信息检索与Web搜索

第12讲 概率检索模型

Probabilistic Information Retrieval

授课人：高曙明

改编自“现代信息检索”网上公开课件（http://ir.ict.ac.cn/~wangbin）

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?

相关概率公式

- **乘法公式：**
  - $P(AB)=P(A)P(B|A)$
- **全概率公式：** $P(B)=\sum_{i=1}^{n}P(A_i)P(B\mid A_i)$
- **贝叶斯公式：** $P(A\mid B)=\frac{P(A,B)}{P(B)}=\frac{P(B\mid A)P(A)}{P(B)}$
- **优势率：** $O(A)=\frac{P(A)}{P(\bar{A})}=\frac{P(A)}{1-P(A)}$

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表示为一个变量的取值（或范围），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)

概率检索模型

- 检索系统中，给定查询，计算每个文档与查询的相关度
- 检索系统对用户查询的理解是非确定的(uncertain)，对返回结果的确定也是非确定的
- 而概率理论为非确定推理提供了坚实的理论基础
- 可以基于概率计算文档和查询相关的可能性大小

概率检索模型

- **概率检索模型：**通过概率的方法确定查询与文档之间的相关度
  - 定义3个随机变量R、Q、D：相关度$R=\{0,1\}$，查询$Q=\{q_1,q_2,\cdots\}$，文档$D=\{d_1,d_2,\cdots\}$，则可以通过计算条件概率$P(R=1|Q=q,D=d)$来度量文档和查询的相关度
- 概率模型包括一系列模型：最经典的二值独立概率模型BIM、BM25模型等等
- 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)

- **简单地说：**如果文档按照与查询的相关概率大小返回，那么该返回结果是所有可能获得结果中效果最好的
- **严格地说：**如果文档按照与查询的相关概率大小返回，而这些相关概率又能够基于已知数据进行尽可能精确的估计，那么该返回结果是所有基于已知数据获得的可能的结果中效果最好的

https://blog.csdn.net/guoziqing506/article/details/81633285

二值独立概率模型BIM

- 二值独立概率模型(Binary Independence Model)
  - 文档和查询都表示为词项出现与否的布尔向量，d表示为：

    $\vec{x} = (x_1, \ldots, x_m)$，$x_i=0$ 或 $x_i=1$

  - 独立性：假设词项在文档中的出现是相互独立的
  - 用$P(R|\vec{x}, \vec{q})$对概率$P(R|d, q)$建模
- BIM模型通过Bayes公式对所求条件概率$P(R|\vec{x}, \vec{q})$展开进行计算

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

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

公式推导

$$P(R=1|\vec{x},\vec{q}\,) \xrightarrow{\text{贝叶斯定理}} \frac{P(\vec{x},\vec{q}|R=1)P(R=1)}{P(\vec{x},\vec{q})}$$

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

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

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

排序函数推导

- 给定查询q，按 $P(R=1|\vec{x},\vec{q})$ 来排序
- 不需要直接计算概率值
- 故，可采用文档相关性的优势率来排序
  - 它是相关性概率的单调递增函数
  - 可以忽略 $P(\vec{x}|\vec{q})$，简化了计算

排序函数推导（续）

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

- 在独立性假设下，我们有：

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

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

- 由于x_t为布尔变量，所以有：

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

排序函数推导（续）

- 令：

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

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

- 则：

document	relevant (R = 1)	nonrelevant (R = 0)
Term present $x_t = 1$	p_t	u_t
Term absent $x_t = 0$	$1 - p_t$	$1 - u_t$

- 假定当q_t=0时，$p_t = u_t$

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

- 进一步有：

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

仅关于query，与文档无关

文档与query中出现的词的并集

排序函数推导（续）

- 排序中唯一需要估计的量是：

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

- 定义：

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

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

- C_t 是查询词项的优势率比率的对数值

理论上的概率估计方法

- 给定一词项t，相关数据列表为：

	documents	relevant	nonrelevant	Total
Term present	$x_t = 1$	s	$\mathrm{df}_t - s$	df_t
Term absent	$x_t = 0$	$S - s$	$(N - \mathrm{df}_t) - (S - s)$	$N - \mathrm{df}_t$
	Total	S	$N - S$	N

  - N为总文档数目，df_t是包含t的文档数目

- 假定上述数据已知，则有：

$$p_t = s/S, \quad u_t = (\mathrm{df}_t - s)/(N - S)$$

$$c_t = K(N, \mathrm{df}_t, S, s) = \log \frac{s/(S-s)}{(\mathrm{df}_t - s)/((N - \mathrm{df}_t) - (S - s))}$$

实际中的概率估计方法

- 假设给定一查询q，其相关文档只占全部文档的极小部分，则S、s可忽略，$u_t=df_t/N$, 于是有：

$$\log[(1-u_t)/u_t] = \log[(N-df_t)/df_t] \approx \log N/df_t$$

- p_t的估计方法如下：
  - 如果已知某些相关文档，则以其为基础估计 p_t
  - 假设 p_t 是一个常数，比如0.5
  - $p_t = \frac{1}{3} + \frac{2}{3}df_t/N$ ，因为直观上p_t会随df_t的增长而增长

面向概率计算的相关反馈

- **基本思想：**通过利用用户反馈信息不断提高p_t的精确性来提高检索效果
- **基本过程**
  - 给出p_t和u_t的初始估计
  - 利用当前的p_t、u_t确定相关文档集返回给用户
  - 用户交互选择相关文档
  - 利用已知的相关文档和不相关文档对p_t、u_t进行重新估计
  - 重复第2-4步，直到用户满意为止

面向概率计算的相关反馈

- **p_t、u_t的重新估计：** 设检索出的结果集合为V(可以把V看成全部的相关文档结合)，其中集合V_t包含词项t，则可以如下进一步估算p_t、u_t：

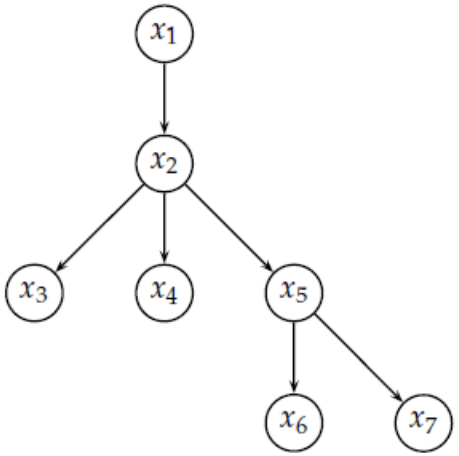
$$p_t = \frac{|V_t| + \frac{1}{2}}{|V| + 1} \qquad u_t = \frac{df_t - |V_t| + \frac{1}{2}}{N - |V| + 1}$$

BIM模型存在问题分析

- BIM概率模型的思想很好，理论基础好，但性能并不理想
- 原因: 其建立在若干不太合理的假设之上
  - 文档、查询及相关性的布尔表示
  - 词项之间具有独立性
  - 查询中不出现的词项不会影响最后的结果
  - 不同文档的相关性之间是互相独立的

词项独立性假设去除

- 该假设不合实际，例如：
  - Hong Kong之间存在很强的相关性
  - {New, York, England, City, Stock, Exchange, University} 之间存在复杂的依赖关系
- 词项之间的树型依赖



Okapi BM25: 一个非二值模型

- **基本思想：** 在BIM模型中增加对词项频率和文档长度的考虑，以提高检索效果
- 考虑词项在文档中的tf权重，有：

$$RSV_d = \sum_{t \in q} \log\left[\frac{N}{\mathrm{df}_t}\right] \cdot \frac{(k_1+1)\mathrm{tf}_{td}}{k_1((1-b)+b\times(L_d/L_{\mathrm{ave}}))+\mathrm{tf}_{td}}$$

  - tf_{td}：词项t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比重的调节参数
  - b：用于控制文档长度比重的调节参数

Okapi BM25: 一个非二值模型

- 如果查询比较长，则加入查询的tf

$$RSV_d = \sum_{t \in q} \left[\log \frac{N}{\mathrm{df}_t} \right] \cdot \frac{(k_1+1)\mathrm{tf}_{td}}{k_1((1-b)+b \times (L_d/L_{\mathrm{ave}}))+\mathrm{tf}_{td}} \cdot \frac{(k_3+1)\mathrm{tf}_{tq}}{k_3+\mathrm{tf}_{tq}}$$

  - tf_{tq}：词项 t 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
  - 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

Okapi BM25: 一个非二值模型

- 如果存在相关性判断结果，则有：

$$RSV_d = \sum_{t \in q} \log \left[\left[\frac{(|VR_t| + \frac{1}{2})/(|VNR_t| + \frac{1}{2})}{(\mathrm{df}_t - |VR_t| + \frac{1}{2})/(N - \mathrm{df}_t - |VR| + |VR_t| + \frac{1}{2})} \right] \right.$$

$$\left. \times \frac{(k_1+1)\mathrm{tf}_{td}}{k_1((1-b) + b(L_d/L_{ave})) + \mathrm{tf}_{td}} \times \frac{(k_3+1)\mathrm{tf}_{tq}}{k_3 + \mathrm{tf}_{tq}} \right]$$

Okapi BM25: 一个非二值模型

- 被广泛使用，获得成功
- 详细内容请参考：
- S.E Roberson and S. Walker, Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval, SIGIR’94
- S.E Roberston, S. Walker, S. Jones, Okapi at TREC-3, in Proceedings of TREC-3

基于语言建模的IR模型

- 统计语言模型：
  - 是“生成”一段文本的概率机制；
  - 对于一个文档片段$d=w_1w_2...w_n$，统计语言模型是指概率$P(w_1w_2...w_n)$求解；
  - 一元模型：$P(w_1w_2w_3w_4)=P(w_1)P(w_2)P(w_3)P(w_4)$
  - 二元模型：$P(w_1w_2w_3w_4)=P(w_1)P(w_2|w_1)P(w_3|w_2)P(w_4|w_3)$
- 基本思想：给定一个查询，对文档通过它们生成该查询的可能性的大小进行排序

基于语言建模的IR模型

- **方法步骤**（查询似然模型）：
  - 对每篇文档d，推导其语言模型M_d，在一元模型下，即计算所有词项 w 的概率$P(w|M_d)$;
  - 计算查询在每个文档d_i的语言模型M_{di}下的生成概率$P(q|M_{di})$;
  - 按照计算出的概率对所有文档进行排序
- 查询生成概率的估计：$\hat{P}(q|M_d) = \prod_{t \in q} \hat{P}_{\text{mle}}(t|M_d) = \prod_{t \in q} \frac{\text{tf}_{t,d}}{L_d}$

参考资料

- 《现代信息检索》第11-12章
- http://ifnlp.org/ir

课后作业

- 见课程网页:

http://10.76.3.31