软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2019,30(5):1510−1521 [doi: 10.13328/j.cnki.jos.005726]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

一种基于迭代的关系模型到本体模型的模式匹配方法*

王 丰[1,2], 王亚沙[1,3,4], 赵俊峰[1,2,4], 崔 达[1,2]

[1](高可信软件技术教育部重点实验室(北京大学),北京 100871)

[2](北京大学 信息科学技术学院,北京 100871)

[3](软件工程国家工程中心(北京大学),北京 100871)

[4](北京大学(天津滨海)新一代信息技术研究院,天津 300450)

通讯作者: 王亚沙, E-mail: wangyasha@pku.edu.cn

摘 要: 语义网的飞速发展,使得各领域出现了以本体这种形式来表达的知识模型.但在实际的语义网应用中,常常面临本体实例匮乏的问题.将现有关系型数据源中的数据转化为本体实例是一种有效的解决办法,这需要利用关系模型到本体模型的模式匹配技术来建立数据源和本体之间的映射关系.除此之外,关系模型到本体模型的模式匹配还被广泛用于数据集成、数据语义标注、基于本体的数据访问等领域中.现有的研究工作往往会综合使用多种模式匹配算法,计算异构数据模式中元素对的综合相似度,辅助人工建立数据源到本体的映射关系.现有的工作针对单一模式匹配算法准确率不高的问题,试图通过综合多种模式匹配算法的结果来进行调和.然而,这种方法当多种匹配算法同时出现不准时,难以得出更加准确的最终匹配结果.对单一模式匹配算法匹配不准的成因进行深入的分析,认为数据源的本地化特征是导致这一现象的重要因素,并提出了一种迭代优化的模式匹配方案.该方案利用在模式匹配过程中已经得到匹配的元素对,对单一模式匹配算法进行优化,经过优化后的算法能够更好地兼容数据源的本地化特征,从而显著提升准确率.在“餐饮信息管理”领域的一个实际案例上开展实验,模式匹配效果显著高于传统方法,其中,F 值超过传统方法 50.1%.

关键词: 模式匹配;迭代优化;本地化特征

中图法分类号: TP311

中文引用格式: 王丰,王亚沙,赵俊峰,崔达.一种基于迭代的关系模型到本体模型的模式匹配方法.软件学报,2019,30(5):1510−1521. http://www.jos.org.cn/1000-9825/5726.htm

英文引用格式: Wang F, Wang YS, Zhao JF, Cui D. Iterative-based relational model to ontology schema matching approach. Ruan Jian Xue Bao/Journal of Software, 2019,30(5):1510−1521 (in Chinese). http://www.jos.org.cn/1000-9825/5726.htm

Iterative-based Relational Model to Ontology Schema Matching Approach

WANG Feng[1,2], WANG Ya-Sha[1,3,4], ZHAO Jun-Feng[1,2,4], CUI Da[1,2]

[1](Key Laboratory of High Confidence Software Technologies (Peking University), Ministry of Education, Beijing 100871, China)

[2](School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China)

[3](National Engineering Research Center for Software Engineering (Peking University), Beijing 100871, China)

[4](Peking University Information Technology Institute (Tianjin Binhai), Tianjin 300450, China)

Abstract: The rapid development of the semantic web makes the various fields in smart city have emerged in the form of ontology to express the knowledge model. However, in the practical semantic Web application, it is often faced with the problem of lack of ontology

* 基金项目: 国家重点研发计划(2017YFB1002002); 国家自然科学基金(61772045)

Foundation item: National Key Research and Development Program of China (2017YFB1002002); National Natural Science Foundation of China (61772045)

本文由智能化软件新技术专刊特约编辑申富饶教授和李戈副教授推荐.

收稿时间: 2018-08-31; 修改时间: 2018-10-31, 2018-12-14; 采用时间: 2019-02-03

instance. It is an extremely effective solution to transform the data in the existing relational data source into ontology instance, which requires the use of the relational model to the ontology model matching technology to establish the mapping between the data source and the ontology. In addition, the schema matching to the ontology model is widely used in data integration, data semantic annotation, ontology-based data access, and other fields. The existing related work tends to use a variety of schema matching algorithms to calculate the similarity of element pairs in heterogeneous data patterns. However, when multiple matching algorithms fail at the same time, it is difficult to obtain a more accurate final matching result. In this study, the weekness of the matching of the single schema matching algorithm are analyzed deeply, the localization feature of the data source is an important factor leading to this phenomenon, and an iterative optimization schema matching scheme is proposed. The scheme uses the matched element pairs from matching process to optimize the single schema matching algorithm. The optimized algorithm can be better compatible with the localization features of the data source, with much higher accuracy, and more matching elements can be obtained. The process continues to iterate until the end of the match. In this study, experiments are carried out through a practical case in the fields of "food information management" which have shown that the proposed approach significantly outperforms state-of-the-art method by increasing up to 50.1% of *F*-measure.

Key words: schema matching; iterative optimization; localization feature

语义网(semantic Web)作为下一代互联网规范,在促进数据交互和知识共享等方面具有重大意义.本体是语义网的核心,是特定领域共享概念模型的形式化规范说明,被广泛地用于刻画特定领域的知识模型[1,2].但是在实际的语义网应用中,常常面临本体实例匮乏的问题,考虑到现有的城市系统中,大量的实例数据的主流存储方式仍然是关系型数据库[3],将关系数据库中结构化数据转化为本体实例能够有效地对领域本体实例进行扩充.为了实现这种转化,首要任务是建立关系数据源中的模式到本体中概念的映射关系;此外,建立这种映射关系的需求,还广泛存在于数据集成、数据语义标注、基于本体的数据访问等多个与本体密切相关的领域.

关系数据模式到本体映射关系的建立,是一类典型的模式匹配问题[4−6].所谓模式匹配问题,指的是在不同的数据模式中找出语义相同或相似的元素对,并构造映射关系的一类问题[7,8],即建立数据库表到本体中类的映射以及数据库表中字段到本体类的属性的映射.图 1 所示是一个关系数据库模式到本体的模式匹配示例.

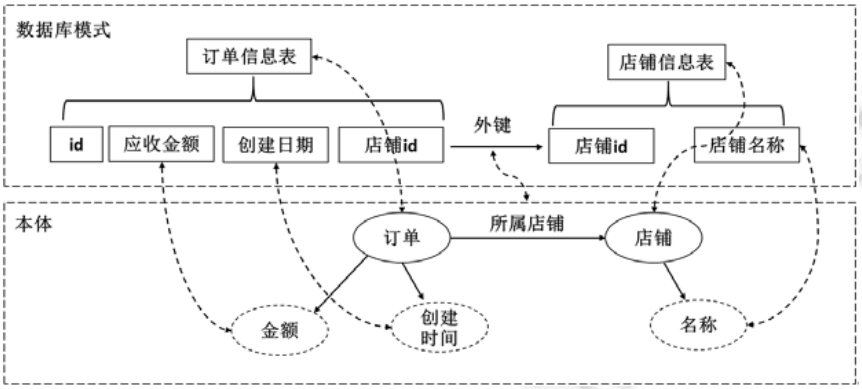


Fig.1 Example of schema matching

图 1 模式匹配示例

人工地进行模式匹配工作过于费时费力,且容易出现误差.为了降低人力成本,提高模式匹配的准确率,研究者们提出了许多自动化方法和框架,通过利用模式本身的元信息或者元素所含实例特征等信息,计算得到不同模式之间元素对的相似度,来辅助人工来进行模式匹配.根据所利用的信息的不同,可以划分为若干种基本模式匹配算法,例如使用元素标签信息的基于字符串的匹配算法、使用实例统计特征的基于统计的匹配算法等.单一的模式匹配算法只考虑元素在某个特征上的相似度,在应用时可能会出现匹配不准确的情况,所以现有的模式匹配框架往往会使用多种模式匹配算法,综合考虑元素对在多个特征上的相似性,来获得更为全面准确的结果,然而,这种做法并未从本质上分析导致匹配准确率不高的原因,当多种模式匹配算法均存在较大偏差时,仍然无法得到准确的综合结果.

针对以上问题,本文从数据源本地化特征的角度分析了单一模式匹配算法匹配不准的原因.数据源的本地化特征,主要体现在两个方面.

- 一方面是由各数据源模式独立、自主设计而导致的模式结构的本地化.

例如在关系数据库模式设计时,设计者一般不会参照同领域中其他数据源的模式,也不会刻意使用领域中标准化的专业术语来建立数据字典,而是根据其对业务的理解独立完成数据库模式设计.而本体与某个特定的数据库模式不同,是表达了领域共性知识的规范化说明.数据库模式与本体这一本质区别,必然导致了本体和实际的关系数据模式存在差别,而在术语的使用上体现得尤为明显.为了解决术语使用的差异化问题,现有的基于字符串的模式匹配算法常常需要同义词词典来解决这个问题.而特定业务领域的术语和同义词,并不一定包含在已有的通用的同义词词典中,故而匹配准确率低.

- 本地化特征的另一方面是由于业务特征差异导致的数据实体统计特征的本地化.

真实数据源中,其实例的统计特征往往与其服务的业务相关.以餐饮领域的收银管理相关数据为例,主营商务宴会的餐饮品牌,其每单金额的均值和方差都远远高于主营地方小吃的餐饮品牌.本体模型中包含的实例,往往是由多个数据源中的数据转化而来,其实例数据的统计特征反应了不同餐饮品牌的综合平均值,与主营商务宴会或地方小吃的餐饮品牌的实际数据,其实例统计特征都存在存在较大差异,从而导致基于数据实例统计特征的匹配算法失效.

综上所述,数据源的本地化特征是导致数据源在模式和实例上与本体存在较大差异的重要原因,而由于在进行模式匹配前无法预先确悉数据源的本地化特征,直接应用模式匹配算法时则必然导致匹配不准确的情况.

针对数据源存在的本地化特征的客观情况,本文提出一种迭代优化的模式匹配方案,其基本思想是:利用在模式匹配过程中得到的一部分匹配的元素对来对各单一模式匹配算法进行优化,从而提高单一算法的准确性,最终提高整个模式匹配过程的准确率.具体地,本文对两种典型的模式匹配算法——基于字符串的模式匹配算法以及基于实例的模式匹配算法进行优化.对于已经得到匹配的元素对,其标签可以看作是一对同义词,自动加入到同义词词典中,基于字符串的模式匹配算法利用该自动生成的同义词词典,就能够兼容数据源在术语使用上的本地化;已经得到匹配的元素对,其实例统计特征可以作为一种匹配知识,作为训练集进行训练,得到一个分类模型,该分类模型由于吸纳了先前匹配的经验,故而可以很好地兼容数据源在实例上的本地化特征.

本文以餐饮信息管理领域的一个实际案例开展模式匹配实验,并与现有的相关工作进行对比,证明了本文模式匹配算法的有效性和准确性.本文的主要贡献如下.

(1) 以餐饮系统为例对数据源的本地化特征进行了分析,分析了本地化特征的种类与其产生的原因.

(2) 提出一种迭代优化的模式匹配方法 IOSMA(iterative optimization schema matching algorithm),算法在迭代过程中,利用已经匹配成功的元素对优化模式匹配算法,使模式匹配算法随着迭代可以逐渐取得更好的匹配效果.

(3) 在餐饮数据集上进行了测试,结果显示,本文提出的迭代优化的模式匹配算法效果优于基线算法.

本文第 1 节介绍现有的模式匹配算法与模式匹配框架.第 2 节详细分析数据源本地化特征的原因和对模式匹配算法的影响.第 3 节详细介绍本文的模式匹配方案.第 4 节介绍实验设计和实验结果.第 5 节对本文进行总结.

1 相关工作

1.1 模式匹配算法

模式匹配算法衡量不同数据模式的元素对在某种特征上的相似性,对输入的两个来自不同数据模式中的元素,输出一个[0,1]区间上的实数值作为相似程度.经过调研,本文对基本的模式匹配算法的分类进行总结,如图 2 所示.

模式匹配算法按照所利用的信息的不同,可以分为基于模式信息的模式匹配算法和基于实例的模式匹配算法两个大类.

- 基于模式信息的模式匹配算法关注于数据模式的元信息,数据模式的元信息包括组成数据模式的基本元素以及这些基本元素之间的关系.基于元素的模式匹配算法利用元素本身的信息来判断元素的相似性,例如通过字符串的编辑距离来计算元素标签的相似性[9,10],通过数据类型的兼容性判断元素对匹配的可能性.基于结构的模式匹配算法利用数据模式中元素之间的关系来判断元素的相似性,例如通过子节点相似度计算父节点相似度的子节点匹配算法[9,10],或者将输入的源模式转换为有向图或者树的形式,然后利用路径匹配[11]、图匹配[12−14]等已经成熟的图算法计算元素对的相似度.
- 基于实例的模式匹配算法关注元素所含实例的特征,分为基于语言学的实例匹配算法和基于统计的实例匹配算法.
  - 基于语言学的实例匹配算法主要面向数据类型为字符串的元素,通过抽取实例文本的关键词[9,10,15],然后比较关键词的相似性;或者以一个现有的知识库为基础,在其中寻找与该元素最符合的概念作为映射[15],然后比较元素对所映射概念之间的相似性.
  - 基于统计的实例匹配算法主要有两种:一种利用两个元素实例上的重合度作为相似度,极易造成漏判;另一种更为主流,主要通过实例的各种统计量上的相似度,例如平均值、最大值、最小值、方差等,来计算元素对的相似度[16,17].

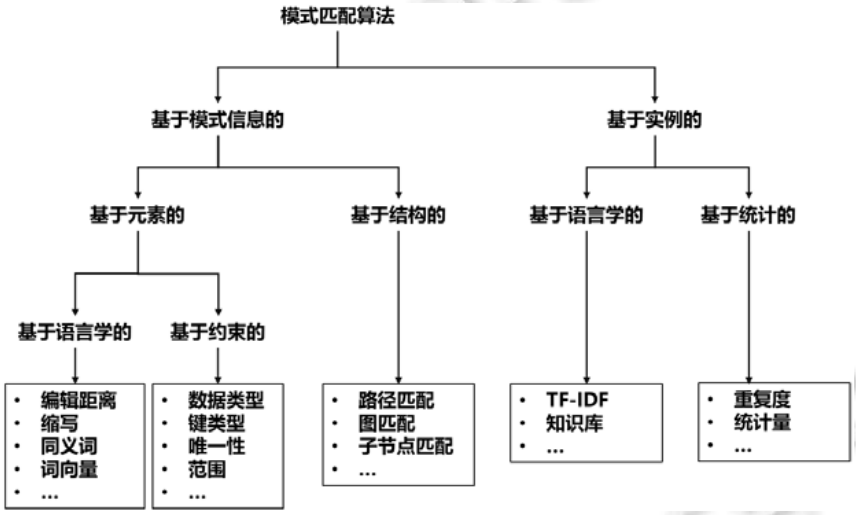


Fig.2 Classification of schema matching algorithm

图 2 模式匹配算法分类

由于用于匹配的数据源在模式信息和实例上存在一定程度的本地化特征,单一地利用模式信息或者实例来进行相似度的计算必然会出现匹配不准确的情况.此外,这些模式匹配算法虽然具有一定程度上的通用性,但是不具备自学习能力,且难以兼容数据源的本地化特征.本文借鉴了传统的模式匹配算法,综合并优化了多种已有的模式匹配算法,在迭代的过程中,利用已匹配的信息来优化传统的模式匹配算法,在不断的迭代中,提高传统匹配算法在具有本地化特征的数据上的正确率.

1.2 模式匹配框架

上一节讨论了多种利用单一特征的模式匹配算法,这些模式匹配算法单独使用极易引起误差,因而现有的模式匹配框架往往采用多种模式匹配算法相结合的方式,例如 SEMINT[17]、COMA++[9]、RONTO[3]等,其中,COMA++、RONTO 支持关系模型到本体模型的匹配.本文对一般的关系模型到本体模型的模式匹配框架进行总结,其一般流程如图 3 所示.该流程对于输入的两个异构数据模式中的元素对的处理,主要包含 3 个阶段.

(1) 相似度计算阶段:在这一阶段,调用多种模式匹配算法,对输入的来自于两个不同数据模式的元素对,

计算其在多个特征维度上的相似度.所选的模式匹配算法和执行的先后顺序可以由人工来进行配置.

(2) 相似度综合阶段:上一阶段得到了一个元素对的多种相似度,分别来自于所使用的各个模式匹配算法,为了对元素对的匹配进行排序、判定、筛选,还需要对这些相似度进行综合,具体的综合方式可以是加权平均,也可以是人工定义的其他规则.

(3) 相似度判定阶段:在这一阶段,两个异构模式中各个元素对的综合相似度已然计算完毕,按照一定的规则对这些元素对进行是否匹配的事实判定,可以人工地按相似度大小排序后审阅并选择,也可以根据人工设定的规则,例如阈值,来自动地加以判定.

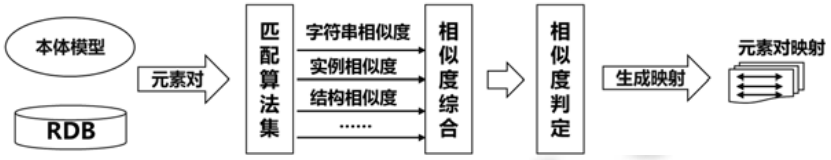


Fig.3 General process of schema matching framework

图 3 模式匹配框架的一般流程

现有的模式匹配框架通过综合利用多种模式匹配算法的结果来缓解单一模式匹配算法匹配不准确所产生的误差,但它们没有对单一模式匹配算法匹配不准确的问题进行分析与解决,因而仍然存在匹配准确率低下、需要较多人工参与等问题.本文算法利用迭代的方法对元素对进行匹配,每一轮匹配借鉴了现有的模式匹配框架的思路,在每一轮匹配完成后,部分元素得到匹配,利用已匹配的元素对可以优化模式匹配算法,通过迭代可以自动地使模式匹配框架适应具有本地化特征的数据,使一些难以匹配的元素对在迭代过程中得到匹配.

2 数据源本地化特征分析

本体被设计用于规范化地表达领域的知识模型,包含领域中概念以及概念之间的关系.由于语义网仍然处在发展阶段,在很多领域中只有本体的定义而缺乏本体的实例.而真实存在的数据源往往用来为特定的应用提供数据,对于数据存取性能方面要求较高,大多采用关系数据模型作为存储方式.

根据应用环境的不同,实际的数据源可以按照应用场景、应用系统这两个维度进行划分,见表 1.

Table 1 Partition of data sources

表 1 数据源的划分

	应用场景 1	应用场景 2
应用系统 1	*A*	*B*
应用系统 2	–	*C*
应用系统 3	*D*	–

如表 1 所示,*A*,*B*,*C*,*D* 分别是 4 个数据源,它们均属于某个特定的应用场景,例如在餐饮信息管理领域,应用场景可以是“宴会”“特色小吃”“自助”等,属于同一个应用场景的这些数据源,表达相同概念的元素,其实例特征具有高度的相似性.而属于不同应用场景的数据源,语义相同的元素,其实例特征可能差别很大,如图 4 所示,左侧是主营宴会餐饮的北京宴品牌,其某个门店数据源中订单应收金额和订单应收服务费的平均值(单位:元),右侧是主营特色小吃的热河食府品牌,其某个门店数据源中订单应收金额和服务费的平均值(单位:元).通过比较我们发现:这两个数据源中同为表示应收金额的数据元素,其平均值相差 10 倍以上.这种不同应用场景下实例特征上的差异,本文称之为实例的本地化特征.

同样地,领域中存在多个不同的应用系统,例如在餐饮信息管理领域,有“餐行健”“品智”“轩亚”等多个餐饮信息管理系统,属于同一应用系统的这些数据源,由于采用相同的数据定义,所以表达相同概念的元素,无论是元素的标签还是元素的组成结构,都完全相同.而术语不同应用系统的数据源,其所使用的数据定义可能差别很大,如图 5 所示,左侧是餐行健餐饮信息管理系统,关于“用餐区域”这一概念,其使用“section(英文)/桌台区(中文

注释)”这样的术语;而右侧的品智餐饮信息管理系统,同样的概念,使用“business_loc(英文)/营业区(中文注释)”这样的术语.在预先不知道它们描述的均是“用餐区域”这一概念的条件下,机器甚至人都无法仅凭字符串判断出“section”和“business_loc”“桌台区”和“营业区”表达的含义相同这一结论.这种不同应用系统下使用术语上的差异,本文称其为术语的本地化特征.

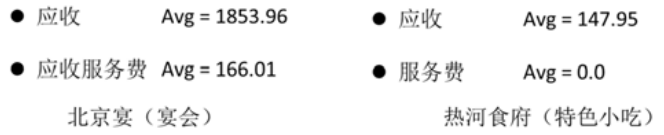


Fig.4 Instance feature under different application scenarios

图 4 不同应用场景下的实例特征

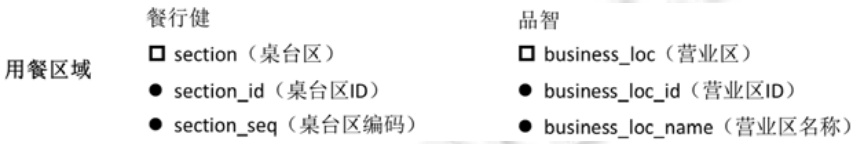


Fig.5 Concept definition under different application systems

图 5 不同应用系统下概念定义

实例和术语的本地化特征,其根源来自于不同数据源在应用场景和应用系统上的划分不同,而这种本地化特征会导致通用的模式匹配算法——基于实例统计特征的模式匹配算法和基于字符串的模式匹配算法出现匹配不准的情况.如何在领域知识本体模型无法预先明晰这种本地化特征的情况下,兼容关系型数据源所存在的本地化特征,提高模式匹配的准确度,是本文所需要解决的主要问题.

3 迭代优化的模式匹配算法

3.1 方案概述

本文针对现有的关系模型到本体模型的模式匹配框架在处理数据源的本地化特征时存在的不足,提出了一种迭代优化的模式匹配方法 IOSMA,如图 6 所示.

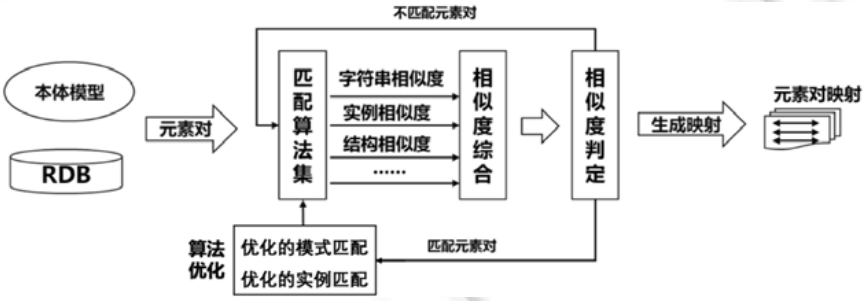


Fig.6 Iterative optimization schema matching algorithm

图 6 迭代优化的模式匹配算法

算法对于异构的元素对的基本匹配流程和现有的模式匹配框架相似,包含了相似度计算、相似度综合和相似度判定这 3 个阶段:在相似度计算阶段,本文采用了包含基于模式的相似度计算和基于实例的相似度计算,可以计算得到表与本体中类的相似度以及列与本体中的属性的相似度;在相似度综合阶段,也分为表相似度综合和列相似度综合,即对前一步得到的多种相似度进行加权平均,可以得到每个元素对之间的综合相似度;在相似度判定阶段,对与一个待匹配的数据库模式信息,会利用其和本体中所有待匹配的类或属性之间的相似度计算

信息熵,利用熵来衡量匹配的不确定性,这里,人工可以设置阈值,当不确定性小于阈值,即认为匹配成功,选择相似度最高的一组匹配作为匹配结果.当一轮匹配结束后,算法会得到匹配的元素对和不匹配的元素对.

与现有模式匹配框架不同的是,本文在相似度判定环节后引入了算法优化,并将算法流程改为迭代式的.本文方案的主要思想是:利用模式匹配过程中已经判定匹配的元素对,对原有的模式匹配算法进行优化,从而达到提高单一模式匹配算法准确率,进而提升整体的匹配准确率.原有的低于相似度阈值而无法得到匹配的元素对,重新进入匹配流程.由于模式匹配算法的改进,可以正确地得到匹配,因而得到了更多的匹配元素对用于算法优化,形成一个良性循环.而已经判定匹配的元素对能够对原有的模式匹配算法进行优化的原因在于:该元素对蕴含了本体概念和数据源元素的等价关系,数据源元素具有的本地化特征可以用本体进行标注与衡量,之后遇到具备相似本地化特征的元素时,能够更好地加以判断.

具体地,为了兼容数据源术语的本地化特征和在实例的本地化特征,本文利用已匹配的元素对,对基于字符串的模式匹配算法以及基于实例的模式匹配算法进行一定的优化.

3.2 优化:基于字符串的模式匹配算法

已经形成匹配的元素对,其语义是相同的,故而其元素标签是同义的,而同一种应用系统中,为了避免混乱,对于同一个概念,往往倾向于使用相同的标签进行表述.因而一旦能够获得一组同义词,就意味着所有包含该同义词所含字符串的元素对的相似度可以进一步优化.如图 5 所示,关于用餐区域的表述,如果在综合了多种模式匹配算法之后,能够得出两个系统中以 section 为标签的节点和以 bussiness_loc 为标签的节点表达相同含义的话,则可以将“section”和“bussiness_loc”作为一组同义词,添加到同义词词典中.之后,在进行“section_id”和“bussiness_loc_id”的匹配时,基于字符串的模式匹配算法能够给出更为准确的结果.

传统的针对英文字符串的模式匹配算法有编辑距离法、词向量距离法等方法.本文在传统的编辑距离法上结合了同义词词典,首先利用词典对字符串进行同义替换,然后消除标签对的同义部分,最后计算剩余部分的编辑距离,得出字符串的相似度.

编辑距离指的是两个字符串之间由一个转成另一个所需的最少编辑次数,编辑操作包括增加、删除、替换.与传统的编辑距离计算不同的是,对于替换操作,除了原本的字符替代以外,本文系统还允许代价为 0 的同义词替换.显然,两个字符串的编辑距离最大值即为二者长度的最大值.根据编辑距离,可以计算出两个字符串的相似度.例如,对于字符串“bill_tabls”和“order_table_cnt”,已知 bill 和 order 是同义词,将 bill 替换为 order,并且添加 4 个字符_cnt,所以编辑距离为 4,而最大编辑距离为较长字符串的长度,即 order_table_cnt 的长度 15,那么字符串的相似度为 1−5/15=0.66.

算法 1. 考虑同义词的英文字符串匹配算法.

Input:*S*1,*S*2:2 个待匹配字符串,*ThesaurusDictionary*:同义词词典.

Output:*String_Similarity*:字符串相似度.

```
1.  begin
2.    for each word∈ThesaurusDictionary do
3.      if S1 contains word
4.        then for each synonym_word∈synonymSet(word) do
5.          if S2 contains synonym_word
6.            then
7.              S1.delete(word)
8.              S2.delete(synonym_word)
9.          end if
10.       end for
11.   end for
12.   edit_distance=editDistance(S1,S2)
```

13. *max_edit_distance*=Max(*S*1.*length*,*S*2.*length*)
14. *String_Similarity*=1−*edit_distance*/*max_edit_distance*
15. **end**

对于中文字符串的匹配,不能使用类似英文字符串中求编辑距离的办法,因为表达同样信息的中文字符串长度远小于英文,稍有偏差就会差距很大.本文使用 Word2Vec 训练出领域相关的词向量模型,词向量的夹角即为两个词的相似度,夹角的大小通常使用余弦函数来衡量.

两个单词 W_i 和 W_j,其对应的词向量分别为 $V_i=\langle v_{i1},v_{i2},\ldots,v_{in}\rangle$ 和 $V_j=\langle v_{j1},v_{j2},\ldots,v_{jn}\rangle$,则单词 W_i 和 W_j 的相似度为

$$w2v\cos(W_i,W_j)=\cos(V_i,V_j)=\frac{\sum_{t=1}^{n}v_{i_t}\times v_{j_t}}{\sqrt{\sum_{t=1}^{n}v_{i_t}^2\times\sum_{t=1}^{n}v_{j_t}^2}}=\sum_{t=1}^{n}v_{i_t}\times v_{j_t}.$$

为了衡量任意两个中文字符串的相似度,首先要将两个字符串切分成一个个单词,通过计算单词间的相似度,得到整体字符串的相似度.分词工具切分出的两个单词集合分别为 *TokenList*1 和 *TokenList*2,对于 *TokenList*1 中的每个单词,在 *TokenList*2 中找相似度最大的那个单词,将该相似度进行累计,最终除以 *TokenList*1 集合的大小,即得到字符串相似度大小.

算法 2. 考虑同义词的中文字符串匹配算法.

Input:*S*1,*S*2:2 个待匹配中文字符串,ThesaurusDictionary:同义词词典.

Output:*String_Similarity*:字符串相似度.

1. begin
2. *TokenList*1←*Tokenize*(*S*1)
3. *TokenList*2←*Tokenize*(*S*2)
4. *String-Similarity*←0
5. **for each** *token*1∈*TokenList*1 **do**
6. **max_similarity**←0
7. **for each** *token*2∈*TokenList*2 **do**
8. **if** *isSynonym*(*token*1,*token*2) **then**
9. *max_similarity*=1
10. **else**
11. **if** *w*2*tSim*(*token*1,*token*2)>*max_similarity*
12. **then** *max_similarity*=*w*2*v*cos(*token*1,*token*2)
13. **end if**
14. **end if**
15. **end for**
16. *String_Similarity*+=*max_similarity*
17. **end for**
18. *String_Similarity*/=*sizeof*(*TokenList*1)
19. **end**

3.3 优化:基于实例的模式匹配算法

传统的基于实例的模式匹配算法常常假设两个具有相同语义的元素,在实例的统计特征上具有较高的相似性,例如平均值、方差、中位数等数学统计量,对于两个待匹配的元素,计算各个数学统计量的值,为每个元素生成统计特征向量,然后比较统计特征向量之间的距离,作为衡量相似度的标准.

本文主要关注了最大值、最小值、中位数、平均数、区间范围、DC(distinct count:不同值数量)、变异系

数、DC 占比、非空值占比,这些信息可以作为区分不同列的统计特征.

以 M 种不同类型的统计量作为不同的特征维度,为数据库中的每个表列,生成 M 维的向量,记为实例统计向量,由于本体中的每一个属性都会映射到至少 1 个数据库中的表列,因此其实例统计向量的计算方法与数据库表列相同.

在计算得到实例统计向量之后,一种直观的相似度计算方法是使用向量间的欧氏距离衡量元素对实例层次上的相似度,对于两个向量 $V_i=\langle v_{i1},v_{i2},\ldots,v_{in}\rangle$ 和 $V_j=\langle v_{j1},v_{j2},\ldots,v_{jn}\rangle$,欧氏距离为

$$Dist(V_i,V_j)=\sqrt{\sum_{k=1}^{n}(v_{ik}-v_{jk})^2}.$$

n 维向量的欧氏距离最大值为 $\sqrt{n}$,使用线性映射法将欧氏距离映射到[0,1]区间上作为相似度.因此对于两个数据库列 A_i 和 A_j,其对应的向量分别为 V_i 和 V_j,得到它们的相似度:

$$EuclideanSim(A_i,A_j)=1-\frac{Dist(V_i,V_j)}{\sqrt{n}}.$$

根据第 2 节的分析,实际的数据源可能存在实例本地化特征,即其数学统计量可能明显偏离于其他数据源或者本体实例相对应元素的统计量,这时,应用上述方法得到的相似度是不准确的.

本文利用已经形成匹配的元素对,加上由匹配排他性所推导出的不匹配元素对作为训练数据,生成一个分类模型,分类模型的输入是两个元素的统计特征向量,输出是这两个元素是否形成匹配的概率.分类模型随着已匹配元素的增多,训练集也不断增加,分类效果也不断增强,对实例本地化特征的兼容性也越来越好,其基本思想如图 7 所示.

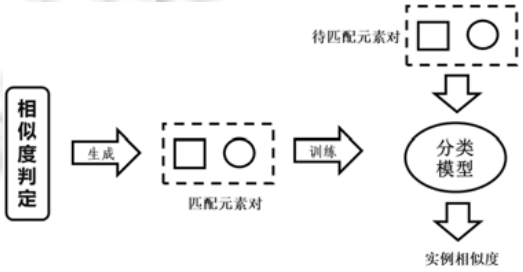


Fig.7 Classifier-based instance schema matching algorithm

图 7 基于分类器的实例模式匹配算法

但模式匹配的前期没有足够的训练集可用,因此前期主要依赖欧氏距离计算实例相似度,大致能够区分不同列即可;在匹配的过程中,元素对不断得到匹配,也给分类器提供了训练数据.匹配的后期由于拥有大量的训练数据,分类模型的准确度也得到了提升,因此得到的相似度也更为可信.因此,我们设置了参数 δ 来调整两种算法得到的相似度的比例.假设当前有 δ 比例的列得到了匹配,*EuclideanSim* 表示欧拉距离给出的相似度,*MLSim* 表示分类器给出的相似度,则最终的实例相似度为

$$InstanceSimilarity(E_1,E_2)=(1-\delta)EuclideanSim(E_1,E_2)+\delta MLSim(E_1,E_2).$$

以餐饮信息管理为例,假设当前本体中的实例数据来自于某些特色小吃餐饮品牌,而待匹配数据源的数据来源于主营宴会的餐饮品牌,起初,基于实例的模式匹配算法并不能反映出这种差异,但随着匹配元素对越来越多,分类模型获取足够多的训练数据之后,就能够对两个数据模式之间存在的差异进行学习,之后,在利用实例统计特征进行相似度判定时,就会变得更加准确.

4 实验验证

4.1 实验设定

本节对本文迭代优化的模式匹配算法进行实验验证,本体选用的是基于餐行健餐饮信息管理系统构造出

的本体模型,待匹配数据源为品智餐饮信息管理系统,本体模型和关系模型中所含元素数量的统计见表 2.

模式匹配的主要目标是寻找本体中的类和关系模式中表的映射关系、本体中的数据属性和关系模式中列的映射关系.本文的评估标准采用精确率(precision)、召回率(recall)、F 值(F-measure),其中,F 值为精确率与召回率的调和平均值,统一度量精确率与召回率.记 TP 为判断正确的匹配,FP 为判断错误的匹配,FN 为没有判断出来的正确匹配.3 个评估标准的计算方法如下:

$$Precision=\frac{TP}{TP+FP}\times 100\%,$$

$$Recall=\frac{TP}{TP+FN}\times 100\%,$$

$$F\text{-}measure=2\times\frac{Presicion\times Recall}{Precision+Recall}.$$

Table 2 Statistics of schema elements

表 2 待匹配模式元素统计

	类/表	属性/列
本体	19	136
关系模式	15	103

实验分为 3 组:第 1 组是现有的模式匹配框架 COMA++,第 2 组是未进行迭代的本文算法 NSMA(non-iterative schema matching algorithm),第 3 组是本文的迭代优化的模式匹配算法 IOSMA.在模式匹配的过程中,完全依靠机器自动完成,无任何专家参与.

4.2 匹配结果

实验结果如图 8 所示.由于没有利用已匹配元素对来对模式匹配算法进行优化,COMA++无法很好地兼容数据源的本地化特征.COMA++使用的不带优化的字符串匹配算法,没有根据已形成匹配的元素对对其自身进行改进,从而导致很多原本可以借助同义词转化提高相似度的元素对,达不到匹配阈值,从而得不到匹配.COMA++使用的基于实例的匹配算法单一地考虑统计量上的近似程度,而没有利用已经匹配的元素对所提供的知识,训练分类模型,无法很好地应对相同语义的元素对,其实例特征有较大差异的情况.相反地,IOSMA 较好地考虑了数据源的本地化特征,并对匹配算法进行迭代式的改进,ISOMA 利用已匹配的元素对在迭代过程中可以不断提高匹配效果,在实验数据集上达到了 91%的精确率、83%的召回率和 87%的 F 值.相对于 COMA++,分别取得了 39.8%、59.6%、50.1%的提升;相对于非迭代版本的算法,分别取得了 24.7%、33.9%、29.9%的提升.

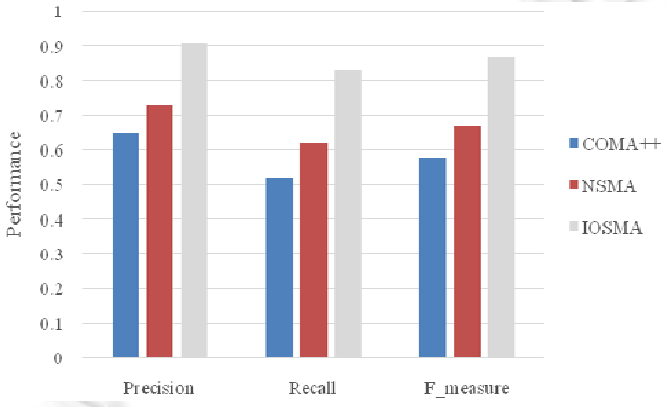


Fig.8 Result of schema matching experiments

图 8 模式匹配实验结果

4.3 案例分析

为了更好地展示本文的匹配效果,表 3 展示了利用 ISOMA 得到的已匹配的数据库表名和本体类名.在这些已匹配的元素对中,某些字段具有显著的本地化特征,在迭代过程中,这些元素对也得到了匹配.本节通过两个方面分别举例说明 ISOMA 在迭代过程中兼容数据源本地化特征的做法.

Table 3 Schema maching instance

表 3 模式匹配示例

数据库表含义	数据库表名	本体类名
商户账单表	sh_shbill	o_order_history
商户账单明细表	sh_shbilldetailed	o_order_item_history
商户账单支付表	sh_shbillpay	c_pay_history
品牌表	sys_brand	sls_brand
营业区设置表	sys_businessloc	o_section
菜品表	sys_dishes	o_dish
组织机构表	sys_organizations	sys_shop
支付方式表	sys_payment	c_pay_type
菜品类别表	sys_reportcategory	o_dish_kind
桌台表	sys_table	o_table
系统用户表	sys_userinfo	sys_user

4.3.1 处理模式结构的本地化特征

在第 1 轮匹配过程中,商户账单表得到匹配,算法可以提取出 order 和 bill 的同义词关系,形成该数据源的同义词典.同义词典可以有效地改善基于字符串的模式匹配算.例如数据库中的订单金额的名称是 bill_total,本体中的订单金额为 order_total_amount,在确定 bill 和 order 为同义词后,订单金额的相似度会得到明显的提升.从而使订单金额字段得到匹配.

4.3.2 处理实例信息的本地化特征

数据库和本体中相同含义数据的统计特征也存在一定的差异,在实验中,基于餐行健系统生成的本体模型中多为高端餐饮企业,而品智餐饮系统中则是中小餐馆居多,因此两个系统中表述相同含义的元素的统计特征(最大值、平均值、标准差等)存在较大差异(见表 4).在匹配过程中,商户账单支付表在两个系统中模式结构相似度较高,得到了匹配,利用这个信息可以得到很多匹配的元素对,利用已匹配的元素对的统计特征生成样本并训练分类器.随着分类器的训练样本增加,分类器更容易识别出数据库和本体中的统计特征差异,可以把具有类似差异的相同概念进行匹配.在本例中,商户账单明细表数据库和本体的匹配通过这种方式得到了提升.也体现 ISOMA 基于迭代更好地兼容了数据源的本地化特征.

Table 4 Difference of statistical features

表 4 统计特征差异

	餐行健			品智		
	最大值	平均值	标准差	最大值	平均值	标准差
商品单价	7 000	38.4	643.8	288	20.5	42.2
商品数量	94	15.8	20.4	38	8.4	9.5
订单价格	48 493	583.2	1 983.2	1 468	132.5	403.2

5 结 论

本文研究一类关系模型到本体模型的模式匹配问题,在充分调研了现有模式匹配相关研究工作的前提下,对实际数据源具有的本地化特征进行了深入分析和论证,并指出本地化特征是导致现有模式匹配框架中单一模式匹配算法匹配失效的深层次原因;然后提出一种迭代优化的模式匹配算法,该算法利用已经得到匹配的元素对,对传统的基于字符串的模式匹配算法和基于实例的模式匹配算法进行优化,使之更好地兼容数据源的本地化特征,从而提高了模式匹配的准确率;最后,以餐饮信息管理领域的一个实际案例开展相关实验,证明本文

算法的有效性和准确性.

本文未来有以下研究方向.首先,探究如何更加科学地综合不同匹配算法的结果.目前主流的方式都是由用户来制定相似度的综合方式,然而在很多情况下,用户也很难给出一个准确的综合相似度计算公式.为此,需要分析不同模式匹配算法的特性,例如基于实例的模式匹配算法的可信度是否高于基于字符串的模式匹配算法的可信度,根据匹配算法的可信度为其赋予相应的权重是一种简单的解决方法.而根据已经得到匹配的元素对来进行学习,利用表示学习的方式挖掘匹配元素对的深层次特征,可以得到更好的匹配结果.其次,对于匹配错误的元素对的纠错机制也是一个值得探讨的方向,首先需要进一步提高模型的准确率,其次可以加入群智,利用人机协作来对机器出现的错误进行纠正.

References:

[1] Gagnon, M. Ontology-based integration of data sources. In: Proc. of the Int'l Conf. on Information Fusion IEEE Xplore. 2007. 1–8.

[2] Wache H, *et al*. Ontology-based integration of information—A survey of existing approaches. In: Proc. of the IJCAI-01 Workshop: Ontologies and Information Sharing, Vol.2001. 2001.

[3] Papapanagiotou P, *et al*. RONTO: Relational to ontology schema matching. AIS Sigsemis Bulletin, 2006,3(3-4):32–36.

[4] Madhavan J, Bernstein PA, Rahm E. Generic schema matching with cupid. In: Proc. of the Int'l Conf. on Very Large Data Bases Morgan Kaufmann Publishers Inc. 2001. 49–58.

[5] Rahm, Erhard, Bernstein PA. A survey of approaches to automatic schema matching. The VLDB Journal, 2001,10(4):334–350.

[6] Bernstein PA, Madhavan J, Rahm E. Generic schemRONa matching, ten years later. Proc. of the VLDB Endowment, 2011, 4(11):695–701.

[7] Jiménez-Ruiz E, *et al*. BootOX: Practical mapping of RDBs to OWL 2. In: Proc. of the Int'l Semantic Web Conf. Springer Int'l Publishing, 2015.

[8] Santoso HA, Haw SC, Abdul-Mehdi ZT. Ontology extraction from relational database: Concept hierarchy as background knowledge. Knowledge-Based Systems, 2011,24(3):457–464.

[9] Aumueller D, *et al*. Schema and ontology matching with COMA++. In: Proc. of the 2005 ACM SIGMOD Int'l Conf. on Management of Data. ACM Press, 2005.

[10] Shvaiko P, Euzenat J. A survey of schema-based matching approaches. Journal on Data Semantics IV. Berlin, Heidelberg: Springer-Verlag, 2005. 146–171.

[11] Liu C, Wang JW, Han YB. Mashroom+: An interactive data mashup approach with uncertainty handling. Journal of Grid Computing, 2014,12(2):221–244.

[12] Melnik S, Garcia-Molina H, Rahm E. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In: Proc. of the 18th Int'l Conf. on Data Engineering. IEEE, 2002. 117–128.

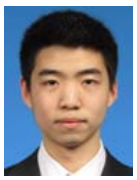
[13] Euzenat J, Valtchev P. Similarity-based ontology alignment in OWL-lite. In: Proc. of the European Conf. on Artificial Intelligence (ECAI). 2004. 333–337.

[14] Doan AH, Madhavan J, Domingos P, Halevy AY. Learning to map between ontologies on the semantic Web. In: Proc. of the WWW. 2002. 662–673.

[15] Li WS, Clifton C, Liu SY. Database integration using neural networks: Implementation and experiences. Knowledge and Information Systems, 2000,2(1):73–96.

[16] Doan AH, Domingos P, Halevy A. Reconciling schemas of disparate data sources: A machine-learning approach. In: Proc. of the ACM SIGMOD Conf. 2001. 509–520.

[17] Li WS, Clifton C. SEMINT: A tool for identifying attribute correspondences in heterogeneous databases using neural networks. Data & Knowledge Engineering, 2000,33(1):49–84.



王丰(1995－),男,河南周口人,硕士生,CCF 学生会员,主要研究领域为智慧城市.

赵俊峰(1974－),女,博士,副教授,CCF 高级会员,主要研究领域为软件工程,软件复用,Web 服务,普适计算,大数据分析技术.

王亚沙(1975－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为普适计算,大数据分析技术.

崔达(1993－),男,硕士,主要研究领域为智慧城市.