



数字媒体技术基础

Meng Yang

www.smartllv.com

SUN YAT-SEN University



**机器智能与先进计算教
育部重点实验室**



**智能视觉语言
学习研究组**

第七章 文本媒体信息表示

第七章 文本媒体信息表示



□ 文本是数字媒体的基础：

常见的媒体形态

文本
图片
音频
视频
.....



其中， 文本是涉及面最广的一种形态。

第七章 文本媒体信息表示

- 音频通常需要被转化为文本，再进行处理：



- 图片也可以转化为文本：



A horse carrying a large load of hay and two people sitting on it.



Bunk bed with a narrow shelf sitting underneath it.



The man at bat readies to swing at the pitch while the umpire looks on.

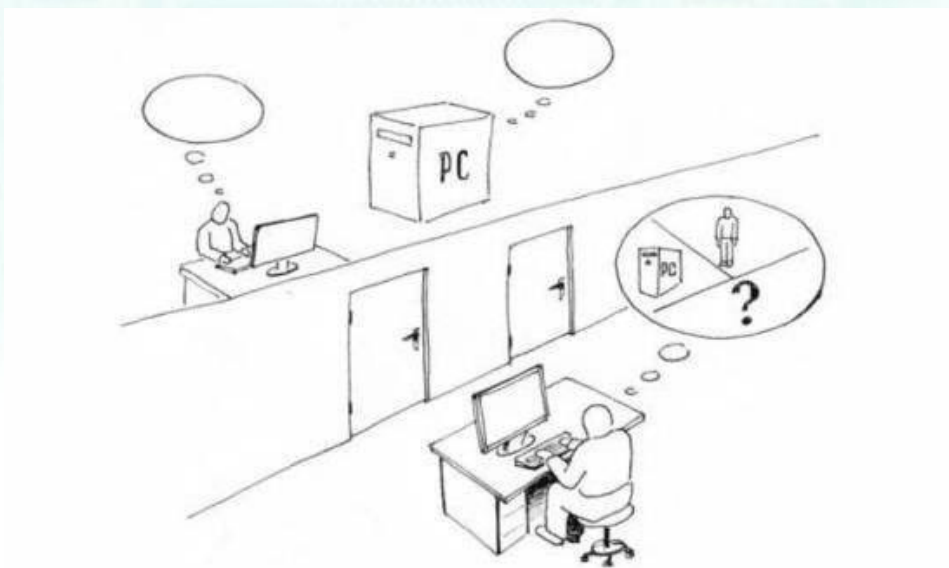
(图片描述生成)

Course Outline

- 7.1 语言模型
- 7.2 词的表示方法
 - 7.2.1 0-1表示
 - 7.2.2 词向量表示
- 7.3 文本的表示方法
 - 7.3.1 文本分词
 - 7.3.2 词袋模型
 - 7.3.3 TF-IDF表示
 - 7.3.4 基于词的聚合表示

7.1 语言模型

- 语言模型：用来计算一个句子的概率
(判断一个句子是否通顺合理、像是人说的)
- E. g. , “我在学习” vs “学习在我”



语言模型——上下文无关模型

- 认为句子中的所有词相互独立，互不相关。
- 在计算某个词的概率时，仅仅考虑当前词本身的概率，不考虑上下文的任何其他词。
- 也称“Unigram语言模型”或“一元语言模型”。

我	在	学	习
---	---	---	---

相互独立，
互不相关

我
在
学
习



语言模型——上下文无关模型

E. g.

假设训练语料是这个简单的语料：

我	来	自	中	山	大	学	，	我	爱	学	习	自	然	语	言	处	理	。
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

就能得到各词的频率：

词	频率
我	2/19
来	1/19
自	1/19
中	1/19
.....

那么，如果想估计下面的句子的概率：

我	爱	学	习
---	---	---	---

$$p = p(\text{我}) * p(\text{爱}) * p(\text{学}) * p(\text{习}) = \frac{2}{19} * \frac{1}{19} * \frac{1}{19} * \frac{1}{19}$$





语言模型——上下文无关模型

也可以直接估计某个单词的概率：

问题？

?	爱	学	习
---	---	---	---

$$p(?=我) = 2/19$$

$$p(?=来) = 1/19$$

$$p(?=自) = 1/19$$

.....

词	频率
我	2/19
来	1/19
自	1/19
中	1/19
.....

由于过于独立，估计句子中任何一个位置是某个词的概率，都会得到相同的结果（例如，预测句子中任何一个位置是“我”的概率，都是2/19）：

我	?	学	习
---	---	---	---

$$p(?=我) = 2/19$$

$$p(?=来) = 1/19$$

$$p(?=自) = 1/19$$

.....

预测结果不佳



语言模型——上下文无关模型



并且，当计算这两个句子的概率时：

我	爱	学	习
---	---	---	---

$$p = p(\text{我}) * p(\text{爱}) * p(\text{学}) * p(\text{习}) = \frac{2}{19} * \frac{1}{19} * \frac{1}{19} * \frac{1}{19}$$

习	学	爱	我
---	---	---	---

$$p = p(\text{习}) * p(\text{学}) * p(\text{爱}) * p(\text{我}) = \frac{1}{19} * \frac{1}{19} * \frac{1}{19} * \frac{2}{19}$$

两个句子的出现概率竟相同，显然不合理。

Unigram语言模型仅考虑各个单词本身的概率，没有考虑句子是否通顺合理。



语言模型——N-gram模型




N-gram语言模型不再简单地将各词独立看待。

当N=2时，为Bi-gram，每个词的概率受到前一个词的影响：

我	爱	学	习
---	---	---	---

$$p = p(\text{我} | \langle s \rangle) * p(\text{爱} | \text{我}) * p(\text{学} | \text{爱}) * p(\text{习} | \text{学})$$


$$p(\text{习} | \text{学}) = \frac{c(\text{学}, \text{习})}{c(\text{学})}$$



语言模型——N-gram模型



我	爱	学	习
---	---	---	---

$$p(\text{习}|\text{学}) = \frac{c(\text{学}, \text{习})}{c(\text{学})}$$

显然, $p(\text{习}|\text{学}) > p(\text{学}|\text{习})$

学	习	爱	我
---	---	---	---

$$p(\text{学}|\text{习}) = \frac{c(\text{习}, \text{学})}{c(\text{习})}$$

由于考虑了上下文, 这两个句子的概率将不会相同。



问题： N的取值？ N-gram的个数？

如果词库有20,000个词：

N	N-gram个数
2	400,000,000
3	8,000,000,000,000
4	1.6×10^{17}
.....

N取得越大：

- 考虑到的相关的词数越多，模型更加合理。

然而：

- 需要的语料更加庞大，或是数据过于稀疏；
- 时间复杂度高。

- N-gram的数据稀疏问题：由于训练数据不足，出现N-gram频次为0的情况，导致估算新句子的概率为0.

E. g. , 当计算5-gram时,

我	爱	学	习	这	门	课	程
---	---	---	---	---	---	---	---

如果“我爱学习这”没在语料中出现过：

$$p(\text{这}) = p(\text{这}|\text{我爱学习}) = \frac{c(\text{我爱学习这})}{c(\text{我爱学习})} = 0$$

语言模型——数据稀疏

❑ 为什么N-gram会出现数据稀疏问题？

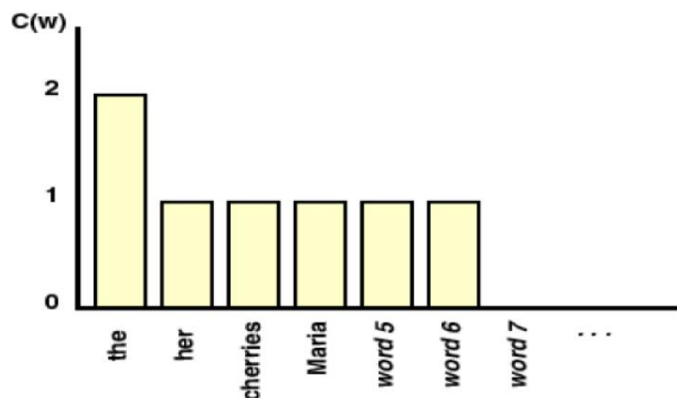
如果词库有20,000个词：

N	N-gram个数
2	400,000,000
3	8,000,000,000,000
4	1.6×10^{17}
.....

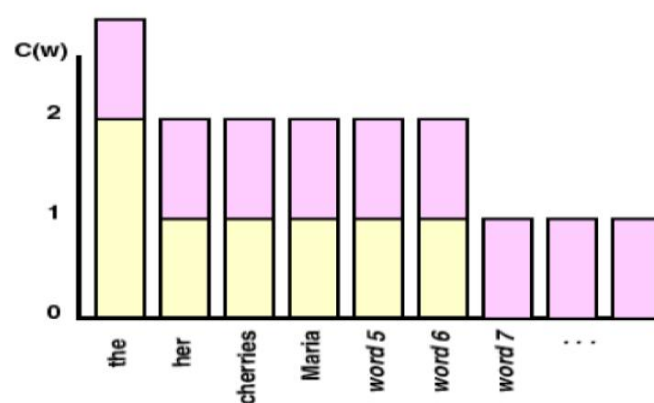
- 当N取2的时候，要完全避免数据稀疏问题，训练语料至少需要出现400,000,000个N-gram，**比较难**达到；
- 当N取4的时候，要完全避免数据稀疏问题，训练语料至少需要出现 1.6×10^{17} 个N-gram，**几乎不可能**达到；

数据稀疏问题的解决方法：

- 从数据维度解决：增大语料
- 从模型维度解决：降低 N 的值
- 从数据处理维度解决：平滑。拉普拉斯平滑将词库中所有词的计数值加一，从而避免计算词频时分子为0：



平滑



语言模型——N-pos模型

实际上，许多词出现的概率条件依赖于它前面词的语法功能。因此，可以不再具体统计每个词，而是统计每个词的词性。

- **N-pos (part-of-speech) 模型**：一个词出现的概率条件依赖于前面N个词的词性。

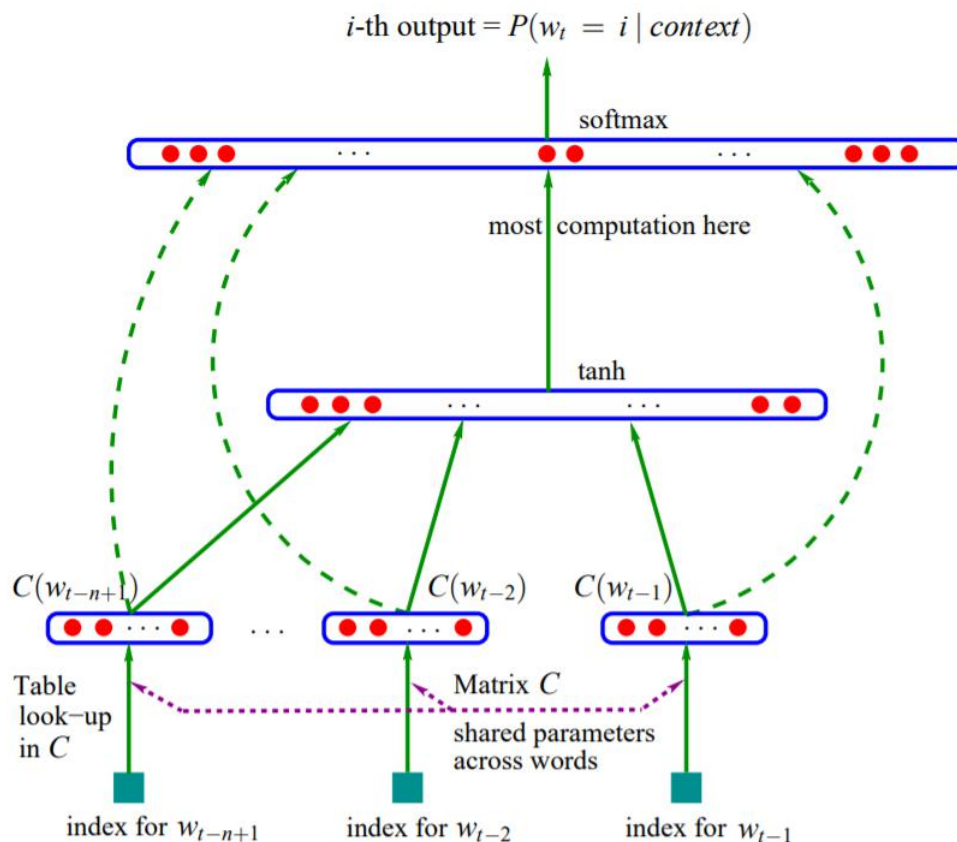
我	在	这	里
---	---	---	---

$$\begin{aligned} p(\text{这}) &= p(\text{这} | g(\text{我})g(\text{在})) \\ &= p(\text{这} | \text{人称代词, 动词}) \\ &= \frac{c(\text{人称代词, 动词, 这})}{c(\text{人称代词, 动词})} \end{aligned}$$

语言模型——神经网络语言模型



先给每个词在连续空间中赋予一个词向量，再通过神经网络去学习这种分布式表征。

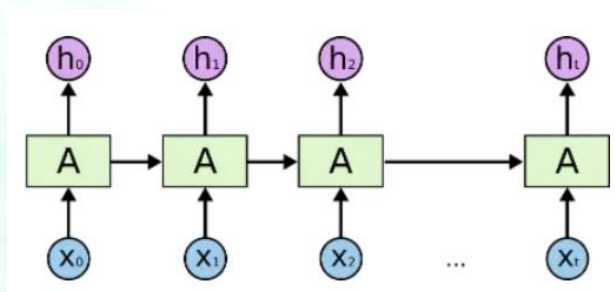


Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model[J]. The journal of machine learning research, 2003, 3: 1137-1155.

语言模型——神经网络语言模型



或是采用循环神经网络（RNN），更加符合自然语言具有方向和顺序的特性：



以及如今火热的BERT模型：



Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model[C]//Eleventh annual conference of the international speech communication association. 2010.

Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.





神经网络语言模型的参数来是多少？

时间	机构	模型名称	模型规模	数据规模	计算时间
2018.6	<u>OpenAI</u>	GPT	110M	4GB	3 天
2018.10	Google	BERT	330M	16GB	50 天
2019.2	<u>OpenAI</u>	GPT-2	1.5B	40GB	200 天
2019.7	Facebook	<u>RoBERTa</u>	330M	160GB	3 年
2019.10	Google	T5	11B	800GB	66 年
2020.6	<u>OpenAI</u>	GPT-3	175B	2TB	355 年
2021	预计		~1000B	~10TB	~1000 年



语言模型——常见模型的对比



模型	优势	劣势
上下文无关模型	非常少的语料	统计信息不充分，精确度太低
N-gram模型	划分精细，效果较好	需要语料较多，数据稀疏问题
N-pos模型	需要的语料比N-gram少，模型参数空间小	条件概率依赖于词性，划分不够精细
神经网络语言模型	泛化性强，缓解了数据稀疏问题	复杂度高



7.2 词的表示方法

0-1表示

- 词语是人类的抽象总结，是符号形式的（比如中文、英文、拉丁文等等）。要想将让电脑处理文字，需要将文字转换成数字，那么如何做到？

- 最简单的方法——逐个编号

他	我	你	学习	游戏	男人	女人	西瓜	冬瓜
0	1	2	3	4	5	6	7	8

- 但是，不同的词有不同的数值，凭什么“你”的值大于“我”的值？

0-1表示

- 0-1编码：假设词典大小为 n ，某个词在词典中的位置为 k 。创建一个 n 维向量，第 k 维置1，其余维全都置0：

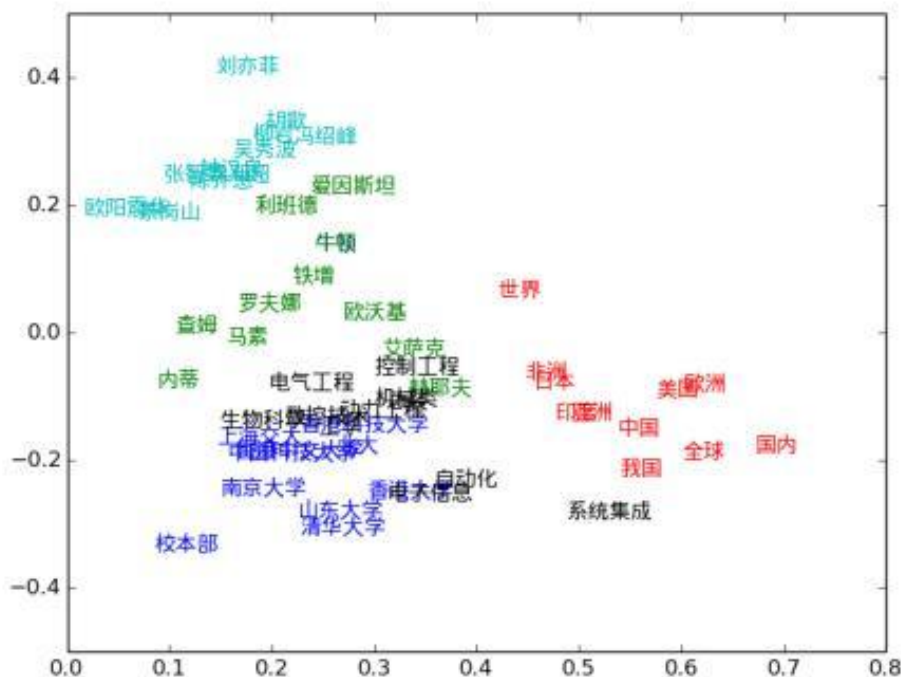
他	我	你	学习
[1,0,0,0,.....]	[0,1,0,0,.....]	[0,0,1,0,.....]	[0,0,0,1,.....]

问题？

- 但是，仍然没法表示词与词之间的关系，例如“男人”和“女人”
- 即，能区分不同的词，但不能表达词的含义

词向量表示

- 分布式词向量：将词映射到一个数学空间里，含义相近的词，在词的表示空间中也处于相近的位置。



和“电脑”相似的词为：

('个人电脑', 0.789919912815094)
('晶片', 0.7822093963623047)
('计算机', 0.7611304521560669)
('硬体', 0.759285032749176)
('应用程式', 0.7552173733711243)
('数位', 0.7427370548248291)
('软体', 0.7418122291564941)
('作业系统', 0.7361161708831787)
('微处理器', 0.7316363453865051)
('手机', 0.7304278016090393)

得到的大多是和计算机硬件相关的词。

和“香蕉”相似的词为：

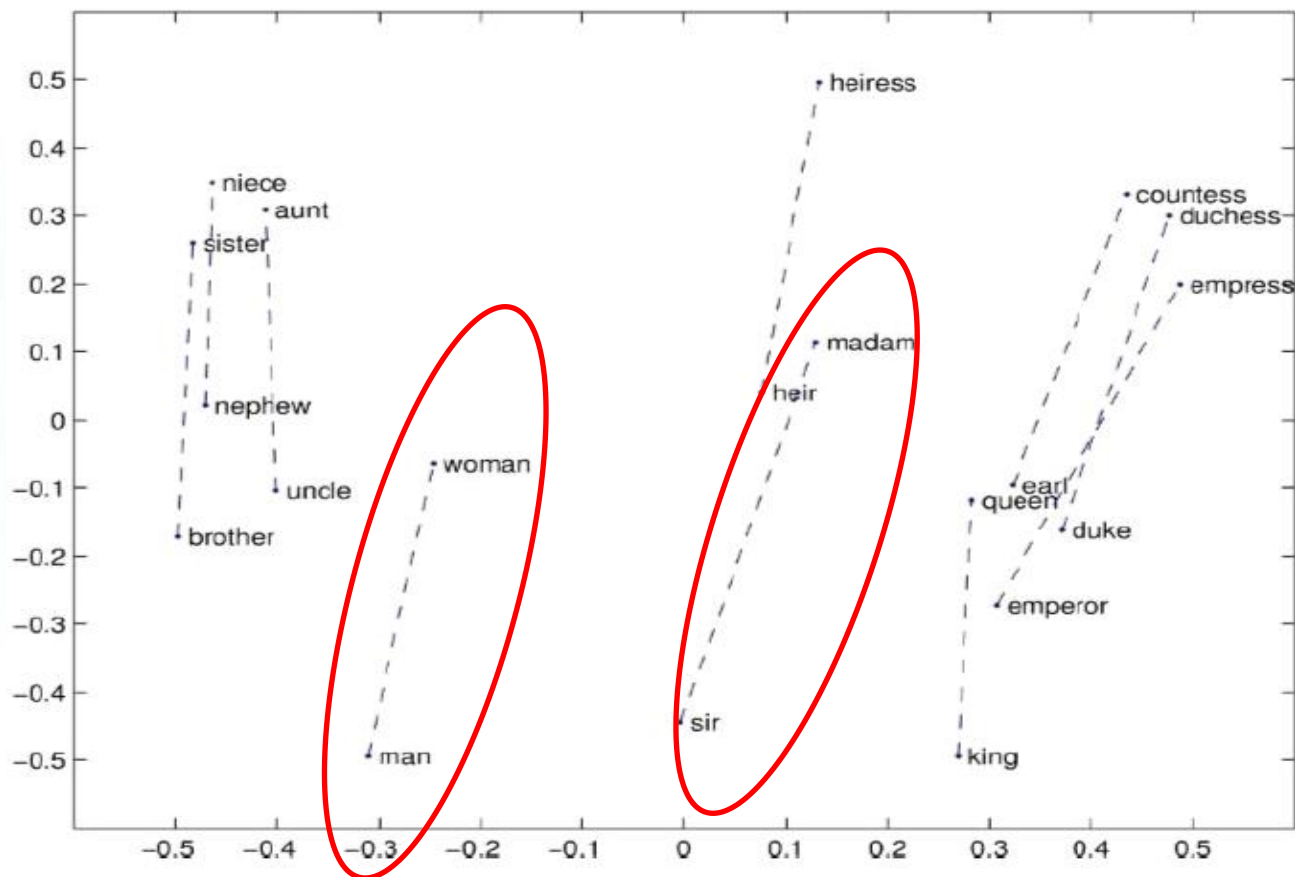
('马铃薯', 0.8815776705741882)
('玉米', 0.8770232200622559)
('椰子', 0.8681879043579102)
('水果', 0.8671873211860657)
('豆类', 0.8657207489013672)
('大豆', 0.862913966178894)
('花生', 0.8584224581718445)
('柑橘', 0.8531949520111084)
('蔬菜', 0.8528922200202942)
('洋葱', 0.8478729128837585)

得到的大多是水果和蔬菜。

词向量表示



man 平移到 women \approx sir 平移到 madam



词向量表示

- 那么，如何得到这种表示？
- 如果让我们从头学习中文，假设我们在教材里看到了这两句话：

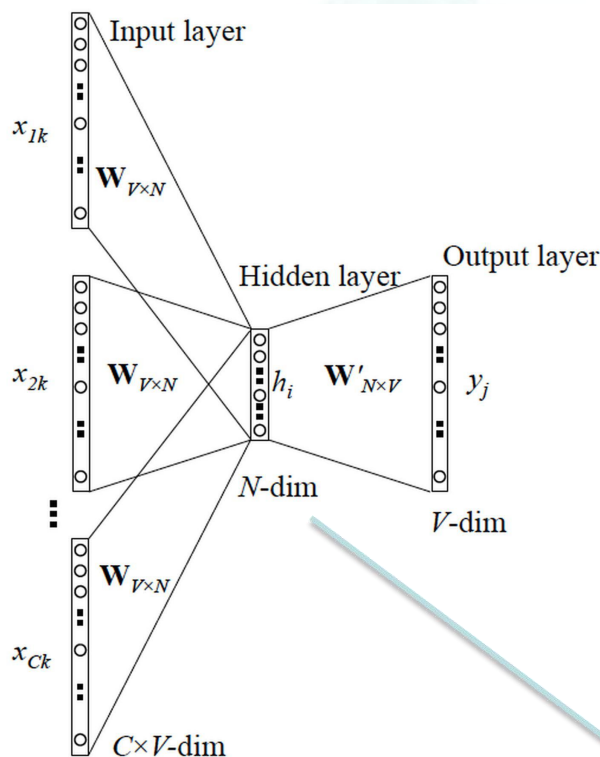
我	非常	喜欢	学习	。
---	----	----	----	---

我	非常	爱	学习	。
---	----	---	----	---

- 也许就能认为“喜欢”和“爱”的含义是接近的，因为它们周围能具有相同的词。

词向量表示

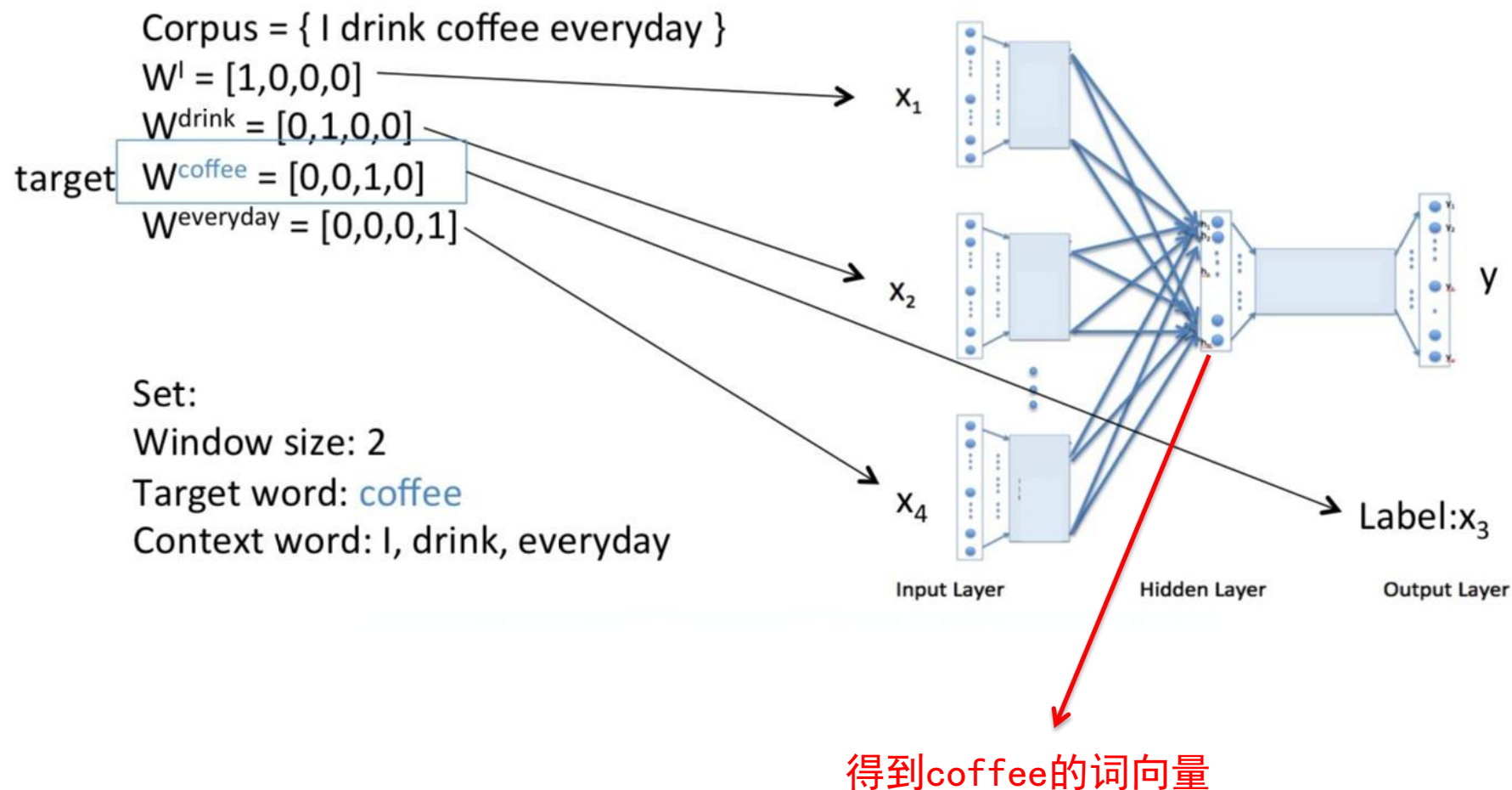
- Word2vec: 用周围的词来预测自身，训练得到的隐含层表示作为该词的词向量。
- 主要思想：从周围的词中学习自己的表示。如果词*i*和词*j*周围的词是相似的，那么词*i*和词*j*也是相似的。



作为 $w(t)$ 的词向量

词向量表示

训练过程:

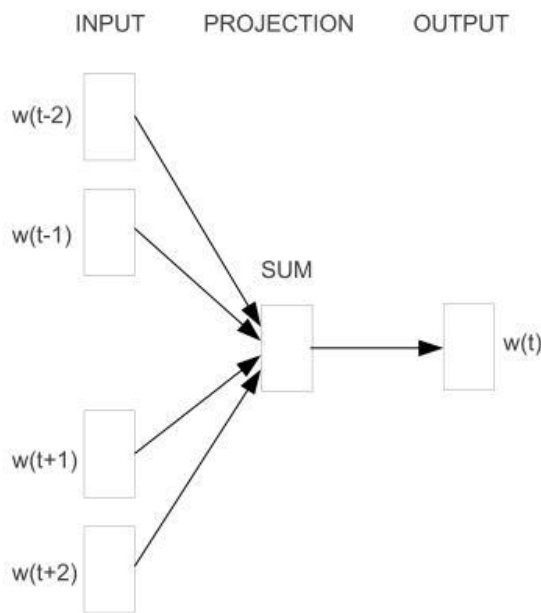


词向量表示

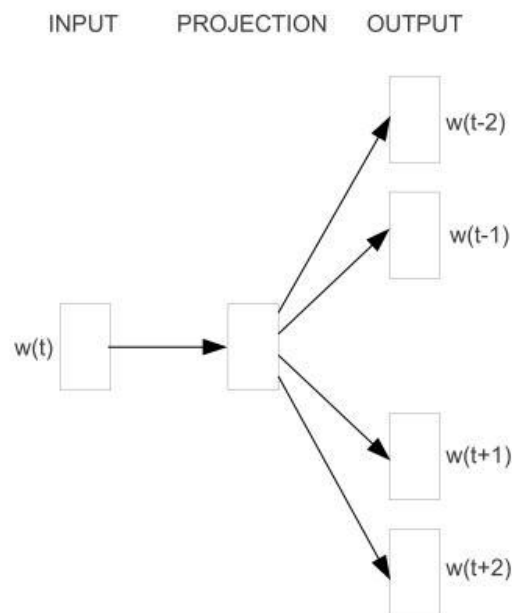


事实上，Word2vec提出了两种训练方式：

- ❑ CBOW模型：拿一个词语的上下文作为输入，来预测这个词语本身；
- ❑ Skip-gram模型：用一个词语作为输入，来预测它的上下文。



CBOW



skip-gram



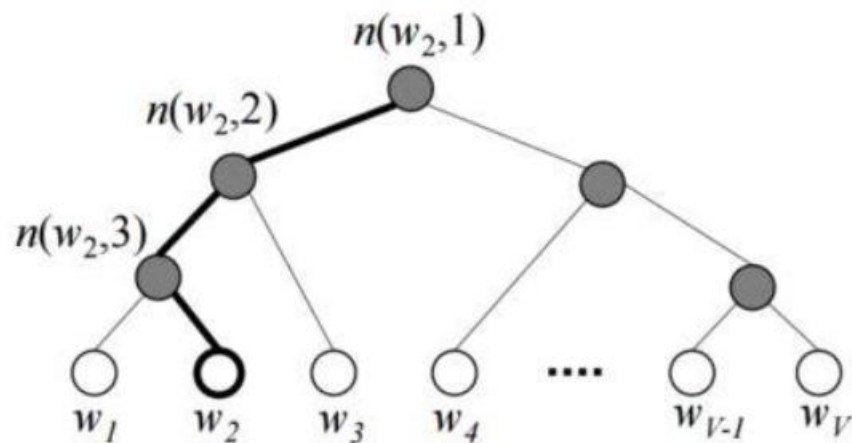
词向量表示

然而，词库很大时，计算softmax十分困难。

问题：如何处理？

问题？

Word2vec提出了Hierarchical softmax把N分类问题拆解成 $\log(N)$ 次二分类。



$$O(N) \rightarrow O(\log N)$$

根据学校课堂纪律的要求



请同学们坐在前五排

