

Глубинное обучение

Лекция 12
Рекуррентные нейронные сети

Михаил Гущин

mhushchyn@hse.ru

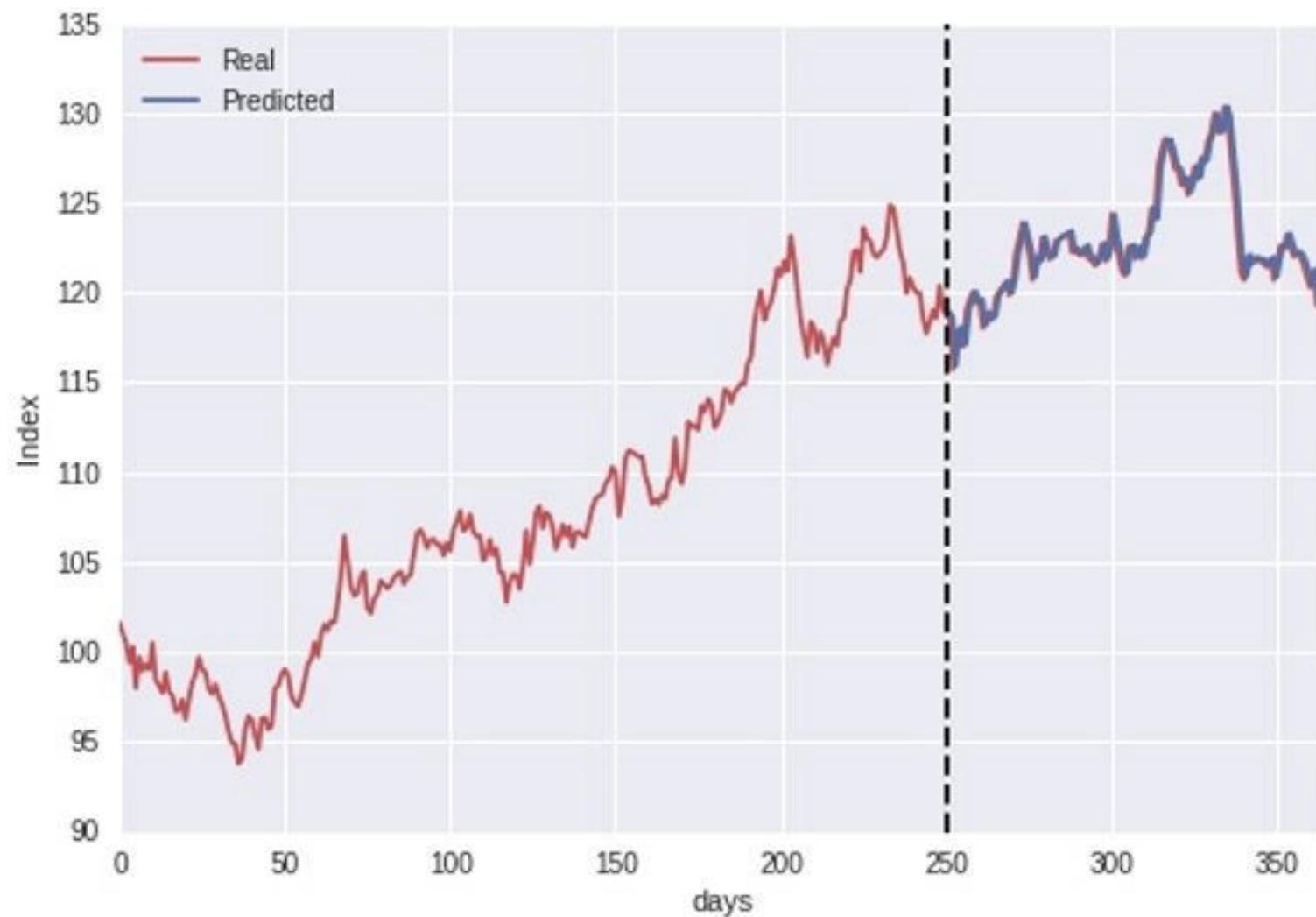
НИУ ВШЭ, 2024



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Анализ последовательностей

Прогноз временных рядов



Прогноз временных рядов

- ▶ Пусть дан временной ряд из Т наблюдений:

$$y_1, y_2, \dots, y_m, \dots, y_{T-1}, y_T$$

- ▶ Хотим делать прогноз на один шаг вперед.
- ▶ Autoregression (AR) model для прогноза:

$$\hat{y}_{t+1} = f(y_t, y_{t-1}, \dots, y_{t-k})$$

где $f()$ – любая модель регрессии: линейная регрессия, нейронная сеть и д.р.

Прогноз временных рядов

- ▶ Как бы вы обучали модель прогноза на один шаг вперед?
- ▶ Что такое вход модели x ?
- ▶ Что такое выход модели y ?
- ▶ Какую полносвязную сеть вы бы использовали?

Прогноз временных рядов

- ▶ Autoregression (AR) model для прогноза:

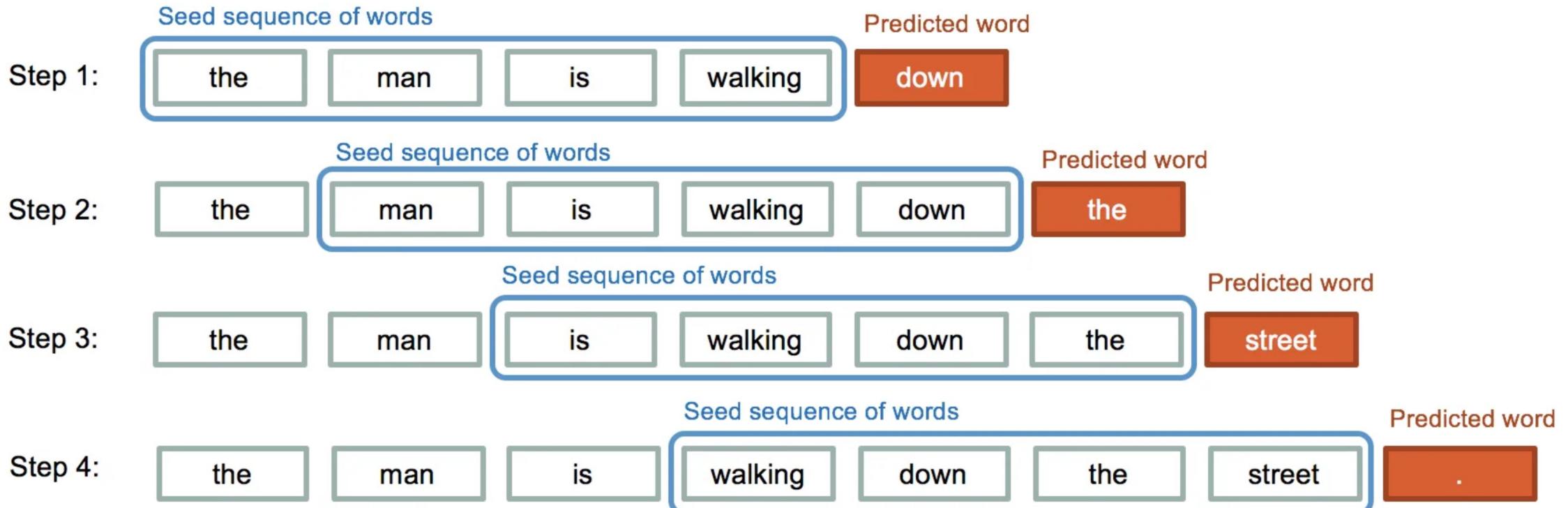
$$\hat{y}_{t+1} = f(y_t, y_{t-1}, \dots, y_{t-k})$$

- ▶ На языке машинного обучения модель можем записать как условную вероятность:

$$\hat{y}_{t+1} \sim P(y_{t+1}|y_t, y_{t-1}, \dots, y_{t-k})$$

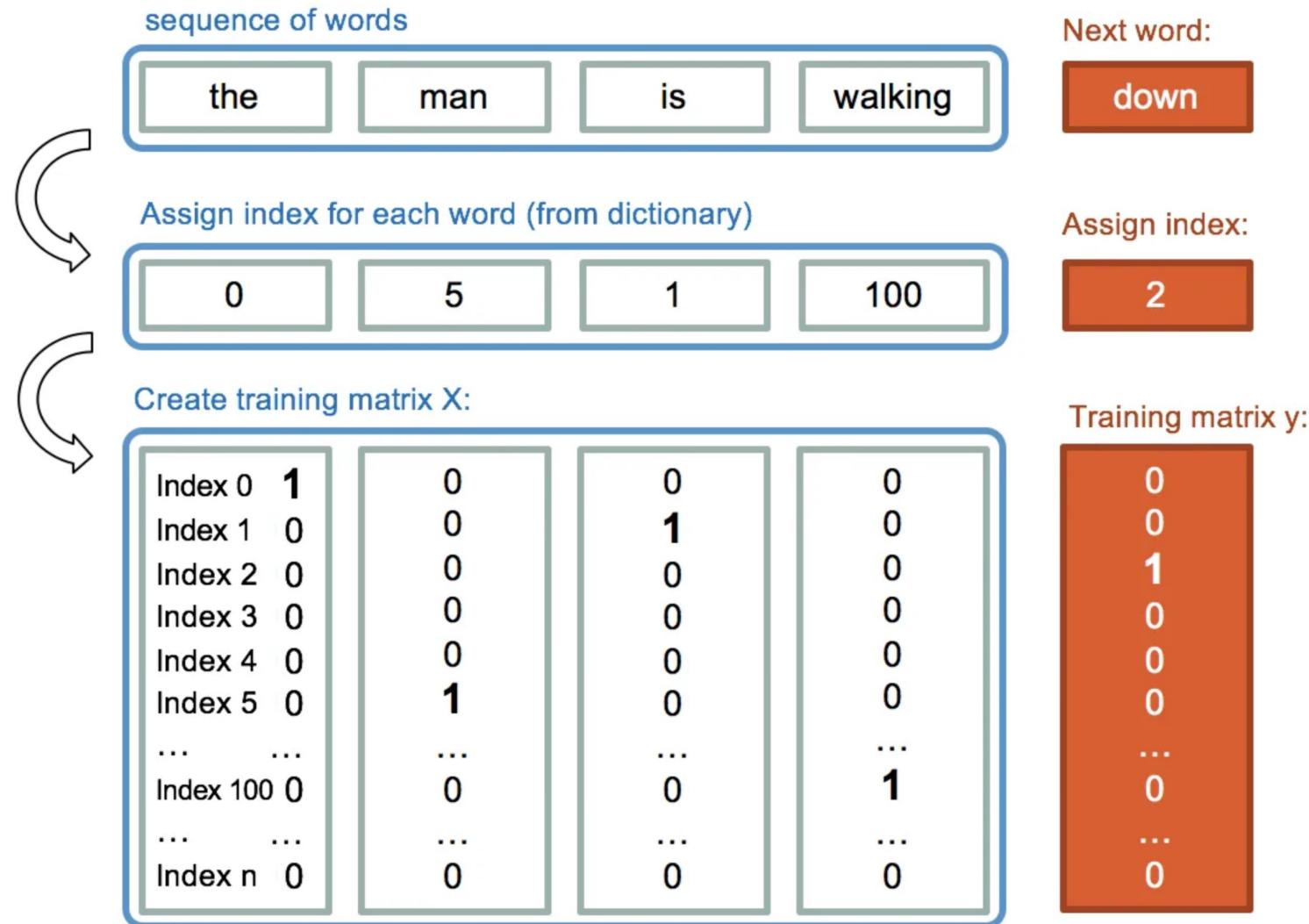
где вероятность P нужно оценить с помощью нейронный сетей

Генерация текста



Link: <https://bansalh944.medium.com/text-generation-using-lstm-b6ced8629b03>

Генерация текста



Link: <https://medium.com/@david.campion/text-generation-using-bidirectional-lstm-and-doc2vec-models-1-3-8979eb65cb3a>

Другие задачи

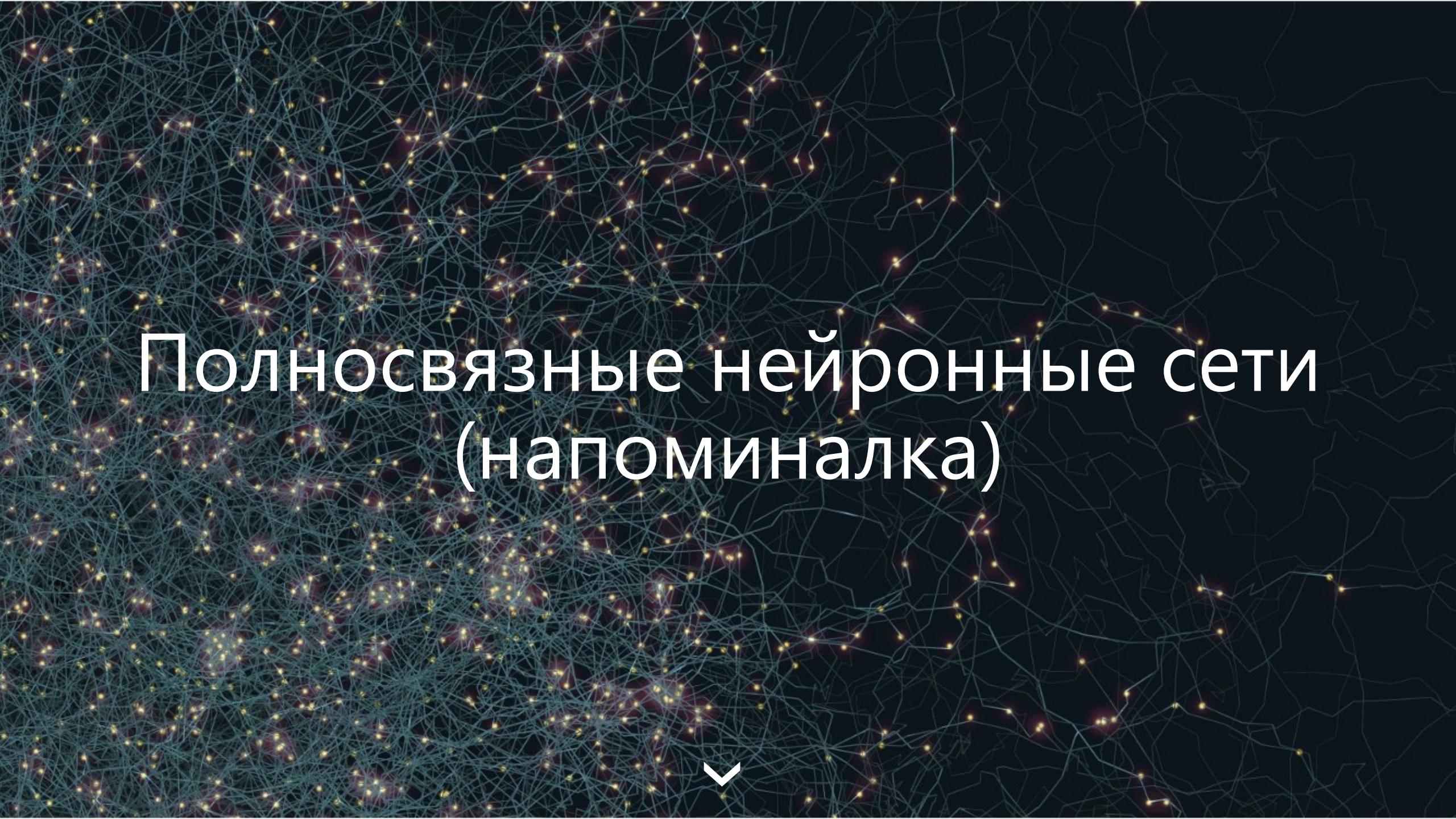
- ▶ Классификация временных рядов
 - По аналогии с прогнозом рядов
- ▶ Классификация текстов
 - По аналогии с генерацией текста
- ▶ Регрессии на текстах
 - По аналогии с генерацией текста
- ▶ Машинный перевод (на следующей лекции)

Мотивация

Недостатки полносвязных сетей для анализа последовательностей:

- ▶ Если в последовательности K векторов размерности D , то у полносвязной сети будет KD входов
- ▶ Потребуется больше нейронов слоях, чтобы выучит временные зависимости в данных
- ▶ Сеть будет обучаться дольше

Можно ли как-то улучшить нейронную сеть?

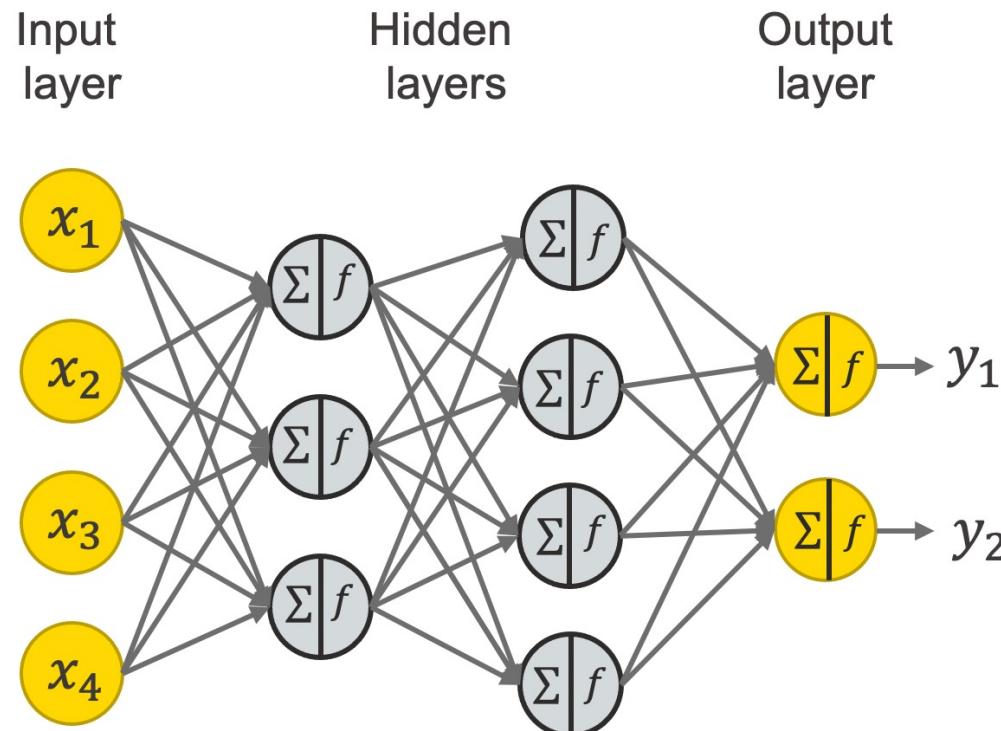


Полносвязные нейронные сети (напоминалка)



Нейронная сеть

- ▶ Пусть дан набор наблюдений $\{x_i, y_i\}_{i=1}^n$, где $x_i \in R^d$, $y_i \in R^q$
- ▶ Построим нейронную сеть



Нейронная сеть в матричной форме

- Матрица наблюдений $X \in R^{n \times d}$ из n объектов, каждый из которых имеет d входных признаков:

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}$$

- Число строк – число наблюдений (объектов)
- Число столбцов – число входных признаков

Нейронная сеть в матричной форме

$$H^{(1)} = f_1(XW^{(1)} + b^{(1)})$$

$$H^{(2)} = f_2(H^{(1)}W^{(2)} + b^{(2)})$$

$$O = f_3(H^{(2)}W^{(3)} + b^{(3)})$$

Размеры матриц:

- ▶ $X \in R^{n \times d}$, $W^{(1)} \in R^{d \times h}$, $b^{(1)} \in R^{1 \times h}$, h - число нейронов в первом слое
- ▶ $H^{(1)} \in R^{n \times h}$, $W^{(2)} \in R^{h \times m}$, $b^{(2)} \in R^{1 \times m}$, m - число нейронов во втором слое
- ▶ $H^{(2)} \in R^{n \times m}$, $W^{(3)} \in R^{m \times q}$, $b^{(3)} \in R^{1 \times q}$, q - число выходов сети
- ▶ $O \in R^{n \times q}$ - матрица прогнозов сети

Полносвязный слой

$$H^{(1)} = f_1(XW^{(1)} + b^{(1)})$$

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}, \quad W^{(1)} = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1h} \\ w_{21} & w_{22} & \cdots & w_{2h} \\ \vdots & \vdots & \ddots & \vdots \\ w_{d1} & w_{d2} & \cdots & w_{dh} \end{pmatrix}, \quad H^{(1)} = \begin{pmatrix} h_{11} & h_{12} & \cdots & h_{1h} \\ h_{21} & h_{22} & \cdots & h_{2h} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n1} & h_{n2} & \cdots & h_{nh} \end{pmatrix}$$

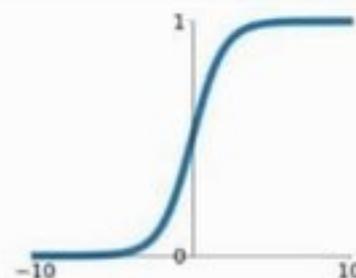
Размеры матриц:

- $X \in R^{n \times d}, W^{(1)} \in R^{d \times h}, b^{(1)} \in R^{1 \times h}, h$ - число нейронов в первом слое
- $H^{(1)} \in R^{n \times h}$ - выход слоя
- $f_1(z)$ - **функция активации**

ФУНКЦИИ АКТИВАЦИИ f

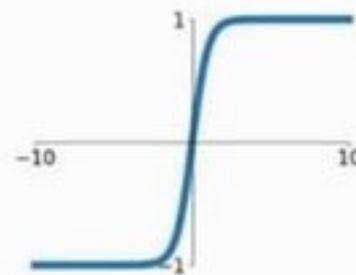
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



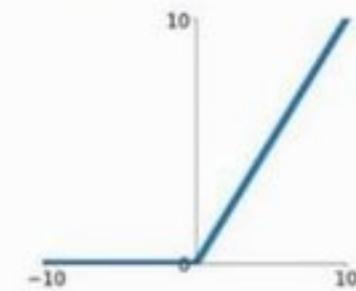
tanh

$$\tanh(x)$$



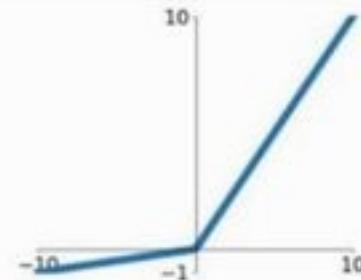
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

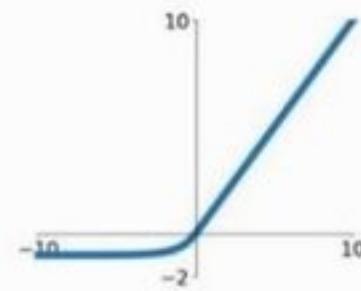


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Рекуррентные нейронные сети

(Recurrent Neural Networks)

Задача

- ▶ Пусть дано n последовательностей из T наблюдений:

$$x_1^1, x_2^1, \dots, x_t^1, \dots, x_{T-1}^1, x_T^1$$

$$x_1^2, x_2^2, \dots, x_t^2, \dots, x_{T-1}^2, x_T^2$$

...

$$x_1^n, x_2^n, \dots, x_t^n, \dots, x_{T-1}^n, x_T^n$$

где $x_t^i \in R^d$

- ▶ Хотим делать прогноз на один шаг вперед:

$$\hat{x}_{T+1}^i = f(x_1^i, x_2^i, \dots, x_t^i, \dots, x_{T-1}^i, x_T^i)$$

Задача

- ▶ Матрица наблюдений $X_t \in R^{n \times d}$ из n объектов в момент времени t , каждый из которых имеет d входных признаков:

$$X_t = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}$$

- ▶ Число строк – число наблюдений (объектов) в момент времени t
- ▶ Число столбцов – число входных признаков

Рекуррентный нейронный слой

$$H_t = f(X_t W_{xh} + H_{t-1} W_{hh} + b_h)$$

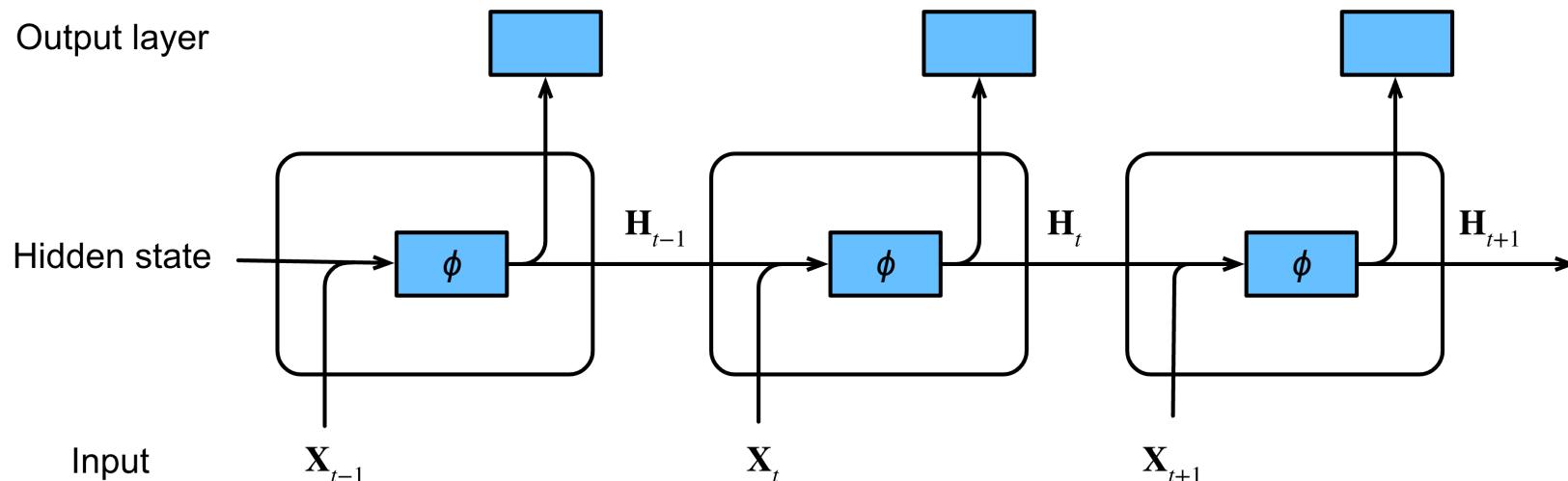
$$O_t = f_o(H_t W_{ho} + b_o)$$

- ▶ $X_t \in R^{n \times d}$ - матрица из n наблюдений в момент времени t
- ▶ $H_t \in R^{n \times h}$ - матрица из n скрытых состояний слоя
- ▶ $W_{xh} \in R^{d \times h}, W_{hh} \in R^{h \times h}, W_{ho} \in R^{h \times o}, h$ - число нейронов в слое
- ▶ $b_h \in R^{1 \times h}, b_o \in R^{1 \times o}$

Рекуррентный нейронный слой

$$H_t = f(X_t W_{xh} + H_{t-1} W_{hh} + b_h)$$

$$O_t = f_o(H_t W_{ho} + b_o)$$



FC layer with
activation function



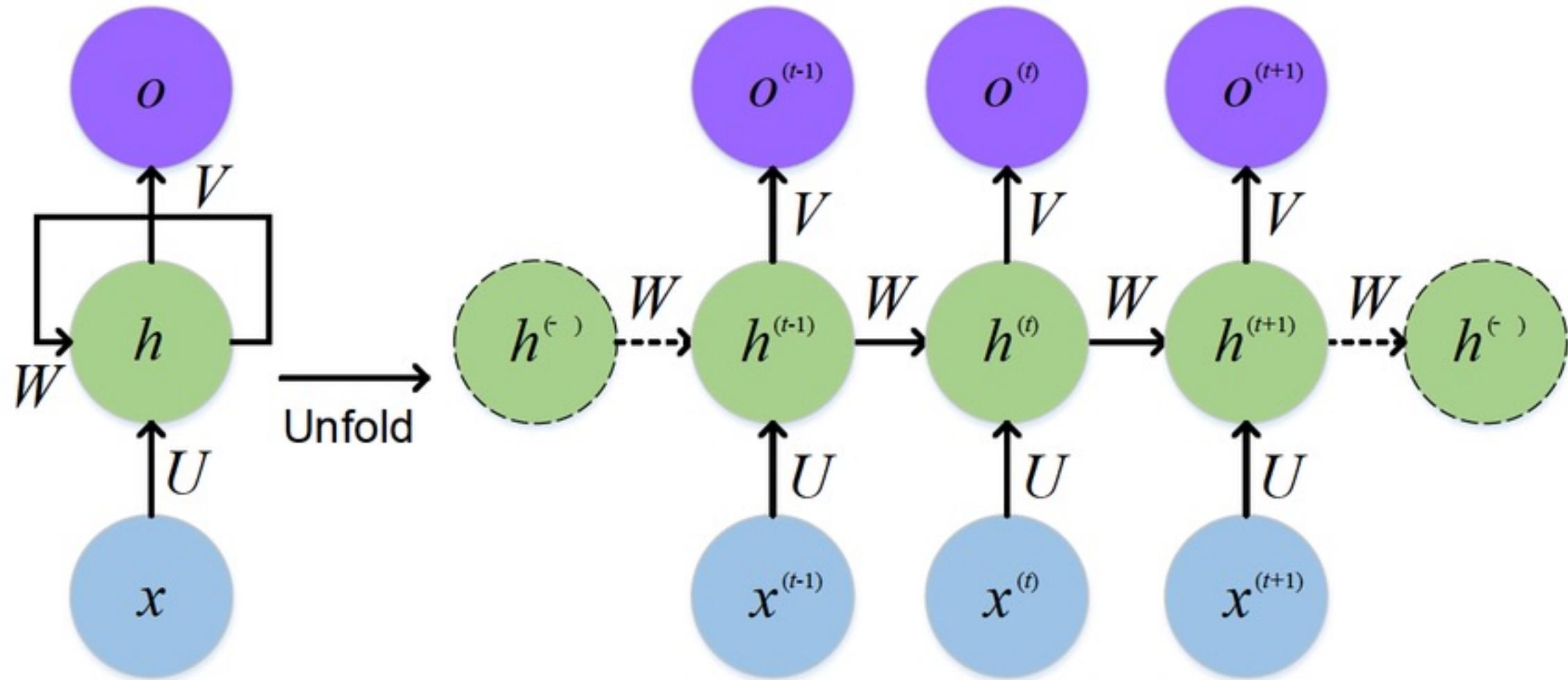
Copy



Concatenate

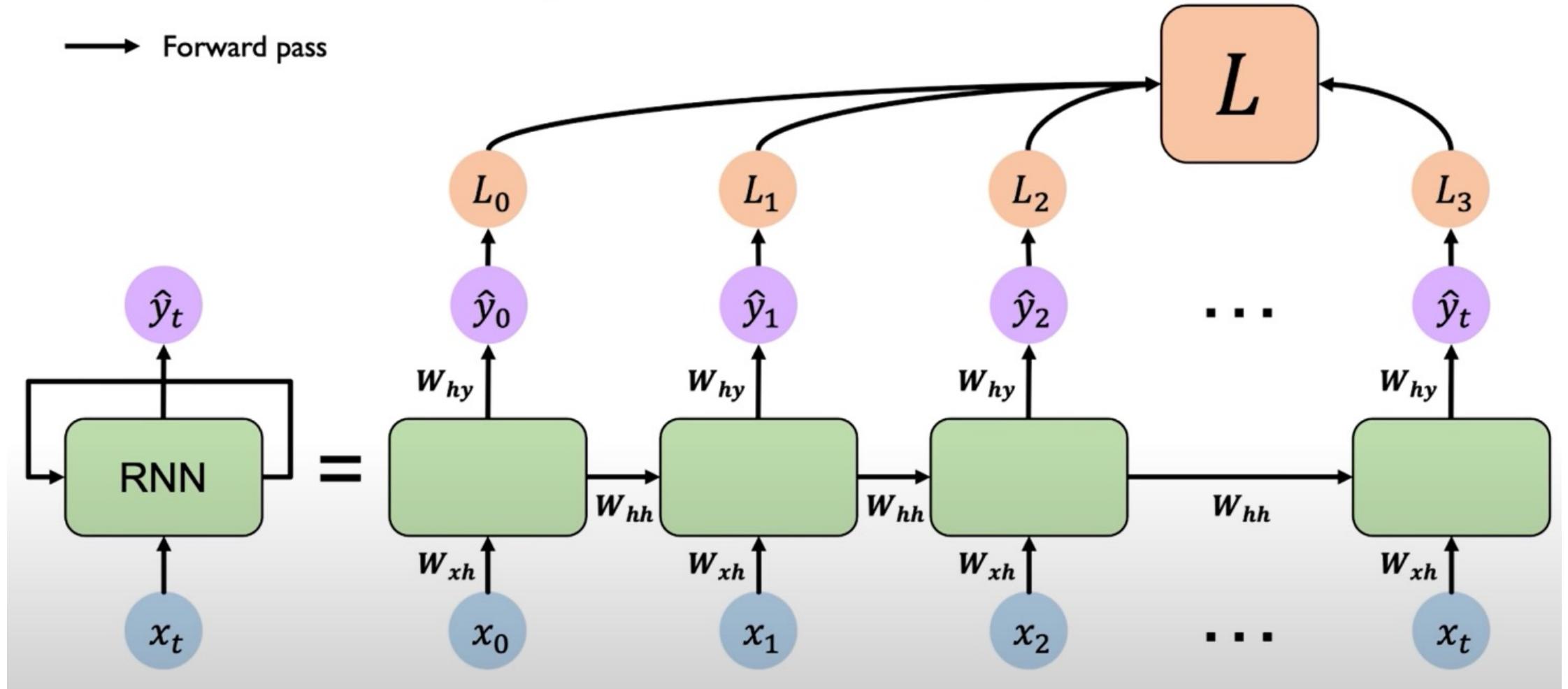
Link: https://d2l.ai/chapter_recurrent-modern/lstm.html

Рекуррентный нейронный слой



Обучение RNN

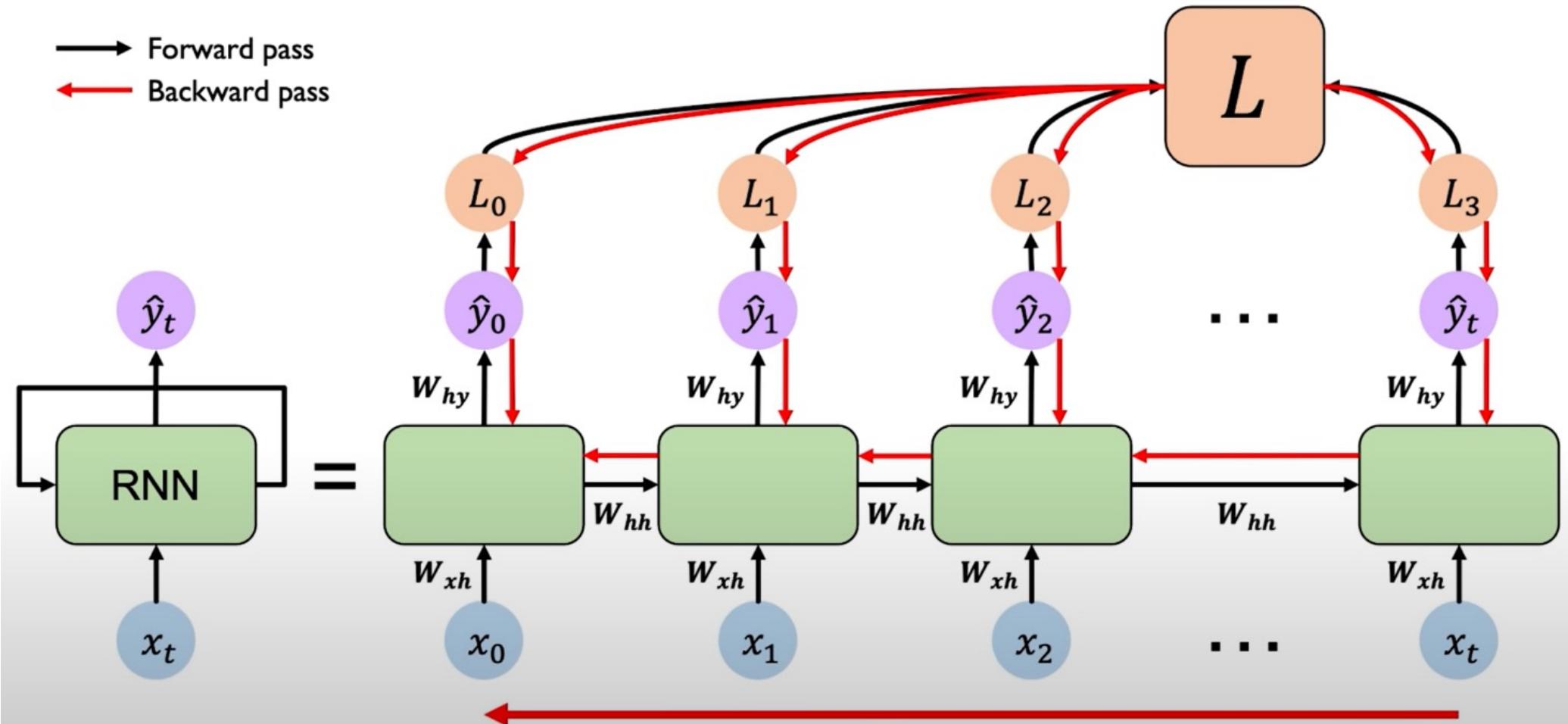
Forward pass



Link: https://authurwhywait.github.io/blog/2021/12/02/introduction_to_dl02/

Backward pass для подсчета градиента

RNNs: Backpropagation Through Time

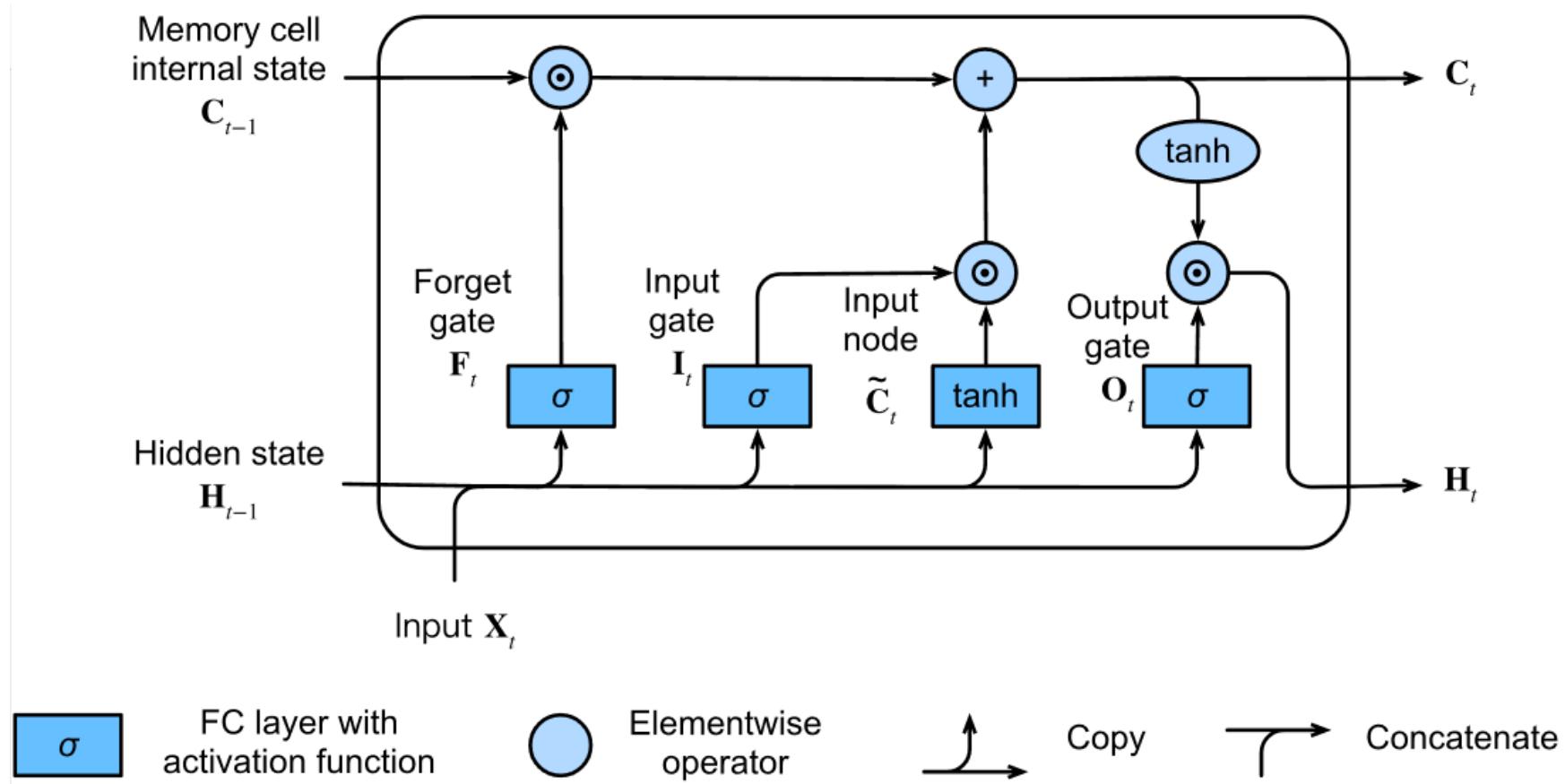


Проблемы с градиентами

- ▶ Информация теряется по мере прохождения во времени
- ▶ Не факт, что сеть сможет выучить зависимость финального вектора от первых векторов

Long Short-Term Memory (LSTM)

LSTM



LSTM

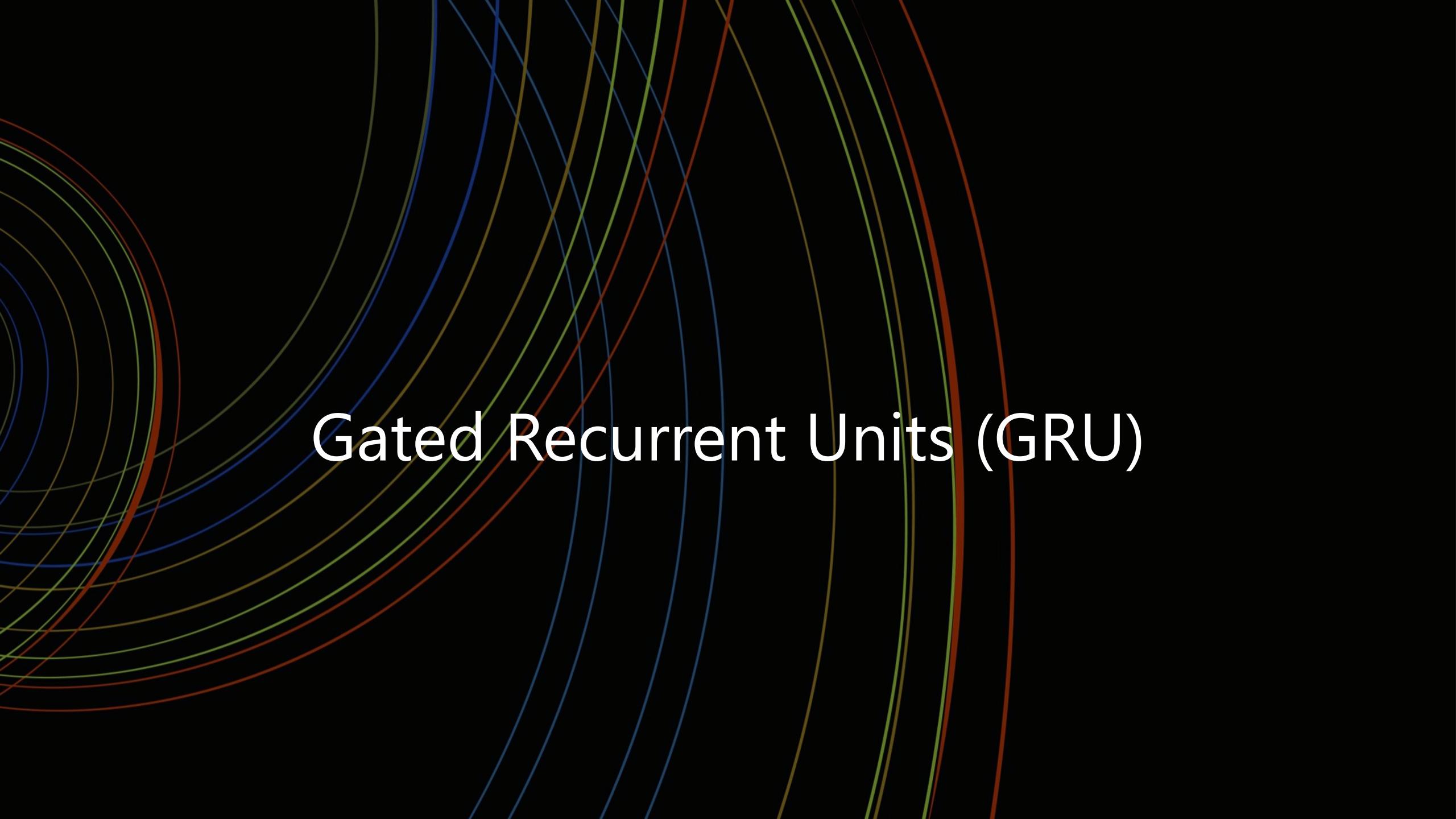
$$\begin{aligned}I_t &= \sigma(X_t W_{xi} + H_{t-1} W_{hi} + b_i) \\F_t &= \sigma(X_t W_{xf} + H_{t-1} W_{hf} + b_f) \\O_t &= \sigma(X_t W_{xo} + H_{t-1} W_{ho} + b_o)\end{aligned}$$

$$\begin{aligned}\tilde{C}_t &= \tanh(X_t W_{xc} + H_{t-1} W_{hc} + b_c) \\C_t &= F_t \odot C_{t-1} + I_t \odot \tilde{C}_t\end{aligned}$$

$$H_t = O_t \odot \tanh(C_t)$$

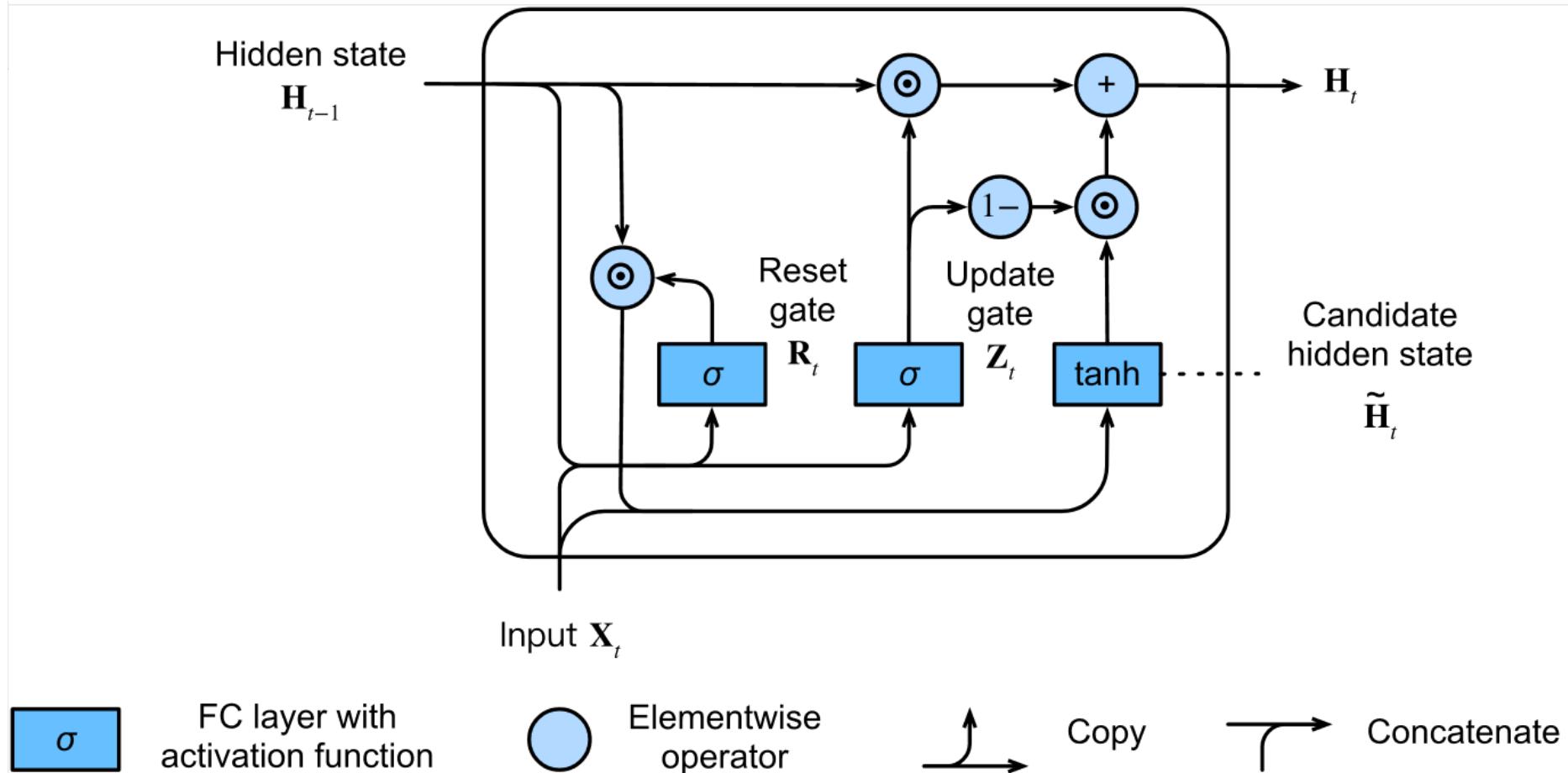
LSTM

- ▶ Позволяет предыдущему состоянию перейти в текущее без домножений на матрицы
- ▶ Модель сможет «протаскивать» информацию из начала текста в конец



Gated Recurrent Units (GRU)

GRU



GRU

$$R_t = \sigma(X_t W_{xr} + H_{t-1} W_{hr} + b_r)$$

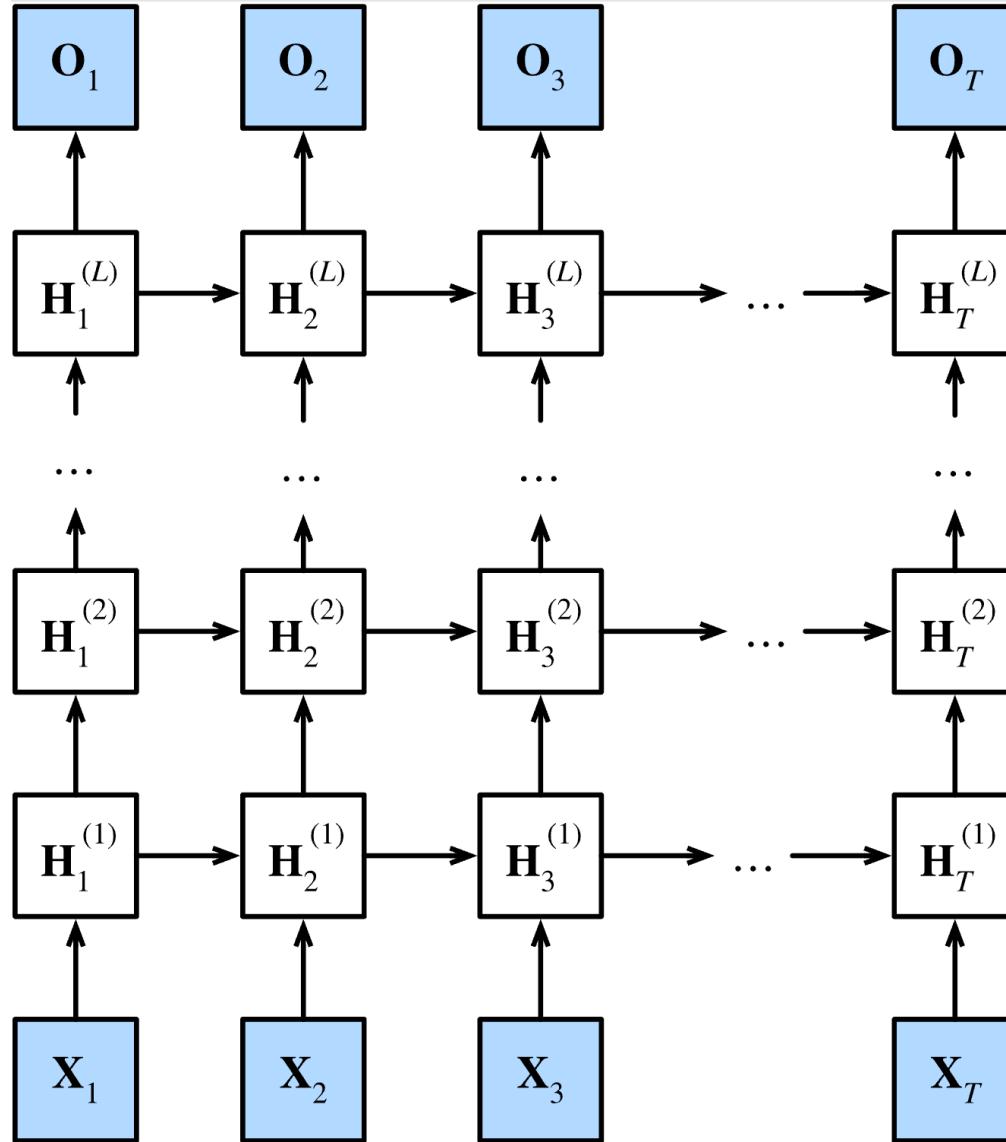
$$Z_t = \sigma(X_t W_{xz} + H_{t-1} W_{hz} + b_z)$$

$$\tilde{H}_t = \tanh(X_t W_{xh} + (R_t \odot H_{t-1}) W_{hh} + b_h)$$

$$H_t = Z_t \odot H_{t-1} + (1 - Z_t) \odot \tilde{H}_t$$

Глубокие рекуррентные сети

Глубокие RNN



Глубокие RNN

$$H_t^1 = f(X_t W_{xh}^1 + H_{t-1}^1 W_{hh}^1 + b_h^1)$$

$$H_t^2 = f(H_t^1 W_{xh}^2 + H_{t-1}^2 W_{hh}^2 + b_h^2)$$

$$H_t^3 = f(H_t^2 W_{xh}^3 + H_{t-1}^3 W_{hh}^3 + b_h^3)$$

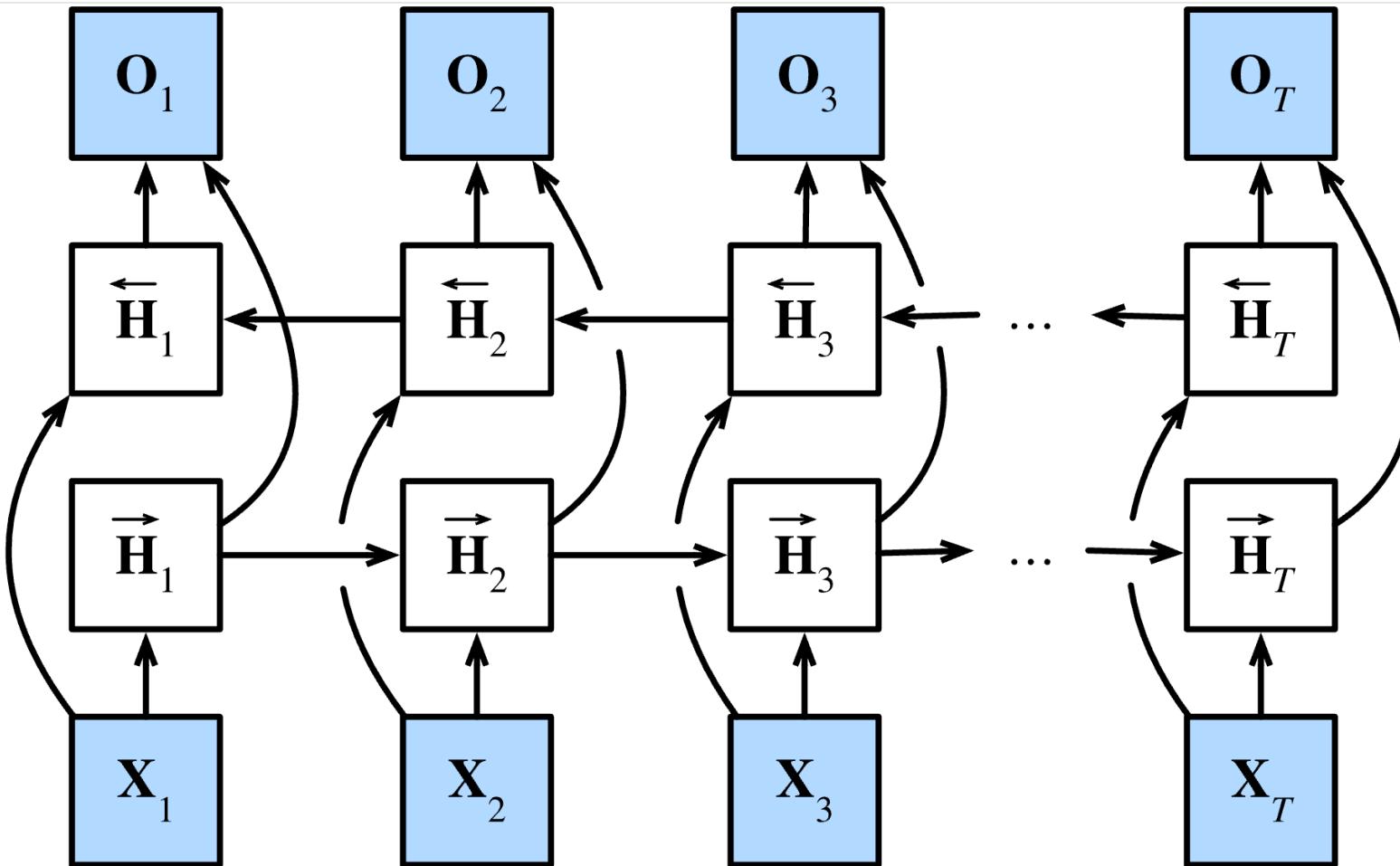
...

$$H_t^L = f(H_t^{L-1} W_{xh}^L + H_{t-1}^L W_{hh}^L + b_h^L)$$

$$O_t = f_o(H_t^L W_{ho} + b_o)$$

Двунаправленные RNN

Двунаправленные (Bidirectional) RNN



Link: https://d2l.ai/chapter_recurrent-modern/bi-rnn.html

Двунаправленные (Bidirectional) RNN

$$H_t^f = f(X_t W_{xh}^f + H_{t-1}^f W_{hh}^f + b_h^f)$$

$$H_t^b = f(X_t W_{xh}^b + H_{t-1}^b W_{hh}^b + b_h^b)$$

$$H_t = concat(H_t^f, H_t^b)$$

$$O_t = f_o(H_t W_{ho} + b_o)$$