$

"Conditional language model"

$P(y^{<1>}, \ldots, y^{<T_y>} \mid x^{<1>}, \ldots, x^{<T_x>})$

Finding the most likely translation

Jane visite l'Afrique en septembre.

$$P(\overbrace{y^{<1>}, \dots, y^{<T_y>}}^{\text{English}} \mid \overset{\text{French}}{x})$$

→ Jane is visiting Africa in September.

→ Jane is going to be visiting Africa in September.

→ In September, Jane will visit Africa.

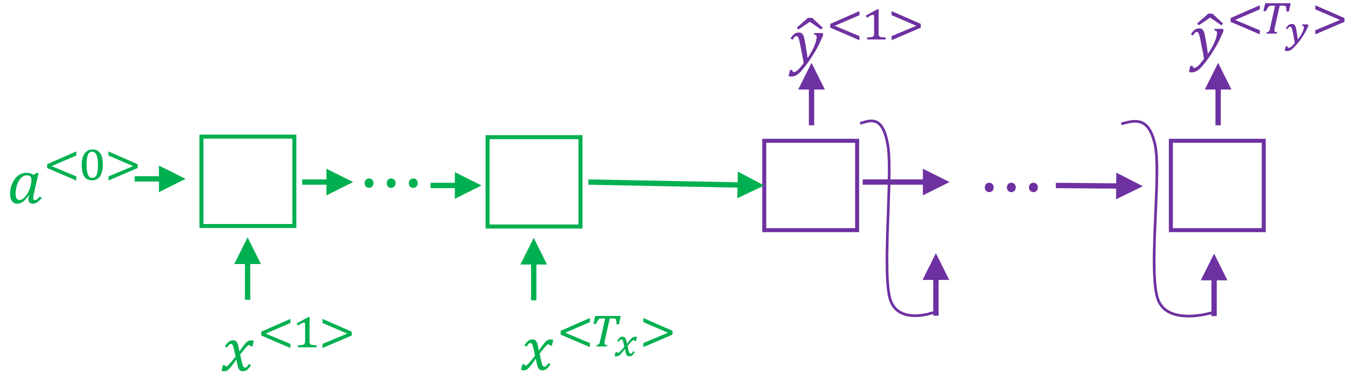
→ Her African friend welcomed Jane in September.

$$\underset{y^{<1>}, \dots, y^{<T_y>}}{\arg\max} \; P(y^{<1>}, \dots, y^{<T_y>} \mid x)$$

Andrew Ng

Why not a greedy search?

$P(\hat{y}^{<1>} \mid x)$



$$\arg\max_{y} P(\underbrace{\hat{y}^{<1>}, \hat{y}^{<2>},}\ \ldots, \hat{y}^{<T_y>} \mid x)$$

10,000

10

$10{,}000^{10}$

$P(y \mid x)$

→ Jane is visiting Africa in September.

→ Jane is going to be visiting Africa in September.

P(Jane is going | x) > P(Jane is visiting | x)

deeplearning.ai

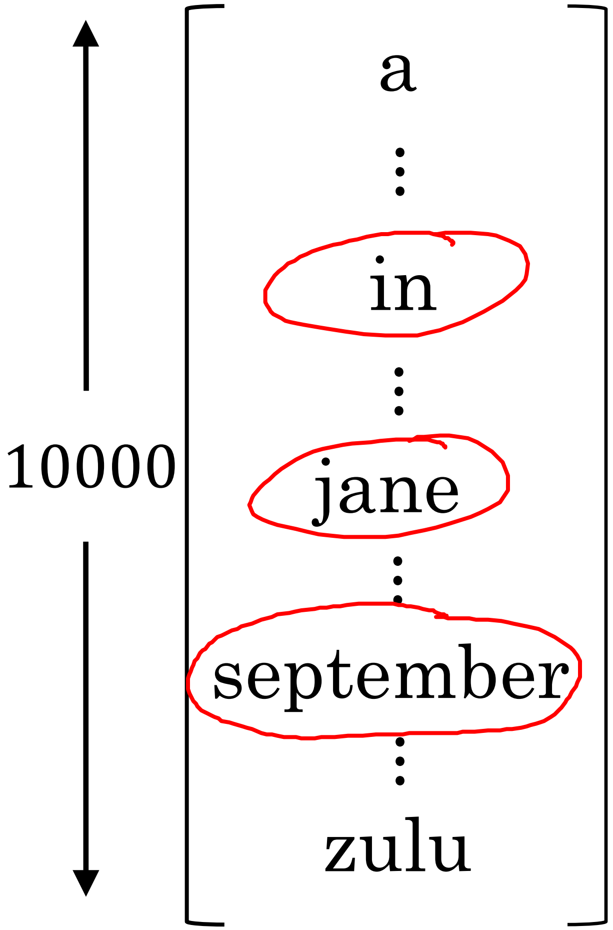
Sequence to sequence models

Beam search

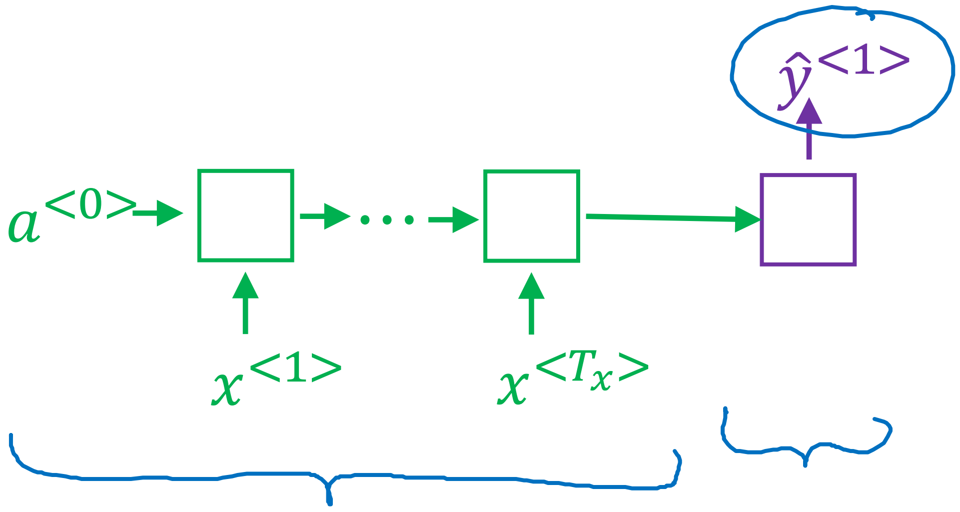
Beam search algorithm

$\underline{B = 3}$ (beam width)

Step 1

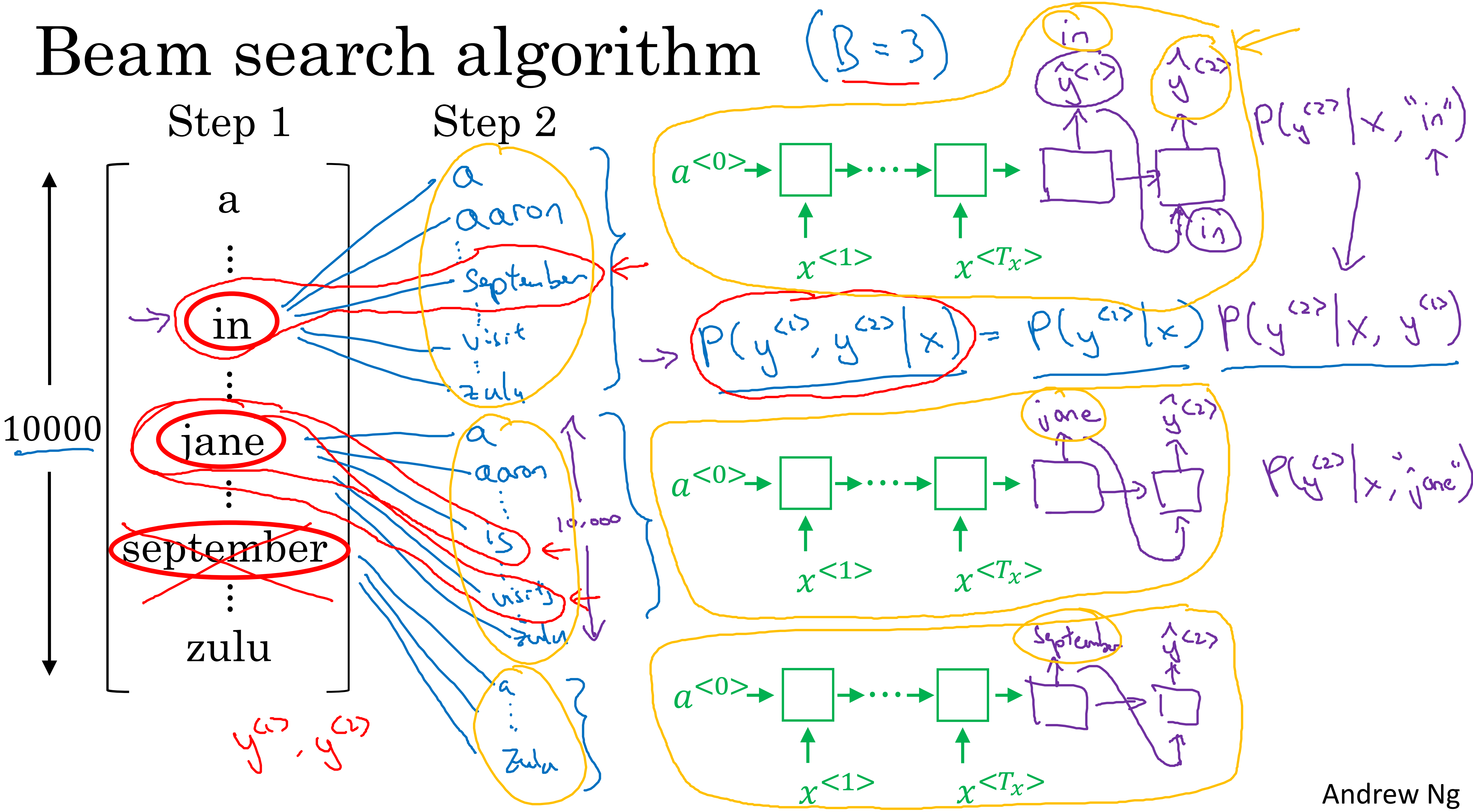


$\rightarrow \underline{P(y^{<1>} \mid x)}$



Andrew Ng

Beam search algorithm

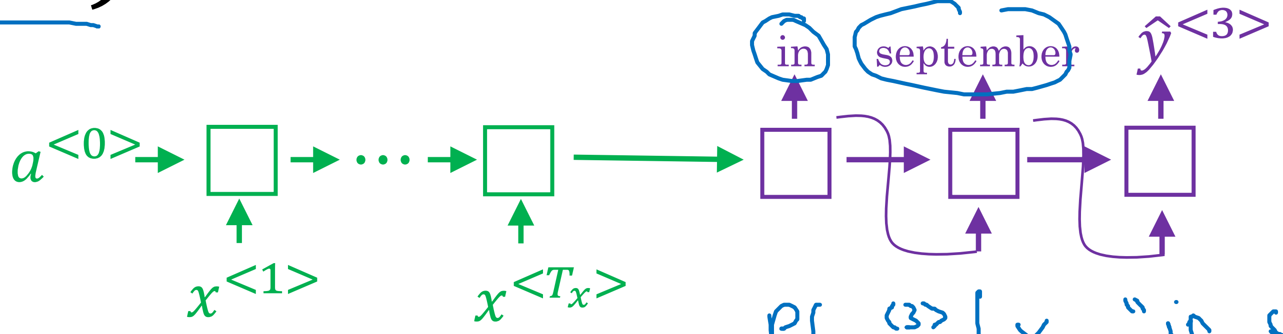


Andrew Ng

Beam search ($B = 3$)

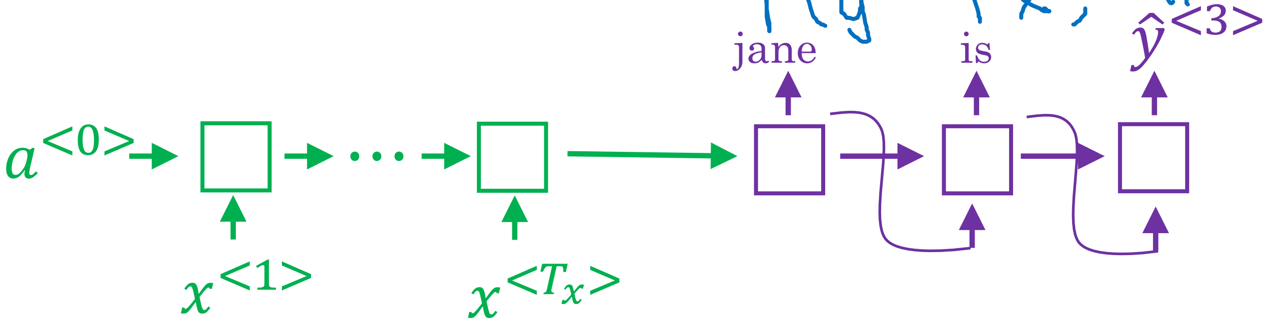
$B = 1$ ⇝ greedy search

in september — a, aaron, jane, zulu

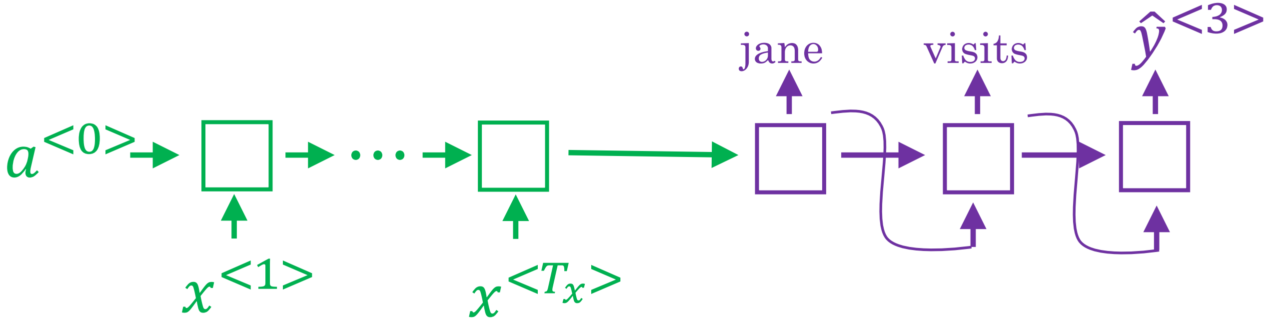


$P(y^{<3>} \mid x, \text{"in september"})$

jane is — a, visits, zulu



jane visits — a, africa, zulu



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

jane visits africa in september. <EOS>

Andrew Ng

deeplearning.ai

Sequence to sequence models

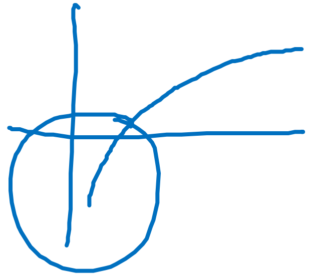
Refinements to beam search

Length normalization

$$P(y^{<1>} \dots y^{<T_y>} | x) = P(y^{<1>}|x)\, P(y^{<2>}|x, y^{<1>}) \dots P(y^{<T_y>}|x, y^{<1>}, \dots, y^{<T_y-1>})$$

$$\arg\max_{y} \prod_{t=1}^{T_y} P(y^{<t>} | x, y^{<1>}, \dots, y^{<t-1>})$$

log



$\log P(y|x) \leftarrow$

$P(y|x) \leftarrow$

$$\arg\max_{y} \sum_{t=1}^{T_y} \log P(y^{<t>} | x, y^{<1>}, \dots, y^{<t-1>}) \leftarrow$$

$T_y = 1, 2, 3, \dots, 30.$

$$\rightarrow \boxed{\frac{1}{T_y^{\alpha}} \sum_{t=1}^{T_y} \log P(y^{<t>} | x, y^{<1>}, \dots, y^{<t-1>})}$$

$\alpha = 0.7$

$\alpha = 1$

$\alpha = 0$

Beam search discussion

large B: better result, slower
small B: worse result, faster

Beam width B?

$1 \rightarrow 3 \rightarrow 10$, 100, 1000, $\rightarrow$ 3000

Unlike exact search algorithms like BFS (Breadth First Search) or DFS (Depth First Search), Beam Search runs faster but is not guaranteed to find exact maximum for $\arg\max_{y} P(y|x)$.

deeplearning.ai

Sequence to sequence models

Error analysis on beam search

Example

→ RNN

→ Beam Search

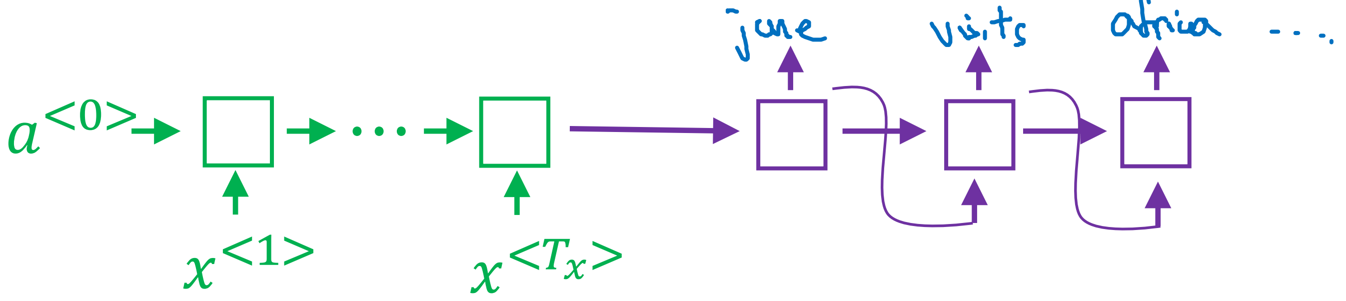
$\boxed{BT}$

Jane visite l'Afrique en septembre.

Human: Jane visits Africa in September. (y^*)

Algorithm: Jane visited Africa last September. $(\hat{y})$ ←

RNN computes $P(y^*|x) \; \overset{>}{\leq} \; P(\hat{y}|x)$



Error analysis on beam search

$P(y^*|x)$

Human: Jane visits Africa in September. (y^*)

$P(\hat{y}|x)$

Algorithm: Jane visited Africa last September. ($\hat{y}$)

Case 1: $P(y^*|x) > P(\hat{y}|x)$ $\quad \arg\max_y P(y|x)$

Beam search chose $\hat{y}$. But y^* attains higher $\boxed{P(y|x)}$.

Conclusion: Beam search is at fault.

Case 2: $P(y^*|x) \leq P(\hat{y}|x)$

y^* is a better translation than $\hat{y}$. But RNN predicted $\boxed{P(y^*|x)} < P(\hat{y}|x)$.

Conclusion: RNN model is at fault.

Error analysis process

Human	Algorithm	$P(y^*\|x)$	$P(\hat{y}\|x)$	At fault?
Jane visits Africa in September.	Jane visited Africa last September.	2×10^{-10}	1×10^{-10}	B
				R
				B
				R
				R
				⋮

Figures out what faction of errors are "due to" beam search vs. RNN model

deeplearning.ai

Sequence to sequence models

Bleu score (optional)

Evaluating machine translation

French: Le chat est sur le tapis.

Reference 1: The cat is on the mat.

Reference 2: There is a cat on the mat.

MT output: the the the the the the the.

Precision: Modified precision:

Bleu

bilingual evaluation understudy

[Papineni et. al., 2002. Bleu: A method for automatic evaluation of machine translation]

Bleu score on bigrams

Example: Reference 1: The cat is on the mat.

Reference 2: There is a cat on the mat.

MT output: The cat the cat on the mat.

	Count	Count$_{clip}$
the cat	2	1
cat the	1	0
cat on	1	1
on the	1	1
the mat	1	1

$$\frac{4}{6}$$

[Papineni et. al., 2002. Bleu: A method for automatic evaluation of machine translation]

Andrew Ng

Bleu score on unigrams

Example: Reference 1: The cat is on the mat.

Reference 2: There is a cat on the mat.

→ MT output: The cat the cat on the mat. ($\hat{y}$)

$p_1, p_2, = 1.0$

n-gram

$$p_1 = \frac{\sum_{unigram \in \hat{y}} count_{clip}(unigram)}{\sum_{unigram \in \hat{y}} count(unigram)}$$

$$p_n = \frac{\sum_{ngram \in \hat{y}} count_{clip}(ngram)}{\sum_{ngram \in \hat{y}} count(ngram)}$$

$$p_n = \frac{\sum_{n\text{-}grams \in \hat{y}} count_{clip}(n\text{-}gram)}{\sum_{n\text{-}grams \in \hat{y}} count(n\text{-}gram)}$$

unigram

[Papineni et. al., 2002. Bleu: A method for automatic evaluation of machine translation]

Andrew Ng

Bleu details

p_n = Bleu score on n-grams only

p_1, p_2, p_3, p_4

Combined Bleu score: $BP \exp\left(\frac{1}{4}\sum_{n=1}^{4} p_n\right)$

BP = brevity penalty

$$BP = \begin{cases} 1 & \text{if } \underline{\text{MT\_output\_length}} > \underline{\text{reference\_output\_length}} \\ \exp(1 - \text{MT\_output\_length}/\text{reference\_output\_length}) & \text{otherwise} \end{cases}$$

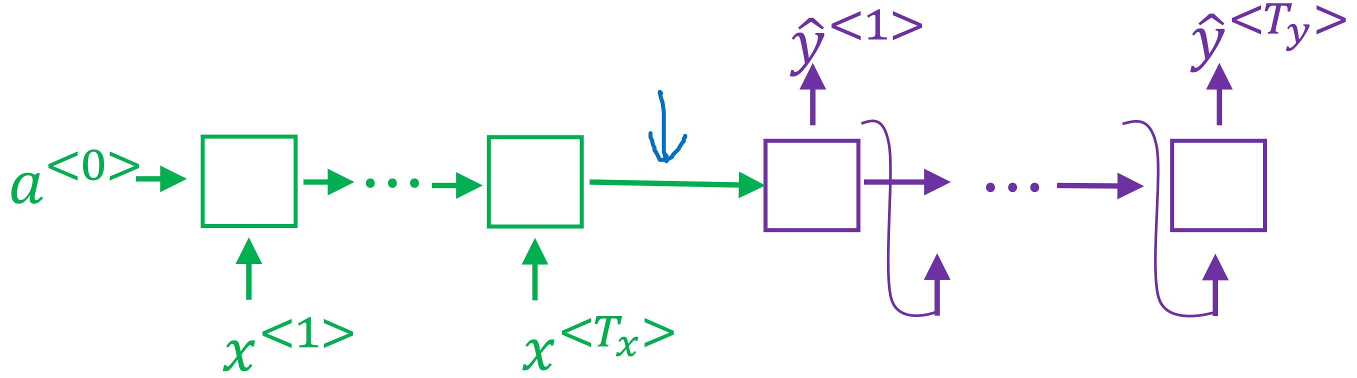
[Papineni et. al., 2002. Bleu: A method for automatic evaluation of machine translation]

Sequence to sequence models

Attention model intuition

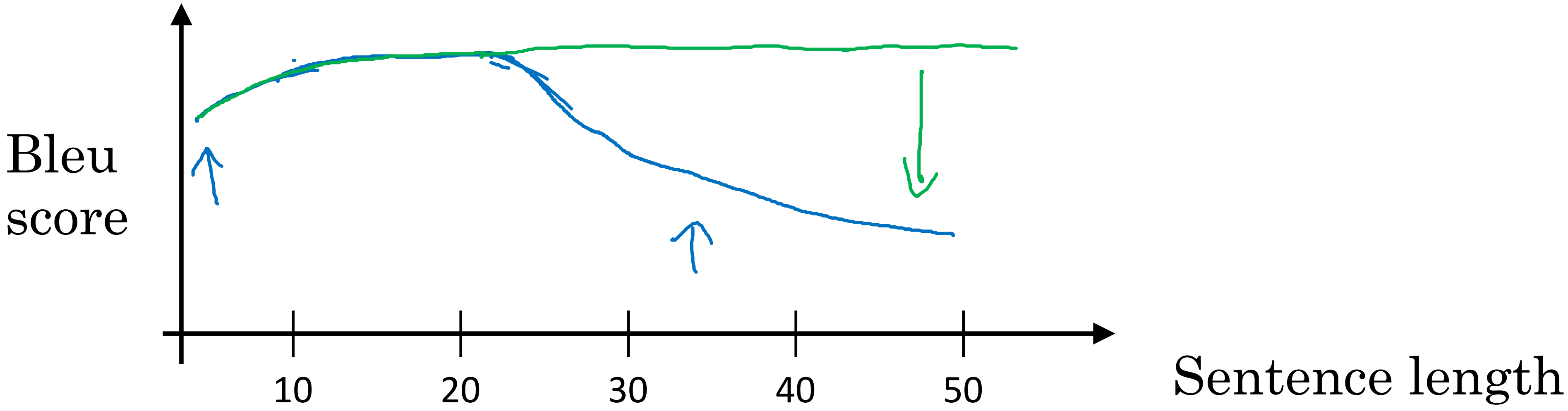
deeplearning.ai

The problem of long sequences

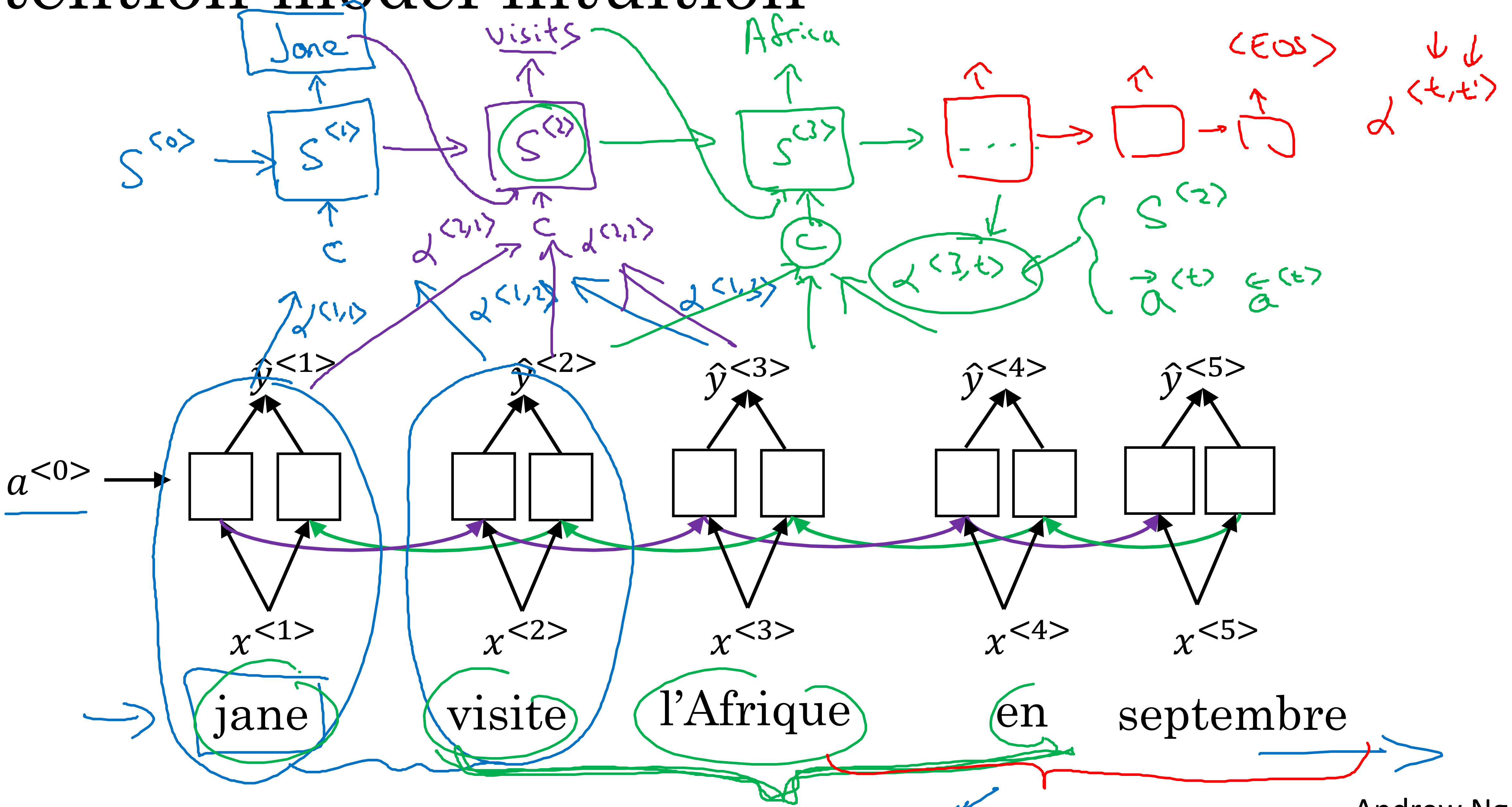


Jane s'est rendue en Afrique en septembre dernier, a apprécié la culture et a rencontré beaucoup de gens merveilleux; elle est revenue en parlant comment son voyage était merveilleux, et elle me tente d'y aller aussi.

Jane went to Africa last September, and enjoyed the culture and met many wonderful people; she came back raving about how wonderful her trip was, and is tempting me to go too.



Attention model intuition



[Bahdanau et. al., 2014. Neural machine translation by jointly learning to align and translate]

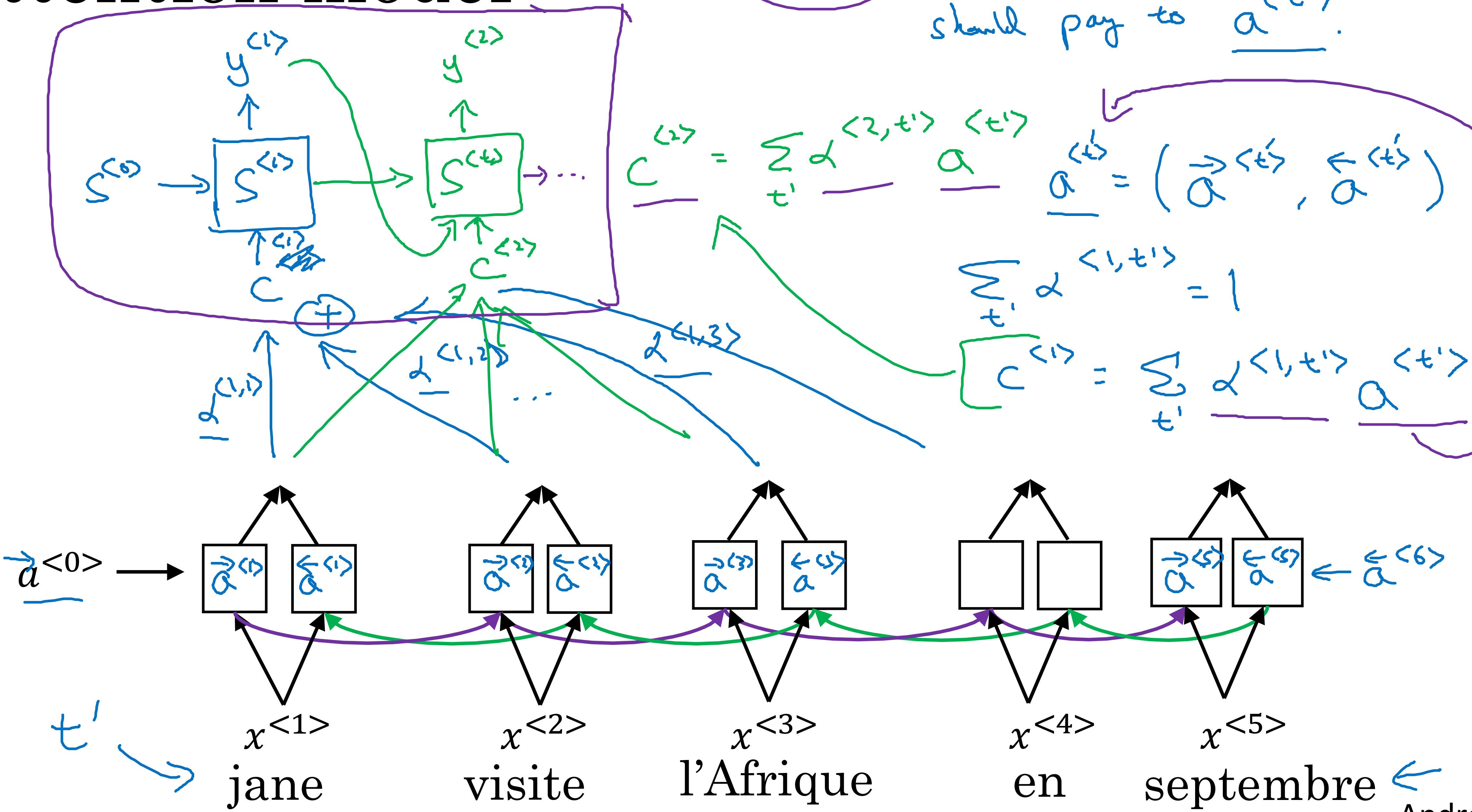
deeplearning.ai

Sequence to sequence models

Attention model

Attention model

$\alpha^{<t,t'>}$ = amount of "attention" $y^{<t>}$ should pay to $a^{<t'>}$.



$c^{<2>} = \sum_{t'} \alpha^{<2,t'>} a^{<t'>}$

$a^{<t'>} = (\overrightarrow{a}^{<t'>}, \overleftarrow{a}^{<t'>})$

$\sum_{t'} \alpha^{<1,t'>} = 1$

$\boxed{c^{<1>}} = \sum_{t'} \alpha^{<1,t'>} a^{<t'>}$

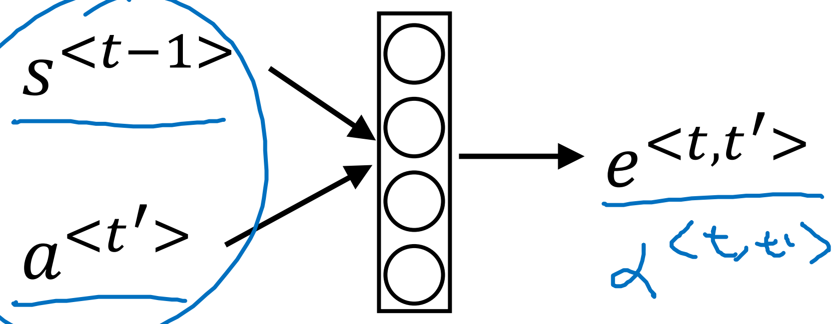
[Bahdanau et. al., 2014. Neural machine translation by jointly learning to align and translate]

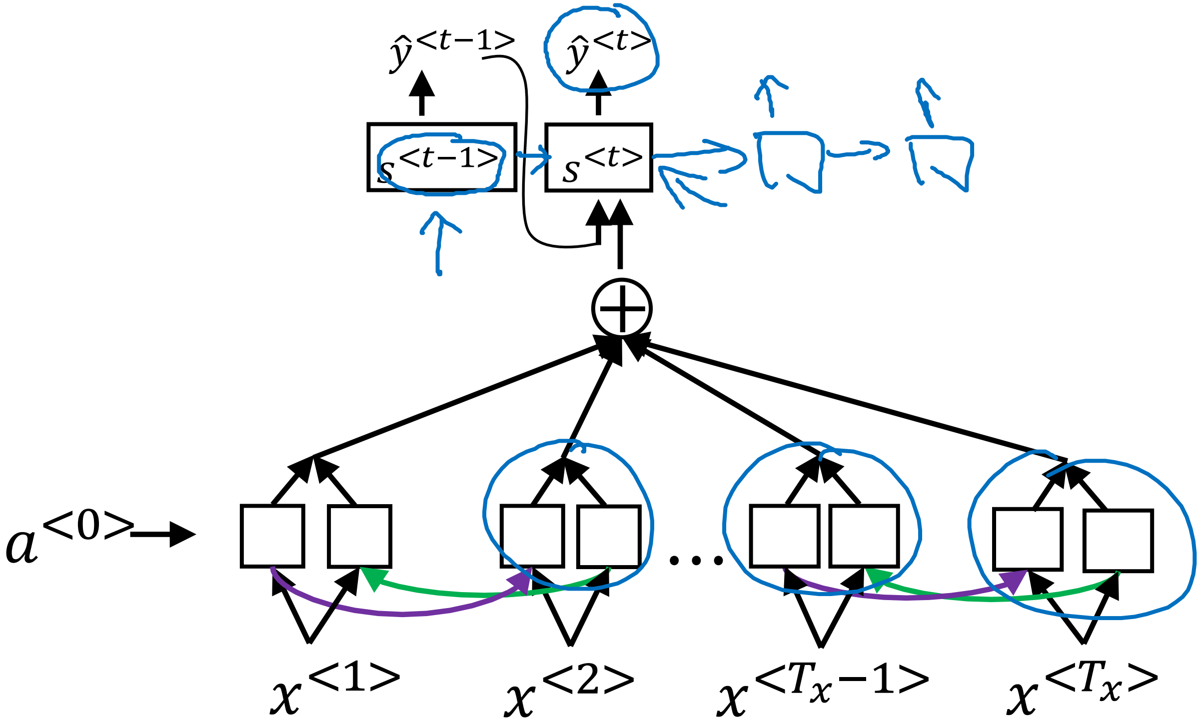
Computing attention $\alpha^{<t,t'>}$

T_x T_y

$\alpha^{<t,t'>}$ = amount of attention $y^{<t>}$ should pay to $a^{<t'>}$

$$\alpha^{<t,t'>} = \frac{\exp(e^{<t,t'>})}{\sum_{t'=1}^{T_x} \exp(e^{<t,t'>})}$$





[Bahdanau et. al., 2014. Neural machine translation by jointly learning to align and translate]

[Xu et. al., 2015. Show, attend and tell: Neural image caption generation with visual attention]

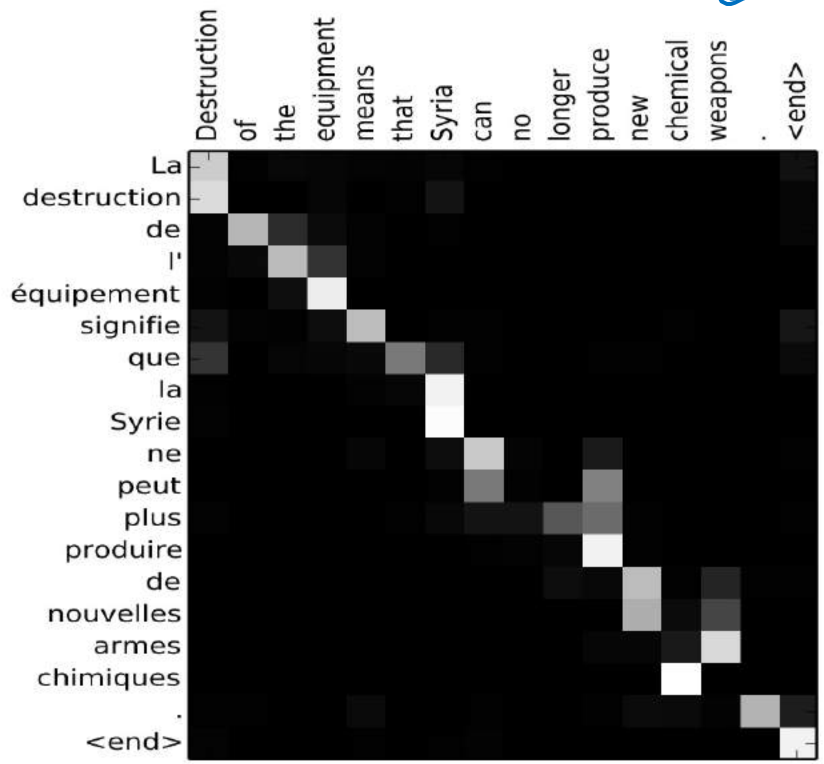
Andrew Ng

Attention examples

July 20th 1969 $\longrightarrow$ 1969 – 07 – 20

23 April, 1564 $\longrightarrow$ 1564 – 04 – 23

Visualization of $\alpha^{<t,t'>}$:



Andrew Ng

deeplearning.ai

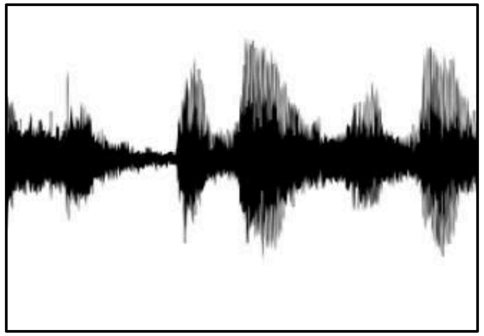
Audio data

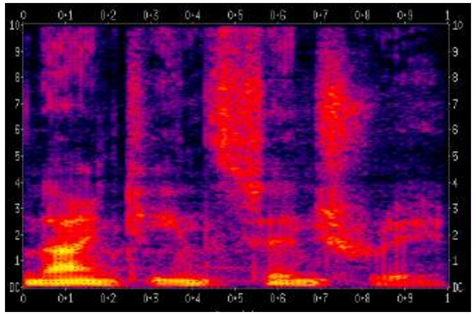
Speech recognition

Speech recognition problem

x → y

audio clip → transcript





→ "the quick brown fox"

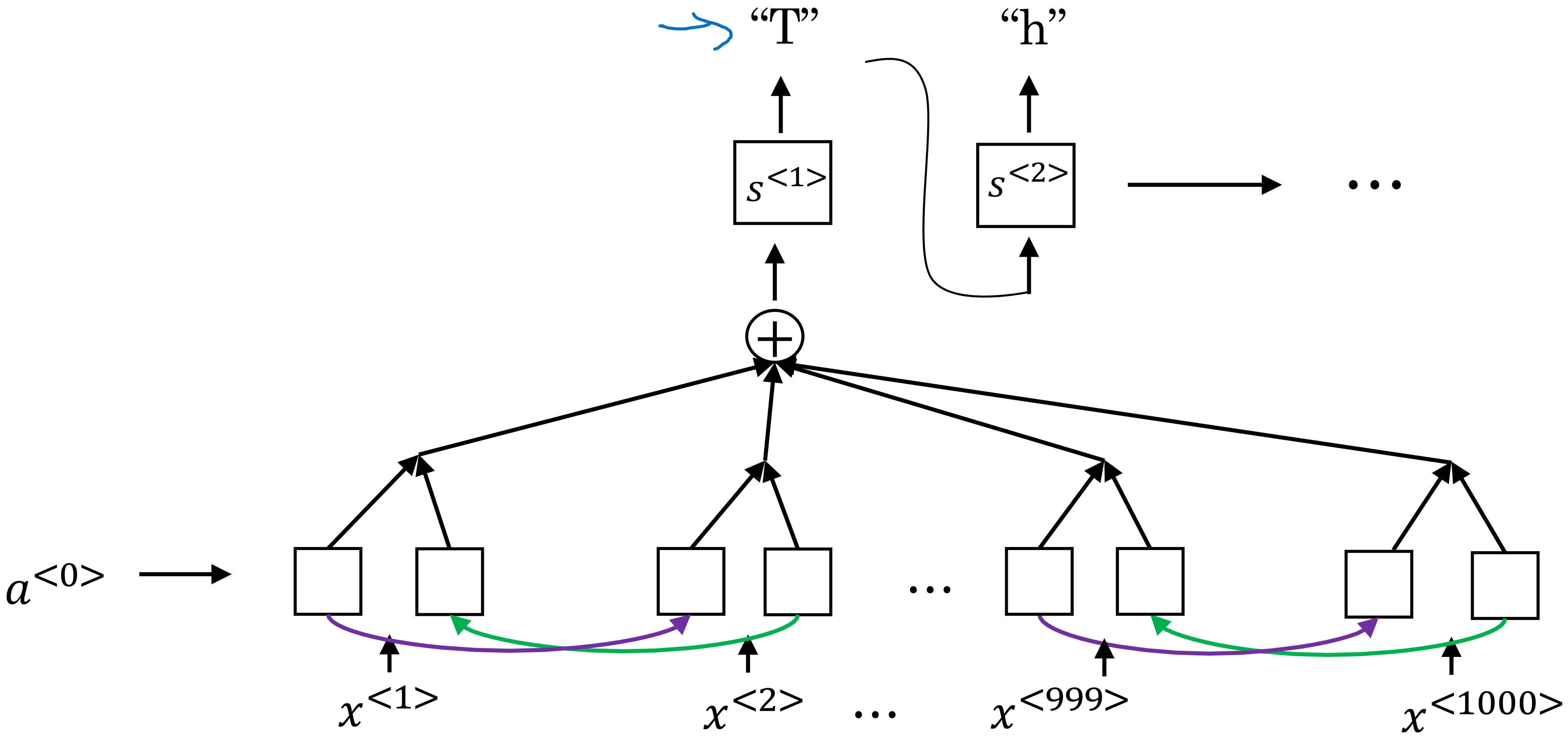
→ phonemes: de kwik braun

300h

3000h

100,000h

Attention model for speech recognition

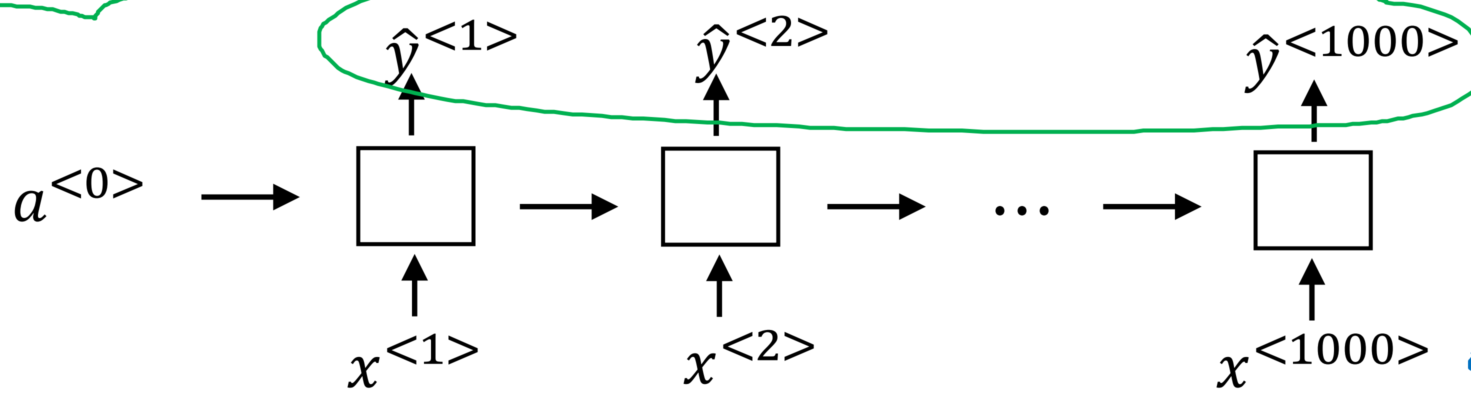


Andrew Ng

CTC cost for speech recognition

(Connectionist temporal classification)

"the quick brown fox" 19 characters



ttt_h_eee___⊔___qqq__ the⊔q

"space" "blank"

Basic rule: collapse repeated characters not separated by "blank"

[Graves et al., 2006. Connectionist Temporal Classification: Labeling unsegmented sequence data with recurrent neural networks]

deeplearning.ai

Audio data

Trigger word detection

What is trigger word detection?

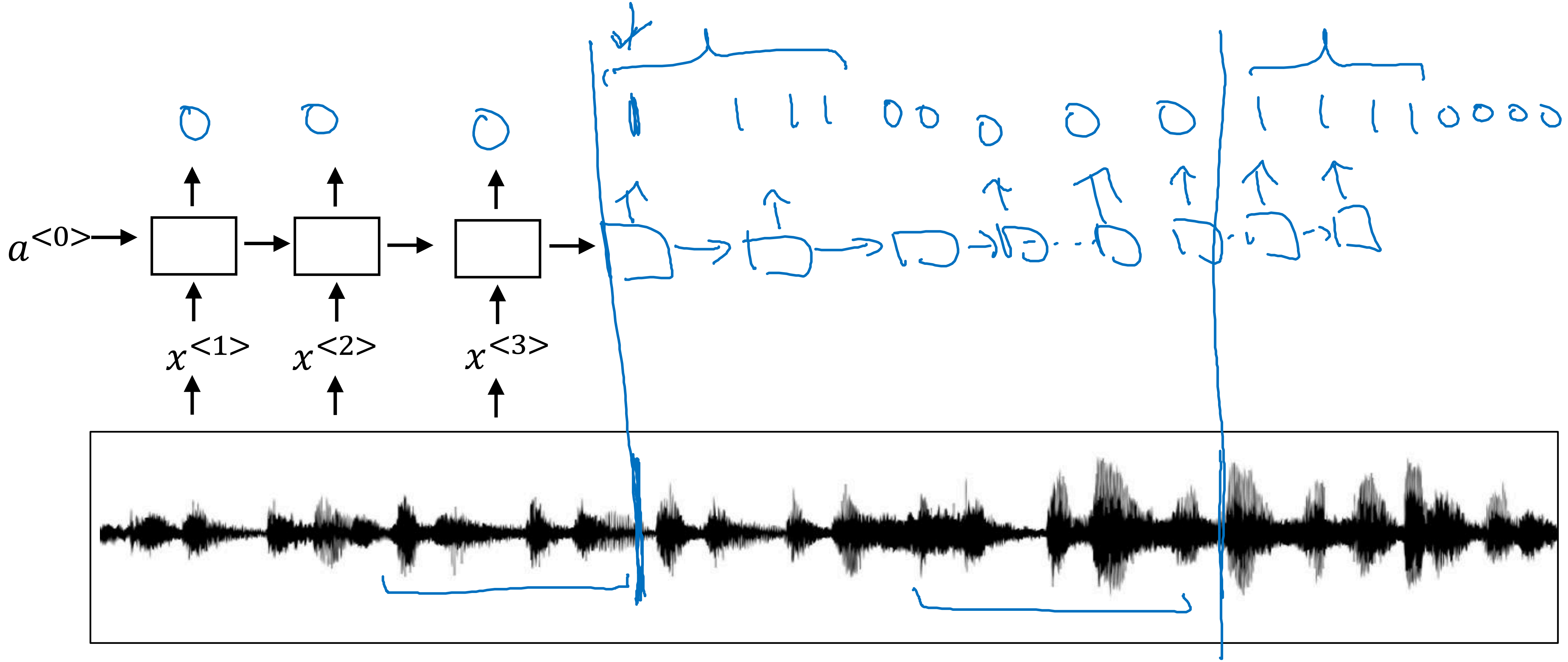
Amazon Echo
(Alexa)

Baidu DuerOS
(xiaodunihao)

Apple Siri
(Hey Siri)

Google Home
(Okay Google)

Trigger word detection algorithm



deeplearning.ai

Conclusion

Summary and thank you

Specialization outline

1. Neural Networks and Deep Learning
2. Improving Deep Neural Networks: Hyperparameter tuning, Regularization and Optimization
3. Structuring Machine Learning Projects
4. Convolutional Neural Networks
5. Sequence Models

Deep learning is a super power

Please buy this from shutterstock and replace in final video.

Thank you.

-Andrew Ng

Copyright Notice

Sequence to sequence models

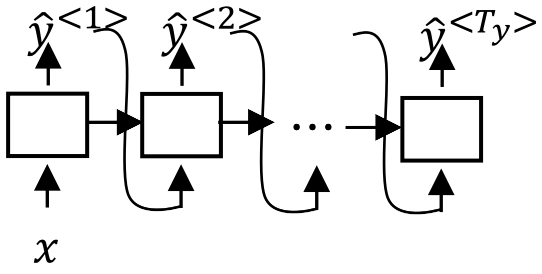
deeplearning.ai

Transformers Intuition

Transformers Motivation

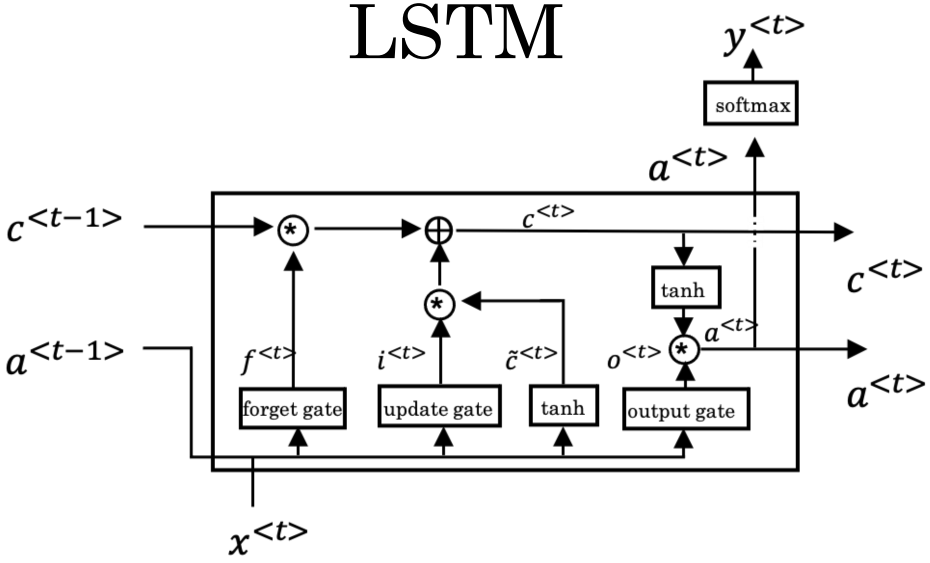
Increased complexity,
sequential

RNN



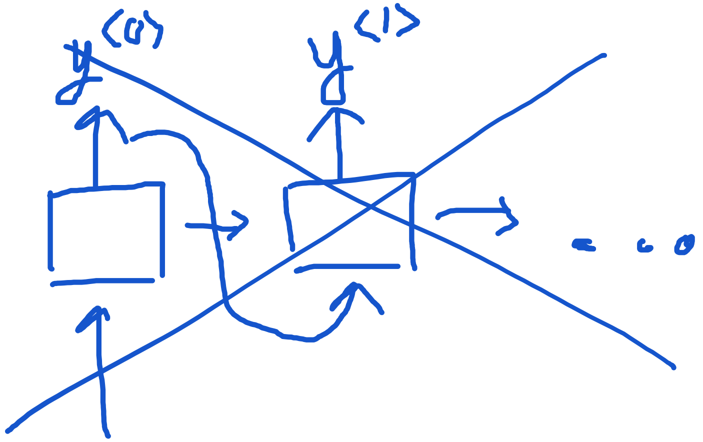
GRU

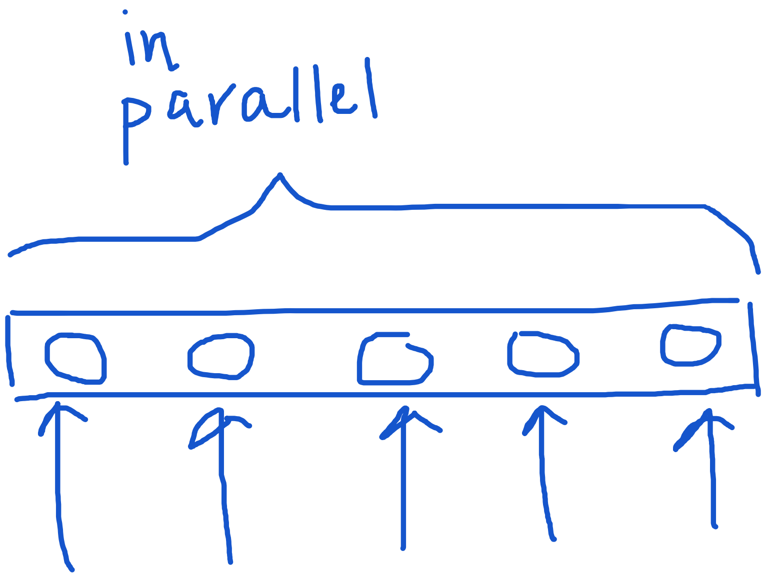
LSTM



Transformers Intuition

- Attention + CNN
  - Self-Attention
  - Multi-Head Attention





[Vaswani et al. 2017, Attention Is All You Need]

Andrew Ng

Sequence to sequence models

Self-Attention

deeplearning.ai

Self-Attention Intuition

$A(q, K, V)$= attention-based vector representation of a word

calculate for each word

RNN Attention

$$\alpha^{\cancel{<t,t'>}} = \frac{\exp(e^{<t,t'>})}{\sum_{t'=1}^{T_x} \exp(e^{<t,t'>})}$$

Transformers Attention

$$A(q, K, V) = \sum_i \frac{\exp(q \cdot k^{<i>})}{\sum_j \exp(q \cdot k^{<j>})} v^{<i>}$$

$x^{<1>}$	$x^{<2>}$	$x^{<3>}$	$x^{<4>}$	$x^{<5>}$
Jane	visite	l'Afrique	en	septembre

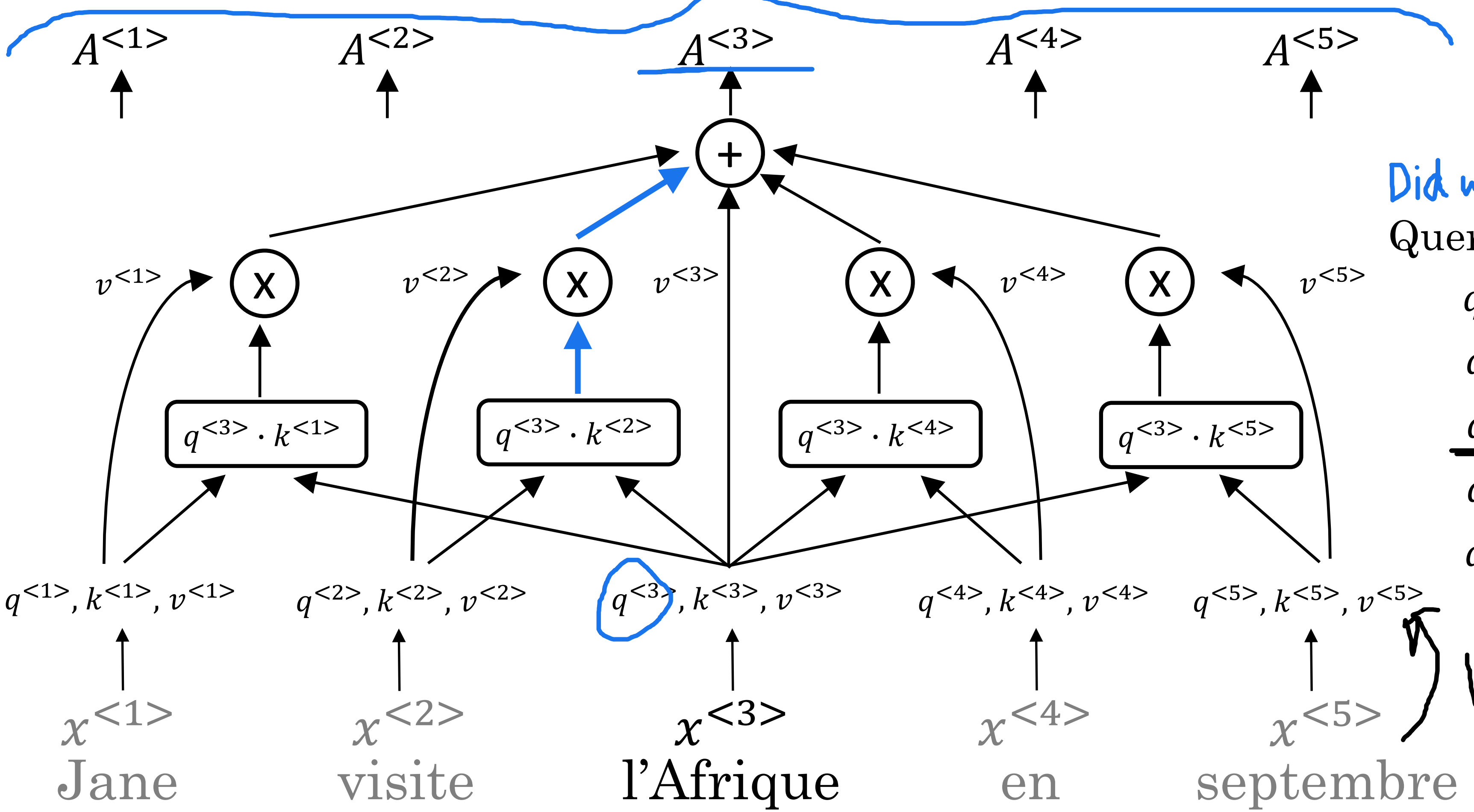
[Vaswani et al. 2017, Attention Is All You Need]

Self-Attention

softmax

$$A(q, K, V) = \sum_i \frac{\exp(e^{<q \cdot k^{<i>}>})}{\sum_j \exp(e^{<q \cdot k^{<j>}>})} v^{<i>}$$

$$\boxed{Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V}$$



Did what?

Query (Q)	Key (K)	Value (V)
$q^{<1>}$	$k^{<1>}$	$v^{<1>}$
$q^{<2>}$	$k^{<2>}$	$v^{<2>}$
$q^{<3>}$	$k^{<3>}$	$v^{<3>}$
$q^{<4>}$	$k^{<4>}$	$v^{<4>}$
$q^{<5>}$	$k^{<5>}$	$v^{<5>}$

person

action

Jane

visit

What's happening there?

W^Q, W^K, W^V

[Vaswani et al. 2017, Attention Is All You Need]

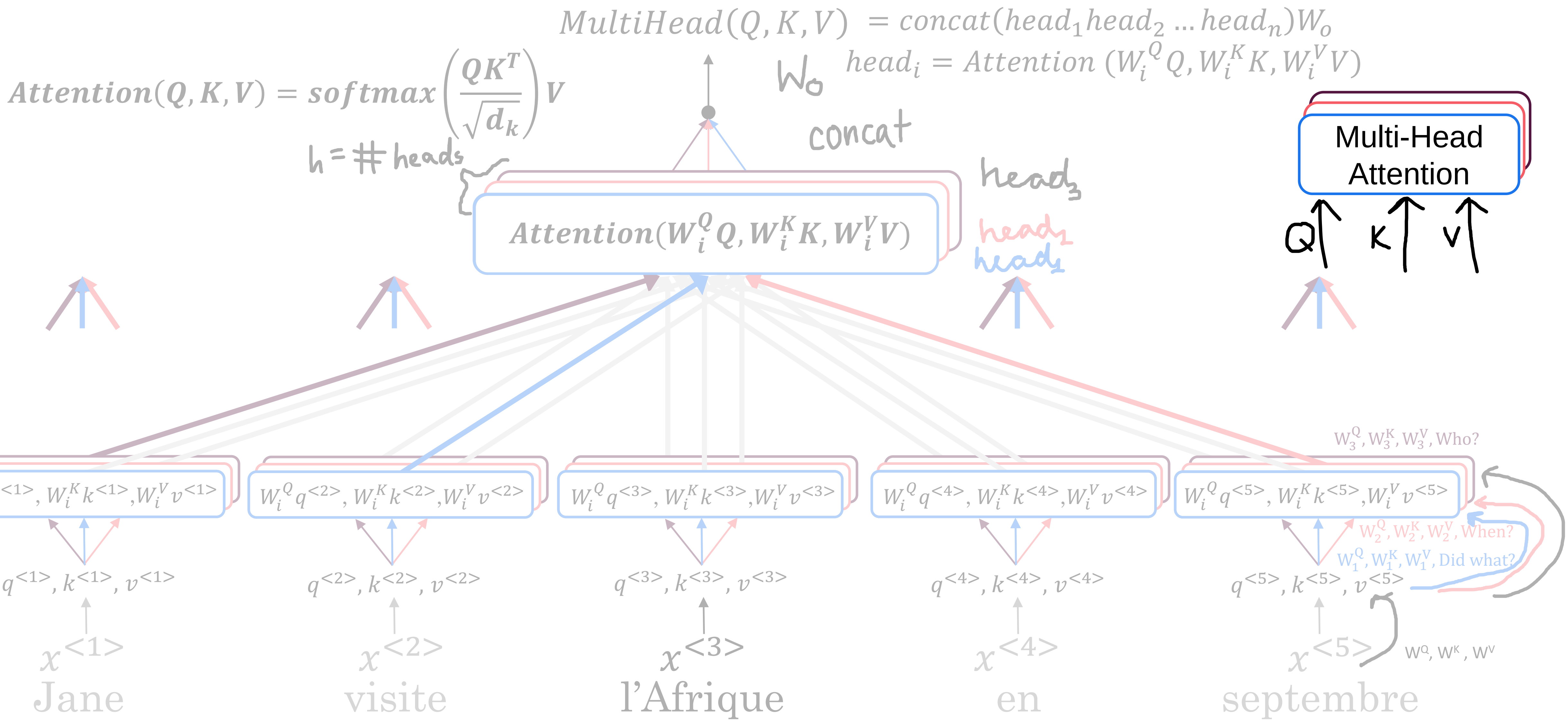
Andrew Ng

deeplearning.ai

Sequence to sequence models

Multi-Head Attention

Multi-Head Attention



[Vaswani et al. 2017, Attention Is All You Need]

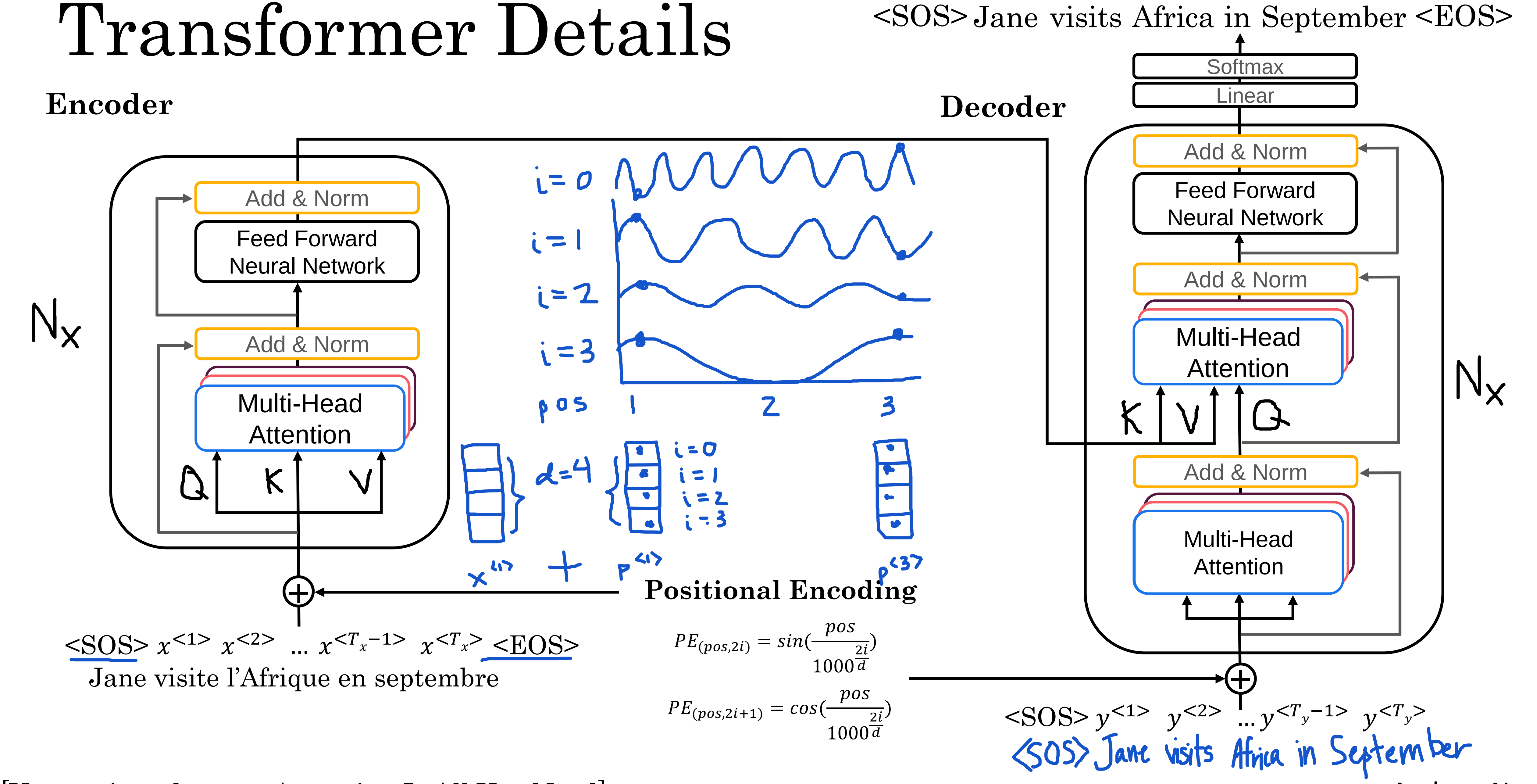
Andrew Ng

Sequence to sequence models

Transformers

deeplearning.ai

Transformer Details



[Vaswani et al. 2017, Attention Is All You Need]

Andrew Ng