知识小料

「电计 2203 班」周常规知识整理共享

第 40 期

日期：2024-11-4
学科：人工智能

本文档用于对《人工智能》课程作出简明复习。

1 人工智能概述

人工智能的英语为 Artificial Intelligence，日语为「人工知能（じんこうちのう）」。

狭义：人工的方法在机器上实现的智能；广义：人类智能行为规律、智能理论方面的研究。

三次低谷：

- 第一次低谷：1973 年英国发表 James Lighthill 报告；
- 第二次低谷：日本智能（第五代）计算机研制失败；
- 第三次低谷：知识词典日趋势微、网络百科兴起

三个发展阶段：弱人工智能、强人工智能、超人工智能。

三个主流方法：符号主义、数据驱动、探索与利用。

- 符号主义人工智能为核心的逻辑推理；
- 数据驱动为核心的机器学习；
- 探索与利用为核心的强化学习。

2 知识与谓词逻辑

知识的特性：相对正确性、不确定性、可表示性与可利用性。其中，不确定性由随机性、模糊性、经验性、不完全性引起。

神经网络属于知识的一种表示形式。

事实与规则的区别。事实：「p 是 q」；规则：「若 p 则 q」。

2.1 一阶谓词逻辑

涉及到以下概念：

- 命题、命题逻辑；谓词、个体；
- 连接词 5 个（依次为 $\neg\ \wedge\ \vee\ \rightarrow\ \leftrightarrow$）、量词 2 个、谓词公式；
- 辖域、约束变元、自由变元、等价式
- 解释：永真、永假、可满足

常见的推理规则：

- 假言推理：$P,\ P \rightarrow Q \implies Q$
- 拒取式推理：$\neg Q,\ P \rightarrow Q \implies \neg P$
- 假言三段论：$P \rightarrow Q,\ Q \rightarrow R \implies P \rightarrow R$
- 全称固化、存在固化、反证法。

2.2 产生式与框架

四种产生式表示法[1]：

- 确定性规则的知识：IF P THEN Q
- 不确定性规则的知识：IF P THEN $Q\quad (x)$
- 确定性事实的知识：$(obj, attr, val)$ 或者 (rel, obj_1, obj_2)
- 不确定性事实的知识：$(obj, attr, val, x)$ 或者 (rel, obj_1, obj_2, x)

产生式的形式表示：BNF「巴科斯范式」。
产生式系统例子：动物识别系统。
框架、槽、侧面，相当于类与对象的关系。
知识图谱是互联网环境下的知识表示方法，通用表示形式是三元组。举例如维基百科、百度百科等。

3 确定性推理（离散数学・数理逻辑）

推理方式分类：

- 演绎推理：一般 $\rightarrow$ 个别；
- 归纳推理：个别 $\rightarrow$ 一般；
- 默认推理：假设某些条件默认成立的推理。

还有的按照确定性/不确定性推理、单调/非单调推理、启发式/非启发式推理。默认推理为非单调推理。

[1] object: 对象；attribute: 属性；value: 值；relationship: 关系，x: 置信度。

推理方向：正向、逆向、混合、双向推理。

多种匹配成功，则采用冲突消解策略。

自然演绎推理，包含规则：P 规则（给定前提）、T 规则（产生的结论作前提）、假言推理、拒取式推理等。

3.1 谓词公式化子句集

谓词公式化子句集一般包含以下九个步骤：

1. 消去条件、双条件符号（$\rightarrow \leftrightarrow$）；
2. 把 $\neg$ 移到紧靠字母的位置；
3. 存在量词标准化；
4. 消去存在量词：$(\forall x)(\exists y)P(x,y) \iff (\forall x)P(x,f(x))$；
5. 化为前束范式：前缀 $+$ 主表达式；
6. 化为斯科伦范式：$\bullet \wedge \bullet \wedge \bullet$；
7. 删除全称量词；
8. 消去 $\wedge$ 并改成集合形式；
9. 子句变量标准化。

谓词公式不可满足 $\iff$ 子句集不可满足。

3.2 鲁宾逊归结原理

只要能归结出 空子句，则证明子句集不可满足。

应用归结反演求解问题。可能引入 $ANSWER$ 子句参与归结。（需要练习）

4 不确定性推理

不确定性的表示与度量：

- 知识不确定性的表示——知识的静态强度：$CF(H,E)$
- 证据不确定性的表示——证据的动态强度：$CF(E)$

4.1 可信度分析：C-F 模型

对于产生式 IF E THEN $H(CF(H,E))$，$CF(H,E) \in [-1,1]$，正数表示 E 支持 H 为真，负数表示 E 支持 H 为假。

具体包括以下规则：（参见「知识小料」· 其三十五）

- 合取规则：$CF(E_1 \text{ AND } E_2) = \min\{CF(E_1),\ CF(E_2)\}$
- 析取规则：$CF(E_1 \text{ OR } E_2) = \max\{CF(E_1),\ CF(E_2)\}$
- 传递规则：$CF(H) = CF(H, E) \cdot \max\{0, CF(E)\}$
- 合成规则：若有 $\begin{cases} \text{IF} \quad E_1 \quad \text{THEN} \quad H \quad (CF(H, E_1)) \\ \text{IF} \quad E_2 \quad \text{THEN} \quad H \quad (CF(H, E_2)) \end{cases}$，须先由传递规则算出 $a = CF_1(H)$、$b = CF_2(H)$。设 $S = CF_{12}(H)$，则

  1. 当 $a \geqslant 0, b \geqslant 0$ 时，$S = a + b - ab$
  2. 当 $a < 0, b < 0$ 时，$S = a + b + ab$
  3. 当 a, b 异号时，$S = \dfrac{a + b}{1 - \min\{|a|, |b|\}}$

证据理论：概率分配函数、信任函数 Bel、似然函数 Pl 等。

4.2 模糊集

模糊集，可理解为刨除「确定性」的集合，允许元素对集合的隶属度有一个浮动的值。

假设论域（全集）为 $U = \{a, b, c\}$，则下表 1 给出了模糊集与普通集合的一个对照表。

	普通集合	模糊集
举例	$A = \{a, b\}$、$B = \{a, c\}$	$A = \left\{\frac{0.5}{a}, \frac{0.4}{b}\right\}$、$B = \left\{\frac{0.3}{a}, \frac{0.6}{c}\right\}$
交集	$A \cap B = \{a\}$	$A \cap B = \left\{\frac{\min(0.5,0.3)}{a}, \frac{\min(0.4,0)}{b}, \frac{\min(0,0.6)}{c}\right\} = \left\{\frac{0.3}{a}\right\}$
并集	$A \cup B = \{a, b, c\}$	$A \cup B = \left\{\frac{\max(0.5,0.3)}{a}, \frac{\max(0.4,0)}{b}, \frac{\max(0,0.6)}{c}\right\} = \left\{\frac{0.5}{a}, \frac{0.4}{b}, \frac{0.6}{c}\right\}$
补集	$\bar{A} = U \backslash A = \{c\}$	$\bar{A} = \left\{\frac{1-0.5}{a}, \frac{1-0.4}{b}, \frac{1}{c}\right\} = \left\{\frac{0.5}{a}, \frac{0.6}{b}, \frac{1}{c}\right\}$
笛卡尔积	$A \times B = \{\langle a, a\rangle, \langle a, c\rangle, \langle b, a\rangle, \langle b, c\rangle\}$	$A \times B = \begin{bmatrix} 0.5 \\ 0.4 \end{bmatrix} \circ \begin{bmatrix} 0.3 & 0 & 0.6 \end{bmatrix} = \begin{bmatrix} 0.3 & 0 & 0.5 \\ 0.3 & 0 & 0.4 \end{bmatrix} = \left\{\frac{0.3}{\langle a, a\rangle}, \frac{0.5}{\langle a, c\rangle}, \frac{0.3}{\langle b, a\rangle}, \frac{0.4}{\langle b, c\rangle}\right\}$

表 1：普通集合与模糊集的对比

其中模糊集的笛卡尔积（叉乘积），为 $\mu_{A\times B}=\mu_A^{\mathrm{T}}\circ\mu_B$，$\circ$ 运算像矩阵乘法一样结合，只是每一项由乘积变为取较小值。

还有模糊集的容斥加法、有界和、有界积等运算。

模糊推理：$B'=A'\circ R$，其中 R 为从 A 到 B 的模糊关系。

模糊决策：最大隶属度法、加权平均判别法、中位数法。

例子：温度低和风门开大的模糊关系。

5 EA「进化算法」 evolutionary algorithm

进化算法包括遗传算法（GA）、遗传编程（GP）等。

5.1 GA「遗传算法」 genetic algorithm

GA 和 GP 共用一个流程图，如图 1 所示。

编码与解码，基因型与表现型。编码有二进制编码、实数编码等方式，用于对染色体进行操作。

在 GA 中，个体通常为染色体。

GA 简要流程

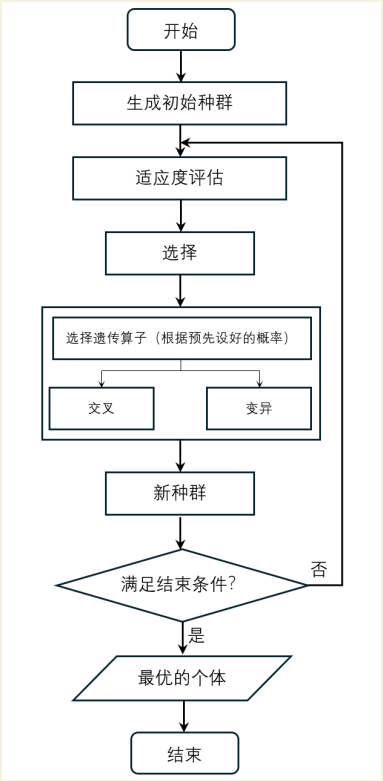
1. 初始化种群，个体编码。
2. 评估适应度：$Fit=f(x)$（最大化问题）或 $Fit=\dfrac{1}{f(x)}$（最小化问题）。可能需要尺度变换。
3. 选择：轮盘赌法、锦标赛法。综合考虑精英策略。
4. 交叉：一点交叉、两点交叉。
5. 变异：单点变异、多点变异。
6. 如果达到终止条件则终止，否则转第 2 步继续迭代。

GA 的特点：数学要求不高，搜索效率高，易于并行化。

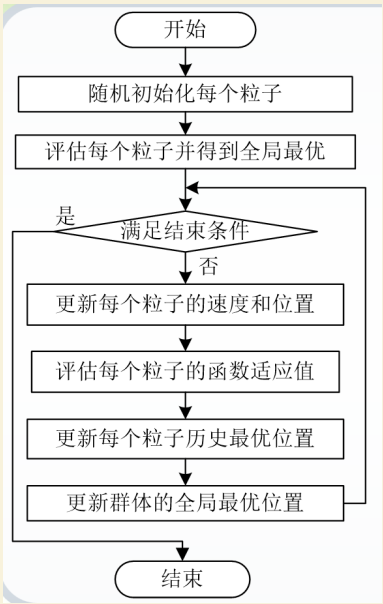
5.2 GP「遗传编程」 genetic programming

在 GP 中，个体通常为种群树、问题的解等。

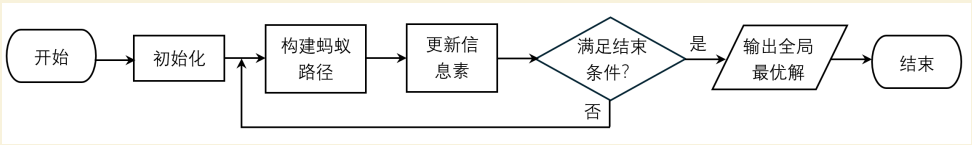
- 终端集：叶节点，代表每个 GP 个体的输入。
- 功能集：非叶节点，代表终端数据间的运算。



(a) GA 与 GP 流程图



(b) PSO 流程图



(c) ACO 流程图

图 1: 遗传算法 (GA)、**遗传编程** (GP)、**粒子群优化算法** (PSO)、**蚁群优化算法** (ACO) **流程图**

- 此外，需考虑充分性和闭包，功能集的函数应完备（有加必有减），除法受除 0 保护等。

GP 简要流程

1. 种群初始化：Full 法、Grow 法、Ramped half-and-half 法（一般取最后一种居多）；
2. 评估适应度：一般为分类成功率等；
3. 选择：轮盘赌法、锦标赛法，综合考虑精英策略；
4. 子树交叉：随机选择两个个体的子树互换；
5. 子树变异：随机选择一个个体的子树，替换为随机生成的另一子树；
6. 如果达到终止条件则终止，否则转第 2 步继续迭代。

GP 的流程与 GA 一致，算法是共通的。

GP 的应用：分类，演化出最优的分类器个体。

GP 的缺点：训练时间较长。对时间有要求的不建议用 GP。（数据规模：PSO < GA < GP）

代价敏感学习：对分类结果不平衡的一种特殊情境。可以将代价敏感规则结合 GA,GP,PSO,ACO 等设计算法。

相关问题：TSP「旅行商问题（traveling salesman problem）」。

6 PSO「粒子群优化算法（particle swarm optimization）」

PSO 是一种模仿鸟类觅食的算法，加入群体协作对个体认知的影响。PSO 的流程图如图 1 所示（见上页）。

PSO 和 GA 共通之处：种群需要初始化、个体需要编码、适应度需要评估，都是通过多轮迭代选出最优个体或者全局最优位置。

PSO 的速度决定式分三部分组成：

1. 惯性权重 ω：控制前一速度对当前速度的惯性影响；
2. 个体加速系数 c_1：控制粒子 i 向自身极值 $pBest_i$ 推进的加速权值；
3. 群体加速系数 c_2：控制例子向全局极值 $gBest$ 推进的加速权值。

没有惯性权重，则粒子的速度失去记忆性；没有个体加速系数，则粒子没有认知能力，完全「随大流」；没有群体加速系数，则粒子之间不存在交互，完全「各干各的」。

PSO 简要流程

1. 粒子初始化，评估每个 $pBest_i$ 并得到全局最优 $gBest$。
2. 更新粒子的速度和位置。
3. 评估粒子的适应度。
4. 更新每个 $pBest_i$ 和全局最优 $gBest$。
5. 若满足结束条件则结束，否则转步骤 2 迭代。

典型拓扑结构：星型结构、环型结构、齿型结构、冯诺依曼结构。

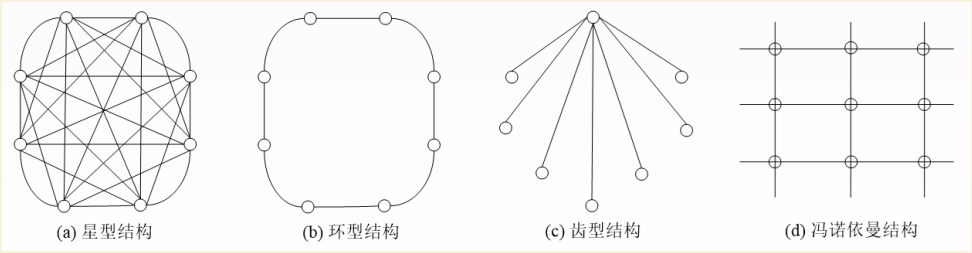


图 2：典型拓扑结构

全局 PSO 和局部 PSO 的区别：

- 全局 PSO （GPSO）收敛更快，但容易陷入局部最优。
- 局部 PSO （LPSO）多样性更高。
- 邻域较小，推荐局部 PSO；
- 邻域较大，推荐全局 PSO。

应用：基于粗糙集的粒子群和遗传算法综合算法（PSO-GA-RS）：对于 p 的数据用 PSO 处理，剩下 $1-p$ 的数据用 GA 处理。

ant colony optimization

7 ACO「 蚁群优化算法 」

ACO 是一种模仿蚂蚁觅食的算法，通过信息素来寻找最优解。蚂蚁在寻找食物的过程中往往是随机选择路径的，但它们能感知当前地面上的信息素浓度，并倾向于往信息素浓度高的方向行进。

ACO 有两个基本要素：路径构建、信息素更新。

ACO 简要流程

1. 初始化——设置蚂蚁数量、信息素权重、距离权重、蒸发率、最大迭代轮数等。
2. 构建蚂蚁路径——将每只蚂蚁随机放至出发点，计算蚂蚁前往各结点的概率，用轮盘赌法决定要访问的下一结点。重复这一步直至蚂蚁路径构建完成。
3. 更新信息素——根据路径更新信息素，并记录最优解。
4. 判断结束——如果达到结束条件则结束算法，输出最优解；否则转第 2 步迭代计算。

改进版本：精华蚂蚁系统、基于排列的蚂蚁系统、蚁群系统等。

8 搜索求解策略

按搜索方向分类：

- 正向搜索——数据驱动；
- 逆向搜索——目的驱动；
- 双向搜索。

按搜索方式分类：盲目搜索、启发式搜索。

8.1 状态空间表示

在状态空间中，状态为结点，操作为边。

- 状态：$Q = [q_1, q_2, \ldots, q_n]^{\mathrm{T}}$
- 操作：$F = \{f_1, f_2, \ldots, f_m\}$（也用 O 表示）
- 状态空间：(S, O, S_0, G)——（状态集，操作集，初始状态集，目标状态集）
- 求解路径：S_0 到 G 的路径。
- 可行解：由 S_0 迁移到 G 的那些边：$O_1, O_2, \ldots, O_k$。

（其中 n, m, k 为自然数）

8.2 盲目搜索策略：DFS、BFS 算法

一般的回溯思想：

- PS 表——当前路径的状态
- NPS 表——未处理的新路径状态
- NSS 表——无解状态
- open 表即为 NPS 表（未搜索的状态）
- closed 表为 PS ∪ NSS 表（已搜索的状态）。

宽度优先搜索（BFS）的次序是，先搜索第 1 层的结点，再搜索第 2 层，再搜索第 3 层……以此类推。（例：积木问题）

深度优先搜索（DFS）的次序是，保证每个结点的子状态比兄弟状态更先被搜索。（例：卒穿阵问题）

8.3 启发式搜索策略：A、A* 算法

相关概念：

- 启发式信息：可以用于简化搜索过程的信息。
- 启发式策略：利用与问题有关的启发信息进行搜索。（注：回忆第 3 章，有一种推理方式叫启发式推理。）
- 估价函数：$f(n) = g(n) + h(n)$，从初始结点经过 n 结点到目的结点的最小代价估计值。
- A 算法：对于下一结点，优先选择估价 $f(n)$ 较小的结点进行扩展。
- A* 算法：A 算法的最优形式。

例子：八数码问题。使用的 A 算法的估价函数是 $f(n) = d(n) + w(n)$，其中 $d(n)$ 为搜索树深度（根为 0），$w(n)$ 为不在最终位置的数字个数。

9 机器学习

9.1 机器学习概述

机器学习的分类：

- 监督学习——数据带标签，一般为回归、分类等任务
- 无监督学习——数据无标签，一般为聚类等任务
- 强化学习——序列数据决策学习，一般通过与环境交互来学习

典型的机器学习过程（以分类任务为例）：训练数据 → 训练学习算法 → 得到模型 → 输出新样本的分类结果

「泛化误差」与「经验误差」：

- 泛化误差——在未来样本上的误差，自然是越小越好；
- 经验误差——在训练集上的误差，但并不是越小越好（过小反而易导致过拟合）。

「过拟合」与「欠拟合」：

- 过拟合——模型学习了一些无关紧要的特征，甚至包括噪声；
- 欠拟合——模型学习不到位，遗漏了重要特征；
- 通常训练样本少、维度高，更有可能出现过拟合。

典型的评估方法（获取测试集）：

- 留出法——将训练集与测试集分开。
  - 这种方法需要假设数据是平衡的；
  - 可通过分层采样保持数据一致性；
  - 可多次重复划分，且测试集所占比例一般为 $\frac{1}{5} \sim \frac{1}{3}$ 为宜。
- k 折交叉验证法——将原始数据集均分为 k 份，进行 k 轮，每轮取一份作测试集，交叉变换，取 k 轮测试结果的平均值作为最终结果。

典型的性能度量指标：错误率、精度、查准率、查全率、F-score、F_1、F_β 等。对于分类问题，还可以用真阳性率关于假阳性率的图像（ROC 曲线）作为分类图，其曲线下方面积 AUC 常用作分类指标。

9.2 k 近邻（KNN）

对于测试样本，找到离其「最近」的 k 个样本，用投票法或平均法获知分类或预测结果。当 k 取不同值时，分类结果会有不同。

特点：没有训练过程，也称为「懒惰学习」、「急切学习」。这种方法的测试过程比较耗时，需要大量计算距离。

「距离」的定义也是重要的一环：计算维度更高的数据的「距离」会更慢，而且需要统一量纲。

9.3 决策树算法

决策树是基于树形结构决策的算法，具体包含 ID3、C4.5、CART 等算法。

- 叶结点——分类结果
- 非叶结点——属性
- 边——属性的值

决策树算法简要流程

对决策树的每一层，遵循以下流程：

1. 对于每一种属性（如色泽、纹理），计算其每一个子集的信息熵；
2. 计算这种属性的信息增益；
3. 选取信息增益最大的属性作为当前层的划分属性；
4. 对子结点递归地操作，直到所有属性划分完成，得到决策树。

相关指标：信息熵、信息增益、增益率、基尼指数等。

9.4 SVM「支持向量机 support vector machines」

SVM 的目标：找到超平面，使得它能够尽可能正确地区分两类数据，并使两类数据点距离该超平面最远。

SVM 是一个强分类器。

SVM 有以下两种「法宝」：

- 核函数——用于将线性不可分的数据点映射到更高维度
- 软间隔——允许少量数据点不满足约束

9.5 集成学习

集成学习是指，将多个学习器（弱分类器）结合在一起，提升性能（得到强分类器）。集成个体之间应满足「好而不同」原则。

集成学习器有以下组织方式：

1. Boosting——串行生成，个体之间依赖性强，每次调整训练数据的样本分布；
2. Bagging——并行生成，个体之间依赖性弱，采用自助采样法；
3. 随机森林——是 Bagging 的一个变种，强调采样与属性选择的随机性。

集成学习器有以下结合策略：

1. 平均法——简单、加权平均
2. 投票法——绝对多数、相对多数、加权投票法

此外，加权平均法未必优于简单平均法。

9.6 k-means 聚类

聚类目标：将数据集样本划分为若干个「簇」。

聚类既可用作单独过程，也可用作其他任务的前置过程。

k-means 聚类简要流程

1. 初始化每个簇的均值向量；
2. 更新簇划分；
3. 计算每个簇的均值向量；
4. 如果当前均值向量均为更新，结束；否则转步骤 2 迭代计算。

10 人工神经网络

10.1 神经元与神经网络

神经网络方法是一种隐式的知识表示方法。

神经元数学模型：从各输入端接收输入信号，求出加权和，用激励函数转换输出。

几个相关函数：

ReLU 函数	硬极限函数	对称硬极限函数
$y=\begin{cases}0, & x<0\\ x, & x\geqslant 0\end{cases}$	$y=\begin{cases}0, & x<0\\ 1, & x\geqslant 0\end{cases}$	$y=\begin{cases}-1, & x<0\\ 1, & x\geqslant 0\end{cases}$

表 2： 神经网络领域的常用函数

- 神经网络的结构：前馈型、反馈型
- 神经网络的工作方式：同步方式、异步方式

10.2 BP 神经网络与 BP 算法

BP 神经网络是典型的 前馈型 结构。作为对比：Hopfield 神经网络是典型的「反馈型」。

求解 BP 神经网络权重的原理是「梯度下降算法」。

BP 算法也称为「误差反向传播 (back propagation) 算法」，是对神经网络进行更新的一种算法。

BP 算法的简要流程

1. 初始化——对所有连接权和阈值赋值为随机任意小值 $w_{ij}(0),\ \theta_i(0)$；
2. 输入——从 N 组样本中选取一组样本输入到 BP 神经网络中；
3. 计算输出——计算各层结点的输出 y_i；
4. 计算误差——计算网络实际输出与期望输出的误差 e_i；
5. 反向传播——从输出层计算到第一个隐层，依次修改连接权值；
6. 判断——如果 N 组样本的误差均达到要求，结束；否则取另一组样本转步骤 2 计算。

BP 神经网络的优缺点：

优点 逼近特性强，泛化能力强，容错性高；
缺点 收敛速度慢，局部极值，难以确定隐层信息（解释性较差）。

10.3 深度学习、卷积神经网络

深度学习的流程是：获取数据—数据清洗—特征提取—特征选择—推理预测识别。其中特征提取、特征选择两个步骤又合称为「特征表达」，这是识别成功的关键。

深度学习是以端到端的方式逐层抽象，逐层学习的方式。

CNN「卷积神经网络 convolutional neural networks」是一个多层的神经网络，每一个卷积层都跟着一个池化层。

卷积层 也称 C （convolutional）层，用于特征提取；
池化层 也称 S （subsampling）层，用于特征选择。

对于卷积层，求卷积的过程应当掌握。对图像用卷积核进行卷积运算，实际上是滤波的过程，可以得到显著的边缘特征。

对于池化层，常用的池化操作有最大池化、平均池化等。池化层在语义上把相似的特征合并，降低了空间分辨率。

减少参数数量的方法：局部连接、权值共享。

10.4 生成对抗网络

GAN「生成对抗网络 generative adversarial networks」有两个角色：生成器和判别器，如同假币制造机与验钞机一般。

生成器用来生成数据，判别器用来判断数据的真假。通过不断训练，尽可能让生成器生成更真实的数据，直至判别器无法再判别数据的真实性。

GAN 有两个相互交替的学习阶段：

- 固定生成网络，训练判别网络；
- 训练生成网络，固定判别网络。

典型应用：图像处理。

GAN 的训练结果随机性较大，具有强烈的不稳定性，难以收敛。