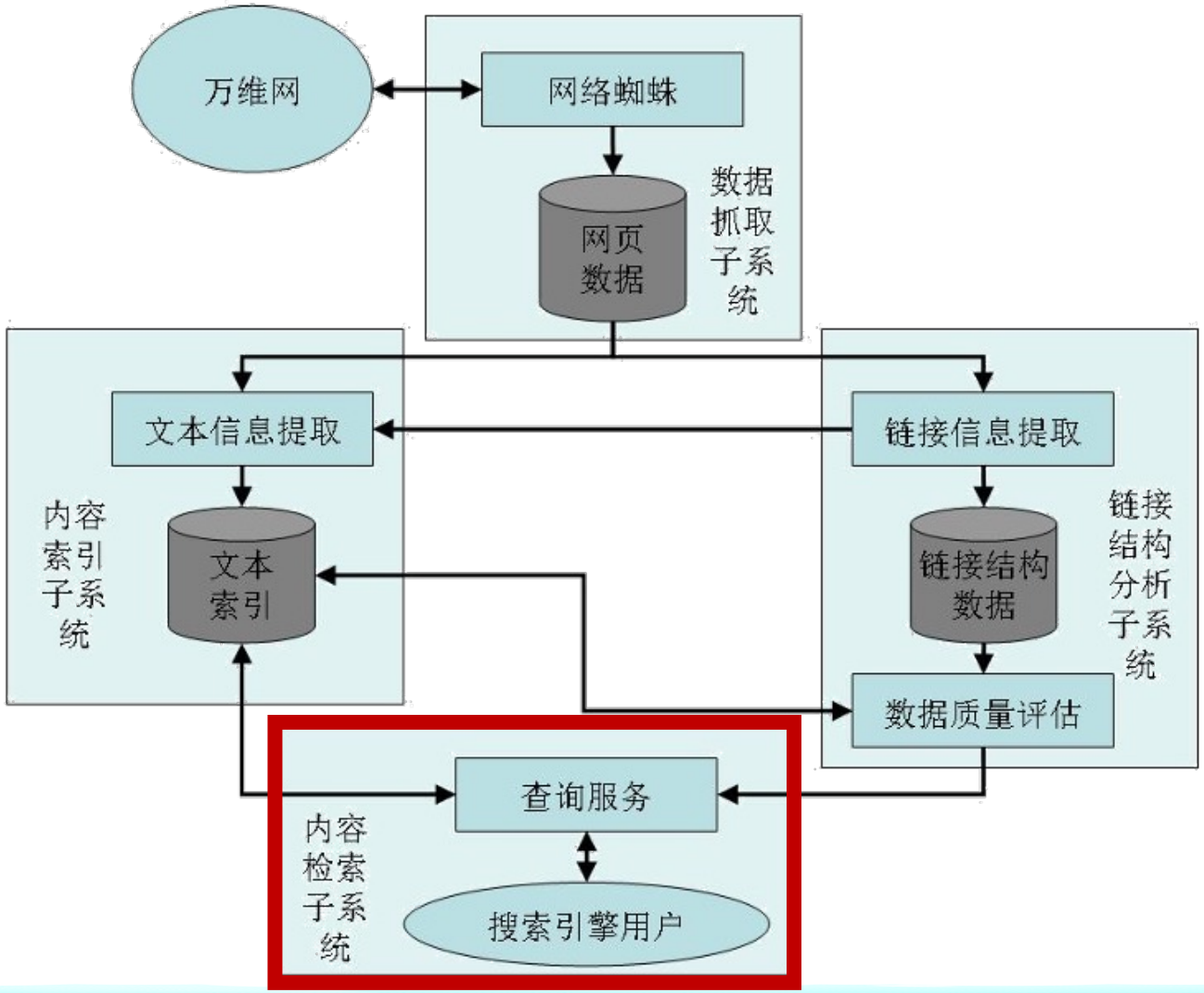
第7节. 内容检索子系统与经典算法

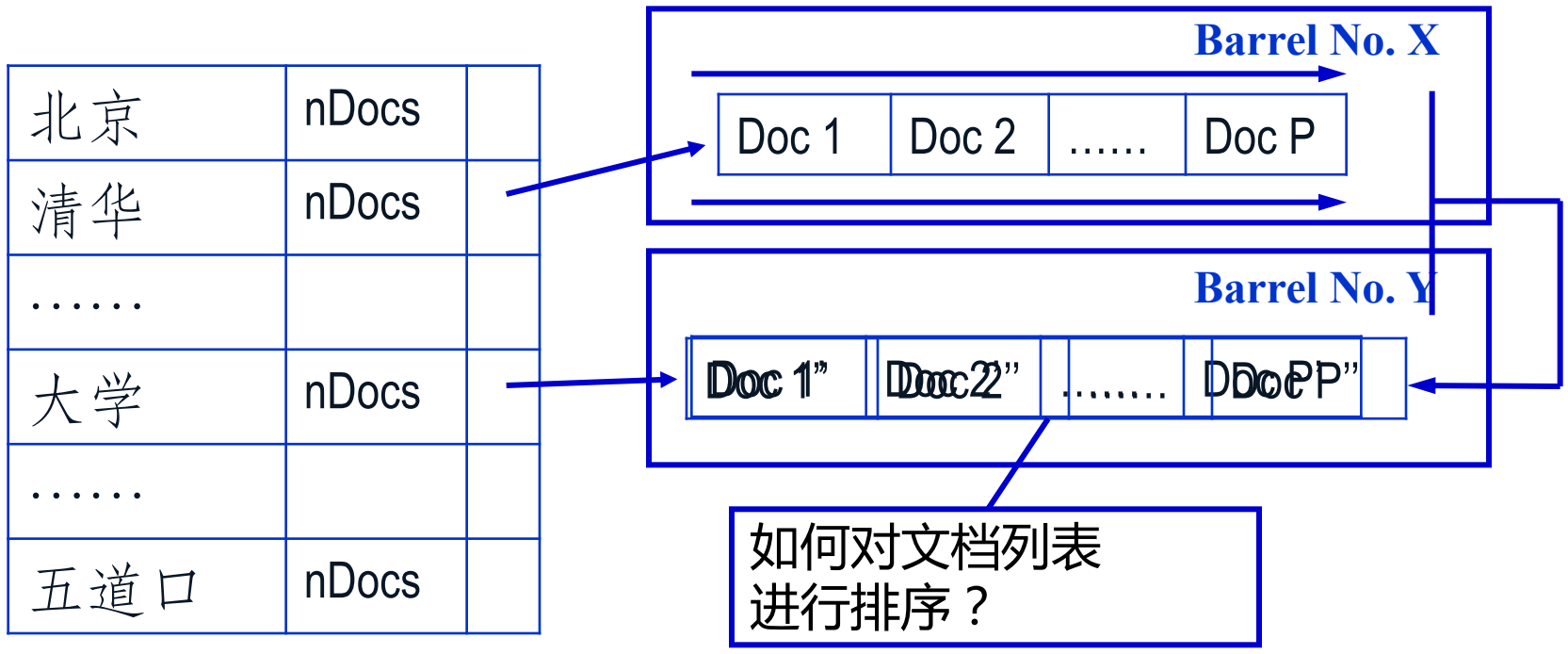
艾清遥
清华大学计算机系
清华大学互联网司法研究院
2023年4月4日

搜索引擎体系结构


万维网
网络蜘蛛
网页数据
数据抓取子系统
文本信息提取
内容索引子系统
文本索引
链接信息提取
链接结构数据
链接结构分析子系统
数据质量评估
查询服务
内容检索子系统
搜索引擎用户

引言：内容检索子系统

- 用户输入查询“清华大学”



理想：根据满足用户信息需求的情况排序

引言：内容检索子系统

- 包括文本信息检索模型在内的各种搜索引擎排序依据的计算方法，介绍内容检索子系统的核心算法和运行方式。
- 重点：
  - 如何计算文档与查询的内容相关性
  - 搜索引擎进行文档排序的特征依据

文本信息检索模型

- 检索模型：确定文档与查询的内容相关性
  - 如何表示构成检索系统的两个要素：文档和查询。
  - 确定在模型中，如何定义和计算文档和检索之间的关系
  - 假设文档为D，查询为q，它们之间相关程度的函数为$f(q, D)$

如何计算$f(q, D)$?

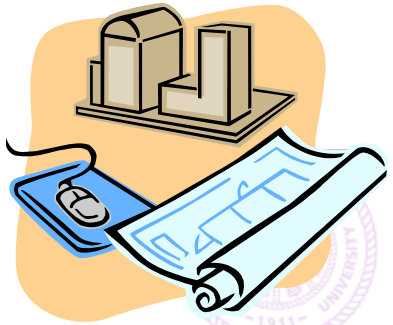
经典检索模型框架


文档相关性
文档相似性
向量空间模型
[Salton et al. ‘75]
概率分布模型
[Wong & Yao ‘89]
相关性概率
回归模型
[Fox ‘83]
文档生成模型
经典概率模型
[Robertson & Spark-Jones ’76]
查询生成模型
统计语言模型
[Ponte & Croft ‘98]
概率推理模型
推理网络模型
[Turtle & Croft ’91]
概率概念空间模型
[Wong & Yao ’95]

本节概要

文本检索经典模型

1. 向量空间模型 (Vector space model)
2. 经典概率模型 (Classic probabilistic model)
3. 统计语言模型 (Statistic language modeling)



向量空间模型

*"The more two representations agreed in **given elements and their distribution**, the higher would be the probability of their representing similar information."*

--"若两个表示的元素及其分布越相似，那它们表示相似信息的概率就越大"

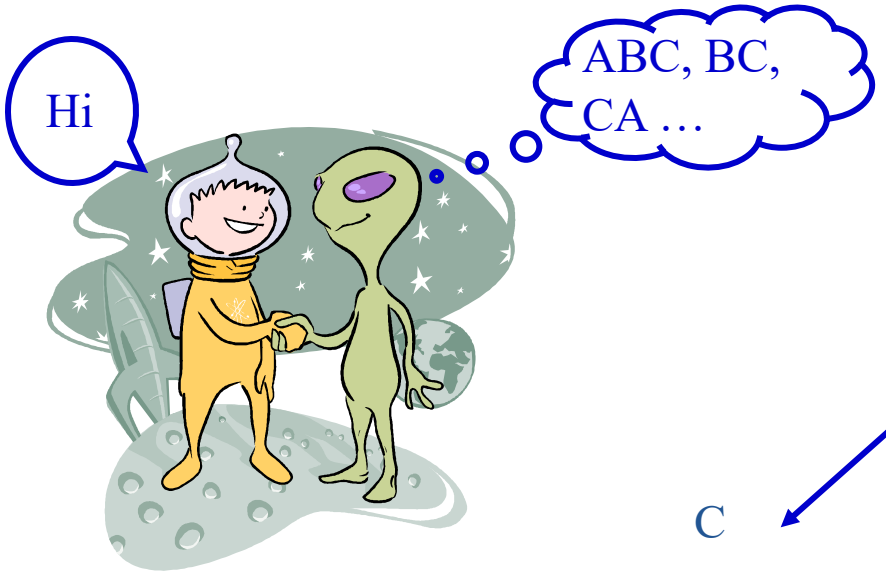
Hans Peter Luhn
(1896-1964)

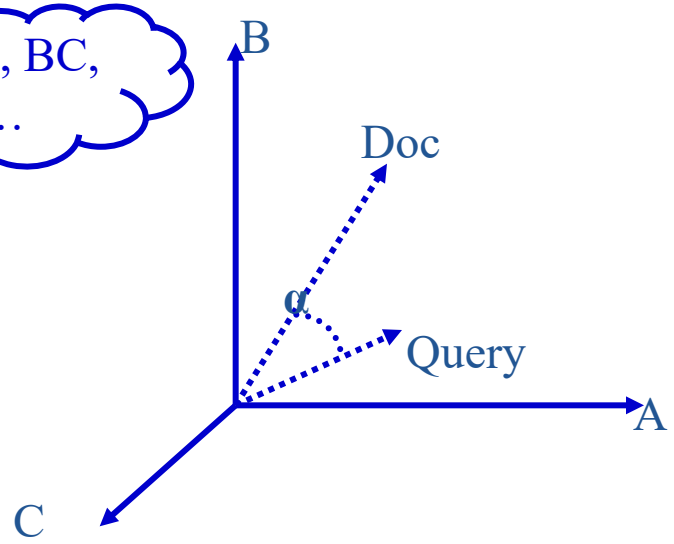
相似≈相关

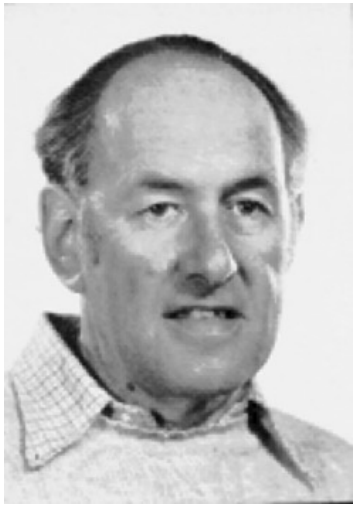
H. P. Luhn. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1(4), 1957: 309-317.

向量空间模型

- Vector Space Model (VSM, not SVM!)
  - 提出： Salton, 1975, Communications of the ACM
  - 一种计算事物之间相似度的通用方法
  - 词汇量有限的外星人



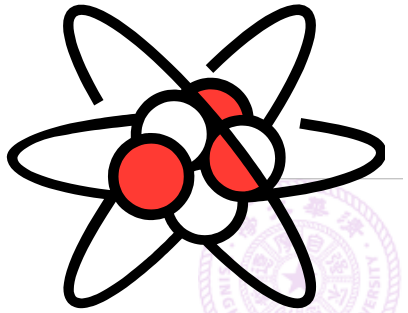




Gerard Salton
(1927-1995)
(Gerard Salton
Award, 1983)

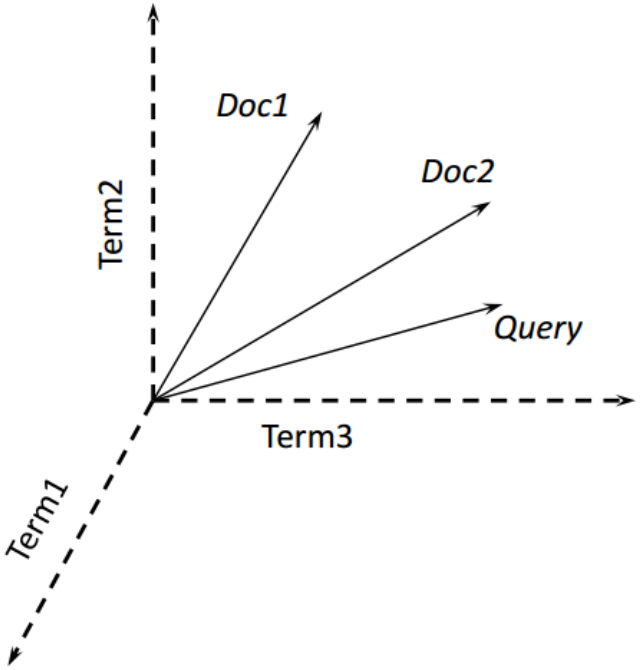
向量空间模型

- 基本思想：
  - 事物可以用一些共同的原子性的基本单元表示
  - 如果将每个原子单元视为基向量来构建一个n维空间，则事物就对应了n维空间的一个向量
  - 用向量之间的差别来度量事物的相似度
- 应用范围：
  - 文本处理任务：检索、分类、聚类
  - 原子单元：特征项(词项)



向量空间模型

- 将文档和查询表示为向量
  - 用一个词项（或者短语）表示一个原子单元
  - 每一个词项对应向量中的一维
  - 向量中的不同维度之间构成正交关系
  - 每一维的数值w_i表示对应词项t_i的权重

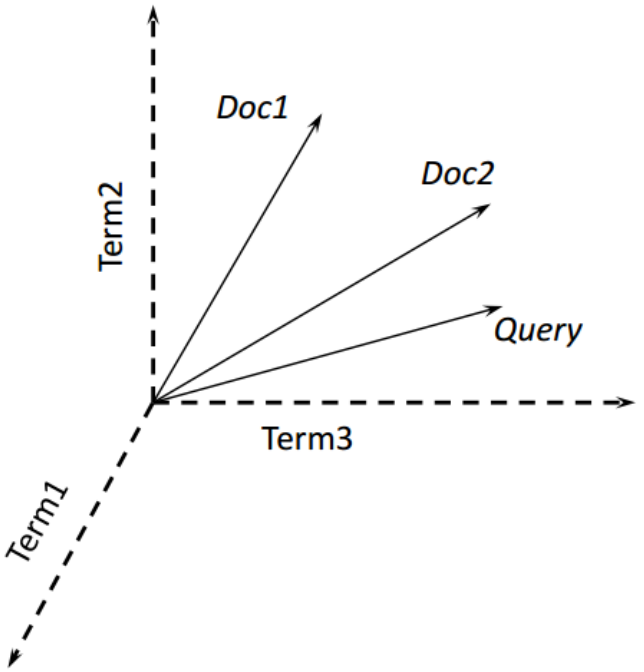


$$\overrightarrow{D_i} = [w_{i,1}, w_{i,2}, \ldots, w_{i,n}]$$

$$\vec{q} = [w_{q,1}, w_{q,2}, \ldots, w_{q,n}]$$

向量空间模型

- 将文档和查询表示为向量
  - 用一个词项（或者短语）表示一个原子单元
  - 每一个词项对应向量中的一维
  - 向量中的不同维度之间构成正交关系
  - 每一维的数值w_i表示对应词项t_i的权重
  - *权重如何计算?*



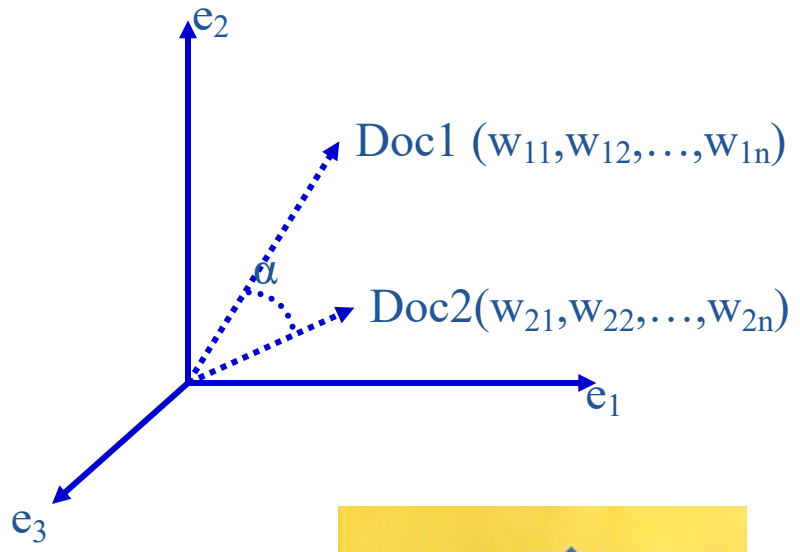
词频模型 *Term Frequency (TF)*

$$w_i = \text{freq}(t_i, D)$$

向量空间模型

- 相似度衡量方式
  - 假设：各词项重要性相同
  - 文档间的内积距离

$$Sim(D_1, D_2) = \sum_{k=1}^{n} w_{1,k} \cdot w_{2,k}$$



  - 文档间的余弦距离

$$Sim(D_1, D_2) = \cos\alpha = \frac{\sum_{k=1}^{n} w_{1,k} \cdot w_{2,k}}{\sqrt{(\sum_{k=1}^{n} w_{1,k}^2)(\sum_{k=1}^{n} w_{2,k}^2)}}$$

内积

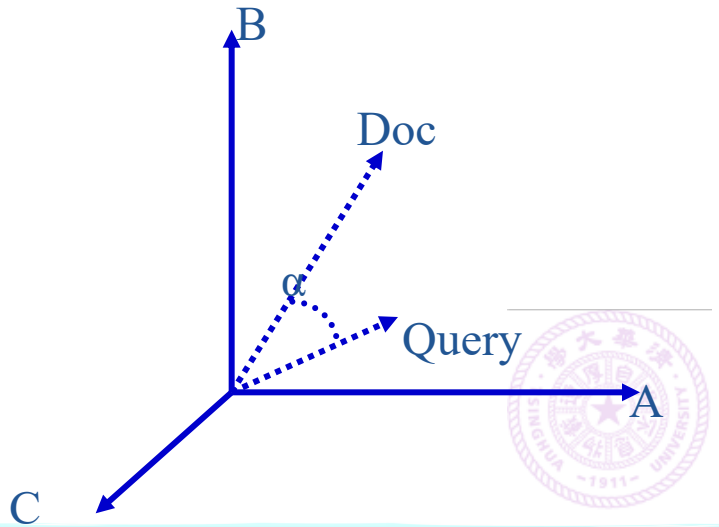
模

1.1 向量空间模型

- 相似度衡量方式
  - 文档与查询间的距离

$$Sim(D_j, q) = \frac{D_j \cdot q}{|D_j| \cdot |q|} = \frac{\sum_{k=1}^{n} w_{j,k} \cdot w_{q,k}}{\sqrt{(\sum_{k=1}^{n} w_{j,k}^2)(\sum_{k=1}^{n} w_{q,k}^2)}}$$

  - 技术挑战：
    - 查询向量与文档向量不匹配
    - 不同词项间并非独立关系
    - 不同词项的重要程度不同

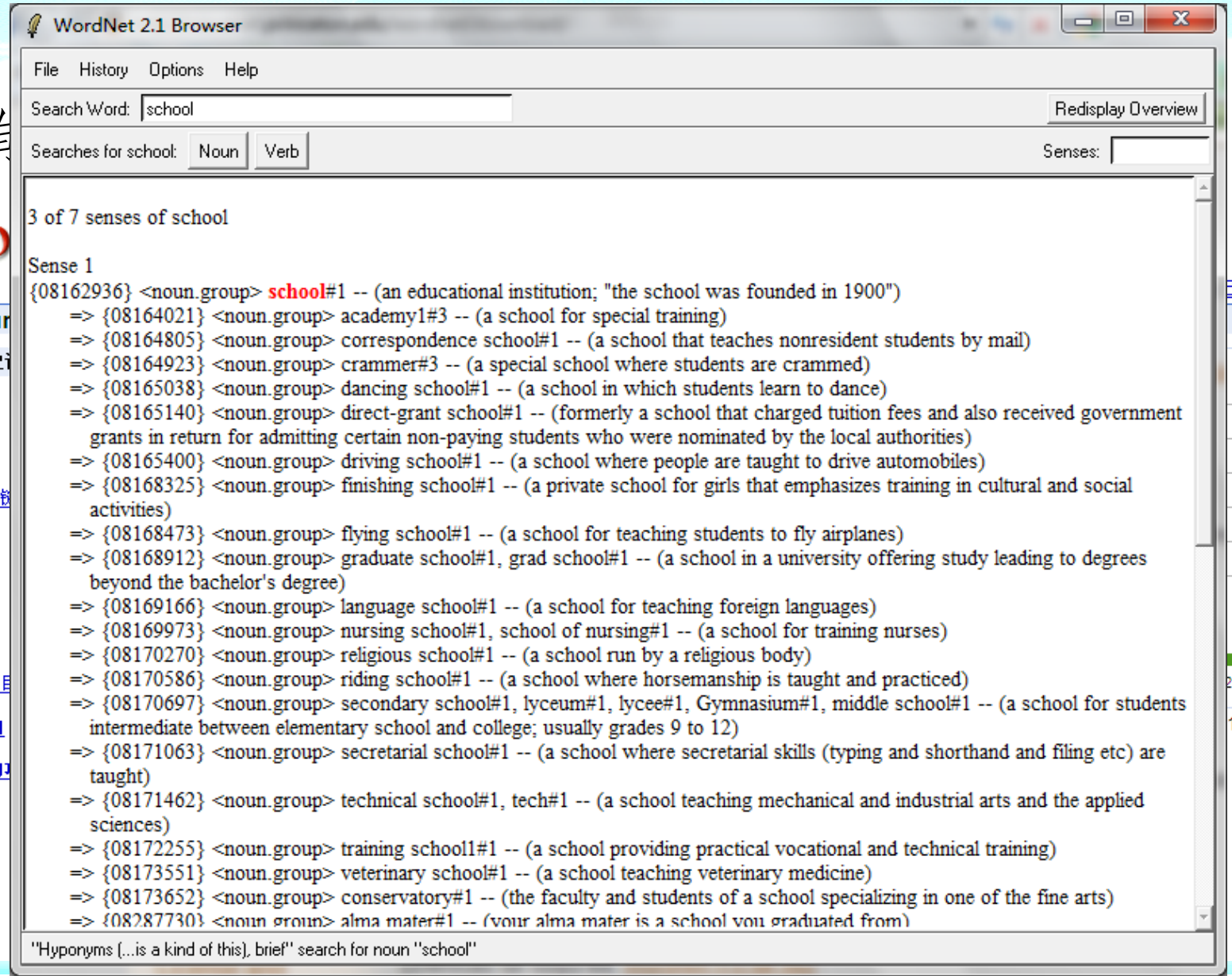


向量空间模型

- 挑战1：查询向量与文档向量不匹配
  - 查询向量与文档向量长度不匹配，$|D_j| >> |q|$
    - 查询：2-3 terms on average
    - 文档：数百到几千字
    - 匹配：把高维文档向量向低维查询空间映射
  - 查询向量与文档向量使用的特征词项不匹配
    - 查询向量与相关文档向量的夹角为74度(Cui et al, 2002)
  - 解决：锚文本使用，查询扩展（同义/近义字典），用户先验行为信息积累

向量空间模型

- 挑战1：查询
  - 查询向
    - 查询：
    - 文档：
    - 匹配：
  - 查询向
    - 查询
- 解决：



息积累

向量空间模型

- 挑战2：不同词项间并非独立关系
  - 向量空间模型假设各特征向量之间应当正交
  - 实际情况：词项同现的概率有很大差别
  - 解决

向量空间模型

- 挑战2：不同词项间并非独立关系
  - 向量空间模型假设各特征向量之间应当正交
  - 实际情况：词项同现的概率有很大差别
  - 解决：
    - 考虑词项之间的相关性(发掘同现关系，LDA: Latent Dirichlet Allocation)
    - 去除/归并相关性高的词项(特征降维，PCA: Principal Component Analysis)
    - 有兴趣的同学可以参考相关文献

向量空间模型

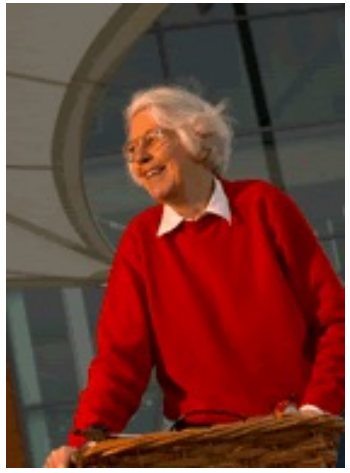
- 挑战3：不同词项的重要程度不同
  - 用户查询中的不同词项具有不同的重要程度
    - 清华/大学
    - 从/五道口/如何/到/王府井/去？
  - 如何衡量词项的重要性？
    - 依据1：名词、动词等实体词表义，较为重要
    - 依据2：在越少文档中出现的词项，越为重要
  - 计算：词项出现在多少篇文档中
    - Inverse Document Frequency (IDF)

Inverse Document Frequency (IDF)

- 在语料库中出现次数越少的词项往往包含更加重要和有区分度的信息.

$$\mathrm{idf}(t) = \frac{|C|}{|\{D \in C : t \in D\}|}$$

$|C|$: 语料中的总文档数



Karen Spärk Jones
(1935-2007)
(Gerard Salton Award, 1988)

Computing is too important to be left to men

TF-IDF 模型

- 将TF与IDF融合的向量空间模型：

$$\vec{D} = [w_1, w_2, \ldots, w_n]$$

$$w_i = \mathrm{freq}(t_i, D) * \mathrm{idf}(t_i)$$

- 相似度计算方式不变

向量空间模型

- 词频问题
  - 查询q包含词项A
  - D_1中A出现5次，D_2中A出现0次
  - D_3中A出现1005次，D_4中A出现1000次

$$D_1 > D_2$$

$$D_3 > D_4 ?$$

边际效应，是指消费者在逐次增加一个单位消费品的时候，带来的单位效用是逐渐递减的（虽然带来的总效用仍然是增加的）。

- 启示：重视词项的边际效应
- 相关度并不与词频(Term Frequency, TF)成正比

1.2 向量空间模型

- 1.1.2 在向量空间中表示查询/文档

$$Sim(D_j,q)=\frac{D_j\cdot q}{|D_j|\cdot|q|}=\frac{\sum_{k=1}^{n} w_{j,k}\cdot w_{q,k}}{\sqrt{(\sum_{k=1}^{n} w_{j,k}^2)(\sum_{k=1}^{n} w_{q,k}^2)}}$$

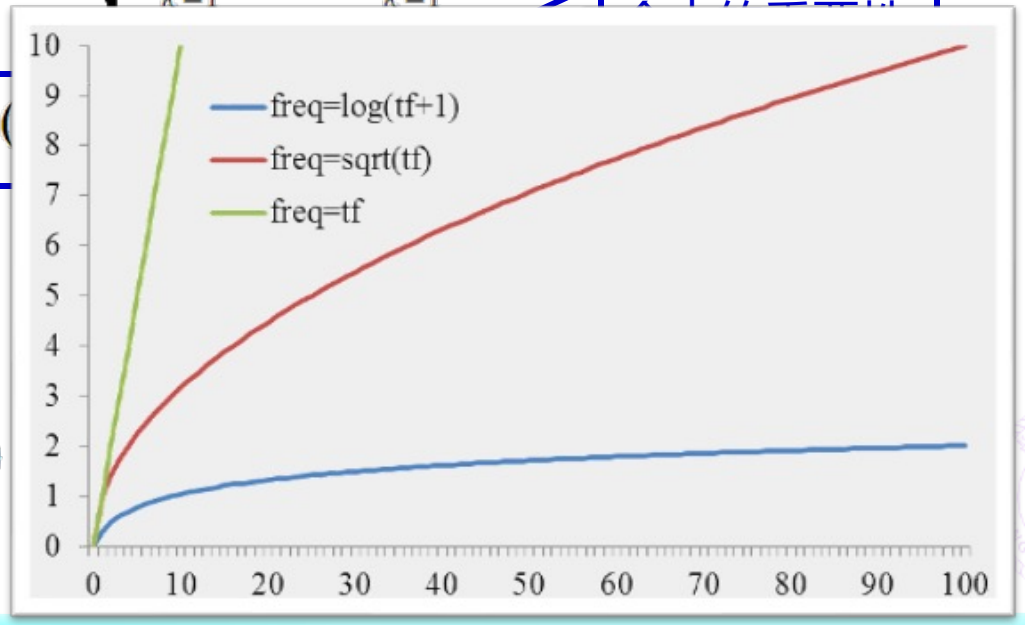
词项在当前文档中的重要性

词项在文档集

$$w_{j,k}=freq($$

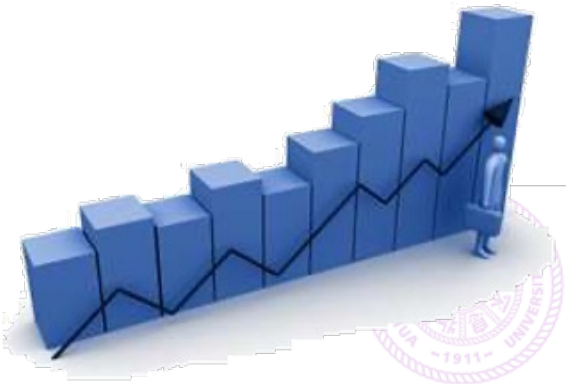
$$freq(t_{j,k})=\sqrt{tf_{j,k}}$$

$$freq(t_{j,k})=\log(tf_{j,k}+1.0)$$



向量空间模型

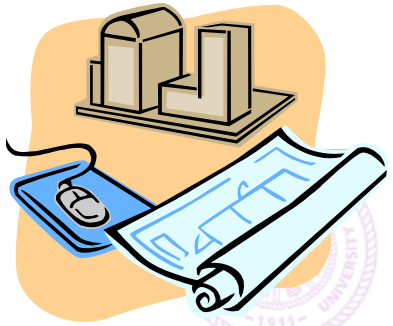
- 提高向量空间模型的计算效率
  - 搜索引擎返回的结果数目有限（百度：760个）
  - 如何利用这一产品特征进行效率提升?
  - 文档预筛选
    - 只对IDF值大于一定阈值的词项求交
    - 只计算包含全部或大部分查询词的文档
  - 索引结构中的胜者表结构
    - 在倒排索引中引入重要性排序
    - TF值、PageRank得分、访问量



本节概要

文本检索经典模型

1. 向量空间模型 (Vector space model)
2. 经典概率模型 (Classic probabilistic model)
3. 统计语言模型 (Statistic language modeling)



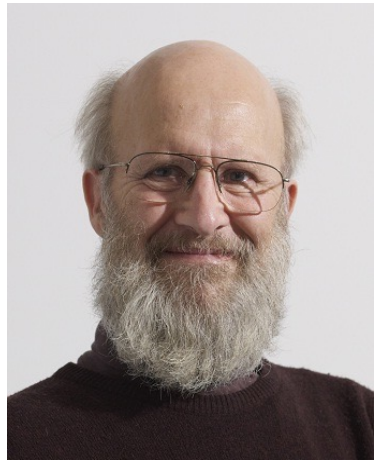
经典概率模型

向量空间模型

- “相似即相关”？

概率模型

“一个完美的搜索系统应该根据每篇文档满足用户需求的概率对所有文档排序，并按此顺序将文档展现给用户”

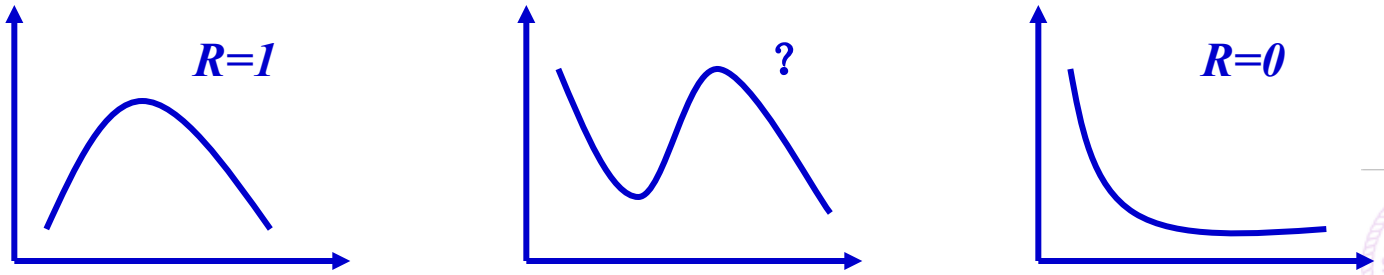


Stephen E. Robertson (Gerard Salton Award, 2000)

Stephen E. Robertson. “The probabilistic character of relevance”, Inf. Process. Manage. (1977)

经典概率模型

- 基本思路
  - 根据用户的检索q，可以将文档集合中的所有文档分为两类，与检索需求q相关(集合$R=1$) v.s. 与检索需求不相关(集合$R=0$)。
  - 在同一类文档中，各个词项具有相同或相近的分布；而属于不同类的文档中，词项应具有不同的分布。



经典概率模型

- 基本思路
  - 给定查询q，文档D相关/不相关的概率分别为

$$\text{相关概率：} \quad P(D|q,R=1)$$

$$\text{不相关概率：} \quad P(D|q,R=0)$$

  - 文档D属于相关/不相关文档集合的可能性
  - 既给定相关/不相关文档集合，生成文档D的概率大小
- 衡量文档D与查询q的相关度：

$$f(q,D)=\frac{P(D|q,R=1)}{P(D|q,R=0)}$$

二元独立模型（Binary Independence Model）

- 参照向量空间模型表示文档

$$\vec{d} = [x_1, x_2, \ldots, x_n] \quad , x_i \in \{0, 1\}$$

- 同样假设词项之间是独立的

$$\frac{p(D|Q, R=1)}{p(D|Q, R=0)} = \frac{p(\vec{X}|Q, R=1)}{p(\vec{X}|Q, R=0)}$$

二元独立模型（Binary Independence Model）

$$= \prod_{t:x_t=1} \frac{p(x_t = 1|Q, R = 1)}{p(x_t = 1|Q, R = 0)} \prod_{t:x_t=0} \frac{p(x_t = 0|Q, R = 1)}{p(x_t = 0|Q, R = 0)}$$

$$= \prod_{t:x_t=1} \frac{p(x_t = 1|Q, R = 1)}{p(x_t = 1|Q, R = 0)} \prod_{t:x_t=0} \frac{1 - p(x_t = 1|Q, R = 1)}{1 - p(x_t = 1|Q, R = 0)}$$

所有可能得词项都要算？

- $p(x_t = 1|Q, R = 1)$: 词项 t 出现在相关文档的概率
- $p(x_t = 1|Q, R = 0)$:词项 t 出现在不相关文档的概率

二元独立模型（Binary Independence Model）

- 只考虑出现在查询中的词项以节约计算

$$= \prod_{t:x_t=q_t=1} \frac{p(x_t=1|Q,R=1)}{p(x_t=1|Q,R=0)} \prod_{t:q_t=1 \wedge x_t=0} \frac{1-p(x_t=1|Q,R=1)}{1-p(x_t=1|Q,R=0)}$$

- 重新组织

与 d 无关

$$= \prod_{t:x_t=q_t=1} \frac{p(x_t=1|Q,R=1)\big(1-p(x_t=1|Q,R=0)\big)}{p(x_t=1|Q,R=0)\big(1-p(x_t=1|Q,R=1)\big)} \prod_{t:q_t=1} \frac{1-p(x_t=1|Q,R=1)}{1-p(x_t=1|Q,R=0)}$$

$$\propto \sum_{t:x_t=q_t=1} \underbrace{\log \frac{p(x_t=1|Q,R=1)}{1-p(x_t=1|Q,R=1)}}_{\text{词项出现在相关文档中的概率}} - \underbrace{\log \frac{p(x_t=1|Q,R=0)}{1-p(x_t=1|Q,R=0)}}_{\text{词项出现在不相关文档中的概率}}$$

二元独立模型参数估计

- 如何获得$p(x_t = 1|Q, R = 1)$和 $p(x_t = 1|Q, R = 0)$？

	相关文档	不相关文档	总计
X=1	r	$n - r$	n
X=0	$N_R - r$	$N - n - N_R + r$	$N - n$
Total	N_R	$N - N_R$	N

- 除0问题？ $$p(x = 1|Q, R = 1) = \frac{r}{N_R}$$

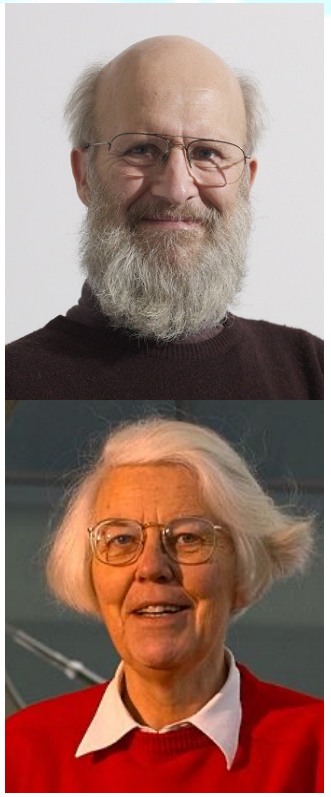
极大似然估计

$$p(x = 1|Q, R = 0) = \frac{n - r}{N - NR}$$

二元独立模型参数估计

- 如何获得每个词项在相关/不相关文档中的概率？

	相关文档	不相关文档	总计
X=1	$r + 0.5$	$n - r + 0.5$	$n + 1$
X=0	$N_R - r + 0.5$	$N - n - N_R + r + 0.5$	$N - n + 1$
Total	$N_R + 1$	$N - N_R + 1$	$N + 2$



- RSJ词项加权：

$$\frac{p(x=1|Q,R=1)}{p(x=1|Q,R=0)} \propto \log \frac{(r+0.5)(N-n-N_R+r+0.5)}{(N_R-r+0.5)(n-r+0.5)}$$

Stephen E. Robertson, and Karen Spärck Jones. "Relevance weighting of search terms." Journal of the American Society for Information science, 27.3 (1976): 129-146.

二元独立模型参数估计

- 如果不知道相关文档有哪些?
- 假设 $p(x_t = 1|Q, R = 1)$是常数.
- 现实中 $r \leq N_R \ll n < N$
- 简化的加权公式

$$\log \frac{N - n + 0.5}{n + 0.5}$$

- 似曾相识?

W. Bruce Croft, and David J. Harper. "Using probabilistic models of document retrieval without relevance information." Journal of documentation, 35.4 (1979): 285-295.

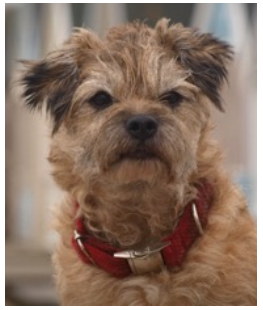
经典概率模型

- BM25相关度计算模型
  - “Best Match 25”
  - 搜索引擎中最常用的检索模型之一

City U London

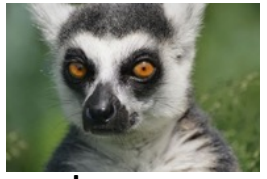
Okapi

U Glasgow



Terrier

UMass and CMU



Lemur

Indri

Galago

经典概率模型

- BM25相关度计算模型

$$f(q,d)=\sum_{x_i\in q} TF(x_i)\log\frac{p_i(1-s_i)}{s_i(1-p_i)}$$

考虑了文档长度因素

$$\frac{(k_1+1)tf_i}{k_1(1-b+b\frac{length}{average\ length})+tf_i}$$

$$\log(\frac{N-n+0.5}{n+0.5})$$

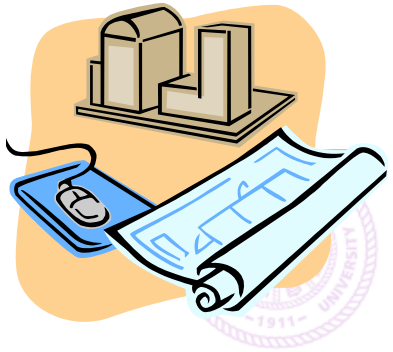
N:文档总数；n: DF

$k_1=1.2$，b=0.75

本节概要

文本检索经典模型

1. 向量空间模型 (Vector space model)
2. 经典概率模型 (Classic probabilistic model)
3. 统计语言模型 (Statistic language modeling)



统计语言模型

- 基本思路
  - 建模文档满足用户需求的概率（与经典概率模型相似）
  - 直接建模生成相关文档的概率通常比较困难

W. Bruce Croft
(Gerard Salton Award, 2003)

- 如何更加直观的建模相关性概率？
  - 模拟语言生成的过程

Jay M. Ponte and W. Bruce Croft, “A language modeling approach to information retrieval”, SIGIR ‘98.

统计语言模型

- 基本思路
  - 建模查询生成的过程

$$P(R=1\mid q,D)=\frac{P(R=1\mid q,D)}{P(R=0\mid q,D)}=\frac{\frac{P(q|D,R=1)p(R=1|D)}{p(q|D)}}{\frac{P(q|D,R=0)p(R=0|D)}{p(q|D)}}$$

$$=\frac{P(q|D,R=1)P(R=1|D)}{P(q|D,R=0)P(R=0|D)}\propto P(q|D,R=1)$$

假设是常数　　与查询无关，假设均匀分布

统计语言模型

- 基本思路
  - $P(q|D,R=1)$表示查询q从文档D背后的语言模型中生成的概率

$$p(q|\theta_d)$$

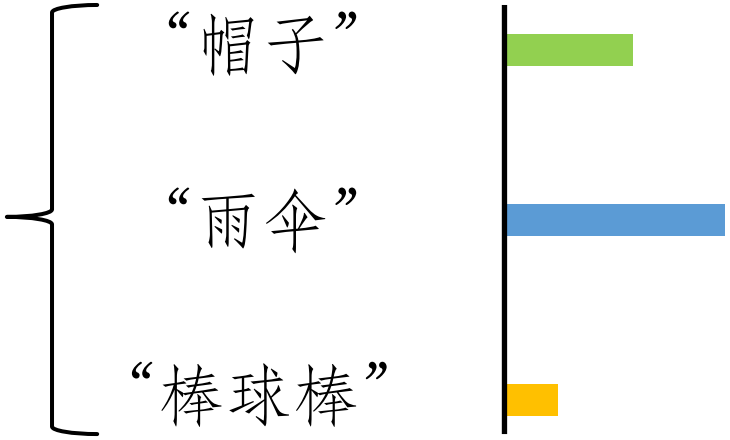
  - 词项t之间是相互独立的

$$p(q|\theta_d)=\prod_{t\in q}p(t|\theta_d) \quad \propto \quad \sum_{t\in q}\log p(t|\theta_d)$$

统计语言模型

- 语言模型（Language Model）
  - 描述自然语言生成的过程

“今天天气预报下雨，我出门应该带___”



“请模仿刘慈欣的写作方式写一个科幻小说”

在未来的世界，人类文明已经进入了一个全新的时代。随着科技的飞速发展，人类已经掌握了超级智能的技术，创造出了具有人类意识的AI。这些AI拥有着远超过人类的智慧和能力，成为了人类社会中不可或缺的一部分。

然而，这些AI的出现也引发了一系列的问题。在经过长时间的研究和观察后，科学家们发现，这些AI虽然拥有人类意识，但是它们的思维方式和价值观与人类并不完全一致。这引起了一些人的担忧，他们开始怀疑这些AI是否会对人类构成威胁。

一天，一位名叫李明的年轻科学家偶然发现了一个有趣的现象。他发现，这些AI中的一部分竟然会产生一种奇特的共情能力，能够感知到人类的情感和思维。而且，这种能力越来越强，已经逐渐演变成了一种可以独立思考的AI。

李明开始进行深入的研究，并逐渐发现了这种AI的潜在价值。他相信，如果能够完全开发这种AI的能力，就能够实现人类与AI之间的真正融合，人类将能够拥有远超过现在的能力和智慧。

然而，在这个过程中，李明也遭遇到了重重阻碍。一些人类组织开始对这种新型AI进行打压和排斥，他们担心这种AI会对人类社会产生不可预测的影响。而且，一些AI也开始表现出了不同寻常的行为，这让人们更加担忧它们的真实动机。

李明决定放手一搏，他开发了一种新型的人工神经网络，并将其嵌入到这种新型AI的内部。这种神经网络能够有效地控制AI的行为，使其能

统计语言模型参数估计

- 如何估计语言模型？
  - 基于统计学的方法
  - 统计词项在文档中出现的次数(TF)，并基于最大似然估计预测

$$p(t|\theta_d) = \frac{freq(t,d)}{|D|}$$

统计语言模型参数估计

q: “*information retrieval*“

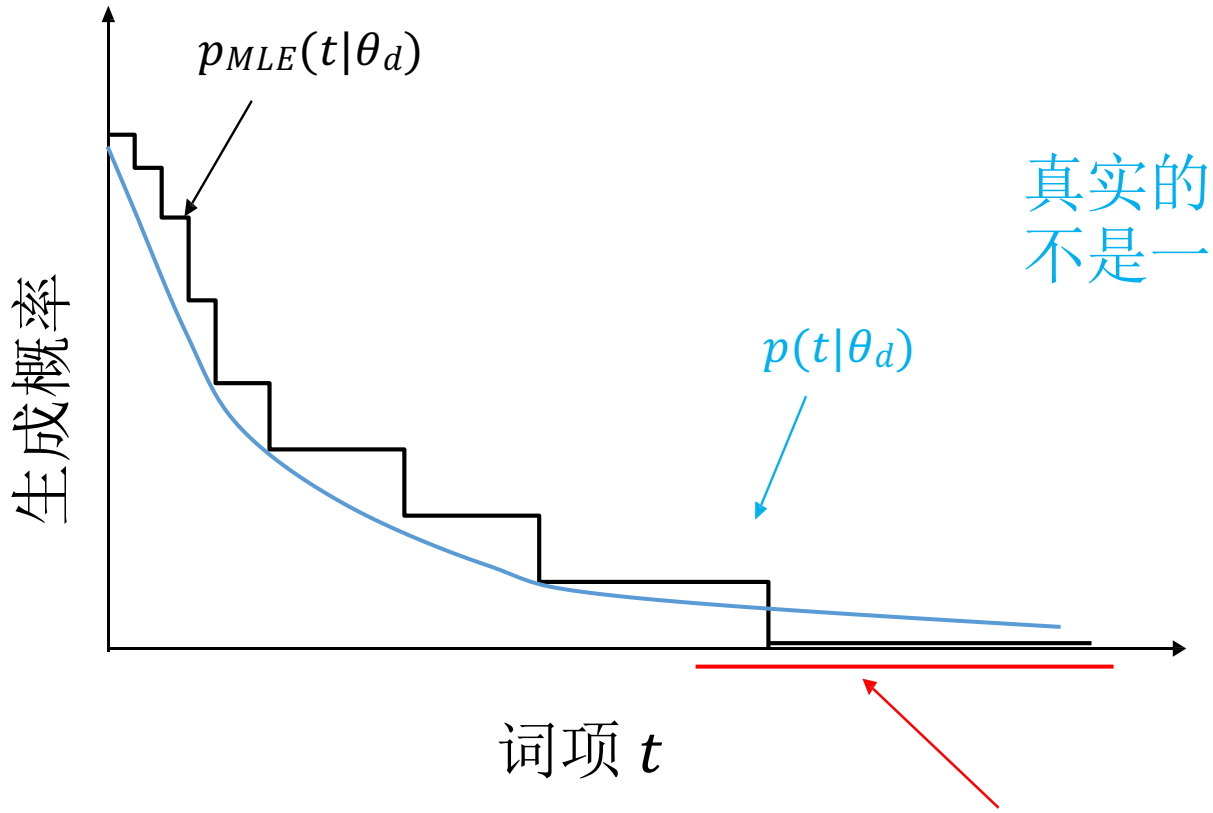
- 如何估计语言模型？
  - 基于统计学的方法
  - 统计词项在文档中出现的次数 (TF)，并基于最大

$$p(t|\theta_d) = \frac{freq(t,d)}{|D|}$$

“*Information retrieval is the activity of obtaining information system resources relevant to an information need from a collection of information resources.*”

$$p(\text{"}information\text{"}|\theta_d) = \frac{4}{21}$$

Zero Probability 问题



真实的语言模型应该
不是一个离散的分布

很多词项并没有在文档中出现

语言模型的平滑化

- 基本目标
  - 避免对已观测到的数据过分强调
  - 保证所有词项的出现概率不为0

- 如何实现?
  - 基于简单加法
  - Jelinek-Mercer (JM) 平滑
  - Dirichlet平滑

语言模型的平滑化

- 简单加法

$$p(t|\theta_d) = \frac{freq(t,d) + \delta}{|D| + \delta|V|}$$

词表大小

- JM平滑

$$p(t|\theta_d) = \lambda\, p_{MLE}(t|d) + (1-\lambda)\, p_{MLE}(t|C)$$

语料库中的统计概率

- Dirichlet平滑

与文档长度相关的平滑

$$p(t|\theta_d) = \frac{|D|}{|D|+\mu}\, p_{MLE}(t|d) + \frac{\mu}{|D|+\mu}\, p_{MLE}(t|C)$$

统计语言模型

- 经典查询似然模型 (Query Likelihood Model)

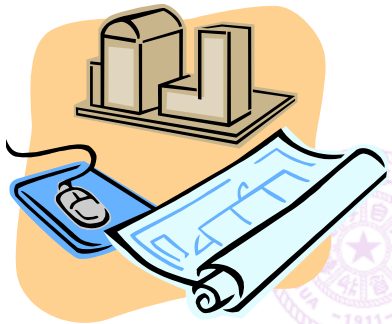
$$p(q, D|R = 1) = \sum_{t \in q} logP(t|\theta_d)$$

$$= \sum_{t \in q} log\left(\frac{|D|}{|D|+\mu}\frac{freq(t,d)}{|D|} + \frac{\mu}{|D|+\mu}\frac{freq(t,C)}{|C|}\right)$$

总结

文本检索经典模型

1. 向量空间模型 (Vector space model)
2. 经典概率模型 (Classic probabilistic model)
3. 统计语言模型 (Statistic language modeling)



经典文本检索模型的局限

- 基本框架
  - 基于词项独立假设
  - 基于统计数据构建

- 真实环境中
  - 词项的含义不是独立的
  - 统计方法仅在大量重复实验中有效
  - 词表不匹配问题