“搜你所想”之用户搜索意图识别

桂洪冠 AI前线 2017-05-15

本文是“达观数据技术主题月“5 月 12 日大数据杂谈社群分享内容整理。

大家晚上好，很高兴在这个美好的周末的晚上来到这里给大家做分享~ 也非常感谢 infoq 平台提供的这次机会，分享我们达观数据在垂直搜索引擎，尤其在用户查询意图分析挖掘方面的一些经验。

首先自我介绍一下，我是达观数据的联合创始人 & 技术副总裁桂洪冠，目前主要负责面向企业客户的垂直搜索引擎产品的技术研发、架构设计和效果优化等工作。原先曾担任新浪微博广告系统架构师，和阿里巴巴国际搜索系统架构师和技术专家职务。

人类自诞生以来就伴随着各种信息的生产和获取，如今这个信息爆炸的 DT 时代，人们更是被各种信息所包围。我们知道，人获取信息的方式主要有被动获取和主动获取两种，其中被动获取就是推荐的方式、主动获取就是搜索的方式。

获取信息是人类认知世界、生存发展的刚需，搜索就是最明确的一种方式，其体现的动作就是“出去找”，找食物、找地点等，到了互联网时代，搜索引擎（Search Engine）就是满足找信息这个需求的最好工具，你输入想要找的内容（即在搜索框里输入查询词，或称为 Query），搜索引擎快速的给你最好的结果，这样的刚需催生了谷歌、百度这样的互联网巨头。

上周我们达观的同事于敬老师分享了达观在智能推荐引擎的建设方面的相关经验心得，本次分享结合达观在垂直搜索引擎建设方面的经验，主要围绕以下内容展开：

1. 用户搜索意图的理解及其难点解析
2. 如何进行用户搜索意图理解
3. 达观数据用户搜索意图理解引擎介绍

用户搜索意图的理解及其难点解析

首先，让我们从偏技术的角度来看看搜索引擎发展的几个阶段：

第一个阶段，使用倒排索引解决匹配的效率问题，使用文档模型解决基本的相关性，使搜索引擎变得可用、可扩展，代表比如 Infoseek 。但这一阶段只保证了基本的文字相关性，搜索的真正效果是无法保证的。

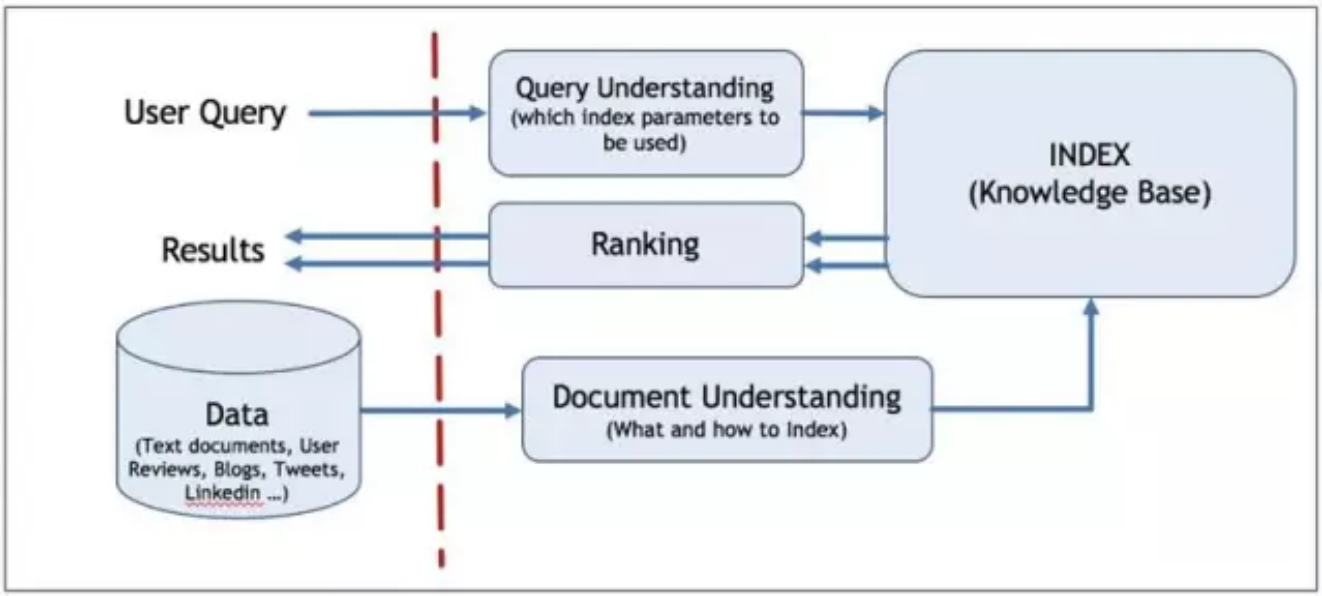
第二个阶段，使用超链模型，比如谷歌的 pagerank 和百度的超链分析。解决权威性问题，使搜索质量提升一个档次。从这一阶段搜索引擎开始快速普及与并进入商业化，为谷歌和百度这样的公司带来了丰厚的利润。

第三个阶段，一方面使用更复杂的规则和机器学习排序模型，综合考虑了用户的行为特征，如商品评论、点赞、收藏、购买等，使得搜索引擎的结果再次提升一个档次，这些在电商等垂直搜索上表现的会更加明显；另一方面，基于各种先进的自然语言处理技术，充分挖掘用户搜索行为日志，对 query 进行分析改写以召回更多更好的结果。

第四个阶段，从“有框”搜索时代步入更加人工智能的“无框”搜索时代。人机交互方式也将更多的是问答式的自然语言加语音的方式，而搜索引擎也更像一个智能机器人，理解人的自然语言问题，提供更加直接有效的知识和答案。这一阶段目前尚处于起步阶段，谷歌、Amazon 以及一些优秀的创业公司都在进行积极的探索。

搜索引擎涉及的技术非常的繁复，既有工程架构方面的，又有算法策略方面的。综合来讲，一个搜索引擎的技术构建主要包含三大部分：

1. 对 query 的理解
2. 对内容（文档）的理解
3. 对 query 和内容（文档）的匹配和排序



我们今天主要探讨其中的 Query Understanding，即对 query 的理解。对 query 的理解， 换句话说就是对用户搜索意图的理解。先看垂直搜索中的一些例子：

“附近的特价酒店”

“上海到扬州高速怎么走”

“小龙虾最新报价”

“华为最新款手机”

“水”

这几个例子都不能直接根据 query 的字面意思去搜索，而是要理解用户输入文字背后的真实意图。不过要准确理解 query 背后的用户搜索意图可不是那么容易的。

我们来分析一下理解用户搜索词背后的真实意图识别存在哪些**难点**：

1. 用户输入不规范，输入方式多样化，使用自然语言查询，甚至非标准的自然语言。比如上面提到的“附近的特价酒店” 、“上海到扬州高速怎么走”都是自然语言查询的例子，又如 “披星（）月”、“吾尝终日而思矣, 下面 “
2. 用户的查询词表现出多意图，比如用户搜索“变形金刚”，是指变形金刚的电影还是游戏？ 搜索“仙剑奇侠传”是指游戏还是游戏软件？ 电影？ 小说？ 电商网站搜索“水”

是指矿泉水？还是女生用的护肤水？

3. 意图强度，表现为不同用户对相同的查询有不同的需求强度。比如：宫保鸡丁。宫保鸡丁菜，菜谱需求占 90%。宫保鸡丁歌曲，歌曲下载需求占 10%。又比如：荷塘月色。荷塘月色歌曲，歌曲下载需求占 70%。荷塘月色小区，房产需求占 20%。荷塘月色菜，菜谱需求占 10%。
4. 意图存在时效性变化，就是随着时间的推移一些查询词的意图会发生变化。比如：华为 P10 国行版 3 月 24 日上市。3 月 21 日的查询意图：新闻 90%，百科 10%3 月 24 日的查询意图：新闻 70%，购买 25%，百科 5%5 月 1 日的查询意图：购买 50%，资讯 40%，其他 10%5 年以后的查询意图：百科 100%
5. 数据冷启动的问题，用户行为数据较少时，很难准确获取用户的搜索意图。
6. 没有固定的评估的标准，CTR、MAP、MRR、nDCG 这些可以量化的指标主要是针对搜索引擎的整体效果的，具体到用户意图的预测上并没有标准的指标。

到这里，第一部分：用户搜索意图的理解及其难点解析 就讲完了。回顾一下，首先我们从相对技术的角度分析了搜索引擎发展的四个主要阶段，接着分析了搜索引擎的三个主要构件：对 query 的理解、对内容的理解、匹配和排序，最后对用户搜索意图理解的难点进行了分析。

如何识别用户搜索意图

首先我们来看一下用户搜索意图有哪些分类。一般把搜索意图归类为 3 种类型：导航类、信息类和事务类雅虎的研究人员在此基础上做了细化，将用户搜索意图划分如下类别：

1， **导航类**：用户明确的要去某个站点，但又不想自己输入 URL，比如用户搜索“新浪网“

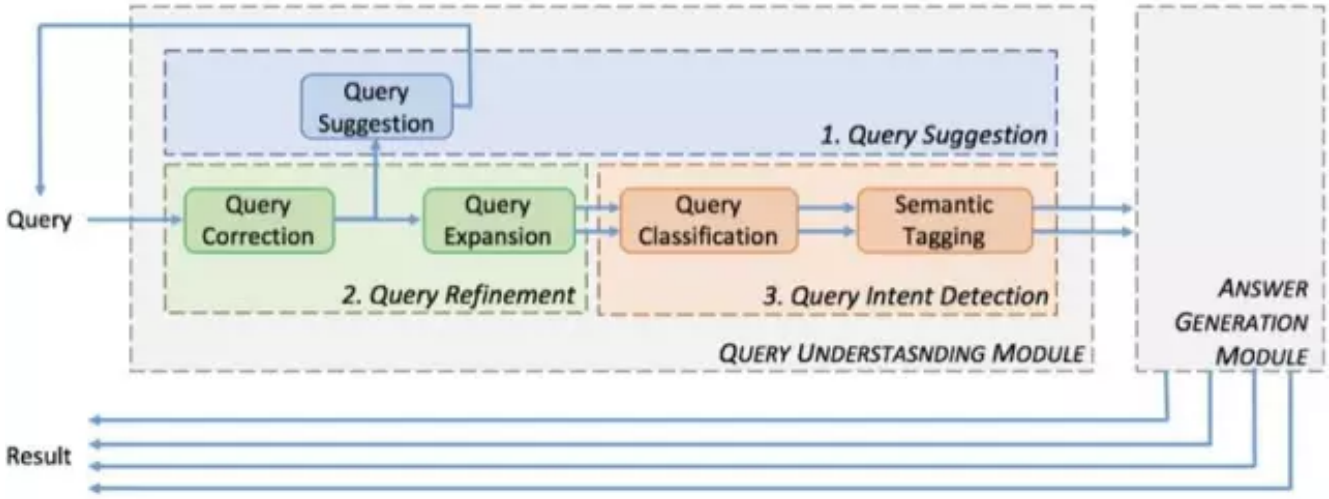
2， **信息类**：又可以细分为如下几种子类型，

直接型：用户想知道关于一个话题某个方面明确的信息，比如“地球为什么是圆的”、“哪些水果维生素含量高”。**间接型**：用户想了解关于某个话题的任意方面的信息，比如粉丝搜索“黄晓明”。**建议型**：用户希望能够搜索到一些建议、意见或者某方面的指导，比如“如何选股票”。**定位型**：用户希望了解在现实生活中哪里可以找到某些产品或服务，比如“汽车维修”。**列表型**：用户希望找到一批能够满足需求的信息，比如“陆家嘴附近的酒店”。

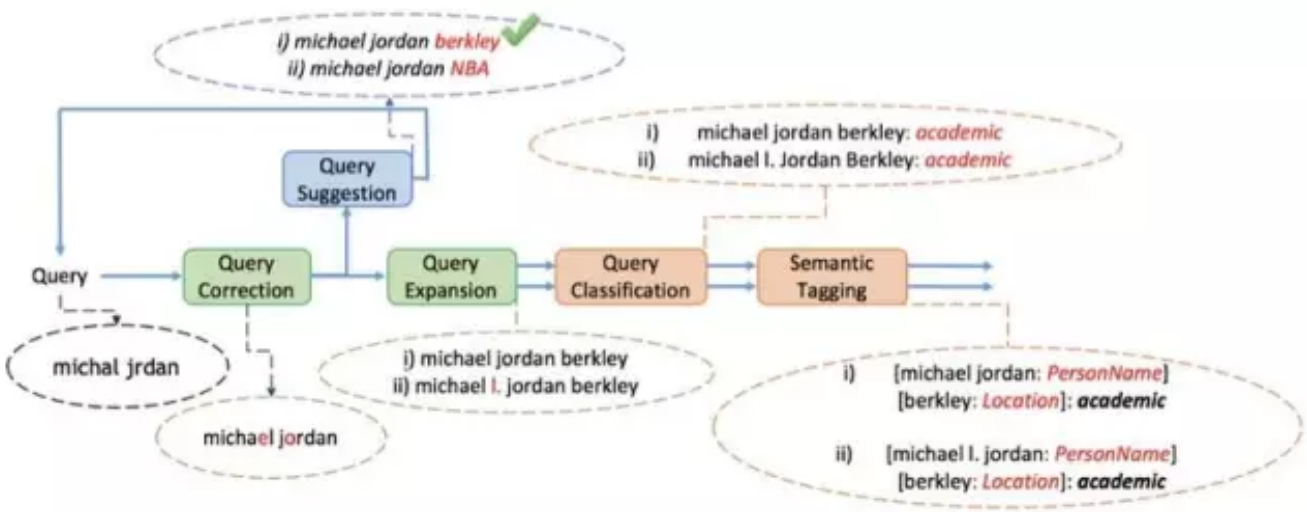
3， **资源类**：这种类型的搜索目的是希望能够从网上获取某种资源，又可以细分为以下几种子类型，

下载型：希望从网络某个地方下载想要的产品或者服务，比如“USB 驱动下载”。**娱乐型**：用户出于消遣的目的希望获得一些有关信息，比如“益智小游戏”。**交互型**：用户希望使用某个软件或服务提供的结果，用户希望找到一个网站，这个网站上可以直接计算房贷利息。**获取型**：用户希望获取一种资源，这种资源的使用场合不限于电脑，比如“麦当劳优惠券”，用户希望搜到某个产品的折扣券，打印后在现实生活中使用。

查询意图理解的过程就是结合用户历史行为数据对 query 进行各种分析处理的过程，包括查询纠错、查询词自动提示、查询扩展，查询自动分类、语义标签等。



下面这张图是一个具体的例子说明 query understanding 的工作过程：



稍微解释一下这张图：

1. 用户的原始 query 是 “michal jrdan”
2. Query Correction 模块进行拼写纠错后的结果为：“Michael Jordan”

3. Query Suggestion 模块进行下拉提示的结果为：“Michael Jordan berkley”和“Michael Jordan NBA”，假设用户选择了“Michael Jordan berkley”
4. Query Expansion 模型进行查询扩展后的结果为：“Michael Jordan berkley”和“Michael I. Jordan berkley”
5. Query Classification 模块进行查询分类的结果为：academic
6. 最后语义标签（Semantic Tagging）模块进行命名实体识别、属性识别后的结果为：[Michael Jordan: 人名][berkley:location]:academic

下面我们逐一的来看看这面这些模块内部细节。

首先看一下 Query Correction 模块，也即查询纠错模块。

对于英文，最基本的语义元素是单词，因此拼写错误主要分为两种：一种是 Non-word Error，指单词本身就是拼错的，比如将“happy”拼成“hbppy”，“hbppy”本身不是一个词。另外一种是 Real-word Error，指单词虽拼写正确但是结合上下文语境确是错误的，比如“two eyes”写成“too eyes”，“too”在这里是明显错误的拼写。

而对于中文，最小的语义单元是字，往往不会出现错字的情况，因为现在每个汉字几乎都是通过输入法输入设备，不像手写汉字也许会出错。虽然汉字可以单字成词，但是两个或以上的汉字组合成的词却是更常见的语义元素，这种组合带来了类似英文的 Non-word Error，比如“大数据”写成“大树据”，虽然每个字是对的，但是整体却不是一个词，也就是所谓的别字。

query 纠错的具体方案有：

1. 基于编辑距离
2. 基于噪声信道模型

先看看编辑距离的方法

Error	Candidate Correction	Correct Letter	Error Letter	Type
acress	actress	t	-	deletion
acress	cress	-	a	insertion
acress	caress	ca	ac	transposition
acress	access	c	r	substitution
acress	across	o	e	substitution
acress	acres	-	s	insertion
acress	acres	-	s	insertion

编辑距离包括删除（del）、增加（ins）、替换（sub）和颠倒（trans）四种转移操作，对应四种转移矩阵。这些转移矩阵的概率可以通过对语料库统计大量的正确词和错误词对来得到。

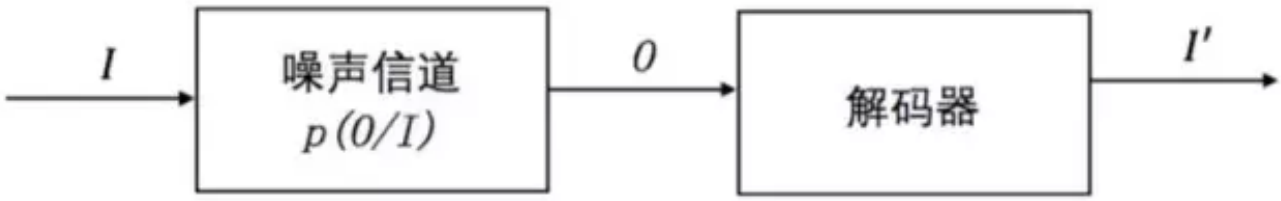
```
del[x,y]:    count(xy typed as x)
ins[x,y]:    count(x typed as xy)
sub[x,y]:    count(x typed as y)
trans[x,y]:  count(xy typed as yx)
```

$$P(x|w) = \begin{cases} \frac{\text{del}[w_{i-1},w_i]}{\text{count}[w_{i-1}w_i]}, & \text{if deletion} \\ \frac{\text{ins}[w_{i-1},x_i]}{\text{count}[w_{i-1}]}, & \text{if insertion} \\ \frac{\text{sub}[x_i,w_i]}{\text{count}[w_i]}, & \text{if substitution} \\ \frac{\text{trans}[w_i,w_{i+1}]}{\text{count}[w_iw_{i+1}]}, & \text{if transposition} \end{cases}$$

中文词语的编辑距离转换存在较大的转义风险。比如，雷锋 ->雷峰塔。中文纠错通常以拼音为基础，编辑距离等其他方式为辅的策略。

基于噪声信道模型的纠错方法：

噪声信道模型（Noisy Channel Model）最早是香农为了模型化信道的通信问题，在信息熵概念上提出的模型，目标是优化噪声信道中信号传输的吞吐量和准确率。对于自然语言处理而言，噪声信道模型如下图，其中 I 表示输入，O 表示经过噪声信道后的输出，I’ 表示经过解码后最有可能的输入。



自然语言处理中的机器翻译，词性标注，语音识别等多个问题都可以使用信道噪声模型来解决，对于纠错问题也可以使用信道噪声模型来解决，相应的求解问题可以用公式表达：

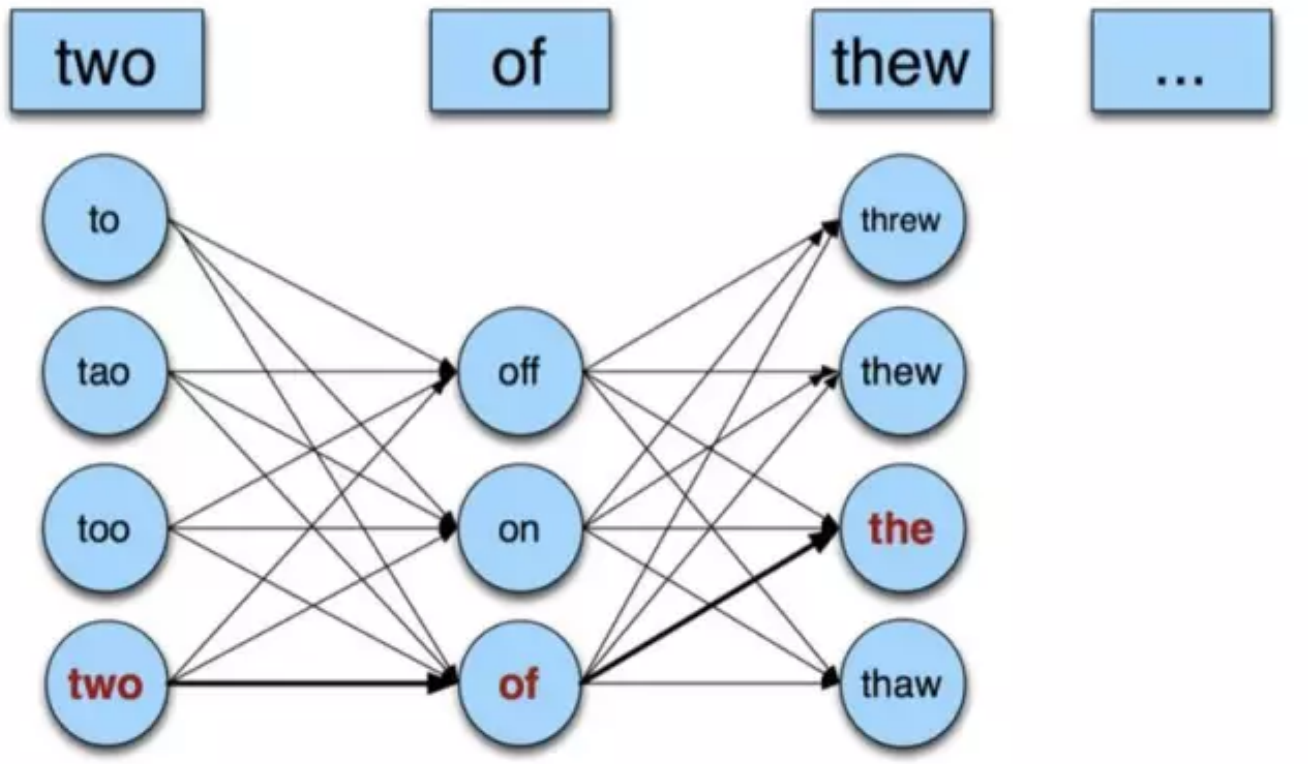
$$\hat{w} = \underset{w \in V}{\operatorname{argmax}} P(w \mid x)$$

$$= \underset{w \in V}{\operatorname{argmax}} \frac{P(x \mid w) P(w)}{P(x)}$$

$$= \underset{w \in V}{\operatorname{argmax}} P(x \mid w) P(w)$$

其中 p(x|w) 是正确的词编辑成为错误词 x 的转移概率，p(w) 是正确词的概率，p(x) 是错误词的概率。

噪声信道模型和 Non-word 纠错类似，只是计算目标从某个候选词的最大概率变成不同位置候选词组合形成的句子 p(s) 的最大概率，这个问题可以使用 HMM（Hidden Markov model，隐马尔可夫模型）求解。



除了上述 2 种方法，真实的场景中，还会结合搜索日志的 session 分析和点击模型来进行纠错结果的排序调整。这里不再展开。

现在我们再看看 Query Suggest 模块,Query Suggest，也即输入下拉提示，根据用户输入的查询词前缀自动提示用户最有可能输入的完整查询词列表。

这里涉及几个问题：

1. Suggest 词条来从哪里来
2. 如何根据当前的输入快速匹配到候选 suggest 词条列表
3. 如何对候选 suggest 词条列表进行排序

suggest 词条通常主要来自用户搜索历史 query log，但存在数据冷启动的问题，开始时缺少 query log 时如何处理？对于一些垂直的应用场景，比如小说搜索中，suggest 词条

也可以是作品的标题、标签、作家名等，电商搜索中可以是品牌词库、产品列表等。

对于 suggest 词条列表的存储结构与快速匹配，如果 suggest 词条列表不是很大，Trie 树（又称字典树）是一个不错的选择，用 Trie 树实现的主要优点是利用字符串的公共前缀来降低查询时间的开销以达到提高效率的目的，实现也比较简单，但缺点是节点数增加到一定程度内存开销会成为瓶颈。如果 suggest 词条列表很大，可以选择 Ternary Tree(又称三叉搜索树)， 三叉搜索树对 Trie 的内存问题（空的指针数组）进行了专门优化，具体细节大家可以 google，这里不再深入。

Suggest 候选词的排序通常根据候选词项的整体热门指数，即搜索的多的排在前面。当然实际应用场景中的排序会考虑更多的排序因素，比如个性化的因素，当下热门指数等因素。

Query Expansion 查询扩展模块

查询词扩展技术通过将与用户查询词相近、相关的词扩展到用户查询词中的方法, 更准确地描述用户的信息需求, 去除用户查询词的多义性, 从而更精确地查询用户所需信息。在信息检索技术中, 查询词扩展是一种能够有效提高查询效率的技术。通过用户搜索日志和点击日志可以挖掘出查询扩展词。

我们在实践中采用一种基于搜索日志会话局部上下文和全局上下文为语料库使用 word2vec 构建 skip-gram 词向量模型，根据词向量模型可以取得与查询词最相似的前 N 个词构成初步的相关候选词表，然后再利用 K 近邻算法从相关词候选词表选取出语义最相关的候选词作为查询词的扩展词。

搜索日志会话局部上下文是指与当前 query 在同一个会话上下文中的共现 query, 也是用户对 query 的查询重构，比如初始 query 为“变形金刚”，用户在查询会话中可能将 query 变换为 “变形金刚电影”进行搜索，则“变形金刚电影”为原始 query 的局部上下文。

query 的全局上下文挖掘思路：

根据查询词和查询所点击的结果构建二部图，利用随机游走模型计算出每个查询词的文档分布作为查询词的查询向量，再利用 KL 距离来计算两查询向量之间的相似性。

Query Classification 查询意图分类模块

通常有基于规则模板的分类方法和基于机器学习的分类方法。

一种是基于规则模板的分类方法，这种方法比较适用于查询非常符合规则的类别，通过规则解析的方式来获取查询的意图。比如：今天天气怎么样， 可以转化为 [日期][实体: 天气][询问词: 怎么样]上海到曼谷的机票价格， 可以转化为 [地点] 到 [地点][机票 / 车票 / 火车票] 价格

这种方法的对比较明确的规则性强的方式有精确的识别度，缺点是覆盖度低，用户查询稍作变换可能就不 match 了，另外规则的发现和制定主要靠人工进行。

另一种是基于机器学习分类的方法。

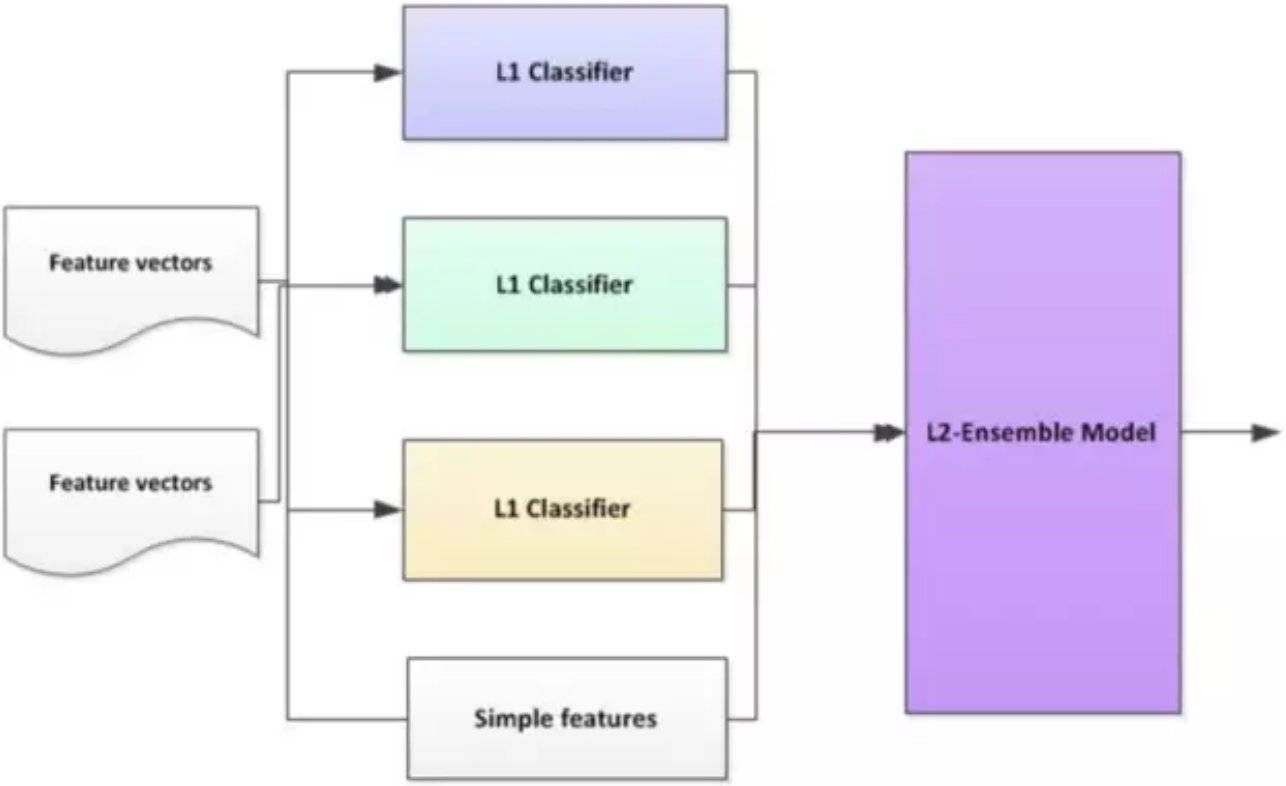
如果有确定的查询类别体系，基于机器学习的查询意图分类是一个不错的选择，可以选择 SVM 作为分类器，关键在分类特征的选择, 还有训练样本的准确标注。

这个和我们之前参加过的 2014 ACM CIKM 竞赛的问题类似，那年 CIKM 竞赛的题目是自动识别用户的查询意图（Query Intent Detection，QID）：给定一批标注过类别的搜索日志包括查询日志和点击日志作为训练样本，其中也有部分未标注的，类别为 unknown。

在特征的选择方面，除了基本的 Query 的长度、Query 的频次、Title 的长度、Title 的频次、BM-25、Query 的首字、尾字等，我们通过对 log session 上下文的分析，进行了 Query 间特征词汇的挖掘，运用了 query 在相同 session 中的共现关系，挖掘 query 之间的子串包含关系，query 和点击的 title 之间的文本特征关系等。

在分类模型的选择方面，我们选择了 Ensemble 框架。Ensemble 的基本思想是充分运用不同分类算法各种的优势，取长补短，组合形成一个强大的分类框架。不过 Ensemble 不是简单的把多个分类器合并起来结果，或者简单将分类结果按固定参数线性叠加 (例如不是 a1 * ALGO1 + a2 * ALGO2 + a3 * ALGO3)，而是通过训练 Ensemble 模型，来实现最优的组合。

在 Ensemble 框架下，我们分类器分为两个 Level: L1 层和 L2 层。L1 层是基础分类器，L2 层基于 L1 层，将 L1 层的分类结果形成特征向量，再组合一些其他的特征后，形成 L2 层分类器（如 SVM）的输入。这里需要特别留意的是用于 L2 层的训练的样本必须没有在训练 L1 层时使用过。



Semantic Tagging 模块

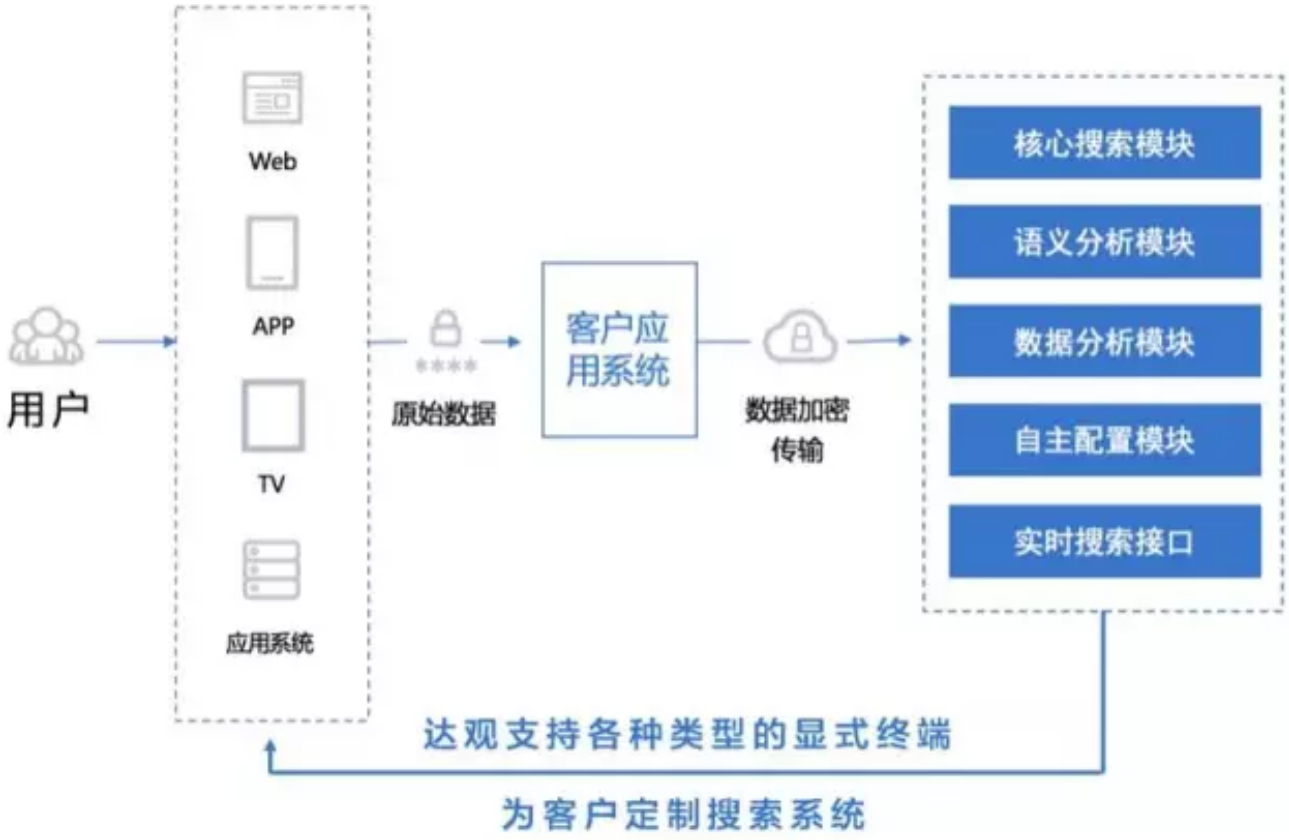
这个模块主要是对 query 中的命名实体进行识别，比如对电商搜索 query 中的品牌词、产品词、属性词、地址进行识别。对 query，用一个相对准确的词典 (品牌词 / 产品词 / 属性词 / 地址词) 去标注语料。

比如对于 ” 新西兰安佳奶粉二段 “ 标注语料如下所示：新 B-loc 西 I-loc 兰 I-loc 安 B-brand 佳 I-brand 奶 B-product 粉 I-product 二 B-attr 段 I-attr实体词识别模型可以通过 crf 来进行训练。

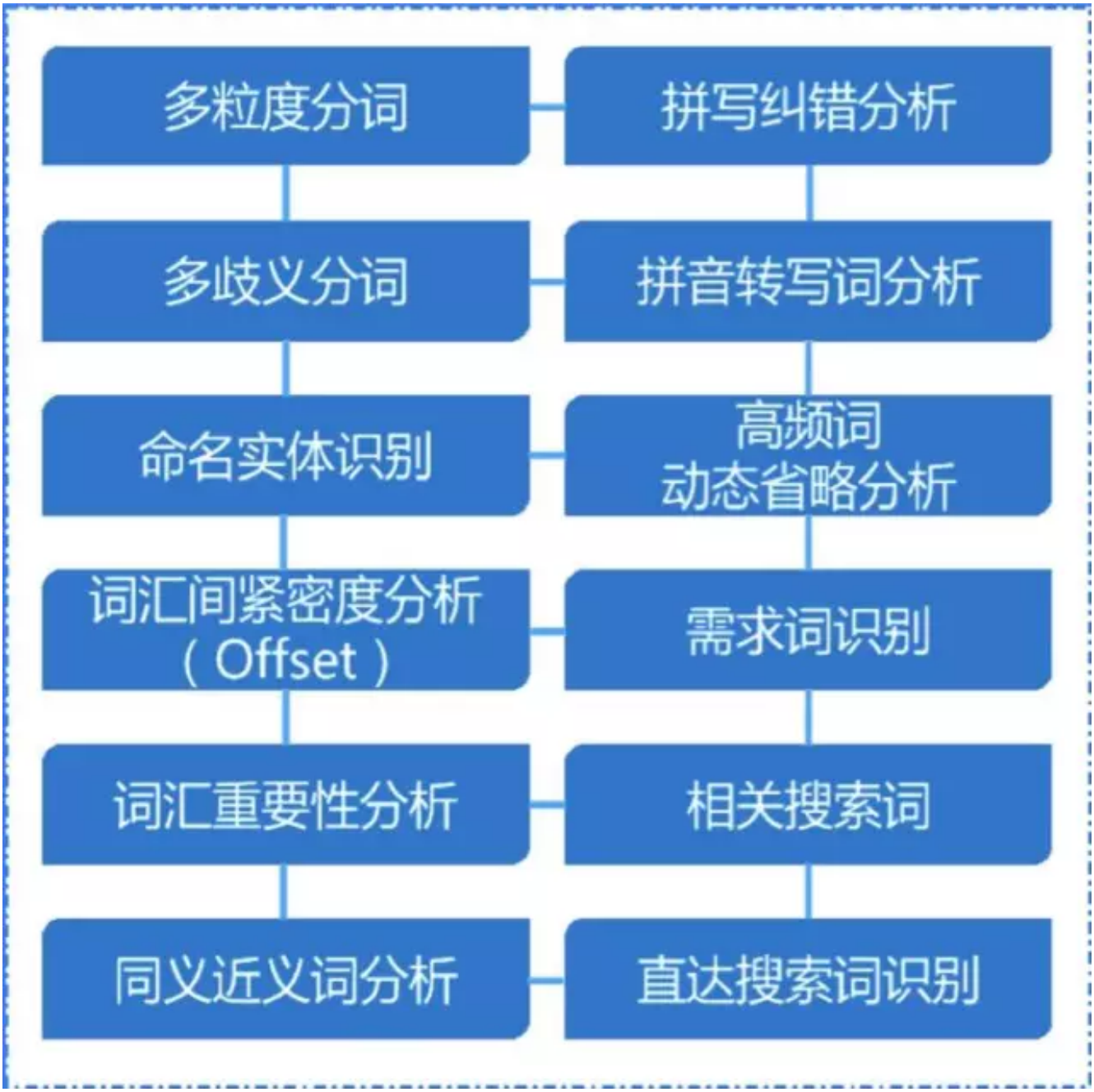
至此，第二部分 如何识别用户搜索意图 也讲完了总结一下，我们首先简单说明了用户搜索意图的主要分类：导航类、信息类、资源类，然后对搜索意图识别的主要功能模块查询纠错、查询词自动提示、查询扩展，查询自动分类、语义标签等实现思路分别进行了介绍。

达观搜索意图识别引擎

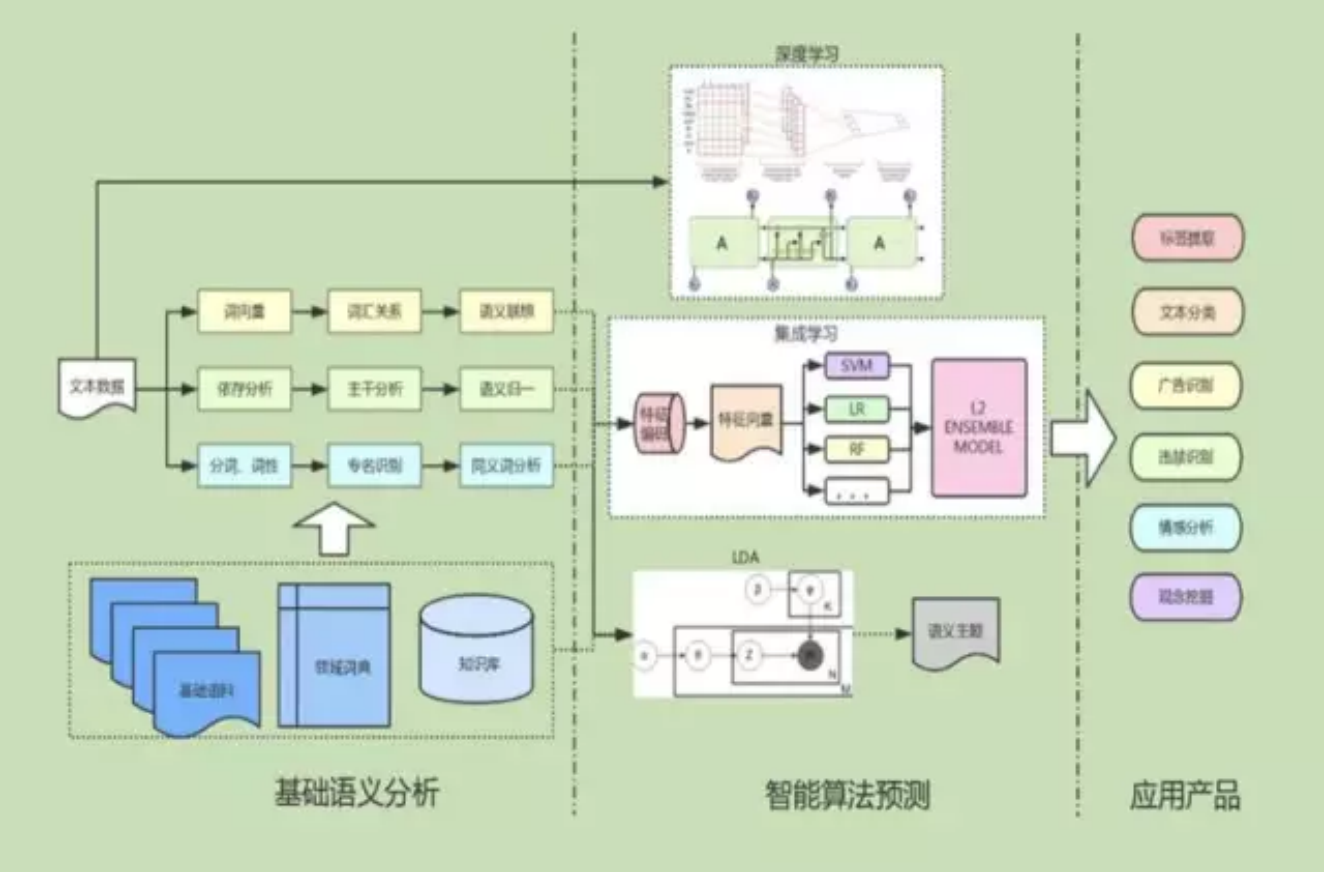
达观通过 RESTAPI 接口的方式向客户提供基于公有云和私有云的搜索服务。其中语义分析模块包含了对用户 query 意图的离线挖掘和在线预测。



达观 query 意图识别引擎内部组合了多粒度分词、多歧义分词、命名实体识别、词汇间紧密度分析、词汇重要性分析、同义近义词分析、拼写纠错、拼音转写、高频词动态省略分析、需求词识别、相关搜索词、直达搜索词分析等多个功能模块。



引擎底层基于达观统一的文本语义挖掘算法平台：



达观文本语义挖掘算法平台是一个融合了多种算法的集成学习平台，既包括经典的 SVM、LR、RF、LDA 等算法，也包括 CNN、RNN、LSTM、BILSTM 等深度学习算法。比如在实践中，我们尝试将线性统计模型 CRF 和神经网络结构 LSTM 相融合的方法，在 LSTM 的输出端将 softmax 与 CRF 结合起来，使用 LSTM 解决提取序列的特征问题，使用 CRF 有效利用了句子级别的标记信息，取得了不错的效果。

由于时间有限，以上是本次分享的全部内容，感谢大家的参与，感兴趣的同学后面可以继续交流。

答疑环节

Q1：搜索领域目前在大数据场景越来越重要，真正识别用户所需不是一件容易事，比如输入一个 cpu，怎么准确猜出用户的意途？

桂洪冠：像这种短 query 在电商搜索的场景中尤其多，比如用户在某海淘网站中输入的 query 很多为 “包”，“水”，这时挖掘用户意图要靠搜索的历史行为数据，首先可以从统计的角度看搜索了这些 query 的用户大多点击的是哪种类型的结果。另外我们也会结合用户个体的行为数据，识别出用户个性化的标签，比如他的类目偏好、品牌偏好等。

Q2：query 意图识别中除了从用户行为角度做特征挖掘外，还会从平台本身资源数据中做核心词分析以作为映射特征数据（像通过新词挖掘来识别 query 中的网络潮流词）。但是对于垂搜而言，有些情况下会出现，文本资源较少，像短 query，垂搜平台文本数据较少这些情况。这样做意图识别时，效果会不会比较差？像我们常用的文本相关性分析方法、意图分析方法是不是就不太管用了（像基于互信息 + 信息熵的核心词分析等）？

桂洪冠：除了对搜索日志和点击日志的挖掘，实际我们还会对 item（文档）的语料数据进行挖掘分析。对于垂直场景中 item 文本语料不足的问题，我们一般会根据不同的垂直场景，比如电商或视频，去爬取相关的语料作为补充，另外我们也会把相同场景的多个客户的 item 文本语料进行综合处理，类似迁移学习的思路。

Q3：达观的搜索服务使用了大量的模型，有遇到性能问题吗？

桂洪冠：我们的算法平台是基于分布式的大规模数据处理平台，底层基于阿里云提供的分布式基础设施，平台的水平扩展性保证了大规模数据模型训练的性能。下周我的同事会给大家分享大规模机器学习算法，大家对这个专题感兴趣的可以参加。

Q4： 搜索引擎在 AI 方面的发展方向是？

桂洪冠： 我今天在分享的开始提到过搜索引擎发展经历了四个阶段，目前正在进行中的第四阶段其实正是搜索引擎在往 AI 的发展阶段，我觉得搜索引擎往 AI 方向的发展会表现出如下特征：

1. 从有框搜索走向无框搜索
2. 从关键字搜索走向更自然语言的问答式搜索
3. 搜索的场景从文字转向语音，而且未来搜索的场景入口远远不至于现在的 APP，智能音箱、智能电视、智能冰箱等设备将呈现出更多的人机交互的接口。人通过自然语言给设备发送执行进行执行，机器要理解人的语言并执行相应的指令，这本质上就是搜索。

作者介绍

桂洪冠，达观数据技术副总裁，中国计算机学会（CCF）会员，首席数据官联盟成员，达观数据架构师，搜索组总负责人，担任搜索系统的架构设计与开发、搜索效果优化、客户搜索技术支持、搜索团队日常管理等。拥有 15 年系统架构设计和管理经验，曾担任新浪微博广告系统架构师，和阿里巴巴国际交易系统架构师和高级技术专家职务。在大规模数据挖掘和系统架构领域有丰富的经验和实践经验，曾经在 InfoQ、CSDN 等知名技术社区上发表高水平技术文章，擅长高并发、大容量、海量规模应用系统的架构设计和工程开发。在搜索引擎、广告系统、自然语言处理等技术领域申请有多项国家发明专利。