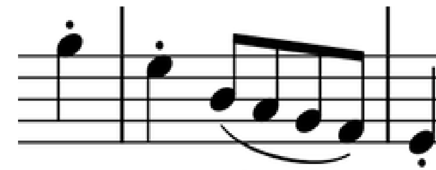
Deep learning

Dr. Aissa Boulmerka
a.boulmerka@centre-univ-mila.dz

2023-2024

CHAPTER 9
RECURRENT NEURAL NETWORK (RNN)

Examples of sequence data

Speech recognition		→	"The quick brown fox jumped over the lazy dog."
Music generation	∅	→	
Sentiment classification	"There is nothing to like in this movie."	→	
DNA sequence analysis	AGCCCCTGTGAGGAACTAG	→	AGCCCCTGTGAGGAACTAG
Machine translation	Voulez vous un verre de jus d'orange?	→	Do you like a glass of orange juice?
Video activity recognition		→	Running
Name entity recognition	Yesterday, Harry Potter met Hermione Granger.	→	Yesterday, Harry Potter met Hermione Granger.

Motivating example

x: Pierre and Marie Curie discovered a radioactive element radium.

$x^{<1>}$ $x^{<2>}$ $x^{<3>}$ .... $x^{<t>}$ ... $x^{<9>}$

$T_x = 9$

y: 1 0 1 1 0 0 0 0 0

$y^{<1>}$ $y^{<2>}$ $y^{<3>}$ .... $y^{<t>}$ ... $y^{<9>}$

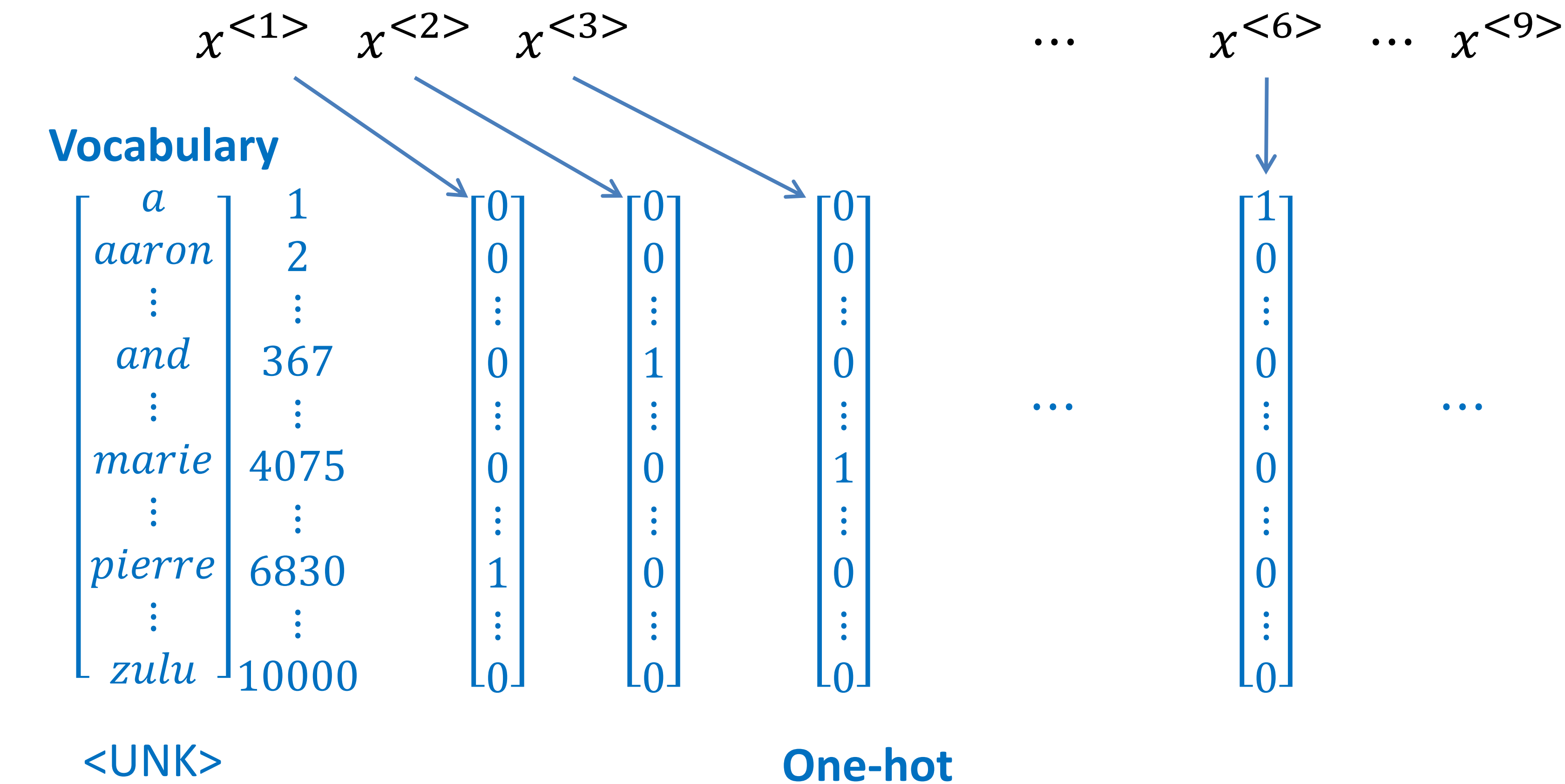
$T_y = 9$

$x^{(i)\langle t \rangle}$ $T_x^{(i)} = 9$

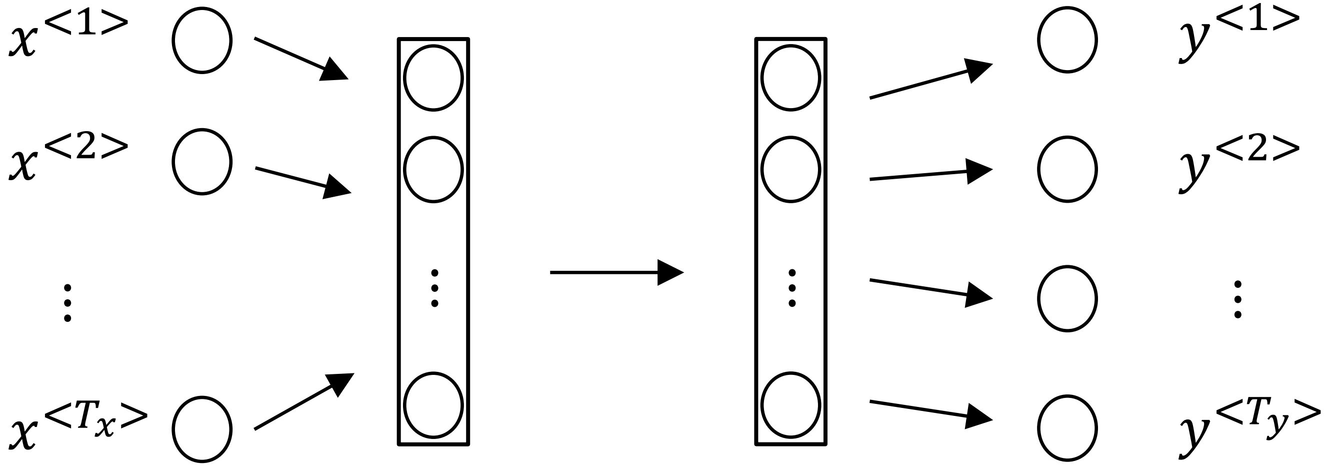
$y^{(i)\langle t \rangle}$ $T_y^{(i)} = 9$

Representing words

x: Pierre and Marie Curie discovered a radioactive element radium.



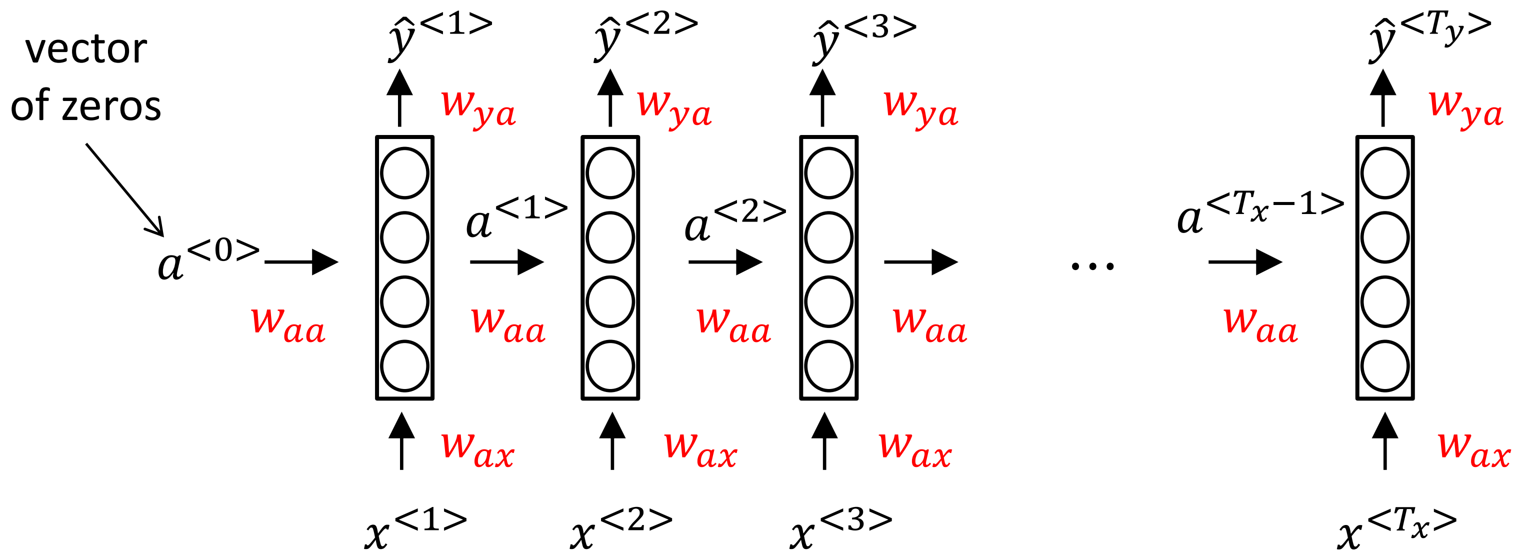
Why not a standard network?



Problems:

- Inputs, outputs can be different lengths in different examples.
- Doesn’t share features learned across different positions of text.

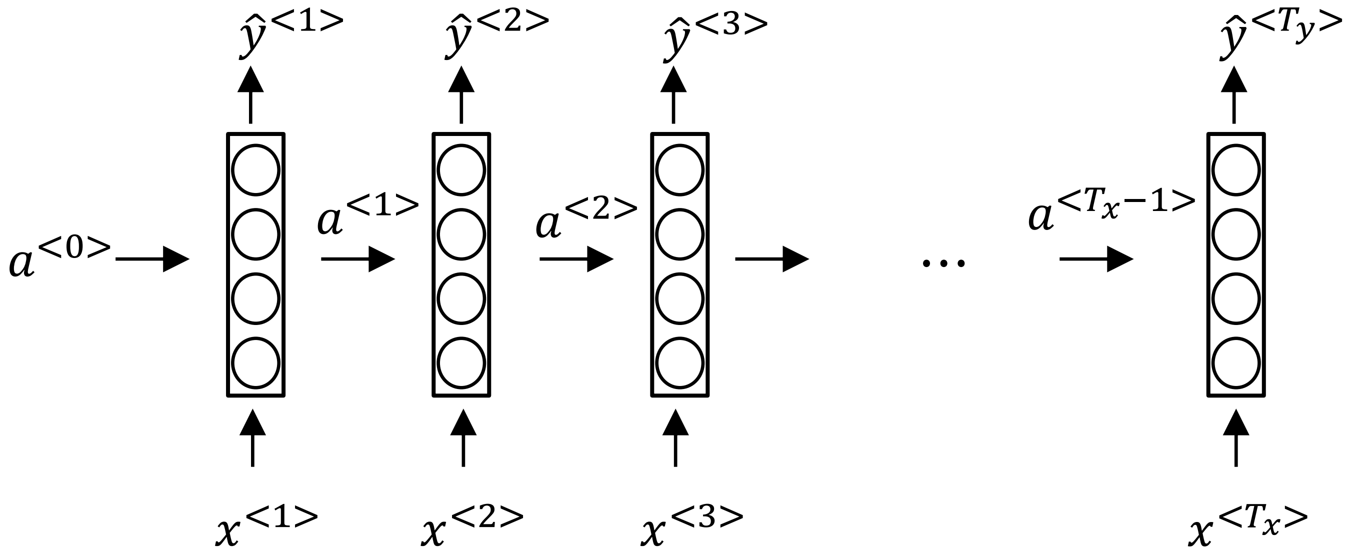
Recurrent Neural Networks



He said, "Teddy Roosevelt was a president."

He said, "Teddy bears are on sale!"

Forward Propagation



$a^{<0>} = \vec{0}$

$a^{<1>} = g_1(W_{aa}a^{<0>} + W_{ax}x^{<1>} + b_a)$ ← tanh/ReLU

$\hat{y}^{<1>} = g_2(W_{ya}a^{<1>} + b_y)$ ← Sigmoid

$$a^{<t>} = g(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a)$$
$$\hat{y}^{<t>} = g(W_{ya}a^{<t>} + b_y)$$

Simplified RNN notation

$$a^{<t>} = g(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a)$$

(100,100) 100 (100,10000) 10000

$$\hat{y}^{<t>} = g(W_{ya}a^{<t>} + b_y)$$

$$\hat{y}^{<t>} = g(W_y a^{<t>} + b_y)$$

$$a^{<t>} = g(W_a[a^{<t-1>}, x^{<t>}] + b_a)$$

$$100 \updownarrow \begin{bmatrix} W_{aa} & \vdots & W_{ax} \end{bmatrix} = W_a$$

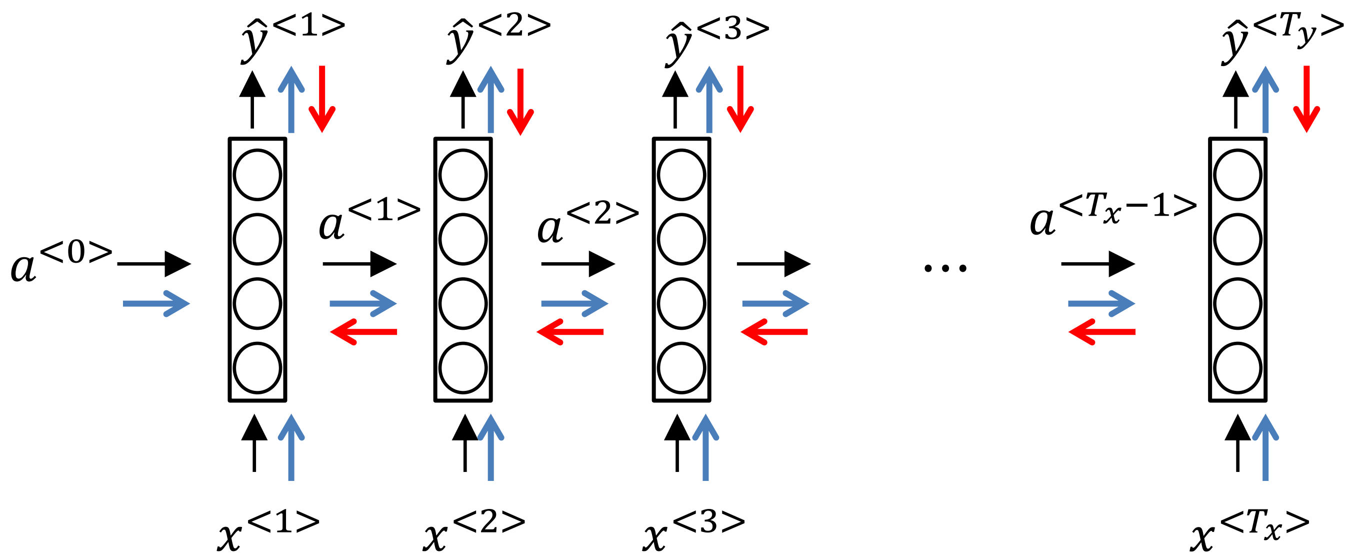
100 ↔ 10000 ↔ (100,10100)

$$[a^{<t-1>}, x^{<t>}] = \begin{bmatrix} a^{<t-1>} \\ \hline x^{<t>} \end{bmatrix}$$

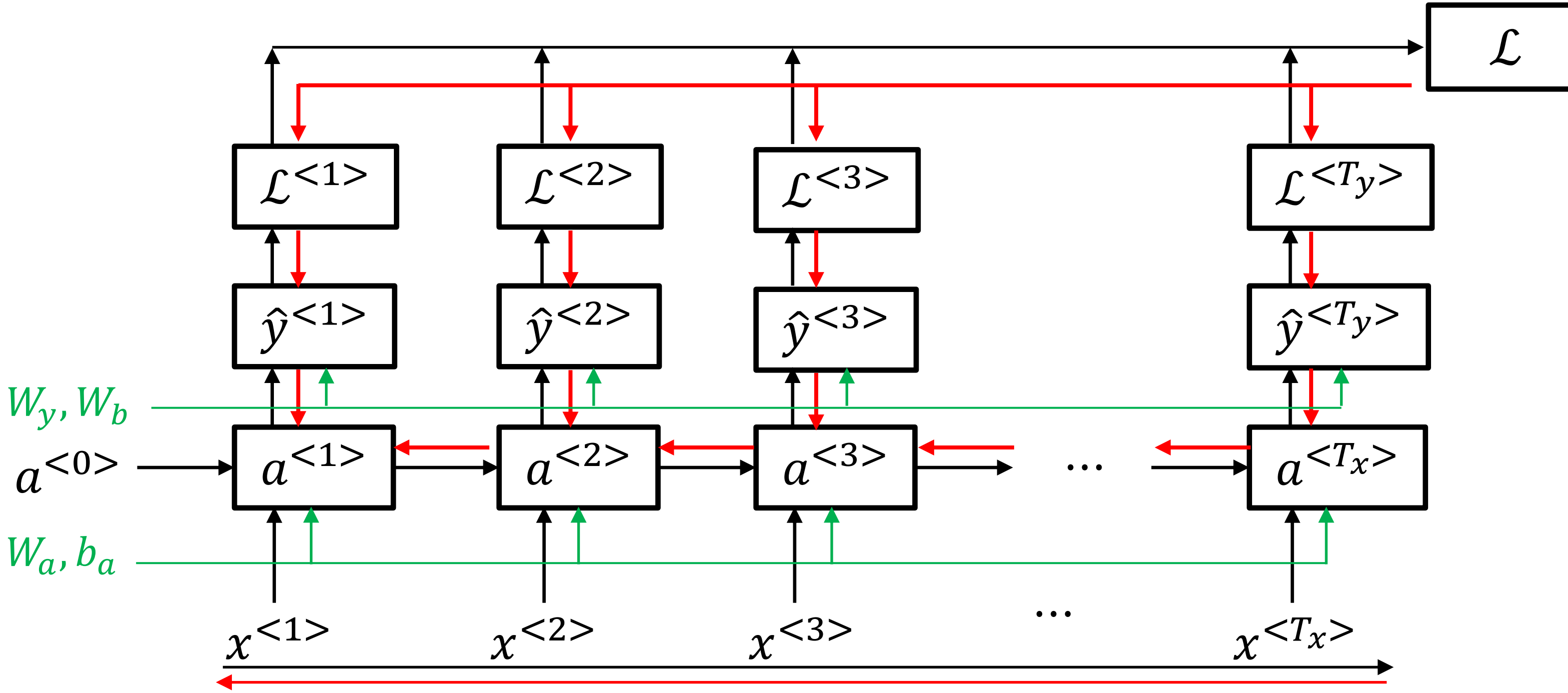
↕ 100 ↕ 10000 ↕ 10100

$$[W_{aa} \;\vdots\; W_{ax}] \begin{bmatrix} a^{<t-1>} \\ x^{<t>} \end{bmatrix} = W_{aa}a^{<t-1>} + W_{ax}x^{<t>}$$

Forward propagation and backpropagation



Forward propagation and backpropagation



$$\mathcal{L}^{<t>}(\hat{y}^{<t>}, y^{<t>}) = -y^{<t>} \log \hat{y}^{<t>} - (1 - y^{<t>}) \log(1 - \hat{y}^{<t>})$$

$$\mathcal{L}(\hat{y}, y) = \sum_{t=1}^{T_y} \mathcal{L}^{<t>}(\hat{y}^{<t>}, y^{<t>})$$

Backpropagation through time

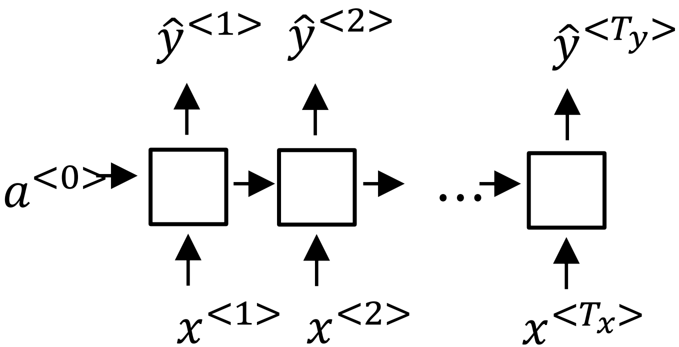
Examples of RNN architectures

Sentiment classification

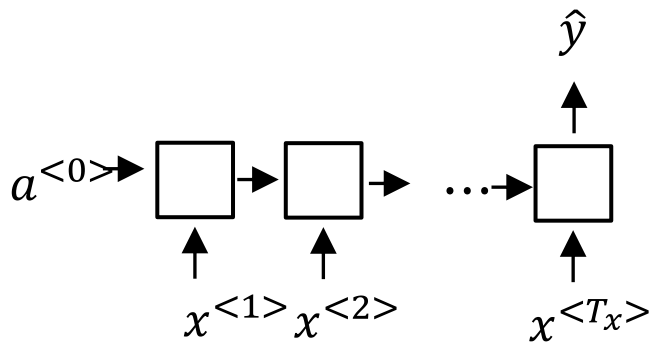
$x = text$

$y = 0/1 \ \ or \ \ 1 \ldots 5$

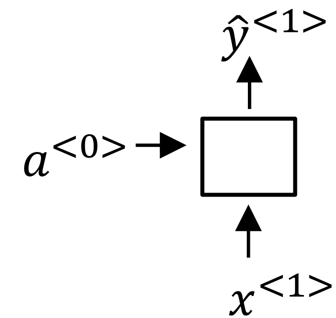
$T_x = T_y$



Many–to–many



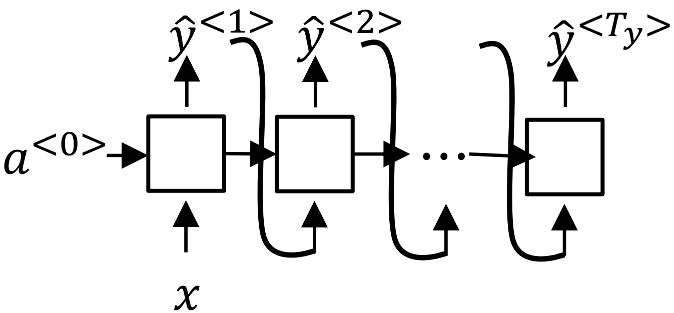
Many–to–one



One–to–one

Examples of RNN architectures

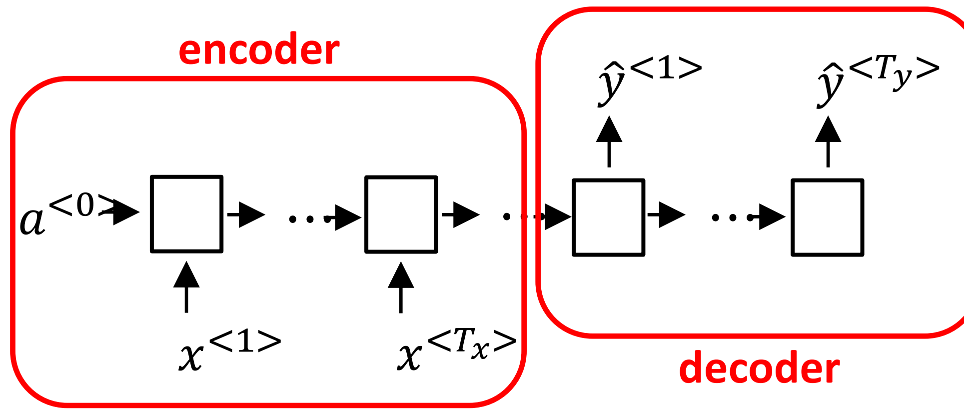
Music generation



$x = \emptyset$

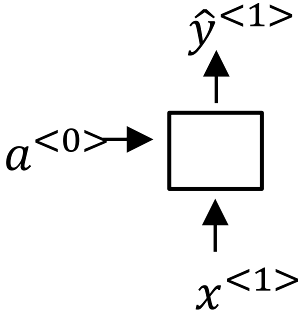
One–to–many

Machine translation

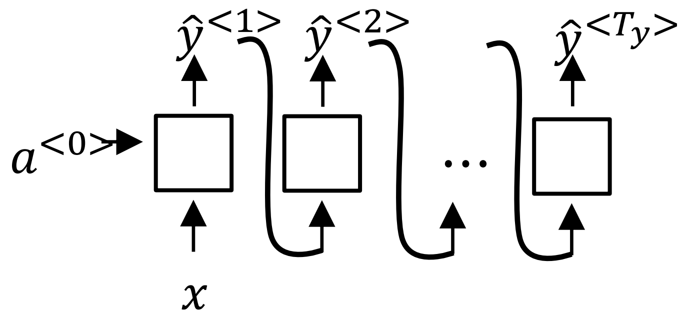


Many–to–many

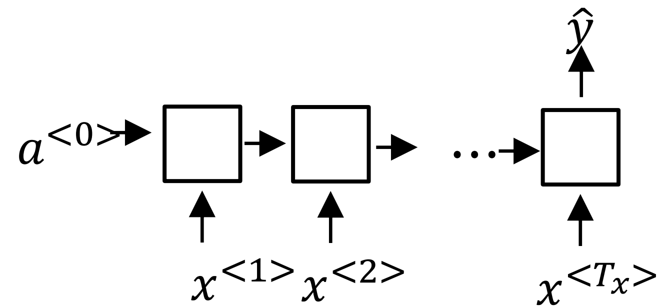
Summary of RNN types



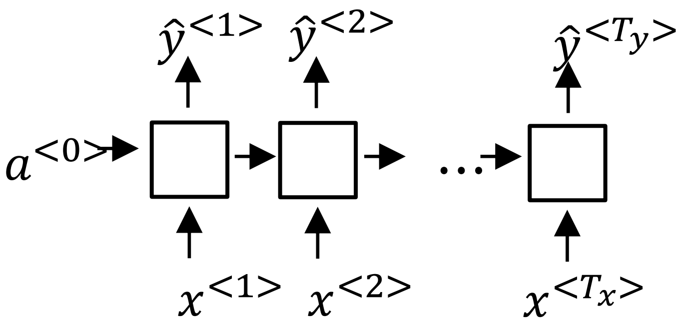
One-to-one



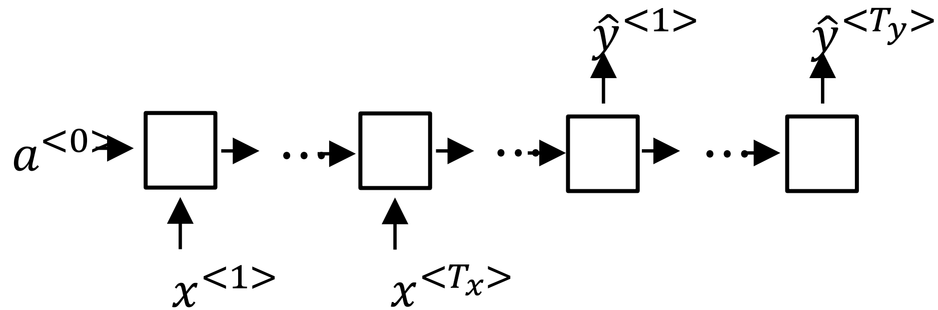
One-to-many



Many-to-one



Many-to-many



Many-to-many

What is language modelling?

Speech recognition

The apple and pair salad.

The apple and pear salad.

$P(\text{The apple and pair salad}) = 3.2 \times 10^{-13}$

$P(\text{The apple and pear salad}) = 5.7 \times 10^{-10}$

$P(\text{Sentence}) = ?$ $\quad P\left(y^{<1>}, y^{<2>}, \ldots, y^{<T_y>}\right)$

Language modelling with an RNN

Training set: large corpus of english text.

Tokenize

Cats average 15 hours of sleep a day. $< \boldsymbol{EOS} >$

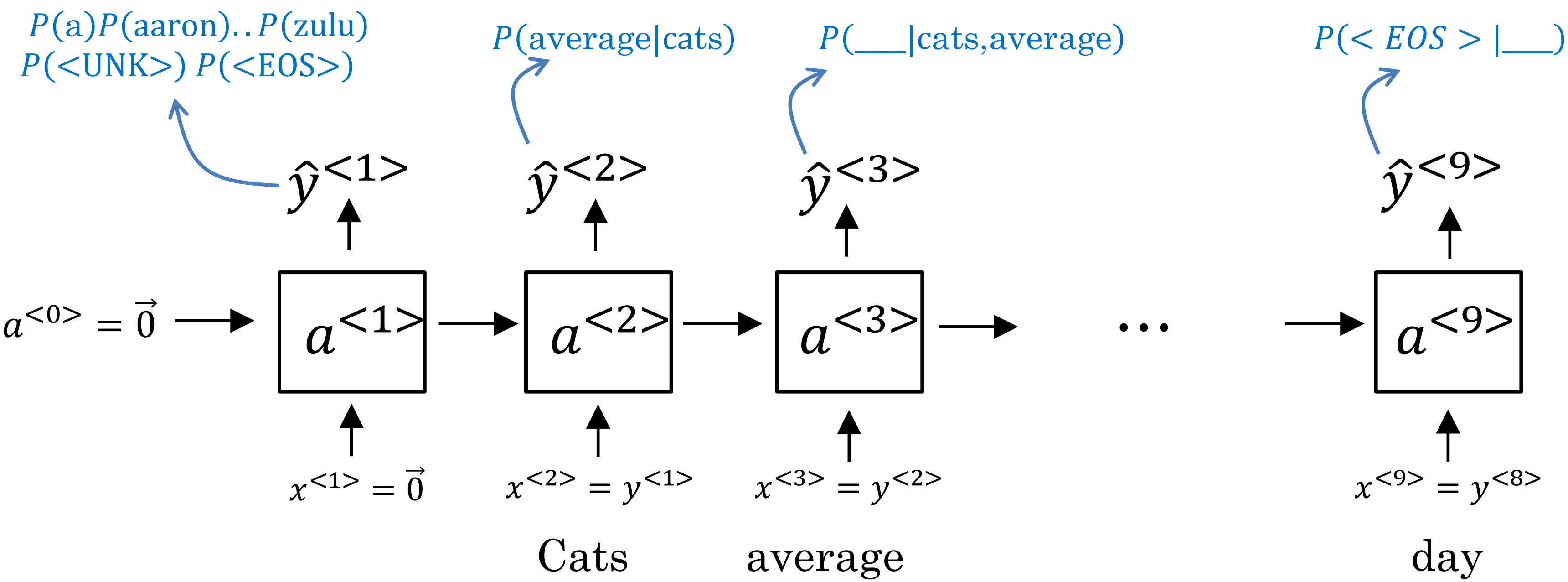
$y^{<1>}$ $y^{<2>}$ $y^{<3>}$ …. $y^{<t>}$ … $y^{<8>}$ $y^{<9>}$

$x^{<t>} = y^{<t-1>}$

The Egyptian ~~Mau~~ is a bread of cat. <EOS>

$< \boldsymbol{UNK} >$

RNN model



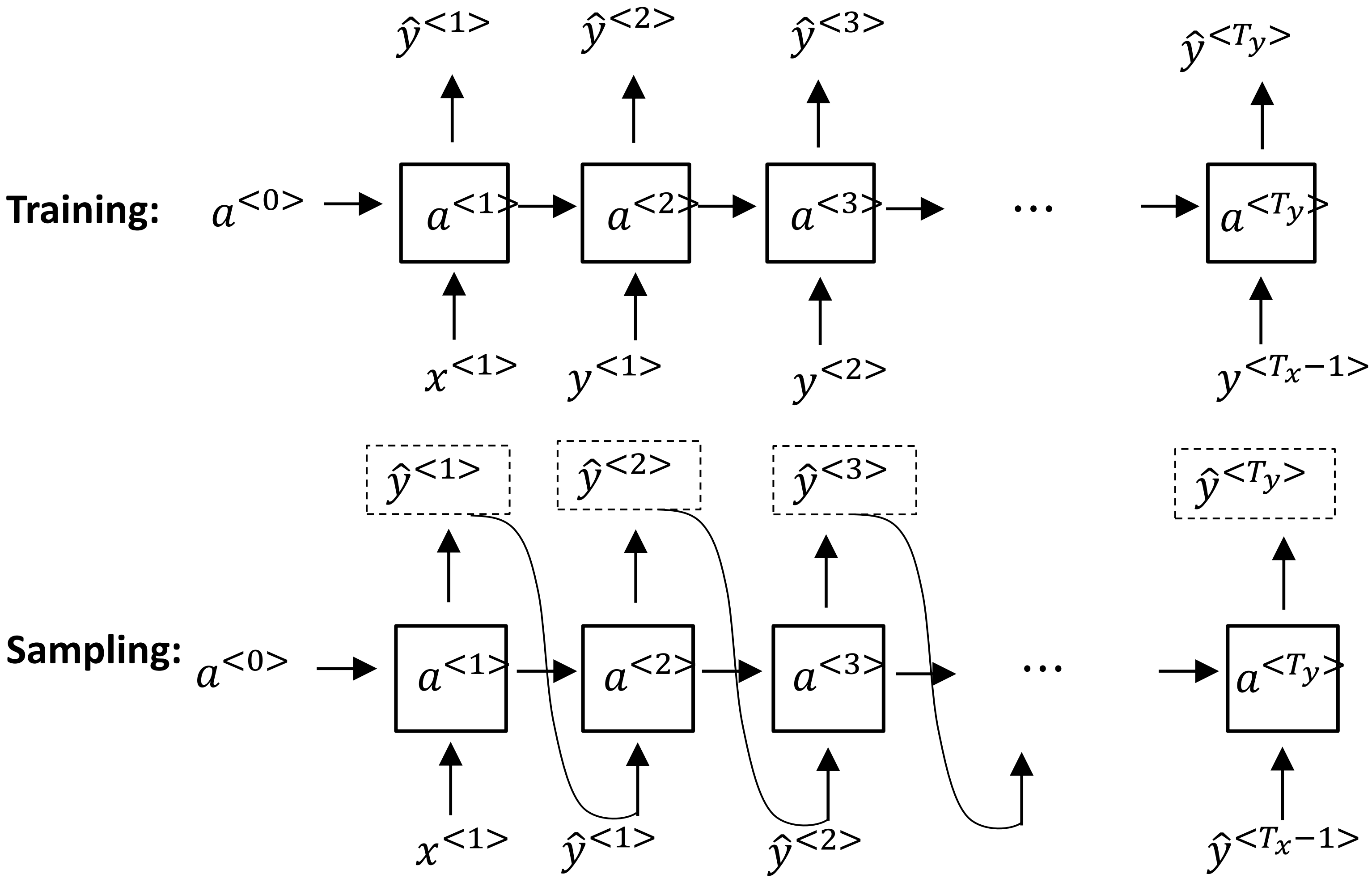
Cats average 15 hours of sleep a day. <EOS>

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

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

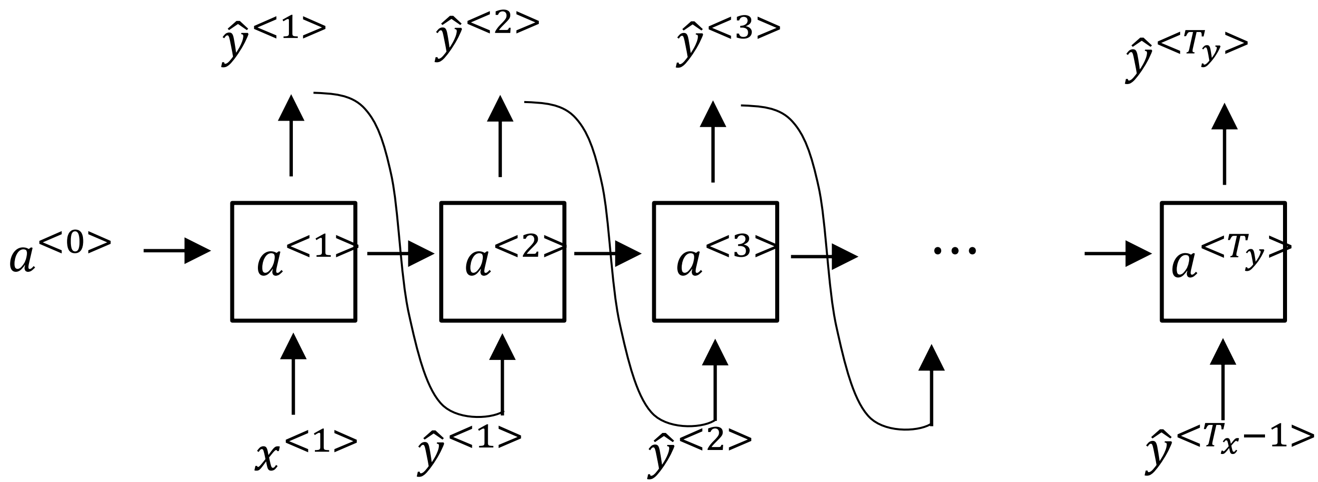
$$P(y^{<1>}, y^{<2>}, y^{<3>})$$
$$= P(y^{<1>}) \times P(y^{<2>}|y^{<1>})$$
$$\times P(y^{<3>}|y^{<1>}, y^{<2>})$$

Sampling a sequence from a trained RNN



Character-level language model

- Vocabulary = [a, aaron, …, zulu, <UNK>]
- Vocabulary = [a, b, c …, z,‘ ’, ‘.’ , ‘,’,’;’,0,1,..,9, A,B,…,Z]



Sequence generation

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.

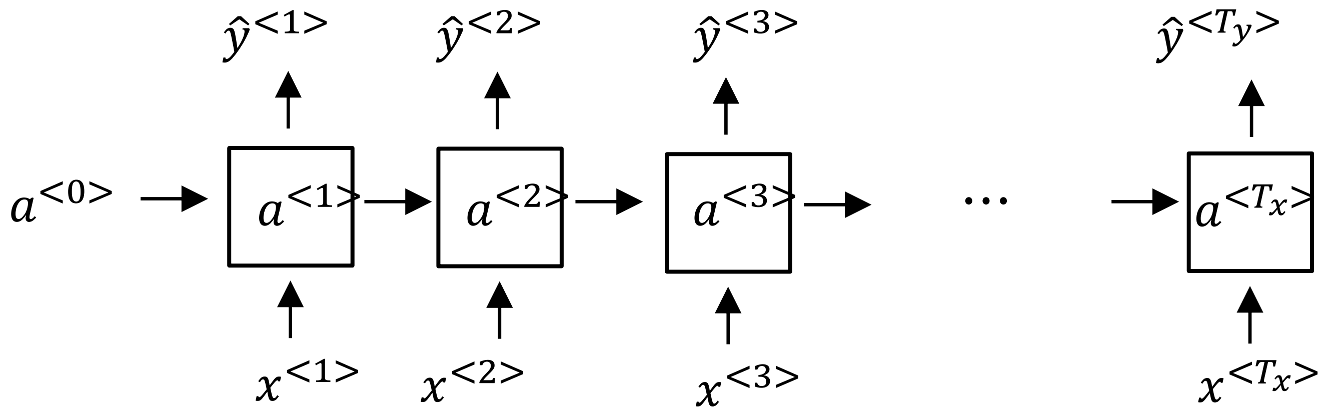
When besser be my love to me see sabl's.

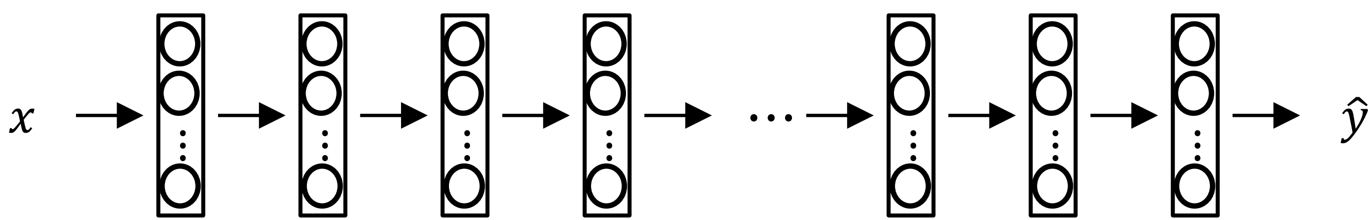
For whose are ruse of mine eyes heaves.

Vanishing gradients with RNNs

The cat, which already ate ..., was full.

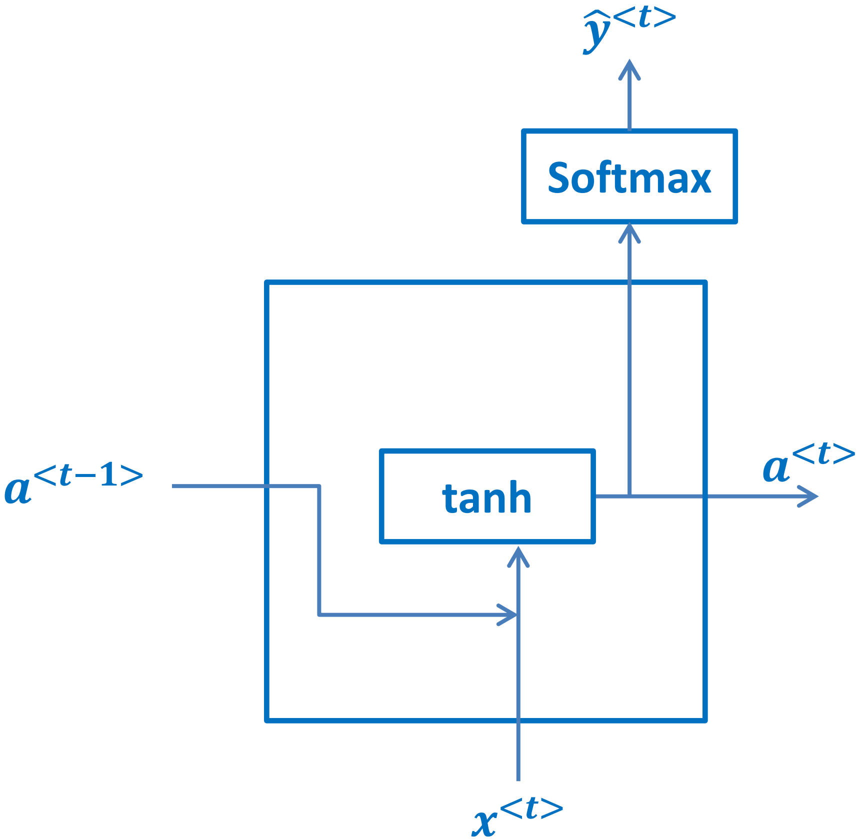
The cats, which already ate ..., were full.





Exploding gradients.

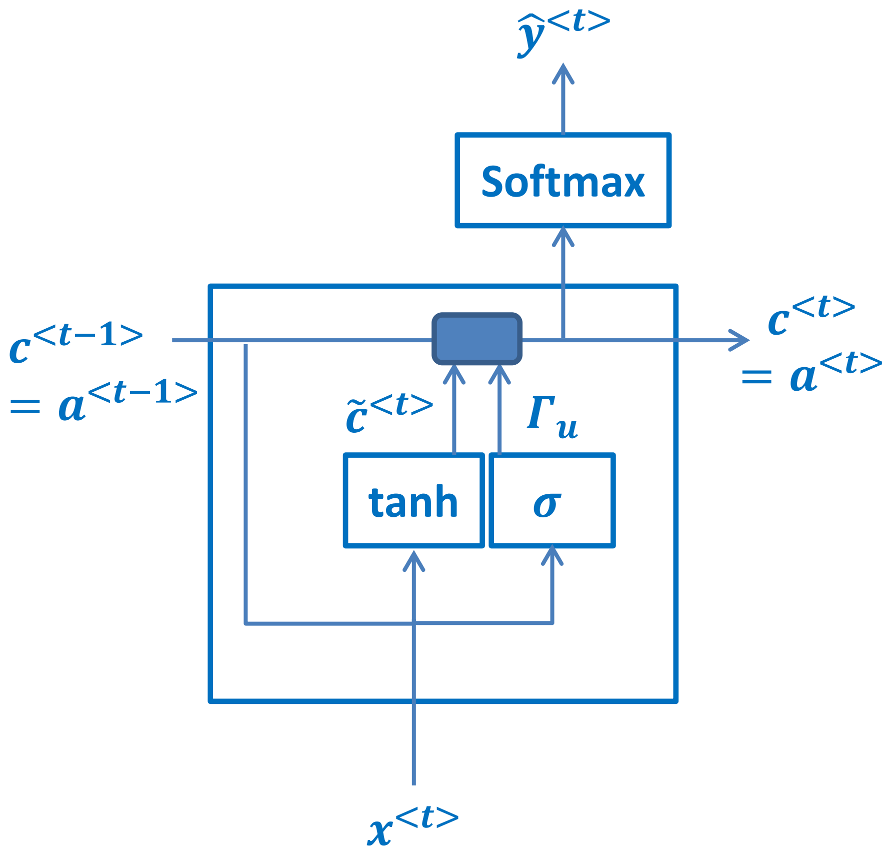
RNN unit



$$a^{<t>} = g(W_a[a^{<t-1>}, x^{<t>}] + b_a)$$

tanh

GRU (simplified)



c = memory cell

$$c^{<t>} = a^{<t>}$$

$$\tilde{c}^{<t>} = \tanh(W_c[c^{<t-1>}, x^{<t>}] + b_c)$$

$$\Gamma_u = \sigma(W_u[\, c^{<t-1>}, x^{<t>}] + b_u)$$

$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + (1 - \Gamma_u) * c^{<t-1>}$$

The cat, which already ate …, was full.

[Cho et al., 2014. On the properties of neural machine translation: Encoder-decoder approaches]
[Chung et al., 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling]

Full GRU

$$\tilde{c}^{<t>} = \tanh(W_c[\Gamma_r * c^{<t-1>}, x^{<t>}] + b_c)$$

$$\Gamma_u = \sigma(W_u[\, c^{<t-1>}, x^{<t>}] + b_u)$$

$$\Gamma_r = \sigma(W_r[\, c^{<t-1>}, x^{<t>}] + b_r)$$

$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + (1 - \Gamma_u) * c^{<t-1>}$$

The cat, which ate already, was full.

LSTM in pictures

$$\tilde{c}^{<t>} = \tanh(W_c[a^{<t-1>}, x^{<t>}] + b_c)$$

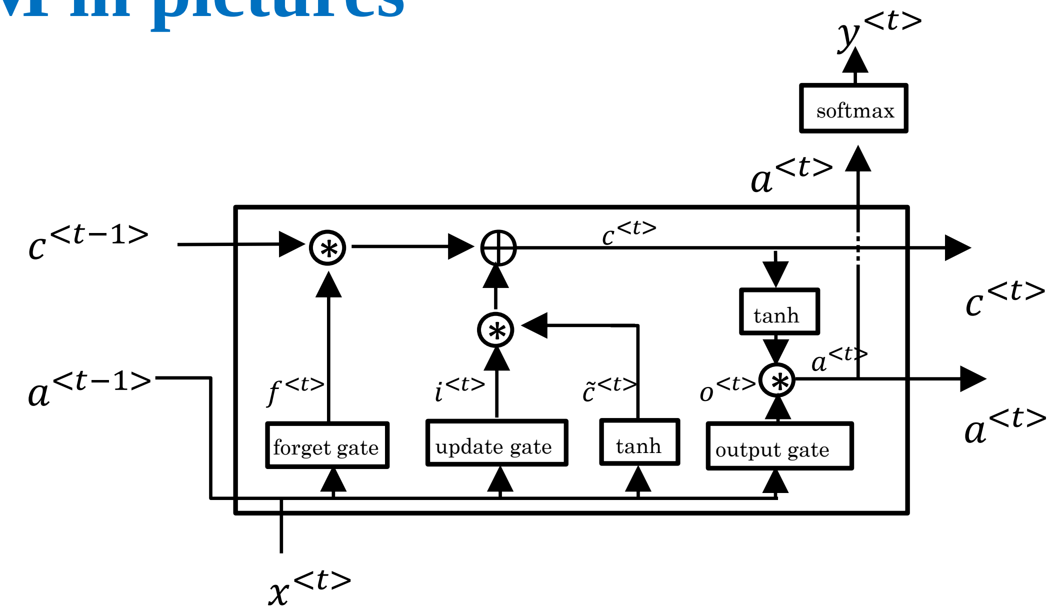
$$\Gamma_u = \sigma(W_u[\, a^{<t-1>}, x^{<t>}] + b_u)$$

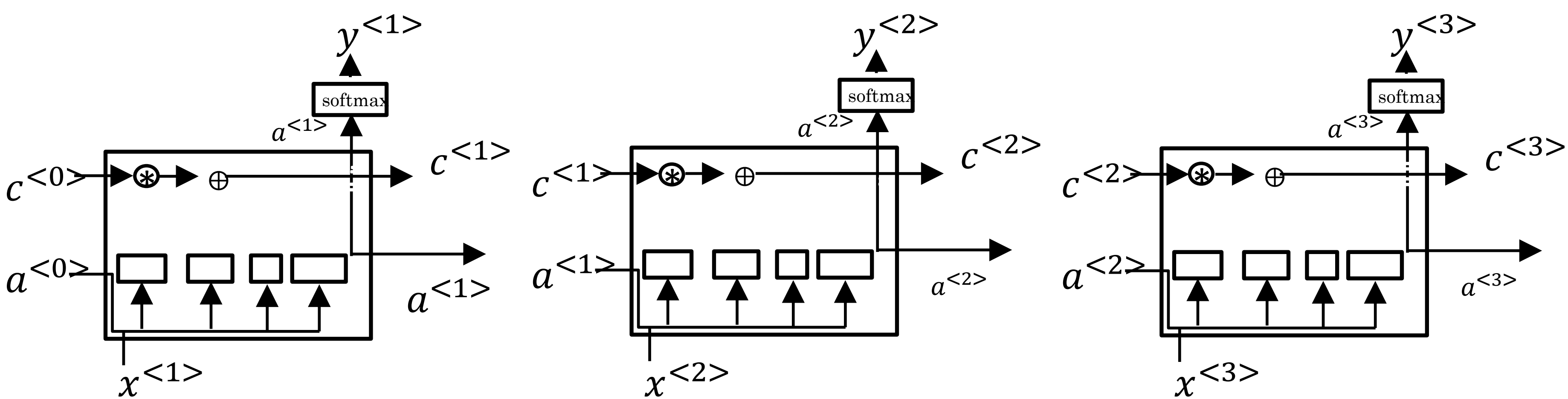
$$\Gamma_f = \sigma(W_f[\, a^{<t-1>}, x^{<t>}] + b_f)$$

$$\Gamma_o = \sigma(W_o[\, a^{<t-1>}, x^{<t>}] + b_o)$$

$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + \Gamma_f * c^{<t-1>}$$

$$a^{<t>} = \Gamma_o * \tanh c^{<t>}$$

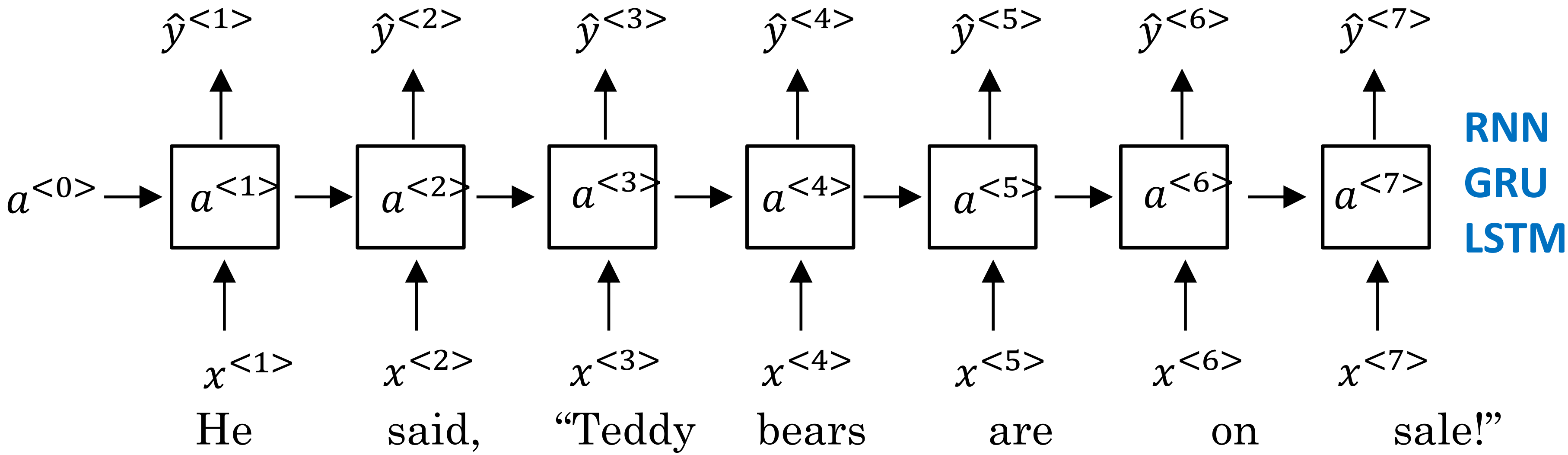




Getting information from the future

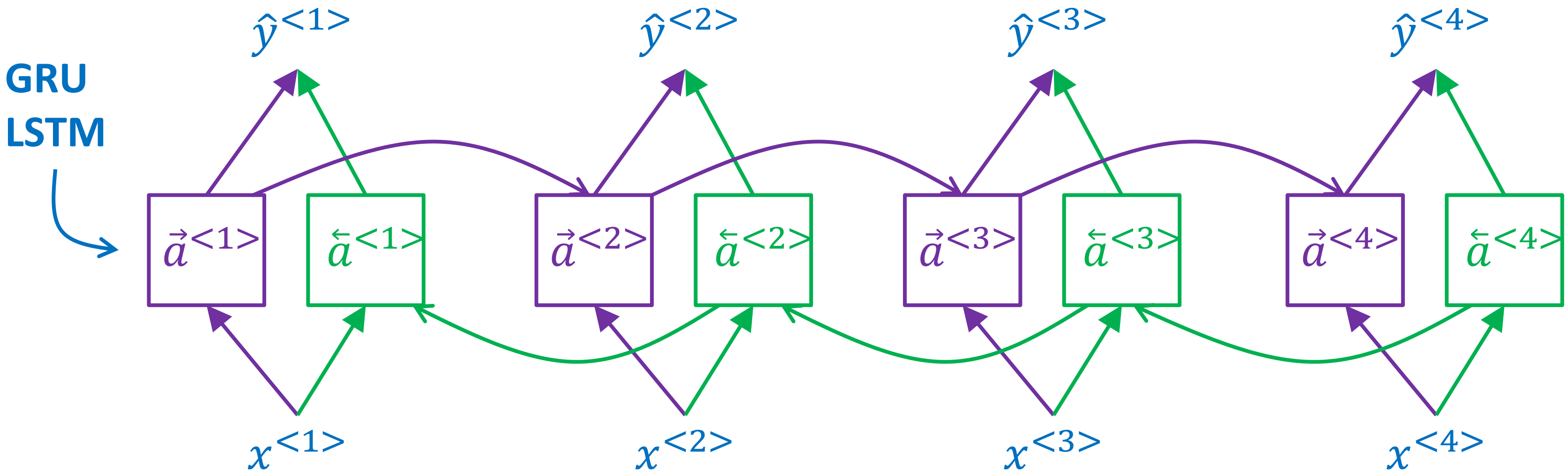
He said, "Teddy bears are on sale!"

He said, "Teddy Roosevelt was a great President!"



Bidirectional RNN (BRNN)

$$\hat{y}^{<t>} = g(W_a[\vec{a}^{<t>}, \overleftarrow{a}^{<t>}] + b_a)$$

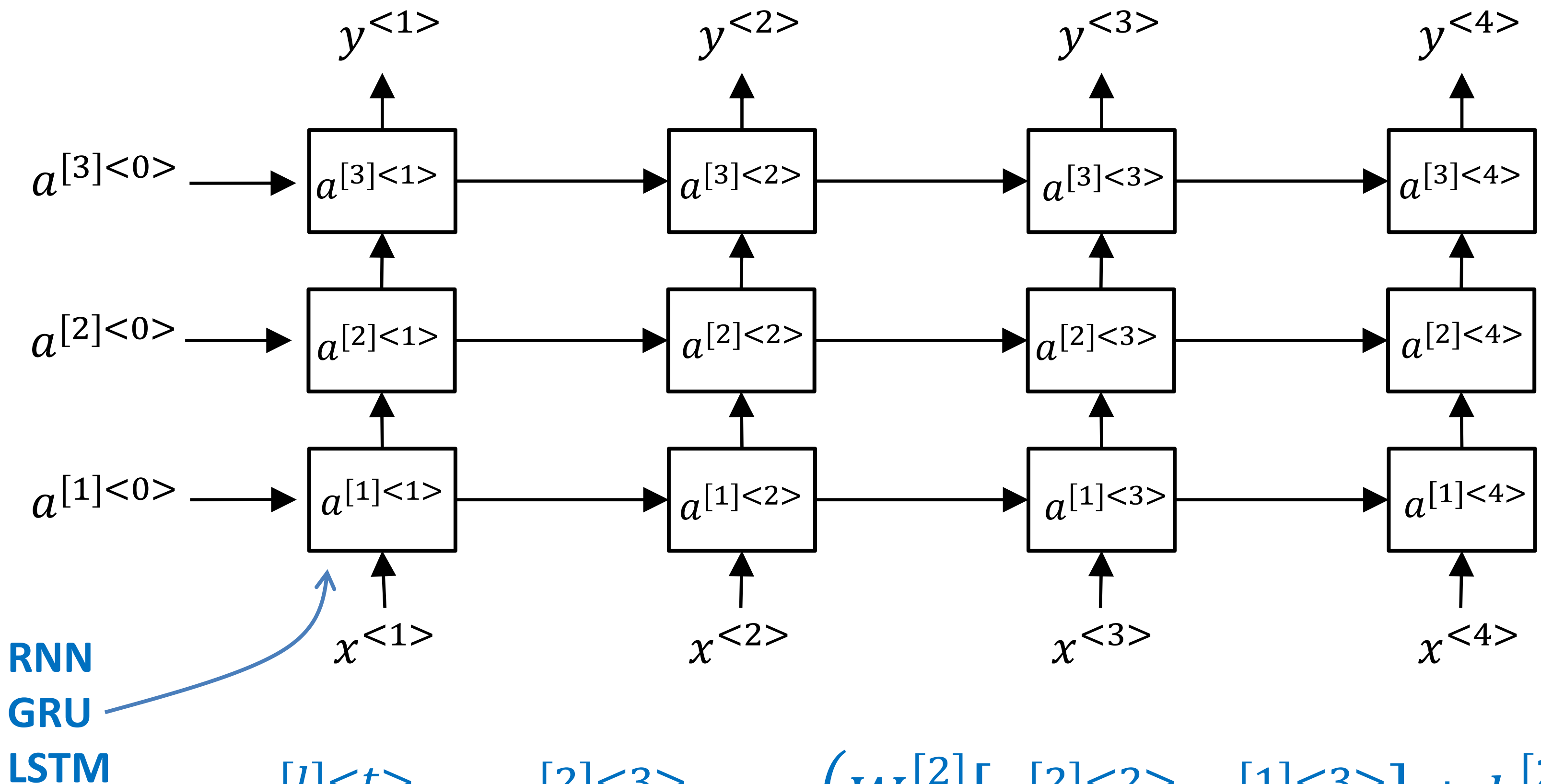


Acyclic graph

He said "Teddy Roosevelt ...."

BRNN w/LSTM

Deep RNN example



$$a^{[l]<t>} \rightarrow a^{[2]<3>} = g\left(W_a^{[2]}\left[a^{[2]<2>}, a^{[1]<3>}\right] + b_a^{[2]}\right)$$

References

- Andrew Ng. Deep learning. Coursera.
- Geoffrey Hinton. Neural Networks for Machine Learning.
- Kevin P. Murphy. Probabilistic Machine Learning An Introduction. MIT Press, 2022.
- MIT Deep Learning 6.S191 (http://introtodeeplearning.com/)