

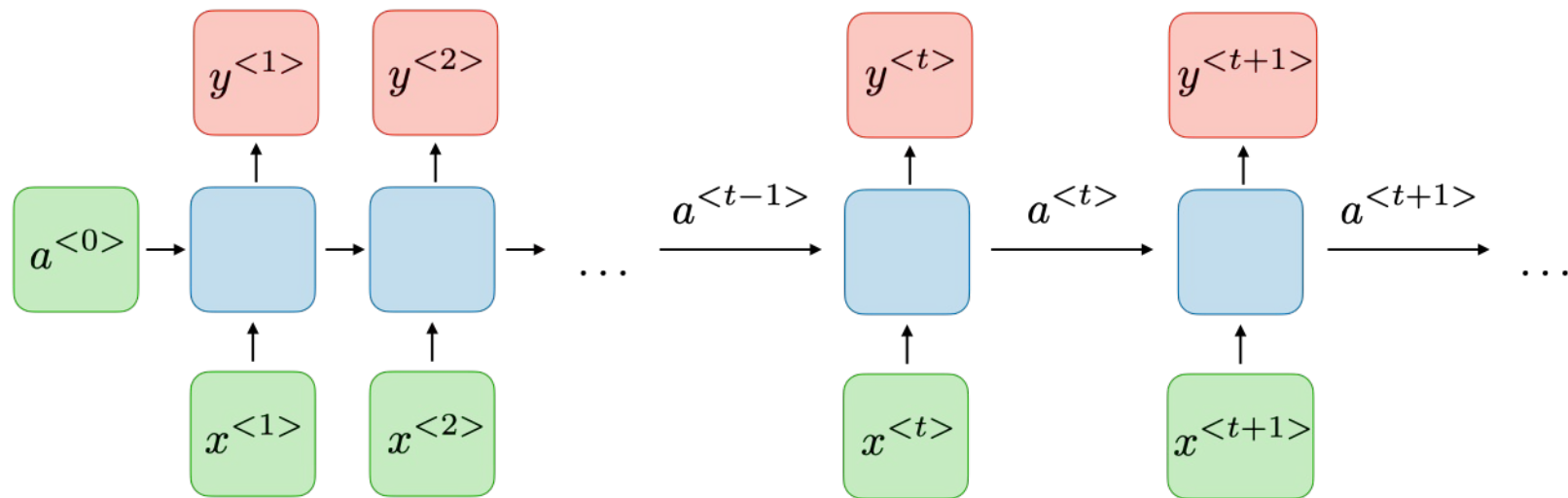
Анализ временных рядов

Азиз Темирханов

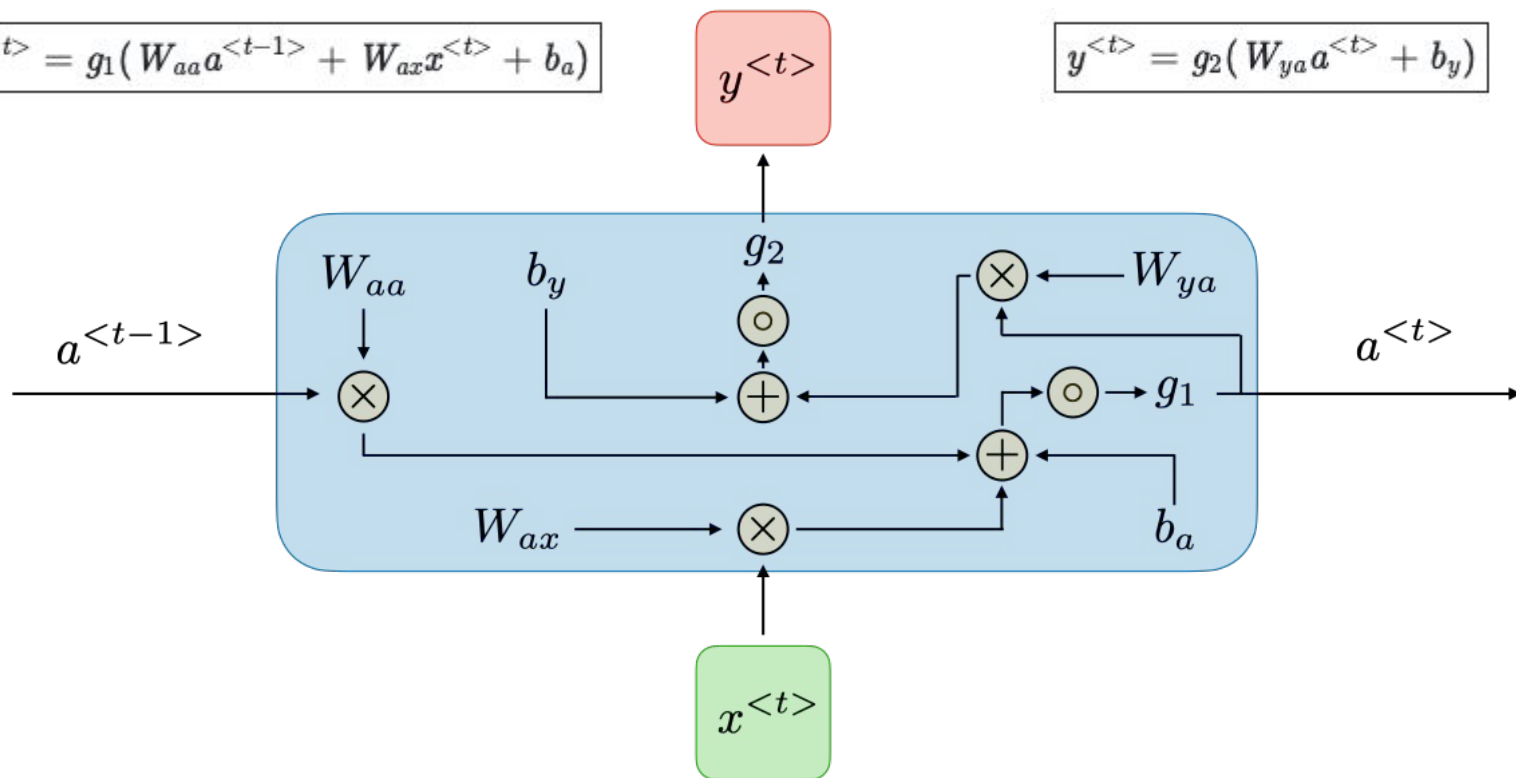
План лекции

- Вспомним рекуррентные сети
- Узнаем про свертки
- Разберем специализированные сетки

RNN



RNN

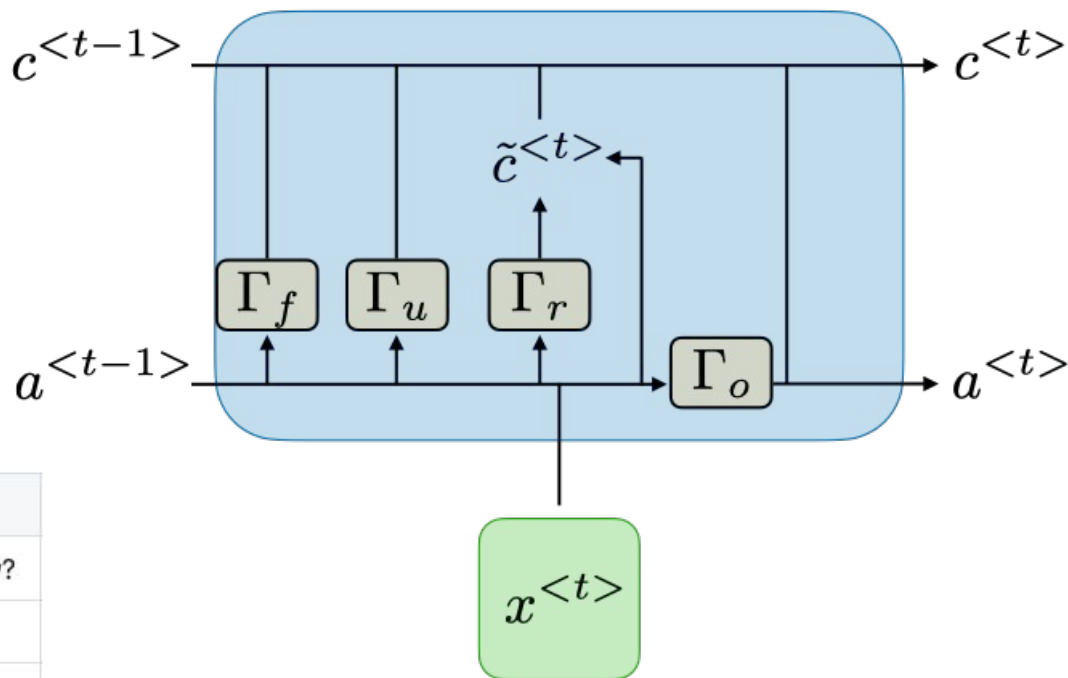


LSTM

$$\begin{aligned}\tilde{c}^{<t>} & \tanh(W_c[\Gamma_r \star a^{<t-1>}, x^{<t>}] + b_c) \\ c^{<t>} & \Gamma_u \star \tilde{c}^{<t>} + \Gamma_f \star c^{<t-1>} \\ a^{<t>} & \Gamma_o \star c^{<t>}\end{aligned}$$

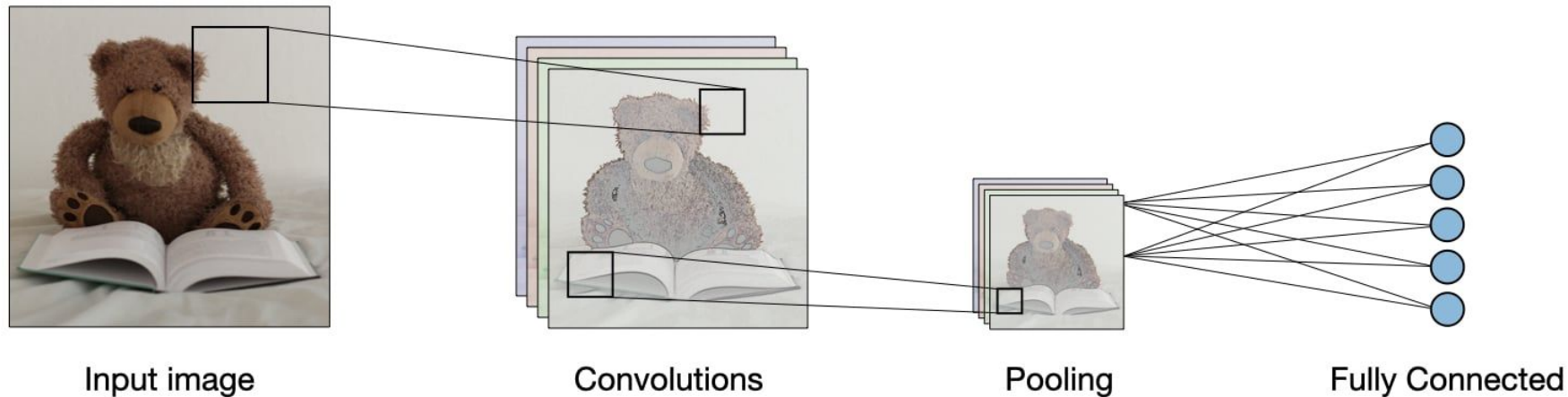
$$\Gamma = \sigma(Wx^{<t>} + Ua^{<t-1>} + b)$$

Type of gate	Role
Update gate Γ_u	How much past should matter now?
Relevance gate Γ_r	Drop previous information?
Forget gate Γ_f	Erase a cell or not?
Output gate Γ_o	How much to reveal of a cell?

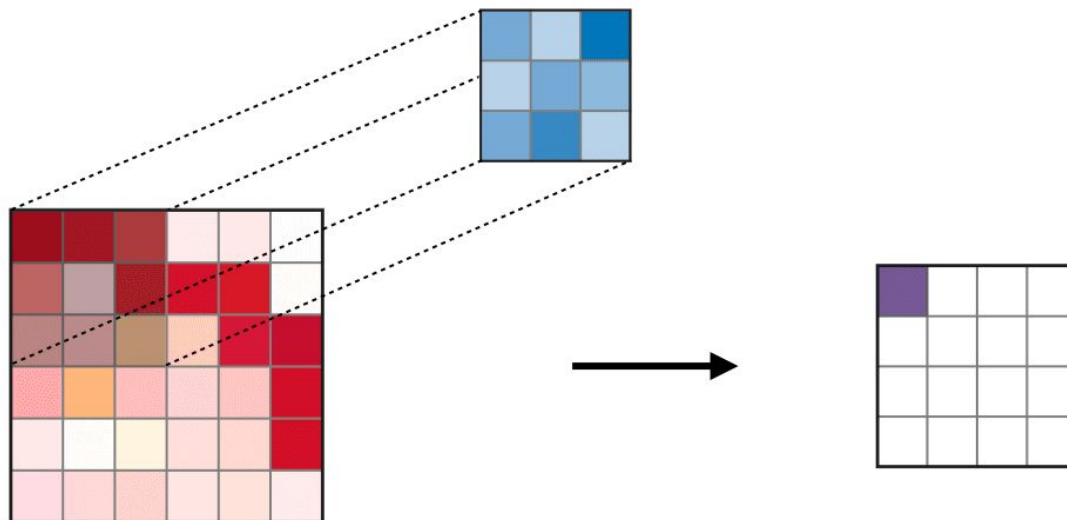


Свертки

CNN. Свертки

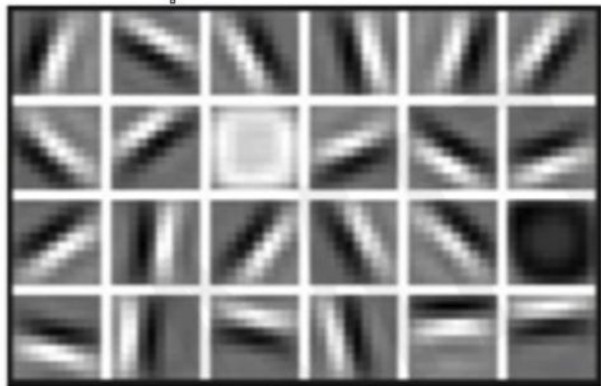


Свертка



Свертки и их смысл

Low Level Features



Lines & Edges

Mid Level Features



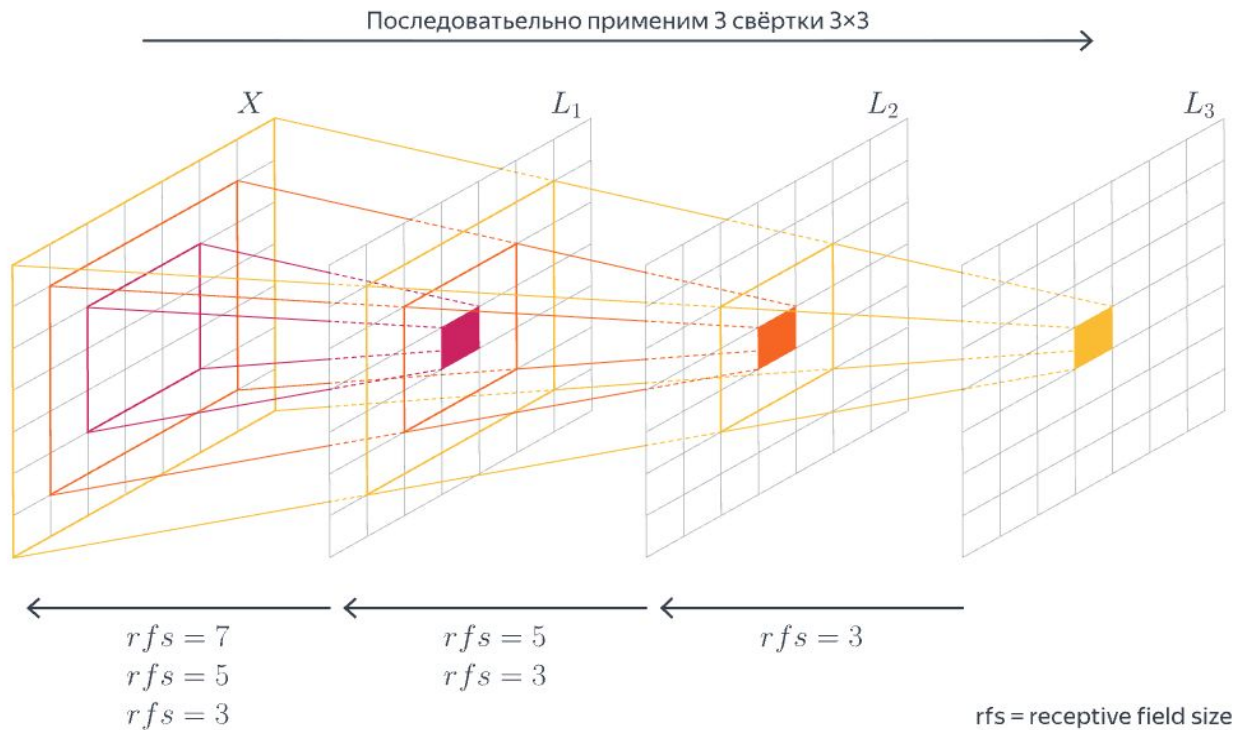
Eyes & Nose & Ears

High Level Features



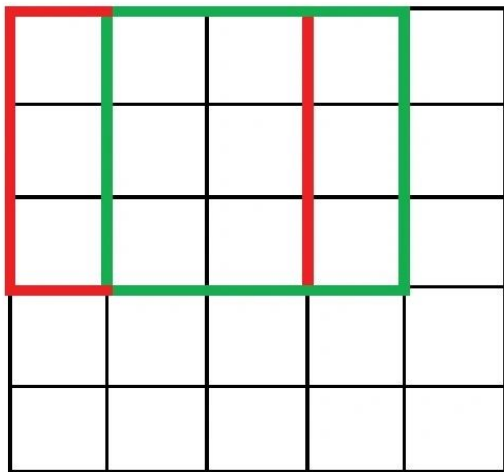
Facial Structure

Свертки и поле восприятия

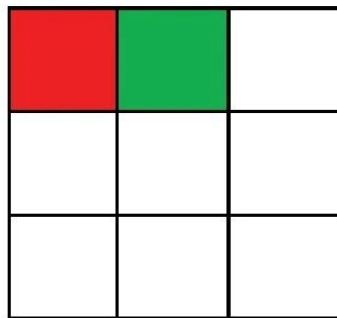


Некоторые гиперпараметры. Шаг

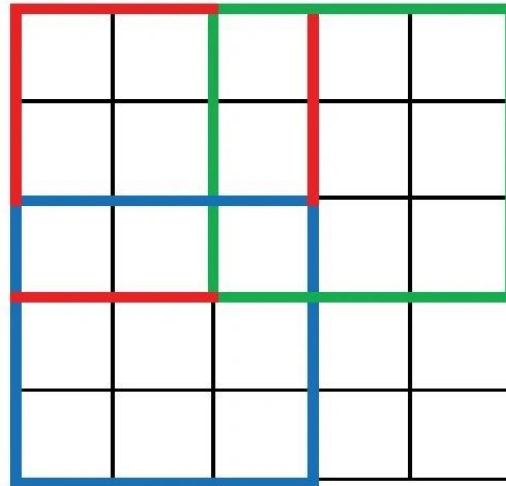
Convolution
with Stride=1



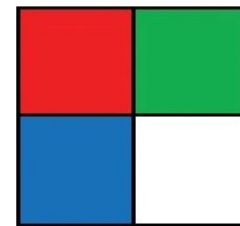
Output



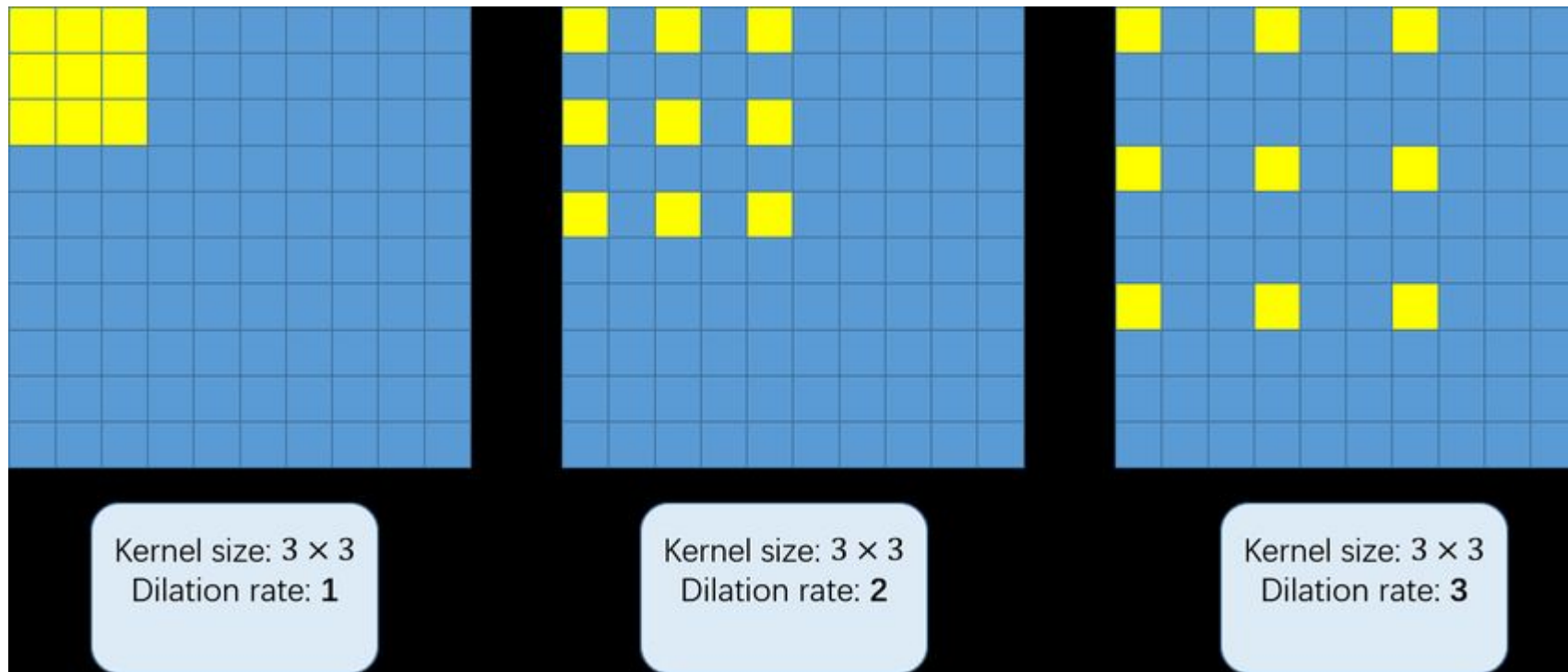
Convolution
with Stride=2



Output



Некоторые гиперпараметры. Расширение.



Некоторые гиперпараметры. Padding.

0	0	0	0	0	0	0	0
0	3	3	4	4	7	0	0
0	9	7	6	5	8	2	0
0	6	5	5	6	9	2	0
0	7	1	3	2	7	8	0
0	0	3	7	1	8	3	0
0	4	0	4	3	2	2	0
0	0	0	0	0	0	0	0

$6 \times 6 \rightarrow 8 \times 8$

*

1	0	-1
1	0	-1
1	0	-1

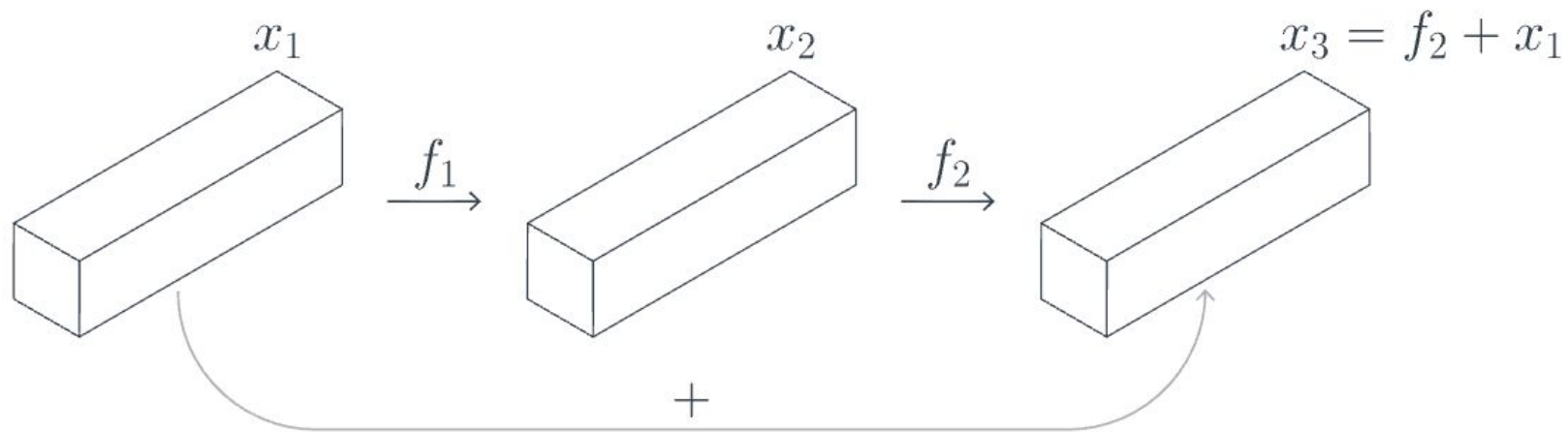
3×3

=

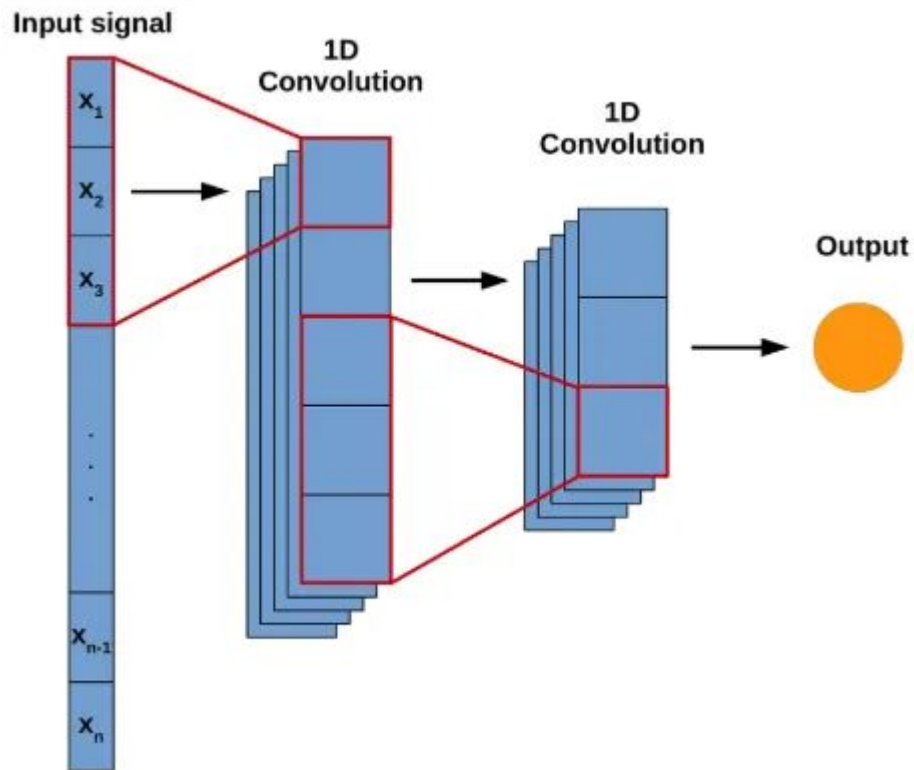
-10	-13	1			
-9	3	0			

6×6

Residual

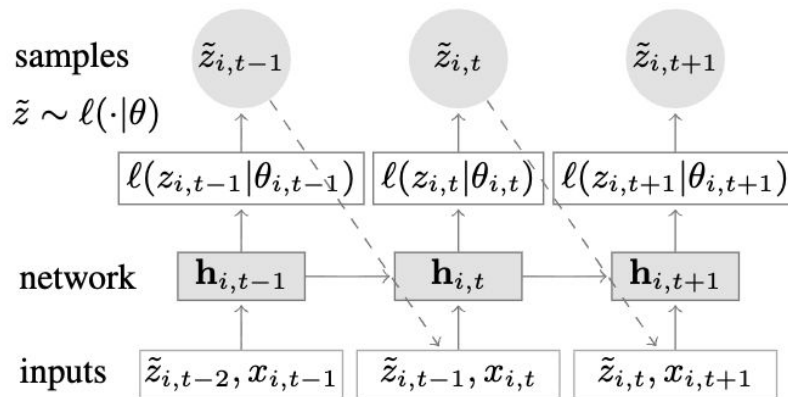
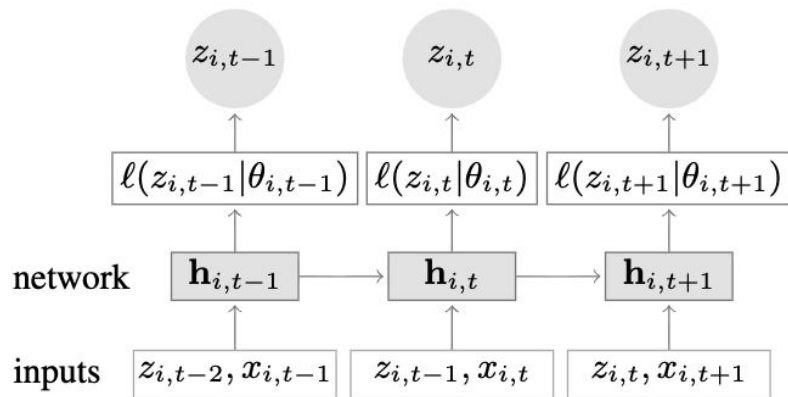


1D свертки



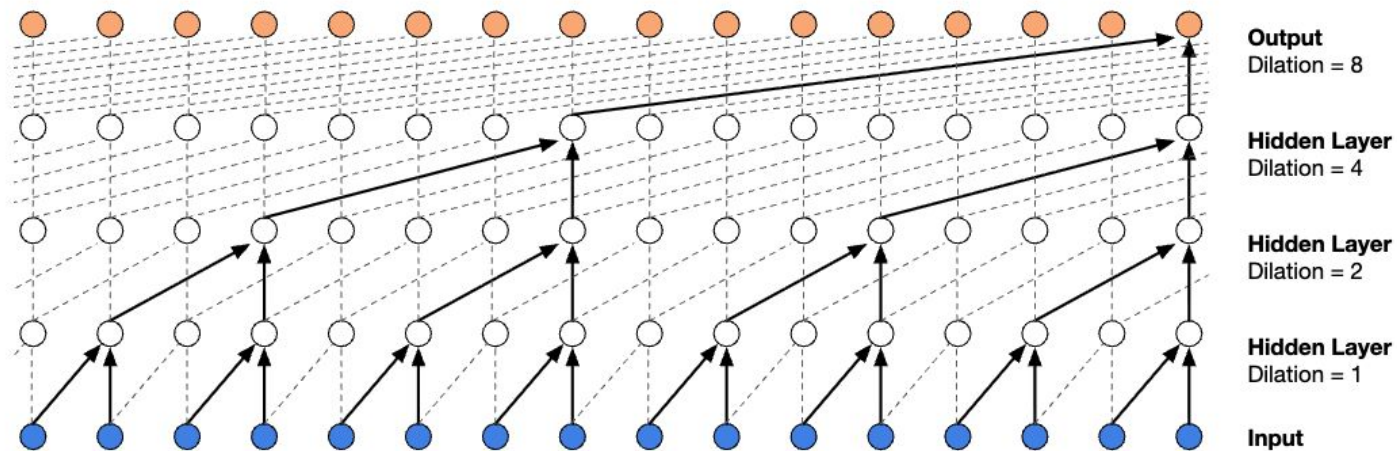
Специализированные модели

DeepAR (2020, 3275 citations)



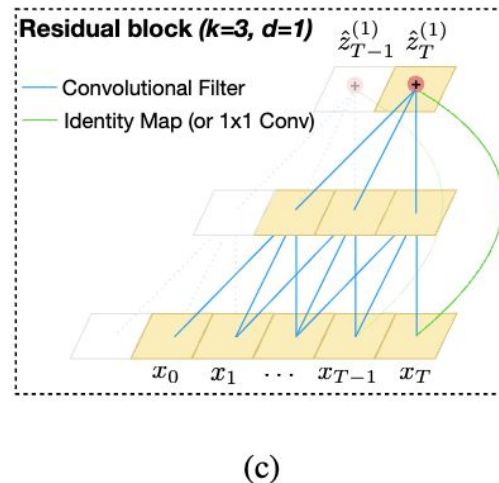
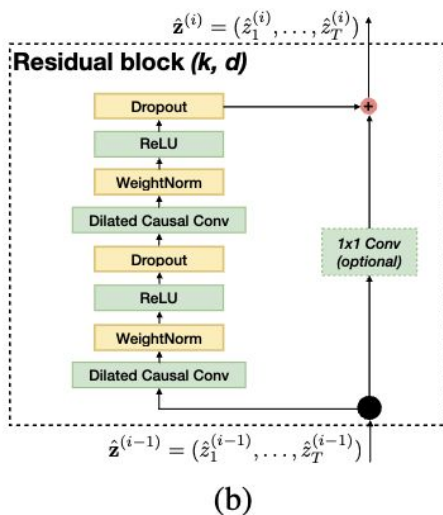
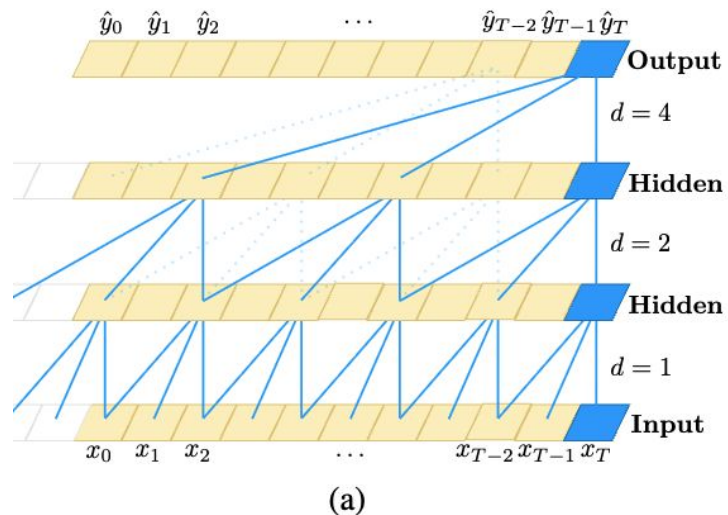
- Моделируют $p(\mathbf{y}_{t:T} | \mathbf{y}_{<t}, \text{covariates})$ с помощью RNN (LSTM) предсказывать параметры распределения каждый шаг
- Обучение через teacher forcing — на вход подают всегда истинное значение, максимизируют логарифм правдоподобия
- Инференс через семплирование из распределения, получают квантили
- Популяризировали global постановку

WaveNet (2016, 2155 citations)



- Наличие каузальности в свертках, поскольку они устроены так, что предсказание x_t видит только $x_{<t}$
- Расстояние между элементами в ядре растет за счет расширения, поэтому сеть агрегирует информацию с ранних токенов
- Лучше параллелится по сравнению с RNN
- Часто используется как backbone для feature-extraction

TCN (2018, 9340 citations)



Каузальные дилатированные свертки, residual блоки, большой эффективный receptive field.

TCN часто выигрывает у канонических RNN по точности, лучше держит очень длинные зависимости, проще/стабильнее обучается и лучше параллелится.

LSTNet (2018, 3271 citations)

Идея: объединить локальные паттерны (CNN), долгие зависимости (RNN), и линейный масштаб (AR).

Ключевые компоненты:

CNN без pooling извлекает короткие локальные временные шаблоны (и взаимодействия между признаками в многомерном ряде).

GRU/RNN моделирует более долгую динамику.

Skip-RNN, которая прыгает через период (например, через 24 часа), чтобы лучше улавливать сезонность.

AR-компонента — просто линейная регрессия по последним значениям; добавляется к нелинейному прогнозу, чтобы сеть не теряла чувствительность к уровню/масштабу.

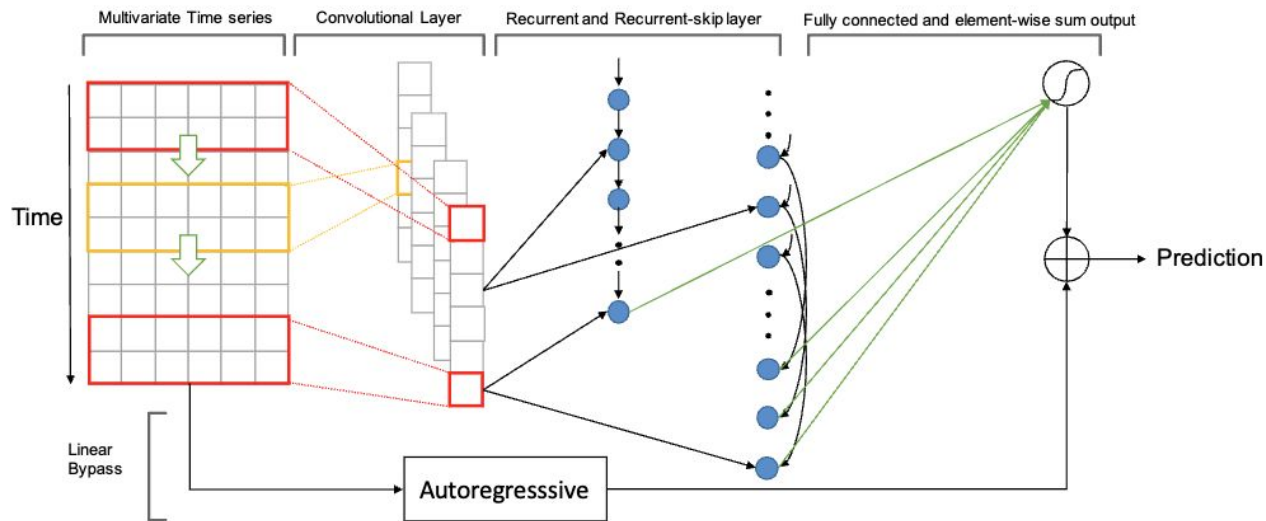
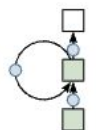


Figure 2: An overview of the Long- and Short-term Time-series network (LSTNet)

xLSTM (2024, 538 citations)

Вводятся новые механизмы гейтинга и стабилизации (с экспоненциальными/нормализованными элементами), чтобы лучше контролировать, что записывать в память и как забывать.

Предлагаются два типа ячеек:

- **sLSTM** (скалярная память с новыми правилами смешивания/нормализации),
$$c_t = f_t c_{t-1} + i_t z_t$$
$$h_t = o_t \psi(c_t)$$

- **mLSTM** (идея матричной памяти, дающей более богатое представление состояния).

Всё это собирается в блоки и стакается как современная глубокая архитектура

