

**评价**

**指标：**性能/效率/结果多样性/权威性/时新性与更新频率

**单查询评价 (1)无序结果评价**

**P/N:**算法对样本的判断;**T/F:**算法判断的正确与否

**准确率Precision:** TP/(TP+FP)

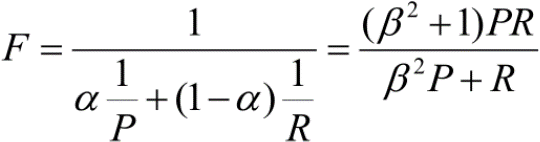
**召回率Recall:** TP/(TP+FN)

召回率的近似 缓冲池Pooling:将各算法给出检索结果中的TopN文档汇集并人工标注，得到相关的文档池

**F-measure：**准确率与召回率的调和平均

取alpha=0.5

或belta=1

 得F=2PR/(P+R)

通过绘制不同阈值下的指标变化曲线可选择合适阈值。**ROC曲线：**真正率TP/(TP+FN)和假正率FP/(FP+TN)作为轴线，兼顾正负

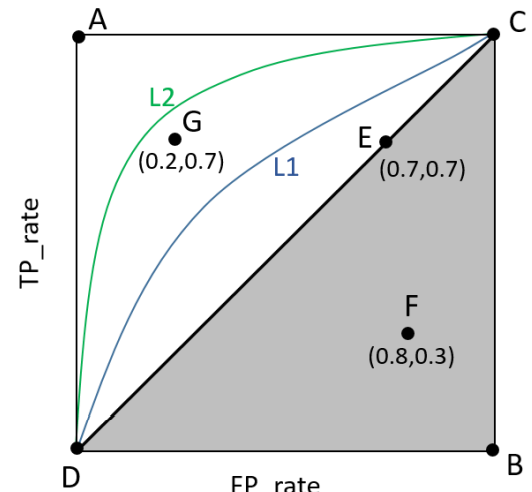
对角线上无区分能力

（即随机猜测）

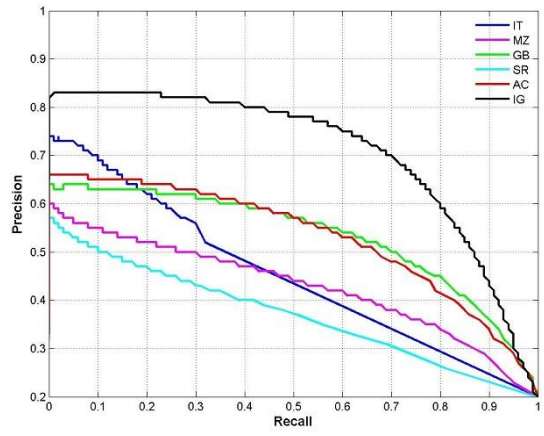
对角线上端越远越好

低于对角线无意义

（无区分度）



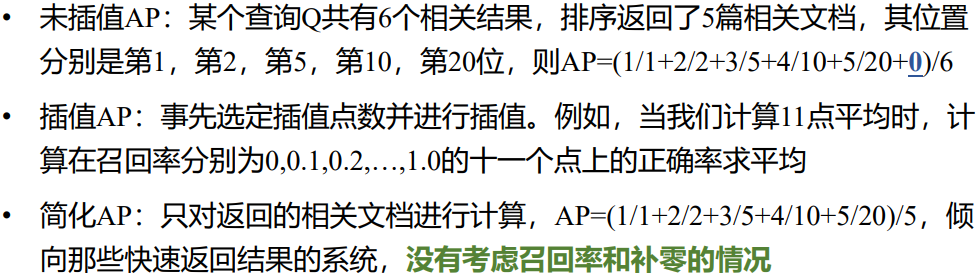
**P-R曲线：**x轴召回率，y轴准确率，不同阈值不同曲线。（只考虑正例）



判别算法好坏：(1)计算曲线下面积AUC，越高效果越好 (2)PR曲线平衡点（P=R）值越高越好

**(2)有序结果评价**

**P@N:**前N个检索结果文档的准确率；**R@N:**前N个检索结果找回的相关文档比例；**R-Precision=**P@相关文档总数**；平均准确率AP：**不同召回率点上的正确率平均

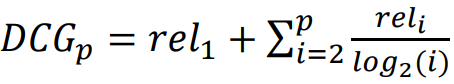


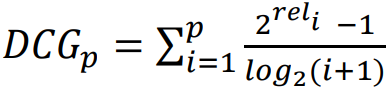
（召回率为0点的正确率为1）

**(3)相关度分级**

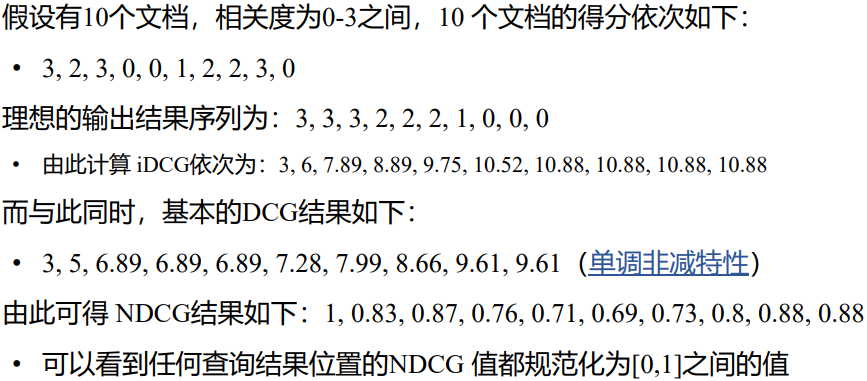
**累计增益CG：**位置1到p的检索结果相关度之和;

**折损累计增益DCG:**若把相关度高的文档排在后面给惩罚;

（ppt用第一个）

或

**归一化折损累计增益NDCG**=DCG/iDCG，其中iDCG是根据文档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。



**多查询评价**

**MAP：**对所有查询的（简易）AP求算数平均

（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）

**GMAP：**



**MRR：**多个查询所得的倒数(第一相关文档)排序求评价

例：两个查询,第一个的第一个相关文档在位置2,第二个的第一个相关查询在位置4,则MRR=(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

**ERR：**表示用户需求被满足时停止的位置的倒数的期望



其中Rr表示位置为r的文档的相关度

**方差：**一个检索系统对不同查询的方差通常大于多个检索系统对相同查询的方差，说明查询难度差异大。

**实体识别**

**信息抽取基本任务：**命名实体NE(实体抽取)、模板元素TE(属性抽取)、共指关系CR(等价)、模板关系TR(关系抽取)、背景模板ST(事件抽取)

**知识图谱 基本单位：三元组（实体-关系-实体）**

节点表示概念和实体；边表示关系和属性

**应用：**语义搜索、问答系统、推荐系统

**拓展类型**：事理图谱（描绘逻辑社会，研究对象是谓词性事件及其内外联系），多模态图谱（属性/实体多模态）

基于路径的知识图谱：提升多样性和可解释性！

**命名实体识别NER 子任务：**判别实体边界；判别实体类型 **识别内容：**实体类（人名/地名/机构名），时间类（日期/时间），数值类（货币/百分比）

**难点：**新实体，歧义，别名/缩略词/音译，实体类型多样

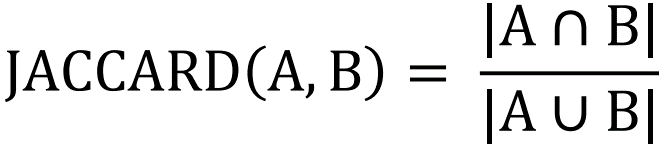
**方法(1)基于词典:**简单快速，与具体语境无关，容易部署和更新；难枚举所有命名实体名，构建和维护词典代价大，难有效处理实体歧义**(2)基于规则:**手工构造规则模板，选用特征包括统计信息/标点符号/关键字/指示词和方向词/位置词/中心词(依赖具体语言、领域和文本风格，代价大)**(2)基于统计:**分类（词性标注）、序列模型

**评价方式：**Precision/Recall/F-value，分子为“返回的正确答案数量（+1/2的部分正确答案数量）”

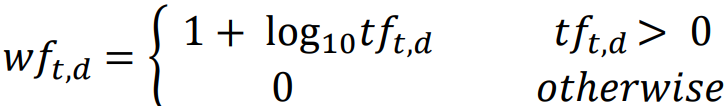
**实体对齐**、**实体消歧**🡨语义表征

**信息检索模型：**文档与查询的表示形式与相关性的框架，实质是对文档基于相关性进行排序，在理解用户的基础之上产生近似用户决策的结果从而在顶部返回最相关的信息。[D,Q,F,R(Di,q)] **布尔模型**D:文档表达-词项的组合；Q:查询表达-布尔表达式；F:完全(二值)匹配；R:满足布尔表达式,相关性为1(否则0)

**相关度计算：**

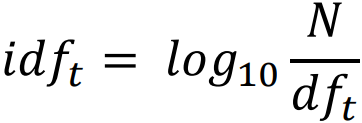
**Jaccard系数：** 

**词项频率TF(t,d):** 词项t在文档d中出现的次数

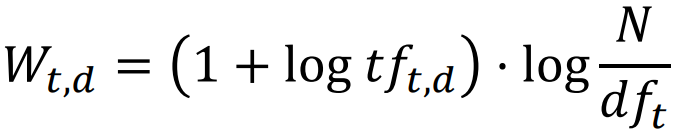
对数词频： 

**文档频率DF(t)：**出现词项t的文档数量

逆文档频率IDF:



**TF-IDF:**

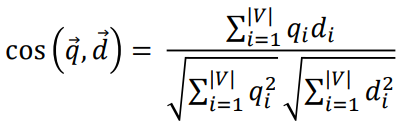


**向量空间模型VSM**文档和查询视作一个词项权重构成的向量->计算向量之间相似性->按相似度大小排序返回Top-K文档给用户。

D:文档表达，每个文档可视作一个向量，其中每一维对应词项的tf-idf值；Q:查询表达，可视作一个向量，其中每一维对应词项的tf-idf值；F:非完全匹配方式；R:使用两个向量之间的相似度来度量文档与查询之间的相关性

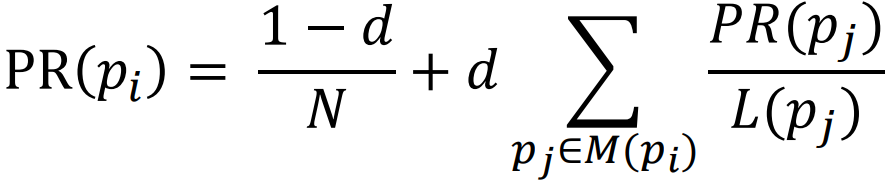
**优点：**简洁直观，可支持多种不同度量和权重方式，实用效果不错；**缺点：**缺乏语义层面的理解和匹配，同时依赖tf-idf值也可能造成干扰，用户无法描述词项之间的关系，词项之间的独立性假设实际上不成立。

**VSM下相似度：**欧式距离；**余弦相似度：**



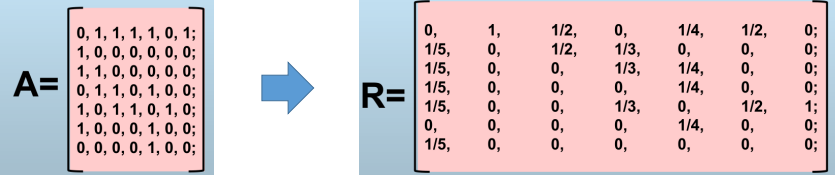
**网页权威性计算**

**PageRank：**优质网页引用或推荐的网页一定也优质。



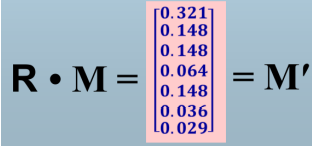
PR(pi)为网页pi的PageRank值，初值为1/N；PR(pi)为指向网页pi的某个网页pj的PageRank值；L(pj)为网页pj发出的链接数量；d为阻尼系数，取值在0-1之间；N为网页总数，M(pi)为链入pi的页面集合。

●邻接矩阵->跳转矩阵：转置后各数值除以出边数量

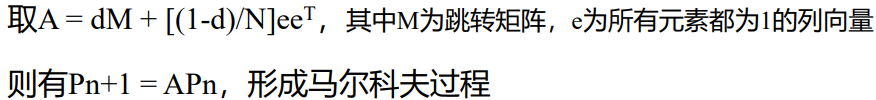


●M=[1/7,…1/7]T。

若M与M’对应维元素差值

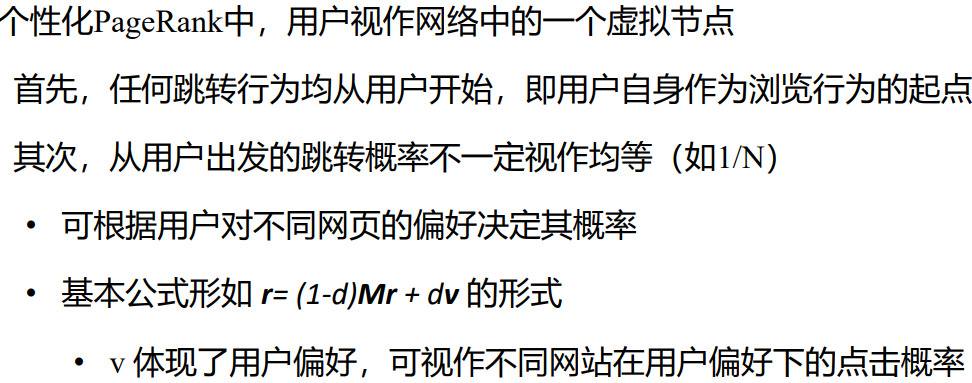
高于阈值则继续迭代至收敛：

**陷阱节点:**只有一条指向自己的边，没有其它出边；**终止节点:**没有出边，如同黑洞；**孤立结点：**没有任何入边，单纯使用markov链无法跳转，仅有初始概率，不再更新也不影响其它结点。**解决：重启动：**d或alpha （随机跳转概率是1-d）。**why？**其中的(1-d)/N的部分相当于以一定等概率被选中作为新起点，由此跳出了陷阱和黑洞的干扰（所有节点全联通）。**收敛？**马尔可夫过程的三个收敛条件（转移矩阵A所有元素非负，且每一列元素和为1，为马尔科夫矩阵；Restart保障图是强连通，则A不可约；A非周期）Pagerank算法都满足。

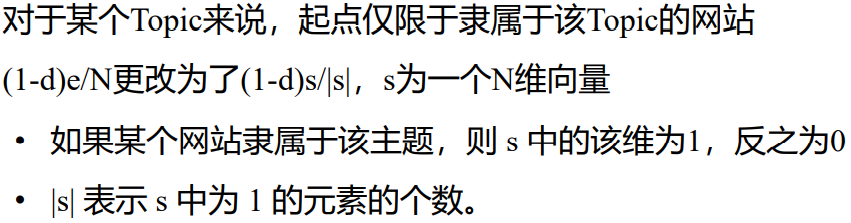


解特征方程（A-I）x=0且=1得概率x。(λ=1)

**PageRank拓展 1.个性化：**



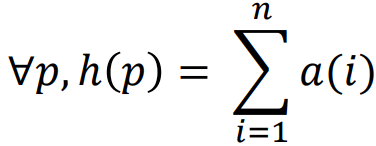
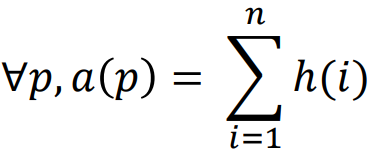
**2. 主题敏感：**仅有同行的评价才最有价值

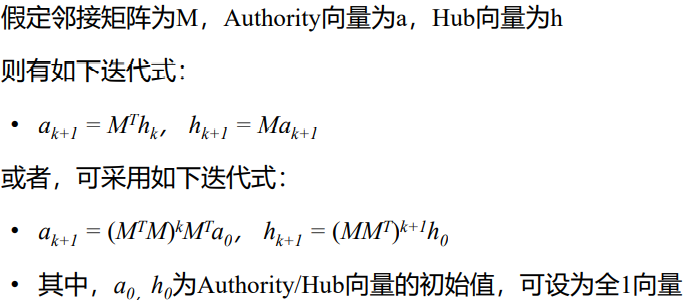


**3. Hiltop算法**：非从属组织网页(从属组织网页：IP前三个字段相同，URL中的主域名段相同),专家页面(包括K个”指向的所有页面相互非从属组织”的出边)，依赖专家页面相关性得分

**PageRank优点：**可离线完成；全局排序；结果通用。**缺点：**主题无关，难区分恶意链接和广告；旧网页得分高；不能单独用，需与相关性排序结合；效率问题。

**HITS：权威网页Authority：**某个领域或话题相关的高质量网页。**中心网页（枢纽Hub）：**指向很多高质量的权威网页。





**每次迭代都要归一化！**

,。

解特征方程（A-λI）x=0且=1得概率x。

**HITS优点:**更好地描述互联网组合特点；主题相关，可单独使用；**缺点：**需要在线计算，时间代价大；链接敏感。

**网页文字处理**

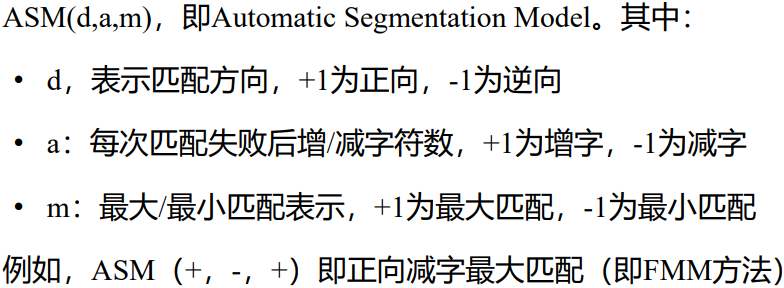
**文本处理：**信息检索基本组件，为后续应用提供支撑。将原始文档转化为词项以建立索引，使面向查询条件的精准的文档匹配成为可能。与查询解析相辅相成。

**词条化处理**

**词条化：**将给定字符序列拆分成一系列子序列的过程

**英文分词**(词组切分？标点符号？专有名词？)；**中文分词**(没有显式分隔符？虚词？分词歧义？**未登录词**（专有名词；专业术语；新词&变异词）？)；专业术语中文字与符号结合？新类型字符序列；多语言混杂。

**分词方法** **基于字符匹配（机械分词）：**



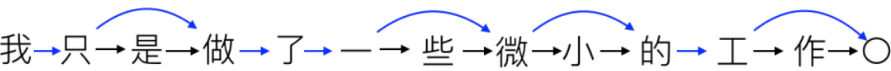
**正向最大匹配FMM：**使用户满意->使用/户/满意

**反向最大匹配RMM：**使用户满意->使/用户/满意

**双向最大匹配：BM**=FMM+RMM(选词数最少)

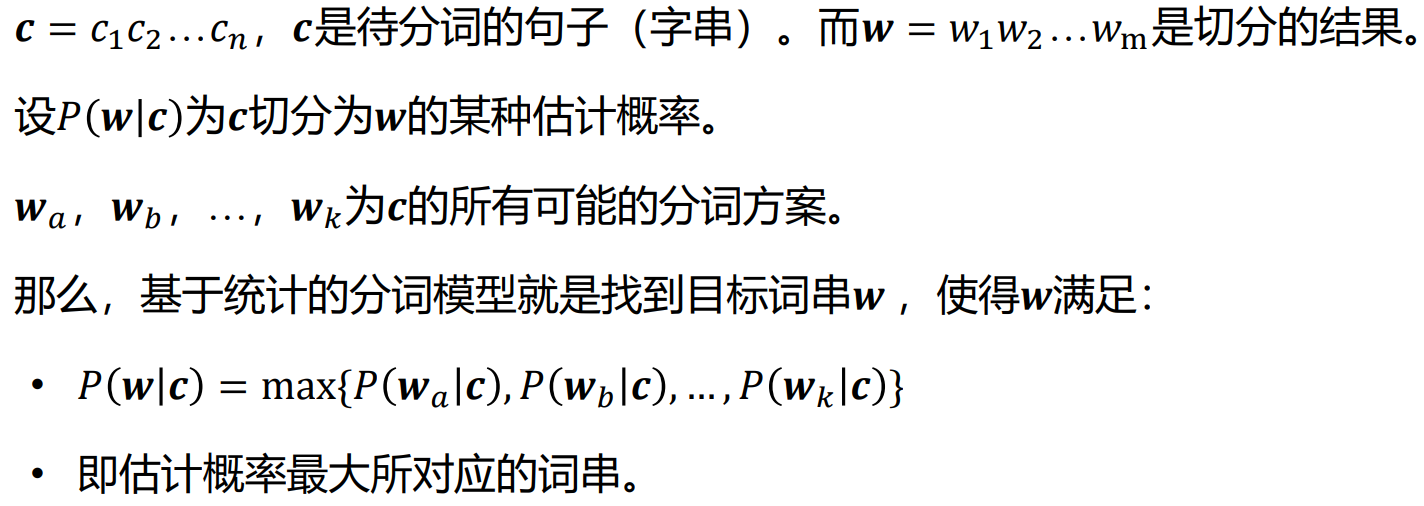
南京市长江大桥->南京市长/江/大桥(FMM)，南京市/长江大桥(RMM)->合并为词数最少的“南京市/长江大桥“

**最少切分分词**（搜索最短路径，边权可视作1，也可根据词频决定，N-最短路径法）；

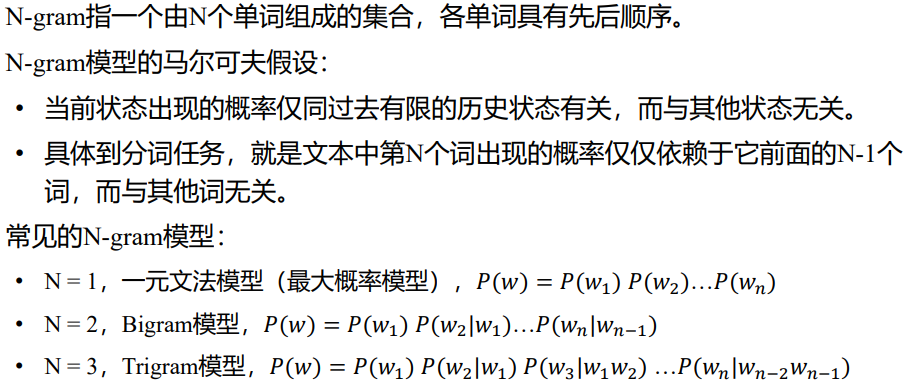


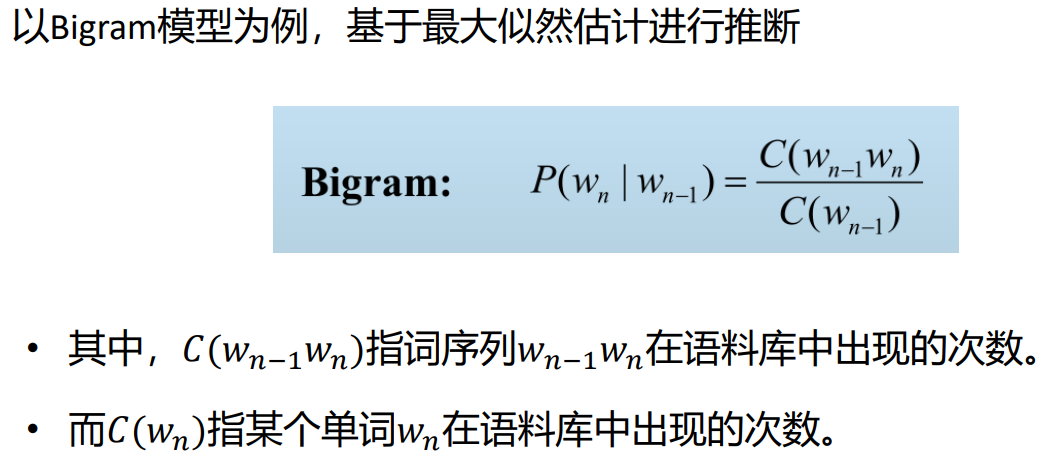
**优点：**效率高、直观；**缺点：**依赖词典（维护词典开支大，难应对新生词汇，词频/重要性不影响结果）

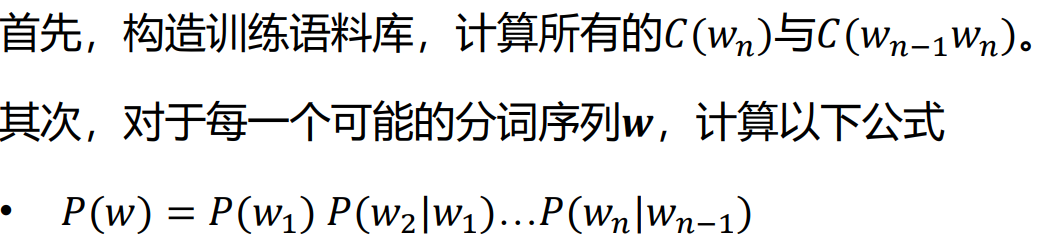
**基于统计：**

(基于统计的分词方法不需要词典，但实际中可以用机械分词方法进行分词，以获得候选的分词集合：既发挥**匹配分词切分速度快、效率高**的特点，又利用了**无词典分词结合上下文识别生词、自动消除歧义的优势**)

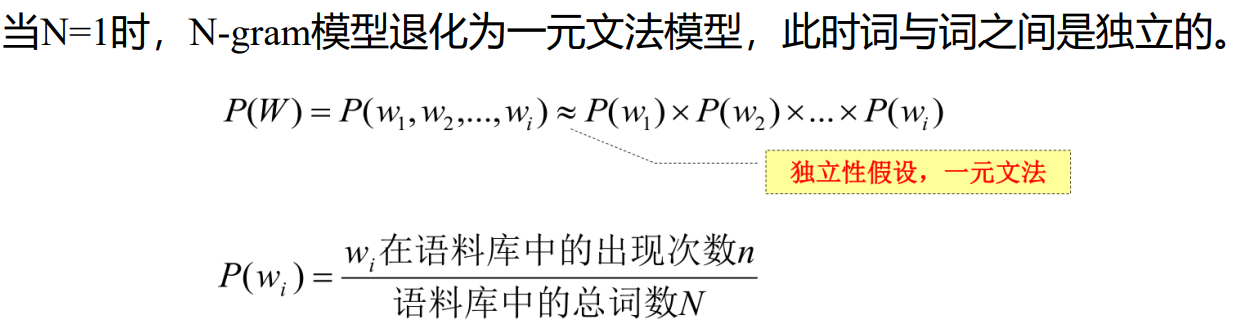
**N-gram模型**



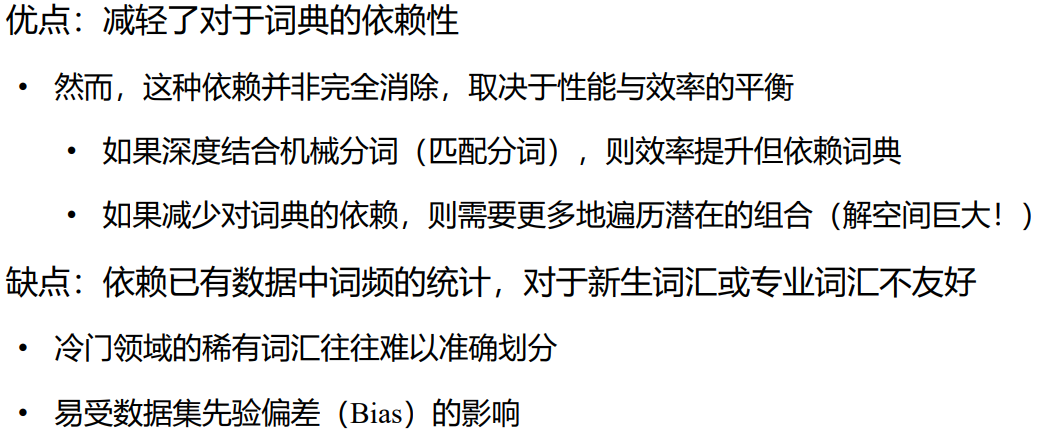








基于统计文法模型的**优缺点**：



**基于序列标注：**

四类标注：B(词的开始)、M(词的中间)、E(词的结束)、S(单字词)。例：中国科学技术大学是中国最好的大学：BMMMMMME S BE BME BE, 中国科学技术大学/是/中国/最好的/大学

**隐马尔可夫模型HMM**（初始状态概率矩阵，隐含状态转移矩阵、观测状态概率矩阵；观测值集合（字符集合），隐藏状态值集合（BEMS））：维特比算法

1. 齐次假设：当前隐藏状态只与上一个状态有关系 2. 观测独立性假设：观测值之间相互独立的，只与生成它的状态有关系。

**条件随机场CRF：**具有长距离依赖性和恶交叠性特征的能力；所有特征可以进行全局归一化，求得全局最优解

**长短时记忆模型LSTM：**利用上下文关系RNN

**停用词处理**

**停用词：**文档中频繁出现或对实际语义影响不大的词语

**停用词识别方法：**文本频率、词频统计、熵计算

**去停用词隐患：**特殊场景下意义/停用词组合有意义

**趋势：压缩技术**降低停用词表存储开支；**引入词项权重**降低高频词影响；**索引去除技术**排除低于权重的词项

**规范化处理**

**归一化/词根化处理:**词干提取stemming(基于规则去除单词前后缀获得词根:复数/时态)->词性还原lemmatisation(基于词典将复杂形态转变成基础形态)

**拼写错误处理：**基于词典或**编辑距离**（两个字符串之间转换所最少需要的编辑操作步数，Dis(“Kitten”,”Sitting”)=3）进行检查和校对

**同义词/相关词处理**:基于人工维护知识库获取词项关系

**网页索引**

**布尔检索：**文档被表示为关键词集合，所有的查询式都被表示为关键词的布尔组合，一个文档当且仅当满足布尔查询式时才能检索出来，检索策略是二值匹配

**关联矩阵：**文档的表示每列都是0/1向量，每个0/1对应一个词项，1(0)表示(不)包含该词。将给定的查询条件转化成行向量计算。高度稀疏

**倒排索引**

词汇表：词项的集合；倒排表PostingList：文档ID列表，列举词项在哪些文档中出现。

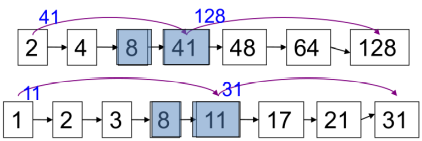
**优点：**搜索快；**缺点：**构建和维护成本高；**Key：**Term。

**建立PL流程：**检索每篇文档获得<词项，文档ID>对，并写入临时索引->对临时索引中的词项进行排序->遍历临时索引，对相同词项的文档ID进行合并

基于倒排表的索引：查询是PL的“合并“

**动态索引问题：**主从索引(大的主索引，新文档存储在小的辅助索引，检索时同时遍历两个索引且合并，利用新的无效位向量删除，定期将辅助索引合并到主索引中)

**优化(1)**(A OR B)AND(C OR D)，先估计(A OR B)和(C OR D)的文档频率并排序，从小到大执行AND **(2)**跳表(改进合并的O(x+y))：利用**跳表指针**快速合并



**扩展：**二元词索引/位置信息索引

**索引存储**

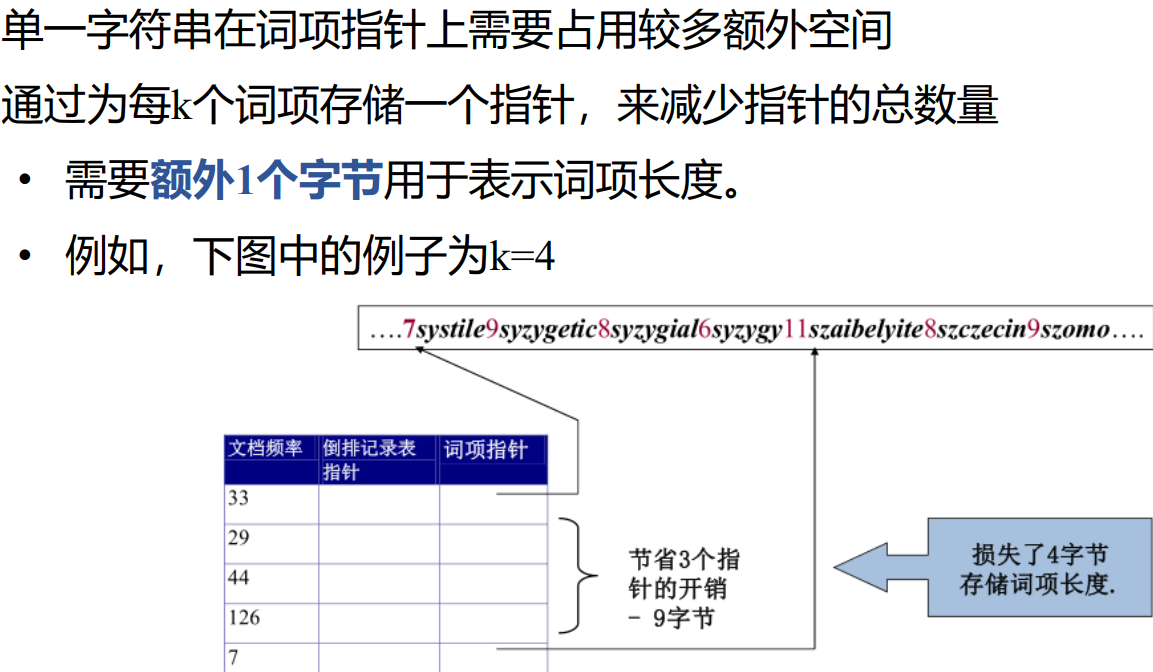
词典与倒排表分开存储（可以支持并行、分布式查询）

词汇表存储：顺序、哈希、B/B+树、Trie树（前缀树）

**索引压缩**

**词典:1.**将词典视作单一字符串（词项间用指针分割）

**2.按块存储**（每k个词项存储一个指针，需要额外1个字节表示词项长度）

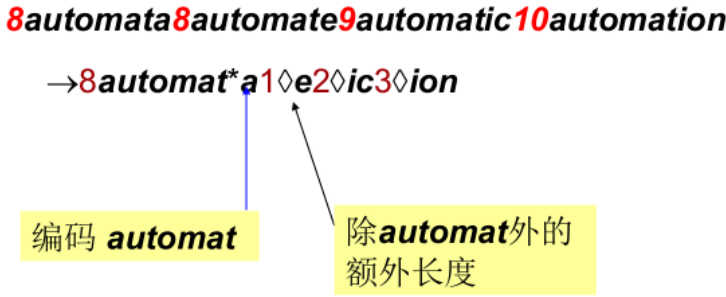


二分查找只能在块外进行，存储空间和查询时间的平衡

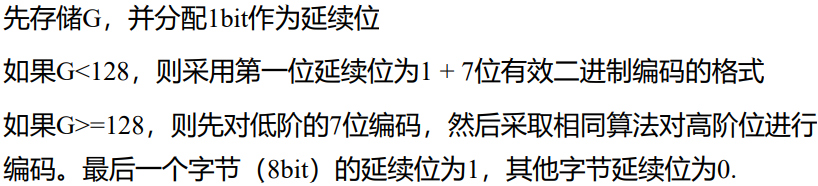
**3. 前端编码**

（公共前缀用

特殊字符表示）



**倒排表：可变长度编码（**第一个存原值，其它**存间距！）**



214577的二进制为1101/0001100/0110001，

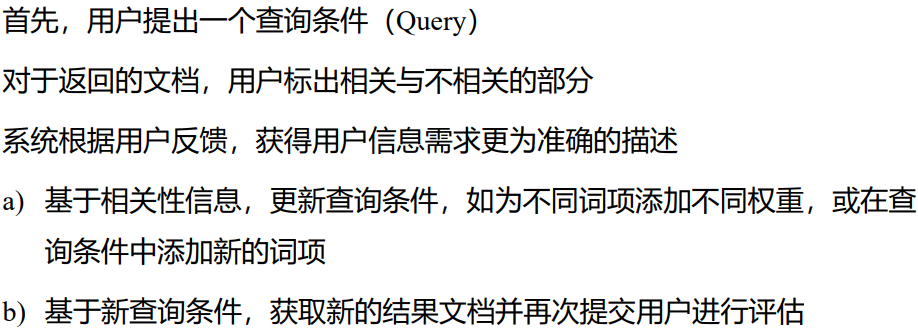
VB编码为00001101/00001100/10110001

**查询**

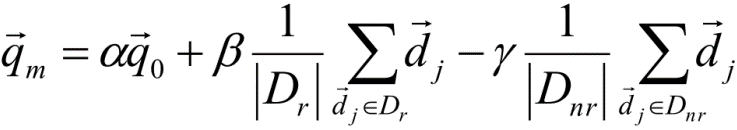
**查询表达的难点：**用户表达的精简性和歧义性；用户表述方式差异性(同义词/方言，信息错漏的误导)；侧重点。

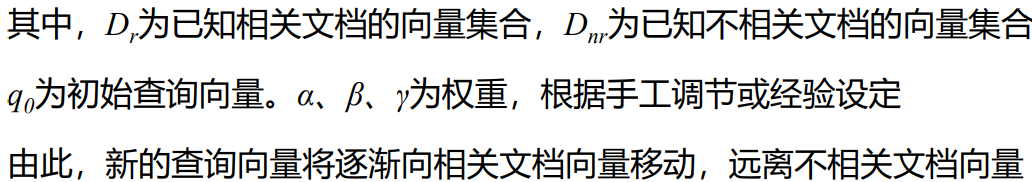
**理解用户查询：**基于查询的自然语言处理；相关性反馈；引导用户表达真实查询意图；借助间接反馈和情景信息

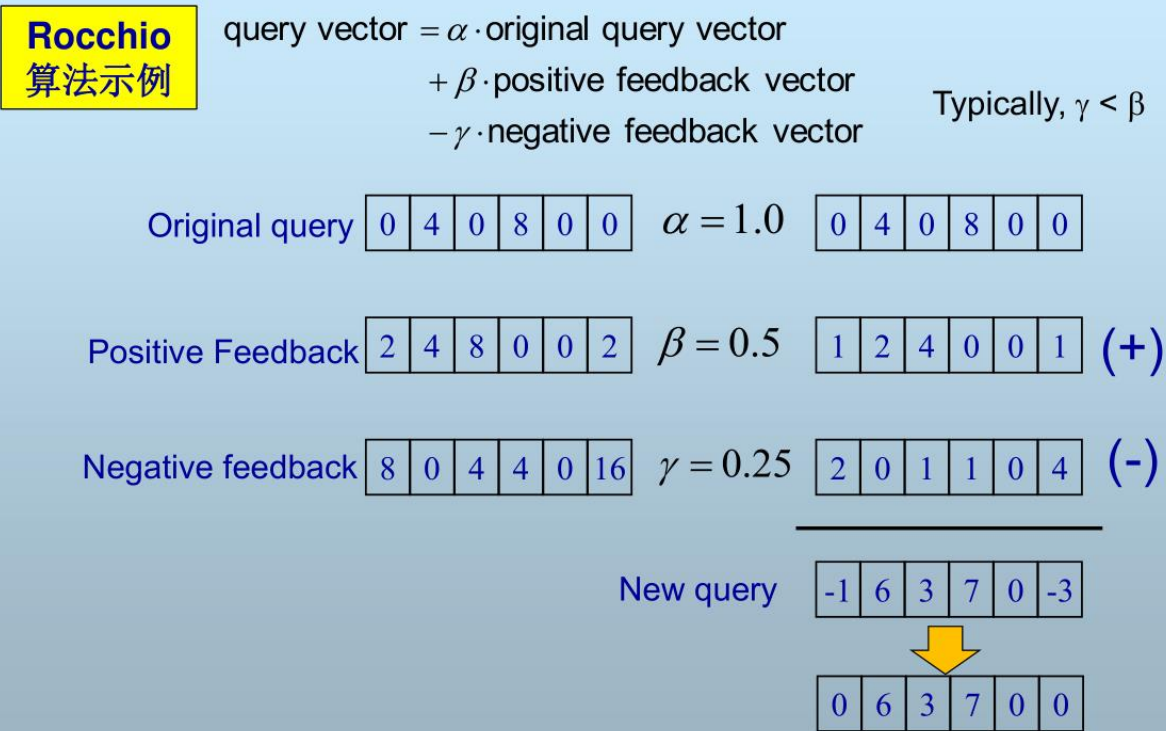
**相关性反馈：**用户在查询后标记文档的相关/不相关，然后迭代更新查询，以获得更好的结果；**流程**如下：



**技术：Rocchio算法**（使查询尽可能离与之相关的文档更近，离与之不相关的文档更远）





（负分量->0）

正反馈（价值更大）vs负反馈（收集难度高）

**反馈分类 显式反馈：**用户显式参与交互过程（用户点击记录只有正样本、用户评论）；**隐式反馈：**系统追踪用户行为来推测返回文档的相关性（鼠标键盘/眼球动作，判断不一定很准确，分析难度高；但省却了用户的显式参与行为，减轻用户负担，提升体验感）；**伪反馈：**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（结果未经用户判断，难以保证其准确性，可能出现查询漂移）。

**查询扩展** 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完整的查询条件

(1)同义词辞典：有助于提升查询召回率，但影响准确率

(2)基于搜索日志进行优化：查询日志，挖掘查询等价类

**上下文感知：**查询重组/查询特化/查询泛化/一般关联

**排序**

**排序的难点：**Web网页的信息组织与内容质量参差不齐，用户庞杂且缺乏知识和经验，用户意图多样差异巨大

**好的排序：**网页内容匹配程度+网页内容质量



**数据聚合 多源数据归并时的问题：**不同属性名称、单位尺度、属性统计方式、不同数据源的统计不一致性

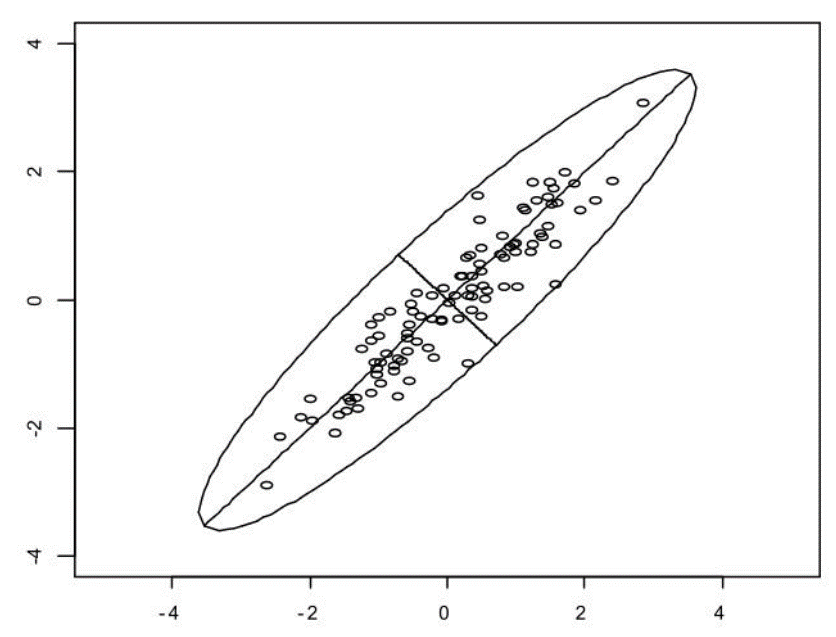
**解决：**(1)换算和汇总 (2)实体对齐

**数据采样** 采样代表性；启发式采样规模确定方法（分组采样，组内数据高度相似，不同组对象差异大，每组至少取一个）

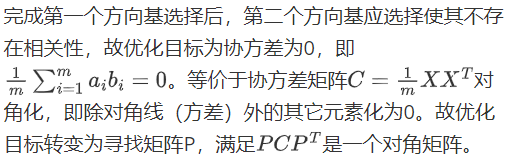
**数据规约**

**维度规约**：删除不具有区分度的特征，可能降低噪声，避免维度灾难的同时使模型更容易理解，也易可视化

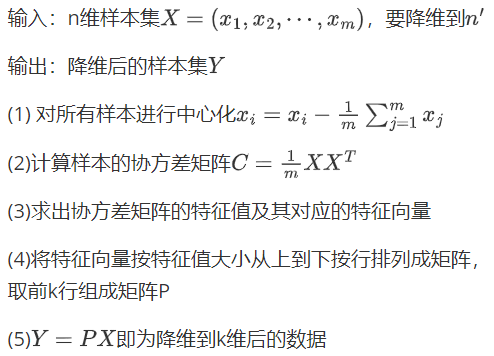
**降维方法1主成分分析PCA**：通过正交变换将一组可能存在相关性的变量转换为一组线性不相关的变量，转换后的这组变量叫主成分。多维椭球：找出主轴与几个最长的轴作为新维度。选择投影方差最大的轴。



**最大特征值对应的特征向量可以最大化投影方差。**



最大K个特征值的特征向量对应的线性组合是K个新指标（K个特征值的比重反映了主成分的信息量，一般>0.85）



**PCA特点** (1)依赖原始变量也只能反映原始变量 (2)PCA内在假设之一是原始变量直接存在一定关联。若原始变量相互独立则降维失败，数据越相关，降维效果越好(3)PCA的结果未必清晰可解释

**降维方法2 特征子集选择**（而不是归纳新特征）去除冗余特征和不相关特征，或为特征赋予不同权值

**数据离散**（将连续属性变换为分类属性）

**(1)二元化**将连续或离散属性转化为多个二元属性0/1

**(2)非监督离散化** 不用类别信息 (等宽/等频率/等深）

**(3)有监督离散化**（基于熵）

先进行二分，选择熵最小的点进行分割。对其中具有较大熵（即纯度不高，信息混乱）的部分再下一轮分割。

**分类**

**基于规则的分类 规则分类器的基本形式：**

规则condition（属性组合，前提）->标签

**规则分类器的基本原理：**

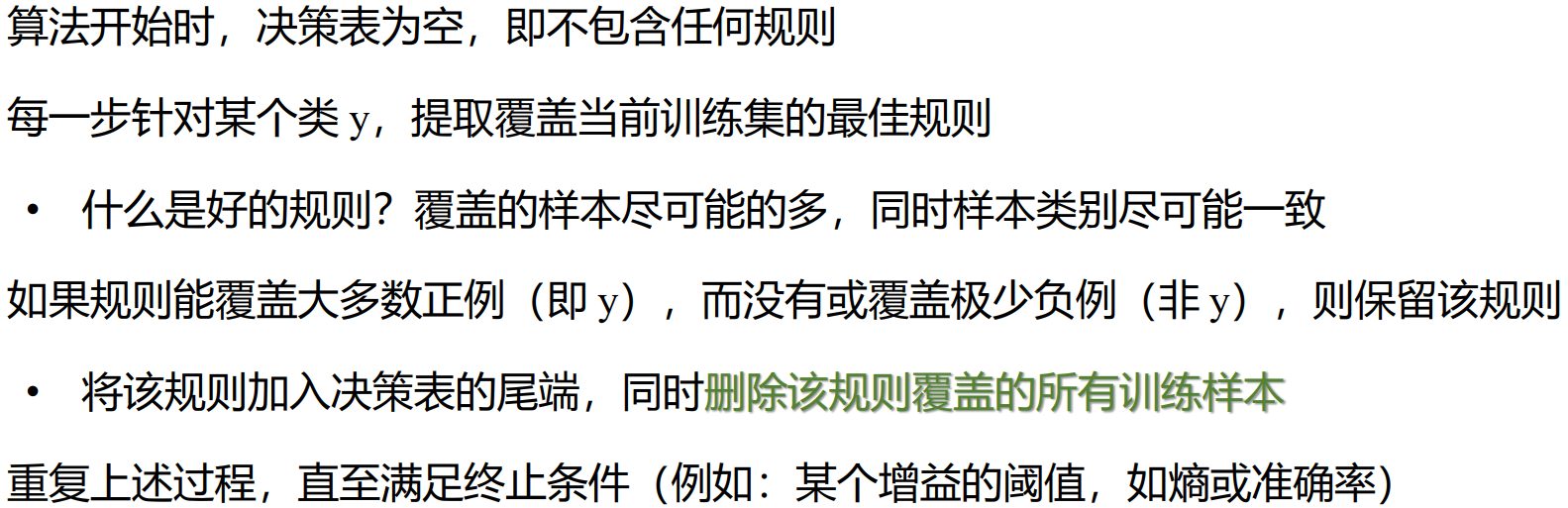
互斥原理（每条记录至多被一条规则所覆盖）&穷举原理（每条记录至少被一条规则所覆盖）

**规则分类器的有序性** 基于规则的排序方案：按照规则的质量（如准确性）进行排序；基于类的排序方案：同类规则排在一起，相对顺序被忽略。

**如何制定规则分类器：**

(1)直接从数据中自动学习规则：RIPPER、CN2

●**顺序覆盖**：贪心



**规则增长策略：**

“从一般到特殊”：初始规则条件为空，给定目标标签y，逐步加入合取项(AND相连)来提高规则质量

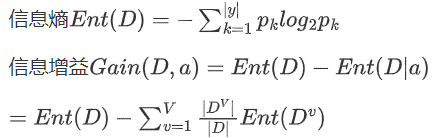
“从特殊到一般”：随机选择一个正样本作为初始种子，逐步删除规则中的合取项，来覆盖更多的同类别正例

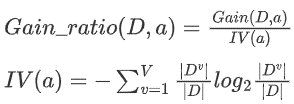
(2)间接借助其他分类模型学习规则：决策树

**基于监督学习的分类**

**决策树：从根节点到叶节点的每一条路径都对应一条分类规则。**

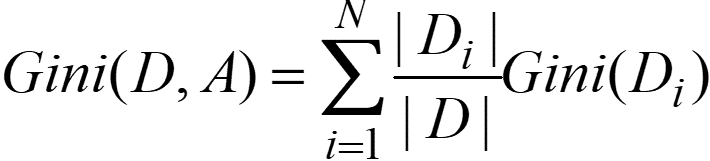
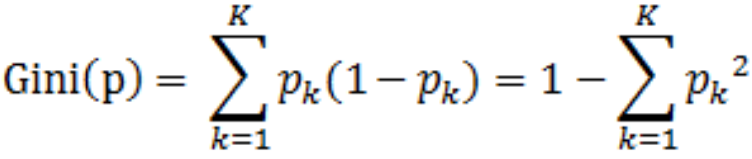
特征选择（有较强区分能力的特征）->生成决策树->决策树剪枝，避免过拟合





**●特征选择准则**(1)信息增益Gain(偏好取值较多的属性) (2)信息增益率Gain\_ratio(引入惩罚项，但倾向于选择取值较少的属性) 先从候选特征中找到信息增益高于平均水平的集合，再从中找到信息增益率最大特征 (3)基尼指数（一个随机样本被分错的概率，Gini指数低则信息纯度高）

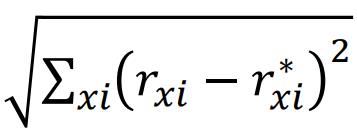
K个类，样本点属于第k类的概率为pk:



例如：

**聚类问题的评估** **非监督评估：**(1)基于邻近度矩阵(理想结果为“簇内点邻近度全为1，簇间为0”，呈对角模式)；(2)凝聚度(簇内两两点的邻近度之和or簇内个点到簇中心的邻近度之和)与分离度(簇间两两点的邻近度之和or簇中心到其他簇中心的邻近度之和) **有监督评估：**(1)分类度量：熵、纯度（簇在多大程度上包含单个类的对象，以最多类的比例计算）、准确率、召回率、F值 (2)相似性度量：分类对应的矩阵，同一类的样本对应的元素为1,不同类为0，比较两矩阵相关性

**推荐系统**

**推荐评估：**(1)评分均方根误差RMSE  (2)分类(推荐正确/错误:Precision/Recall/F；TopN:Pre@N,Rec@N) (3)排序评估

**基于内容的推荐**：物品画像&用户画像(基于用户评分进行加权)采用余弦相似性度量进行评分

优点：用户推荐过程独立，个性化，推荐结果可解释

缺点：难以提取物品特征，难建立新用户画像

**多样化评估：**最大边界相关性MMR

**基于路径推荐：**知识图谱代替向量化画像，基于图谱上的游走实现推荐，路径可作为推荐的依据。

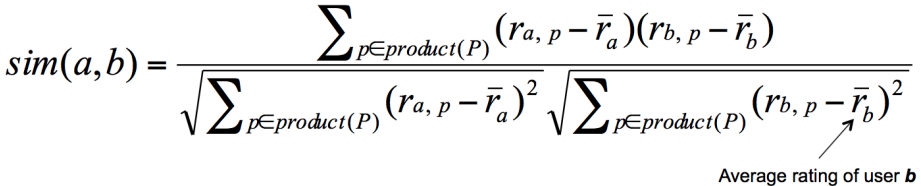
**偏见Bias问题：**位置(第一更受关注)，模态(与众不同的模态吸引关注)，关键词效应(标题党的威力)

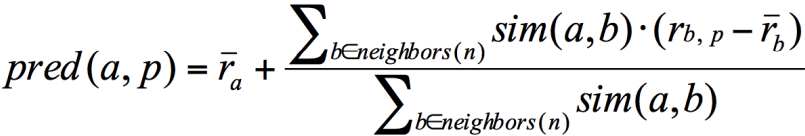
**双向选择问题：**用户<->物品，稳定匹配

**协同过滤 本质：矩阵补全**

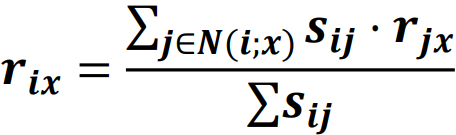
**基于内存(Memory-based)：**

**User-based:** (预测时忽略空值)





**Item-based：**

(无需平均分修正)

相似度计算时，如果未评分，直接设为0，不用减平均数。

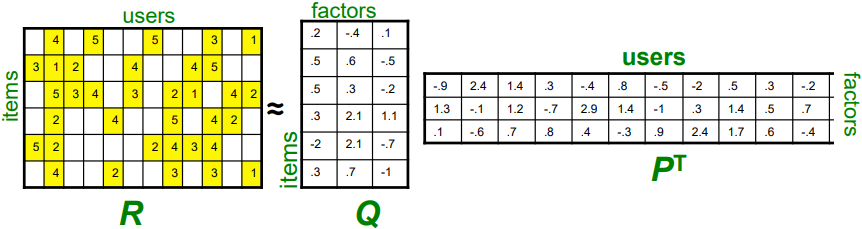
基于物品的推荐效果更好？物品属性单一，用户偏好多样。

基于内存**优点**：不受多模态非结构化信息表征与特征选取困扰；**缺点**：冷启动、稀疏性、热度偏差。

**冷启动问题：**(1)先非个性化推荐 (2)借助个人信息或其它网站浏览信息 (3)诱导式推荐迭代收集用户反馈 (4)与基于内容混合 (5)Side information：众包文本、知识图谱

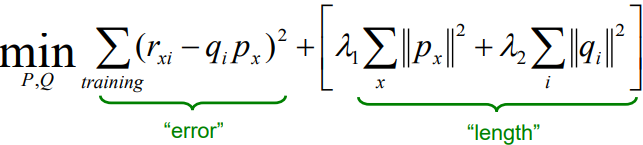
**基于模型(Model-based)：**潜在因子

**矩阵分解Matrix Factorization:**





解决过高的K带来的过拟合？引入正则项，避免过大参数值。



**(1)非负矩阵分解(2)概率矩阵分解(3)社交约束**

**社会网络**

**基本元素：**节点（网络中实体），有向/无向边（关系），邻居/出入度，连通性/连通组件。

**节点角色：**意见领袖、结构洞（作用：为组织引入外部信息，衡量方式：聚集系数（任意两好友也互为好友的概率）低）

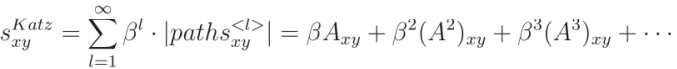
**链接预测** 三元闭包

(1)两节点间存在边的概率，与它们共同好友的个数呈正比

（如何削弱好友个数影响？对指标进行正则化）

(2)共同好友作为中介的引荐力度：还要考虑好友的好友数

(3)考虑“共同好友”的好友：基于多条关系的链接预测



**社团挖掘：**

**边介数：**网络中任意两点的最短路径，有多少条会通过该边

(1)**基于层次聚类**

Girvan-Newman算法：计算网络中所有边的边介数；去除边介数最高的边；重新计算去除边后网络中所有边的边介数；跳至步骤二，直至网络中没有边存在

(2)基于划分聚类：K-means，谱聚类(基于最小割，面向图)

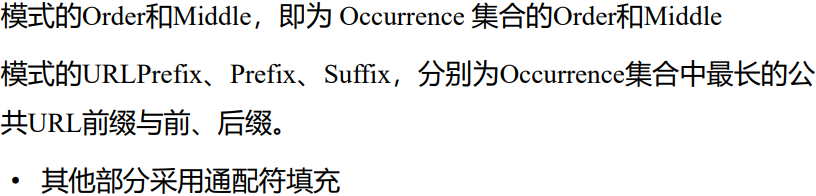
**关系抽取**

**关系抽取方法 (1)基于规则：**纯手工定值规则，文本匹配

**“模板”和“槽”**；难制定规则，领域专家构筑大规模知识库，领域移植困难。

**(2)基于模式：**从种子关系中获得模式，再由模式寻找更多种子，迭代优化。（基于字面匹配，移植性差，适合特定关系）

**双重迭代模式关系提取DIPRE:**



**Snowball:**仅信任支持度(满足每个模式的元组数量)和置信度高(符合该模式的元组确实符合相应关系的概率)的模式

**(3)基于机器学习：**转化为分类问题，训练模型求解

**开放关系抽取** (1)基于知识监督 (2)基于句法

**远程监督** 如果某个实体对之间具有某种关系，那么所有包含这个实体对的句子都是用于描述这种关系；目的：获取足够数量的、高质量的标注

**语义漂移:**不是所有包含该实体对的句子都表达该关系且错误不断放大 **优化:动态转移矩阵**描述各类相互标错的概率;**规则学习**设计相应否定模式列表去除错误的标签;**注意力机制**

**事件抽取**的模板：选定相应模板后，通过事件元素(事件参与者)与事件元素角色（事件元素在事件中充当的角色）的识别，将相应的元素填入模板合适的**槽**（描述命名实体基本信息，内容可包括名称/类别/种类）内

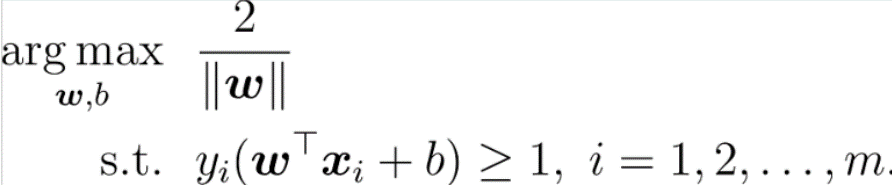
**●决策树生成** 计算当前结点各个属性的信息增益率，基于最大信息增益率属性，迭代式对节点进行分类，直到某个节点的样本类别统一

**●剪枝** 预剪枝：每个节点划分前衡量当前节点的划分能否提高决策树的**泛化能力**（验证集上精度）

后剪枝：自底向上考虑每个非叶子节点被替换成叶子节点后能否提高泛化性能

**最近邻分类** 计算与未知样本与其他样本的距离找到K-最近邻，基于K-最近邻的类别确定分类结果。**动机：**表征空间上相似的文档是相似的；基于实例的学习，不需要对数据进行抽象（如提取特征）；消极学习，不需要模型，但开销大；基于局部信息判别，受噪声影响大。**距离度量：**欧式距离，汉明距离（0/1向量 统计多少维数字不同），余弦相似度，马氏距离，无穷范数（分量最大值） **K**小则易受噪声干扰，**K**大则错误涵盖其他类别样本。

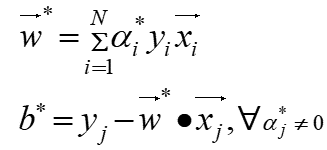
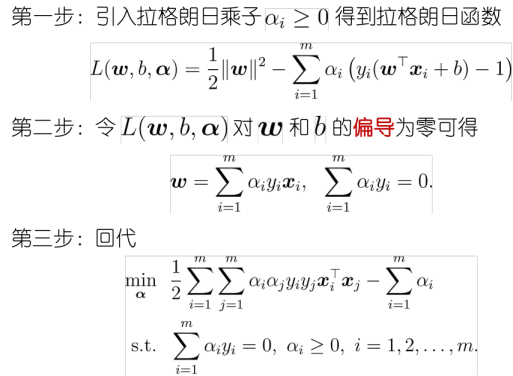
**支持向量机SVM：**二分类转化为寻找最大间隔超平面，实现对高维空间中节点进行有效分割，使得**超平面**和**支持向量**（离该超平面最近的点）的间隔最大化



求解对偶问题得到

最优解

计算超平面：

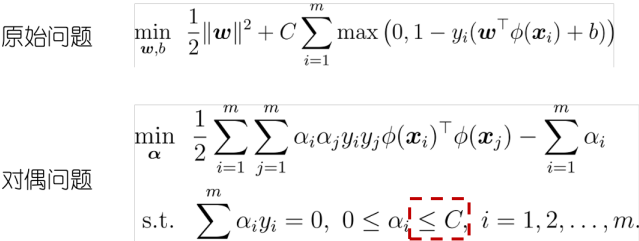




**求解对偶问题即求极值时:**(1)内部偏导均为0,解方程组,检查解是否非负(2)边界也可取极值(比如单一分量为0)

**核函数：**将高维空间下的内积运算转化为低维空间下的核函数计算。

**软间隔：**允许少数样本不满足超平面约束（引入惩罚项C）



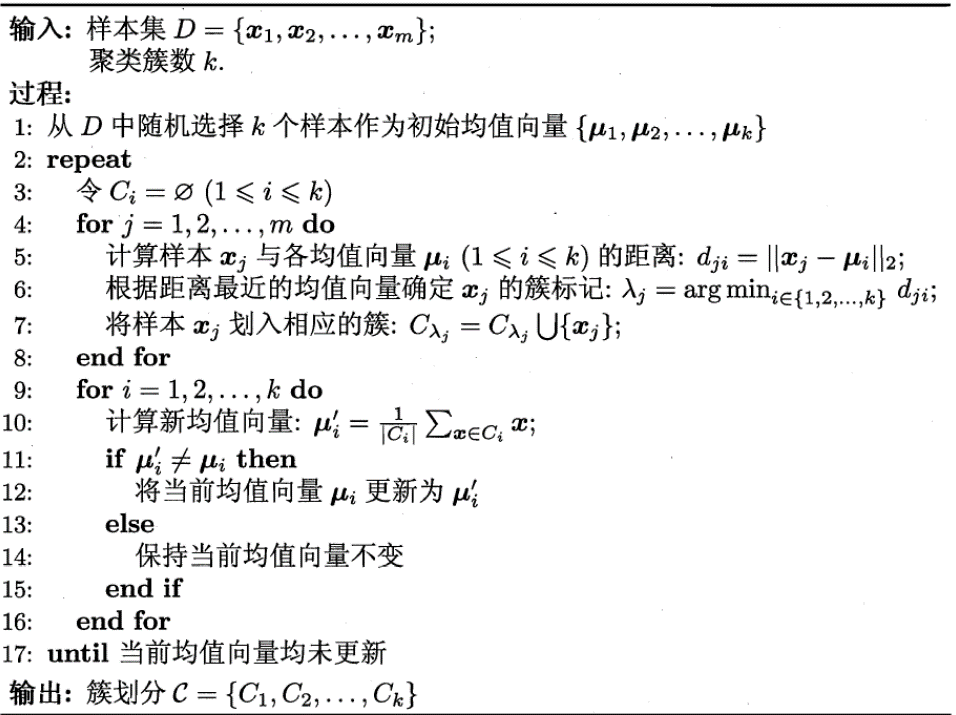
**不平衡分类问题解决方案：(1)代价敏感学习：**代价矩阵衡量将一个类错分到另一个类的代价，优化目标由准确/召回变为加权后的代价 **(2)抽样：**少数类过采样、多数类欠采样、K-最近邻用少数类样本生成新样本

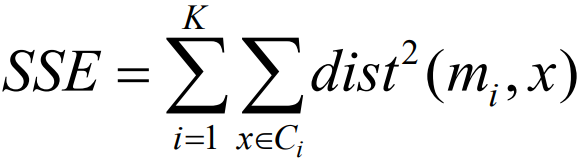
**聚类**

**基本问题：**聚类依据？相似性度量？簇的数量？

**方法：**层次聚类/划分聚类

**K均值聚类(K-means)**



（mi为簇Ci的中心）

**初始中心的选择**(1)K均值的后处理：清除较小的可能代表离群点的簇，对SSE高的簇拆分，对SSE低的合并(2)**二分K均值聚类**：先分为2个簇，再不断选择一个分裂

**空簇的处理：**将最大SSE的簇拆分，或选择一个最远样本点新生成一个簇。

**缺点：**易受离群点干扰，当簇存在不同规模、密度及不规则形状的情况下，K均值聚类效果较差。

**层次聚类**

**优点：**不需预设簇的数量，结果有意义对应到分类学目录上；**缺点：**局部最优，每步合并决策都是最终的

**凝聚式聚类：**(1)合并邻近度最高的两个簇 (2)基于更新的簇重新计算邻近度，更新邻近度矩阵

**邻近度**(1)**单链MIN**：不同簇最近的点之间的邻近度（擅长处理非椭圆形状的簇，但对噪声比较敏感） (2)**全链MAX**：不同簇最远的点之间的邻近度（对噪声不太敏感，但可能使得较大的簇变得支离破碎） (3)组平均：所有来自不同簇的两点之间的平均邻近度 (4)中心距离：两个簇中心之间的邻近度 (5)沃德法：合并后簇中各点到新中心的距离平方和。

**分裂式聚类：**(1)二分K均值聚类 (2)最小生成树聚类：由差异矩阵生成一颗最小生成树(节点之间权值最小)，每步断开差异最大的一条边，从而创建一个新的簇。

MST聚类结果与单链凝聚聚类的结果相同。

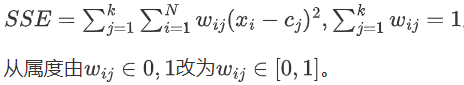
**基于密度聚类**

**密度**：样本一定半径的样本数量

**DBSCAN算法** (1)将所有点区分为核心点(半径内样本数超过阈值的点）、边界点(非核心点但处于稠密区域边界内/上)和噪声点(处于稀疏区域的点) (2)删除噪声点 (3)将所有距离在预定半径内的核心点之间连一条边 (4)连通的核心点形成一个簇 (5)将所有的边界点指派到一个与之关联的核心点所在的簇中

**优点：**对噪声鲁棒；**缺点：**密度变化大的簇受影响

**模糊聚类** (计算归属改为计算从属度)



**数据准备**

**数据挖掘方法：(1)分类：**有监督学习，面向预定义的类别 **(2)聚类：**没有预先定义的类别，借助相似度度量自动生成 **(3)关联规则 (4)异常检测**

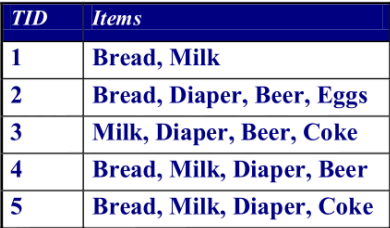
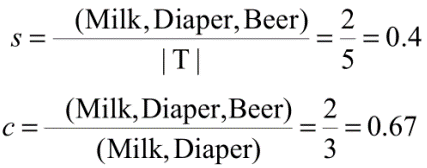
**事务型数据Transaction data**：一条记录对应一个项目(Item)集合(无序)

**关联规则：**分析事务型数据，从而根据一部分项的存在记录来判断另一部分项目是否同时存在于事务中

基本形式：A->B，A、B均为集合

**支持度**为{A+B}在全体事务中的比重s(A->B)=，**置信度**{A+B}占A出现的事务中的比重 c(A->B)=。

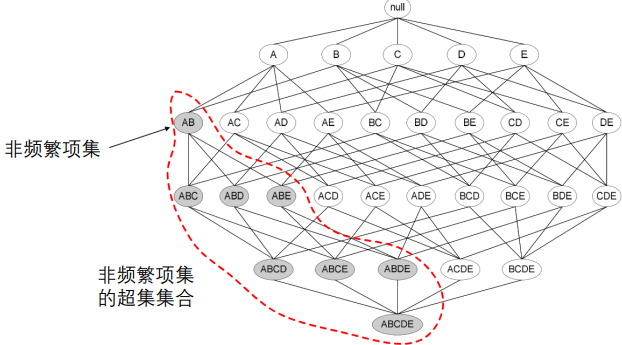
例：考虑{Diaper,Milk->Beer}关联规则



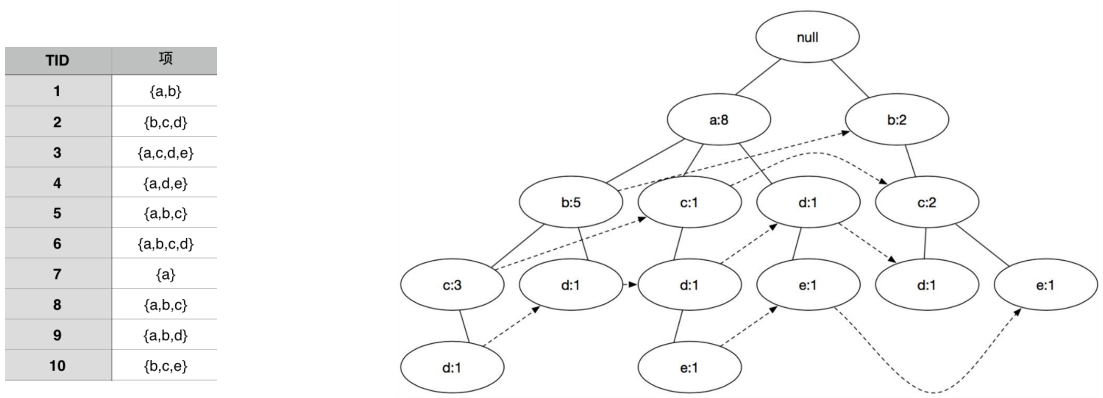
**频繁项集**：支持度高于**阈值**的项目集合A

**频繁项集生成方法** (1)最基本的：穷举所有可能集合，计算支持度，繁杂度过高！

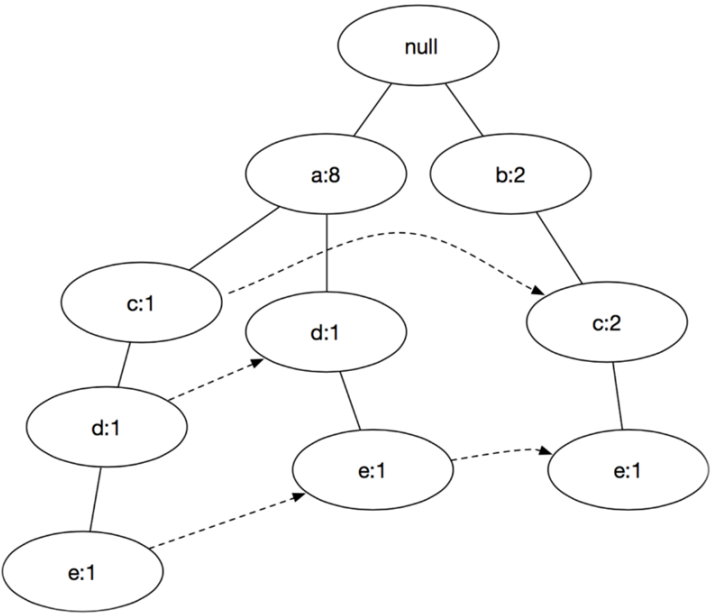
**(2)Apriori**（剪枝思想，但仍是”生成-测试”范式）：逐步减去所有的非频繁项集，然后基于频繁项集生成其超集。（先验原理：如果一个项目集是频繁的，那么它的所有子集也是频繁的。则非频繁项集的所有超集也是非频繁的。）



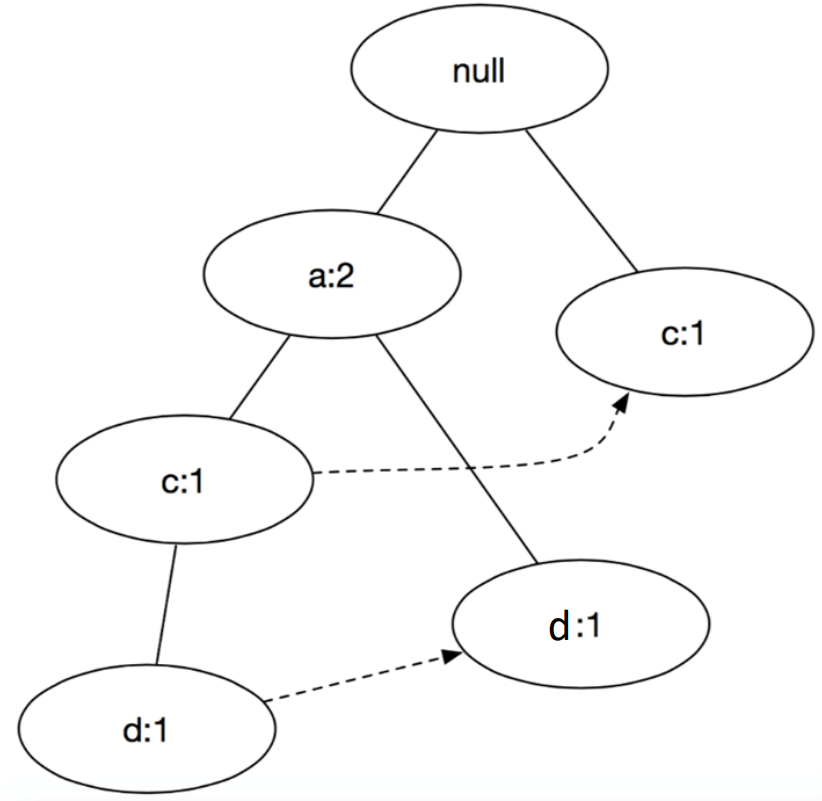
**(3)FP-Growth：**本质是输入数据的压缩表示，通过逐个读入事务，并将事务映射到FP树中的某条路径来构造。●建树：对各个项按支持度排序，将排序后的项集逐步读入并建立树状结构，对相同项节点采用指针连接，方便快速访问。



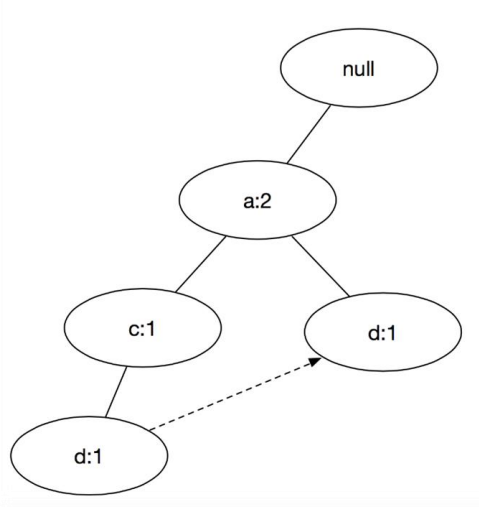
●基于FP-树，生成频繁项集：◇FP-Growth本质上是自底向上的探索。首先查找以e为结尾的频繁项集，其次是d/c/b，最后是a，包含e的路径如下：



◇判定e本身是否为频繁项集（此处设阈值为2，高于阈值），将e的前缀路径转化为条件FP树（需要更新路径上的支持度计数，只有包含e的事务会被统计），删除那些非频繁的项(例如b)



◇考虑更长结尾的频繁项集的子问题，以{de}为结尾的频繁项集判定为例，在前一张图上统计与d相关的支持度求和为2，为频繁项集；以de为结尾，得到前缀路径如下，通过其条件FP树发现{ade}支持度为2，也频繁。



**异常检测：**异常数据≠错误数据，而是包含不同寻常规律的数据。

**(1)基于分布：**前提是识别数据集的具体分布，错误识别会导致错误检测；基于一元正态分布的**离群点判定**:已知参数的前提下根据正态分布判定离群的概率，舆情监测。

**(2)基于度量：**基于距离或密度的方式，检测空间中远离大多数数据点的离群点。●求数据点到K最近邻的平均距离：高于阈值则判定为异常点 ●基于密度：采用K近邻距离的倒数作为密度或给定半径内点的个数

**(3)基于聚类**●抛弃远离其他簇的小簇，但簇的个数将影响结果。●先聚类所有对象，再评估对象属于簇的程度（点到簇中心的距离或相对距离）

**数据预处理 数据质量问题：**(1)数据测量、采集等过程中出现的错误(2)噪声、离群点 (3)缺失数据(->删除和填补并重) (4)重复数据(多源数据归并:实体歧义,多马甲账号)

