

Copyright Notice

These slides are distributed under the Creative Commons License.

[DeepLearning.AI](#) makes these slides available for educational purposes. You may not use or distribute these slides for commercial purposes. You may make copies of these slides and use or distribute them for educational purposes as long as you cite [DeepLearning.AI](#) as the source of the slides.

For the rest of the details of the license, see <https://creativecommons.org/licenses/by-sa/2.0/legalcode>



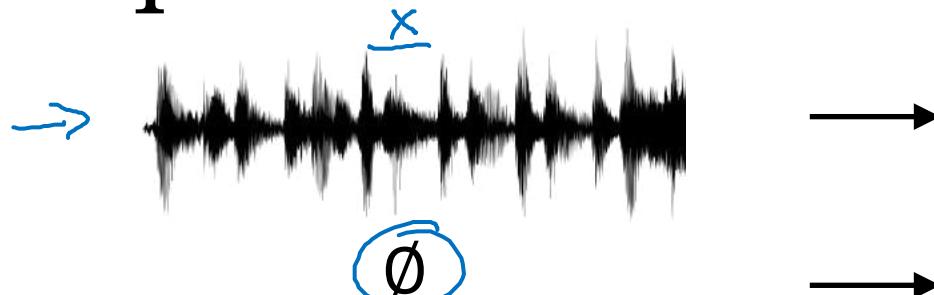
deeplearning.ai

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.

$\rightarrow \underline{x^{<1>}}$ $x^{<2>}$ $x^{<3>}$ - - - - $x^{<t>}$ - - - - $x^{<q>}$

$$T_x = q$$

$\rightarrow y:$

| | 0 | | 0 0 0 0
 $y^{<1>}$ $y^{<2>}$ $y^{<3>}$ - - - - $y^{<q>}$

$$T_y = q$$

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

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

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

$$T_y^{(i)}$$

15

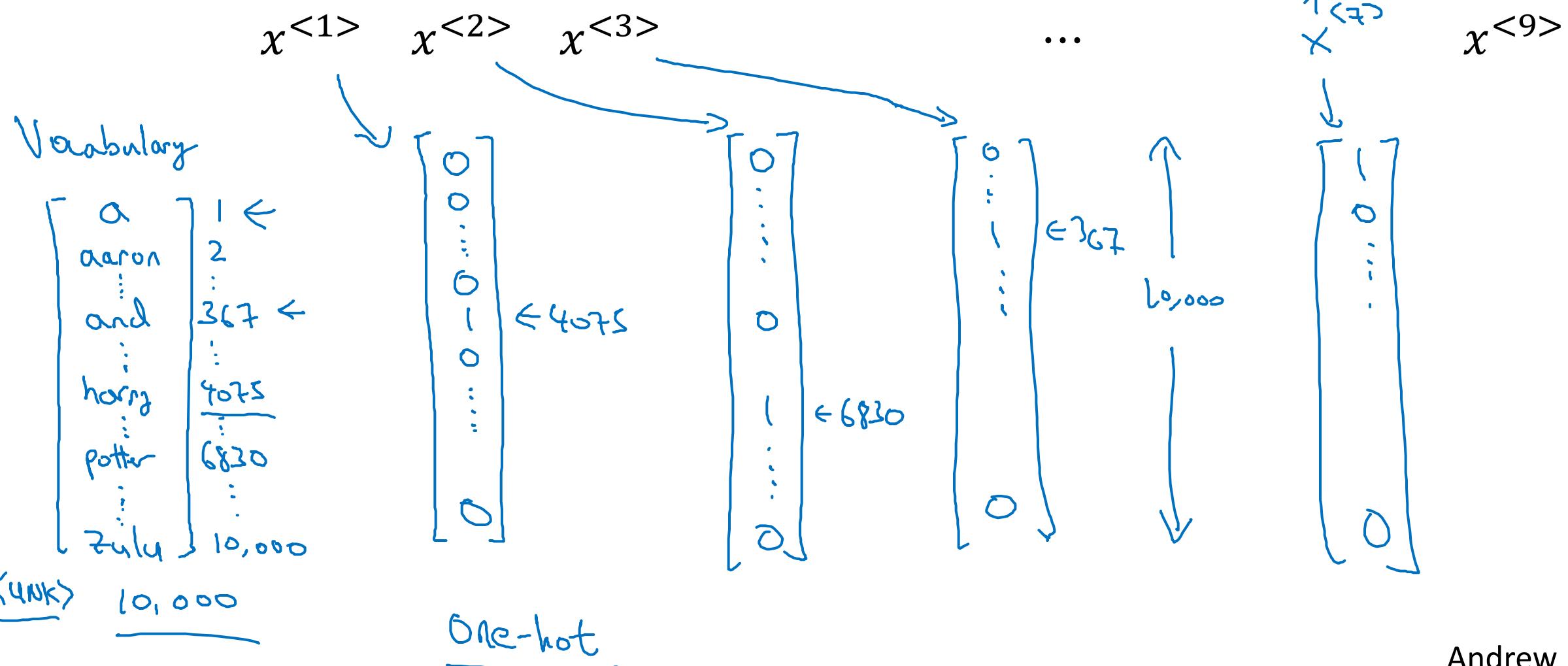
Representing words

$$x^{<\leftrightarrow>} \quad x \rightarrow y$$

(x, 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

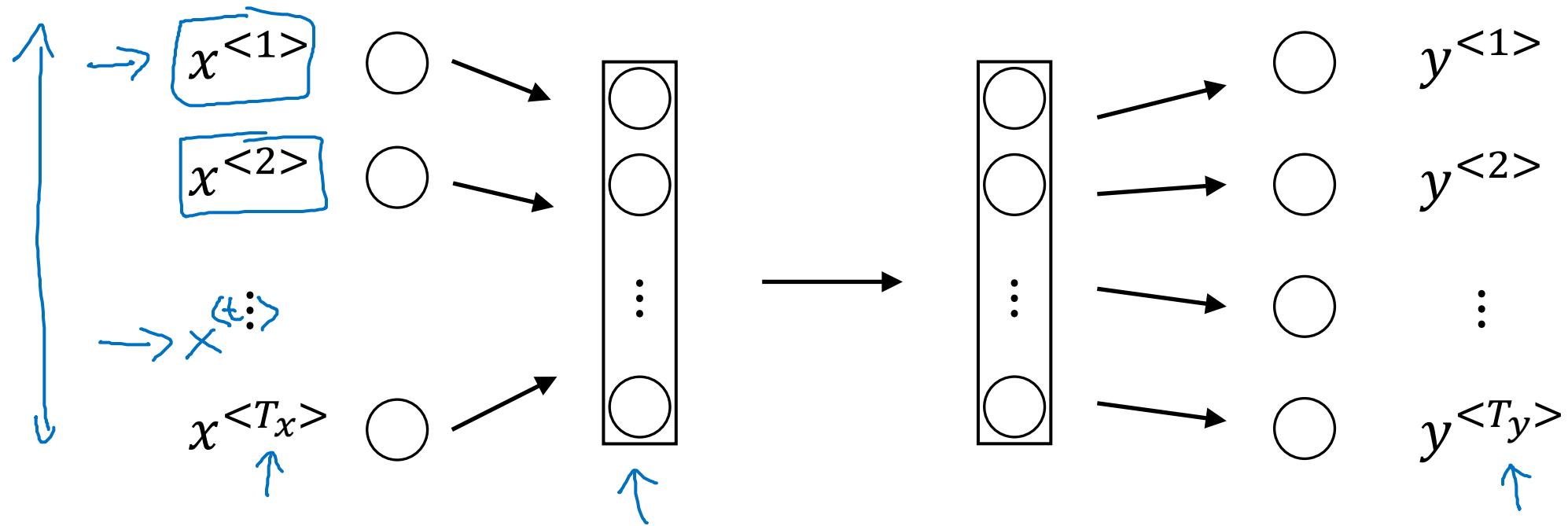


deeplearning.ai

Recurrent Neural Networks

Recurrent Neural Network Model

Why not a standard network?

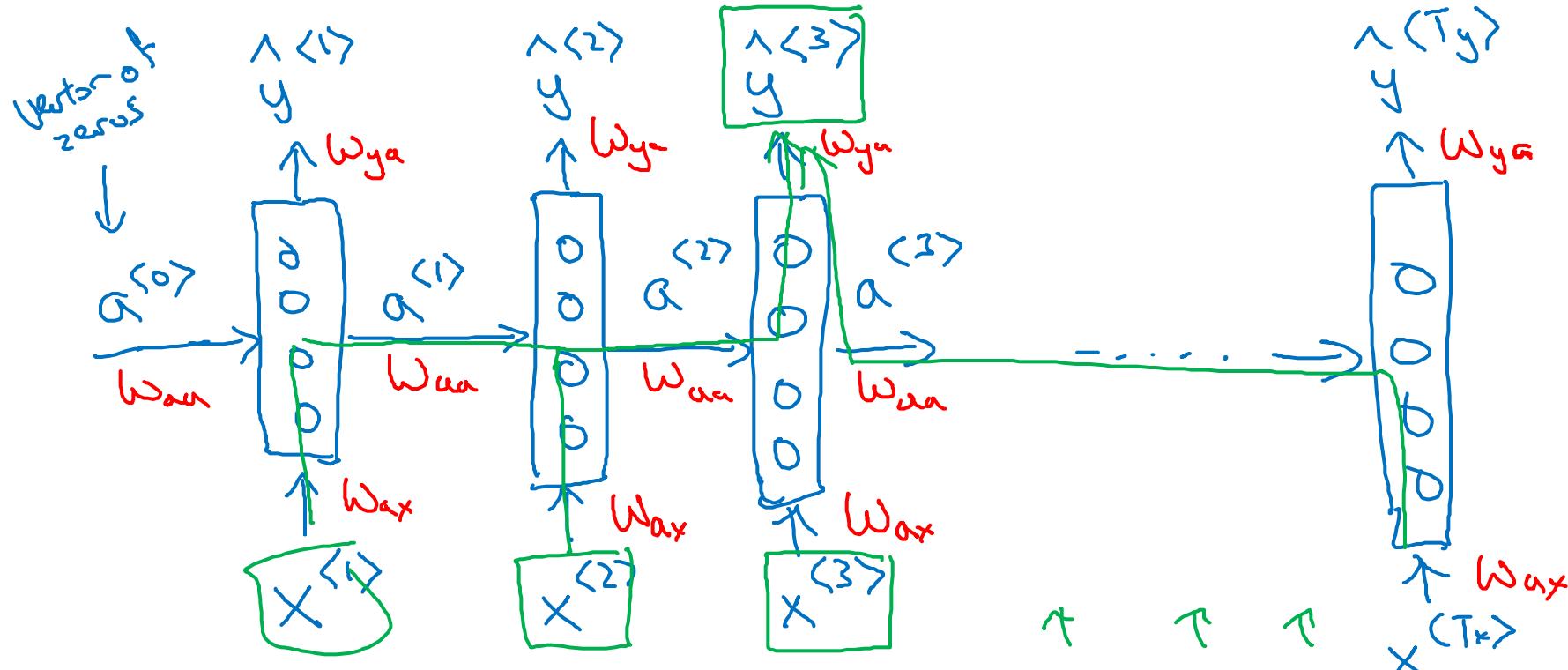


Problems:

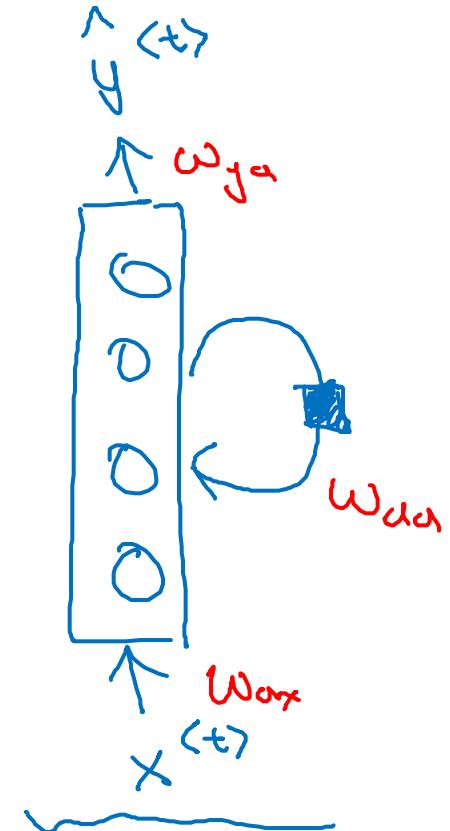
- - Inputs, outputs can be different lengths in different examples.
- - Doesn't share features learned across different positions of text.

Recurrent Neural Networks

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



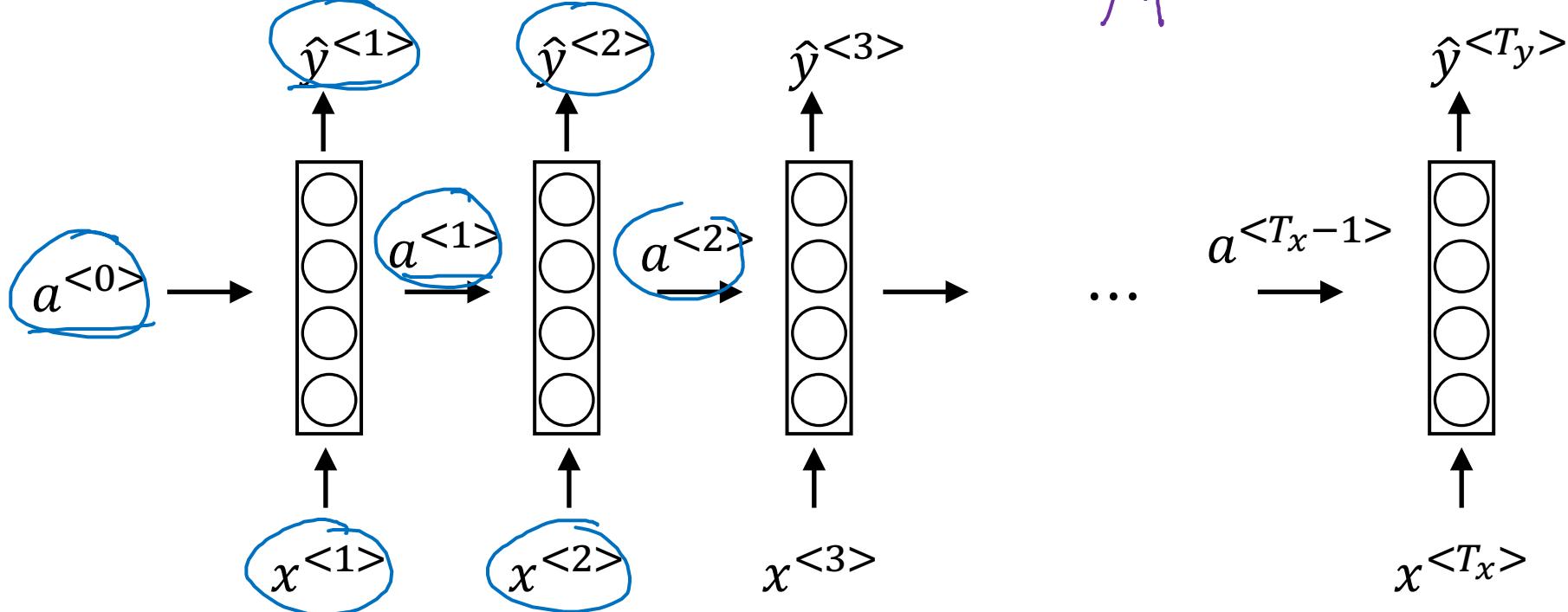
Bidirectional RNN (BRNN)



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
 $x^{(t)}$: $10,000$

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

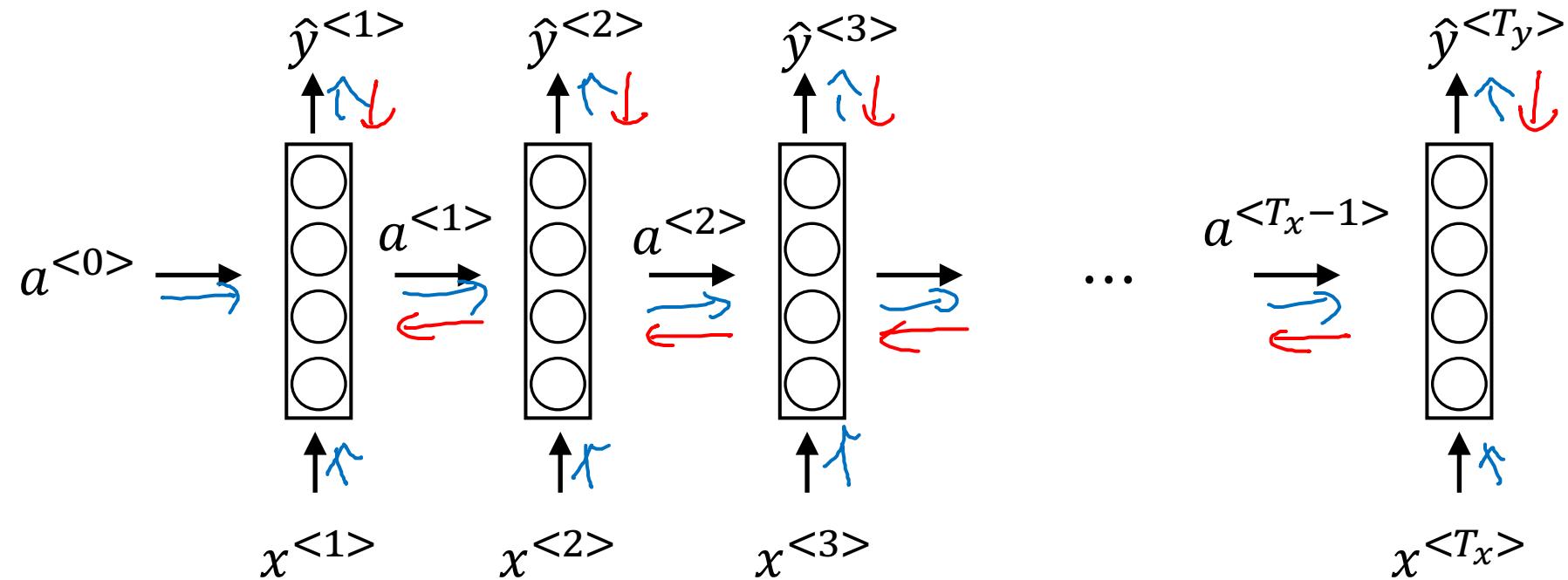


deeplearning.ai

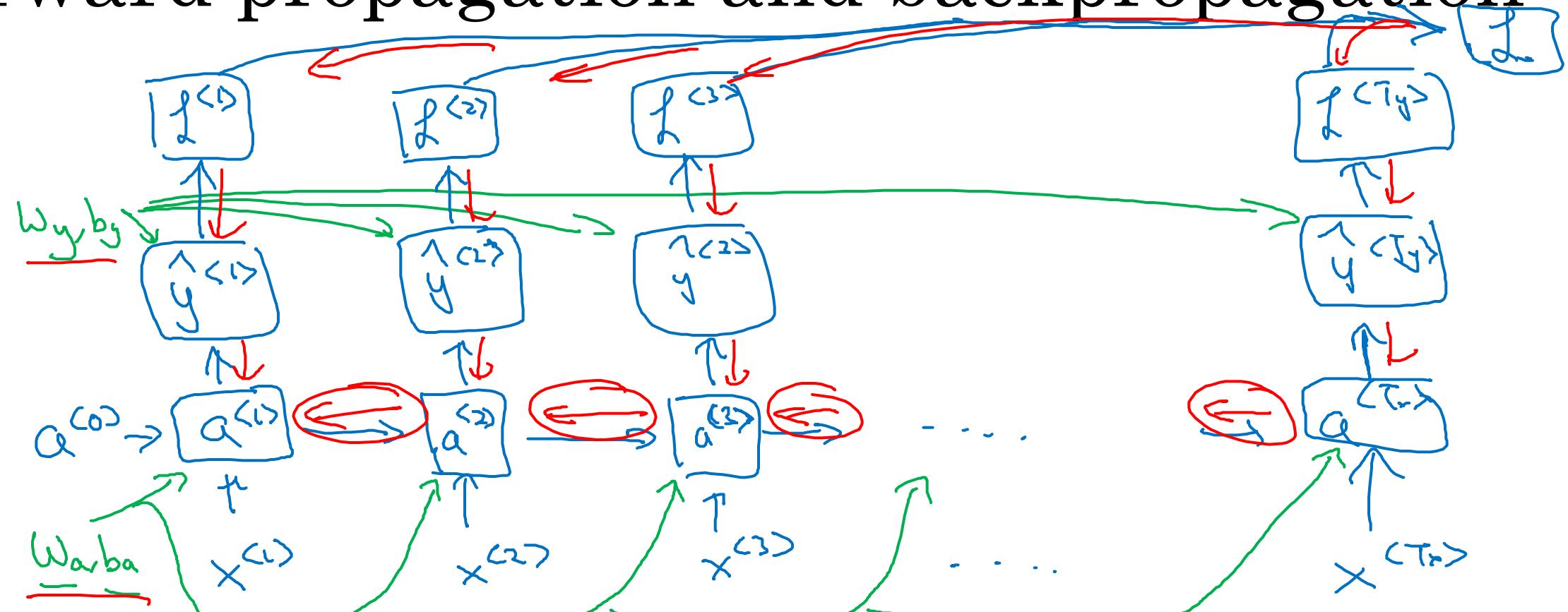
Recurrent Neural Networks

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



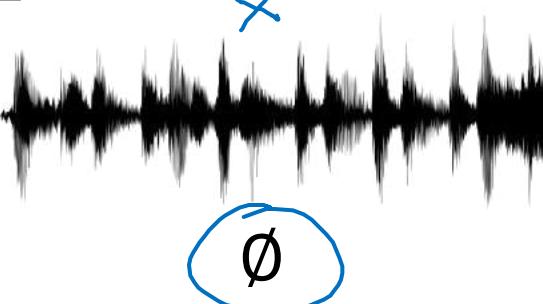
deeplearning.ai

Recurrent Neural Networks

Different types
of RNNs

Examples of sequence data

Speech recognition



T_x T_y

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



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.

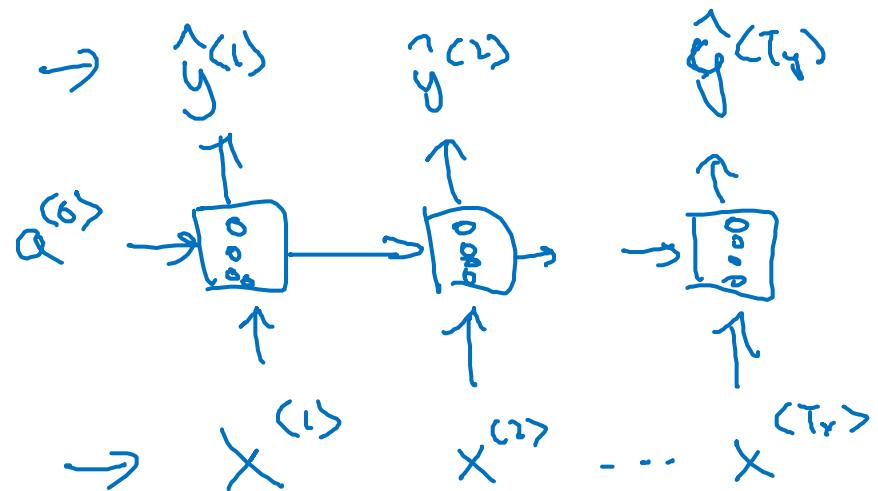


Yesterday, **Harry Potter**
met **Hermione Granger**.

Andrew Ng

Examples of RNN architectures

$$T_x = T_y$$

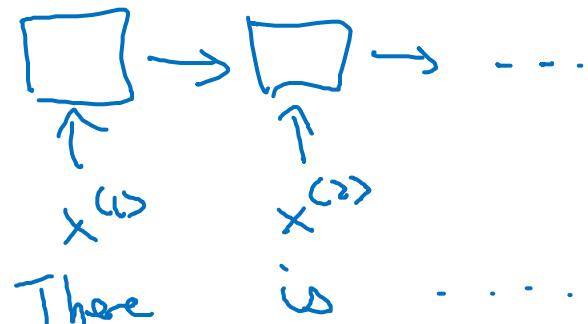


Many-to-many

Sentiment classification

$x = \text{text}$

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

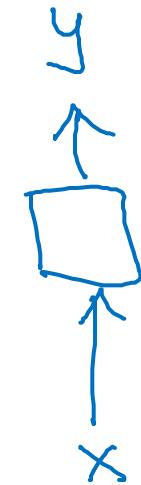
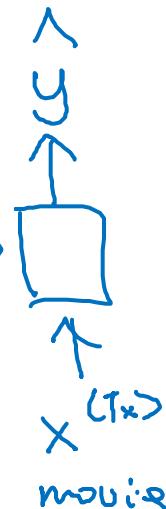


These

is

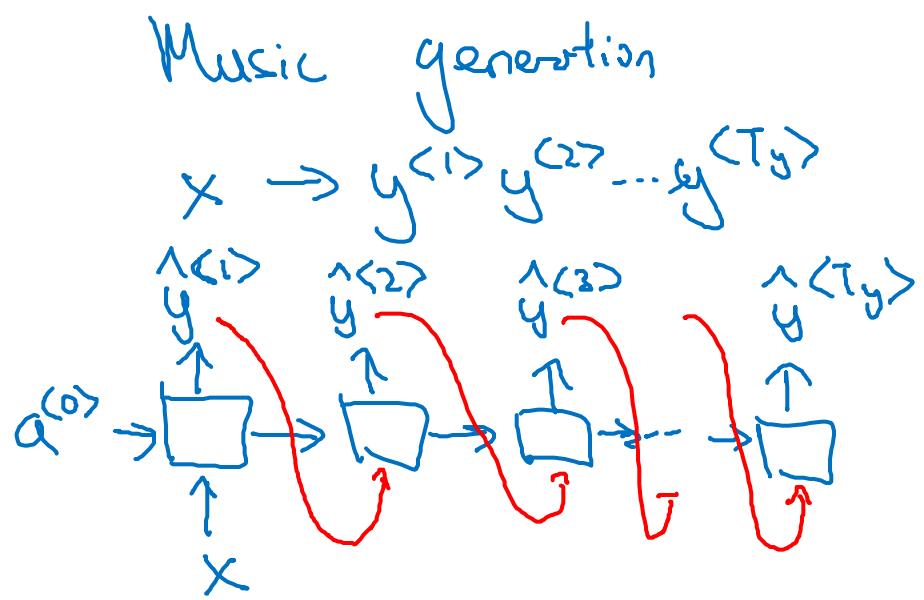
.....

Many - to - one



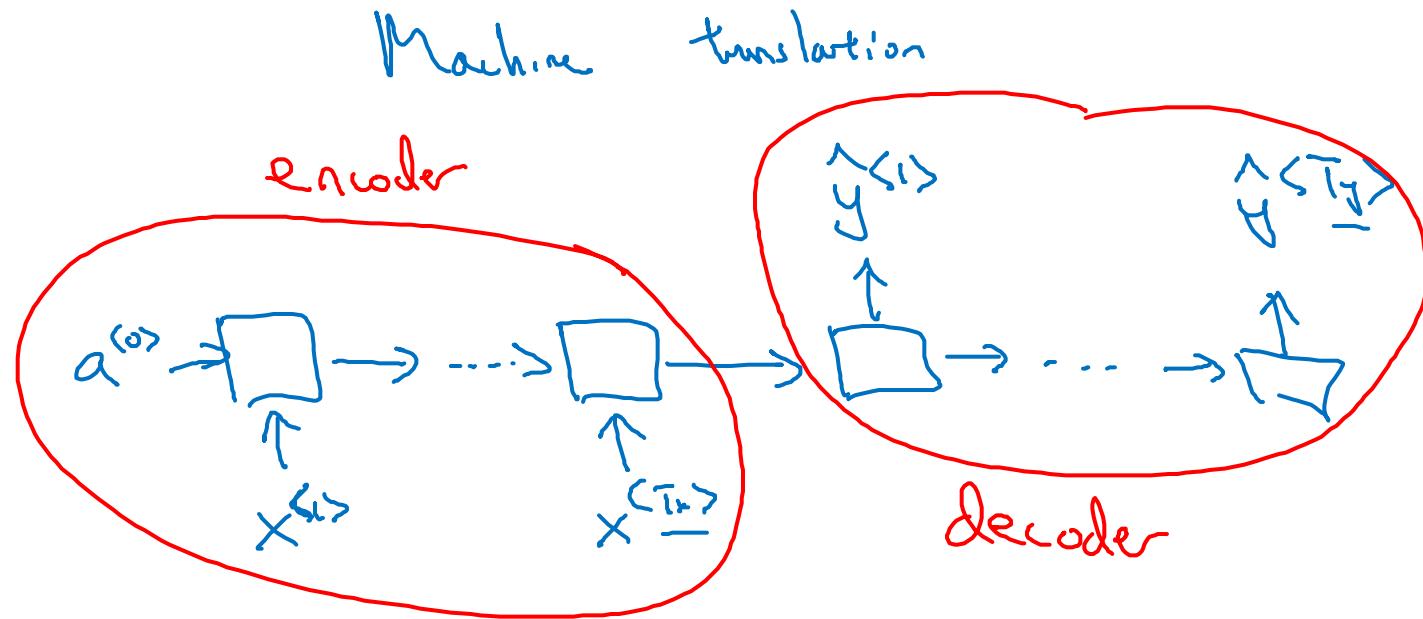
One - to -
one

Examples of RNN architectures



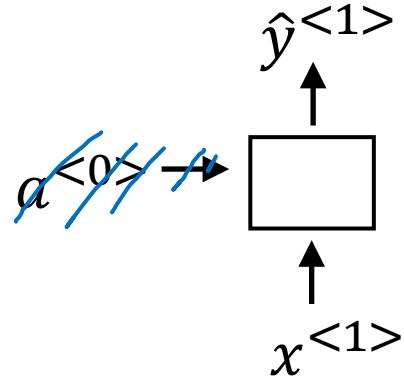
One-to-many

$$x = \phi$$

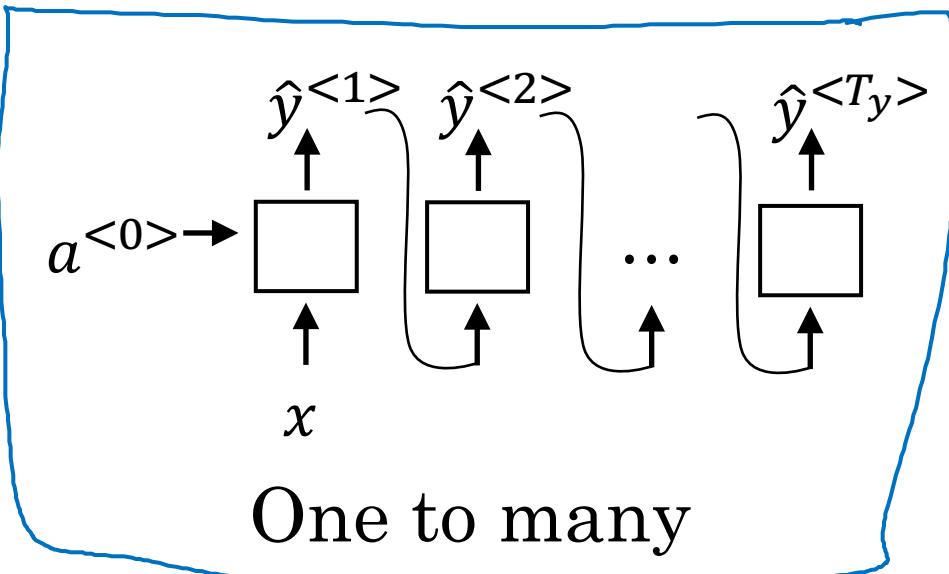


Many - to - many

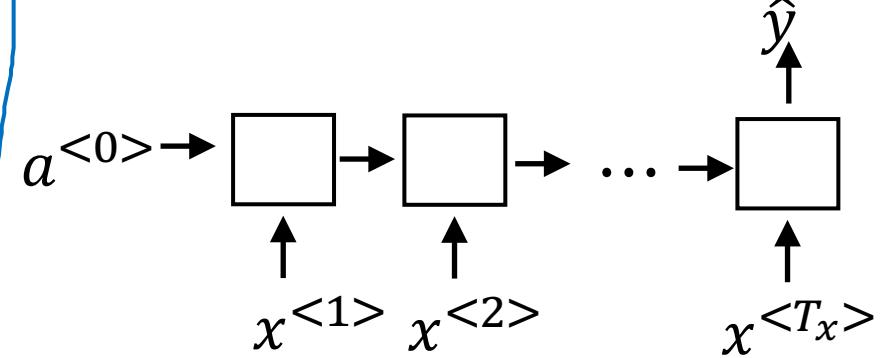
Summary of RNN types



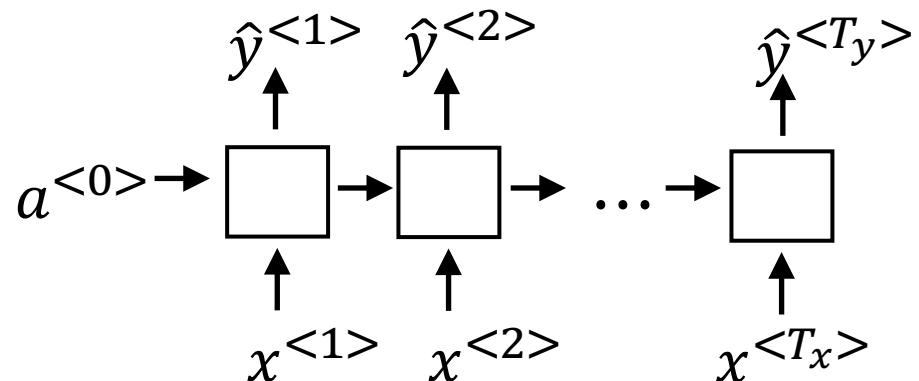
One to one



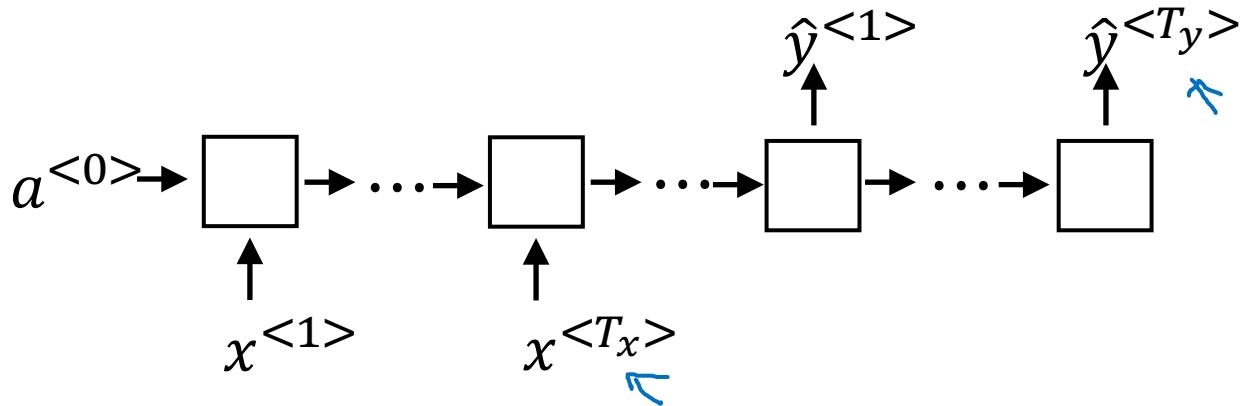
One to many



Many to one



Many to many
 $T_x = T_y$



Many to many
 $T_x < T_y$



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. $\downarrow <\text{EOS}>$

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

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

...

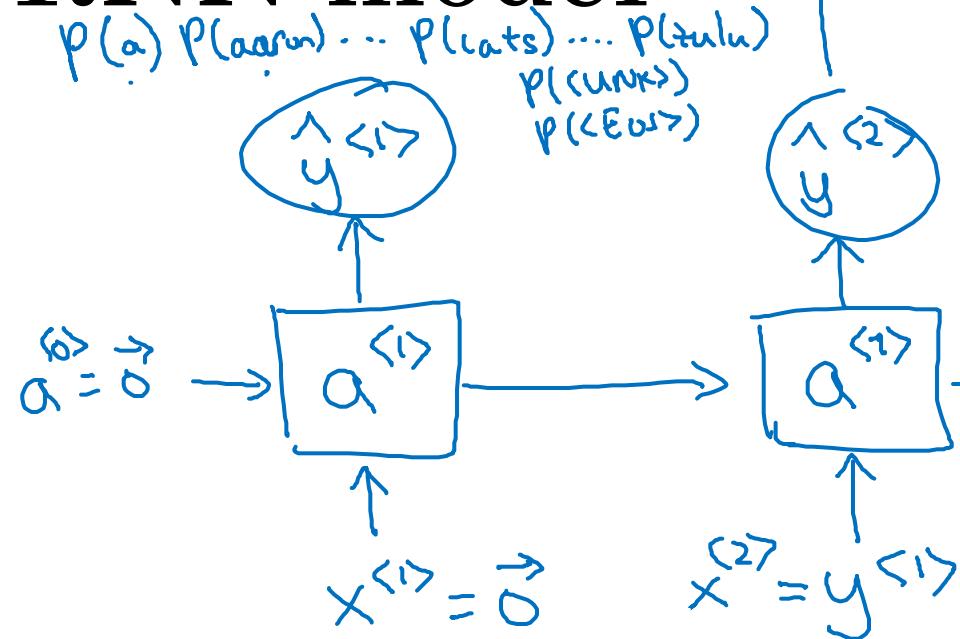
$y^{(8)}$ $y^{(9)}$

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

$<\text{UNK}>$

10,000

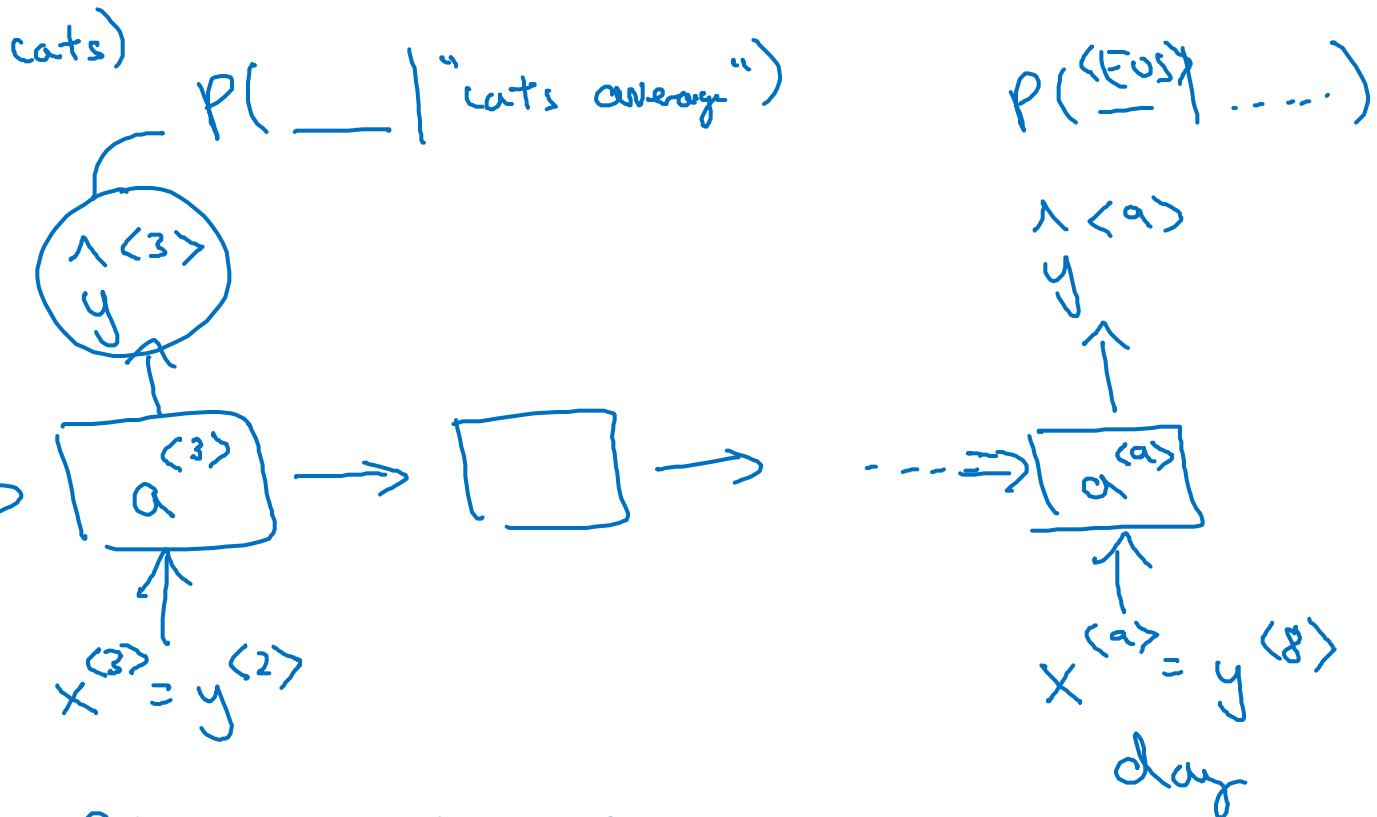
RNN model



\rightarrow Cats average 15 hours of sleep a day. $\underline{\text{Cats}}$ $\underline{\text{Average}}$ $\underline{\text{hours}}$ $\underline{\text{day}}$

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

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



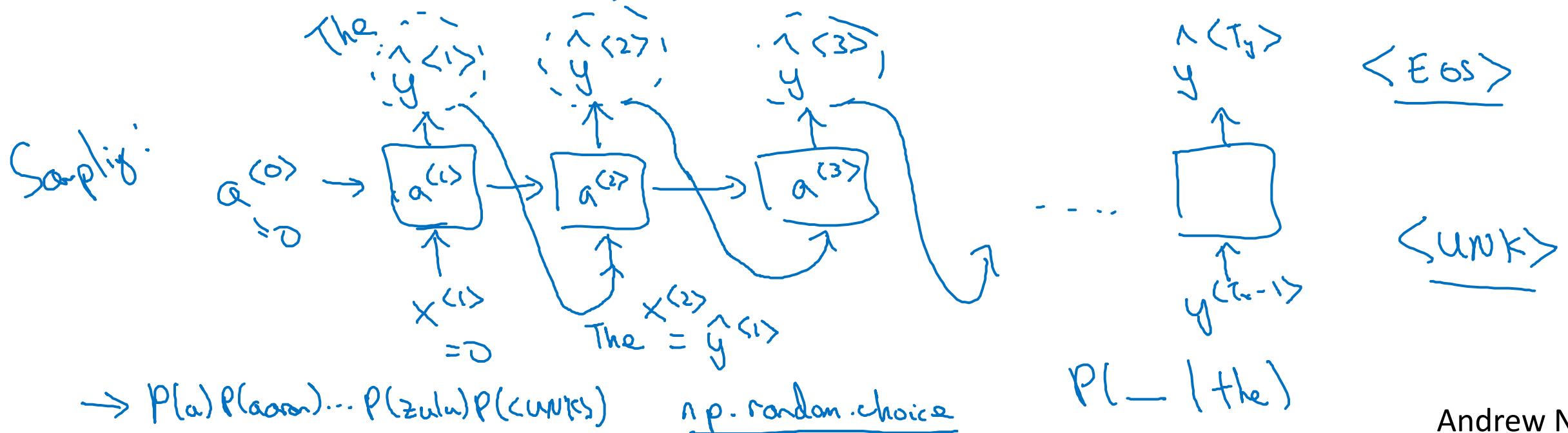
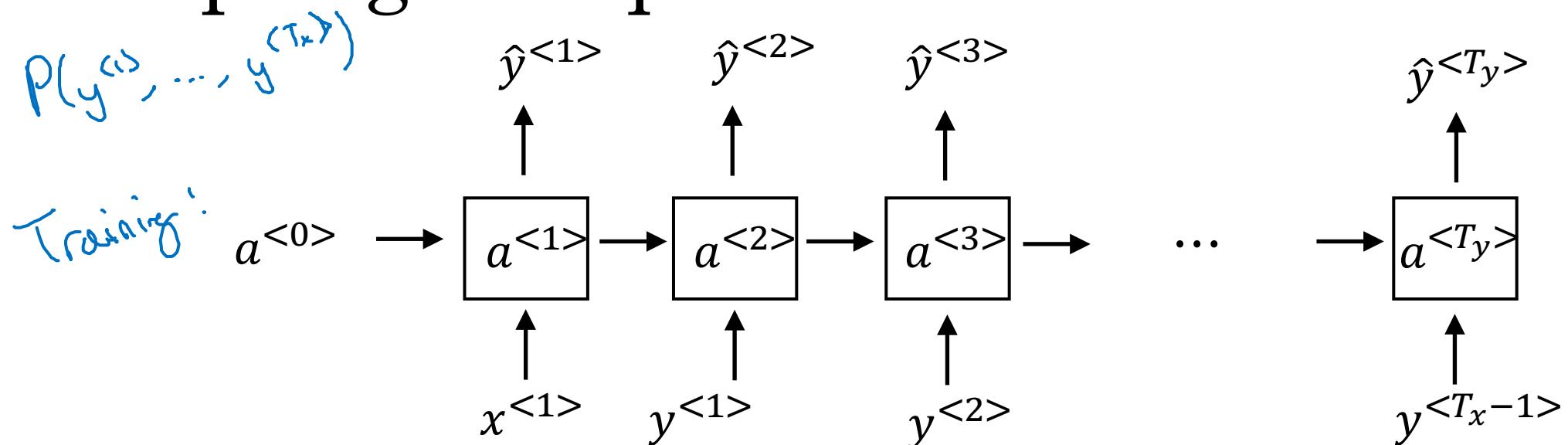


deeplearning.ai

Recurrent Neural Networks

Sampling novel
sequences

Sampling a sequence from a trained RNN



Character-level language model

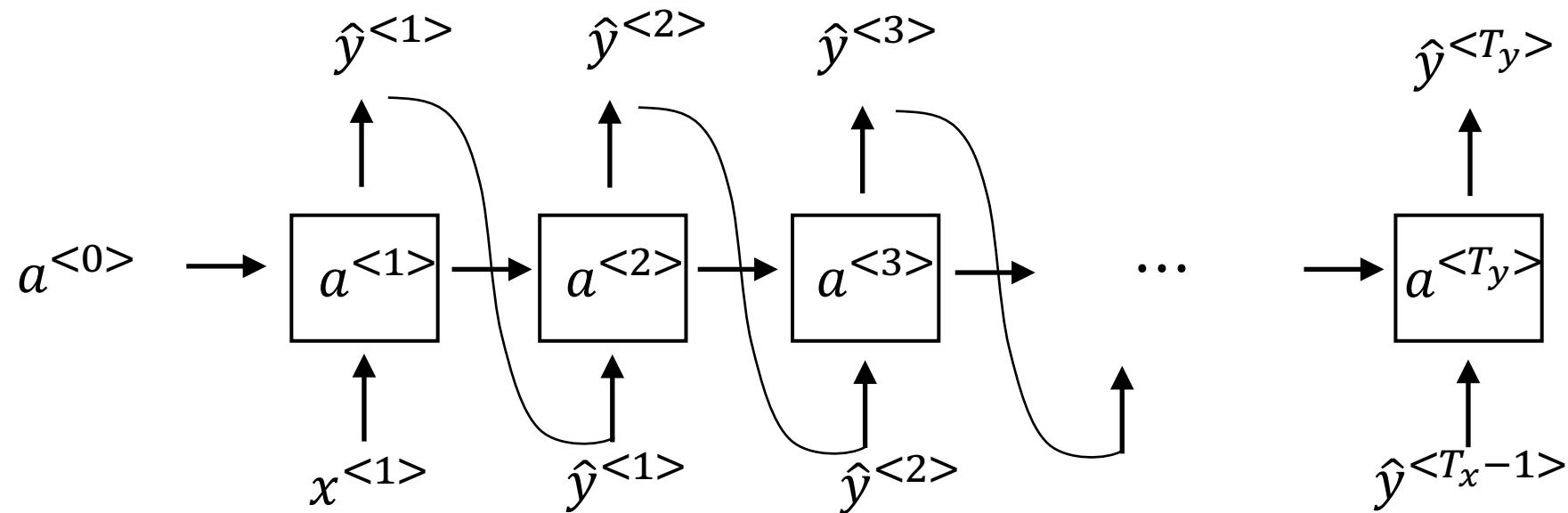
→ Vocabulary = [a, aaron, ..., zulu, <UNK>] ←

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

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

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.

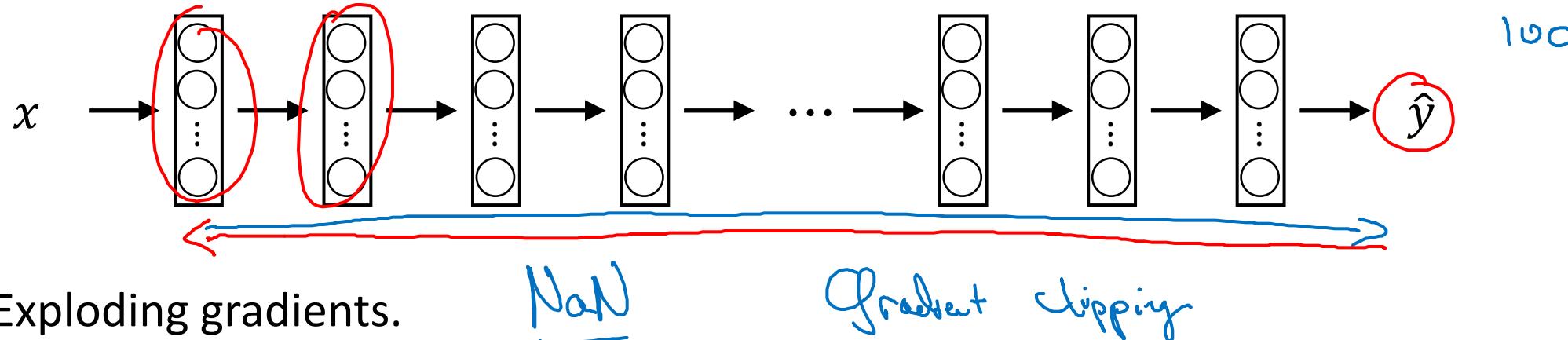
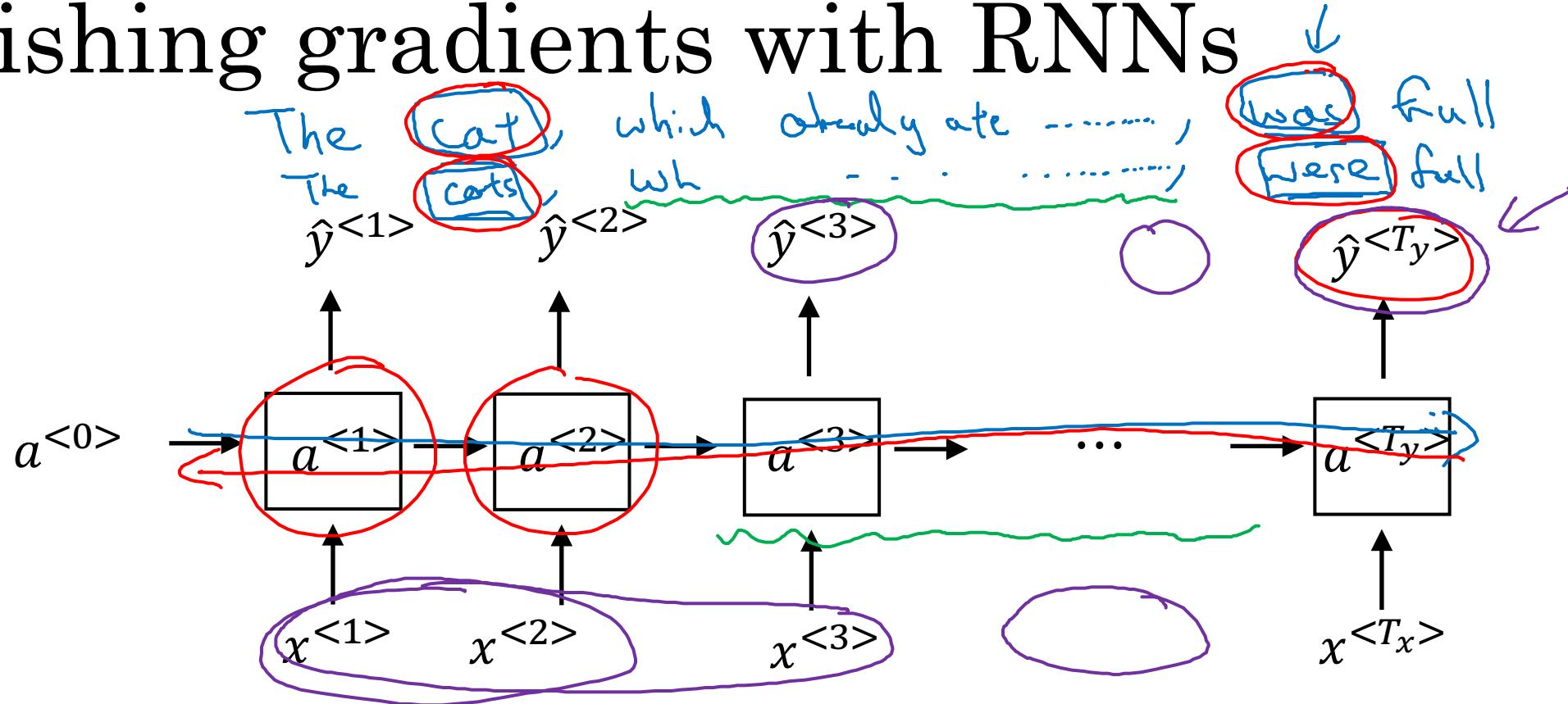


deeplearning.ai

Recurrent Neural Networks

Vanishing gradients with RNNs

Vanishing gradients with RNNs



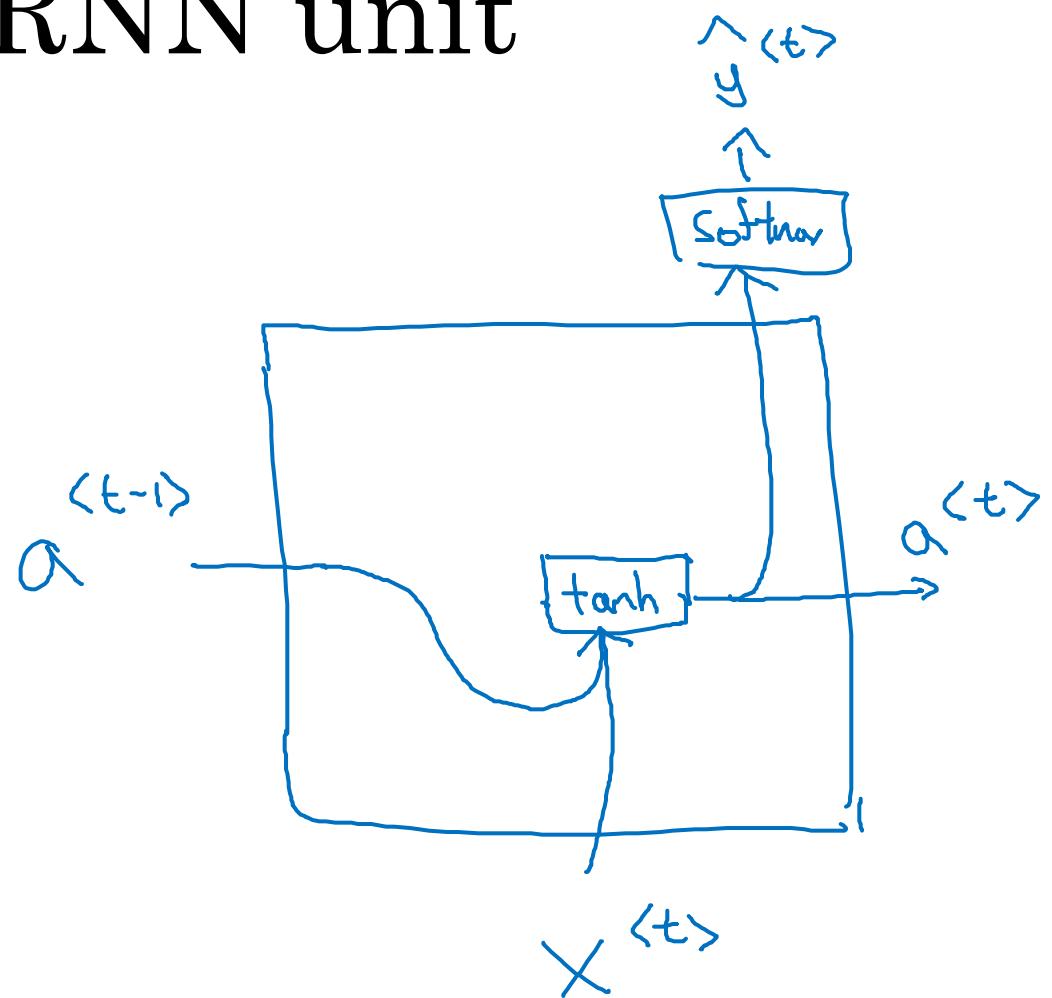


deeplearning.ai

Recurrent Neural Networks

Gated Recurrent Unit (GRU)

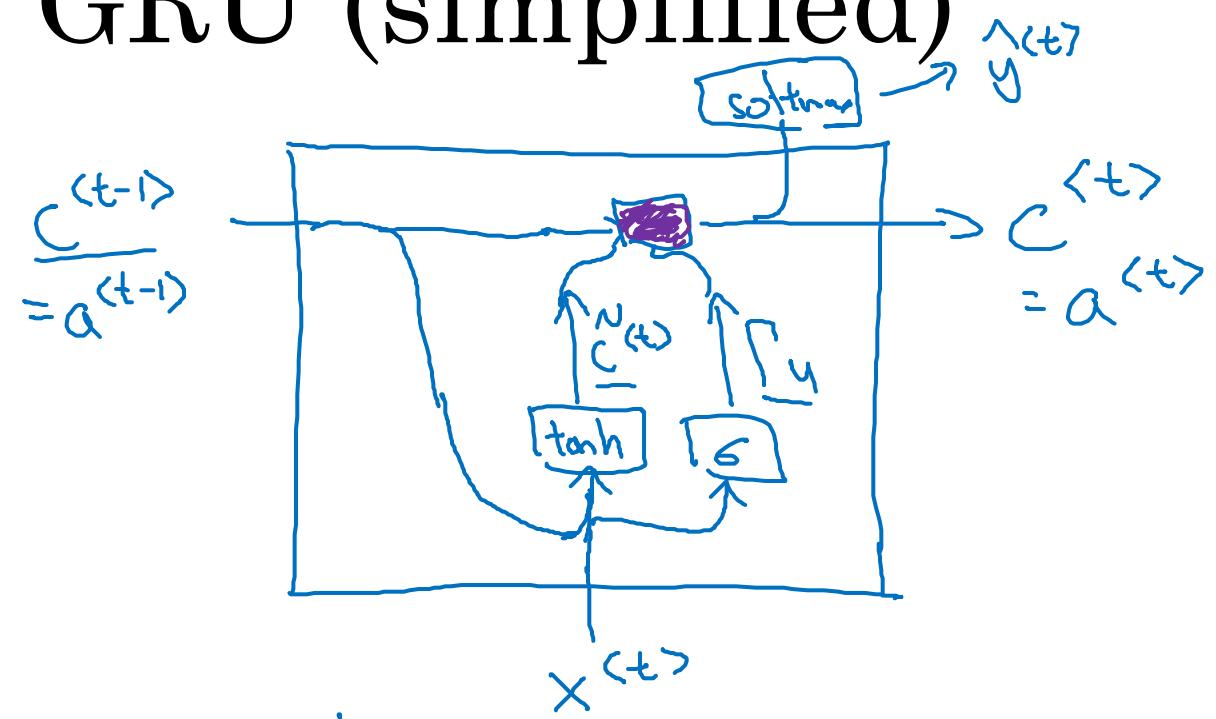
RNN unit



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

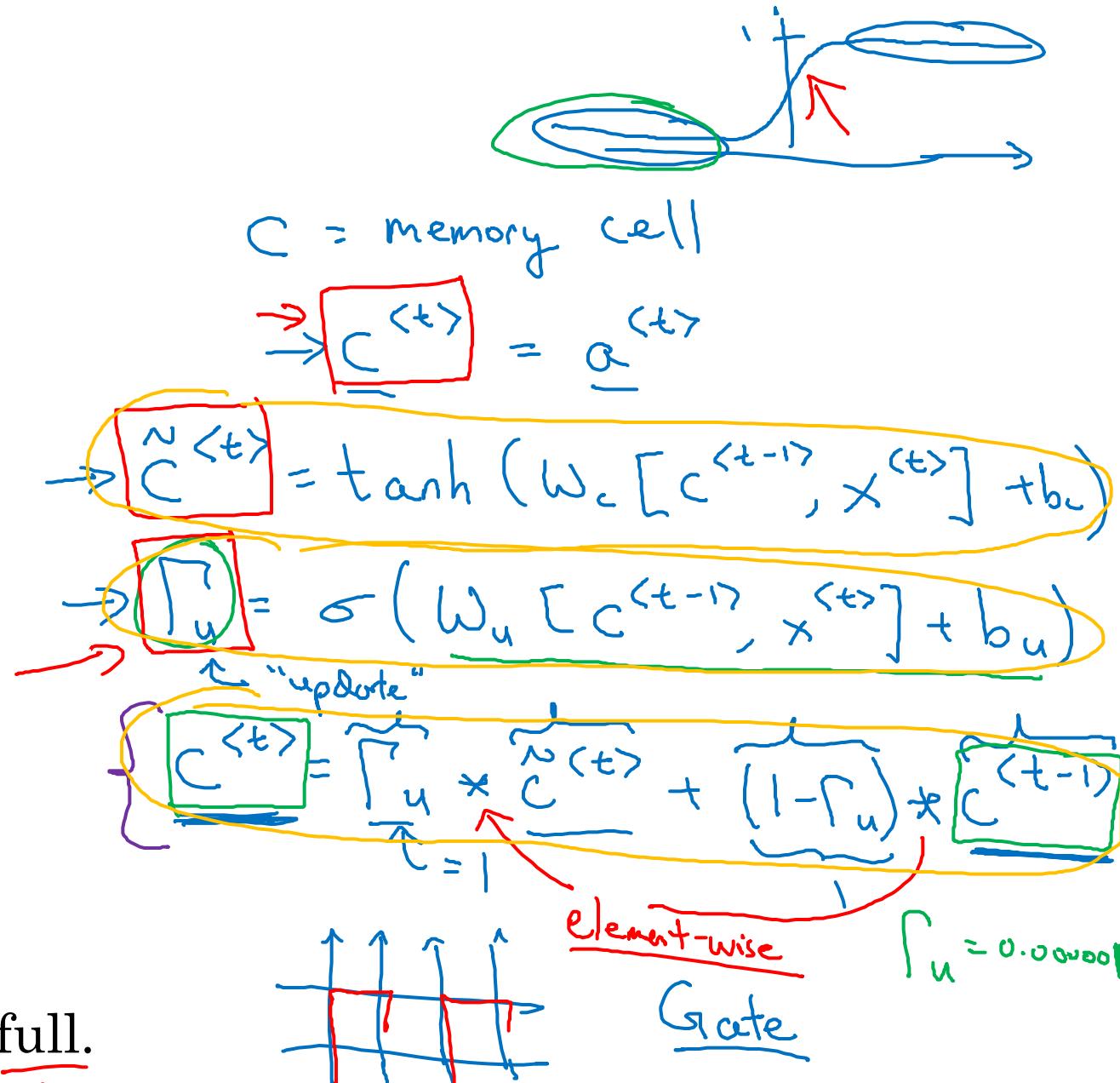
Below the equation, there is a wavy line representing the hidden state $a^{(t)}$. A blue bracket under the wavy line indicates its dimensionality. An arrow labeled "tanh" points to the tanh function in the diagram above.

GRU (simplified)



$r_u = 1$ $r_u = 0$ $r_u = 0$ $r_u = 0$...

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

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

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

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

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

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

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

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

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

Γ_u

$$a^{<t>} = c^{<t>} \quad (\Gamma_e)$$

LSTM

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

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

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

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

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

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

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

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

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

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

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

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

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

LSTM in pictures

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

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

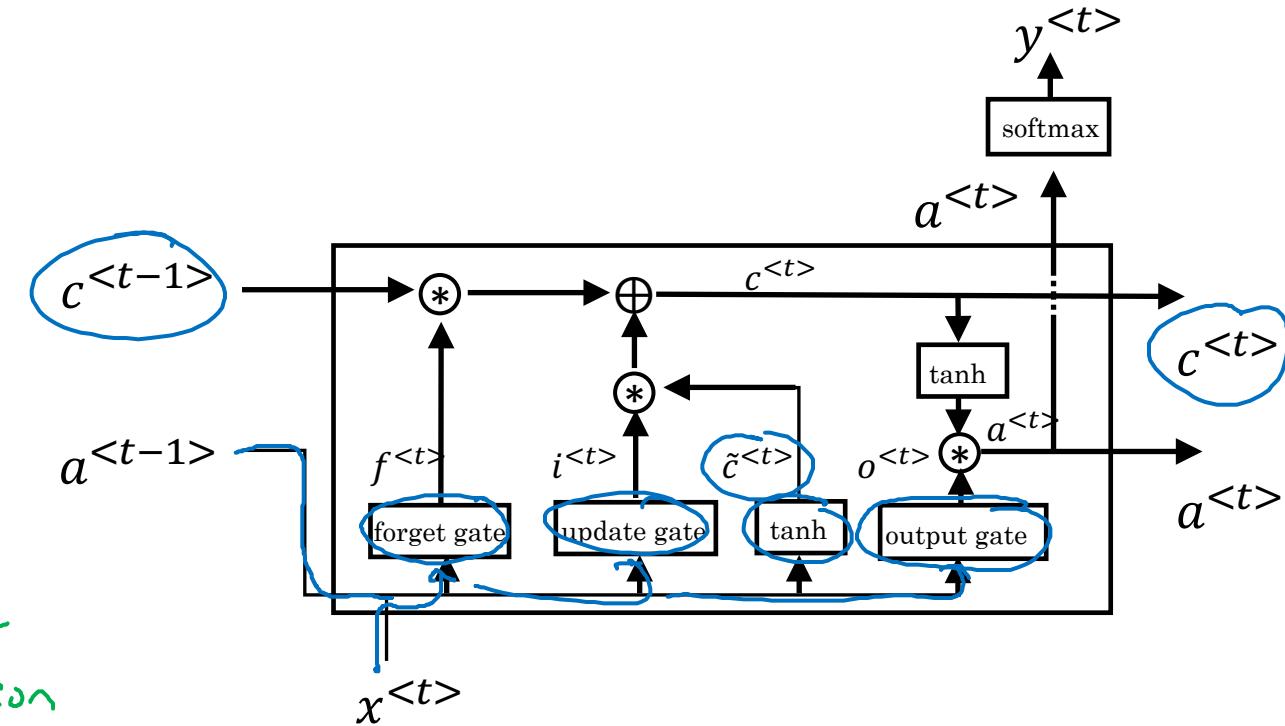
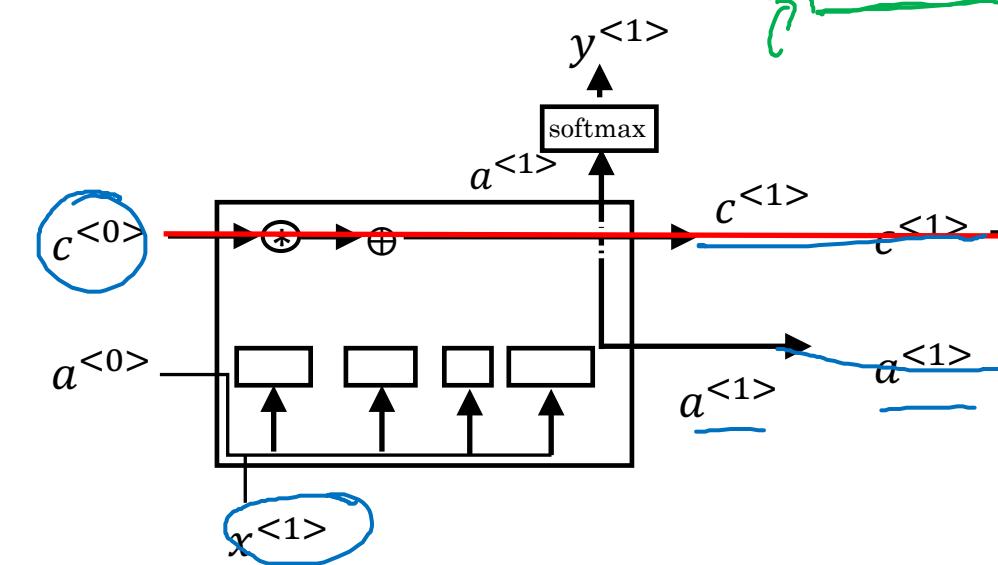
$$\Gamma_f = \sigma(W_f[a^{<t-1>}, x^{<t>}] + b_f)$$

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

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

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

peephole connection



Andrew Ng



deeplearning.ai

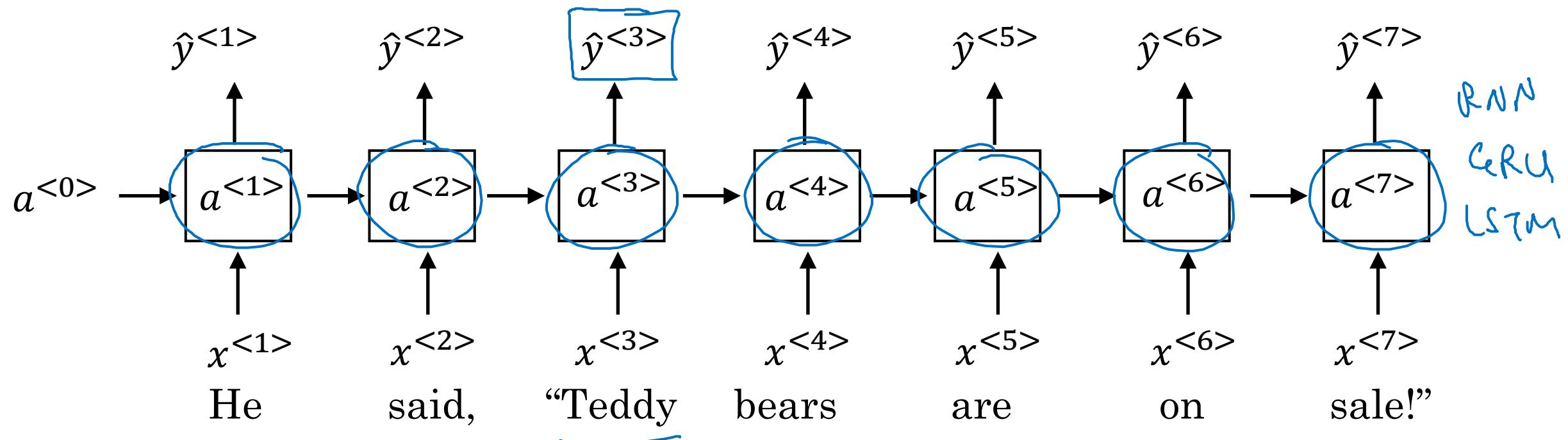
Recurrent Neural Networks

Bidirectional RNN

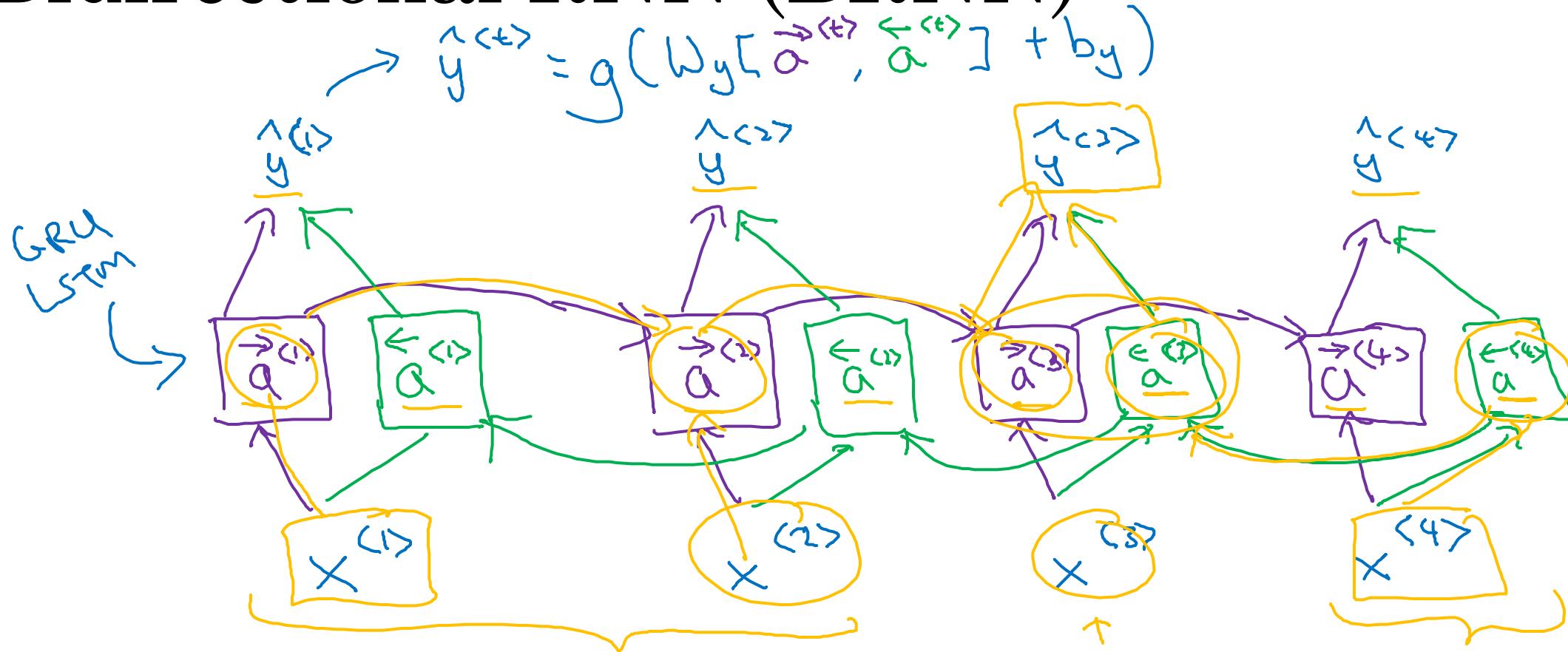
Getting information from the future

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 ..."

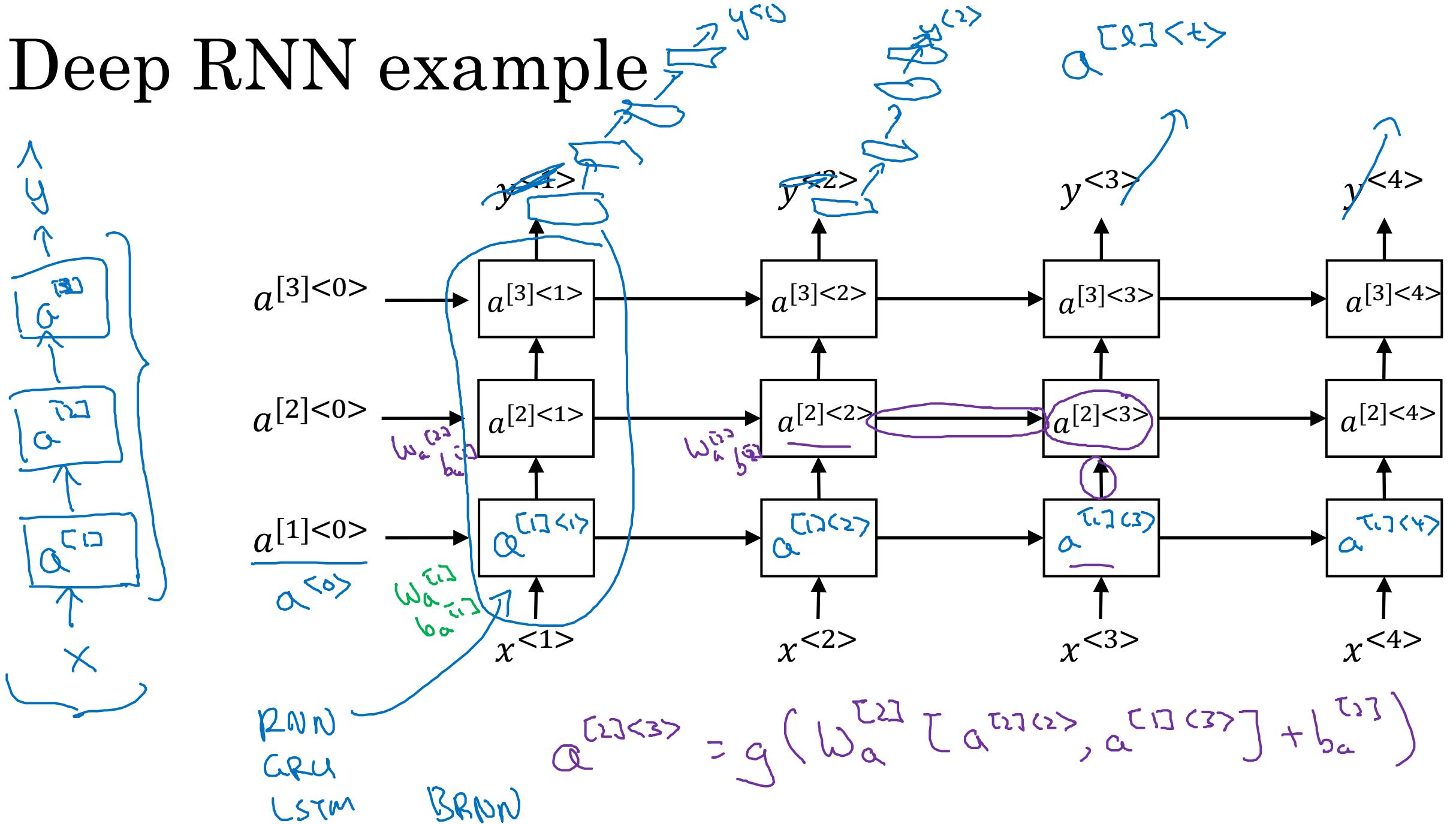


deeplearning.ai

Recurrent Neural Networks

Deep RNNs

Deep RNN example



Copyright Notice

These slides are distributed under the Creative Commons License.

[DeepLearning.AI](#) makes these slides available for educational purposes. You may not use or distribute these slides for commercial purposes. You may make copies of these slides and use or distribute them for educational purposes as long as you cite [DeepLearning.AI](#) as the source of the slides.

For the rest of the details of the license, see <https://creativecommons.org/licenses/by-sa/2.0/legalcode>



deeplearning.ai

NLP and Word Embeddings

Word representation

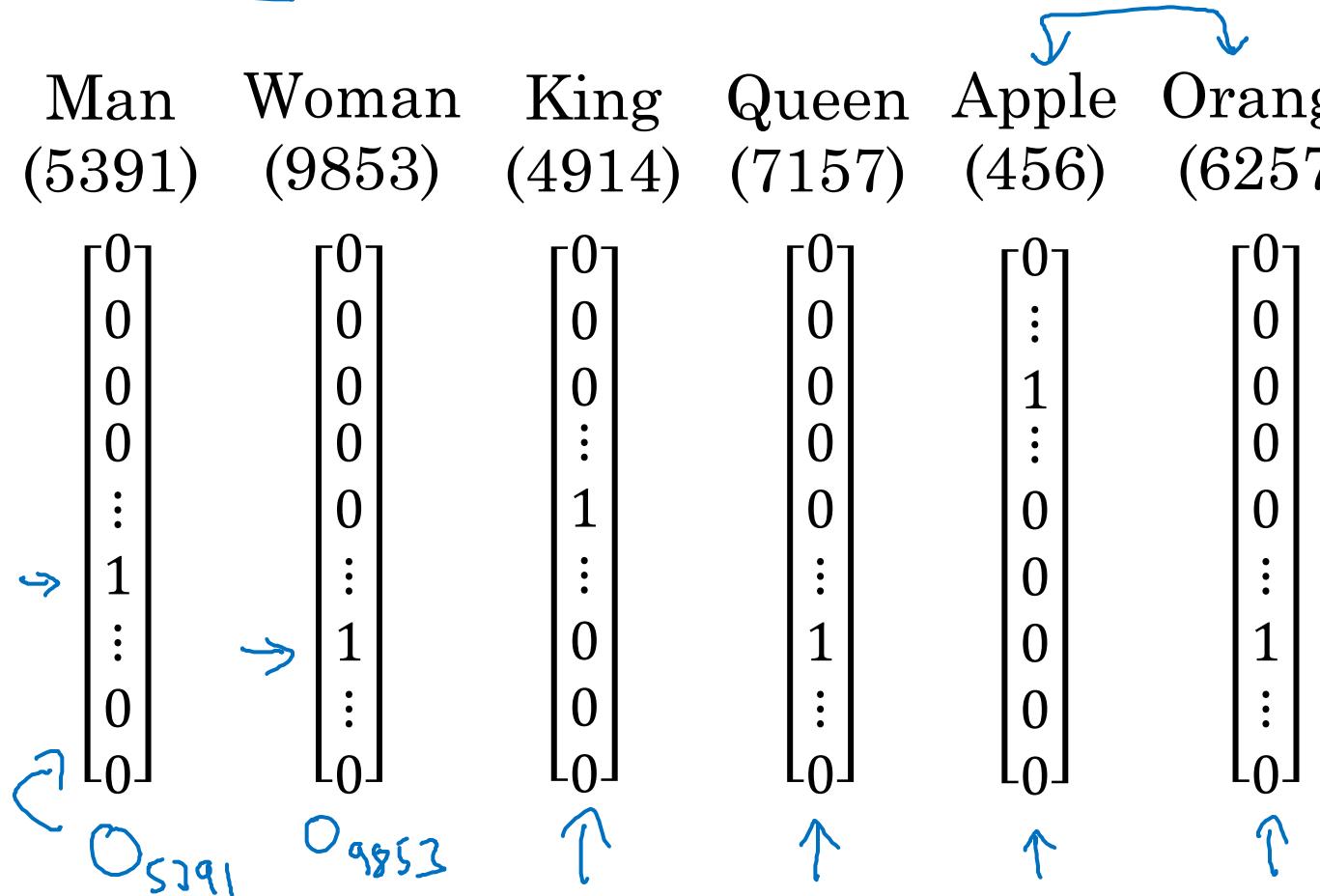
Word representation

$$V = [a, \text{aaron}, \dots, \text{zulu}, \text{<UNK>}]$$

$$|V| = 10,000$$

1-hot representation

Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
---------------	-----------------	----------------	-----------------	----------------	------------------



I want a glass of orange juice.
I want a glass of apple ____.

Featurized representation: word embedding

	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						
Color						
Verb						

Handwritten annotations:

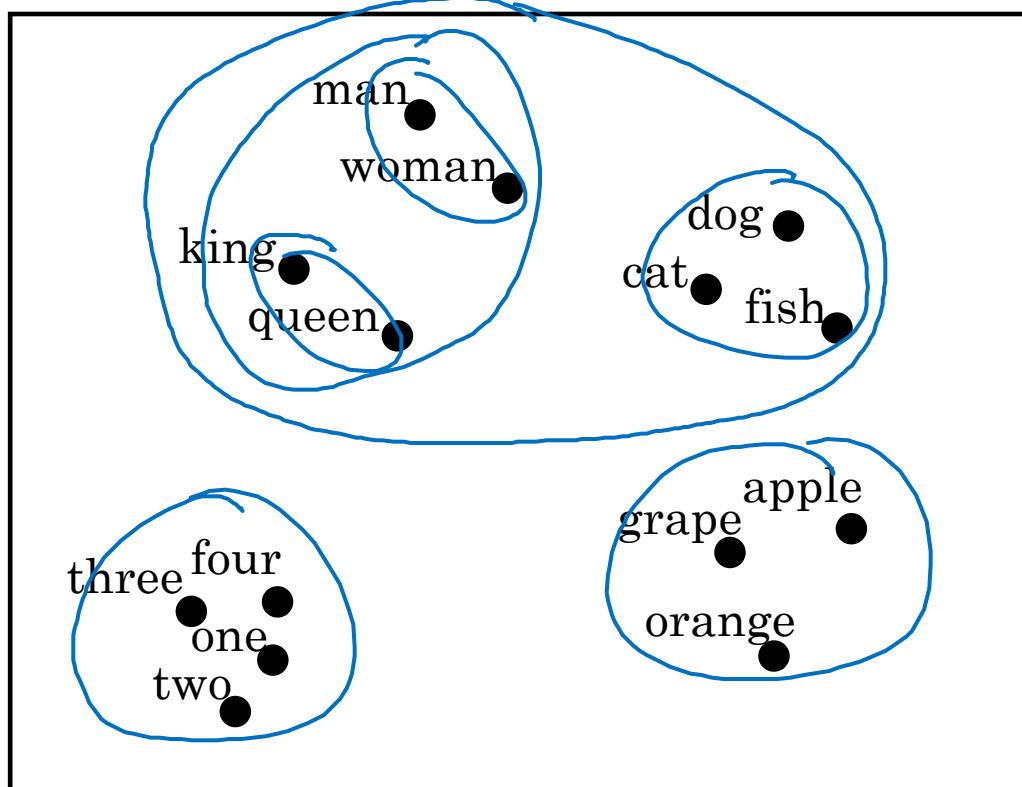
- Column 1 (Man) has values: -1, 0.01, 0.03, 0.04, : (colon), , , , . A bracket below it is labeled e_{5391} .
- Column 2 (Woman) has values: 1, 0.02, 0.02, 0.01, : (colon), , , . A bracket below it is labeled e_{9853} .
- Row 1 (Gender) has values: -1, 1, -0.95, 0.97, 0.00, 0.01. A blue bracket on the right groups the last five values.
- Row 2 (Royal) has values: 0.01, 0.02, 0.93, 0.95, -0.01, 0.00. A blue bracket on the right groups the last three values.
- Row 3 (Age) has values: 0.03, 0.02, 0.7, 0.69, 0.03, -0.02.
- Row 4 (Food) has values: 0.04, 0.01, 0.02, 0.01, 0.95, 0.97.
- Row 5 (Size) has values: :, :, , , , .
- Row 6 (Cost) has values: , , , , , .
- Row 7 (Color) has values: , , , , , .
- Row 8 (Verb) has values: , , , , , .

Text at the bottom right:

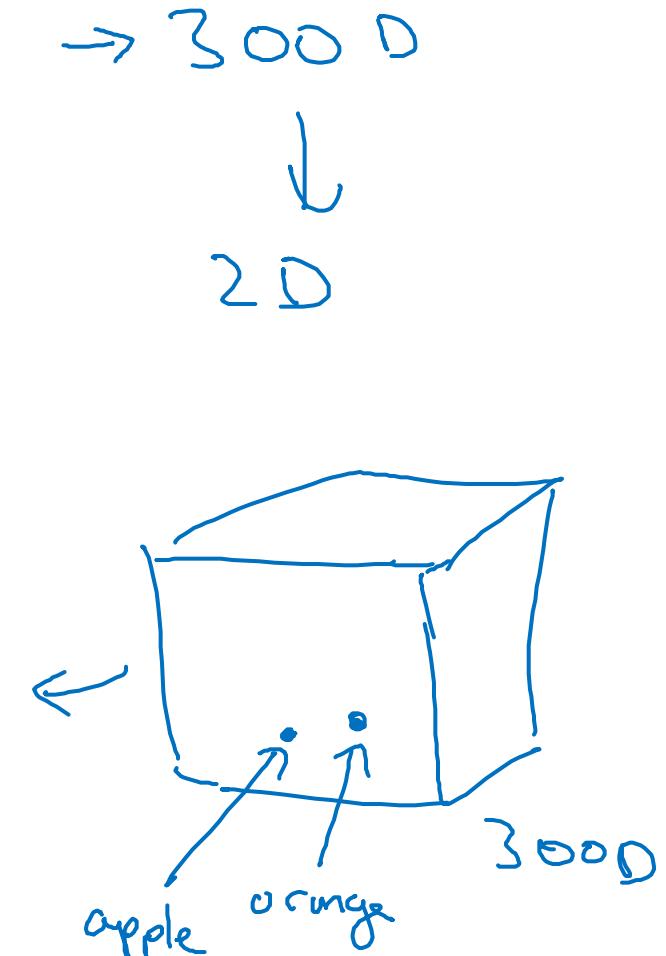
I want a glass of orange juice.
I want a glass of apple juice.

Andrew Ng

Visualizing word embeddings



t-SNE



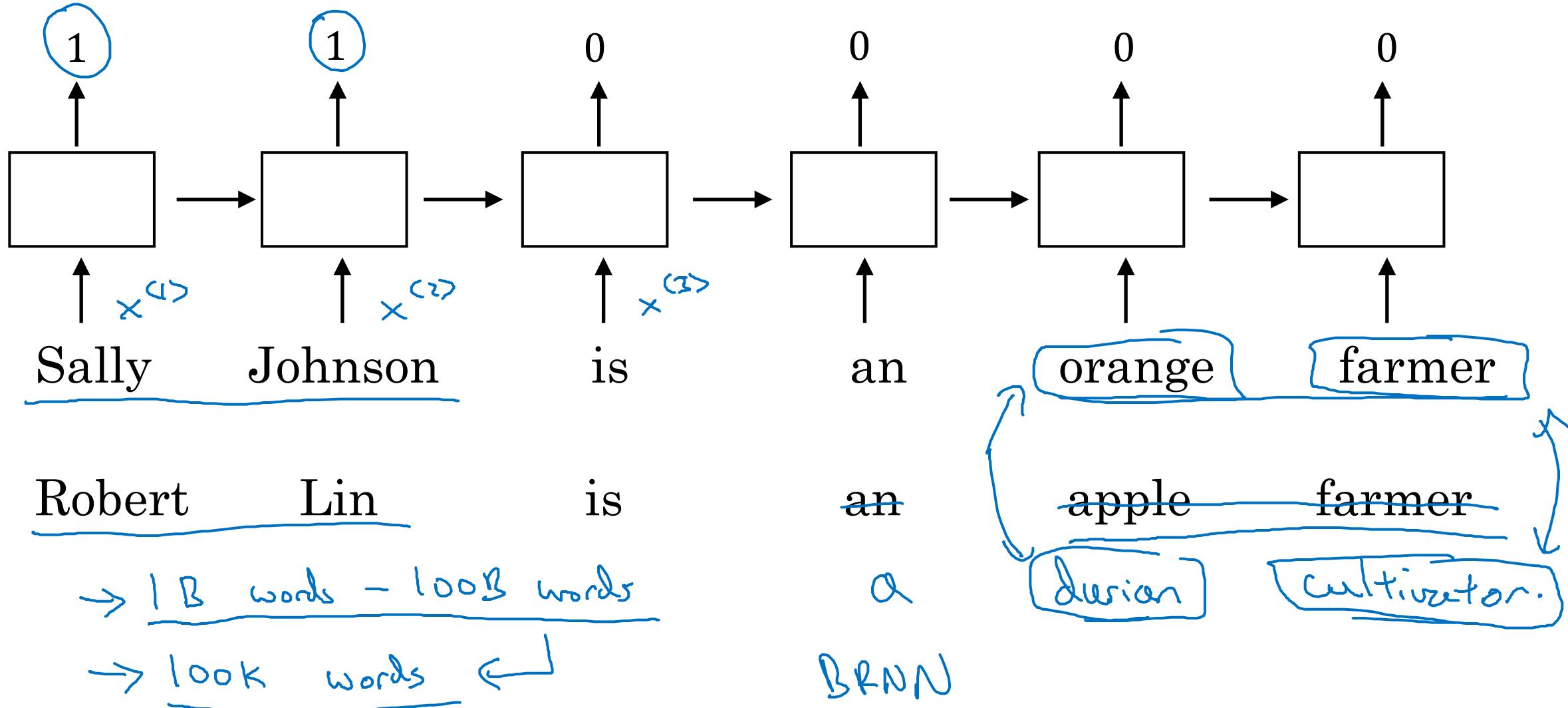


deeplearning.ai

NLP and Word Embeddings

Using word embeddings

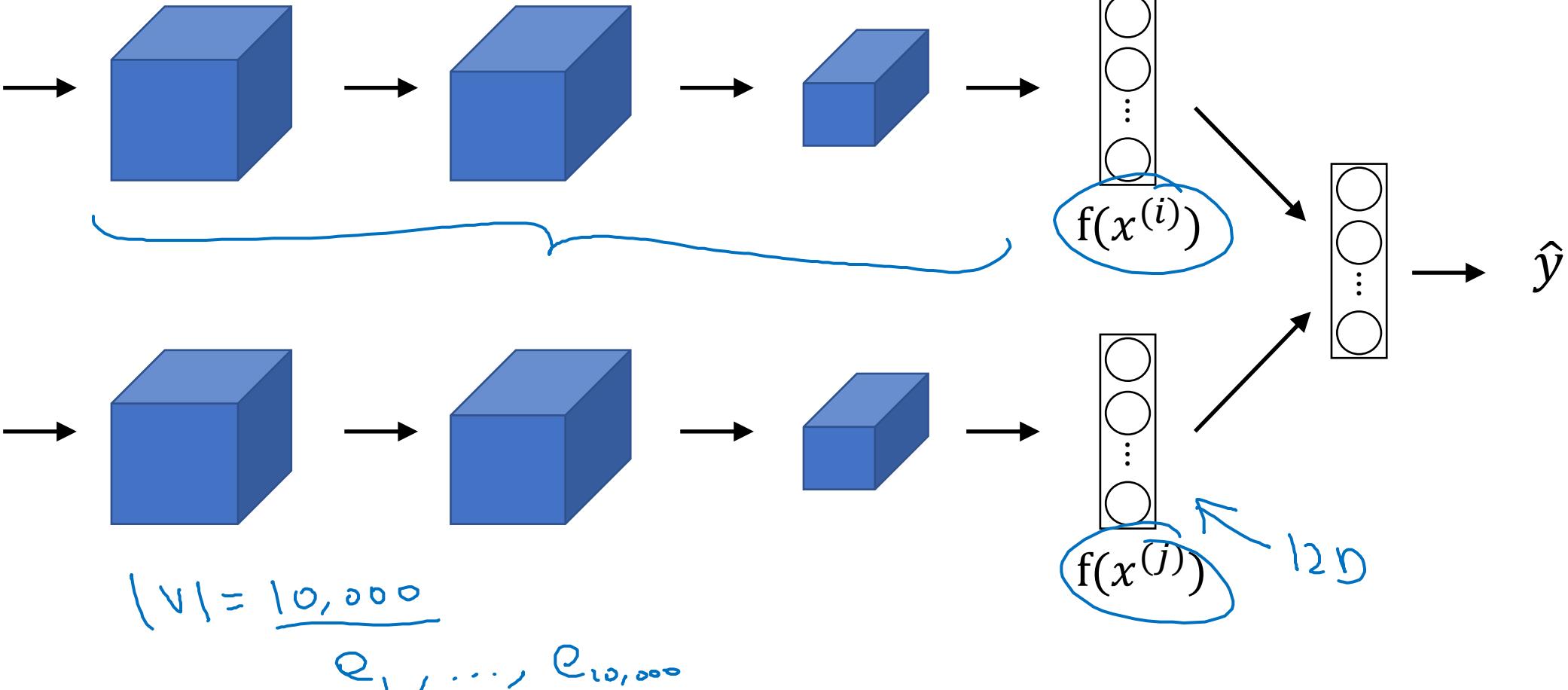
Named entity recognition example



Transfer learning and word embeddings

-
- The diagram consists of two blue-outlined brackets. The top bracket, labeled 'A' in a circle, encloses the first two steps. The bottom bracket, labeled 'B' in a circle, encloses the third step. A vertical blue arrow points downwards from the end of the 'A' bracket to the start of the 'B' bracket.
1. Learn word embeddings from large text corpus. (1-100B words)
(Or download pre-trained embedding online.)
 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) 128D





deeplearning.ai

NLP and Word Embeddings

Properties of word embeddings

Analogy

	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

$$\begin{matrix} e_{5391} \\ e_{\text{man}} \end{matrix}$$

$$\underline{\text{Man} \rightarrow \text{Woman}}$$

$$e_{\text{man}} - e_{\text{woman}}$$

$$e_{\text{woman}}$$

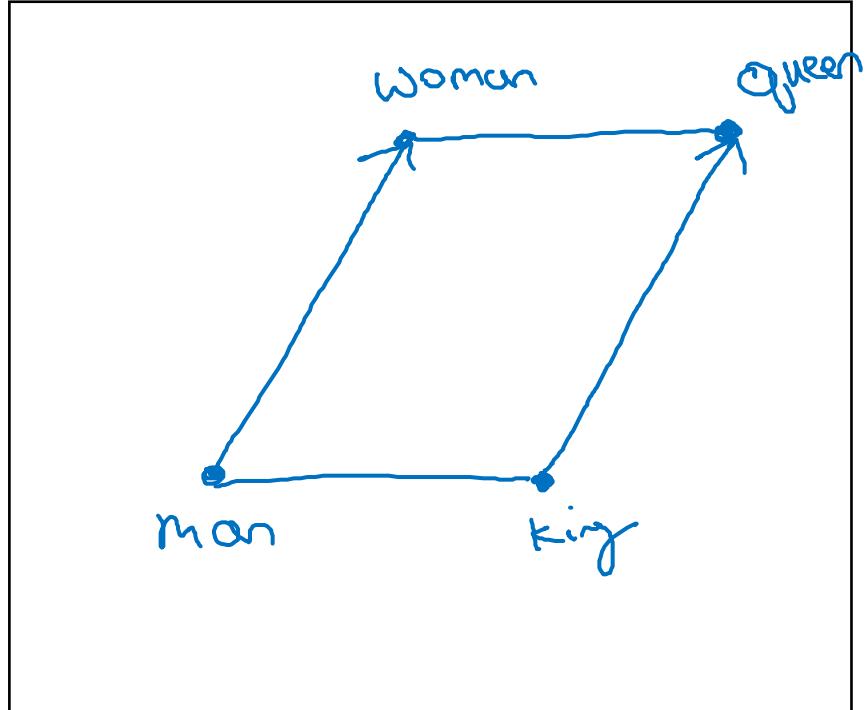
$$\underline{\text{King} \rightarrow ? \text{ Queen}}$$

$$e_{\text{king}} - e_{? \text{ Queen}}$$

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

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

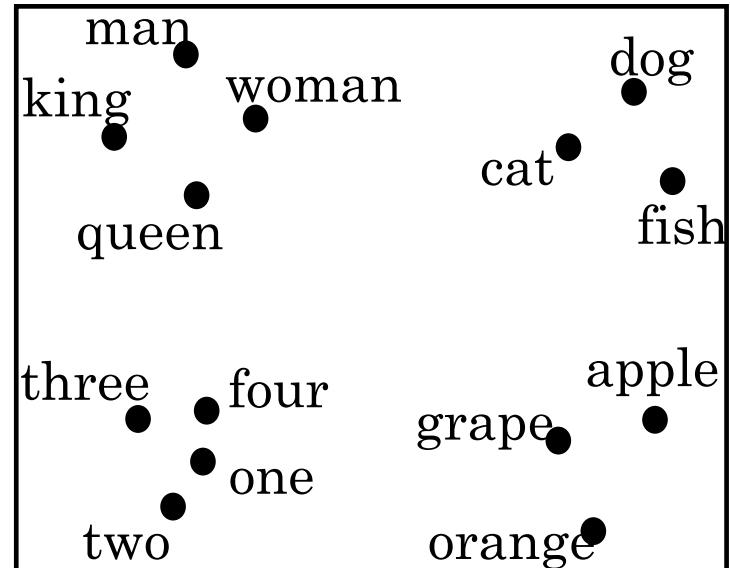
Analogies using word vectors



300 D

Find word $w_i: \arg \max_w$

300D \rightarrow 2D
↑



t-SNE

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

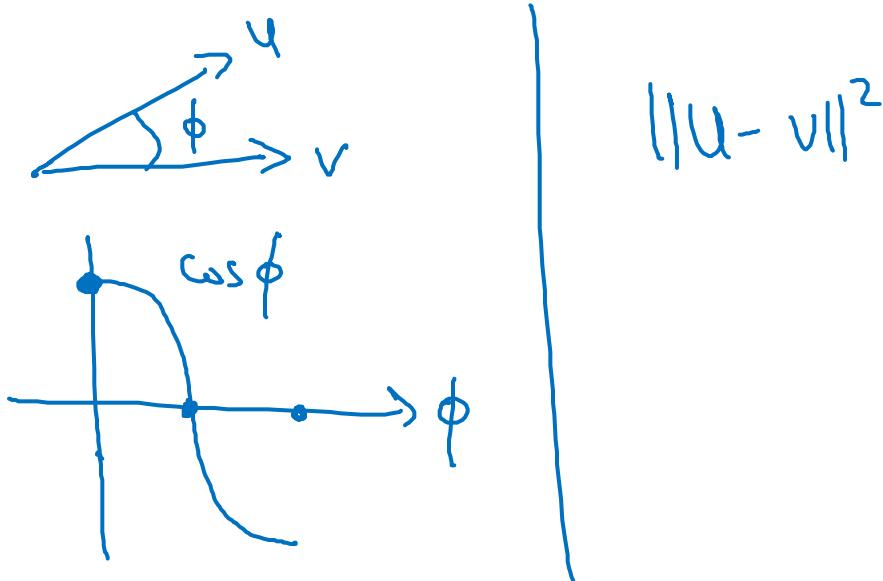
Sim(e_w , $e_{\text{king}} - e_{\text{man}} + e_{\text{woman}}$)

30 - 75%

Cosine similarity

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

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



Man:Woman as Boy:Girl

Ottawa:Canada as Nairobi:Kenya

Big:Bigger as Tall:Taller

Yen:Japan as Ruble:Russia

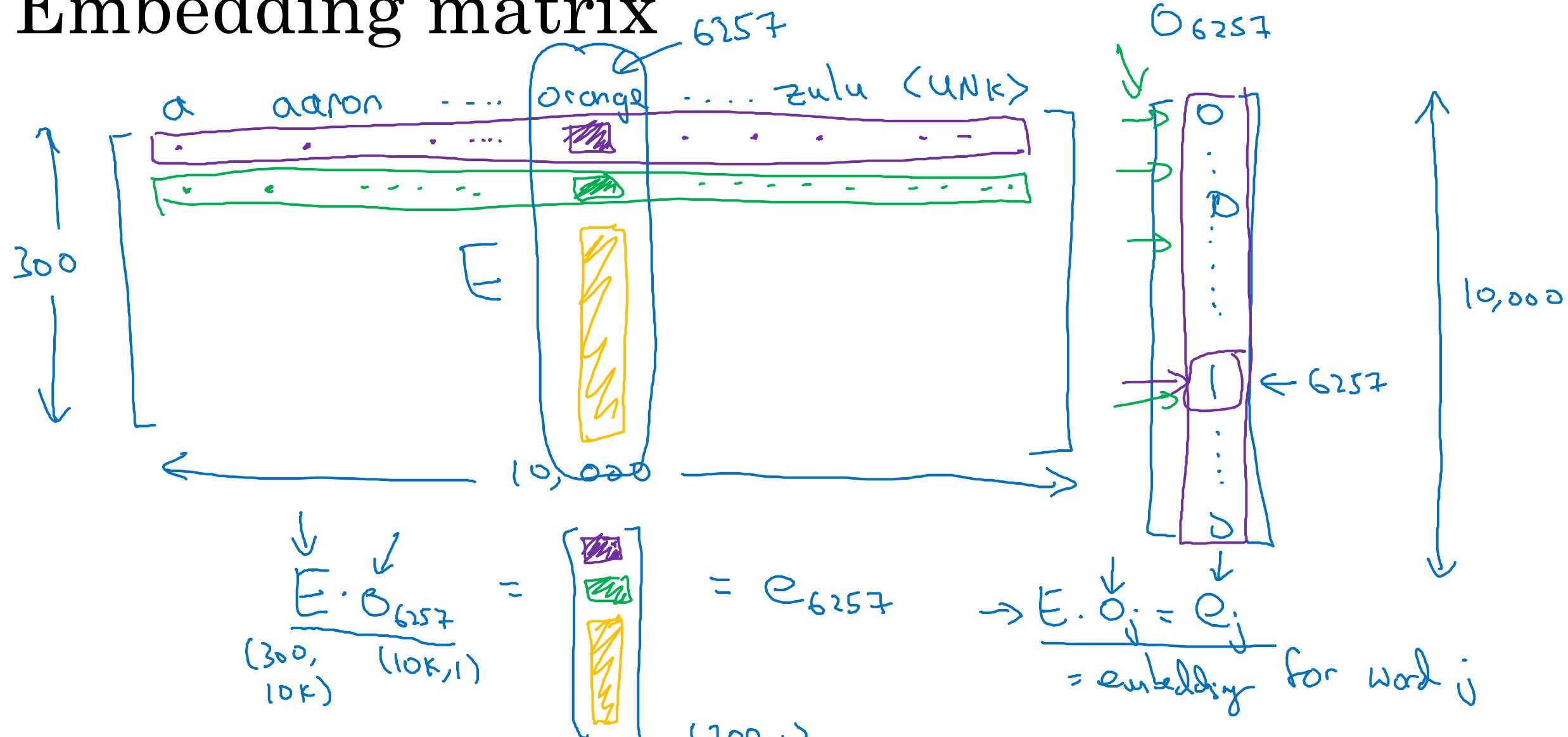


deeplearning.ai

NLP and Word Embeddings

Embedding matrix

Embedding matrix



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

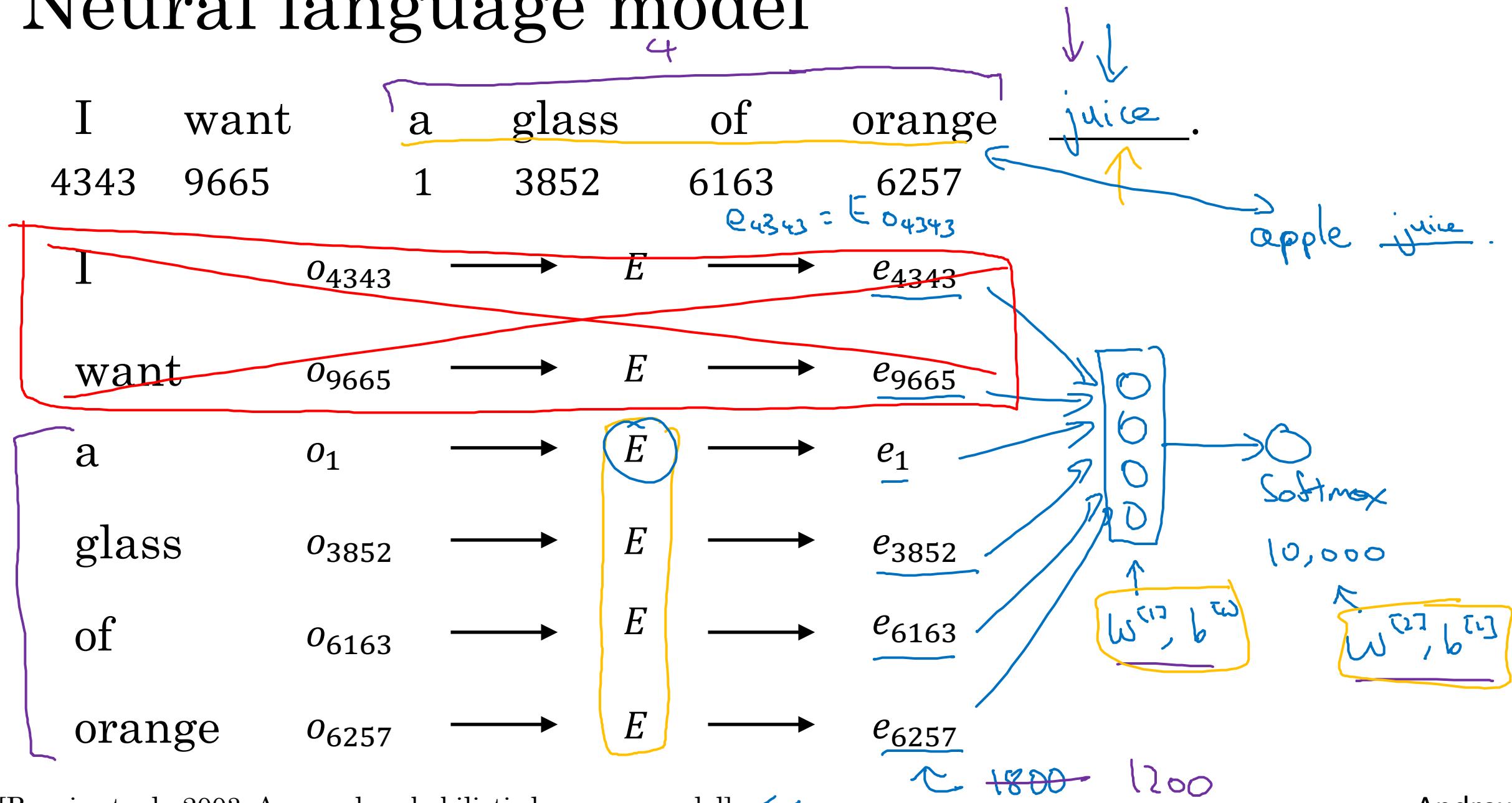


deeplearning.ai

NLP and Word Embeddings

Learning word embeddings

Neural language model



Other context/target pairs

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

Context: Last 4 words.

4 words on left & right

Last 1 word

Nearby 1 word

skip gram

a glass of orange ? to go along with

orange ?

glass . ?



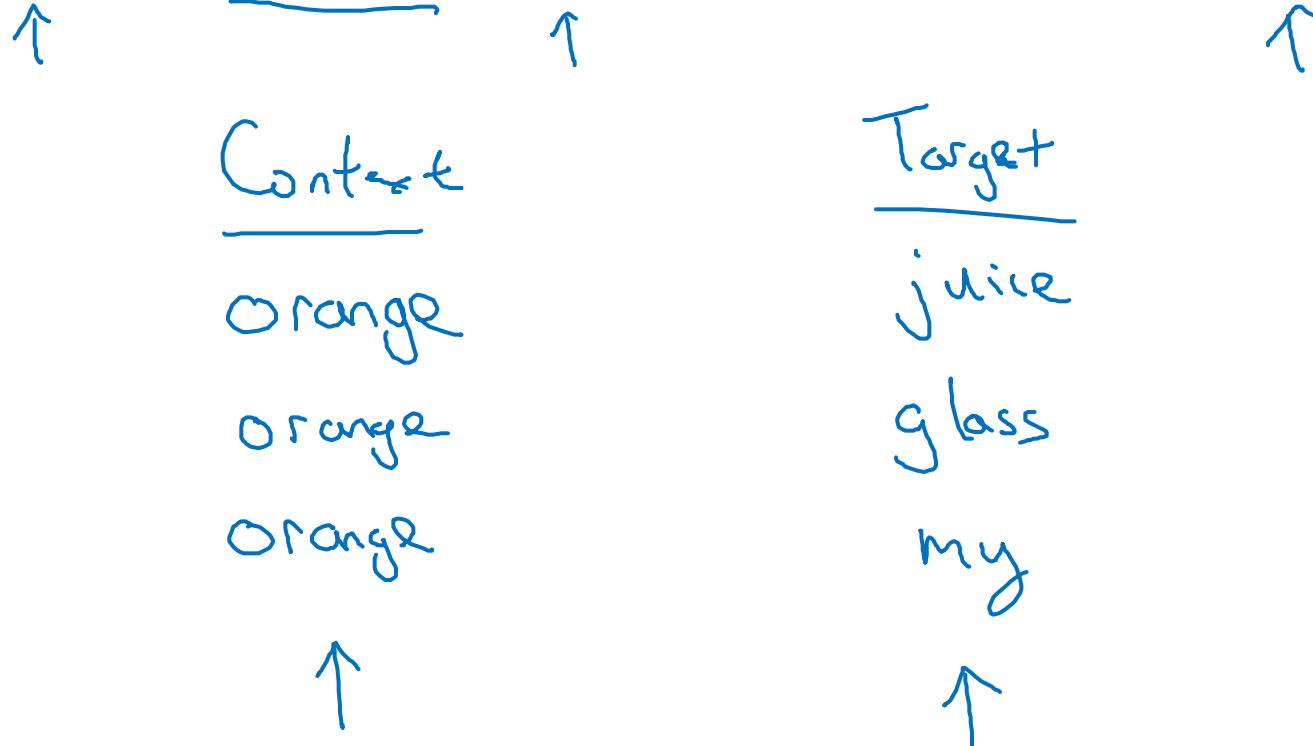
deeplearning.ai

NLP and Word Embeddings

Word2Vec

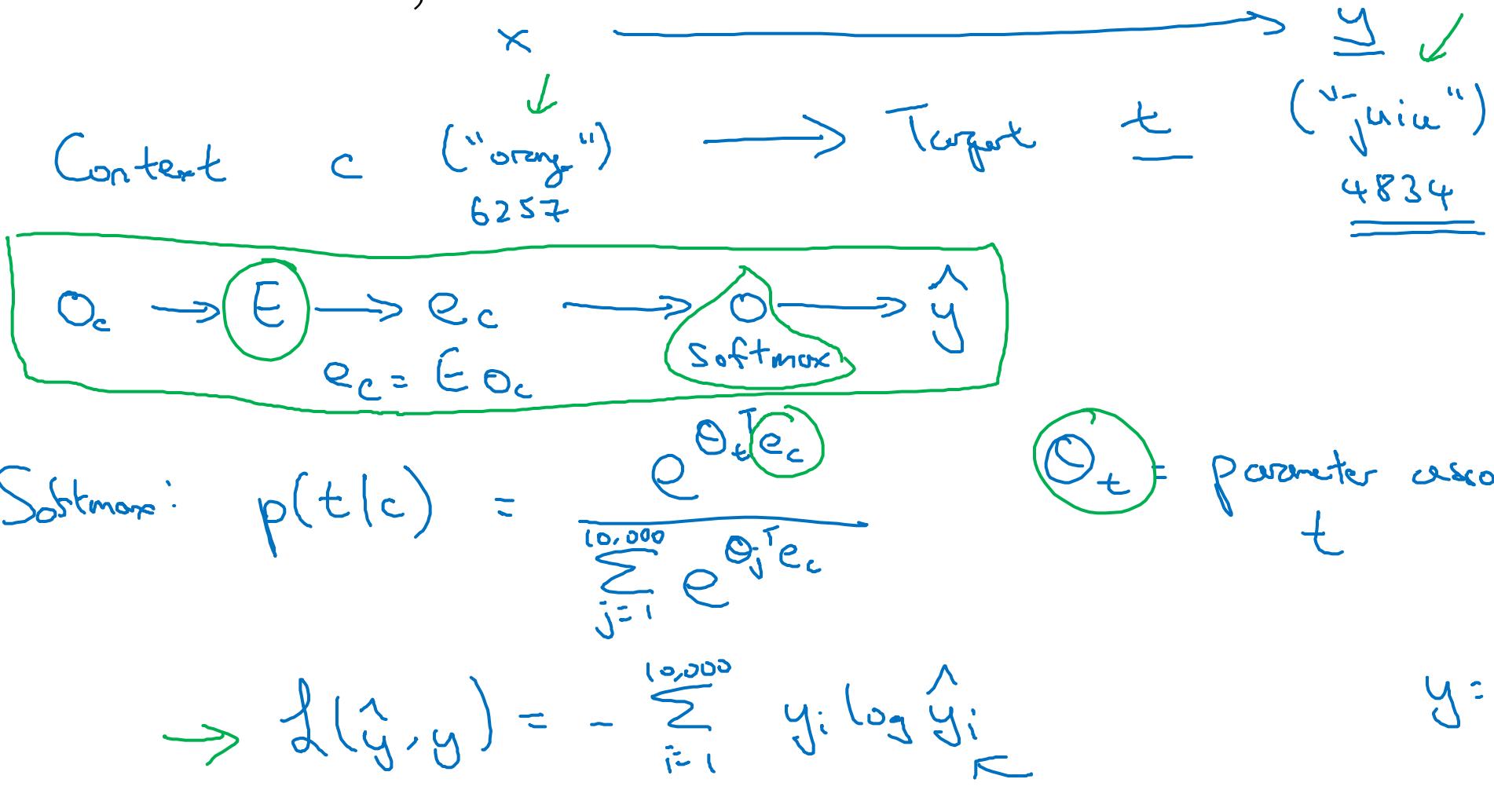
Skip-grams

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



Model

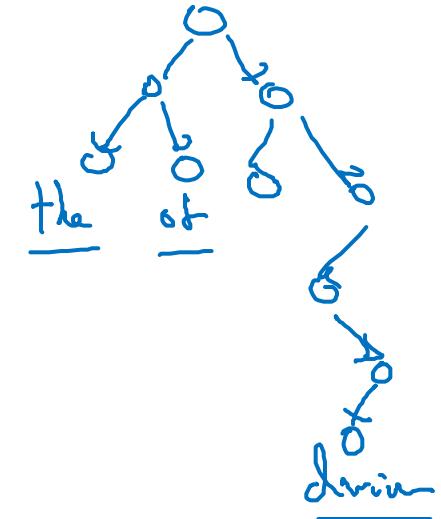
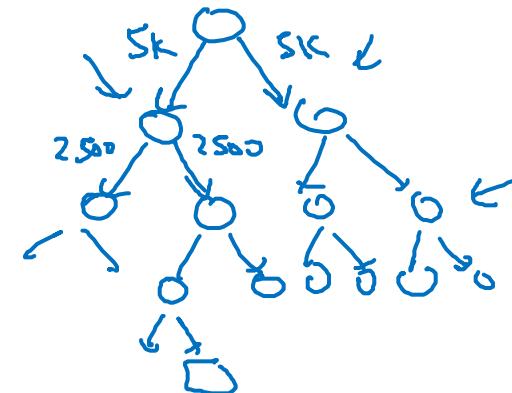
Vocab size = 10,000k



Problems with softmax classification

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

Hierarchical softmax.



How to sample the context c ?

→ the, of, a, and, to, ...

→ orange, apple, durian

P_{durian}

t
 $c \rightarrow t$
 $P(c)$



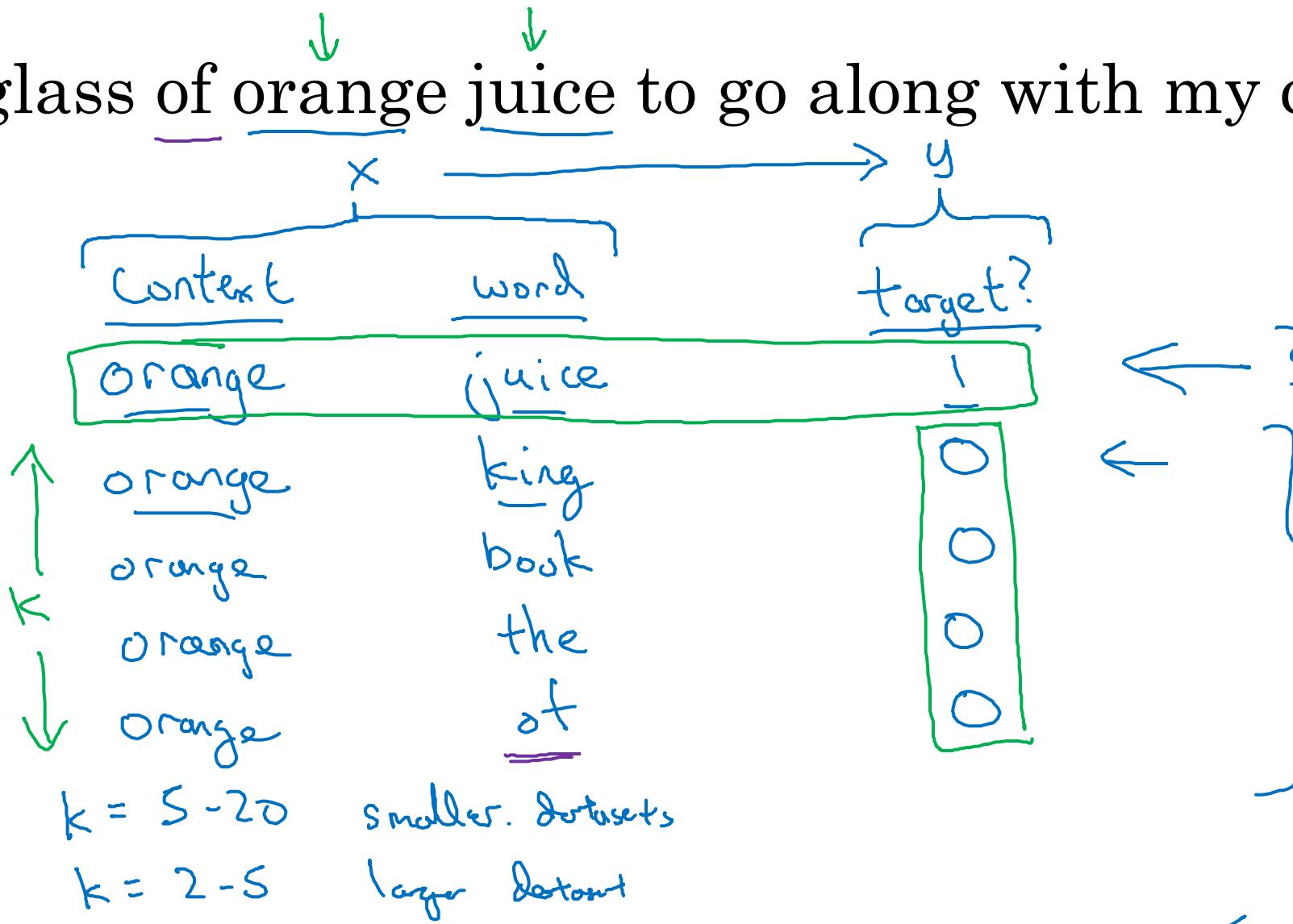
deeplearning.ai

NLP and Word Embeddings

Negative sampling

Defining a new learning problem

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



Model

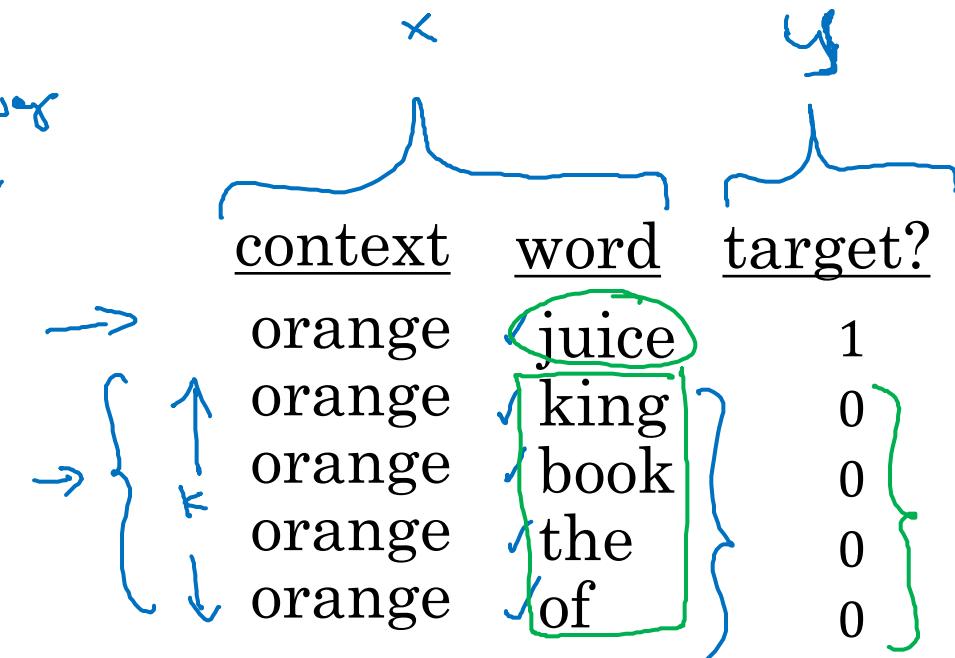
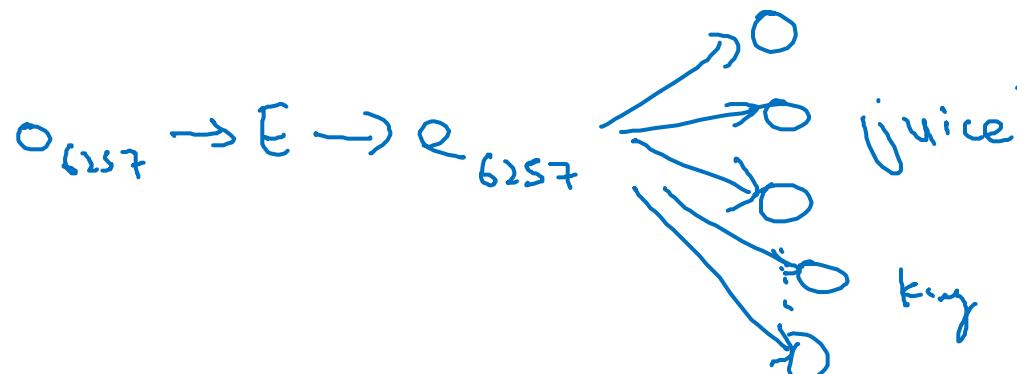
Softmax:

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

10,000-way softmax

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

Orange
6257



10,000
↓

10,000 binary
classification
problem
 $k+1$

Andrew Ng

Selecting negative examples

<u>context</u>	<u>word</u>	<u>target?</u>
orange	juice	1
orange	king	0
orange	book	0
orange	the	0
orange	of	0

the , of, and, ...

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

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



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

$x_{i,j} = \# \text{ times } i \text{ appears in context of } j.$

$x_{i,j}$ i j
↑ ↑ ↑
c t c

$$x_{ij} = x_{ji} \leftarrow$$



Model

Minimize

$$\sum_{i=1}^{10,000} \sum_{j=1}^{10,000} f(x_{ij}) (\mathbf{o}_i^T \mathbf{e}_j + b_i + b_j' - \log \underline{x}_{ij})^2$$

" $\mathbf{o}_i^T \mathbf{e}_c$ "

weight_{ij}
term

$f(x_{ij}) = 0$ or $x_{ij} = 0$. "0 log 0" = 0

this, is, of, a, ...

derian

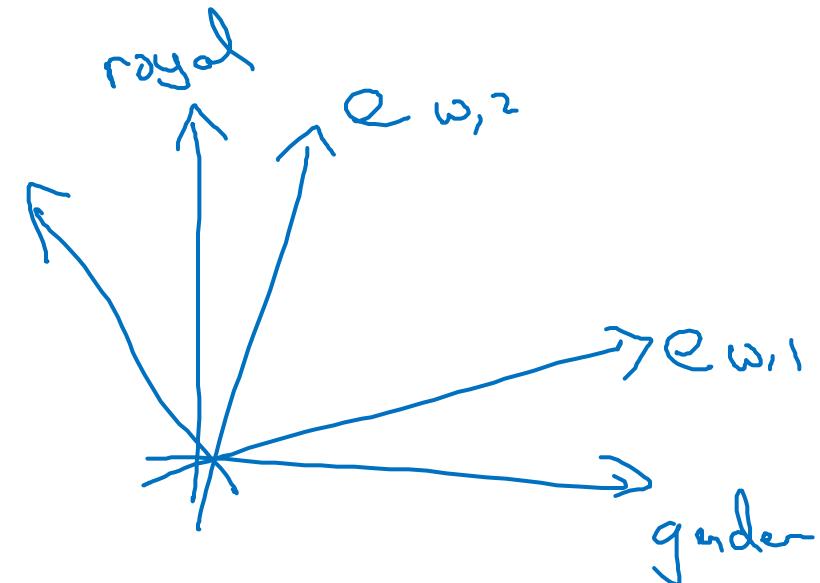
$\mathbf{o}_i, \mathbf{e}_j$ are symmetric

$\mathbf{e}_w^{(\text{final})} = \frac{\mathbf{e}_w + \mathbf{o}_w}{\sqrt{2}}$

Andrew Ng

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



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

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



deeplearning.ai

NLP and Word Embeddings

Sentiment classification

Sentiment classification problem



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 \rightarrow 100,000 words

Simple sentiment classification model

The dessert is excellent
8928 2468 4694 3180



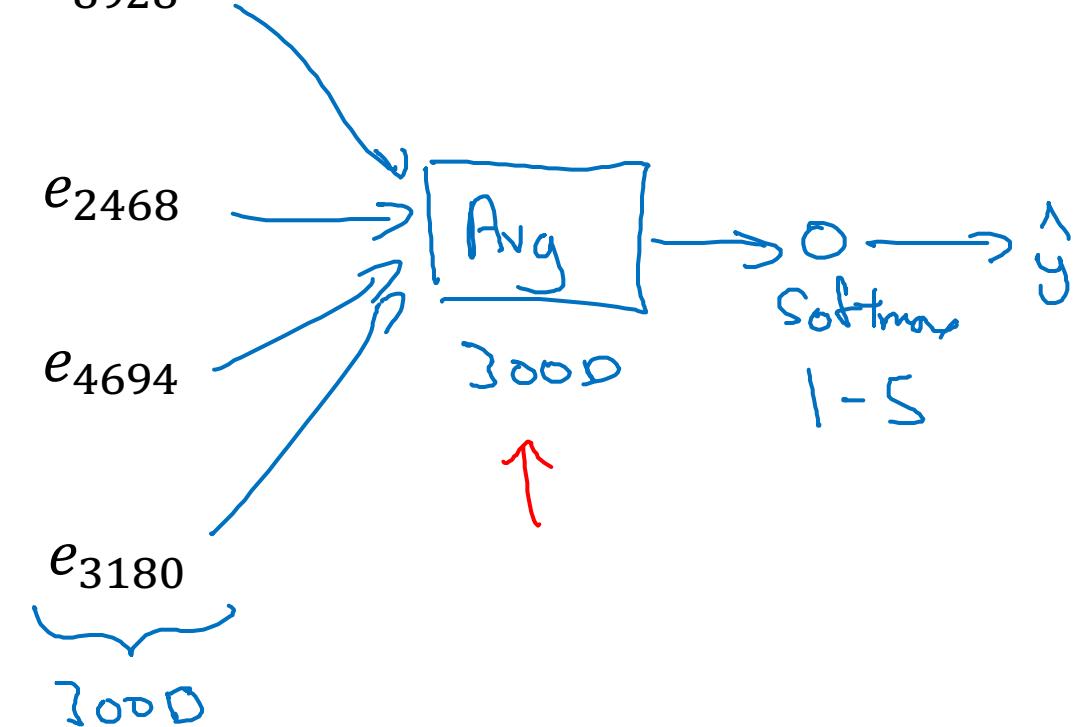
The o_{8928} $\rightarrow E \rightarrow e_{8928}$

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

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

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

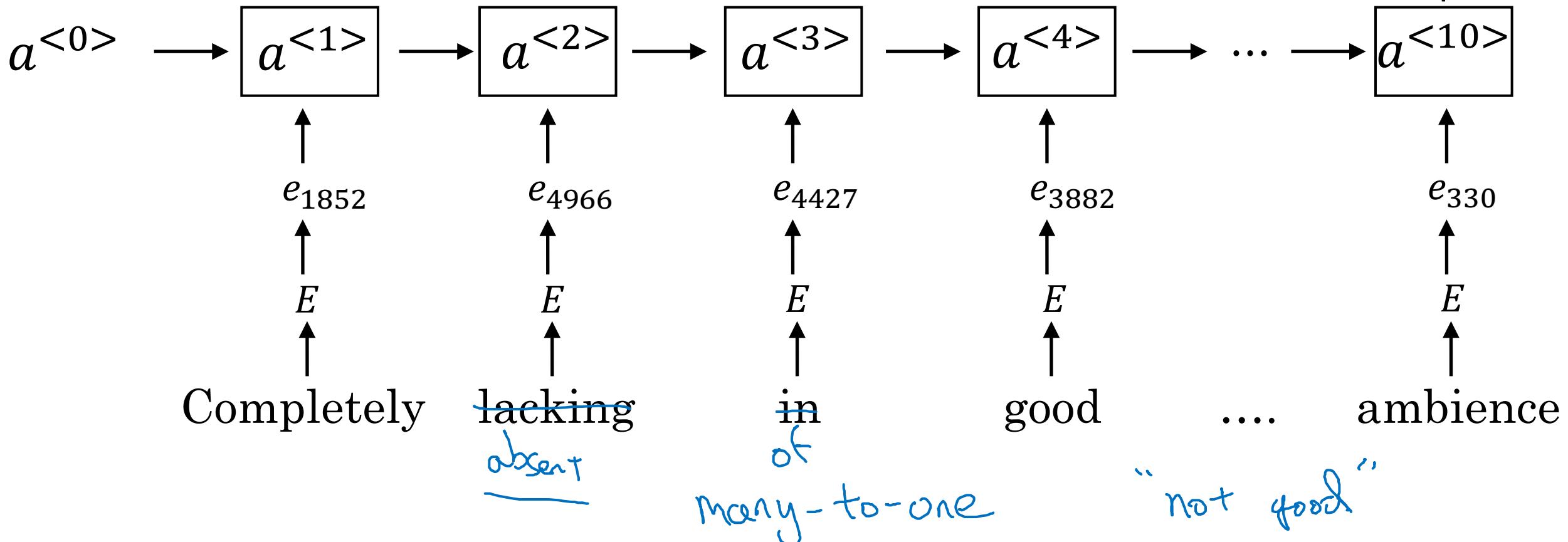
“Completely lacking in good taste, good service, and good ambience.”
↑
100 B words



RNN for sentiment classification

\hat{y}

softmax





deeplearning.ai

NLP and Word Embeddings

Debiasing word embeddings

The problem of bias in word embeddings

Man:Woman as King:Queen

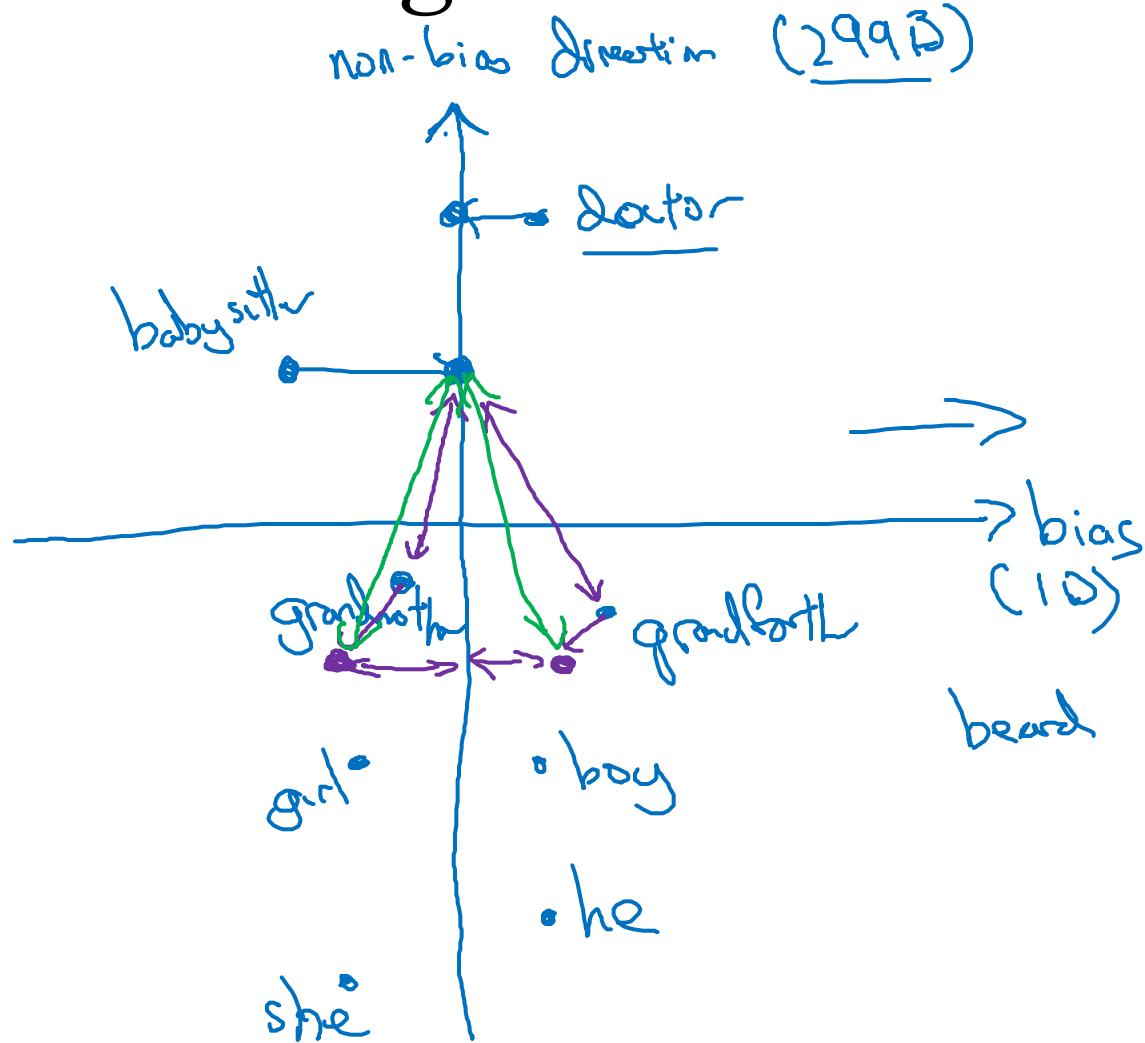
Man:Computer_Programmer as Woman:Homemaker 

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.



Addressing bias in word embeddings



1. Identify bias direction.

$$\left\{ \begin{array}{l} e_{\text{he}} - e_{\text{she}} \\ e_{\text{male}} - e_{\text{female}} \\ \vdots \\ \text{average} \end{array} \right.$$

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

3. Equalize pairs.

$$\rightarrow \text{grandmother} - \text{grandfather} = \text{girl} - \text{boy}$$

Copyright Notice

These slides are distributed under the Creative Commons License.

[DeepLearning.AI](#) makes these slides available for educational purposes. You may not use or distribute these slides for commercial purposes. You may make copies of these slides and use or distribute them for educational purposes as long as you cite [DeepLearning.AI](#) as the source of the slides.

For the rest of the details of the license, see <https://creativecommons.org/licenses/by-sa/2.0/legalcode>



deeplearning.ai

Sequence to sequence models

Basic models

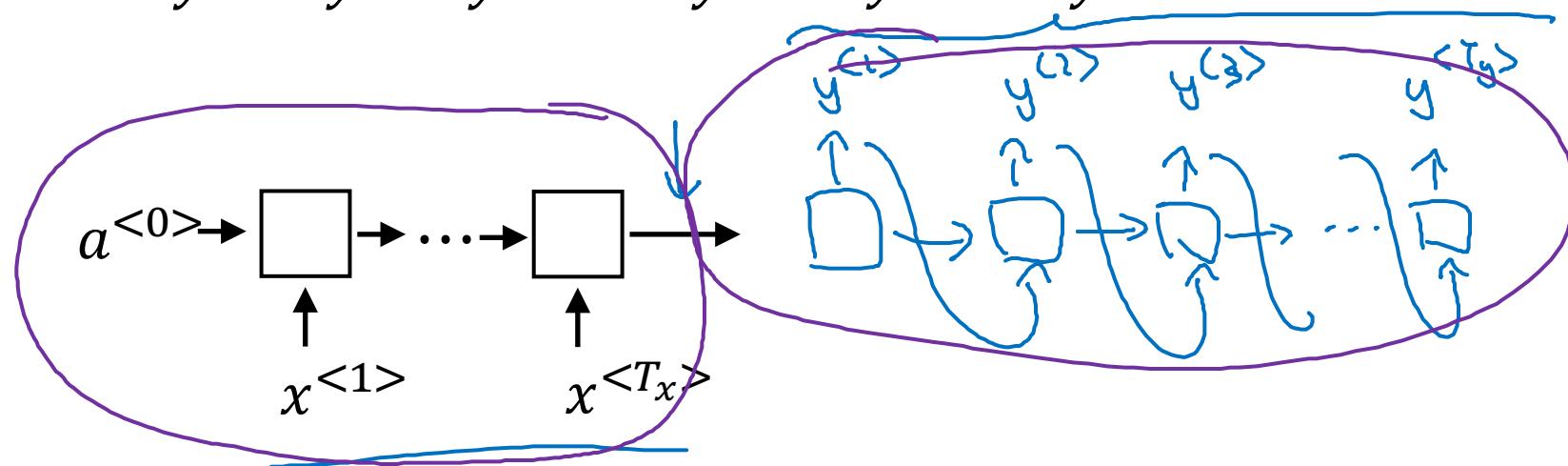
Sequence to sequence model

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

Jane visite l'Afrique en septembre

→ Jane is visiting Africa in September.

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

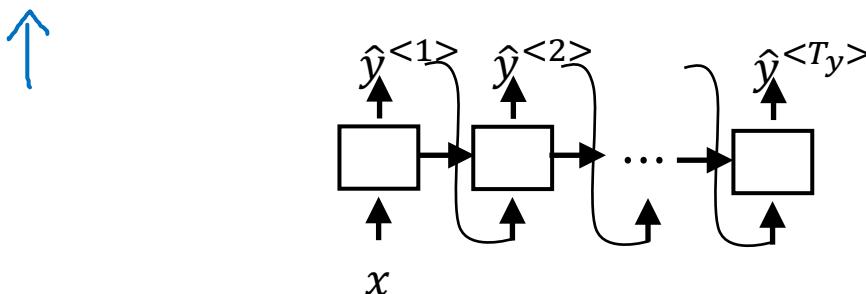
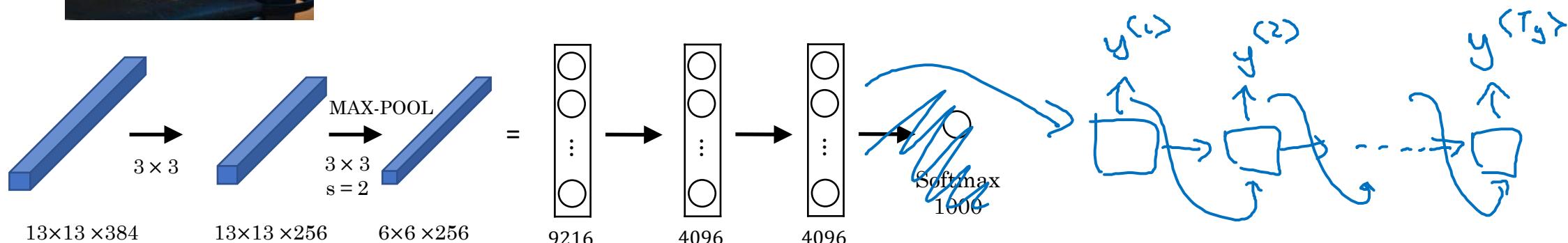
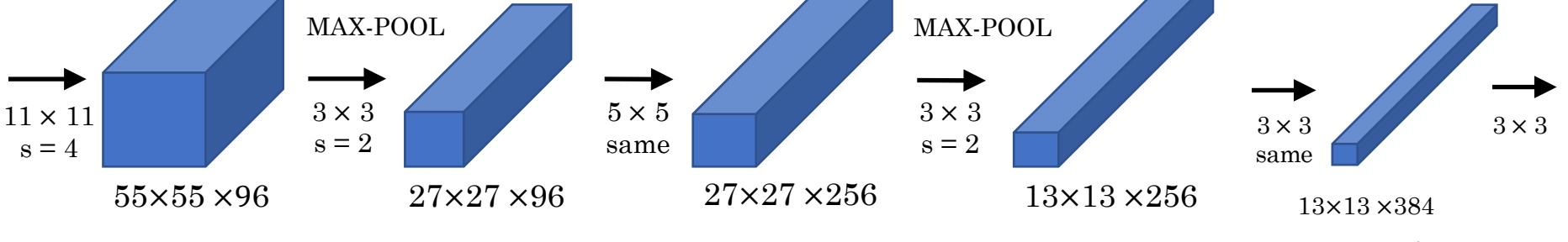
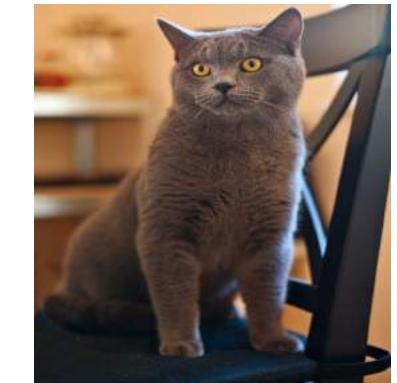


[Sutskever et al., 2014. Sequence to sequence learning with neural networks] ↪

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

Andrew Ng

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]

Andrew Ng



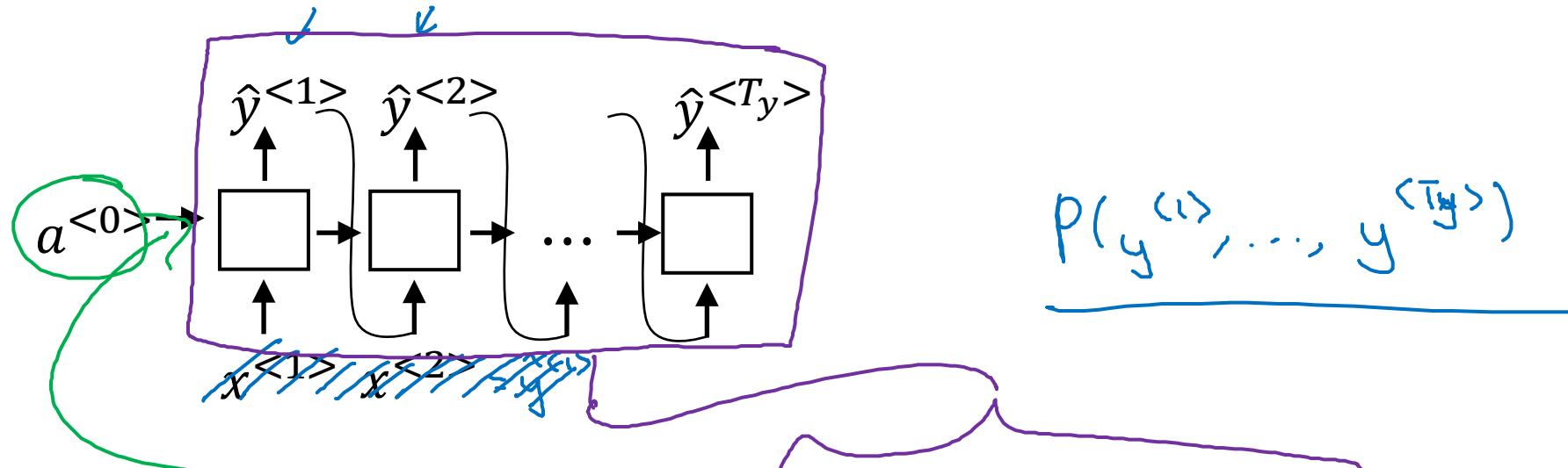
deeplearning.ai

Sequence to sequence models

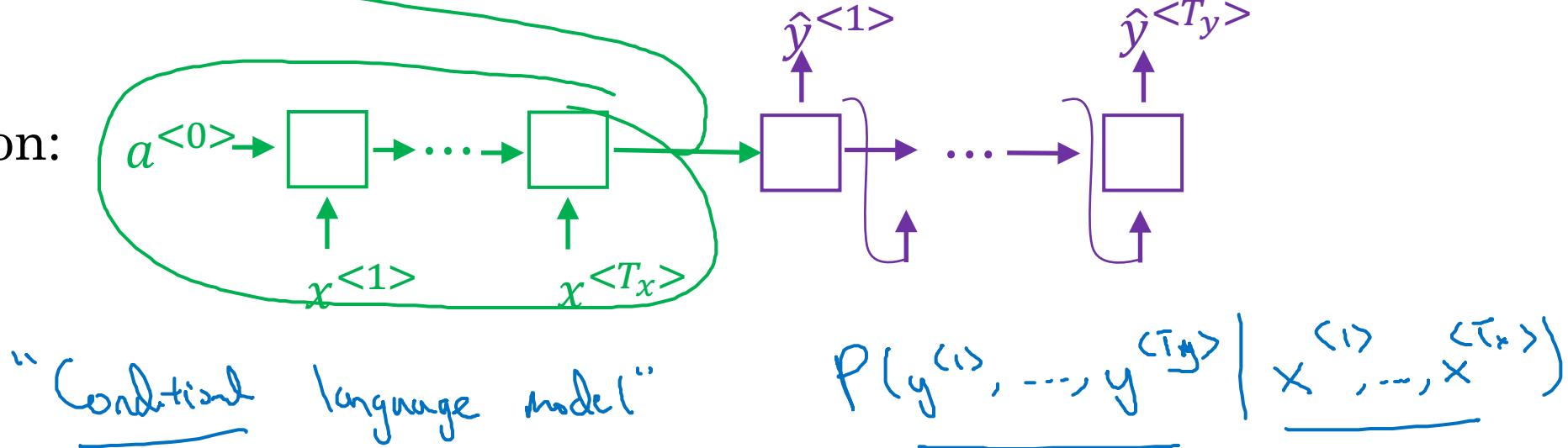
Picking the most
likely sentence

Machine translation as building a conditional language model

Language model:



Machine translation:



Andrew Ng

Finding the most likely translation

Jane visite l'Afrique en septembre.

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

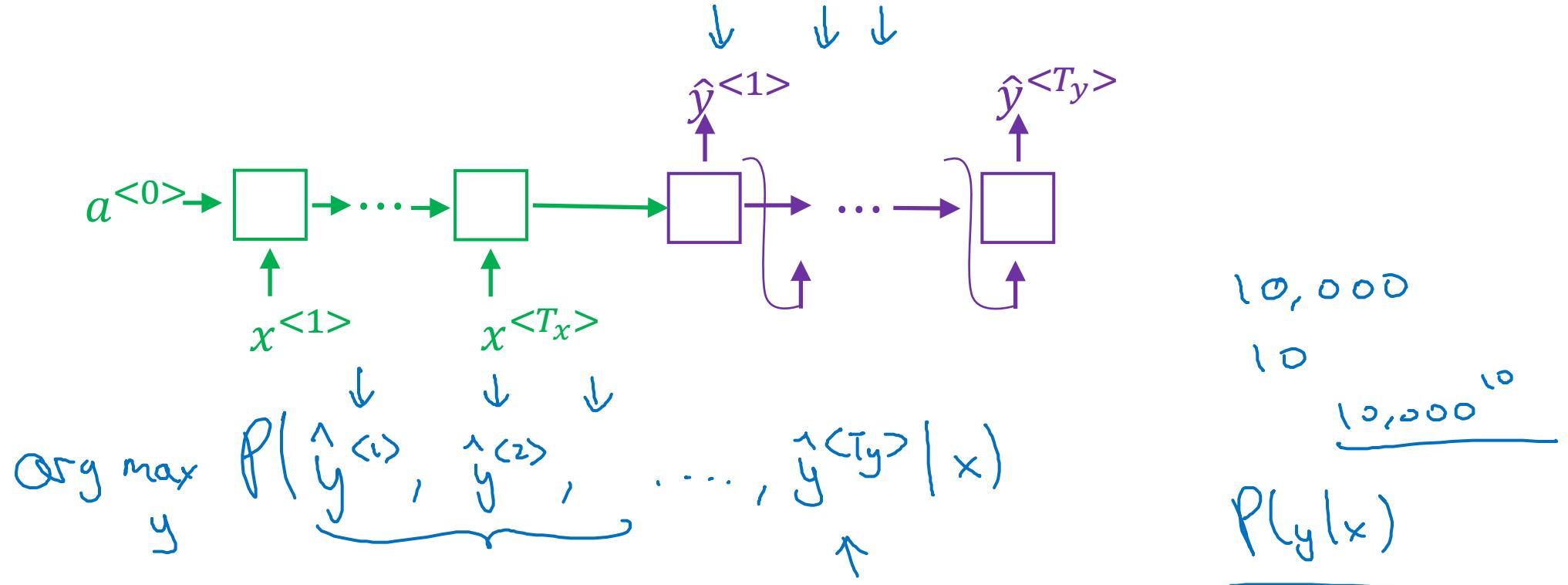
French
↓
English

- 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.

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

Why not a greedy search?

$$P(\hat{y}^{(1)} | x)$$



$$\begin{aligned} & 10,000 \\ & 10 \\ & \underline{\frac{10,000}{10}} \\ & \underline{P(y|x)} \end{aligned}$$

→ Jane is visiting Africa in September.

→ Jane is going to be visiting Africa in September.

$$P(\text{Jane is going } | x) > P(\text{Jane is visiting } | x)$$



deeplearning.ai

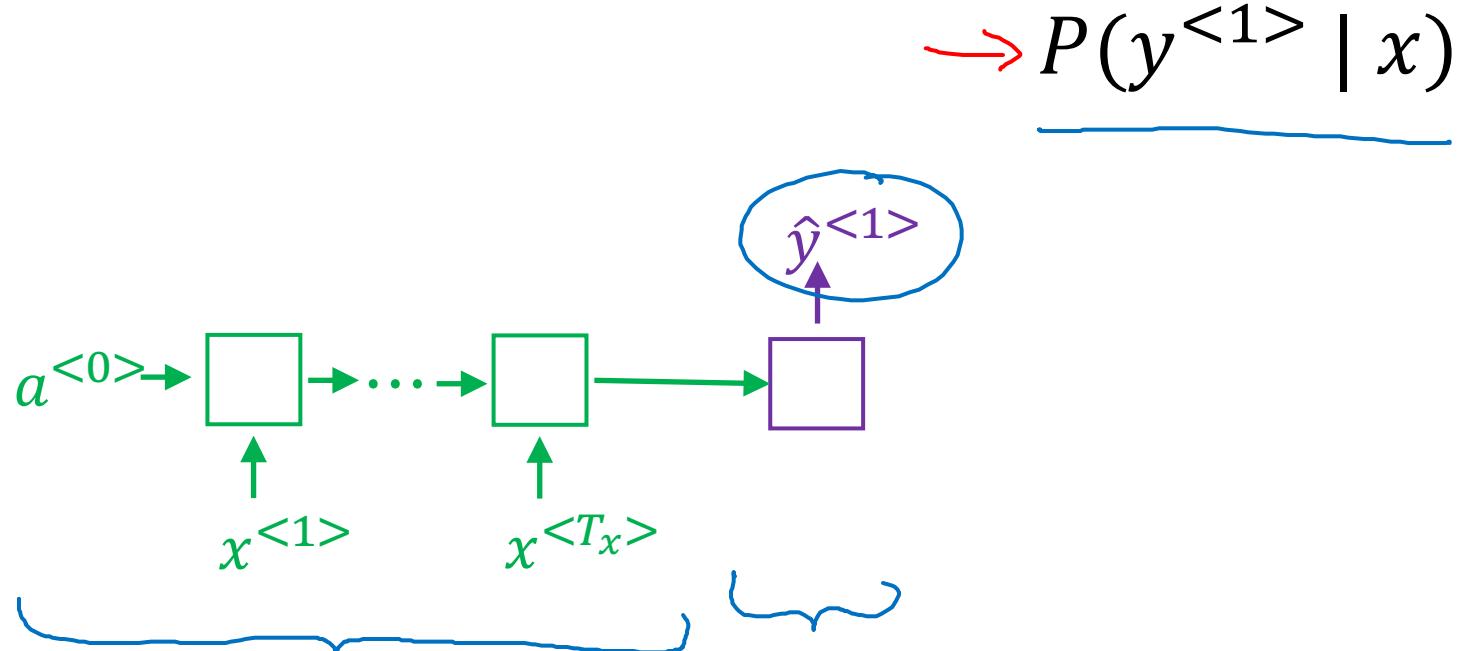
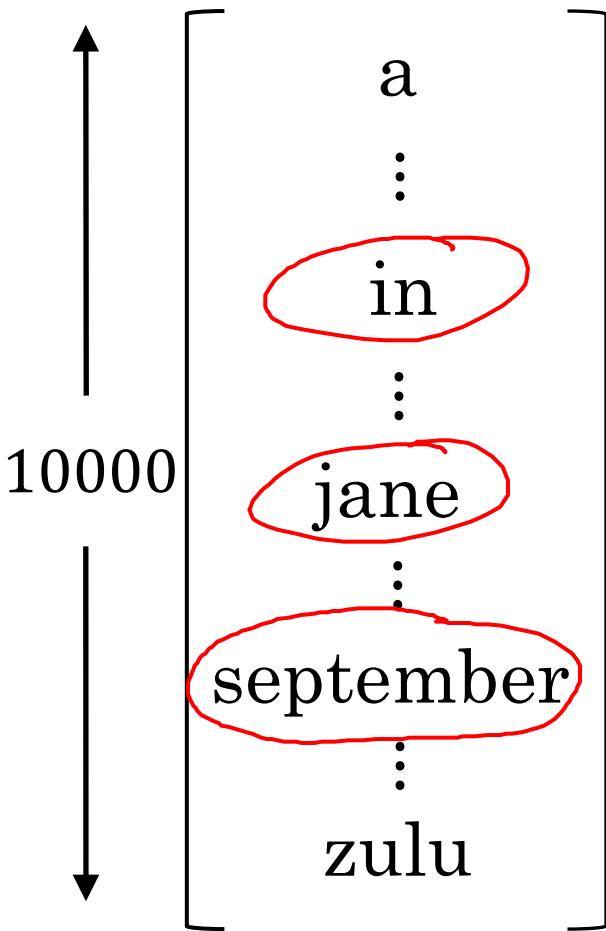
Sequence to sequence models

Beam search

Beam search algorithm

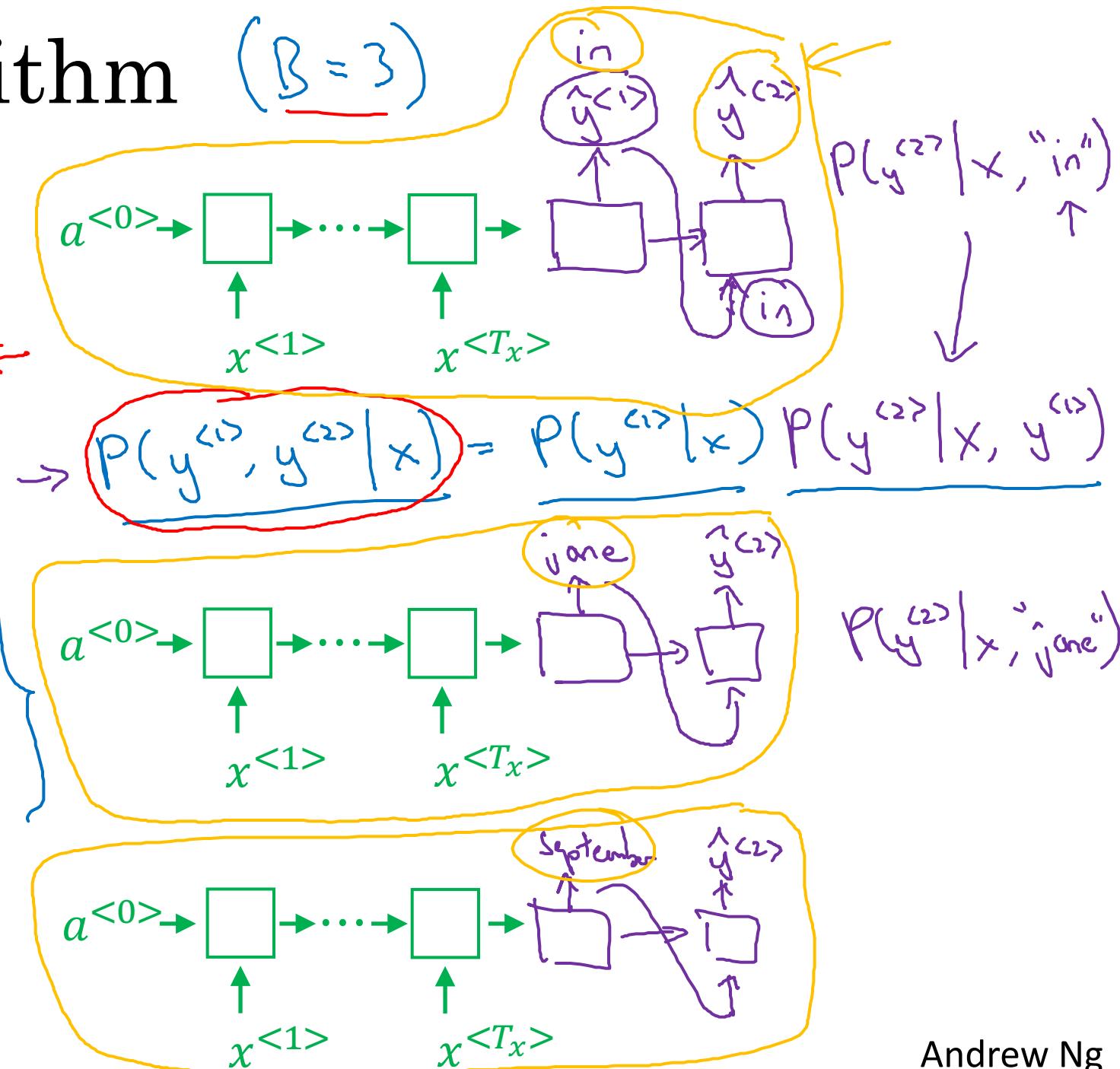
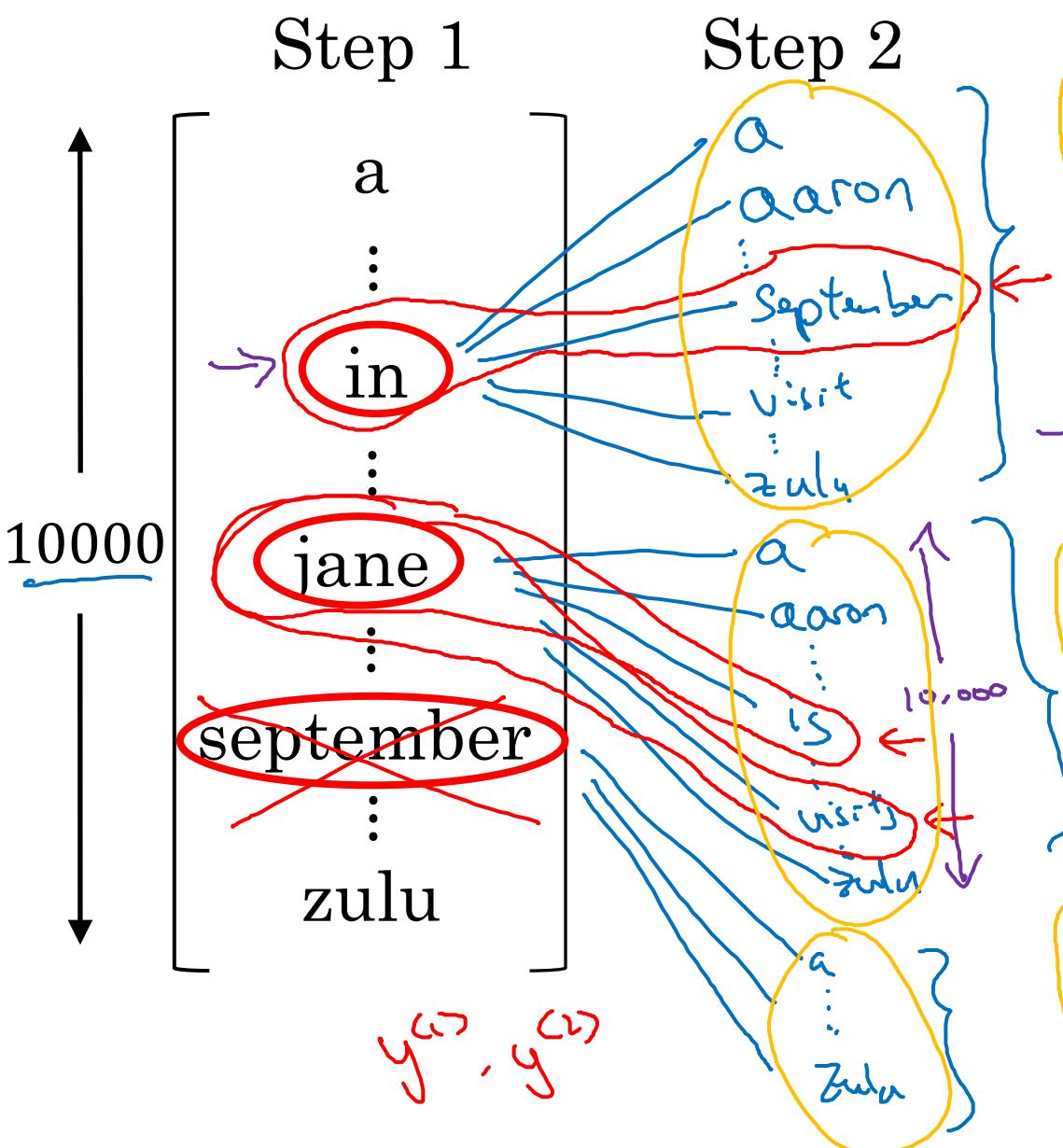
B = 3 (beam width)

Step 1



Beam search algorithm

$$(B = 3)$$



Beam search ($B = 3$)

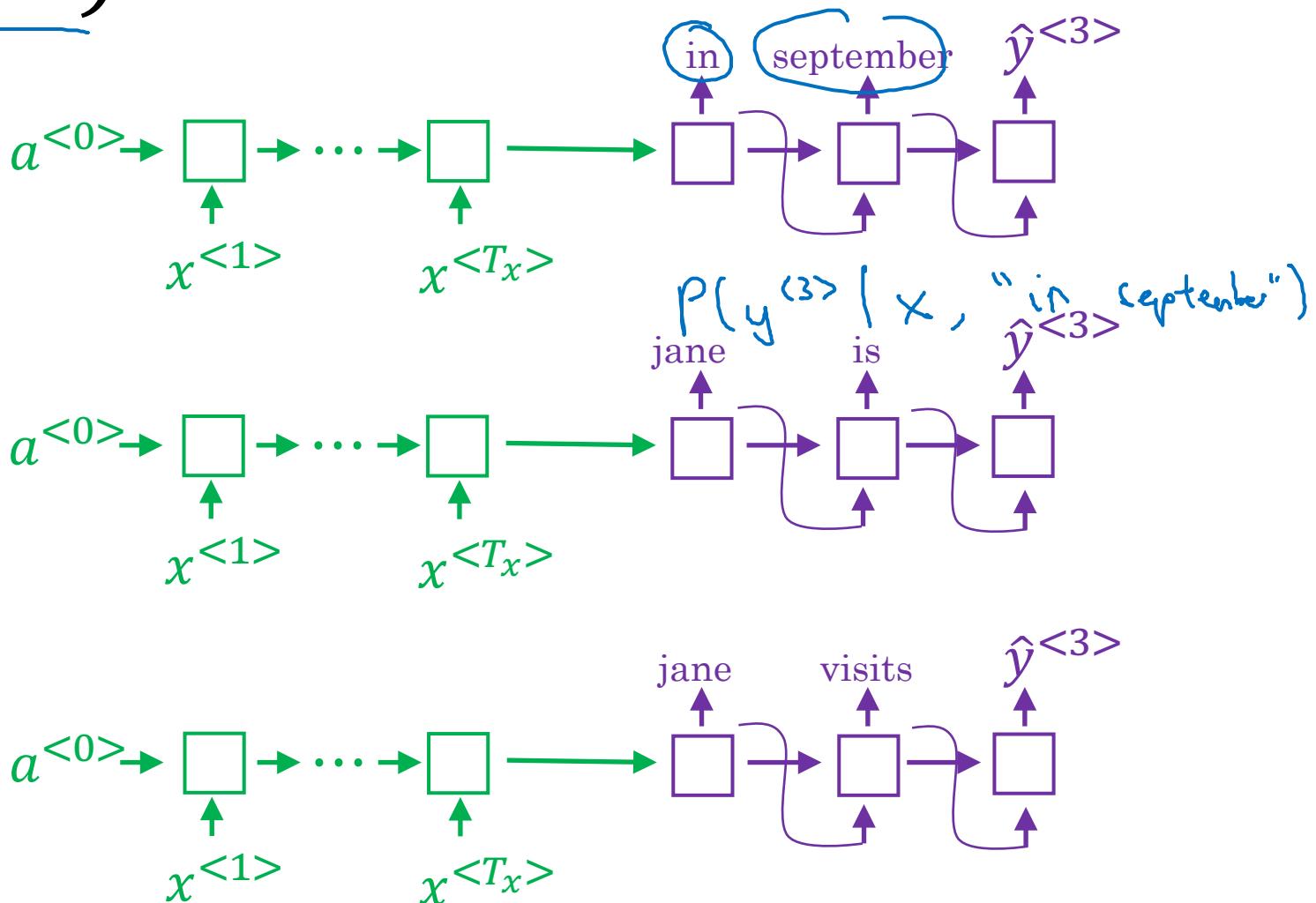
in september

jane is

jane visits

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

$B=1 \rightsquigarrow$ greedy search



jane visits africa in september. <EOS>



deeplearning.ai

Sequence to sequence models

Refinements to beam search

Length normalization

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

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

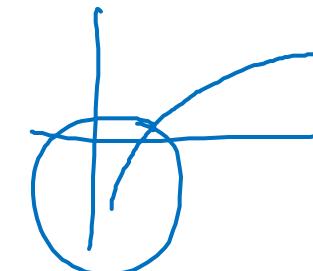
$$\arg \max_y \sum_{t=1}^{T_y} \log P(y^{<t>} | x, y^{<1>}, \dots, y^{<t-1>})$$

$\overleftarrow{Ty = 1, 2, 3, \dots, 30.}$

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

$$\alpha = 0.7$$

$$\begin{cases} d = 1 \\ d = 0 \end{cases}$$



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

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

Beam search discussion

Beam width B?

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

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

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 $\arg \max_y P(y|x)$.

y



deeplearning.ai

Sequence to sequence models

Error analysis on beam search

Example

Jane visite l'Afrique en septembre.

→ RNN

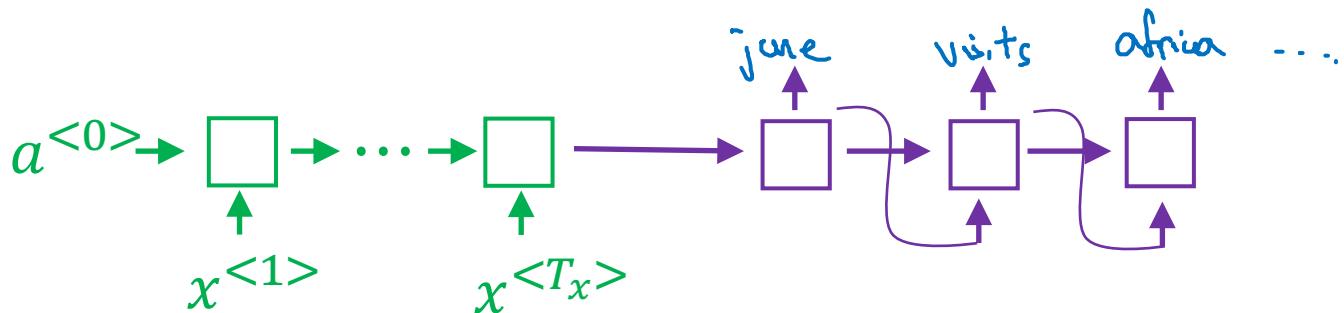
→ Beam Search

B↑

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

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

$$\text{RNN} \text{ computes } P(y^*|x) \geq P(\hat{y}|x)$$



Error analysis on beam search

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

$$P(y^*|x)$$

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

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

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

$$\arg \max_y P(y|x)$$

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

Conclusion: Beam search is at fault.

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

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

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.	<u>2×10^{-10}</u>	<u>1×10^{-10}</u>	B
...	...	—	—	R
...	...	—	—	R
			—	R
				:

Figures out what fraction 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.

Precision:

Modified precision:

Bleu
bilingual evaluation understudy

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 ←	

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

Bleu score on unigrams

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

$$P_1, P_2, = 1.0$$

Reference 2: There is a cat on the mat.

→ MT output: The cat the cat on the mat. (↑)

$$P_1 = \frac{\sum_{\text{unigrams} \in \hat{y}} \text{count}_{clip}(\text{unigram})}{\sum_{\text{unigrams} \in \hat{y}} \text{count}(\text{unigram})}$$

↑
Unigram

Count (unigram) Count (unigram)

count_{clip}(unigram) count_{clip}(unigram)

unigrams n-grams

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

↑
n-gram

Count (n-gram) Count (n-gram)

count_{clip}(n-gram) count_{clip}(n-gram)

n-grams n-grams

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

$$BP = \begin{cases} 1 & \text{if } \underline{\text{MT_output_length}} > \underline{\text{reference_output_length}} \\ \exp(1 - \text{MT_output_length}/\text{reference_output_length}) & \text{otherwise} \end{cases}$$



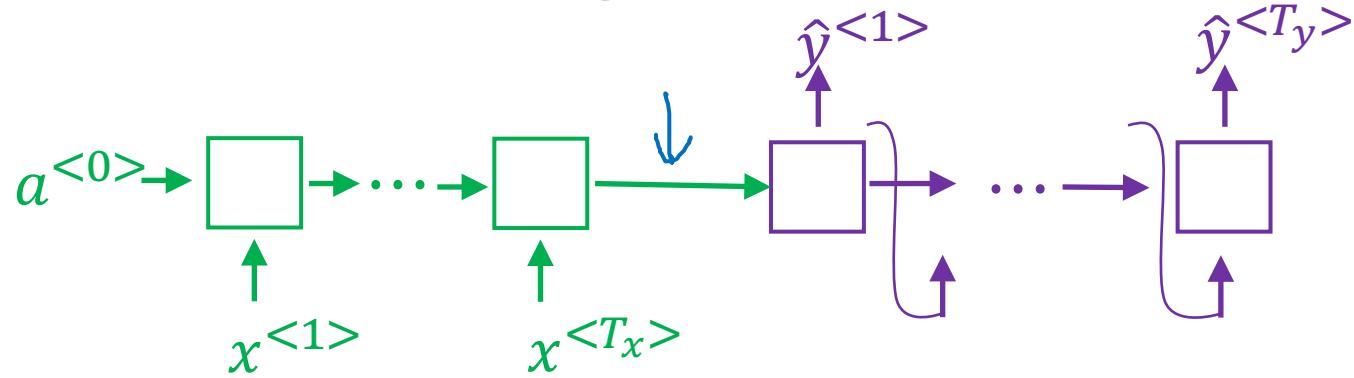


deeplearning.ai

Sequence to sequence models

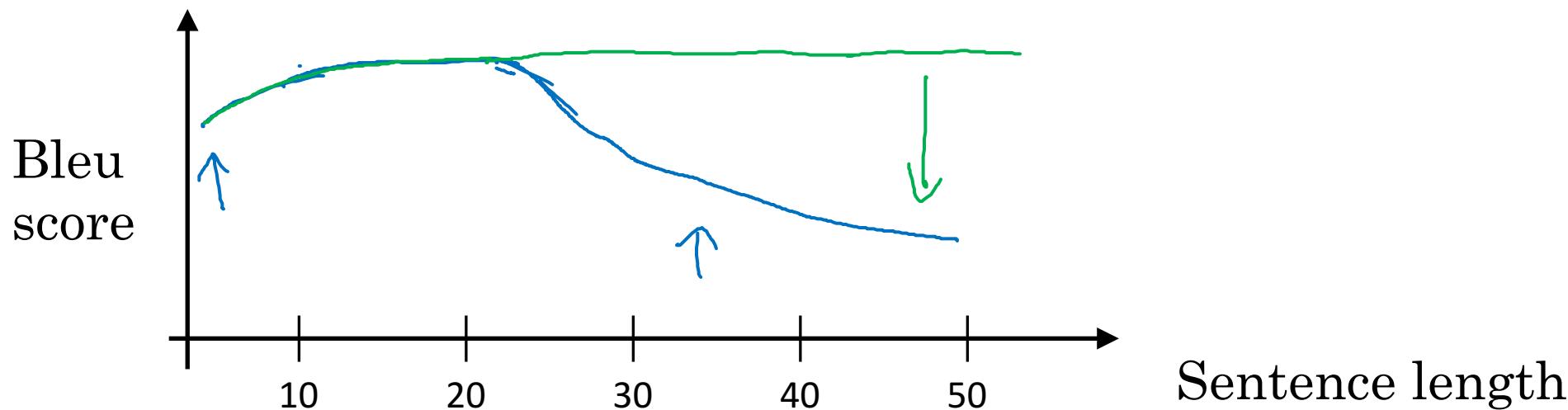
Attention model intuition

The problem of long sequences

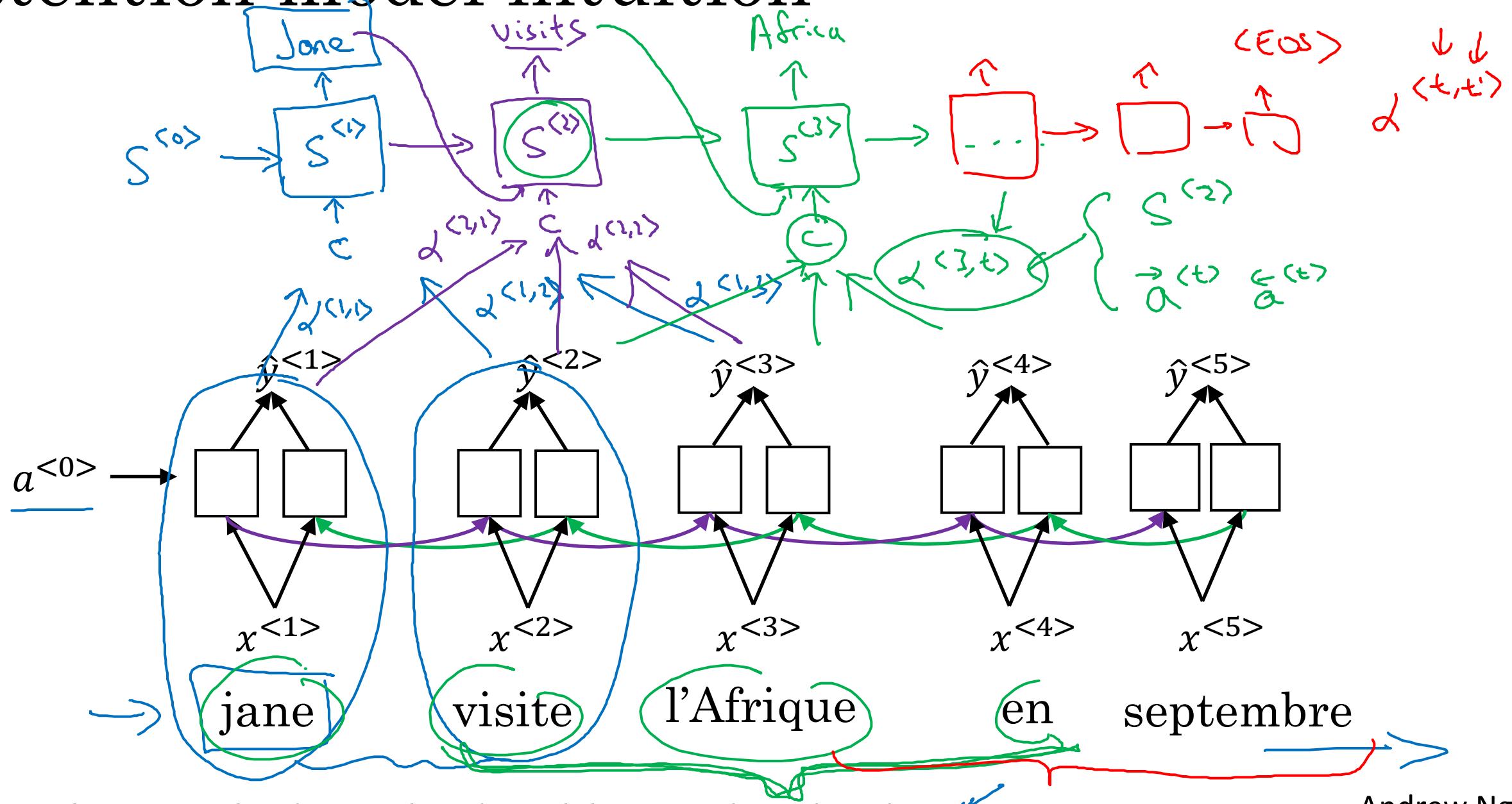


Jane s'est rendue en Afrique en septembre dernier, a apprécié la culture et a rencontré beaucoup de gens merveilleux; elle est revenue en parlant comment son voyage était merveilleux, et elle me tente d'y aller aussi.

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



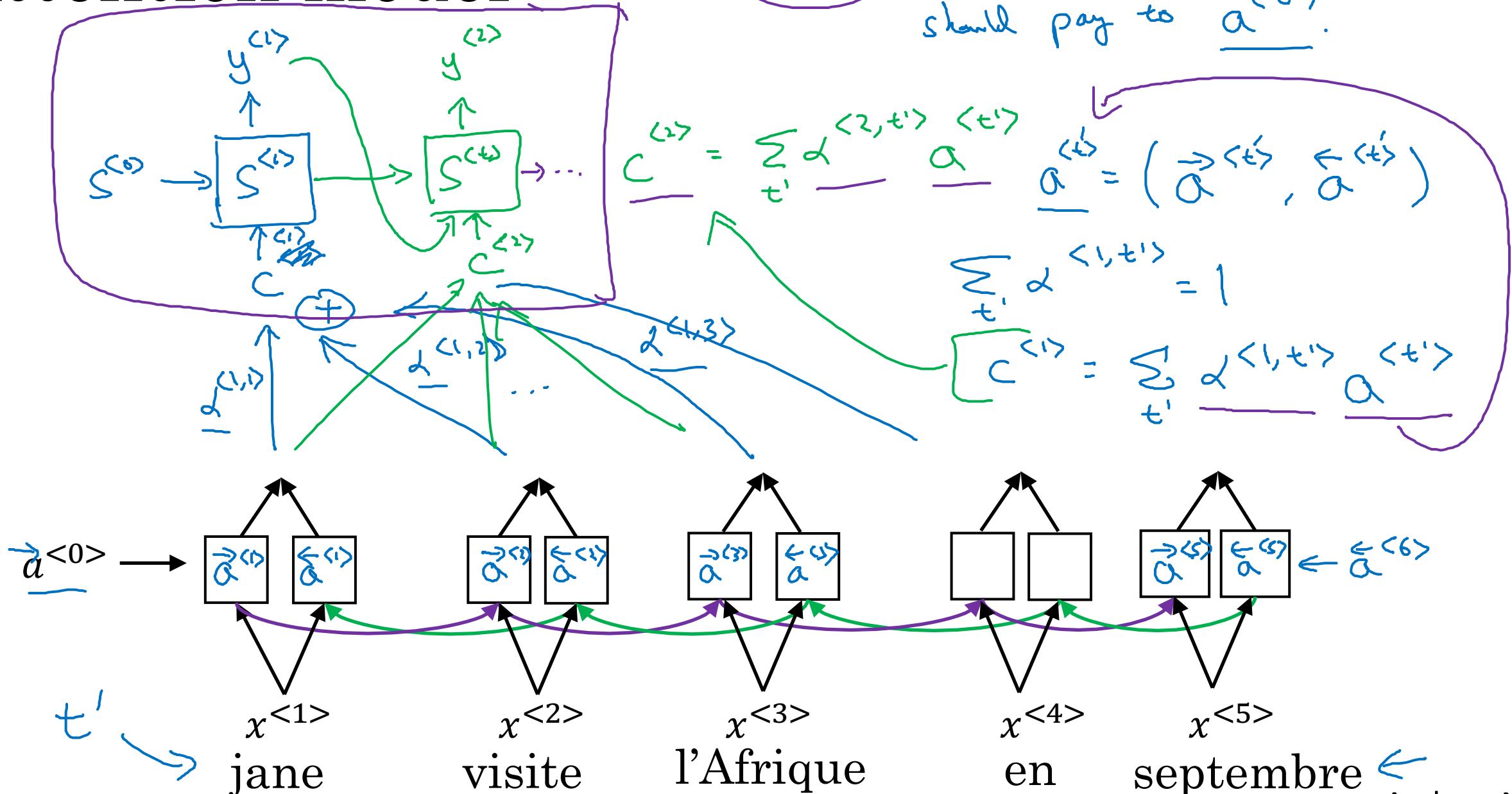


deeplearning.ai

Sequence to sequence models

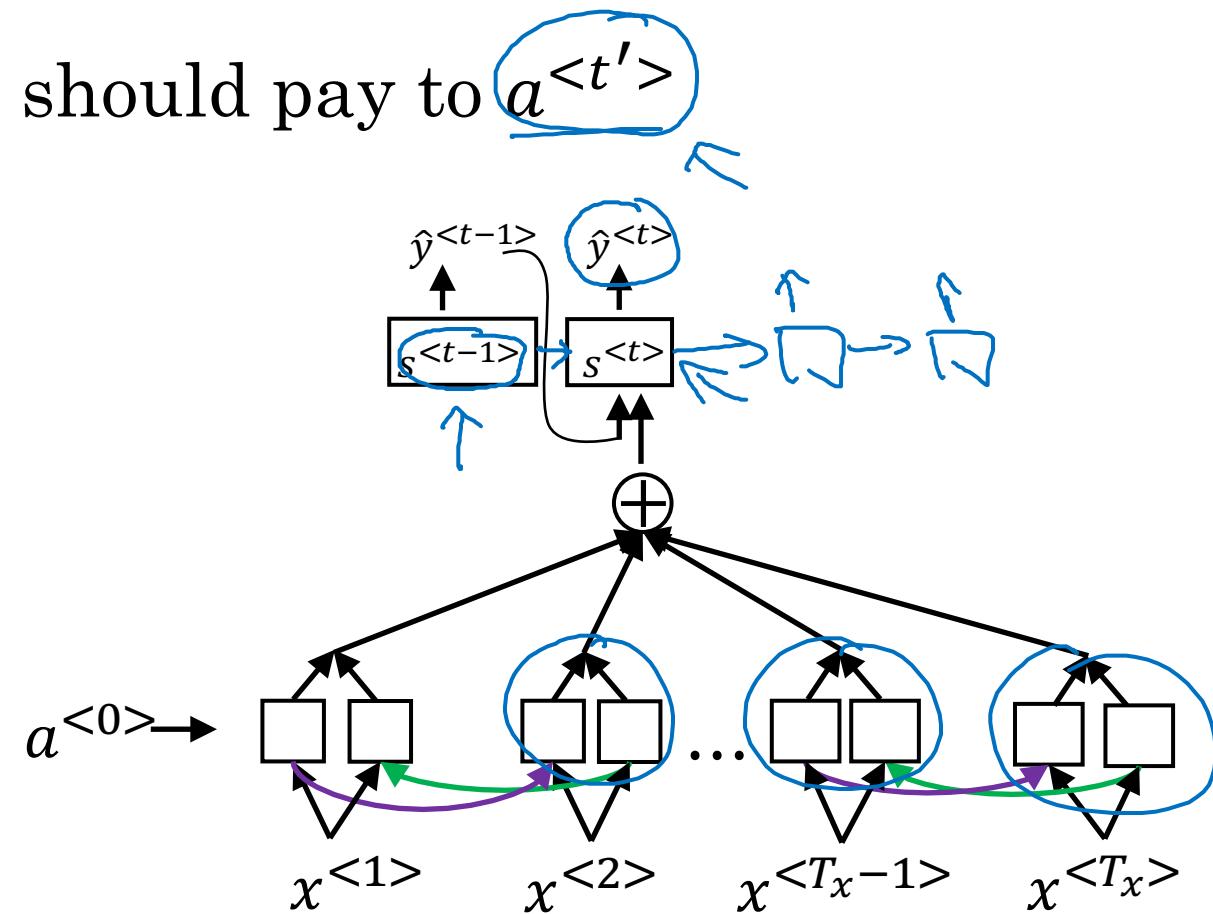
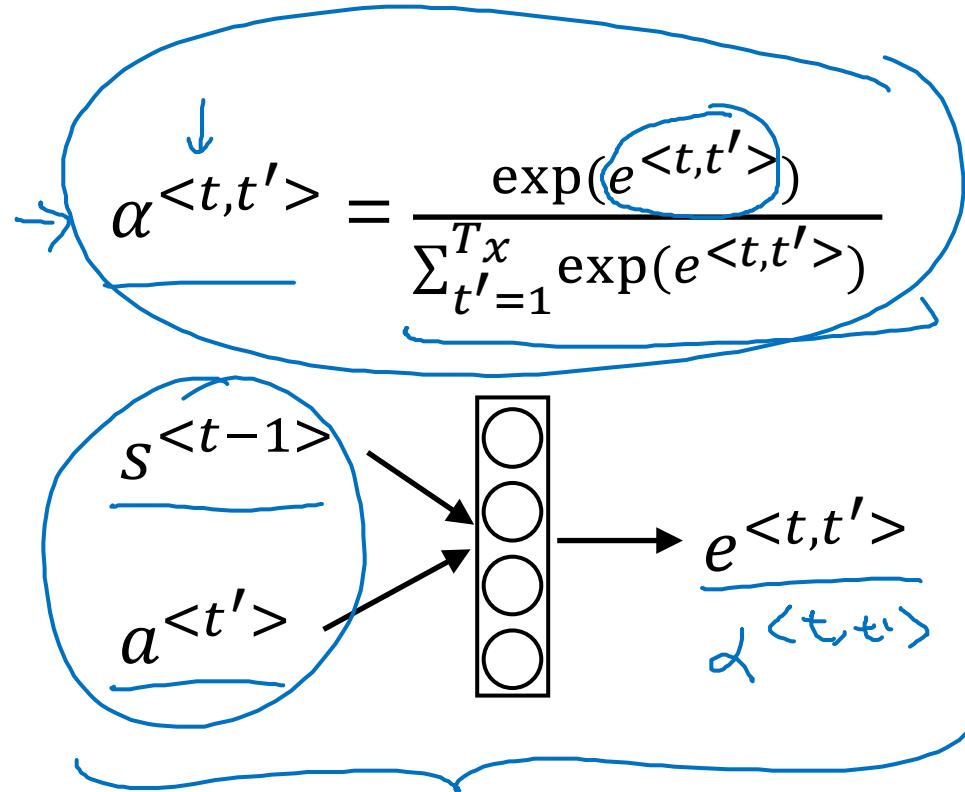
Attention model

Attention model



Computing attention $\underline{\alpha^{'}}$

$\alpha^{'}$ = amount of attention $y^{'}$ should pay to $a^{'}$



[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]

T_x

T_y

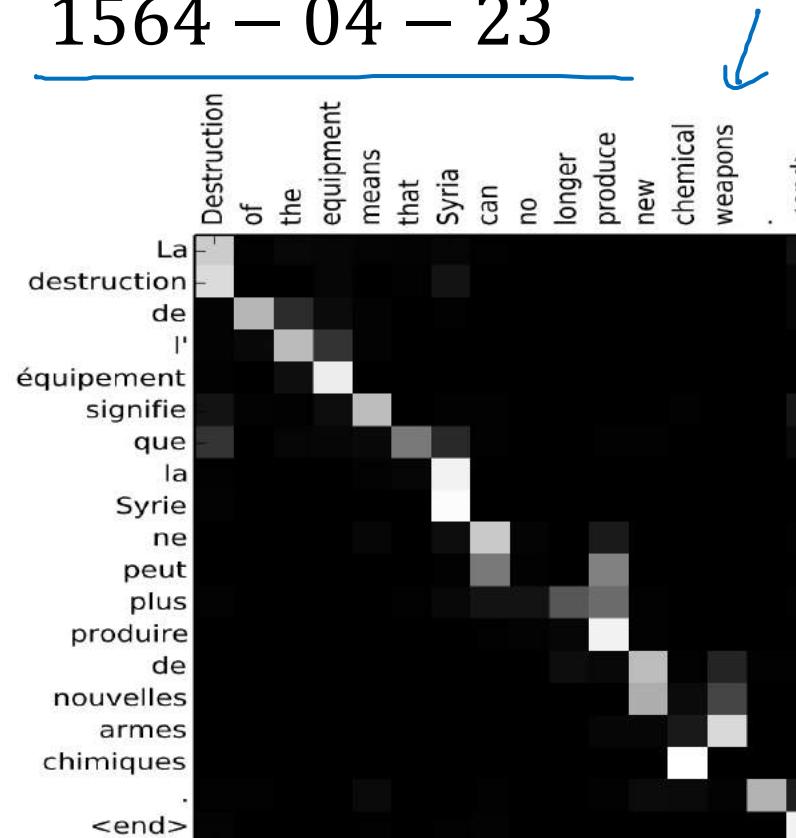
Andrew Ng

Attention examples

July 20th 1969 → 1969 – 07 – 20

23 April, 1564 → 1564 – 04 – 23

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



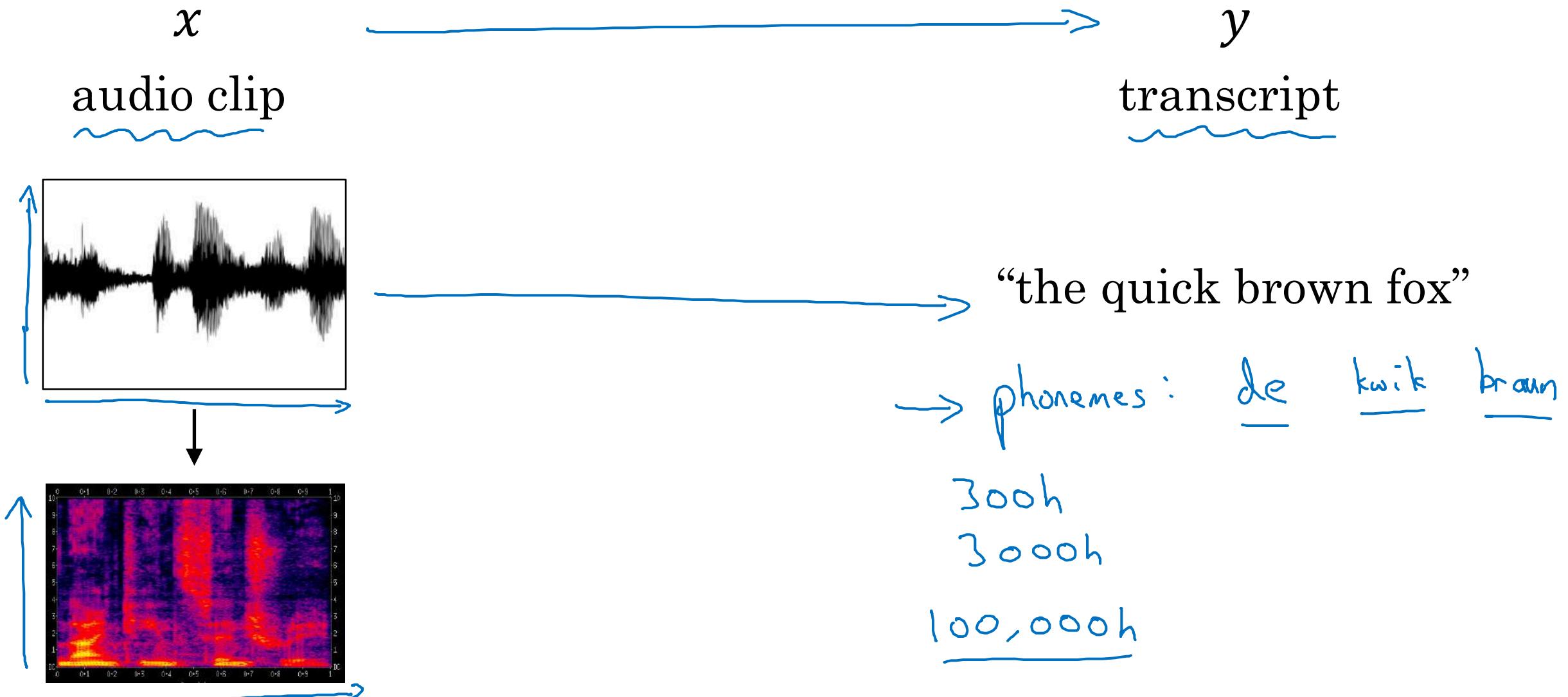


deeplearning.ai

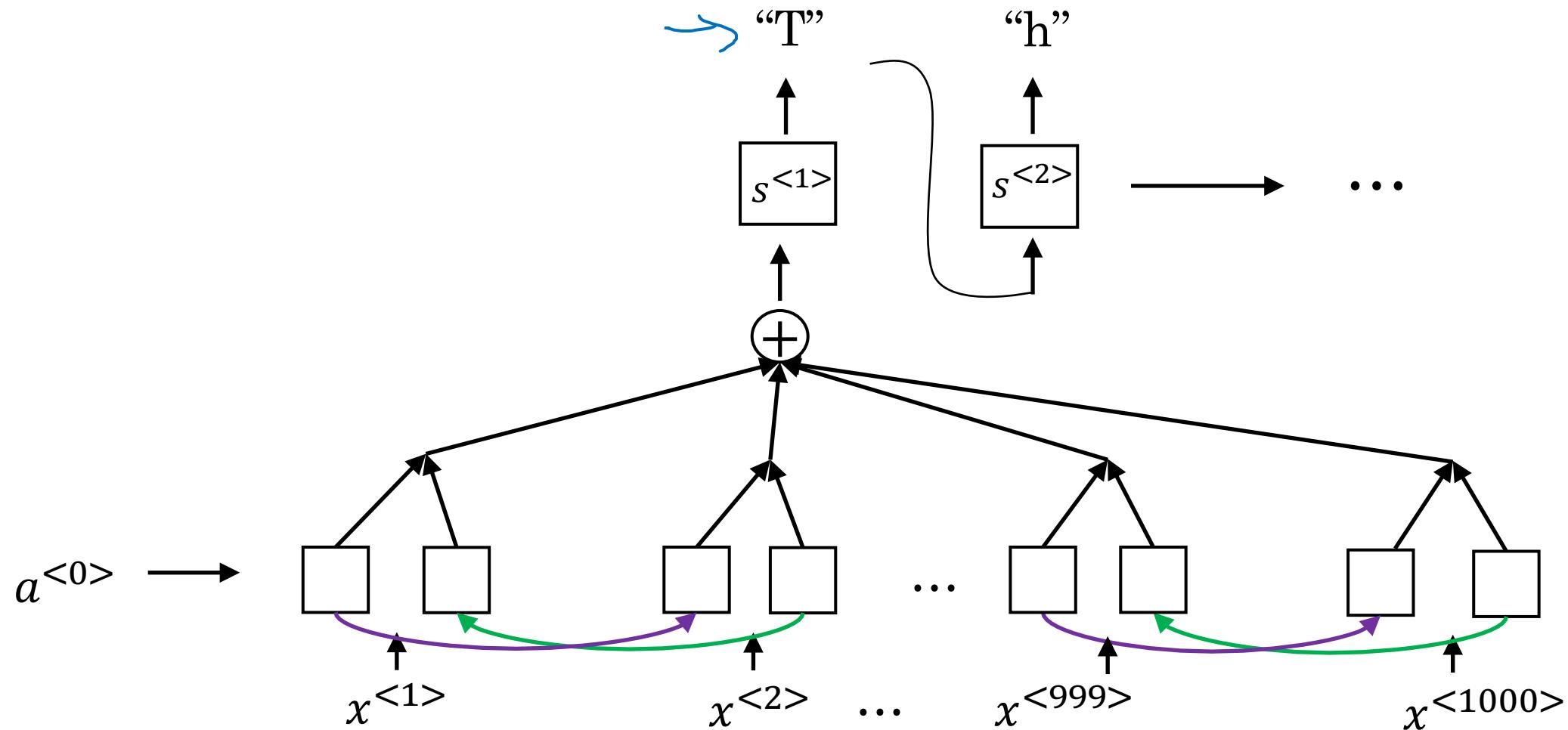
Audio data

Speech recognition

Speech recognition problem

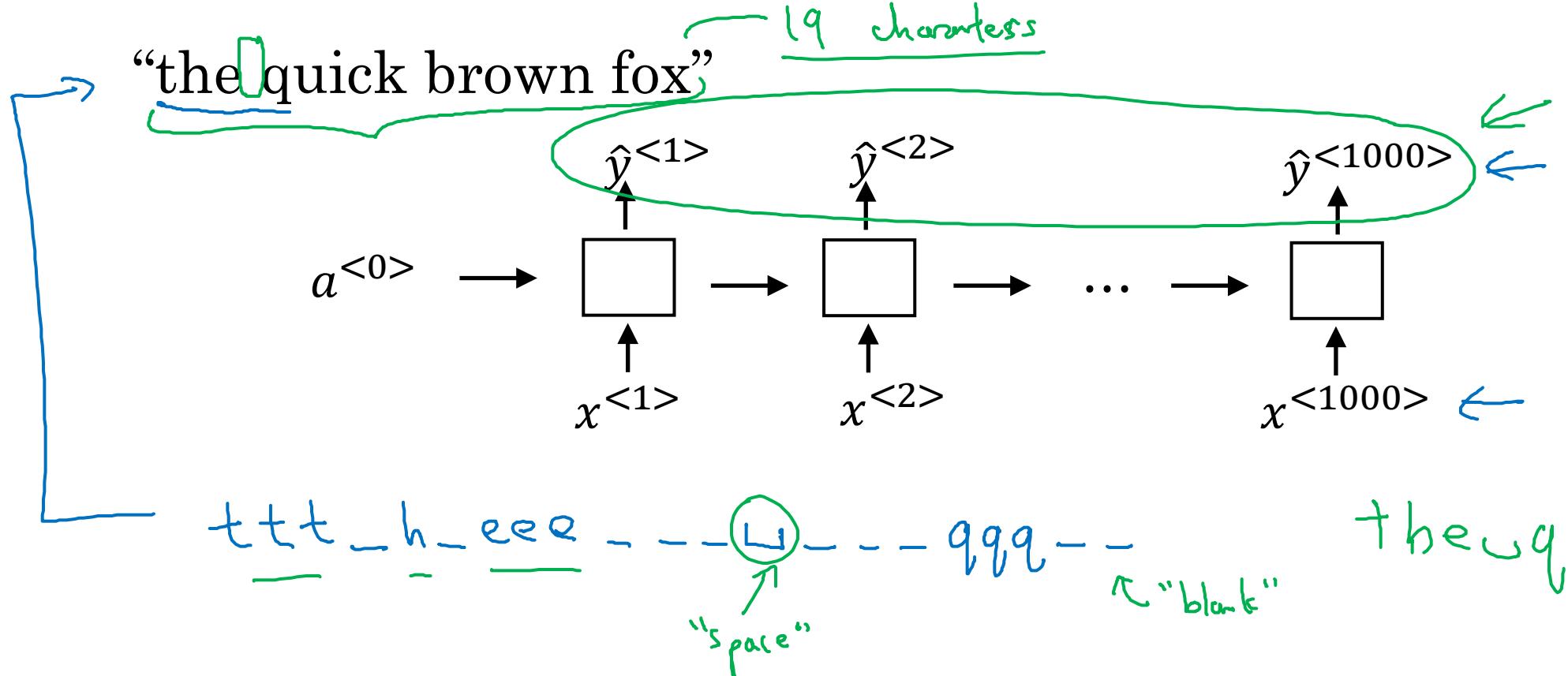


Attention model for speech recognition



CTC cost for speech recognition

(Connectionist temporal classification)



Basic rule: collapse repeated characters not separated by “blank” ↴

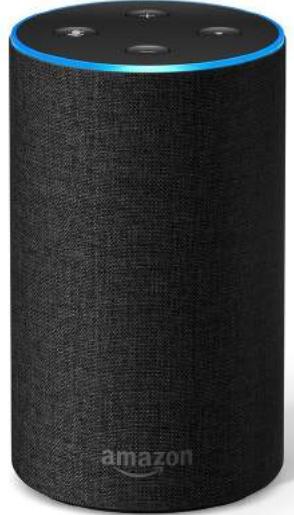


deeplearning.ai

Audio data

Trigger word
detection

What is trigger word detection?



Amazon Echo
(Alexa)



Baidu DuerOS
(xiaodunihao)

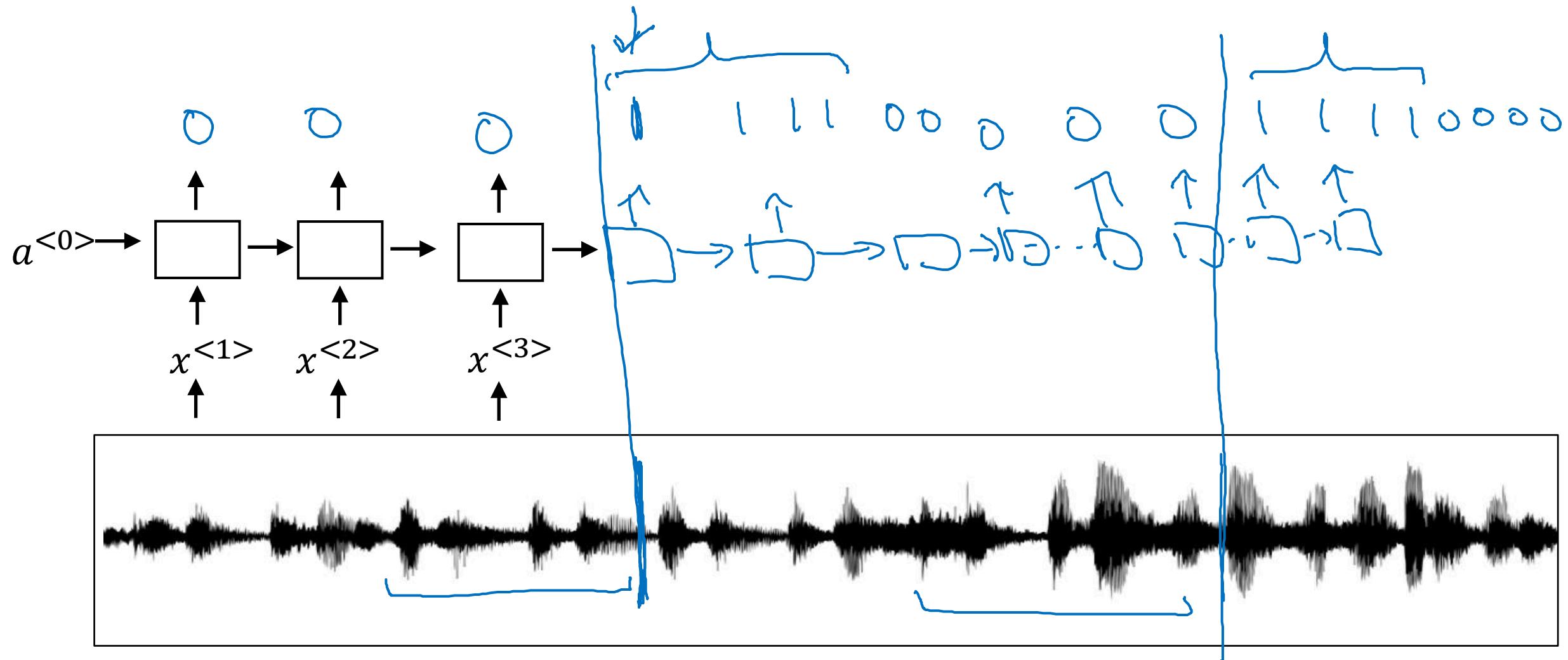


Apple Siri
(Hey Siri)



Google Home
(Okay Google)

Trigger word detection algorithm



Copyright Notice

These slides are distributed under the Creative Commons License.

[DeepLearning.AI](#) makes these slides available for educational purposes. You may not use or distribute these slides for commercial purposes. You may make copies of these slides and use or distribute them for educational purposes as long as you cite [DeepLearning.AI](#) as the source of the slides.

For the rest of the details of the license, see <https://creativecommons.org/licenses/by-sa/2.0/legalcode>



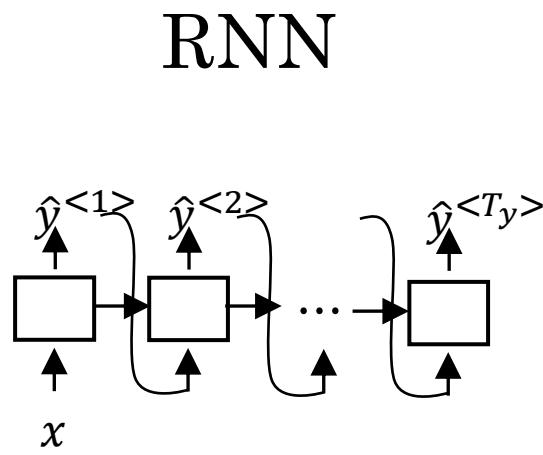
deeplearning.ai

Sequence to sequence models

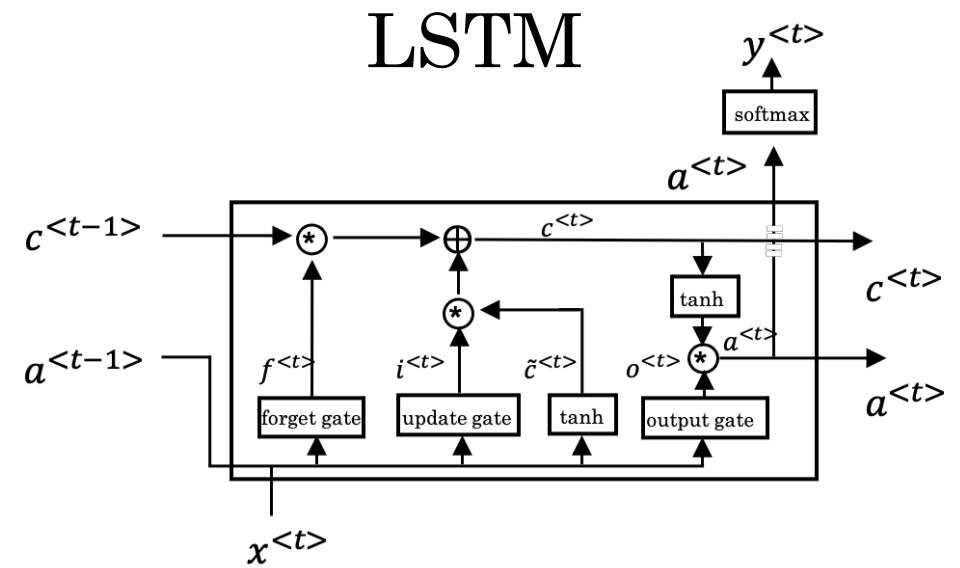
Transformers Intuition

Transformers Motivation

Increased complexity,
sequential

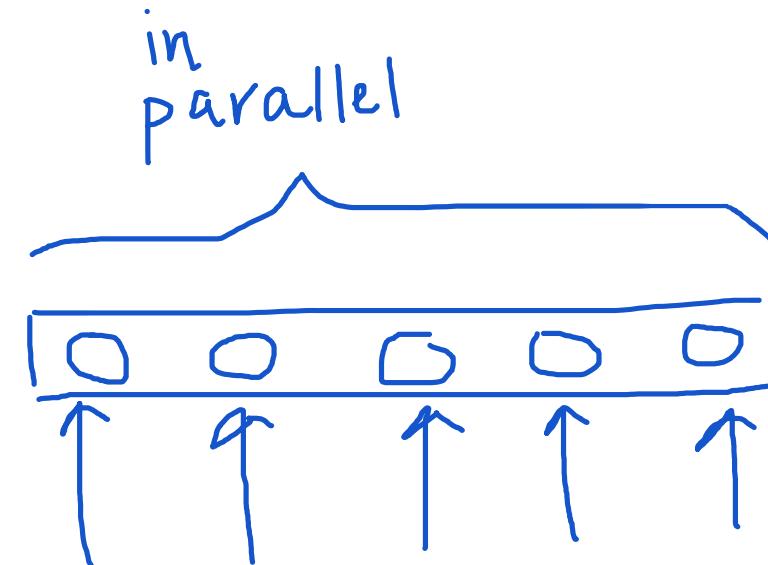
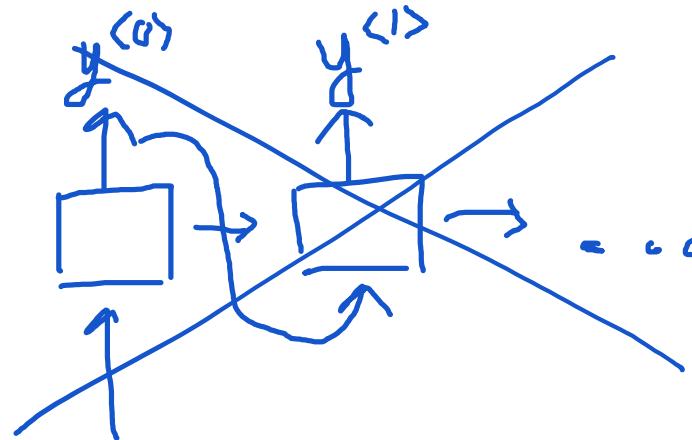


GRU



Transformers Intuition

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





deeplearning.ai

Sequence to sequence models

Self-Attention

Self-Attention Intuition

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

RNN Attention

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

$x^{<1>}$
Jane

$x^{<2>}$
visite

$x^{<3>}$
l'Afrique

$x^{<4>}$
en

$x^{<5>}$
septembre

Transformers Attention

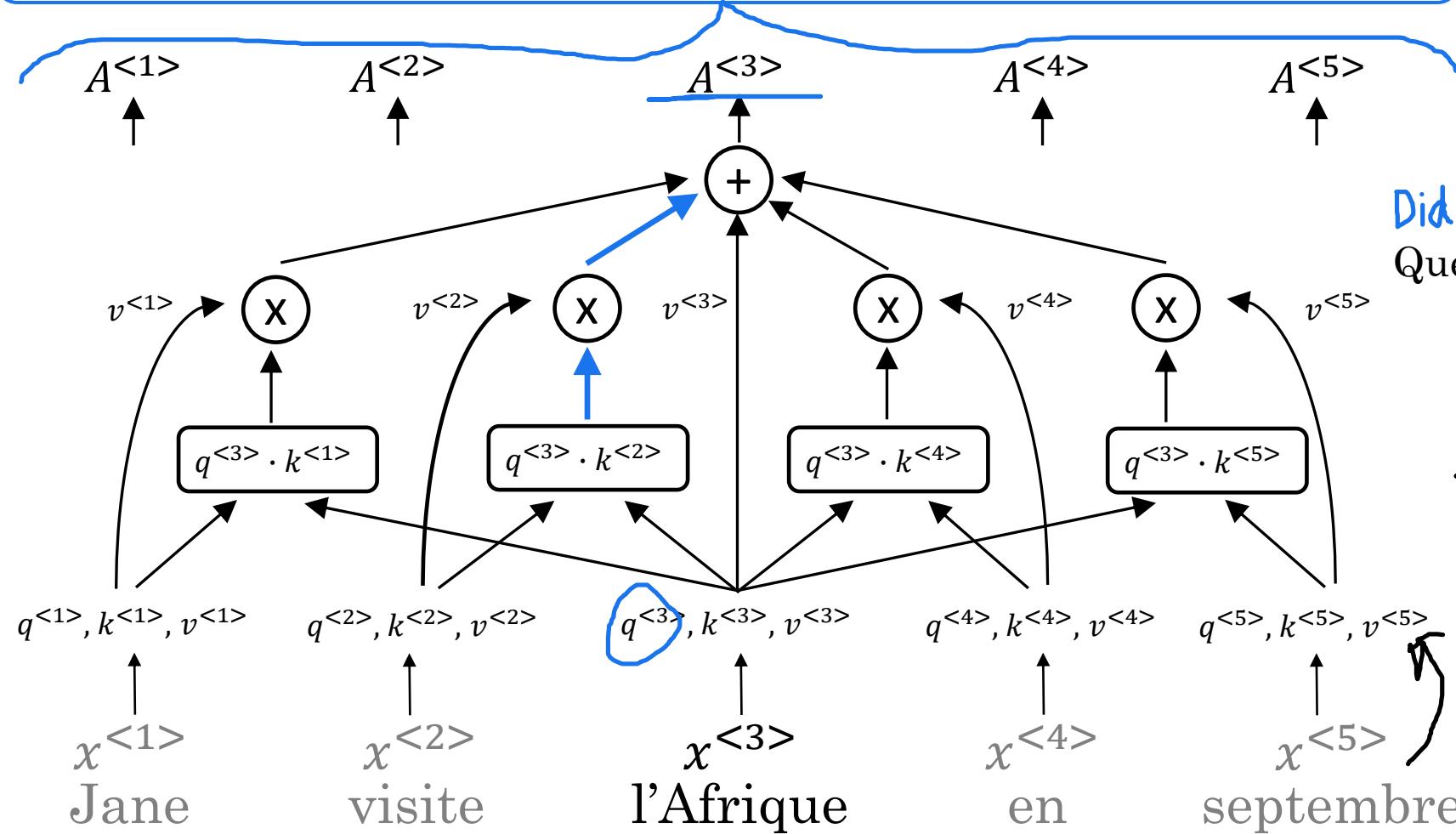
$$A(q, K, V) = \sum_i \frac{\exp(e^{}})}{\sum_j \exp(e^{}})} v^{<i>}$$

Self-Attention

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

Softmax

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



Did what?

Query (Q)

$q^{<1>}$
 $q^{<2>}$
 $\underline{q^{<3>}}$
 $\underline{q^{<4>}}$
 $q^{<5>}$

Key (K)

$k^{<1>}$
 $k^{<2>}$
 $\underline{k^{<3>}}$
 $\underline{k^{<4>}}$
 $k^{<5>}$

Value (V)

$v^{<1>}$
 $v^{<2>}$
 $\underline{v^{<3>}}$
 $v^{<4>}$
 $v^{<5>}$

W^Q, W^K, W^V

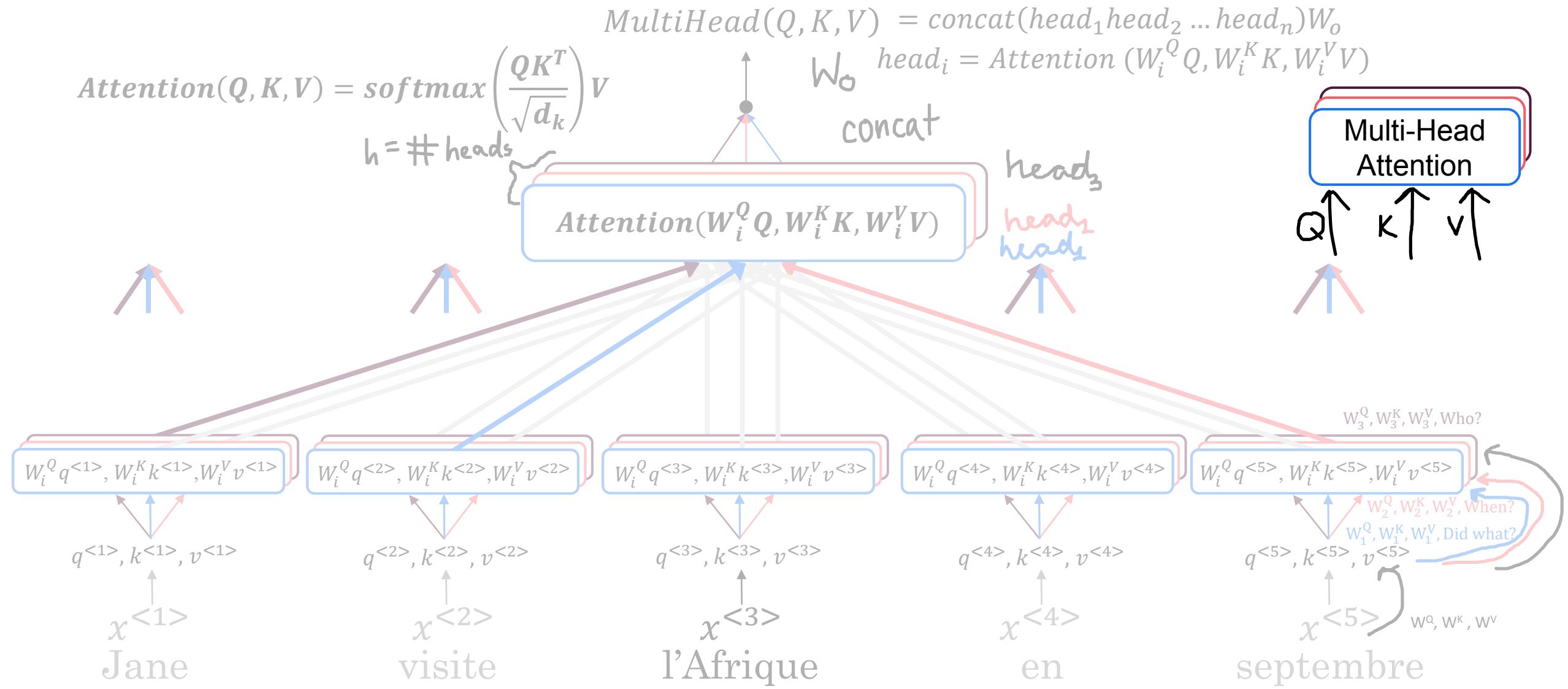


deeplearning.ai

Sequence to sequence models

Multi-Head Attention

Multi-Head Attention





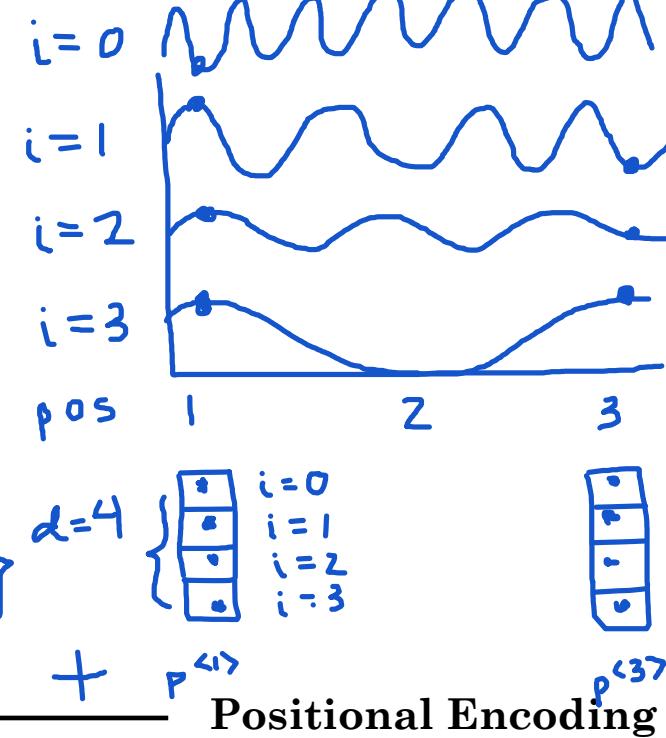
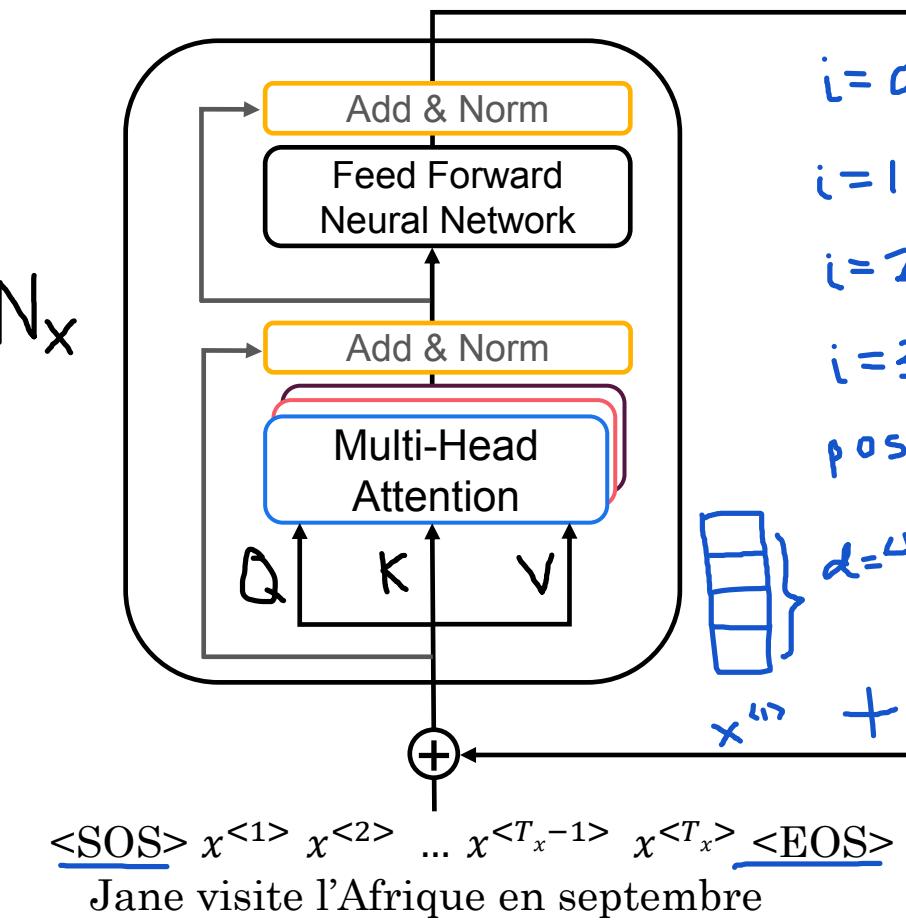
deeplearning.ai

Sequence to sequence models

Transformers

Transformer Details

Encoder

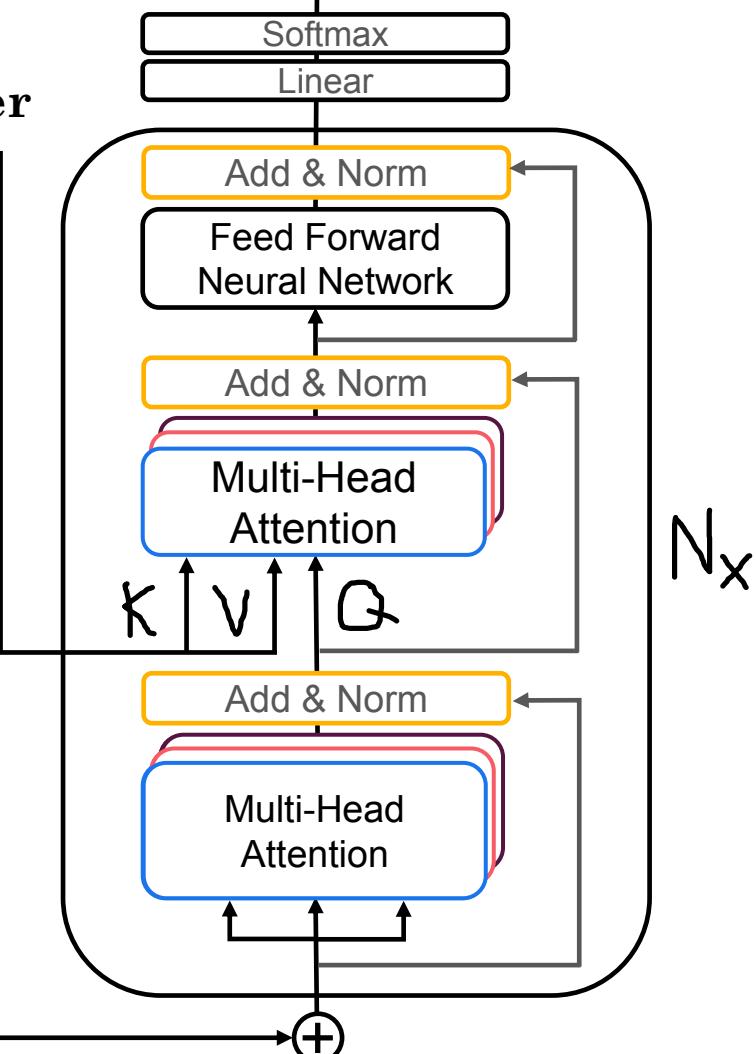


$$PE_{(pos,2i)} = \sin\left(\frac{pos}{1000^{\frac{2i}{d}}}\right)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{1000^{\frac{2i+1}{d}}}\right)$$

<SOS> Jane visits Africa in September <EOS>

Decoder



$<\text{SOS}>$ Jane visits Africa in September



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.



www.shutterstock.com • 331201091

Thank you.

- Andrew Ng