

The lecture will be started at 5:05PM sharply!

COMP5046

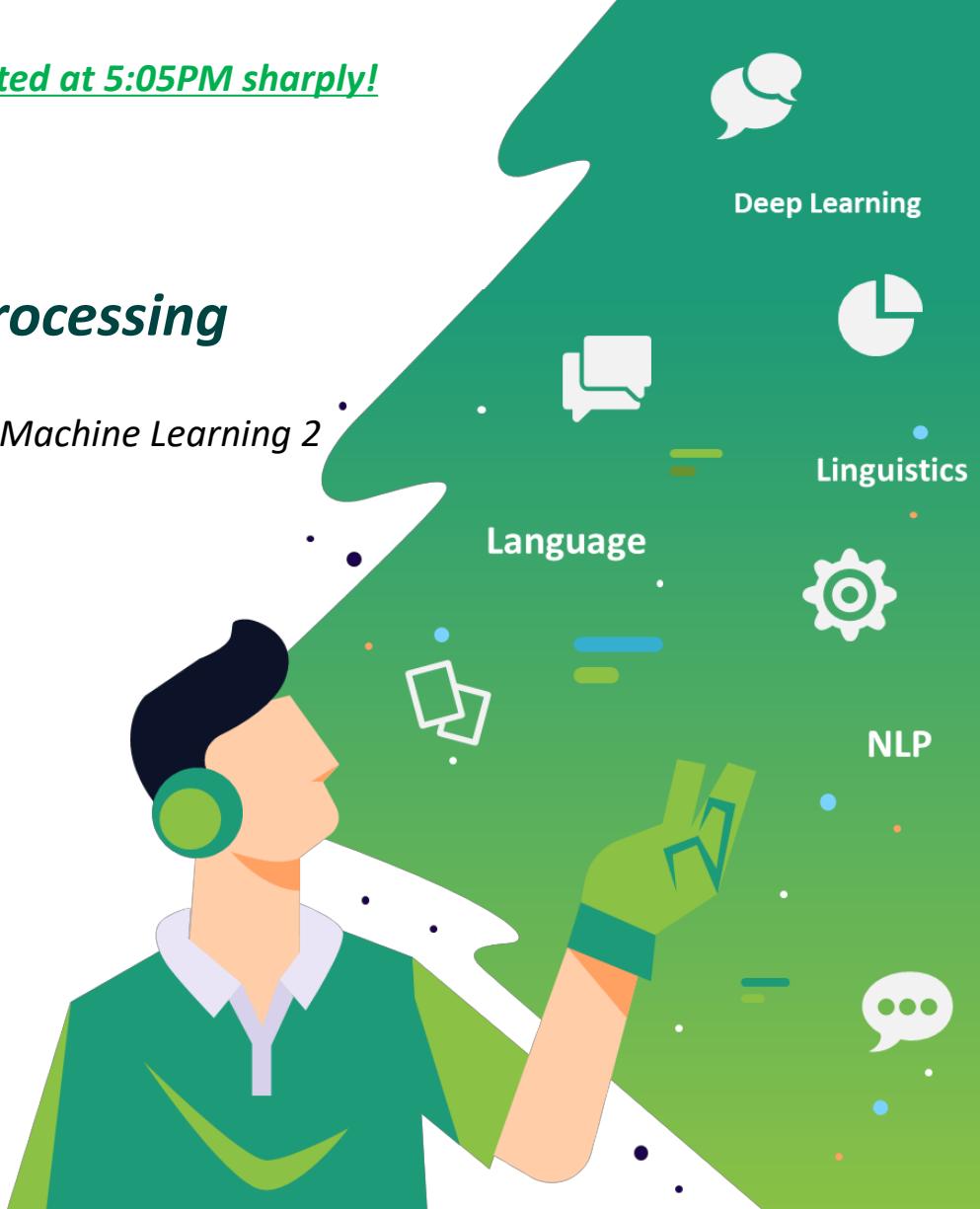
Natural Language Processing

Lecture 4: Word Classification and Machine Learning 2

Dr. Caren Han

Semester 1, 2022

*School of Computer Science,
University of Sydney*



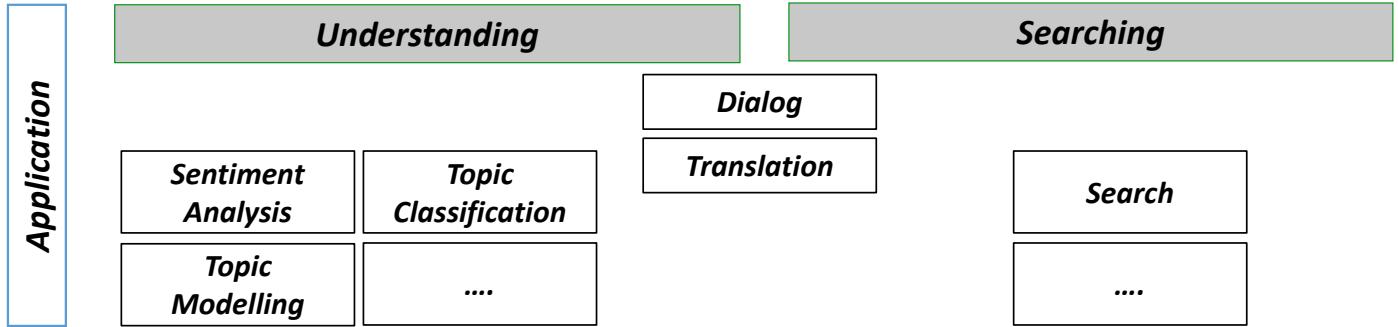
0 LECTURE PLAN

Lecture 4: Word Classification and Machine Learning 2

1. Machine Learning and NLP: Finish
2. Seq2Seq Learning
3. Seq2Seq Deep Learning
 1. RNN (Recurrent Neural Network)
 2. LSTM (Long Short-Term Memory)
 3. GRU (Gated Recurrent Unit)
4. Data Transformation for Deep Learning NLP
5. Next Week Preview
 - Natural Language Processing Stack

1 Machine Learning and NLP

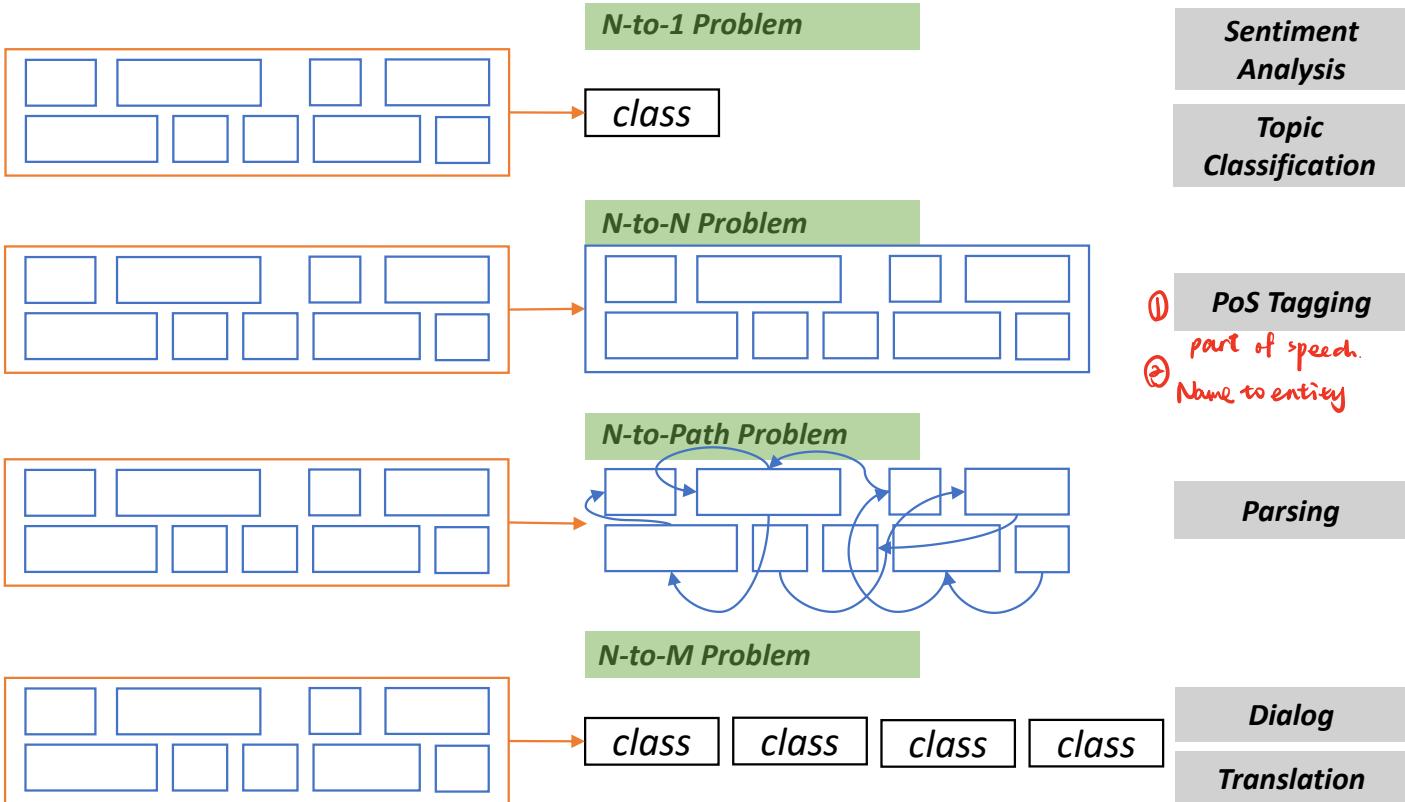
The purpose of Natural Language Processing: Overview



<i>NLP Stack</i>	Entity Extraction	When Sebastian Thrun ...	When Sebastian Thrun PERSON started at Google ORG in 2007 DATE
	Parsing	Claudia sat on a stool	<pre> graph TD S --- NP1[NP] S --- VP NP1 --- N1[N] VP --- V1[V] VP --- PP[PP] PP --- P1[P] PP --- AT1[AT] PP --- NP2[NP] NP2 --- N2[N] N1 --- Claudia[N] V1 --- sat[V] P1 --- on[P] AT1 --- a[AT] N2 --- stool[N] </pre>
	PoS Tagging	She sells seashells	[she/PRP] [sells/VBZ] [seashells/NNS]
	Stemming	Drinking, Drank, Drunk	Drink
	Tokenisation	How is the weather today	[How] [is] [the] [weather] [today]

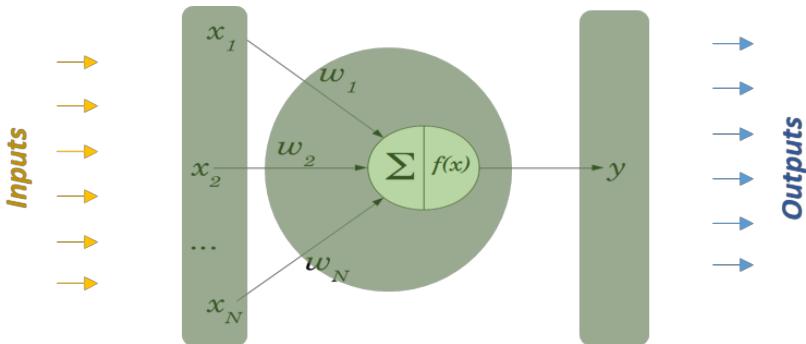
1 Machine Learning and NLP

Problem Abstraction



1 Machine Learning and NLP

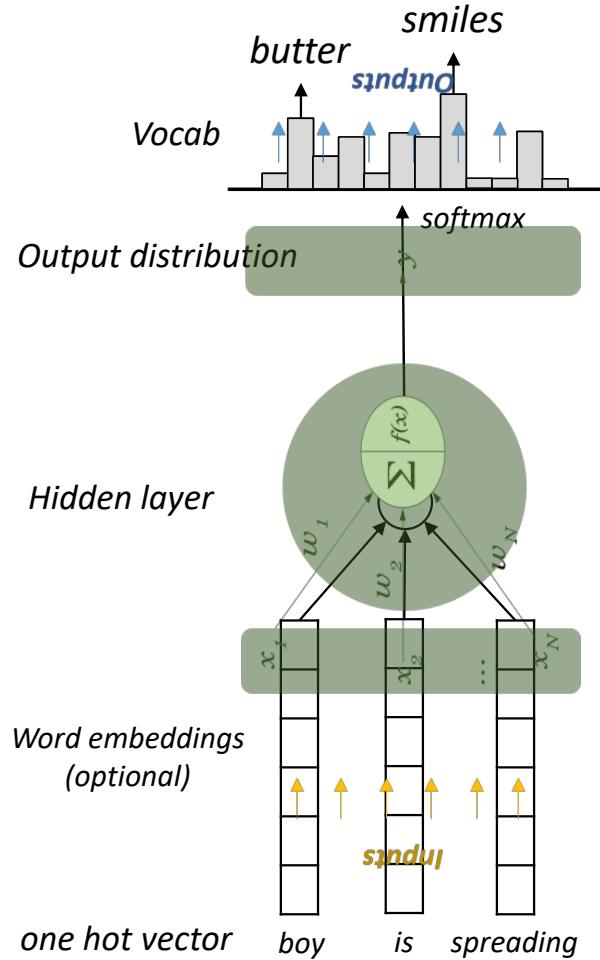
Prediction



x_i	Inputs	Features words (indices or vectors!), context windows, sentences, documents, etc.
y_i	Outputs (labels)	What we try to predict/classify <ul style="list-style-type: none"> E.g. word meaning, sentiment, name entity

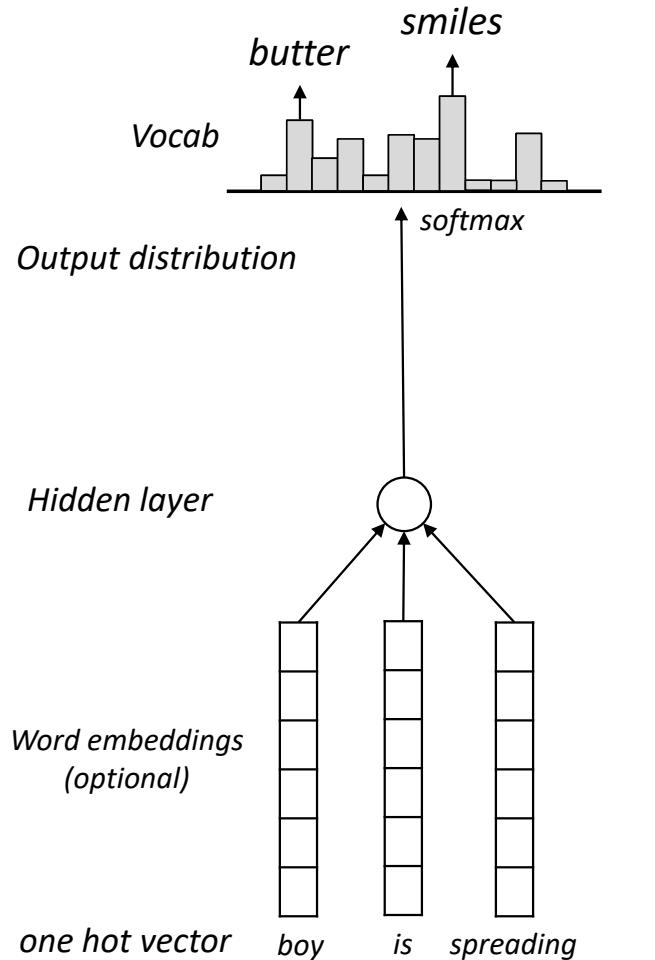
1 Machine Learning and NLP

Prediction



1 Machine Learning and NLP

Prediction



order matters

What if we consider this as
a sequential input?
 Let's add the concept 'time'

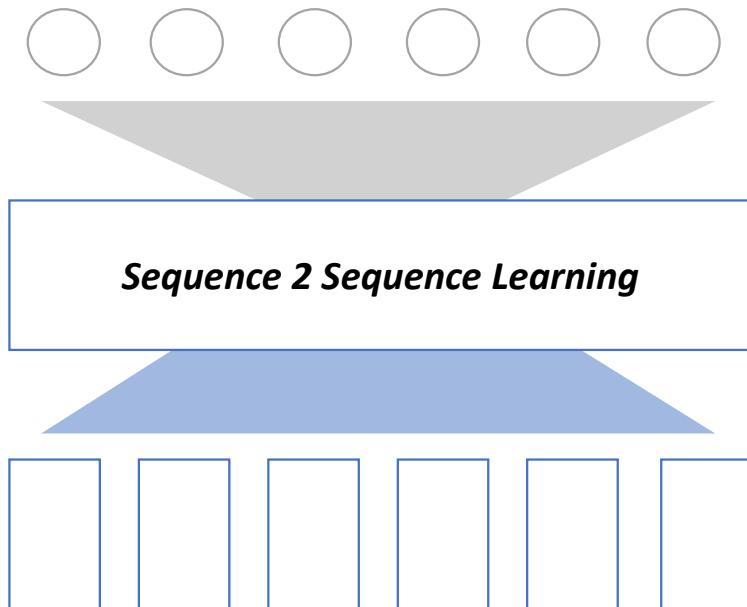
0 LECTURE PLAN

Lecture 4: Word Classification and Machine Learning 2

1. Machine Learning and NLP: Finish
2. **Seq2Seq Learning**
3. Seq2Seq Deep Learning
 1. RNN (Recurrent Neural Network)
 2. LSTM (Long Short-Term Memory)
 3. GRU (Gated Recurrent Unit)
4. Data Transformation for Deep Learning NLP
5. Next Week Preview
 - Natural Language Processing Stack

2 Sequence 2 Sequence Learning

Illustration



2 Sequence 2 Sequence Learning

Running time

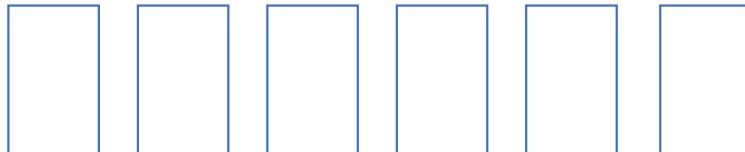
$$M = \# \text{ of } \textcolor{green}{\circlearrowleft} \quad \textcolor{blue}{\text{\# of output}}$$



Sequence Generation



Sequence 2 Sequence Learning



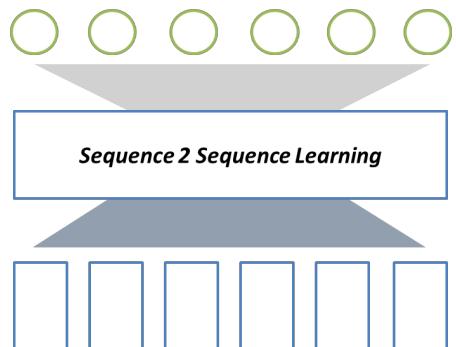
Sequence Feeding

$$N = \# \text{ of } \textcolor{blue}{\square} \quad \textcolor{blue}{\text{\# of input}}$$

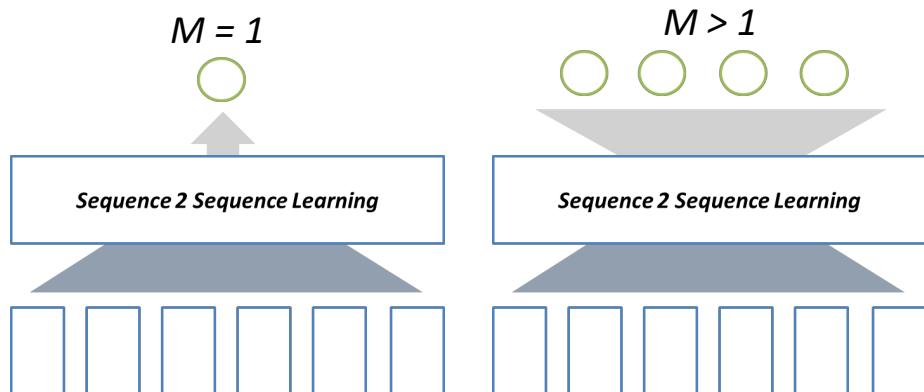
2 Sequence 2 Sequence Learning

Sequence 2 Sequence Learning

$$N = M$$



$$N \neq M$$



2 Sequence 2 Sequence Learning

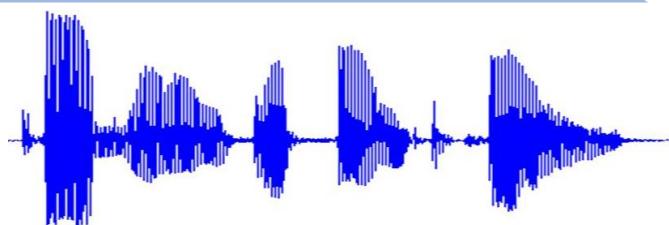
Seq2Seq – Speech Recognition

How is the weather today

Output: Text



Sequence 2 Sequence Learning



Input: Speech Signal

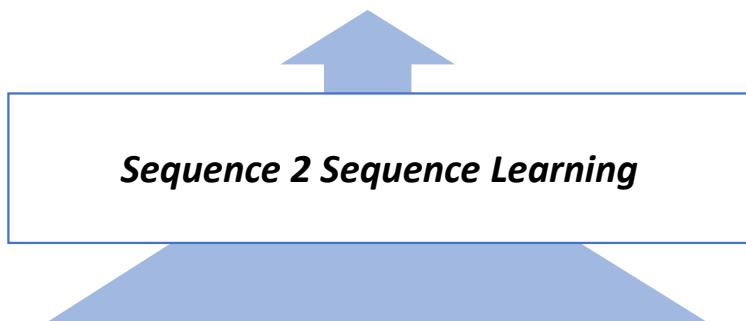
2 Sequence 2 Sequence Learning

Seq2Seq – Movie Frame Labelling



Output: Scene Labels

Swing Swing Hit Bat_Broken



Input: Video Frame

2 Sequence 2 Sequence Learning

Seq2Seq – PoS Tagging

ADV VERB DET NOUN NOUN

Output: Part of Speech



Sequence 2 Sequence Learning

How is the weather today

Input: Text

2 Sequence 2 Sequence Learning

Seq2Seq – Arithmetic Calculation

4. A farmer has 7 ducks.
He has 5 times as many chickens as ducks.
How many more chickens than ducks does he have?

ducks 

chickens 

Find the number of
chickens first.

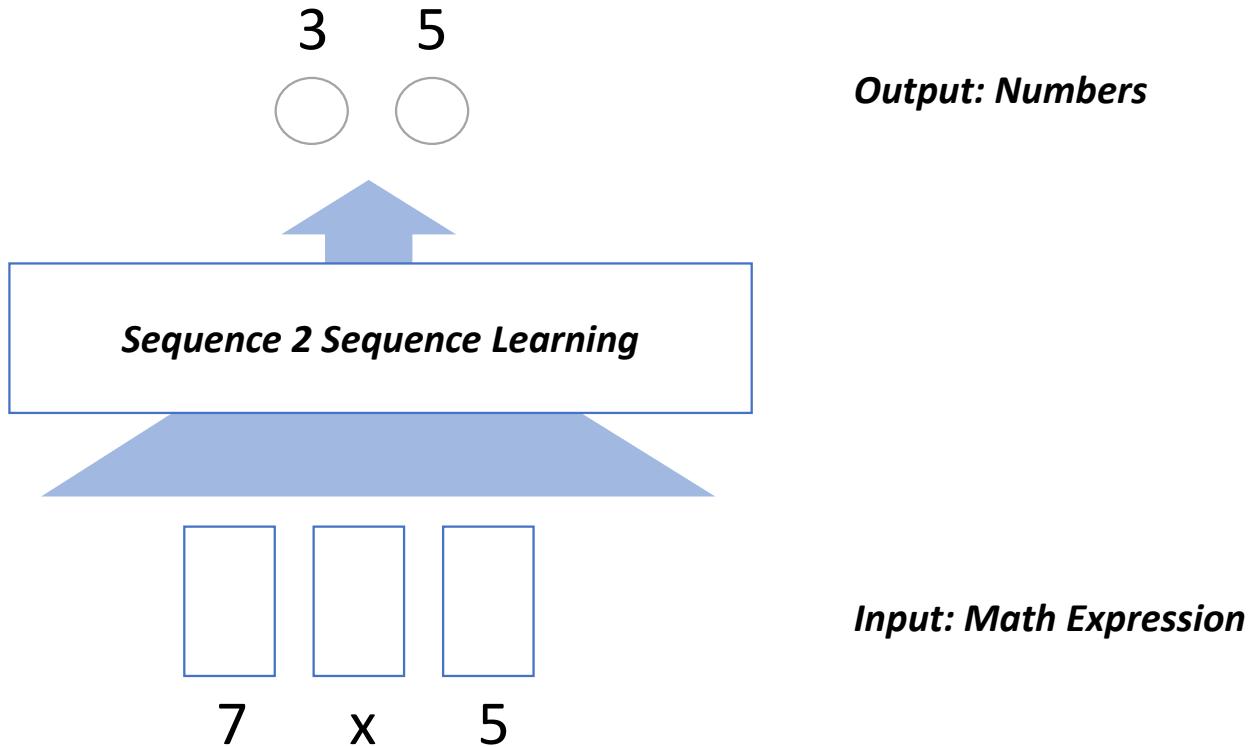


$$7 \times 5 = 35$$

X Y

2 Sequence 2 Sequence Learning

Seq2Seq – Arithmetic Calculation



2 Sequence 2 Sequence Learning

Seq2Seq – Machine Translation

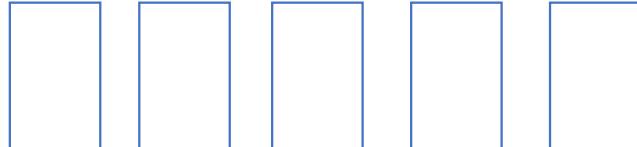


今天 天气 怎么 样?

Output: Chinese Text



Sequence 2 Sequence Learning



How is the weather today

Input: English Text

2 Sequence 2 Sequence Learning

Seq2Seq – Sentence Completion

How is the weather today?

How long does it take?

Let's go to the opera house

It is quite hot inside

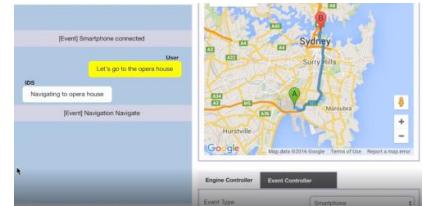
I may need to stop by Darling Harbour

When is the dinner appointment

Change the schedule

Text him that I cannot meet at 6:30pm

I like learning Natural Language Processing



2 Sequence 2 Sequence Learning

Seq2Seq – Sentence Completion

How is the weather today?

How long does it take?

Let's go to the opera house

It is quite hot inside

I may need to stop by Darling Harbour

When is the dinner appointment

Change the schedule

Text him that I cannot meet at 6:30pm

X

I like learning Natural Language Processing

Y

2 Sequence 2 Sequence Learning

I like learning Natural Language Processing

Seq2Seq – Sentence Completion

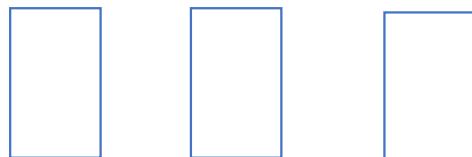
Natural Language Processing



Output: Partial Sentence



Sequence 2 Sequence Learning



Input: Partial Sentence

I like learning

2 Sequence 2 Sequence Learning

Seq2Seq – Conversation Modelling

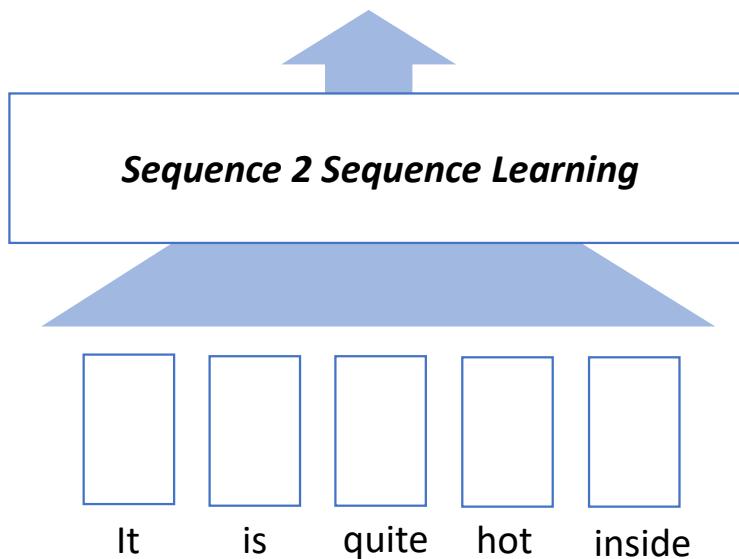


2 Sequence 2 Sequence Learning

Seq2Seq – Conversation Modelling

Okay. I will open windows for you

Output: Utterance



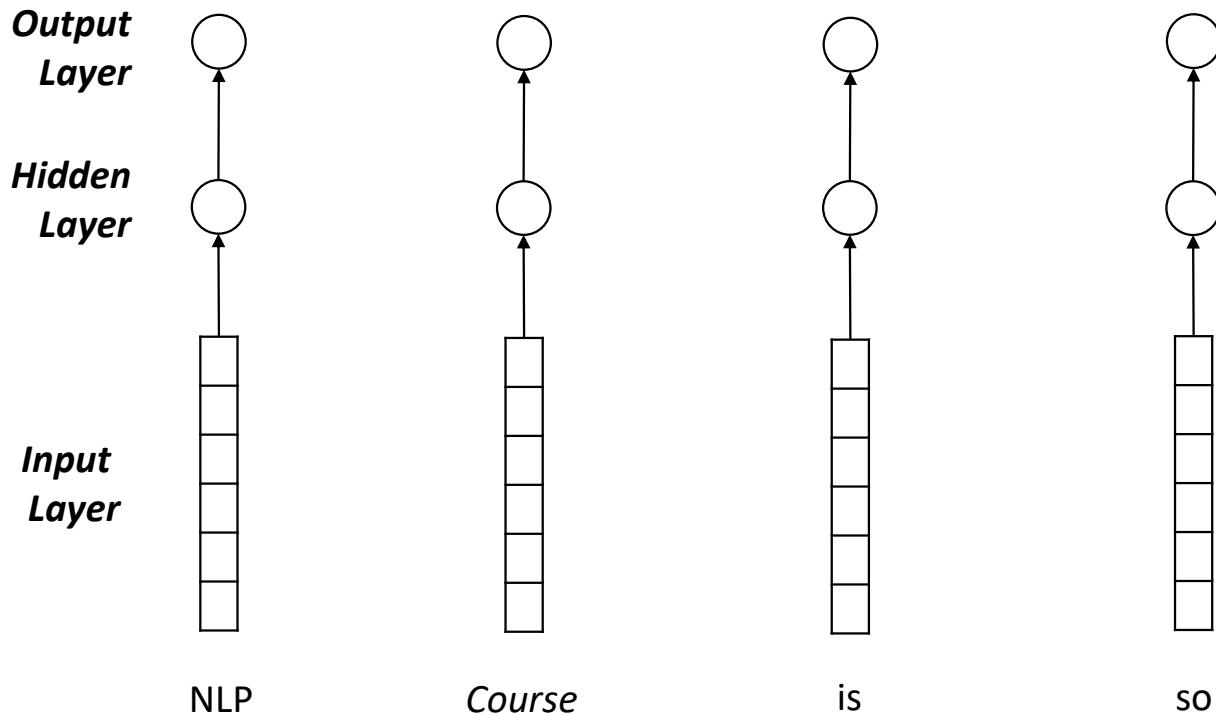
0 LECTURE PLAN

Lecture 4: Word Classification and Machine Learning 2

1. Machine Learning and NLP: Finish
2. Seq2Seq Learning
3. **Seq2Seq Deep Learning**
 1. RNN (Recurrent Neural Network)
 2. LSTM (Long Short-Term Memory)
 3. GRU (Gated Recurrent Unit)
4. Data Transformation for Deep Learning NLP
5. Next Week Preview
 - Natural Language Processing Stack

3 Seq2Seq with Deep Learning

Prediction

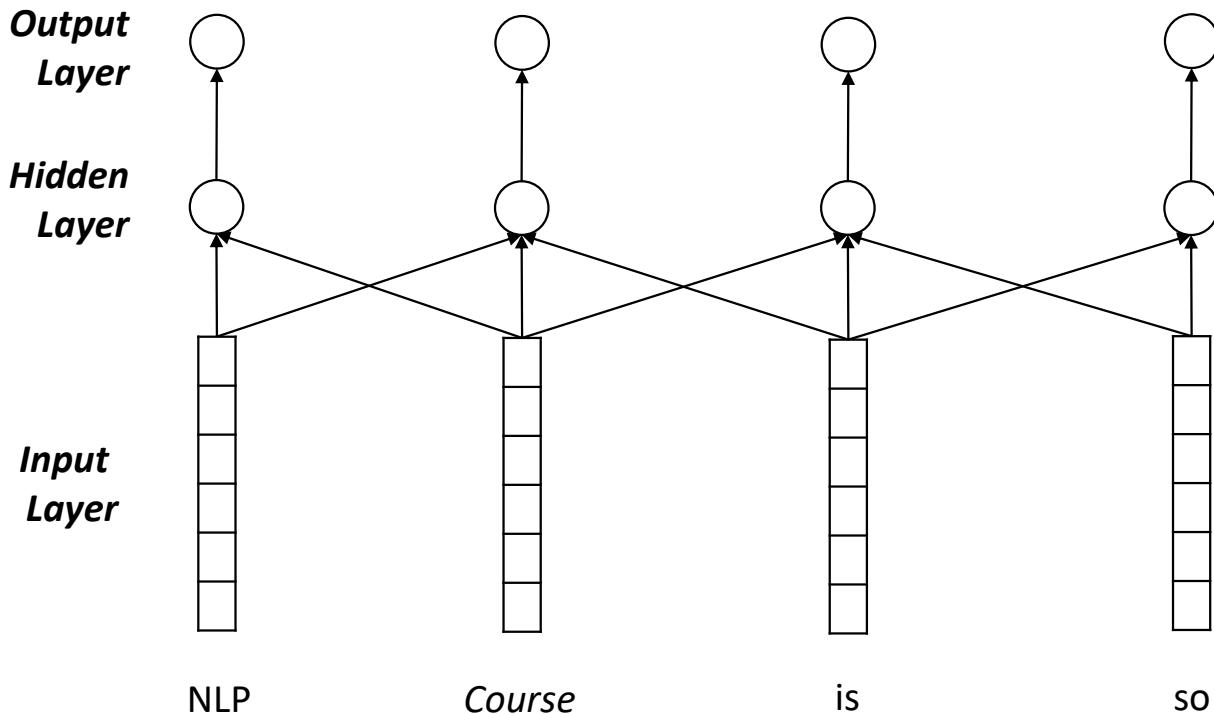


3 Seq2Seq with Deep Learning

Prediction + Convolution Idea

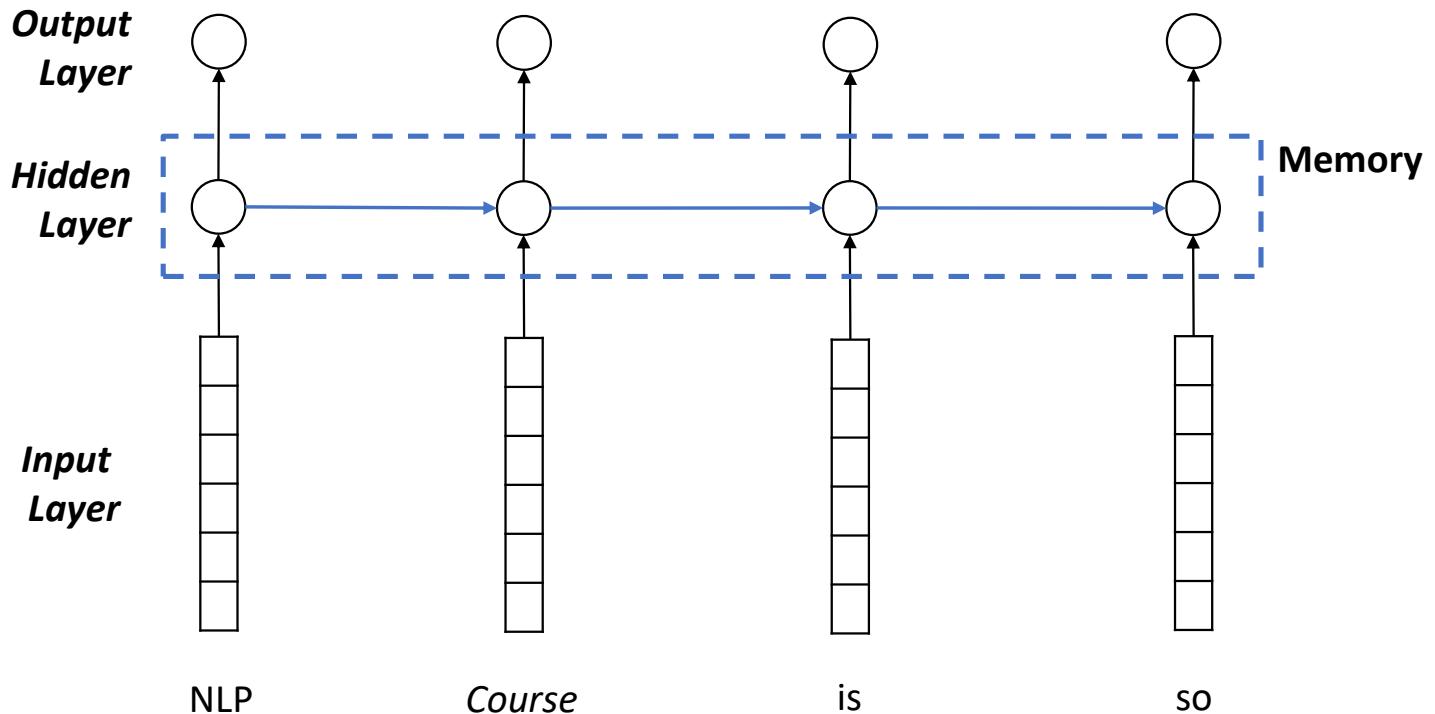
(before & after word)

Not care other things.



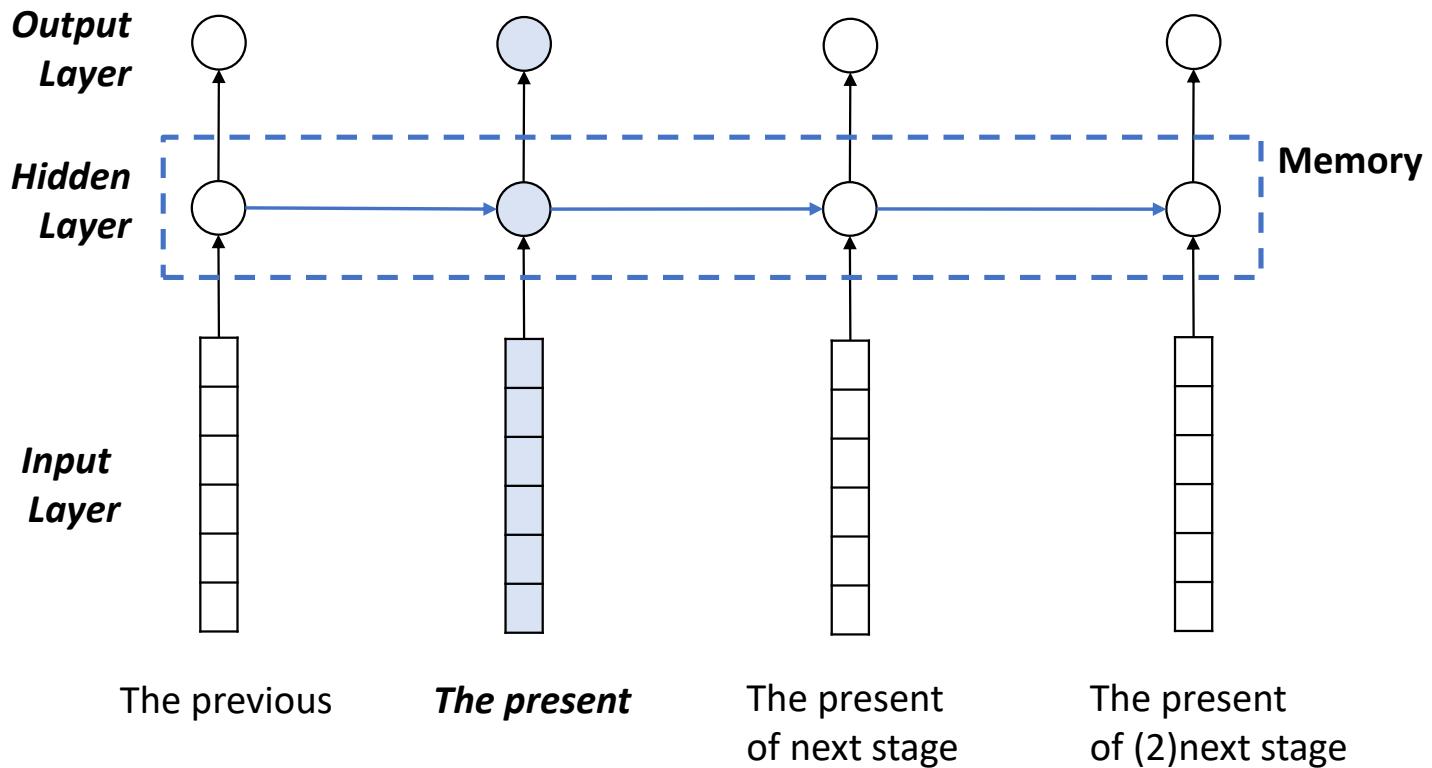
3 Seq2Seq with Deep Learning

Prediction + Memory = Sequence Modelling



3 Seq2Seq with Deep Learning

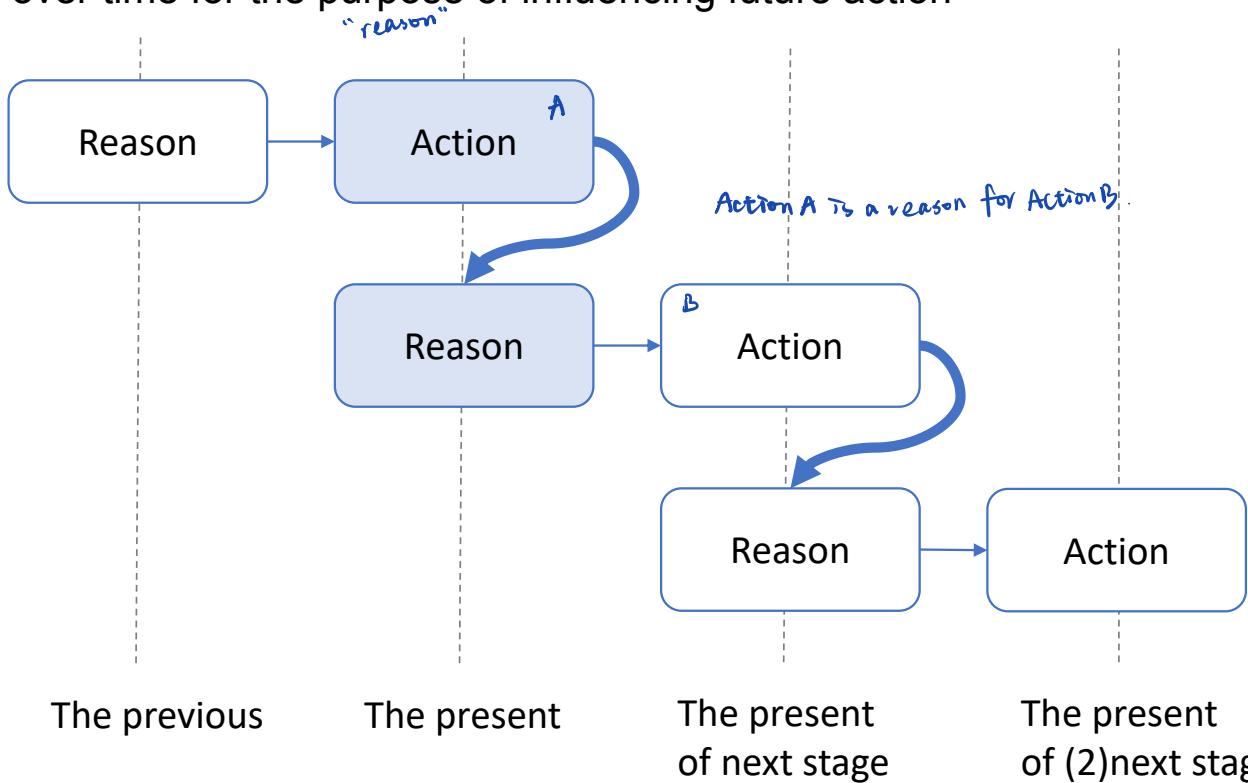
Prediction + Memory = Sequence Modelling



3 Seq2Seq with Deep Learning

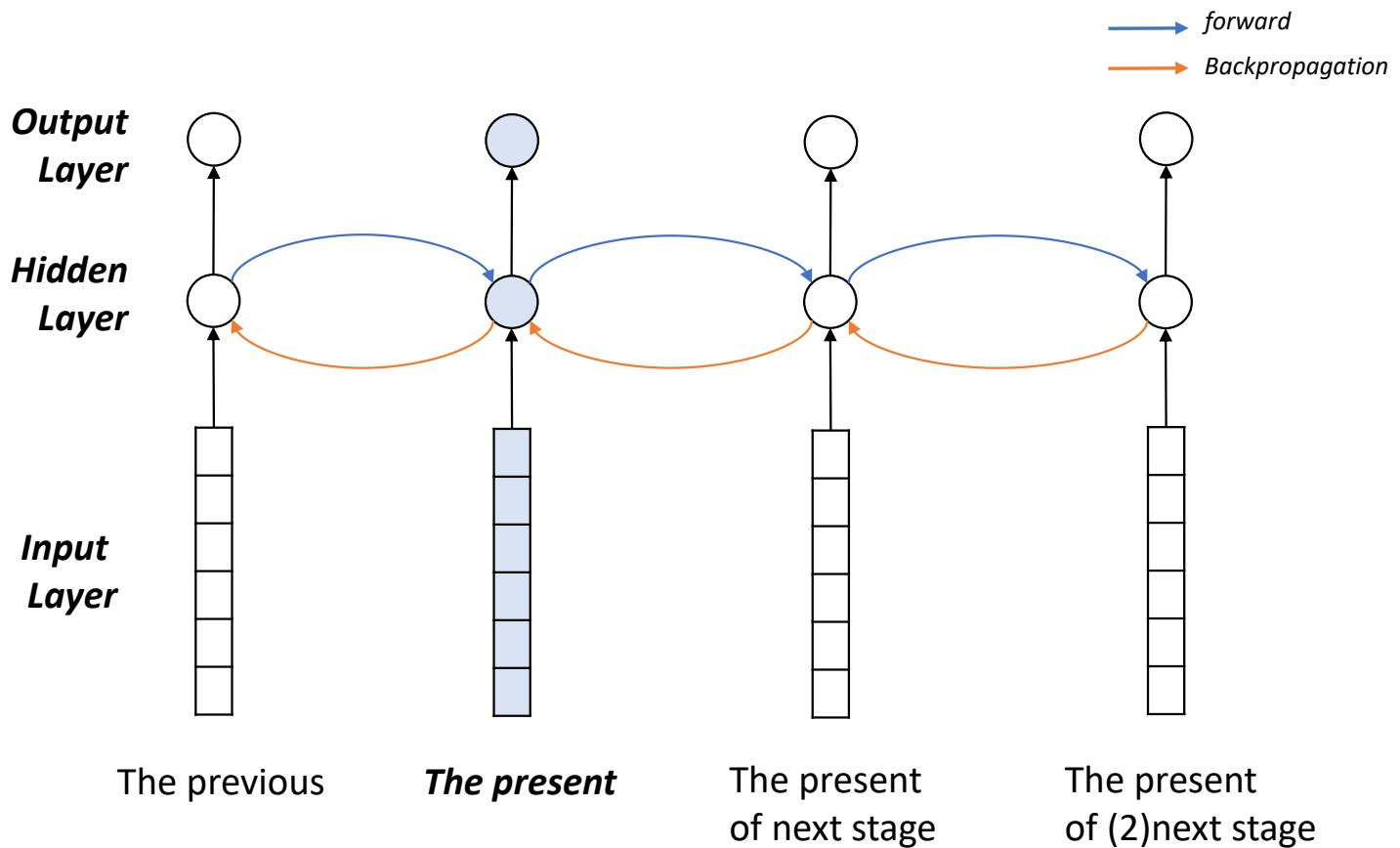
Neural Network + Memory

Memory is vital to experiences; it is the retention of information over time for the purpose of influencing future action



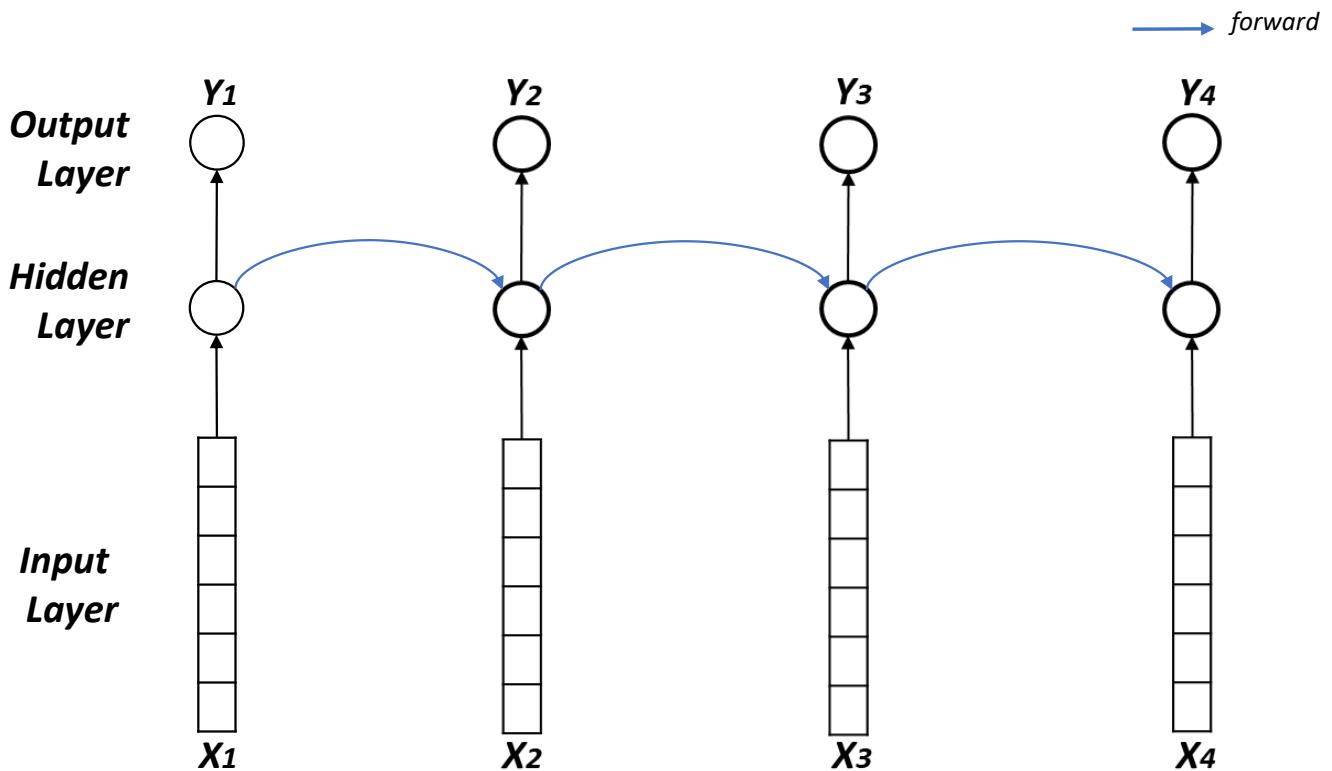
3 Seq2Seq with Deep Learning

Neural Network + Memory



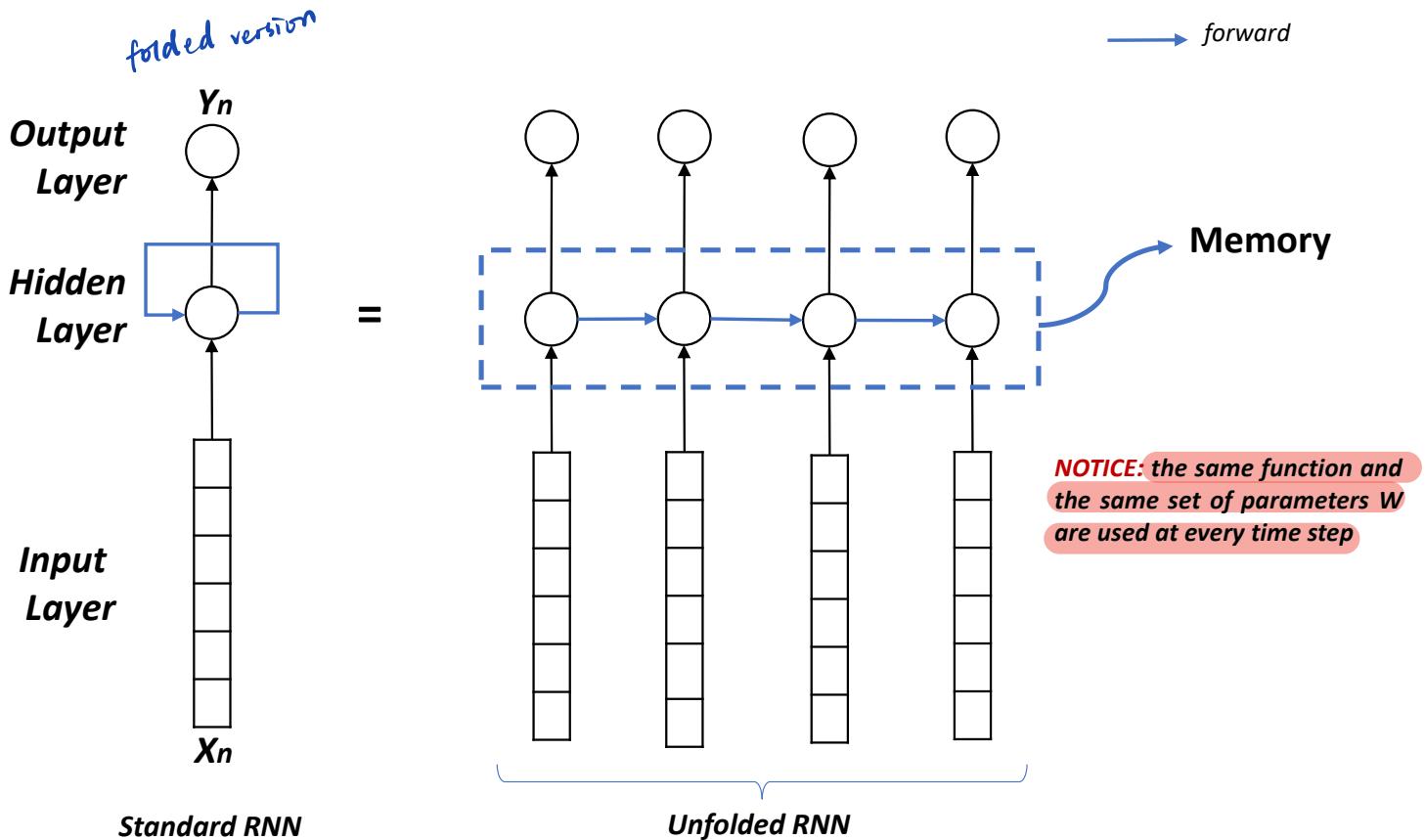
3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network



3 Seq2Seq with Deep Learning

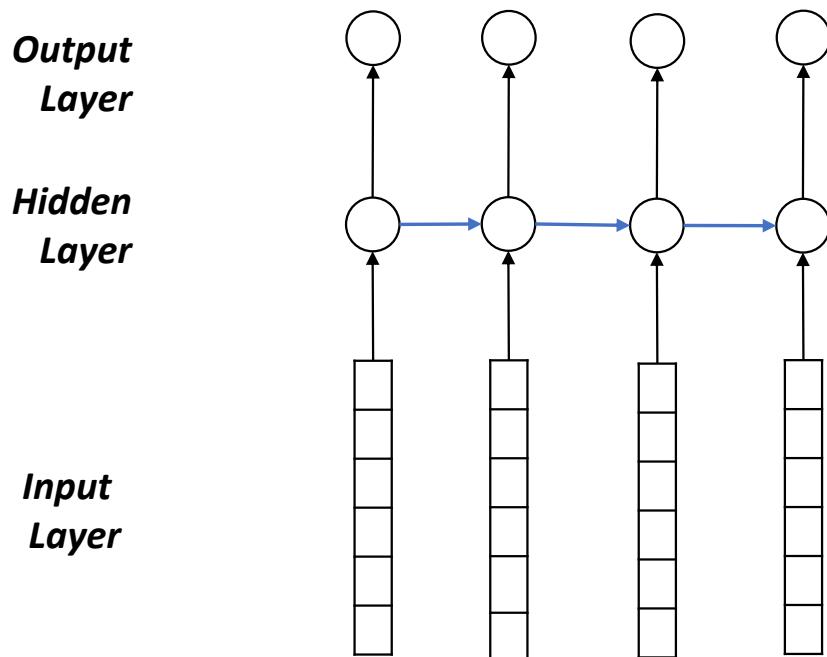
Neural Network + Memory = Recurrent Neural Network (RNN)



3

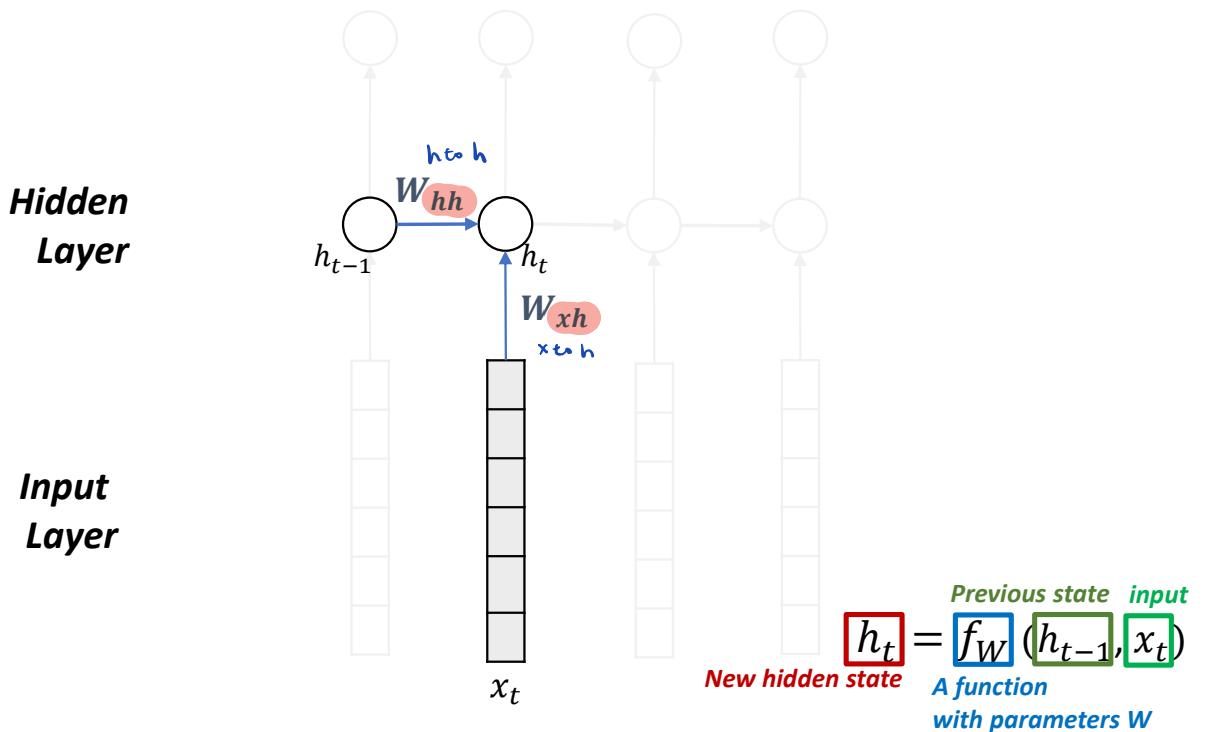
Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network (RNN)



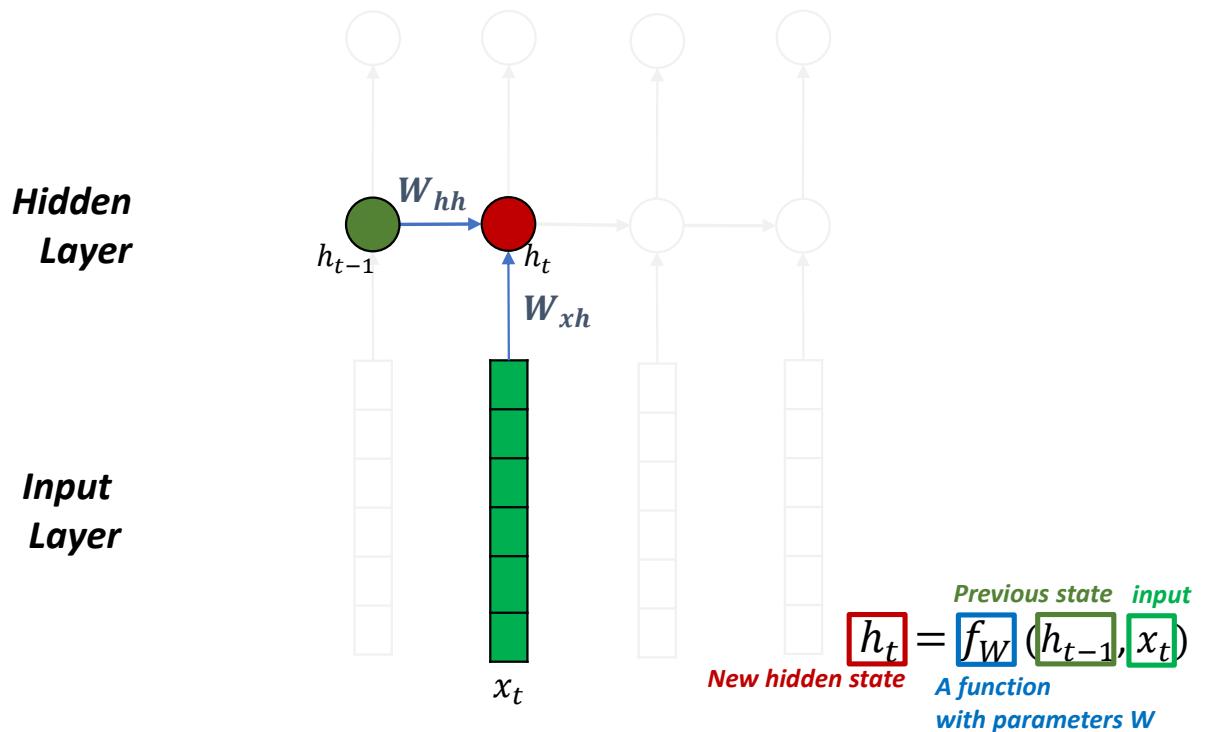
3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network



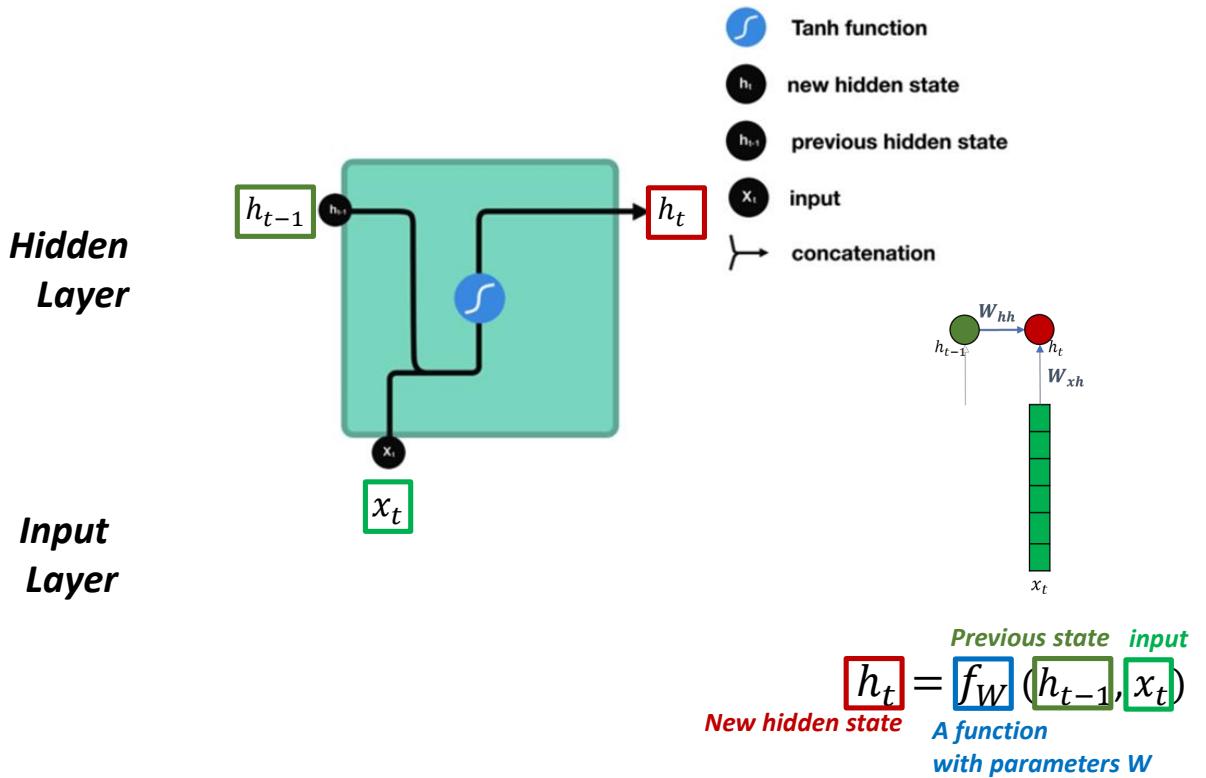
3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network



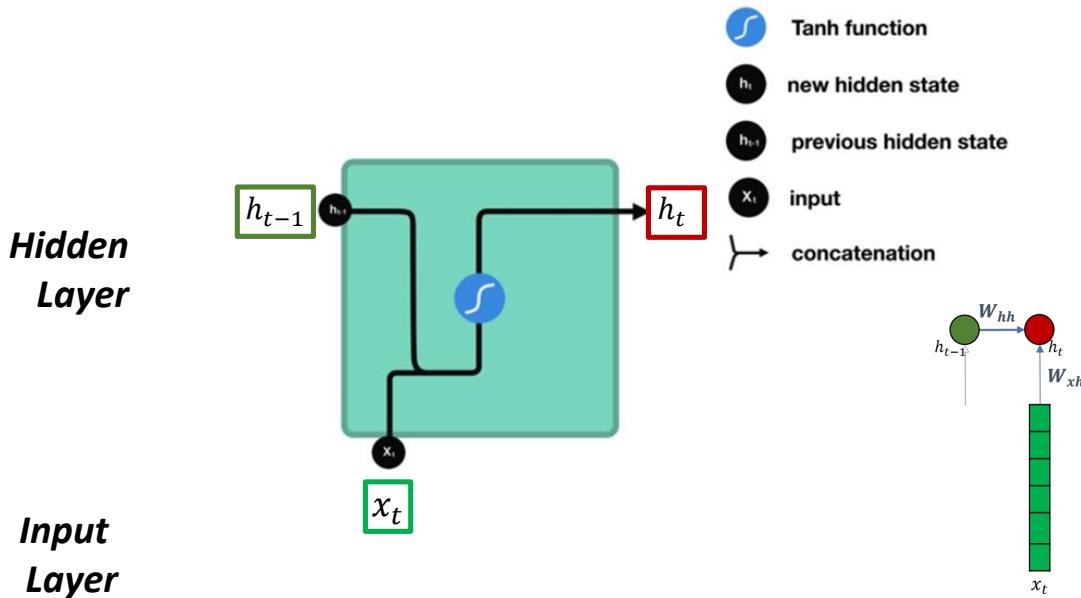
3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network



3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

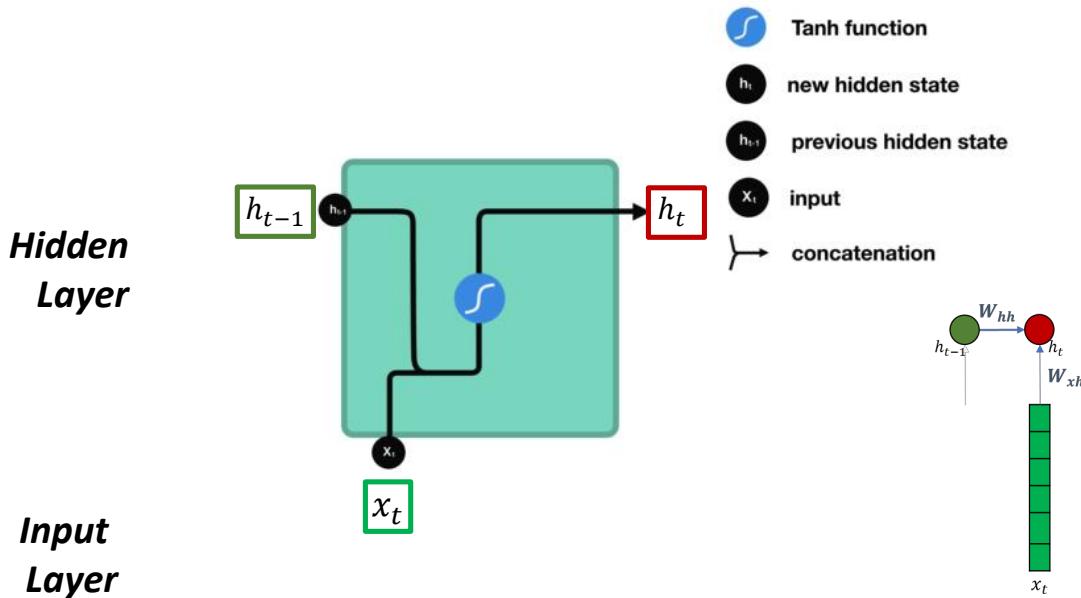


$$h_t = \tanh(W_{hh} h_{t-1} + W_{xh} x_t + b_h)$$

New hidden state Previous state input
 A function with parameters W

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network



$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h)$$

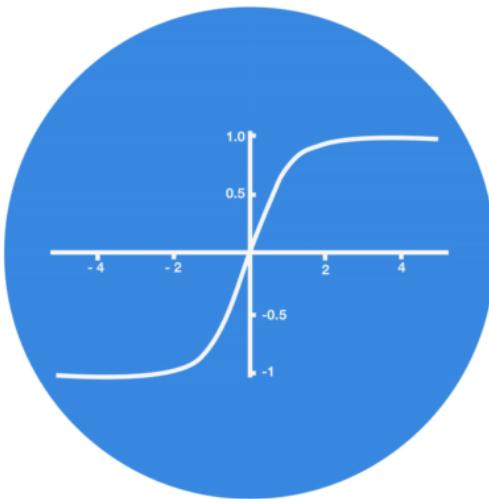
New hidden state Previous state input
 A function with parameters W

3 Seq2Seq with Deep Learning

Tanh activation

The tanh activation is used to help regulate the values flowing through the network. The tanh function squishes values to always be between -1 and 1.

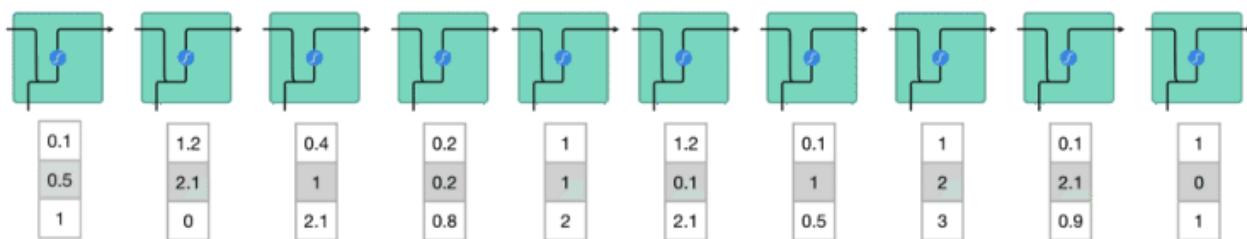
5
0.1
-0.5



3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

With Sequence Input

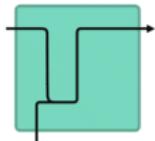
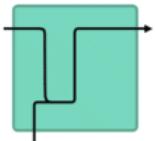
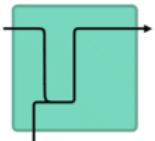
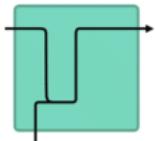


3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

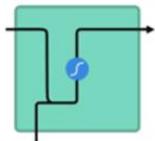
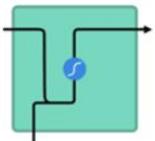
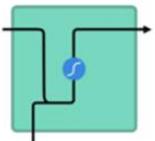
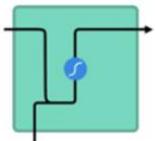
Q: *Why do we need tanh function?*

5
0.01
-0.5



Vector Transformations without tanh

5
0.01
-0.5



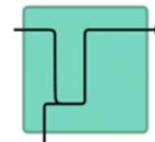
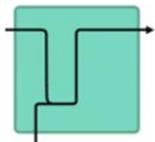
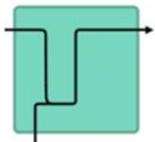
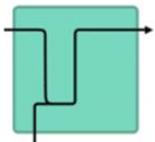
Vector Transformations with tanh

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

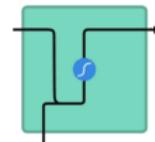
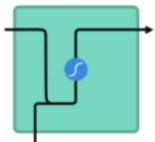
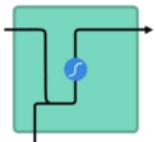
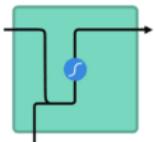
Q: *Why do we need tanh function?*

5
0.01
-0.5



Vector Transformations without tanh

5
0.01
-0.5

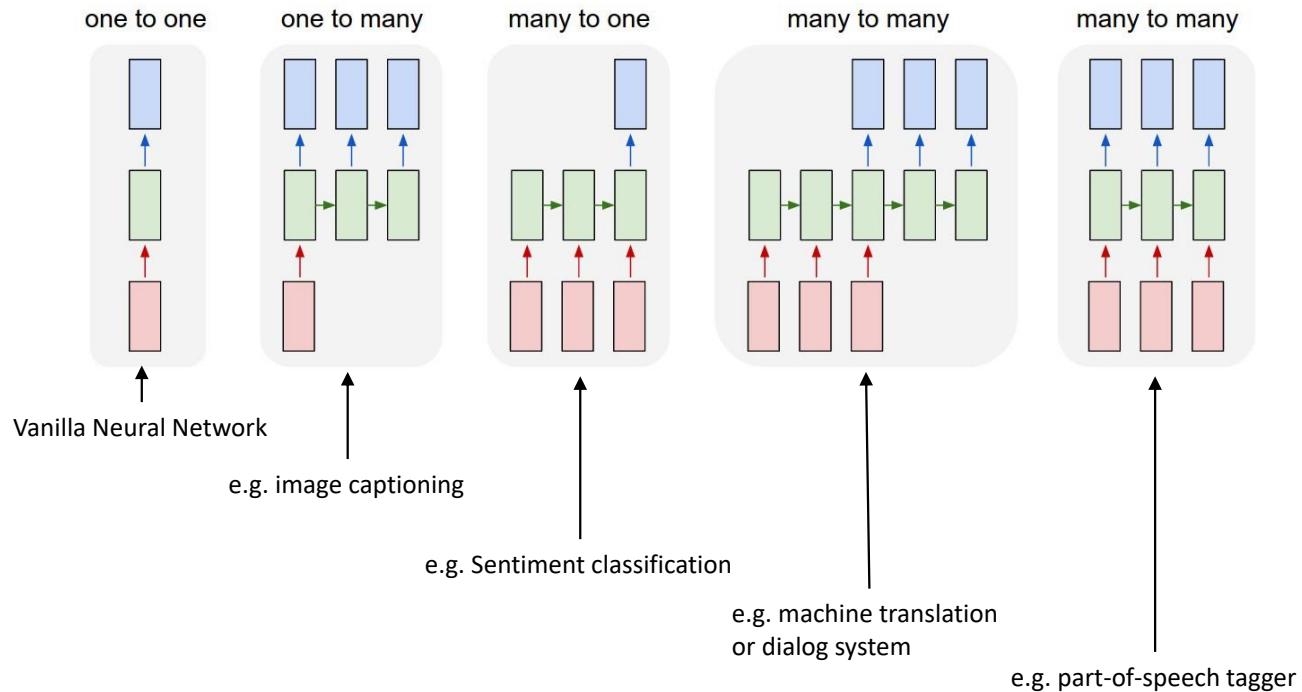


Vector Transformations with tanh

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

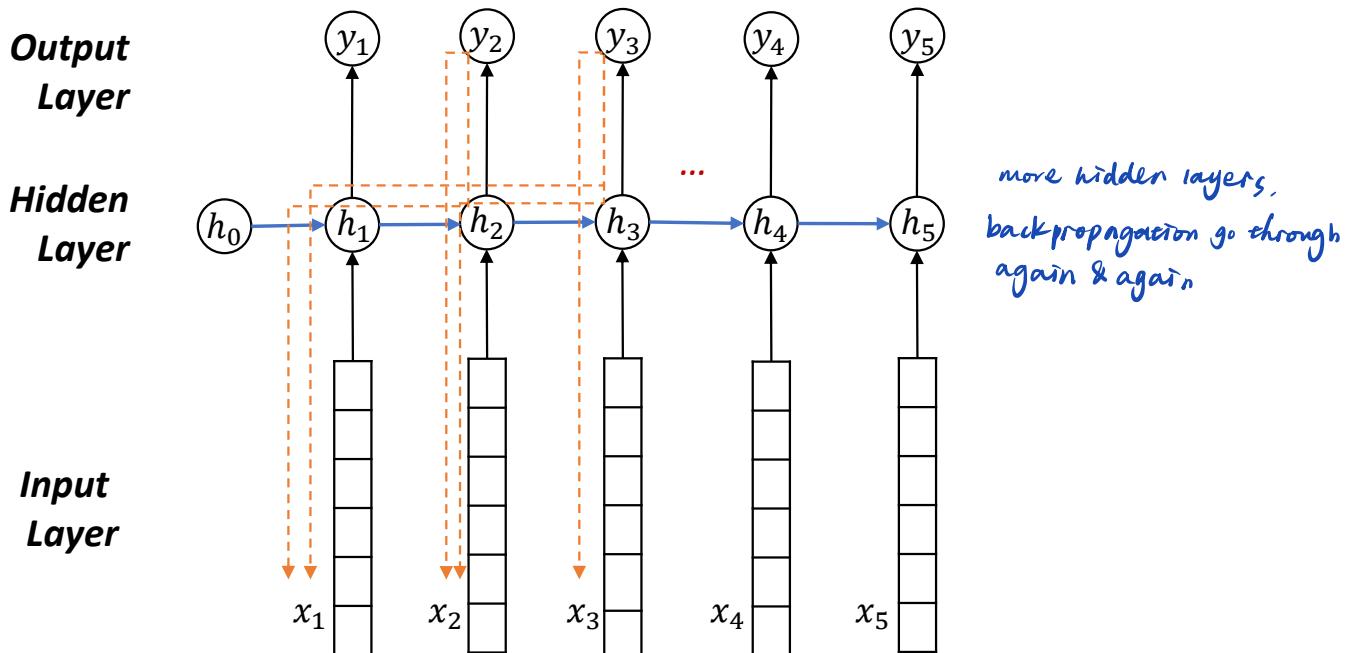
Several Variants of RNN



3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

Backpropagation through time

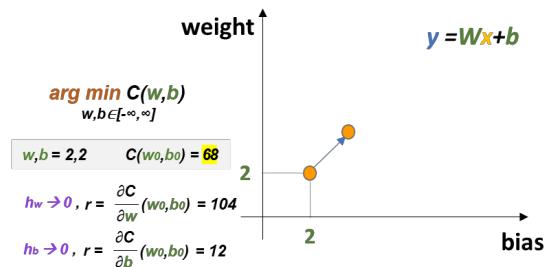
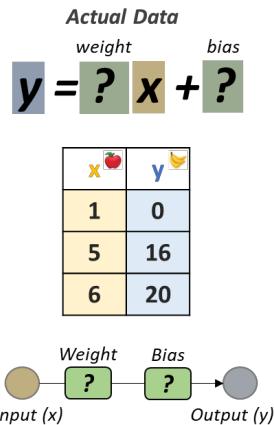
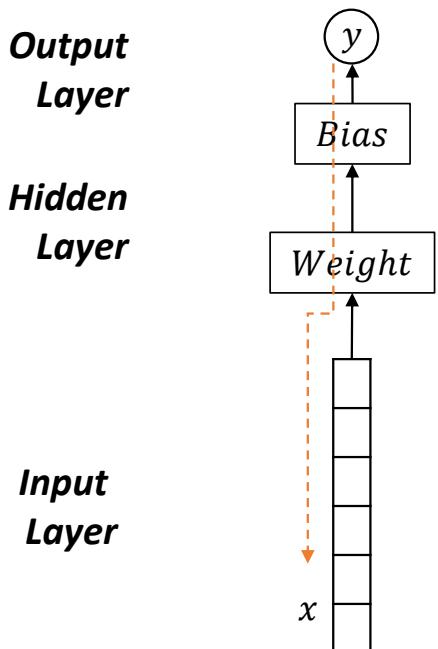


- Similar as **standard backpropagation** on unrolled network
- Similar as **training very deep networks** with tied parameters

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

Remember? Backpropagation in the basic neural networks

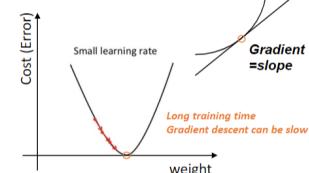
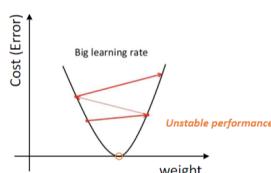


$$w_1 = w_0 - r_w a$$

$$b_1 = b_0 - r_b a$$

a = learning rate or step size

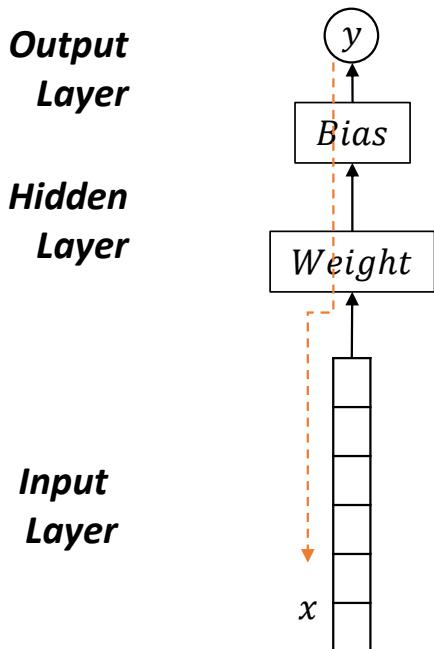
$$\arg \min c(w, b)$$



3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

Remember? Backpropagation in the basic neural networks



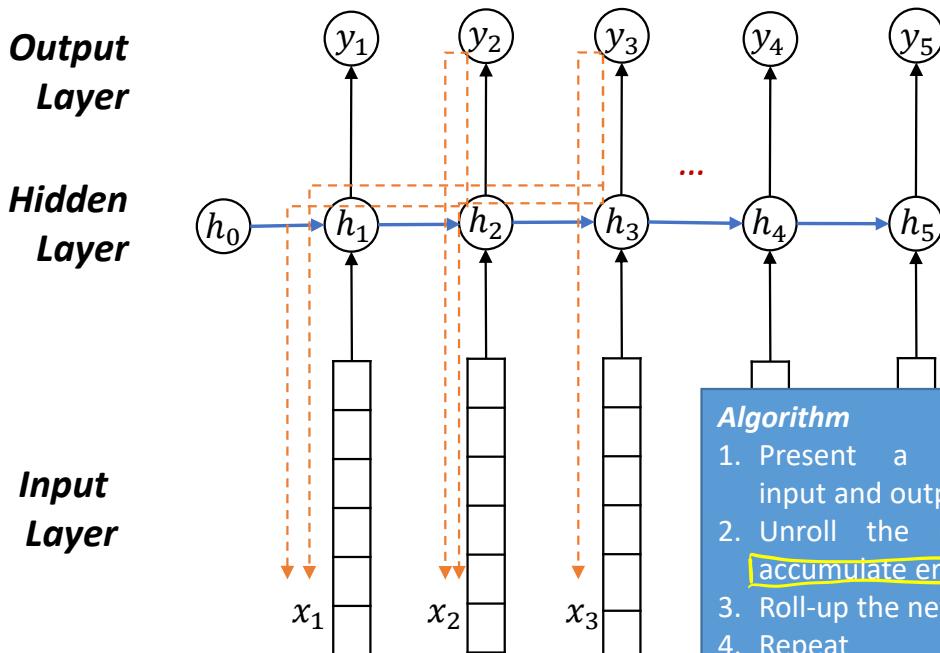
Algorithm

1. Present a training input pattern and propagate it through the network to get an output.
2. Compare the predicted outputs to the expected outputs and calculate the error.
3. Calculate the derivatives of the error with respect to the network weights.
4. Adjust the weights to minimize the error.
5. Repeat

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

Backpropagation through time



Algorithm

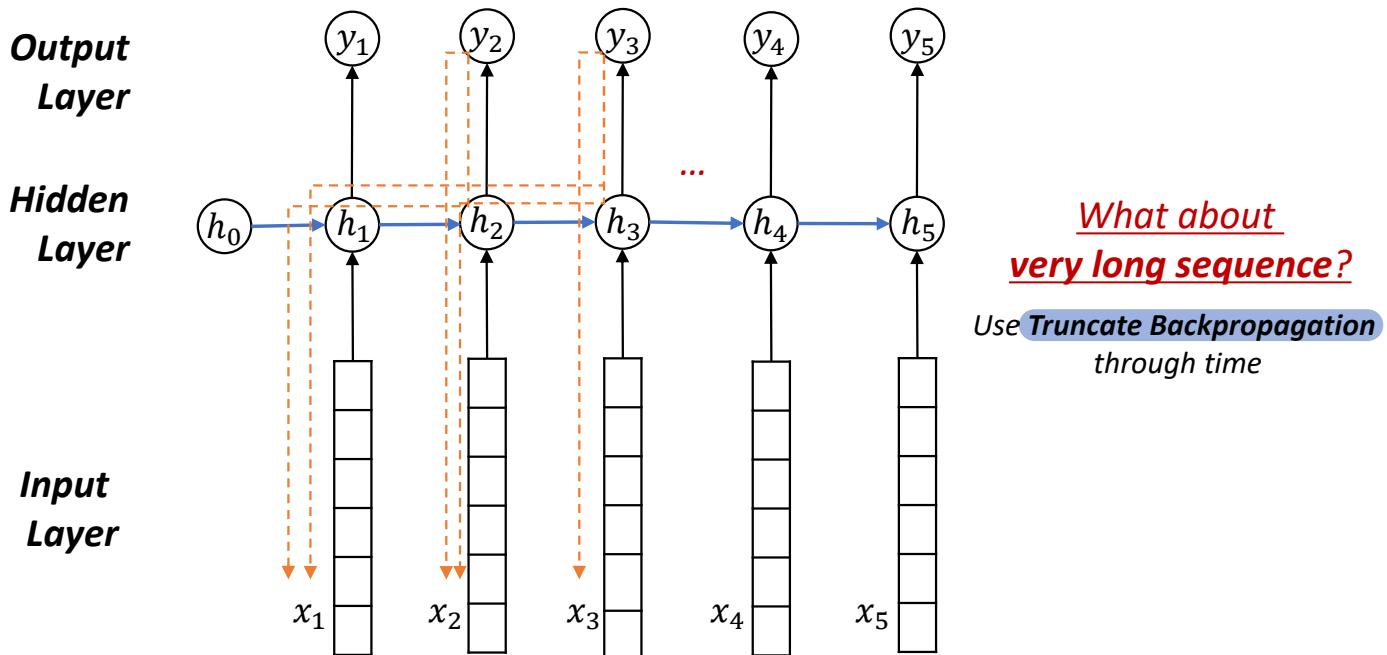
1. Present a sequence of timesteps of input and output pairs to the network.
2. Unroll the network then calculate and **accumulate errors across each timestep**
3. Roll-up the network and update weights.
4. Repeat

- *Similar as standard backpropagation on unrolled network*
- *Similar as training very deep networks with tied parameters*

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

Backpropagation through time

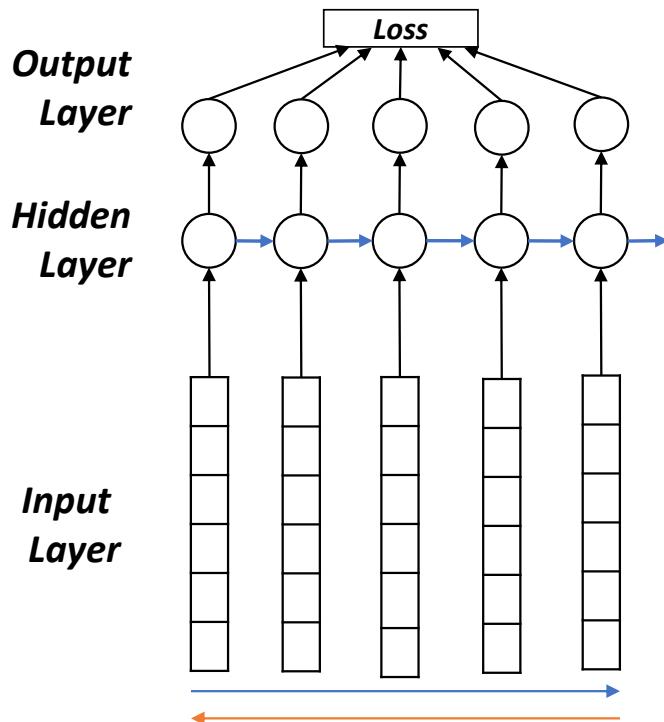


- Similar as **standard backpropagation** on unrolled network
- Similar as **training very deep networks** with tied parameters

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

Truncated Backpropagation through time

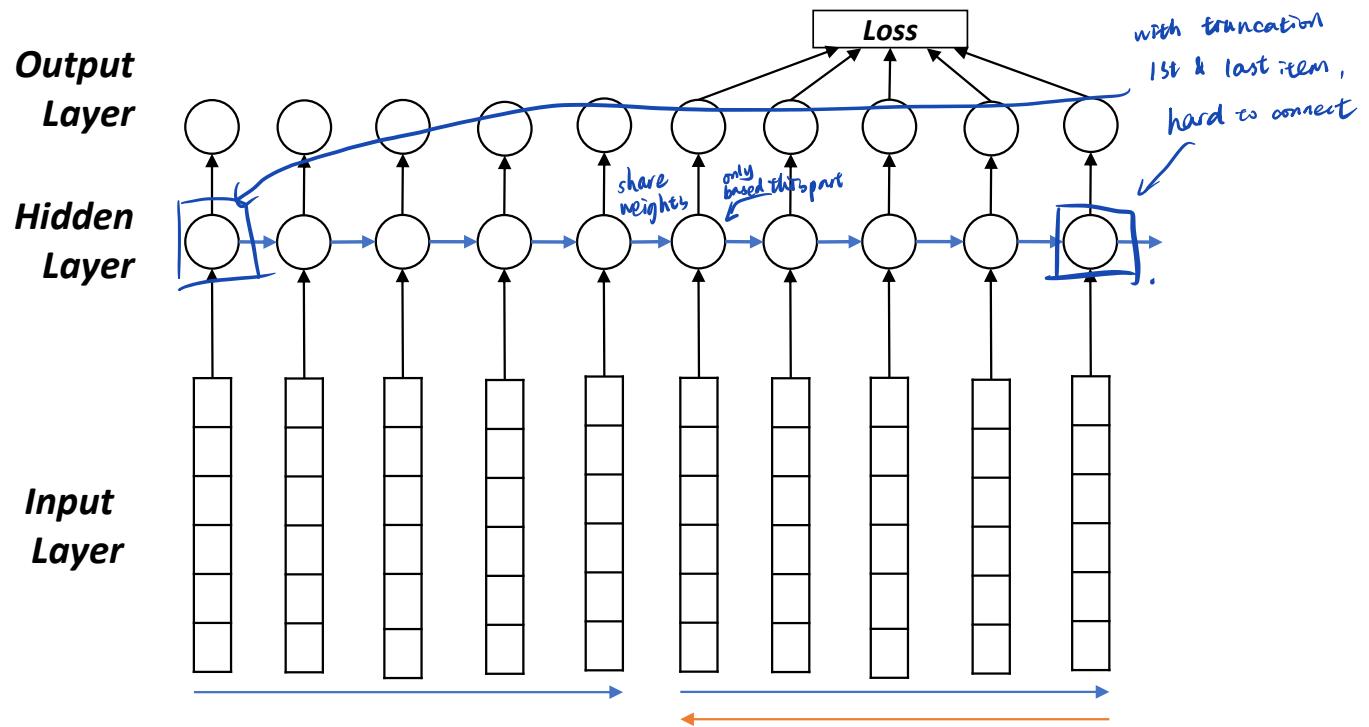


Run forward and backward through chunks of the sequence instead of whole sequence

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

Truncated Backpropagation through time

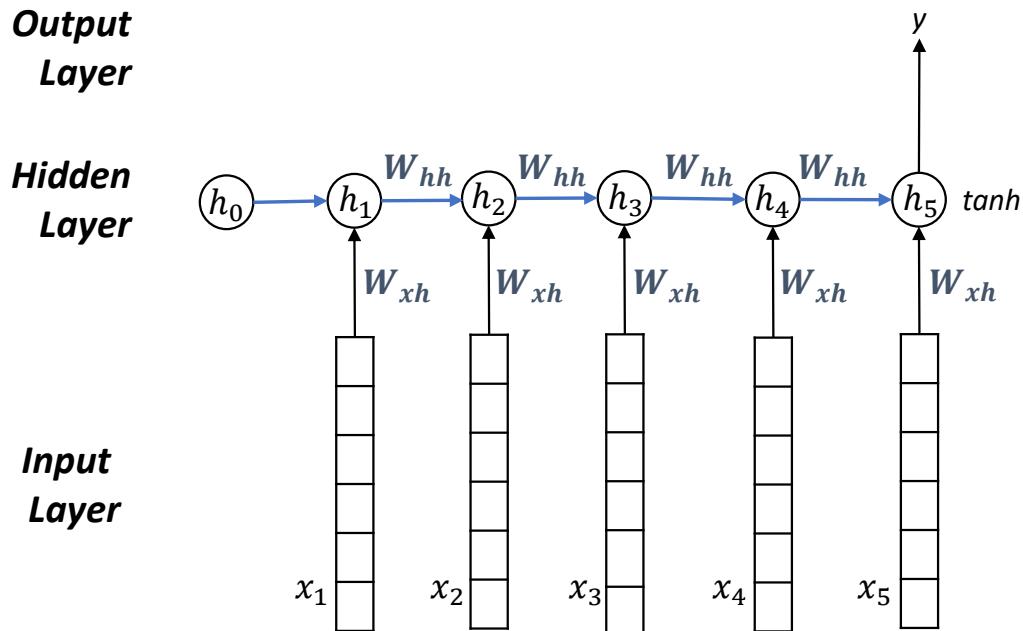


Carry hidden states forward in time forever, but only backpropagate for some smaller number of steps

3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

Many to 1



3 Seq2Seq with Deep Learning

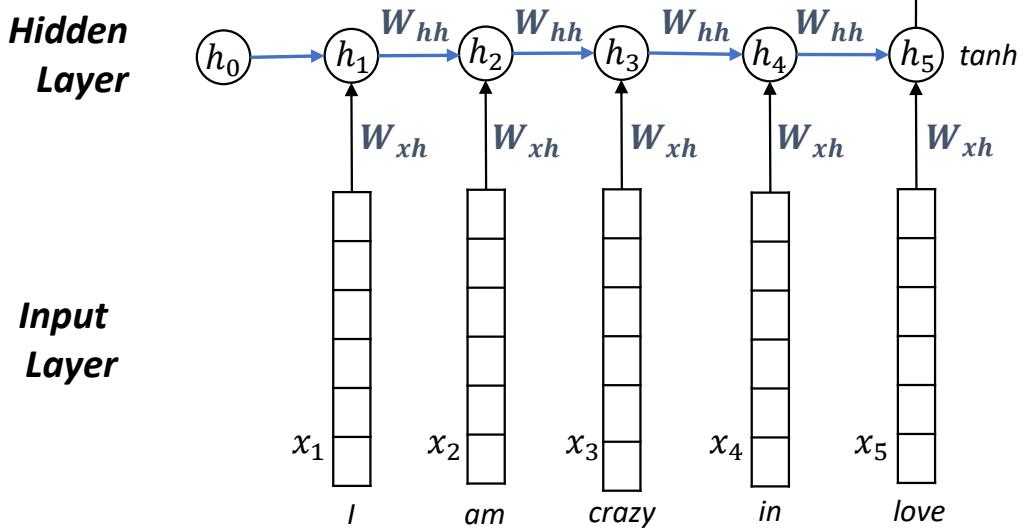
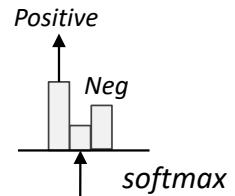
Neural Network + Memory = Recurrent Neural Network

Many to 1 – Text Classification

**Output
Layer**

**Hidden
Layer**

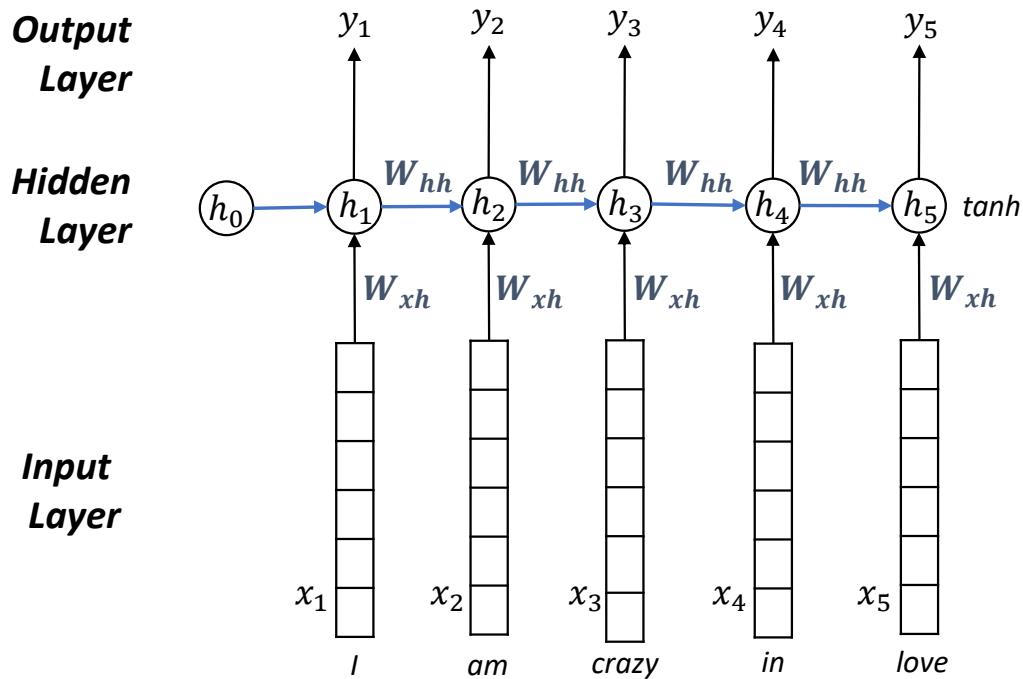
**Input
Layer**



3 Seq2Seq with Deep Learning

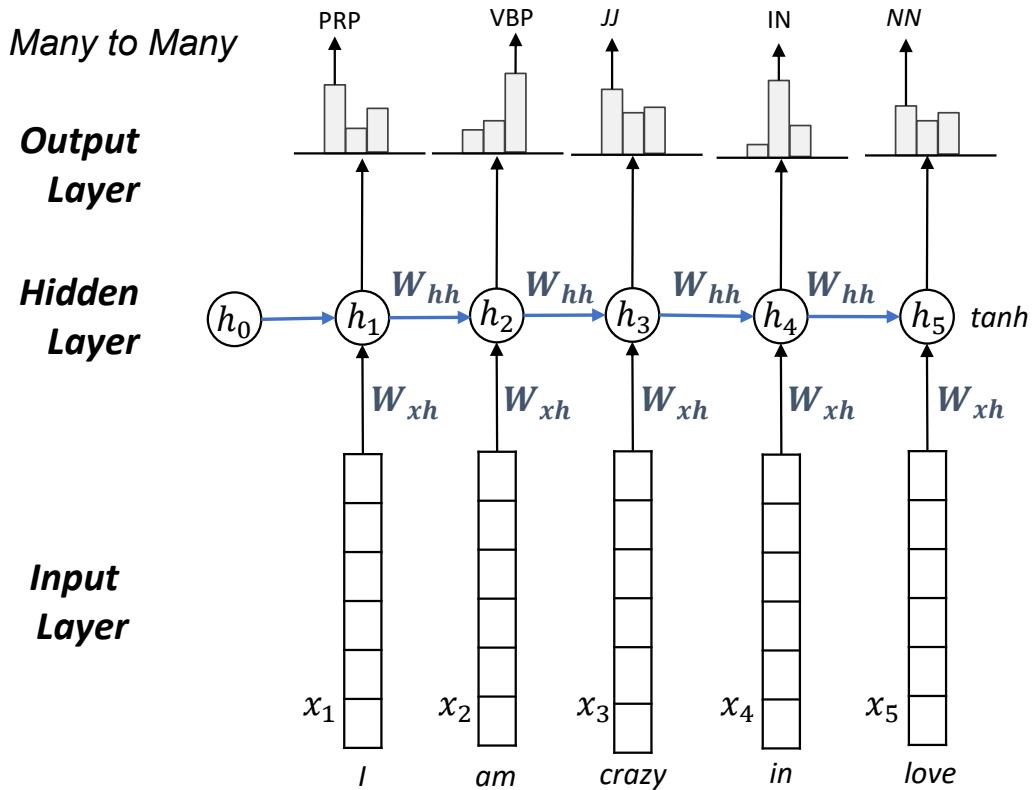
Neural Network + Memory = Recurrent Neural Network

Many to Many



3 Seq2Seq with Deep Learning

Neural Network + Memory = Recurrent Neural Network

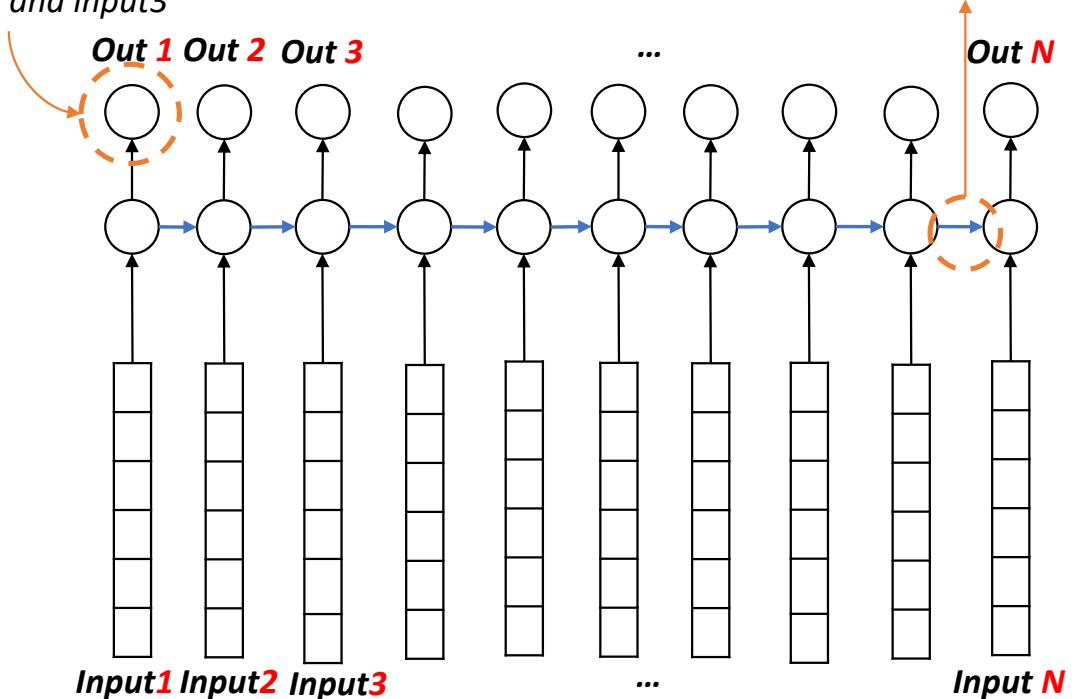


3 Seq2Seq with Deep Learning

Limitation of Vanilla RNN

Out1 does not cover the input2 and input3

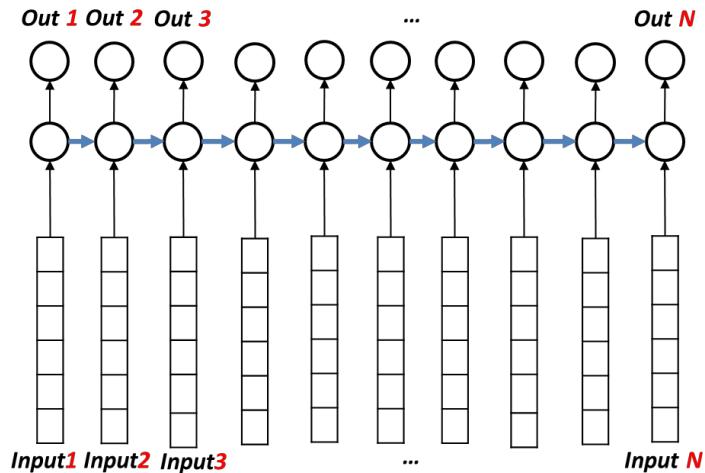
If Input1 is too far, it cannot cover the input1



The Problem of Learning Long-Range Dependencies

3 Seq2Seq with Deep Learning

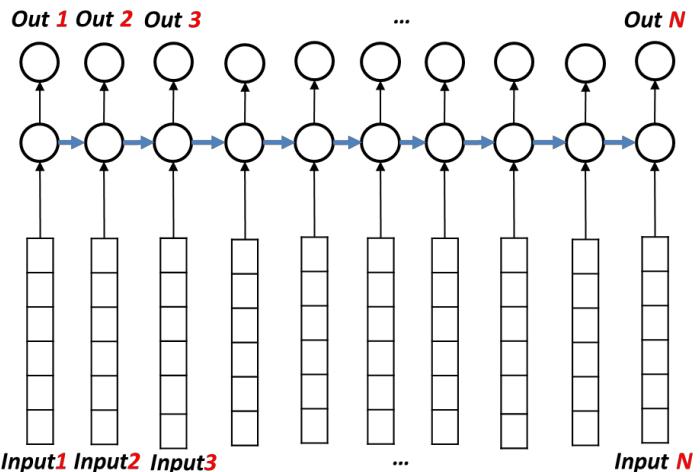
Limitation of Vanilla RNN



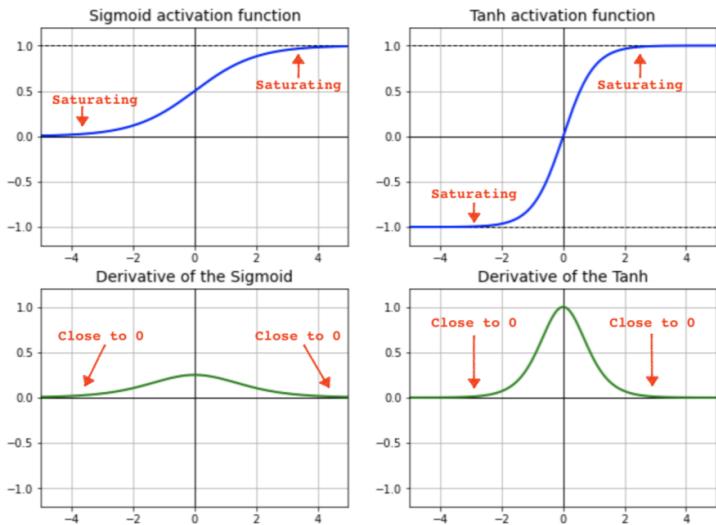
*“I grew up in Italy ... (5 more sentences)...
My grandma’s house was very cosy and...
(5 more sentences)... I speak fluent ____”*

3 Seq2Seq with Deep Learning

Limitation of Vanilla RNN



Assume that f is activation function

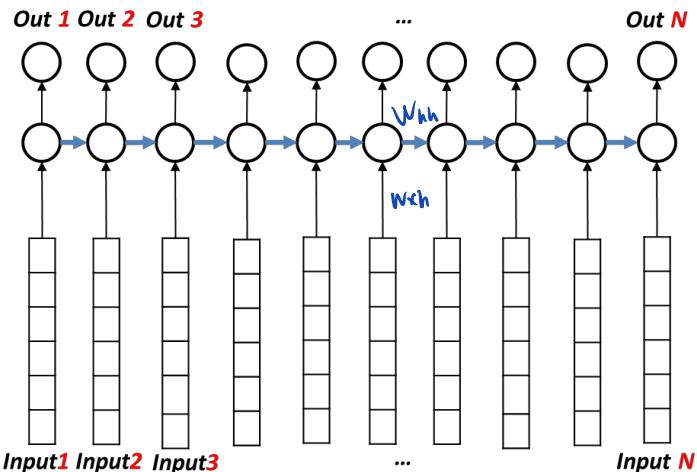


*"I grew up in Italy ... (5 more sentences)...
 My grandma's house was very cosy and...
 (5 more sentences)... I speak fluent ____"*

derivative of tanh \rightarrow derivative of sigmoid
 \downarrow
 "distinguishable" (faster)

3 Seq2Seq with Deep Learning

Limitation of Vanilla RNN



(-1.1)

Assume that f is activation function

$$O_5 = f(f(f(f(f(x, W_{xh})W_{hh})W_{hh})W_{hh})W_{hh})$$

$$O_5 \approx x \cdot W_{xh} \cdot W_{hh}^4$$

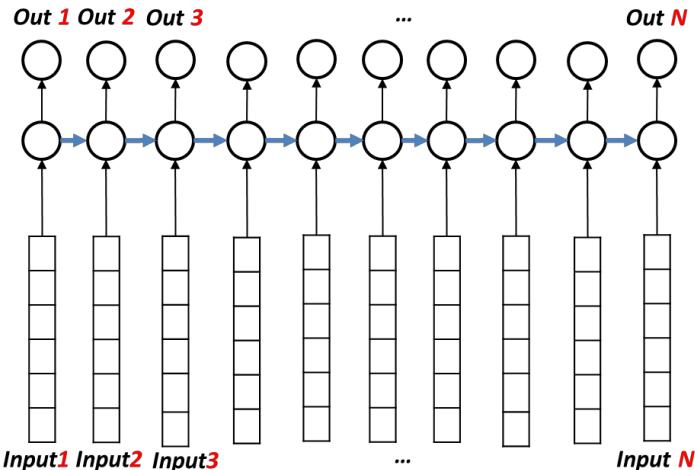
Lower than 1
Exponential decay

Higher than 1
Exponential grow

*"I grew up in Italy ... (5 more sentences)...
 My grandma's house was very cosy and...
 (5 more sentences)... I speak fluent ____"*

3 Seq2Seq with Deep Learning

Limitation of Vanilla RNN



Assume that f is activation function

$$O_5 = f(f(f(f(f(x, W_{xh})W_{hh})W_{hh}) W_{hh}) W_{hh})$$

$$O_5 \approx x \cdot W_{xh} \cdot W_{hh}^4$$

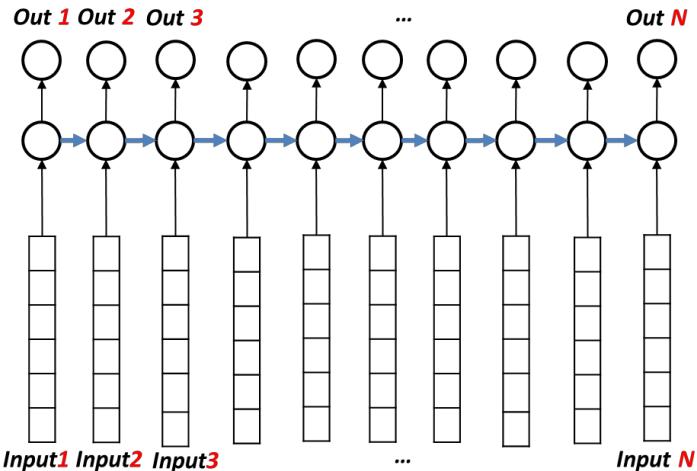
Lower than 1
Exponential decay
 Vanishing gradient
 Higher than 1
Exponential grow

Limitation 1: Vanishing Gradient Issue

During back-propagation and calculating gradients, it tends to get smaller and smaller as we keep on moving backward in the Network. This means that the neurons in the Earlier layers learn very slowly as compared to the neurons in the later layers in the Hierarchy.

3 Seq2Seq with Deep Learning

Limitation of Vanilla RNN



Assume that f is activation function

$$O_5 = f(f(f(f(f(x, W_{xh})W_{hh})W_{hh}) W_{hh}) W_{hh})$$

$$O_5 \approx x \cdot W_{xh} \cdot W_{hh}^4$$

Lower than 1
Exponential decay

Higher than 1
Exponential grow

Limitation2: Exploding Gradient

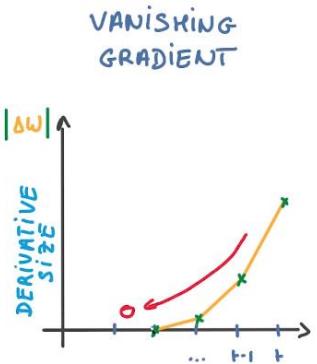
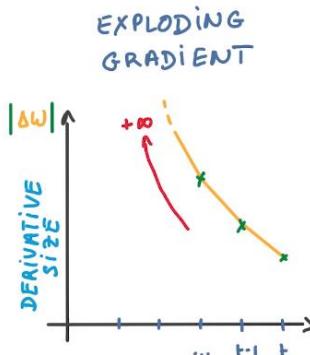
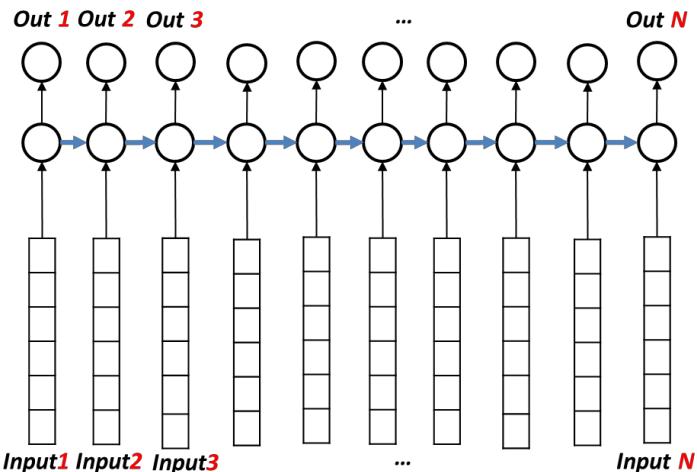
In RNN, error gradients can accumulate during an update and result in very large gradients.

These in turn result in large updates to the network weights, and an unstable network.

At an extreme, the values of weights can become so large as to overflow and result in **Nan** weight values that can no longer be updated.

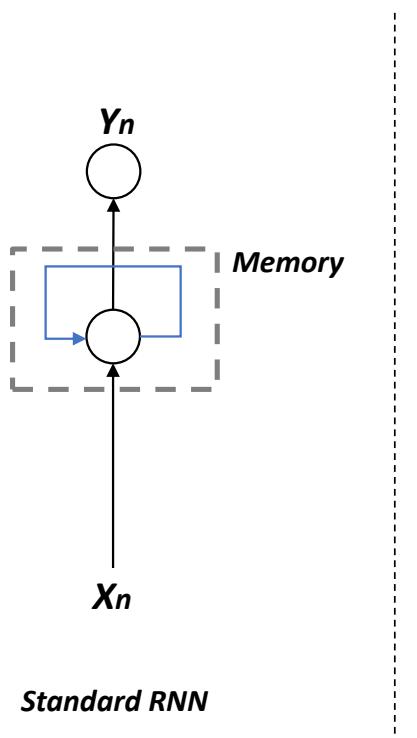
3 Seq2Seq with Deep Learning

Limitation of Vanilla RNN



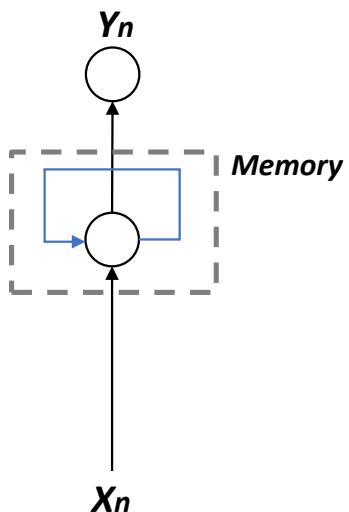
3 Seq2Seq with Deep Learning

LSTM (Long Short-Term Memory) - Idea

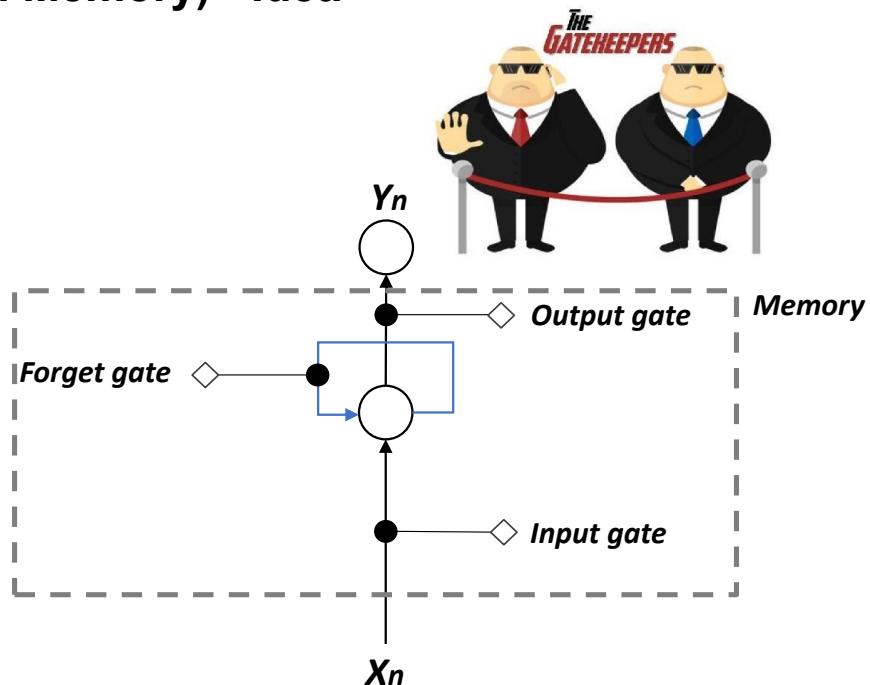


3 Seq2Seq with Deep Learning

LSTM (Long Short-Term Memory) - Idea



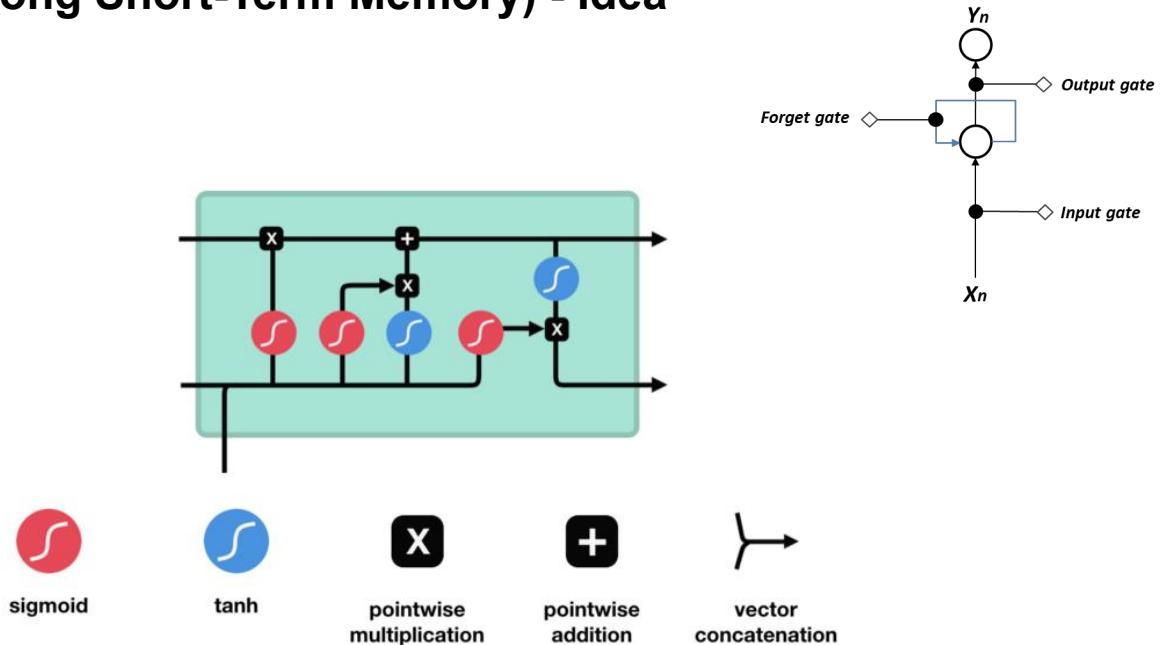
Standard RNN



Long Short-Term Memory

3 Seq2Seq with Deep Learning

LSTM (Long Short-Term Memory) - Idea



- 4 times more parameters than RNN
- Mitigates **vanishing gradient** problem through **gating**
- Widely used and was SOTA in many sequence learning problems

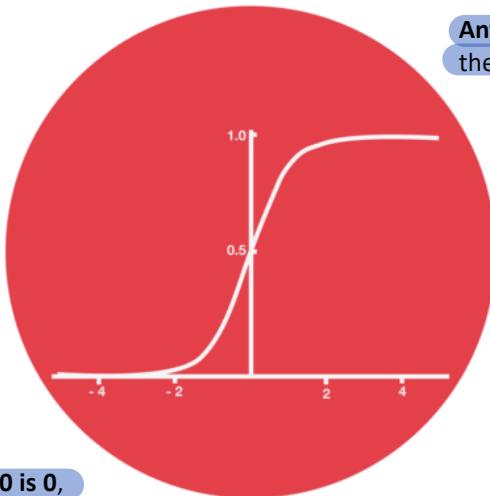
State-Of-The-Art

3 Seq2Seq with Deep Learning

Sigmoid activation

A sigmoid activation is similar to the tanh activation. Instead of squishing values between -1 and 1, it squishes values between 0 and 1.

5
0.1
-0.5

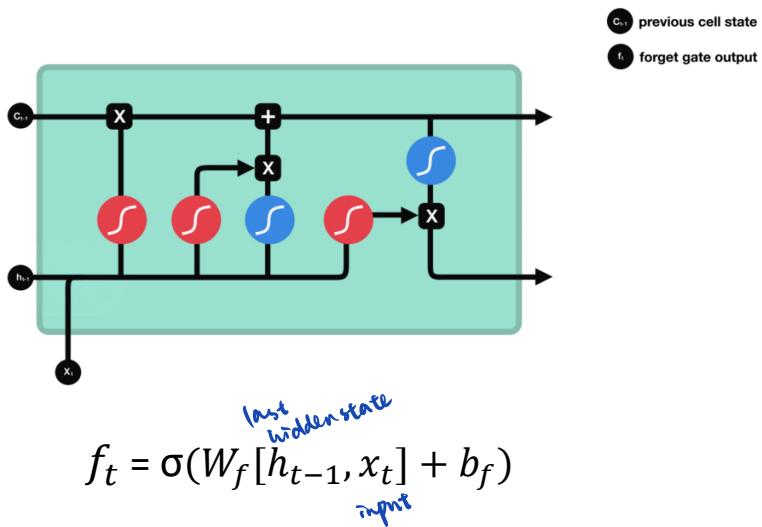


Any number multiplied by 1 is the same value
therefore that value stays the same or is “kept.”

Any number getting multiplied by 0 is 0,
causing values to disappear or be “forgotten.”

3 Seq2Seq with Deep Learning

LSTM (Long Short-Term Memory) – Forget Gate

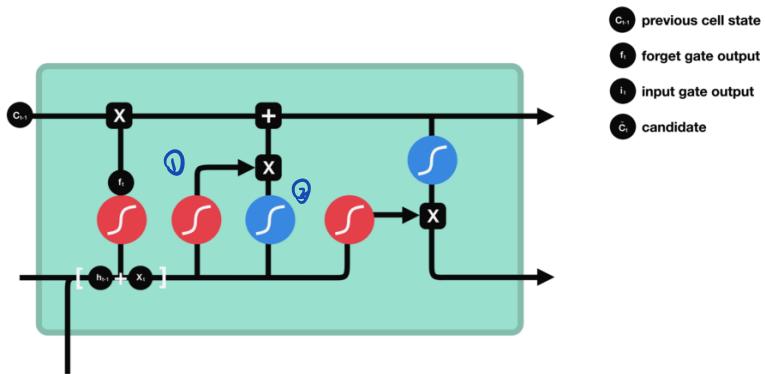


Decides what information should be thrown away or kept

Information from the previous hidden state and information from the current input is passed through the **sigmoid function**. Values come out between 0 and 1. The closer to 0 means to forget, and the closer to 1 means to keep.

3 Seq2Seq with Deep Learning

LSTM (Long Short-Term Memory) – Input Gate



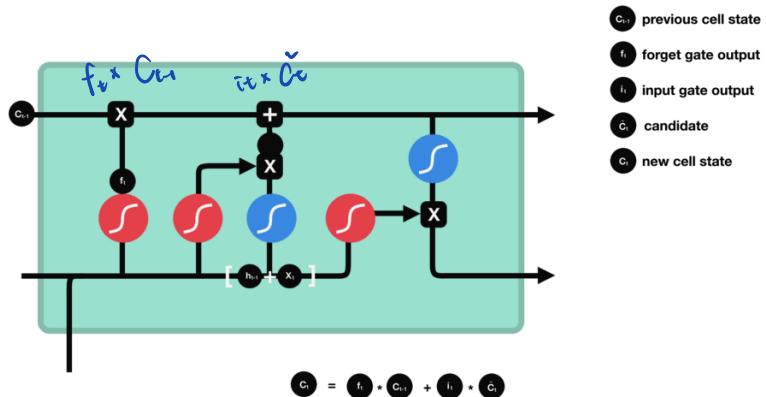
$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$

$$\tilde{c}_t = \tanh(W_C[h_{t-1}, x_t] + b_C)$$

1. Pass the *previous hidden state and current input* into a *sigmoid function*
 2. Pass the *hidden state and current input* into the *tanh function* to *squish values between -1 and 1 to help regulate the network*
 3. *Multiply the tanh output with the sigmoid output*
- *sigmoid output will decide which information is important to keep from the tanh output

3 Seq2Seq with Deep Learning

LSTM (Long Short-Term Memory) – Cell States

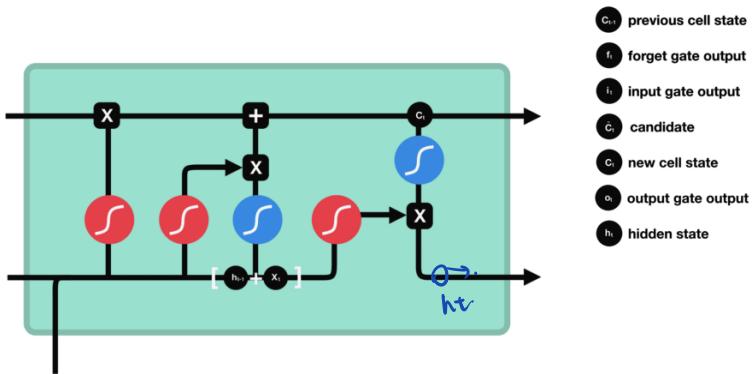


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

- the cell state gets pointwise multiplied by the forget vector
- take the output from the input gate and do a pointwise addition which updates the cell state to new values that the neural network finds relevant
- That gives us our new cell state

3 Seq2Seq with Deep Learning

LSTM (Long Short-Term Memory) – Output Gate



$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

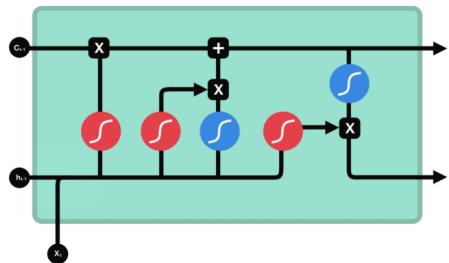
decides what the next hidden state should be.

- pass the previous hidden state and the current input into a sigmoid function
- pass the newly modified cell state to the tanh function
- multiply the tanh output with the sigmoid output to decide what information the hidden state should carry

3 Seq2Seq with Deep Learning

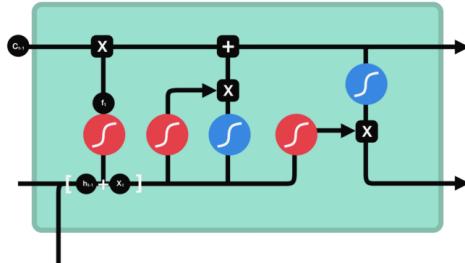
LSTM (Long Short-Term Memory) - Overall

Forget gate



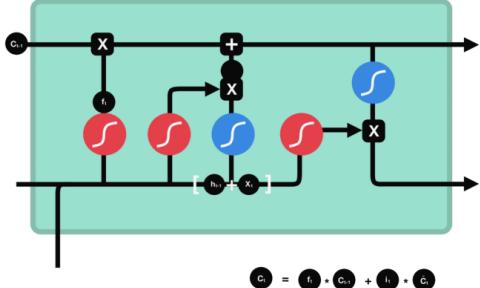
c_{t-1} previous cell state
 f_t forget gate output

Input gate



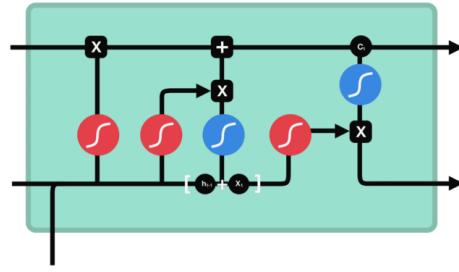
c_{t-1} previous cell state
 f_t forget gate output
 i_t input gate output
 c_t^* candidate

Cell state



c_{t-1} previous cell state
 f_t forget gate output
 i_t input gate output
 c_t^* candidate
 c_t new cell state

Output gate

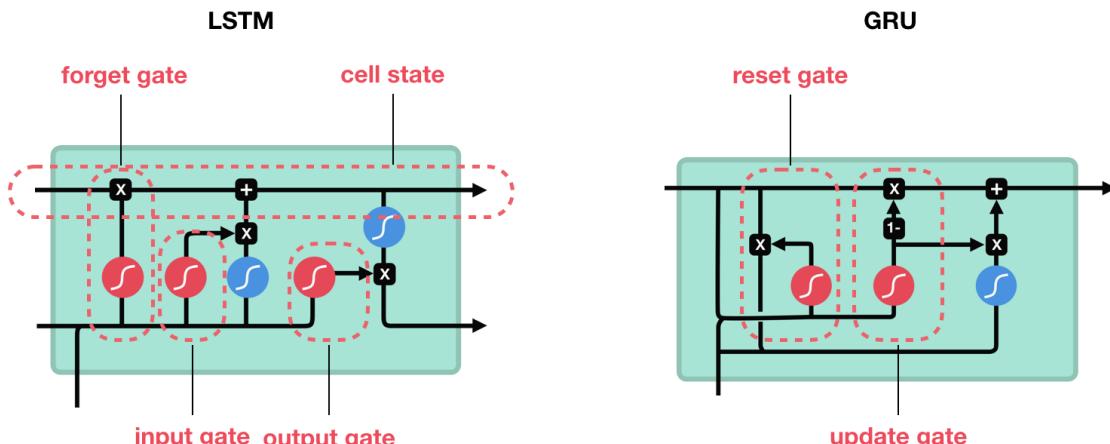


c_{t-1} previous cell state
 f_t forget gate output
 i_t input gate output
 c_t^* candidate
 c_t new cell state
 o_t output gate output
 h_t hidden state

$$c_t = f_t \cdot c_{t-1} + i_t \cdot c_t^*$$

3 Seq2Seq with Deep Learning

Gated Recurrent Unit



sigmoid



tanh



pointwise multiplication



pointwise addition

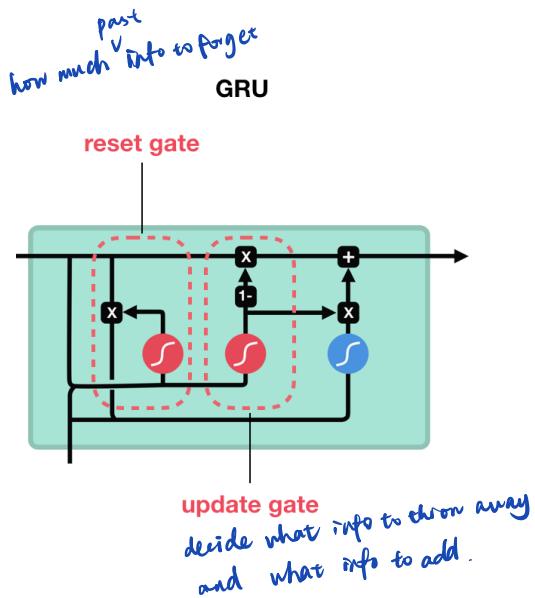


vector concatenation

3 Seq2Seq with Deep Learning

Gated Recurrent Unit

- GRU first computes an **update gate** based on **current input word vector** and **hidden state**
- Compute reset gate similarly but with different weights
 - If reset gate unit is ~ 0 , then this ignores previous memory and only stores the new word information
- Final memory at time step combines current and previous time steps



3 Seq2Seq Modelling

Seq2Seq – PoS tagger

ADV VERB DET NOUN NOUN

Output: Part of Speech



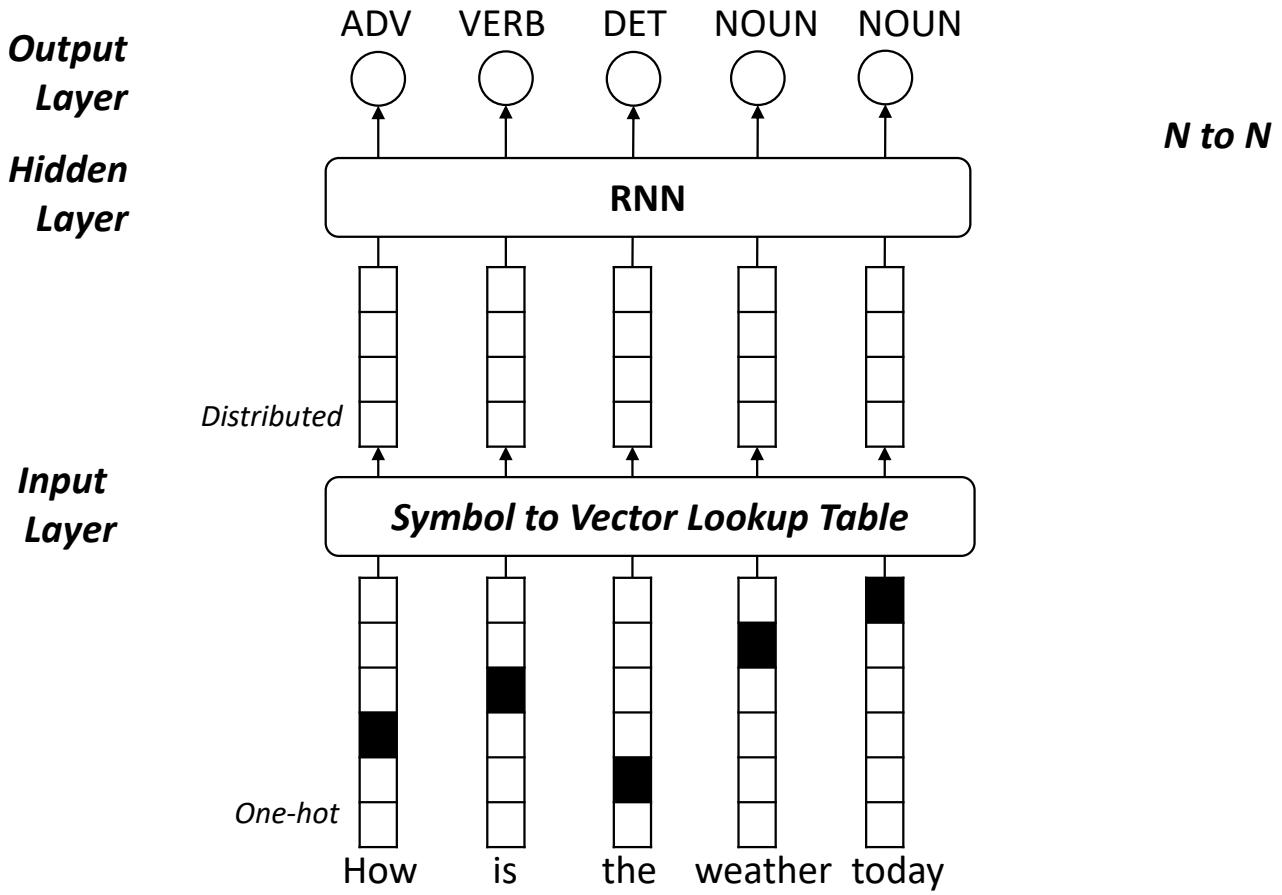
Sequence 2 Sequence Learning

How is the weather today

Input: Text

3 Seq2Seq Modelling

Sequence Modelling for POS Tagging



0 LECTURE PLAN

Lecture 4: Word Classification and Machine Learning 2

1. Machine Learning and NLP: Finish
2. Seq2Seq Learning
3. Seq2Seq Deep Learning
 1. RNN (Recurrent Neural Network)
 2. LSTM (Long Short-Term Memory)
 3. GRU (Gated Recurrent Unit)
4. **Data Transformation for Deep Learning NLP**
5. Next Week Preview
 - Natural Language Processing Stack

ImageNet: Image Classification

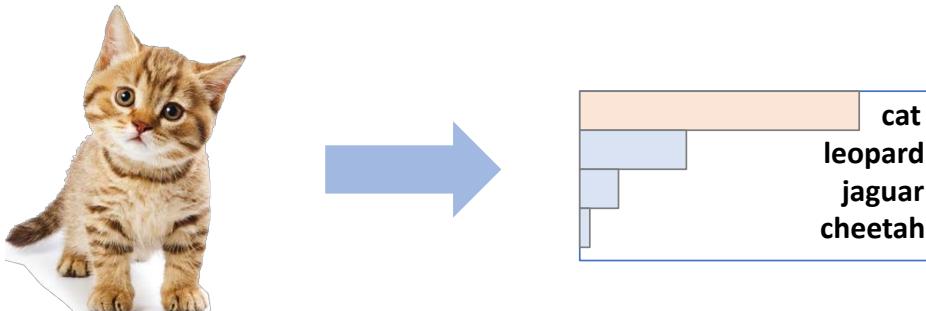
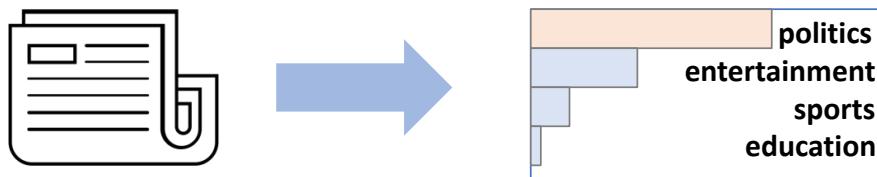


Image Pixel

Topic Classification



News Articles

Visual Question Answering



Visual Question Answering



What color of the shirt does he wear

Submit

Predicted top-5 answers with confidence:

orange

99.999%

yellow

0.001%

orange and
white

0.000%

yellow and
orange
orange and
black

0.000%

0.000%

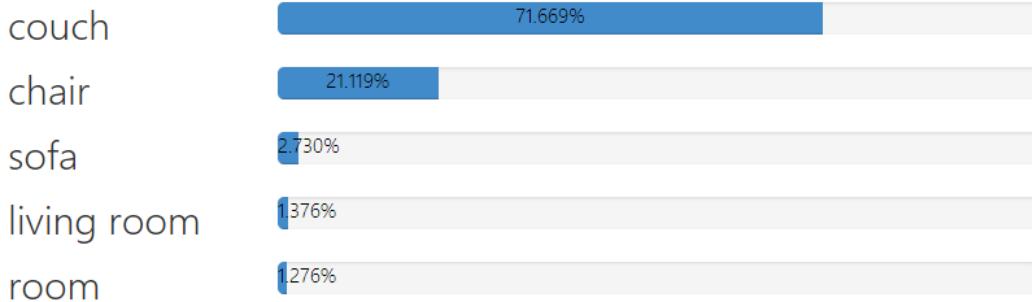
Visual Question Answering



Where is he sitting

Submit

Predicted top-5 answers with confidence:



Visual Question Answering



Why is he surprised

Submit

Predicted top-5 answers with confidence:

playing

36.734%

game

13.713%

game

5.481%

playing

5.481%

video games

4.104%

playing wii

4.104%

hungry

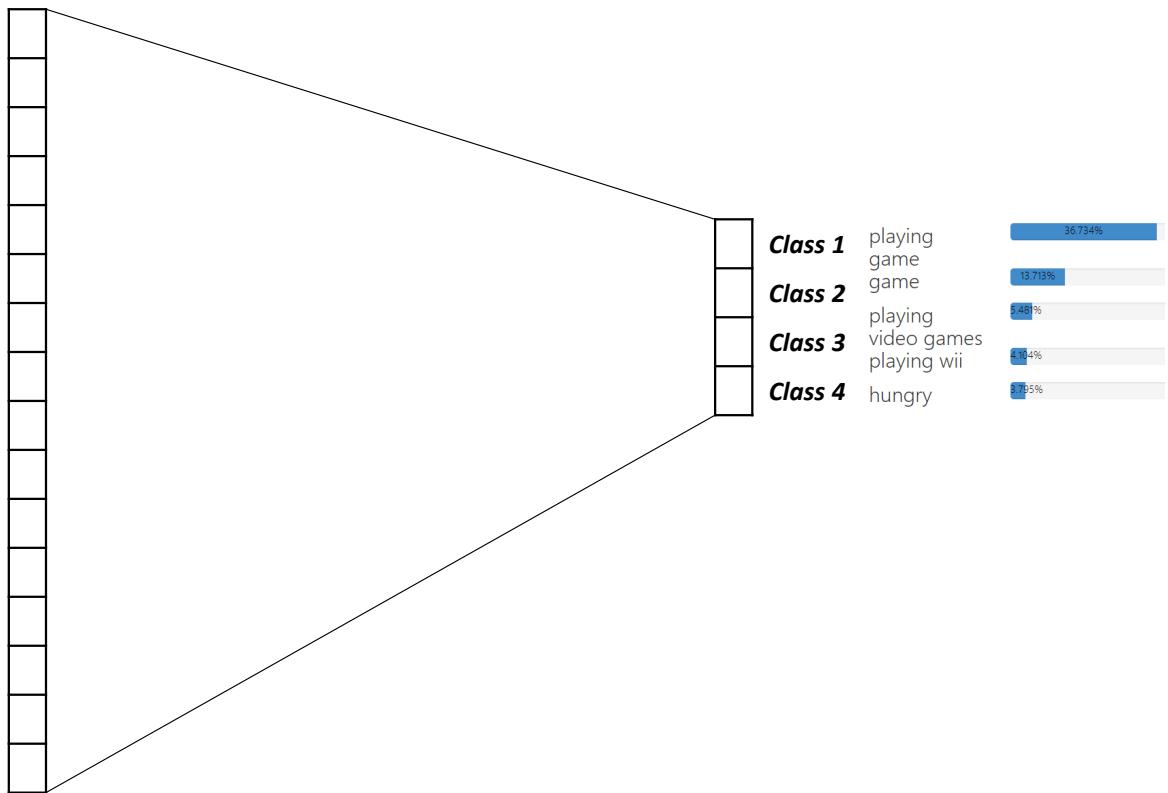
3.795%

Classification Formulation

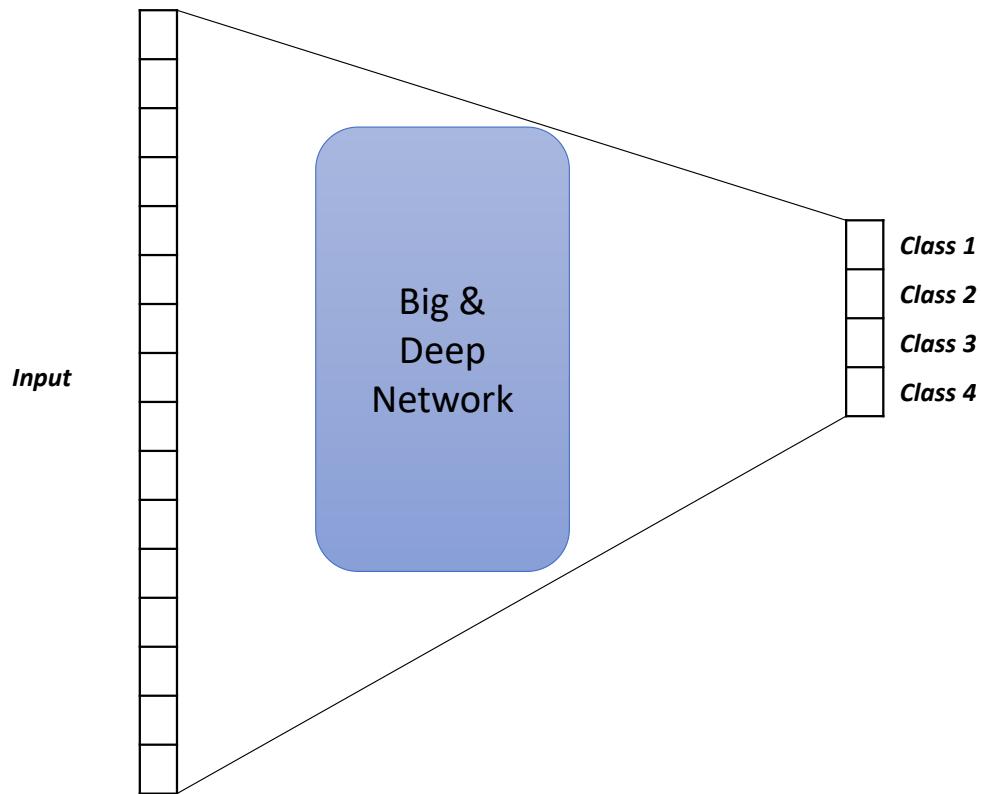


Input

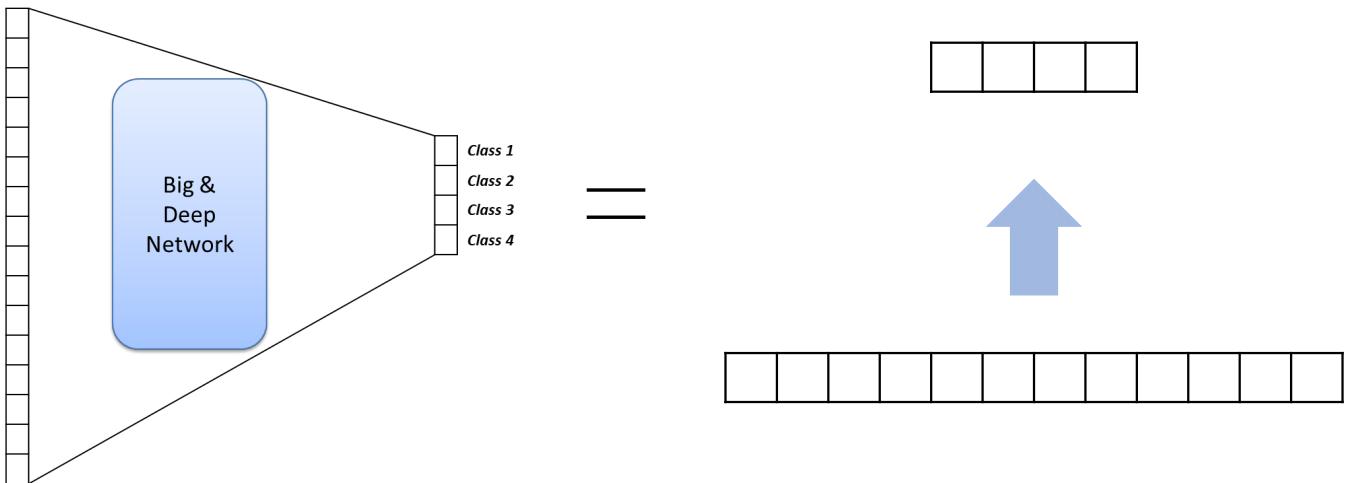
Why is he surprised



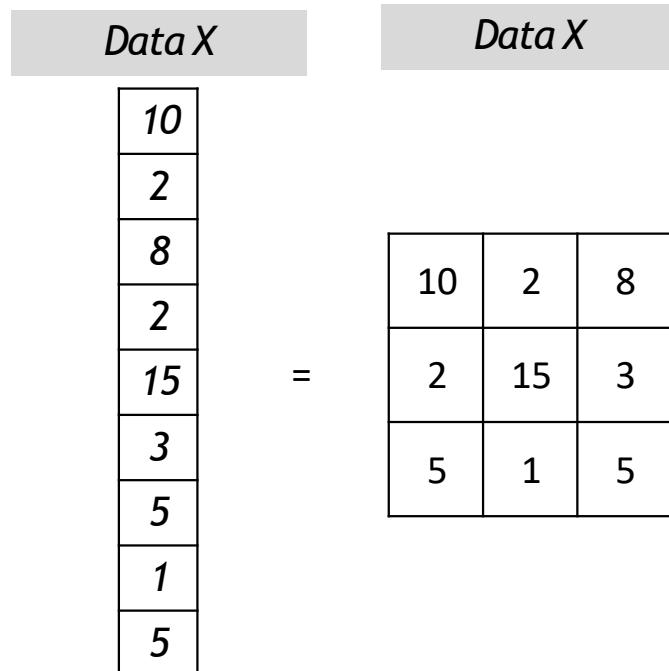
Classification Formulation



Classification



Graphical Notation for Data



4

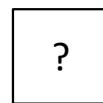
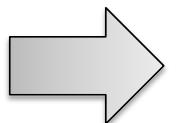
Data Transformation for Deep Learning NLP

V to 1

10
2
8
2
15
3
5
1
5

=

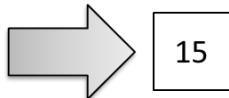
10	2	8
2	15	3
5	1	5



V to 1 – Simple Method

center one

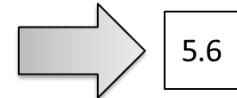
10	2	8
2	15	3
5	1	5



15

average

10	2	8
2	15	3
5	1	5



5.6

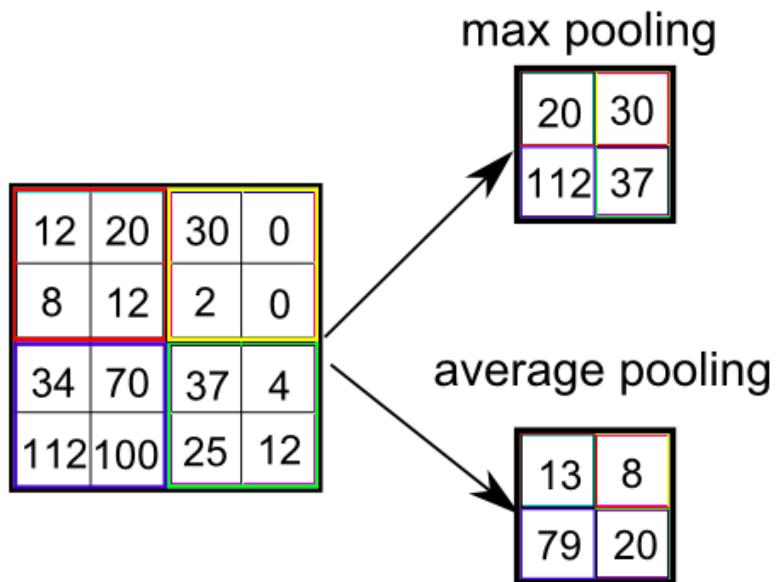
median

10	2	8
2	15	3
5	1	5



5

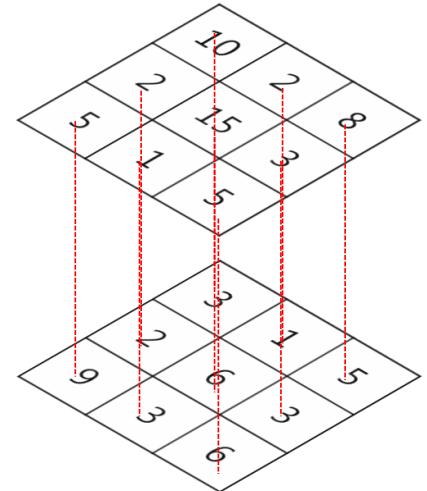
V to 1 – Simple Method



V to 1 – Weighted Method

Weighted Sum								
Value			Weight					
10	2	8						
2	15	3						
5	1	5						

→ 253

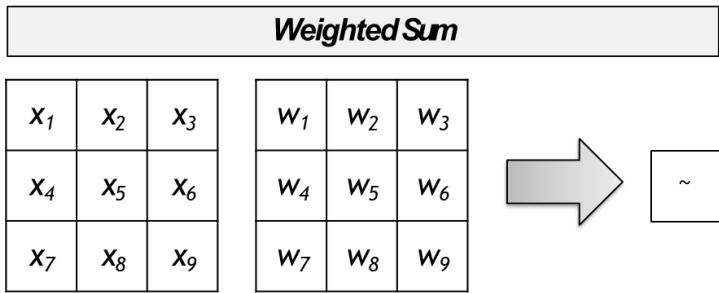


Element-wise multiplication

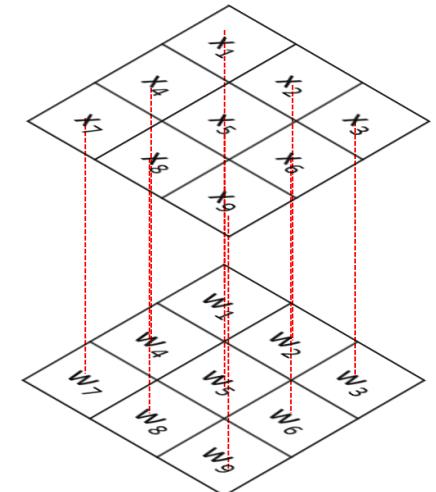
Weighted Average								
Value			Weight					
10	2	8						
2	15	3						
5	1	5						

→ 6.65

V to 1 – General Form



$$v = x_1 * w_1 + x_2 * w_2 + \dots + x_9 * w_9$$



Element-wise multiplication

V to 1 – Linear Algebra

[9x1] matrix

w_1
w_2
w_3
w_4
w_5
w_6
w_7
w_8
w_9

[1 x 9] matrix

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
-------	-------	-------	-------	-------	-------	-------	-------	-------

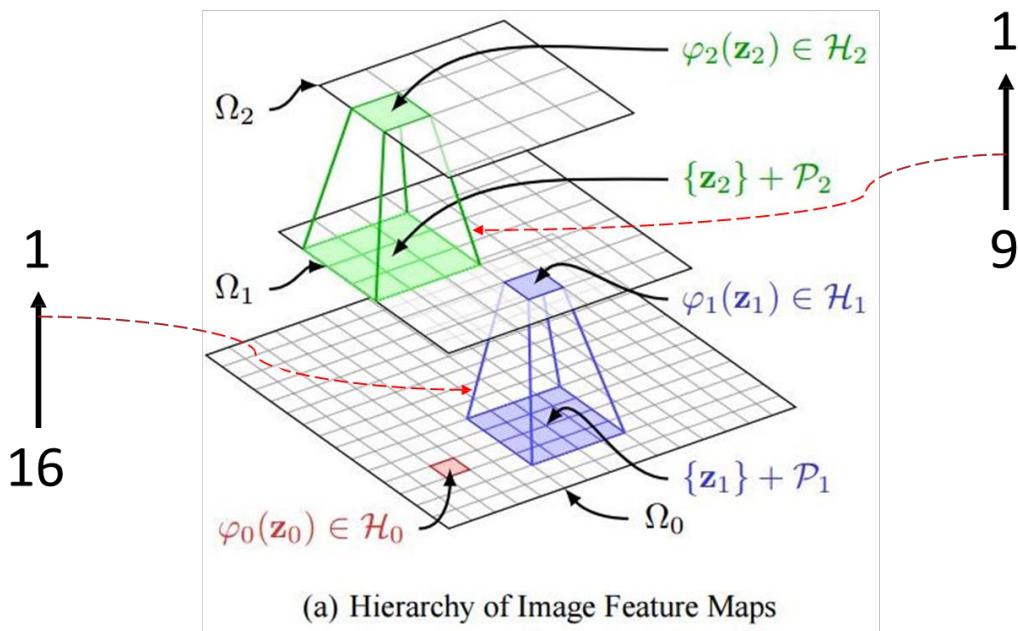
X

[1x1] matrix

$$\sum_i^9 x_i * w_i$$

=

Convolution Neural Network (1)



Data Abstraction

Convolution Neural Network (2)

1	0	1
0	1	0
1	0	1

filter

1 <small>$\times 1$</small>	1 <small>$\times 0$</small>	1 <small>$\times 1$</small>	0	0
0 <small>$\times 0$</small>	1 <small>$\times 1$</small>	1 <small>$\times 0$</small>	1	0
0 <small>$\times 1$</small>	0 <small>$\times 0$</small>	1 <small>$\times 1$</small>	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved
Feature

Convolution Neural Network (2)

1	0	1
0	1	0
1	0	1

filter

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

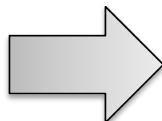
Convolved
Feature

V to V'

$V = 9$

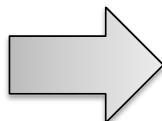
10	2	8
2	15	3
5	1	5

$V' = 2$



?
?

10	2	8	2	15	3	5	1	5
----	---	---	---	----	---	---	---	---



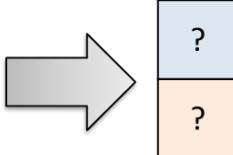
?	?
---	---

V to V' – generalized method

Weighted Sum

x_1	x_2	x_3
x_4	x_5	x_6
x_7	x_8	x_9

$w_{1,1}$	$w_{1,2}$	$w_{1,3}$
$w_{1,4}$	$w_{1,5}$	$w_{1,6}$
$w_{1,7}$	$w_{1,8}$	$w_{1,9}$



$w_{2,1}$	$w_{2,2}$	$w_{2,3}$
$w_{2,4}$	$w_{2,5}$	$w_{2,6}$
$w_{2,7}$	$w_{2,8}$	$w_{2,9}$

$$\left. \begin{array}{l} v_1 = x_1 * w_{1,1} + x_2 * w_{1,1} + \dots + x_9 * w_{1,9} \\ v_2 = x_1 * w_{2,1} + x_2 * w_{2,1} + \dots + x_9 * w_{2,9} \end{array} \right\}$$

V to V' – generalized method

Weighted Sum

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
-------	-------	-------	-------	-------	-------	-------	-------	-------

[1 x 9] matrix

[9x2] matrix

$w_{1,1}$	$w_{2,1}$
$w_{1,2}$	$w_{2,2}$
$w_{1,3}$	$w_{2,3}$
$w_{1,4}$	$w_{2,4}$
$w_{1,5}$	$w_{2,5}$
$w_{1,6}$	$w_{2,6}$
$w_{1,7}$	$w_{2,7}$
$w_{1,8}$	$w_{2,8}$
$w_{1,9}$	$w_{2,9}$

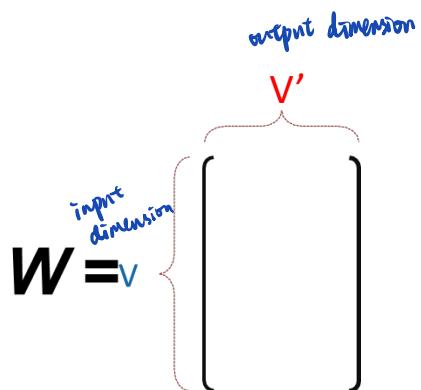
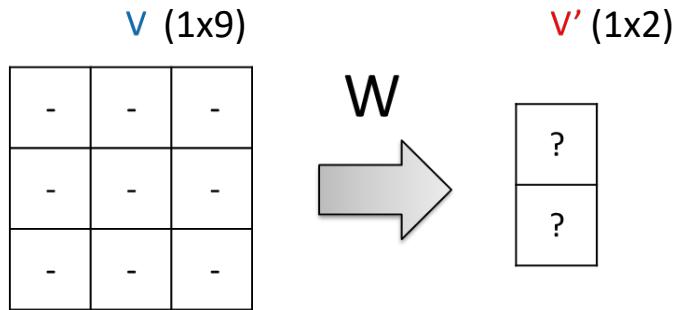
X

[1x2] matrix

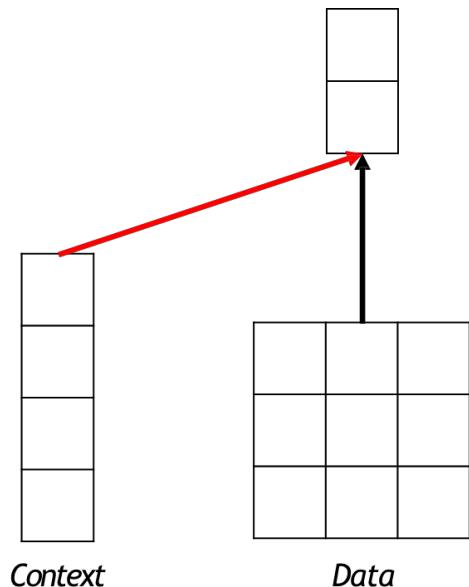
$$= \left[\sum_i^9 x_i * w_{1,i} , \sum_i^9 x_i * w_{2,i} \right]$$

Fully Connected Network

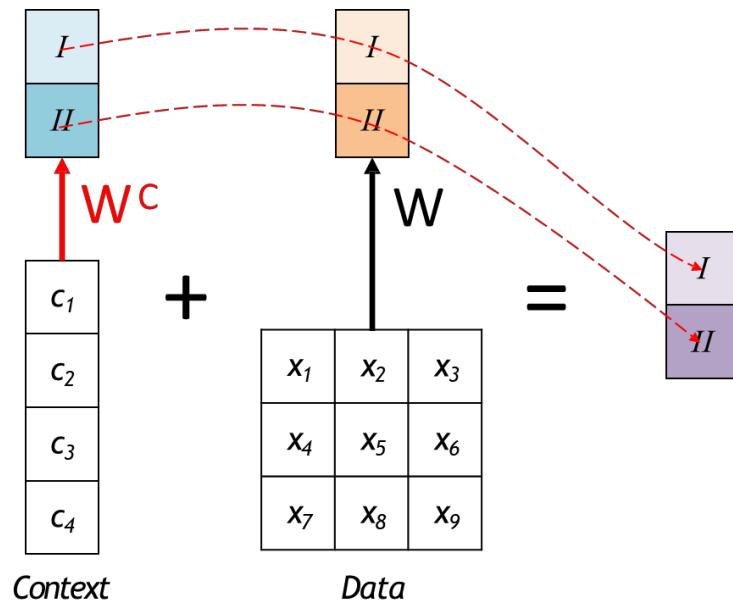
V to V' – Projection Notation



V to V' – Projection with Context (1)



V to V' – Projection with Context (2)



4

Data Transformation for Deep Learning NLP

V to V' with Context - Linear Algebra

[1 x 9] matrix

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
-------	-------	-------	-------	-------	-------	-------	-------	-------

[9x2] matrix

$w_{1,1}$	$w_{2,1}$
$w_{1,2}$	$w_{2,2}$
$w_{1,3}$	$w_{2,3}$
$w_{1,4}$	$w_{2,4}$
$w_{1,5}$	$w_{2,5}$
$w_{1,6}$	$w_{2,6}$
$w_{1,7}$	$w_{2,7}$
$w_{1,8}$	$w_{2,8}$
$w_{1,9}$	$w_{2,9}$

X

[1x2] matrix

$$= \begin{pmatrix} \sum_i^9 x_i * w_{1,i}, & \sum_i^9 x_i * w_{2,i} \end{pmatrix}$$

I

II

[1 x 4] matrix

c_1	c_2	c_3	c_4
-------	-------	-------	-------

X

[1x2] matrix

$w_{1,1}^c$	$w_{2,1}^c$
$w_{1,2}^c$	$w_{2,2}^c$
$w_{1,3}^c$	$w_{2,3}^c$
$w_{1,4}^c$	$w_{2,4}^c$

$$= \begin{pmatrix} \sum_i^4 c_i * w_{1,i}^c, & \sum_i^4 c_i * w_{2,i}^c \end{pmatrix}$$

I

II

V to V' with Context - Linear Algebra (Simplified)

[1 x (9+4)] matrix

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	c_1	c_2	c_3	c_4
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

X

[9+4] x 2 matrix

$w_{1,1}$	$w_{2,1}$
$w_{1,2}$	$w_{2,2}$
$w_{1,3}$	$w_{2,3}$
$w_{1,4}$	$w_{2,4}$
$w_{1,5}$	$w_{2,5}$
$w_{1,6}$	$w_{2,6}$
$w_{1,7}$	$w_{2,7}$
$w_{1,8}$	$w_{2,8}$
$w_{1,9}$	$w_{2,9}$
$w_{1,C,1}$	$w_{2,C,1}$
$w_{1,C,2}$	$w_{2,C,2}$
$w_{1,C,3}$	$w_{2,C,3}$
$w_{1,C,4}$	$w_{2,C,4}$

=
$$\begin{pmatrix} \sum_i^9 x_i * w_{1,i} & \sum_i^9 x_i * w_{2,i} \\ + \sum_i^4 c_i * w_{1,C,i} & + \sum_i^4 c_i * w_{2,C,i} \end{pmatrix}$$

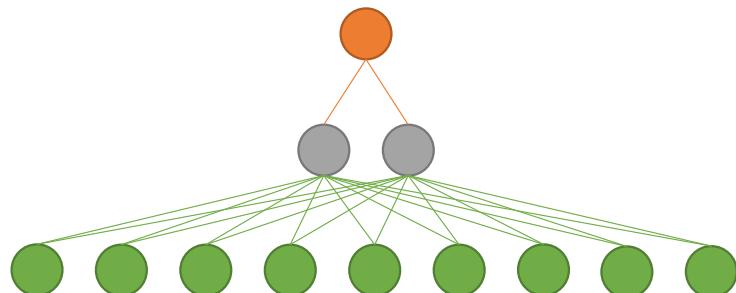
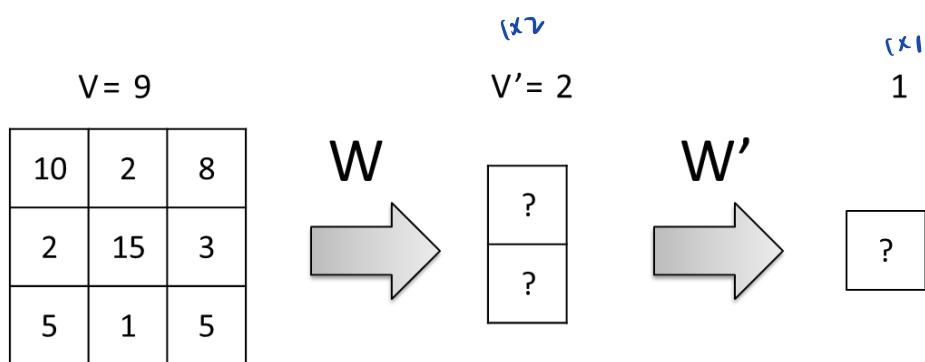
I II

4

Data Transformation for Deep Learning NLP

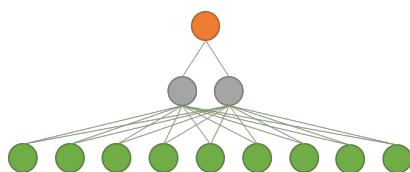
 $V \rightarrow V' \rightarrow 1$

Can be \geq steps.

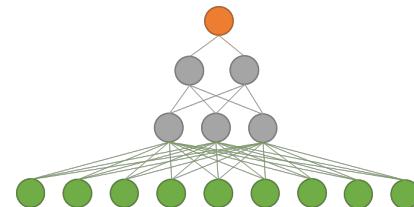


4

Data Transformation for Deep Learning NLP

 $V \rightarrow V' \rightarrow 1$ 

Single Layer

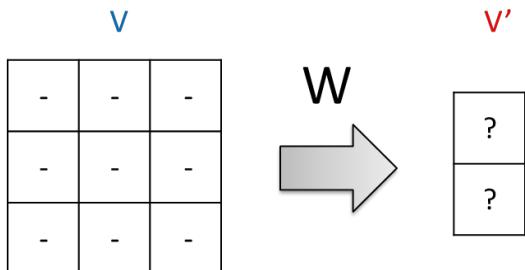
 $V \rightarrow V' \rightarrow 1$ 

Multilayer

 $V \rightarrow V' \rightarrow V'' \rightarrow 1$

Seq2Seq Encoding

*Single Item
Summarisation*



*Multiple Item
Summarisation*

?

Multiple Item Summarisation

10	2	8
2	15	3
5	1	5

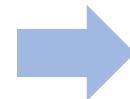
Data 1

13	4	8
4	5	2
1	45	31

Data 2

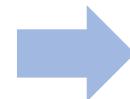
6	3	4
1	7	1
3	4	0

Data 3



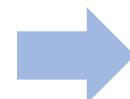
?	?	?
?	?	?
?	?	?

V



?
?

V'



?

1

Data Transformation for Deep Learning NLP

Vs to V'

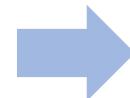
10	2	8
2	15	3
5	1	5

$$(10+13+6)/3$$

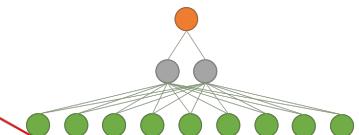
13	4	8
4	5	2
1	45	31

$$(8+8+4)/3$$

6	3	4
1	7	1
3	4	0



9.6	?	6.6
?	?	?
?	?	?



Element-wise Average

4

Data Transformation for Deep Learning NLP

Vs to V'

10	2	8
2	15	3
5	1	5

13	4	8
4	5	2
1	45	31

6	3	4
1	7	1
3	4	0

$$w^1 \stackrel{x}{=} 0.2$$



2	0.4	1.6
0.4	3	0.6
1	0.2	1.0

$$w^2 \stackrel{x}{=} 0.4$$



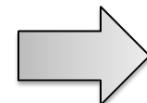
5.2	1.6	3.2
1.6	2	0.8
0.4	18	12.4

$$w^3 \stackrel{x}{=} 0.4$$



2.4	1.2	1.6
0.4	2.8	0.4
1.2	1.6	0

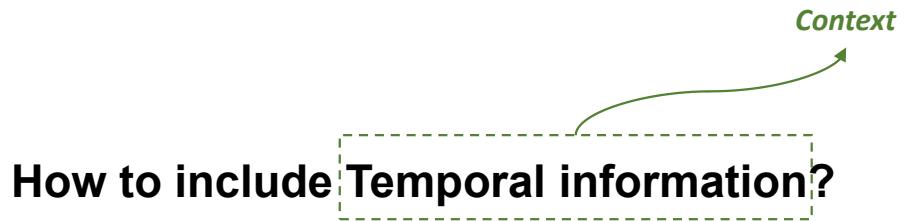
Element-wise multiplication



Element-wise summation

9.6	3.2	6.4
2.4	7.8	1.8
2.6	19.8	13.4

Temporal Summarisation



How to include Temporal information?

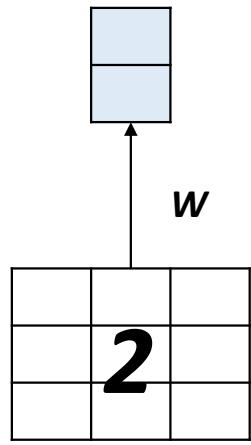
Context

A diagram illustrating the concept of temporal summarisation. A dashed green rectangular box encloses the question "How to include Temporal information?". A curved green arrow originates from the bottom right corner of this box and points towards the word "Context" located above and to the right of the box.

4

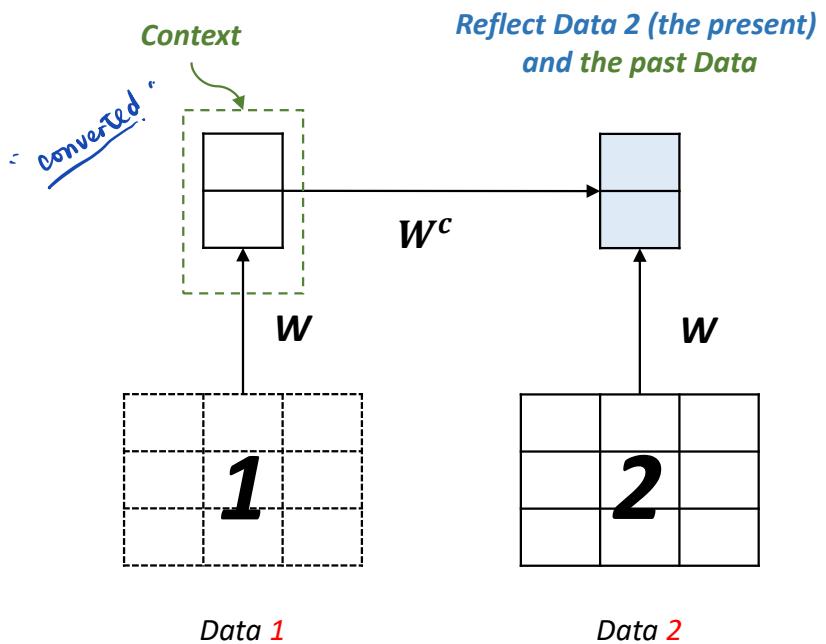
Data Transformation for Deep Learning NLP

$V_s \rightarrow V'_s \rightarrow V'$



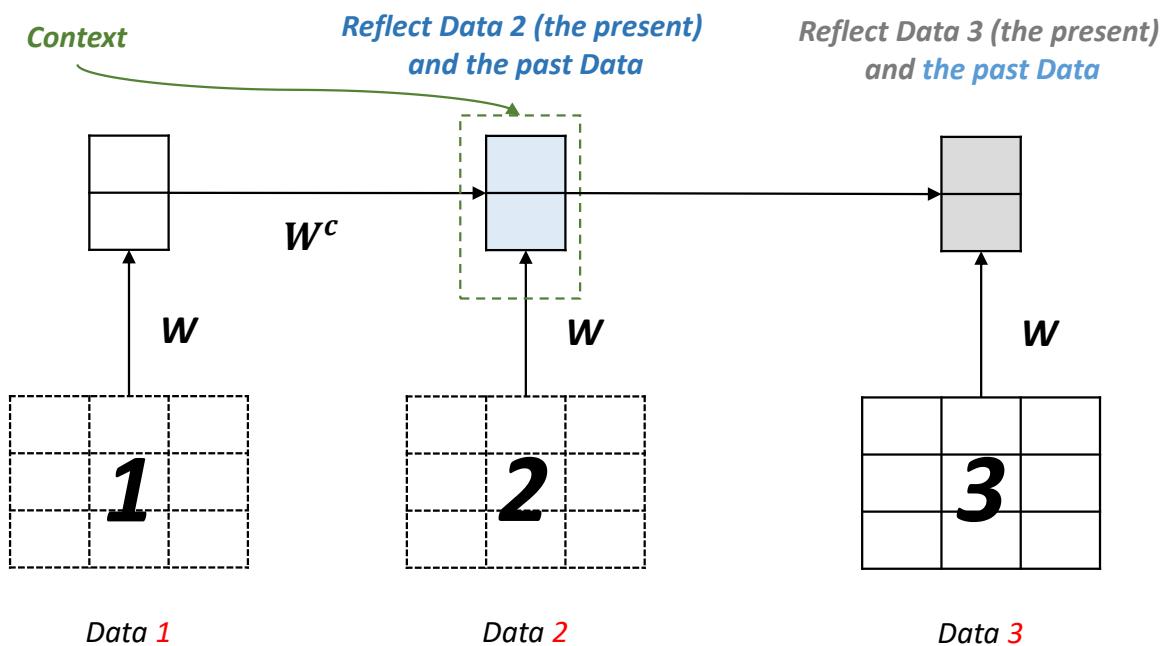
Data 2

$V_s \rightarrow V's \rightarrow V'$



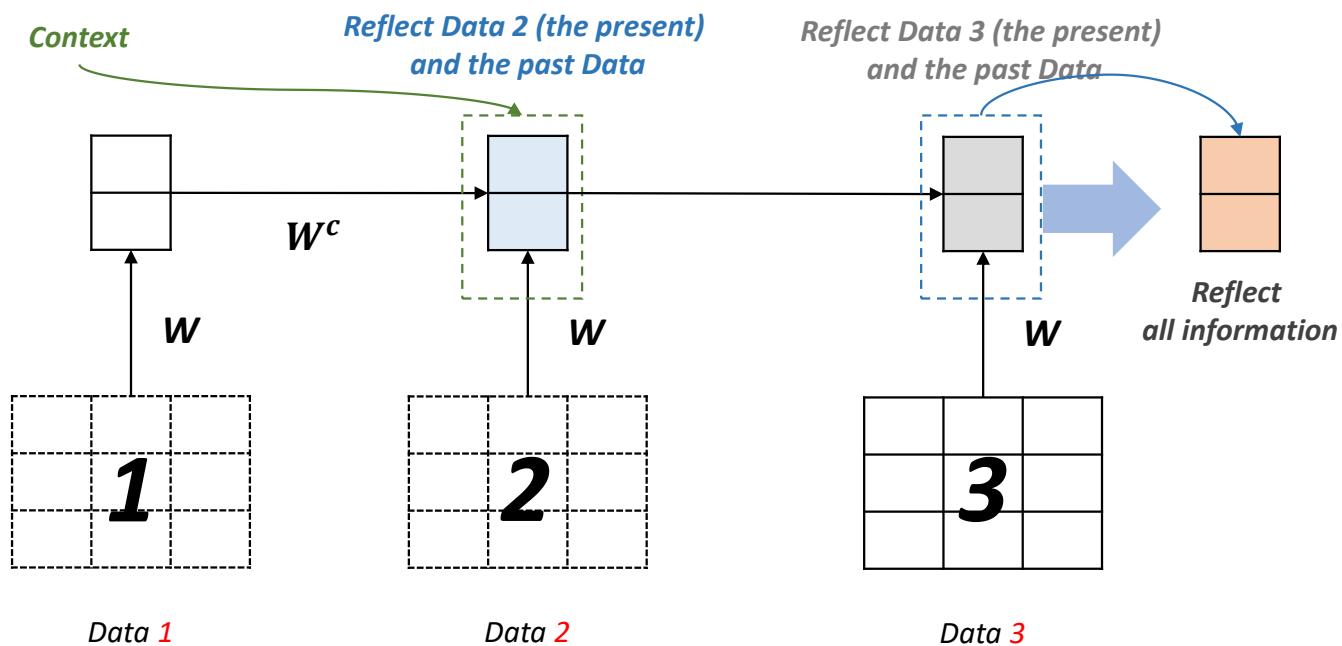
Data Transformation for Deep Learning NLP

Vs → V's → V'

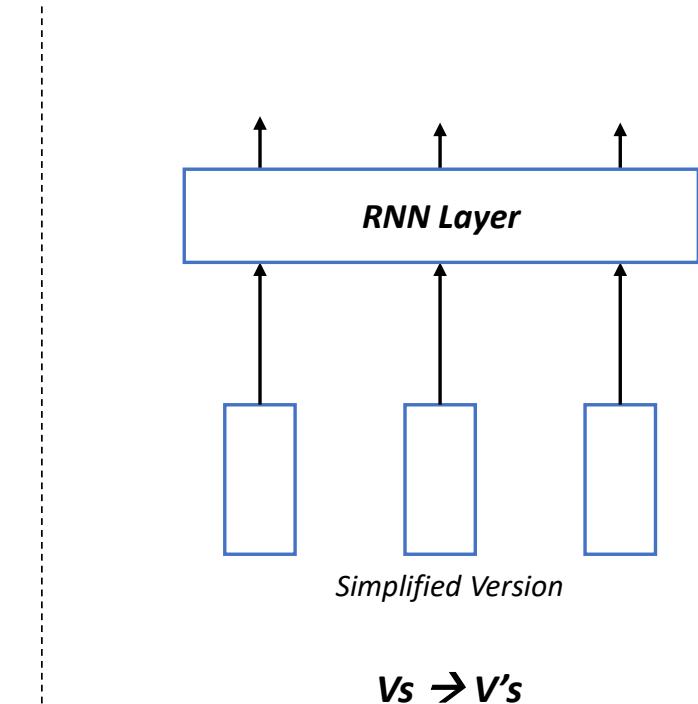
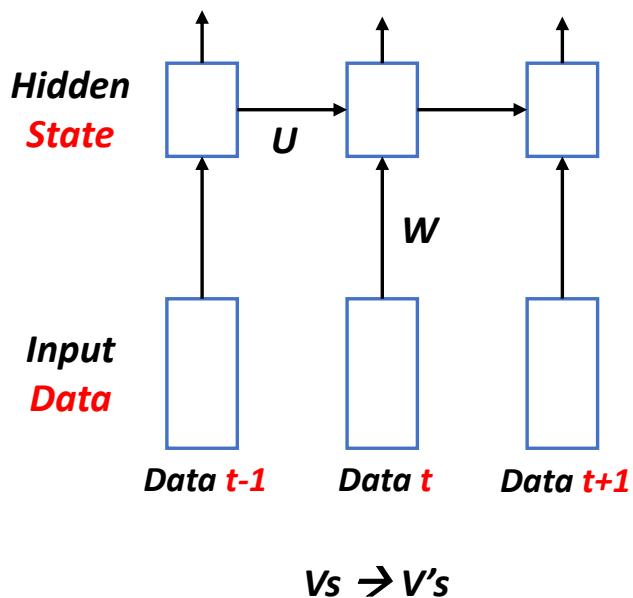


Data Transformation for Deep Learning NLP

V_s → V's → V'

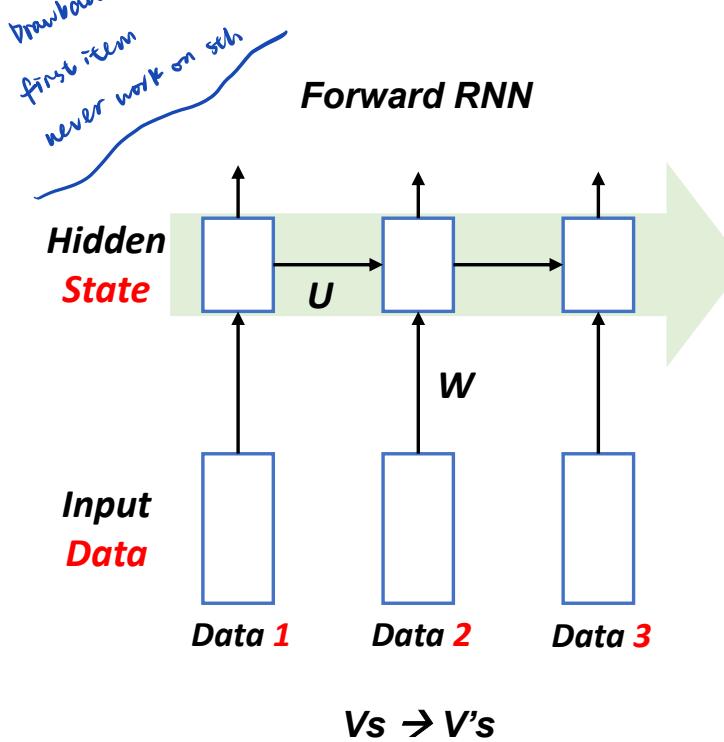


Graphical Notation

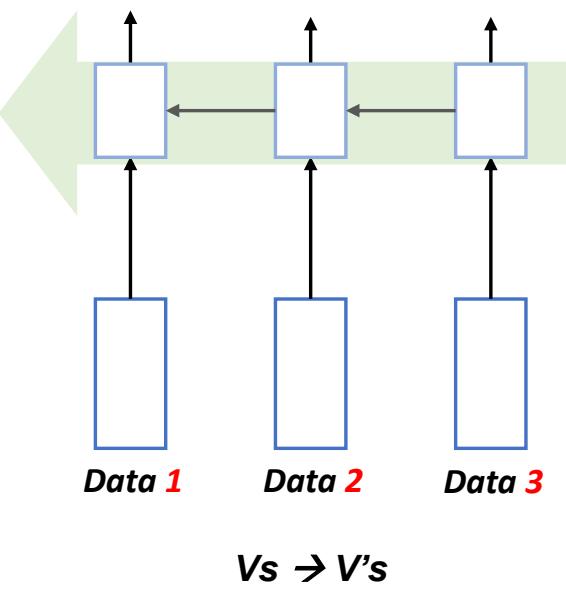


Data Transformation for Deep Learning NLP

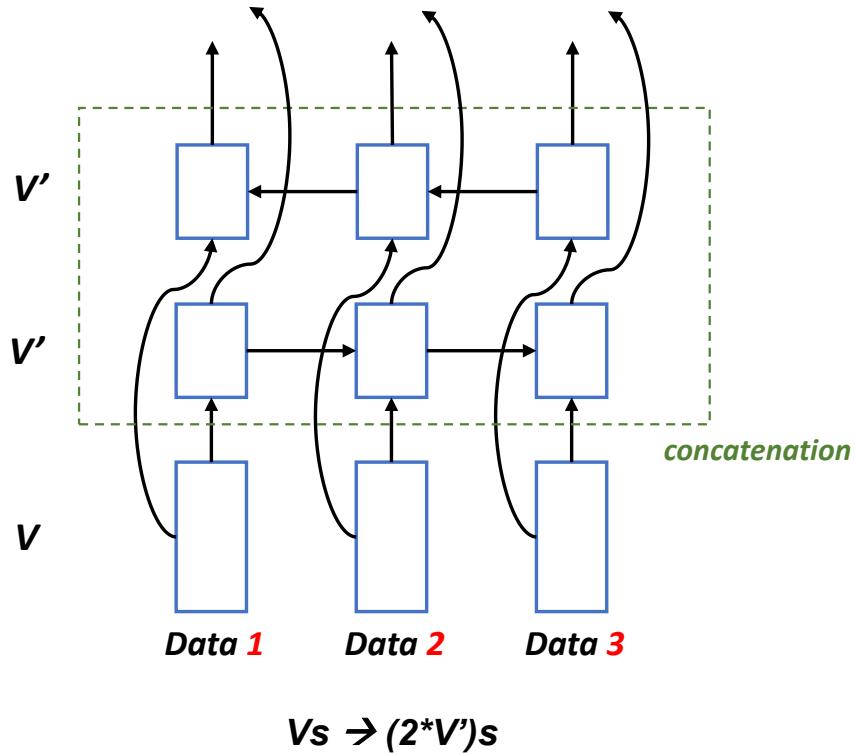
Forward/Backward RNN



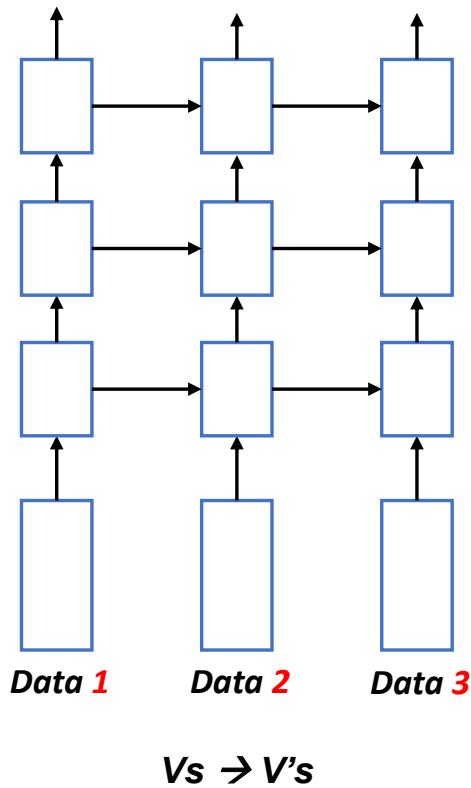
Backward RNN



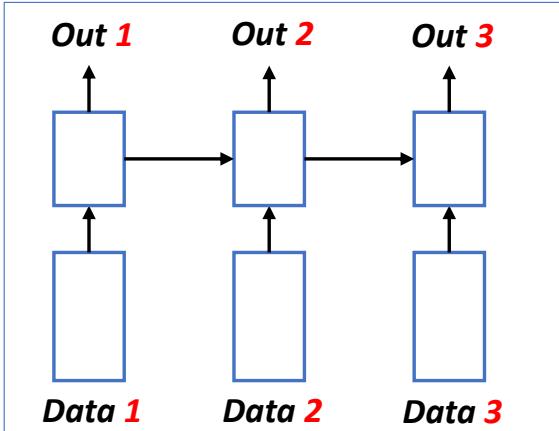
Bidirectional RNN



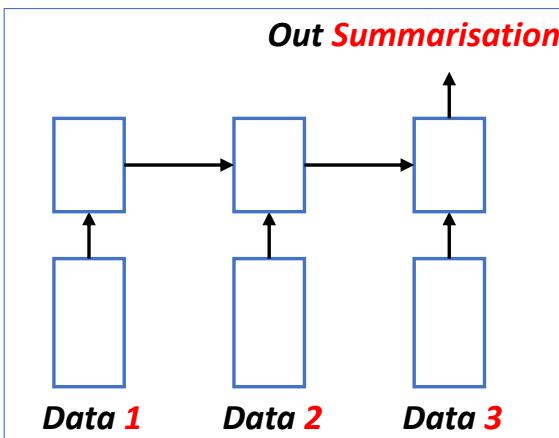
Stacking RNN



RNN: Input and Output

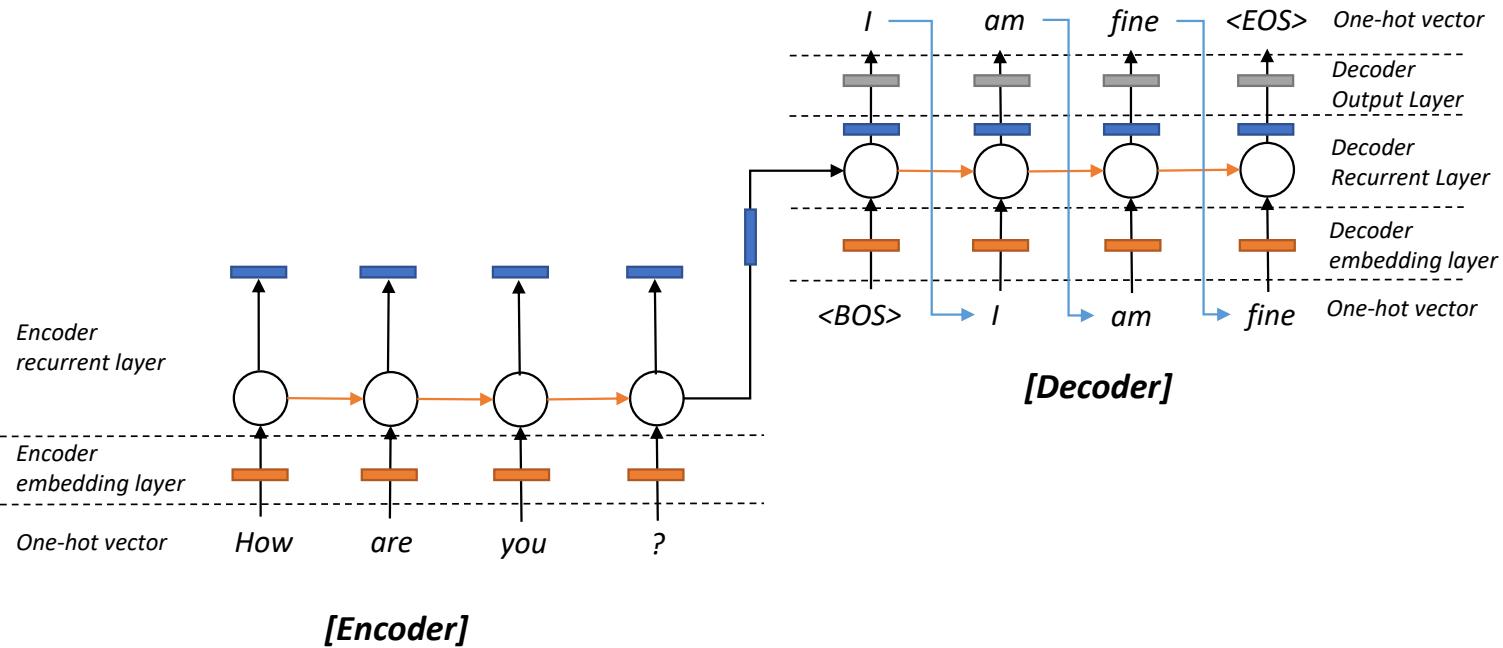


- ✓ $Vs \rightarrow V's$
- ✓ $\text{Len}(Vs) \rightarrow \text{Len}(V's)$



- ✓ $Vs \rightarrow 1$

Seq2Seq Encoding and Decoding- Dialog System



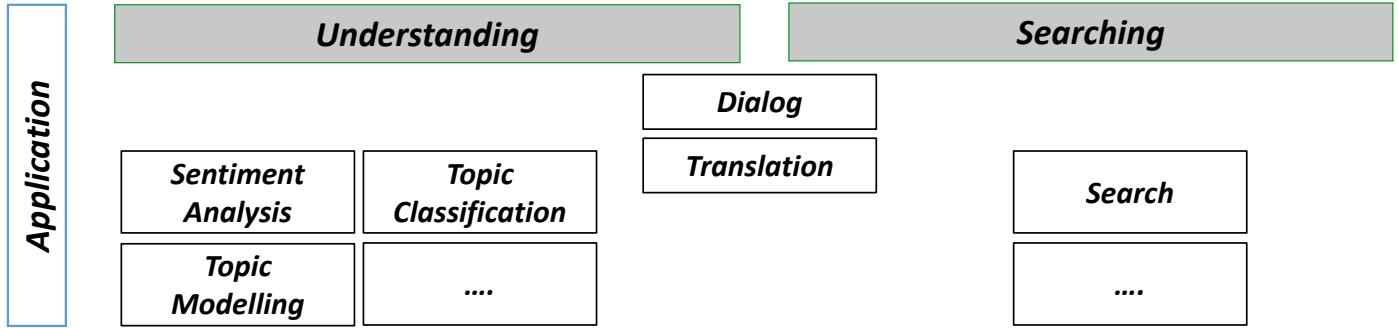
0 LECTURE PLAN

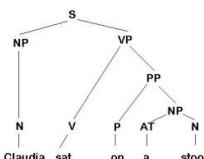
Lecture 4: Word Classification and Machine Learning 2

1. Machine Learning and NLP: Finish
2. Seq2Seq Learning
3. Seq2Seq Deep Learning
 1. RNN (Recurrent Neural Network)
 2. LSTM (Long Short-Term Memory)
 3. GRU (Gated Recurrent Unit)
4. Data Transformation for Deep Learning NLP
5. **Next Week Preview**
 - Natural Language Processing Stack

5 Next Week Preview

The purpose of Natural Language Processing: Overview



NLP Stack	Entity Extraction	When Sebastian Thrun ...	When Sebastian Thrun PERSON started at Google ORG in 2007 DATE
	Parsing	Claudia sat on a stool	 <pre> graph TD S --- NP1[NP] S --- VP S --- PP NP1 --- N1[N] NP1 --- V VP --- P VP --- AT VP --- NP2[NP] NP2 --- N2[N] PP --- on[on] PP --- a[a] PP --- stool[stool] N1 --- Claudia[Claudia] V --- sat[sat] P --- a AT --- a N2 --- stool </pre>
	PoS Tagging	She sells seashells	[she/PRP] [sells/VBZ] [seashells/NNS]
	Stemming	Drinking, Drank, Drunk	Drink
	Tokenisation	How is the weather today	[How] [is] [the] [weather] [today]

/ Reference

Reference for this lecture

- Deng, L., & Liu, Y. (Eds.). (2018). Deep Learning in Natural Language Processing. Springer.
- Rao, D., & McMahan, B. (2019). Natural Language Processing with PyTorch: Build Intelligent Language Applications Using Deep Learning. " O'Reilly Media, Inc.".
- Manning, C. D., Manning, C. D., & Schütze, H. (1999). Foundations of statistical natural language processing. MIT press.
- Blunsom, P 2017, Deep Natural Language Processing, lecture notes, Oxford University
- Manning, C 2017, Natural Language Processing with Deep Learning, lecture notes, Stanford University
- Sordoni, A., Bengio, Y., Vahabi, H., Lioma, C., Grue Simonsen, J., & Nie, J. Y. (2015, October). A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In Proceedings of the 24th ACM International Conference on Information and Knowledge Management (pp. 553-562). ACM.

Figure Reference

- <https://towardsdatascience.com/types-of-optimization-algorithms-used-in-neural-networks-and-ways-to-optimize-gradient-95ae5d39529f>
- <https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21>