计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2018.20170664
55(11):2522-2531,2018

HSMA:面向物联网异构数据的模式分层匹配算法

郭 帅 郭忠文 仇志金
(中国海洋大学信息科学与工程学院 山东青岛 266100)
(guoshuaiouc@163.com)

HSMA: Hierarchical Schema Matching Algorithm for IoT Heterogeneous Data

Guo Shuai, Guo Zhongwen, and Qiu Zhijin
(*College of Information Science and Engineering, Ocean University of China, Qingdao, Shandong* 266100)

Abstract With the rapid development of IoT technology, everything in the world is going to interconnect via IoT, which has become a popular technology that permits users to connect anywhere, anytime, anyplace, anyone and any device, involving many domains such as smart home, intelligent transportation and so on. However, most of IoT heterogeneous data are isolated, which holds back the progress of IoT interconnection. Schema matching techniques are widely used in the scenario of data interconnection to solve the problem above. Because of the characteristics of IoT heterogeneous data such as heterogeneity and increasing growth, the existing schema matching approaches can't solve the problems caused by schema auto-matching under new IoT environment. In this paper, we attempt to solve this problem by introducing a new algorithm based on hierarchical method that could fulfill automatic schema matching for IoT heterogeneous data. Our algorithm has three parts: classification matching, clustering matching and mixed element matching. By each step, we keep narrowing down matching space and improving matching quality. We demonstrate the utility and efficiency of our algorithm with a set of comprehensive experiments on real datasets from the scenario of IoT industrial household appliances testing. The result shows that our algorithm has good performance.

Key words Internet of things(IoT); data interconnection; schema matching; heterogeneous data; time series

摘 要 随着物联网技术的快速发展,万物互联已经成为必然趋势.物联网技术涉及智能家居、智慧交通等多个领域,使得人们能够随时随地地连接任何人或者设备.然而目前大多数物联网异构数据是孤立的存储,阻碍了万物互联的步伐.数据模式匹配被广泛应用于数据互联并且能够较好地解决以上问题.由于海量物联网数据的异构性以及高增长性,目前的模式匹配方式人工参与度高,与实际应用的契合度低,难以解决物联网新环境下的模式自动匹配问题.提出了一种面向物联网异构数据的模式分层匹配算法,构建3层映射匹配:特征分类匹配、关系特征聚类匹配和混合元素匹配.该算法通过逐层匹配,不断缩小匹配空间,从而提高了匹配质量,减少了元素间匹配次数和人工参与度,较好地实现了自动模式匹配.运用大量数据样本来检验算法的效率和性能,结果证明该算法可行有效.

关键词 物联网;数据互联;模式匹配;异构数据;时间序列

中图法分类号 TP393

收稿日期:2017-09-12;**修回日期**:2018-01-08

基金项目:国家自然科学基金项目(61170258,61379127,61103196,61379128)
This work was supported by the National Natural Science Foundation of China (61170258, 61379127, 61103196, 61379128).

通信作者:郭忠文(guozhw@ouc.edu.cn)

据物联网行业统计数据显示,2014 年中国物联网产业规模达到 5 800 亿元,同比增长 18.46%.2015 年全球物联网市场规模达到 624 亿美元,同比增长 29%.到 2018 年全球物联网设备市场规模有望达到 1 036 亿美元,2019 年新增的物联网设备接入量将从 2015 年的 16.91 亿台增长到 30.54 亿台[1].随着物联网的快速发展以及物联网平台的逐步成熟,无处不在的末端设备和设施,如智能传感器、移动终端等通过物联网连接在一起,万物互联已经成为必然趋势.随之兴起的物联网技术被广泛应用到各个领域,产生了海量的异构数据.Mccrory[2]在 2010 年提出了"数据引力"的概念,将数据、软件应用、接口服务等比作星体,含有各自的质量和密度,数据质量不断地增加并且远大于其他"星体",而其他"星体"都将被巨大的引力吸引过来并以数据"星体"为中心.由此可见,对于物联网异构数据,其处理技术的发展将会直接影响整个物联网技术的发展和进步.目前大部分的异构数据都是独立、分散地存储于各个地区,形成了大量信息孤岛,由结构化数据(如关系型数据库)、半结构化数据(如 XML,HTML)和非结构化数据(如 NoSQL 数据库、图片、视频)等多种形式组成,如何通过"2 次采集"将这些信息孤岛互联起来受到人们的关注.

模式匹配技术在数据互联过程中发挥着巨大的作用[3],尽管目前对物联网异构数据可以通过制定统一数据接口标准的方式进行互联,但在实际应用中却很难实现,因为对于已经存在的海量数据存储模式的改造成本是巨大的,可以说数据异构的问题是不可避免的,数据模式匹配技术为解决以上问题提供了一个良好的解决方案.模式匹配是一个构建源模式和目标模式中元素间映射关系的过程,而传统的模式匹配操作大多都是由 IT 技术人员手工完成,随着数据规模的扩大以及模式复杂度的提升,手工匹配会耗费巨大的人力物力,并且容易破坏数据的完整性和准确性.目前已有的研究成果利用元素自身信息、语义信息、数据实例信息和结构信息来挖掘模式正确的元素映射关系.在之前的工作中,我们构建了一个面向物联网的异构数据转换模型,实现快速高效的物联网异构数据互联,然而在互联过程中模式匹配过程的人工参与度过高,匹配效率低下.为了解决该问题,我们首先分析了物联网异构数据的特征,即时间序列特征.并以此为切入点,通过解析时间序列数据来对物联网异构数据进行分类识别,为自动模式匹配提供了可能.基于以上研究成果,本文提出了一种面向物联网异构数据的模式分层匹配算法(hierarchical schema matching algorithm,HSMA),该算法基于数据模式的实例信息,通过 3 层映射匹配:特征分类匹配、关系特征聚类匹配和混合元素匹配,实现了物联网异构数据的模式自动匹配,减小了复杂模式下元素间匹配空间,提高了自动匹配的效率和准确度.

1　相关研究及问题分析

1.1　相关研究

目前比较典型且已经广泛应用的匹配算法有 Auto-match,Clio,COMA,iMAP,Cupid,LSD,GLUE 等.Cupid[4]利用属性名称、数据类型等模式信息计算属性之间的语义相似性,将语义相似度和结构相似度结合起来进行匹配.COMA[5]利用多种匹配器协同工作,通过过滤、筛选来提高匹配结果的准确率和效率.LSD[6]应用机器学习算法实现匹配并且自动整合匹配结果,可以很好地发现 1∶1和 1∶n 的匹配.基于数据实例信息的模式匹配的 iMAP[7]方法是一种综合利用模式中多种类型的信息同时得到模式间的简单和复杂映射关系的方法,可以很好地发现 1∶1和 1∶n 的匹配.

1.2　问题分析

文献[8]提出了一种基于实体分类的数据库模式匹配方法,根据数据实例信息的语义信息进行特征提取,运用朴素贝叶斯算法对特征进行分类,最后对子模式进行相应的模式匹配.由于物联网数据的海量性和异构性,无论数据库设计得多么规范化,但其属性名称和意义的定义因人而异,单一地以属性名等语义信息进行模糊分类的方式是不可取的,因为这种分类仅仅基于人为理解而非源数据的本质特征.文献[9]利用数据的概率分布统计特征,从另一个角度对数据特征进行提取,利用 BP 神经网络算法提高了模式匹配效率和准确度.然而该算法针对的仅仅是 1∶1 列之间的匹配,不适用于复杂模式匹配.随着数据规模不断增大,匹配空间和匹配次数也会急剧增加,导致匹配效率低下.与文献[8-9]的算法相比,本文提出的 HSMA 算法能够较好地解决在现实应用中自动获取解析数据源模式信息困难的问题,通过解析海量异构的原始数据,按照物联网数据的时间序列性特征将解析的特征进行分类,从而代表数据源的模式特征;其次,HSMA 算法中的关系特征聚类方法具有一定创新性,尤其对物联网

传感数据的特征聚类有良好的效果,消除了数据集数据间不同数据类型差异的影响;最后,HSMA 运用分层的方法逐步缩小匹配空间,减少匹配次数,提高了匹配的效率.

2 HSMA 算法

为了实现物联网异构数据(结构化数据)的模式匹配的智能化,我们设计了一种模式分层匹配算法 HSMA. 对于输入的一个未知源数据,我们首先根据时间序列特征和数据类型特征对其所有的数据集进行分类,获得子模式集合. 与此同时,通过之前的研究成果 IFRAT 算法,获取未知数据源领域特征信息,根据领域的不同初始化相应的领域标准集合以及标准化数据库进行关系特征聚类匹配,建立各类中源数据集合和目标数据库之间的映射关系(第 1 层匹配);然后依据所提取的数据集特征,运用机器学习算法,对每个子模式中的数据集进行聚类,进一步缩小匹配空间和范围,对聚类结果与相应的标准集合数据集进行相似度计算,建立匹配映射(第 2 层匹配);最后将相匹配的聚类中的元素混合并进行相似度计算(第 3 层匹配),产生源数据与目标数据库之间的模式匹配结果. 通过以上 3 层匹配,HSMA 逐步缩小匹配空间,降低了匹配次数,从而提高匹配效率. HSMA 算法的整体架构如图 1 所示:

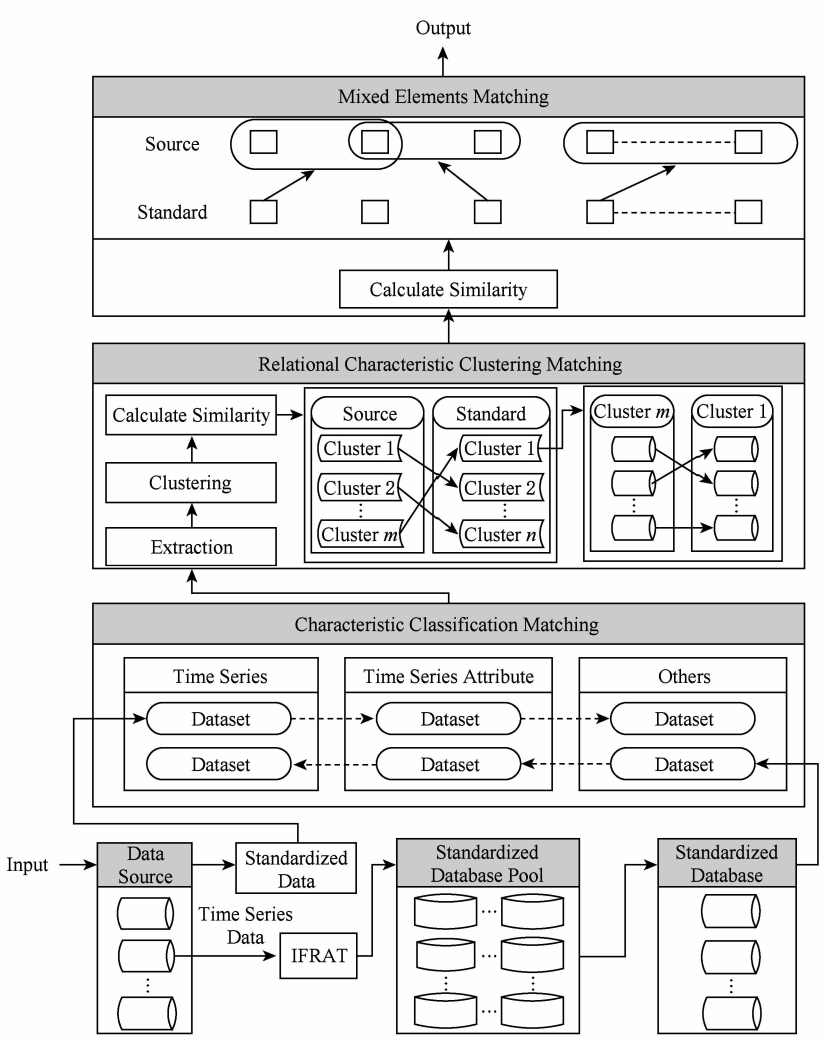

Fig. 1 Architecture of HSMA

图 1 HSMA 算法架构

2.1 源数据标准化

每个数据集中存在多种类型数据,以往的模式匹配算法都是对同一个数据集中不同类型元素进行单独处理,处理方法多样,虽然能够保留单一元素特征,但是却破坏了不同类型元素间的关联关系,失去了整个数据集的特征. 因此,针对数据集中常见的

数据格式(如数值、字符、日期等),需要寻找一个统一的数据处理方式以保证其特征的完整性.首先,我们将每个结构化数据集都看作是一个存在行列关系的2维矩阵,矩阵中的相邻行元素的差值变化结果作为新矩阵的行,我们将新的矩阵成为关系矩阵.差值变化结果用−,0,+表示,通过该方法,我们用关系值代替了原来的内容值,将不同类型的数据转化为同一标准,最终得到关系矩阵.关系矩阵作为数据集的一个标准化形式,能够更好地被用于数据集的特征挖掘.

为了更好地对HSMA算法进行理解,我们对本文中所用到的一些名词进行定义.

定义1. 关系矩阵.对于任意常见的结构化数据$\boldsymbol{D}_{m\times n}$,都可以通过相邻行做差的方法,差值结果用−,0,+这3种符号表示,我们将这样的矩阵定义为关系矩阵,用$\boldsymbol{M}$表示.

$$\boldsymbol{D}_{m\times n}=\begin{pmatrix} D_{11} & \cdots & D_{1n} \\ \vdots & & \vdots \\ D_{m1} & \cdots & D_{mn} \end{pmatrix}\rightarrow$$

$$\begin{pmatrix} D_{21}-D_{11} & \cdots & D_{2n}-D_{1n} \\ \vdots & & \vdots \\ D_{m1}-D_{(m-1)1} & \cdots & D_{mn}-D_{(m-1)n} \end{pmatrix}\rightarrow$$

$$\boldsymbol{M}=\begin{pmatrix} + & \cdots & - \\ \vdots & & \vdots \\ 0 & \cdots & - \end{pmatrix}.$$

定义2. 关系键对.关系矩阵$\boldsymbol{M}$的每一行中元素两两组合所形成的键对.

定义3. 特征列.对于关系矩阵$\boldsymbol{M}$的任意一列,当且仅当同时满足2个条件:1)该列元素的数值类型为日期类型或可以转化成日期类型的其他数据类型;2)该列中的元素“+”或元素“−”的出现频率大于阈值θ(我们取$\theta=90\%$),这样的列被称作时间序列特征列.如果仅满足条件2,则该列被称为主特征列.

定义4. 时间序列数据.物联网传感器产生的原始数据,按照时间顺序以结构化的形式存储,这样的数据集合被称为时间序列数据.

2.2　3层映射匹配

本节对HSMA的分层匹配算法进行了详细的描述,共分为3层:第1层为特征分类匹配(时间序列特征和数据类型特征);第2层为关系特征聚类匹配;第3层为混合元素匹配.基于分层的思想,我们将复杂的异构数据模式匹配过程分割成3步,逐步缩小其匹配空间.对于每一步的匹配,都是基于物联网异构数据特性,比较源数据与标准数据间的相似性,构建映射关系,最终实现物联网异构数据的模式自动匹配.

2.2.1　特征分类匹配

HSMA算法在特征分类匹配过程中提供2种分类方法:

1) 时间序列特征分类

时间序列是物联网异构数据的标志性特征,可以被用来进行物联网异构数据分类.首先我们将物联网异构数据的按时间序列特性分为3类:

① 时间序列数据类P.所有包含时间序列特征列的数据集属于时间序列数据类(如传感网传感数据).

② 时间序列属性类Q.包含主特征列的数据集归于时间序列数据属性类(如传感网监测对象元数据),如描述时间序列属性的相关信息.

③ 非时间序列类R.包含与时间序列无关的数据集,一般由许多相关辅助信息组成.

HSMA算法将物联网异构数据按照时间序列特征分类,最后得到数据分类集合{时间序列数据类,时间序列属性类,非时间序列类},其具体的分类算法如算法1所示:

算法1. 时间序列特征分类算法.

输入:物联网异构数据集合$X=\{x_1,x_2,\cdots,x_n\}$(结构化数据);

输出:时间序列特征类集合$\{P,Q,R\}$.

① 获取物联网异构数据集合(结构化数据),$X=\{x_1,x_2,\cdots,x_n\}$;

② 对X中所有数据集x_i进行数据标准化,得到与X相对应的关系矩阵集合$Y=\{\boldsymbol{M}_1,\boldsymbol{M}_2,\cdots,\boldsymbol{M}_n\}$;

③ $\forall x_i\in X$,遍历x_i的每一列,依据时间序列特征对x_i进行分类;

④ for each $x_i\in X$ do

⑤　for each col 属于$\boldsymbol{M}_i$的列集合 do

⑥　$n=Column\_num(\boldsymbol{M}_i)$;

⑦　$\theta\leftarrow\max\left\{\dfrac{Count(+)}{\sum_{i=1}^{n}i},\dfrac{Count(-)}{\sum_{i=1}^{n}i}\right\}$;

⑧　if $\theta\geqslant 90\%$ then

⑨　if $col\in Type.date$ then

⑩　$x_i\in P$;

⑪　else

⑫　$x_i\in Q$;

⑬ end if
⑭ else
⑮ if x_i 的所有列遍历结束 then
⑯ $x_i \in R$;
⑰ end if
⑱ end if
⑲ end for
⑳ end for
㉑ return $\{P,Q,R\}$.

2) 数据类型特征分类

物联网异构数据中,不同领域的源数据集合具有不同的数据类型分布特征,依据该特征对物联网异构数据分类.数据类型按照物联网异构数据的常见类型被分为 5 类:数值型(value)、字符类型(char)、时间类型(date)、字符-数值类型(char-value)、字符-时间类型(char-date),考虑在复杂模式匹配下,相同的数据可能被赋予不同的数据类型,数据类型间的相互转换会导致其分布特征偏离,因此将字符串-数值等可以相互间进行转换的情况列入数据类型特征.利用各个数据类型出现的频率值,构建一个 5 维的特征向量,然后依据特征向量对物联网异构数据进行特征分类:

① 时间主导类 E.由于时间类型的特殊性,只要包含时间类型的数据集都属于时间主导类.

② 数值主导类 F.数值类型(如 int,long,float,double 等)占比重大的数据集属于数值主导类.

③ 字符主导类 G.字符类型(如 char,varchar,nvarchar 等)占比重大的数据集属于字符导类.

HSMA 算法将物联网异构数据按照数据类型特征分类,最后得到数据分类集合$\{E,F,G\}$,其具体的分类过程如算法 2 所示:

算法 2. 数据类型特征分类算法.

输入:物联网异构数据集合 $X=\{x_1,x_2\cdots,x_n\}$(结构化数据);

输出:数据类型特征类集合$\{E,F,G\}$.

① 获取物联网异构数据集合(结构化数据),$X=\{x_1,x_2\cdots,x_n\}$;
② 对 X 中所有数据集 x_i 进行数据结构标准化;
③ $\forall x_i \in X$,遍历 x_i 的每一列,依据数据类型分布特征构建 x_i 的特征向量 $\boldsymbol{v}_i$,得到 X 相对应的特征矩阵 $\boldsymbol{V}=(\boldsymbol{v}_1,\boldsymbol{v}_2,\cdots,\boldsymbol{v}_n)$;
④ for each $x_i \in X$ do
⑤ for each $col \in x_i$ do
⑥ 统计过程①中各个数据类型发生次数;
⑦ end for
⑧ 构建数据类型特征向量 $\boldsymbol{v}_i$;
⑨ $n=Column\_num(x_i)$;

⑩ $$\boldsymbol{v}_i \leftarrow \left\{\frac{Count(\text{“value”})}{\sum_{i=1}^{n} i}, \frac{Count(\text{“char”})}{\sum_{i=1}^{n} i}, \frac{Count(\text{“date”})}{\sum_{i=1}^{n} i}, \frac{Count(\text{“char-value”})}{\sum_{i=1}^{n} i}, \frac{Count(\text{“char-date”})}{\sum_{i=1}^{n} i}\right\};$$

⑪ if $\dfrac{Count(\text{“date”})}{\sum_{i=1}^{n} i} \neq 0$ 或 $\dfrac{Count(\text{“char-date”})}{\sum_{i=1}^{n} i} \neq 0$
⑫ $x_i \in E$;
⑬ else
⑭ if $\left(\dfrac{Count(\text{“value”})}{\sum_{i=1}^{n} i}+\dfrac{Count(\text{“char-value”})}{\sum_{i=1}^{n} i}\right) > \left(\dfrac{Count(\text{“char”})}{\sum_{i=1}^{n} i}+\dfrac{Count(\text{“char-date”})}{\sum_{i=1}^{n} i}\right)$
⑮ $x_i \in F$;
⑯ else $x_i \in G$;
⑰ end if
⑱ end if
⑲ end for
⑳ return $\{E,F,G\}$.

2.2.2 关系特征聚类匹配

通过 2.2.1 节中的特征分类匹配,对源数据 S 和标准化数据库 T 中的数据分别进行时间序列特征分类和数据类型特征分类,得到的结果集合统称为 S_{cla} 和 T_{cla}.在本节,我们采用 SOM 聚类方法对 S_{cla} 和 T_{cla} 中的数据集进行归类,进一步缩小匹配空间.在聚类过程中,我们采用关系矩阵 $\boldsymbol{M}$ 作为数据集的特征矩阵,因为关系矩阵 $\boldsymbol{M}$ 具有以下特点:1) $\boldsymbol{M}$ 可以将数据集中不同类型的数据转换成统一的符号表示;2) $\boldsymbol{M}$ 可以保留元素间的相互关系,而传统的匹配方法忽略了这一点;3) $\boldsymbol{M}$ 解决了复杂模式匹配下元素排列顺序多变的问题,更好地表现数据集特征.关系矩阵 $\boldsymbol{M}$ 中,每一行中所有项两两组合

形成关系键对,HSMA 算法通过对数据集进行行遍历,提取数据集关系键对的频率分布特征,作为该数据集的特征向量 $\boldsymbol{v}_i$,基于得到的特征向量进行聚类匹配.详细过程如算法 3 所示:

算法 3. 关系特征聚类匹配算法.

输入:S_{cla},T_{cla};

输出:S_{cla}和 T_{cla}的聚类集合.

① 获取 S_{cla} 和 T_{cla} 中的分类集合:S_{cla} 或 $T_{\text{cla}}=\{Classification_1, Classification_2, Classification_3\}$;

② $dataset\in\{E,F,G\}$;

③ for each $Classification\in S_{\text{cla}}$ do

④　for each $ds\in dataset$ do

⑤　　$ds\rightarrow\boldsymbol{M}$;

⑥　　for each row 属于 $\boldsymbol{M}$ 的行集合 do

⑦　　关系键 $Key\leftarrow\{(-,-),(-,0),(-,+),(0,0),(0,+),(+,+)\}$;

⑧　　统计每种关系键 Key 出现的次数;

⑨　　end for

⑩　　$n\leftarrow$统计所有关系键 Key 出现的总次数;

⑪　　$$\boldsymbol{v}_i\leftarrow\left\{\frac{Count(-,-)}{\sum_{i=1}^{n}i},\frac{Count(-,0)}{\sum_{i=1}^{n}i},\frac{Count(-,+)}{\sum_{i=1}^{n}i},\frac{Count(0,0)}{\sum_{i=1}^{n}i},\frac{Count(0,+)}{\sum_{i=1}^{n}i},\frac{Count(+,+)}{\sum_{i=1}^{n}i}\right\};$$

⑫　end for

⑬ end for

⑭ 通过过程②～⑪,得到 S_{cla} 所有数据集特征向量矩阵 $\boldsymbol{V}_S$;

⑮ 对 T_{cla} 进行过程②～⑪,得到 T_{cla} 所有数据集的特征向量矩阵 $\boldsymbol{V}_T$;

⑯ SOM 对属于 $\boldsymbol{V}_S$ 中的特征向量进行聚类,计算其聚类中心(平均值法),得聚类中心集合 $C=\{\{\boldsymbol{c}_1,\boldsymbol{c}_2,\cdots,\boldsymbol{c}_p\},\{\boldsymbol{c}_{p+1},\boldsymbol{c}_{p+2},\cdots,\boldsymbol{c}_q\},\{\boldsymbol{c}_{q+1},\boldsymbol{c}_{q+2},\cdots\}\}$;

⑰ 对于 $\boldsymbol{V}_T$ 中任意一个特征向量 $\boldsymbol{v}_i$,用欧氏距离计算其与各聚类中心 C 的相似度;

⑱ for each $\boldsymbol{v}_i$ in $\boldsymbol{V}_T$ do

⑲　for each $\boldsymbol{c}_j\in C$ do

⑳　　$sim(\boldsymbol{v}_i,\boldsymbol{c}_j)$;

㉑　end for

㉒　将max:$sim(\boldsymbol{v}_i,\boldsymbol{c}_j)$中 $\boldsymbol{v}_i$ 对应数据集放入 $\boldsymbol{c}_j$ 对应的聚类中;

㉓ end for

㉔ return 聚类集合.

在本节,我们基于关系矩阵中提取的特征向量,对 S_{cla}中的所有数据集进行聚类,得到聚类集合$D=\{d_1,d_2,\cdots,d_l\}$,通过计算欧氏距离,确定 T_{cla}中的每一个数据集在 S_{cla}中最相似的聚类集合,从而进一步缩小匹配空间.我们假设并认定 S 中的每一个聚类对应 T 中唯一的数据集,即数据集之间只存在 $1:n$ 的关系.

2.2.3　混合元素匹配

基于 2.2.2 节得到的聚类集合 D. $\forall d_i\in D$,我们将 d_i 中的所有元素混合在一起,采用文献[10]的算法对于单一元素进行聚类匹配,该算法能够快速有效地发现聚类中心,能够在短时间内完成聚类过程且过程精简,最终得到匹配结果集合 $R=\{r_1,r_2,\cdots,r_l\}$.考虑到现实情况的复杂性,我们很难做到元素 1:1的精准匹配,所以 $\forall r_i\in R$,我们规定 r_i 中只包含 1 个标准化数据元素和 φ 个源数据的元素(这里的 φ 人为设定,本文选取 $\varphi=1,2,3$),这 φ 个源数据的元素作为最相似的元素被推荐用户,具体如算法 4 所示:

算法 4. 混合元素匹配算法

输入:聚类集合 $D\leftarrow\{d_1,d_2,\cdots,d_l\}$;

输出:匹配结果集合 $R\leftarrow\{r_1,r_2,\cdots,r_l\}$.

① 获取聚类集合:$D\leftarrow\{d_1,d_2,\cdots,d_l\}$;

② for each $d_i\in D$ do

③　运用文献[10]算法进行匹配;

④　设定 φ 的数值;

⑤　$r_i\leftarrow d_i$ 的匹配结果;

⑥ end for

⑦ return 匹配结果集合 $R\leftarrow\{r_1,r_2,\cdots,r_l\}$.

3　实验验证及结果分析

3.1　数据选取

为了证明 HSMA 的可行性和有效性,我们选取工业物联网家电产品测试领域的家用空调性能测试中 13 个不同厂家的 30 个数据库作为源数据,将基于国际标准 IEEE 1851 的空调测试数据库作为

标准化数据库(采用 Oracle),数据详细信息如表 1 所示:

Table 1 Performance Test Data of Residential Air Conditioner

表 1 家用空调性能测试源数据

No	DataSource	Number	Data Size/GB
1	Sqlserver2005	15	12
2	Txtfiles	8	10
3	Access	7	9.2

我们认为一般情况下存在 2 种常见的异构形式:行列转换和拆分.

1) 行列转换. 不是完全属于同一列的内容被列入同一列中,为了消除结构不同对匹配造成的影响,我们需要对这一类型的列进行逻辑行转换,同时记录相应映射关系,如图 2 所示:

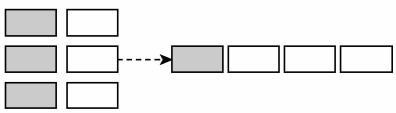

Fig. 2 Row-column transforming of IoT heterogeneous data

图 2 物联网异构数据行列转换

2) 拆分. 通常情况下列的合并要么是字符串由特殊字符隔开,要么布尔型直接合并,所以找到合并字段进行逻辑拆分,同时记录相应的映射关系,如图 3 所示:

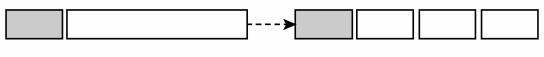

Fig. 3 Logic splitting of IoT heterogeneous data

图 3 物联网异构数据逻辑拆分

3.2 算法对比

我们采用不同的基于数据实例和数据库模式信息的模式匹配算法与 HSMA 算法进行对比,详细的对比信息如表 2 所示:

1) SMEC[8]. 通过朴素贝叶斯学习将实体分为不同的类,以同样的类来匹配子模式之间的模式元素.

2) SMDD[9]. 基于神经网络的模式匹配算法,通过分析模式元素所包含的数据实例的分布规律,自动完成模式匹配.

3) iMAP[7]. 一种综合利用模式中多种类型的信息同时得到模式间的简单和复杂映射关系的方法,在得到匹配关系的同时保存对每个匹配关系的判断过程,当用户对最终结果进行调整时,向用户提供保存的判断过程作为用户进行调整的依据.

Table 2 Comparison of Different Algorithms

表 2 不同算法之间比较

Algorithm	Feature	Classification	Clustering	Machine Learning
SMEC	Schema	Yes	No	Naïve Bayes
SMDD	Instance	Yes	No	BP-Neural Network
iMAP	Instance	Yes	No	Naïve Bayes
HSMA	Instance	Yes	Yes	SOM and Ref[10]

3.3 实测结果

为了评估 HSMA 的匹配质量,我们采用 3 个通用指标[11]进行评估:

1) 查准率(*Precision*). 匹配结果中正确匹配结果的比率.

$$Precision=\frac{T}{P}=\frac{T}{T+F}, \tag{1}$$

其中,T 为正确识别的匹配结果,P 为匹配方法返回的匹配结果,F 为错误的匹配.

2) 查全率(*Recall*). 匹配结果中正确匹配结果占实际匹配结果的比率.

$$Recall=\frac{T}{R}, \tag{2}$$

其中,R 为手工匹配结果.

3) *F*1_*measure*. 能够全面评价匹配质量的统计量.

$$F1\_measure=\frac{2\times Precision\times Recall}{Precision+Recall}. \tag{3}$$

3.3.1 HSMA 算法自测

1) HSMA 算法的匹配质量受到参数 φ 选取的影响. 通过对 φ 设定不同的值,分析比较其查准率、查全率和 *F*1-*measure*,结果如图 4 所示. 与 $\varphi=1$ 相比,当 $\varphi>1$ 时,查全率、查准率和全面性都有提高,因为 HSMA 算法无法保证精准的 1∶1匹配,模式的异构性导致存在多个相近的相似度结果. 但并不是 φ 值越大越好,φ 的增大会导致匹配干扰项增加,降低匹配质量,当 $\varphi=3$ 时,可以看出其匹配质量大于 $\varphi=1$ 但小于 $\varphi=2$.

2) HSMA 算法的匹配质量受到输入数据库数量的影响. 输入数据库的个数会直接影响关系特征聚类的效果,充足的数据会使数据特征更加明显. 如图 5 所示,对于不同的 φ 值,当输入数据库数量在(0,15)范围时,其匹配质量随着输入数据库个数的

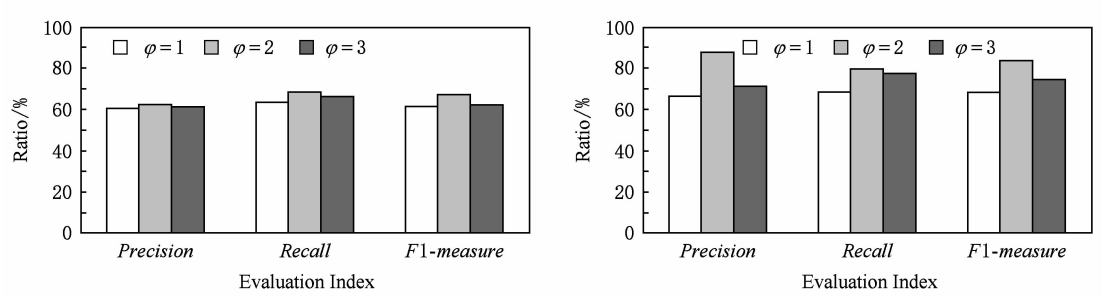

Fig. 4　Matching quality of HSMA under different φ

图 4　在不同的 φ 下 HSMA 的匹配质量

增加而不断提高且变化较为明显;但当输入数据库数量大于 15 后,匹配质量增长缓慢.

3) HSMA 算法的时间效率受参数 φ 选取的影响. 图 6 比较了在 φ 值的不同情况下源数据个数对 HSMA 算法时间效率的影响. 如图 6 所示,φ 越大,HSMA 算法所用的匹配时间就越大,因为 φ 值的增加导致在混合元素匹配过程中匹配次数的增多,提高了时间复杂度;当 φ 值固定时,采用时间序列特征分类的 HSMA 算法的匹配时间低于采用数据类型特征分类的 HSMA 算法.

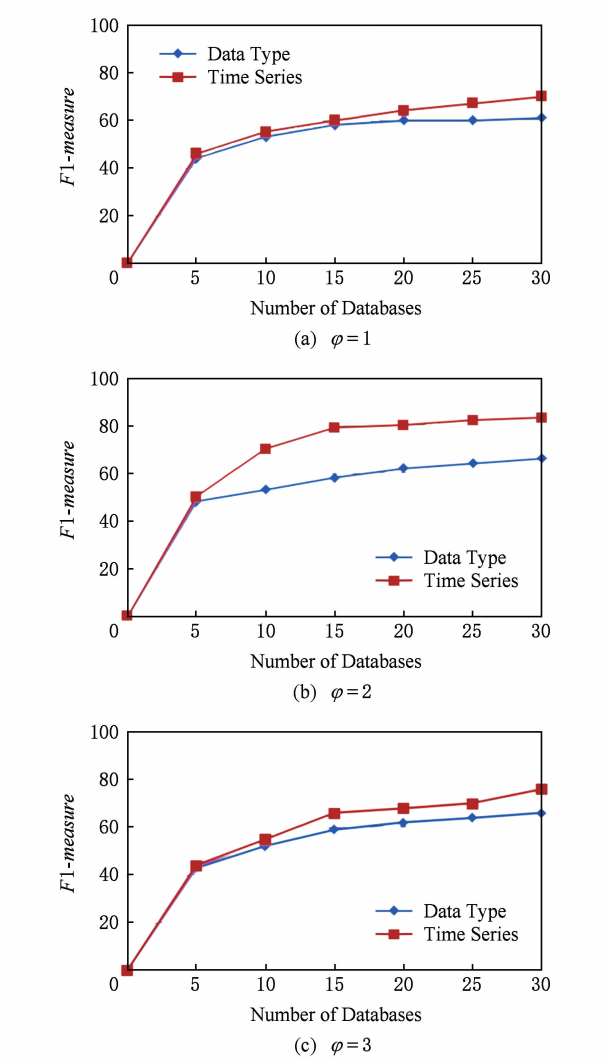

Fig. 5　Matching quality of HSMA with the increasing number of input databases

图 5　随着输入数据库数量的增加 HSMA 的匹配质量

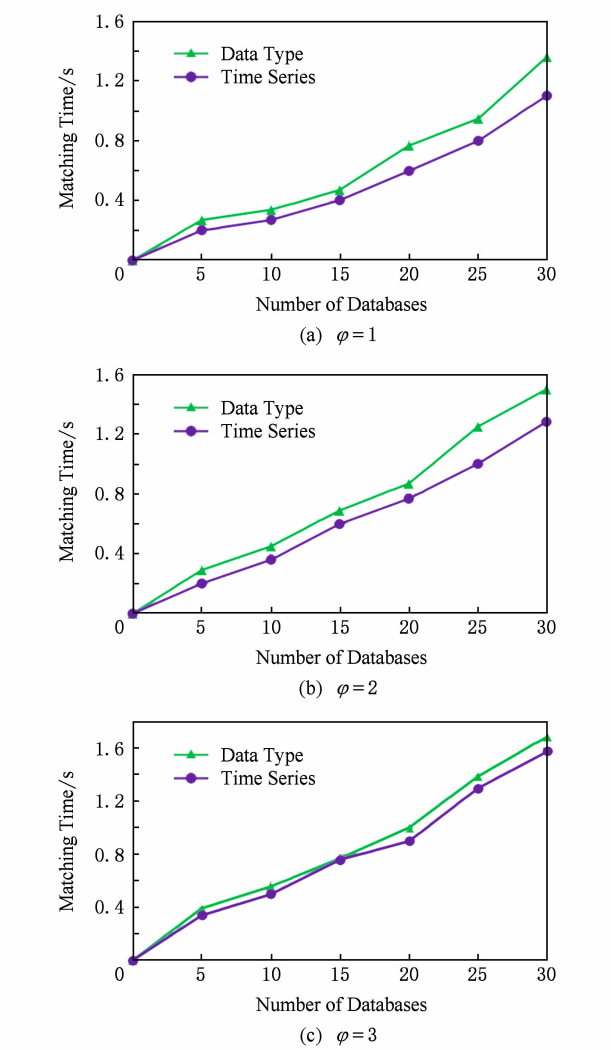

Fig. 6　Matching time of HSMA

图 6　HSMA 需要的匹配时间

4) HSMA 算法的时间效率受特征分类的影响. 我们随机选取 15 个数据源作为 HSMA 算法的输入,分别采用无分类、时间序列特征分类和数据类型特征分类并且设定不同的值,分析比较其每一层匹配所用时间,结果如图 7 所示. φ 的选取仅仅会影响混合元素匹配的匹配时间,φ 值越大混合元素匹配过程所用的时间就越多;当 φ 值固定时,采用时间序列分类的 HSMA 算法的匹配时间相对较小.

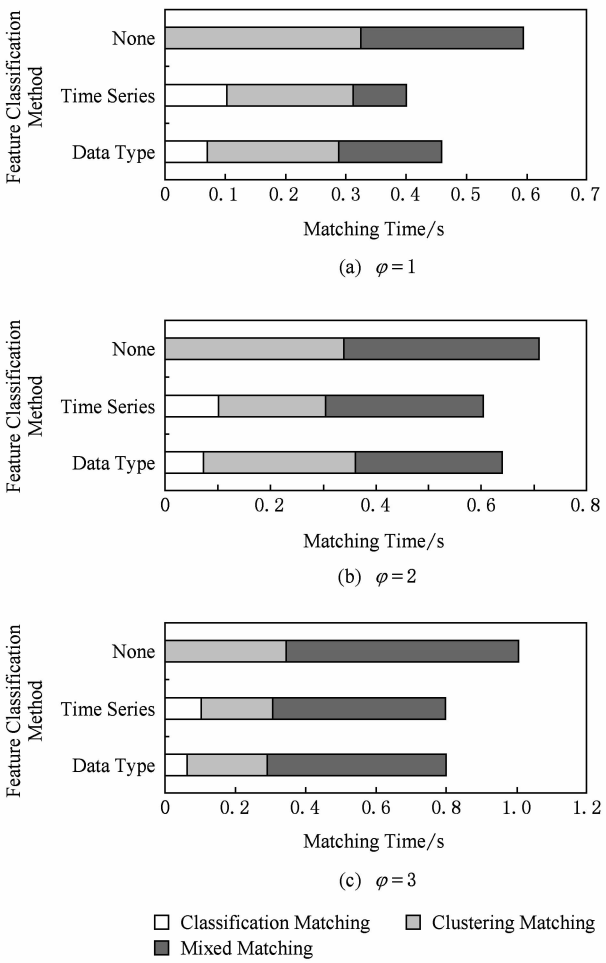

Fig. 7 Hierarchical matching time of HSMA

图 7 HSMA 的每一层匹配时间

3.3.2 算法性能对比

由 3.3.1 节可得,当选取 $\varphi=2$,采用时间序列特征分类的 HSMA 算法的性能最高,因此将其与 3.2 节中各个模式匹配算法进行比较. 我们选取了 30 个异构数据库作为 HSMA 的输入,比较所得结果如图 8 所示. 如图 8(a)所示,HSMA 算法的匹配质量与其他算法相比较高,其中源数据库的异构性和模式信息的不完整导致 SMEC 效率最差. 如图 8(b)所示,由于 HSMA 算法采用预处理以及多次聚类,导致其所用时间明显高于其他算法.

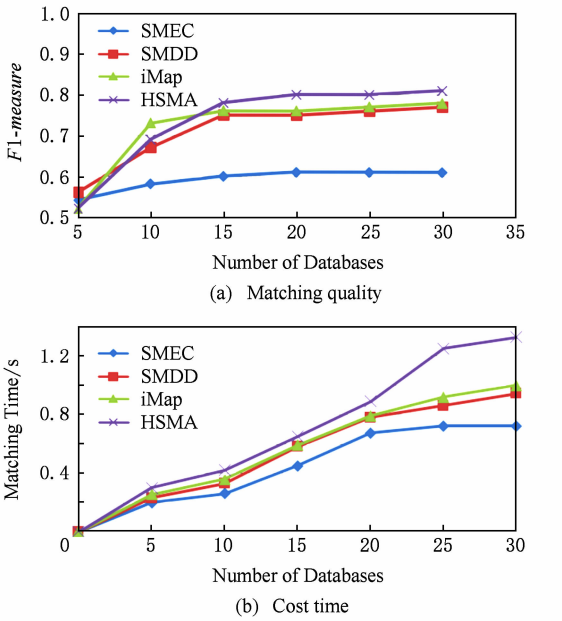

Fig. 8 Performance comparison of HSMA with other algorithms

图 8 HSMA 算法与其他算法的性能比较

4 总结和未来工作

我们设计了一种模式分层匹配算法 HSMA,对于输入的一个未知源数据,根据领域的不同初始化相应的领域标准集合以及标准化数据库进行特征分类匹配,建立各分类中源数据集合和标准数据库数据集合的映射关系;然后提取源数据集和目标数据库表中的关系特征,运用 SOM 聚类算法,对每一个子模式中的数据集进行聚类,对聚类结果与相应的标准数据库数据集进行相似度计算,建立匹配映射;最后将匹配结果集合中的混合元素进行相似度计算,进行单一元素匹配. 通过以上 3 层匹配,HSMA 算法逐步缩小匹配空间,降低了匹配次数,从而提高匹配质量和效率.

由于在关系特征聚类匹配及混合元素匹配的过程中,聚类算法的好坏直接决定了最后的匹配质量和总体的匹配时间,聚类虽然能够缩小匹配空间,但同时有可能带来匹配错误,导致相关元素并不在同一个类中. 未来会尝试使用不同的机器学习算法对 HSMA 算法进行改进,并且寻找最好的组合模式来提高聚类的效果. 此外本文在实验验证环节采用 iMAP,SMDD 等经典算法与 HSMA 算法进行对比,提高了对比结果的可靠性,但是这些算法偏于老旧,下一步将采用近几年的新算法进行比较,以验证 HSMA 算法的先进性.

参　考　文　献

[1] Chinabgao. Statistic data of Chinese IoT market [OL]. (2017-03-18) [2017-04-06]. http://www.chinabgao.com/k/wulianwang/26922.html

[2] Mccrory D. Data gravity [OL]. (2010-12-07) [2010-12-07]. http://blog.mccrory.me/2010/12/07/data-gravity-in-the-clouds

[3] Edhy S, Retantyo W, Khabib M, et al. Survey: Models and prototypes of schema matching [J]. International Journal of Electrical and Computer Engineering, 2016, 6(3): 1011-1022

[4] Madhavan J, Bernstein P A, Rahm E. Generic schema matching with Cupid [C] //Proc of the 27th VLDB Conf. San Francisco: Morgan Kaufmann, 2001: 49-58

[5] Do H. COMA: A system for flexible combination of schema matching approach [C] //Proc of the 28th VLDB Conf. San Francisco: Morgan Kaufmann, 2002: 610-621

[6] Doan A H. Reconciling schemas of disparate data sources: A machine-learning approach [C] //Proc of the 1st ACM SIGMOD Int Conf on Management of Data. New York: ACM, 2001: 509-520

[7] Dhamankar R. iMAP: Discovering complex semantic matches between database schemas [C] // Proc of the 4th ACM SIGMOD Int Conf on Management of Data. New York: ACM, 2004: 383-394

[8] Yu Bo, Tang Shiwei, Zhang Peng. An approach to database schema matching based on entity classification [J]. Journal of Computer Scinece, 2004, 31(10): 157-159 (in Chinese)
(于波,唐世渭,张鹏. 基于实体分类的数据库模式匹配方法[J]. 计算机科学, 2004, 31(10): 157-159)

[9] Li You, Liu Dongbo, Zhang Weiming. A method for automatic schema matching using characteristic of data distribution [J]. Journal of Computer Science, 2005, 32(11): 85-87 (in Chinese)
(李由,刘东波,张维明. 基于数据实例分布特征的自动模式匹配方法[J]. 计算机科学, 2005, 32(11): 85-87)

[10] Rodriguez A, Laio A. Clustering by fast search and find of density peaks [J]. Science, 2014, 334(6191): 1492

[11] Cyril G, Eric G. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation [J]. International Journal of Radiation Biology, 2005, 51(5): 199-239

Guo Shuai, born in 1989. PhD candidate. His main research interests include sensor networks, Internet of things, machine learning and so on.

Guo Zhongwen, born in 1965. PhD, professor and PhD supervisor. His main research interests include sensor networks, distributed measurement systems, ocean monitoring and so on.

Qiu Zhijin, born in 1987. PhD candidate. His main research interests include sensor networks, distributed measurement systems, machine learning and so on.