deeplearning.ai

Sequence to sequence models

Basic models

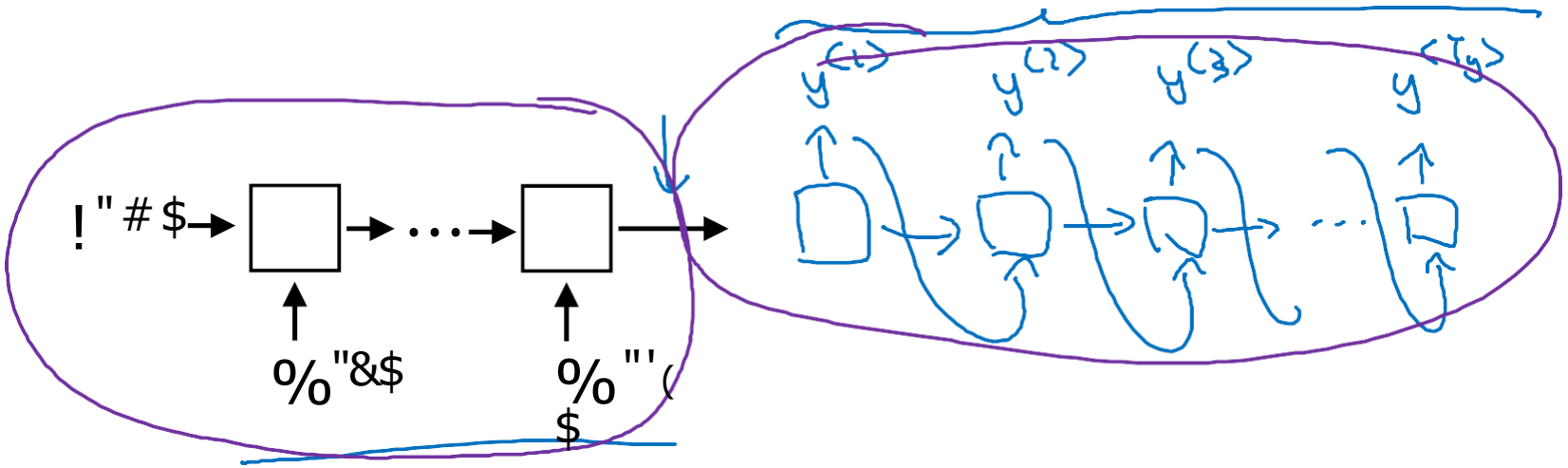
Sequence to sequence model

%"&$ %"*$ %"+$ %",$ %"-$

Jane visite l'Afrique en septembre

⟶ Jane is visiting Africa in September.

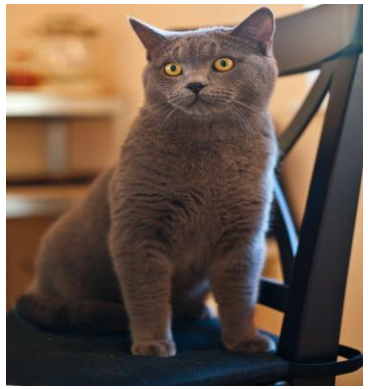
."&$."*$."+$."/$

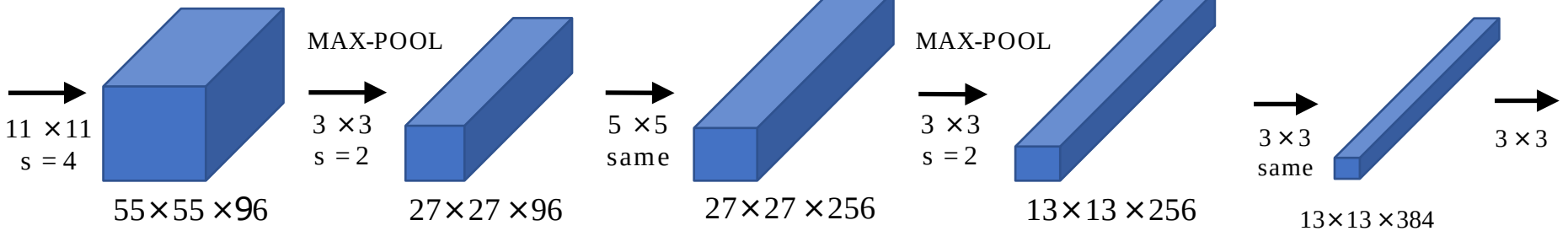


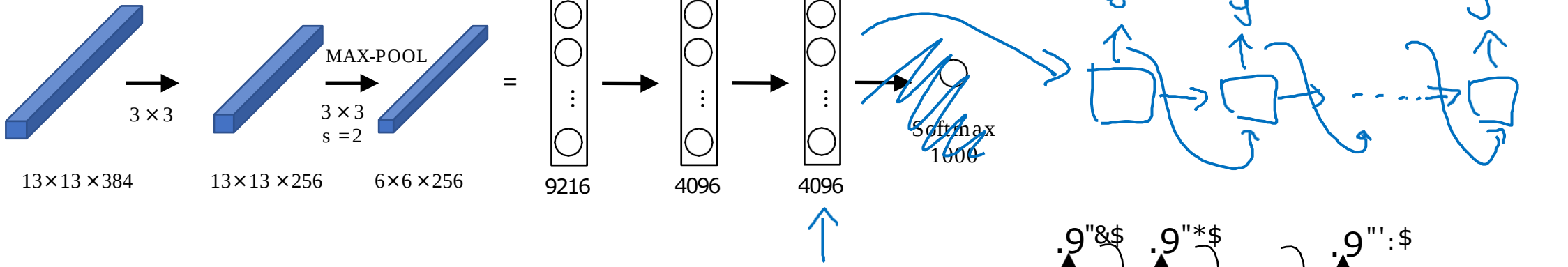
[Sutskever et al., 2014. Sequence to sequence learning with neural networks]

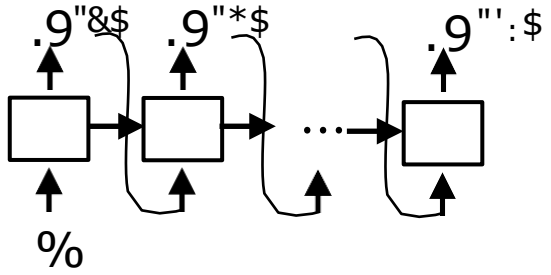
[Cho et al., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation]

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]

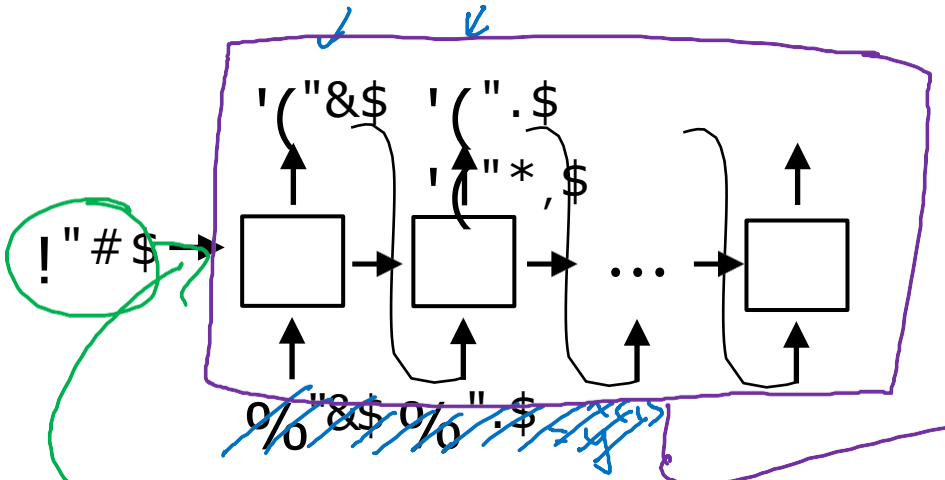
deeplearning.ai

Sequence to sequence models

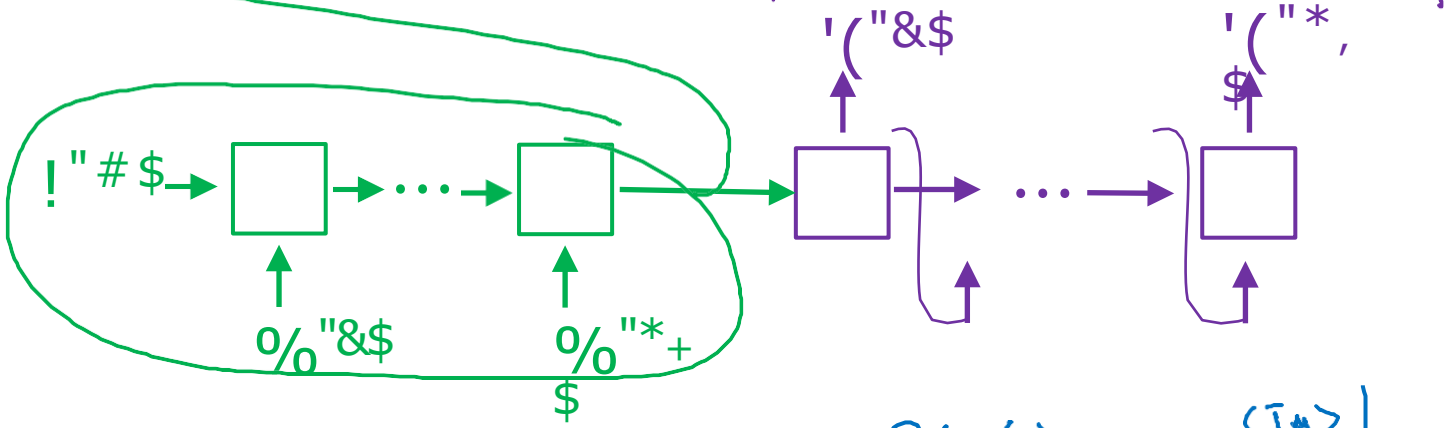
Picking the most likely sentence

Machine translation as building a conditional language model

Language model:



Machine translation:



"Conditional language model"

Andrew Ng

Finding the most likely translation

Jane visite l'Afrique en septembre.

$$/('"\&\$, \ldots, '"*, \$| \%)$$

English

French

→ Jane is visiting Africa in September.

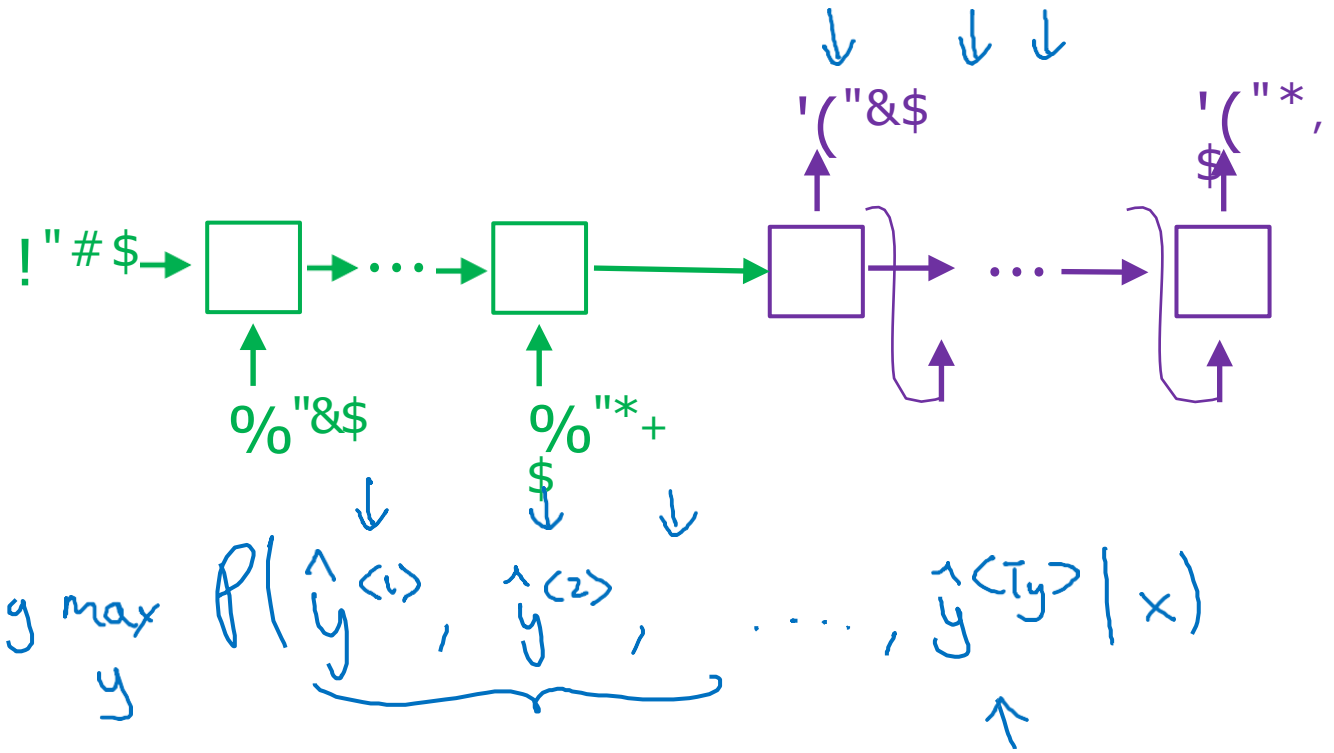
→ Jane is going to be visiting Africa in September.

→ In September, Jane will visit Africa.

→ Her African friend welcomed Jane in September.

$$\underset{:;<=,\ldots,:;>,=}{\arg\max} /('"\&\$, \ldots, '"*, \$| \%)$$

Why not a greedy search?



10,000

10

$10{,}000^{10}$

$P(y|x)$

→ 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

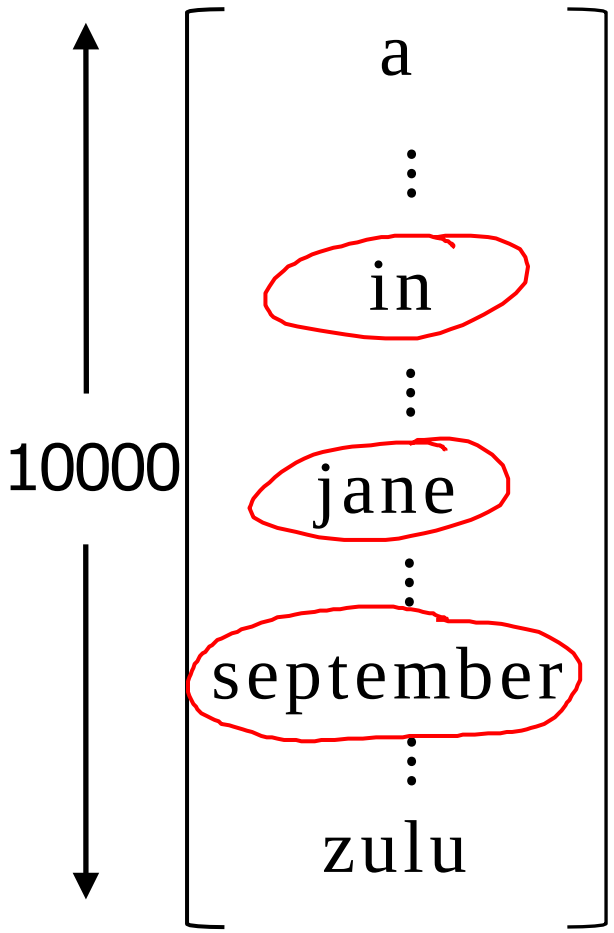
Sequence to sequence models

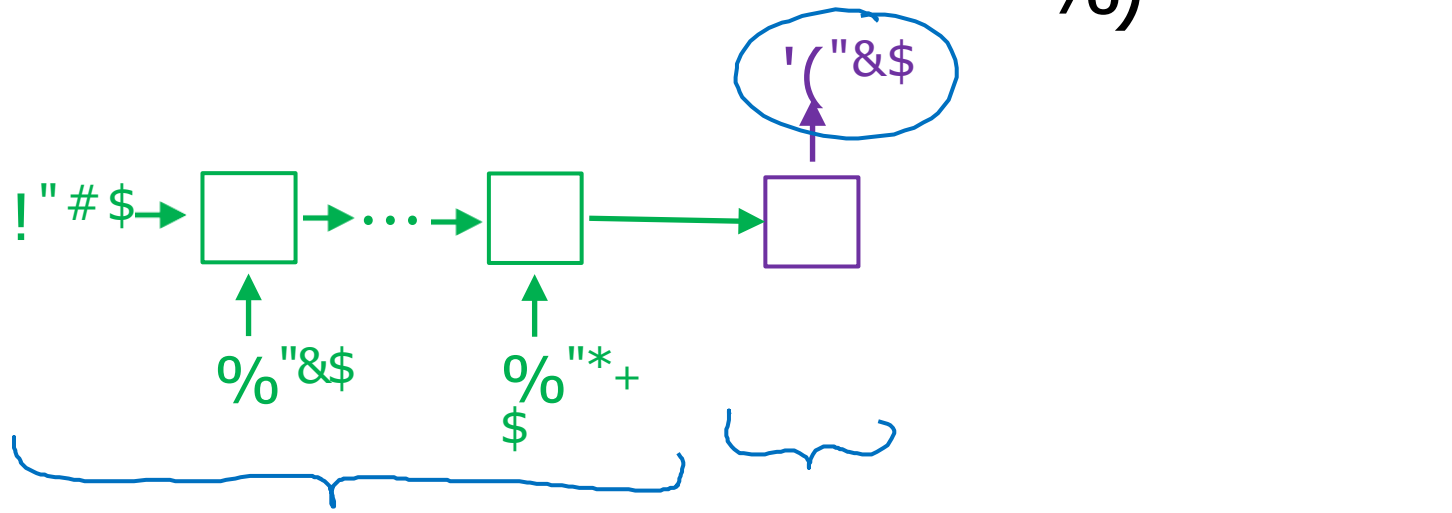
Beam search

Beam search algorithm

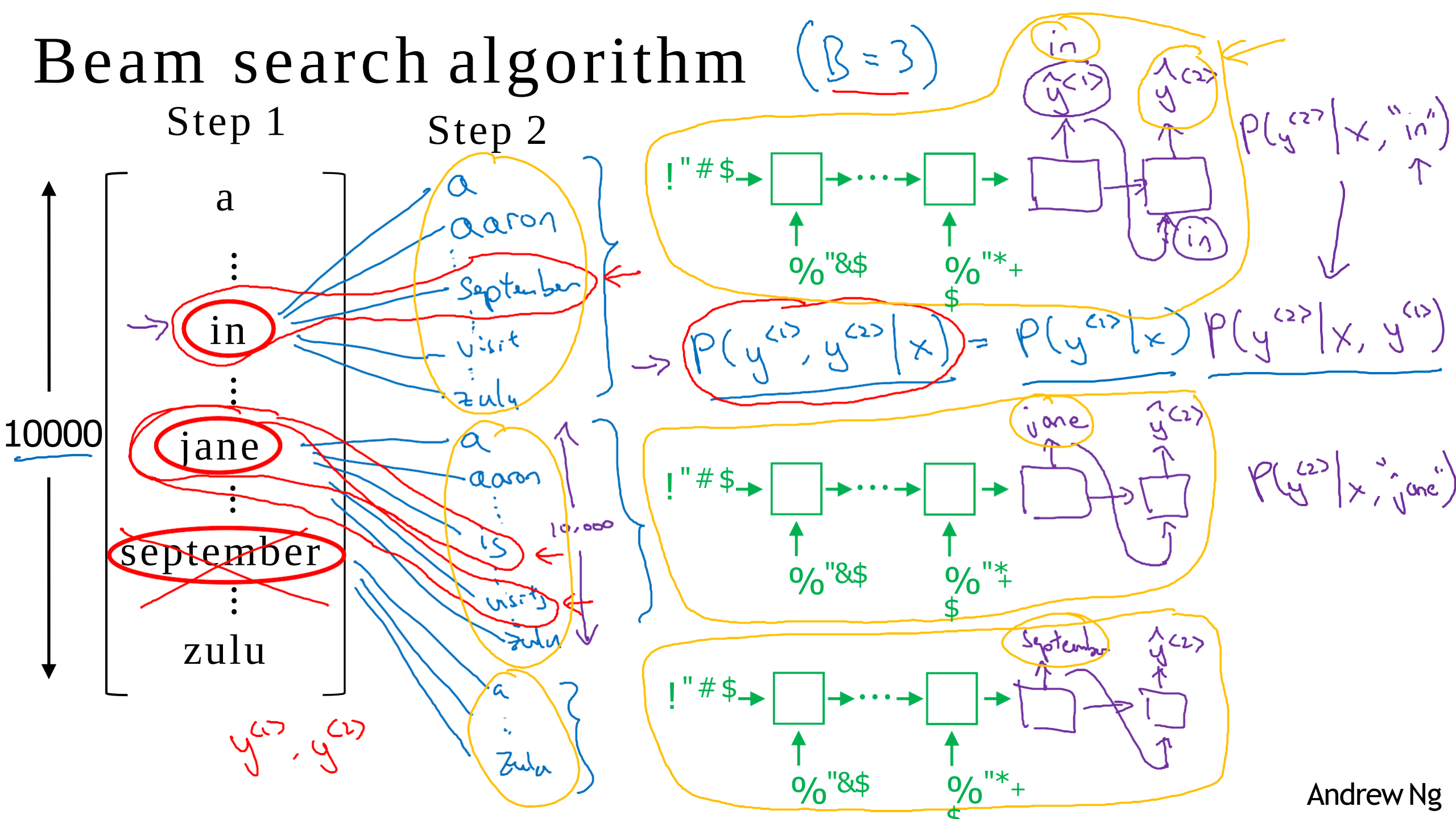
B = 3 (beam width)

Step 1





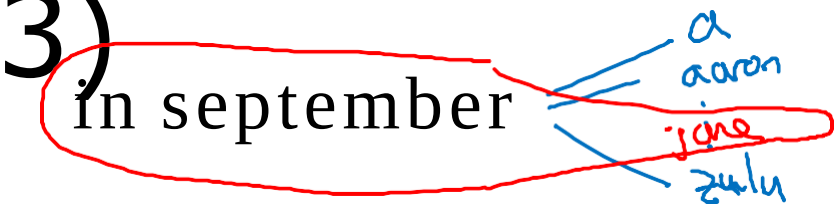
Beam search algorithm

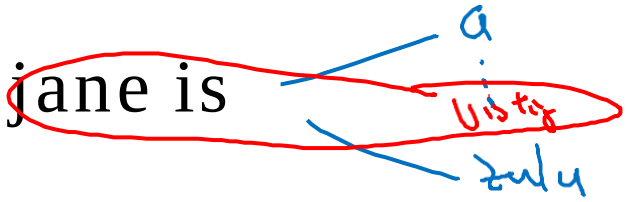


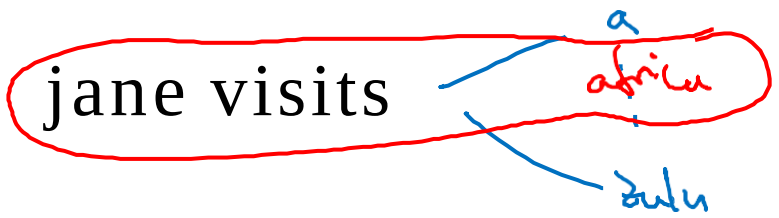
Andrew Ng

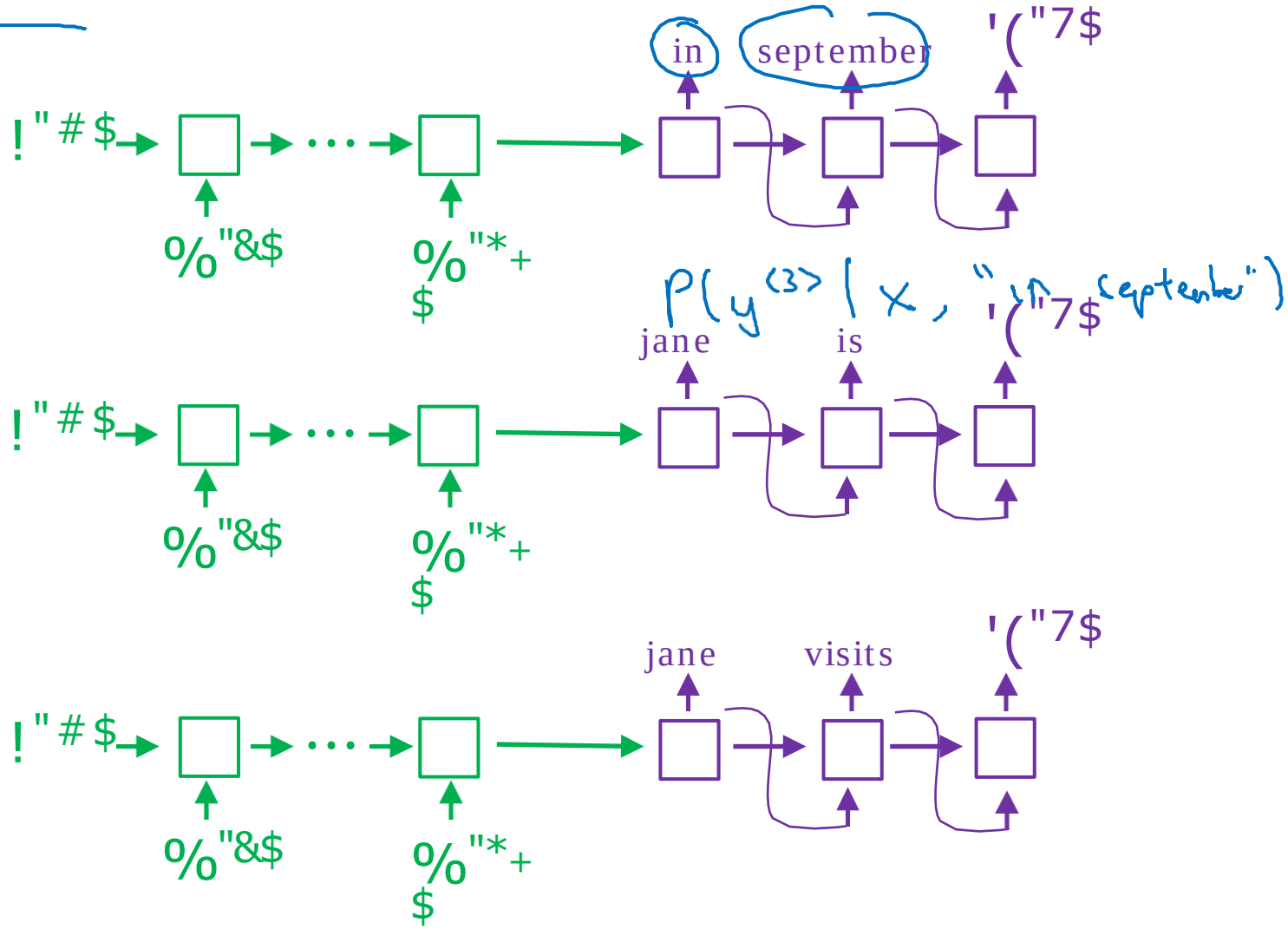
Beam search (4 = 3)

B = 1 ⇝ greedy search









0('"&$,'"9$|%)

jane visits africa in september. <EOS>

Andrew Ng

deeplearning.ai

Sequence to sequence models

Refinements to beam search

Length normalization

$$P(y^{\langle 1 \rangle} \cdots y^{\langle T_y \rangle} | x) = P(y^{\langle 1 \rangle} | x)\, P(y^{\langle 2 \rangle} | x, y^{\langle 1 \rangle}) \cdots P(y^{\langle T_y \rangle} | x, y^{\langle 1 \rangle}, \ldots, y^{\langle T_y - 1 \rangle})$$

$$\arg\max_{y} \prod_{t=1}^{T_y} P(y^{\langle t \rangle} | x, y^{\langle 1 \rangle}, \ldots, y^{\langle t-1 \rangle})$$

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

$$P(y|x)$$

$$\arg\max_{y} \sum_{t=1}^{T_y} \log P(y^{\langle t \rangle} | x, y^{\langle 1 \rangle}, \ldots, y^{\langle t-1 \rangle})$$

$y = 1, 2, 3, \ldots, 30.$

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

$\alpha = 0.7$

$\alpha = 1$

$\alpha = 0$

Andrew Ng

Beam search discussion

large B: better result, slower

Small B: worse result, faster

Beam width B?

1 -> 3 -> 10, 100, 1000, -> 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

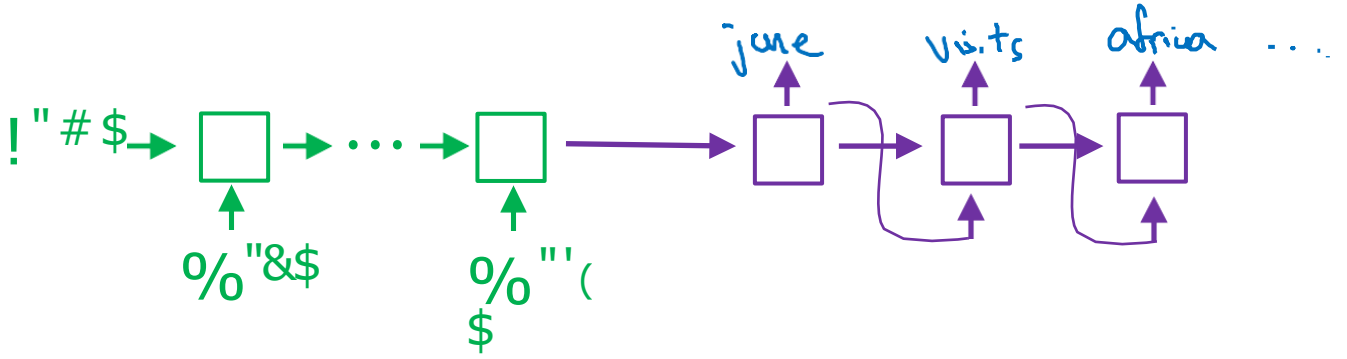
BT

Jane visite l'Afrique en septembre.

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

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

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



Error analysis on beam search

$P(y^*|x)$

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

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

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

Case 1: $P(y^*|x) > P(\hat{y}|x)$ ←

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

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

Conclusion: Beam search is at fault.

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

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	$P(y^* \mid x)$	$P(y^*) / P(y)$	At fault?
Jane visits Africain 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

Andrew Ng

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]

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.

$p_1 = \frac{\sum_{unigram \in y} Count_{clip}(unigram)}{\sum_{unigram \in \hat{y}} Count(unigram)}$

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

$p_1, p_2, \ldots = 1.0$

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

Bleu details

!=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

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

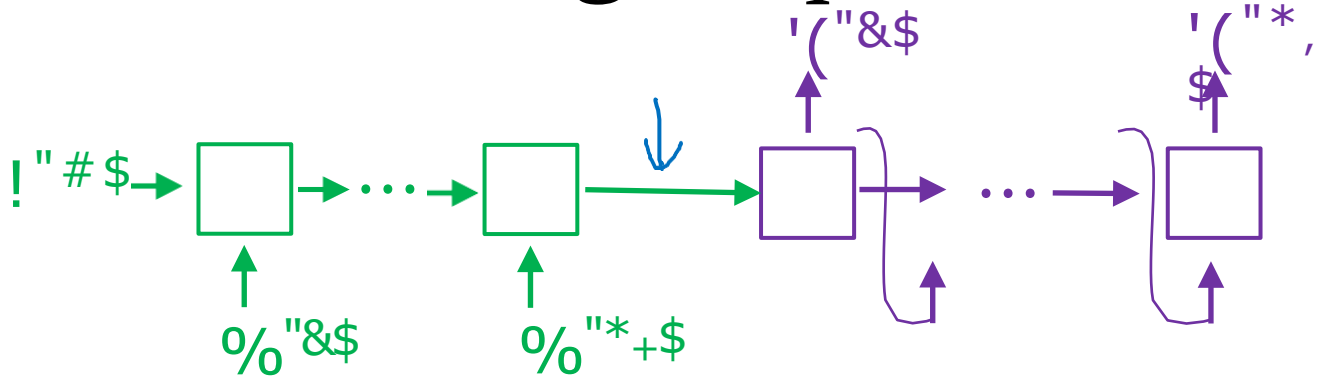
[Papineni et. al., 2002. Bleu: A method for automatic evaluation of machine translation]

deeplearning.ai

Sequence to sequence models

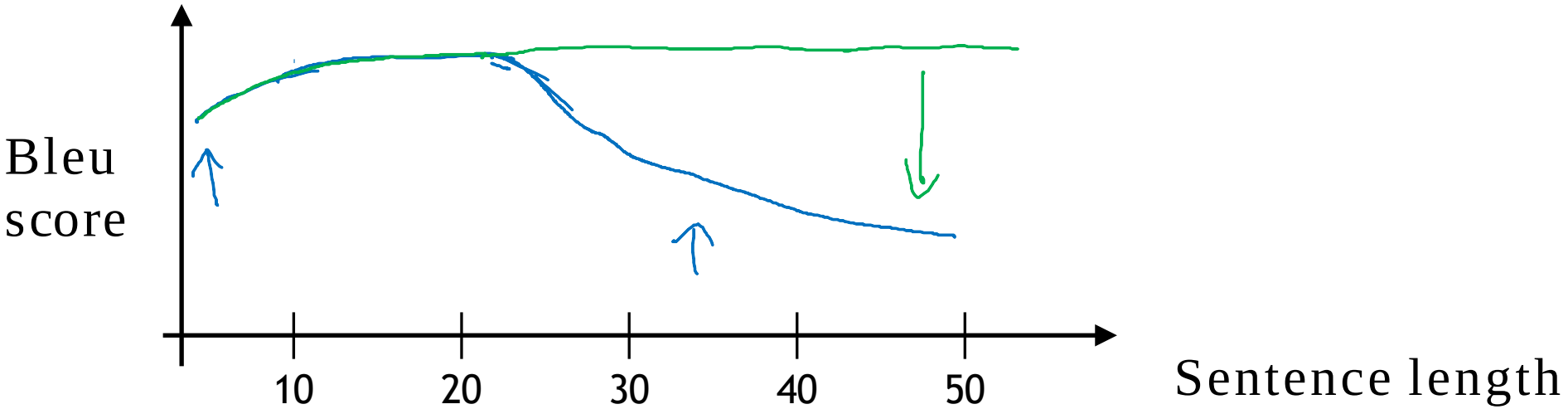
Attention model intuition

The problem of long sequences

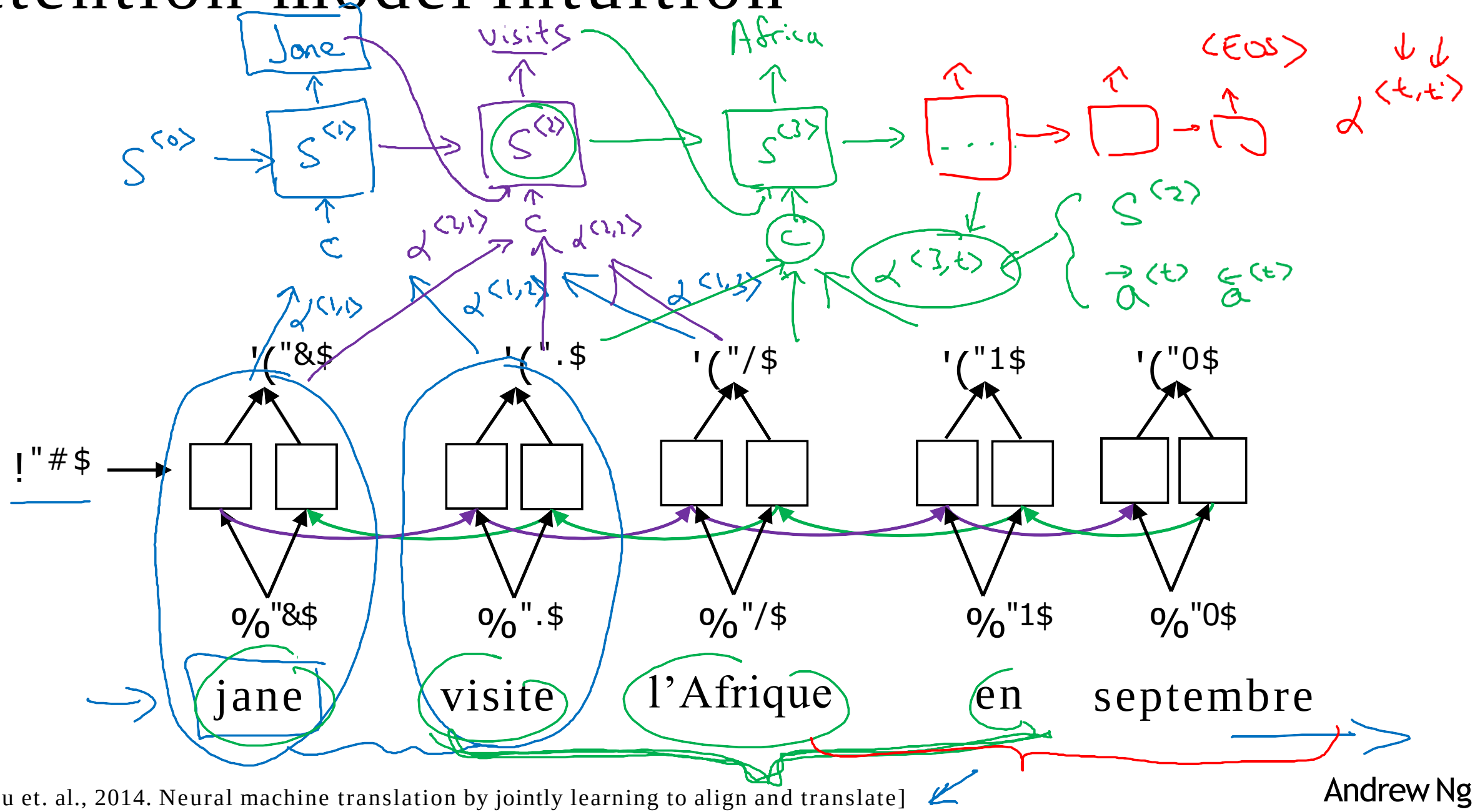


Jane s'est rendue en Afrique en septembre dernier, a apprécié la culture et a rencontré beaucoup de gens merveilleux; elle est revenue en parlant comment son voyage était merveilleux, et elle me tente d'y aller aussi.

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



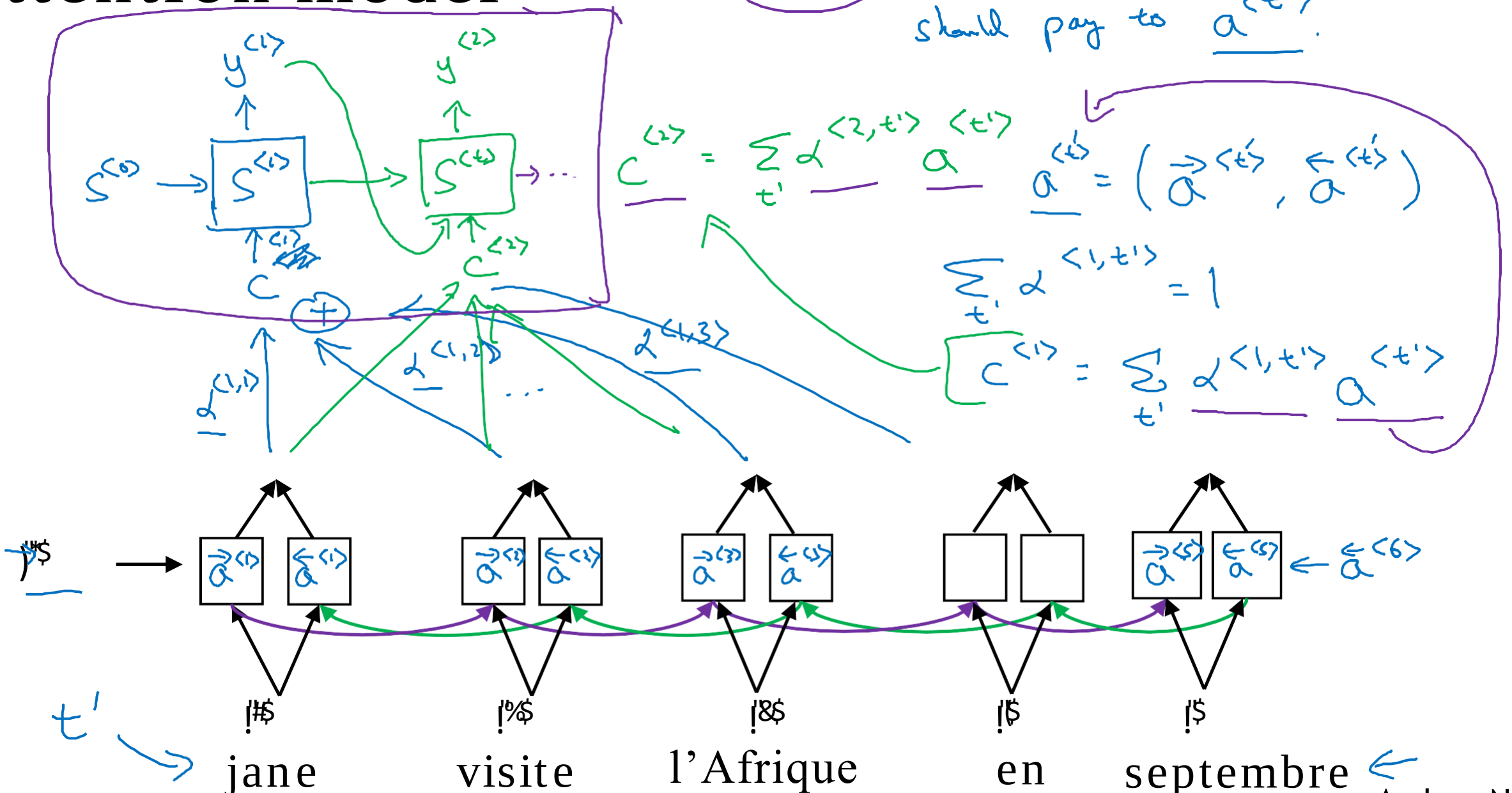
[Bahdanau et. al., 2014. Neural machine translation by jointly learning to align and translate]

deeplearning.ai

Sequence to sequence models

Attention model

Attention model

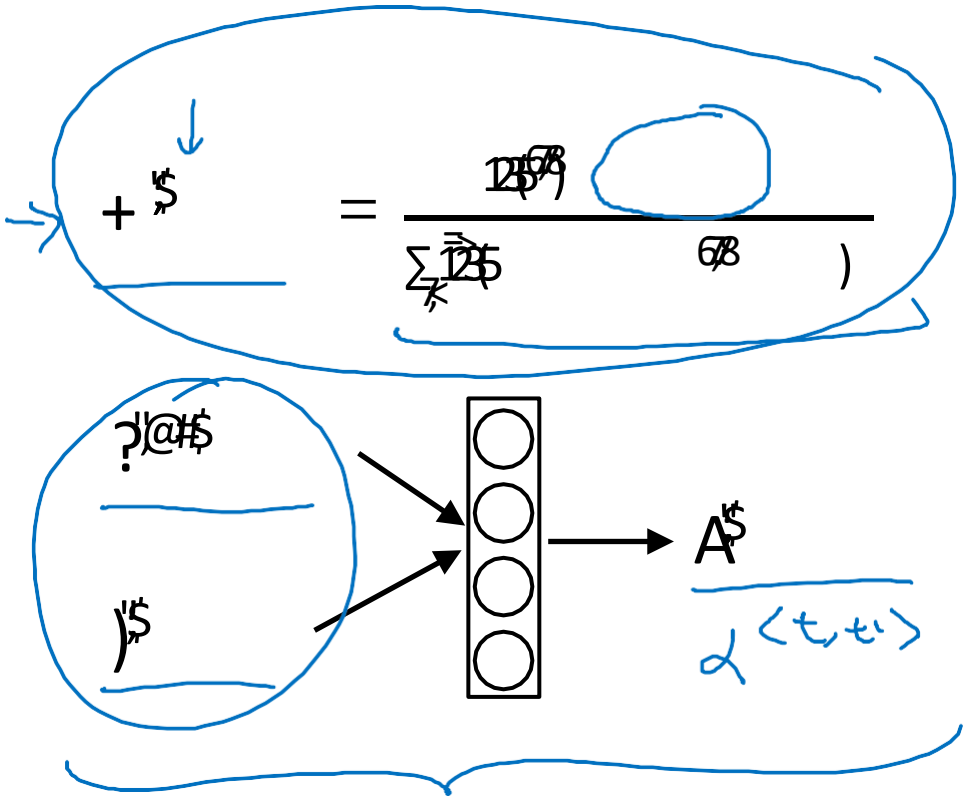


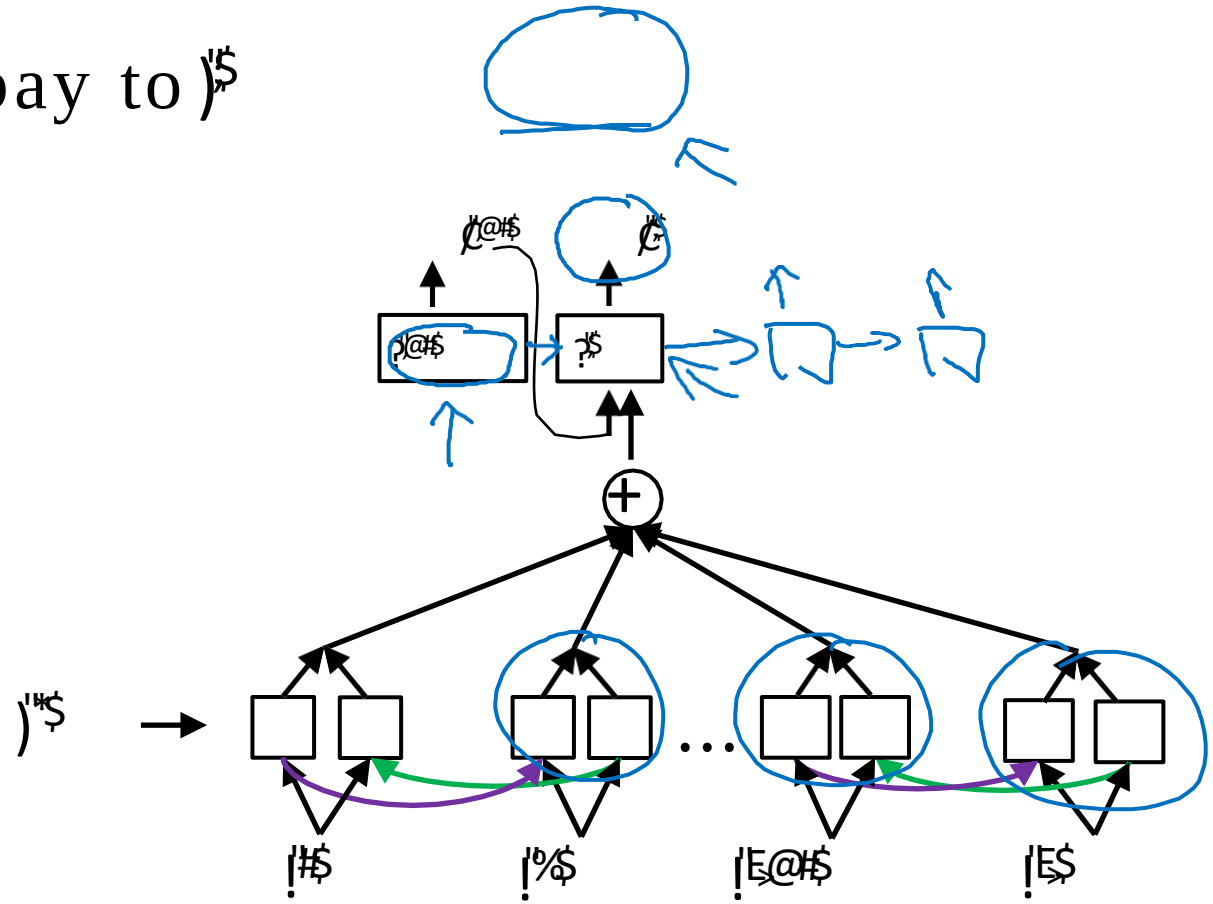
[Bahdanau et. al., 2014. Neural machine translation by jointly learning to align and translate]

Computing attention

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

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





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

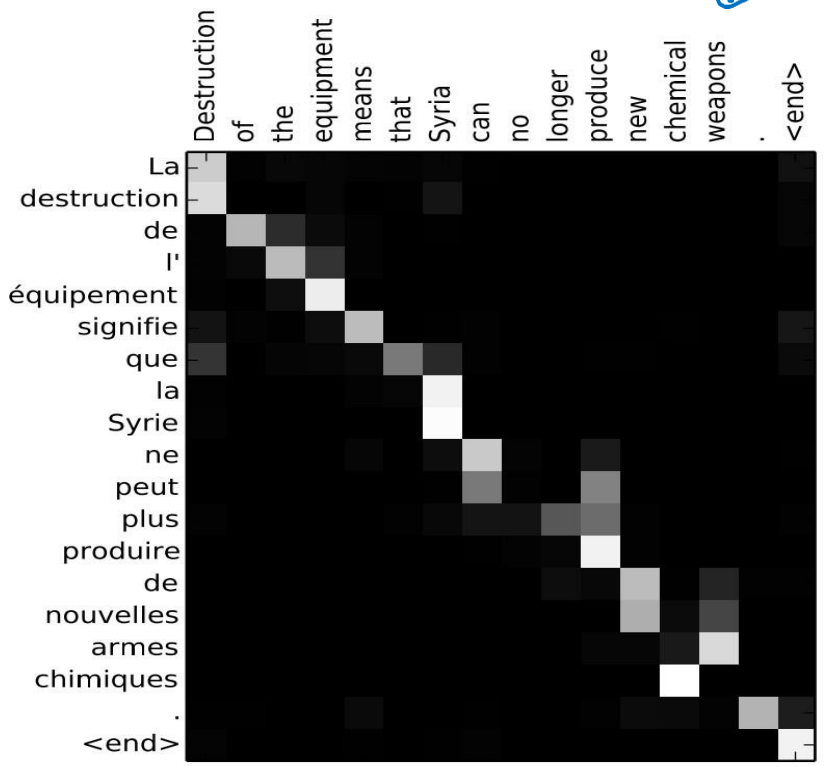
[Xu et. al., 2015. Show, attend and tell: Neural image caption generation with visual attention]

Attention examples

July 20th 1969 → 1969-07-20

23 April, 1564 → 1564-04-23

Visualization of $\alpha^{\langle t,t' \rangle}$:



deeplearning.ai

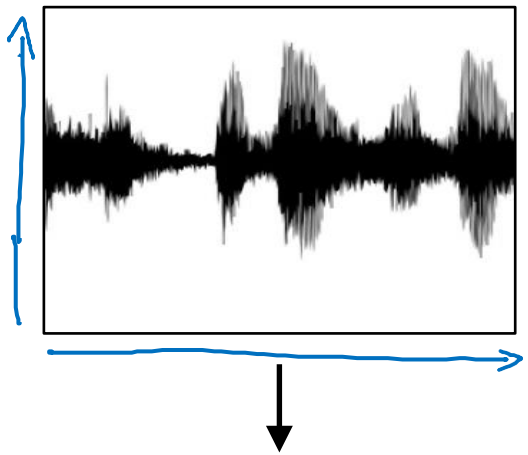
Audio data

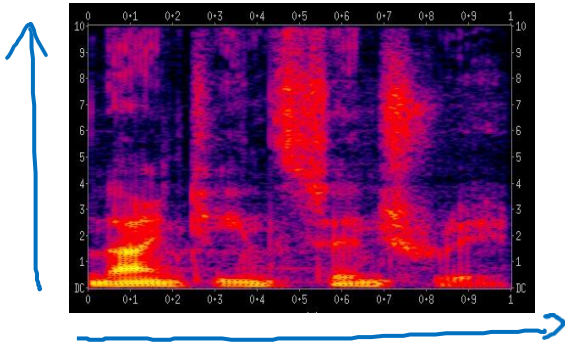
Speech recognition

Speech recognition problem

!

audio clip





#

transcript

"the quick brown fox"

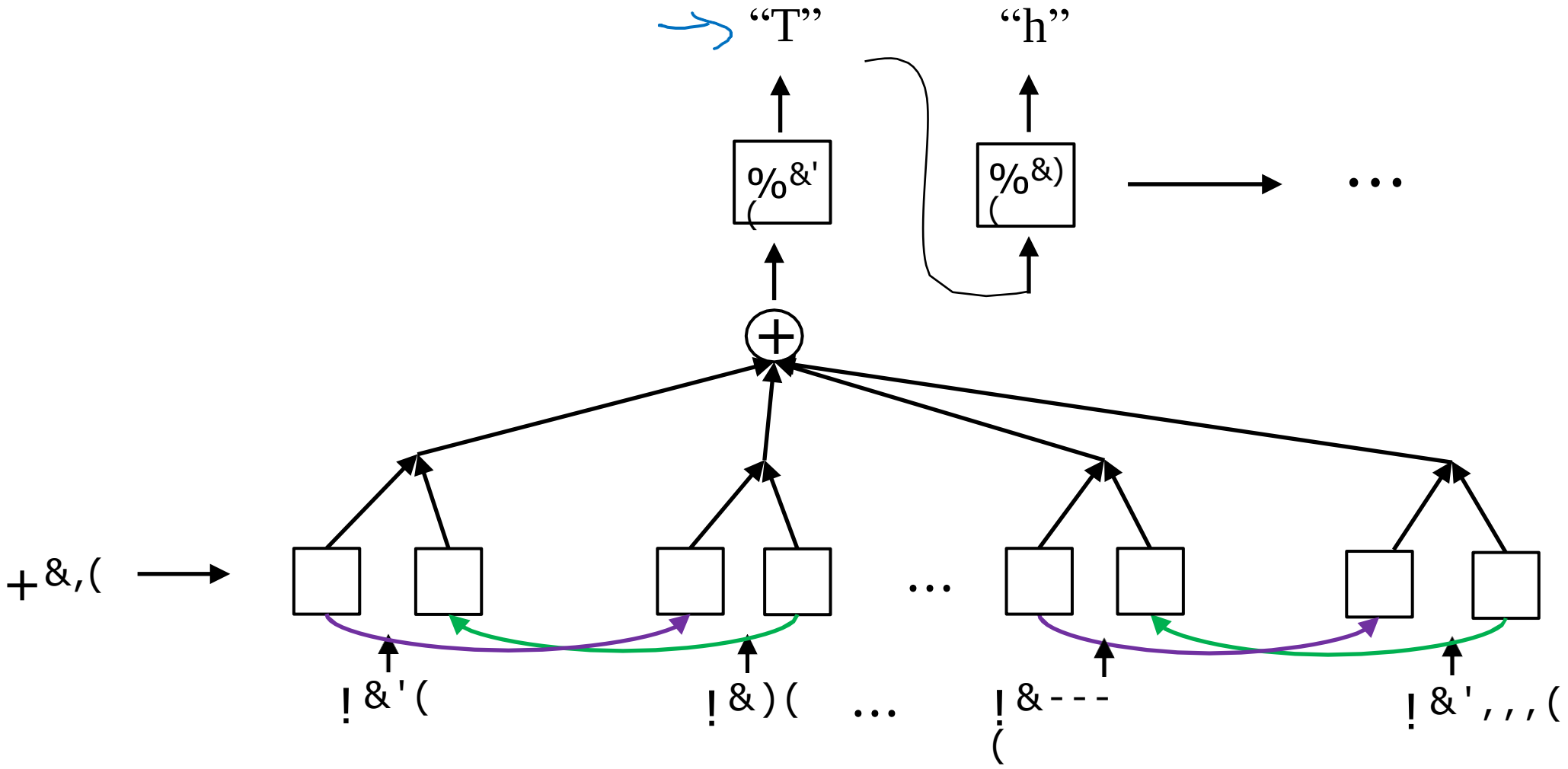
phonemes: de kwik braun

300h

3000h

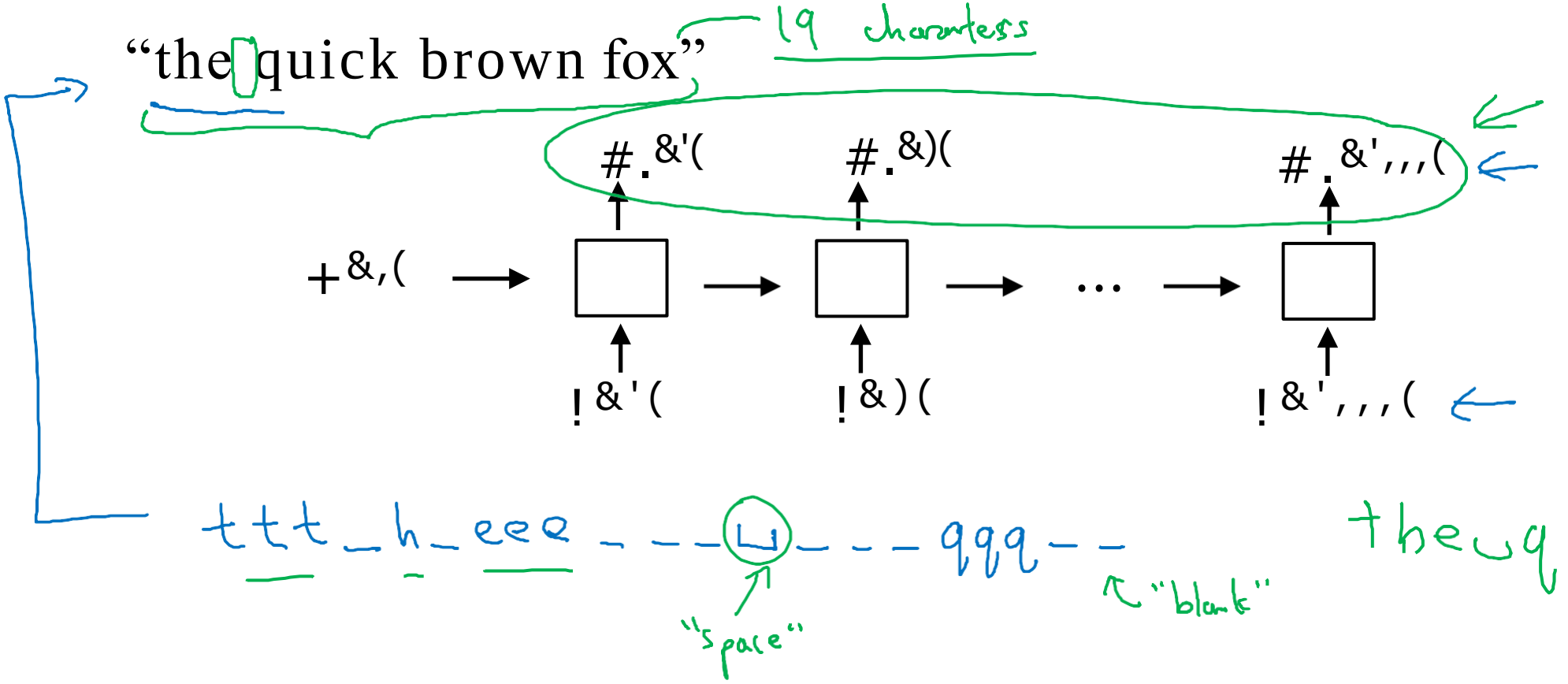
100,000h

Attention model for speech recognition



CTC cost for speech recognition

(Connectionist temporal classification)



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

[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?

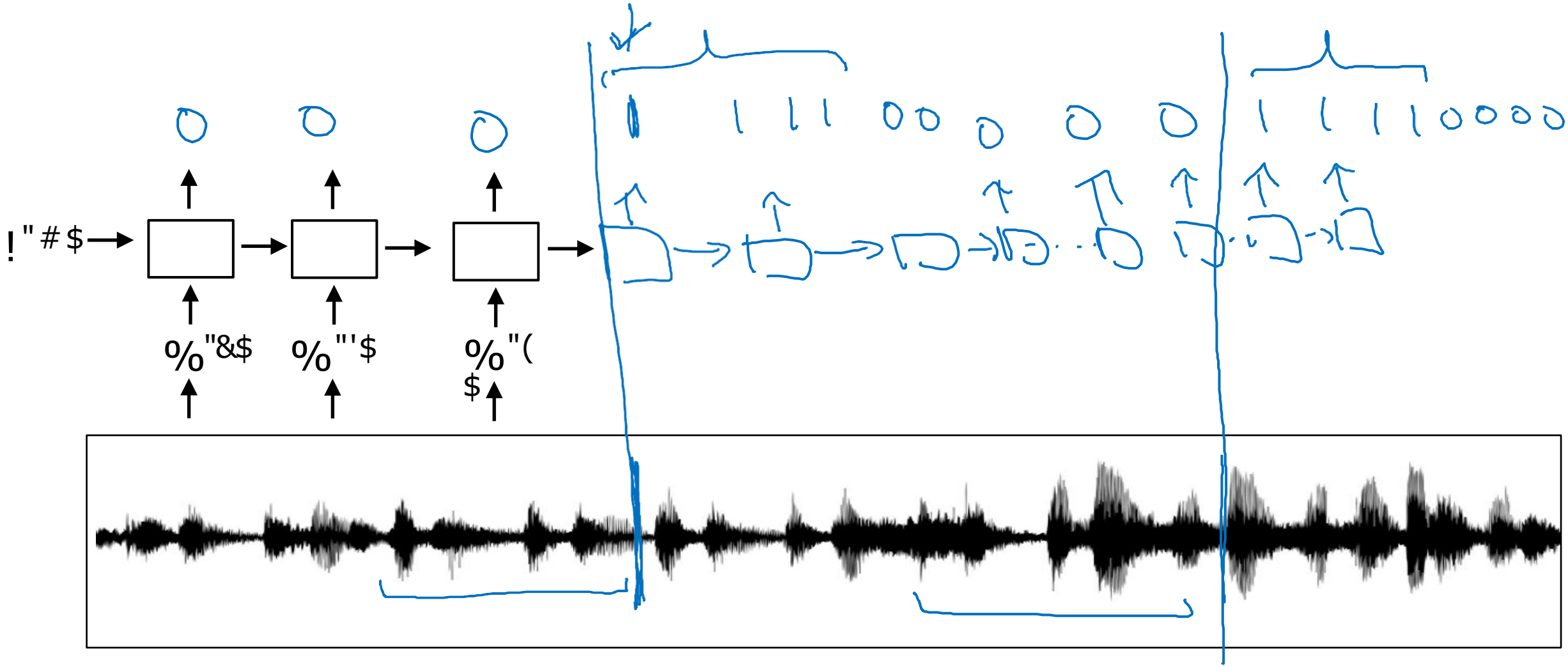
Amazon Echo
(Alexa)

Baidu DuerOS
(xiaodunihao)

Apple Siri
(Hey Siri)

Google Home
(Okay Google)

Trigger word detection algorithm



Andrew Ng