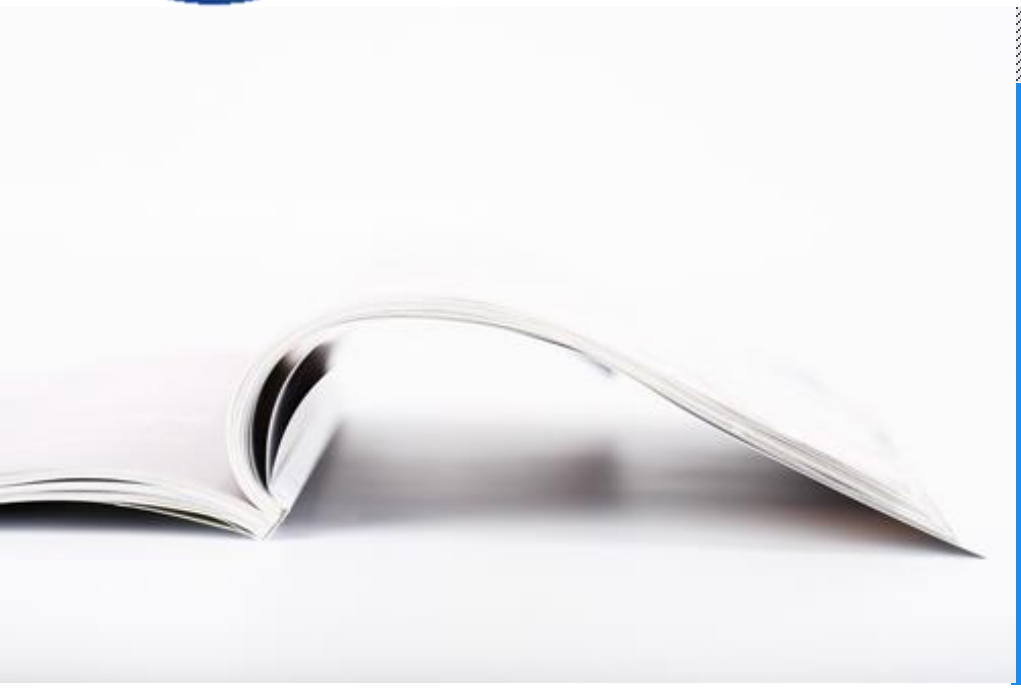


Web信息处理与应用

第五节

查询与评估

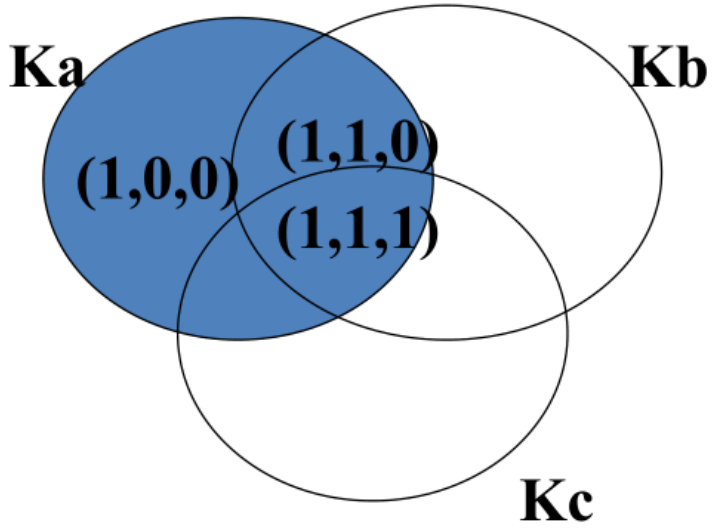
徐童

2022.10.8

上节回顾

- **布尔检索的概念**

- 在布尔检索中，文档被表示为**关键词的集合**。
- 所有的查询式都被表示为关键词的**布尔组合**。
  - 采用“与、或、非”关系加以连接
- 相关度计算
  - 一个文档当且仅当它能够满足布尔查询式时，才会将其检索出来。
  - 检索策略是**二值匹配**。



- **布尔检索的优缺点**

优点

- 查询简单，易于理解
- 使用布尔表达式，可以方便地控制查询结果
- 可通过扩展来包含更多功能

缺点

- 功能较弱，不支持部分匹配
- 所有匹配文档均返回，不考虑权重和排序
- 很难进行自动的相关性反馈

本节引导

- **布尔检索的重要局限性**

盛宴 or 饥荒

- **需要花费大量的精力设计查询条件（Query），才能获得较为合适的结果**
  - 搜索“中国科学技术大学”，可以得到将近3000万条结果。
  - 搜索“中国科学技术大学的XX老师”，结果无限趋近于0
  - 如何获得数量适中、内容符合需求的查询结果?

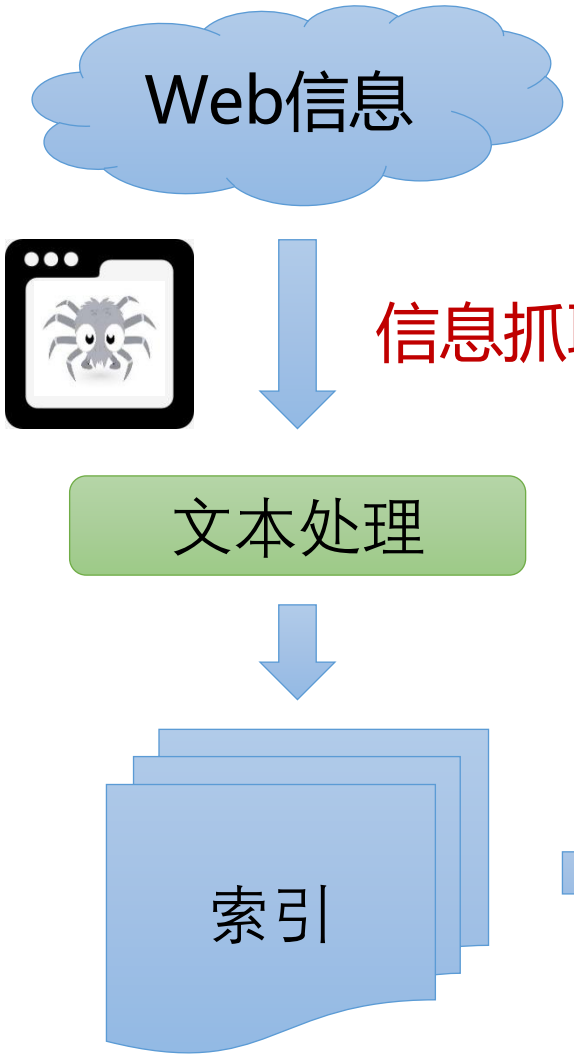
本节引导

- **更一般的检索方式**

- 采用**排序方式**代替严格匹配模式
  - 在排序检索中，系统根据文档与查询的**相关性排序**返回文档集合中的合适文档，而不是简单匹配查询条件
  - 自由文本查询：用户查询条件是**自然语言描述**，而不是由查询词项构造的表达式。
- 当系统给出的是**有序**的查询结果时，结果数目将不再是个问题
  - 着眼于给出Top N结果，而不是完整结果

本节主题

- 本课程所要解决的问题



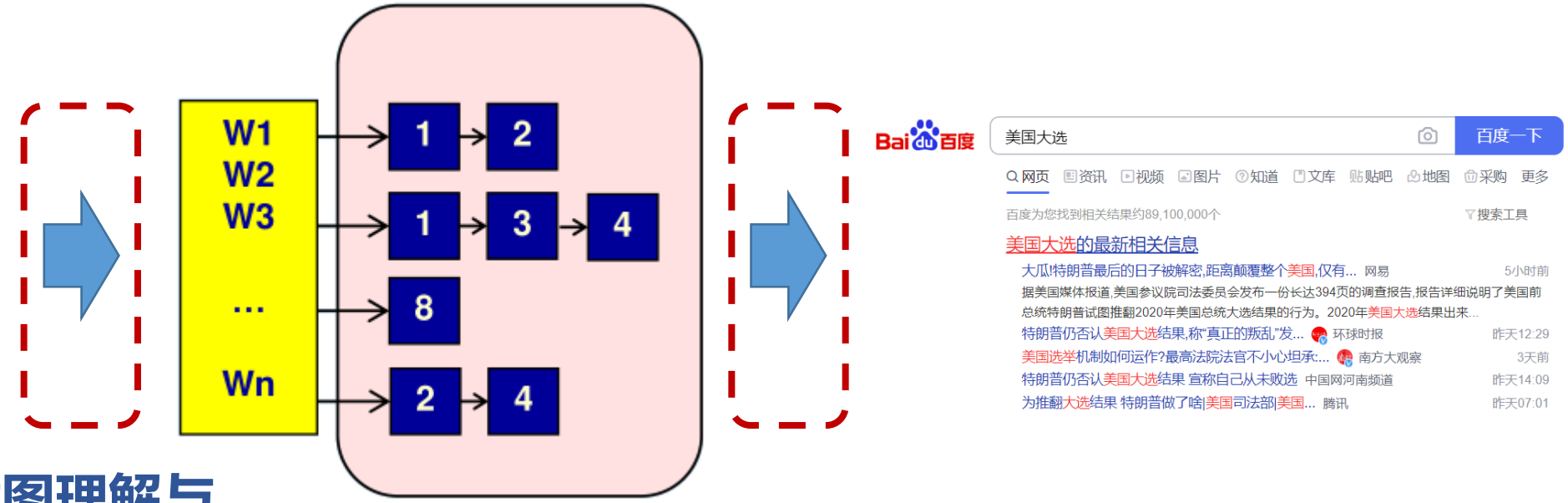
第四个问题：

如何衡量用户对于查询结果的满意程度？

本节目录

- **查询表达理解**
- 相关性反馈
- 查询评估
  - 单查询评估
  - 多查询评估
- 结果多样性评估

- **查询理解所涉及的环节**



意图理解与
查询优化

查询结果改进

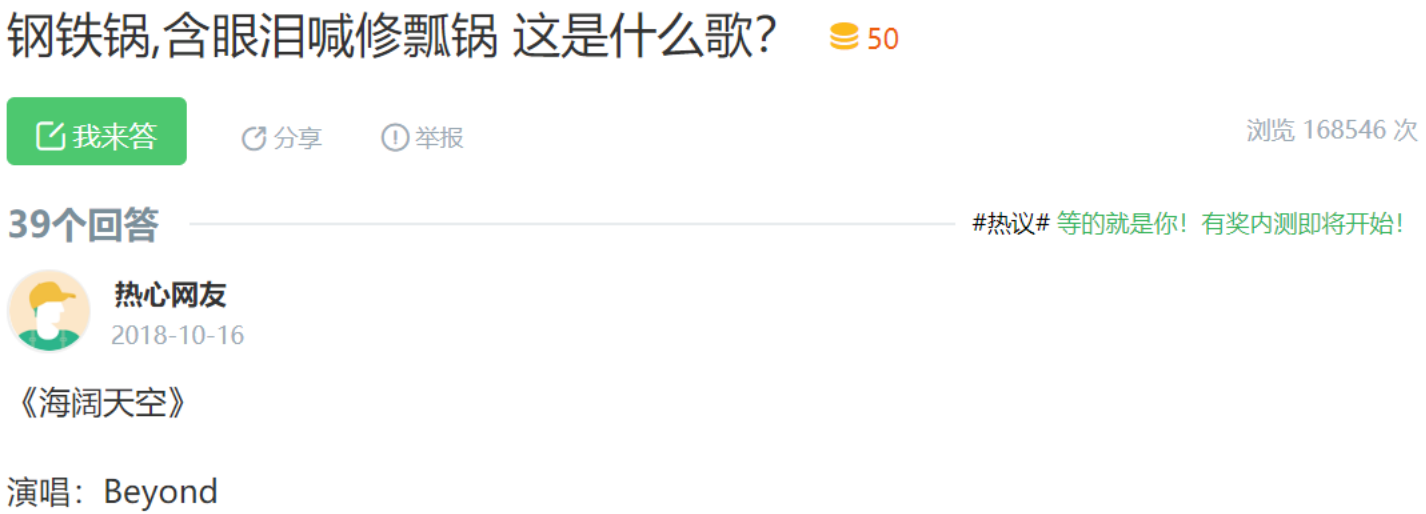
满足用户需求的第一步，在于准确理解用户的查询意图

- **查询表达的重要性**

- 从信息检索定义的视角看查询条件的重要性
  - 信息检索：给定查询条件，从文档集合中找出与查询条件相关的文档
  - 查询条件：用户对信息需求的表达
  - 文档集合：待检索的文档库
  - 相关度：返回文档对于信息需求的满足程度

- **为什么查询条件难以理解?**

- 用户表述方式的差异性
  - 可能是用户所用词汇与索引词项的差异，如同义词、方言等
  - 也可能是表述方式的糟糕，或者信息错漏导致的误导



- **为什么查询条件难以理解?**

- 用户表达的精简性和歧义性
  - 常用简单词汇表达查询需求，缺乏精准描述

动物世界_央视网(cctv.com)

2019年9月28日 - CCTV-3综艺频道《动物世界》《动物世界》栏目已经走过20多年,通过专家的讲述、优美的画面、感人的故事去告诉观众、打动观众,使观众认识到我们不能没有...

tv.cctv.com/lm/dw... - 百度快照

动物世界|动物世界全集视频 - CCTV1直播网

栏目标题:动物世界 播放频道:CCTV-1综合 播出时间:每天00:20(除周二) 持续时间:30分钟 栏目介绍:《动物世界》栏目于1981年12月31日开播,主旨在于向电视观众介绍...

www.cctv1zhibo.com/don... - 百度快照

or

- **为什么查询条件难以理解?**

- 用户表达中可能存在侧重点，不同词项的重要性不尽相同
  - 然而，侧重点无法直接从字面意义上看出

- **理解用户查询的几种方式**

- 最基本的途径：基于查询的自然语言处理
- 引入相关性反馈
  - 用户直接对查询结果进行评价
  - 引导用户表达真实查询意图（查询建议与查询扩展）
- 借助其他信息，完善对于用户的理解
  - 用户间接性反馈

本节目录

- 查询表达理解
- **相关性反馈**
- 查询评估
  - 单查询评估
  - 多查询评估
- 结果多样性评估

- **何为相关性反馈 (Relevant feedback)**

- 用户在查询后标记相关/不相关，然后迭代更新查询，以获得更好的结果
- 相关性反馈的动机
  - 你也许无法表达想要找的内容，但是你至少能够判断所看到的内容
    - “为我提供更多*相似的文档*……”
  - 精准的查询条件或许无法一蹴而就，但可以通过迭代逐渐趋于精准

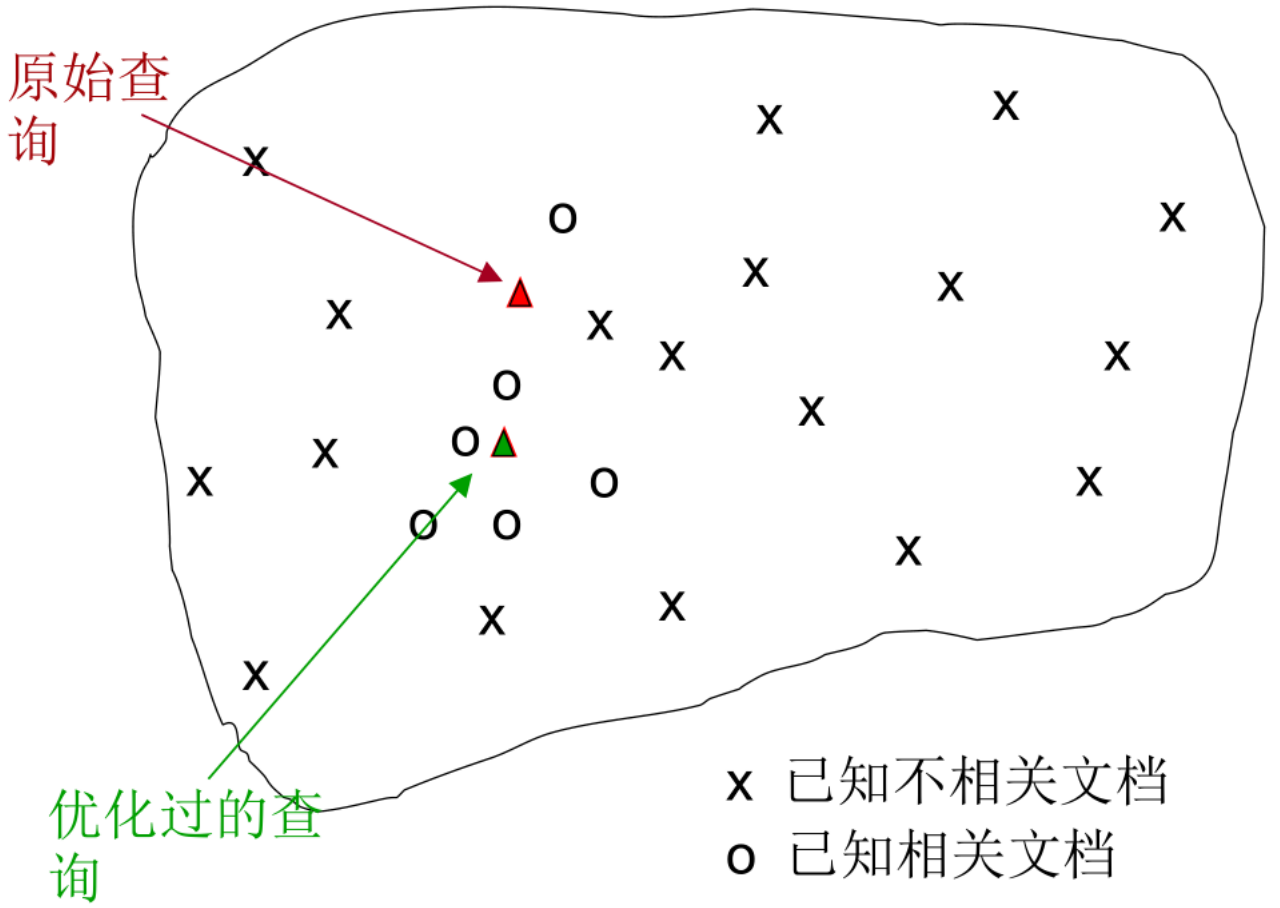
- **相关性反馈的基本流程**

1. 首先，用户提出一个查询条件（Query）
2. 对于返回的文档，用户标出相关与不相关的部分
3. 系统根据用户反馈，获得用户信息需求更为准确的描述
   a) 基于相关性反馈，更新查询条件
   b) 基于新查询条件，获取新的结果文档并再次提交用户进行评估

➢ 上述过程将根据情况进行**一次或多次的迭代**，从而**不断接近最优查询条件**

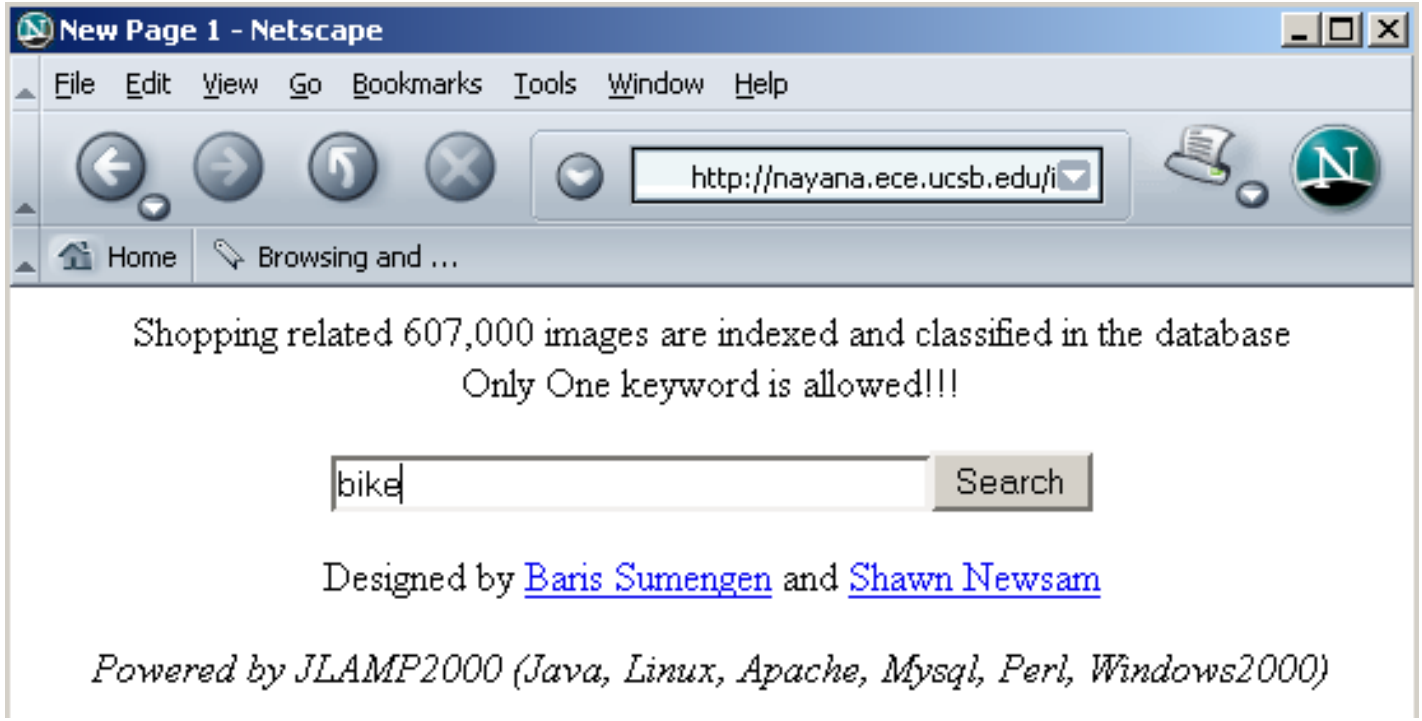
- **相关性反馈的最终目的**

- 通过相关性反馈，获得用于表达用户查询意图的最优查询条件
  - 常见方式：为已有词项添加不同权重，或增加新的词项
  - 这一过程应对用户隐藏



- **相关性反馈实例**

  - 用户的初始查询需求：搜索“Bike”相关的图片



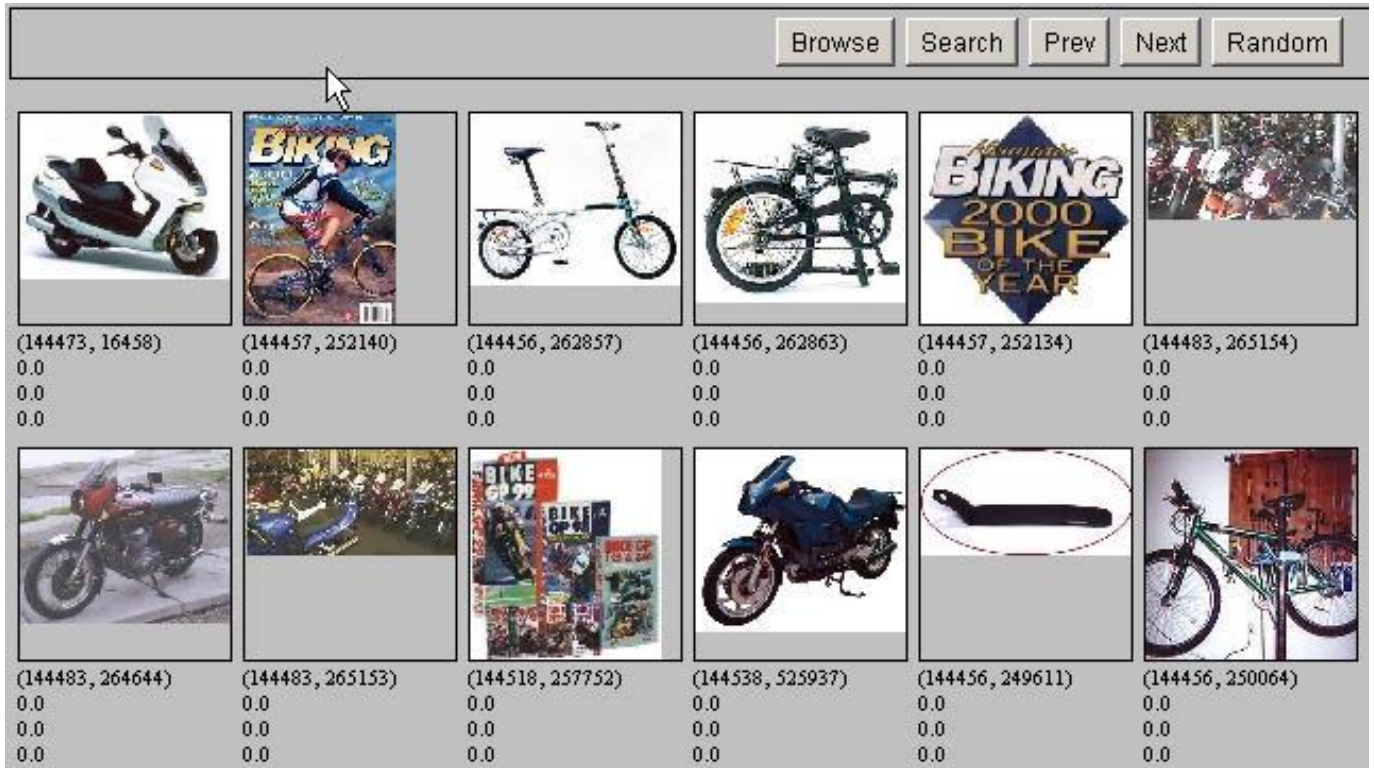
- **相关性反馈实例**

  - 用户的初始查询需求：搜索“Bike”相关的图片

bike

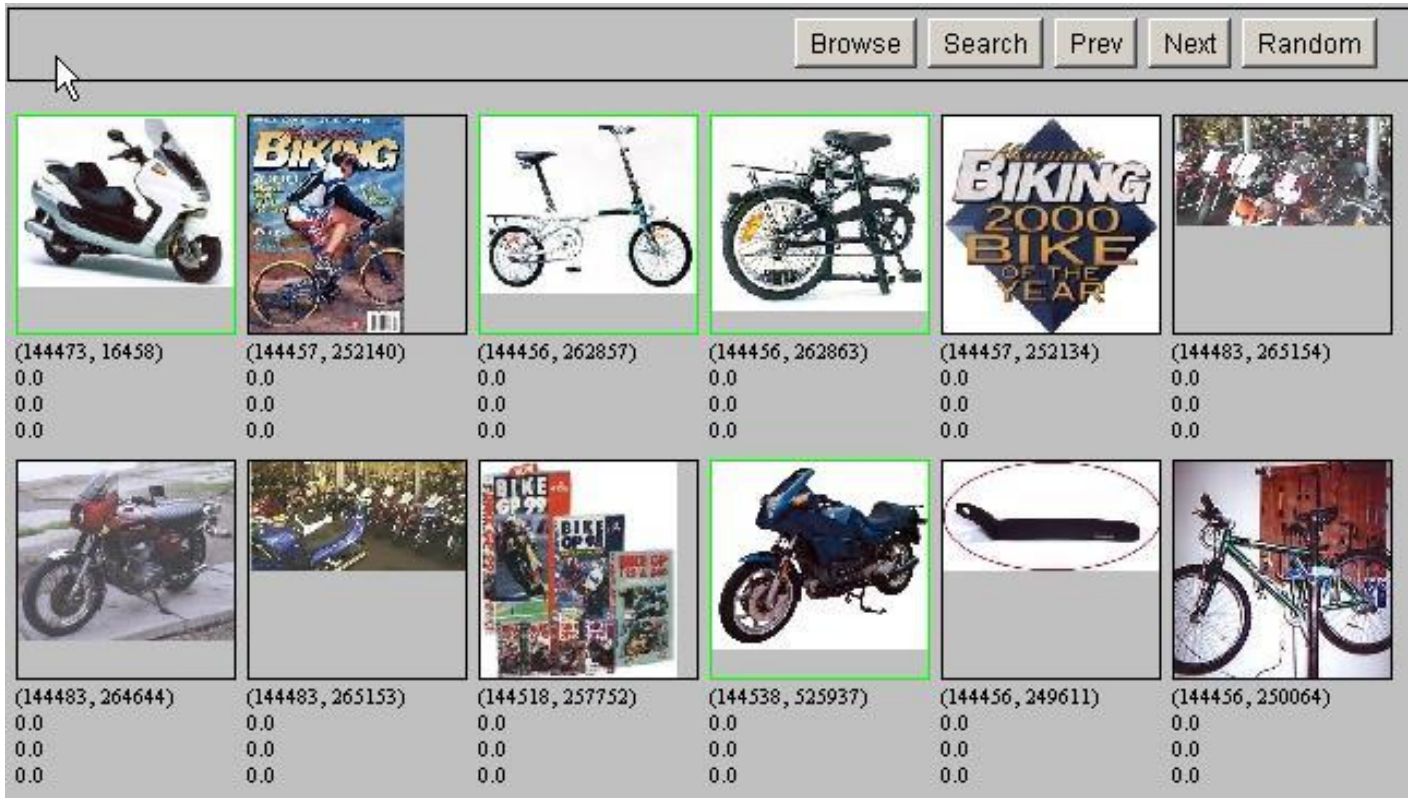
- **相关性反馈实例**

  - 基于查询条件的初始检索结果



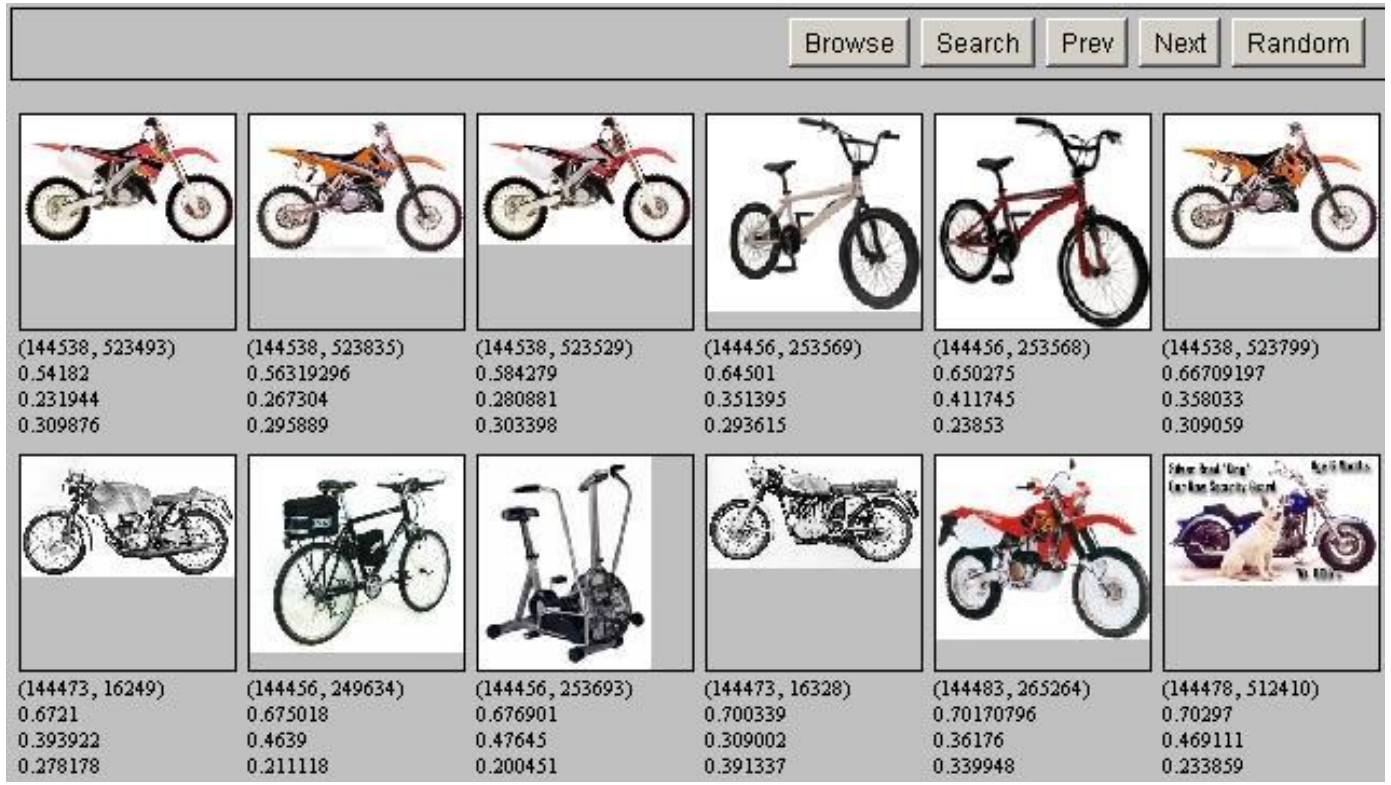
• 相关性反馈实例

- 用户对部分相关图片进行了标记（或点击等行为）



- **相关性反馈实例**

  - 基于用户反馈，得到了更新的搜索结果，以车型图片为主



- **相关性反馈存在的问题**

- 相关性反馈可能影响用户体验
  - 用户不愿意提供显式的相关反馈
  - 用户不希望因为相关性反馈（迭代）而显著延长搜索时间
- 相关反馈生成的新查询往往很长，降低系统效率，增加计算开支
  - 一种做法是只改变重要词项权重而不增加新词项，但效果有限
- 有时很难理解，为什么经过相关性反馈后，会返回不相关的文档
  - 被相关性反馈捕捉的词项，未必是用户需要的内容

- **常见的相关性反馈类型**

- 显式反馈（Explicit Feedback）
  - 用户显式地参与交互过程
- 隐式反馈（Implicit Feedback）　　← **更为常见！**
  - 系统追踪用户行为来推测返回文档的相关性
- 伪反馈（Pseudo Feedback）
  - 在没有用户参与的前提下，直接假设返回结果是相关的，并进行反馈

- **显式反馈**

- 最基础的显式反馈：用户点击记录
  - 显而易见的缺陷：只有正样本
    - 用户不点击，不代表完全不相关！
- 拓展的显式反馈：收集负面评价的渠道日益丰富

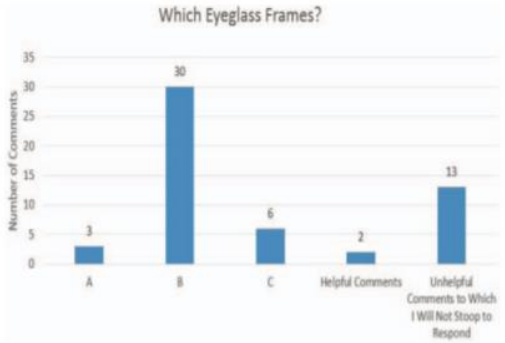
- **显式反馈**

- 更为复杂的显式反馈：用户评论

- 基于用户评论，可以收集更为完整的相关性反馈
- 同时，对于网页质量有更为可靠的判断

- **隐式反馈**

- 通过观察用户对当前检索结果采取的行为，来判断检索结果的相关性
  - 判定不一定很准确，但省却了用户的显式参与行为
- 常见的用户行为种类
  - 鼠标键盘动作，如点击、停留、翻页、拷贝等
  - 用户眼球动作，如凝视、移动、拉近、拉远等



- **隐式反馈**

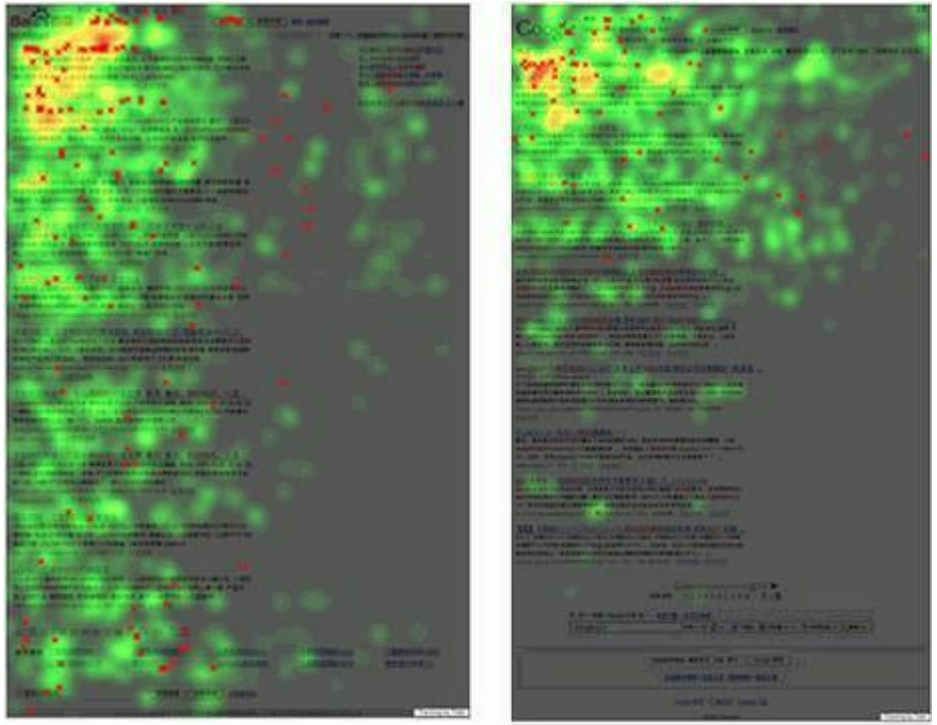
- 鼠标键盘动作可能揭示用户身份特征
  - 他山之石：《暗算》，“手迹”识别报务员
  - 不同用户在击键频率、时延、习惯、错误率等方面存在一定差异

[发明专利]通过键盘鼠标输入习惯识别实现操作用户身份判别的方法 有效

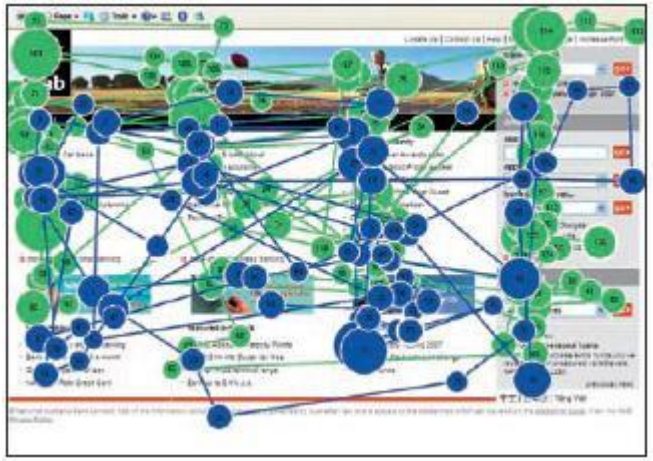
申请号：	CN201110110807.7	文献下载	
申请日：	2011-04-29	公开/公告日：	2011-09-14
公开/公告号：	CN102184359A	主分类号：	G06F21/00

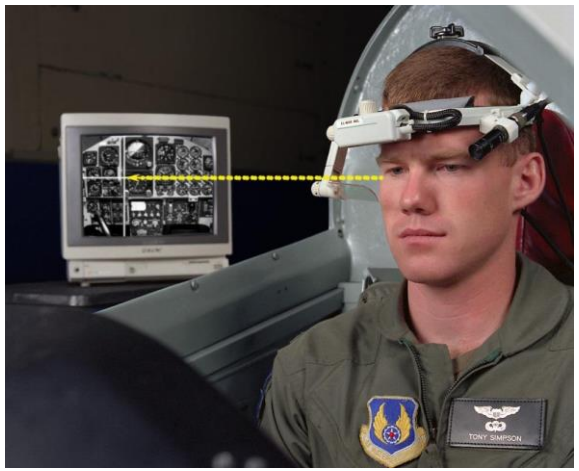
- **隐式反馈**

- 用户眼球动作，可以揭示用户关注的内容及关联（视觉注意特征）



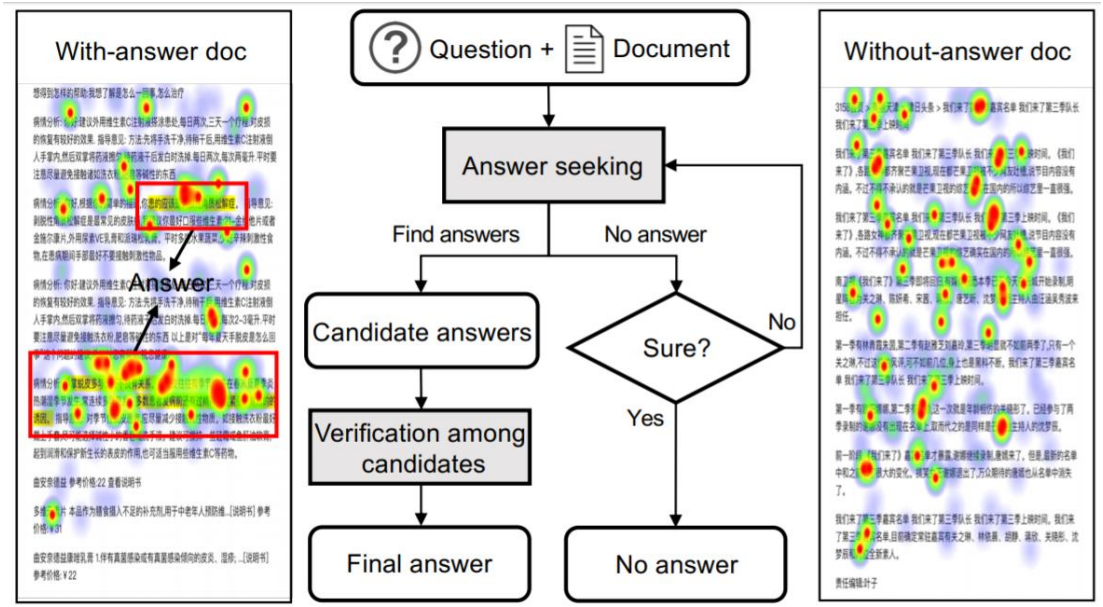
Baidu Google





- **隐式反馈**

- 借助眼球动作捕捉，还可以支撑其他相关的应用
  - 例如，判断文本之间的相关性，甚至揭示问题的答案



Y Zheng, et al., *Human Behavior Inspired Machine Reading Comprehension*, SIGIR 2019

- **隐式反馈的优缺点**

- 优点：
  - 不需要用户显式参与，减轻用户负担，提升用户体验
  - 用户行为某种程度上可以反应其兴趣，因此具有可行性
- 缺点：
  - 对行为分析有着较高的要求
  - 准确度难以保证
  - 某些情况下需要增加额外设备（且很贵！）

- **伪相关性反馈**

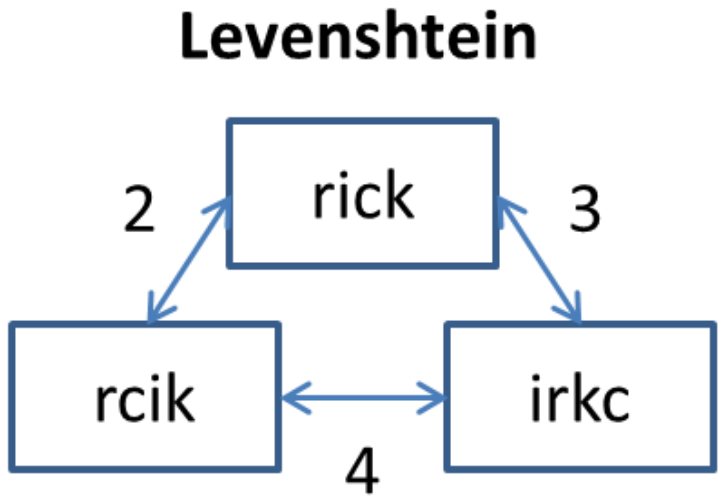
- 无需用户参与反馈过程，而直接根据检索结果自动反馈，较为简单
  - 对于用户查询返回的有序结果，假定前K篇文档是相关的
  - 在此基础之上，进行相关性反馈
- 实验证实，利用伪相关性反馈技术可以提升检索的效果
- 但是，伪相关性反馈存在显著的隐患：
  - 结果未经用户判断，难以保证其准确性
  - 某些查询可能结果很差，甚至出现查询漂移（被Top K文档带了节奏）

- **查询扩展：一种用户相关性反馈的特殊方式**

- 在相关性反馈中，用户针对文档给出反馈，这些反馈将被用来更新查询条件。
- 而在查询扩展中，用户针对词项的合适程度给出反馈，这些反馈将被用来构建更为完整的查询条件。
  - 暗含的功能：用户选择和确认的查询扩展能够更好表达其查询意图。

- **内容回顾：拼写检查**

- 拼写检查在本质上也是对查询的修改和完善
  - 通常采用基于词典或编辑距离的方式进行检查和校对。
  - 编辑距离（Levenshtein Distance）：两个字符串间转换最少需要的编辑步数



- **查询扩展的实现**

- 利用同义词辞典，可以实现查询条件的扩展
  - 对于某个查询词汇，使用辞典中的同义词或相关词进行扩展
    - Feline（猫科）→ Feline cat
    - 相对于原始的查询词汇，可以给扩展的词汇设定更小的权重
  - 一般而言，查询扩展有助于提升查询的召回率（找得更全）
    - 但是，可能会影响准确率，尤其在扩展词存在歧义的情况下
- **编纂和维护同义词词典需要很大的代价**

- **查询扩展的类型**

- 利用人工编纂的同义词词典
  - 例如，第三节“网页文字处理”中提到的How-Net、大词林等
- 全局分析与同义词词典的自动生成
  - 基于统计词汇之间的共现关系（Co-occurrence），自动构建词典
- 基于搜索日志进行优化
  - 通过查询日志，挖掘查询的等价类

- **自动构建相关词词典的两种思路**

- 通过分析文档集中的词项分布，来自动生成同义词/近义词词典。
- 基本的想法是计算词语之间的相似度，常见的两种思路如下：
  - 思路1：如果两个词经常和相似的词共同出现，则它们很可能是相似的
    - E.g., Car和Motocycle很可能相似，因为它们和road、gas等经常共现
  - 思路2：如果两个词经常与相同的词以一种特定的语义关系共同出现，那么他们很可能是相似的
    - E.g., 可烹调并食用的实体往往都属于食物

- **基于搜索日志的查询扩展**

- 搜索日志目前是搜索引擎查询扩展的重要方式
  - 实例1：提交查询Herbs（草药）后，用户往往搜索Herbal remedies（草本疗法）。
    - 此时，Herbal Remedies是Herbs的潜在扩展查询
  - 实例2：用户搜索Flower Pix与Flower Clipart时，往往都会点击URL：photobucket.com/flower
    - 此时，Flower Pix与Flower Clipart可能互为潜在扩展查询

- **相关词扩展的潜在问题**

- 词项关联的质量是一个问题
  - 有歧义的查询词可能导致统计上相关，而意思上不相关的词
    - 如“Apple Computer”可能导致“Apple Red Fruit Computer”
  - 同时，由于扩展的查询词与原查询词高度相关，扩展后的查询也未必能够获得更多的相关文档。

本节目录

- 查询表达理解
- 相关性反馈
- **查询评估**
  - 单查询评估
  - 多查询评估
- 结果多样性评估

- **结果评价的常见内容**

- 最主要（用户最关注）的两方面内容
  - 性能（Effectiveness）
    - 返回了多少相关文档，是否有遗漏，排序是否靠前
  - 效率（Efficiency）
    - 响应速度如何，时间和空间开销有多高
- 除此之外，其他指标也可衡量查询的结果
  - 结果的多样性、权威性、时新性与更新频率

- **结果评价的常见内容**

- 评价效率（Efficiency）的常见指标

指标	英文名	解释
索引时间	Elapsed Indexing Time	特定系统中构建文档索引所需的总时间
索引处理器时间	Indexing Processor Time	构建文档索引所需要的CPU秒数， 即不考虑文件读写时间与并行加速影响
查询吞吐量	Query Throughput	每秒能够处理的查询数量
查询延迟	Query Latency	用户在输入查询后得到查询结果的等待时间
索引临时空间	Indexing Temporary Space	为构建索引所临时占据的硬盘空间
索引空间	Indexing Size	存储索引所需的硬盘空间

上节回顾

- **信息检索模型中的性能指标**

- 性能通常可通过以下两个维度进行衡量
  - 主题相关：文档与查询在主题上的一致性
    - 某种程度可体现在字面意义上的匹配性
  - 用户相关：文档在多大程度上满足用户需求
    - 可通过前面所介绍的用户反馈进行判定
- 相应的，可以通过二元（相关/不相关）或排序的方式进行评判

- **结果评价的常见内容**

- 评价性能（Effectiveness）的常见指标
  - 面向单个查询的评价指标
    - 无序/二元结果：Precision、Recall、F-value…
    - 有序/多元结果：P@N、R@N、AP、NDCG…
  - 面向多个查询的评价指标
    - MAP、MRR及其各种拓展指标

本节目录

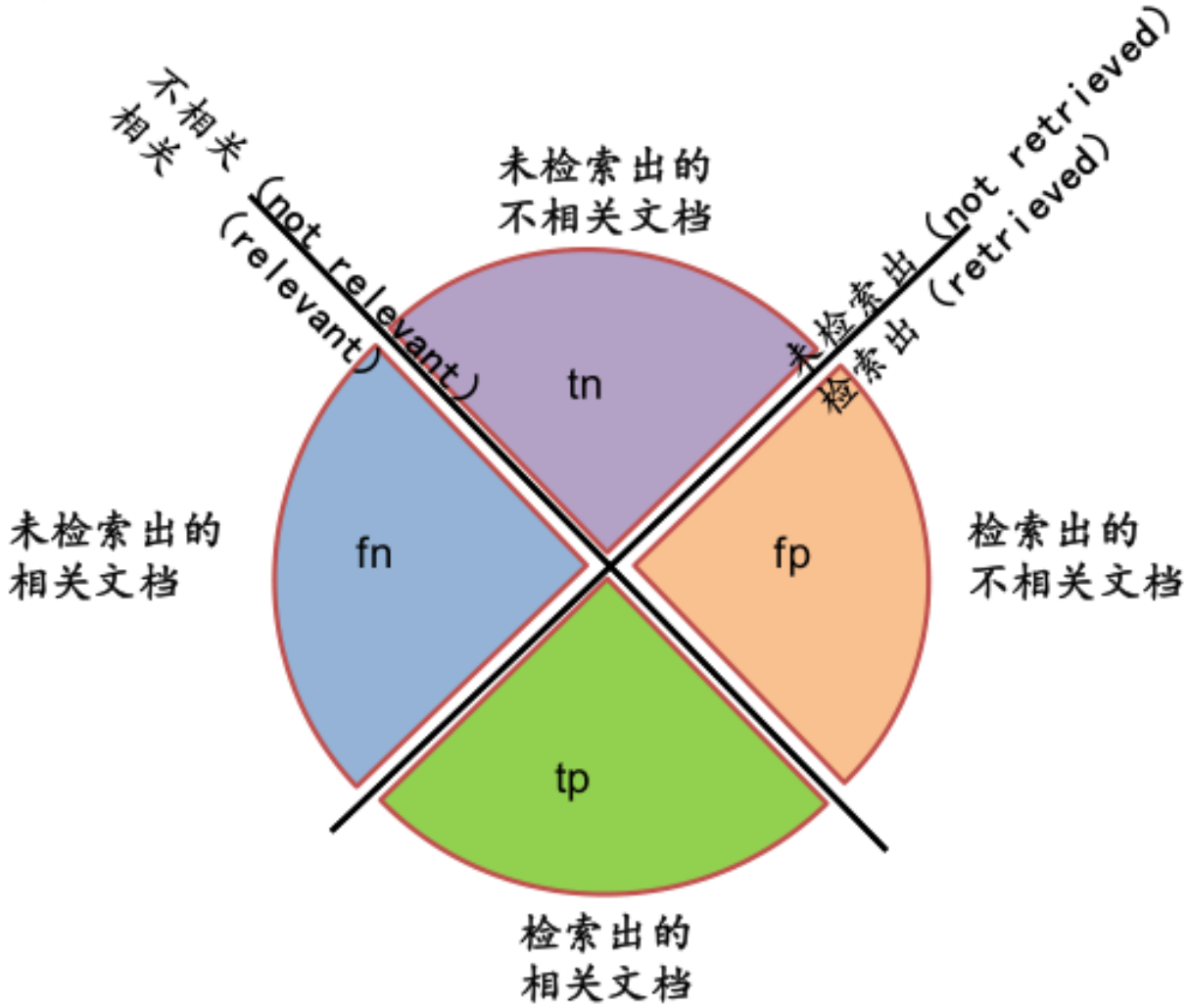
- 查询表达理解
- 相关性反馈
- **查询评估**
  - **单查询评估**
  - 多查询评估
- 结果多样性评估

- **基本评价指标的矩阵化表示**

  - 两个视角下对于数据集的切分
    - P/N：Positive or Negative，表示算法对样本的判断
    - T/F：True or False，表示算法判断的正确与否（和结果是否一致）
    - 四种简写的含义：
      - TP：True Positive，样本为正例，且被判定为正，即真正
      - FN：False Negative，样本为正例，但错误地被判定为负，即假负
      - FP：False Positive，样本为负例，但错误地被判定为正，即假正
      - TN：True Negative，样本为负例，且被判定为负，即真负

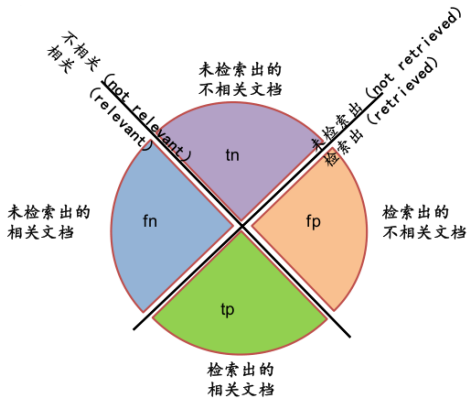
	被检索文档	未检索文档
相关文档	TP	FN
不相关文档	FP	TN

- **文档集合的基本划分**


不相关
相关
(not relevant)
(relevant)
未检索出 (not retrieved)
检索出 (retrieved)
未检索出的
不相关文档
tn
fn
fp
未检索出的
相关文档
检索出的
不相关文档
tp
检索出的
相关文档

- **面向单查询的基本评价指标**

- 准确率（Precision）
  - 指检索出的文档中，相关文档所占的比例，也称查准率
  - 计算公式为TP/(TP+FP)
- 召回率（Recall）
  - 指所有相关文档中，被检索出来的部分的比例，也称查全率
  - 计算公式为TP/(TP+FN)



- **一个P-R指标计算的实例**

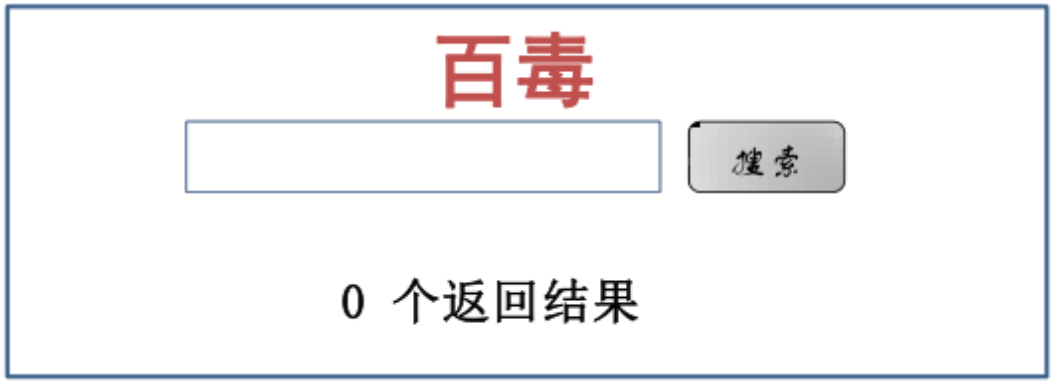
- 当查询1 的标准答案集合为 {d3,d4,d6,d9}时，可知:
  - 对于系统1，查询1：正确率2/5，召回率2/4
  - 对于系统2，查询1：正确率2/4，召回率2/4

系统&查询	1	2	3	4	5
系统1，查询1	d3√	d6√	d8	d10	d11
系统1，查询2	d1	d4	d7	d11	d13
系统2，查询1	d6√	d7	d2	d9√	／
系统2，查询2	d1	d2	d4	d13	d14

- **为什么某种方案被抛弃?**

- 既然TP与TN都是正确结果，为什么不直接计算(TP+TN)的全局比例?
  - (TP+TN)/(TP+TN+FP+FN)，即Accuracy，在模式分类中经常被使用
  - 然而，它在信息检索的相关任务中并不常见，为什么?

如何以最低的代价做一个Accuracy接近 100% 的搜索引擎?



- **准确率与召回率的平衡**

- 不同应用场景中，对于准确率和召回率有着不同的侧重
  - 邮件分类：宁愿放过一些垃圾邮件，也不能错杀正常邮件
    - 牺牲（对垃圾邮件的）召回率，保证较高准确率
  - 智慧医疗：宁愿多判断一些疑似患者，不能漏掉一个真实病人
    - 牺牲（对确诊病人的）准确率，保证较高召回率

- **召回率的近似计算**

- 对于大规模文档集合，列举每个查询的所有相关文档是不可能的事情
  - 因此，不可能准确地计算召回率

- **召回率的近似计算**

- 解决方法：缓冲池（Pooling）方法
  - 针对某一检索问题，各个算法分别给出检索结果中的Top N个文档
  - 将这些结果汇集起来并进行人工标注，从而得到一个相关的文档池
  - 潜在假设：大多数相关文档都在这个文档池（Doc Pooling）中
- 这一方法的可行性在于，虽然它实际上仍然无法得到全部相关文档，因此并不能得到召回率的绝对值。但是，它可以比较各个算法的相对优劣
  - 因此，这一算法在各个测评中被广泛采用，N通常取50-200

- **从P-R的平衡到F值**

- 如前所述，准确率与召回率之间存在权衡
  - 如何综合评价一个算法在这两项指标上的性能?
- F值（F-measure），即准确率与召回率的加权调和平均数

$$F = \frac{1}{\alpha \frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R}$$

- 通常情况下，我们取α=0.5或β=1（即两者同等重要）
  - 此时，可得基本的F1值，即F=2PR/(P+R)

- **从P-R的平衡到F值**

- 为何不使用算数平均，而使用调和平均综合这两个指标？如何综合评价一个算法在这两项指标上的性能？
  - 本质上说，我们希望的目标是Precision和Recall都比较高
    - 调和平均在本质上对一方偏低的情况是有惩罚的，必须两方都较高才会高
  - 算数平均和几何平均在处理极端情况下的效果并不够合理
    - 一个不好的例子：如果采用算术平均计算F值，那么一个返回全部文档的搜索引擎的其F 值就不低于50%，显然这不是一个好结果
- 准确率、召回率与F值是信息检索任务中最为基础和常用的三个指标

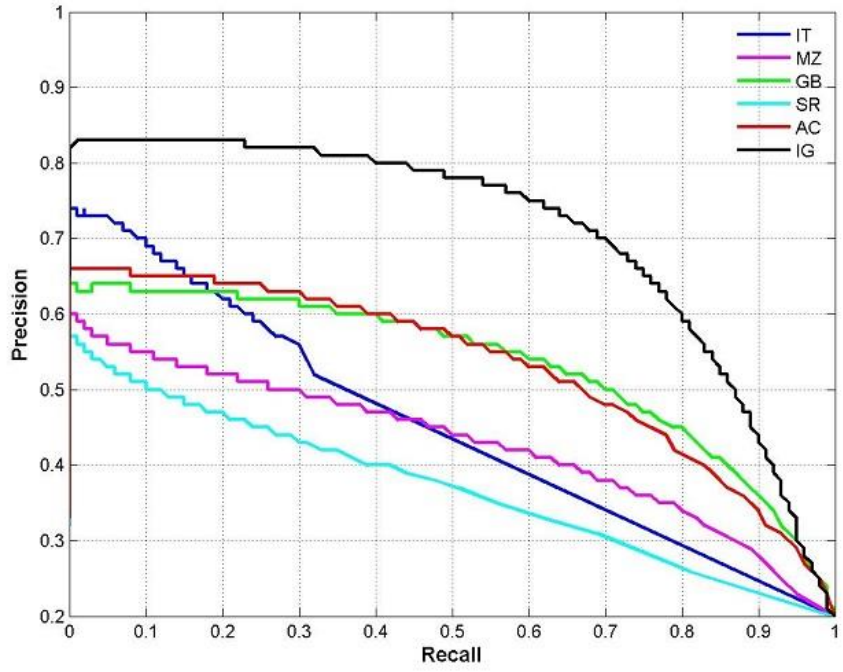
P-R曲线与ROC曲线

- 在分类问题中，准确率与召回率的平衡是通过选定不同阈值实现的
  - 例如，通过调控相关性阈值θ，可以控制检索所得的文档数量
  - 较低的阈值可以使得返回更多文档，但也混入大量不相关的文档
  - 较高的阈值可以保障文档的相关性，但也会遗漏许多相关的文档
  - 如何选择合适的阈值？
    - 通过绘制不同阈值下的指标变化曲线，可以帮助我们做出选择

Tips：对每个文档，计算其与查询的相关性系数，若大于θ即认定为相关文档

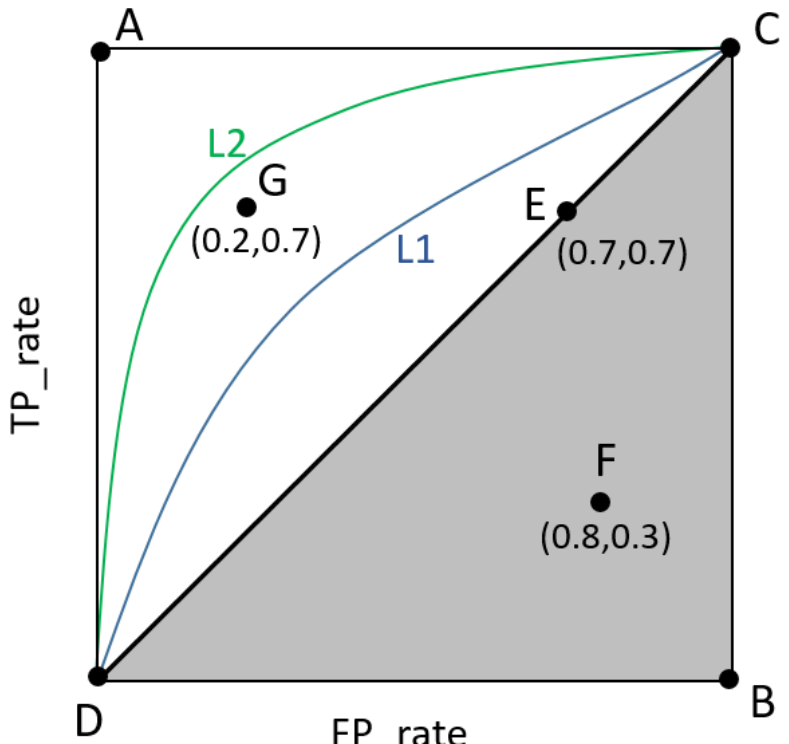
- **P-R曲线与ROC曲线**

- P-R曲线（Precision-Recall Curve）
  - 以准确率和召回率分别作为两条轴线
  - 通过选定不同的阈值得到不同的P-R点并连接成线
  - 通过P-R曲线，可以直观地看出准确率与召回率之间的平衡关系



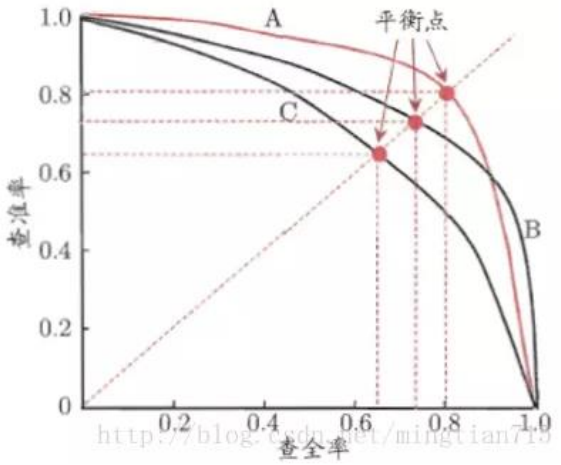
- **P-R曲线与ROC曲线**

- ROC曲线（Receiver Operating Characteristic Curve，接受者操作特征曲线）
  - 以真正率【TP/(TP+FN)】和假正率【FP/(FP+TN)】作为两条轴线
    - 真正率/命中率，假正率/误报率
  - 通过选定不同的阈值得到不同的真正率-假正率点并连接成线
  - 对角线表示区分能力为0，即随机猜测
  - 在对角线上端越远，效果越好
  - 低于对角线的结果无意义（无区分度）



- **P-R曲线与ROC曲线**

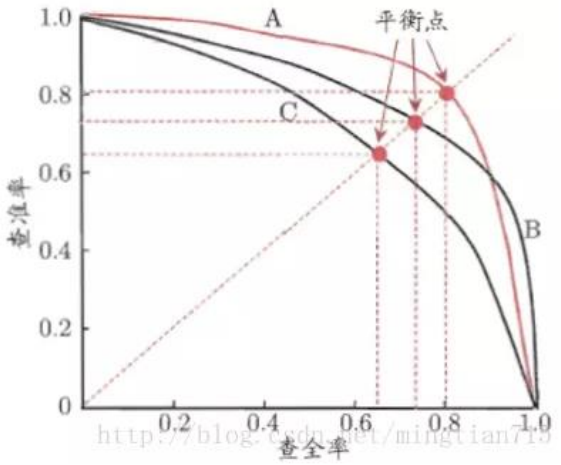
- 如何基于P-R曲线或ROC曲线判别算法好坏
  - 如果线A将线B完全包住，显然线A对应的算法效果更好
  - 如果两条线发生重合，则可依据以下规则判别：
    - 计算AUC，AUC更高者效果更好
      - Area under curve，即曲线下面积
      - 可通过积分近似计算
    - 另外，当使用P-R曲线时，可使用平衡点计算
      - 平衡点即Precision = Recall的点，值越高越好



查准率-查全率曲线

- **P-R曲线与ROC曲线**

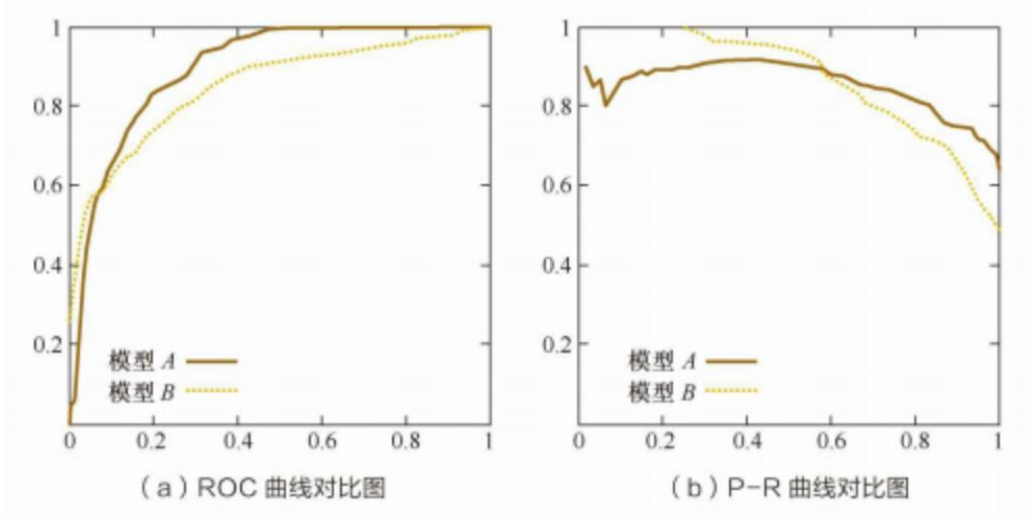
- P-R曲线与ROC曲线的选择
  - ROC曲线兼顾正负样例，更为全面，而P-R曲线则只考虑正例
    - 用户往往更关心正样本，如果面向特定应用场景（如检索），P-R曲线是个好选择
  - 单份数据正负样本比例失调时，P-R曲线更合适
    - 当负样本比重过高时，负例的数目众多致使FPR的增长不明显，导致ROC曲线呈现一个过分乐观的效果估计，从而难以体现出性能的差异性



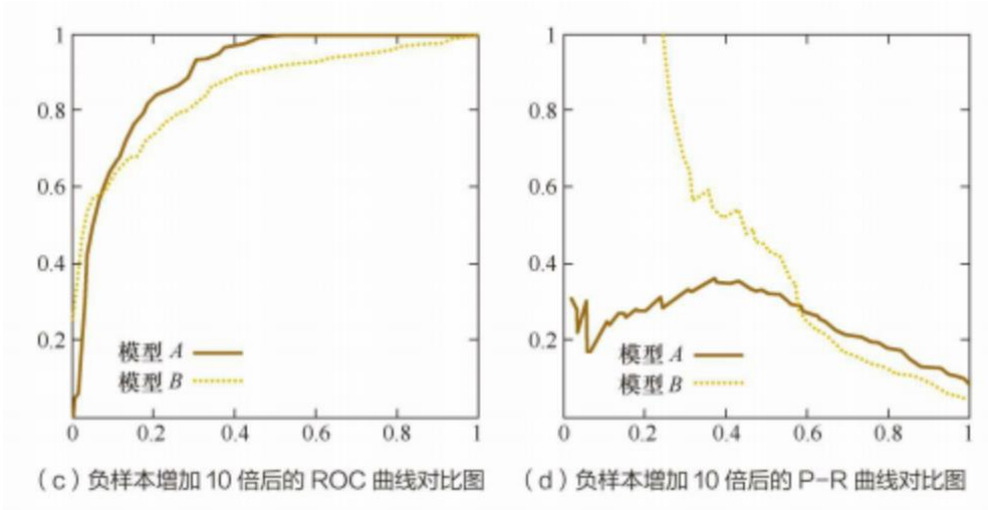
查准率-查全率曲线

- **P-R曲线与ROC曲线**

- P-R曲线与ROC曲线的选择
  - P-R曲线受分布影响大，多份数据且正负比例不一时ROC曲线更合适
    - ROC曲线两个指标各自针对正负样本，而Precision只针对正样本，受影响较大



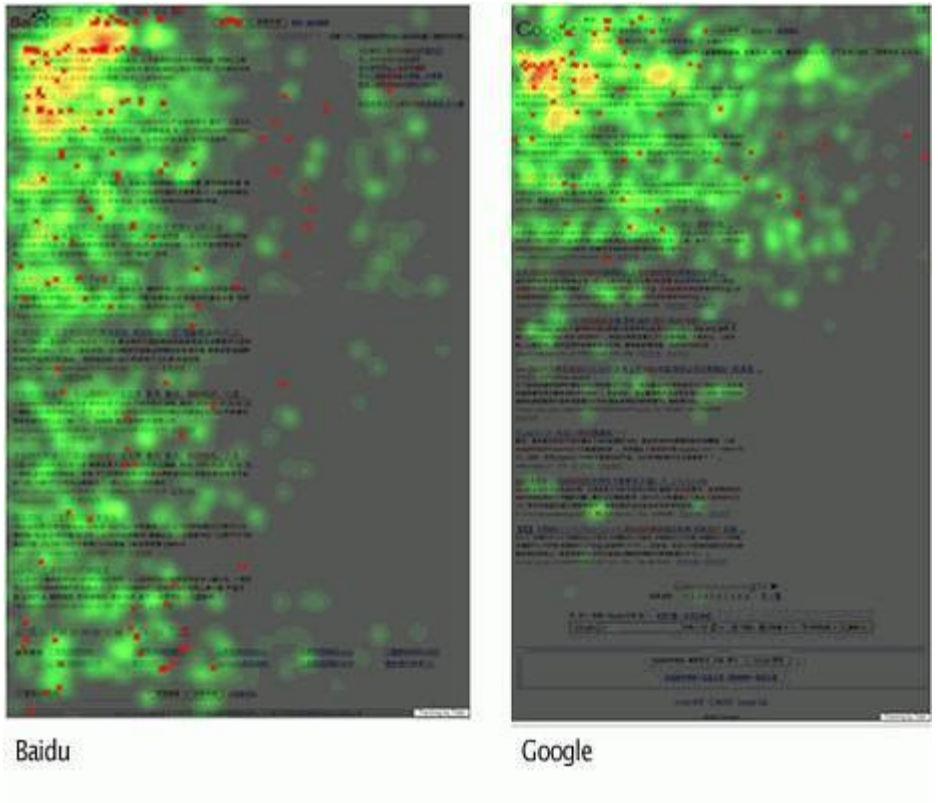
（a）ROC 曲线对比图　（b）P-R 曲线对比图



（c）负样本增加 10 倍后的 ROC 曲线对比图　（d）负样本增加 10 倍后的 P-R 曲线对比图

- **用户更关心有序结果**

- 一个好的搜索引擎，应该尽可能将用户需要的文档排在较为靠前的位置



- **从无序的P、R到有序的P@N、R@N**

- P、R、F都是基于集合计算的指标，面向无序文档集合进行计算
  - 因此，它们无法直接应用于有序文档集合，需要进行拓展
  - 而这一点显然局限了它们的功能，并且可能导致误导
    - 例如，两个系统对某查询都返回20个文档，其中相关文档数都是10，但第一个系统是前10条结果，后一个系统是后10条结果。
    - 显然，在这一情况下，前者更为优越，但单纯P、R难以区分。
- 解决方案：引入序的作用和区分
  - P@N、R@N、AP、NDCG…

P、R@N的概念与意义

- P@N，即Precision@N，指前N个检索结果文档的准确率
  - 由于大多数用户只关注第一页或前几页，因此P@10、P@20等对于大规模搜索引擎来说是很合适的评价指标。
  - 如果相关文档数小于N，P@N的理论上限必定小于 1
- 同理，可得R@N，即Recall@N，指前N个检索结果找回的相关文档比例
  - 由于返回结果有限，Recall@N值，甚至其理论上限往往都远小于 1
    - 理论上限为 N/相关文档数，即使通过Pooling加以控制仍然较小

- **P、R@N的实例**

- 如果查询1 的标准答案集合为 {d3,d4,d6,d9}，那么有：
  - 系统1 查询1: P@2=1, P@5=2/5;
  - 系统2 查询2: P@2=1, P@5=3/5

系统&查询	1	2	3	4	5
系统1，查询1	d3 √	d6 √	d8	d10	d11
系统1，查询2	d1 √	d4	d7	d11	d13 √
系统2，查询1	d6 √	d7	d2	d9 √	
系统2，查询2	d1 √	d2 √	d4	d13 √	d14

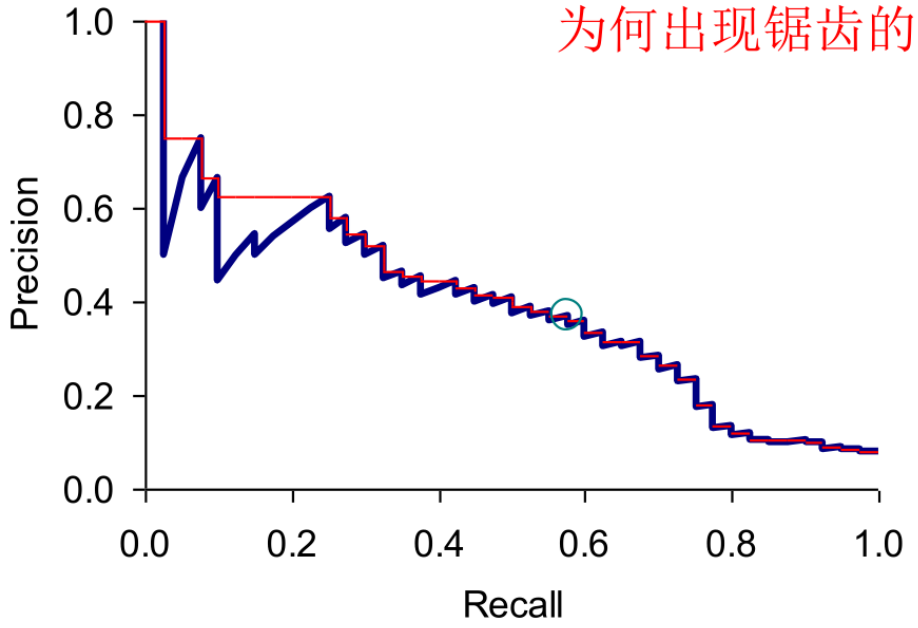
- **一种特例：R-Precision**

- 检索结果中，在所有相关文档总数位置上的正确率
  - 由于N往往小于相关文档总数，因此设计了这一特殊指标
  - 例如，如果相关文档总数为3，那么考察P@3作为R-Precision
  - 实例：如果查询2 的标准答案集合为 {d1,d2,d13}，那么：
    - 系统1 : R-Precision=1/3； 系统2 : R-Precision=2/3;

系统&查询	1	2	3	4	5
系统1，查询1	d3 √	d6 √	d8	d10	d11
系统1，查询2	d1 √	d4	d7	d11	d13 √
系统2，查询1	d6 √	d7	d2	d9 √	
系统2，查询2	d1 √	d2 √	d4	d13 √	d14

- **面向P、R@N的P-R曲线**

- 在有序结果情况下，可以不再采用不同阈值作为P-R值的依据，而是通过依次计算前N个结果对应的P-R值绘制曲线



  - 新的不相关文档被检索时，Recall不变，Precision下降

- **更多的评价准则：AP**

- 平均准确率（Average Precision，AP）
  - 用于对不同召回率点上的正确率进行平均
  - 通常情况下，AP有三种不同的定义与计算方法
    - 未插值AP：查询Q共有6个相关结果，排序返回了5篇相关文档，位置分别是第1，第2，第5，第10，第20位，则AP=(1/1+2/2+3/5+4/10+5/20+0)/6 （注意补0）
    - 插值AP：事先选定插值点数并进行插值。例如，当我们计算11点平均时，计算在召回率分别为0（第一条），10% , 20%,…, 100%的十一个点上的正确率求平均
    - 简化AP：只对返回的相关文档进行计算，AP=(1/1+2/2+3/5+4/10+5/20)/5，倾向那些快速返回结果的系统，**没有考虑召回率和补零的情况** （不补0）

- **更多的评价准则：AP**

- 三种AP的计算实例(假设共有十篇相关文档)

Example

1. d123 •(1/1)	6. d9 • (4/6)	11. d38 • (7/11)
2. d84	7. d511	12. d48
3. d56 •(2/3)	8. d129 • (5/8)	13. d250
4. d6 •(3/4)	9. d187	14. d113 • (8/14)
5. d8	10. d25 • (6/10)	15. d3

- 未插值的AP = (1/1+2/3+3/4+4/6+5/8+6/10+7/11+8/14+0+0)/10 (注意补0)
- 11点插值的AP = (1+1/1+2/3+3/4+4/6+5/8+6/10+7/11+8/14+0+0)/11
- 简化的AP = (1/1+2/3+3/4+4/6+5/8+6/10+7/11+8/14)/8 (不补0)

- **分级的必要性与考虑相关度加和的度量**

- 先前的各种指标，都是基于相关/不相关的二元评判
- 然而在现实中，用户对文档的评价往往更为复杂，无法用二元简单概况
  - 例如，如果用户只是想了解《动物世界》这部电影的概况，那么百度百科即可。
  - 然而，如果用户是想观看《动物世界》这部电影，那么右侧图示的排序就相对较差，因为相关度更高的“在线观看”内容排序靠后。

- **基础的相关度加和**

- 累计增益（Cumulative Gain，CG）
  - 用于衡量位于位置1 到 p 的检索结果的相关度之和。

$$\mathrm{CG_p} = \sum_{i=1}^{p} rel_i$$

  - Rel 用于描述文档相关性，可以根据需求选取多个数值 / 级别。
  - 较高的CG表明文档的整体相关性较高。
  - 然而，由于CG并未考虑文档位置，并不能体现靠前部分文档的质量。

- **改进的相关度加和**

- 如何区别位置的作用？直观想法：对不同位置赋予不同折损
- 折损累计增益（Discounted Cumulative Gain，DCG）
  - 基本思想：若搜索算法把相关度高的文档排在后面，则应该给予惩罚。
  - 一般用log函数来表示这种惩罚，如log(i+1)，i 为文档位置
  - 往往有以下两种计算公式（后者采用指数，更突出相关性）：

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{log_2(i)}$$

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(i+1)}$$

• 归一化的相关度加和

- 然而，DCG由于其随着长度单调非减的特性，仍具有其局限性
  - DCG与具体查询和结果列表的长度p有关，不利于不同算法之间的对比
  - 不同查询的结果有多有少，因此其DCG值无法实现相互比较
- 对DCG进行规范化：归一化折损累计增益（Normalized DCG，NDCG）
  - 基本思路：将DCG除以完美结果下得到的理想结果，iDCG（ideal DCG）
    - 即：NDCG = DCG / iDCG
    - 其中，iDCG是根据文档根据相关性从大到小排序得到理想化的最优序列，并对此序列计算DCG值所得到的。

- **NDCG计算实例**

- 假设有10个文档，相关度为0-3之间，10 个文档的得分依次如下：
  - 3, 2, 3, 0, 0, 1, 2, 2, 3, 0
- 理想的输出结果序列为：3, 3, 3, 2, 2, 2, 1, 0, 0, 0
  - 由此计算 iDCG依次为：3, 6, 7.89, 8.89, 9.75, 10.52, 10.88, 10.88, 10.88, 10.88
- 而与此同时，基本的DCG结果如下：
  - 3, 5, 6.89, 6.89, 6.89, 7.28, 7.99, 8.66, 9.61, 9.61（单调非减特性）
- 由此可得 NDCG结果如下：1, 0.83, 0.87, 0.76, 0.71, 0.69, 0.73, 0.8, 0.88, 0.88
  - 可以看到任何查询结果位置的NDCG 值都规范化为[0,1]之间的值

本节目录

- 查询表达理解
- 相关性反馈
- **查询评估**
  - 单查询评估
  - **多查询评估**
- 结果多样性评估

从AP到MAP

- 从单查询拓展至多查询评价，可以更全面地体现排序算法的综合性能
  - 如何对多查询的结果进行综合?
- MAP（Mean AP），对所有查询的AP求算数平均
- 例如：假设有一个检索系统
  - 对查询1 返回4 个相关网页，其rank 分别为1, 2, 4, 7
  - 对查询2 返回3 个相关网页，其rank 分别为1, 3, 5
  - 查询1 共有4 个相关文档，查询2 共有5个相关文档

查询1：AP = (1/1+2/2+3/4+4/7)/4=0.83
查询2：AP = (1/1+2/3+3/5+0+0)/5=0.45
MAP = (0.83+0.45)/2 = 0.64

←此处使用了未插值AP

- **MAP的变形：GMAP**

- MAP可以反映全部查询的综合效果，但在查询难度不平衡的条件下有误导

系统	主题	AP	提升	MAP
系统A	主题1	0.02	-	0.113
	主题2	0.03	-	
	主题3	0.29	-	
系统B	主题1	0.08	+300%	0.107
	主题2	0.04	+33.3%	
	主题3	0.20	-31%	

- 单纯从 MAP 来看，系统A 好于系统B 。
- 但是从每个查询来看，3 个 主题中有2 个主题，系统 B 都比 A 有提高，其中一个提高的幅度达到 300%

- **MAP的变形：GMAP**

- 通过引入基于几何平均值的GMAP（Geometric MAP）解决这一问题
  - 削弱绝对数值的影响，从而提升相对强弱的影响
- $GMAP = \sqrt[n]{\prod_{i=1}^{n} AP_i} = \exp(\frac{1}{n}\sum_{i=1}^{n} \ln AP_i)$
- 基于这一方法，上述那个例子的结果可修正为：
  - $GMAP_a$= 0.056， $GMAP_b$ = 0.086，因此系统B更为出色
- 从这个例子可以看出，MAP与GMAP各有所长
  - 当各个查询间难度不均，或存在较难排序的主题时，GMAP或许更合适

- **注重首个相关文档的MRR**

- 在许多查询任务中，用户只关心第一个相关的文档，越靠前越好
  - 该位置的倒数被称作Reciprocal Rank（RR），数字越大效果越好
- 对多个查询所得的倒数排序求平均，即Mean RR，MRR
- 例如，两个查询，第一个查询的第一个相关文档在位置2，第二个查询的第一个相关查询在位置4
  - 则MRR为(1/2+1/4)/2 = 3/8
  - 即平均在8/3的位置上找到第一个相关文档

- **MRR的拓展模型：ERR**

- 当用户发现了第一篇相关文档后，后面的内容可能就不再关注了
- 一篇文档可能被用户点击的概率大致估计为：$PP_r = R_r \prod_{i=1}^{r-1}(1 - R_i)$
  - 其中Rr表示位置为r的文档的相关度
- 由此，可以定义预期的倒数排序（Expected RR，ERR）如下：
  - $ERR = \sum_{r=1}^{n} \frac{1}{r} PP_r = \sum_{r=1}^{n} \frac{1}{r} R_r \prod_{i=1}^{r-1}(1 - R_i)$
  - 表示用户的需求被满足时停止的位置的倒数的期望

- **一个值得关注的指标：方差**

- 对于一个测试文档集合，检索系统常常对有的查询表现的很好，而对有的查询表现很差。
  - 先前的例子可以揭示这一现象，e.g., GMAP时的例子。
  - 通常情况下，一个检索系统对不同查询的方差，往往大于多个检索系统对相同查询的方差。
  - 由此可见，不同查询的难度差异较大，有些查询确实很难。

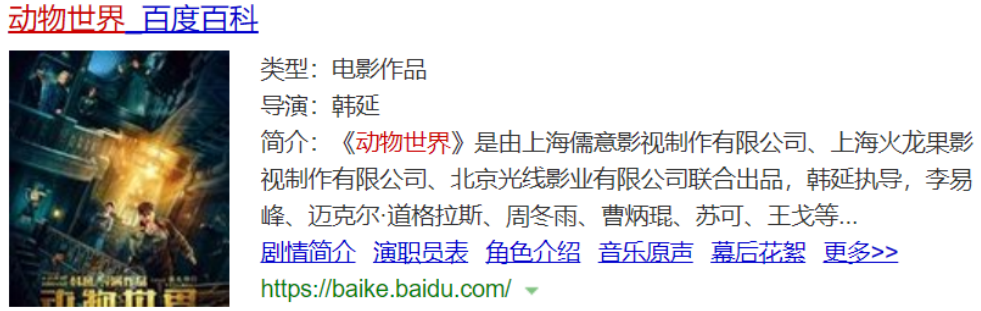
本节目录

- 查询表达理解
- 相关性反馈
- 查询评估
  - 单查询评估
  - 多查询评估
- **结果多样性评估**

• 为什么要考虑多样性

- 一方面，用户的单次搜索可能体现出多方面的需求
- 另一方面，用户搜索可能存在歧义，需要展示多方面内容加以确认

or



- 此外，也要避免信息茧房的产生

- **多样性的形式化定义**

- 基本形式：给定一个查询q，返回一个多样化的结果文档集合R(q)。
- 其中，R(q)作为一个整体，应满足以下条件：
  - R(q)中所有的结果文档都与查询q本身有较大的相关性。
  - 总体上要有较小的冗余度，以覆盖q的不同方面。

- 核心思想：降低用户无法获得所需信息的风险
  - 尽可能确保排序靠前的结果中至少有一个结果满足用户的需求。

- **多样性的两种衡量方式**

- 总体需求：衡量不同文档之间的主题差异性。
- 一般而言，衡量方式有以下两种：
  - 隐式模型：只计算文档之间的差异性
    - 文档是什么内容，不会也无法进行详细考量
  - 显式模型：更加具体地考量文档所对应的用户意图
    - 会从文档中抽取主题，并显式地实现主题的多样化

- **隐式模型代表：MMR**

- 最大边界相关性（Maximal Marginal Relevance，MMR）
  - 最早的相关性衡量模型，由Carbonell和Goldstein在1998年提出

$$MMR^{def} = Argmax_{d_i \in R|S}[\lambda P(d_i \,|\, q) - (1-\lambda) max_{dj \in S} P(d_i | d_j)]$$

    - 该公式由两部分组成，其中：
      - 前一半表示文档集与查询 q 的相似性
      - 后一半表示文档之间的多样性
      - λ 用于调节两部分之间的比例
      - 文档可以采用tf-idf，或者word2vec等文本表征工具进行表征

- **显式模型代表：FM-LDA**

- Facet model with LDA (FM-LDA)
  - 由Carterette与Chandar在2009年提出，考虑文档的不同子主题（Facet）

$$L(y_i|F, D) = \prod_{j=1}^{m}\left(1 - \prod_{i=1}^{n}(1 - p(f_j \in d_i))^{y_i}\right)$$

    - 其中，$p(f_j \in d_i)$ 表示文档Di包含主题Fj的概率，Yi=1表示某文档被选中
    - 由此，联乘部分表示主题Fj至少被一个文档所涵盖的概率
    - 当有约束 $\sum y_i \leq l$ 时，返回的文档数量被限定为 l 篇

本章小结

查询与评估

- 查询条件理解
- 相关性反馈
- 面向查询结果的评估
  - 单次查询评估：无序结果/有序结果/分级结果
  - 多次查询评估
- 结果多样性评估

tongxu@ustc.edu.cn