doi:10.11959/j.issn.1000-436x.2015316

大数据探索式搜索研究

杜小勇[1,2]，陈峻[1,2]，陈跃国[1,2]
（1. 中国人民大学 教育部数据工程与知识工程教育部重点实验室，北京 100872；2. 中国人民大学 信息学院，北京 100872）

摘 要：数据探索(data exploration)是有别于数据服务与数据分析的第 3 种体现大数据价值的技术手段。数据服务强调从微观层面获取满足用户需求的精准信息；数据分析强调从宏观层面为用户提供数据洞察，进而提供决策支持；而数据探索是一种支持用户在微观层面和宏观层面进行自由切换的、深入浅出的、交互式发掘数据价值的方式。首先，简要介绍大数据价值发掘的传统技术手段和特点，并引入探索式搜索；其次，详细阐述探索式搜索的定义与模型，总结探索式搜索的特点；随后，基于组件化的思想，设计探索式搜索系统框架，并综述每个组件所涉及到的挑战与关键技术；最后简要介绍了笔者在知识库探索式搜索方面的尝试。

关键词：大数据；知识库；探索式搜索；数据探索
中图分类号：TP392 **文献标识码**：A

Exploratory search on big data

DU Xiao-yong[1,2], CHEN Jun[1,2], CHEN Yue-guo[1,2]
(1. MOE Key Laboratory of Data Engineering and Knowledge Engineering, Renmin University of China, Beijing 100872, China;
2. School of Information, Renmin University of China, Beijing 100872, China)

Abstract: Exploratory search is a new approach for discovering the value of big data, compared with data serving and data analysis. Data serving emphasizes to meet users' information need at the micro-level, and data analysis emphasizes to discover insights among data at the macro-level. However, exploratory search is a way to support user to freely swap between micro-level to macro-level and interactively explore the value of data as well. Firstly, approaches for discovering the value of big data were discussed. Secondly, the definition, model and characteristics of exploratory search were illustrated. Thirdly, the architecture of exploratory search systems was designed, and a review of the challenges and techniques of each component of the architecture were given. Finally, preliminary results of exploratory search in RDF knowledge bases were introduced.

Key words: big data; knowledge base; exploratory search; data exploration

1 引言

关于大数据的讨论，除了被广泛认可的海量(volume)、异构(variety)、快变(velocity)3V 特性[1]外，人们更关注于大数据的价值(value)。现阶段，主要通过 2 种技术手段来体现大数据的价值：数据服务(data serving)和数据分析。

1.1 数据服务与数据分析

数据服务是指将大数据组织管理起来，提供高效的数据查询与信息检索服务。数据查询主要面向结构化类型的数据，采用基于键值对模型的 NoSQL 数据库技术，以行键、列名、版本号来确定数据的逻辑单元，并通过行键、列名和版本等信息来进行基于键值的数据查询。由于 NoSQL

收稿日期：2015-10-09；**修回日期**：2015-12-12
基金项目：国家重点基础研究发展计划（“973”计划）基金资助项目（2012CB316205）；国家自然科学基金资助项目（61472426）；中国人民大学科学研究（中央高校基本科研业务费专项资金）基金资助项目（14XNLQ06）
Foundation Items: The National Basic Research Program of China (973 Program)（2012CB316205); The National Natural Science Foundation of China (61472426); Fundamental Research Funds for the Central Universities; The Research Funds of Renmin University of China (14XNLQ06)

数据库弱化了数据事务一致性准则（采用最终一致性），数据索引相对简单，事务类型单一，适于并行化处理，其在一定规模的集群下能够达到较高的数据读写吞吐率。信息检索是指从大规模的数据集中快速查找满足用户需求的资料或数据片段的过程[2]。用户通过关键词（或自然语言语句）来表达信息需求。为了快速得到反馈，必须预先构建好数据索引。完成检索后，结果要根据与查询的相关度进行排序。无论数据查询还是信息检索，一般都采用“提交问题—返回结果”的一次性交互模式，查询处理利用索引，快速定位满足用户需求的数据。因此，数据服务对数据价值的利用是最直接的。

数据分析是指用适当的统计分析方法对大量数据进行分析或建模，然后提取有用信息并形成结论，进而辅助人们决策的过程[3]。在这个过程中，用户会有一个明确的目标，通过“数据清理、转换、建模、统计”等一系列复杂的操作，获得对数据的洞察，从而协助用户进行决策。常见的数据分析有在线联机分析处理(OLAP 分析)与深度分析。OLAP 分析一般采用 SQL 查询语句对结构化数据进行多维度的聚集查询处理；而深度分析则采用了复杂度较高的数据挖掘和机器学习中的一些方法，可以处理结构化数据甚至是非结构化数据。数据分析一般基于大量数据和较为复杂的运算模型，其结果信息量通常很大，适用于宏观决策。而对于细节层面信息的获取，数据分析缺乏如索引和访问控制等方面的技术支持。

表 1 总结归纳了数据服务和数据分析 2 种方式的特点：1）在用户信息需求层面，这 2 种手段都要求用户有明确的信息需求，相比数据分析，数据服务的信息需求更加单一；2）在搜索对象层面，数据服务的对象是数据集合内的某些元素，而数据分析的对象是整个数据集或其子集；3）在观察角度层面，数据服务的角度是微观的，数据分析的角度是宏观的；4）在用户目的层面，数据服务是侧重于查询资料和数据片段，而数据分析的目的侧重于决策支持；5）在交互模式层面，数据服务与数据分析主要是一次性的交互模式。但在交互式场景中，它们也会遇到查询调整的问题，用户通过多轮的交互来满足信息需求，而各轮之间却是独立的查询或者分析任务。

表 1 各类大数据价值挖掘方式比较

特点	数据服务	数据分析	数据探索
信息需求	单一、明确	多样、明确	多样、变化
搜索对象	点	面	点面结合
观察角度	微观	宏观	微观为主，辅之宏观分析
目的	查阅资料	决策支持	学习、调研
交互模式	一轮、多轮	一轮、多轮	多轮

1.2 数据探索

以上 2 种方式分别从 2 个角度发掘大数据的价值，数据服务强调从微观层面获取满足用户需求的精准信息，数据分析强调从宏观层面为用户提供数据洞察，进而提供决策支持。这 2 种方式能有效帮助用户解决很多常见问题，发现大数据固有的价值。但仍然存在诸多场景（例如学习、调研），单纯的微观层面的信息获取和宏观层面的数据分析都不能有效协助用户去发现和探索数据中的价值，用户更需要的是一种可以在微观层面和宏观层面进行自由切换的、深入浅出的、交互式的探索数据价值的方式。下面的旅行规划问题是一个典型的例子。

小明第一次去某地旅游，为了旅途顺利，想事先规划一下。他大致思路如下。第 1 步，要选择交通方式；第 2 步，要调研当地值得体验的地方，如景点和小吃等；第 3 步，需要确定住宿；第 4 步，要设计规划住宿地点到景点的交通路线。以上过程没有明确的先后顺序，但都需浏览、对比大量信息。在持续的浏览过程，他的某个决定随时可能诱发其他某个环节的更改，进而引发全局的调整，比如更换住宿地点，那么交通路线需要重新设计。在这个过程中，小明需要不断地重复“搜索—思考”的过程来完成这次旅行规划。

结合上述例子，小明起初的目标是比较模糊的，他需要在不断获取信息的过程来调整搜索目标。此外，小明需要系统提供额外的信息进行引导，引导的过程中，目标随时可能改变，这种改变的动机可能出自于获取必要信息，也可能出于好奇心。出于这样的目的，探索式搜索(exploratory search)的概念应运而生。

探索式搜索主要是针对目标可变的、持续的、多角度的搜索任务，其搜索过程是有选择的、有策略的和反复进行的[4]。它将以找到信息为目的的传统信息检索模式变为以发现、学习和决策为目的的信息搜寻模式。这样的搜索模式结合了大量的分析与人机交互过程，适合于人们从数据中发现和学习

更多的内容。在某些领域，数据的探索式搜索也被称为数据探索。

目前，随着大数据研究的兴起，探索式搜索这种交互式的分析和探索数据价值的方式，逐渐引起人们的重视[5]。很多数据类型已经有了探索式搜索的应用研究，如媒体数据[6]、网页[7]、图数据[8]、异构信息网络[9]、关系型数据[10]、RDF 知识库[11]等。在这些应用中，尤其是面向大数据的探索式搜索方面，还有很多问题等待研究者们进行深入的研究。

2　探索式搜索

最近几年，探索式搜索逐渐获得相关领域研究者们的关注。数据库领域顶级会议 SIGMOD 于 2014 年针对探索式搜索举办了首次研讨会，与会专家从多个角度讨论了探索式搜索的重要性与必要性，并将探索式搜索与以往的交互模式做了区分[12]。次年，SIGMOD 会议再次针对探索式搜索的技术实现举办了研讨会，与会专家从系统实现层面讨论了探索式搜索所需要克服的技术挑战[13]。

探索式搜索的概念是于 2006 年被数字图书馆领域的权威学者 Marchionini 在 ACM 通信上首次明确提出的[4]。而对于探索式搜索的讨论最早源于 2005 年，马里兰大学的几位专家主导举办了有关探索式搜索界面设计的交叉学科研讨会，该研讨会召集了人机交互、信息检索、信息搜寻以及信息可视化等领域的专家，探讨这门交叉学科的界面设计、评价方法以及认知过程[14]。此后，一系列研讨会在相关顶尖会议上举办，如 2006 年 SIGIR[15]讨论了如何评估探索式搜索系统，2007 年 SIGCHI[16]讨论满足探索式搜索界面设计的要求以及面临的挑战。

2.1　探索式搜索的定义

Marchionini 将人类对信息的需求从低到高分为 3 个层次[4]：1）探寻基本的事实，辅助解决一个短期的任务；2）联系相关概念，帮助人们理解某个现象或者执行某项较为复杂的任务；3）整合相关策略与知识，帮助成为某个领域的专家。为了支持后 2 个层次的需求，用户需要通过不断的交互过程，调整自己的信息搜寻目标，全方位多角度地了解相关领域的信息。因此，交互模式需要获得更大的突破。

然而，“提交查询—返回结果”的一次性交互模式仍是众多数据库和信息检索系统所采用的交互模式（如图 1 所示），用户只需提出一个查询，即可获得与该查询相关的结果。事实上，很多信息系统的实际应用却经常伴随着多次的交互过程，用户经常要花费大量精力去反复浏览、对比和分析反馈查询结果，用户体验糟糕。其本质原因在于：1）用户不够了解数据域(data domain)，抽象而成的查询不够准确；2）一次性交互模式不能很好地适应用户在检索过程中对信息需求的多样性与动态性，并且忽略了查询过程的上下文语境等因素[17]，无法很好地协助用户与系统交互。

为了改善上述缺陷，信息检索引入了迭代式查询的理念，帮助用户逐步缩小查询范围，最终定位到他们所需的信息。但是，很多情况下，用户并没有明确的搜索目标，对知识的好奇是他们搜索的动机，他们需要在搜索过程中被引导，从而明确他们的目标。基于这样的背景，探索式搜索的概念被提出来了。

根据 Marchionini[4]与 White[17]给出的定义，探索式搜索由问题上下文与搜索进程 2 个相辅相成的主体构成，其问题上下文由用户的信息需求驱动，这种需求是开放式的、持续的、多角度的；其搜索进程由用户的行为组成，这种行为是有选择的、有策略的和反复多次进行的。

2.2　探索式搜索的模型

通过分析用户的信息需求，Marchionini[4]将用户的搜索任务分为 3 类（如图 2 所示）：1）查阅(lookup)：通过构建一个简单、有效的查询，在特定数据域中完成基本的信息检索；2）学习(learn)：通过多次迭代查询，对反馈的结果进行查阅、对比，最终整合吸收；3）调研(investigate)：通过多轮多次迭代查询，不断关联此前学习到的知识，加以辅助，进一步对反馈的结果进行更深层次的探索。这些任务之间存在不同程度的交集，查阅作为最基本的搜索任务，经常被其他两项搜索任务所涵盖，而学习是调研的重要组成部分。探索式搜索的目的是为了更好地解决学习与调研 2 项搜索任务。

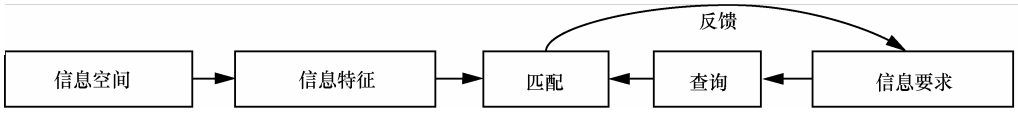

图 1　基于“提交查询—返回结果”的一次性交互模式

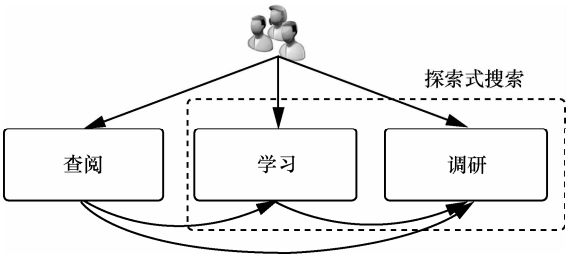

图 2 搜索任务

对于探索式搜索用户群体而言，其最大的特点是因缺乏对背景知识的了解，没能形成明确的搜索目标，其搜索的兴趣点是被当前的查询结果和与当前结果紧密关联的数据内容所引导和转移的。与此同时，若用户对某个兴趣点感兴趣，其随时可以深入该兴趣点，进一步挖掘信息。

为此，White[17]将探索式搜索抽象成为 2 个重要的过程：1）探索式浏览(exploratory browsing)；2）集中式搜索(focused searching)。探索式浏览的目的是为了更加开放地探索数据，在用户未确定他们真正的搜索意图前，探索式浏览会有策略地提供用户更多的相关知识，帮助用户在海量的数据中找到他们感兴趣的内容。集中式搜索目的是为了让用户更加深入地探索数据，当用户确定他们某个阶段的兴趣点，集中式搜索会协助用户不断深入该领域，帮助用户挖掘细节。

如图 3 所示，为了让用户获取更多的知识，以上 2 种模式会交替出现在整个搜索进程中，用户随时可以从某个兴趣点转移到另外一个兴趣点。这种交替模式促进了用户与系统间的良性交互，系统在搜索过程中更加了解用户的习惯与特点，从而提供更相关的兴趣点与更准确的内容。此外，用户的搜索目的也会随着搜索进程的推进不断波动，最终趋于稳定。

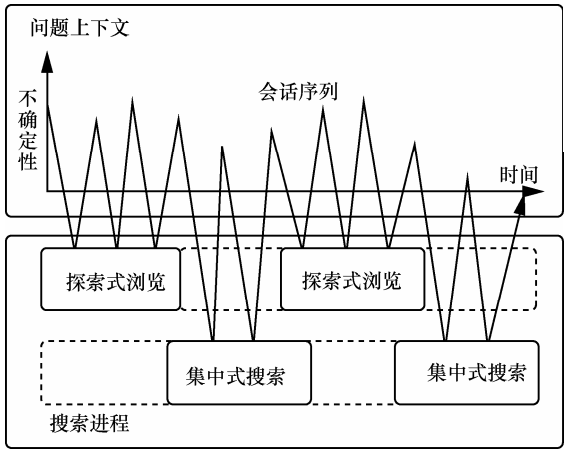

图 3 探索式搜索模型

2.3 探索式搜索的特点

结合 Marchionini[4]与 White[17]的观点，探索式搜索有以下几项特点。1）搜索过程是长期的：用户的每次搜索会话都应该被记录下来，系统会分析利用这一连串会话，对用户的行为进行分析，从而更好地协助用户去探索数据；2）信息需求是开放式的、持续的、多角度的，用户具备好奇的属性，好奇会导致他们的信息需求在搜索进程中不断发生变化，他们的搜索意图也将会随着搜索进程的推进而不断波动，这种变化将会让用户了解更多面的信息；3）探索与发现是重点，相比基本的查阅，探索式搜索强调发现更多相关的内容，从而帮助用户更加全面地了解某个话题。相比表 1 另外 2 种传统价值发现的方式，探索式搜索强调用户的充分参与，在搜索进程中，该方式会为用户提供大量相关信息，引导用户明确信息需求，并拓展用户知识面。因此，该方式更适合人们从数据中发现和学习更多的内容。

探索式搜索涉及多方面的技术挑战，既包括大数据的高效管理与查询执行等系统层面的技术，也涉及用户与系统间交互的创新与突破，如人机交互、数据可视化等。下一节将从系统的角度分析探索式搜索系统需要应对的具体挑战与关键技术。

3 系统框架、挑战与关键技术

White[17]归纳了探索式搜索系统的几大要素。1）查询构建：协助用户构建查询，并支持查询的快速重构；2）分类详情：对返回结果的进行分类，方便用户进行筛选；3）搜索上下文：记录搜索进程的上下文，理解用户行为；4）可视化支持：提供可视化支持，便于用户更加直观地了解数据；5）辅助学习：提供充分的信息，协助用户在搜索的过程中学习、理解知识；6）社交化操作：提供社交化的功能，提升用户的参与感与兴趣；7）会话记录：记录用户的行为，方便用户推进自己的搜索进程；8）任务管理：支持多会话、多用户的场景。

根据上述观点，参考现有研究的思路，设计了一个探索式搜索系统的参考框架，包括人机交互层、查询处理层和数据管理层（如图 4 所示）。在设计的过程中，采用了组件化的思想，其中，人机交互层涵盖了交互界面组件、社交化组件以及可视化组件；查询处理层涵盖了查询构造组件、查询执行组件以及结果重构组件；数据管理层涵盖了会话管理组件、数据管理组件以及元数据管理组件。每个组件都有

各自的功能与特点，组件之间相辅相成。

3.1　人机交互层

人机交互层是用户与系统直接对话的平台，好的人机交互层设计能让用户与系统之间的信息交换过程更加有效。因此，探索式搜索系统需要在人机交互层引入必要的交互元素，协助用户更准确表达、获取自己的信息需求。

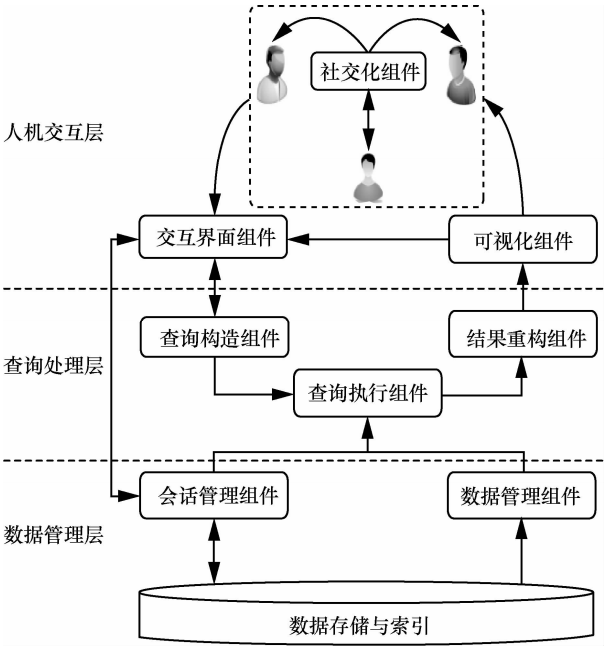

图 4　探索式搜索系统框架

3.1.1　交互界面组件

交互界面组件的设计需要关注以下几点：1）交互界面各个元素的设计需要秉持用户友好的准则，尽量降低用户的学习成本；2）探索式搜索是一个长期的搜索进程，用户需要知道自己所处搜索进程的确切位置；3）需要协助用户快速地构建查询，并能提供高效的查询重构方案，降低用户输入代价，提高查询构建的准确性；4）需要提供与当前查询结果紧密关联的数据内容，发散用户的兴趣。

目前，交互界面方面已经有很多工作。Agapie 等[18]认为长查询利于系统返回相关结果，但用户一般习惯输入短查询，为此，他设计了一种交互式查询输入系统。该系统随着用户输入查询的长短，输入框的颜色发生变化，以此提高用户输入长查询的概率。SearchPanel[19]的作者观察到用户在搜索的过程中，会重复性地访问同个内容，于是他们基于 Chrome 设计了一个插件，该插件记录用户的浏览过程，帮助用户管理他们的搜索进程。Querium[20]系统是一个探索式学术搜索系统，该系统在用户界面设计方面集成了很多交互式的元素，包括提供搜索记录、结果筛选、查询提示等功能，有效地协助用户找到他们所需的论文。Querium 系统在交互的实时性上也提供很多借鉴，如图 5 所示，该系统在每条答案右侧提供了支持与反对 2 个按钮，用户在点击之后，系统会根据用户的选择情况，实时更新答案列表，这让用户与系统之间的交互性更强。

3.1.2　社交化组件

社交化组件强调帮助用户进行协同搜索(collaborative search)，并基于用户社交行为为用户提供更加精准的内容推荐。

一些大型搜索任务（如医学领域的搜索）不是单个用户能完成的，往往需要支持多名用户协同搜索。Golovchinsky 等[21]认为协同搜索可以融合不同用户的个人见解、经验、专业领域知识等，从而发挥群体优势。在团队协作下，用户彼此的交流能帮助用户更加明确个人的信息需求。此外，用户可以在协同搜索的过程中共享他人的搜索结果、吸收他人的知识。SearchTogether[22]是一个协同搜索领域较早的原型系统，该系统支持团队式的搜索，大型的

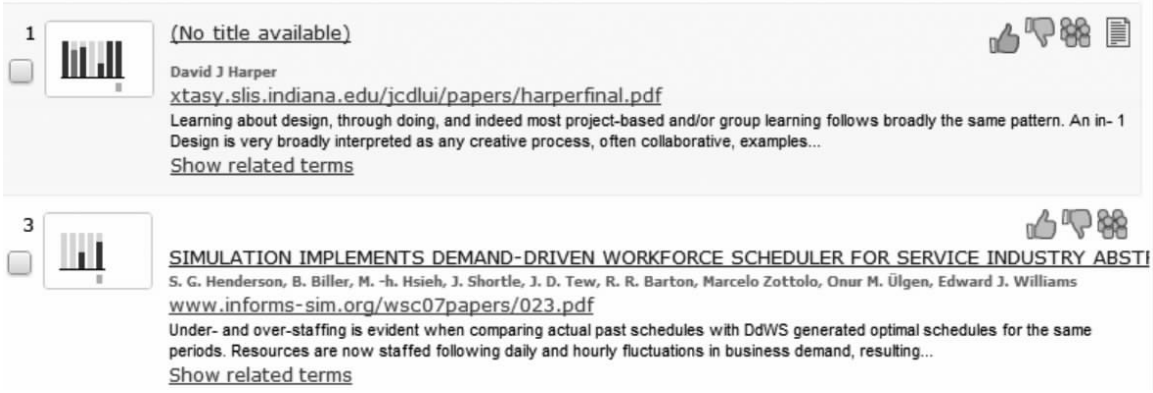

图 5　Querium 查询结果

搜索任务可以被拆分成多个子任务，团队成员可以在搜索的进程中交互交流，并共享搜索成果。

除团队式的协同搜索外，系统可以提供其他形式的社交元素，让用户在搜索进程中激发更多的兴趣。目前，社交媒体包括微博、Twitter 在这方面有诸多工作可以借鉴。以微博为例，微博为每条内容提供收藏、转发、评论以及点赞等社交化元素，这些元素不仅能吸收用户的智慧，还能让用户对这些信息有其他维度的认知。与此同时，微博能通过分析用户的社交圈，为用户推荐其他感兴趣的内容。这种社交化的模式可以最大程度地发挥用户群体的智慧，非常适合探索式搜索的理念。

3.1.3 可视化组件

可视化组件能加强用户对信息的认知，使用户能够目睹、探索以至快速理解大量的信息。据研究表明，人类从外界获得的信息约 80%以上来自于视觉系统[23,24]，当数据以图像的形式展现时，用户往往能够一眼洞悉数据背后所隐含的价值，而这种价值可能在其他形式下不易发觉。例如，图 6 是 Google 知识图谱的一个查询，当用户输入达芬奇，系统自动反馈与达芬奇相关的实体，实体间关系的强弱、远近在可视化地展示下，更适合用户从视觉上获取容易被忽略的信息。因此，为了更有效地探索数据价值，数据的可视化分是不可或缺的重要手段与工具[25]。

图 6 Google 知识图谱

目前，可视化研究领域主要关注文本可视化、网络可视化、时空数据可视化以及多维数据可视化的研究[26]。然而，可视化技术与探索式搜索的结合还不深入，但已经逐渐有各方面的尝试。如图 5 所示，Querium 系统提供了一个位于返回结果左侧的可视化插件。该插件以直方图的形式直观地反应了结果与最近几个查询的关联程度。Polaris[27]系统将多维数据进行可视化展示，让用户对多维数据有更加直观地认识。VizDeck[28]是一个自动化的可视化组件管理工具，可以通过分析查询结果，给出适合的可视化方案，帮助用户获取更多隐含信息。

3.2 查询处理层

探索式搜索对信息的获取也是通过查询来实现的。因为目标不确定是探索式搜索的重要特点，因此查询层需要提供更多的功能支持交互层。

3.2.1 查询构造组件

查询构造组件支持交互层的查询推荐与查询重构。查询推荐在传统的搜索引擎中已得到充分的运用，每当用户输入部分关键词，系统会快速地补齐缺失的语义，并在下拉框内提供多条查询建议，降低了用户的操作代价。

当前查询若不满足用户的意图，用户会开始下一轮的查询，但用户往往缺乏对数据的了解，因此系统需要支持用户快速重构查询。目前，Web 与数据库领域都有相关研究。在数据库中，用户常常因为不熟悉表之间的关联结果，导致 SQL 查询的构建连接操作时存在问题，DataPlay[29]对关系型数据的表结构进行了图形化展示，方便用户调整 SQL 语句。此外，为了获取准确的信息，用户需要在查询的基础上加上限制条件，但往往因为缺乏对数据的了解，导致难以提供准确的限制条件，Qarabaqi[30]对于上述情况提出了一个交互式框架，协助用户逐步构建准确的查询。Tran 等[31]发现有些用户很难将他们的信息需求抽象成查询，但当他们获取到一些有关信息之后，可以顺利重构查询。

3.2.2 查询执行组件

获取查询之后，查询执行组件会返回查询结果与相关内容。因为探索式搜索是个长期的过程，系统可以有效地关联用户的搜索进程，进而提升返回结果的准确性。Shokouhi[32]在文中指出，短查询容易产生歧义，但通过分析用户的搜索记录，搜索结果会更加精准。与此同时，通过关联用户的操作行为，系统会对用户的搜索意图具备更深层次的理解，从而优化得搜索结果[33~35]。

此外，为了引导用户进一步探索数据，相关内容的推荐不可或缺。对于信息推荐而言，数据挖掘、机器学习有大量工作值得借鉴。例如，YmalDB[36]通过对关系数据库查询结果的分析，推荐出用户可能感兴趣属性值对的组合，作为查询结果的附加信息呈现给用户，以引导用户进一步探索数据库

中的数据。现阶段，像百度、Google 以及 Bing 这些大型搜索引擎都提供了类似的功能，用户不但可以在获取与查询有关的文档，还能探索与结果相关的内容。

3.2.3 结果重构组件

传统搜索引擎返回给用户的是与查询最为相关的多个文档，但用户仍然需要花很多的精力在文档内找寻他们想要的信息。因此，为了让用户更加直观地获取信息，系统需要将返回的结果加以抽取、重构，以更加结构化的方式展示给用户。目前，大量的信息抽取与信息集成领域的工作与该组件密切相关。MobEx[37]是一个基于移动设备的探索式搜索系统，该系统通过 Web 端结果获取页面信息之后，通过信息抽取的方式将文本信息以图的形式展现给了用户，类似的系统还有微软的人立方。

与此同时，用户在浏览过程中会不断扩大、缩小他们的浏览深度，这要求系统对返回结果进行分类，从而为人机交互层提供辅助用户快速筛选反馈结果的信息。目前，很多系统提供了类似功能，如 Hippalus[38]系统通过分析返回结果，将内容以多级层次的形式展现给用户，用户可以通过筛选层次以及分类来快速定位到他们所需要的信息。除此之外，返回结果的元数据也可以作为分类的依据，如学术搜索引擎会将论文的年份、学科以及作者等数据作为分类信息，帮助用户快速过滤掉无关的内容。

3.3 数据管理层

高性能查询处理是探索式搜索能被广大用户接受的前提。与此同时，系统同时需要具备良好的可扩展性。为了满足上述需求，数据管理层的设计尤为关键。

3.3.1 会话管理组件

会话管理组件管理用户在搜索进程中的行为，系统会在用户的搜索进程中记录用户每个操作以及用户浏览的信息。虽然用户在探索初期的目的不太明确，但通过分析用户的操作上下文，系统能猜测用户的大致目标与兴趣，从而更加高效地引导用户。为了支持记录与分析功能，会话层需要同时支持不断记录和实时分析用户的操作行为。

3.3.2 数据管理组件

数据管理组件不同于会话管理组件，没有数据持久化的事务性要求，因此，快速的获取信息以及支持小规模数据量的分析是数据管理层需要面对的挑战。目前，有部分研究通过数据预取(data prefetching)降低查询时间。该技术通过分析用户当前查询的内容，提前载入未来可能需要的数据，进而降低用户在下个查询时所需要的 I/O 开销，该技术已在空间数据查询[39]得到验证。除此之外，若用户可以接受一定范围内的误差，查询近似(query approximation)是可采取的技术之一，该技术通过采样数据[40~44]牺牲部分精度，目的是为了快速返回近似结果，帮助用户对数据有初步的了解。

4 知识库的探索式搜索

近年来，信息抽取和数据集成等技术发展迅速，催生了大量大规模的 RDF(resource description framework)[注1]数据集。如 DBPedia [注2]、Freebase [注3]、OpenCyc [注4]、Wikidata [注5]、YAGO[45]等。目前常见的 RDF 数据查询检索方法有 2 种：使用关键词查询 RDF 数据或者利用 SPARQL [注6]查询语言检索 RDF 数据。但 SPARQL 查询受限于用户对 RDF 数据的了解程度，而关键词查询语义表达能力太弱，无法对 RDF 数据给出结构层面的约束。面对结构复杂、规模庞大的 RDF 数据库，用户通常很难明确自己的信息需求，很难通过简单的查询检索到理想的数据。探索式搜索的提出能有效地协助用户解决上述问题，用户通过多轮的交互和探索过程，可以逐步调整搜索目标，进而从庞大复杂的 RDF 数据库中找到感兴趣的数据。在这节中，以 RDF 知识库上的探索式搜索为例，探讨探索式搜索所要面临的一些挑战性问题和解决这些问题的关键技术。

4.1 RDF 知识库

RDF 是由 WWW 提出的对万维网(world wide web)上信息进行描述的一个框架，它为 Web 上的各种应用提供信息描述规范[46]。RDF 用主语、谓词、宾语的三元组形式来描述 Web 上的资源。其中，主语一般用统一资源标识 URI(uniform resource identifiers)表示 Web 上的信息实体（或者概念）；谓词描述实体所具有的相关属性；宾语为对应的属性

注1　http://www.w3.org/RDF/。
注2　http://dbpedia.org/。
注3　http://www.freebase.com/。
注4　http://opencyc. org/。
注5　http://wikidata.org/。
注6　http://www.w3.org/TR/rdf-sparql-query/。

值。这样的表述方式使 RDF 可以用来表示 Web 上的任何被标识的信息[47]。

此外，人们还提出了关联数据(linking open data)[注7]的概念，用于将不同组织机构发布的数据关联起来，形成规模更为庞大的 RDF 数据集。据统计数据显示，关联数据的规模在近几年快速增加，已经从 2011 年的 295 个数据增加到 2014 年的 1 014 个[注8]。很多海量的 RDF 数据集由于包含了大量来自不同领域的实体以及实体之间的关联信息，也常被称为 RDF 知识库（或知识图谱）。一些应用开始借助 RDF 知识库所能提供的知识，支持实体搜索、语义搜索、问答系统等应用，谷歌的 Knowledge Graph 就是其中一个典型的例子。

4.2 知识库探索的挑战与关键技术

面对规模庞大的 RDF 知识库，用户通常难以明确自己的信息需求。然而，在探索式搜索的协助下，用户可以逐步调整和改进搜索目标，更有效地从庞大复杂的 RDF 知识库中找到感兴趣的数据。在交互过程中，用户还可以深入了解 RDF 数据的结构（包括数据间的关联）、数据的分布、数据的丰富度等有价值的信息，也能够发现一些因各种原因造成的数据质量问题。

4.2.1 人机交互

用户交互界面是 RDF 知识库探索系统研制的一个重要环节，该环节可以根据应用层的不同需求进行个性化的设计。用户界面设计的好坏直接影响到系统的易用性，在追求功能的同时，需要保证界面的直观简洁，第 3 节所提到一些研究成果可以作为系统实现的参考。在另一方面，搜索结果的可视化也是需要研究的内容，针对 RDF 图数据的特点，使用一些信息可视化技术展示查询结果以及数据之间的关联，促进用户对查询结果的理解，降低查询结果上下文语境理解的难度，以增强 RDF 知识库数据可视化的交互式数据分析的功能。

4.2.2 查询处理

现阶段，人们对海量 RDF 知识库的存储、信息查询以及分析等方面已经做了大量的研究工作[47]。然而，目前的解决方案存在的一个较大问题是缺少表达能力强且简单易用的 RDF 数据查询方法。关键词查询目的是在 RDF 数据库中，找到包含所有关键词的、结构紧凑的子图/树。其虽然灵活度大、实用性强，却很难保障结果的查准率和查全率。而且关键词查询语义表达能力弱，不能对 RDF 图数据给出结构上的约束。在另一方面，结构化的 SPARQL 查询力图在数据库中找到满足 SPARQL 查询条件的子图，其有着较为复杂的语法定义，需要用户熟悉它的语法规则并了解 RDF 数据的模式信息（如谓词和前缀等），才能够使用该语言查询 RDF 数据，这对于一些包含简单模式的垂直应用尚可。但对于谓词数量繁多的、面向开放领域的海量 RDF 数据集而言，SPARQL 语言对于普通用户甚至专业开发人员都不具备良好的实用性。为此，需要研究针对 RDF 知识库的探索式搜索所需要的基本操作，设计新的基本原语。

在设计基本原语的过程中，需要结合 RDF 数据与探索式搜索的特点。在每次交互过程中，系统能够分析出上几次交互的查询结果的特征，以及和这些结果紧密关联的相关数据的特征。在此基础上，识别用户可能进一步感兴趣的数据内容，简明合理地向用户展示查询结果和与其紧密关联的用户潜在感兴趣的数据内容，以引导用户改进和调整查询目标，探索新的关联信息。这其中会存在一些基于顶点、路径、子图的图数据探索和分析操作，他们可以抽象成为 RDF 数据的一些基本原语。对于每个基本原语，需要明确定义其输入数据的形式、所执行的基本运算操作、输出结果的形式，并研究相应的计算复杂性。在此基础上，还要研究不同基本原语之间的关联关系，研究如何在不同基本原语之间建立逻辑上的关联，以及如何通过基本原语的组合，逻辑上形成一个完整的探索式搜索会话过程，作为探索式搜索系统的基础交互模型。

4.2.3 数据管理

在海量 RDF 数据上进行探索式搜索是本项目面临的最大挑战。图数据处理的算法复杂性通常远高于关系数据处理的复杂性，且算法需要经常随机读取数据。即便是当前一些包含上亿三元组的 RDF 数据集，已经是超大规模的图数据。单节点的基于外存模式的很多图数据处理算法都远不能满足在这样的数据集上交互式查询处理的性能需求（亚秒级）。因此，需要从体系结构的角度研究支撑海量 RDF 数据探索式搜索的数据存储与

注7 http://linkeddata.org/。

注8 http://lod-cloud.net/。

索引策略，而现有的图数据库[48~51]、MPP 分析型数据库[52~54]、分布式内存数据库[55~58]等相关工作均可以作为借鉴。

目前，分布式图数据库系统是针对大规模 RDF 数据管理常用的技术手段，典型的有 Pregel[48]、GraphLab[49]、GraphX[50]、Trinity[51]等。但是，在分布式的计算环境下，很多图算法因计算同步很容易造成过多的消息传递，影响性能。如 Pregel、GraphLab 以及 GraphX 都是基于 BSP 计算模型[59]，它们将图数据分析过程分解成一系列超步，计算以图的顶点为中心，并利用超步的状态传递中间计算结果、同步节点间的计算，获得了高性能、扩展性好的大规模图数据分析解决方案。然而，这些方法都是针对全图的离线分析，在大规模数据的情况下无法提供实时地返回分析结果。Trinity 则通过内存云的引入，使用键值对方式分布式存储图数据，并借助内存数据存取来提升图数据随机访问的性能，进而支持一些高性能的图数据在线查询处理。

因此，以分布式的方式存储和处理海量 RDF 数据是提高大规模图数据处理可扩展性的一条重要途径。此外，内存数据管理方法的使用也是性能提升的重要保障。因为探索过程中会涉及到很到信息片段，高效的索引支持是必须的。与此同时，存储管理方面的优化，如数据压缩、存储格式都会是提升性能的重要方式[58]。

4.3　原型系统 SEED

目前，笔者在人机交互层面与查询处理层做了一些尝试，基于前期研究，现已实现了一个原型系统 SEED。该系统采用实体集合扩展的方法来探索 RDF 知识库，用户通过交互界面输入若干个实体，该系统可以挖掘实体在知识库中存在的语义关联，获得该实体集合的共同特征，进而获取所有其他的相关实体，并将语义关系呈现给用户。如用户输入数据库领域的专家 Jim Gray、Edgar F Codd、Charles Bachman 与 Michael Stonebraker，系统会返回所有该领域的专家，并提供实体集合的语义关系(如 subject-category: database researches)，帮助用户快速获取知识。

SEED 的架构（如图 7 所示）与第 3 节所描述的框架一致，包含了人机交互层、查询处理层与数据管理层。人机交互层为用户提供可视化的界面，方便用户探索知识库。查询处理层涵盖 2 个模块，实体集合扩展模块和实体关系预测模块。为了高效地探索知识库，数据管理层需要引入索引。

用户在探索知识库时，可能会发现知识库信息不完善的缺陷。基于上述原因，系统为用户提供了纠错的功能，目前已提供知识库信息补全的功能，用户可以结合自己的背景知识和系统的推荐信息

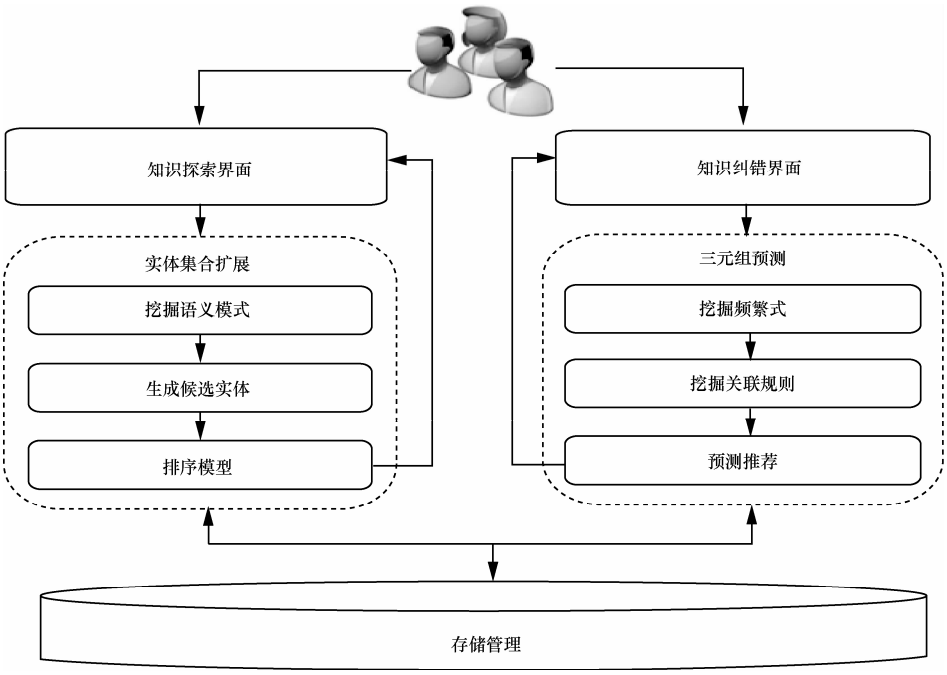

图 7　SEED 系统架构

进行添加操作。如图 8 所示，当用户在左侧实体列表中点击 Michael Stonebraker 时，右侧会即时返回该实体与全部语义关系之间的联系，加号表示该实体与语义关系所形成的三元组不存在于数据库，因获取数据集的时候，Michael Stonebraker 未获得图灵奖，但 SEED 通过分析相关实体的语义关系，可以预测 Michael Stonebrake 获得图灵奖的概率，为用户的操作提供相应的推荐。

此外，为了充分了解各个实体的信息，用户可以通过点击实体，获取与该实体直接联系的实体，这些实体与相应的关系将以有向图的方式展示给用户（如图 9 所示）。

5　结束语

探索式搜索是适合大数据价值挖掘的新手段。本文在对比了传统的数据价值发掘方式基础上，着重介绍了探索式搜索的概念与模型，并总结了探索式搜索的特点与需要面临的挑战。随后，基于组件化思想，设计了探索式搜索系统的系统框架，包括人机交互层、查询处理层以及数据管理层，分别阐述了各个组件的功能要求，并综述相关工作。本文最后以 RDF 知识库为例，梳理知识库探索式搜索在各个层面需要应对的挑战与关键技术，并简要介绍了笔者的原型系统。探索式搜索作为一个新的研究方向，仍然有大量

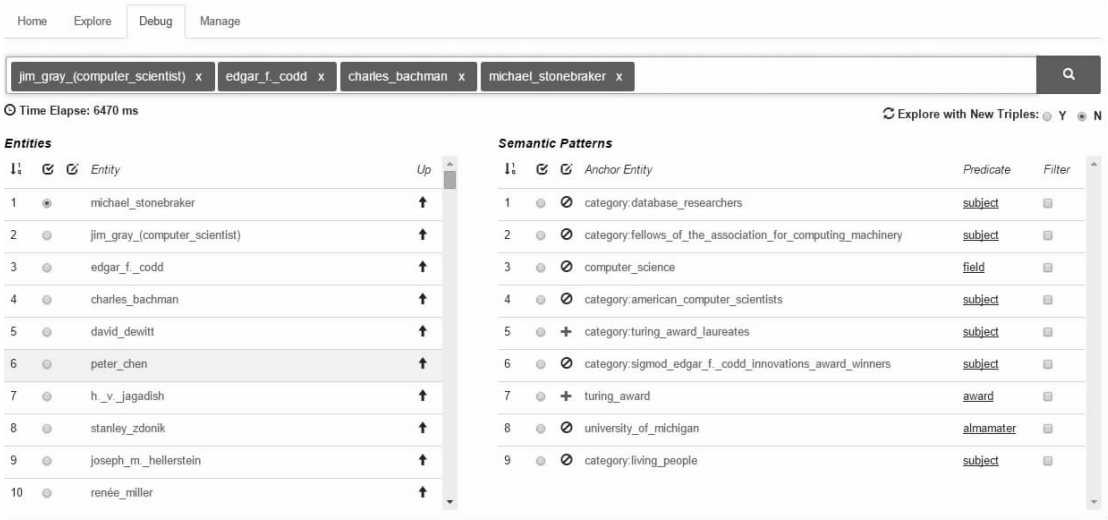

图 8　SEED 系统纠错功能

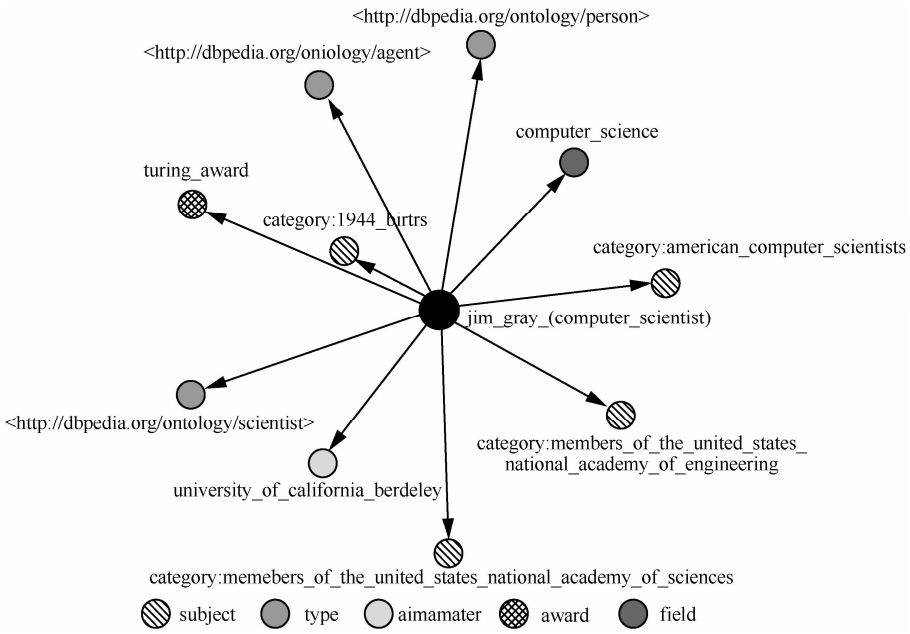

图 9　实体关联展示

的问题与挑战需要深入的研究与突破。下一步，将借鉴现有的前沿研究成果，在支持大规模知识库探索式搜索的关键技术上取得突破。

参考文献：

[1] MENG X F, CI X. Big data management: concepts, techniques and challenges[J]. Journal of Computer Research and Development, 2013, 50(1): 146-169.

[2] MANNING C, RAGHAVAN P, SCHÜTZE H. Introduction to Information Retrieval[M]. Cambridge University Press, 2008.

[3] JUDD C, MCCLELLAND G, RYAN C. Data Analysis: a Model comparison approach[M]. Routledge Press, 2009.

[4] MARCHIONINI G. Exploratory search: from finding to understanding[J]. Communication of the ACM, 2006, 49(4):41-46.

[5] HECHT B, CARTON S, QUADERI M, et al. Explanatory semantic relatedness and explicit spatialization for exploratory search[A]. SIGIR[C]. 2012.415-424.

[6] ROITMAN H, YOGEV S, TSIMERMAN Y, et al. Exploratory search over social-medical data[A]. CIKM[C]. 2011, 2513-2516.

[7] BOZZON A, BRAMBILLA M, CERI S, et al. Exploratory search in multi-domain information spaces with liquid query[A]. WWW[C]. 2011.189-192.

[8] HAM F, PERER A. Search, show context, expand on demand: supporting large graph exploration with degree-of-interest[J]. IEEE Transaction on Visualization and Computer Graphics, 2009, 15(6): 953-960.

[9] DUNNE C, RICHE N, LEE B, et al. GraphTrail: analyzing large multivariate, heterogeneous networks while supporting exploration history[A]. CHI[C]. 2012.1663-1672.

[10] YOGEV S, ROITMAN H, CARMEL D, et al. Towards expressive exploratory search over entity-relationship data[A]. WWW[C]. 2012. 83-92.

[11] MIRIZZI R, RAGONE A, SCIASCIO E. Like breadcrumbs in the forest: a tool for semantic exploratory search[A]. EDBT/ICDT Workshop on Linked Web Data Management[C]. 2011. 32-33.

[12] KOUTRIKA G, LAKSHMANAN L, RIEDEWALD M, et al. Report on the first international workshop on exploratory search in databases and the Web[J]. SIGMOD Record, 2014, 43(2): 49-52.

[13] IDREOS S, PAPAEMMANOUIL O, CHAUDHURI S. Overview of data exploration techniques[A]. SIGMOD[C]. 2015.277-281.

[14] WHITE R, KULES B, BEDERSON B. Exploratory search interfaces: categorization, clustering and beyond[J]. SIGIR Forum, 2005, 39(2): 52-56.

[15] WHITE R, MURESAN G, MARCHIONINI G. Report on ACM SIGIR 2006 workshop on evaluating exploratory search systems[J]. SIGIR Forum, 2006, 40(2): 52-60.

[16] WHITE R, DRUKER S, MARCHIONINI G, et al. Exploratory search and HCI: designing and evaluating interfaces to support exploratory search interaction[A]. SIGCHI[C]. 2007.2877-2880.

[17] WHITE R, ROTH R. Exploratory search: beyond the query-response paradigm[M]. Morgan & Claypool Publishers, 2009.

[18] AGAPIE E, GOLOVCHINSKY G, QVARFORDT P. Leading people to longer queries[A]. CHI[C]. 2013. 3019-3022.

[19] TRETTER S, GOLOVCHINSKY G, QVARFORDT P. SearchPanel: a browser extension for managing search activity[A]. EuroHCIR[C]. 2013. 51-54.

[20] GOLOVCHINSKY G, DIRIYE A, DUNNIGAN T. The future is in the past: designing for exploratory search[A]. IIiX[C]. 2012.52-61.

[21] GOLOVCHINSKY G, QVARFORDT P, PICKENS J. Collaborative information seeking[J]. IEEE Computer Society, 2009, 42(3):47-51.

[22] MORRIS M, HORVITZ E. SearchTogether: an interface for collaborative web search[A]. UIST[C]. 2007. 3-12.

[23] REN L. Research on Interaction Techniques in Information Visualization[D]. Beijing: Chinese Academy of Sciences. 2009.

[24] CARD K, MACKINLAY D, SHNEIDERMAN B. Readings in Information Visualization: Using Vision to Think[M]. San Francisco: Morgan-Kaufmann Publishers, 1999.

[25] KEIM D. Information visualization and visual data mining[J]. IEEE Transaction on Visualization and Computer Graphics, 2002, 8(1):1-8.

[26] REN L, DU Y, MA S, ZHANG XL, et al. Visual analytics towards big data[J]. Journal of Software, 2014,25(9):1909-1936.

[27] STOLTE C, TANG D, HANRAHAN P. Polaris: a system for query, analysis and visualization of multi-dimensional relational databases[J]. IEEE Transactions on Visualization and Computer Graphics, 2002. 8(1)

[28] KEY A, HOWE B, PERRY D, et al. VizDeck: self-organizing dashboards for visual analytics[A]. SIGMOD[C]. 2012.681-684.

[29] ABOUZIED A, HELLERSTEIN J, SILBERSCHATZ A. Playful query specification with dataplay[J]. Proceedings of the Very Large Data Bases Endowment, 2012, 5(12): 1938-1941.

[30] QARABAQI B, RIEDEWALD M. User-driven refinement of imprecise queries[A]. ICDE[C]. 2014. 916-927.

[31] TRAN Q, CHAN CY, PARTHASARATHY S. Query by output[A]. SIGMOD[C]. 2009.535-548.

[32] SHOKOUHI M, SLOAN M, BENNETT PN, et al. Query suggestion and data fusion in contextual disambiguation[A]. WWW[C]. 2015. 971-980.

[33] GAO J, YUAN W, LI X, et al. Smoothing click through data for Web search ranking[A]. SIGIR[C]. 2009.355-362.

[34] GUO F, LIU C, KANNAN A, et al. Click chain model in Web search[A]. WWW[C]. 2009.11-20.

[35] AGICHTEIN E, BRILL E, DUMAIS S. Improving Web search ranking by incorporating user behavior information[A]. SIGIR[C]. 2006. 19-26.

[36] DROSOU M, PITOURA E. YmalDB: exploring relational databases via result-driven recommendations[J]. Proceedings of the Very Large Data Bases Endowment, 2013, 22(6):849-874.

[37] SCHMEIER S. Exploratory search on mobile devices[D]. German Research Center for Artificial Intelligence and Saarland University. 2013.

[38] PAPADAKOS P, TZITZIKAS Y. Hippalus: preference-enriched faceted exploration[A]. EDBT/ICDT Workshops[C]. 2014.167-172.

[39] TAUHEED F, HEINIS T, SCHURMANN F, et al. SCOUT: prefetching for latent structure following queries[J]. Proceedings of the Very Large Data Bases Endowment, 2012, 5(11): 1531-1542.

[40] SIDIROURGOS L, KERSTEN M L, BONCZ PA. Scientific discovery through weighted sampling[A]. Big Data Conference[C]. 2013. 300-306.

[41] SIDIROURGOS L, KERSTEN M L, BONCZ P A. SciBORQ: scientific data management with bounds on runtime and quality[A]. Biennial Conference on Innovative Data Systems Research (CIDR)[C]. 2011.296-301.

[42] ACHARYA S, GIBBONS P, POOSALA V, et al. The aqua approximate query answering system[A]. SIGMOD[C]. 1999.574-576.

[43] AGARWAL S, MILNER H, KLEINER A, et al. Knowing when you're wrong: building fast and reliable approximate query processing systems[A]. SIGMOD[C]. 2014.481-492.

[44] AGARWAL S, MOZAFARI B, PANDA A, et al. BlinkDB: queries with bounded errors and bounded response times on very large data[A]. EuroSys[C]. 2013.29-42.

[45] HOFFART J, SUCHANEK F, BERBERICH K, et al. YAGO2: exploring and querying world knowledge in time, space, context, and many languages[A]. WWW [C]. 2011. 229-232.

[46] RDF model and syntax specification[S]. 1999.

[47] DU F, CHEN Y G, DU X Y. Survey of RDF query processing techniques. Journal of Software, 2013, 24(6):1222-1242.

[48] MALEWICZ G, AUSTERN M, BIK A, et al. Pregel: a system for large-scale graph processing[A]. SIGMOD[C]. 2010.135-146.

[49] LOW Y C, GONZALEZ J, KYROLA A, et al. Distributed GraphLab: a framework for machine learning in the cloud[J]. Proceedings of the Very Large Data Bases Endowment, 2012, 5(8):716-727.

[50] GONZALEZ J E, XIN RS, DAVE A, et al. GraphX: graph processing in a distributed dataflow framework[A]. OSDI[C]. 2014.599-613.

[51] SHAO B, WANG H, LI Y. Trinity: a distributed graph engine on a memory cloud[A]. SIGMOD[C]. 2013.505-516.

[52] CHANG L, WANG ZW, M A T, et al. HAWQ: a massively parallel processing SQL engine in hadoop[A]. SIGMOD[C]. 2015. 1223-1234.

[53] LI J Z, GAO H, LUO J Z, et al. InfiniteDB: a pc-cluster based parallel massive database management system[A]. SIGMOD[C]. 2007. 899-909.

[54] Cloudera Impala[EB/OL]. http://www.cloudera.com/.

[55] DIACONU C, FREEDMAN C, ISMERT E, et al. Hekaton: SQL server‘s memory-optimized OLTP engine[A]. SIGMOD[C]. 2013. 1243-1254.

[56] SAP HANA[EB/OL]. http://www.saphana.com/.

[57] MonetDB[EB/OL]. http://www.monetdb.org/.

[58] ANTOVA L, EL-HELW A, SOLIMAN M, et al. Optimizing queries over partitioned tables in MPP systems[A]. SIGMOD[C]. 2014. 373-384.

[59] VALIANT L. A bridging model for parallel computation[J]. Communication on ACM, 1990, 33(8):103-111.

作者简介：

杜小勇（1963-），男，浙江衢州人，博士，中国人民大学教授、博士生导师，主要研究方向为智能信息检索、高性能数据库、知识工程。

陈峻（1991-），男，浙江温州人，中国人民大学博士生，主要研究方向为探索式搜索。

陈跃国（1976-），男，辽宁盖州人，博士，中国人民大学副教授、博士生导师，主要研究方向为大数据分析系统和语义搜索。