

# 什么是专家系统

## 1. 专家系统定义

专家系统是模拟人类专家解决某一类具体问题的人工智能系统，如疾病诊疗、机械设计等。那么，如何让计算机模拟人类专家呢？思路很简单：想办法将专家掌握的知识抽取出来，利用这些知识，计算机就可以象专家一样工作了。

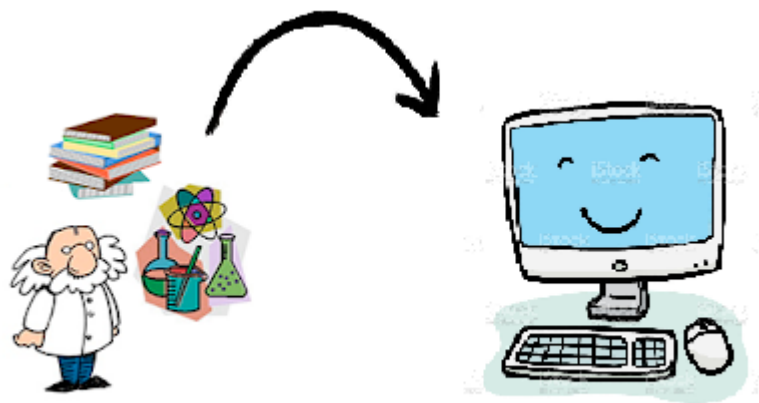


图 1：专家系统利用人类专家的知识完成工作

## 2. 如何实现专家系统

实现一个专家系统需要解决两个主要问题：一是如何表示知识，二是如何利用知识解决问题[1,2]，我们逐个说明。

首先是知识表示的问题，即如何将专家的知识梳理出来，并表示成计算机能读懂的结构。知识表示有很多种方法，最简单的是写成“如果...就...”这样的判断句，称为“产生式规则”。更形式化一点，一条知识可以表示成如下结构：

**IF <前提> THEN <结论>**

这条知识意思是当<前提>被满足时，可以得到<结论>。例如：

**IF 吃坏东西 THEN 拉肚子**

如果条件和结论包含更复杂的情况，则可以表示更复杂的知识，如：

**IF 发烧 AND 打喷嚏 THEN 感冒**

这条知识表示如果高烧并且还打喷嚏，那么病人应该是感冒了。我们把专家掌握的所有

知识按这种格式总结出来，就构造出了一个**领域知识库**。

**第二个问题涉及到推理方法**，即如何利用既有知识解决问题。还是以诊疗系统为例，如果病人的表现是打喷嚏和发烧，那么该怎么治呢？人类的医生会基于打喷嚏和发烧这些表现，判断病人可能是感冒了，而治疗感冒的常用药物是阿司匹林，因此医生会告诉病人：嗯，吃几片阿斯匹林吧。这个过程就是人类的推理过程。计算机医生会模仿这个过程，首先将病人发烧和打喷嚏作为前提在知识库中查找，发现了如下匹配知识：

**“IF 发烧 AND 打喷嚏 THEN 感冒”**

基于该知识推论出病人是感冒了。得到这一结论后，计算机再以“感冒”为前提在知识库中继续查找，发现如下知识：

**“IF 感冒 THEN 开阿司匹林”**

基于这一知识，计算机医生知道了要给病人开阿司匹林。

上述顺序搜索过程即是一个简单的推理过程。在实际应用场景中，推理要复杂的多，特别是要处理现实场景中的非确定性问题（见下节）。

### 3. 专家系统的典型结构

解决了知识表示和推理方法这两个核心问题，就可以构造一个专家系统了。一个典型专家系统的基本结构如图 2 所示。总体来说，**知识工程师**将专家知识提取出来并保存到**知识库**中；**推理机**基于这些知识进行推理，以处理用户的输入请求；**解释器**对推理结果给出解释。推理的中间结果会暂时存放在**动态数据库**中，以提高推理效率。

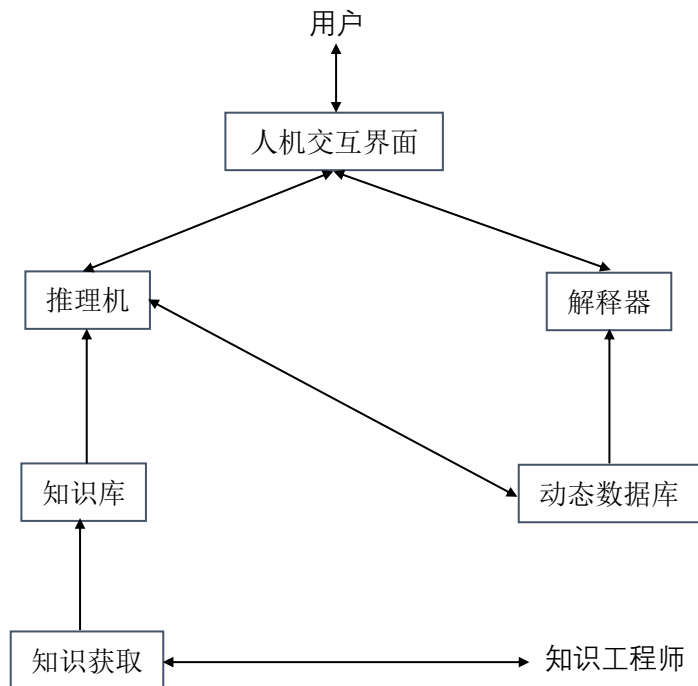


图 2. 专家系统的基本结构

在图 2 所示的基础结构中，推理机是引擎，负责解析知识库中的知识，并基于这些知识进行推理。在实际应用中，推理机除了执行上一节所述的基础推理算法，很重要的一点是处理现实场景中的非确定性。比如：“如果阴天，那么会下雨”，这一表述中就充满了非确定性：是有些云彩就算阴天呢，还是乌云滚滚算阴天？即便是乌云滚滚也不一定就下雨吧，只是天阴的越厉害，下雨的可能性就越大而已。随机性、模糊性和不完全性均可导致非确定性。要处理非确定性问题，不仅需要在定义事实和规则时把非确定性考虑在内，在进行推理时也要定义合理的操作，包括逻辑运算、规则运算、规则合成等。因此，实际应用中的专家系统一般执行的是非确定性推理。

解释器是另一个专家系统独有的“特色模块”，这一模块负责向用户解释系统得到的答案是怎么来的。解释一般分为 Why 解释和 How 解释两种，Why 解释回答“为什么有这样的答案”，How 解释回答“如何得到这样的答案”。例如，在一次诊疗过程中，系统给出让病人验血的建议，如果病人想知道为什么让自己去验血，可以通过交互接口输入 Why，系统会根据推理过程给出让病人验血的原因，让用户明白验血的意义。再比如，如果系统诊断病人患有肺炎，而病人想了解这个结论是如何得到的，他可以通过交互接口输入 How，系统就会解释做出肺炎诊断所依据的临床证据。解释器给出的原因和依据不仅使用户更加信服，也极大提高了系统在实际应用中的安全性。

#### 4. 专家系统开发工具

专家系统的一个设计原则是知识库与系统的其他部分互相分离，其中知识库和具体领域密切相关，而推理机和其他模块则独立于具体领域，具有通用性。基于这一原则，可以设计一些开发工具，用于快速构造专家系统。

**骨架型工具：**借助之前开发好的专家系统，将描述领域知识的部分“挖掉”，只保留推理机等通用部分，这样得到的开发工具称为骨架型工具，因为它保留了原有系统的主要框架。最早的专家系统工具 EMYCIN (Empty MYCIN) 就是一个典型的骨架型专家系统工具，从名称就可以看出，它来源于著名的专家系统 MYCIN。骨架型专家系统工具使用起来简单方便，只需将领域知识按照规定的格式表示出来，就可以构造一个新的专家系统，从而提高专家系

统的构建效率。

**语言型工具:** 骨架型工具灵活性不够,除了知识库以外,使用者不能改变其他任何东西。语言型工具允许用户构建新的推理机,使用起来更加灵活方便。著名的 OPS5 就是这样的工具系统,它以产生式系统为基础,综合了通用的控制和表示机制,为用户提供建立专家系统所需要的基本功能。这样做的好处是系统构建更加灵活方便,但也增加了构建的难度。尽管如此,使用语言型工具还是比基于通用计算机语言从头进行系统设计要简便的多。

## 5. 专家系统的局限性

专家系统取得了很大成功,同时也存在一些局限性,这些局限性影响了专家系统的推广使用。

**知识获取瓶颈:** 知识是专家的经验总结,而专家通常是非常稀有的,愿意配合的专家更少。即便专家愿意配合,知识总结也不是那么容易的事。举一个简单的例子,很多人都会骑自行车,但如果有人问:如何可以骑车不倒呢?估计能总结出答案的人也不多。

**知识库覆盖问题:** 知识库总是有限的,它不可能包含所有知识。人类的智能体现在可以从有限的知识中学习模式和特征,就是我们常说的“规则是死的,人是活的”。与之相反,专家系统奉行的是“规则是死的,计算机也是死的”,它只能基于已有知识进行推理,无法自动扩展出新知识。因此,在知识库涵盖的范围内,专家系统可能表现的很好,但当出现哪怕一点点偏离,系统的性能就可能急剧下降甚至不能求解,体现出系统的脆弱性。

**知识冲突问题:** 专家知识是经验总结,发生冲突的情况并不少见,特别是当考虑多个专家的意见时,不同专家给出的知识互相矛盾就更正常了。即便是同一个专家所总结的知识也可能是互相矛盾的,只不过专家自己都可能没有意识到。研究者提出了一些解决知识冲突的方法,但这些方法无法改变知识本身固有的不一致性 (inconsistency) 。

## 6. 总结

专家系统产生于 20 世纪 60 年代,当时人工智能研究的一个主流思想是基于一些通用的推理方法来解决广泛领域的实际问题,就象定理证明那样,给出一些基础假设,利用一些通用的推理算法,即可以构建一个完整的知识体系(如几何学)。事实证明,现实问题并不是一些基础假设上的完美推理,想要通过一套推理系统把所有问题都解决是不可能的。

通用 AI 的失败给学者们带来反思。1965 年,费根鲍姆等人开始构造一种新的 AI 体系,在

这一体系中，推理不再基于一些基础假设，而是基于知识。和基础假设相比，这些知识和问题的相关性更直接，因此推理更容易；特别是，这些知识提供了真实场景下的复杂性，从而使后续推理更符合实际。知识天然专业的，是特定领域的，因此这一以知识为核心的新的AI体系被称为专家系统。

半个多世纪以来，人们建造了数千个专家系统，很多系统的性能甚至超过了人类专家。近年来，随着技术的进步和数据的积累，新一代专家系统更加强大，不仅有更强的推理能力，也更加注重学习[3,4,5]。基于学习的专家系统避免了传统专家系统知识加工上的困难，但也带来了可解释性上的问题，丢掉了传统基于知识的专家系统最宝贵的东西。将知识与数据结合起来，有可能构造更强大、更有建设性的专家系统，服务人类。

[1]Hayes-Roth F, Waterman D, Lenat D. Building expert systems[J]. 1984.

[2]Waterman, D. A guide to expert systems. United States: N. p., 1986. Web.

[3]Van Pham H., Son L.H., Tuan L.M. (2020) A Proposal of Expert System Using Deep Learning Neural Networks and Fuzzy Rules for Diagnosing Heart Disease. In: Satapathy S., Bhateja V., Nguyen B., Nguyen N., Le DN. (eds) Frontiers in Intelligent Computing: Theory and Applications. Advances in Intelligent Systems and Computing, vol 1013. Springer, Singapore.

[4]Gu W, Foster K, Shang J, et al. A game-predicting expert system using big data and machine learning[J]. Expert Systems with Applications, 2019, 130: 293-305.

[5]Osowski S, Hoai L T, Markiewicz T. Support vector machine-based expert system for reliable heartbeat recognition[J]. IEEE transactions on biomedical engineering, 2004, 51(4): 582-589.