Hans Journal of Data Mining 数据挖掘, 2021, 11(2), 59-76
Published Online April 2021 in Hans. http://www.hanspub.org/journal/hjdm
https://doi.org/10.12677/hjdm.2021.112007

Hans汉斯

基于映射关系的领域词典抽取算法

崔　晨，李　贵，李征宇，韩子扬，曹科研

沈阳建筑大学，信息与控制工程学院，辽宁 沈阳
Email: 724079015@qq.com, ligui21c@sina.com

收稿日期：2021年3月1日；录用日期：2021年4月1日；发布日期：2021年4月8日

摘　要

领域词典是一种领域知识的表现形式，是数据规范化和数据清洗的重要参考信息。映射关系指表格中某两列间的对应关系。领域词典构建与扩充以Web表格为主要数据来源，需要对众多Web表格中的局部映射关系进行联结和扩展，但Web表格中存在异构和数据质量问题，不能单纯地依靠模式匹配等数据集成技术。本文提出了一种基于映射关系的领域词典抽取算法。首先利用带IDF权重的Jaccard最大包含度和编辑距离进行近似字符串匹配，并利用高斯混合模型实现数值离散化，从而解决了数据层面的异构性问题。然后由点互信息和函数依赖确定包含映射关系的候选表；接下来定义了候选表间的相容性和相斥性，构造出映射关系图模型，以进行候选表联结，实现了以映射关系为形式的领域词典抽取。最后，为保证领域词典的质量，加入了冲突消解过程。在实验验证阶段，本文利用房地产领域数据集，与其他从Web获取领域知识的算法进行比较，验证了本文所提出算法的有效性和可靠性。

关键词

领域词典，映射关系，近似匹配，离散化

A Domain Dictionary Extraction Algorithm Based on Mapping Relationships

Chen Cui, Gui Li, Zhengyu Li, Ziyang Han, Keyan Cao

School of Information & Control Engineering, Shenyang Jianzhu University, Shenyang Liaoning
Email: 724079015@qq.com, ligui21c@sina.com

Received: Mar. 1st, 2021; accepted: Apr. 1st, 2021; published: Apr. 8th, 2021

Abstract

The domain dictionary is a form of expression of domain knowledge and the important reference

文章引用: 崔晨, 李贵, 李征宇, 韩子扬, 曹科研. 基于映射关系的领域词典抽取算法[J]. 数据挖掘, 2021, 11(2): 59-76. DOI: 10.12677/hjdm.2021.112007

information for data normalization and data cleaning. The mapping relationships refer to the corresponding relationship between two columns in a table. The construction and expansion of domain dictionary takes Web tables as the main data source, and it is necessary to connect and expand the local mapping relationships in many Web tables. However, there are heterogeneous and data quality problems in Web tables, data integration technologies, for example, pattern matching cannot be relied on. This paper proposes a domain dictionary extraction algorithm based on mapping relations. Firstly, we use the IDF-Jaccard maximum containment and edit distance for approximate string matching, and use Gaussian mixture model to achieve numerical discretization, thereby solving the heterogeneity problem at the data level. Next, the candidate table containing mapping relationships is determined by the pointwise mutual information and functional dependence; then the compatibility and repulsion between the candidate tables are defined, and the mapping relationship graph model is constructed to connect the candidate tables, and the domain dictionary with the form of mapping relationships is extracted. Finally, to ensure the quality of the domain dictionary, a conflict resolution process was added. In the experimental verification, this paper used real estate data sets, compared with other algorithms that obtain domain knowledge from the Web, so the effectiveness and reliability of the algorithm proposed was verified.

Keywords

Domain Dictionary, Mapping Relationship, Approximate Match, Discretization

1. 引言

映射关系是指数据集当中的两个或多个属性间的对应关系。作为一种语义范畴的概念，映射关系与函数依赖等数据依赖相类似，可作为一种约束条件，来检测和修复数据库中的拼写错误、空值和不一致值。与函数依赖相比，映射关系的严格性较低，容许近似映射，而且映射关系能够跨越多个关系表(关系模式)，可实现更复杂的表达组合。

领域词典，是用来存储领域相关的领域词及它们之间关系的领域内专业词典，其基本组成单位是领域词对，不仅可表示为(属性，值)对，还可表示为具有隐含关系的(实体，实体)对。在领域词典中，符合同一关系的领域词对被联结成一张子表，在同一张子表中的关系是一致的，关系可作为隐含条件而省略掉。因而，领域词典与知识图谱中的垂直划分存储方案[1]相类似。垂直划分存储方案为知识图谱中的每种关系建立一张两列的表(主语，宾语)，表中存放由谓语(关系)连接的主语和宾语[2] [3]。

传统上，领域词典来源于专家或其他额外信息，这些来源有很大的局限性，如：以领域专家为来源，需要大量的人工配合，难以使知识详尽化，且易受主观因素影响。而以知识库为来源，又无法做到与实际数据的特异化对应。随着爬虫等自动化网络信息采集技术的发展，网页上的表格日渐成为一大重要的数据源，这些结构化数据中蕴含大量有用信息，在构建或完善知识库、扩充现有数据库等场景中被大范围应用。因此，本文中的领域词典以大量 Web 表格为数据来源。

尽管不同来源的 Web 表格中可能存在着相一致的映射关系，但在大多数情况下，不同来源的 Web 表格中的值是难以比较的。这种不同数据源中属性取值的差异性称为数据层面的异构性，其包括标度差异性和相同实体的多重表示两大类。标度差异性指的是在不同来源的 Web 表格中存在一些数值转换，比较典型的情况是：对于同一范畴的有序变量，一些 Web 表格中用连续的数值来表示，而另一部分 Web

表格中则用字符串表示成有序等级的形式。如有些 Web 表用连续的数值量来表示房屋的面积大小，而另一些 Web 表格用字符或字符串型的分类等级变量表示，这使得难以挖掘它们之间的关系。相同实体的多重表示，也就是实体命名中的同义词替换问题。指的是由于 Web 表格的来源网站不同，异构 Web 的表格间对同一实体的表述形式不尽相同。这给数据的比较和匹配带来了困难。另一方面由爬虫程序从 Web 获取的表格语料库中往往存在数据质量问题。这两大原因导致了某些键值对不符合函数依赖。映射关系允许部分关系不满足函数依赖约束，同时还允许键值对间的近似匹配，其广泛适用性得以体现。

本文所用的 Web 表格语料库就是由很多类似于表 1~表 5 的 Web 表格构成的。它们的来源不同，表达了不同的主题，但其中某些部分反映了相同的映射关系。实际中，Web 表格语料库是由上千个这样的表格组成的，每个 Web 表格中可能只反映了某个映射关系的局部。领域词典的抽取需要将符合同一映射关系的子表连结起来，面临如下困难：1) 不同来源的异构 Web 表格对同一实体的命名方式不尽相同，如表 1、表 4 和表 5 中的学校名称；2) 取值类似的映射关系间存在干扰，如学区与周边学校、街道－社区与区域－板块等。这给领域专家和数据分析师带来了繁重的负担。

本文将提取和联结 Web 表格中的映射关系。采用近似字符串匹配算法来解决相同实体的多重表示问题和数据质量问题；在解决标度差异性问题时，本文将借助高斯混合模型对连续的数值量进行离散化。在解决了不同来源的 Web 表格间数据层面的异构性的基础上，本文将利用点互信息和函数依赖过滤来获取映射关系。并在映射关系图模型中同时考虑映射关系间的相容性和相斥性，进行映射关系联结，最后进行冲突消解。从而在保证质量的前提下，提高领域词典的完备性。

Table 1. A table of school district allocation
表 1. 学区分配表

小区名称	街道	社区	小学学区	初中学区
世纪新城	五三	世纪新城	浑南区第二小学	辽宁省实验中学浑南一中
三隆世纪新城	迎宾路	沈新路	应昌大明湖分校	沈阳市杏坛中学
……	……	……	……	……

Table 2. A table of real estate information
表 2. 房地产信息表

楼盘名称	楼栋地址	建筑形式	层数	上市时间
韩国新城	沈阳市皇姑区长江南街 1 号	高层	28	2005/9/28
世纪新城	浑南新区沈营路 17 号	多层	6	2006/9/25
西江俪园	皇姑区淮河街 45 号	小高层	9	2005/9/29
……	……	……	……	……

Table 3. A table of house price
表 3. 房屋售价表

小区	面积	户型	单价	楼栋地址
工人新村	45.75	1 室 1 厅	5700	铁西区南十一西路 15 号
世纪新城	79.45	2 室 2 厅	5200	浑南新区沈营路 17 号
汇宝国际花园	149.2	3 室 2 厅	6900	皇姑区华山路 109 号
……	……	……	……	……

Table 4. A table of block information
表 4. 房屋区块信息表

名称	区域	板块	公交线路	学区 A	学区 B
韩国新城	长江	北行	209、210、215	昆山三校	126 中学
世纪新城	五三	浑河堡	612、333、4355	浑南区第二小学	辽宁省实验中学浑南一中
三隆世纪新城	迎宾路	北李官	240、604、111	应昌大明湖分校	沈阳市杏坛中学
……	……	……	……	……	……

Table 5. A table of supporting facility
表 5. 房屋周边配套设施表

项目名称	地址	结构形态	周边小学	周边中学	……
韩国新城	皇姑区三洞桥地区长江街东	高层	昆山三校	沈阳市第 126 中学	……
汇宝国际花园	皇姑区华山路 109 号	高层	珠江五校	43 中学	……
……	……	……	……	……	……

本文主要贡献如下：1) 在原有的 Jaccard 最大包含度的基础上，加入了逆文档频率 IDF (Inverse Document Frequency)权重，实现了映射关系间各实体对、属性－值对的近似匹配。2) 引入了一种与字符串长度(或字段提供的信息量、标签)相关的相对编辑距离概念，更适合于在汉语语境下实现高效的近似字符串匹配。3) 改进了一种基于高斯混合模型的离散化方法。4) 依据属性对间的集合相似性和函数依赖约束，在原有的仅考虑关联强度的 Web 表合成算法中加入了相斥性约束，避免了将有值重叠的不同映射关系错误地合并在一起，提高了映射关系联结算法的可靠性。

论文的其余章节组织如下：本文在第 2 节中介绍了现阶段关于领域词典抽取的一些相关工作；第 3、4 节提出了用于解决数据层面异构性的算法，其中第 3 节提出了一种适应于本文 Web 表格语境的字符串近似匹配算法，第 4 节给出了用于连续值离散化的高斯混合模型；第 5 节提出用于提高领域词典完备性的映射关系联结算法；第 6 节对算法进行了实验验证；第 7 节做了总结。

2. 相关工作

文献[2] [3]介绍了列存储数据库的思想和知识图谱的垂直划分存储方案，这与本文的领域词典相似，且适用于当今以 Hadoop 为基础的大数据计算体系。文献[4]介绍了领域词典的概念，并提出了一种自划分模型来解决覆盖度不足问题。文献[5]提出了一种基于词共现和词上下文的领域观点词抽取方法，对基于点互信息的抽取方法加以改进。文献[6]介绍了一种通过表格单元的值来挖掘列标签和列对间关系的算法。文献[7]介绍了用于挖掘形如(X, Y)的 Hearst 模式数据库的算法，从而构建 Web 知识库。文献[8]介绍了挖掘实体间关系(以二元关系为主)的开放信息抽取算法，该算法在抽取前不需要人为指定关系或关系集。文献[9]指出 Web 表数据集成与知识抽取中的主要挑战是 Web 表中明确模式信息缺失、高度异构性和碎片化(小尺寸且不完备)。传统的基于模式匹配的方法，不能很好地处理碎片化严重的 Web 表。文献[10]结合了基于模式和基于实例的匹配技术，提高了表格合并的质量，并将其用于 Web 开放数据的集成与知识挖掘。处理不同来源的异构 Web 表难度更大。文献[11]针对没有属性名称或属性名称不透明的异构表格研究了模式匹配。文献[12]从集合的重叠度、语义学、自然语言学角度讨论了开放 Web 表格的可联结性，并用于领域信息发现。

3. 短文本近似字符串匹配算法

如引言中所述，如将 Web 表格语料库作为领域词典抽取的数据源，首先需解决的问题是数据层面的异构性。本节将解决相同实体的多重表示问题和 Web 表格语料库中的部分数据质量问题。首先给出两种字符串相似度，其一是带有逆文档词频权重的对称化 Jaccard 包含相似度，其二是基于编辑距离的相似度。接下来提出短文本近似字符串匹配算法。从而使本文中的领域知识抽取算法能适应于不同来源的、可能存在拼写错误或表格提取问题的 Web 表格，避免一些重要数据被后文中的映射关系挖掘算法过滤掉。

3.1. 带 IDF 权重的 Jaccard 最大包含度

首先，将字符串看作由标签组成的集合，字符串中的每一个语义单元看作一个标签，例如：将“广东省深圳市光明新区马田街道新庄社区”转化为{广东省，深圳市，光明新区，马田街道，新庄社区}。标签化不仅可以令多个表示相同意义的词指向同一个标签，解决同义词转换的问题；还可以解决一词多义问题，在处理诸如“吉林(省)吉林(市)”、不同城市的“铁西区”等元素时，尽管它们的字符串表示形式相同，但标签化足以避免混淆。

在解决上述两大问题的同时，将字符串转化为标签集后，可以用集合相似度来反映字符串间的相似性。比较典型的集合相似度衡量方式是 Jaccard 相似系数(Jaccard Similarity Coefficient)，是通过样本间交集与总集之比衡量相似度的算法，用于字符串匹配时，可表示为 $\frac{|\mathrm{Str}_1 \cap \mathrm{Str}_2|}{|\mathrm{Str}_1 \cup \mathrm{Str}_2|}$。在字符串 Str_1 与字符串 Str_2 间，两者的交集越大，表示两字符串的相似度越高。Jaccard 相似系数与元素(字符)在集合(字符串)中的顺序无关，仅与元素在集合中是否出现有关。但如果一个字符串完全包含于另一个长字符串时 $\left(\mathrm{Str}_1 \supset \mathrm{Str}_2, |\mathrm{Str}_1| \gg |\mathrm{Str}_2|\right)$，其相似度很高，而实际计算出的 Jaccard 相似系数却很低。

Jaccard 包含度(Jaccard Containment) [13]，表示为 JCont，其公式为

$$\mathrm{JCont}\left(\mathrm{Str}_1, \mathrm{Str}_2\right) = \frac{|\mathrm{Str}_1 \cap \mathrm{Str}_2|}{|\mathrm{Str}_1|} \tag{1}$$

公式(1)表示 Str_2 关于 Str_1 的 Jaccard 包含度，若 $\mathrm{Str}_1 \supset \mathrm{Str}_2, |\mathrm{Str}_1| \gg |\mathrm{Str}_2|$，$\mathrm{JCont}\left(\mathrm{Str}_1, \mathrm{Str}_2\right) = \frac{|\mathrm{Str}_2|}{|\mathrm{Str}_1|} \approx 0$，而 $\mathrm{JCont}\left(\mathrm{Str}_2, \mathrm{Str}_1\right) = \frac{|\mathrm{Str}_2|}{|\mathrm{Str}_2|} = 1$，这在一定程度上解决了上述问题。大多数情况下 $\frac{|\mathrm{Str}_1 \cap \mathrm{Str}_2|}{|\mathrm{Str}_1|} \neq \frac{|\mathrm{Str}_1 \cap \mathrm{Str}_2|}{|\mathrm{Str}_2|}$，称 Jaccard 包含度不对称。

在本文研究的问题中，来自不同 Web 表格中的数据往往将同一元素的简写或别名掺杂在一起，同时字符串也有可能会缺失部分字段，但缺失的部分有可能对其表意并没有影响，公式(1)会过度惩罚这样的字符串。当然，如果多次迭代运行本文的领域词典抽取算法，利用新发现的领域词典来替换同义词，缺失的部分会得到逐步的填充，字符串会逐步完备化，但这么做势必会大幅提高计算量。如果在计算时，令每个标签的权重随其在语料库中出现的频率逆序改变(即逆文档频率 IDF，inverse document frequency)，即可解决此问题[14]。通常，IDF 作为 TF-IDF (Term Frequency-Inverse Document Frequency，词频-逆文档频率)的一个分量，表示记录中某个词的重要性与它在其他记录中出现的次数成反比，IDF 削弱了语料库中较高频度词语的影响，带来了较好的内容区分能力。

带有 IDF 权重的 Jaccard 包含度公式如下：

$$\mathrm{JCont}_{\mathrm{IDF}}\left(\mathrm{Str}_1, \mathrm{Str}_2\right) = \frac{\mathrm{weight}_{\mathrm{IDF}}\left(\mathrm{Str}_1 \cap \mathrm{Str}_2\right)}{\mathrm{weight}_{\mathrm{IDF}}\left(\mathrm{Str}_1\right)} \tag{2}$$

与公式(1)相比，这里用 IDF 权重和代替了集合中元素个数，设化为标签集后的字符串为 $\mathrm{Str}=\{a_1,a_2,a_3,\cdots,a_n\}$，$\mathrm{weight}_{\mathrm{IDF}}(\mathrm{Str})=\sum_1^n \mathrm{IDF}(a_n)$，其中 $\mathrm{IDF}(a_n)=\log\frac{|A_{\mathrm{all}}|}{|\{A_m:a_n\in A_m\}|+1}$，$|A_{\mathrm{all}}|$ 指表格语料库中字符串的总数，$|\{A_m:a_n\in A_m\}|$ 指表格语料库中包含标签 a_n 的字符串的个数。

显然，同 Jaccard 包含度一样，IDF 加权包含度也是不对称的。在 Web 表格语料库中，由于初始时可能缺乏规范的完备数据，更需要一种对称化的方式去衡量两个字符串间的相似性。由最大包含度可得：

$$\mathrm{MJCont}_{\mathrm{IDF}}(\mathrm{Str}_1,\mathrm{Str}_2)=\max\left\{\frac{\mathrm{weight}_{\mathrm{IDF}}(\mathrm{Str}_1\cap\mathrm{Str}_2)}{\mathrm{weight}_{\mathrm{IDF}}(\mathrm{Str}_1)},\frac{\mathrm{weight}_{\mathrm{IDF}}(\mathrm{Str}_1\cap\mathrm{Str}_2)}{\mathrm{weight}_{\mathrm{IDF}}(\mathrm{Str}_2)}\right\} \tag{3}$$

其中，Str_1 和 Str_2 是待匹配字符串，$\mathrm{MJCont}_{\mathrm{IDF}}(\mathrm{Str}_1,\mathrm{Str}_2)$ 为 Str_1 和 Str_2 间的相似性。当两个字符串的相似度不小于阈值 θ_{String} 时，即 $\mathrm{MJCont}_{\mathrm{IDF}}(\mathrm{Str}_1,\mathrm{Str}_2)\geq\theta_{\mathrm{String}}$，可认为这两个字符串指向同一实体。

并且，如果这两个字符串Str_1和Str_2间的相似度很高却都不完备，可对它们取并集来获得更为完备和规范的表述 $\mathrm{Str}_{\mathrm{complete}}$，即 $\mathrm{Str}_{\mathrm{complete}}=\mathrm{Str}_1\bigcup\mathrm{Str}_2$。

例 1：如有两个字符串，“公明区一小”、“公明第一小学”，“光明实验初中”、“光明区实验学校”，由于“区”、“第”、“学”、“学校”等词在 Web 表格语料库中出现的频率较高，其 IDF 权重相应较低，若上述每组中两个字符串间的相似度高于阈值，可归并为完整字符串“公明区第一小学”、“光明区实验初中学校”，这在一定程度上实现了缩写的补全。而更具实际意义的是，上述字符串相似度与字符串完备化方法适用于匹配表示地址的字符串，如遇到“罗湖区海山街道田东社区梧桐路 1968 号”和“广东省深圳市罗湖区田东社区梧桐路 1968 号”时，可判定这两个字符串表示相同的实体，并可补全为“广东省深圳市罗湖区海山街道田东社区梧桐路 1968 号”。

3.2. 相对编辑距离阈值

在这里用相对编辑距离阈值来代替传统的固定编辑距离阈值，这是因为固定的编辑距离阈值难以兼顾长、短字符串。由表 1~表 5 中可发现，Web 表格中的某些字段较短，仅有 2 个字，匹配时要求编辑距离阈值为 0，而有的字段较长，多达 10 余个字符，约束可适当放宽。

设待匹配的分类值 Str_1 和 Str_2 间的编辑距离为 $d_{\mathrm{ed}}(\mathrm{Str}_1,\mathrm{Str}_2)$。当编辑距离 $d_{\mathrm{ed}}(\mathrm{Str}_1,\mathrm{Str}_2)$ 小于阈值 θ_{ed} 时，即可将 Str_1 和 Str_2 视为匹配。相对编辑距离阈值可定义为

$$\theta_{\mathrm{ed}}=\min\left\{\left\lfloor|\mathrm{Str}_1|\cdot f_{\mathrm{ed}}\right\rfloor,\left\lfloor|\mathrm{Str}_2|\cdot f_{\mathrm{ed}}\right\rfloor,k_{\mathrm{ed}}\right\} \tag{4}$$

公式(4)中的相对编辑距离阈值 θ_{ed} 是根据字符串的长度 $|\mathrm{Str}_1|$、$|\mathrm{Str}_2|$ 和分度值 f_{ed} 动态确定的。选择分段表示的相对编辑距离，使所需的编辑距离随待匹配值的长度变化，其原因是在处理汉语时，短字符串必须做到完全精确匹配。更进一步，将阈值限定在固定值 k_{ed} 以内，以避免误匹配。在本文的实验中，$f_{\mathrm{ed}}=0.2$，$k_{\mathrm{ed}}=10$。

例 2：对于“东乐花园”、“东安花园”两个字符串，如果按单字拆分，其编辑距离为 1，而相对编辑距离阈值为 $\min\{\lfloor 4*0.2\rfloor,\lfloor 4*0.2\rfloor,10\}=0$，避免了误匹配。而进一步地，如果将“花园”整体视为一个标签，则相对编辑距离阈值为 $\min\{\lfloor 3*0.2\rfloor,\lfloor 3*0.2\rfloor,10\}=0$，避免误匹配的效果会更好。

3.3. 近似字符串匹配

后续章节中的算法是以近似字符串匹配算法为基础的，如果两列中元素个数分别为 m、n，需要进行 $O(mn)$ 级相似字符串匹配计算。设待匹配的字符串中字符数量分别为 $|\mathrm{Str}_1|$、$|\mathrm{Str}_2|$，在基于动态规划矩阵的编辑距离算法中，每次匹配的时间杂度和空间复杂度均为 $O(|\mathrm{Str}_1||\mathrm{Str}_2|)$。这需要消耗大量的时间和空

间。为提高效率，本文做如下优化：由于编辑距离阈值 θ_{ed} 普遍较小，根据 Ukkonen 算法[15]，在动态规划矩阵中，仅需计算对角线附近的较小范围内的值，这使得每次匹配的时间复杂度减小为 $O\left(\theta_{ed} * \min\left\{\left|\mathrm{Str}_1\right|,\left|\mathrm{Str}_2\right|\right\}\right)$。

由 3.1 节和 3.2 节，基于 IDF-Jaccard 最大包含度和相对编辑距离阈值，可得出近似字符串匹配算法，如算法 1 所示。

算法 1： 近似字符串匹配

输入： 字符串 Str_1 和 Str_2，相似度下限 θ_{String}，编辑距离上限 θ_{ed}

输出： 表示是否匹配的布尔值

1 $\mathrm{Matched} \Leftarrow \left(\mathrm{MJCont}_{\mathrm{IDF}}\left(\mathrm{Str}_1,\mathrm{Str}_2\right) \geq \theta_{\mathrm{String}}\right)$

2 **break**

3 **if** $\left|\mathrm{Str}_1\right| > \left|\mathrm{Str}_2\right|$ **then**

4 swap(Str_1 , Str_2)

5 $\mathrm{dist}_{|\mathrm{Str}_1|,|\mathrm{Str}_2|} \Leftarrow \infty$

6 $\mathrm{dist}_{i,0} \Leftarrow i, \forall\left(i \in \left[0,\left|\mathrm{Str}_1\right|\right]\right)$

7 $\mathrm{dist}_{0,j} \Leftarrow j, \forall\left(j \in \left[0,\left|\mathrm{Str}_2\right|\right]\right)$

8 **for** $i \in \left[1,\left|\mathrm{Str}_1\right|\right]$ **do**

9 $\mathrm{lower} \Leftarrow \max\left\{1, i-\theta_{ed}\right\}$

10 $\mathrm{upper} \Leftarrow \min\left\{\left|\mathrm{Str}_2\right|, i+\theta_{ed}\right\}$

11 **for** $j \in \left[\mathrm{lower},\mathrm{upper}\right]$ **do**

12 $\mathrm{dist}_{i,j} \Leftarrow \infty$

13 **if** $\mathrm{dist}_{i-1,j} \neq \mathrm{NULL}$ **then**

14 $\mathrm{dist}_{i,j} \Leftarrow \min\left\{\mathrm{dist}_{i-1,j}+1, \mathrm{dist}_{i,j}\right\}$

15 **if** $\mathrm{dist}_{i,j-1} \neq \mathrm{NULL}$ **then**

16 $\mathrm{dist}_{i,j} \Leftarrow \min\left\{\mathrm{dist}_{i,j-1}+1, \mathrm{dist}_{i,j}\right\}$

17 **if** $\mathrm{dist}_{i-1,j-1} \neq \mathrm{NULL}$ **then**

18 $\mathrm{dist}_{i,j} \Leftarrow \min\left\{\mathrm{dist}_{i-1,j-1}+1, \mathrm{dist}_{i,j}\right\}$

19 $\mathrm{Matched} \Leftarrow \left(\mathrm{dist}_{|\mathrm{Str}_1|,|\mathrm{Str}_2|} \leq \theta_{ed}\right)$

在算法 1 的伪代码中，第 1 行表示基于集合相似度的字符串匹配，在这里我们给它设定了较高的优先级。而第 3 到第 19 行是在编辑距离上限 θ_{ed} 较低的前提下的高效字符串匹配算法，与 Ukkonen 算法类似。

4. 基于高斯混合模型的离散化算法

数据级的异构性(Data-level heterogeneity)不仅表现在相同实体的多重表示上，还表现在标度的差异性上。标度差异性在这里指不同 Web 数据源间对应数据间的数值转换，比如房屋建筑面积与户型、建筑形式与总楼层间存在着离散分类值与连续数值间的映射关系。如表 2 和表 3，Web 表格中不仅包含了字符

串型属性，还包含有数值型属性。显然，第 3 节中的近似字符串匹配算法仅适用于处理离散的分类值属性与字符(串)值属性。与取值较为有限的离散值相比，连续的数值型属性更易被 5.2.1 节中的算法过滤掉。对于表格中的连续数值量，需进行离散化处理，将其分割为区间，并以区间对应的标号作为分类等级值，应用于后续的算法。

算法的目的在于自动化的处理过程中去发现由映射关系构成的约束规则，在映射关系合成之前，规则大部分都是未知的，传统上以规则为基础的离散化处理算法在这里并不适用。基于连续变量分布函数的数据离散算法曾在数据预处理中得到了广泛的应用。本文借助高斯混合模型进行区间分割，以实现离散化。

高斯混合模型[16]是指具有如下形式的概率分布模型：

$$P(y|\theta)=\sum_{k=1}^{K}\alpha_k\phi(y|\theta_k) \tag{5}$$

其中，α_k 为系数且 $\alpha_k\geq 0$，$\sum_{k=1}^{K}\alpha_k=1$，$\phi(y|\theta_k)$ 为高斯分布密度，$\theta_k=(\mu_k,\sigma_k^2)$，$\mu_k$ 为均值，σ_k 为标准差，且称 $\phi(y|\theta_k)=\frac{1}{\sigma_k\sqrt{2\pi}}e^{-\frac{(y-\mu_k)^2}{2\sigma_k^2}}$ 为第 k 个高斯分量($1\leq k\leq K$)。

通常，高斯混合模型与 EM (Expectation Maximization，期望最大)算法结合在一起，以一种类似于无监督聚类的形式来确定连续数据分布中的各个高斯分量。根据高斯分量对连续值进行区间划分，由区间标签得到离散的分类值。

除观测值外，离散化算法初始还需要由外部来源的统计数据确定高斯分量数 K，由经验值确定 EM 算法中的迭代次数。应用 EM 算法，将观测值分解为 K 个高斯分量，确定各个高斯分量对应的 μ_k,σ_k。由高斯分布的性质 $\int_{\mu_k-2\sigma_k}^{\mu_k+2\sigma_k}\frac{1}{\sigma_k\sqrt{2\pi}}e^{-\frac{(y-\mu_k)^2}{2\sigma_k^2}}=95\%$，以 $(\mu_k-2\sigma_k,\mu_k+2\sigma_k)$ 确定各划分区间，各划分区间标号为对应的离散值。在第 5 节中，近似映射关系的定义中允许 5%的属性值不符合映射关系，与之类似，在划分区间时取 95%，以避免异常值的干扰。

例 3.1：图 1 为 Web 表格语料库中某市房屋面积数据经过 GMM (Gaussian Mixture Model，高斯混合模型)区间划分的结果。

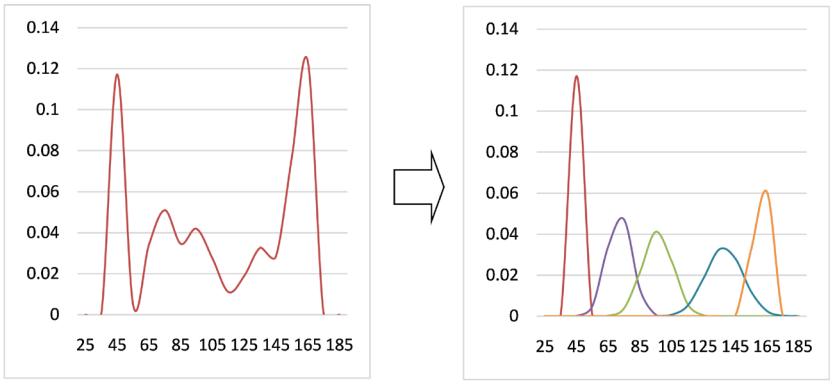

Figure 1. Preliminary division of intervals based on GMM
图 1. 基于 GMM 的区间初步划分

由高斯分布的性质，初步划分区间，得区间 A 为(38.15, 47.85)，区间 B 为(56.25, 87.75)，区间 C 为

(76.71, 115.29)，区间 D 为(114.32, 161.68)，区间 E 为(153, 169)。取各区间标号作为离散等级值。

对于区间重叠部分的处理，由文献[16]，引入了类典型度 CT (Class Typicality)的概念，其公式如下：

$$\mathrm{CT}_k = \frac{N_k}{N_{\mathrm{All}}} * \alpha_k \tag{6}$$

其中 N_k 为经过 EM 算法得到的第 k 个区间划分中的观测值个数；N_{All} 为观测值总数；α_k 为系数，即第 k 个高斯分量对应的权重。类典型度用于衡量各高斯分量间的“不均匀性”，较为典型的分量有较大的权重和较高的样本数，利用类典型度可以使争议样本点更集中于比较典型的分量。

基于类典型度，对区间重叠部分及间隔部分进行重分配：假设区间 k 与区间 $k+1$ 相邻且其间有重叠部分，区间 k 为 $\left(\mathrm{lower}_k, \mathrm{upper}_k\right)$，区间 $k+1$ 为 $\left(\mathrm{lower}_{k+1}, \mathrm{upper}_{k+1}\right)$，所得到的新区间为 $\left(\mathrm{lower}_k, \mathrm{upperr}'_k\right)$ 和 $\left(\mathrm{lower}'_{k+1}, \mathrm{upper}_{k+1}\right)$，由

$$\mathrm{upperr}'_k = \mathrm{upper}_k + \frac{\mathrm{CT}_k}{\mathrm{CT}_k + \mathrm{CT}_{k+1}} \mathrm{lower}_{k+1} - \mathrm{upper}_k \tag{7}$$

和

$$\mathrm{lower}'_{k+1} = \mathrm{lower}_{k+1} - \frac{\mathrm{CT}_{k+1}}{\mathrm{CT}_k + \mathrm{CT}_{k+1}} \mathrm{lower}_{k+1} - \mathrm{upper}_k \tag{8}$$

确定新的划分点。公式(7)和公式(8)表示依照类典型度，将重叠部分按比例分配给区间 k 和区间 $k+1$。

例 3.2：接例 3.1，区间 A、B、C、D、E 中的观测值个数分别为 71、85、435、232、49，观测值总数 N_{All} 为 762，权重均为 0.2，经计算得类典型度分别为：0.093、0.112、0.571、0.304、0.064。对区间重叠部分进行重分配后，区间 A 为(38.15, 51.68)，区间 B 为(51.68, 78.52)，区间 C 为(78.52, 114.95)，区间 D 为(114.95, 161.68)，区间 E 为(161.68, 169)。使用区间标签作为面积等级值，实现离散化。

5. 基于映射关系联结的领域知识抽取算法

第 3 节和第 4 节解决了 Web 表格语料库中的数据值层次的异构性问题。接下来，本节中将介绍一种在大量 Web 表格中抽取领域词典的算法，首先从 Web 表格语料库中去提取碎片化的映射关系，继而将这些映射关系联结起来，提高映射关系的完备性，最后通过冲突消解来进一步提高数据质量，成为可信的领域词典。

5.1. 相关概念

定义 1. 映射关系：令 R 为两个属性 X、Y 间的概念上的关系。如果对于所有的 $x \in X$，x 函数地确定有且仅有一个值 $y \in Y$，这一关系即为映射关系，表示为 $M(X,Y)$或 $X \to Y$。

定义 2. 近似映射关系：令 R 为两个属性 X，Y 之间概念上的关系，如果仅对 R 的某一子集 $\overline{R} \subset R$ 且 $\left|\overline{R}\right| \geq \theta|R|$，在这个子集中对于 X 中的任意值 $x \in X$，在 Y 中有且仅有一个值 $y \in Y$ 与之函数地对应，即仅在超过 $\theta|R|$ 大小的子集中满足映射关系，则称这一关系是 θ-近似映射关系，表示为 $M_\theta(X,Y)$ 或 $X \to_\theta Y$。在下文不引起混淆的情况下，将简略地用 θ 近似映射关系来代替映射关系，且 θ 取值为 95%。

定义 3. 映射关系图：一个无向的映射关系图 $G = (V,E)$，其中 V 是结点的集合，E 是边的集合。每个结点 v 对应一个属性对，边对应属性对间的关系权重(相容或相斥)。映射关系图中，边有权重，结点没有权重。边权重的绝对值介于 0 和 1 之间，反映结点间的相容(或相斥)性强度。

定义 4. 领域词典：领域词典的基本组成单位是(属性，值)或(实体 1，实体 2)二元组，子表是由符合相同映射关系的元组构成的二列表，领域词典是由子表构成的集合。由于领域词典由两种基本组成单位，

下文用列来表示映射关系的左、右部分，称二列表为列对。

5.2. 映射关系提取

在这一节中，进行基于点互信息的列过滤、并利用函数依赖约束来剔除低质量的候选列对。

5.2.1. 基于 PMI (点互信息)的列过滤

在领域知识的研究中，实体用来表示具有可区别性且独立存在的某种事务，语义类指具有某种共同属性的实体的集合，关系则是连接不同实体集合的“边”。抽取映射关系的首要任务是识别出 Web 语料库中各 Web 表中共有的语义类。

自然语言处理领域的相关研究中，代表相同语义领域或主题的两个词间具有语义相似性。基于这种理解，如果词汇涉及到的事物在现实世界中经常共同出现，从统计学上讲，尽管这些词汇可能指代不同的主体，但这些词汇属于同一类别[4]。

令 $s(u,v)$ 为值 u 和值 v 间的语义相似性，$\mathcal{C}(u)=\{C|u\in C,C\in T,T\in\mathcal{T}\}$ 表示在 Web 表格语料库 $\mathcal{T}$ 中包含值 u 的列集，同样 $\mathcal{C}(v)=\{C|v\in C,C\in T,T\in\mathcal{T}\}$ 指 Web 表格语料库 $\mathcal{T}$ 中包含值 v 的列集。如果集合 $\mathcal{C}(u)\cap\mathcal{C}(v)$ 的基数较大，就意味着 u 和 v 会频繁地共同出现(例如，当 u 为辽宁，v 为黑龙江时)。它们间具有较高的语义相似性。

在自然语言处理中，常采用 PMI (Pointwise Mutual Information，点互信息)来量化共现程度。

$$\mathrm{PMI}(u,v)=\log\frac{p(u,v)}{p(u)p(v)} \tag{9}$$

其中 $p(u)$ 和 $p(v)$ 是 Web 表格语料库 $\mathcal{T}$ 的所有 N 列中出现 u 和 v 的概率，定义为：$p(u)=\frac{|\mathcal{C}(u)|}{N}$，$p(v)=\frac{|\mathcal{C}(v)|}{N}$ 和 $p(u,v)=\frac{|\mathcal{C}(u)\cap\mathcal{C}(v)|}{N}$。

在这里，采用 NPMI (Normalized PMI，归一化 PMI) [17]，将 $s(u,v)$ 的取值范围调整为 $[-1,1]$：

$$s(u,v)=\mathrm{NPMI}(u,v)=\frac{\mathrm{PMI}(u,v)}{-\log p(u,v)} \tag{10}$$

公式(10)中的 NPMI 降低了 PMI 对频率的敏感性，同时使共现性高、低的分化更为显著。

由语义相似性 $s(u,v)$，列 $C=\{v_1,v_2,\cdots\}$ 中的语义一致性是该列中所有值对语义相似性的平均值 $\bar{C}$，即

$$\bar{C}=\frac{\sum_{v_i,v_j\in C,i<j}s(v_i,v_j)}{\mathbf{C}_{|C|}^2} \tag{11}$$

其中 $\sum_{v_i,v_j\in C,i<j}s(v_i,v_j)$ 表示列 C 中所有值对语义相似性总和，$\mathbf{C}_{|C|}^2$ 为列 C 中值对总数。

若列 C 的语义一致性低于阈值，则将列 C 过滤掉。

例 4：在表 1~表 5 中，大部分列中的实体或属性值都有着较高的共现程度，这些列的语义一致性也相应较高。但是表 2 中的“上市时间”列、表 3 的“单价”列、表 4 的“交通线路”列，由于它们的语义一致性较低，将这些列从语料库中移除。

而另一方面，如果在生成 Web 表格语料库时，部分 Web 表格中出现了数据提取错误、数据混淆，如数据对齐错误等，也会造成语义一致性 $\bar{C}$ 的大幅降低。

5.2.2. 基于函数依赖的候选表过滤

在移除个别语义一致性较低的列后，将每一个 Web 表格中保留下来的列两两连接为列对，即对于每个表 $T=\{C_1,C_2,C_3,\cdots,C_n\}$ 中的 n 列，可以预连接出 $2C_n^2$ 个有序列对 $\left\{\left(C_i,C_j\right)\middle|i,j\in[n],i\neq j\right\}$。这是因为 Web 表格中的每一列都有可能会与同一个表中其他任意一列连接成为关系，这些关系中可能包含有用信息，是待联结的候选映射关系。但这 $2C_n^2$ 个列对中并不都包含有意义的映射。

接下来用函数依赖检查这些二列表，来移除非映射列对。由定义 2 近似映射关系，将函数依赖的限制放宽到 95%，允许一部分命名歧义(像(铁西区→沈阳)和(铁西区→鞍山))。

通过上述两种过滤方法，可将大部分的无意义候选关系过滤掉。候选映射关系提取的算法如算法 2 所示。

算法 2：映射关系提取

输入：Web 表格语料库 $\mathcal{T}$

输出：候选属性对(列对)的集合 $\mathcal{B}$

```
1   B ⇐ φ
2   for each T ∈ 𝒯 do
3       T′ ⇐ φ
4       for each C_i ∈ T do
5           if C_i is not removed by PMI filter then
6               T′ ⇐ T′ ∪ {C_i}
7       for each C_i, C_j ∈ T′(i ≠ j) do
8           B ⇐ (C_i, C_j)
9           if B is not removed by FD filter then
10              ℬ ⇐ ℬ ∪ {B}
```

在算法 2 中，4~6 行和 7~9 行分别对应基于 PMI 的属性过滤和基于函数依赖的属性对过滤。

5.3. 关系图模型下的映射关系联结

在基于 PMI 的列过滤和基于函数依赖的列对过滤后，保留下来的列对就可被视为映射关系了，本文称其为候选表。在这一步中，我们将具有相同映射关系且彼此相容、没有冲突的候选表联结起来。

5.3.1. 映射关系间的相容性

首要的任务是确定将候选列对是否应当被拼合起来，需要以候选表之间的相容性作为支持映射关系联结的依据。

定义 5.相容性：相容性用以衡量候选表间包含的是同一映射关系的可能性大小。假设候选表 B 满足映射关系 R，候选表 B'满足映射关系 R'，若 R 和 R'是同一映射关系，则称候选表 B 和候选表 B'是相容的。相容性用以衡量候选表间包含的映射关系一致性程度。

首先，可由值对的重叠度来计算两个映射关系间的相容性。令 $B=\left\{\left(l_i,r_j\right)\right\}$ 和 $B'=\left\{\left(l_i',r_j'\right)\right\}$ 为两个二列映射关系，其中每一个都是由(左，右)值对构成的集合。如果这两个关系中共有很多相同的值对，也就是说其交集中元素个数 $\left|B\cap B'\right|$ 很大，那么这两个映射关系可能是一致的。

基于重叠度的相容性表示为 $J^+\left(B,B'\right)$：

$$J^{+}\left(B,B'\right)=\max\left\{\frac{\left|B\cap B'\right|}{\left|B\right|},\frac{\left|B\cap B'\right|}{\left|B'\right|}\right\} \tag{12}$$

其中，$J^{+}\left(B,B'\right)$为 B 与 B'间的相容性。与公式(3)类似，$J^{+}\left(B,B'\right)$也被定义成一种对称的包含度，因为 B,B'间的相容性和 B',B 间的相容性在本质上是相同的，即 $J^{+}\left(B,B'\right)=J^{+}\left(B',B\right)$。

其次，提高完备性的目的在于去确定哪些候选表源于相同的广义域。本文采用的是开放世界假设，这与经典数据库中的封闭世界假设不同。单一的Web表格数据不是完备的，其中的映射关系覆盖面有限，需要进行扩展。而仅仅依靠重叠度作为相容性判据是不够的。

局部映射关系与全局映射关系的共有部分遵循超几何分布：假设全局域 D 是由有限离散值对组成的集合，是理想中的完备映射关系，候选表 B 和候选表 B'分别取自两个不同的 Web 表格。为了评估候选表 B 和候选表 B'是否来自同一域 D，通过超几何分布，可使用 B 与 B'的交集大小来评估它们共同来自域 D 的概率[18]。B 和 B'中的值对都属于同一个域 D 的可能性大小表示为公式(13)，

$$P\left(m\mid n_a,n_b,n_D\right)=\frac{C_{n_a}^{S}\cdot C_{n_D-n_a}^{n_b-m}}{C_{n_D}^{n_b}} \tag{13}$$

其中 $n_a=\left|B\right|$，$n_b=\left|B'\right|$，$n_D=\left|D\right|$。由超几何概型，假设候选表 B 中的某值对属于域 D，候选表 B'中的值对由域 D 中随机抽取(不放回)，若取到的值对恰好属于 B，则视为随机试验成功，$P\left(m\mid n_a,n_b,n_D\right)$表示 m 次随机试验成功($m\in\left\{1,2,3,\cdots,\left|B\cap B'\right|\right\}$)的概率。

在开放世界假设下，不可能获取到域 D 的基数，在此将 D 近似为 B 和 B'的并集，$n_D=n_a+n_b$。实际上，在交集大小固定的情况下，选择更大的 D 会增加 B 和 B'间的相容性。

由超几何分布的分布函数，可得到候选表 B 和候选表 B'间的集合相容性 $C_h^{+}\left(B,B'\right)$：

$$C_h^{+}\left(B,B'\right)=F\left(t\mid B,B'\right)=\sum_{m\in[0,t]}P\left(m\mid n_a,n_b,n_D\right) \tag{14}$$

其中 $t=\left|B\cap B'\right|$，是候选表 B 和候选表 B'共有值对的数量。

最终的属性对相容性是综合重叠度相容性和超几何分布相容性之后的结果，取最大值：

$$C^{+}\left(B,B'\right)=\max\left\{J^{+}\left(B,B'\right),C_h^{+}\left(B,B'\right)\right\} \tag{15}$$

例 5：候选表提取后，将表 1 中“小区名称－小学学区”记为 a，表 4 中“名称－学区 A”记为 b，表 5 中“项目名称－周边小学”记为 c，以重叠度来计算相容性，a 与 b 间为 0.75，a 与 c 间为 0.5，b 与 c 间为 0.66。而实际中表 1 与表 4 的覆盖面不足，若考虑基于超几何分布的相容性，a 与 b 间为 0.923，a 与 c 间为 0.085，b 与 c 间为 0.615。现实中的 Web 表格中通常仅涵盖了很小一部分信息，如有些表格仅包含某个区或某个开发商的信息，借助超几何分布可进一步提高完备性。

5.3.2. 映射关系间的相斥性

以统计学习为基础的算法难以像领域专家一样去理解映射关系的真正含义。在本文的例子中，关于“学区”与关于“周边学校”的映射关系就难以被自动算法区分，类似的还有“街道－社区”、“区域－板块”。

因此，仅依靠相容性来联结映射关系是不够的，这有可能引入新的错误。有的取值重合的属性在语义上未必是相关的。同样，隐含不同关系的表格间，有时也会有大量的值对重叠。这正是传统的基于模式匹配算法的不足之处。但不同的映射关系间也包含有冲突的值对，这违反了映射关系的定义，表明了它们不相容，尽管由公式(15)计算出的相容性值较高。

与 5.3.1 节的相容性相反，定义候选表间的相斥性。由于两个表 B 和 B'间相斥性也是对称的，与公式(12)类似，相斥性 $R^{-}\left(B,B'\right)$ 为：

$$R^{-}\left(B,B'\right)=-\max\left\{\frac{\left|\text{Conflict}\left(B,B'\right)\right|}{\left|B\right|},\frac{\left|\text{Conflict}\left(B',B\right)\right|}{\left|B'\right|}\right\} \tag{16}$$

其中 $\left|\text{Conflict}\left(B,B'\right)\right|$ 和 $\left|\text{Conflict}\left(B',B\right)\right|$ 分别表示 B 和 B'、B'和 B 间冲突集的元素个数。给定两个表格 B 和 B'，它们间的冲突集为 $\text{Conflict}\left(B,B'\right)=\left\{l\middle|\left(l,r\right)\in B,\left(l,r'\right)\in B',r\neq r'\right\}$，也就是左值相同，而右值不同的值对集合。

为提高效率，利用倒排索引重组候选表集合，将仅在 B 和 B'共有多于阈值 θ_{overlap} 的值对时(左值、右值同时)才去计算 $C^{+}\left(B,B'\right)$。类似地，仅在 B 和 B'共有多于阈值 θ_{overlap} 的左值时才去计算 $R^{-}\left(B,B'\right)$。$\theta_{\text{overlap}}=10$。

例 6：接例 5，a 与 c 间的相斥性为−0.5，b 与 c 间的相斥性为−0.4。

5.3.3. 映射关系联结

在提出了评估可联结性的相容性和相斥性后，用图 $G=\left(\mathcal{B},E\right)$ 来表示候选属性对及它们之间关系，其中 $\mathcal{B}$ 是所有候选属性对的集合，图 G 中每个顶点表示一个列对 $B\in\mathcal{B}$。对于图 G 中每对顶点 B,B'，其间的关系 E 由相容性值 $C^{+}\left(B,B'\right)$ 和相斥性值 $R^{-}\left(B,B'\right)$ 表示，分别表示为带权边的正权重和负权重。

在图 $G=\left(\mathcal{B},E\right)$ 中，执行映射关系的联结做法如下：首先，相斥性 $R^{-}\left(B,B'\right)$ 具有“绝对否决权”，若两顶点 B,B' 间存在负权重边，则这两个属性对就不会被联结；而对于两顶点 B,B' 间的正权重边，从全图中最大的开始联结，合并顶点后更新顶点 B^{*} 与其他顶点间的正、负权重，相容性更新为 B,B' 与相应顶点的正权重之和，而相斥性取 B,B' 与相应顶点的负权重中绝对值的最大值。映射关系联结的过程如图 2 所示，其中的相容性和相斥性数值来源于例 5 和例 6。

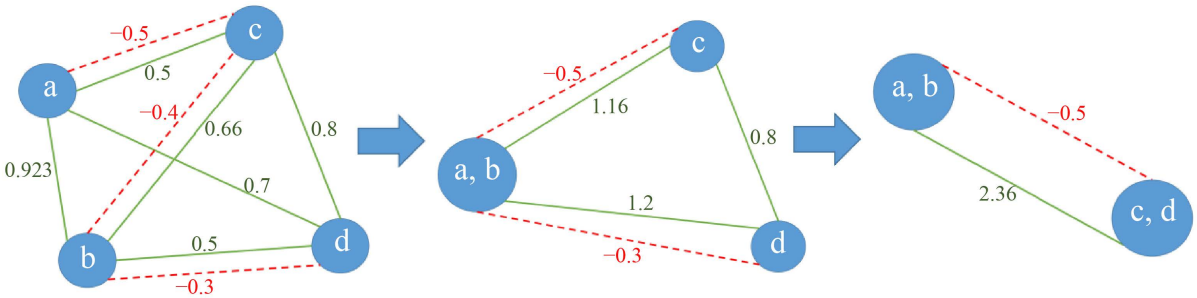

Figure 2. A diagram of mapping relationships synthesizing
图 2. 映射关系联结示意图

为降低计算复杂性，采取分治策略，在分区(子图)中进行映射关系联结[19]。将图 G 中的顶点集 $\mathcal{B}$ 视为分区(子图) P_i 的集合 $\mathcal{B}=\left\{P_1,P_2,\cdots\right\}$，($P_i\subseteq\mathcal{B}$)。由于映射关系联结的目标是将相容的候选表合成为更为完备、更全局化的映射关系，分区(子图)中的元素是能够被联结在一起的映射关系。$\mathcal{P}$ 为所有分区的集合，其中不同的分区对应不同的映射关系，分区间应当是不相交的($P_i\bigcap P_j=\phi,i\neq j$)，且它们应当合在一起构成 $\mathcal{B}$，即 $\bigcup_{P\in\mathcal{P}}P=\mathcal{B}$。

将 $\mathcal{B}$ 划分为不相交的分区(子图)的原则是：1) 将相容表分到一起，以尽可能地增加单个映射关系的覆盖范围；2) 相斥表不应当放到同一个分区(子图)中。具体来说：

一方面，每个分区(子图) P 中应当包含尽可能多的相容候选表。设 $C^{+}\left(P\right)$ 为分区 P 中候选表相容性

之和， $C^+(P)=\sum_{B_i,B_j\in P,i<j}C^+(B_i,B_j)$，在所有分区(子图)中应最大化其总和，即 $\text{maximize}\sum_{P\in\mathcal{P}}C^+(P)$。

另一方面，将相斥性 R^- 小于阈值 τ 的边视为强制约束，即不容许将这样的相斥表放在同一个分区(子图)中。在这里为了不过度惩罚由轻微质量问题和提取问题引起的略微相斥的表，用阈值 τ 来代替 0。在实际的操作中，若 $0>R^-(P)>\tau$，将 R^- 赋值为 0。设 $R^-(P)$ 为 P 中候选表相斥性之和，$R^-(P)=\sum_{B_i,B_j\in P,R^-(B_i,B_j)<\tau}R^-(B_i,B_j)$，所有分区(子图)中应满足约束 $\sum_{P\in\mathcal{P}}R^-(P)=0$。

基于上述原则，映射关系联结算法如算法 3 所示。

算法 3：映射关系联结

输入：图 $G=(B,E)$，阈值 τ

输出：分区 $\mathcal{P}$ 的集合

$P(B_i)\Leftarrow\{B_i\},\ \forall(B_i\in\mathcal{B})$

$\mathcal{B}_P\Leftarrow\bigcup_{B_i\in\mathcal{B}}\{P(B_i)\}$

$E_P\Leftarrow\bigcup_{(B_i,B_j)\in E}\{(P(B_i),P(B_j))\}$

$C_P^+(P(B_i),P(B_j))\Leftarrow C^+(B_i,B_j)$

$R_P^-(P(B_i),P(B_j))\Leftarrow R^-(B_i,B_j)$

$G_P\Leftarrow(\mathcal{B}_P,E_P)$

while true **do**

  $e(P_1,P_2)\Leftarrow\underset{P_1\neq P_2,R_P^-(P_1,P_2)>\tau}{\arg\max}\left(C_P^+(P_1,P_2)\right)$

  if $e=\text{NULL}$ **then**

    break

  $P'\Leftarrow P_1\cup P_2$

  Add P' and related edges into $\mathcal{B}_P$ and E_P

  for each $P_i\notin\{P_1,P_2\}$ **do**

    $C_P^+(P_i,P')\Leftarrow C_P^+(P',P_i)\Leftarrow C_P^+(P_i,P_1)+C_P^+(P_i,P_2)$

    $R_P^-(P_i,P')\Leftarrow R_P^-(P',P_i)\Leftarrow\min\{R_P^-(P_i,P_1),R_P^-(P_i,P_2)\}$

  Remove P_1,P_2 and related edges from $\mathcal{B}_P$ and E_P

$\mathcal{P}\Leftarrow\mathcal{B}_P$

在算法 3 中利用了贪心算法思想，首先将每一个顶点视为一个分区。然后，迭代合并当前相容性最高的一对分区 (P_1,P_2) 以获得新的分区 P'，同时更新剩下的正权边和负权边。当没有可以合并的分区时，算法终止。其中阈值 τ 是由参数优化实验取得，篇幅所限，本文未予介绍，取值为 $\tau=-0.2$，在性能和输出质量之间得到了平衡。

5.4. 冲突消解

映射关系联结的结果中存在着不一致的映射，也就是说同一映射中的两对值具有相同的左侧值，但不同的右侧值(这违反了映射关系的定义)。此问题的出现有三大原因：1) 某些不相容候选表之间的冲突

值对不足，造成相斥性较低，无法阻止映射关系联结；2) 近似映射关系的定义中允许了部分不一致、映射关系联结时允许将小于阈值 τ 的相斥性赋值为零；3) 爬虫算法获取数据时出现 Web 表格提取错误、数据质量问题。在此加入冲突消解步骤，删除联结结果中的部分不一致映射，以提高质量。

冲突消解的目标是找到 $\mathcal{P}$ 的最大子集 P_T，使 P_T 中任两个表彼此间都没有冲突，形式如下：

$$\begin{aligned} &\text{maximize} \left| \bigcup_{B_i \in P_T} B_i \right| \\ &\text{subject to Conflict}\left(B_i, B_j\right) = \varnothing,\ \forall B_i, B_j \in P_T \end{aligned} \tag{17}$$

其中冲突集 $\text{Conflict}\left(B_i, B_j\right) = \left\{ l \middle| (l,r) \in B_i, (l,r') \in B_j, r \neq r' \right\}$ 由 5.3.2 节定义，$\mathcal{P}$ 为由候选表 $\{B_1, B_2, \cdots\}$ 组成的集合，$\mathcal{P}$ 中每一个 B_i, B_j 都是值对 (l,r) 的集合。

算法 4： 冲突消解

输入： 候选表(映射关系)集合 $\mathcal{P}$

输出： 无冲突候选表(映射关系)集合 P_T

1 $P_T \Leftarrow P$

2 **while** $\exists B_i, B_j \in P_T, \left|\text{Conflict}\left(B_i, B_j\right)\right| > 0$ **do**

3 InitSet $\Leftarrow \bigcup_{B_i \in P_T} B$

4 **for each** $(v_1, v_2) \in$ InstSet **do**

5 $\text{count}_{\mathcal{P}}(v_1, v_2) \Leftarrow$ #conflicting value pairs in InitSet

6 **for each** $B_i \in P_T$ **do**

7 $\text{count}_B(B_i) \Leftarrow \max_{(v_1, v_2) \in B_i} \left\{ \text{cnt}_{\mathcal{P}}(v_1, v_2) \right\}$

8 $B_i \Leftarrow \arg\max_{B_i \in P_T} \text{cnt}_B(B_i)$

9 $P_T \Leftarrow P_T \setminus \{B_i\}$

冲突消解如算法 4 所示，第 3 行到第 5 行对冲突的值对的数量进行计数。第 6 行到第 9 行查找引入冲突最多的候选属性对，并将其删除。

6. 实验

6.1. 数据集

本文实验的 Web 表格语料库来源于中国土地网、链家网等房产信息网站，通过爬虫技术，从上述网站获得了 100 多万条 Web 表格记录，如表 1~表 5 所示，以从中抽取映射关系。

6.2. 评价指标

为了评估本文所提出的映射关系联结算法的可行性和有效性，并与其他方法进行比较，本文采用精确度(precision)、召回率(recall)、f1-score 作为算法评价方式。由领域专家为实验提供了高质量的映射关系作为参考标准，表示为 $B^* = \left\{ \left(l^*, r^*\right) \right\}$，通过算法从表格语料库中自动抽取的映射关系表示为 $B = \left\{ (l,r) \right\}$，算法的精确度为 $\text{Precision} = \dfrac{\left|B \cap B^*\right|}{\left|B\right|}$，召回率为 $\text{recall} = \dfrac{\left|B \cap B^*\right|}{\left|B^*\right|}$，f1-score 为 $\text{f1}_{\text{score}} = 2\text{precision} * \text{recall} / \text{precision} + \text{recall}$。

6.3. 实验结果

表 6 是本文方法(映射关系联结)与其他两种可用于 Web 表格属性融合的算法 PROSPERA [20]、WISE-Integrator [21]之间的运行结果比较。如表 6 所示，在对 Web 表格语料库上的抽取结果中，选取“面积－户型”、“街道－社区”和“小区－学区”关系，以对比抽取的领域词典质量。本文所提出的方法都能获得较好的精确度、召回率和 f1-score。WISE-Integrator 算法的质量较低，这是因为该算法依靠属性名称和值类型等信息来衡量属性之间的语义相似性，并通过聚类将相似的属性归为一组，而没有考虑冲突。

Table 6. The comparison of experimental results
表 6. 实验结果对比

	WISE-Integrator			PROSPERA			映射关系联结		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
面积－户型	78.62	65.33	71.36	86.21	68.48	76.33	97.63	78.37	86.95
街道－社区	86.46	65.30	74.40	95.41	81.28	81.78	96.85	85.66	90.91
小区－学区	75.83	64.91	69.94	93.45	81.52	87.08	95.23	86.98	90.92

为了比较算法的运行时间，本文分别从表格语料库中选取 20%、50%、100%的数据作为输入。比较结果如图 3 所示，WISE-Integrator 算法的运行时间相比更短，但该算法牺牲的是知识抽取质量，其精确度、召回率和 f1-score 都明显落后于 PROSPERA 算法和映射关系联结算法。在 50%的数据规模下，PROSPERA 算法和映射关系联结算法的运行时间相近，且各项指标相近，但在更大的数据规模下，由于 PROSPERA 算法需要靠迭代来保证输出质量，运行时间也相应大幅度提高，这是由于 PROSPERA 算法侧重于处理复杂的自然语言，未对 Web 表格语料库进行优化，而映射关系联结算法，即使在 100%数据规模下，也能保持较短的运行时间，并获得与 PROSPERA 算法相近的输出质量。

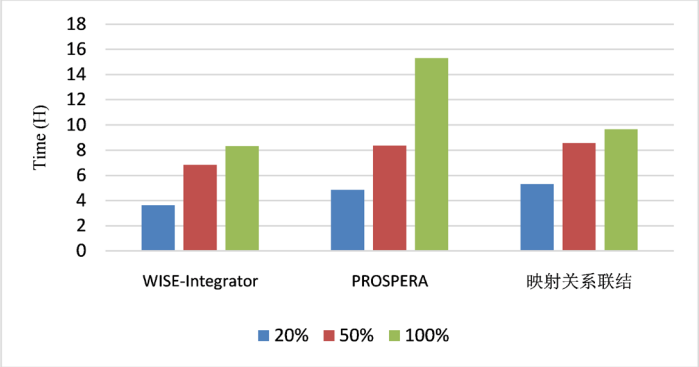

Figure 3. Run time comparison
图 3. 运行时间对比

7. 总结

本文提出了一种基于映射关系的领域词典抽取算法。首先介绍了映射关系与领域词典的联系，并从领域词典的来源方面，指出了 Web 表格中的数据异构性及不完备问题。为解决数据异构性问题，本文提出了 IDF-Jaccard 最大包含度、相对编辑距离阈值的概念，给出一种短文本近似字符串匹配算法，并提出

了一种基于高斯混合模型的离散化算法。接下来通过基于 PMI 的列过滤和函数依赖过滤，移除了非映射关系候选表，得到了 Web 表格中的映射关系。为提高完备性，提出了映射关系间的相容性和相斥性，在映射关系图中实现了映射关系联结，并通过冲突消解保证了算法输出结果质量。最后，在房地产大数据上的对比实验中，验证了本文算法的有效性、可靠性和可扩展性。

参考文献

[1] 王鑫, 邹磊, 王朝坤, 彭鹏, 冯志勇. 知识图谱数据管理研究综述[J]. 软件学报, 2019, 30(7): 2139-2174. http://www.jos.org.cn/1000-9825/5841.htm

[2] Abadi, D.J., Marcus, A. and Madden, S.R. (2009) SW-Store: A Vertically Partitioned DBMS for Semantic Web Data Management. *VLDB Journal*, **18**, 385-406. https://doi.org/10.1007/s00778-008-0125-y

[3] Abadi, D.J., Marcus, A. and Madden, S.R. (2007) Scalable Semantic Web Data Management Using Vertical Partitioning. In: Klas, W., Ed., *Proceedings of the* 33*rd International Conference on Very Large Data Bases*, VLDB Endowment, Vienna, 411-422.

[4] 陈文亮, 朱靖波, 朱慕华, 姚天顺. 基于领域词典的文本特征表示[J]. 计算机研究与发展, 2005, 42(12): 2155-2160.

[5] 宋施恩, 樊兴华. 基于词共现和词上下文的领域观点词抽取方法[J]. 计算机工程与设计, 2013, 34(11): 4012-4015.

[6] Venetis, P., Halevy, A.Y., Madhavan, J., Pasca, M., Shen, W., Wu, F., Miao, G.X. and Wu, C. (2011) Recovering Semantics of Tables on the Web. *PVLDB*, **4**, 528-538. https://doi.org/10.14778/2002938.2002939

[7] Hearst, M.A. (1992) Automatic Acquisition of Hyponyms from Large Text Corpora. *Proceedings of the* 14*th Conference on Computational Linguistics*, Volume 2, 539-545. https://doi.org/10.3115/992133.992154

[8] Banko, M., Cafarella, M.J., Soderland, S., Broadhead, M. and Etzioni, O. (2007) Open Information Extraction from the Web. *Proceedings of IJCAI*, Volume 7, 2670-2676.

[9] Lehmberg, O. and Bizer, C. (2017) Stitching Web Tables for Improving Matching Quality. *PVLDB*, **10**, 1502-1513. https://doi.org/10.14778/3137628.3137657

[10] Ling, X., Halevy, A.Y., Wu, F. and Yu, C. (2013) Synthesizing Union Tables from the Web. *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, Beijing, 3-9 August 2013, 2677.

[11] Kang, J. and Naughton, J.F. (2003) On Schema Matching with Opaque Column Names and Data Values. *SIGMOD* 2003, San Diego, 9-12 June 2003, 205-216. https://doi.org/10.1145/872757.872783

[12] Nargesian, F., Zhu, E.K., Pu, K.Q. and Miller, R.J. (2018) Table Union Search on Open Data. *Proceedings of the VLDB Endowment*, **11**, 813-825. https://doi.org/10.14778/3192965.3192973

[13] Chaudhuri, S., Ganti, V. and Kaushik, R. (2006) A Primitive Operator for Similarity Joins in Data Cleaning. 22*nd International Conference on Data Engineering* (*ICDE*'06), Atlanta, 3-7 April 2006, 5. https://doi.org/10.1109/ICDE.2006.9

[14] Agrawal, P., Arasu, A. and Kanshik, R. (2010) On Indexing Error-Tolerant Set Containment. In: *Proceedings of the* 2010 *International Conference on Management of Data*, ACM Press, Indianapolis, 927-938. https://doi.org/10.1145/1807167.1807267

[15] Ukkonen, E. (1985) Algorithms for Approximate String Matching. *Information and Control*, **64**, 100-118. https://doi.org/10.1016/S0019-9958(85)80046-2

[16] Khanmohammadi, S. and Chou, C.-A. (2016) A Gaussian Mixture Model Based Discretization Algorithm for Associative Classification of Medical Data. *Expert Systems with Applications*, **58**, 119-129. https://doi.org/10.1016/j.eswa.2016.03.046

[17] Bouma, G. (2009) Normalized (Pointwise) Mutual Information in Collocation Extraction. *Proceedings of GSCL*, Potsdam, September 2009, 31-40.

[18] Miller, R.J., Nargesian, F., Christodoulakis, C., Pu, K.Q. and Andritsos, P. (2018) Making Open Data Transparent: Data Discovery on Open Data. *IEEE Data Engineering Bulletin*, **41**, 59-70.

[19] Wang, Y. and He, Y. (2017) Synthesizing Mapping Relationships Using Table Corpus. *Proceedings of the* 2017 *ACM International Conference on Management of Data*, *SIGMOD Conference* 2017, Chicago, 14-19 May 2017. https://doi.org/10.1145/3035918.3064010

[20] Nakashole, N., Theobald, M. and Weikum, G. (2011) Scalable Knowledge Harvesting with High Precision and High Recall. *Conference on Web Search and Data Mining* (*WSDM* 2011), Hong Kong, 9-12 February 2011, 227-236.

https://doi.org/10.1145/1935826.1935869

[21] He, H., Meng, W., Yu, C.T. and Wu, Z. (2003) WISE-Integrator: An Automatic Integrator of Web Search Interfaces for E-Commerce. *Proceedings* 2003 *VLDB Conference*, Berlin, 9-12 September 2003, 357-368. https://doi.org/10.1016/B978-012722442-8/50039-2