



# 一种适合多元异构图数据管理系统的交互分析框架<sup>\*</sup>

赵子豪<sup>1,2</sup> 沈志宏<sup>1</sup>

<sup>1</sup>(中国科学院计算机网络信息中心 北京 100190)

<sup>2</sup>(中国科学院大学 北京 100049)

**摘要:**【目的】提出一套交互分析框架,以屏蔽多元图数据模型、管理系统、查询接口协议的差异性,提供面向图数据的在线交互分析服务。【方法】对图数据管理系统的服务接口和前端分析需求进行抽象,设计形成开放可扩展的交互协议和总体框架,并实现数据适配、交互分析应用等软件模块。【结果】该框架有效屏蔽了 Neo4j、Jena 等系统的异构性,并满足图数据浏览、探索以及路径发现等交互分析需求,为图数据的相关应用提供了良好的基础。【局限】需要针对大规模数据进行优化和完善。【结论】本框架可有效实现图数据管理系统与分析应用的接口解耦,具有较大的应用价值。

**关键词:** 图数据 交互分析 分析框架

**分类号:** TP393

**DOI:** 10.11925/infotech.2096-3467.2019.0252

## 1 引言

图数据模型具有对实体关系的表达能力强、属性及结构可扩展性好、关联查询高效等优势。在对海量的多元异构信息进行建模时,图数据模型较关系模型、键值模型、文档模型等而言具有更好的直观效果,更有利于使用者对数据结构和语义关系的理解。此外,图数据模型还具有可推理、可深度挖掘的特点,因此在智能问答、金融风险预测等方面也得到较为广泛的应用。

图数据的管理、处理以及分析等技术发展已久,随着应用的不断深化和图数据生态的不断繁荣,图数据模型及图数据管理系统逐渐呈现出显著的多样性<sup>[1]</sup>。这种不同数据管理系统间的多元异构特征直接造成了

现行的图数据分析和工具服务的差异性,从而导致这类工具系统及其上层应用系统缺乏通用性。以图的可视化分析为例,目前主流分析工具往往局限于特定的图数据管理系统、数据格式以及协议,可移植性和推广泛化能力不足,缺少一套适用于多种图数据管理系统(如 Neo4j, Jena, Titan 等)的通用可视化分析工具。

为屏蔽不同图数据管理系统、数据模型和协议之间的差异,简化图数据交互分析操作的复杂度并提高上层服务应用的通用性,本文在分析现有图数据存储管理系统、查询协议和可视化分析工具的基础上,提出一种适合多元异构图数据管理系统的交互分析框架。

## 2 相关研究

图数据技术的研究涉及存储与管理、查询与检索、

通讯作者: 沈志宏, ORCID: 0000-0002-0113-0478, E-mail: bluejoe@cnic.cn。

<sup>\*</sup>本文系国家重点研发计划项目“科学大数据管理系统”(项目编号: 2016YFB1000605)和中国烟草总公司科技重大专项项目“烟草科研数据融合与关联挖掘关键技术研究”(项目编号: 110201801019(SJ-01))的研究成果之一。

计算与分析等方面。随着图数据研究热度的持续增高,相关工具的种类越来越多,功能越来越丰富。本文从图数据模型、图数据查询语言、图数据管理系统、图数据可视化分析工具等方面对现有技术系统及其差异性进行调研。

## 2.1 图数据管理系统及查询协议

属性图<sup>[2]</sup>是一种常用的图数据模型,通过节点和关系表达实体及其之间的关系。图数据模型的另外一个重要分支是资源描述框架(Resource Description Framework, RDF)语义数据描述模型<sup>[3]</sup>。属性图和 RDF 两种模型所采用的存储结构有所区别,属性图采用“顶点+边+属性”的方式进行存储, RDF 则采用“主语-谓语-宾语”三元组的方式存储属性和链接<sup>[4]</sup>。

目前已存在很多图数据管理系统。基于属性图模型的管理系统如 Neo4j<sup>[5]</sup>、Titan<sup>[6]</sup>、Cosmos DB<sup>[7]</sup>、InfiniteGraph<sup>[8]</sup>等;基于 RDF 数据模型的管理系统如 Jena<sup>[9]</sup>、Virtuoso<sup>[10]</sup>、AllegroGraph<sup>[11]</sup>等;另有多模型的图数据管理系统,如 Amazon 的 Neptune<sup>[12]</sup>,在初始化

时可以选择属性图和 RDF 中的一种。

不同的图数据管理系统支持的查询语言协议存在较大的差异性。与关系型数据库具有统一的 SQL 查询语言不同,图数据查询语言门类很多。如: Liu 等提出了一门用于复杂图的查询语言<sup>[13]</sup>。GraphQL<sup>[14]</sup>是一种基于形式化语法构建和操作图结构的查询语言,可以使用简单图组合的形式描述复杂图模式。函数式数据流语言 Gremlin<sup>[15]</sup>面向路径,对传统图查询语言进行扩展,在多种图数据库和分析引擎上得到了支持。Cypher<sup>[16]</sup>是一种声明型语言,其设计思想是用简单的语句描述图中的模式,用模式匹配的方法在数据库中查询匹配的数据。Oracle 提出的 PGQL<sup>[17]</sup>语法与 SQL 相似,支持图模式匹配查询、规则路径查询。在 RDF 数据查询方面,经历了 TRIPLE<sup>[18]</sup>、RQL<sup>[19]</sup>、SeRQL<sup>[20]</sup>等语言之后, W3C 发布了 SPARQL<sup>[21]</sup>查询语言,并将其作为推荐的 RDF 标准查询语言。

笔者对主流图数据查询语言及图数据管理系统对语言协议和数据模型的支持情况做分析对比,如表 1 和表 2 所示。

表 1 主流图数据查询语言及适用模型总结

查询语言	适用模型	特性对比
TRIPLE	RDF 模型	仅支持简单图模式的查询,学习成本高。
RQL	RDF 模型	引入聚合操作,与 SQL 语法较为相似,学习成本稍高。
SeRQL	RDF 模型	与 SPARQL 标准较为接近,学习成本低。
SPARQL	RDF 模型	被 W3C 推荐为标准,语法与 SQL 相似,学习成本低。
GraphQL	属性图模型	API 型查询语言,灵活性好,但是学习成本较高。
Gremlin	属性图模型	图灵完备,类似于编程语言,较灵活,但学习成本高;主要用于遍历。
Cypher	属性图模型	语法与 SQL 相似,比较成熟,学习成本低。
PGQL	属性图模型	语法与 SQL 相似,学习成本较低,但目前使用者不多。

表 2 图数据管理系统支持语言对比

图数据管理系统	数据模型	支持查询语言	特性对比
Neo4j	属性图模型	Cypher, Gremlin	很成熟,生态良好,但不支持数据分片。
Titan	属性图模型	Gremlin	可挂载 HBase 等存储后端,支持数据分片。
InfiniteGraph	属性图模型	Gremlin	支持数据分片,免费版只支持 100 万节点。
Cosmos DB	属性图模型	Gremlin	基于云平台,支持数据分片,但不开源。
AllegroGraph	RDF 模型	SPARQL	支持数据分片,免费版支持 5 000 万三元组。
Jena	RDF 模型	SPARQL	完全开源,比较成熟,开发使用便捷。
Virtuoso	RDF 模型	SPARQL	基于关系表实现 RDF 管理,在大规模数据下性能不足。
Neptune	属性图/RDF	Gremlin/SPARQL	基于云平台,支持数据分片,但不开源。

## 2.2 图数据可视化分析工具

Palmer 等证明,图比其他可视化展示形式更适合

探索数据的内部关系<sup>[22]</sup>,图的可视化分析包括图数据的基本统计、检索与映射、关系发现、结果展示等任

务, 现有的可视化分析工具在支持的数据源、使用的接口等方面存在差异。

RelFinder<sup>[23]</sup>可以实现实体间关系的交互式发现, 可以将任何支持 SPARQL 的 RDF 数据集作为数据源。开源的可视化分析工具 Gephi<sup>[24]</sup>支持以 GDF、GraphML、GML、NET、GEXF 等格式文件为数据源导入数据, 并支持多种统计功能以分析数据。

Bloom<sup>[25]</sup>、WebVOWL<sup>[26]</sup>、PGV(Paged Graph Visualization)<sup>[27]</sup>等可视化工具提供了可交互的图数据展示功能。Bloom 是 Neo4j 提供的交互式图数据可视化工具。WebVOWL 用于 RDF 数据的可视化, 可以对 5MB 以内的 RDF 文件进行绘制。Sayers 提出的以节点为中心的 RDF

图可视化工具(Node-centric RDF Graph Visualization)<sup>[28]</sup>以树的形式展示 RDF 数据, 但只能显示某个节点的两层祖先和孩子节点, 对于大度节点的支持效果十分有限。PGV 支持大度节点的展示, 将大度节点的邻居节点显示在其周围, 形成一个环。PGV 不显示全部图数据, 当用户在同一个方向上探索图时, 视图的可读性会变差。

此外, 还有一些 JavaScript 库, 如 Vis.js<sup>[29]</sup>和 Alchemy.js<sup>[30]</sup>, 基于浏览器为图数据分析提供了基础的可视化功能。

针对上述图数据可视化和分析工具, 在其支持的数据源和分析能力、灵活性等方面进行总结, 结果如表 3 所示。

表 3 图数据可视化和分析工具对比

图谱可视化分析系统	支持数据源	特性对比
RelFinder	支持 SPARQL 的 RDF 数据集	有一定分析能力, 但仅支持 RDF 数据的可视化。
Gephi	CSV、GML 等文件格式	分析能力较强, 与主流数据管理系统融合不够。
Bloom	Neo4j	Neo4j 提供的可视化工具, 需要与 Cypher 语言结合实现分析功能。
Vis.js	Gephi/DOT 语言	JavaScript 库, 部署方便; 无法显示大规模的图数据, 图分析功能不足。
Alchemy.js	JSON	JavaScript 库, 部署方便, 仅提供节点和边的可视化。
Node-centric RDF Graph Visualization	RDF	仅支持 RDF 数据, 对大度节点显示效果不好。
PGV	RDF	仅支持 RDF 数据, 环状显示节点, 分析功能不足。
WebVOWL	RDF	仅支持 5MB 以内的 RDF 数据文件。

通用协议作用示意图如图 1 所示。现有的图数据可视化分析工具与底层的图数据管理系统及其服务接口存在较强的耦合性(如图 1(a)所示), 尚缺少对 Neo4j、Titan、Jena 等异构系统的统一支持, 同时它们在交互分析的效果上也存在一些局限。基于此, 本文设计一套开放可扩展的通用图数据交互协议 InteractiveGraphProtocol(简称 IGP), 并基于该协议设计软件框架 InteractiveGraph, 屏蔽可视化分析工具底层数据管理系统及数据结构上的差异, 其原理如图 1(b)所示。

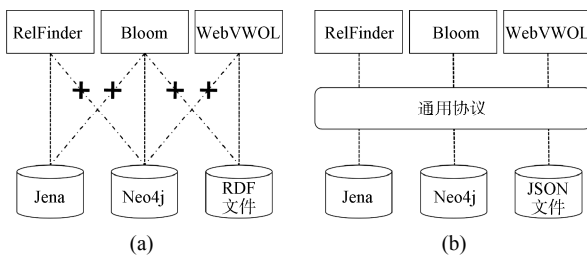


图 1 通用协议作用示意图

### 3 交互协议与框架设计

交互分析框架 InteractiveGraph 的核心概念是交互协议 IGP, InteractiveGraph 框架将前端的可视化需求转换为 IGP 的数据请求(Request), 在数据服务端通过软件模块将请求转义为对应图数据管理系统的图查询语句或者服务调用, 并将查询的结果封装为 IGP 的数据响应(Response), 从而实现对不同图数据管理系统在查询语言、数据模型、服务结构等方面差异的屏蔽, 增强交互分析应用的可扩展性和灵活性, 结构如图 2 所示。

#### 3.1 图谱在线分析交互协议 InteractiveGraphProtocol

IGP 协议用于定义图谱分析应用和数据服务端之间的交互协议和数据格式。它定义了连接类、浏览类、探索类、实体匹配类、路径查询类等 5 类接口, 这 5 类接口的功能及输入输出等详细信息如表 4 所示。

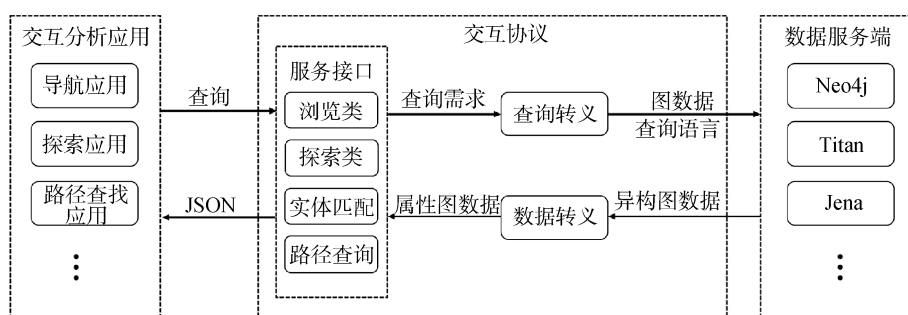


图 2 InteractiveGraph 结构示意图

表 4 接口类别、功能及输入输出信息

接口类别	接口名称	功能描述	输入	输出
1 连接类	1.1 Connect	连接当前数据源并初始化	无	当前数据源的元数据(规模、社区等)
2 浏览类	2.1 LoadGraph	加载当前图的信息	当前位置	图中全部顶点和边
	2.2 GetCommunityData	加载挖掘社区的信息	无	社区的轮廓及包含的顶点
	2.3 GetNodesInfo	获取节点的描述信息	节点 id	节点描述信息
	2.4 GetNodeCategories	获取节点的类别(label)	无	类别及其描述
3 探索类	3.1 GetNeighbours	获取节点邻边及邻居节点	节点 id	节点邻边及邻居节点
4 实体匹配类	4.1 FilterNodesByCategory	根据节点类别过滤节点集	节点 id 数组、指定类别	获取节点数组中属于指定类别的节点
	4.2 Search	以关键词和限制数为条件搜索节点	关键词、限制数	指定字段中包含关键词的节点, 输出节点数不超过限制数
5 路径查询类	5.1 FindRelations	在起止节点之间查找不超过最大深度的路径	起止节点 id、最大深度	查询任务 id
	5.2 GetMoreRelations	从路径查询任务的缓冲中取出更多结果	查询任务 id	查询得到的路径
	5.3 StopFindRelations	停止查询任务	查询任务 id	查询任务 id 及状态

以路径查询类为例, 路径查询类接口共有三个, 分别是查找路径的接口 FindRelations, 获取路径的接口 GetMoreRelations, 停止查询任务的接口 StopFindRelations。

(1) FindRelations 传递的参数是起始节点 id、终止节点 id 和最大路径深度。最大路径深度指从起始节点到终止节点的跳数, 若两个节点直接邻接, 则跳数为 1。FindRelations 的功能是在服务端启动一个路径查询的任务, 并返回该查询任务的任务 id, 用于后续定位查询任务。

(2) GetMoreRelations 接口传递的参数是查询任务的 id, 它的功能是从服务端获取查询任务的相关信息, 包括一个查询任务是否完成的标志位, 以及查询得到的起止节点之间符合要求的路径。

(3) StopFindRelations 传递的参数是查询任务的 id, 它的功能是通知服务端停止该 id 所代表的查询任

务, 并返回任务 id 及一个表示任务是否停止的标志位。当可视化应用需要查询某两个节点之间的路径关系时, 先通过查找路径接口在服务端开启查询任务并获取任务 id, 然后通过获取路径接口, 按任务 id 从服务端获取该查询任务得到的路径, 最终通过停止查询任务接口停止服务端的查询任务。路径查询类接口的工作流程如图 3 所示。

### 3.2 交互分析框架 InteractiveGraph

在提出 IGP 协议的基础上, 本文设计了交互分析框架 InteractiveGraph, 框架结构如图 4 所示。

(1) 数据层: 提供图数据的存储和查询服务, 该层包含异构的数据管理系统, 如: Neo4j、Titan、Jena 等。

(2) 服务协调层: 通过适配器适配不同的数据源, 管理离线查询和数据分析任务, 同时支持通过图算法对映射后得到的数据进行处理和挖掘。

(3) 应用协调层: 提供一个支持可视化分析应用



表 5 项目托管地址列表

项目名称	托管地址	开发语言
InteractiveGraphServer-Neo4j 适配器	<a href="https://github.com/grapheco/InteractiveGraph-neo4j">https://github.com/grapheco/InteractiveGraph-neo4j</a>	Scala
InteractiveGraphServer-RDF 适配器	<a href="https://github.com/grapheco/InteractiveGraph-RDF">https://github.com/grapheco/InteractiveGraph-RDF</a>	Java、Scala
AppFrame 及应用	<a href="https://github.com/grapheco/InteractiveGraph">https://github.com/grapheco/InteractiveGraph</a>	Typescript

#### 4.1 Neo4j 适配器设计与实现

InteractiveGraphServer-Neo4j 基于 Neo4j 数据库后端实现从属性图数据到 IGP 数据格式的适配。Neo4j 适配端支持三种数据源: JSON 文件、Neo4j 的数据库以及通过 Bolt 协议连接远程的 Neo4j 服务器。在服务端的实现中,这三者之间存在继承关系。如果使用 JSON 文件作为数据源,服务端将会开启一个新的 Neo4j 数据库,将 JSON 文件中的数据导入到数据库中,而数据库服务又继承了 Bolt 服务,最终都通过 Bolt 协议在指定端口上对服务端提供图数据的存储和检索服务。

服务端从各个接口中获取查询参数后,根据接口功能将参数转义为对应的 Cypher 语句。如在查询路径的接口中,获取到的参数为起始节点 id-startNodeId、终止节点 id-endNodeId 和最大深度 maxDepth,其意义为查询起始节点和终止节点之间不超过指定最大深度 maxDepth(跳数)的路径,则转义后得到的 Cypher 语句为“start m=node(\$startNodeId), n=node(\$endNodeId) match p=(m)-[\*1..\$maxDepth]-(n) RETURN p”。

如果数据源中存在符合查询条件的数据,则本条查询得到的结果是 Neo4j 中的路径,服务端从该路径中取出节点和边,转换为 InteractiveGraph 中的格式,以 JSON 字符串的形式通过接口送回给可视化应用。

InteractiveGraphServer-Neo4j 中节点转换的方法是:从节点中取出 id,作为属性图中节点的 id,对于其他属性,将属性键和属性值一一对应地以字符串 Map 的格式转换到属性图中。关系转换的方法是:从 Neo4j 中属性图模型的关系中提取关系 id、关系类型、起始节点 id、终止节点 id,并实现到 id、label、from、to 这 4 个字段的映射。

#### 4.2 RDF 适配器设计与实现

InteractiveGraphServer-RDF 适配 Jena 框架,支持 N3 文件、三元组数据库 Jena-TDB、远程数据服务器

Jena-Fuseki 三种数据源。

当 InteractiveGraphServer-RDF 从 IGP 接口获得查询涉及的参数进行对应的查询操作时,首先对节点 id“URI 化”。在 id 前附加预设的前缀,将节点 id 转化为 URI 字符串,然后根据接口功能,将查询转义为对应的 SPARQL 语言,在数据管理工具中通过 SPARQL 语句检索以该 URI 为主语的全部三元组。如在某数据集中,节点的前缀是“<http://interactivegraph.org/data/node/>”,服务端从 GetNodesInfo 中获取到的参数 id 为 3779,其含义为查询 id 为 3779 的节点的全部属性信息,则首先将该 id 转化为 URI:“<http://i nteractivegraph.org/data/node/3779>”,进一步得到的 SPARQL 语句为:

```
SELECT ?p ?o
WHERE {<http://interactivegraph.org/data/node/3779> ?p ?o }
```

从数据管理工具中检索出的数据是全部以该 URI 为主语的三元组的谓语和宾语。

将 RDF 数据转换为属性图的操作为:针对某个表示节点属性的三元组,将主语“去 URI 化”,即去掉预设的前缀,作为属性图中节点的 id。将谓语“去 URI 化”,作为属性图中该节点的属性键,将三元组的宾语作为属性值。框架内 RDF 数据到属性图数据的适配如图 5 所示。

执行深度为  $m$  的路径查询,由于节点间边方向的不同,需要借助中间节点,对于每一对相邻节点,其间的边都有两个方向。在深度为  $m$  的路径中,相邻节点共有  $m$  对。每条查询语句实际上是对  $m$  对相邻节点间边方向的一种描述。

深度为  $m$  的路径查询需要  $2^m$  条 SPARQL 查询语句。最大深度为  $m$  的路径查询,实际上是从深度为 1 的路径到深度为  $m$  的路径的并集。对于最大深度为  $m$  的路径查询,计算方法如公式(1)所示,共需要  $2^{m+1}-2$  条 SPARQL 语句。

$$2^1 + 2^2 + 2^3 + \dots + 2^m = 2^{m+1} - 2 \quad (1)$$



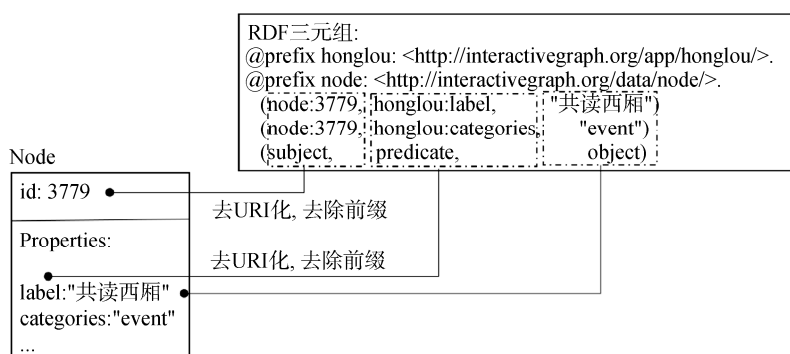


图 5 RDF 数据到属性图的适配示意图

### 4.3 交互分析应用的设计与实现

基于 AppFrame, 本文已成功开发了 GraphNavigator、GraphExplorer 和 GraphRelFinder 等三个应用。

(1) 图谱交互式导航 GraphNavigator: 图谱交互式

导航是用于全图展示的应用, 其主要功能是向服务端请求指定数据源中的全部数据并将图数据以点和边的形式在屏幕上进行展示。图谱交互式导航效果如图 6 所示。

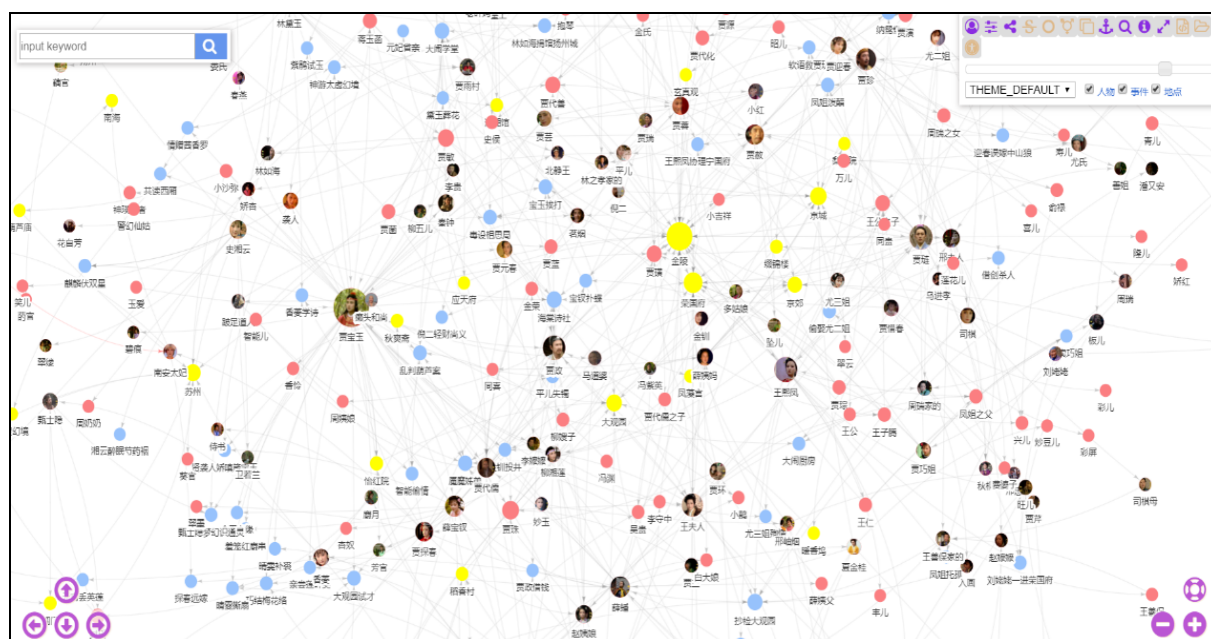


图 6 图谱交互式导航效果

(2) 图谱步进式探索 GraphExplorer: 用于从某个节点开始的图浏览, 采用交互步进式的方式解决了大图无法在画布上一次性加载展示的问题。用户通过搜索选择一个主节点进行展示。可以展示这个主节点的全部属性信息, 以及其邻边和相邻节点及属性, 更进一步地, 可以获取主节点的相邻节点的邻边和相邻节点及属性进行展示。图谱步进式探索效果如图 7 所示。

(3) 图谱关系发现 RelFinder: 用以在大图中寻找感兴趣的关系路径。用户通过搜索确定起始节点和终

止节点, 并指定最大路径深度。应用通过接口向服务端发起关系查找请求, 并对得到的路径进行展示。路径查找的耗时较大, 因此可以在关系查找器中发送请求停止服务端的查询任务。图谱关系发现效果如图 8 所示。

得益于 IGP 对不同数据源差异的屏蔽, Interactive Graph 框架内的可视化应用可以很方便地切换数据源, 并支持使用文件和数据管理系统两种数据源。针对远程数据服务器和 JSON 文件的配置如图 9 所示。

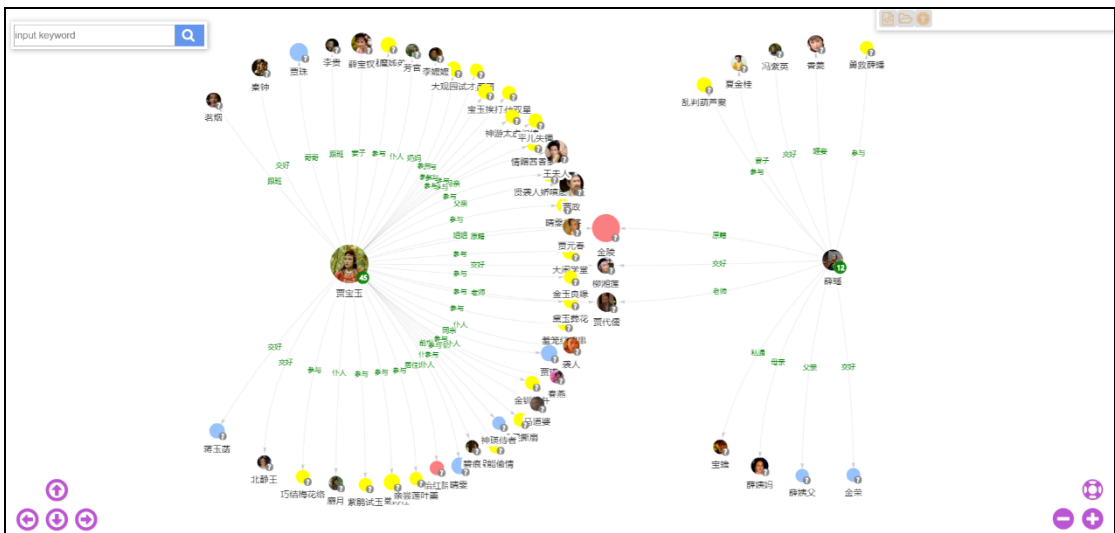


图 7 图谱步进式探索效果

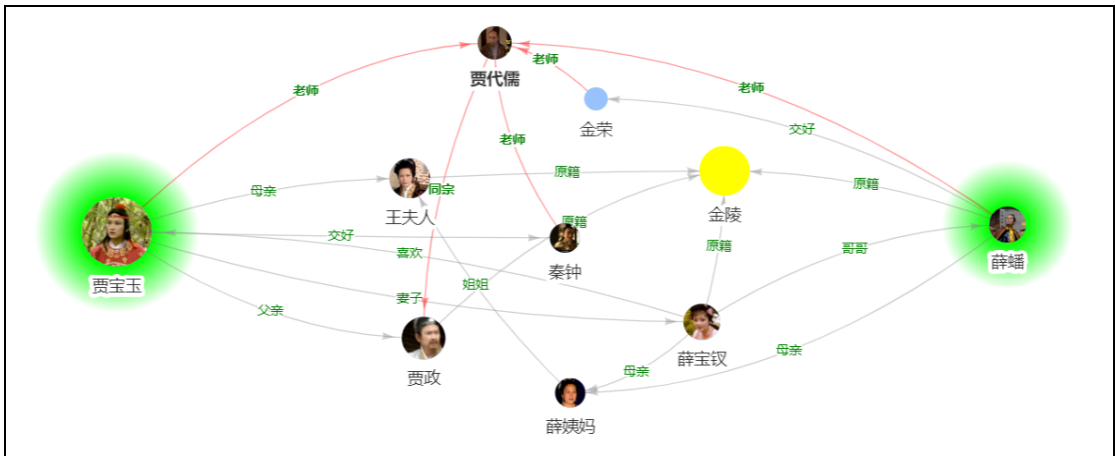


图 8 图谱关系发现效果

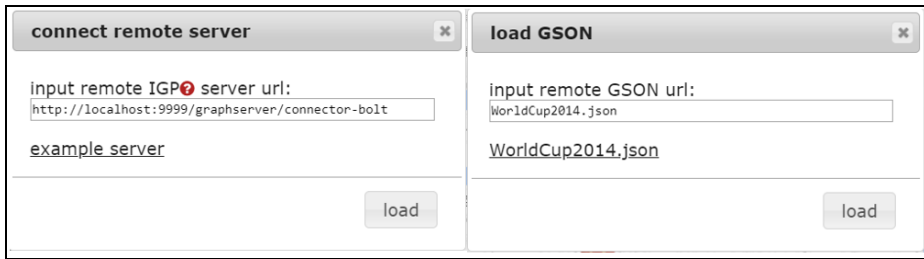


图 9 InteractiveGraph 应用中数据源的配置

## 5 结 语

InteractiveGraph 已经应用于中国烟草总公司科技重大专项项目“烟草科研数据融合与关联挖掘关键技术研究”。烟草科技知识图谱服务平台采集了烟草领域的 646 119 个科研人员、22 199 个科研

项目、16 111 个科研成果。其中, 通过 GraphExplorer 实现了对烟草科技知识图谱的定向浏览发现; 通过 RelFinder 实现了科研人员、科研机构之间的学术合作路径的发现。RelFinder 在烟草知识图谱上的应用效果展示如图 10 所示(敏感信息已做模糊处理)。



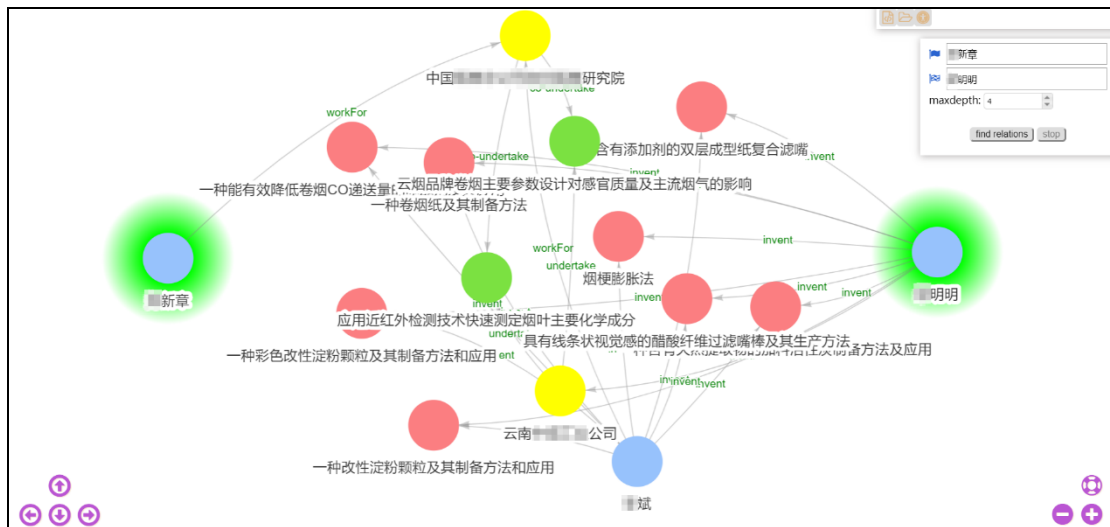


图 10 RelFinder 在烟草知识图谱上的应用效果

同时 InteractiveGraph 也应用于微生物知识图谱的服务平台, 通过对亿级规模 RDF 数据库的适配, 快速搭建了面向“物种”-“基因”-“蛋白质”-“酶”的关联路径发现应用与可视化展示环境。

实践证明, InteractiveGraph 具有较好的通用性, 可以有效屏蔽 Neo4j 和 Jena 在数据格式和功能接口上的差异, 解决了异构知识图谱系统的在线交互式可视化需求, 具有一定的实际意义和推广性。

InteractiveGraph 还存在一些不足, 如在大规模数据的可视化展示性能、RDF 数据的路径查询效率以及针对大规模数据的相关处理等方面仍需提升。另外, InteractiveGraph 的交互分析应用和数据服务端需要考虑扩展新的应用和数据后端。未来将在这些方面继续完善和优化 InteractiveGraph。

### 参考文献:

- [1] Angles R, Gutierrez C. An Introduction to Graph Data Management: Fundamental Issues and Recent Developments[A]// Fletcher G, Hidders J, Larriba-Pey J L, et al. Graph Data Management[M]. Springer International Publishing, 2018.
- [2] What is a Graph Database? [EB/OL]. [2018-12-05]. <http://Neo4j.com/developer/graph-database/>.
- [3] RDF Model and Syntax Specification[EB/OL]. [2018-12-02]. <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/>.
- [4] Alocci D, Mariethoz J, Horlacher O, et al. Property Graph vs RDF Triple Store: A Comparison on Glycan Substructure Search[J]. PLoS One, 2015, 10(12): e0144578.
- [5] Neo4j Open Source NoSQL Graph Database[EB/OL]. [2018-12-08]. <http://Neo4j.org/>.
- [6] Titan-Distributed Graph Database[EB/OL]. [2018-12-12]. <http://titan.thinkaurelius.com/>.
- [7] Microsoft Azure Cosmos DB[EB/OL]. [2019-01-11]. <https://docs.microsoft.com/zh-cn/azure/cosmos-db/graph-introduction>.
- [8] InfiniteGraph[EB/OL]. [2019-01-15]. <https://www.objectivity.com/products/infinitegraph/>.
- [9] Carroll J, Dickinson I, Dollin C, et al. Jena: Implementing the Semantic Web Recommendations[C]// Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters. ACM, 2004: 74-83.
- [10] Erling O, Mikhailov I. RDF Support in the Virtuoso DBMS[A]// Pellegrini T, Auer S, Tochtermann K, et al. Networked Knowledge-Networked Media[M]. Springer, 2009.
- [11] AllegroGraph-Semantic Graph Database[EB/OL]. [2019-01-11]. <https://allegrograph.com/products/allegrograph/>.
- [12] Amazon Neptune-Fast, Reliable Graph Database Build for Cloud[EB/OL]. [2019-01-11]. <https://aws.amazon.com/neptune/>.
- [13] Liu Y A, Stoller S D. Querying Complex Graphs[C]// Proceedings of the 8th International Symposium on Practical Aspects of Declarative Languages. Springer, 2006: 199-214.
- [14] GraphQL[EB/OL]. [2019-01-16]. <http://graphql.cn/>.
- [15] Rodriguez M A. The Gremlin Graph Traversal Machine and Language (Invited Talk)[C]// Proceedings of the 15th Symposium on Database Programming Languages. ACM, 2015: 1-10.
- [16] Francis N, Green A, Guagliardo P, et al. Cypher: An Evolving Query Language for Property Graphs[C]// Proceedings of the 2018 International Conference on Management of Data. ACM, 2018: 1433-1445.
- [17] van Rest O, Hong S, Kim J, et al. PGQL: A Property Graph

Query Language[C]// Proceedings of the 4th International Workshop on Graph Data Management Experiences & Systems. 2016: Article No.7.

- [18] Sintek M, Decker S. TRIPLE—A Query, Inference, and Transformation Language for the Semantic Web[C]// Proceedings of the 1st International Semantic Web Conference. 2002: 364-378.
- [19] Karvounarakis G, Alexaki S, Christophides V, et al. RQL: A Declarative Query Language for RDF[C]// Proceedings of the 11th International Conference on World Wide Web. 2002: 592-603.
- [20] Broekstra J, Kampman A. SeRQL: An RDF Query and Transformation, Language DRAFT[C]// Proceedings of the 3rd International Semantic Web Conference. 2004.
- [21] SPARQL 1.1 Query Language [EB/OL]. [2018-12-18]. <https://www.w3.org/TR/sparql11-query/>.
- [22] Palmer S, Rock I. Rethinking Perceptual Organization: The Role of Uniform Connectedness[J]. Psychonomic Bulletin & Review, 1994, 1(1):29-55.
- [23] Heim P, Hellmann S, Lehmann J, et al. RelFinder: Revealing Relationships in RDF Knowledge Bases[C]// Proceedings of the 4th International Conference on Semantic and Digital Media Technologies. 2009: 182-187.
- [24] The Open Graph Viz Platform[EB/OL]. [2019-01-03]. <https://gephi.org/>.
- [25] Graph Visualization with Neo4j[EB/OL]. [2018-12-25]. <https://Neo4j.com/graph-visualization-Neo4j/>.
- [26] Lohmann S, Link V, Marbach E, et al. WebVOWL:

Web-based Visualization of Ontologies[C]// Proceedings of the 2014 International Conference on Knowledge Engineering and Knowledge Management. 2014: 154-158.

- [27] Deligiannidis L, Kochut K J, Sheth A P. RDF Data Exploration and Visualization[C]// Proceedings of the ACM 1st Workshop on CyberInfrastructure: Information Management in eScience. ACM, 2007: 39-46.
- [28] Sayers C. Node-Centric RDF Graph Visualization[J]. Mobile and Media Systems Laboratory, HP Labs, 2004.
- [29] Vis.js[EB/OL]. [2018-12-30]. <http://visjs.org/>.
- [30] Alchemy.js[EB/OL]. [2019-01-18]. <https://github.com/GraphAlchemist/Alchemy>.

### 作者贡献声明:

赵子豪: 设计研究方案, 进行实验, 撰写论文主体内容; 完成 InteractiveGraph 框架中 RDF 适配器的开发;  
沈志宏: 提出研究思路, 设计研究方案, 论文最终版本修订; InteractiveGraph 开源项目的发起人, 完成核心框架和主体功能的开发。

### 利益冲突声明:

所有作者声明不存在利益冲突关系。

收稿日期: 2019-03-05  
收修改稿日期: 2019-04-16

## An Interactive Analysis Framework for Multivariate Heterogeneous Graph Data Management System

Zhao Zihao<sup>1,2</sup> Shen Zhihong<sup>1</sup>

<sup>1</sup>(Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190, China)

<sup>2</sup>(University of Chinese Academy of Sciences, Beijing 100049, China)

**Abstract:** [Objective] An open and scalable interactive analysis framework is proposed to shield the differences between multivariate graph data models, management systems, interfaces and protocols, and supply the online interactive analyzing service faced with graph data. [Methods] By abstracting the multi-analysis requirements and heterogeneous service interfaces, an open, scalable and interactive protocol is designed. Based on the protocol, an interactive framework is designed to implement the interactive module. [Results] This interactive analysis framework is well abstracted, shields the heterogeneity of graph management systems like Neo4j and Jena effectively, and provides a good foundation for front-end applications. [Limitations] Need to be optimized and adjusted on large-scale data. [Conclusions] The interactive analysis framework of heterogeneous knowledge graph has practical significance and deserves promotion.

**Keywords:** Graph Data Interactive Analysis Interactive Framework