# 基于关键词的RDF数据图查询模型研究

郑志蕴 刘 博 李 伦 王振飞
(郑州大学信息工程学院 郑州 450001)

**摘 要** 随着语义网数据的海量涌现,人们更加关注RDF图的数据查询效率,通过关键词匹配直接查询RDF数据图成为一个研究热点。针对关键词查询中普遍存在的结果冗余与偏离等问题,提出了一种基于关键词的RDF数据图查询模型。该模型首先采用提出的基于迭代的图查询算法(ISGR)对所查询关键词进行子图匹配,得到唯一且最大的结果子图集合;然后根据关键词图与结果子图之间的结构信息,利用统计语言模型,给出了一种结果子图排序方法(SimLM)。对比实验表明,提出的查询模型及排序方法在一致性和相关性方面的性能优于传统模型。

**关键词** RDF数据图,关键词查询,子图,相似度矩阵,统计语言模型
**中图法分类号** TP391.3 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.7.050

## Research of Keyword Search Model over RDF Data Graph

ZHENG Zhi-yun LIU Bo LI Lun WANG Zhen-fei
(School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China)

**Abstract** As huge amounts of the semantic Web data have sprung up, people are more concerned about query efficiency over RDF data graph. Retrieving RDF data graph directly by keyword matching is an area of research focus. In this paper, a retrieval model was proposed, which enables keyword search for RDF graph. First, for the improvement of query efficiency, an algorithm named ISGR (an Iterative way to SubGraph Retrieval) was proposed, in which query keywords can be matched with subgraphs from RDF data graph, and a collection of subgraphs which should be unique and maximal is got. Next, in order to solve the problems of redundant results and deviation that frequently emerge in keyword search, a mixture ranking model(SimLM) was proposed, which considers the structural information between keyword graph and result graph, and mixs statistical language model. A numbers of contrast experiments over two kinds of open source real datasets prove that the retrieval and ranking model proposed in this paper outperforms well-known techniques in the field of consistency and relevance.

**Keywords** RDF data graph, Keyword search, Subgraph, Similarity matrix, Statistical language model

## 1 引言

资源描述框架(Resource Description Framework, RDF)是对语义数据进行描述的标准,正被广泛应用于元数据的描述及语义网中。众多机构和项目均采用RDF表达元数据,例如Wikipedia、DBLP。丰富的RDF数据使构建大规模知识库成为现实,如IBM智慧地球、Freebase知识库等。RDF数据由主体(Subject, S)、谓词(Predicate, P)和客体(Object, O)三元组组成。RDF数据图(简称RDF图)作为RDF数据最直观的表现方式,包含两个结点与连接结点的有向边,分别与三元组中的主体、客体和谓词相对应,边的方向由主体指向客体。表1表示一系列从某电影知识库中得到的RDF三元组集合,图1以RDF数据图的形式描述了表1中的RDF三元组。

表1 某电影知识库中的一些RDF元组片段

| 主体(S) | 谓词(P) | 客体(O) |
|---|---|---|
| Traffic | hasWonPrize | Academy_Award |
| Innerspace | hasWonPrize | Academy_Award |
| Innerspace | hasGenre | Comedy |
| Joe_Dante | directed | Innerspace |
| Toy_Story | hasWonPrize | Academy_Award |
| Road_Trip | hasGenre | Comedy |
| Toy_Story | hasGenre | Comedy |
| Tom_Hanks | actedIn | Toy_Story |
| Diner | hasWonPrize | Academy_Award |
| Diner | type | Comedy_films |
| Steve_Guttenberg | actedIn | Diner |
| The_Pink_Panther | type | Criminal_comedy_films |
| The_Pink_Panther | hasWonPrize | Academy_Award |
| Police_Academy | type | Comedy_films |
| Steve_Guttenberg | actedIn | Police_Academy |
| The_Darwin_Awards | type | Comedy_films |

到稿日期:2014-06-17 返修日期:2014-07-21 本文受河南省国际科技合作项目(144300510007)资助。
郑志蕴(1962-),女,博士,教授,主要研究方向为分布式计算、智能信息处理;刘 博(1988-),女,硕士生,主要研究方向为RDF数据管理;李 伦(1976-),男,讲师,主要研究方向为云计算;王振飞(1973-),男,博士,副教授,主要研究方向为分布式计算、信息安全,E-mail:iezfwang@zzu.edu.cn(通信作者)。

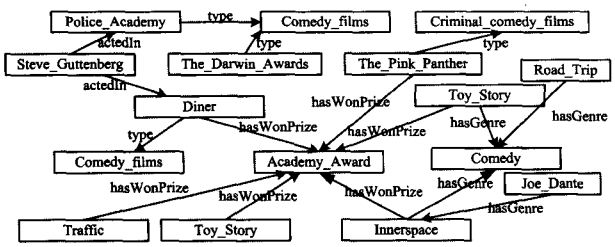


图 1　RDF 数据图示例

随着应用领域的不断扩大，人们开始关注 RDF 数据的查询效率。RDF 数据查询方式分为元组模式查询和关键词查询两种。元组模式查询(Triple-Pattern Query)是一种主要的 RDF 查询方式，由于所需查询语言(如 SPARQL 等)、语法比较复杂[1]，因此为高阶用户或搜索 API 接口所定制。一般用户采用第二种方式，即关键词查询。但是将关键词查询技术直接应用到 RDF 数据查询会出现结果冗余与偏离等问题，例如在 RDF 图中关键词查询往往会得到大量包含关键词的信息图(包含关键词的元组记录)，对这些候选结果进行简单的随机排列或者仅仅采用 IR 或 DB 排序方法，而不考虑 RDF 图数据的结构和内容特性，将降低查准率。因此基于 RDF 数据图的关键词查询研究很有必要。

本文提出一种针对 RDF 数据图的关键词查询模型，支持对实体、属性或关系名进行查询。该模型在 RDF 数据图中对所查询关键词进行子图匹配，得到结果子图。在排序阶段，考虑关键词图(关键词所构成的数据图)与结果子图之间的结构信息，并借助统计语言模型 SLM(Statistical Language Models)对结果子图进行排序，输出 Top-$k$ 个结果。

## 2　相关工作

基于关键词的 RDF 数据图查询包括查询和结果排序两个阶段。有关 RDF 数据图的查询研究主要有两种方法：关键词结构化方法和关键词直接匹配方法[2]。

### 2.1　关键词结构化方法

关键词结构化方法是指由关键词构造结构化查询语句，再利用转换后的查询语句检索结果。文献[3]认为关键词查询是基于元组模式图结构化查询的一种隐式表示，首先在模式图上找到包含用户查询关键词的最小子图，得到 Top-$k$ 个子图，将其提供给用户，由用户选择最合适的待评估结构化查询。然后结合数据图将子图映射成查询语句，翻译成 SPARQL 查询语句，进行数据库的查询操作并返回结果。但其策略的执行需要用户反馈，而且易受到 RDF 模式信息缺失的影响。

Ladwig G 等人将关键词查询转换成结构化查询语句返回[4]。该方法从 RDF 数据中抽取结构信息，构造查询搜索图，搜索符合要求的子图生成结构化查询，再结合实体索引和关系索引直接得到查询结果。但是该方法抽取结构信息的时间开销大。其由于响应时间等于查询转换时间加查询结果生成时间，实时响应速度并不理想。

### 2.2　关键词直接匹配方法

关键词直接匹配结果采用图的关键词查询方法。由于 RDF 的数据模型是图，因此 RDF 上的查询处理可以被转换为子图匹配问题，即在 RDF 数据图上定位包含查询词的斯坦纳树(Steiner Tree)[5]。此方法通常需要借助有效的索引来匹配子图并快速搜索结果。文献[6]提出了 gStore 系统并使用 VS-tree 索引，在标签图上对查询语句有效匹配，VS-query 算法借助摘要图(Summary Graph)使查询空间大幅减小，查询效率显著提高。

### 2.3　结果排序方法

目前学者对 RDF 数据图关键词查询算法的研究成果有很多，对应的查询性能有很大提高，然而这些工作最终都会面临同一个问题，即如何对结果进行排序。例如，大规模 RDF 知识库可能包含噪声或错误信息，因此查询结果往往会高度分散，如何使最相关的结果排序最靠前而不是随机序列，方便用户快速获取需要的信息，仍是值得关注的焦点。

学术界对结果排序方法的研究仍在起步阶段。基于关键词图查询的结果排序有基于图结构信息和基于内容的 IR 式排序两种方法。文献[5]基于图的结构性考量，将结点和边的权值作为排序依据，根据图中结点和边的权重制定一个相关性评分函数对结果进行评分和排序。但由于模式图中结点和边的权重很难获取，大大降低了整个查询的效率。基于内容的排序方法主要包括 tf-idf 模型和 SLM[7-9]。相对于 tf-idf 模型，SLM 更倾向于对整个文档之间的数据进行统计排序，统计语言模型更适用于 RDF 数据。文献[8]借助 LM 排序模型对排序对象(RDF 三元组中的实体)进行排序，但其仅将三元组中的实体看作检索元素而忽略掉实体之间的关联；Elbassuoni S 等人对该排序模型进行改进[9]，从全局角度考虑三元组的实体与关联，将实体间关系也作为检索和排序的元素，并通过实验验证了其高效性。

根据 RDF 数据图的特性，结合上文所提到的查询及结果排序方法，本文构建关键词查询模型，提出基于迭代方法的图查询算法，高效完成查询过程；结果排序则考虑关键词图与结果子图的两个相关性因素：结构相似性和内容相关性，与单一方法相比更全面有效。

## 3　基于关键词的 RDF 数据图查询模型

基于关键词的 RDF 数据图查询模型分为预处理、图查询和结果排序 3 个阶段，工作流程如图 2 所示。

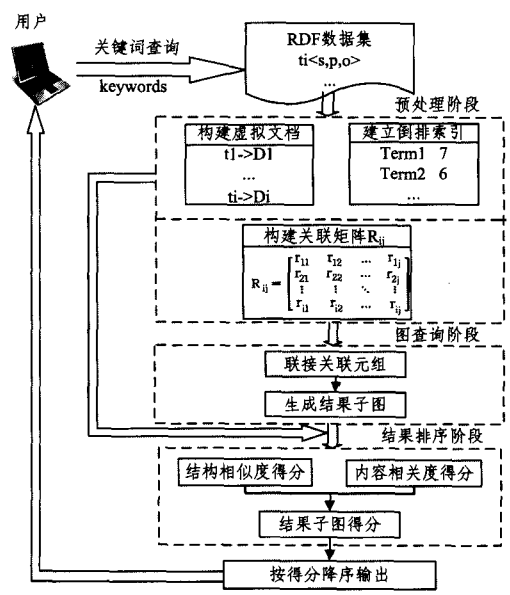


图 2　模型的基本流程

用户输入关键词对 RDF 数据进行查询，首先进行预处

理，建立关联矩阵、构建虚拟文档和倒排索引。关联矩阵由不同关键词对所有三元组进行分组构成，作为图查询阶段的输入数据。给每个三元组 $t_i$ 构建一个虚拟文档 $D_i$，选取关键词和相关术语(Term)的倒排索引作为索引方式，是结果排序阶段的排序依据。然后经过图查询阶段，将关联矩阵中对应的元组进行匹配，得到元组的邻接表，生成结果子图。最后在结果排序阶段，根据关键词图与结果子图的结构相似度和内容相关度进行综合评分，并降序输出。

举例说明，如查询条件“获得奥斯卡金像奖的喜剧电影(finding comedies that have won the Academy Award)”，进一步假设用户借助关键词查询来表达该查询“comedy academy award”。表 2 表示表 1 中 RDF 数据片段执行查询后的所有结果子图。其中第二列表示与对应子图中元组匹配成功的关键词集合，图 3 表示表 2 中第二个结果子图。

表 2　查询“comedy academy award”产生的子图

| 子图 | 关键词 |
|---|---|
| Traffic hasWonPrize Academy_Award | Academy，award |
| Innerspace hasGenre Comedy | Comedy |
| Innerspace hasWonPrize Academy_Award | Academy，award |
| Toy_Story hasGenre Comedy | Comedy |
| Toy_Story hasWonPrize Academy_Award | Academy，award |
| Road_Trip hasGenre Comedy | Comedy |
| Diner type Comedy_films | Comedy |
| Diner hasWonPrize Academy_Award | Academy，award |
| Diner type Comedy_films | Comedy |
| Diner hasWonPrize Academy_Award | Academy，award |
| The_Pink_Panther type Criminal_Comedy_films | Comedy |
| The_Pink_Panther hasWonPrize Academy_Award | Academy，award |
| Police_Academy type Comedy_films | Academy，comedy |
| The_Darwin_Awards type Comedy_films | Award，comedy |

$$R[m][n]=\begin{bmatrix} 0 & 1 & -1 & -1 & 4 & -1 & -1 & -1 & 8 & -1 & -1 & -1 & 12 & 13 & 14 & -1 \\ -1 & -1 & 2 & -1 & -1 & 5 & 6 & -1 & -1 & 9 & -1 & -1 & -1 & 13 & -1 & 15 \\ 0 & 1 & -1 & -1 & 4 & -1 & -1 & -1 & 8 & -1 & -1 & -1 & 12 & -1 & -1 & 15 \end{bmatrix}$$

**定义 2**(邻接表)

(1)给定边 $e_p=t_p=\langle s_p,p_p,o_p\rangle$，如果 $s_p=s_q,s_p=o_q,o_p=s_q$ 或 $o_p=o_q$，则称另一边 $e_q=t_q=\langle s_q,p_q,o_q\rangle$ 与 $i$ 相邻。

(2)对第 $i$ 个关键词命中的列表 $L_i$ 中元组 $t_p$，它的邻接表 $A(t_p)$ 应为包含所有从非 $L_i$ 的列表中获得的邻边 $t_q$。为保证各个邻边分属于不同关键词命中的集合中，当且仅当 $t_i\notin L_j(j\geqslant i+1)$ 且 $t_q\notin L_i$ 时，将 $t_q$ 加入 $t_p$ 的邻接表 $A(t_p)$。

将 RDF 数据转化为关联矩阵，将其作为图查询阶段的输入数据。构造元组邻接表即为匹配并联接元组，以及生成结果子图的过程，元组邻接表即为对应结果子图。

为避免图查询的迭代过程无限长，对已检出子图规定如下属性作约束：

(1)子图必须具有唯一性。即子图所包含的三元组分别对应不同的关键词，若同一关键词有两个元组与其匹配，那么这两个元组必须分属两个不同子图。

(2)子图必须具有最大性。即每一个检出子图都不是任何其他检出子图的子集。

ISGR 算法的 5 个步骤：

Step1　读入 RDF 数据，将输入关键词拆分为单个术语的集合，并按倒排索引中词频的顺序进行预排。

Step2　按处理后的关键词对 RDF 三元组分组，构成关联矩阵。

Step3　对不同分组中的三元组进行匹配联接，创建元组邻接表。

Step4　输出每个元组的邻接表(结果子图)。

ISGR 算法的具体描述：

**算法 1**　INITIALIZE　$R[m][n]$

```
Input:ti(si,pi,oi)
Output:R[m][n]
1. Begin
2.     R[m][n]←-1
3.     if(containsAny(s[id].toLowerCase(),
       strkeyword[j].toLowerCase())
       || containsAny(o[id].toLowerCase(),
       strkeyword[j].toLowerCase()))
4.         R[i][j]=id
5.     else R[i][j]=-1(∀i∈m,∀j∈n)
6. End
```

**算法 2**　EXTENDSUBGRAPH A($t_i$)

```
Input:R[m][n]
Output:A
1. Begin
2.     while (R[i][j]!=-1)∀i∈m,∀j∈n
3.     for each R[k][q] (k≥i+1,0<q≤m)
```

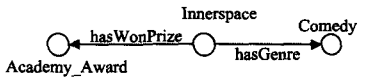


图 3　结果子图示例

示例中图查询的过程将在第 4 节进行详细描述。由于关键词查询中查询条件所表达的信息模糊，导致图查询所得结果子图的序列随机(如表 2 所列)，因此对结果评分排序很有必要。第 5 节提出一个综合排序方法，第 6 节通过实验验证其有效性。

## 4　图查询算法

为提高图查询效率，对网络社团探测算法(Network-Motif Detection)中的迭代过程进行改进[10]，提出基于迭代的图查询算法(Iterative way to SubGraph Retrieval，ISGR)。该算法利用图论中边的邻接表，从数据图中检出所有的结果子图。在图模型基础上，给出 ISGR 相关定义。

给定 RDF 数据图 $E=\{t_1,t_2,\cdots,t_n\}$，$t_j=\langle s_j,p_j,o_j\rangle$ 为一个三元组，以及关键词查询 $Q=\{k_1,k_2,\cdots,k_m\}$($k_i$ 为单个关键词)。$E$ 是未连通查询图(Query Graph)[2]，每个三元组作为图 $E$ 中的边(Edge)，元组的主体、客体作为结点(Node)。

**定义 1**(关联矩阵)　构造一个 $m$ 行 $n$ 列的稀疏矩阵 $R[m][n]$，假设共有 $m$ 个关键词，给数据图 $E$ 中每个三元组编号，每个关键词命中的三元组集合记为列表 $L_i$，“命中”是指关键词 $k_i$ 出现在三元组 $t_j$ 的 $s_j$ 和 $o_j$ 中。对 $R[m][n]$初始化，如果 $t_j\in L_i$，则 $R[i][j]=j$($j$ 为三元组的编号)，否则 $R[i][j]=-1$。初始化后的稀疏矩阵 $R[m][n]$即为关联矩阵。

例如，表 1 给定的 RDF 数据片段的初始化结果如下所示：

```
4.      if(s_i=s_j ‖ s_i=o_j ‖ o_i=s_j ‖ o_i=o_j)
5.    {
6.        if(R[i][j]∉L_k&R[k][m]∉L_i)
7.        {
8.            A[Aa]+=R[k][q]+",";
9.            R[k][q]=-1;
10.         }else R[i][j]=-1;
11.   }
12. End
```

算法 1 遍历图中所有边，构造关联矩阵 $R[m][n]$，时间复杂为 $O(n)$。算法 2 遍历关联矩阵 $R[m][n]$，判断 $t_j$ 是否与 $t_i$ 相邻，若相邻则把边 $t_j$ 加入 $A(t_i)$。算法 2 的 6－10 行确保子图的唯一性，使所构建结果子图的各条边分别匹配不同的关键词。一旦有边加入当前子图，同时将该边的邻居结点加入邻接表并继续迭代，以确保所有检出子图是最大的，算法时间复杂度为 $O(n^2)$。文献[9]中的子图匹配算法，最坏情况下的时间复杂度为 $O(n^m)$，相比之下本算法在执行时间方面有较大优势。

## 5 综合排序方法

在结果排序阶段，本文提出一种基于结构相似度和语言模型的排序方法(Sorting in Similarity and Language Models，SimLM)，考虑到关键词图与结果子图之间的结构信息，根据图特性提出度量策略，得出相似度得分；并借助统计语言模型 SLM 对最接近用户意图的结果子图得出相关度得分，对每个结果子图得出一个二者线性联合的总分，降序排列输出 Top-$k$ 个结果。过程中涉及的主要符号及其含义如表 3 所列。

表 3 主要符号含义对照表

| 符号 | 含义 |
|---|---|
| $\|V\|$ | 结点集合 V 的基数 |
| n×m | 两个向量 n 和 m 的叉积 |
| $[A_S]_{pq}$ | 矩阵 $A_S$ 的第 p 行 q 列元素值 |
| vec(X) | 对矩阵 X 进行向量操作，将矩阵中的列转化为向量 |
| A⊗B | 矩阵 A 和 B 的克罗内克积 |
| $\|Z\|_2$ | 向量 Z 的欧氏范数 |
| $G^T$ | 矩阵 G 的转置矩阵 |

### 5.1 图间元素的相似度度量策略

许多应用都需要对两图间的相似度进行定量分析，本文中关键词图与结果子图之间的相似度也需定量分析，以便从图结构方面得出与关键词图更相似的结果并进行排序。现有的相似度度量方法[11-13]大多基于如下理论：

存在图 $G_A$ 中结点 $i$ 与图 $G_B$ 中结点 $j$，若各自图中的邻居结点相似，那么 $i$ 和 $j$ 也是相似的。推广到边的度量中，如果图 $G_B$ 中边 $p$ 的起点和终点与图 $G_A$ 中边 $q$ 的起点和终点彼此相似，那么这两条边也是相似的。

在计算图中元素的相似度得分时应用该理论，即在每个迭代时间内，图中元素的相似度得分沿着邻居元素传播。本文提出使用图间相似度度量方法来计算关键词图与结果子图之间的结构相似度，以得到更加理想的排序结果。下面介绍图间元素的相似度度量策略，首先介绍相关概念。

**定义 3**(源(终)-边矩阵) 设 $s_A(q)$ 为图 $G_A$ 中边 $q$ 的源点，$t_A(q)$ 为图 $G_A$ 中边 $q$ 的终点，结点集合的势(基数) $|V_A|$ 可以用 $n_A$ 来表示，边集合的势(基数) $|E_A|$ 可以用 $m_A$ 来表示，由此图 $G_A$ 的邻接结构可以用成对的 $n_A \times m_A$ 矩阵表示，源-边矩阵 $A_S$ 和终-边矩阵 $A_T$ 可定义如下：

$$[A_S]_{pq}=\begin{cases}1, & s_A(q)=p\\ 0, & \text{otherwise}\end{cases};[A_T]_{pq}=\begin{cases}1, & t_A(q)=p\\ 0, & \text{otherwise}\end{cases}$$

**定义 4**(图间元素相似度得分矩阵) 设矩阵 $X$ 表示图 $G_A$ 和图 $G_B$ 的结点相似度得分矩阵 $SimX$，则元素 $x_{ij}$ 为图 $G_B$ 中结点 $i$ 和图 $G_A$ 中结点 $j$ 的结点相似度得分；设矩阵 $Y$ 表示图 $G_A$ 和图 $G_B$ 的边相似度得分矩阵 $SimY$，则 $y_{pq}$ 为图 $G_B$ 中边 $p$ 和图 $G_A$ 中边 $q$ 的边相似度得分。这里借鉴文献[11]中结点与边得分的求解方法，经过 $k$ 次迭代后得分公式如下所示：

$$x_{ij}(k)\leftarrow\sum_{t(k)=i,t(l)=j}y_{kl}(k-1)+\sum_{s(k)=i,s(l)=j}y_{kl}(k-1) \quad (1)$$

$$y_{pq}(k)\leftarrow x_{s(p)s(q)}(k-1)+x_{t(p)t(q)}(k-1) \quad (2)$$

矩阵 $X(k)$和 $Y(k)$分别为 $k$ 次迭代后的结点与边的得分矩阵。参考文献[11]，将矩阵中的列转化为向量，定义列向量 $x(k)=vec(X(k))$，$y(k)=vec(Y(k))$，进一步求得列向量为：

$$x(k)\leftarrow(A_S\otimes B_S+A_T\otimes B_T)y(k-1)\equiv G^T y(k-1) \quad (3)$$

$$y(k)\leftarrow(A_S^T\otimes B_S^T+A_T^T\otimes B_T^T)x(k-1)\equiv Gx(k-1) \quad (4)$$

其中 $k=0,1,2,\cdots$，若 $A$ 是一个 $m\times n$ 的矩阵，而 $B$ 是一个 $q\times p$的矩阵，$A$ 和 $B$ 的克罗内克积则是一个规模为 $mp\times nq$ 的矩阵。

由此可知，求出得分向量 $x$ 和 $y$ 是一个迭代过程。这里给出文献[13]中的定理 2，并根据 $x(k)$和 $y(k)$的初始状态 $y_0=\alpha Gx_0$(其中 $x_0$ 是元素值全部为 1 的向量，$\alpha$ 为一非零常数)对式(3)和式(4)进一步求解。

**定理 1** 设矩阵 $M$ 为一非负的对称矩阵，$z_0$ 是一个分量式的、维度合适的正向量：

$$z_0=x_0>0,z_{k+1}=\frac{Mz_k}{M_{z_{k2}}},k=0,1,2,\cdots \quad (5)$$

将 $y_0=\alpha Gx_0$ 代入，求得矩阵 $X(k)$和 $Y(k)$的列向量分别为：

$$x(k)\leftarrow\begin{cases}(G^TG)^{k/2}x_0=\dfrac{1}{\alpha}(G^TG)^{(k-2)/2}G^Ty_0, & k\text{ 为奇数}\\ (G^TG)^{(k-1)/2}G^Ty_0, & k\text{ 为偶数}\end{cases} \quad (6)$$

$$y(k)\leftarrow\begin{cases}(G^TG)^{k/2}y_0, & k\text{ 为奇数}\\ (G^TG)^{(k-1)/2}G^Ty_0=(GG^T)^{(k-1)/2}\dfrac{1}{\alpha}y_0, & k\text{ 为偶数}\end{cases} \quad (7)$$

矩阵 $X(k)$和 $Y(k)$可表示为：

$$SimX=X(k)=\{x_1(k),x_2(k),\cdots\} \quad (8)$$

$$SimY=Y(k)=\{y_1(k),y_2(k),\cdots\} \quad (9)$$

对应文中的关键词图与结果子图的全局相似度得分可由近似方法进行估算，取图间元素相似度的平均值作为全局相似度得分，如下式所示：

$$Score_{SIM}=\frac{1}{n_An_B}\sum_{i=0}^{n_An_B}SimX+\frac{1}{m_Am_B}\sum_{j=0}^{m_Am_B}SimY \quad (10)$$

如图 4 所示，对关键词图 $Q=\{k_1,k_2,k_3\}$及其中某一结果子图 $G$ 进行相似度度量，得出 $Score_{SIM}$ 为 0.402，结点及边的相似度得分如表 4 所列，得分值越高说明两元素间的相似度越高。

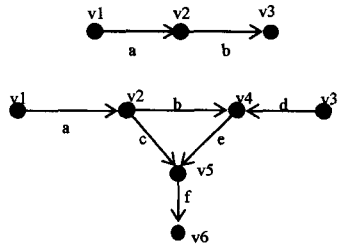


图 4　两个有向图 $Q$ 和 $G$

表 4　结点与边相似度得分

| 结点 | 结点相似度得分(SimX) | | | 边相相似度得分(SimY) | | |
|---|---|---|---|---|---|---|
| | v1 | v2 | v3 | 边 | a | b |
| v1 | 0.124 | 0 | 0 | a | 0.265 | 0 |
| v2 | 0.348 | 0.445 | 0 | b | 0.426 | 0.297 |
| v3 | 0.157 | 0.054 | 0 | c | 0.320 | 0.389 |
| v4 | 0.094 | 0.563 | 0.193 | d | 0.336 | 0.115 |
| v5 | 0 | 0.338 | 0.340 | e | 0.202 | 0.445 |
| v6 | 0 | 0 | 0.094 | f | 0 | 0.202 |

### 5.2　统计语言模型

关键词图与结果子图之间不仅有结构关联，而且有较为紧密的语义关联。因此本文利用统计语言模型 SLM[9]，将关键词图 $Q$ 与结果子图 $G$ 的相似程度用查询项似然估计概率 $P(Q|G)$ 表示，并根据概率值对结果子图排序，即选取与关键词所隐含的语义信息最接近的结果子图进行排序。为求得查询项似然估计概率 $P(Q|G)$，在数据预处理阶段构建元组虚拟文档及倒排索引。

虚拟文档用于记录元组内容信息，为 RDF 源数据的每个三元组 $t_i$ 构建一个虚拟文档 $D_i$，$D_i$ 包含一系列术语，取自三元组主体和客体，选取一些具有代表性的谓词作为术语，也可以从与三元组 $t_i$ 相关联的文本信息（如被提取元组的上下文文本信息）中提取相关术语，将其加入虚拟文档 $D_i$。例如，三元组〈Innerspace hasWonPrize Academy_Award〉分解成虚拟文档 $D_i$＝{innerspace, won, prize, academy, award}。

倒排索引用于存储术语并记录其在虚拟文档中出现的频度 $c(term, D_i)$，为每个虚拟文档建立一个倒排索引，并给文档中出现过的所有术语集合（$Col$）建立一个综合倒排索引。该技术在信息检索领域已经被广泛使用。下面介绍基于统计语言模型 SLM 的相似度度量策略，求得查询项似然估计 $P(Q|G)$。

给定一关键词查询 $Q=\{k_1, k_2, \cdots, k_m\}$，其中 $k_i$ 为单个关键词（也称术语），某一结果子图 $G=\{t_1, t_2, \cdots, t_p\}$，其中 $t_j$ 为三元组，基于查询项似然估计（Query Likelihood）对结果子图进行排序，主要是指通过计算 $P(Q|G)$，对不同子图按照概率值进行降序排列，也就是说通过计算子图能在多大程度上产生查询 $Q$ 的概率来排序。

本文考虑查询词之间为相互独立的情况，因此，查询项似然估计 $P(Q|G)$ 的计算如式(11)所示：

$$P(Q|G)=\prod_{i=1}^{m}P(k_i|G)=\prod_{i=1}^{m}\left(\sum_{j=1}^{n}\frac{1}{n}P(k_i|t_j)\right) \tag{11}$$

其中，$P(k_i|G)$ 指图 $G$ 的语言模型 LM 能产生该术语 $k_i$ 的概率，用 $k_i$ 在众多元组的语言模型中产生的概率 $P(k_i|t_j)$ 的平均值来表示（图 $G$ 有远大于 1 个的元组能够与同一术语 $k_i$ 匹配，因此对所有元组取平均值是合理的）。

为求得 $P(k_i|t_j)$，引入虚拟文档 $D_j$，用文档产生查询 $Q$ 中出现的术语的概率 $P(k_i|D_j)$ 来替代。本文 $P(k_i|D_j)$ 通过对极大似然估计值 $\frac{c(k_i, D_j)}{|D_j|}$ 进行平滑处理后求得：

$$P(k_i|D_j)=\alpha\frac{c(k_i, D_j)}{|D_j|}+(1-\alpha)\frac{c(k_i, Col)}{|Col|} \tag{12}$$

其中，$c(w, D_j)$ 为文件 $D_j$ 中术语 $w$ 的出现频度，$|D_j|$ 为文件 $D_j$ 的长度（例如，文件 $D_j$ 中所有术语频度的总和），$Col$ 为所有文档中术语所组成的集合，$|Col|$ 为整个集合的长度。而参数 $\alpha$ 是一个平滑参数，参照狄利克雷方法（Dirichlet Prior Smoothing）所设置。

将 $P(k_i|D_j)$ 代入式(11)中求得查询项似然估计 $P(Q|G)$，即为关键词图 $Q$ 和结果子图 $G$ 的语义相关度得分，记为 $Score_{P(Q|G)}$：

$$Score_{P(Q|G)}=P(Q|G)=\prod_{i=1}^{m}\left(\sum_{j=1}^{n}\frac{1}{n}\left(\alpha\frac{c(k_i, D_j)}{|D_j|}+(1-\alpha)\frac{c(k_i, Col)}{|Col|}\right)\right) \tag{13}$$

### 5.3　综合排序得分

将 5.1 节中计算得到的关键词图 $Q$ 与结果子图 $G$ 的结构相似度得分 $Score_{SIM}$ 和 5.2 节中的查询项似然估计 $Score_{P(Q|G)}$ 进行线性联合，得到对任一关键词查询 $Q$ 产生的结果子图 $G$ 的得分值，如下所示：

$$Score_G=\lambda Score_{SIM}+(1-\lambda)Score_{P(Q|G)} \tag{14}$$

其中，$\lambda$ 为可变参数，取值满足 $\lambda\in[0,1]$，用来调节结构信息和内容信息在结果打分时所占比重。计算结果子图 $G$ 的分数，对其进行降序排列，能将最优结果排在结果序列的最前面。

## 6　实验与评价

为验证本文提出的综合结构及内容信息的排序方法 SimLM 的有效性，设计实验，首先从基于结构性和内容性的两种相似度度量方法中选取单一方法与 SimLM 方法进行对比，其次选取经典的结构化数据关键词查询及排序模型，即结构化语言模型（LMs 模型）[8] 及 BANKS 系统[6]。本节分别对同一模型不同方法所产生的查询结果、不同模型所产生的查询结果从相关性和正确性两方面进行评价。

### 6.1　实验环境

实验环境配置如表 5 所列。

表 5　实验环境

| 组件 | 详细描述 |
|---|---|
| JDK 版本 | 1.6.0_31 |
| MyEclipse | 9.0 |
| OS | Windows 7 sp1 32 位 |
| CPU | Intel I3-2130 3.40G |
| 内存 | 4G |
| 网络 | 局域网 |

### 6.2　实验数据

实验数据主要分为两部分：第一部分取自两个真实数据集，并解析转换为 RDF 三元组；第二部分为查询结果的相关性评估数据。

#### 6.2.1　数据集

对两个数据集进行实验并产生查询结果，为评估排序方法提供依据。第一个数据集来源于互联网电影资料库（Internet Movie Database，IMDb），第二个数据集来自图书馆之物（Library Thing community），该网站为书籍提供在线编目。表 6 为两个数据集的主要情况。

表 6　实验所用数据集

| 数据集 | 元组数量 | 实体数量 | 实体举例 | 关系举例 |
|---|---|---|---|---|
| IMDb Dataset | 600000 | 58729 | director, actor producer, movie country, | won, actedIn directed marriedTo |
| Library Thing Dataset | 700000 | 46532 | book, author user, tag | wrote, hasFriend |

### 6.2.2　相关性评估

由于缺乏一个针对 RDF 数据查询结果的测试基准，因此很难对结果排序的质量进行评价，参考文献[9]中创建的测试基准，收集查询结果的相关性评估。该基准中包含一系列结构化的关键词查询语句，从中抽取 30 条查询语句，将其使用的关键词作为查询条件，平均分配给两个数据集。

在评估阶段，对每一个查询条件从上述各个查询模型分别产生 top-50 的已排序结果，采用众包方法将所有结果唯一且随机地呈现给 10 位计算机领域的专家，结合查询条件的描述进行判断。这里所指的结果形式均为表 2 中所列的子图。要求研究者按照如下 6 个协议阶段对结果进行评判。在收集评估结果阶段，收集所有研究者的打分信息，即对 30 个查询条件产生 15000 个唯一的相关性评估。

5 阶结果：与查询描述中所需的信息完全一致

4 阶结果：高度相关

3 阶结果：不相关但有意义，给用户提供有价值的信息

2 阶结果：无法决策

1 阶结果：不相关且毫无意义

0 阶结果：完全错误

## 6.3　一致性检验

为判断本模型与经典模型所产生的排序结果与实际结果是否具有一致性，采用一致性检验方法，并选取 Kappa 系数作为评价判断一致性程度的指标。首先从本模型提出的基于结构性和内容性的两种排序度量方法选取单一方法，分别计算 Kappa 值，同时使用二者结合的 SimLM 方法计算 Kappa 值，然后对 SimLM 方法计算 Kappa 值，得到表 7 中数据。

表 7　一致性检验

| 方法/模型 | 数据集 | |
|---|---|---|
| | IMDb Dataset | Library Thing Dataset |
| 结构性度量方法 | 0.409 | 0.382 |
| 内容性度量方法 | 0.502 | 0.429 |
| SimLM 方法 | 0.617 | 0.508 |

Fleiss 在文献[14]中规定 Kappa 值的大小与可信度的关系，如表 8 所列，并参考 TREC 会议所公布的检索任务 Kappa 值(0.34～0.54)。如表 7 所列，SimLM 方法在两个数据集上所得 Kappa 值分别为 0.617 和 0.508，从表 8 可信度分布可知，方法 SimLM 基本符合一致性检验中的理想情况，本文模型得到的结果序列与实际结果序列具有较高的相关性。表 7 数据表明，单一的度量方法排序性能明显不如二者结合的 SimLM 方法，例如结构性度量方法在数据集 Library Thing 上的 Kappa 值仅为 0.382，可信度并不理想，说明单独一种度量方法使结果序列产生偏差，二者结合的 SimLM 方法性能更优。通过对两个数据集分别进行验证，可知在不同的数据集上查询结果的相关性并不相同，即对不同种类的知识库查询效率各有不同。

表 8　可信度分布表

| Kappa 值 | 可信度 |
|---|---|
| 0.40 以下 | 可信度不合格 |
| 0.41～0.60 | 可信度中等 |
| 0.61～0.80 | 可信度优 |
| 0.81～1.0 | 完全可信 |

结构化语言模型 LMs 与 SimLM 方法对比实验所得结果如表 9 所列。相比 LMs 模型，本文方法考虑了关键词图与结果子图的结构化信息，因此得出更优的查询排序性能，结果序列的相关性明显优于 LMs 模型。

表 9　一致性检验

| 方法/模型 | 数据集 | |
|---|---|---|
| | IMDb Dataset | Library Thing Dataset |
| SimLM 方法 | 0.617 | 0.508 |
| LMs 模型 | 0.533 | 0.476 |

## 6.4　相关性评价

在对不同模型的排序性能度量中，DCG(Discounted Cumulative Gain)作为一种对搜索引擎或相关结果有效性的度量，较为广泛地使用在查询结果相关性评价策略[8,9]中。由于不同查询条件的结果数量不同，不同查询结果的 DCG 值无法进行对比。为解决规范化问题，本文使用 NDCG[15]值代替 DCG 值对查询结果进行相关性评价。

专家对结果子图序列打分，分数值按照与实际结果相关度在 5 分到 0 分之间取值，对每一个结果序列的得分进行平均，将平均值作为某一结果的相关度得分，设为 $rel_i$($i=1,2,\cdots,p$)，$p$ 为当前结果序列的结果个数，因此 DCG 可以用式(15)描述，其中当结果数为 1 时，$DCG(1)=rel_1$。由于不同查询条件的子图结果数量不同，因此不同查询条件的 DCG 值无法进行对比。考虑到正规化问题，定义 NDCG(Normalized DCG)如下：

$$\begin{cases} DCG(p)=rel_1+\sum_{i=2}^{p}\dfrac{rel_i}{\log_2 i} \\ NDCG(p)=\dfrac{DCG(p)}{IDCG(p)} \end{cases} \tag{15}$$

其中，IDCG(Ideal DCG)是指理想的 DCG。IDCG 的计算方式为对已查询出的结果人工进行排序，排到最好的状态后，算出理想结果序列下的 DCG，即为 IDCG。由于 NDCG 是一个相对比值，不同的查询条件之间可以通过比较 NDCG 来统一度量查询性能。在输出结果为 top-5、top-10、top-20 3 个数量级时，本文 SimLM 方法与 LMs 模型及 BANKS 系统的 NDCG 对比值如表 10、表 11 所列。

表 10　最优参数下的平均 NDCG 值

| 数据集(DataSet) | 模型/方法 | NDCG top-5 | NDCG top-10 | NDCG top-20 |
|---|---|---|---|---|
| IMDb Dataset | SimLM 方法 | 0.821 | 0.785 | 0.689 |
| | LMs 模型 | 0.791 | 0.754 | 0.667 |
| | BANKS 系统 | 0.569 | 0.560 | 0.513 |
| Library Thing Dataset | SimLM 方法 | 0.912 | 0.896 | 0.874 |
| | LMs 模型 | 0.889 | 0.880 | 0.861 |
| | BANKS 系统 | 0.710 | 0.734 | 0.762 |

表 11　相同数据集的平均 NDCG 值

| 模型/方法 | NDCG top-5 | NDCG top-10 | NDCG top-20 | NDCG top-50 |
|---|---|---|---|---|
| SimLM 方法 | 0.853 | 0.834 | 0.779 | 0.767 |
| LMs 模型 | 0.840 | 0.817 | 0.764 | 0.751 |
| BANKS 系统 | 0.639 | 0.647 | 0.637 | 0.642 |

(下转第 249 页)

human action recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(1): 221-231

[9] Mobahi, Hossein, Collobert R, et al. Deep learning from temporal coherence in video[C]// Proceedings of the 26th Annual International Conference on Machine Learning. ACM, 2009: 737-744

[10] Karpathy, Andrej, et al. Large-scale video classification with convolutional neural networks[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2014

[11] Memisevic R. Learning to relate images[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(8): 1829-1846

[12] Pollock D S G. Circulant matrices and time-series analysis[J]. International Journal of Mathematical Education in Science and Technology, 2002, 33(2): 213-230

[13] Taylor G W, Fergus R, Lecun Y, et al. Convolutional learning of spatio-temporal features[M]// Computer Vision--ECCV 2010. Springer, 2010: 140-153

[14] Le Q V, Zou W Y, Yeung S Y, et al. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis[C]// 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2011: 3361-3368

[15] Bregonzio M, Xiang T, Gong S. Fusing appearance and distribution information of interest points for action recognition[J]. Pattern Recognition, 2012, 45(3): 1220-1234

(上接第 239 页)

如图 5 所示，30 组查询语句在输出结果数量分别为 top-5、top-10、top-20、top-50 时，3 种不同模型的平均 NDCG 值各有差异，各个模型的参数均被设为最优值，如本模型中权重参数 $\alpha$ 取 0.9，$\lambda$ 取 0.25；LMs 模型中 $\beta$ 取 0.9；BANKS 系统中 $\gamma$ 取 1。从图 5 中得出，应用相同的现实世界数据集，本文中 SimLM 方法的 NDCG 值明显高于其他模型及系统，排序效果较为优异，例如在 IMDb 数据集条件下，进行单侧配对 $T$ 测试（$p$ 值=0.047），SimLM 方法的 NDCG 统计值比 BANKS 系统高 17%以上。

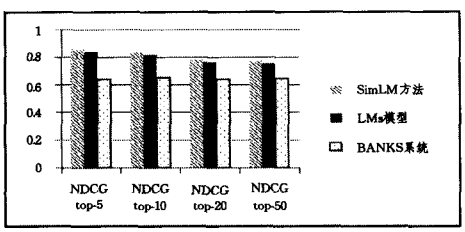


图 5 相同数据集的平均 NDCG 值

**结束语** 关键词查询是 RDF 图数据管理的重要组成部分。针对传统查询模型效率低下、结果冗余与偏离等问题，本文提出一种基于迭代的图查询算法 ISGR 对所查询关键词进行子图匹配，在降低时间复杂度的同时，能保证得到的每个结果子图唯一且最大；在结果排序阶段，提出一种结合 RDF 图数据结构性和内容性的综合排序方法 SimLM，考虑关键词图与结果子图之间的结构信息，并借助统计语言模型 SLM 对结果子图进行综合打分，按分数降序输出 Top-$k$ 个的结果。实验结果表明，本模型能够较快速、有效地完成用户对数据中的所有关键词（包括实体和关系名）的查询需求。通过对比，SimLM 方法的结果可信度符合实际需求，排序效果明显优于 LM 模型及 BANKS 系统。本文的后续工作考虑将 RDF 图的关键词查询和元组模式查询相结合，并设计统一框架来解决针对 RDF 图数据的查询问题。

## 参 考 文 献

[1] 李江华，时鹏，胡长军. 本体搜索与排序方法研究综述[J]. 小型微型计算机系统，2013，34(10)：2396-2406
Li Jiang-hua, Shi Peng, Hu Chang-jun. Overview of ontology search and ranking[J]. Journal of Chinese Computer Systems, 2013, 34(10): 2396-2406

[2] 杜方，陈跃国，杜小勇. RDF 数据查询处理技术综述[J]. 软件学报，2013，24(6)：1222-1242
Du Fang, Chen Yue-guo, Du Xiao-yong. Survey of RDF query processing techniques[J]. Journal of Software, 2013, 24(6): 1222-1242

[3] Tran T, Wang H, Rudolph S, et al. Top-k exploration of query candidates for efficient keyword search on graph-shaped (rdf) data[C]// IEEE 25th International Conference on Data Engineering (ICDE'09). IEEE, 2009: 405-416

[4] Ladwig G, Tran T. Combining query translation with query answering for efficient keyword search[M]// The Semantic Web: Research and Applications. Springer Berlin Heidelberg, 2010: 288-303

[5] Bhalotia G, Hulgeri A, Nakhe C, et al. Keyword searching and browsing in databases using BANKS[C]// Proceedings of 18th International Conference on Data Engineering, 2002. IEEE, 2002: 431-440

[6] Zou L, Özsu M T, Chen L, et al. gStore: a graph-based SPARQL query engine[J]. The VLDB Journal, 2013, 23(4): 565-590

[7] Nie Z, Ma Y, Shi S, et al. Web object retrieval[C]// Proceedings of the 16th international conference on World Wide Web. ACM, 2007: 81-90

[8] Elbassuoni S, Ramanath M, Schenkel R, et al. Language-model-based ranking for queries on RDF-graphs[C]// Proceedings of 18th ACM Conference on Information and Knowledge Management. ACM, 2009: 977-986

[9] Elbassuoni S, Blanco R. Keyword search over RDF graphs[C]// Proceedings of 20th ACM International Conference on Information and Knowledge Management. ACM, 2011: 237-242

[10] Wernicke S. A faster algorithm for detecting network motifs[M]// Algorithms in Bioinformatics. Springer Berlin Heidelberg, 2005: 165-177

[11] Zager L A, Verghese G C. Graph similarity scoring and matching [J]. Applied Mathematics Letters, 2008, 21(1): 86-94

[12] Kleinberg J M. Authoritative sources in a hyperlinked environment[J]. J ACM, 1999, 46(5): 614-632

[13] Blondel V D, Gajardo A, Heymans M, et al. A measure of similarity between graph vertices: Applications to synonym extraction and web searching[J]. SIAM review, 2004, 46(4): 647-666

[14] Wallenstein S, Zucker C L, Fleiss J L. Some statistical methods useful in circulation research[J]. Circulation Research, 1980, 47(1): 1-9

[15] Järvelin K, Kekäläinen J. IR evaluation methods for retrieving highly relevant documents[C]// Proceedings of 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2000: 41-48