

# Анализ временных рядов

---

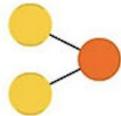
Азиз Темирханов

# План лекции

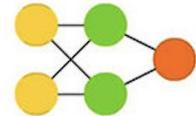
- Введение
- Вспоминаем про FF и RNN\LSTM
- Специализированные сетки

# Введение. Таксономия нейросетей

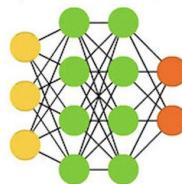
## Perceptron (P)



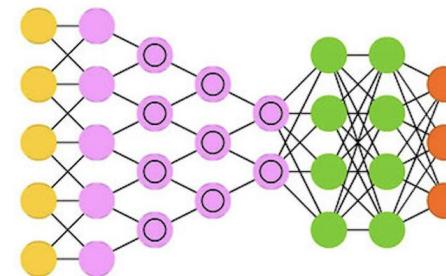
## Feed Forward (FF)



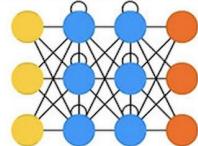
## Deep Feed Forward (DFF)



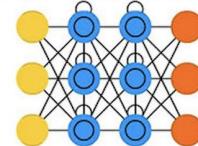
## Deep Convolutional Network (DCN)



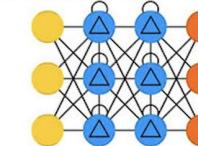
## Recurrent Neural Network (RNN)



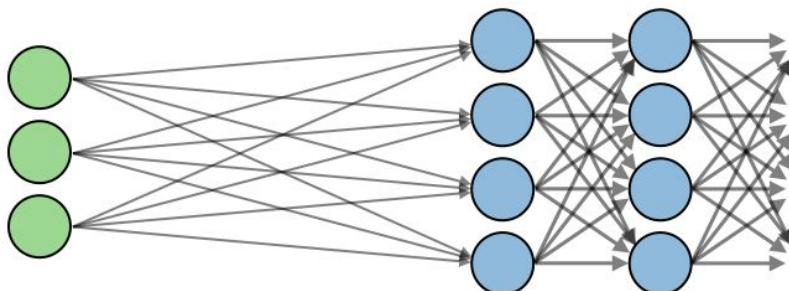
## Long / Short Term Memory (LSTM) Gated



) Gated Recurrent Unit (GRU)

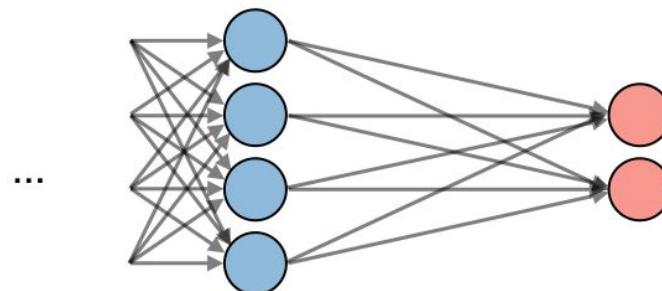


# Введение. Таксономия нейросетей



Input layer

Hidden layer 1

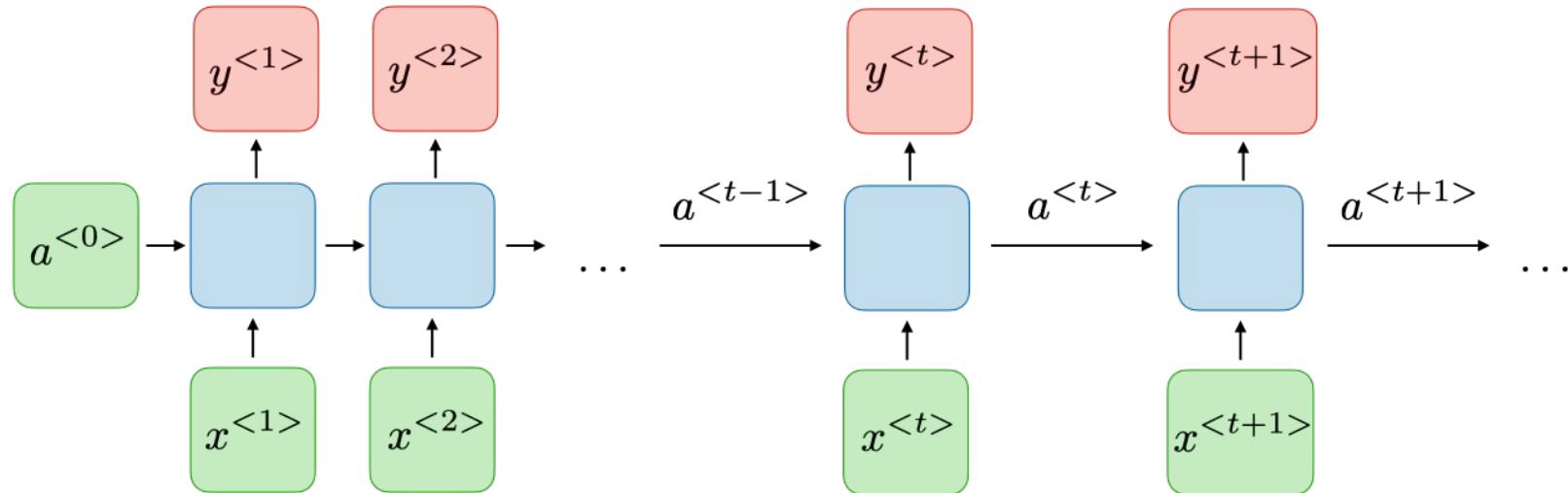


...

Hidden layer  $k$

Output layer

# Введение. Таксономия нейросетей

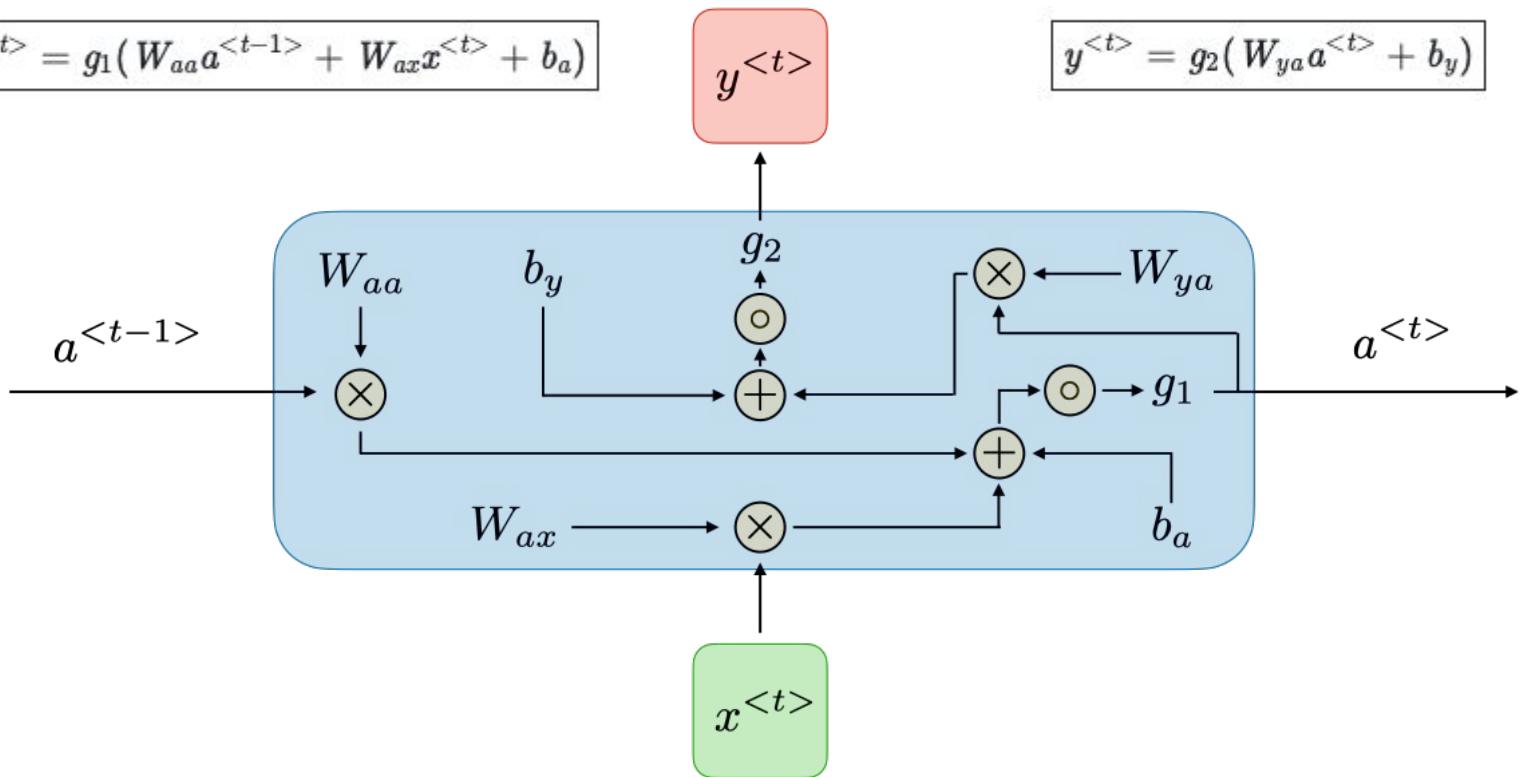


# Введение. Таксономия нейросетей. RNN

$$a^{} = g_1(W_{aa}a^{} + W_{ax}x^{} + b_a)$$

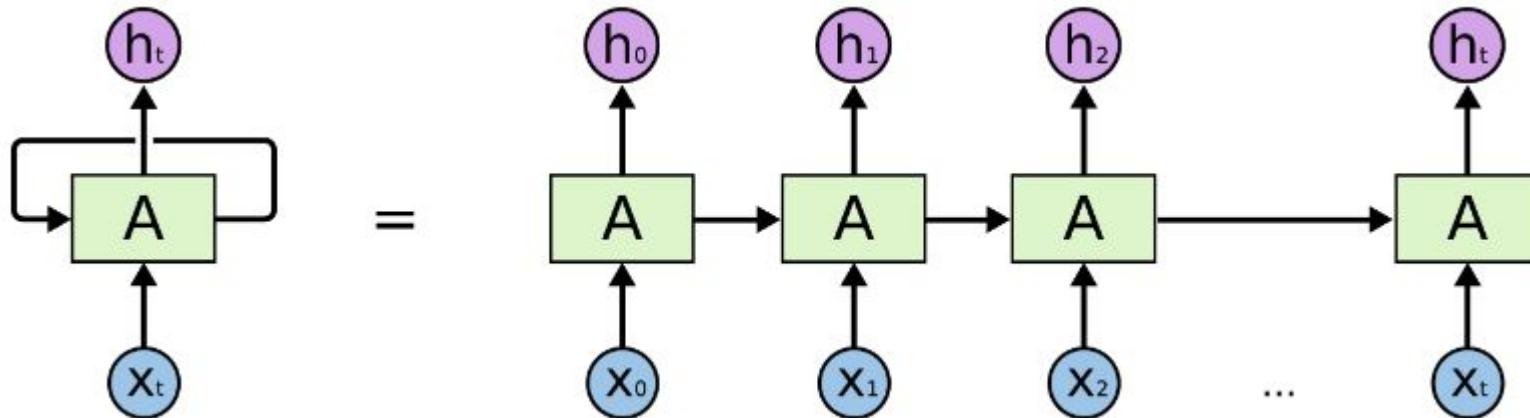
$$y^{}$$

$$y^{} = g_2(W_{ya}a^{} + b_y)$$



# Backpropagation through time

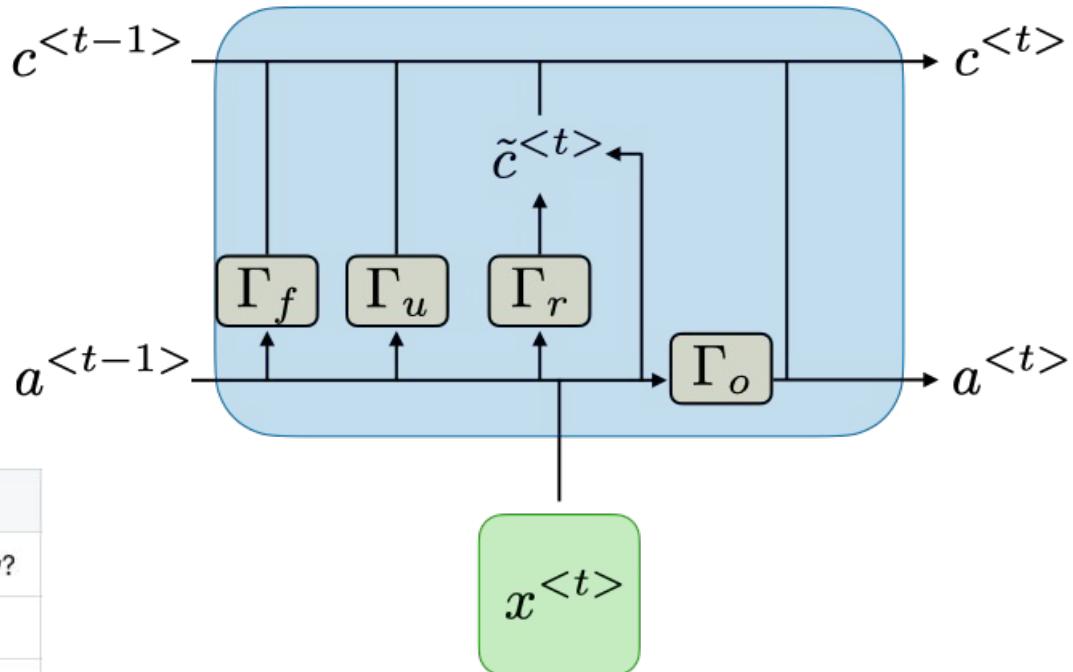
- Для обучения RNN (и LSTM тоже) используют ВТТ
- Сеть “разворачивают” и обучаю как одну большую полно связную сетку
- Такая модель принимает весь временной ряд как вход



# Введение. Таксономия нейросетей. LSTM

$$\begin{aligned}\tilde{c}^{} &= \tanh(W_c[\Gamma_r \star a^{}, x^{}] + b_c) \\ c^{} &= \Gamma_u \star \tilde{c}^{} + \Gamma_f \star c^{} \\ a^{} &= \Gamma_o \star c^{}\end{aligned}$$

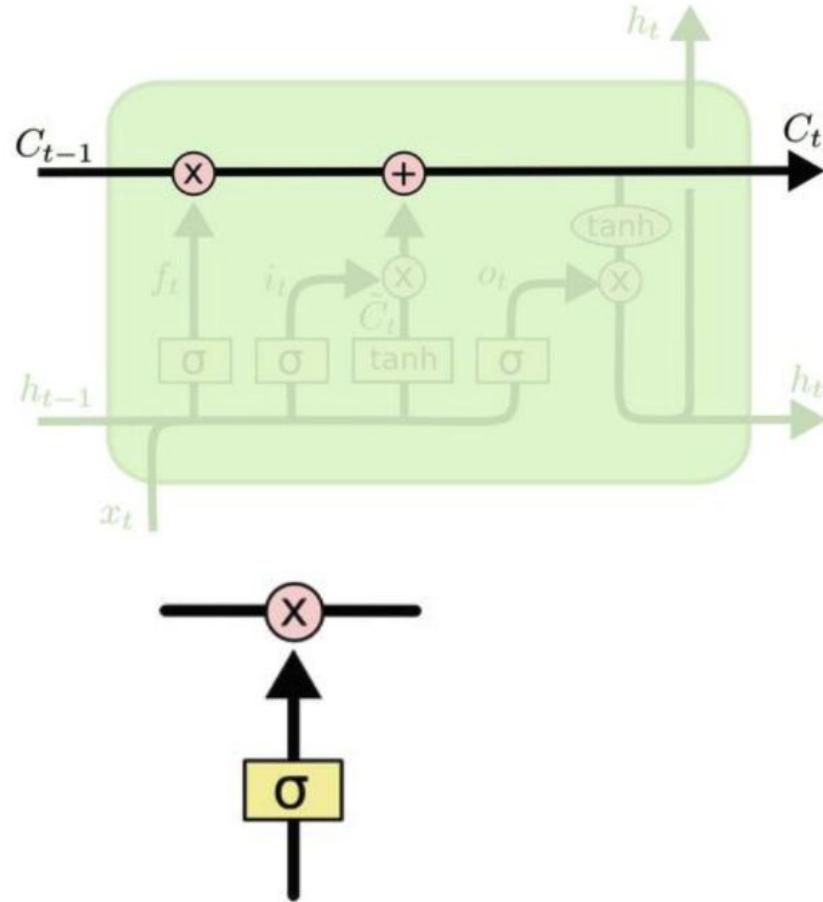
$$\Gamma = \sigma(Wx^{} + Ua^{} + b)$$



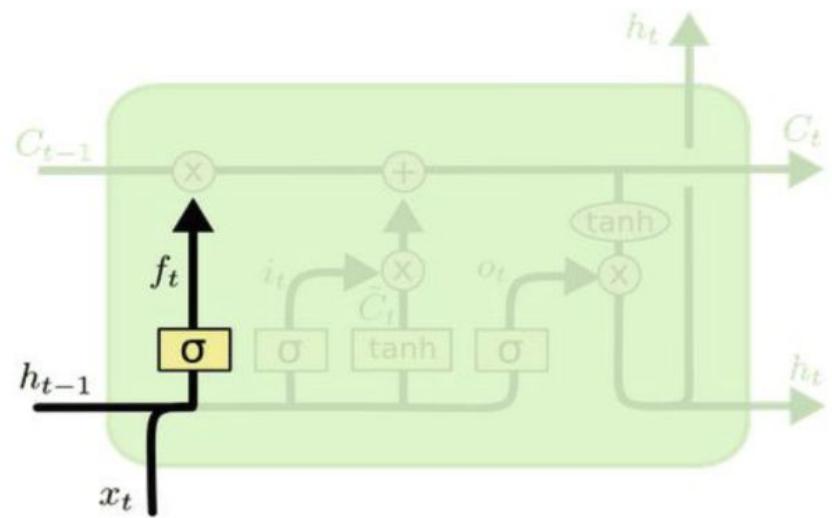
| Type of gate              | Role                             |
|---------------------------|----------------------------------|
| Update gate $\Gamma_u$    | How much past should matter now? |
| Relevance gate $\Gamma_r$ | Drop previous information?       |
| Forget gate $\Gamma_f$    | Erase a cell or not?             |
| Output gate $\Gamma_o$    | How much to reveal of a cell?    |

# LSTM

- Конвейер — главный компонент LSTM. Здесь находится “внутреннее” состояние модели (вместо  $h$ )
- Еще одно изменение — ворота (gates)

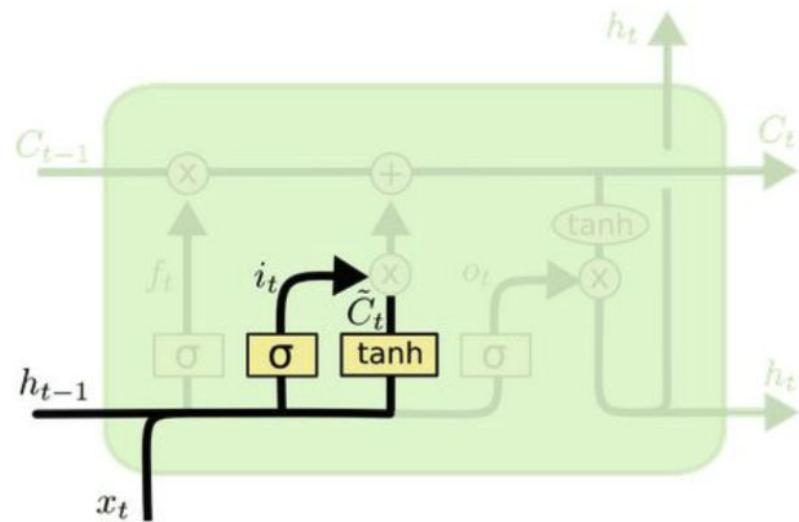


# LSTM. Forget Gate



$$f_t = \sigma (W_f \cdot [h_{t-1}, x_t] + b_f)$$

# LSTM. Relevance Gate



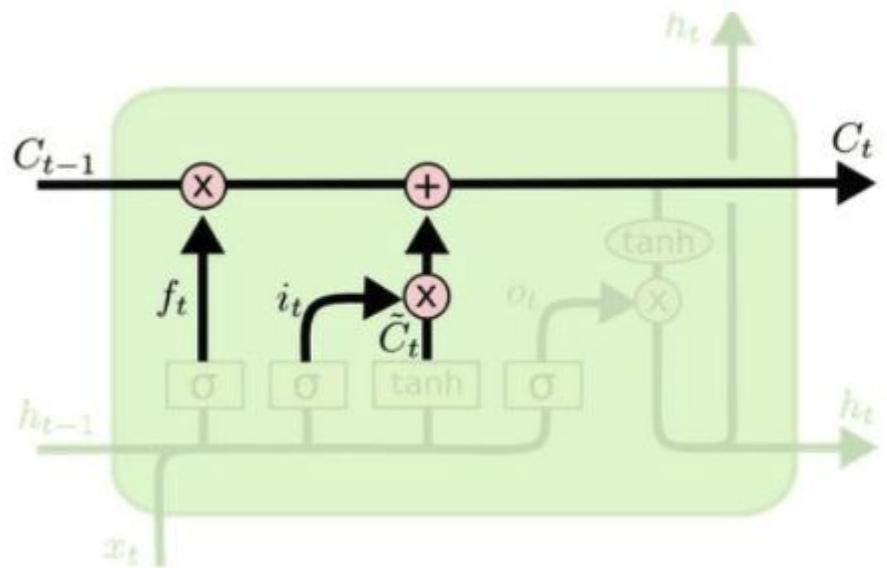
*input gate layer*

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

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

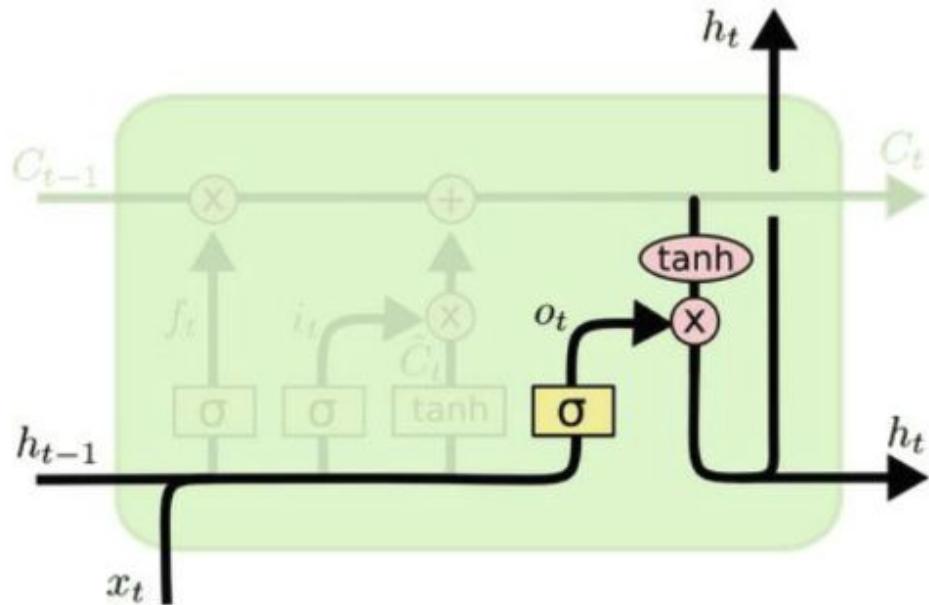
*new candidates vector*

# LSTM. Update Gate



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

# LSTM. Output Gate



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

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

# Специализированные модели

---

## DLinear (2023, 4319 citations)

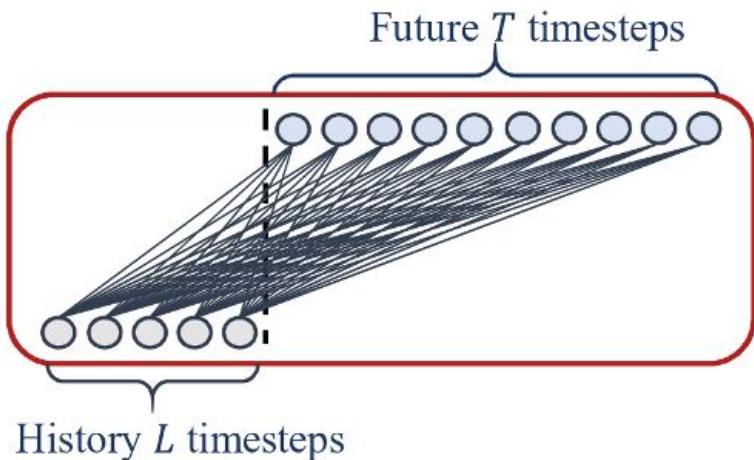
Идея: использовать простые линейные модели

$$\hat{X}_i = WX_i \quad W \in \mathbb{R}^{T \times L}$$

Предобработка:

- Декомпозиционный модуль из [Autoformer](#)
- Извлечение тренда скользящим средним

$$\begin{aligned} \mathcal{X}_t &= \text{AvgPool}(\text{Padding}(\mathcal{X})) \\ \mathcal{X}_s &= \mathcal{X} - \mathcal{X}_t, \end{aligned} \quad \Leftrightarrow \text{отдельные модели}$$



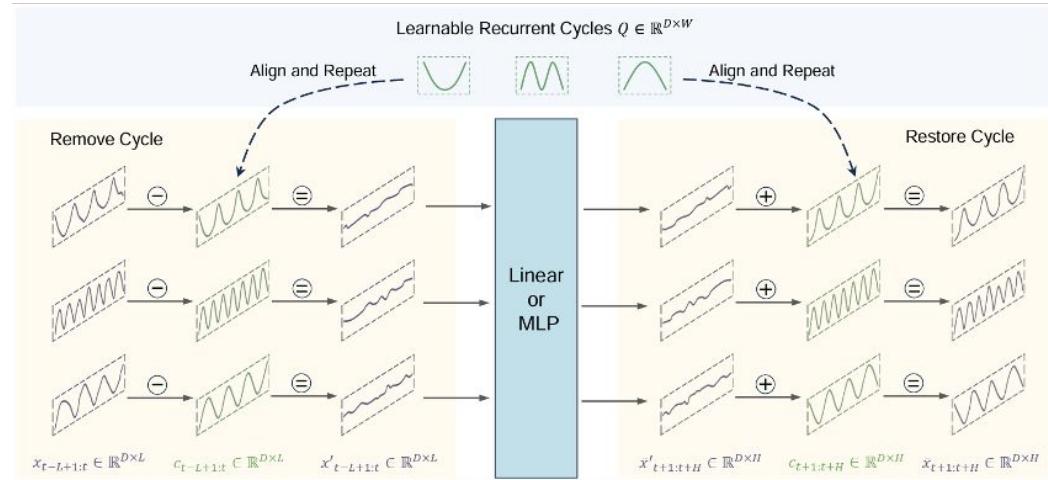
Код: <https://github.com/vivva/DLinear>  
Статья: <https://arxiv.org/abs/2205.13504>

# CycleNet (2024, 106 citations)

**Идея:** inductive bias для периодических паттернов.

**Архитектура:**

- Извлечение циклической компоненты обучаемыми шаблонами
- Моделирование остатков линейной моделью или MLP
- Возвращение циклической части



Code: <https://github.com/ACAT-SCUT/CycleNet>

Paper: <https://arxiv.org/abs/2409.18479>