

第4章 知识与推理

南京信息工程大学
计算机学院

应龙
2025年秋季

内容概要

1. 基于知识的智能体

2. 知识表示和推理概述

3. 语义网络与知识图谱

4. 知识图谱推理

Bibliography:

- [1] 王万良 编著, “人工智能导论” (第5版), 高等教育出版社, 2020. Ch 2, Ch3
- [2] 吴飞 编著, “人工智能导论: 模型与算法”, 高等教育出版社, 2020. Ch 2.1-2.3
- [3] Stuart J. Russel, Peter Norvig, “Artificial Intelligence: A Modern Approach” (4th Ed., 2020), Pearson; 中译版 “人工智能 现代方法” (4rd Ed., 2022), 人民邮电出版社. Ch 7, Ch 10
- [4] 王万森 编著, “人工智能原理及其应用” (第4版), 电子工业出版社, 2018. Ch 2

基于知识的智能体

人类一直强调人的智能是如何获得的：不仅仅是靠反射机制，还通过对知识的内部表示进行操作的推理过程。

In AI, **knowledge-based agents** use a process of **reasoning (推理)** over an internal **representation of knowledge** to decide what actions to take.

- Problem-solving Agents 具有知识，但这种知识非常有限的而且死板。它们知道有哪些行动可用，以及从一个特定的状态执行一个特定的行动会有什么结果。但它们并不了解一般的事实。
- Problem-solving Agents 所使用的原子表示也很有局限性。在部分可观察环境中，Agent 表示它对当前状态的认识的唯一选择是列出所有可能的具体状态，在大型环境中这毫无希望。

用变量的赋值表示状态的思想，使得 Agent 的部分工作可以独立于领域，并允许更有效的算法。将逻辑作为支持基于知识的智能体的一类通用表示。这样的 Agent 通过对信息的组合和再组合以适应各种用途。

基于知识的 Agent 能够以显式描述目标的形式接受新任务；通过被告知或者主动学习环境的新知识从而快速获得能力；通过更新相关知识，它们可以适应环境的变化。

基于知识的智能体

基于知识的 Agent 的核心部件是知识库 (KB)。知识库是一个语句集合。这些语句用知识表示语言表达，表示关于世界的某些断言 (assertion)。如果某条语句是直接给定，而不是推导得到的，称其为公理 (axiom)。

TELL (告诉) 和 ASK (询问)：将新语句添加到知识库以及查询目前所知内容。

```
function KB-AGENT(percept) returns an action
  persistent: KB, a knowledge base
               t, a counter, initially 0, indicating time

  TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))
  action  $\leftarrow$  ASK(KB, MAKE-ACTION-QUERY(t))
  TELL(KB, MAKE-ACTION-SENTENCE(action, t))
  t  $\leftarrow$  t + 1
  return action
```

基于知识的 Agent 不是计算行动的任意程序。Agent 要服从知识层的描述，在知识层只需描述 Agent 知道的内容和它的目标，以便修正它的行为。

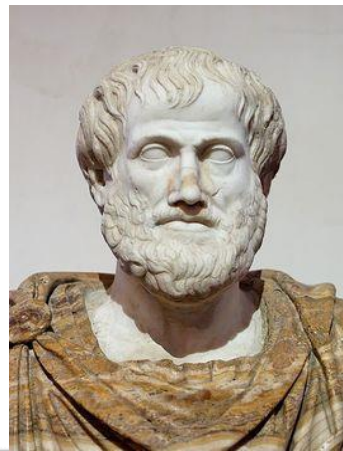
通过 TELL 告知 Agent 必须的知识便可以构建一个基于知识的 Agent。知识库刚开始是空的，Agent 设计者通过 TELL 告知一条条语句，直到 Agent 知道如何在环境中工作。这被称为构建系统的陈述性 (declarative) 方法。相反，过程性 (procedural) 方法把需要的行为直接编码 (encode) 为程序代码。

基于知识的智能体

● 逻辑与推理是人工智能的核心问题之一

- 人类思维活动一个重要功能是逻辑推理，即通过演绎和归纳等手段对现有观测现象进行分析，得出判断。在人工智能发展初期，脱胎于逻辑推理的符号主义人工智能 (symbolic AI) 是人工智能研究的一种主流学派。
- 在符号主义人工智能中，所有概念均可通过人类可理解的“符号”及符号之间的关系来表示。例如：如果使用符号A来表示对象概念、IsCar() 来表示某个对象是否为“汽车”，那么 IsCar(A) 表示“A是一辆轿车”这样的概念。注意 IsCar(A) 由对象A和 IsCar() 两部分所构成。如果A是轿车，则 IsCar(A) 为正确描述、否则为错误描述。
- 符号主义人工智能方法基于如下假设：可通过逻辑方法来对符号及其关系进行计算，实现逻辑推理，辨析符号所描述内容是否正确。

逻辑是探索、阐述和确立有效推理原则的学科，提出了演绎推理中“三段论”方法的古希腊学者亚里士多德被誉为“逻辑学之父”。一般而言，逻辑是用数学方法来研究关于推理和证明等问题的研究。



亚里士多德 (Aristotle
公元前384-前322，古
代先哲、古希腊人)

内容概要

1. 基于知识的智能体

2. 知识表示和推理概述

3. 语义网络与知识图谱

4. 知识图谱推理

Bibliography:

- [1] 王万良 编著, “人工智能导论” (第5版), 高等教育出版社, 2020. Ch 2, Ch3
- [3] Stuart J. Russel, Peter Norvig, “Artificial Intelligence: A Modern Approach” (4th Ed., 2020), Pearson; 中译版 “人工智能 现代方法” (4rd Ed., 2022), 人民邮电出版社. Ch 7, Ch 10
- [4] 王万森 编著, “人工智能原理及其应用” (第4版), 电子工业出版社, 2018. Ch 2

知识表示概述

● 知识的概念

知识的一般性解释

知识是人们在长期的生活及社会实践中、在科学研究及实验中积累起来的对客观世界的认识与经验。

知识的信息加工观点

知识是对信息进行智能性加工所形成的对客观世界规律性的认识（把有关信息关联在一起所形成的信息结构）。

即：知识 = 信息 + 关联

常用的关联形式： 如果 ... , 则 ...

有代表性的观点

- 费根鲍姆（Feigenbaum）：知识是经过剪裁、塑造、解释、选择和转换了的信息
- 伯恩斯坦（Bernstein）：知识由特定领域的描述、关系和过程组成
- 海叶斯-罗斯（Heyes-Roth）：知识=事实+信念+启发式

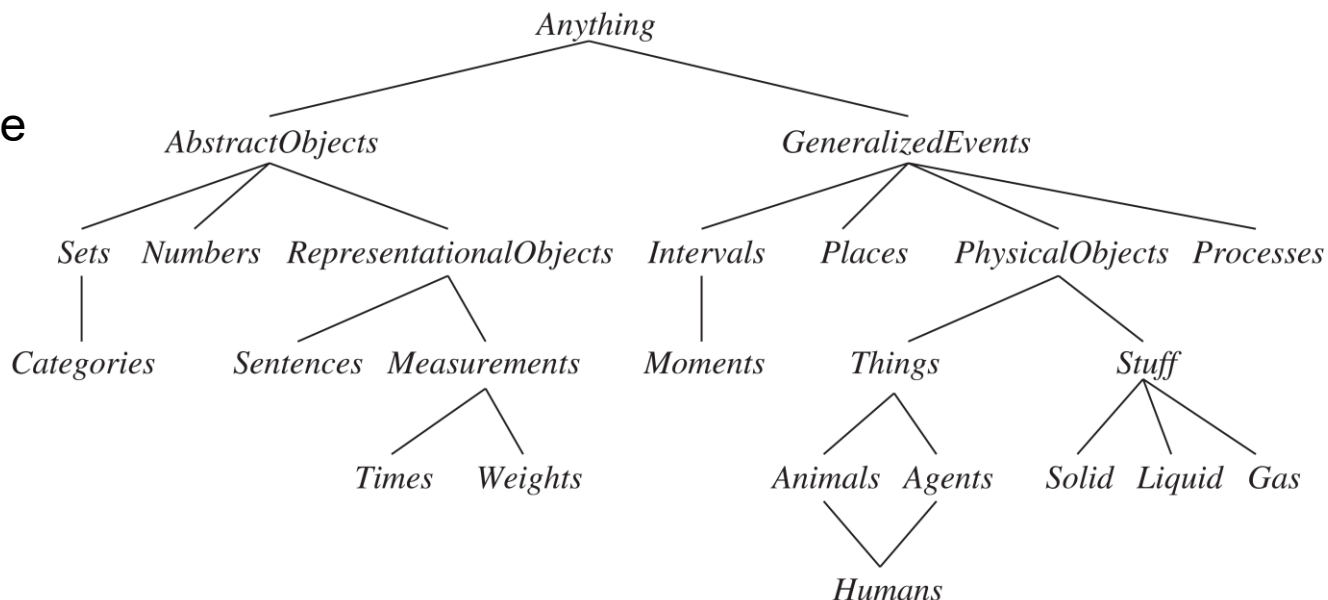
Philosophy

Reality Knowledge 知识 Universality, Generalizability {普遍性, 概括性}

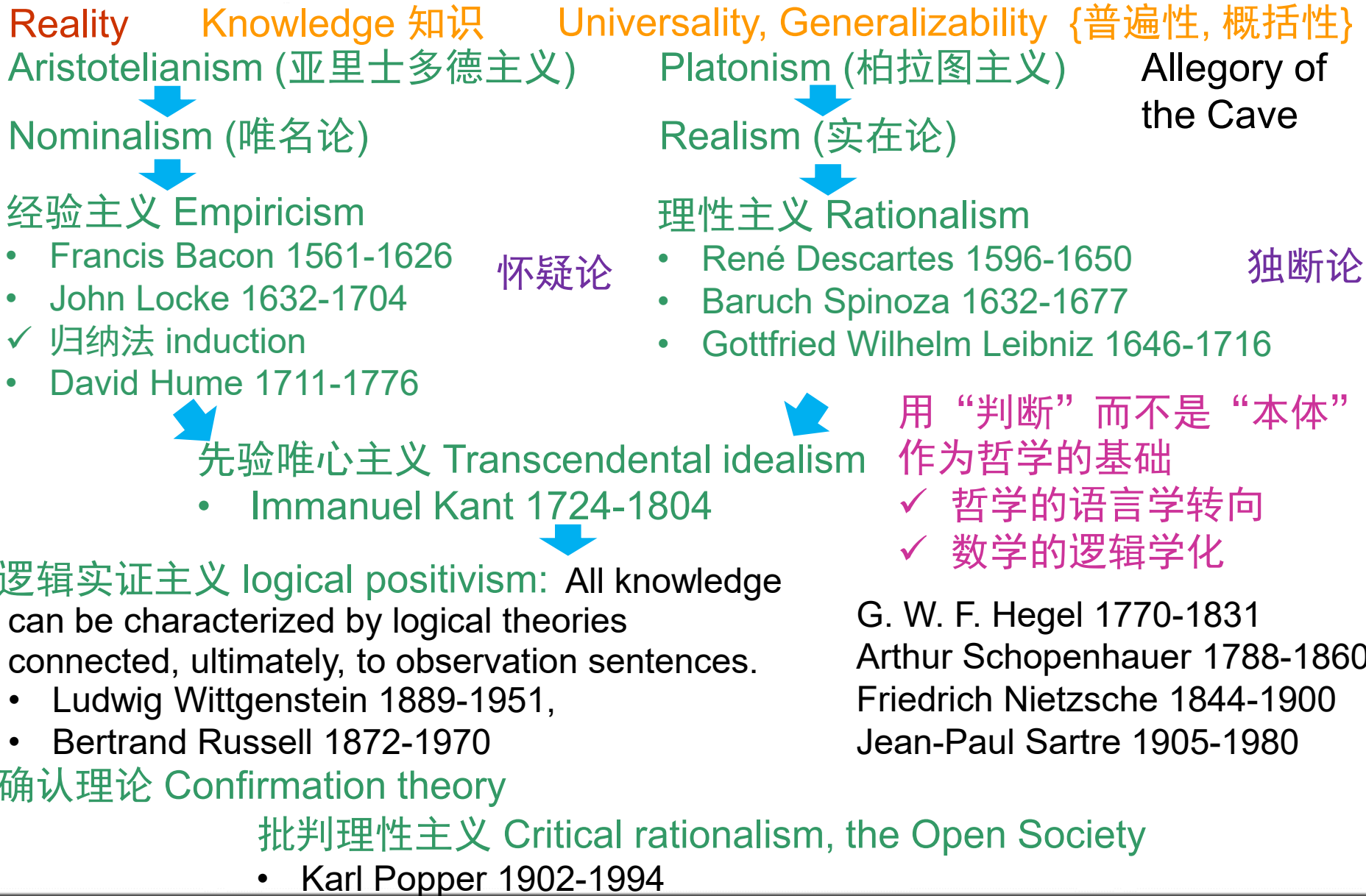
Ontology {本体论} is a branch of philosophy concerned with the nature of being, existence, and reality. It explores fundamental questions about:

- ① **What exists?:** What are the different types of things in the universe? For example, do numbers, minds, and physical objects all exist?
- ② **What are the fundamental categories of being?:** How do we classify entities in the universe? For example, are there categories for material objects, events, ideas, etc.?
- ③ **How do entities relate to each other?:** How are different kinds of entities related? Is there a hierarchy or structure to the categories of being?

Ontology aims to define the basic categories and structures that constitute the world, both in terms of physical objects and abstract concepts.



Philosophy



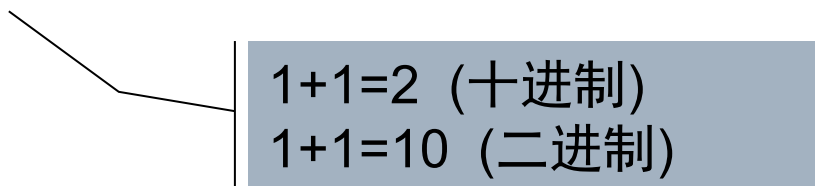
知识表示概述

● 知识的特性

➤ 相对正确性

任何知识都是在一定的条件及环境下产生的，在这种条件及环境下才是正确的。

范畴



王安石(1021~1086):西风昨夜过园林

吹落黄花满地金

苏 轼(1037~1101):秋花不比春花落

说与诗人仔细吟

后来，苏轼被贬到黄州任团练使，见到了落花的菊花。

知识表示概述

➤ 不确定性

“如果头痛且流涕，则有可能患了感冒”

① 随机性引起的不确定性

《三国演义》火烧赤壁：

操升帐谓众谋士曰：若非天命助吾，安得凤雏妙计？铁索连舟，果然渡江如履平地。

程昱曰：彼若用火攻，难以回避，不得不防。

操曰：凡用火攻，必借风力。方今隆冬之际，但有西风北风，安有东风南风耶？吾居于西北之上，彼兵皆在南岸，彼若用火，是烧自己之兵也，吾何惧哉？

诸将皆拜服曰：丞相高见，众人不及。

② 模糊性引起的不确定性

模糊的概念



天气冷热



雨的大小



风的强弱



人的胖瘦



年龄大小



个子高低

知识表示概述

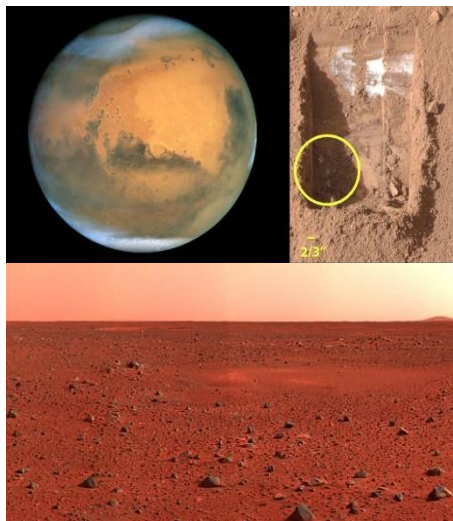
③ 经验引起的不确定性

尽管领域专家能够运用在长期实践及研究中积累起来的经验性知识，但精确表述出来却相当困难。

另外由于经验性自身就蕴涵着不精确性及模糊性，这就形成了知识的不确定性。

④ 不完全性引起的不确定性

火星上可能有水、生命？



按知识的确定性

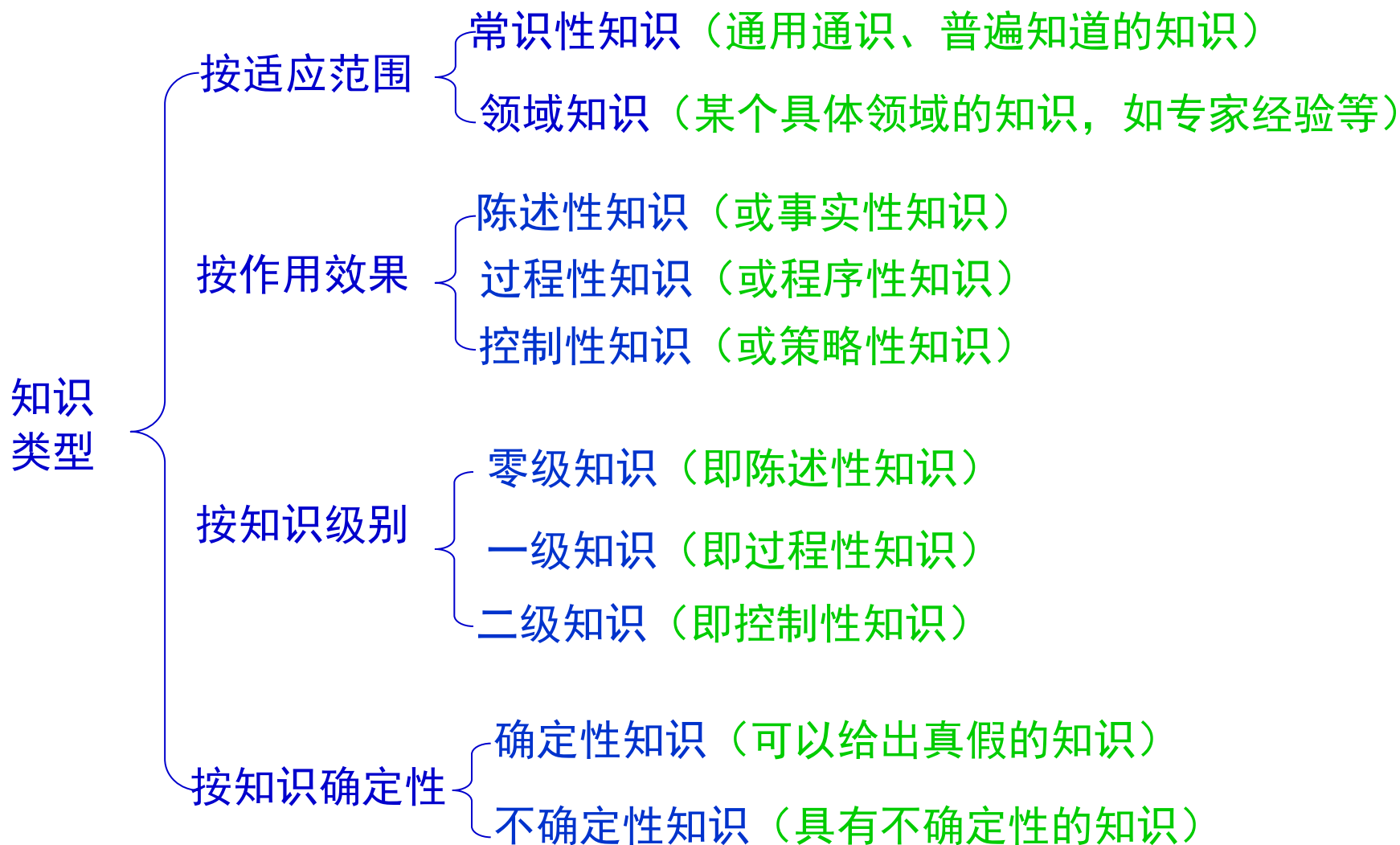
- 确定性知识：可以给出其“真”、“假”的知识。
- 不确定性知识：具有不确定特性（不精确、模糊、随机、不完备、非单调）的知识。

知识表示概述

➤ 知识的层级

- **陈述性知识**或事实性知识（零级）：用于描述事物的概念、定义、属性，或状态、环境、条件等；回答“是什么？”、“为是么？”
- **过程性知识**或程序性知识（一级）：用于问题求解过程的操作、演算和行为的知识，即如何使用事实性知识的知识。回答“怎么做？”
- **控制性知识**或策略性知识（二级）：是关于如何使用过程性知识的知识，如：推理策略、搜索策略、不确定性的传播策略。

知识表示概述



知识表示概述

● 知识表示 (knowledge representation)

知识表示 (knowledge representation) 是将人的知识形式化或者模型化。是对知识的描述，即用一组符号把知识**编码**成计算机可以接受的某种结构。其表示方法不唯一。

知识表示的要求

- **表示能力：**
是指能否正确、有效地将问题求解所需要的知识表示出来。
- **可利用性：**是指表示方法应有利于进行有效的知识推理。包括：对推理的适应性，对高效算法的支持程度
- **可组织性与可维护性：**
可组织性是指可以按某种方式把知识组织成某种知识结构。
可维护性是指要便于对知识的增、删、改等操作
- **可理解性与可实现性：**
可理解性是指知识应易读、易懂、易获取等
可实现性是指知识的表示要便于计算机上实现

知识表示概述

➤ 选择知识表示方法的原则：

- 充分表示领域知识。
- 有利于对知识的利用。
- 便于对知识的组织、维护与管理。
- 便于理解与实现。

知识表示方法的类型 按知识的不同存在方式，可分为：

- **陈述性知识表示**：知识用某种数据结构来表示；知识本身和使用知识的过程相分离。
- **过程性知识表示**：知识和使用知识的过程结合在一起。 **机器学习模型**

陈述性知识表示方法

- **非结构化方法**：
一阶谓词逻辑, 产生式规则
- **结构化方法**：
语义网络 (Semantic Network), 框架 (frame)

知识表示的其它方法

状态空间法和问题归约法等。

推理的概念

推理的心理学观点

按照心理学的观点，推理是由具体事例归纳出 (induce) 一般规律，或者根据已有知识推出新的结论的思维过程。比较典型的观点：

- **结构观点：**推理由两个以上的判断所组成，是一种对已有判断进行分析和综合，再得出新的判断的过程。

例如，若有以下两个判断：

- ① 计算机系的学生都会编程序；
- ② 程强是计算机系的一名学生；

则可得出下面第三个判断：

- ③ 程强会编程序。

- **过程观点：**认为推理是在给定信息和已有知识的基础上的一系列加工操作。

Kurtz 提出了如下人类推理的公式：

$$y = F(x, k)$$

其中， x 为推理时给出的信息， k 为推理时可用的领域知识和特殊事例， F 为可用的一系列操作， y 为推理过程所得到的结论。

推理的概念

推理的心理学过程

从心理学的角度，推理是一种心理过程。可有以下几种主要形式：

(1) **三段论推理**，它是由两个假定真实的前提和一个可能符合也可能不符合这两前提的结论组成。例如，上面给出的计算机系学生的例子。

(2) **线性推理**，或称线性三段论，这种推理的三个判断之间具有线性关系。例如“5比4大”、4比3大”，因此可推出“5比3大”。

(3) **条件推理**，即前一命题是后一命题的条件，例如，“如果一个系统会使用知识进行推理，那么我们就称它为智能系统”。

(4) **概率推理**，即用概率来表示知识的不确定性，并根据所给出的概率来估计新的概率。

推理的机器实现

人工智能中的推理是由推理机完成的。所谓推理机，是指系统中用来实现推理的那段程序。

根据推理所用知识的不同，推理方式和推理方法的不同，推理机的构造也有所不同。

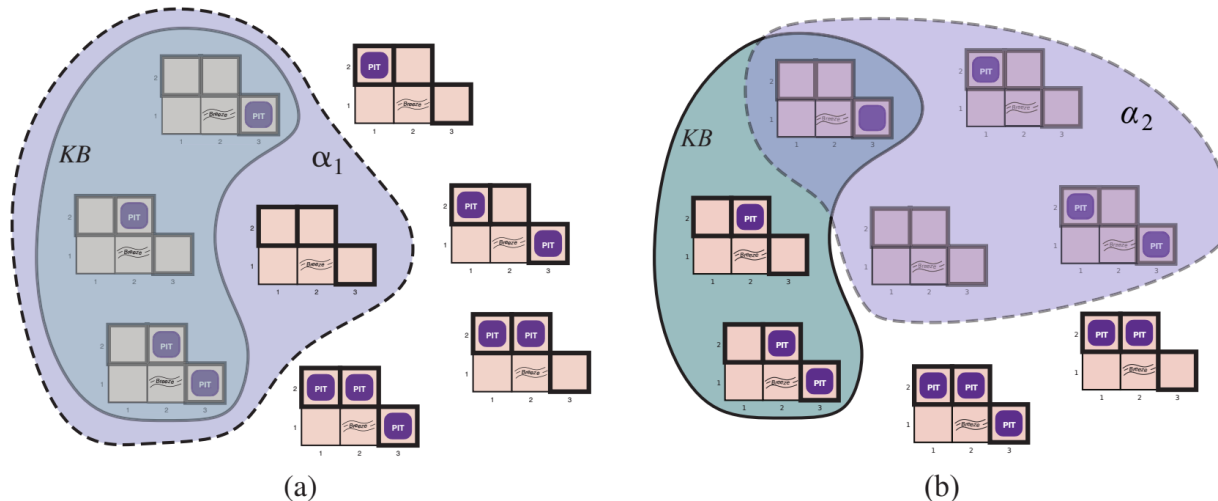
Logic

Sentences are expressed according to **the syntax of the representation language**.

A logic must also define the semantics or meaning of sentences. The **semantics** defines the truth of each sentence with respect to each possible world.

Use the term **model** in place of “**possible world**”. If a sentence α is true in model m , we say that m satisfies α or sometimes m is a model of α . The notation $M(\alpha)$ means **the set of all models of α** .

Entailment (逻辑蕴涵): $\alpha \models \beta$ if and only if $M(\alpha) \subseteq M(\beta)$



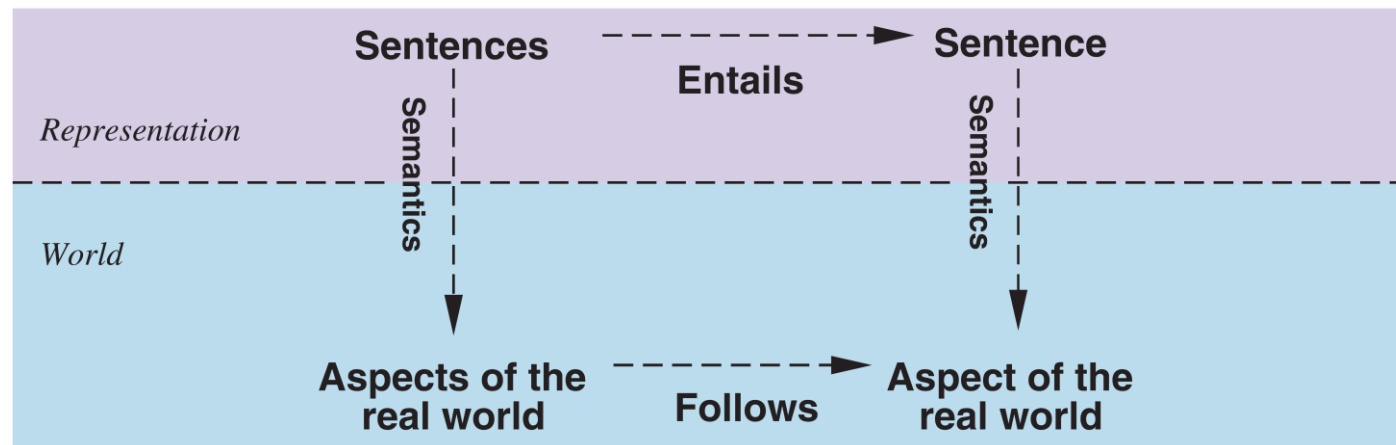
Logic

The KB can be thought of as **a set of sentences** or as a **single sentence** that asserts all the individual sentences. The KB is false in models that contradict what the agent knows.

If an **inference algorithm** i can derive α from KB, we write $KB \vdash_i \alpha$

An **inference algorithm** that derives only entailed sentences is called **sound** or **truth-preserving**. Soundness is a highly desirable property.

An inference algorithm is **complete** if it can derive any sentence that is entailed.



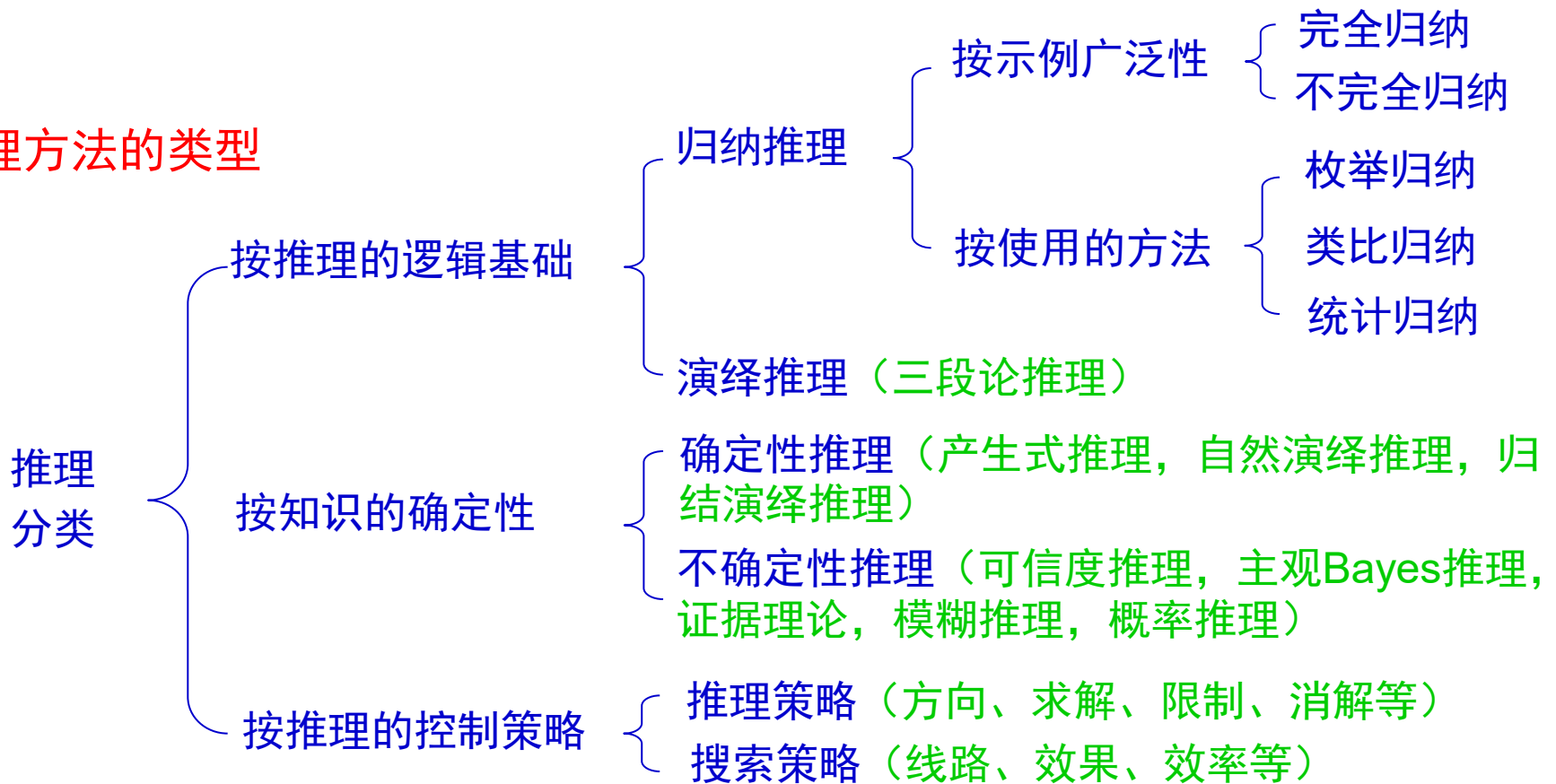
Sentences are physical configurations of the agent, and reasoning is a process of constructing new physical configurations from old ones. Logical reasoning should ensure that the new configurations represent aspects of the world that actually follow from the aspects that the old configurations represent.

推理方法

推理方法：是指实现推理的具体办法。

推理解决的主要问题：推理过程中前提与结论之间的逻辑关系；以及不确定性推理中不确定性的传递问题。

推理方法的类型



内容概要

1. 基于知识的智能体

2. 知识表示和推理概述

3. 语义网络与知识图谱

4. 知识图谱推理

Bibliography:

- [1] 王万良 编著, “人工智能导论” (第5版), 高等教育出版社, 2020. Ch 2, Ch3
- [3] Stuart J. Russel, Peter Norvig, “Artificial Intelligence: A Modern Approach” (4th Ed., 2020), Pearson; 中译版 “人工智能 现代方法” (4rd Ed., 2022), 人民邮电出版社. Ch 7, Ch 10
- [4] 王万森 编著, “人工智能原理及其应用” (第4版), 电子工业出版社, 2018. Ch 2

知识表示方法

● 语义网络 (Semantic Network)

语义网络是一种用实体 (entity) 及其语义关系来表达知识的有向图 (directed graph)。

结点：代表实体，表示事物、概念、情况、属性、状态、事件、动作等。

弧：代表语义关系，表示所连两个实体之间的语义联系，必须带有标识。

语义基元

语义网络中最基本的语义单元称为语义基元，可用三元组表示为：

(结点1, 弧, 结点2)



图：基本网元结构

基本网元

指一个语义基元对应的有向图，是语义网络中最基本的结构单元

例如：语义基元 (A, R, B) 所对应的基本网元，如上图。

例2.6 用语义基元表示“鸵鸟是一种鸟”这一事实。

解：如右图。



图：鸵鸟的基本网元

说明：弧的方向不可随意调换。

知识表示方法

● 语义网络 (Semantic Network)

语义网络是一种用实体 (entity) 及其语义关系来表达知识的有向图 (directed graph)。

结点：代表实体，表示事物、概念、情况、属性、状态、事件、动作等。

弧：代表语义关系，表示所连两个实体之间的语义联系，必须带有标识。

语义基元

语义网络中最基本的语义单元称为语义基元，可用三元组表示为：

(结点1, 弧, 结点2)



图：基本网元结构

基本网元

指一个语义基元对应的有向图，是语义网络中最基本的结构单元

例如：语义基元 (A, R, B) 所对应的基本网元，如上图。

例2.6 用语义基元表示“鸵鸟是一种鸟”这一事实。

解：如右图。



图：鸵鸟的基本网元

说明：弧的方向不可随意调换。

知识表示方法

➤ 语义网络 基本语义关系

实例关系：ISA

体现的是“具体与抽象”的概念，含义为“是一个”，表示一件事物是另一件事物的一个实例。例“李刚是一个人”，如图。



图：实例关系



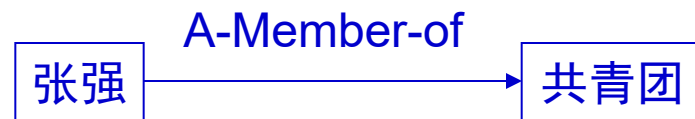
图：分类关系

分类关系：AKO

也称泛化关系，体现的是“子类与超类”的概念，含义为“是一种”，表示一个事物是另一个事物的一种类型。例“机器人是一种机器”，如图。

成员关系：A-Member-of

体现的是“个体与集体”的关系，含义为“是一员”，表示一个事物是另一个事物的一个成员。例“张强是共青团员”，如图。



图：成员关系

上述关系的主要特征

属性的继承性，即处在具体层的结点可以继承抽象层结点的属性。

知识表示方法

属性关系

指事物和其属性之间的关系。常用的有：

Have: 含义为“有”，表示一个结点具有另一个结点所描述的属性

Can: 含义为“能”、“会”，表示一个结点能做另一个结点的事情

例如：“鸟有翅膀”，如图



图：属性关系

包含关系（聚类关系）

指具有组织或结构特征的“部分与整体”之间的关系。常用的包含关系是：

Part-of: 含义为“是一部分”，表示一个事物是另一个事物的一部分。

例如，“大脑是人体的一部分”，如图

再如，“黑板是墙体的一部分”，如图



图：包含关系一



图：包含关系二

聚类关系与实例、分类、成员关系的主要区别

聚类关系一般不具备属性的继承性。

如上例，大脑不一定具有人的各种属性

黑板也不具有墙的各种属性。

知识表示方法

时间关系

指不同事件在其发生时间方面的先后次序关系。常用的时间关系有：

Before: 含义为“在前”

After: 含义为“在后”

如：“机器人Master在机器人AlphaGo之后”，如图



图：时间关系

位置关系

指不同事物在位置方面的关系。常用的有：

Located-on: 含义为“在...上面”

Located-under: 含义为“在...下面”

Located-at: 含义为“在...”

如，“书在桌子上”，如图



图：位置关系

相近关系

指不同事物在形状、内容等方面相似或接近。常用的相近关系有：

Similar-to: 含义为“相似”

Near-to: 含义为“接近”

如，“猫似虎”，如图



图：相似关系

知识表示方法

➤ 用语义网络表示一元关系

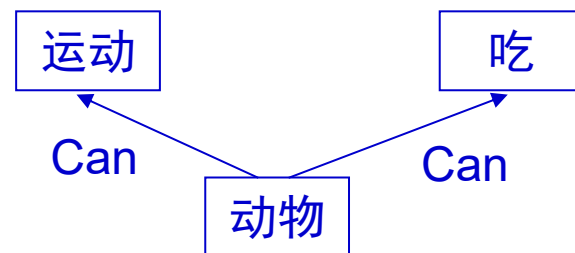
一元关系是指可以用一元谓词 $P(x)$ 表示的关系。谓词 P 说明实体的性质、属性等。

常用：“是”、“有”、“会”、“能”等语义关系来说明。如，“雪是白的”。

一元关系的描述

一个一元关系就是一个语义基元，可用一个基本网元来表示。其中，结点1表示实体，结点2表示实体的性质或属性等，弧表示语义关系。

如前，“李刚是一个人”为一元关系，其语义网络如前所示。



例2.7 用语义网络表示“动物能运动、会吃”。

图：动物的属性

知识表示方法

➤ 用语义网络表示二元关系

二元关系是指可用二元谓词 $P(x, y)$ 表示的关系。其中, x, y 为实体, P 为实体之间的关系。

单个二元关系可直接用一个基本网元来表示。

有些**复杂关系**, 可以分解成一些相对独立的二元或一元关系。

例2.8 用语义网络表示:

动物能运动、会吃。

鸟是一种动物, 鸟有翅膀、会飞。

鱼是一种动物, 鱼生活在水中、会游泳。

解: 其语义网络表示如图所示

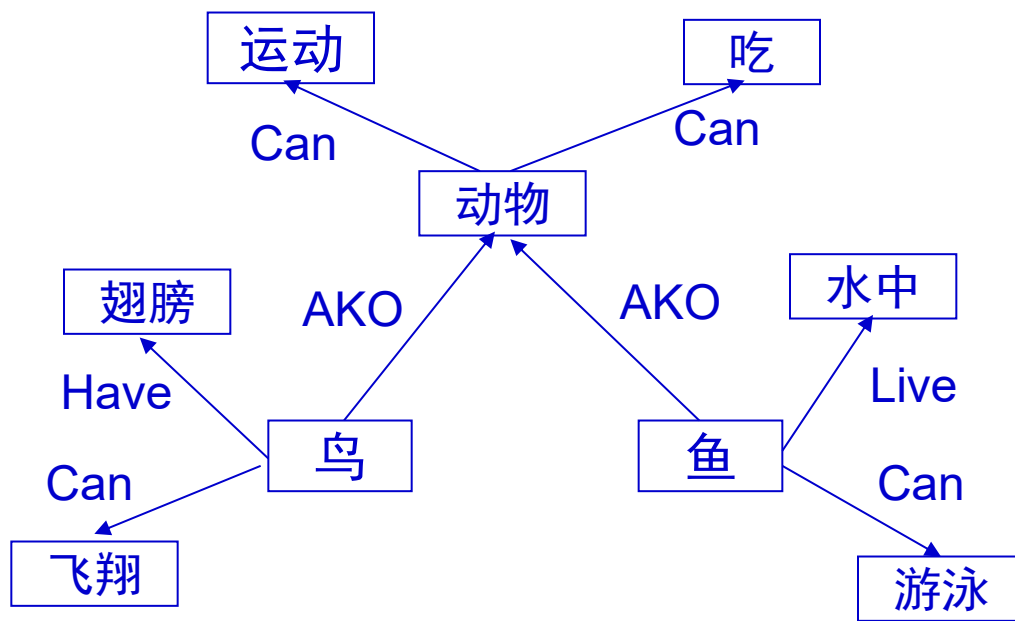


图: 动物分类的语义网络

知识表示方法

➤ 用语义网络表示多元关系

多元关系

是指可用多元谓词 $P(x_1, x_2, \dots)$ 表示的关系。其中，个体 x_1, x_2, \dots 为实体，谓词 P 说明这些实体之间的关系。

多元关系的表示法

用语义网络表示多元关系时，可把它转化为多个一元或二元关系的组合，然后再利用合取关系的表示方法，把这种多元关系表示出来。

知识表示方法

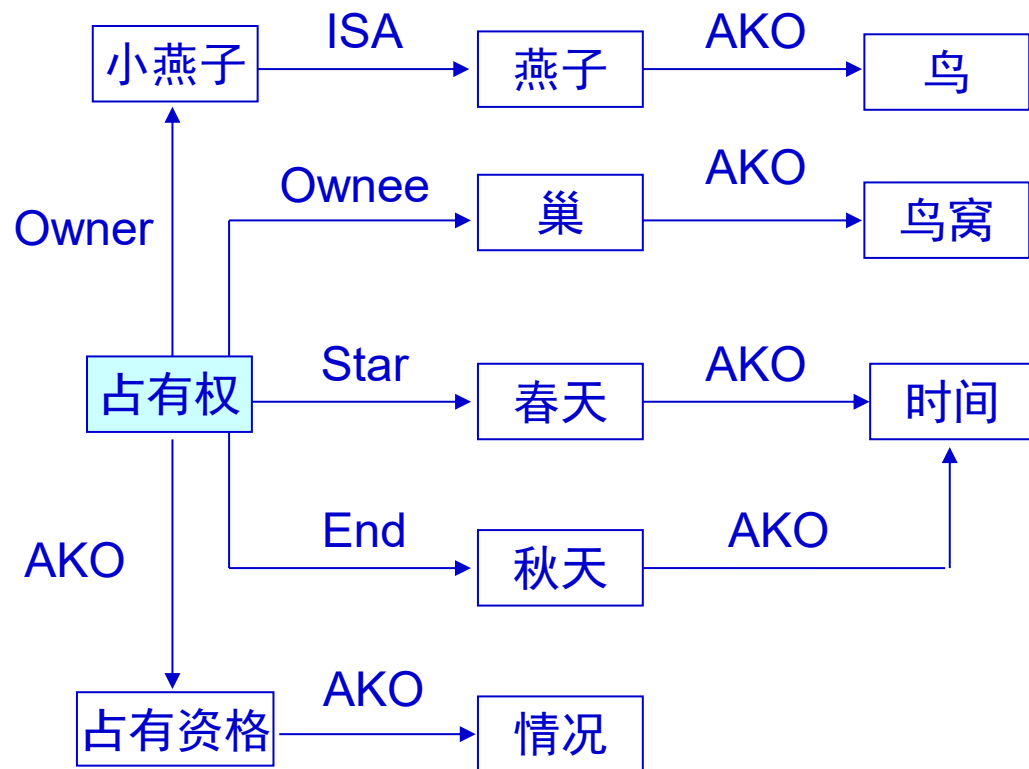
情况的表示

表示方法：西蒙提出了增加情况和动作结点的描述方法。

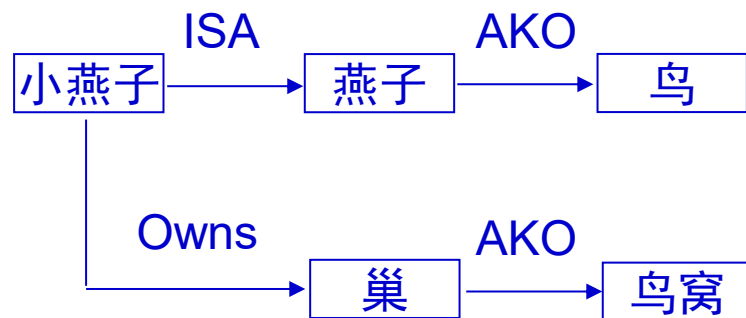
例2.11：用语义网络表示：

“小燕子这只燕子从春天到秋天占有一个巢”

解：需要设立一个占有权结点，表示占有物和占有时间等。



对上述问题，也可以把占有作为一种二元关系，并用一条弧来表示，但在这种表示方法下，占有关系就无法表示了



知识表示方法

➤ 事件和动作的表示

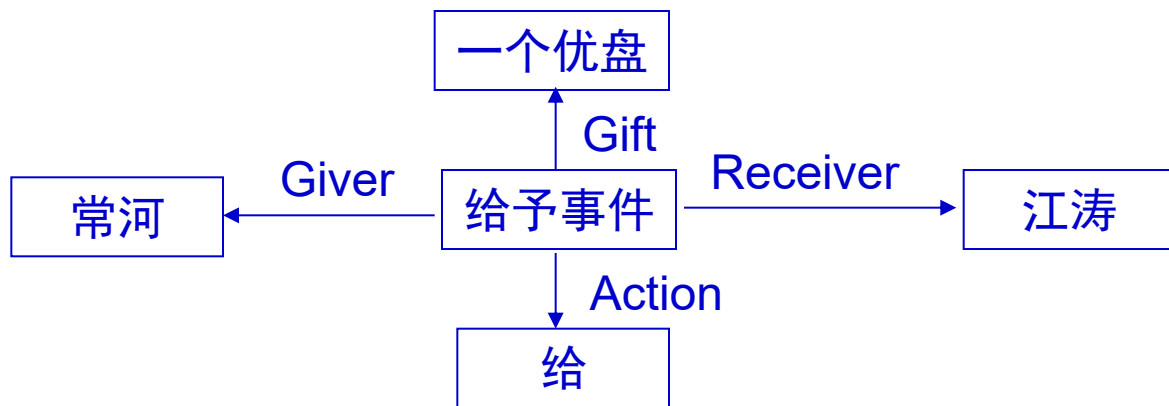
用这种方法表示事件或动作时，需要设立一个事件节点或动作结点。其中，事件节点由一些向外引出的弧来指出事件行为及发出者与接受者。动作结点由一些向外引出的弧来指出动作的主体与客体。

例2.13 用于语义网络表示：

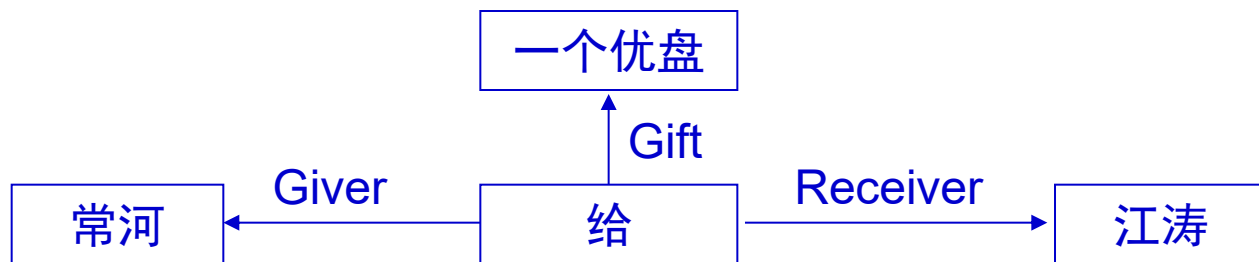
“常河给江涛一个优盘”

解：用事件节点表示如右图所示。

用动作结点节点表示如下图所示。



例2.13 图1 带有事件节点的语义网络



例2.13 图2 带有动作节点的语义网络

知识表示方法

➤ 语义网络的基本推理过程

用语义网络表示知识的问题求解系统主要由两大部分所组成，一部分是由语义网络构成的知识库，另一部分是用于问题求解的推理机构。语义网络的推理过程主要有两种，一种是继承，另一种是匹配。

✓ 继承的概念

是指把对事物的描述从抽象结点传递到实例结点。通过继承可以得到所需结点的一些属性值，它通常是沿着ISA、AKO等继承弧进行的。

继承的一般过程

- (1) 建立一个结点表，用来存放待求解结点和所有以ISA、AKO等继承弧与此结点相连的那些结点。初始情况下，表中只有待求解结点。
- (2) 检查表中的第一个结点是否有继承弧。如果有，就把该弧所指的所有结点放入结点表的末尾，记录这些结点的所有属性，并从结点表中删除第一个结点。如果没有继承弧，仅从结点表中删除第一个结点。
- (3) 重复(2)，直到结点表为空。此时，记录下来的所有属性都是待求解结点继承来的属性。

继承的例子

在图2-18所示的语义网络中，通过继承关系可以得到“鸟”具有“会吃”、“能运动”的属性

知识表示方法

➤ 语义网络的基本推理过程

✓ **匹配**是指在知识库的语义网络中寻找与待求解问题相符的语义网络模式。

匹配的过程

- (1) 根据待求解问题的要求构造一个网络片断，该网络片断中有些结点或弧的标识是空的，称为询问处，它反映的是待求解的问题。
- (2) 根据该语义片断到知识库中去寻找所需要的信息。
- (3) 当待求解问题的网络片断与知识库中的某语义网络片断相匹配时，则与询问处相匹配的事实就是问题的解。

匹配的例子

例2.13：假设图2.18的语义网络已在知识库中，问王强在哪个公司工作？

解：根据这个问题的要求，可构造如如下语义网络片断。



当用该语义网络片断与图2-18所示的语义网络进行匹配时，由“Work-for”弧所指的结点可知，职员王强工作在“理想公司”，这就得到了问题的答案。若还想知道职员王强的其它情况，则可在语义网络中增加相应的空结点。

知识表示方法

➤ 语义网络表示的特性

主要优点：

- **结构性** 把事物的属性以及事物间的各种语义联系显式地表示出来，是一种结构化的知识表示方法。在这种方法中，下层结点可以继承、新增、变异上层结点的属性。
- **联想性** 本来是作为人类联想记忆模型提出来的，它着重强调事物间的语义联系，体现了人类的联想思维过程。
- **自然性** 语义网络可以比较直观把知识表示出来，符合人们表达事物间关系的习惯。

主要缺点：

- **非严格性** 没有象谓词那样严格的形式表示体系，一个给定语义网络的含义完全依赖于处理程序对它所进行的解释，通过语义网络所实现的推理不能保证其正确性。
- **复杂性** 语义网络表示知识的手段是多种多样的，这虽然对其表示带来了灵活性，但同时也由于表示形式的不一致，使得它的处理增加了复杂性。

内容概要

1. 基于知识的智能体

2. 知识表示和推理概述

3. 语义网络与知识图谱

4. 知识图谱推理

- 知识图谱 (Knowledge Graph)

Bibliography:

[1] 王万良 编著, “人工智能导论” (第5版), 高等教育出版社, 2020. Ch 2.5

知识表示方法

● 知识图谱 (Knowledge Graph)

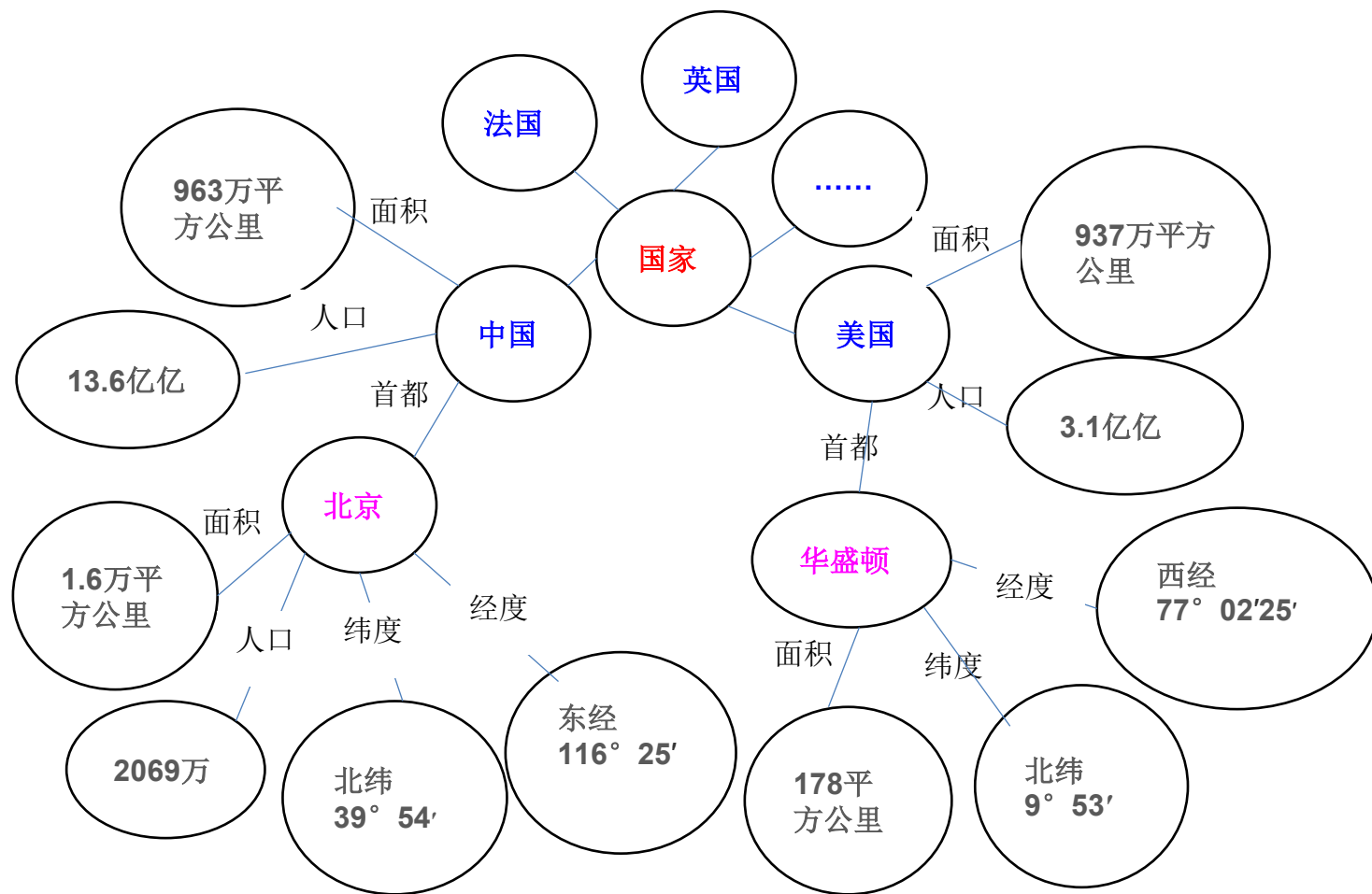
- 由于互联网内容的大规模、异质多元、组织结构松散的特点，给人们有效获取信息和知识提出了挑战。谷歌于2012年5月16日首先发布了知识图谱 (Knowledge Graph)。
- 知识图谱是一种互联网环境下的知识表示方法。
- 知识图谱的目的是为了提高搜索引擎的能力，改善用户的搜索质量以及搜索体验。
- Google、百度和搜狗等搜索引擎公司构建的知识图谱，分别称为知识图谱、知心和知立方。

知识图谱 (Knowledge Graph/Vault)，又称科学知识图谱，用各种不同的图形等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。

- 知识图谱是由一些相互连接的实体及其属性构成的。
- 三元组是知识图谱的一种通用表示方式：
 - (实体1-关系-实体2)：中国-首都-北京
 - (实体-属性-属性值)：北京-人口-2069万

知识表示方法

知识图谱也可被看作是一个图(graph)，图中的节点表示实体 (entity) 或概念 (concept)，而图中的边则由属性 (attribute) 或关系 (relation) 构成。



知识表示方法

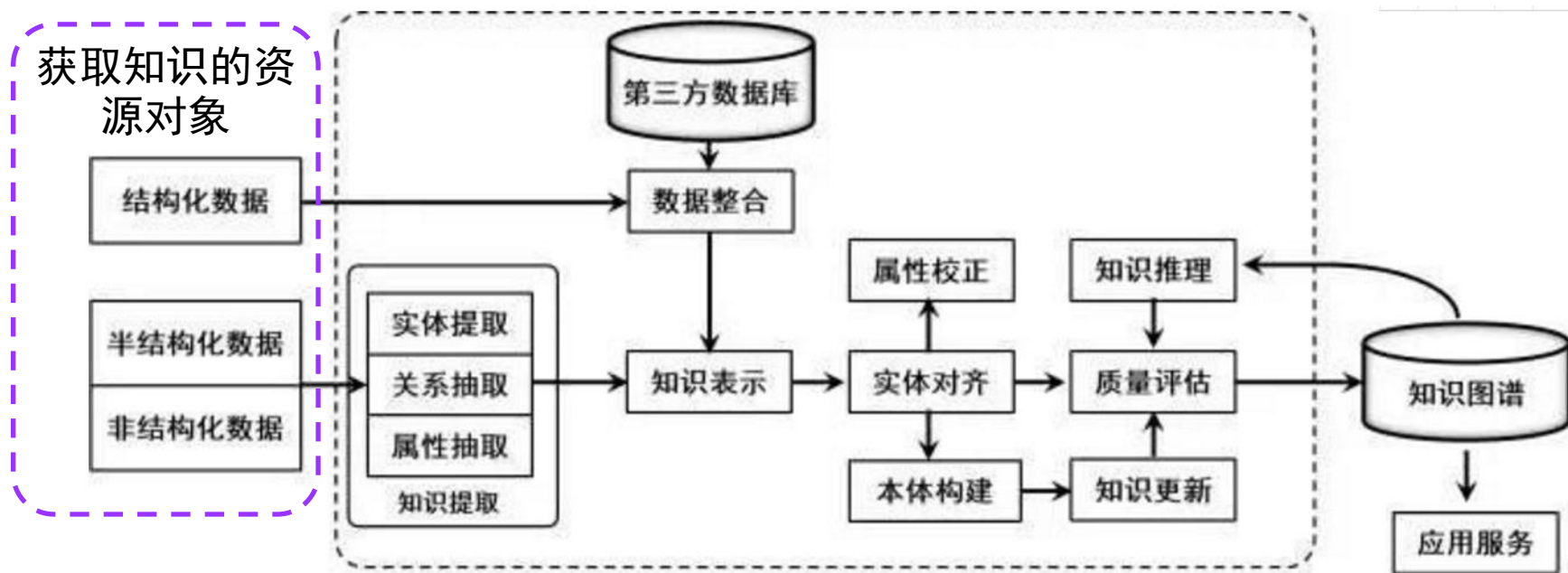
➤ 知识图谱的逻辑结构：模式层与数据层

数据层主要是由一系列的事实组成，而知识以事实为单位进行存储。可选择图数据库作为存储介质。

模式层构建在数据层之上，是知识图谱的核心。通常采用**本体 (ontology)** 库来管理。

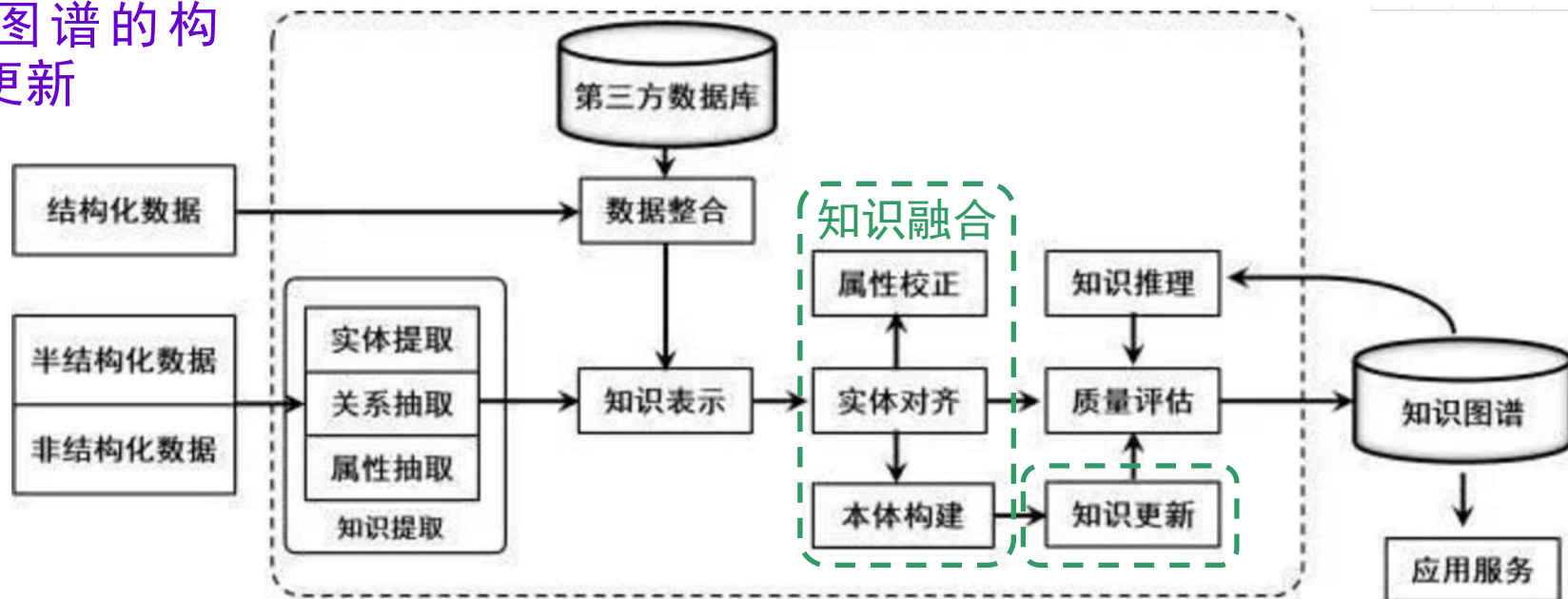
本体 (ontology) 是结构化知识库的概念模板，通过本体库形成的知识库不仅层次结构较强，并且冗余程度较小。

➤ 知识图谱的体系架构 知识图谱的体系架构是指其构建模式结构



知识表示方法

➤ 知识图谱的构建和更新



采用一系列自动或半自动技术（包括机器学习和信息抽取技术），从原始或第三方数据库中提取知识事实，并将其存入知识库的数据层和模式层，包括：**知识提取**、**知识表示**、**知识融合**、**知识推理**四个过程。

目前大多数通用的知识图谱通过对维基百科进行结构化来构建。

- ✓ **自顶向下 (top-down)** 指的是先为知识图谱定义好本体与数据模式，再将实体加入到知识库。需要利用一些现有的结构化知识库作为基础知识库。
- ✓ **自底向上 (bottom-up)** 指的是从一些开放链接数据中提取出实体，选择其中置信度较高的加入到知识库，再构建顶层的本体模式。

知识表示方法

➤ 知识图谱的典型应用

维基百科 (Wikipedia) 由维基媒体基金会负责运营的一个自由内容、自由编辑的多语言知识库。

Freebase 已经关闭。绝大部分数据从维基百科中得到。

Dbpedia 由2007年德国柏林自由大学以及莱比锡大学的研究者从维基百科里萃取结构化知识的项目开始建立

YAGO 由德国马克斯-普朗克研究所（MPI）构建的大型多语言的语义知识库，从10个维基百科以不同语言提取事实和事实的组合。

XLORE 是清华大学构建的基于中、英文维基和百度百科的开放知识平台，是第一个中英文知识规模较为平衡的大规模中英文知识图谱。

内容概要

1. 基于知识的智能体

2. 知识表示和推理概述

3. 语义网络与知识图谱

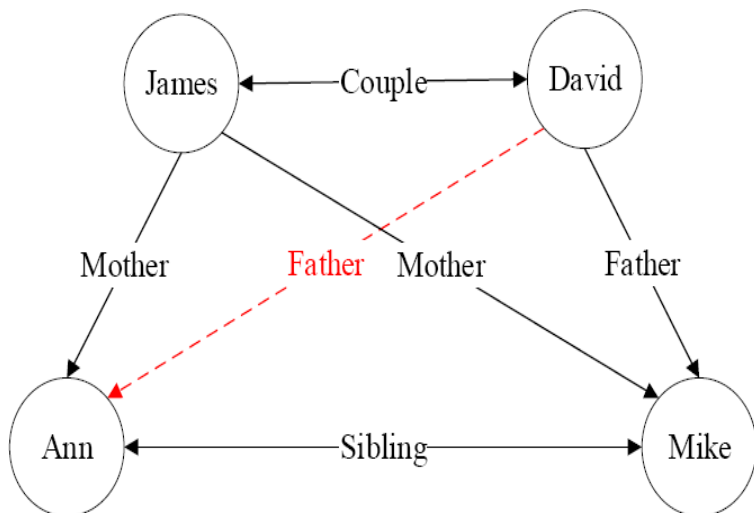
4. 知识图谱推理

Bibliography:

[2] 吴飞 编著, “人工智能导论: 模型与算法”, 高等教育出版社, 2020. Ch 2.1-2.3

知识图谱推理

- 知识图谱中存在连线的两个实体可表达为 $\langle \text{left_node}, \text{relation}, \text{right_node} \rangle$ 的三元组形式，这种三元组也可以表示为一阶逻辑(first order logic, FOL)的形式，从而为基于知识图谱的推理创造了条件。
- 可利用一阶谓词来表达刻画知识图谱中节点之间存在的关系。图谱中形如 $\langle \text{James}, \text{Couple}, \text{David} \rangle$ 的关系可用一阶逻辑的形式描述，即 $\text{Couple}(\text{James}, \text{David})$ 。 $\text{Couple}(x, y)$ 是一阶谓词， Couple 是图中实体之间具有的关系， x 和 y 是谓词变量
- 从图中已有关系可推知David和Ann具有 father 关系，但这一关系在图中初始图(无红线)中并不存在，是需要推理的目标。



一个简单的家庭关系知识图谱

问题：如何从知识图谱中推理得到

$\text{father}(\text{David}, \text{Ann})$



$$(\forall x)(\forall y)(\forall z)(\text{Mother}(z, y) \wedge \text{Couple}(x, z) \rightarrow \text{Father}(x, y))$$

如果能够学习得到这条规则，该有多好？
(从具体例子中学习，这是归纳推理的范畴)

知识图谱推理

- 归纳学习 (inductive learning)

归纳逻辑程序设计 (inductive logic programming, ILP) 算法：归纳逻辑程序设计 (ILP) 是机器学习和逻辑程序设计交叉领域的研究内容。

ILP使用一阶谓词逻辑进行知识表示，通过修改和扩充逻辑表达式对现有知识归纳，完成推理任务。

作为ILP的代表性方法，FOIL (First Order Inductive Learner) 通过序贯覆盖实现规则推理。

$$(\forall x)(\forall y)(\forall z)(Mother(z, y) \wedge Couple(x, z) \rightarrow Father(x, y))$$



前提约束谓词
(学习得到)



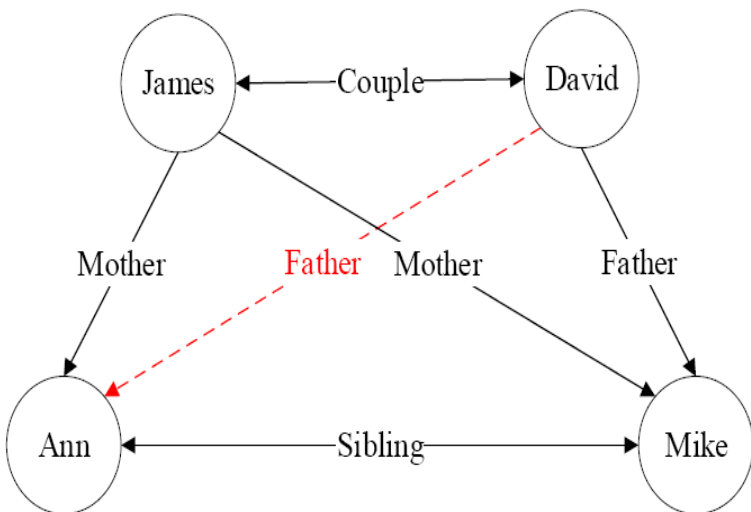
目标谓词
(已知)

推理手段:

positive examples + *negative examples* + *background knowledge examples*
 \Rightarrow **hypothesis**

知识图谱推理

● 归纳学习 FOIL (First Order Inductive Learner)



一个简单的家庭关系知识图谱

- 目标谓词: $Father(x, y)$
- 目标谓词只有一个正例 $Father(David, Mike)$ 。
- 反例在知识图谱中一般不会显式给出, 但可从知识图谱中构造出来。

如从知识图谱中已经知道 $Couple(David, James)$ 成立, 则 $Father(David, James)$ 可作为目标谓词 P 的一个反例, 记为 $\neg Father(David, James)$ 。

只能在已知两个实体的关系且确定其关系与目标谓词相悖时, 用于构建目标谓词的反例。不能在不知两个实体是否满足目标谓词前提下将它们用于构造目标谓词的反例。

- 背景知识: 知识图谱中目标谓词以外的其他谓词实例化结果, 如 $Sibling(Ann, Mike)$

知识图谱推理

- 归纳学习 FOIL (First Order Inductive Learner)

$$(\forall x)(\forall y)(\forall z)(Mother(z, y) \wedge Couple(x, z) \rightarrow \boxed{Father(x, y)})$$



前提约束谓词
(学习得到)



目标谓词
(已知)

| | | | |
|--------------|----------------------|----------------|-----------------------------|
| 背景知识 样例集合 | Sibling(Ann, Mike) | 目标谓词 训练样例集合 | Father(David, Mike) |
| | Couple(David, James) | | \neg Father(David, James) |
| | Mother(James, Ann) | | \neg Father(James, Ann) |
| | Mother(James, Mike) | | \neg Father(James, Mike) |
| | | | \neg Father(Ann, Mike) |

知识图谱推理

- 归纳学习 FOIL (First Order Inductive Learner)

$$(\forall x)(\forall y)(\forall z)(Mother(z, y) \wedge Couple(x, z) \rightarrow \boxed{Father(x, y)})$$



前提约束谓词
(学习得到)



目标谓词
(已知)

推理思路：从一般到特殊，逐步给目标谓词添加前提约束谓词，直到所构成的推理规则不覆盖任何反例。

从一般到特殊：对目标谓词或前提约束谓词中的变量赋予具体值，如将 $(\forall x)(\forall y)(\forall z)(Mother(z, y) \wedge Couple(x, z) \rightarrow Father(x, y))$ 这一推理规则所包含的目标谓词 $Father(x, y)$ 中 x 和 y 分别赋值为David和Ann，进而进行推理。

知识图谱推理

- 归纳学习 FOIL (First Order Inductive Learner)

$$(\forall x)(\forall y)(\forall z)(Mother(z, y) \wedge Couple(x, z) \rightarrow \boxed{Father(x, y)})$$



前提约束谓词
(学习得到)



目标谓词
(已知)

哪些谓词好呢？可
以作为目标谓词的
前提约束谓词？

FOIL中信息增益值
(information gain)

FOIL信息增益值计算方法如下：

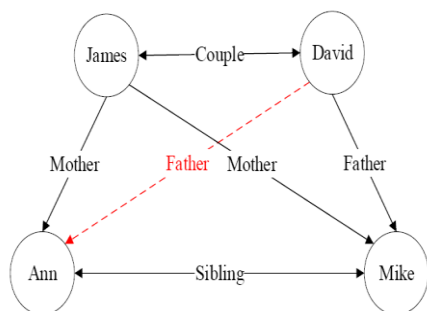
$$FOIL_Gain = \hat{m}_+ \cdot \left(\log_2 \frac{\hat{m}_+}{\hat{m}_+ + \hat{m}_-} - \log_2 \frac{m_+}{m_+ + m_-} \right)$$

其中， \hat{m}_+ 和 \hat{m}_- 是增加前提约束谓词后所得新推理规则覆盖的正例和反例的数量， m_+ 和 m_- 是原推理规则所覆盖的正例和反例数量。

知识图谱推理

归纳学习 FOIL (First Order Inductive Learner)

$$(\forall x)(\forall y)(\forall z)(Mother(z, y) \wedge Couple(x, z) \rightarrow \boxed{Father(x, y)})$$



前提约束谓词
(学习依次加入)

目标谓词
(已知)

- $Mother(\cdot, \cdot)$
- $Sibling(\cdot, \cdot)$
- $Couple(\cdot, \cdot)$

- 依次将谓词加入到推理规则中作为前提约束谓词
- 计算所得到新推理规则的FOIL增益值
- 基于计算所得FOIL增益值来选择最佳前提约束谓词

知识图谱推理

| 推理规则 | | 推理规则涵盖的正例和反例数 | | FOIL信息增益值 |
|---------------------------|----------------------------------|---------------------------------------|---------------------------------------|--------------|
| 目标谓词 | 前提约束谓词 | 正例 | 反例 | 信息增益值 |
| $Father(x, y) \leftarrow$ | 空集 | $m_+ = 1$ | $m_- = 4$ | $FOIL_Gain$ |
| $Father(x, y) \leftarrow$ | $Mother(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 2$ | NA |
| | $Mother(x, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 2$ | NA |
| | $Mother(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Mother(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Mother(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Mother(z, y)$ | $\widehat{m}_+ = 1$ | $\widehat{m}_- = 3$ | 0.32 |
| | $Sibling(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Sibling(x, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Sibling(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Sibling(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Sibling(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Sibling(z, y)$ | $\widehat{m}_+ = 1$ | $\widehat{m}_- = 2$ | 0.74 |
| | $Couple(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Couple(x, z)$ | $\widehat{m}_+ = 1$ | $\widehat{m}_- = 1$ | 1.32 |
| | $Couple(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Couple(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Couple(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 2$ | NA |
| | $Couple(z, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |

给定目标谓词，此时推理规则只有目标谓词，因此推理规则所覆盖的正例和反例的样本数分别是训练样本中正例和反例的数量，即1和4，因此， $m_+ = 1$ ， $m_- = 4$ 。

知识图谱推理

| 推理规则 | | 推理规则涵盖的正例和反例数 | | FOIL信息增益值 |
|---------------------------|----------------------------------|-----------------------------------|-----------------------------------|--------------|
| 目标谓词 | 前提约束谓词 | 正例 | 反例 | 信息增益值 |
| $Father(x, y) \leftarrow$ | 空集 | $m_+ = 1$ | $m_- = 4$ | $FOIL_Gain$ |
| $Father(x, y) \leftarrow$ | $Mother(x, y)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 2$ | NA |
| | $Mother(x, z)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 2$ | NA |
| | $Mother(y, x)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 1$ | NA |
| | $Mother(y, z)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 1$ | NA |
| | $Mother(z, x)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 1$ | NA |
| | $Mother(z, y)$ | $\hat{m}_+ = 1$ | $\hat{m}_- = 3$ | 0.32 |
| | $Sibling(x, y)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 1$ | NA |
| | $Sibling(x, z)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 1$ | NA |
| | $Sibling(y, x)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 0$ | NA |
| | $Sibling(y, z)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 0$ | NA |
| | $Sibling(z, x)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 0$ | NA |
| | $Sibling(z, y)$ | $\hat{m}_+ = 1$ | $\hat{m}_- = 2$ | 0.74 |
| | $Couple(x, y)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 1$ | NA |
| | $Couple(x, z)$ | $\hat{m}_+ = 1$ | $\hat{m}_- = 1$ | 1.32 |
| | $Couple(y, x)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 0$ | NA |
| | $Couple(y, z)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 0$ | NA |
| | $Couple(z, x)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 2$ | NA |
| | $Couple(z, y)$ | $\hat{m}_+ = 0$ | $\hat{m}_- = 1$ | NA |

- 将 $Mother(x, y)$ 作为前提约束谓词加入，可得到推理规则
 $Mother(x, y) \rightarrow Father(x, y)$
- 在背景知识中， $Mother(x, y)$ 有两个实例：
 $Mother(\text{James}, \text{Ann})$
 $Mother(\text{James}, \text{Mike})$
- 对于 $Mother(\text{James}, \text{Ann})$ 这一实例，将 $x = \text{James}, y = \text{Ann}$ 代入 $Father(x, y)$ 得到 $Father(\text{James}, \text{Ann})$ ，是训练样本中一个反例
- 对于 $Mother(\text{James}, \text{Mike})$ 这一实例，将 $x = \text{James}, y = \text{Mike}$ 代入 $Father(x, y)$ 得到 $Father(\text{James}, \text{Mike})$ ，是训练样本中一个反例
- 覆盖正例和反例数量分别为0和2，即 $\hat{m}_+ = 0, \hat{m}_- = 2$
- $FOIL_Gain$ 记为 NA (Not Available)

知识图谱推理

| 推理规则 | | 推理规则涵盖的正例和反例数 | | FOIL信息增益值 |
|---------------------------|----------------------------------|---------------------------------------|---------------------------------------|-----------|
| 目标谓词 | 前提约束谓词 | 正例 | 反例 | 信息增益值 |
| $Father(x, y) \leftarrow$ | 空集 | $m_+ = 1$ | $m_- = 4$ | FOIL_Gain |
| $Father(x, y) \leftarrow$ | $Mother(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 2$ | NA |
| | $Mother(x, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 2$ | NA |
| | $Mother(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Mother(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Mother(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Mother(z, y)$ | $\widehat{m}_+ = 1$ | $\widehat{m}_- = 3$ | 0.32 |
| | $Sibling(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Sibling(x, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Sibling(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Sibling(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Sibling(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Sibling(z, y)$ | $\widehat{m}_+ = 1$ | $\widehat{m}_- = 2$ | 0.74 |
| | $Couple(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $Couple(x, z)$ | $\widehat{m}_+ = 1$ | $\widehat{m}_- = 1$ | 1.32 |
| | $Couple(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Couple(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $Couple(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 2$ | NA |
| | $Couple(z, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |

如果将 $Couple(x, z)$ 作为前提约束谓词加入，可得到如下推理规则
 $Couple(x, z) \rightarrow Father(x, y)$

在背景知识中， $Couple(x, z)$ 只有一个实例 $Couple(David, James)$ ，即 $x=David$ ， $z=James$ ，将其代入 $Father(x, y)$ 得到 $Father(David, y)$ 。

在训练样本中存在正例 $Father(David, Mike)$ 以及反例 $\neg Father(David, James)$ ，即 $Couple(x, z) \rightarrow Father(x, y)$ 覆盖正例和反例数量分别为1和1。信息增益值为：

$$\widehat{m}_+ \cdot \left(\log_2 \frac{\widehat{m}_+}{\widehat{m}_+ + \widehat{m}_-} - \log_2 \frac{m_+}{m_+ + m_-} \right) \\ = 1 \cdot \left(\log_2 \frac{1}{1 + 1} - \log_2 \frac{1}{1 + 4} \right) = 1.32$$

知识图谱推理

| | |
|--------------------------------------|--|
| Back-ground knowledge | Sibling(Ann, Mike) Couple(David, James) Mother(James, Ann) Mother(James, Mike) |
| Positive and negative samples | Father(David, Mike) \neg Father(David, James) \negFather(James, Ann) \negFather(James, Mike) \negFather(Ann, Mike) |

- $Couple(x, z)$ 加入后信息增益最大
- 将 $Couple(x, z)$ 加入推理规则，得到 $Couple(x, z) \rightarrow Father(x, y)$ 新推理规则
- 将训练样例中与该推理规则不符的样例去掉。这里不符指当 $Couple(x, z)$ 中 x 取值为 David 时，与 $Father(David, \quad)$ 或 $\neg Father(David, \quad)$ 无法匹配的实例。
- 去掉无法匹配实例后，训练样本集中只有正例 $Father(David, Mike)$ 和负例 $\neg Father(David, James)$ 两个实例

知识图谱推理

| 推理规则 | | 推理规则涵盖的 正例和反例数 | | FOIL信息增益值 |
|--|--------------------------------|------------------------------|------------------------------|--------------|
| 现有规则 | 拟加入前提 约束谓词 | 正例 | 反例 | 信息增益值 |
| $Father(x, y) \leftarrow Couple(x, z)$ | | $m_+ = 1$ | $m_- = 1$ | 1.32 |
| $Father(x, y) \leftarrow Couple(x, z)$ | $\wedge Mother(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Mother(x, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Mother(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Mother(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Mother(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge \mathbf{Mother(z, y)}$ | $\widehat{m}_+ = \mathbf{1}$ | $\widehat{m}_- = \mathbf{0}$ | $\mathbf{1}$ |
| | $\wedge Sibling(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Sibling(x, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Sibling(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Sibling(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Sibling(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Sibling(z, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Couple(x, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 1$ | NA |
| | $\wedge Couple(x, z)$ | $\widehat{m}_+ = 1$ | $\widehat{m}_- = 1$ | 0 |
| | $\wedge Couple(y, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Couple(y, z)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Couple(z, x)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |
| | $\wedge Couple(z, y)$ | $\widehat{m}_+ = 0$ | $\widehat{m}_- = 0$ | NA |

- $Mother(z, y)$ 加入信息增益最大
- 将 $Mother(z, y)$ 加入，得到新推理规则

$$\mathbf{Mother(z, y) \wedge Couple(x, z) \rightarrow Father(x, y)}$$
- $x = \text{David}, y = \text{Mike}, z = \text{James}$ 时，该推理规则覆盖训练样本集中正例 $Father(\text{David}, \text{Mike})$ 且不覆盖任意反例，因此算法学习结束。

$$\mathbf{Mother(z, y) \wedge Couple(x, z) \rightarrow Father(x, y)}$$



已知：

$Mother(\text{James}, \text{Ann})$

$Couple(\text{David}, \text{James})$

于是： $Father(\text{David}, \text{Ann})$

知识图谱推理

● 归纳学习 FOIL (First Order Inductive Learner)

$$(\forall x)(\forall y)(\forall z)(Mother(z, y) \wedge Couple(x, z) \rightarrow Father(x, y))$$



前提约束谓词
(依次学习得到)



目标谓词
(已知)

推理手段:

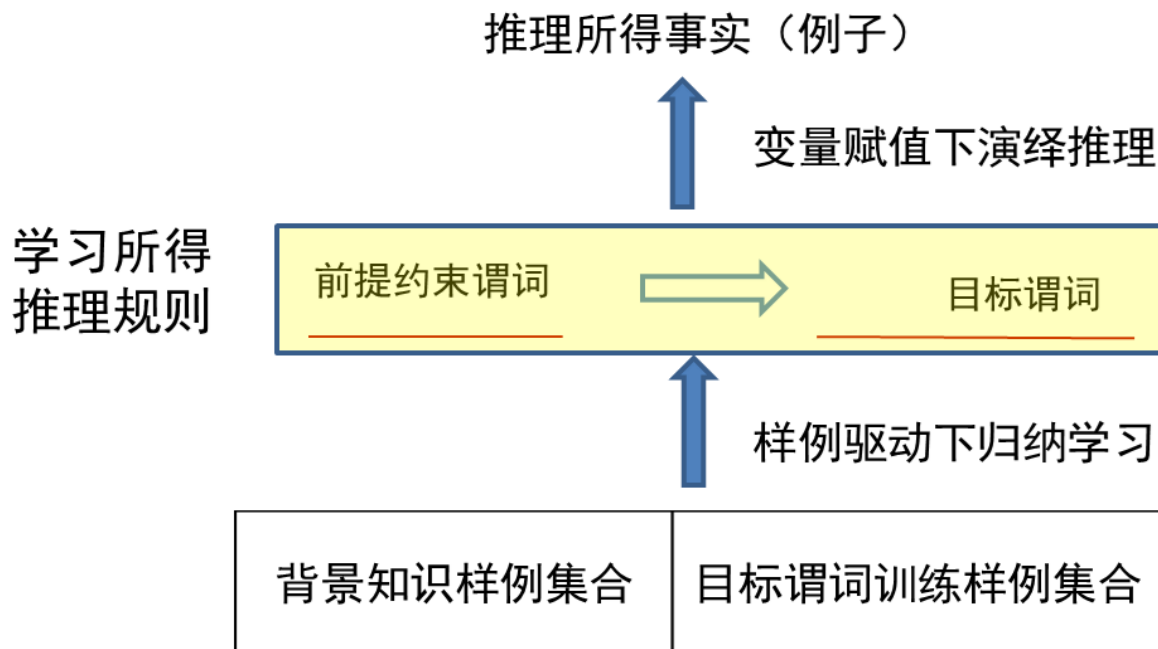
positive examples + *negative examples* + *background knowledge examples* \Rightarrow **hypothesis**

| | | | |
|--------------|----------------------|----------------|-----------------------------|
| 背景知识 样例集合 | Sibling(Ann, Mike) | 目标谓词 训练样例集合 | Father(David, Mike) |
| | Couple(David, James) | | \neg Father(David, James) |
| | Mother(James, Ann) | | \neg Father(James, Ann) |
| | Mother(James, Mike) | | \neg Father(James, Mike) |
| | | | \neg Father(Ann, Mike) |

给定目标谓词，FOIL算法从实例（正例、反例、背景样例）出发，不断测试所得到推理规则是否还包含反例，一旦不包含负例，则学习结束，展示了“**归纳学习**”能力。

知识图谱推理

- 归纳学习 FOIL (First Order Inductive Learner)



给定目标谓词，FOIL算法从实例（正例、反例、背景知识样例）出发，不断测试所得推理规则是否还包含反例，一旦不包含，则学习结束，由此充分展示了“归纳学习”的能力。在学得推理规则后，再给推理规则中的变量赋予具体例子，经过“演绎”得到新的知识

知识图谱推理

● 路径排序推理算法(PRA)

与FOIL算法不同，路径排序推理算法(PRA)的基本思想是将实体之间的关联路径作为特征，来学习目标关系的分类器。路径排序算法的工作流程主要分为以下三步。

- 1) 特征抽取：生成并选择路径特征集合。生成路径的方式有随机游走(random walk)、广度优先搜索、深度优先搜索等。
- 2) 特征计算：计算每个训练样例的特征值 $P(s \rightarrow t; \pi_j)$ 。该特征值可以表示从实体结点 s 出发，通过关系路径 π_j 到达实体结点 t 的概率；也可以表示为布尔值，表示实体 s 到实体 t 之间是否存在路径 π_j ；还可以是实体 s 和实体 t 之间路径出现的频次、频率等。
- 3) 分类器训练：根据训练样例的特征值，为目标关系训练分类器。当训练好分类器后，即可将该分类器用于推理两个实体之间是否存在目标关系。

知识图谱推理

● 路径排序推理算法(PRA)

例 2.16 对图2.1中的知识图谱采用路径排序算法。

(1) 目标关系: *Father*

(2) 对于目标关系*Father*, 生成四组训练样例, 一个为正例、三个为负例:

正例: (David, Mike)

负例: (David, James), (James, Ann), (James, Mike)

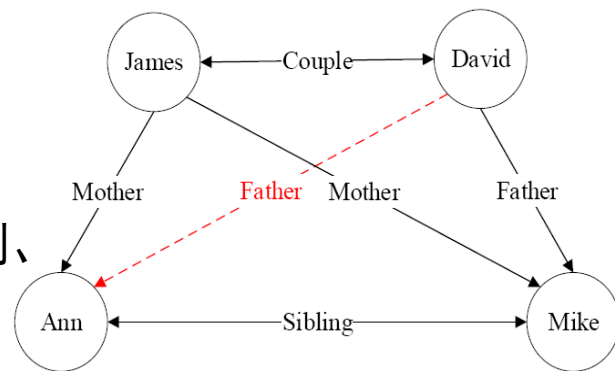
(3) 从知识图谱采样得到路径, 每一路径链接上述每个训练样例中两个实体:

(David, Mike)对应路径: *Couple* \rightarrow *Mother*

(David, James)对应路径: *Father* \rightarrow *Mother*⁻¹ (*Mother*⁻¹与*Mother*为逆向关系)

(James, Ann)对应路径: *Mother* \rightarrow *Sibling*

(James, Mike)对应路径: *Couple* \rightarrow *Father*



知识图谱推理

● 路径排序推理算法(PRA)

(4) 对于每一个正例/负例，判断上述四条路径可否链接其包含的两个实体，将可链接（记为1）和不可链接（记为0）作为特征，于是每一个正例/负例得到一个四维特征向量：

(David, Mike): $\{[1, 0, 0, 0], 1\}$

(David, James): $\{[0, 1, 0, 0], -1\}$

(James, Ann): $\{[0, 0, 1, 0], -1\}$

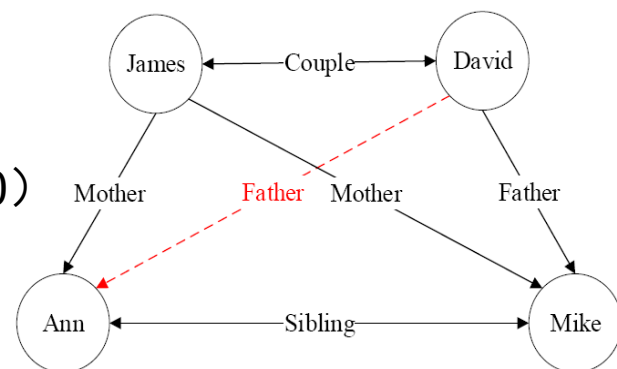
(James, Mike): $\{[0, 0, 1, 1], -1\}$

2 hop 路径粒度(granularity)的词典

(*Mother* → *Sibling* 和 *Couple* → *Father* 两条路径均包含了 James 和 Mike)

(5) 依据(4)中的训练样本，训练分类器 M 。

(6) 预测。对于图2.1中形如(David, Ann)的样例，得到其特征值为 $[1, 0, 0, 0]$ （*Couple* → *Mother* 这一条路径包含了 David 和 Ann），将特征向量输入到分类器 M 中，如果分类器 M 给出分类结果为1，则*Father*(David, Ann)成立。



知识图谱推理

● 基于分布式表示的知识推理

- 随着深度学习应用的进展，在自然语言处理 (Natural Language Processing, NLP) 中，通过自监督学习将一个单词表示成一个隐向量 (latent vector) 的分布式表示方法取得了良好效果。研究者将知识图谱中的节点和节点之间的关系均映射到连续数值空间中 (即分布式表示 distributed representation)，同时对映射过程进行约束，使得知识图谱中节点之间的拓扑结构得以保留，从而有效地对知识图谱进行表示和推理 [Bordes 2013]。
- 给定所有 $\langle left_node, relation, right_node \rangle$ 形式的样例知识，学习分布式表示 (distributed representation) 和度量函数 ψ ，使得 $\psi(left_node, relation, right_node)$ 取值最小。
- 一旦学习得到度量函数 ψ ，就可用其去判断任意两个节点之间是否具有某一关系。

知识图谱推理

● 基于分布式表示的知识推理

➤ TransE及其改进

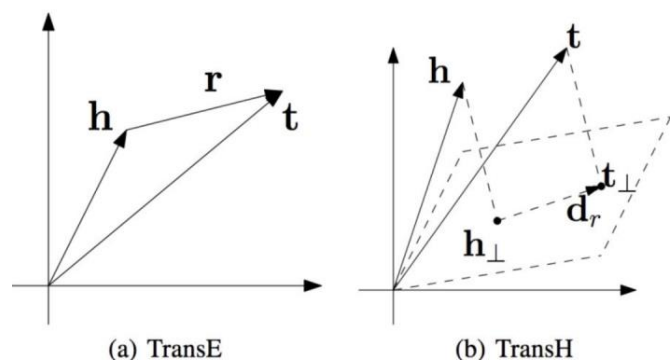


Figure 1: Simple illustration of TransE and TransH.

✓ TransH

处理一对多/多对一/多对多关系，并且不增加模式的复杂性和训练难度。

其基本思想是将关系解释为超平面上的转换操作。每个关系都有两个向量，超平面的法向量 \mathbf{w}_r 和超平面上的平移向量 \mathbf{d}_r 。

$f_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_{L_1/L_2}$ 关系变换是加法操作

$$\mathcal{L} = \sum_{(\mathbf{h}, \mathbf{r}, \mathbf{t}) \in \Delta} \sum_{(\mathbf{h}', \mathbf{r}', \mathbf{t}') \in \Delta} \max(f_r(\mathbf{h}, \mathbf{t}) + \gamma - f_r(\mathbf{h}', \mathbf{t}'), 0)$$

这个模型只能处理一对一的关系，不适合一对多/多对一关系，例如，有两个知识 (skytrees, location, tokyo) 和 (gundam, location, tokyo)。经过训练，实体 'skytrees' 的向量将非常接近实体 'gundam' 的向量。但实际上它们没有这样的相似性。

$$\mathbf{h}_\perp = \mathbf{h} - \mathbf{w}_r^T \mathbf{h} \mathbf{w}_r, \mathbf{t}_\perp = \mathbf{t} - \mathbf{w}_r^T \mathbf{t} \mathbf{w}_r$$

$$f_r(\mathbf{h}, \mathbf{t}) = \|(\mathbf{h} - \mathbf{w}_r^T \mathbf{h} \mathbf{w}_r) + \mathbf{d}_r + (\mathbf{t} - \mathbf{w}_r^T \mathbf{t} \mathbf{w}_r)\|_2^2$$

$$\mathcal{L} = \sum_{(\mathbf{h}, \mathbf{r}, \mathbf{t}) \in \Delta} \sum_{(\mathbf{h}', \mathbf{r}', \mathbf{t}') \in \Delta} \max(f_r(\mathbf{h}, \mathbf{t}) + \gamma - f_r(\mathbf{h}', \mathbf{t}'), 0) + C \left\{ \sum_{\mathbf{e} \in E} [\|\mathbf{e}\|_2^2 - 1]_+ + \sum_{r \in R} \left[\frac{(\mathbf{w}_r^T \mathbf{d}_r)^2}{\|\mathbf{d}_r\|_2^2} - \epsilon^2 \right]_+ \right\}$$

知识图谱推理

◆ 基于分布式表示的知识推理

✓ TransR

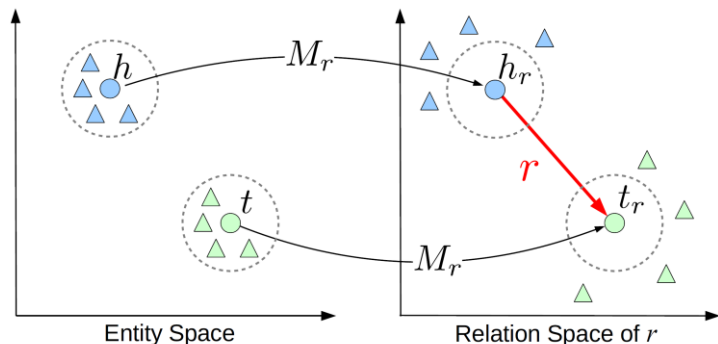


Figure 1: Simple illustration of TransR.

TransE和TransH模型都假设实体和关系是语义空间中的向量，因此相似的实体在同一实体空间中会非常接近。然而，每个实体可以有許多方面，不同的关系关注实体的不同方面。例如: (location, contains, location) 的关系是'contains', (person, born, date) 的关系是'born'。这两种关系非常不同。为了解决这个问题，TransR在关系特定的实体空间中建模实体，并在对应的空间中进行关系转换。

$$\mathbf{h}_r = \mathbf{h}M_r, \mathbf{t}_r = \mathbf{t}M_r$$
$$f_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h}_r + \mathbf{r} - \mathbf{t}_r\|_2^2$$

CTransR (Cluster-based TransR): 仅构建一个关系向量来执行从head到tail实体的所有转换是不够的，可能有不同的模式。例如三元组 (location, contains, location) 具有许多模式: country-city, country-university, continent- country 等。

All entity pairs (h, t) are represented with their vector offsets $(\mathbf{h} - \mathbf{t})$ for clustering, where \mathbf{h} and \mathbf{t} are obtained with TransE.

$$\mathbf{h}_r = \mathbf{h}M_r, \mathbf{t}_r = \mathbf{t}M_r$$
$$f_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h}_r + \mathbf{r}_c - \mathbf{t}_r\|_2^2 + \alpha \|\mathbf{r}_c - \mathbf{r}\|_2^2$$

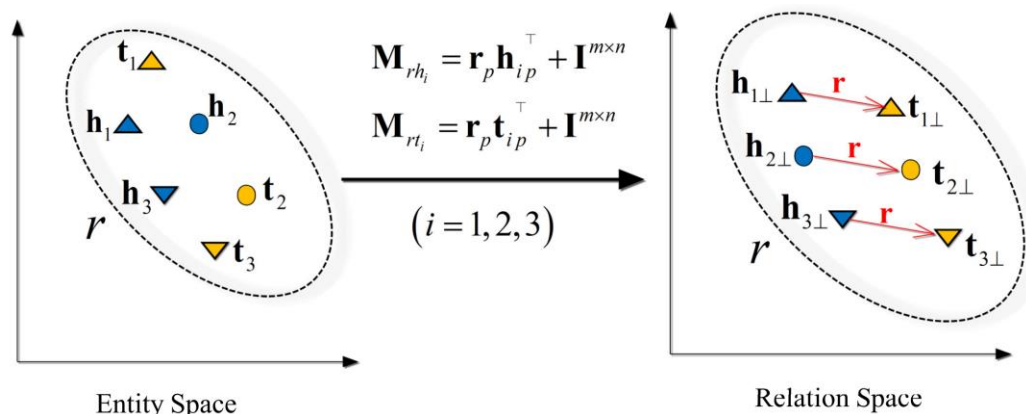
知识图谱推理

◆ 基于分布式表示的知识推理

✓ TransD

TransR的不足之处:

- 1) head和tail使用相同的转换矩阵将自己投射到超平面上, 但是head和tail通常是一个不同的实体, 例如 (Bill Gates, founder, Microsoft), 'Bill Gate'是一个人, 'Microsoft'是一个公司, 这是两个不同的类别。所以他们应该以不同的方式进行转换。
- 2) 这个投影与实体和关系有关, 但投影矩阵仅由关系决定。
- 3) TransR的参数数大于TransE和TransH。由于其复杂性, TransR/CTransR难以应用于大规模知识图谱。



TransD使用两个向量来表示每个实体和关系。第一个向量表示实体或关系的意义, 另一个向量 (称为投影向量) 将用于构造映射矩阵。

$$\mathbf{h}_\perp = M_{rh} \mathbf{h}, \mathbf{t}_\perp = M_{rt} \mathbf{t}$$
$$f_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h}_\perp + \mathbf{r} - \mathbf{t}_\perp\|_2^2$$

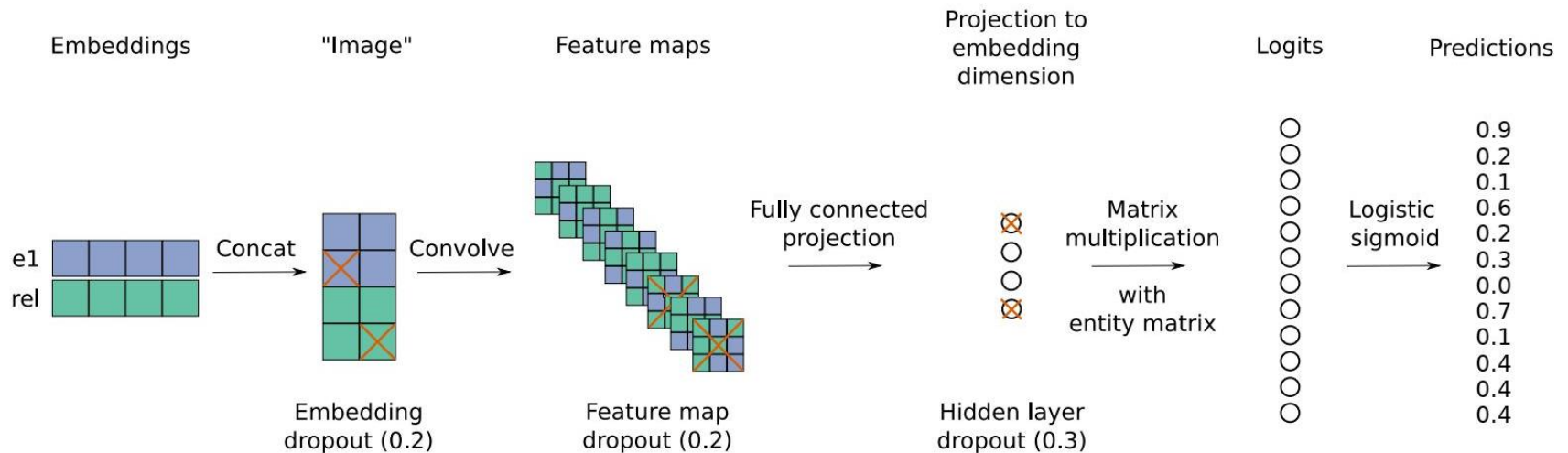
知识图谱推理

◆ 基于分布式表示的知识推理

➤ ConvE

$$f(\text{vec}(f([\bar{h}, \bar{r}] * w))W)t$$

与关系作用是进行非线性映射



- ✓ Rossi, A., Firmani, D., Matinata, A., Merialdo, P., & Barbosa, D. (2021). Knowledge Graph Embedding for Link Prediction: A Comparative Analysis. ACM Trans. Knowl. Discov. Data, 15, 14:1-14:49.

| Family and Group | | Model | Loss | Constraints | Space Complexity |
|-----------------------------|-----------------------|----------|--|---|--|
| Tensor Decomposition Models | Bilinear | DistMult | $\mathbf{h} \times \mathbf{r} \times \mathbf{t}$ | $\forall \mathbf{r} \in \mathcal{R} : \mathbf{r}$ is diagonal; | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d)$ |
| | | ComplEx | $\mathbf{h} \times \mathbf{r} \times \bar{\mathbf{t}}$ | $\mathbf{h} \in \mathbb{C}^d; \mathbf{t} \in \mathbb{C}^d; \mathbf{r} \in \mathbb{C}^{d \times d};$ $\forall \mathbf{r} \in \mathcal{R} : \mathbf{r}$ is diagonal; | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d)$ |
| | | Analogy | $\mathbf{h} \times \mathbf{r} \times \mathbf{t}$ | $\forall \mathbf{r} \in \mathcal{R} : \mathbf{r} \times \mathbf{r}^T = \mathbf{r}^T \times \mathbf{r};$ $\forall (r_1, r_2) \in \mathcal{R} \times \mathcal{R} : \mathbf{r}_1 \times \mathbf{r}_2 = \mathbf{r}_2 \times \mathbf{r}_1;$ | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d)$ |
| | | Simple | $\frac{1}{2}(\mathbf{h}_h \times \mathbf{r} \times \mathbf{t}_t) + \frac{1}{2}(\mathbf{h}_t \times \mathbf{r}_{-1} \times \mathbf{t}_h)$ | $\forall \mathbf{r} \in \mathcal{R} : \mathbf{r}, \mathbf{r}_{-1}$ are diagonal; | $\mathcal{O}(2 \mathcal{E} d + 2 \mathcal{R} d)$ |
| | Non-bilinear | HolE | $(\mathbf{h} \star \mathbf{t}) \times \mathbf{r}$ | | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d)$ |
| | | TuckER | $\mathbf{W} \times_1 \mathbf{h} \times_2 \mathbf{r} \times_3 \mathbf{t}$ | | $\mathcal{O}(\mathcal{E} d_e + \mathcal{R} d_r + d_e d_r d_e)$ |
| Geometric Models | Pure Translation | TransE | $\ \mathbf{h} + \mathbf{r} - \mathbf{t} \ $ | | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d)$ |
| | Additional Embeddings | STransE | $\ \mathbf{W}_r^h \times \mathbf{h} + \mathbf{r} - \mathbf{W}_r^t \times \mathbf{t} \ $ | | $\mathcal{O}(\mathcal{E} d + \mathcal{R} (d + 2d^2))$ |
| | | CrossE | $\sigma(\tanh(\mathbf{h} \odot \mathbf{c}_r + \mathbf{r} \odot \mathbf{h} \odot \mathbf{c}_r) \times \mathbf{t}^T)$ | | $\mathcal{O}(\mathcal{E} d + 2 \mathcal{R} d)$ |
| | Roto-translation | TorusE | $\min_{(x, y) \in ([h]+[r]) \times [t]} \ \mathbf{x} - \mathbf{y} \ _i$ | | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d)$ |
| | | RotatE | $-\ \mathbf{h} \odot \mathbf{r} - \mathbf{t} \ $ | $\mathbf{h} \in \mathbb{C}^d; \mathbf{r} \in \mathbb{C}^d; \mathbf{t} \in \mathbb{C}^d;$ $\forall \mathbf{r}_i \in \mathbf{r} : \mathbf{r}_i = 1;$ | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d)$ |
| Deep Learning Models | Convolution | ConvE | $g(\mathbf{W} \times g([\mathbf{h}; \mathbf{r}] \odot \boldsymbol{\omega}) + \mathbf{b}) \times \mathbf{t}$ | | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d + Tmn + Td(2d_m - m + 1)(d_n - n + 1))$ |
| | | ConvKB | $g(\mathbf{W} \times g([\mathbf{h}; \mathbf{r}; \mathbf{t}] \odot \boldsymbol{\omega}) + \mathbf{b})$ | | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d + 4T)$ |
| | | ConvR | $g(\mathbf{W} \times g([\mathbf{h}] \odot \boldsymbol{\omega}_r) + \mathbf{b}) \times \mathbf{t}$ | | $\mathcal{O}(\mathcal{E} d_e + \mathcal{R} d_r + Tmn + Td_e(2d_{e_m} - m + 1)(d_{e_n} - n + 1))$ |
| | Capsule | CapsE | $\ \text{capsnet}(g([\mathbf{h}; \mathbf{r}; \mathbf{t}] \odot \boldsymbol{\omega})) \ $ | | $\mathcal{O}(\mathcal{E} d + \mathcal{R} d + 3T + Td)$ |
| | Recurrent | RSN | $\sigma(\text{rsn}(\mathbf{h}, \mathbf{r}) \times \mathbf{t})$ | | $\mathcal{O}(2 \mathcal{E} d + 2 \mathcal{R} d + Lknd)$ |

Table 1. Loss Function, constraints and space complexity for the models included in our analysis.

知识表示方法

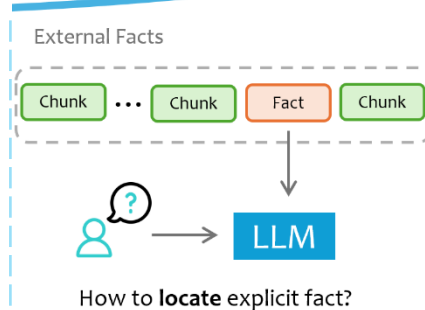
- 谓词 (predicate) 逻辑
- 产生式 (production) 表示法
- 框架 (frame) 表示法
- 语义网络 (Semantic Network)
- Large Language Models (LLMs)
 - LLM 中的 FNN 神经元 事实性知识
 - LLM 完成某项任务的 circuits
- Multimodal Large Language Models (MLLMs)

Retrieval Augmented Generation

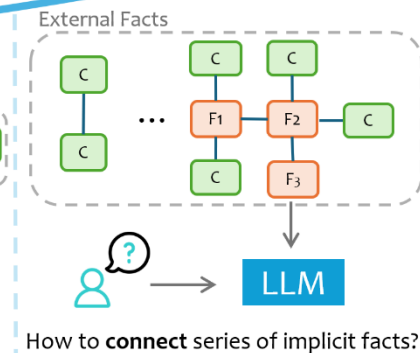
Retrieval Augmented Generation (RAG)

- ✓ Zhao, Siyun, Yuqing Yang, Zilong Wang, Zhiyuan He, Luna Qiu and Lili Qiu. "Retrieval Augmented Generation (RAG) and Beyond: A Comprehensive Survey on How to Make your LLMs use External Data More Wisely." ArXiv abs/2409.14924 (2024): n. pag.

Level 1: Explicit Fact Queries

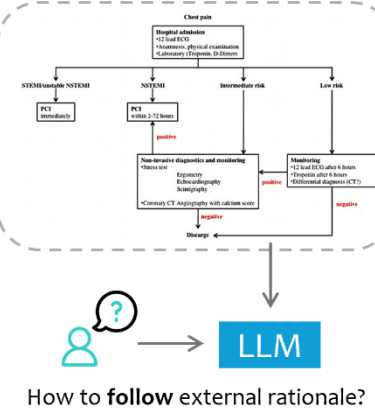


Level 2: Implicit Fact Queries

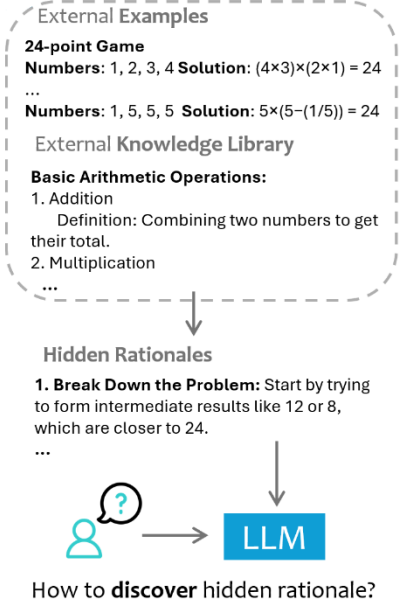


Level 3: Interpretable Rationale Queries

External Rationales



Level 4: Hidden Rationales Queries



Level 0:
Bare LLM

Chain of Thoughts

RAG

Iterative RAG

Prompt Tuning

Offline Learning

1.1 Multi-modal Documents Parsing

1.2 Chunking Optimization

2.1 Indexing

2.2 Query Document Alignment

2.3 Re-ranking and Correction

2.4 Recursive Retrieval or Iterative Retrieval

3. Response Generation Enhancement

RAG on Graph / Tree

Text-to-SQL

CoT Prompting

In Context Learning

Model Fine Tuning

Thinking system 1 and system 2

- ✓ Kahneman, Daniel. Thinking, Fast and Slow (2011). Farrar, Straus and Giroux. ISBN 9780374275631.

SYSTEM 1 Fast Thinking

- **Intuitive**
- Often **unconscious**
- **fast & efficient**
- Based on available knowledge, past experience, long established **mental models**
- More susceptible to **cognitive biases**

- ① 作为系统2的组件
- ② 作为平行通路生成初始结果



- ① 精炼初始结果
- ② 反思，自我矫正
- ③ 参与训练平行系统1
- ④ 参与训练组件系统1

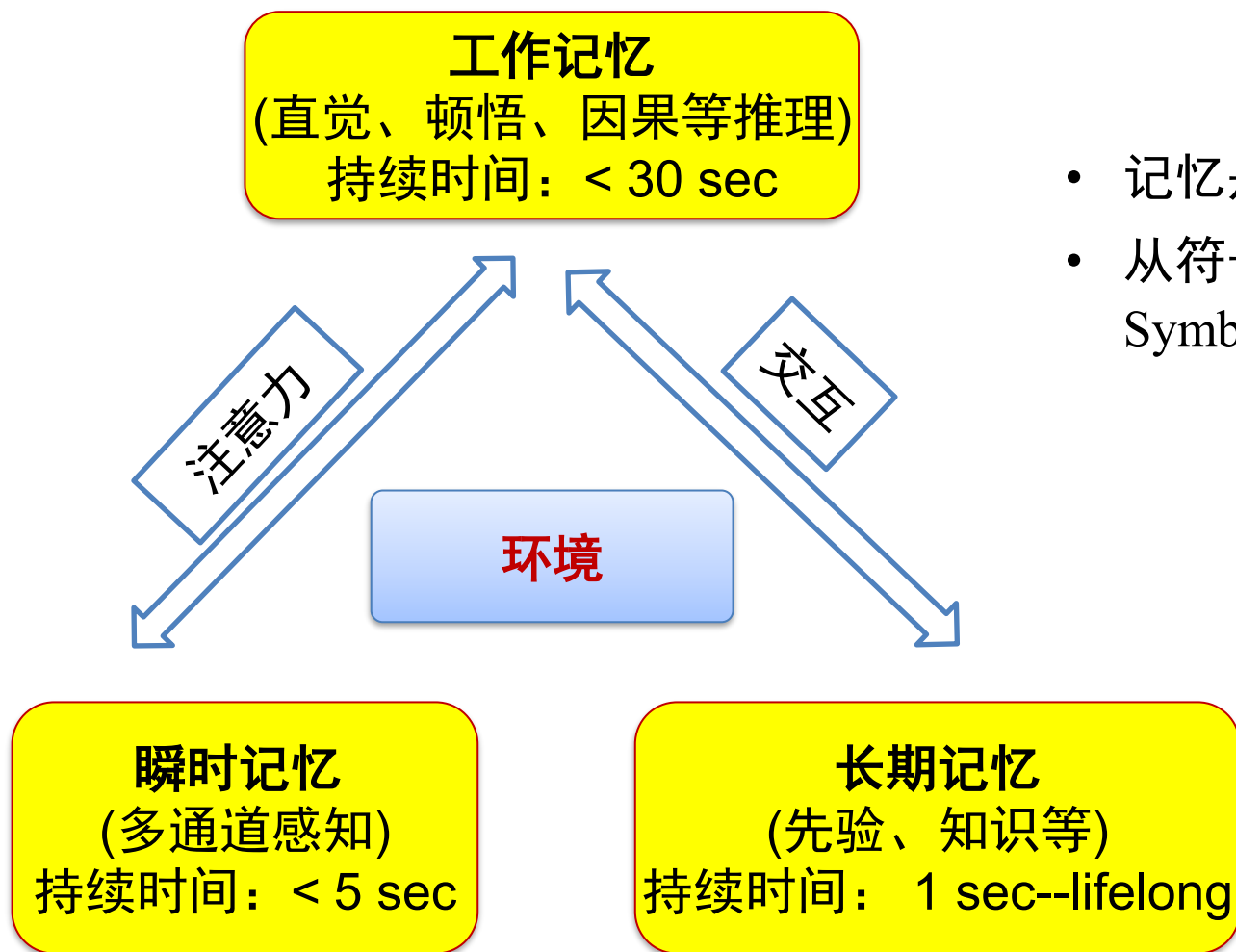
SYSTEM 2 Slow Thinking

- **Analytic**
- Deliberate **conscious reasoning**
- **Slow**
- Indicative of **critical thinking**, structured analytic techniques
- Used with **qualitative and quantitative methods**

丹尼尔·卡尼曼（Daniel Kahneman, 1934.3.5-2024.3.27）以色列裔美国心理学家，以其在判断和决策心理学以及行为经济学方面的工作而闻名，因此与 Vernon L. Smith 共同获得 2002 年诺贝尔经济学奖。卡尼曼发表的实证研究结果挑战了现代经济理论中盛行的人类理性假设，被称为“行为经济学之父”。

基于知识的智能体

逻辑与推理是人工智能的核心问题



- 记忆是逻辑推理的重要部分。
- 从符号主义推理到 Neural-Symbolic 结合的推理