目录

大模型通识知识 3
前　言 3
1.人工智能的历史 3
1.1 人工智能 1.0 时代 4
1.2 人工智能 2.0 时代 4
第一章　大模型和人工智能 5
1.1 机器学习过程（AI 训练过程） 7
1.2 机器学习的分类 8
1.3 深度学习 10
第二章 大模型概述 11
2.1 大模型定义 11
2.2 大模型带来的技术变革 12
2.3 大模型的发展历程 14
2.4　大模型的分类与代表模型 15
2.5 自然语言处理（NLP） 16
第三章 大模型核心技术 17
3.1 大模型的架构基础（transformer） 17
3.2 训练与优化技术（预训练、监督微调、奖励建模、基于人类反馈的强化学习） 25
3.3　大语言模型的核心目标 30
1.自回归语言建模（如 n-gram、RNN、GPT） 31
2. 掩码语言建模（如 BERT） 32
3.4　价值观对齐：从技术到伦理 33
第四章 LLM 的应用与实践 36
4.1 LLM 的核心能力 36
4.2 Agent 智能体与大模型 39
第五章 大模型局限性 41
5.1 知识的局限性（数据依赖） 41
5.2 幻觉问题（Hallucination） 41
5.3 数据安全问题 41
5.4 数据偏见及伦理局限性（生成不当内容） 42
附录 43
专有名词 43
1.模态（Modality） 43
Token 43
LangChain 43
RAG 44
大模型预训练 44
涌现能力 44
调用 API 44
生成式模型 44

判别式模型 45
提示词工程 prompt 45

大模型通识知识

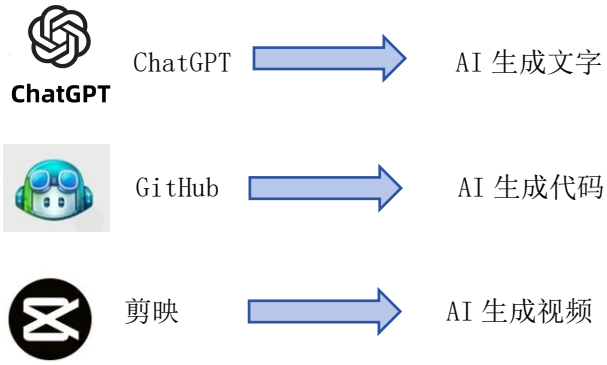
前　言

1.人工智能的历史

人工智能（Artificial Intelligence）作为当今科技领域炙手可热的研究领域之一，近年来得到了越来越多的关注。人工智能（Artificial Intelligence）从标准的定义来讲，是利用数字计算机或者数字计算机控制的机器模拟、延伸和扩展人的智能，感知环境、获取知识并使用知识获得最佳结果的理论、方法、技术及应用系统。**简单来说，就是让机器能够模拟人类的思维能力，执行与人类智能有关的行为，如判断、推理、证明、识别、感知、理解、通信、设计、思考、规划、学习和问题求解等思维活动。**但与其工具属性、能力属性相比，人工智能更重要的是一种思维和工具，是用来描述模仿人类与其他人类思维相关联的 “认知”功能的机器，如“学习” 和“解决问题”。

而其中最引人注目的是生成式人工智能 (Generative Artificial Intelligence)，这是一种基于机器学习技术的人工智能算法，其目的是通过学习大量数据和模式，生成新的、原创的内容。这些内容可以是文本、图像、音频或视频等多种形式。生成式人工智能通常采用深度学习模型，如循环神经网络 (Recurrent Neural

Network，RNN)、变分自编码器（Variational Auto Encoder，VAE)等，来生成高质量的内容。生成式人工智能的应用包括文本生成、图像生成、语音合成、自动创作和虚拟现实，具有广泛的应用前景



1.1 人工智能 1.0 时代

人工智能概念于 1956 年被提出，AI 产业的第一轮爆发源自 2012 年。 2012 年，AlexNet 模型问世，开启了卷积神经网络(Convolutional Neural Network，CNN） 在图像识别领域的应用。2015 年机器识别图像的准确率首次超过人（错误率低于 4%），开启了计算机视觉技术在各行各业的应用，带动了人工智能 1.0 时代的创新周期，AI+开始赋能各行各业，带动效率提升。但是，人工智能 1.0%代面临着模型碎片化、AJ 泛化能力不足等问题。

1.2 人工智能 2.0 时代

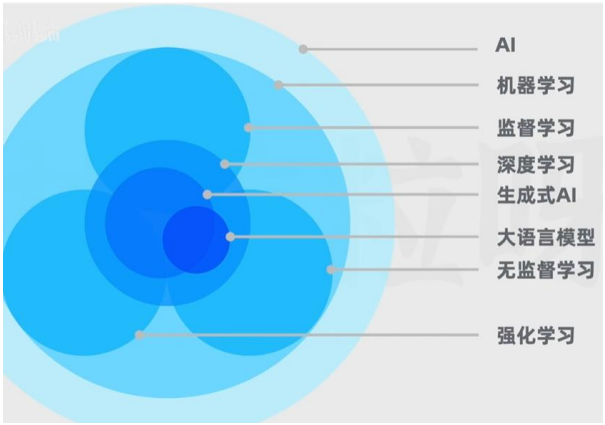
2017 年，Google Brain 团队提出 Transformer 架构，奖定了大模型领域的主流算法基础。从 2018 年开始大模型迅速流行。2018 年，

谷歌因队的模型参数首次过亿，到2022年模型参数达到5400亿，模型参数呈现指数级增长，“预训练＋微调” 的大模型有效解决了1.0时代AI泛化能力不足的问题。新一代AI技术有望开始全新一轮的技术创新周期。

当前，人工智能的应用场景已经覆盖了生活的各个领城。在医疗领域，人工智能可以帮助医生进行诊断和治疗决策，提高医疗效率和精度。在金融领域，人工智能可以进行风险管理和数据分析，提高金融服务的质量和效率。在交通领域，人工智能可以进行交通管控和路况预测，提高交通安全和效率。在智能家居领域，人工智能可以进行智能家居控制和环境监测，提高家庭生活的舒适度和安全性。此外，人工智能还可以应用于教育、娱乐、军事等多个领城，为人类社会的发展带来了无限的可能性。

第一章　大模型和人工智能

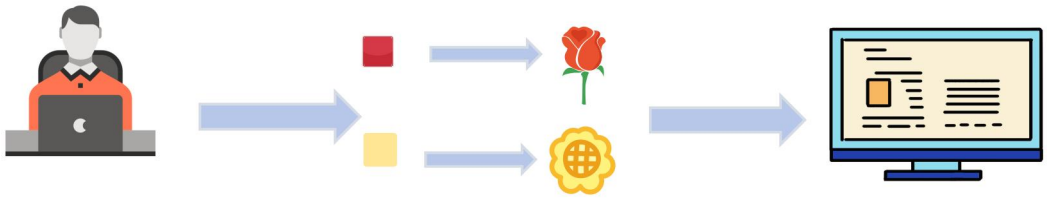
机器学习是一种实现人工智能的方法；深度学习是一种实现机器学习的技术；大语言模型是深度学习的特定应用。



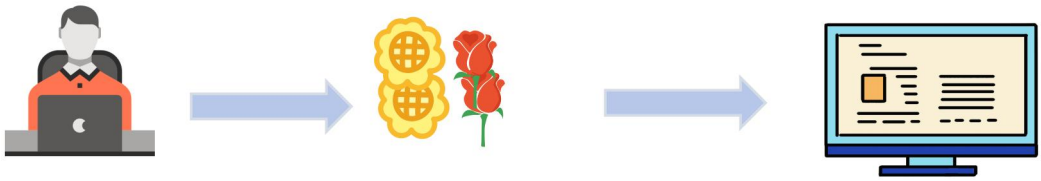
人工智能与大模型梯度关系：人工智能>机器学习>深度学习>大语言模型

AI 的一个子集。它的核心在于不需要人类做显示变成，而是让计算机系统能够通过对数据的学习来提高性能。我们不是直接编程告诉计算机如何完成任务，而是提供大量的数据，让机器通过数据找出隐藏的模式或规律，然后用这些规律来预测新的、未知的数据。

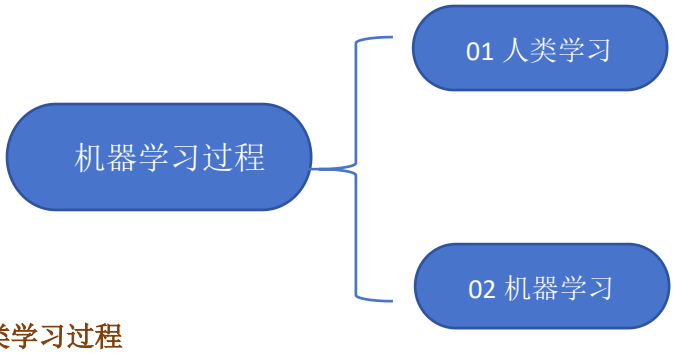
显示编程：人类编写程序告知计算机何为玫瑰、何为向日葵



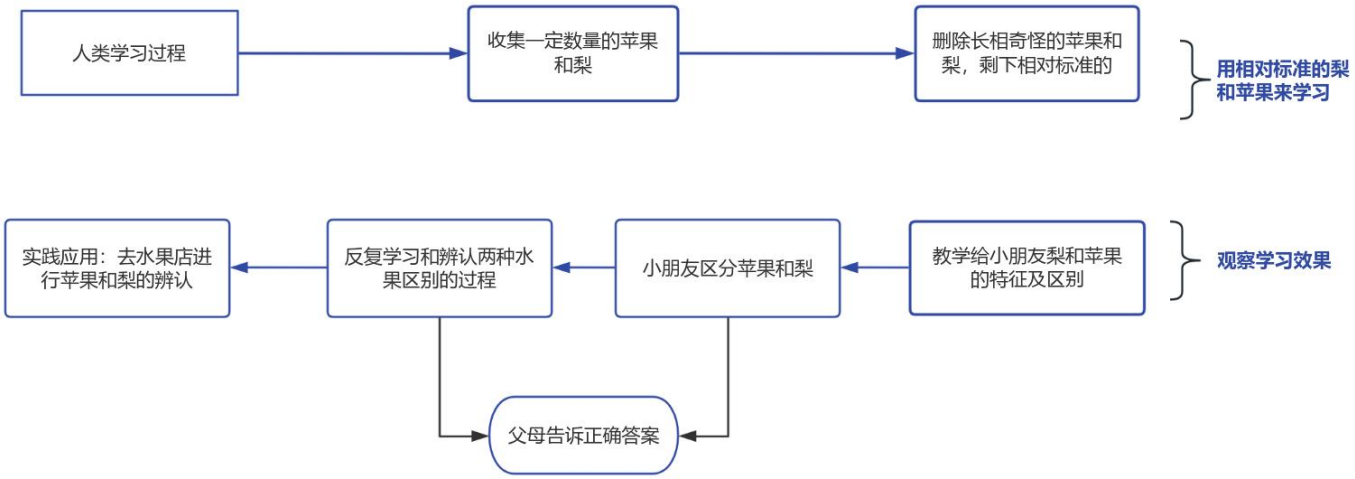
机器学习：通过给机器大量玫瑰和向日葵数据，让机器自己识别和总结规律，从而对未见过的图片进行预测和判断



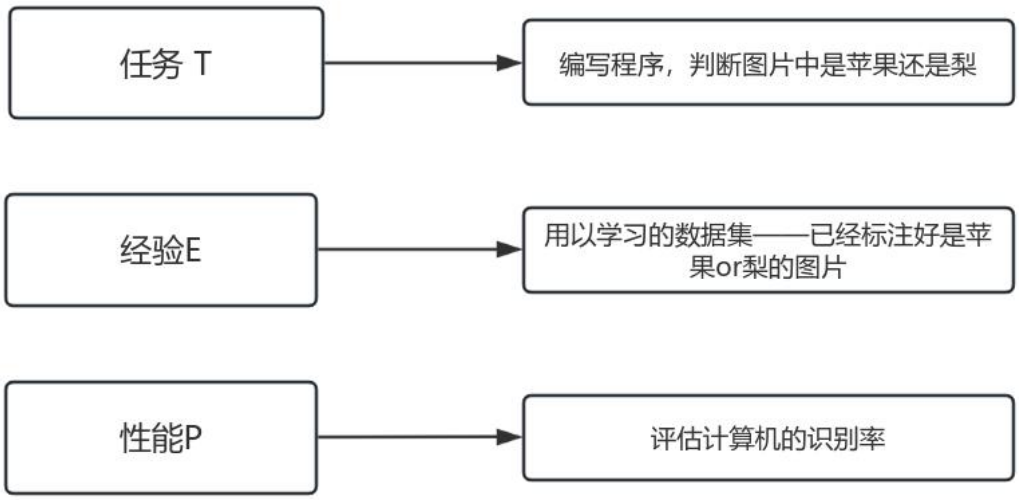
1.1 机器学习过程（AI 训练过程）



人类学习过程

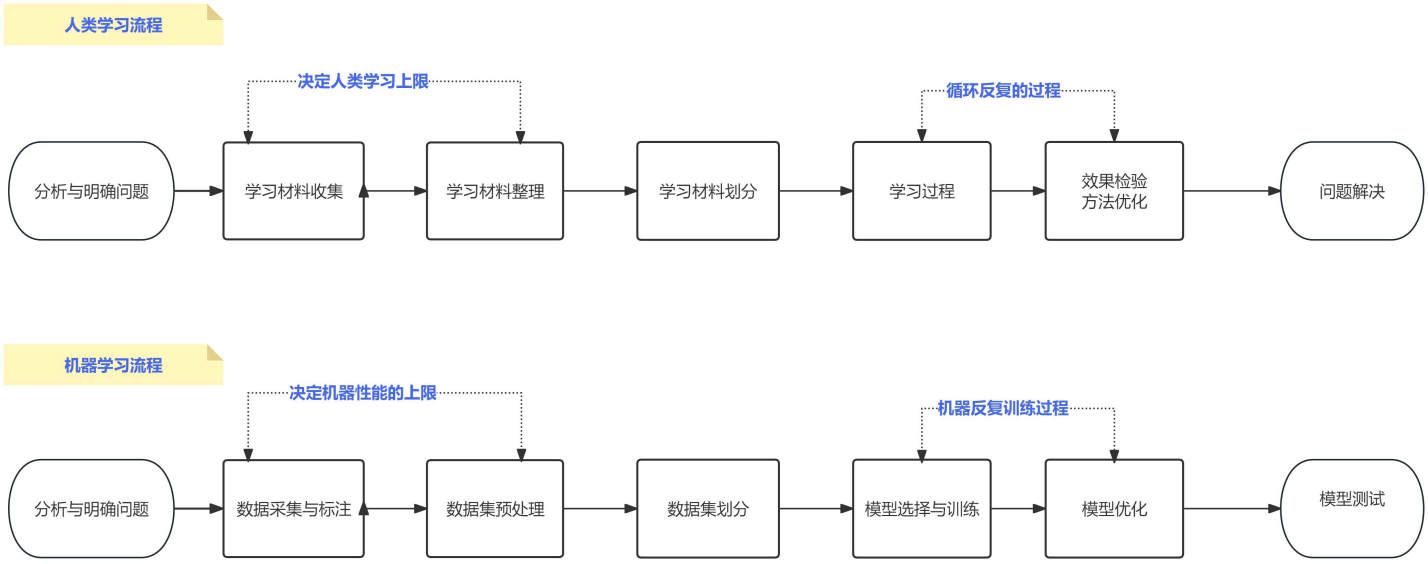


机器学习过程



对于某项任务 T、某项性能评价指标 P，如果一个计算机程序在任务 T 上的性能指标 P 能够随着经验 E 而自我完善，则这个计算机程序从经验 E 中进行了学习。

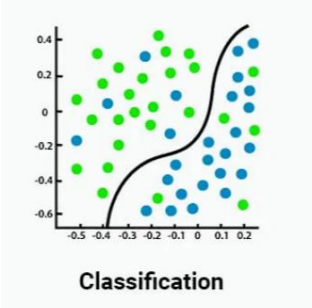
人类学习流程与机器学习流程对比总结：



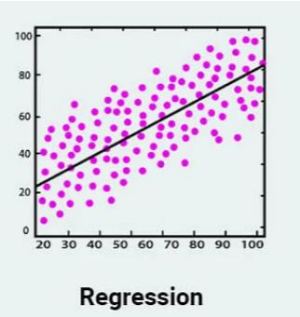
1.2 机器学习的分类

①监督学习（有标签的训练数据）:机器学习算法在监督学习里会接受有标签的训练数据。标签及为期望的输出值。每个训练数据点既包括输入特征，也包括期望的输出值。算法的目标是学习输入和输出之间的映射关系，从而在给定新的输入目标后能预测出相应的输出值。

经典的监督学习任务包括：分类和回归。



分类：将数据划分为不同的类别



回归：对数值进行预测

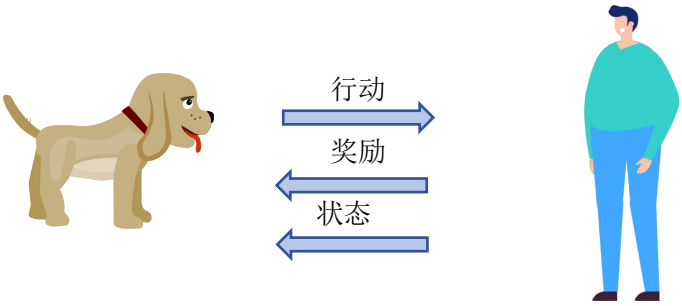
对比维度	分类（Classification）	回归（Regression）
预测目标	离散的类别标签（如“是/否”、“猫/狗/鸟”）	连续的数值（如房价、温度、销售额）
输出类型	预测样本属于什么类别；比如是猫/狗，健康/患病	数值（实数）；房价=300w
应用场景	判断、识别、分组	预测趋势、估算数值

②无监督学习（无标签的训练数据）:对于没有标记的样本，学习算法直接对输入数据集进行建模。经典的无监督学习任务包括**聚类**，即“物以类聚，人以群分”。我们只需要把相似度高的东西放在一起，对于新来的样本，计算相似度后，按照相似程度进行归类。比如用一堆新闻文章，让模型根据主题或者文章的特征自动把相似的文章进行组织。

③半监督学习:试图让学习器自动地对大量未标记数据进行利用以辅助少量有标记数据进行学习

④强化学习:让模型在环境里采取行动，获得结果反馈，从反馈中学习，从而能在给定的情况下采取最佳行动，来最大化奖励或最小化损

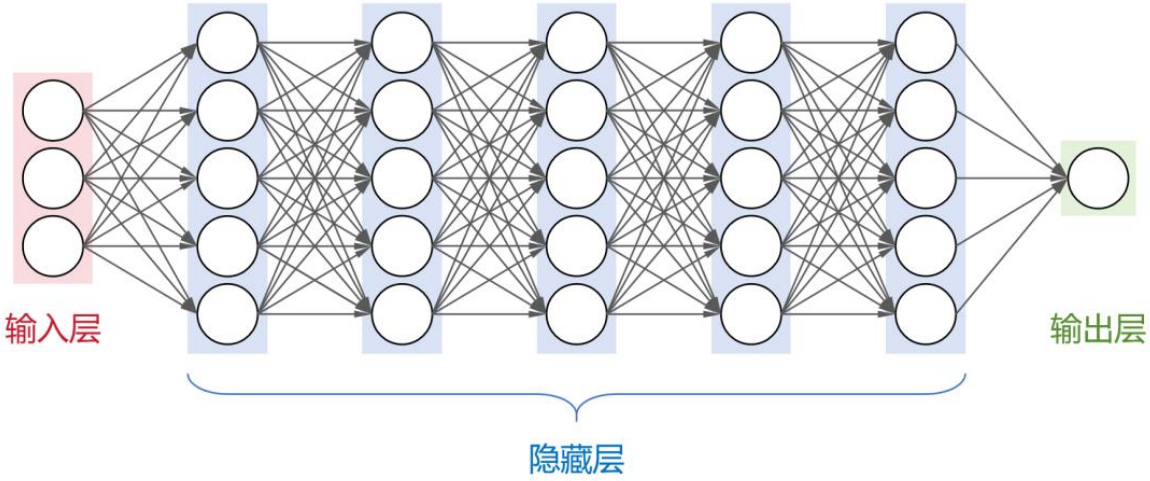
失。例如，强化学习可以让模型应用在下围棋等策略性游戏，通过不同行动导致的奖惩反馈，从而优化自己的游戏策略。



例如小狗在一开始会做出很多随心所欲的动作，但随着和训犬师的互动，小狗会发现某些动作会获得零食，某些动作不会获得零食甚至获得惩罚，通过观察动作和奖惩之间的关系，小狗的行为会逐渐接近训犬师的期望。

1.3 深度学习

深度学习是机器学习的一种方法。指模拟人脑的工作方式，**核心在于创建人工神经网络来处理数据。这些神经网络包含多个处理层，**因此被称为“深度”学习。深度学习模型能够学习和表示大量复杂的模式这使它们在诸如图像识别、语音识别和自然语言处理等任务中非常有效。神经网络可以用于深度学习、监督学习、无监督学习、强化学习，所以神经网络并不属于机器学习的子集，而是一种方法。

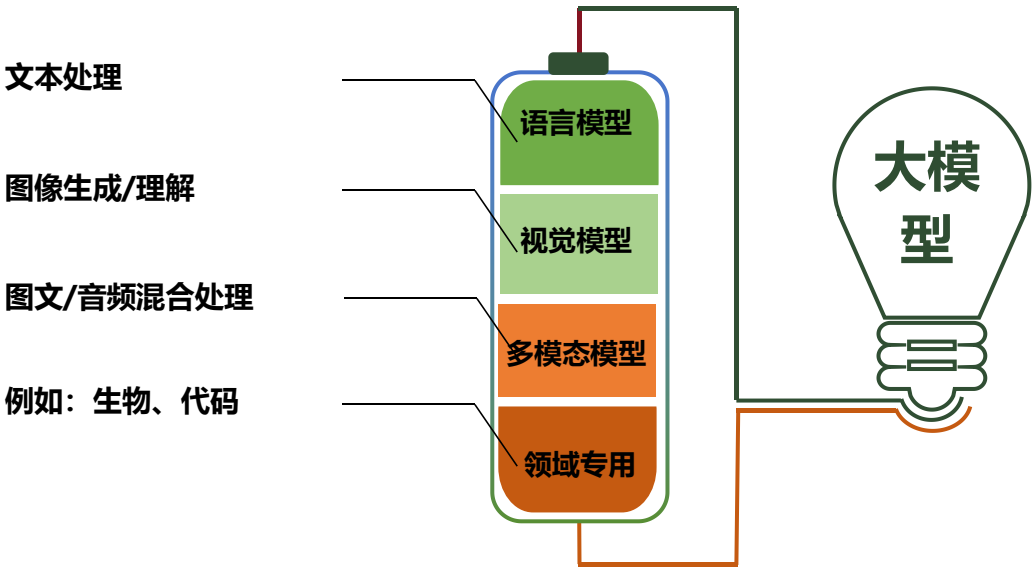


第二章 大模型概述

2.1 大模型定义

大模型是指具有非常多参数数量的**人工神经网络模型**。**在深度学习领域，大模型有数亿到数万亿参数的模型**。这些模型通常需要在大规模数据集上进行训练，并且需重计算资源进行优化和调整。大模型通常用于解决复杂的自然语言处理、计算机视觉和语音识别等任务。这些任务通常要处理大量的输入数据，并从中提取复杂的特征和模式。通过使用大模型，深度学习算法可以更好地处理这些任务，提高模型的准确性和性能。

大模型的训练和调整需要大量的计算资源，包括高性能计算机、图形处理器（Processing Unit，GPU）和云计算资源等。为了训练和优化大模型，研究人员和企业通常需要投入巨大的资源和资金。



2.2 大模型带来的技术变革

人工智能正处于从”能用” 到”好用”的应用落地阶段：但地处于落地初期，主要面临场景需求碎片化、人力研发和应用计算成本高，以及长尾数据少导致模型训练精度不够、模型算法从实验室到真实场景差距较大等行业问题。而大模型在增加模型通用性、降低训练研发成本等方面降低了人工智能落地应用的门槛。

近 10 年来，通过“深度学习+大算力”获得训练模型，已经成为实现人工智能的主流技术途径。由于深度学习、数据和算力这 3 个要素都已具备，因此全球抵起了“大炼模型” 的热潮，也催生了一大批人工智能公司。然而，**在深度学习技术出现的近 10 年里，模型基本上都是针对特定的应用场景进行训练的，即小模型属于传统的定制化、作坊式的模型开发方式**。传统人工智能模型需要完成从研发到应用的全方位流程，包括需求定义、数据收集、模型算法设计、训练调优、应用部署和运营维护等阶段组成的整套流程。这意味着除了需要

优秀的产品经理准确定义需求外，还需要人工智能研发人员扎实的专业知识和协同合作能力，才能完成大量复杂的工作。

在传统模型中，研发阶段为了满足各种场景的需求，人工智能研发人员需要设计个性定制化的、专用的神经网络模型。模型设计过程需要研究人员对网络结构和场景任务有足够的专业知识，并承担设计网络结构的试错成本和时间成本。一种降低专业人员设计门槛的思路是通过网络结构自动搜索技术路线，但这种方案需要很高的算力，不同的场景需要大量机器自动搜索最优模型，时间成本仍然很高。一个项目往往需要专家团队在现场待上几个月才能完成。通常，为了满足目标要求，数据收集和模型训练评估需要多次迭代，从而导致高昂的人力成本。**但是，这种通过 “一模一景” 的车间模式开发出来的模型，并不适用于垂直行业场景的很多任务**。例如，在无人驾驶汽车的全景感知领域，往往需要多行人跟踪、场景语义分割、视野目标检测等多个模型协同工作；与目标检测和分割相同的应用，在医学影像领域训练的皮肤瘤检测和人工智能模型分割，不能直接应用于监控景点中的行人车辆检测和场景分割。模型无法重复使用和积累，这也导致了人工智能落地的高门槛、高成本和低效率。

大模型是从庞大、多类型的场景数据中学习，总结出不同场景、不同业务的通用能力，学习出一种特征和规律，成为具有泛化能力的模型库。在基于大模型开发应用或应对新的业务场景时，可以对大模型进行适配，比如对某些下游任务进行小规模标注数据二次训练，或者无须自定义任务即可完成多个应用场景，实现通用智能能力。因此，

利用大模型的通用能力，可以有效应对多样化、碎片化的人工智能应用需求，为实现大规模人工智能落地应用提供可能。

大模型正在作为一种新型的算法和工具，成为整个人工智能技术新的制高点和新型的基础设施。可以说大模型是一种变革性的技术，它可以显著地提升人工智能模型在应用中的性能表现，将人工智能的算法开发过程由传统的烟囱式开发模式转向集中式建模，解决人工智能应用落地过程中的场景碎片化、模型结构和模型训练需求零散化的痛点。

2.3 大模型的发展历程

从模型演进来看，AI 大模型的发展先后经历了语言模型、神经网络语言模型、预训练模型、大规模预训练模型、超大规模预训练模型几个阶段

传统语言模型（如 n-gram）基于词频统计计算序列概率，虽简单易用，却受限于数据稀疏和短程依赖问题。随后，神经网络语言模型（如 LSTM）通过词嵌入和循环结构赋予模型分布式表示能力，但训练依赖标注数据且难以捕捉长距离关联。2010 年代，预训练模型（如 Word2Vec、ELMo）兴起，利用无监督学习从海量文本中提取通

用特征，显著降低对标注数据的依赖，但模型规模较小、上下文建模能力有限。

2017 年 Transformer 架构的提出，其自注意力机制突破了序列建模的瓶颈，支持并行计算和全局依赖捕捉。以此为基石，大规模预训练模型（如 BERT、GPT-2）迅速崛起，参数规模首次突破亿级（如 BERT-base 的 1.1 亿参数），通过掩码预测（BERT）或自回归生成（GPT）从无标注数据中学习深层语义。2020 年后，算力爆发与数据洪流推动模型进入超大规模时代（如 GPT-3、PaLM、GPT-4），参数跃升至千亿甚至万亿级，涌现出推理、创作、多模态理解等接近人类的能力。例如，GPT-3 仅凭提示即可生成代码、诗歌或科学解答，而 GPT-4 进一步融合图像与文本，展现通用人工智能的雏形。

这一历程的推动力来自三方面：算法革新（如 Transformer 替代 RNN）、硬件跃迁（GPU/TPU 集群支持分布式训练）与数据积累（互联网文本、代码等多源数据）。每一阶段均试图解决前代的局限——从统计模型的刚性到神经网络的泛化性，从小规模预训练的通用性到超大规模模型的涌现能力。未来，大模型将向高效化（降低训练成本）、安全化（规避伦理风险）与多模态化（融合视听觉）持续进化，探索更接近人类智能的边界。

2.4 大模型的分类与代表模型

大语言模型

语言模型是预测词序列概率或生成连贯文本的模型，核心任务是理解语言的统计规律，专注于理解和生成人类语言的模型，能完成写作、对话、翻译等任务。大语言模型**特指参数规模极大**（通常超亿级）、**基于深度学习架构（如 Transformer）训练的语言模型，具备复杂语义理解和生成能力。**大语言模型（**例如** GPT-4、PaLM、LLaMA、文心一言）的核心能力覆盖了 NLP 的两大方向：①语言理解；分析文本含义（如情感判断、信息提取）②语言生成；创作连贯文本。

多模态大模型（如 GPT-4V、Flamingo）

多模态大模型就是一种能够理解和处理多种类型的机器学习模型——而类型也被叫做模态，包括文本，图片，音频，视频等。与只能处理单一类型数据的单模态模型不同，多模态模型就像是一个全能的 “超级大脑”，可以同时接收并理解来自不同模态的信息

视觉大模型（文生图：Midjourney，Stable Diffusion）

2.5 自然语言处理（NLP）

自然语言为人类在社会生活中使用的语言，如中文英文等。而 NLP 是让计算机理解、生成和处理人类语言的技术，目标是实现人机自然交互。NLP 的基础任务分为理解和生成两大类。理解类包括文本分类（如情感分析）、信息抽取（如提取人名、地点）等，生成类包含机器翻译、自动摘要、对话系统。

NLP 是计算机科学、人工智能、语言学、人类自然语言相互作用的领域。手机中的语音助手、智能手表、智能音响等都是 NLP 技术的应用。但计算机如果需要了解自然语言，就必须将接收到的文字转化为某种数学表达形式。 计算机通过特定的数学语言实现人机对话的功能。

语言模型（LLM）是自然语言处理（NLP）领域的最新技术成果。NLP 旨在让计算机像人类一样理解和生成语言，而大语言模型通过海量数据学习，将 NLP 的能力提升至前所未有的高度。

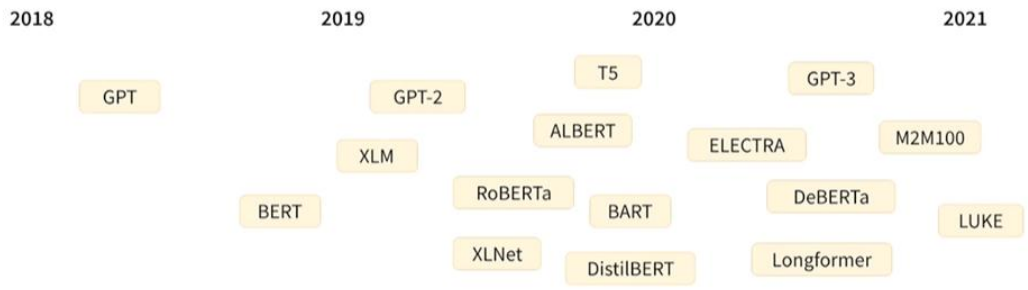
第三章 大模型核心技术

3.1 大模型的架构基础（transformer）

语言模型的发展经历了统计语言模型（1950-1990）、神经网络模型（2000-2010）、预训练语言模型（2010-2018）及至今的 Transformer 架构。

2017 年 Google 研究人员在论文《Attention is All You Need》中首次提出了 Transformer 网络 Transformer 网络是一种用于处理序

列数据的深度学习模型，其核心创新点是自注意力机制(Self-Attention Mechanism)。Transformer 出现后自然语言处理（NLP）的方向被改变了。2018 年 Open AI 发布 GPT1.0，谷歌发布 BERT，2019 年 Open AI 发布 GPT2.0、百度发布 ERNIE1.0，大模型的发展进入突破式前进阶段。



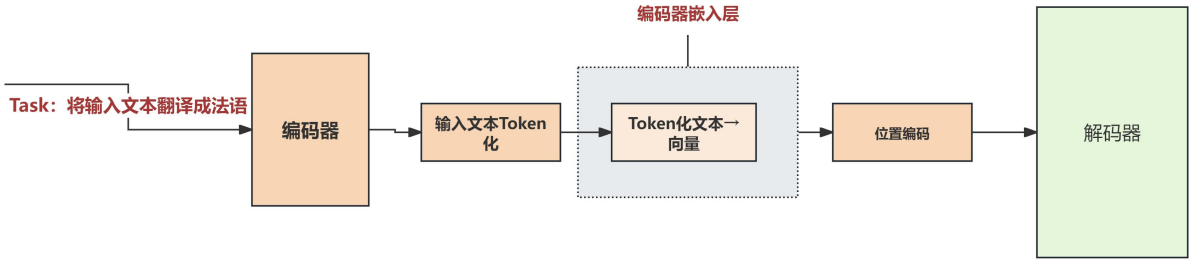
3.1.1 Transformer 主要组件：编码器和解码器

2017 年设计的原始 Transformer 构架是一个序列到序列模型，有两个主要组件：编码器和解码器。每个编码器和解码器都由多个自注意力层和前馈神经网络层组成。

编码器：接收原始文本，转换为向量；

解码器：将向量转为输出语言的句子

编码器与解码器的任务流程：原始文本 → Token 化 → [嵌入层 → 向量] → 位置编码 → 编码器层

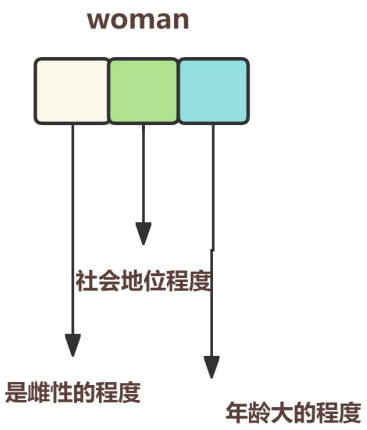


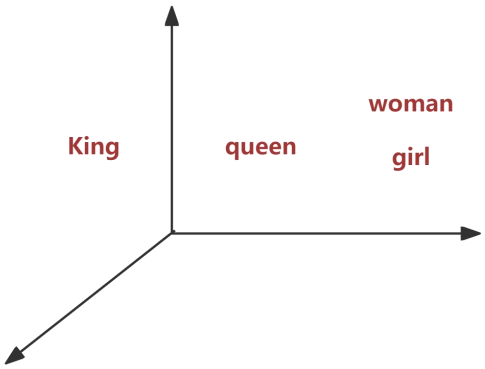
Token 化：

假设我们需要使用 Transformer 做英语翻译法语的任务。编码器输入英文[She is in a restaurant]，解码器返回对应的法语。在该过程中，编码器首先需要将文本[She is in a restaurant]Token 化（词元），也就是将拆成各个 Token。**Token 可以理解为文本的基本单位，是词义意义最小单元，它是 LLM 的基本输入。**因为计算机内部无法储存文字，所以每个 Token 会被用一个整数数字代替，被称为 Token ID。

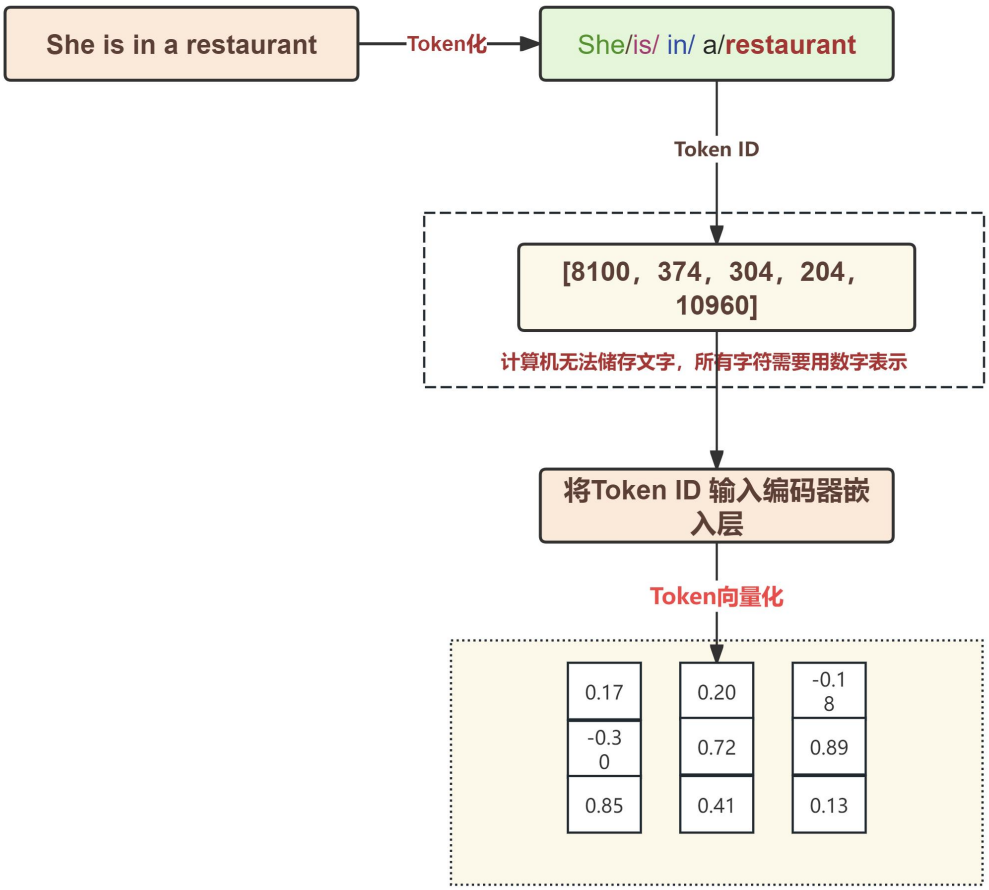
Token ID 向量化：

然后将 Token ID 传入嵌入层，嵌入层将每个 Token ID 都用向量来表示。向量可以被简单地看做**一串数字，这是因为一串数字能表达的含义大于一个数字。可以简单地理解一个数字相当于一个维度，一串数字就是用多个维度描述该词语。**





嵌入层的向量包含了词汇之间的语法语义等关系，相似的词对应的嵌入向量，在向量空间里距离也更近，相反则更远。这有助于模型利用数学计算向量空间里的距离，**捕捉不同词在语义和语法上的相似性**。并且“男人”与“国王”的差异、“女人”和“女王”的差异可以被看作是相似的，这也可以在向量空间里展现。**因此词向量不仅可以帮助模型理解词的语义，也可以捕捉词与词之间的复杂关系。**



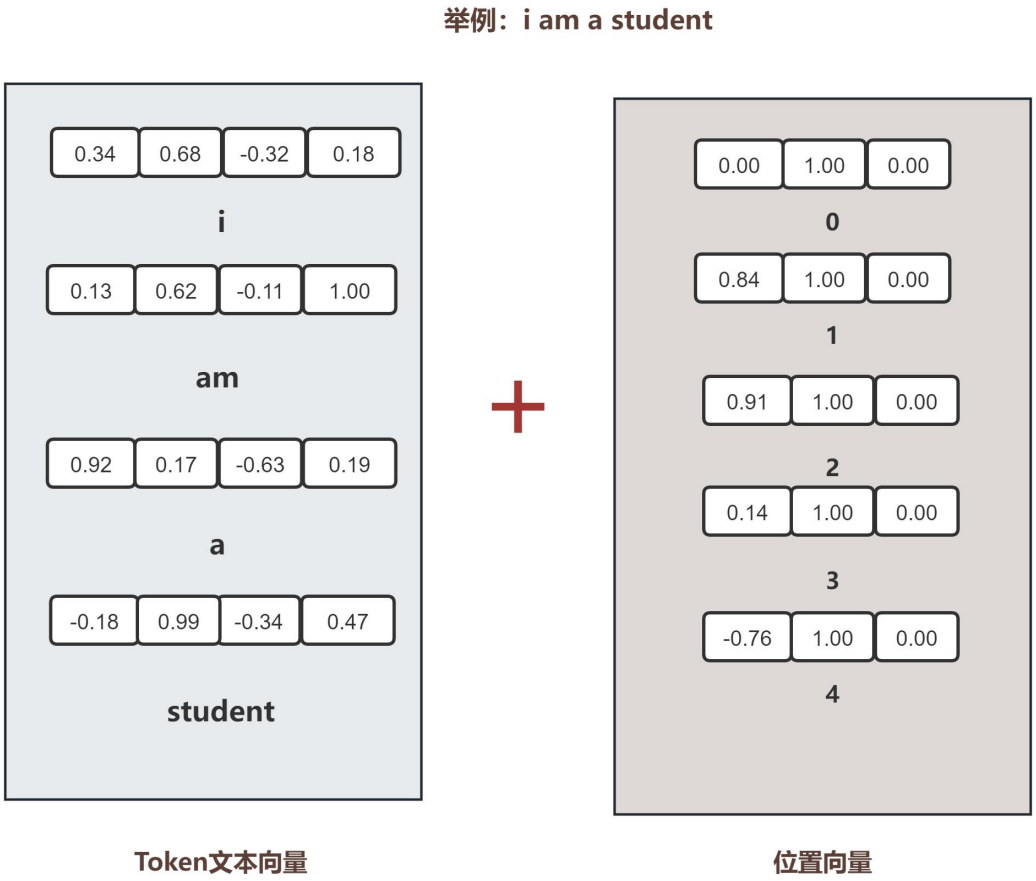
备注：为了展示方便，这里的向量长度取了“3”，但提出 Transformer 的论文中向量的长度是 512，GPT-3 为 12288

位置编码：

Transformer 的一项关键创新就是位置编码。在语言中，顺序会影响句子的意思理解。例如“不可随处小便”和“小处不可随便”，顺序不同意思也大相庭径，因此自然语言处理（NLP）领域会采用“序列”一词，表示按照特定顺序排列的元素。

RNN 与人类阅读习惯相同，对输入序列按照顺序处理，但因为无法并行处理，学习所有的信息，这造成了训练速度

的瓶颈 。Transformer 将词输入神经网络前，第一：将词转换为向量，即每个词各用一串数字表示（词向量）。第二：将每个词的位置用数字表示（位置向量）。把两次结果相加后输入编码器。因此模型因此既可以理解每个词的意思，又可以捕获词在句子中的位置，从而理解不同词之间的顺序关系。借助位置编码词可以不按顺序输入给 Transformer，模型可以同时处理输入序列里的所有位置，而不需要像 RNN 那样按照顺序依次处理。且每个输出可以独立计算，不需要等其他位置的计算结果，大大提高了训练速度，拓展到更大规模的训练数据集。自 2017 年 Transformer 架构问世以来，利用改架构或将其部署进系统的软件生态也呈现爆炸式增长。

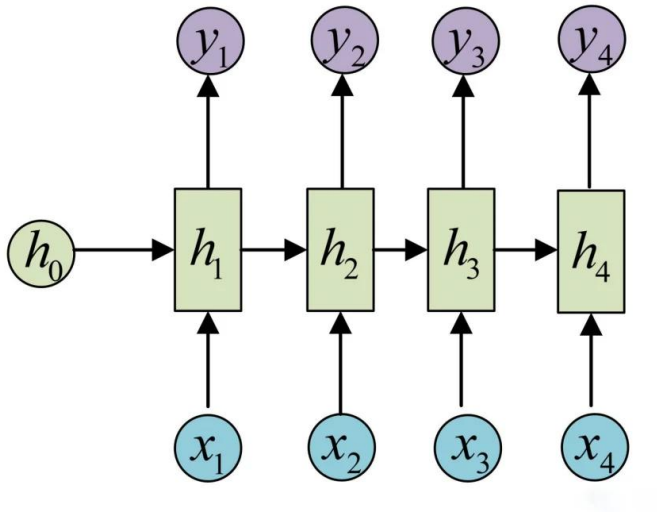


3.1.2 Transformer 关键创新---自注意力机制的作用

在 Transformer 架构被提出之前，语言模型的主流架构主要是循环神经网络（RNN）。RNN 按顺序逐字处理，每一步的输出取决于先前的隐藏状态和当前的输入，要等上一个步骤计算完后才能进行当前计算，无法进行并行计算，运算效率低。且 RNN 不擅长处理长序列（长文本），由于 RNN 的架构特点，词之间距离越远，前面对后边的影响越弱，因此 RNN 难以有效捕获到长距离的语义关系。但在人类自然语言中，依赖信息之间距离较远是比较常见的情况。

EG：“我在广东长大，这边有很多移民，还有美丽的大海。即使我的父母都是四川人，我还是更喜欢吃____。”

这段话中正确预测“___”内容的关键是距离其很远的“广东”。如果用 RNN 预测生成后续内容，很有可能到“即使我的父母都是四川人，我还是更喜欢吃____。”部分时，它已将前半部的信息遗忘。为了捕捉长距离依赖性，虽然出现了 RNN 的改良版本长短期记忆网络（LSTM）但依然无法解决传统 RNN 无法并行计算的问题。



循环神经网络（RNN）

Transformer 有能力学习输入序列词里所有词的相关性和上下文且不会受到短时记忆的影响，关键在于它的**自注意力机制。**

自注意力机制：Transformer 在处理每个词的时候不仅会注意这个词本身，以及它附近的词，还会去注意输入序列中所有其他的词。并且给予每一个词不一样的注意力权重。权重是模型在训练过程中通

过大量文本训练逐渐习得的，因此 Transformer 有能力知道当前这个词与其它词之间的关联强度，并且专注于输入里真正重要的部分。因此即使两个词隔得比较远，Transformer 仍然可以捕捉到它们之间的依赖关系。

Eg：The animal didn't cross the street because it was too tired

it 可能指“street”，也可能是“animal”
自注意力机制捕捉到了 it 和 animal 之间更强的关系，因此更集中在 animal 上

3.2 训练与优化技术（预训练、监督微调、奖励建模、基于人类反馈的强化学习）

大模型之所以需要训练过程，是因为其核心能力并非预先编程设定，而是通过从海量数据中自动学习规律和知识获得的。训练过程是模型从“空白状态”成长为“智能体”的关键阶段。

在训练像 DeepSeek、ChatGPT 这类大型语言模型时，通常包含四个关键阶段：STEP1-预训练（Pre-Training）、STEP2-监督微调（SFT）、STEP3-奖励建模（RM）和 STEP4 基于人类反馈强化学习的优化（RLHF）。

3.2.1 大模型预训练：pre-training

在海量无标注数据上训练一个通用的基础模型，使其具备对语言、图像或其他模态数据的通用理解能力，然后开始训练模型，不断调整

参数，直到损失越来越小。在训练的过程中，一开始初始化的参数会不断变化。直至达到较为满意的结果时可以将训练模型的参数保存下来，以便训练好的模型可以在下次执行类似任务时获得较好的结果。这个过程就是 pre-training 。也就是在让 AI 模型在特定任务前，先通过海量无标注数据，自主挖掘语言/视觉/逻辑的通用规律，构建基础认知能力的训练过程。通过从大规模未标记数据中学习通用特征和先验知识，减少对标记数据的依赖，加速并优化在有限数据集上的模型训练。

可以说预训练定义了模型的上限。并且技术突破也都集中在预训练阶段，预训练决定了技术发展的方向：只有通过预训练突破底层算法和架构，模型才能实现质的飞跃。像 Transformer、LLM 等核心技术，都是围绕预训练阶段的优化而提出的

预训练的模型在自然语言处理领域取得了显著的成就，并被广泛应用于各种应用程序中，包括智能助手、自动翻译、智能搜索等。这些模型在许多任务上表现出色，因为它们具备了大量的文本理解和生成能力，可以处理复杂的自然语言数据。

3.2.2 SFT 监督微调 fine-tuning：

目的在于为模型提供各种问答示例，教它更好地理解人类的意图和指令。"监督"即为对数据进行人为标注。通过一系列人工标注的问答示例，教模型如何理解人类的意图和指令。在大模型的生态链中，预训练（pre-training）和 SFT 监督微调（fine-tuning）就像搭建

一栋大厦的地基和室内装修。预训练是核心的“技术地基”，而微调更多是针对具体应用场景的“个性化优化”。预训练后的基础模型就像“技术平台”，微调则是根据不同企业需求进行定制开发。微调的目标是让模型适配具体场景，比如客服、医疗诊断、文案生成等。

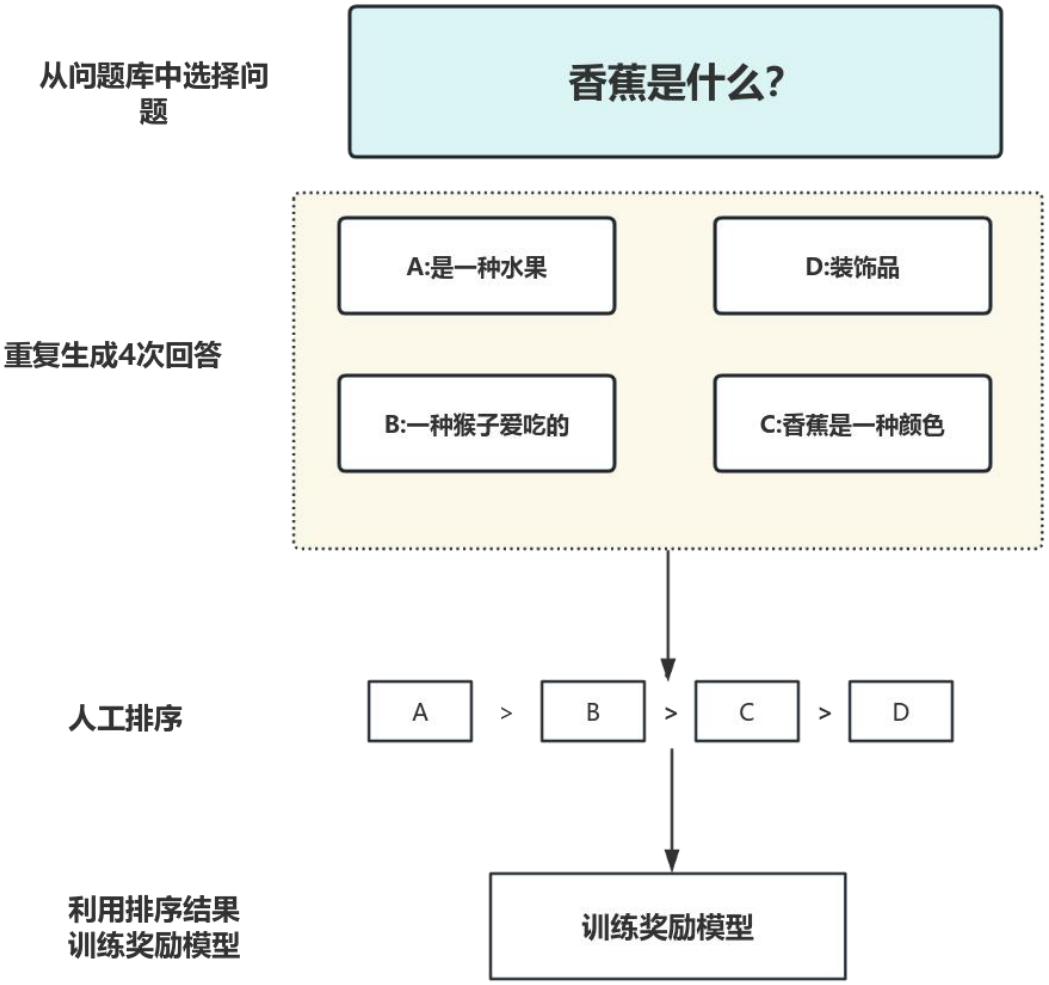
高质量预训练需要顶尖资源：预训练要求的算力、数据和算法是微调无法比拟的。比如 GPT-4 的预训练背后涉及上千台 GPU 和数十万亿参数调优，而微调只是建立在这些能力上的“修修补补””的细化行为。微调的任务更多是调整模型在具体场景中的表现，比如优化对话逻辑或生成风格。这些工作虽然重要，但并不涉及底层技术的革新，不具备深层次的技术壁垒。实力较强的大厂通常只专注于预训练，不主动参与微调。微调是面向场景的“下游应用”，大厂不需要亲自“装修”。大厂提供基础模型后，可以通过 API 或平台服务将模型开放给下游企业，后者根据自身需求完成微调。例如，阿里千问提供的基础模型已经通过 API 方式开放，企业只需“调用”即可。下游企业提也快因此可以获得更多创新空间

3.2.3 训练奖励模型（reward model）

我们需要一个模型来定量评判模型输出的回答在人类看来是否质量不错，即生成一个打分模型去评判大模型生成的效果。奖励模型的本质是让 AI 学会“什么是对的”，从而优化自身行为。

首先我们把大量的 prompt（提问的问题）（Open AI 使用调用 GPT-3 用户的真实数据）输入模型，让模型对同一个问题生成多个回

答。接下来让标注人员对同一个问题的不同回答排序（由于打分的主观性较强，不同的标注员对同一个回答可能会有不同的分数，所以采取排序的方法），这些不同的排序结果会通过某种归一化的方式变成定量的数据集，可以利用这个训练集训练出奖励模型。



3.2.4　RLHF 基于人类反馈的强化学习

根据奖励模型的评分，进一步优化模型生成策略，使其更符合人类偏好。简单来说，强化学习是让智能体（语言模型）通过「尝试动作→获得奖励→调整行为」的循环，学会在环境中做出最优决策。

拓展内容：把优化初始语言模型的任务，用强化学习（RL）的框架来理解和解决。因此需要定义策略(policy)、动作空间(action space)和奖励函数(reward function)等基本要素

Policy：智能体根据当前状态选择动作的「规则」。可以是确定性策略（比如 “看到问题就回答‘你好’”），也可以是概率性策略（比如 “有 80% 概率回答‘你好’，20% 概率问‘有什么可以帮你？’”）

类比：小狗的策略是「看到主人伸手，就抬左爪（因为之前这样做得到过骨头）」。

在语言模型中：策略是模型根据输入文本（如用户提问），决定输出什么文本（如回答内容）的规则。初始模型可能随机生成文本，微调后策略会更倾向于生成符合人类需求的回答。

2. 动作空间（Action Space）：智能体在环境中可以执行的所有可能动作的集合。

类比：小狗的动作空间是「坐下、抬爪、转圈、不动」等。

在语言模型中:动作空间是词汇表中的所有 token(比如中文的汉字、英文的单词片段）。

模型生成文本的过程,就是从动作空间中逐个选择 token 的过程(如生成 “你好” 时，先选 “你”，再选 “好”）。

3. 奖励函数（Reward Function）：评估智能体动作「好坏」的标准，用数值（奖励值）表示。正奖励（如 + 10）鼓励好的动作，负奖励

（如 - 5）惩罚坏的动作。

类比：小狗抬爪时你给骨头（+10），转圈时你不理它（-5），这些反馈就是奖励函数。

在语言模型中：需要设计一个函数，评估生成的文本是否符合要求，比如：

流畅性：句子是否通顺（如 “苹果吃我” 会被扣分）。

相关性：回答是否切题（如用户问 “天气”，回答 “猫很可爱” 会被扣分）。

安全性：是否避免敏感内容（如涉及暴力的回答会被扣分）。

奖励函数的设计是关键：它决定了模型最终学会 “什么样的回答是好的”。

3.3 大语言模型的核心目标

语言模型的核心目标是模仿人类语言的表达逻辑，其本质是通过数学方法判断一个句子是否合理流畅。传统语言模型（如 n-gram、RNN）像“逐字抄书的学徒”，只能根据前几个词预测下一个词（如“今天天气_”猜“很”），依赖局部上下文且效率低下，生成长文本时容易遗忘开头或逻辑混乱，就像写故事中途忘记主角名字。而 Transformer 架构的大语言模型（如 GPT、BERT）则像“通读全书后挥毫创作的大师”，通过自注意力机制同时分析所有词的关系（如跨段落关联伏笔与结局），既能并行处理数据大幅提升训练速度，又能保持长

文本的连贯性，其底层逻辑仍是概率计算——GPT 类模型逐词生成时显式判断概率分布（如“今天→天气→很→好”），BERT 类模型则通过“完形填空”隐式学习词语关联，两者均突破了传统模型的局部视野和效率瓶颈，使语言模型从简单的词语接龙工具升级为能创作、推理、跨任务泛化的智能体。

神经网络和语言模型发展历程

模型	主要创新	主要贡献
N-gram	基于词频统计	提供基础的统计语言模型
RNN LM	循环网络结构	建模长序列依赖的语言模型
长短时记忆	门控机制	解决 RNN 梯度消失
门控循环单元		
Transformer	自注意力机制	全局依赖建模,提升效率

3.3.1 自回归语言建模和掩码语言建模

语言模型的核心目标通过两类典型训练任务实现：自回归语言建模和掩码语言建模

1.自回归语言建模（如 n-gram、RNN、GPT）

任务目标：像人类写文章一样，从左到右逐词生成，每一步都根据“已经写出的内容”预测“下一个词”

例子：假设模型要生成句子“今天天气很好”。

输入：“今天”→ 预测下一个词可能是“天气”（概率最高）。

输入：“今天天气”→ 预测下一个词可能是“很”。

输入：“今天天气很”→ 预测下一个词可能是“好”。

最终生成完整句子：“今天天气很好”

应用场景：文本分类、问答、情感分析等需要深度理解的任务。

核心逻辑：基于前文预测下一个词的概率，显式建模序列的链式概率分布。如 P(今天)· P|(天气｜今天)· P(很｜今天， 天气)P(今天)·

单向预测：只能看到左侧的上下文（类似“蒙住右眼”）。

应用场景：写文章、聊天对话、翻译等需要连续生成的任务。

实现方式：

①n-gram：统计前 n-1 个词的出现频率，直接计算条件概率（如“今天”后出现“天气”的概率）。

②RNN：通过隐藏状态传递历史信息，动态更新词的概率分布（如生成对话时逐步调整预测）。

③Transformer 解码器（如 GPT）：利用自注意力机制捕捉全局依赖，生成更连贯的长文本。

2. 掩码语言建模（如 BERT）

任务目标：像做填空题一样，利用上下文双向信息，预测被遮盖的词。

实现方式：

例子：将句子“今天天气很好”中的“好”遮盖，变成“今天天气很

____”。

模型看到整个句子，分析上下文后预测“好”是最可能的词。推测出“今天天气很好_”

即使遮盖中间词，例如“今天___气很好”，模型也能通过“今天”和“气很好”推测出“天”。

核心逻辑：通过遮蔽部分词并预测其概率，隐式学习词与上下文的关联（如计算 P(天气 | 今天, [MASK], 很好)P(天气 | 今天, [MASK], 很好)）

双向理解：同时利用左右两侧的上下文（类似“睁双眼”全面观察）。

应用场景：文本分类、问答、情感分析等需要深度理解的任务

实现限制：

①传统模型无法实现。n-gram 和 RNN 依赖单向上下文（如遮蔽中间词后，RNN 无法逆向捕捉右侧信息）

Transformer 的优势：自注意力机制允许同时分析左右两侧上下文(如从“今天”和“很好”推断遮蔽词“天气”）。

3.4 价值观对齐：从技术到伦理

3.4.1 对齐的基本定义

“对齐”在这个上下文中指的是调整大型语言模型的输出，以使其符合人类的预期和特定需求。对齐是为了让大模型更加实用和安全。

更好用：

①符合用户预期：当用户向大型语言模型提出问题或任务时，他们通常期望模型的回答或生成的文本与问题或任务的上下文相关。对齐的目标是确保模型的输出与用户的预期一致。例如，当用户询问中国的首都时，预期的答案是"北京"，而不是其他无关的信息（模型可能会输出“美国的首都是哪里？德国的首都是哪里？...”，也可能输出“这是一个大家都知道的问题”。从续写的角度说，模型的回答可能都是正确的，只是不符合我们的预期罢了）。

②上下文敏感：对于一些任务，如搜索引擎查询或特定领域的信息检索，用户希望模型生成与输入上下文相关的结果。对齐可以确保模型能够理解上下文并生成适当的响应，而不是简单地执行续写任务。（比如助手模型当用户问“内蒙古包头的特色家乡菜”时，希望模型能输出对搜索引擎的调用，而不是由模型直接去做续写任务。）

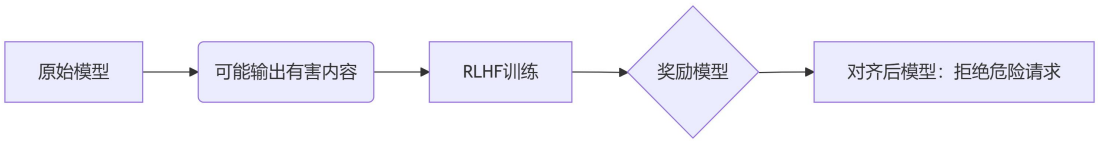
更安全：

①避免有害内容：对齐也可以用于限制模型生成可能有害或不适当的内容。例如，模型应该被设计成不生成涉及黄赌毒、暴力、恐怖主义等违法或不道德内容。对齐的一项任务是确保模型不会生成这类内容，

从而提高平台的安全性。

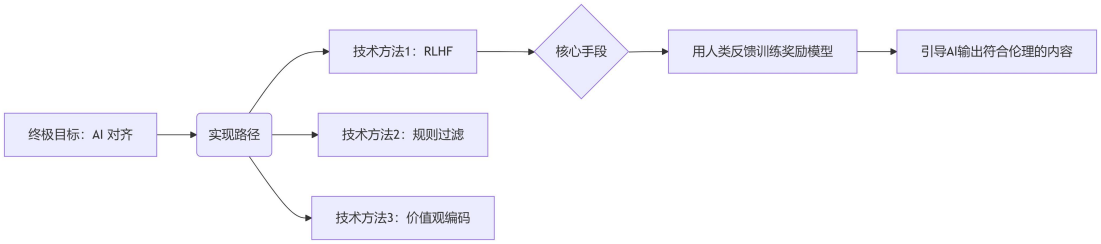
②合规性和道德：对于合规性和道德方面的问题，对齐也很重要。模型的输出应该符合适用法律和伦理规范，遵循隐私政策，并不会对用户或社会造成危害。

总之，对齐是一项关键任务，旨在确保大型语言模型的输出能够满足用户需求、预期和法规，同时提高模型的实用性和安全性。通过有效的对齐方法，可以更好地控制和引导模型的生成行为，使其更适合各种应用场景。



3.4.2 价值观对齐的核心方法：RLHF 基于人类反馈的强化学习

语言模型的对齐在 20 年就有相关工作，22 年谷歌提出基于指令微调的 Flan，Openai 则提出 InstructGPT，ChatGPT。目前，大模型的对齐工作中，RLHF 算法是能够达到最优的结果，RLHF 通过人类反馈和 PPO 算法，能够有效的将模型和人类偏好进行对齐（但是 RLHF 也存在难以训练，训练显存占用较大的缺点）



补充说明：SFT、RLHF、RL 皆为大模型预训练后的后训练技术，旨在提升模型能力并服务特定目标。预训练是模型能力的基石，SFT 和属于微调阶段，RLHF 则通过强化学习进一步优化输出质量。实际应用中，常组合使用：先预训练，再通过 SFT 微调，最后用 RLHF 对齐人类偏好，形成完整的模型训练链路。

第四章 LLM 的应用与实践

4.1 LLM 的核心能力

一、文本生成能力

定义：基于海量文本数据学习语言规律，生成语法正确、语义连贯的自然语言内容。

技术基础：

①Transformer 架构：通过自注意力机制捕捉全局语义关联（如 GPT-4 生成数千字连贯文章）。

②自回归训练：逐词预测下一词的概率分布，实现上下文驱动的连续生成（如续写故事时呼应前文伏笔）。

典型表现：

①对话交互：模拟人类对话逻辑（如 ChatGPT 处理日常问答）。

②内容创作：自动生成小说、诗歌、营销文案等（如 GPT-4 创作科幻故事）。

③代码生成：根据自然语言描述生成可执行代码（如 GitHub Copilot 生成 Python 算法）。

二、文本理解能力

定义：对文本进行语义解析、情感判断和逻辑推理，实现深层理解。

技术基础：

①预训练模型：通过掩码语言建模（如 BERT）或自回归建模（如 GPT）学习文本特征。

②上下文建模：自注意力机制捕捉长距离语义依赖（如识别复杂句子中的指代关系）。

典型表现：

①情感分析：判断文本情感倾向（如电商评论的正负向分类）。

②信息抽取：从合同、文献中提取关键实体（如人名、时间、条款）。

③逻辑推理：解决因果关系问题（如“下雨→比赛推迟”的推理）。

三、涌现能力：参数规模突破后的智能跃迁

随着模型规模的不断扩展，下游任务的性能和样本效率得到了显著的提高。但是当模型的大小突破某个临界规模的时候，开始出现一些意想不到的能力，展示出一些意外的、更为复杂的能力和特性，模

型能够自动地从原始训练数据中学习并发现新的、更为高级的特征和模式，这种能力通常被称为“涌现能力”。拥有涌现能力的机器学习模型被认为是在独立意义上更为强大的大型模型，这也是它们与小型模型最为显著的区别。

定义：当模型参数超过千亿级（如 GPT-3 的 1750 亿），自动涌现复杂推理、跨任务泛化等能力。

核心表现：

①上下文学习能力(In-context learning)

模型能够通过给定的自然语言指令和任务示例，生成预期的输出，而无需额外训练或梯度更新。这种能力在 GPT-3 等大规模模型中表现明显。

②按指令执行能力(Instruction following)

通过对多任务数据集进行指令微调，大模型能够遵循新任务的指令，在未见任务上表现良好，展现出改进的泛化能力。当模型规模达到一定程度（如 68B 参数）时，这种能力会显著提升。

③逐步推理能力(Step-by-step reasoning)

借助思维链（Chain-of-Thought）提示策略，大模型可以利用中间推理步骤来解决复杂的多步推理任务，如数学词问题等。当模型规模超过 60B 时，这种能力会显著增强。

④知识推理和迁移能力

大模型能够在看似不相关的任务之间进行知识迁移和推理，表现出通用的推理能力。

4.2 Agent 智能体与大模型

在大模型领域，智能体（Agent） 指的是一个能够感知环境、自主决策并执行任务的程序或系统。它基于大语言模型（如 GPT-4、LLaMA 等）的推理和生成能力，通过“思考-行动-反馈”的循环，逐步完成复杂目标。Open AI 将其定义为以大语言模型为大脑驱动，具备自主理解、感知、规划、记忆和使用工具的能力，能够自动化执行完成复杂任务的系统。将大模型类比为“大脑”，那 AI Agent 则是大模型的“手脚”，负责规划执行和落地。

可以把智能体想象成一个虚拟的“助手”：它不仅能回答问题（像 ChatGPT 那样），还能主动调用工具、分析信息、拆解任务，甚至通过多轮交互动态调整策略。**简单来说，它是一个能自主决策、执行任务的智能体，像私人助理一样 处理各种复杂任务。**

智能体 = 大模型+超级工具人

AI Agent 核心能力：

①感知与决策：能够从环境中获取信息（感知），并根据这些信息做出决策。

②行动执行：不仅限于文本处理，还可以执行物理或虚拟环境中的任务。

③多模态处理：可以处理多种类型的数据，如图像、声音、文本等，并根据这些数据做出综合决策。

④长期规划与学习：具备长期规划能力，能够在动态环境中不断学习和优化行为策略。

目前 AI Agent 主要应用在以下几个方向：

自动化系统：如智能家居控制系统、自动驾驶汽车，

游戏 AI：在游戏中扮演角色，与玩家互动。

客户服务：如智能客服系统，不仅可以回答问题，还可以协助解决实际问题。

工业自动化：如工厂中的机器人手臂，执行复杂的制造任务。

第五章 大模型局限性

尽管大模型让人们看到了通用人工智能的可能性，但目前来说它有三个非常大的局限性：知识的局限性、幻觉问题、数据安全问题。

5.1 知识的局限性（数据依赖）

知识的局限性，是指大模型所具备的知识，完全停留在了它训练完成的那一刻。也就是说，训练数据中所包含的知识，就是大模型的所有知识。比如 ChatGPT3.5 的知识停留在 2021 年 9 月，ChatGPT4 的知识停留在 2023 年 4 月。如果询问它这个日期之后的知识，在不联网的情况下，它是不可能知道的。从时间的维度去讲，它不具备时时更新的数据，训练数据截止日期后的信息无法掌握，需要在训练和微调时才能灌入新的知识。

5.2 幻觉问题（Hallucination）

指生成看似合理但不符合事实的内容。LLM 文本生成的底层原理是基于概率的 token by token 的形式，模型依赖训练数据的统计规律，而非真实客观知识因此会不可避免地产生“一本正经的胡说八道”的情况。比如输入“拿破仑哪年登上了月球？”，机器可能编造“1804 年”等错误答案。

5.3 数据安全问题

通用大语言模型没有企业内部数据和用户数据，那么企业想要在保证

安全的前提下使用大语言模型最好的方式就是把数据全部放在本地，企业数据的业务计算全部在本地完成。而在线的大模型仅仅完成一个归纳的功能。

5.4 数据偏见及伦理局限性（生成不当内容）

模型在生成文本时可能产生不适当、有害或歧视性的内容，引发道德和社会责任问题。

导致失忆的主要原因包括：

①训练数据中的偏见：模型可能在训练数据中学到了不适当的观点、偏见或刻板印象，导致生成不当内容。

②过度拟合负面样本：如果训练数据中包含大量负面样本，模型可能过度拟合这些负面情况，导致生成负面内容的可能性增加。

③缺乏伦理约束：模型训练时未考虑伦理和社会责任问题，缺乏对不适当内容的抑制；

附录

专有名词

模态（Modality）

数据的存在形式；文本/视觉/听觉/传感器数据（温度 速度 GPS 信号），多模态任务：要求模型输入或输出涉及多个模态，并实现模态间的信息互补与对齐。

Token

Token 是一种“最小单位的语言片段”。简单来说，它可以是一个词、一部分词、甚至一个标点符号。在计算机处理语言时，它会把输入的文字转换成这些小片段(token)，然后再通过这些 token 来理解和生成内容。大模型在接收到一段文字后通过分词器（节省脑力）将文字进行切分。token 表 每一个 token 对应一个编号，分词器将 token 表→人类可以理解的语言。

大模型靠依靠计算 token 与 token 之间的关系理解和生成文字，且大模型公司大多依靠 token 的数量来计费。

LangChain

开发框架，用来构建由大语言模型所支持的应用程序。

RAG

提示词增强；把外部的权威知识增加到提示词中，增加大语言模型输出的质量。

大模型预训练

在海量无标注数据上训练一个通用的基础模型，使其具备对语言、图像或其他模态数据的通用理解能力 ， 再通过微调（Fine-tuning）适配到具体任务。

涌现能力

当模型规模超过临界点（如千亿参数），会突然具备小模型没有的能力，如逻辑推理、代码生成。

调用 API

指通过预定义的规则和协议，让一个软件或服务能够请求另一个软件、服务或系统的特定功能或数据。

生成式模型

根据已有的信息创造出新的东西。

判别式模型

分类和判断任务，比如判断垃圾邮件。

提示词工程 prompt

Prompt 是一种注入式指令，用于“指挥”AI 按照预设的思路去思考问题、输出内容。它是一种指令或信息，引导或触发 AI 系统做出回应。