概率检索模型

提纲

- 向量空间模型回顾
- 基本概率统计知识
- 概率排序原理
- BIM模型
- BM25模型

向量空间模型

- 文档、查询都表示成向量
- 计算两个向量之间的相似度：余弦相似度、内积相似度等等
- 在向量表示中的词项权重计算方法主要是tf-idf公式，实际考虑tf、idf及文档长度3个因素

向量空间模型的优缺点

- 优点
  - 简洁直观，可以应用到很多其他领域(文本分类、生物信息学)
  - 支持部分匹配和近似匹配，结果可以排序
  - 检索效果不错
- 缺点
  - 理论上不够：基于直觉的经验性公式
  - 索引项之间的独立性假设与实际不符：实际上，term的出现之间是有关系的，不是完全独立的。如：“王励勤”“乒乓球”的出现不是独立的。

本讲内容

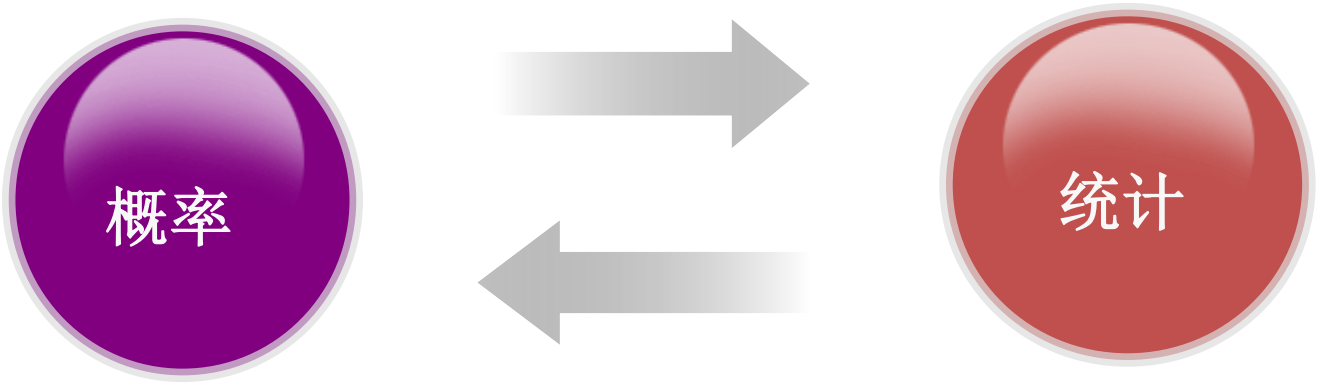
- 概率基础知识
- 基于概率理论的检索模型
- 二值独立概率模型 BIM：不考虑词项频率和文档长度
- BM25模型：考虑词项频率和文档长度

提纲

- 向量空间模型回顾
- 基本概率统计知识
- 概率排序原理
- BIM模型
- BM25模型

概率 vs. 统计

概率是统计的理论基础



统计是概率的实际应用

典型问题：已知某数据总体满足某分布，抽样得到某数据的概率是多少？

典型问题：已知某抽样数据(或总体分布)，判断总体的分布(或分布参数)是多少？

概率统计初步

- 随机试验与随机事件
- 概率和条件概率
- 乘法公式、全概率公式、贝叶斯公式
- 随机变量
- 随机变量的分布

随机试验和随机事件

- 随机试验：可在相同条件下重复进行；试验可能结果不止一个，但能确定所有的可能结果；一次试验之前无法确定具体是哪种结果出现。
  - 掷一颗骰子，考虑可能出现的点数
- 随机事件：随机试验中可能出现或可能不出现的情况叫“随机事件”
  - 掷一颗骰子，4点朝上

概率和条件概率

- 概率：直观上来看，事件A的概率是指事件A发生的可能性，记为P(A)
  - 掷一颗骰子，出现6点的概率为多少？
- 条件概率：已知事件A发生的条件下，事件B发生的概率称为A条件下B的条件概率，记作P(B|A)
  - 30颗红球和40颗黑球放在一块，请问第一次抽取为红球的情况下第二次抽取黑球的概率？

乘法公式、全概率公式和贝叶斯公式

- 乘法公式：
  - $P(AB)=P(A)P(B|A)$
  - $P(A_1A_2\ldots A_n)=P(A_1)P(A_2|A_1)\,P(A_3|A_2,A_1)\ldots P(A_n|A_1\ldots A_{n-1})$
- 全概率公式：$A_1A_2\ldots A_n$是整个样本空间的一个划分

$$P(B)=\sum_{i=1}^{n}P(A_i)P(B\mid A_i)$$

- 贝叶斯公式：$A_1A_2\ldots A_n$是整个样本空间的一个划分

$$P(A_j\mid B)=\frac{P(A_j)P(B\mid A_j)}{\sum_{i=1}^{n}P(A_i)P(B\mid A_i)},(j=1,\ldots,n)$$

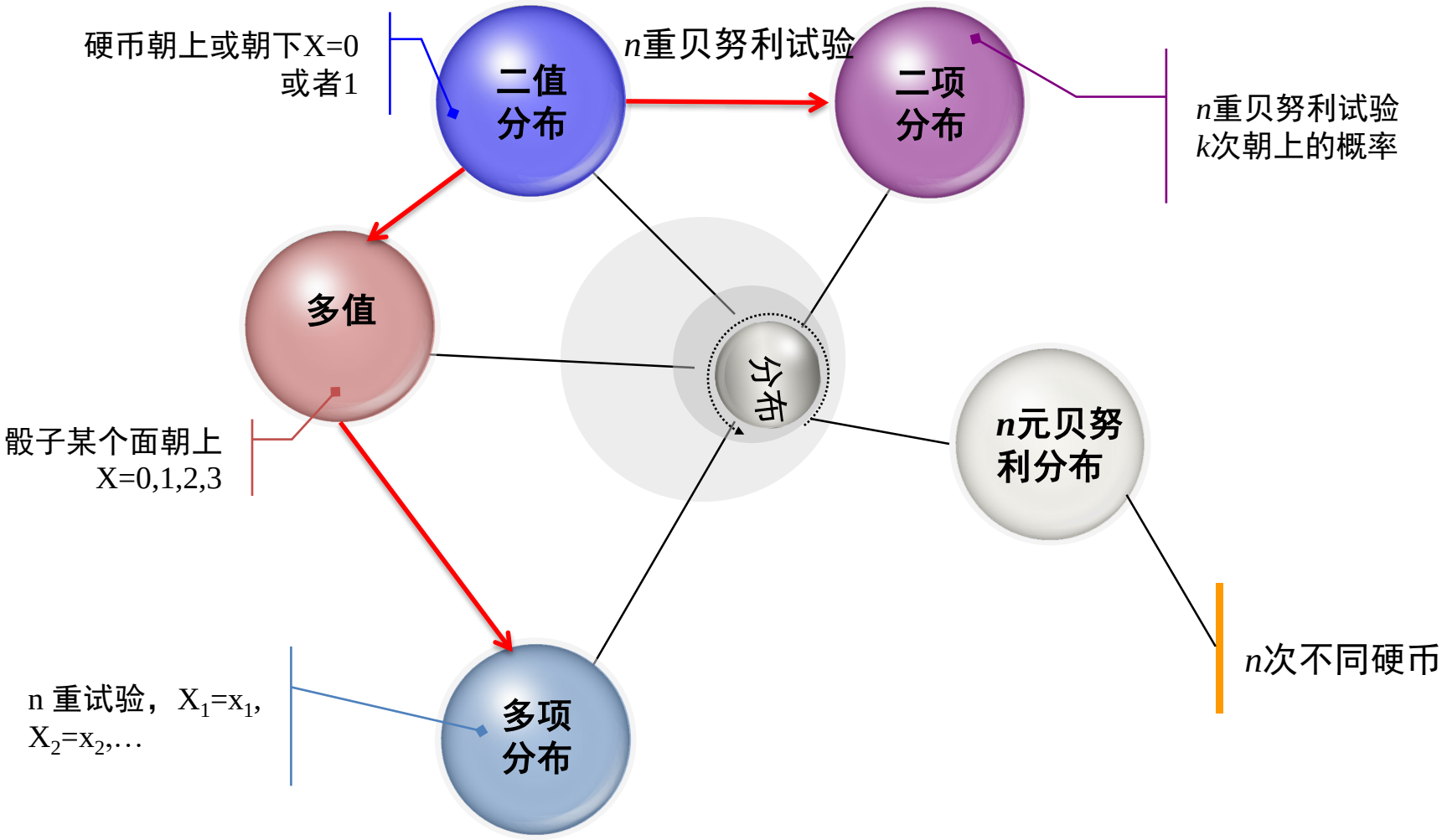
事件的独立性

- 两事件独立：事件A、B，若P(AB)=P(A)P(B)，则称A、B独立
- 三事件独立：事件A B C，若满足P(AB)=P(A)P(B), P(AC)=P(A)P(C),P(BC)=P(B)P(C), P(ABC)=P(A)P(B)P(C)，则称A、B、C独立
- 多事件独立：两两独立、三三独立、四四独立….

随机变量

- 随机变量：若随机试验的各种可能的结果都能用一个变量的取值(或范围)来表示，则称这个变量为随机变量，常用X、Y、Z来表示
  - (离散型随机变量)：掷一颗骰子，可能出现的点数X (可能取值1、2、3、4、5、6)
  - (连续型随机变量)：北京地区的温度(-15~45)

各种分布关系图


硬币朝上或朝下X=0
或者1
二值
分布
n重贝努利试验
二项
分布
n重贝努利试验
k次朝上的概率
多值
分布
n元贝努
利分布
骰子某个面朝上
X=0,1,2,3
n 重试验，$X_1=x_1$,
$X_2=x_2$,…
多项
分布
n次不同硬币

提纲

- 向量空间模型回顾
- 基本概率统计知识
- 概率排序原理
- BIM模型
- BM25模型

概率检索模型

- 给定查询，计算每个文档的相关度
- 检索系统对用户查询的理解是非确定的(uncertain)，对返回结果的猜测也是非确定的
- 而概率理论为非确定推理提供了坚实的理论基础
- 概率检索模型可以计算文档和查询相关的可能性

概率检索模型

- 概率检索模型：通过概率的方法将查询和文档联系起来
  - 定义3个随机变量R、Q、D：相关度R={0,1}，查询Q={q_1,q_2,…}，文档D={d_1,d_2,…}
  - 通过计算条件概率$P(R=1|Q=q,D=d)$来度量文档和查询的相关度。
- 概率模型包括一系列模型，如最经典的二值独立概率模型BIM、BM25模型等(还有贝叶斯网络模型)。
- 1998出现的基于统计语言建模的信息检索模型本质上也是概率模型的一种。

概率排序原理PRP(Probability Ranking Principle)

- 利用概率模型来估计每篇文档和需求的相关概率$P(R=1|d,q)$，然后对结果进行排序。
- 最简单的PRP情况
  - 检索没有任何代价因子，或者说不会对不同行为或错误采用不同的权重因子。
  - 在返回一篇不相关文档或者返回一篇相关文档不成功的情况下，将失去1分(在计算精确率时这种基于二值的情形也往往称为1/0风险)。
  - 检索的目标是对于用户任意给定的k值，返回可能性最高的前k篇文档作为结果输出。即，PRP希望可以按照$P(R=1|d,q)$值的降序来排列所有文档。
- 定理**11-1** 在1/0损失的情况下，PRP对于最小化期望损失(也称为贝叶斯风险)而言是最优的。

基于检索代价的概率排序原理

- C_0表示检索到一篇不相关文档发生的代价
- C_1表示未检索到一篇相关文档所发生的代价
- PRP认为，如果对于一篇特定的文档***d***及所有其他未返回的文档***d′*** 都满足
  - $C_0 \cdot P(R=0|d) - C_1 \cdot P(R=1|d) \leq C_0 \cdot P(R=0|d') - C_1 \cdot P(R=1|d')$
    - $C_0 \cdot [1-P(R=1|d)] - C_1 \cdot P(R=1|d) = C_0 - (C_0+C_1)P(R=1|d)$
    - 两者相减表示返回文档d的代价函数，也即此时前者越低越好，后者越高越好，即$P(R=1|d)$越高越好
  - 那么d 就应该是下一篇被返回的文档。

提纲

- 向量空间模型回顾
- 基本概率统计知识
- 概率排序原理
- BIM模型
- BM25模型

二值独立概率模型BIM

- 二值独立概率模型(Binary Independence Model，简称BIM)
  - 伦敦城市大学Robertson及剑桥大学Sparck Jones 1970年代提出，代表系统OKAPI
- Bayes公式是理解BIM的关键
- 通过Bayes公式对条件概率$P(R=1|q,d)$进行计算
- 是一种生成式(generative)模型

贝叶斯公式

设$A_1,A_2,\ldots,A_n$是完备事件组，则对任一事件B，有

先验概率
Prior Prob.

相似度
Likelihood

后验概率
Posterior Prob.

$$P(A_i \mid B)=\frac{P(A_i)P(B\mid A_i)}{\sum_{j=1}^{n}P(A_j)P(B\mid A_j)} \qquad i=1,2,\cdots,n$$

全概率
Sum over Space hypotheses

该公式于1763年由贝叶斯(Bayes)给出. 它是在观察到事件B已发生的条件下，寻找导致B发生的每个原因A_i的概率.

二值独立概率模型BIM

- 为了对概率函数$P(R|q,d)$进行估计，引入一些简单假设。
  - “二值”等价于布尔值：文档和查询都表示为词项出现与否的布尔向量。也就是说，文档d表示为向量$\vec{x}=(x_1, \ldots, x_m)$，其中当词项$t$出现在文档$d$中时，$x_t=1$，否则$x_t=0$。由于不考虑词项出现的次数及顺序，许多不同的文档可能都有相同的向量表示。
  - 类似地，将查询q表示成词项出现向量$\vec{q}$。
  - “独立性”：指词项在文档中的出现是互相独立的，BIM 并不识别词项之间的关联。

BIM Bayes公式的使用

- 在BIM 模型下，基于词项出现向量的概率

$P(R|\vec{x},\vec{q})$对概率$P(R|d,q)$建模，利用贝叶斯定理，有

$$P(R=1|\vec{x},\vec{q})=\frac{P(\vec{x}|R=1,\vec{q})P(R=1|\vec{q})}{P(\vec{x}|\vec{q})}$$

$$P(R=0|\vec{x},\vec{q})=\frac{P(\vec{x}|R=0,\vec{q})P(R=0|\vec{q})}{P(\vec{x}|\vec{q})}$$

$P(\vec{x}|R=1,\vec{q})$ 和 $P(\vec{x}|R=0,\vec{q})$ 分别表示当返回一篇相关或不相关文档时生成文档$\vec{x}$的概率

$P(R=1|\vec{q})$ 和 $P(R=0|\vec{q})$ 分别表示对于查询$\vec{q}$ 返回一篇相关和不相关文档的**先验**概率。

BIM排序函数的推导

- 对每个d定义优势率函数：

优势率：$O(A)=\frac{P(A)}{P(\bar{A})}=\frac{P(A)}{1-P(A)}$

$$O(R|\vec{x},\vec{q})=\frac{P(R=1|\vec{x},\vec{q})}{P(R=0|\vec{x},\vec{q})}=\frac{\frac{P(R=1|\vec{q})P(\vec{x}|R=1,\vec{q})}{P(\vec{x}|\vec{q})}}{\frac{P(R=0|\vec{q})P(\vec{x}|R=0,\vec{q})}{P(\vec{x}|\vec{q})}}$$

对同一q是常量，与文档无关，对排序不起作用

$$=\frac{P(R=1|\vec{q})P(\vec{x}|R=1,\vec{q})}{P(R=0|\vec{q})P(\vec{x}|R=0,\vec{q})}$$

$$O(R|\vec{q})=\frac{P(R=1|\vec{q})}{P(R=0|\vec{q})}$$

$$\frac{P(\vec{x}|R=1,\vec{q})}{P(\vec{x}|R=0,\vec{q})} \overset{\text{词项间独立性}}{=} \prod_{t=1}^{m}\frac{P(x_t|R=1,\vec{q})}{P(x_t|R=0,\vec{q})}$$

$$O(R|\vec{x},\vec{q})=O(R|\vec{q})\prod_{t=1}^{m}\frac{P(x_t|R=1,\vec{q})}{P(x_t|R=0,\vec{q})}$$

$$O(R|\vec{x},\vec{q}) = O(R|\vec{q})\prod_{t=1}^{m}\frac{P(x_t|R=1,\vec{q})}{P(x_t|R=0,\vec{q})}$$

二值：X_t的取值={0,1}

$$O(R|\vec{x},\vec{q}) = O(R|\vec{q})\prod_{t:x_t=1}\frac{P(x_t=1|R=1,\vec{q})}{P(x_t=1|R=0,\vec{q})}\prod_{t:x_t=0}\frac{P(x_t=0|R=1,\vec{q})}{P(x_t=0|R=0,\vec{q})}$$

p_t词项出现在一篇相关文档中的概率

$$p_t = P(x_t=1|R=1,\vec{q})$$

$$u_t = P(x_t=1|R=0,\vec{q})$$

u_t词项出现在一篇不相关文档中的概率

$$O(R|\vec{x},\vec{q}) = O(R|\vec{q})\prod_{t:x_t=q_t=1}\frac{p_t}{u_t}\prod_{t:x_t=0,q_t=1}\frac{1-p_t}{1-u_t}$$

q_t=1表明词项在查询中出现 (q_t=0不考虑)

$$O(R|\vec{x},\vec{q}) = O(R|\vec{q}) \prod_{t:x_t=q_t=1} \frac{p_t}{u_t} \prod_{t:x_t=0,q_t=1} \frac{1-p_t}{1-u_t}$$

出现在文档和查询中的查询词项的概率

出现在查询，但不出现在文档中的查询词项的概率

$$O(R|\vec{x},\vec{q}) = O(R|\vec{q}) \prod_{t:x_t=q_t=1} \frac{p_t}{u_t} \frac{\prod_{t:x_t=0,q_t=1} \frac{1-p_t}{1-u_t} \prod_{t:x_t=1,q_t=1} \frac{1-p_t}{1-u_t}}{\prod_{t:x_t=1,q_t=1} \frac{1-p_t}{1-u_t}}$$

上下同乘

两项合并

$$= O(R|\vec{q}) \prod_{t:x_t=q_t=1} \frac{p_t}{u_t} \frac{\prod_{t:q_t=1} \frac{1-p_t}{1-u_t}}{\prod_{t:x_t=1,q_t=1} \frac{1-p_t}{1-u_t}} = O(R|\vec{q}) \frac{\prod_{t:x_t=q_t=1} \frac{p_t}{u_t}}{\prod_{t:x_t=q_t=1} \frac{1-p_t}{1-u_t}} \prod_{t:q_t=1} \frac{1-p_t}{1-u_t}$$

$$= O(R|\vec{q}) \prod_{t:x_t=q_t=1} \frac{p_t(1-u_t)}{u_t(1-p_t)} \prod_{t:q_t=1} \frac{1-p_t}{1-u_t}$$

基于出现在文档中的查询词项来计算

考虑所有查询词项，对于给定的查询而言是常数

RSV(Retrieval Status Value，检索状态值)

排序函数只需计算 $\prod_{t:x_t=q_t=1} \frac{p_t(1-u_t)}{u_t(1-p_t)}$

最终用于排序的是

$$RSV_d = \log \prod_{t:x_t=q_t=1} \frac{p_t(1-u_t)}{u_t(1-p_t)} = \sum_{t:x_t=q_t=1} \log \frac{p_t(1-u_t)}{u_t(1-p_t)}$$

c_t查询词项的优势率比率的对数值

$$c_t = \log \frac{p_t(1-u_t)}{u_t(1-p_t)} = \log \frac{p_t}{1-p_t} + \log \frac{1-u_t}{u_t}$$

$$RSV_d = 1 \sum_{t:x_t=q_t=1} c_t$$

c_t如何计算？

求c_t：理论上的概率估计方法

文档	相关R=1	不相关R=0
词项出现x_t=1	p_t	u_t
词项不出现x_t=0	1-p_t	1-u_t

df_t包含词项t的文档数目

文档	相关R=1	不相关R=0	总计
词项出现x_t=1	s	df_t-s	df_t
词项不出现x_t=0	S-s	(N-df_t)-(S-s)	N-df_t
总计	S	N-S	N

$$p_t = P(x_t = 1 | R = 1, \vec{q}) = s / S$$ p_t词项出现在一篇相关文档中的概率

$$u_t = P(x_t = 1 | R = 0, \vec{q}) = (df_t - s) / (N - S)$$

u_t词项出现在一篇不相关文档中的概率

$$c_t = \log \frac{p_t(1-u_t)}{u_t(1-p_t)} = K(N, df_t, S, s) = \log \frac{s/(S-s)}{(df_t - s)/((N - df_t) - (S - s))}$$

平滑(smoothing)

- 在减少出现事件的概率估计值的同时提高未出现事件的概率估计值的方法称为平滑

$$c_t = \log \frac{p_t(1-u_t)}{u_t(1-p_t)} = K(N, df_t, S, s) = \log \frac{s/(S-s)}{(df_t - s)/((N-df_t)-(S-s))}$$

能出现的0 概率？

一种最简单的平滑方法就是对每个观察到的事件的数目都加上一个数α

相当于在所有词汇表上使用了均匀分布作为一个贝叶斯先验α的大小表示对均匀分布的信心强度

$$\hat{c}_t = K(N, df_t, S, s) = \log \frac{(s+\frac{1}{2})/(S-s+\frac{1}{2})}{(df_t - s + \frac{1}{2})/((N-df_t)-(S-s+\frac{1}{2}))}$$

求c_t：实际中的概率估计方法

$$c_t = \log \frac{p_t(1-u_t)}{u_t(1-p_t)} = \log \frac{p_t}{1-p_t} + \log \frac{1-u_t}{u_t}$$

u_t的估算：假设相关文档S只占所有文档的极小一部分，那么可通过整个文档集的统计数字来计算与不相关文档有关的量。

$$u_t = (df_t - s)/(N - S) \approx df_t / N$$

p_t的估算：

- 如果知道某些相关文档，那么可以利用这些已知相关文档中的词项出现频率来对p_t进行估计
- Croft和Harper（1979）在组合匹配模型（combination match model）中提出了利用常数来估计p_t 的方法。
- Greiff（1998）提出

$$p_t = \frac{1}{3} + \frac{2}{3}\frac{df_t}{N}$$

BIM模型小结

- 目标是求排序函数

$$O(R|\vec{x},\vec{q}) = \prod_{t=1}^{m} \frac{P(x_t|R=1,\vec{q})}{P(x_t|R=0,\vec{q})}$$

  - 首先估计或计算每个term分别在相关文档和不相关文档中的出现概率$p_t=P(t|R=1)$及$u_t=P(t|R=0)$
  - 然后根据独立性假设，将$P(d|R=1)/P(d|R=0)$ 转化为p_t和u_t的某种组合，将p_t和u_t代入即可求解。
    - 转化为后验概率的估算

$$\prod_{t:x_t=q_t=1} \frac{p_t(1-u_t)}{u_t(1-p_t)}$$

$$P(R=1|\vec{x},\vec{q}) = \frac{P(\vec{x}|R=1,\vec{q})P(R=1|\vec{q})}{P(\vec{x}|\vec{q})}$$

$$P(R=0|\vec{x},\vec{q}) = \frac{P(\vec{x}|R=0,\vec{q})P(R=0|\vec{q})}{P(\vec{x}|\vec{q})}$$

转化为p_t和u_t的估算

BIM模型的优缺点

- 特点
  - 优点
    - 建立在数学基础上，理论性较强
  - 缺点
    - 需要估计参数
    - 原始的BIM没有考虑TF、文档长度因素
    - BIM中同样存在词项独立性假设

提纲

- 向量空间模型回顾
- 基本概率统计知识
- 概率排序原理
- BIM模型
- BM25模型

BIM→BM25

- Okapi BM25: 一个非二值的模型
- BIM模型最初主要为较短的编目记录(**catalog record**)和长度大致相当的摘要所设计，在这些环境下它用起来也比较合适。但是对现在的全文搜索文档集来说，很显然模型应该重视词项频率和文档长度。
- **BM25**权重计算机制(**BM25 weighting scheme**)或**Okapi**权重计算机制(**Okapi weighting**) (**Spärck Jones**等人**2000**)
  - 基于词项频率、文档长度等因子来建立概率模型
  - 不会引入过多的模型参数

Okapi BM25: 一个非二值模型

- 考虑词项t_i在文档中的tf-idf权重，有：

$$RSV(Q,D)=\sum_{t_i\in D\cup Q} W_i^{\mathrm{IDF}}\frac{(k_1+1)tf_{t_i,D}}{k_1((1-b)+b\times(L_D/L_{ave}))+tf_{t_i,D}}$$

  - W_i^{IDF}：词项t_i的IDF权重
  - $tf_{t_i,D}$：词项t_i在文档D中的词项频率
  - L_D (L_{ave}): 文档D的长度(整个文档集的平均长度)
  - k_1: 用于控制文档中词项频率权重的调节参数
  - b: 用于控制文档长度权重的调节参数

实验的baseline: b = 0.75, k_1=2

- 如果查询比较长，则加入查询的tf

$$RSV(Q,D)=\sum_{t_i\in D\cup Q} W_i^{\mathrm{IDF}}\cdot\frac{(k_1+1)tf_{ti,D}}{k_1((1-b)+b\times(L_D/L_{ave}))+tf_{ti,D}}\cdot\frac{(k_3+1)tf_{t_i,Q}}{k_3+tf_{t_i,Q}}$$

  - W_i^{IDF}：词项t_i的IDF权重
  - $tf_{t_i,D}$: 词项t_i在Q中的词项频率
  - k_3:用于控制查询中词项频率比重的调节参数
  - 没有查询长度的归一化 (由于查询对于所有文档都是固定的)
  - 理想情况下，上述参数都必须在开发测试集上调到最优。一般情况下，实验表明，k_1 和 k_3 应该设在 1.2到2之间， b 设成 0.75。

另一个BM25写法

$$RSV(Q,D)=\sum_{t_i\in D\cup Q}\ln\frac{N-df_i+0.5}{df_i+0.5}\cdot\frac{(k_1+1)tf_{ti,D}}{k_1((1-b)+b\times(L_D/L_{ave}))+tf_{ti,D}}\cdot\frac{(k_3+1)tf_{t_i,Q}}{k_3+tf_{t_i,Q}}$$

- df_i是词项t_i的df