智能化设计研究报告

**中国石油大学**

**2018年7月20日**

目 录

[1 人工智能 1](#_Toc519801805)

[1.1 概念和研究领域 1](#_Toc519801806)

[1.2 发展历史和发展趋势 1](#_Toc519801807)

[1.3 学派分类 2](#_Toc519801808)

[1.3.1 经验主义（机器学习） 2](#_Toc519801809)

[1.3.1.1 人工智能、机器学习、深度学习之间的关系 3](#_Toc519801810)

[1.3.1.2 机器学习与人类思考的类比 4](#_Toc519801811)

[1.3.1.3 机器学习的应用范围 4](#_Toc519801812)

[1.3.1.4 机器学习的方法分类 6](#_Toc519801813)

[1.3.1.5 机器学习的过程 7](#_Toc519801814)

[1.3.2 连接主义（神经网络、深度学习） 7](#_Toc519801815)

[1.3.3 符号主义（逻辑和知识） 9](#_Toc519801816)

[1.3.3.1 基于案例的推理 10](#_Toc519801817)

[1.3.3.2 基于规则的推理 17](#_Toc519801818)

[1.4 应用领域和局限性 18](#_Toc519801819)

[1.4.1 机器学习和深度学习 19](#_Toc519801820)

[1.4.2 逻辑推理 21](#_Toc519801821)

[1.4.3 有限的人工智能 25](#_Toc519801822)

[2 大数据 26](#_Toc519801823)

[2.1 方法 26](#_Toc519801824)

[2.1.1 无监督学习 27](#_Toc519801825)

[2.1.2 有监督学习 28](#_Toc519801826)

[2.1.2.1 y是连续值的回归问题 29](#_Toc519801827)

[2.1.2.2 y值为类别的分类问题 29](#_Toc519801828)

[2.1.3 其他数据预处理算法 32](#_Toc519801829)

[2.2 思维 32](#_Toc519801830)

[2.3 前提要求 32](#_Toc519801831)

[2.4 应用领域和局限性 33](#_Toc519801832)

[3 化工工程设计 33](#_Toc519801833)

[3.1 工艺包设计 33](#_Toc519801834)

[3.2 工程详细设计 34](#_Toc519801835)

[4 智能化设计方案 38](#_Toc519801836)

[4.1 实现路径和目标 38](#_Toc519801837)

[4.2 智能化层次 39](#_Toc519801838)

[4.3 可行的方法 40](#_Toc519801839)

[5 总结 40](#_Toc519801840)

**我有一个好想法，就缺一个AI了！**

人工智能因深度学习带来的种种突破，引起了各个领域的高度关注。化工工程设计领域如何实现智能化设计是一个令人感兴趣的问题。

广泛调研得到的结果是，不仅是化工工程设计，广泛意义上的设计领域也鲜有人工智能技术涉足其中。幸运的是，可行的智能化方案最终还是存在的，以下章节就描述了方案的探索领域和过程。

# 1 人工智能

## 1.1 概念和研究领域

人工智能（Artificial Intelligence），英文缩写为AI。它是研究、开发用于模拟、延伸和扩展人的智能的理论、方法、技术及应用系统的一门新的技术科学。

人工智能是一门边缘学科，属于自然科学和社会科学的交叉，但学科基础是计算机科学，或者说人工智能是为了解决计算机问题而诞生的。研究领域包括：知识表示、自动推理和搜索方法、机器学习和知识获取、知识处理系统、自然语言理解、计算机视觉、智能机器人、自动程序设计等方面。

## 1.2 发展历史和发展趋势

人工智能的发源是1956年在达特茅斯学院。达特茅斯学院在美国东北部一个小城市，那在一年他们办了一个暑期班，其中有西蒙、麦卡锡（John McCarthy）、明斯基（Marvin Minsky）等人。大家都有这么一个想法，可以用计算机来模拟人的思考。花了大概一个多月的时间，在达特茅斯学院这个地方一起讨论，最后就形成了人工智能这个学科。形成这个学科之后，大家非常兴奋，以前从来没有想过机器可以模仿人的智能，所以当这一件事情已经发生了之后，大家就觉得好像我们对于用机器来解决我们现在面临的各种复杂的问题，已经有答案了。明斯基说：“十到十五年之内我们可以解决所有的智能问题”。

到目前为止，人工智能经历了三次浪潮，如图1-1所示。

1956-1976是第一次浪潮。达特茅斯会议后人们陆续发明了第一款感知神经网络软件和聊天软件，证明了数学定理，人类惊呼“人工智能来了”、“再过十年机器人会超越人类”。这个时期的标志是：符号主义盛行，在统计方法中引入符号方法进行语义处理，人机交互开始成为可能。

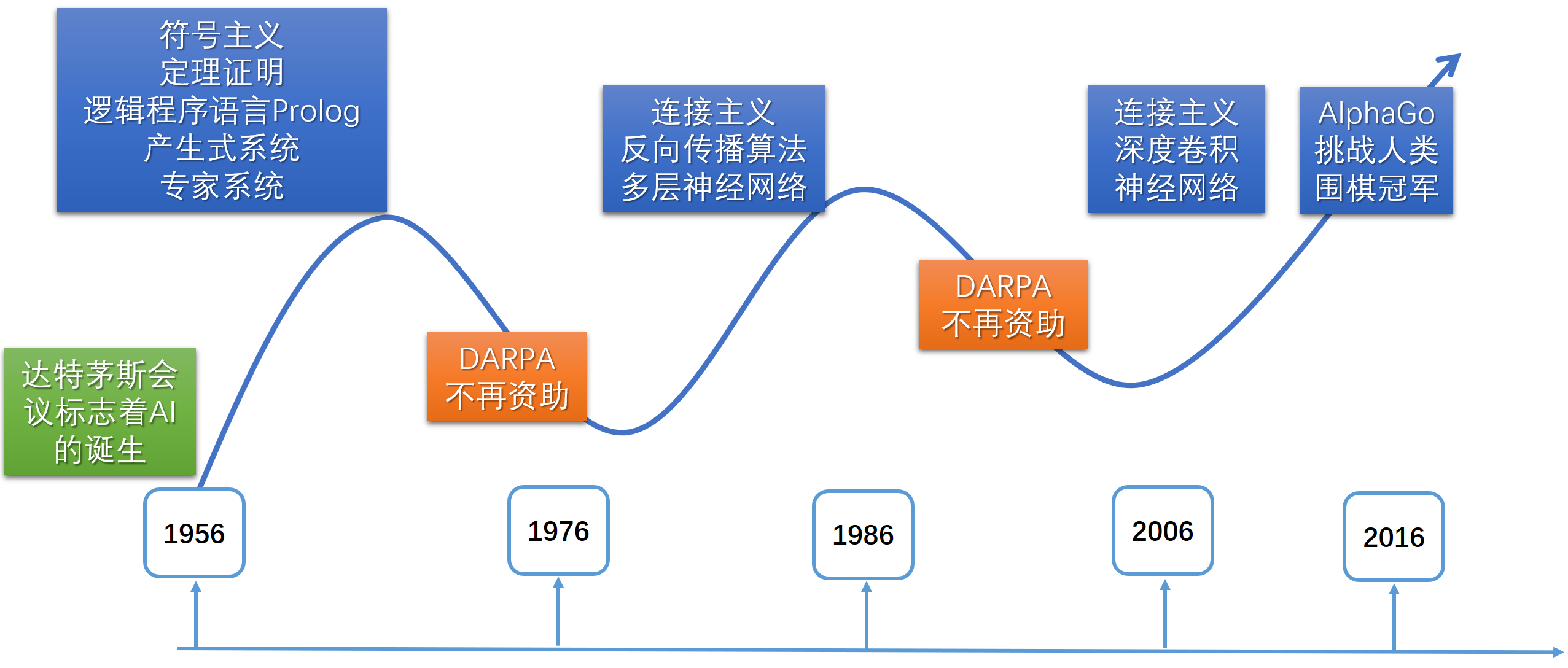


图1-1 人工智能的三次浪潮

1976-2006是第二次浪潮。80年代Hopfield神经网络和BT训练算法的提出，使得人工智能再次兴起，出现了语音识别、语音翻译计划，以及日本提出的第五代计算机。但这些设想迟迟未能进入人们的生活之中，第二次浪潮又破灭了。这个时期的标志是：连接主义持续多年，后来发现神经元网络可解决单一问题，但无法解决复杂问题，积累一定的数据量，有些结果在一定程度就不再上升。

2006年-至今是第三次浪潮。随着2006年Hinton提出的深度学习技术，以及2012年ImageNet竞赛在图像识别领域带来的突破，人工智能再次爆发。这一次，不仅在技术上频频取得突破，在商业市场同样炙手可热，创业公司层出不穷，投资者竞相追逐，创投界一片欣欣向荣，民间一片忧心重重。这个时期的标志是：海量的数据、不断提升的算法能力和计算机运算能力。

在这三次浪潮中，各个学派起起伏伏，各领风骚。

## 1.3 学派分类

人工智能有三个最基本的学派：经验主义（机器学习）、联结主义（神经网络、深度学习）和符号主义（逻辑和知识）。还有一个行为主义学派主要研究机器人，跟我们的研究主题相关性不大，因此不再赘述。

### 1.3.1 经验主义（机器学习）

第一个学派我们称为**经验主义**，最经典的代表就是**机器学习**。机器学习可能有一万个不同的变种，我们把它抽象到最抽象的角度，其实就是图1-2讲的东西，它就是在学习一个函数。函数的输入就是图中“010101”这些数字，图像、声音、文本都是数据。通过中间的算法，去逼近这个世界上存在的一个函数，这就是机器学习。什么样的东西是函数呢？比如人脸，人脸识别出来这是女生，这是男生，这就是一个函数，这个函数的输出是0或者1，这是所谓的分类问题。当然现实中遇到的分类问题，会比这复杂很多，可能不是分两类，可能是分1000个类。

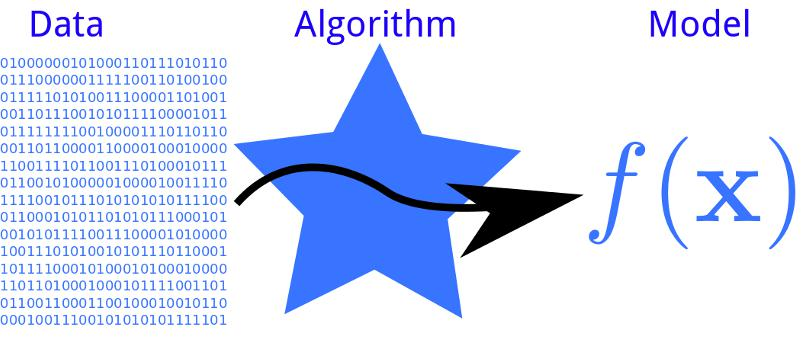


图1-2 机器学习的抽象

聚类问题。如果把屋子里边的人聚成两类，一类坐在这边，一类是坐在那边的，实际上就是所谓的一个分布，这就是另外一种类型的函数。所以机器学习就是，有了一堆输入的数据，声音、图像、文本都行，把它变成一个数学上可以精准描述的，通过这个数据本身可以去预测的一种东西。通常这种东西就是概率，一个概率分布，这是机器学习目前这个阶段最常用的方法。

所以，经验主义又通常把它称为**统计主义**，机器学习是最典型代表。

#### 1.3.1.1 人工智能、机器学习、深度学习之间的关系

目前业界最常见的划分是，机器学习是人工智能的一个重要分支，可以在大数据的支撑下通过各种算法让机器对数据进行深层次的统计分析以进行“自学”；深度学习则是机器学习算法的一种，隶属于人工神经网络体系，现在很多应用领域中性能最佳的机器学习都是基于模仿人类大脑结构的神经网络设计而来的，相比较其他方法，在解决更复杂的问题上表现更优异，深度学习是可以帮助机器实现“独立思考”的一种方式。总体来说，人工智能涵盖了机器学习和深度学习的所有范畴，人工智能研究的主要目标是使机器能够胜任一些通常需要人类才能完成的复杂工作，人工智能、机器学习、深度学习三者之间是逐层包含的关系,如图1-3所示。



图1-3 人工智能、机器学习、深度学习三者之间的关系

#### 1.3.1.2 机器学习与人类思考的类比

从广义上来说，机器学习是一种能够赋予机器学习的能力，以此让它完成直接编程无法完成的功能。但从实践的意义上来说，机器学习是一种通过利用数据训练出模型，然后使用模型预测的一种方法。国外有些学者对机器学习进行了定义，有学者认为，机器学习是对能通过经验自动改进的计算机算法进行的研究；也有学者认为，机器学习是利用数据或以往的经验，以此优化计算机程序的性能标准。不管是以上哪一种，我们都可以认为机器学习是利用经验或数据来改进算法的研究，通过算法让机器从大量历史数据中学习和寻找规律，得到某种模型并利用此模型预测未来，如图1-4所示。机器在学习的过程中，处理的数据越多，预测就会越精准。



图1-4 机器学习与人类思考的类比

#### 1.3.1.3 机器学习的应用范围

从范围上来说，机器学习跟模式识别，统计学习，数据挖掘是类似的，同时，机器学习与其他领域的处理技术的结合，形成了计算机视觉、语音识别、自然语言处理等交叉学科，如图1-5所示。



图1-5 机器学习与其他领域的关系

（1）模式识别

模式识别=机器学习。两者的主要区别在于前者是从工业界发展起来的概念，后者则主要源自计算机学科。在著名的《Pattern Recognition And Machine Learning》这本书中，Christopher M. Bishop在开头是这样说的：“模式识别源自工业界，而机器学习来自于计算机学科。不过，它们中的活动可以被视为同一个领域的两个方面，同时在过去的十年间，它们都有了长足的发展”。

（2）数据挖掘

数据挖掘=机器学习+数据库。数据挖掘仅仅是一种方式，但不是所有的数据都具有价值，所以数据挖掘思维方式才是关键，加上对数据深刻的认识，这样才可能从数据中导出模式指引业务的改善。大部分数据挖掘中的算法是机器学习的算法在数据库中的优化。

（3）统计学习

统计学习近似等于机器学习。统计学习是个与机器学习高度重叠的学科。因为机器学中的大多数方法来自统计学，甚至可以认为，统计学的发展促进机器学习的繁荣昌盛。例如著名的支持向量机算法，就是源自统计学科。两者的区别在于：统计学习者重点关注的是统计模型的发展与优化，偏数学；而机器学习者更关注的是能够解决问题，偏实践，因此机器学习研究者会重点研究学习算法在计算机上执行的效率与准确性的提升。

（4）计算机视觉

计算机视觉=图像处理+机器学习。图像处理技术用于将图像处理为适合进入机器学模型中的输入，机器学习则负责从图像中识别出相关的模式。计算机视觉相关的应用非常的多，例如百度识图、手写字符识别、车牌识别等等应用。这个领域将是未来研究的热门方向。随着机器学习的新领域深度学习的发展，大大促进了计算机图像识别的效果，因此未来计算机视觉界的发展前景不可估量。

（5）语音识别

语音识别=语音处理+机器学习。语音识别就是音频处理技术与机器学习的结合。语音识别技术一般不会单独使用，一般会结合自然语言处理的相关技术。目前的相关应用有苹果的语音助手Siri等。

（6）自然语言处理

自然语言处理=文本处理+机器学习。自然语言处理技术主要是让机器理解人类的语言的一门领域。在自然语言处理技术中，大量使用了编译原理相关的技术，例如词法分析，语法分析等等，除此之外，在理解这个层面，则使用了语义理解，机器学习等技术。作为唯一由人类自身创造的符号，自然语言处理一直是机器学习界不断研究的方向。

#### 1.3.1.4 机器学习的方法分类

机器学习方法通常可以分为三种类型，方法体系如图1-6所示。



图1-6 机器学习方法体系

（1）有监督学习（Supervised Learning）：给学习算法提供的训练数据是有标记的，对于每一个输入，学习者都被提供了一个回应的目标。监督学习主要被应用于快速高效地教会人工智能掌握现有的知识。常见的算法有决策树、Adaboost算法、人工神经网络算法、SVM(即支持向量机)算法、朴素贝叶斯、K-近邻、逻辑回归、随机森林算法。

（2）无监督学习（Unsupervised Learning）：给学习算法提供的训练数据是未标记的，并且要求算法识别输入数据中的模式，它先建立一个模型，再用其试着对输入的数据进行解释，并用于下次输入。在现实世界中,大部分数据是不带标签的, 所以无监督学习要比监督学习应用更广泛。常见的算法分为两大类：聚类算法和降维算法。最有代表性的算法有K-means算法和主成分分析算法（PCA算法）。

（3）强化学习（Reinforcement Learning）：是一种在线的、无导师机器学习方法，该方法与动态环境相互作用，把环境的反馈作为输入，通过学习选择能达到其目标的最优动作。强化学习这一方法背后的数学原理与监督/非监督学习略有差异。监督/非监督学习更多地应用了统计学，而强化学习更多地结合了离散数学、随机过程这些数学方法。常见的算法有：TD(λ)算法、Q\_learning算法、Sarsa算法。

常见算法在2.1节中有更具体的介绍。

#### 1.3.1.5 机器学习的过程

所谓机器学习过程，是指观察有n个样本数据组成的集合，并根据这些数据来预测未知数据的性质，那么在给定数据集（所谓大数据）和具体问题的前提下，一般解决问题的步骤可以概括如下：

（1）获取数据；

（2）分析预处理数据；

（3）训练模型：再使用训练数据来构建使用相关特征的模型；

（4）验证模型：使用测试数据检查被验证的模型的性能表现；

（5）使用模型：使用完全训练好的模型在新数据上做预测；

（6）调优模型：使用更多数据、不同的特征或调整过的参数来提升算法的性能表现。

### 1.3.2 连接主义（神经网络、深度学习）

第二个学派就是现在最火的东西——**深度学习**。最初把它称为神经网络。图1-3是最常见的一种神经网络。实际上，神经网络是很简单的。神经网络就是一堆矩阵。输入了一堆数字，经过了一个矩阵的变化之后，变成了另外一堆数字；如果输入是一个向量，经过一个矩阵变成另外一个向量，实际上这就是神经网络的一个最基础的数学模型。

用图来表现就是图1-7的形式，比如那些连线就代表一个矩阵。为什么把它称为神经网络呢？因为它来源于人脑的神经元模型。神经元之间也有很多突触，相互关联在一起，所以一个神经的脉冲可以从一个神经元传递到另外一个神经元，不断地去进行传导。

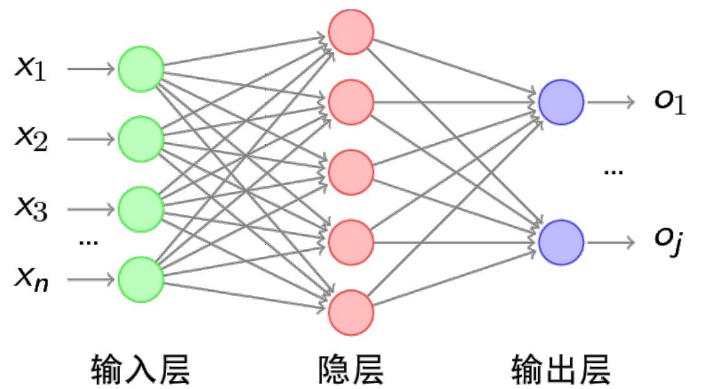


图1-7 典型神经网络

图1-7的神经网络有三层，叫输入层、输出层，中间的叫隐层。在80年代的时候，就曾经证明了像这样的一个神经网络模型，是世界上最强大的计算模型。图灵机是现在最强大的计算模型，而这样一个非常简单的神经网络模型跟图灵机是等价的，数学上可以严格地证明。

发现了这个等价性之后，人们认为，既然一层二层三层的隐层都是等价的，那只用一层就好了。但是后来发现遇到了很多问题，就是速度很慢，还有是泛化能力不好。速度和泛化的问题，神经网络一直都解决不了，所以到了90年代被人说神经网络是骗子。直到2006年Geoffrey Hinton提出可以构造多层人工神经网络模型，并采用采用逐层训练方法解决训练达到最优的问题。在计算能力的提升、算法的提升和大数据的出现的基础上，诞生了这个深度网络，效果真的很好，这就是现在的深度学习。所以深度学习也就是一种神经网络。

深度学习与传统模式识别方法的最大不同在于它是从大数据中自动学习特征，而非采用手工设计的特征。好的特征可以极大提高模式识别系统的性能。在过去几十年模式识别的各种应用中，手工设计的特征处于同统治地位。它主要依靠设计者的先验知识，很难利用大数据的优势。由于依赖手工调参数，特征的设计中只允许出现少量的参数。深度学习可以从大数据中自动学习特征的表示，其中可以包含成千上万的参数。手工设计出有效的特征是一个相当漫长的过程。回顾计算机视觉发展的历史，往往需要五到十年才能出现一个受到广泛认可的好的特征。而深度学习可以针对新的应用从训练数据中很快学习得到新的有效的特征表示。

深度学习有很多模型，常见的模型和适用问题如下：

* DBN: 通用的分类问题
* RNN与LSTM: 序列化学习，时间序列
* CNN: 图像、音频、文本分类
* RBM: 特征提取
* Autoencoder: 特征提取

### 1.3.3 符号主义（逻辑和知识）

第三个学派叫做符号主义，也叫理性主义。在不同的历史时期，它有不同的说法。在四十年代、五十年代叫它逻辑，后来叫它知识，再后来叫它语义网，这两年叫它知识图谱。其实它都是一个学派传下来的东西，实际就是**逻辑推理**：人必有一死，苏格拉底是人，所以苏格拉底必死。就是这种大前提、小前提、结论，三段论。

在2000年前，亚里士多德就已经提出三段论了，所以逻辑并不是新的东西。在过去一百多年时间里，有很多逻辑学家，从布尔到罗素最后到图灵，研究基于逻辑的通用计算模型，最终导致了计算机的产生。后来有人想，我们人是怎么思考的，能不能用逻辑模拟。

有一个斯坦福大学的计算机科学家麦卡锡认为，这个世界上的真理就是**人的思考是基于一种符号的形式来表现的**。只要把整个世界全部用符号系统来进行承载的时候，就可以表达整个世界了。以这个思路往前走，50年代的时候就想也许十年或者二十年之后，所有人类的智力问题全部可以用这种逻辑方法来解决。在70年代的时候，确实取得了很多的成功。当时有一个逻辑分支叫定理证明，逻辑程序证明了《几何原理》里所有的定理。当时那种冲击跟今天Alpha Go那种冲击是一样的，觉得人工智能真的要毁灭世界了的感觉。但后来证明并没有，因为很快发现逻辑有它自己的问题。

从符号主义的观点来看，知识是信息的一种形式,是构成智能的基础，知识表示、知识推理、知识运用是人工智能的核心（如图1-8所示），知识可用符号表示,认知就是符号的处理过程，推理就是采用启发式知识及启发式搜索对问题求解的过程，而推理过程又可以用某种形式化的语言来描述，因而有可能建立起基于知识的人类智能和机器智能的同一理论体系。



图1-8 符号主义的知识体系

符号主义的代表成果是1957年纽威尔和西蒙等人研制的成为“逻辑理论家”的数学定理证明程序LT。LT的成功，说明了可以用计算机来研究人的思维过程，模拟人的智能活动。以后，符号主义走过了一条启发式算法——专家系统——知识工程的发展道路，尤其是专家系统的成功开发与应用，使人工智能研究取得了突破性的进展。

符号主义主张用逻辑方法来建立人工智能的统一理论体系，但却遇到了“常识”问题的障碍，以及不确知事物的知识表示和问题求解等难题，因此，受到其他学派的批评与否定。

符号主义学派两大技术问题和主流方法下节介绍。

#### 1.3.3.1 基于案例的推理

化工设计的特点是弱理论、强经验，没有很强的理论模型、领域知识不完全，但经验知识丰富。

传统的专家系统一般采用基于规则的推理（Rule-based reasoning，简称RBR），在知识和规则的获取上遇到难以克服的困难，停滞不前。受人类认知过程的启示，另一种求解问题的推理模式——基于案例的推理（Case-based reasoning，简称CBR）就被提出来了。

CBR，也可以翻译为案例推理、实例推理、范例推理或事例推理，是由Schank Ableson教授与1977年提出的，其认知科学基础是：相似问题具有相似解。CBR类似于人类解决问题的过程，相对于RBR而言，具有以下显著的优势：

* 案例是以前问题的优化解，蕴含着难以规则化的设计经验与知识；
* 案例获取比规则获取要容易，大大缓解规则知识获取的瓶颈问题；
* 重用求解结果，提高求解效率，近似解决不熟悉的问题；
* 推理学习能力持续不断提高——不断添加案例，增量式学习。

基于案例的推理的知识体系如图1-9所示。



图1-9 基于案例的推理的知识体系

CBR作为人工智能领域中融合类比推理与机器学习的一种解决问题的方法，在各个领域得到了广泛应用，例如：

* 基于案例学习的内燃机油产品设计系统EOFDS
* 基于案例推理的天气预报系统
* 基于案例推理的轧钢规程系统
* 基于案例推理的淮河王家坝洪水预报调度系统FOREZ
* 东海渔业资源评估专家系统Fisher（2002年国家科技进步二等奖）
* 基于案例推理的股价操纵行为判别系统
* 基于案例推理的供应商选择决策支持系统
* 基于案例推理的应急决策支持系统
* 基于案例推理的城市典型灾害应急处置专家系统
* 基于案例推理的竞争情报系统
* 基于案例推理的故障诊断系统
* 基于案例推理的造林专家系统
* 基于案例推理的供水调度系统
* 基于案例推理的高铁信号系统车载设备故障诊断
* 基于案例推理的高炉热风炉燃烧专家控制系统
* 基于案例推理的城市轨道交通应急系统
* 基于案例推理的铁矿粉烧结配料系统
* 基于案例推理的工程造价估算系统
* 基于案例推理的智能化地图制图
* 大型舰船舱室布局设计专家系统
* 基于案例推理的产品包装设计系统
* 基于案例汽车焊装夹具智能设计系统
* 基于案例推理的煤矿工作面设计系统
* 基于案例的智能数控工艺设计系统
* 基于案例推理的水平定向钻机动力头快速设计
* 基于实例推理的抓斗三维设计系统
* 基于实例推理的强夯设计辅助系统
* 基于实例推理的减速器参数化变型设计
* 基于实例推理的枪械方案快速设计
* 基于实例推理的毛衫快速变型设计
* 基于实例推理的转向架设计
* 基于事例推理的模具辅助设计系统研究
* 基于事例推理的车身主断面设计
* 基于CBR/RBR集成推理的注塑模架智能化设计系统
* 基于CBR技术的立体仓库货架设计系统
* 基于实例推理的余热锅炉模块化设计系统
* 基于实例推理的振动料斗智能选型系统

CBR的基本任务包括案例表示、案例库构建、案例检索、案例复用、案例修改、案例学习，工作流程如图1-10所示。

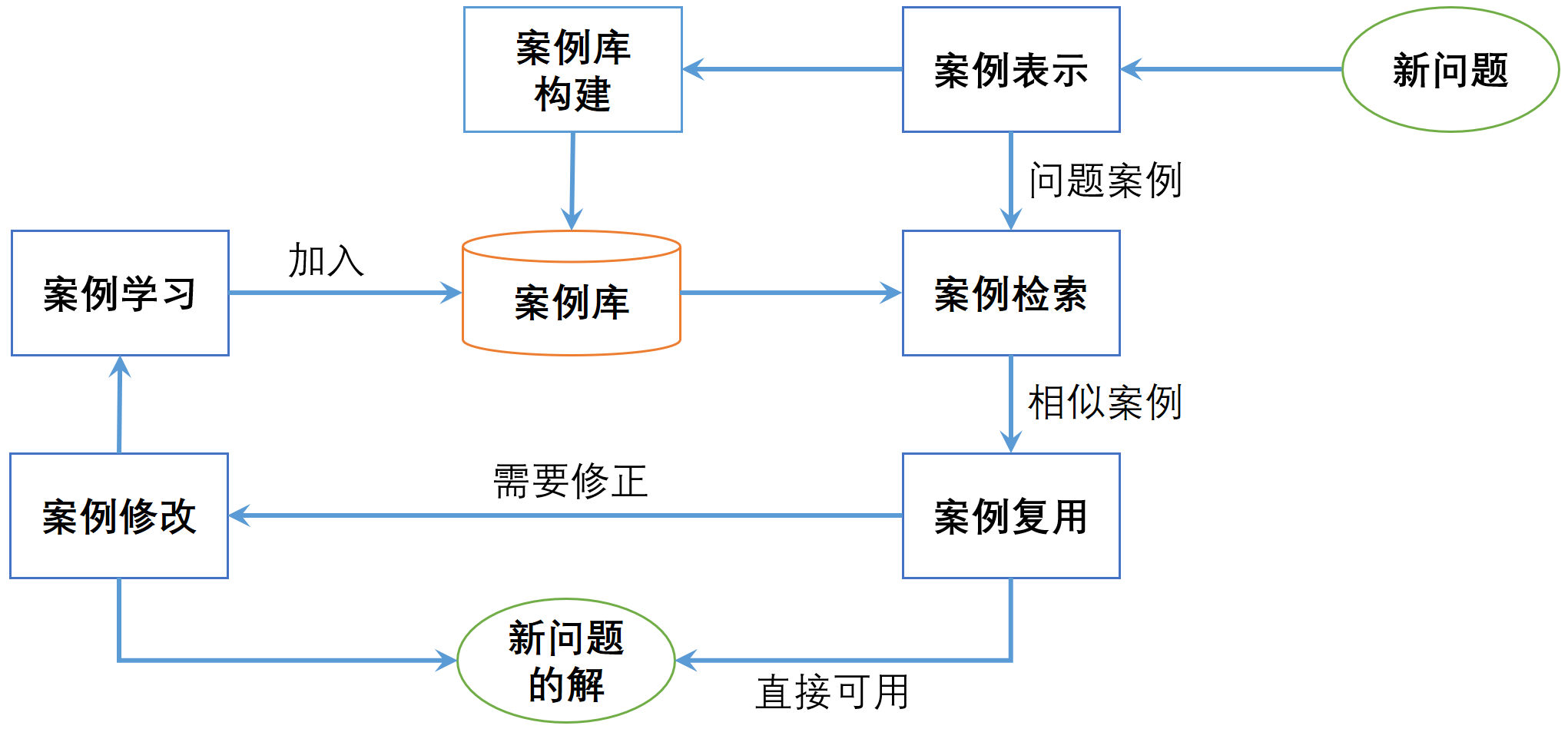


图1-10 基于案例的推理工作流程

**（1）案例表示**

案例表示是**知识符号化**的过程，以便把人类知识表示成计算机能处理的知识结构，可以视为数据结构及其处理机制的综合：案例表示=数据结构+处理机制。**案例表示是CBR的第一个关键技术，它的合理与否直接影响案例检索、案例复用、案例修正、案例库维护的难易程度。**案例表示主要包括：案例的表示内容、案例的表示方法和案例的存储。

**1）案例的表示内容**

一个案例是一个具体问题求解的完整表达，是对具体问题以及相关的环境、状态、解决方案的描述，一般包括：问题的描述、解的描述和结果的描述。

**1、问题的描述**

在进行问题描述的时候，首先应当分析搜集到的大量的源案例，抽取其主要的共同特征，根据这些共同的特征来进行案例描述。

特征的提取可以采用技术手段自动提取，也可以人机互动提取。

问题的描述，包括对初始条件、约束条件、背景和设计目标的描述。它是问题的说明信息，也是进行推理的求解条件。

可以考虑划分为主特征和副特征两种特征层次来进行问题的描述。主特征可以用于案例检索，主特征+副特征可以作为结果的描述。

**2、解的描述**

解的描述根据具体问题而定，例如设计问题，可以考虑将设计规则、设计经验和修改记录描述为解，在案例检索结果中提供给用户参考，当然也可以帮助新设计师学习设计的过程。

**3、结果的描述**

结果的描述可以考虑是主特征+副特征的内容。

**2）案例的表示方法**

**案例采用什么方法描述，取决于问题类型和求解方案。**在具体的基于案例推理的系统中，往往是在分析问题领域的基础上通过选择、综合或修改现有的各种知识表示方法来实现案例的表示。

问题表示成案例的形式，称为**问题案例，可以采用上述的主特征，用于案例检索**。

常见的案例表示方法有：

* 框架表示法：现在用的少
  + 框架的表示格式：
  + 框架名：<>
  + 槽名1：<>
  + 侧面11< >(值111，值112，⋯值llkl)
* 面向对象表示法：编程、存储较为方便
* 一阶谓词逻辑表示法：方便规则推理
* 语义网络表示法：增加语义信息
* 过程表示法：侧重过程描述
* 产生式规则表示法：可以利用众多语法分析方法，方便推理
  + - 基本表示形式为： IF<前提>THEN<动作>或<结论>

如果采用深度学习技术进行学习，则需要考虑将案例表示为向量形式。不过，对于文本、图形、图像、视频等非结构化问题如何描述，仍无法解决。而在这些领域，并不存在有效、通用、统一的描述解决方法，因此需要针对这些问题提出相应的解决方案，例如考虑制定编码规则，将非结构化表达转为结构化表达。

**3）案例的存储方法**

案例的存储跟案例库构建紧密相关。通常考虑层次化分类组织，在数据表中关联分散存储。

另外，树形的JSON数据格式符合人的认知，满足了结构化和非结构化混合表示的需要，兼容现有的工程实现，也是可以考虑的方法。

**（2）案例检索**

案例检索实质上是问题案例的特征匹配过程，包括案例索引、特征权重确定、相似度度量、检索策略相似度计算和案例检索等关键环节：

**1）案例索引**

* 主特征和副特征，主特征用于检索
* 索引标识编码

**2）特征权重确定**

* 层次分析法AHP
* 群组决策AHP法

**3）相似度度量**

* 欧氏距离
* 海明距离
* Kolodner距离
  + 数值型
  + 数值、区间数值、模糊值三者之间的相似度
  + 布尔型、枚举型、字符串型、文本型

**4）检索策略**

目标：检索出的案例尽可能少；尽可能与当前问题相关或相似

思路：分级检索

* 串行
  + 层次法
  + 决策树
* 并行
  + K-近邻
  + 知识引导法

**5）相似度计算**

基于上述的相似度距离模型计算案例之间的相似度

**6）案例检索**

* 数据库技术-模板法
  + 精确SQL查询
  + 模糊SQL查询
  + 全文检索
* K-近邻算法
* K-D树
* 决策树-归纳法
* 知识导引法

**（3）案例复用**

如果检索到的案例可以直接使用，则称之为案例复用或者案例重用。如果案例修改存在困难，可以到这一步就结束推理过程。

**（4）案例修改**

由于检索到的最相似案例一般与问题案例不可能完全相同，这时对最相似案例进行相应的修改是必需的，一般要考虑采用基于规则的推理方法协助案例的修改。

根据参考案例的数量，修改类型分为两种：

* 单案例修改
  + 空修改
  + 转换式修改
  + 替换式修改
  + 结构化修改
  + 诱导式修改
* 组合案例修改
  + 问题分解
  + 子问题组合

常见的用于案例修改的方法有：

* 规则推理
* 约束满足
* 归纳学习
* 模糊逻辑

**（5）案例学习**

案例学习是体现CBR方法增量式学习的关键。案例学习在工程上可以指案例库维护，基本思路是：去除无益的案例，限制案例库规模的大小，既保持性能，也保持功能。

* 基于效用度量的案例库维护
  + 效用与应用频率、平均时间和匹配代价有关
  + 基于能力度量的案例库维护
* 根据覆盖度和可达度分类
  + 删除或筛选
  + 不删除案例

#### 1.3.3.2 基于规则的推理

基于规则推理的专家系统是一个将大量的相关领域的专家知识及经验进行提取总结并根据一定的形式储存，用来解决那些需要领域专家处理的问题。基于规则推理的专家系统通过模拟领域专家在解决问题时进行推理、回顾和判断等过程来达到实现解决实际问题的目的。因此基于规则的专家系统就是模拟人类理性思维的一种计算机程序，而且系统在处理问题时具有简单高效等特点。基于规则推理的专家系统主要包括五大部分：知识库、推理机制、人机接口、解释程序、数据库。其中核心部分是知识库和推理引擎。

**（1）规则表示**

规则表示是研究如何用最合适的形式来组织规则知识，是对待解决的问题最有利。用于规则表示的方法应该具有充分表达、有效推理、便于管理、易于理解等性能。目前常见的表示方法有：框架表示法、面向对象表示法、一阶谓词逻辑表示法、语义网络表示法、过程表示法、产生式规则表示法等等。

产生式规则表示法也称启发式规则表示法，由产生式规则构成的系统成为产生式系统，这是一个很自然的问题求解系统。产生式规则的表示形式为：**IF<条件1>，AND<条件2>，AND<条件3>，…THEN<动作>或<结论>**，如果一条规则的条件被满足，则这条规则的动作或结论就被启用。产生式规则表示形式中的条件1, 2,…n是条件表达式，各种形式的条件表达式都可以应用。结论根据知识的不同而具有不同的现实指导意义。不同的知识的结论需要不同的解释程序来解释。

产生式规则具有以下优点：

自然性：产生式规则以“if-then”的形式表示，接近人的自然推理方式，易于理解；

灵活性：产生式规则可以自由增、删、修改，便于加入启发性知识、学习和解释；

通用性：产生式规则可以表示各种知识和和推理的过程；

可分解性：产生式规则可以把复杂问题分解为子问题，最后综合来解决问题。

**虽然产生式规则也存在效率低、表达能力不强等缺点，但是其“if-then”的表示形式足以表示设计选型中的规则性知识。**

**（2）推理引擎**

如果不考虑使用规则引擎执行规则推理，则每一条规则知识其实就可以写成代码片段，跟其他常见的编程方法没什么区别，但这条规则就已经写死到程序里了（hard code），难以修改。因此对于规则推理，一般需要规则推理引擎去执行，而不是写死到程序中。

考虑到设计选型规则并不是很多，因此对于推理引擎的选择，有三种思路：

**一是自行开发羽量级的推理引擎**。自行开发的技术难度和复杂度不会超过企业自动化办公系统（OA）中的工作流引擎：

* 可以使用数据库的**存储过程**，按模块来组织代码，提供修改界面即可。当然缺点是会增加一点数据库服务器的负担；
* 也可以**代码实现**，常用的编程语言都可以实现，不存在技术难度。

二是使用轻量级的脚本引擎，例如BeanShell、JEXL、Groovy等，功能和性能比自行开发的好一些、规范些，适用于规则不是很多的情况。

三是使用重量级的开源规则引擎，例如Rate、Drools等，功能强大、一致性、冲突检查等完整，适用于存在大量规则的情况。

## 1.4 应用领域和局限性

目前的人工智能技术并不是很可靠，还存在不少局限性。明确方法的局限性对于应用领域问题是必要的。

### 1.4.1 机器学习和深度学习

从1.3节的描述中，我们已经知道机器学习和深度学习其实是一类相似的方法，深度学习是机器学习的一种，只不过是连接主义学派的代表而已。深度学习与机器学习最大的不同在于特征提取上，机器学习需要人工构造特征，这就是所谓的特征工程。深度学习可以自动提取特征，省下不少人工。但两者能做的事情是差不多的，都是统计学的拟合函数能做的事情。

纵观历史上机器学习和深度学习能够做到的事情，包括对数值、图像、视频、语音、文字的分类、识别和预测，并在此基础上搭建出上层的应用，例如字符识别（OCR）、语音识别、人脸识别、自动摘要、视频行为分析等等，我们可以把这些应用统称为“识别系统”。从某种意义上说，这些应用扩展了人作为动物的那一部分感知功能，换句话说，动物也具备类似的感知能力，现在流行的说法叫“智能感知”。

当然，识别系统还是很有实用价值的。OCR是pdf文档内容识别的关键之一，很有用；手机上的语音输入法也很方便；人脸识别对于公安机关显然意义重大；视频行为识别是安防领域竞争的热点。

然而很多人因此夸口，说我们可以用同样的方法（机器学习、深度学习），实现“人类级别的智能”，取代所有的人类工作，这就是神话了。

识别系统跟真正理解语言的“人类智能”，其实相去非常远。说白了，这些识别系统，也就是统计学的拟合函数能做的事情。比如OCR和语音识别，就是输入像素或者音频，输出单词文本。OCR和语音识别系统，虽然能依靠统计的方法，“识别”出你说的是哪些字，它却不能真正“理解”你在说什么。

“识别”和“理解”的差别，就像程序语言里面“语法”和“语义”的差别。程序语言的文本，首先要经过词法分析器（lexer）和语法分析器（parser），才能送进解释器（interpreter），只有解释器才能实现程序的语义。类比一下，自然语言的语音识别系统，其实只相当于程序语言的词法分析器（lexer）。词法分析和语法分析，只不过是实现一个语言的万里长征的“第0步”。大部分的AI系统里面连语法分析器（parser）都没有，所以主谓宾、句子结构都分析不清楚，更不要说理解其中的含义了。

举一个很简单的例子。如果我对Siri说：“我想看一些猫的照片。”它会给出图1-11的回答：“我在网上没有找到与‘一些猫’有关的资料。”

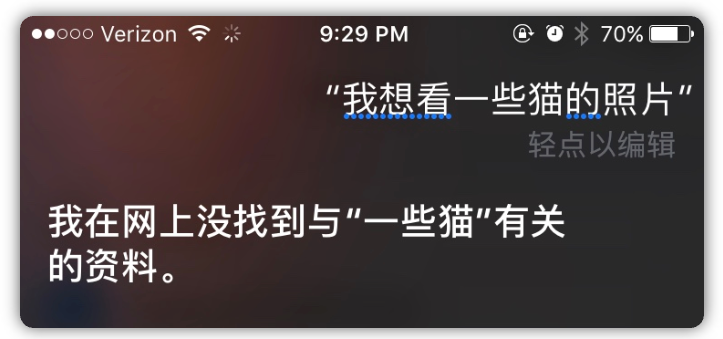


图1-11 Siri问答示例

Siri里面没有语法分析器，甚至它的n-gram语言模型居然连“一些”和“猫”都没分开，这就是为什么它去找“一些猫”，而不是“猫”。它甚至把“照片”这么重要的词都忽略了。所以 Siri 虽然正确的进行了“语音识别”，知道我说了那些字。但由于没有语法分析器parser，没有语法树，它不可能正确的理解我到底在说什么，它甚至不知道我在说“关于什么”。

实现自然语言的语法分析器parser有多难？很难，非常难，想不到的困难。你不仅要深刻理解编程语言的语法分析理论（LL、LR、LALR、GLR……），还需要依靠大量的例子和数据，才能解开人类语言里的各种歧义。经过了语法分析，得到一棵“语法树”，才能传给人脑里语言的理解中心（类似程序语言的“解释器”）。解释器“执行”这个句子，为相关的名字找到对应的“值”，进行计算，才能得到句子的含义。至于人脑如何为句子里的词汇赋予“意义”，如何把这些意义组合在一起，形成“思维”，这个问题还没有准确的回答。至少，这需要大量的实际经验，这些经验是一个人从生下来就开始积累的。机器完全不具备这些经验，我们也不知道如何才能让它获得经验。我们甚至不知道这些经验在人脑里面是什么样的结构，如何组织的。所以机器要真的理解一个句子，真是跟登天一样难。

这就是为什么Douglas Hofstadter（《哥德尔、艾舍尔、巴赫：集异璧之大成》的作者，中文名：侯世达）说：“一个机器要能理解人说的话，它必须要有腿，能够走路，去观察世界，获得它需要的经验。它必须能够跟人一起生活，体验他们的生活和故事……”最后你发现，制造这样一个机器，比养个小孩困难太多了。

总之，一方面，大数据的兴起带来了机器学习和深度学习新一轮的繁荣。更多的数据可以让识别系统变得更好，深度学习技术渗透到了具有大数据的各个领域，但能从数据中得到的终有极限。现在可供渗透的领域已基本殆尽，收益曲线慢慢变平。另一方面，机器学习和深度学习，能产生实用价值，跟统计密切相关，但跟人的“思维”无关，并不具备“思维”的能力。

### 1.4.2 逻辑推理

1.3.3节已经描述了符号主义学派将知识表示为符号，逻辑推理则物化为计算机的计算过程。所以，符号主义方法实现的两大难点是知识表示和逻辑推理。相对而言，知识表示更难，尤其是那些人都无法表达的经验性知识。

人类智能的两大特点在于“理解”和“思考”。

1.4.1节机器学习和深度学习已经能较好地识别物体了，可以说现在处于感知智能阶段。如果把“理解”视为深度感知，也就是语义理解，那么机器在“理解”上仍存在重重困难，可以说目前仍处于深度感知智能的初级阶段。

符号主义学派试图解决“思考”的问题。模仿人类逻辑推理的过程，将规则知识符号化，机器推理（计算）出结论，将逻辑推理转化为计算问题。最初的成功应用是机器定理证明。定理是一种接近形式化语言（只含有语法结构，不含语义，这叫形式）的描述，可以比较容易地进行符号化，因此机器能成功地实现四色定理、几何定理和《数学原理》定理的证明。

符号主义的另一代表应用是专家系统。我们看看被看作是成功的专家系统是什么情况，有什么特点。

第一个专家系统是DENDRAL。DENDRAL输入的是质谱仪的数据，输出是给定物质的化学结构。把化学分析知识提炼成规则，这个专家系统**有时**做得比人还准，取得了一定的成功，但这个系统并不是商用产品。实验科学与理论科学比较，是相对原始的，原始经验也相对容易转换成规则，或者说是从专家采集来的数据做机器学习形成规则。

广为人知的专家系统MYCIN，是一个针对细菌感染的诊断系统。MYCIN的处方准确率是69%，当时专科医生的准确率是80%。MYCIN从来没被临床使用过。

第一个据说能赚钱的专家系统室DEC的专家配置系统XCON。XCON是一种进行计算机系统配置的专家系统。它运用计算机系统配置的知识，依据用户的定货，选出最合适的系统部件，如中央处理器的型号，操作系统的种类及与系统相应的型号，存储器和外部设备以及电缆的型号。XCON是用产生式规则表达知识，采用正向推理的控制结构，用OPSS语言写成，系统的规则从最初的750条扩展到3000多条，被认为是一个成功的专家系统。

20世纪80年代初到20世纪90年代初，专家系统经历了十年的黄金期，随着日本五代机的幻灭，“专家系统”变成了一个不仅不时髦，反而有负面含义的词。互联网催生的电子商务，有很多和XCON类似的应用场景，于是新瓶装旧酒，专家系统摇身一变，改名**规则引擎**，成为中间件的标配。

附：日本人的第五代计算机：这种机器将以Prolog为机器的语言，其应用程序将达到知识表达级，具有听觉、视觉甚至味觉功能，能够听懂人说话，自己也能说话，能认识不同的物体，看懂图形和文字。人们不再需要为它编写程序指令，只需要口述命令，它自动推理并完成工作任务。五代机计划最终目标是组装1000台要素信息处理器来实现并行处理，解题和推理速度达到每秒10亿次；与此相连接的是容量高达10亿信息组的数据库和知识库，包括1万个日语和外国语言的基本符号，以及语法规则2000条，可以分析95％以上的文章，自然语言识别率达到95％。此外，还将配置语音识别装置和储存10万个图像的模式识别装置等等。

专家系统从诞生到没落，受机器定理证明成功的影响，关注的重点一直是语法层面的逻辑推理，而不是语义推理。只有语法不含语义，那就谈不上“理解”，走上了一条纯粹的语法推理之路，离人所能理解的“思考”渐行渐远。

1998年，万维网之父之一伯纳斯.李（Tim Berners-Lee）提出“语义网”（Semantic Web）的概念，用本体承载语义，以本体推理实现语义推理。这条路线后来发展为链接数据（Linked Data）、知识图谱以及最新的名词：智能数据（Smart Data），一脉相承。现在最流行的是知识图谱，最新的马甲是智能数据。知识图谱有成功的商业产品吗？并不好找。现在谈语义推理的成功，为时尚早。

为什么实现“思考”这么难？不管是基于规则的语法推理，还是基于本体的语义推理，构建领域知识推理系统，都存在目前难以克服的瓶颈问题：

**（1）知识获取困难**

领域专家开展工作用到的规范性知识一般是以自然语言描述的，相对严谨，可以利用各种知识表示技术进行表示。

领域专家在实践中总结形成的经验性知识，有些是可以描述的，那就描述出来，由知识工程师加工成规则；有些是难确切的表达的，例如那些只在特定场景中出现的经验，最笨的办法是知识工程师跟随领域专家，在实践中总结出规则；还有些是只能意会不能言传的经验，知识工程师也没办法，只能放弃。

**（2）知识难以穷尽**

如果领域问题是相对封闭的，知识的范围有明确的界限，例如上述的化学分析知识、计算机系统配置知识，知识的获取和规则的表示还是可以处理的。

实践中，领域知识看似是有界限的，实际上却是开放性的，例如领域知识通常不会把前、后、左、由、上、下、外、里、山、水、日、月、重力作用等等常识性知识的解释写到规范中，默认人是具备的。但要让机器推理，就需要让机器懂得这些常识。举一个机器端茶倒水的例子吧。

让一个机器来给你倒水，有多难呢？意想不到的难！看看图1-12这个场景，如果你的电脑配备有摄像头，那么它怎么知道茶壶在哪里呢？要知道，茶壶的材料、颜色、形状和角度，可以有几乎无穷多的变化。甚至有些茶壶跟哈哈镜一样，会把旁边的物体的形状都扭曲反射出来。桌上的物品附近都有各种反光和阴影，不同材料的反光特性还不一样，这些都会大幅度的影响机器对物品的识别。



图1-12 端茶倒水

为了识别物体，机器需要常识知识，它的头脑里必须有概念，必须知道什么样的东西才能叫做“茶壶”和“茶杯”。不要小看这一步的难度，这意味着机器必须理解基本的“拓扑结构”，什么叫做“连续的平面”，什么叫做“洞”，什么是“凹”和“凸”，什么是“里”和“外”…… 另外，这机器必须能够分辨物体和阴影。它必须知道水是什么，水有什么样的运动特性，什么叫做“流动”。它必须知道“水往低处流”，然后它又必须知道什么叫“低”和“高”…… 它必须知道茶杯为什么可以盛水，茶壶的嘴在哪里，把手在哪里，怎样才能拿起茶壶。如果一眼没有看见茶壶的把手，那它在哪里？茶壶的哪一面是“上面”，要怎样才可以把水从茶壶的嘴里倒出来，而不是从盖子上面泼出来？什么是裂掉的茶杯，它为什么会漏水，什么是缺口的茶杯，它为什么仍然可以盛水而不漏？干净的茶杯是什么样子的，什么是脏的茶杯，什么是茶垢，为什么茶垢不算是脏东西？如何控制水的流速和落点，什么叫做“水溅出来了”，要怎么倒水才不会溅出来？……

也许没有想到，倒茶这么简单的事情，需要用到如此多的常识。所有这些变数加在一起，其实远远的大于围棋棋局的数量，人却可以不费力的完成。这能力，人却对此不以为然，称之为“琐事”！这其实是非常宝贵的、让机器难以匹敌的能力。

专家系统没落的根源之一就是领域问题其实往往需要常识为基础才能解决，但是常识并未进入专家系统的知识库，造成了规则难以穷尽的问题。

对于常识性知识，需要构造通用知识库。领域知识库+通用知识库才有可能完善地解决领域问题。2018年4月21日，阿里巴巴宣布联合清华大学、浙江大学、中国科学院自动化所、中科院软件所、苏州大学5所高校发布藏经阁（知识引擎）研究计划。该计划以结构化的形式描述客观世界中概念、实体及其之间的关系，将互联网信息内容表达成更接近人类认知世界的形式，从而使计算机具备类脑推理能力。藏经阁就是一个通用知识库。

**（3）推理效率低下**

规则很多，推理过程很长而分支又多，会导致规则组合爆炸，难以进行。例如，当规则很多时（多于100条规则），系统速度就会很慢。

本体推理实践中会更慢，三层的短程推理都需要几分钟，长程推理不可想象。

**（4）没有记忆和学习能力**

一般的基于规则的专家系统并不具备记忆功能，同样的推理会重复执行，同样的错误会反复出现。

专家系统也往往不具备从经验中学习的能力，并不能自动修改知识库，例如调整规则、添加规则。修改和维护系统的任务仍然由知识工程师来做。

**（5）健壮性不足**

如果专家系统对于一个问题找不到任何匹配的规则，系统无法给出任何结论。系统的知识库局限于存储的规则，因此如果没有规则可以应用，系统也无法给出备用选项。因此，专家系统是脆弱的。

如果能对上述五个瓶颈做适应性限制和完善，基于知识的逻辑推理系统还是具备实践落地的可能的。

### 1.4.3 有限的人工智能

计算的所有基本要素包括：

1、基础的数值，例如整数、字符串、布尔值等；

2、表达式，包括基本的算术表达式、嵌套的表达式；

3、变量和赋值语句；

4、分支语句；

5、函数和函数调用。

上述“计算要素”就是几乎所有类型的计算系统所需要的东西，所以这种“机械化的信息处理”，也就是计算机器，擅长做的事情是：

* 在庞大、高纬度、复杂数据中，耐心、精准、快速地工作；
* 人工智能不只是提供结构化数据的查询，也可以处理非结构化数据。

机器不能做的事情：

* 机器可以感知的信息很少，不可能替代设计师的工作，例如分析甲方基础数据，判断xx…，决策xxx。
* 在领域知识这块，机器对信息和数据的要求极为严苛，绝大部分的信息是只有人可感知可理解。

如果我们对信息进行细分，可以更清楚地看到设计师和机器的能力界限，如图1-13所示。

机器可以处理的信息包括：结构化信息、非结构化信息、部分语言可以表达的信息、一小部分人可以感知的信息，其中最擅长的是结构化信息。处理这些信息，可以是计算、转化、识别、分类、预测等等。在人工智能领域，这种处理体现的是**替代人作为动物的那一部分能力，称之为感知智能**，例如深度学习可以把视觉识别的准确率提升了10-20%，让语音和图像的识别能力前进了一大步。



图1-13 设计师、机器与信息的关系

但是，**人工智能技术的瓶颈在于替代我们最近几千年发展起来的那些认知能力，也就是我们有了符号思维能力之后的认知智能**。深度学习这种算法在自然语言处理或是知识图谱这方面的提高只有1-2%。**人工智能在“理解”或“思考”的认知能力上，只能算是认知智能的初级阶段**。至于未来的高级认知智能是不是人类智能，在目前人脑的运作机理都不清楚的情况下，那种智能肯定不是人类智能，是一种机器智能，这个问题跟我们的问题关系不大，不再展开。

所以，现在的算法是有局限性的，我们在构建人工智能系统的时候要理解它的局限性。

# 2 大数据

大数据是机器学习的粮食，也是目前第三次人工智能浪潮的动力源泉。大数据和人工智能如此密切，对其展开调研是有必要的。

大数据并没有明确的定义，但有一种说法可以容易的区分出大数据和小数据：单台机器处理不了的叫大数据，否则叫小数据。有一定道理。

## 2.1 方法

一切科学研究（包括自然科学和社会科学）的目的只有两个：

（1）对过去发生的（观测到）现象进行总结和解释

（2）对未发生的（未观测到）现象进行预测和判断

预测能力是检测一种理论或方法是否有效的真正依据。预测帮助我们决策，只有预测未发生的事情的结果我们才能获益。对过去的解释和对未来的预测，必须采用统一的知识、理论，不可以解释过去用一套理论，预测未来用另外一套理论。

建立统一的知识和理论也有两种方法：

（1）归纳法：从过去发生的现象（包括人工设计的实验）找出共同点和规律，提炼出规则和理论知识，并得到科学共同体的检验和认可。典型的学科有物理、化学和工程学科。

（2）演绎法：从共同认可的理论知识中，继续推理进而衍生出新的理论，典型的学科就是数学，从几条公理开始，演绎出现在数学学科的参天大树。目前物理学科也是朝着这个方向在走，由于研究尺度已经大到或者小到无法设计实验来验证，只能在人脑中演绎。

显然，大数据使用归纳法来获得知识。

具体方法有很多，而且和机器学习、深度学习的基本上类似，前面没有介绍的这里补充一下方法介绍。

### 2.1.1 无监督学习

用来建立预测模型的称为有监督学习算法（supervised learning），用来建立描述模型的称为无监督学习算法（unsupervised learning）。简单地说，假设**x表示输入，y表示输出，观测数据中有X也有y就是有监督学习；观测数据中只有X，那就是无监督学习。**

产生直接价值的应用都是预测，存在y。无监督学习往往是为没有y的观测数据人工创造y，从而为下一步应用即预测打下基础。聚类（Clustering）是最常见的无监督学习，即将观测数据分成k类，k最小为2，例如将图2-1.A中的点分成2类。注意，图中横坐标为特征x1，纵坐标为特征x2，两者是等价的，这是简单的数据点在二维空间的表示。机器学习时，一定要注意纵坐标是表示特征x2还是目标值y，否则容易混淆。

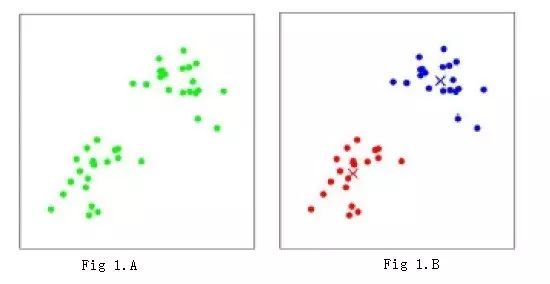


图2-1 二维数据点的聚类（左A右B）

最简单的聚类方式，就是按照类内的点距离最近，类间距离足够远的原则进行数据聚类。这个例子为二维空间，聚类结果一目了然，如图2-1.B所示，采用红、蓝两色表示类别。

聚类的目的还是为了分类，通过聚类发现k种模式或类别，然后人工分析这k种模式的特征，并给每种模式贴上模式（类别）标签，这相当于人为给每个数据点加上一个y。其主要应用为商业营销，例如精准广告营销。通过所有用户的注册信息、IP位置、浏览行为和购买行为，为用户画像，将用户分成几个典型类别；再针对某个具体用户的以上信息，将他/她归类，并针对性投放广告和推荐产品。亚马逊的荐书系统就是这个原理，这是是亚马逊最初成功的一个关键因素。

无监督学习主要是商业和电子商务应用，在工业中应用不多，可能的一个应用是在数据预处理发现异常点。

### 2.1.2 有监督学习

有y值的有监督学习才具工业应用价值。

预测模型用于这样的学习任务：利用数据集中的其他数据（X）来预测另一个值（y）。**学习算法的目的是发现并且建立目标特征（需要预测的特征）和其他特征之间的关系。**（是的，就是统计学上的拟合函数，还是纯数值的模型）

定量描述有三种方法：表格、图形和公式。公式压缩的信息量最大，也最容易被计算机处理。机器学习在工业中的应用以描述X->y 的定量模型最为常见，包含y值的监督学习又分为回归和分类两类问题：

* **若y值是连续的数值，则这个学习就是回归问题**
* **若y值为有限离散值，则这个学习就是分类问题**

无论分类还是回归，其数学模型中都有参数向量Θ，学习的任务就是计算得到最优参数向量ΘML，使得观测数据集（X，y）出现的概率最大，即最大似然估计：

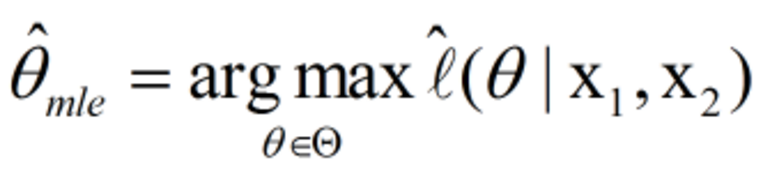


图2-2 最大似然估计

#### 2.1.2.1 y是连续值的回归问题

**（1）线性回归（Linear Regression）**

数理统计和实验数据处理的教材中到处都是，是的，这算是一种机器学习方法。

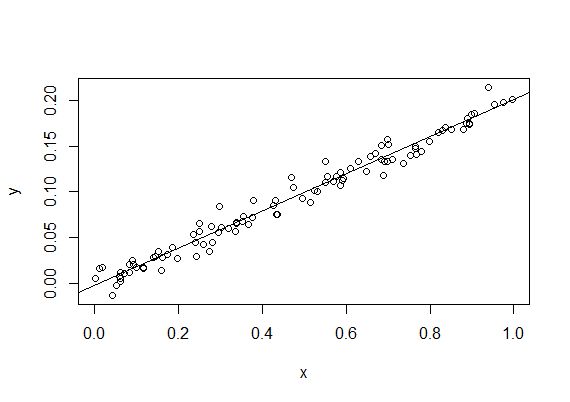


图2-3 线性回归

**（2）人工神经网络（Artificial Neural Networks，ANN）**

比较简单的ANN模型为三层结构前馈网络，第一层为输出层，只接受输入信号，没有神经元函数计算，所以输入层的节点并不是神经元；**第二层为中间隐含层，每一个节点都是一个神经元**；第三层为输出层，每个节点也是神经元，并将每个神经元的输出作为最终输出；若系统只有一个输出y，则输出层只有一个神经元；若有多个输出yi，则输出层有多个神经元。

**理论证明，两层神经网络(不含输入层）可以无限逼近任意连续函数。**ANN模型中并没有用复杂的函数和计算，只在每层使用一次矩阵计算，却实现了任意逼近，确实很神奇！

#### 2.1.2.2 y值为类别的分类问题

分类问题的输出是一个类别标签，典型的有两类分类，例如是/否，好/坏，男/女；也有多类分类，例如一级/二级/…/五级，数字0、1、2….9，字母A、B、C...Z。

典型应用有车牌号码识别、垃圾邮件识别、语音识别、手写体识别等。

**（1）Logistic回归**

二类问题可以转为为目标值y为0/1的回归问题。

如果直接使用一般线性方程 y=βx+b，由于y只能取值0和1导致不连续性，所以一般线性方程无法回归。Logistic回归模型是将特性的线性组合作为自变量，由于自变量的取值范围是负无穷到正无穷，因此，使用logistic函数将自变量再次映射到(0,1)上，映射后的值可以被认为是y=1的概率。Logistic回归原理如图2-4所示。

注意，这里logistic函数和神经网络中激活函数Sigmoid函数完全一样，所以Logistic回归本质上就是一个神经元模型，两者的区别是训练算法的原理不同。

若是线性可分，采用Logistic回归、神经网络感知机Perception模型都可以得到一条直线或超平面将测试数据集分开。

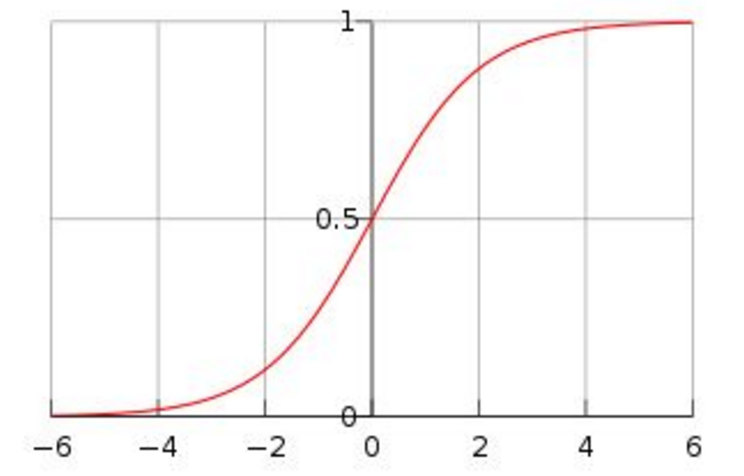


图2-4 Logistic回归

**（2）支持向量机（Support Vector Machine，SVM）**

一个线性可分二类问题中的所有可行分割平面中，有且只有一个分割平面，使得每个类别中离这个平面的距离最近的点到平面的距离相等，并且达到最大。如图2-5中的最优超平面。

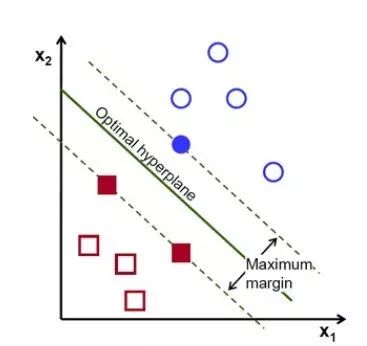
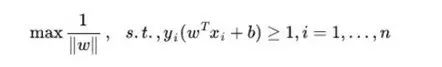


图2-5 最优分割超平面

将该表达转化为一个优化问题，表达为：



在SVM算法中不同的类别用-1/1表示，所以上式中yi的取值为-1和1。这个问题可以转化为一个二次规划问题，存在全局最优解。最终最优平面是由落在虚线上的几个点决定，这几个点被称作支持向量（Support Vector）。

对于线性不可分问题，支持向量机采用核函数将问题从低维空间映射到高维空间，从而在高维空间线性可分，如图2-6所示。

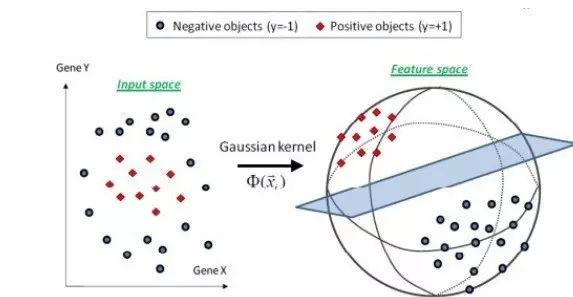


图2-6 低维向高维的核函数映射

这些优点是SVM迅速打败了神经网络的原因，但是SVM算法中每个数据点都有一个模型参数，对真正的大数据（GB级）问题，其计算量令人和计算机无法接受，所以它现在并不是大数据级别人工智能的主流算法。

分类还有其他算法例如朴素贝叶斯分类、决策树等，不再赘述。

### 2.1.3 其他数据预处理算法

除了算法，机器学习在实际应用项目中还有很多工作：

* 数据预处理，使数据结构标准化、符合算法接受的格式
* 特征提取，在所有可能潜在的输入变量中识别出真正同目标相关的变量
* 大型矩阵计算和并行处理，提高计算速度
* 模型性能评价，找到最佳的模型和结构

据说，实际项目中80%的工作都是数据收集和数据预处理这种脏活（dirty work），但是可以说这些脏活才是最基础、最重要的事情，否则后面的所有工作都是没有保证，Rubbish in，Rubbish out！

## 2.2 思维

大数据技术的说白了就是机器学习技术在大数据上的扩展，得到的仍然是统计/概率结果，反映的是相关性，而不是因果性。

因此，大数据思维的核心是要放弃对绝对真理的奢求，不追求因果性，不追求完美，但追求高效率；通过数据获取便捷和计算速度快来确保预测效率。

## 2.3 前提要求

不管是大数据还是小数据，数据的完备性是对数据的第一要求，即收集到数据能够覆盖到各种情形；第二要求是准确性，但是由于大数据下数据量大，可以克服部分数据不准的问题，所以大数据条件下对数据准确性的要求要远小于小数据条件。

弄清楚大数据方法、思维和前提要求后，再来分析一下大数据将会在那些领域取得突破，超越目前的技术水平。**取得突破的领域应同时具备如下要求：**

* **系统复杂，关联因素多且因素间难以独立分离，系统不受控制**
* **难以找到或短时内难以找到系统的因果关系，系统运行时可靠性要求并不高**
* **数据收集比较容易，并且数据覆盖范围广**

举一个符合这些要求的例子：医学。

人的身体是一个极其复杂的系统，现代科学对我们身体如何运行和生病机理的了解还是很少，对药物治疗某种疾病的机理并不真正了解（所以，目前统计学、双盲随机实验是医学研究和药物开发的主要方法），某人要得什么病基本也不是谁能控制的；虽说人命关天，不管在发达国家还是发展中国家，误诊率都非常高，大不了过几天换种药再试试，医生把人医治死也并不少见，所以说治病的可靠性要求并不高；病历档案极多，尤其是医院采用电子化病历后，每个病人的医治记录都保存记录，各种疾病都有。医疗领域完全符合上述几点，基于大数据的人工智能必然改变医疗行业！

## 2.4 应用领域和局限性

**不能满足前提要求任何一条的领域，大数据技术会作用有限或无能为力：**

* 系统已经被研究的相对透彻，因果关系明确
* 数据获取困难，或取得的数据不具完备性
* 系统可靠性要求极高，出现故障后必须可以追溯原因

也举两个例子：

* 大型流程工业生产系统，例如石化装置、核电厂。各个环节的机理在实验室研究的比较透彻，各种关系明确；运行数据虽多但却单一；系统可靠性要求极高，一旦停车，轻则损失千万元钱，重则爆炸死伤多人。
* 地震预测，系统很复杂，问题难度很大，但是数据太难收集，靠天吃饭，遇到地震才能收集到一个样本数据，数据量太少。

所以，工业界用户必须深入理清自己的业务流程，到底有哪些能创造价值的目标，为了优化这些目标需要哪些数据支持；无论如何，利用数据量化和优化自己的业务流程则是永远正确，采用大数据还是小数据并不重要。在具体科技领域，没有领域知识和领域专家，仅仅指望大数据和人工智能想实现技术的飞升是不现实的想法。

# 3 化工工程设计

以上分析人工智能和大数据，化工工程设计才是落脚点。分析设计过程和过程中的数据特点，有助于形成智能化设计方案。

## 3.1 工艺包设计

实验室开发完成后，工业化设计的第一步是概念设计，或者叫工艺包设计，包含系统结构优化、系统参数优化和单元设计优化。前两项是过程系统工程的内容，后一项是化工原理或者单元操作（unit operation）的内容。大数据技术和这三者都没啥关系，如果非要扯点关系，就是单元设计中用到的经验系数，例如精馏的塔板总效率、换热中的污垢阻力系数。

估算塔板总效率最常用的一张图就是图3-1的O'Connell关联图。O'Connell收集了32个工业塔和5个实验塔的数据，绘制了关联图。这张图体现了大数据的思维，为什么会这样关联？说不清因果关系。这张图最大的创新便是找到用液体粘度和挥发度的乘积作为横坐标。

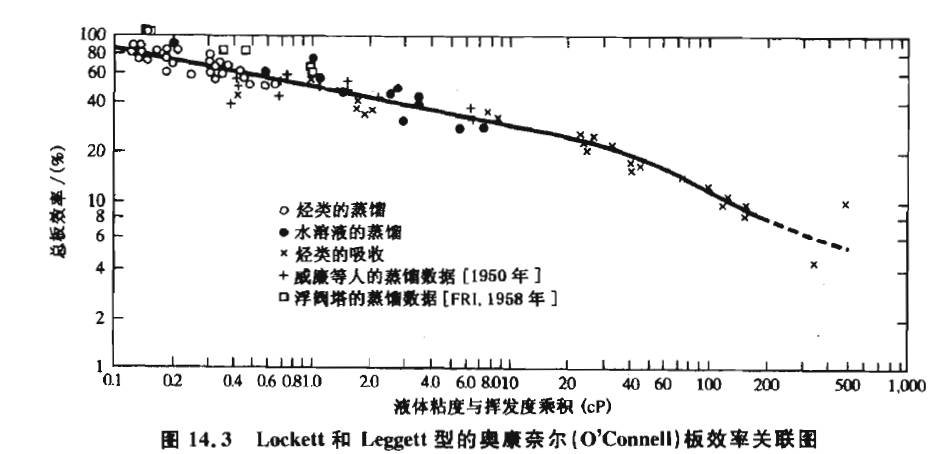


图3-1 Lockett和Leggett型的O'Connell板效率关联图

理论上，如果有人能收集了上万个工业塔的设计数据和运行数据，若要设计一个新塔，当确定物料体系、分离要求和处理量后，无需任何化工知识，便可直接从之前收集到的塔的信息中学习、设计出所要求的塔来，无非就是根据相似性确定塔的结构和各个尺寸。问题是，什么组织能够收集这么多工业塔数据？

## 3.2 工程详细设计

化工装置的工程设计，主要依据是设计规范和设计经验。假设把设计师看作是一台计算机，输入是条件、设计规范和经验，处理过程就是画图的过程，输出就是合格的图纸。下面以管道热补偿为例，解剖这只麻雀，分析一下设计过程哪些环节是机器可以参与的。

管道热补偿设计原则：某些压力管道输送的介质常常具有高温、高压特征或由于工作环境温度的变化，造成金属材料的热胀冷缩，管道布置设计应充分考虑吸收热位移。一旦布置不当，将造成管道某些部位产生过大的热应力而破坏或对支架产生过大的推力而影响管道支架的安全，从而影响管道系统的正常运行。

设计案例：图3-2是一条连接两台换热器间的管道，工作压力P=4.0MPa，工作温度t=316℃，管道外径Ф219，换热器中心距2.1m，中间连接管道水平管中心距2m，A处设有一异径管DN200/DN150，管道设计安装尺寸如图3-2所示。

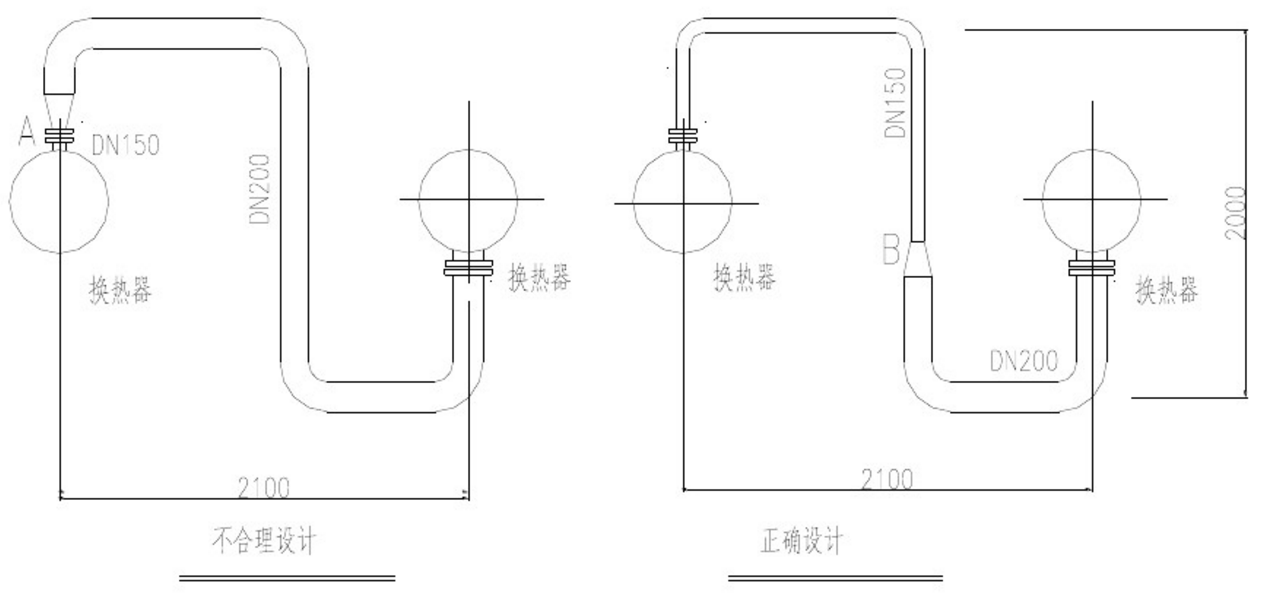


图3-2换热器间的管道布置示例

不合理的设计：在运行中，异径管与法兰连接处焊缝多次出现裂纹，原因是：由于设备布置太近，中间接管短、管径大，柔性差，吸收热膨胀变形的能力较小，对管系柔性分析可知，A处受到很大的横向推力和推力距。

正确的设计：由于场地所限，不可能改变管子长度，只好改变管子直径，并将异径管接头由A处移到B处，管子AB的外径由219mm降到159mm，使管道柔性得到改善，同时异径管与管子连接处由于形状突变引起的应力集中的位置也由较高应力区域移到了较低应力区域。这样，管道的最大应力值下降了，经过几年的生产考验，获得了满意的结果。

这个设计案例中有很多环节值得分析。

**设计条件到底是什么？**

给定设计条件：工作压力P=4.0MPa，工作温度t=316℃，管道外径Ф219，换热器中心距2.1m，中间连接管道水平管中心距2m。

这个给定的设计条件给设计师看是可以的，给机器看的话，还差很多信息，例如，哪个图例是换热器？需要连接哪两个换热器？哪个图例是法兰？是连接两个换热器的法兰吗？液体流向是从哪里到哪里？换热器间的管道有哪几种类型可以选择？只有“S”型吗？S型是左右对称分布吗？S型是上下对称分布吗？S型上下可以出来多少？S型管道由一根还是多跟组合连接而成？组合的话各段尺寸是多少？S型怎么与A处换热器相连？异型管的大头外径是多少？S型管壁厚是多少？异型管大头能接上S型管道吗？异性管的小头外径是多少？换热器A处的开孔大小够不够？S型管道有布置空间吗？连接点的推力和推力距怎么计算？各个连接点的横向推力和推力距多少合适？各个连接点的纵向推力和推力距多少合适？……

上述问题，设计师和机器的解决方案是大不相同的，设计师可以凭借人类的经验和智慧完成设计，但机器还不行，还需要更详细的设计规则，各自的差别蕴含着人工智能技术应用的可行性，列表如3-1所示。

表3-1 管道布置设计设计师与机器的解决方案对比

|  |  |  |
| --- | --- | --- |
| 问题 | 设计师的解决方案 | 机器的解决方案 |
| 哪个图例是换热器？  哪个图例是法兰？ | 观察 | 有图例：根据图例读取规则读取即可；  无图例：根据公式计算图形以识别对象；无法计算的，训练机器学习这种图形，让机器学会识别 |
| 需要连接哪两个换热器？  液体流向是从哪里到哪里？ | 观察 | 设计条件细化 |
| 是连接两个换热器的法兰吗？ | 观察 | 设计规范 |
| 换热器间的管道有哪几种类型可以选择？只有“S”型吗？ | 经验 | 依据设计规范推理得到 |
| S型是左右对称分布吗？  S型是上下对称分布吗？ | 经验 | 基于案例推理历史案例得到经验 |
| S型上下可以出来多少？ | 经验数据 | 训练机器学习历史案例得到数据 |
| S型管道由一根还是多跟组合连接而成？组合的各段尺寸是多少？ | 经验和经验数据 | 基于案例推理历史案例得到经验 |
| S型怎么与A处换热器相连？ | 经验 | 依据设计规范推理得到 |
| 异型管的大头外径是多少？ | 简单计算 | 依据设计规范推理并计算 |
| S型管壁厚是多少？ | 观察 | 设计条件细化 |
| 异型管大头能接上S型管道吗？ | 口算 | 简单计算 |
| 异性管的小头外径是多少？ | 经验 | 依据设计规范推理并计算 |
| 换热器A处的开孔大小够不够？ | 口算 | 简单计算 |
| S型管道有布置空间吗？ | 观察 | 设计师可以直观估算，机器需要多边形空间计算得到可用布置空间 |
| 连接点的推力和推力距怎么计算？ | 依据设计规范计算 | 依据设计规范计算 |
| 各连接点的横向推力和推力距多少合适？ | 经验数据 | 训练机器学习历史案例得到数据 |
| 各连接点的纵向推力和推力距多少合适？ |

细分下来，设计过程中人工智能技术可以发挥作用的地方有：

* 图形条件识别：可以通过机器学习训练，让机器学会识别各种图例和图形，这属于感知智能；
* 设计规范：可以通过基于规则推理的方法，正向推理生成图形；
* 经验数据：可以通过训练机器学习历史案例得到数据，但问题是不同工况的历史案例不好整理，需要投入大量精力构建案例库。
* 设计经验：可以通过基于案例推理历史案例得到经验，但问题是不同工况的历史案例不好加工，需要投入大量精力构建案例库。

总结一下，机器学习可以用于图形、文字识别，也可以用于找到更好的经验值；基于规则的推理，可以正向推理生成图形；基于案例的推理，可以复用历史经验。难点在于训练集或者案例库构建困难，且数据量有限。

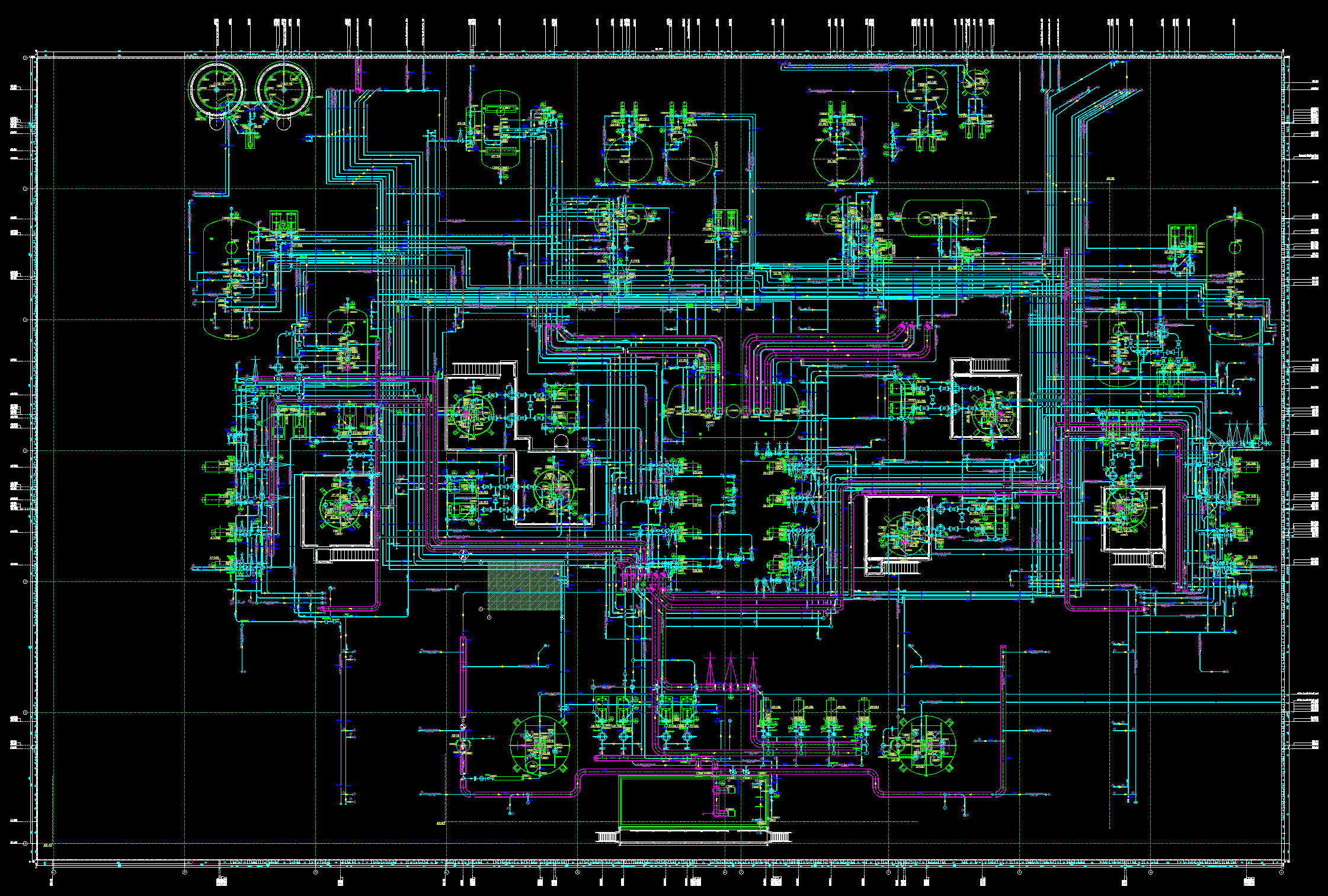


图3-3 管道布置图

需要注意的是，机器学习的图纸只能是局部，从小范围中学习知识。如果是整张图纸，例如图3-3，这是设计师这台人体计算机根据条件、规范、经验得到的结果。机器学习整张图纸，相当于将人类设计师曾用到的各种条件、规范、经验逆推得到，就目前的人工智能技术而言，“理解”和“思考”能力非常弱，想要做到，难如登天。

# 4 智能化设计方案

用历史眼光来看，整个设计IT建设的历史，可以分为三个阶段：信息化、集成化、智能化。根据我们对公司的初步了解，公司的信息化工作基本完成，集成化工作刚刚开始，智能化工作还没开始。

我们能够真正建立起来的合理的追求目标，是建立起一个智能化辅助设计的协作系统。

## 4.1 实现路径和目标

智能化设计的定位就是助手，不是取代设计师，而是让设计师有更多的时间去创造价值，让设计行为减少对个人经验的依赖，从而减少人员流动对公司的冲击，建立新型的智能化辅助设计协作系统。

建设一个人工智能的系统平台，不能直奔平台而去，因为通常人工智能系统都是一个非常复杂的系统，都是要几百个人年才能做得到的。

John Gall提出过一个盖尔定律（Gall’s Law）：“一个切实可行的复杂系统势必是从一个切实可行的简单系统发展而来的。从头开始设计的复杂系统根本不切实可行，无法修修补补让它切实可行。你必须由一个切实可行的简单系统重新开始。”这个盖尔定律，并不仅仅可以用在人工智能方面，它是说我们做任何一个复杂的东西，你很难一下子从复杂的事情开始，你必须要从简单的事情开始。

这可能是大型复杂系统的普遍规律，它不是设计出来的，它是生长出来的，它是一点一点地通过一些很小很小的组件，一点一点地逐渐地去碰撞、去组合，慢慢地生成的。直奔这个平台而去，基本上可以收获100%的失败。

因此，人工智能产品由于其复杂性，其**设计核心是其中间路径而非最终目标**。

所以智能化设计的切入点，一定不是直接出图，而是从流程分析、基础设计、详细设计、出图和统计报表的各种困难，从这些小事情开始做，从每一个设计师心烦却无创造性的问题开始，演化出一个一个组件，生长出一个一个方案，最终总结出智能化设计系统。这个过程真的有点像做教育，只是教育的是机器而不是儿童。

## 4.2 智能化层次

人工智能最大的作用不是模仿人类，而是**把人类智力劳动中可机械化的部分机械化**，这个可以分层次逐步实现：

（1）工具：Excel数据提取、Word数据提取、PDF数据提取、CAD元件提取、OCR文字识别、Word模板生成、各种专业软件之间的数据转换等等，一是靠信息化手段实现，主要是用数据结构解决机器与机器之间的通信摩擦力，填补公司信息化那些缺失的部分。二是靠人工智能技术实现，将这些人类可读的数据高效转化成机器可读的数据结构，解决人与机器之间的通信摩擦力。这些工具看起来小，但往往成为制约效率的关键。

（2）解决方案：智能选型、智能选材、智能报价、智能搜索、智能推荐等等，以底层工具为基础，解决某一方面的智力劳动机械化问题，例如物理支吊架智能选型辅助设计师完成在几百个支吊架间选择并简单计算的麻烦，让设计师有时间优化完善支吊架和管道布置设计，提升设计质量。

（3）系统：从公司层面是智能化辅助设计系统，从专业层面看是XX专业智能辅助设计系统，是设计师的助手。

## 4.3 可行的方法

根据上面的种种分析，智能化设计是可行的。可行的方法上面已经总结过了，这里单列一下以明示。

设计过程中人工智能技术可以发挥作用的地方有：

* 图形条件识别：可以通过机器学习训练，让机器学会识别各种图例和图形，这属于感知智能；
* 设计规范：可以通过基于规则推理的方法，正向推理生成图形；
* 经验数据：可以通过训练机器学习历史案例得到数据，但问题是不同工况的历史案例不好整理，需要投入大量精力构建案例库。
* 设计经验：可以通过基于案例推理历史案例得到经验，但问题是不同工况的历史案例不好加工，需要投入大量精力构建案例库。

总结一下，机器学习可以用于图形、文字识别，也可以用于找到更好的经验值；基于规则的推理，可以正向推理生成图形；基于案例的推理，可以复用历史经验。难点在于训练集或者案例库构建困难，且数据量有限。

业界有句话：“**有多少人工就有多少智能**”，深以为然。

# 5 总结

有句话说得好：设计不过是一门失传的艺术的别名，这门艺术的名字叫做“思考”。没有任何机器可以代替人的思考，所以设计师是一种不可被机器取代的工作。

**智能化辅助设计系统是承载你的知识和逻辑的记忆体。它是你耐心安静的秘书，它是你永不疲倦的助手，它是你精准的硅基大脑。它不再是像传统设计终端那样冰冷疏离，它和你一起学习前行在设计道路上！**