



回顧

3.1

知识与知识表示的概念

3.2

产生式表示法

3.3

框架表示法

3.4

状态空间表示法





中國石油大學(华东)
CHINA UNIVERSITY OF PETROLEUM

计算机科学与技术学院

第四章 知识图谱

人工智能课程组

智能科学系





目录

CONTENTS

4.1

知识图谱

4.2

本体知识表示

4.3

万维网知识表示

4.4

知识图谱构建

4.5

知识图谱的现状与发展



4.1 知识图谱

■ 知识图谱 (Knowledge Graph)

- 用可视化技术描述知识资源及其载体，挖掘、分析、构建、绘制和显示知识及它们之间的相互联系。
- 本质上，知识图谱是一种揭示实体之间关系的语义网络，可以对现实世界的事物及其相互关系进行形式化地描述。
- 例如，Google用于增强其搜索引擎功能的知识库。
- 现在的知识图谱已被用来泛指各种大规模的知识库。

4.1 知识图谱

■ Google Knowledge Graph

- 2012年5月16日，Google发布了知识图谱，他们应用知识图谱技术极大提升了Google搜索引擎返回结果的价值，从而掀起了一场知识图谱技术的热潮

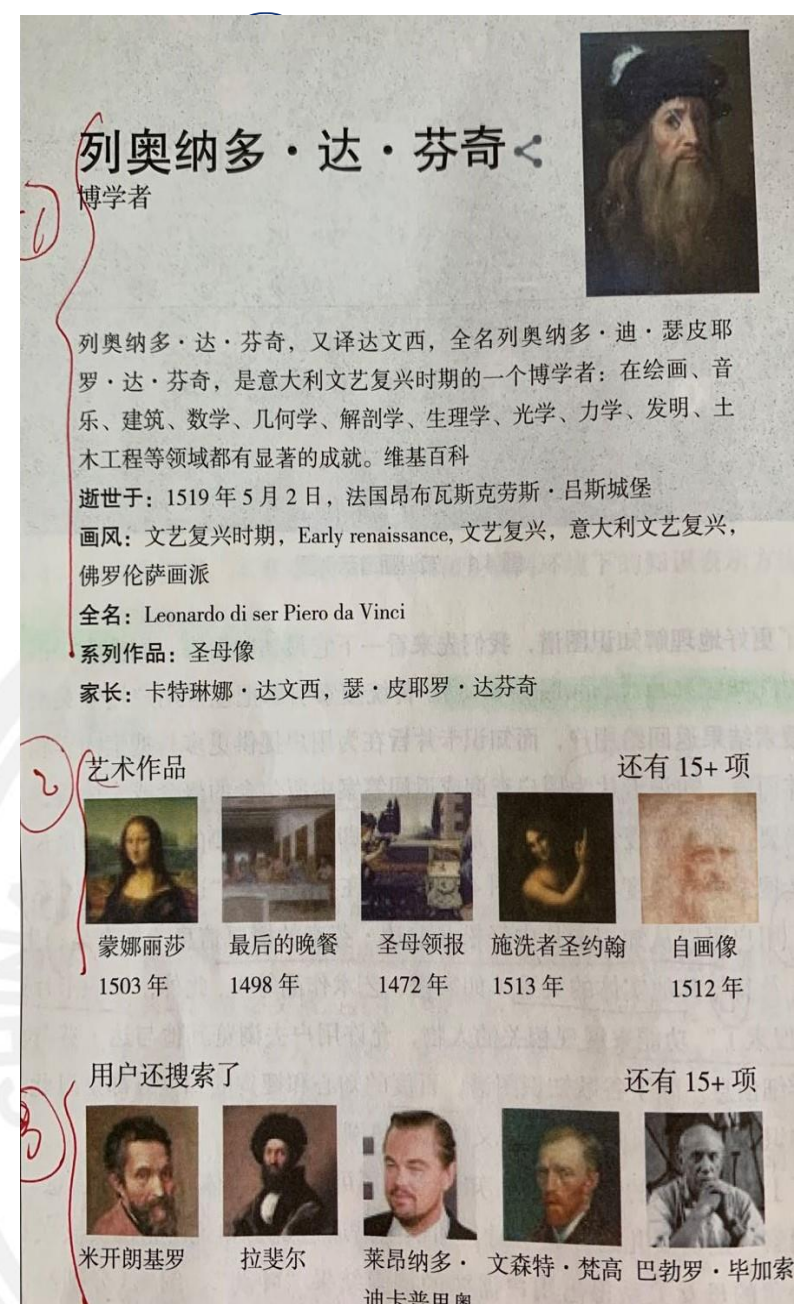


知识图谱示例图

4.1 知识图谱

■ 知识图谱的使用实例1

- 知识卡片（知识图谱的最初形态）：为用户查询或者返回答案中所包含的概念或者实体，所提供的详细的结构化摘要。
- 搜索效果的三方面的提升：
 1. 达芬奇的概要信息，例如全名，生卒年、画风
 2. 与其他实体的关系，例如他的家谱，他的艺术作品等
 3. 通过“用户还搜索了”这个功能来展现相关的人物信息。



知识卡片（图片来源于google）

4.1 知识图谱

■ 知识图谱的使用实例2

• 展示实体间的关系

1. 与搜索相关实体间的关系：姚沁蕾/母亲、
配偶：姚明
2. 搜索推理逻辑：姚明的女儿是姚沁蕾，
姚沁蕾的母亲是叶莉。

姚沁蕾 / 母亲

叶莉



叶莉，上海人，中国女子篮球运动员，曾经效力上海东方八爪鱼队，位置是中锋，她身高 1.93 米。叶莉 1996 年进入上海体育运动技术学院，1998 年入伍国家青年队，1999 年底进国家集训队。生涯最好的成绩是 2004 年代表中国国家女子篮球队夺得亚洲锦标赛冠军。维基百科
生于：1981 年 11 月 20 日（36 岁），中华人民共和国
身高：193 厘米
配偶：姚明（结婚时间：2007 年）
子女：姚沁蕾

用户还搜索了

还有 5+ 项



姚明 陆菲菲 孙明明 赵蕊蕊 王治郅 易建联

丈夫

姚明女儿的妈妈

叶莉

叶莉，1981 年 11 月 20 日出生于上海市，是一位中国篮球运动员，司职中锋，曾经效力上海东方八爪鱼队，1998 年，叶莉入选国家青年队，1999 年，叶莉第一次入选国家队。2002 年 5 月，叶莉被重新召回了国家队。2003 年 8 月 30 日，…详情>>

推理说明：姚明的女儿是姚沁蕾。姚沁蕾的母亲是叶莉。

搜狗立知 | 反馈

关系搜索中知识卡片示例（上图google，下图搜狗）

4.1 知识图谱

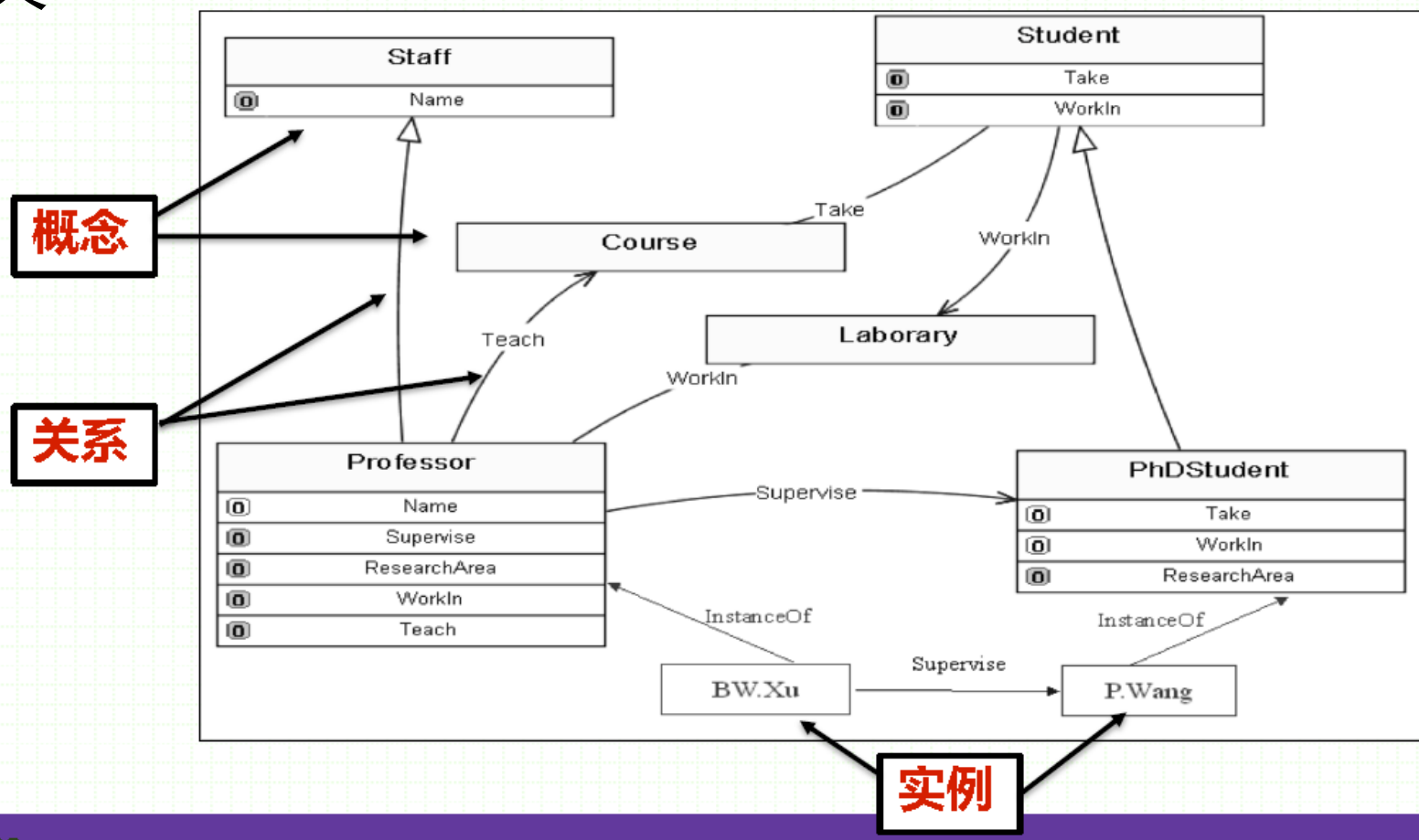
- 知识图谱旨在描述真实世界中存在的各种实体、概念和它们之间的关系。
- 每个实体或概念用一个全局唯一确定的ID来标识，称为它们的标识符(identifier)
- 每个 属性-值 对(attribute-value pair, 又称AVP)用来刻画实体的内在特性。
- 关系(relation)用来连接两个实体，刻画它们之间的关联。
- 知识图谱亦可被看作是一张巨大的图，图中的节点表示实体或概念，而图中的边则由属性或关系构成。
- 这些图模型可用W3C提出的资源描述框架RDF或属性图(property graph)来表示。

4.2 本体知识表示

A formal explicit specification of a shared conceptualization

■ 本体(Ontology)的定义

- 作为一种知识表示方法，本体表达了概念的结构、概念之间的关系等。
- 领域共享知识的描述。
- 本体通常由概念、实例、关系组成，有的还包含公理。



4.2 本体知识表示

本体是语义Web、语义搜索、知识工程和很多人工智能应用的基础，其特征：

- **概念化**：是指对世界上某些现象的一种抽象模型，对于事物的描述表示成一组概念；
- **形式化**：意味着本体是可以被计算机理解的；
- **精确性**：意味着概念以及概念被使用的约束要被精确的定义；
- **共享性**：指本体中体现的知识是共同认可的，反映在领域中公认的术语集合，有效促进知识共享。

4.2 本体知识表示

本体的组成：

- **概念**（类）：描述领域内的实际概念，既可以是实际存在的事物，也可以是抽象的概念，如大学、电影、人等；
- **关系**（属性）：用于描述类（概念）之间的关系，如part-of、kind-of等；
- **函数**：函数是一类特殊的关系，在这种关系中前n-1个元素可以唯一决定第n个元素，如mother-of (x, y) 表示y是x的母亲，x可以唯一确定它的母亲y；
- **公理**：公理代表领域内存在的事实，即领域规则，可以对领域内类或者关系进行约束，如教授属于教师的范围；
- **实例**：表示具体某个类的实际存在，如中国石油大学是大学的一个实例。

4.3 万维网知识表示

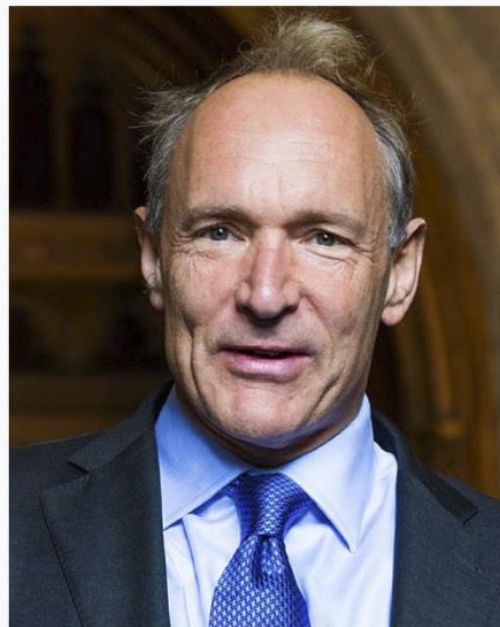
- 超文本是用超链接的方法，将各种不同空间的文字信息组织在一起的网状文本。
- 万维网（WWW）：将超文本嫁接到因特网

*The **World Wide Web** (abbreviated WWW or the Web) is an information space where **documents** and other web resources are identified by Uniform Resource Locators (URLs), interlinked by **hypertext links**, and can be accessed via the Internet. English scientist **Tim Berners-Lee** invented the World Wide Web in **1989**. He wrote the first web browser computer program in 1990 while employed at **CERN** in Switzerland. The Web browser was released outside CERN in 1991, first to other research institutions starting in January 1991 and to the general public on the Internet in August 1991.*

Web=文档+超链接

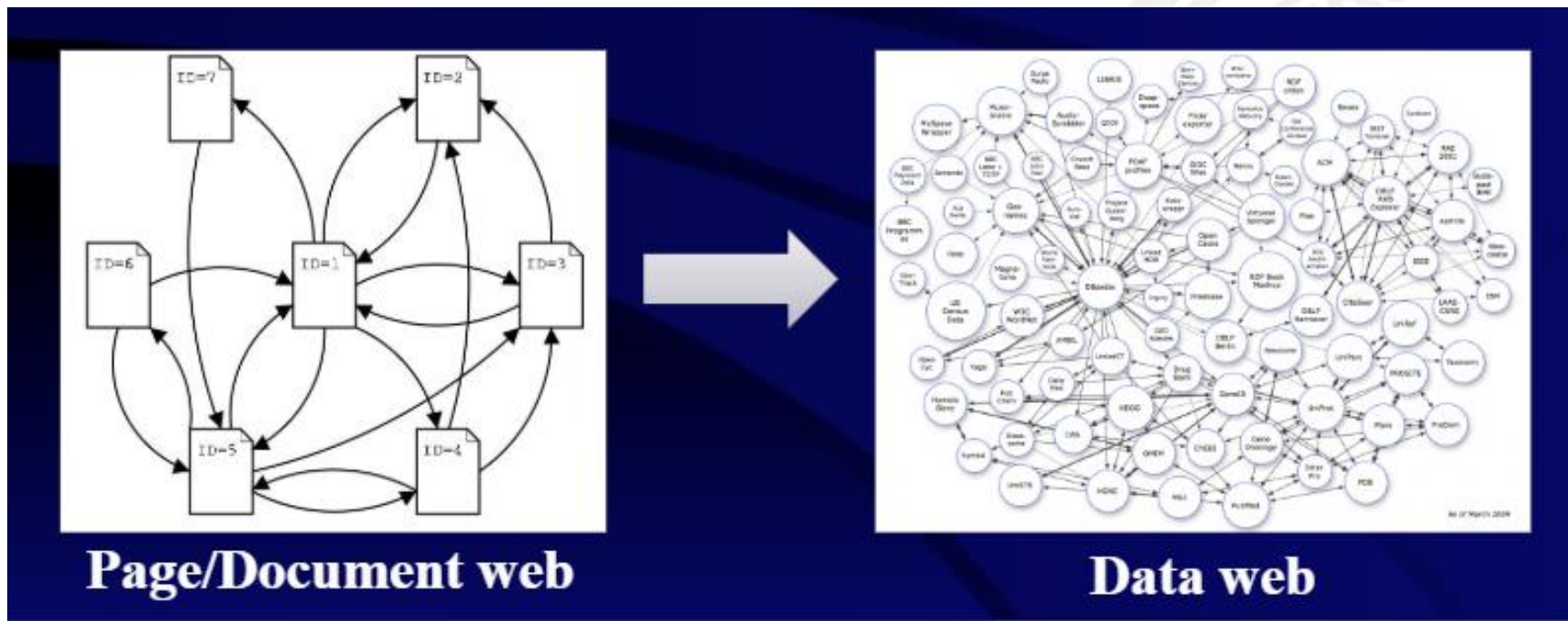
Tim Berners-Lee

OM KBE FRS FREng FRSA FBCS



4.3.1 语义网络

- 语义网络（语义Web）：信息内容具有良好的语义定义，在资源之间建立起机器可处理的各类语义联系，开发“以计算机可处理形式来表示信息含义的语言”，使计算机能自动理解语义并自动存取语义信息。



4.3.1 语义网络

语义Web的使用例子：张三和张四一起要带妈妈找专科大夫看病，做定期物理治疗。

- 张三通过手持设备，设置它的语音个人助理，个人助理很快调出张三以及张四的近一段时间安排，并通过医生代理，获取妈妈的病情及治疗情况。
- 启动搜索服务，搜索满足以下条件的医疗机构：1、离母亲家20英里范围内；2、在母亲保险公司的理赔列表中；3、第三方权威机构给出的评分是好评。
- 张三的个人助理与选择的服务提供商的智能代理进行交互，预约合适的时间。
- 个人助理把预约的时间分别发给张三和张四，协商时间是否合适？
- 张四提出时间不合适，因为那个时间是交通高峰期，堵车严重。
- 张三的个人助理重新与服务商代理交互，预约另一个合适的时间。

4.3.1 语义网络

1. 基本命题的语义网络表示

- 语义网络形式上是一个有向图
- 结点：问题领域中的实体、概念、事件、动作或状态
- 边：结点之间的语义联系
 - 语义网络中语义联系非常丰富，分为两种：
 - 1、以个体为中心组织知识的语义联系
 - 2、以谓词或关系为中心组织知识的语义联系

4.3.1 语义网络

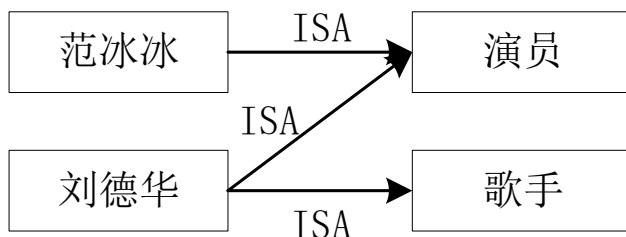
1.1、以个体为中心组织知识的语义联系

- 结点通常是名词: 个体或概念
- 连线表示
 - 1.1.1、实例和类（实例联系）之间的关系；
 - 1.1.2、类和类（泛化联系）之间的关系；
 - 1.1.3、组成部分与整体（聚集关系）之间的关系；
 - 1.1.4、属性与值（属性联系）之间的关系。

4.3.1 语义网络

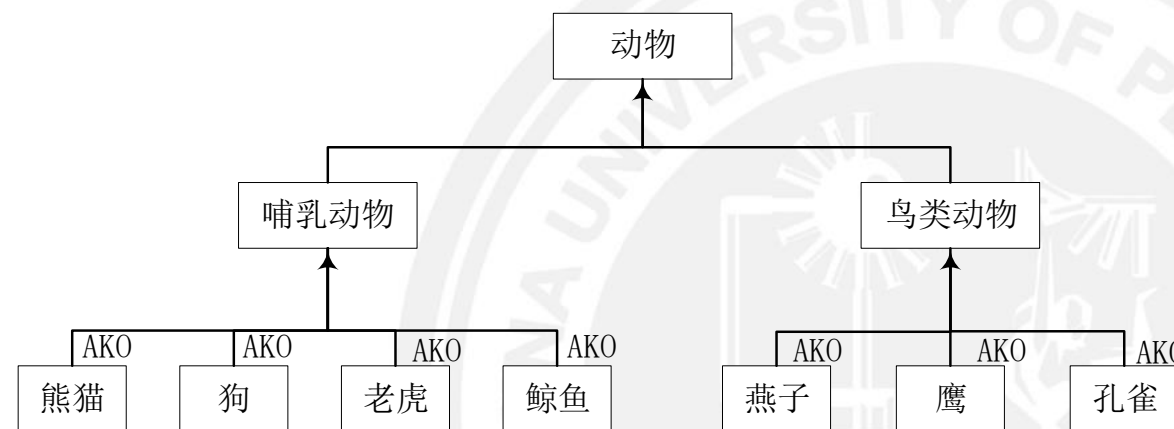
1.1、以个体为中心组织知识的语义联系

1.1.1、实例联系：ISA



一个实例结点可以连接多个类结点
多个实例结点可以连接一个类结点

1.1.2、泛化联系：AKO

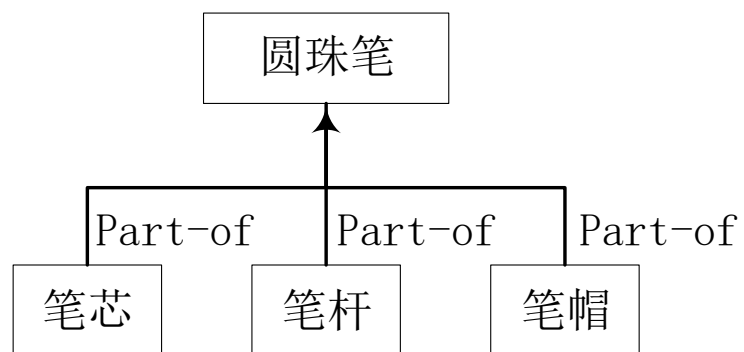


类结点与抽象层次更高的类结点之间的联系

4.3.1 语义网络

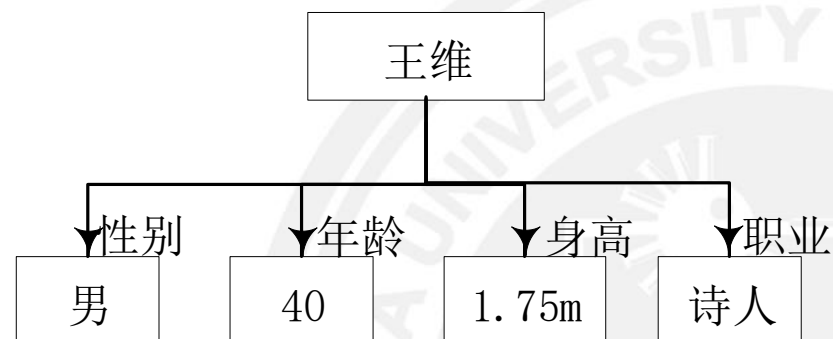
1.1、以个体为中心组织知识的语义联系

1.1.3、聚集联系:Part-of



组成成分与整体之间的关系

1.1.4、属性联系



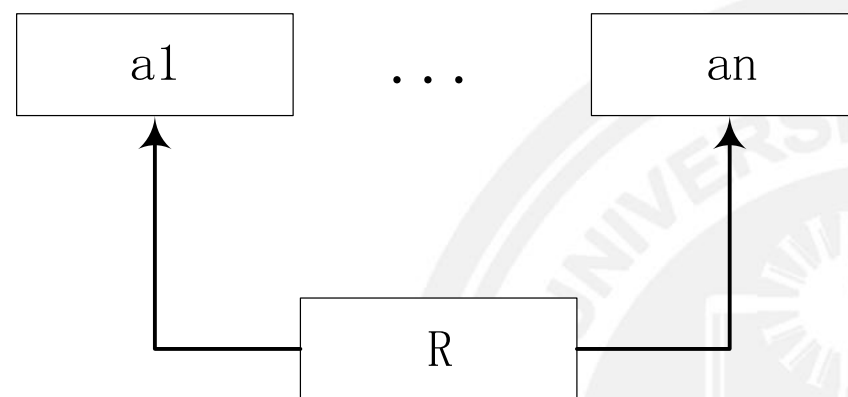
个体、属性及其取值之间的关系



4.3.1 语义网络

1.2、以谓词或关系为中心组织知识的语义联系

N元谓词或关系 $R(r_1, r_2, \dots, r_n)$, 分别取值为 $a_1, a_2, \dots, a_n$

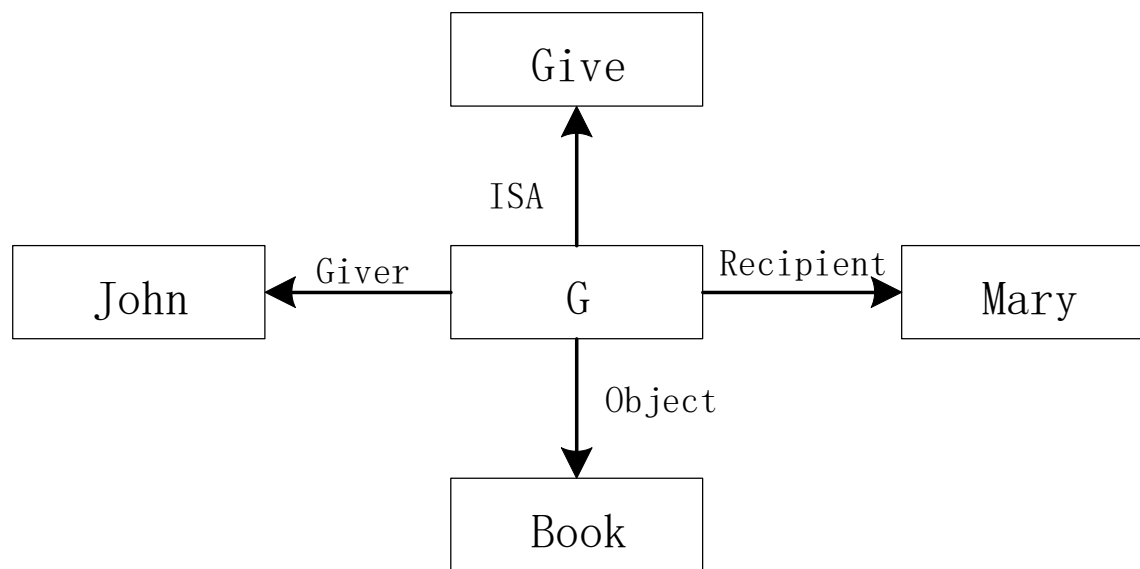


4.3.1 语义网络

2. 联结词在语义网络中的表示

- 合取

John gives Mary a book.



G为Give的一个实例

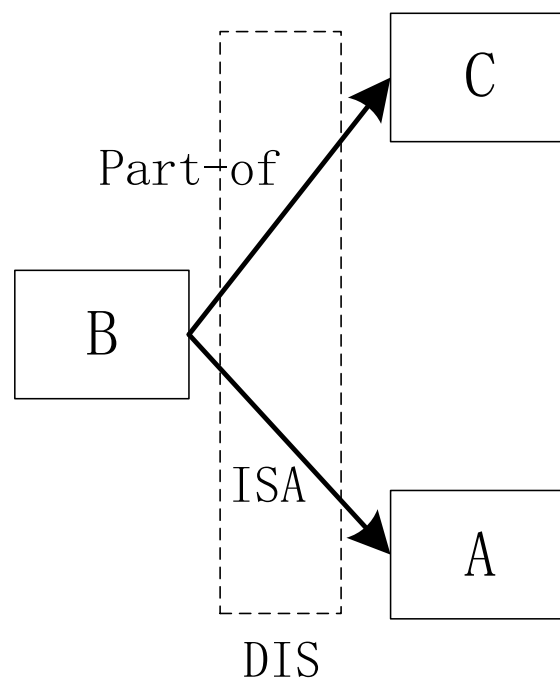


4.3.1 语义网络

2. 联结词在语义网络中的表示

- 析取

$ISA(A,B) \vee Part-of(B,C)$



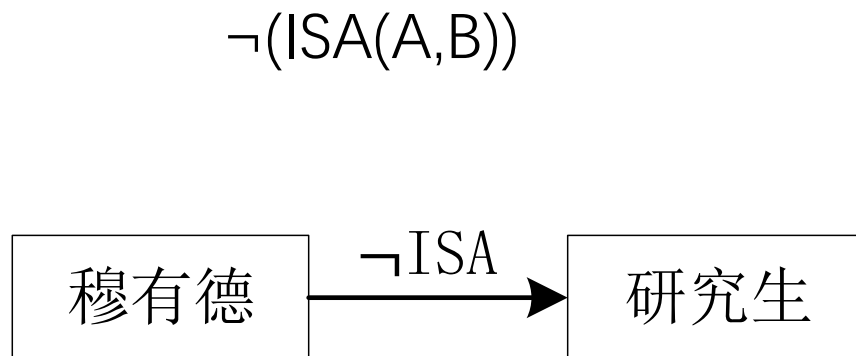
在连接上标注DIS表示析取
如省略表示合取



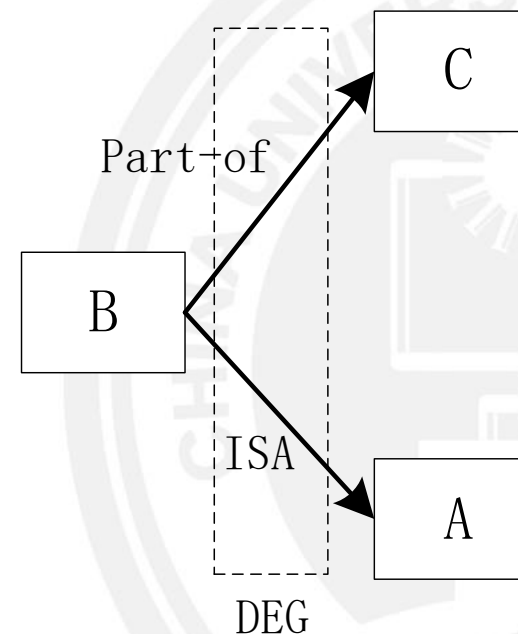
4.3.1 语义网络

2. 联结词在语义网络中的表示

- 否定



在连接上标注DEG表示多个关系的否定





4.3.1 语义网络

2. 联结词在语义网络中的表示

- 蕴含

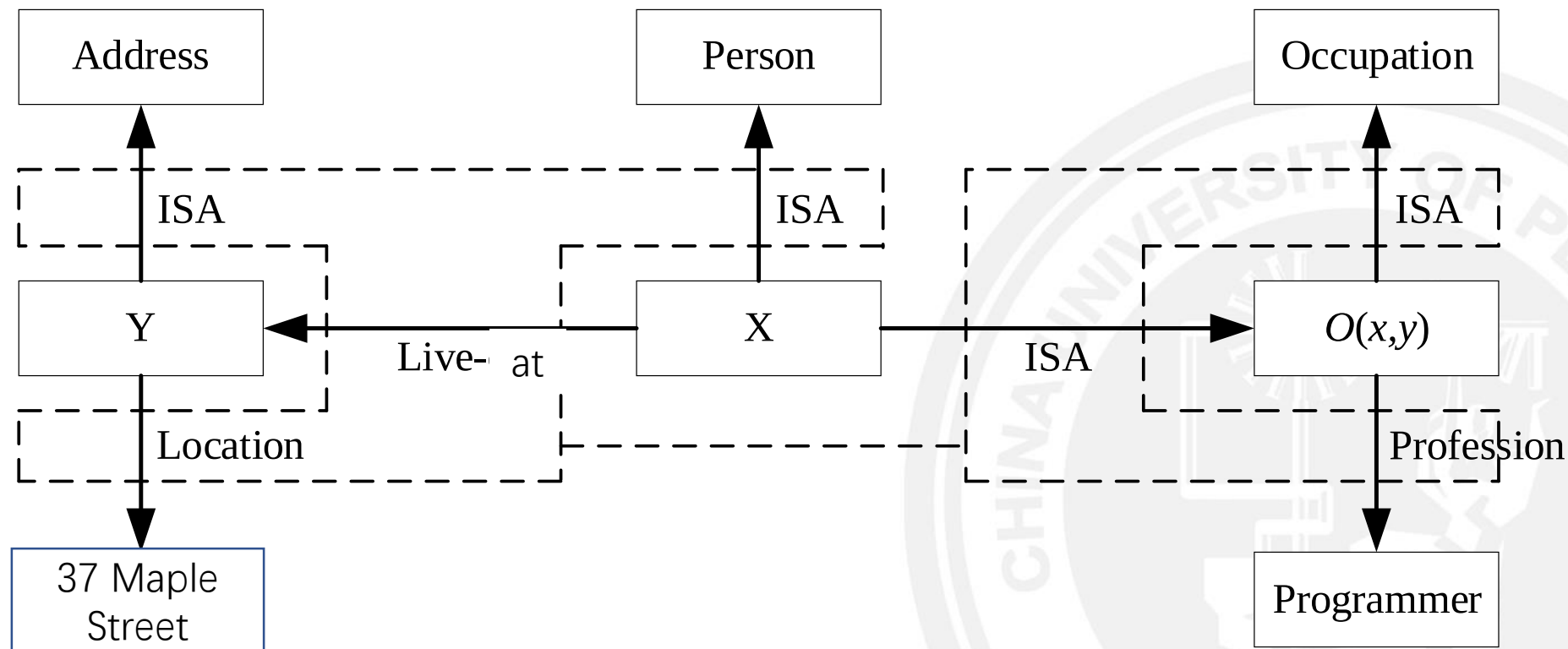
Every one who lives at 37 Maple Street is a programmer.



4.3.1 语义网络

2. 联结词在语义网络中的表示

- 蕴含



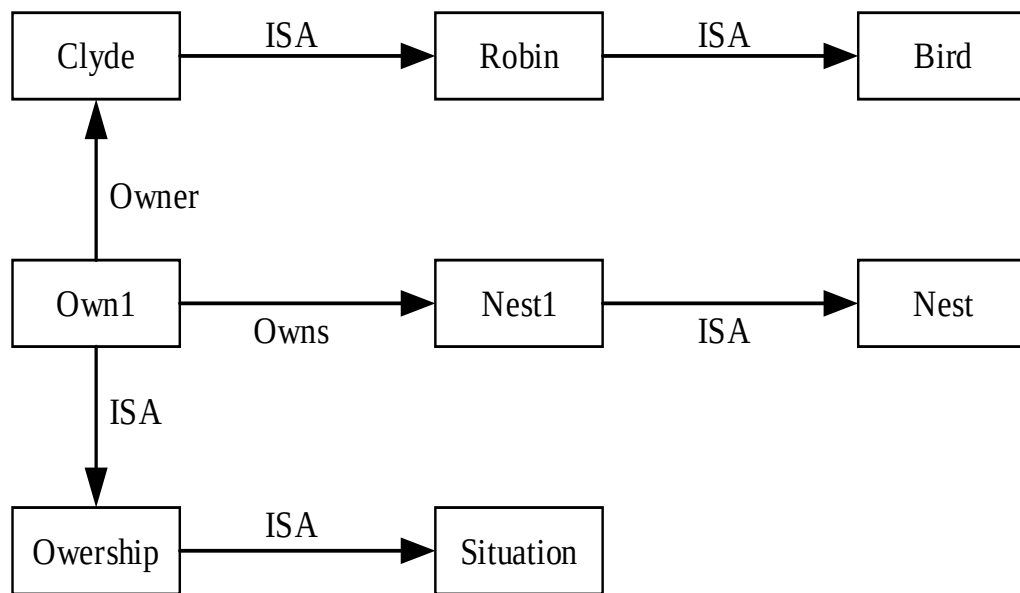
Every one who lives at 37 Maple Street is a programmer.

4.3.1 语义网络

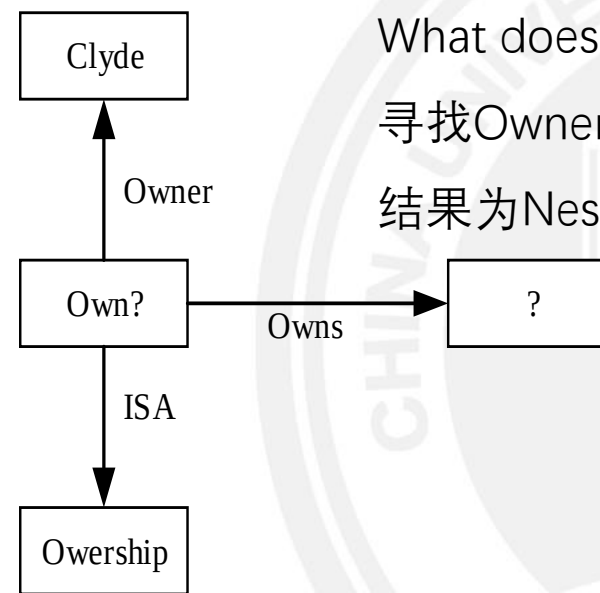
3. 语义网络的推理

3.1 匹配:

通过语义网络这种结构描述事物的匹配为结构上的匹配：从一条弧的两个节点匹配开始，再匹配与这两个节点相连接的所有其他节点，直到问题得到解答。



(a) 事实网络



(b) 目标网络

What does Clyde own?

寻找Owner弧指向Clyde的Own结点，
结果为Nest1

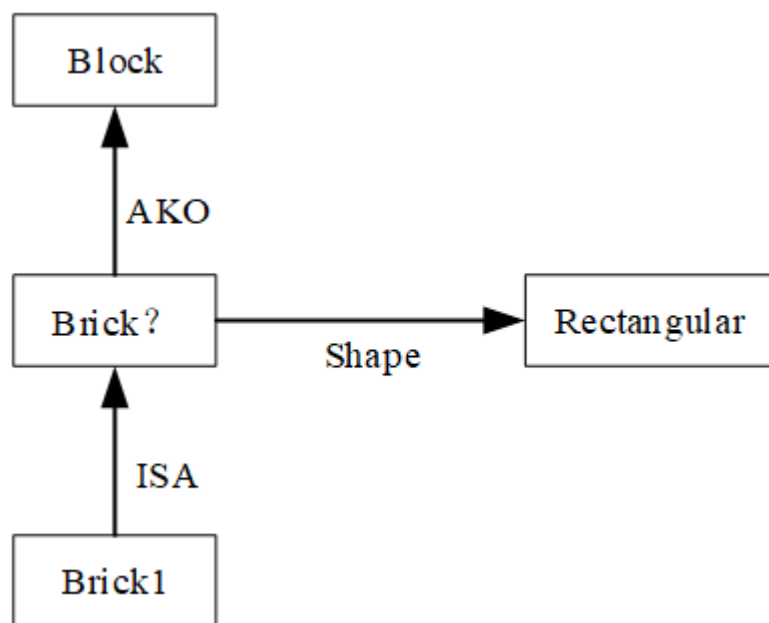
4.3.1 语义网络

3. 语义网络的推理

3.2 继承:

1、值继承

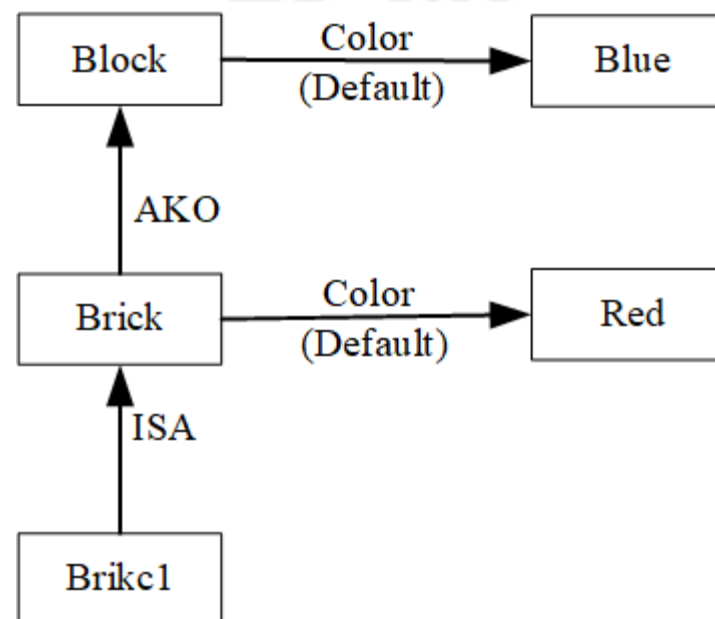
- 从Brick1结点出发, 检查是否有以其为出发点的Shape弧
- 如果没有, 依次检查与Brick1结点相连的ISA弧指向的结点



语义网络的值继承

2、默认继承

- 搜索带有Default标记的弧



语义网络的“默认”继承

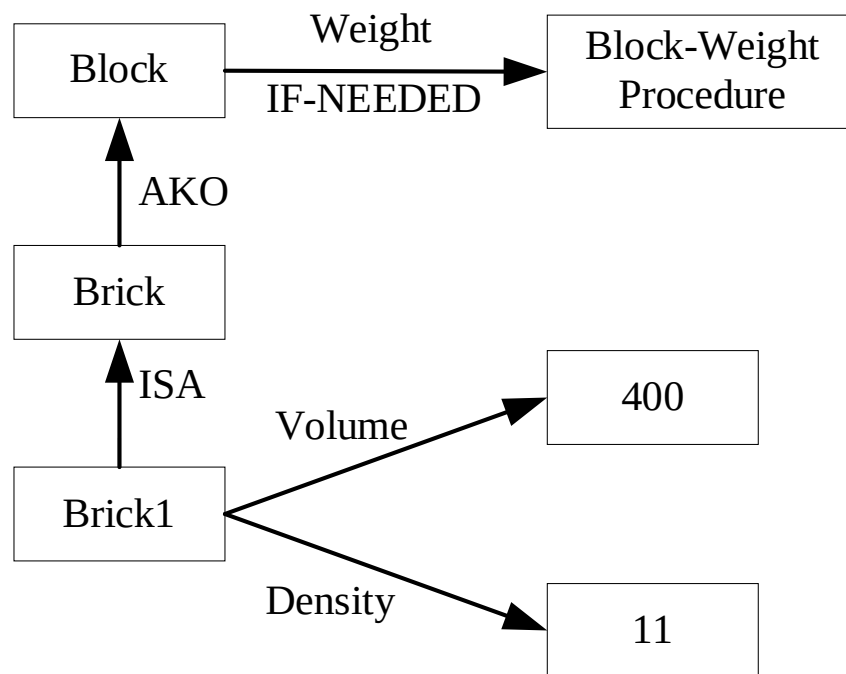
4.3.1 语义网络

3. 语义网络的推理

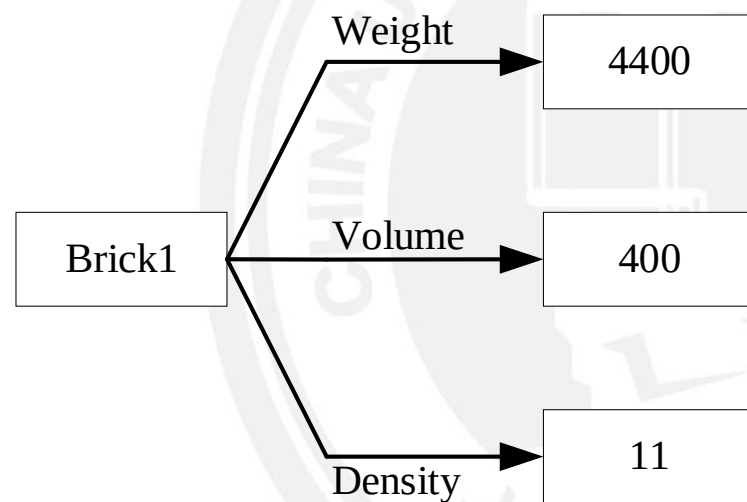
3.2 继承:

3、附加过程继承

- 搜索IF-NEEDED标记的Weight弧，执行if-needed程序，将结果填入Weight弧指向的结点



(a)



(b)

4.3.2 万维网知识描述语言

1. 可扩展标记语言

- 可扩展标记语言 (extensible markup language, XML)，是一种使用标签来组织互联网信息内容的标记语言。
- 标签不固定，可根据需要进行扩展。
- 简化了互联网环境下数据共享、数据传输、数据交换的过程。
- 构成元素
 - (1) 标签：用于标识一段数据，成对出现。形式：<tagname>……</tagname>。
可以嵌套使用：<a>……。
 - (2) 元素：被标签包围的数据。例如，<name>John</name>，John就是一个元素。
 - (3) 属性：为元素提供额外信息。例如，<person type=student> <name> John
</name> </person>，其中，type=student为该元素的属性。

4.3.2 万维网知识描述语言

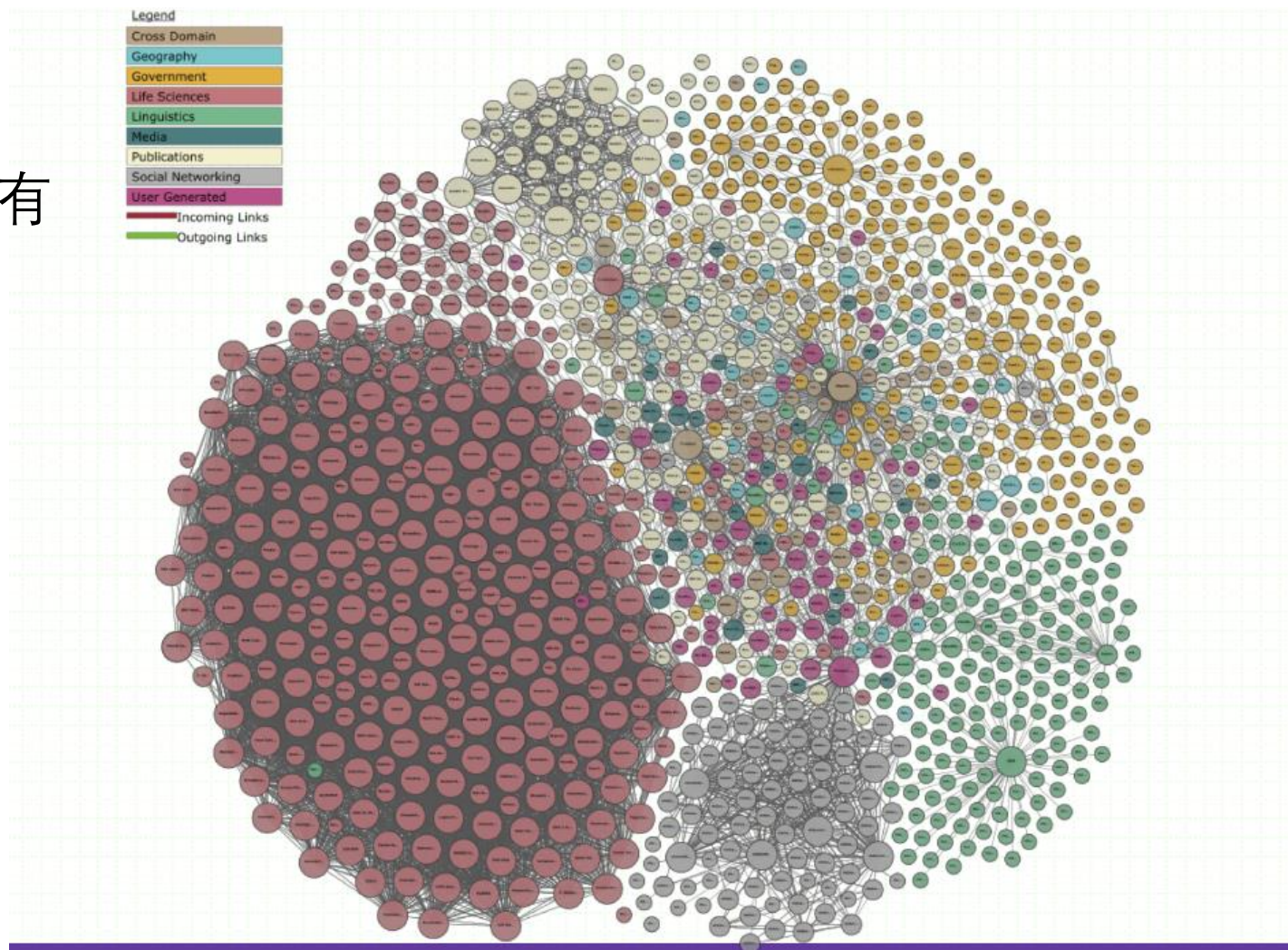
2.资源描述框架RDF

- 2004年成为W3C标准
- RDF包含：
 1. 资源：在全球统一资源标识符（uniform resources identifier,URI）的事物。
 2. 属性：属性是一种特殊的资源，描述了资源之间的关系。
 3. 陈述：一个由主语，谓语和宾语构成的三元组，这三个均为资源。
- 例如：人工智能导论的授课教师是张三，可以表示成为（人工智能导论，授课教师，张三）这个三元组，而其中人工智能导论、张三都是互联网上具有URI的对象。

4.3.2 万维网知识描述语言

2. 资源描述框架RDF

- 主语和宾语->节点；之间的关系->有向边
==》互联网的知识结构->图结构
- linked open data (LOD) 项目
 - 初期，包含Dbpedia, FOAF
 - 现在，1163个数据库





4.3.2 万维网知识描述语言

3. 网络本体语言OWL

- OWL (web ontology language) 这项工作的目的是通过增加关于那些描述或提供网络内容的资源的信息, 从而使网络资源能够更容易地被那些自动进程访问。
- 属性刻画:
 - 定义域(domain)和值域(range)
 - 子属性(subPropertyOf)
 - 属性的不同特性

4.3.2 万维网知识描述语言

3. 网络本体语言OWL

- 水系本体

(1)对称性: 关系是相互的。对于概念

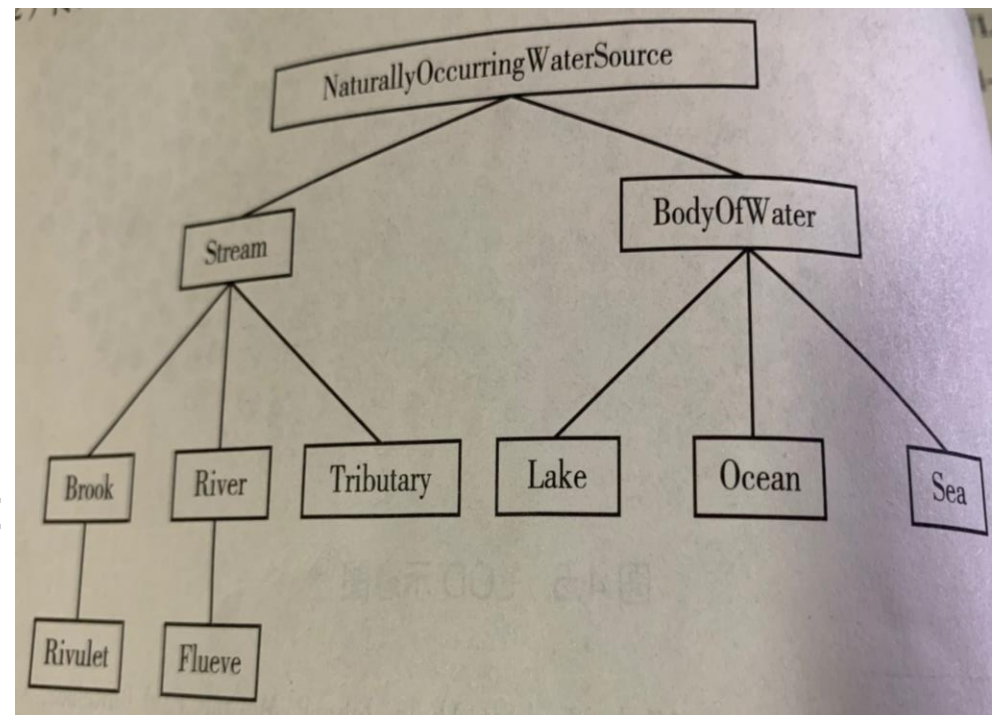
NaturallyOccurringWaterSource定义属性connectsTo:

NaturallyOccurringWaterSource表示水域相互连接,

connectsTo具有对称性表示: 如果“长江connectsTo东海”推出“东海connectsTo长江”。

(2)传递性: 包含关系。对于BodyOfWater定义属性containedIn:BodyOfWater为传递性属

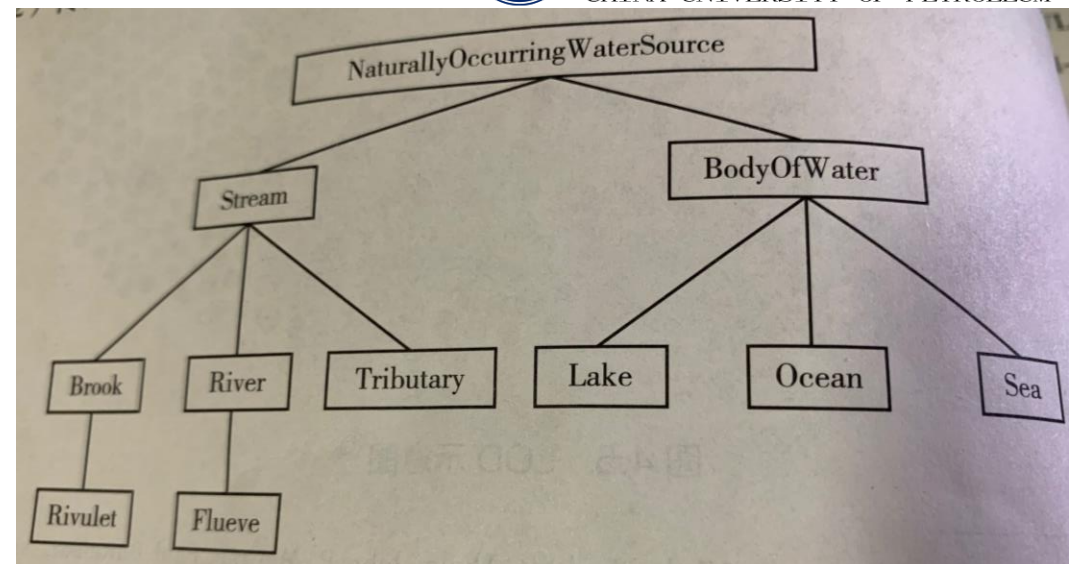
性: 如果“杭州湾containedIn东海”和“东海containedIn太平洋”, 可以推出“杭州湾containedIn太平洋”。



4.3.2 万维网知识描述语言

3. 网络本体语言OWL

- 水系本体



(3)函数性: 每个实例只有一个属性值。如对概念River定义函数属性emptiesInto:BodyOfWater,

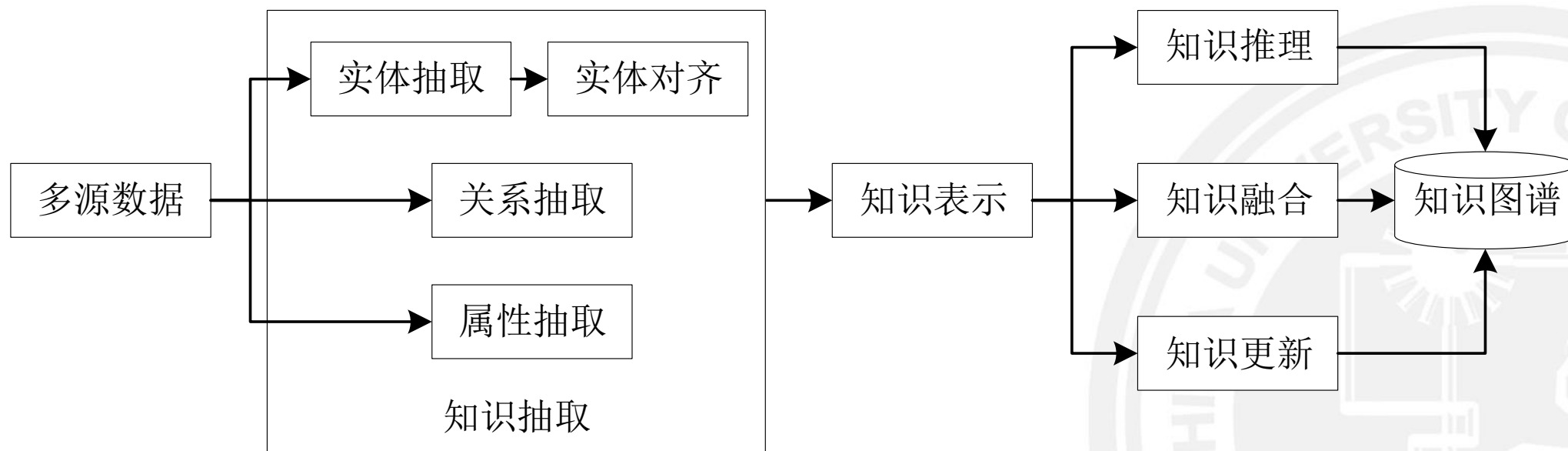
如果“长江emptiesInto东海”和“长江emptiesInto Sea02”, 则可以推断Sea02是东海。

(4)可逆性: 两个属性互为逆属性。如对概念BodyOfWater定义属性feedsFrom:River, 并指定它和emptiesInto互为逆属性, 则“长江emptiesInto东海”可以推断出“东海feedsFrom长江”。

(5)反函数性: 一个属性值对应的定义域中自变量唯一。如feedsFrom。

4.4 知识图谱构建

■ 构建知识图谱过程



4.4 知识图谱构建

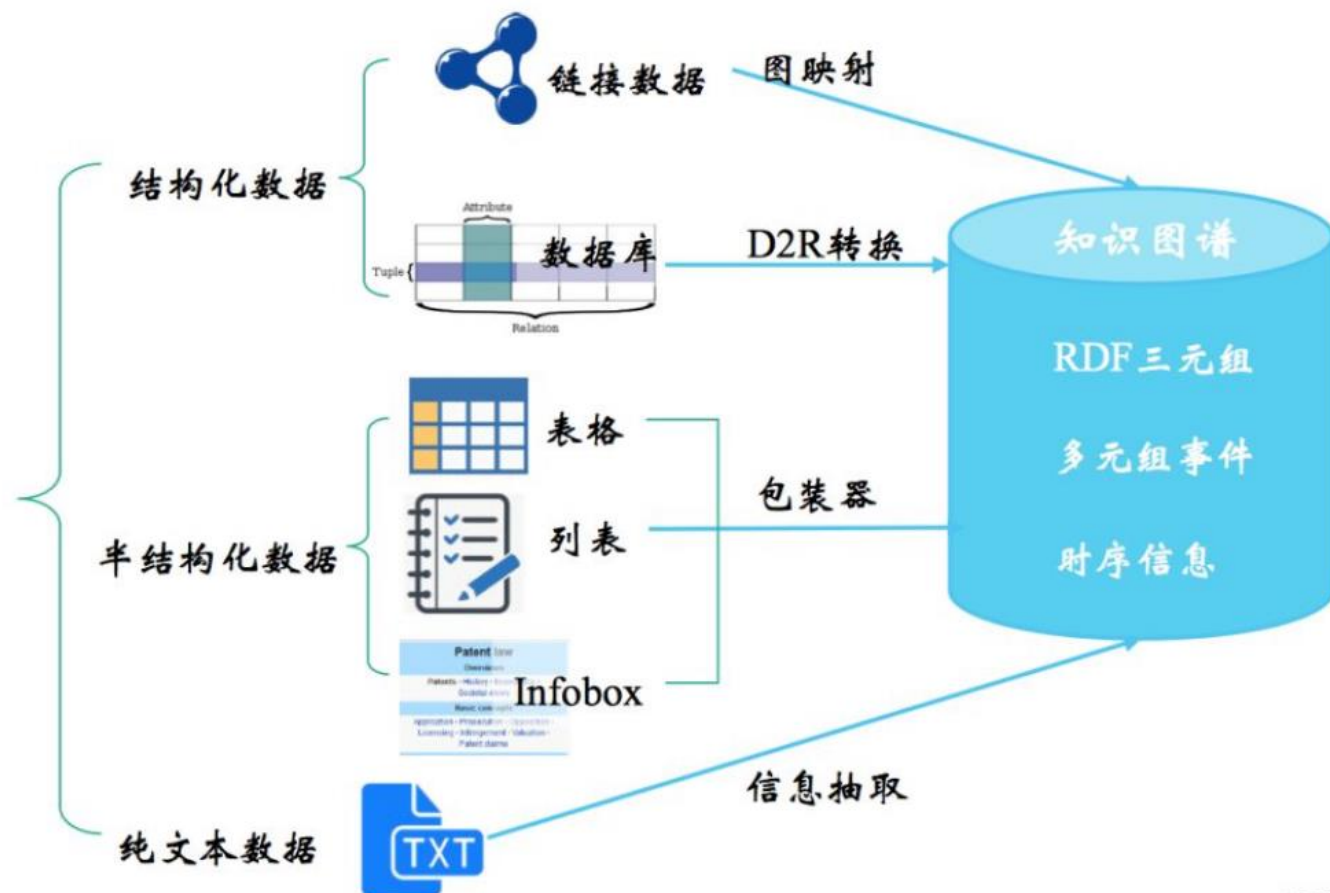
1. 知识抽取

- 面向开放的链接数据，通过自动化的技术提取出可用的知识单元
- 知识单元主要包括实体(概念的外延)、关系以及属性3个知识要素
- 形成一系列高质量的事实表达，为上层模式层的构建奠定基础。

4.4 知识图谱构建

1. 知识抽取

知识抽取包括从多种数据源获取数据，包括结构化数据，如链接数据、数据库等；半结构化数据，如表格、列表等和非结构化数据即文本数据。



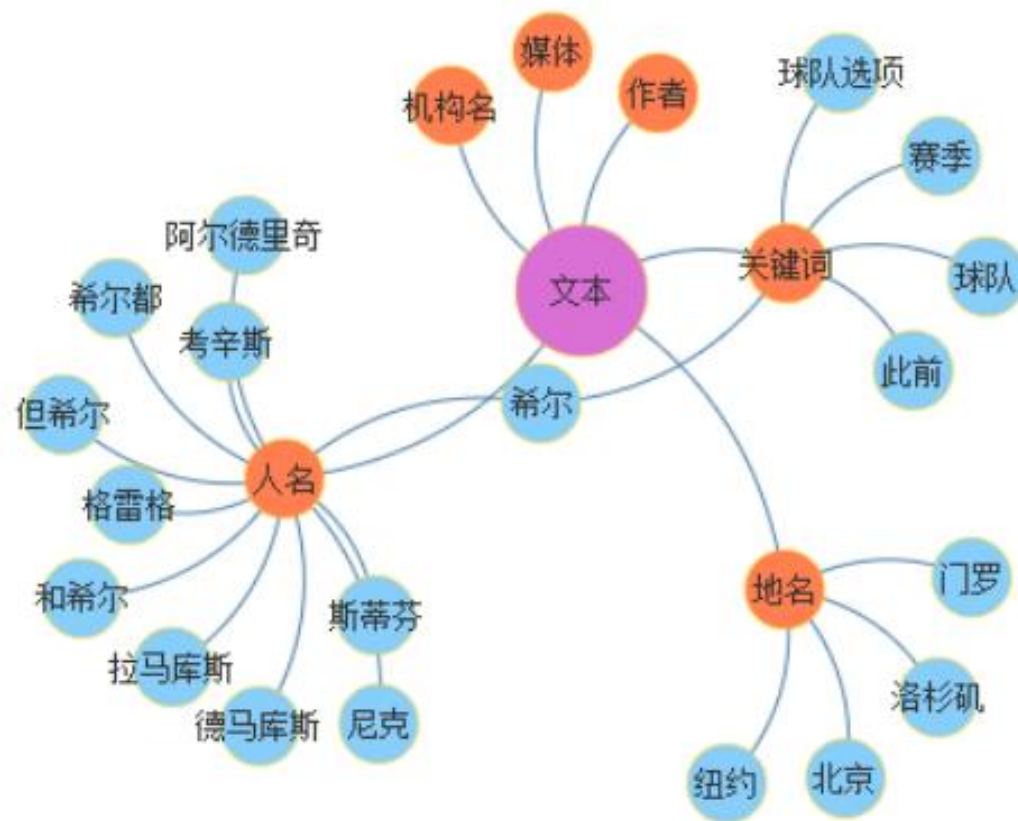
4.4 知识图谱构建

2. 实体抽取

- 从原始语料中自动识别出命名实体。
- 由于实体是知识图谱中的最基本元素，其抽取的完整性、准确率、召回率等将直接影响到知识库的质量。
- 实体抽取是知识抽取中最为基础与关键的一步

4.4 知识图谱构建

2. 实体抽取



应用NLPIR实体抽取系统智能识别文本中出现的人名、地名、机构名、媒体、作者及文章的主题关键词。左图为一 段文本的实体抽取的结果展示，图中橘色的点为实体类型，蓝色点为实体内容。

4.4 知识图谱构建

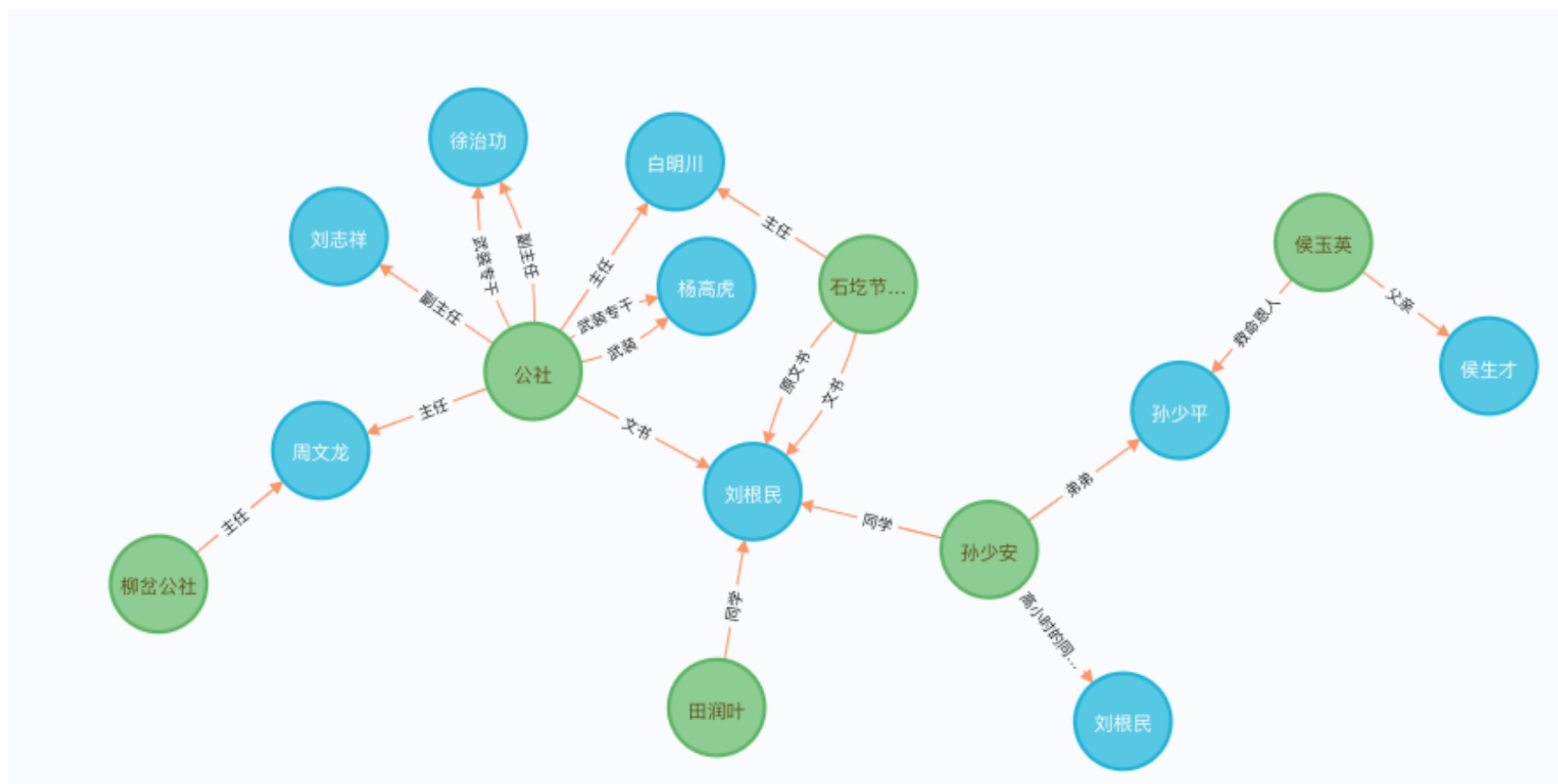
3. 关系抽取

- 关系抽取 (Relation Extraction, RE) 是知识抽取的重要子任务之一，主要目的是从文本中识别实体并抽取实体之间的语义关系。
- 关系的基本信息包括参数类型、满足此关系的元组模式等。
- 关系抽取的完整性、准确率、召回率等将直接影响到知识库的质量。

4.4 知识图谱构建

3. 关系抽取

小说《平凡的世界》中部分实体关系抽取结果如下：



4.4 知识图谱构建

4. 知识表示

- 基于三元组的知识表示形式受到了人们广泛的认可，但是其在计算效率、数据稀疏性等方面却面临着诸多问题。
- 以深度学习为代表的表示学习技术取得了重要的进展，可以将实体的语义信息表示为稠密低维实值向量，在低维空间中高效计算实体、关系及其之间的复杂语义关联，用于对知识库的构建、推理、融合以及应用。

4.4 知识图谱构建

4. 知识表示

- RDF三元组是表示知识图谱的一种常见表示形式，以(subject, predicate, object)的三元组形式就足以清晰表示实体之间的许多复杂联系。三元组形式如下：
 - ['清政府' , '签订' , '天津条约']
 - ['袁世凯' , '复辟' , '帝制']
 - ['孙中山' , '就任' , '临时大总统']

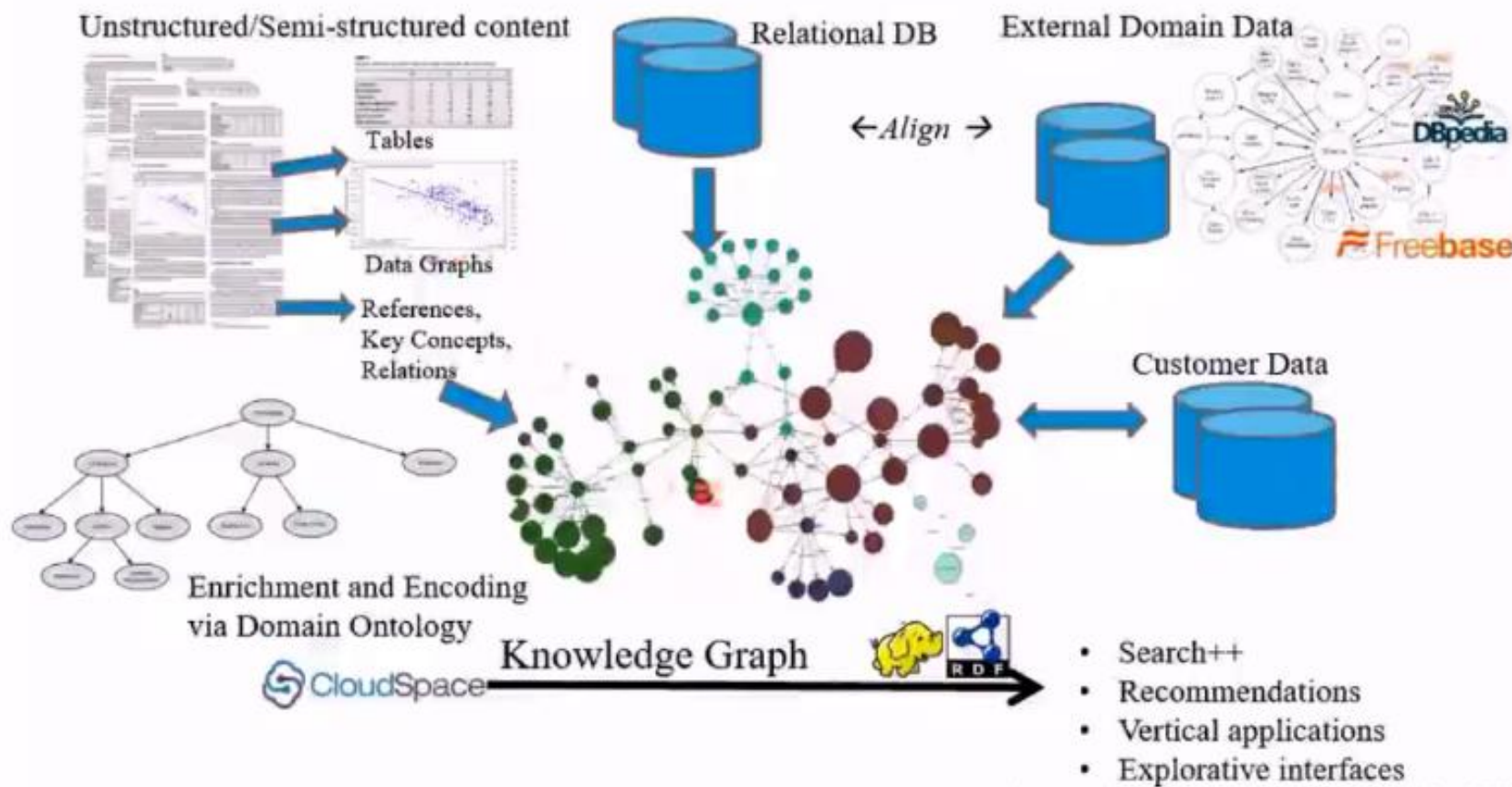
4.4 知识图谱构建

5. 知识融合

- 由于知识图谱中的知识来源广泛，存在知识质量良莠不齐、来自不同数据源的知识重复、知识间的关联不够明确等问题。
- 知识融合是高层次的知识组织，使来自不同知识源的知识在同一框架规范下进行异构数据整合、消歧、加工、推理验证、更新。
- 达到数据、信息、方法、经验以及人的思想的融合，形成高质量的知识库。

4.4 知识图谱构建

5. 知识融合



知识图谱的融合需要融合来自多种数据源的数据，包括结构化数据，如链接数据、数据库等；半结构化数据，如表格、列表等和非结构化数据。

4.4 知识图谱构建

6. 实体对齐

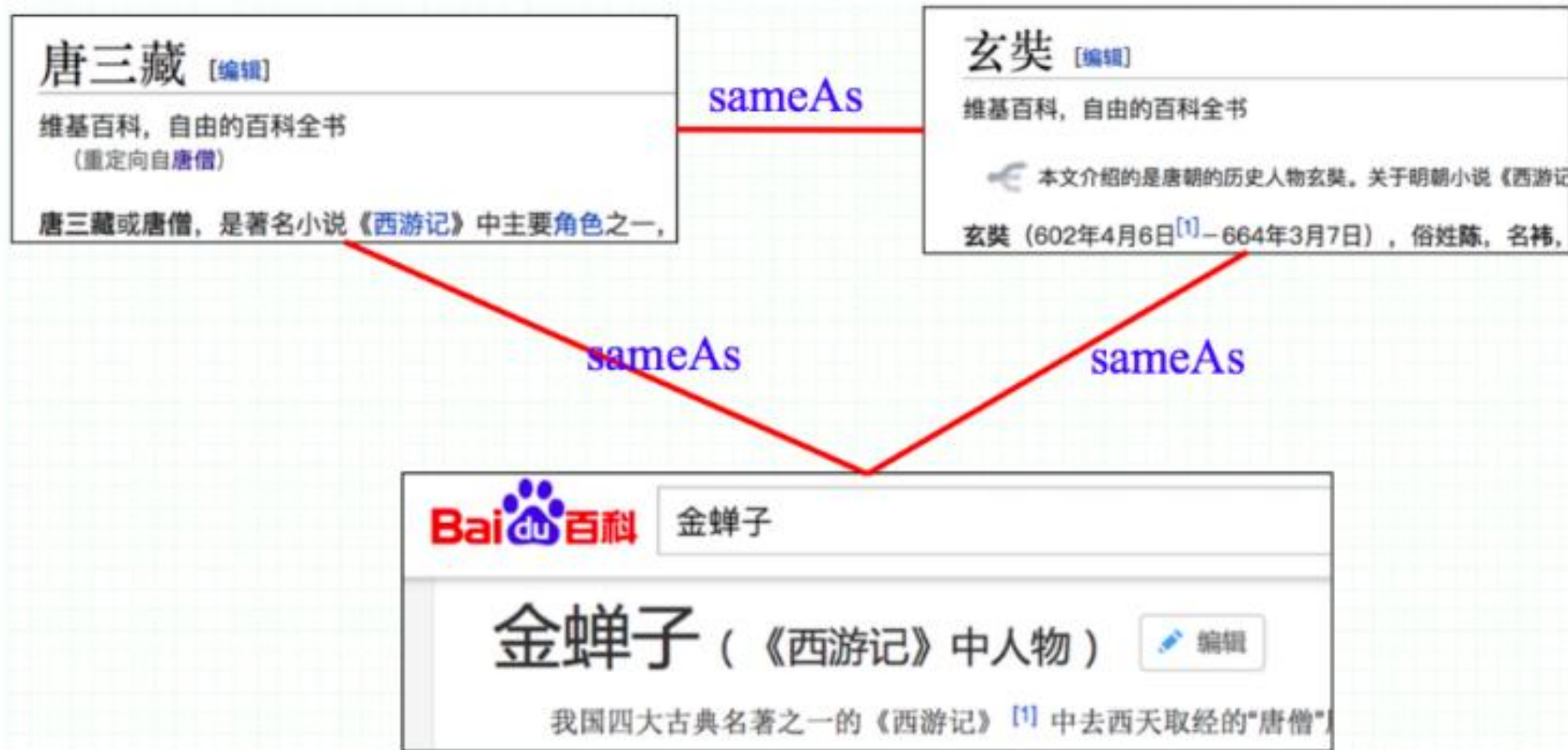
实体对齐(entity alignment)也称为实体匹配(entity matching)或实体解析(entity resolution), 主要是用于消除异构数据中实体冲突、指向不明等不一致性问题。

- 可以从顶层创建一个大规模的统一知识库, 从而帮助机器理解多源异质的数据, 形成高质量的知识。

4.4 知识图谱构建

6. 实体对齐

实体对齐将来自多个来源的关于同一个实体或概念的描述信息融合起来得到唯一表示。



4.4 知识图谱构建

7. 知识更新

- 根据知识图谱的逻辑结构，其更新主要包括模式层的更新与数据层的更新。
- 模式层的更新是指本体中元素的更新，包括概念的增加、修改、删除，概念属性的更新以及概念之间上下位关系的更新等。
- 概念属性的更新操作将直接影响到所有直接或间接属性的子概念和实体。
- 模式层的增量更新方式消耗资源较少，但是多数情况下是在人工干预的情况下完成的，例如需要人工定义规则人工处理冲突等，实施起来并不容易
- 数据层的更新指的是实体元素的更新，包括实体的增加、修改、删除，以及实体的基本信息和属性值。由于数据层的更新一般影响面较小，因此通常以自动的方式完成。

4.4 知识图谱构建

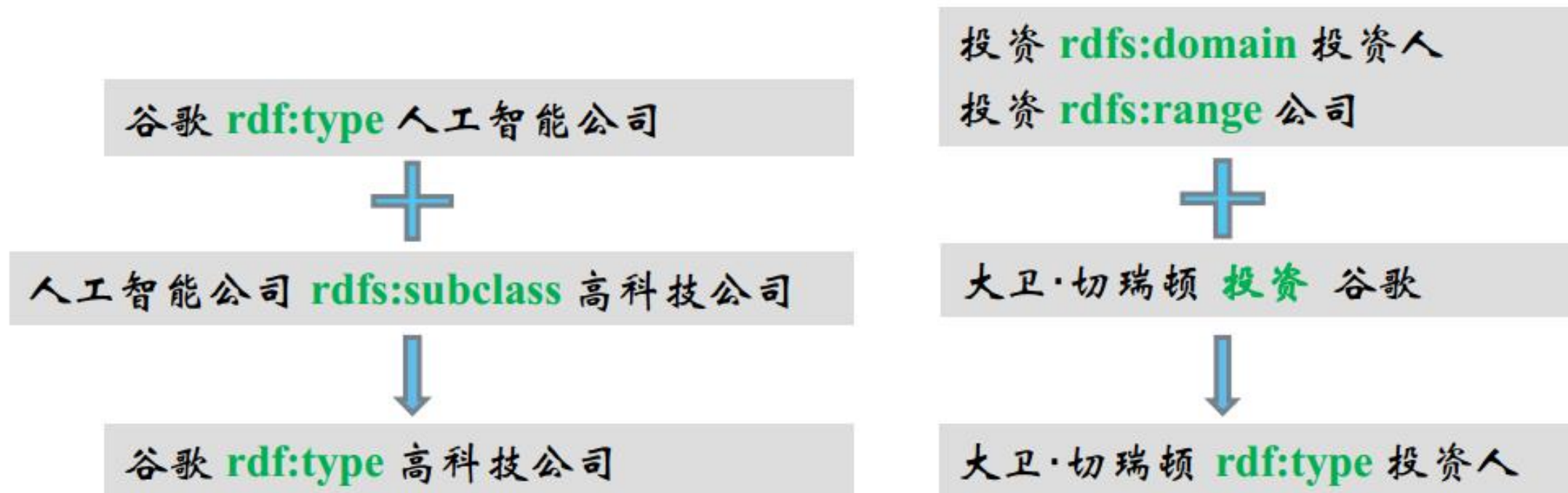
8. 知识推理

- 知识推理则是在已有的知识库基础上进一步挖掘隐含的知识，从而丰富、扩展知识库。
- 在推理的过程中，往往需要关联规则的支持。
- 由于实体、实体属性以及关系的多样性，人们很难穷举所有的推理规则，一些较为复杂的推理规则往往是手动总结的。
- 对于推理规则的挖掘，主要还是依赖于实体以及关系间的丰富同现情况。
- 知识推理的对象可以是实体、实体的属性、实体间的关系、本体库中概念的层次结构等。
- 知识推理方法可分为基于逻辑的推理与基于图的推理两种类别。

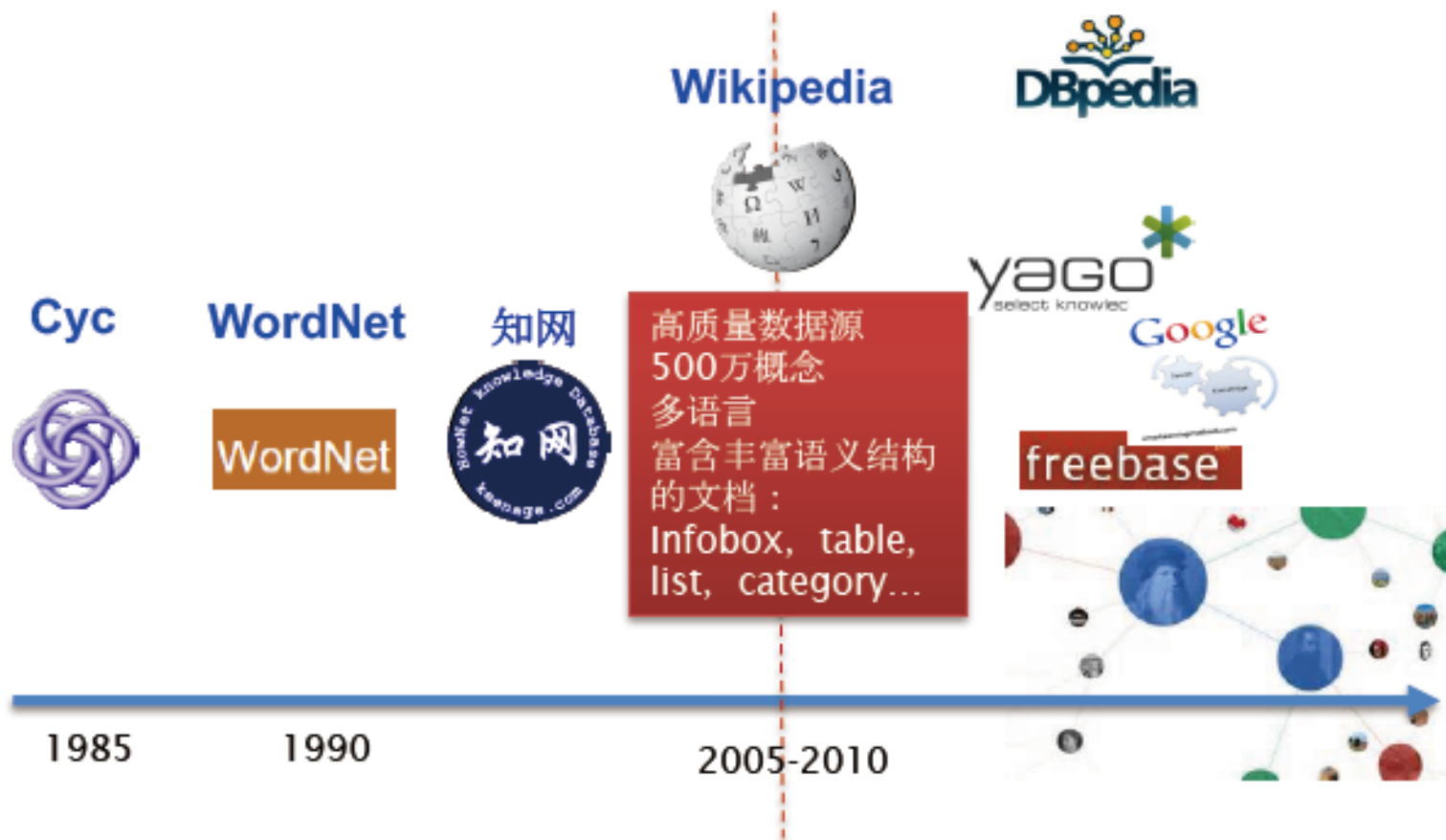


4.4 知识图谱构建

8. 知识推理



4.5 知识图谱的现状与发展

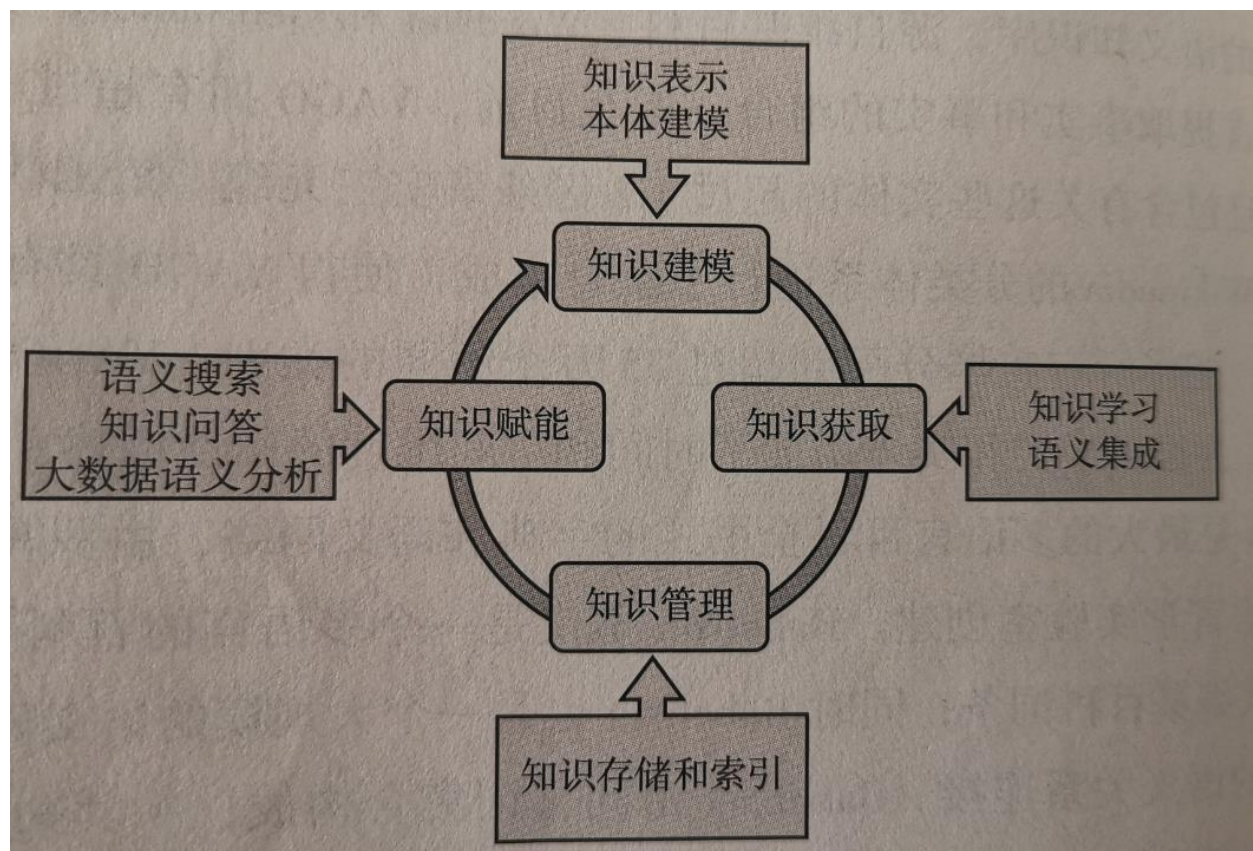


4.5.1 维基类知识结构化与常用知识图谱

- 维基百科 (Wikipedia) ， 是一个基于维基技术的多语言百科全书协作计划，用多种语言编写的网络百科全书。
- 非营利组织维基媒体基金会负责营运维基百科
- 285种语言版本
- 每个词条包含对应语言的客观实体、概念的文本描述、丰富的属性信息、属性值信息

4.5.2 知识图谱的生命周期

- 生命周期：知识建模、知识获取、知识管理、知识赋能
- 知识在数据中的分布特点：
 - (1) 多媒体性
 - (2) 隐蔽性
 - (3) 分布性
 - (4) 异构性
 - (5) 海量性



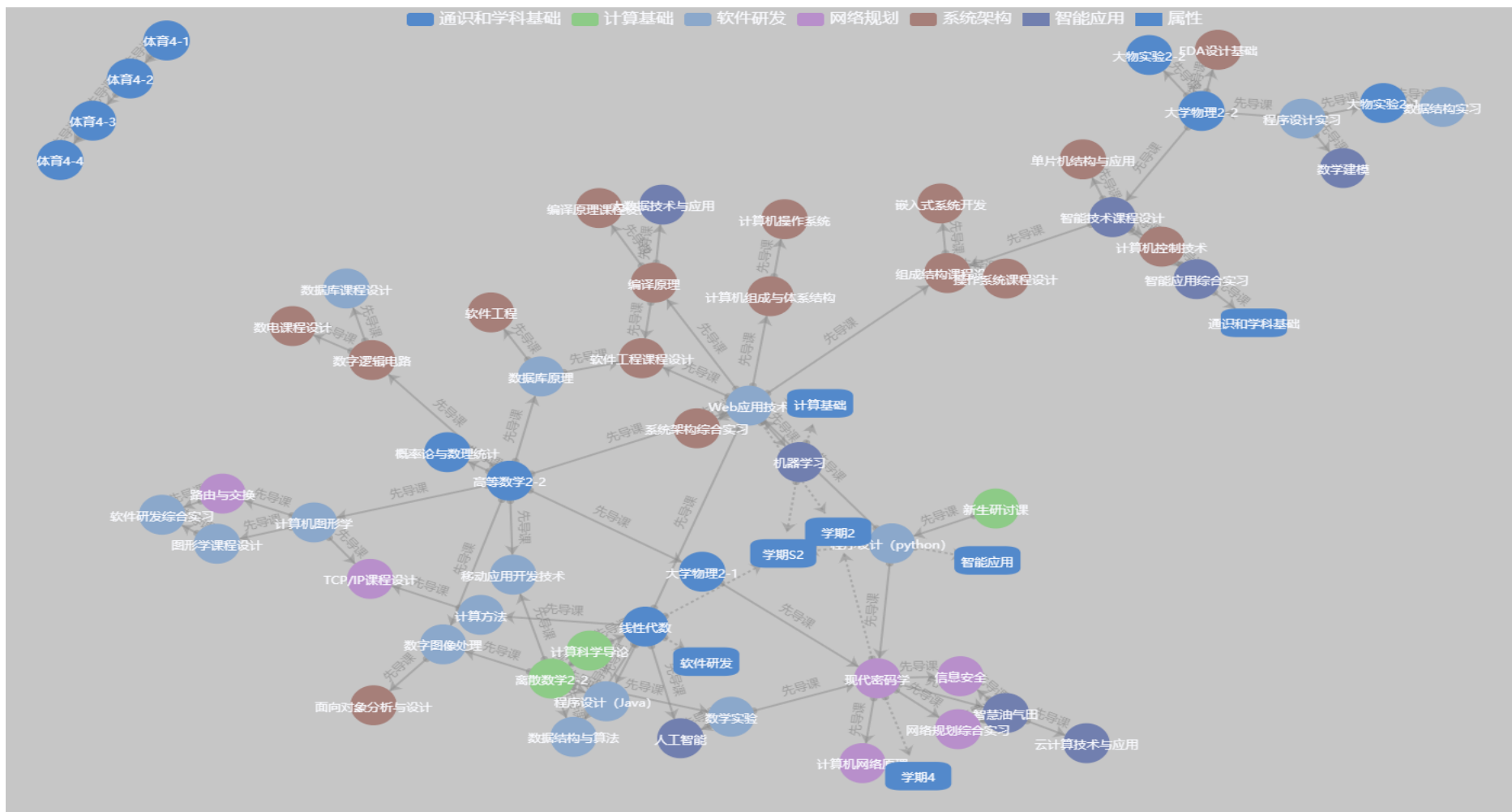
4.5.2 知识图谱的生命周期

- 知识建模：定义领域知识描述的概念、事件、规则及相互关系的知识表示方法，建立知识图谱的概念模型，主要包括领域概念及概念层次学习、概念属性学习等等。
- 知识获取：对知识建模定义的知识要素进行实例化的过程。
- 知识管理：主要研究知识图谱的存储和索引，方便快速访问和查询。
- 知识赋能：知识图谱最初提出的目的是增强搜索结果，改善用户搜索体验，即语义搜索。



4.5.2 知识图谱的应用实例

1. 课程知识图谱



4.5.2 知识图谱的应用实例

1.课程知识图谱

- 构建过程

(1) 统计并整理所有课程信息，从中抽取课程实体，找出课程实体之间存在的先后选修或必修关系。

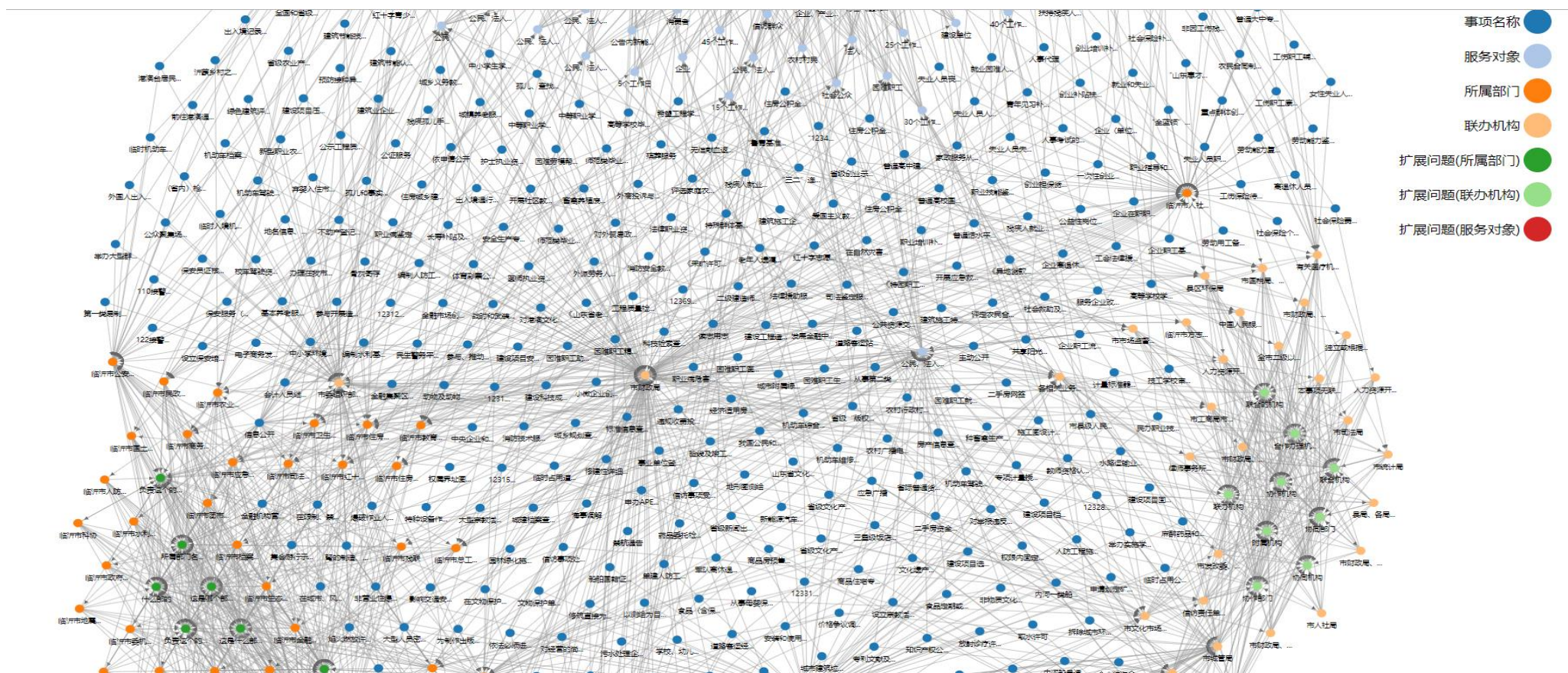
(2) 对所有课程实体分类并建立各自的属性标签，其中属性包括课程的开课学期、课程所属类别、课程性质等。

(3) 采用echarts可视化展示课程实体、实体间关系及各个实体的属性得到课程知识图谱。



4.5.2 知识图谱的应用实例

2.部门知识图谱



4.5.2 知识图谱的应用实例

2.部门知识图谱

- 部门知识图谱构建过程

(1) 基于山东政务服务网爬取得到的数据，抽取数据中的所属部门、事项名称、服务对象、联办机构作为实体，分析各个实体间存在的关系。

(2) 将抽取得到的实体及实体间关系存储于Neo4j数据库中，Neo4j数据库是以图形结构的形式存储数据的数据库，它以<实体,关系,实体>三元组的形式存储数据，便于知识的存储与查询。

(3) 采用D3可视化技术创建svg图层，在图层中创建力引导图展示各个实体和实体间关系。



4.5.2 知识图谱的应用实例

• Neo4j数据库部分存储展示

\$ MATCH p()-[r: `所属部门 (实施主体)`]->() RETURN p LIMIT 25

Graph

Table

Text

Code

"p"

[{"事项类型": "公共教育", "是否收费": "不收费", "监督电话": "0539-8303939", "承诺办结时限": "1个工作日", "权限划分": "本事项无权限划分", "运行系统": "省级", "办件类型": "即办件", "通办范围": "跨市", "法定办结时限": "1个工作日", "行使层级": "市级", "网上支付": "否", "咨询电话": "0539-8313810", "数量限制": "本事项无数量限制", "物流快递": "是", "办理形式": "仅窗口办理", "name": "中小学生学籍信息修改"}, {"name": "临沂市教育局", "实施主体性质": "法定机关"}]

[{"事项类型": "公共教育", "是否收费": "不收费", "承诺办结时限": "即办", "监督电话": "0539-8316047", "权限划分": "本事项无权限划分", "运行系统": "省级", "办件类型": "即办件", "通办范围": "未实现", "法定办结时限": "0个工作日", "行使层级": "市级", "网上支付": "否", "咨询电话": "0539-8316047", "数量限制": "本事项无数量限制", "物流快递": "是", "办理形式": "仅网上办理", "name": "中等职业学校学生学历证明"}, {"name": "临沂市教育局", "实施主体性质": "法定机关"}]

[{"事项类型": "公共教育", "是否收费": "不收费", "承诺办结时限": "即办", "监督电话": "0539-8316047", "权限划分": "本事项无权限划分", "运行系统": "省级", "办件类型": "即办件", "通办范围": "跨市", "法定办结时限": "0个工作日", "行使层级": "市级", "网上支付": "否", "咨询电话": "0539-8316047", "数量限制": "本事项无数量限制", "物流快递": "是", "办理形式": "仅窗口办理", "name": "中等职业学校学生休学"}, {"name": "临沂市教育局", "实施主体性质": "法定机关"}]

[{"事项类型": "公共教育", "是否收费": "不收费", "监督电话": "0539-8317064", "承诺办结时限": "1个工作日", "权限划分": "本事项无权限划分", "运行系统": "省级", "办件类型": "承诺件", "通办范围": "跨市", "法定办结时限": "1个工作日", "行使层级": "市级", "网上支付": "否", "咨询电话": "0539-8317064", "数量限制": "本事项无数量限制", "物流快递": "是", "办理形式": "窗口、网上均可办理", "name": "师范类毕业生就业手续办理"}, {"name": "临沂市教育局", "实施主



4.5.2 知识图谱的应用实例

• 部分代码展示

采用D3可视化技术中的力引导图实现图谱的展示，D3 将任意数据绑定至文档对象模型(DOM)，然后基于数据驱动原理操作文档的生成。采用D3中的JavaScript 函数库，通过定义图层高度、宽度等属性建立svg图层，读取json文件数据，将json文件中的节点和边绑定到DOM上，基于读取的数据在图层上定义节点间斥力、引力等属性创建力引导图。

```
d3.json("query.json", function(error,data) {  
    if(error) throw error;  
    var color = d3.scaleOrdinal(d3.schemeCategory10);  
    var width = window.innerWidth;  
    var height = window.innerHeight;  
    data.links.forEach(function (e) {  
        if (typeof e.source == "number" && typeof e.target == "number") {  
            var sourceNode = data.nodes.filter(function (n) {  
                return n.id === e.source;  
            })[0],  
            targetNode = data.nodes.filter(function (n) {  
                return n.id === e.target;  
            })[0];  
            e.source = sourceNode;  
            e.target = targetNode;  
        }  
    });  
});
```

```
vis = d3.select("svg")  
    .attr("width", width)  
    .attr("height", height)  
    .call(zoom).on("dblclick.zoom", null);  
  
//定义force对象，力导图  
force = d3.forceSimulation()  
    .nodes(data.nodes)  
    .force("link",  
d3.forceLink(data.links).distance(dis))  
    .force("charge", d3.forceManyBody())  
    .force("center", d3.forceCenter(width / 2,  
height / 2))  
    .force("charge", d3.forceManyBody())
```



THANKS

