一种支持用户偏好的RDF模糊查询方法

王海荣 马宗民 程经纬

(东北大学信息科学与工程学院 沈阳 110819)

摘 要 RDF模糊查询是实现语义Web智能检索的重要组成部分,利用Zadeh的Ⅱ型模糊集合理论、α-截集及语言变量概念,提出了支持用户偏好的RDF模糊查询方法,其扩展了SPARQL语言来实现模糊及偏好表达,构造了有序语言值子域表来实现模糊值到相应子域的映射,以确定隶属度区间。利用去模糊化规则,将扩展的查询转换为标准SPARQL,利用现有的SPARQL查询引擎实现模糊查询操作。为验证提出的方法,开发了fp-SPARQL实验系统。实验结果表明,该方法提高了RDF模糊查询效率,增强了用户对查询结果的满意度。

关键词 Ⅱ型模糊集理论,语言变量,模糊查询,SPARQL,fp-SPARQL

中图法分类号 TP391 **文献标识码** A

Approach for Querying RDF with Fuzzy Conditions and User Preferences

WANG Hai-rong MA Zong-min CHENG Jing-wei

(School of Information Science & Engineering, Northeastern University, Shenyang 110819, China)

Abstract RDF fuzzy retrieval is an important module for realizing intelligent retrieval in Semantic Web. In this paper, Zadeh's type-II fuzzy set theory, as well as the concepts of α-cut set and linguistic variable was adopted to put forward the RDF fuzzy retrieval mechanism supporting user preference, which extends SPARQL to express fuzzy and preference conditions. Moreover, ordered sub-domain table of linguistic values was constructed to realize the projection from the fuzzy values to relayed sub-domains in the table, so as to figure out the interval of membership. On this basis, extended queries were then converted into standard SPARQL queries with a set of defuzzification rules, so as to achieve fuzzy retrieval operations. In order to test the ideology proposed in this paper, the fp-SPARQL retrieval system was developed. According to the result of this experiment, the method improves the performance of RDF fuzzy retrieval, and correspondingly, users' satisfaction rate on the retrieval results is also enhanced.

Keywords Type-II fuzzy set theory, Linguistic variables, Fuzzy query, SPARQL, fp-SPARQL

1 引言

万维网的创始人Berners-Lee在1998年提出语义Web(Semantic Web)的概念。语义Web是对现有万维网的一种扩展,目标是使计算机能够更好地理解Web信息、支持基于语义的信息搜索与导航以及数据整合与自动化处理等[1]。语义Web应用的关键技术是资源描述框架RDF(Resource Description Framework)[2]和RDF查询语言SPARQL[3]。随着语义Web的发展,RDF数据呈现爆炸式增长趋势,有效的RDF查询已经成为实现语义Web目标的重要组成部分。应当看到的是,现有SPARQL只支持RDF数据的精确查询,而实际应用中信息获取有时需要通过模糊查询得到近似结果[4],目前已有一些研究工作致力于RDF的模糊查询。Guéret等[5]提出了基于进化计算的RDF查询方法,Huang等[6]提出了基于RDF数据库的松弛查询方法。为实现RDF多模糊属性查询的个性偏好,Cheng等[7]依据文献[8]提出的基于关系数据库的模糊查询转换方法,构建了一种模糊查询方法f-SPARQL。该方法通过给定隶属度实现RDF模糊查询,利用Top-k排序算法实现用户偏好查询。Jin等[9]提出了一种结合用户主观权重的多模糊属性的排序机制,它通过量化的权重方法表达用户对多个模糊属性的权重度量,利用排序机制实现用户个性查询需求。

用户在使用精确的量化方式表达模糊查询时通常会有一定的困难,为实现自然、直观的模糊查询操作,达到以用户为中心的交互目的,提出一种支持用户偏好的模糊SPARQL查询方法(fp-SPARQL,fuzzy query with preference-SPARQL)。该方法构建有序语言值子域表,扩展SPARQL解修饰符,并调用"偏好函数"计算偏好度,可实现多目标权重的偏好查询。经测试,fp-SPARQL方法能很好地满足RDF模糊及偏好查询需要,其操作方式符合用户自然的查询习惯。

本文第2节介绍了与RDF模糊查询相关的基本概念,重点介绍模糊集相关理论;第3节提出并详细阐述了fp-SPARQL方法,并对其语法结构进行了描述;第4节描述了fp-SPARQL模糊查询转换方法;第5节构建了一个实验系统,通过典型查询实例对比分析了f-SPARQL和fp-SPARQL方法,验证了fp-SPARQL方法的可行性;最后是对支持用户偏

到稿日期:2012-10-09 返修日期:2013-03-27 本文受国家自然科学基金项目(61073139)资助。

王海荣(1977—),女,博士生,讲师,主要研究方向为语义Web RDF模糊查询,E-mail:cleverbh@126.com;马宗民(1965—),男,博士,教授,博士生导师,主要研究方向为智能数据与知识工程。

好的 RDF 模糊查询方法的总结，并探讨了未来可能的研究方向。

2 基本概念

自 Zadeh 提出模糊集合理论以来，出现了许多基于该理论的模糊查询方法，现将其中用到的相关概念进行简要介绍。

定义 1(经典模糊集)[10] 设 X 是一个论域，X 的一个模糊子集 A 是由隶属度函数 μ_A 决定的，μ_A 的定义域是 X，值域是[0,1]，即 $\mu_A: X \to [0,1]$。

对任意 $x \in X$，$\mu_A(x)$ 称为 x 属于 A 的程度。一般情况下 A 可定义为：$A=\{(x,\mu_A(x)) \mid x \in X\}$。

隶属函数的确定是处理模糊对象的首要任务，实际应用中根据求解问题的特征选择并使用适合的隶属函数。

经典模糊集表达不确定事物具有局限性，为实现用户不同精度要求的模糊查询，本文利用Ⅱ型模糊集来描述语言层次上的模糊表达。

定义 2(Ⅱ型模糊集)[10] 假设 A 是论域 U 中的一个模糊子集，而 A 的隶属函数 $\mu_A(\chi)$ 是区间[0,1]上的模糊子集，这样的集合被称为Ⅱ型模糊集合。

模糊信息处理最直接的方法是把模糊概念精确化，这一处理通常需借助阈值 $\alpha(0 \leqslant \alpha \leqslant 1)$ 来确定隶属关系，而 α-截集就是将经典模糊集转化为精确集的有效方法。

定义 3(α-截集)[10] 若 $A \in F(\chi)$，而 $\alpha \in [0,1]$，记 $A_\alpha = \{\chi \in X \mid \mu_A(\chi) \geqslant \alpha\}$，称 A_α 为模糊集 A 的 α-截集(α-cut set)。若 $A_\alpha = \{\chi \in X \mid \mu_A(\chi) > \alpha\}$，则称其为 A 的强 α-截集(strong α-cut set)，用 $A_{\bar{\alpha}}$ 表示。

模糊概念的表达通常使用的是凸模糊集。

定义 4(凸模糊集)[10] 如果 $\forall \chi_1, \chi_2 \in X$，$\forall \lambda \in [0,1]$，有 $A(\lambda\chi_1+(1-\lambda)\chi_2) \geqslant \min(A(\chi_1), A(\chi_2))$，则称 A 是凸模糊集。

设 A,B 为论域 U 上的两个凸模糊集，A_α 和 B_α 为 A,B 上的 α-截集，记 $A_\alpha=[a,b]$，$B_\alpha=[c,d]$，根据 α-截集定理($A \cup B)=A_\alpha \cup B_\alpha$，$(A \cap B)=A_\alpha \cap B_\alpha$ 可得如下定理：

$$A_\alpha \cup B_\alpha = \begin{cases} [a,c] \cup [b,d], & \text{如果 } A_\alpha \cap B_\alpha = \Phi \\ [\min(a,c), \max(b,d)], & \text{如果 } A_\alpha \cap B_\alpha \neq \Phi \end{cases} \quad (1)$$

$$A_\alpha \cap B_\alpha = \begin{cases} \Phi, & \text{如果 } A_\alpha \cap B_\alpha = \Phi \\ [\max(a,c), \min(b,d)], & \text{如果 } A_\alpha \cap B_\alpha \neq \Phi \end{cases} \quad (2)$$

为实现自然方式的模糊概念表达，本研究以 Zadeh 提出的语言变量概念为基础，提出了一种基于语言变量的模糊查询方法。该方法以近似的方式采用模糊集合而不是精确数来描述模糊查询条件。

定义 5(语言变量)[10] 一个语言变量是一个五元组(x, U,$W(x)$,G,M)，其中，x 是变量名称；U 是语言变量 x 取值的论域；$W(x)$ 表示语言值(即语言变量 x 的取值)全体构成的集合；G 是生成规则(也称语法规则)，用以产生 x 语言值的名称；M 是语义规则，即对 $W(x)$ 中的每个语言值定义一个 U 上的模糊集，也就是由其隶属函数表征 $W(x)$ 中的每个语言值。

3 模糊 SPARQL 语法

SPARQL 是 RDF 查询标准，其基本查询由 3 部分构成：SELECT 子句指定了在查询结果中出现的变量；FROM 子句限定了查询的数据源；WHERE 子句描述了图匹配的基本模式。应当指出的是，经典的 SPARQL 为 RDF 的精确查询提供了便利，但其不支持现实应用中广泛存在的模糊及多模糊属性的偏好查询。为满足用户查询中语义模糊性及个性化偏好两方面的需求，本文提出了支持偏好的模糊 SPARQL 查询方法(简称 fp-SPARQL)。在分析了语言值有序特点的基础上，构造了有序语言值子域表，从而使 fp-SPARQL 能够通过使用语言变量描述模糊 SPARQL 查询要求。此外，fp-SPARQL 增加了解修饰符“preference”，以实现模糊 SPARQL 的偏好表达。

3.1 模糊 SPARQL 结构

根据有序语言值特点，基于Ⅱ型模糊集理论，采用梯型隶属函数，构造了 13 个有序语言值子域，每个子域的模糊值使用一个四元组(a_i,b_i,α_i,β_i)来描述，其中 a_i 和 b_i 分别为子域隶属度区间的下限值和上限值，α_i 和 β_i 为每一个子域与其上、下相邻子域的模糊调整距离。13 个有序语言值的描述如表 1 所列。

表 1 有序语言值子域表

子域标识符	模糊术语	模糊值
S_{12}	昂贵、绝对高、绝对多、Absolutely high 等	(1.0,1.0,0.0,0.0)
S_{11}	极贵、极高、极多、Extremely high 等	(0.90,1.0,0.0,0.5)
S_{10}	非常贵、非常高、非常多、Very high 等	(0.81,0.95,0.05,0.04)
S_9	贵、高、多、High 等	(0.71,0.85,0.04,0.05)
S_8	有点贵、有点高、有点多、Fairly high 等	(0.62,0.76,0.05,0.04)
S_7	稍微有点贵、稍微有点高、稍微有点多、Somewhat high 等	(0.52,0.66,0.04,0.05)
S_6	不贵也不便宜、不高也不低、不多也不少、适中、Medium 等	(0.43,0.57,0.05,0.04)
S_5	稍微有点便宜、稍微有点低、稍微有点少、Somewhat low 等	(0.33,0.47,0.04,0.05)
S_4	有点便宜、有点低、有点少、Fairly low 等	(0.24,0.38,0.05,0.04)
S_3	便宜、低、少、Low 等	(0.14,0.28,0.04,0.05)
S_2	非常便宜、非常低、非常少、Very low 等	(0.05,0.19,0.05,0.04)
S_1	极便宜、极低、极少、Extremely low 等	(0.0,0.9,0.04,0.0)
S_0	绝对便宜、绝对低、绝对少、Absolutely low 等	(0.0,0.0,0.0,0.0)

依据有序语言值子域表，可对标准 SPARQL 进行模糊扩展。首先，允许在 FILTER 子句中增加模糊术语和语言值子域标识。其次，为了能够清楚地表达用户查询的模糊程度，在 FILTER 中还需要增加“with S_i”子句来指定模糊量词查询限制范围，其中 S_i 为全局变量。例如，查询“工作量非常重的教师信息”，可表示为“FILTER(?count=“非常重”) with S_{10}”。对于上述模糊 SPARQL 查询，系统将根据用户提交的模糊查询请求，自动提取请求中的语言值并映射到表 1 中的相应子域，从而确定隶属度区间并生成查询表达式。

包含上述扩展子句的模糊 SPARQL 查询语句其结构形式如下：

```
SELECT ?object
WHERE {?object hasproperty ?property.
    FILTER(?property=FuzzyTerm) with Si. }
```

3.2 偏好 SPARQL 结构

在模糊查询的基础上，为满足用户个性偏好检索的需求，fp-SPARQL 方法引用了解修饰符“preference”来扩展 SPARQL 对偏好的描述，以实现支持偏好的模糊查询。通过调用偏好度函数(见定义 6)实现偏好度的计算，依据偏好度降序

排列，以用户期望的顺序返回结果。

定义 6(偏好度函数)[9] 设 r_i 为模糊查询属性 P_i 的偏好权重特征量，令模糊查询条件中的属性个数为 k，则 $\sum_{i=1}^{k}\gamma_i=1,\gamma_i\in[0,1]$。假设 $\gamma_1\leqslant\cdots\leqslant\gamma_k$，则偏好度

$$\begin{aligned}&\omega(p_1(e_1),\cdots,p_k(e_k))\\&=(\gamma_1-\gamma_2)\cdot f(p_1(e_1))+2(\gamma_2-\gamma_3)\cdot f(p_1(e_1),p_2(e_2))+3(\gamma_3-\gamma_4)\cdot f(p_1(e_1),p_2(e_2),p_3(e_3))+\cdots+k\gamma_k\cdot f(p_1(e_1),p_2(e_2),\cdots,p_k(e_k))\\&=k\lambda_k\cdot f(p_1(e_1),\cdots,p_k(e_k))+\sum_{i=1}^{k-1}i(\gamma_i-\gamma_{i+1})\cdot f(p_1(e_1),\cdots,p_i(e_i))\end{aligned}\tag{3}$$

其中，聚合函数为：$f(p_1(e_1),\cdots,p_k(e_k))=AVERAGE(p_1(e_1)+p_2(e_2)+\cdots+p_k(e_k))$ 或 $f(p_1(e_1),\cdots,p_k(e_k))=\min(p_1(e_1),p_2(e_2),\cdots,p_k(e_k))$。

偏好度为[0,1]区间的实数，其值越接近 1，则表示记录越满足用户个性查询目标。为清晰地表达多属性模糊查询中用户的偏好权重及偏好度，fp-SPARQL 方法中定义了 prefer(numeric r_i)和 score(numeric ω_i)两个函数，分别用于计算多条件模糊查询中的偏好权重和偏好度。

支持用户偏好的模糊 SPARQL(简称为 fp-SPARQL)查询语句其结构形式如下：

```
xsd:double fun:prefer(numeric r_i)
xsd:double fun:score(numeric ω_i)
SELECT ?object
WHERE {?object hasproperty ?property..
  FILTER(?property 〈comparison operator〉 typed-literal)}
    PREFERENCE
      ?object fun:prefer(r_i)
      ?object fun:score(ω_i)
    ORDER BY DESC(?score)
    LIMIT n
```

在上述结构中，函数 $prefer=f(p_1(e_1),\cdots,p_k(e_k))$ 是一个聚合函数，实现偏好查询中 k 个模糊属性隶属度的聚合运算(根据目标对象特点可为最小值 min、平均值 average、累加和 sum 等)，参数 r_i 为多模糊属性查询中用户偏好属性(用户查询中最关注的那个属性)的权重，此函数返回值为一个 0 到 1 区间的实数。本文提出如下公式来设置多模糊属性的偏好权重：

$$\gamma_{偏好属性}=\frac{2}{count()+1},\gamma_{非偏好属性}=\frac{1}{count()+1}\tag{4}$$

式中，$r_{偏好属性}$ 为多属性查询条件中用户最关心的偏好属性的权重，$count()$ 统计查询条件中的属性个数，$r_{非偏好属性}$ 为偏好属性外的其它属性的权重。

函数 $score(numeric\ \omega_i)$ 实现偏好度计算，在完成模糊查询之后，调用式(3)计算结果集中每条记录的偏好度，其中参数 ω_i 为记录的偏好度，偏好度为一个 0 到 1 区间的实数。LIMIT 用于控制返回结果数量，n 为用户设定的需要返回的结果数量。

3.3 fp-SPARQL 语法

在实际应用中，用户查询请求可能含有多个模糊条件，处理时可通过逻辑运算符("and","or","not")连接查询条件实现操作。fp-SPARQL 语法如表 2 所列。

表 2 fp-SPARQL 语法

SelectQuery	::=	'SELECT' ('DISTINCT' | 'REDUCED')? (Var+ | '*') DatasetClause * WhereClause SolutionModifier
WhereClause	::=	'WHERE' GroupGraphPattern
Filter	::=	'FILTER' Constraint
Constraint	::=	?(Var+'='+FuzzyTerm)[with S_i]
SolutionModifier	::=	PREFERENCEClause? OrderClause? LimitClauses? OffsetClauses?
LimitOffsetClauses	::=	(LimitClause OffsetClause? | OffsetClause LimitClause?)
LimitClause	::=	'LIMIT' INTEGER
PREFERENCEClause	::=	'Preference' ? (var+FuzzyFunctionCall)
FuzzyFunctionCall	::=	IRIref ArgList
ArgList	::=	(NIL | '(' fun:fuzzyfunname(',' Expression) * ')')
OrderClause	::=	'ORDER' 'BY' OrderCondition+
OrderCondition	::=	(('ASC' | 'DESC') BracketedExpression) | (Constraint | Var)

4 fp-SPARQL 查询处理

fp-SPARQL 具有直观描述模糊查询条件及偏好的能力，但其查询处理的实现还需要去模糊化处理，从而将含有偏好的模糊 SPARQL 转换成标准 SPARQL，并依据用户偏好对查询结果进行排序。

4.1 模糊查询条件及转换方法

对于 RDF 数值型属性上的模糊查询条件，本文将表征模糊特性的语言变量值区分为表示程度和表示范围两类情况分别进行讨论。

4.1.1 表示程度的模糊查询条件转换

自然语言中，有些词如"很"、"稍许"、"极"等作为前缀加在一个词的前面便调整了该词词义的肯定程度，把原来的词变为一个新词，这类词称为语气算子。由语气算子构成的模糊术语用于模糊查询，则构成表示程度的模糊查询。例如，语言变量 x 为"价格"，加上语气算子后便调整为一组表示程度的模糊术语 $W(x)=$｛昂贵，极贵，非常贵，贵，适中，有点便宜，便宜，非常便宜｝。此类模糊术语是有序语言值[11]，处理时可将 $W(x)$ 中的每个术语映射到表 1 中的相应子域，从而确定模糊术语的隶属度区间，该区间是[0,1]上的数据集。

表示程度的模糊查询转换过程如下：

1)提取用户模糊查询请求中表示程度的模糊术语，映射到相应子域，求出阈值 α 的隶属度区间。如查询条件中包含"非常重"这个模糊术语，将其映射到表 1，提取相应子域 S_{10}，则可求得表示模糊程度的阈值 α 的隶属度区间为[0.81,0.95]，进而得到：[非常重]$=S_{10}=\{x,\mu_A(x)\mid\mu_A(x)\in\alpha,\alpha\in[0.81,0.95]\}$。

2)调用梯型隶属度函数求得查询字段的取值范围。假设查询条件中的语言变量为"工作量"，且其取值区间为 $n\in[2,30]$，要查询"工作量非常重"的教师，则调用梯型隶属度函数式(5)，确定阈值 α 的隶属度区间，求得"工作量非常重"的取值范围为 $22\leqslant n\leqslant29$。

$$\mu_A(x)=\begin{cases}0, & x\leqslant a\\ \dfrac{x-a}{b-a}, & a<x\leqslant b\\ 1, & x>b\end{cases}\tag{5}$$

3)生成标准的 SPARQL 查询表达式。

4.1.2 表示范围的模糊查询条件转换

模糊表达常用的还有一类词，如"接近"、"大约"、"至多"、

“不少于”等，放在精确量词前将确切词义模糊化，形成模糊量词，则构成了表示范围的模糊查询。模糊量词一般可分为绝对模糊量词和相对模糊量词。

• 绝对模糊量词：一个精确数的模糊化表示，可表示为“大约 Y”、“至少 Y”、“至多 Y”、“接近 Y”、“非常接近 Y”等形式(Y 为一个精确数)。例如，“大约 3000”、“至少 3000”。

• 相对模糊量词：一个取值范围不确定的数值型属性的模糊化表达，如“至少一半”、“最多一半”等。

表示范围的模糊术语一般用凸模糊集(见定义 4)隶属函数来定义。根据查询条件中表示范围的模糊变量特点，现将 fp-SPARQL 中数值型模糊属性的查询分为绝对模糊量词及相对模糊量词查询两类，本文重点讨论绝对模糊量词的查询方法。

绝对模糊量词处理时，可根据具体的模糊查询条件把绝对模糊量词查询处理分为两种情况：

1)极值处理：模糊量词中出现的上限或下限约束条件(例如“至多 Y”或“至少 Y”)，可直接转换为“$\leqslant Y$”或“$\geqslant Y$”。

2)范围处理：模糊量词中出现的范围约束条件(例如“大约 Y”或“接近 Y”)，可利用表 1 进行模糊化的隶属区间映射，之后，选取隶属函数确定取值区间，转换查询条件。以“大约 Y”为例，利用 Cauchy 分布的中间型隶属函数的扩展方法[12]式(6)，根据映射区间隶属度的左边缘值求得 α-截集右半部分的边缘值，进而确定右部取值范围，之后利用凸模糊集的对称性，确定左部取值范围，最终求得语言变量的取值范围。

$$\mu_{\text{大约}Y}(x)=\frac{1}{1+(\frac{x-y}{\beta})^2} \tag{6}$$

式中，y 为目标对象，是一个精确数；β 为曲线宽度的调整值，根据 β 可调整查询精度范围。

例 1　假设在某高校教师本体上查询“工资大约 3000 元的教师”，相应的 fp-SPARQL 查询形式如下：

```
SELECT ?p ?Name ?Count ?Salary
WHERE {?p ex:hasSalary ?Salary.
  FILTER(?Salary=“大约 3000”)}
```

经过转换，得到如下标准的 SPARQL 语句：

```
SELECT ?p ?Name ?Count ?Salary
WHERE {?p ex:hasSalary?Salary.
  FILTER(?Salary>=2568 && ?Salary<=3432)}
```

4.1.3　fp-SPARQL 去模糊化转换规则

表 3　fp-SPARQL 部分转换规则

模糊查询条件(FQC)	语言标识符子域	精确查询条件(QC)	备注
?x=“至少 Y”	/	FILTER(?x≥Y)	
?x=“至多 Y”	/	FILTER(?x≤Y)	
?x=“接近 Y”	S_6-S_{11}	FILTER (?x≥a && ?x≤b)	
?x=“有点接近 Y”	S_8-S_{11}	FILTER (?x≥a && ?x≤b)	
?x=“大约 Y”	$S_{10}-S_{11}$	FILTER (?x≥a && ?x≤b)	
?x=“极多”	S_{11}	/	依据表 1 确定阈值区间求 α-截集，确定下限、上限值
? x=“有序语言值”	参见表 1	/	依据表 1 确定阈值区间，确定变量取值范围，实现转换

数值型属性的 RDF 模糊查询最终要转换成标准 SPARQL，则需要一套转换规则来实现模糊查询的自动转换。fp-SPARQL 基于语言变量的去模糊化转换规则如表 3 所列。

根据此转换规则可将模糊条件精确化，进而实现 SPARQL 查询操作，从而更好地满足用户模糊查询需求。

4.2　基于偏好的 fp-SPARQL 查询排序

fp-SPARQL 的偏好查询，使用定性的方法实现多模糊查询条件中的偏好表达，采用偏好属性排序机制实现查询。首先，进行模糊查询，形成初始结果集；其次，调用式(4)计算各查询属性的偏好权重；最后，调用偏好度函数式(3)计算初始结果集中每条记录的偏好度，以偏好度为关键字降序排列结果并返回给用户。

例 2　假设在某高校教师本体上查询“工作量重且工资大约 3000 元的教师，更偏重于工作量重的教师”，相应的 fp-SPARQL 查询描述如下：

```
xsd:double fun:prefer(numeric r_i)
xsd:double fun:score(numeric ω_i)
SELECT ?name ?count ?salary
WHERE {?p ex:hasCount ?count.
  FILTER(?count=“非常重”)with S_i
  ?p ex:hasSalary ?salary.
  FILTER(?salary=“大约 3000”)}
  PREFERENCE
    ?object fun:prefer(r_工作量)
    ?object fun:score(?ω_工作量)
  ORDER BY DESC(?score)
```

假设依据模糊查询条件首先得到如表 4 所列的初始结果集。对于结果集中的每条记录，分别计算其偏好度，进而实现查询结果的排序。为此，分别计算“工作量”和“工资”的偏好权重，得到：

$$\gamma_{\text{工作量}}=\frac{2}{count()+1}=\frac{2}{3},\gamma_{\text{工资}}=\frac{1}{count()+1}=\frac{1}{3}$$

之后将各属性偏好权重代入偏好度计算公式：

$$\omega_{\text{工作量}}=(\gamma_{\text{工作量}}-\gamma_{\text{工资}})\times\mu_{\text{工作量重}}(i)+2\times\gamma_{\text{工资}}\times\frac{\mu_{\text{工作量重}}(i)+\mu_{\text{工资大约3000}}(i)}{2}=(\frac{2}{3}-\frac{1}{3})\times\mu_{\text{工作量重}}(i)+2\times\frac{1}{3}\times\frac{\mu_{\text{工作量重}}(i)+\mu_{\text{工资大约3000}}(i)}{2}$$

式中，i 为初始结果集中的标号。

表 4　模糊查询初始结果集

编号	姓名	工作量	工资(元)
1	王强	22	2800
2	张玲	28	3100
3	周涛	28	2600
4	李华	26	3000
5	刘超	24	2800

最终经过排序的查询结果如表 5 所列。

表 5　最终结果集

编号	姓名	工作量	工资	偏好权重
2	张玲	28	3100	0.91
4	李华	26	3000	0.91
3	周涛	28	2600	0.88
5	刘超	24	2800	0.84
1	王强	22	2800	0.79

5　实验系统

本文使用 Eclipse 工具开发了一个可运行于 Windows

XP 平台的 fp-SPARQL 实验系统。本系统以 LUBM[13] 为测试框架，使用 Protégé 工具扩展了 univ_Bench 本体，测试数据集总共包含 6800000 多个三元组，添加了 200 多个测试数据，调用 Jena 工具包实现本体文件的解析和操作，构造了多个查询实例进行方法验证。

fp-SPARQL 系统包含 3 个模块：用户查询接口，接收用户查询请求并返回查询结果的可视化界面；SPARQL 查询处理，实现用户模糊查询请求的规范化转换并执行 SPARQL 查询；排序，对初始的无序结果进行偏好度计算，根据偏好度降序排列，以满足用户偏好查询需要。系统结构如图 1 所示，运行主界面如图 2 所示。

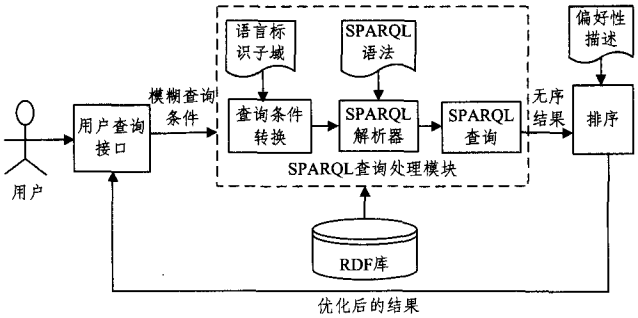

图 1　fp-SPARQL 实验系统结构图

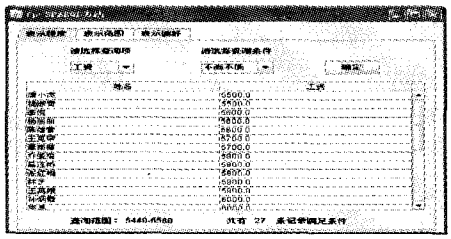

图 2　fp-SPARQL 系统主界面

为验证本文提出的方法，我们选择了与本文工作相近的 f-SPARQL 方法（用户给出隶属度及偏好权重的方法）进行比较，分别从模糊查询和偏好查询两个方面进行对比分析，以验证两种方法的执行效率与可用性。结果如表 6 所列。

表 6　典型实例查询结果分析表

fp-SPARQL				f-SPARQL	
查询实例	隶属区间	去模糊查询条件	运行时间(ms)	查询实例	运行时间(ms)
工作量极重	[0.9,1.0]	27≤工作量	30	工作量=0.9	230
工作量不轻也不重	[0.43,0.57]	15≤工作量≤18	30	工资=0.5	560
工作量极轻	[0.09,0.0]	工作量≤6	25	工资=0.2	26

对比分析两种方法的操作方式及运行结果发现，f-SPARQL在查询时先要将 RDF 文件存入数据库，且每执行一次查询都要重新读取本体文件，导致执行效率较低。此外，要求用户给出隶属度或偏好权重的操作方式导致系统操作复杂性提高，并且基于一个给定的隶属度值的查询其结果的召回率较低。fp-SPARQL 方法使用语言的方法描述模糊性，符合普通用户的使用习惯，且查询时一次性将数据读入内存，提高了系统的执行效率，因查询时将模糊值扩展到一个范围内进行检索，其召回率也大大提高。此外，通过偏好权重及偏好度的计算，使用近似的算法减少了用户因调整结果而造成的操作复杂性，提高了执行效率，进而增强了方法的可用性。

综合来看，fp-SPARQL 方法具有查询方便、执行效率高、支持多条件偏好查询等优势，在很大程度上提高了用户对 RDF 模糊查询的满意度。

结束语　RDF 模糊查询是实现定性语义信息检索的有效方法，本文重点讨论了数值型 RDF 模糊查询方法，提出了支持偏好的 RDF 模糊检索方法 fp-SPARQL，开发了实验系统，构建了若干查询实例。通过对比分析查询结果表明，此方法执行效率高、操作简单、查询方式更符合用户习惯，一定程度上提高了用户对定性问题查询处理的满意度。以本文为基础，下一步我们将深入研究相对模糊量词的查询优化问题，并将 RDF 文本类型属性的模糊查询作为研究重点。

参 考 文 献

[1] 叶育鑫，欧阳丹彤. 语义 Web 搜索技术研究进展 [J]. 计算机科学，2010，37(1)：1

[2] 陆建江，张亚非. 语义网原理与技术 [M]. 北京：科学出版社，2007：32-40

[3] 高志强，潘越. 语义 Web 原理及应用 [M]. 北京：机械工业出版社，2009：60-63

[4] Hogan A, Mellotte M, Powell G, et al. Towards Fuzzy Query-Relaxation for RDF [C]// 9th Extended Semantic Web Conference. Berlin Heidelberg: Springer-Verlag, 2012: 687-702

[5] Guéret C, Oren E, Schlobach S, et al. An Evolutionary Perspective on Approximate RDF Query Answering [J]. Scalable Uncertainty Management, 2008, 5291: 215-228

[6] Huang Hai, Liu Cheng-fei, Zhou Xiao-fang. Approximating query answering on RDF databases [J]. World Wide Web, 2012, 15(1): 89-114

[7] Cheng Jing-wei, Ma Zong-min. f-SPARQL: A Flexible Extension of SPARQL [C]// 21st Database and Expert Systems Applications. Berlin Heidelberg: Springer-Verlag, 2010: 487-494

[8] Ma Zong-min, Yan Li. Generalization of strategies for fuzzy query translation in classical relational databases [J]. Information & Software Technology, 2007, 49(2): 172-180

[9] Jin Hai, Ning Xiao-min, Jia Wei-jia, et al. Combining weights with fuzziness for intelligent semantic web search [J]. Knowledge-Based Systems, 2008, 21(2008): 655-665

[10] 胡宝清. 模糊理论基础 [M]. 武汉：武汉大学出版社，2010

[11] Herrera-Viedma E. Modeling the Retrieval Process for an Information Retrieval System Using an Ordinal Fuzzy Linguistic Approach [J]. Journal of The American Society For Information Science And Technology, 2001, 52(6): 460-475

[12] Chen Shyi-ming, Jong W-T. Fuzzy Query Translation for Relational Database Systems [J]. IEEE Transactions on Systems, 1997, 27(4): 714-721

[13] Guo Yuan-bo, Pan Zheng-xiang, Heflin J. An evaluation of knowledge base systems for large owl datasets [C]// Third International Semantic Web Conference. Berlin Heidelberg: Springer-Verlag, 2004: 274-288