人工智能的昨天、今天、明天

人工智能的 70 年历程

演讲者：耿子介

时间：2026 年 2 月

目录

昨天：苦涩的教训

昨天：苦涩的教训

70 年路程：从定义智能到深度学习突破

① 什么是「人工智能」？

② 人工智能的发展历程 · 三起两落 70 年

③ 深度学习的突破

什么是「人工智能」？

人工智能

Artificial Intelligence

如何人工实现?

工程路径

什么是智能?

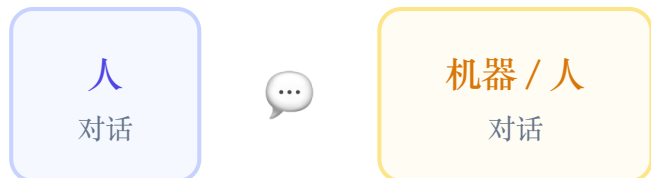
智能的定义

图灵之问：怎么给「智能」一个工程上的定义？

1950，「机器能思考吗？」与模仿游戏

① **时代背景：**1930–40 年代形式逻辑、可计算理论已成熟；「计算」有数学定义，但「智能」「思考」仍是哲学概念

② **图灵 1950：**《Mind》发表《计算机器与智能》——认为「机器能思考吗？」无意义，转而提出可操作的模仿游戏



判断：谁是人？

③ **图灵核心观点：**不定义「思考」而用可观测行为界定智能；无法可靠区分 → 行为上即达人类智能；智能 = 输入–输出行为模式

④ **延伸至今天：**「15 个人有 20 种定义 AGI 的方法」——但共识是 **智能 ≈ 能处理复杂任务**

达特茅斯会议：1956，人工智能作为学科的诞生

一、历史背景

40–50 年代神经生理学、控制论、可计算理论各自突破但彼此分散。McCarthy、Minsky、Shannon、Rochester 于 1956 年夏发起研究项目，将「**智能能否被形式化并在机器中实现**」集中讨论。

二、核心主张

> *「学习的各个方面或智能的任何其他特征，原则上都可以被精确描述，从而使机器能够模拟它。」*

智能 = 可形式化对象；目标 = 构造能推理、学习、自我改进的程序。**这是研究宣言：智能可成为工程与科学问题。**

三、命名：「AI」为何胜出

当时有控制论、自动机理论、复杂信息处理、机器智能等叫法，皆未胜出。McCarthy 选「AI」：制度化（提案→资助基础）；理论开放（不限定方法论）；修辞力量（直接提出「智能能否被人工实现」）。

达特茅斯会议虽未立即产出成熟技术，但首次明确提出：「人工智能」是可被系统研究的独立科学领域。

「人工」的那一半：三种实现智能的工程路线

智能有了定义，如何用工程手段造出来？

图灵回答了「什么是智能」，达特茅斯确立了学科边界；接下来是工程问题——如何造？

符号主义

Symbolicism · 拟人心

用规则和推理「写」出智能

规则与推理 · 知识表示 · 逻辑

代表：逻辑理论机、专家系统

连接主义

Connectionism · 拟人脑

让机器从数据中「学会」智能

数据与学习 · 表示学习 · 神经网络

代表：感知机、深度学习

行为主义

Behaviorism · 拟人身

在环境中通过奖励塑造行为

环境与奖励 · 试错 · 强化

代表：强化学习、进化算法

三条路线回答同一问题——如何用工程手段造出智能行为？

方法论不同，目标一致；后续七十年的突破多源于多派融合

符号主义：用规则和推理「写」出智能

核心思想：智能 = 显式知识 + 形式推理

一看就懂：专家系统如何「推理」？

规则 1：IF 患者**发热** ∧ **血培养阳性** → 疑似血液感染（置信度 0.7）
规则 2：IF 疑似血液感染 ∧ **革兰氏阴性** → 推荐**庆大霉素**（置信度 0.8）
↑ MYCIN（1972）用约 600 条这样的 IF-THEN 规则诊断血液感染，准确率达 65%，超过部分人类专家

关键里程碑

1956 **Logic Theorist** — 首次用程序证明数学定理，部分证明比原著更优雅
1965 **DENDRAL** — 从质谱推断分子结构，首个实用专家系统
1972 **MYCIN** — 用规则链诊断感染，开创基于规则的医学 AI
1980s **XCON** — 为 DEC 配置计算机，年省 $4000 万，专家系统商业化高峰

优点与局限

✓ 推理过程**透明可解释**，每一步有据可查
✓ 适合**形式化领域**（数学证明、医学诊断、棋类博弈）
✗ **组合爆炸**：国际象棋每步 35 种走法，10 步 → $35^{10} \approx 2.7\times10^{15}$
✗ **知识瓶颈**：XCON 规则从 2,500 膨胀到 17,000+，维护成本不堪重负
✗ **常识难符号化**：「水倒了会洒」需大量物理/因果知识才能形式化

核心洞察：智能可以被「写」出来——但世界的复杂度远超有限规则所能描述

案例：CYC 项目（1984） — 耗资 2 亿美元、累积 3,000 万条断言，却始终未展现通用智能。**常识的边界永远在后退，手写规则无法穷尽世界。**

连接主义：让机器「从数据中学会」智能

核心思想：智能 = 从数据中自动提取表示

一看就懂：神经网络如何「学习」？

人脑
860 亿神经元 → **突触连接** → 从经验中学习

≈

人工神经网络
数字神经元 → **权重参数** → 从数据中学习

输入 [猫的照片] → 第 1 层提取**边缘** → 第 2 层组合**纹理** → 第 3 层识别**部件**(眼/耳) → 输出 「**猫！**」

关键里程碑

1943 **McCulloch-Pitts** — 提出数学神经元模型，证明神经元可执行逻辑运算

1958 **感知机** — Rosenblatt 首次实现「机器从数据中学习」

1986 **反向传播** — Hinton 等提出 BP 算法，多层网络变得可训练，连接主义复兴

1998 **LeNet-5** — LeCun 用卷积网络识别手写数字，准确率超传统方法

2012 **AlexNet** — ImageNet 错误率从 26% 降至 16%，深度学习时代正式开启

优点与局限

✓ 无需手写规则，从数据中自动学习特征表示

✓ 感知类任务（图像、语音）远超符号方法

✓ 容错能力强，能从带噪声数据中学习

✗ **黑盒**：难以解释内部决策过程

✗ **数据饥渴**：对数据量和算力要求高，早期硬件不足是两次低谷主因

✗ **泛化有限**：分布外泛化仍是开放问题

核心洞察：与符号主义「手写规则」相反，连接主义相信**让数据说话**——同样的网络，喂不同数据就能完成不同任务

案例：AlexNet 时刻（2012） — Hinton 团队用 GPU 训练深度卷积网络，在 ImageNet 上将错误率骤降 10 个百分点，震惊学界，**直接引爆了持续至今的深度学习革命**。

行为主义：在环境中通过奖励塑造行为

核心思想：智能 = 与环境交互，最大化长期奖励

一看就懂：强化学习如何「训练」？

训练小狗		训练 AI 下棋
坐下 → 给零食 **(+1)** → 下次更愿意坐下	≈	赢棋 → 奖励信号 **(+1)** → 强化获胜策略
咬沙发 → 被训斥 **(-1)** → 下次减少咬沙发		输棋 → 惩罚信号 **(-1)** → 避免失败走法

关键里程碑

1948 **控制论** — Wiener 提出反馈控制的数学框架

1950s **动态规划** — Bellman 奠定最优序贯决策的理论基础

1992 **TD-Gammon** — 通过自对弈学会双陆棋，达到世界级水平

2013 **DQN** — DeepMind 用深度 Q 网络玩 Atari 超越人类，深度 RL 诞生

2016 **AlphaGo** — 击败李世石；2017 AlphaGo Zero 从零自对弈超越所有前代

优点与局限

✓ **无需标注数据**，从交互经验中学习

✓ 天然适合**序贯决策**（游戏、机器人控制、对话）

✓ 可与连接主义结合——**深度 RL** 是当前最强决策框架

✗ **奖励设计难**：稍有不慎就会 reward hacking

✗ **样本效率低**：AlphaGo 自对弈数百万局才达人类水平

✗ **安全性**与可预测性仍是开放挑战

核心洞察：RLVR（可验证奖励的强化学习）正是行为主义与连接主义的最新融合——今天的大模型对齐离不开它

案例：AlphaGo Zero（2017） — 不使用任何人类棋谱，仅靠自对弈 + 奖励信号，3 天超越所有人类棋手。**证明纯粹的「试错 + 奖励」就能涌现超人智能。**

三起两落：理想与工程条件的拉锯

热潮与寒冬的本质——理想超前于工程条件

① 第一次起落 · 推理期（1956–1974）

达特茅斯会议、Logic Theorist、SHRDLU、早期专家系统；过度承诺「20 年达人类智能」。目标落空——机器翻译笑话百出、定理证明发展乏力、计算力难突破。

寒冬（74–80）：感知机局限（《Perceptrons》）、Lighthill 报告、经费锐减

② 第二次起落 · 知识期（1980–1987）

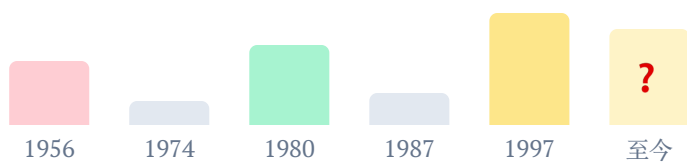
专家系统（Dendral、MYCIN）遍地开花，XCON 商业化；知识工程热，应用驱动型繁荣。

寒冬（87–93）：维护成本高、知识获取瓶颈、神经网络发展受阻、AI 硬件公司倒闭、政府投入缩减

③ 第三次浪潮 · 学习期（1997–至今）

统计机器学习（1990s–2012）将智能转化为概率建模与优化，为深度学习奠定方法基础。
随着算力提升、互联网推动 AI 不断创新与实用，取得一系列突破。
1997 Deep Blue；2012 ImageNet；2016 AlphaGo；2020 AlphaFold2；2022 后大模型爆发——深度学习与大数据兴起带来 AI 爆发。

是否会出现第三次寒冬?



共同逻辑：理想与承诺超前于工程条件（数据、算力、算法）的成熟速度

从统计机器学习到深度学习：原理的延续与跃迁

从最简单的模型出发：线性回归

核心思想：给定数据 `(x, y)`，找到函数 `y = f(x)` 拟合数据。最基础的形式——线性回归：

$$y = w_1x_1 + w_2x_2 + \ldots + w_nx_n + b$$

参数 w 和 b 就是模型需要「学习」的东西

🏠 例：预测房价

x = [面积, 楼层, 房间数, 到地铁距离, ...] → y = 房屋价格（万元）

若改为分类（涨/跌），套一层 Sigmoid 就变成逻辑回归

① 定义优化目标

损失函数 Loss：衡量预测与真实值的差距。如均方误差 `L = Σ(ŷ-y)²/n`，目标是让 Loss 尽可能小

② 求解参数

小规模可用解析解直接算；大规模则用梯度下降——沿 Loss 下降最快的方向逐步逼近最优 w, b

在此之上衍生出逻辑回归、SVM 等经典模型，共同点：**人工设计特征 + 浅层模型 + 数学优化**。

瓶颈：线性模型表达能力有限——现实世界的关系很少是线性的。如何突破?

从浅层到深层：为什么需要深度神经网络？

如果把多个线性函数叠加起来会怎样？

x → 线性层₁ → 线性层₂ → 线性层₃ → y

多层线性 = 一层线性（矩阵乘法可合并），表达力没有任何提升！

关键突破：**激活函数**——在每层之间插入**非线性变换**（如 ReLU、Sigmoid），打破线性限制。

x → 线性层₁ → σ → 线性层₂ → σ → 线性层₃ → y

线性层 + 非线性激活 = 深度神经网络 → 理论上可逼近任意连续函数（万能近似定理）

统计 ML（浅层）

数据 → [人工特征工程] → 浅层模型 → 输出

特征要靠人设计，模型容量有限

深度学习（多层）

数据 → [可学习的深层神经网络] → 输出

特征自动学习，端到端训练，容量可扩展

那么问题来了：这么多层、这么多参数，怎么训练？答案仍然是梯度下降——只不过需要一个高效计算梯度的方法：**反向传播（Backpropagation）**。

> 底层原理一脉相承：从线性回归到深度网络，本质都是「定义模型 → 定义损失 → 梯度下降优化参数」。深度学习的跃迁在于：模型从浅层线性变成深层非线性，表达力从有限到（近似）无限。

规模转折：深度学习的突破

数据、算力、算法三要素在规模上同时成熟

数据
ImageNet

算力
GPU

算法
ResNet

数据：ImageNet

李飞飞团队 2009 年发布，1,400 万张标注图片、2 万个类别。此前最大数据集仅数万张。ImageNet 大赛（ILSVRC）成为深度学习的竞技场——2012 年 AlexNet 以碾压之势夺冠，错误率从 26% 骤降至 16%，拉开深度学习时代的序幕。

算力：GPU 革命

神经网络的核心计算是矩阵乘法，GPU 天生擅长大规模并行运算。AlexNet 用 2 块 NVIDIA GTX 580，训练时间从 CPU 的数月缩短到数天。此后 NVIDIA 推出 CUDA 生态与专用 GPU（V100→A100→H100），算力成为 AI 的基础设施。

算法：ResNet（何凯明）

网络越深理论上越强，但实际中超过 20 层就因梯度消失而无法训练。2015 年何凯明提出残差连接（skip connection）：让每一层学习「残差」而非完整映射。ResNet 一举突破 152 层，错误率 3.57%，首次超越人类水平（5.1%）。

2012 AlexNet | 2014 VGG / GoogLeNet | 2015 ResNet | 2016 AlphaGo | 2017 Transformer | 2018 BERT / GPT | 2020 AlphaFold2

三要素同时到位后，深度学习证明：**通用学习 + 规模 = 超越手工系统**

「苦涩的教训」：通用学习与计算力的胜出

Rich Sutton，2019 年 3 月

Sutton 为强化学习奠基人之一；《The Bitter Lesson》总结 AI 七十年历史

> **核心论点：**「70 年 AI 研究得出的最大教训是：**利用计算的通用方法最终最有效，且优势巨大。**」长期看，唯一重要的是对算力的利用——摩尔定律使算力成本持续下降。

手工知识
短期有效，长期平台

→ 长期胜出

通用方法 + 算力
搜索与学习

苦涩教训四步：① 研究者塞进人类知识 ② 短期有效 ③ 长期平台期 ④ 突破来自搜索与学习放大计算

例证：国际象棋（1997 深度搜索）、围棋（2016 搜索+自对弈）、语音（hidden Markov models 胜音素知识）

> 「我们想要的 AI 是能像我们一样发现的智能体，而不是塞进我们已经发现的东西。」

今天：压缩即泛化，泛化即智能

今天：压缩即泛化，泛化即智能

语言成为通用接口，Chat → Reasoning → Agent

① GPT 技术核心原理

② 从 Chat 到 Reasoning 再到 Agent（OpenAI 5L 演进框架）

③ 大模型竞争现状：2026 春节档发布潮与应用大战

ChatGPT：大模型第一次被「看见」

- **2022 年 11 月 30 日**发布，两个月破亿用户，**史上增长最快的消费级应用**
- 对话式界面，几乎零学习成本
- 能理解模糊、不完整、甚至带歧义的自然语言指令
- 能在多轮对话中保持上下文

为什么是「语言模型」首先爆发，而不是别的什么？

语言成为通用接口

> *「我的语言的边界意味着我的世界的边界。」*
>
> —— 维特根斯坦 · 逻辑哲学论 (1921)

> *「词语的意义就是它在语言中的用法。」*
>
> —— 维特根斯坦 · 哲学研究 (1953)

语言为什么是「通用接口」？

◆ **知识的载体**
人类知识、规范、制度、文化，绝大多数由自然语言承载
掌握语言 = 拿到人类世界的接口

◆ **行动的工具**
代码也是语言；写代码 = 用语言改造现实世界
软件正在吞噬一切

◆ **协议与共识**
法律、合同、API、通信协议……
语言本身就是人与机器之间的协议层

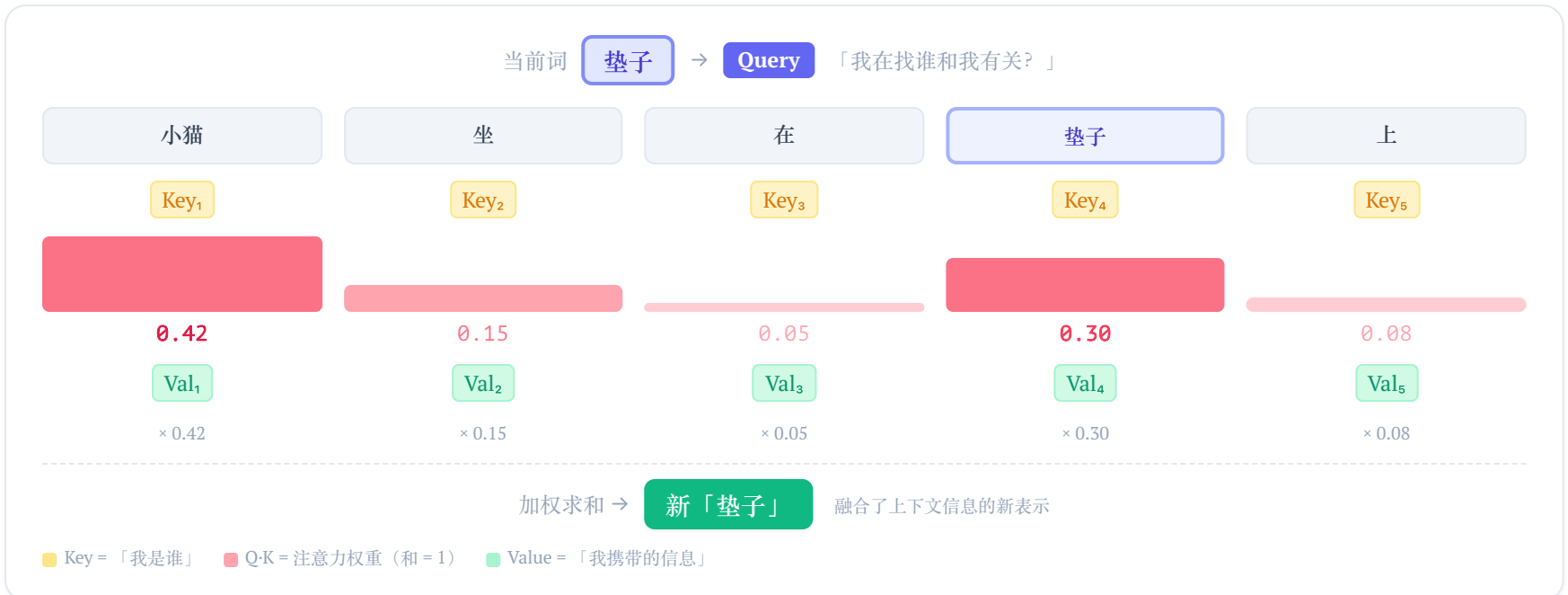
◆ **天然的训练信号**
互联网有近乎无限的文本，天然适合大规模无监督学习
「预测下一个词」就能涌现理解能力

语言既是最大的知识库，也是最好的训练数据 —— 接下来看如何用深度学习把它「学会」

Transformer 核心：Self-Attention

直觉：理解一个词时，你会自动关注句中其他相关的词 —— Self-Attention 让模型做同样的事

2017 · Vaswani et al. · *Attention Is All You Need*

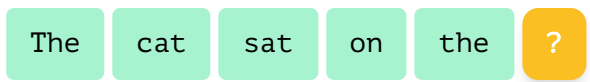


$$\text{Attention}(Q, K, V) = \text{softmax}(Q \cdot K^T / \sqrt{d_k}) \cdot V$$

GPT（Generative Pre-trained Transformer）的核心：自回归语言建模

一个简单到优雅的任务设定——预测下一个词

任务：给定前 t 个 token，预测第 t+1 个 token



← 前文信息汇聚 → 预测下一个词

$$P(x_{t+1} \mid x_1, \ldots, x_t)$$

条件概率

$$L = -\Sigma \log P(x_t \mid x_{<t})$$

负对数似然损失

① 看似只是「预测下一个词」，但要预测好，模型必须理解语法、语义、逻辑、事实知识、推理链条……

② 训练数据 = 整个互联网文本（数万亿 token），不需要人工标注

③ 一个统一目标函数 → 翻译、问答、摘要、代码、数学全部纳入

GPT 训练全流程：Pre-training → Post-training

从「会说话」到「说人话」再到「说有用的话」

Pre-training

预训练

海量无标注文本 · 自回归语言建模

数万亿 token · 数百亿到千亿参数

= 博学但不受控的续写机器

→

Post-training 后训练

SFT 指令微调

人工编写「指令—回答」对

学会遵循用户意图

「接话」→「完成任务」

RL 强化学习对齐

RLHF：奖励模型学习人类偏好

RLVR：可验证奖励（数学/代码对错）

「完成任务」→「对齐人类价值」

RLHF — 人类标注偏好排序 → 训练奖励模型 → PPO 优化 | 适合开放式任务

RLVR — 规则判定对/错作为奖励信号 | 适合数学、代码等可验证任务（如 DeepSeek R1）

三阶段的本质：通用能力 → 任务遵循 → 价值对齐 | 从「每任务一模型」→「一个大模型 + 不同提示完成不同任务」

为什么 GPT 有效：压缩即泛化，泛化即智能

Ilya Sutskever 的核心洞察

OpenAI 联合创始人、首席科学家 Ilya Sutskever 的核心洞察——「压缩即泛化，泛化即智能」

用一个数列理解「压缩」

$\{1, 4, 9, 16, 25, \ldots\} \rightarrow a_n = n^2$

「记住」整个数列需要无限存储；但发现 $a_n = n^2$ 就压缩了全部信息

你可以预测 $a_{100} = 10000$，即使从未「见过」这个点 → 压缩 = 发现规律 = 泛化

世界 → 有损压缩 语言 → 有损压缩 模型参数

压缩的每一环都在提取「本质」

压缩即泛化，泛化即智能

—— Ilya Sutskever

GPT 系列（上）：GPT-1 与 GPT-2

预训练范式的建立与规模涌现的发现

GPT-1 2018 年 6 月 · 1.17 亿参数 · 12 层

论文：*Improving Language Understanding by Generative Pre-Training*

核心突破：首次证明「生成式预训练 + 判别式微调」两阶段范式有效

意义：确立 pre-train → fine-tune 范式，为后续所有 GPT 奠基

GPT-2 2019 年 2 月 · 15 亿参数 · 48 层

论文：*Language Models are Unsupervised Multitask Learners*

核心突破：规模上去后 **zero-shot 能力涌现**——不经微调，仅凭提示就能做翻译、问答、摘要

关键事件：OpenAI 以「担心滥用」最初仅发布小版本，引发 AI 安全与开放性讨论

论文标题即核心观点：语言模型天然就是无监督的多任务学习器

GPT 系列（中）：GPT-3 与 In-Context Learning

提示即编程，推理即学习

GPT-3 2020 年 5 月

规模：**1750 亿参数**，96 层，~3000 亿 token

论文：*Language Models are Few-Shot Learners*

核心突破：

- In-Context Learning：不更新参数，仅在 prompt 中给出几个示例，模型就能「临时学会」新任务
- Zero-shot / One-shot / Few-shot 三种模式，规模越大效果越好
- Scaling Law：1.5B → 175B，能力发生质变

In-Context Learning 示意

```
示例1：猫 → cat
示例2：狗 → dog
示例3：鸟 → bird
---
输入：鱼 → fish
```

无需更新参数，prompt 即程序

标题的核心观点：**语言模型是少样本学习器**，不需要任务特定的海量标注

Prompt Engineering 从此成为重要课题

GPT 系列（下）：InstructGPT 与 ChatGPT

从「能力涌现」到「对齐人类偏好」再到「产品化爆发」

InstructGPT 2022.3

论文：*Training language models to follow instructions with human feedback*

首次系统性将 RLHF 应用于大语言模型

13 亿参数 InstructGPT > 1750 亿原版 GPT-3（人类偏好评估）

证明「对齐比规模更重要」

路径：SFT → Reward Model → PPO

ChatGPT 2022.11.30

基于 InstructGPT 方法论 + GPT-3.5

突破不在技术，而在产品与交互：

- Chat UI + 多轮上下文管理
- 更强安全策略与内容审核
- 海量用户反馈快速迭代

让大模型从实验室走进千家万户，两个月破亿

1 GPT-1 2018.6

2 GPT-2 2019.2

3 GPT-3 2020.5

I Instruct 2022.3

C ChatGPT 2022.11

不是一夜冒出来，而是五年持续迭代

Scaling Law：大力出奇迹

规模增长带来可预测的性能提升——大模型时代最核心的经验法则

OpenAI 2020 Kaplan et al., *Scaling Laws for Neural Language Models*

$$L(x) \propto x^{-\alpha}$$

x = N（参数量）/ D（数据量）/ C（计算量），损失随规模幂律下降

N 参数量	D 数据量	C 计算量
α ≈ 0.076	α ≈ 0.095	α ≈ 0.050

Chinchilla 2022 Hoffmann et al., DeepMind

修正 Kaplan 结论：给定算力预算，参数和数据应**等比例扩展**（N ∝ D）。GPT-3（175B）其实数据量不够——同等算力下，**更小模型 + 更多数据**效果更好。

「涌现」：真的还是幻觉？

2022 Wei et al.：某些能力在规模突破阈值时**突然出现**——被称为「涌现能力」

2023 Schaeffer et al. 反驳：涌现是评估指标的假象——换用连续指标后，能力随规模平滑增长，并无突变

启示：Scaling Law 本身是平滑的，「突变」更多源于离散指标的测量偏差

Training Scaling → Test-time Scaling

Training Scaling：更大模型、更多数据、更长训练

→ 算力成本指数增长，收益递减（撞墙）

Test-time Scaling：推理时让模型多「想一会儿」

→ 增加推理计算 = 思考更深、搜索更广

把算力从训练阶段部分转移到推理阶段，打开全新的 Scaling 维度

Scaling Law 不仅指导了训练范式，也催生了 Reasoning 模型——当 Training Scaling 趋近天花板，Test-time Scaling 成为新前沿

从 Chat 到 Reasoning 再到 Agent：OpenAI 五级 AI 框架

2024 年 7 月，OpenAI 内部披露的 AGI 进展衡量体系

L5 组织者 **Organizations** 统筹多智能体，运行整个组织

L4 创新者 **Innovators** 产出原创性成果——全新科学理论、突破性创意

L3 智能体 **Agents** 代表用户自主行动，多步骤任务 | Codex · Claude Code · Openclaw ⬅️ 迅猛发展

L2 推理者 **Reasoners** 解决博士级复杂问题 | o1 · o3 · DeepSeek-R1 ✅ 基本实现

L1 对话者 **Conversational AI** 自然语言对话、多轮交互 | ChatGPT · GPT-4 ✅ 已实现

我们正从「单轮对话」走向「可插入真实工作流的系统」

Reasoning 模型关键里程碑

从 Prompt 技巧到原生推理能力

2022.01 **Chain-of-Thought Prompting** Wei et al., NeurIPS 2022
在 prompt 中加入推理步骤示例，540B 模型在 GSM8K 数学基准达 SOTA

2022.05 **Zero-shot CoT** Kojima et al.
仅添加「Let's think step by step」即可激发推理，无需示例

2024.09 **OpenAI o1-preview / o1-mini**
首个原生推理模型，模型自主「内部思考」而非依赖 prompt 技巧

2025.01 **DeepSeek-R1** 深度求索
开源推理里程碑，RLVR + GRPO 算法，训练成本仅 ~$294K

2025.04 **OpenAI o3 + o4-mini**
比 o1 减少 20% 重大错误，支持 agentic 多工具调用

从「prompt 技巧」到「模型原生能力」，推理正在成为大模型的标配

Reasoning 模型：让 AI「想一想再回答」

Test-time Scaling 的直觉理解

Reasoning 模型在给出最终回答前，先进行一段**内部思考过程**（`<think>...</think>`），类似人类在纸上打草稿

输入输出示例

A 管 6 小时灌满，B 管 4 小时灌满，同时开几小时灌满?

```
<think>
  A 每小时 1/6，B 每小时 1/4
  合计 = 1/6 + 1/4 = 5/12
  需要 12/5 = 2.4 小时
</think>
```

同时开 **2.4 小时**（2 小时 24 分钟）灌满。

普通 ChatBot

直接输出答案（常出错）

Reasoning 模型

先推理再回答（准确率大幅提升）

Agent：会「自己干活」的 AI

Agent = LLM + Context + Tools

用户指令 → Agent 思考 → 调用工具 → 获取结果 → 继续/回答

感知 → 思考 → 行动 → 反馈（闭环）

Chatbot 输出 **文本**，Agent 输出 **行动**

Agent 实战：一个真实的工具调用示例

从用户请求到自主完成任务的全过程

U

帮我看 `data.csv` 中销售额最高的产品，生成柱状图保存到 `output.png`

🧠

思考：需要读取 CSV → 分析数据 → 生成图表 → 保存文件

🔧

`tool_call: read_file("data.csv") → 返回 CSV 内容`

🔧

`tool_call: execute_code("import pandas; ...plt.savefig('output.png')") → 执行成功`

A

销售额最高的是**产品 A**（¥128,500），柱状图已保存到 `output.png`。

Agent 的核心：自主规划 + 工具调用 + 结果反馈循环

Agent 前沿：原生基座 × Agent 框架

从模型能力到应用生态的全面爆发

原生 Agent 基座 — Function Calling / Tool Use 已成标配

GPT · Claude · Gemini · DeepSeek · Qwen · Kimi · Seed · ...

主流模型（闭源 + 开源）均已内置结构化工具调用能力，Agent 不再依赖 prompt hack

Agent 框架与应用

Coding Agent

Cursor
AI IDE · 多 Agent 并行 · Background Agent

Claude Code
终端原生 · Subagent · 15 万词上下文

Codex
OpenAI 异步编程代理 · 云端沙盒

Gemini CLI
Google 开源终端 Agent · Gemini 驱动

通用 Agent

Manus
规划 + 执行 + 验证 · 全自主多步骤

Kimi Agent Swarm
100 子 Agent 并行 · PARL 技术

Claude Cowork
桌面自主 Agent · 编码 + 知识工作

OpenClaw
开源 20 万 ⭐ · 本地部署 · 技能扩展

开发者工具链

开发框架
LangGraph · CrewAI · AutoGen · OpenAI Agents SDK

低代码平台
Dify（开源）· Coze 扣子（字节）

协议与标准
MCP（Agent ↔ 工具）· A2A（Agent ↔ Agent）· Skills

2026 春节档：史上最密集的 AI 发布潮

模型、应用、生态的全面竞赛

📅 1 月中旬 – 2 月上旬

🇺🇸 **Claude Cowork** Anthropic · 1.12
非编码 Agent，自主操作本地文件与数据

🌐 **OpenClaw** 开源社区 · 1.25
开源 AI 助手 / Agent 平台，200K GitHub Stars

🇨🇳 **Qwen3-Max-Thinking** 阿里 · 1.25
万亿参数推理模型，260K 上下文，自适应工具调用

🇨🇳 **Kimi K2.5** 月之暗面 · 1.27
万亿参数 MoE，Agent 集群并行调度，开源

🇨🇳 **Step 3.5 Flash** 阶跃星辰 · 2.2
196B/11B MoE，开源，2 天登顶 OpenRouter

🇺🇸 **Claude Opus 4.6** Anthropic · 2.5
1M token 上下文，Terminal-Bench 2.0 SOTA

🇺🇸 **GPT-5.3-Codex** OpenAI · 2.5
SWE-Bench Pro 首破 50%，首个由 AI 参与构建的模型

📅 2 月中旬 · 春节档高潮

🇨🇳 **GLM-5** 智谱 · 2.12
745B/44B MoE，全昇腾训练，编程接近 Opus 4.5

🇨🇳 **Seedance 2.0** 字节 · 2.12
AI 视频生成最强，音画一体，60 秒出片

🇨🇳 **MiniMax M2.5** 2.13
229B/10B MoE，SWE-bench 80.2%，Lightning 100 TPS

🇨🇳 **Seed 2.0** 字节 · 2.14
四款大模型（Pro/Lite/Mini/Code），定价仅竞品 1/10

🇨🇳 **Qwen 3.5** 阿里 · 2.16
Agentic AI 时代，视觉 Agent，成本↓60%

🇨🇳 **DeepSeek V4** 深度求索 · 预计 2 月
mHC + Engram 架构，上下文已升 1M，蓄势待发

竞争重心从「参数规模」转向「**AI 编程 + 智能体能力**」

C 端 AI 应用的「春节红包大战」

模型之争的背后是用户入口之争

字节豆包

ByteDance

春晚 + 10 亿+

10 万份科技好礼（机器人/无人机/电车）

DAU 破亿 · 留存率 44.5%

阿里千问

Alibaba

30 亿免单

AI 点奶茶 1 分钱 · 9h 破 1000 万单

DAU ↑727% · 登顶 App Store

腾讯元宝

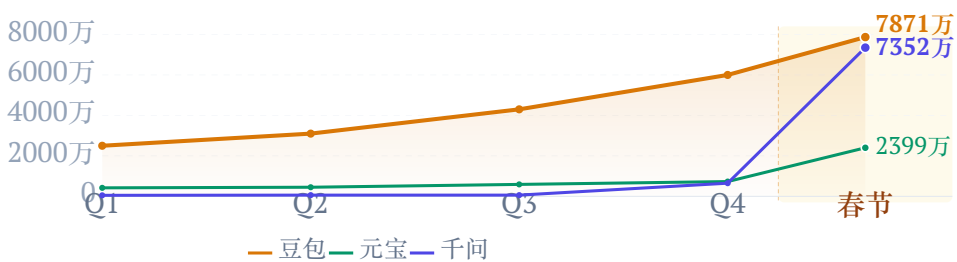
Tencent

10 亿红包

微信搜「元宝」触发福袋红包

DAU ↑210% · 年投流 150 亿

DAU 趋势：2025 季度均值 → 2026 春节峰值（QuestMobile / AppGrowing）



春节 DAU 峰值（2026.2.7）

豆包	7871 万
千问	7352 万 ↑10.4x
元宝	2399 万 ↑3.1x

30 日留存率（2025 年均值）

豆包	44.5%
元宝	30.1%
千问	23.5%

深层目标：不是「撒钱」，而是争夺下一代 AI 入口——培养用户「遇事找 AI」的心智

明天：人工智能的下半场

明天：人工智能的下半场

从「能不能造出来」到「该让它做什么」

① Agent 大规模应用：从开发者工具到普惠 Agent

② 关键技术突破：自进化、多模态、具身智能、评估部署

③ 社会伦理与治理：就业、治理与个人应对

AI 的「下半场」：从 Benchmark 到真实世界

姚顺雨（OpenAI 研究员）的关键洞察

2025 年 4 月，OpenAI 研究员姚顺雨（ReAct 框架作者、清华姚班 + 普林斯顿博士）发表博文《*The Second Half*》，提出 AI 发展已进入「下半场」

📖 上半场：训练与 Benchmark

核心命题：「能不能造出来」

发明新方法（Transformer、预训练、RLHF、Scaling Law），在 benchmark 上不断刷新纪录

英雄是算法研究者

🚀 下半场：评估与落地

核心命题：「该让它做什么、怎么用好它」

不再问「能不能解决 X」，而是「应该做什么，如何衡量真正的进步」

需要产品经理思维，而非纯技术思维

姚顺雨的洞察：上半场过度聚焦算法创新，现有 benchmark 脱离现实——任务被假设为独立的、AI 自主完成的，但真实世界高度关联、需要人机交互

Agent 的大规模应用：从 Coding Agent 到普惠 Agent

Agent 正在从开发者工具走向大众应用

阶段一：Coding Agent

开发者专属

产品：Cursor · Codex Agent · Claude Code

场景：在 IDE / 终端中工作

能力：自主编程、代码审查、Bug 修复

面向程序员，需要技术背景

→

阶段二：Local Work Agent

本地工作助手

产品：Claude Cowork (2026.2)

场景：操作本地文件、桌面应用

能力：Think-See-Act（推理 + 视觉 + 操控）

面向知识工作者

→

阶段三：Personal Agent

本地运行、消息即用

产品：OpenClaw（GitHub 10 万+ ⭐）

场景：WhatsApp / Telegram / 微信对话直接调用

能力：3500+ 社区技能 · 本地记忆 · 多模型切换

面向普通大众

Agent 应用不断下沉：开发者 → 知识工作者 → 普通大众

Agent 部署的关键挑战：权限管理与协作协议

当 AI 开始「动手做事」，信任与边界成为核心问题

🔐 权限管理

- Agent 需要访问文件系统、执行代码、调用 API、操控桌面——权限粒度如何设计？
- **沙盒隔离**：限制可执行的操作范围，防止越权
- **分级授权**：只读 → 有限写入 → 完全控制，按任务授权
- **安全审计**：每一步操作可追溯、可回滚

案例：Claude Agent SDK Sandbox · Cursor 沙盒机制

🤝 协作与雇佣机制

- **三种协作模式**：监督式（人审批每步）→ 委托式（人定目标）→ 团队式（多 Agent 协作）
- **Agent「雇佣」问题**：如何衡量工作质量？如何定价？如何问责？
- **协议标准化**：MCP 作为通用协议；未来需要 Agent-to-Agent 协议

展望：权限与协作框架将成为 AI 基础设施的核心

Agent 不只是技术问题——**信任、权限、协作协议**是规模化落地的前提

延伸阅读：《从 Moltbook 看 AI Agent 的权限、协作与雇佣》

AI 未来亟待突破的关键技术

四个方向决定 AI 的下一个十年

评估与部署

- ⚠️ 核心痛点：Benchmark 高分 ≠ 生产可用，评估与部署存在巨大鸿沟
- 📐 新方向：真实场景评估（如 OpenAI GDPval）、持续动态评测
- 🔒 部署难题：多步推理漂移、工具调用错误、长链路可靠性不足

多模态能力深化

- 🎬 长视频生成：时空一致性、物理规律遵循
- 🔍 长视频理解：跨事件长程依赖、细粒度时序建模
- ⚡ 实时多模态感知：流式处理、无限长视频流的低延迟理解

自进化与持续学习

- ⚠️ 核心痛点：灾难性遗忘——学新忘旧，能力持续退化
- 🧠 记忆机制：短期上下文推理 + 长期轻量微调，双轨协同
- 🧪 经验进化：从交互反馈中自我蒸馏、迭代优化策略

具身智能

- ✅ 小脑已成熟：运动控制、轨迹规划、Sim2Real 趋于可用
- 🧩 核心瓶颈：大脑与小脑的实时连接——推理决策 ↔ 运动执行
- 🎯 关键目标：低延迟感知-决策-执行闭环，提升实时交互能力

评估与部署：Benchmark 高分 ≠ 生产可用

弥合评估与部署之间的巨大鸿沟

📏 评估：从刷榜到真实场景

困境：传统 benchmark（MMLU、GSM8K）已被刷榜，分数高 ≠ 实际好用

新范式① 经济价值驱动评估——OpenAI GDPval：用真实工作任务替代学术数据集

新范式② 动态真实任务评估——让模型修真实 Bug

新范式③ 持续动态评测——部署后持续监控，而非一次性打分

🚀 部署：从实验室到生产环境

多步推理漂移：长链路任务中输出变异率高达 20–30%，错误逐步累积

工具调用可靠性：函数调用错误、参数幻觉在生产环境频发

推理成本与延迟：量化、蒸馏、推测解码是降本的关键技术

可观测性：部署后行为监控、退化检测、快速回滚

核心认知：LLM 可靠性是系统工程问题，而非单纯的模型问题——需要评估、监控、回滚全链路闭环

多模态能力深化

从「能看能听」到「长时理解与实时感知」

当前基础： GPT-4o / Gemini 多模态理解 · Sora / Seedance 视频生成 · Whisper 语音 · 原生多模态模型涌现

长视频生成

从秒级片段到分钟级连续叙事

痛点：时空一致性、物理规律遵循、角色/场景连贯

方向：时序扩散架构、长程注意力机制

长视频理解

从短片分析到小时级视频的精准理解

痛点：跨事件长程依赖、细粒度时序建模、Token 效率

方向：时序编码器 + 状态空间模型、高效 Token 压缩

实时多模态感知

边看边说边理解的流式多模态处理

痛点：感知与生成耦合、无限流的内存爆炸

方向：并行流式架构、解耦位置编码

关键挑战： 多模态对齐与幻觉——不同模态信息的准确融合，以及多模态场景下更严重的幻觉问题

自进化与持续学习

AI 如何像人一样「越用越聪明」？四个未解难题

🧠 记忆机制：Context Window 之外怎么办？

上下文窗口终究有限，Memory 是核心问题

短期记忆靠上下文，长期记忆靠什么？——本地文件？向量库？参数化存储？

如何在有限窗口中高效检索、压缩、遗忘不重要的信息？

📥 知识获取：训进网络 vs 外部知识库？

新知识该微调进参数还是放在外部检索？行业尚无定论

Fine-tuning：深度整合但成本高，且面临灾难性遗忘风险

RAG / 外部知识库：灵活易更新，但检索质量与深度推理能力受限

📦 自动积累：如何把经验沉淀下来？

系统能否自动将交互中学到的内容持久化为可复用知识？

从对话反馈中提炼策略、从错误中总结规则、从成功中固化流程

关键：自动化程度——人工标注不可扩展，纯自动又难保质量

👥 群体进化：多 Agent 如何共享能力？

单体学习之上，多个 Agent 之间能否共享技能、协同进化？

类似 Skills 机制——一个 Agent 学会的能力，其他 Agent 直接复用

挑战：技能的标准化表示、跨场景迁移、版本管理与质量控制

本质问题：当前的 AI 是「无状态的工具」——用完即忘。要成为真正的智能体，必须具备记住、学习、积累、共享的能力闭环

具身智能：AI 从数字世界走进物理世界

小脑已就绪，关键在于大脑与小脑的连接

✅ 小脑：运动能力趋于成熟

- 运动控制：灵巧操作、双足/四足运动已达高水平
- 轨迹规划：RL 驱动的底层控制策略日趋稳定
- Sim2Real 迁移：仿真到真实环境的迁移方法逐步成熟

代表：Figure · Tesla Optimus · 宇树科技

🧩 大脑 ↔ 小脑：核心瓶颈

- 高层推理太慢：LLM 推理延迟 vs 物理世界毫秒级响应要求
- 决策-执行鸿沟：抽象语义规划难以精确转化为连续运动指令
- 实时闭环不足：感知-决策-执行需要低延迟全链路协同

方向：分层架构（大脑粗规划 + 小脑快执行）、多级记忆系统

核心观点：小脑的运动能力不再是瓶颈——打通大脑与小脑的实时连接，才是具身智能突破的关键

AI 未来：有泡沫，但承诺终将兑现

人总是高估技术一年的进展，却低估技术十年的发展

长期确定性：AI 将全面渗透

① AI for Coding：从辅助补全到自主编程，软件开发效率 10 倍提升已在发生

② AIGC：文章、视频、设计、音乐——内容创作的边际成本趋近于零

③ AI for Science：药物发现、材料设计、蛋白质结构预测——科研范式正在被重写

④ AI Personal Assistant：人人拥有万能私人助理，对话即可完成复杂任务

短期泡沫：冷静看待

能力边界：幻觉、推理不可靠、长链路任务失败率高——离「通用智能」仍有距离

商业化落差：投入巨大但盈利模式尚未清晰，大量 AI 创业公司面临洗牌

期望过载：市场宣传远超实际能力，用户期望与体验之间存在落差

泡沫会破，但技术不会倒退

今天的每一个承诺，十年内都会以某种形式兑现

核心判断：AI 是这个时代最大的确定性——不是要不要拥抱的问题，而是**如何拥抱、何时拥抱**的问题

AI 与个人：冲击已来，如何应对？

不是「AI 取代人」，而是「会用 AI 的人取代不会用的人」

⚡ 就业冲击：哪些正在发生

- **重复性工作加速替代**：客服、数据录入、基础翻译、初级编程
- **中间层被压缩**：高技能与低技能需求增长，中等技能岗位「两极化」
- **技能迭代加速**：半衰期从十年缩短到两三年，传统教育来不及响应
- **新岗位涌现**：Prompt Engineer、AI 训练师、Agent 运维、AI 安全审计

🧭 个人行动指南

1. **拥抱而非回避**：主动学习和使用 AI，让它成为你的「外挂」而非对手
2. **培养 AI 协作力**：提出好问题、做好判断、创造性地整合 AI 输出
3. **找到不可替代性**：领域专长 + AI = 超级个体，创业门槛前所未有地低
4. **持续学习**：唯一不变的是变化本身，保持好奇心和学习力才是最大的护城河

AI 不会取代人类，但会用 AI 的人会取代不会用的人

AI 与社会：技术在跑，制度在追

就业治理与伦理规范，决定 AI 走向何方

⚠ 核心治理议题

- **技术滥用与数据安全**：深度伪造、版权争议、个人数据被模型「记住」
- **价值对齐与问责**：模型对齐谁的价值观？AI 造成损害时责任如何归属？
- **就业冲击与贫富分化**：AI 加速替代中等技能岗位，收益向资本与技术精英集中
- **产能过剩风险**：边际成本趋零带来供给爆炸，如何避免产能过剩冲击整个经济循环
- **生产关系滞后**：生产力跃升后，现有的分配体系、雇佣关系、社会保障亟需重构

🏛 政策与制度应对

全球治理框架：

欧盟 AI Act（2024）· 中国《生成式 AI 服务管理暂行办法》（2023）· 联合国 AI 治理高级别咨询机构

- **人社部 2026.1**：将出台《应对 AI 影响促就业文件》——中央首次专门针对 AI 就业出台政策
- **国务院 2025.8**：《"人工智能+"行动意见》——从「替代」转向「赋能」
- **配套措施**：五项 AI 技能培训行动 · 新就业形态权益保障 · 职业伤害保障全国扩围

> AI 带来的不只是技术革命，更是一场深刻的社会经济结构变革——当生产力足够强大时，真正的挑战是让每个人都能从中受益

谢谢

欢迎提问与交流