第37卷 第16期 计 算 机 工 程 2011年8月
Vol.37 No.16 Computer Engineering August 2011

·软件技术与数据库· 文章编号：1000—3428(2011)16—0077—03 文献标识码：A 中图分类号：TP311

基于语义分析的查询扩展方法

王水利，黄广君，霍亚格

(河南科技大学电子信息工程学院，河南 洛阳 471003)

摘 要：查询扩展是优化信息检索的有效途径。为此，提出一种基于语义分析的查询扩展方法，利用基于互信息的共现模型分析初检文档，并将其作为部分扩展源，用模型的统计结果剪枝由语义词典 WordNet 生成的语义树，限制扩展范围。从初检文档和语义词典两方面选取扩展词对原查询进行扩展形成新的查询集。对返回结果进行重排序，调整前 n 篇文档的查准率。实验证明该方法是切实可行的。

关键词：查询扩展；语义树；互信息；文档重构

Query Expansion Method Based on Semantic Analysis

WANG Shui-li, HUANG Guang-jun, HUO Ya-ge

(College of Electronic Information Engineering, Henan University of Science & Technology, Luoyang 471003, China)

【Abstract】 Query expansion is an effective way to optimize information retrieval. A method for automatic query expansion based on semantic analysis is proposed. This method uses a co-occurrence model based on mutual information to analyze the retrieved documents, which is a part of the extended source, and uses the results of the analysis to prune the semantic tree generated by the semantic dictionary WordNet to limit the expansion. Extended words selected from both retrieved documents and the semantic dictionary are employed to form a new query set. The new retrieval results are re-ranked to adjust the retrieval precision. Experimental results show this method is feasible.

【Key words】 query expansion; semantic tree; mutual information; document reconstruction

DOI: 10.3969/j.issn.1000-3428.2011.16.026

1 概述

查询扩展是在原查询词的基础上加入与用户用词相关的词或者词组，组成新的、更准确的查询序列。使扩展后的查询序列能更清晰地表达用户的查询意愿。查询扩展的主要任务是在相关模型的约束下对原查询进行可控范围内的扩展，降低查询的漂移量，在保证查准率的同时提高查全率。

现有的查询扩展方法主要有 2 种：(1)基于反馈的扩展方法，从中提取和原查询信息关联度较高的特征词或者概念加入查询；(2)利用某种资源对查询进行直接扩展。

第(1)种查询扩展比较传统，主要分为全局分析、局部分析等。这种查询扩展只局限于符号匹配层面上的扩展，而忽略了查询特征词或概念的语义关系，无法从根本上提高信息检索性能。常见的方法是“伪相关反馈”，即对初检文档进行分析，将相应权值较高的词加入查询式。例如文献[1]提出的基于矩阵加权关联规则挖掘的伪相关反馈查询扩展算法，文献[2]用到的基于潜在语义分析的查询扩展方法。最近几届的 TREC 会议的研究结果表明，使用这种扩展方式通常可以较显著地提高信息检索系统的检索效果。但同时也有研究表明，因为其效果强烈依靠第 1 次检索的结果并不稳定。

第(2)种查询扩展需要利用语义词典或者领域本体等包含词或概念关联信息的资源。这种扩展方式是一种基于概念的查询扩展，能同时对初始查询词的同义词、近义词、广义词、狭义词等进行检索，有利于提高检索的查全率。这种扩展方式的优势是不需要大规模的语料库支持和长时间的训练，并且有 WordNet、HowNet 等越来越成熟的大规模语义词典。但是这些优点也是它的软肋，如果选择扩展词时只考虑扩展词在词典中的概念相似度，由于词典是一个系统化的资源，只是根据概念的简单组织结合，而书面常用词语只占其中的很小一部分，会导致许多不相关的词加入，在提高查全率的同时却降低了查准率。如文献[3]基于 WordNet 本体的 Web 检索模型。这种方法仅依赖于语义词典而与实际语料库无关，对查询质量也产生了影响。

综合以上扩展方式的优缺点，有学者提出了结合 2 种扩展方式的方法，例如文献[4]提出的基于局部上下文分析剪枝概念树的方法。但是以往学者提出的方法只是 2 种方式的简单的叠加或者只考虑词间的简单共现，没有从根本上克服扩展的机械性。对此，本文提出一种基于语义分析的查询扩展方法。用扩展词与原查询所有关键词的互信息和在词典中的相似度组成的综合权值作为扩展词重要性的衡量标准，使扩展词的质量更高，得到质量更高的查询式，从而提高查询的质量。

2 理论基础

2.1 构造语义树

本文算法用 WordNet 词间关系完整的层次结构构建原查询的语义树，用根结点为原查询词的树结构来存储初始扩展的扩展词。由于每个词都有数个不同的义项，为了方便下面实验中的剪枝算法，根据不同词义构造其在该义项上的子树。假设该特征词有 m 个义项，这样就得到一个含有 m 棵子树的语义树，记为 $SemTree$，其中，子树的根结点为原查询的同义词。子树根结点下为该义项上的上下位词构成的子树。本文只对名词进行上位关系、下位关系和整体部分&部分整体关系的扩展。每个概念的同义词在同一个结点上表示。为减少重复计算，本算法给每个节点增加成员数据 $Weight\_node$ (初始值为 0)存储这个节点所有概念在算法中的相关权值的

基金项目：河南省科技攻关计划基金资助项目(102102210159)

作者简介：王水利(1983－)，男，硕士研究生，主研方向：语义 Web，信息检索；黄广君，副教授、博士；霍亚格，硕士研究生

收稿日期：2011-01-25 **E-mail：**wangshuili1985@163.com

最大值。除名词外，其他的查询词不予扩展。语义树构建的具体算法在 3.1 节给出。图 1 为“bank”的语义树。

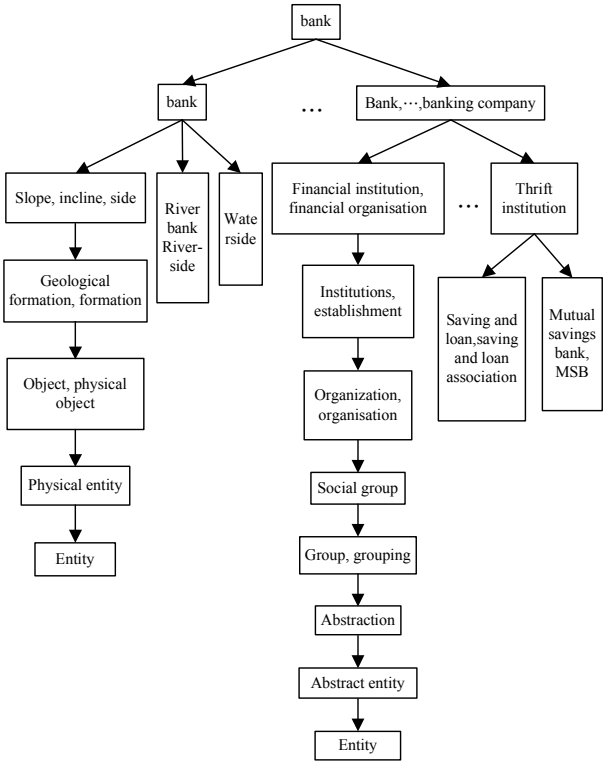

图 1 bank 的语义树

2.2 互信息

一般认为文档中经常共现于同一窗口单元的 2 个词的相关度比较高，共现的频率越高说明 2 个词的关联越紧密。根据这个假设，可以根据统计模型来计算词语之间的相关性，本文算法用互信息来表征这种相关性。

这里用带有衰减因子的互信息 $DMI(x,y)$：

$$DMI(x,y)=MI(x,y)\cdot D(x,y) \tag{1}$$

其中：

$$MI(x,y)=\mathrm{lb}\left(\frac{p(x,y)}{p(x)\cdot p(y)}\right) \tag{2}$$

$$D(x,y)=\mathrm{e}^{-\alpha(Dis(x,y)-1)} \tag{3}$$

$$p(x,y)=\frac{c(x,y)}{\sum_{x',y'}c(x',y')} \tag{4}$$

$$p(x)=\frac{c(x)}{\sum_{x'}c(x')} \tag{5}$$

式(4)中 $c(x,y)$ 是指在训练集中词 x 和词 y 在同一窗口中共现的频率，$c(x)$ 表示词 x 在训练集中出现的频率。一般情况下，认为 2 个词的相关性是随着在同一窗口的距离的增加而减小的，因此，在式(1)中增加了反映词间距离信息的衰减项 $D(x,y)$。其中，$Dis(x,y)$ 表示词 x 和词 y 在所有窗口单元中的平均距离；α 表示词间相关性随词间距离衰减的剧烈程度，本文实验选取 a=0.75。

3 扩展算法

本算法首先利用 WordNet 对初始查询进行扩展(WordNet 提供接口 WordNet.Net)，然后用词间互信息作为衡量标准对语义树进行剪枝控制扩展范围，并加入初检文档中与原查询关联度较高的特征词得到最终查询集。最后对返回文档进行基于文档重构的重排序调整前 top-k 篇文档的查准率。具体算法流程如图 2 所示，下面对该扩展方法进行详细描述。

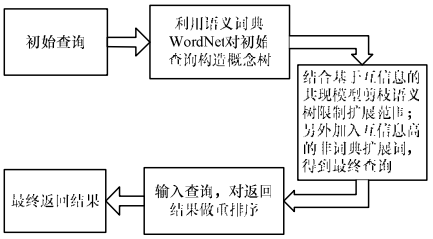

图 2 扩展算法流程

定义 1(语义候选扩展词集) 为初始查询构建的语义树，记为 Sem_canSet。

定义 2(统计候选扩展词集) 在返回文档集中出现的所有特征词的集合，记为 Sta_canSet。并为每个特征词添加对应的成员数据 Mi_k (初始值为 0)用来存储对应的特征词在训练集中与原查询的互信息值。

定义 3(语义权重) 根据关键词在语义树中与原查询的距离定义的权重，反映扩展词 k 和原查询词语义关联度。记为：$Sem\_weight(k)$，其中，$Sem\_weight(k)=1/length$，$length$ 为关键词 k 到根结点路径上的边数，同结点上的关键词 $length=0$。

定义 4(统计权重) 是根据扩展词与原查询的互信息值定义的权重，反映了扩展词和原查询在实际语料集中的关联度。记为 $Sta\_weight(k)$。

定义 5(综合权重) 结合扩展词的统计权重和语义权重重新计算得到的新的权重，记为：

$$wight(k)=\left(\frac{(1+\beta)\times Sem\_weight\times Sta\_weight}{\beta\times Sem\_weight+Sta\_weight}\right)$$

其中，β 为 2 个权值的调节因子，其决定 2 个权值对综合权值的贡献度。

3.1 初始扩展

首先在搜索引擎输入原查询，对返回的前 100 篇文档进行处理，把提取出的所有特征词作为统计候选扩展词放入 Sta_canSet 中。对原查询中的每个特征词 q_i 利用 2.1 节方法构建语义树 $SemTree_i$，形成语义候选扩展词集 $Sem\_canSet=\{SemTree_1,SemTreee_2,\mathrm{L}\ ,SemTree_n\}$。

原查询的语义树构造具体算法如下：

1. 对原查询 q 进行标注、词缀还原、去停用词，概念提取后提取特征词 n 个：$q=(q_1,q_2,\mathrm{L}\ ,q_n)$

2. For i=1 to n

初始化以 q_i 为根结点的语义树

3. For j=1 to m // q_i 有 m 个义项

4. 调用 WordNet 的接口函数 findtheinfo 分别得到 q_i 该义项的上位词、下位词、整体部分词，构建该义项上的子树。其中，该子树的根结点为原查询词同义词、子树为上下位词和整体部分词构成的子树

5. End j

6. End i

3.2 基于互信息的扩展范围控制

对返回的 100 篇初检文档组成的文档集，用现在比较成熟的技术实现文档的标注、词缀还原等工作，编程实现词共现统计(包括距离信息)。在统计时，选择句子作为窗口，在本文所做的实验里只考虑句号、问号、感叹号 3 种句子终止符。利用统计信息和式(6)计算所有的特征词 k 和原查询的互信息 $Mi(k)$，其中：

$$Mi(k)=\sum_{q_i\in q}DMI(q_i,k) \tag{6}$$

为了设置阈值的方便，排除不同查询对应的互信息值的波动对阈值设定的影响，本实验调节式(6)中的互信息值进行归一化处理得到介于[0,1]的 $NMi(k)$，如式(7)所示：

$$NMi(k)=\frac{Mi(k)}{\max_{k'\in Sta\_sanSet}(Mi(k'))} \tag{7}$$

下面利用互信息分析结果对语义树进行剪枝，限制扩展范围，提取最终扩展词。操作流程如下：

(1) Sta_canSet 和 Sem_canSet 的初始化。利用式(7)计算 Sta_canSet 中每个特征词 k 的互信息，并用计算结果更新其对应的 Mi_k 域的值；遍历 Sem_canSet 中所有的语义树，对于每个非根节点的每个概念，结合 $Sem_weight(k)$和 $Sta_weight(k)$计算 $weight(k)$，用该结点中所有概念的 $weight(k)$ 的最大值更新节点的 $Weight\_node$ 域。规定根结点和其子树的根结点 $Weight\_node=1$。

(2)删除 Sem_canSet 中与原查询相关度较低的结点与子树。具体操作是：遍历 Sem_canSet 中每棵语义树 $SemTree$ 上的每个节点，如果在某结点下的所有节点满足以下条件：$\max Weight\_node<\lambda_1$，则删除该结点及该结点下的所有节点；如果该语义树的某棵子树除根结点外所有结点都已被删除，则删除该子树，防止原查询词极少用的同义词的加入。

例如，如果在文档集中很少出现 River bank、waterside 等信息，这时 bank“河岸”义项上子树的上位关系子树的特征词 $Weight\_node$ 值会非常低，则删除该关系下的子树及其下的所有节点；如果 bank 义项的子树的其他子树也都被删除，则删除该义项上的子树。

(3)遍历整个 Sem_canSet 里的每一棵语义树，将剩余的结点中的扩展词加入最终查询 $ResultSet$，每个特征词权重为其所在结点的 $Weight\_node$。

(4)把统计候选扩展词集中的与原查询相关度较高的特征词加入最终查询集 $ResultSet$。如果满足条件：$Mi\_k>\lambda_2$，$k\in Sta\_canSet$，$k\notin Sem\_canSet$，则把特征词 k 加入 $ResultSet$，在最终查询集中的权重设为 Mi_k。为了防止与原查询无关的词的加入，这个阈值可以取一个比较大的值。

例如当输入“bank”时，由于近来的热门话题次贷危机、通货膨胀等，初检文档中可能会频繁出现 subprime crisis、inflation 等在 Sem_canSet 没有出现的词，这些信息可能就是用户想要知道的，因此有选择地把它们加入最终查询集。

3.3 基于文档重构的重排序

由于查询扩展通常只提高查全率不能明显地提高查准率，而一般用户的使用习惯是只关心前 top-k 篇文档是否命中，因此在实际应用中只要提高前 top-k 篇文档的查准率就提高了系统的实用价值。在此算法中，根据文本与查询表达式的相似度来对返回文档进行重排序。一般都是采用夹角余弦值表示两向量的相似度，这里采用一种改进的方法来计算相似度。

用空间向量模型表示原查询和返回文档，文档权值用传统的 tf_idf 方法。由于在人类的检索过程中，是把文档中那些具有密切语义联系的词语纳入到一个概念中，是信息聚合和加强的过程[5]，因此文献[5]提出了文档重构的思想，并用实验证明了它的可行性和优越性。文档重构的本质是把文档中含义相似的具体词集中到一个概念下。文档重构由于要求对所有的待查询文档进行处理，因此不可能实时地运用到互联网资源中去，只能在封闭的文档集中运用。正因为这个特性，此算法非常适合在返回文档重排序时对返回文档进行初始化。然后计算重置后的文档向量与原查询向量的相关度，根据相关度的高低对返回文档进行排序返回用户界面。下面是加入文档重构的重排序算法：

1：For i=1 to r //返回 r 篇文档
2：For j=1 to m //文档向量中共 m 个关键词
3：调用 WordNet 接口函数提取 t_{ij} 的同义词和上位词并生成扩展词集 Hyper_Set
4：For k=1 to n //原查询共 n 个特征词
5：If $q_k\in Hyper\_Set$
6：在重置文档中 $w_{q_k}=w_{q_k}+w_{ij}$，$w_{ij}=0$
7：Else
8：在重置文档中 $w_{ij}=w_{ij}$
9：End k
10：End j
11：计算文档向量和查询式的相关度
12：End i
13：按与查询式的相关度高低返回文档集并输出到用户界面

4 实验

4.1 实验数据集

本实验在 TREC2 第 1 张盘中选取 3 600 篇文档作为测试集，使用的 topic101~120 共 20 个 topic，在构造查询时，这里只用 topic 中的 title 的信息，每个 title 经过去停用词、词缀还原后一般包含 3 个~5 个特征词。

4.2 实验评测指标

本算法采用的评测指标是 MAP(Mean of Average Precision)和 prec@20。MAP 是对所有查询的平均查准率再进行取平均，反映系统在全部相关文档上的性能的单值指标，可以体现查全率和查准率两方面的信息。Prec@20 表示前 20 篇返回文档的查准率。

4.3 阈值的设定

本实验中对扩展范围进行控制的环节用到了 2 个阈值 λ_1 和 λ_2，下面是对其值大小的测定：

(1)当 λ_2=1 时，即仅从语义词典 WordNet 中选取扩展词时的情况，λ_1 取不同值时的情况如图 3 所示，所以选择 λ_1=0.55。

(2)当取 λ_1=0.55 时，λ_2 取不同值时的情况如图 3 所示，所以取 λ_2=0.9。

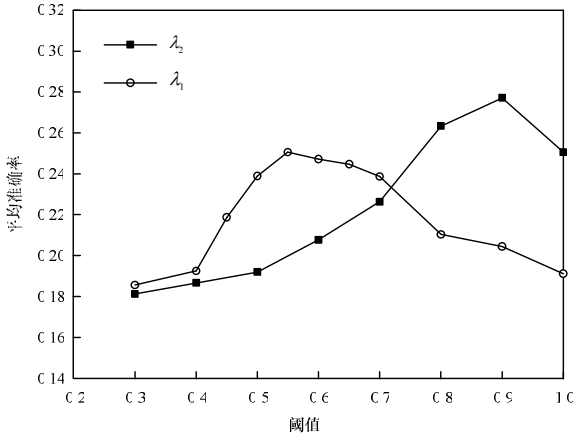

图 3 λ_1 和 λ_2 对查询的影响

表 1 是当 λ_1=0.55、λ_2=0.9 时，本实验方法与未扩展、使用 WordNet 进行概念扩展、使用互信息(MI)法的对比。

(下转第 93 页)