信息检索导论

An Introduction to Information Retrieval

第11讲 基于语言建模的IR模型

Language Models for IR

授课人：林政

中国科学院信息工程研究所/国科大网络空间安全学院

*改编自”An introduction to Information retrieval”网上公开的课件，地址 http://nlp.stanford.edu/IR-book/

提纲

1. 上一讲回顾
2. 语言模型
3. 基于统计建模的IR模型
4. SLMIR模型讨论

提纲

1. 上一讲回顾
2. 语言模型
3. 基于统计建模的IR模型
4. SLMIR模型讨论

概率检索模型

- 概率检索模型是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$Q=\{q_1,q_2,\ldots\}$，文档$D=\{d_1,d_2,\ldots\}$，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，如Logistic Regression(回归)模型及最经典的二值独立概率模型BIM、BM25模型等。
- 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

- BM25这一经典概率模型已经在商业搜索引擎的网页排序中广泛应用。
- 基本思想：给定一个用户查询，若搜索系统能在搜索结果排序时按照文档和用户查询的相关性由高到低排序，那么这个搜索系统的准确性是最优的。
- 实际实现
  - 根据用户的查询将文档集合划分为两个集合：相关文档子集和不相关文档子集。
  - 将相关性衡量转换为分类问题，对某个文档D来说，若其属于相关文档子集的概率大于属于不相关文档的概率，就认为它与查询相关。

几种概率检索模型

- 基于Logistic回归的检索模型
- 经典的二值独立概率模型BIM
- 经典的BM25模型 (BestMatch25)

Logistic 回归IR模型

- 基本思想：为了求Q和D相关的概率$P(R=1|Q,D)$，通过定义多个特征函数$f_i(Q,D)$，认为$P(R=1|Q,D)$是这些函数的组合。
- Cooper等人提出一种做法*：定义$\log(P/(1-P))$为多个特征函数的线性组合。则P是一个Logistic函数，即：

$$\log\frac{P}{1-P}=\beta_0+\sum_i\beta_i f_i(Q,D)$$

$$P=\frac{1}{1+e^{-\beta_0-\sum_i\beta_i f_i(Q,D)}}$$

*William S. Cooper , Fredric C. Gey , Daniel P. Dabney, Probabilistic retrieval based on staged logistic regression, Proceedings of ACM SIGIR'92, p.198-210, June 21-24, 1992, Copenhagen, Denmark

BIM模型(续)

- 对每个Q定义排序(Ranking)函数RSV(Q,D)：

$$\log\frac{P(R=1|D)}{P(R=0|D)}=\log\frac{P(D|R=1)P(R=1)/P(D)}{P(D|R=0)P(R=0)/P(D)}$$
$$\propto\log\frac{P(D|R=1)}{P(D|R=0)}$$

将判断文档和查询是否相关的问题转化成一个生成问题，其中，$P(D|R=1)$、$P(D|R=0)$分别表示在相关和不相关情况下生成D的概率。Ranking函数显然是随着$P(R=1|D)$的增长而增长。

BIM模型(续)

假设D的生成过程符合多元贝努利分布 $\wedge_{t_i \in D} t_i \wedge_{t_j \notin D} t_j$

$$P(D \mid R=1) = \prod_{t_i \in D} P(t_i \mid R=1) \prod_{t_i \notin D} P(\bar{t}_i \mid R=1)$$

$$= \prod_{t_i} p_i^{e_i} (1-p_i)^{1-e_i}, if \ \ t_i \in D \ \ then \ \ e_i = 1, else \ e_i = 0$$

$$P(D \mid R=0) = \prod_{t_i \in D} P(t_i \mid R=0) \prod_{t_i \notin D} P(\bar{t}_i \mid R=0)$$

$$= \prod_{t_i} q_i^{e_i} (1-q_i)^{1-e_i}, if \ \ t_i \in D \ \ then \ \ e_i = 1, else \ e_i = 0$$

$$p_i = P(t_i \mid R=1)$$

$$q_i = P(t_i \mid R=0)$$

p_i q_i参数的计算

理想情况下，可以将整个文档集合根据是否和查询相关、是否包含t_i分成如下四个子集合，每个集合的大小已知。

	相关 R_i (100)	不相关 N-R_i (400)
包含t_i n_i (200)	r_i (35)	n_i- r_i (165)
不包含t_i N-n_i (300)	R_i-r_i (65)	N-R_i-n_i+r_i (235)

其中，N、n_i分别是总文档以及包含t_i的文档数目。R_i、r_i分别是相关文档及相关文档中包含t_i的文档数目。括号中列举的数值是给出的一个总文档数目为500的计算例子。则：

$$p_i = \frac{r_i}{R_i} = \frac{35}{100} = 0.35$$

$$q_i = \frac{n_i - r_i}{N - R_i} = \frac{165}{400} = 0.413$$

引入平滑因子

$$p_i = \frac{r_i + 0.5}{R_i + 1}$$

$$q_i = \frac{n_i - r_i + 0.5}{N - R_i + 1}$$

p_i q_i参数的计算(续)

- 由于真实情况下，对于每个查询，无法事先得到相关文档集和不相关文档集，所以无法使用理想情况下的公式计算，因此必须进行估计
- 有多种估计方法
  - 初始检索：第一次检索之前的估计
  - 基于检索结果：根据上次检索的结果进行估计

p_i q_i参数的计算(续)

- 初始情况：检索初始并没有相关和不相关文档集合，此时可以进行假设：p_i是常数，q_i近似等于term i在所有文档集合中的分布(假定相关文档很少，R_i=r_i=0)

$$p_i = 0.5$$

$$q_i = \frac{n_i}{N}$$

$$\sum_{t_i \in q \cap d} \log \frac{p_i /(1-p_i)}{q_i /(1-q_i)} = \sum \log \frac{N-n_i}{n_i}$$

因此，BIM在初始假设情况下，其检索公式实际上相当于对所有同时出现在q和d中的term的IDF的求和

Okapi BM25: 一个非二值模型

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

$$W_i^{IDF}=\log\frac{N-n_i+0.5}{n_i+0.5}$$

本讲内容

- (统计)语言模型

- 基于统计语言建模的IR模型
  - (基本)查询似然模型
  - 一些扩展的模型

提纲

统计语言模型(Statistical Language Modeling，SLM)

- SLM广泛使用于语音识别和统计机器翻译领域，利用概率统计理论研究语言。
  - 规则方法：词、句、篇章的生成比如满足某些规则，不满足该规则就不应存在。
  - 统计方法：任何语言片断都有存在的可能，只是可能性大小不同
- 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，根据Bayes公式，有

$$P(w_1w_2...w_n)=P(w_1)P(w_2...w_n\mid w_1)=...=P(w_1)\prod_{i=2}^{n}P(w_i\mid w_{i-1}...w_1)$$

历史(history)

- 无历史，一元模型
- 最近一个历史，二元模型(Bigram)
- 最近N-1个历史，N元模型(N-gram

类比：打扑克中的出牌策略

只根据当前牌出牌，一元模型；
根据上一轮牌出牌，二元模型；

用自己的2张底牌和5张公共牌结合在一起，选出5张牌，不论手中的牌使用几张，凑成最大的成牌，跟其他玩家比大小。

不同模型的例子

- 一元模型(unigram)：$P(w_1w_2w_3w_4) = P(w_1)P(w_2)P(w_3)P(w_4)$
- 二元模型(bigram)：$P(w_1w_2w_3w_4) = P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_2)P(w_4 \mid w_3)$
  - 一阶马尔科夫链
- 三元模型(trigram)：$P(w_1w_2w_3w_4) = P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_1w_2)P(w_4 \mid w_2w_3)$
  - 二阶马尔科夫链
- n-gram模型也称为n-1阶马尔科夫模型，它有一个有限历史假设：当前词的出现概率仅仅与前面n-1个词相关

- 对于N元模型(N-gram)，N越大，则模型越复杂，估计的参数(即估计的概率)也越多。当然，当数据量足够大的情况下，模型阶数越高越对片段概率的计算也越准确。

课堂思考

- 设词典大小为M，试估计N元模型要估计的参数(概率)空间大小。

$$P(w_1w_2w_3w_4)=P(w_1)P(w_2)P(w_3)P(w_4)$$

$$P(w_1w_2w_3w_4)=P(w_1)P(w_2\mid w_1)P(w_3\mid w_2)P(w_4\mid w_3)$$

$$P(w_1w_2w_3w_4)=P(w_1)P(w_2\mid w_1)P(w_3\mid w_1w_2)P(w_4\mid w_2w_3)$$

- 估计的参数数目为：$M+M^2+\ldots+M^N=(M^{N+1}-M)/(M-1)$

- 假定M=1000，N=4，则需要估计约10^{12}=1万亿个参数，参数空间巨大！
- 最常用的是bigram，其次是unigram和trigram，n取≥4的情况较少。

SLM的一个应用例子

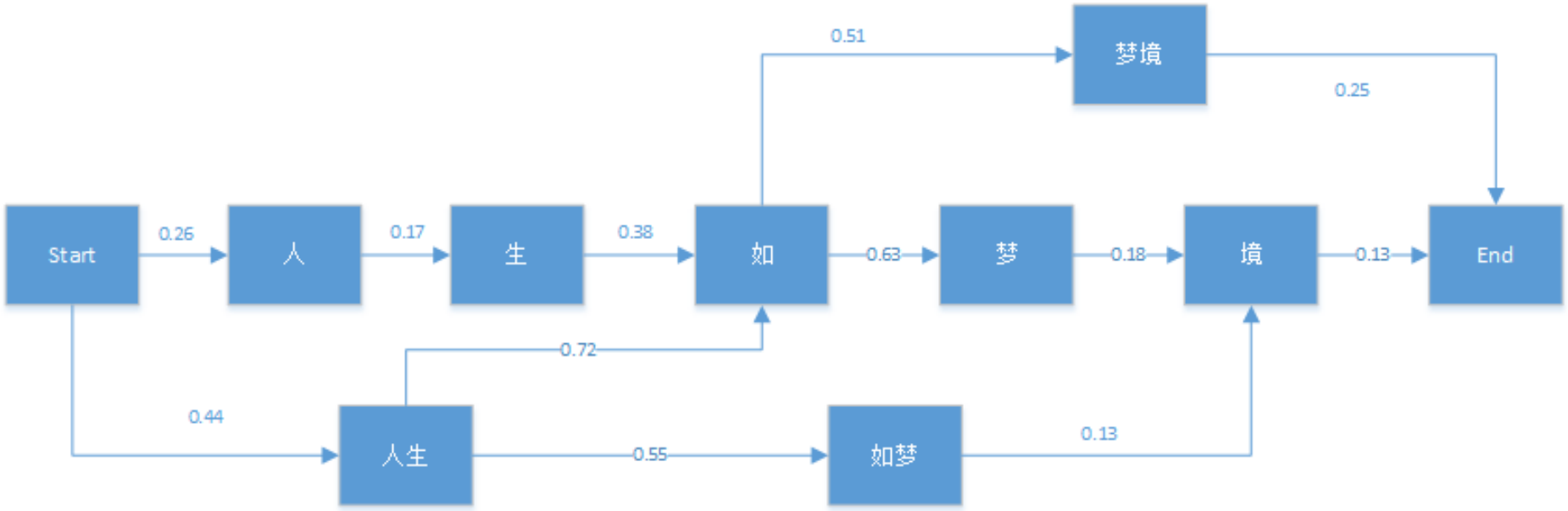
- 拼音输入法(以下例子中将字看成语言单位)：
  - 输入zhong guo ke xue yuan，到底是：种过科雪园？重果可薛原？还是 中国科学院？……
  - 一种利用SLM的解决思路：计算 P(种过科雪园) P(重果可薛原) P(中国科学院)，看谁大！
- 一元模型(Unigram)*：
  - P(种过科雪园)=P(种) P(过) P(科) P(雪) P(园)
  - P(重果可薛原)=P(重) P(果) P(可) P(薛) P(原)
  - P(中国科学院)=P(中) P(国) P(科) P(学) P(院)
  - 训练：在训练语料库中估计以上各 P(X)的值
- 课堂思考：一元模型存在的问题？

SLM的一个应用例子

- 二元模型(Bigram)： P(中国科学院)=P(中)P(国|中)P(科|国)P(学|科)P(院|学)，等价于一阶马尔科夫链(Markov Chain)
- 三元模型(Trigram)：P(中国科学院)= P(中)P(国|中)P(科|中国)P(学|国科) P(院|科学)
- 根据语料，估计所使用模型的参数，然后在搜索空间中搜索概率最大的语言片段。

SLM的一个应用例子（分词）

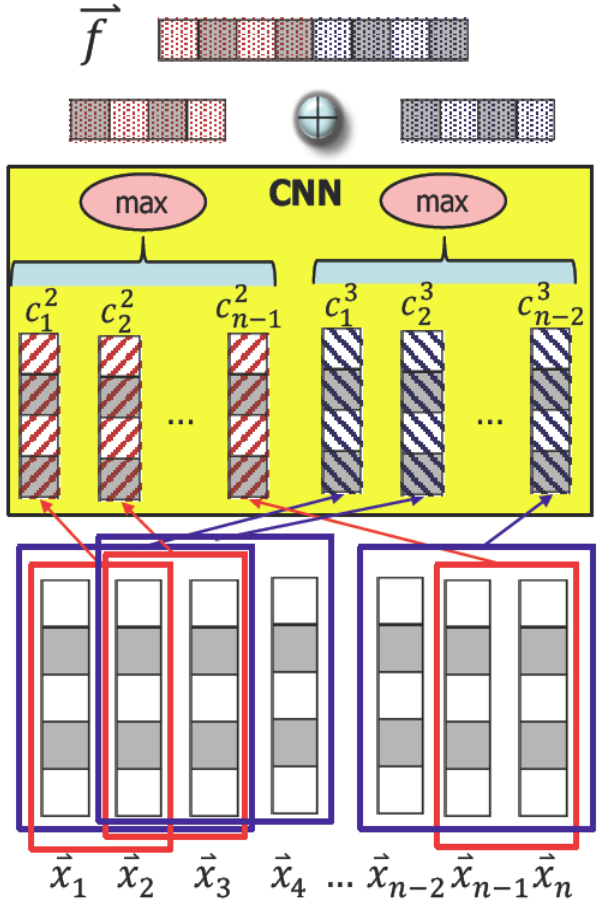
- **2-gram模型的解码算法**



用的解码算法是viterbi(最短路径)算法，它采用动态规划的原理，能够很快地确定最合适的路径。

CNN如何对词序建模

- CNN (Kim, 2014)
  - Feature combinations
  - Single CNN layer
  - Varied-window-size convolutional filters
  - Multichannel (1 static+ 1 nonstatic)



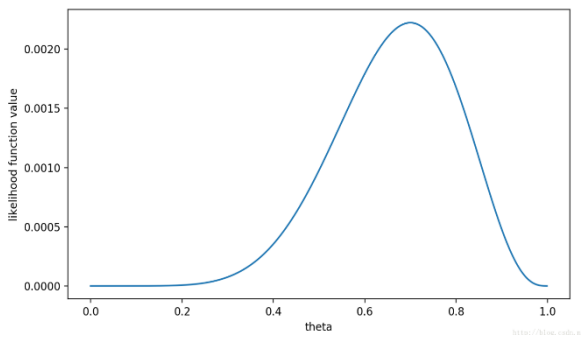
SLM的参数估计

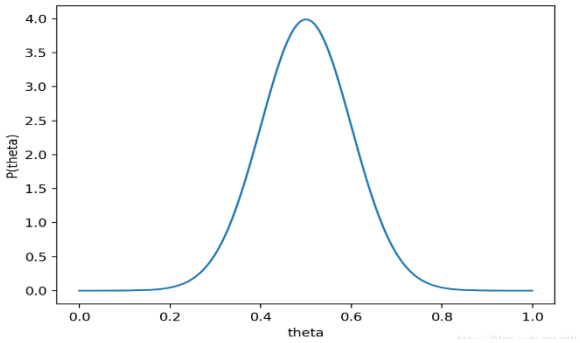
- 一般采用最大似然估计（Maximum Likelihood Estimation，MLE）的方法对模型的参数进行估计
- 理论上说，在数据充足的情况下，利用更多的历史(高阶)的模型更准确，但是总计算量也越大
- 数据规模总是有限的，即用于训练模型参数的语料存在稀疏性(Data Sparseness，即某参数在训练语料中没有出现)问题。如二元模型中，在训练语料中恰巧没有出现“国科”组合。

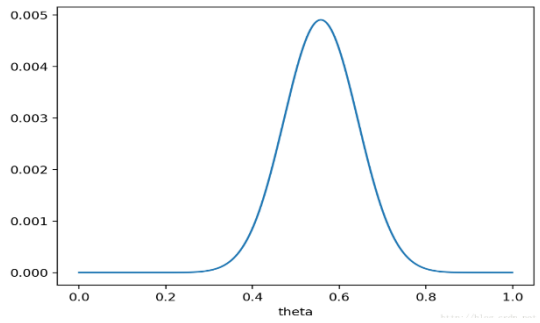
最大似然估计

- 最大似然估计的核心思想是：找到参数 θ 的一个估计值，使得当前样本出现的可能性最大。
- 似然（likelihood）与概率（probability）的区别：
  - 概率是在特定环境下某件事情发生的可能性，比如抛硬币
  - 似然是在确定的结果下去推测产生这个结果的可能环境（参数）

举个例子，如果抛10个硬币，实验结果（反正正正正反正正正反）的似然函数是多少呢？最大似然估计认为正面向上的概率是0.7。一些人可能会说，硬币一般都是均匀的啊，就算正面向上7次我也不信。这里就包含了贝叶斯学派的思想了——要考虑先验概率。 为此，引入了最大后验概率估计。







数据稀疏性

- 数据稀疏性导致零概率问题，上述稀疏性情况下，如果直接计算，那么P(中国科学院)=0，但是在训练集上不出现的事件并不代表在新的语料上不出现。

- SLM的一个重要工作就是进行平滑(Smoothing)：重新分配概率，即使没出现的事件也会赋予一个概率。

两种简单的平滑策略

- 加法平滑

基本思想是将每个n元对的出现次数加上一个常数δ（0<δ≤1）

$$P(W_i|W_{i-n+1},\dots,W_{i-1}) = (C(W_{i-n+1},\dots,W_{i-1},W_i)+\delta)/(C(W_{i-n+1},\dots,W_{i-1})+N\delta)$$

- 线性插值平滑

利用低元n-gram模型对高元n-gram模型进行线性插值。因为在没有足够的数据对高元n-gram模型进行概率估计时，低元n-gram模型通常可以提供有用的信息。

$$P_{interp}(W_i|W_{i-n+1},\dots,W_{i-1}) =$$

$$\lambda_n \cdot P_{MLE}(W_i|W_{i-n+1},\dots,W_{i-1}) + (1-\lambda_n)\cdot P_{interp}(W_i|W_{i-n+2},\dots,W_{i-1})$$

思考题

除了输入法，统计语言模型还有哪些应用？

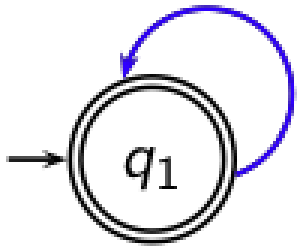
另一个角度看语言模型

我们可以把一个有穷状态自动机(finite state automaton)看成一个确定性语言模型(deterministic language)

上述模型可以生成片段“I wish I wish I wish I wish . . .”但是不能生成片段“wish I wish”

如果上述自动机是带有概率的，则是概率语言模型(probabilistic LM，也称统计语言模型SLM)

一个概率语言模型的例子



w	$P(w\|q_1)$	w	$P(w\|q_1)$
STOP	0.2	toad	0.01
the	0.2	said	0.03
a	0.1	likes	0.02
frog	0.01	that	0.04
		...	...

单状态概率有穷状态自动机—一元语言模型—状态发射概率分布如右表。其中STOP不是词，而是表示自动机结束的一个标识符。这样，概率

P(frog said that toad likes frog STOP) = $0.01 \cdot 0.03 \cdot 0.04 \cdot 0.01 \cdot 0.02 \cdot 0.01 \cdot 0.2 = 0.0000000000048$

加法平滑

基本思想是为避免零概率问题，将每个n元对得出现次数加上一个常数δ（0<δ≤1）：

思考题

SLM与IR有什么关系？

基于语言模型的IR

- **基本思想**

区别于其他大多数检索模型从查询到文档（即给定用户查询，如何找出相关的文档），语言模型**由文档到查询**，即为每个文档建立不同的语言模型，**判断由文档对应的语言模型抽样出用户查询的可能性**有多大，然后**按照概率由高到低排序**，作为搜索结果。（读3遍）

- **生成查询概率**

为每个文档建立一个语言模型，语言模型代表了单词（或单词序列）在文档中的分布情况。针对查询中的单词，每个单词都有一个抽取概率，将这些单词的抽取概率相乘就是文档生成查询的概率。

两个不同的语言模型

language model of d_1

w	$P(w\|.)$	w	$P(w\|.)$
STOP	.2	toad	.01
the	.2	said	.03
a	.1	likes	.02
frog	.01	that	.04
		...	...

language model of d_2

w	$P(w\|.)$	w	$P(w\|.)$
STOP	.2	toad	.02
the	.15	said	.03
a	.08	likes	.02
frog	.01	that	.05
		...	...

string = frog said that toad likes frog STOP

则 $P(\text{string}|M_{d1}) = 0.01 \cdot 0.03 \cdot 0.04 \cdot 0.01 \cdot 0.02 \cdot 0.01 \cdot 0.2 = 0.000000000048 = 4.8 \cdot 10^{-12}$

$P(\text{string}|M_{d2}) = 0.01 \cdot 0.03 \cdot 0.05 \cdot 0.02 \cdot 0.02 \cdot 0.01 \cdot 0.2 = 0.000000000120 = 12 \cdot 10^{-12}$

$P(\text{string}|M_{d1}) < P(\text{string}|M_{d2})$

因此, 相对于d_1，文档 d_2 与字符串 “frog said that toad likes frog STOP” 更相关

统计语言建模IR模型(SLMIR)

- 马萨诸塞大学(University of Massachusetts, UMass)大学Ponte、Croft等人于1998年提出。随后又发展了出了一系列基于SLM的模型。代表系统Lemur。
  - **查询似然模型**：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
  - **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
- 本讲义主要介绍查询似然模型

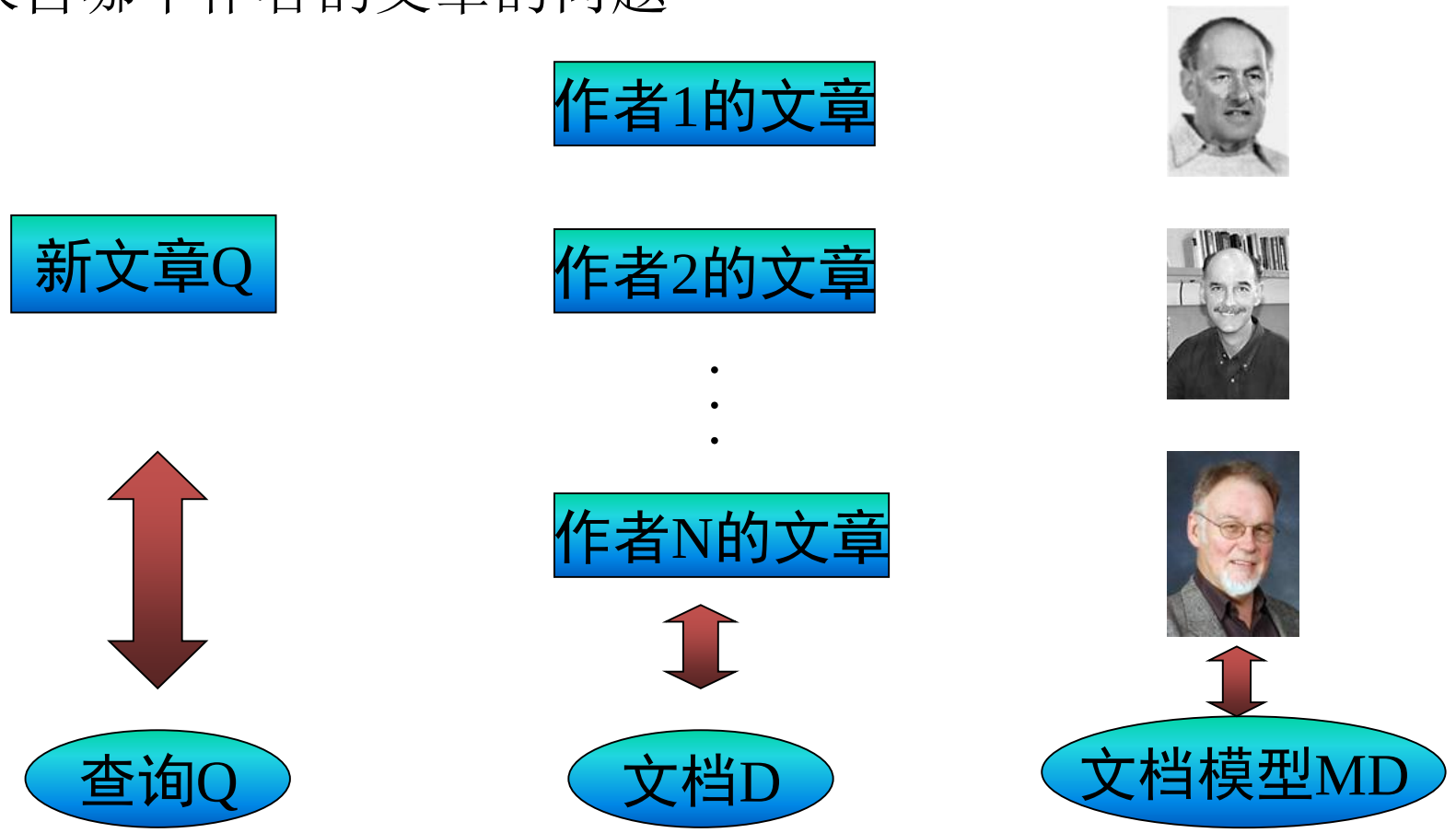
从一个问题开始

- 课堂思考题：设有N个作者，每人都有一篇文章，对于不在上述N篇文章中的一篇新文档Q，请问最有可能是哪个作者写的？

- 一个解决思路：根据每个作者写的文章，总结出每个作者的写作风格，然后再根据写作风格来判断Q与谁的风格最近。

和检索的类比

把查询看成新文章，判断查询和哪篇文档更相关，看成新文章是来自哪个作者的文章的问题



总体分布&抽样

- 文档的模型(风格)实际上是某种总体分布
- 文档和查询都是该总体分布下的一个抽样样本实例
- 根据文档，估计文档的模型，即求出该总体分布(一般假设某种总体分布，比如多项式分布，然后求出其参数)
- 然后计算该总体分布下抽样出查询的概率

查询似然模型(Query Likelihood Model)

- 查询Q的检索排序函数定义如下（将文档按照其与查询相关概率$P(D|Q)$排序）：

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

$$P(D|Q)=\frac{P(Q|D)P(D)}{P(Q)}\propto P(Q|D)P(D)$$

- 文档D的先验分布$P(D)$假定为均匀分布，则这一项可以去掉。

- $P(D)$也可以采用某个与查询无关的量，如PageRank。QLM中不考虑这一项。

查询似然模型QLM

- QLM计算公式

$$RSV(Q,D) = P(Q|D) = P(Q|M_D)$$ 已知文档D，抽样出查询Q的概率

$$= P(q_1 q_2 ... q_m | M_D)$$ M是文档的语言模型

$$= P(q_1|M_D)P(q_2|M_D)...P(q_m|M_D)$$

$$= \prod_{w \in Q} P(w|M_D)^{c(w,Q)}$$

- 于是检索问题转化为估计文档D的一元语言模型M_D，也即求所有词项 w的概率$P(w|M_D)$

QLM概念理解

- QLM中$P(Q|D)$本质上是$P(Q|M_D)$，不能把$P(Q|D)$称为文档D生成查询Q的概率

（注意：有时M_D简写成D，本质上M_D不是D）

- 文档D和Q都是某个总体分布（M_D）的样本(实例)，样本(实例)是不会产生样本(实例)的
  - 样本是不会再生成其他东西的，样本只能用来推断总体的某些信息，比如总体的某些未知参数(通过一篇文章来推断作者的风格)
- 同样，不能把$P(w|M_D)$或P$(w|D)$理解为“w在文档D中的概率”

QLM求解步骤

- 第一步：根据文档D(样本)，估计文档模型M_D(总体)，在一元模型下，即计算所有词项 w的概率 $P(w|M_D)$
- 第二步：计算在模型M_D下生成查询Q的似然(即概率)
- 第三步：按照得分对所有文档排序

M_D的估计

- 问题：已知样本D，求其模型M_D的参数$P(w|M_D)$。
- 对于该参数估计问题，可以采用最大似然估计(Maximum Likelihood Estimation，MLE)。
- MLE：使得观察样本出现概率(似然)最大的估计。
  - 一射击世界冠军和一菜鸟打靶，其中一人放一枪得到10环，请问是谁打的？显然世界冠军打的可能性大，也就是说这是使得10环这个事件出现概率最大的估计。

M_D的MLE估计

- 设词项词典的大小为L，则模型M_D的参数可以记为：

$$\vec{\theta}_D = (\theta_1, \theta_2, ..., \theta_L)$$
$$= (P(w_1 | M_D), P(w_2 | M_D), ..., P(w_L | M_D))$$

- MLE估计：

$$\vec{\theta}_D^{\,*} = \arg\max_{\vec{\theta}_D} P(D | \vec{\theta}_D)$$

- 关键是如何求 $P(D | \vec{\theta}_D)$，也就是说假设这些参数已知的情况下，如何求上述概率。

总体分布M_D的假设

- 两种文本生成模型：
  - 多元贝努利模型(概率模型BIM中使用)：D是抛L个(L是词项词典的大小)不同的硬币生成的，每个硬币对应一个词项，统计所有向上和向下的硬币对应的词项便生成文本D。多元贝努利模型中的参数是每个硬币朝上的概率，共有 L个。
  - 多项式模型：D是抛1个L面的骰子抛$|D|$次生成的，将每次朝上的那面对应的词项集合起来便生成文本D。
- QLM在1998年提出时采用的是多元贝努利模型，后来才有人用多项式模型并发现多项式模型通常优于贝努利模型。所以后来介绍QLM时大都用多项式模型。

文本生成的多项式模型

- 有一个L个面的不规则骰子，在第i个面上写着w_i，文档$D=d_1d_2...d_n$可以认为是抛n次骰子得到的

- 检索过程就是根据观察样本D的分布估计Q的生成概率，即在已知抛n次的结果为文档D的条件下，抛m次的结果为查询Q的概率$P(Q|M_D)$=?

- $D = (c(w_1,D),c(w_2,D),\ldots,c(w_L,D))$， $c(w_i,D)$是文档D中w_i的出现次数

- D = 我 喜欢 基于 统计 语言 模型 的 信息 检索 模型
- D = (<我,1>,<喜欢,1>,<基于,1>,<统计,1>,<语言,1>, <模型,2>,<的,1>,<信息,1>,<检索,1>)

多项随机试验

- 多项(Multinomial)随机试验是二项随机试验(贝努利试验)的扩展，一篇文档D可以看成多项随机试验的结果
  - 多项随机试验由n次相互独立的子试验组成
  - 每个子试验含有L个互斥且完备的可能结果$w_1, w_2, \ldots, w_L$。
  - 每个子试验中w_i发生的概率不变，记为 θ_i

多项式分布考虑词项的多次出现

多项式分布不考虑词项的不出现

多项式分布同样不考虑词项的出现位置和次序

多项随机试验(续)

- 设随机变量$X_1,X_2,\ldots,X_L$用于记录n次子试验中$w_1,w_2,\ldots,w_L$的发生次数，实际记录值为$x_1,x_2,\ldots,x_L$，$x_1+x_2+\ldots+x_L=n$，如果某个w_i不出现，则对应的$x_i=0$
- 则该多项随机试验中$w_1,w_2,\ldots,w_L$发生次数的联合分布是一个多项式分布：

$$f(x_1,x_2,\ldots,x_L)=P(X_1=x_1,X_2=x_2,\ldots,X_L=x_L)=C_n^{x_1}\theta_1^{x_1}C_{n-x_1}^{x_2}\theta_2^{x_2}\ldots C_{n-x_1-\ldots-x_{L-1}}^{x_L}\theta_L^{x_L}$$

$$=\frac{n!}{x_1!(n-x_1)!}\times\frac{(n-x_1)!}{x_2!(n-x_1-x_2)!}\times\ldots\times\frac{(n-x_1-\ldots-x_{L-1})!}{x_L!(n-x_1-\ldots-x_L)!}\prod_{i=1}^{L}\theta_i^{x_i}=n!\prod_{i=1}^{L}\frac{\theta_i^{x_i}}{x_i!}$$

M_D的参数求解

- 求解 $\vec{\theta}_D^{\,*} = \arg\max_{\vec{\theta}_D} P(D|\vec{\theta}_D) = \arg\max n! \prod_{i=1}^{L} \frac{\theta_i^{c(w_i,D)}}{c(w_i,D)!}$

$$\sum_{i=1}^{L} \theta_i = 1$$

- 条件极值问题，采用拉格朗日法求解，得到拉格朗日函数：

$$L(\lambda, \vec{\theta}_D) = n! \prod_{i=1}^{L} \frac{\theta_i^{c(w_i,D)}}{c(w_i,D)!} + \lambda(1 - \sum_{i=1}^{L} \theta_i)$$

- 对每个θ_i求偏导，令其为0，解得：

$$\theta_i^* = P_{ML}(w_i|M_D) = \frac{c(w_i,D)}{\sum_{j=1}^{L} c(w_j,D)} = \frac{c(w_i,D)}{|D|}$$

一个MLE估计的例子

- D = (<我,1>,<喜欢,1>,<基于,1>,<统计,1>,<语言,1>, <模型,2>,<的,1>,<信息,1>,<检索,1>)

- 采用MLE估计有：

  $P(我|M_D)=P(喜欢|M_D)=P(基于|M_D)$
  $=P(统计|M_D)=P(语言|M_D)=P(的|M_D)$
  $=P(信息|M_D)=P(检索|M_D)=0.1$

  $P(模型|M_D)=0.2$

  其他词项的概率为0

MLE估计的零概率问题

- 对于任意不属于D的词项，其概率的MLE估计值为0(数据的稀疏性)。然而，样本D中不出现的词，并不代表在新文本中不出现。

- 类比：作者写的一篇文章里面不用某个词，并不代表以后不用这个词

MLE估计零概率的一个例子

- 一个不规则的骰子，分别用1-6之间的6个数字代表每个面。假设连续抛10次后，观测到的结果序列为2132461232(一篇文档)。问：抛3次的结果序列为325(一个查询)的概率是多大？

例子(续)

- 其实就是求：在已知抛10次的观测结果为2132461232的条件下，抛3次的观测结果为325的概率P(325| 2132461232).
- 用p_i表示第i个面朝上的概率，则p_i是一个常数
- 若已知p_i(1<=i<=6)，则显然抛3次的观测结果为325的概率为$p_3p_2p_5$
- 但是每个p_i未知，已知的只是抛10次的观测结果2132461232。

- 需要由观测结果2132461232去估计各个p_i（1<=i<=6）。

例子中的MLE估计

- 抛骰子的例子中：$P(325|\ 2132461232)=?$ ⟶ 估计 p_i ,i=1,2,..,6
- 使用最大似然估计，根据样本2132461232估计p_i
  p_1=2/10, p_2=4/10, p_3=2/10, p_4=1/10, p_5=0/10, p_6=1/10
- 显然，5没有出现在样本中，不能说明该骰子第5面永远不会朝上。
  更严重的是，$P(325|\ 2132461232)=p_3p_2p_5=0$!
- 因此，上述的估计结果需要调整，使所有$p_i>0$。
- 思想：从p_1,p_2,p_3,p_4的估计值中扣出一点点给p_5
- 最简单的方法：Add-One
  p_1=3/16, p_2=5/16, p_3=3/16, p_4=2/16, p_5=1/16, p_6=2/1　　p1,p2,p3,p4,p6各减1/80
- 从上述例子总结出以下几点:
  (1) 因样本的数据稀疏性，最大似然估计(MLE)导致零概率问题
  (2) 必须设法调整MLE使得所有事件的概率都大于0 → 平滑(smoothing)

数据平滑的一般形式

Discounted(折扣后的) Maximum Likelihood

$$p(w|D)=\begin{cases}p_{DML}(w|D) & w\in D\\ \alpha_D p(w|REF) & otherwise\end{cases}$$

Reference Language Model

$$\alpha_D=\frac{1-\sum_{w\in D}p_{DML}(w|D)}{\sum_{w\notin D}p(w|REF)}$$

Collection Language Model

- 在IR中，一般取 $p(w|REF)=p(w|C)=\frac{\sum_D c(w,D)}{\sum_w\sum_D c(w,D)}$
- $p(w|C)$等于w出现的次数除以所有词出现的次数。

几种QLM中常用的平滑方法

- Jelinek-Mercer(JM)，$0 \le \lambda \le 1$ 文档估计和文档集估计的混合

$$p(w|D) = \lambda p_{ML}(w|D) + (1-\lambda) p(w|C)$$

  - 课堂提问，对于$w \in D$，折扣后的$P_{DML}(w|D)$是不是一定小于$P_{ML}(w|D)$？

- Dirichlet Priors(Dir)，$\mu \ge 0$

$$p(w|D) = \frac{c(w,D) + \mu p(w|C)}{|D| + \mu}$$

- Absolute Discounting(Abs)，$0 \le \delta \le 1$，$|D|_u$表示D中不相同的词个数(u=unique)

$$p(w|D) = \frac{\max(c(w,D) - \delta, 0)}{|D|} + \frac{\delta |D|_u}{|D|} p(w|C)$$

QLM的求解过程图示

	w_1	w_2	w_3	…	w_L
D_1	P_{11} √	P_{12}	P_{13} √	…	P_{1L}
D_2	P_{21} √	P_{22} √	P_{23}	…	P_{2L}
…	…	…	…	…	…
D_N	P_{N1}	P_{N2} √	P_{N3} √	…	P_{NL}

先计算P_{ML}，然后采用平滑公式计算P_{ML}➔P_{DML}

文档排名函数的转换

- $P(Q|D)=\prod_{w\in Q} p(w|D)^{c(w,Q)}$ $\quad p(w|D)=\begin{cases} p_s(w|D) & w\in D \\ p_u(w|D) & otherwise \end{cases}$

$$\log P(Q|D)=\sum_{w\in Q} c(w,Q)\log p(w|D)=\sum_{w} c(w,Q)\log p(w|D)$$

w不属于Q时，c(w,Q)=0

$$=\sum_{w\in D} c(w,Q)\log p_s(w|D)+\sum_{w\notin D} c(w,Q)\log p_u(w|D)$$

$$=\sum_{w\in D} c(w,Q)\log p_s(w|D)+\sum_{w} c(w,Q)\log p_u(w|D)-\sum_{w\in D} c(w,Q)\log p_u(w|D)$$

$$=\sum_{w\in D\cap Q} c(w,Q)\log \frac{p_s(w|D)}{p_u(w|D)}+\sum_{w\in Q} c(w,Q)\log p_u(w|D)$$

w不属于Q时，c(w,Q)=0

- 将 $p_s(w|D)=p_{DML}(w|D), p_u(w|D)=\alpha_D p(w|C)$ 代入

$$\log P(Q|D)=\sum_{w\in Q\cap D} c(w,Q)\log \frac{p_s(w|D)}{\alpha_D p(w|C)}+|Q|\log\alpha_D+\sum_{w\in Q} c(w,Q)\log p(w|C)$$

查询中w的总次数

不影响排名

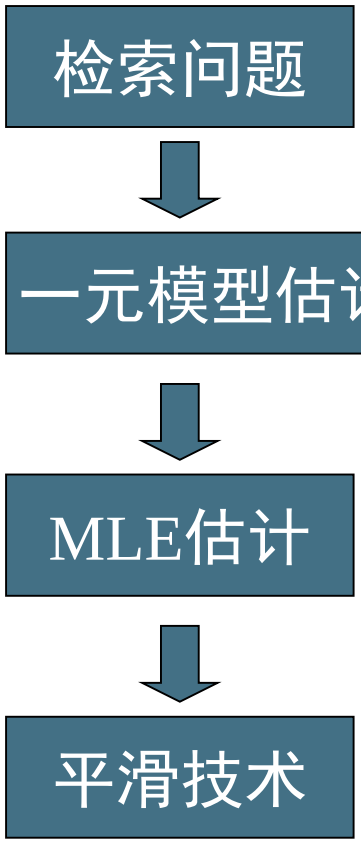
- 最终排名函数：$RSV(Q,D)=\sum_{w\in Q\cap D} c(w,Q)\log \frac{p_{DML}(w|D)}{\alpha_D p(w|C)}+|Q|\log\alpha_D$

TF

D长度有关

DF

QLM模型小结



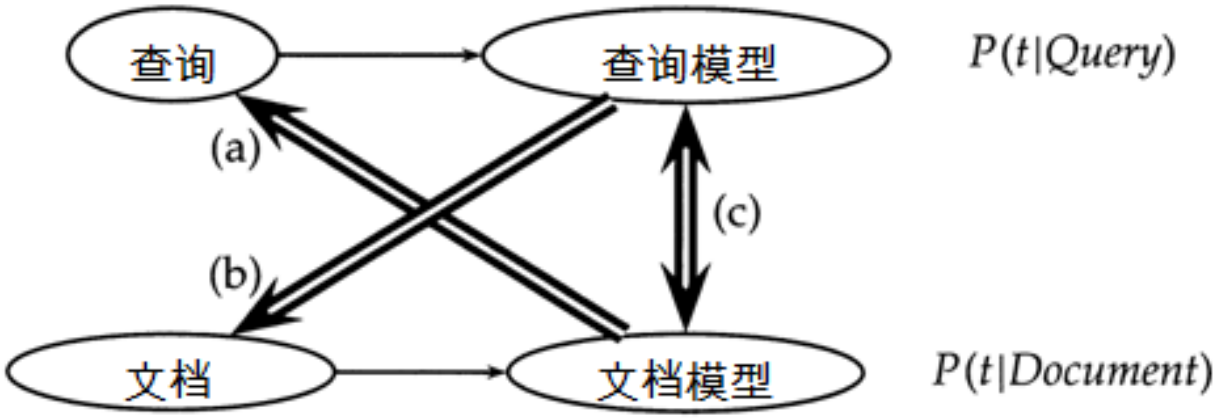
①把相关度看成是每篇文档对应的语言下生成该查询的可能性

②求文档背后的语言模型。在一元模型下，即求所有词w的概率$P(w|M_D)$

多项式分布假设

③D是抛1个L面的骰子抛$|D|$次生成的，将每次朝上的那面对应的词项集合起来便生成文本D。

基本SLMIR模型的扩展



IR 中使用统计语言建模的 3 种方式：(a) 查询似然；(b) 文档似然；(c) 模型比较

- 查询似然类：文档建模，计算查询的似然，例子--基本QLM模型、翻译模型等
- 文档似然类：查询建模，计算文档的似然，例子--BIM模型、相关性模型(Relevance模型)等
- 模型比较类：文档建模、查询建模，计算两个模型的距离，KL距离模型

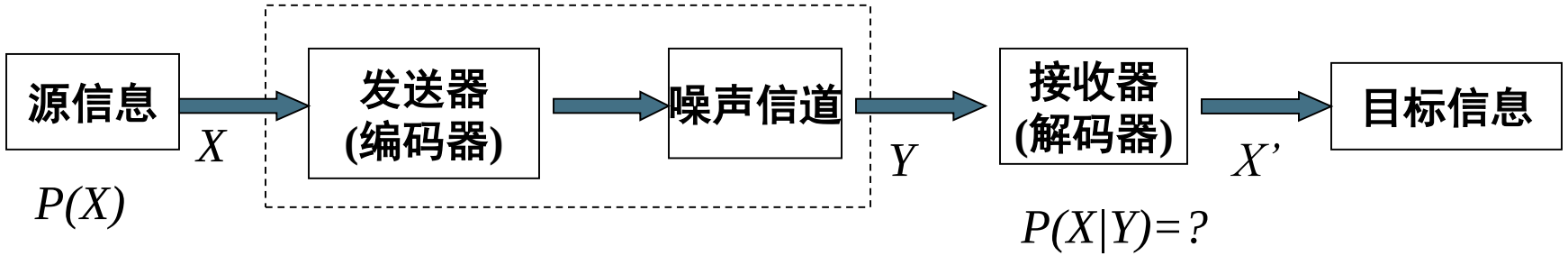
其它SLMIR模型

- 翻译模型(Translation Model)

- KL距离模型(KL Divergence Model)

香农(Shannon)信道

语音识别:	Y=语音信号	X=词序列
机器翻译 (中->英) :	Y=中文句子	X=英文句子
OCR :	Y= 错误的词语	X=纠错后的词
文档摘要:	Y=文档	X=摘要
信息检索:	Y=查询	X=文档



$$\hat{X} = \arg\max_{X} p(X \mid Y) = \arg\max_{X} p(Y \mid X)p(X)$$

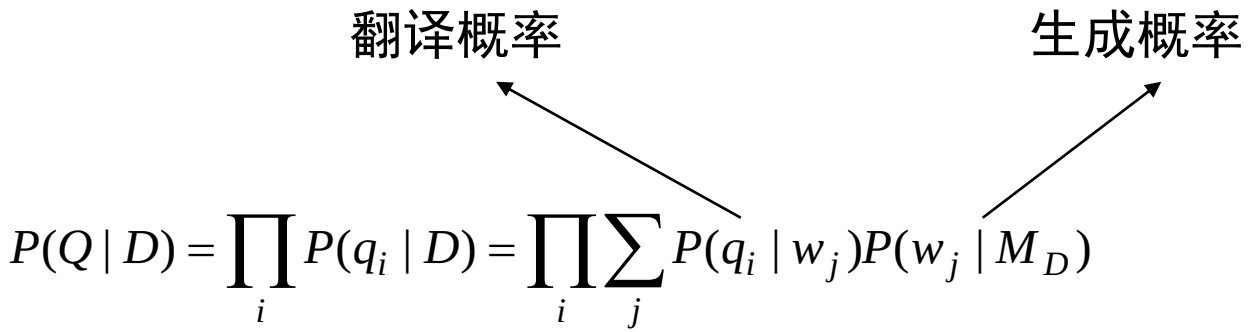
文档经过噪声信道变成查询，
查询经过解码还原成文档

则由文档还原成该查询
的概率可以视为相关度

当X是文本时，$p(X)$就是一个语言模型

基于翻译模型的IR模型

- 基本的QLM模型不能解决词语失配(word mismatch)问题，即查询中的用词和文档中的用词不一致，如：电脑 vs. 计算机
- 假设Q通过一个有噪声的香农信道变成D，从D估计原始的Q



- 翻译概率$P(q_i|w_j)$在计算时可以将词项之间的关系融入。
  - 基于词典来计算(人工或者自动构造的同义词/近义词/翻译词典)
  - 基于语料库来计算(标题、摘要 vs. 文本；文档锚文本 vs. 文档)

KL距离(相对熵)模型

$$Score(Q,D)=\log\frac{P(Q|M_D)}{P(Q|M_C)}$$

$$Score(Q,D)=\sum_{q_i\in Q} tf(q_i,Q)*\log\frac{P(q_i|M_D)}{P(q_i|M_C)}$$

词频与概率成正比

$$\propto \sum_{q_i\in Q} P(q_i|M_Q)*\log\frac{P(q_i|M_D)}{P(q_i|M_C)}$$

分子分母同乘$P(q_i|M_Q)$

$$=\sum_{q_i\in Q} P(q_i|M_Q)*\log\frac{P(q_i|M_D)}{P(q_i|M_Q)}-\sum_{q_i\in Q} P(q_i|M_Q)*\log\frac{P(q_i|M_C)}{P(q_i|M_Q)}$$

$$=-KL(M_Q,M_D)+KL(M_Q,M_C)$$

对同一Q，为常数

$$\propto -KL(M_Q,M_D)=\sum_{q_i\in Q} P(q_i|M_Q)*\log P(q_i|M_D)-\sum_{q_i\in Q} P(q_i|M_Q)*\log P(q_i|M_Q)$$

负的交叉熵

$$\propto \sum_{q_i\in Q} P(q_i|M_Q)*\log P(q_i|M_D)$$

$$P(Q|M_D)=\frac{|Q|!}{\prod_{q_i\in Q} tf(q_i,Q)!}\prod_{q_i\in Q} P(q_i|D)^{tf(q_i,Q)}$$

$$P(Q|M_C)=\frac{|Q|!}{\prod_{q_i\in Q} tf(q_i,Q)!}\prod_{q_i\in Q} P(q_i|C)^{tf(q_i,Q)}$$

多项分布

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

统计语言建模IR模型优缺点

- 优点：
  - 理论上具有解释性，有扩展空间
  - 有些模型虽然计算上仍然依赖于term独立性假设，但是模型本身并不依赖于term独立性假设。
- 缺点：
  - 数据稀疏性，需要参数估计

提纲

1. 上一讲回顾
2. 语言模型
3. 基于统计建模的IR模型
4. SLMIR模型讨论

SLMIR vs. VSM (1)

- SLMIR中有一些东西和VSM一样
- 词项频率直接在模型中使用
  - 但是在SLMIR中没有进行放缩变化(scaled)
- 本质上概率表示已经进行了长度归一化
  - VSM中的余弦归一化也做了类似工作
- 文档频率和文档集频率混合以后和idf的效果相当
  - 那些文档集中比较罕见，但是某些文档中比较普遍的词项将对排序起更重要的影响。

SLMIR vs. VSM (2)

- SLMIR vs. VSM：共性
  - 模型中都直接使用了词项频率
  - 本质上概率表示已经进行了长度归一化
  - 文档频率和文档集频率混合以后和idf的效果相当
- SLMIR vs. VSM：不同
  - SLMIR：基于概率论
  - VSM： 基于相似度，一个线性代数中的概念
  - 文档集频率 vs. 文档概率
  - 词项频率、归一化等计算细节

基于统计建模的IR模型: 假设

- 简化假设：查询和文档是同一类对象，与实际并不相符!
  - 已经出现了一些不采用上述假设的SLMIR模型
  - VSM也基于同一假设
- 简化假设：词项之间是独立的
  - 同样，VSM中也采用了词项独立性假设
- 比向量空间中的假设表述更清晰
- 因此，比VSM具有更好的理论基础
  - … 但是纯语言模型的效果会大大低于经过精心调参的向量模型的效果。

参考资料

- 《信息检索导论》第12章
- Ponte and Croft's 1998 SIGIR paper (one of the first on LMs in IR)
- Lemur toolkit (能够很好地支持语言模型)
- ChengXiang Zhai. Statistical Language Models For Information Retrieval, 一本综述书籍