

# HCT NLP Week 1

课程介绍  
以及word2vec

# Outline

1. 课程安排
2. 词向量Word2vec
3. Skip-gram、CBOW模型
4. 项目介绍

# 1. 课程安排



## 问答摘要与推理

进行中

要求使用汽车大师提供的11万条技师与用户的多轮对话与诊断建议报告数据建立模型，基于对话文本、用户问题、车型与车系，输出包含摘要与推断的报告文本，考验模型的归纳总结与推断能力。

立即报名



算法

职业

目标



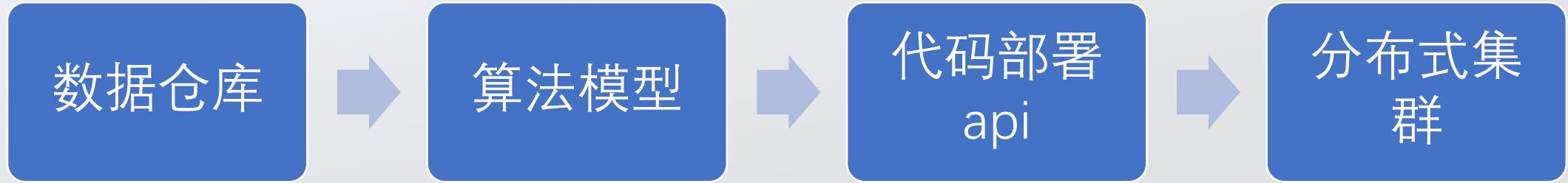
NLP/CV等



tob, toc

算法工程师/研究员

相关复合型人才



课程项目的定位  
算法能力的匹配

## Week1

## 问答摘要与推理-项目课程简介

- ✓ 项目介绍和课程安排
- ✓ 初始词向量Word2vec
- ✓ Skip-gram、CBOW模型

## Week2

## 问答摘要与推理-词向量实践

- ✓ 词向量计算优化方法
- ✓ 初识深度学习框架
- ✓ Gensim代码实践

## Week3

## 问答摘要与推理-Seq2Seq（一）

- ✓ RNN、LSTM、GRU
- ✓ Encoder-Decoder结构
- ✓ Attention机制

## Week4

## 问答摘要与推理-Seq2Seq（二）

- ✓ Seq2Seq训练与预测
- ✓ 模型Layer、Model构建
- ✓ Seq2Seq代码实践

## Week5

## 问答摘要与推理- Abstractive Text Summarization

- ✓ 深度学习框架图计算理论
- ✓ Beam search
- ✓ 生成式文本摘要训练代码实践

## Week6

## 问答摘要与推理- 项目模型算法提升

- ✓ OOV 和Word-repetition解决
- ✓ Training Strategies

## 相关代码实践

## Week7

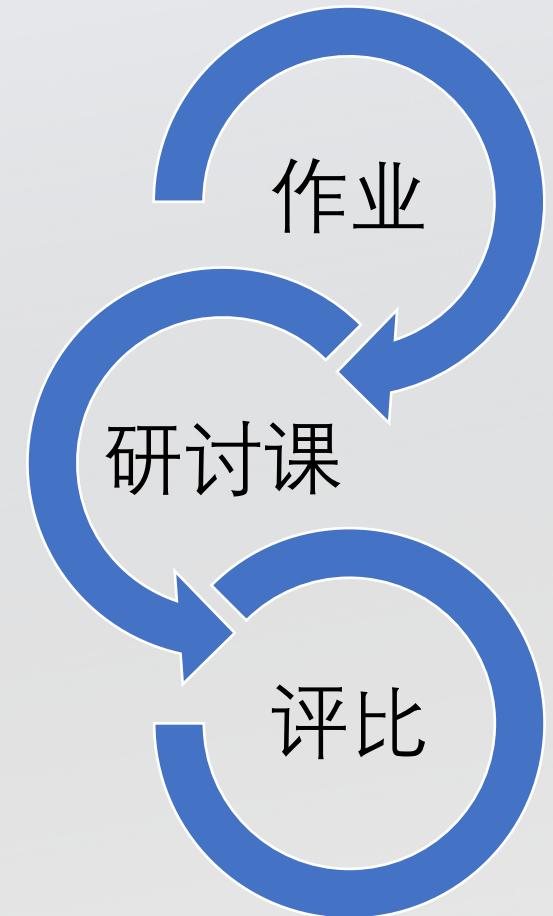
## 问答摘要与推理- Extractive Text Summarization

- ✓ 抽提式文本摘要基本方法
- ✓ 相关代码实践

## Week8

## 问答摘要与推理- 项目代码部署

- ✓ 项目结果评估
- 模型部署相关知识



## 2. 词向量Word2vec

# discrete symbols

我/爱/北京  $[1, 0, 0]$ ,  $[0, 1, 0]$ ,  $[0, 0, 1]$

I\_love\_you

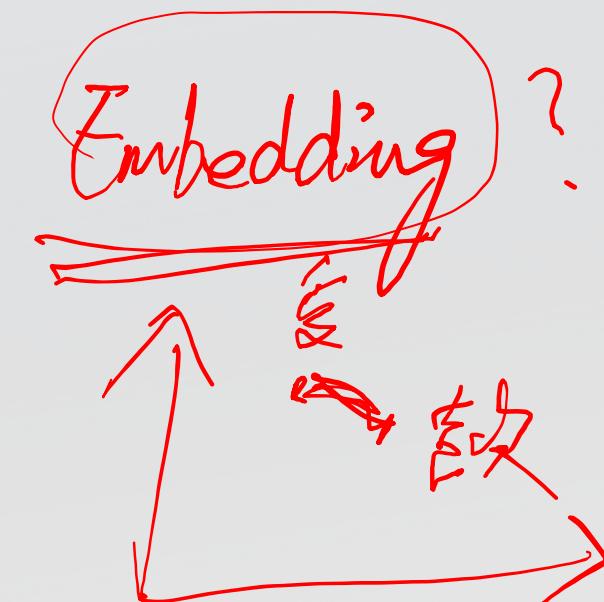
by their context

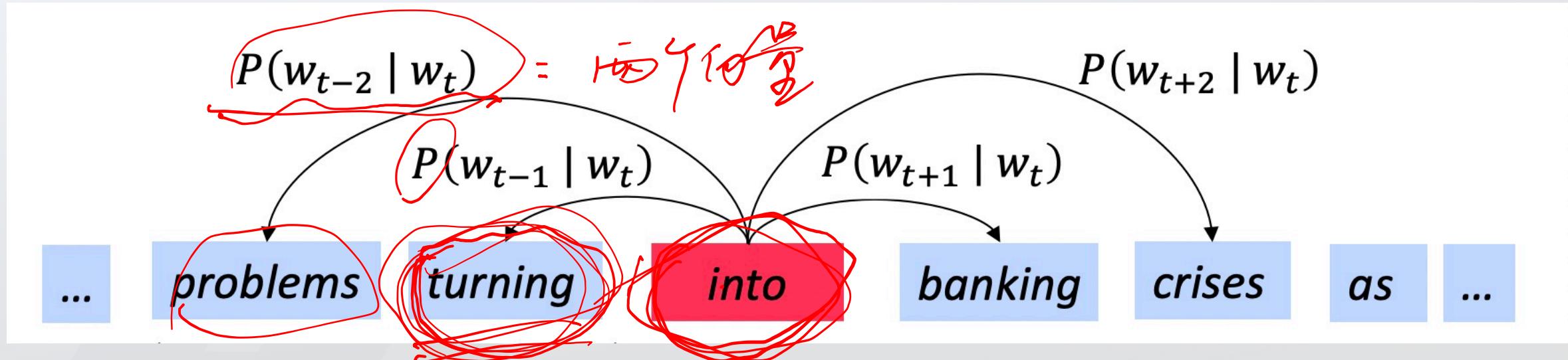


word2vec

$[$   
 $[$   
cat      kitten      dog  
 $]$   
 $]$

kitten - cat + dog  $\approx$  puppy ✓





*banking* =

$$\begin{bmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{bmatrix}$$

$$A = [0, 1, 2]$$

$$B = [1, 2, 3]$$

- 有一个很大的词表库
- 在词表中的每一个词都可以通过向量表征
- 有一个中心词c, 有一个输出词o
- 用词c和o的相似度来计算他们之间同时出现的概率
- 调整这个词向量来获得最大的输出概率

# Word2vec-objective function

对每个位置  $t=1, \dots, T$

窗口size为m

且包含<sup>包含</sup>该字<sup>该字</sup>的窗口

$$L(\theta) = \prod_{t=1}^T \prod_{\substack{j \leq m \\ j \neq 0}} P(w_{t+j} | w_t; \theta)$$

$$\log P_1 + \log P_2$$

交叉熵

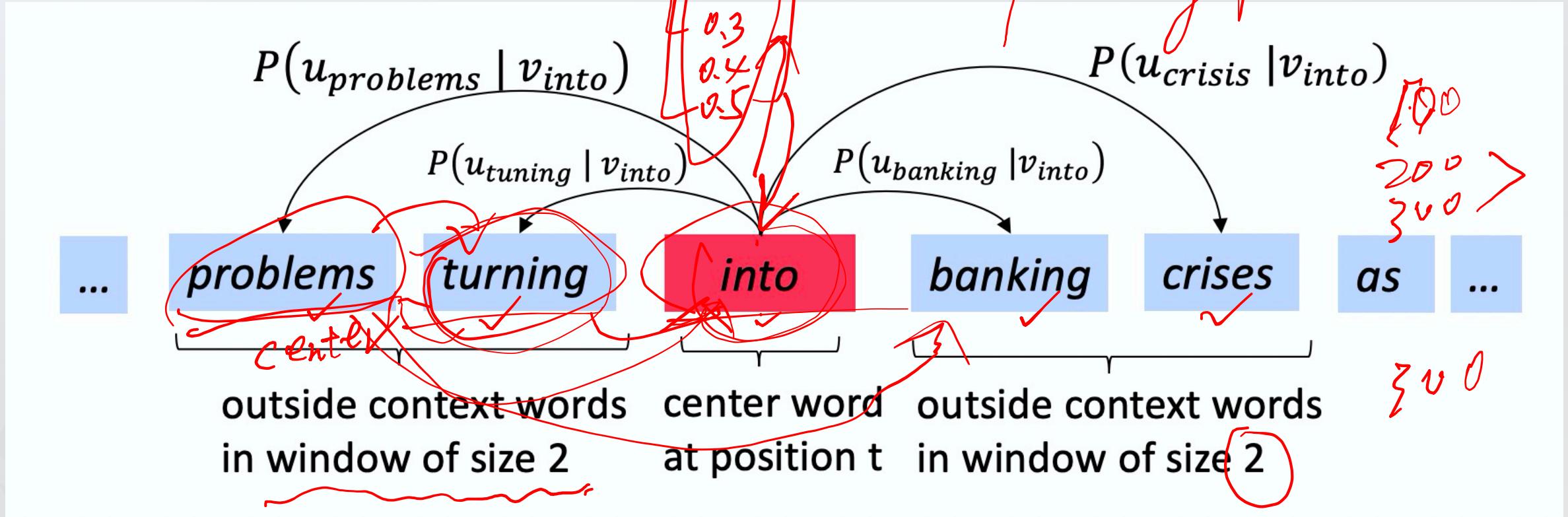
$$\log(P_1 P_2)$$

目标函数  $J(\theta) = -\frac{1}{T} \log(L(\theta)) = -\frac{1}{T} \sum_{t=1}^T \sum_{\substack{-m \leq j \leq m \\ j \neq 0}} \log P(w_{t+j} | w_t; \theta)$

$J(\theta)$  最小  $\rightarrow P$  最大

Fake 任务

# Word2vec-objective function



$$P(w_{t+j} | w_t; \theta)$$

$v_w$   $w \in$  center word.

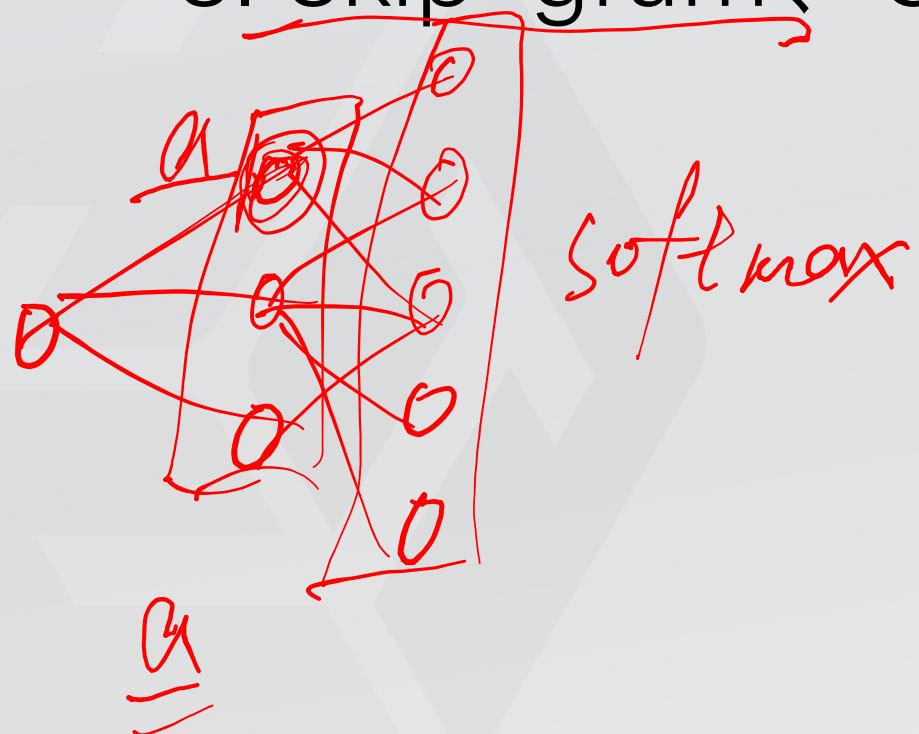
$u_w$   $w \in$  a context word

$$P(w_t | c) = \frac{\exp(u_0^T v_c)}{\sum_{w \in V} \exp(u_w^T v_c)}$$

多义词

continuous bags of words

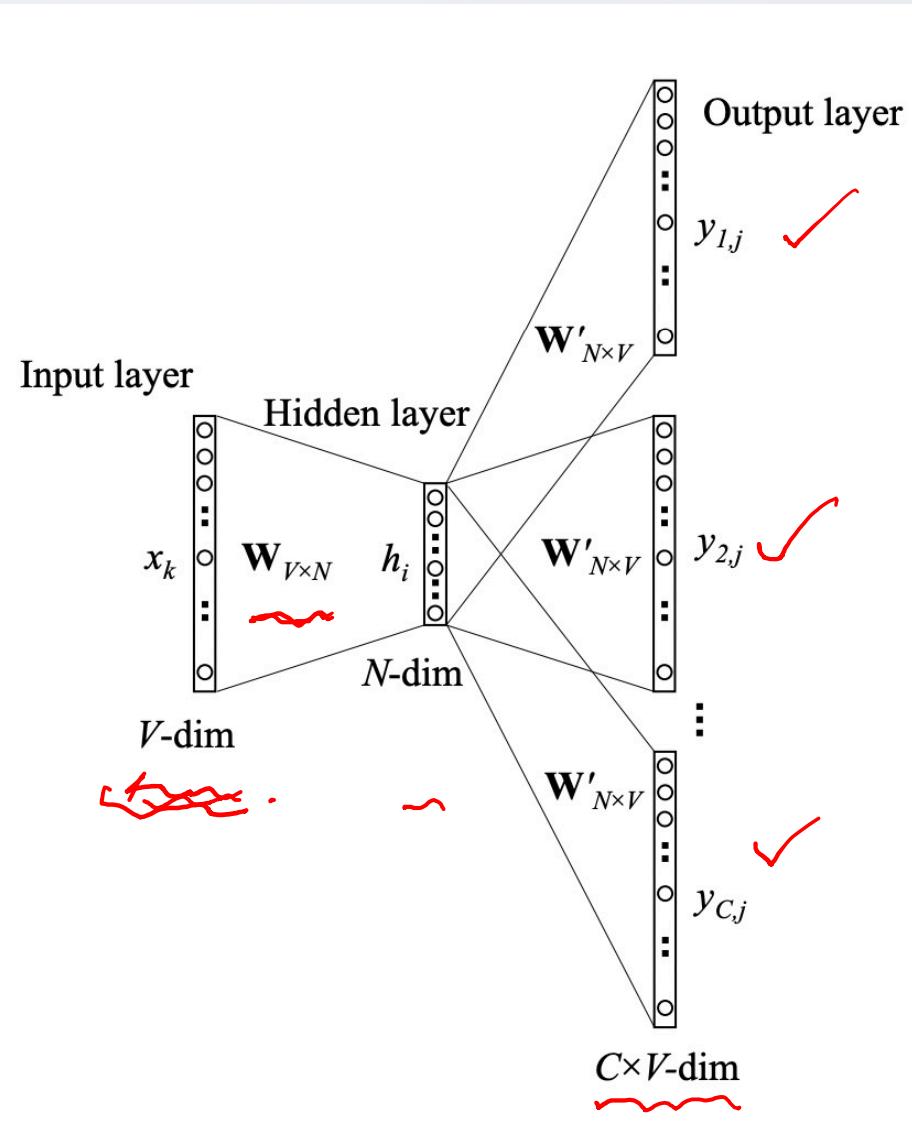
### 3. Skip-gram、CBOW模型



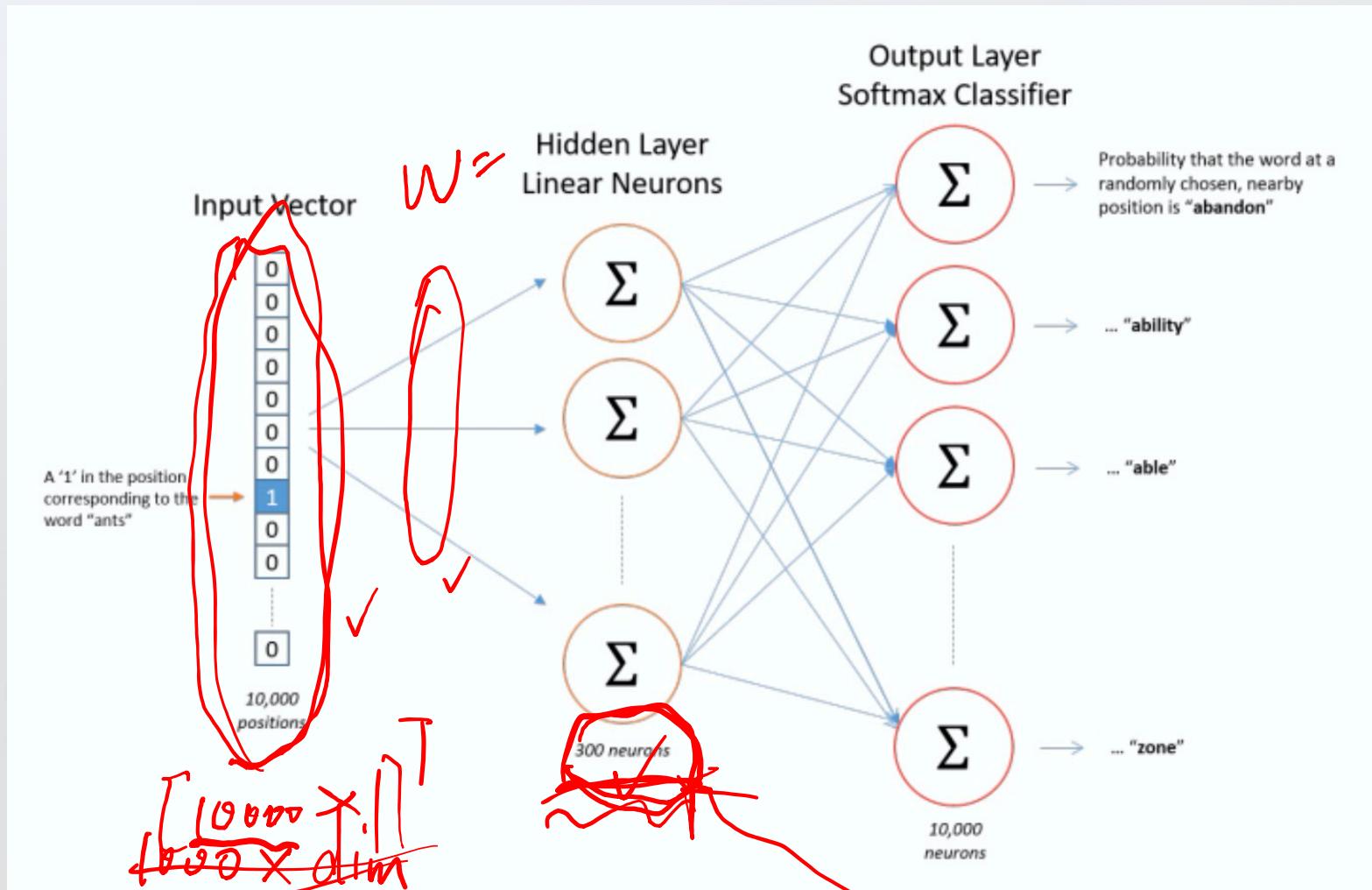
21:05 8

# Skip-gram模型

object loss



The dog barked at the mailman.  
skip\\_window=2  
['The', 'dog', 'barked', 'at']  
num\_skips = 2 (input, output)  
('dog', 'barked') ('dog', 'at')



① Vocab  $\checkmark 2^W$  one-hot

If  $\text{dog} = [0, 1, 0, 0, 0]$

$[ \dots \dots 1, 0, 0 ]$

$W$

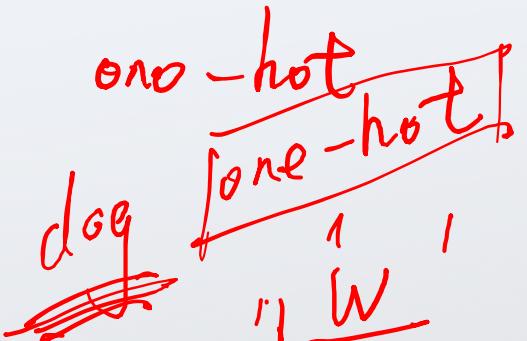
$[0, 1, 0, 0, 0] \times \begin{bmatrix} 1 & 2 & 4 & 1 \\ 2 & 3 & 5 & 7 \\ 3 & 7 & 6 & 8 \\ 5 & 8 & 3 & 2 \\ 1 & 2 & 3 & 0 \end{bmatrix}$

$=$

$5 \times 3 \approx$

$$[10000 \times 1] \times [1 \times 10000] = [10000 \times 300] = [1, 300]$$

① input



② 隐层

$$[0, 1, 0, 0, 0] \times \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 8 & 9 & \\ 10 & 12 & 13 \\ 14 & 15 & 17 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 4 & 5 & 6 & 0 & 0 \\ 8 & 9 & 0 & 0 & 0 \\ 10 & 12 & 13 & 0 & 0 \\ 14 & 15 & 17 & 0 & 0 \end{bmatrix}$$

$1 \times 5 \quad 1 \times 5$

$1 \times 10 \times 5 \quad 10 \times 5 \times 3$

$1 \times 10 \times 5 \times 3 \quad 10 \times 5 \times 3 \times 3$

③

$\text{Input: } [0, 1, 0, 0, 0] \times \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 8 & 9 & \\ 10 & 12 & 13 \\ 14 & 15 & 17 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 4 & 5 & 6 & 0 & 0 \\ 8 & 9 & 0 & 0 & 0 \\ 10 & 12 & 13 & 0 & 0 \\ 14 & 15 & 17 & 0 & 0 \end{bmatrix}$

$\text{Hidden Layer: } \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 4 & 5 & 6 & 0 & 0 \\ 8 & 9 & 0 & 0 & 0 \\ 10 & 12 & 13 & 0 & 0 \\ 14 & 15 & 17 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} W_1 & b_1 \\ W_2 & b_2 \\ \vdots & \vdots \\ W_n & b_n \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}$

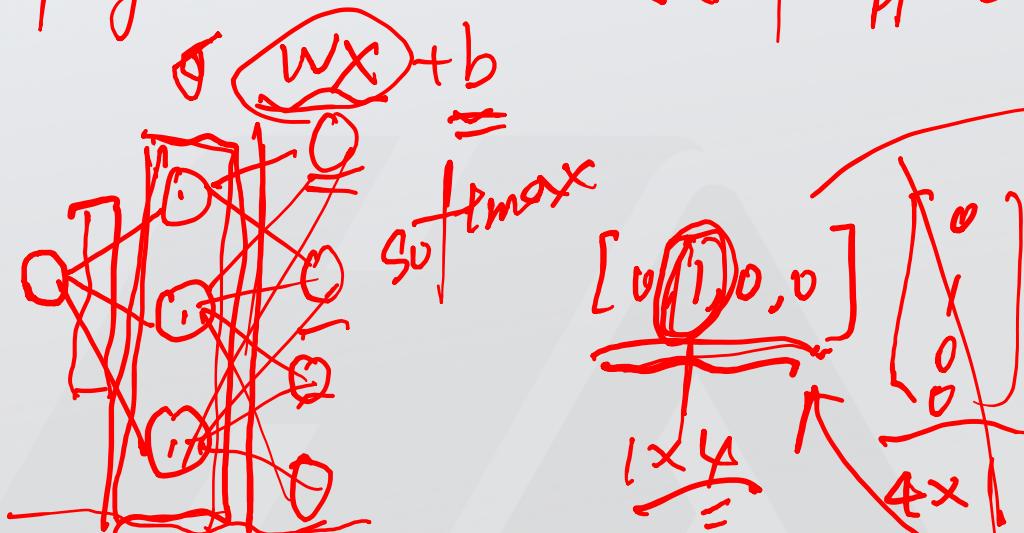
$P(o|c) = \frac{\exp(u_o^T v_c)}{\sum_{i=1}^n \exp(u_i^T v_c)}$

$\text{Output: } \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \rightarrow \text{softmax} = \begin{bmatrix} 5, 3, 2, 1, 4 \end{bmatrix} \rightarrow \text{softmax} = \begin{bmatrix} 0.2, 0.3, 0.2, 0.1, 0.2 \end{bmatrix}$

我 / 爱 / 北京 / 天安门  $\rightarrow$  vocab [ . . . . ]

$\rightarrow$  one-hot 我 [1, 0, 0, 0] 爱 [0, 1, 0, 0]

skip-gram  $\rightarrow$  center 邻近 | output



one-batch  $\leftarrow$

(我, 爱), (爱, 我), (我, 北京), (北京, 爱)  
(北京, 爱), (北京, 天安门), (天安门, 北京)

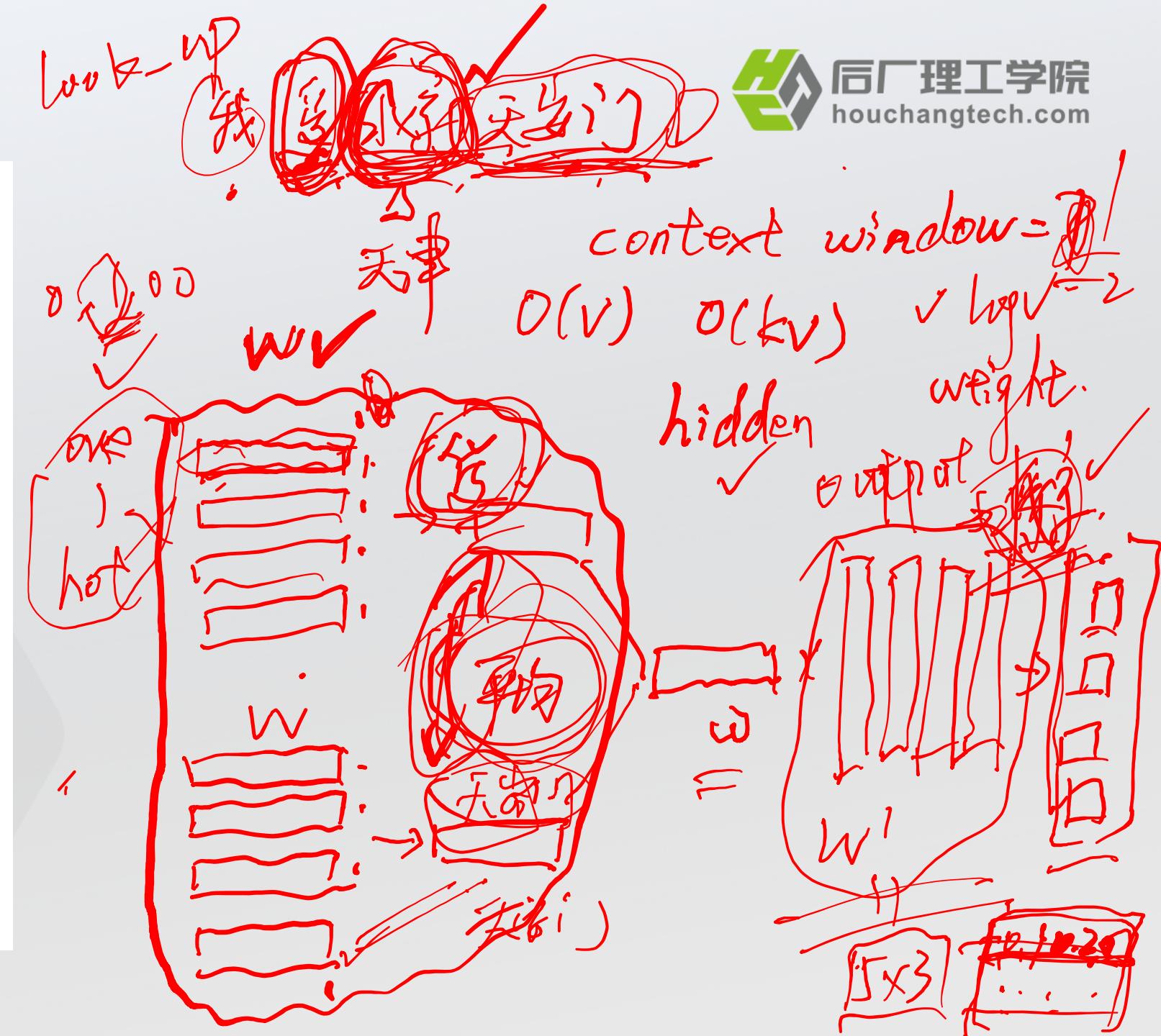
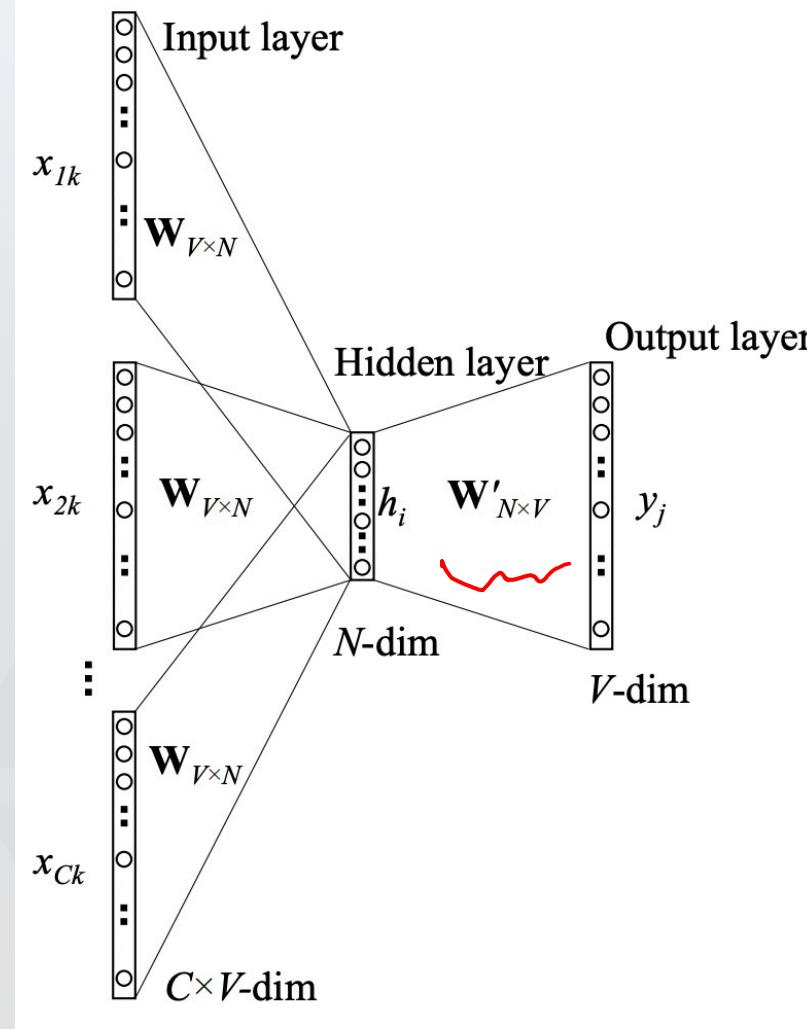
$$P = \frac{\exp(x_i)}{\sum_{i=1}^K \exp(x_i)}$$

batch\_size = 1 loss

batch\_size / num\_skips = 0 output

$$\begin{aligned} & \text{out.} \\ & \begin{bmatrix} 5 & 7 & 8 \end{bmatrix} = \begin{bmatrix} 5 & 7 & 8 & 9 \end{bmatrix} \\ & \begin{bmatrix} 5 & 7 & 8 & 9 \end{bmatrix} = \begin{bmatrix} e^5 & e^7 & e^8 & e^9 \end{bmatrix} \\ & \text{loss} = -\log(P(\text{我}, \text{北京} | \text{爱})) \\ & = -(\log P(\text{我} | \text{爱}) + \log P(\text{北京} | \text{爱})) \end{aligned}$$

# CBOW模型



## 4. 项目介绍

6

22:35见

Bye !