软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2020,31(3):893–908 [doi: 10.13328/j.cnki.jos.005911]

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

数据集成方法发展与展望*

王 淞, 彭煜玮, 兰 海, 罗倩雯, 彭智勇

(武汉大学 计算机学院,湖北 武汉 430072)

通讯作者: 王淞, E-mail: xavierwang@whu.edu.cn

摘 要: 数据集成在数据管理与分析领域起着重要的作用.尽管从学术界首次提出并开始研究数据集成问题已经过去 30 多年,但在各个领域仍然存在着大量与数据集成问题密切相关的问题亟待解决.对数据集成领域从 2001 年开始到现在相关工作的发展脉络进行了梳理与总结.通过追踪数据集成方法的发展轨迹,不仅可以了解前人在解决该问题时所作出的努力以及发掘出的研究方向,还可以进一步了解各个数据发展领域所研究问题的成因以及发展脉络.最终,通过分析近几年数据集成方面的工作,可以进一步展望未来在数据集成领域的潜在研究方向,为从事相关领域研究的学者提供参考.

关键词: 大数据;数据集成;数据管理;网页表;众包

中图法分类号: TP18

中文引用格式: 王淞,彭煜玮,兰海,罗倩雯,彭智勇.数据集成方法发展与展望.软件学报,2020,31(3):893–908. http://www.jos.org.cn/1000-9825/5911.htm

英文引用格式: Wang S, Peng YW, Lan H, Luo QW, Peng ZY. Survey and prospect: Data integration methodologies. Ruan Jian Xue Bao/Journal of Software, 2020,31(3):893–908 (in Chinese). http://www.jos.org.cn/1000-9825/5911.htm

Survey and Prospect: Data Integration Methodologies

WANG Song, PENG Yu-Wei, LAN Hai, LUO Qian-Wen, PENG Zhi-Yong

(School of Computer Science, Wuhan University, Wuhan 430072, China)

Abstract: Data integration plays a very important role in data management and analytical area. Although there have been decades since the data integration problem was first proposed, there are many data integration problems that remain unsolved. This study surveys the works in data integration area from 2001 until now. By categorizing these papers and their methodologies, it is able to summarize how these works develop and how their research topics shift from time to time. Several research topics are also filtered out that draw much attention recently and hopefully the survey and conclusions may provide guidance to the related researchers.

Key words: big data; data integration; data management; Web table; crowdsourcing

大数据管理是大数据时代所面临的重要挑战之一,学术界、工业界都对大数据管理相关内容展开了研究[1].大数据管理场景中重要的一个问题是数据集成(data integration).数据集成工作对大数据管理具有重要的意义,也是大数据管理任务需要解决的重要问题之一[2].目前,尽管已经有若干对数据集成的相关综述性文献[3,4],现有的综述性文献描述的都是框架层面的数据集成方法,而对数据集成的具体技术发展脉络的总结与提炼则比较少.因此在本文中,主要对数据集成问题的发展脉络进行归纳与总结.通过对前人工作的学习与理解,我们不仅可以把握之前针对数据集成研究的发展脉络,还可以对未来数据集成研究的发展方向进行一定的预测.

* 基金项目: 国家重点研发计划(2016YFB1000701)
Foundation item: National Key Research and Development Program of China (2016YFB1000701)
本文由人工智能赋能的数据管理、分析与系统专刊特约编辑李战怀教授、于戈教授和杨晓春教授推荐.
收稿时间: 2019-07-20; 修改时间: 2019-09-10; 采用时间: 2019-11-25; jos 在线出版时间: 2020-01-10
CNKI 网络优先出版: 2020-01-10 13:34:52, http://kns.cnki.net/kcms/detail/11.2560.TP.20200110.1334.011.html

数据集成指通过对数据源进行融合,发现来自不同数据源中指代相同含义、实体的数据之间的关联关系的研究.在实际应用中,数据集成工作对大数据的管理、查询以及分析场景都具有重要的作用.

- 数据管理.对海量数据的高效管理是大数据时代所面临的重要挑战之一,良好的数据管理策略可以极大地提升数据质量、访问效率,同时降低对数据的维护成本.而有效的数据集成方法则可以帮助用户提出更好的数据管理策略.例如,数据集成技术可以识别出具有关联关系的数据和文件,通过将具有关联关系的数据存储于集群中的相同节点上,在访问这批数据时可以极大地提升对数据的访问效率;通过梳理海量数据之间的逻辑关联结构,可以让管理者更容易理清当前所管理的数据结构以及基本信息;
- 数据查询.对海量数据的快速查询与检索,需要高质量索引结构的帮助.而数据集成技术可以辅助构建更高质量的索引.例如,通过理解不同数据源之间的数据关联关系,结合实际应用需求,用户可以针对集成后的数据集构建索引.相比传统索引技术只能对单一数据源的查询进行加速,面向全局模式的索引可以对面向多数据源的复合查询进行加速;
- 数据分析.数据分析与数据集成工作是相互融合与促进的,数据分析技术可以帮助用户更好的理解数据的含义,提升数据集成的质量;而对数据进行集成则可以帮助数据分析工作得到更加丰富的成果.例如在数据分析过程中,对一些含义不明确的数据,可以进一步集成互联网上的相关信息(维基百科等),从而更好地理解数据含义,提升数据分析质量.

通过以上分析可以看出,数据集成对于大数据的管理、查询与分析都具有重要价值.在本文中,我们主要梳理了从 2001 年开始的数据集成领域的相关文献,对不同文献所处理的问题以及挑战进行归纳总结,从而发现数据集成领域研究近十几年的发展脉络.在此基础上,通过分析近几年数据集成领域的研究成果,我们可以通过分析当前主流的研究兴趣与方向,为未来数据集成领域的研究方向进行一定的预测.

值得一提的是,尽管本文对现有的文献进行了分类,但这种分类方法非常主观,是作者根据自身理解,结合自身研究经验给出的一种粗略的分类方法.该分类方法可以借鉴,但不应当作为严格的分类标准.同时,考虑到学术研究的综合性,将一个工作划分给其中一个分类并不代表其不具备其他分类的特征或内容.相反,本文所介绍的绝大多数文献都是综合性的,其中内容涵盖了多个分类区域.文中的分类主要是根据文章的主要贡献更加符合的类别来划分的.

本文首先介绍早期的数据集成算法所解决的问题及其方法.接下来依据后续文献主要解决的问题以及贡献,将其划分为算法、系统、网页表格与众包这 4 个大方向进行介绍.在此基础上,我们进一步挑选不同领域具有代表性的文献,对其解决的问题以及方法进行深入介绍.最后重新梳理数据集成领域研究的发展脉络,并对未来的发展方向提出预测.

1 发展概要

为了能够更好地理解数据集成技术的发展脉络,本节中首先介绍不同分类下的数据集成技术的主要特征,结合具体的例子说明数据集成技术在不同发展阶段的主要目标,并介绍不同分类下的代表性技术与文献.

数据集成最初的任务目标是找到给定数据集内数据列、数据元组之间的关系.通过将代表相同属性的属性列、指代相同实体的元组之间建立联系,可以起到从数据集中发现更多知识的目的.以图 1 为例,假设最初的数据集中只有表 1 和表 2,表 1 描述的是一些书籍名字和对应的书籍作者、语言等信息,表 2 描述的是一些作者及其国籍.通过数据集成,我们可以将表 1 中的元组与表 2 的元组按作者名字对应起来,从而得到例如“《悲惨世界》的作者是法国人”这样的信息.而这条信息并不单独存在于表 1 或表 2 中,这就是通过数据集成从而获得更加丰富的数据信息的实例.

当然,这样的规则是可以通过人工定义的,比如用户完全可以在查询语句中手动将表 1 的作者与表 2 的名字列进行联系,或者将这样的规则手动写入数据库中.然而,当数据库中存在成千上万甚至更多张表时,依靠人工将所有的关联关系全部找出来是不现实的.这也是数据集成研究最初的动力:设计一种方法,能够自动地识别出数据集中潜在的数据集成关系.围绕这一目标,有许多潜在问题需要解决,例如,图 1 中的表 1 与表 3 描述的实

体都是“悲惨世界”“哈利波特”等,但是表的第 2 列关于人物的信息则完全不同.人之所以能分辨表 1 属于书籍而表 3 属于电影,是因为我们具有“作者与书籍相关”而“导演与电影相关”这两条先验知识.但是由于这两条先验知识并不天然存在于数据库中,因此,如何确定这些依靠原始数据集无法发现的数据关联,是后续数据集成研究需要解决的问题.事实上,针对这些问题,后续的数据集成文献开始向不同的研究方向进行推进,包括引入互联网数据、结合众包技术、研发数据集成系统等.在介绍这些工作之前,我们首先介绍围绕数据集成算法本身的文献后续发展过程及其研究目标.

Fig.1 Example of data integration

图 1 数据集成应用示例图

为了解决少量数据集提供的语义信息不足以支撑数据集成分析问题时,一种可行的思路是继续扩充数据集.例如,当我们发现只依靠表 1 与表 3 的信息无法准确消除“悲惨世界”这个实体在两张表中引起的歧义时,我们可以进一步导入新的数据集.例如,我们可以将表 4 导入到数据库中.表 4 描述的是一些经典世界名著的名字以及作者,其中就包含了“悲惨世界”以及“雨果”这两条信息.由于这两条信息与表 1 中的内容相匹配,我们可以认为表 1 的“悲惨世界”这一条元组描述的是书籍,进而推断表 1 中的数据全部是书籍的信息.然而,能够发现这条知识的前提是我们能够在表 4 中及时发现“悲惨世界”这条与表 1 信息相匹配的内容.可以看到,表 4 中包含的内容非常多,因此我们需要设计一种方法,能够快速从表 4 中发现我们需要的信息.这正是后续数据集成技术发展的主要方向之一:如何从庞大的数据集中快速、高效地发现其中潜在的关联关系.

另一种数据集成的技术思路是引入互联网数据.互联网中蕴藏着大量潜在的知识信息,例如,我们可能在互联网上看到类似图 1 中新闻 1 这样的信息.这一信息可以帮助我们识别“李安”属于电影导演,将这条知识与表 3 中的数据相结合,我们也可以得出表 3 的数据描述的是电影信息这样的结论.这样将互联网中的信息与本地数据库数据结合进行数据集成的技术,也是目前数据集成领域采用的主流技术之一.在此基础上,人们甚至可以直接从互联网上获取数据来进行集成,直接通过互联网挖掘有价值的知识信息.

在对网页数据进行集成时,有几大挑战需要解决.

- 首先是数据信息的提取.网络上的很多数据可能都是类似于图 1 中的文本数据,在对这些数据进行集成时,首先需要将其转换成数据集成处理的结构化数据,在这一过程中,如何实现数据格式的转换是第一大挑战;
- 除此之外,网络上的数据源质量参差不齐,有的数据集中包含的有效信息很少.如何及时过滤掉低质量的数据源,或者从海量数据源中找到所需要的信息,也是面向网络数据的数据集成需要解决的问题;
- 考虑到互联网所蕴藏的数据规模,如何高效地对海量数据进行集成,也是这类研究需要考虑的问题.

有些数据集成工作使用自动化算法很难实现,但如果由人来分辨就会很容易.正是针对这一特性,基于众包的数据集成技术被提出.众包技术的核心思路是:将计算机无法判断的任务分发出去,交由人来进行判断(人在

进行判断时一般会获得一些奖励),并将多人的判断结果进行整合,得出最可信的结论.可以看到,众包技术可以更加精确地发现数据集成问题中的数据关联关系.因此,使用众包技术作为解决数据集成问题的思路是一种非常热门的方法.

然而在使用这类技术时,仍然需要解决众包领域的一系列通用性问题.

- 首先是成本问题.一般的众包任务会对参与众包的人进行一定的奖励,这些奖励就是设计众包任务时的开销.在设计众包任务时,应当考虑如何用最小的开销得到最精确的结果.为了达到这一目标,需要选择最合适的问题以及最关键的数据作为众包任务.因此,如何妥善选择合适的众包任务,成为了这类问题需要解决的挑战之一.
- 另一个需要考虑的问题是众包任务的时间开销.与计算机的运算速度相比,将一个问题提交给众包再得到结果的时间开销是非常大的.而考虑到大数据时代对算法效率的追求,如何取得众包任务的时间开销与算法效率之间的有效平衡,也是基于众包的数据集成领域的重要挑战.

真正完成一个数据集成任务,除了数据集成算法本身以外,还需要考虑到数据输入、预处理、数据清洗以及后续的数据输出、可视化等一系列问题.除此之外,在有些时候,单一的数据集成算法可能不足以处理复杂的数据集成任务,因此需要引入多种数据集成技术用于处理不同的应用场景和数据集.针对这些场景,一些学者和企业开发了数据集成系统用于处理实际生活中碰到的数据集成问题.

与传统的数据集成方法或系统不同,数据集成系统往往是由多个数据集成方法或者相关的方法、技术集成起来的系统,这类系统一般都由一个或多个实际应用案例驱动.此外,介绍这类工作的文献在介绍所使用的数据集成技术的基础上,一般还会进一步介绍系统的具体应用场景、整体框架以及具体实现方法等.这一标准是在本文中区分数据集成系统与一般数据集成方法的标准之一.

2 技术综述

在本节中,我们针对数据集成技术的不同分类方向,介绍这些方向中较为具有代表性的文献与工作.为了让读者能够更加直观地理解数据集成技术发展脉络,我们将本文所设计的主要文献分类以及对应索引通过图 2 的形式展现出来.其中,每个小点代表一篇文献,而大的红点则代表综述性文献.不同的颜色分别对应了算法、网页表、众包和系统这 4 个分类的文献.

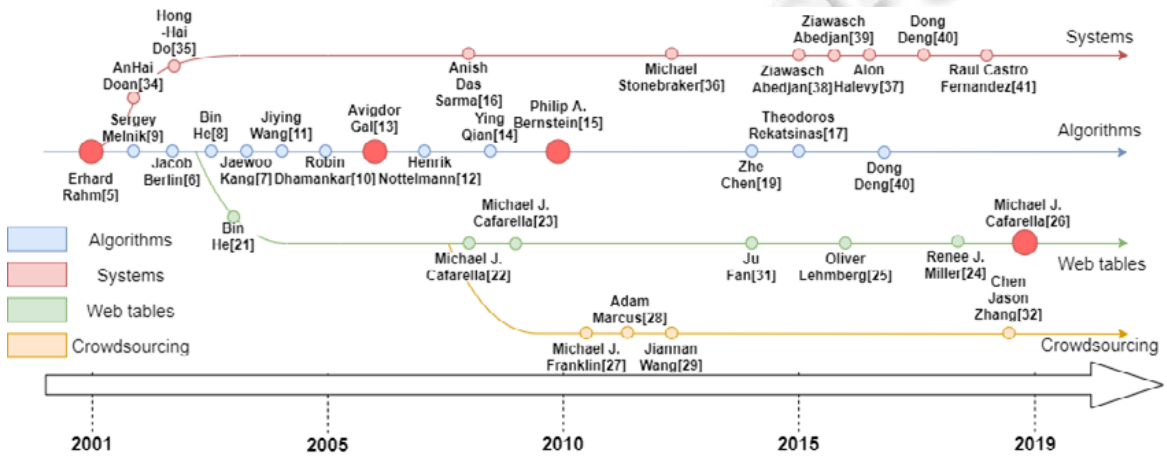

Fig.2 Categories and indices of related work

图 2 文献分类图与相关文献索引

2.1 早期数据集成技术

早期的数据集成领域研究主要集中在针对给定数据源以及数据集,如何识别出描述相同属性,相同实体的

数据表、数据列与数据元组之间关联起来.在关联关系的挖掘方面,主要采用的是较为简单、基于字符串进行直接匹配,通过人工识别等方式完成.相关的文献由 Erhard 等人[5]进行了较为全面的介绍与总结.在此基础上,后续的文献引入了在更复杂的问题定义与应用场景下解决数据集成问题的方法.Jacob 等人[6]基于贝叶斯模型对数据之间的关联关系进行评估,再将最优的概率模型作为最终的数据集成模式.Kang 等人[7]针对之前的数据集成方法需要依赖准确的数据名进行识别的问题,基于数据之间的互信息,提出了针对列名不明确或缺失场景下的数据集成方法.He 等人[8]针对在给定数据集范围内可能无法得出准确数据集成结果的问题,考虑到了进一步引入网络上的知识与数据用于协助数据集成工作.Sergey 等人[9]将原始数据模式映射成图模型,并提出了图匹配方法对图之间进行集成.在此基础上,本文还进一步引入了人工监督(human supervision)来进一步提高数据集成的准确度.Robin 等人[10]针对之前的数据集成方法主要解决一对一对应的问题,提出了更加复杂的数据对应关系(一对多、多对一,多对多),进一步增加了数据集成的灵活性;同时,通过对搜索空间进行切分来提升算法的效率.Wang 等人[11]针对传统的数据集成只服务于查询的功能,提出了双模式,包括界面模式(面向查询)与结果模式(面向检索).与传统方法相比,面向双模式的数据集成可以满足更加丰富的应用场景.Nottelmann 等人[12]提出了面向异构数据集成场景下的自动数据集成方法.Gal[13]对之前的数据集成工作进行了总结,并分析了数据集成领域后续所面临的挑战.

根据之前的总结我们可以发现:早期的数据集成工作的主要目标在于进一步提升数据集成的准确率,包括提出更加灵活的集成模式、引入人工监督等,而对数据源的规模、查询效率的讨论则相对比较有限.

2.2 后续集成算法发展

随着大数据时代的到来,数据集成研究所解决问题的重心开始逐渐发生改变.具体来说,数据集成算法从传统的注重准确率、基于限定数据模式开始向对算法效率和处理更加复杂的数据类型转变[14,15].其中较为有代表性的文献包括由 Sarma 等人[16]提出的基于 pay-as-you-go 框架的数据集成方法,该方法主要是为了解决传统数据集成算法普遍复杂度较高而导致的效率低下的问题,其核心思想是:先对数据进行快速集成,形成初步(但不一定准确或完善)的数据集成模式,随后再不断优化集成模式,随着时间的推移,逐渐演化成高质量的数据集成结果.

除了数据规模庞大以外,大数据带来的另一大挑战是数据质量问题.低质量的数据往往影响着算法的准确度,同时也影响了算法的搜索空间,提升了算法的时间开销.因此,有一批文献针对低质量数据的处理展开了研究.其中,Rekatsina 等人[17]通过快速对数据源的质量(quality)进行判别,并过滤掉低质量数据源,从而提升了数据集成算法的效率.Deng 等人[18]提出了 SILKMOTH 系统,该系统对数据中一定程度的不相似性(dissimilarity)鲁棒.该方法可以处理不同数据源中,相同实体的属性值存在一定误差导致传统方法无法识别的问题;同时,由于该算法初始复杂度较高,论文进一步引入了剪枝策略来降低算法的时间复杂度,从而提升算法效率.

此外,针对特殊数据类型的数据集成算法同样是具有前景的研究目标.例如,Chen 等人[19]针对报表数据(spreadsheet)的特征,提出了一种面向报表数据的元数据提取与数据集成方法.Chanial 等人[20]针对媒体环境下需及时集成来自不同数据源的信息以快速形成第一时间媒体报道的应用场景,提出了 CONNECTIONLENS 系统,该系统主要面向的是互联网上的文本等数据源的信息快速提取与集成场景.

2.3 面向网页表格的数据集成技术

在最初的设计中,数据集成技术主要解决的是给定数据源范围(例如某一企业内部)的数据集成.然而随着数据集成技术不断发展以及互联网、大数据环境对数据集成问题的冲击,人们很快发现,有时仅仅依靠企业内部数据不足以生成足够精确的数据集成结果.与此同时,通过将来自互联网的数据与企业数据知识相结合,可以生成更加精确的集成结果.因此,面向网页表格(webtable)数据的数据集成工作成为了一个热门的研究方向.该方向的工作主要目标在于通过将互联网上的数据进行进一步集成,发现有价值的潜在信息,或者将网络上的数据与本地数据结合起来,进一步提升数据集成的准确率等.同时,由于网络上数据规模十分庞大,如何提升面向网络规模数据的查询、集成效率也是一个重要的研究问题.

早期具有代表性的面向网页表格数据的相关工作包括 He 等人[21]提出的 MetaQuerier 项目,该项目基于互联网中潜藏的庞大数据量以及其中可能蕴藏的大量有价值信息,提出为用户开放可以对深网(deep Web)的查询与数据集成接口,从而帮助用户在互联网上更有效地查找自己所需要的信息.这篇文献属于早期认识到对互联网数据进行数据集成的重要性的工作之一.Cafarella 等人[22]则首次提出了网页表格的概念,因为其在对互联网数据进行提取时,首先从互联网上爬取了大量信息,再将爬取的信息中的高质量关系表提取出来进行进一步的集成,这一集成系统被称作 WEBTABLE.该系统的一大优势在于:通过集成互联网上具有结构信息的关系表,使得对互联网信息的集成与传统的数据集成方法可以有效地对接起来.这一定义一直被后续问题沿用,即后续的大量针对网页表格的集成问题所研究的主要对象也是网页中的表格数据.随后,同样是 Cafarella 等人[23]提出了 OCTOPUS 系统,在该系统中,作者详细描述了网页表格数据集成的挑战,包括数据源选择、数据清洗、元数据抽取等问题,并提出 OCTOPUS 系统用于解决这些潜在问题与挑战.

随着大数据时代对互联网数据的冲击,近期的面向网页表格数据的集成方法也面临着与研究大数据集成时类似的问题,所研究问题的发展趋势也与其他算法的发展趋势类似,即主要解决数据质量以及大规模数据处理的效率等问题[24].具有代表性的工作包括 Lehmberg 等人[25]提出的数据表格缝合(stitching)技术,在该工作中,作者发现:传统的面向网页表格的集成效果往往受到表格大小的影响,针对较小的数据表的集成效果往往比较差.因此在这篇文献中,作者通过将小的数据表缝合成大表的方法,解决了小数据表在数据集成中效果不佳的问题,从而进一步提升了数据集成的准确率.最后,Cafarella 等人[26]对网页表格技术 10 多年来的发展进行了总结.

2.4 基于众包的数据集成技术

在很多情况下,特别是针对特殊领域、特殊专业的数据,仅仅依靠有限的数据源信息可能不足以分辨出数据之间的关联关系,而互联网上可能也缺乏相关领域的数据,或者这些数据的知识模型难以提取,从而导致这些数据的集成工作无法仅仅通过自动化算法完成.此时,就需要引入人工监督来帮助识别这些数据.在早期的工作中,已经有方法将人工监督考虑到数据集成工作中[9].然而,系统性地将人工监督作为数据集成工作一环的仍然是将众包技术(crowdsourcing)与数据集成技术相结合的领域.与传统的考虑人工监督技术的应用场景相比,众包技术将引入“人”的开销、众包结果的准确率与代价估算等概念进行了更加全面与规范的定义,因此成为了需要引入人工监督时采用的主要技术标准.自然而然地,将众包技术与数据集成问题相结合的工作也应运而生.

Franklin 等人[27]首次提出了将众包技术与数据库相融合的系统 CrowdDB,该系统通过加入众包方法来回答通过数据库与搜索引擎都无法回答的查询.在技术上,该系统解决了传统的数据库模型是闭合的(close-world)、没有考虑到将“人”的知识作为输入条件的问题.针对这一问题,本文具体讨论了在数据库中引入人工知识作为接口时需要解决的问题,并描述了 CrowdDB 系统的具体实现.由于 CrowdDB 系统的思路是让人在参与回答查询时反馈一定的金钱作为奖励,因此如何合理规划问题的设计、选择合适的问题以及回答人和判断结果的准确度等问题,都需要进一步分析.针对这些问题,Marcus 等人[28]提出了 Qurk 系统,该系统可以对 CrowdDB 的流程进行进一步优化,从而实现用最小的开销满足尽可能多的查询.

类似的问题,诸如问题设计、评估答题质量、结果与数据模型结合的问题,不仅出现在 CrowdDB 中,也是整个众包技术所面临的共通性问题.因此,后续基于众包技术的数据集成技术有大量文献都在集中解决这些问题.Wang 等人[29]认为,完全依赖众包技术无法解决对大规模数据的集成问题,因此其提出了一种混合系统,该系统首先对数据进行初步处理,随后过滤出部分数据提交给众包.与传统方法相比,该技术在效率上与准确率上具有更大优势.Whang 等人[30]与 Fan[31]针对众包环境下对问题的选择进行了进一步研究,他们都考虑到了众包模型下的代价问题,因此需要选择最有价值的答案来得到最终结果.Whang 等人的工作主要针对的是选择未知的实体(entity)进行识别,而 Fan 等人的工作则主要面向的是选择表格中的不确定数据列(column).Zhang 等人[32,33]考虑到众包平台中,询问关系是否正确这样的二元问题(correspondence correctness question)效果最好,提出了针对这类问题的众包问题选择策略.在此基础上,该工作进一步考虑到了众包场景下结果可能不准确导致的不确定性问题,提出了基于信息熵的不确定性缩减方法,并为结果的准确性提出上下界,进一步提升了结果的可信度.

2.5 数据集成系统发展

在早期的数据集成算法被提出以后,很快人们就发现,单一的算法往往不能适用于全部的数据集成问题场景.在有些应用场景下,使用基于多重数据集成算法的混合模型在处理数据集成问题时会达到更好的效果.基于这一发现,Doan 等人[34]提出了 GLUE 系统,该系统可以允许使用多种相似度度量来挖掘数据之间的关联关系.与此同时,Do 等人[35]也提出了 COMA 系统,该系统集成了多个传统的数据集成方法,通过混合模型进一步提升数据集成的准确率.这两个系统也可以被看作早期的数据集成平台的代表.

随着数据集成技术的进一步发展,逐渐涌现出企业级规模的数据集成平台用于对企业数据进行管理、搜索与分析等工作.其中具有代表性的成果包括 Stonebraker 等人[36]提出的 DataTamer 系统,该系统主要任务是快速过滤,找到用户感兴趣的数据集.在实现技术上,该系统集成了之前的一系列技术亮点,包括人工监督、模型随时间演化等技术.在此基础上,该系统还引入了数据的可视化功能,该功能使得用户能更加方便快捷地获取所需要的信息.

另一个具有代表性的系统是 Google 研发的 GOODS[37]系统,该系统的设计、研发初衷是为了能够更加有效地利用 Google 企业内部的数据,通过将多数据源异构数据有效集成,进一步挖掘企业数据的价值.在实现上,该系统充分发挥了Google在搜索引擎方面的技术积累,GOODS系统可以在企业级数据集上进行类似于搜索引擎的搜索方法,并为用户反馈最合适的搜索结果.

除此之外,也不断有新的数据集成平台概念与技术被提出.Abedjan 等人[38,39]提出了 DataXFormer 系统.在数据集成工作中,由于异构数据源带来的原始数据结构、模式不一致问题导致大量的时间需要花费在数据结构的统一化过程中.为了解决这一问题,DataXFormer 系统通过对数据源进行分析,为用户推荐可行的数据变化策略,从而节省在统一数据模式上的时间开销.Deng 等人[40]提出了 DATA CIVILIZER 系统用于对企业内部的大量数据进行集成,该系统使用一个动态更新的关联图(linkage graph)模型来表示企业内部不同数据之间的信息,同时使用数据发现(data discovery)模组从关联图中高效地查询与用户任务相关的数据.类似的基于图模型对数据关联关系进行建模的技术也被应用于 Aurum 系统[41],该系统针对传统的数据集成系统只能解决某些特定场景下的数据集成工作这一局限性,提出了更加灵活的数据集成技术,通过将企业数据建模成企业知识图(enterprise knowledge graph),并将数据集成与数据发现问题转换成在知识图上的查询问题,使得该平台能够支持更加灵活的数据集成与知识挖掘任务.

3 代表性工作介绍

在本节中,我们针对不同分类中的技术和文献,选择其中具有代表性的工作进行详细说明,从而进一步加深读者对相关领域的理解.基于第 2 节中对论文的分类,我们挑选出在相关分类方向上具有代表性的 5 篇文献,对其研究动机、研究问题以及解决问题所使用的方法进行较为详细的介绍,从而进一步加深读者对相关领域研究内容的认识.我们之所以将这些文献选择出来作为代表性文献,首先是因为这些文献一般在相关研究领域中,发表在的期刊或会议水平比较高,或者是因为这些文献的引用次数较多;其次,这些文献所研究的问题往往是相关领域具有较强代表性的问题,类似的问题是该方向很多工作中都会碰到的问题,因此解决这些问题的意义是非常重大的;最后,这些文献在解决问题时,所用的方法一般都具有较强的普适性和巧妙性,能够将问题化繁为简,为读者在后续推进自身研究时能够起到一定的参考和指导作用.

(1) 面向模糊列名与数据取值的数据集成方法[7]

文献[7]解决的是针对数据集成中,存在数据列名与数据取值模糊而导致传统数据集成技术无法有效识别数据之间关联关系的问题.我们通过一个简单的例子来说明该文的问题与贡献.

在简单的数据集成场景中,数据集的列名与其中元素的含义都是十分明确的,如图 1 中所示,我们需要做的工作是将代表相同含义的数据列建立映射(表 1 作者-表 2 名字)以及具有相同含义的数值建立映射(表 1 雨果-表 2 雨果).这些映射关系可以通过分析数据语义,或者计算编辑距离实现.然而在一些数据库和数据表中,可能使用特殊的编码或者抽象的字符对数据进行表示.例如在书库中,有些表内可能会用书籍的编码而非书名来标

识书籍.这种情况下,传统的通过语义信息来识别同一含义或实体的方法不再有效.因此,该文解决的是针对模糊数据集之间的映射问题.具体来说,给定如图 3 所示的原始数据,其中相同编号的实体代表的含义相同(例如所有的 c1 都指代相同数值),不同表之间可能存在映射关系(例如左表的 A 列与右表的 Z 列可能代表相同含义),该文所研究的就是如何对给定的若干个这样的原始数据进行集成的问题.

A	B	C	D
a1	b2	c1	d1
a3	b4	c2	d2
a1	b1	c1	d2
a4	b3	c2	d3

W	X	T	Z
w2	x1	y1	z2
w4	x2	y3	z3
w3	x3	y3	z1
w1	x2	y1	z2

Fig.3 Example of opaque data sources

图 3 原始模糊数据实例

该文采用的是基于信息熵与互信息的方法,其中:信息熵衡量了每个数据列所包含的信息量,而互信息则衡量不同数据列之间的信息量差异.如果两个数据列所包含的信息量类似,即信息熵类似,则两个数据列代表同一含义的可能性更大.类似地,如果两列数据之间的信息量差异在两个表中类似,则两个数据列分别对应的可能性更大.

直观来说,图 3 左表的数据列 A 可能与右表的数据列 X 代表的含义相同,因为这两列的数据分部较为类似,都包含 3 个不同的数据值(a1,a3,a4-x1,x2,x3),且有一个数据值重复出现了一次(a1,x2).这样的抽象数据分部可以通过信息熵进行度量.例如,我们可以通过计算信息熵得出 A 与 X 两个数据列的信息熵都为 1.5,从而推断 A 与 X 大概率是相互对应的.

然而,仅仅依靠信息熵来进行数据集匹配只能处理当两个数据源的数据列数量一致,且必定存在两两属性对应的情况.在更多情况下,我们不能确定两个数据源是一定两两对应的.例如图 1 中的表 1 与表 2 之间,只有人名字是对应的,而其他的数据列则并不存在对应关系.当需要挖掘这类数据之间的匹配关系时,仅仅依靠信息熵是无法有效识别的.为了解决这一问题,进一步提出了支持图(dependency graph)模型.支持图为一个加权图,图中的节点代表数据源中的数据列,节点权重代表该数据列的信息熵.而边则代表两个节点之间的互信息.互信息反映了两个数据属性之间的支持度(dependency).例如:如果两个变量是完全独立分布,则两者之间的互信息应当为 0.在本文的支持图模型中,引入互信息可以进一步帮助我们识别哪些数据列可能代表相同的信息.例如,图 4 中展示了基于图 3 数据所衍生出的支持图模型.从中我们可以看到:属性 A-B 之间的互信息与 W-X 的互信息均为 1.5,而 A 的信息熵与 X 相同,B 的信息熵也与 W 相同.由此我们可以大概率推测出 A,B 两个属性分别与 X,W 两个属性对应.

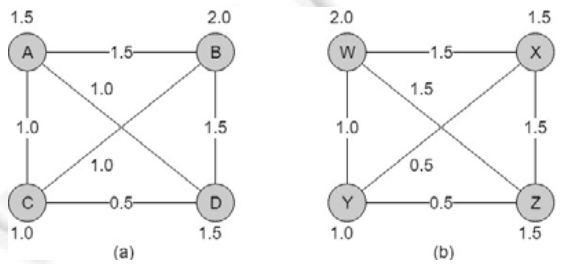

Fig.4 Dependency graph model derived from example data sources

图 4 由图 2 数据衍生出的支持图模型

通过使用信息熵和互信息两个度量,有效解决了在数据缺乏语义信息或者数据被加密后,不同数据源的数据融合问题.该方法是早期数据集成领域中普适性较高的一个算法.

(2) 对数据源进行快速筛选的方法[18]

尽管之前的很多方法中(例如上文提到的基于信息熵与互信息的方法)已经可以有效识别任意两个给定数据源之间可能存在的关联关系,但这些方法仍然不能很好地处理大数据环境下的数据集成问题.其中一个重要的原因在于:由于需要进行集成的数据源候选集非常庞大,而传统方法的计算复杂度又较高,因此,如果对候选集中全部数据源的关联关系进行两两计算的话,整体时间开销和空间开销是非常高的.因此,为了解决这一问题,本文提出了 SilkMoth 系统,旨在快速过滤掉那些关联关系不高的数据源.与简单的数据过滤策略不同,本文的数据源筛选方法可以保证筛选出的数据集的准确性,即:相似的数据集一定会出现在过滤出的候选集中,不会出现两个数据集本身相似,但算法没有识别的情况.

识别数据源之间的关联关系的通用方法之一是 Bipartite Matching[42]算法,该算法将两个给定数据集构造成一张二分图,每一个数据集分别代表图其中的一个子图,子图中的点代表集合中的元素,可以是关键词、关键词组等.而跨子图的边则代表两个元素之间的相似性,可以通过编辑距离(edit distance)等度量来计算.Bipartite Matching 算法的核心是找到一批跨子图的边的集合,使得集合中边的权重和最大.该算法的优点在于其较强的鲁棒性,可以有效识别集合中元素不完全一致的情况.而其缺点在于需要计算两个集合中,所有元素之间的相似性.如果给定多个集合,则需要计算任意两个集合之间、任意元素之间的相似性.这个过程的时间复杂度是非常大的.针对这一问题,本文的主要贡献在于如何减少针对海量数据集集成过程中需要进行 Bipartite Matching 计算的候选集,从而压缩算法的时间开销.

文献[18]的方法主要通过以下两个步骤.

- 首先提出了一种 Signature 选取策略.Signature 可以被看作集合中某些关键词的集合,代表了该集合的特征.直观上来说,如果两个集合的 Signature 相似或相同,则有很大概率两个集合具有较高的相似性;而如果两个集合的 Signature 差异性很大,则可以认为两个集合没有相似性,无需再进行相似性的精确计算.因此,该文的第 1 个贡献在于如何选择出高质量的 Signature 来有效反映集合的特征.值得注意的是:所提出的 Signature 选取策略有非常高的求全率,可以保证不会出现 False Negative 的情况,即,没有识别出相似数据集的情况.然而为了保证高求全率,只通过 Signature 识别出的相似候选集中仍然存在大量相似度较低的数据集,因此在第 2 步工作中,主要通过两次过滤来进一步筛选需要过滤的候选集;
- 在第 2 步中采取了最近邻过滤(nearest neighbor filter)策略.最近邻过滤策略的原理是:基于样本中的近邻关系,发现任意两个相邻样本所能达到的最大相似度,这些相似度的累加即是两个集合之间相似度所能到达的相似度上限.因此,如果该相似度上限仍然低于用户给出的阈值,则可以将这两个集合的关系从候选集中剔除.由于最近邻计算算法本身复杂度较高,本文在文中继续讨论了若干可以对此处最近邻计算进行进一步加速的策略.

从以上总结可以看出:在这篇论文中,作者主要的工作是针对历史中已经存在的效果很好但时间复杂度较高的算法,对其进行尽可能的加速与优化,从而使得经典算法能够被应用于处理海量大数据的场景.这也是大数据时代下,从事数据集成方向研究的主流思路之一.

(3) 面向网页表数据的集成方法[25]

大数据时代带来的一大改变是互联网应用的普及,随之而来的是互联网上,特别是网页中包含的数据量提升.如果能将这些数据通过数据集成方法进行处理,很可能发现更多潜藏在网络上的信息.然而,传统的数据集成方法主要处理的都是数据库中的数据表,这些数据表往往都遵循某种设计模式,而且每个表都比较饱满,其中包含的数据都比较多.相比之下,存在于网页中的数据表往往都是一些比较小、比较离散的表.例如,介绍一个电影院近期电影的网页表中往往只会包含近期几个上映电影的信息,而不会把历史中的全部电影信息都存在表中.而考虑到传统的数据集成问题所处理的往往是含有丰富属性的大表,针对网页中存在的这些小表的集成问题是面向网页表示数据集成方向的一大重要挑战.

考虑到直接对小的网页表进行集成不够理想,在本文中,作者提出了一种将网页表进行缝合(stitching)的方法,将多个小表缝合成一张大表,再用于后续的数据集成等分析工作中.本文首先用了 3 种直观的方法对网页表

进行缝合,并通过实验验证对网页表进行缝合确实可以有效提高数据集成的准确度.接下来,作者针对直观方法无法解决的网页表缝合问题,进一步提出了优化策略.作者所采用的直观缝合方法包括 3 种:基于列名的方法、基于数值类型的方法与基于重复的方法.其中:基于列名的方法是通过直接匹配两个表中属性的名字,将具有相同名字的属性列看作具有相同含义的属性列进行缝合;基于数值类型的方法是通过判断两个数据列中所储存的数据类型是否一致(例如 Int-Int)来判断属性之间的匹配关系;基于重复的方法是直接判断列中所包含的数据数值是否相同(例如雨果-雨果).然而,只通过这些方法不足以解决网页表的缝合问题.

以图 5 为例,当对图 5 中的两张表进行缝合时,直观方法不能很好地识别两张表中数据的对应关系,其原因在于:首先,两张表的属性名所使用的语言不同,因此,尽管两张表的第 1 列数据都代表名字(name),由于语言不通,算法无法识别出这两列数据属性所指代的含义是相同的;其次,以最后一列第 1 行数据为例,尽管两个表中的数据数值均为 0.99,但两张表中的单位是不一样的,前表中是“磅”而后表中是“欧元”,这一差异也让我们在设计算法识别数据的对应情况时存在困难;最后,两张表的列数量是不同的,此时,我们需要判断哪些数据列是存在对应关系的,哪些是可以被删去的.

Fig.5 Example of Web table data

图 5 面向网页表的数据集成示例

针对上文例子中存在的、直观方法无法识别的网页表缝合问题,本文进一步提出了混合匹配算法(hybrid matcher).该算法的主要功能是针对直观方法可能错误识别的网页表匹配模式进行检查,并发现潜在的匹配模式,并排除掉那些可能错误的匹配模式.具体来说,本文首先将直观方法识别出的匹配模式构建成图模型.例如,如果通过直观方法得出以下属性对应关系:artist-artista,artista-interpret,naam-titulo,titulo-name,我们则可以构建出图 6 中由节点和虚线边组成的初始图.接下来,我们可以首先发现一些之前没有识别出的匹配关系.例如,由于 artista 与 artist 和 interpret 都存在匹配关系,那么可以推断出 artist 与 interpret 也存在匹配关系,同理也可以识别出 naam 与 name 之间存在匹配关系.接下来,作者计算图中任意两个节点之间是否存在通路,以及通路的路径所走过的边.如果一条边被走过太多次,那么这条边很可能代表了一个错误的匹配关系.

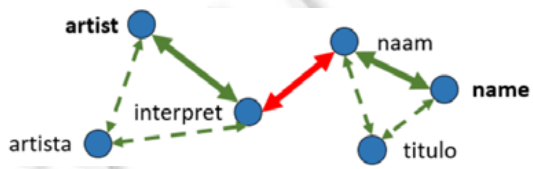

Fig.6 A graph model generated using hybrid matcher

图 6 混合匹配算法示例

该方法基于的假设是:如果该匹配图正确,那么图中的点只会与具有匹配关系的点产生边的链接.例如{artist,artista,interpret}集合与{naam,name,titulo}集合,这些集合中,任意两个节点之间的路径都较短,而且走过的边各不相同,并且两个集合之间则不应该存在通路.反之,如果图中存在一个错误的匹配关系 interpret-naam,那么原本两个不连通的集合就通过这条错误的边被连同起来,此时,当我们计算两个集合之间任意两个节点的

通路时,这些通路都会经过interpret-naam这条边.因此,可以通过追踪那些被频繁走过的边来识别哪些边所代表的匹配关系是错误的.

该文主要是针对网页表中数据的融合问题,论证了将网页表进行缝合的有效性与必要性;并针对网页表缝合问题,提出了一种较为鲁棒的网页表缝合策略.本文所提出的将网页表缝合的方法也是处理网页表集成时,能够有效提高网页表数据融合准确度的方法之一.

(4) 基于众包的数据集成方法[31]

随着数据集成方法所需要处理的数据集来源越来越广泛,很可能出现一些数据集的知识过于抽象或者过于复杂,导致传统的机器学习方法很难有效识别表中的数据含义,从而造成数据集成结果的不准确.针对这一问题,一种可行的解决思路是通过众包策略,通过人工的方法来帮助识别那些有歧义的数据含义.然而在众包问题中,每一个问题都伴随着一定开销的产生,而在一个众包项目中,提供众包项目的研究者的总预算往往是有限的.因此,如何在有限的预算内尽可能选择最有意义、对提升准确度帮助最大的问题,是众包场景下需要解决的主要问题之一.

在这篇文章中,作者所研究的就是针对网页表数据集成的背景下,将哪些不清晰的数据匹配关系提交给众包进行判断,从而能够最大地提升数据融合准确率的问题.值得注意的是:尽管该文所研究的是针对网页表数据的集成问题,但所提出的解决问题的方法是具有通用性的,可以广泛应用于各种数据融合问题场景中.

该文解决数据融合问题的核心思想是:

首先,借助网上的知识库(例如 FREEBASE[43])对网页表中的数据进行初步识别,对于那些识别出来结果存在较大歧义的数据属性,选择对结果准确度提升贡献最大的属性提交给众包进行分析处理.在对数据进行初步识别时,该文采用了基于信息熵的方法.该方法的直观思路是:针对输入数据集中的任意一个属性,计算它与FREEBASE 数据集中各个概念(concept)的相关度(correlation),最后再计算这一批相关度的信息熵.如果识别结果对该属性的歧义比较低,那么该属性应当只会对一个或者少数几个相关联的概念相关度非常高,而对其他概念的相关度很低.基于这样的数据分部所计算出的信息熵应当是较低的.然而,如果该属性的歧义比较高,则势必会导致该属性在多个概念上的相关度均有较高的数值,这就会导致信息熵的升高.因此,可以通过计算每个属性在多个概念上相关度的信息熵,来判断哪些属性是可以被有效识别,而哪些属性是仍然具备较大歧义的.

通过上一步的处理,已经可以过滤出一批歧义比较大的数据属性.接下来需要解决的问题是选择哪些属性提交给众包任务进行进一步识别.考虑到众包任务的开销,我们希望选择一些具有更大价值的数据属性来提交给众包任务进行识别.例如通过识别某一属性,可以同时帮我们确定多个数据属性的含义.基于这一思路,本文通过两个指标来判各个属性的价值.

- 首先是表内影响力(intra-table influence).

直观来说,该影响力反映了一个属性与对其他属性的影响力强弱.例如,以图1中表1的数据为例,假设我们知道了第2列的作者与第1列的作品之间存在创作与被创作的关系,那么如果我们确定了第2列数据代表的作者是“书籍作者”,我们同时也可以确定第1列的名字代表的是“书籍名字”;反之,如果第2列数据代表的是“电影作者”,那么第1列数据代表的则一定是“电影名字”.因此,我们可以通过一次众包任务(识别第2列属性含义)确定两列数据的属性.这一关系就代表着第 2 列数据具有较大的“表内影响力”,因为对其含义的确定也影响了对其他数据列的含义识别.

- 文中采取的第2个判断指标是表间影响力(inter-table influence).

在两张表中可能存在一些数据属性,我们并不知道它们的准确含义,但是我们知道它们代表的是相同的意思.这一关系就代表这两个属性之间具备较强的表间影响力.例如,考虑图1中表1的“作者”属性与表2的“名字”属性,尽管我们不知道它们代表的是电影作者还是书籍作者亦或是其他作者,但我们知道它们所代表的含义是大概率相同的.在本文的方法中,考虑到网页表一般都比较小,包含的数据可能不够全面,因此作者并没有选择基于表内数值的方法来计算属性之间的表间影响力.相对的,作者再次利用了该属性在FREEBASE上的相关度.直观来说,如果两个属性所代表的含义相同,那么它们在FREEBASE上的多个概念上的相关度分部也是类似的.

基于这一思路,本文提出了概念向量(concept vector)用来描述一个属性在不同概念上的相关度分部.以图 7 为例,左边的 3 个概念向量分别对应着右边 3 个属性(表 1 的 Title、表 2 的 Movie、表 3 的 Title)的概念向量.接下来,可以通过计算概念向量之间的 Cosine 相似度,从而发现前两个概念向量的相似度更高,因此可以认为表 1 的 Title 属性和表 2 的 Movie 属性之间具有较强的表间影响力.

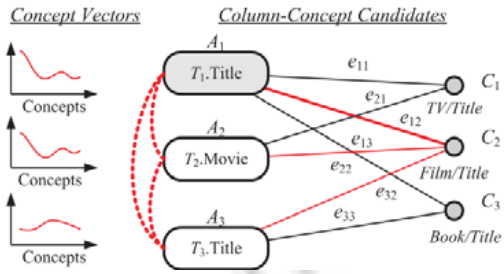

Fig.7 Example of concept vectors

图 7 概念向量示意图

在计算出了每个属性的表内影响力与表间影响力以后,可以通过对两个影响力进行加权求和,计算出综合影响力(integrated influence),综合影响力将作为判断是否选择该属性作为众包任务提交的标准.在选择众包任务时,将优先选择综合影响力更大的属性.

(5) 大数据集成系统[37]

在上文中我们已经介绍了很多数据集成的算法.然而在真正使用中,我们需要将这些方法融合到一个系统框架中,才能发挥其作用.因此,在本文中,我们选择 GOODS,一款 Google 开发的面向其企业内部数据的管理系统(图 8).这篇文章主要介绍了在管理海量企业数据时所面临的挑战,以及 Google 所采取的解决方案.

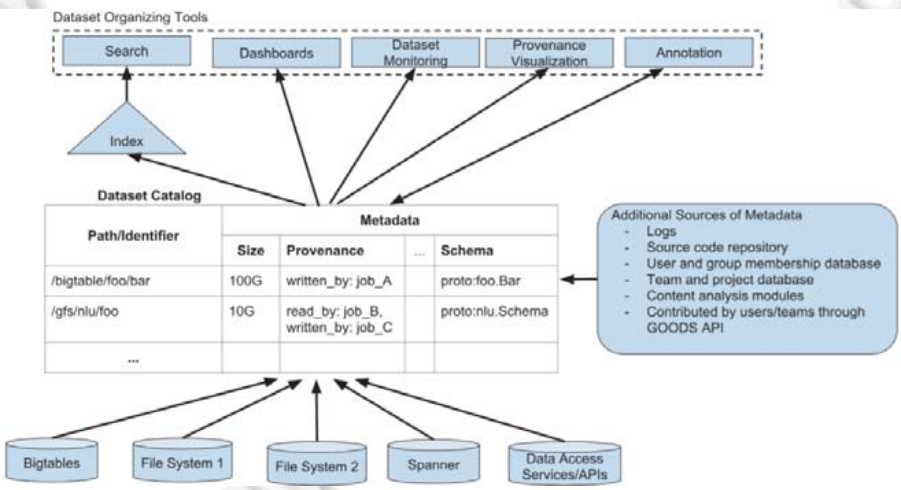

Fig.8 Architecture of GOODS system

图 8 GOODS 系统架构

文献[22]主要介绍了当前的大数据企业在企业数据量增加、数据模式增多的环境下所面临的数据集成以及数据分析的问题.目前,企业为了能够快速发展,往往会不断启用新的数据库、新的编译平台用于快速实现短期项目成果.然而,这一做法导致了企业中的很多数据存在于不同数据库、不同数据平台中.由于跨平台的异构性导致企业很难将这些跨平台数据统一应用起来.因此,该文提出一种标准化的数据管理系统,能够将来自企业不同引擎的数据集成在一起.

图 8 是一个数据应用场景.图的最底层是 Bigtable,File System 等不同数据源.通过将来自不同数据源的数据通过在中间进行整合以及统一建模,使得来自不同数据源的数据能够被标准化.将这些数据标准化以后,接下来就可以用于数据查询、数据分析、绘制报表、可视化等一系列应用场景中.其中,中间的数据集目录(dataset catalog)是该系统的核心组件,该目录中存储了从底层不同引擎、不同数据集中所提取的各种元数据信息,用于执行上层的数据检索、分析、报表等功能.在组织元数据的过程中,主要需要解决以下六大挑战.

- 数据集规模.以 Google 数据为例,目前 Google 的 Catalog 索引数量已经超过了 260 亿条数据.将这些数据的元数据全部收集并存储下来是不可能做到的.因此需要提出一种新的方法,能够在收集数据的同时,通过采样的思路用部分元数据代表其他数据;
- 数据多样性.数据的多样性主要体现在数据本身的格式多样性以及存储平台多样性上.格式多样性包括数据可能以 txt,csv 等形式存在,而平台的多样性则体现在数据可能存在文件系统或者数据库等存储平台中.如何将来自不同平台的数据集成起来是一大挑战;
- 数据更新.企业所产生的这些数据并不是静态的,每天都有新的数据生成与新的数据被删除.如何控制这些数据的元数据信息与实际数据之间的同步状态是很重要的:如果同步效率太高,可能造成太高的代价;如果同步效率太低,则会导致数据可用性变差;
- 数据集重要性.数据集重要性指这些数据集信息的真实性、可用性以及同步性等特性对用户使用的影响.有些数据集是十分重要的(类似于数据库系统表),这些数据应当被及时更新;而有的数据则重要性较低,允许降低其可用性;
- 数据不确定性.这里主要指元数据的不确定性.在企业大数据环境下,不可避免地存在一些数据的元数据是难以解析的,如何处理这些不确定的元数据,是该数据管理平台需要解决的问题;
- 数据集语义信息.有些数据集的语义信息十分重要,通过提取数据集中的重要语义信息,不仅可以更好地帮助用户对数据集进行检索与筛选,在管理数据时,也可以基于数据之间关联关系对数据存储策略进行优化.

数据集成方法在解决这些挑战的过程中扮演着重要的角色,例如:针对数据的不确定性信息,我们可以通过上文介绍的基于众包或是基于数据缝合的方法来更加有效地识别数据集中的不确定性数据;针对数据集的语义信息提取功能,可以使用对属性价值进行判断的方法,选择那些具有较高影响力的属性进行提取.在管理大规模数据时,可以通过数据集成功能,将具有相同或相似功能的数据源集成起来统一管理,用少量的元数据信息描述这一批数据,达到压缩元信息存储的目的.在本文中,由于篇幅所限,作者并没有详细介绍解决这些挑战所使用的具体方法细节,但是其提出的许多解决问题思路是与数据集成方法紧密相关的.由此可见:数据集成问题不仅可以应用于数据集成领域本身,也可以在大数据管理、数据分析等领域产生重要价值.

4 总结与展望

在本文中,我们梳理了数据集成领域从 2001 年开始到现在的发展脉络.通过梳理脉络可以发现,早期的数据集成文献主要解决的是在给定数据集上进行数据集成与数据融合的问题.随着大数据时代的到来,数据集成方向的研究开始往多个方向展开,一个方向是针对海量数据处理场景下的数据算法加速研究,包括对数据集进行采样、对候选集进行压缩等;第 2 个方向是针对互联网的迅速发展,对网页表数据的数据融合、集成与知识发现研究;第 3 个方向是针对数据源越来越复杂,传统方法难以识别的问题,使用基于众包的方法,借助人的力量对数据集进行集成与融合.最后,随着时代发展,数据集成平台、或者是与数据集成技术紧密相关的数据管理、分析平台的发展也是十分具有前景的研究领域.

对于未来的数据集成领域研究,我们认为,目前比较有潜力的方向主要集中在对算法加速、对复杂数据源的集成以及基于众包的方法.

- 在算法加速方面,随着目前硬件设备计算能力的快速升级以及分步式计算框架的逐渐成熟,针对分步式环境下的数据集成方法研究是十分有潜力的一个方向,具体的研究内容包括将历史的经典集成算

法在分步式框架下进行实现,亦或是设计全新的面向分步式系统的算法.这一方向的优势在于:通过最大化利用分步式计算的效能,可以有效解决传统算法中算法的准确率与效率无法同时满足的问题;

- 第 2 个比较有潜力的方向是面向复杂数据源的数据集成问题,该问题可以被看作是基于网页表的数据集成问题的扩展.在互联网中,除了网页表以外,还有各种各样的有意义的数据源,包括图像、视频等.目前,在相关领域中,已经存在一些研究工作着眼于如何从多媒体信息中抽取出有意义的标签信息,那么基于这些抽取的标签信息的数据集成研究可以有效服务于媒体识别、网络知识库构建等一系列研究方向.因此,对包括网页表在内的复杂数据源的数据集成问题也是十分有潜力的研究方向;
- 最后,我们认为基于众包的数据集成问题同样会是具有持续热度的一个研究方向.我们认为,无论数据集成技术发展到什么程度,都难以避免由于知识库缺失,或者集成可信度不足导致的无法精确识别数据实体之间关联关系的问题.这种问题最终都需要借助“人”的知识来解决.而众包作为目前最为广泛使用的利用“人”的知识服务于研究领域的技术,将会在数据集成这一领域持续性的扮演举足轻重的作用.

值得说明的是,以上的总结与展望都是我们通过分析、梳理与研究目前的数据集成技术的发展脉络而总结出来的.这些内容包含一定的主观判断因素,读者可以将这些内容作为参考,但不必当作客观规律来遵循.不仅如此,除了我们所总结出的这些方向以外,在未来很有可能有崭新的研究问题与研究方法被提出.总体来说,对数据集成领域的研究工作,不仅会在未来具有相当高的价值,这些研究所贡献的方法也将在其他研究领域发挥重要的作用.

References:

[1] Du XY, Lu W, Zhang F. History, present, and future of big data management systems. Ruan Jian Xue Bao/Journal of Software, 2019,30(1):127−141 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5644.htm [doi: 10.13328/j.cnki.jos.005644]

[2] Meng XF, Du ZJ. Research on the big data fusion: Issues and challenges. Journal of Computer Research and Development, 2016, 53(2):231−246 (in Chinese with English abstract).

[3] Chen YG, Wang JC. A review of data integration. Computer Science, 2004,31(5):48−51 (in Chinese with English abstract).

[4] Yang XD, Peng ZY, Liu JQ, *et al*. An overview of information integration. Computer Science, 2006,33(7):55−59 (in Chinese with English abstract).

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

[6] Berlin J, Motro A. Database schema matching using machine learning with feature selection. In: Proc. of the Int’l Conf. on Advanced Information Systems Engineering. Berlin, Heidelberg: Springer-Verlag, 2002. 452−466.

[7] Kang J, Naughton JF. On schema matching with opaque column names and data values. In: Proc. of the 2003 ACM SIGMOD Int’l Conf. on Management of Data. ACM, 2003. 205−216.

[8] He B, Chang KCC. Statistical schema matching across Web query interfaces. In: Proc. of the 2003 ACM SIGMOD Int’l Conf. on Management of Data. ACM, 2003. 217−228.

[9] 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.

[10] Dhamankar R, Lee Y, Doan AH, *et al*. iMAP: Discovering complex semantic matches between database schemas. In: Proc. of the 2004 ACM SIGMOD Int’l Conf. on Management of Data. ACM, 2004. 383−394.

[11] Wang J, Wen JR, Lochovsky F, *et al*. Instance-based schema matching for Web databases by domain-specific query probing. Proc. of the VLDB Endowment, 2004. 408−419.

[12] Nottelmann H, Straccia U. Information retrieval and machine learning for probabilistic schema matching. Information Processing & Management, 2007,43(3):552−576.

[13] Gal A. Why is schema matching tough and what can we do about it? SIGMOD Record, 2006,35(4):2–5.

[14] Ying Q, Li Y, Song J, *et al*. Discovering complex semantic matches between database schemas. Computer Engineering & Science, 2008,29(10):61−63.

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

[16] Das Sarma A, Dong X, Halevy A. Bootstrapping pay-as-you-go data integration systems. In: Proc. of the 2008 ACM SIGMOD Int'l Conf. on Management of Data. ACM, 2008. 861−874.

[17] Rekatsinas T, Dong XL, Getoor L, *et al*. Finding quality in quantity: The challenge of discovering valuable sources for integration. In: Proc. of the Biennial Conf. on Innovative Data Systems Research. 2015.

[18] Deng D, Kim A, Madden S, *et al*. Silk Moth: An efficient method for finding related sets with maximum matching constraints. Proc. of the VLDB Endowment, 2017,10(10):1082−1093.

[19] Chen Z, Cafarella M. Integrating spreadsheet data via accurate and low-effort extraction. In: Proc. of the 20th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2014. 1126−1135.

[20] Chanial C, Dziri R, Galhardas H, *et al*. Connectionlens: Finding connections across heterogeneous data sources. Proc. of the VLDB Endowment, 2018,11(12):2030−2033.

[21] He B, Zhang Z, Chang KCC. Knocking the door to the deep Web: Integrating Web query interfaces. In: Proc. of the 2004 ACM SIGMOD Int'l Conf. on Management of Data. ACM, 2004. 913−914.

[22] Cafarella MJ, Halevy A, Wang DZ, *et al*. Webtables: Exploring the power of tables on the Web. Proc. of the VLDB Endowment, 2008,1(1):538−549.

[23] Cafarella MJ, Halevy A, Khoussainova N. Data integration for the relational Web. Proc. of the VLDB Endowment, 2009,2(1): 1090−1101.

[24] Miller RJ. Open data integration. Proc. of the VLDB Endowment, 2018,11(12):2130−2139.

[25] Lehmberg O, Bizer C. Stitching Web tables for improving matching quality. Proc. of the VLDB Endowment, 2017,10(11): 1502−1513.

[26] Cafarella M, Halevy A, Lee H, *et al*. Ten years of webtables. Proc. of the VLDB Endowment, 2018,11(12):2140−2149.

[27] Franklin MJ, Kossmann D, Kraska T, *et al*. CrowdDB: Answering queries with crowdsourcing. In: Proc. of the 2011 ACM SIGMOD Int'l Conf. on Management of Data. ACM, 2011. 61−72.

[28] Marcus A, Wu E, Karger DR, *et al*. Crowdsourced databases: Query processing with people. In: Proc. of the Biennial Conf. on Innovative Data Systems Research. 2011.

[29] Wang J, Kraska T, Franklin MJ, *et al*. Crowder: Crowdsourcing entity resolution. Proc. of the VLDB Endowment, 2012,5(11): 1483−1494.

[30] Whang SE, Lofgren P, Garcia-Molina H. Question selection for crowd entity resolution. Proc. of the VLDB Endowment, 2013,6(6): 349−360.

[31] Fan J, Lu M, Ooi BC, *et al*. A hybrid machine-crowdsourcing system for matching Web tables. In: Proc. of the 2014 IEEE 30th Int'l Conf. on Data Engineering. IEEE, 2014. 976−987.

[32] Zhang CJ, Chen L, Jagadish HV, *et al*. Reducing uncertainty of schema matching via crowdsourcing. Proc. of the VLDB Endowment, 2013,6(9):757−768.

[33] Zhang C, Chen L, Jagadish HV, *et al*. Reducing uncertainty of schema matching via crowdsourcing with accuracy rates. IEEE Trans. on Knowledge and Data Engineering, 2018.

[34] Doan AH, Madhavan J, Domingos P, *et al*. Learning to map between ontologies on the semantic Web. In: Proc. of the 11th Int'l Conf. on World Wide Web. ACM, 2002. 662−673.

[35] Do HH, Rahm E. COMA: A system for flexible combination of schema matching approaches. Proc. of the VLDB Endowment, 2002, 610−621.

[36] Stonebraker M, Bruckner D, Ilyas IF, *et al*. Data curation at scale: The data tamer system. In: Proc. of the Biennial Conf. on Innovative Data Systems Research. 2013.

[37] Halevy A, Korn F, Noy NF, *et al*. Goods: Organizing Google's datasets. In: Proc. of the 2016 Int'l Conf. on Management of Data. ACM, 2016. 795−806.

[38] Abedjan Z, Morcos J, Gubanov MN, *et al*. Dataxformer: Leveraging the Web for semantic transformations. In: Proc. of the Biennial Conf. on Innovative Data Systems Research. 2015.

[39] Abedjan Z, Morcos J, Ilyas IF, *et al*. Dataxformer: A robust transformation discovery system. In: Proc. of the 2016 IEEE 32nd Int'l Conf. on Data Engineering. IEEE, 2016. 1134−1145.

[40] Deng D, Fernandez RC, Abedjan Z, *et al*. The data civilizer system. In: Proc. of the Biennial Conf. on Innovative Data Systems Research. 2017.

[41] Fernandez RC, Abedjan Z, Koko F, *et al*. Aurum: A data discovery system. In: Proc. of the 2018 IEEE 34th Int'l Conf. on Data Engineering. IEEE, 2018. 1001−1012.

[42] Galil Z. Efficient algorithms for finding maximal matching in graphs. In: Proc. of the Colloquium on Trees in Algebra and Programming, 1983. 90−113.

[43] Bollacker KD, Evans C, Paritosh P, *et al*. Freebase: A collaboratively created graph database for structuring human knowledge. In: Proc. of the 2008 ACM SIGMOD Int'l Conf. on Management of Data. ACM, 2008.

附中文参考文献:

[1] 杜小勇,卢卫,张峰.大数据管理系统的历史、现状与未来.软件学报,2019,30(1):127−141. http://www.jos.org.cn/1000-9825/5644.htm [doi: 10.13328/j.cnki.jos.005644]

[2] 孟小峰,杜治娟.大数据融合研究:问题与挑战.计算机研究与发展,2016,53(2):231−246.

[3] 陈跃国,王京春.数据集成综述.计算机科学,2004,31(5):48−51.

[4] 杨先娣,彭智勇,刘君强,李旭辉.信息集成研究综述.计算机科学,2006,33(7):55−59.

王淞(1991－),男,湖北武汉人,博士,主要研究领域为数据库,数据管理,数据挖掘.

彭煜玮(1980－),男,博士,副教授,CCF 专业会员,主要研究领域为数据库,数字水印.

兰海(1993－),男,硕士,CCF 学生会员,主要研究领域为数据库与数据挖掘.

罗倩雯(1996－),女,硕士,主要研究领域为数据库.

彭智勇(1963－),男,博士,教授,博士生导师,CCF 会士,主要研究领域为数据库,大数据.