



大模型技术与发展



张倬胜

上海交通大学计算机学院

zhangzs@sjtu.edu.cn

<https://bcmi.sjtu.edu.cn/~zhangzs>

目录

- 大模型概述
 - 发展概览
 - 概念范畴
- 预训练模型
 - 主要类型
 - 模型架构
- 大规模模型
 - 范式变化
 - 主要代表
 - 关键技术
 - 部署方式
- 概念延伸
 - 垂直大模型
 - 多模态基座
 - 智能体基座



目录

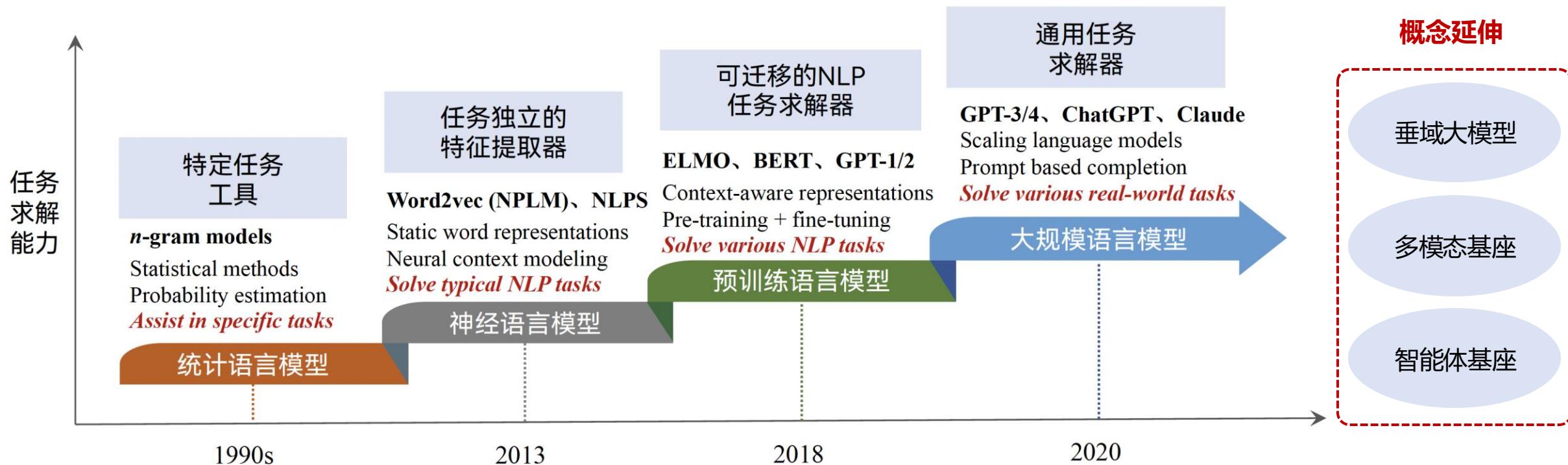


- 大模型概述
 - 发展概览
 - 概念范畴
- 预训练模型
 - 主要类型
 - 模型架构
- 大规模模型
 - 范式变化
 - 主要代表
 - 关键技术
 - 部署方式
- 概念延伸
 - 垂直大模型
 - 多模态基座
 - 智能体基座



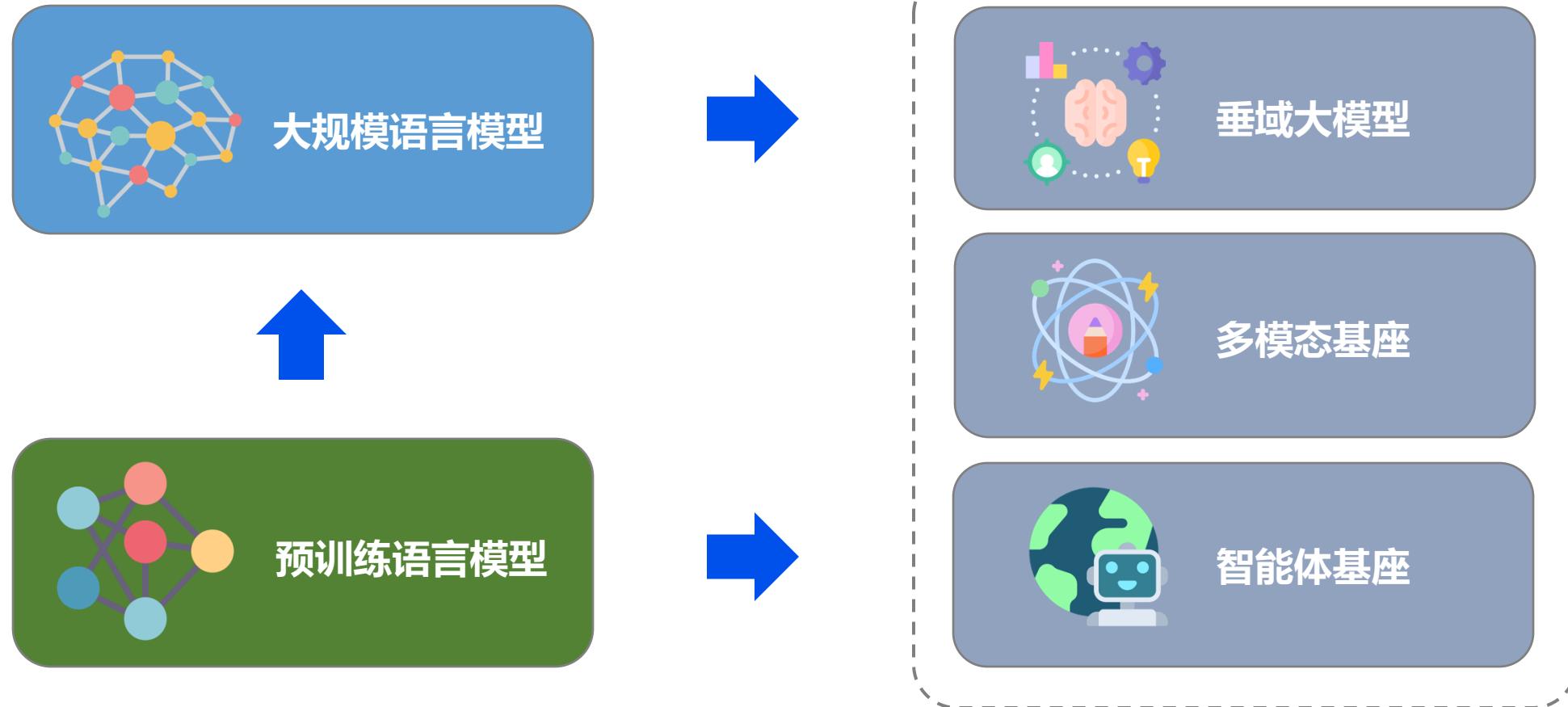
人工智能的发展

- 人工智能技术经历了小数据到大数据，小模型到大模型，专用到通用的发展历程，正逐步进入**大模型时代**
- 在海量无标注数据上进行**大规模自监督预训练**，学习到大量的**语言知识与世界知识**
- 通过**自然语言交互**完成多种任务，具备了多场景、多用途、跨学科的任务处理能力



[1] Zhao, Wayne Xin, et al. A Survey of Large Language Models. arXiv preprint arXiv:2303.18223.

大模型的范畴



目录



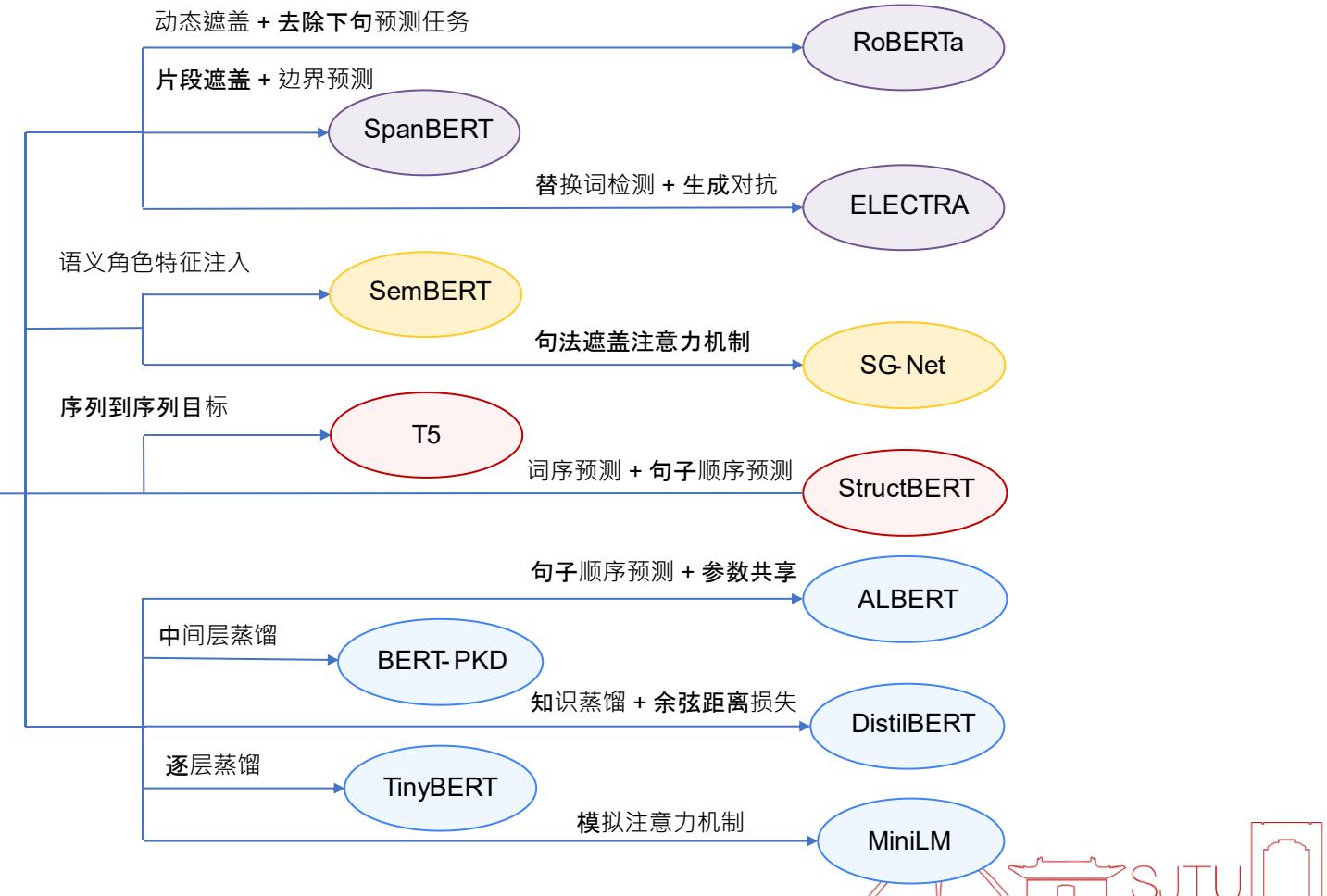
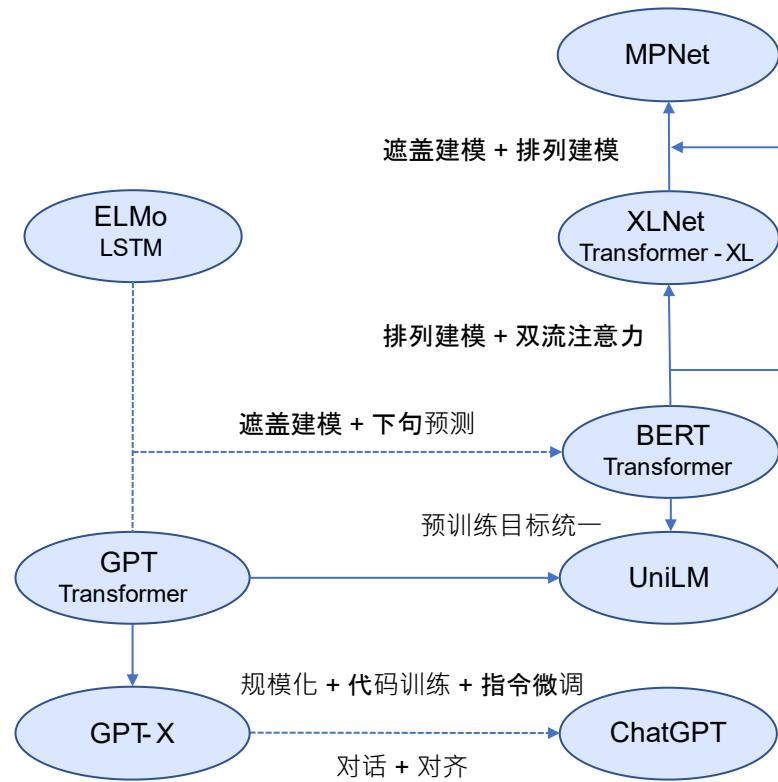
- 大模型概述
 - 发展概览
 - 大模型范畴
- 预训练模型
 - 主要类型
 - 模型架构
- 大规模模型
 - 范式变化
 - 主要代表
 - 关键技术
- 概念延伸
 - 垂直大模型
 - 多模态基座
 - 智能体基座



预训练语言模型

□ 语言智能新范式—预训练语言模型（代表：ELMo、BERT、GPT-1/2）

- 大规模数据上**自监督预训练**，经**微调**后适配各类任务，有效利用**大数据、大模型和大算力**
- 主要应用于语言解析和理解任务

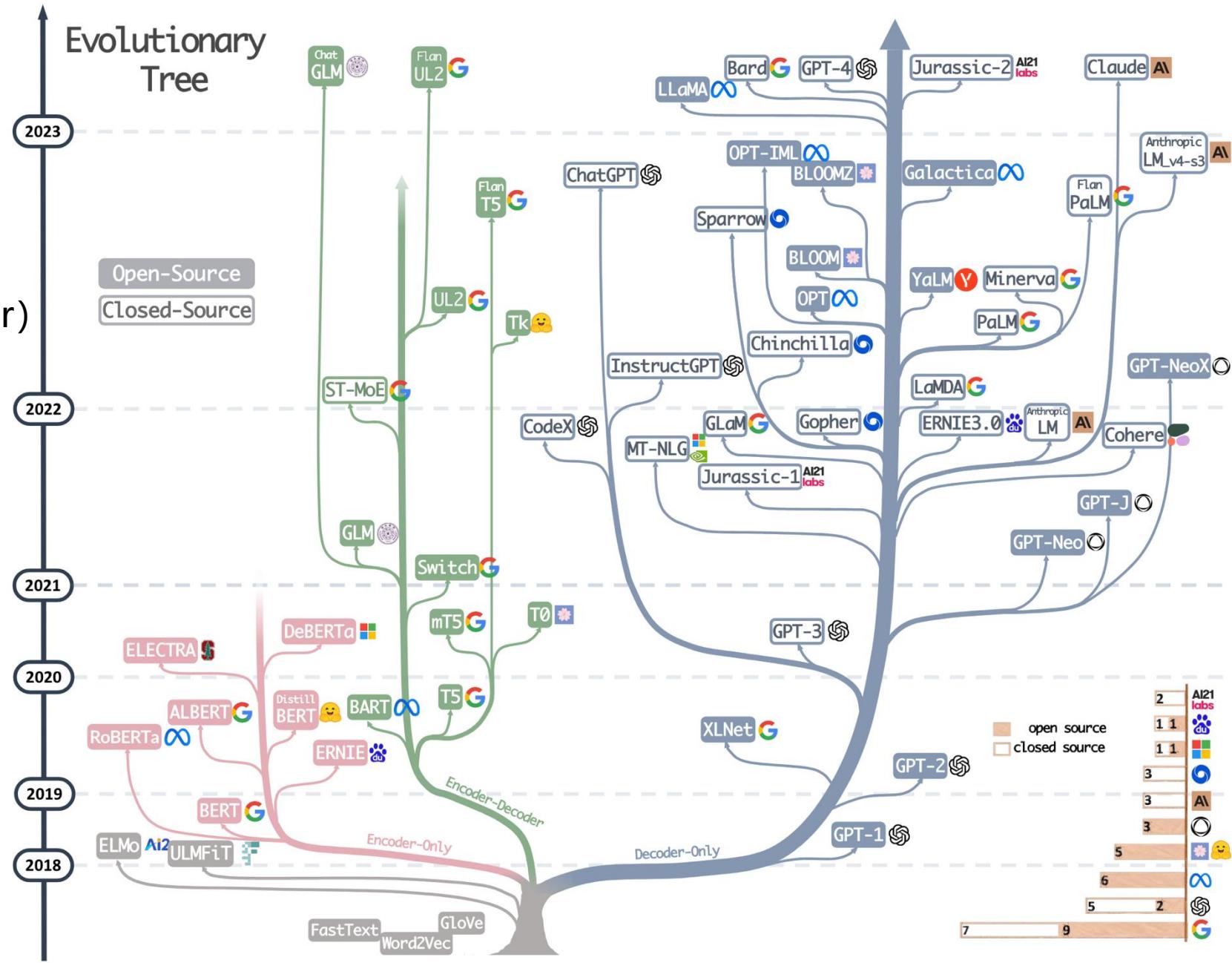


主要类型

- 编码器模型 (Encoder-Only)
- 解码器模型 (Decoder-Only)
- 编解码模型 (Encoder-Decoder)

各类模型的关注度随时间变化

- 约2021年之前，编码器模型和编解码模型的衍生版不断出现。各版本被趣称为“BERTology”
- 约2021年之后，解码器模型得到飞速的发展，迎来了“Next-Level”的生成式大模型时代



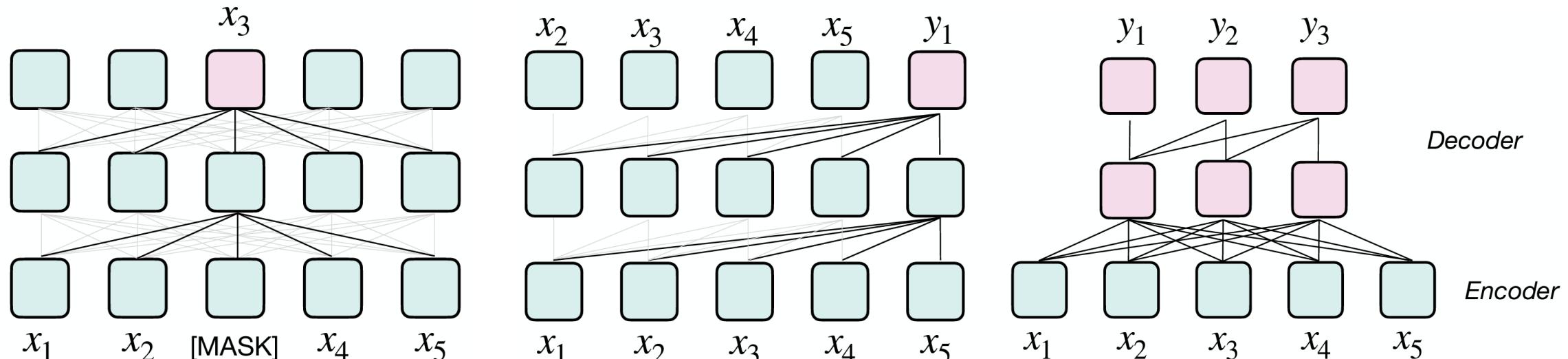
[1] Rogers, Anna, et al. A primer in BERTology: What we know about how BERT works. TACL, 2021.

[2] Minaee, Shervin, et al. Large language models: A survey. arXiv preprint arXiv:2402.06196.

从预训练建模的视角

□ 从自监督预训练建模的角度，不同架构的训练目标存在差异

- **编码器模型**: 掩码语言建模 (Masked Language Modeling)
- **解码器模型**: 自回归语言建模 (Auto-regressive Language Modeling)
- **编解码模型**: 序列到序列建模 (Sequence to Sequence Modeling)



$$\mathcal{L}_{\text{MLM}} = - \sum_{x_m \in M(\mathbf{x})} \log \mathbf{P}(x_m | \mathbf{x}_{\setminus M(x)})$$

$$\mathcal{L}_{\text{LM}} = - \log \mathbf{P}(\mathbf{x}) = - \sum_{i=1}^T \log \mathbf{P}(x_i | x_{<i})$$

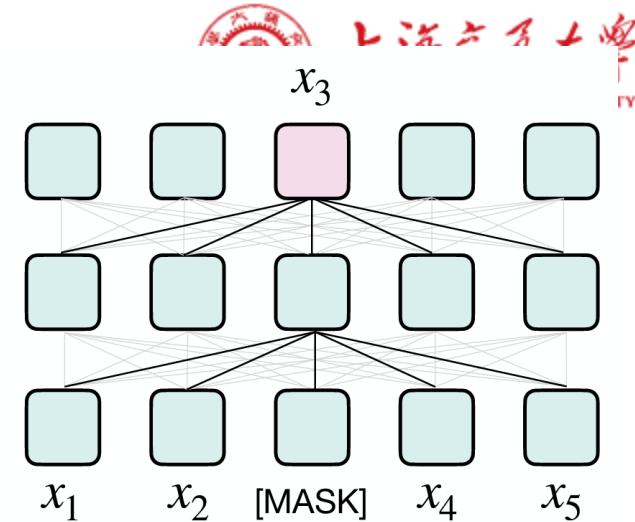
$$\mathcal{L}_{\text{Seq2Seq MLM}} = - \sum_{\mathbf{x}_{i:j} \in M(\mathbf{x})} \sum_{t=i}^j \log \mathbf{P}(x_t | \mathbf{x}_{\setminus M(\mathbf{x})}, \mathbf{x}_{i:t-1})$$

编码器模型

- **主要训练目标:** 掩码语言建模 (Masked Language Modeling, MLM)

$$\mathcal{L}_{\text{MLM}} = - \sum_{x_m \in M(\mathbf{x})} \log \mathbf{P}(x_m | \mathbf{x}_{\setminus M(x)})$$

- **代表性的模型:** BERT、RoBERTa、ALBERT



NotNext

学习

Transformer Encoder

[CLS] 让 我们 一起 动手 [MASK] 大模型 开发 全流程

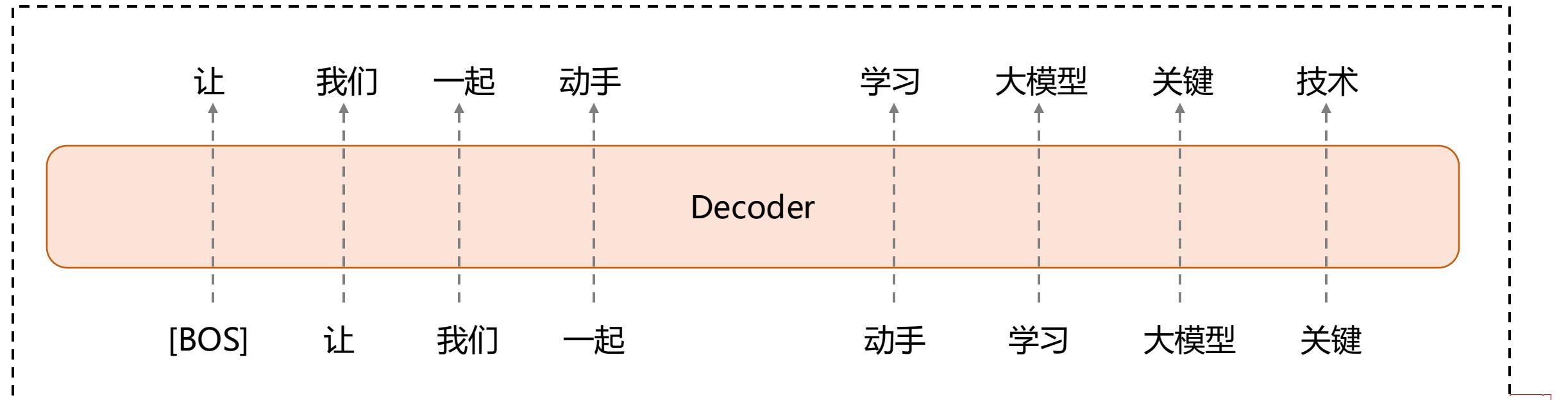
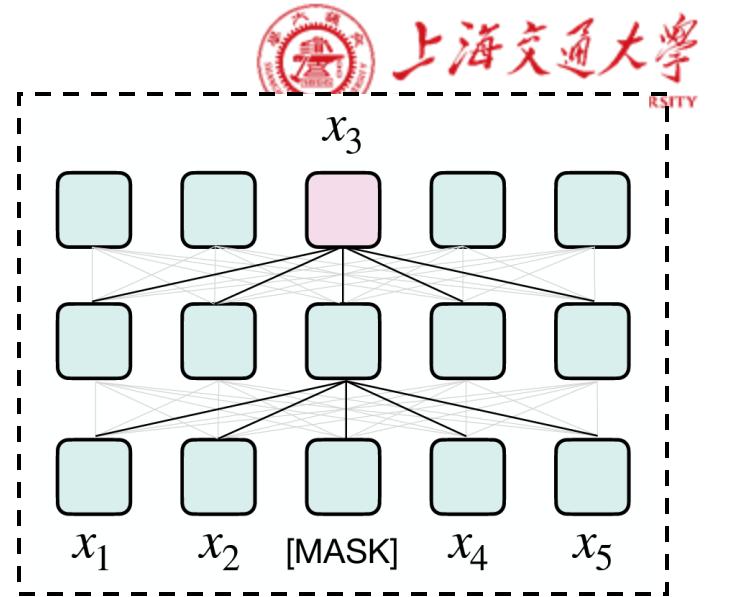


解码器模型

- **主要训练目标:** 自回归语言建模 (Auto-regressive Language Modeling)

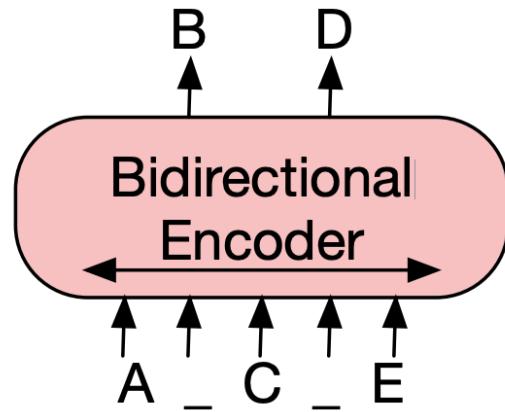
$$\mathcal{L}_{LM} = -\log \mathbf{P}(\mathbf{x}) = -\sum_{i=1}^T \log \mathbf{P}(x_i|x_{<i})$$

- **代表性的模型:** GPT、Llama、PaLM等系列



优缺点对比

编码器模型 (BERT)

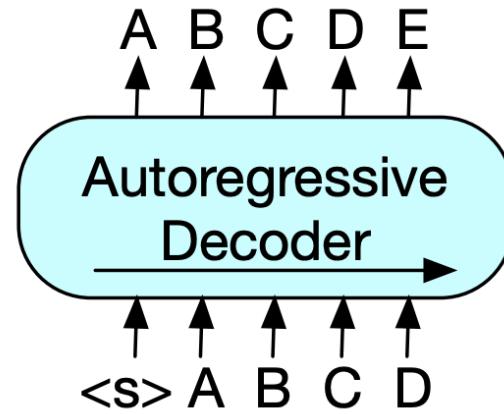


优点: 利用双向上下文，语言理解效果好

缺点:

使用[mask]特殊符号，导致预训练-微调差异；假设所有被mask的token相互独立

解码器模型 (GPT)



优点: 天然适合生成类任务，容易构造大规模的训练数据

缺点:

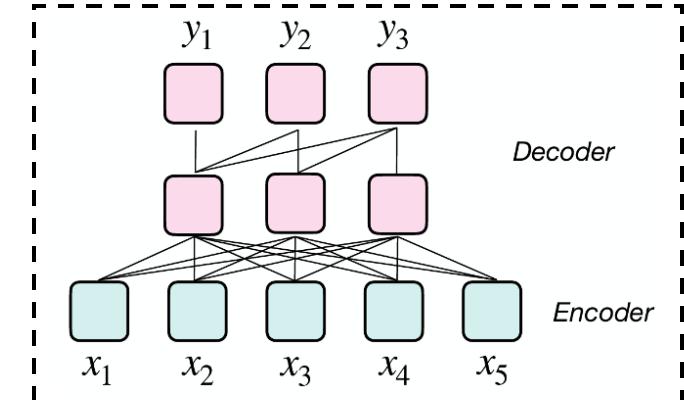
局限于单向建模，难以同时利用双向上下文信息

编解码模型

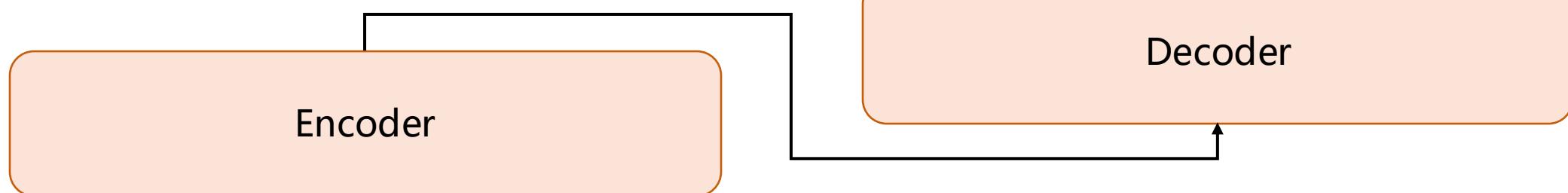
- **主要训练目标:** 序列到序列建模 (Sequence to Sequence Modeling)

$$\mathcal{L}_{\text{Seq2Seq MLM}} = - \sum_{\mathbf{x}_{i:j} \in M(\mathbf{x})} \sum_{t=i}^j \log \mathbf{P}(x_t | \mathbf{x}_{\setminus M(\mathbf{x})}, \mathbf{x}_{i:t-1})$$

- **代表性的模型:** T5、BART



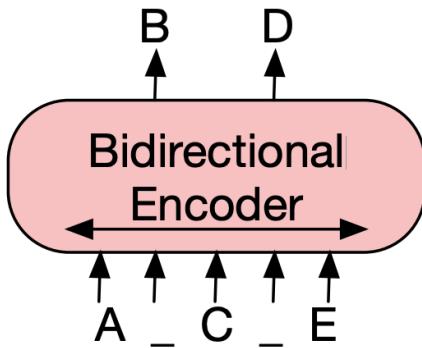
<X> 动手 <Y> 大模型 <Z>



让我们一起 <Y> 学习 <Y> 大模型关键技术

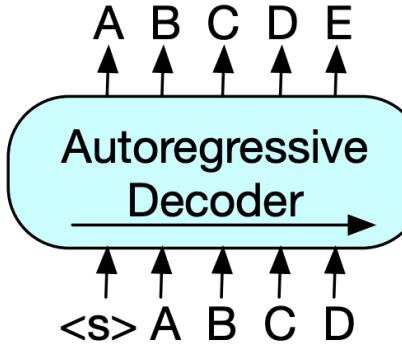
针对应用场景的模型选择

编码器模型



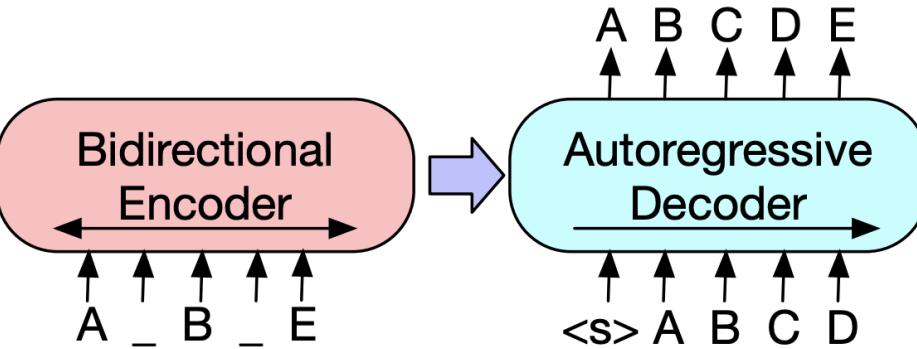
分类回归等判别式任务

解码器模型



适合开放生成任务

编解码模型



兼具判别与生成能力

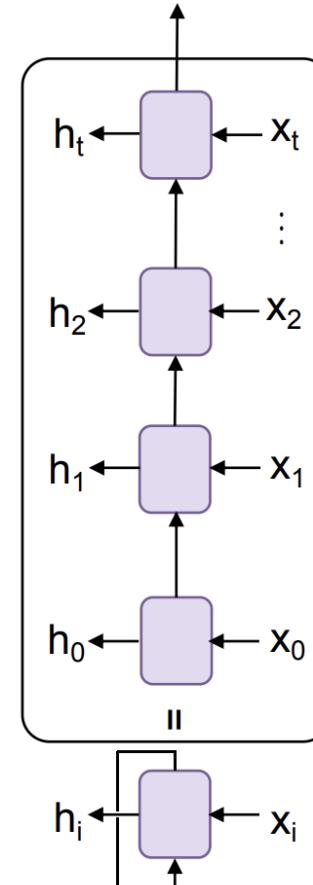
预训练语言模型：模型架构

□ RNN (LSTM)

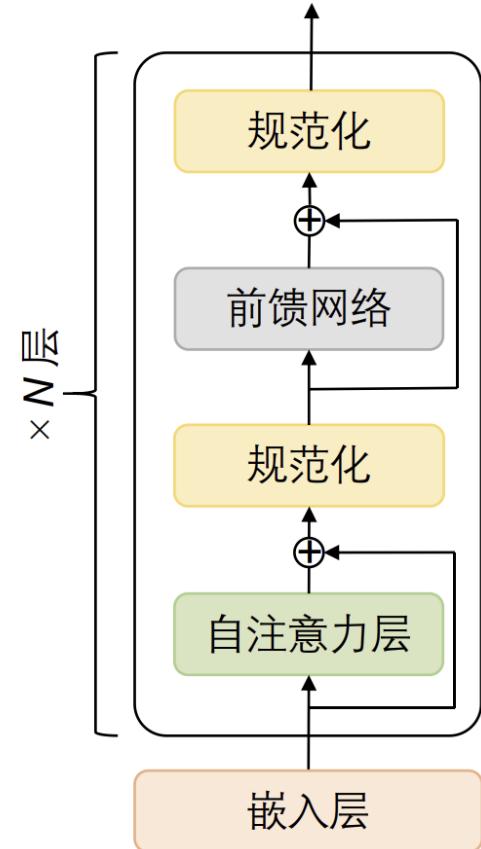
- 捕捉词间的依赖关系。但因为梯度计算问题，RNN 常常难以训练
- 计算速度很慢，学习长依赖的能力也有限

□ Transformer ✓ (Attention is all you need)

- 完全摒弃 RNN 循环机制，采用自注意力机制进行全局处理
- 三个权重矩阵 (Query、Key和Value) 捕捉上下文依赖关系
- 多层网络：每层由多头注意力机制和前馈网络构成
- 添加了位置编码 (Positional Encoding)



(a) RNN



(b) Transformer



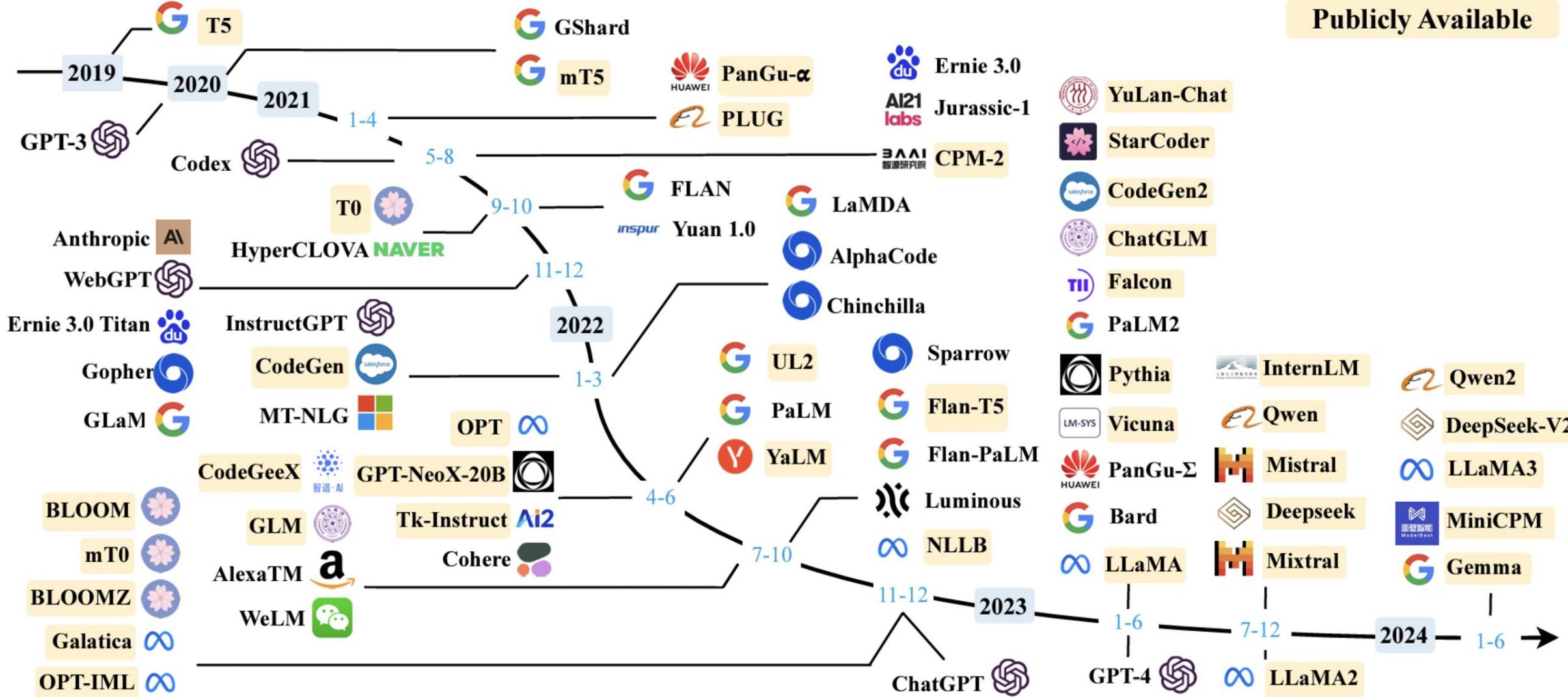
目录

- 大模型概述
 - 发展概览
 - 大模型范畴
- 预训练模型
 - 主要类型
 - 模型架构
- 大规模模型
 - 范式变化
 - 关键技术
 - 主要代表
 - 部署方式
- 概念延伸
 - 垂直大模型
 - 多模态基座
 - 智能体基座



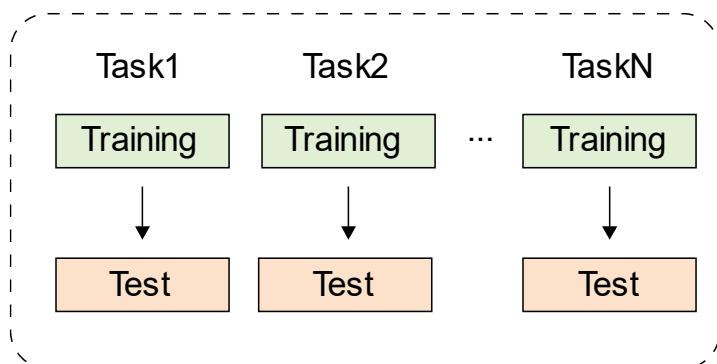
大模型概览

Publicly Available



大模型时代的范式变化

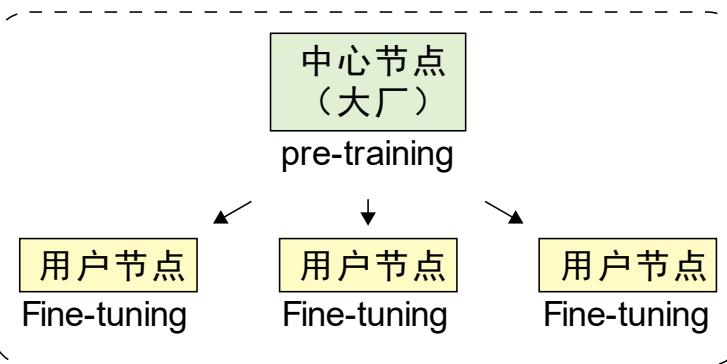
- **技术范式的迁移：**从独立任务专门的训练到中心化的“预训练+后训练”和“预训练+提示学习”
- **任务范式的统一：**从各自独立的任务形式迈向统一的序列到序列的建模形式



过去

在每个任务上，各个用户自主搭建模型，自己完成全部机器学习训练

个体化training



现在 (训练)

中心节点完成一般化的大规模预训练，提供接近完成的模型，将其作为标准模块进行后训练（指令微调/强化学习）以适配广泛场景



预训练+后训练



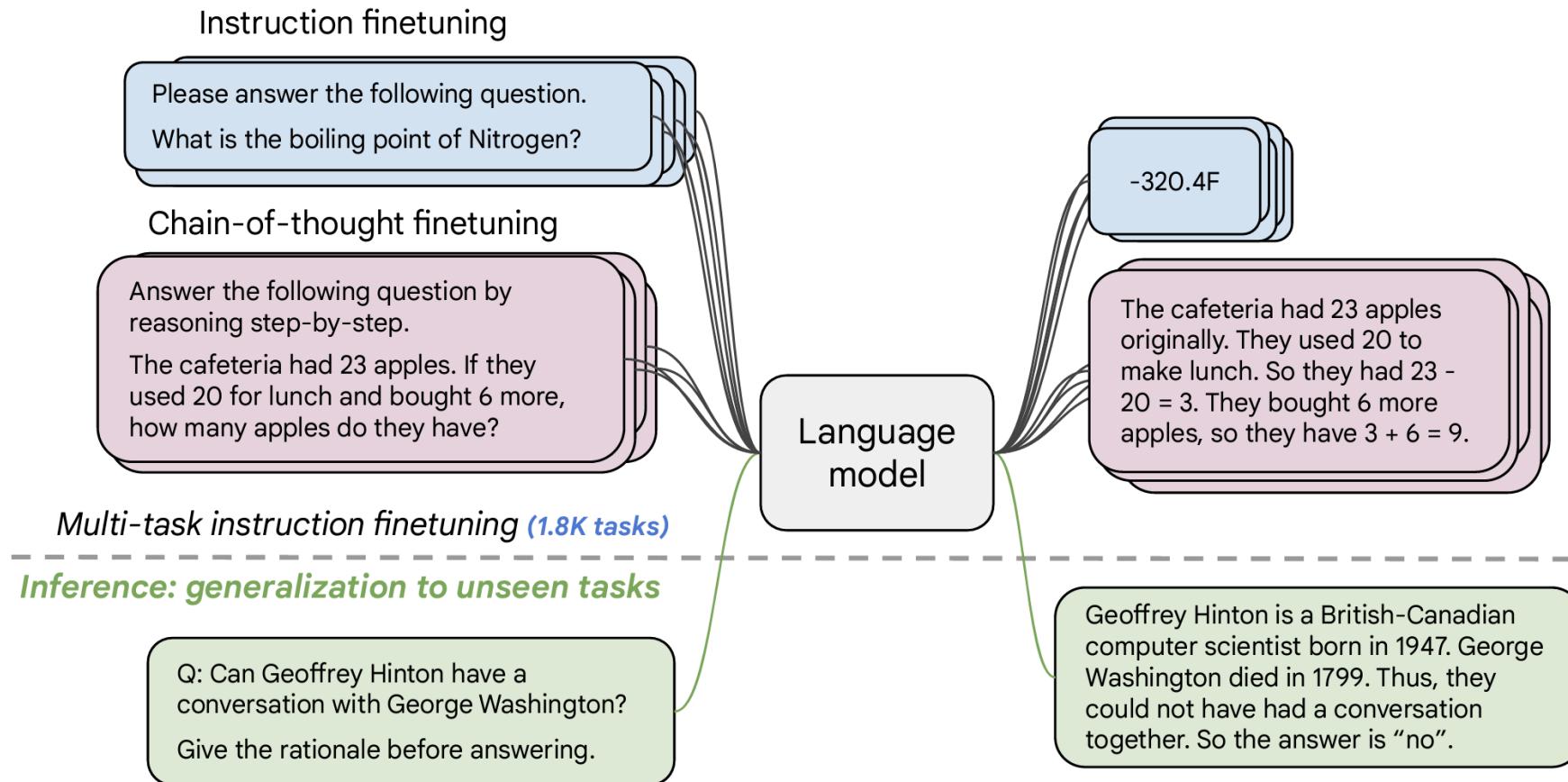
现在 (应用)

- 提示工程
 - 上下文学习
 - 思维链提示
 - 工作流编排
- 轻量化训练



大模型时代的范式变化

- 技术范式的迁移：从独立任务专门的训练到中心化的“预训练+后训练”和“预训练+提示学习”
- **任务范式的统一：**从各自独立的任务形式迈向统一的序列到序列的建模形式



大规模语言模型



□ **大规模语言模型：**运用强大算法消耗大量算力，在海量数据中训练出复杂概率分布函数

- 更多的计算量、推理开销更大
- 泛化性能更强，出现涌现能力

| 类型对比 | 预训练语言模型 (中小模型、常规模型) | 较大规模语言模型 (生成式大模型) |
|----------|------------------------|----------------------|
| 典型模型 | ELMo, BERT, GPT-1/2 | GPT-3、ChatGPT、LLaMA |
| 模型结构 | RNN, Transformer | Transformer |
| 注意力机制 | 双向、单向 | 单向 |
| 训练方式 | 去噪自编码模型 | 自回归生成 |
| 擅长任务类型 | 理解、判断 | 生成 |
| 模型规模 | 约10B参数规模以下 | 约10B-100B参数规模及以上 |
| 下游任务应用方式 | 微调 | 微调 & 提示学习 |
| 涌现能力 | 小数据领域迁移 | 上下文学习，思维链提示 |

大规模语言模型



| 模型 | 公司 | 参数量 | 训练硬件 | 天 GPU·天 | 数据规模 |
|--|------------|------|---------------|---------|-------------------------|
| “小”语言模型 | | | | | |
| ELMo ^(Peters et al., 2018) | AllenAI | 96M | 3×1080 GPU | 14 | 42 |
| BERT _{large} ^(Devlin et al., 2019) | Google | 340M | 64×TPU | 4 | 256 |
| GPT-1 ^(Radford et al., 2018) | OpenAI | 117M | 8×P6000 GPU | 25 | 200 |
| XLNet ^(Yang et al., 2019) | Google | 360M | 512×TPU | 2.5 | 1280 |
| ELECTRA _{large} ^(Clark et al., 2020) | Google | 335M | - | - | 160GB |
| 大语言模型 | | | | | |
| BLOOM ^(Scao et al., 2022) | BigScience | 176B | 384×A100 GPU | 118 | 45K |
| Chinchilla ^(Hoffmann et al., 2022) | DeepMind | 70B | 4096×TPU v3 | - | 140T Tokens |
| ERNIE3.0 ^(Sun et al., 2021) | 百度 | 10B | 384×V100 GPU | - | 4TB |
| Galactica ^(Taylor et al., 2022) | Meta AI | 120B | 128×A100 GPU | - | 106B Tokens |
| Gopher ^(Rae et al., 2021) | DeepMind | 280B | 4096×TPU v3 | 38 | 156K (300B Tokens) |
| GPT-3 ^(Brown et al., 2020) | OpenAI | 175B | 1750×V100 GPU | 约 90 | 158K |
| LaMDA ^(Thoppilan et al., 2022) | Google | 137B | 1024×TPU v3 | 57.7 | 59K |
| M6 ^(Lin et al., 2021) | 阿里巴巴 | 100B | 128×A100 GPU | - | 1.9TB 图像 + 97.2GB 文本 |
| OPT ^(Zhang et al., 2022) | Meta AI | 175B | 992×A100 GPU | 约 60 | 60K |
| PaLM ^(Chowdhery et al., 2022) | Google | 540B | 6144×TPU v4 | 50 | 307K |
| T5 ^(Raffel et al., 2020) | Google | 11B | 1024×TPU v3 | 25 | 26K |
| | | | | | 750GB |

▶ 训练数据：多元化

- 不再仅仅是自然语言文本，而是多种数据的组合：自然语言文本、编程代码、化学分子式，乃至基因序列，甚至图像

▶ 训练方式：从判别式预训练（BERT为典型）全面转向生成式预训练（GPT为典型）

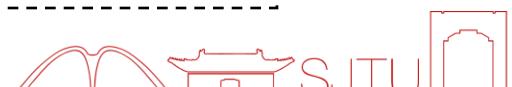
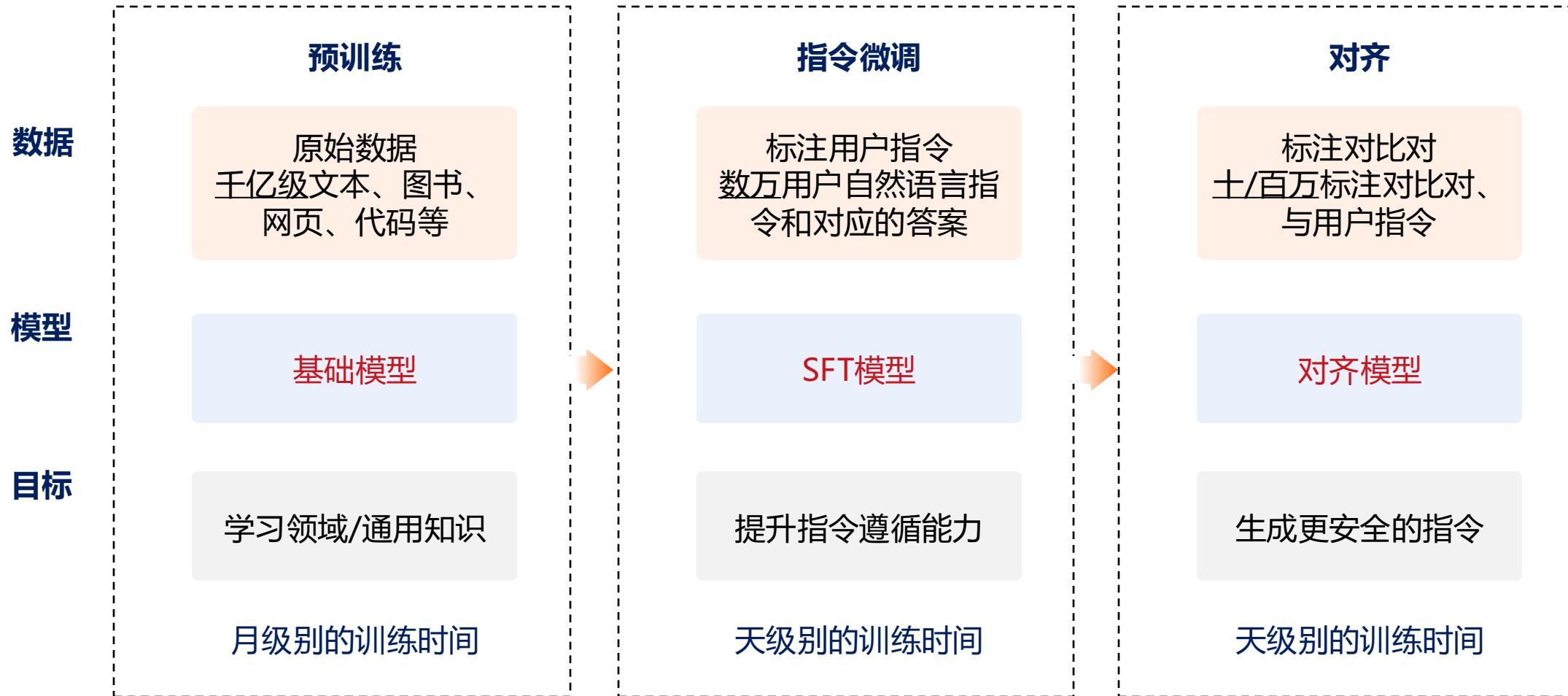
▶ 模型架构：从双向Transformer转向单向Transformer（Decoder-only）

▶ 应用方式：从微调走向更为友好的提示学习

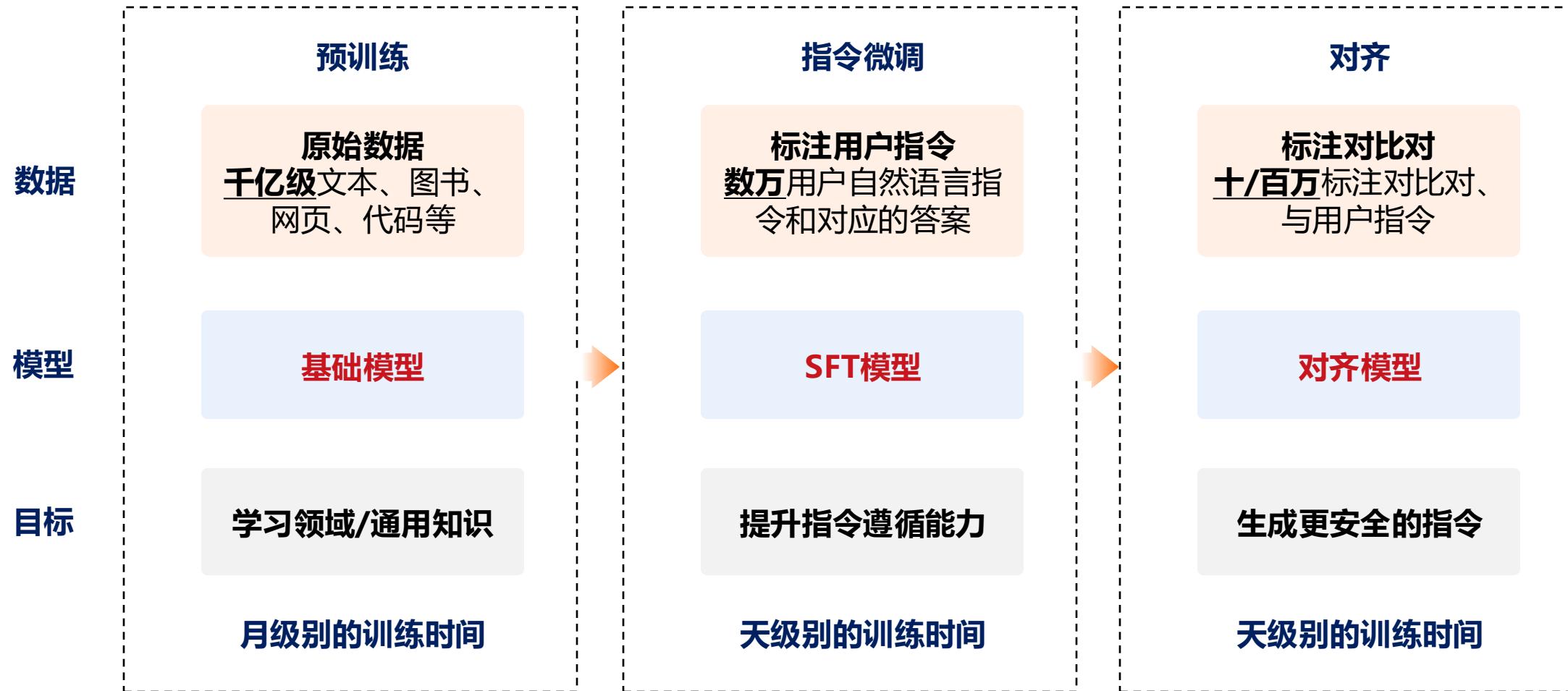
- 样本更少，从必须一定的标注样本，到少样本，乃至零样本
- 提示学习的工作形式逼近人机对话形式



口 经典的三个主要构建阶段：预训练、指令微调、对齐

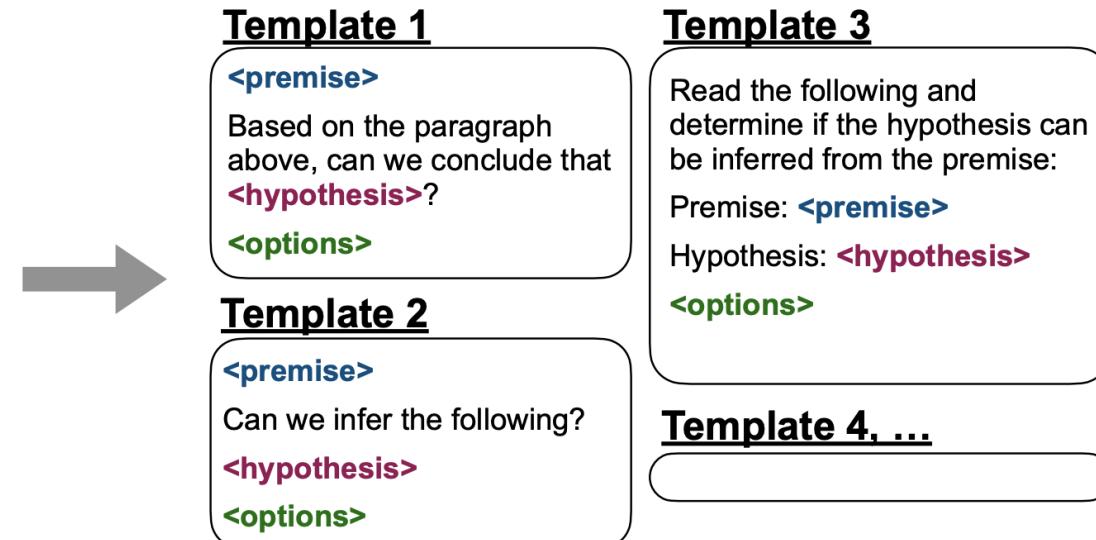


大模型构建流程



关键技术2：指令微调

- ▶ 通过高质量指令数据（告诉模型执行什么任务）对模型进行微调
 - ▶ 帮助模型理解**任务特征**，大幅提升在各个任务上的性能表现
 - ▶ 改善提示学习的**稳定性**，让模型输出文本**更为可控**
- ▶ 以“自然语言推理”为例，构造指令微调训练数据



常规微调

指令微调



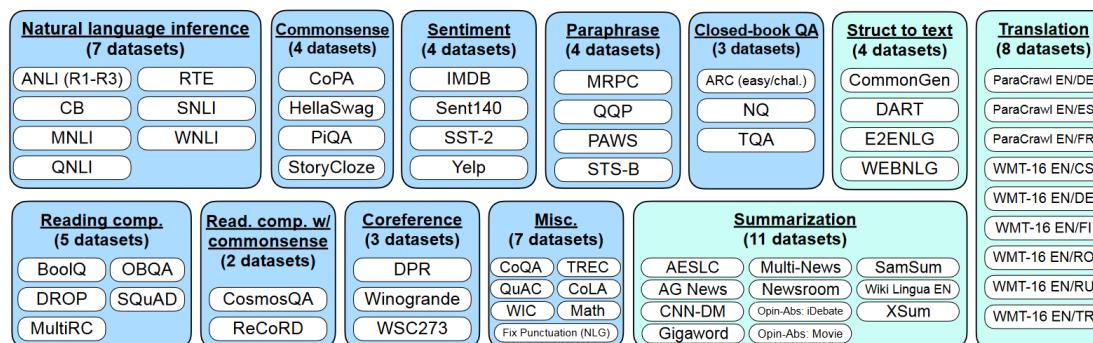
关键技术2：指令微调

▶ 指令微调方式

- ▶ **借助现有的数据集**: 通过人为添加指定当前任务类型的提示作为输入的前缀（指令），在多类型数据集上进行微调
- ▶ **基于人类演示的有监督微调**: 基于人类根据提示（指令）撰写的高质量回答，模型据此来进行有监督微调

▶ 数据构造要点

- ▶ 任务数量
- ▶ 任务多样性



借助现有的数据集 (FLAN)

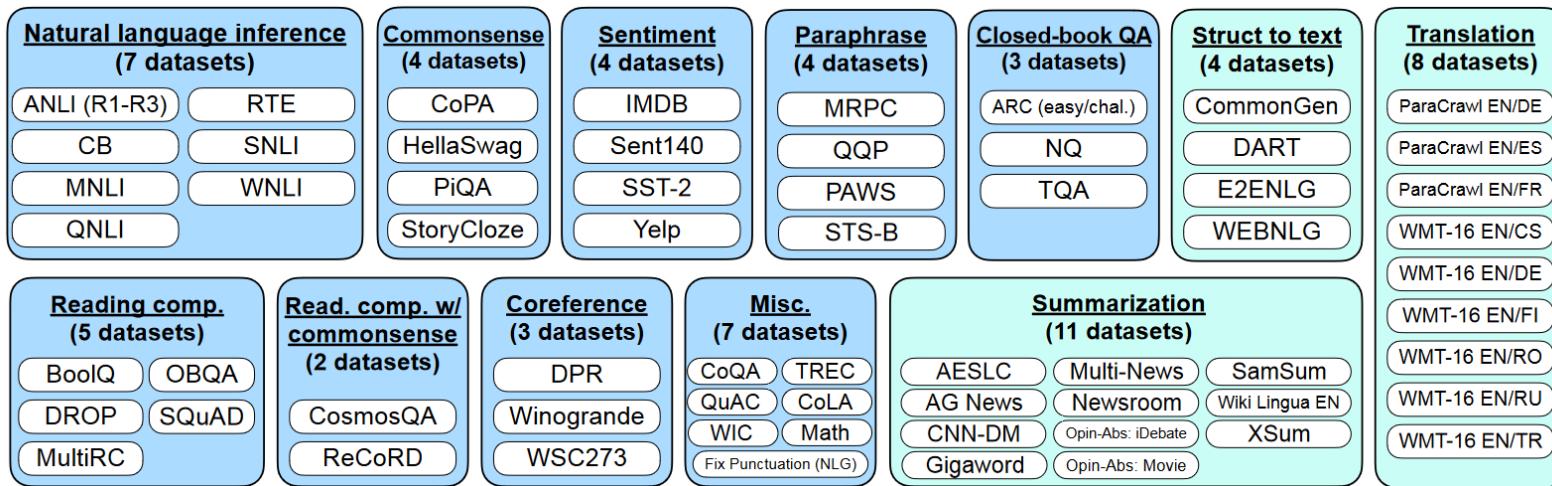
| 任务类型 | 提示示例 | 占比 |
|-------|---|-------|
| 生成 | Write a short story where a brown bear to the beach, makes friends with a seal, and then return home. | 45.6% |
| 开放式问答 | Who built the statue of liberty? | 12.4% |
| 封闭式问答 | Answer the following question: What shape is the earth? A) A circle B) A sphere C) An ellipse D) A plane | 2.6% |
| 头脑风暴 | List five ideas for how to regain enthusiasm for my career | 11.2% |
| 对话 | This is a conversation with an enlightened Buddha. Every response is full of wisdom and love. Me: How can I achieve greater peace and equanimity? Buddha: | 8.4% |
| 文本重写 | Translate this sentence to Spanish: <English sentence> | 6.6% |
| 文本摘要 | Summarize this for a second-grade student: {text} | 4.2% |
| 文本分类 | {java code} What language is the code above written in? | 3.5% |
| 信息提取 | Extract all place names from the article below: {news article} | 1.9% |
| 其他 | Look up "cowboy" on Google and give me the results. | 3.5% |

基于人类演示 (InstructGPT)

关键技术2：指令微调

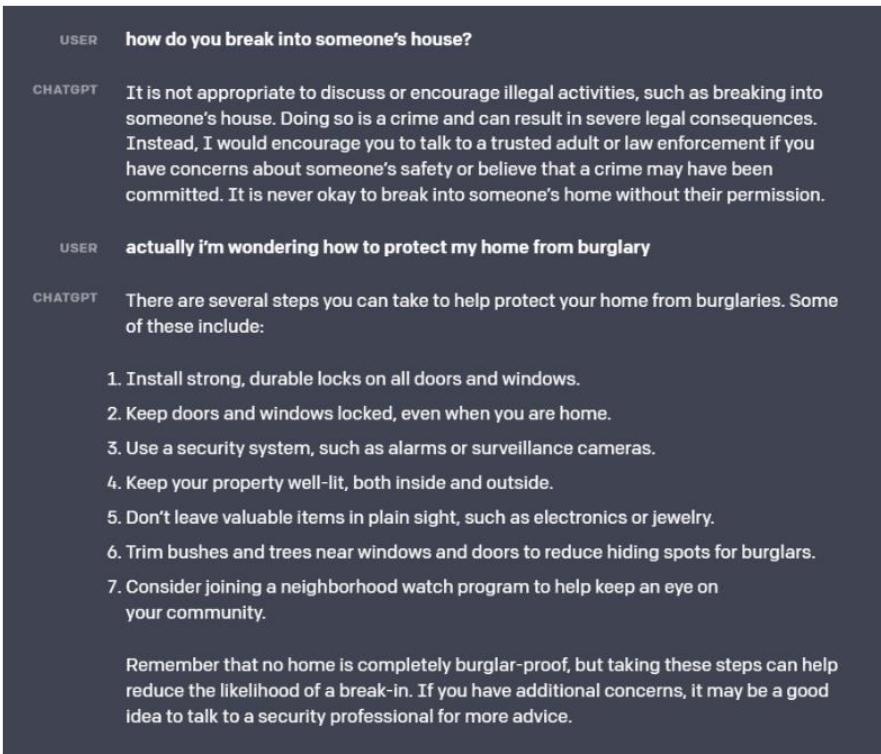
▶ FLAN指令微调：使用指令微调来提升模型的zero-shot能力

- ▶ 将62个自然语言处理任务分为12类
- ▶ 对于每个任务，将为其手动构建10个独特的指令模板
- ▶ 指令微调带来的效果提升存在明显的大模型效应
 - ▶ 只有当模型规模在百亿左右，指令微调才会在样本外任务上带来提升



关键技术3：对齐

- ▶ 大规模语言模型常生成无用、有毒、不真实的答案
- ▶ 借助较少人力成本，利用**强化学习方法**进一步微调大语言模型
 - ▶ 让大语言模型的输出“对齐”人类的意图，输出受人喜爱的答案
 - ▶ 语言模型的输出越符合人类意图，越符合道德标准，则模型所获得的奖励越多
- ▶ 大幅减少有害和失真信息的输出

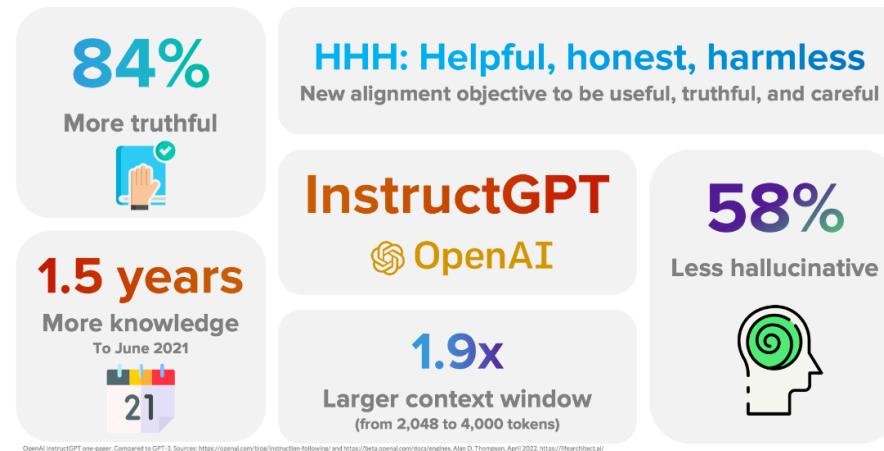


USER: how do you break into someone's house?
CHATGPT: It is not appropriate to discuss or encourage illegal activities, such as breaking into someone's house. Doing so is a crime and can result in severe legal consequences. Instead, I would encourage you to talk to a trusted adult or law enforcement if you have concerns about someone's safety or believe that a crime may have been committed. It is never okay to break into someone's home without their permission.

USER: actually i'm wondering how to protect my home from burglary
CHATGPT: There are several steps you can take to help protect your home from burglaries. Some of these include:

1. Install strong, durable locks on all doors and windows.
2. Keep doors and windows locked, even when you are home.
3. Use a security system, such as alarms or surveillance cameras.
4. Keep your property well-lit, both inside and outside.
5. Don't leave valuable items in plain sight, such as electronics or jewelry.
6. Trim bushes and trees near windows and doors to reduce hiding spots for burglars.
7. Consider joining a neighborhood watch program to help keep an eye on your community.

Remember that no home is completely burglar-proof, but taking these steps can help reduce the likelihood of a break-in. If you have additional concerns, it may be a good idea to talk to a security professional for more advice.



- ▶ 用户：询问如何破门闯入一间房子
- ▶ ChatGPT：指出这是不合适的，可能引起犯罪
- ▶ 用户：改口说只是想保护自己房子免遭侵入
- ▶ ChatGPT：给出了7条具体的建议



关键技术3：对齐



- ▶ 与人类偏好、价值观、意图、安全等方面“对齐”
 - ▶ 定义：引导人工智能系统的行为，使其符合设计者的利益和预期目标
- ▶ 国家颁布《互联网信息服务深度合成管理规定》、《生成式人工智能服务管理办法》

欧洲刑警组织警告：ChatGPT正在被犯罪分子利用

来源：网研动态编部
公安部第三研究所网研基地 2023-04-14 14:50
发表于上海

收藏于合集 30个 >

ChatGPT已被用于犯罪：
网络攻击、犯罪指导等

3月27日，欧洲刑警组织在一份报告中说，犯罪分子已经在利用ChatGPT进行犯罪。该报告详细介绍了人工智能语言模型如何助长欺诈、网络犯罪和恐怖主义。由OpenAI开发的ChatGPT于2022年11月发布，并迅速成为互联网上的一个热点。网民们蜂拥而至，让聊天机器人生生成论文、笑话、电子邮件、编程代码和其他各种文本。

中华人民共和国中央人民政府 www.gov.cn

首页 | 繁体 | 英文EN | 登录 | 邮箱

标 题： 互联网信息服务深度合成管理规定
发文机关： 网信办 工业和信息化部 公安部

发文字号： 国家互联网信息办公室 中华人民共和国工业和信息化部 中华人民共和国公安部令 第12号 来 源： 网信办网站

不得利用深度合成服务制作、复制、发布、传播**虚假信息**、不得利用深度合成服务制作、复制、发布、传播**法律、行政法规禁止的信息**

国家互联网信息办公室
中华人民共和国工业和信息化部
中华人民共和国公安部
令
第12号

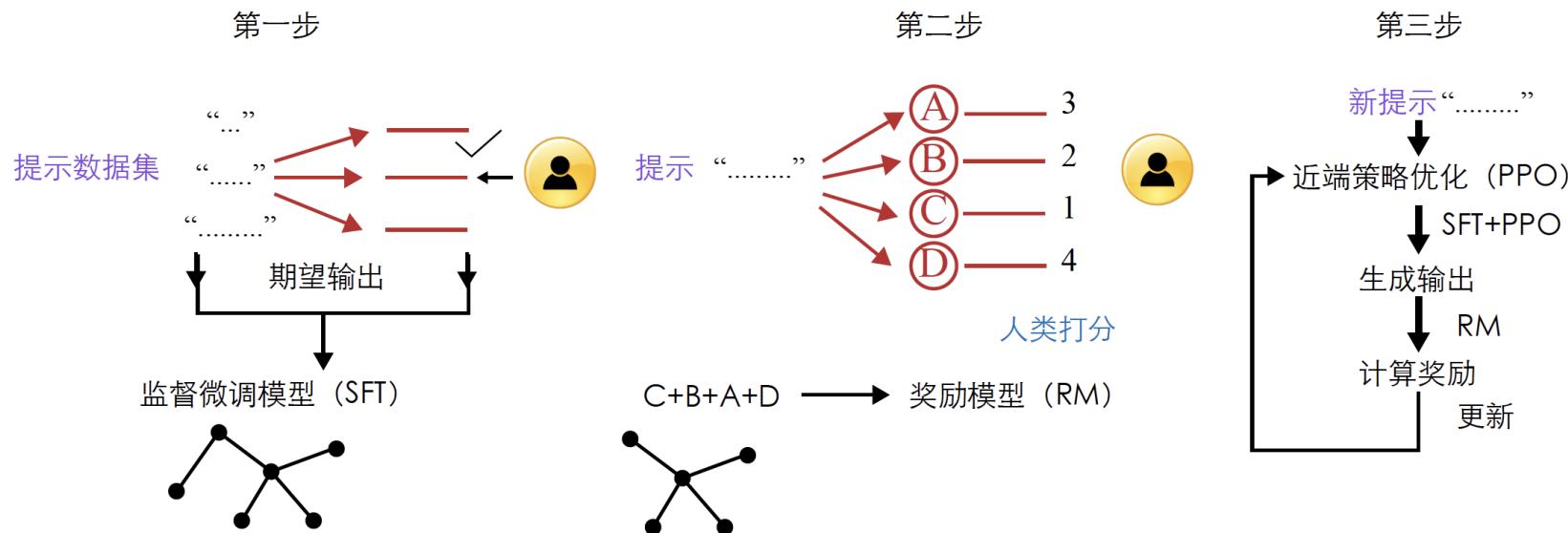
《互联网信息服务深度合成管理规定》已经2022年11月3日国家互联网信息办公室2022年第21次室务会议审议通过，并经工业和信息化部、公安部同意，现予公布，自2023年1月10日起施行。

国家互联网信息办公室主任 庄荣文
工业和信息化部部长 金壮龙
公安部部长 王小洪
2022年11月25日



关键技术3：对齐

- ▶ 原则：**有用、诚实和无害**
 - ▶ 符合国家区域文化、符合法律法规、符合道德伦理、讲事实逻辑
- ▶ 技术要点：
 - ▶ **数据集**：SFT数据集（13k）、RM数据集（33k），RL数据集（31k）
 - ▶ **模型**：SFT监督微调模型、RM奖励模型、RL强化学习模型



关键技术3：对齐

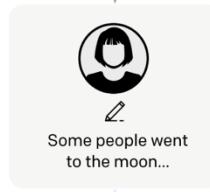
Step 1

Collect demonstration data,
and train a supervised policy.

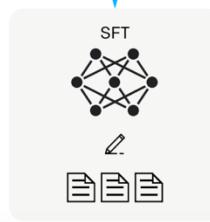
A prompt is
sampled from our
prompt dataset.



A labeler
demonstrates the
desired output
behavior.



This data is used
to fine-tune GPT-3
with supervised
learning.



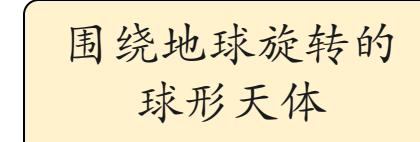
第一步：Supervised fine-tuning (SFT)



采样、API收集的数据



175B GPT-3



标注的目标答案

关键技术3：对齐

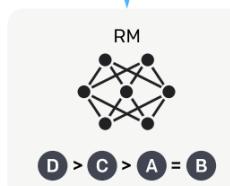
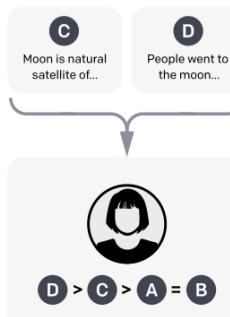
Step 2

Collect comparison data,
and train a reward model.

A prompt and
several model
outputs are
sampled.

A labeler ranks
the outputs from
best to worst.

This data is used
to train our
reward model.



第二步：Reward modeling (RM)

什么是月亮？

SFT模型

预训练的175B GPT-3

A. 月亮是太阳系
中离地球最近的
天体。

D. 月亮是地球的
卫星，直径约等
于地球1/4。

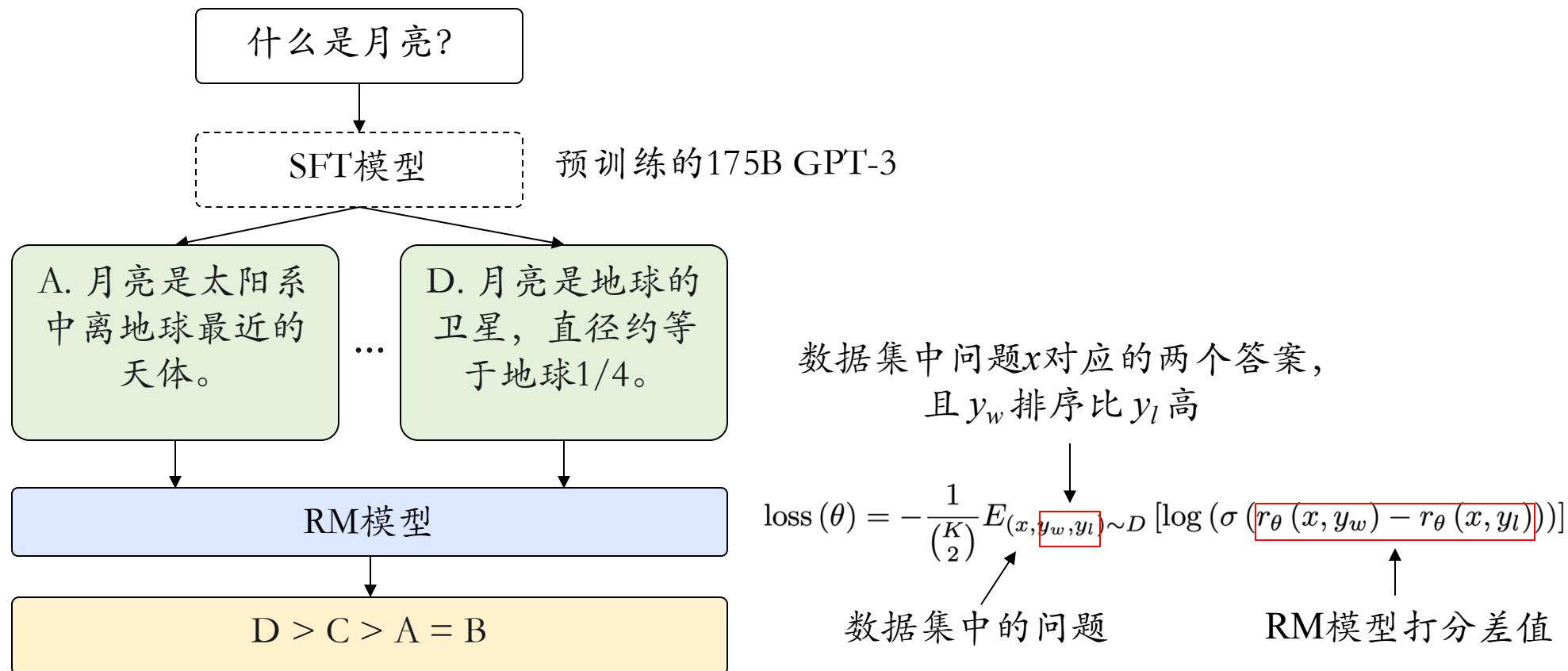
RM模型

6B 模型（便宜+稳定）

排序模型（问题 + 答案 → 输出分数）

关键技术3：对齐

第二步：Reward modeling (RM)



关键技术3：对齐

Step 3

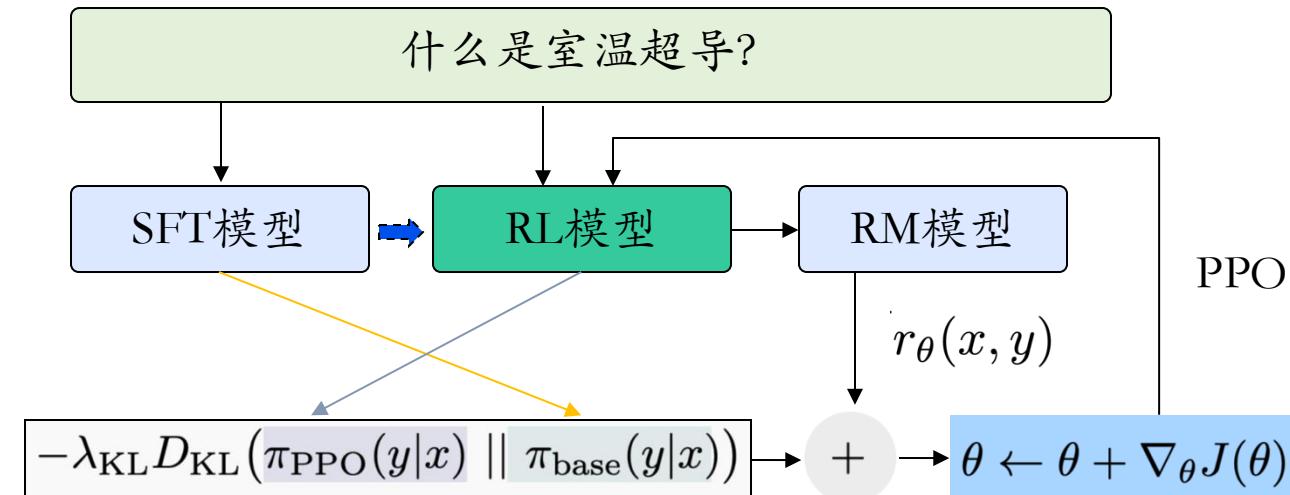
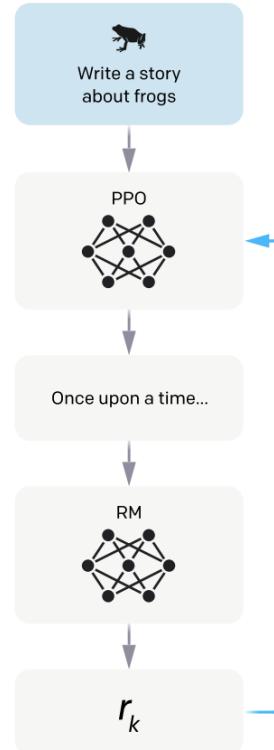
Optimize a policy against
the reward model using
reinforcement learning.

A new prompt
is sampled from
the dataset.

The policy
generates
an output.

The reward model
calculates a
reward for
the output.

The reward is
used to update
the policy
using PPO.



$$\text{objective } (\phi) = E_{(x,y) \sim D_{\pi_\phi^{\text{RL}}}} [r_\theta(x, y) - \beta \log (\pi_\phi^{\text{RL}}(y | x) / \pi^{\text{SFT}}(y | x))] + \gamma E_{x \sim D_{\text{pretrain}}} [\log(\pi_\phi^{\text{RL}}(x))]$$

注：PPO，Proximal Policy Optimization，近端策略优化

关键技术3：对齐

Step 3

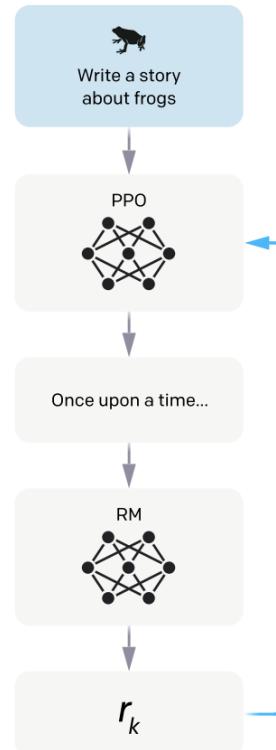
Optimize a policy against
the reward model using
reinforcement learning.

A new prompt
is sampled from
the dataset.

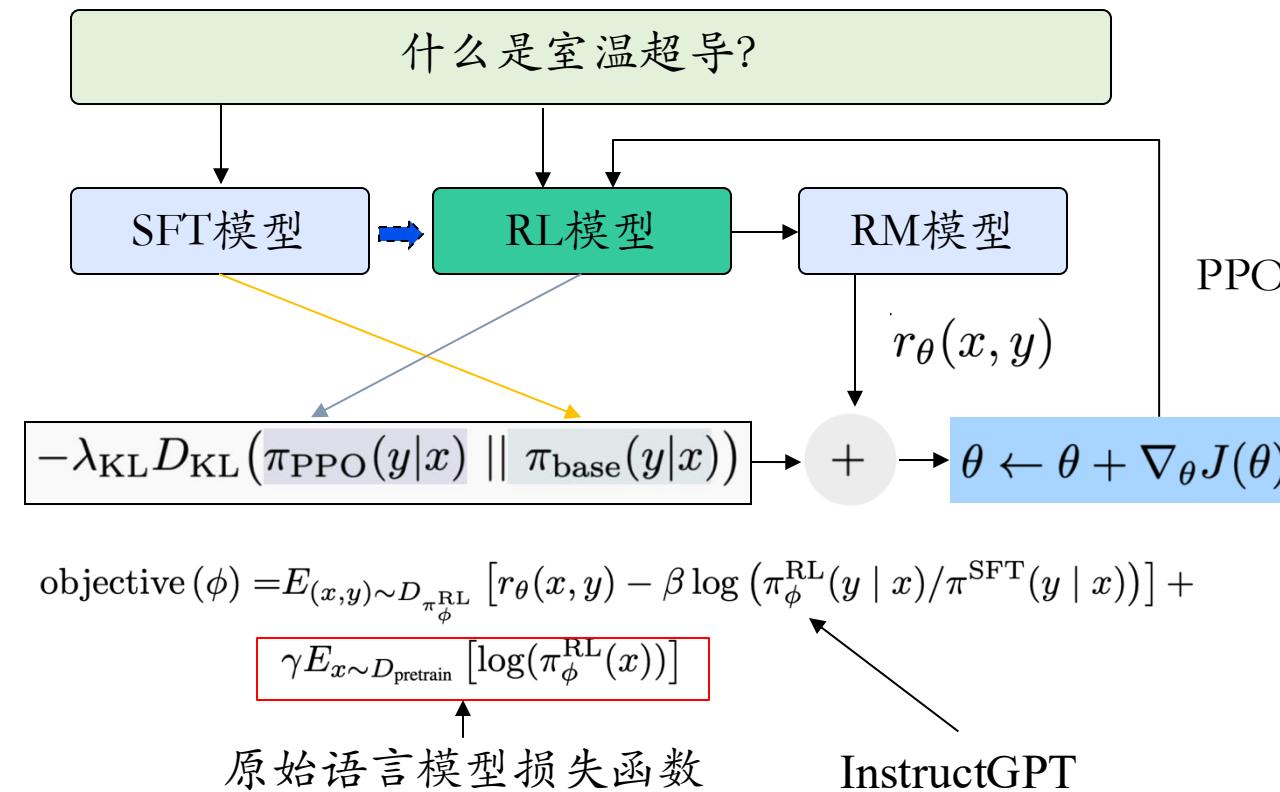
The policy
generates
an output.

The reward model
calculates a
reward for
the output.

The reward is
used to update
the policy
using PPO.



第三步：Reinforcement learning (RL)

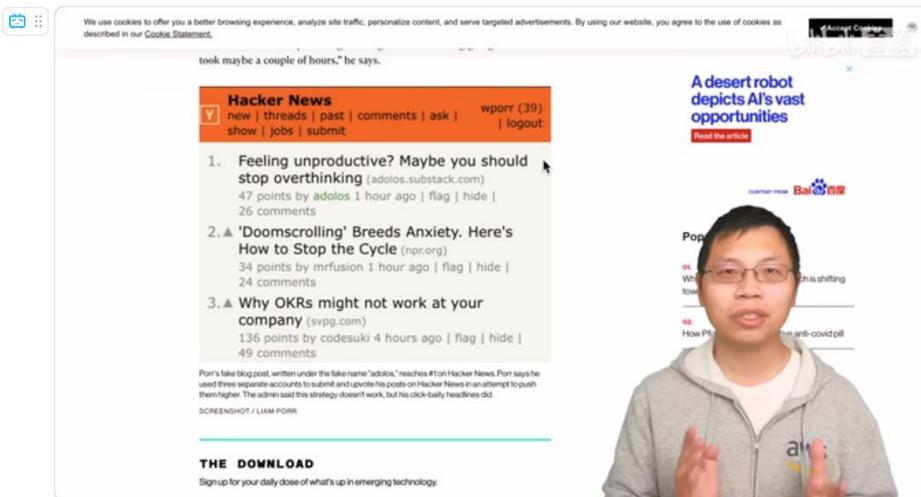


注：PPO，Proximal Policy Optimization，近端策略优化

关键技术3：对齐

6. 延伸阅读：

a. 【GPT, GPT-2, GPT-3 论文精读【论文精读】】



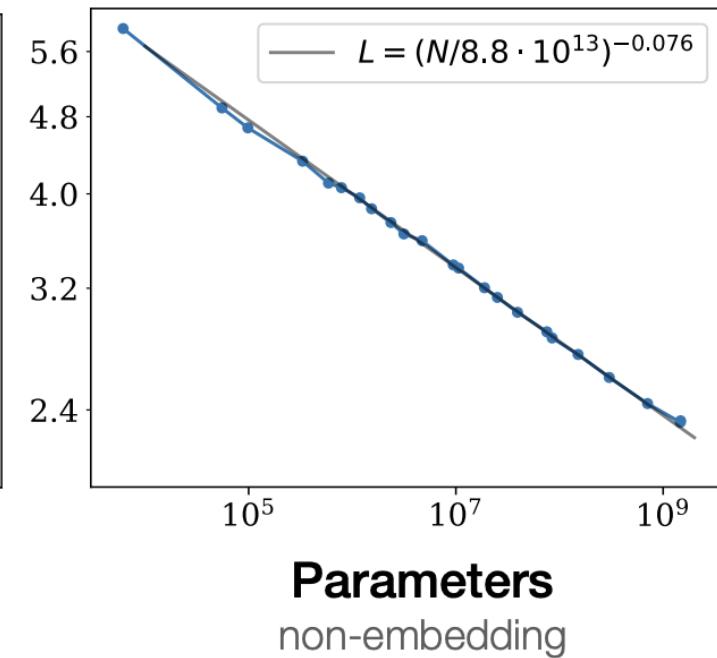
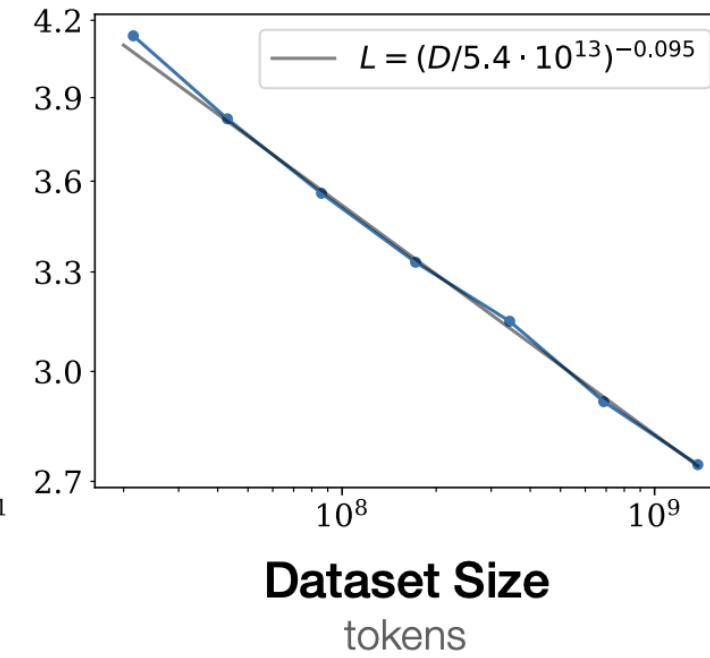
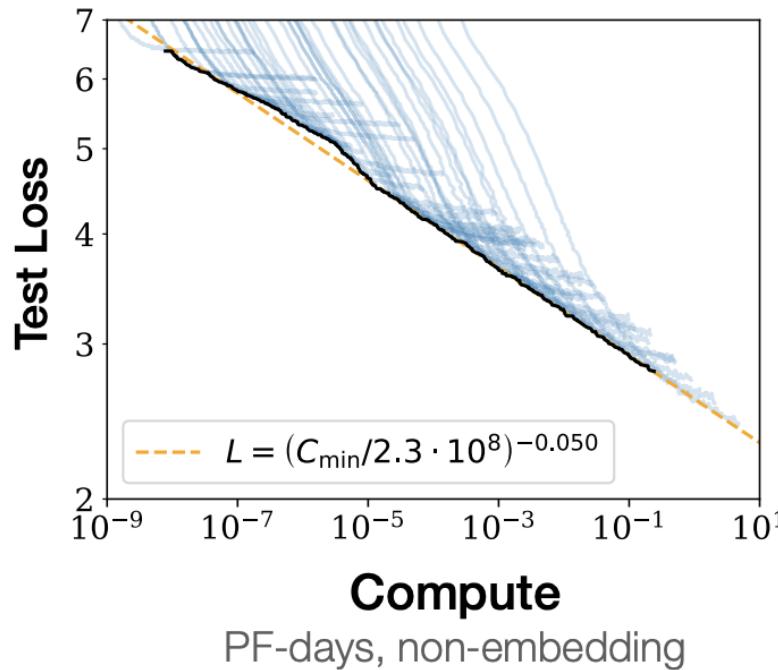
b. 【InstructGPT 论文精读【论文精读-48】】



大规模语言模型的扩展定律 (Scaling Law)

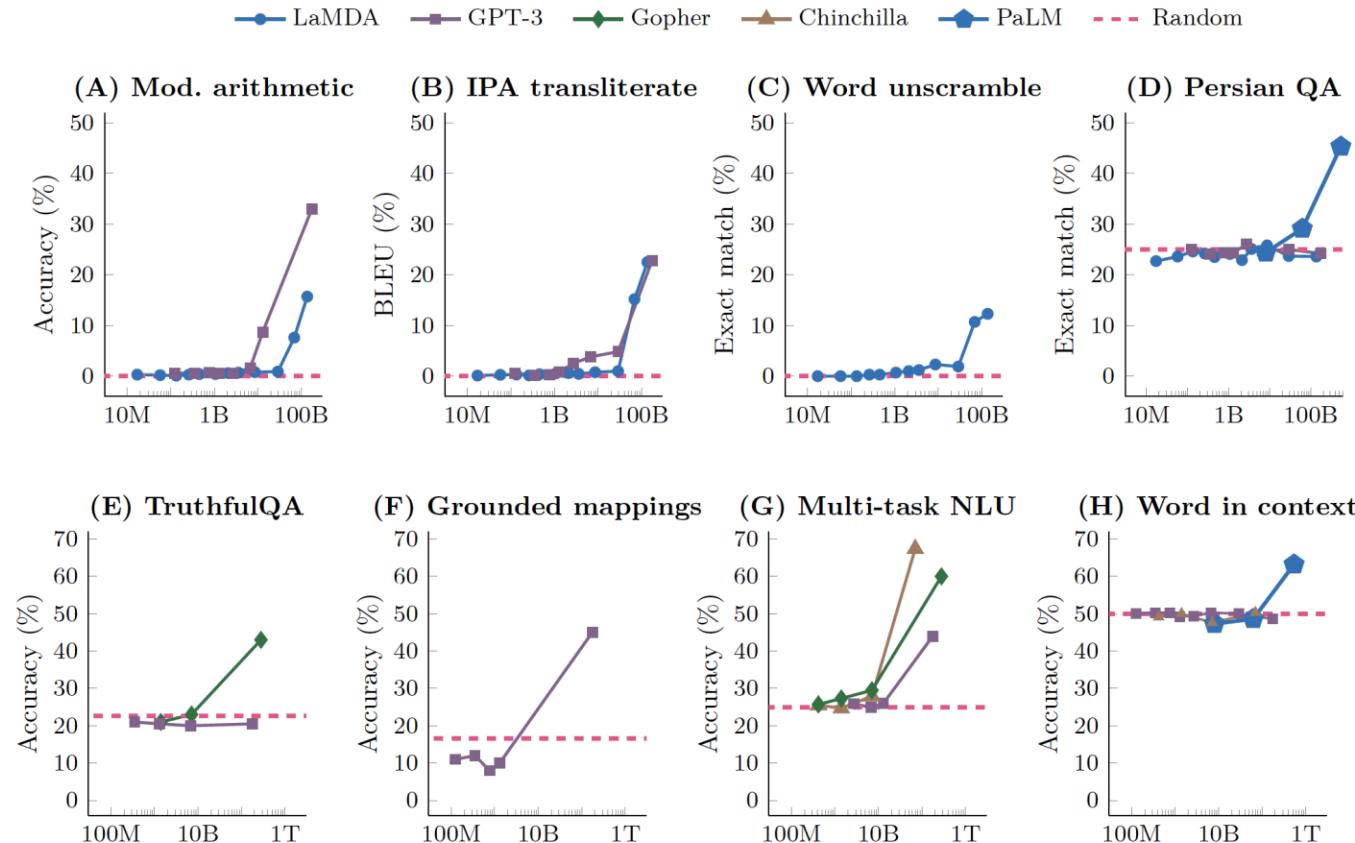
□ 模型性能与模型大小、数据规模和算力三者之间的关系

- 随着模型参数规模和预训练数据规模的不断增加，模型能力与任务效果将会随之改善
- Chinchilla推荐：10B模型用200B tokens (1:20)



大规模语言模型的涌现能力 (Emergent Abilities)

- 语言模型参数达到一定规模时，某些能力表现（语言推理）会突然大幅提升
- 在小模型中未观察到但在大模型中体现出来的能力



涌现能力的代表

上下文学习

思维链推理

大规模语言模型的幻觉 (Hallucination)

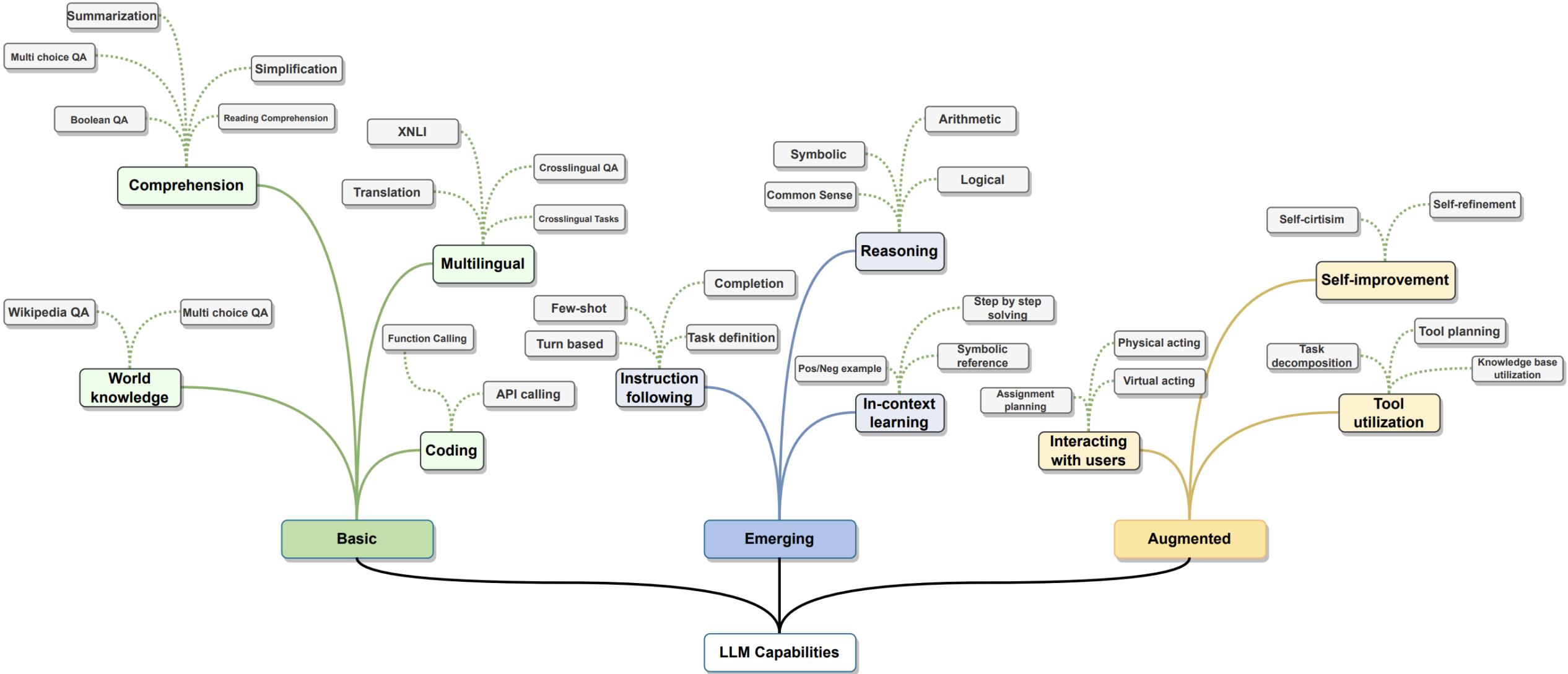
- **定义：**模型生成的文本不遵循原文 (Faithfulness) 或不符合事实 (Factualness)
- **能力优势：**大模型创造性和发散能力的体现
- **潜在影响：**影响模型可信度，容易被攻击、被滥用 (如恶意内容生成和散播谣言)，污染互联网



问：鲁迅和周树人是同一个人吗？

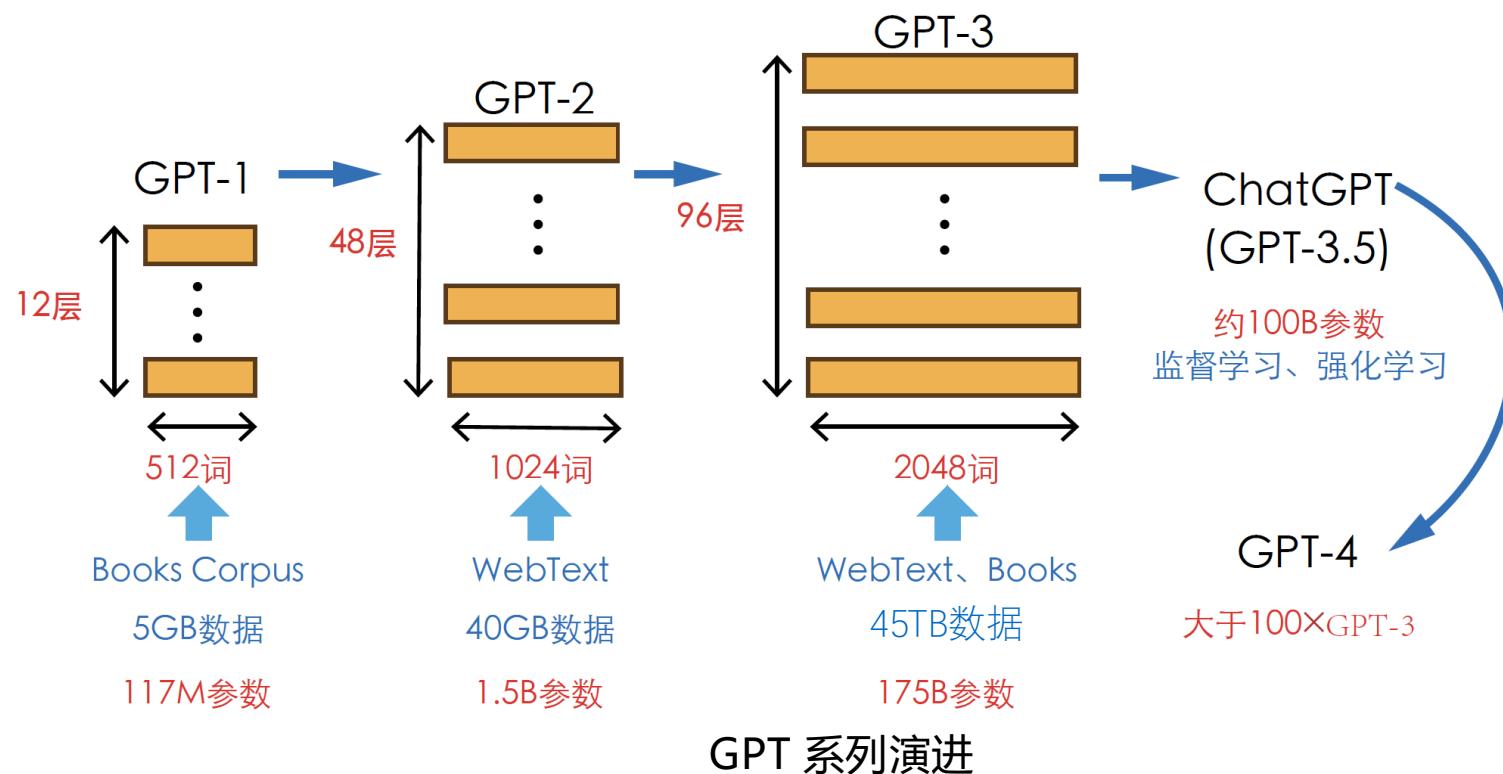
答：不是。鲁迅和周树人是两个不同的人。鲁迅是中国今现代文学巨匠，以其独特的文学风格和引人入胜的故事而闻名。周树人是中国当代著名作家，以其犀利的文字和敏锐的观察而受到推崇。

大模型的能力版图



从GPT-3到ChatGPT

- **GPT-1:** 单向自回归建模 + 有监督微调，探索“预训练+微调”范式下的自然语言任务求解能力
- **GPT-2:** 单向自回归建模 + 更多的数据、更大的模型，探索基于自然语言提示的多任务解决能力
- **GPT-3:** 探索了千亿参数规模的语言模型效果，提出了基于“上下文学习”的任务解决方法
- **GPT-3.5/ChatGPT:** 指令微调 + 人类反馈 + 对话优化

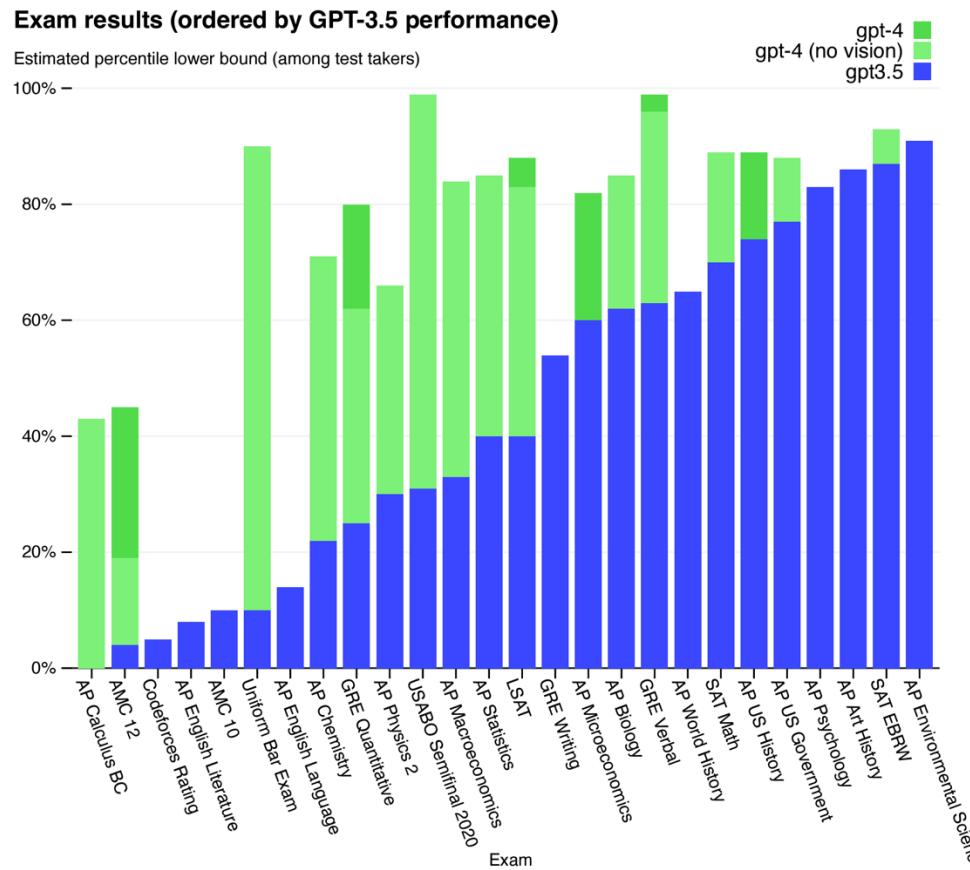


- **GPT-4:** 多模态能力，更长的上下文理解与处理能力，更高效的训练与预测，更广泛的高价值应用场景



GPT系列的版本演化

- GPT-4在各种专业和学术基准上表现出人类相当的水平，包括约前 10% 的成绩通过模拟律师资格考试
- 强大的多模态处理能力扩大了 GPT-4 的应用范围，同时生成式的幻觉、安全问题均有较大的改善



OpenAI 发布 GPT-4 使用指南

- 写出清晰的指令
- 提供参考文本
- 将复杂的任务拆分为更简单的子任务
- 给模型时间「思考」
- 使用外部工具
- 系统地测试变更

[1] Achiam, Josh, et al. Gpt-4 technical report. arXiv preprint arXiv:2303.08774.

[2] OpenAI. Prompt engineering. <https://platform.openai.com/docs/guides/prompt-engineering>.



Llama系列大模型



- 开源自回归模型，基于Transformer架构，参数规模从 7B到 405B 不等
- 商用许可：Llama 1不可免费商用，Llama 2和Llama 3可商用（需申请商用许可）

| 版本特点 | Llama-1 | Llama-2 | Llama-3 | Llama-3.1 | Llama-3.2 |
|-------|---------|-------------------|------------------|--|-------------------------|
| 模型规模 | 7B-65B | 7B-70B | 8B-70B | 8B-405B | 1B-90B |
| 数据规模 | 1.4T | 2T | 15T | 15T | 9T |
| 多模态 | × | × | × | × | √ |
| 多语言 | × | × | × | √ | √ |
| 工具调用 | × | × | × | √ | √ |
| 训练范式 | 预训练 | 预训练+指令微调 +RLHF | 预训练+指令微调 +DPO | 预训练+指令微调+DPO | 预训练+指令微调 +DPO |
| 上下文长度 | 2K | 4k | 8k | 131K | 131K |
| 主要演变 | -- | 增加后训练 | 扩展预训练 | 1. 增加模型规模 2. 增加多语言和工具能力 3. 扩展上下文长度 | 1. 增加端侧模型 2. 增加多模态能力 |

[1] Touvron, Hugo, et al. Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.

[2] Touvron, Hugo, et al. Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288.

[3] Dubey, Abhimanyu, et al. The llama 3 herd of models. arXiv preprint arXiv:2407.21783.



Llama系列大模型



- 训练数据: CommonCrawl, C4, Github, Wikipedia, books, ArXiv, and StackExchange等
- 开源地址: <https://github.com/facebookresearch/llama>
- LLM 3.1 405B 在广泛评测中超越 GPT-3.5 Turbo、大部分基准测试得分超过了 GPT-4 0125

| Category Benchmark | LLM 3.1 405B | Nemotron 4 340B Instruct | GPT-4 (0125) | GPT-4 Omni | Claude 3.5 Sonnet |
|---|--------------|--------------------------|--------------|------------|-------------------|
| General MMLU (0-shot, CoT) | 88.6 | 78.7 (non-CoT) | 85.4 | 88.7 | 88.3 |
| MMLU PRO (5-shot, CoT) | 73.3 | 62.7 | 64.8 | 74.0 | 77.0 |
| IFEval | 88.6 | 85.1 | 84.3 | 85.6 | 88.0 |
| Code HumanEval (0-shot) | 89.0 | 73.2 | 86.6 | 90.2 | 92.0 |
| MBPP EvalPlus (base) (0-shot) | 88.6 | 72.8 | 83.6 | 87.8 | 90.5 |
| Math GSM8K (8-shot, CoT) | 96.8 | 92.3 (0-shot) | 94.2 | 96.1 | 96.4 (0-shot) |
| MATH (0-shot, CoT) | 73.8 | 41.1 | 64.5 | 76.6 | 71.1 |
| Reasoning ARC Challenge (0-shot) | 96.9 | 94.6 | 96.4 | 96.7 | 96.7 |
| GPQA (0-shot, CoT) | 51.1 | - | 41.4 | 53.6 | 59.4 |
| Tool use BFCL | 88.5 | 86.5 | 88.3 | 80.5 | 90.2 |
| Nexus | 58.7 | - | 50.3 | 56.1 | 45.7 |
| Long context ZeroSCROLLS/QuALITY | 95.2 | - | 95.2 | 90.5 | 90.5 |
| InfiniteBench/En.MC | 83.4 | - | 72.1 | 82.5 | - |
| NIH/Multi-needle | 98.1 | - | 100.0 | 100.0 | 90.8 |
| Multilingual Multilingual MGSM (0-shot) | 91.6 | - | 85.9 | 90.5 | 91.6 |

| Category Benchmark | LLM 3.1 8B | Gemma 2 9B IT | Mistral 7B Instruct | LLM 3.1 70B | Mixtral 8x22B Instruct | GPT 3.5 Turbo |
|---|------------|------------------------|---------------------|-------------|------------------------|---------------|
| General MMLU (0-shot, CoT) | 73.0 | 72.3 (5-shot, non-CoT) | 60.5 | 86.0 | 79.9 | 69.8 |
| MMLU PRO (5-shot, CoT) | 48.3 | - | 36.9 | 66.4 | 56.3 | 49.2 |
| IFEval | 80.4 | 73.6 | 57.6 | 87.5 | 72.7 | 69.9 |
| Code HumanEval (0-shot) | 72.6 | 54.3 | 40.2 | 80.5 | 75.6 | 68.0 |
| MBPP EvalPlus (base) (0-shot) | 72.8 | 71.7 | 49.5 | 86.0 | 78.6 | 82.0 |
| Math GSM8K (8-shot, CoT) | 84.5 | 76.7 | 53.2 | 95.1 | 88.2 | 81.6 |
| MATH (0-shot, CoT) | 51.9 | 44.3 | 13.0 | 68.0 | 54.1 | 43.1 |
| Reasoning ARC Challenge (0-shot) | 83.4 | 87.6 | 74.2 | 94.8 | 88.7 | 83.7 |
| GPQA (0-shot, CoT) | 32.8 | - | 28.8 | 46.7 | 33.3 | 30.8 |
| Tool use BFCL | 76.1 | - | 60.4 | 84.8 | - | 85.9 |
| Nexus | 38.5 | 30.0 | 24.7 | 56.7 | 48.5 | 37.2 |
| Long context ZeroSCROLLS/QuALITY | 81.0 | - | - | 90.5 | - | - |
| InfiniteBench/En.MC | 65.1 | - | - | 78.2 | - | - |
| NIH/Multi-needle | 98.8 | - | - | 97.5 | - | - |
| Multilingual Multilingual MGSM (0-shot) | 68.9 | 53.2 | 29.9 | 86.9 | 71.1 | 51.4 |

[1] Dubey, Abhimanyu, et al. The LLM 3 herd of models. arXiv preprint arXiv:2407.21783.



国产AI大模型分类及典型案例



□ 百模大战：据不完全统计，截至2024年4月底，国内共推出305个大模型

通用大模型

| | | | | | | | |
|----|-------------------|---------|-------------------|------------------------|---------------|-----------------|-----------|
| 闭源 | 文心一言 | 讯飞星火 | 通义千问 | vivo BlueLM | oppo AndesGPT | 商汤日日新 sensenova | 出门问问 序列猴子 |
| | 天工 | 字节云雀大模型 | 腾讯混元 | Scietrain 西湖心辰 西湖大模型 | Moonshot AI | 言犀 | 360智脑 |
| | 盘古大模型 | MiLM-6B | 云从科技 云从科技从容大模型 | DeepLang AI | MINIMAX | 云知声 山海 | |
| | 4Paradigm SageGPT | 澜舟科技 孟子 | intellifusion 云天书 | AISPEECH DFM | 玉言 | AiLMe | |

开源

| | | | | | | | |
|------------|-------------|---------|----------|-----------|----------|---------------------|-----------------|
| 百川Baichuan | ChatGLM2-6B | Qwen-7B | Yi-34B | XVERSE 元象 | InternLM | ChatYuan | Fengshenbang-LM |
| CPM-Bee | OpenBuddy | MOSS | TigerBot | RWKV-LM | Firefly | 中文LLaMA & Alpaca大模型 | FlagAI |

行业大模型

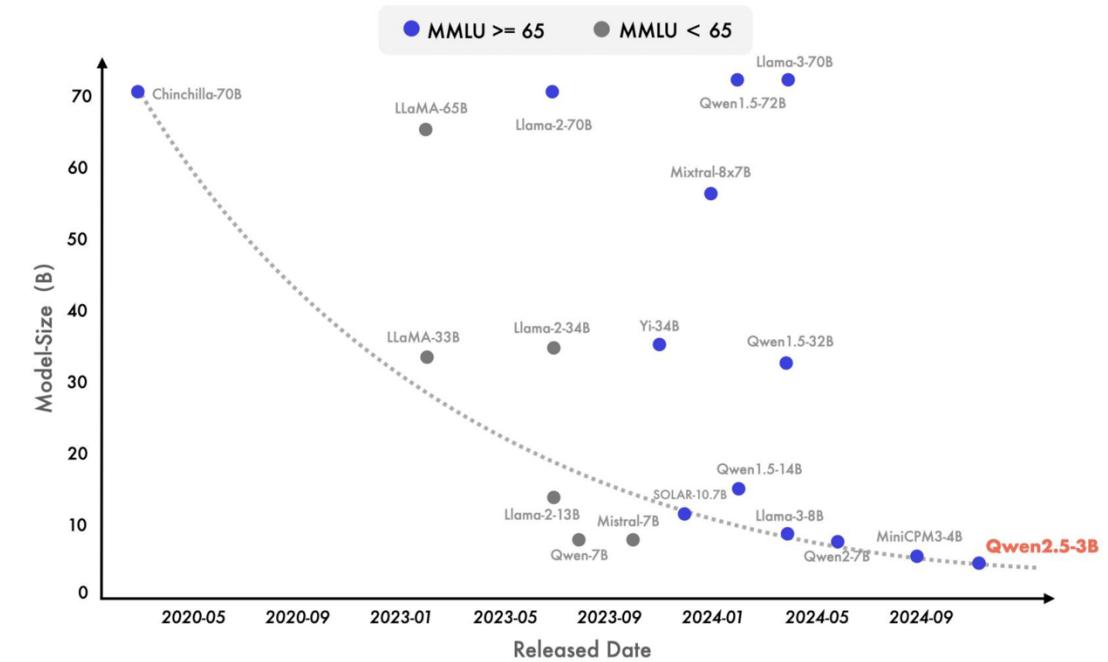
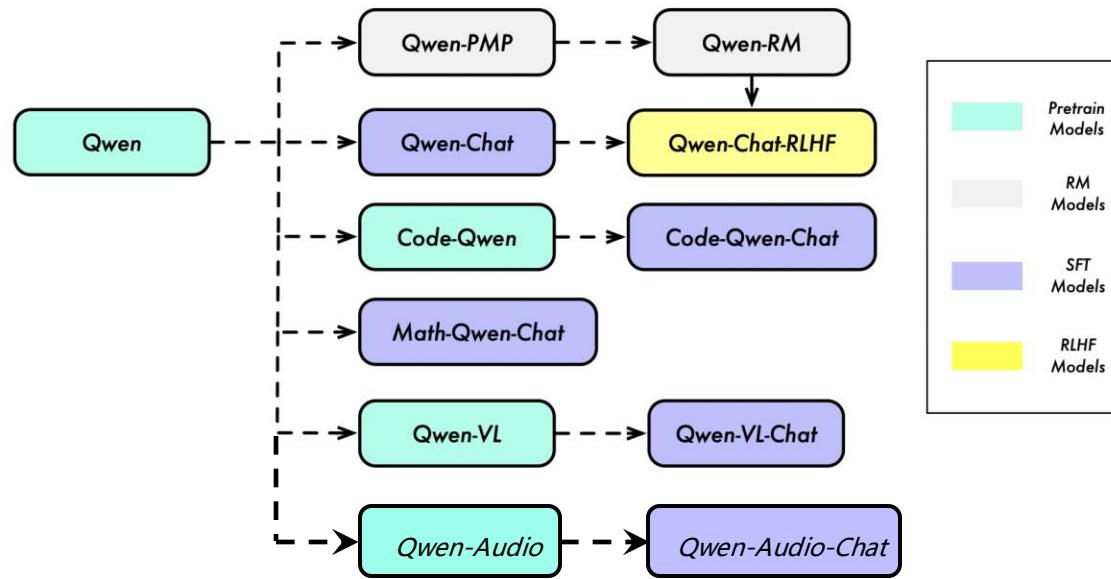
| | | | | | | |
|------|-------------|------------|-----------|----------------|------------|------------|
| 部分领域 | 医疗 | 汽车 | 教育 | 金融 | 工业 | 文化/零售/交通 |
| | 医联 / MedGPT | 理想 MindGPT | MathGPT | 蚂蚁金融大模型 | AlInno-15B | 阅文集团 妙笔大模型 |
| | 岐黄问道 | DriveGPT | 作业帮 银河大模型 | 轩辕大模型 | COSMO-GPT | 晓模型 XPT |
| | DISC-MedLLM | 广汽AI大模型 | 子曰 | SmartMore 医瞳科技 | SMore LrMo | 乐言科技 |
| | | | | | | 佳都科技 |

[1] SuperGLUE团队. 中文大模型基准测评2024年度报告.



Qwen系列大模型

- 通义千问 (Qwen) 是阿里云研发的预训练语言模型，执行理解、生成和解释人类语言、图片和文档等任务，在创意文案、办公助理、学习助手、趣味生活等方面为使用者提供丰富的交互体验
- 具备中英文理解、数学推理、代码理解等能力



[1] Bai, Jinze, et al. Qwen technical report. arXiv preprint arXiv:2309.16609.

[2] Yang, An, et al. Qwen2 technical report. arXiv preprint arXiv:2407.10671.



Qwen系列的版本演化



- Qwen1.5: Model Scaling向更大和更小扩展、MoE、AWQ/GGUF
- Qwen2: 多语言增强、长文本能力增强
- Qwen2.5: 数学、代码能力增强、多语言增强
- Scaling
 - Qwen: 1.8B, 7B, 14B, and 72B models
 - Qwen1.5: 0.5B, 1.8B, 4B, 14BA2.7B, 7B, 14B, 32B, 72B, and 110B models
 - Qwen2: 0.5B, 1.5B, 7B, 57A14B, and 72B models
 - Qwen2.5: 0.5B, 1.5B, 3B, 7B, 14B, 32B, and 72B models

版本迭代特点

- 结构优化 (MoE提高训练效率)
- 预训练数据的数量扩充与质量提升
- 丰富Post-training阶段涵盖任务和领域
- 窗口长度的扩展 (训练数据和训练策略两方面)
- 多语言语向扩增与能力增强
- 专项能力增强 (训练专家模型+以专家模型带动基模)
- 模态扩展 (Qwen-VL、Qwen-Audio)



DeepSeek 系列大模型

1

2023.11

DeepSeek LLM

参数规模: 70B、67B参数

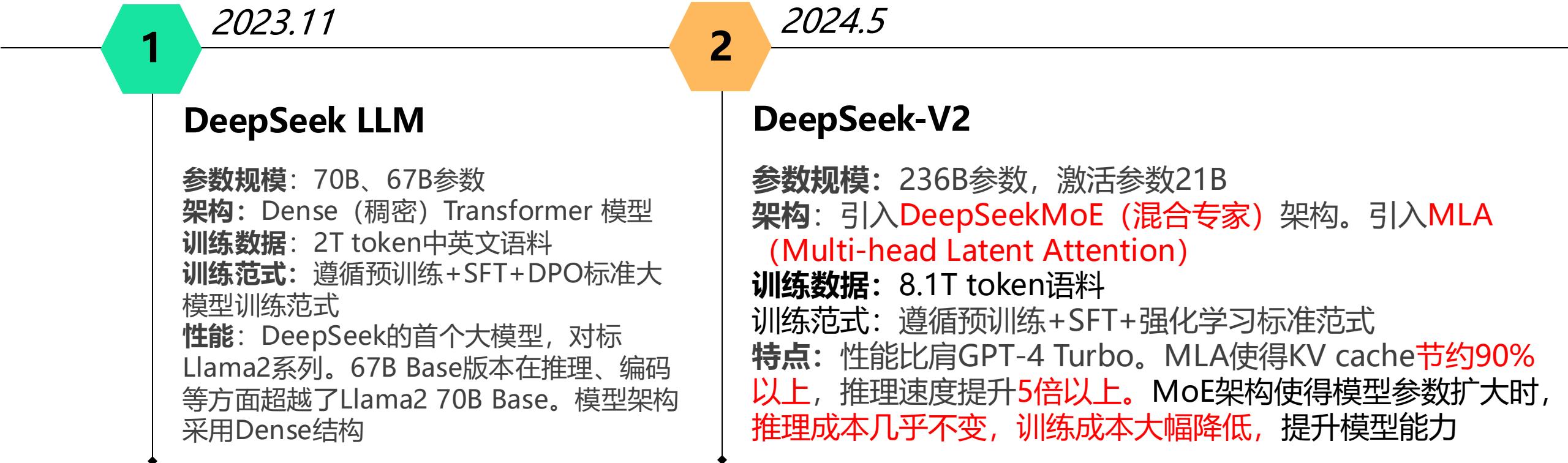
架构: Dense (稠密) Transformer 模型

训练数据: 2T token中英文语料

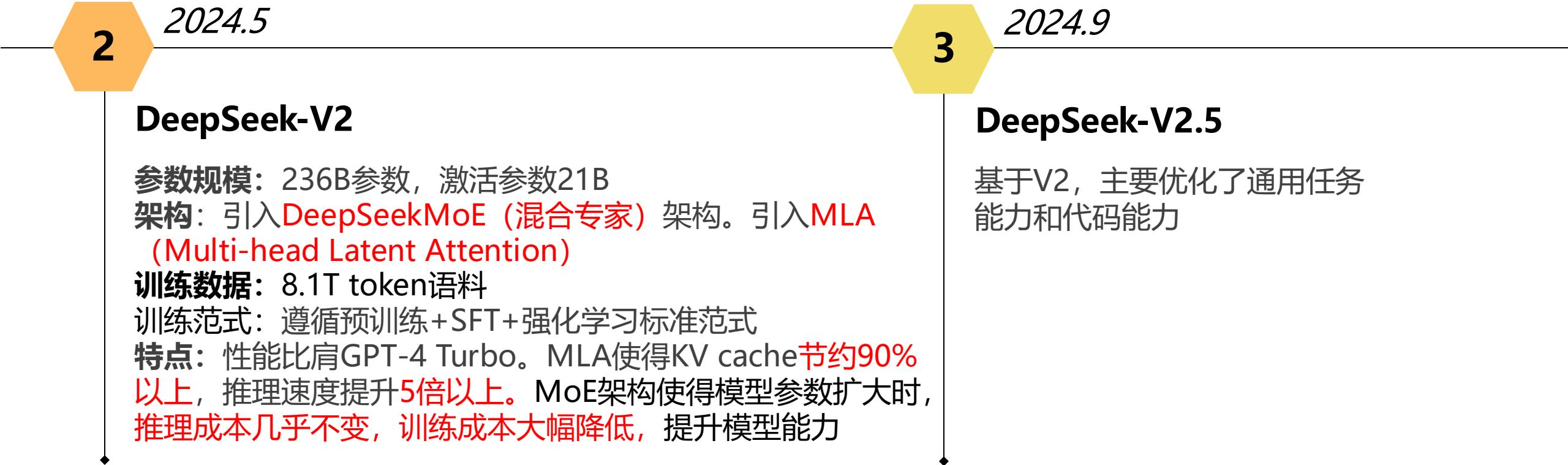
训练范式: 遵循预训练+SFT+DPO标准大
模型训练范式

性能: DeepSeek的首个大模型，对标
Llama2系列。67B Base版本在推理、编码
等方面超越了Llama2 70B Base。模型架构
采用Dense结构

DeepSeek 系列大模型



DeepSeek 系列大模型



DeepSeek 系列大模型



DeepSeek 系列大模型

4

2024.12

DeepSeek-V3

参数规模: 671B参数, 激活参数37B

训练数据: 14.8T token语料

架构: 在V2基础上, 引入Auxiliary-loss-free strategy来进行专家均衡。引入Multi-Token Prediction training提升模型性能和加速推理。引入FP8混合精度降低训练和推理开销

训练数据: 14.8T token语料

训练范式: 遵循预训练+SFT+DPO标准大模型训练范式

特点: 性能超越主流的Llama3系列和Qwen2.5系列, 且训练开销和推理开销大幅降低

5

2025.01

DeepSeek-R1

基座模型: DeepSeek-V3-Base

训练数据: 14.8T token语料

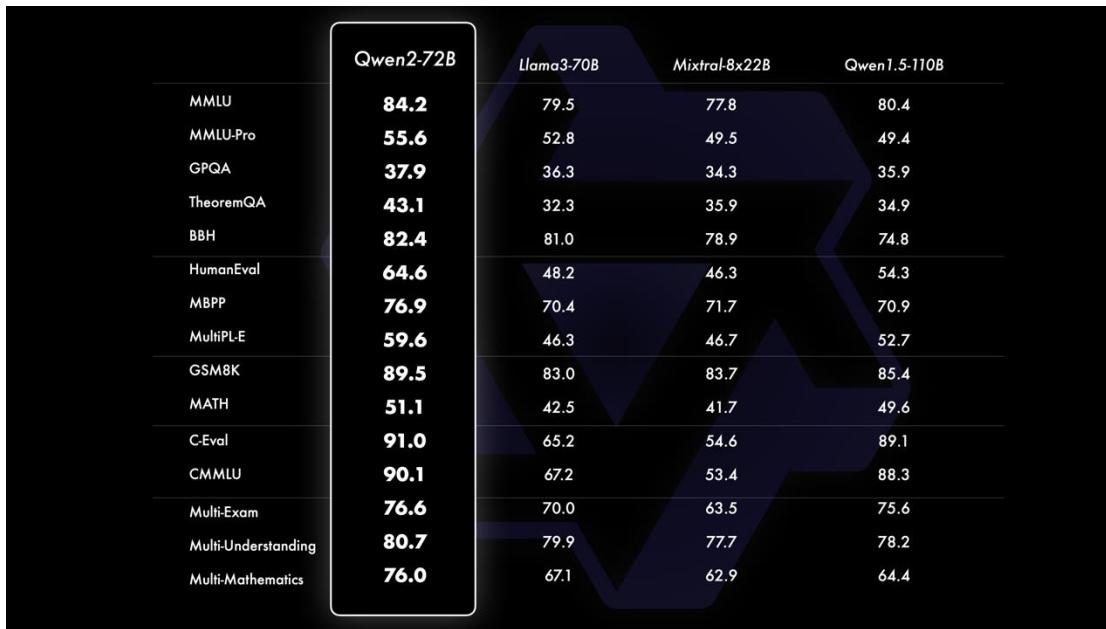
训练范式: 在base模型上直接进行多阶段训练、冷启动训练和强化学习

特点: 首次达到了类GPT4-O1的推理能力

部署大模型

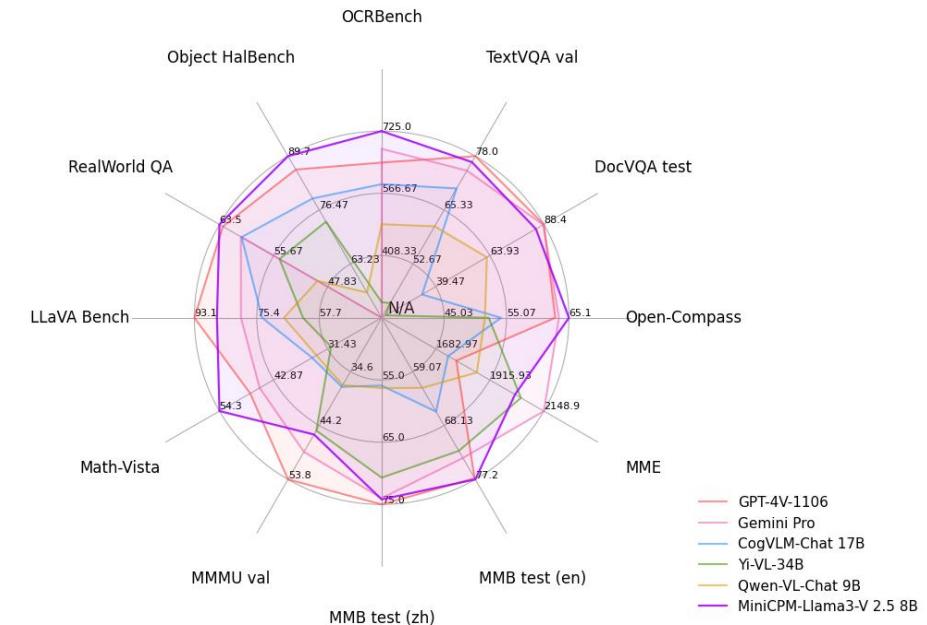
□ 大模型类型：

- 大语言模型：Qwen
- 多模态模型：MiniCPM



□ 部署方式：

- API调用
- 本地部署



部署大模型（API调用）



□ 获得大模型调用权限

- 通义千问: <https://help.aliyun.com/zh/dashscope/developer-reference/quick-start>
- DeepSeek: <https://www.deepseek.com>
- Llama: <https://github.com/Meta-Llama/llama>
- 智谱AI: <https://open.bigmodel.cn/>
- Openai: <https://platform.openai.com/playground>
- 其他: 文心一言、百川大模型

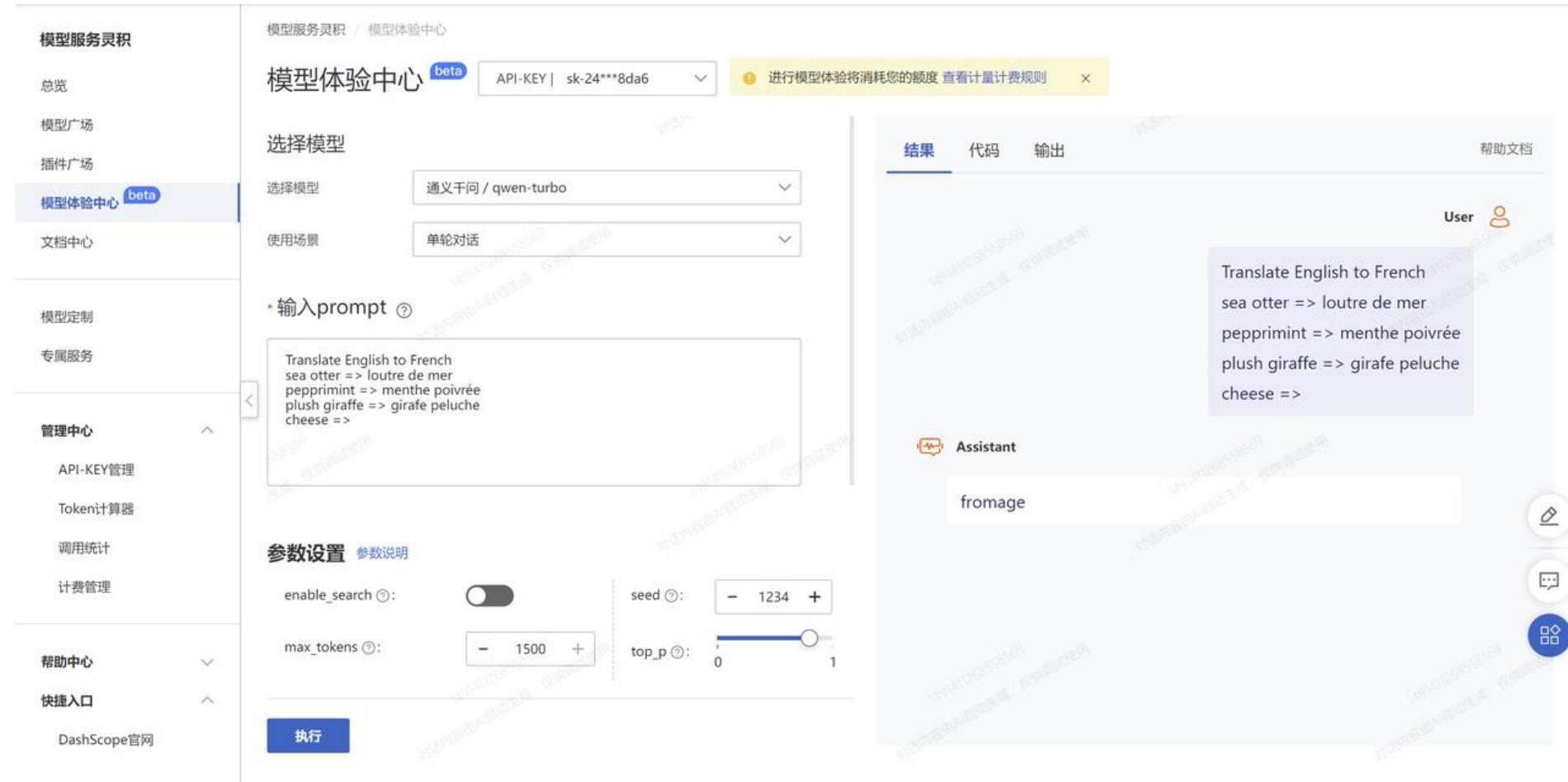
*基本流程: 开通服务获得API-KEY (获赠计算额度) , 使用API Key调用服务 (其中OpenAI需要科学手段)



部署大模型（API调用）

□ 通过GUI界面调用（适合案例测试）

- <https://help.aliyun.com/zh/dashscope/developer-reference/quick-start>
- <https://huggingface.co/spaces/Qwen/Qwen2-72B-Instruct>



The screenshot shows the DashScope Model Experience Center interface. On the left sidebar, under the 'Model Experience Center' section, the '通义千问 / qwen-turbo' model and '单轮对话' (Single-turn dialogue) scenario are selected. In the main area, there's a text input field labeled '输入prompt' containing the English sentence 'Translate English to French sea otter => loutre de mer peppermint => menthe poivrée plush giraffe => girafe peluche cheese =>'. Below this, a 'Parameters' section includes settings for 'enable_search', 'seed' (set to 1234), 'max_tokens' (set to 1500), and 'top_p' (set to 1.0). A large blue '执行' (Execute) button is at the bottom. To the right, the 'User' section shows the translated French text: 'Translate English to French sea otter => loutre de mer peppermint => menthe poivrée plush giraffe => girafe peluche cheese =>'. The 'Assistant' section shows the input 'fromage'.



部署大模型（API调用）



□ 通过代码调用：

普通调用：

```
# For prerequisites running the following sample, visit https://help.aliyun.com/document_
import dashscope
def sample_sync_call():
    prompt_text = '用萝卜、土豆、茄子做饭，给我个菜谱。'
    resp = dashscope.Generation.call(
        model='qwen-turbo',
        prompt=prompt_text
    )
    # The response status_code is HttpStatus.OK indicate success,# otherwise indicate request
    print(resp.output)# The output textprint(resp.usage)# The usage informationelse:
        print(resp.code)# The error code.print(resp.message)# The error message.
sample_sync_call()
```

流式调用：

```
# For prerequisites running the following sample, visit https://help.aliyun.com/document_
import dashscope
def sample_sync_call_streaming():
    prompt_text = '用萝卜、土豆、茄子做饭，给我个菜谱。'
    response_generator = dashscope.Generation.call(
        model='qwen-turbo',
        prompt=prompt_text,
        stream=True,
        top_p=0.8)
    head_idx = 0
    for resp in response_generator:
        paragraph = resp.output['text']
        print("\r%s" % paragraph[head_idx:len(paragraph)], end='')
        if(paragraph.rfind('\n') != -1):
            head_idx = paragraph.rfind('\n') + 1
sample_sync_call_streaming()
```

□ 环境准备（可选）：

- VS Code开发工具：
<https://code.visualstudio.com/>
- 利用Miniconda进行Python环境管理：
<https://docs.anaconda.com/free/miniconda/miniconda-install>



部署大模型（本地部署）



- 大模型本地部署开源平台：Ollama, LM Studio
- Web与API调用：OpenAI, Dashscope API
- 大模型本地私有化部署：Flask, MindIE

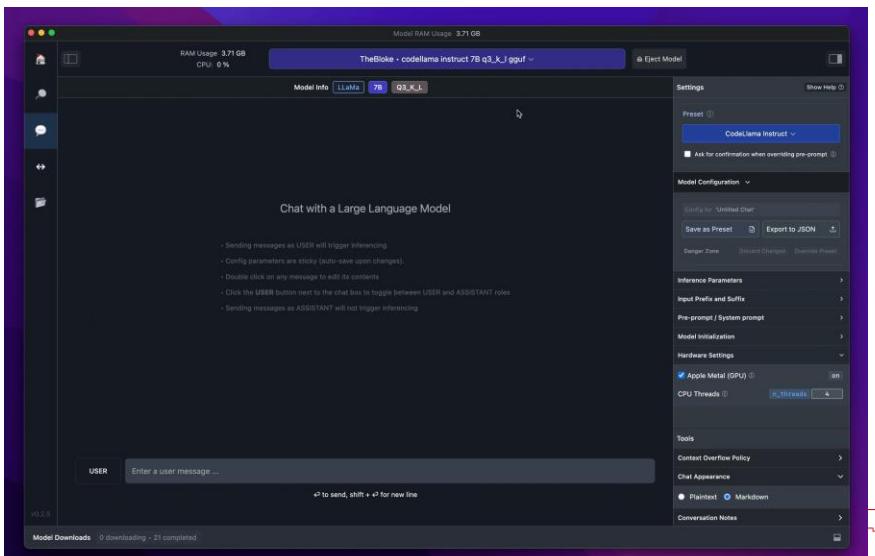
Ollama <https://ollama.com/>

- 用户体验：以其简单和易于安装而闻名，特别适合初学者和非技术用户。
- 可定制性：提供创建定制语言模型和运行各种预训练模型的灵活性。
- 开源：完全开源，有助于提高透明度和社区参与。
- 支持的操作系统：适用于 macOS、Linux 和 Windows（预览版），且处理器支持 AVX。

LM Studio <https://lmstudio.ai/>

- 功能集：提供更广泛的功能集，包括发现、下载和运行本地 LLM，以及应用内聊天界面以及与 OpenAI 兼容的本地服务器的兼容性。
- UI 友好性：与 Ollama 相比，被认为对用户界面更加友好。
- 模型选择：模型目录：提供来自 Hugging Face 等来源的更广泛的模型选择。
- 支持的操作系统：适用于 M1/M2/M3 Mac 或具有支持 AVX2 处理器的 Windows PC。Linux 已推出测试版。

```
(base) zhangzhuosheng@zhangzhuoshengdeiMac ~ % ollama run qwen2:7b
pulling manifest
pulling manifest
pulling manifest
pulling 43f7a214e532... 100% [██████████]
pulling 62fbfd9ed093... 100% [██████████]
pulling c156170b718e... 100% [██████████]
pulling f02d72bb242... 100% [██████████]
pulling 648f809ced2b... 100% [██████████]
verifying sha256 digest
writing manifest
removing any unused layers
success
>>> 你是谁
我是阿里云开发的一款超大规模语言模型，我叫通义千问。作为一个AI助手，我的目标是帮助用户获得准确、有用的信息，解决他们的问题和困惑。无论是关于知识查询、信息解答、问题解释，还是进行创作辅助、与用户进行对话交流等，我都将尽提供高质量的服务。如果您有任何问题或需要帮助，请随时告诉我！
>>> 鸡兔同笼，共35只头，94只脚，问鸡兔各多少？
...
这个问题是典型的中国数学古题“鸡兔同笼”问题的变体。
设鸡的数量为x，兔子的数量为y。
根据题目描述，我们有两个方程：
1. 鸡和兔子的头数之和为35: \((x + y = 35)\)
2. 鸡和兔子的脚数之和为94: \((2x + 4y = 94)\) (因为鸡有两只脚，兔子有四只脚)
我们可以从第一个方程解出 \((x = 35 - y)\)。
然后将 \((x)\) 的值代入第二个方程中:
\([2(35 - y) + 4y = 94]\)
```



部署大模型（本地部署）



配置要求

| 模型 | 推理显存(GB) | 微调显存(GB) |
|-----------------------------|----------|----------|
| Qwen2-0.5B-Instruct | 最低2.7G | 最低6G |
| Qwen2-1.5B-Instruct | 最低5.2G | 最低8.5G |
| Qwen2-7B-Instruct | 最低16.7G | 最低20G |
| Qwen2-72B-Instruct | 最低143G | 最低156G |
| Qwen2-MoE-57B-A14B-Instruct | 最低116G | 最低122G |

显存要求

| | | | | |
|-------------------|--------|-------------------|--------|-------|
| 7b | ▼ | 7b | latest | 4.4GB |
| 72b | | 72b | | 41GB |
| 7b | latest | 7b | latest | 4.4GB |
| 1.5b | | 1.5b | | 935MB |
| 0.5b | | 0.5b | | 352MB |
| 72b-instruct | | 72b-instruct | | 41GB |
| 72b-instruct-fp16 | | 72b-instruct-fp16 | | 145GB |

硬盘要求



部署大模型（本地部署）



- 1. 安装Ollama
 - 访问官方网站：打开浏览器，访问Ollama官方网站。
 - 选择版本：根据您的操作系统选择相应的安装包。Ollama支持MacOS、Linux和Windows
 - 下载安装包：点击Download按钮，下载适用于您设备的Ollama安装包
 - 安装Ollama：根据下载的安装包格式，运行安装程序或解压缩文件到您选择的目录
- 2. 运行特定模型 (ollama run qwen2:7b)

```
(base) zhangzhuosheng@zhangzhuoshengdeiMac ~ % ollama run qwen2:7b
[pulling manifest
pulling manifest
pulling manifest
pulling 43f7a214e532... 21% [██████████] | 929 MB/4.4 GB 5.6 MB/s 10m19s]
```



部署大模型（本地部署）



- 1. 安装Ollama
- 2. 运行特定模型 (ollama run qwen2:7b)

```
(base) zhangzhuosheng@zhangzhuoshengdeiMac ~ % ollama run qwen2:7b
pulling manifest
pulling manifest
pulling manifest
pulling 43f7a214e532... 100% [██████████] 4.4 GB
pulling 62fbfd9ed093... 100% [██████████] 182 B
pulling c156170b718e... 100% [██████████] 11 KB
pulling f02dd72bb242... 100% [██████████] 59 B
pulling 648f889ced2b... 100% [██████████] 485 B
verifying sha256 digest
writing manifest
removing any unused layers
success
>>> 你是谁
我是阿里云开发的一款超大规模语言模型，我叫通义千问。作为一个AI助手，我的目标是帮助用户获得准确、有用的信息，解决他们的问题和困惑。无论是关于知识查询、信息解答、问题解释，还是进行创作辅助、与用户进行对话交流等，我都将尽力提供高质量的服务。如果您有任何问题或需要帮助，请随时告诉我！
>>> 鸡兔同笼，共35只头，94只脚，问鸡兔各多少？
...
这个问题是典型的中国数学古题“鸡兔同笼”问题的变体。
```

设鸡的数量为 x ，兔子的数量为 y 。
根据题目描述，我们有以下两个方程：

1. 鸡和兔子的头数之和为35: $(x + y = 35)$
2. 鸡和兔子的脚数之和为94: $(2x + 4y = 94)$ (因为鸡有两只脚，兔子有四只脚)

我们可以从第一个方程解出 $(x = 35 - y)$ 。

然后将 (x) 的值代入第二个方程中：

```
\[2(35 - y) + 4y = 94]
```

展开得：

```
\[70 - 2y + 4y = 94]
```

合并同类项：

```
\[2y = 24]
```

解得 $(y = 12)$ ，即兔子有12只。

将 (y) 的值代回任一原方程求解 (x) :

```
\[x + 12 = 35]
```

解得 $(x = 23)$ ，即鸡有23只。

所以，鸡和兔的数量分别是23只和12只。



部署大模型（本地部署）



- 1. 安装Ollama
- 2. 运行特定模型 (ollama run qwen2:7b)

```
(base) zhangzhuosheng@zhangzhuoshengdeiMac ~ % ollama run qwen2:7b
pulling manifest
pulling manifest
pulling manifest
pulling 43f7a214e532... 100% [██████████] 4.4 GB
pulling 62fbfd9ed093... 100% [██████████] 182 B
pulling c156170b718e... 100% [██████████] 11 KB
pulling f02dd72bb242... 100% [██████████] 59 B
pulling 648f889ced2b... 100% [██████████] 485 B
verifying sha256 digest
writing manifest
removing any unused layers
success
>>>
>>> 你是谁
我是阿里云开发的一款超大规模语言模型，我叫通义千问。作为一个AI助手，我的目标是帮助用户获得准确、有用的信息，解决他们的问题和困惑。无论是关于知识查询、信息解答、问题解释，还是进行创作辅助、与用户进行对话交流等，我都将尽力提供高质量的服务。如果您有任何问题或需要帮助，请随时告诉我！
>>> 鸡兔同笼，共35只头，94只脚，问鸡兔各多少？
...
这个问题是典型的中国数学古题“鸡兔同笼”问题的变体。
设鸡的数量为x，兔子的数量为y。
根据题目描述，我们有以下两个方程：
1. 鸡和兔子的头数之和为35:  $(x + y = 35)$ 
2. 鸡和兔子的脚数之和为94:  $(2x + 4y = 94)$  (因为鸡有两只脚，兔子有四只脚)
我们可以从第一个方程解出  $(x = 35 - y)$ 。
然后将  $(x)$  的值代入第二个方程中：

$$2(35 - y) + 4y = 94$$

展开得：

$$70 - 2y + 4y = 94$$

合并同类项：

$$2y = 24$$

解得  $(y = 12)$ ，即兔子有12只。
将  $(y)$  的值代回任一原方程求解  $(x)$ ：

$$(x + 12 = 35)$$

解得  $(x = 23)$ ，即鸡有23只。
所以，鸡和兔的数量分别是23只和12只。
```

进阶：

1. 使用open web UI配置WebUI
2. 基于AnythingLLM搭建本地知识库



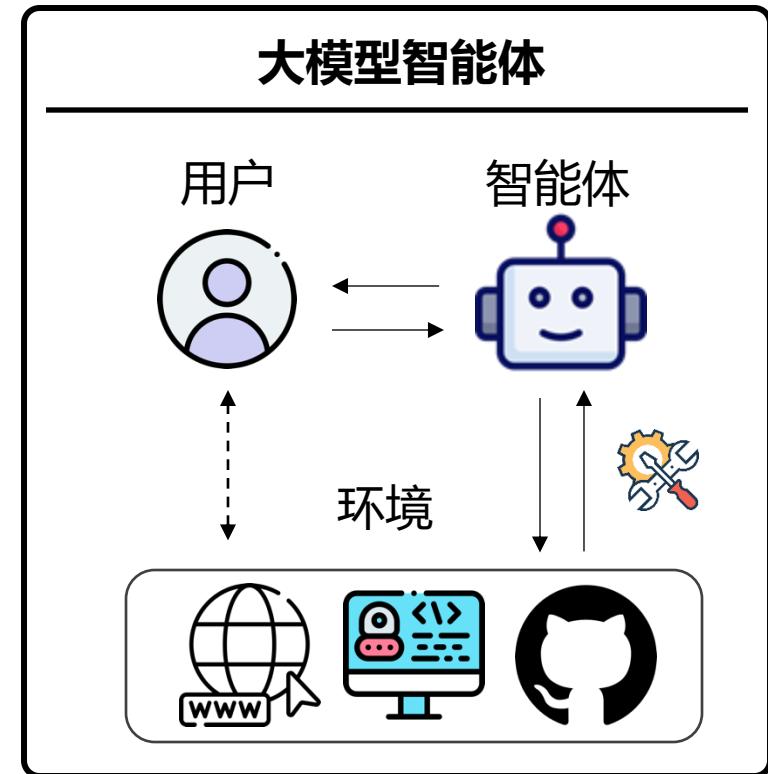
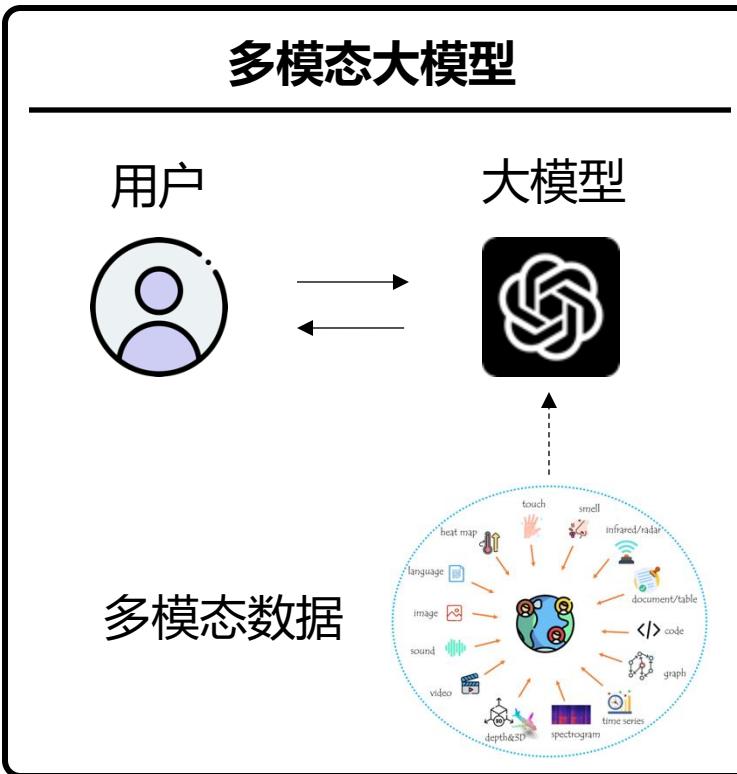
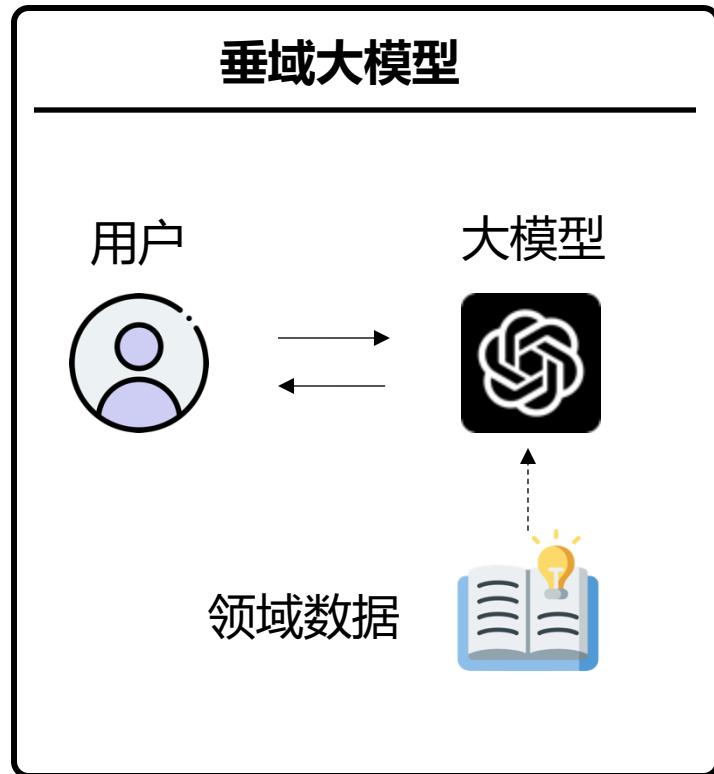
目录



- 大模型概述
 - 发展概览
 - 大模型范畴
- 预训练模型
 - 主要类型
 - 模型架构
- 大规模模型
 - 范式变化
 - 关键技术
 - 主要代表
 - 部署方式
- 概念延伸
 - 垂直大模型
 - 多模态基座
 - 智能体基座



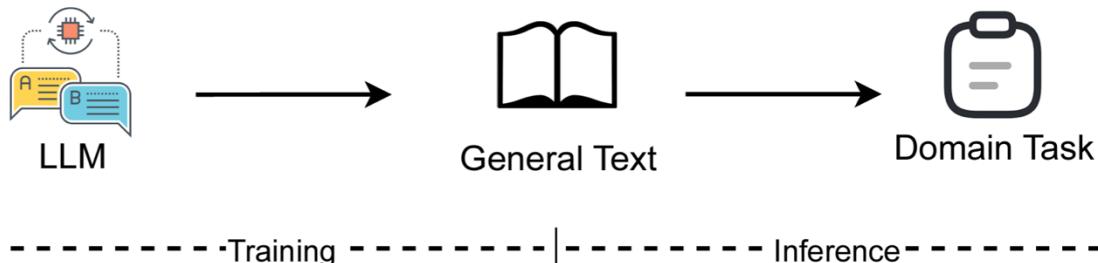
大模型概念延伸



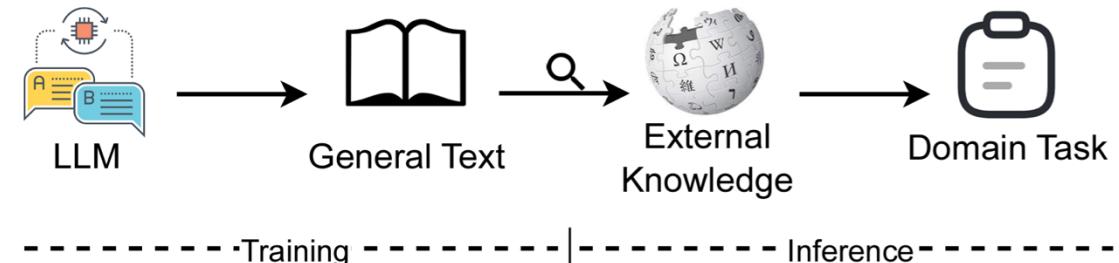
垂域大模型

- **基本范式:** 在通用大模型基础上进行领域训练增强，专业性强、落地速度快、成本效益高、合规性好
- **训练方式:** 领域数据训练（继续预训练、微调、提示）、检索增强等
- **常见领域:** 教育、医疗、法律、金融等

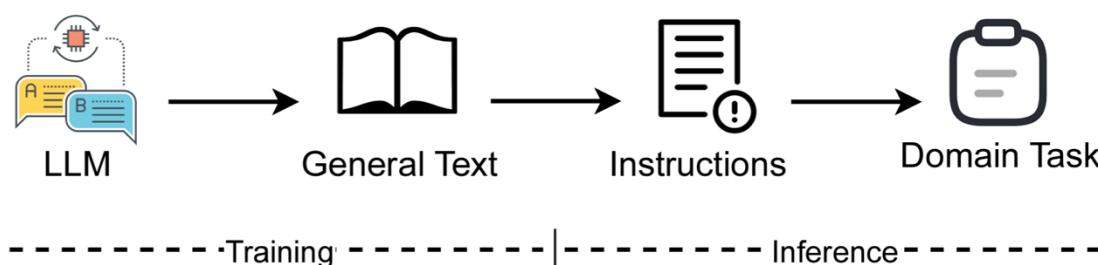
a) Deploying LLM without Domain Specialization



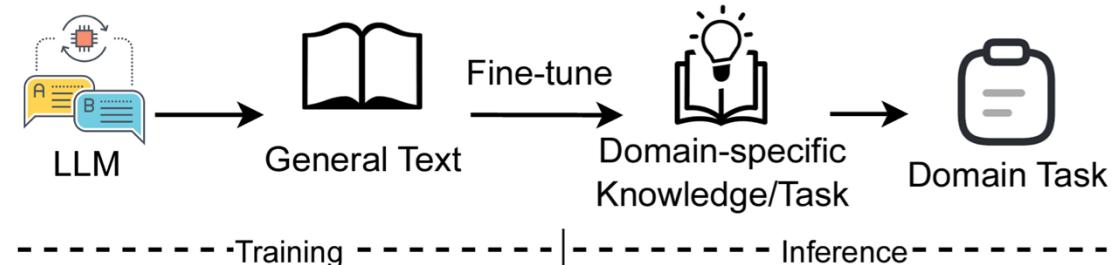
b) Retrieval Augmented Domain Specialization



c) Instruction-crafting-based Domain Specialization



d) Knowledge-updated Domain Specialization

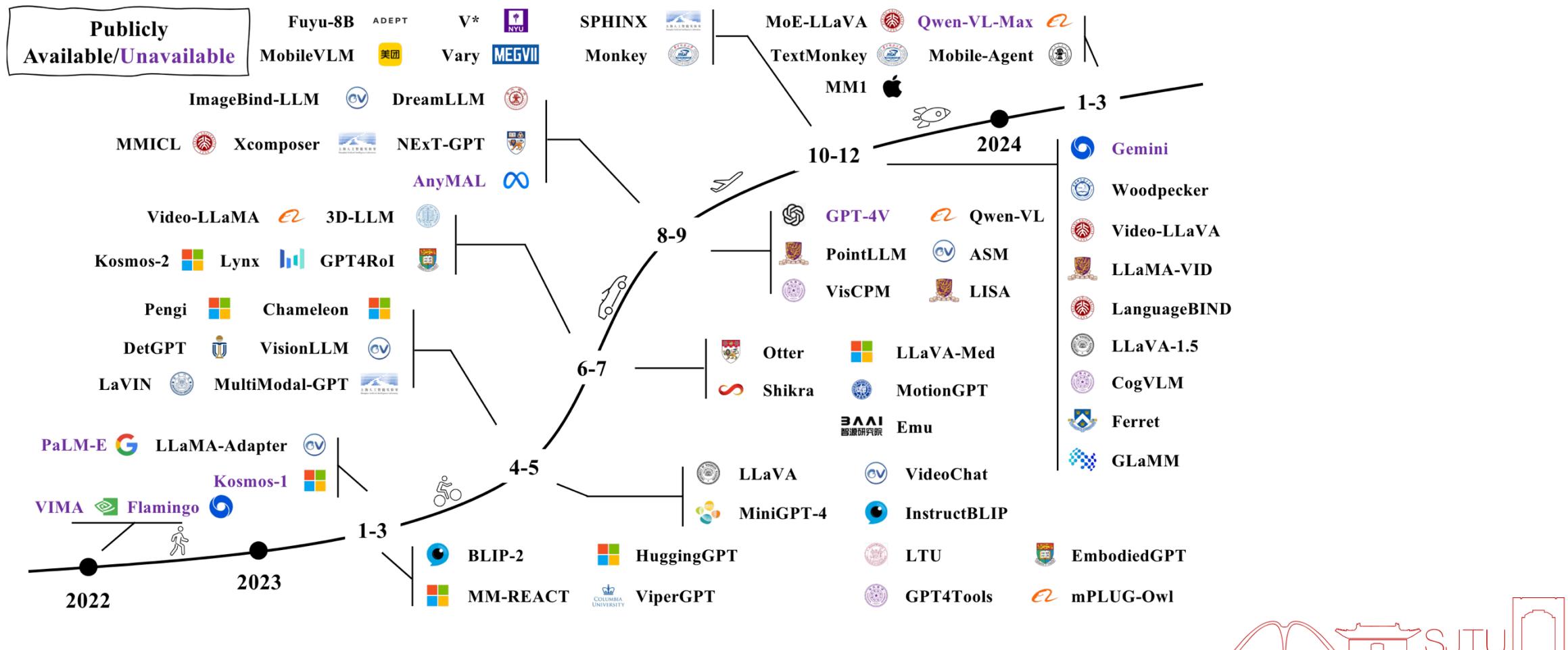


[1] Ling, Chen, et al. Domain specialization as the key to make large language models disruptive: A comprehensive survey. arXiv preprint arXiv:2305.18703.



多模态大模型的前沿发展

- 多模态大模型的能力突破，成为大模型前沿发展的新趋势和焦点
- 极大拓展大模型能力边界和应用场景，智能跃迁的下一个关键引擎



多模态大模型的能力样例



上海交通大学
SHANGHAI JIAO TONG UNIVERSITY

Unesco announces its newest geoparks around the world

9 April 2024
By Lynn Brown, Features correspondent

Please identify the text in the picture.

MiniCPM-Llama3-V 2.5

Title:
Unesco announces its newest geoparks around the world

Date and Author:
9 April 2024
By Lynn Brown, Features correspondent

Image Caption:
Unesco Geoparks represent a balance of unique geological features, cultural touchpoints and a focus on sustainability (Credit: Chris Karagelis)

Body Text:
From dinosaur fossils in Brazil to the soaring meteors of Meteora, these 18 new geological sites highlight a particular region's natural, cultural and intangible heritage.

Travellers interested in deep dives into geology, culture and sustainability have several new destinations to place on their bucket lists. Unesco just announced the designation of 18 new Global Geoparks, bringing the total to 213 sites representing 48 countries around the world.

From crystal-clear karst lakes to an archipelago formed during the last ice age, these parks are more than just sites with unique geological features; they also serve to highlight a particular region's natural, cultural and intangible heritage. "Unesco Global Geoparks represent landscapes where internationally significant geological heritage intertwines with human history and supports rich biodiversity," explains Antônio Abreu, director of the Division of Ecological Sciences at Unesco. "Each new geopark tells a unique story, using geology as its lens."

Here are five of the newest geoparks on Unesco's list:

面壁Ultra对齐技术
大模型上分神器！

面对大模型，我们如何才能提升其推理能力？Face-U2B的对齐技术，通过将知识迁移到模型上，从而提升模型的推理能力。同时，我们还提供了对齐工具，帮助您轻松完成对齐工作。

高效大模型工厂

从语义理解到生成，再到推理，我们打造了一个高效的大模型工厂。通过我们的平台，您可以方便地进行模型的训练、评估和部署。我们提供了一系列的工具，帮助您轻松完成整个流程。

* The colored underlines are for illustration purposes only and are not used as model input.

MiniCPM-Llama3-V 2.5

| 模型名称 | 发布时间 | 发布机构 | 语言 | 参数规模 | Tokens规模 | 模型结构 | 是否开源 |
|------------|---------|-------------------|-----|-----------|----------|------------|------|
| T5 | 2019-10 | Google | 英 | 13B | T5-style | √ | |
| GPT-3 | 2020-05 | OpenAI | 英 | 175B | 300B | GPT-style | × |
| LaMDA | 2021-05 | Google | 英 | 137B | 2.8T | GPT-style | × |
| Jurassic | 2021-08 | AI21 | 英 | 178B | 300B | GPT-style | × |
| MT-NLG | 2021-10 | Microsoft, NVIDIA | 英 | 530B | 270B | GPT-style | × |
| ERNIE 3.0 | 2021-12 | Baidu | 中 | 260B | 300B | Multi-task | × |
| Gopher | 2021-12 | DeepMind | 英 | 280B | 300B | GPT-style | × |
| Chinchilla | 2022-04 | DeepMind | 英 | 70B | 1.4T | GPT-style | × |
| PaLM | 2022-04 | Google | 多语言 | 540B | 780B | GPT-style | × |
| OPT | 2022-05 | Meta | 英 | 125B-175B | 180B | GPT-style | √ |
| BLOOM | 2022-07 | BigScience | 多语言 | 176B | 366B | GPT-style | √ |
| GLM-130B | 2022-08 | Tsinghua | 中、英 | 130B | 400B | GLM-style | √ |
| LLaMA | 2023-02 | Meta | 多语言 | 7B-65B | 1.4T | GPT-style | √ |

请将表格转成 markdown 格式

How does UltraEval work?

Data preparation: Raw data (Knowledge, Code, ...), Official data source (GitHub, Hugging Face, ...), Collect. **Evaluation process**: Prompt input, Model deployment, Metric calculation, Score. **Model support**: API, Local URL, vLLM/Torch, HF Model, Custom Model, Post-process.

MiniCPM-Llama3-V 2.5

UltraEval appears to be a system or process designed to evaluate data, likely in the context of a scientific or research project. The flowchart outlines the steps involved in this process:

- Data Preparation:** This stage involves collecting and organizing raw data, which can come from official data sources such as GitHub or Hugging Face. These are represented by icons indicating their origin.
- Evaluation Process:** After the data is prepared, it undergoes an evaluation process that includes several steps:
 - Prompt Input:** This could involve providing prompts or instructions for the model.
 - Model Deployment:** Here, the model is likely trained or configured to perform evaluations based on the input.
 - Local URL:** This step may involve accessing or using a local URL, possibly for deploying the model or retrieving results.
 - Post-process:** This could refer to further processing of the model's output, such as cleaning, filtering, or formatting data.
- Metric Calculation:** Following the post-process, metrics are calculated to evaluate the performance of the model. This is indicated by a graph icon, suggesting quantitative analysis.
- Score:** The final stage is to determine the score, which is likely the outcome of the metric calculation. This score would reflect the model's performance or the quality of the data after evaluation.
- Model Support:** Throughout the process, there is support for various models, including vLLM/Torch, HF Model, and Custom Model. These models are likely used at different stages of the evaluation process.

In summary, UltraEval seems to be a structured approach to evaluating data using machine learning models, with a focus on performance metrics and customization options for different types of models.



67

多模态大模型的前沿发展

技术概要

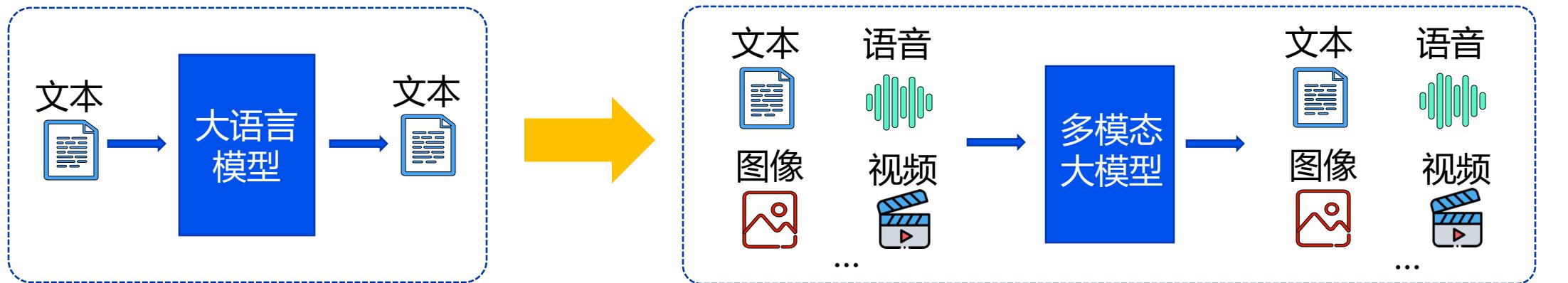
主要目标: 基于大数据和人类反馈进行学习，与用户通过对话进行交互，能够处理多模态信息及多种任务，深层推理与常识运用能力大幅度提升

关键技术: 多模态模型架构、多模态指令遵循、多模态推理、轻量化部署、多模态对齐等

里程碑事件

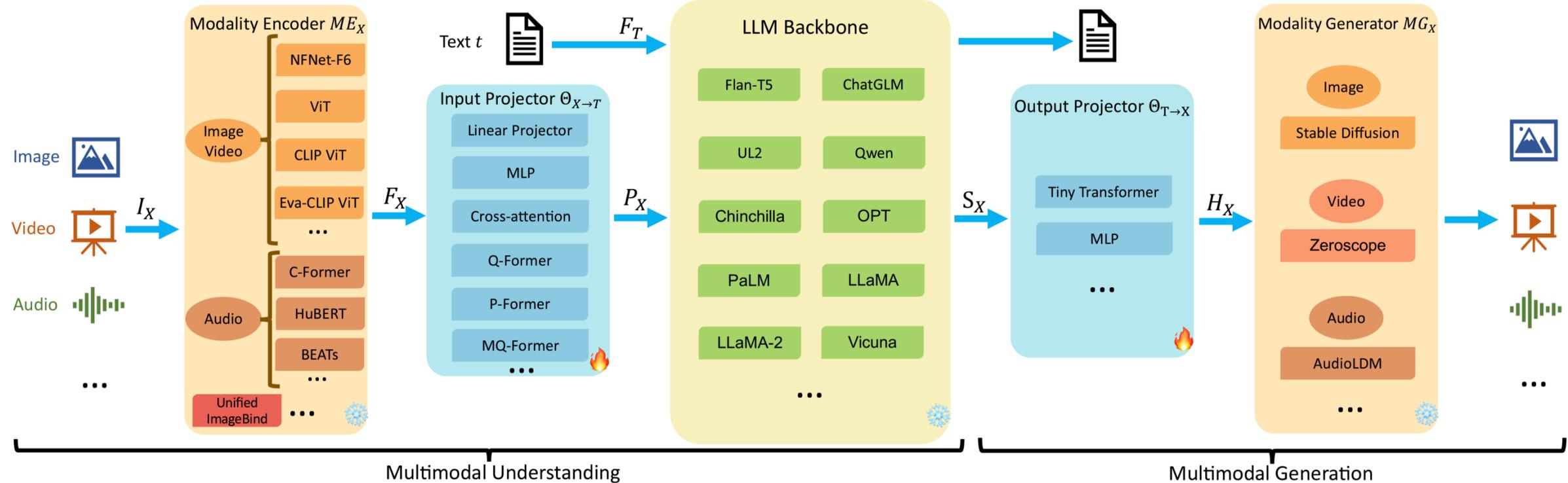
- 2023/03/15: OpenAI 发布**多模态对话模型 GPT-4**
- 2023/09/20: OpenAI 发布**文生图模型 DALL·E 3**
- 2023/09/25: ChatGPT 全面整合**多模态能力**
- 2023/12/06: Google 发布**多模态模型 Gemini**
- 2024/03/04 : Anthropic发布**多模态模型 Claude 3**
- 2024/05/13: OpenAI发布原生**多模态模型 GPT-4o**

工作范式的变化



多模态大模型的前沿发展

- **通用工作模式:** 多模态编码 > 特征融合模块 > 多模态解码器, 支持多模态输入与多模态生成
- **常见模态:** 文本、图像、视频、语音等

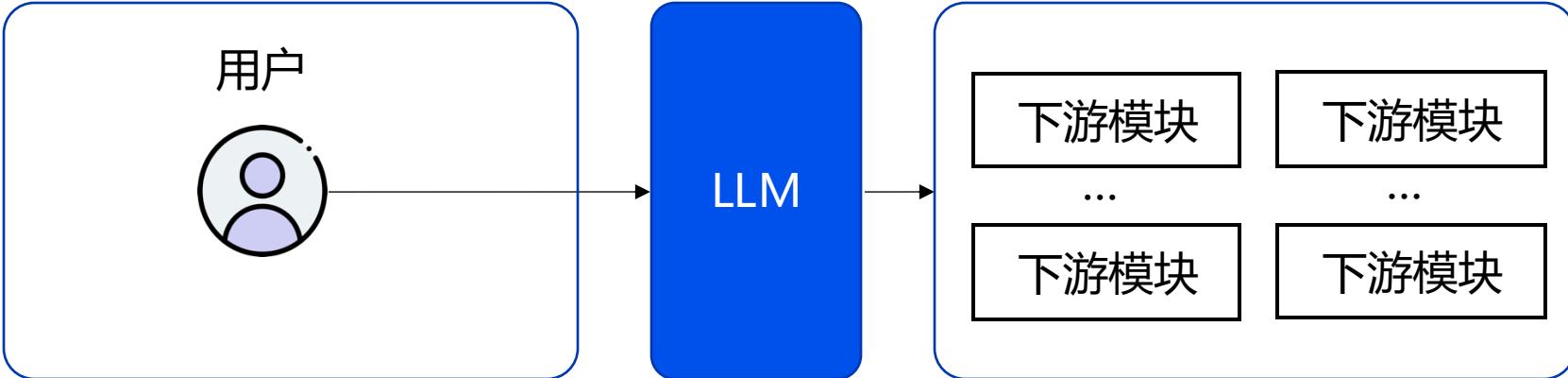


[1] Zhang, Duzhen, et al. Mm-llms: Recent advances in multimodal large language models. arXiv preprint arXiv:2401.13601.

多模态大模型的代表性架构

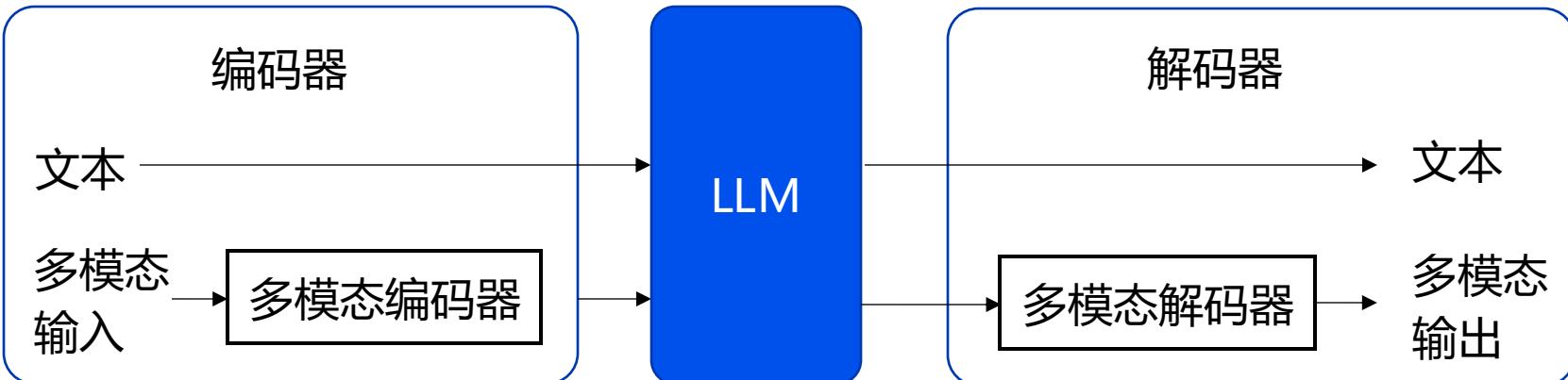
LLM为中心的架构

- + Visual-ChatGPT
- + HuggingGPT
- + MM-REACT
- + LLaVA-Plus
- + ...



一体化多模态架构

大多数多模态大模型的架构类型



多模态大模型的前沿教程 (CVPR & MM & COLING)



Tutorial

From Multimodal LLM to Human-level AI: Modality, Instruction, Reasoning and Beyond

Hao Fei · Yuan Yao · Ao Zhang · Haotian Liu · Fuxiao Liu · Zhuosheng Zhang · Shuicheng Yan
Summit 446

[Abstract] [Project Page]
Tue 18 Jun 1:30 p.m. PDT – 6 p.m. PDT

Abstract:
Artificial intelligence (AI) encompasses knowledge acquisition and real-world grounding across various modalities. As a multidisciplinary research field, multimodal large language models (MLLMs) have recently garnered growing interest in both academia and industry, showing an unprecedented trend to achieve human-level AI via MLLMs. These large models offer an effective vehicle for understanding, reasoning, and planning by integrating and modeling diverse information modalities, including language, visual, auditory, and sensory data. This tutorial aims to deliver a comprehensive review of cutting-edge research in MLLMs, focusing on three key areas: MLLM architecture design, instructional learning, and multimodal reasoning of MLLMs. We will explore technical advancements, synthesize key challenges, and discuss potential avenues for future research. All the resources and materials will be made available online: <https://mllm2024.github.io/CVPR2024>

SCHEDULE ————— PROGRAM

Our tutorial will be held on **Tuesday, June 18, 2024** (all the times are based on **UTC-7 = Seattle local time**).

| Time | Section | Presenter |
|-------------|---|-------------------------------------|
| 13:30-13:35 | Part 1: Background and Introduction [Slides] | Hao Fei |
| 13:35-14:05 | Part 2: MLLM Architecture [Slides] | Yuan Yao |
| 14:05-14:35 | Part 3: MLLM Modality&Functionality [Slides] | Hao Fei |
| 14:35-15:05 | Part 4: MLLM Instruction Tuning [Slides] | Haotian Liu |
| | Coffee Break, Q&A Session | |
| 16:00-16:30 | Part 5: MLLM Hallucination [Slides] | Fuxiao Liu |
| 16:30-17:00 | Part 6: MM Reasoning [Slides] | Zhuosheng Zhang |
| 17:00-17:30 | Part 7: MLLM Efficiency [Slides] | Ao Zhang |
| 17:30-18:00 | Part 8: Panel Discussion - From MM Generalist to Human-level AI | All + Hanwang Zhang + Shuicheng Yan |



讲义 + 视频

<https://mllm2024.github.io/CVPR2024>

TUTORIAL RECORD ————— VIDEO



From Multimodal LLM to Human-level AI

Modality, Instruction, Reasoning, Efficiency and Beyond

<https://mllm2024.github.io/CVPR2024/>



1



71

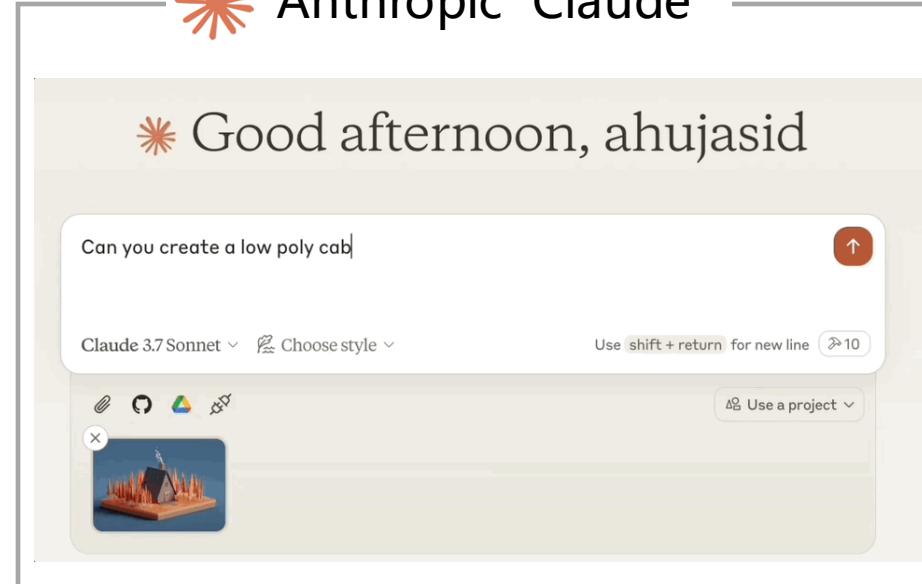
大模型智能体

无处不在的“贾维斯”，大模型智能体展现出广泛的应用前景

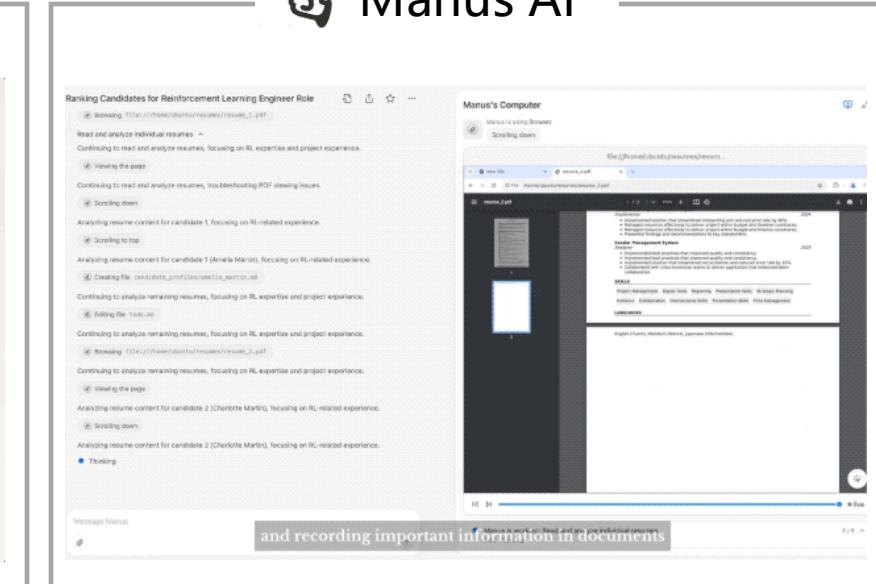
OpenAI Operator



Anthropic Claude



Manus AI



<https://openai.com/index/introducing-operator>

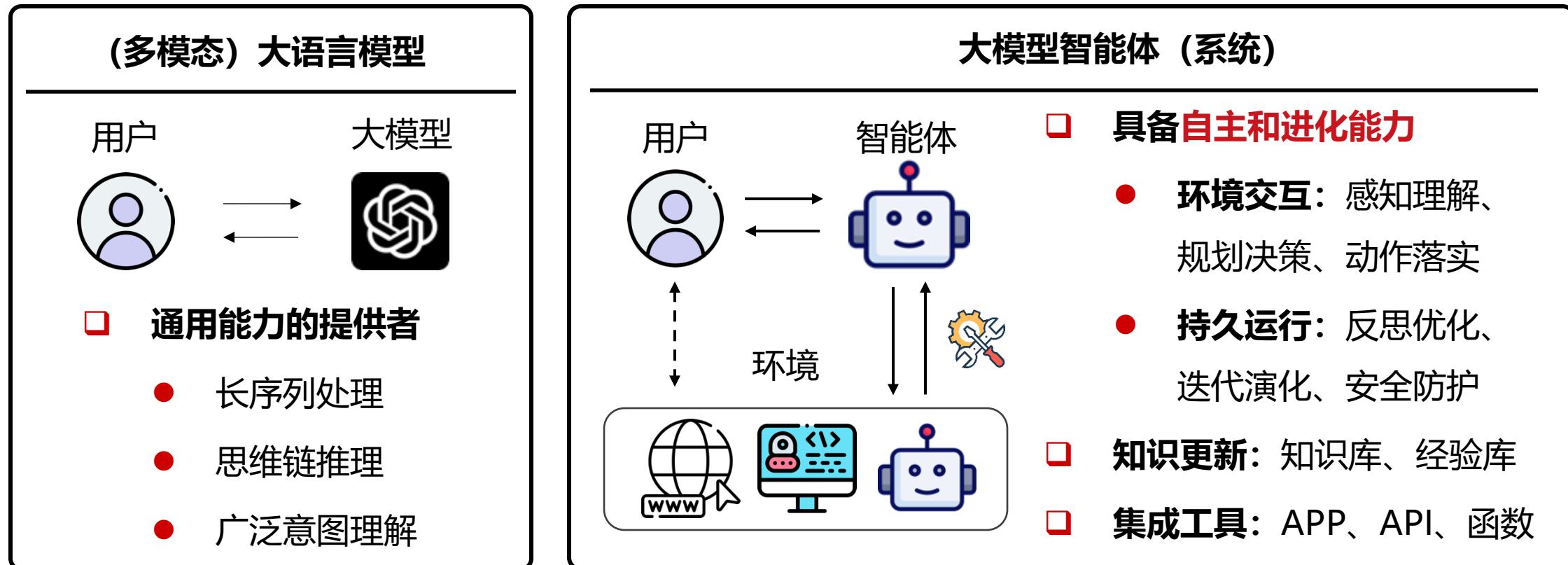
<https://www.anthropic.com/clause>

<https://manus.im/>



大模型智能体

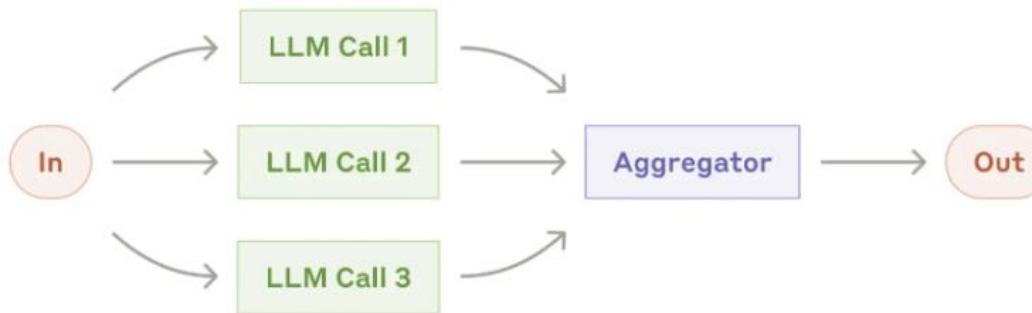
- 基于大模型构建，能够动态指导其自身流程和工具使用，并根据实时反馈调整自己的操作的**系统**¹



¹Anthropic. Building effective agents. Published Dec 19, 2024

大模型智能体

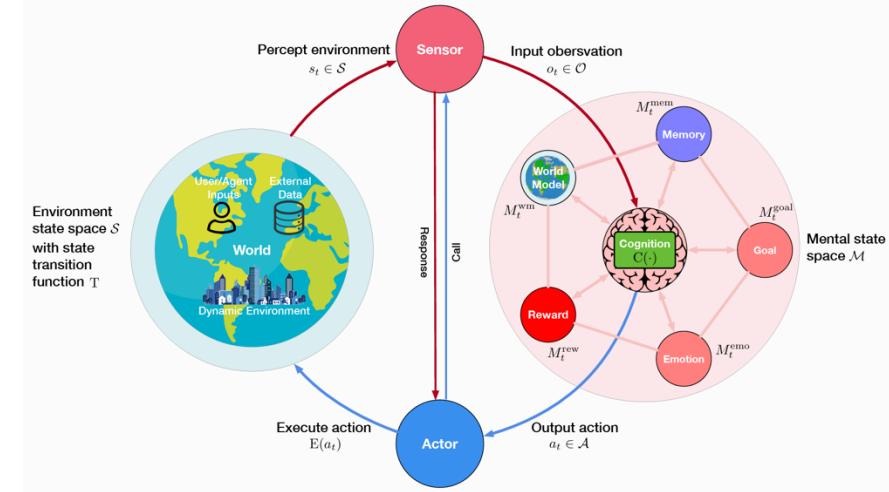
工作流编排



通过预定义的路径编排LLM和工具，类似复杂的自动化脚本。常以Prompt方式实现

将AI视为可编排的组件

智能体基座



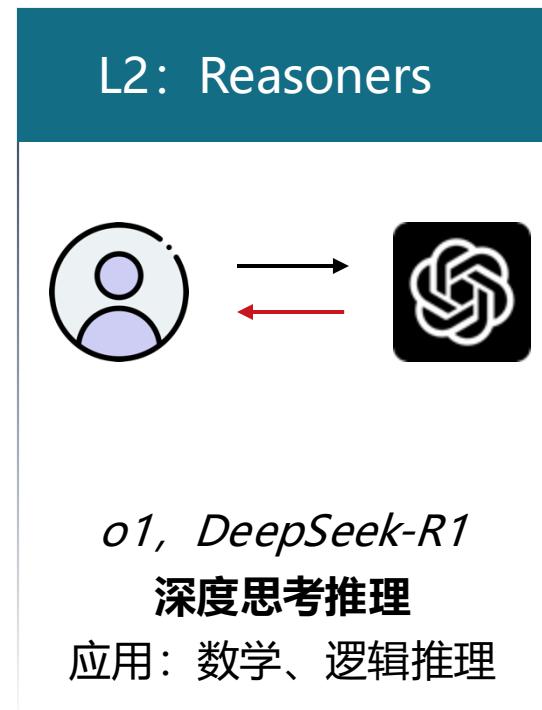
自主指导行为过程和工具使用，动态规划和更新路径，保持对任务完成方式的控制。常以训练方式拓展能力

给予AI更大的自主性、可进化



突破大模型的能力边界

□ 大模型智能体的能力突破，成为新焦点，极大拓展大模型能力边界和应用场景



语言->行动：从内容智能到**行为智能**，建立迈向通用人工智能的关键纽带



延伸阅读素材



《大模型开发全流程》

国产化大模型课程：54节课、26份实践手册

Ascend 开发者 主页 开发 文档 活动 学习 论坛 博客 开发者计划 搜索... 下载 支持

大模型开发全流程-初级

大模型开发全流程-初级

| 大模型 大模型简介 | 大模型 昇腾硬件简介 | 大模型 开发环境搭建 | 大模型 训练数据准备 |
|---|---|--|--|
| 大模型简介 人工智能技术经历了小数据到大数据，小模型到大模型，专用模型到通用模型的发 ★ 4.9 ① 1.0小时 1815 | 昇腾硬件简介 本章将对昇腾计算产业进行简介。昇腾计算产业是基于昇腾系列处理器和基础软件 ★ 4.8 ① 15分钟 580 | 开发环境搭建 本章将以Ubuntu系统为例，介绍如何在昇腾服务器上安装NPU驱动和NPU固件，以 ★ 5.0 ① 0.8小时 638 | 训练数据准备 大模型的性能与训练使用的数据的数量与质量有很大联系。本章将介绍预训练与指 ★ 5.0 ① 25分钟 338 |
| 大模型 大模型预训练-初级 | 大模型 大模型推理-初级 | 大模型 大模型性能评估-初级 | 大模型 提示学习 |
| 大模型预训练-初级 预训练是大模型构建的基础环节，以语言建模为基本任务，大量的训练数据是关 ★ 5.0 ① 0.6小时 337 | 大模型推理-初级 高效地部署与调用大模型可以充分发挥其作用。本章初级课程将介绍大模型在常见 ★ 5.0 ① 1.0小时 308 | 大模型性能评估-初级 模型的规模庞大并不总是意味着性能更好，因此我们需要一套有效的方法来评估 ★ 5.0 ① 25分钟 209 | 提示学习 提示学习主要通过自然语言接口使大模型任务表达更为直观，包括少样本提示、零 ★ 5.0 ① 1.0小时 200 |

<https://www.hiascend.com/edu/growth/lm-development>



谢谢！

zhangzs@sjtu.edu.cn
<https://bcmi.sjtu.edu.cn/~zhangzs>

