HCT NLP Week 2

问答摘要与推理
词向量实践与RNNs

Outline

- 词向量计算两种优化方法
- 词向量在工程中的具体实现
- RNN递归神经网络结构
- RNN、LSTM、GRU

RNNs

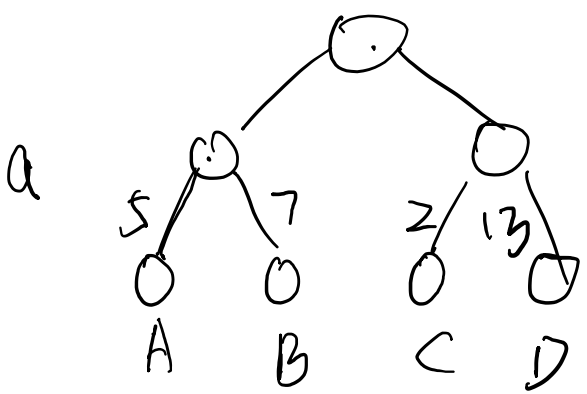
Outline

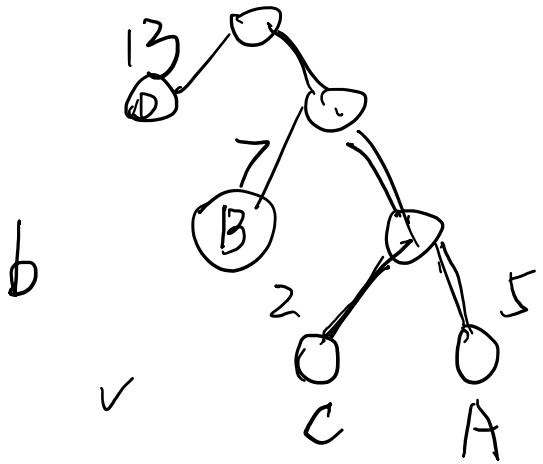
- 词向量计算两种优化方法
- 词向量在工程中的具体实现
- RNN递归神经网络结构
- RNN、LSTM、GRU

Hierarchical Softmax $O(V)$

Huffman Tree（哈夫曼树）

最优二叉树





$Path\_a = 5\times2 + 7\times2 + 2\times2 + 13\times2 = 54$

$Path\_b = 5\times3 + 2\times3 + 7\times2 + 13\times1 = 48$

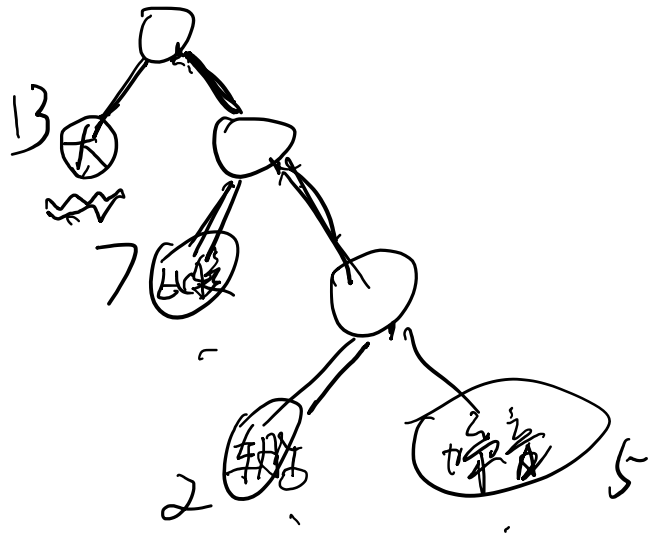
$P\_a > P\_b$

Hierarchical Softmax

Huffman Tree（哈夫曼树）

文本： 车船 噪音 比较 大 ....

词频 2 5 7 13



左 0 右 1

大：0

比较：10

车船：110

噪音：111

1) V-1个内部节点

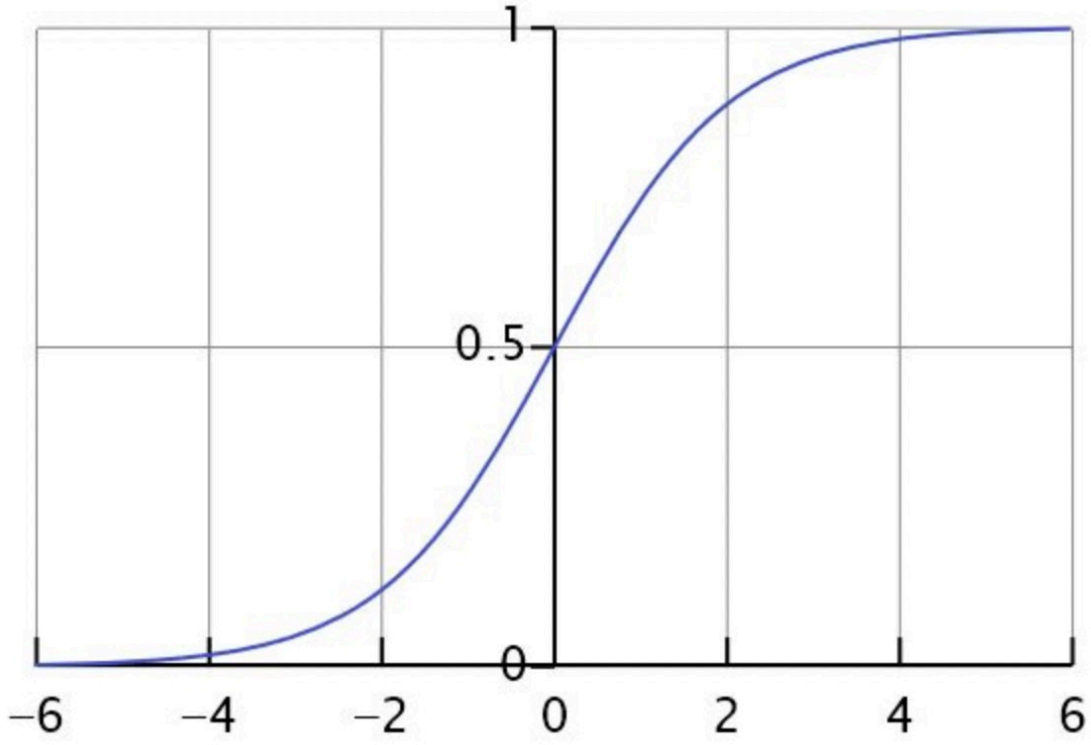
V个叶节点

2) 一一对应

3)

Hierarchical Softmax

Logistic Regression



$$L(\hat{y}, y) = -y \log \hat{y} - (1-y) \log(1-\hat{y})$$

召回 预估/排序

1w Too 1-100 LR. GBDT XGBoost

$$\hat{y} = P(y=1|x) \quad 0 \le \hat{y} \le 1$$

$$\hat{y} = w^T x + b \quad w^T x = \sum_{i=1}^{n} w_i x_i$$

$$= w_0 + w_1 x_1 + w_2 x_2 + \dots + w_n x_n$$

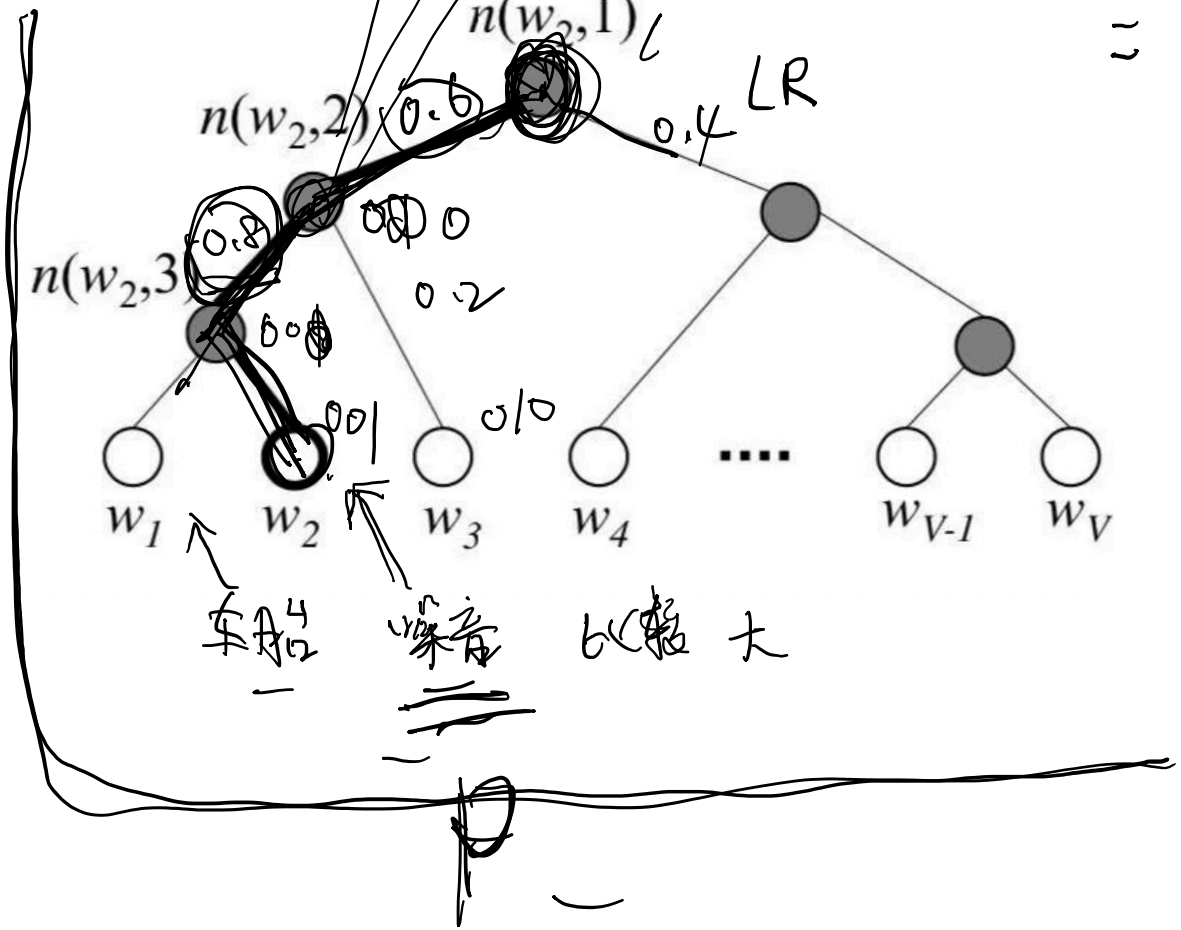
$$\sigma(z) = \frac{1}{1+e^{-z}}$$

$$= \frac{1}{1+e^{-(w^T x + b)}}$$

Loss: $P(y|x) = \begin{cases} \hat{y} & y=1 \\ 1-\hat{y} & y=0 \end{cases} = \hat{y}^y (1-\hat{y})^{(1-y)}$

$\log(P(y|x))$

Hierarchical Softmax



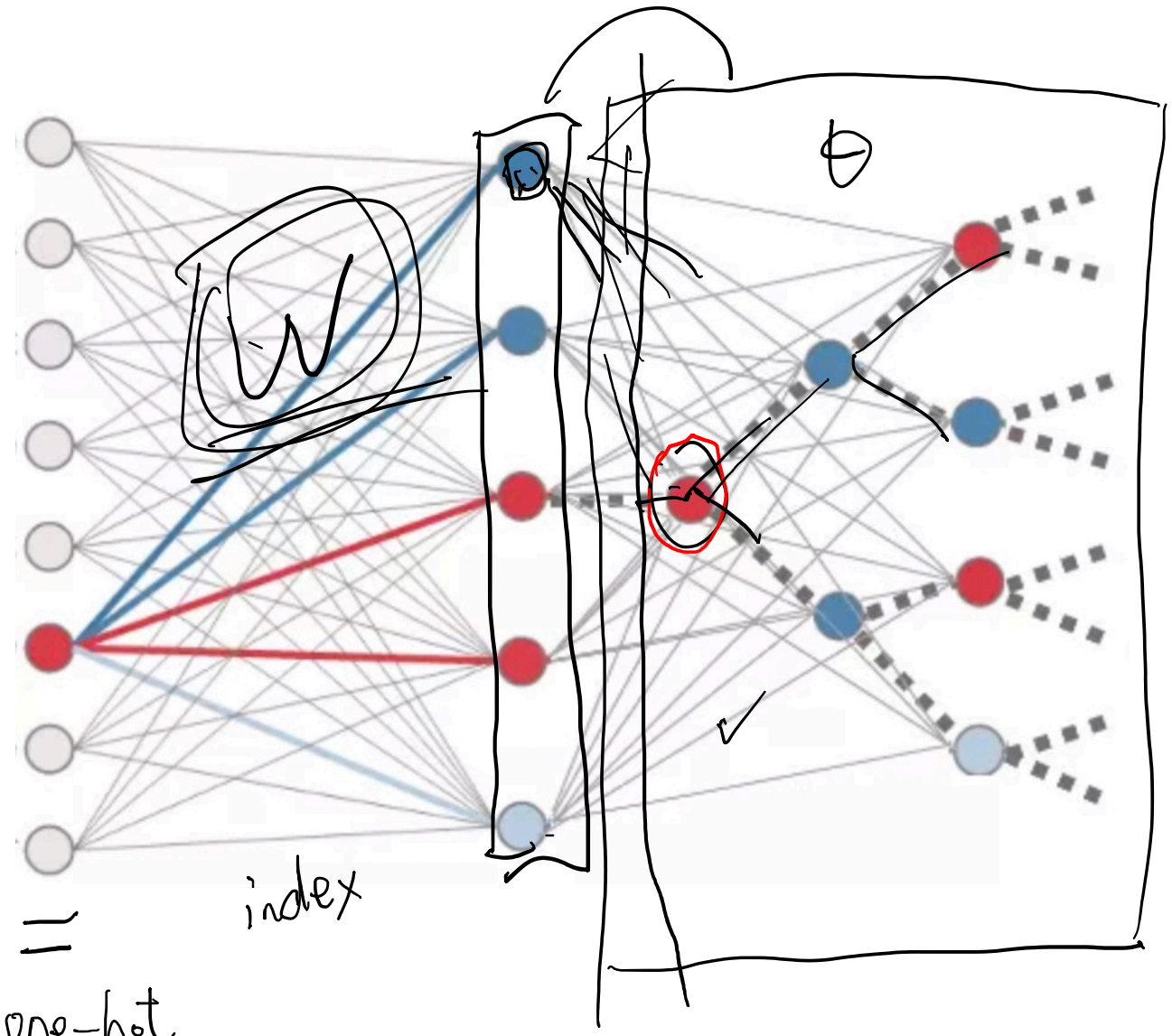
$$p(w \mid context(w))$$

$$= \prod_{j=2}^{l^w} p(d_j^w \mid x_w, \theta_{j-1}^w)$$

$$p(d_j^w \mid x_w, \theta_{j-1}^w) = \begin{cases} \sigma(x_w^T \theta_{j-1}^w) & d_j^w = 0 \\ 1 - \sigma(x_w^T \theta_{j-1}^w) & d_j^w = 1 \end{cases}$$

$$L(w,j) = (1 - d_j^w) \cdot \log[\sigma(x_w^T \theta_{j-1}^w)] + d_j^w \log[1 - \sigma(x_w^T \theta_{j-1}^w)]$$

Hierarchical Softmax



V

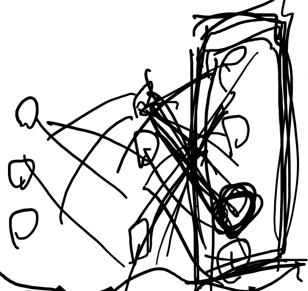
log V

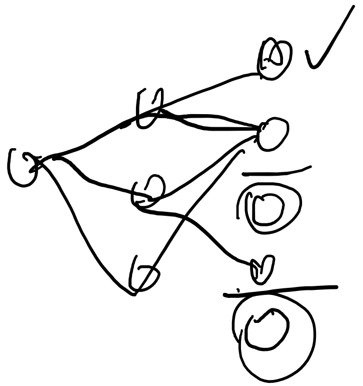
① O(|V|) → O(log|V|)

② 高频 路径更短，计算更快

Negative Sampling

每次让一个训练样本仅仅更新一小部分的权重



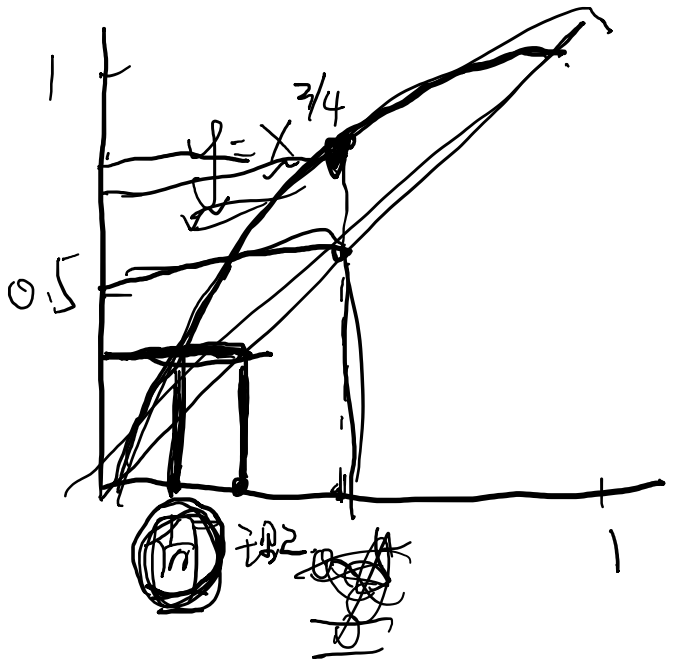


Negative Sampling

9:10

总结：

$$P(w_i) = \frac{f(w_i)^{3/4}}{\sum_{j=0}^{n} f(w_j)^{3/4}}$$



1)

$$y = f(w_i)^{3/4}$$

word2vec → sg. cbow

$\frac{5}{10000}$

$\frac{1}{100}$

$y = x$

$V = 10000$

↓

2~5

5~20

$y = x_{0.5}$

$y = x$

$y = 1$

$y = 1$

word2vec trick. 2-10个

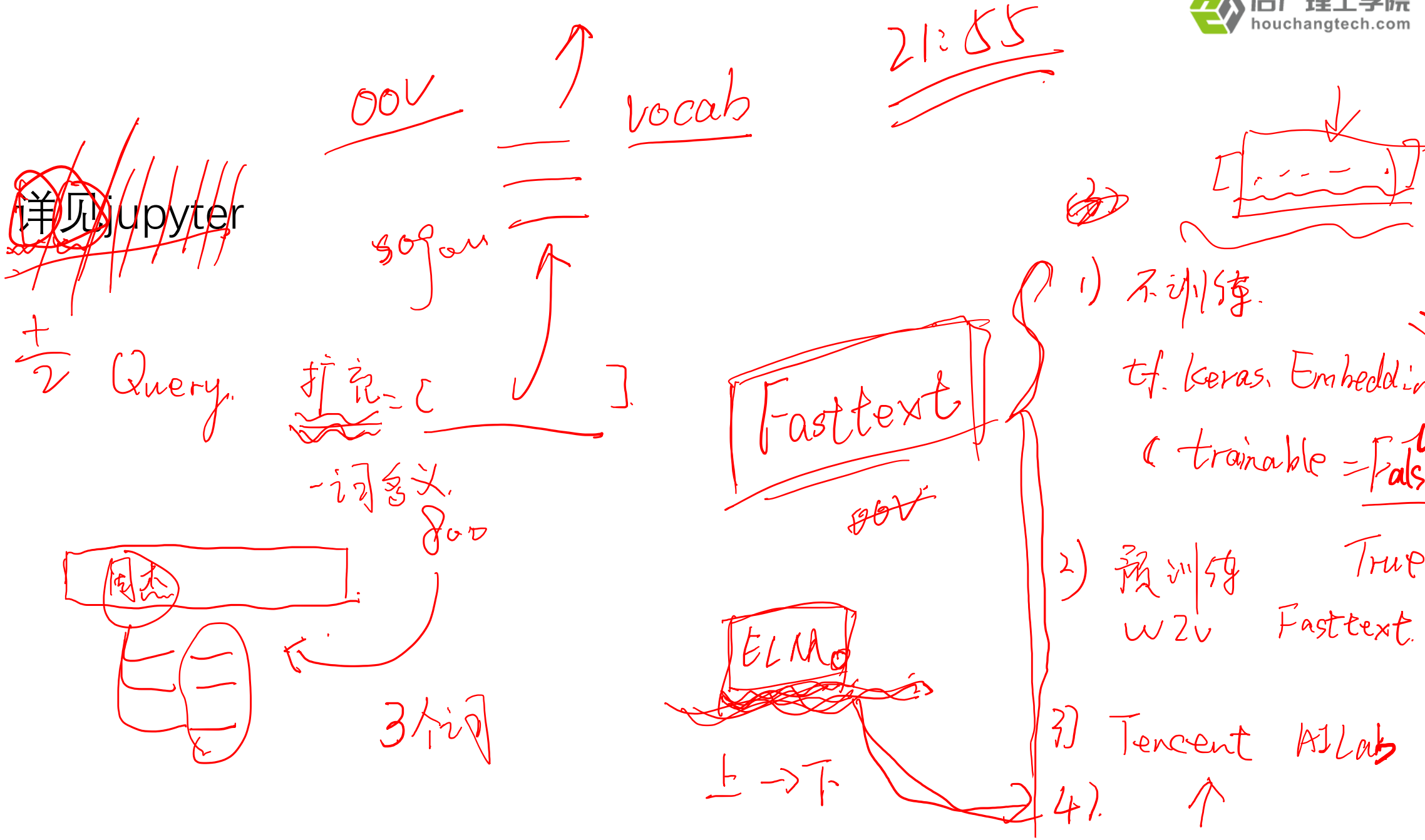
高频词的亚采样

$P(w_i) = 1 - \sqrt{\frac{sample}{freq(w_i)}}$ → ↓ $10^{-5} \sim 10^{-2}$

Outline

- 词向量计算两种优化方法
- 词向量在工程中的具体实现
- RNN递归神经网络结构
- RNN、LSTM、GRU

- 详见jupyter

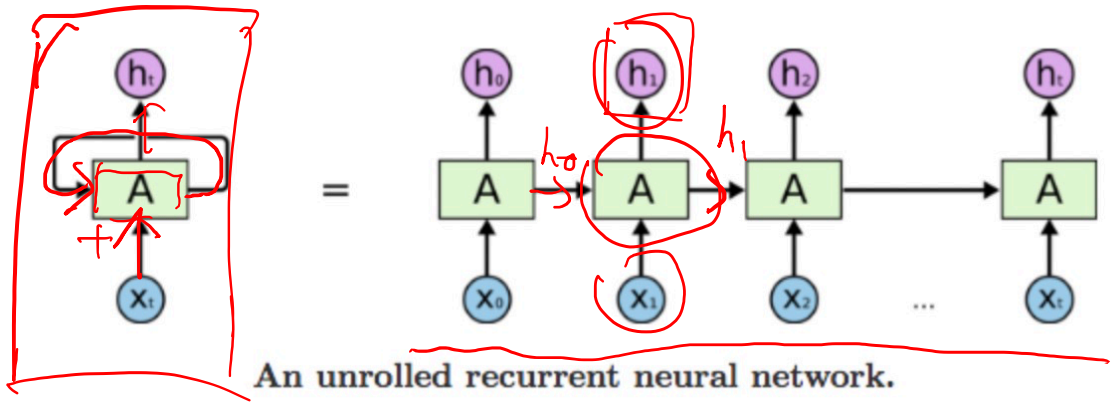


Outline

- 词向量计算两种优化方法
- 词向量在工程中的具体实现
- RNN递归神经网络结构
- RNN、LSTM、GRU

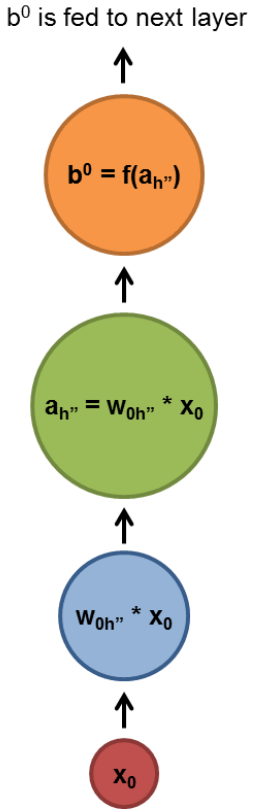
RNN

Recurrent Neural Network



An unrolled recurrent neural network.

$h_1 = f(h_0, x_1)$

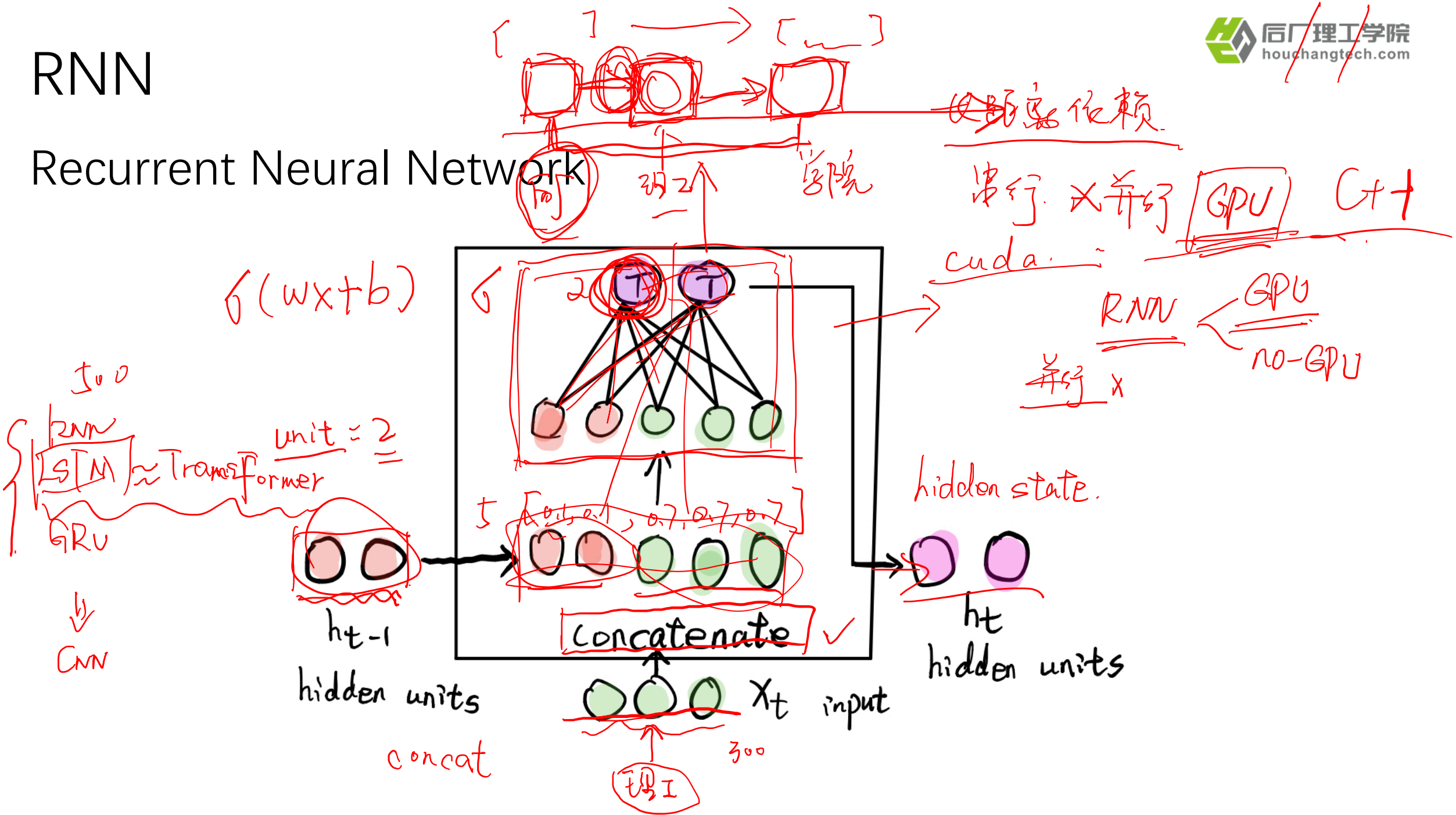


RNN

Recurrent Neural Network

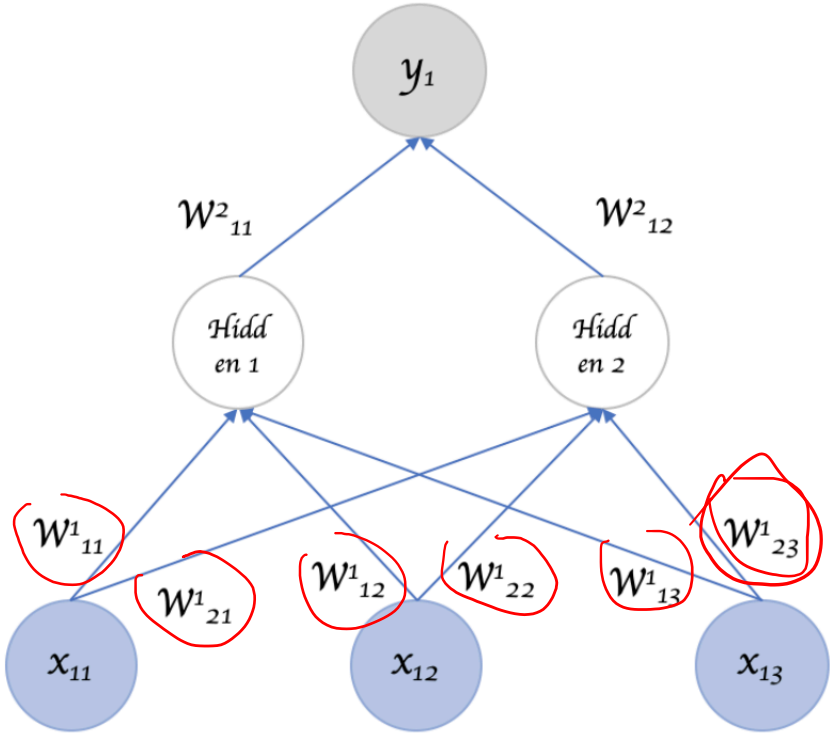
RNN

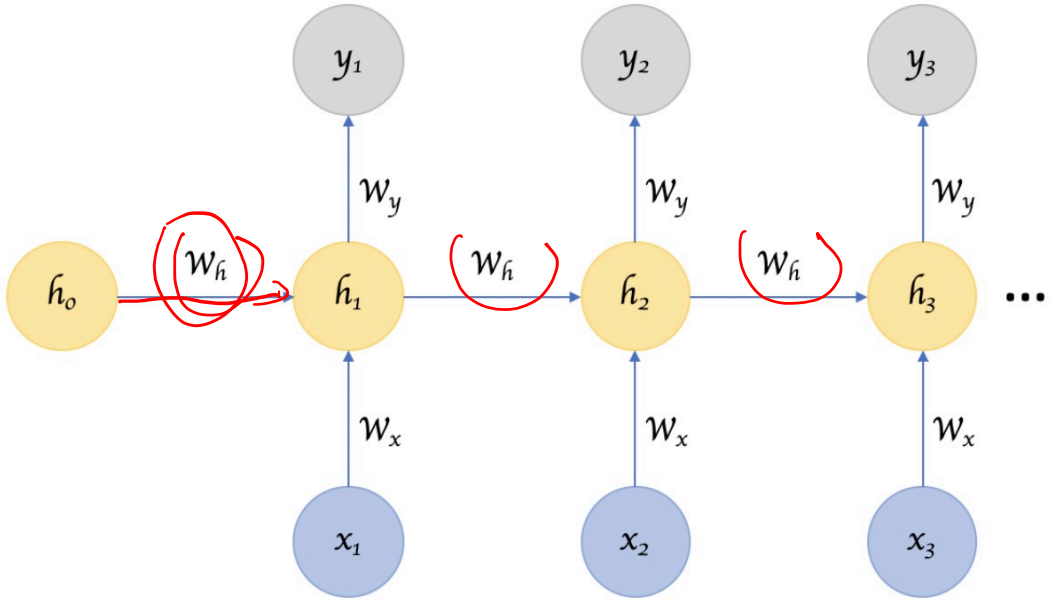
Recurrent Neural Network



RNN

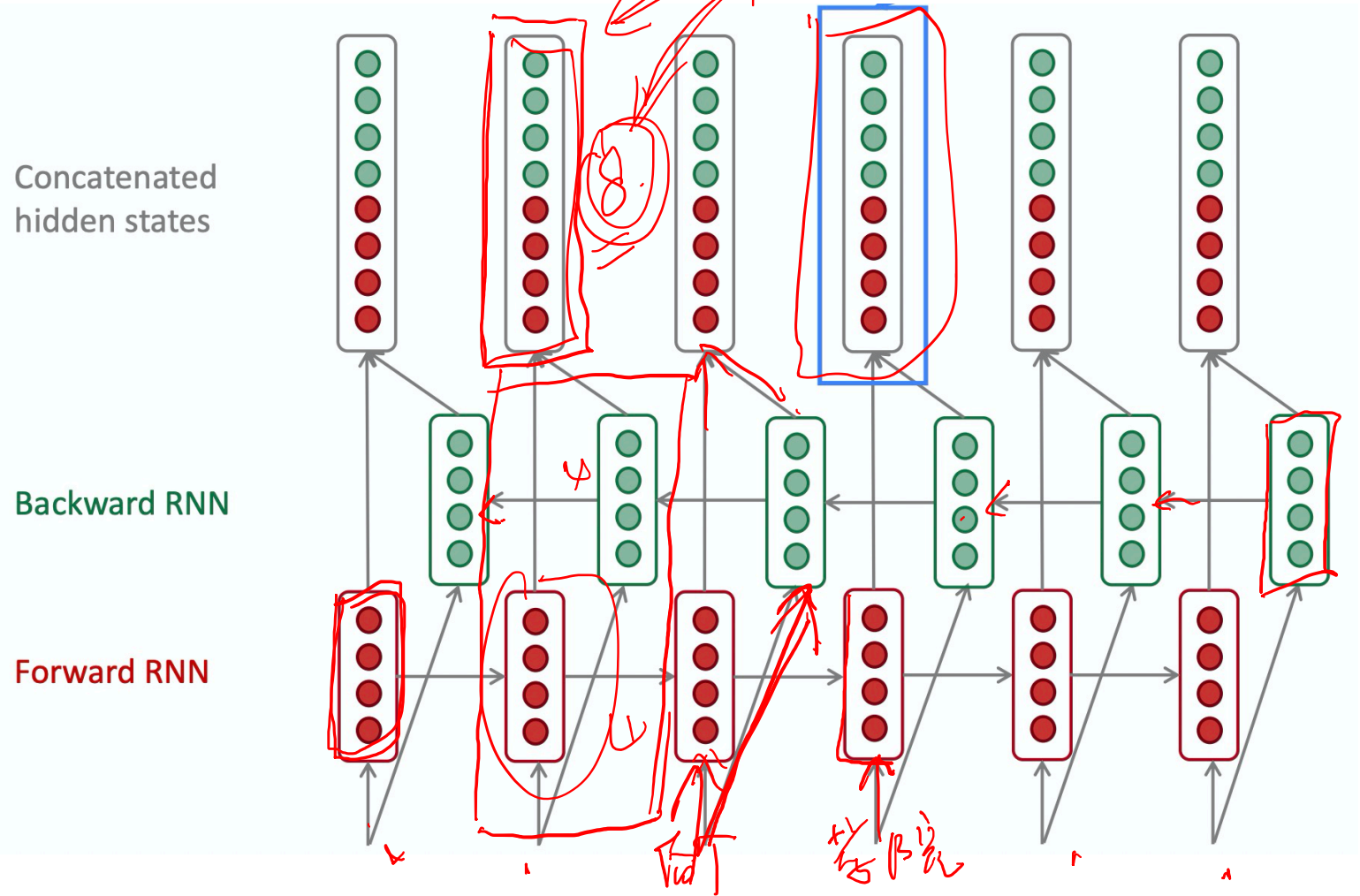
parameter sharing





RNN

Bidirectional RNNs



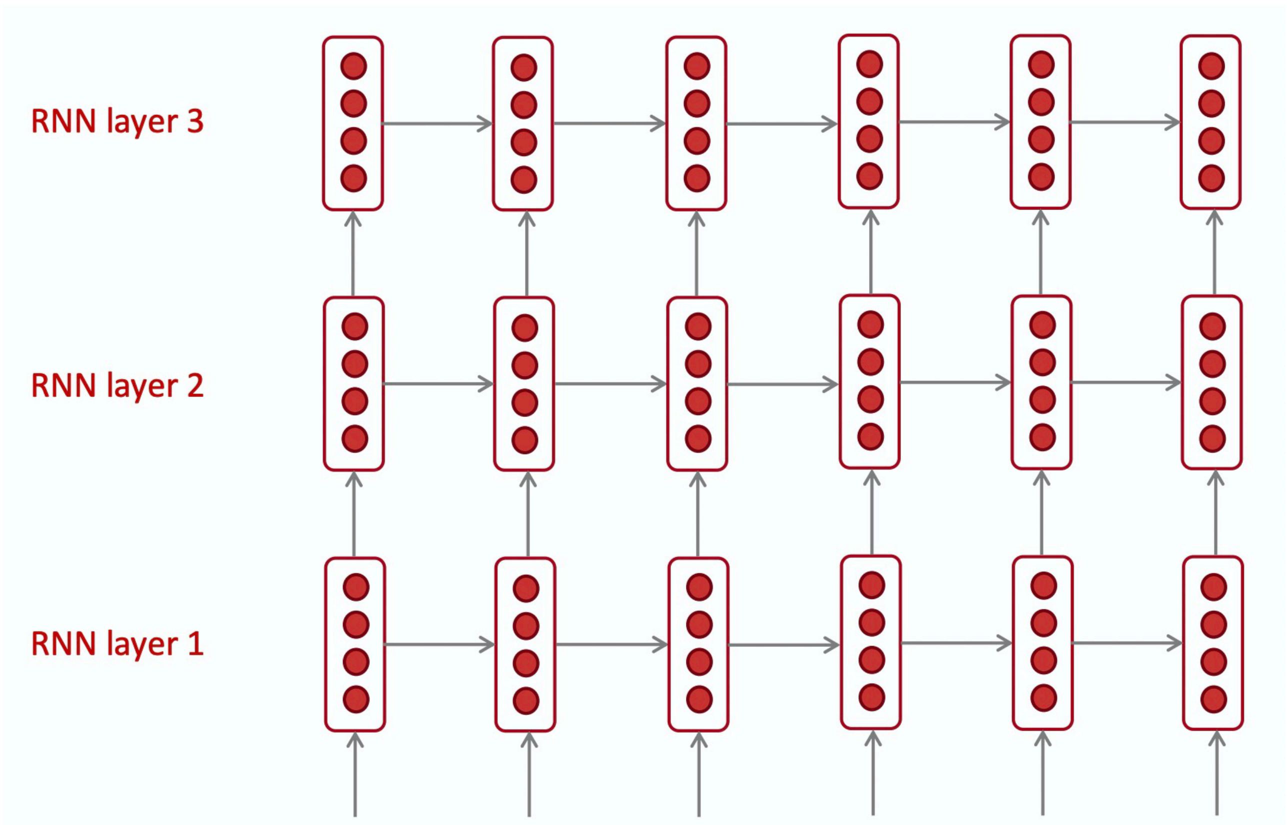
$$h^t = [\overrightarrow{h}^{(t)}; \overleftarrow{h}^{(t)}] \quad \text{concat.}$$

$$\overleftarrow{h}^{(t)} = RNN_{BW}(\overleftarrow{h}^{(t+1)}, x^{(t)})$$

$$\overrightarrow{h}^{(t)} = RNN_{FW}(\overrightarrow{h}^{(t-1)}, x^{(t)})$$

RNN

Deep RNNs

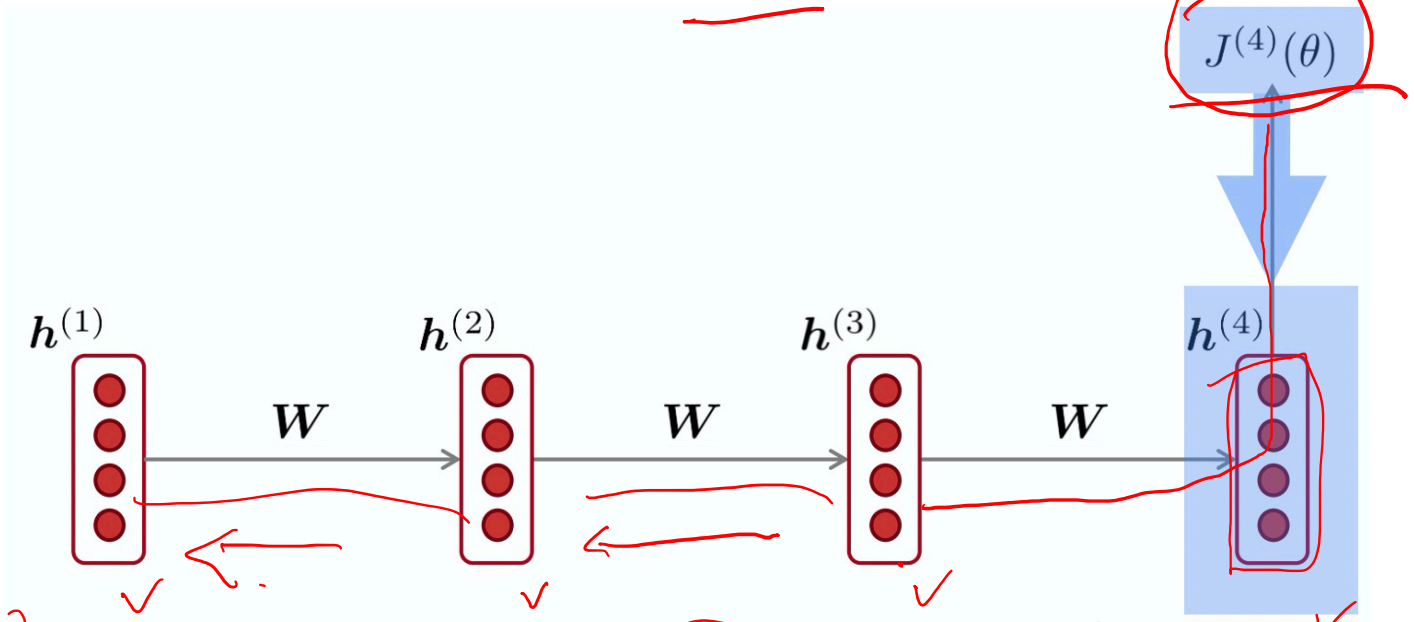


encoder
NMT → 2-4 layers
decoder
4 layers

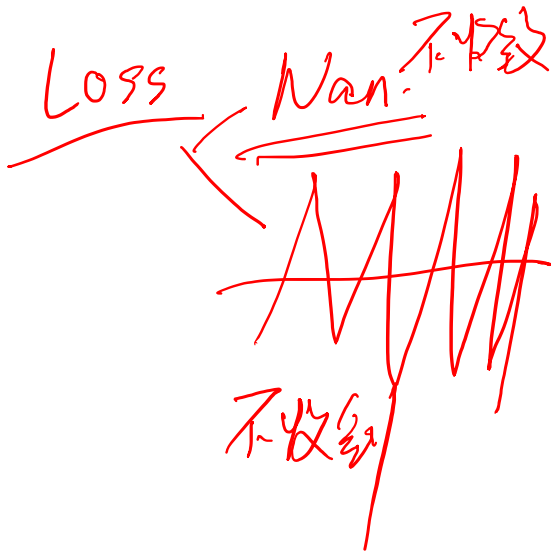
文本摘要 单向 双向

RNN GRU LSTM

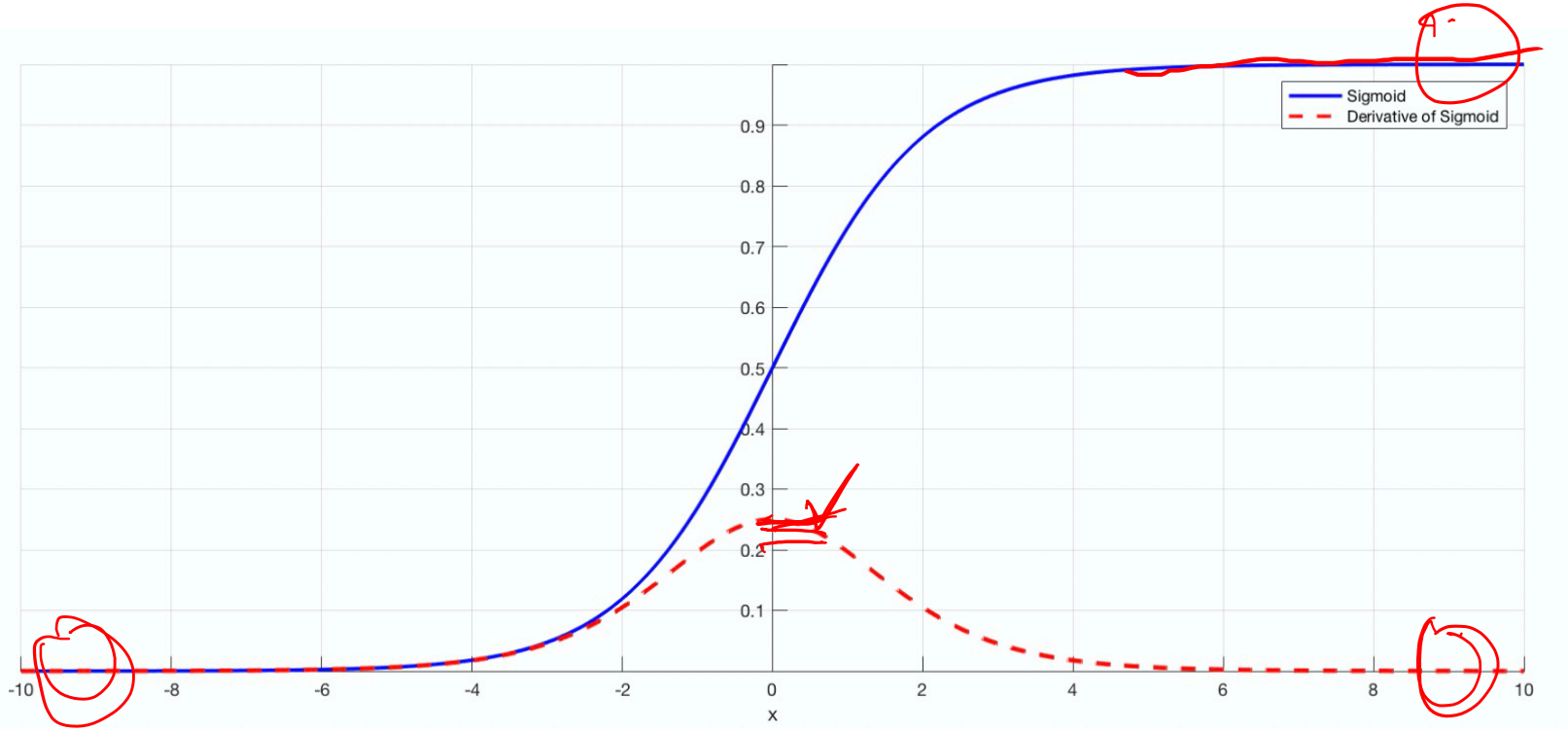
Exploding and Vanishing Gradient Problem



$$\frac{\partial J^{(4)}}{\partial h^{(1)}} = \frac{\partial h^{(2)}}{\partial h^{(1)}} \times \frac{\partial h^{(3)}}{\partial h^{(2)}} \times \frac{\partial h^{(4)}}{\partial h^{(3)}} \times \frac{\partial J^{(4)}}{\partial h^{(4)}}$$



Exploding and Vanishing Gradient Problem



Exploding Gradient

- gradient clipping

Vanishing Gradient

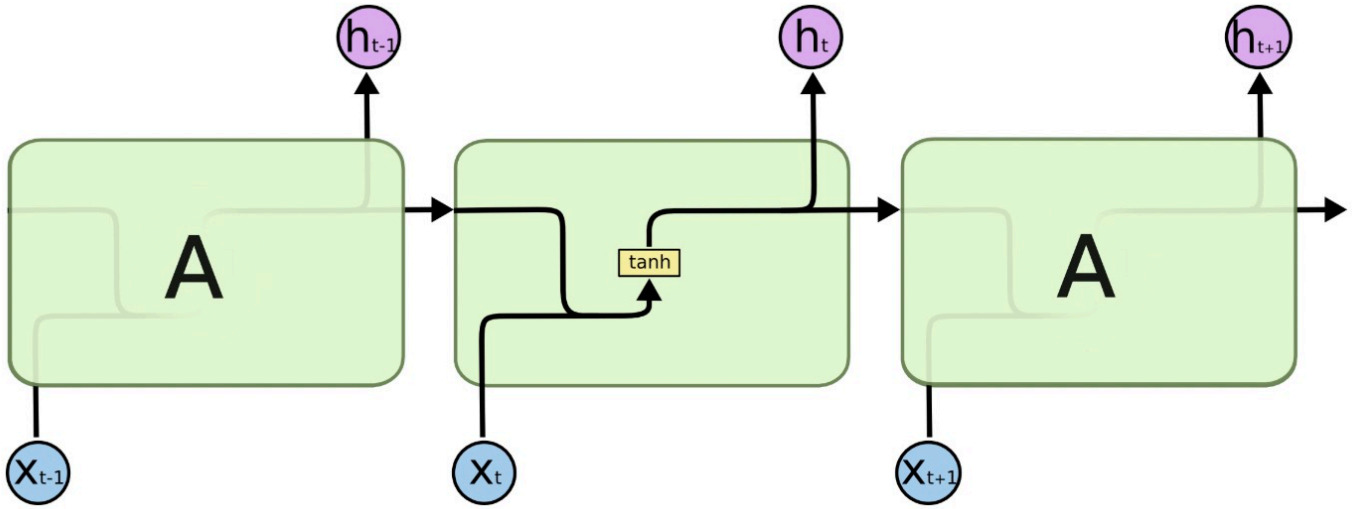
- Identity Initialization
- LSTM
- Residual Networks
- Batch Normalization

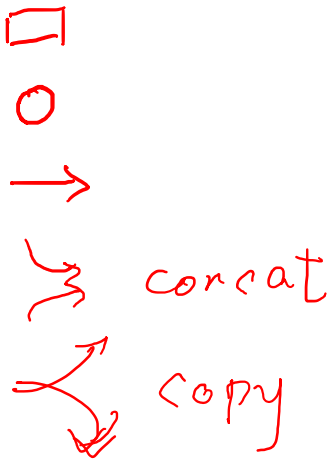
Outline

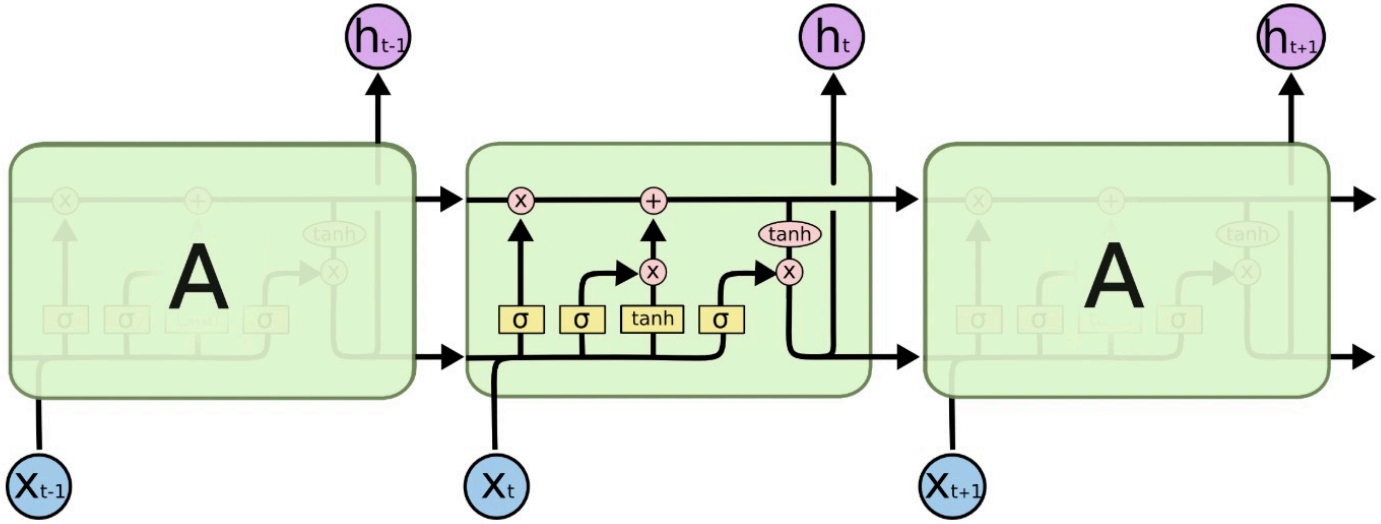
- 词向量计算两种优化方法
- 词向量在工程中的具体实现
- RNN递归神经网络结构
- RNN、LSTM、GRU

LSTM

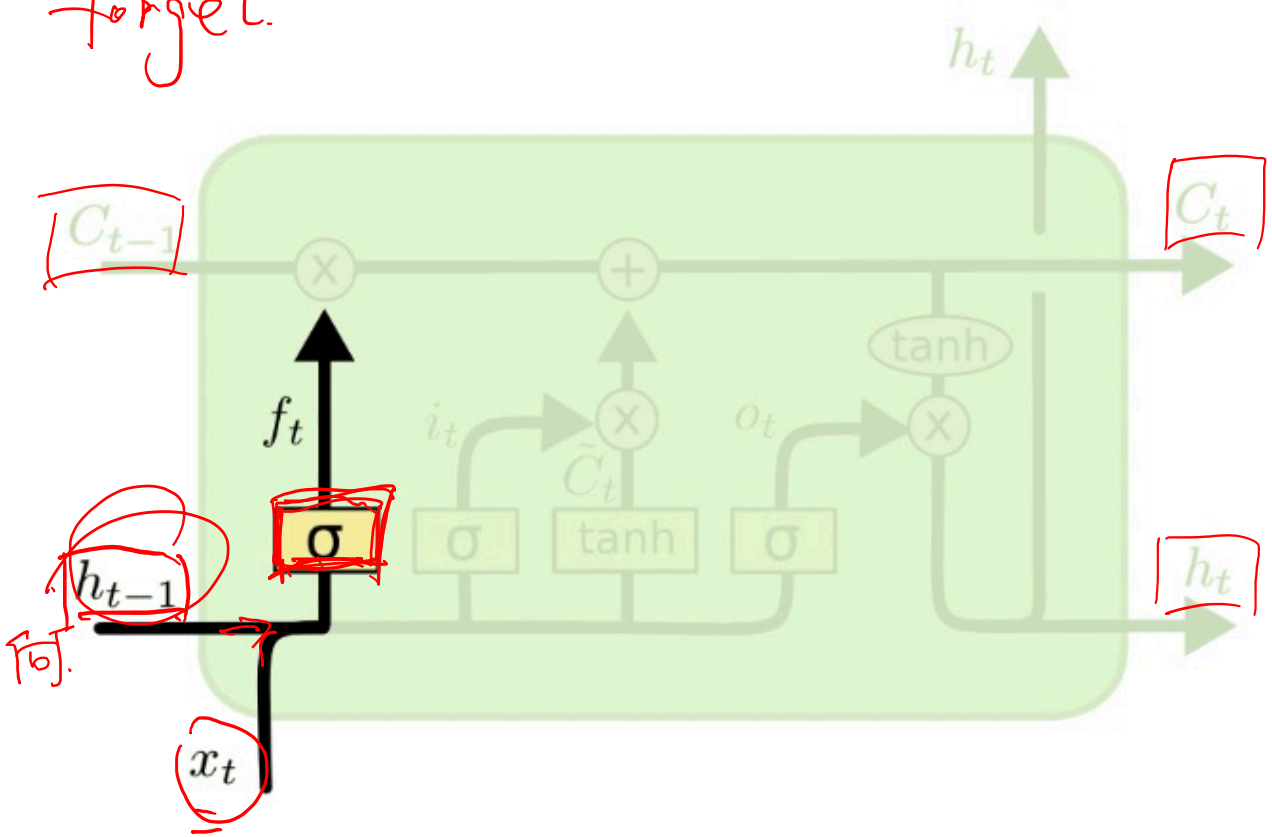
Long Short Term Memory







LSTM

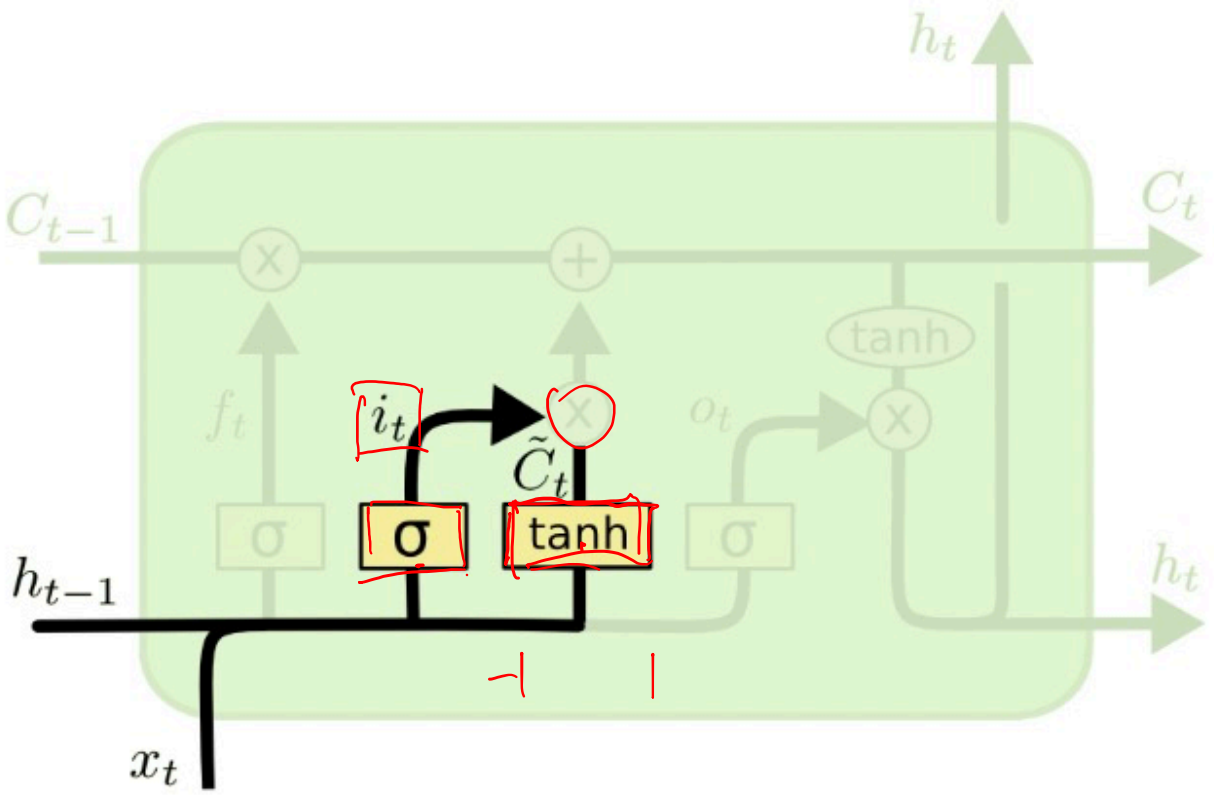


$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Sigmoid

LSTM

input gate.

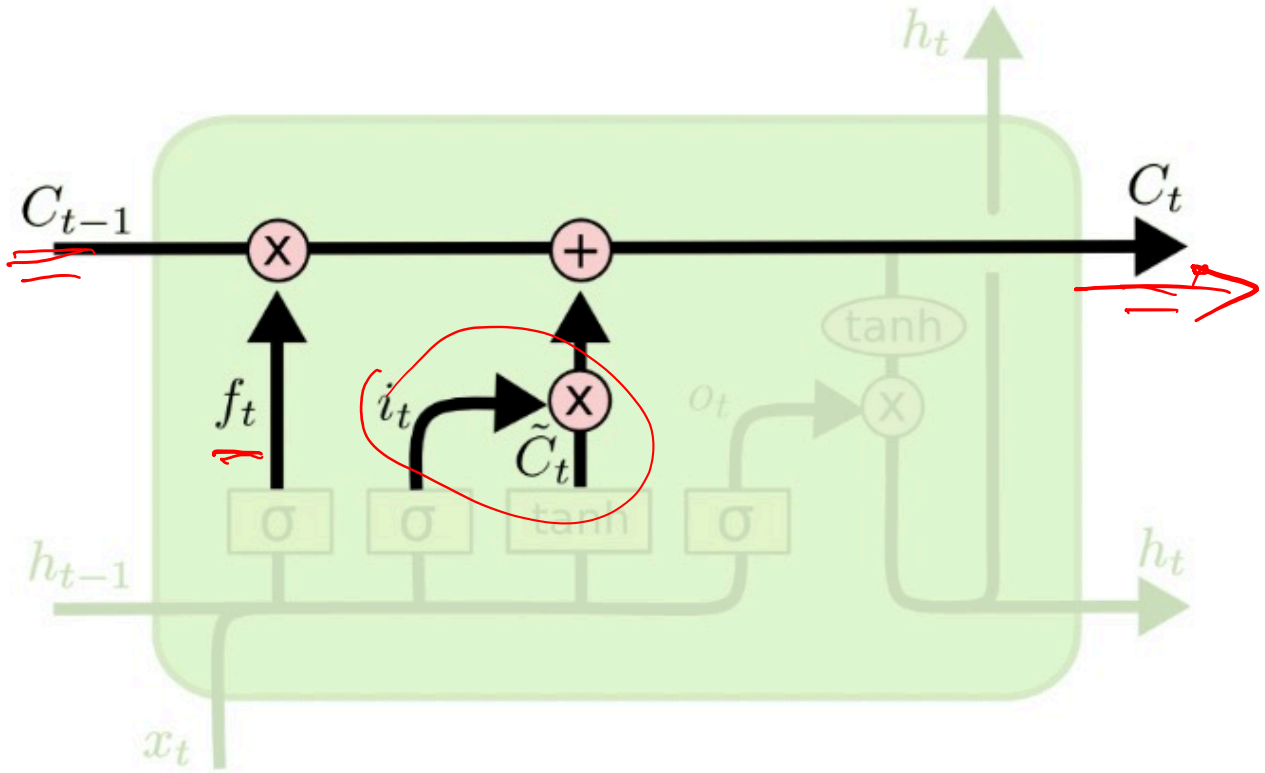


$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$

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

LSTM

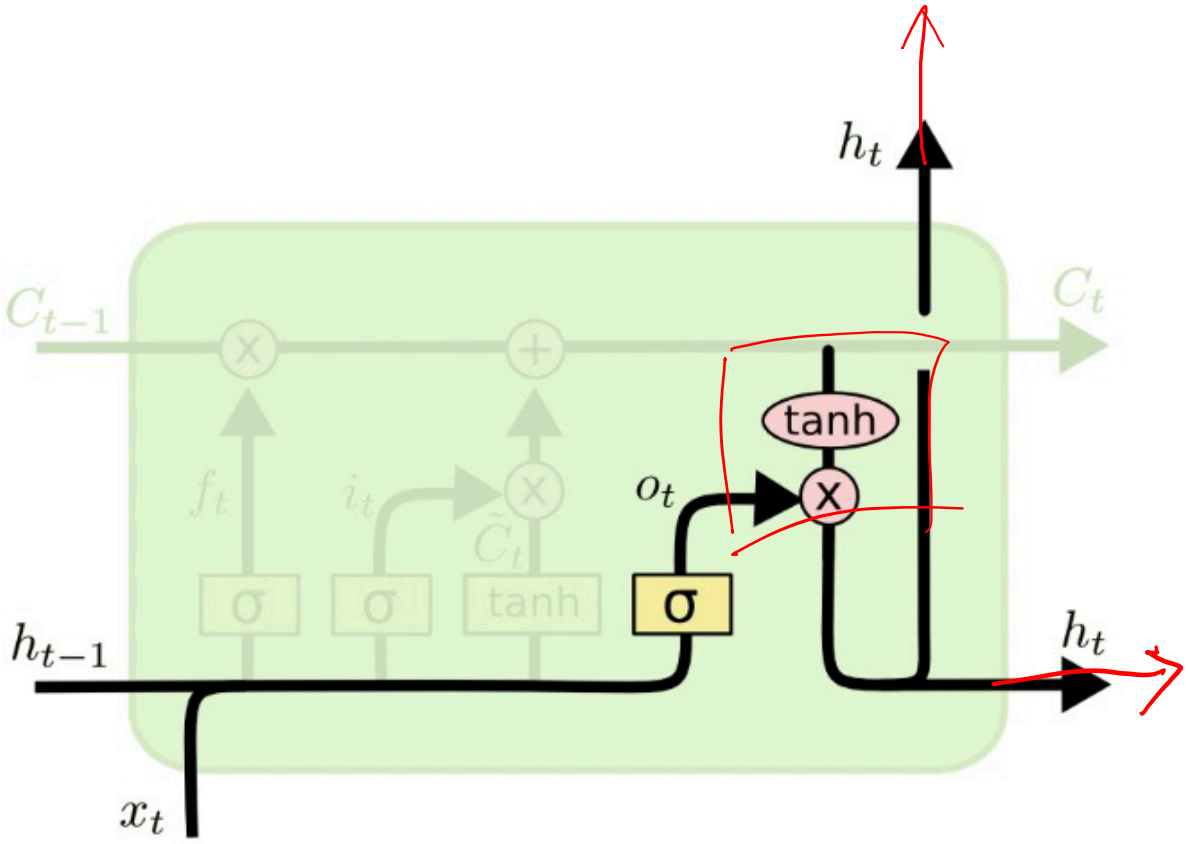
Memory



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

LSTM

output



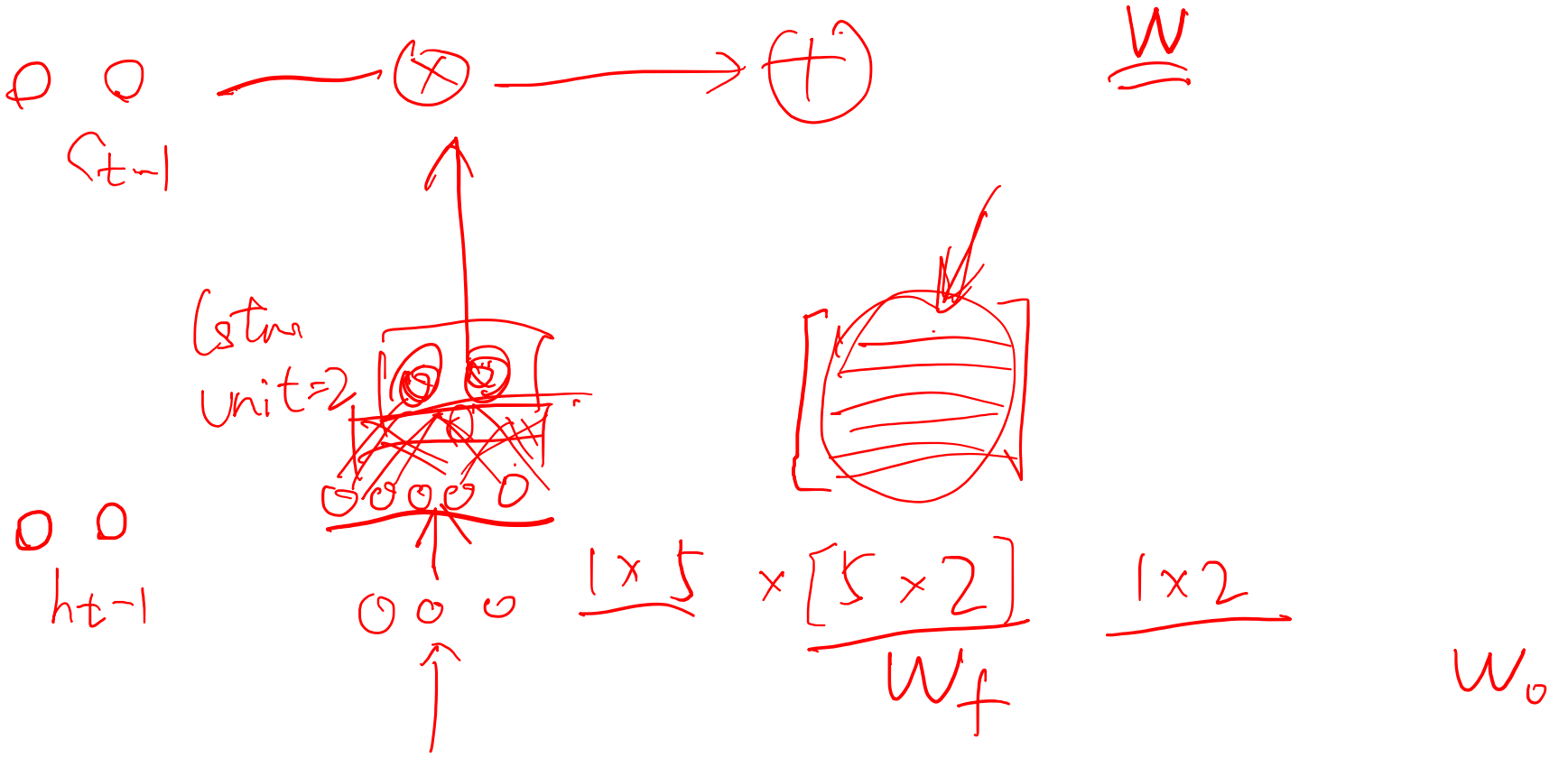
$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$$

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

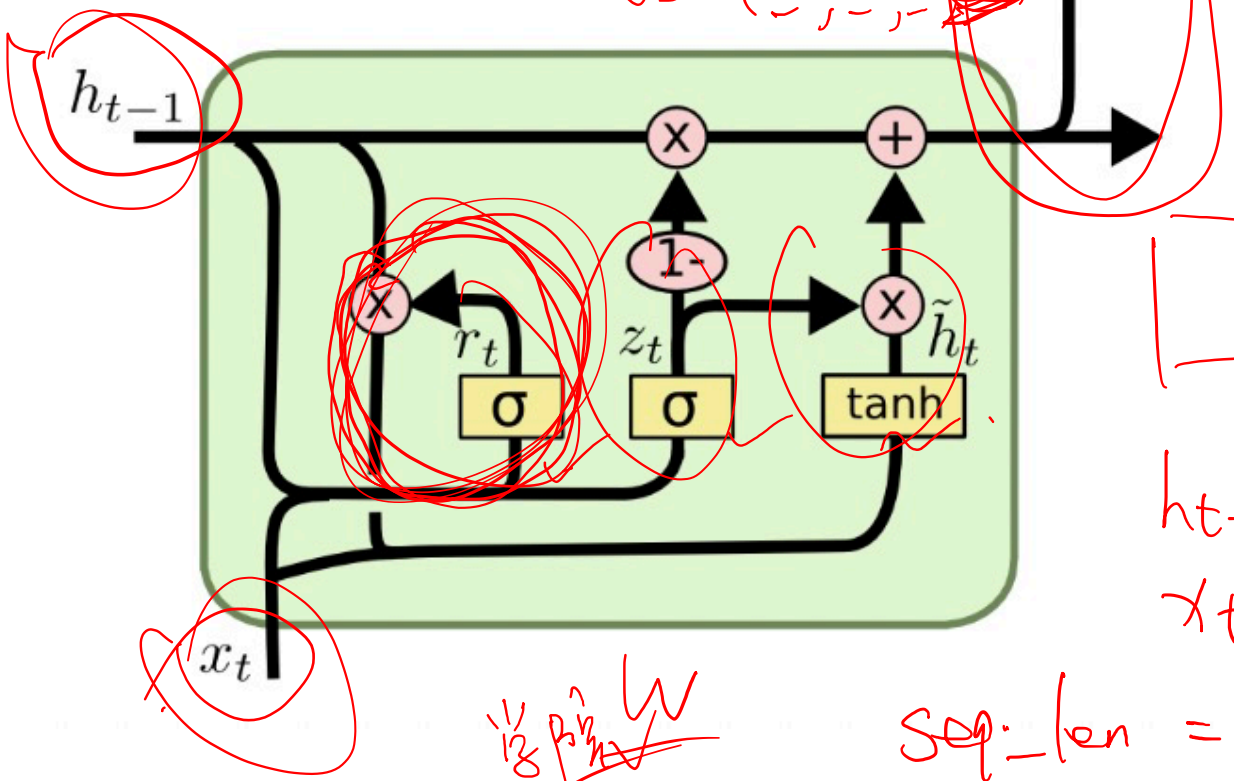
2 h_t C_t

$h_t, h_t, o_t = \mathrm{lstm}(\quad)$

LSTM



GRU / LSTM



Ct-1 → ht ht Ct

GRU ht-1

Xt

lstm(_,_,_)

[batch_size, seq_len, dim]

(1) GRU 快 [16, 200, 256]

(2) LSTM → GRU 数据集

(3) LSTM → GRU

ht-1 ht

Xt ht

速度高

seq_len = 200

batch_size = 16

dim = 256

作业

Bye ！