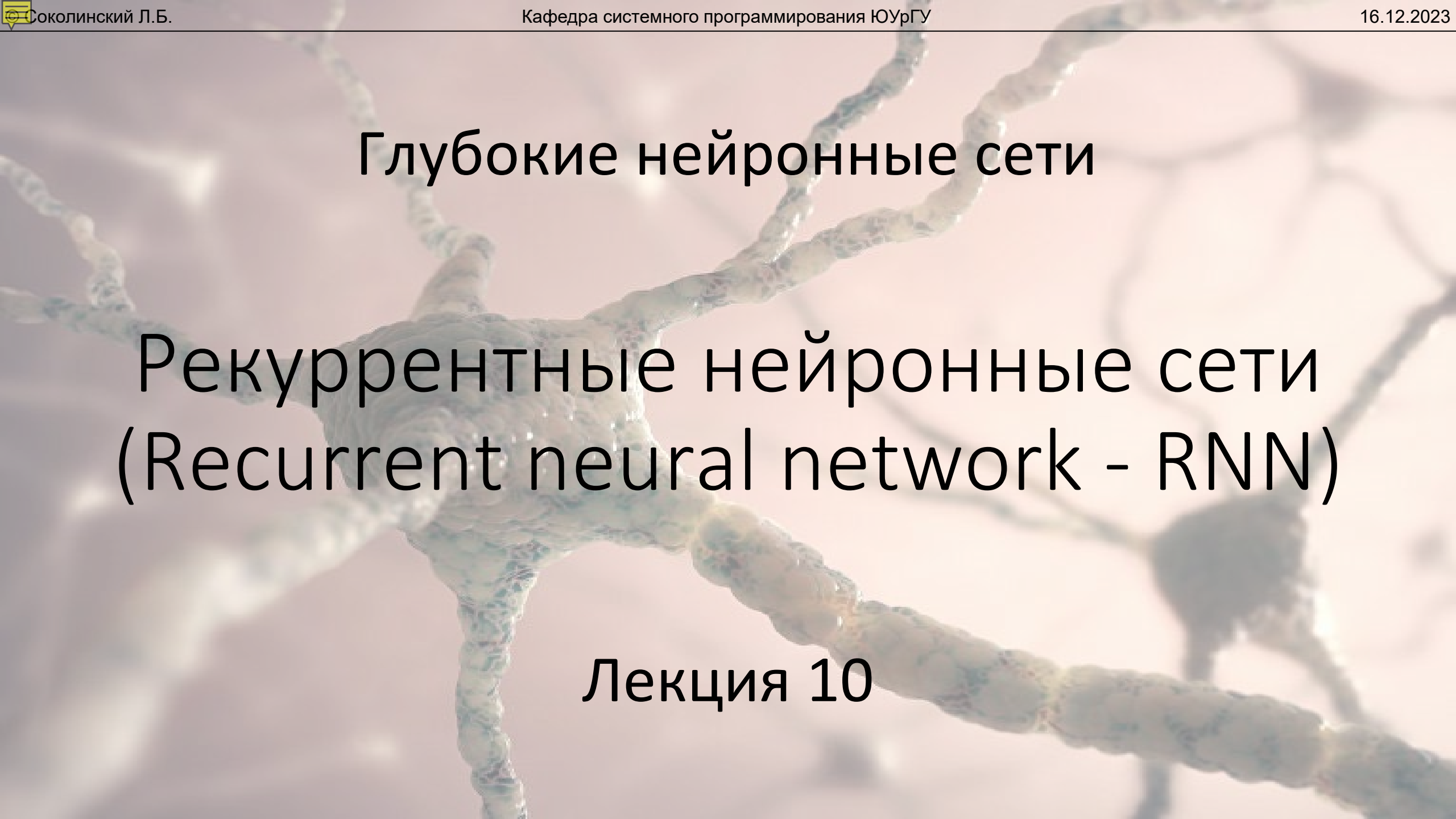


Глубокие нейронные сети

Рекуррентные нейронные сети (Recurrent neural network - RNN)

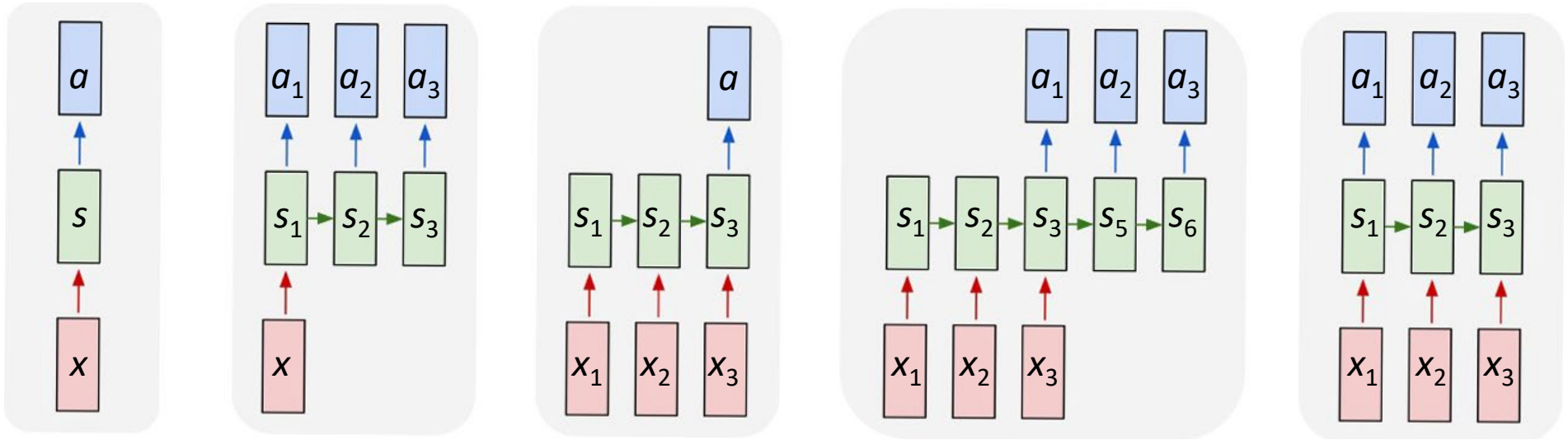
Лекция 10

Отличие RNN от CNN

- CNN ориентированы на анализ изображений, являющихся моментальными снимками состояний группы объектов в некоторый момент времени (нет анализа предыстории)
- RNN ориентированы на анализ временной последовательности состояний группы объектов

Типы приложений

a) 1:1 b) 1:M c) M:1 d) 1×M : 1×M e) M:M



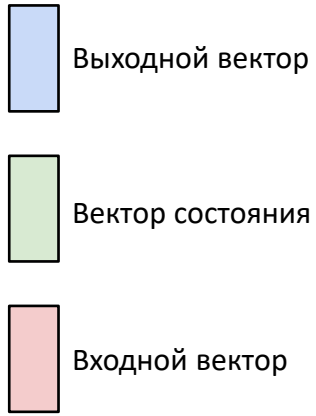
a) Один вход, один выход без запоминания состояния (распознавание изображений)

b) Последовательность на выходе (вход – картинка, выход – ее словесное описание)

c) Последовательность на входе (вход – рецензия на фильм, выход – оценка по десятибалльной шкале)

d) Последовательность на входе, последовательность на выходе (перевод текста с одного языка на другой)

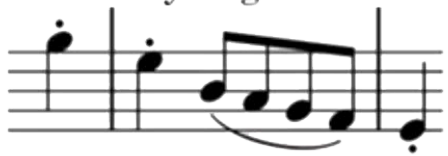
e) Синхронизированные последовательности на входе и выходе (разметка смены кадра на видео; очистка аудиопотока от посторонних шумов)



Область применения рекуррентных нейронных сетей

- Анализ временных рядов
  - Изменения цен акций
  - Показания датчиков
- Понимание (письменных) текстов на естественном языке
- Машинный перевод
- Понимание человеческой речи (перевод речи в текст)
- Обнаружение аномалий (определение предынфарктного состояния по кардиограмме)
- Классификация и аннотация текстов
- Предсказательная аналитика

Applications of RNNs

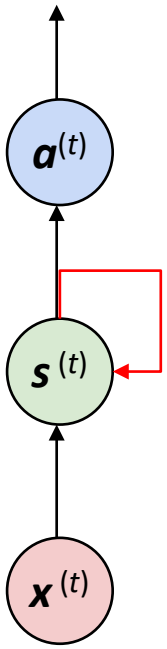
Speech recognition		"The quick brown fox jumped over the lazy dog."
Music generation	∅	
Sentiment classification	"There is nothing to like in this movie."	★☆☆☆☆
DNA sequence analysis	AGCCCCTGTGAGGAACTAG	AGCCCCTGTGAGGAACTAG
Machine translation	Voulez-vous chanter avec moi?	Do you want to sing with me?
Video activity recognition		Running
Name entity recognition	Yesterday, Harry Potter met Hermione Granger.	Yesterday, Harry Potter met Hermione Granger.

Andrew Ng

Идея

Запоминать что-то из истории приходящих на вход данных, сохраняя некое внутреннее состояние, которое можно было бы потом использовать для анализа текущих данных

Простейшая рекуррентная сеть

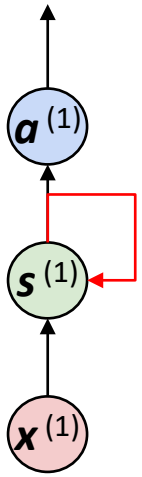


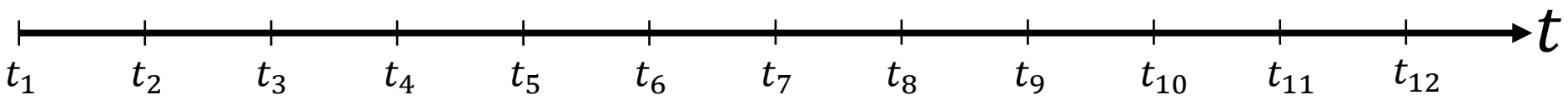
Выходной слой

Слой внутреннего состояния

Входной слой

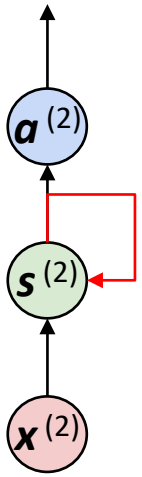
Работа рекуррентной нейронной сети

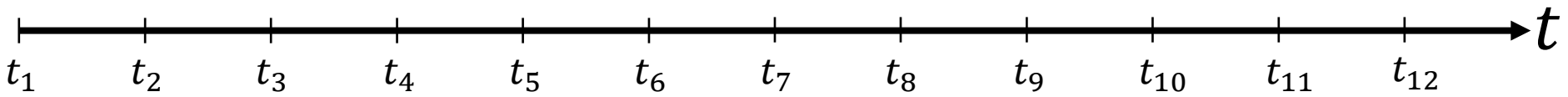




Время (порядок поступления сигналов)

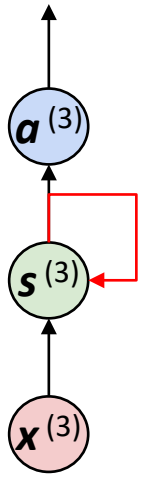
Работа рекуррентной нейронной сети

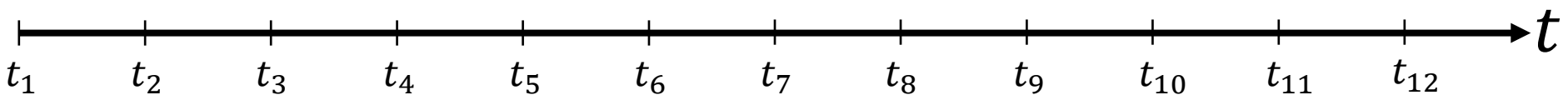




Время (порядок поступления сигналов)

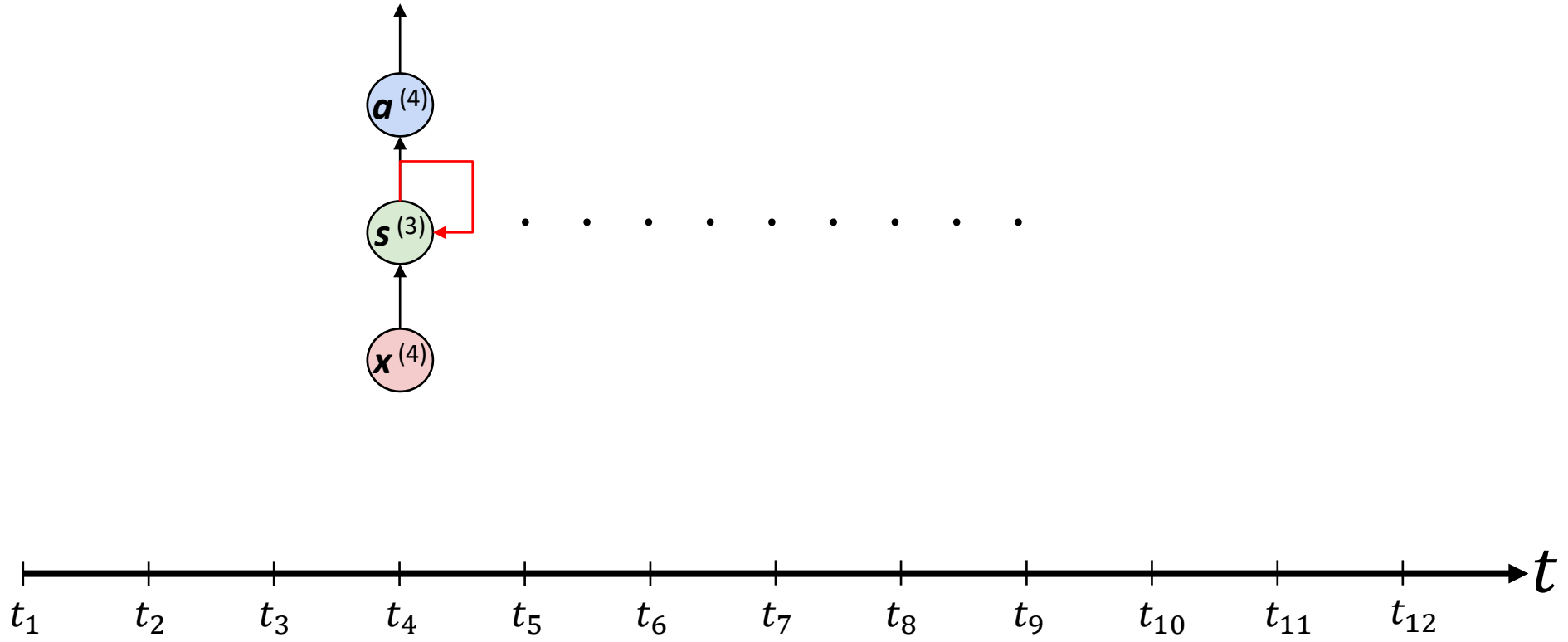
Работа рекуррентной нейронной сети



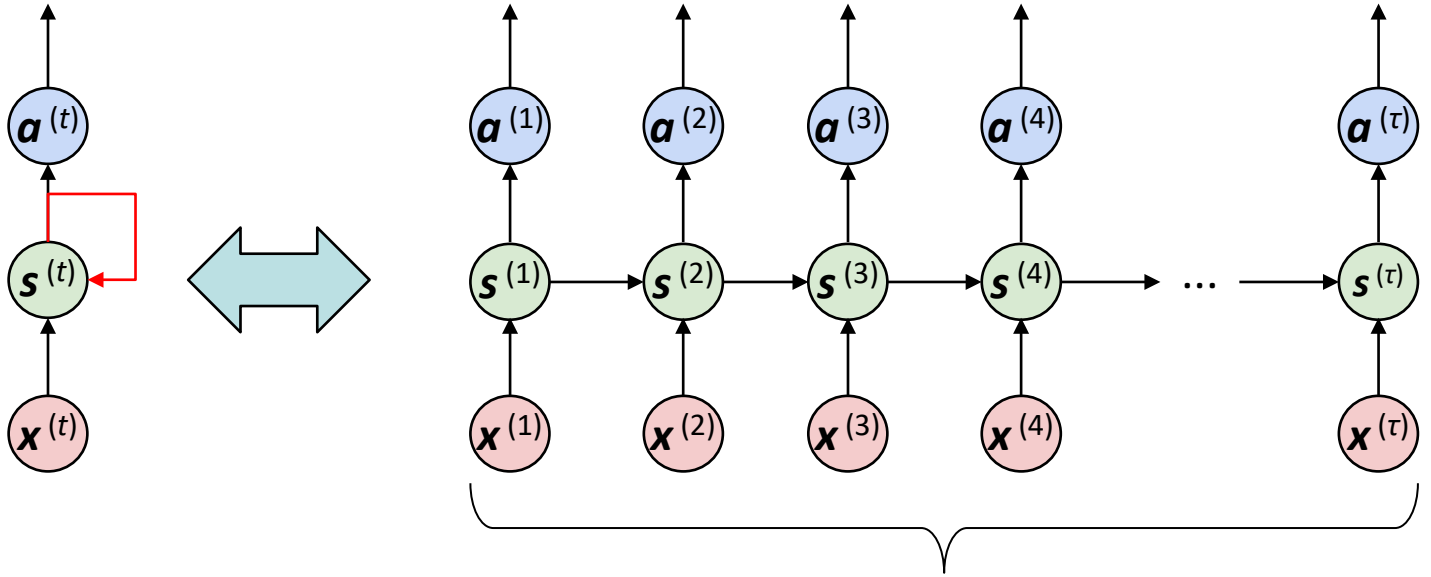


Время (порядок поступления сигналов)

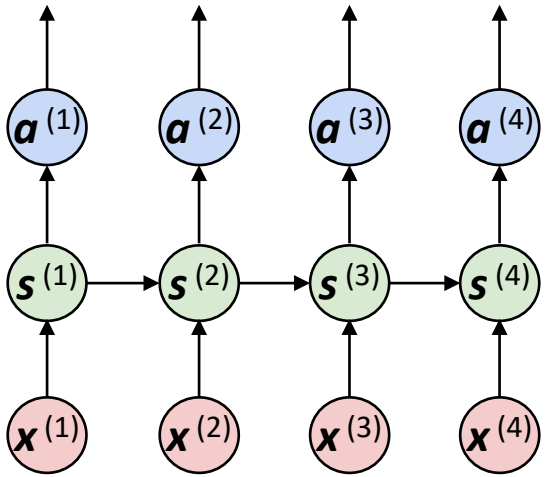
Работа рекуррентной нейронной сети

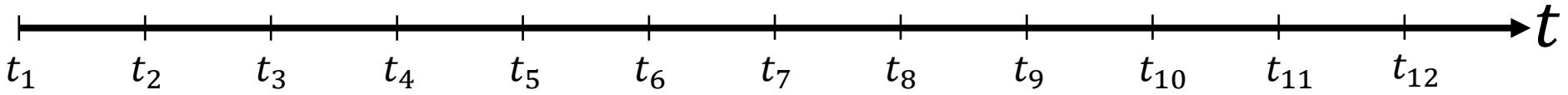


Развертка рекуррентной сети



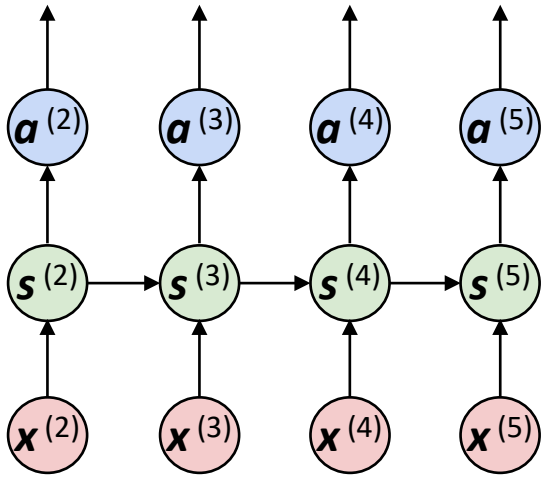
Работа развертки ($\boldsymbol{\tau = 4}$)

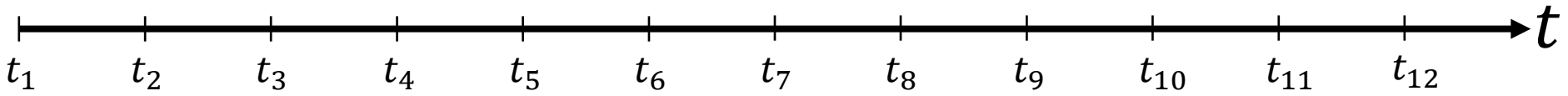




Время (порядок поступления сигналов)

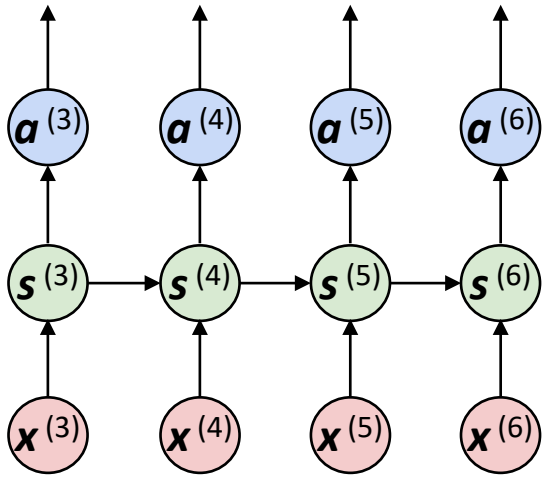
Работа развертки ($\boldsymbol{\tau = 4}$)

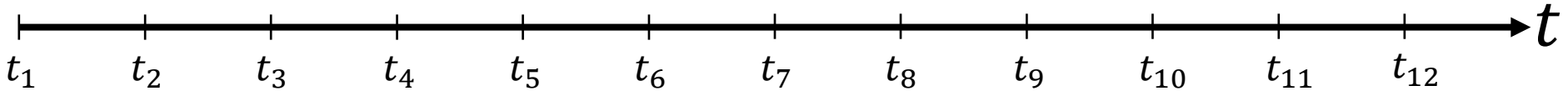




Время (порядок поступления сигналов)

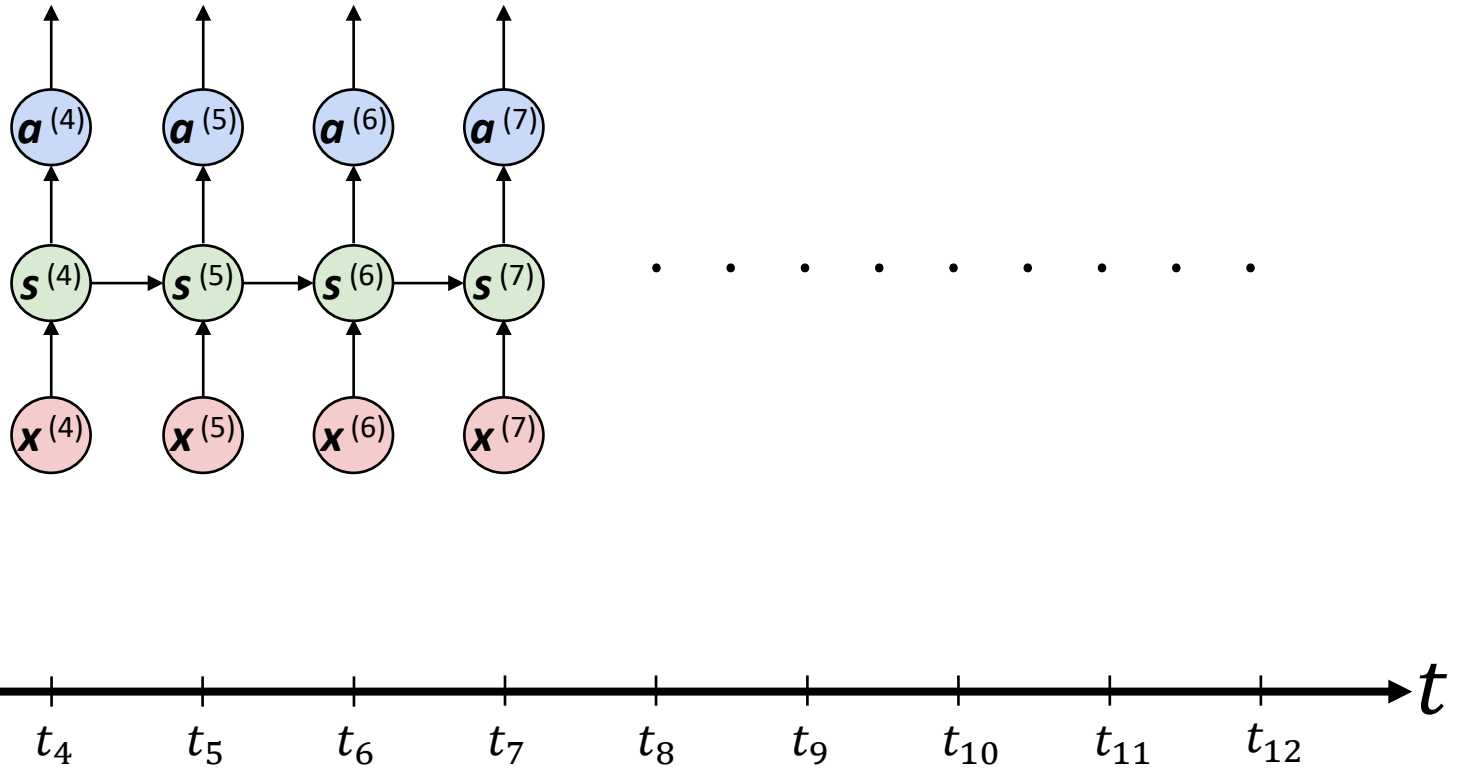
Работа развертки ($\boldsymbol{\tau = 4}$)



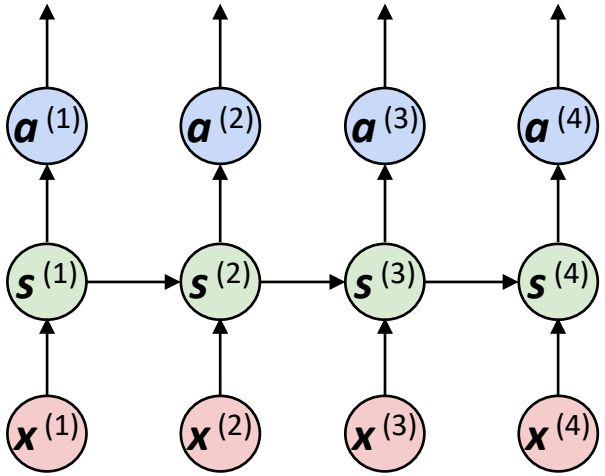


Время (порядок поступления сигналов)

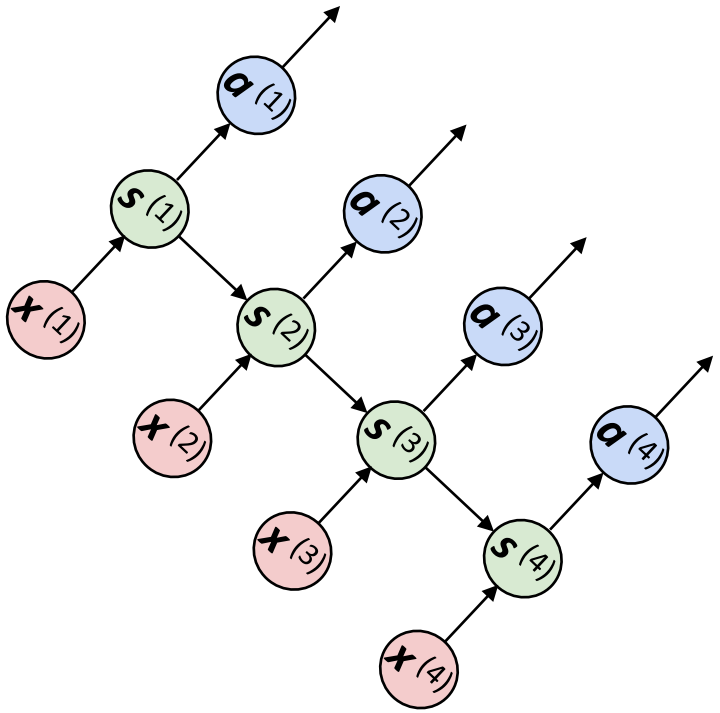
Работа развертки ($\boldsymbol{\tau = 4}$)



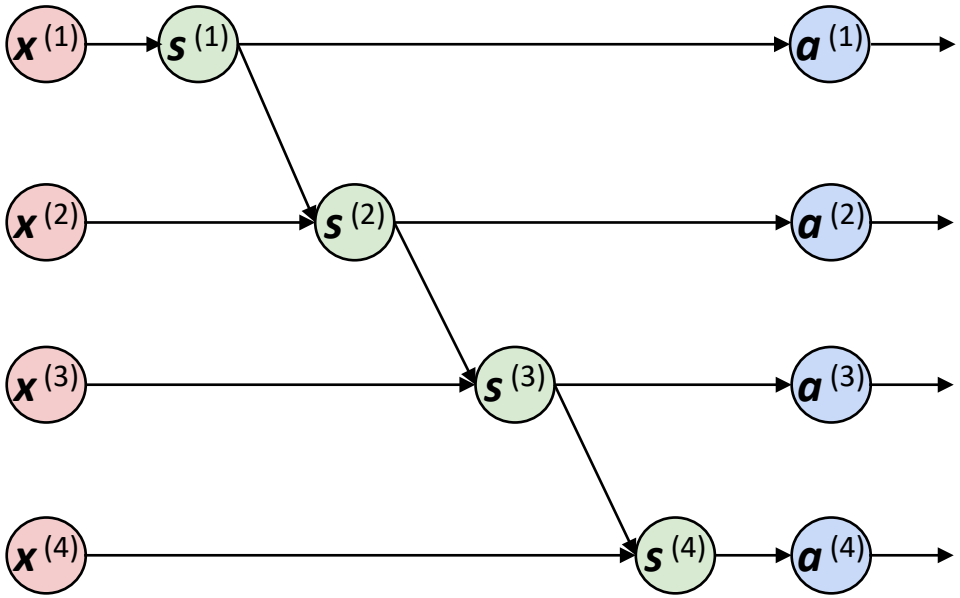
Развертка => сеть прямого распространения



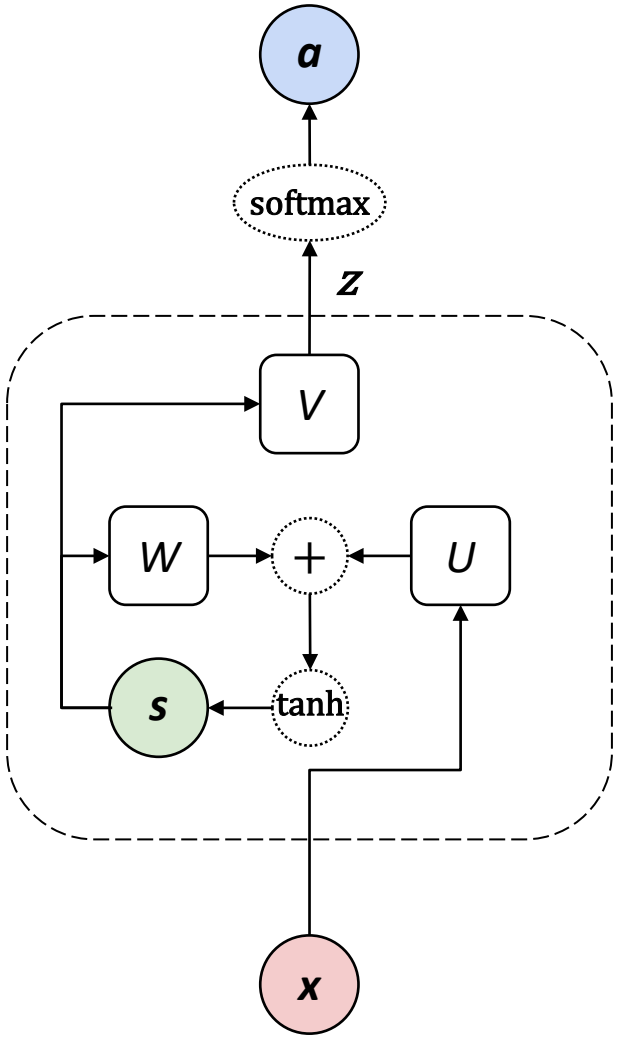
Развертка => сеть прямого распространения



Развертка => сеть прямого распространения



Модель Simple RNN



1) $\boldsymbol{s}^{(t)} = \boldsymbol{tanh}\left(W\boldsymbol{s}^{(t-1)} + U\boldsymbol{x}^{(t)} + \boldsymbol{b}\right)$

2) $\boldsymbol{z}^{(t)} = V\boldsymbol{s}^{(t)} + \boldsymbol{d}$

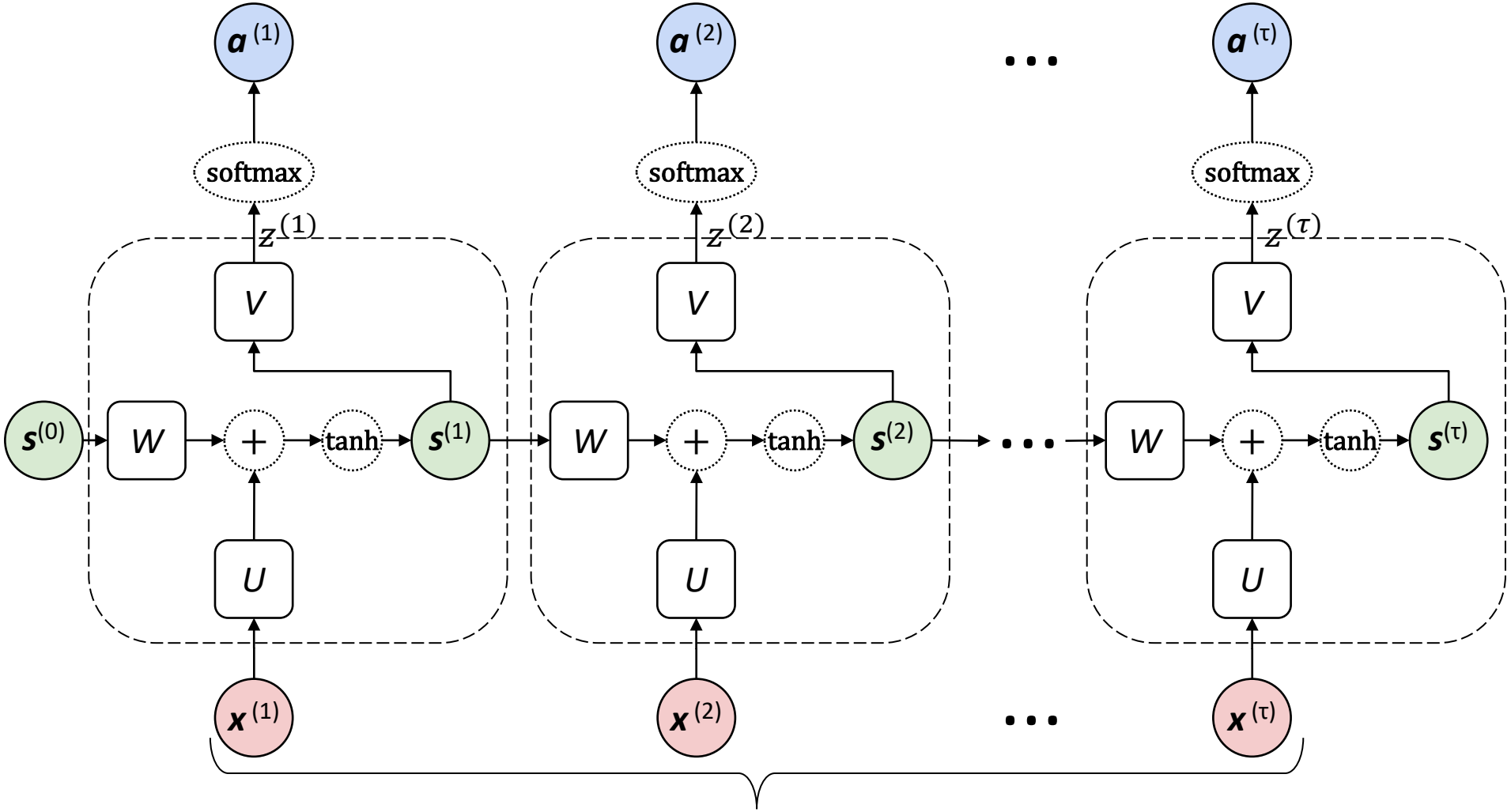
3) $\boldsymbol{a}^{(t)} = \boldsymbol{softmax}\left(\boldsymbol{z}^{(t)}\right)$

$$t = 1,2,3, \ldots$$

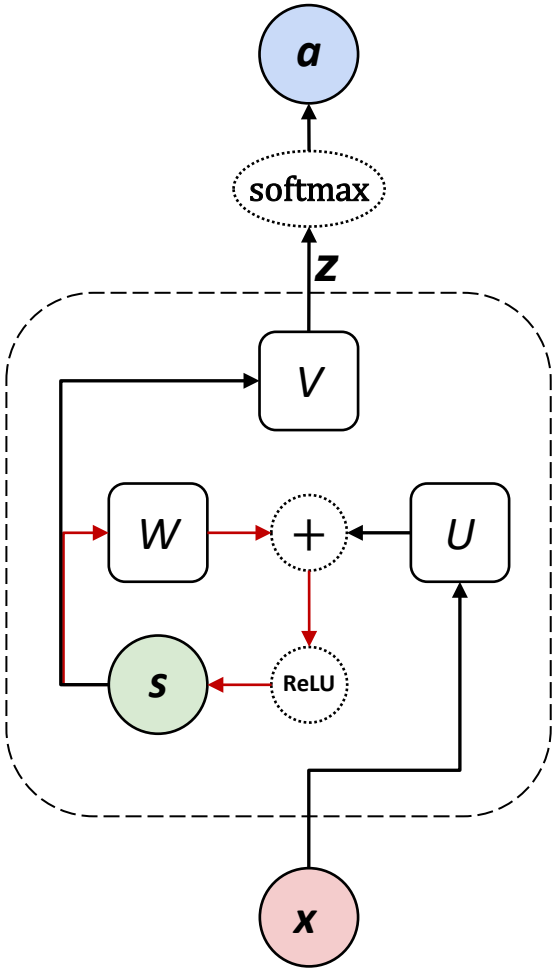
$\boldsymbol{s}^{(t)}$ – *вектор начального состояния*

$\boldsymbol{b}, \boldsymbol{d}$ – *векторы смещений*

Развертка Simple RNN



Задача: вычислить $\boldsymbol{a}^{(2)}$



$b = 0$

$d = 0$

$\boldsymbol{s}^{(0)} = (0, 0)$

$\boldsymbol{x}^{(1)} = (1, 2); \boldsymbol{x}^{(2)} = (2, 1)$

$$U = \begin{bmatrix} 0.9 & 0.2 \\ 0.4 & 0.6 \end{bmatrix}; W = \begin{bmatrix} 0.5 & 0.8 \\ 0.4 & 0.2 \end{bmatrix}; V = \begin{bmatrix} 0.3 & 0.8 \\ 0.9 & 0.4 \end{bmatrix}$$

$\boldsymbol{s}^{(1)} = U\boldsymbol{x}^{(1)} + W\boldsymbol{s}^{(0)} = (\,1.3\,,1.6\,)$

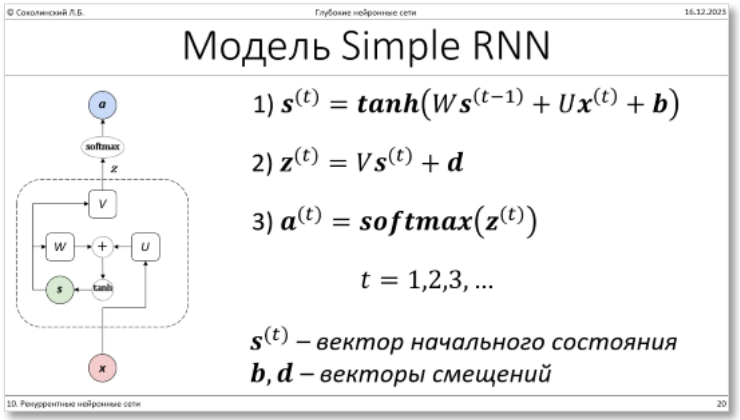
$\boldsymbol{s}^{(2)} = U\boldsymbol{x}^{(2)} + W\boldsymbol{s}^{(1)} = (\,3.93\,,2.24\,)$

$\boldsymbol{z}^{(2)} = V\boldsymbol{s}^{(2)} = (\,2.971\,,4.433\,)$

$e^{z_1^{(2)}} = 19.51142\,;\, e^{z_2^{(2)}} = 84.18359$

$e^{z_1^{(2)}} + e^{z_2^{(2)}} = 103.69501$

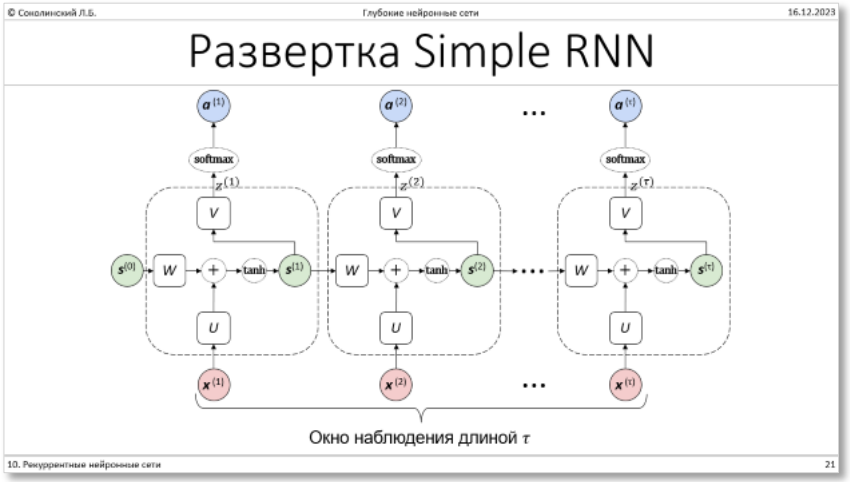
$$\boldsymbol{a}^{(2)} = \left(\frac{e^{z_1^{(2)}}}{e^{z_1^{(2)}} + e^{z_2^{(2)}}}, \frac{e^{z_2^{(2)}}}{e^{z_1^{(2)}} + e^{z_2^{(2)}}} \right) = (\,0.18816\,,0.81184\,)$$

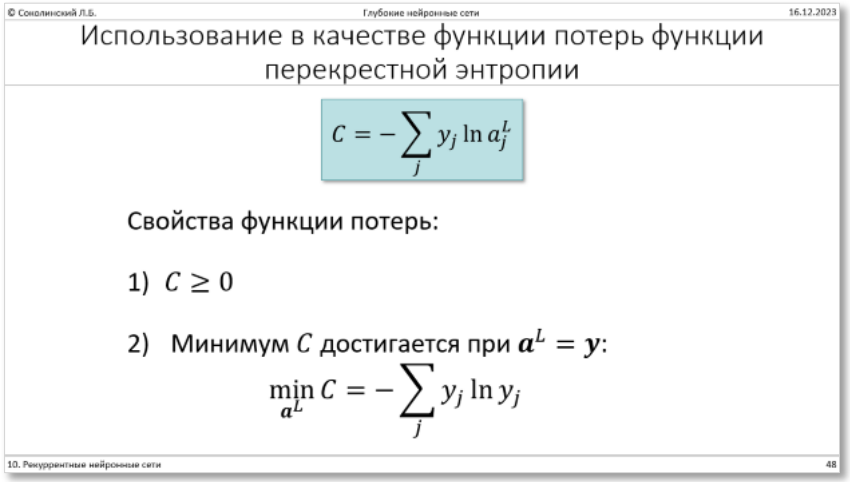


Обратное распространение во времени (Back propagation through time)

Ошибка в момент времени t:

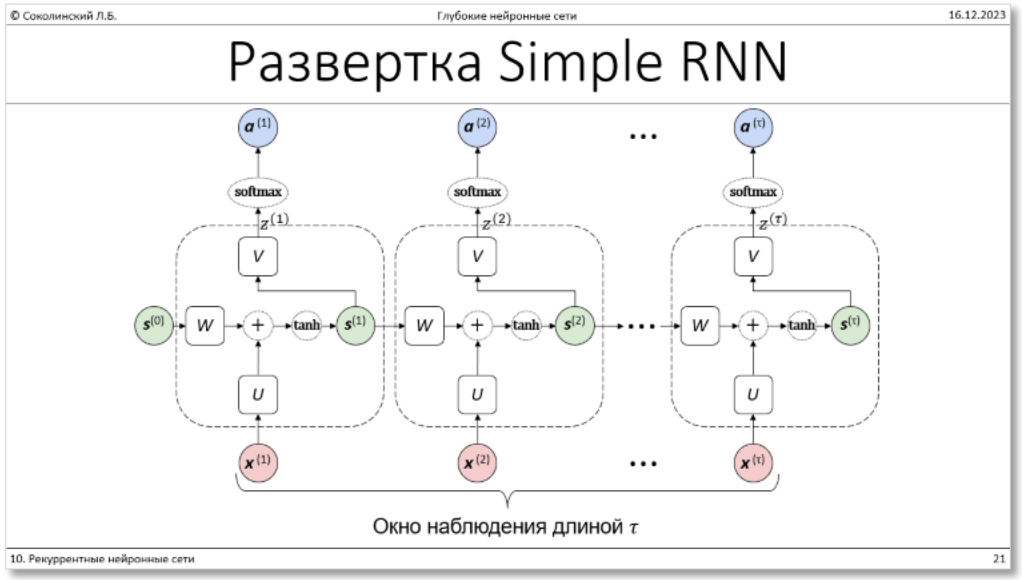
$$C_{(\boldsymbol{x}^{(t)},\boldsymbol{y}^{(t)})} = \sum_j y_j^{(t)} \ln a_j^{(t)}$$





Суммарная ошибка

$$\mathbb{C}_{(\{\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(\tau)}\},\{\boldsymbol{y}^{(1)},\ldots,\boldsymbol{y}^{(\tau)}\})} = \sum_{t=1}^{\tau} C_{(\boldsymbol{x}^{(t)},\boldsymbol{y}^{(t)})}$$



Производная для суммарной ошибки

Обозначим

$$\mathbb{C} = \mathbb{C}_{(\{\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(\tau)}\},\{\boldsymbol{y}^{(1)},\ldots,\boldsymbol{y}^{(\tau)}\})}$$

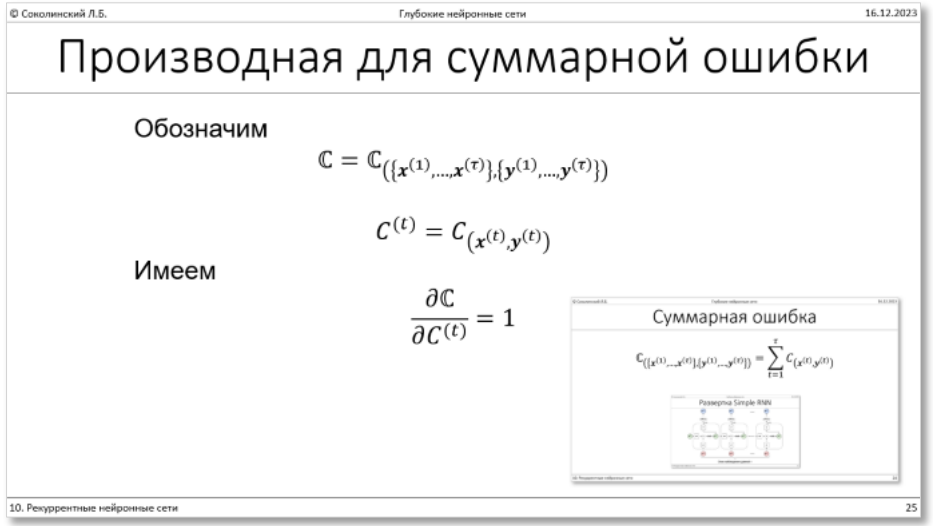
$$C^{(t)} = C_{(\boldsymbol{x}^{(t)},\boldsymbol{y}^{(t)})}$$

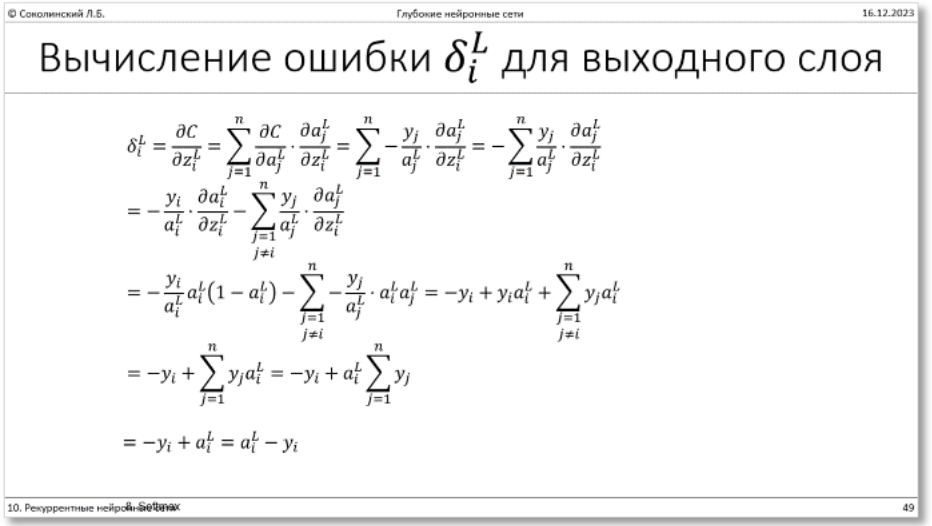
Имеем

$$\frac{\partial \mathbb{C}}{\partial C^{(t)}} = 1$$

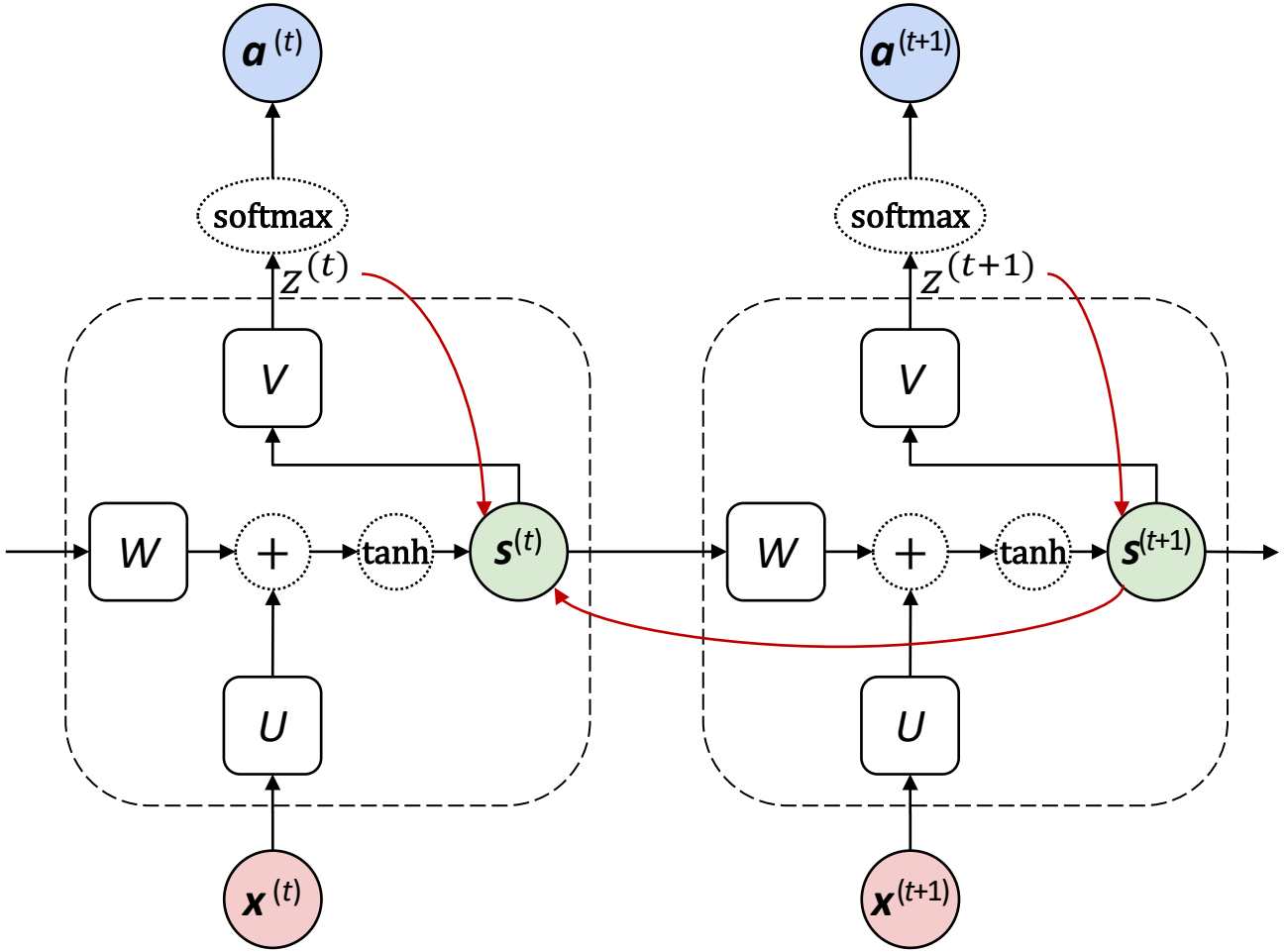
Вычисление градиента для выходного слоя

$$\left(\nabla_{\boldsymbol{z}^{(t)}}\mathbb{C}\right)_j = \frac{\partial \mathbb{C}}{\partial z_j^{(t)}} = \frac{\partial \mathbb{C}}{C^{(t)}}\frac{\partial C^{(t)}}{\partial z_j^{(t)}} = a_j^{(t)} - y_j$$





Обратные зависимости для слоя состояний



Вычисление градиентов для слоя состояний

Для $t = \tau$:

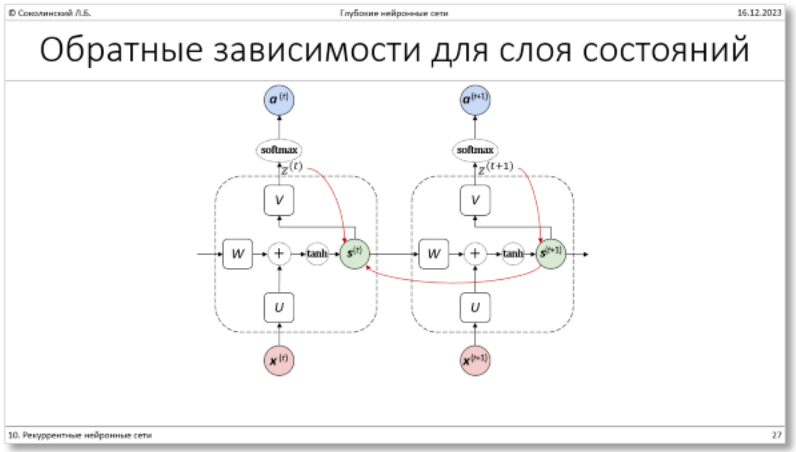
$$\nabla_{\boldsymbol{s}^{(\tau)}} \mathbb{C} = \frac{\partial \boldsymbol{z}^{(t)}}{\partial \boldsymbol{s}^{(t)}} \circ \nabla_{\boldsymbol{z}^{(t)}} \mathbb{C} = V^{\top} \nabla_{\boldsymbol{z}^{(\tau)}} \mathbb{C}$$

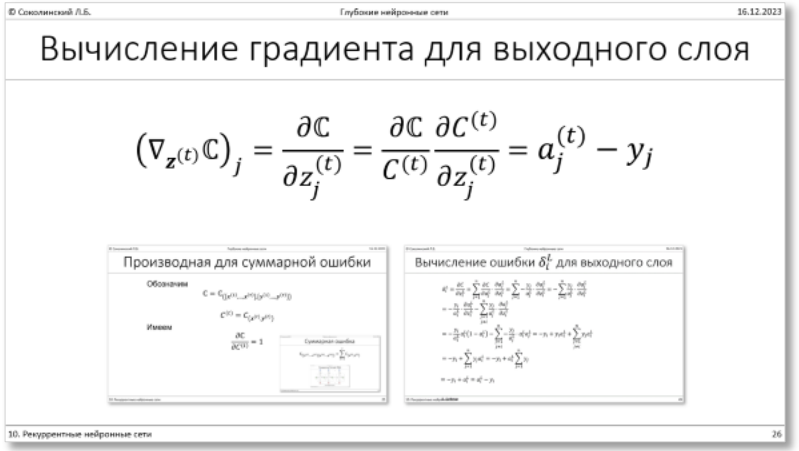
Для $t = \tau$-1,…,1:

$$\nabla_{\boldsymbol{s}^{(t)}} \mathbb{C} = \frac{\partial \boldsymbol{s}^{(t+1)}}{\partial \boldsymbol{s}^{(t)}} \circ \nabla_{\boldsymbol{s}^{(t+1)}} \mathbb{C} + \frac{\partial \boldsymbol{z}^{(t)}}{\partial \boldsymbol{s}^{(t)}} \circ \nabla_{\boldsymbol{z}^{(t)}} \mathbb{C}$$
$$= \mathrm{W}^{\top} \left(\nabla_{\boldsymbol{s}^{(t+1)}} \mathbb{C} \right) \cdot diag \left(1 - \left(\boldsymbol{s}^{(t+1)} \right)^2 \right) + \mathrm{V}^{\top} \left(\nabla_{\boldsymbol{z}^{(t)}} \mathbb{C} \right)$$

$diag \left(1 - \left(s^{(t+1)} \right)^2 \right)$ - диагональная матрица с элементами $1 - \left(s_j^{(t+1)} \right)^2$

$\tanh'(x) = 1 - \tanh^2(x)$ (см. https://socratic.org/questions/what-is-the-derivative-of-tanh-x)





Вычисление градиентов для весов и смещений

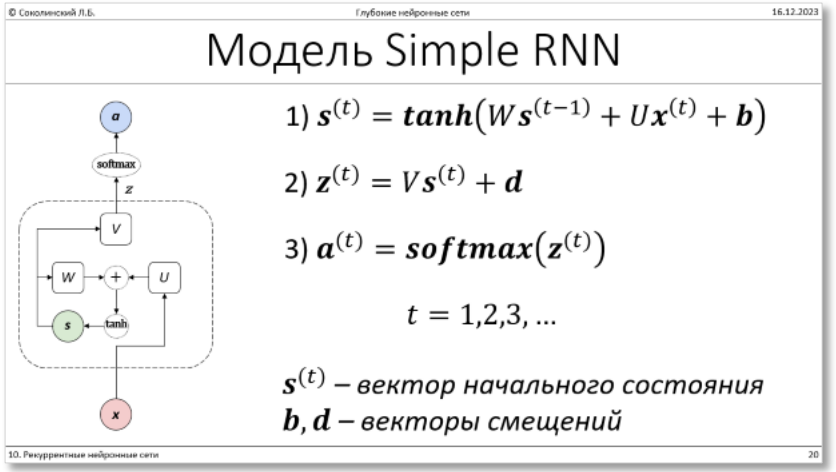
$$\nabla_{\boldsymbol{d}}\mathbb{C} = \sum_t \nabla_{\boldsymbol{z}^{(t)}}\mathbb{C}$$

$$\nabla_{\boldsymbol{b}}\mathbb{C} = \sum_t diag\left(1 - \left(\boldsymbol{s}^{(t)}\right)^2\right)\nabla_{\boldsymbol{s}^{(t)}}\mathbb{C}$$

$$\nabla_V\mathbb{C} = \sum_t \left(\nabla_{\boldsymbol{z}^{(t)}}\mathbb{C}\right) \circ \boldsymbol{s}^{(t)}$$

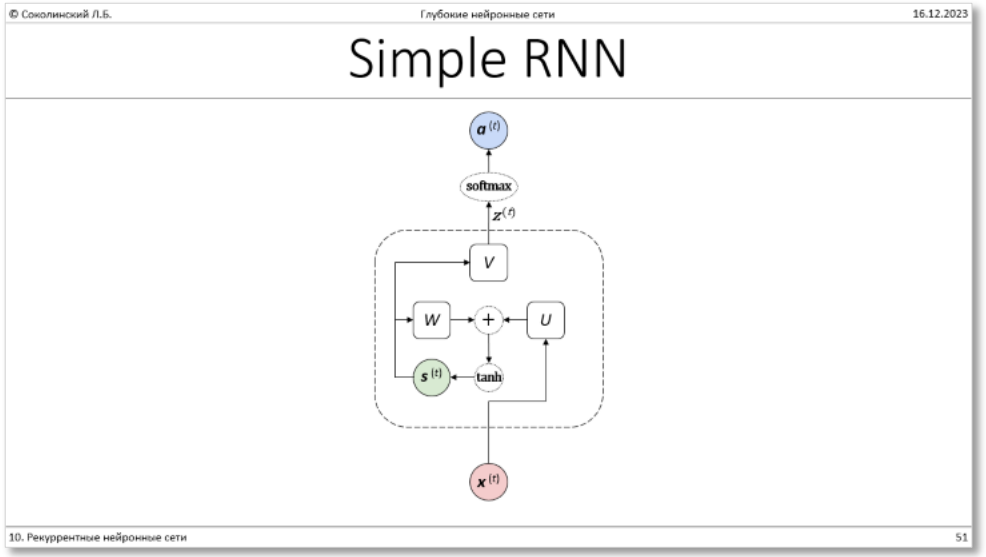
$$\nabla_W\mathbb{C} = \sum_t diag\left(1 - \left(\boldsymbol{s}^{(t)}\right)^2\right)\left(\nabla_{\boldsymbol{s}^{(t)}}\mathbb{C}\right) \circ \boldsymbol{s}^{(t-1)}$$

$$\nabla_U\mathbb{C} = \sum_t diag\left(1 - \left(\boldsymbol{s}^{(t)}\right)^2\right)\left(\nabla_{\boldsymbol{s}^{(t)}}\mathbb{C}\right) \circ \boldsymbol{x}^{(t)}$$

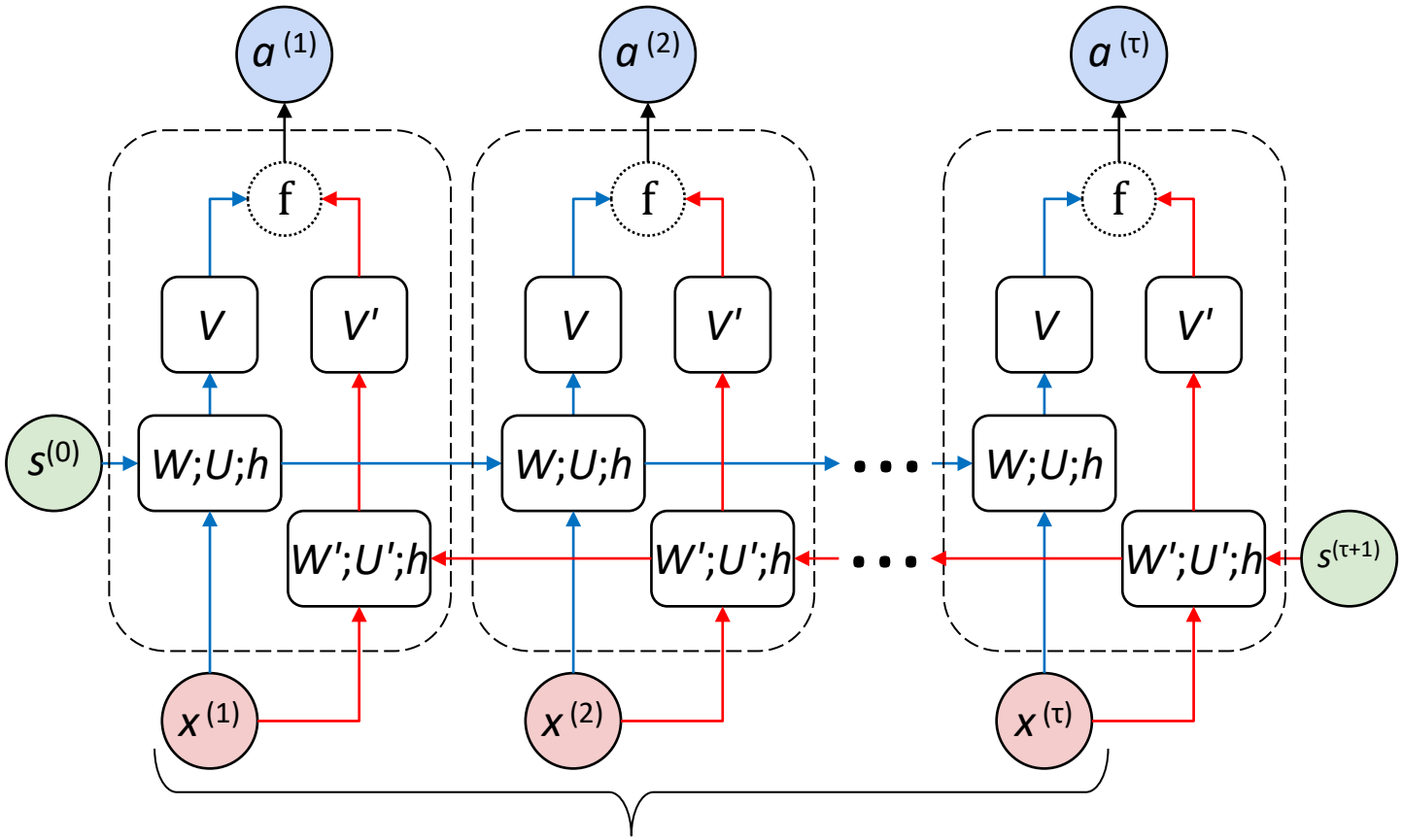


Глубокие RNN

1. Вставить глубокую сеть перед слоем внутреннего состояния (позволяет понимать временную структуру информации, например, выделять слова при распознавании речи)
2. Вставить глубокую сеть после слоя внутреннего состояния (позволяет выявлять закономерности, не зависящие от времени)
3. Использовать глубокую сеть вместо *tanh* (фрагментация изображений)
4. Использовать выходы одной RNN в качестве входов другой RNN (каждый слой действует в своем масштабе времени)



Двунаправленные RNN



Окно наблюдения длиной τ

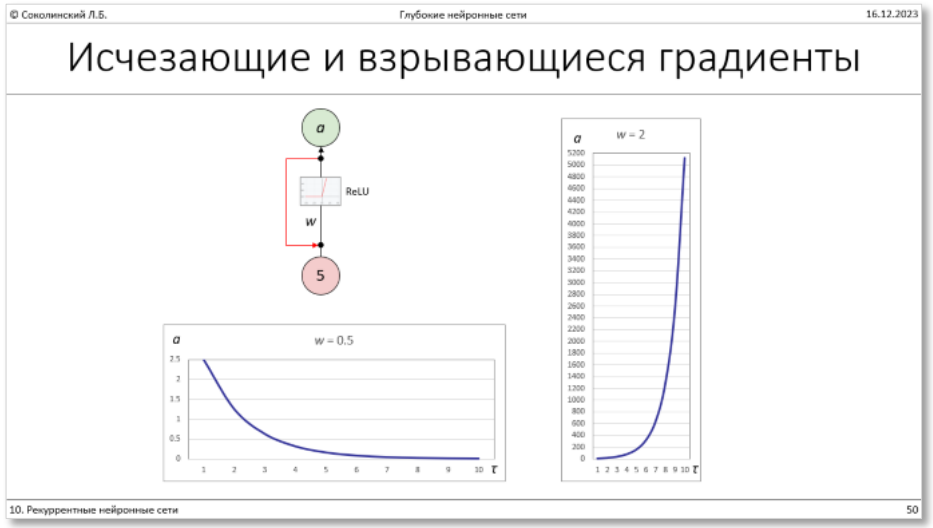
Позволяют получить два состояния, отражающие контекст как слева, так и справа для каждого элемента последовательности

Пример: разметка слов в предложении по частям речи

Недостатки RNN

- **Не позволяют надолго запоминать информацию во временной последовательности: влияние текущего состояния на будущие состояния экспоненциально затухает**

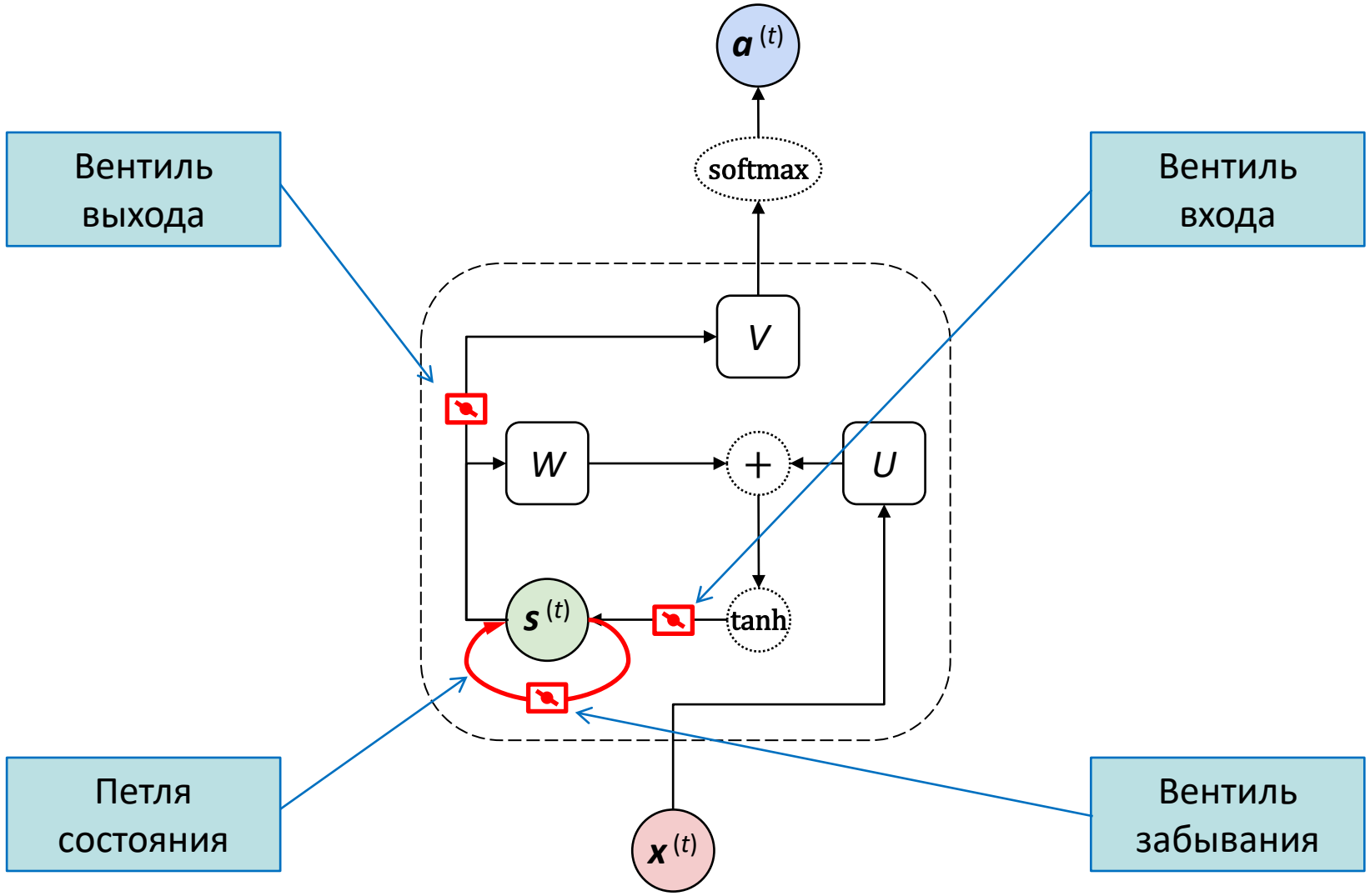
- Градиенты, распространяющиеся через много одинаковых слоев, либо исчезают (в большинстве случаев), либо начинают взрывообразно расти (редко, но с большим уроном для обучения)



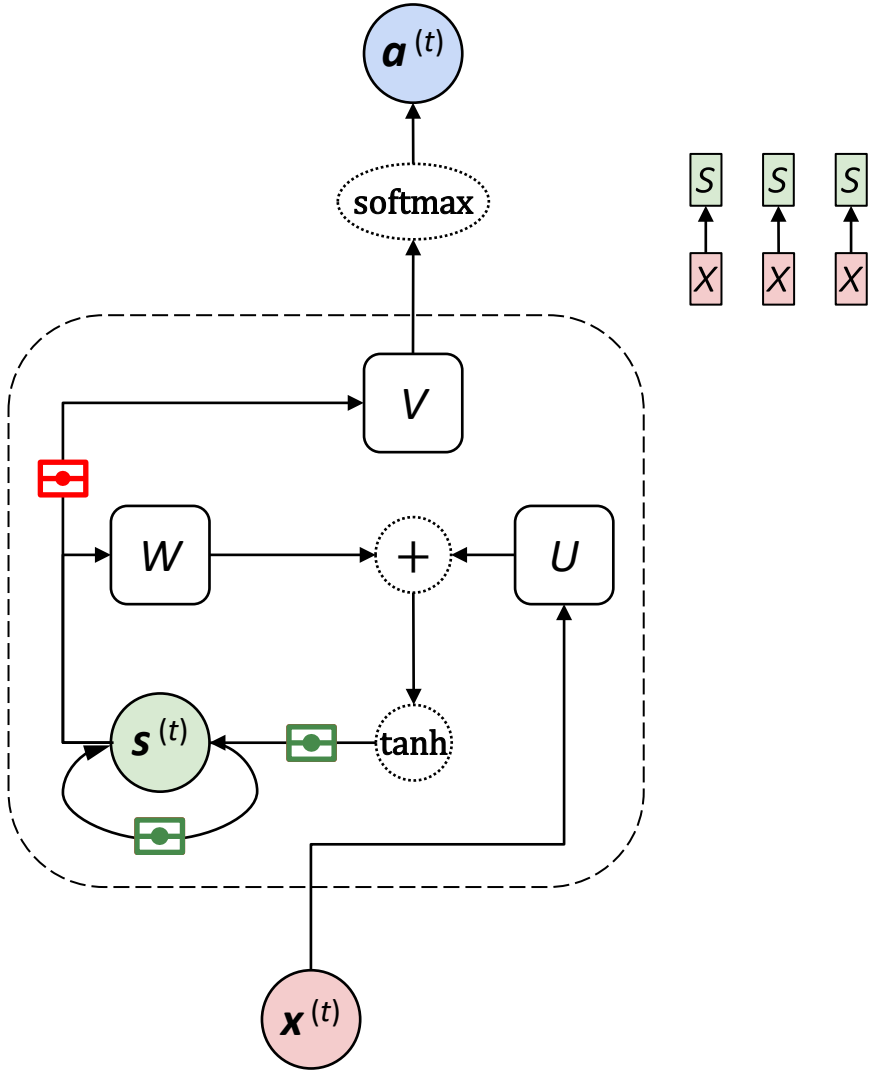
- Высокая вычислительная сложность: не удается обучать RNN с окном наблюдения размером 10 и более

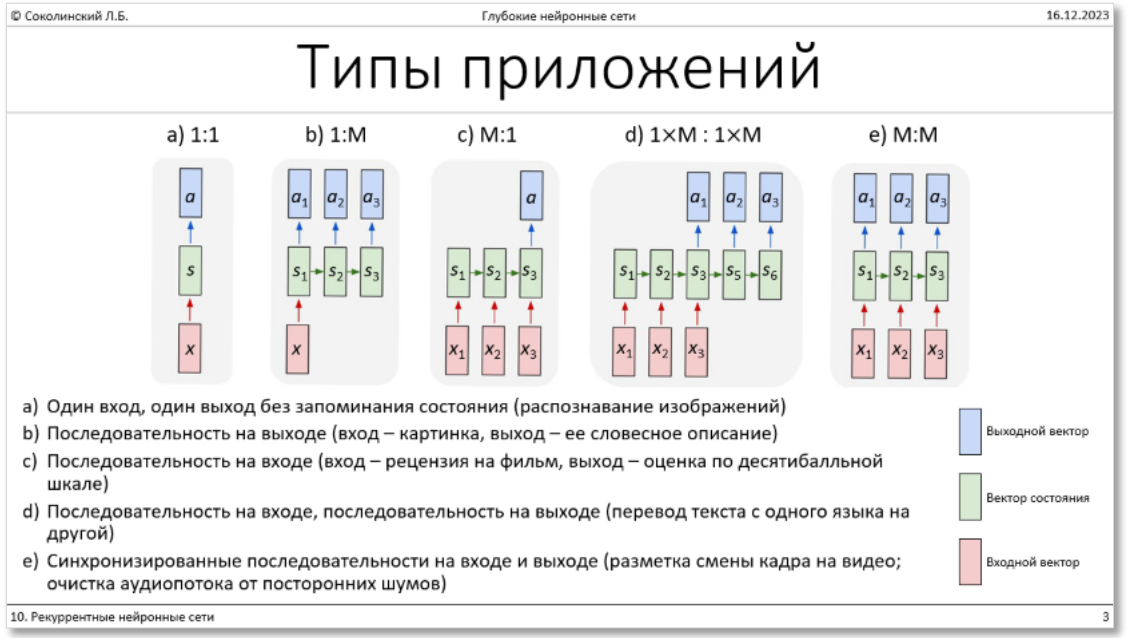
Как преодолеть забывчивость RNN?

Идея! Вентильные РНС (Gated RNN)

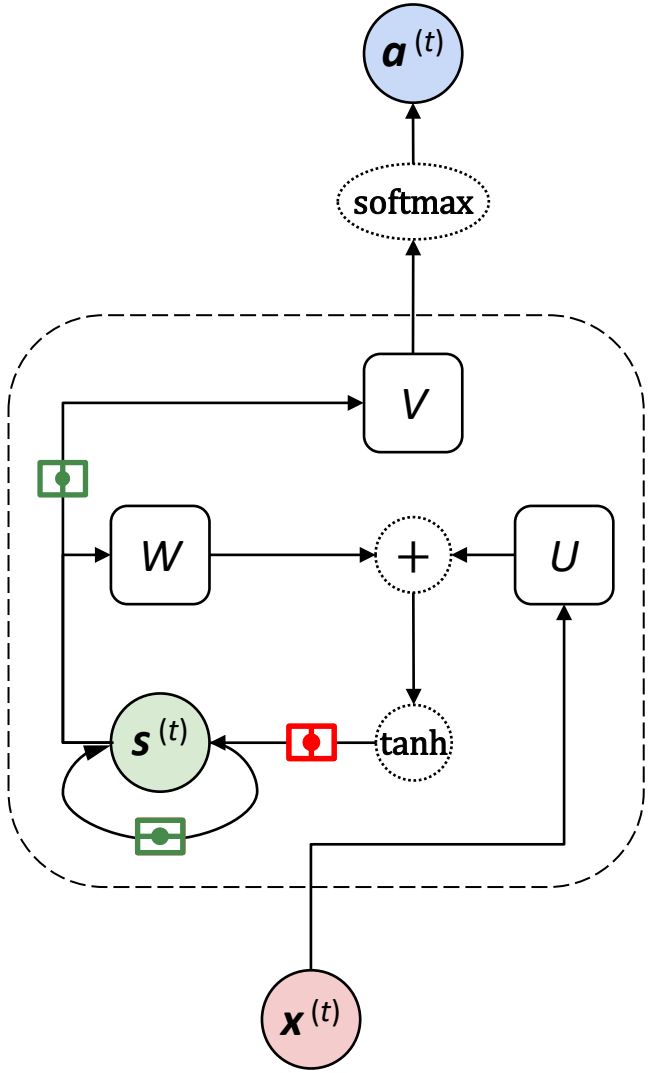


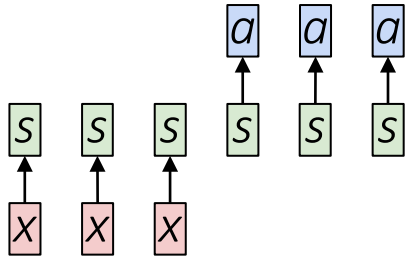
Пример: синхронный машинный перевод

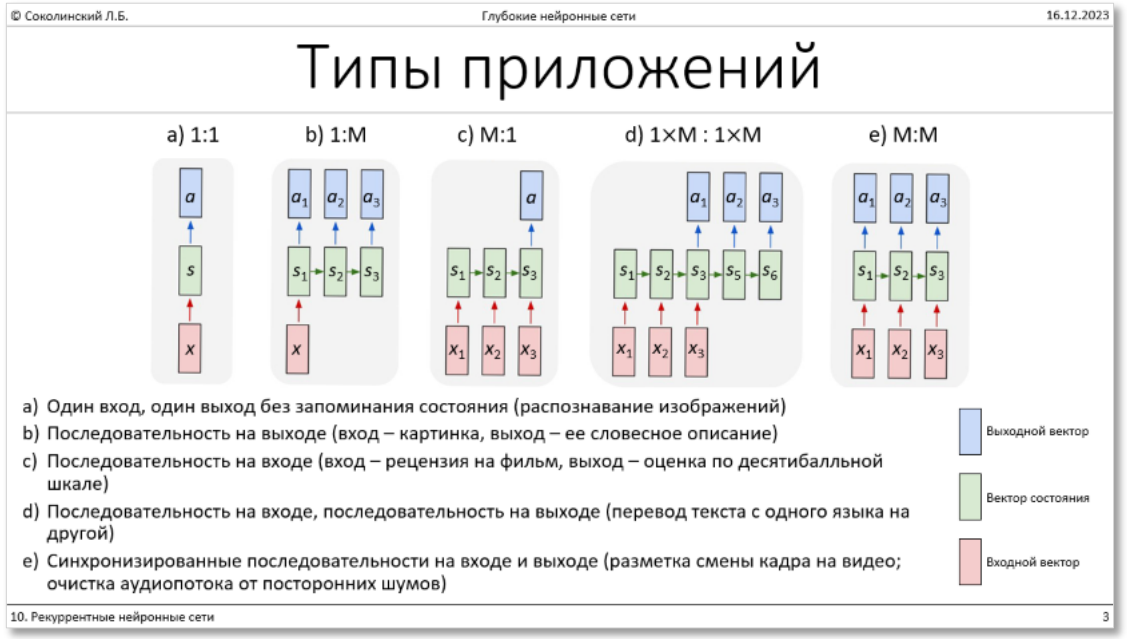




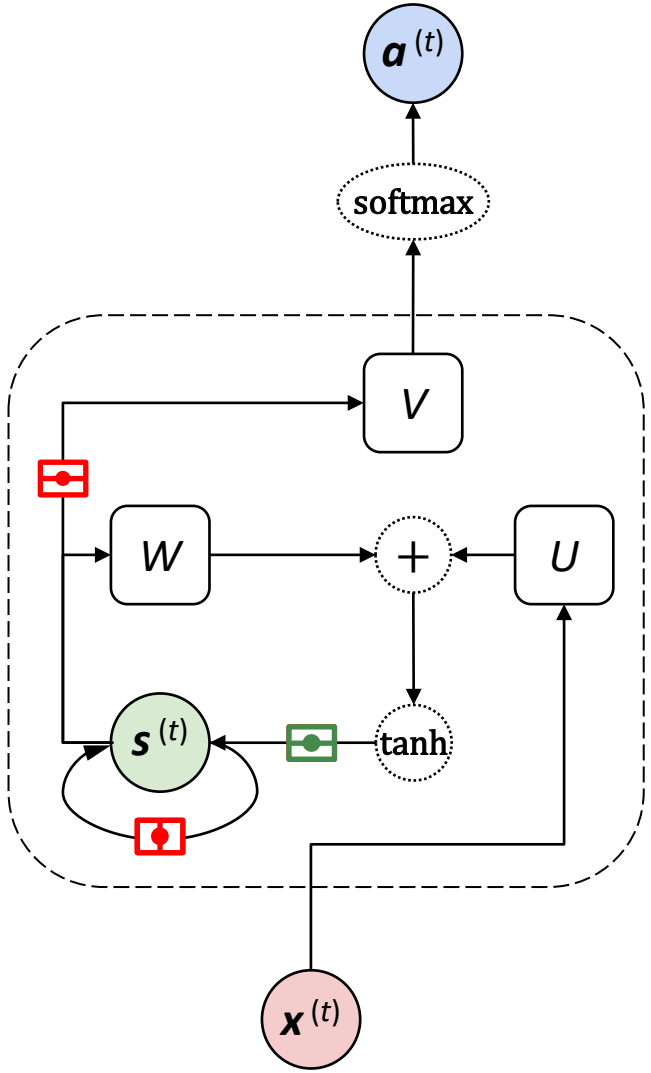
Пример: синхронный машинный перевод

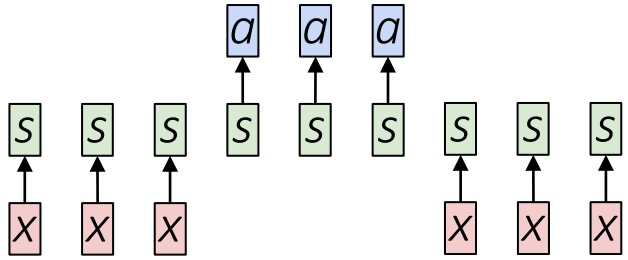


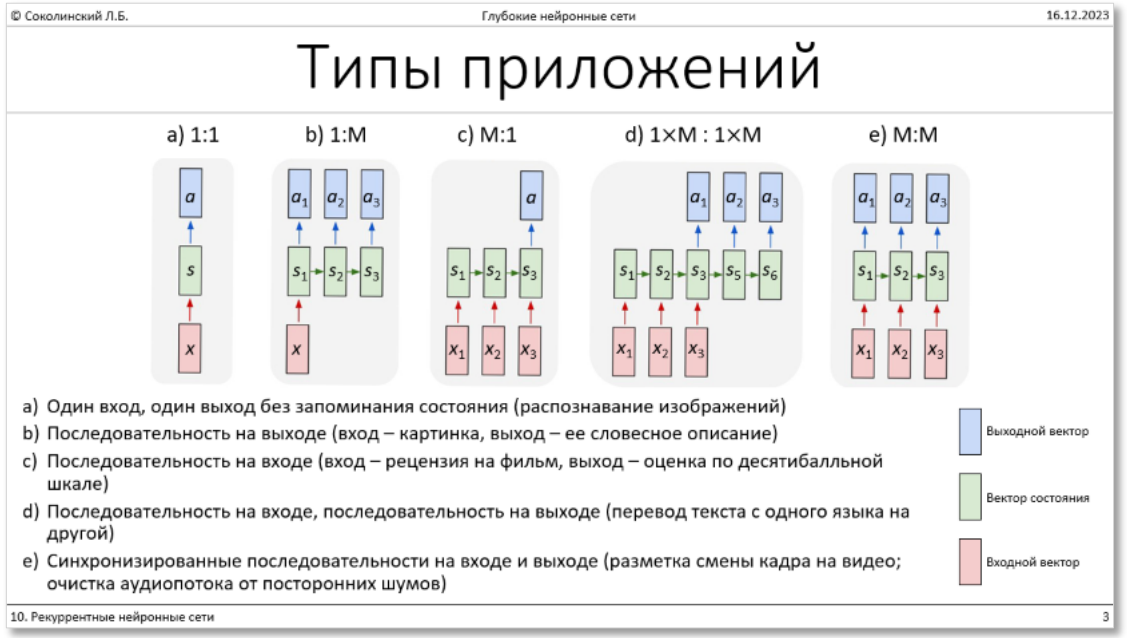




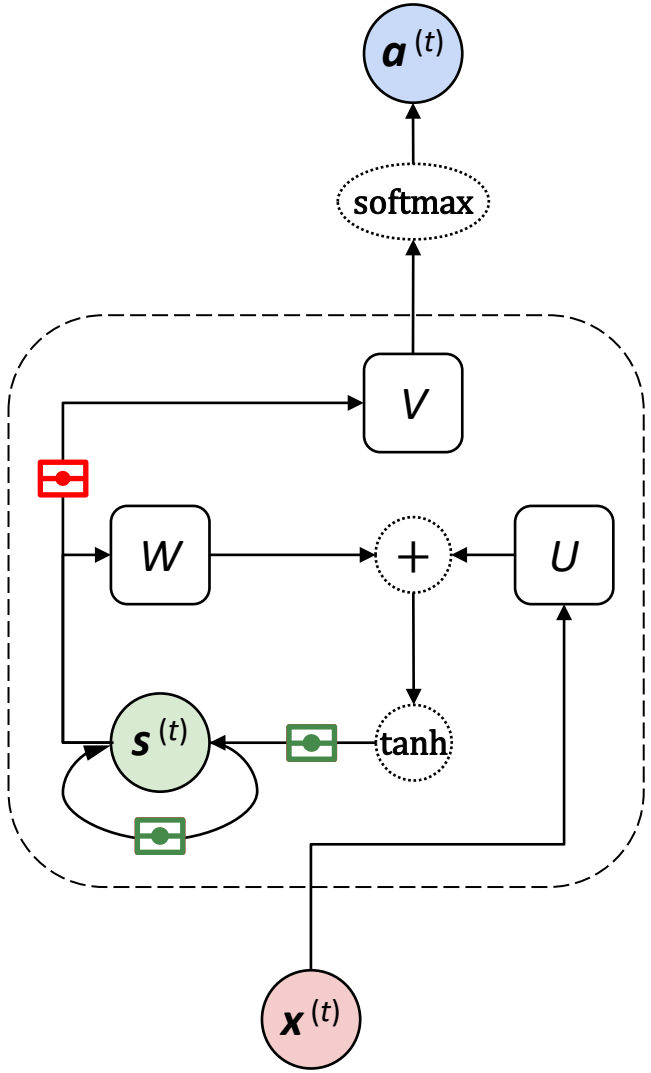
Пример: синхронный машинный перевод

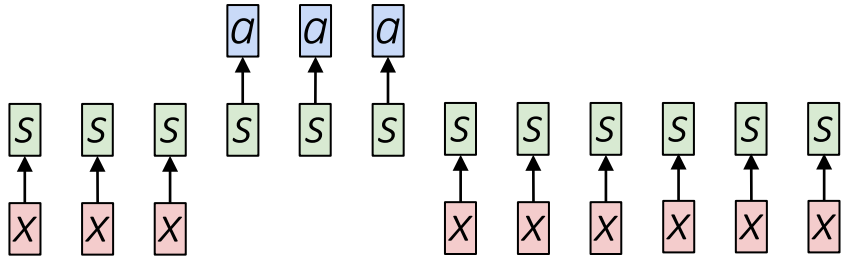


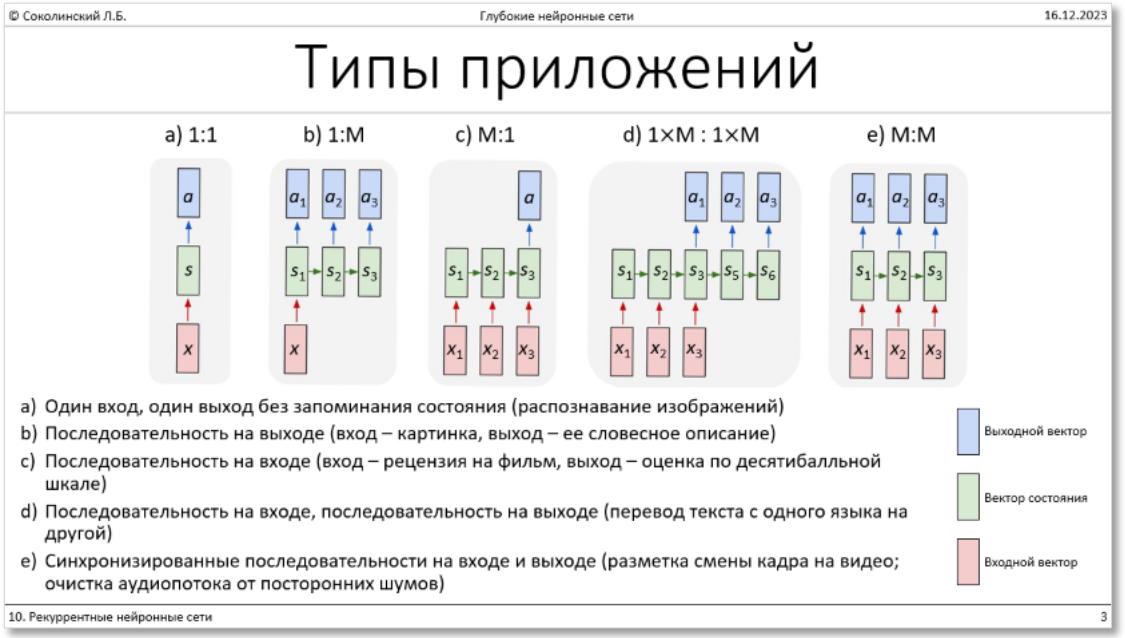




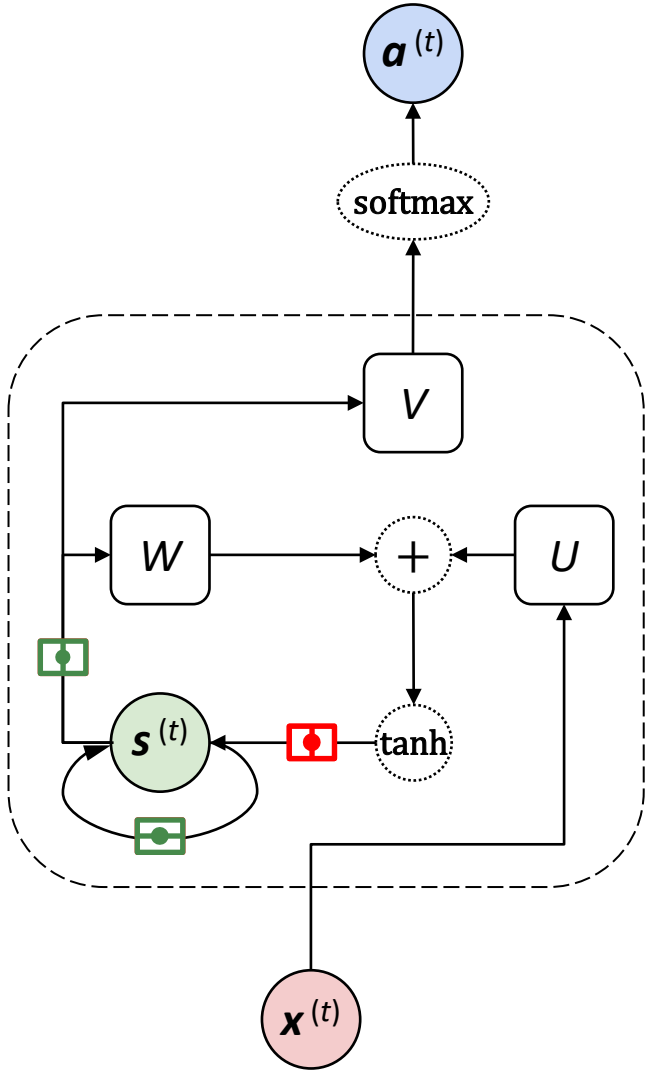
Пример: синхронный машинный перевод

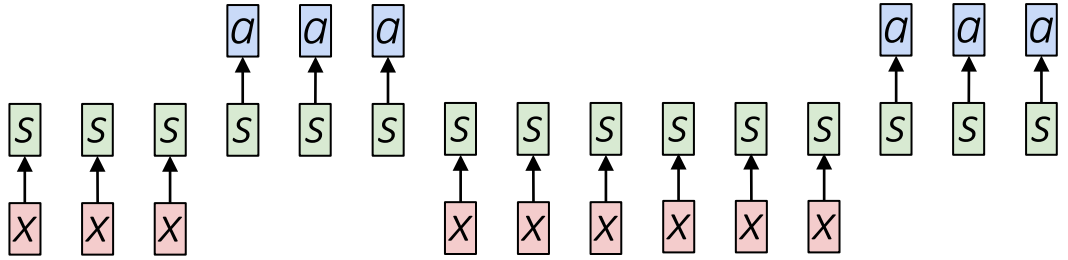


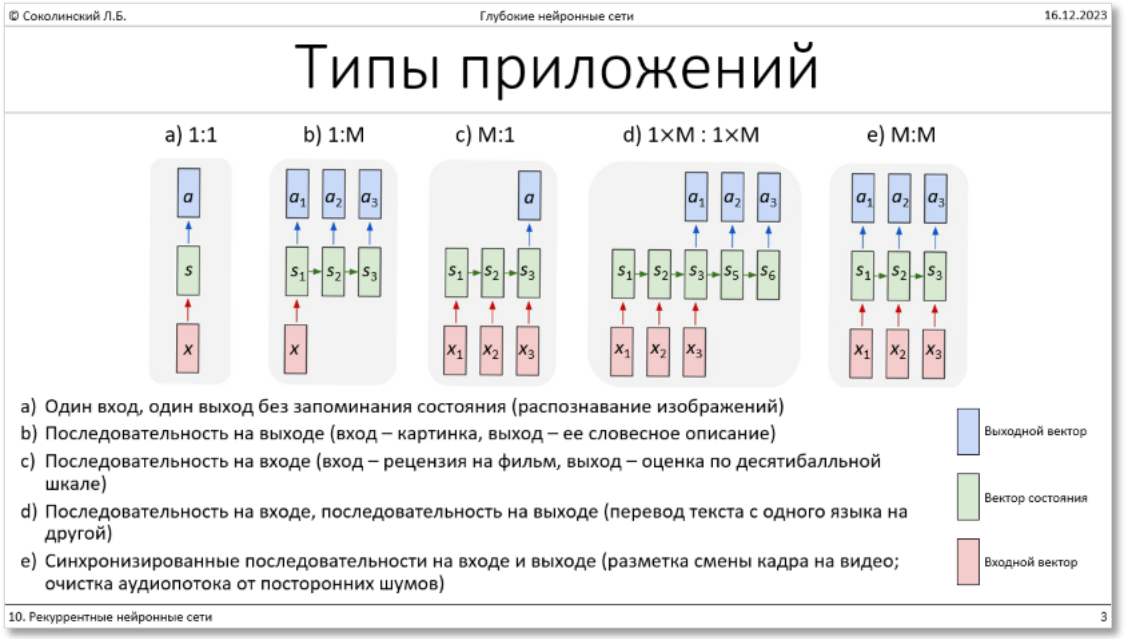




Пример: синхронный машинный перевод

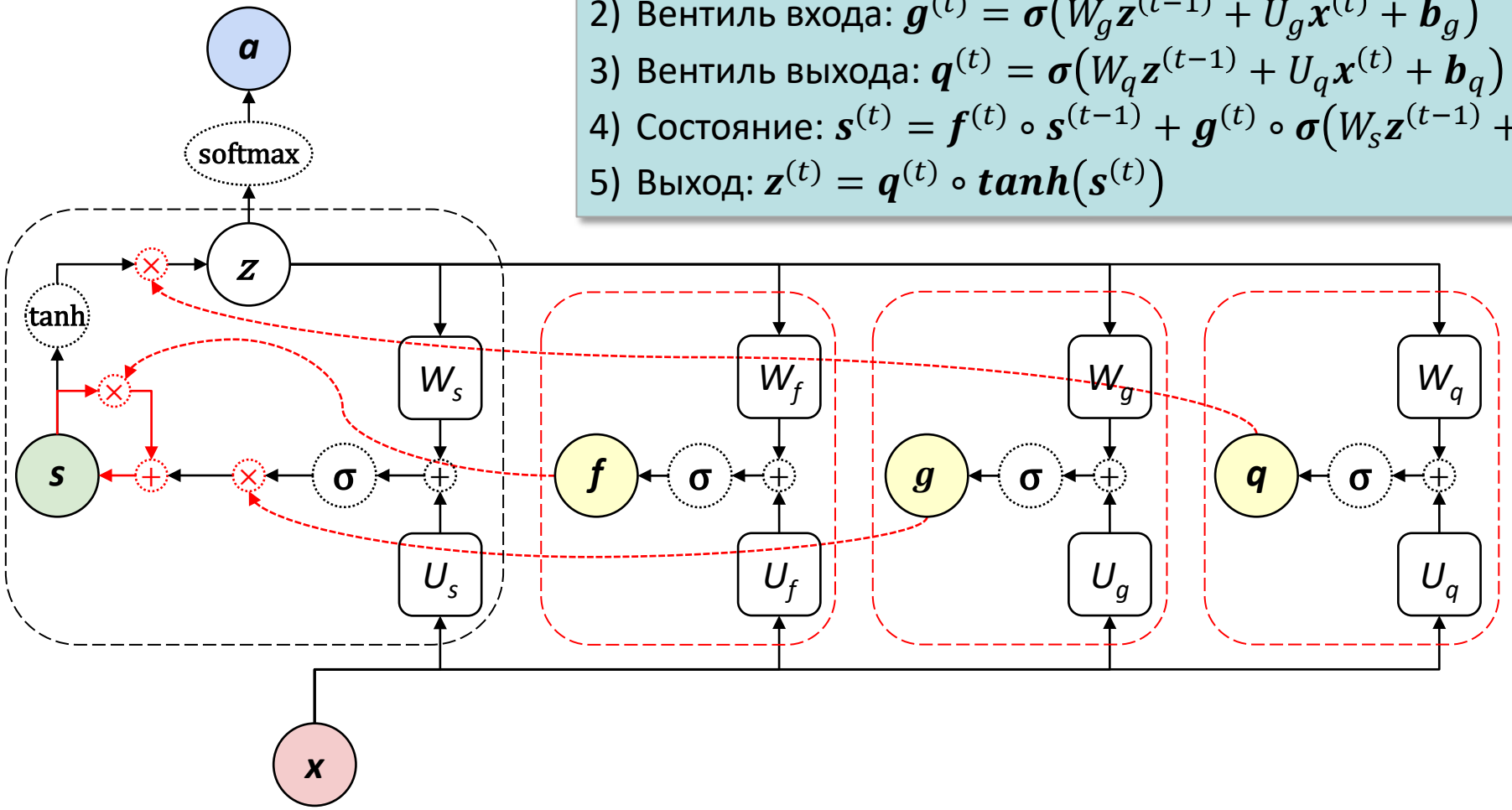






Модель долгой краткосрочной памяти (long short-term memory – LSTM)

1) Вентиль забывания: $\boldsymbol{f}^{(t)} = \boldsymbol{\sigma}\left(W_f \boldsymbol{z}^{(t-1)} + U_f \boldsymbol{x}^{(t)} + \boldsymbol{b}_f\right)$
2) Вентиль входа: $\boldsymbol{g}^{(t)} = \boldsymbol{\sigma}\left(W_g \boldsymbol{z}^{(t-1)} + U_g \boldsymbol{x}^{(t)} + \boldsymbol{b}_g\right)$
3) Вентиль выхода: $\boldsymbol{q}^{(t)} = \boldsymbol{\sigma}\left(W_q \boldsymbol{z}^{(t-1)} + U_q \boldsymbol{x}^{(t)} + \boldsymbol{b}_q\right)$
4) Состояние: $\boldsymbol{s}^{(t)} = \boldsymbol{f}^{(t)} \circ \boldsymbol{s}^{(t-1)} + \boldsymbol{g}^{(t)} \circ \boldsymbol{\sigma}\left(W_s \boldsymbol{z}^{(t-1)} + U_s \boldsymbol{x}^{(t)} + \boldsymbol{b}_s\right)$
5) Выход: $\boldsymbol{z}^{(t)} = \boldsymbol{q}^{(t)} \circ \boldsymbol{tanh}\left(\boldsymbol{s}^{(t)}\right)$



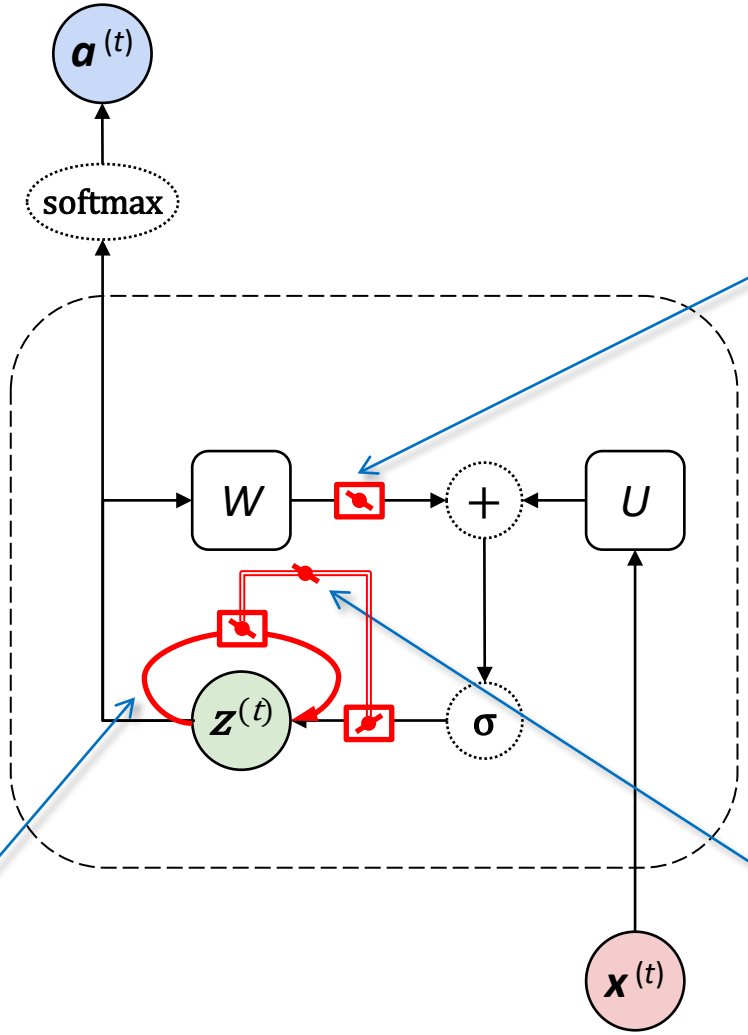
Применения LSTM

- Большинство применений RNN используют модель LSTM
- Ключевыми компонентами LSTM являются вентиль забывания и функция активации *tanh* перед вентилем выхода (без вентиля забывания качество сразу сильно падает, а без функции активации на выходе этот самый выход теоретически может расти неограниченно, что приводит к нежелательным эффектам)
- Основной недостаток LSTM – большое количество матриц весов (8 против 3 у Simple RNN)

Идея!

Вентильный рекуррентный модуль (Gated recurrent unit – GRU)

Объединить вентили входа и забывания в один *вентиль обновления*

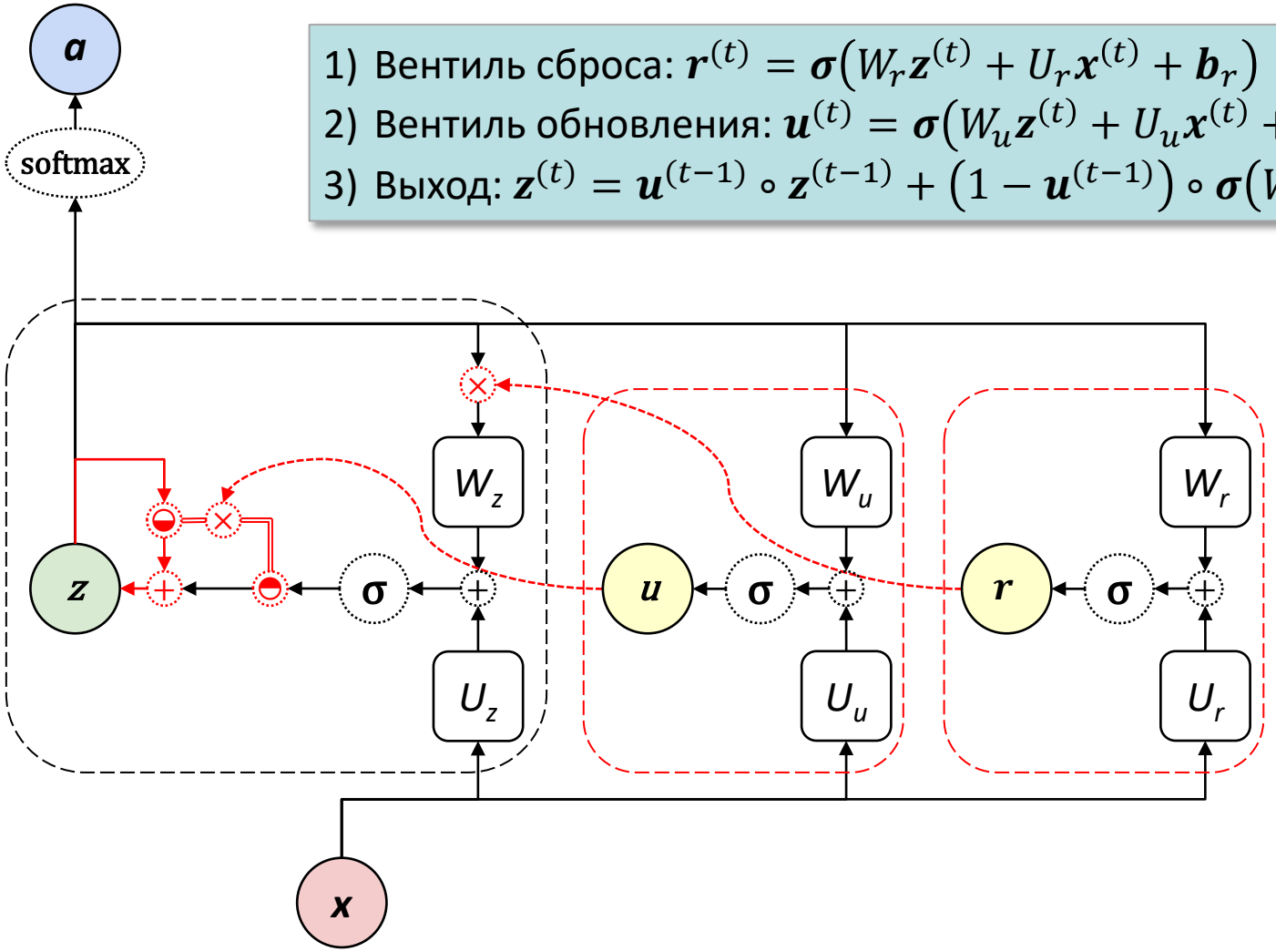


Вентиль сброса

Петля состояния

Вентиль *обновления*

Модель GRU



1) Вентиль сброса: $\boldsymbol{r}^{(t)} = \boldsymbol{\sigma}\left(W_r \boldsymbol{z}^{(t)} + U_r \boldsymbol{x}^{(t)} + \boldsymbol{b}_r\right)$
2) Вентиль обновления: $\boldsymbol{u}^{(t)} = \boldsymbol{\sigma}\left(W_u \boldsymbol{z}^{(t)} + U_u \boldsymbol{x}^{(t)} + \boldsymbol{b}_u\right)$
3) Выход: $\boldsymbol{z}^{(t)} = \boldsymbol{u}^{(t-1)} \circ \boldsymbol{z}^{(t-1)} + \left(1 - \boldsymbol{u}^{(t-1)}\right) \circ \boldsymbol{\sigma}\left(W_z\left(\boldsymbol{r}^{(t-1)} \circ \boldsymbol{z}^{(t-1)}\right) + U_z \boldsymbol{x}^{(t-1)} + \boldsymbol{b}_z\right)$

Применения модели GRU

- Практика показывает, что модель GRU практически всегда работает так же хорошо, как LSTM, а параметров у нее гораздо меньше: 6 матриц весов против 8 в LSTM
- GRU быстрее обучается
- GRU требует меньше памяти, что позволяет расширить окно наблюдения

Задача: вычислить $\boldsymbol{a}^{(2)}$

$$\boldsymbol{s}^{(0)} = (0, 0)$$

$$\boldsymbol{x}^{(1)} = (4, 6);$$

$$\boldsymbol{x}^{(2)} = (9, 7)$$

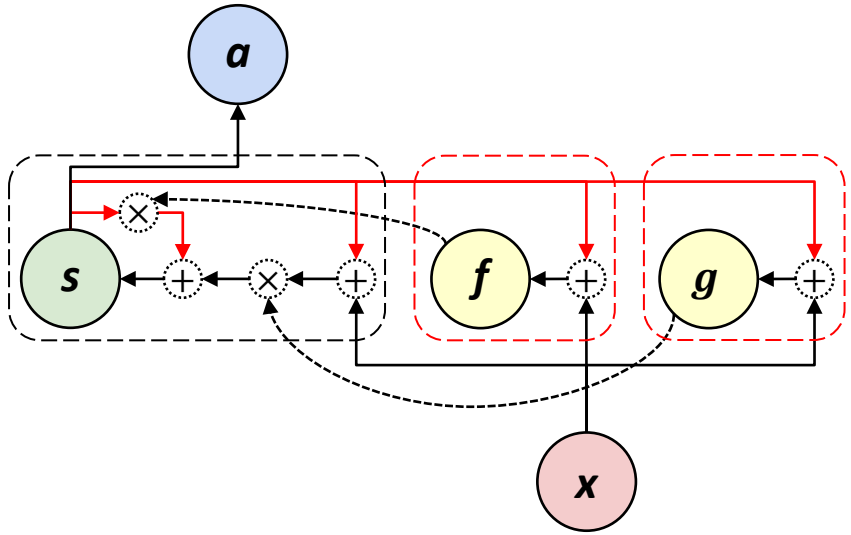
$$\boldsymbol{f}^{(1)} = \boldsymbol{g}^{(1)} = \boldsymbol{x}^{(1)} + \boldsymbol{s}^{(0)} = (\,4\,,6\,)$$

$$\boldsymbol{s}^{(1)} = \left(\boldsymbol{x}^{(1)} + \boldsymbol{s}^{(0)}\right) \circ \boldsymbol{g}^{(1)} + \boldsymbol{s}^{(0)} \circ \boldsymbol{f}^{(1)} = (\,16\,,36\,)$$

$$\boldsymbol{f}^{(2)} = \boldsymbol{g}^{(2)} = \boldsymbol{x}^{(2)} + \boldsymbol{s}^{(1)} = (\,25\,,43\,)$$

$$\boldsymbol{s}^{(2)} = \left(\boldsymbol{x}^{(2)} + \boldsymbol{s}^{(1)}\right) \circ \boldsymbol{g}^{(2)} + \boldsymbol{s}^{(1)} \circ \boldsymbol{f}^{(2)} = (\,1025\,,3397\,)$$

$$\boldsymbol{a}^{(2)} = \boldsymbol{s}^{(2)} = (\,1025\,,3397\,)$$



Конец лекции 10

Вспомогательные слайды

Использование в качестве функции потерь функции перекрестной энтропии

$$C = -\sum_j y_j \ln a_j^L$$

Свойства функции потерь:

1) $C \geq 0$

2) Минимум C достигается при $\boldsymbol{a}^L = \boldsymbol{y}$:

$$\min_{\boldsymbol{a}^L} C = -\sum_j y_j \ln y_j$$

Вычисление ошибки δ_i^L для выходного слоя

$$\delta_i^L = \frac{\partial C}{\partial z_i^L} = \sum_{j=1}^{n} \frac{\partial C}{\partial a_j^L} \cdot \frac{\partial a_j^L}{\partial z_i^L} = \sum_{j=1}^{n} -\frac{y_j}{a_j^L} \cdot \frac{\partial a_j^L}{\partial z_i^L} = -\sum_{j=1}^{n} \frac{y_j}{a_j^L} \cdot \frac{\partial a_j^L}{\partial z_i^L}$$

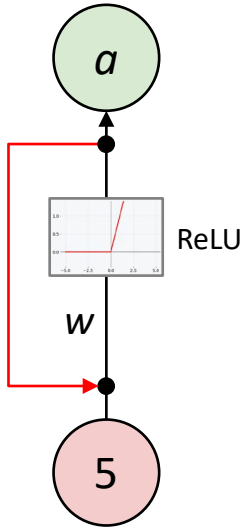
$$= -\frac{y_i}{a_i^L} \cdot \frac{\partial a_i^L}{\partial z_i^L} - \sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{y_j}{a_j^L} \cdot \frac{\partial a_j^L}{\partial z_i^L}$$

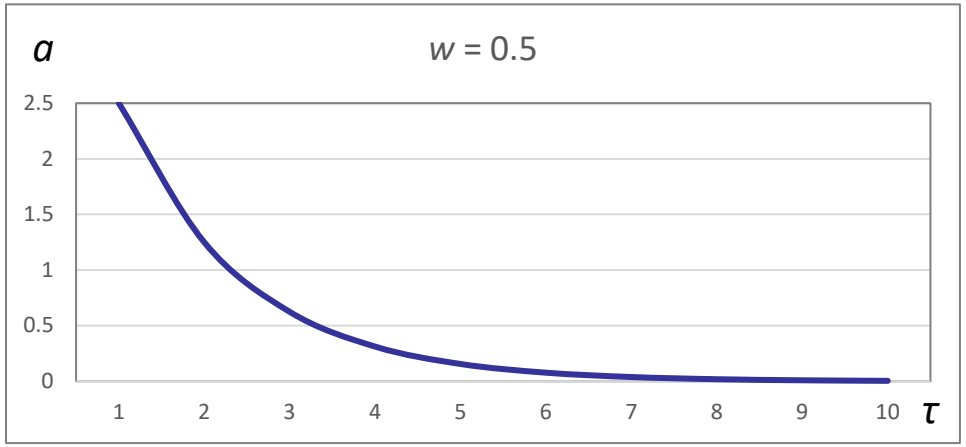
$$= -\frac{y_i}{a_i^L} a_i^L \left(1 - a_i^L\right) - \sum_{\substack{j=1 \\ j \neq i}}^{n} -\frac{y_j}{a_j^L} \cdot a_i^L a_j^L = -y_i + y_i a_i^L + \sum_{\substack{j=1 \\ j \neq i}}^{n} y_j a_i^L$$

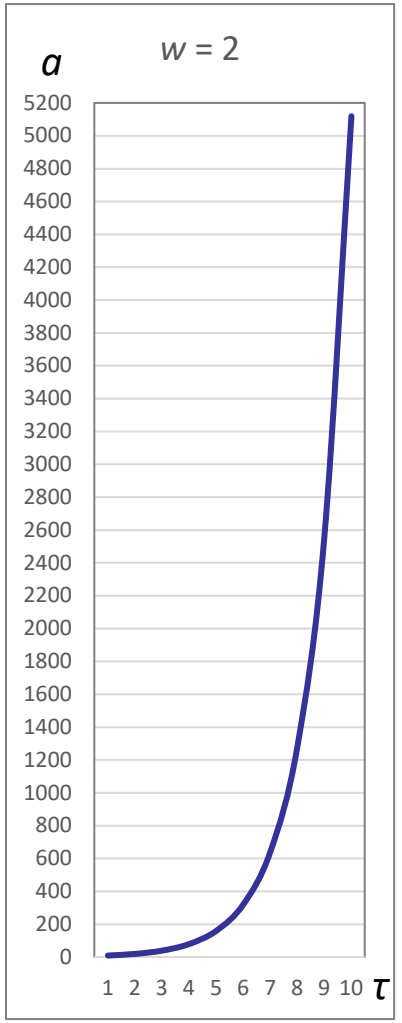
$$= -y_i + \sum_{j=1}^{n} y_j a_i^L = -y_i + a_i^L \sum_{j=1}^{n} y_j$$

$$= -y_i + a_i^L = a_i^L - y_i$$

Исчезающие и взрывающиеся градиенты







Simple RNN