网络信息检索

第三讲　检索模型（2）

董守斌
sbdong@scut.edu.cn

华南理工大学计算机学院
广东省计算机网络重点实验室
Communication & Computer Network Laboratory (CCNL)

复习

- 什么是布尔模型？
  - 文档和查询的表示
  - 相似度计算方法
  - 优缺点
- 什么是向量空间模型？
- 什么是**TF-IDF**权重方法？ **TF-IDF**权重方法能否用在布尔模型中？为什么？

主要内容

- 检索模型的基本概念
- 布尔模型
- 向量空间模型
- 概率模型
- 扩展的检索模型

概率模型原理

- 给定用户查询 q 和文献集中的文献 d_j，概率模型估计用户发现d_j相关的可能性 （**probability**）
- 假设
  - 相关性的概率仅依赖查询和文献的表征**(representations)**
  - 文献集中存在一个子集，用户可以用它来做查询的结果集，即存在一个理想结果集（**ideal answer set**），这个集合仅包含完全相关文献
- 查询处理过程
  - 指定理想答案集性质的过程
- 难题：性质是什么？
  - 用索引项刻画理想结果集的属性
  - 我们并不能确切地知道这些属性，我们所知道的是存在索引词来刻画这些属性

概率模型的实现思路

- 初始估计
  - 由于在查询期间这些属性都是不可见的，这就需要在初始阶段来估计这些属性
  - 这种初始阶段的估计允许我们对首次检索的文档集合返回理想的结果集，并产生一个初步的概率描述
- 相关反馈(relevance feedback)：为了提高理想结果集的描述概率，系统需要与用户进行交互式操作，具体处理过程如下：
  - 用户大致浏览一下结果文档，决定哪些是相关的，哪些是不相关的
  - 然后系统利用该信息重新定义理想结果集的概率描述
  - 重复以上操作，就会越来越接近真正的结果文档集

概率模型过程描述

- 检索出一个初始文献集
- 用户检视这些文献集，找出相关的文献（事实上，只有排在前10~20的文献才会被检视）
- IR系统使用这些用户返回的信息使理想结果集的描述更加精确
- 通过重复这个过程，希望理想结果集的描述得到改善
- 记住，始终猜测理想结果集的描述
- 理想结果集的描述通过概率属性来模拟

参数定义

- q: 查询，索引词的子集
- R: 已知的相关文献集
- $\overline{R}$（R的补集）：不相关的文献集
- $g_i(d_j) \in \{0,1\}$, $g_i(q) \in \{0,1\}$: 索引词权重变量都是二值独立（不相关）的
- $P(R|d_j)$: 文献d_j与查询q相关的概率
- $P(\overline{R}|d_j)$: 文献d_j与查询q不相关的概率

如何定义相似度？

- 给定一个查询，概率模型为每个文档d_j赋予一个关于查询的相似度（**similarity measure**）：

$$\frac{P(d_j\ relevant-to\ q)}{P(d_j\ nonrelevant-to\ q)}=\frac{P(R/d_j)}{P(\overline{R}/d_j)}$$

- $P(R|d_j)$：文献d_j与查询q相关的概率
- $P(\overline{R}|d_j)$：文献d_j与查询q不相关的概率

相似性测度的公式

式2-16

$$sim(d_j,q) \sim \sum_{i=1}^{t} g_i(d_j) g_i(q) \times \left(\log \frac{P(k_i \mid R)}{(1-P(k_i \mid R))}\right) + \log \frac{(1-P(k_i \mid \overline{R}))}{P(k_i \mid \overline{R})}\right)$$

- $P(k_i|R)$：表示索引词k_i在集合R的某篇文档中随机出现的概率
- $P(k_i|\overline{R})$：表示索引词k_i在集合$\overline{R}$的某篇文档中随机出现的概率

相似性测度公式的推导

- ***sim*(d_j,*q*)：** 文档d_j与查询 *q*的相似度

$$sim(d_j,q)=\frac{P(R\mid d_j)}{P(\overline{R}\mid d_j)}$$ **(定义)**

$$sim(d_j,q)=\frac{P(d_j\mid R)\times P(R)}{P(d_j\mid \overline{R})\times P(\overline{R})}$$ **(Bayes定理)** $$P(X\mid Y)=\frac{P(X)P(Y\mid X)}{P(Y)}$$

$$sim(d_j,q)\approx\frac{P(d_j\mid R)}{P(d_j\mid \overline{R})}$$ **(*P*(*R*)和 *P*($\overline{R}$)对所有文档而言都是一样的)**

$P(d_j\mid R)$：从相关文档集合**R**中随机选择文档d_j的概率

***P*(*R*) (*P*($\overline{R}$))：** 从整个文档集中随机选择一个文档作为相关（不相关）文档的概率

$$P(d_j \mid R) \sim \prod_{g_i(d_j)=1} (P(k_i \mid R)) \times \prod_{g_i(d_j)=0} (P(\bar{k}_i \mid R))$$

$$P(d_j \mid \bar{R}) \sim \prod_{g_i(d_j)=1} (P(k_i \mid \bar{R})) \times \prod_{g_i(d_j)=0} (P(\bar{k}_i \mid \bar{R}))$$

假设索引词之间是独立的

- 只考虑在查询中出现的索引词

$$sim(d_j, q) \approx \frac{P(d_j \mid R)}{P(d_j \mid \bar{R})} \sim \frac{\prod_{i=1}^{t} (P(k_i \mid R))^{g_i(d_j) g_i(q)} \times (P(\bar{k}_i \mid R))^{1 - g_i(d_j) g_i(q)}}{\prod_{i=1}^{t} (P(k_i \mid \bar{R}))^{g_i(d_j) g_i(q)} \times (P(\bar{k}_i \mid \bar{R}))^{1 - g_i(d_j) g_i(q)}}$$

- $P(k_i|R)$：表示索引词k_i在集合R的某篇文档中随机出现的概率
- $P(\bar{k}_i|R)$：表示索引词k_i不在集合R的某篇文档中随机出现的概率

$$sim(d_j,q) \sim \log \frac{\prod_{i=1}^{t} (P(k_i \mid R))^{g_i(d_j)g_i(q)} \times (P(\bar{k}_i \mid R))^{1-g_i(d_j)g_i(q)}}{\prod_{i=1}^{t} (P(k_i \mid \bar{R}))^{g_i(d_j)g_i(q)} \times (P(\bar{k}_i \mid \bar{R}))^{1-g_i(d_j)g_i(q)}}$$

$$= \sum_{i=1}^{t} \log \frac{(P(k_i \mid R))^{g_i(d_j)g_i(q)} \times (P(\bar{k}_i \mid R))^{1-g_i(d_j)g_i(q)}}{(P(k_i \mid \bar{R}))^{g_i(d_j)g_i(q)} \times (P(\bar{k}_i \mid \bar{R}))^{1-g_i(d_j)g_i(q)}}$$

$$= \sum_{i=1}^{t} \log \frac{(P(k_i \mid R) \times P(\bar{k}_i \mid \bar{R}))^{g_i(d_j)g_i(q)} \times (P(\bar{k}_i \mid R))}{(P(k_i \mid \bar{R}) \times P(\bar{k}_i \mid R))^{g_i(d_j)g_i(q)} \times (P(\bar{k}_i \mid \bar{R}))}$$

$$= \sum_{i=1}^{t} g_i(d_j)g_i(q) \times \log \frac{P(k_i \mid R) \times P(\bar{k}_i \mid \bar{R})}{P(k_i \mid \bar{R}) \times P(\bar{k}_i \mid R)} + \sum_{i=1}^{t} \log \frac{P(\bar{k}_i \mid R)}{P(\bar{k}_i \mid \bar{R})}$$

$$= \sum_{i=1}^{t} g_i(d_j)g_i(q) \times \log \frac{P(k_i \mid R) \times (1 - P(k_i \mid \bar{R}))}{P(k_i \mid \bar{R}) \times (1 - P(k_i \mid R))} + \sum_{i=1}^{t} \log \frac{P(\bar{k}_i \mid R)}{P(\bar{k}_i \mid \bar{R})}$$

相似性测度公式的推导（2）

式2-16

$$sim(d_j,q) \approx \frac{P(d_j \mid R)}{P(d_j \mid \overline{R})}$$

$$= \sum_{i=1}^{t} g_i(d_j) g_i(q) \times \log \frac{P(k_i \mid R) \times (1 - P(k_i \mid \overline{R}))}{P(k_i \mid \overline{R}) \times (1 - P(k_i \mid R))} + \sum_{i=1}^{t} \log \frac{P(\overline{k}_i \mid R)}{P(\overline{k}_i \mid \overline{R})}$$

$$= \sum_{i=1}^{t} g_i(d_j) g_i(q) \times (\log \frac{P(k_i \mid R)}{(1 - P(k_i \mid R))}) + \log \frac{(1 - P(k_i \mid \overline{R}))}{P(k_i \mid \overline{R})}) + \sum_{i=1}^{t} \log \frac{P(\overline{k}_i \mid R)}{P(\overline{k}_i \mid \overline{R})}$$

$$\approx \sum_{i=1}^{t} g_i(d_j) g_i(q) \times (\log \frac{P(k_i \mid R)}{(1 - P(k_i \mid R))}) + \log \frac{(1 - P(k_i \mid \overline{R}))}{P(k_i \mid \overline{R})})$$

问题：集合R在哪里？

初始猜测

- $P(k_i|R)$ 对所有的k_i，是一个常数：

$$p(k_i \mid R) = 0.5$$

- 索引词在不相关文献中的分布类似于索引词在所有文献集中的分布

$$P(k_i \mid \overline{R}) = \frac{n_i}{N}$$

（假设 $N>>|R|, N\approx|R|$）

初始排序

- V：初始检出的文献子集，采用概率模型排序（排在前r位的文献集）
- V_i：包含索引词k_i的文献集，是V的子集

文档	相关	不相关	
包含索引词 k_i	V_i	n_i-V_i	n_i
不包含索引词 k_i	$V-V_i$	$N-n_i-V+V_i$	$N-n_i$
	V	$N-V$	

初始排序（2）

- 假设$P(k_i|R)$大致由索引词k_i在已经检出的文献集中的分布替代

$$P(k_i \mid R) = \frac{V_i}{V}$$ 式2-18

- 假设$P(k_i|\overline{R})$大致通过认为所有的非检出文献是不相关的来替代

$$P(k_i \mid \overline{R}) = \frac{n_i - V_i}{N - V}$$ 式2-19

当 V=1 ，V_i=0 ?

V 和 V_i 可能太小

- 解决办法 1

$$P(k_i \mid R) = \frac{V_i + 0.5}{V + 1}$$

$$P(k_i \mid \overline{R}) = \frac{n_i - V_i + 0.5}{N - V + 1}$$

式2-21

- 解决办法 2

$$P(k_i \mid R) = \frac{V_i + \frac{n_i}{N}}{V + 1}$$

$$P(k_i \mid \overline{R}) = \frac{n_i - V_i + \frac{n_i}{N}}{N - V + 1}$$

基于概率模型的检索方法

1. 用布尔向量表示文档和查询；
2. 设定概率初值，根据相似度计算公式计算每个文档向量和查询向量的相似度；
3. 按照文档向量和查询向量的相似度，排序输出初始排序结果集；
4. 在初始结果集，用户指定或按缺省约定选择相关文档，形成相关文档集合；
5. 根据初始概率的改进公式或其变形公式，计算初始概率分布
6. 重新计算各文档与查询的相似度，排序输出最终结果集

计算实例

例2-7

- Q: “gold silver truck”

  D1: “Shipment of gold damaged in a fire”

  D2: “Delivery of silver arrived in a silver truck”

  D3: “Shipment of gold arrived in a truck”

- IDF (选择关键词，N=3)
  - a = in = of = 0 = log $^{3/3}$

    arrived = gold = shipment = truck = 0.176 = log $^{3/2}$

    damaged = delivery = fire = silver = 0.477 = log $^{3/1}$
- 选择8 个关键词 (维)
  - arrived(1), damaged(2), delivery(3), fire(4), gold(5), silver(6), shipment(7), truck(8)

计算实例：表示

- 文档表示：
  - d_1={0,1,0,1,1,0,1,0}
  - d_2={1,0,1,0,0,1,0,1}
  - d_3={1,0,0,0,1,0,1,1}
- 查询表示：
  - q={0,0,0,0,1,1,0,1}

维	1	2	3	4	5	6	7	8
n_i	2	1	1	1	2	1	2	2

计算实例：初始猜测

$$p(k_i \mid R) = 0.5$$

$$p(k_i \mid \overline{R}) = \frac{n_i}{N}(N = 3)$$

$$Sim(d_j, q) = \sum_{i=1}^{t} g_i(d_j) \times g_i(\mathrm{q}) \times \log\left(\frac{P(k_i \mid R) \times (1 - P(k_i \mid \overline{R}))}{P(k_i \mid \overline{R}) \times (1 - P(k_i \mid R))}\right)(t = 8)$$

$$Sim(d_1, q) = \log\left(\frac{0.5 \times \frac{1}{3}}{\frac{2}{3} \times 0.5}\right) = \log\left(\frac{1}{2}\right) = -\log 2 = -0.30103$$

$$Sim(d_2, q) = 0$$

$$Sim(d_3, q) = -2 \times \log 2 = -0.60206$$

$$Sim(d_2, q) > Sim(d_1, q) > Sim(d_3, q)$$

计算实例：排序

- 至此，已经检索出文献并排序输出：
  - D_2、D_1、D_3
- 问题：
  - 需要检出多少文献？
  - 如何逼近理想结果集？
- 和用户交互？
  - 相关反馈

计算实例：重算分布

$V = 1 \& N = 3$

- 假设检出一个文档，d_2相关

$$P(k_i \mid R) = \frac{V_i + 0.5}{V + 1}$$

$$P(k_i \mid \overline{R}) = \frac{n_i - V_i + 0.5}{N - V + 1} (N = 3)$$

$$Sim(d_i, q) = \sum_{i=1}^{t} g_i(d_i) \times g_i(q) \times \log\left(\frac{P(k_i \mid R) \times (1 - P(k_i \mid \overline{R}))}{P(k_i \mid \overline{R}) \times (1 - P(k_i \mid R))}\right) (t = 8)$$

维	1	2	3	4	5	6	7	8
V_i	1	0	1	0	0	1	0	1
n_i	2	1	1	1	2	1	2	2

计算实例：最终排序

$$Sim(d_1,q) = \log\left(\frac{\frac{0.5}{2}\times\frac{0.5}{3}}{\frac{2.5}{3}\times\frac{1.5}{2}}\right) = -(\log 5 + \log 3) = -1.17609$$

$$Sim(d_2,q) = 2\times\log 3 + \log 5 = 1.65321$$

$$Sim(d_3,q) = -\log 5 = -0.69897$$

$$Sim(d_2,q) > Sim(d_3,q) > Sim(d_1,q)$$

- 检出2个文献： d_2 (相关) 和 d_1不相关？
- 检出2个文献： d_2 和d_1 (不相关)？

概率模型的分析

- 优点
  - 文献可以以相关概率的降序排列
- 缺点
  - 需要猜测初始相关集和不相关集(索引词的出现与否)
  - 不考虑索引词出现在文献内部的频率
  - 假设索引词之间是独立的

经典模型的比较

- 布尔模型：最弱（ **weakest** ）的经典模型
- 概率模型是否比向量模型好，还存在争议
  - 对一般的文献集，向量模型应该比概率模型的效果好**(Salton and Buckley)**
  - 目前采用的主流检索模型是向量模型

总结：经典模型

- 布尔模型—集合模型
  - 文档和查询被表示成索引词的集合
  - 比较布尔查询语句与代表文档内容的索引词集
- 向量模型—代数模型
  - 文档和查询被分别表示成t维空间的向量
  - 计算查询和文档间的全局相似度
- 概率模型—概率论
  - 在概率论的基础上，文档和查询被分别表示
  - 计算文档集的相关概率（**relevance probabilities**）

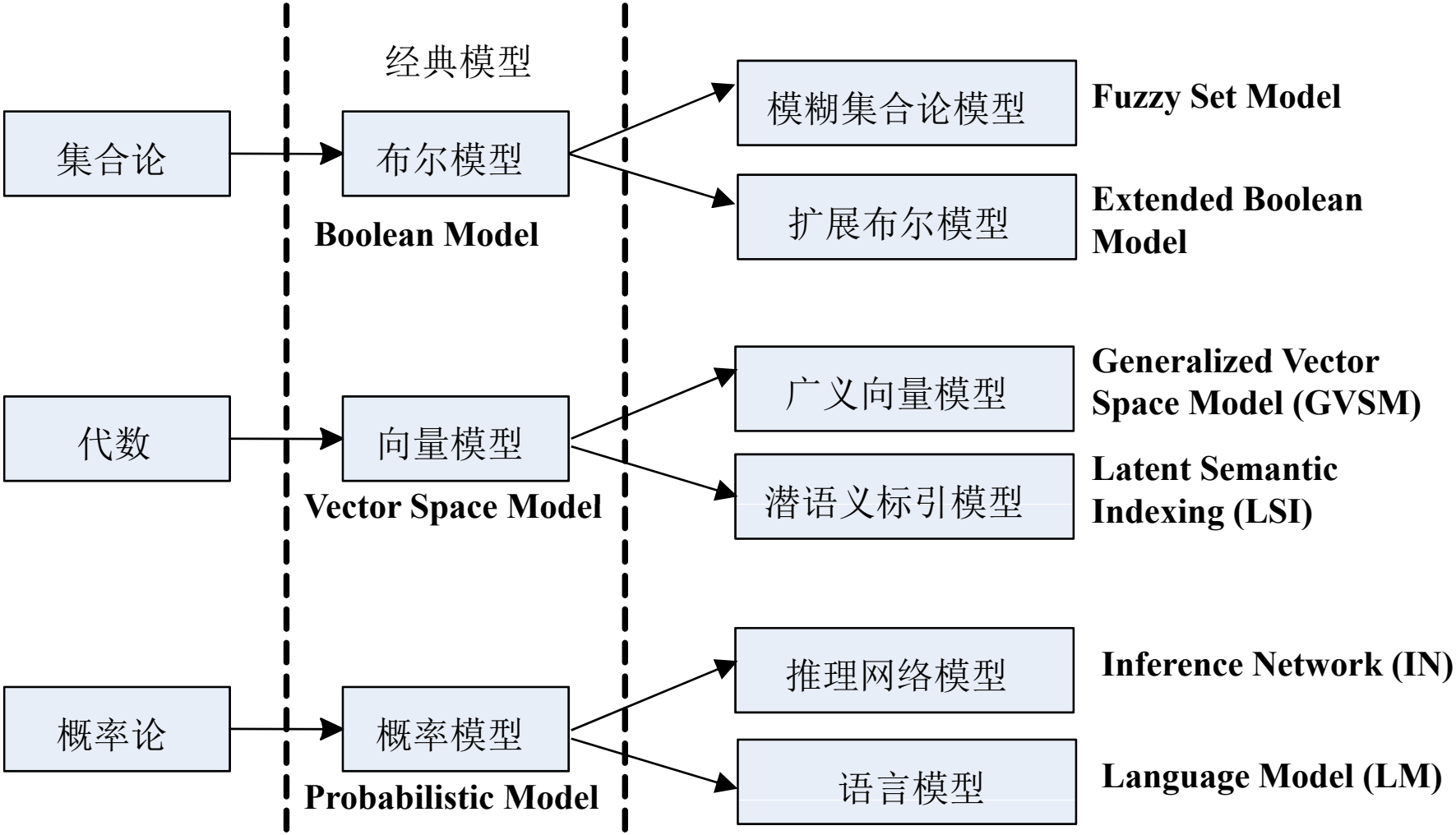
总结：经典模型

理论基础	经典模型	性质	缺点
集合理论	布尔模型	●索引词的集合 ●布尔查询语句	●准确的匹配 ●不能排序
代数理论	向量空间模型	●向量表示 ●通过计算全局相似度排序文档 ●最好/部分 的匹配	●假设词项独立 ●丢失了语义等信息
概率理论	概率模型	● 概率表示 ●通过计算相关概率来排序文档 ●最好/部分 的匹配	●二值权重 ●文档、词项独立 ●丢失了语义等信息

主要内容

- 检索模型的基本概念
- 布尔模型
- 向量空间模型
- 概率模型
- 扩展的检索模型

扩展检索模型


经典模型
集合论
布尔模型
Boolean Model
模糊集合论模型
Fuzzy Set Model
扩展布尔模型
Extended Boolean Model
代数
向量模型
Vector Space Model
广义向量模型
Generalized Vector Space Model (GVSM)
潜语义标引模型
Latent Semantic Indexing (LSI)
概率论
概率模型
Probabilistic Model
推理网络模型
Inference Network (IN)
语言模型
Language Model (LM)

如何改进布尔模型？

- 改进的地方：
  - 改完全匹配为部分匹配
  - 增加结果排序
- 布尔模型的两个扩展
  - 模糊集合模型（**Fuzzy Set Model**）**[Ogawa, Morita, and Kobayashi (1991)]**）
  - 扩展布尔模型（**Extended Boolean Model**）**[Salton,Fox and Wu (1983)]**）

主要内容

- 检索模型的基本概念
- 布尔模型
- 向量空间模型
- 概率模型
- 扩展的检索模型
  - 模糊集合模型
  - 扩展布尔模型
  - 潜语义索引模型
  - 广义向量空间模型
  - 语言模型

模糊集合模型的基本思想

- 引入集合元素的隶属度（**degree of membership**）概念
- 通过定义词－词关联矩阵（**term-term correlation matrix**），对查询词进行扩展，以提取更多的相关的文档
- 通过放松对集合成员的约束条件来得到排序结果

模糊集合论

- 将权值表示二值化改为：
  - $w_{i,j} \in \{0,1\} \rightarrow w_{i,j} \in [0,1]$
- 隶属度
  - 采用模糊集合论的隶属度表示权值
  - 1表示完全属于某个集合
  - 0表示完全不属于某个集合
- 例子：明天下雨吗？

$$\mu_{\overline{A}}(u) = 1 - \mu_A(u)$$

$$\mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u))$$

$$\mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u))$$

模糊信息检索：查询词扩展

- 把系统中的所有索引项定义成一个词－词关联矩阵（**term-term correlation matrix**）C，该矩阵的行和列分别对应文档集中的索引项，矩阵C的每个元素$c_{i,l}$，叫做索引项k_i和k_l的标准化关联因子

$$c_{i,l} = \frac{n_{i,l}}{n_i + n_l - n_{i,l}} \qquad (2\text{-}25)$$

  - n_i：包含k_i的文档个数
  - n_l：包含k_l的文档个数
  - $n_{i,l}$：同时包含k_i和k_l的文档个数

模糊信息检索：隶属度

- 每个索引项k_i都存在一个相关联的模糊集合，在这个集合里，文档d_j的隶属度定义如下：

$$\mu_{i,j} = 1 - \prod_{k_l \in d_j} \left(1 - c_{i,l}\right) \qquad (2\text{-}26)$$

- 以上的表达式对d_j所包含的所有索引项计算一个代数和
  - 如果文档d_j包含的语词和k_i有关，则该文档属于k_i的模糊集合
  - 文档d_j中至少有一个索引词k_l与索引词k_i密切相关，如$c_{i,l} \approx 1$，则：$\mu_{i,j} \approx 1$

模糊信息检索：相似度测度

- 相似度测度定义为：

$$\mu_{q,j} = 1 - \prod_{i=1}^{M} \left(1 - \mu_{cci,j}\right) \qquad (2\text{-}27)$$

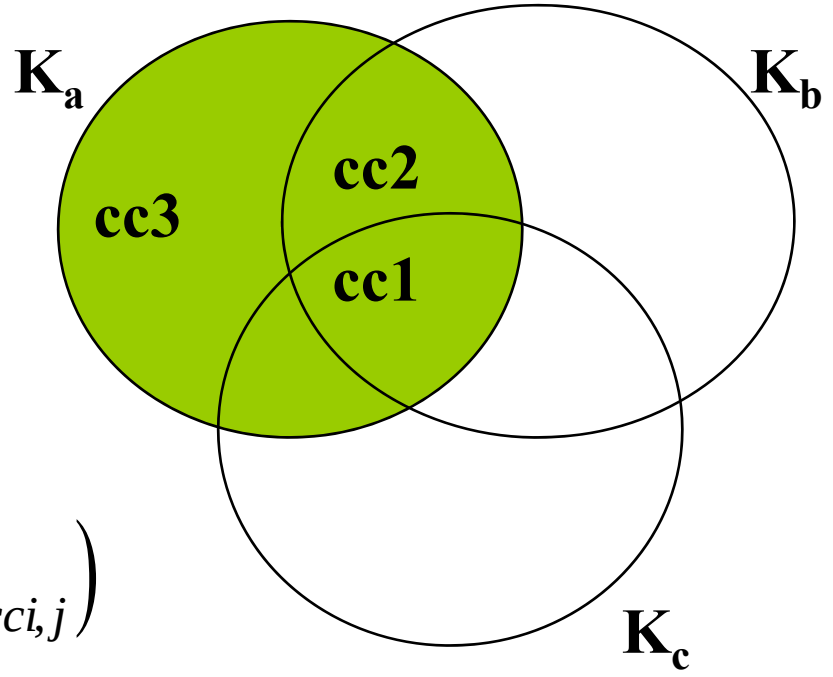
- M是查询析取范式的合取分量个数，$\mu_{cci,j}$是合取分量集合中文档d_j的隶属度，它可以用查询词相关联的文档d_j的隶属度来表示

$$q_{dnf} = cc_1 \vee cc_2 \vee \ldots \vee cc_M$$

例子

- $q = k_a \wedge (k_b \vee \neg k_c)$

  $= (1,1,1) \vee (1,1,0) \vee (1,0,0)$



$$\mu_{q,j} = \mu_{cc_1+cc_2+cc_3,j} = 1 - \prod_{i=1}^{M} (1 - \mu_{cci,j})$$

$$= 1 - (1 - \mu_{a,j}\mu_{b,j}\mu_{c,j}) \times (1 - \mu_{a,j}\mu_{b,j}(1 - \mu_{c,j}))$$

$$\times (1 - \mu_{a,j}(1 - \mu_{b,j})(1 - \mu_{c,j}))$$

图2-8

主要内容

- 检索模型的基本概念
- 布尔模型
- 向量空间模型
- 概率模型
- 扩展的检索模型
  - 模糊集合模型
  - 扩展布尔模型
  - 潜语义索引模型
  - 广义向量空间模型
  - 语言模型

扩展布尔模型

- 用部分匹配和权重计算扩展布尔模型
- 将矢量模型的特性与布尔代数结合

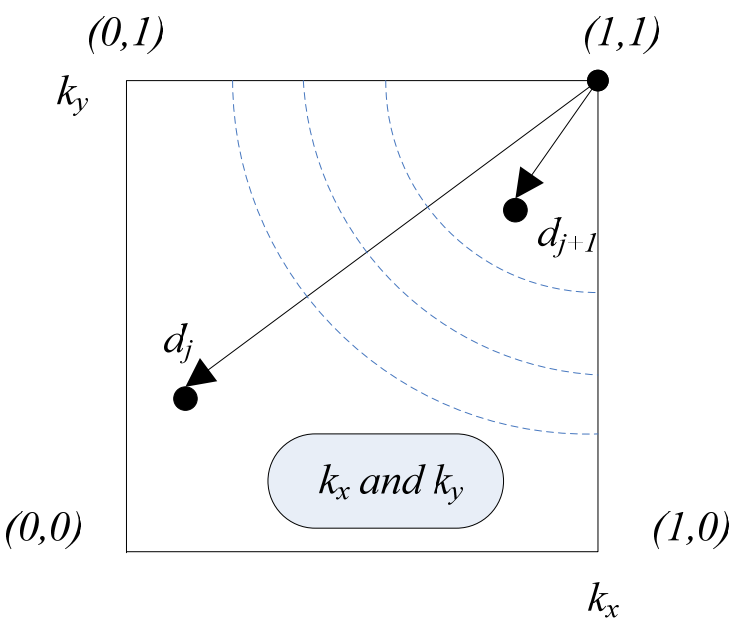
扩展布尔模型的基本思想

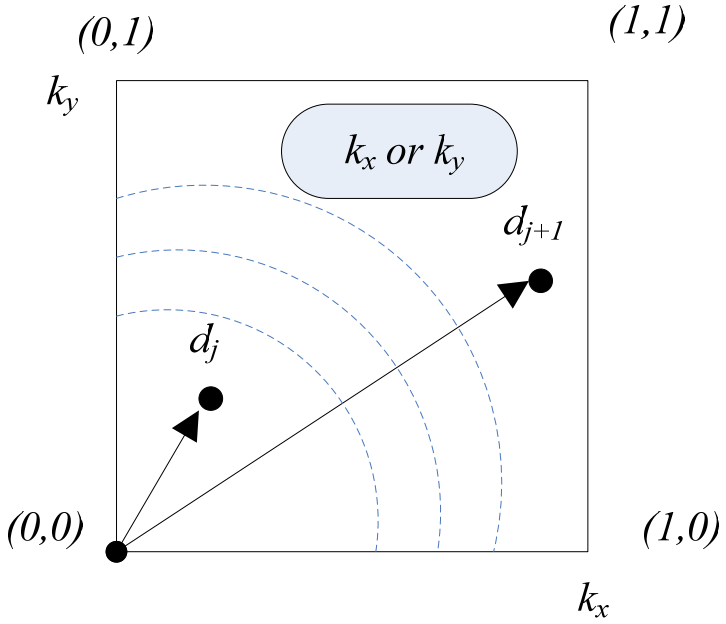
- 考虑一个简单的例子（只有两个索引词）

$$q_{or} = k_x \vee k_y \qquad q_{and} = k_x \wedge k_y$$

- 对于上述合取查询，只含有一个索引词的文档和两个索引词都不包含的文档一样不能被检索到
- 怎样放松条件，检索到部分匹配的词？

二维图例





We want a document to be as close as possible to (1,1)

We want a document to be as far as possible from (0,0)

$$sim(q_{and}, d) = 1 - \sqrt{\frac{(1-x)^2 + (1-y)^2}{2}}$$

$$sim(q_{or}, d) = \sqrt{\frac{x^2 + y^2}{2}}$$

图2-9

查询-文档相似度比较

传统的布尔检索

文档	索引词		相似度	
	K_x	K_y	K_x OR K_y	K_x AND K_y
D_1	1	1	1	1
D_2	1	0	1	0
D_3	0	1	1	0
D_4	0	0	0	0

扩展的布尔检索

文档	索引词		相似度	
	K_x	K_y	K_x OR K_y	K_x AND K_y
D_1	1	1	1	1
D_2	1	0	$1/\sqrt{2}$	$1-1/\sqrt{2}$
D_3	0	1	$1/\sqrt{2}$	$1-1/\sqrt{2}$
D_4	0	0	0	0

讨论

- 如何在扩展布尔模型中集成权重计算方法？
- 各个模型的优缺点？
- 模糊集合模型
  - 优点：克服原始布尔模型不能部分匹配的缺点
  - 缺点：通常在模糊集研究领域涉及，在IR领域不流行，缺乏大规模语料上的实验证实其有效性
- 扩展布尔模型
  - 优点：提供了一个统一的框架，将向量空间模型、基于模糊集的模型都包括在一个框架内
  - 缺点：不够自然简洁，目前在IR领域使用较少

超越布尔方式

- 对所有用户都很自然的查询方式
  - 句子，短语，词语
  - 没有**AND，OR，NOT**之类的东西
  - 没有括弧（没有结构）
- 系统能够注意到重要的词
  - **Q：**我现在想了解激光打印机的问题
- 发现我所想的，不仅是我所说的（**find what I mean, not just what I say**）
  - **Q：** ***cheap car insurance***

```
(pAND (pOR "cheap" [1.0] "inexpensive" [0.9]
           "discount" [0.5])]
      (pOR "car" [1.0] "auto" [0.8]
           "automobile" [0.9] "vehicle" [0.5])
      (pOR "insurance" [1.0] "policy" [0.3]))
```

向量空间模型的扩展

- **VSM中将每篇文档看成多个Term为坐标轴的空间上的点**
- **VSM假设多个Term之间的出现是互相独立的，这与实际情况显然不符**

- 潜语义模型/隐性语义索引
- 广义向量空间模型

主要内容

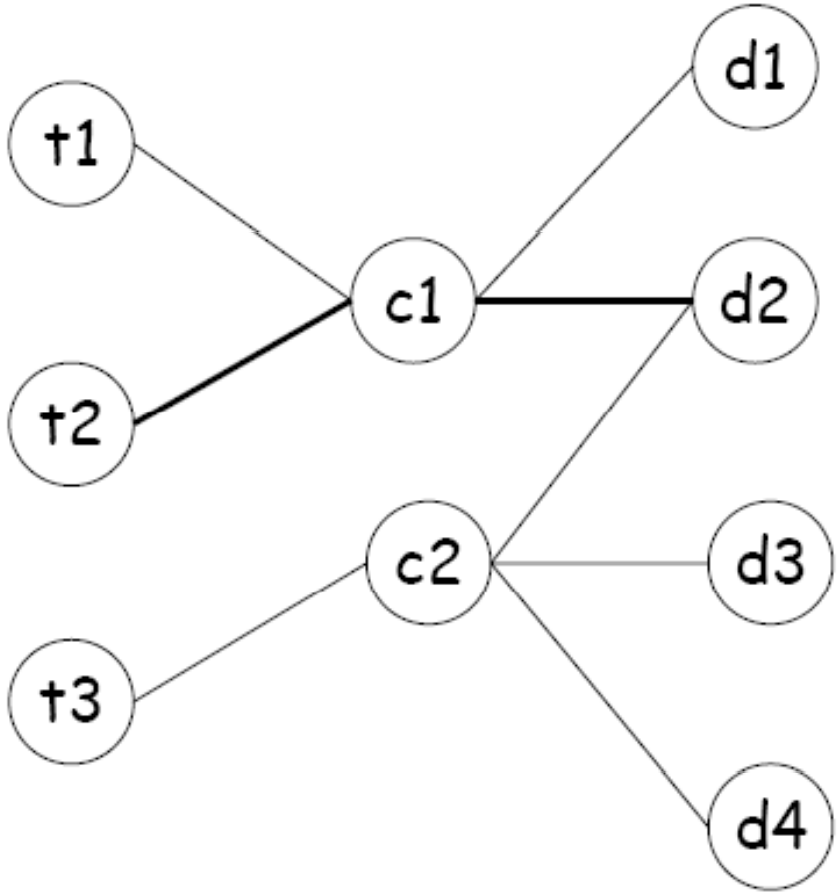
- 检索模型的基本概念
- 布尔模型
- 向量空间模型
- 概率模型
- 扩展的检索模型
  - 模糊集合模型
  - 扩展布尔模型
  - 潜语义索引模型
  - 广义向量空间模型
  - 语言模型

存在的问题

- 自然语言文本中的词汇（术语）具有一词多义（**polysemy**）和一义多词（**synonymy**）的特点
- 由于一词多义， 基于精确匹配的检索算法会报告许多用户不要的东西
  - 如：处理
    - 什么地方处理旧家具？
    - 你去把那个叛徒处理了
    - 处理自然语言很难
- 由于一义多词，基于精确匹配的检索算法又会遗漏许多用户想要的东西
  - 如：“互联网”，“万维网”，“因特网”，“国际互联网”等
- 导致:
  - 很多无关文献可能出现在结果集中
  - 没有出现关键词的相关文献可能无法被检索

潜语义索引模型
（Latent Semantic Indexing，LSI)

- 以概念匹配代替词项匹配（**Furnas**等在**1988**提出）
  - 每维代表一个基本的概念（**concept**）
  - 文档和查询被映射到概念空间



概念空间如何得到？

词项—文档矩阵
（Term-Document Matrix，TDM）

- 文档集，包含***n***个文档，用到了***t***个词汇，对矩阵的每一个元素，可以为其分配一个权值，表示词汇k_i在文档d_j中的权重。

$$M_{t\times n}=[m_{i,j}]=\begin{bmatrix} & d_1 & d_2 & ....d_n \\ k_1 & w_{1,1} & w_{1,2} & w_{1,n} \\ k_2 & w_{2,1} & w_{2,2} & w_{2,m} \\ . & & & \\ k_t & w_{t,1} & w_{t,2} & w_{t,n} \end{bmatrix}$$

- 由于任意一个文档总是由有限个词汇，而不是由所有***t***个词汇构成，所以***M***必是一个稀疏矩阵

LSI的计算方法

- 利用奇异值分解（Single Value Decomposition，SVD）

$$M = KSD^T$$

- 如果仅保留最大的s个奇异值，得到新的矩阵

$$M_s = K_s S_s D_s^T$$

- 将文档和查询向量映射到与概念相关联的维数较低的空间

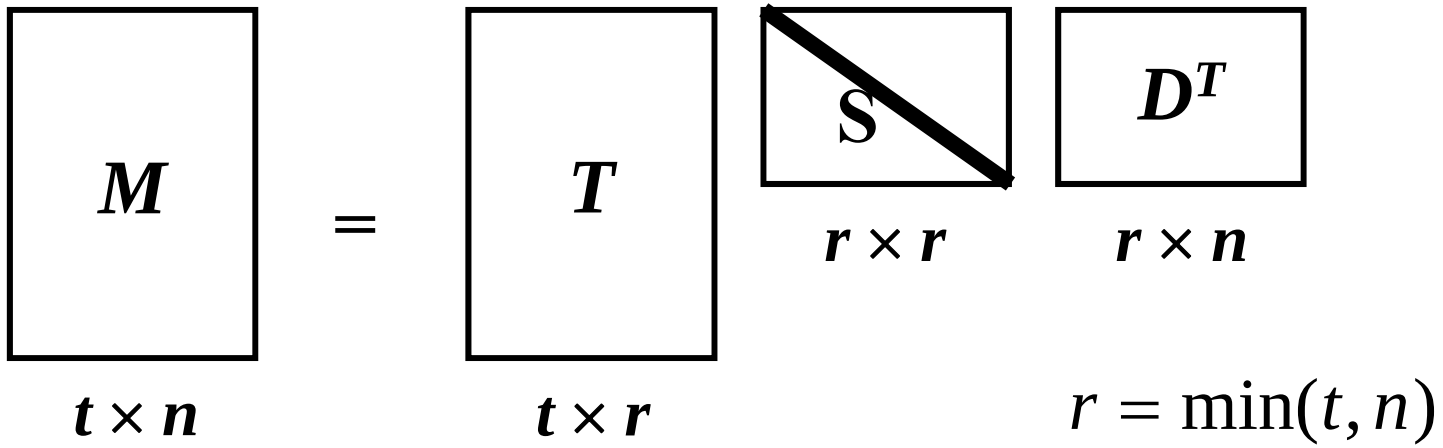
$$q^* = q^T K_s S_s^{-1}$$

- 用标准的余弦相似度计算相似度:

$$sim(q^*, d_i) = \frac{q^* \bullet (D_s^T)_i}{\left| q^* \right\| (D_s^T)_i \right|} \qquad (2\text{-}41)$$

- LSI的思想：在维数降低了的空间中的检索可能优于在索引项集合中的检索（通过SVD分解等处理，去掉噪音，消除了同义词、多义词的影响，提高了后续处理的精度）

SVD分解



LSI例子：文档

B1	A Course on Integral Equations
B2	Attractors for Semigroups and Evolution Equations
B3	Automatic Differentiation of Algorithms: Theory, Implementation, and Application
B4	Geometrical Aspects of Partial Differential Equations
B5	Ideals, Varieties, and Algorithms – An Introduction to Computational Algebraic Geometry and Commutative Algebra
B6	Introduction to Hamiltonian Dynamical Systems and the N-Body Problem
B7	Knapsack Problems: Algorithms and Computer Implementations
B8	Methods of Solving Singular Systems of Ordinary Differential Equations
B9	Nonlinear Systems
B10	Ordinary Differential Equations
B11	Oscillation Theory for Neutral Differential Equations with Delay
B12	Oscillation Theory of Delay Differential Equations
B13	Pseudodifferential Operators and Nonlinear Partial Differential Equations
B14	Sinc Methods for Quadrature and Differential Equations
B15	Stability of Stochastic Differential Equations with Respect to Semi-Martingales
B16	The Boundary Integral Approach to Static and Dynamic Contact Problems
B17	The Double Mellin-Barnes Type Integrals and Their Applications to Convolution Theory

LSI例子：词项-文档矩阵

表2-6

Terms	Documents																
	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	B14	B15	B16	B17
algorithms	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
application	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
delay	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
differential	0	0	0	1	0	0	0	1	0	1	1	1	1	1	1	0	0
equations	1	1	0	1	0	0	0	1	0	1	1	1	1	1	1	0	0
implementation	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
integral	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
introduction	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
methods	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
nonlinear	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
ordinary	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
oscillation	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
partial	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
problem	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0
systems	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
theory	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1

LSI例子：SVD分解后矩阵

K_s

-0.0154227	-0.422647
-0.0242622	-0.382996
-0.178994	-0.196573
-0.603612	0.0992276
-0.668904	0.135237
-0.0143585	-0.354329
-0.0123052	-0.174656
-0.0063823	-0.0905044
-0.150975	0.119194
-0.0816699	0.0728611
-0.150975	0.119194
-0.178994	-0.196573
-0.142064	0.0983069
-0.00711902	-0.144923
-0.0954645	0.0687813
-0.203256	-0.579569

S_s

4.52655	0
0	2.74066

S=2

D_s^t

-0.147774	0.0493447
-0.147774	0.0493447
-0.0568424	-0.634717
-0.312508	0.12142
-0.00481714	-0.187237
-0.0240726	-0.0608051
-0.00015196	-0.336379
-0.36892	0.197629
-0.0391324	0.0516819
-0.314476	0.129042
-0.405113	-0.26937
-0.405113	-0.26937
-0.33055	0.148005
-0.314476	0.129042
-0.281123	0.0855504
-0.00271846	-0.0637277
-0.0529817	-0.414944

LSI例子：将查询矢量映射到文档空间

- 对于查询“**application theory**”

$$\begin{pmatrix} -0.076442 & -0.605047 \end{pmatrix} = \begin{pmatrix} 0 \\ 2.14007 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1.44692 \end{pmatrix} \begin{pmatrix} -0.0154227 & -0.422647 \\ -0.0242622 & -0.382996 \\ -0.178994 & -0.196573 \\ -0.603612 & 0.0992276 \\ -0.668904 & 0.135237 \\ -0.0143585 & -0.354329 \\ -0.0123052 & -0.174656 \\ -0.0063823 & -0.0905044 \\ -0.150975 & 0.119194 \\ -0.0816699 & 0.0728611 \\ -0.150975 & 0.119194 \\ -0.178994 & -0.196573 \\ -0.142064 & 0.0983069 \\ -0.00711902 & -0.144923 \\ -0.0954645 & 0.0687813 \\ -0.203256 & -0.579569 \end{pmatrix} \begin{pmatrix} 0.220919 & 0. \\ 0. & 0.364876 \end{pmatrix}$$

LSI例子：排序结果

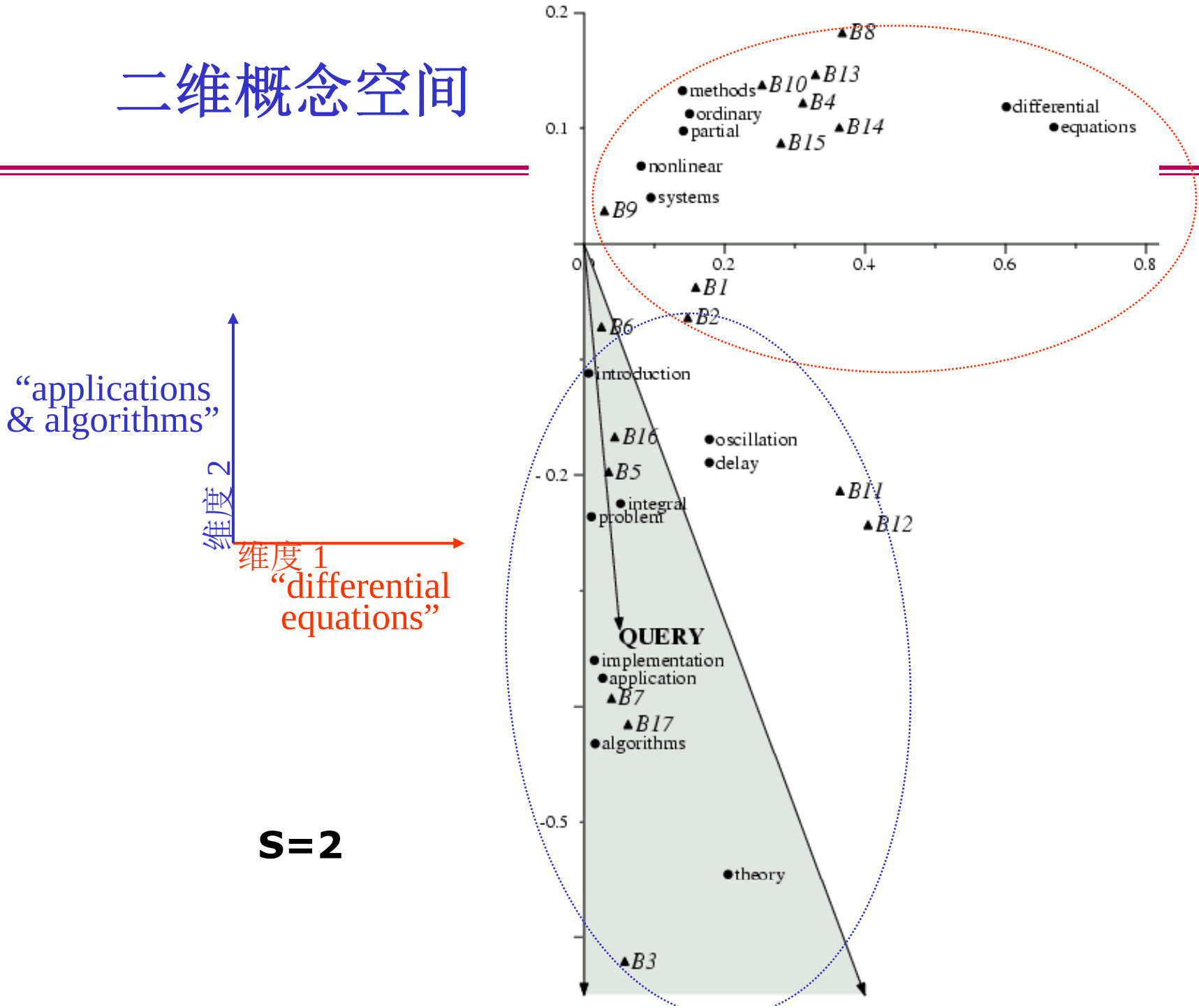
s=2

0.999999	{17}
0.999339	{3}
0.996554	{16}
0.995009	{5}
0.994859	{7}
0.968592	{6}
0.653706	{12}
0.653706	{11}
-0.168923	{15}
-0.19534	{1}

s=4

0.992173	{17}
0.970698	{16}
0.837632	{3}
0.537269	{12}
0.537269	{11}
0.434723	{7}
0.348928	{5}
-0.101838	{6}
-0.125203	{15}
-0.131291	{4}

二维概念空间


0.2
0.1
- 0.2
-0.5
0.2
0.4
0.6
0.8
B8
B10
B13
B4
B14
B15
B1
B2
B6
B16
B5
B11
B12
B7
B17
B3
methods
ordinary
partial
nonlinear
systems
differential
equations
introduction
oscillation
delay
integral
problem
QUERY
implementation
application
algorithms
theory
维度 2
“applications & algorithms”
维度 1
“differential equations”

S=2

关于LSI的讨论

- 优点
  - 潜在语义索引提供了关于IR问题的概念化描述，很有意义
  - 降低了底层表示框架的复杂性
- 缺点：
  - 计算开销较大，SVD非常耗时，目前还没有特别快的方法
    - K（新的维度）通常取经验值（200~1000）
  - 在一些小规模数据集上面效果不错，但是在大规模数据集（如TREC）上效果一般
- 可以应用到分类、过滤、跨语言检索等多个领域

主要内容

- 检索模型的基本概念
- 布尔模型
- 向量空间模型
- 概率模型
- 扩展的检索模型
  - 模糊集合模型
  - 扩展布尔模型
  - 潜语义索引模型
  - 广义向量空间模型
  - 语言模型

广义向量空间模型
（Generalized VSM, GVSM）

- **VSM的问题：**
  - 假设索引向量间两两正交
  - 丢失了索引项之间的关系
- 解决方法
  - 广义向量空间模型**GVSM**放宽了对索引项向量的限制。在广义向量空间模型中，两个索引项向量可能不是正交的，这就意味着索引项向量是由更小的分量所组成

[S.Wong, W.Ziarko and P.Wong (1985)]

最小项

- 设 $\{\vec{k}_1, \vec{k}_2, ..., \vec{k}_t\}$是文档集合的索引项向量，定义词项－文档矩阵（**Term Document Matrix**）。索引项在文档内同时出现（**co-occurrence**）的所有可能模式可以用一个有2^t个元素的最小项（**minterm**）集合来表示

$$\vec{m}_1 = (1,0,...,0,0)$$

$$\vec{m}_2 = (0,1,...,0,0)$$

$$...$$

$$\vec{m}_{2^t} = (0,0,...,0,1)$$

索引项的向量

- 索引项k_i的向量

$$k_i = \frac{\sum_{\forall r, g_i(m_r)=1} c_{i,r}\,\mathbf{m}_r}{\sqrt{\sum_{\forall r, g_i(m_r)=1} c_{i,r}^2}} \qquad (2\text{-}36)$$

$$c_{i,r} = \sum_{d_j \,|\, g_l(\mathbf{d}_j)=g_l(m_r)\ for\ all\ l} w_{i,j} \qquad (2\text{-}35)$$

- 文档d_j和用户查询q

$$\boldsymbol{d}_j = \sum_{\forall i} w_{i,j}\boldsymbol{k}_i \qquad \boldsymbol{q} = \sum_{\forall i} w_{i,q}\boldsymbol{k}_i$$

GVSM例子

- 有如下的索引项-文档矩阵，查询 $\vec{q} = \vec{k_1} + \vec{k_2}$

$$W = \begin{bmatrix} & k_1 & k_2 & k_3 \\ d_1 & 2 & 0 & 1 \\ d_2 & 1 & 0 & 0 \\ d_3 & 0 & 1 & 3 \\ d_4 & 2 & 0 & 0 \end{bmatrix}$$

- 最小项:

$$m_1 = k_1\bar{k}_2k_3, m_2 = k_1\bar{k}_2\bar{k}_3, m_3 = \bar{k}_1k_2k_3, m_4 = k_1k_2k_3$$
$$m_5 = k_1k_2\bar{k}_3, m_6 = \bar{k}_1\bar{k}_2k_3, m_7 = \bar{k}_1k_2\bar{k}_3, m_8 = \bar{k}_1\bar{k}_2\bar{k}_3$$

GVSM例子：最小项

- 这些最小项可以用以下正交的基向量表示：

$\overline{m}_1=\{1,0,0,0,0,0,0,0\}, \overline{m}_2=\{0,1,0,0,0,0,0,0\}$ $\overline{m}_3=\{0,0,1,0,0,0,0,0\}, \overline{m}_4=\{0,0,0,1,0,0,0,0\}$

$\overline{m}_5=\{0,0,0,0,1,0,0,0\}, \overline{m}_6=\{0,0,0,0,0,1,0,0\}$ $\overline{m}_7=\{0,0,0,0,0,0,1,0\}, \overline{m}_8=\{0,0,0,0,0,0,0,1\}$

$$
\begin{aligned}
k_1 &= k_1 \wedge (k_2 \vee \overline{k_2}) \wedge (k_3 \vee \overline{k_3}) \\
&= [(k_1 \wedge k_2) \vee (k_1 \wedge \overline{k_2}) \wedge (k_3 \vee \overline{k_3})] \\
&= (k_1 \wedge k_2 \wedge k_3) \vee (k_1 \wedge \overline{k_2} \wedge k_3) \vee (k_1 \wedge k_2 \wedge \overline{k_3}) \vee (k_1 \wedge \overline{k_2} \wedge \overline{k_3}) \\
&= m_4 \vee m_1 \vee m_5 \vee m_2 \\
k_2 &= k_2 \wedge (k_1 \vee \overline{k_1}) \wedge (k_3 \vee \overline{k_3}) \\
&= m_4 \vee m_3 \vee m_5 \vee m_7 \\
k_3 &= k_3 \wedge (k_1 \vee \overline{k_1}) \wedge (k_1 \vee \overline{k_1}) \\
&= m_4 \vee m_1 \vee m_3 \vee m_6
\end{aligned}
$$

GVSM例子：索引项的表示

- 索引项的表示:

$$\vec{k}_1 = \frac{2\vec{m}_1 + (1+2)\vec{m}_2 + 0\vec{m}_4 + 0\vec{m}_5}{[2^2 + 3^2]^{1/2}} = 0.55\vec{m}_1 + 0.83\vec{m}_2$$

$$\vec{k}_2 = \frac{\boldsymbol{1}\vec{m}_3 + \boldsymbol{0}\vec{m}_4 + \boldsymbol{0}\vec{m}_5 + \boldsymbol{0}\vec{m}_7}{[\boldsymbol{1}^2]^{1/2}} = \vec{m}_3$$

$$\vec{k}_3 = \frac{1\vec{m}_1 + 3\vec{m}_3 + 0\vec{m}_4 + 0\vec{m}_6}{[1^2 + 3^2]^{1/2}} = 0.32\vec{m}_1 + 0.95\vec{m}_3.$$

$$c_{1,1} = w_{1,1} = \boldsymbol{2},\ c_{1,2} = w_{1,2} + w_{1,4} = \boldsymbol{1+2} = \boldsymbol{3},\ c_{1,1} = w_{1,1} = \boldsymbol{2},\ c_{1,4} = c_{1,5} = \boldsymbol{0}$$

$$c_{2,3} = w_{2,3} = \boldsymbol{1},\ c_{2,4} = c_{2,5} = c_{2,7} = \boldsymbol{0}$$

$$c_{3,1} = w_{3,1} = \boldsymbol{1},\ c_{3,3} = w_{3,3} = \boldsymbol{3},\ c_{3,4} = c_{3,6} = \boldsymbol{0}$$

GVSM例子：文档和查询的表示

$$\vec{d_1} = 2\vec{k_1} + \vec{k_3} = 2(0.55\vec{m_1} + 0.83\vec{m_2}) + (0.32\vec{m_1} + 0.95\vec{m_3})$$
$$= 1.42\vec{m_1} + 1.66\vec{m_2} + 0.95\vec{m_3}$$

$$\vec{d_2} = \vec{k_1} = 0.55\vec{m_1} + 0.83\vec{m_2}$$

$$\vec{d_3} = \vec{k_2} + 3\vec{k_3} = \vec{m_3} + 3(0.32\vec{m_1} + 0.95\vec{m_2})$$
$$= 0.96\vec{m_1} + 3.85\vec{m_3}$$

$$\vec{d_4} = 2\vec{k_1} = 2(0.55\vec{m_1} + 0.83\vec{m_2}) = 1.1\vec{m_1} + 1.66\vec{m_2}$$

$$\vec{q} = (0.55\vec{m_1} + 0.83\vec{m_2}) + (\vec{m_3})$$
$$= 0.55\vec{m_1} + 0.83\vec{m_2} + \vec{m_3}$$

GVSM例子：相似度计算

$$sim(d_1, q) = \frac{(1.42)(.55) + (1.66)(.83) + (.95)(1)}{[1.42^2 + 1.66^2 + 0.95^2]^{1/2}[0.55^2 + 0.83^2 + 1^2]^{1/2}} = 0.9234 ,$$

$$sim(d_2, q) = \frac{(.55)(.55) + (.83)(.83) + (0)(1)}{[0.55^2 + 0.83^2]^{1/2}[0.55^2 + 0.83^2 + 1^2]^{1/2}} = 0.7056 ,$$

$$sim(d_3, q) = \frac{(.96)(.55) + (0)(.83) + (3.85)(1)}{[0.96^2 + 3.85^2]^{1/2}[0.55^2 + 0.83^2 + 1^2]^{1/2}} = 0.7819 ,$$

$$sim(d_4, q) = \frac{(1.1)(.55) + (1.66)(.83) + (0)(1)}{[1.1^2 + 1.66^2]^{1/2}[0.55^2 + 0.83^2 + 1^2]^{1/2}} = 0.7056 .$$

- 因此对于查询，输出文档的排序为：

$$d_1 > d_3 > d_2 \geq d_4$$

关于GVSM的讨论

- 优点：
  - 从理论上来看，广义向量空间模型确实提出了相当重要的新见解
  - 广义向量空间模型**GVSM**可以较好地应用在如跨语言信息检索中，其基本思想是根据双语训练文档集分别建立源语与目标语的检索词一文档关联矩阵
- 缺点：
  - 由于词一词关联的使用并不必然产生已改进的检索效果，所以广义向量空间模型在哪些方面优于经典向量模型是不明确的
  - 由于那些计算向量时考虑的有效的最小项与集合中文档的数目成正比，在大型集合中用广义模型计算排序，其代价是相当高的

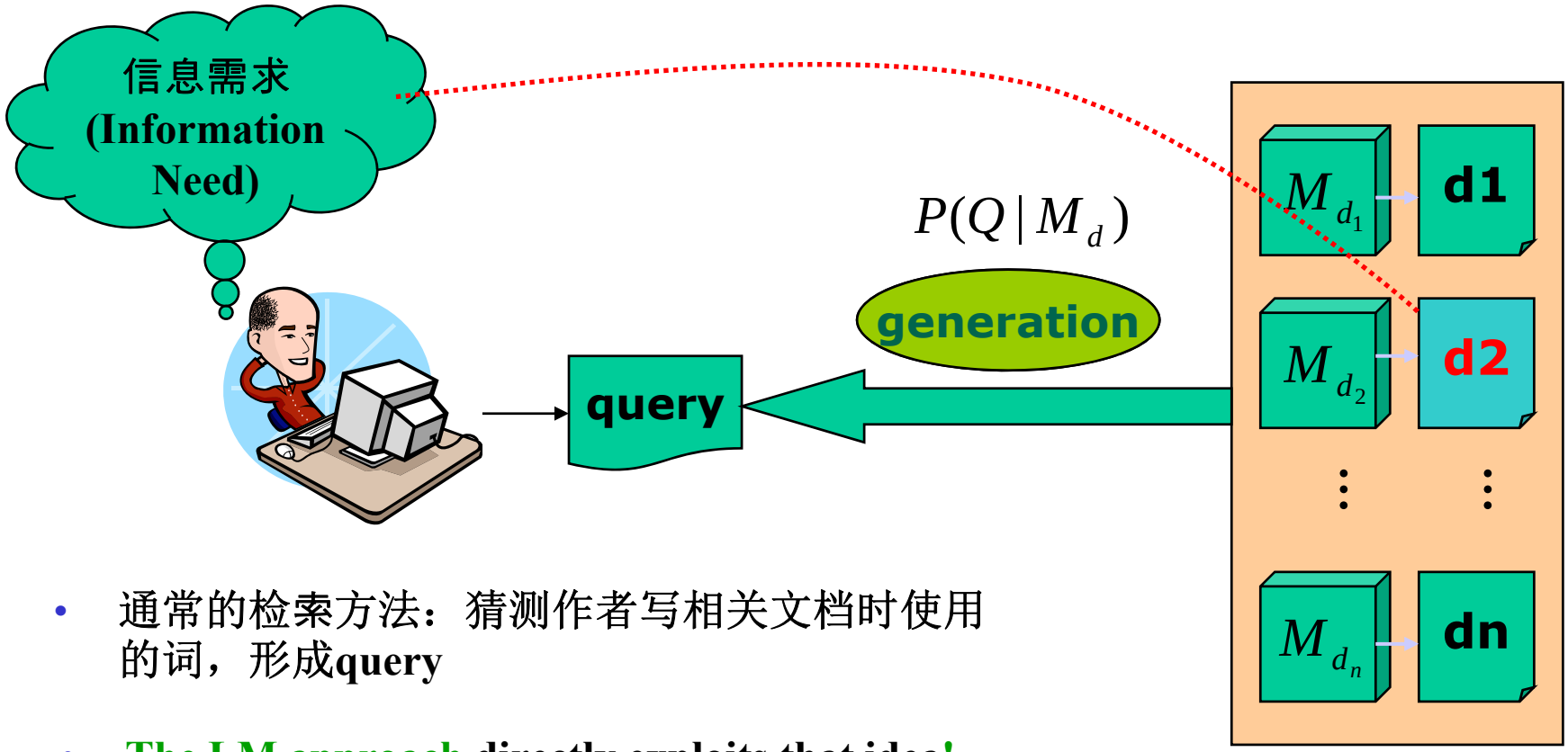
主要内容

- 检索模型的基本概念
- 布尔模型
- 向量空间模型
- 概率模型
- 扩展的检索模型
  - 模糊集合模型
  - 扩展布尔模型
  - 潜语义索引模型
  - 广义向量空间模型
  - 语言模型

语言模型
(Language Model, LM)

- 麻省(University of Massachusetts, UMass)大学Bruce Croft等人于1998年提出，包括一系列模型，代表系统Lemur（http://lemurproject.org）
  - 最大似然模型：把相关度看成是每篇文档对应的语言下生成该查询的可能性
    - 类比：作者A和作者B写的文章用词风格很不相同，可以统计用词的概率（语言），然后对应一篇新的文章，判断是A写的还是B写的
  - 翻译模型：假设查询经过某个噪声信道变形成某篇文章，则由文档还原成该查询的概率（翻译模型）可以视为相关度
  - KL（Kullback-Leibler）距离模型：查询对应某种语言，每篇文档对应某种语言，查询语言和文档语言的KL距离作为相关度度量

语言模型的原理



- 通常的检索方法：猜测作者写相关文档时使用的词，形成query
- **The LM approach directly exploits that idea!**

随机的语言模型

- 建模语言中产生语句的概率

Model M

0.2	the
0.1	a
0.01	man
0.01	woman
0.03	said
0.02	likes
…	

the	man	likes	the	woman
0.2	0.01	0.02	0.2	0.01

multiply

P(s | M) = 0.00000008

随机的语言模型（2）

Model M_1	
0.2	the
0.01	class
0.0001	sayst
0.0001	pleaseth
0.0001	yon
0.0005	maiden
0.01	woman

Model M_2	
0.2	the
0.0001	class
0.03	sayst
0.02	pleaseth
0.1	yon
0.01	maiden
0.0001	woman

the	class	pleaseth	yon	maiden
0.2	0.01	0.0001	0.0001	0.0005
0.2	0.0001	0.02	0.1	0.01

$$P(s|M_2) > P(s|M_1)$$

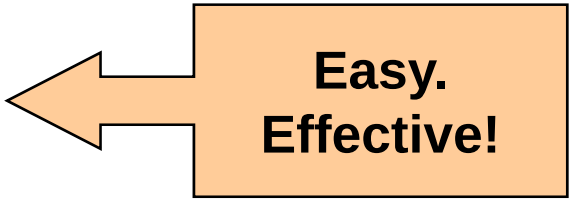
一元和多元语言模型
(Unigram and higher-order models)

P (● ● ● ●)

= P (●) P (● | ●) P (● | ● ●) P (● | ● ● ●)

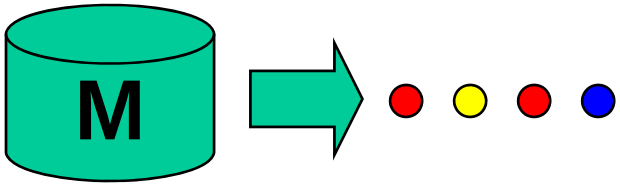
- 一元（Unigram ）语言模型

  P (●) P (●) P (●) P (●)



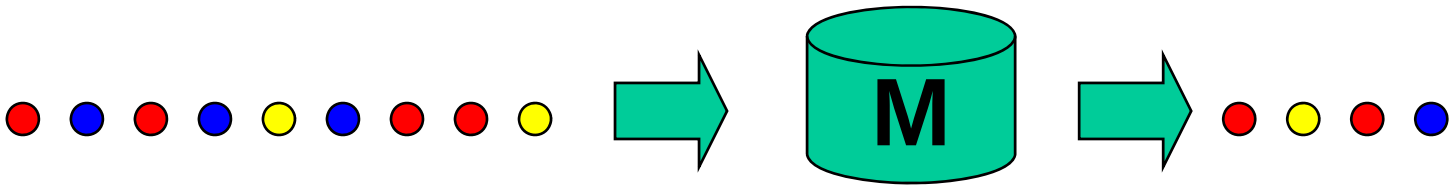
- 二元（Bigram）语言模型

  P (●) P (● | ●) P (● | ●) P (● | ●)



IR中的语言模型

- 每篇文档对应一个模型M
- 按P(*d* | *q*)对文档排序
- P(*d* | *q*) = P(*q* | *d*) × P(*d*) / P(*q*)
  - P(*q*) 对所有文档而言都是相同的，所以忽略
  - P(*d*)（先验概率）也可以视为相同
    - 但也利用某些先验知识，如权威性、长度、类型等
  - P(*q* | *d*) 就是在给定*d*的模型下*q*的概率
- 但模型 M 是不知道的
  - 只有代表这个模型的样例文本
- 从样例文本中来估计Model
- 然后计算观察到的文本概率



基于概率语言模型的检索过程

- 把查询（query）的产生当作一个随机过程
- 方法：
  - 为每个文档建立一个语言模型
  - 估计每个文档模型产生这个query的概率
  - 排序：按这个概率对文档排序
  - 通常使用一元语言模型（Unigram model）

一元模型的假设：
Given a particular language model, the query terms occur independently

最大似然估计（MLE，Maximum Likelihood Estimation）

- 在文档d的语言模型M_d下，采用最大似然估计得到的查询生成概率

$$P(q \mid M_d) = \prod_{t \in q} \hat{P}_{mle}(t \mid M_d) = \prod_{t \in q} \frac{tf_{t,d}}{L_d} \qquad (2\text{-}55)$$

- 平滑：对未出现的词项赋予一个合理的假设概率（没有出现在文档中的词项按它出现在文档集中的概率来代替）

$$P(t \mid d) = \lambda P_{mle}(t \mid M_d) + (1-\lambda) P_{mle}(t \mid M_c)$$

- 对于查询q：

$$P(d \mid q) \propto p(d) \prod_{t \in q} [(1-\lambda) P(t \mid M_c) + \lambda P(t \mid M_d)] \qquad (2\text{-}58)$$

general language model

individual-document model

查询例子

- 假设一个文档集合包含两个文档，如下：

  D_1: Xyzzy reports a profit but revenue is down

  D_2: Quorus narrows quarter loss but revenue decreases further

  Q: *revenue down*

- 用最大似然的一元语言模型建模　　$\lambda = 1/2$

$$P(q \mid d_1) = (\frac{1}{2}P(t_{revenue} \mid M_c) + \frac{1}{2}P(t_{revenue} \mid M_d))(\frac{1}{2}P(t_{down} \mid M_c) + \frac{1}{2}P(t_{down} \mid M_d))$$

$$= [\frac{1}{2}(\frac{2}{16} + \frac{1}{8})] \times [\frac{1}{2}(\frac{1}{16} + \frac{1}{8})] = \frac{3}{256}$$

$$P(q \mid d_2) = [\frac{1}{2}(\frac{2}{16} + \frac{1}{8})] \times [\frac{1}{2}(\frac{1}{16} + \frac{0}{8})] = \frac{1}{256}$$

- 文档排序输出是**$D_1 > D_2$**

关于语言模型的讨论

- 从基于概率的语言模型来处理文本检索问题的新颖的方法
  - 概念简单，可解释
  - 规范化的数学模型
  - 关于文档集统计信息的自然使用，而不是启发式的
- 如果以下条件满足，语言模型可以成为有效的检索模型
  - 语言模型是数据的精确表示
  - 用户有关于词项分布的认识

推荐阅读和站点

- **An Introduction to Information Retrieval**
  - **Ch11: Probabilistic information retrieval**
- **《网络信息检索》第二章**
  - **2.5，2.6，2.7**

- **M. P. Jay and W. B. Croft, "A language modeling approach to information retrieval,“ ACM SIGIR 1998**
- **中国科学院研究生院秋季课程“现代信息检索”，http://ir.ict.ac.cn/ircourse/**
- **Lemur, http://lemurproject.org**