



RNN - Рекуррентные нейронные сети

Преподаватель: Герард Костин

Зачем нужны RNN



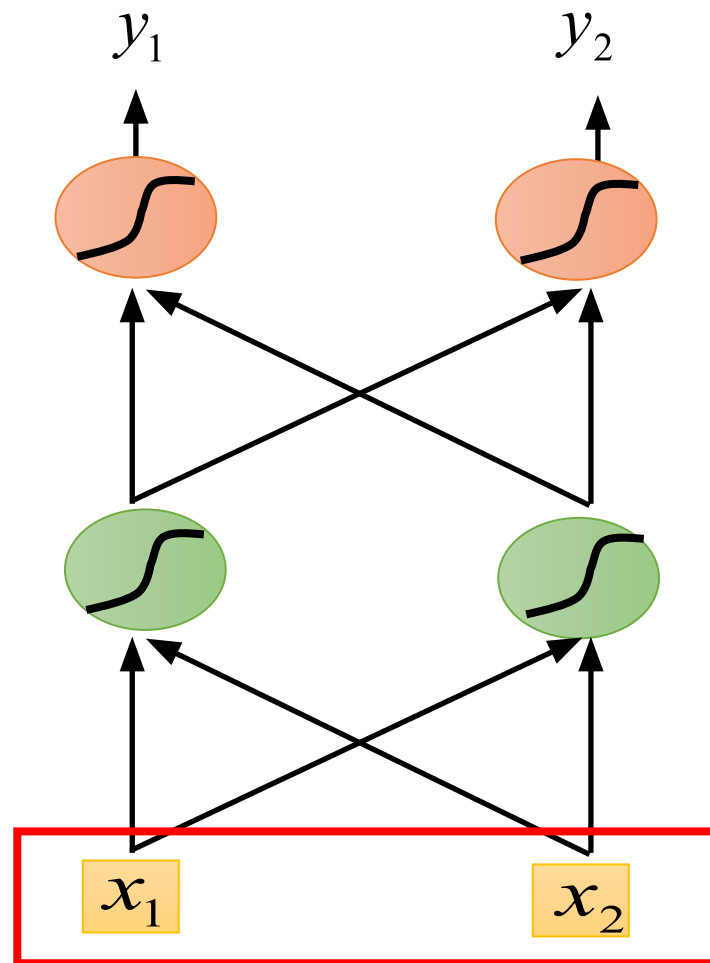
Пример использования

Как проблему бронирования
сетью Feedforward?

На входе: слово

(Каждое слово представлено в
виде вектора)

Taipei



Bag-of-word

Как представить слово в виде вектора?

vocab = {apple, bag, cat, dog, elephant}

вектор - это размер словаря.

Каждое измерение соответствует слову в vocab.

Размерность слова - 1,
остальные - 0.

apple = [1 0 0 0 0]

bag = [0 1 0 0 0]

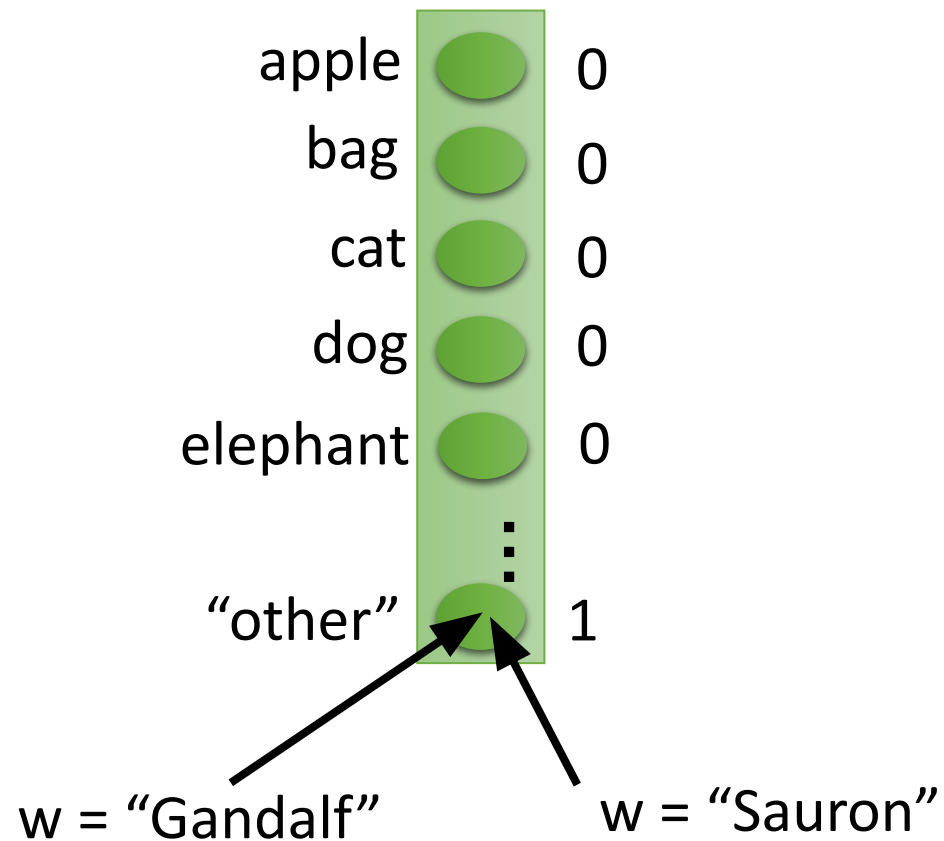
cat = [0 0 1 0 0]

dog = [0 0 0 1 0]

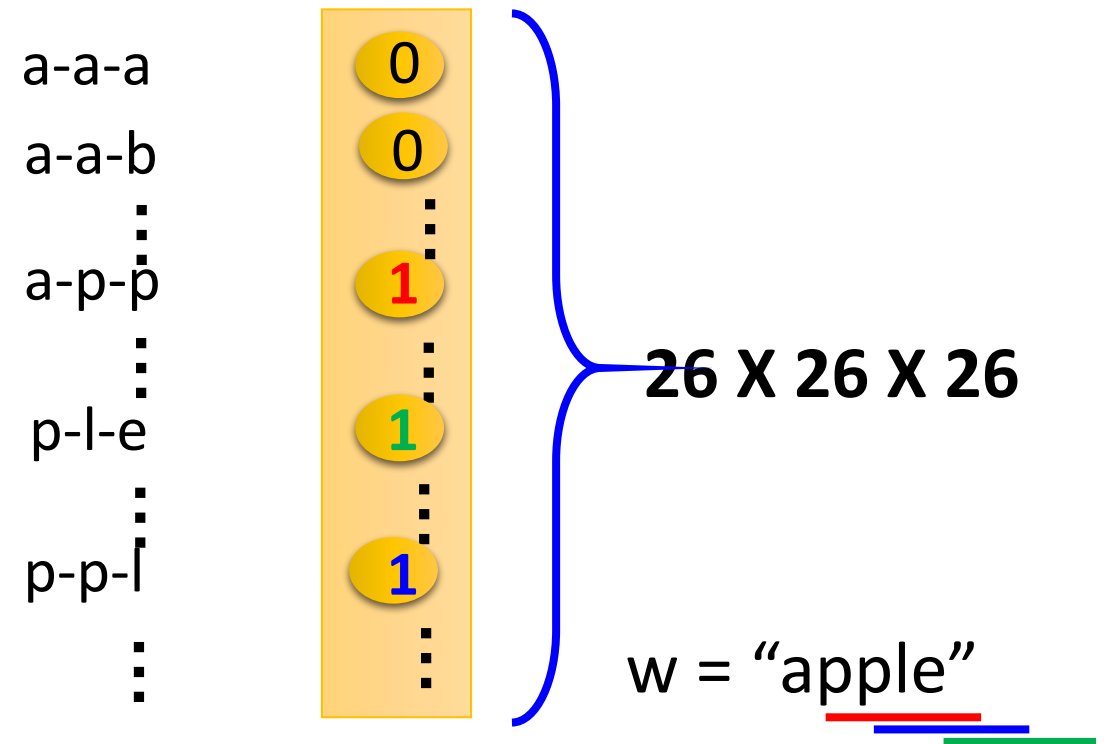
elephant = [0 0 0 0 1]

Другие методы

Dimension for "Other"



Word hashing



Пример использования

Как проблему бронирования
сетью Feedforward?

На входе: слово

(Каждое слово представлено в
виде вектора)

На Выходе:

Распределение

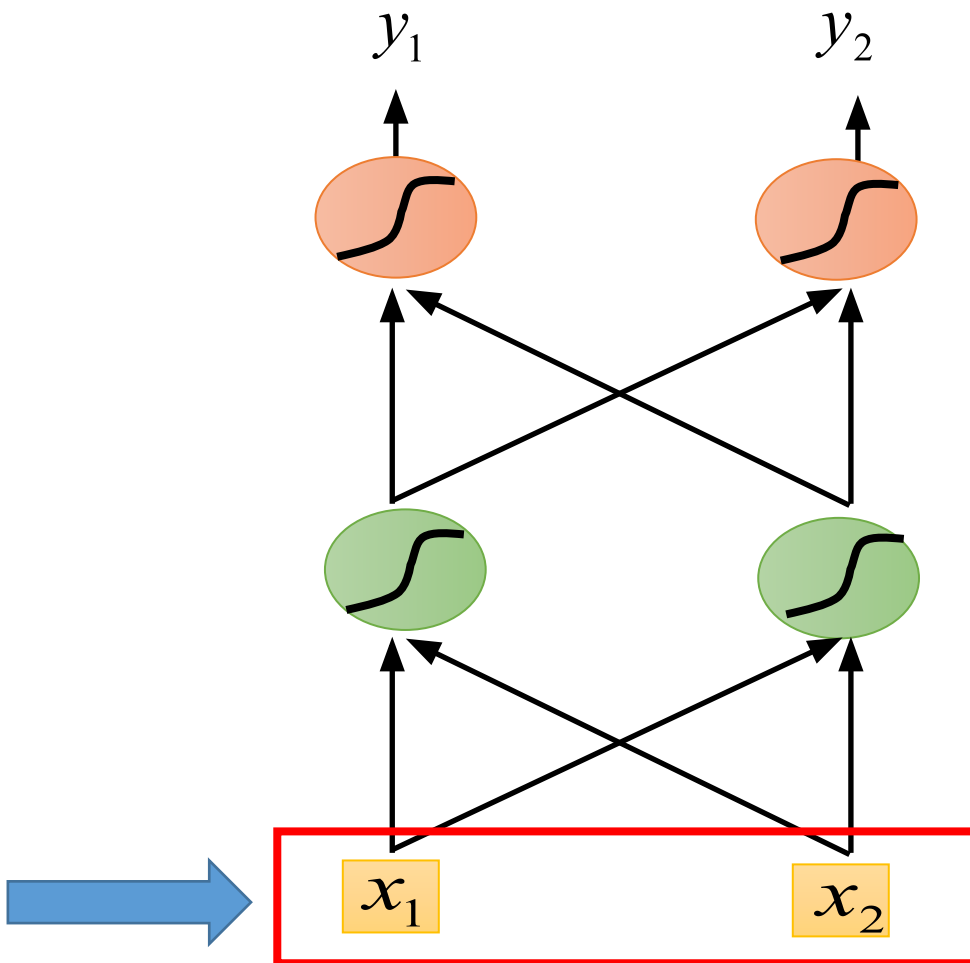
вероятности того, что

входное слово

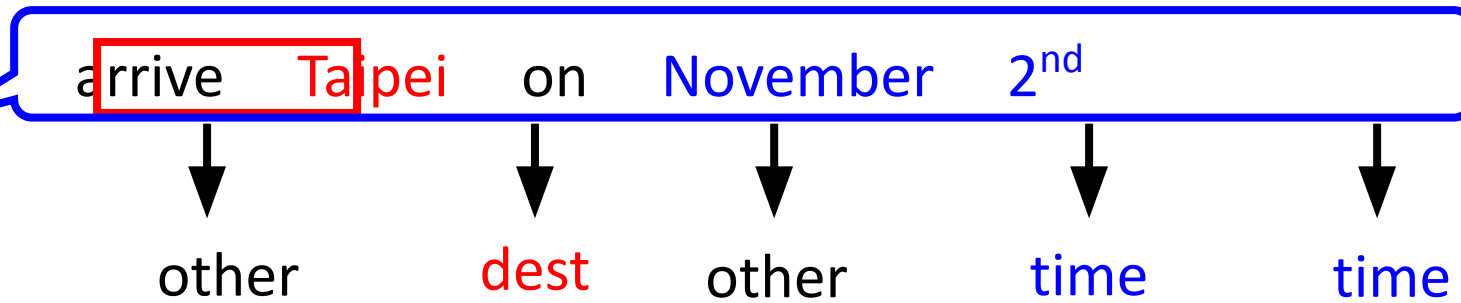
принадлежит искомому

классу

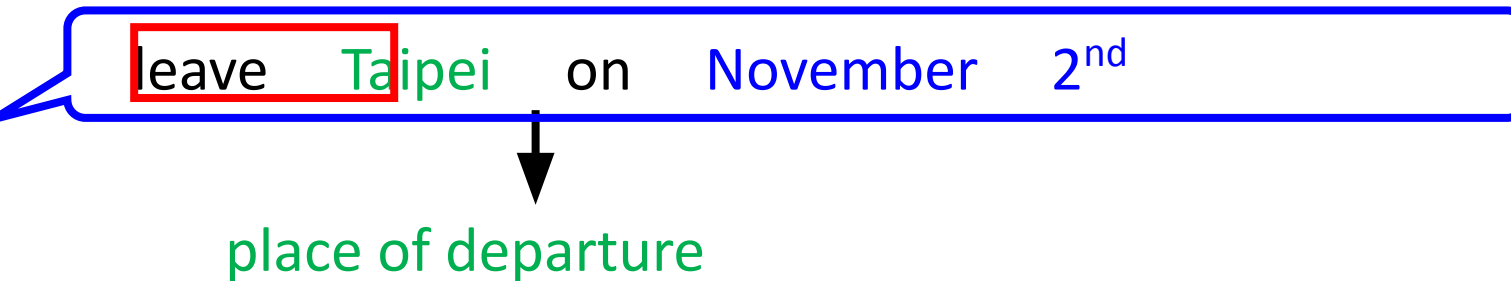
Taipei



Example Application

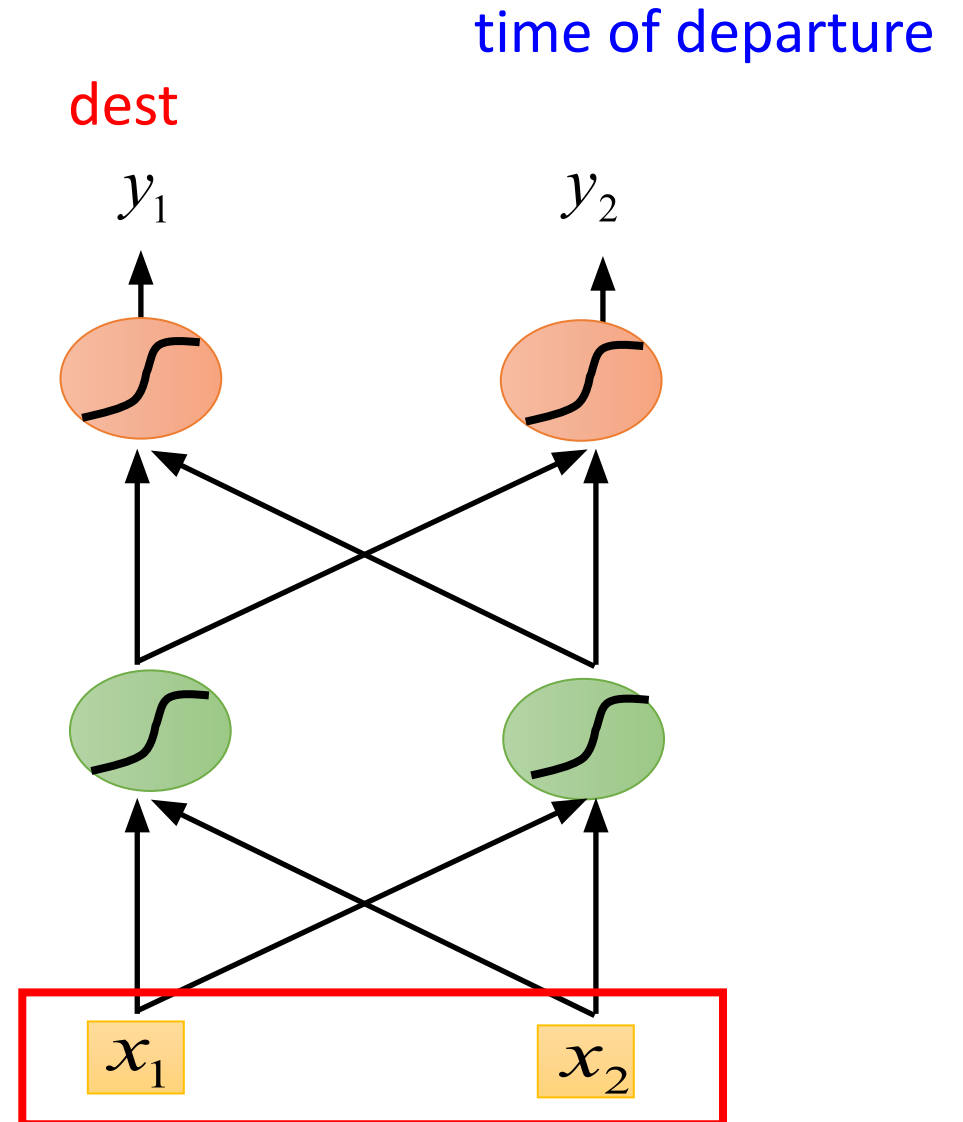


Проблема!



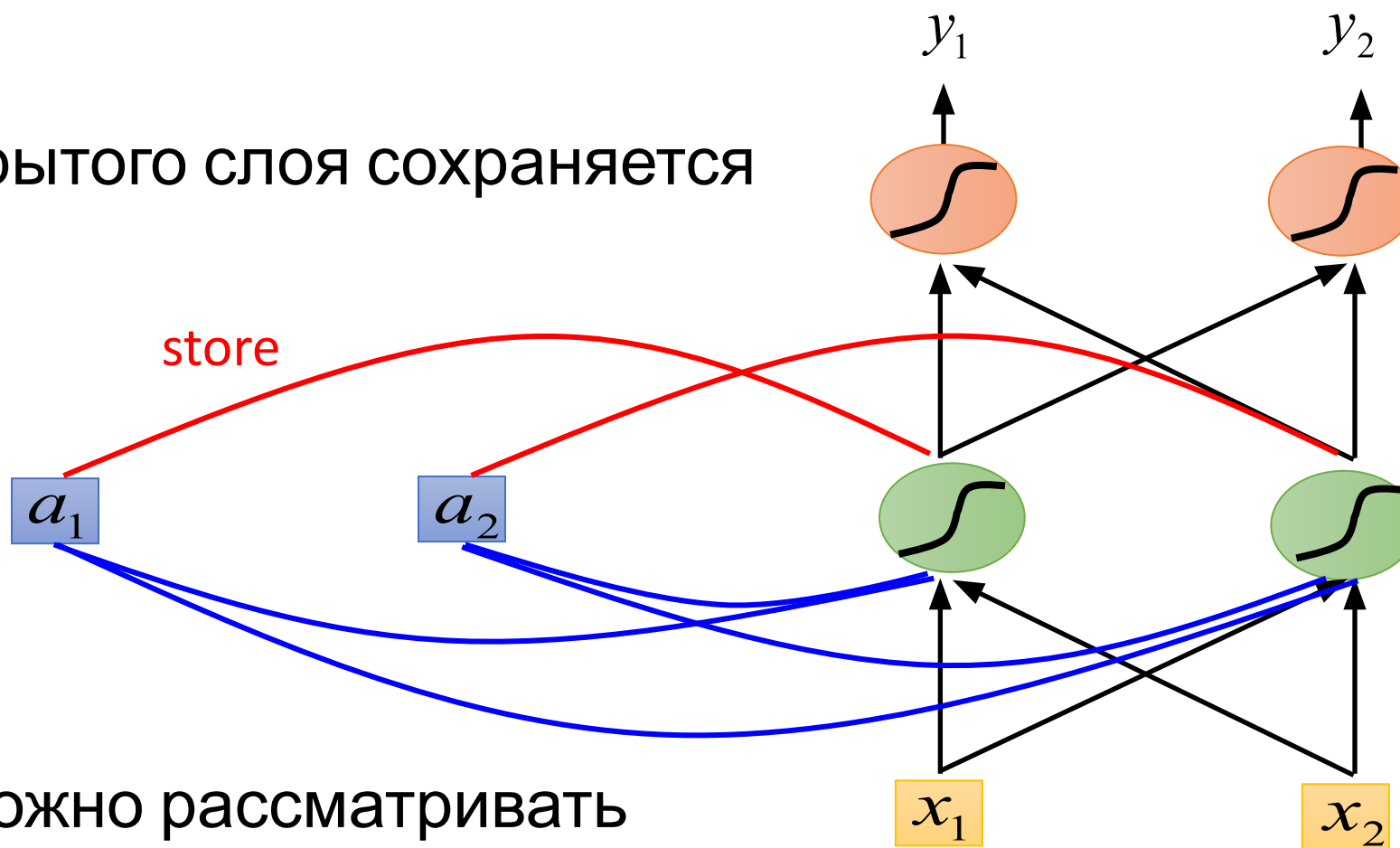
Нейронной сети нужна
память!

Taipei



Recurrent Neural Network (RNN)

Вывод скрытого слоя сохраняется в памяти.



Память можно рассматривать как еще один вход.

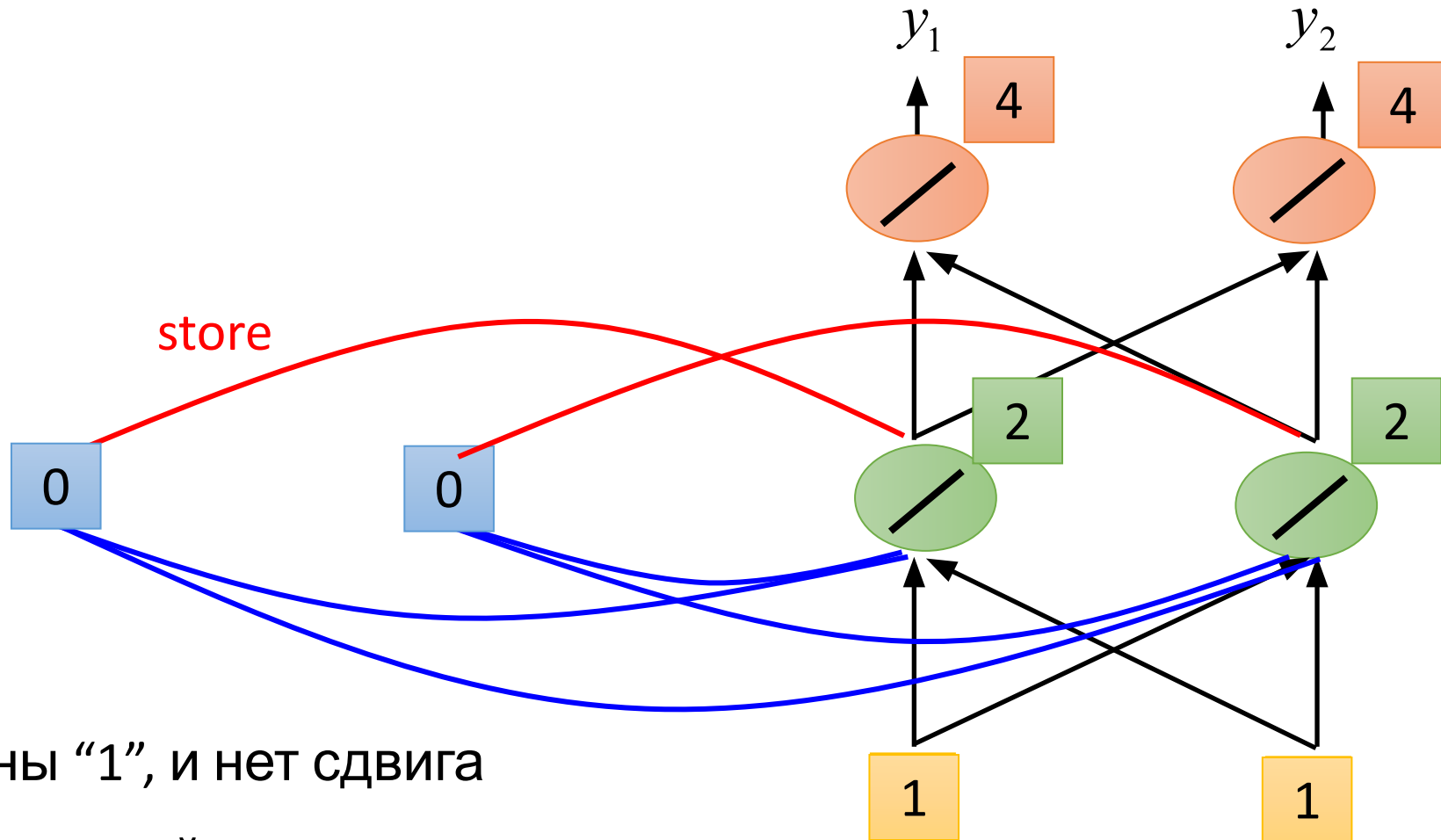
Вход:

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \end{bmatrix} \dots \dots$$

Выход:

$$\begin{bmatrix} 4 \\ 4 \end{bmatrix}$$

с учетом
начальных
значений



Все веса равны "1", и нет сдвига

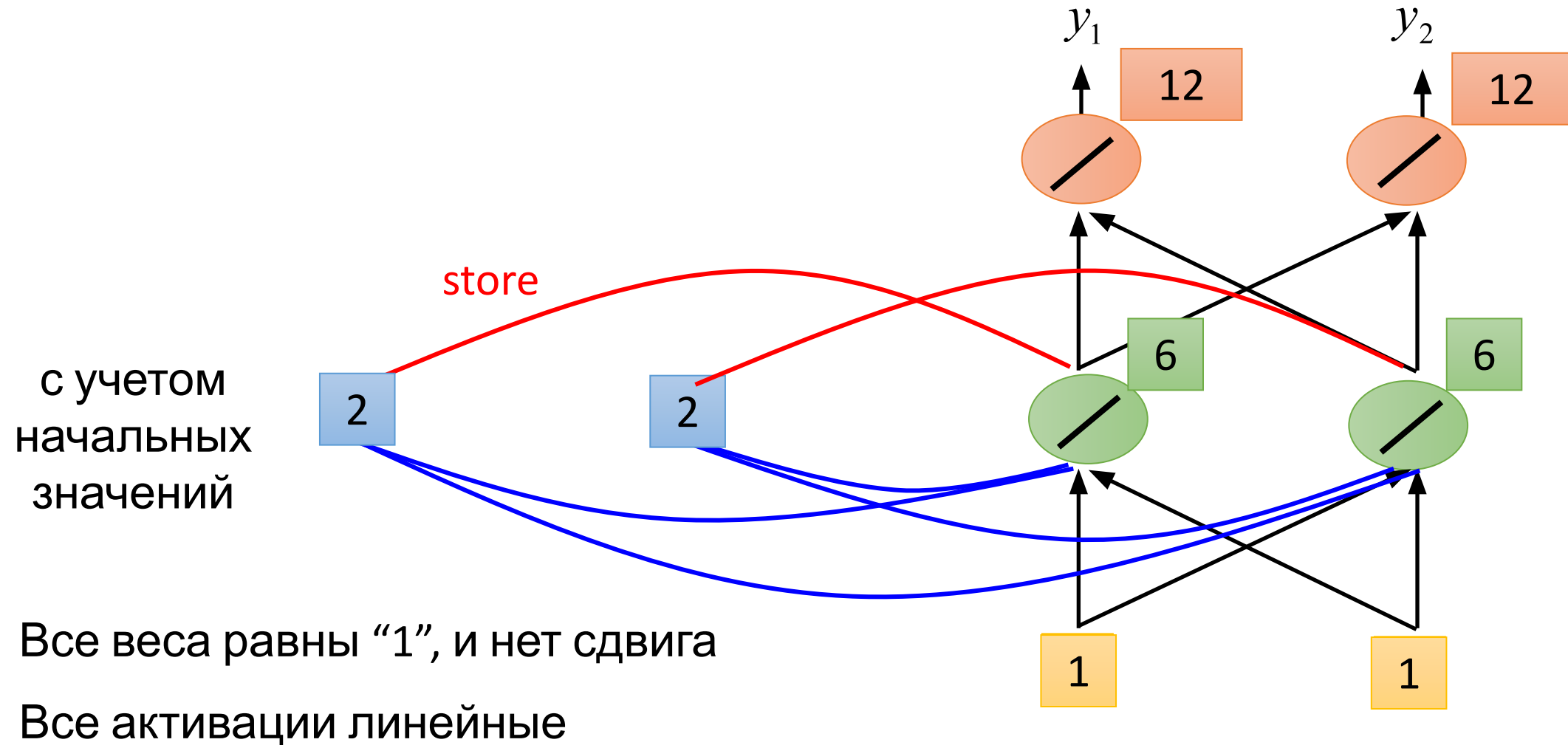
Все активации линейные

Вход:

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \end{bmatrix} \dots \dots$$

Выход:

$$\begin{bmatrix} 4 \\ 4 \end{bmatrix} \begin{bmatrix} 12 \\ 12 \end{bmatrix}$$



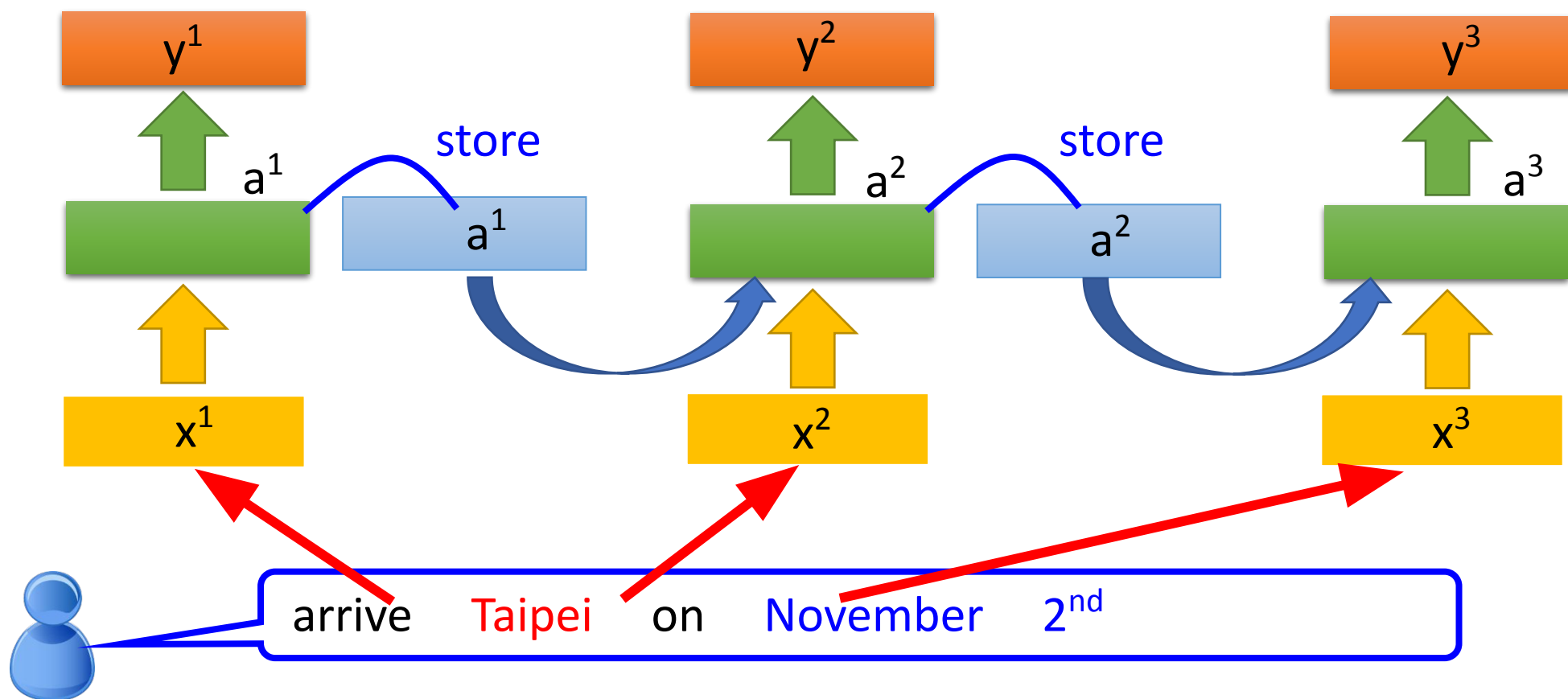
RNN

Одна и та же сеть используется снова и снова.

Вероятность “arrive”

Вероятность “Taipei”

Вероятность “on”



RNN

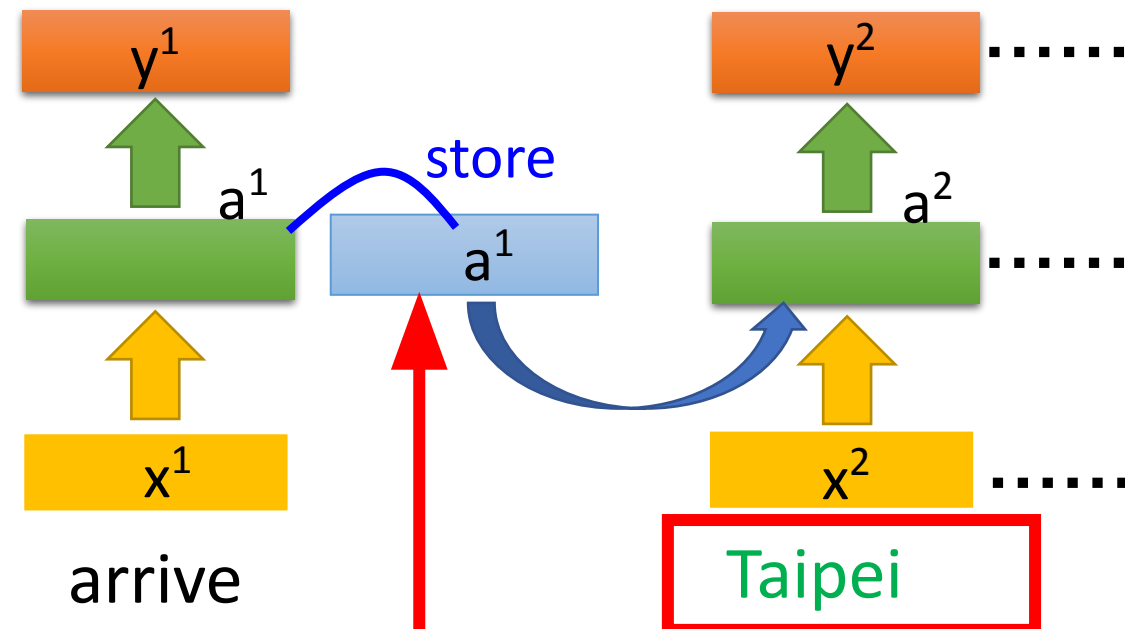
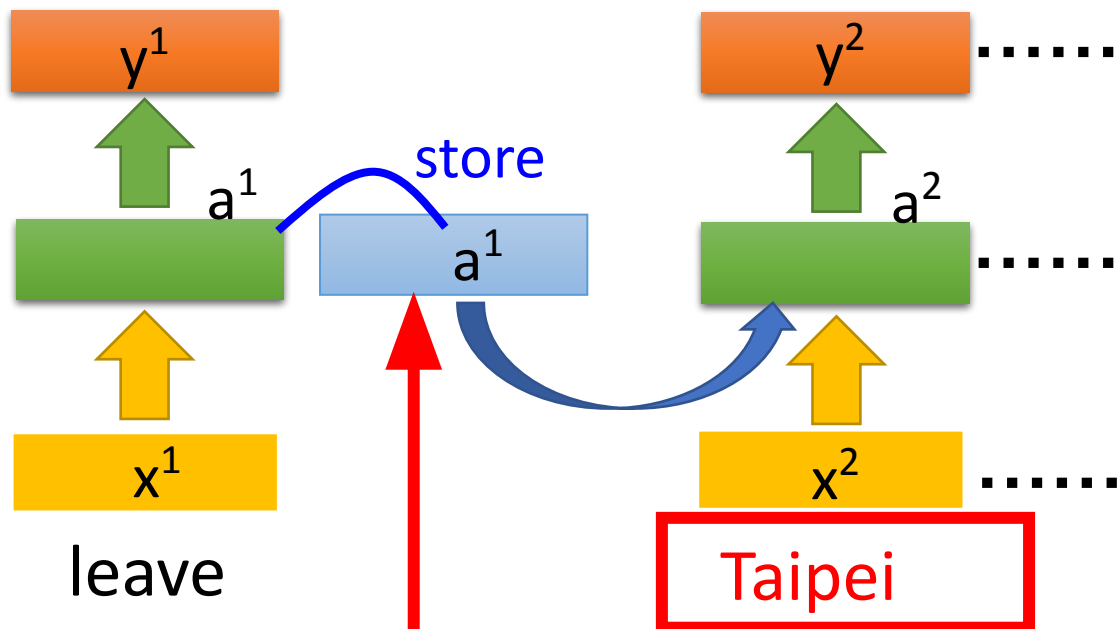
Разные веса!

Вероятность "leave"

Вероятность "Taipei"

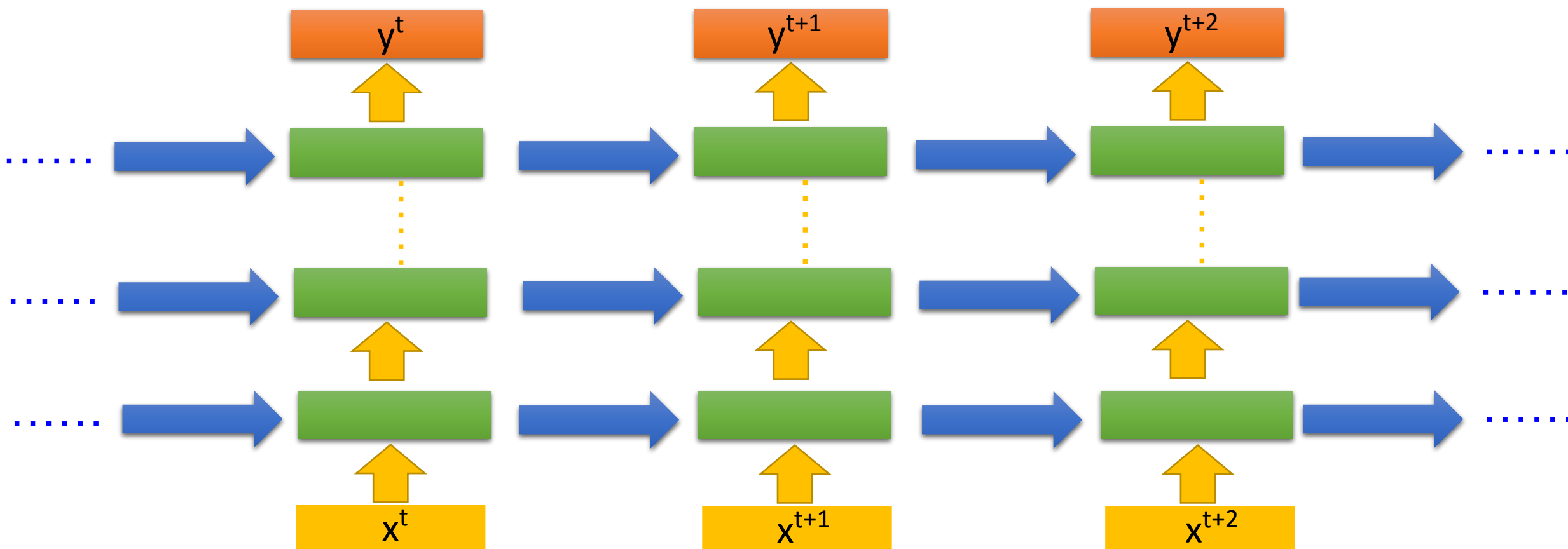
Вероятность
"arrive"

Вероятность "Taipei"



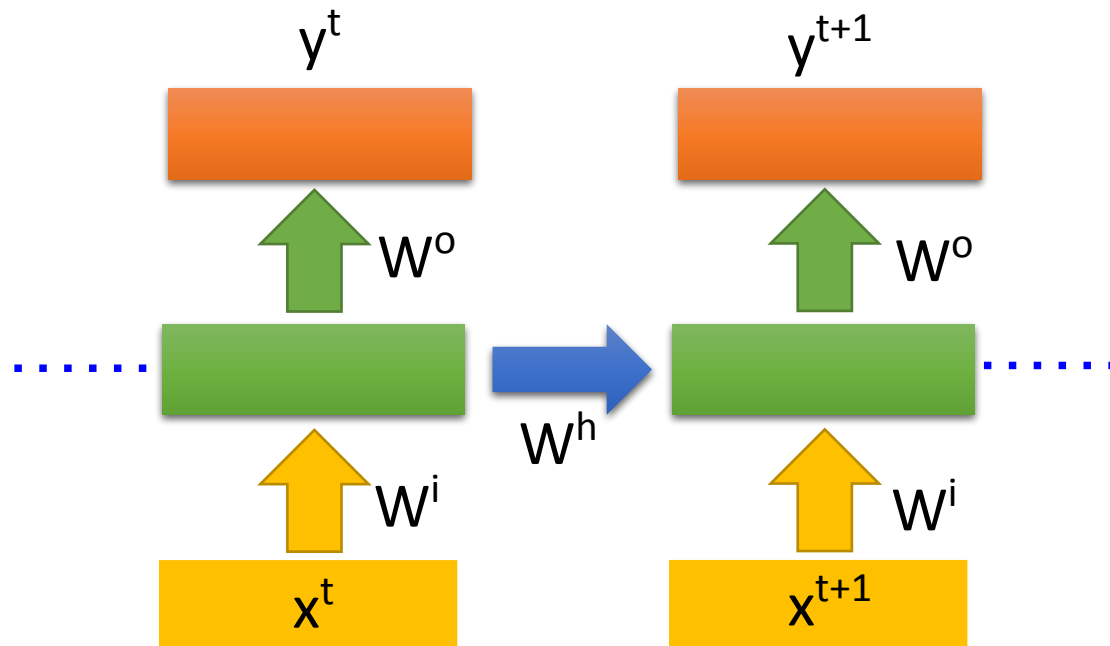
Значения, хранящиеся в памяти, разные.

RNN может быть глубокой ...

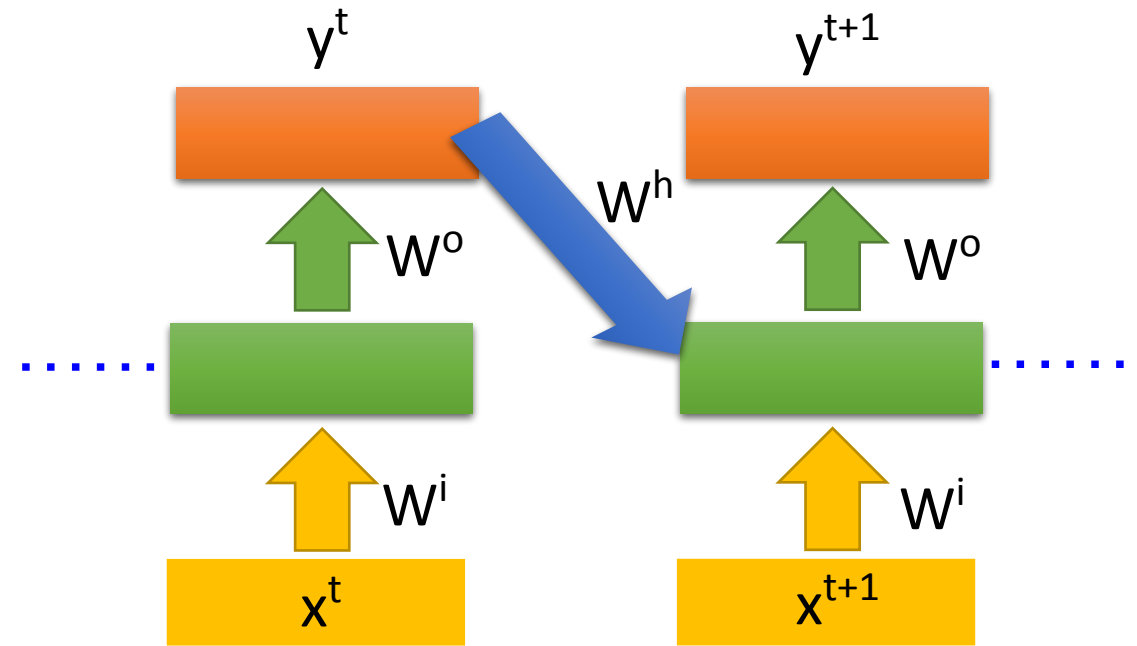


Elman Network & Jordan Network

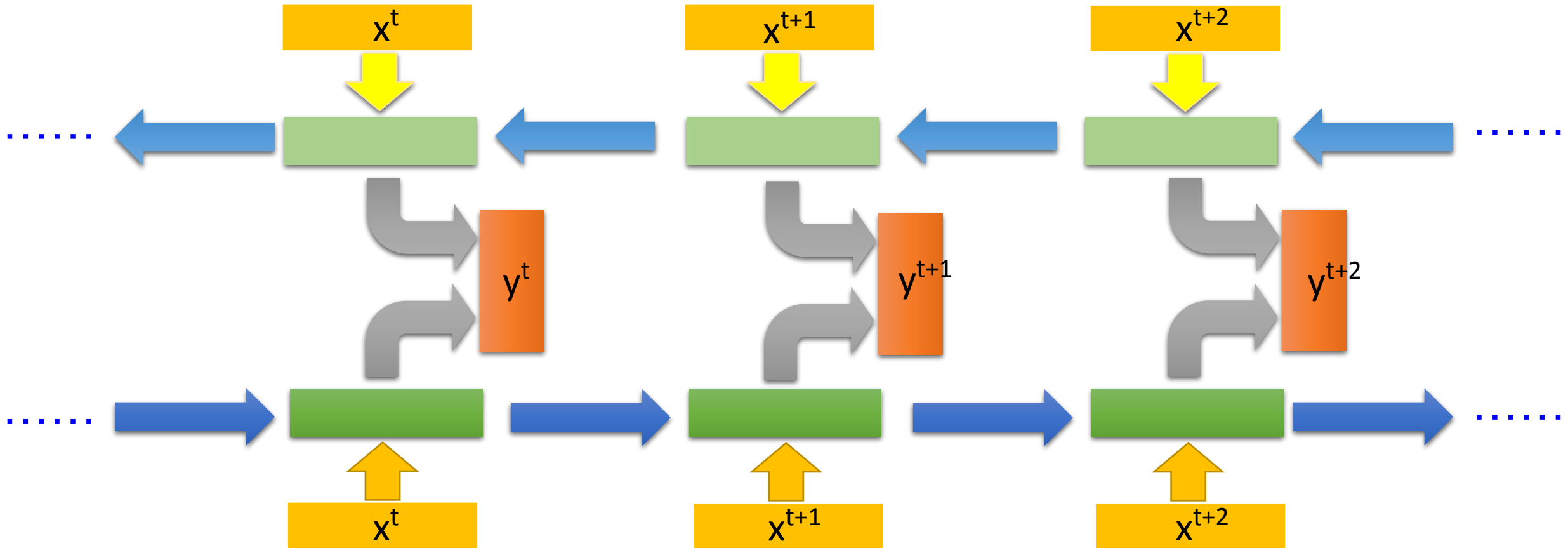
Elman Network



Jordan Network

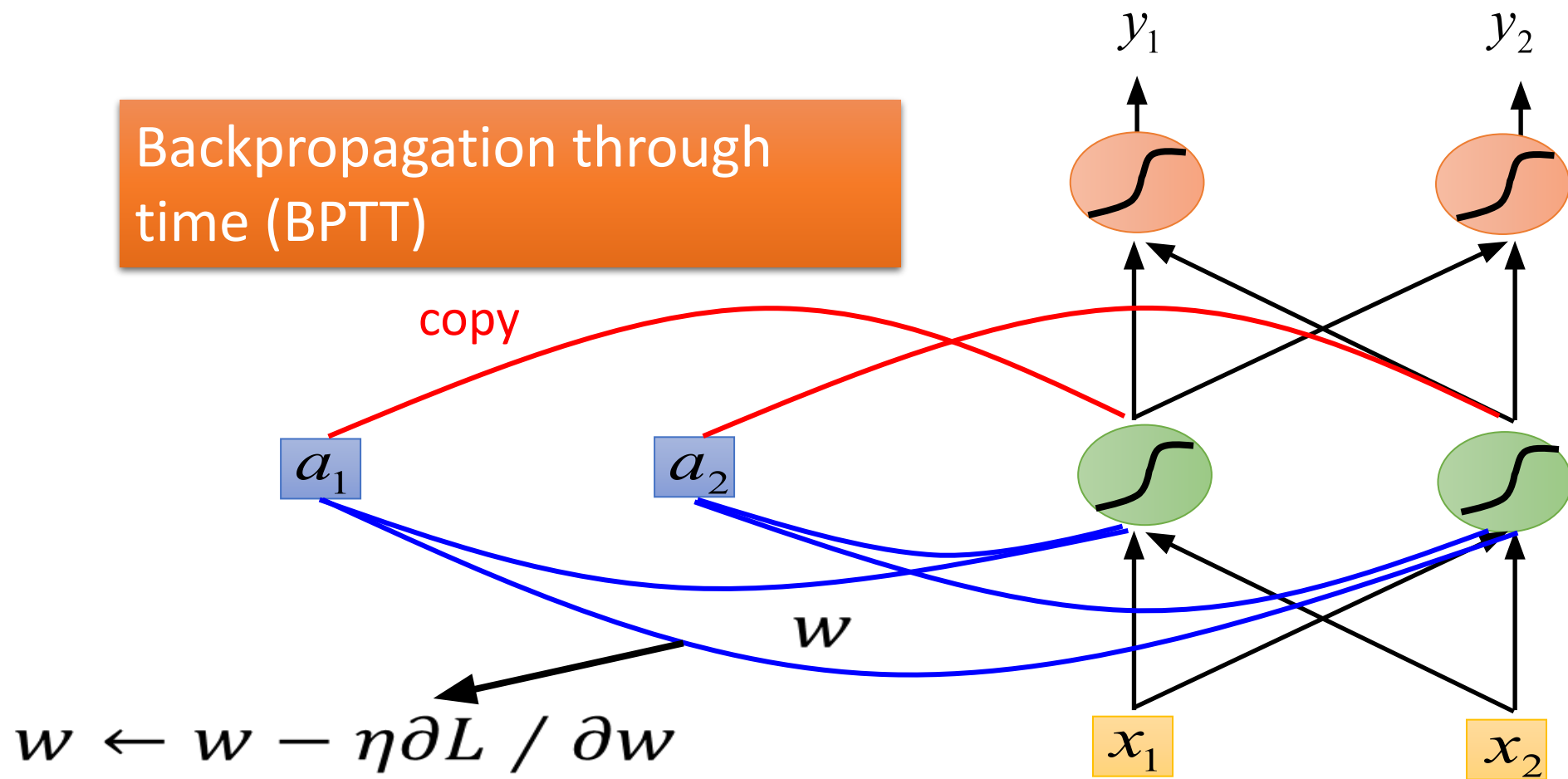


Bidirectional RNN



Обучение

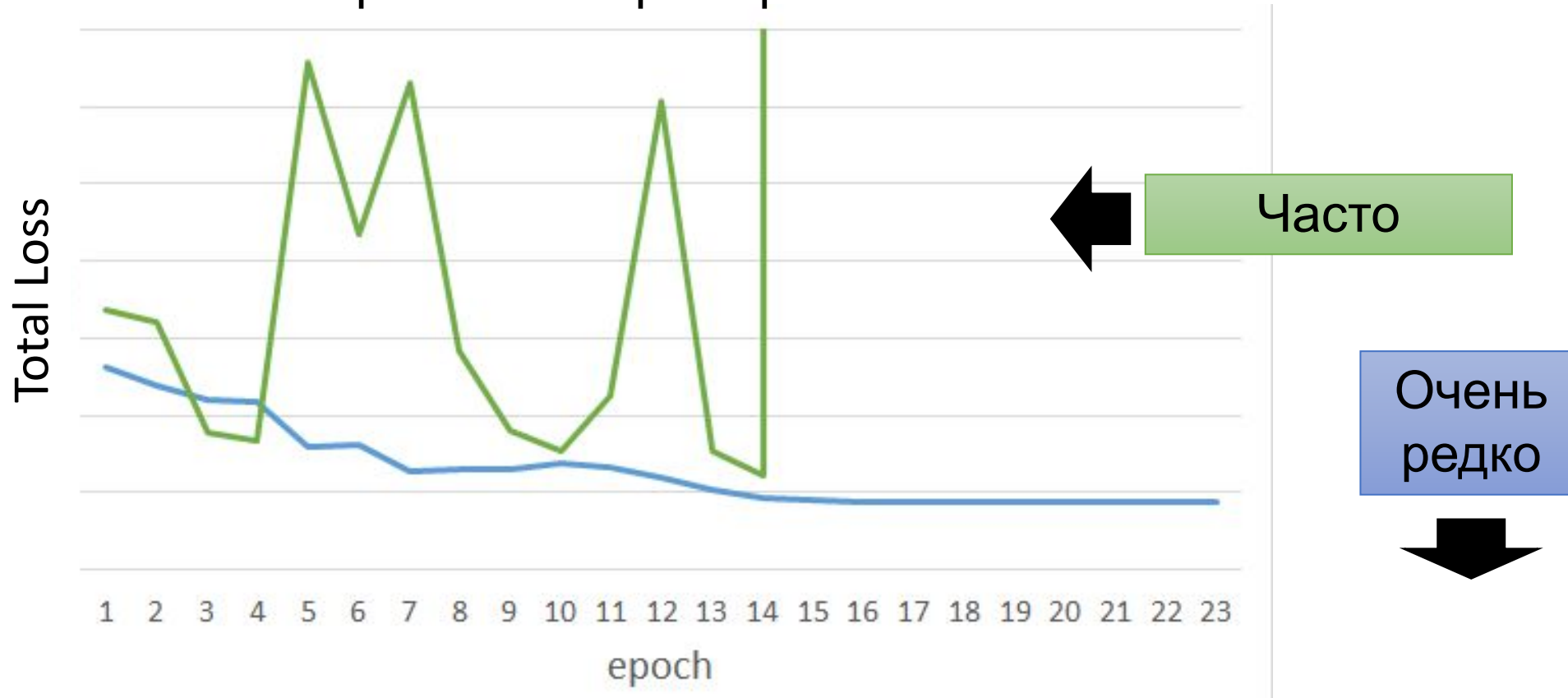
Backpropagation through
time (BPTT)



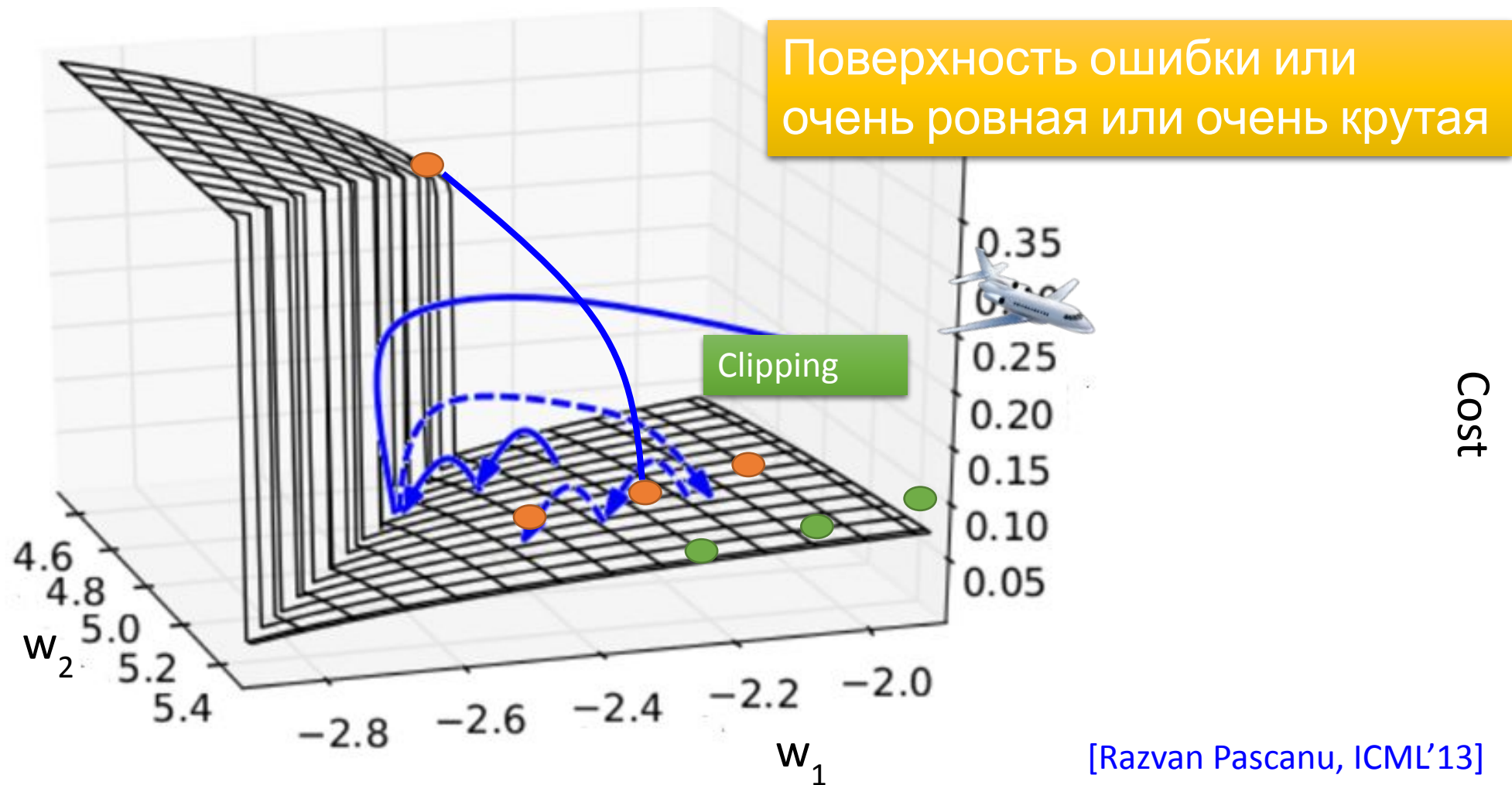
К сожалению

- RNN сети очень сложно учить

На реальных примерах



Сложная поверхность ошибки



[Razvan Pascanu, ICML'13]

Пропадающий градиент/ Взрыв Градиента

$$\begin{array}{ll} w = 1 & \longrightarrow y^{1000} = 1 \\ w = 1.01 & \longrightarrow y^{1000} \approx 20000 \end{array}$$

Large
 $\partial L / \partial w$

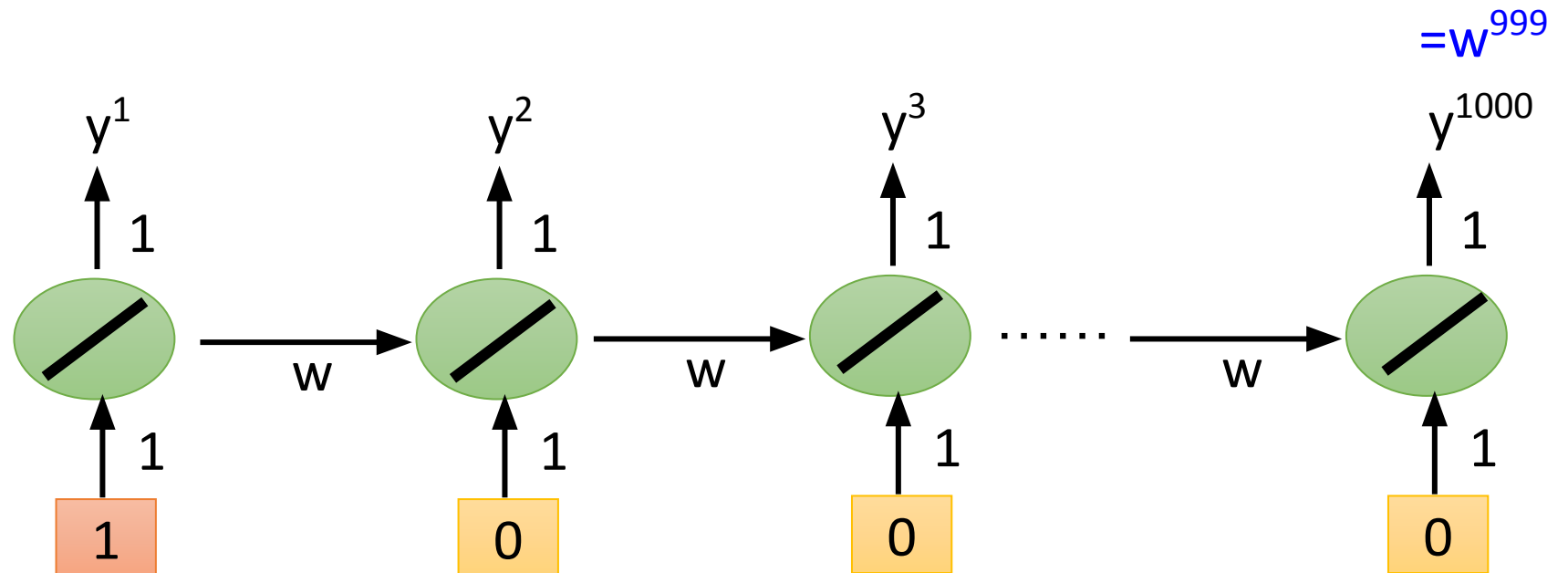
Взрыв

$$\begin{array}{ll} w = 0.99 & \longrightarrow y^{1000} \approx 0 \\ w = 0.01 & \longrightarrow y^{1000} \approx 0 \end{array}$$

small
 $\partial L / \partial w$

Пропадание
градиента

Пример



Many to Many (Output is shorter)

$$\frac{\partial h_t}{\partial h_k} = \prod_{t \geq i > k} \frac{\partial h_i}{\partial h_{i-1}} = \prod_{t \geq i > k} W_{hh}^T \text{diag}[g'(h_{i-1})]$$

$$\left\| \frac{\partial h_i}{\partial h_{i-1}} \right\| \leq \|W_{hh}^T\| \|\text{diag}(g'(h_{i-1}))\|$$

Largest Singular
value of W_{hh}

$$\leq \gamma_W \gamma_g$$

$\gamma_W \gamma_g$ = an upper bound for the norm of jacobian!

$$\left\| \frac{\partial h_3}{\partial h_k} \right\| \leq (\gamma_W \gamma_g)^{t-k}$$

Достаточное условие для исчезающего градиента

$\gamma_W \gamma_g < 1$ и $(t-k) \rightarrow \infty$ то долгосрочные вклады уходят в 0 экспоненциально быстро с $t-k$ (метод степенной итерации).

Следовательно,

достаточное условие возникновения исчезающего градиента:

:

$\gamma_W \gamma_g < 1/\gamma_W \gamma_g$ i.e. for sigmoid,

$\gamma_W \gamma_g < 4$ i.e., for tanh,

Необходимое условие для взрывающегося градиента

$\gamma_W \gamma_g > 1$ и $(t-k) \rightarrow \infty$ то долгосрочные

вклады уходят в бесконечность экспоненциально быстро с $t-k$ (метод степенной итерации).

Следовательно,

достаточное условие возникновения исчезающего градиента:

:

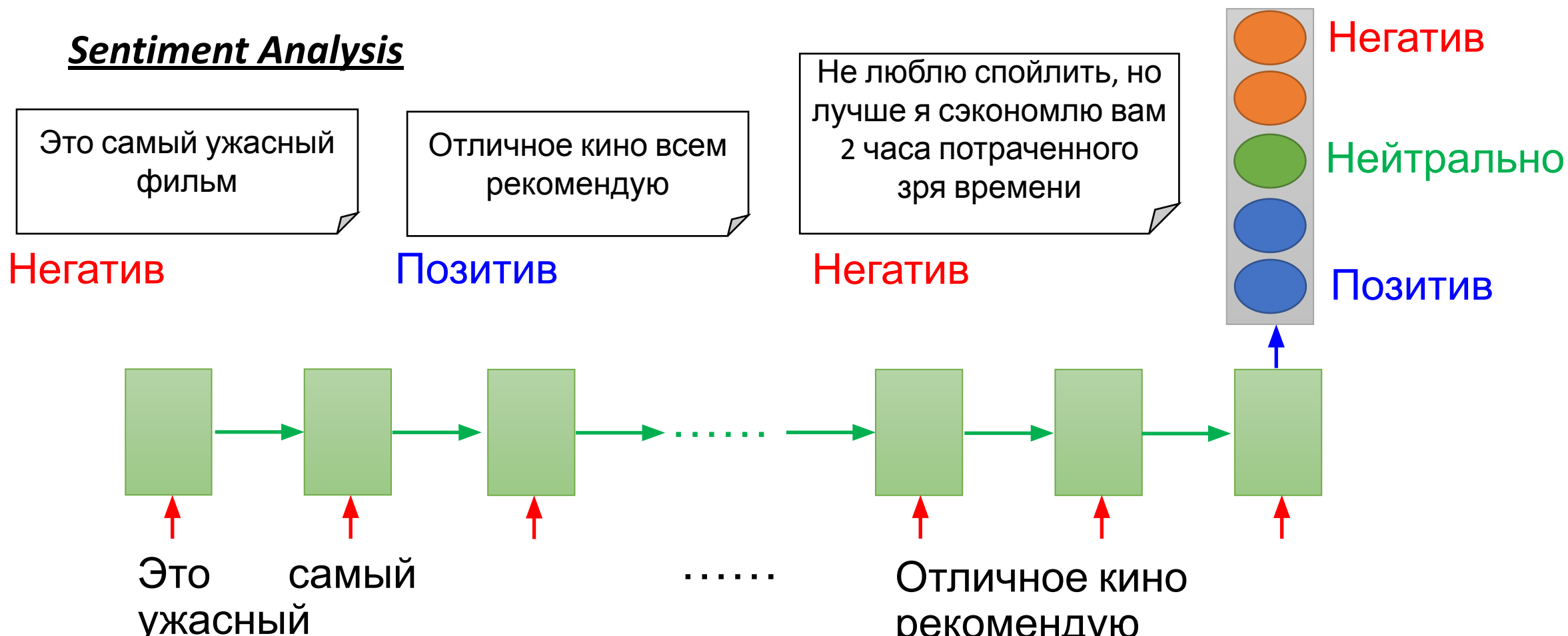
$\gamma_W \gamma_g < 1/\gamma_W \gamma_g$ i.e. for sigmoid,

$\gamma_W \gamma_g < 4$ i.e., for tanh,

Many to one

Вход - это векторная последовательность, а на выходе - только одно значение

Sentiment Analysis



Many to Many (Output is shorter)

- И вход, и выход - это последовательности, но выход короче.
 - Распознавание речи (character sequence)

Output: В белом плаще с кровавым подбоем, шаркающей походкой...

