深度学习学习笔记

从机器学习到大语言模型——一步步搞懂 AI 的底层逻辑

图文版 · 通俗易懂 · 2026

一、机器学习 vs 深度学习：思路的转变

不管是机器学习还是深度学习，终极目标都一样：让机器完成分类（这张图是猛还是狗）、回归（这套房子值多少錢）之类的任务。但做法截然不同——

> **积木比喻：** 传统机器学习像直接给你分类好的积木，告诉你“方积木搭底座，圆积木搭房顶”；深度学习像给你一堆原始木料，只说“搭一栋坚固的房子”——让模型自己切割、组合。

机器学习 vs 深度学习 对比

对比维度	传统机器学习	深度学习
特征获取方式	人工手动设计特征	模型自动学习特征
适用数据	结构化表格数据为主	图像、语音、文本等非结构化数据
数据量需求	相对较少	需要大量数据
代表方法	SVM、随机森林、XGBoost	CNN、RNN、Transformer

发展里程碑

- 1943 年：MP 神经元模型诞生，会“算”但不会“学”
- 1958 年：感知机出现，能简单学习，但只解决线性问题
- 1986 年：反向传播算法提出，多层神经网络成为可能
- 2012 年：AlexNet 横空出世，深度学习时代正式到来
- 2017 年+：Transformer、大语言模型（GPT、千问）相继涌现

二、深度学习的四大核心要素

训练一个深度学习模型，就像运营一家公司，必须把握四个关键环节：

要素	公司类比	在 AI 中的作用
数据（Data）	公司的“原材料”	没有数据什么都学不了；高质量数据是最核心资产
模型（Model）	公司的“组织架构”	神经网络的层数和结构，决定了模型的学习潜力
损失函数（Loss）	公司的“KPI 考核”	衡量预测与真实答案差距，指引训练方向
优化器（Optimizer）	公司的“管理路径”	梯度下降等方法，负责每步调整参数缩小差距

> **当前（2026 年）四要素重要性排序：数据 > 优化器 > 模型 > 损失函数 高质量的行业数据才是最宝贵的资产，有数据的公司才有真正的护城河！**

三、神经网络与激活函数

神经网络的结构

神经网络由三类层组成：输入层（接收原始数据）→ 隐藏层（反复加工特征，可以有多层）→ 输出层（给出最终答案）。每两层之间都有带“权重”的连接，像公路一样传递信息。

前向传播与反向传播

- 前向传播：数据从输入层流向输出层，每层做加权求和后经过激活函数处理，最终给出预测。
- 反向传播：预测错误时，从输出层往回计算每层参数的“责任”（梯度），再用优化器调整权重。这是深度学习的“炼金术”核心。

三种常用激活函数对比

激活函数	规则简述	优点	缺点
ReLU	正数原样输出，负数输出 0	计算快，训练效率高	负数区域“死亡”
Sigmoid	把输入压到(0,1)区间	输出平滑，适合概率输出	深层易出现梯度消失

Tanh	把输入压到(-1,1)区间	比 Sigmoid 收敛更快	同样存在梯度消失问题

算法示例

1. 一张猫咋图片输入网络，某神经元经加权求和得到 -0.3，经过 ReLU 输出 0，“这个特征不重要不往下传”。

2. 另一神经元得到 2.1，ReLU 后仍输出 2.1，“这个特征（猫耳朵轮廓）很强烈，继续传下去”。

3. 多层选择性传递后，输出层只收到最重要特征信号，模型准确判断“这是一只猫”。

四、梯度下降：模型学习的“导航系统”

核心比喻：蒙眼下山

把模型训练目标想象成“找到山地中最低的山谷”，纵轴是损失值（越低越好），模型的参数就是你在山上的位置。梯度下降的策略是：每次找到最陨的下坡方向（梯度的反方向），然后迈出一小步——这小步的大小叫做学习率。

> **图示说明：** 损失曲面示意图中，蓝色点为初始位置，红色点为全局最优（损失最低点），橙色点为局部最优险阱。蓝色虚线表示梯度下降的前进路径，步长即为学习率。

学习率的影响

- 学习率过大：在山谷两侧来回蘅踅，永远到不了最低点（震荡）
- 学习率过小：进展极慢，需要非常多步才能收敛，容易降在局部最优
- 合适的学习率：稳定快速向最低点收敛，是调参最重要的技巧之一

梯度下降的四种变体

变体	每次使用数据量	特点
批量梯度下降(BGD)	全部数据	方向准确但非常慢，内存消耗大
随机梯度下降(SGD)	1 条数据	速度快但方向噪声大，不稳定
小批量梯度下降	一小批数据	平衡了速度与稳定性，实际最常用
Adam 优化器	小批量（自适应）	自动调节学习率，目前最广泛使用

算法示例

1. 模型预测“这张图是猫的概率 0%”，真实答案是 100%，损失值高；优化器计算出“应增大与猫相关特征的权重”。

2. 梯度下降根据导数方向，把对应权重从 0.2 调整到 0.35，迈出一小步（学习率=0.1）。

3. 经过数千次迭代，预测概率从 30%上升到 92%，损失不断下降，最终收敛到较优参数。

五、卷积神经网络（CNN）

为什么需要 CNN?

一张 224×224 彩色图片有 5 万多个像素点。普通全连接网络处理时参数会爆炸式增长，而且会破坏像素之间的“位置关系”（如眼睛在鼻子上面这种空间信息）。CNN 专门为此设计。

> **卷积操作说明：** 卷积核（3×3 的小滑动窗口）在图像上逐步滑过，每步做局部匹配计算，就像一个“探照灯”专门探测图像中的特定模式（垂直边缘、角点等）。见图 5-1 示意。

CNN 层级特征提取原理

- 第 1-2 层：检测线条、边缘等低级特征
- 第 3-5 层：组合成纹理、形状等中级特征
- 第 6-8 层：识别眼睛、耳朵等高级特征
- 输出层：综合高级特征做出分类决策

正是这种层次化的特征提取，让 CNN 在图像识别上如此强大——它像人的视觉系统一样，逐步从细节到全貌理解图像。

算法示例

1. 猫咋图片经第一卷积层，各 3×3 滤波器分别检测到水平边缘、垂直边缘、对角线等基本线条，形成多张“边缘特征图”。

2. 经第三卷积层，模型从边缘特征拼出“眼睛的圆形轮廓”“三角形猫耳”等中级特征，并通过池化层减小特征图尺寸。

3. 最后全连接层将高级特征组合后输出：猫的概率 0%96.3%，狗的概率 2.1%，其他 1.6%，成功完成识别任务。

六、深度学习训练的三大难题

深度学习界把调参训练模型叫做“炼丹”，因为效果充满不确定性。主要难在以下三个方向：

难题一：梯度消失（Vanishing Gradient）

在层数很深的网络里，反向传播的梯度信号每经过一层就乘以一个小于 1 的数，像公司里“指令传达十个部门后已经没人记得说了什么”。Sigmoid/Tanh 激活函数特别容易放大这个问题。

> **解决方案：** 换用 ReLU 激活函数，或者加入残差连接（让梯度跳过几层直接传递，如 ResNet）。

难题二：局部最优与鞍点

模型参数构成超高维地形，里面有无数梯度为零但不是最低点的“鞍点”，模型很容易误以为到达最优而停止更新——就像困在某个小山谷里出不去。

> **解决方案：** 用带动量的优化器（如 Adam）给梯度下降加“惯性”，帮助它冲过鞍点。

难题三：超参敏感（“炼丹”的来源）

学习率、批大小、网络层数、初始化方式……这些超参没有通用最优解，只能靠经验和大量实验摸索。但好消息是：使用预训练模型作为初始化，相当于“直接站在巨人袂膀上”，大幅降低了炼丹难度。

算法示例

1. 20 层使用 Sigmoid 的深度网络，反向传播时每层梯度乘以在 0.25，到第 1 层时梯度≈0.0000000001，参数几乎无法更新，第一层永远在“睡觉”。

2. 将激活函数换成 ReLU 后，梯度在正数区域等于 1，不衰减，第一层能正常接收更新并学习到有效特征。

3. 进一步引入残差连接（ResNet 思路），梯度可“跳过”多层直接回传，训练 100+层超深网络也不再有梯度消失问题。

七、序列模型的进化：从 RNN 到 LSTM

为什么需要序列模型?

文字、语音、股票价格……都有“顺序依赖”：知道了“今天天气晴”，再看到“明天”预测就更准。普通神经网络每次处理数据时会“忘掉”前面的内容，RNN（循环神经网络）专门为这种“记忆”需求而生。

RNN 应用模式

应用模式	输入	输出	场景举例
序列→类别	一段序列	单个标签	文本情感分类
同步序列→序列	序列	等长序列	词性标注
异步序列→序列	序列（编码器）	另一序列（解码器）	机器翻译

LSTM：给记忆开一条“高速公路”

LSTM 引入贯穿全序列的细胞状态（Cell State），以及三个“门”来控制信息流：

- 遗忘门：决定“要不要忘记”旧信息
- 输入门：决定“要不要加入”当前新信息
- 输出门：决定“要不要输出”当前隐藏状态

算法示例

1. 翻译“这本书是我上大学时在图书馆读过的”，RNN 处理到“图书馆”时已基本忘记开头“这本书”主语，导致翻译时主语混乱。

2. LSTM 通过遗忘门保留了“这本书”关键主语信息在细胞状态中，即使经过十几个时间步，信息仍然清晰传递到翻译输出阶段。

3. 最终 LSTM 能准确生成 "The book that I read in the library when I was in college"，正确匹配了句子首尾的语义关系。

八、Transformer：颧夏性的注意力机制

RNN 的根本局限：串行计算

不管是 RNN 还是 LSTM，处理序列时必须按顺序来——处理第 5 个词时，必须先处理完前 4 个。这导致训练无法并行，超长文本效果受限。2017 年 Google 提出 Transformer，彻底告别递归结构。

自注意力机制：每个词都能“直接看”其他词

Transformer 的核心是自注意力（Self-Attention）机制。处理文字时，每个词直接计算它与句子中其他所有词的“相关度”（注意力分数），并加权汇聚这些信息。

> **举例：** “它追了一只猫”中，“它”通过自注意力可以直接“关注”到“猫”和“追”，理解语义关系——而不是僄 RNN 一样等到读完全句才能回看。

Transformer 的四大关键组件

组件	功能
嵌入层（Embedding）	把每个词转化成向量，让计算机能做数学运算
位置编码（Positional Encoding）	告诉模型每个词是第几个（因为注意力不是顺序处理）
多头注意力（Multi-Head Attention）	同时从多个角度分析词语关系（语法/语义等）
前馈网络（FFN）	在注意力之后对每个位置独立做进一步变换

算法示例

1. “科学家告诉记者他们会通报结果”，Transformer 同时计算“他们”与“科学家”和“记者”的关联度，注意力并行计算无需逐词等待。

2. 多头注意力：一个头关注句子语法结构（主语），另一个头关注语义逻辑（谁有资格通报），综合判断“他们”指“科学家”（注意力分数更高）。

3. 由于计算完全并行，即使句子有 500 个词，Transformer 仍能在同一时刻计算所有词对之间的注意力，效率远高于 RNN/LSTM。

九、大语言模型（LLM）的崛起

从 Transformer 到大语言模型

Transformer 架构 + 海量数据 + 超强算力 = 大语言模型（LLM）。GPT、ChatGPT、千问、文心等模型，本质上都是超大规模的 Transformer。它们先在互联网海量文本上“预训练”，学到语言深层规律，再针对特定任务微调。

大语言模型训练三阶段

阶段	训练方式	目标
① 预训练	在万亿词互联网文本上自监督学习	学习语言的通用模式和知识（基础能力）
② 指令微调(SFT)	用高质量问答对有监督训练	教模型如何回答各类指令（任务能力）
③ RLHF 对齐	根据人类反馈做强化学习	让输出更安全、更有用（价值对齐）

涌现能力与幻觉：双刃剑效应

涌现能力（好事）： 当模型参数超过某个阈值（10B+），突然展现出训练时没有明确教过的能力，比如数学推理、写代码。就像水在 00°C 突然结冰——量变引发质变。

幻觉问题（坏事）： 模型有时“一本正经地胡说八道”，生成听起来合理但实际上错误的内容。本质是模型在预测“下一个词最可能是什么”，而不是真正“知道”事实。

算法示例

1. 同一问题“如何减肥”，模型生成三个答案；标注员根据安全性、准确性、实用性打分（A>B>C），训练“奖励模型”学会预测哪种回答更受欢迎。

2. 用强化学习（PPO 算法）让原始语言模型的输出尽量拿到高分，经过数千转 RLHF 迭代，模型输出变得更安全（不宣扬极端节食）、更实用。

3. 最终模型能给出结合运动和饮食的合理建议，同时也更符合人类表达习惯，幻觉问题有所减少，可用性大幅提升。

十、总结与展望：AI 时代的工程师转型

模型演化的历史脉络

每一代突破都是在解决上一代核心瓶颈：感知机解决“能否学习”，多层网络+反向传播解决“非线性表达”，CNN 解决“图像理解”，LSTM 解决“长序列记忆”，Transformer 解决“并行与全局建模”，大语言模型把这一切推向通用智能的边界。

工程师技能的三阶段演变

时代	核心技能	最大成本/价值
传统机器学习时代	手写规则 / 特征工程 / 数学推导	算法能力和数学建模
深度学习时代	架构设计 / 数据标注 / 炼丹调参	数据资产和模型架构设计
大模型时代	业务洞察 / 提示词工程 / 微调/RAG	场景理解和数据洞察力

未来展望

当前主流的“反向传播+梯度下降”范式虽然验证了十几年，但在优化精度和计算效率上仍有局限。前沿研究正在探索：高阶优化方法（利用二阶导数信息）、基于物理或生物机制的新型学习框架，以及用 AI 来设计更好的 AI 训练方法（元学习）。

AI 正在进化，它进化的方式也可能被 AI 加速。

> **给学习者的核心启示： AI 工具的崛起让基础编码门槛大降，但深刻理解数据、掌握业务场景、能与 AI 高效协作，才是这个时代最稀缺的能力。理解算法的本质——不是为了实现它，而是为了更好地使用它、评估它、信任它。**