# 知识表示与建模

## 知识表示由来与发展

知识表示在通俗意义上是指将人类社会经历上千年的思想、经验、理论通过特定的表达方法和一定的组织结构以一种统一而容易理解的方式描述出来。但是，随着计算机技术的出现和发展，人们开始逐渐尝试将知识融入到计算机语言中，让机器代替人工去理解和处理知识，这就要求我们创造出一种知识表示方法既可以满足人脑对知识的直观性理解又可以便于计算机表示和处理知识。而我们今天要介绍的“知识表示”正是这样一种神奇的方法。

随着人工智能的愈演愈烈，知识表示作为一种最底层的技术已经成为了科技智能化的基础。因为知识表示既可以让机器像人脑一样存储理解知识，又可以按照一定的规则去处理并运用知识解决实际生活中存在的诸多问题。从某种意义上讲，知识表示是一种计算机可接受的数据结构和处理机制的综合。如常见的链表、表、树、图等均属于知识表示范畴的数据结构类型。

比如，在数学上我们知道任意一个三角形的内角和度数为180度，但是我们怎样将它表示出来呢？这时我们就需要借助数学上统一规定的符号和表示方式将它展示出来，即如下：



这种知识表示既直观又便于理解，因此比文字描述更具有采纳性，且可以将知识通过符号逻辑有机统一。

知识表示在不断发展主要总结出以下几种早期的知识表示方法：1.一阶谓词逻辑；2.产生式表示法；3.框架；4.语义网络。

### 6.1.1 一阶谓词逻辑

一阶谓词逻辑属于逻辑表示法中的内容。逻辑表示法以谓词形式来表示动作的主体和客体，是一种叙述性的知识表示方法，它研究的是假设与结论之间的蕴涵关系，即用逻辑方法推理的规律，可以看成自然语言的一种简化形式，由于它精确、无二义性，容易为计算机理解和操作，同时又与自然语言相似，故而至今仍被广泛采用。逻辑表示法主要包括命题逻辑和谓词逻辑两种。

在提及一阶谓词逻辑前，我们首先需要掌握命题逻辑，因为谓词逻辑是在命题逻辑的基础上衍生而来的。命题逻辑是一种以命题形式表达事实，用逻辑符号描述关系的知识表示方式，命题的定义为用语言、符号、式子表示且可以判断真假的陈述句。我们举如下例对命题加以解释：

**（1）明天下雨 （2）明天会下雨么 （3）明天可能下雨**

以上三句话中只有（3）属于命题。因为命题的关键要素是可以判断真假且为陈述句。

在命题概念中，表达单一意义的命题叫做原子命题，即只表达一个事实，而不是多个事实的集合，在命题逻辑中原子命题是不可细分的整体，作为命题逻辑的基本单元存在。

但是在现实世界中存在很多交叉和具有相似性的事实，如果我们皆用命题逻辑表示，不仅操作复杂占据空间而且存在大量冗余。如下例：

以上为三个极为简单的原子命题，它们刻画的均为同一个概念——大学生，那么如果有上千名大学生需要一一用命题逻辑表达，岂不是费时费力，因此衍生出一种称为一阶谓词逻辑的逻辑表达方式来统一这种具有内在关联的原子命题。

一阶谓词逻辑的思想即为分离原子命题中的主语、谓语、宾语，通过函数形式来表示谓词，而将主语、宾语作为函数的个体元素，这样一来，对于具有相同概念、属性、行为的个体我们均可以通过同一谓词，不同元集的形式表述出来。同时，在谓词逻辑中，我们考虑具有一般性和个别性的事实，从而引入全称量词和存在量词统一表达。

一阶谓词逻辑的一般形式可以表示为：



这里P代表谓词，通常是用来刻画个体从属的概念或具有的属性的词语；x代表个体元素，在一阶谓词中可以为常量、变量和函数；n代表个体数，即谓词的元。当n=1时，为一元谓词，以此类推。

特别地，当x为一阶谓词函数充当个体元素时，P为刻画属性x的谓词函数，此时，P为二阶谓词，以此类推。这里我们举一下简单的例子加以区分：

1. 海伦有一个孩子叫杰克 has\_children(Helen,Jack)
2. 这种红色花朵颜色很鲜艳 bright\_colored(red(flower))

以上两句话中，例（1）为一阶谓词，例（2）为二阶谓词。因为在例（2）中红色作为属性描述花朵的颜色，而鲜艳则用来修饰红色的明暗程度，刻画属性的属性。

另外，为了方便表达一般性事实和特殊性事实，在一阶谓词中引入量词，量词的作用是限定谓词描述的个体与个体域之间的包含关系，分为全称量词和存在量词两种。其中，全称量词表示个体域中的所有个体都服从谓词约束，用符号 表示，存在量词则指部分个体满足谓词约束，用符号表示。下面举例展示它们在一阶谓词逻辑中的用法：

1. 语句：所有的理科生都需要修读高等数学。



1. 语句：13个人中至少存在两个人生日月份相同。



一阶谓词逻辑作为逻辑表示法的重要内容，在知识表示领域具有很强的普适性。它主要具有以下四大优点：（1）自然性：贴合自然语言描述，容易被人接受；（2）精确性：用于表示确切知识；（3）严密性：有严格且系统的形式定义和推理规则；（4）易实现性：容易转化为计算机内部语言。当然，在长期的运用实践中我们也发现了一阶谓词逻辑存在的弊端，有如下三点：（1）无法表示不确定性知识，难以运用于启发性知识和元知识表示；（2）经常出现事实规则等的组合爆炸；（3）推理复杂度通常较高，推理效率低。

### 6.1.2 产生式

产生式通常表示事物的一种因果关系，其基本形式可以表示为

 或***IF P THEN Q CF=*[0,1]**

其中P是产生式的前提，Q是产生式的结论或操作，CF为确定性因子，也叫置信度，通常表示由前提可推导出结论或操作的概率。产生式中由于CF的存在可以表示一些不确定性知识，在谓词逻辑的强规则中是无法实现的。我们可以举例表示产生式的描述方法：

**IF 该人物是第一次工业革命的代表人物**

**发明了一种为工业生产提供动力的机器**

**THEN 该人物为瓦特 CF=0.7**

这里我们以人物瓦特作为例子，IF中为我们做出推断的前提条件，THEN中为我们得到的结论，在该例子中我们认为由前提中的两个条件推导出人物为瓦特的置信度为0.7。

产生式也可以理解由固定逻辑结构来表达知识的方式。在自然语言中，人们广泛使用的逻辑结构主要包括“原因->结果”，“条件->结论”，“前提->操作”，“事实->进展”，“情况->行为”等 ，这些逻辑结构都可以归结为产生式的知识形式。在如下示例中我们对各种逻辑结构进行了举例。

1. 天下雨，地上湿。
2. 如果温度降到0℃以下，水就会结成冰。
3. 给我一根合适的杠杆，我可以翘起地球。
4. 夜来风雨声，花落知多少。

以上例子中，（1）表示下雨使地面湿漉，是一种因果逻辑，（2）则是在0摄氏度条件下水会结成冰，属于条件推导结论，（3）则是在假设前提下的操作，（4）属于事实发生后的进展。

产生式与谓词逻辑具有一定的关联，也存在差异。首先，从逻辑规则上来讲，产生式与谓词逻辑的大部分规则的基本形式是相似的，都是从人类长期总结出的逻辑经验中获得，不过产生式的逻辑更为宽泛，谓词逻辑构成的蕴含式从结构上而言更像是产生式的特例。但是，谓词逻辑的规则只能表示精确知识，其值非真即假，而产生式不仅可以表示精确知识，还可以表示非精确知识，即在通过置信度CF来表示知识的精确度。正是由于这种区别，用产生式表示知识的系统中，“事实”与产生式的“前提”中所规定的条件进行匹配时，可以是“精确匹配”，也可以是基于相似度的“不精确匹配”，只要相似度落入某个预先设定的范围内，即可认为匹配。但对谓词逻辑的规则而言，其匹配必须是精确的。

产生式使用因果逻辑的方式表达知识使得它具有和谓词逻辑一样自然直观的特点。同时，产生式的规则形式相同，易于模块化管理。它能够表示各种知识，包括确定性知识、不确定性知识、启发性知识以及过程性知识。然而，产生式具有规模较大的规则库，对事实进行规则匹配时相当费时，求解和推导时容易产生组合爆炸。另外，产生式只能表达一般的因果逻辑，对结构化知识

无法通过规则表现出知识间的关联性。

### 6.1.3 框架

框架理论最早是由“人工智能之父”之称的马文·明斯基（Minskey）在1975年发表的论文《A framework for representing knowledge》中提出的。它的基本思想是认为人们对现实世界中各种事物的认识都是以一种类似于框架的结构存储在记忆中。当面临一个新事物时，就从记忆中找出一个合适的框架，并根据实际情况对其细节加以修改、补充，从而形成对当前事物的认识。

由此可见，框架是一种描述对象 (事物、事件或概念等)属性的数据结构。这种数据结构在框架理论中是知识表示的基本单位

一般来说，一个框架由若干个“槽”（Slot）结构组成，每个槽又可以细分为若干个“侧面”。在框架中一个槽用于描述所论对象某一方面的属性；一个侧面用于描述相应数学的一个方面。槽和侧面所具有的属性值分别称为槽值和侧面值。在大部分框架中可能还存在某些约束条件，主要用于限定框架描述的对象范围。我们在下面给出一个简单的框架结构示意。

<框架名>

<槽名1>： <侧面名1> 值1，值2，...，值p1

<侧面名2 >值1，值2，...，值p2

.......

<侧面名m1> 值1，值2，...，值pm1

<槽名n>： <侧面名1> 值1，值2，...，值r1

<约束>： 约束条件1

约束条件n

我们在这里给出一个应用上述结构刻画的某人犯罪未遂杀人事件的描述框架。

框架名： t x 未遂杀人案

犯罪意图： x

犯罪结果： 杀人

被杀者： y

杀人动机： x 未遂被 y 发现

知情人： { }

罪犯： t

条件一： 若 x 为强奸，则 t 必须是男性

条件二： 有某个 指控 t

条件三： t 招认

在该框架中我们将杀人未遂案件的结构剖析为多个表征项，其中包括该人物原始的犯罪意图、最终的犯罪结果、罪犯、被杀者、罪犯的杀人动机以及案件相关的知情人等，同时为框架提供支撑其成立的约束条件，如当犯罪行为X为强奸时，那么罪犯t应当为男性；在案件中必须有某个知情人指控t且t招认自己的犯罪行为我们才能认定t为罪犯。类似的例子还有很多，这些框架也为我们在描述很多领域内的知识时提供叙述结构上依据，使得知识表示更为条理化。其实从上述的例子中我们也可以看出，框架表示法更适用于表示一些结构化的知识，它能够把知识内部的构造关系表示出来，同时将与某个实体或实体集的特性集中到一起。

知识内部具有关联，而知识之间同样具有关联，在框架表示法中，知识概念之间的从属关系可以通过层次型框架的方式表示。具体的表示方法大家可以参考下面给出的示例。

框架名：地震

ISA：自然灾害

震源深度：单位（千米）

强度：单位（里氏级）

余震次数：单位（次）

框架名：自然灾害

地点：字符串

时间：单位（年，月，日）

受灾人数：单位（人）

经济损失：单位（万元）

框架名：6.17长宁地震

ISA：地震

地点：四川省宜宾市长宁县

时间：2019年6月17日

震源深度：16km

强度：6.0级

余震次数：182次

受灾人数：16.8万人

经济损失1081.8万元

在上述三幅图中，图一为描述自然灾害的框架，图二为描述地震概念的框架，图三为描述四川长宁地震事件的框架。三者之间具有从属概念关系，在框架描述中我们通常运用ISA槽来表示。因为地震属于自然灾害，所以自然灾害的描述项可以作为底层框架为地震框架所调用，同理，长宁地震就可以调用自然灾害和地震框架来描述我们所关心的事件详细情况。这种层级框架不仅清楚地展现了知识概念间的结构关系，同时也在很大程度上减少了知识描述的冗余。

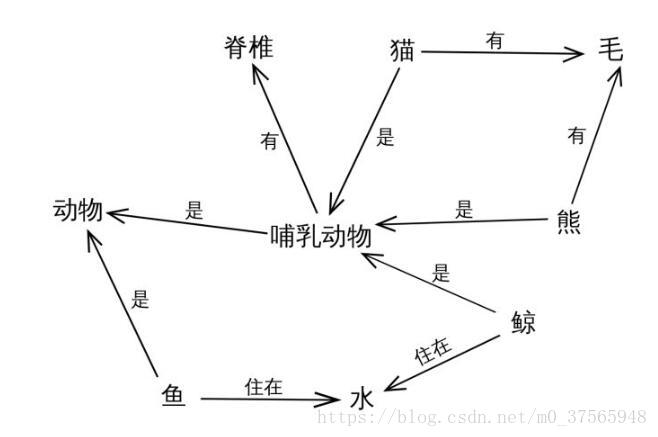
在长期的积累和实践中我们发现，框架表示法由于其结构化的特征，非常适合用于构建领域内的知识库，通常来说基于框架的知识库质量都比较高，因为框架对于知识的描述相对完整和全面，而且框架允许数值叠加和计算，有利于知识中数学方面的处理。然而一个精良的知识框架其构建成本通常是高昂的，对于我们从各种渠道获取的知识质量要求较高，另一方面，由于其结构化特征明显，框架的表达形式相较其他方式更为固化，不够灵活，很难同其他形式的数据知识关联使用，因此无法应用于一些特殊场景。

### 6.1.4 语义网络

语义网络作为一种知识表示方法是在近代自然语言处理系统的发展中衍生出来的。1968年，奎林（J.R.Quillian）在其博士论文中最先提出语义网络，把它作为人类联想记忆的一个显式心理学模型，并在他设计的可教式语言理解器TLC (Teachable Language Comprehenden)中用作知识表示方法。 1972年，西蒙在他的自然语言理解系统中也采用了语义网络表示法。1975年，亨德里克（G.G.Hendrix)又对全称量词的表示提出了语义网络分区技术。语义网络由于灵活性的结构特征而迅速受到知识表示领域的肯定。

那么语义网络究竟是一种怎样的知识表示方法呢？其实，语义网络又叫做联想网络，它是根据人们脑海中对于知识的一种关联记性方法形成的，在结构上语义网络是用实体及其语义关系表达知识的有向图。其中，实体由节点表示，语义关系则由连接节点的弧来表示，又称为“联想弧”。

图以动物为例给出了语义网络的表示方法，这种表示方法既直观又符合人类的认知特点，能够清晰地将知识之间的语义关系展示出来。比如通过下图我们知道了哺乳动物都具有脊椎，猫、熊、鲸鱼都是哺乳动物等。



对于一个普通的语义网络而言，通常是由节点和弧构成。节点常用来表示各种事物、概念、情况、属性、动作、状态等，带有若干属性，而弧则指明它所连接的事物或者概念间存在的某种语义关系。此外，节点还可以是一个语义子网络，形成一个多层次的嵌套结构。如图我们以瓦特作为例子展示了其语义网络的表示，我们可以从语义网络中获取到与瓦特相关的各类属性信息。

起源地

中文名是

国籍是

改良

出生日期

位于

出生在

在语义网络中，实体概念之间的语义关系主要分为以下几类：

1. 类属关系。表示具有共同属性的不同事物之间的分类关系、分类关系或示例关系。体现的是“具体与抽象”、“个体和群体”的概念，如我们常见的“is-a”、“a-kind-of”、“a-member-of”关系等。
2. 包含关系。也被称为聚类关系。是指具有组织或结构特征的“部分和整体”之间的关系。它与类属关系的区别在于包含关系一般不具有属性的继承性。常见的包含关系有“part-of”“piece-of”等。
3. 属性关系。表示事物与其属性之间的关系。常见的属性关系有“have”、“can”等
4. 时序关系。不同事件在其发生时间上的先后次序关系。比如“before”和“after”等。
5. 位置关系。不同事物在其所处空间上的方位关系。比如“in-font-of”和“behind”等。
6. 因果关系。也可以叫做推论关系，两个事物或事件之间具有因果逻辑或可 以由其中一个概念推导出另一个的概念的语义关系。

语义网络以节点和弧形式把事物属性以及事物间的语义联想显式地表示出来，具有较为清晰的结构，符合人类的认知特征。由于它是作为人类联想记忆的模型提出的，所以更能直观地表达知识属性和联系，便于理解，从自然语言到语义网络的转换也比较简单。因此语义网络无论从结构灵活性还是语义表达能力而言都要优于前几种知识表达方式，更能为机器所运用。

但是语义网络也具有缺陷。首先一点是它的弱模式化， 语义网络无公认的形式表示体系，具体知识完全依赖处理程序的解释形式，因此基于语义网络的知识推理无法保证其正确性，在逻辑上也可能不充分，有可能产生二义性。其次是语义网络在处理上的复杂性，语义网络表示知识的手段多种多样，虽然灵活性很高，但同时也由于表示形式的不一致使得对其处理的复杂性提高，对知识的检索也就相对复杂，要求对网络的搜索要有强有力的组织原则。

## 基于RDF的知识表示方法

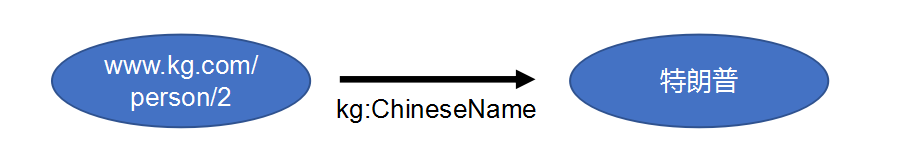
19世纪末随着人工智能的兴起，Web语言逐渐走进人类视角，将知识表示正式带向规范化进程。1989年，英国科学家Tim Berners-Lee发明出万维网（World Wide Web），专家可以将知识以文档形式直接存放到网站上供用户阅览和借鉴，此时人类正式进入Web1.0时代。1994年，万维网联盟（World Wide Web Construction,W3C）成立,成为Web技术领域最具权威和影响力的技术标准机构，它发布了互联网信息的内容标记语言HTML和XML，为互联网环境下的知识表示和共享奠定了基础。随后，一般用户也逐渐参与到网站内容的生成和共享中，万维网逐渐从封闭化管理变为开放互联，由集中处理变为分布式开发，由专家定义变为群体智慧，互联网正式占据主导地位，Web2.0时代到来。



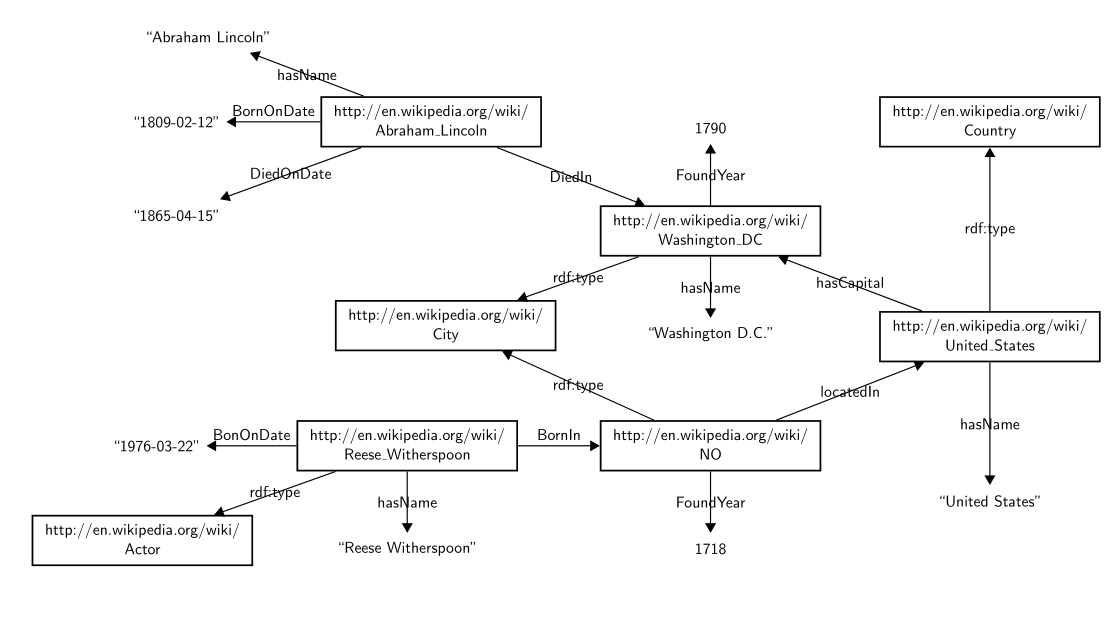
然而随着互联网进入人们的工作生活，人类对于互联网的知识需求也在不断提升，如何满足现实生活和工作需求成为了互联网发展的重心。2000年，Tim Berners-Lee正式提出语义网的概念，将人们生活事务、交易活动中涉及到的一切网络数据包括内容、链接等都在计算机上得以理解和处理。人类从此进入智能化的Web3.0时代。

语义网在知识表示领域提供了一种更为简单、有效的数据模型，被称为RDF（Resource Description Framework）。在RDF中一切可以在Web上描述的事物（可以是具体的事物也可以是抽象的概念）比如书（具体）、计算机（具体）、量子力学（抽象）等统称为“资源”，用URI来表示，URI是一个用来标识资源的字符串，它是万维网体系结构的重要组成部分，我们常用的网址叫做统一资源定位符（URL），是URI的一种。属性则通常用来描述资源的关系

，同样使用URI来标识。以下展示了一个简单的RDF图，表示资www.kg.com/person/2具有属性ChineseName，且其对应的属性值为“特朗普”。



RDF作为数据模型，常以三元组形式构成。其中主语由表示某个事物的实体构成；谓语由实体的属性或与其他实体间的逻辑关系构成；宾语可以是属性值也可以是其他实体。其中值得需要注意的是在RDF中宾语可以是空白节点，当谓语为属性时，宾语通常为一个常量，而当谓语表达为实体间的语义关系时，宾语则通常为一个URI。如图我们以Abraham Lincoln为例展示了以RDF数据模型构造出的语义图。



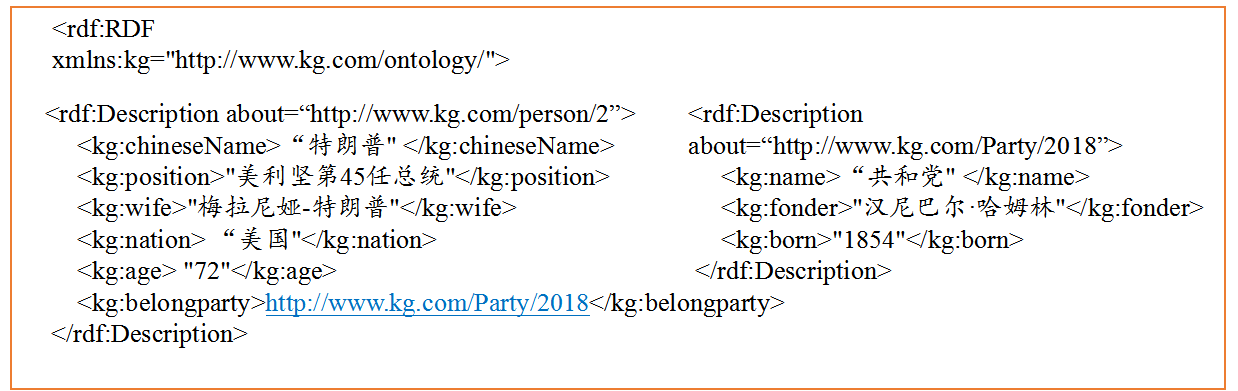
从图中我们可以看出在RDF数据模型下所有的实体都被简化为了一个URI，我们可以在对应的URI中获取到实体的相关属性信息，其中rdf:type表示的为所属关系，相当于is-a关系，如图中Reese Witherspoon通过rdf:type指向Actor表示Reese Witherspoon是一名演员。

需要注意的是，RDF只是一种用于表示Web资源的数据模型和框架，而并非一种表示形式，常见的基于RDF的序列化表示方式主要包括以下几种：

RDF/XML,N-Triples,Turtle。下面我们会以美国总统特朗普的RDF数据为例来展示这几种表示形式的差异。

1. RDF/XML

RDF/XML是用XML的格式来表示RDF数据。XML的技术比较成熟，有许多现成的工具来存储和解析XML，为数据分析提供便利。然而，XML的格式过于冗长，不便于阅读，因此使用较少。如图，左边的RDF数据展示了特朗普的相关属性信息，最后一行中belongparty属性下为一个URI，该URI表明特朗普所属党派——共和党为一个概念，右边即为共和党概念的具体属性信息。



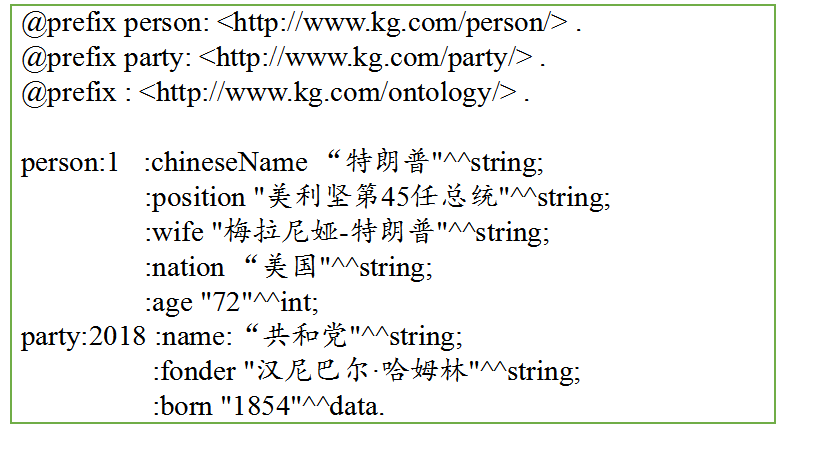
1. N-Triples

N-Triples，从语义上理解即为用多个三元组来表示RDF数据集，是最直观的表示方法。在文件中，每一行表示一个三元组，方便机器解析和处理。这种表示方法相对于XML的形式简化了许多更为高需求处理所接受。



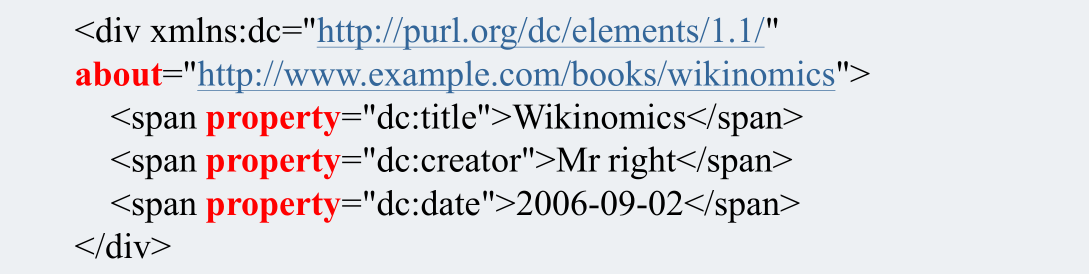
1. Turtle

Turtle更类似于我们现实中使用的编程语言的表示方式，它会在头部先声明出用到的概念或实体以及它们的URI，再在后面依次对应每个概念和实体声明它们的属性部分。它比RDF/XML更为紧凑，且可读性比N-Triples好。

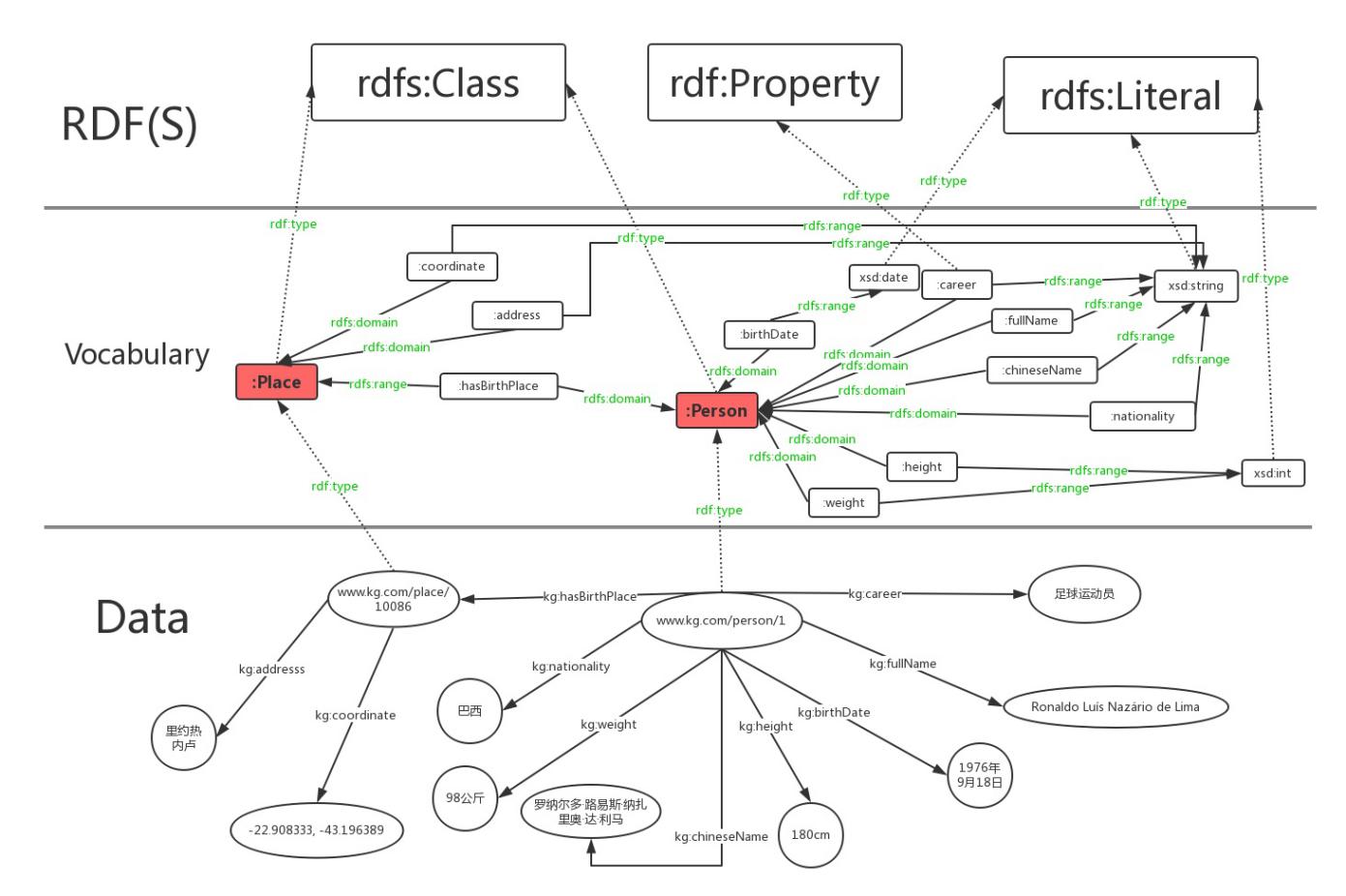


为了从机器可理解的层面优化搜索，提升访问性以及网页数据的关联性。

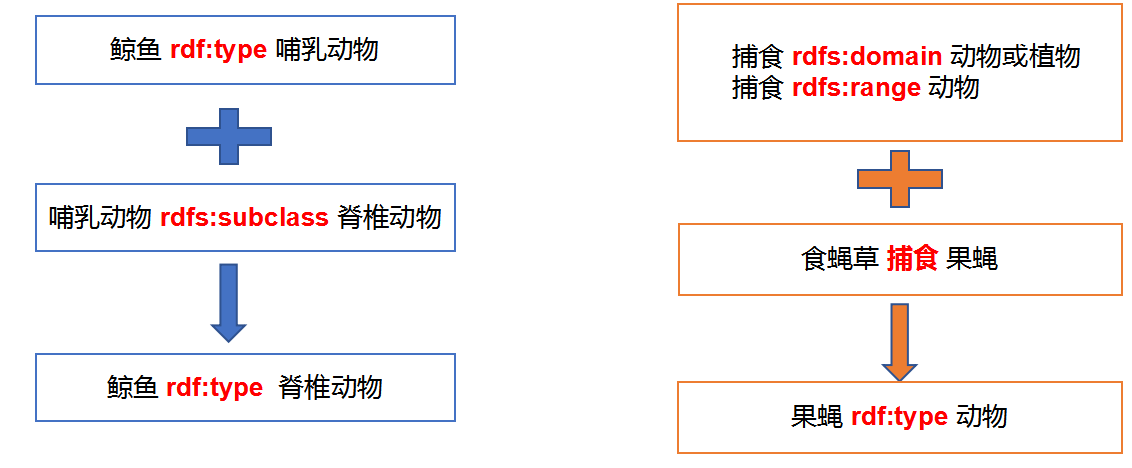
RDFa被添加到RDF的知识表示构建中，它的本质是为RDF提供一些需要的属性，这些属性载有后设资料可以被用在XML文件中，并被支持RDFa技术的浏览器或者搜索引擎解析出来。如图一段基于HTML表示的RDF数据中，引入了about属性以及property属性，机器可以通过检索about标签识别下面这段语句的描述对象为“http://www.example.com/books/wikinomics”，再通过检索property属性识别出对象的标题、创建者以及创建日期属性。



尽管RDF引入了URI使得知识可以作为资源存在于Web中并相互关联，但是RDF的框架过于简单，我们无法通过RDF判断出一个资源是否是类（概念）、属性，以及确定资源的描述范围等。因此在RDF的基础上人们又引入了一种新的知识表示方法RDFs。RDFs在RDF数据层的基础上引入模式层，定义类、属性、关系、属性的定义域与值域(Class,subClassOf, Property,subPropertyOf,Domain, Range)来描述与约束资源,构建最基本的类层次体系和属性体系，支持简单的上下位推理。



如图所示，在RDF数据层之上引入RDFs后可以明确指出URI对应的概念和属性，如Place、Person等，并通过定义域与值域限定我们所用属性的描述范围，使得RDF的知识表示能力进一步增强。有了RDFs的模式层定义后，我们可以进行一些简单的推理。如下左图是通过模式层中的类进行推理，我们知道鲸鱼是一种哺乳动物，哺乳动物又是脊椎动物的子类，因此我们可以推断出鲸鱼是一种脊椎动物；右图是基于属性的定义域和值域的推理，我们已知了捕食动作的主语为动物和植物，宾语只能是动物，那么对于食蝇草捕食果蝇这句话我们就可以推断出果蝇是一种动物。



基于类型的推理 基于属性定义域与值域推理

虽然RDF(S)可以表示一些简单的语义具有一定的推理能力，但在更复杂的场景下，RDF(S)语义表达能力显得太弱，还缺少诸多常用的特征。

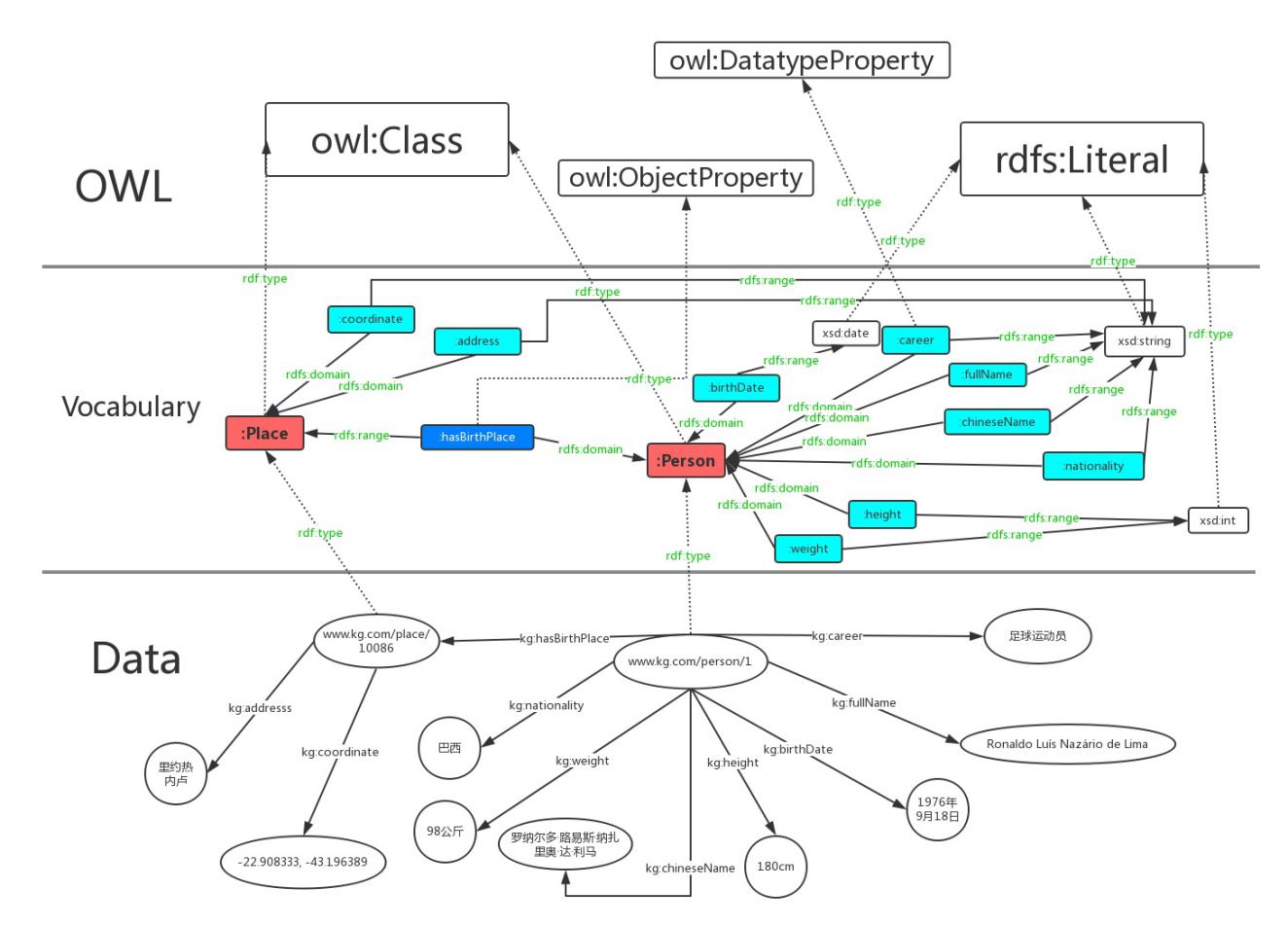
（1）对于局部值域的属性定义：RDF(S)中通过rdfs:range定义了属性的值域，该值域是全局性的，无法说明该属性应用于某些具体的类时具有的特殊值域限制。

（2）类 、 属性 、个体的等价性：RDF(S)中无法声明两个或多个类、属性和个体是等价还是不等价 。

（3）不相交类的定义： 在RDF(S)中只能声明子类关系，如男人和女人都是人的子类，但无法声明这两个类是不相交的 。

因此在RDFs的基础上又引入了新的表示方法OWL。它进一步扩展RDFS词汇，可声明类间互斥关系、属性的传递性等复杂语义，支持基于本体的自动推理，提供了一组合适Web传播的描述逻辑的语法，对机器友好，但由于认知复杂性的存在限制了其在工程上的应用。

如图OWL的引入丰富了属性层定义，如hasBirthPlace作为客体属性存在，因为它连接的是两个实体，而career等则作为数据类型属性存在，因为它此时连接的是实体与其所具有的特征值。



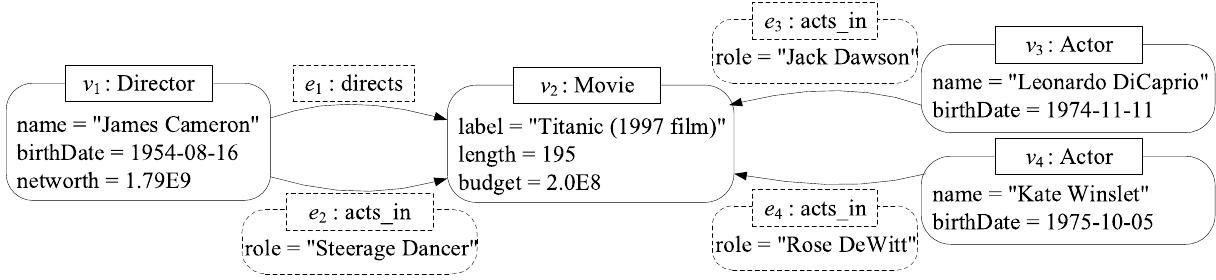
## 其他知识表示方法

* + 1. **基于属性图的知识表示**

属性图也是知识表示及图谱构建中常用的一种数据模型。与RDF数据模型相比，属性图模型对顶点属性和边属性具备内在支持。在RDF图中，属性通常作为一个独立的边存在，而在属性图中，凡是实体自身属性一律作为实体节点具备的内在特征而存在，不再以边的形式表示，这样当我们需要对属性进行操作时，不再需要改变图的结构，而只需要对具备属性的节点进行操作即可，同时实体间关系也可以具有一定内在属性，从而易于区分关系类别。

一个属性图G可以理解为一个五元组 ，其中：(1) V 是顶点或顶点的有限集合；(2) E 是边的有限集合，且；（3）函数将边关联到顶点对，如表示e是从顶点到顶点的有向边；(4) 设 Lab 是标签集合，函数为顶点和边赋予标签，如，则 l 为顶点 v的标签；(5) 设 Prop 是属性集合，Val 是值集合，函数 为顶点或边关联属性，如，则val 为顶点 v上属性 p 的值。

如图给出了基于属性图的电影知识图谱。在属性图中，每一个节点和边都有一个唯一标识的id，如图等，所有的节点和边又具有相应的标签，如的标签为Director，的标签为directs，这里标签的作用就相当于RDF中的资源类型，主要用于区分节点类型，方便查询；顶点和边又具有各自的属性，属性由属性名和属性值构成，如图顶点属性name、birthDate和networth，其中属性name值为“James Cameron”。



电影知识图谱

由于属性图相较RDF更为灵活方便，可以一定程度上降低知识表示的复杂性，在很多场景下可以避免对图结构的更改，而被工业界所广泛推崇。工具上，现有的著名图数据库Neo4j便是利用属性图的结构来存储和表示数据，其存储管理层为属性图结构中的节点、节点属性、边、边属性等设计了专门的存储方案，这使得 Neo4j 在存储层对于图数据的存取效率天生就优于关系数据库。

另外，Neo4j也为属性图数据提供了相应的查询语言Cypher，Cypher是一种声明式的查询语言，用户只需要声明查询的目标而无需声明查询的方式，它借鉴了SQL语法，查询语义易于理解，Cypher 的一个主要特点是使用“ASCII 艺术”（ASCII art）语法表达图模式匹配。如下基于上图的电影知识图谱展示Cypher的基本功能。

（1）查询 James Cameron 执导的电影及其片名。

查询语句：

MATCH (:Director {name: "James Cameron"})-[:directs]->(x:Movie)

RETURN x, x.label

查询结果：



（2）查询 1950 年之后出生的资产大于 1.0E9 的导演执导的电影的出演演员。

查询语句：

MATCH (x1:Director)-[:directs]->(:Movie)<-[:acts\_in]-(x2:Actor)

WHERE x1.networth > 1.0E9 AND x1.birthDate >= date("1950-01-01")

RETURN x2

查询结果：



* + 1. **基于本体的知识表示方法**

本体的概念源自于哲学领域，在哲学中的定义为“对世界上客观事物的系统描述，即存在论”。哲学中的本体关心的是现实的抽象本质，而在计算机领域，本体可以在语义层次上描述知识，可以看成描述某个学科领域知识的通用概念模型。德国学者Studer在1998年提出知识表示中本体的相关定义“本体是共享概念模型的形式化规范说明”。定义包含了四层含义：共享、概念化。明确性和形式化。

（1）共享：指在本体中提供的知识是共同认可的，反映在领域中公认的术语集合。

（2）概念化：指本体对于事物的描述表示成一组概念。

（3）明确性：指本体中全部的术语、属性及公理都有明确的定义。

（4）形式化：指本体能够被计算机所处理，是计算机可读的。

本体可以理解为学科领域的术语以及术语之间的关系，也可以理解为概念的集合以及概念之间的关系。如下简单展示了生物学上的物种分类本体。

腔肠动物

无脊椎动物

软体动物

节肢动物

动物

扁形动物

环节动物

鱼类

两栖动物

脊椎动物

爬行动物

哺乳动物

鸟类

本体是对领域实体存在本质的抽象，他强调实体间的关联，并通过多种知识表示元素将这些关联表达和反映出来，这些知识表示元素也被称为元本体，主要包括：

1. 概念——表示领域知识元，包括一般意义上的概念以及任务、功能、策略、行为、过程等，在本体的实现中，概念通常用类（class）来定义，而且通常具有一定的分类层次关系；
2. 属性——描述概念的性质，是一个概念区别于其他概念的特征，通常用槽（slot）或者类的属性（Properties）来定义；
3. 关系——表示概念之间的关联，例如一些常用的关联：父关系、子关系、相等关系；
4. 函数——表示一类特殊的关系，即由前n-1个要素来唯一决定第n个要素，如：长方形的长和宽唯一决定其面积；
5. 公理——表示永真式，在本体论中，对于属性、关系和函数都具有一定的关联和约束，这些约束就是公理，公理一般用槽的侧面（fact）来定义；
6. 实例——表示某个概念类的具体实体。

一般来说本体中概念之间主要存在四种基本关系，如下表：

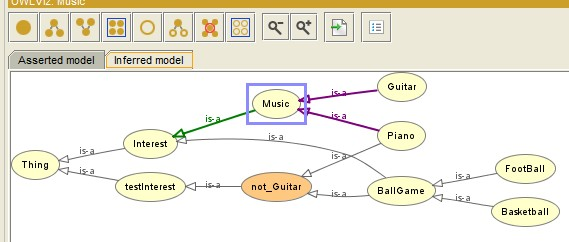
|  |  |
| --- | --- |
| 关系 | 关系描述 |
| Part-of | 局部与整体的关系 |
| Kind-of | 父类与子类的关系 |
| Instance-of | 在类中填充实例，类与实例之间的关系 |
| Attribute-of | 类的属性，有对象属性和数据属性 |

本体作为一种知识表示方法，与谓词逻辑、框架等其他方法的区别在于他们属于不同层次的知识表示方法，本体表达了概念的结构、概念之间的关系等领域中实体的固有特征，即“共享概念化”，而其他的知识表示方法如语义网络等，可以表达某个体对实体的认识，不一定是实体的固有特征。这正是本体层与其它层的知识表示方法的本质区别。

除此之外针对本体的构建方法也是多种多样，常见的有如下两种：

1. 七步法：斯坦福大学医学院开发的七步法，用于领域本体构建。七步骤为：确定领域本体的范畴；复用现有的本体；列出领域内的术语；定义类和类的等级关系；定义类的属性；定义属性的分面；填充实例。
2. 基于叙词表的领域本体构建：叙词表又称为主题词表，是一种语义词典，由术语及术语之间的关系组成，能够反映某学科领域的语义相关概念。由于叙词表包含丰富的领域概念和一定的语义关系，在表达和知识结构上与本体有着天然联系，包含了本学科比较完整的术语，因此，国内外很多学者都在尝试将叙词表转换为本体。国内目前基于叙词表已经转化成型的本体原型有《国防科学技术叙词表》和《中国农业科学叙词表》。

大规模领域本体的构建通常比较复杂，耗时耗力，而且需要具备领域背景知识的专家参与。手工构建一个规模化的领域本体是不可能的，为了方便领域本体构建，斯坦福大学医学院开发出了一款半自动化构建领域本体的工具Protege,它是基于Java语言实现的，提供了一个图形化和交互式的知识本体开发环境。

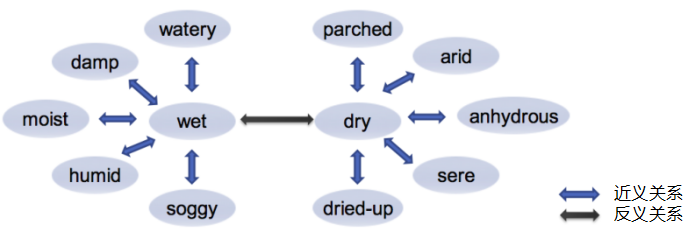


## 知识表示与建模实例

知识表示技术为知识建模提供了基础，而目前国内外众多行业和领域都需要大规模的知识集成和应用，为具体的信息化服务提供基础，因此知识表示被各领域广泛使用。其中面向语言领域的知识库最为基础，也为后续其他领域的知识库构建开辟了先河。语言领域中两个尤为突出的知识库是WordNet和ConceptNet，它们都是面向词语语义，基于RDF数据模型构建的知识库。

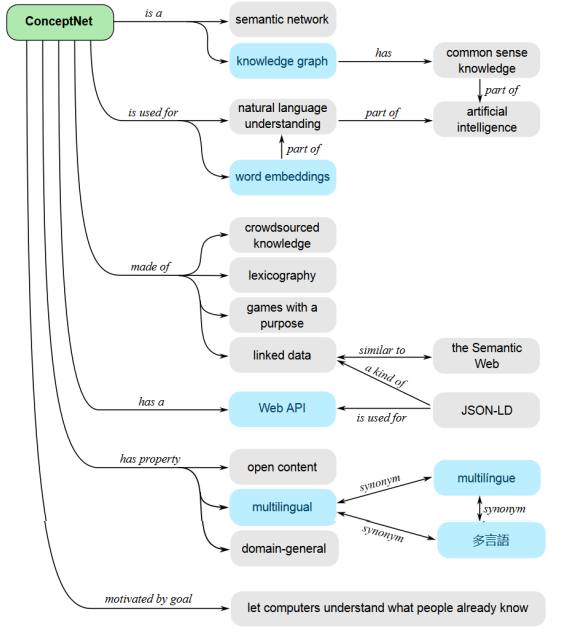
1. WordNet

WordNet是由Princeton 大学的心理学家，语言学家和计算机工程师联合设计的一种基于认知语言学的英语词典。它不是光把单词以字母顺序排列，而且按照单词的意义组成一个“单词的网络”。它是一个覆盖范围宽广的英语词汇语义网。名词，动词，形容词和副词各自被组织成一个同义词的网络，每个同义词集合都代表一个基本的语义概念，并且这些集合之间也由各种关系连接,比如当两个同义词网络的核心词汇恰好语义相反时，这时两个同义词网络具有反义关系。目前，WordNet：155,327个单词，同义词集117,597个，同义词集之间有22种关系



1. ConceptNet

ConceptNet是常识知识库。最早源于MIT媒体实验室的Open Mind Common Sense (OMCS)项目。它主要依靠互联网众包、专家创建和游戏三种方法来构建。ConceptNet知识库以RDF三元组形式的关系型知识构成。目前，ConceptNet5版本已经包含有2800万关系描述。ConceptNet比较侧重于词与词之间的关系。从这个角度看，ConceptNet更加接近于WordNet，但是又比WordNet包含的关系类型多。



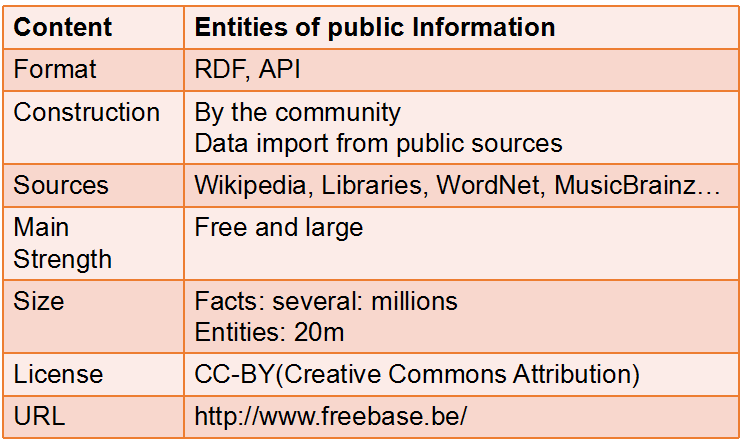
在信息交互日益成熟的背景下，为了满足用户对知识概念的搜索需求，百科类知识库应运而生。其中以维基百科作为代表的大型知识库系统基于本体框架和RDF语言创建了对于常用概念的语义解释并对概念通过关系加以链接，使得知识得以融合和利用，也在很大程度上提升了知识表示的质量。

维基百科（Wikipedia）是一个自由内容、公开编辑且多语言的网络百科全书协作计划，它透过Wiki技术使得所有人都可以简单地使用网页浏览器修改其中的内容。根据不同的知识领域对概念知识进行划分，再根据人们对概念的习惯性特征认知结构去展示概念的语义，维基百科注重对知识本体的构建，通过明确化的定义、分类、阐释来完整地展示概念的本体架构。后续越来越多的知识库开发也都是在维基百科的基础之上形成，并通过一定的知识结构更新和内容丰富化来完善知识库系统，具有代表性的如FreeBase、DBpedia以及Yago。



1. FreeBase

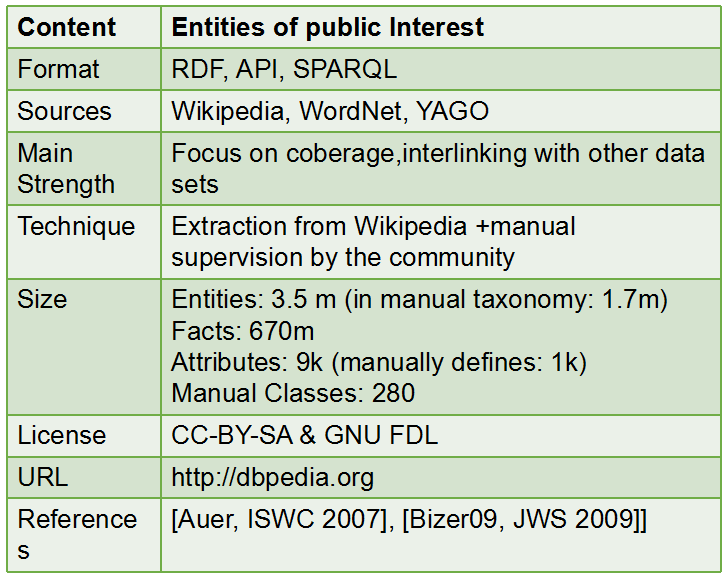
FreeBase是2007年Metaweb公司发布的知识库，于2010年被Google收购。它也是一种大规模协同构建的知识库，其数据来源主要为Wikipedia以及其他数据源(如 IMDB、MusicBrainz)。FreeBase在知识建模上的主要特征为注重知识的结构化表示。



1. DBpedia

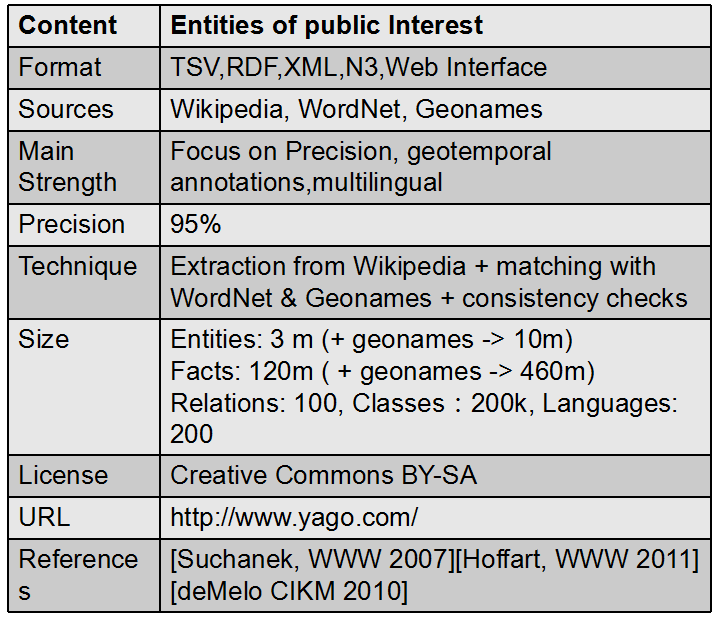
DBpedai于2007年开放，其目标是构建一个社区，通过社区成员定义和撰写准确的抽取模板，进而从维基百科中抽取结构信息，并将其发布到Web上。

其通过人工方式构建分类，包含了280个概念类别，覆盖了约50%的维基百科实体。

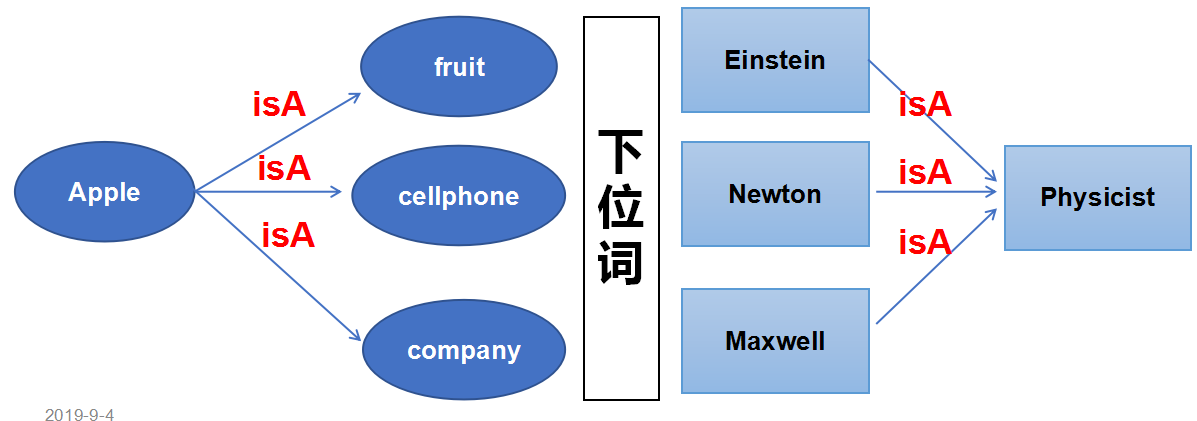


1. Yago

Yago是2007年由德国马普研究所发起创建的大型知识库。它融合了WordNet和Wikipedia的丰富语义和属性特征，并通过人工采样评估对知识质量做出了一定的改善。目前Yago知识库有超过一亿的事实三元组和100出了多种属性关系。

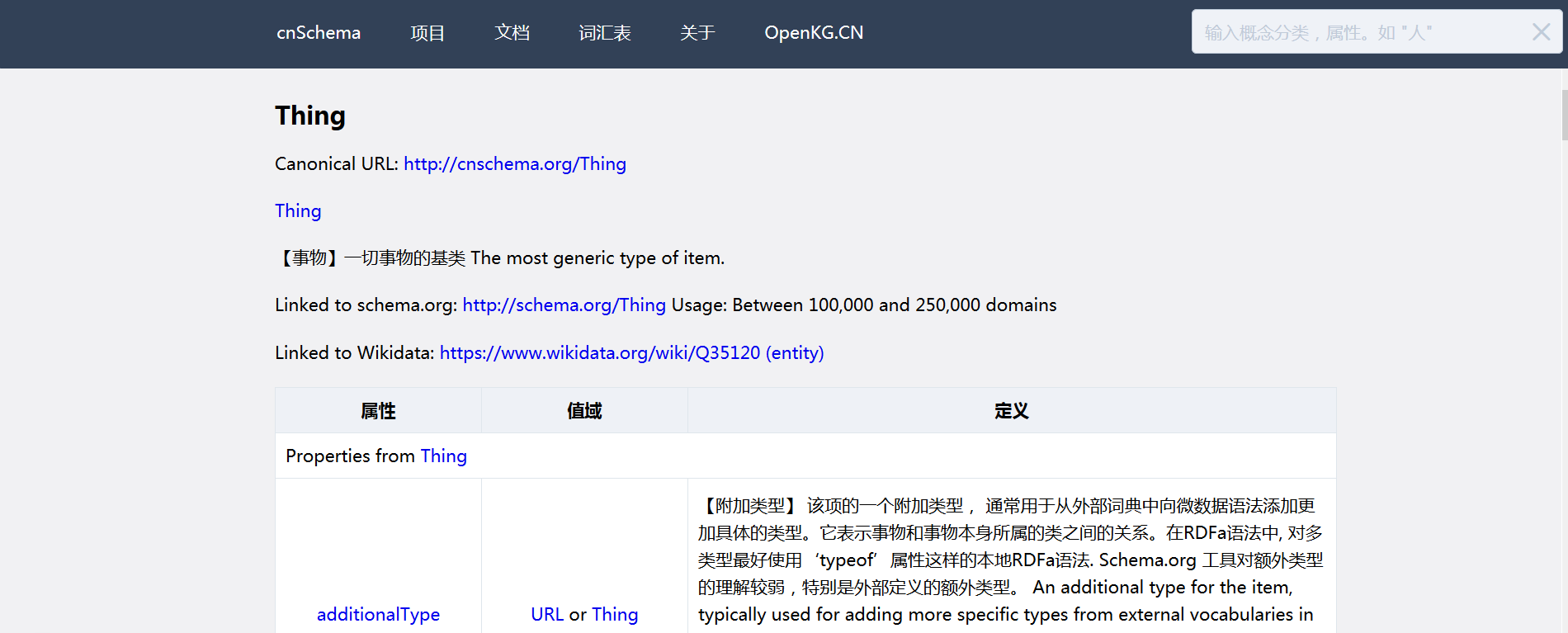


除了大规模的知识库构建外，一些专注于模式层的知识图谱应用是知识表示的典型案例，它们通常专注于高质量的概念性知识的表示和关联，作为领域内其他知识图谱应用的参考。其中ProBase尤具代表性，它是由微软亚洲研究院发布的一个大型英文概念知识图谱。Probase的结构规模庞大，包含有千万级别的概念和实体，以及千万级别的从语料中抽取的关系。其最主要的特征是只关注于现实世界中存在IsA关系的概念和实体，它可以是概念与实体间的包含关系也可以是概念与概念间的包含关系，这种集中式的类属关系表达尤其适合RDF模型的应用和拓展。



而在国内，为了便于开放领域知识图谱的开发，中文信息学会语言与知识计算委员会也大力推动了OpenKG等知识图谱项目，其中CN-Schmea就是在OpenKG中发布的一项重要的知识图谱Schema标准。它包括了上千种概念分类(classes)、数据类型(data types)、属性(propertities)和关系(relations)等常用概念定义，以支持知识图谱数据的通用性、复用性和流动性。CN-Schema也为中文领域的开放知识图谱提供了可供参考和扩展的数据描述和接口定义标准。其主要特征包含四点：

1. 完全开放。普通用户可以直接访问也可以参与到Schema的构建中；
2. 立足中文，对接世界。概念对象为中文词汇，同时基于schema.org与英文等词汇相互关联，语义来源具有充分的依据和参考性。
3. 面向应用，支持开放数据生态。通过概念的精准定义为开放域知识图谱提供本体规范。
4. 社区共识，知识图谱专家指导。通过群体智慧和专业评估提高质量。



CN-Schema的核心数据结构包括三点：数据、元数据以及对接NLP。其中数据是Schema层的核心，它是由实体、实体描述概念的本体框架、框架具有属性的相关陈述构成，其中实体与属性之间的联系又由相应的二元关系以及更为复杂的关系来描述。

CN-Schema支持知识表示的三个层次：逻辑层，存储层和计算层，其主要侧重于前两层。逻辑层上支持RDF，WIDE TABLE,Property Graph逻辑存储方案，支持互相映射；存储层上支持JSON-LD， N-TRIPLE，EXCEL/CSV 序列化文件交换格式；计算层上支持显式（实体关系网络）与隐式（分布式表征）的知识关联与推理。

知识表示除了广泛地应用于概念性知识图谱的构建，也在很多特定领域的知识本体建模中发挥着至关重要的作用。其中最主要的领域包括医疗、金融以及文学。我们将分别在这三个领域中简单介绍具有代表性的知识图谱实例以加深大家的理解。

医疗领域中，为了服务于公共医疗体系，美国国立医学图书馆（NLM）研究和开发了一体化医学语言系统UMLS。UMLS是计算机化的情报检索语言集成系统，它不仅是语言翻译、自然语言处理及语言规范化的工具，而且是实现跨数据库检索的词汇转换系统，可以帮助用户在连接情报源的同时，对其中的电子式生物医学情报作一体化检索。



UMLS的结构主要包括以下四个部分：

（1）超级叙词表：是UMLS知识源的核心，由来自各种受控词表的概念和术语以及它们 的关系所构成；

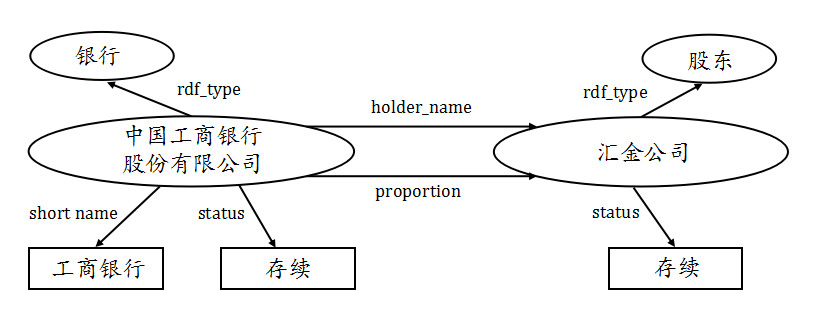
（2）语义网络：是对超级叙词表中概念的分类以及分类之间的关系

（3）专家辞典：是一个词典信息库，用于自然语言处理；

（4）支持性的软件和工具：各种利用UMLS的工具和程序

在金融方面，基于RDF的知识表示方法构建出金融本体，金融本体为金融领域提供了统一的知识表示模型，用来描述金融概念间的语义关系。其中金融本体主要包括静态本体和社会本体，其中静态本体主要包括实体、机构、金融合约、事件、时间和空间，表达的是一种金融概念的上下位关系；社会本体包括直接金融关系、间接金融关系、服务关系等，表达的是金融机构的各种社会关系。金融本体的应用有利于金融概念共享、数据交换和多源数据的融合。

目前金融知识图谱中最具影响力的就是股权关系图谱，其涉及的金融主体繁多，包含银行、保险、证券、信托、期货、资管、基金等，股权网络规模庞大、结构复杂，由众多的金融机构股权网络集合形成，而在单个金融股权网络中又存在股东、股东的股东等层次型实体。图展示了一个用RDF数据模型表示的简单股权关系。



在股权网络中引入了丰富的语义信息以加强知识表示能力。其中以节点和边为例语义表现如下：

结点：包含类型、行业、注册资本、人员、历史变

边：1. 控股——持股占比大于50%或单一最大股东

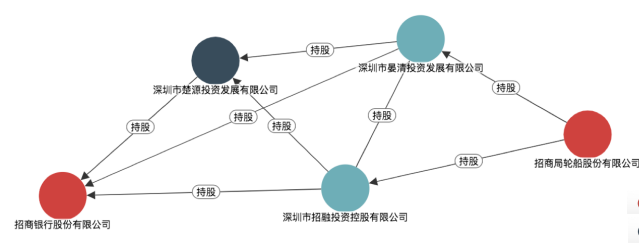
2. 共同股东——A、B有共同（高层级）股东C

3. 一致行动——A、B共同股东C有控制权（或其他信息）

4. 关联交易——e.g. 向股东借款、担保

5. 共同人员——A、B有共同高管、董事

图中展示了由以上语义信息构成的股权网络关系图。



在文化领域，最新出现的唐诗知识图谱尤其引人注目。它基于RDF/RDFS三元组表示方法构建出诗歌-诗人二元本体模型。在模型中，针对每一首唐诗，将诗歌和诗人作为唐诗知识的核心实体，很自然的诗歌与诗人之间是创作关系，对于诗歌而言，其具有丰富的内部属性，可以与其他诗歌相互关联，而诗人与诗人之间又存在着一定的社会关系或流派关系。如何展示了唐诗知识图谱的本体结构框架，其丰富的语义属性和关系将进一步提高了唐诗知识的文学解释性，也打破了唐诗知识的碎片化状态。



唐诗知识图谱在二元实体模型的基础上又引入了诗人的“经历”作为第三实体，划分四个规范类即人物、地点、时间和事件来描述一个完整的经历。如图展示了诗人经历的本体模型，因为人物部分已经包含在二元模型中，所以在这里，主要列举了时间、地点以及事件的相关属性，其表达的具体含义如下：

（1）地点（Location）：对应现实世界中存在着或存在过的空间实体；

（2）时间（Time）存在中国历史纪年与公元纪年的复杂转换问题；

（3）事件（Event）具有时序化的特点，事件的发生会影响诗人诗歌创作内容不同。



1. 早期知识表示方法主要有哪几种？它们之间的区别什么？

答：早期的知识表示方法主要包括一阶谓词逻辑、产生式、框架和语义网络。

其中一阶谓词逻辑侧重于使用逻辑语言描述精确知识；产生式则侧重于使用一定的规则模板结合确定性因子表示非精确的因果逻辑知识；框架侧重于描述结构化实体和概念知识，其知识结构相对完整；语义网络来源人类联想记忆法，侧重于通过有向图的形式表述语义关系。

1. 用框架表示法来表示以下内容：

李华是武汉大学2015级人工智能专业学生，他是一名汉族男生，今年21岁，家住北京朝阳区，父亲李刚是公务员，母亲刘红为银行职员。

|  |  |
| --- | --- |
| 姓名 | 李华 |
| 就读学校 | 武汉大学 |
| 所修专业 | 人工智能 |
| 年级 | 2015级 |
| 性别 | 男 |
| 民族 | 汉 |
| 年龄 | 21 |
| 居住地 | 北京市朝阳区 |
| 父母姓名 | 李刚、刘红 |
| 父母职业 | 公务员、银行职员 |

1. 基于RDF的知识表示和基于属性图的知识表示方法有什么优缺点？

答：基于RDF的知识表示方法源于语义网络，具有规范化的模式层表达和语义关系表述，它通过三元组的形式能够更清晰地展示语义概念关系，具有很强的适应性，但是RDF数据通常会耗费大量存储资源，当需要更新或删除知识时，通常需要修改整个语义网络，成本过高。属性图将属性信息蕴含于节点信息中，单一实体更新只需要修改节点内部信息即可，避免了更改整个数据结构的困扰，但是由于节点承载的信息量多，节点丢失造成的代价就会相对更大。

1. 请列举你所了解的一些前沿的知识表示与建模案例，并加以分析。

答：略。

# 知识抽取

## 什么是知识抽取

知识图谱的构建通常包括知识表示建模、知识抽取、知识融合、知识推理这几个部分。知识抽取指的是自动从数据中抽取实体、实体属性、实体关系、事件等信息并形成结构化数据输出。这里的文本数据可以是结构化数据、半结构化数据，也可以是非结构化数据。结构化数据和半结构化数据的抽取方法较为成熟。然而，实际任务中的数据通常是非结构化的数据，这些数据规模大、表达形式复杂多样，是知识抽取的难点。

知识抽取的子任务通常包括实体抽取、关系抽取、概念抽取和事件抽取。本章会在接下来的部分，分别讲述这四类抽取的方法。

## 结构化数据

所谓结构化数据就是数据库中的关系表数据，也称作行数据，可以用二维表结构来逻辑表达实现的数据。各项之间存在明确的关系名称和对应关系，同时严格地遵循数据格式与长度规范。能够作为高质量知识的来源，在实际的存储过程中，主要是存储在诸如MySQL或Oracle的关系型数据库中统一进行管理。比如现在有一张关于詹姆斯·瓦特的个人信息和相关人员信息的关系表存储在关系数据库中，表的结构和内容如下图7.1所示。

|  |  |  |  |
| --- | --- | --- | --- |
| name | birthDate | birthPlace | type |
| James\_Watt | 1739-01-19 | Greenock | Scientist |
| Adam\_Smith | 1723-06-16 | Kirkcaldy | Philosopher |
| Joseph\_Black | 1728-04-16 | Bordeaux | doctor |

图7.1

可以清楚的看出数据形式化存储在数据库中，每一个列都有具体的含义，分别代表人名，出生日期，出生地和职业。

结构化数据特点

由上面的例子我们可以看出，结构化数据存储起来很严格，置信度比较高。目前许多应用也是基于结构化数据，在网站进行展示的过程中使用结构化数据能让网站中的搜索结果清晰明了。目前搜索引擎都支持标准的结构化数据标记，以便为用户提供更好的上网体验。结构化数据是“无模式”的。更准确地说，其数据是自描述的。它携带了关于其模式的信息，并且这样的模式可以随时间在单一数据库内任意改变[1]

同时，结构化数据的存储规模一般比较小，大部分的数据是没有结构化的。而且由于结构化数据严格遵循格式与长度规范，因此缺乏个性化的属性信息，无法随意的添加属性。

结构化数据知识获取

结构化数据知识获取是把原始数据类型为结构化的数据转化为转化为RDF或其他形式的知识库内容。在知识图谱构建过程中，一个重要的高质量知识来源是企业或者机构自己的关系数据库。为了将这些结构化的历史数据融人到知识图谱中，可以采用资源描述框架(RDF)作为数据模型。[2]由于结构化数据有明显的结构特征，各项之间存在明确的关系名称和对应关系。因此可以把关系库中的主键值作为知识库中的主语，其他键的值作为宾语，其他键的字段名作为谓语，形成主谓宾三元组结构。对于多表转化，可以根据外键代表的表间关系实现谓语关系的转化。

结构化数据的知识获取可以依据W3C于2012年推出的RDB2RDF working group制定的标准，包括Direct Mapping(A direct mapping ofrelational data to RDF)和R2RML(RDB to RDFmapping language)两种映射标准。Direct Mapping的基本思想是通过明确关系模式中编码的语义，将关系数据直接映射转换为RDF，在RDF图中所用到的用于表示类和谓词的术语与关系数据库中的表名和字段名保持一致。这种思想主要适用于关系数据标准化良好，有主键，外键等。

本质上是通过编写启发式规则将数据库中的表转换为RDF三元组。其对应规则如下：表对应类，列对应属性，行对应资源，表单元格值对应文字值，存在外键约束的表单元格对应URI。下图7.2和7.3介绍了如何使用Direct Mapping思想进行结构化数据的转化。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | name | birthDate | →birthPlace | type |
| 1 | James\_Watt | 1739-01-19 | 10 | Scientist |
| 2 | Adam\_Smith | 1723-06-16 | 13 | Philosopher |
| 3 | Joseph\_Black | 1728-04-16 | 16 | doctor |

图7.2 people表

|  |  |  |
| --- | --- | --- |
| id | Place | Country |
| 10 | Greenock | [Britain](javascript:;) |
| 13 | Kirkcaldy | [Britain](javascript:;) |
| 16 | Bordeaux | [France](javascript:;) |

图7.3 place表

有两个表分别为people表和place表，其中两张表中的id均为主键，people表中的birthPlace为place表的外键。在转化时，关系表名转化为RDF类名：people→< People>，关系表中的列名转化为RDF中的属性名:<people#id> < people# birthPlace >，由外键引起的引用转化为额外属性：<People#ref-a birthPlace >。

另外一种转化思想是R2RML，它是关系数据转化为RDF数据的特殊的自定义映射语言，这种映射提供了查看RDF数据模型中现有关系数据的能力。R2RML映射本身就是RDF图，并以Turtle语法写下来，支持不同类型的映射实现。映射的任务是将逻辑表作为输入，然后依据规则triples map将其转换成三元组的集合。基于R2RML语言的方式相较于direct mapping更为灵活，具有很强的定制性。允许为给定的数据库结构定制词汇表，可以将关系数据库通过R2RML映射为RDF数据集，其中所用的术语如类的名称，谓词均来自定义词汇表。D2RQ平台是一个以虚拟只读RDF图的形式访问关系数据库的系统。它提供了对关系数据库内容的基于RDF的访问，而无需将其复制到RDF存储中。使用D2RQ可以: 用SPARQL查询非RDF数据库，以RDF格式创建数据库的自定义转储，以便加载到RDF存储中，用Apache Jena API访问非RDF数据库，D2RQ有自己的mapping语言，R2RML-kit，可自动生成和定制mapping文件。在D2R Server目录下使用generate-mapping命令从关系表中生成.n3文件，文件中主要包括关系库转换成RDF的d2rq:ClassMap和d2rq:PropertyBridge，即转换规则。在D2R Server目录下使用使用dump命令把生成的.n3文件转存为。nt三元组文件（N-triples）。当前已经出现了大量RDB2RDF的工具，如D2R，Virtuoso，Morph，r2rml4net，Ontop，Ultrawrap，GraphDB，Oracle SW等。Ontop 工具最先进的OBDA 系统，兼容RDFs、OWL 2 QL、R2RML、SPARQL标准，支持主流关系数据库： Oracle、MySQL、SQL Server、Postgres。Ultrawrap是唯一能够像SQL一样快速执行SPARQL的系统。Ultrawrap经过精心设计，可以最大限度地利用现有的SQL基础架构，尤其是数据库自带的元数据及其SQL优化器。在基准测试和关系数据库平台上，Ultrawrap的可伸缩性始终与底层关系数据库系统的可伸缩性相匹配。GraphDB是最具扩展性的语义存储库。它包括三重存储，推理引擎和SPARQL查询引擎。GraphDB使用TRREE引擎执行 RDFS， OWL DLP，OWL Horst推理和OWL 2 RL。支持最具表现力的语言是OWL 2 RL，包含 RDFS。GraphDB提供可配置的推理支持和性能。

## 半结构化数据

半结构化数据是一种[结构化数据](https://en.wikipedia.org/wiki/Structured_data)形式，它不遵循与[关系数据库](https://en.wikipedia.org/wiki/Relational_database)或其他形式的[数据表](https://en.wikipedia.org/wiki/Table_(database))相关联的数据模型的形式结构，但仍然包含[标记](https://en.wikipedia.org/wiki/Tag_(metadata))或其他标记来分离语义元素并强制执行记录层次结构和数据中的字段。因此，它也被称为[自描述](https://en.wikipedia.org/wiki/Self-describing)结构。[[3]](https://en.wikipedia.org/wiki/Semi-structured_data#cite_note-1)半结构化数据是介于结构化数据和非结构化数据之间的一种数据类型，但却是不严格的、多变的和不完整的。在半结构化数据中，属于同一类的实体可能具有不同的[属性，](https://en.wikipedia.org/wiki/Attribute_(research))即使它们被组合在一起，并且属性的顺序并不重要。自[互联网](https://en.wikipedia.org/wiki/Internet)出现以来，半结构化数据越来越多，[全文](https://en.wikipedia.org/wiki/Full-text)[文档](https://en.wikipedia.org/wiki/Documents)和[数据库](https://en.wikipedia.org/wiki/Databases)不再是唯一的数据形式，不同的应用程序需要一种[交换信息](https://en.wikipedia.org/wiki/Information_exchange)的媒介。在[面向对象的数据库中](https://en.wikipedia.org/wiki/Object_database)，人们经常会发现半结构化数据。和[普通](https://baike.baidu.com/item/%E6%99%AE%E9%80%9A/9010)纯文本相比，半结构化数据具有一定的结构性，在数据库系统中有着独特的地位：它是一种适于数据库集成的数据模型，也就是说，适于描述包含在两个或多个数据库（这些数据库含有不同模式的相似数据）中的数据。它是一种标记服务的基础模型，用于Web上共享信息。[[13]](https://en.wikipedia.org/wiki/Semi-structured_data#cite_note-1)

[XML](https://en.wikipedia.org/wiki/XML)和[JSON](https://en.wikipedia.org/wiki/JSON)都是半结构化数据的代表，XML具有灵活的结构，用于存储Web contant，是一种数据处理手段，可以被html重用，html在xml内容上加了样式结构，但是XML并不具有语义描述能力。[JSON](https://en.wikipedia.org/wiki/JSON)是一种开放标准格式，它是由属性 - 值对组成的数据对象。它主要用于在服务器和Web应用程序之间传输数据，作为XML的替代方案，[MongoDB](https://en.wikipedia.org/wiki/MongoDB)等数据库就是以JSON格式本地存储数据。

在数据存储的过程中，有些信息都可以作为结构化数据存储在关系数据库的不同表中，比如上文提到的关于詹姆斯·瓦特的人员表和地区表，但是我们可能得到的不是完全的结构化数据，也就是说很难用关系表中的字段表示所有实体的属性信息，不同实体具有的属性是截然不同的，有一些出乎意料的信息。我们可以将得到的信息进行简单的查询和统计，将真正有价值的信息存储在关系数据库中并且建立一些子表存储不同分类的信息。但是这样对于数据的扩展不能很好的适应。另外还可以使用XML对半结构化数据进行存储，具有很好的扩展性，非常适合半结构化数据的存储。

半结构化数据特点

半结构化数据中数据自身就描述了其相应结构模式。具体来说，半结构化数据具有下述特征：

先拥有模式，后拥有数据，模式信息是隐含的。模式不是强制性的约束，而是对结构信息进行描述。因此半结构化的数据在结构上具有自描述性，结构与数据相交融，不需要区分[元数据](https://baike.baidu.com/item/%E5%85%83%E6%95%B0%E6%8D%AE)和一般数据。

半结构化的数据结构描述起来相对于结构化数据来说更为复杂且不规则，含有噪声。结构难以纳入现有的各种描述框架，实际应用中不易进行清晰的理解与把握。数据规模较大，同时支持嵌套或分层数据，支持对列表的操作。

半结构化数据需要动态更新数据和结构的变化。数据的变化通常会导致结构模式发生改变，整体上具有动态的结构模式。这样动态的模式存在着不精确和不完备的问题，不过半结构化数据需具有个性化、多样化、灵活的属性形式，因此能够适应更为广泛的应用需求。

半结构化数据知识获取

半结构化数据具有一定的结构性和自描述性，因此可以按照相对固定的模式对实体信息进行抽取。我们可以直接把半结构化数据转化为RDF等知识库内容存储在图数据库中或者可以先把半结构化数据转化为结构化数据存储在关系库中然后再由结构化数据转化为相关知识库的内容存储在图库里，后者在实际的使用中比较常见，因为这样可以使知识内容更加规整和有条理。常见的百科类网页数据或商品类网页数据，大部分是半结构化的。对于一些特殊的网站，相对于一般的有规律的页面，可以使用简单网络爬虫的方式，利用正则表达式或者selenium等方式来提取网页中的元素，但是这种方法适合于简单和特定的网页，通用性很差。所以对于半结构化的网站，我们可以采用包装器的方式对网站进行处理。包装器是一个能够将数据从HTML网页中抽取出来，并且进行相关处理将它们还原为结构化的数据的软件程序。

首先把要处理的网页输入到生成的包装器中，经过包装器的处理得到所需的结构化信息存储到关系数据库中，对于维基百科或者商品网站这样有着明显结构特征的网页可以采取统一的模式进行抽取。我们可以使用多种方法进行包装器的生成，例如能通用的抽取任何网页中的信息的手工标注的方法，这种方法虽然比较简单但是需要耗费大量的人力或者使用有监督学习的方式训练样例集合中学习数据抽取规则，这种方式可以处理结构比较相似的网页，不具有普适性，同样也需要对数据进行标注。目前比较流行的包装器生成方式是采用无监督学习的方式，以相似的网页作为输入。无需进行人工的标注，批量抽取能力比较强，但是由于是无监督学习，可能在准确性方面还需要进一步提高。

下面就来看一下如何对维基百科中的关于詹姆斯瓦特的信息进行抽取，我们在维基百科中输入詹姆斯瓦特，首先会出现詹姆斯瓦特的标题和摘要，如下图7.4所示

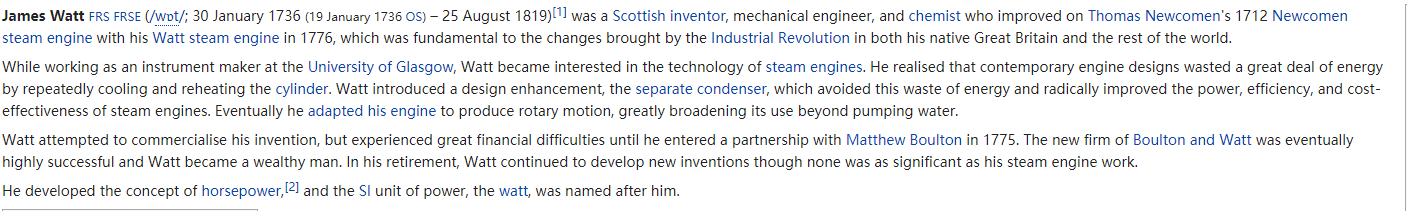


图7.4 维基百科——詹姆斯瓦特简介

同时在搜索页面的最右边会有詹姆斯瓦特的基本信息，如图7.5所示：

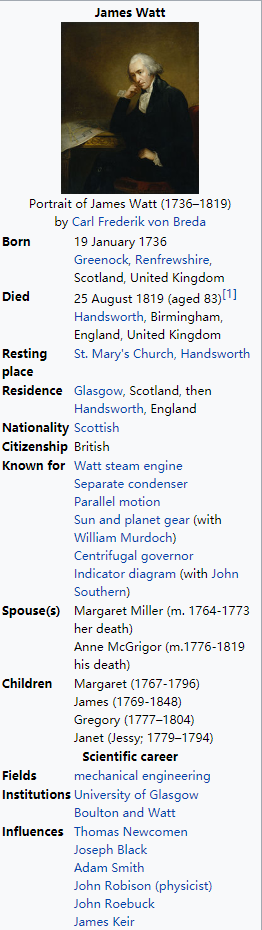


图7.5维基百科——詹姆斯瓦特基本信息

我们可以根据所需要的知识在维基百科中找到对应的位置，然后进行信息的抽取，比如我们想要关于詹姆斯瓦特相关人员的信息，可以在上面的基本信息图中找到influences，点击链接就会跳到相关人员的网页，用同样的方式对信息进行处理。可见对于维基百科中的数据我们可以采用固定的模式对实体信息进行抽取，包括标题，摘要，基本信息和链接等等，最后把抽取的结果存储到关系数据库或图数据库中。

结构化数据和半结构化数据的知识获取相对于非结构化数据的获取还是比较简单的，由于非结构化数据的多变性和复杂性抽取起来比较困难，但是也更加丰富和有趣，接下来就让我们一起来探索一下非结构化数据的知识获取吧。

## 实体抽取

什么是实体和实体抽取？

实体是文本之中承载信息的词语或者短语，也是知识图谱的基本单元。在知识图谱之中，知识是通过<实体、关系、实体>或者<实体、属性、属性值>三元组来进行存储和展示的。一个知识图谱通常是以实体为节点，关系为边的巨大网络。所以实体是知识图谱的核心单元，知识图谱中最基本的元素。

实体是文本语义理解的基础。在“1794年，瓦特与博尔顿合伙组建了专门制造蒸汽机的公司”这句话中，包含着时间实体“1794年”，人名实体“瓦特”和“博尔顿”。这三个实体可以有效地描述这句话中包含的信息。

例7.6实体关系三元组合实体属性三元组

如图7.6所示，在瓦特的知识图谱之中，存在合伙人这一关系，用三元组表示为<瓦特、合伙人、罗巴克>、<瓦特、合伙人、博尔顿>；瓦特的国籍信息表示为<瓦特、国籍、英国>；其中，瓦特、罗巴克、博尔顿是三个实体。

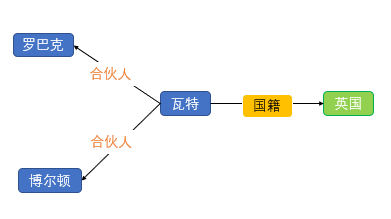


图7.6 瓦特的合伙人与国籍

实体抽取，又称为命名实体识别（Name Entity Recognition，NER），指的是识别文本中的命名性实体，并把这些实体划分到指定的类别。常用的实体包括七类，分别是：人名、地名、机构名、时间、日期、货币、百分比。在不同的领域的知识图谱中，还会有其他类型的实体。例如我们要构建第一次工业革命的知识图谱时，可能就要加入“机器”这一类专有名词实体。本章所讲的方法和例子，主要以抽取通用的实体为主。

在例7.7的这句话中，就包含了人名，地名和时间等实体信息。对这句话进行实体抽取，就是抽取出“1973年1月19日”、“詹姆斯·瓦特”、“苏格兰”、“格拉斯哥”、“克莱德河湾”、“格林诺克”

**例7.7 实体抽取**

1736年1月19日詹姆斯·瓦特出生于苏格兰格拉斯哥附近，克莱德河湾（Firth of Clyde）上的港口小镇格林诺克。

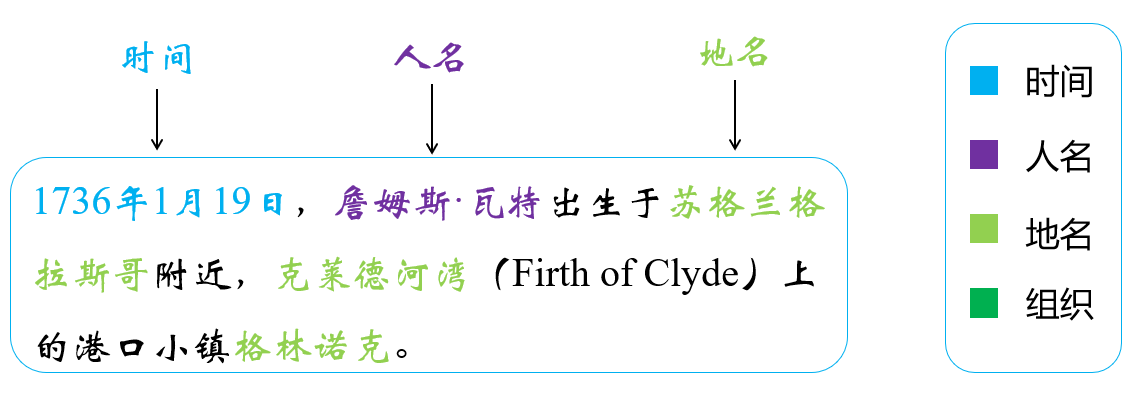


图7.7实体抽取实例

实体抽取的挑战：

实体抽取实际上有两个过程组成：第一是识别出实体和非实体，也就是识别出实体的边界。第二就是分类，把实体归入到特定的类型。就实体的种类来说，时间、日期、货币、百分比这四类实体都有非常明显的特征，而且出现较为规律。所以实体抽取的难点在于人名、地名、组织名这三类实体。此外，由于中英文存在差异，英文文本词语之间有空格边界、实体首字母大写，而中文文本则没有这样的特点，所以中文文本的实体抽取比英文更加复杂。

（1）实体表达不规律，命名实体的内部结构非常复杂，仅仅只关注实体名称是难以捕做到这种规律的，往往需要建立上下文模型。

人名：中文人名一般由姓氏和名组成。一般来说，姓氏是限制在一定数量的汉字之中，存在一个封闭的集合之中。但是名的用字和组成非常灵活。同时，中文对人的指称呼还会使用职务，或者字、号。此外，如果是由外文翻译得到的中文外国人名，用字多是音译，不同的翻译方式还会使用不同的汉字，因此规律又不相同。如“詹姆斯·瓦特”这一人名与中国人名有着不同的规律。

地名：地名的用字十分非常灵活，结构多样；同时地名也有很多简称，别称。例如，“湖北”和“鄂”，“英国”和“不列颠”。

组织名：组织机构的名称往往有各种简称。而且，一些组织机构名称中还会嵌套其他实体，有的是地名，如“武汉大学”中“武汉”就是地名；有的是其他组织机构名称，如“联合国安理会”中“联合国”就是一个组织机构；还有的可能是人名，比如苏格兰知名的学院“詹姆斯·瓦特学院。

即便是时间、日期、货币、百分比这一类实体，也存在着阿拉伯数字和汉字的不同表达方式。

（2）实体的多样性和开放性。同样的一个词语，在不同情景下，会被认为是不同类型的实体。甚至在一种情景下是实体，而在其他情景下就不是实体。同时，实体并非是一个封闭的集合，而是随着时间不断变化。在变化的过程中，可能会增加一些新的实体，也可能会被划为新的类别。

（3）缺乏语料。目前的实体抽取方法往往采用有监督的方法。这种方法需要大量的训练语料。然而标注语料的成本是巨大的，而且对于不同语言、不同文风、不同领域、不同任务是有区别的。所以，绝大数情况下，都不可能获得足够的语料来训练模型。解决这一问题的方法是采用无监督、半监督，以及迁移学习的方法。

实体抽取的方法：

按照实体抽取的领域来进行划分，可以将实体抽取划分为单一领域的实体抽取和开放域的实体抽取。

早期的实体抽取方法主要面向单一领域，关注如何识别文本中的人名地名等专有名词、有意义的时间信息和专业领域的实体。早期的实体抽取方法，主要采用基于规则的方法。随着统计学习方法的发展，隐马尔科夫模型、条件随机场等模型被应用到实体抽取任务之中。近些年来，深度学习方法成为实体抽取的主流模型。与此同时，注意力模型，迁移学习等方法，成为实体抽取任务中新的思路。

开放域的实体抽取不再限定于特定的知识领域，而是面向开放的互联网，研究和解决全网信息抽取问题，不需要（也不可能）为每个领域或每个实体类别建立单独的语料库作为训练集。其主要思想是对于任意给定的实体，采用统计机器学习的方法，从目标数据集 （通常是网页等文本数据）中抽取出与之具有相似上下文特征的实体，从而实现实体的分类和聚类。开放域实体抽取，会在本节最后一小节进行介绍。

7.1.1、7.1.2、7.1.3、7.1.4小节将分别介绍常见的单一领域实体抽取的方法。7.2.5小节将介绍开放域实体抽取的方法。

### 7.4.1基于规则的方法

首先，我们看一些大家熟悉的人名：

张良 诸葛亮 李世民 李白 杜甫 苏轼

詹姆斯·瓦特 艾萨克·牛顿 阿尔伯特·爱因斯坦

上面列出了大家非常熟悉的中外人名，观察这些人名，我们可以发现其实是有一定的规律可循的。这些人名往往是由姓氏和名组成的，不仅仅中文如此，翻译后英文人名之中也有这样的规律。在中文人名中，姓氏的用字是有一定规律的，而名往往是有一定字数限制。那么我们在抽取人名的时候，首先可以找到姓氏的集合，构建一个姓氏的字典。如果文本之中存在这些姓氏，那么与姓氏后的文字（一般是1-2个汉字）结合起来，就是一个人名了。

我们再看一些例子：

张良是战国时期韩国人。

诸葛亮出生于琅琊郡阳都县。

公元726年，李白去往扬州。

这里面都出现了人名和地名，人名与地名之间存在一些动词：“是”，“出生于”，“去往”。并且这些人名和地名，换成其他人名地名，也依然是成立了。这说明这是一个普遍的规律。

这说明，在自然语言中，实体与上下文之间往往存在很多的规律。如果我们能够找到这些规律，那么就可以抽取出这些实体。这就是基于规则的实体抽取方法。下面的例子展示了一些中文实体抽取的规则。

例子：中文命名实体识别规则：

<人名>生于<地名>：詹姆斯·瓦特出生于苏格兰格拉斯哥。

<人名>创建<组织名>：乔布斯和沃兹尼亚克一起创建了苹果公司。

<组织名>位于<地名>：武汉大学位于湖北武汉。

在实体抽取的早期研究之中，基于规则的方法占据主导地位。这种方法多采用语言学专家构造规则模板，利用词性、标点符号、关键字、指示词和方向词、位置词、句法等信息，定义出规则。基于规则的方法往往与字典结合起来，采用字符串匹配的方法来识别命名实体。

总的来说，基于规则的方法，更加接近人类的思维，而且表达直观，便于推理，也不需要标注语料。在特定领域中，基于规则和字典的方法能够取得非常高的准确率和召回率。但是，规则需要语言专家来构建，需要花费大量的人力，成本昂贵。同时，规则和字典的维护任务也非常繁重。如果出现新的词汇，就需要对规则和字典进行更改。而且，在某一个领域取得高准确率的规则，移植到另一个领域时，准确率会大幅降低。

那么有没有不需要依赖人力构建和维护的方法呢？

### 7.4.2传统统计方法

上一节中我们列出了一些用来识别实体的规则。有一部分规则是利用了上下文的信息。例如，<人名>出生于<地名>。这反映了一个规律，“出生于”这个词后面，接地名的概率非常大。显然，在自然语言之中，这样的例子非常多。一个词语是不是实体，可能与其前后的许多词语都有关系。

基于这种上下文的思路，学者们提出了一系列的统计模型。基于统计模型的方法，不需要人工定义规则，也不需要领域知识。这种方法使用人工标注的数据，选择合适的特征和模型对数据进行学习，得到训练模型，然后使用训练模型来预测实体。

传统的统计方法，一般包括三个步骤。

（1）选择特征。传统统计方法需要人工选择特征。这些特征包括词语上下文信息、词缀、词性上下文特征等等。

（2）选择模型并训练模型。用于实体抽取任务的统计模型包括：最大熵模型（maximum entropy model，MaxEnt）、支持向量机（support vector machines，SVM）、隐马尔科夫模型（hidden markov model，HMM）、条件随机场（Conditional random field ，CRF）等。不同的模型有着不同的优缺点，所以选择合适的模型至关重要。

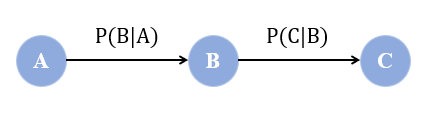
（3）预测实体。这一步是使用已经训练好的模型来预测实体。一般使用统计模型是把实体抽取任务转化为序列标注问题。所以这一步是输入序列，获得序列对应的标注，处理得到实体。

概率图模型

在机器学习中，有许多问题是在处理相互独立的数据，比如说识别图像中的手写数字。但是很显然，语言的上下文之间是存在依赖的，自然语言处理中有很多问题都不在此列。有一系列的模型用来处理这种存在依赖的问题。最有代表性的就是概率图模型。本节后续要介绍的隐马尔科夫模型和条件随机场都是概率图模型。

为什么这些模型叫概率图模型呢？这是因为这些模型在形式上是由图结构组成的。图中的节点表示一个随机变量，图中的边表示随机变量之间的依赖关系，在不同模型中有不同的解释。图可以分为有向图和无向图。而概率图模型也同样具有这种特性，基于此，可以将概率图模型分为有向图和无向图模型，分别对应贝叶斯网络（ Bayesian network）和马尔可夫网络（Markov networks）。这里的有向描述的是后面的节点对前面节点的单向依赖，而无向描述的是相互依赖。

贝叶斯网络，又称为信念网络（Belief Network），是一个有向无环图。如下图所示，节点A指向节点B，节点B指向节点C，而其中的边描述的是条件概率。可以看得出来，在贝叶斯网络之中，随机变量的联合分布概率等于局部条件概率相乘。如P（A，B，C）=P（C|B，A）P（B|A）P（A）。如果后一个节点只依赖于前一个节点，此时是一个马尔科夫链。



马尔科夫网络是无向图，但并不要求无环。其随机变量的联合分布概率一般通过因子分解的方法进行计算。

序列标注

在使用统计模型进行实体抽取的时候，CRF等模型通常把实体抽取任务转化为序列标注问题。序列标注问题指的是对给定的一个序列，标注出其中特定的词语。实体识别，可以看作是一种序列标注的问题。即输入一个句子，可以得到这个句子中每个词是实体还是非实体。

在命名实体识别的任务中，通常使用IO方法、BIO方法和BIOES方法来进行序列标注 。IO方法中的I表示需要标注的词语（实体），而O则表示不属于实体。BIO方法，I表示该字属于一个实体的中间部分，O表示该字不属于一个实体，而B表示该字处于一个实体的开始。BIOES方法和IOB方法类似，其中B也表示是一个实体的第一个字，I表示实体的中间部分，O表示不属于实体，但是增加了E表示实体的最后一个字，S表示单独一字就可以组成一个实体。

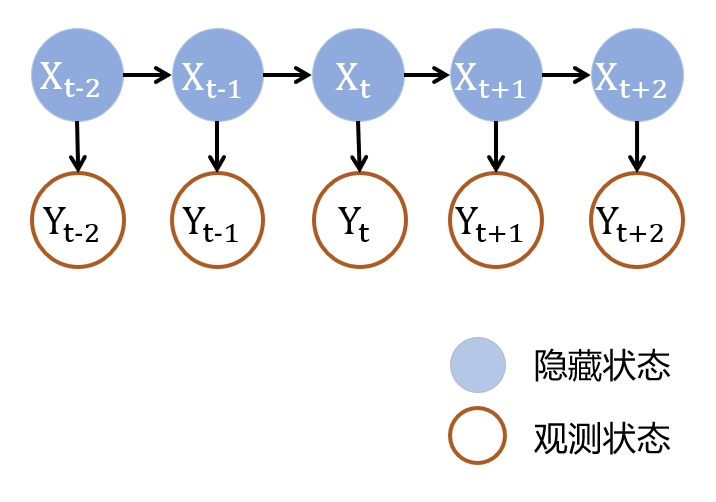
**例子**：对“瓦特出生于苏格兰”这个序列进行标注，PER表示人名，LOC表示地名。分别使用IO、BIO、BIOES方法进行标注，得到的结果如下图所示。

图片包含 屏幕截图

已生成高可信度的说明

隐马尔科夫模型

隐马尔科夫模型是关于时序的概率模型。模型由两个主要的序列，由隐藏的马尔科夫链随机生成的不可观测的隐藏状态随机序列，而每个隐藏状态生成一个可以观测的状态，这些观测状态组成观测序列。隐马尔科夫模型有两个基本的假设：第一是任意时刻的隐藏状态只与前一时刻的隐藏状态有关；第二是任意时刻的观测状态只与当前时刻的隐藏状态有关。



隐马尔科夫模型由初始概率分布、状态转移概率分布、观测状态概率分布确定。接下来，以隐马尔科夫模型进行实体抽取为例来介绍这一模型。

使用隐马尔科夫模型进行实体抽取，是将实体抽取任务转化为序列标注问题。序列对应的标注，即为隐藏状态；而序列本身就是观测状态。在这一模型中，有5个基本的元素。这5个基本元素可以从训练语料中统计得到。

Q：隐藏状态的有限集合。在实体抽取任务中，表现为序列对应的标注。

V：观测状态的有限集合。在实体抽取任务中，表现为序列本身。

π：初始概率矩阵。在实体抽取任务中，表现为隐藏状态中每一个标注的初始概率。

A：状态转移概率矩阵。状态转移矩阵是在时刻t处于状态而在在时刻t+1处转移到另一状态的概率。在实体抽取任务中，表现为此刻的标注转移到下一个标注的概率。

B：观测状态概率矩阵。观测状态矩阵是在时刻t处于隐藏状态生成观测状态的概率。在实体抽取任务中，表现为某个标注下，生成某个词的概率。

通过训练确定隐马尔科夫模型的5个基本元素之后，可以通过近似算法或者维比特算法来进行预测实体。近似算法把时刻t最有可能出现的状态作为你其隐藏状态。显然，局部的最可能状态并不一定是整个序列的隐藏状态是最可能出现的，有些隐藏状态实际可能不发生。而维比特算法采用动态规划的思路，求从初始时刻到t时刻概率最大的路径，得到的对应序列也就是最可能出现的状态。

条件随机场

条件随机场模型是Lafferty[1]于2001年，在最大熵模型和隐马尔科夫模型的基础上，提出的一种判别式概率无向图学习模型，是在给定一组随机变量的条件下，输出另外一组随机变量的条件概率模型。

统计模型的对比

传统的统计模型虽然避免了人工维护和需要领域专家的问题，但是这些模型依赖人工设计的特征和现有的自然语言处理工具（如分词工具）。人工设计的特征和自然语言处理工具直接影响到统计模型的性能。而基于深度学习的方法能够自动捕获文本的特征。

### 7.4.3深度学习方法

基于深度学习的方法是目前实体抽取的主流方法。总的来说包括两类，第一类是NN-CRF架构，也就是使用神经网络结合条件随机场的模型，包括CNN-CRF、RNN-CRF、BiLSTM-CRF。在这一类架构中，CNN/RNN/LSTM 被用来学习每一个词位置处的向量表示，基于该向量表示，NN-CRF解码该位置处的最佳标签。第二类是N-Gram，采用滑动窗口分类的思想，使用神经网络学习句子中的每一个N-Gram 的表示，然后预测该 N-Gram 是否是一个目标实体。这些模型之中，性能最好的是BiLSTM-CRF模型。

BILSTM-CRF

在众多的深度学习模型之中，BiLSTM-CRF是最为常用的模型之一。BiLSTM是双向长短时神经网络，是循环神经网络（RNN）的一个变种。循环神经网络是一种用来处理序列数据的神经网络模型。在一个特定的序列之中，某个特定的输入往往和它的上下文之间有着非常重要的联系，所以在处理序列问题的是，单独地分析某一个词的特征是不够，需要对整个序列进行分析。RNN最关键的地方就是不仅可以学习当前序列的信息，还会依赖前面序列的信息。

但是当序列很长的时候，RNN模型就存在着不足，此时RNN会因为历史特征过多而无法全部表述。为了克服RNN存在的不足，Hochreiter和Schmidhuber 提出了长短期神经网络（Long Short-Term Memory，LSTM ），LSTM在RNN的基础上，增加了三个门来对上一个神经元传递过来的特征进行取舍，舍去部分不重要的特征，使得LSTM能够存储长距离的重要特征。

在实际处理序列标注问题时，由于某一个字的上下文都对该字存在着影响。所以，在进行序列中，为了充分捕捉该字对上下文的语义依赖，往往使用两层双向LSTM（Bi-LSTM）。BiLSTM由前向LSTM和后向LSTM构成，同时考虑了前向特征和后向特征。如图所示，在对“汤姆患有自闭症”这句话进行编码时，前向LSTM依次输入，得到几个向量，后向LSTM依次输入，得到几个向量。将前向和后向的向量进行拼接，得到了最终的向量。

如果仅仅使用BiLSTM来进行实体识别，可能会出现诸如B-PER后紧接着是I-LOC这样的错误。而CRF层可以为最后预测的标签，添加一些约束来保证预测的标签是合乎标注规则的。比如说，保证B的后面跟的是I，每一个词都是由B开头，B和I表示的是同一种实体类型。

使用BiLSTM-CRF来进行实体抽取，模型一般分为5个部分。主要结构如图所示

（1）输入层：输入的内容是句子序列，可以按照每个字符输入，也可以按照词输入。

（2）嵌入层：将序列转化为字向量或词向量编码。

（3）BiLSTM层：前向LSTM和后向LSTM分别对序列进行表示，获取当前序列的上下文信息，构建序列的双向表达，并将之合并。

（4）CRF层：使用CRF学习上下文标签，结合BiLSTM层的输出，预测当前序列概率最大的标签。

（5）输出层：输出每个字符和它的标签。

### 7.1.4新的研究

采用深度学习的方法进行实体识别，取得的效果比传统的统计模型要好。但是深度学习模型也需要不断的优化，才能进一步提高。另一方面，深度学习的方法需要大量标注语料，人工标注的方法需要花费时间的精力。所以，目前对实体抽取的研究，一方面是致力于深度学习方法的优化。另一方面则是解决标注语料过少的问题。

注意力机制

注意力机制（Attention，Att）是一种模拟人类在阅读听说中注意力行为的模型。人类在阅读文字时，往往会把注意力集中在有价值的词语上，而忽略低价值的词语，这样能更快地理解文章的含义。深度学习中的注意力机制从本质上讲和人类的选择性视觉注意力机制类似，核心目标也是从众多信息中选择出对当前任务目标更关键的信息。注意力机制通过计算注意力概率分布，对序列中关键的部分分配更大的权重，突出某一个输入对整个输出的影响，可以很好地优化传统模型。

例子：引入注意力机制的实体抽取

实体抽取的主流神经网络模型都是用的是词向量，但是无法很好的表示一些不在词典中的词。于是又加入了字符向量来处理这个问题。字符向量和词向量的拼接就成为一个关键的问题。使用注意力机制获得词向量与字符向量的权重，从而允许模型动态的决定使用多少信息。

迁移学习

虽然基于深度学习的方法，不需要人工来设计特征，同时能够取得较高的准确率和召回率。但是在实际的实体抽取任务中，由于领域的不同、任务的不同，往往需要重新设计和训练模型。另一方面，无论是统计模型还是深度学习模型都需要大量的标注数据，而现实情况是虽然有大量的数据，却往往是没有标注，标注语料的缺乏为模型的训练带来了极大的困难。同时，深度学习模型依赖于计算资源和大数据，对于普通研究人员来说，这些都是难以拥有的。针对这些问题，专家们提出了迁移学习。

迁移学习是将某个领域或任务上学习到的知识或模式应用到不同但相关的领域或问题中。深度学习需要大量的高质量标注数据，然而在实际研究中，往往只有少量标注数据，所以，基于迁移学习的Pre-training + fine-tuning 是现在进行实体抽取最新的方法。

目前，已经有大量的迁移学习模型被使用到实体抽取任务之中。其中最有代表性的当属BERT模型。

BERT

BERT是一个迁移学习模型，在双向深度网络Transformer 的基础上做多任务学习得到预训练模型，然后在模型的最后添加一层迁移到具体的任务上去。BERT的结构用了Transformer，Transformer 在处理长期依赖方面优于 LSTM，Transformer的双向self-attention可以同时考虑左右两边的上下文信息。

使用BERT进行实体抽取，只需要加Softmax输出层，就可以进行序列标注任务了。

### 7.1.5开放域实体抽取

开放域实体抽取的任务是在给出某语义类的若干实体（称为“种子”）的情况下，从海量文本中找出该语义类包含的其他实体。

开放域实体抽取，初始信息少，种子数量少，同时语义类别难以确定。

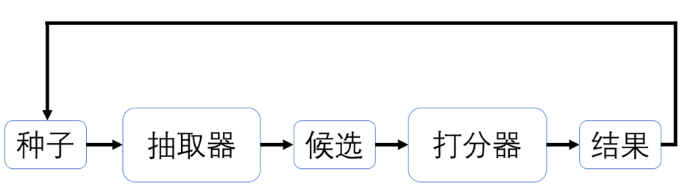
开放式实体抽取的基本假设是“同类实体在网络上具有相似的网页结构或者相似的上下文特征。

流程：

（1）根据种子进行建模；

（2）候选实体的选取；

（3）候选实体的置信度计算和排序。



开放域实体抽取方法：

根据已知实体实例解析特征见面，利用模型处理海量数据得到新的命名实体列表，不断迭代。基于实体特征从搜索引擎服务器日志中识别出命名实体，然后采用聚类算法对实体对象聚类。

一种迭代扩展实体语料库的解决方案，基本思路是根据已知的实体实例进行特征建模，利用该模型对处理海量数据集得到新的命名实体列表，然后针对新实体建模，迭代地生成实体标注语料库。

另一种思路是通过搜索引擎的服务器日志获取 新出现的命名实体，即事先并不给出实体分类，而是基于实体的语义特征从搜索日志中识别出命名实体，然后采用聚类算法对识别出的实体对象进行聚类，该方法已经在搜索引擎技术中得到应用，用于根据用户输入的关键字自动补全信息。

## 关系抽取

什么是关系？

关系指的是两个实体之间的某种联系，这种联系一般是某种语义关系。例如，我们在完成实体抽取之后，从句子“1794年，瓦特与博尔顿合伙组建了专门制造蒸汽机的公司。”之中得到“瓦特”和“博尔顿”两个人名，这两个人是“合伙人”，这就是“瓦特”和“博尔顿”二人之间的关系。

关系抽取的任务就是自动地识别实体之间的这种联系。关系抽取通常是得到一个三元组<实体1，关系，实体2>，例如前面的“瓦特”和“博尔顿”之间的关系可以表示为<瓦特，合伙人，博尔顿>。

关系，可以是预先定义好类别的，这种抽取叫做限定域的关系抽取。而如果没有预先定义关系的类别，那么这种抽取叫做开放域的关系抽取。

关系抽取的挑战

（1）关系表达的多样性、隐含性、复杂性和时空性。

多样性是指自然语言的表达方式多。同一种关系，可以有多重表达方式。而且这些表达方式之中，还有很大一部分并不包含描述这种关系的标识，这是关系的复杂性。同时在实际任务之中，两个实体之间可能同时存在多种关系。关系还具有时空性，两个实体之间的关系会随着时间空间的变化而变化。

（2）依赖于自然语言处理工具。关系抽取依赖句法分析、语法分析这些工具。而在目前，这些自然语言处理工具的性能并不高。

关系抽取的方法

限定域关系抽取的方法可以分为基于规则的方法，基于学习的方法。基于学习的方法，又分为有监督，半监督和远程监督。由于关系类别已经预先定义，所以一般可以人工或者基于启发式地规则自动构建标注语料。所以半监督和远程监督的方法是关系抽取中的主要研究内容。

开放域的关系抽取，由于没有定义关系的类别，所以主要是无监督的方法。

本节的7.5.1小节将介绍基于规则的方法，7.5.2将介绍无监督的方法，7.5.3小节将介绍监督学习方法，7.5.4小节将介绍半监督学习方法，7.5.5小节将介绍远程监督方法，7.5.6小节将介绍开放域关系抽取。

### 7.5.1基于规则的方法

在实体抽取之中，我们已经讨论过基于规则的实体抽取方法。同样的，基于规则的关系抽取方法，也是由语言专家根据语言特征（词汇、句法、语义特征）设计一些手工规则，然后在文本中寻找和这些模式相匹配的语句，从而得到实体之间的语义关系。其中比较常用的语言特征，一是触发词，二是句法分析。

触发词

在两个实体之间，往往具有能表征二者关系的词语。自然的，如果我们能够把这些词语都收集起来，放入到词典之中。在判断实体对的关系时，只需要把匹配实体对之间或者附近是否存在这些词语。如果存在这些词语，就可以判定实体对之间存在这样的关系。

不过，由于自然语言的复杂性，很多实体对虽然存在某种关系，但是实体对之间可能并不存在能够只表达这一种关系的词语。例如，“瓦特和罗巴克一起开始了新式蒸汽机的试制”。“瓦特”和“罗巴克”是合伙人的关系，但是这句话中的词语“一起”并不能单一地表明这种关系。

依存句法分析

句法分析是另一种较为的常见的基于规则的关系抽取方法。

基于规则的方法需要设计规则。然而自然语言表达的多样性，。规则难以维护。在规则集合小的时候，召回率较低。

### 7.5.2无监督关系抽取（开放域）

无监督关系抽取的方法主要是采用聚类的方法。这种方法基于这样一个假设：如果两个实体之间具有相似的语义环境，那么这两个实体具有相同的语义关系。基于这个假设，我们只需要表示出实体对之间的语义环境特征，对这些特征进行聚类。经过对实体抽取的学习，我们非常自然地就可以想到使用上下文特征来表示这种语义环境特征。

所以，无监督关系抽取的一种方法是把同一个实体对的所有上下文都收集起来，把上下文作为语义特征，采用层次聚类的方法把相似度较高的实体对作为同一种关系的实体，最后赋予这种关系一个名称，可以是频率最高的词。

### 7.5.3监督学习

大多数有监督的关系抽取方法是把关系抽取视为一种多分类的问题。

有监督学习的关系抽取方法主要分为两类。一类是把实体抽取和关系抽取当做不同的任务来进行，被称为流水线方法。但是，实体抽取和关系抽取之间有着非常紧密的关系，所以很多专家提出实体抽取和关系抽取的联合模型，这种方法被称为联合抽取。

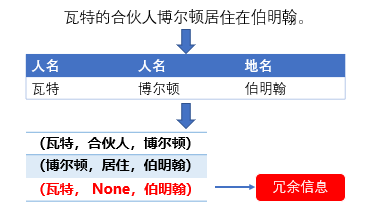
流水线方法

流水线方法包括基于特征向量的方法、核函数的方法和基于神经网络的方法。所谓特征向量。

基于神经网络的方法，主要是使用CNN和BiLSTM对句子进行编码。

Att-BiLSTM

在本节的开始，就提到过流水线方法没有考虑到实体抽取和关系抽取之间紧密的联系。因此，在使用流水线方法进行关系抽取的时候，会存在两个主要的问题。第一是实体抽取的错误会导致关系抽取的错误，错误的传播会使信息抽取的准确率不高。第二是关系抽取中，会对实体抽取得到的实体两两配对来进行关系的判断，这样会产生大量的冗余信息，提高了关系抽取的错误率。



联合抽取

一种联合抽取的思路是通过底层模型的参数共享来实现，主要思想是共享神经网络的底层表达。一般来说，这种方法还是把实体抽取和关系抽取分成两个任务来进行，只是在抽取时共享一些参数。

例子：首先双向的LSTM层来对输入进行编码。然后分别使用一个LSTM来进行命名实体识别（NER）和一个CNN来进行关系抽取。通过底层的模型参数共享，在训练时两个任务都会通过后向传播算法来更新共享参数来实现两个子任务之间的依赖。

另一种联合抽取的思路是，使用一种新的标注策略把原来涉及到序列标注任务和分类任务的关系抽取完全变成了一个序列标注问题。在这种方法里，一个实体的标注会包含这个实体的关系信息。然后通过一个端对端的神经网络模型直接得到关系实体三元组。

例子：提出一种新的标注方法。这种方法融合了序列标注中的BIOES方法和关系标注的方法。在一个句子中，非实体的单词依然是使用“O”进行标注，而实体单词，则在BIOES标注的基础上加入预先定义的关系类别和关系角色的标注。

如果对“瓦特与博尔顿合伙组建了专门制造蒸汽机的公司”这一句话使用这种标注方法，那么结果就是：

### 7.5.4半监督学习

监督学习的方法，无论是流水线模型还是联合抽取，都需要大量的语料。然而实际任务中，往往缺乏大量的语料。那么

Bootstrapping：

语义漂移：

7.2.5远程监督

在进行知识抽取的过程中，大规模的标注语料往往是难以获得的。使用人工标注，需要花费时间和人力。而另一方面，学术界和工业界现在已经构建了大量的知识库，这些知识库里包含着丰富的知识。那么有没有什么关系抽取的方法能够把这些知识利用起来呢？远程监督就是这样一种方法。

远程监督是由Mike Mintz等人在ACL2009上首次提出[2]。这种方法不同于传统意义上的监督学习，而是将非结构化的文本通过与知识库对齐的方式，自动标注数据，减少对人工标注的依赖。这种方法基于一个基本假设：知识库中的一个实体对存在某种关系，那么包含两个实体的非结构化的句子，能够表示出这种关系。这些句子通常是未标注的语料，利用知识库中的信息可以把这些句子就可以转化为标注语料。

使用远程监督，一般包含两个步骤。第一是从知识库中抽取存在关系的实体对。第二就是从非结构化文本中抽取含有实体对的句子作为训练样例。

例子；

图片包含 文字

已生成极高可信度的说明

远程监督可以利用丰富的知识库信息，使用大量的未标注数据，并且不需要大量的标注数据，能够减轻人工标注的代价。然而在远程监督之中，由于假设过于肯定，会引入大量的噪声，同样会出现语义漂移的现象，所以噪声的过滤是远程监督中至关重要的一个问题。而且，与bootstraping不同的是，远程监督的方法难以发现新的关系，这是因为远程监督中关系都是在知识库中已经定义好的。

噪声过滤是远程监督方法目前的研究重点，大致有三类方法。一种方法是多示例学习（Multiple Instance Learning，MIL）。第二种方法是使用注意力机制。在远程监督中，所有包含统一实体对的句子中，可能存在这一个实体对的多种关系。给予这些句子不同的注意力权重，就可以减少噪声句子对关系抽取的影响。第三种方法是使用强化学习的方法。

7.2.5开放域关系抽取

开放域关系抽取是为了处理大量异构数据而设计的，其所抽取的关系类型不 受限制，数量也不定。开放域关系抽取的目的是处理单个句子，将其变成三元组 样式的结构化表示。华盛顿大学在这方面做了大量代表性的工作，如 TextRunner，Kylin，WOE，ReVerb等。

TextRunner能够直接从网页纯文本中抽取实体关系，在这一过程中只考虑文本中词与词之间的关系特征，而不考虑网页内部的结构特征。TextRunner首先利用简单的启发式规则，提取一些浅层句法特征，训练一个分类器，用来判断两个实体间是否存在语义关系；然后在海量网络数据上，找到候选句子，提取浅层句法特征，利用分类器判断所抽取的关系对是否可信；最后利用网络数据的冗余信息，对初步认定可信的关系进行评估。其过程类似于语义角色标注。它把动词作为关系名称，通过动词链接两个实体。

## 概念抽取

## 事件抽取

## 本章习题

问答题：

1、2、3、4、5、6、7、8

应用题：

9、10、11、12、13

设计题：

14、使用BILSTM-CRF进行实体识别。

15、使用Att-BiLSTM进行关系抽取。

16、事件抽取

# 知识融合

## 8.1 概述

在获取到知识之后，需要把知识存储起来。但是由于获取到的知识可能不是由同一种渠道或者方法得到的，因此需要对这些不同方式获得的知识进行处理。知识融合，顾名思义就是把不同来源的知识融合在一起构成不同知识间的关联，这些知识可能具有不同的概念和结构或者包含大量的冗余信息，因此核心问题是把同一个实体或概念的信息融合起来，消除来自不同知识之间的冗余概念和歧义，使知识能够形成一个整体进行一致表达，保证获取知识具有很高的质量。

知识融合过程中要注意融合的实体、属性和种类，把各个层面的知识融合在一起，主要包括概念层和数据层。但是想要完美的融合不同来源和层面的知识并不是那么容易的，会由许多的挑战，首先数据可能量级特别大，计算起来比较困难，数据的种类也很繁杂，同时数据的质量也难以保证，数据的结构和格式可能完全不同，命名规则也存在差异。这些挑战都需要使用相对应的技术来解决，最终得到一个良好的知识库。知识融合主要可以分为两个部分的内容，首先是进行实体的链接，然后是知识的合并，下面将分别介绍一下这两个部分的内容。

## 8.2 实体链接

实体链接技术是知识融合的主要工作，实体链接也可以称为实体对齐、实体匹配或实体解析，是指对于从多样化的信息中比如一段文本抽取得到人们感兴趣的实体对象和实体的上下文，然后将其链接到知识库中对应的正确实体对象的操作，这些知识库可能是结构化的知识库也有可能是半结构化的知识库，若是知识库中没有想要的实体会返回一个空的标记。实体链接中的文本包括同一来源和不同来源的信息，我们需要从这些文本中抽取到想要的多个实体。实体链接的主要作用是消除不同来源、不同结构的数据中实体冲突、指向不明等不一致性问题，这样就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。实体链接的基本思想是首先根据从文本中得到给定的实体指称项，从知识库中选出一组候选实体对象，然后通过相似度函数或相似性算法计算进行实体消歧和共指消解，判断知识库中的同名实体与之是否代表不同的含义以及知识库中是否存在其他命名实体与之表示相同的含义，将指称项链接到正确的实体对象[2]。

随着技术的发展和数据规模的增长，实体对齐也会受到越来越多的挑战，包括数据质量、计算复杂度等问题。所以要进行实体链接首先需要对数据进行清洗和标注，也可以将待对齐数据进行分区索引，以降低计算的复杂度。原始数据的质量会影响到最后链接的结果，所以我们需要对原始数据做一下清洗工作，那么数据清洗应该如何进行呢？首先要保证语法的正确性，不存在语法上的错误，同时数据格式要保持一致，移除或者添加相应的符号，然后可以对数据进行分块处理，这样可以提高计算的速度，最终得到一份质量较好的数据进行下一步处理。在处理好输入数据之后，接下来就是提取输入数据中的实体然后和知识库中的内容进行相似度的计算。我们可以先计算出实体的属性相似度，然后根据属性相似度得到最后的实体相似度。

在得到属性相似度之前，首先要了解一些基本概念，编辑距离是针对二个[字符串](https://baike.baidu.com/item/%E5%AD%97%E7%AC%A6%E4%B8%B2)（例如英文字）的差异程度的量化量测，量测方式是看至少需要多少次的变化才能将一个字符串变成另一个字符串。在计算编辑距离时，1965年[Vladimir Levenshtein](https://en.wikipedia.org/wiki/Vladimir_Levenshtein) 发现了最小编辑距离Levenshtein distance，[14] Levenshtein distance可以用在计算属性相似度的过程中，用最少的编辑操作将一个字符串转换成为另一个，Levenshtein distance允许的编辑操作包括单个字符的替换/插入/删除，编辑思想与动态编程“矩阵”无关。而菲舍尔-瓦格纳距离Wagner–Fischer distance是Levenshtein distance的一个扩展，利用动态规划的思想把编辑过程中不同操作的代价赋予了不同的权重。[Wagner–Fischer distance algorithm](https://en.wikipedia.org/wiki/Wagner%E2%80%93Fischer_algorithm)是一种动态编程算法，用于计算两个字符串之间的编辑距离。

另外一种计算属性相似度的方式是计算属性集合的相似性，下面介绍三种方法来计算集合的相似度，分别是Dice系数、Jaccard 系数、Ochiai coefficient。Dice系数是以[Lee Raymond Dice](https://zh.wikipedia.org/w/index.php?title=Lee_Raymond_Dice&action=edit&redlink=1)命名的，计算公式为 ，其中x和y是字符串的长度，可以把字符串理解为一种集合，它的范围在0到1之间。而Jaccard 系数主要是用来处理短文本的相似度，与Dice系数公式类似，Jaccard 系数的计算公式为，x和y分别代表符合某种条件的集合，公式的含义为：两个集合交集的大小/两个集合并集的大小，交集等于并集意味着2个集合完全重合。而两者之间的差异是Dice系数相应的差异函数不是一个合适的距离度量措施，因为它没有三角形不等性的性质。Ochiai coefficient通过计算两个集合的夹角的[余弦](https://zh.wikipedia.org/wiki/%E4%BD%99%E5%BC%A6)值来度量它们之间的相似性。计算公式为。x，y分别代表符合某种条件的集合，公式的含义为：两个集合的交集大小/两个集合大小的几何平均值。Ochiai coefficient是余弦相似度的一种形式，0度角的余弦值是1，而其他任何角度的余弦值都不大于1，并且其最小值是-1。两个向量有相同的指向时，余弦相似度的值为1；两个向量夹角为90°时，余弦相似度的值为0，两个向量指向完全相反的方向时，余弦相似度的值为-1。相似性范围从-1到1：两个向量之间的角度的余弦值确定两个向量是否大致指向相同的方向。-1意味着两个属性向量指向的方向正好截然相反，1表示它们的指向是完全相同的，0通常表示它们之间是独立的，而在这之间的值则表示中间的相似性或相异性。这结果是与向量的长度无关的，仅仅与向量的指向方向相关。余弦相似度通常用于正空间，因此给出的值为0到1之间。这上下界对任何维度的[向量空间](https://zh.wikipedia.org/wiki/%E5%90%91%E9%87%8F%E7%A9%BA%E9%96%93)中都适用，而且余弦相似性最常用于高维正空间。[3]另外一种计算属性相似度的方式是基于向量的相似度 ：TF-IDF值是用来评估某个字或者某个词对文档的重要程度计算公式为 。比如现在有10万篇文章，关于人物传记方面的有1万，现在有一篇文章共有2000词，詹姆斯瓦特共出现20次，那么最后的TF-IDF值为。

在得到单个属性相似度之后可以由单个属性相似度得到训练集m个实体属性相似度向量：，然后利用属性相似度向量得到实体相似度。首先可以采用聚合的方式也就是基于传统概率模型的计算方法，主要就是考虑两个实体各自属性的相似性，而并不考虑实体间的关系。建立了计算实体相似度的概率模型，采用直接聚合属性相似度的方式，这种方法的1缺点是没有体现重要属性对于实体相似度的影响。所以可以对属性相似度进行加权平均，具体公式为或者手动制定规则为每个匹配的属性对分配了不同的权重，这样的匹配准确度会有所提高。

另外可以使用标注数据利用监督学习的方法将基于属性相似度评分来判断实体是否匹配的问题转化为一个分类问题。通过属性比较向量来判断实体对匹配与否可称为成对实体对齐。这类方法中的典型代表有逻辑回归LR、决策树DT、支持向量机SVM、集成学习、朴素贝叶斯模型等。具体来介绍一下，逻辑回归Logistic Regression是一种用于解决二分类问题的机器学习方法，用于估计某种事物的可能性。逻辑回归是一种广义的线性回归分析模型是，目的是使假设函数找到合适的参数值使得函数值和样本值接近，模型清晰而且便于理解数据。逻辑回归的假设函数形式如下：， 其中x是我们的输入，θ为我们要求取的参数。逻辑回归模型所做的假设是， 使用最大似然估计方法对模型中的参数进行估计。代价函数使用交叉熵，。逻辑回归对于特征之间存在关联的数据不太好处理。

决策树也称作分类树或回归树，在这些树的结构里，[叶子](https://zh.wikipedia.org/w/index.php?title=Leaf_node&action=edit&redlink=1)节点给出类标而内部节点代表某个属性。在进行实体链接的过程中可以使用分类回归树，一棵树的训练过程为根据一个特征指标，分裂训练集为几个子集。这个过程不断的在产生的子集里重复递归进行，即[递归分割](https://zh.wikipedia.org/w/index.php?title=%E9%80%92%E5%BD%92%E5%88%86%E5%89%B2&action=edit&redlink=1)。当一个训练子集的类标都相同时[递归](https://zh.wikipedia.org/wiki/%E9%80%92%E5%BD%92)停止。这种决策树的自顶向下归纳是[贪心算法](https://zh.wikipedia.org/wiki/%E8%B4%AA%E5%BF%83%E7%AE%97%E6%B3%95)的一种， 也是目前为止最为常用的一种训练方法。[3]决策树算法有CART树、ID3算法、C4。5算法等，使用决策树作为分类模型比较直观、易于理解和解释同时只需很少的数据准备、算法简单容易实现，而且训练和预测的准确率都比较高应用广泛。但同时使用决策树模型缺少足够的理论支持，需要选择合适的树结构，决策树创建如果过于复杂会导致过拟合，可以采用剪枝的方式减少这种问题，剪枝分为前剪枝和后剪枝，前剪枝就是在树的构建过程（只用到训练集），设置一个阈值（样本个数小于预定阈值或GINI指数小于预定阈值），使得当在当前分裂节点中分裂前和分裂后的误差超过这个阈值则分列，否则不进行分裂操作，前剪枝使得树更加简洁。后剪枝是在用训练集构建好一颗决策树后，利用测试集进行的操作。效果相对来说更好一些。

支持向量机support vector machine是找到一个使数据尽可能分开且分类间隔尽可能大的超平面，属于结构风险最小化。在介绍支持向量机之前，首先了解两个概念：函数间隔和几何间隔。在超平面确定的情况下，通过观察的符号与类标记y的符号是否一致可判断分类是否正确，所以可以用的正负性来表示分类的正确性。那么函数间隔的表达式为。只定义函数间隔并不足以反映样本点到超平面的距离，因为如果成比例的增减w或b的值，函数间隔会相应变化，但是超平面是没有改变的，所以我们需要对w加一些约束，这就是几何间隔，用公式表示为。几何间隔真正表示了样本点到超平面的距离。最大间隔分类器是支持向量机的前身，它的主要目的就是使几何间隔最大化，从而确定超平面的参数值。求解超平面的过程是找到凸优化最优解的过程，通过对原始问题或者对偶问题的求解使得几何间隔最大化得到最优参数进而得到分割平面。支持向量机具有很好的鲁棒性，同时有很好的泛化能力。硬间隔支持向量机用于处理线性可分的情况，而软间隔支持向量机引入了松弛变量用于处理线性不可分的情况，对于非线性可分的情况需要使用核函数。但是需要对输入数据进行完全标记、解出的模型的参数很难理解。SVM的求解可以使用[内点法](https://baike.baidu.com/item/%E5%86%85%E7%82%B9%E6%B3%95/115627)和[序列最小优化算法](https://baike.baidu.com/item/%E5%BA%8F%E5%88%97%E6%9C%80%E5%B0%8F%E4%BC%98%E5%8C%96%E7%AE%97%E6%B3%95/22742660)等二次凸优化问题的数值方法，在拥有充足学习样本时也可使用随机梯度下降的方法。

对于集成学习是使用多种学习器进行学习，并且把学习的结果整合到一起，从而获得更好的学习效果。由多个同质的弱学习器得到一个强学习器。对于大的数据集可以划分成多个小数据集，学习多个模型进行组合。对于小的数据集可以利用Bootstrap方法进行抽样，得到多个数据集，分别训练多个模型再进行组合。集成学习的基础算法主要分为Boosting和Bagging算法，Boosting算法有很多，主要原理是带权重的随机抽样，各个学习器之间存在个体依赖，目的是优化偏差bias。最具代表性的boosting算法是AdaBoost算法，它是其他Boosting算法的基础，主要过程是首先给出任意一个弱学习算法和训练集，输入是某个域或实例空间，输出是在分类问题中是一个带类别标志的合。初始化时， Adaboost 为训练集指定分布为，即每个训练例的权重都相同为。接着，调用弱学习算法进行 T次迭代，每次迭代后， 按照训练结果更新训练集上的分布，对于训练失败的训练例赋予较大的权重，使得下一次迭代更加关注这些训练例，从而得到一个预测函数序列，对每个预测函数https://gss3.bdstatic.com/-Po3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D16/sign=3a265dd1bd3533faf1b69728aad31ba2/377adab44aed2e73a82a52f38b01a18b86d6fa2a.jpg也赋予一个权重， 预测效果好的，相应的权重越大。T次迭代之后，在分类问题中最终的预测函数采用带权重的投票法产生。单个弱学习器的学习准确率不高，经过运用Boosting算法之后，最终结果准确率将得到提高。[4]其它的Boosting算法是在Adaboost算法上发展起来的。另外一种集成学习的方式是Bagging算法，Bagging是booststrap和aggregation的结合，bootstrap也称为自助法，它是一种有放回的抽样方法，目的为了得到统计量的分布以及置信区间。从已知的数据集中模拟其他类似样本，重复进行采样得到不同的模型，然后所有模型表决出最后的输出，主要目的是减少方差。随机森林算法是Bagging算法和决策树算法的结合，是以决策树为基分类器的一个集成学习模型，它包含多个由Bagging集成学习技术训练得到的决策树， 当输入待分类的样本时， 最终的分类结果由单个决策树的输出结果投票决定。随机森林有三种方式，第一种是Bagging算法和完全的分类与回归树的结合，第二种是Bagging和随机抽取子特征分类与回归树的结合，第三种是Bagging和子特征线性组合的结合。随机森林是一种优秀的鲁棒的通用模型，解决了决策树性能瓶颈的问题，对噪声和异常值有较好的容忍性，对高维数据分类问题具有良好的可扩展性和并行性，但是对于高维稀疏数据不是特别好的选择。随机森林是由数据驱动的一种非参数分类方法，只需通过对给定样本的学习训练分类规则，并不需要先验知识[5]。

朴素贝叶斯是假设特征之间[独立](https://zh.wikipedia.org/wiki/%E7%B5%B1%E8%A8%88%E7%8D%A8%E7%AB%8B%E6%80%A7)分布的情况下运用[贝叶斯定理](https://zh.wikipedia.org/wiki/%E8%B4%9D%E5%8F%B6%E6%96%AF%E5%AE%9A%E7%90%86)为基础的[概率分类器](https://zh.wikipedia.org/w/index.php?title=%E6%A6%82%E7%8E%87%E5%88%86%E7%B1%BB%E6%B3%95&action=edit&redlink=1)，是一种易于解释的生成模型，简单易懂，学习效率高，需要较少的训练数据。贝叶斯公式为，只需计算出后面等式的三个部分，X事件的先验概率P(X)、Y属于某类的先验概率P(Y)，以及在已知Y的某个分类下，事件X的后验概率P(X|Y)就可以算出Y属于某类的后验概率，选出最大的后验概率就可以确定属于哪类。由于该算法以自变量X之间的独立（条件特征独立）性和连续变量的正态性假设为前提，所以， P(X)和P(Y)都比较好算出。但是由于需要假设输入变量特征独立，所以就会导致算法精度在某种程度上受影响。

上面介绍了多种分类的方式，我们需要根据特定场景和要求选择最适合的方式或者把几种方式进行融合，最终得到一个好的分类结果。与此同时这种基于传统概率和有监督学习的方式可能会存在一些问题，比如训练集的处理过程，需要对数据进行清洗同时对标签进行标注，可能花费大量的人力和时间。另外在分类过程中很容易出现分类不平衡或者分类错误的问题，所以需要对各种分类器和属性权重占比进行尝试。为了解决这些问题，我们可以采用无监督或者半监督学习的方式进行实体链接。无监督学习的实体相似度计算主要采用聚类的方式进行，聚类有两种情况，一种是层次聚类，另外一种是相关性聚类。聚类的主要思想是将相似的实体尽量聚集到一起，再进行实体对齐

层次聚类是通过计算不同类别数据点之间的相似度对在不同的层次的数据进行划分，最终创建一个有层次的嵌套的树。算法的主要流程是：首先初始化把每个样本都视为一个聚类，然后计算各个聚类样本之间的距离即相似度。每次将距离最近的两个聚类合并到同一个类，再重复计算各个聚类之间的相似度，不停的合并，直到合成了一个类，或者可以设置所需分类的类别个数，作为迭代的终止条件。以树状图的形式展示出来就是层次聚类树。其中类与类之间的距离的计算方法有：SL最短距离法，CL最长距离法，AL中间距离法，类平均法等。最短距离法又称最邻近算法是将类与类的距离定义为两个类数据点中最近的两个数据点之间的相似度。而最长距离法则是把类与类的距离定义为两个类数据点中最远的两个数据点之间的相似度。中间距离法是用两个类中所有点之间相似度的均值作为两个类之间的相似度。在算距离的时候，可以根据不同情况选择不同的距离度量方式，常见的距离有欧式距离、曼哈顿距离、汉明距离、杰卡德距离等等。层次聚类可以直接得到聚类树，再分类时不用重新计算，还可以发现类别之间的层次关系，规则容易理解和定义。但是该算法也存在缺陷，计算的复杂度较高不适合处理数据量特别大的情况。

另外一种聚类算法是K-means聚类，K-means算法首先随机初始化k个聚类中心点。然后计算每个数据点到中心点的距离，数据点距离哪个中心点最近就划分到哪一类中。对于每一个类，重新计算该类的中心点作为新的中心点。重复以上步骤，直到每一类中心在每次迭代后变化不大为止。也可以多次随机初始化中心点，然后选择运行结果最好的一个。 K-means算法原理比较简单、可解释度比较好，同时运行速度快，计算简便，时间复杂度接近线性而且聚类中心点可变，聚类效果也比较好。但是对于K值的选取不好确定，同时聚类得到的结果只是局部最优。如果各隐含类别的数据不平衡，比如各隐含类别的数据量严重失衡，或者各隐含类别的方差不同，则聚类效果不佳。而且K-means算法对噪音和异常点比较的敏感。可以采用手肘法或者轮廓系数法等方法来确定K值，或者利用K-Means++算法、Canopy算法等算法对K-Means随机初始化质心和计算点之间的距离进行优化。这样会使聚类过程和效果更好，实际价值比较大。还可以使用相关性聚类算法，使用最小代价找到一个聚类方案。表示x，y分配在同一类中，代表x，y是同一类的概率（x，y之间的相似度）。同时和分别代表切断x，y之间的边的代价和保留边的代价。我们最后的聚类目标时使聚类代价最小化即，这是一个NP-Hard问题，可以用贪婪算法近似求解。

还可以通过知识表示学习的方式计算实体相似度，这是一种知识嵌入的方式把知识图谱中的实体和关系映射到低维空间向量，直接用数学表达式来计算各个实体间的实体相似度。不依赖任何文本信息，获取的都是数据的深度特征。构建的各种各样不同的知识图谱，这些知识图谱既有不同，也会存在一定共性。知识嵌入可以使用TransR模型，对每个三元组，关系表示成从头实体到尾实体的向量，关系可以看做从头实体到尾实体的翻译。对于实体与向量之间的关系，我们可以举一个例子，比如数据集1中存在头实体詹姆斯瓦特，数据2中存在实体蒸汽机之父，詹姆斯瓦特的出生地是苏格兰格拉斯哥，蒸汽机之父的出生地也是苏格兰格拉斯哥，因此可以考虑詹姆斯瓦特和蒸汽机之父是等价实体。

我们可以利用知识表示学习的方式把这些不同来源的知识图谱给融合成一个更大的知识图谱。有两种方式进行知识图谱的嵌入，一种是双向监督训练，两个知识图谱可以分别学习两个空间，然后用已知的两个知识图谱里面对齐的实体，就可以把这两个空间真正关联起来。我可以知道这个空间里面的一个位置，跟这个空间里面的另一个位置，它们之间有关联，相当于分别学习两个Knowledge Graph的表示空间，然后用非常有限的种子的实体，把这两个空间给融合在一起。[6]另外一种是联合知识嵌入，把两个知识图谱的三元组糅合在一起共同训练，把想要链接的实体对视为具有相同关系的三元组，对两个知识图谱的空间进行约束。[7] 在链接时，找到两个知识图谱中距离最近的实体向量进行链接。另外还可以采用主动学习的方式中，可通过与人员的不断交互来解决很难获得足够的训练数据问题。

接下来介绍两种常用的实体链接方式，实体消歧和共指消解。在了解实体消歧之前首先要明白一个概念知识知名度。实体知名度是表示一个实体被人们知道了解的程度，具有高知名度的实体更有可能在文章中被提起。比如水果苹果、手机苹果和电影苹果的实体知名度排序是水果苹果>手机苹果>电影苹果。实体消歧可以简单理解为对同一名称表达的不同意思或对应于多个命名实体对象的实体明确其具体指代的对象和含义。专门用于解决同名实体产生歧义问题的技术，通过实体消歧，就可以根据当前的语境，准确建立实体链接，实体消歧其实也可以看做基于上下文的分类问题，类似于词性消歧和词义消歧。实体上下文是指特定实体的上下文规律性比如实体苹果周围出现iPad，酷，视网膜屏的苹果更可能是苹果公司，周围出现好吃、甜、一斤的苹果更可能是水果苹果。实体消歧主要采用的算法是聚类法。聚类法是指以实体对象为聚类中心，将所有指向同一目标实体对象的指称项聚集到以该对象为中心的类别下。聚类法消歧的关键问题是如何定义实体对象与指称项之间的相似度。可以使用以下几种模型：空间向量模型、语义模型、百科知识模型、社会网络模型。实体消歧技术能够帮助搜索引擎更好地理解用户的搜索意图，从而给出更好的上下文推荐结果，提高搜索服务质量，其中还有一个很重要的问题是如何对存在歧义的实体进行重要性评估，以确定推荐内容的优先级，思路是为实体赋予权重，用于表示该实体出现的频率或先验概率。 [2]

共指消解主要用于解决多个指称对应同一实体对象的问题，确 定 其 在 真 实 世 界 中 所 指 向 的 实 体。实体主要有三种情况，普通名词、专有名词和代词。共指消解又被称为对象对齐、实体匹配和实体同义。主要有两种存在共指的情况，一种发生在RDF数据，RDF数据中存在共指的原因是RDF将数据以<资源，属性，属性值>的形式去表示，并不保证某一实体或者资源的唯一性，所以在RDF数据中会存在实体共指的现象。另外一种是普通文本中存在实体共指的现象:比如在某文本中存在这样一句话：“詹姆斯瓦特James von Breda Watt在1736年1月19日生于[苏格兰](https://zh.wikipedia.org/wiki/%E8%8B%8F%E6%A0%BC%E5%85%B0)[格拉斯哥](https://zh.wikipedia.org/wiki/%E6%A0%BC%E6%8B%89%E6%96%AF%E5%93%A5)附近[克莱德湾](https://zh.wikipedia.org/wiki/%E5%85%8B%E8%8E%B1%E5%BE%B7%E6%B9%BE)上的港口小镇[格林诺克](https://zh.wikipedia.org/wiki/%E5%8D%B0%E5%BC%97%E5%85%8B%E8%8E%B1%E5%BE%B7)。瓦特的父亲是熟练的造船工人并拥有自己的船只与造船作坊，还是小镇的官员。他的母亲Agnes Muirhead出身于一个贵族家庭并受过良好的教育。他们都属于[基督教长老会](https://zh.wikipedia.org/wiki/%E9%95%B7%E8%80%81%E6%95%99%E6%9C%83)并且是坚定的[誓约派](https://zh.wikipedia.org/w/index.php?title=%E8%AA%93%E7%BA%A6%E6%B4%BE&action=edit&redlink=1)。尽管瓦特出自于宗教家庭，但他后来还是成为了[自然神论者](https://zh.wikipedia.org/wiki/%E8%87%AA%E7%84%B6%E7%A5%9E%E8%AE%BA)。……诞生了蒸汽机之父的格拉斯哥大学……”，那么在该文本中存在的实体共指现象为：詹姆斯瓦特={James von Breda Watt，蒸汽机之父，瓦特，他}。

共指消解不仅可以用传统的统计学方式来解决，随着机器学习的快速发展，共指消解问题常常使用机器学习的算法来解决，包括有监督学习和无监督学习。有监督学习处理共指消解问题主要有四种模型：表述对模型是最常见的共指消解框架，实体-表述模型能够整合已经形成的实体，表述排序模型可以独立判断不同表述对的共指结果，实体排序模型是采用排序学习算法。这四种模型中效果最好的是实体排序模型，其次是表述排序模型，然后是实体-表述模型，效果最差的是表述对模型。共指消解技术除了将共指消解问题视为有监督分类问题之外，还可以将其作为无监督问题来求解。无监督学习包括特征向量抽取，约束聚类算法，非参数贝叶斯模型，基于图的方法，马尔可夫逻辑网络框架。其中应用比较广泛的是约束聚类法，主要方式是以实体指称项为中心，通过利用相似度的计算实现实体聚类从而使指称项与实体对象能够匹配。

## 8.3 知识合并

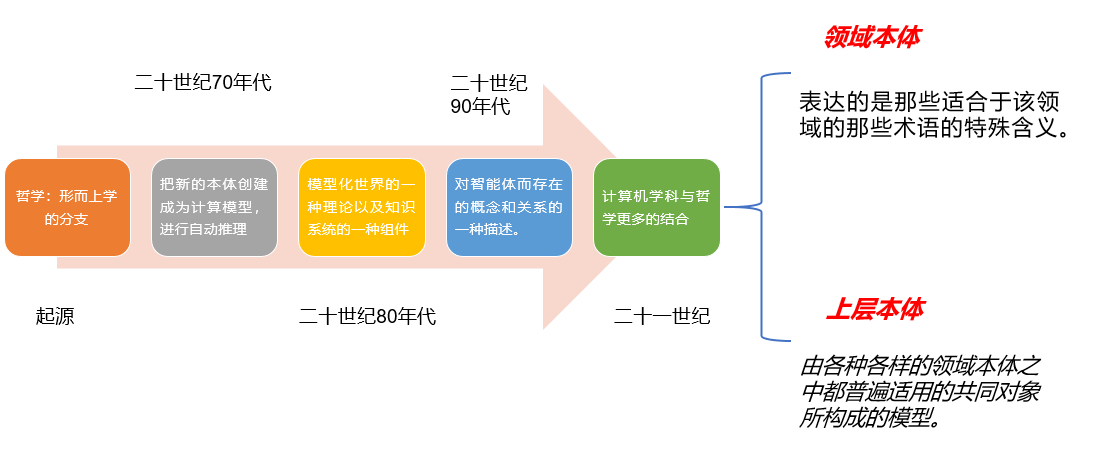
接下来介绍知识融合另外一种方式——知识合并。知识合并是指把外部知识库或者关系数据库合并到目前本地的知识库。在合并外部知识库的过程中主要包括两个层面：数据层的融合和模式层的融合。数据层的融合包括实体的指称、属性、关系以及所属类别等，主要的问题是如何避免实例以及关系的冲突问题，造成不必要的冗余。而模式层的融合是将新得到的本体融入已有的本体库中。然后是合并关系数据库，正如前文所提到的是把关系数据库转换为RDF或其他形式的知识库内容。结构化数据的知识获取可以依据RDB2RDF working group制定的标准，包括Direct Mapping和R2RML。知识合并的流程可以描述为：获取知识、概念匹配、实体匹配和知识评估。

## 8.4 本章习题

# 知识加工

## 9.1本体构建

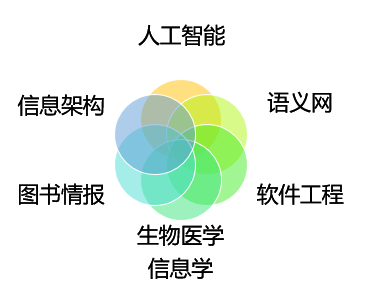
本体源于哲学的概念，在不同的领域有着不同的解释和应用。从哲学领域发展而来的本体在计算机科学和信息科学领域得到语义层面的不同解读，是指一种“形式化的，对于共享概念体系的明确而又详细的说明”。 [8] 本体特定领域之中那些存在着的对象类型或概念及其属性和相互关系，实际上就是对特定领域之中某套概念及其相互之间关系的形式化表达。[9]本体是用于描述一个领域的术语集合，其组织结构是层次结构化的，可以作为一个知识库的骨架和基础。语义网中的本体通常是用RDFS或OWL描述的本体。本体的发展可以用下面图来表示：



本体起源于[哲学](https://zh.wikipedia.org/wiki/%E5%93%B2%E5%AD%A6)“[形而上学](https://zh.wikipedia.org/wiki/%E5%BD%A2%E8%80%8C%E4%B8%8A%E5%AD%A6)”的分支，应用广泛，本体论探索活动的传统目标尤其是，为了揭示那些基本的类别或者说种类，而在关键之处将现实世界划分为对象的自然类别。[10]到了二十世纪70年代，知识获取逐渐成为人工智能领域的重要环节，可以把新的本体创建成为[计算模型](https://zh.wikipedia.org/wiki/%E8%AE%A1%E7%AE%97%E6%A8%A1%E5%9E%8B)获取新的知识。二十世纪80到90年代，本体逐渐成为AI领域关于模型化世界的一种理论以及知识系统的一种组件。在二十一世纪本体的发展更需要多学科包括计算机和哲学等的结合。

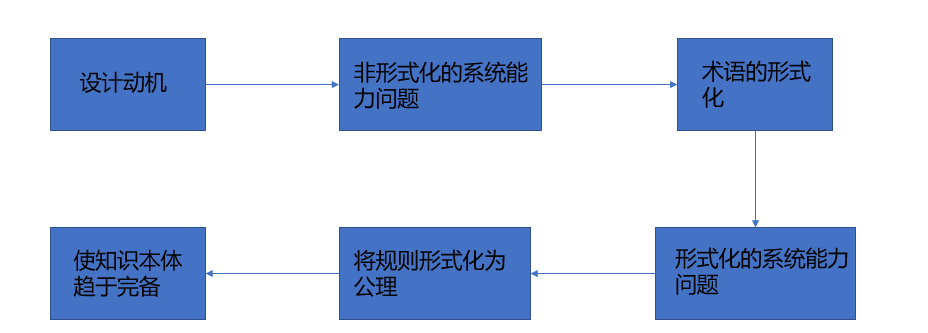
本体按应用主题可分为领域本体、通用本体、知识本体、语言学本体和任务本体。领域本体（domain ontolog）所建模的是某个特定领域，或者现实世界的一部分。领域本体所表达的是那些适合于该领域的那些术语的特殊含义。[通用本体](https://zh.wikipedia.org/w/index.php?title=%E4%B8%8A%E5%B1%82%E6%9C%AC%E4%BD%93&action=edit&redlink=1)（upper ontology）又称常识本体或上层本体是指一种由那些在各种各样的领域本体之中都普遍适用的共同对象所构成的模型。[3] 本体按详细程度和领域依赖度可分为较高级本体(top-level Ontologies)、领域本体(domain Ontologies)、任务本体(task Ontologies)和应用本体(application Ontologies)等。本体主要由类或概念、关系、函数、公理、实例五大部分构成，可以应用于人工智能、信息架构、系统建模、信息处理、图书情报、信息检索和语义Web、生物信息学、软件复用和面向对象技术等领域。

如下图所示：

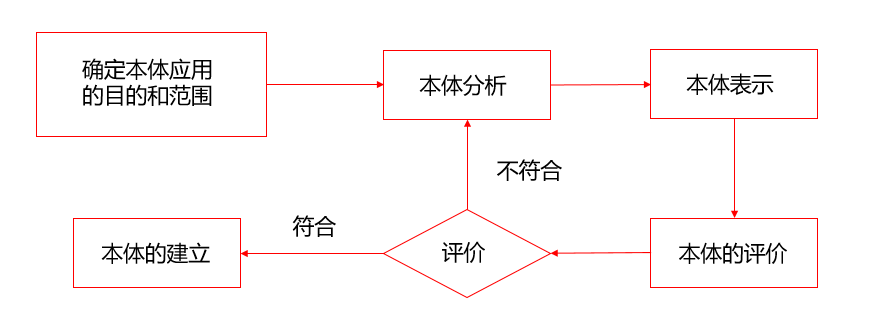


本书主要对领域本体的概念、构建和应用进行详细介绍。领域本体是描述指定领域知识的一种专门本体，它给出了领域实体概念及相互关系领域活动以及该领域所具有的特性和规律的一种形式化描述。主要有三种构建方法包括手工构建、复用已有本体（半自动构建）以及自动构建本体。这三种构建方法的主要区别就是人工参与的程度，目前来说由于领域问题的专业性更多的还是采用半自动构建的方法。通过领域本体可以形式化领域上的专业术语和关系，形成一定程度的知识共享和复用，常见的领域本体有金融本体、生物本体、医学本体等等。在构建本体时要按照一定的标准。相对于一般的系统，本体更强调共享、重用，可以为不同系统提供一种统一的语言，因此本体构建的工程性更为明显，保证明确性、客观性、完全性、一致性和可扩展性。接下来介绍几种手工构建本体工程的方法。

首先是TOVE企业建模法，TOVE法是通过本体建立指定知识的逻辑模型。用一阶逻辑构造了形式化的集成模型，包含企业设计本体、项目本体、调度本体或服务本体。构建的主要流程如下图所示：



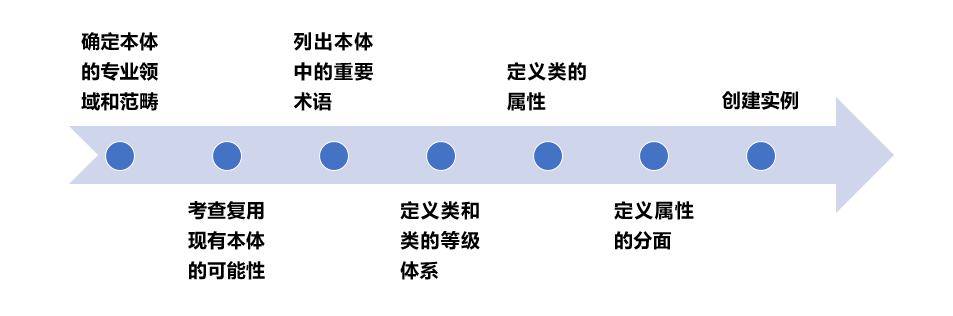
骨架法（Skeletal Methodology），又称Enterprise法，专门用来创建企业本体(有关企业建模过程的本体)。骨架法流程如下图所示：



Methontology方法是在结合了骨架法和GOMEZ-PEREZ方法后，提出的一种更为通用的本体建设方法。这个本体开发方法更接近软件工程开发方法。它将本体开发进程和本体生命周期两个方面区别开来，并使用不同的技术予以支持。专用于创建化学本体(有关化学元素周期表的本体)。它的流程包括：管理、开发和维护。管理阶段主要是系统的整体规划，开发阶段包括规范说明、概念化、形式化、执行和维护五个部分，而维护阶段由规范说明、概念化、形式化、执行以及维护五个步骤构成。

五步法（IDEF5）用于描述和获取企业本体的方法。通过使用图表语言和细化说明语言，获取关于客观存在的概念、属性和关系，并将它们形式化成本体。五步法创建本体的5个主要步骤是：定义课题、组织队伍；收集数据；分析数据；本体初步开发；本体优化与验证。

七步法由斯坦福大学医学院开发用于领域本体的构建。流程如下：



最后一个是循环获取法，它是是一种环状的结构。主要流程包括资源选取、概念学习、领域集中、关系学习和评价。资源选取是环形的起点，任何大型的通用本体都可以作为这个过程的开始。选定基础本体后，用户必须确定用于抽取领域相关实体的文本。概念学习是从选择的文本中获取领域相关的概念，建立概念之间的分类关系。领域集中是除去领域无关的概念，只留下和领域相关的。关系学习是指除了基础本体中继承的关系，其它关系通过学习从文本中抽取。最后对得到的领域相关的本体进行评价，还可以进一步地重复上述过程。

本体构建过程中会使用到本体构建工具，包括本体服务器和本体编辑器，如下图所示：



## 9.2知识推理

知识推理是知识图谱中很重要的一部分，是知识图谱构建的重要手段和关键环节，主要用于推理暗含的知识，丰富现有知识图谱，检查知识库的不一致，挖掘出隐含信息，自动补全逻辑关系。推理对象可以是实体、实体的属性、实体间的关系、本体库中概念的层次结构等。简单来说，知识推理就是利用已知的知识推出新知识的过程。比如已知张三的父亲是李四，李四的父亲是王五，就可以得到张三的爷爷是王五。所以推理就是通过各种方法获取新的知识或者结论，这些知识和结论满足语义。王永庆[11]认为:推理是人们对各种事物进行分析、综合和决策，从已知的事实出发，通过运用已掌握的知识，找出其中蕴含的事实，或归纳出新的事实的过程。 严格地说，就是按照某种策略由已知判断推出新的判断的思维过程。

推理方法按新判断推出的途径划分可分为演绎推理和归纳推理，演绎推理是从一般性的前提出发，通过推导，得到具体描述或个别结论（三段论），结论已经蕴含一般性知识中，只是通过演绎推理揭示出来，不能得到新知识。归纳推理是从一类事物的大量特殊事例出发，去推出该类事物的一般性结论（数学归纳法），推出的结论没有包含在已有内容中，增加了新知识。另外推理方法还可以分为符号推理和数值推理，符号推理是在知识图谱的实体和关系符号上直接推理，利用传统逻辑规则（基于一阶谓词逻辑、描述逻辑等）或抽象层面的本体约束，推理出新的实体关系，还可对知识图谱进行逻辑的冲突检测，特点是覆盖率小，依赖现有数据，不易迁移。数值推理是基于知识表示学习，利用关系机器学习方法，通过统计规律从知识图谱中学习到新的实体关系。特点是缓解数据稀疏性问题，使符号数据可直接参与运算且计算速度非常快。

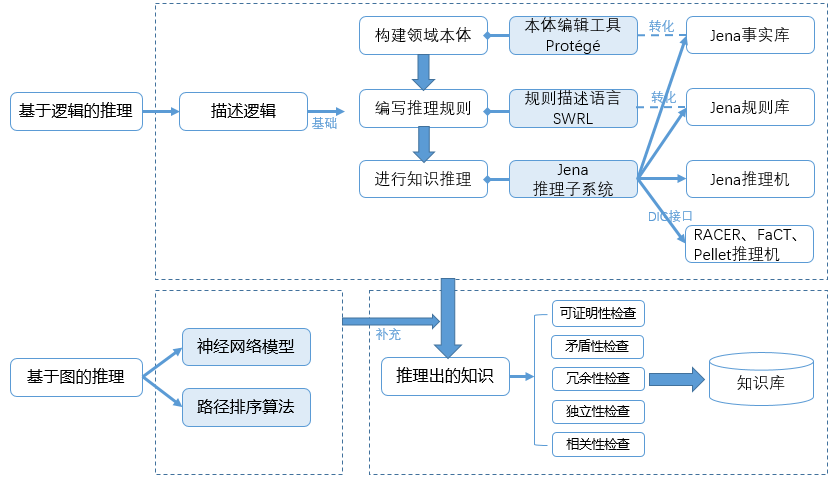
接下来分别介绍一下知识图谱上的符号推理和数值推理。符号推理主要包括一阶谓词逻辑、描述逻辑以及基于规则的推理。一阶谓词逻辑建立在命题的基础上，在一阶谓词逻辑中，命题被分解为个体、关系和概念三部分，个体是指可独立存在的客体，是一个领域内的实例，例如詹姆斯瓦特，概念指的是一个抽象的概念，解释为一个领域的子集，例如发明家。关系是用来表现个体的性质以及不同事物之间关系的词，例如三元组(詹姆斯瓦特，职业，发明家)中的职业就是表达个体詹姆斯瓦特和发明家的关系。对于人际关系可以采用一阶谓词逻辑进行推理，方法是将关系视为谓词，将人物视为变元，采用逻辑运算符号表达人际关系，然后设定关系推理的逻辑和约束条件，就可以实现简单关系的逻辑推理[2]。

对于复杂的实体关系，可以采用描述逻辑进行推理。描述逻辑是一种基于对象的知识表示的形式化工具，也叫概念表示语言或术语逻辑，是一阶谓词逻辑的一个可判定子集，它是本体语言推理的重要设计基础。基于描述逻辑的系统主要包括四个组成部分，最基本的元素（个体、概念和关系）、TBox术语集(terminology box描述概念和关系的公理集合)、ABox断言集(assertion box描述个体的公理集合)以及TBox和ABox上的推理机制。TBox引入了概念和关系的名称，由于概念之间存在包含关系，TBox知识形成类似格的结构，这种数学结构是由包含关系决定的，例如父亲这个概念被人这个概念包含。ABox分为概念断言和关系断言，概念断言表示一个对象是否属于某个概念，关系断言表示两个对象是否满足特定关系。借助这TBox和ABox两个工具，可以将基于描述逻辑的推理最终归结为ABox的一致性检验问题，从而简化并最终实现关系推理。

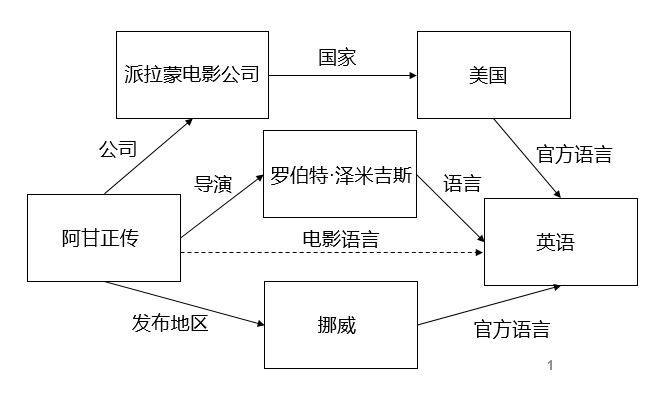
一般的推理往往需要一些规则的支持，把这些信息有效地与推理算法结合在一起是关键。OWL本体语言是知识图谱语言中最规范、最严谨、表达能力最强的语言，具有很强的知识描述能力，允许进行逻辑推理。可以利用抽取加工后的概念、属性、关系以及规则，借鉴七步法和骨架法等本体构建方法，在Protégé等本体编辑平台构建本体。然后，利用专门的规则描述语言SWRL编写推理规则，其目的是让规则可与本体描述语言OWL相结合，从而提高本体的推理能力。最后，使用Jena推理子系统实现反诈骗语义推理，从而获得本体数据中隐含的信息。

Jena自身提供的推理机是针对本体、基于规则的推理机，以OWL2 RL这一专用于高效推理的本体语言为基础，在扩展RDFs表达能力的同时，保持了较低的复杂度Ptime-Complete，可以执行从实例到类的推理；同时，可根据后期项目推理的实际需要，通过DIG接口方便地将RACER、FaCT、Pellet等更“专业”的推理机引入到Jena中。本质是一种CLIPS配合本体领域产生式规则的推理系统。

Jena推理机有两个内部规则引擎：RETE引擎和一个tabled Datalog引擎，支持前向链、后向链以及混合执行模型。Jena结合本体子系统与推理子系统进行推理，本体子系统处理基于RDF的本体模型数据，支持OWL，DAML和RDFS等语言。Jena推理引擎主要由事实库、规则库和推理机 3部分组成：先根据本体，创建相应的实例，Jena推理机将本体转化为Jena事实库，再将SWRL规则转化为Jena规则库，推理机结合Jena事实库与Jena规则库，实现本体语义推理，揭示潜在的知识。支持本体/词林的上下位和等价推理、不一致性推理等。



数值推理主要思想是存在一定数量的三元组关系数据作为训练集和测试集，训练集和测试集的头/尾实体、关系都来自于固定的实体集合和关系集合（也有基于开放域的），通过训练集训练一个评分函数的参数，再通过该函数给测试三元组关系做一个评估打分，然后获得一个评分排名。可以分为很多种，基于路径的建模、分布式表示学习、基于神经网络、混合推理等，基于神经网络模型的推理利用神经网络强大的学习能力建模知识图谱事实元组,获得很好的推理能力和泛化能力，将知识库中的实体表达为词向量的形式，进而采用神经张量网络模型（neural tensor networks）进行关系推理，这一方法的推理准确性已被实证高于85%。基于路径排序算法的推理，根据两个实体间连通路径作为特征来判断两个实体是否属于某个关系。首先进行特征抽取，生成并选择路径特征集合，可以使用的方法有随机游走，广度优先搜索，深度优先搜索。然后进行特征计算，计算每个训练样例的特征值，主要方法有随机游走概率，布尔值（出现/不出现），出现频次/概率等。最后进行分类器训练，根据训练样例，为每个目标关系训练一个分类器，可以使用单任务学习，为每个关系单独训练二分类器和多任务学习，不同关系联合学习。



## 9.3质量评估

质量评估也是知识库构建技术的重要组成部分。在知识图谱的构建过程中质量评估是必不可少的，在知识获取和知识融合的过程中得到的知识都可能存在错误，经过知识推理得到的知识的质量同样也是没有保障的，因此在将其加入知识库之前，需要有一个质量评估的过程。引入质量评估的意义在于:可以对知识的可信度进行量化，通过舍弃置信度较低的知识，可以保障知识库的质量[2]。质量评估按类型可分为三种，数据冗余度评估、数据可信度评估和数据动态性评估。

知识图谱由于数据来源的多源性，经过知识融合之后还是难免会存在冗余数据，数据的冗余意味着必须浪费空间去存储冗余信息。根据RDF三元组的特征分为结构冗余，语义冗余，空节点冗余三个方面。结构冗余是指三元组呈现的形式不同会导致RDF数据集的大小也不同，但是表达的意思是一样的。语义冗余主要依赖于RDF数据的规则推理特性。一个数据集通过规则推理使与另一数据集有共同语义，说明存在语义冗余。空节点在RDF语义中只是作为一个标签来表示某些存在性的事物，因此当某些具有空节点三元组的RDF数据集内有另外一些不具备空节点的相同的三元组时，就会出现空节点冗余。数据可信度评估包括客观可信性、主观可信性和主客观结合可信性。客观可信性指数据内容本身的可信性，可以从数据的排名、来源、第三方机构验证等方面去获取。主观可信度指数据用户主观的认为数据是可信的或者不可信的，例如用户对数据使用的反馈、他人对数据的推荐等。数据动态性评估是指对于有很强的时间性知识图谱，随着时间数据也不一样，因此评估动态环境下数据波动性分析很有必要，某一节点改变产生了什么样的影响。根据数据集和属性，数据集中的属性的作用主要用于构建不同数据之间的语义关系，因而可以通过相关的分析得到数据集中元素的可达矩阵，在可达矩阵中，非０元素代表列元素对行元素存在 依赖关系，即行元素的变化会对列元素产生波及影响。因而可以轻易看出数据集中任一个元素对其他各元素的影响。**[12]**

## 9.4知识更新

知识图谱在构建完成后随着时间的变化，许多内容可能都要进行调整，知识图谱的内容需要紧跟时间的脚步，不断迭代更新。知识图谱的更新方式包括全面更新和增量更新。所谓全面更新是指以更新后的全部数据为输入，从零开始构建知识图谱。这种方式比较简单，但资源消耗大，而且需要耗费大量人力资源进行系统维护;而增量更新，则是以当前新增数据为输入，向现有知识图谱中添加新增知识，这种方式资源消耗小，但目前仍需要大量人工干预(定义规则等)，因此实施起来十分困难。[2]

增量更新主要包括概念层的更新和数据层的更新。顾名思义，概念层的更新是添加新的概念到知识图谱中或从对原有概念进行调整，而数据层的更新是更新知识图谱中的三元组内容，在更新的过程中要注意知识融合的问题，保证知识图谱的正确性。具体来看知识更新可以把当前流行的知识库中所需要的概念和数据更新到知识图谱中，同时也可以采用众包的模式，众包从传统意义上来说是指群众外包，利用大量的网络用户来对内容进行更新，具有价格低廉、灵活等优势，传统众包任务和要求单一，人物评价方法简单，门槛较低。而知识图谱上的众包需要采用知识型众包，是沟通机器与人的桥梁。而对于概念层的更新，则需要借助专业团队进行人工审核。OpenBase是OpenKG实现的开放知识图谱众包平台，主要目标是支撑以中文为核心的开放领域知识图谱的协同构建。促进中文知识图谱数据的开放与互联，促进知识图谱和语义技术的普及和广泛应用。

## 9.5本章习题

**参考文献**

1. 郑煜. 结构化数据异构同步技术的研究[D].北京林业大学,2013.
2. 知识图谱构建技术综述
3. 维基百科
4. 董乐红， 耿国华， 高原。 Boosting算法综述[J]。 计算机应用与软件， 2006， 23(8):27-29。
5. 王奕森， 夏树涛。 集成学习之随机森林算法综述[J]。 信息通信技术， 2018(1)。
6. 知识表示学习及应用
7. Hao Zhu et al。 Iterative Entity Alignment via Knowledge Embeddings，IJCAI 2017
8. Tom Gruber (1993)。 ["A translation approach to portable ontology specifications"](http://tomgruber.org/writing/ontolingua-kaj-1993.pdf)。 In: Knowledge Acquisition。 5: 199-199。
9. Fredrik Arvidsson and Annika Flycht-Eriksson。 [Ontologies I](http://www.ida.liu.se/~janma/SemWeb/Slides/ontologies1.pdf)。 Retrieved 26 Nov 2008。
10. Perakath C。 Benjamin et al。 (1994)。 [IDEF5 Method Report](http://www.idef.com/pdf/Idef5.pdf)。 Knowledge Based Systems， Inc。
11. Wang YQ。 Principles and Methods of Artificial Intelligence。 Xi’an: Xi’an Jiaotong University Press， 1998 (in Chinese)。
12. 顾进广,朱婷婷,黄莉,田萍芳. 知识图谱中链接数据质量评价研究综述[J]. 武汉大学学报(理学版),2017,63(01):22-38.
13. JEFFREYD.ULLMAN，JENNIFERWIDOM编著．数据库系统基础教程 ．北京：由机械工业出版社，2009：289-295
14. Navarro, Gonzalo. A guided tour to approximate string matching (PDF). ACM Computing Surveys. 1 March 2001, 33 (1): 31–88 [19 March 2015]. doi:10.1145/375360.375365.
15. 庄严,李国良,冯建华.知识库实体对齐技术综述[J].计算机研究与发展,2016,01: 165-192.

[1] LAFFERTY J, MCCALLUM A, PEREIRA F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data[J]. Proc ICML, 2002:{}.

[2] MINTZ M, BILLS S, SNOW R, et al. Distant supervision for relation extraction without labeled data: Joint Conference of the Meeting of the Acl & the International Joint Conference on Natural Language Processing of the Afnlp: Volume[C], 2009.