Deep Learning (1470)

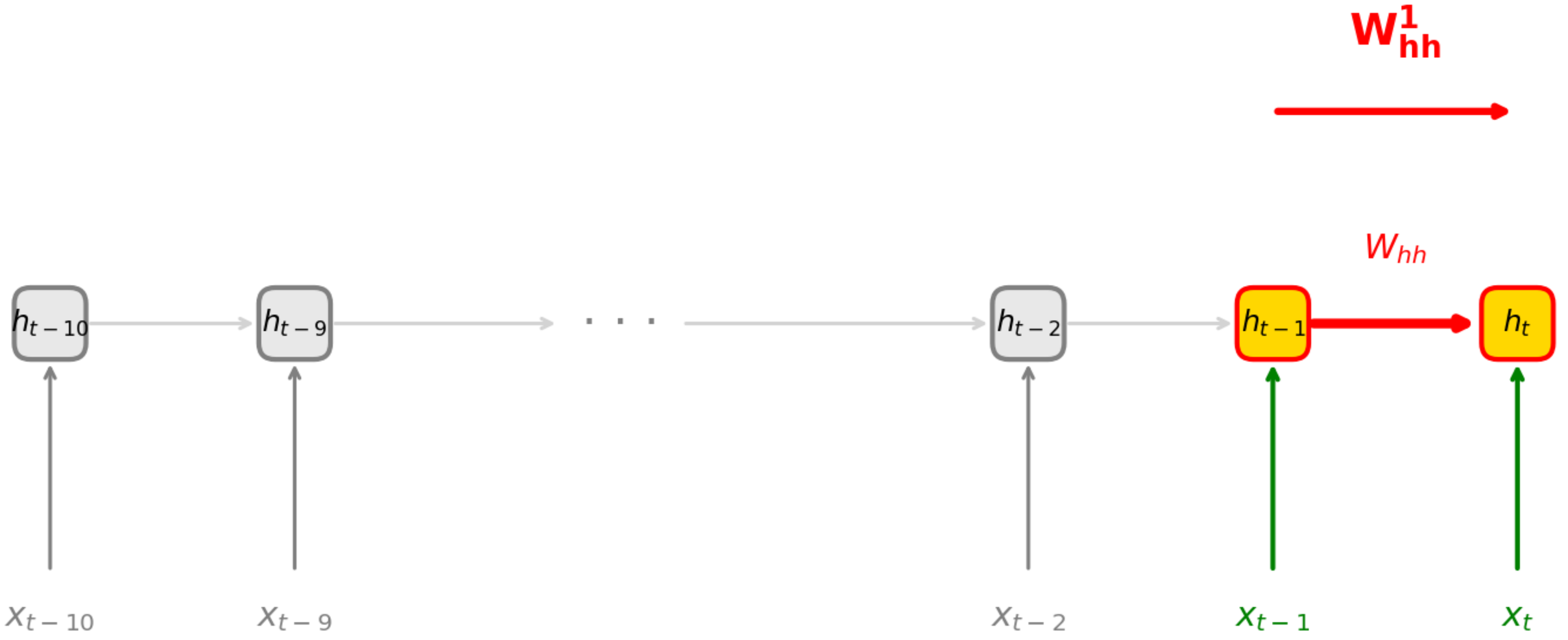
Randall Balestriero

Class 12: seq2seq

Recap!

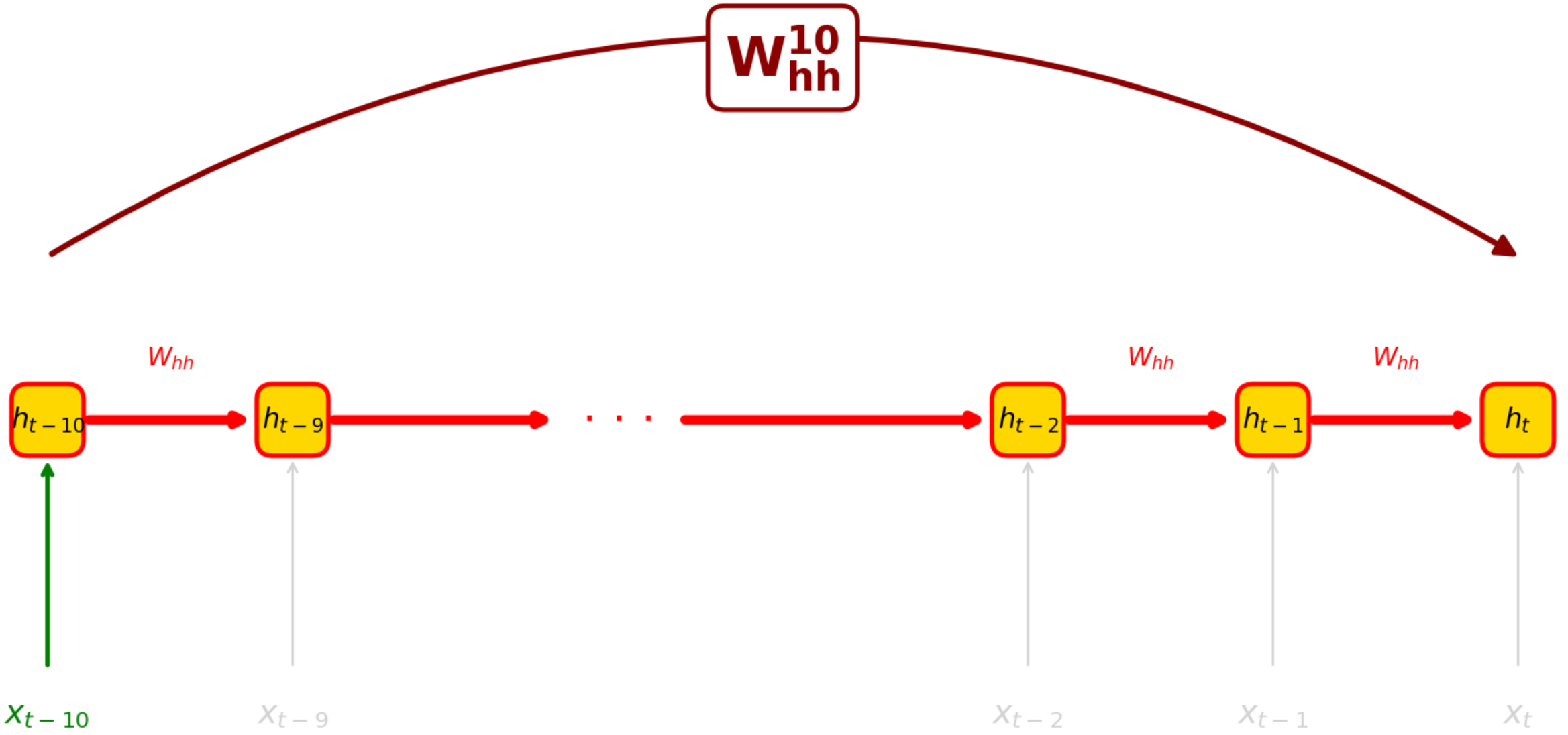
Information Flow from h_{t-1} to h_t

Contribution: $W_{hh}^1 \cdot h_{t-1}$



Information Flow from h_{t-10} to h_t

Contribution: $W_{hh}^{10} \cdot h_{t-10}$

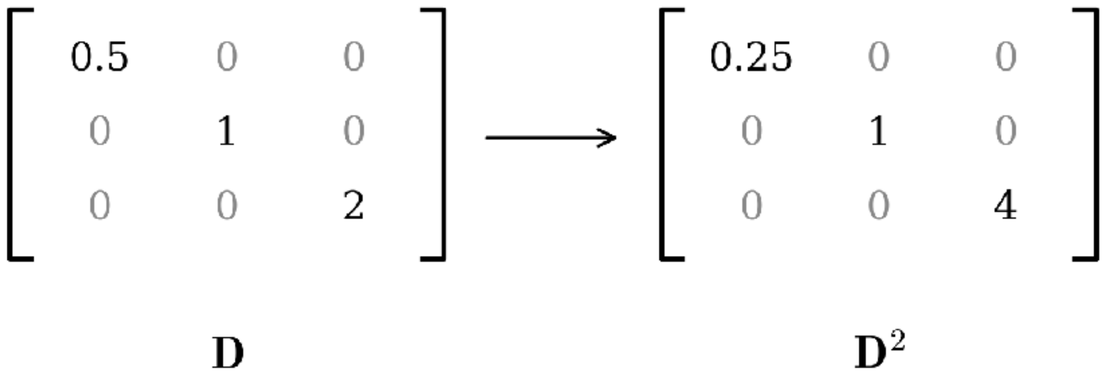


Exponential Memory Loss: Example

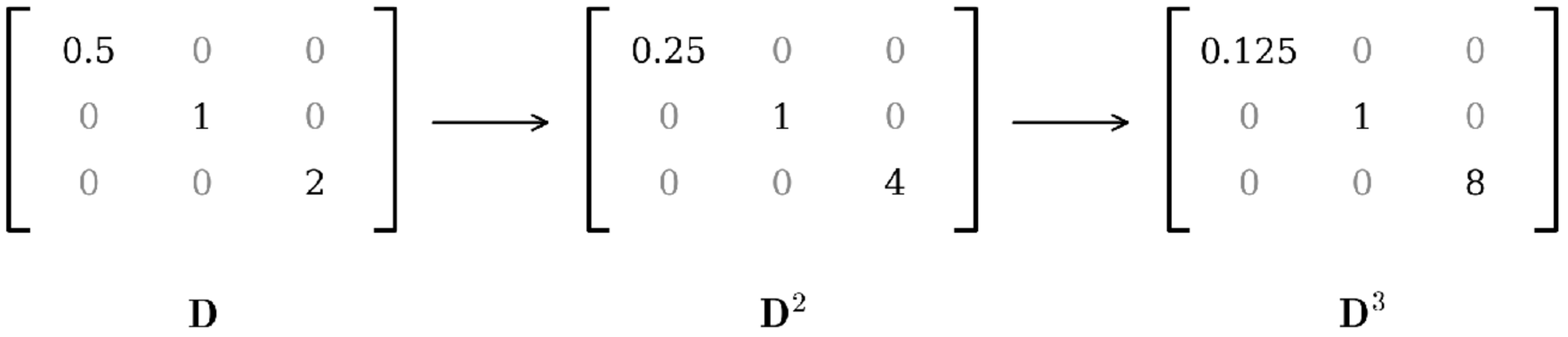
$$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

$\mathbf{D}$

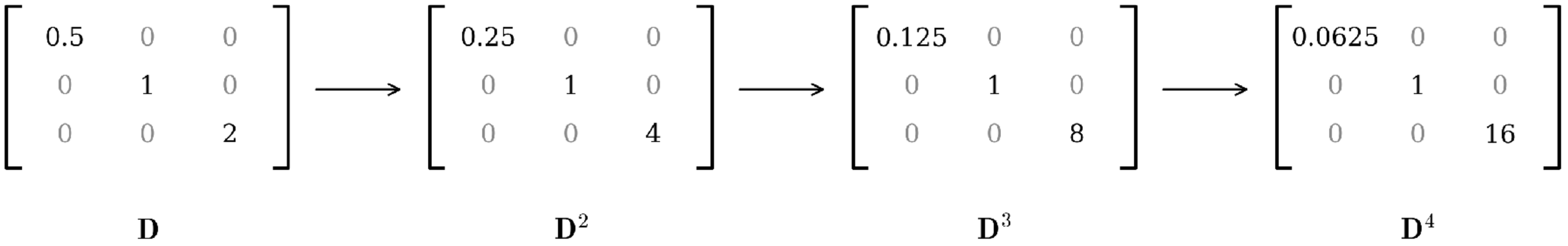
Exponential Memory Loss: Example



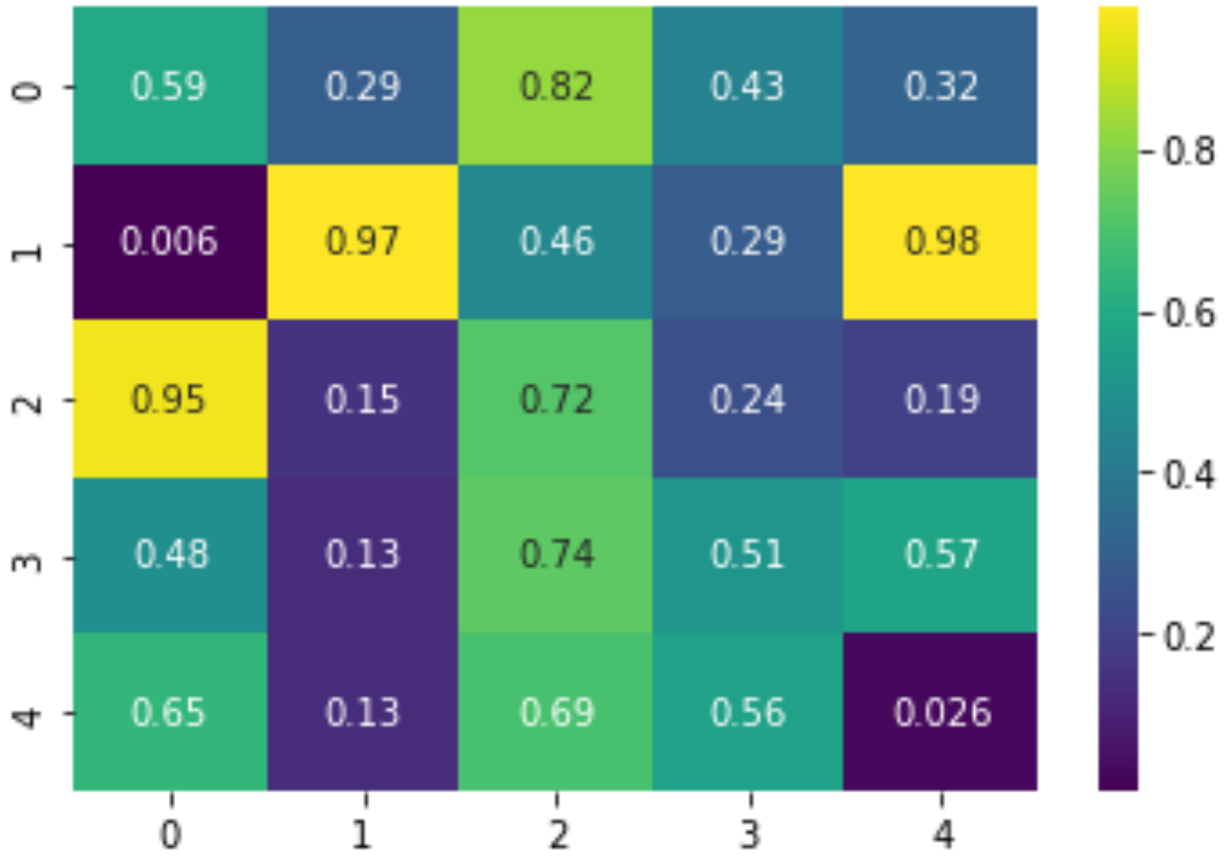
Exponential Memory Loss: Example



Exponential Memory Loss: Example

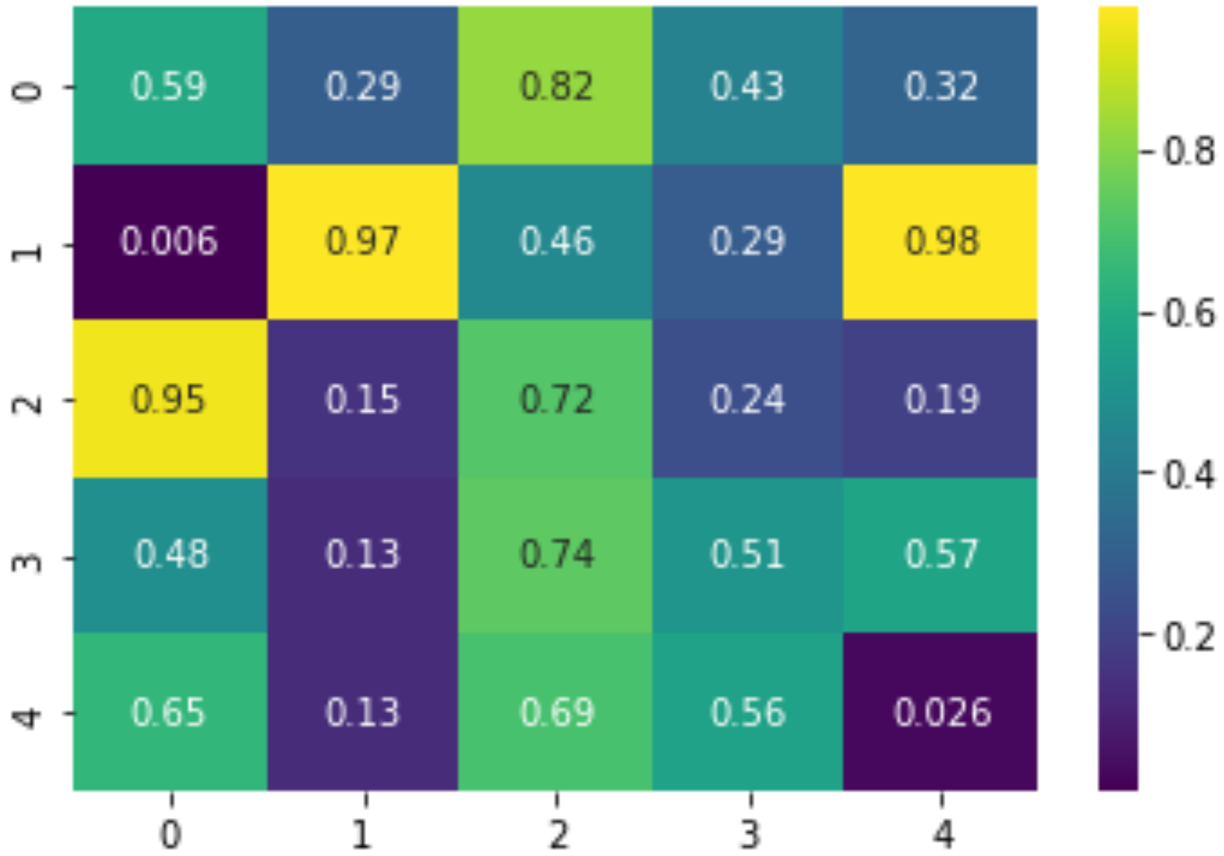


Exponential Memory Loss: Example



What about now?

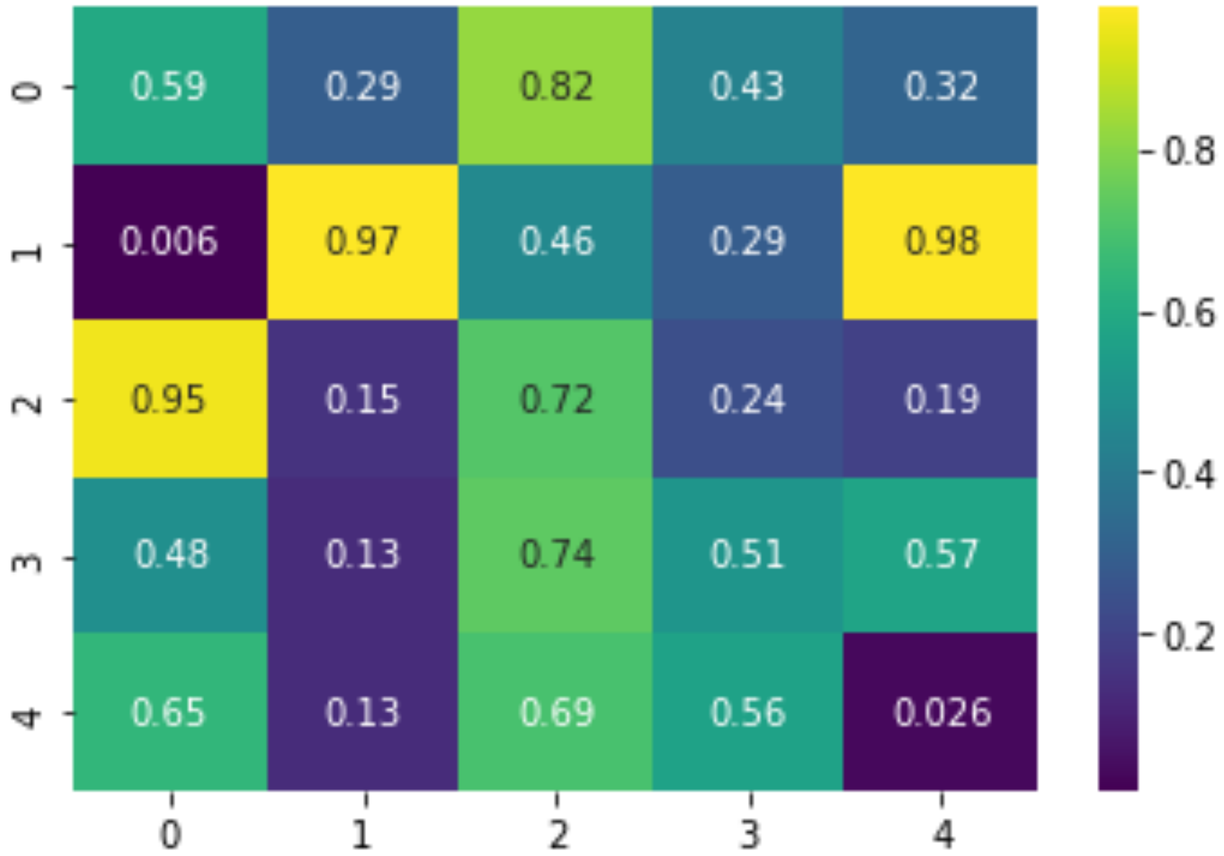
Exponential Memory Loss: Example



What about now?

$$W = U\Sigma V^{\top}$$

Exponential Memory Loss: Example

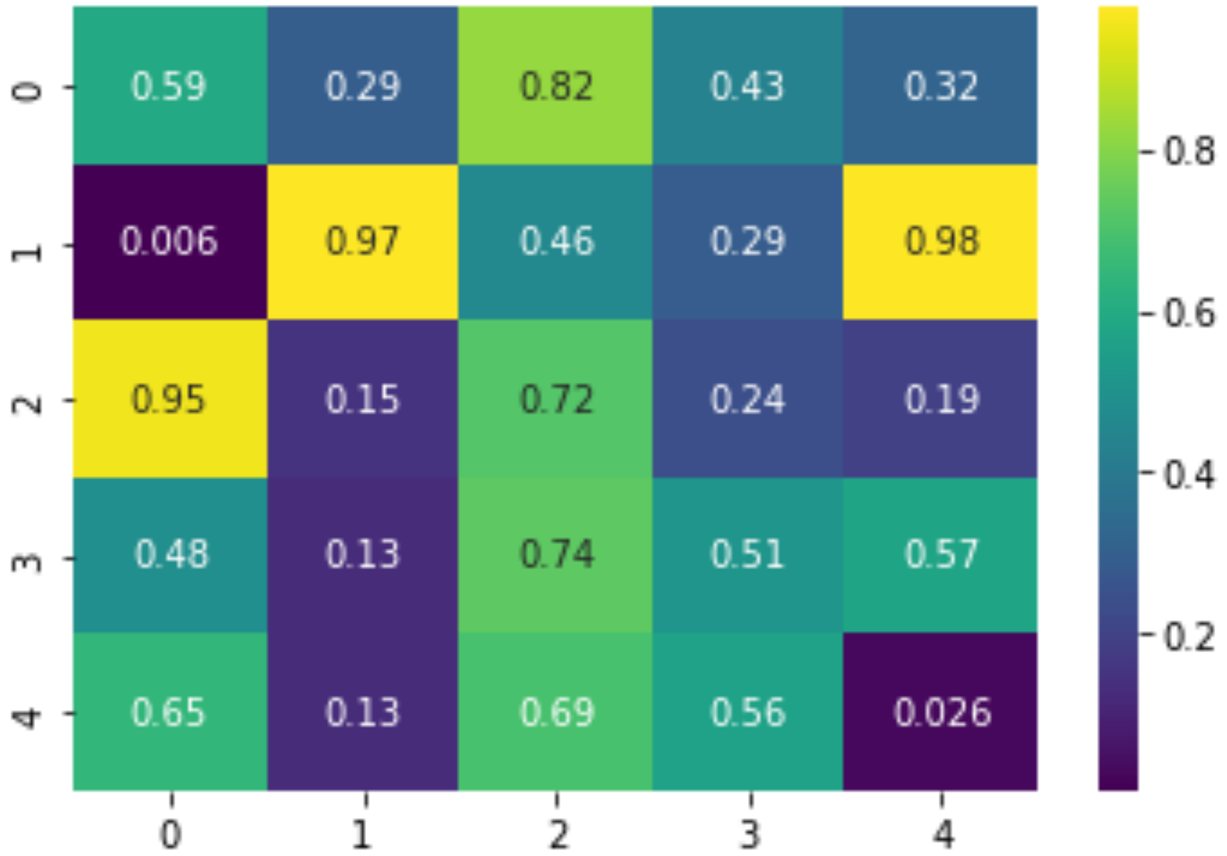


What about now?

$$W = U\Sigma V^\top$$

$$W^p = U\Sigma^p V^\top$$

Exponential Memory Loss: Example



What about now?

$$W = U\Sigma V^\top$$

$$W^p = U\Sigma^p V^\top$$

Can you come up with a condition to prevent explosion/collapse?

Beyond ORNNs/LSTMs/GRUs

- Instead of fixing the computation at time-step… we can?

Beyond ORNNs/LSTMs/GRUs

- Instead of fixing the computation at time-step… we can?

orthogonal like representation. We fix a pool size k, and then the update equations for this model are:

$$h_t = \sigma(Ux_t + b) + Vh_{t-1}$$
$$y_t = W_I h_t + W_P P_k(h_t) \tag{6}$$

where if h is the kd dimensional vector $h = [h_1, ..., h_{kd}]^T$, then $P(h)$ is the d dimensional vector defined by

$$P(h)_i = \sqrt{\sum_{j=k(i-1)+1}^{ki} h_j^2}$$

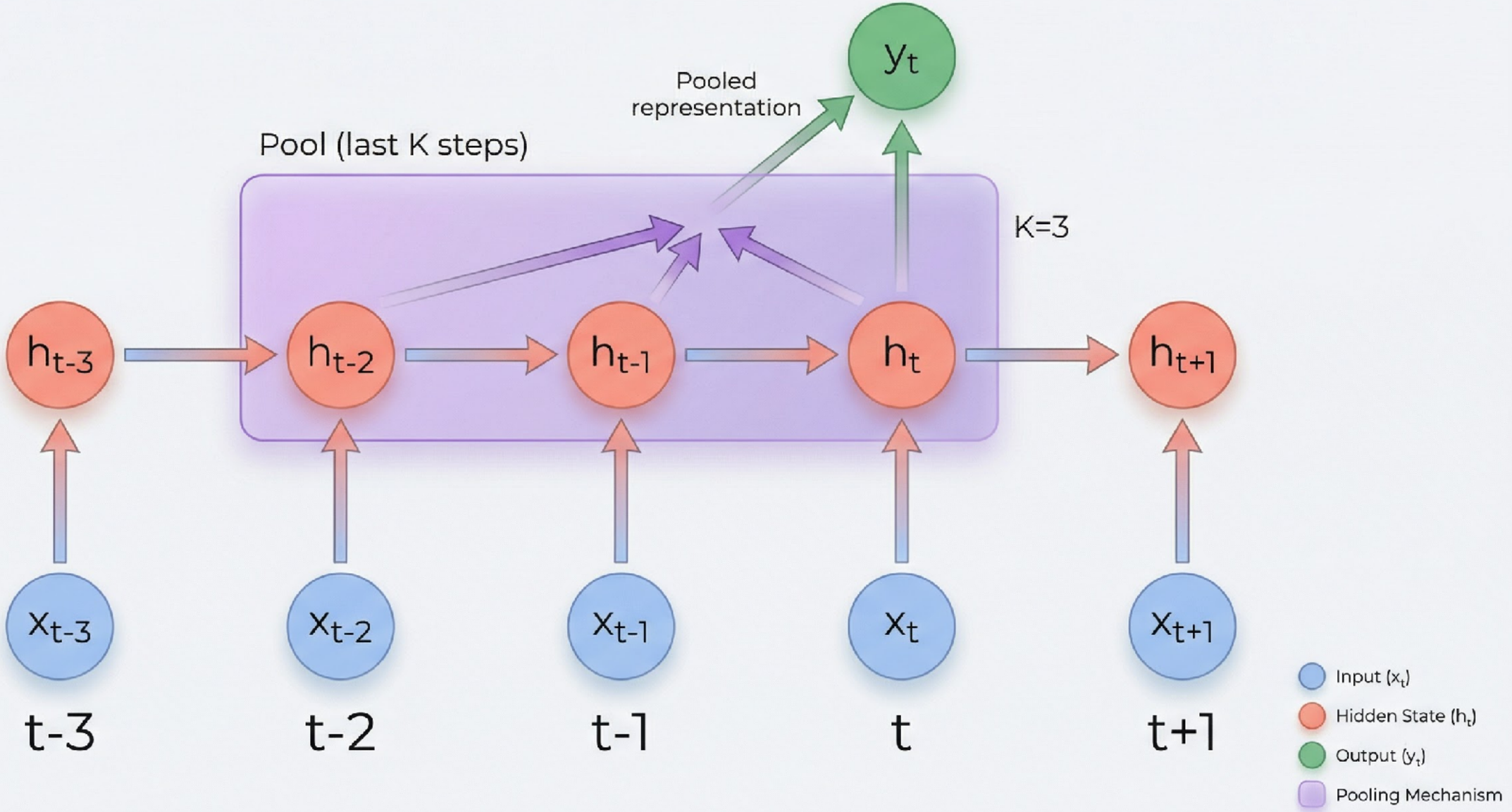
Recurrent Orthogonal Networks and Long-Memory Tasks

Mikael Henaff MBH305@NYU.EDU
New York University, Facebook AI Research

Arthur Szlam ASZLAM@FACEBOOK.COM
Facebook AI Research

Yann LeCun YANN@CS.NYU.EDU
New York University, Facebook AI Research

Recurrent Neural Network with Pooling Architecture



Recurrent Neural Network with Pooling Architecture

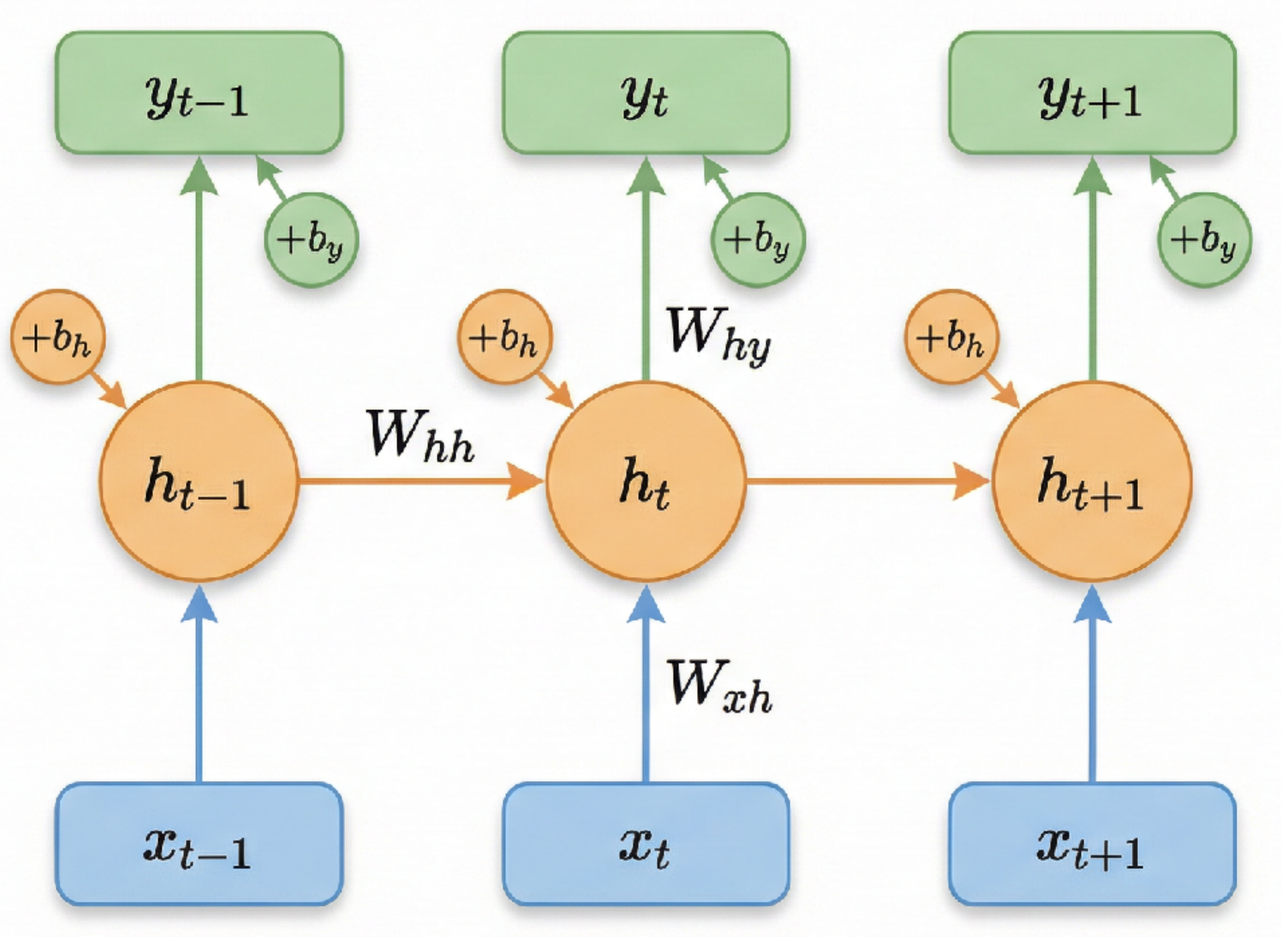
h_{t-3} h_{t+1}

x_{t-3} x_{t-2} x_{t-1} x_t x_{t+1}

t-3 t-2 t-1 t t+1

Input (x_t)
Hidden State (h_t)
Output (y_t)
Pooling Mechanism

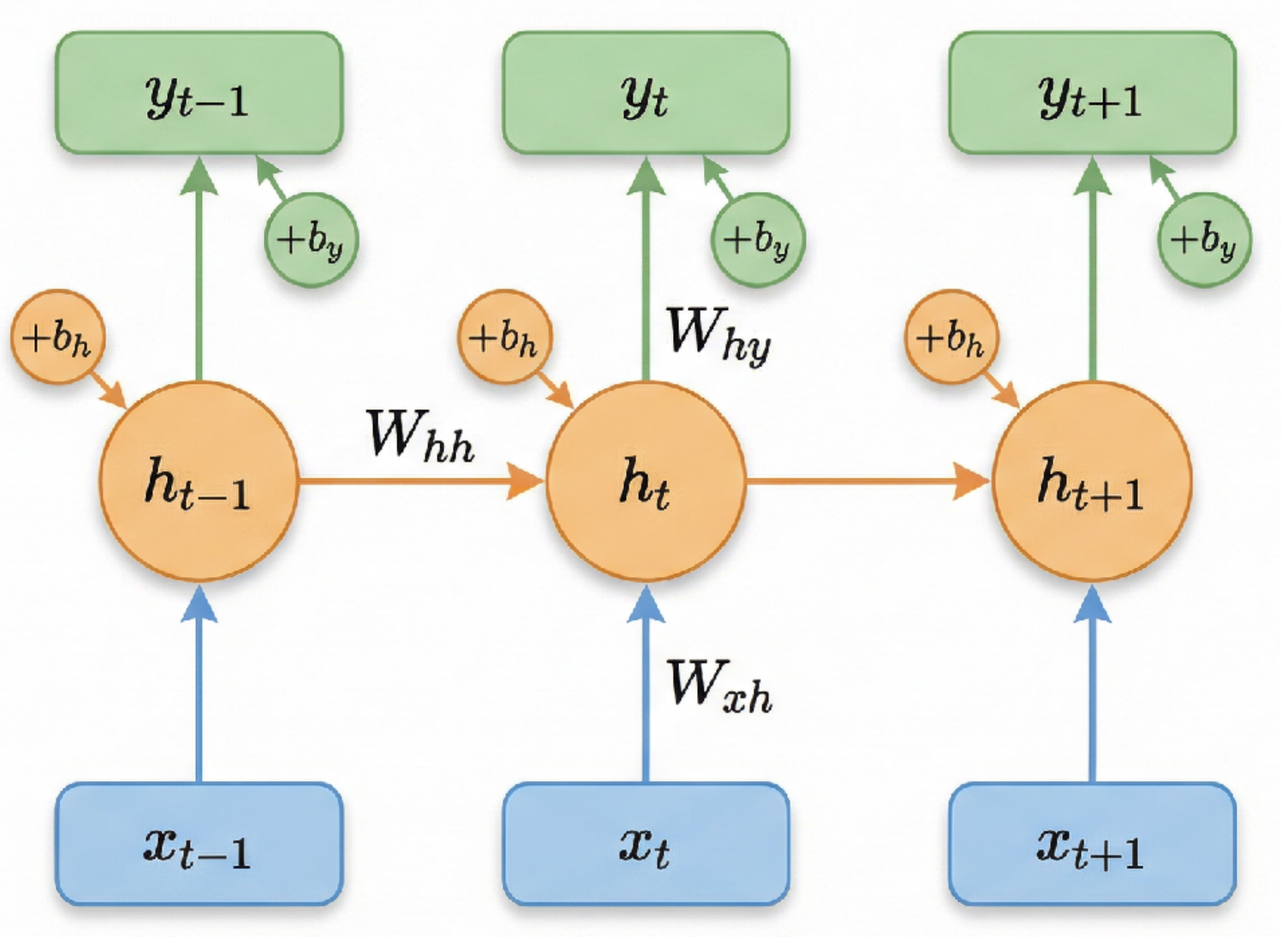
Back to Language Modeling



$$y_t = x_{t+1}$$

$$p(x_{t+1} \mid x_t, \ldots, x_1) \approx f_\theta(x_t, h_{t-1})$$

Back to Language Modeling

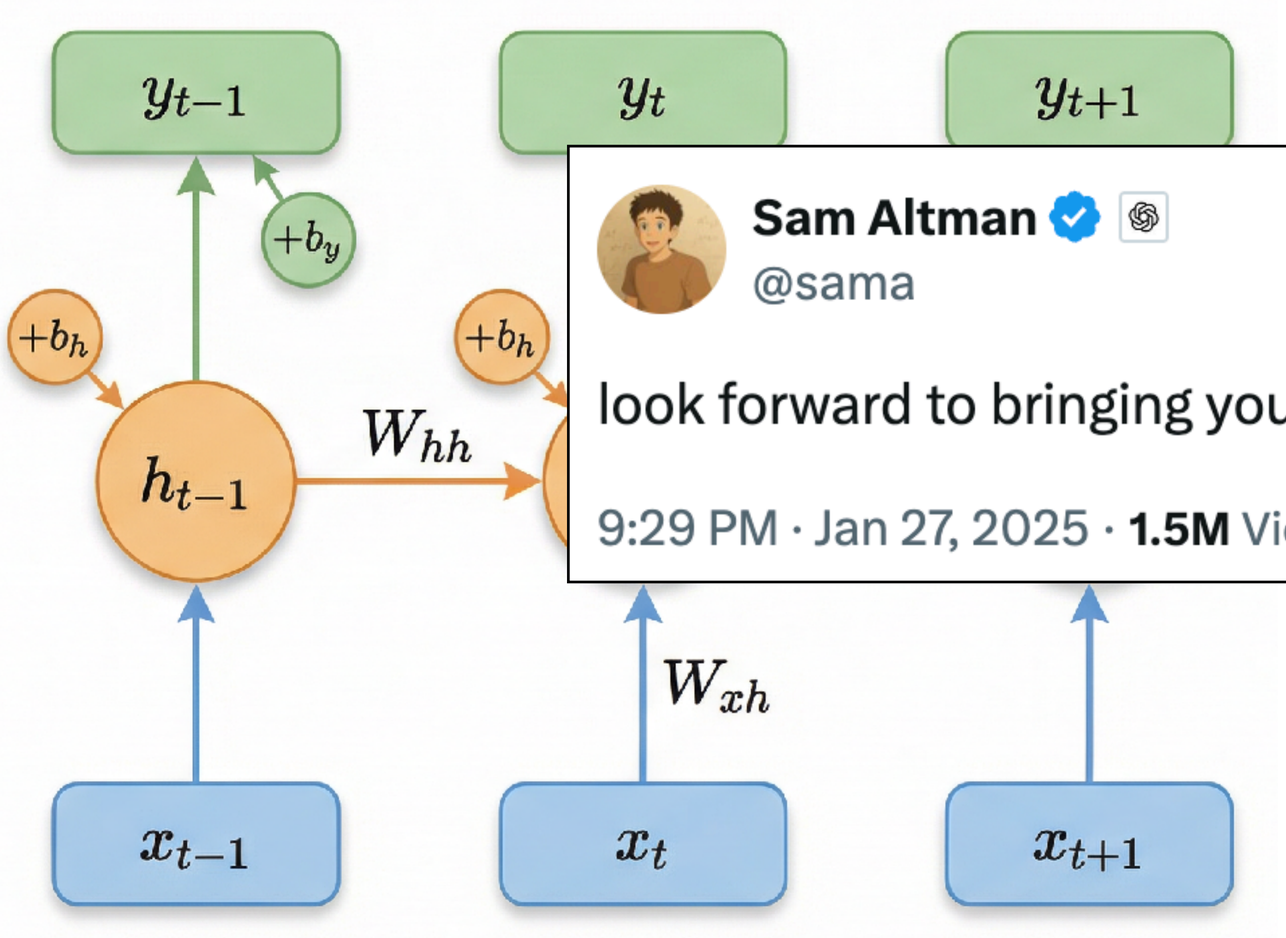


$$y_t = x_{t+1}$$

$$p(x_{t+1} \mid x_t, \ldots, x_1) \approx f_\theta(x_t, h_{t-1})$$

How do you “generate” language now?

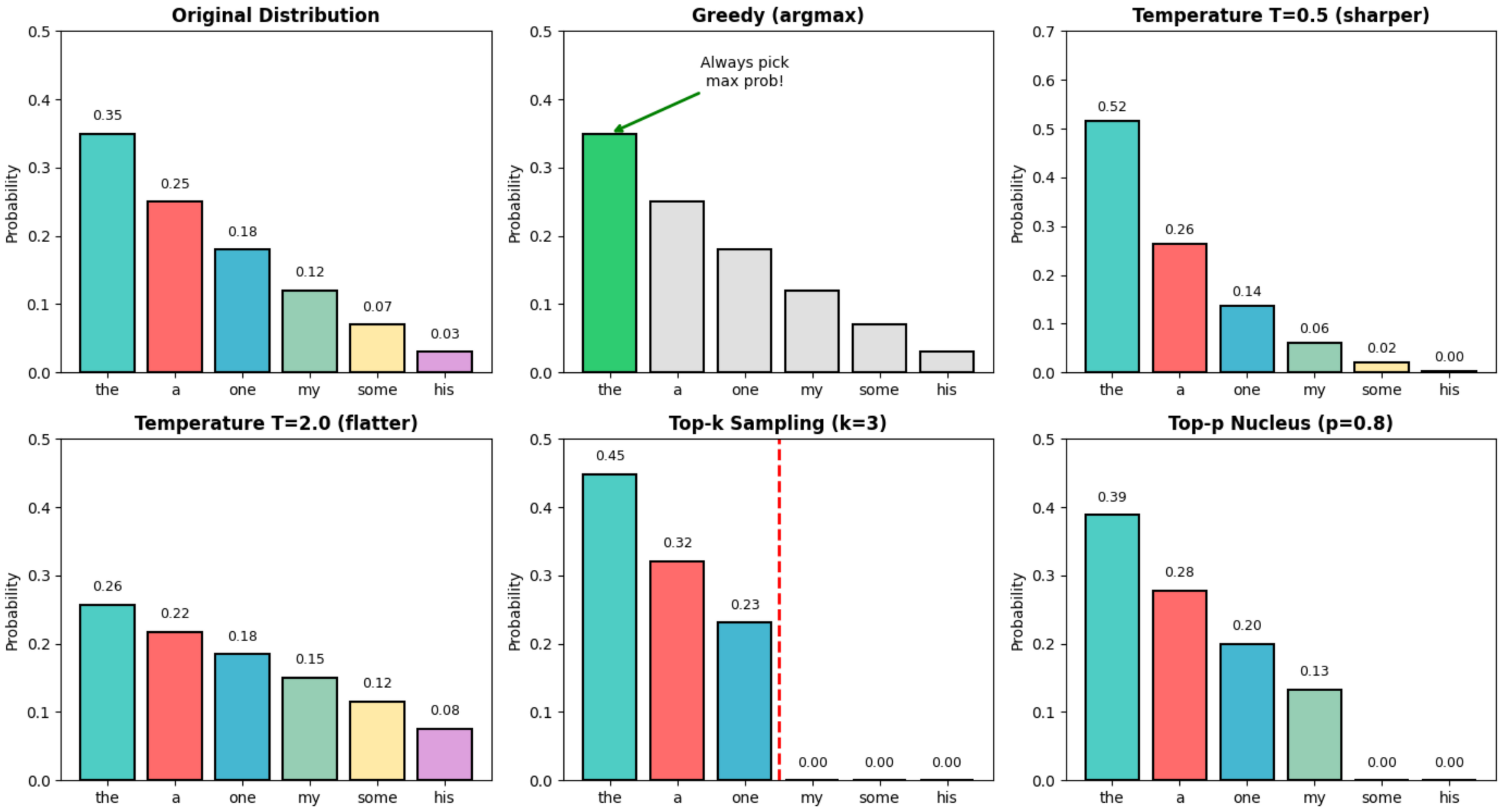
Back to Language Modeling

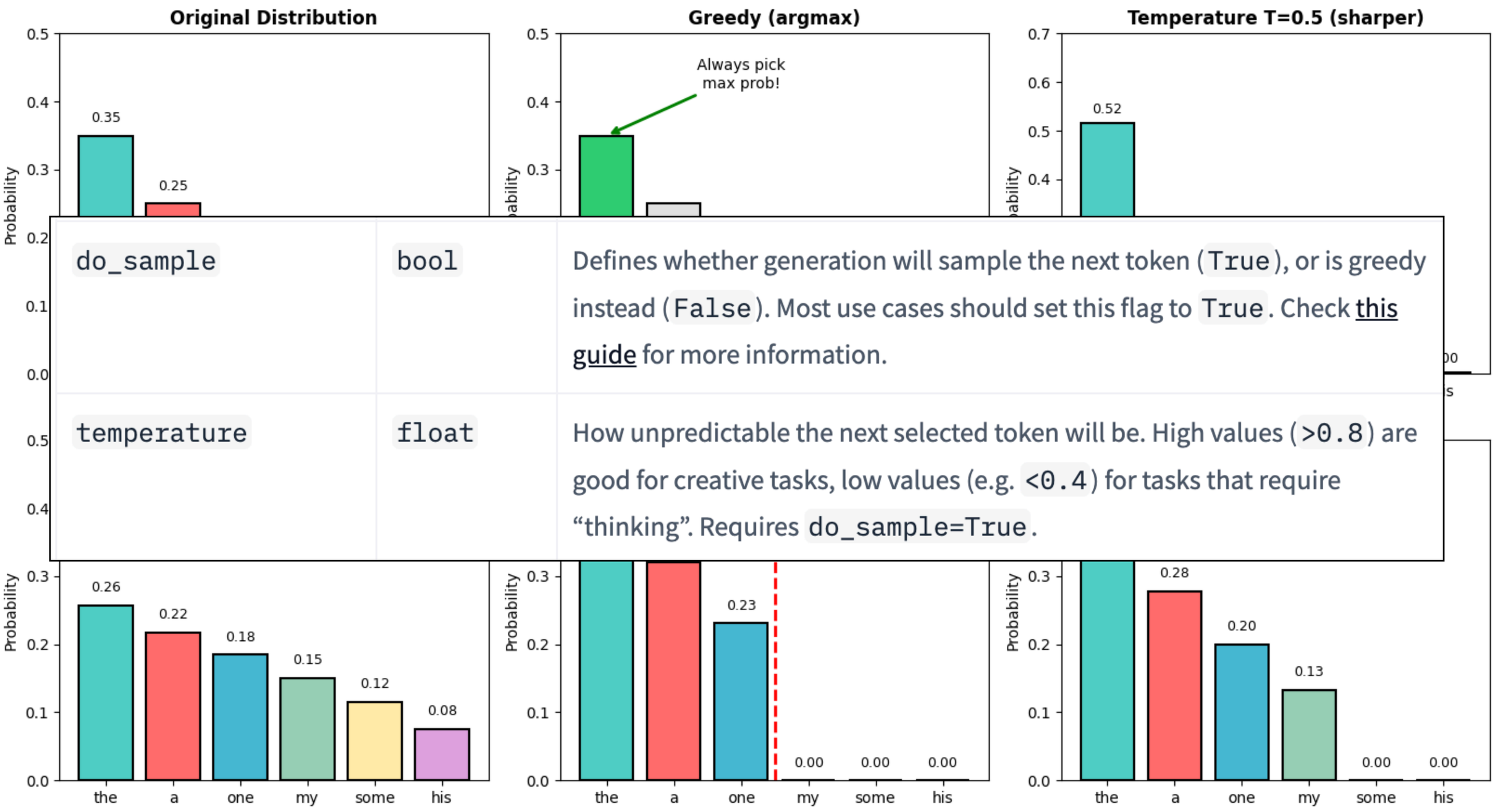


$= x_{t+1}$

$x_1) \approx f_\theta(x_t, h_{t-1})$

How do you “generate” language now?


Sampling Strategies from Next-Token Distribution
Original Distribution
Probability
0.35
0.25
0.18
0.12
0.07
0.03
the
a
one
my
some
his
Greedy (argmax)
Always pick
max prob!
Temperature T=0.5 (sharper)
0.52
0.26
0.14
0.06
0.02
0.00
Temperature T=2.0 (flatter)
0.26
0.22
0.18
0.15
0.12
0.08
Top-k Sampling (k=3)
0.45
0.32
0.23
0.00
Top-p Nucleus (p=0.8)
0.39
0.28
0.20
0.13



`do_sample`	`bool`	Defines whether generation will sample the next token (`True`), or is greedy instead (`False`). Most use cases should set this flag to `True`. Check this guide for more information.
`temperature`	`float`	How unpredictable the next selected token will be. High values (`>0.8`) are good for creative tasks, low values (e.g. `<0.4`) for tasks that require “thinking”. Requires `do_sample=True`.

What about translation?

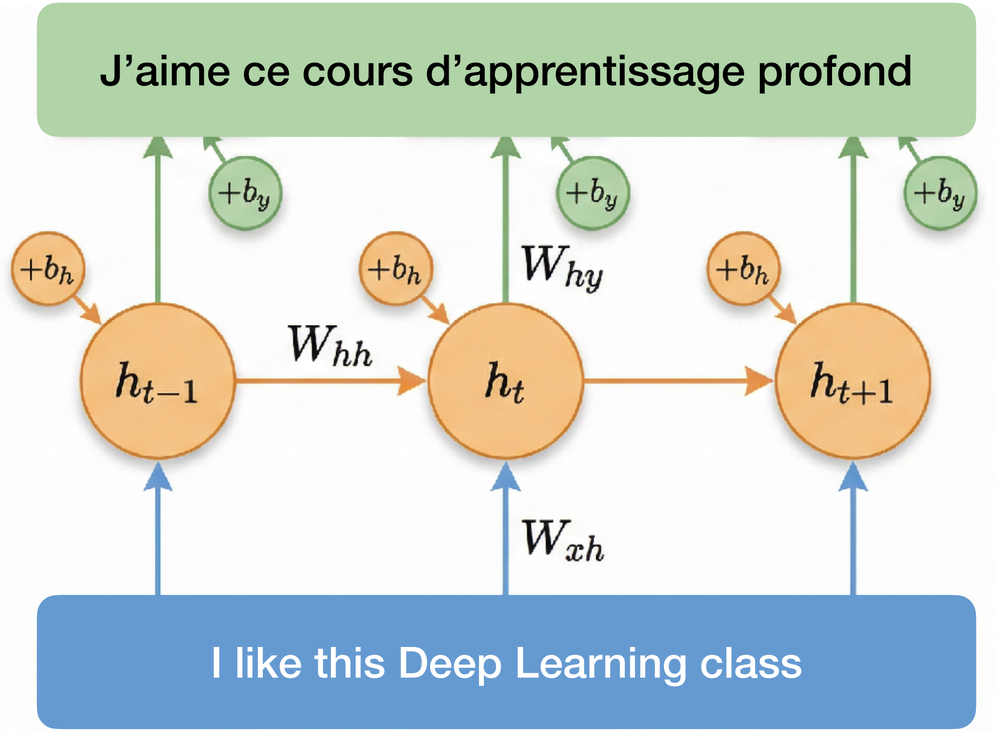
- I like this Deep Learning class <-> J’aime ce cours d’apprentissage profond

What about translation?

- I like this Deep Learning class <-> J’aime ce cours d’apprentissage profond

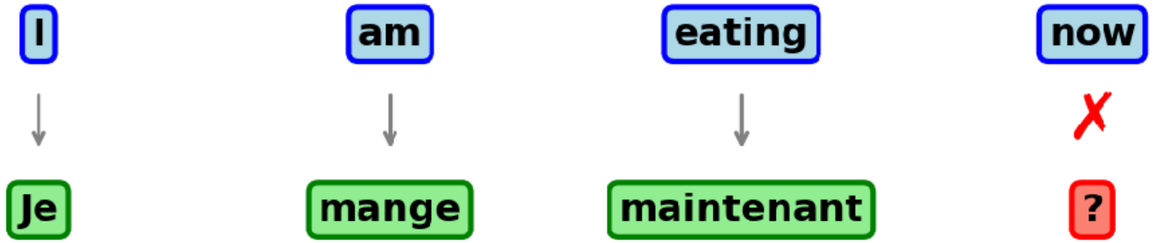
How can we do **Neural Machine Translation**?

What about translation?



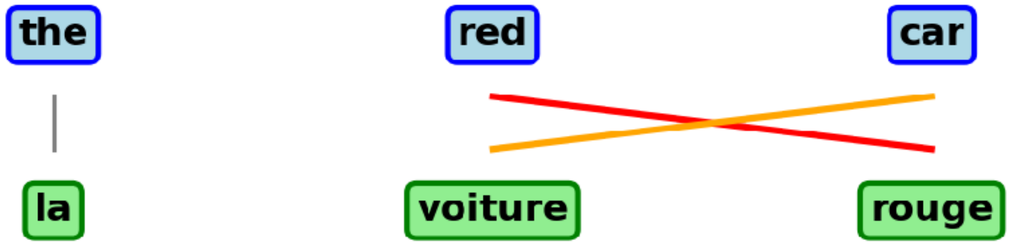
Why Any-to-Any Fails for Translation

① Length Mismatch



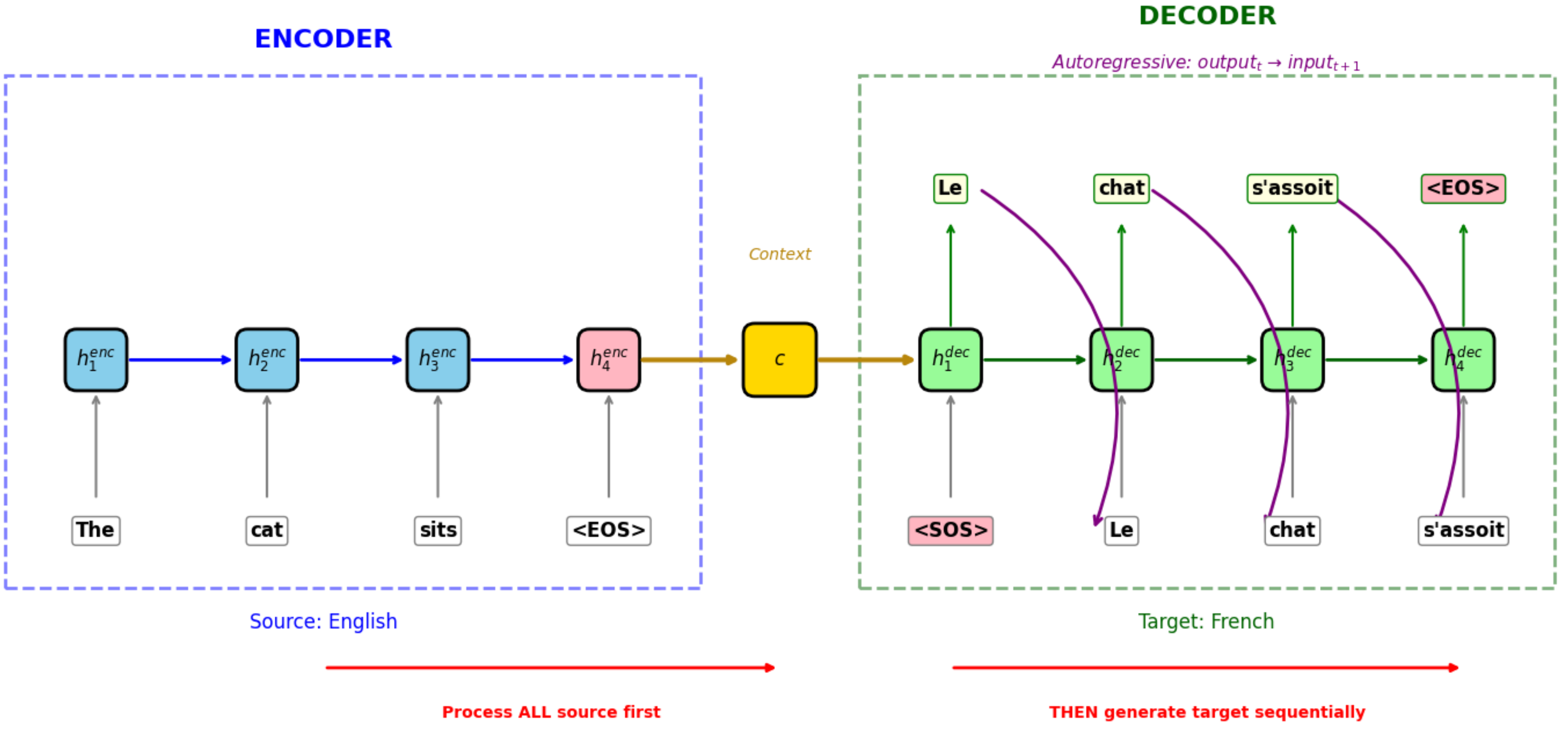
4 → 3 tokens

② Word Order Differs



Alignments cross!

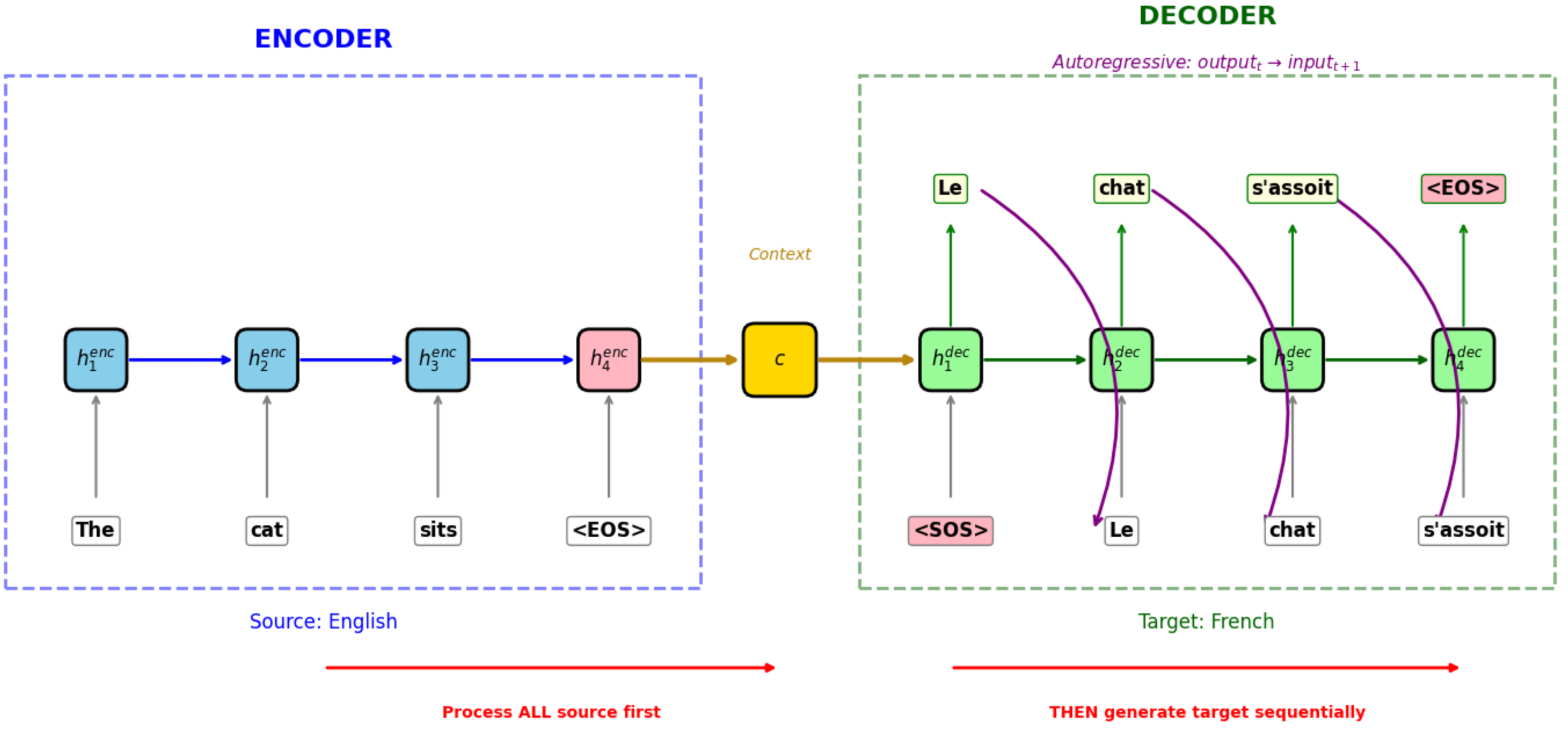
→ Must see ALL source tokens before generating ANY target


Seq2Seq Translation with Autoregressive Decoding
Strict left-to-right generation: cannot peek at future tokens
ENCODER
DECODER
Autoregressive: $output_t \rightarrow input_{t+1}$
Context
h_1^{enc}
h_2^{enc}
h_3^{enc}
h_4^{enc}
c
h_1^{dec}
h_2^{dec}
h_3^{dec}
h_4^{dec}
The
cat
sits
<EOS>
<SOS>
Le
chat
s'assoit
<EOS>
Le
chat
s'assoit
Source: English
Target: French
Process ALL source first
THEN generate target sequentially

What about captioning?

What about captioning?

Randall winning gold at the 100m (sub10s)


Seq2Seq Translation with Autoregressive Decoding
Strict left-to-right generation: cannot peek at future tokens
ENCODER
DECODER
Autoregressive: $output_t \rightarrow input_{t+1}$
Context
h_1^{enc}
h_2^{enc}
h_3^{enc}
h_4^{enc}
c
h_1^{dec}
h_2^{dec}
h_3^{dec}
h_4^{dec}
The
cat
sits
<EOS>
<SOS>
Le
chat
s'assoit
<EOS>
Le
chat
s'assoit
Source: English
Target: French
Process ALL source first
THEN generate target sequentially

Seq2Seq Translation with Autoregressive Decoding
Strict left-to-right generation: cannot peek at future tokens

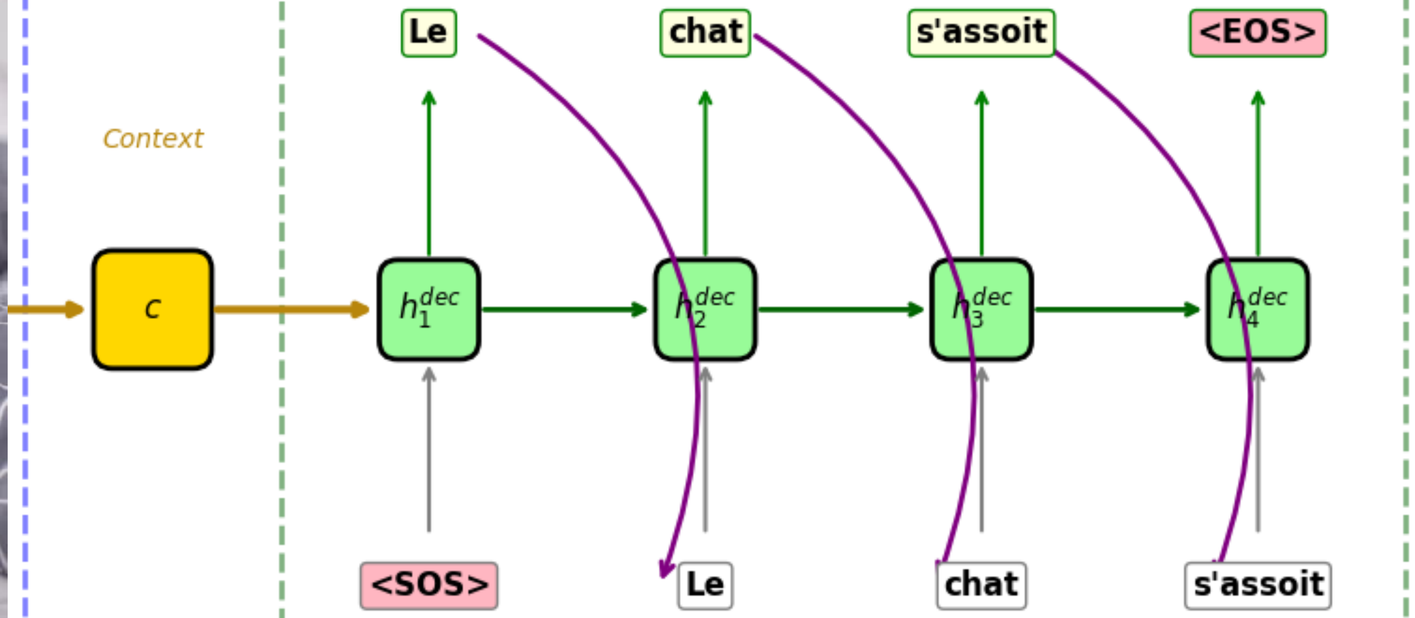
ENCODER

Source: Vision

DECODER

Autoregressive: $output_t \rightarrow input_{t+1}$

Context



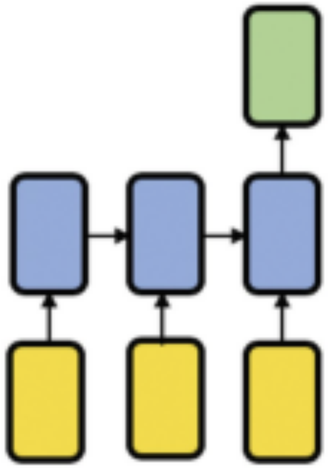
Target: French

Process ALL source first

THEN generate target sequentially

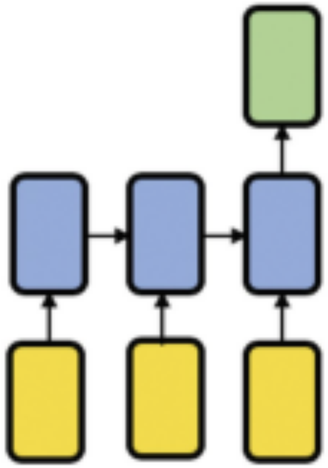
The Sequence Modeling Zoo

The Sequence Modeling Zoo

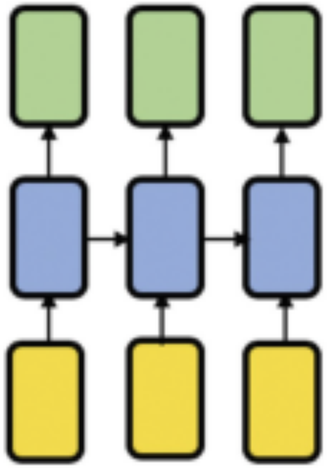


Many-to-One
(e.g., sequence
classification)

The Sequence Modeling Zoo

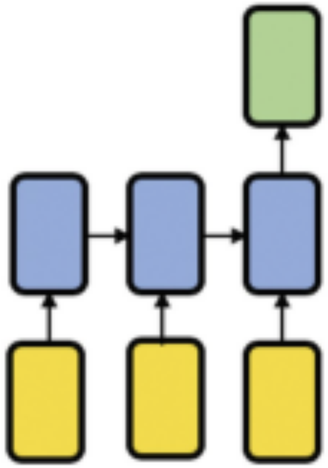


Many-to-One
(e.g., sequence
classification)

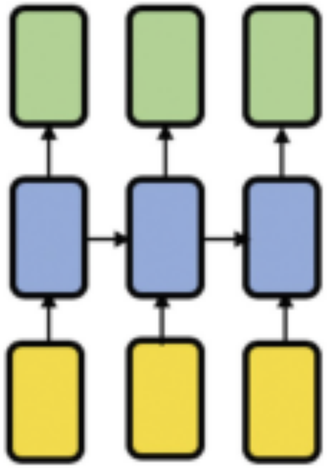


Many-to-Many
(e.g., sequential
prediction)

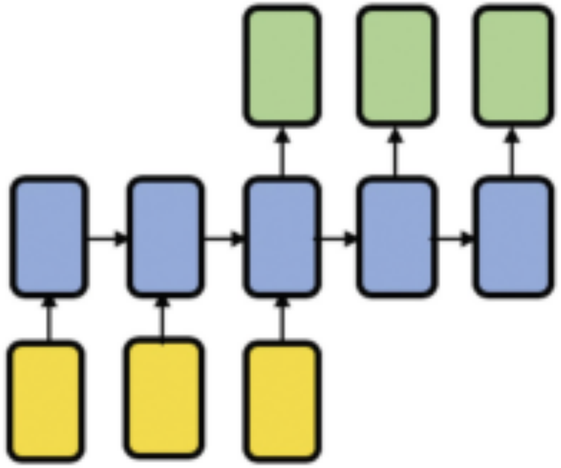
The Sequence Modeling Zoo



Many-to-One
(e.g., sequence classification)

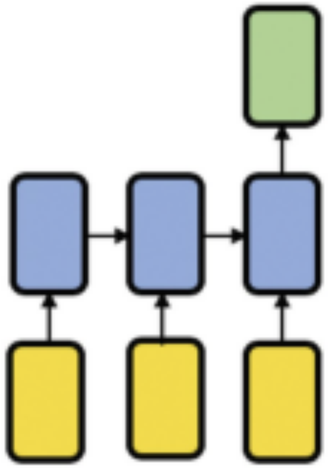


Many-to-Many
(e.g., sequential prediction)

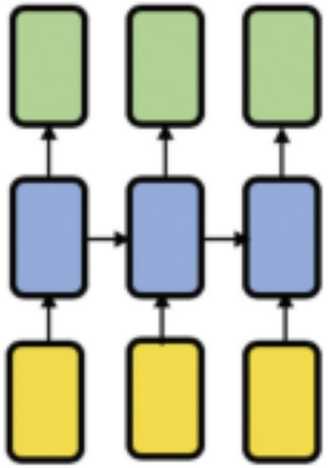


Many-to-Many
(e.g., seq2seq)

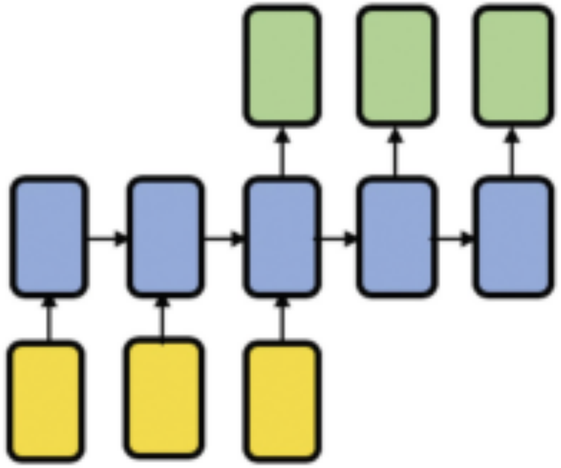
The Sequence Modeling Zoo



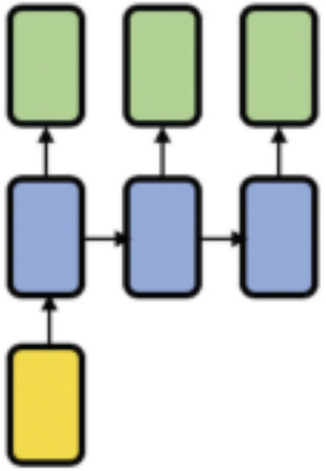
Many-to-One
(e.g., sequence classification)



Many-to-Many
(e.g., sequential prediction)

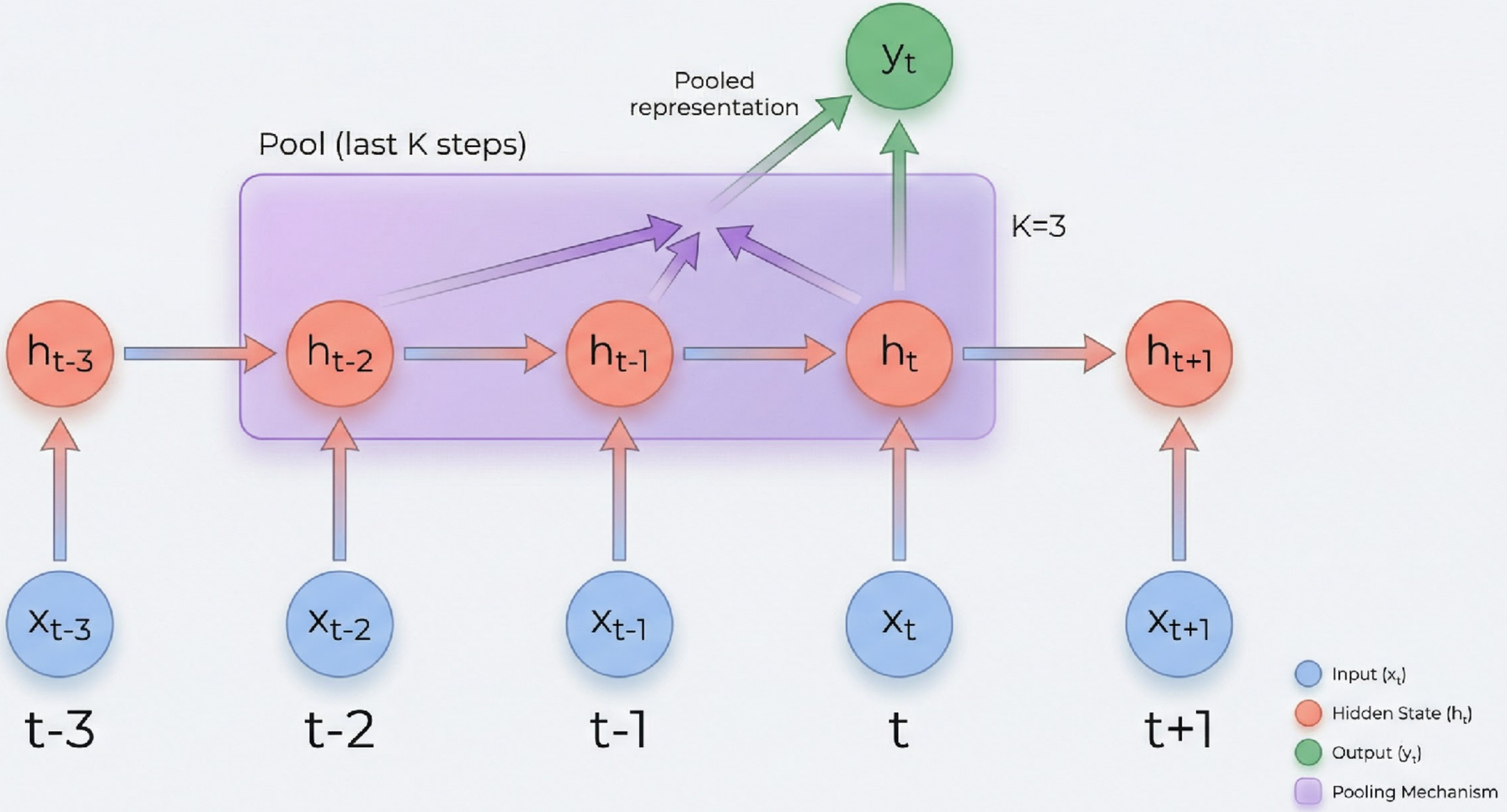


Many-to-Many
(e.g., seq2seq)



One-to-Many
(e.g., image to text)

Recurrent Neural Network with Pooling Architecture



See you Friday!