

Handling Sequences

Notes on sequence modeling

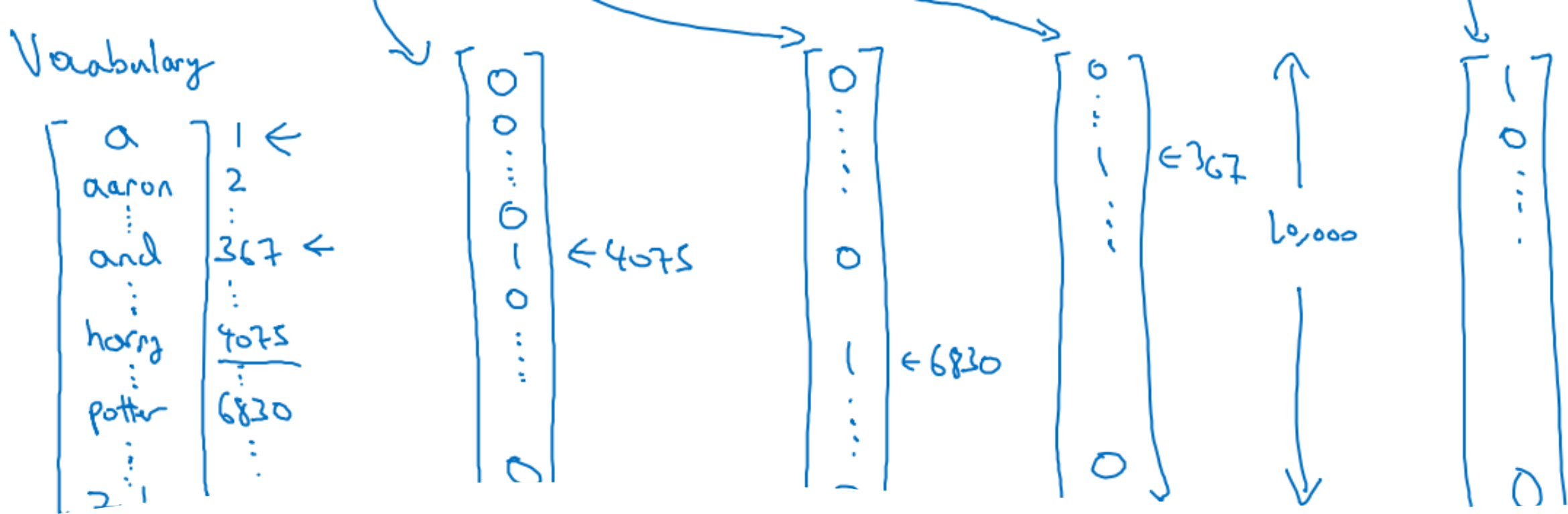
Zabeb

Why we need sequence Models

- Sentiment analysis
- Music Generation
- Speech Recognition
- DNA sequence analysis
- Machine translation
- Video activity recognition
- Name entity recognition

Some Notations

- $X^1 X^2 X^N$ are token to model upon
- T_x = some number is the size of a sequence
- Assume we are modeling upon the x : Harry Potter and Hermi Granger invented a new spell.
- We need to make Name entity recognition
- The label is $Y^1 = 1, Y^2=1, Y^3=0 \dots$
- $T_y = 9$ is the size of label data

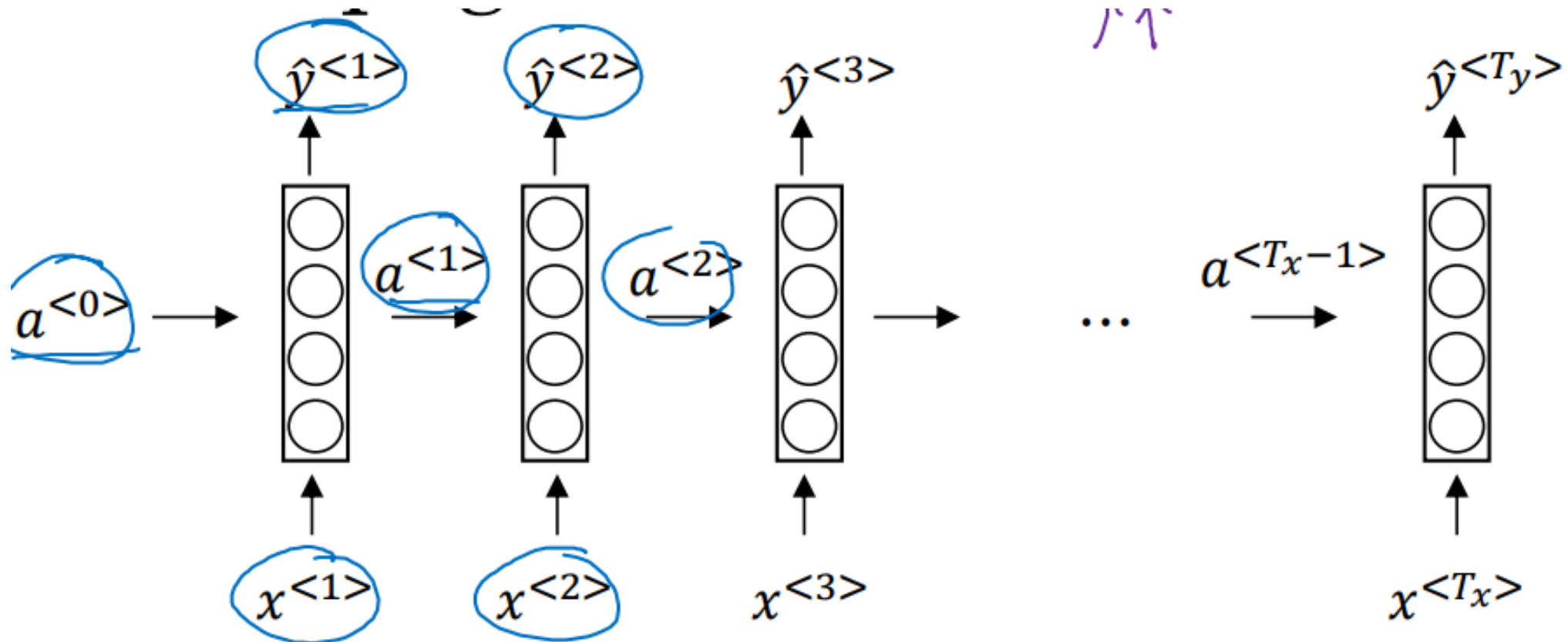


Representing Words

- One way is to make a vocabulary dictionary and then number the words
- Then for each word in the sequence assign a one hot in the vectors so it represent the word
- This is named one hot encoding
- See above a representation

Why we need RNN ?

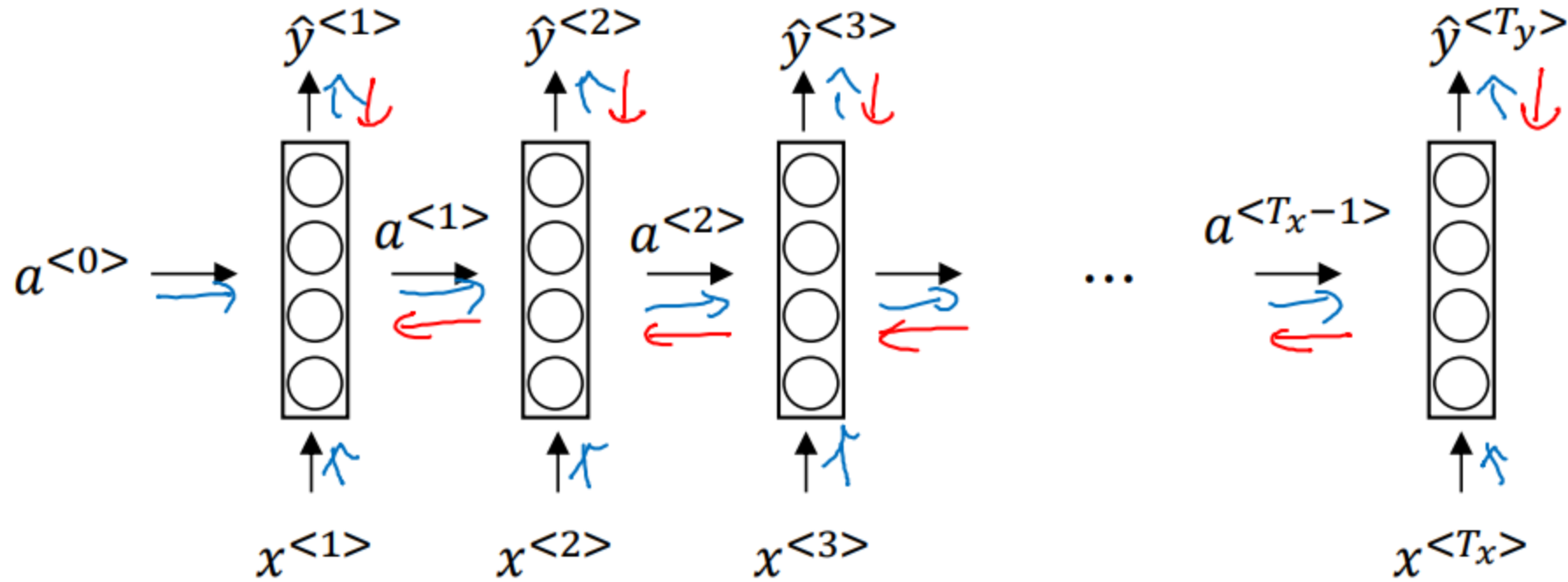
- We need to pass different length inputs and outputs different length output
- We need to share different features through the network about different positions in the sequence



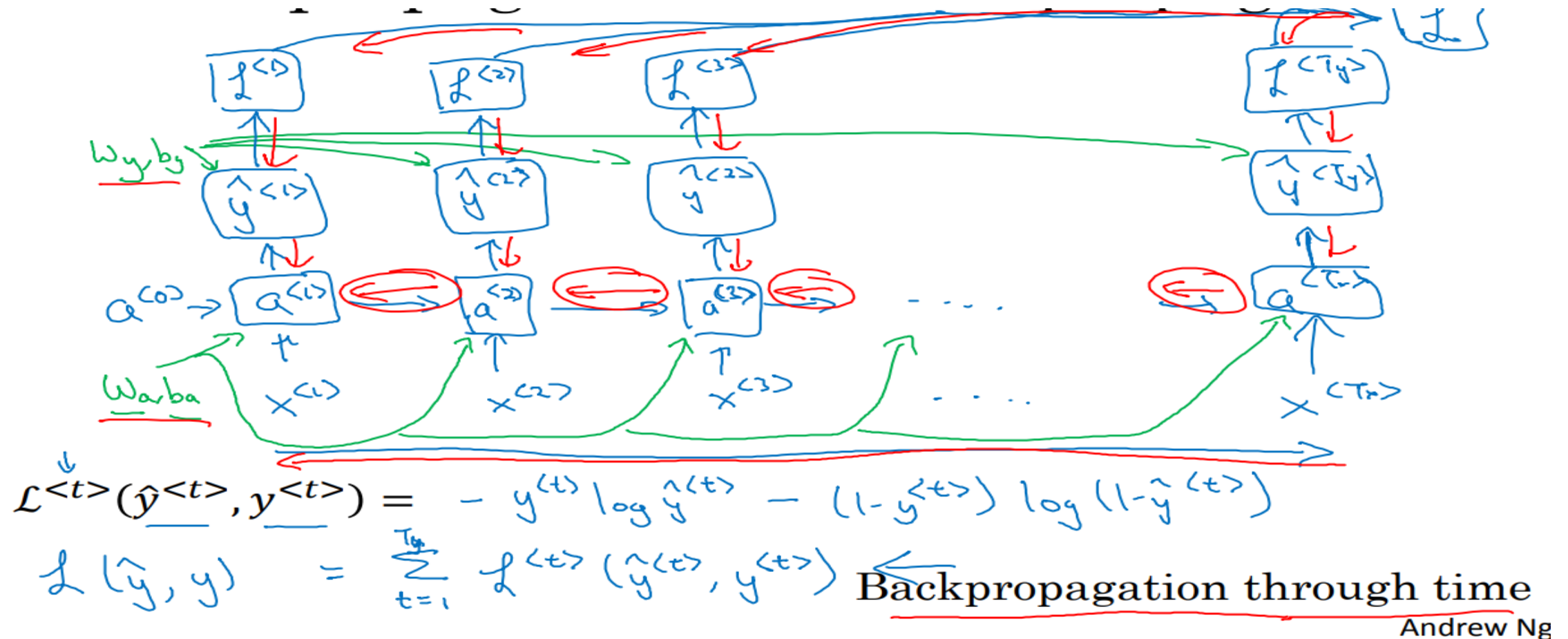
RNN Notation

- $a_{<t>} = g(W_a a_{<t-1>} + W_{ax} x_{<t>} + b_a)$
- $y_{<t>} = g(W_y a_{<t>} + b_y)$

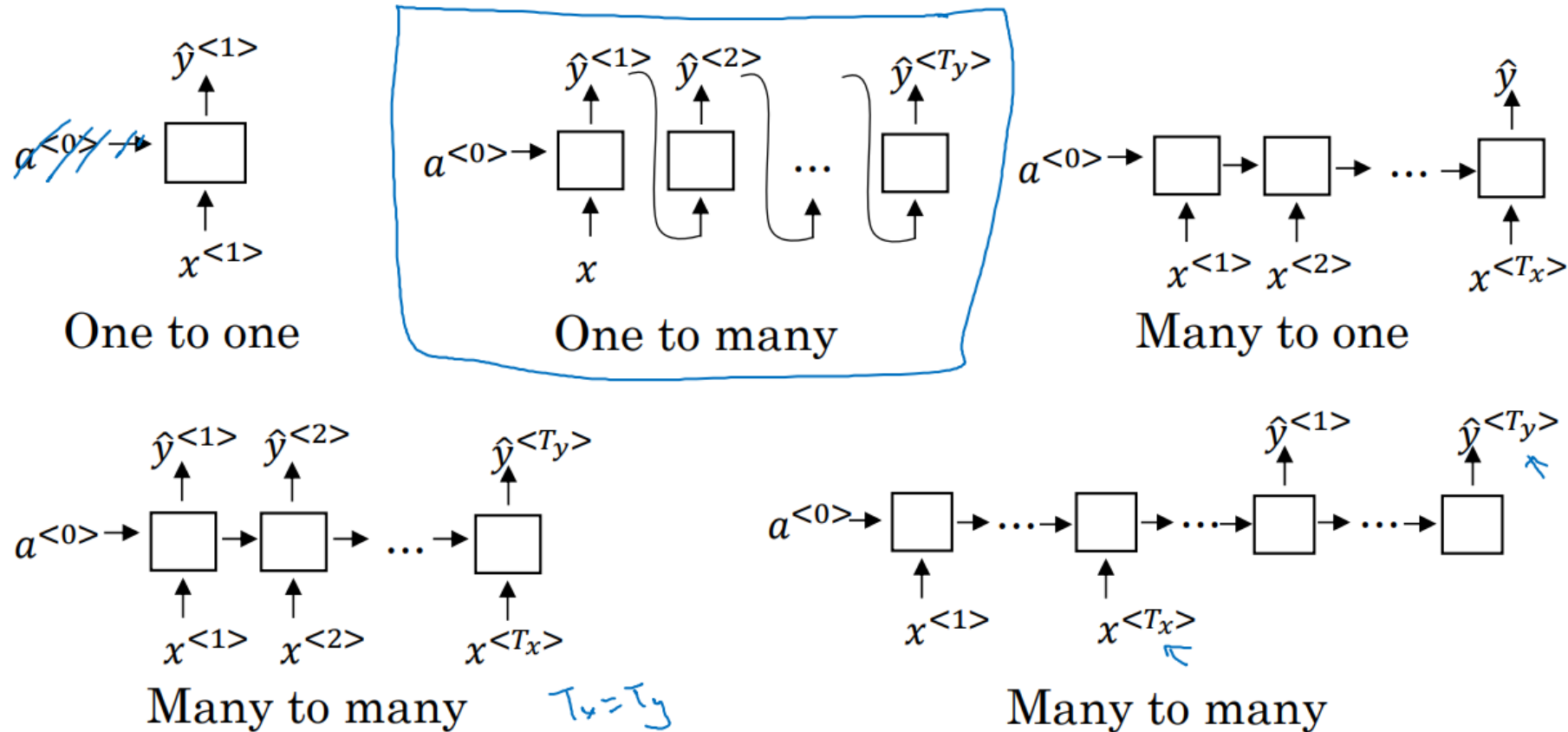
Forward propagation and backpropagation



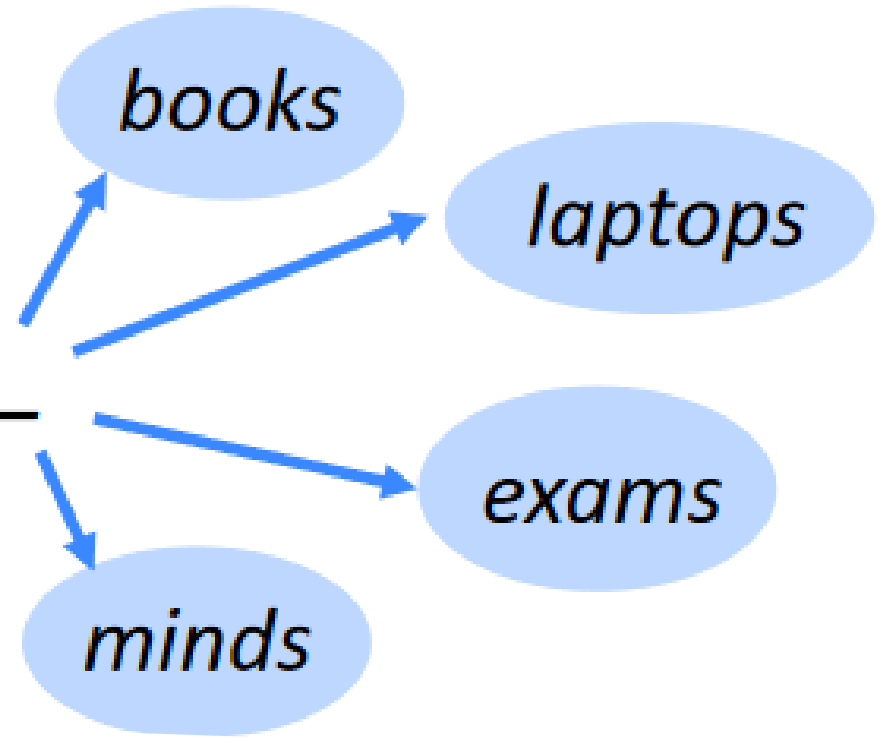
Backpropagation through time



Different Types of RNN



the students opened their _____



What is Language
Modeling ?

- The apple juice --- $> 3.5 \times 10^{-2}$
- The apple pair --- $> 3.1 \times 10^{-10}$
- We ask in Speech recognition what is the probability of the given Output given the sentence

Language modelling with an RNN

- Training set: large corpus of English text.
- We need a model to write : Cats average 15 hours of sleep a day.
- Make three special tokens the <SOS> <EOS> <UNK>
- Make each token as the input output in other words $x^t = y^{t-1}$
- Feed to RNN <SOS> along with dummy state and then output y^1
- Take Y^1 is the x^1 so compute the state accordingly and make y^2

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

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

Sampling Novel sequences

- Character-level language model
- Sampling a sequence from a trained RNN

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.

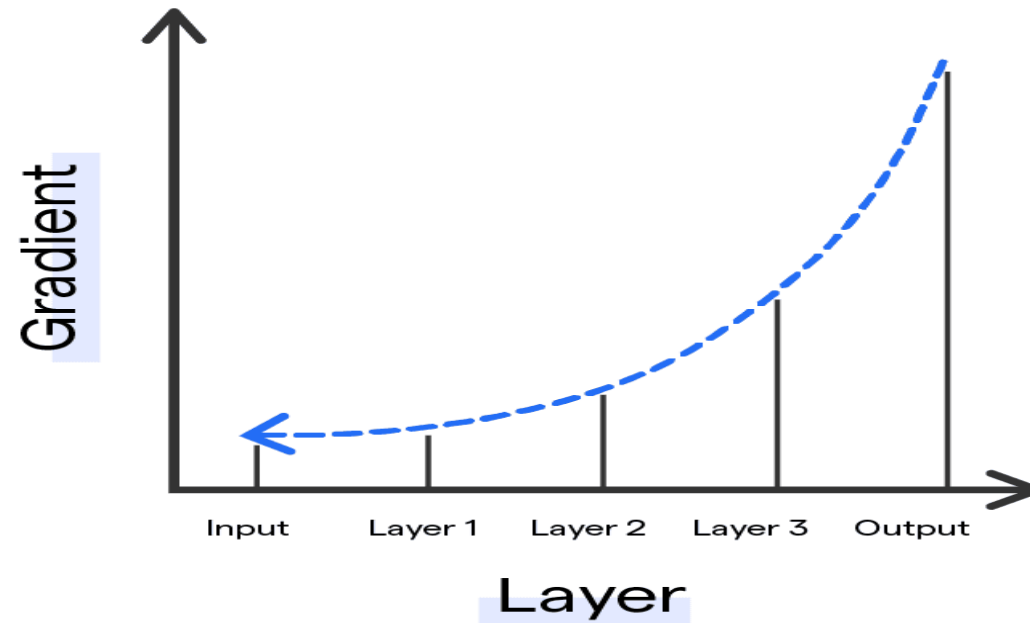
And subject of this thou art another this fold.

When besser be my love to me see sabl's.

For whose are ruse of mine eyes heaves.

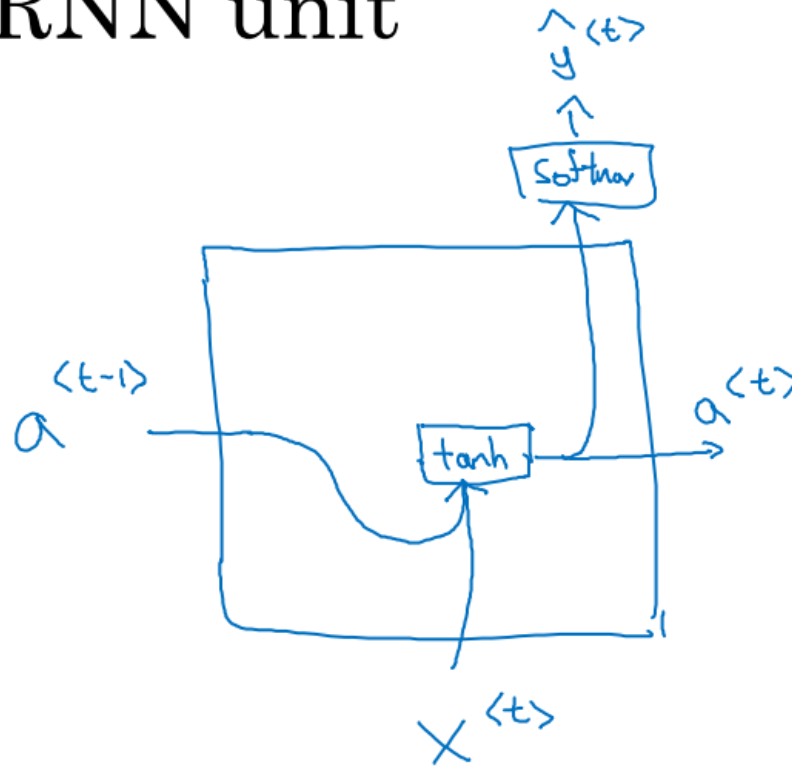
Vanishing gradients with RNNs

Vanishing Gradient Problem



Gated Recurrent Unit (GRU) - recall slide

RNN unit



Gated Recurrent Unit (GRU)

- $\tilde{c}_{<t>} = \tanh(W_c [c_{<t-1>}, x_{<t>}] + b_c)$
- $\Gamma_u = \sigma(W_u c_{<t-1>}, x_{<t>} + b_u)$
- $c_{<t>} = \Gamma_u * \tilde{c}_{<t>} + (1 - \Gamma_u) c_{<t-1>}$

LSTM (long short-term memory) unit

- $\tilde{c}_{<t>} = \tanh(W_c a_{<t-1>}, x_{<t>} + b_c)$
- $\Gamma_u = \sigma(W_u a_{<t-1>}, x_{<t>} + b_u)$
- $\Gamma_f = \sigma(W_f a_{<t-1>}, x_{<t>} + b_f)$
- $\Gamma_o = \sigma(W_o a_{<t-1>}, x_{<t>} + b_o)$
- $c_{<t>} = \Gamma_u * \tilde{c}_{<t>} + \Gamma_f * c_{<t-1>}$
- $a_{<t>} = \Gamma_o * c_{<t>}$

Talking GRU

- At each computation step we will compute the candidate for new state the $\tilde{c}_{<t>}$
- But we will leave the choice for the network decide upon updating the current $c_{<t>}$ by $c_{<t>} = \tilde{c}_{<t>}$ or just pass the past state $c_{<t>} = c_{<t-1>}$
- The way in which we control this by learning a parameter Γ_u this parameter will be learned from $c_{<t-1>}$ and $x_{<t>}$ and it will be sigmoid activated

Talking Long Short-Term Memory

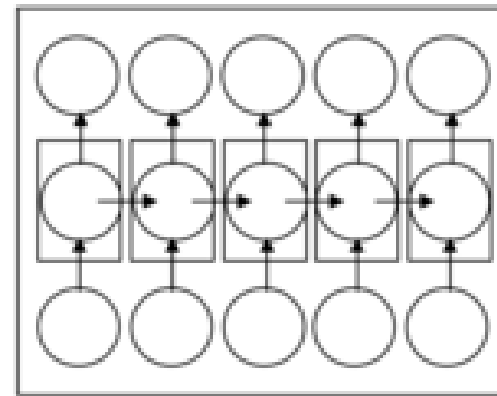
- Here its somehow different but same approach
- We will try to learn 3 parameters from $a_{<t-1>}$ and $x_{<t>}$, Γu , Γf , and Γo the first will control taking

$c_{\sim <t>}$ and one that will control forgetting the $c_{\sim <t>}$

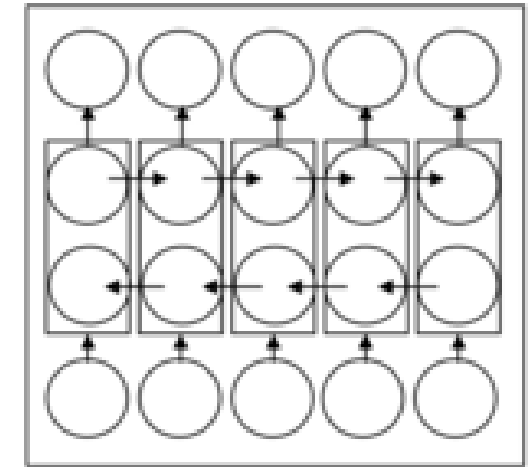
So, taking $a_{<t-1>}$ and one to make $a_{<t>}$ more flexible so that $a_{<t>} = \Gamma o c_{<t>}$, whatever $c_{<t>}$ is.

Getting Information from the Future

- He said, “Teddy bears are on sale!”
- He said, “Teddy Roosevelt was a great President!”
- Bidirectional RNN (BRNN) is an acyclic graph
- Can be with or without the GRU or LSTMs



(a)



(b)

Structure overview

(a) unidirectional RNN

(b) bidirectional RNN

Basic models

- Sequence to sequence model
- $x^{<1>} x^{<2>} x^{<3>} x^{<4>} x^{<5>}$
- Jane visite l'Afrique en septembre.
- Jane is visiting Africa in September.
- $y^{<1>} y^{<2>} y^{<3>} y^{<4>} y^{<5>} y^{<6>}$

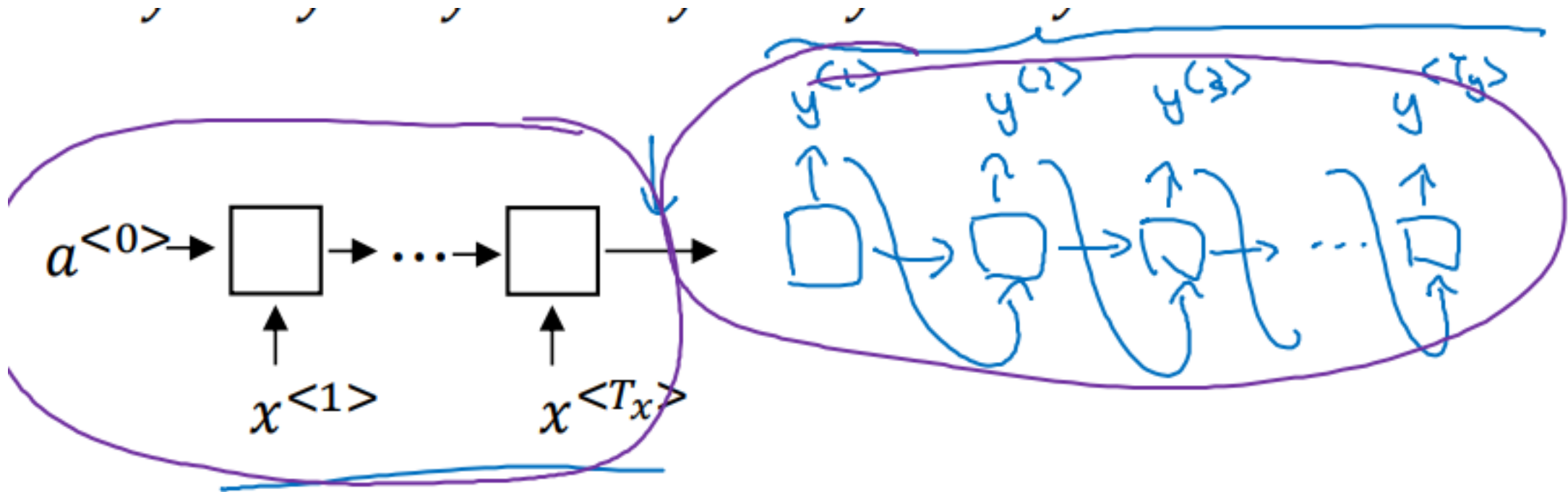
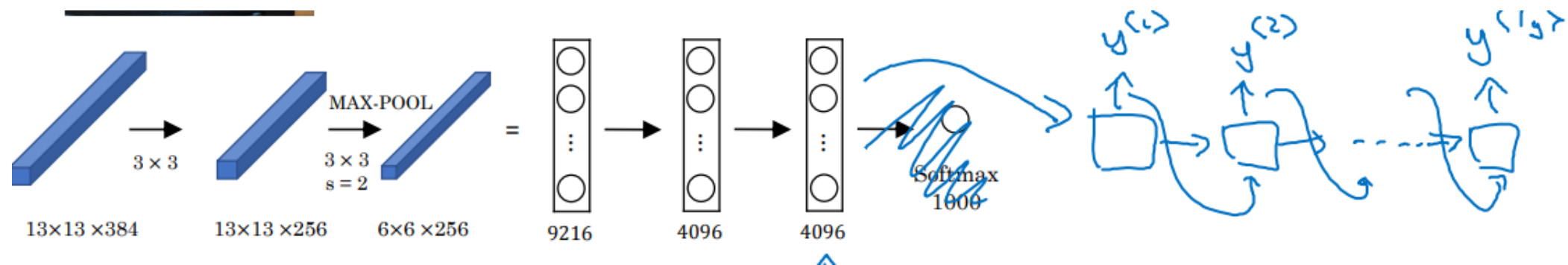
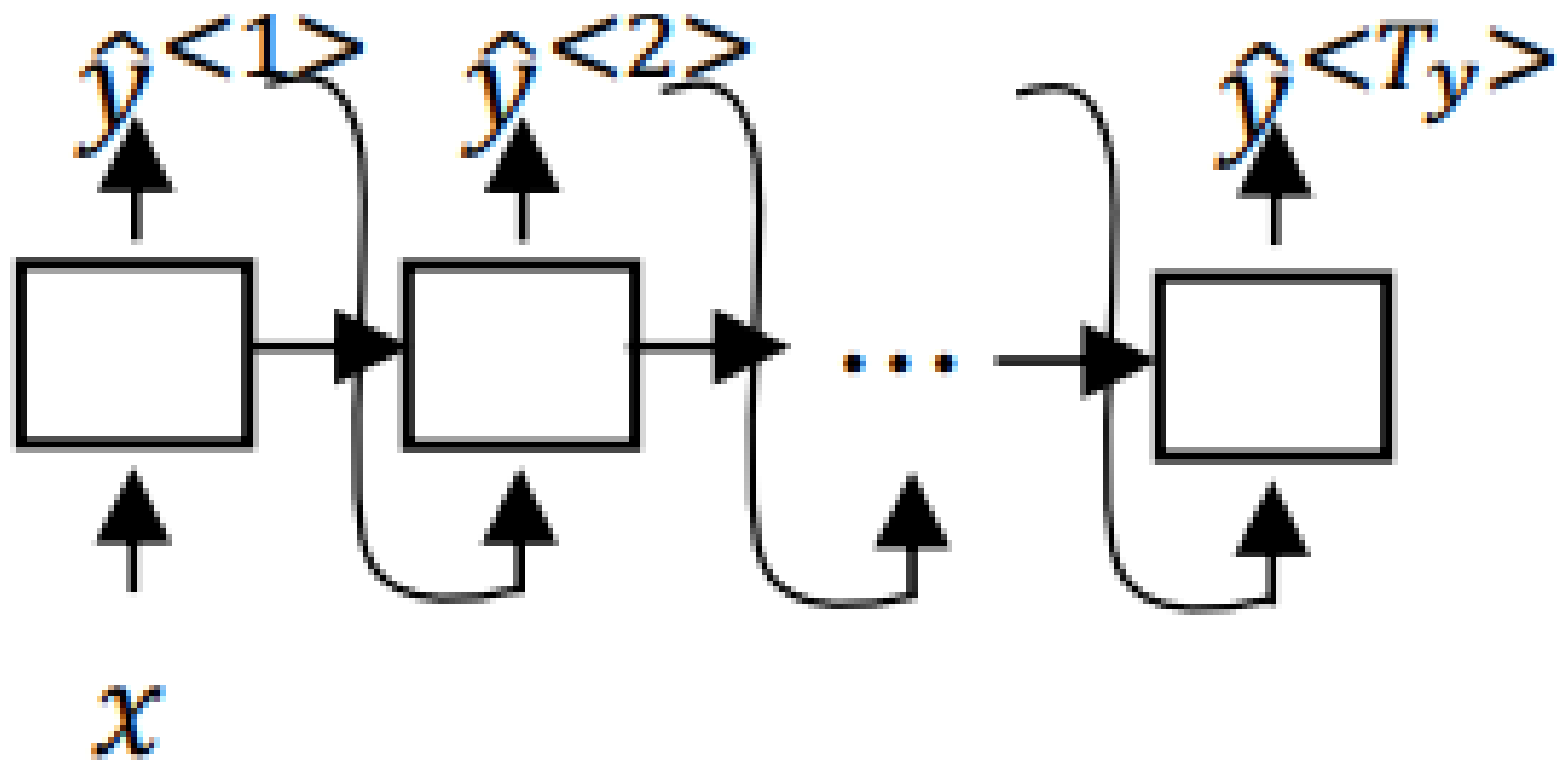


Image captioning





Beam Search

- So instead of each time sampling the world $y_{\sim t}$ at each greedy to be the one with the biggest probability in that distributions
- Take 3(if $B=3$) of the highest probability words and just go expand from there
- You will have many outputs choose the most confident one

Beam Search refined - remember the logx graph

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

log

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

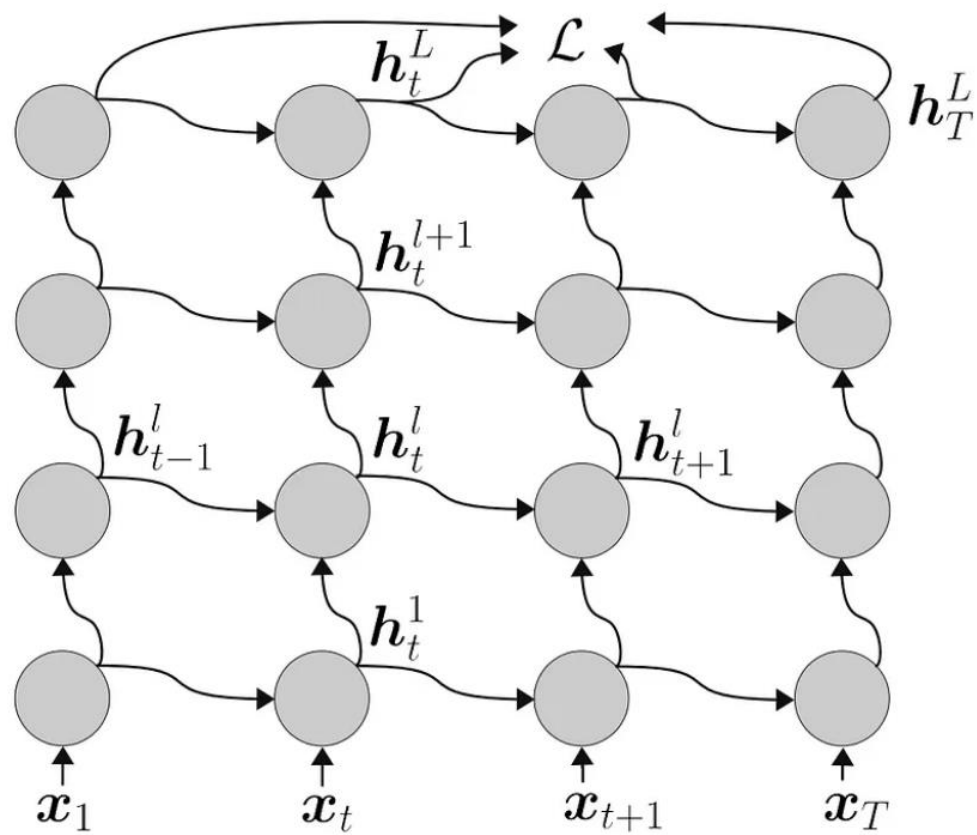
$T_y = 1, 2, 3, \dots, 30.$

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

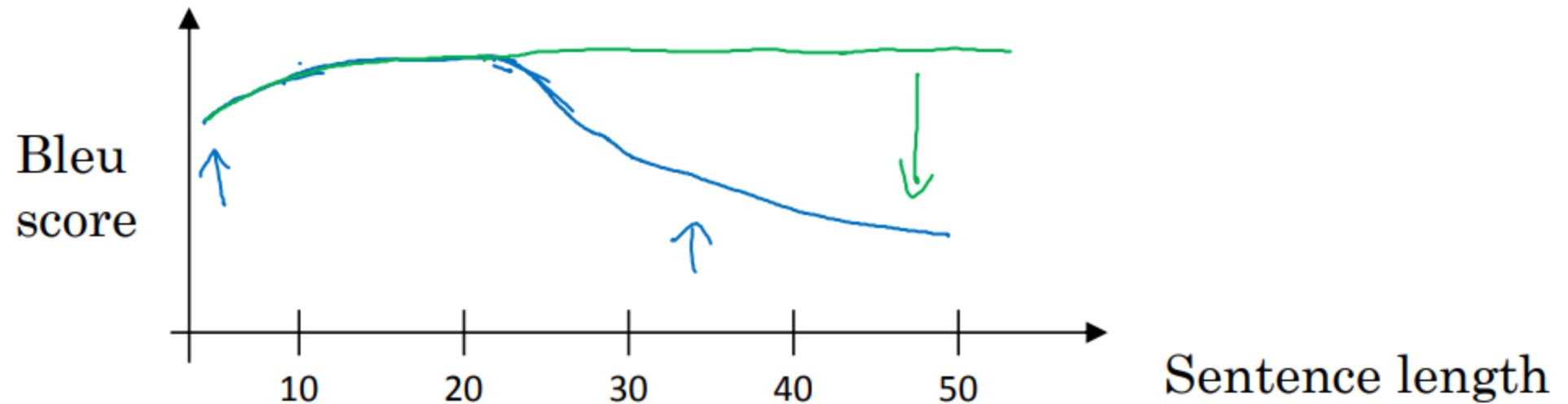
Beam width B?

- Large B means more accurate results
- Large B means faster
- Small B is the complete opposite
- Beam Search runs faster but is not guaranteed to find exact maximum for $\arg \max_y : P(y|x)$.

Deep RNN

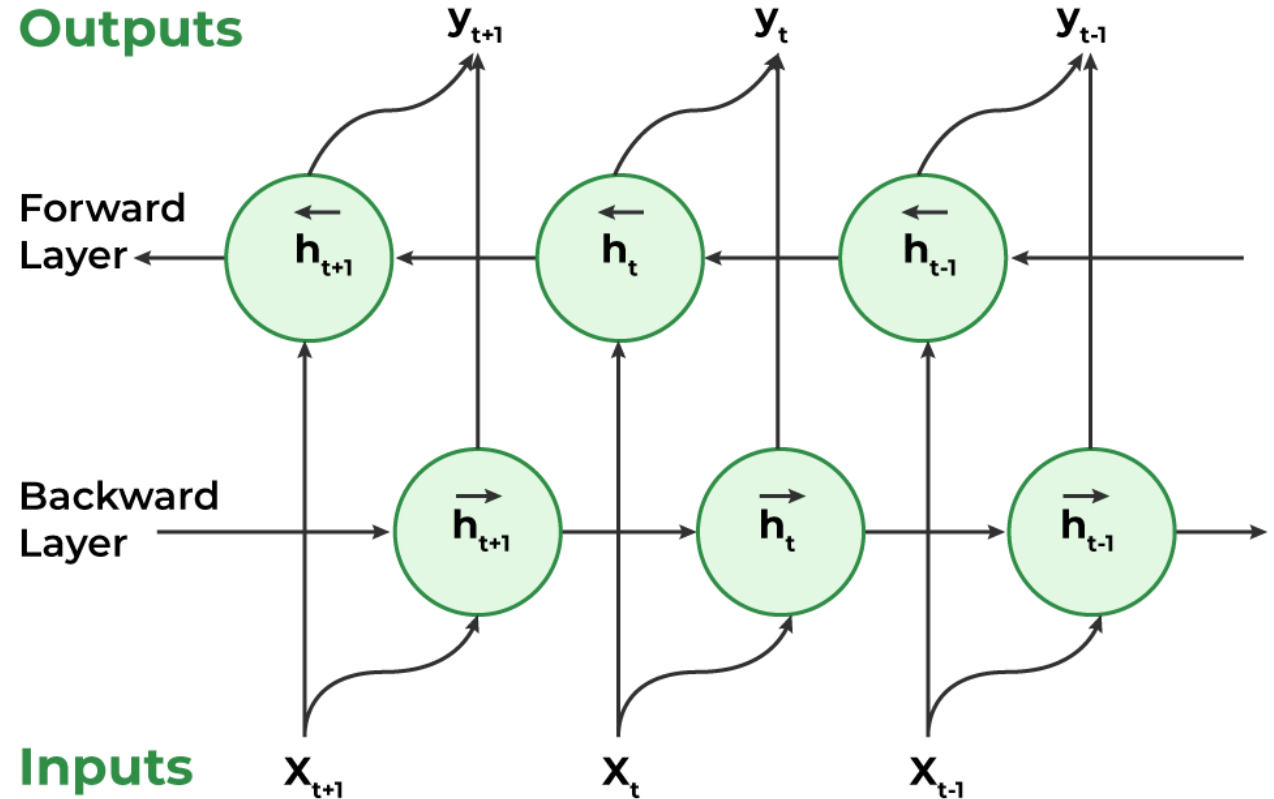


Attention - The problem with size



Attention model

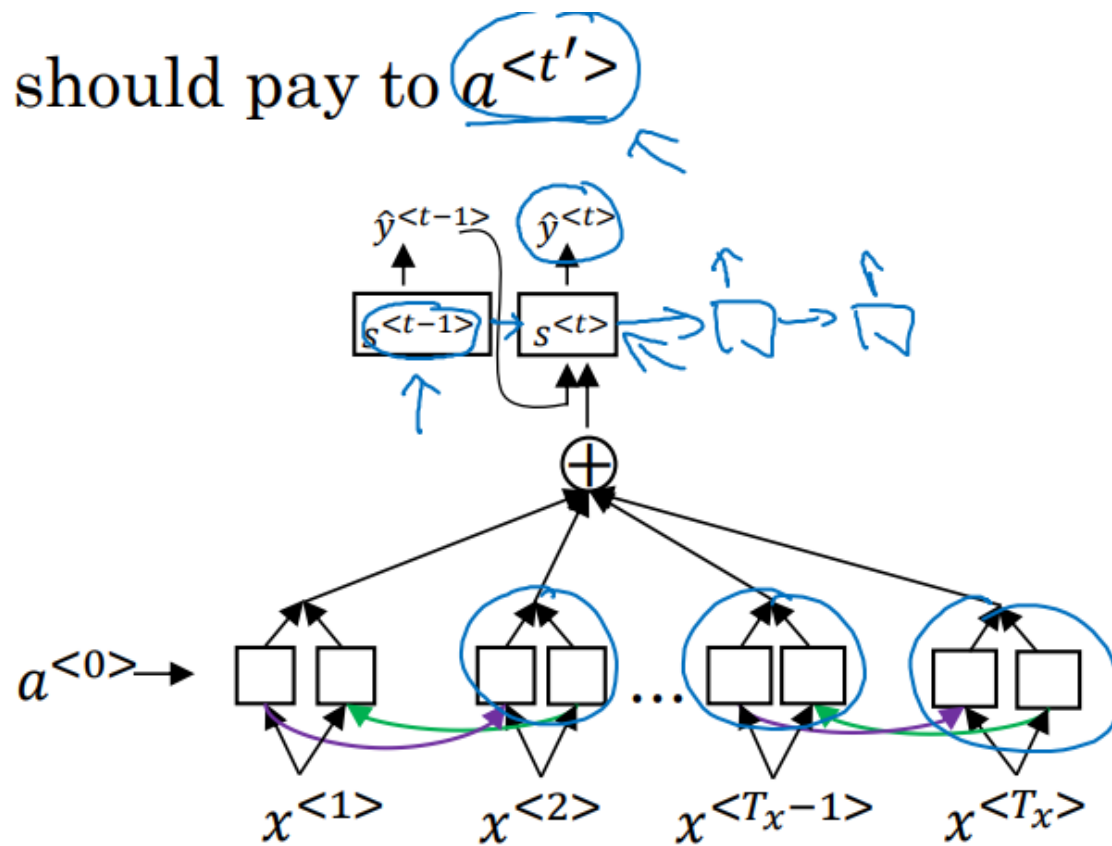
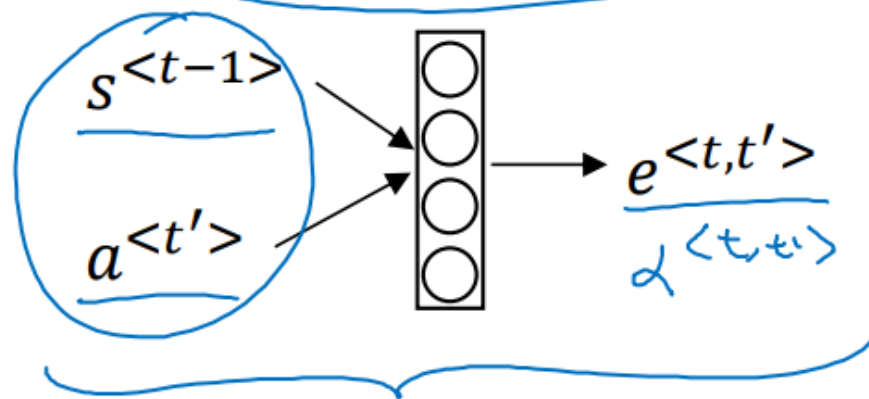
- Recall the BRNN
- In the BRNN each activation $h_{<t>}$ is a combination of 2 flows the forward and the backward flow
- In the attention we will connect the BRNN to a layer of states
- Each state will choose how much to attend to the other words
- By learning a parameter $e_{<t,t'>}$ which will tell us how much t is important to t'



Guide to attention models

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

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



Transformer Intuition

- It's an attention + parallelism
- Self-attention : if we have sentence of 5 words, we will compute 5 representations to the words in parallel
- Multi head attention is for loop over self-attention so we can compute many of the self-attention

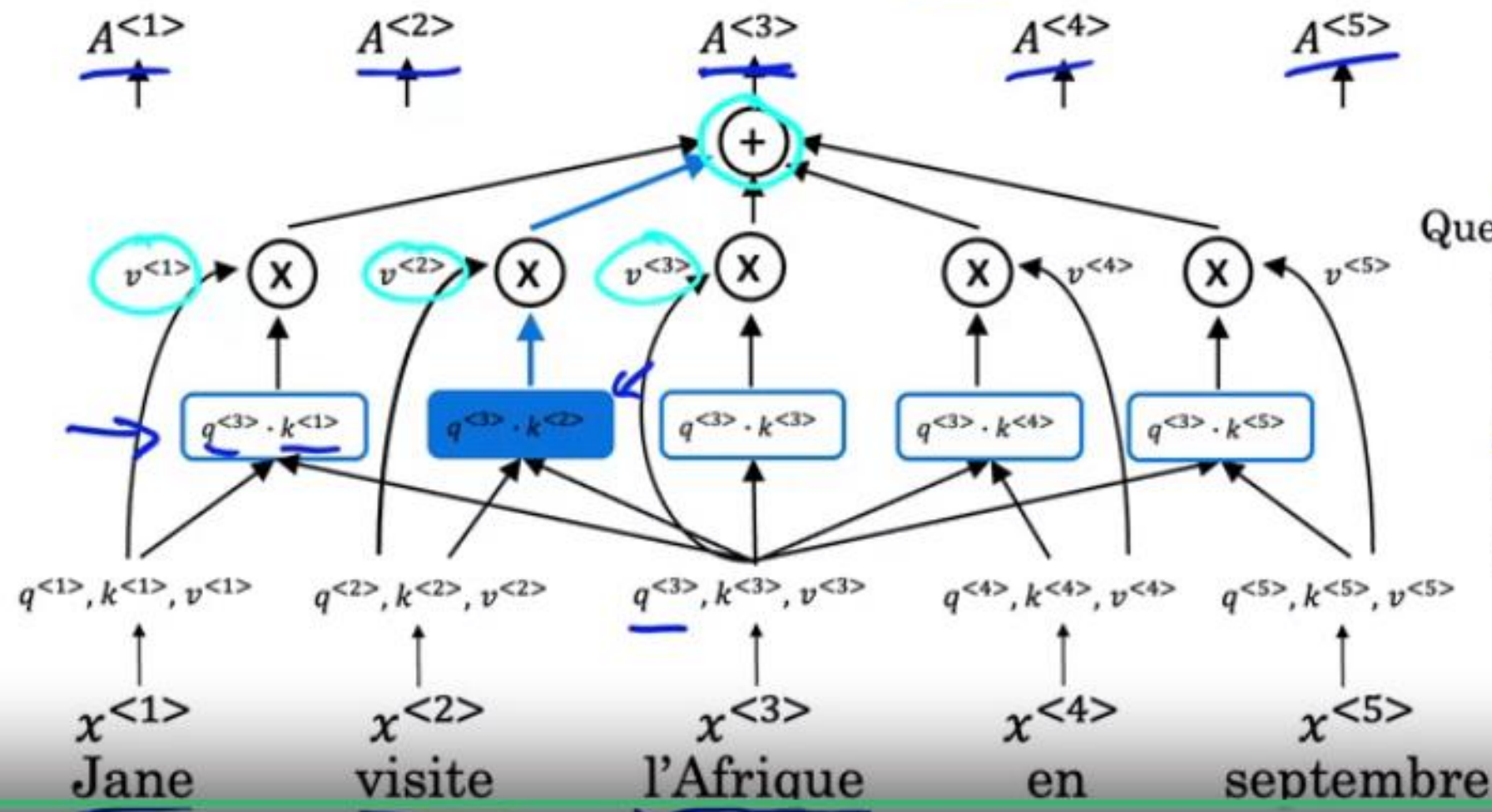
Self-Attention

- For each word we need to create attention-based representation
- $A(q,v,k)$
- For every word I have three values query and key and value

Self-Attention

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

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



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

$$q^{<3>} = W^Q \cdot x^{<3>}$$

$$k^{<3>} = W^K \cdot x^{<3>}$$

$$v^{<3>} = W^V \cdot x^{<3>}$$

Multi-Head Attention

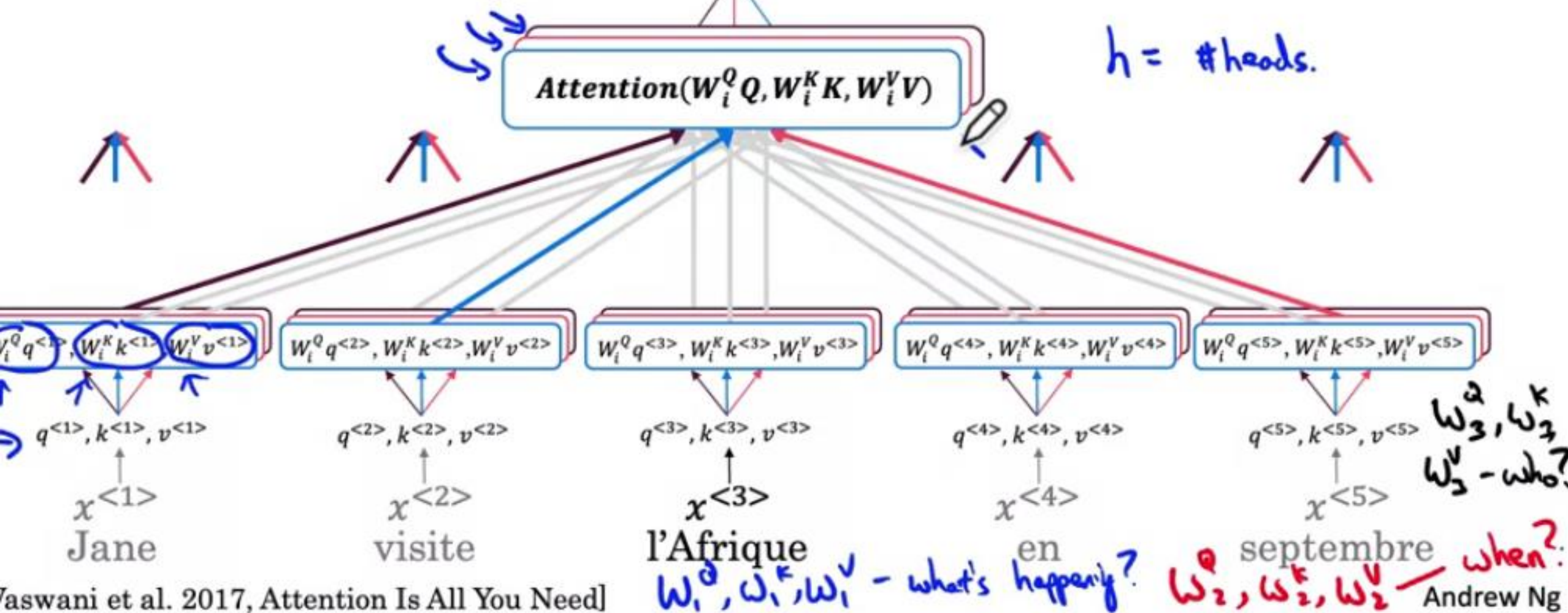
"head"

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


$$MultiHead(Q, K, V) = \text{concat}(\text{head}_1 \text{head}_2 \dots \text{head}_h)W_o$$

$$\text{head}_i = Attention(W_i^Q Q, W_i^K K, W_i^V V)$$

$h = \# \text{heads.}$



Transformer Details

<SOS> Jane visits  Africa in September <EOS>

Encoder

Decoder

