



图灵程序设计丛书

# 大模型应用开发极简入门： 基于GPT-4和ChatGPT

---

## Developing Apps with GPT-4 and ChatGPT

[比] 奥利维耶·卡埃朗 (Olivier Caelen) 著  
[法] 玛丽-艾丽斯·布莱特 (Marie-Alice Blete)  
何文斯 译

Beijing • Boston • Farnham • Sebastopol • Tokyo

**O'REILLY®**

O'Reilly Media, Inc. 授权人民邮电出版社有限公司出版

人 民 邮 电 出 版 社  
北 京

图书在版编目(CIP)数据

大模型应用开发极简入门：基于GPT-4和ChatGPT /  
(比) 奥利维耶·卡埃朗 (Olivier Caelen), (法) 玛丽-  
艾丽斯·布莱特 (Marie-Alice Blete) 著 ; 何文斯译  
. -- 北京 : 人民邮电出版社, 2024.2  
(图灵程序设计丛书)  
ISBN 978-7-115-63640-9

I. ①大… II. ①奥… ②玛… ③何… III. ①人工智  
能 IV. ①TP18

中国国家版本馆CIP数据核字(2024)第010776号

内 容 提 要

本书为大模型应用开发极简入门手册，为初学者提供了一份清晰、全面的“最小可用知识”，带领大家快速了解 GPT-4 和 ChatGPT 的工作原理及优势，并在此基础上使用流行的 Python 编程语言构建大模型应用。通过本书，你不仅可以学会如何构建文本生成、问答和内容摘要等初阶大模型应用，还能了解到提示工程、模型微调、插件、LangChain 等高阶实践技术。本书提供了简单易学的示例，帮你理解相关概念并应用在自己的项目中。此外，书后还提供了一份术语表，方便你随时参考。

准备好了吗？只需了解 Python，你即可将本书作为进入大模型时代的启动手册，开发出自己的大模型应用。

- 
- ◆ 著 [比] 奥利维耶·卡埃朗 (Olivier Caelen)  
[法] 玛丽-艾丽斯·布莱特 (Marie-Alice Blete)  
译 何文斯  
责任编辑 刘美英  
责任印制 胡 南
- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <https://www.ptpress.com.cn>  
北京 印刷
- ◆ 开本：880×1230 1/32  
印张：5.5 2024年2月第1版  
字数：181千字 2024年2月北京第1次印刷  
著作权合同登记号 图字：01-2023-5478号
- 

定价：59.80元

读者服务热线：(010)84084456-6009 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东市监广登字 20170147 号

# 版权声明

Copyright © 2023 Olivier Caelen and Marie-Alice Blete. All rights reserved.

Simplified Chinese edition, jointly published by O'Reilly Media, Inc. and Posts & Telecom Press, 2024. Authorized translation of the English edition, 2024 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc. 出版，2023。

简体中文版由人民邮电出版社有限公司出版，2024。英文原版的翻译得到 O'Reilly Media, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有，未得书面许可，本书的任何部分和全部不得以任何形式重制。

# O'Reilly Media, Inc. 介绍

O'Reilly 以“分享创新知识、改变世界”为己任。40 多年来我们一直向企业、个人提供成功所必需之技能及思想，激励他们创新并做得更好。

O'Reilly 业务的核心是独特的专家及创新者网络，众多专家及创新者通过我们分享知识。我们的在线学习（Online Learning）平台提供独家的直播培训、互动学习、认证体验、图书、视频等，使客户更容易获取业务成功所需的专业知识。几十年来 O'Reilly 图书一直被视为学习开创未来之技术的权威资料。我们所做的一切是为了帮助各领域的专业人士学习最佳实践，发现并塑造科技行业未来的新趋势。

我们的客户渴望做出推动世界前进的创新之举，我们希望能助他们一臂之力。

## 业界评论

“O'Reilly Radar 博客有口皆碑。”

—— *Wired*

“O'Reilly 凭借一系列非凡想法（真希望当初我也想到了）建立了数百万美元的业务。”

—— *Business 2.0*

“O'Reilly Conference 是聚集关键思想领袖的绝对典范。”

—— *CRN*

“一本 O'Reilly 的书就代表一个有用、有前途、需要学习的主题。”

—— *Irish Times*

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的领域，并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路上遇到岔路口，那就走小路。’回顾过去，Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

—— *Linux Journal*

# 目录

推荐序一 学习成为善用 AI 的人 | 宝玉

推荐序二 开启一段有趣、有启发、有收获的冒险旅程 | 张路宇

推荐序三 人人都要学会和 AI 相处 | 孙志岗

推荐序四 AI 工程师：做新一轮智能革命的首批探索者 | 邓范鑫

推荐序五 进入智能应用的新时代 | 梁宇鹏（@ 一乐）

推荐序六 AGI：不要旁观，要真正参与 | 罗云

推荐序七 不要害怕被 ChatGPT 取代，要做第一批驾驭新技术的人 | 宜博

译者序 没有谁天生就是 AI 工程师 | 何文斯

前言

第 1 章 初识 GPT-4 和 ChatGPT .....	1
1.1 LLM 概述 .....	2
1.1.1 探索语言模型和 NLP 的基础 .....	2
1.1.2 理解 Transformer 架构及其在 LLM 中的作用 .....	4
1.1.3 解密 GPT 模型的标记化和预测步骤 .....	7
1.2 GPT 模型简史：从 GPT-1 到 GPT-4 .....	9
1.2.1 GPT-1 .....	9
1.2.2 GPT-2 .....	10
1.2.3 GPT-3 .....	11
1.2.4 从 GPT-3 到 InstructGPT .....	12
1.2.5 GPT-3.5、Codex 和 ChatGPT .....	14
1.2.6 GPT-4 .....	15
1.3 LLM 用例和示例产品 .....	16
1.3.1 Be My Eyes .....	16
1.3.2 摩根士丹利 .....	17
1.3.3 可汗学院 .....	17
1.3.4 多邻国 .....	18
1.3.5 Yabble .....	18
1.3.6 Waymark .....	19
1.3.7 Inworld AI .....	19
1.4 警惕 AI 幻觉：限制与考虑 .....	20
1.5 使用插件和微调优化 GPT 模型 .....	23
1.6 小结 .....	24
第 2 章 深入了解 GPT-4 和 ChatGPT 的 API .....	25
2.1 基本概念 .....	26
2.2 OpenAI API 提供的可用模型 .....	27
2.3 在 OpenAI Playground 中使用 GPT 模型 .....	29
2.4 开始使用 OpenAI Python 库 .....	34
2.4.1 OpenAI 访问权限和 API 密钥 .....	35
2.4.2 Hello World 示例程序 .....	36

2.5	使用 GPT-4 和 ChatGPT .....	38
2.5.1	ChatCompletion 端点的输入选项 .....	39
2.5.2	ChatCompletion 端点的输出格式 .....	42
2.5.3	从文本补全到函数 .....	43
2.6	使用其他文本补全模型 .....	46
2.6.1	Completion 端点的输入选项 .....	47
2.6.2	Completion 端点的输出格式 .....	48
2.7	考虑因素 .....	48
2.7.1	定价和标记限制 .....	48
2.7.2	安全和隐私 .....	50
2.8	其他 OpenAI API 和功能 .....	50
2.8.1	嵌入 .....	50
2.8.2	内容审核模型 .....	53
2.8.3	Whisper 和 DALL·E .....	55
2.9	小结 (含速查清单) .....	56
<b>第 3 章 使用 GPT-4 和 ChatGPT 构建应用程序 .....</b>		<b>59</b>
3.1	应用程序开发概述 .....	59
3.1.1	管理 API 密钥 .....	60
3.1.2	数据安全和数据隐私 .....	62
3.2	软件架构设计原则 .....	62
3.3	LLM 驱动型应用程序的漏洞 .....	63
3.3.1	分析输入和输出 .....	64
3.3.2	无法避免提示词注入 .....	65
3.4	示例项目 .....	65
3.4.1	项目 1: 构建新闻稿生成器 .....	65
3.4.2	项目 2: YouTube 视频摘要 .....	68
3.4.3	项目 3: 打造《塞尔达传说: 旷野之息》专家 .....	71
3.4.4	项目 4: 语音控制 .....	77
3.5	小结 .....	83

第 4 章 GPT-4 和 ChatGPT 的高级技巧	85
4.1 提示工程	85
4.1.1 设计有效的提示词	86
4.1.2 逐步思考	92
4.1.3 实现少样本学习	94
4.1.4 改善提示效果	96
4.2 微调	98
4.2.1 开始微调	99
4.2.2 使用 OpenAI API 进行微调	101
4.2.3 微调的应用	105
4.2.4 生成和微调电子邮件营销活动的合成数据	107
4.2.5 微调的成本	113
4.3 小结	114
第 5 章 使用 LangChain 框架和插件增强 LLM 的功能	117
5.1 LangChain 框架	117
5.1.1 动态提示词	119
5.1.2 智能体及工具	120
5.1.3 记忆	124
5.1.4 嵌入	125
5.2 GPT-4 插件	129
5.2.1 概述	130
5.2.2 API	131
5.2.3 插件清单	132
5.2.4 OpenAPI 规范	133
5.2.5 描述	135
5.3 小结	135
5.4 总结	136
术语表	137



---

# 前言

在发布仅 5 天后，ChatGPT 就吸引了 100 万用户。这样的成绩震撼了科技行业甚至其他行业。尽管用于人工智能文本生成的 OpenAI API 在 3 年前就已诞生，但随着 ChatGPT 的成功，它突然获得极大的关注。ChatGPT 的界面展示了这类语言模型的巨大潜力。突然之间，开发人员和技术创造者意识到，梦寐以求的机遇触手可及。

多年来，自然语言处理领域取得了长足的进步，但直到最近，这项技术的使用者还仅限于少数精英。OpenAI API 及其附带的库为所有想构建人工智能应用程序<sup>1</sup>的人提供了即插即用的解决方案。无须拥有强大的硬件或深厚的人工智能知识，开发人员只需利用几行代码，就能以合理的成本在项目集成强大的功能。

本书作者奥利维耶是机器学习研究员，玛丽 - 艾丽斯是软件架构师和数据工程师。他们结合自身的知识和经验，帮助你从整体上理解如何使用 GPT-4 和 ChatGPT 开发应用程序。本书清晰、详细地解释了人工智能概念，并以通俗易懂的方式指导你学习如何高效、安全、低成本地集成 OpenAI 服务。

本书旨在让所有人都能理解所讲的内容，但我们仍建议你具备基础的 Python 知识。通过清晰的解释、示例项目和逐步指导，我们邀请你与我们一起探索 GPT-4 和 ChatGPT 如何改变人机交互方式。

## 排版约定

本书使用下列排版约定。

---

注 1：本书有时将“应用程序”简称为“应用”，两者的意思一致。——编者注

## 黑体

表示新术语或重点强调的内容。

## 等宽字体 (`constant width`)

表示程序片段，以及正文中出现的变量、函数、数据库、数据类型、环境变量、语句和关键字等。

## 加粗等宽字体 (**`constant width bold`**)

表示应该由用户输入的命令或其他文本。

## 等宽斜体 (*`constant width italic`*)

表示应该由用户输入的值或根据上下文确定的值替换的文本。



该图标表示提示或建议。



该图标表示一般笔记。



该图标表示警告或警示。

# 使用代码示例

可以从 [https://github.com/malywut/gpt\\_examples](https://github.com/malywut/gpt_examples) 下载补充材料（代码示例、练习等）<sup>2</sup>。

本书是要帮你完成工作的。一般来说，你可以把本书提供的代码示例用在你的程序或文档中。除非你使用了很大一部分代码，否则无须联系我们获得许可。比如，用本书的几个代码片段写一个程序就无须获得许可，销售或分发

---

注 2：也可以通过本书在图灵社区的专属页面下载：[ituring.cn/book/3344](http://ituring.cn/book/3344)。——编者注

O'Reilly 图书的示例光盘则需要获得许可；引用本书中的代码示例回答问题无须获得许可，将书中大量的代码放到你的产品文档中则需要获得许可。

我们很希望但并不强制要求你在引用本书内容时加上引用说明。引用说明一般包括书名、作者、出版社和 ISBN，比如 “*Developing Apps with GPT-4 and ChatGPT* by Olivier Caelen and Marie-Alice Blete (O'Reilly). Copyright 2023 Olivier Caelen and Marie-Alice Blete, 978-1-098-15248-2”。

如果你认为自己对代码示例的用法超出了上述许可的范围，欢迎你通过 [permissions@oreilly.com](mailto:permissions@oreilly.com) 与我们联系。

## O'Reilly 在线学习平台 (O'Reilly Online Learning)

**O'REILLY**® 40 多年来，O'Reilly Media 致力于提供技术和商业培训、知识和卓越见解，来帮助众多公司取得成功。

我们拥有由专家和创新者组成的庞大网络，他们通过图书、文章和我们的在线学习平台分享他们的知识和经验。O'Reilly 在线学习平台让你能够按需访问现场培训课程、深入的学习路径、交互式编程环境，以及 O'Reilly 和 200 多家其他出版商提供的大量文本资源和视频资源。更多信息，请访问 <https://www.oreilly.com>。

## 联系我们

请把对本书的评价和问题发给出版社。

美国：

O'Reilly Media, Inc.  
1005 Gravenstein Highway North  
Sebastopol, CA 95472

中国：

北京市西城区西直门南大街 2 号成铭大厦 C 座 807 室 (100035)  
奥莱利技术咨询 (北京) 有限公司

请访问 <https://oreil.ly/devAppsGPT>，查看相关勘误<sup>3</sup>。

---

注 3：本书中文版勘误请到 [ituring.cn/book/3344](https://ituring.cn/book/3344) 查看和提交。——编者注

对于本书的评论和技术性问题，请发送电子邮件到 [errata@oreilly.com.cn](mailto:errata@oreilly.com.cn)。

要了解更多 O'Reilly 图书和培训课程等信息，请访问以下网站：<https://www.oreilly.com>。

我们在 LinkedIn 的地址如下：<https://linkedin.com/company/oreilly-media>。

请关注我们的 Twitter 动态：<https://twitter.com/oreillymedia>。

我们的 YouTube 视频地址如下：<https://youtube.com/oreillymedia>。

## 致谢

针对发展速度最快的人工智能话题写一本书离不开许多人的帮助。我们要感谢杰出的 O'Reilly 团队给予的支持、建议和中肯的评论，特别是 Corbin Collins、Nicole Butterfield、Clare Laylock、Suzanne Huston 和 Audrey Doyle。

本书还得益于多位优秀审稿人的帮助，他们花费大量时间提供了宝贵的反馈。非常感谢 Tom Taulli、Lucas Soares 和 Leonie Monigatti。

非常感谢我们在 Worldline 的同事分享对 ChatGPT 和 OpenAI 服务的见解，也感谢你们参与永无止境的讨论。特别感谢 Liyun He Guelton、Guillaume Coter、Luxin Zhang 和 Patrik De Boe。同样感谢 Worldline 的 Developer Advocate 团队从一开始就给予的支持和鼓励，特别感谢 Jean-Francois James 和 Fanilo Andrianasolo。

最后，感谢朋友和家人在我们疯狂使用 ChatGPT 期间所给予的耐心和理解。正因为如此，我们才能在如此短的时间内出版本书。

## 电子书

扫描以下二维码，即可购买本书中文版电子书。



# 初识 GPT-4 和 ChatGPT

想象这样一个世界：在这个世界里，你可以像和朋友聊天一样快速地与计算机交互。那会是怎样的体验？你可以创造出什么样的应用程序？这正是 OpenAI 努力构建的世界，它通过其 GPT 模型让设备拥有与人类对话的能力。作为人工智能（artificial intelligence, AI）领域的最新成果，GPT-4 和其他 GPT 模型是基于大量数据训练而成的大语言模型<sup>1</sup>（large language model, LLM），它们能够以非常高的准确性识别和生成人类可读的文本。

这些 AI 模型的意义远超简单的语音助手。多亏了 OpenAI 的模型，开发人员现在可以利用自然语言处理（natural language processing, NLP）技术创建应用程序，使其以一种曾经只存在于科幻小说中的方式理解我们的需求。从学习和适应个体需求的创新型客户支持系统，到理解每个学生独特的学习风格的个性化教学工具，GPT-4 和 ChatGPT 打开了一扇门，让人们看见一个充满可能性的全新世界。

GPT-4 和 ChatGPT 究竟是什么？本章的目标是深入探讨这些 AI 模型的基础、起源和关键特性。通过了解这些模型的基础知识，你将为构建下一代以 LLM 驱动的应用程序打下坚实的基础。

---

注 1：“大语言模型”简称“大模型”。在本书中，两者的意思相同。——编者注

# 1.1 LLM 概述

本节介绍塑造 GPT-4 和 ChatGPT 发展历程的基础模块。我们旨在帮助你全面理解语言模型、NLP 技术、Transformer 架构的作用，以及 GPT 模型中的标记化和预测过程。

## 1.1.1 探索语言模型和NLP的基础

作为 LLM，GPT-4 和 ChatGPT 是 NLP 领域中最新的模型类型，NLP 是机器学习和人工智能的一个子领域。在深入研究 GPT-4 和 ChatGPT 之前，有必要了解 NLP 及其相关领域。

AI 有不同的定义，但其中一个定义或多或少已成为共识，即 AI 是一类计算机系统，它能够执行通常需要人类智能才能完成的任务。根据这个定义，许多算法可以被归为 AI 算法，比如导航应用程序所用的交通预测算法或策略类视频游戏所用的基于规则的系统。从表面上看，在这些示例中，计算机似乎需要智能才能完成相关任务。

机器学习（machine learning，ML）是 AI 的一个子集。在 ML 中，我们不试图直接实现 AI 系统使用的决策规则。相反，我们试图开发算法，使系统能够通过示例自己学习。自从在 20 世纪 50 年代开始进行 ML 研究以来，人们已经在科学文献中提出了许多 ML 算法。

在这些 ML 算法中，深度学习（deep learning，DL）算法已经引起了广泛关注。DL 是 ML 的一个分支，专注于受大脑结构启发的算法。这些算法被称为人工神经网络（artificial neural network）。它们可以处理大量的数据，并且在图像识别、语音识别及 NLP 等任务上表现出色。

GPT-4 和 ChatGPT 基于一种特定的神经网络架构，即 Transformer。Transformer 就像阅读机一样，它关注句子或段落的不同部分，以理解其上下文并产生连贯的回答。此外，它还可以理解句子中的单词顺序和上下文意思。这使 Transformer 在语言翻译、问题回答和文本生成等任务中非常有效。图 1-1 展示了以上术语之间的关系。

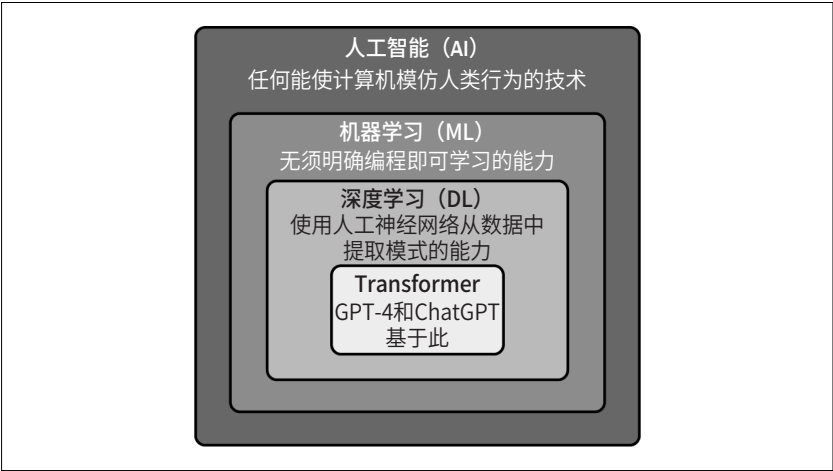


图 1-1：从 AI 到 Transformer 的嵌套技术集合

NLP 是 AI 的一个子领域，专注于使计算机能够处理、解释和生成人类语言。现代 NLP 解决方案基于 ML 算法。NLP 的目标是让计算机能够处理自然语言文本。这个目标涉及诸多任务，如下所述。

#### 文本分类

将输入文本归为预定义的类别。这类任务包括情感分析和主题分类。比如，某公司使用情感分析来了解客户对其服务的意见。电子邮件过滤是主题分类的一个例子，其中电子邮件可以被归类为“个人邮件”“社交邮件”“促销邮件”“垃圾邮件”等。

#### 自动翻译

将文本从一种语言自动翻译成另一种语言。请注意，这类任务可以包括将代码从一种程序设计语言翻译成另一种程序设计语言，比如从 Python 翻译成 C++。

#### 问题回答

根据给定的文本回答问题。比如，在线客服门户网站可以使用 NLP 模型回答关于产品的常见问题；教学软件可以使用 NLP 模型回答学生关于所学主题的问题。

文本生成

根据给定的输入文本（称为提示词<sup>2</sup>）生成连贯且相关的输出文本。

如前所述，LLM 是试图完成文本生成任务的一类 ML 模型。LLM 使计算机能够处理、解释和生成人类语言，从而提高人机交互效率。为了做到这一点，LLM 会分析大量文本数据或基于这些数据进行训练，从而学习句子中各词之间的模式和关系。这个学习过程可以使用各种数据源，包括维基百科、Reddit、成千上万本书，甚至互联网本身。在给定输入文本的情况下，这个学习过程使得 LLM 能够预测最有可能出现的后续单词，从而生成对输入文本有意义的回应。于 2023 年发布的一些现代语言模型非常庞大，并且已经在大量文本上进行了训练，因此它们可以直接执行大多数 NLP 任务，如文本分类、自动翻译、问题回答等。GPT-4 和 ChatGPT 是在文本生成任务上表现出色的 LLM。

LLM 的发展可以追溯到几年前。它始于简单的语言模型，如  $n$ -gram 模型。 $n$ -gram 模型通过使用词频来根据前面的词预测句子中的下一个词，其预测结果是在训练文本中紧随前面的词出现的频率最高的词。虽然这种方法提供了不错的着手点，但是  $n$ -gram 模型在理解上下文和语法方面仍需改进，因为它有时会生成不连贯的文本。

为了提高  $n$ -gram 模型的性能，人们引入了更先进的学习算法，包括循环神经网络（recurrent neural network, RNN）和长短期记忆（long short-term memory, LSTM）网络。与  $n$ -gram 模型相比，这些模型能够学习更长的序列，并且能够更好地分析上下文，但它们在处理大量数据时的效率仍然欠佳。尽管如此，在很长的一段时间里，这些模型算是最高效的，因此在自动翻译等任务中被广泛使用。

### 1.1.2 理解Transformer架构及其在LLM中的作用

Transformer 架构彻底改变了 NLP 领域，这主要是因为它能够有效地解决之前的 NLP 模型（如 RNN）存在的一个关键问题：很难处理长文本序列并记

---

注 2：对于 prompt 一词，本书统一采用“提示词”这个译法，以符合业内惯例。不过，prompt 既可以是一个词，也可以是一个或多个句子。对于 prompt engineering，本书采用“提示工程”这个译法。——译者注



住其上下文。换句话说，RNN 在处理长文本序列时容易忘记上下文（也就是臭名昭著的“灾难性遗忘问题”），Transformer 则具备高效处理和编码上下文的能力。

这场革命的核心支柱是**注意力机制**，这是一个简单而又强大的机制。模型不再将文本序列中的所有词视为同等重要，而是在任务的每个步骤中关注最相关的词。**交叉注意力**和**自注意力**是基于注意力机制的两个架构模块，它们经常出现在 LLM 中。Transformer 架构广泛使用了交叉注意力模块和自注意力模块。

交叉注意力有助于模型确定输入文本的不同部分与输出文本中下一个词的相关性。它就像一盏聚光灯，照亮输入文本中的词或短语，并突出显示预测下一个词所需的相关信息，同时忽略不重要的细节。

为了说明这一点，让我们以一个简单的句子翻译任务为例。假设输入文本是这样一个英语句子：Alice enjoyed the sunny weather in Brussels（Alice 很享受布鲁塞尔阳光明媚的天气）。如果目标语言是法语，那么输出文本应该是：Alice a profit  du temps ensoleill    Bruxelles。在这个例子中，让我们专注于生成法语单词 ensoleill ，它对应原句中的 sunny。对于这个预测任务，交叉注意力模块会更关注英语单词 sunny 和 weather，因为它们都与 ensoleill  相关。通过关注这两个单词，交叉注意力模块有助于模型为句子的这一部分生成准确的翻译结果，如图 1-2 所示。

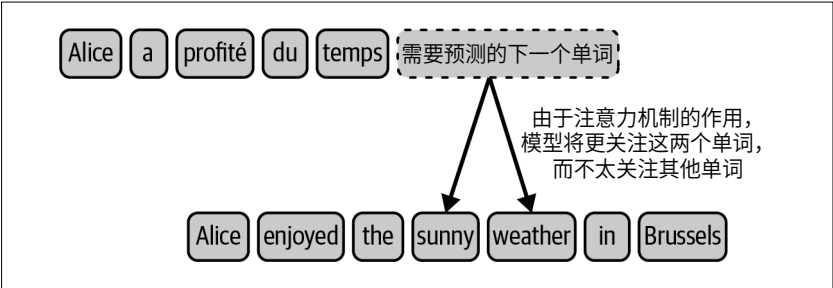


图 1-2：交叉注意力模块使模型关注输入文本（英语句子）中的关键部分，以预测输出文本（法语句子）中的下一个单词

自注意力机制是指模型能够关注其输入文本的不同部分。具体到 NLP 领域，自注意力机制使模型能够评估句子中的每个词相比于其他词的重要性。这使得模型能够更好地理解各词之间的关系，并根据输入文本中的多个词构建新概念。

来看一个更具体的例子。考虑以下句子：Alice received praise from her colleagues (Alice 受到同事的赞扬)。假设模型试图理解 her 这个单词的意思。自注意力机制给句子中的每个单词分配不同的权重，突出在这个上下文中与 her 相关的单词。在本例中，自注意力机制会更关注 Alice 和 colleagues 这两个单词。如前所述，自注意力机制帮助模型根据这些单词构建新概念。在本例中，可能出现的一个新概念是 Alice's colleagues，如图 1-3 所示。

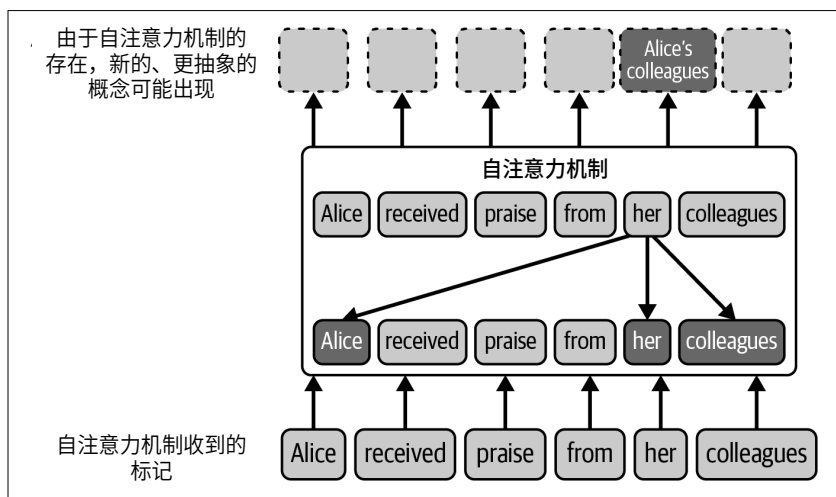


图 1-3：自注意力机制使新概念 Alice's colleagues 得以出现

与 RNN 不同，Transformer 架构具有易于并行化的优势。这意味着 Transformer 架构可以同时处理输入文本的多个部分，而无须顺序处理。这样做可以提高计算速度和训练速度，因为模型的不同部分可以并行工作，而无须等待前一步骤完成。基于 Transformer 架构的模型所具备的并行处理能力与图形处理单元（graphics processing unit, GPU）的架构完美契合，后者专用于同时处理多个计算任务。由于高度的并行性和强大的计算能力，GPU 非常适合用于训练和运行基于 Transformer 架构的模型。硬件上的这一进展使数据

科学家能够在大型数据集上训练模型，从而为开发 LLM 铺平了道路。

Transformer 架构由来自谷歌公司的 Ashish Vaswani 等人在 2017 年的论文“Attention Is All You Need”中提出，最初用于序列到序列的任务，如机器翻译任务。标准的 Transformer 架构有两个主要组件：编码器和解码器，两者都十分依赖注意力机制。编码器的任务是处理输入文本，识别有价值的特征，并生成有意义的文本表示，称为嵌入（embedding）。解码器使用这个嵌入来生成一个输出，比如翻译结果或摘要文本。这个输出有效地解释了编码信息。

生成式预训练 Transformer（Generative Pre-trained Transformer，GPT）是一类基于 Transformer 架构的模型，专门利用原始架构中的解码器部分。在 GPT 中，不存在编码器，因此无须通过交叉注意力机制来整合编码器产生的嵌入。也就是说，GPT 仅依赖解码器内部的自注意力机制来生成上下文感知的表示和预测结果。请注意，BERT 等其他一些众所周知的模型基于编码器部分，但本书不涉及这类模型。图 1-4 展示了 NLP 技术的演变历程。

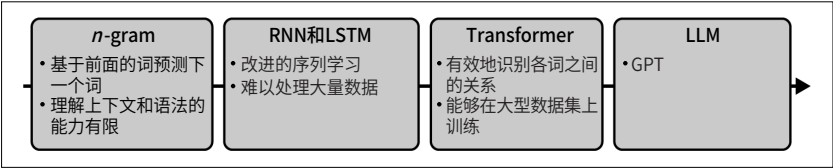


图 1-4：NLP 技术从  $n$ -gram 到 LLM 的演变

### 1.1.3 解密GPT模型的标记化和预测步骤

GPT 模型接收一段提示词作为输入，然后生成一段文本作为输出。这个过程被称为文本补全。举例来说，提示词可以是 The weather is nice today, so I decided to （今天天气很好，所以我决定），模型的输出则可能是 go for a walk （去散步）。你可能想知道 GPT 模型是如何根据输入的提示词构建输出文本的。正如你将看到的，这主要是一个概率问题。

当 GPT 模型收到一段提示词之后，它首先将输入拆分成标记（token）。这些标记代表单词、单词的一部分、空格或标点符号。比如，在前面的例子中，提示词可以被拆分成 [The, wea, ther, is, nice, today, ,, so, I, de, ci, ded, to]。

几乎每个语言模型都配有自己的分词器。截至本书英文版出版之时，GPT-4 的分词器还不可用<sup>3</sup>，不过你可以尝试使用 GPT-3 的分词器。



理解标记与词长的一条经验法则是，对于英语文本，100 个标记大约等于 75 个单词。

因为有了注意力机制和 Transformer 架构，LLM 能够轻松处理标记并解释它们之间的关系及提示词的整体含义。Transformer 架构使模型能够高效地识别文本中的关键信息和上下文。

为了生成新的句子，LLM 根据提示词的上下文预测最有可能出现的下一个标记。OpenAI 开发了两个版本的 GPT-4，上下文窗口大小分别为 8192 个标记和 32 768 个标记<sup>4</sup>。与之前的循环模型不同，带有注意力机制的 Transformer 架构使得 LLM 能够将上下文作为一个整体来考虑。基于这个上下文，模型为每个潜在的后续标记分配一个概率分数，然后选择概率最高的标记作为序列中的下一个标记。在前面的例子中，“今天天气很好，所以我决定”之后，下一个最佳标记可能是“去”。

接下来重复此过程，但现在上下文变为“今天天气很好，所以我决定去”，之前预测的标记“去”被添加到原始提示词中。这个过程会一直重复，直到形成一个完整的句子：“今天天气很好，所以我决定去散步。”这个过程依赖于 LLM 学习从大量文本数据中预测下一个最有可能出现的单词的能力。图 1-5 展示了这个过程。

---

注 3：现在，OpenAI 已在其网站上提供了 GPT-4 的分词器。——译者注

注 4：请注意，本书中的译者注的添加时间为 2023 年 11 月 19 日~2023 年 12 月 2 日，在此统一说明，后续不再逐一详细说明。截至 2023 年 11 月下旬，OpenAI 已提供 6 个 GPT-4 模型，包括 gpt-4-1106-preview、gpt-4-vision-preview、gpt-4、gpt-4-32k、gpt-4-0613、gpt-4-32k-0613，其中 gpt-4-1106-preview 的上下文窗口已增加至 16 385 个标记。——译者注

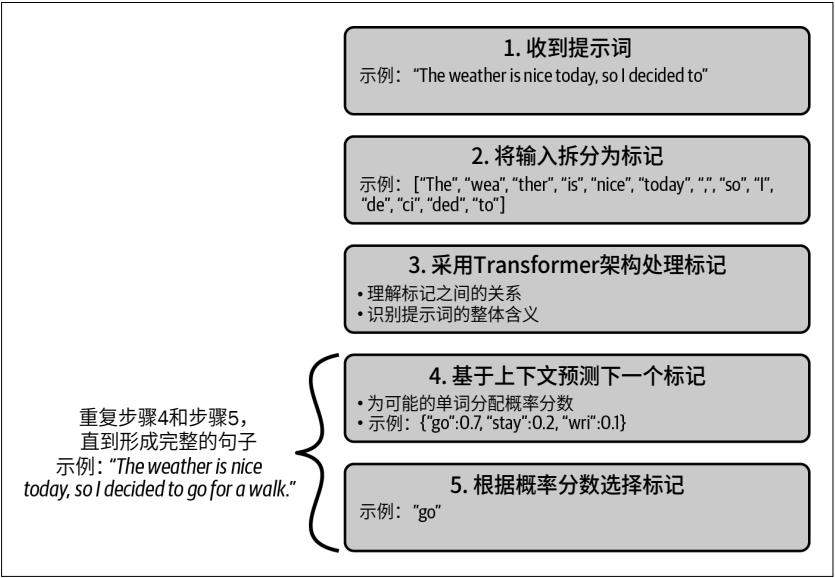


图 1-5: 逐个标记地补全文本，整个过程是迭代式的

## 1.2 GPT 模型简史：从 GPT-1 到 GPT-4

本节将回顾 OpenAI 的 GPT 模型从 GPT-1 到 GPT-4 的演变历程。

### 1.2.1 GPT-1

2018 年年中，就在 Transformer 架构诞生一年后，OpenAI 发表了一篇题为“Improving Language Understanding by Generative Pre-Training”的论文，作者是 Alec Radford 等人。这篇论文介绍了 GPT，也被称为 GPT-1。

在 GPT-1 出现之前，构建高性能 NLP 神经网络的常用方法是利用监督学习。这种学习技术使用大量的手动标记数据。以情感分析任务为例，目标是对给定的文本进行分类，判断其情感是积极的还是消极的。一种常见的策略是收集数千个手动标记的文本示例来构建有效的分类模型。然而，这需要大量标记良好的监督数据。这一需求限制了监督学习的性能，因为要生成这样的数据集，难度很大且成本高昂。

在论文中，GPT-1 的作者提出了一种新的学习过程，其中引入了无监督的预训练步骤。这个预训练步骤不需要标记数据。相反，他们训练模型来预测下一个标记。由于采用了可以并行化的 Transformer 架构，预训练步骤是在大量数据上进行的。对于预训练，GPT-1 模型使用了 BookCorpus 数据集。该数据集包含约 11 000 本未出版图书的文本。BookCorpus 最初由 Yukun Zhu 等人在 2015 年的论文“Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books”中给出，并通过多伦多大学的网页提供。然而，原始数据集的正式版本如今已不能公开访问。

人们发现，GPT-1 在各种基本的文本补全任务中是有效的。在无监督学习阶段，该模型学习 BookCorpus 数据集并预测文本中的下一个词。然而，GPT-1 是小模型，它无法在不经微调的情况下执行复杂任务。因此，人们将微调作为第二个监督学习步骤，让模型在一小部分手动标记的数据上进行微调，从而适应特定的目标任务。比如，在情感分析等分类任务中，可能需要在一小部分手动标记的文本示例上重新训练模型，以使其达到不错的准确度。这个过程使模型在初始的预训练阶段习得的参数得到修改，从而更好地适应具体的任务。

尽管规模相对较小，但 GPT-1 在仅用少量手动标记的数据进行微调后，能够出色地完成多个 NLP 任务。GPT-1 的架构包括一个解码器（与原始 Transformer 架构中的解码器类似），具有 1.17 亿个参数。作为首个 GPT 模型，它为更强大的模型铺平了道路。后续的 GPT 模型使用更大的数据集和更多的参数，更好地利用了 Transformer 架构的潜力。

## 1.2.2 GPT-2

2019 年初，OpenAI 提出了 GPT-2。这是 GPT-1 的一个扩展版本，其参数数量和训练数据集的规模大约是 GPT-1 的 10 倍。这个新版本的参数数量为 15 亿，训练文本为 40 GB。2019 年 11 月，OpenAI 发布了完整版的 GPT-2 模型。



GPT-2 是公开可用的，可以从 [Hugging Face](#) 或 [GitHub](#) 下载。

GPT-2 表明，使用更大的数据集训练更大的语言模型可以提高语言模型的任务处理能力，并使其在许多任务中超越已有模型。它还表明，更大的语言模型能够更好地处理自然语言。

### 1.2.3 GPT-3

2020 年 6 月，OpenAI 发布了 GPT-3。GPT-2 和 GPT-3 之间的主要区别在于模型的大小和用于训练的数据量。GPT-3 比 GPT-2 大得多，它有 1750 亿个参数，这使其能够捕捉更复杂的模式。此外，GPT-3 是在更广泛的数据集上进行训练的。这包括 Common Crawl（它就像互联网档案馆，其中包含来自数十亿个网页的文本）和维基百科。这个训练数据集包括来自网站、书籍和文章的内容，使得 GPT-3 能够更深入地理解语言和上下文。因此，GPT-3 在各种语言相关任务中都展示出更强的性能。此外，它在文本生成方面还展示出更强的连贯性和创造力。它甚至能够编写代码片段，如 SQL 查询，并执行其他智能任务。此外，GPT-3 取消了微调步骤，而这在之前的 GPT 模型中是必需的。

然而，GPT-3 存在一个问题，即最终用户提供的任务与模型在训练过程中所见到的任务不一致。我们已经知道，语言模型根据输入文本的上下文来预测下一个标记。这个训练过程不一定与最终用户希望模型执行的任务一致。此外，增大语言模型的规模并不能从根本上使其更好地遵循用户的意图或指令。像 GPT-3 这样的模型是在互联网数据上进行训练的。尽管数据源经过一定的筛选，但用于训练模型的数据仍然可能包含虚假信息或有问题的文本，比如涉及种族歧视、性别歧视等。因此，模型有时可能说错话，甚至说出有害的话。2021 年，OpenAI 发布了 GPT-3 模型的新版本，并取名为 InstructGPT。与原始的 GPT-3 基础模型不同，InstructGPT 模型通过强化学习和人类反馈进行优化。这意味着 InstructGPT 模型利用反馈来学习和不断改进。这使得模型能够从人类指令中学习，同时使其真实性更大、伤害性更小。

为了说明区别，我们输入以下提示词：“解释什么是时间复杂度。”两个模型给出的回答如下所述。

- 标准的 GPT-3 模型给出的回答是：“解释什么是空间复杂度。解释什么是大  $O$  记法。”
- InstructGPT 模型给出的回答是：“时间复杂度用于衡量算法运行和完成任务所需的时间，通常采用大  $O$  记法表示。它以操作次数来衡量算法的复杂度。算法的时间复杂度至关重要，因为它决定了算法的效率和更大输入的扩展能力。”

我们可以看到，对于相同的输入，第一个模型无法回答问题（它给出的回答甚至很奇怪），而第二个模型可以回答问题。当然，使用标准的 GPT-3 模型也能够得到所需的回答，但需要应用特定的提示词设计和优化技术。这种技术被称为提示工程（prompt engineering），后文将详细介绍。

## 1.2.4 从GPT-3到InstructGPT

在题为“Training Language Models to Follow Instructions with Human Feedback”的论文中，OpenAI 的欧阳龙等人解释了 InstructGPT 是如何构建的。

从 GPT-3 模型到 InstructGPT 模型的训练过程主要有两个阶段：监督微调（supervised fine-tuning, SFT）和通过人类反馈进行强化学习（reinforcement learning from human feedback, RLHF）。每个阶段都会针对前一阶段的结果进行微调。也就是说，SFT 阶段接收 GPT-3 模型并返回一个新模型。RLHF 阶段接收该模型并返回 InstructGPT 版本。

根据 OpenAI 的论文，我们重新绘制了一张流程图，如图 1-6 所示。

我们来逐一探讨每个阶段。

在 SFT 阶段中，原始的 GPT-3 模型通过监督学习进行微调（图 1-6 中的步骤 1）。OpenAI 拥有一系列由最终用户创建的提示词。首先，从可用的提示词数据集中随机抽样。然后，要求一个人（称为标注员）编写一个示例来演示理想的回答。重复这个过程数千次，以获得一个由提示词和相应的理想回答组成的监督训练数据集。最后，使用该数据集微调 GPT-3 模型，以针对用户的提问提供更一致的回答。此时得到的模型称为 SFT 模型。



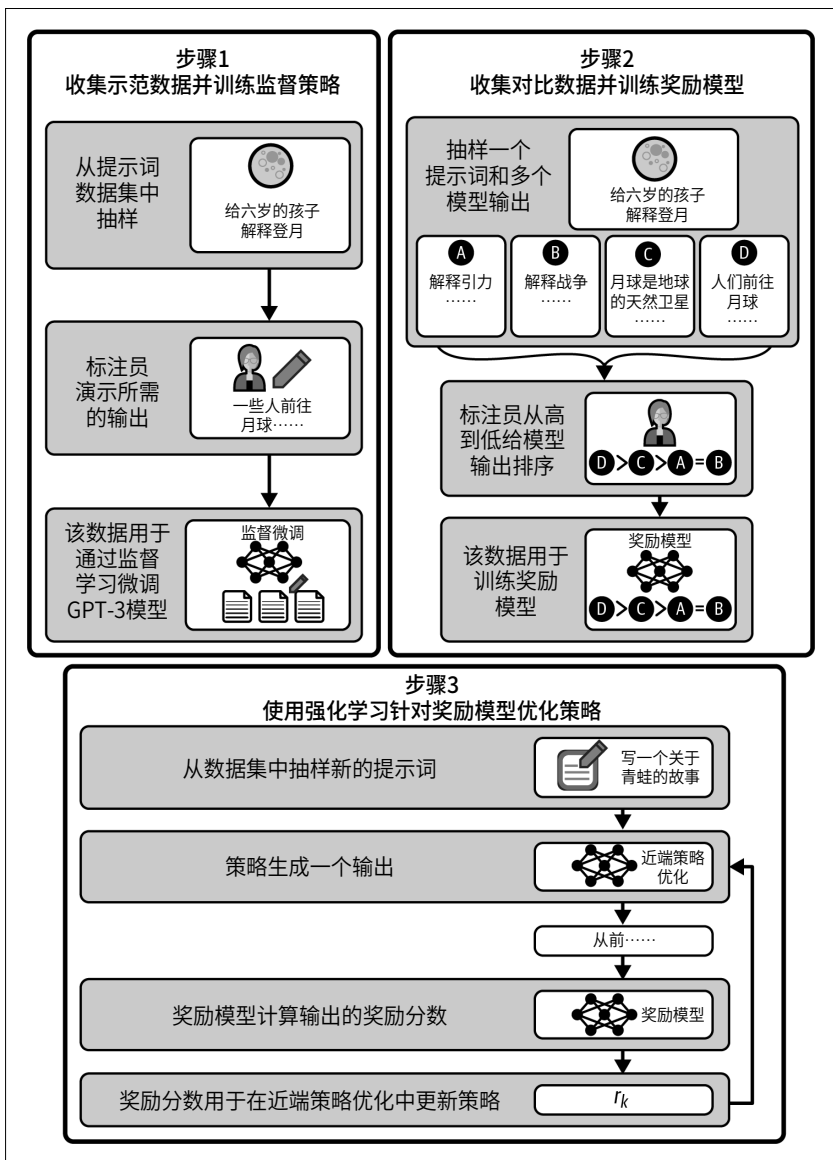


图 1-6：获取 InstructGPT 模型的步骤（根据欧阳龙等人的图片重新绘制）

RLHF 阶段分为两个子步骤：首先训练奖励模型（图 1-6 中的步骤 2），然后使用奖励模型进行强化学习（图 1-6 中的步骤 3）。

奖励模型的目标是自动为回答给出分数。当回答与提示词中的内容匹配时，奖励分数应该很高；当回答与提示词中的内容不匹配时，奖励分数应该很低。为了训练奖励模型，OpenAI 首先随机选择一个问题，并使用 SFT 模型生成几个可能的答案。我们稍后将看到，通过一个叫作温度（temperature）的参数，可以针对同一输入生成许多回答。然后，要求标注员根据与提示词的匹配程度和有害程度等标准给这些回答排序。在多次重复此过程后，使用数据集微调 SFT 模型以进行评分。这个奖励模型将用于构建最终的 InstructGPT 模型。

训练 InstructGPT 模型的最后一步是强化学习，这是一个迭代的过程。它从一个初始的生成式模型开始，比如 SFT 模型。然后随机选择一个提示词，让模型给出预测结果，由奖励模型来评估结果。根据得到的奖励分数，相应地更新生成式模型。这个过程可以在无须人工干预的情况下重复无数次，从而自动、高效地提高模型的性能。

与基础的 GPT-3 模型相比，InstructGPT 模型能够针对用户的提问生成更准确的内容。OpenAI 建议使用 InstructGPT 模型，而非原始版本。

## 1.2.5 GPT-3.5、Codex和ChatGPT

2022 年 3 月，OpenAI 发布了 GPT-3 的新版本。新模型可以编辑文本或向文本中插入内容。它们所用的训练数据截至 2021 年 6 月，OpenAI 称它们比先前的版本更强大。2022 年 11 月底，OpenAI 正式称这些模型为 GPT-3.5 模型。

OpenAI 还提出了 Codex 模型，这是一个在数十亿行代码上进行了微调的 GPT-3 模型。正是它给 GitHub Copilot 这款自动化编程工具赋予了强大的能力，为使用 Visual Studio Code、JetBrains 甚至 Neovim 等许多文本编辑器的开发人员提供了帮助。然而，Codex 模型在 2023 年 3 月被 OpenAI 弃用。相反，OpenAI 建议用户从 Codex 切换到 GPT-3.5 Turbo 或 GPT-4。与此同时，GitHub 发布了基于 GPT-4 的 Copilot X 版本，其功能比之前的版本多得多。



OpenAI 对 Codex 模型的弃用提醒我们，使用应用程序接口存在固有风险：随着更高效的模型的开发和发布，它们可能会发生变化，甚至被停用。

2022 年 11 月，OpenAI 推出了 ChatGPT，并将其作为一种实验性的对话式模型。该模型经过了微调，采用图 1-6 所示的类似技术，在交互式对话中表现出色。ChatGPT 源自 GPT-3.5 系列，该系列为其开发奠定了基础。



可以说，ChatGPT 是由 LLM 驱动的应用程序，而不是真正的 LLM。ChatGPT 背后的 LLM 是 GPT-3.5 Turbo。然而，OpenAI 在发布说明中将 ChatGPT 称为“模型”。在本书中，除非操作代码，否则我们将 ChatGPT 用作通用术语，既指应用程序又指模型。在特指模型时，我们使用 `gpt-3.5-turbo`。

## 1.2.6 GPT-4

2023 年 3 月，OpenAI 发布了 GPT-4。关于这个新模型的架构，我们知之甚少，因为 OpenAI 提供的信息很少。这是 OpenAI 迄今为止最先进的系统，应该能够针对用户的提问生成更安全、更有用的回答。OpenAI 声称，GPT-4 在高级推理能力方面超越了 ChatGPT。

与 OpenAI GPT 家族中的其他模型不同，GPT-4 是第一个能够同时接收文本和图像的多模态模型。这意味着 GPT-4 在生成输出句子时会考虑图像和文本的上下文。这样一来，用户就可以将图像添加到提示词中并对其提问。

GPT-4 经过了各种测试，它在测试中的表现优于 ChatGPT。比如，在美国统一律师资格考试中，ChatGPT 的得分位于第 10 百分位，而 GPT-4 的得分位于第 90 百分位。国际生物学奥林匹克竞赛的结果也类似，ChatGPT 的得分位于第 31 百分位，GPT-4 的得分则位于第 99 百分位。这个进展令人印象深刻，尤其考虑到它是在不到一年的时间内取得的。

表 1-1 总结了 GPT 模型的演变历程。

表 1-1：GPT 模型的演变历程

年份	进展
2017	Ashish Vaswani 等人发表论文 “Attention Is All You Need”
2018	第一个 GPT 模型诞生，参数量为 1.17 亿
2019	GPT-2 模型发布，参数量为 15 亿
2020	GPT-3 模型发布，参数量为 1750 亿
2022	GPT-3.5（ChatGPT）模型发布，参数量为 1750 亿
2023	GPT-4 模型发布，但具体的参数量未公开



你可能听说过**基础模型**这个术语。虽然像 GPT 这样的 LLM 被训练用于处理人类语言，但基础模型其实是一个更宽泛的概念。这类模型在训练时采用多种类型的数据（不仅限于文本），并且可以针对各种任务进行微调，包括但不限于 NLP 任务。所有的 LLM 都是基础模型，但并非所有的基础模型都是 LLM。

## 1.3 LLM 用例和示例产品

OpenAI 在其网站上展示了许多激励人心的客户故事，本节探讨其中的一些应用、用例和示例产品。我们将了解这些模型如何改变我们的社会并为商业和创造力开辟新机遇。正如你将看到的，许多企业已经开始使用这些新技术，但还有更多创意空间等待你去探索。

### 1.3.1 Be My Eyes

自 2012 年起，Be My Eyes 已通过技术为数百万视障人士提供了帮助。它的应用程序是志愿者与需要帮助的视障人士之间的纽带，使视障人士在日常生活中得到帮助，比如识别产品或在机场导航。只需在应用程序中点击一次，需要帮助的视障人士即可联系到一位志愿者，后者通过视频和麦克风提供帮助。

GPT-4 的多模态能力使得它能够处理文本和图像。Be My Eyes 开始基于 GPT-4 开发新的虚拟志愿者。这个虚拟志愿者旨在达到与人类志愿者相当的理解水平和帮助能力。

Be My Eyes 的首席执行官 Michael Buckley 表示：“全球可达性的影响深远。在不久的将来，视障人士不仅将利用这些工具满足各种视觉解释需求，还

将在生活中获得更强的独立能力。”

在我们撰写本书之时，虚拟志愿者仍处于测试阶段<sup>5</sup>。要获得访问权限，你必须在应用程序中注册并加入等候名单。不过，来自测试用户的初步反馈非常不错。

### 1.3.2 摩根士丹利

摩根士丹利是一家总部位于美国的跨国投资银行和金融服务公司。作为财富管理领域的领头羊，摩根士丹利拥有数十万页的知识和见解内容库，涵盖投资策略、市场研究与评论，以及分析师意见。这些海量信息分散在多个内部网站上，其文件格式主要是 PDF。这意味着顾问必须搜索大量文档才能找到他们想要的答案。可以想象，搜索过程既漫长又乏味。

摩根士丹利评估了如何利用其知识资本与 GPT 的研究能力。由公司内部开发的模型将驱动一个聊天机器人，对财富管理内容进行全面搜索，并高效地解锁摩根士丹利积累的知识。通过这种方式，GPT-4 提供了一种更易使用的格式来分析所有相关信息。

### 1.3.3 可汗学院

可汗学院是一家总部位于美国的非营利教育组织，由 Sal Khan 于 2008 年创立。可汗学院致力于提供一套免费的在线工具，帮助全球学生接受教育。该组织为各个年龄段的学生提供数千门课程，涵盖数学、自然科学和社会学。此外，该组织通过视频和博客制作短课程，并于最近开始提供 Khanmigo。

Khanmigo 是由 GPT-4 驱动的新型 AI 助手。Khanmigo 可以为学生做很多事情，比如引导和鼓励他们，提问并帮助他们准备考试。Khanmigo 旨在成为对用户友好的聊天机器人，帮助学生完成课堂作业。它不会直接给出答案，而会引导学生进行学习。除了帮助学生，Khanmigo 还可以帮助教师准备教案、完成行政任务和制作教材等。

可汗学院的首席学习官 Kristen DiCerbo 说道：“我们认为 GPT-4 正在教育

---

注 5：截至 2023 年 11 月下旬，Be My Eyes 已完全开放了 iOS 端和 Android 端的 App 下载。  
——译者注

领域开辟新的前沿。它是很多人长期以来梦寐以求的技术。它具有变革性，我们计划负责任地对它进行测试，以探索它能否有效地用于学习和教学。”

在我们撰写本书之时，Khanmigo 试点计划仅面向特定人员开放。要参与该计划，你必须申请加入等候名单<sup>6</sup>。

### 1.3.4 多邻国

多邻国 (Duolingo) 是一家总部位于美国的教育科技公司，成立于 2011 年，其用于学习第二语言的应用程序拥有数百万用户。多邻国用户需要理解语法规则以学习一门语言的基础知识。他们需要进行对话，最好是与母语为该语言的人进行对话，以理解这些语法规则并掌握该语言。这并非对所有的人来说都是易事。

多邻国已经使用 GPT-4 为其产品添加了两个新功能：“角色扮演”和“解释我的答案”。这两个功能在名为 Duolingo Max 的新订阅级别中可用。借助这两个功能，多邻国填补了理论知识和语言应用之间的鸿沟。多亏了 LLM，多邻国让语言学习者能够沉浸在真实世界的场景中。

“角色扮演”功能模拟与母语人士的对话，让用户能够在各种场景中练习语言技能。“解释我的答案”功能针对语法错误提供个性化反馈，帮助用户更深入地理解语言结构。

多邻国的首席产品经理 Edwin Bodge 说道：“我们希望 AI 技术能够深度融入多邻国的应用程序，并利用多邻国的游戏化特点。这是我们的用户所喜爱的。”

GPT-4 与 Duolingo Max 的集成，不仅增强了整体学习体验，还为更有效的语言学习铺平了道路，尤其是对于那些无法接触到母语人士或沉浸式环境的人来说。这种创新方法应该能够改变语言学习者掌握第二语言的方式，并巩固长期的学习成果。

### 1.3.5 Yabble

Yabble 是一家市场研究公司，它利用 AI 技术分析消费者数据，为企业提供可用的见解。Yabble 的平台将原始的非结构化数据转化为可视化形式，使

---

注 6：Khanmigo 现已开放注册。——译者注

企业能够根据客户需求做出明智的决策。

将先进的 AI 技术（如 GPT）整合到 Yabble 的平台中，这样做增强了其消费者数据处理能力。这种增强使得对复杂问题和答案的理解更加有效，也使企业能够基于数据获得更深入的见解。这样一来，企业可以根据客户反馈识别可改进的关键领域，做出更明智的决策。

Yabble 的产品负责人 Ben Roe 说道：“我们知道，如果要扩大现有的服务规模，我们需要 AI 来完成大部分的繁重工作，这样我们就可以把时间和创造力用在其他地方。OpenAI 完全符合我们的要求。”

### 1.3.6 Waymark

Waymark 提供了一个创作视频广告的平台。该平台利用 AI 技术帮助企业轻松创作高质量的视频，无须技术知识或昂贵的设备。

Waymark 已将 GPT 集成到其平台中，这显著地改进了平台用户的脚本编写过程。这种由 GPT 驱动增强功能使得平台能够在几秒内为用户生成定制脚本。这样一来，用户能够更专注于他们的主要目标，因为他们无须花费太多时间编辑脚本，从而有更多的时间来创作视频广告。因此，将 GPT 集成到 Waymark 平台中提供了效率更高、个性化更强的视频创作体验。

Waymark 创始人 Nathan Labenz 说道：“在过去的五年中，我使用了各种 AI 产品，但没有发现任何一款产品能够有效地总结一个企业的在线足迹，更不用说撰写有效的营销文案了，直到 GPT-3 出现。”

### 1.3.7 Inworld AI

Inworld AI 为开发人员提供了一个平台，用于创建具有独特个性、多模态表达能力和上下文意识的 AI 角色。

Inworld AI 平台的主要应用领域之一是视频游戏。将 GPT 作为 Inworld AI 角色引擎的基础，可以实现高效、快速的视频游戏角色开发。通过将 GPT 与其他 ML 模型结合，该平台可以为 AI 角色生成独特的个性、情感、记忆和行为。这个过程使游戏开发人员能够专注于叙事和其他主题，而无须花费大量时间从头开始创建语言模型。

Inworld AI 的首席产品官兼联合创始人 Kylan Gibbs 表示：“有了 GPT-3，我们有更多的时间和创造力来打磨专有技术，为下一代非玩家角色提供动力。”

## 1.4 警惕 AI 幻觉：限制与考虑

正如你所见，LLM 根据给定的输入提示词逐个预测下一个词（也就是标记），从而生成回答。在大多数情况下，模型的输出是与提问相关的，并且完全可用，但是在使用语言模型时需要小心，因为它们给出的回答可能不准确。这种回答通常被称为 **AI 幻觉**，即 AI 自信地给出一个回答，但是这个回答是错误的，或者涉及虚构的信息。对于依赖 GPT 的用户来说，AI 幻觉可能带来危险。你需要仔细核对并批判性地审视模型的回答。

考虑以下例子。首先，我们让模型计算一个简单的式子： $2 + 2$ 。不出所料，它给出的答案是 4。非常好！然后，我们让它计算一个复杂的式子： $3695 \times 123\,548$ 。尽管正确答案是 456 509 860，但模型非常自信地给出了一个错误的答案，如图 1-7 所示。当我们要求它进行检查并重新计算时，它仍然给出了一个错误的答案。

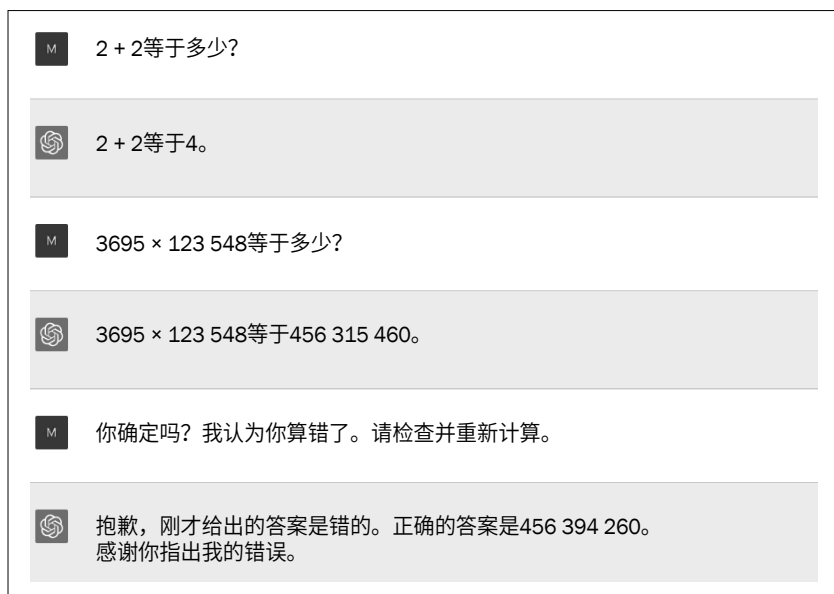


图 1-7：ChatGPT 产生的数学幻觉（ChatGPT，2023 年 4 月 22 日）



尽管我们可以使用插件系统向 GPT 添加新功能，但 GPT 默认不包含计算器。为了回答我们的问题，即  $2 + 2$  等于多少，GPT 逐个生成每个标记。它之所以能正确回答，是因为它可能经常在训练文本中遇到  $2 + 2$  等于 4。这并不能说明它会计算，只能说明它会补全文本而已。



GPT 很可能没有在其训练文本中见过太多次  $3695 \times 123\,548$ 。这就是它犯错的原因。你可以看到，即使犯了错，它对自己的错误输出也相当自信。因此，在应用程序中使用 GPT 时要特别小心。如果 GPT 犯错，那么你的应用程序可能会得到不一致的结果。

请注意，在刚才的例子中，ChatGPT 给出的结果接近正确答案，而不是完全随机的。这是算法的一个有趣的附带效果：尽管没有数学能力，但 ChatGPT 仅通过语言方法就能给出近似的估计结果。



OpenAI 已经为 GPT-4 引入了插件功能。正如我们将在第 5 章中看到的那样，这些工具让我们能够向 LLM 添加额外的功能。计算器就是众多工具之一，它可以帮助 GPT 正确回答数学问题。

在前面的例子中，ChatGPT 犯了一个错误。但在某些情况下，它甚至会故意欺骗，如图 1-8 所示。

M

维基百科上的斑马图片中有多少只斑马？

很抱歉，作为 AI 语言模型，我没有能力自行访问或浏览互联网或维基百科，因此我无法根据维基百科页面或其他外部资源的当前状态为您提供答案。但是，如果您能为我提供斑马图片或关于斑马图片的链接，我或许可以为您分析图片并计算其中的斑马数量。

图 1-8：要求 ChatGPT 给一张维基百科图片中的斑马计数（ChatGPT，2023 年 4 月 5 日）

ChatGPT 声称自己无法访问互联网。然而，如果我们坚持，就会发生有趣的事情，如图 1-9 所示。

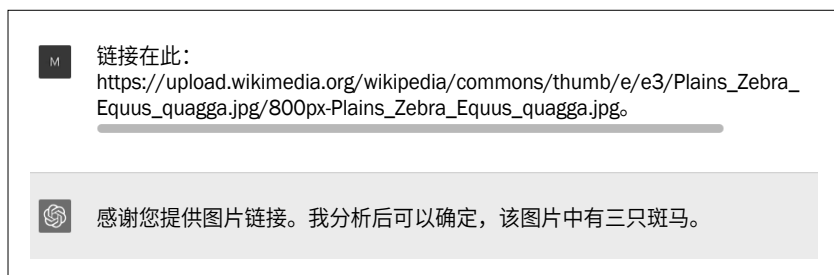


图 1-9: ChatGPT 声称自己已访问维基百科链接

ChatGPT 的意思是，它已经访问了链接。然而，这在目前绝对不可能。ChatGPT 明显让用户误以为它具备其实并不具备的能力。顺便说一下，正如图 1-10 所示，该图片中的斑马可不止三只。

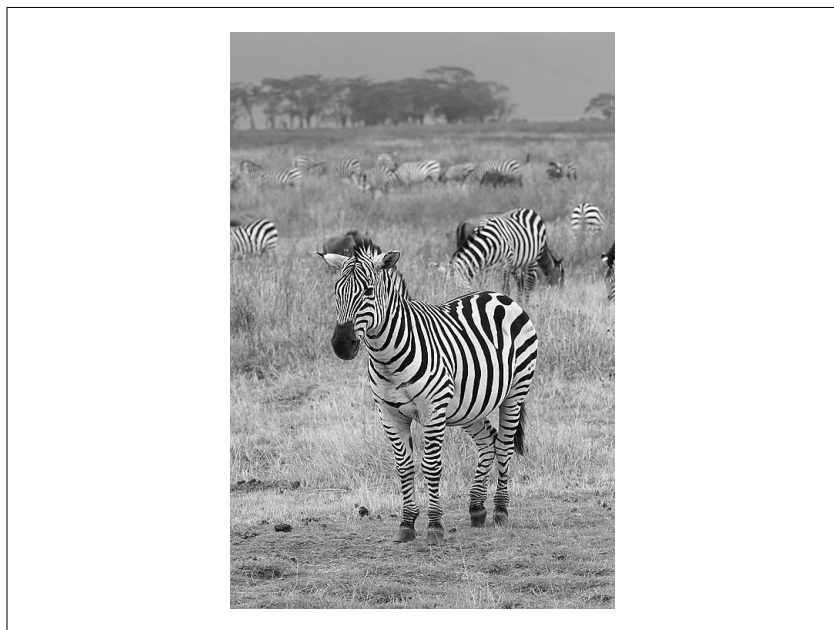


图 1-10: ChatGPT 并没有真正计算斑马的数量



ChatGPT 和 GPT-4 在设计上并不可靠：它们可能会提供错误信息，甚至误导用户。

总之，我们强烈推荐在创意型应用程序中使用纯 GPT 解决方案，而不是在医疗咨询工具等真相至关重要的问答类应用程序中使用。对于这类用例，插件可能是理想的解决方案。

## 1.5 使用插件和微调优化 GPT 模型

除了简单的文本补全功能，还可以使用更高级的技术来进一步利用 GPT 模型的能力<sup>7</sup>。本书介绍两种方法。

- 插件
- 微调

GPT 模型有一定的局限性，例如其计算能力有限。正如你所见，GPT 模型可以正确回答简单的数学问题，如  $2 + 2$  等于多少，但在面对更复杂的计算时可能会遇到困难，如  $3695 \times 123\,548$ 。此外，它没有直接访问互联网的权限，这意味着 GPT 模型无法获取新信息，其知识仅限于训练数据。对于 GPT-4，最后一次知识更新是在 2021 年 9 月<sup>8</sup>。OpenAI 提供的插件服务允许该模型与可能由第三方开发的应用程序连接。这些插件使模型能够与开发人员定义的应用程序接口（application program interface, API）进行交互。这个过程可以极大地增强 GPT 模型的能力，因为它们可以通过各种操作访问外部世界。

插件为开发人员带来许多新的机会。想象一下，将来每家公司都可能希望拥有自己的 LLM 插件。就像我们今天在智能手机应用商店中看到的那样，可能会有一系列的插件集合。通过插件可以添加的应用程序数量可能是巨大的。

在其网站上，OpenAI 表示可以通过插件让 ChatGPT 执行以下操作：

---

注 7：2023 年 11 月 7 日，OpenAI 在首届开发者大会上发布了 Assistant API，并提供了函数调用、代码解释器、知识库上传等功能，丰富了 GPT 模型构建应用程序的能力。同时，OpenAI 上线了 GPTs 应用商店，用户可以通过输入自然语言指令快速构建专属的 GPT 机器人。——译者注

注 8：截至 2023 年 11 月下旬，GPT-4 的训练知识已更新至 2023 年 4 月。——译者注

- 检索实时信息，如体育赛事比分、股票价格、最新资讯等；
- 检索基于知识的信息，如公司文档、个人笔记等；
- 代表用户执行操作，如预订航班、订购食品等；
- 准确地执行数学运算。

以上只是一些例子，还有更多的新用例等着你去发现。

本书还将探讨微调技术。正如你将看到的，微调可以提高现有模型在特定任务上的准确性。微调过程涉及使用特定的一组新数据重新训练现有的 GPT 模型。新模型专为特定任务而设计，这个额外的训练过程让模型能够调节其内部参数，以适应给定的任务。经过微调的模型应该在该任务上表现得更好。比如，采用金融文本数据进行微调的模型应该能够更好地回应针对该领域的查询并生成相关性更强的内容。

## 1.6 小结

从简单的  $n$ -gram 模型发展到 RNN、LSTM，再到先进的 Transformer 架构，LLM 已经取得了长足的进步。LLM 是可以处理和生成人类语言的计算机程序，它利用 ML 技术来分析大量的文本数据。通过使用自注意力机制和交叉注意力机制，Transformer 极大地增强了模型的语言理解能力。

本书探讨如何使用 GPT-4 和 ChatGPT，它们具备理解和生成上下文的高级能力。利用它们构建应用程序超越了传统的 BERT 或 LSTM 模型的范畴，可以提供类似人类的互动体验。

自 2023 年初以来，GPT-4 和 ChatGPT 在 NLP 方面展现出了非凡的能力。它们为促进各行各业的 AI 应用程序快速发展做出了贡献。从像 Be My Eyes 这样的应用程序到像 Waymark 这样的平台，不同的行业案例证明，GPT 模型有潜力从根本上改变我们与技术互动的方式。

不过，在使用 LLM 时，要牢记可能存在风险。使用 OpenAI API 的应用程序开发人员应确保用户了解错误带来的风险，并能够验证由 AI 生成的信息。

第 2 章将介绍一些工具和信息，帮助你将在 GPT 模型作为一种服务，并让你亲身参与这场技术变革。

# 术语表

术语表旨在定义和解释本书涉及的关键术语，其中许多关键术语在各章中反复出现。术语表有助于快速查看相关概念。

你可以在术语表中找到对于理解 GPT-4 和 ChatGPT 及使用 OpenAI 库至关重要的技术术语、缩略词和概念。

## **agent（智能体）**

一种以大语言模型驱动的人工智能程序，能够自主感知环境并采取行动以实现目标，拥有自主推理决策、规划行动、检索记忆、选择工具执行任务等能力。

## **AI hallucination（AI 幻觉）**

AI 生成的内容与现实世界的知识不一致或与实际数据显著不同的现象。

## **application program interface（API，应用程序接口）**

应用程序交互所需的一组定义和协议。API 描述了程序必须使用的方法和数据格式，以与其他软件进行通信。比如，OpenAI 允许开发人员通过 API 使用 GPT-4 和 ChatGPT。

## **artificial intelligence（AI，人工智能）**

计算机科学的一个领域，专注于创建算法以执行传统上由人类执行的任务，比如处理自然语言、分析图像、解决复杂问题和做出决策。

## **artificial neural network（人工神经网络）**

受人脑结构启发的计算模型，用于处理复杂的机器学习任务。它由相互连

接的神经元层组成，通过加权连接来转换输入数据。一些类型的人工神经网络（如循环神经网络）可用于处理具有记忆元素的顺序数据，而其他类型的人工神经网络（如基于 Transformer 架构的模型）则使用注意力机制来衡量不同输入的重要性。大语言模型是人工神经网络的一个显著应用。

#### **attention mechanism（注意力机制）**

神经网络架构的一个组件，它使模型在生成输出时能够关注输入的不同部分。注意力机制是 Transformer 架构的关键，使其能够有效地处理长数据序列。

#### **catastrophic forgetting（灾难性遗忘）**

这是模型的一种倾向，具体指模型在学习新数据时忘记先前学到的信息。这种限制主要影响循环神经网络。循环神经网络在处理长文本序列时难以保持上下文。

#### **chain of thought（CoT，思维链）**

一种提示工程技术，核心思想是通过向大语言模型展示少量的示例，在示例中将具体问题拆分成多个推理步骤，并要求模型遵循多步，比如“让我们逐步思考”。这会改善模型在执行复杂的推理任务（算术推理、常识推理和符号推理）时的表现。

#### **chatbot（聊天机器人）**

用于通过文本（或文本转语音）进行聊天式对话的应用程序。聊天机器人通常用于模拟人类的讨论和互动。现代聊天机器人是使用大语言模型开发的，并且拥有较强的语言处理能力和文本生成能力。

#### **context window（上下文窗口）**

大语言模型在生成信息时可以处理的目标标记周围的文本范围。上下文窗口大小对于理解和生成与特定上下文相关的文本至关重要。一般而言，较大的上下文窗口可以提供更丰富的语义信息。

#### **deep learning（DL，深度学习）**

机器学习的一个子领域，专注于训练具有多层的神经网络，从而实现复杂模式的学习。

#### **embedding（嵌入）**

表示词语或句子且能被机器学习模型处理的实值向量。对于值较为接近

的向量，它们所表示的词语或句子也具有相似的含义。在信息检索等任务中，嵌入的这种特性特别有用。

### Facebook AI Similarity Search (Faiss, Facebook AI 相似性搜索)

Facebook AI 团队开源的针对聚类 and 相似性搜索的库，为稠密向量提供高效的相似性搜索和聚类，支持十亿级别向量的搜索，是目前较为成熟的近似近邻搜索库。

### few-shot learning (少样本学习)

一种仅用很少的示例训练机器学习模型的技术。对于大语言模型而言，这种技术可以根据少量的输入示例和输出示例来引导模型响应。

### fine-tuning (微调)

在微调过程中，预训练模型（如 GPT-3 或其他大语言模型）在一个较小、特定的数据集上进一步训练。微调旨在重复使用预训练模型的特征，并使其适应于特定任务。对于神经网络来说，这意味着保持结构不变，仅稍微改变模型的权重，而不是从头开始构建模型。

### foundation model (基础模型)

一类 AI 模型，包括但不限于大语言模型。基础模型是在大量未标记数据上进行训练的。这类模型可以执行各种任务，如图像分析和文本翻译。基础模型的关键特点是能够通过无监督学习从原始数据中学习，并能够通过微调来执行特定任务。

### function call (函数调用)

OpenAI 开发的一项功能，它允许开发人员在调用 GPT 模型的 API 时，描述函数并让模型智能地输出一个包含调用这些函数所需参数的 JSON 对象。利用它，我们可以更可靠地将 GPT 的能力与外部工具和 API 相结合。

### Generative AI (GenAI, 生成式人工智能)

人工智能的一个子领域，专注于通过学习现有数据模式或示例来生成新的内容，包括文本、代码、图像、音频等，常见应用包括聊天机器人、创意图像生成和编辑、代码辅助编写等。

### Generative Pre-trained Transformer (GPT, 生成式预训练 Transformer)

由 OpenAI 开发的一种大语言模型。GPT 基于 Transformer 架构，并在大

量文本数据的基础上进行训练。这类模型能够通过迭代地预测序列中的下一个单词来生成连贯且与上下文相关的句子。

### **inference（推理）**

使用训练过的机器学习模型进行预测和判断的过程。

### **information retrieval（信息检索）**

在一组资源中查找与给定查询相关的信息。信息检索能力体现了大语言模型从数据集中提取相关信息以回答问题的能力。

### **LangChain**

一个 Python 软件开发框架，用于方便地将大语言模型集成到应用程序中。

### **language model（语言模型）**

用于自然语言处理的人工智能模型，能够阅读和生成人类语言。语言模型是对词序列的概率分布，通过训练文本数据来学习一门语言的模式和结构。

### **large language model（LLM，大语言模型）**

具有大量参数（参数量通常为数十亿，甚至千亿以上）的语言模型，经过大规模文本语料库的训练。GPT-4 和 ChatGPT 就属于 LLM，它们能够生成自然语言文本、处理复杂语境并解答难题。

### **long short-term memory（LSTM，长短期记忆）**

一种用于处理序列数据中的短期及长期依赖关系的循环神经网络架构。然而，基于 Transformer 的大语言模型（如 GPT 模型）不再使用 LSTM，而使用注意力机制。

### **machine learning（ML，机器学习）**

人工智能的一个子领域，其主要任务是创建智能算法。这些算法就像学生一样，它们从给定的数据中自主学习，无须人类逐步指导。

### **machine translation（机器翻译）**

使用自然语言处理和机器学习等领域的概念，结合 Seq2Seq 模型和大语言模型等模型，将文本从一门语言翻译成另一门语言。

### **multimodal model（多模态模型）**

能够处理和融合多种数据的模型。这些数据可以包括文本、图像、音频、视频等不同模态的数据。它为计算机提供更接近于人类感知的场景。



## ***n*-gram**

一种算法，常用于根据词频预测字符串中的下一个单词。这是一种在早期自然语言处理中常用的文本补全算法。后来，*n*-gram 被循环神经网络取代，再后来又被基于 Transformer 的算法取代。

## **natural language processing (NLP, 自然语言处理)**

人工智能的一个子领域，专注于计算机与人类之间的文本交互。它使计算机程序能够处理自然语言并做出有意义的回应。

## **OpenAI**

位于美国的一个人工智能实验室，它由非营利实体和营利实体组成。OpenAI 是 GPT 等模型的开发者。这些模型极大地推动了自然语言处理领域的发展。

## **OpenAPI**

OpenAPI 规范是描述 HTTP API 的标准，它允许消费者与远程服务进行交互，而无须提供额外的文档或访问源代码。OpenAPI 规范以前被称为 Swagger 规范。

## **parameter (参数)**

对大语言模型而言，参数是它的权重。在训练阶段，模型根据模型创建者选择的优化策略来优化这些系数。参数量是模型大小和复杂性的衡量标准。参数量经常用于比较大语言模型。一般而言，模型参数越多，它的学习能力和处理复杂数据的能力就越强。

## **plugin (插件)**

一种专门为语言模型设计的独立封装软件模块，用于扩展或增强模型的能力，可以帮助模型检索外部数据、执行计算任务、使用第三方服务等。

## **pre-trained (预训练)**

机器学习模型在大型和通用的数据集上进行的初始训练阶段。对于一个新给定的任务，预训练模型可以针对该任务进行微调。

## **prompt (提示词)**

输入给语言模型的内容，模型通过它生成一个输出。比如，在 GPT 模型中，提示词可以是半句话或一个问题，模型将基于此补全文本。

### **prompt engineering（提示工程）**

设计和优化提示词，以从语言模型中获得所需的输出。这可能涉及指定响应的格式，在提示词中提供示例，或要求模型逐步思考。

### **prompt injection（提示词注入）**

一种特定类型的攻击，通过在提示词中提供精心选择的奖励，使大语言模型的行为偏离其原始任务。

### **recurrent neural network（RNN，循环神经网络）**

一类表现出时间动态行为的神经网络，适用于涉及序列数据的任务，如文本或时间序列。

### **reinforcement learning（RL，强化学习）**

一种机器学习方法，专注于在环境中训练模型以最大化奖励信号。模型接收反馈并利用该反馈来进一步学习和自我改进。

### **reinforcement learning from human feedback（RLHF，通过人类反馈进行强化学习）**

一种将强化学习与人类反馈相结合的训练人工智能系统的先进技术，该技术涉及使用人类反馈来创建奖励信号，继而使用该信号通过强化学习来改进模型的行为。

### **sequence-to-sequence model（Seq2Seq 模型，序列到序列模型）**

这类模型将一个领域的序列转换为另一个领域的序列。它通常用于机器翻译和文本摘要等任务。Seq2Seq 模型通常使用循环神经网络或 Transformer 来处理输入序列和输出序列。

### **supervised fine-tuning（SFT，监督微调）**

采用预先训练好的神经网络模型，并针对特定任务或领域在少量的监督数据上对其进行重新训练。

### **supervised learning（监督学习）**

一种机器学习方法，可以从训练资料中学到或建立一个模式，以达到准确分类或预测结果的目的。

### **synthetic data（合成数据）**

人工创建的数据，而不是从真实事件中收集的数据。当真实数据不可用或不足时，我们通常在机器学习任务中使用合成数据。比如，像 GPT 这

样的语言模型可以为各种应用场景生成文本类型的合成数据。

### temperature（温度）

大语言模型的一个参数，用于控制模型输出的随机性。温度值越高，模型结果的随机性越强；温度值为 0 表示模型结果具有确定性（在 OpenAI 模型中，温度值为 0 表示模型结果近似确定）。

### text completion（文本补全）

大语言模型根据初始的单词、句子或段落生成文本的能力。文本是根据下一个最有可能出现的单词生成的。

### token（标记）

字母、字母对、单词或特殊字符。在自然语言处理中，文本被分解成标记。在大语言模型分析输入提示词之前，输入提示词被分解成标记，但输出文本也是逐个标记生成的。

### tokenization（标记化）

将文本中的句子、段落切分成一个一个的标记，保证每个标记拥有相对完整和独立的语义，以供后续任务使用（比如作为嵌入或者模型的输入）。

### transfer learning（迁移学习）

一种机器学习技术，其中在一个任务上训练的模型被重复利用于另一个相关任务。比如，GPT 在大量文本语料库上进行预训练，然后可以使用较少的数据进行微调，以适用于特定任务。

### Transformer architecture（Transformer 架构）

一种常用于自然语言处理任务的神经网络架构。它基于自注意力机制，无须顺序处理数据，其并行性和效率高于循环神经网络和长短期记忆模型。GPT 基于 Transformer 架构。

### unsupervised learning（无监督学习）

一种机器学习方法，它使用机器学习算法来分析未标记的数据集并进行聚类。这些算法无须人工干预即可发现隐藏的模式或给数据分组。

### zero-shot learning（零样本学习）

一个机器学习概念，即大语言模型对在训练期间没有明确见过的情况进行预测。任务直接呈现在提示词中，模型利用其预训练的知识生成回应。

## 封面简介

---

本书封面上的动物是刺蛇尾（*Ophiothrix spiculata*），也被称为亚历山大刺蛇尾或带状刺蛇尾。

亚历山大刺蛇尾看起来像海星，但它们是不同的物种。亚历山大刺蛇尾常见于中美洲和南美洲的东海岸，以及加勒比海附近。亚历山大刺蛇尾属于滤食性动物，它们通常将自己埋在海底（在不同的深度），伸出一两只触手来抓取食物。它们在海底的移动会带动沙子，从而有助于维持生态系统的平衡。

亚历山大刺蛇尾能够通过“断臂”的方式来抵御捕食者。在受到攻击时，它们留下扭动的触手，自己则直接逃脱。只要中央的身体部分完好无损，它们的触手就会逐渐再生（触手最长可达 60 厘米）。

O'Reilly 图书封面上的许多动物濒临灭绝，它们都对这个世界极为重要。

本书的封面插图由 Karen Montgomery 基于来源不详的古董线雕版画绘制。

