

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

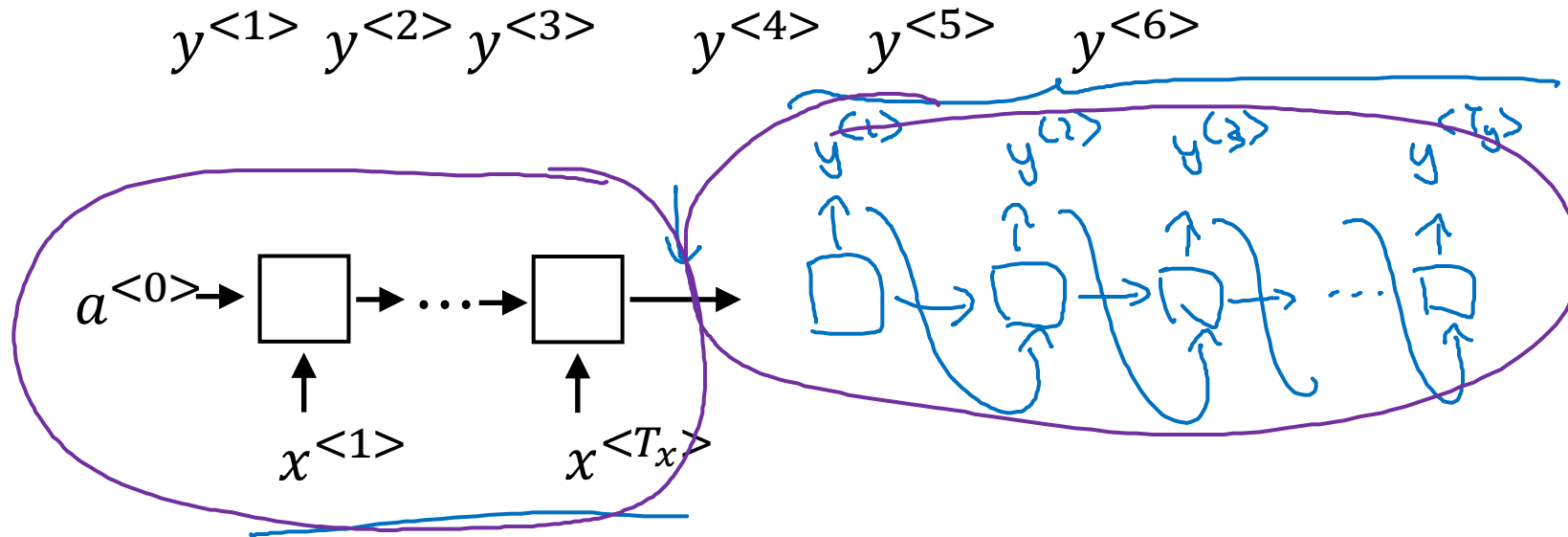
Sequence to sequence models

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

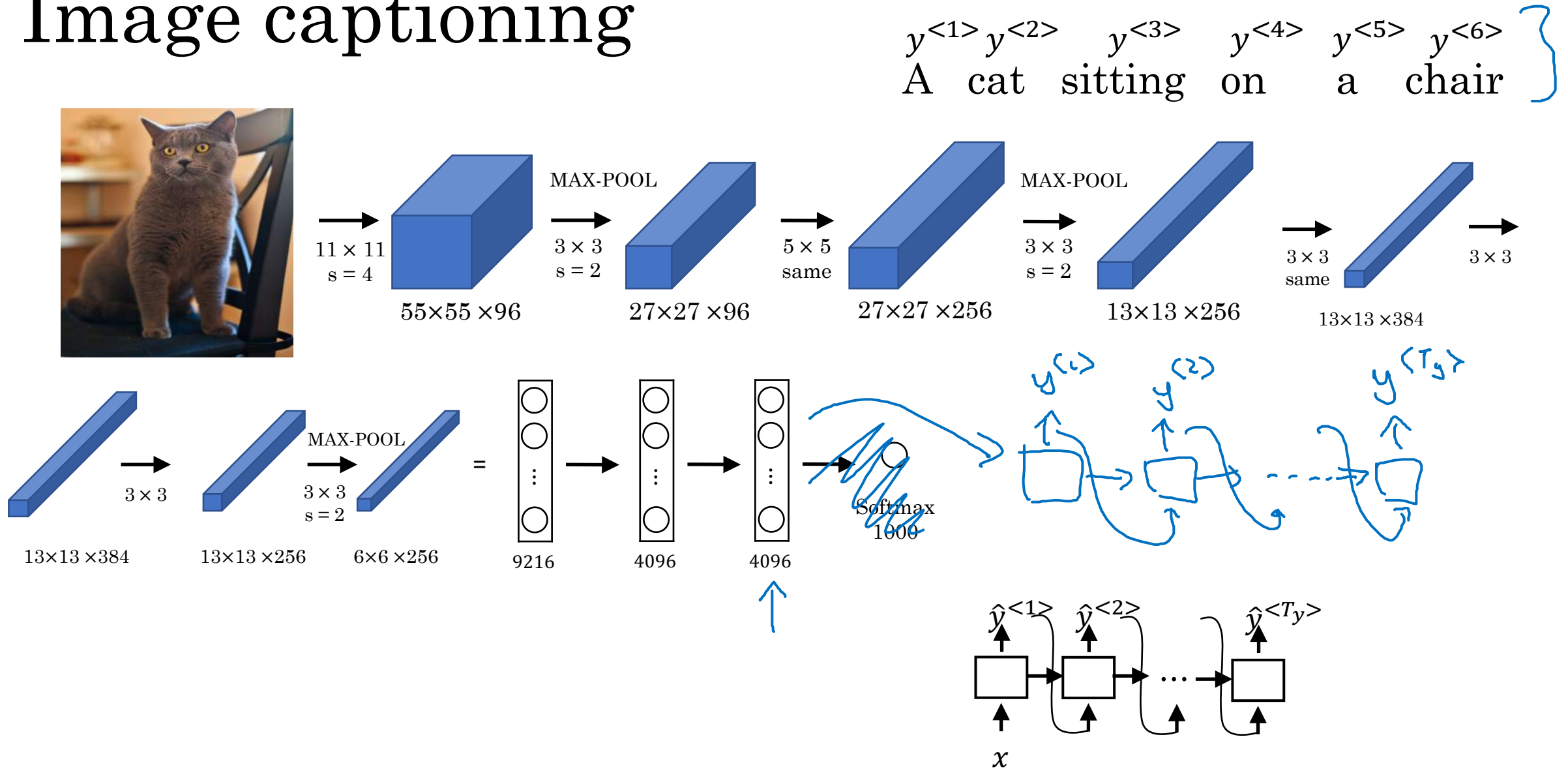
→ Jane is visiting Africa in September.



[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]

Andrew Ng



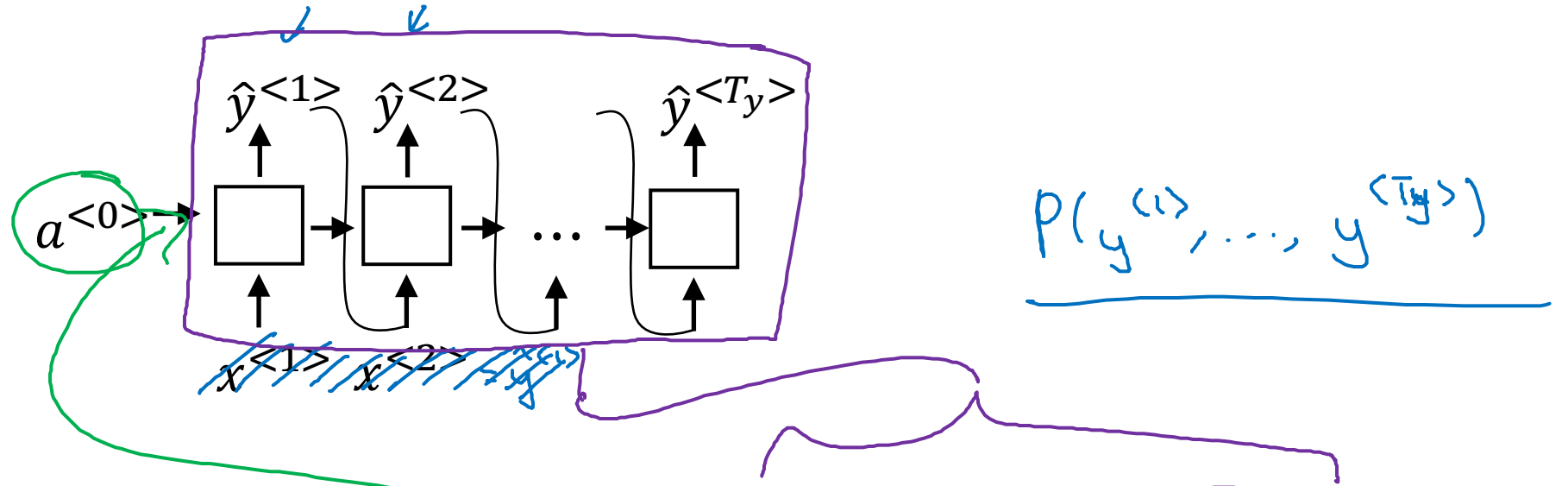
deeplearning.ai

Sequence to sequence models

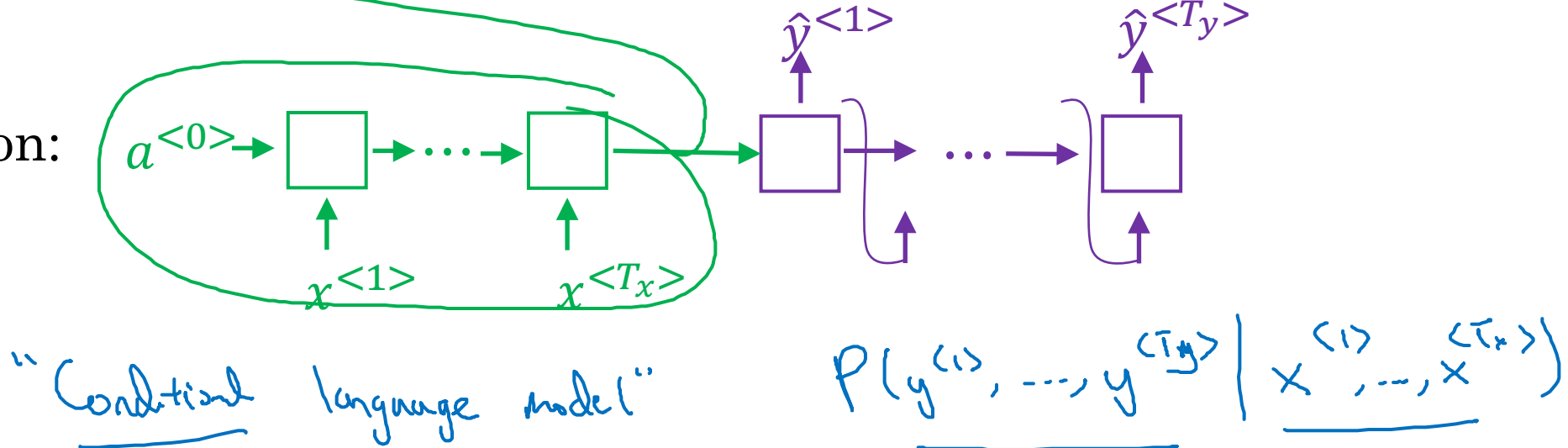
Picking the most likely sentence

Machine translation as building a conditional language model

Language model:



Machine translation:



Finding the most likely translation

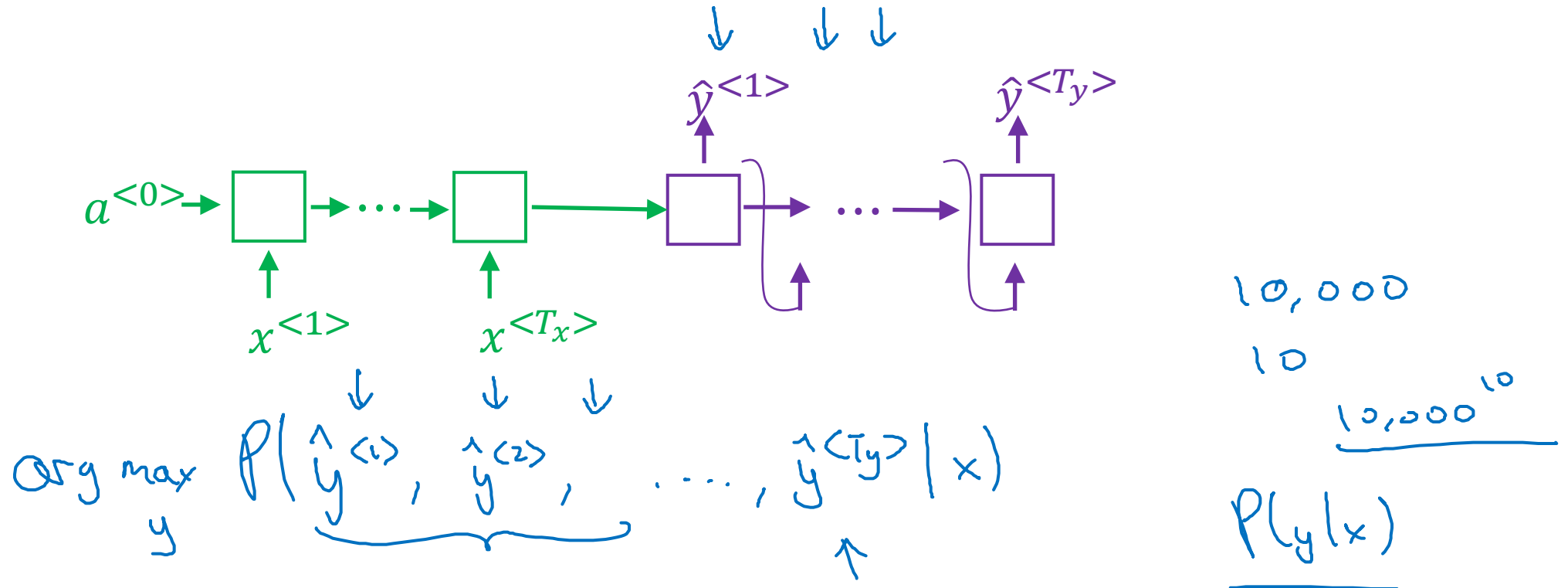
Jane visite l'Afrique en septembre.

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

- 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.

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

Why not a greedy search?



→ 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

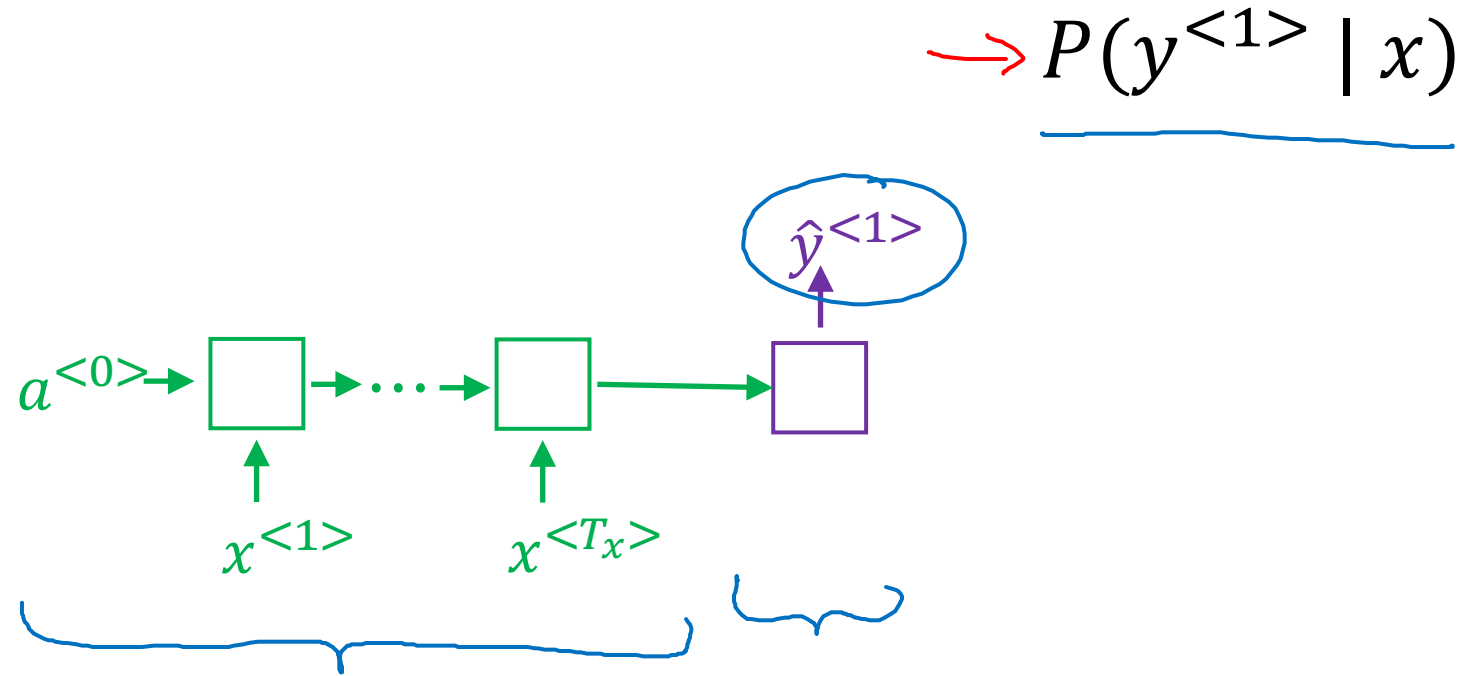
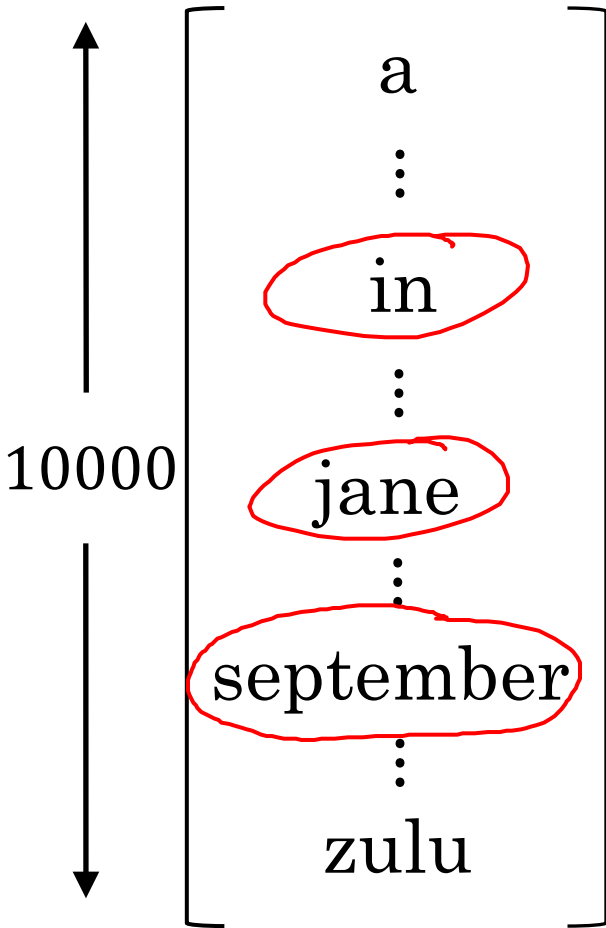
Sequence to sequence models

Beam search

Beam search algorithm

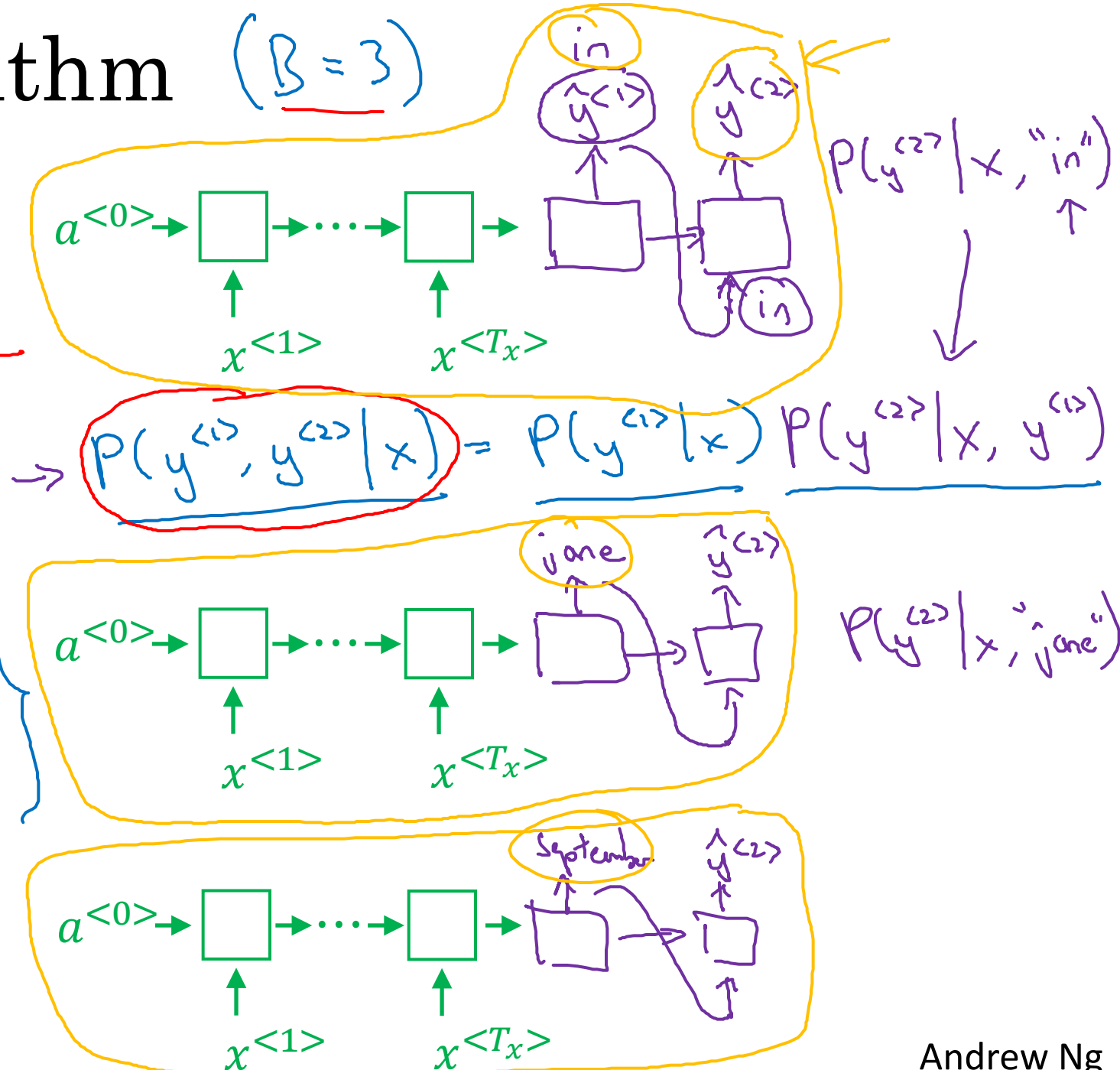
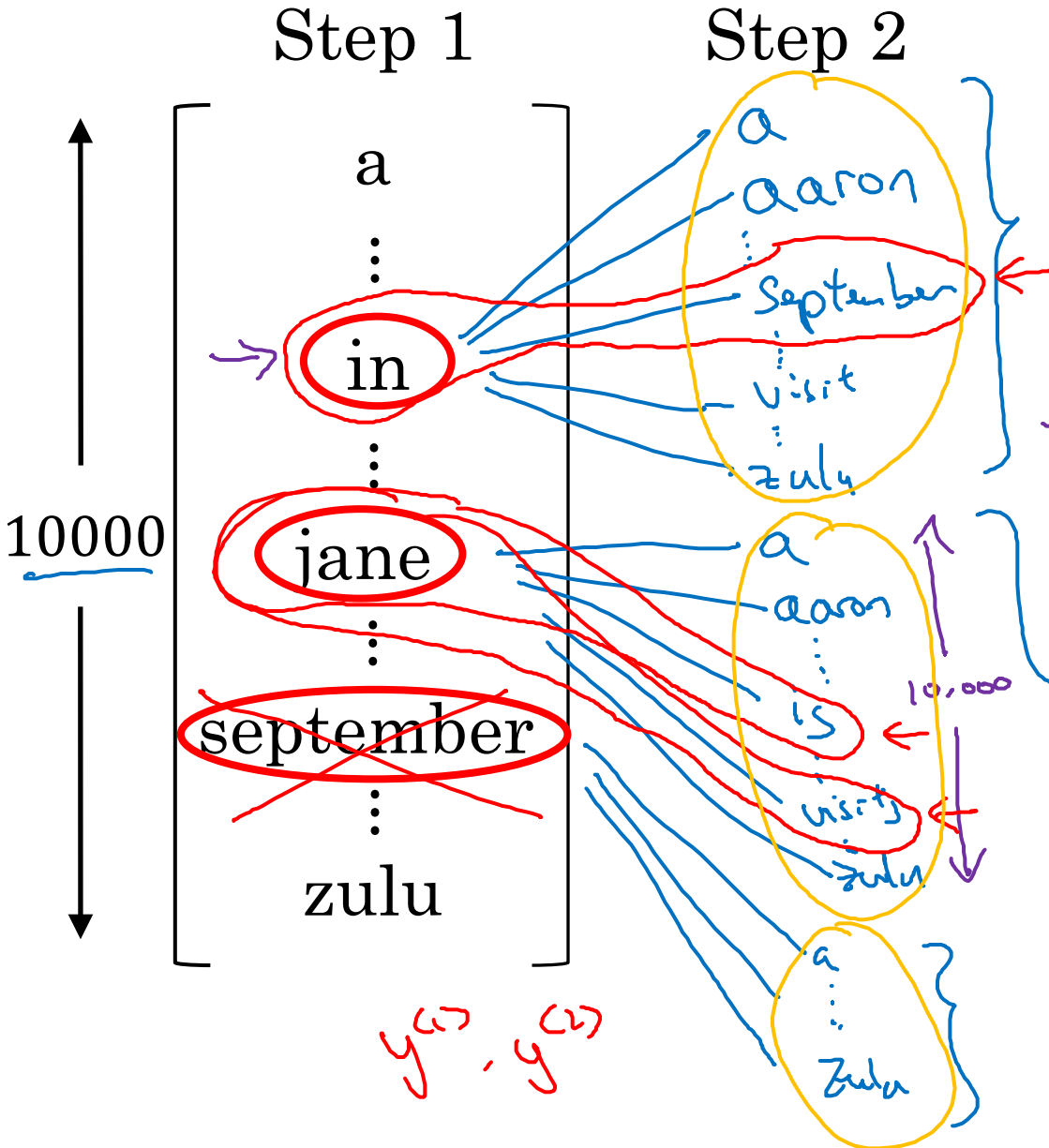
$B = 3$ (beam width)

Step 1



Beam search algorithm

(B=3)

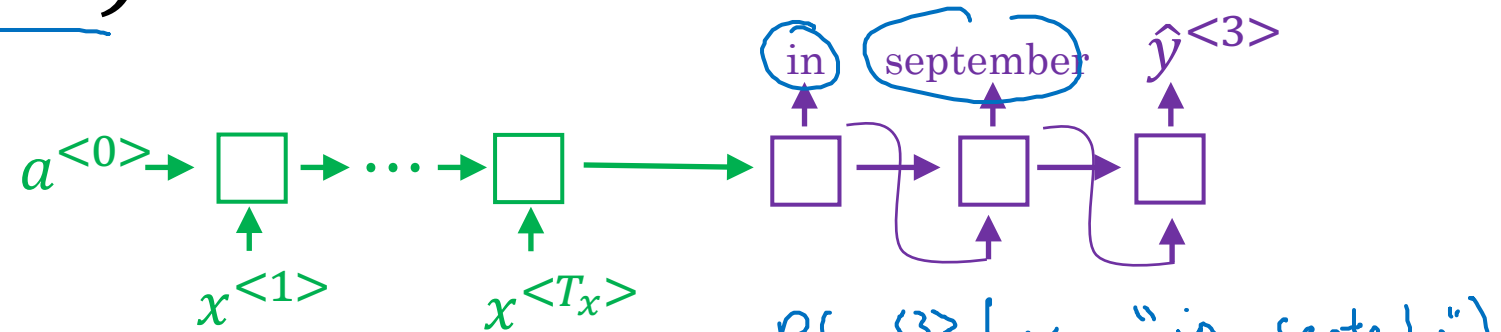


Beam search ($B = 3$)

$B=1 \rightsquigarrow$ greedy search

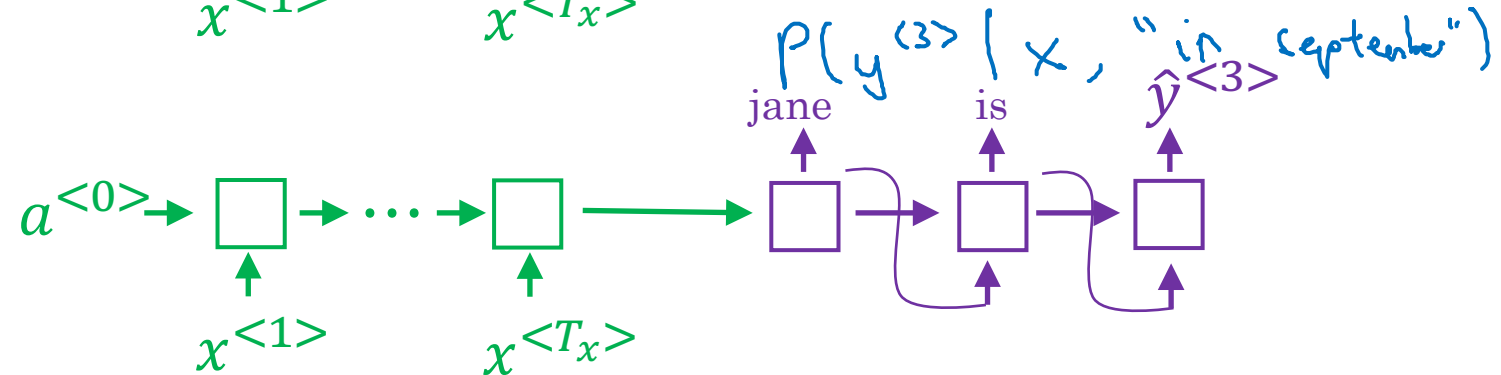
in september

a
aaron
jane
zulu



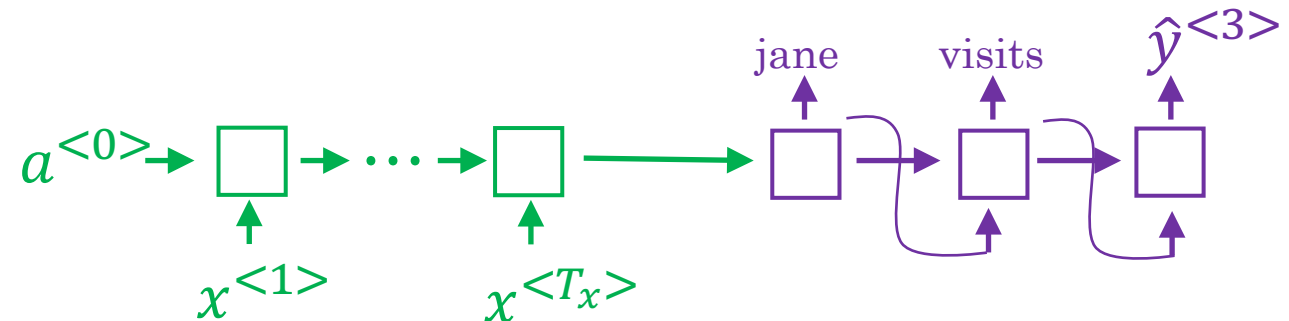
jane is

a
visits
zulu



jane visits

a
africa
zulu



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

jane visits africa in september. <EOS>



deeplearning.ai

Sequence to sequence models

Refinements to beam search

Length normalization

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

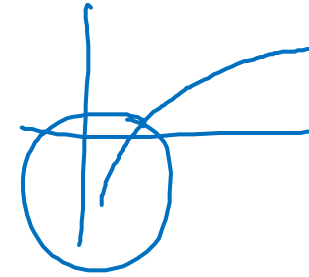
$$\arg \max_y \prod_{t=1}^{T_y} P(y^{(t)} | x, y^{(1)}, \dots, y^{(t-1)})$$

log

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

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

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



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

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

$$\underline{\alpha = 0.7}$$

$$\underline{\alpha = 1}$$

$$\underline{\alpha = 0}$$

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



deeplearning.ai

Sequence to sequence models

Error analysis on beam search

Example

Jane visite l'Afrique en septembre.

→ RNN

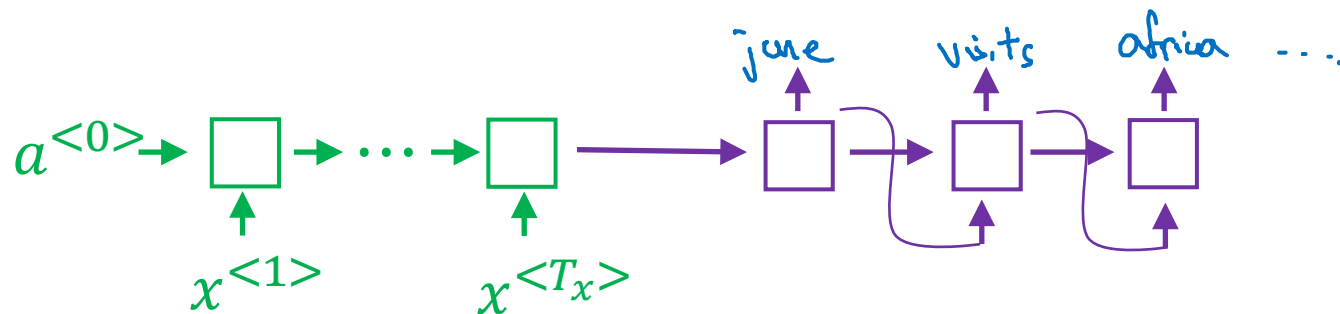
→ Beam Search

BT

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

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

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



Error analysis on beam search

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

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

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

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

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. - - - ...	$\frac{2 \times 10^{-10}}{\text{---}}$ ---	$\frac{1 \times 10^{-10}}{\text{---}}$ ---	<div>B</div> <div>R</div> <div>R</div> <div>R</div> <div>R</div> <div>...</div>

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.

Bleu
bilingual evaluation understudy

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 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 ←	
	↑		

Countclip refers to Max(Count of bigrams in both reference sentences 1 & 2)

$$\frac{\text{Sum of Count_clip's}}{\text{Sum of Count_of_bigram's}} = \frac{4}{6}$$

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})

$$P_1, P_2 = \underline{1.0}$$

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

Handwritten notes: \hat{y} is written above the summation indices. $\text{count}_{clip}(unigram)$ and $\text{count}(unigram)$ are written in blue ink next to their respective terms. An arrow points from the word "unigram" written in blue at the bottom left to the summation index.

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

Handwritten notes: \hat{y} is written above the summation indices. $\text{count}_{clip}(ngram)$ and $\text{count}(ngram)$ are written in blue ink next to their respective terms. An arrow points from the word "ngram" written in blue at the bottom left to the summation index.

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

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

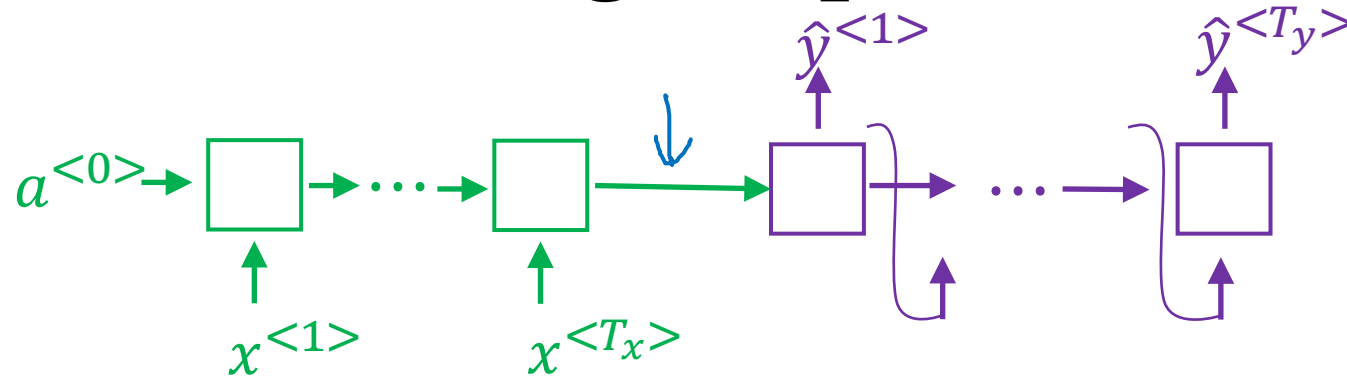


deeplearning.ai

Sequence to
sequence models

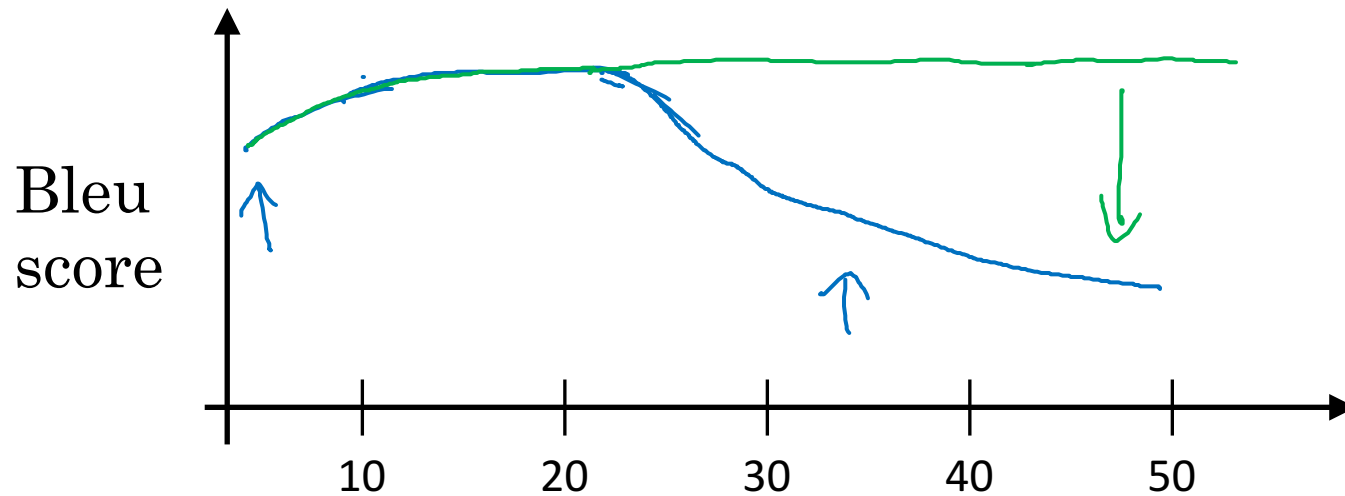
Attention model
intuition

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.

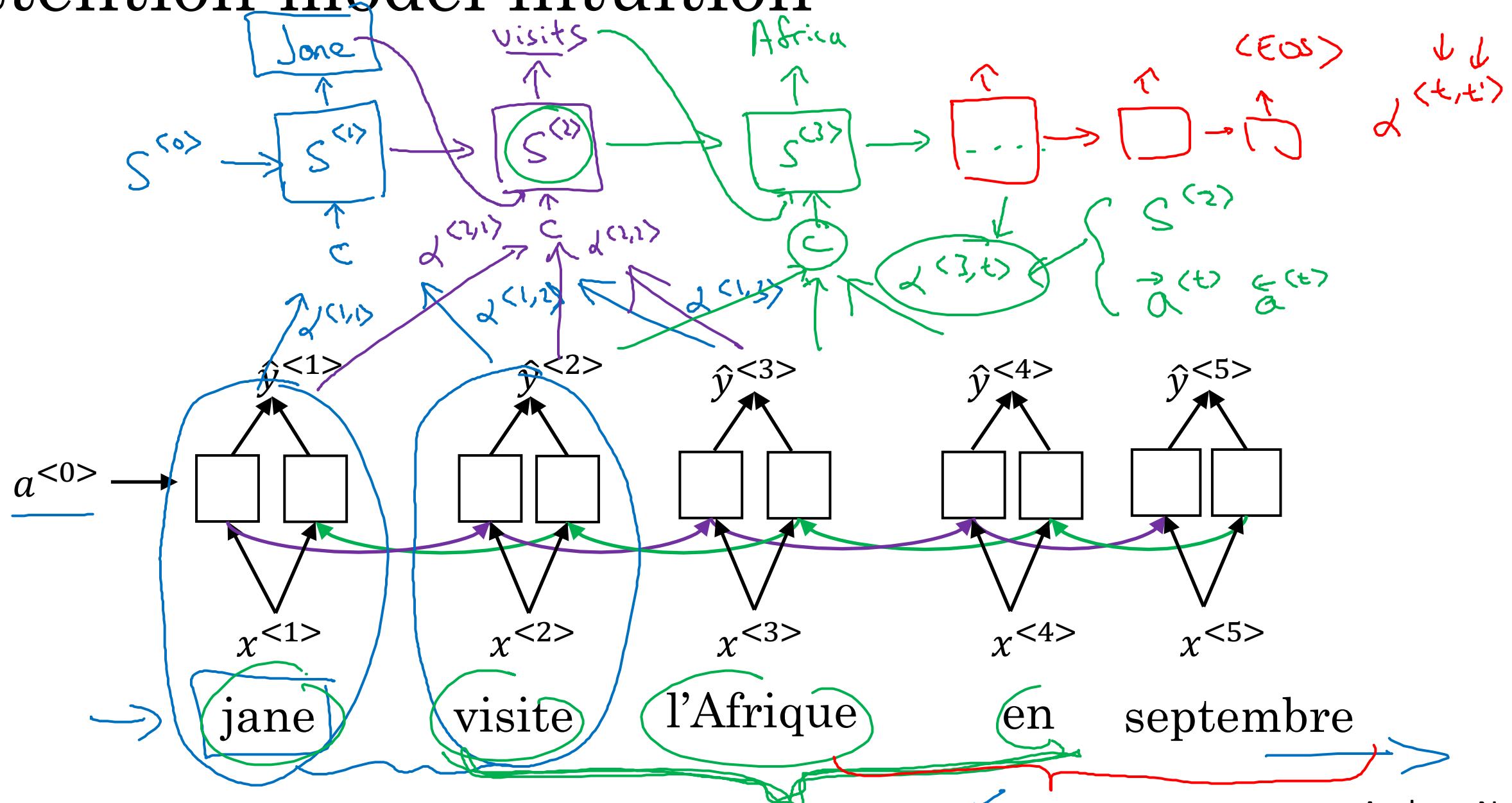
Jane went to Africa last September, and enjoyed the culture and met many wonderful people; she came back raving about how wonderful her trip was, and is tempting me to go too.



With increase in sentence length, the Bleu score tends to drop after a certain number of input sequence length with the traditional many-to-many models.

Sentence length

Attention model intuition





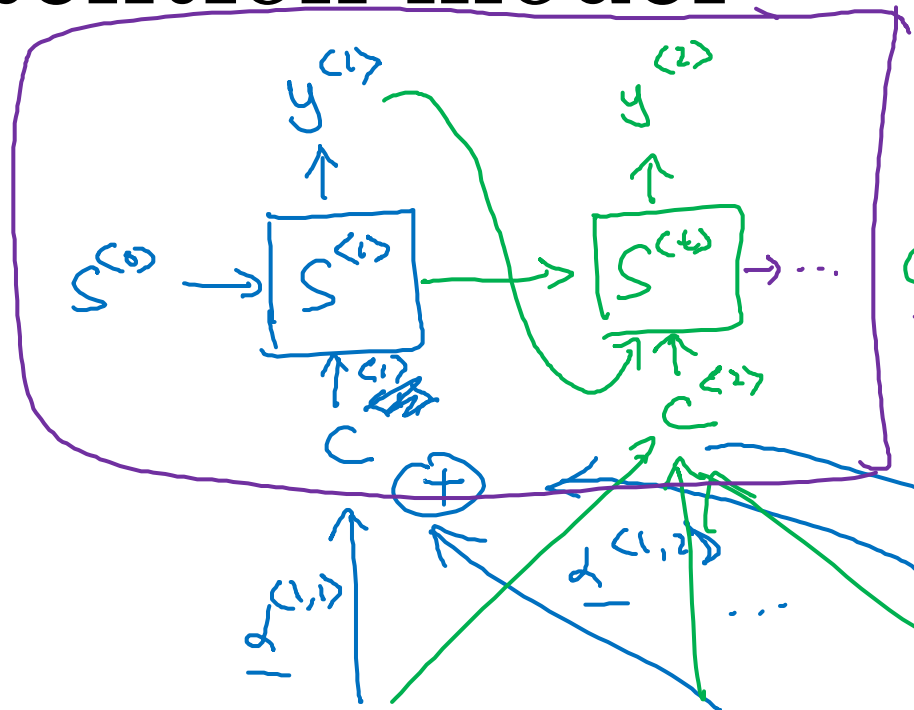
deeplearning.ai

Sequence to sequence models

Attention model

Attention model

$\alpha^{(t,t')}$ = amount of 'attention' $y^{(t)}$ should pay to $a^{(t')}$.

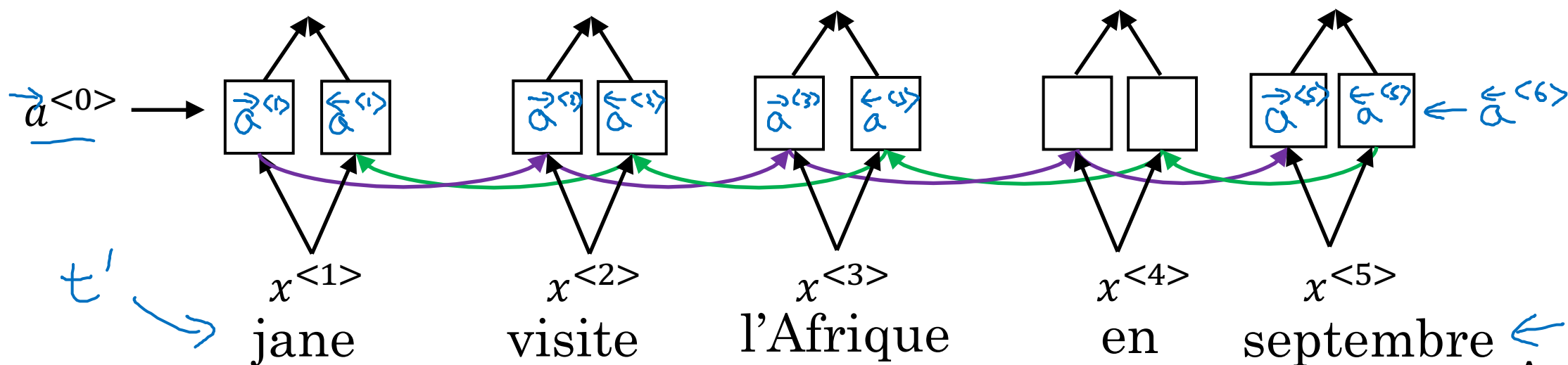


$$c^{(2)} = \sum_{t'} \alpha^{(2,t')} a^{(t')}$$

$$a^{(t')} = (\vec{a}^{(t')}, \leftarrow{a}^{(t')})$$

$$\sum_{t'} \alpha^{(1,t')} = 1$$

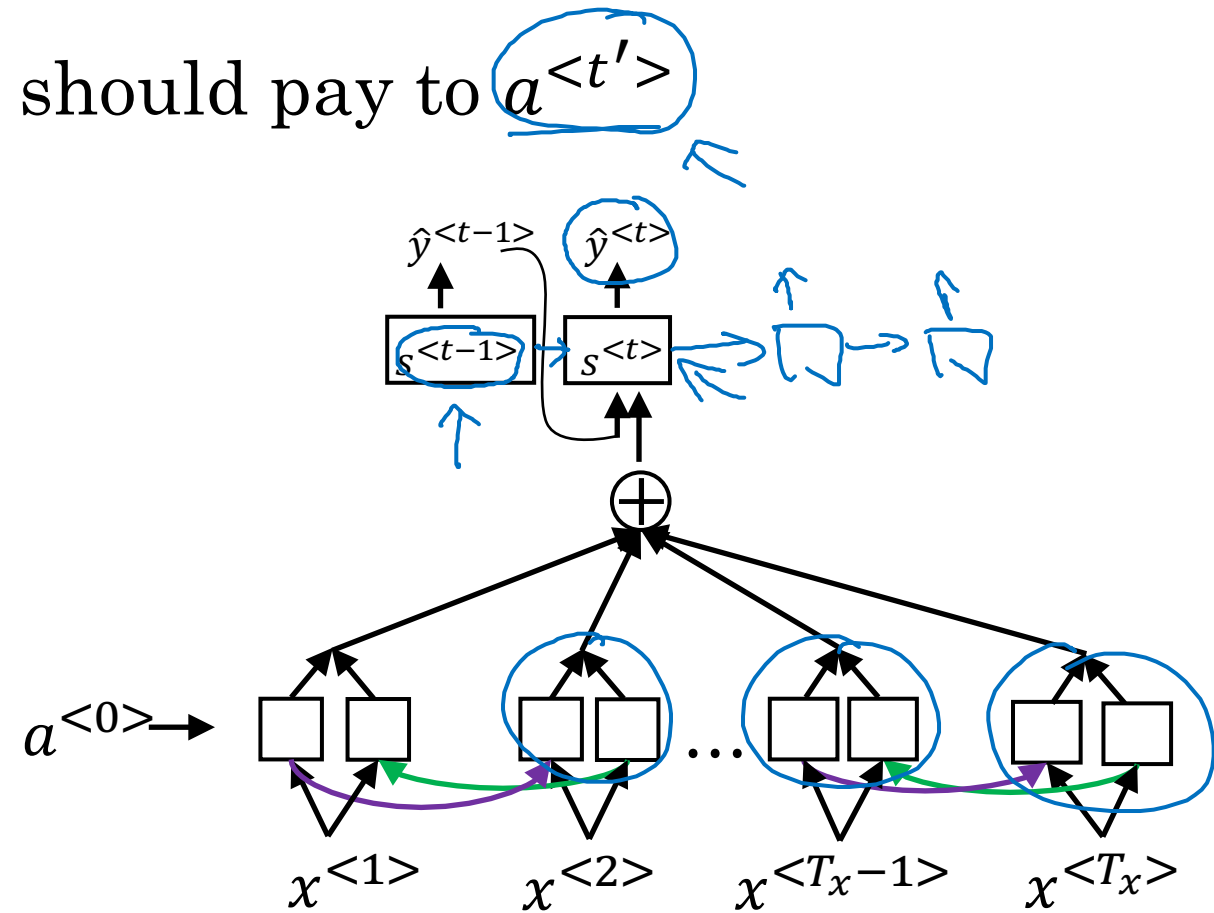
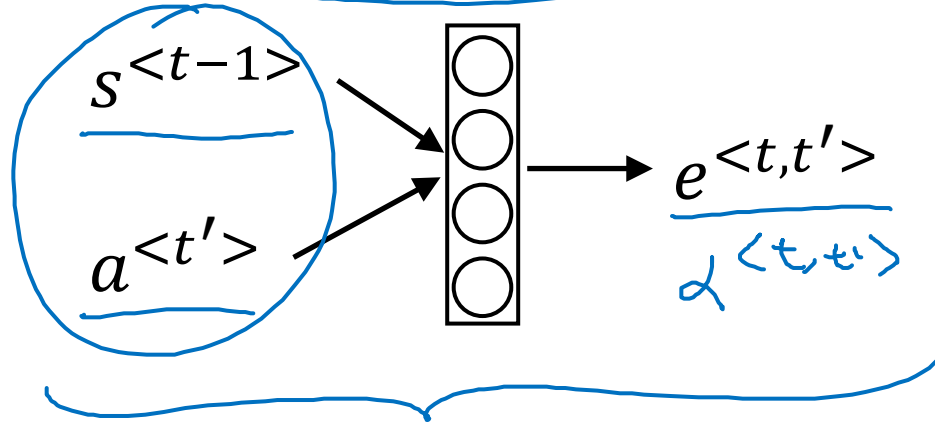
$$c^{(1)} = \sum_{t'} \alpha^{(1,t')} a^{(t')}$$



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

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

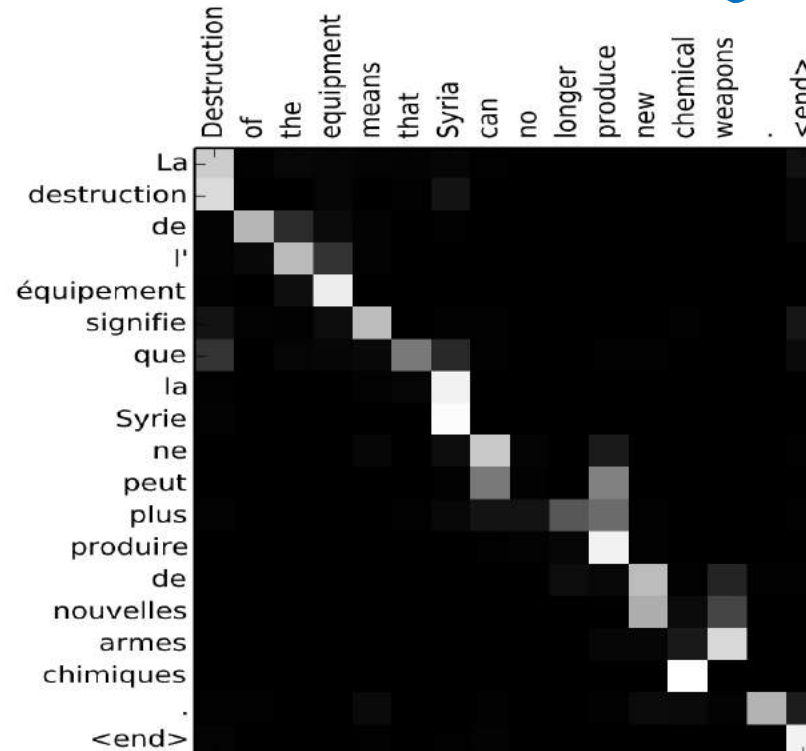
Andrew Ng

Attention examples

July 20th 1969 → 1969 – 07 – 20

23 April, 1564 → 1564 – 04 – 23

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





deeplearning.ai

Audio data

Speech recognition

Speech recognition problem

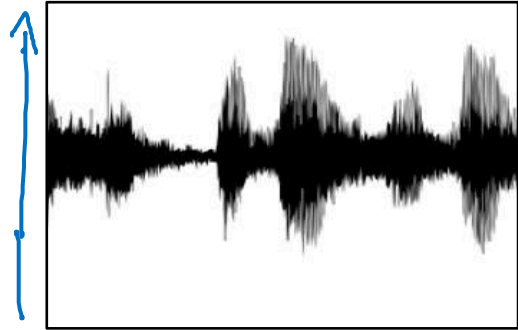
x

audio clip



y

transcript



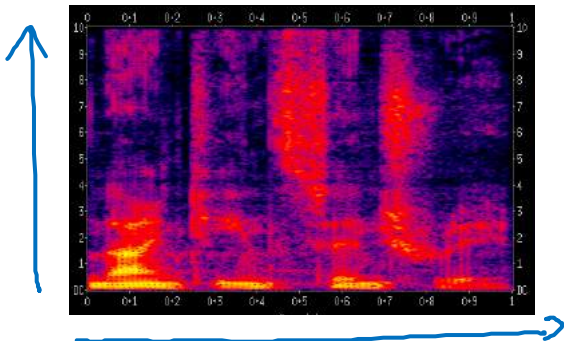
“the quick brown fox”

→ phonemes: de kwi braun

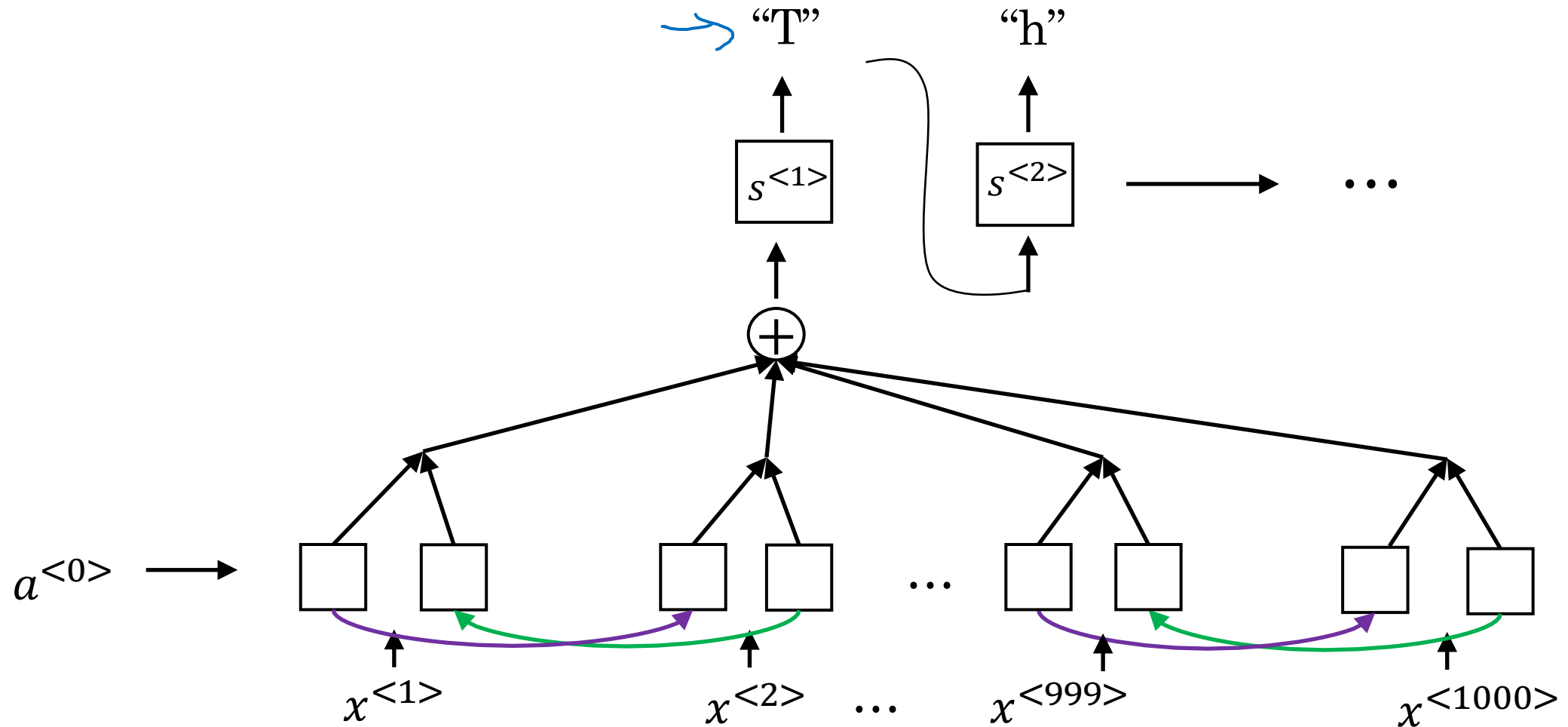
300h

3000h

100,000h

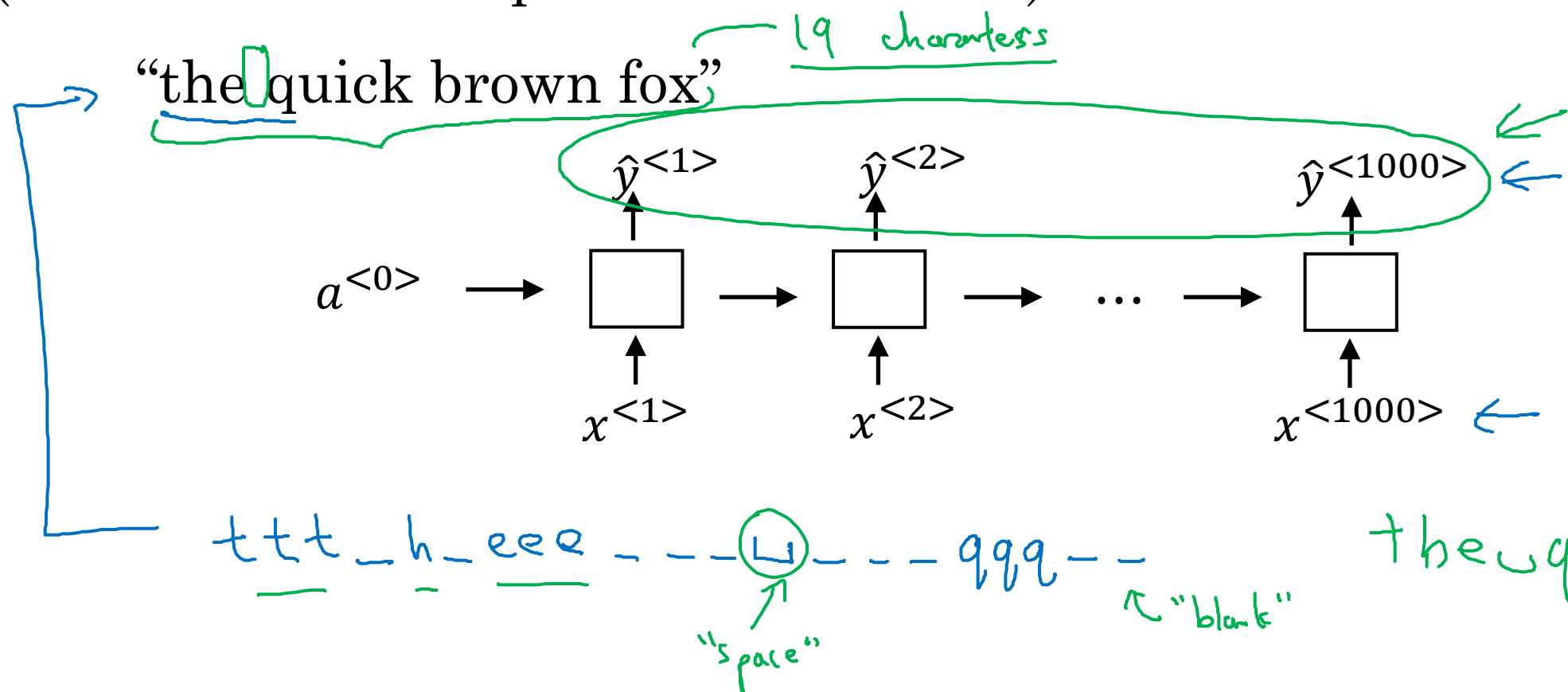


Attention model for speech recognition



CTC cost for speech recognition

(Connectionist temporal classification)



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

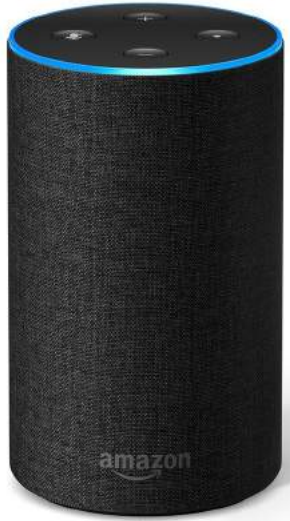


deeplearning.ai

Audio data

Trigger word
detection

What is trigger word detection?



Amazon Echo
(Alexa)



Baidu DuerOS
(xiaodunihao)

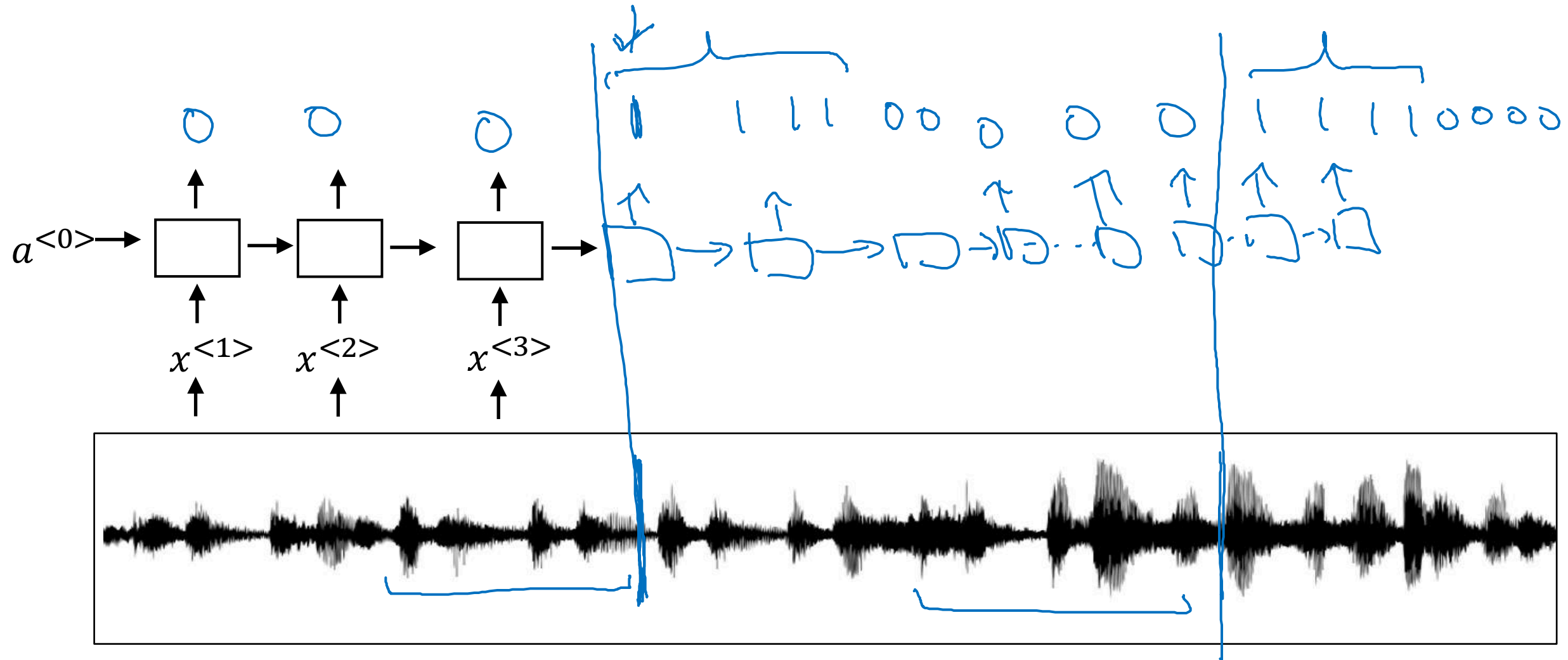


Apple Siri
(Hey Siri)



Google Home
(Okay Google)

Trigger word detection algorithm





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