



Рекурентні нейронні мережі обробки природної мови

Лекція 1-2: Рекурентні нейронні мережі (вступ)

Кочура Юрій Петрович
iuriy.kochura@gmail.com
[@y_kochura](https://twitter.com/y_kochura)

Сьогодні

- Будова RNNs
- Пряме та зворотне поширення в RNNs
- Типи RNNs
- Створення мовної моделі
- Створення вибірки послідовностей
- Проблема зникаючого градієнта
- GRU and LSTM

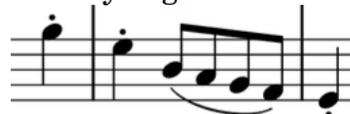
Приклади послідовностей

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

AGCCCCCTGTGAGGAACTAG

→ AGCCC_{red}CTGTGAGGAACTAG

Machine translation

Voulez-vous chanter avec
moi?

→ Do you want to sing with
me?

Video activity recognition



→ Running

Name entity recognition

Yesterday, Harry Potter
met Hermione Granger.

→ Yesterday, Harry Potter
met Hermione Granger.

Позначення

x : Harry Potter and Hermione Granger invented a new spell.

$$x^{<1>} \quad x^{<2>} \quad x^{<3>} \quad x^{<4>} \quad \dots \quad x^{<t>} \quad \dots \quad x^{<9>}$$

y :

$$\begin{matrix} 1 & 1 & 0 & 1 & & 1 & 0 & 0 & 0 \\ y^{<1>} & y^{<2>} & y^{<3>} & y^{<4>} & \dots & y^{<t>} & \dots & y^{<9>} \end{matrix}$$

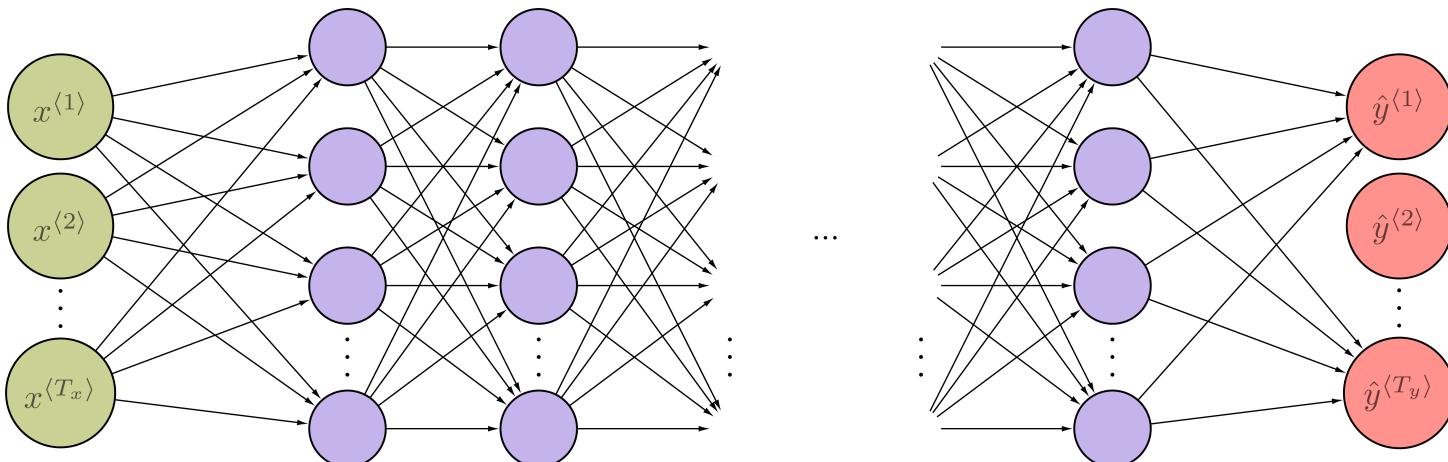
- $T_x = 9$ – довжина вхідної послідовності
- $T_y = 9$ – довжина вихідної послідовності
- $T_x^{(i)}$ – вхідна довжина i -го прикладу
- $T_y^{(i)}$ – вихідна довжина i -го прикладу
- $x^{(i)}$ – i -й навчальний приклад
- $x^{(i)\langle t \rangle}$ – t елемент i -го навчального прикладу
- $y^{(i)}$ – істинна мітка для i -го навчального приклад
- $\hat{y}^{(i)}$ – вихід для i -го навчального приклад
- $\hat{y}^{(i)\langle t \rangle}$ – t елемент на виході для i -го навчального прикладу

Представлення слів

$x:$ Harry Potter and Hermione Granger invented a new spell.
 $x^{<1>} \quad x^{<2>} \quad x^{<3>} \quad x^{<4>} \quad \dots \quad x^{<t>} \quad \dots \quad x^{<9>}$

$$\text{Vocabulary} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{367} \\ \vdots \\ v_{4075} \\ \vdots \\ v_{6830} \\ \vdots \\ v_{10000} \end{bmatrix} = \begin{bmatrix} a \\ aaron \\ \vdots \\ and \\ \vdots \\ harry \\ \vdots \\ potter \\ \vdots \\ bob \end{bmatrix} \quad x^{<1>} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad x^{<2>} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \quad \dots$$

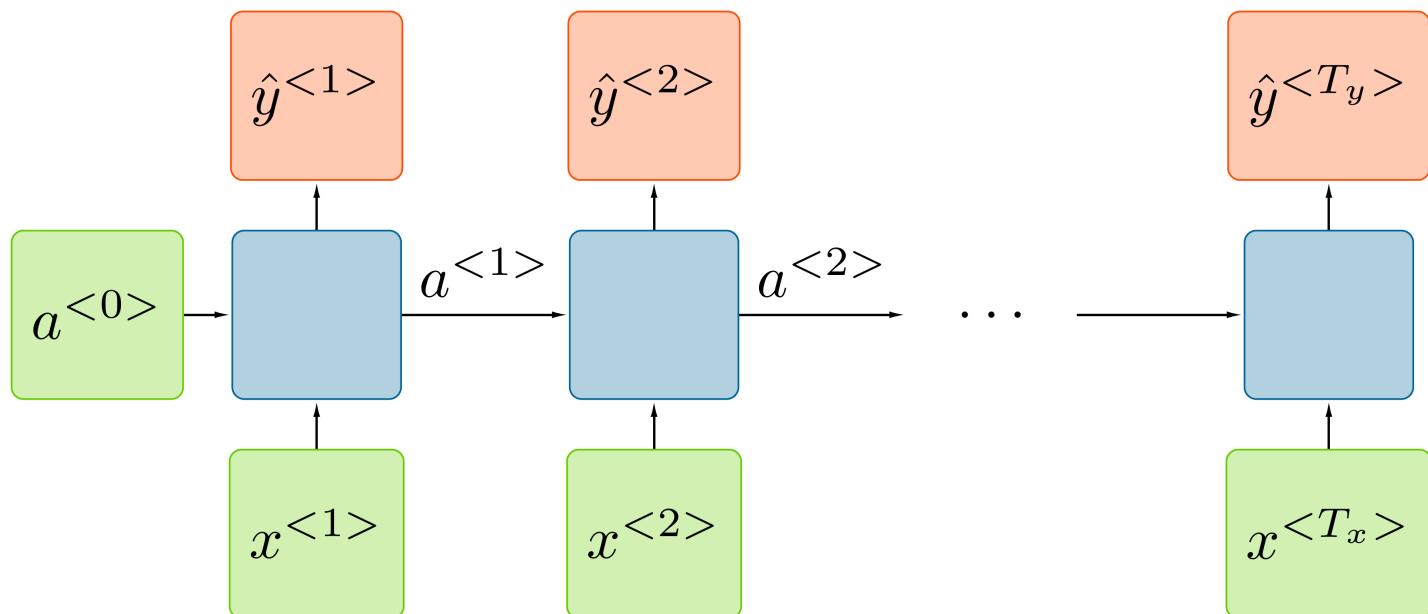
Чому не стандартна мережа?



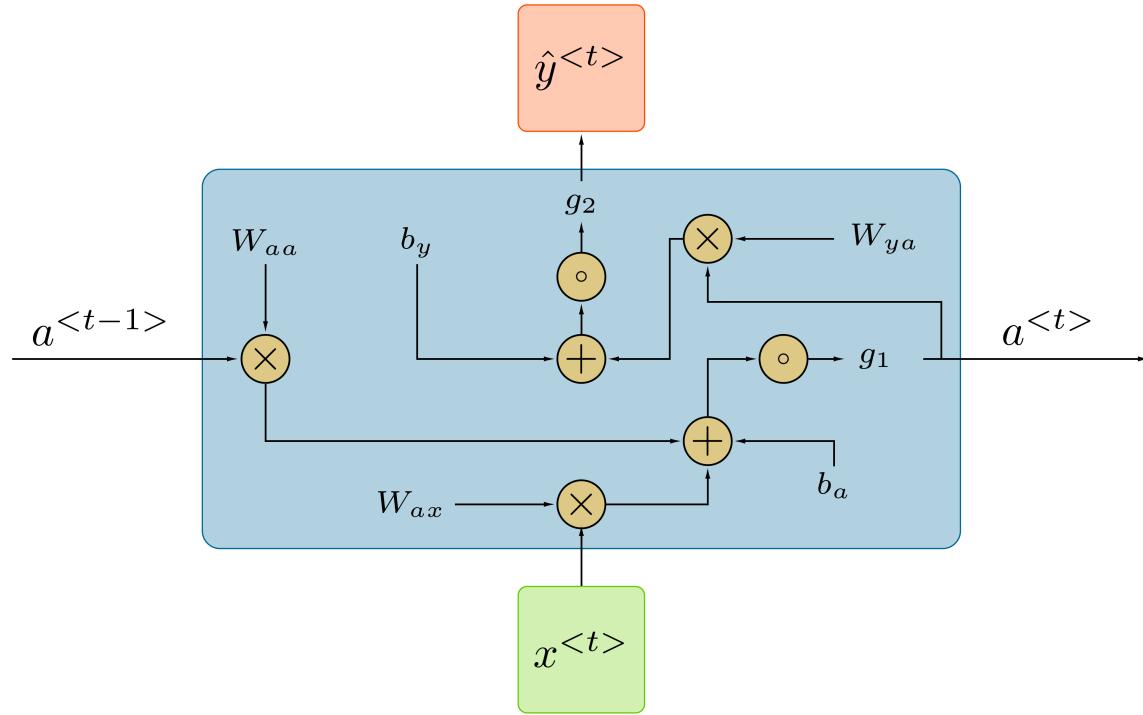
Проблеми

- Входи та виходи можуть мати різну довжину для різних прикладів
- Не поширює ознаки вивчені у різних позиціях тексту

RNN – Many-to-many: $T_x = T_y$



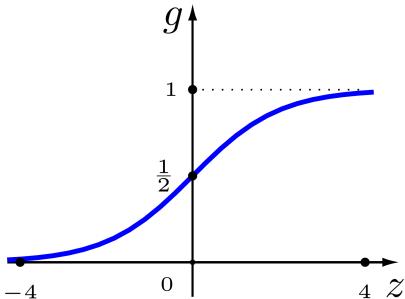
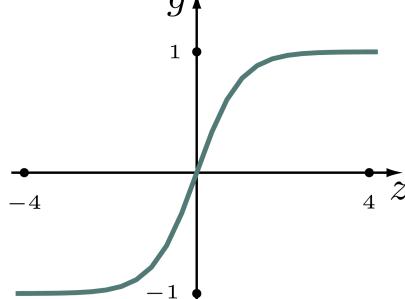
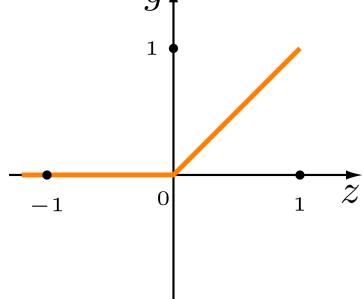
Пряме поширення



Для кожного часового кроку t активація $a^{(t)}$ і вихід $y^{(t)}$ виражаються таким чином:

$$a^{(t)} = g_1(W_{aa}a^{(t-1)} + W_{ax}x^{(t)} + b_a)$$
$$\hat{y}^{(t)} = g_2(W_{ya}a^{(t)} + b_y)$$

Загальнозвживані активаційні функції

Сигмоїда	Гіперболічний тангенс	ReLU
$g(z) = \frac{1}{1 + e^{-z}}$	$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	$g(z) = \max(0, z)$
		

Спрощення позначень в RNN

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

$(100, 100)$ 100 $(100, 10, 000)$

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

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

$$\begin{bmatrix} 100 \\ W_{aa}; W_{ax} \\ \hline 100 & 10,000 \end{bmatrix} = W_a$$

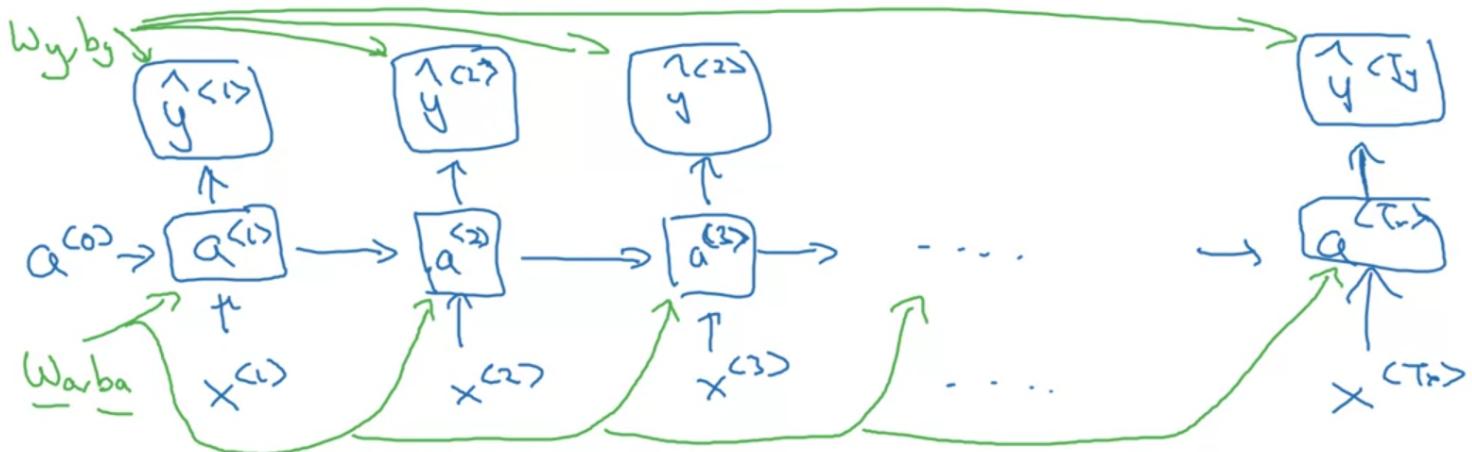
$(100, 10100)$

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

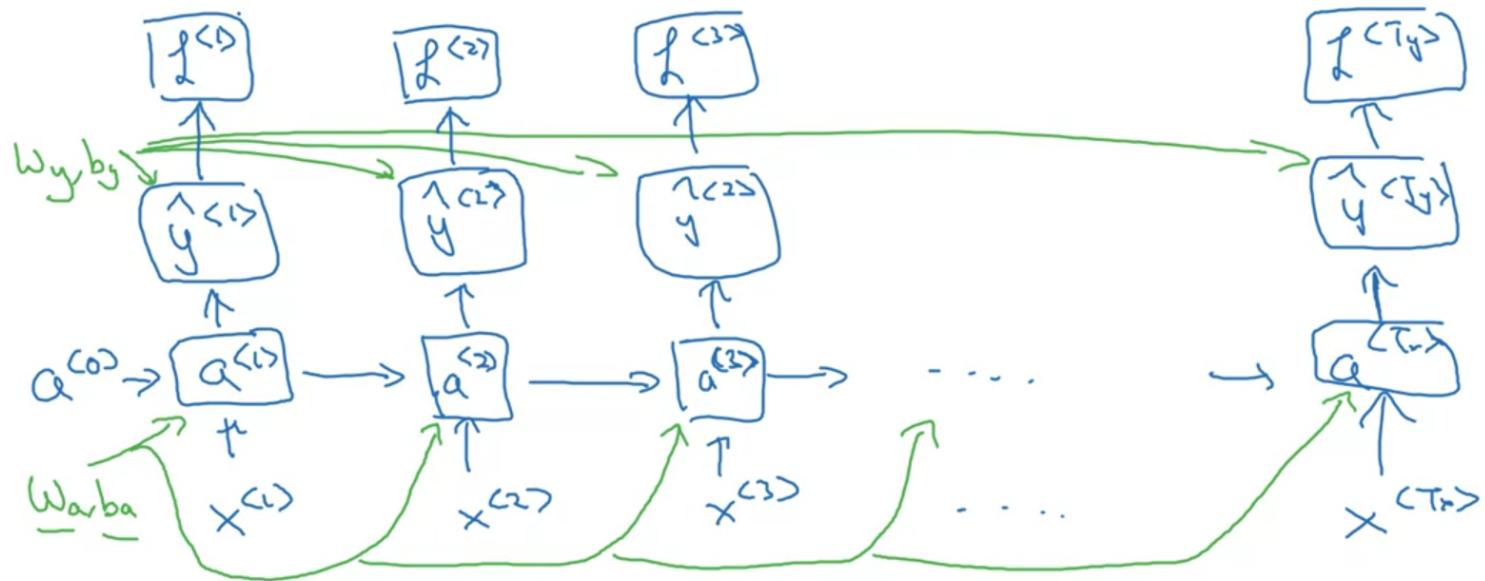
$\begin{array}{c} \uparrow 100 \\ \uparrow 10,000 \\ \downarrow 10100 \end{array}$

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

Зворотне поширення у часі



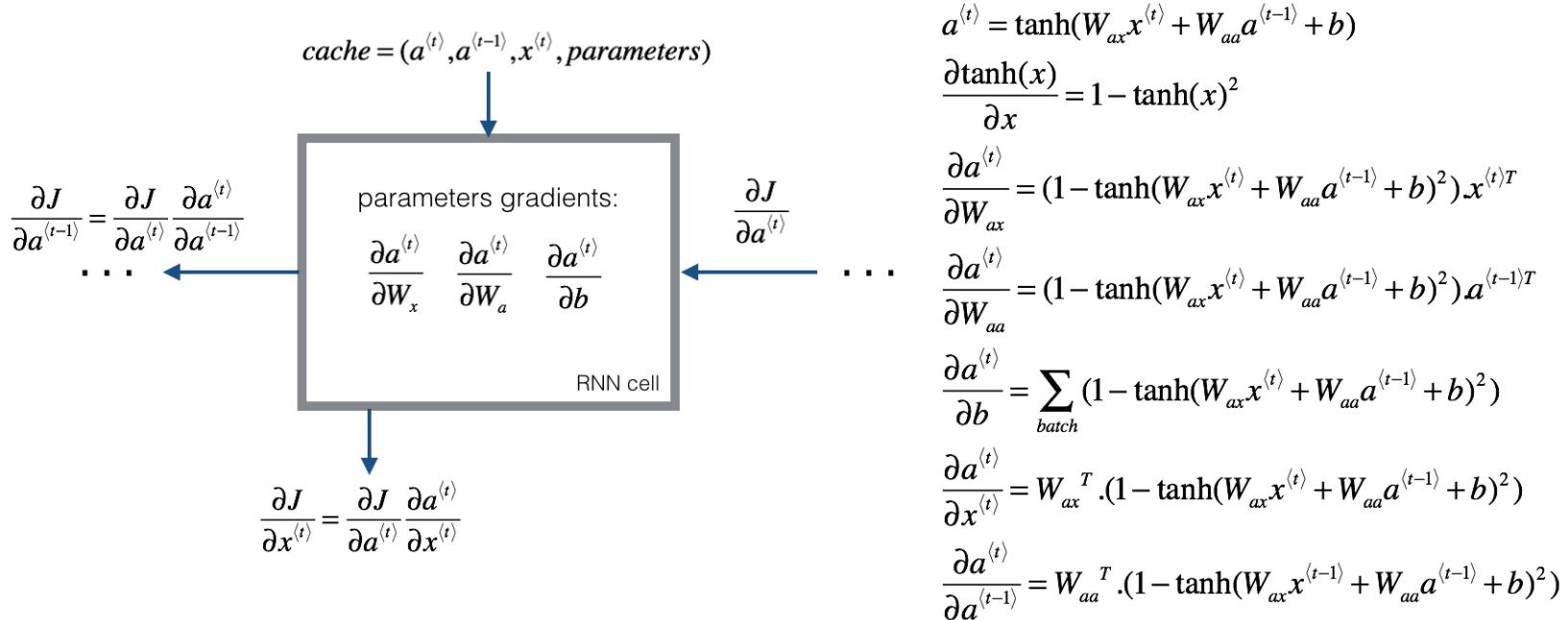
Зворотне поширення у часі



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

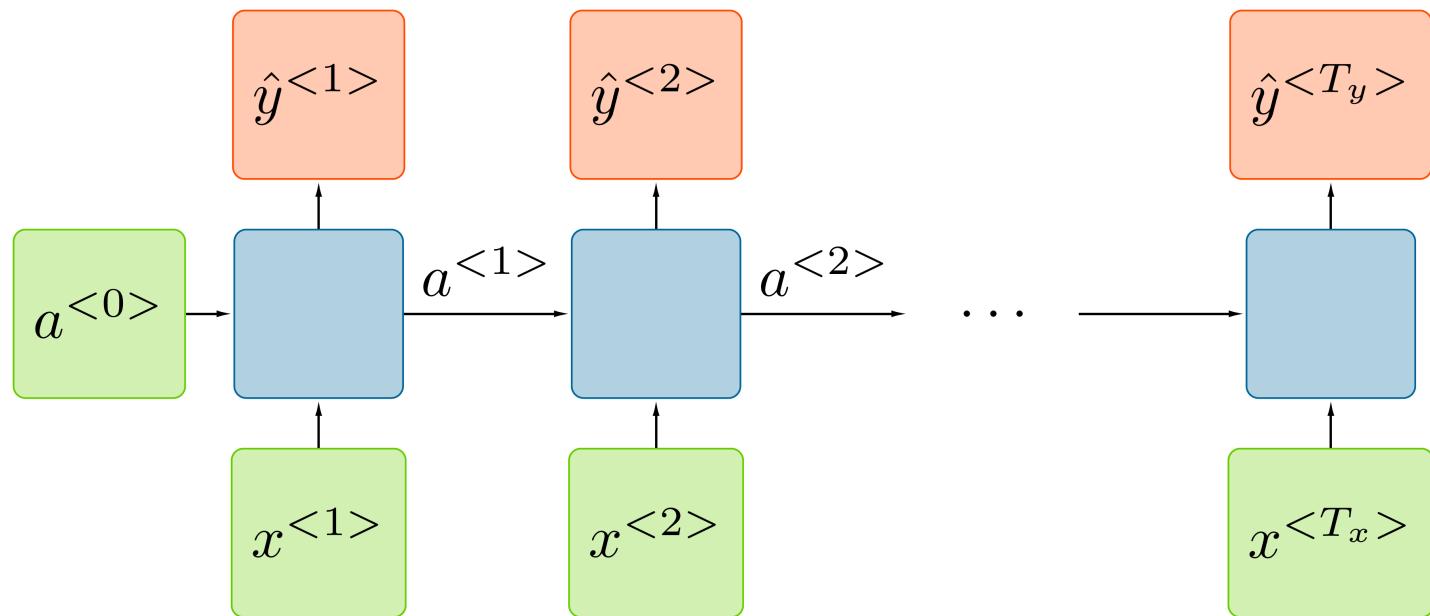
$$J = \sum_{t=1}^{T_y} \mathcal{L}^{(t)}(\hat{y}^{(t)}, y^{(t)})$$

Зворотне поширення у часі



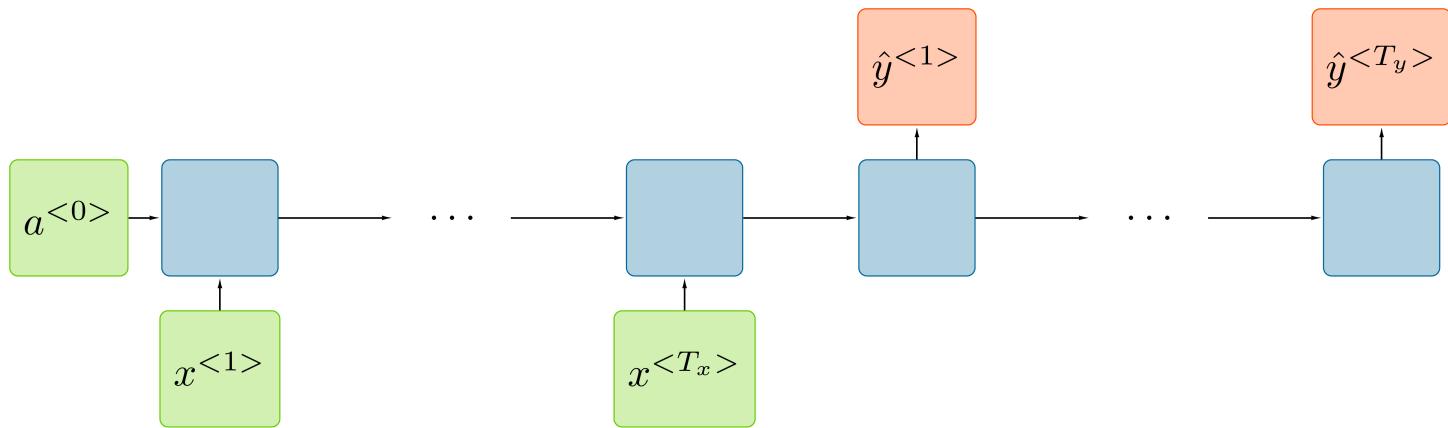
Типи архітектур RNNs

Many-to-many: $T_x = T_y$



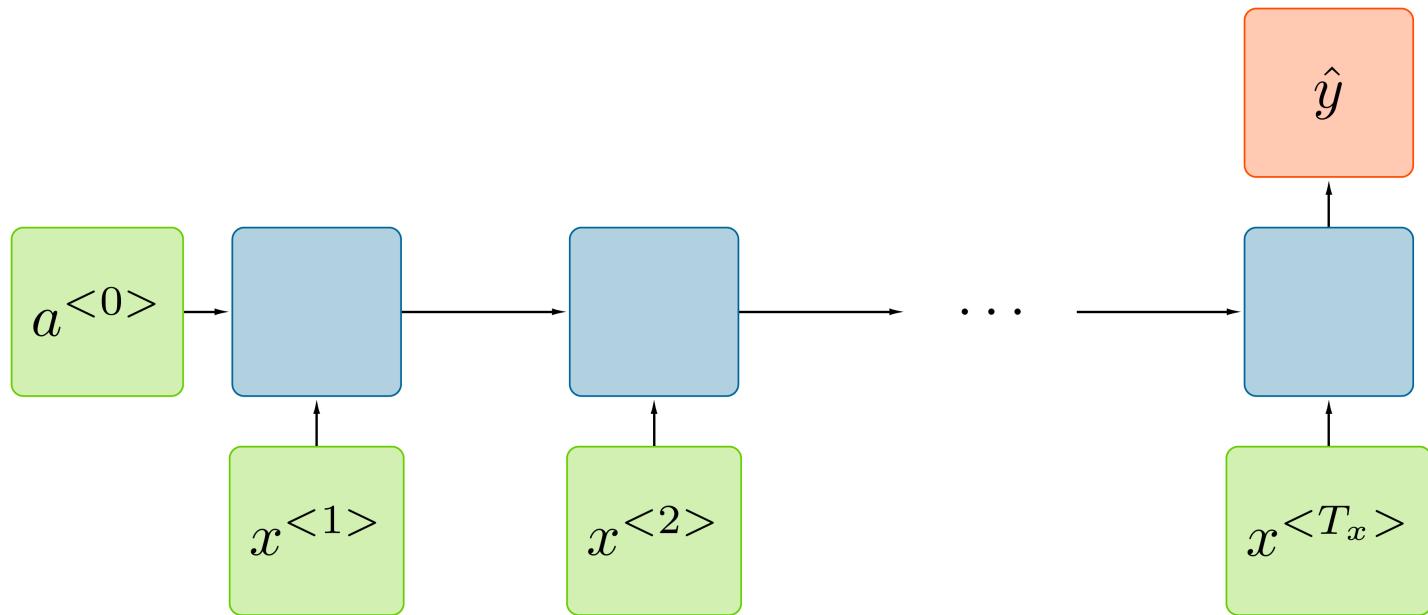
Example: Name entity recognition

Many-to-many: $T_x \neq T_y$



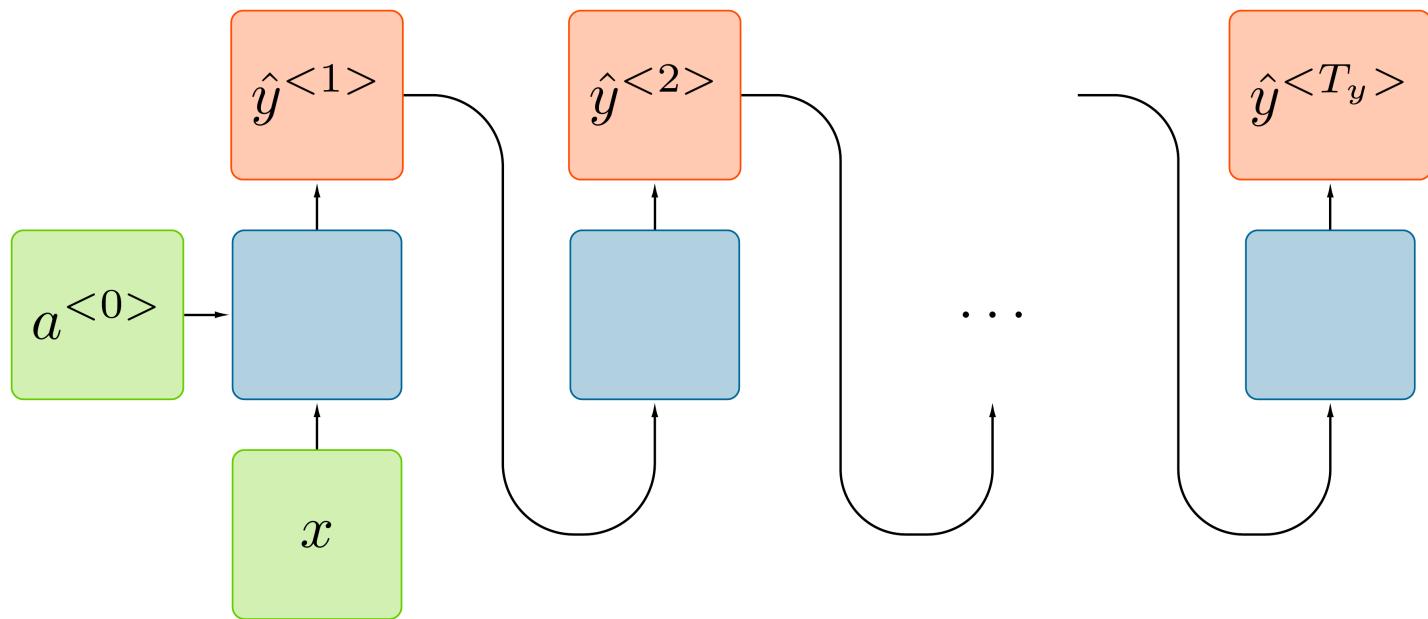
Example: Machine translation

Many-to-one: $T_x > 1$, $T_y = 1$



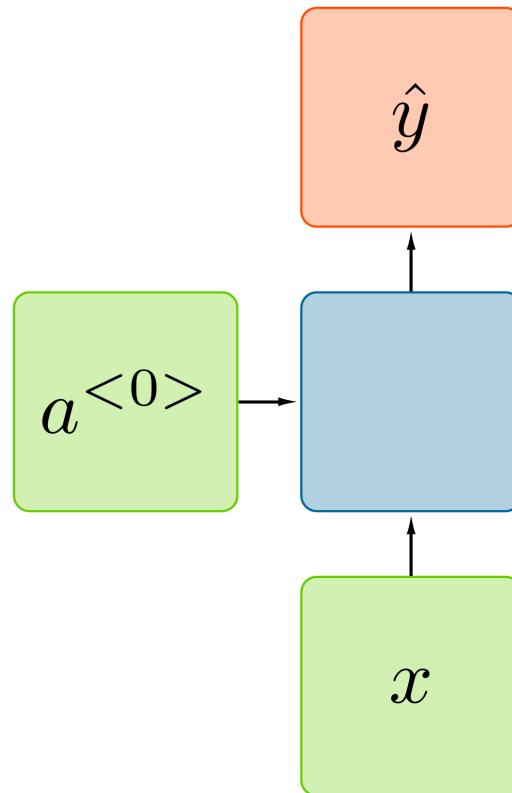
Example: Sentiment classification

One-to-many: $T_x = 1, T_y > 1$



Example: Music generation

One-to-one: $T_x = T_y = 1$



Example: Traditional neural network

Модель МОВИ

Що таке модель МОВИ?

Speech recognition

- The apple and pair salad.
- The apple and pear salad.

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

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

$$P(\text{sentence}) = ?$$

$$P(y^{(1)}, y^{(2)}, \dots, y^{(T_y)})$$

Мовне моделювання

- Training set: large corpus of text (english, ukrainian, ...)

Cats average 15 hours of sleep a day.

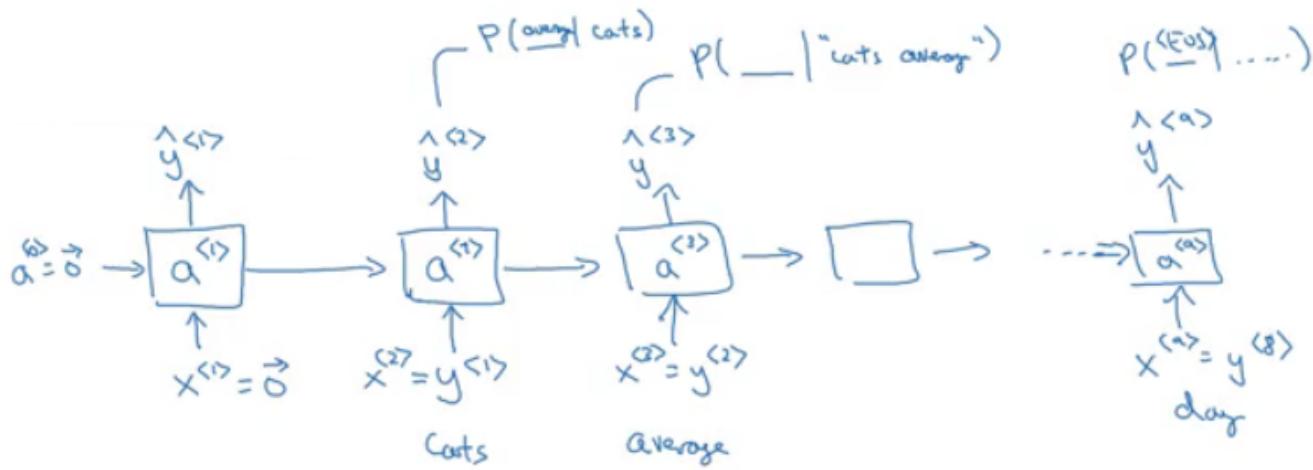
1. Токенізація

2. Для закінчення речень: $\langle \text{EOS} \rangle$

The Egyptian **Mau** is a breed of cat. $\langle \text{EOS} \rangle$

The Egyptian **UNK** is a breed of cat. $\langle \text{EOS} \rangle$

RNN модель



Cats average 15 hours of sleep a day. $\langle \text{EOS} \rangle$

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

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

$$P(y^{(1)}, y^{(2)}, y^{(3)}) = P(y^{(1)}) \cdot P(y^{(2)}|y^{(1)}) \cdot P(y^{(3)}|y^{(1)}, y^{(2)})$$

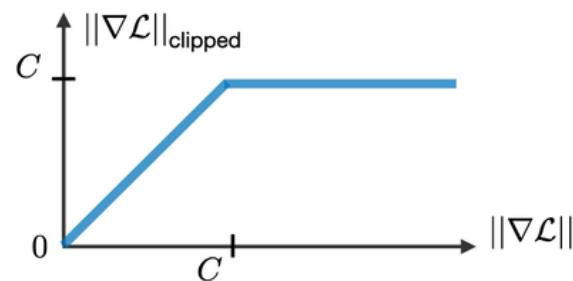
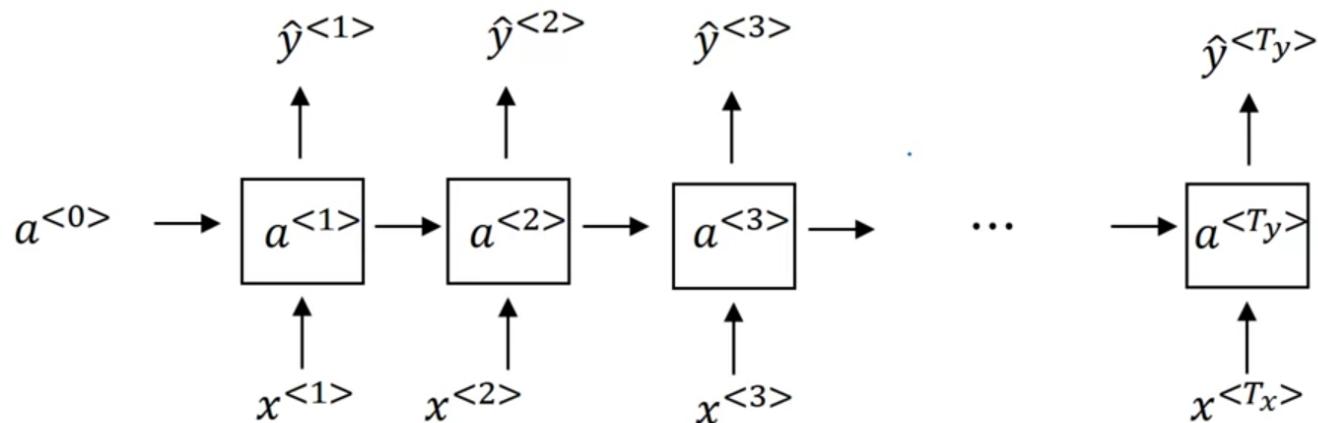
Sampling novel sequences

Проблема зникнення градієнтів в RNNs

RNN модель

The **cat**, which already ate a bunch of food **was** full.

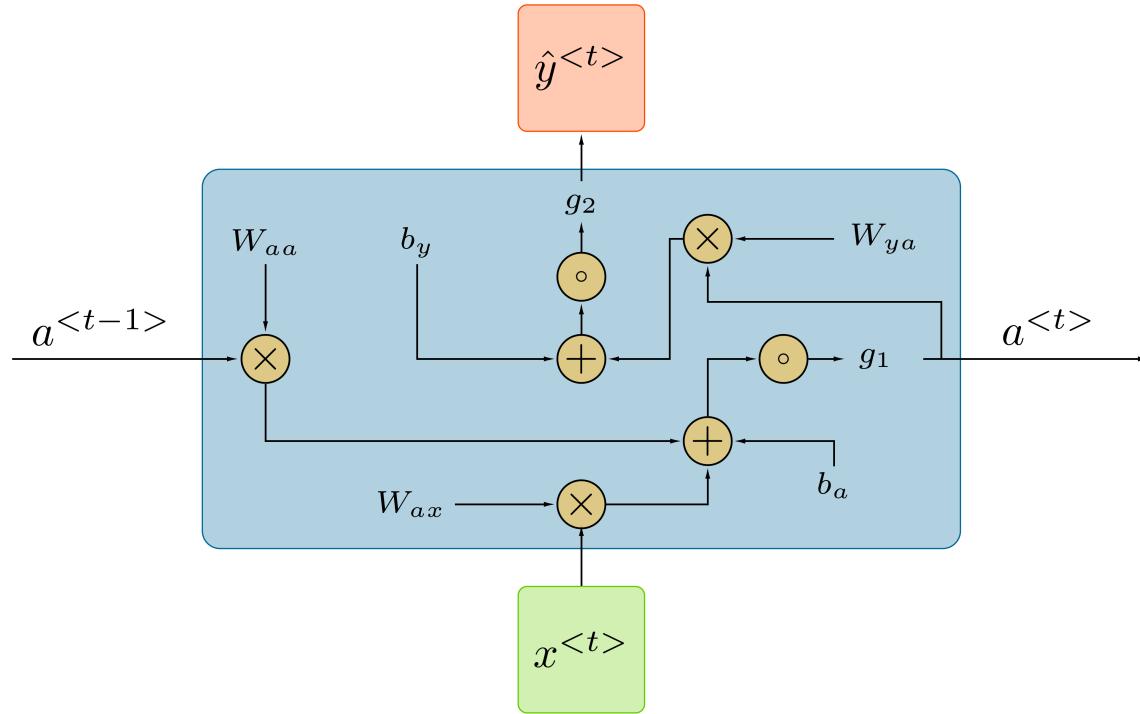
The **cats**, which already ate a bunch of food **were** full.



Gated Recurrent Unit (GRU)

K. Cho, B.v. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio. arXivreprint
arXiv:1406.1078. 2014. [[PDF](#)]

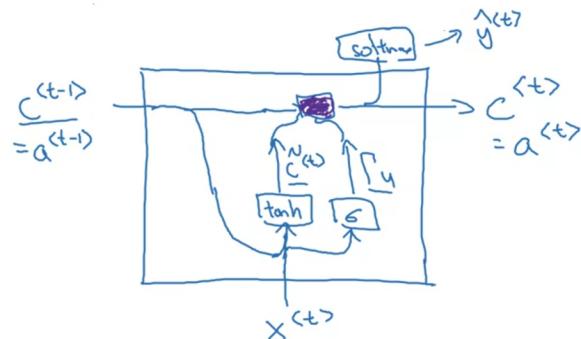
RNN блок



Для кожного часового кроку t активація $a^{<t>}$ і вихід $y^{<t>}$ виражаються таким чином:

$$\begin{aligned} a^{<t>} &= g_1(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a) = g_1(w_a[a^{<t-1>}, x^{<t>}] + b_a) \\ \hat{y}^{<t>} &= g_2(W_{ya}a^{<t>} + b_y) = g_2(w_ya^{<t>} + b_y) \end{aligned}$$

GRU



c = memory cell

$$c^{(t)} = a^{(t)}$$

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

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

$$c^{(t)} = \Gamma_u \cdot \tilde{c}^{(t)} + (1 - \Gamma_u) \cdot c^{(t-1)}$$

The **cat**, which already ate a bunch of food **was** full.

Full GRU

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

$$\Gamma_u = \sigma(W_u [c^{}, x^{}] + b_u)$$

$$\Gamma_r = \sigma(W_r [c^{}, x^{}] + b_r)$$

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

GRU vs LSTM

GRU

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

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

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

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

LSTM

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

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

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

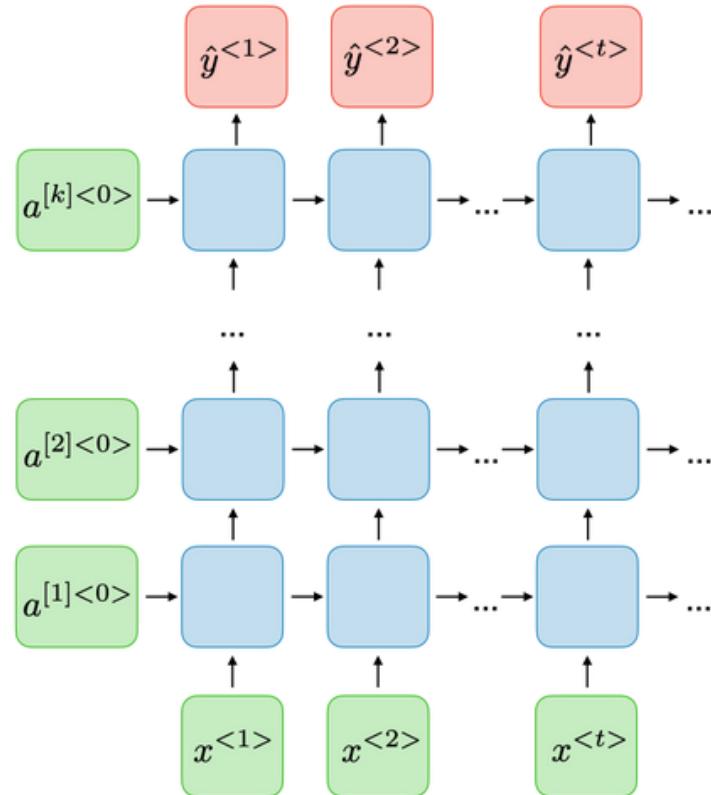
$$\Gamma_o = \sigma(W_o[a^{<t-1>}, x^{<t>}] + b_o)$$

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

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

Deep RNNs

Deep RNNs



Переваги та недоліки звичайних RNN

Переваги

- Можливість обробляти вхідну послідовністі будь-якої довжини
- Розмір моделі не збільшується з розміром вхідної послідовності
- Обчислення враховують історичну інформацію (взаємозв'язок між словами)
- Вага розподіляється в часі

Недоліки

- Ресурсозатратність обчислень
- Труднощі з доступом до інформації на попередніх часових кроках
- Нерозглядаються будь-які майбутні вхідні дані для поточного стану

Література

- Distill, Visualizing memorization in RNNs

Кінець