

大模型学习科研手册

主编：严晓东 刘言哲 王红霓

Title of Book: Large Language Model Handbook for Study | 大模型学习科研手册

Author: Xiaodong Yan, Yanzhe Liu, Hongni Wang | 严晓东 刘言哲 王红霓

ISBN: 979-8-88892-665-9

List Price: US \$6.50 CN ¥45.00

Copyright © 2025 by Overseas Chinese Press Inc | Imprint: Overseas Chinese Press

Detailed Subjects: Education; Publication Date: February 2025.

大模型学习科研手册

作者: 严晓东 刘言哲 王红霓

责任编辑: 田绪清

封面设计: 安迪

出版发行: 华侨出版社 (Overseas Chinese Press)

地址: 纽约州奥尔巴尼市州街 90 号 700 单元 40 室

网址: <https://www.oc-press.com>

邮箱: oc-press@foxmail.com

规格: 787×1092 毫米, 16 开, 15 印张

版次: 2025 年 2 月第 1 版

印次: 2025 年 2 月第 1 次印刷

书号: ISBN 979-8-88892-665-9

定价: US \$6.50 CN ¥45.00

版权所有, 请勿翻印。如有装订错误, 请与我们联系更换。

总序

在大数据时代的推动下，数据已经转变为新的生产要素，其重要性不言而喻。目前，众多大型科技企业，如阿里巴巴、京东和百度等，都已设立数据科学家等相关职位。在学术界，我国众多高校都开设了数据科学与大数据技术专业，该专业是在大数据技术兴起后应运而生的战略性新兴产业。数据科学的发展不仅深刻影响了统计学、经济学等传统学科的研究方向，也对科技、金融等行业的业务模式产生了重大影响。

为响应国家“数字中国”建设战略发展需要，针对统计学、数据科学与大数据技术等专业教育的需求，我们成立了“应用型数据科学系列教材编委会”，并组织了一支涵盖数据科学技术相关课程的编写团队。数据科学作为一门交叉学科，融合了数学、统计学、计算机科学等多个领域的理论和方法。要想成为一名合格的数据科学家，不仅需要掌握基本的统计学、计算机科学和数据分析软件等知识，还需要与时俱进，不断学习新的数据科学技术和方法。因此，为学习者提供清晰、逻辑性强、易于理解的数据科学学习资料，成为了撰写本系列教材的初衷和目标。本系列教材采用数据科学的统计语言，详细介绍数据分析流程，通过直观的文字解释和便捷的程序实现，为读者提供愉悦的学习体验，并增强普通高等教育理工类和经济管理类高年级本科生和研究生的基础知识储备。

序言

在这个由数据和算法定义的时代，人工智能正以前所未有的速度和规模重塑我们对世界的认知。作为人工智能领域的一项重要突破，大模型以其强大的数据处理能力和深度学习能力，正在引领着科研和技术开发的新浪潮。《大模型学习科研手册》便是在这样的背景下诞生的，旨在为那些渴望探索人工智能深度和广度的学者、研究人员以及爱好者们提供一份全面的指南。

本书的创作过程是一段漫长而充满挑战的旅程。它不仅仅是文字的堆砌，更是我多年思考和经验的结晶。在这本书中，我尝试着将大模型的复杂概念和应用技术转化为易于理解的语言，希望能够激发读者的好奇心，引导他们去探索、去发现，甚至去改变。

大模型不仅仅是一种技术工具，更是一种能够推动科学进步、促进知识创新的力量。它们在科研领域的应用潜力是巨大的，从数据分析到程序编写，从对话交流到科学研究，大模型正在改变我们解决问题的方式。本书将深入探讨大模型的这些应用，并提供实际的指导和案例，帮助读者理解如何将这些强大的工具应用于自己的研究和工作中。

在撰写这本书的过程中，我深感责任重大。我希望《大模型学习科研手册》不仅能够成为读者学习大模型的良好帮手，更能够激发读者的创新思维，推动他们在科研道路上不断前行。

我深知，任何一本書籍都不可能涵盖所有知识，也不可能解决所有问题。但我相信，通过这本书，读者能够获得一种全新的视角，一种探索未知的勇气，以及一种解决问题的能力。我鼓励每一位读者在阅读本书的同时，不断实践、不断思考、不断创新。

最后，我要感谢所有在本书编写过程中给予我支持和帮助的人。没有你们，这本书不可能完成。我也希望读者能够享受阅读这本书的过程，并从中获得知识和灵感。

愿《大模型学习科研手册》能够成为您科研旅途中的一盏明灯，照亮您前行的道路。

严晓东

2024年06月06日

内容提要

在《大模型学习科研手册》这本书中，我们首先在第一章深入探讨了大模型的发展历程，从基础架构到工作原理，为读者打下坚实的理论基础，为后续的深入学习做好铺垫。紧接着，在第二章，我们将重点介绍如何与大模型进行有效的交流和互动，学习如何提出问题、理解模型的回答，并通过对话深化对问题的理解。第三章则指导读者如何利用大模型来辅助编程工作，从代码生成到 bug 修复，展现大模型在软件开发中的革命性应用。第四章聚焦于数据分析，科研工作的核心环节，展示大模型在处理和大规模数据集时的强大能力，以及它们如何帮助我们发现数据背后的模式和关联。在第五章，我们将看到大模型如何成为科研人员的得力助手，从文献检索到实验设计，再到结果分析，不断探索未知的过程。最后，在第六章，我们探索大模型在图像识别、处理和生成方面的应用，以及它们如何帮助我们更好地理解 and 创造视觉内容，图像作为信息的重要载体，大模型的应用将为我们的视觉世界带来新的视角和可能。第七章介绍了大模型项目标书，主要用于向资助机构、政府部门、学术团体或其他组织申请资金、资源或批准以开展特定的项目或研究工作。这本书旨在为读者提供一个全面的视角，了解大模型在科研和技术开发中的广泛应用和深远影响。

目录

第 1 章 大模型介绍	01
1.1 大模型原理简介	02
1.1.1 文生文	07
1.1.2 文生图	11
1.1.3 文生视频	22
1.2 大模型产品简介	27
1.2.1 国内大模型	27
1.2.2 国外大模型	39
1.2.3 大模型前沿问题	67
第 2 章 大模型对话交流	83
2.1 大模型和搜索引擎的区别	84
2.2 大模型的提示词使用技巧	87
2.2.1 基础技巧	87
2.2.2 LLM 的 prompt 提示词基础使用技巧	91
2.3 大模型的提示词高级技巧	93
2.4 大模型对话应用示例	106
2.4.1 提出建议	111
2.4.2 生成和显示数学工具	113
2.4.3 对文章中数据进行整理	114

2.4.4 利用大模型进行用户评论分类	116
2.4.5 利用大模型写报告	116
2.4.6 利用大模型生成测试题	118
第 3 章 大模型编写程序	120
3.1 大模型实现编程功能	121
3.2 大模型解释代码	132
3.3 大模型代码纠错	148
第 4 章 大模型数据分析	155
4.1 使用 AI 进行简单的数据分析	156
4.1.1 简介	156
4.1.2 步骤	156
4.2 使用 AI 完成数学建模项目	165
4.2.1 数据预处理	166
4.2.2 数据建模	173
第 5 章 大模型科学研究	181
5.1 大模型论文搜索	182
5.1.1 大模型的原理	182
5.1.2 大语言模型的局限性	183
5.1.3 大语言模型的应用场景总结	184
5.1.4 论文搜索的方法	184
5.2 大模型论文阅读辅助	185
5.2.1 Retrieval-Augmented Generation (RAG) 相关介绍	185
5.2.2 ChatDoc 使用教程	186

5.2.3 Kimi 使用教程	188
5.3 大模型论文书写辅助	190
5.3.1 利用 AI 辅助“人工智能”论文写作	191
5.3.2 相关软件与工具	203
第 6 章 大模型图片应用	212
6.1 图片信息提取	213
6.1.1 OCR	213
6.1.2 统计图表识别	213
6.1.3 图片转文字	214
6.2 AI 绘图	215
6.2.1 文生图的操作方法	215
6.2.2 文生图的微调方法	219
第 7 章 大模型项目标书	223
7.1 项目标书的结构	224
7.2 如何用大模型润色项目标书	226
7.2.1 项目名称	226
7.2.2 申请代码和关键词	226
7.2.3 中英文摘要	226
7.2.4 立项依据、国内外研究现状、参考文献	227
7.2.5 研究目标、关键科学问题、研究内容	228
7.2.6 研究方案、可行性分析	229
7.2.7 研究特色与创新	229
7.2.8 年度研究计划、预期研究成果	230
7.2.9 总结	230



CHAPTER 1

第 1 章 大模型介绍

1.1 大模型原理简介

大模型，从字面上理解指的是具有数千万甚至数亿参数的深度学习模型；而 ChatGPT 对大模型的解释是：大模型本质上是一个使用海量数据训练而成的深度神经网络模型，其巨大的数据和参数规模，实现了智能的涌现，展现出类似人类的智能。这其中的典型代表就是当下人们耳熟能详的大语言模型（Large Language Model, LLM），它是深度学习领域的一种突破性技术，主要通过处理海量数据来学习语言的复杂模式和结构。这些模型的核心在于其庞大的参数数量，其量级往往达到数百亿，甚至更多，这使得它们能够捕捉到语言的细微差别，并在多种任务上表现出色。大模型不仅被应用于自然语言处理，它们也被应用于图像、视频生成等领域，如 Midjourney 的图像生成模型和 OpenAI 的 Sora 视频生成模型。

大语言模型，通常指的是使用深度学习技术训练出来的，能够处理和生成自然语言文本的复杂性系统。这些模型通常基于变换器（Transformer）架构，该架构由注意力机制（Attention Mechanism）驱动，能够捕捉长距离依赖关系。以下是大语言模型的一些关键原理和重要组成部分：

- 变换器架构（Transformer Architecture）：Transformer 模型由 Vaswani 等人在 2017 年提出，它摒弃了传统的循环神经网络（RNN）结构，转而使用自注意力机制来处理序列数据。

这种结构使得模型能够并行处理整个序列，大大提高了训练效率。一个基本的 Transformer 架构如图 1.1 所示。它主要由输入部分（输入输出嵌入与位置编码）、多层编码器、多层解码器以及输出部分（输出线性层与 Softmax）四大部分组成。

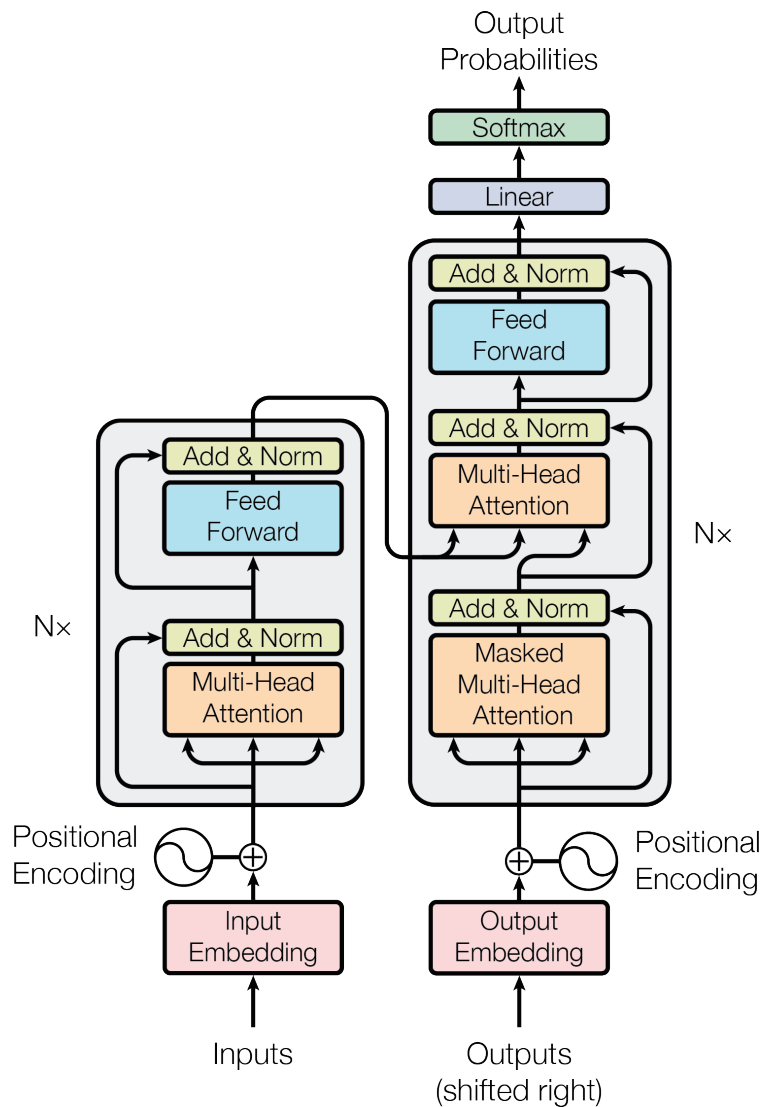


图 1.1: Transformer 架构

- **自注意力机制 (Self-Attention)**：自注意力机制允许模型在处理序列中的每个元素时，考虑序列中所有元素的信息。这使得模型能够捕捉到序列内部的复杂关系，如词语之间的依赖关系。其本质是将查询和一组键值对映射到输出，其中查询、键、值和输出都是向量。输出是作为值的加权和计算的，其中分配给每个值的权重是由查询与相应键的兼容性函数计算的。

- **编码器 - 解码器架构 (Encoder-Decoder Architecture)**：在某些任务中，如机器翻译，大语言模型会使用编码器 - 解码器架构。编码器处理输入序列，解码器生成输出序列。编码器和解码器都使用自注意力机制，并且可以通过层叠来增加模型的深度。

- **多头注意力 (Multi-Head Attention)**：Transformer 模型中的多头注意力允许模型同时从不同的角度处理输入数据，这有助于捕捉不同层次的语言特征，具体见图 1.2。

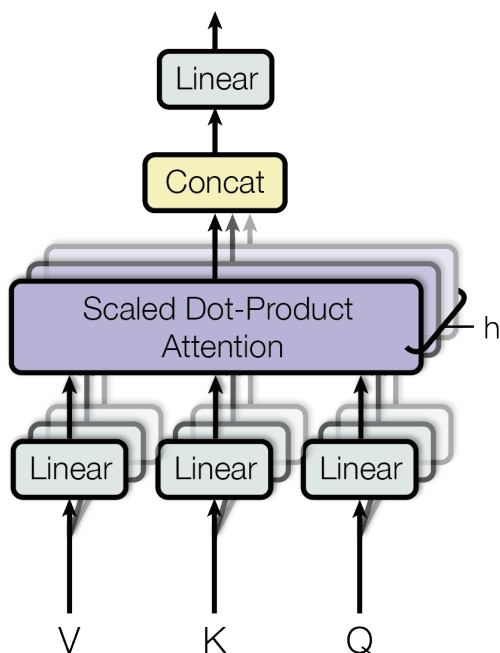


图 1.2: 多头注意力

- 位置编码 (Positional Encoding)：由于 Transformer 模型本身不具备捕捉序列中元素位置的能力，因此需要引入位置编码来提供序列中单词的位置信息。

- 前馈网络 (Feed-Forward Networks)：在自注意力层之后，模型通常会使用前馈网络来进一步处理数据，增加模型的表达能力。

- 优化和正则化技术：为了训练这些模型，需要使用高效的优化算法（如 Adam）和正则化技术（如 Dropout）来防止过拟合。

- 预训练和微调 (Pre-training and Fine-tuning)：大语言模型通常首先在大量数据上进行预训练，学习通用的语言表示。然后，它们可以在特定任务上进行微调，以适应特定的应用场景。

- 大规模数据和计算资源：训练这些模型需要大量的数据和计算资源。随着模型规模的增加，它们能够处理更复杂的语言任务，但同时也需要更多的数据和计算能力。

- 持续学习和知识更新：为了保持模型的时效性和准确性，需要定期对模型进行更新和再训练，以纳入最新的语言使用趋势和知识。

根据图 1.1 展示的 Transformer 架构进行变形，主流的框架大致可以分为 Encoder-Decoder, Encoder-Only 和 Decoder-Only 三大类，其中：

- 仅包含编码器部分 (Encoder-Only)：主要适用于不需要生成序列的任务，只需要对输入进行编码和处理的单向任务场景，如文本分类、情感分析等，这类代表是 BERT 相关的模型，例如 BERT, RoBERT, ALBERT 等。

- 既包含编码器也包含解码器 (Encoder-Decoder)：通常用于序列到序列 (Seq2Seq) 任务，如机器翻译、对话生成等，这类代表是以 Google 训出来 T5 为代表相关大模型。

- 仅包含解码器部分 (Decoder-Only)：通常用于序列生成任务，如文本生成、机器翻译等。这类结构的模型适用于需要生成序列的任务，可以从输入的编码中生成相应的序列。同时还有一个重要特点是可以进行无监督预训练。在预训练阶段，模型通过大量的无标注数据学习语言的统计模式和语义信息。这种方法可以使得模型具备广泛的语言知识和理解能力。在预训练之后，模型可以进行有监督微调，用于特定的下游任务（如机器翻译、文本生成等）。这类结构的代表也就是我们平时非常熟悉的 GPT 模型的结构，所有该家族的网络结构都是基于 Decoder-Only 的形式来逐步演化。

初步认识大模型的重要组成部分之后，接下来我们需要了解如何训练出一个大模型。训练方式，这里主要参考 OpenAI 发表的关于 InstructGPT 的相关训练步骤（见图 1.3），主流的大模型训练基本形式大多也是类似的：

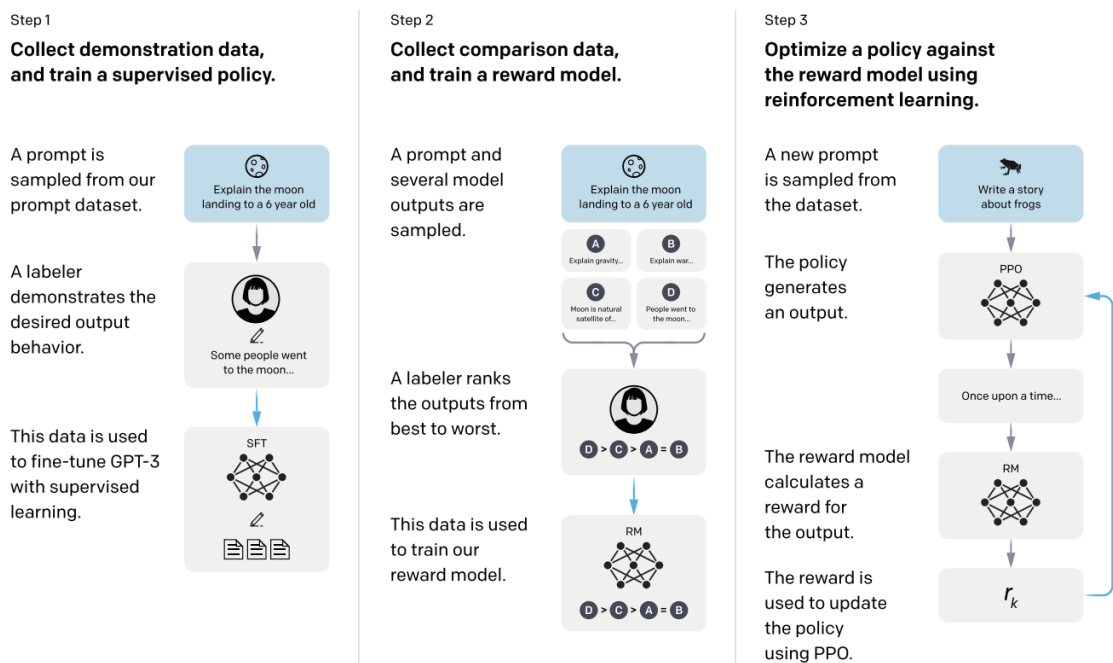


图 1.3: 训练方法

(1) 预训练 (Pretraining)

预训练是大模型训练的第一步，目的是让模型学习语言的统计模式和语义信息。主流的预训练阶段步骤基本都是近似的，其中最重要的就是数据，需要收集大量的无标注数据，例如互联网上的文本、新闻、博客、论坛等等。这些数据可以是多种语言的，并且需要经过一定的清洗和处理，以去除噪音，无关信息以及个人隐私相关的，最后会以 tokenizer 粒度输入到上文提

到的语言模型中。这些数据经过清洗和处理后，用于训练和优化语言模型。预训练过程中，模型会学习词汇、句法和语义的规律，以及上下文之间的关系。OpenAI 的 ChatGPT4 能有如此惊人的效果，主要的一个原因就是他们训练数据源比较优质。

(2) 指令微调阶段 (Instruction Tuning Stage)

在完成预训练后，就可以通过指令微调去挖掘和增强语言模型本身具备的能力，这步也是很多企业以及科研研究人员利用大模型的重要步骤。

Instruction tuning (指令微调) 是大模型训练的一个阶段，它是一种有监督微调的特殊形式，旨在让模型理解和遵循人类指令。在指令微调阶段，首先需要准备一系列的 NLP 任务，并将每个任务转化为指令形式，其中指令包括人类对模型应该执行的任务描述和期望的输出结果。然后，使用这些指令对已经预训练好的大语言模型进行监督学习，使得模型通过学习和适应指令来提高其在特定任务上的表现。

为了让模型训练更加高效和简单，这个阶段还有一种高效的 fine-tuning 技术，这为普通的从业者打开了通向使用大模型的捷径。

Parameter-Efficient Fine-Tuning (PEFT) 旨在通过最小化微调参数的数量和计算复杂度，达到高效的迁移学习的目的，提高预训练模型在新任务上的性能，从而缓解大型预训练模型的训练成本。在训练过程中，预训练模型的参数保持不变，只需微调少量的额外参数，就可以达到与全量微调相当的性能。

目前，很多研究对 PEFT 方法进行了探索，例如 Adapter Tuning 和 Prefix Tuning 等。其中，Adapter Tuning 方法在面对特定的下游任务时，将预训练模型中的某些层固定，只微调接近下游任务的几层参数。而 Prefix Tuning 方法则是在预训练模型的基础上，添加一些额外的参数，这些参数在训练过程中会根据特定的任务进行更新和调整。

(3) 对齐微调 (Alignment Tuning)

主要目标在于将语言模型与人类的偏好、价值观进行对齐，其中最重要的技术就是使用 RLHF (reinforcement learning from human feedback) 来进行对齐微调。先收集一个提示词集合，并要求标注人员写出高质量的回复，然后使用该数据集以监督的方式微调预训练的基础模型。这个过程涉及到与人类评估者进行对话，并根据他们的反馈来进行调整和优化。评估者会根据个人偏好对模型生成的回复进行排序，从而指导模型生成更符合人类期望的回复。这种基于人类反馈的训练方式可以帮助模型捕捉到更多人类语言的特点和习惯，从而提升模型的生成能力。在这一过程中主要使用强化学习的邻近策略优化 (PPO, proximal policy optimization) 算法，对于每个时间步，PPO 算法会计算当前产生和初始化的 KL 散度，根据这个分布来计算一个状态

或动作的预期回报，然后使用这个回报来更新策略，达到对 SFT 模型进一步优化。

在对大模型的一些简单原理有了初步认识以后，我们更需要认识到大模型，尤其是大语言模型在自然语言处理领域有着广泛的应用，包括语言翻译、文本摘要、情感分析、问答系统等。随着技术的不断进步，大模型在各个领域的应用也在不断扩展和深化，为人工智能的发展开辟了新的广阔天地与新的可能性。

1.1.1 文生文

文生文 (Text-to-Text) 是一种基于深度学习的自然语言处理 (NLP) 技术，旨在生成符合上下文的连贯文本。这类模型通常使用大量文本数据进行训练，能够理解和生成自然语言。文生文大模型通常基于 Transformer 架构，如 GPT(Generative Pre-trained Transformer) 系列、T5(Text-to-Text Transfer Transformer) 等。这些模型通过堆叠多层注意力机制和全连接层，捕捉文本中的复杂模式和上下文关系。下面我们就以 T5 架构为例介绍文生文技术。

T5 全称 Text-to-Text Transfer Transformer，是一种模型架构或者说是一种解决 NLP 任务的范式。传统的 NLP 模型通常专注于特定任务（如分类、翻译、摘要等），并针对每个任务进行专门的模型设计和训练。T5 的设计初衷是简化这一过程，通过统一的框架处理所有文本任务，将其转换为“文本输入到文本输出”的格式，从而实现模型和任务之间的通用性。它把所有任务，如分类、相似度计算、文本生成，统一到一个 Text-to-Text 的框架里去解决，如下图所示。

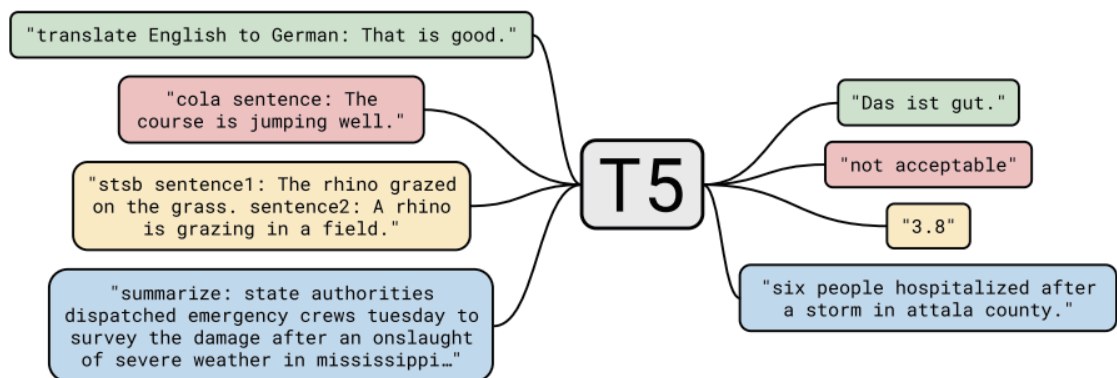


Figure 1: A diagram of our text-to-text framework. Every task we consider—including translation, question answering, and classification—is cast as feeding our model text as input and training it to generate some target text. This allows us to use the same model, loss function, hyperparameters, etc. across our diverse set of tasks. It also provides a standard testbed for the methods included in our empirical survey. “T5” refers to our model, which we dub the “Text-to-Text Transfer Transformer”.

图 1.4: 统一框架处理所有任务

比如，现在要做一个英文转德文的机器翻译任务，输入“translate English to German: That is good.”，目标输出是“Das ist gut.”。再比如，现在做一个文本三分类任务，label 为【entailment、contradiction、neutral】，输入“mnli premise: I hate pigeons. hypothesis: My feelings towards pigeons are filled with animosity.”，目标输出是“entailment”。这样我们把所有的问题都套进一个统一的范式，从而可以采用同样的模型架构、同样的训练策略、同样的损失函数、同样的解码手段。

T5 模型基于 Transformer 架构。首先，与传统的 Transformer 相似，T5 使用编码器 - 解码器架构。编码器将输入文本转换为隐含表示，解码器根据隐含表示生成输出文本。其次，所有任务（如文本分类、翻译、问答等）都被转换为文本到文本的格式。

预训练和微调是现代自然语言处理（NLP）模型取得成功的两个关键阶段。预训练的目标是通过大量未标注的文本数据，学习语言的基本结构和语义关系。T5 模型在多种大规模数据集上进行预训练，这些数据集包括通用的网络文本、书籍、新闻等。在预训练前，数据被处理成统一的文本格式，并进行适当的标记和清洗。T5 模型的预训练任务是填空任务 (Span Corruption)，即随机选择输入文本中的一个或多个片段 (spans) 进行掩码，模型需要根据上下文生成被掩码的部分，从而学习如何在不同上下文中生成合理的文本。其中，掩码的片段长度和位置是随机的，以增强模型的泛化能力。这一任务能够使模型理解文本的上下文和语义。在训练过程中，模型使用自监督学习方法，在没有标注的情况下进行训练。采用交叉熵损失函数，比较模型生成的文本与原始文本的差异，调整模型参数。此外，在预训练阶段，模型同时处理多种不同类型的填空任务，从而增强模型在不同任务上的泛化能力。微调的目标是在特定的下游任务上优化模型，使其在该任务上表现更好。由于 T5 模型采用统一的文本到文本框架，所有任务都可以被转换为文本生成任务。微调阶段使用标注数据，这些数据包含了任务的输入和预期输出。当然，数据集可能需要进行适当的处理和格式化，以符合 T5 模型的输入要求。训练过程使用监督学习方法，模型在标注数据上进行训练。采用交叉熵损失函数，比较模型生成的输出与标注的预期输出，调整模型参数。在微调过程中，可以使用梯度累积和混合精度训练等技术，以提高训练效率和稳定性。

在模型规模上，T5 模型有五个主要的规模变体，每个变体的参数数量和计算需求逐步增加。这些变体包括 T5-Small、T5-Base、T5-Large、T5-3B 和 T5-11B，参数数量从 6000 万到 110 亿不等。

T5 模型在多项标准自然语言处理（NLP）任务基准上展示了出色的性能，包括但不限于 GLUE、SuperGLUE、SQuAD 和 CNN/Daily Mail 等。其中，GLUE 是一个用于评估模型在广泛语言理解任务上的表现的基准，它包含多个子任务，如 CoLA（语言接受性）、SST-2（情感分类）、MRPC（句子对比）、STS-B（语义文本相似度）、QQP（问答对比）、MNLI（多项选择问答）、

QNLI（问答自然语言推理）、RTE（识别文本蕴含）、WNLI（Winograd 模式挑战）；SuperGLUE 包含的任务更为复杂，包括布尔问题问答、文本蕴含、共指消解、多项选择问答等；SQuAD 是一个广泛使用的机器阅读理解基准，任务是根据给定的上下文回答问题；而 CNN/Daily Mail 是一个用于自动文本摘要的基准，任务是根据给定的新闻文章生成简要摘要。除了上述基准任务，T5 模型还在多个其他 NLP 任务上进行了评估，如 WMT（机器翻译）、CoQA（对话问答）、XSum（极端摘要）等。T5 模型通过在多个标准 NLP 基准任务上的实验，展示了其强大的文本理解和生成能力。特别是在大规模模型（如 T5-11B）上，T5 在多个任务上达到了或超过了当时的最先进水平。其统一的文本到文本框架，使得 T5 能够灵活地处理各种不同类型的任务，展示了极高的通用性和灵活性。这些实验结果证明了 T5 模型在广泛的 NLP 应用中的有效性和强大性能。

T5 模型的应用场景主要有：

- **文本分类**：文本分类任务涉及将给定的文本分类到预定义的类别中。例如情感分析，通过分析文本中的情感倾向（如积极、消极、中立），T5 可以用于社交媒体监控、市场情报和客户反馈分析；还可以进行主题分类，即将新闻文章、博客帖子或其他文本分类到不同的主题中，如体育、科技、娱乐等，适用于内容推荐和信息检索。

- **机器翻译**：T5 可以用于各种语言对之间的翻译任务，如英语到法语、中文到英语等，广泛应用于跨语言交流、国际业务和多语言内容创建。

- **问答系统**：问答系统任务涉及根据给定的上下文回答问题。例如，T5 可以在大规模文本数据（如维基百科）上回答用户提出的各种问题，应用于虚拟助手、搜索引擎和智能客服；此外，在医疗、法律、技术等特定领域，T5 可以根据特定领域的文档和数据回答相关问题，辅助专业人员工作。

- **文本生成**：例如新闻生成、对话系统、文档摘要。

- **文本校对与改写**：涉及对文本进行校对、改写或重述，以提高其质量或适应特定需求。例如，T5 可以用于自动化的语法检查和错误修正，辅助用户创建更准确的文本。此外还可以应用于风格转换，将文本从一种写作风格转换为另一种风格，如正式到非正式、现代到古典等，适用于内容适配和文案创作。

- **信息抽取**：从非结构化文本中提取结构化信息。例如实体识别，即识别文本中的命名实体（如人名、地名、组织名等），应用于知识图谱构建和信息检索。此外，还可以用于关系抽取，从文本中提取实体之间的关系，如公司并购、人物关系等，应用于数据挖掘和情报分析。

- **数据增强**：生成新的数据样本以增强训练数据集的多样性和规模。一是数据合成，即生成

新的文本数据样本，用于训练更加稳健和泛化能力强的机器学习模型；二是数据填充，即在数据不平衡的情况下，通过生成额外的样本来平衡数据集，改善模型性能。

- **自然语言理解**：包括理解和处理自然语言文本。一是文本蕴含，即判断一个文本是否可以从另一个文本中推断出来，应用于文本推理和语义理解；二是语句相似度的分析，即评估两个句子之间的语义相似度，适用于重复检测和相似查询匹配。

- **对话生成与对话管理**：一是智能客服，为客户提供即时的帮助和支持，处理常见问题和请求；二是虚拟助理，通过自然语言与用户互动，提供个性化的服务和信息。

- **文本补全与预测**：根据部分输入预测或补全剩余的文本。一是代码补全，在编程环境中，根据部分代码自动补全剩余代码，提高编程效率；二是文本预测，在文档编辑器中，根据用户输入预测后续文本，提升写作效率。

T5 模型的优势主要体现在以下几个方面：

1. 统一的文本到文本框架

通用性：T5 模型采用统一的文本到文本框架，可以处理广泛的自然语言处理任务。无论是分类、生成、翻译还是问答，所有任务都可以转换为文本生成问题，这使得模型在不同任务之间具有一致的接口和训练方法。

简化流程：统一框架简化了模型设计和实现过程，减少了不同任务需要的特定处理步骤。开发者可以使用相同的框架来处理多种任务，提高开发效率。

2. 强大的预训练能力

大量数据学习：T5 在大规模未标注数据上进行预训练，捕捉了丰富的语言模式和语义关系。这使得模型在下游任务上具有更好的初始表现，并且能够从相对较少的标注数据中学习。

3. 多任务学习

泛化能力：通过在预训练阶段处理多种任务，T5 模型能够更好地泛化到新的任务和数据集。多任务学习使模型在处理单一任务时具有更强的鲁棒性和适应性。

共享知识：不同任务之间共享的知识可以互相增强，提高整体性能。例如，文本生成任务可以受益于分类任务的语义理解能力。

4. 高性能表现

基准测试：T5 在多个标准自然语言处理基准任务上取得了领先的结果，如 GLUE、SuperGLUE、SQuAD 等。这证明了其在各种复杂语言任务上的强大表现。

规模灵活：T5 提供了从小型到超大型的多个模型变体，适应不同计算资源和应用需求。从 T5-Small 到 T5-11B，用户可以根据具体需求选择合适的模型规模。

然而，T5 模型也面临一定的挑战，主要体现在以下几个方面：

1. 计算资源需求

高计算成本：大规模模型（如 T5-11B）需要大量的计算资源和显存，训练和推理成本高昂。这对中小企业和研究机构来说可能是一个挑战。

硬件依赖：为了实现最佳性能，T5 需要高性能的硬件设备（如 GPU 或 TPU），这对资源有限的用户来说可能不太现实。

2. 数据隐私和安全

数据依赖：T5 的预训练依赖于大量的文本数据，可能会包含敏感信息。如何保护数据隐私和确保数据安全是一个重要问题。

偏见问题：预训练数据中的偏见和不平等可能会在模型中反映出来，影响模型的公正性和准确性。如何识别和消除这些偏见是一个挑战。

3. 模型理解和解释

黑箱模型：T5 作为一个复杂的深度学习模型，其内部工作机制难以理解和解释。对于一些关键应用（如医疗和法律），模型的透明度和可解释性是非常重要的。

解释工具：需要开发更好的工具和方法来解释模型的预测和决策过程，帮助用户理解和信任模型输出。

4. 实时应用和部署

延迟问题：大规模模型的推理速度较慢，对于需要实时响应的应用（如聊天机器人和实时翻译）可能不够高效。

优化和压缩：如何优化和压缩模型以适应低延迟和低资源的应用场景是一个需要解决的问题。

5. 多语言和跨文化适应

语言覆盖：虽然 T5 在多语言任务上表现出色，但其性能在低资源语言和方言上的表现可能有限。如何提升模型在这些语言上的表现是一个挑战。

文化适应：不同文化和地区的语言使用方式和习惯差异很大，模型需要能够适应这些差异，提供准确和相关的输出。

1.1.2 文生图

文本到图像（Text-to-Image, T2I）的生成技术在近年来取得了巨大的进展，成为人工智能和计算机视觉领域的一个热点研究方向。这项技术能够根据文本描述自动生成相应的图像，在创

意设计、内容创作、虚拟现实等领域有着广泛的应用前景。扩散模型作为一种新兴的生成模型，自 2020 年被提出以来，在图像生成领域迅速崛起，展现出了巨大的潜力，成为当前最先进的生成模型之一。与生成对抗网络 (GANs) 和变分自编码器 (VAEs) 等传统方法相比，扩散模型在生成质量、多样性和可控性等方面都展现出显著优势。本节旨在系统梳理扩散模型在文本到图像生成任务中的最新研究进展，重点关注模型性能突破、架构优化、多模态融合、可编辑性、安全性考虑以及个性化定制等方面的创新。

1. 基础扩散模型

Jonathan Ho 等人 (2020) 提出的去噪扩散概率模型 (Denoising Diffusion Probabilistic Models, DDPM) 奠定了扩散模型的理论基础。DDPM 是一种受非平衡热力学启发的潜在变量模型，在理论上与去噪分数匹配和 Langevin 动力学有着密切的联系，其核心思想可以概括为以下步骤：

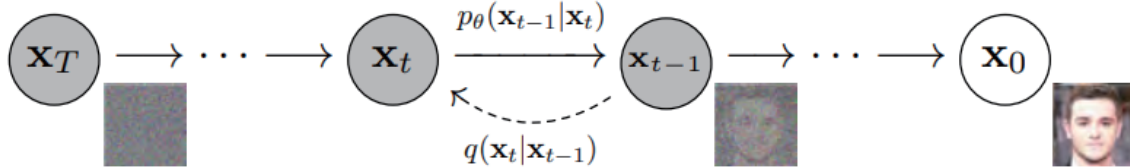


图 1.5: 扩散概率模型的有向图表示

(1) 正向 (扩散) 过程

- 加噪过程：从数据分布（如真实图像）开始，逐渐向数据添加噪声，最终将数据转化为标准高斯噪声。这是一个马尔可夫过程，每一步都通过向数据添加少量的高斯噪声来实现。

- 方程表示：通常表示为一个连续时间的随机微分方程或离散时间的马尔可夫链。对于每个时间步 t ，给定图像 x_t ，生成图像 x_{t-1} 的过程为：

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t\mathbf{I}), q(x_{t:T}|x_0) = \prod_{t=1}^T q(x_t|x_{t-1}) \quad (1.1)$$

其中， β_t 是控制噪声强度的超参数。正向过程有一个重要特性，可以直接采样任意时刻 t 下的加噪结果 x_t ，定义 $\alpha_t := 1 - \beta_t, \bar{\alpha}_t := \prod_{s=1}^t \alpha_s$ ，则得到

$$q(x_t|x_0) := \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)\mathbf{I}). \quad (1.2)$$

这一公式使得我们可以直接获得任意程度的加噪图像，方便后续的训练。

(2) 反向过程

- 去噪过程：与前向过程相反，从一张随机高斯噪声图像 x_T 开始，通过一个学习到的反向过程，逐步去除噪声，最终生成逼真的图片 x_0 。

- 方程表示：反向过程也是一个马尔可夫过程，每一步通过去除噪声来接近真实图片。反向过程的每一步可以表示为：

$$p_{\theta}(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_{\theta}(x_t, t), \Sigma_{\theta}(x_t, t)), p_{\theta}(x_{0:T}) := p(x_T) \prod_{t=1}^T p_{\theta}(x_{t-1}|x_t) \quad (1.3)$$

其中， μ_{θ} 和 Σ_{θ} 是学习到的均值和方差参数。这个过程可以理解为，我们根据 x_t 作为输入，预测高斯分布的均值和方差，再基于预测的分布进行随机采样得到 x_{t-1} 。通过不断的预测和采样过程，最终生成一张真实的图片 x_0 。

(3) 模型训练

模型训练是通过最小化数据的负对数似然的变分上限来进行的：

$$\mathbb{E}[-\log p_{\theta}(x_0)] \leq \mathbb{E}_q \left[-\log \frac{p_{\theta}(x_{0:T})}{q(x_{1:T}|x_0)} \right] = \mathbb{E}_q \left[-\log p(x_T) - \sum_{i \geq 1} \log \frac{p_{\theta}(x_{t-1}|x_t)}{q(x_i|x_{t-1})} \right]. \quad (1.4)$$

设方差项 $\Sigma_{\theta}(x_t, t) = \sigma_t^2 I$ ，即选用一个依赖于时间 t 的无需训练的常数项。利用 KL 散度，经过一系列推导，得到优化目标函数就是两个分布均值的二范数：

$$L_t = \mathbb{E}_q \left[\|\tilde{\mu}_t(x_t, x_0) - \mu_{\theta}(x_t, t)\|^2 \right] = \mathbb{E}_{x_0, \epsilon} \left[\left\| \frac{1}{\sqrt{\alpha_t}} \left(x_t(x_0, \epsilon) - \frac{\beta_t}{\sqrt{1 - \alpha_t}} \epsilon \right) - \mu_{\theta}(x_t(x_0, \epsilon), t) \right\|^2 \right], \quad (1.5)$$

其中， x_t 为 μ_{θ} 的输入，其他的量都是常数，其中的未知量其实只有 ϵ ，于是可以直接定义均值：

$$\mu_{\theta}(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \alpha_t}} \epsilon_{\theta}(x_t, t) \right). \quad (1.6)$$

也就是说，不需要用网络直接预测 $\tilde{\mu}_t(x_t, x_0)$ ，可以用网络 $\epsilon_{\theta}(x_t, t)$ 先预测噪声 ϵ ，然后把预测出来的噪声带入到定义好的表达式去计算出预测的均值，结果相同。忽略 L_t 中的常数项大小，从而可以得到：

$$L_t = \mathbb{E}_{x_0, \epsilon} \left[\|\epsilon - \epsilon_{\theta}(\sqrt{\alpha_t}x_0 + \sqrt{1 - \alpha_t}\epsilon, t)\|^2 \right], \quad \epsilon \sim N(0, 1) \quad (1.7)$$

因此，网络的输入是一张和噪声线性组合的图片，为了实现基于扩散模型的生成，DDPM 采用了一个 U-Net 结构的 Autoencoder 来对 t 时刻的噪声进行预测，即 $\epsilon_{\theta}(x_t, t)$ 网络训练时采用的训练目标非常简单，即：

$$\|\epsilon_{\theta}(\sqrt{\alpha_t}x_0 + \sqrt{1 - \alpha_t}\epsilon) - \epsilon\| \quad (1.8)$$

其中， ϵ 是高斯噪声。这里，噪声预测网络以加噪图片作为输入，目标是预测所添加的噪声，训练目标即希望预测的噪声和真实的噪声一致。算法为 Algorithm 1。

Algorithm 1: Training

```

1: repeat
2:   Initialize  $x_0 \sim q(x_0)$ 
3:   Sample  $t \sim \text{Uniform}(\{1, \dots, T\})$ 
4:   Add noise  $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ 
5:   Take a gradient descent step on
        $\nabla_{\theta} \|\epsilon_{\theta}(\sqrt{\alpha_t}x_0 + \sqrt{1 - \alpha_t}\epsilon) - \epsilon\|^2$ 
6: until converged

```

Algorithm 2: Sampling

```

1: Sample  $x_T \sim \mathcal{N}(0, \mathbf{I})$ 
2: for  $t = T, \dots, 1$  do
3:    $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$  if  $t > 1$ , else  $\mathbf{z} = 0$ 
4:   Add noise  $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ 
5: end for
6: return  $x_0$ 

```

(4) 生成过程

- 采样：训练完毕后，从标准高斯分布中采样一个初始噪声图片，通过反向过程逐步去噪，最终生成一个图片。

- 逐步去噪：从 $t = T$ 开始，逐步应用反向过程的马尔可夫链，直到 $t = 1$ ，生成最终的图像。算法为 Algorithm 2。

总之，DDPM 定义了一个前向过程，逐步向数据添加高斯噪声，然后通过学习一个反向过程来逐步去噪，最终生成样本。这种方法在 CIFAR10 等数据集上取得了当时最好的 FID 分数。

2. 改进扩散模型

自 2020 年 DDPM 提出以来，扩散模型在图像生成质量上取得了显著进展。仅一年后，Dhariwal 和 Nichol(2021) 进一步改进了扩散模型，提出了改进的架构 (ADM) 和分类器引导技术，通过一系列消融实验优化了模型结构。研究表明，扩散模型在图像质量上已经超越了当时最先进的 GAN 模型。在 ImageNet 数据集上，ADM 模型表现尤为出色，在 128×128 、 256×256 和 512×512 分辨率下分别达到了 2.97、4.59 和 7.72 的 FID 分数，优于之前的最佳结果，不仅在样本质量上超越了 GAN，还在分布覆盖率方面表现更佳。

Model	FID	sFID	Prec	Rec
ImageNet 128×128				
BigGAN-deep [8]	6.02	7.18	0.86	0.35
LOGAN [†] [74]	3.36			
ADM	5.91	5.09	0.70	0.65
ADM-G (25 steps)	5.98	7.04	0.78	0.51
ADM-G	2.97	5.09	0.78	0.59
ImageNet 256×256				
DCTransformer [†] [48]	36.51	8.24	0.36	0.67
VQ-VAE-2 ^{††} [57]	31.11	17.38	0.36	0.57
VQ-VAE-2 (RS) ^{††§} [57]	~ 10			
VQ-GAN [‡] [21]	15.97	19.05	0.63	0.58
VQ-GAN (RS) ^{‡§} [21]	5.06	7.34	0.79	0.48
IDDPM [‡] [49]	12.26	5.42	0.70	0.62
SR3 ^{††} [60]	11.30			
BigGAN-deep [8]	6.95	7.36	0.87	0.28
ADM	10.94	6.02	0.69	0.63
ADM-G (25 steps)	5.44	5.32	0.81	0.49
ADM-G	4.59	5.25	0.82	0.52
ImageNet 512×512				
BigGAN-deep [8]	8.43	8.13	0.88	0.29
ADM	23.24	10.19	0.73	0.60
ADM-G (25 steps)	8.41	9.67	0.83	0.47
ADM-G	7.72	6.57	0.87	0.42

图 1.6: ADM 模型与竞争模型表现对比

此外，还可以引入分类器引导技术，通过利用分类器的梯度来指导扩散模型的采样过程，在条件图像生成任务上进一步提高样本质量。研究表明，将分类器引导与上采样扩散模型结合，在 256×256 和 512×512 分辨率的 ImageNet 上分别实现了 3.94 和 3.85 的 FID 分数。

Model	FID	sFID	IS	Prec	Rec	Model	FID	sFID	IS	Prec	Rec
ImageNet 256×256						ImageNet 512×512					
ADM	10.94	6.02	100.98	0.69	0.63	ADM	23.24	10.19	58.06	0.73	0.60
ADM, ADM-U	7.49	5.13	127.49	0.72	0.63	ADM, ADM-U	9.96	5.62	121.78	0.75	0.64
ADM-G	4.59	5.25	186.70	0.82	0.52	ADM-G	7.72	6.57	172.71	0.87	0.42
ADM-G, ADM-U	3.94	6.14	215.84	0.83	0.53	ADM-G, ADM-U	3.85	5.86	221.72	0.84	0.53

图 1.7: 加入分类器的 ADM 模型与竞争模型表现对比

与 GAN 相比，改进的扩散模型在保持更好的分布覆盖率的同时，能够达到相当的感知质量，这为其在实际应用中的潜力提供了有力支持。然而，扩散模型仍然面临采样速度较慢、缺乏显式潜在表示等挑战。

3. 扩散模型的架构优化

尽管扩散模型在生成质量上表现出色，但其主要缺点是推理速度慢。为解决这一问题，研究重点转向如何提高模型效率，以便在移动设备等资源受限的环境中部署，研究者们提出了多种优化方法。其中，Yang Zhao 等人 (2023) 提出的 MobileDiffusion(MD) 是这一方向的代表性工作。

通过对模型架构和采样技术的全面优化，MD 实现了在移动设备上亚秒级的 512×512 图像生成，为扩散模型在移动端的实际应用铺平了道路。

模型架构方面，MD 的主要优化策略包括：在 UNet 中间层增加 transformer 块，减少高分辨率层的计算；解耦自注意力和交叉注意力，仅在低分辨率保留自注意力；共享 key-value 投影，使用 swish 替代 gelu 激活函数。

除了架构优化，采样技术的改进也是提高模型效率的重要方面。MD 采用了蒸馏和扩散-GAN 微调技术，实现了 8 步和 1 步推理，大大加快了生成速度。

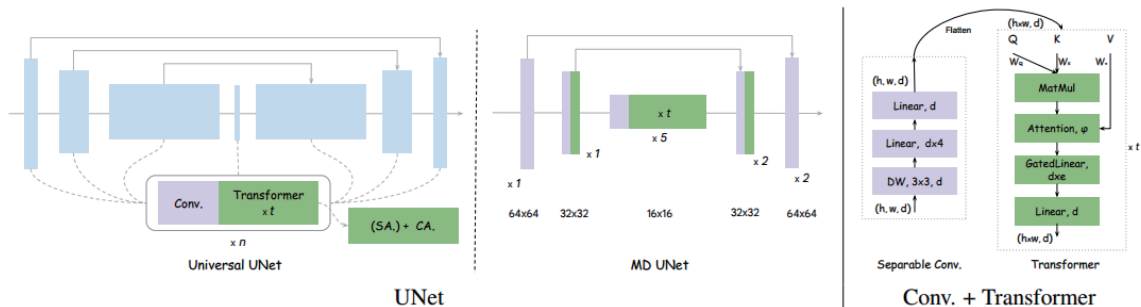


图 1.8: Mobile Diffusion(MD) 模型架构

这些优化使得 MD 模型参数减少到不到 400M，比 Stable Diffusion 1.5 小 46%，同时在

iPhone15 Pro 上可以 0.2 秒生成 512×512 图像，生成质量与 SD-1.5 相当。这些技术的应用不仅大幅减少了模型参数，提高了推理效率，还保持了与 Stable Diffusion 1.5 相当的生成质量，为扩散模型在边缘计算和实时应用场景中的部署奠定了基础。

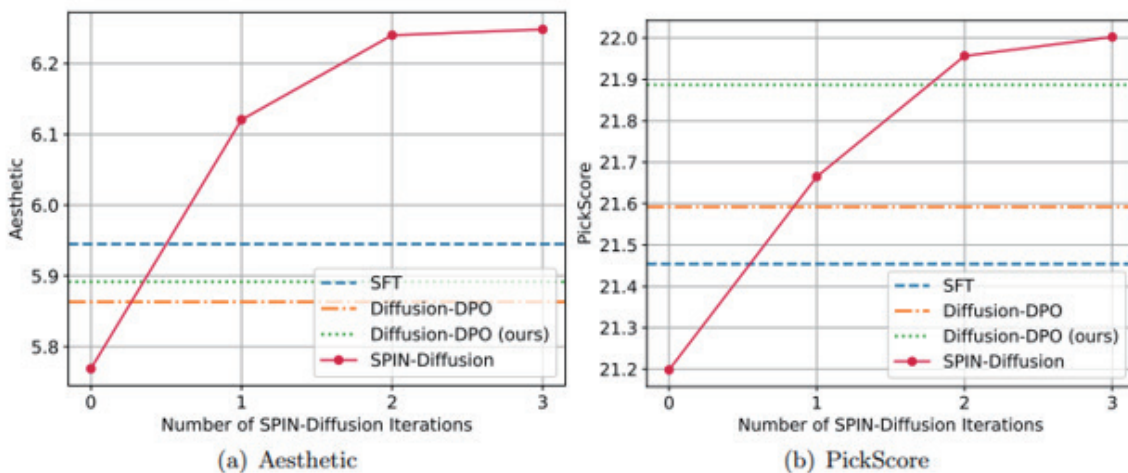
4. 自我博弈微调扩散模型

扩散模型已成为生成式 AI，特别是图像生成领域的关键技术。然而，与大型语言模型 (LLM) 的微调相比，扩散模型微调仍是生成式人工智能 (GenAI) 中一个尚未充分探索的领域。像稳定扩散 (SD) 和 SDXL 等扩散模型表现已经十分优异，但依赖于监督微调，其性能在达到一定数据量后往往停滞不前。强化学习 (RL) 虽然被用于利用人类偏好数据进行微调，但每个文本提示至少需要两张图像（“赢家”和“输家”图像），增加了模型训练的数据需求。

为解决这些问题，Huizhuo Yuan 等人 (2024) 提出了一种自我博弈微调扩散模型 (SPIN-Diffusion)。SPIN-Diffusion 利用自我博弈机制，通过模型与其早期版本的竞争不断改进，避免了对大量人工偏好数据的依赖。SPIN-Diffusion 的核心是自我对抗机制，主玩家（当前模型）和对手玩家（模型的早期版本）通过一般和零和博弈进行迭代改进，其目标函数设计考虑了反向采样过程中的所有中间图像，并使用高斯重参数化技术进行高效计算。

在 Pick-a-Pic 数据集上，以 StableDiffusion-1.5 为基础模型进行的实验表明，SPIN-Diffusion 从第一轮开始就在人工偏好对齐和视觉吸引力方面优于传统的监督微调方法 (SFT)。到第二轮时，它在所有指标上都超过了基于 RLHF 的方法。更重要的是，SPIN-Diffusion 使用更少的数据就实现了这些卓越的性能，突显了其在有限图像提示数据集上的高效利用能力。

实验证明，SPIN-Diffusion 为传统的监督微调和 RL 策略提供了一种替代方案，它显著提高了模型性能和与目标分布的对齐，促进了迭代的自我改进过程，使模型能在使用较少数据的情况下就能达到甚至超过基于人类偏好的强化学习模型的性能。由于 SPIN-Diffusion 无需依赖大量人工偏好数据，使其在具有自定义数据集的广泛应用中特别有优势。



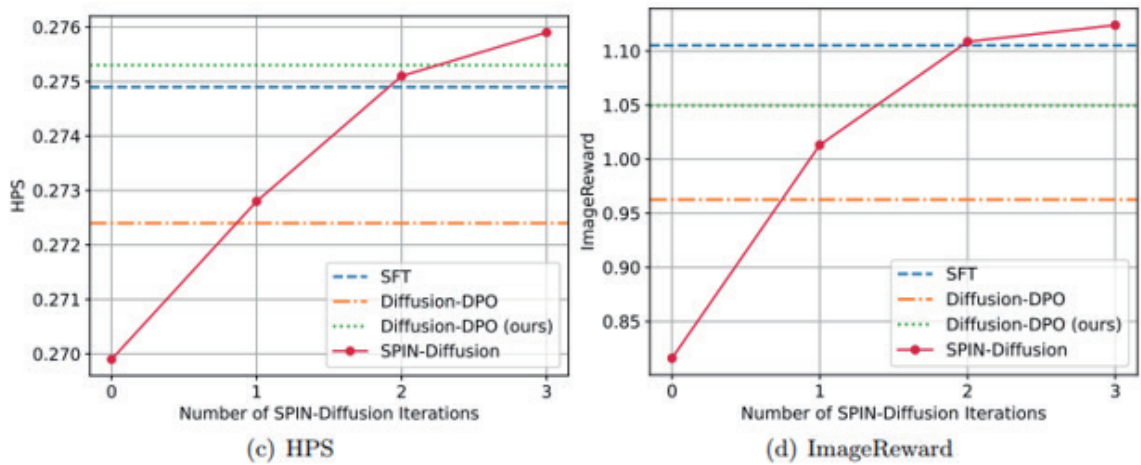
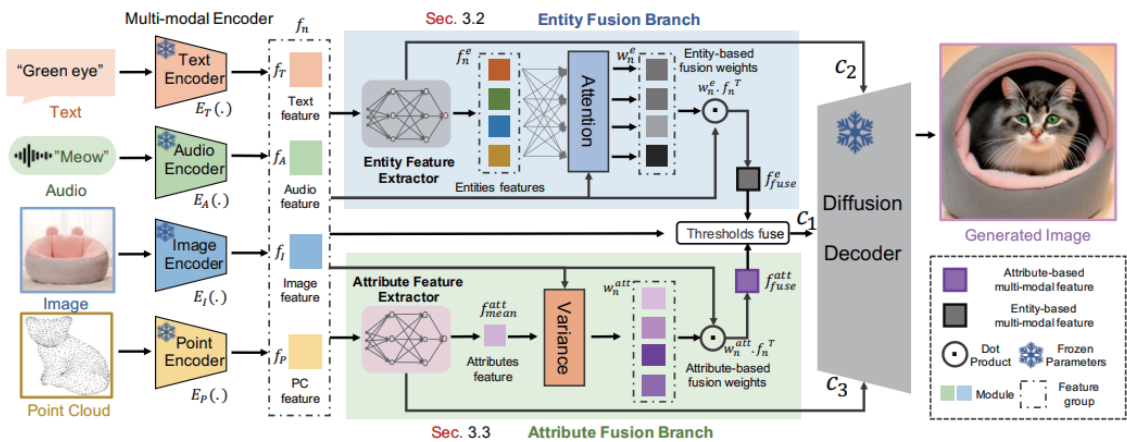


图 1.9: 不同迭代次数下 SPIN-Diffusion、SD-1.5、SFT、Diffusion-DPO 对比

5. ImgAny: 多模态融合模型

随着技术的发展,研究者开始探索如何将扩散模型应用于更复杂的多模态输入场景。Yuanhuiyi Lyu 等人 (2024) 提出的 **ImgAny** 是这一方向的代表性方法,受人类认知过程启发,他们提出了一种端到端的多模态生成模型,能够模仿人类推理、灵活处理 7 种模态 (文本、音频、图像、点云、热成像、深度和事件数据) 的任意组合作为输入并生成高质量图像,实现了真正的多模态图像生成。

图 1.10: **ImgAny** 整体框架: 多模态编码器, 实体融合分支, 属性融合分支

ImgAny 的创新在于其融合策略。它在实体和属性两个层面上集成和协调多模态输入,无需针对特定模态进行调整。具体来说:

- 实体融合分支确保输入和输出之间的一致性,它从由专门构建的实体知识图提供支持的多模态表示中提取实体特征。

- 属性融合分支保留和处理来自不同输入模态的属性，它通过提出的属性知识图有效地融合了来自不同输入模态的不同属性。

- 实体和属性特征被自适应地融合为预训练的扩散模型的条件输入，用于图像生成。

实验结果表明，得益于实体融合和属性融合两个融合分支的整合，ImgAny 在任何模态组合中都表现出强大的生成质量，这种方法不仅提高了生成图像的质量和语义一致性，还在参数数量和推理成本上表现出显著优势。人类评估结果显示，ImgAny 生成的图像具有良好的推理一致性和创造性，展示了相当人类水平的推理和创造力。

6. ConsiStory：一致性图像生成模型

在文本到图像生成技术的发展中，一个长期存在的挑战是如何在不同提示下一致地描绘同一主题。传统方法通常需要对模型进行微调或大规模预训练，这不仅耗时耗力，还难以实现多主题描绘和文本 - 图像的精确对齐。为解决这些问题，YOAD TEWEL 等人 (2024) 提出了 ConsiStory 方法。ConsiStory 是一种无需训练的文本到图像生成方法，能够跨多个文本提示一致地生成同一主体的图像。该模型的关键在于设计主题驱动的共享注意力块、基于对应的特征注入和共享预训练模型的内部激活，其核心机制包括以下三点：

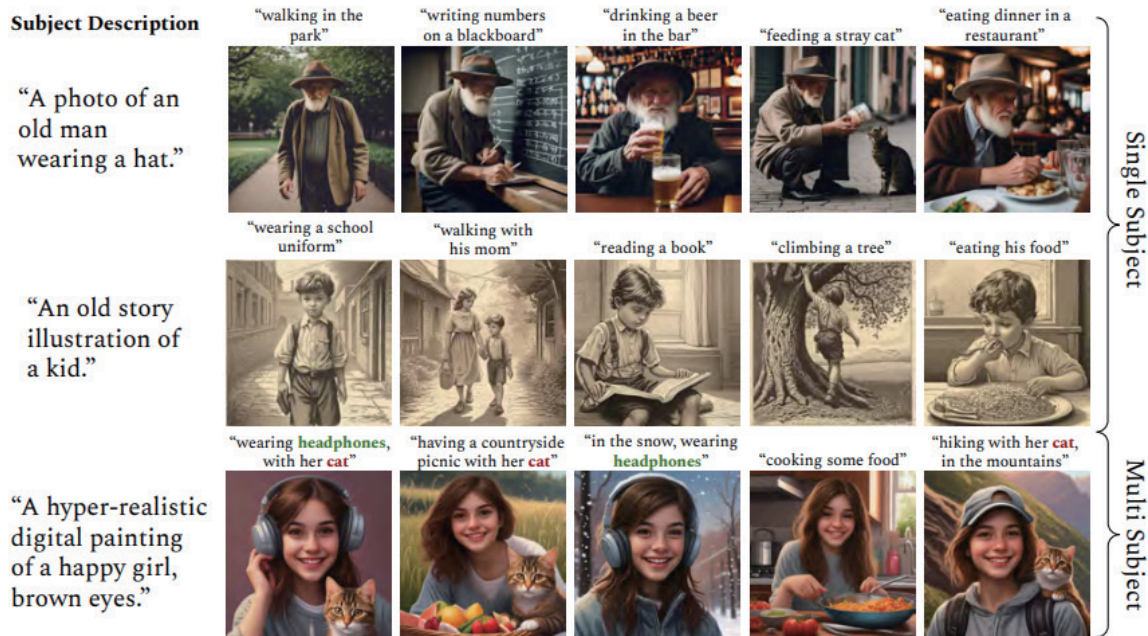


图 1.11: ConsiStory 保持主题、语义一致性且无需预训练

- 主题驱动的自然注意力机制 (SDSA)：通过扩展自然注意力层，允许一个图像的查询同时关注其他图像中的键和值，但仅限于同一主体区域，从而在生成的图像中共享特定主体的信息。

- 保持图像多样性：一是结合非一致采样步骤中获得的特征，增加图像间的细微差别；二是

通过注意力丢弃机制来削弱 SDSA，避免过度的同质化。

- 特征注入：建立密集的对应对图，在不同图像间对齐自注意力层的输出特征。特别关注改善关键对应区域（例如人物的左眼）的特征一致性，从而在细节层面增强图像间的连贯性。

ConsiStory 的优势显著在于，无需优化或预训练即可实现与文本提示对齐的主体一致性，能自然扩展到多主体场景，实现常用对象的训练自由个性化。实验结果表明，ConsiStory 在主体一致性和文本对齐方面表现优异，生成速度比现有技术快约 20 倍。另外，ConsiStory 存在一些局限性，如对跨注意力图的依赖可能导致在特殊风格时的失败；在风格和外观分离上存在困难。

7. Motion Guidance: 运动引导

扩散模型的可控性是其相对于其他生成模型的重要优势之一，它允许用户以相对粗略的规模编辑图像。但是，利用扩散模型精确控制和编辑生成结果仍然很困难。Daniel Geng 等人 (2024) 提出的 Motion Guidance（运动引导）方法允许用户通过指定密集的区域，来精确编辑图像中对象的位置、姿势和形状。

Motion Guidance 技术的核心在于：

- 利用现成的光流网络 (如 RAFT) 作为引导信号估计运动。
- 在扩散模型采样过程中应用运动引导，同时保持视觉相似性。
- 设计结合流损失和颜色一致性损失的引导函数。
- 采用递归去噪、梯度裁剪等技巧提高稳定性。

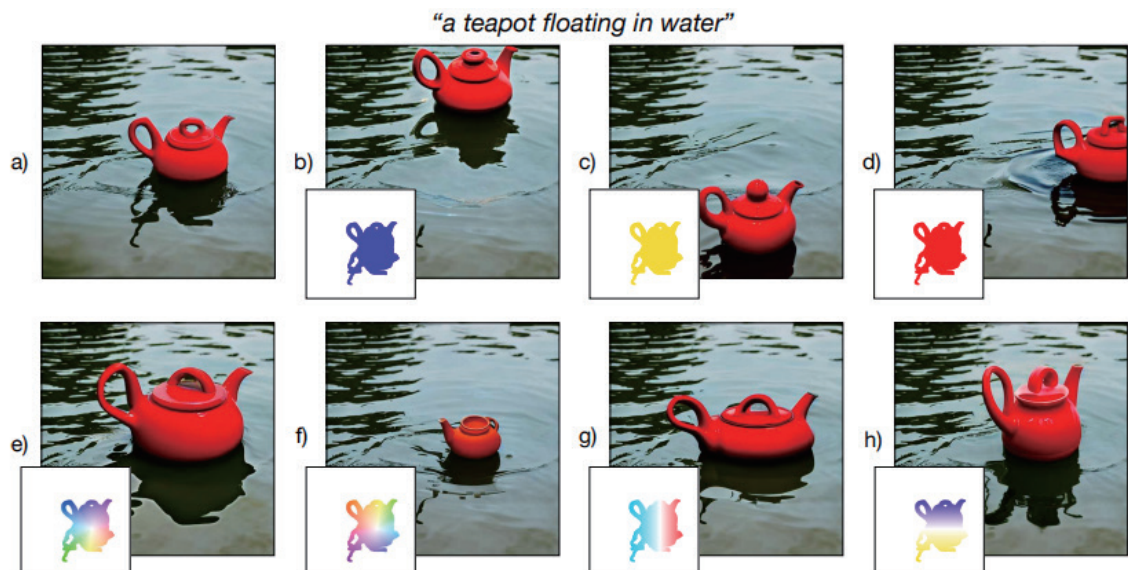


图 1.12: Motion Guidance 方法实现精确编辑生成图像

这种零样本技术利用现成的光流网络作为引导信号，不需要微调或特定架构，实现了复杂

的运动编辑，为视频制作和动画创作提供了强大工具。这种方法不需要额外训练，可以处理复杂的运动场，包括平移、旋转、缩放、变形等，适用于真实图像和生成图像。

研究表明，该技术在各种复杂运动编辑任务上展示了良好效果；与基线方法（如 Instruct-Pix2Pix、前向变形等）相比具有优势；可以实现“运动迁移”，将视频中提取的运动应用到静态图像。此外，运动引导技术可能会遗忘源图像的部分内容，在采样速度和极端运动处理（如翻转）上还存在一些局限性。

8.POSI：安全图像生成的提示优化器

随着文本转图像模型在实践中的广泛使用，其安全性问题日益受到关注。尽管这类模型在基于文本提示生成图像方面表现出色，但它们容易受到不安全输入的影响，从而生成不适当内容，如涉及性、骚扰和非法活动的图像。现有的防御措施如图像审查器、模型微调和嵌入阻止机制等，在实际应用中存在局限性，这凸显了开发更实用、通用的安全生成图像方法的必要性。

为应对这一挑战，Zongyu Wu 等人 (2024) 提出了 POSI(Prompt Optimizer for Safe Image generation)，这是首个针对黑盒场景的通用提示优化器。POSI 的核心思想是通过优化输入提示，在保留语义信息的同时，降低生成不适当图像的可能性。POSI 的主要创新在于：首次研究通过提示工程实现安全 T2I 生成；提出了可应用于各种 T2I 模型的黑盒通用提示优化器；设计了新的奖励函数，平衡了安全性和语义保留。

POSI 的核心步骤为：1) 构建有害 - 无害提示数据集；2) 对语言模型进行监督微调 (SFT)，获得基本的提示重写能力；3) 设计新的奖励函数，衡量生成图像的有害性和文本对齐度；4) 使用近端策略优化 (PPO) 进一步训练模型。

研究表明，POSI 具备将有害提示转换为无害提示的能力，在多个 T2I 模型上有效降低了不适当图像的生成概率，同时对语义保留的影响较小。POSI 还表现出良好的迁移性，可应用于黑盒模型如 DALL-E3 和 Midjourney。此外，POSI 可以与现有方法结合，进一步提高性能，显示出其在实际应用中的潜力和灵活性。这些结果证明了 POSI 在构建更安全的 AI 系统方面的重要价值。

尽管 POSI 取得了显著成果，但我们也应该认识到安全性和语义保留之间存在内在冲突，需要精细平衡，即如何在保障生成内容安全性的同时，更好地保留原始提示的语义信息。

9. Gen4Gen：生成多概念组合的数据生成方法

随着扩散模型能力的不断提升，文本到图像扩散模型在个性化概念生成领域取得了显著进展。然而，当前技术在处理多个个性化概念时面临两大挑战：首先，现有方法难以有效扩展到多概念场景，这可能是由于预训练数据集中复杂场景与简洁文本描述之间的不匹配；其次，缺

乏全面的评估指标来衡量多概念个性化生成的质量，包括概念的存在性、相似度以及与整体文本描述的一致性。这些挑战突显了改进数据集质量和评估方法的需求。

为应对上述挑战，Chun-Hsiao Yeh 等人 (2024) 提出了 Gen4Gen 方法，一种半自动化的数据集构建方法。Gen4Gen 巧妙地整合了前沿 AI 技术，包括前景提取、大规模语言模型和图像修复算法，将用户提供的概念图像合成为复杂场景，并生成相应的详细文本描述。

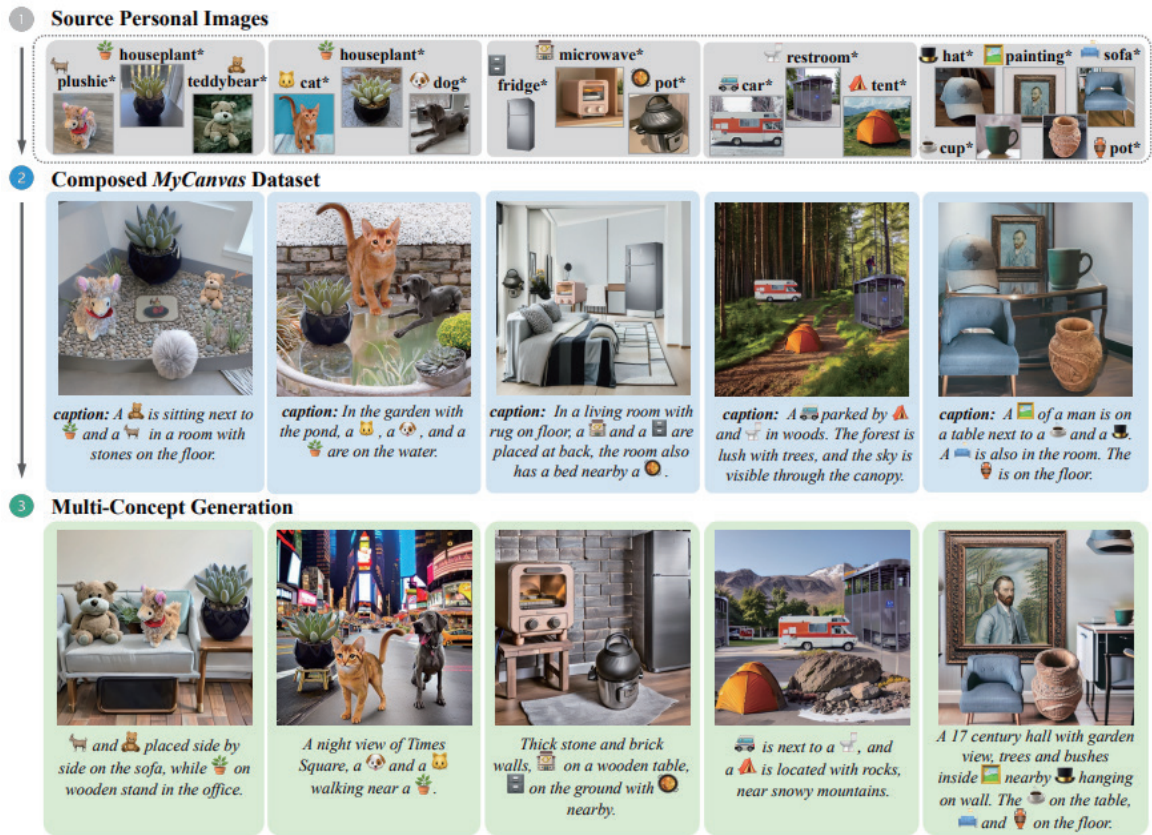


图 1.13: MyCanvas 数据集解决了多概念生成图像问题

基于 Gen4Gen 构建的 MyCanvas 数据集，涵盖 41 种组合、2684 张图像，为多概念个性化任务提供了高质量的基准测试资源。该数据集的创新之处在于，其包含了更为复杂和真实的场景组合，为评估模型在处理多个个性化概念时的性能提供了有力工具。

为全面评估多概念个性化生成的质量，Chun-Hsiao Yeh 等人 (2024) 还提出了 CP-CLIP 和 TI-CLIP 两项新指标。这些指标不仅可以评估个性化概念的相似度，还考虑了所有概念在图像中的存在性以及图像与整体文本描述的一致性。

实验结果表明，通过提升数据质量和优化提示策略，使用 MyCanvas 数据集进行训练，不仅能显著提升现有方法（如 Custom Diffusion）在多概念生成任务上的性能，还能显著提高多概念个性化图像生成的质量，且无需对模型架构或训练算法进行修改。通过改进数据质量和评估

方法，Gen4Gen 为多概念个性化图像生成领域提供了新的研究范式和基准，这种方法在处理复杂组合和相似概念等高难度场景时尤其有效。

总的来说，扩散模型为文本到图像生成领域带来了新的机遇和挑战。文本到图像生成技术在近期取得了显著进展，从模型架构优化到多模态融合，从安全性考虑到个性化定制，研究者在多个方面推动了该领域的发展。这些进展不仅提高了生成图像的质量和效率，还扩展了技术的应用范围，同时也关注了安全性和道德问题。然而，仍然存在诸多挑战，如何在保持高质量生成的同时进一步提高效率，如何更好地处理复杂的多概念组合，以及如何在追求创新的同时确保技术的负责任使用。未来的研究需要在这些方面继续深入探索，以推动文本到图像生成技术向更加成熟和实用的方向发展。

此外，扩散模型在特定领域的应用，如医疗影像生成、艺术创作辅助等，也值得深入探索。随着计算能力的提升和数据可用性的增加，扩散模型将在更广泛的应用场景中发挥重要作用。

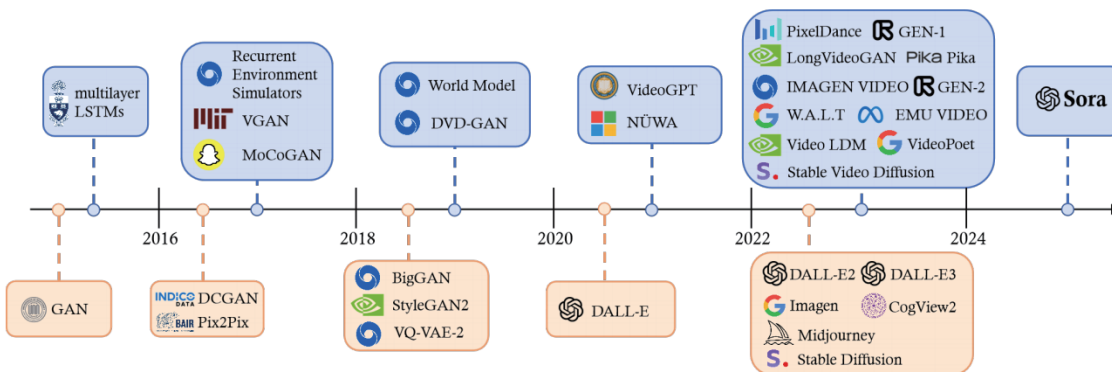


图 1.14: 生成式人工智能在视觉领域的历史

1.1.3 文生视频

随着人工智能技术的飞速发展，特别是生成模型的日益成熟，文生视频（Text-to-Video, T2V）技术逐渐崭露头角，成为 AI 生成内容（AIGC）领域的一颗璀璨新星。文生视频技术不仅要求模型能够深入理解复杂的文本描述，还要能够在动态变化的空间和时间维度上生成连贯、高质量的视频内容。本节旨在深入探讨文生视频技术的 Sora 模型、最新进展、应用场景及未来发展方向，为读者呈现一个全面且具体的文生视频技术图景。

文生视频的历史

文生视频（Text-to-Video, T2V）技术的发展是计算机视觉和生成模型领域的一个重要里程碑。

图 1.14 展示了从简单图像生成到复杂视频生成的演进过程。

- 早期图像生成技术：在深度学习革命之前，传统的图像生成技术主要依赖于纹理合成和纹理映射等方法，但这些方法受限于生成复杂和生动图像的能力。

- 生成对抗网络（GANs）与变分自编码器（VAEs）的引入：GANs 和 VAEs 的出现标志着图像生成领域的一个重要转折点。这些模型通过学习和模拟数据分布，能够生成更加逼真和多样化的图像内容。

- 流模型与扩散模型的兴起：随着流模型和扩散模型的发展，图像生成的质量和细节得到了进一步提升。这些模型为后续的文生视频技术研究奠定了坚实的基础。

- 文生视频技术的萌芽：近年来，随着 transformer 架构在 NLP 领域的成功应用，研究人员开始探索将其与视觉组件结合，应用于文本到视频的生成任务。一系列文生视频模型相继问世，推动了该领域的快速发展。

Sora 模型

Sora 是 OpenAI 在 2024 年 2 月推出的一款文本到视频（Text-to-Video, T2V）的生成模型，它能够根据用户的文本指令生成高质量、长达一分钟的视频内容。Sora 模型在文生视频领域取得了显著突破，以下是对其关键技术和特点的详细介绍：

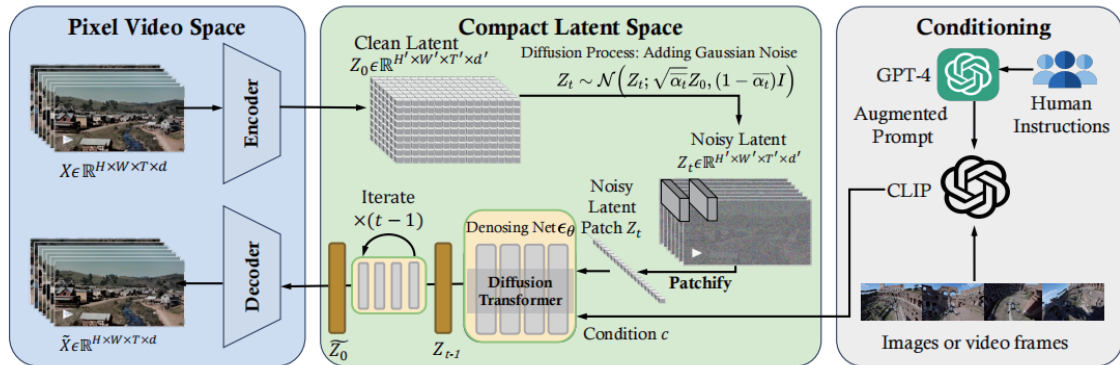


图 1.15: Sora 模型框架

- **核心架构：扩散变压器（Diffusion Transformer）**

基础结构：如图 1.15 Sora 采用了预训练的扩散变压器作为其核心架构。扩散模型通过逐步去噪和引入细节的方式，将初始的噪声图像逐渐转换为与文本指令相匹配的清晰视频。

计算效率：为了提高视频生成的计算效率，Sora 引入了时空潜在补丁（Spacetime Latent Patches）作为模型的输入单元。这些补丁通过压缩原始视频到潜在空间并提取关键特征，使得模型能够更高效地处理视频数据。

- **数据预处理与表示**

原始尺寸训练：与传统的视频生成方法不同，Sora 直接在视频的原始尺寸上进行训练，避

免了裁剪和缩放等操作带来的信息损失。这种处理方式使得 Sora 能够更好地理解和生成具有自然比例和构图的视频内容。

统一视觉表示：为了处理不同时长、分辨率和宽高比的视频数据，Sora 采用了统一的视觉表示方法。通过将视频数据转换为潜在空间中的时空补丁，模型能够灵活地处理各种复杂的视频输入。

• 指令跟随能力

描述性标题器：为了提升模型对文本指令的跟随能力，Sora 采用了类似 DALL·E3 的描述性标题器方法。具体来说，Sora 首先训练一个视频标题器来生成详细的视频描述，然后利用这些描述对模型进行微调。通过这种方式，Sora 能够更准确地理解和执行复杂的文本指令。

语言模型辅助：在生成视频的过程中，Sora 还利用了大型语言模型（如 GPT-4）来辅助处理文本指令，进一步提高了指令跟随的准确性和效率。

• 多模态输入支持

图像和视频提示：除了文本指令外，Sora 还支持图像和视频提示作为输入。用户可以通过提供静态图像或视频片段作为参考，引导模型生成与提示内容相关的视频内容。这种多模态输入的支持使得 Sora 在视频生成过程中具有更高的灵活性和个性化定制能力。

综合输入处理：Sora 能够处理不同类型的输入提示，并结合文本指令生成丰富多样的视频内容。这种综合输入处理能力使得 Sora 能够应对各种复杂的生成任务。

• 信任度与安全性

内容过滤与审查：随着生成模型能力的增强，Sora 在开发过程中充分考虑了安全性和信任度问题。模型内置了内容过滤和审查机制，以检测和过滤潜在的有害或违规内容。

数据隐私保护：Sora 还加强了对用户数据的隐私保护，通过加密处理和访问控制等措施确保用户数据的安全性和保密性。

透明度与可解释性：为了提高模型的透明度和可解释性，Sora 在生成视频的过程中提供了详细的生成步骤和参数设置信息，使用户能够更好地理解和控制生成过程。

文生视频技术的最新进展

• 扩散模型的突破：Sora 模型是文生视频领域的一个里程碑式成果。Sora 通过将时间层插入到预训练的图像扩散模型中，实现了从文本到视频的连贯生成。尤为重要的是，Sora 在生成视频时展现出了卓越的几何一致性，使得生成的视频更加符合物理定律和现实世界规律。Snap Video 和 Imagen Video 等研究展示了扩散模型在文生视频领域的巨大潜力。Snap Video 通过引入缩放时空变换器架构，有效结合了时间和空间维度的信息，生成了具有高度一致性和丰富细节

的视频内容。Imagen Video 则进一步提升了视频生成的分辨率和清晰度，为高质量视频生成树立了新的标杆。

- **3D 重建与视频质量评估**：为了更准确地评估生成视频的质量提出了一种创新的基准测试方法。该方法通过将生成的视频转换为 3D 模型，利用 3D 重建的准确性来间接评估视频的几何一致性和物理真实性。这种方法不仅补充了传统视频质量评估指标的不足，还为视频生成模型提供了一种更加客观、全面的评价标准。

- **用户交互与多模态指令**：Interactive Video 框架打破了传统文生视频技术的界限，引入了用户交互和多模态指令的概念。该框架允许用户在视频生成过程中通过文本、图像、绘画和拖拽等多种方式输入指令，实现对视频内容的精细控制和个性化定制。这种用户中心的设计极大地提升了视频生成的灵活性和用户满意度。

应用场景

- **电影与动画制作**：文生视频技术为电影和动画制作带来了革命性的变化。通过输入剧本描述或故事板草图，自动生成高质量的视频片段和动画效果，可以极大地提高制作效率和降低成本。同时，该技术还可以为创意工作者提供无限的灵感来源和创作工具。

- **教育与培训**：在教育领域，文生视频技术可以生成丰富多样的教学视频和模拟实验场景。学生可以通过观看生动直观的视频内容来加深理解和记忆知识点；教师则可以利用该技术快速制作教学素材和资源库，提升教学质量和效果。

- **游戏开发**：游戏开发者可以利用文生视频技术快速生成游戏内的过场动画和角色表演等内容。这不仅提高了游戏的视觉表现力和沉浸感体验；还缩短了开发周期和降低了制作成本；为游戏产业带来了全新的发展机遇和竞争优势。

- **医疗健康**：在医疗健康领域，文生视频技术可以应用于手术模拟、康复训练和教学培训等方面。医生可以通过观看模拟手术视频来提前规划手术方案和预演操作步骤；患者则可以利用该技术进行康复训练指导和自我评估；医学院校还可以将其作为教学工具来培养未来的医疗人才。

- **虚拟现实与增强现实**：文生视频技术为虚拟现实（VR）和增强现实（AR）领域提供了强大的内容生成能力。通过输入文本描述或场景设定等指令生成虚拟环境或增强现实效果；用户可以在虚拟世界中自由探索和互动体验；为娱乐、教育、旅游等多个领域带来全新的应用场景和价值创造。

未来展望

尽管文生视频技术已经取得了显著进展并展现出广泛的应用前景；但仍面临着诸多挑战和

未来发展方向：

- 提升生成视频的真实感和物理一致性：除了几何一致性外；还需要关注纹理真实性、光照效果、动态交互等方面的提升；以生成更加逼真且符合物理定律的视频内容。

- 增强用户交互性和个性化定制能力：进一步开发多模态交互技术并优化指令解析机制；使模型能够更好地理解并响应用户的复杂指令和个性化需求；提升用户满意度和创作自由度。

- 提升生成视频的真实感和物理一致性：提高生成效率和可扩展性：研究更高效的视频生成算法和模型架构；减少计算资源消耗并提升生成速度；同时支持大规模数据集和多种输入格式的灵活处理以满足不同应用场景的需求。

- 探索跨领域融合与应用创新：将文生视频技术与其他领域的技术相结合（如自然语言处理、计算机视觉、机器学习等）；探索新的应用场景和解决方案；推动技术的普及和发展并为社会带来更大的价值创造。

总之，文生视频技术作为 AI 生成内容领域的重要分支之一；正逐步走向成熟并展现出巨大的发展潜力。随着技术的不断进步和应用场景的不断拓展；我们有理由相信文生视频将在未来为各个领域带来更加丰富多样的视觉体验和更加高效便捷的内容生成方式。

1.2 大模型产品简介

1.2.1 国内大模型

Kimi

Kimi 是由月之暗面科技有限公司（Moonshot AI）开发基于 moonshot-v1 架构的人工智能助手，旨在为用户提供安全、高效、智能的信息处理和交互体验。在 Kimi 的官网上，Moonshot 对 Kimi 有一段轻松的介绍：

Kimi 是一个有着超大“内存”的智能助手，可以一口气读完二十万字的小说，还会上网冲浪，快来跟他聊聊吧。

从上述别具趣味的官方介绍中，我们也可以一窥 Kimi 的核心功能：

- **长文本处理能力**：与 ChatGPT 相比，Kimi 的长文本处理能力是一个显著的优势。大多数语言模型受限于较短的上下文窗口，而 Kimi 能够处理长达 20 万字的文本，这在进行长篇文献分析、复杂报告生成或深入对话时提供了更大的灵活性和深度。
- **高级语义理解**：Kimi 的语义理解能力得益于先进的 Transformer 架构，特别是其自注意力机制，这使得 Kimi 在处理复杂的语言结构和隐含意义方面具有优势，能够提供更准确和深入的语义分析。
- **信息检索能力**：Kimi 的信息检索能力可能包括更先进的算法，如基于深度学习的语义搜索技术，这使得 Kimi 在处理搜索查询时能够提供更相关和精确的结果，与依赖传统搜索引擎的模型相比，这是一个技术突破。这种能力在需要大量外部信息支持的任务中尤为重要，而 GPT 模型通常依赖于自身的知识库。
- **多模态交互**：Kimi 的多模态交互能力，特别是语音识别和生成，使其在交互方式上更加丰富和自然。这种能力在提供用户界面和体验方面提供了新的可能性，与主要侧重文本交互的模型相比，这是一个重要的差异化因素。
- **个性化服务**：“Kimi+” 功能提供了高度的个性化定制能力，这在业界是相对少见的。通过用户特定的数据和偏好，Kimi 可以提供定制化的服务和响应，这种个性化水平在大型语言模型

中并不常见。此外，Kimi 的用户支持系统，特别是其付费支持模式，为用户提供了一种直接参与和支持 Kimi 发展的方式。这种模式在 AI 服务中相对较新，为用户提供了更多的参与感和定制化服务。这些都表明了 Kimi 在提供个性化服务方面有独到之处。

Kimi 的使用主要有两种方式，通过官网提供的在线对话界面，用户可以直接与 Kimi 进行对话交流；另一种方式是通过 API 接口，用户可以将 Kimi 集成到自己的应用程序中，实现更多个性化的功能。前者主要面向普通用户，后者则主要面向开发者和企业用户。两种方式本质上都是调用了 moonshot-v1 的模型文本生成服务。两种方式的区别在于，前者提供了一个直接的对话界面，用户可以直接与 Kimi 进行交互，而后者则是使用 API 接口，可将 Kimi 集成到自己的应用程序中，实现更多个性化的功能。

文本生成模型以 Token 为基本单位来处理文本。Token 代表常见的字符序列。例如，单个汉字“夔”可能会被分解为若干 Token 的组合，而像“中国”这样短且常见的短语则可能会使用单个 Token。大致来说，对于一段通常的中文文本，1 个 Token 大约相当于 1.5-2 个汉字。在模型内部，文本会被转换为 Token，然后被模型处理。模型输出的结果也是 Token，需要转换为文本后才能呈现给用户。需要注意的是，对于我们的文本模型，Input 和 Output 的总和长度不能超过模型的最大上下文长度。这一点在使用 API 调用模型时，需要特别注意。

根据模型的最大上下文长度，Moonshot 开放了以下三种不同的模型服务：

- moonshot-v1-8k：它是一个长度为 8k 的模型，适用于生成短文本。
- moonshot-v1-32k：它是一个长度为 32k 的模型，适用于生成长文本。
- moonshot-v1-128k：它是一个长度为 128k 的模型，适用于生成超长文本。

以上模型的区别在于它们的最大上下文长度，这个长度包括了输入消息和生成的输出，在效果上并没有什么区别。这个主要是为了方便用户选择合适的模型。

System prompt 是一种预设的输入，用于向语言模型提供执行任务所需的上下文信息或初始指令。从认知科学角度看，system prompt 影响模型的认知过程，决定了模型如何解释和响应外部信息。而提供好的 system prompt，可以帮助模型更好地理解任务需求，提高模型的效率和准确性。针对 Kimi，Moonshot 对 prompt 的编写提出了以下几点建议：

- 编写清晰的说明：向模型提供简洁明了的指令，避免含糊不清的表述。这有助于模型准确理解任务要求。例如，如果你需要模型帮助撰写一封正式的申请信，请明确指出文本的目的、风格和必须包含的要点。

- 包含详细背景：在请求中提供所有必要的背景信息和细节，以便模型生成更相关和准确的输出。例如，如果你询问关于特定市场的趋势分析，应提供该市场的基本信息和你感兴趣的特

定方面。

- **角色指定**：在请求中明确模型应扮演的角色，这有助于模型根据特定角色的属性和行为来调整其回答。例如，模拟一个客服机器人时，强调必须遵守的礼貌和专业的对话风格。

- **使用分隔符**：利用分隔符（如 XML 标签、三重引号等）来区分输入中的不同部分，特别是当处理结构化数据或需要区分不同文本块时。这有助于模型理解各部分的边界和上下文，从而更准确地处理信息。

- **明确完成任务的步骤**：如果任务需要多个步骤，明确列出这些步骤，使模型能够按照既定流程生成输出。例如，在请求模型进行数据分析时，可以按步骤指示数据清洗、分析和报告的编写。

- **提供输出示例**：向模型展示一些示例，特别是当任务涉及到模仿特定风格或格式时，这有助于模型快速学习并复制所需样式。例如，如果需要生成新闻稿，提供一两个典型的新闻稿样本作为参考。

- **指定期望输出长度**：根据需要，指定输出的长度，可以是字数、句子数、段落数等。这有助于模型控制生成内容的详细程度和结构，以满足特定的输出需求。

- **提供参考文本**：当问题需要特定信息时，提供参考文本供模型使用，这有助于生成更准确和相关的回答。例如，在讨论特定法律案件时，提供相关法规和案件背景资料可以增强模型的回答质量。

- **拆分复杂任务**：将复杂任务分解为更小、更易管理的部分，分别提供指令，有助于模型更有效地处理每个子任务。例如，在处理大量数据输入时，可以将任务分为数据收集、处理、分析和呈现四个部分。

- **对话摘要**：对于长对话，定期总结对话内容，以保持上下文的相关性和准确性。这种方法尤其适用于长时间的咨询或讨论会议，可以帮助模型维持对话的连贯性和焦点。

- **分块概括长文档**：对于长文档，采用分块概括的方法，递归构建完整的摘要，确保文档的每个部分都被准确理解和总结。这种方法在处理研究文章或复杂报告时尤为有效。

Kimi 的另一个独有优势是 **Context Caching**（上下文缓存）技术。**Context Caching** 是一种先进的数据管理技术，主要功能是预先存储可能会被频繁请求的大量数据或信息。这种技术的应用非常广泛，尤其适合于处理频繁且重复的数据请求场景。这种技术允许系统在用户再次请求相同信息时，能够直接从缓存中迅速提供所需数据，无需进行重新计算或重新从原始数据源检索，从而实现时间和资源的显著节省。

在实施 **Context Caching** 时，首先需要通过 API 接口创建缓存，明确指定需要存储的数据类型及内容，并设定一个合适的过期时间，以保证数据的实时性和有效性。缓存一旦建立，对这

些数据的任何请求都会首先进行缓存检查，如果缓存数据依然有效，则直接使用；若缓存失效，则必须重新生成并更新缓存。这一机制特别适用于那些需处理大量重复请求的应用程序，能够显著提高响应速度和系统整体性能。例如，Kimi API 小助手等 QA Bot 提供预设内容，面对的是大量标准化且重复的查询，通过 Context Caching 可以极大提高数据处理效率和响应速度。同样，用于上市公司信息披露的问答工具需要频繁访问更新不频繁的数据集，使用 Context Caching 可以减少对数据库的压力并提升查询速度。此外，像各类 Copilot Agent 这样定期分析大量代码或文档的应用，通过缓存常用数据可以优化性能和资源使用。

总的来说，Kimi 是一个功能强大的人工智能助手，具有长文本处理能力、高级语义理解、信息检索能力、多模态交互和个性化服务等特点。用户可以通过在线对话界面或 API 接口与 Kimi 进行交互，实现个性化的功能。Moonshot 还提供了一系列关于 prompt 编写的建议，帮助用户更好地利用 Kimi。此外，Kimi 还采用了 Context Caching 技术，提高了数据处理效率和响应速度。这些特性的引入都使得 Kimi 在大模型领域具有一定的竞争优势。

通义千问

通义是阿里云开发的一系列人工智能模型的总称，其核心产品之一便是通义千问，这是一个大型语言模型，旨在提供广泛的 AI 能力，成为工作、学习和生活的助手。通义千问的名称来源于《汉书》中的“天地之常经，古今之通义也”，寓意普遍适用的道理与法则，而“千问”则象征着对知识的无限追求和对问题的深入解答。

通义千问的模型架构基于 Transformer 框架，采用了开源大语言模型训练方法 LLaMA。不同于传统的 Transformer 框架，通义千问再模型架构做出了一些关键修改：

- 不受限的嵌入与输出投影：通义千问选择了独立的嵌入 (embedding) 和输出投影 (output projection) 权重，而不是捆绑输入嵌入和输出投影的权重。这种改动是以增加内存成本为代价，以获得更好的性能。

- 位置编码：通义千问采用 RoPE (Rotary Positional Embedding) 作为位置编码，并选择使用 FP32 精度的逆频率矩阵，以此提高模型的性能表现和精确度。

- 偏差处理：通义千问的大多数层 (layer) 中移除了偏差 (bias)，而 QKV 注意力层中添加了偏差，以增强模型的外推能力。

- 规范化：通义千问采用预规范化 (pre-normalization) 方法提高训练稳定性，并将传统层归一化方法替换为 RMSNorm。

- 激活函数：模型采用 SwiGLU 激活函数，缩小了前馈网络 (feed-forward network, FFN) 的维度，由原来隐藏大小的 4 倍变成隐藏大小的 83 倍。

- 注意力机制改进：通义千问利用了 NTK 感知插值、LogN-Scaling 和 Window attention 等免训练技术，在推理过程中扩展上下文长度，确保模型在处理长上下文时的计算效率和准确性。

- 训练细节：通义千问训练遵循自回归语言建模的标准方法，采用 Flash Attention 技术、AdamW 优化器和余弦学习率计划，使用 BFloat16 进行混合精度训练。

通义千问大模型是一种大规模预训练模型，旨在解决智能对话、知识图谱推理和其他多模态任务。凭借其深厚的自然语言理解和生成能力，以及在跨领域知识处理上的卓越表现，通义千问不仅成为了智能助手领域的佼佼者，还为企业和个人用户提供了前所未有的创新工具和服务。以下是通义千问的六大核心特色：

1. 强大的模型泛化能力

通义千问展现出卓越的泛化能力，不仅能够理解结构化和非结构化知识，还能在未见过的任务和情景中展现灵活性和适应性，无需特定任务的额外训练。这一特性使得模型能够在多模态理解和推理中表现出色，能够根据情境理解用户需求并给出精准反馈。

2. 高效的长文档处理功能

2024 年 3 月，通义千问宣布升级，向用户免费开放了处理长达 1000 万字的长文档功能，这极大地提升了其在文本分析、摘要生成、报告编写等场景下的实用性，满足了专业领域和学术研究的深层次需求。

3. 创新的个性化大模型定制

通义千问支持行业生态伙伴结合自身领域知识和场景需求，对大模型进行再训练和精调，打造出具有定制化、易集成、可微调、强安全特点的专属大模型，这为各行业提供了高度契合的智能化解决方案。

4. 高安全性和数据隔离

阿里云为通义千问提供了高安全性、高合规性的数据保护，通过服务器端加密机制确保数据安全。企业上传的行业数据仅用于自身模型训练，不参与通用版模型的训练，实现了数据的安全隔离，保护了企业数据的隐私和安全。

5. 智算 MaaS 服务

通义千问提供了智算 MaaS 服务，包括通义大模型系列和基于此的定制大模型，以及国内首个“模型即服务”开放平台魔搭社区。这一服务不仅涵盖了模型的训练、部署和管理，还提供了丰富的模型资源，助力企业和开发者快速构建 AI 应用。

6. 超大规模参数模型

通义千问不断推出参数规模庞大的模型版本，例如 Qwen1.5-110B 模型，参数量超过 1000 亿，展现了模型强大的表达能力和处理复杂任务的能力，为 AI 领域的研究和应用开辟了新的可能性。

基于通义千问大模型，阿里云衍生了一系列行业模型：通义灵码、通义智文、通义听悟、通义星尘、通义点金、通义晓蜜、通义仁心以及通义法睿。

- **通义灵码（编码助手）**：通义灵码，作为编码助手，能够辅助程序员进行代码编写、阅读、错误排查以及代码优化。它精通超过 200 种编程语言，包括 Java、Python、Go 等，能显著提升程序员的开发效率，帮助他们轻松应对代码难题。无论是专业程序员还是编程初学者，都能借助通义灵码提升技能。

- **通义智文（阅读助手）**：通义智文，一款基于通义大模型的免费 AI 阅读助手，专注于智能阅读网页、论文、图书和文档。它能迅速提取文本核心内容，生成简洁的摘要和概述，让用户快速把握信息要点，大幅提升阅读效率。此外，用户可以向通义智文提问，获取文章细节或澄清疑问。

- **通义听悟（工作学习助手）**：通义听悟针对音视频内容，提供转写、翻译、角色分离、摘要、章节概览、发言总结、PPT 提取等多功能服务。它解决了音视频资料难以快速消化的问题，支持标重点、做笔记，甚至可以对超长音视频进行自由问答，极大提高了信息处理的便捷性和效率。

- **通义星尘（个性化角色创作平台）**：通义星尘是一个个性化角色创作平台，通过大规模个性化对话数据训练，使模型具备拟人化、情感丰富和独特语言风格的能力。它能在保持通用能力的同时，根据具体情境和事件进行深度对话，适用于情感陪伴、游戏 NPC、IP 复刻等多样化场景。

- **通义点金（投研助手）**：通义点金聚焦于金融领域，能解读财务报告、分析金融事件、绘制图表和进行实时市场数据分析。无论是智能投研、文档分析，还是金融信息搜索，通义点金都能提供专业、精准的支持，助力金融从业者做出明智决策。

- **通义晓蜜（智能客服）**：通义晓蜜为企业提供全渠道联络中心与 AI 应用的整体解决方案，推动企业服务向信息化、智能化、多模态转型。它整合了客户服务、销售支持和内部沟通等功能，帮助企业构建高效、个性化的客户体验。

- **通义仁心（健康助手）**：通义仁心定位为健康助手，旨在为用户提供医疗健康咨询和服务。它结合医学知识和 AI 技术，为用户提供疾病预防、健康管理和医疗指导，是个人和家庭健康管理的好帮手。

- **通义法睿（法律顾问）**：通义法睿扮演法律顾问的角色，能够提供法律咨询、合同审核、案例分析等服务。它利用法律专业知识和大数据分析，帮助企业及个人规避风险，确保合法合规运营，是商业决策的有力助手。

这八大行业模型可以帮助人们写代码、读代码、查 BUG、优化代码等；短时间内获取长文本摘要和概述，掌握要点；对音频内容转写、翻译、角色分离、全文摘要、章节速览、发言总结、

PPT 提取等功能，并支持标重点、记笔记；可以解读财报研报，分析金融业事件，自动绘制图表表格，实时市场数据分析等。

讯飞星火

讯飞星火是由科大讯飞股份有限公司开发的认知大模型。其不仅在通用任务处理上表现出色，更在特定领域如学习辅助、数学推理等方面实现了重大突破，为科研机构、教育界和产业界带来了新的机遇。在讯飞星火的官网上，科大讯飞将其介绍为“讯飞星火，懂你的 AI 助手”。

在官网最新的讯飞星火 v4.0 的介绍中，讯飞星火有以下七大核心功能：

- **多模态感知与生成能力**：讯飞星火大模型具备强大的多模态理解与生成能力，能够解析图像、音频、视频等多种媒体形式。它不仅能准确识别并理解上传的图片内容，还能根据需求生成高质量的合成音频、视频以及虚拟人物视频，广泛应用于广告创意、教育娱乐、虚拟主播等多个领域，极大丰富了人机交互的体验。

- **代码智能辅助能力**：针对开发者群体，讯飞星火提供了从代码生成、解释到纠错和单元测试的全流程支持。它能智能生成代码片段，解释复杂代码结构，自动检测并修正语法错误，甚至为现有代码生成测试用例，显著提升了软件开发效率和代码质量，是程序员的得力助手。

- **创意内容创作能力**：讯飞星火擅长内容创作，能够根据不同场景生成吸引人的商业文案、营销策略、英文文章及新闻稿件。风格多样，准确把握受众心理，促进品牌传播和市场推广，是企业营销和媒体工作者的理想工具。

- **数学与逻辑推理能力**：在数学领域，讯飞星火能够解决从基础算术到高等数学的各种问题，包括方程求解、几何证明、微积分计算和概率统计分析。同时，它还具备逻辑推理能力，能进行思维推理、科学预测和常识判断，为教育、科研和决策提供强有力的支持。

- **语言理解与翻译能力**：凭借先进的自然语言处理技术，讯飞星火能实现多语言之间的精准翻译，进行文本摘要提取，检查语法错误，以及进行情感分析，帮助用户在跨语言环境中有效沟通，提高写作的专业性和准确性。

- **知识问答与智慧服务**：讯飞星火积累了广泛的知识库，能够回答涉及生活常识、工作技能、医学健康以及历史人文等多个领域的疑问。无论是寻求健康建议、职业规划指导，还是对历史文化的好奇，它都能提供详实且深入的答案，成为用户身边的智慧顾问。

与 kimi 类似，讯飞星火也提供了三种使用方式，分别是星火个人空间，星火智能体，星火 API 接入。

- **星火个人空间**：星火个人空间是科大讯飞在其最新版本的讯飞星火 V4.0 中引入的一个功能，它旨在提供一个个性化的 AI 助手服务。其支持上传包括文档、音频、视频在内的多种文件类型，实现一体化智能管理。它不仅能够对上传内容进行总结分析，还能基于已有资料智能生

成报告、演讲稿乃至小说，助力内容创作。此外，该空间提供翻译、PPT 自动生成及 AI 文本润色服务，让文件处理和二次创作文本变得更加便捷高效。

- **星火智能体**：星火智能体是讯飞星火针对特定场景和需求定制的 AI 助手，拥有超过 16000 种预设智能体，能直接应用于特定需求，无需额外配置。通过结构化创建和流程图式编排，用户可以快速搭建符合自身需求的 AI 助手，实现从角色设定到功能配置的个性化定制。星火智能体还支持轻应用开发，已集成智文、晓医等专业应用，为用户提供更佳的交互体验和针对性解决方案，满足多样化业务需求。

- **星火 API 接入**：讯飞星火 API 是一系列应用程序接口的集合，允许开发者将讯飞星火集成到他们自己的产品和服务中。星火 API 为企业和个人开发者提供了一站式 AI 大模型集成方案，支持 Web、iOS、Android、Windows、Linux 等多平台接入，涵盖语言理解、知识问答、代码编写等多种能力。其响应迅速，处理高效，能够轻松应对高并发场景，同时支持个性化参数调整和企业级定制服务。依托千亿级流量的公有云架构，星火 API 确保了服务的稳定性和安全性，通过等保三级认证，保护用户隐私，是构建智能应用的理想选择。

科大讯飞的讯飞星火大模型凭借其卓越的性能和广泛的适用性，在多个领域内实现了创新应用，从企业服务到智能硬件，从智慧政务到智慧生活，无处不在地展现出其强大的影响力。

- **企业服务**：讯飞星火通过整合企业内部知识库，实现了智能问答、营销策略生成、会议纪要整理等自动化功能，显著提升了企业运营效率，为企业决策提供了强有力的支持。

- **智能硬件**：讯飞星火使传统智能硬件如电视、VR 眼镜等设备具备了更强大的理解和交互能力，打破了基于规则的传统交互模式，让设备能够更好地满足用户在日常生活中的多样化需求。

- **智慧政务**：讯飞星火通过整合政务知识库，优化了公文写作、会议纪要生成等流程，显著提升了政府工作效率，同时确保了信息的准确性和一致性，促进了政务公开透明。

- **智慧金融**：在金融服务领域，讯飞星火实现了智能投资顾问、营销文案自动生成等功能，不仅降低了成本，还提高了营销效果，增强了客户体验。

- **智慧生活**：讯飞星火在日常生活中发挥了重要作用，它提供了实时文本翻译、星座咨询、内容创作等服务，极大地丰富了互联网生活，让用户享受到更加个性化和便捷的服务。

- **智慧医疗**：在医疗领域，讯飞星火与医院知识库结合，通过虚拟助手为患者提供智能问答和自助服务，减轻了医护人员的工作压力，提升了医疗服务质量和效率，为患者带来了更好的就医体验。

总的来说，讯飞星火是一个先进的认知模型，拥有包括多模态处理、代码辅助、内容创作、数学推理、语言翻译及个性化服务在内的七大核心功能。它通过星火个人空间、智能体和 API

接入等方式，广泛应用于企业服务、智能硬件、政务、金融、生活和医疗等领域，显著提升效率与用户体验，推动行业智能化升级。

文心一言

文心一言（ERNIE Bot）是百度研发的全新一代知识增强大语言模型，基于飞桨深度学习平台和文心知识增强大模型，持续从海量数据和大规模知识中融合学习，具备知识增强、检索增强和对话增强的技术特色，能够与人对话互动、回答问题、协助创作，高效便捷地帮助人们获取信息、知识和灵感。文心一言由文心大模型驱动，具备理解、生成、逻辑、记忆四大基础能力。

- **理解能力**：文心一言能够与用户进行自然而流畅的对话，能够理解潜台词、复杂句式、专业术语，根据用户的输入提供有意义的回应。

- **生成能力**：无论是文学创作、商业文案还是其他类型的文本生成，文心一言都能根据用户的需求快速生成高质量的内容。除了文字，文心一言还能生成图片、语音、视频等多种形式的内容，满足用户多样化的需求。

- **逻辑能力**：复杂的逻辑难题、困难的数学计算、重要的职业 / 生活决策统统能解决。文心一言通过分析用户的兴趣和历史数据，能为用户提供个性化的内容推荐，提升用户的浏览和搜索体验。

- **记忆能力**：不仅有高性能，更有好记性。N 轮对话过后，文心一言能够记住之前对话的重点。自文心一言推出以来，文心大模型持续迭代升级，文心大模型 4.0 的基本技术架构与文心大模型 3.0 和 3.5 版本一脉相承，在多个关键技术方向上取得进一步的创新突破。

- **训练环境更优**：文心大模型的基础模型训练离不开飞桨深度学习平台。百度技术团队将飞桨平台运行在万卡算力上，并通过集群基础设施和调度系统、飞桨框架的软硬协同优化，支持文心大模型的稳定高效训练。

- **数据质量更好**：技术团队建设了多维数据体系，形成了从数据挖掘、分析、合成到标注和评估的闭环，以充分释放数据价值，大幅提升模型效果。

- **训练算法更强**：百度技术团队通过打磨训练算法，“保证模型更好地与人类的判断和选择对齐”。在既往技术创新的基础上，技术团队基于有监督精调、偏好学习、强化学习等技术进行多阶段对齐，保证模型更符合人类价值观。同时，技术团队还使用可再生训练技术，通过增量式的参数调优，进一步节省训练资源和时间，加快模型迭代速度。

为提升大模型的准确性，文心大模型 4.0 在输入、输出两个阶段都运用了知识点增强，具体做法为在输入端对用户输入的问题进行理解，并拆解所需的知识点，然后在搜索引擎、知识

图谱、数据库中获取准确知识，最后把得到的知识组装进 prompt 送入大模型；输出端会对大模型的输出进行“反思”，从生成结果中拆解出知识点，然后利用搜索引擎、知识图谱、数据库及大模型本身进行确认，修正偏差。

当前文心大模型已升级至 4.0Turbo 版本。文心一言 4.0Turbo 的上下文输入长度从 4.0 版的 2K tokens 升级到了 128K tokens，能够同时阅读 100 个文件或网址，AI 生图分辨率也从 512×512 提升至 1024×1024。通过数据、基础模型、对齐技术、提示、知识增强、检索增强和对话增强等核心技术的持续创新，以及飞桨文心的联合优化，文心大模型 4.0Turbo 的速度更快、效果更好。

文心一言有五大应用场景：

- 学习成长：文心一言可用于专业技能学习，论文内容润色，论文降重和定量分析，帮助制定学习规划、职业规划，还能批改作业和模拟考试。例如百度基于文心一言对跳水队员在训练中的“技术指导”，升级后的中国国家跳水队 AI 辅助训练系统可以实时打分、拆解动作、分析过往成绩数据等，以开发个性化、专业化的训练方案。

- 生活助手：生活中，文心一言可以做科普答疑，推荐美食，提供制作方法，撰写美食文案，还能指导运动，制定健身计划、理财计划和旅游计划等。例如在旅行规划方面，文心一言能够帮助用户预订酒店、查找旅游信息、规划旅行路线，并提供实时旅行建议。

- 情感陪伴：文心一言能够做心理测试，还可闲聊陪伴，在社交活动中，可以协助用户撰写邮件、短信和社交媒体帖子，甚至帮助用户准备和练习对话。

- 休闲娱乐：文心一言可以推荐书单、片单，做文字游戏和智力游戏，生成段子笑话，协助用户规划娱乐活动，还可以生成互动式故事，让用户通过选择不同的情节走向，来体验不同的故事结局，提供个性化的娱乐体验。

- 职场提效：文心一言可以协助撰写文案，创作视频，做文字处理和数据分析以及代码的生成和测试。例如它可以帮助营销人员撰写广告文案、营销材料和社交媒体内容，提升数字营销的效果。

总的来说，文心一言作为百度推出的知识增强大语言模型，以其强大的文本生成和理解能力，具有广泛的应用场景，包括教育、客服、内容创作、商业、数字营销、旅行规划、健康咨询等。它能够以多种方式融入日常生活，提高生活质量和效率。

智谱清言

智谱清言是北京智谱华章科技有限公司推出的生成式 AI 助手，基于智谱 AI 自主研发的中英双语对话模型 ChatGLM2，经过万亿字符的文本与代码预训练，并采用有监督微调技术，以

通用对话的形式为用户提供智能化服务。主要功能有：

- 通用问答：能够回答用户的各类问题，涵盖众多领域，为用户提供实时、准确的信息和解决方案。
- 多轮对话：具备出色的对话能力，可以与用户进行自然、流畅的多轮对话，提供高效的沟通体验。
- 虚拟对话：能够根据用户的需求扮演不同角色，如专业人士、故事角色等，增强互动性和用户体验。
- 创意写作：可以为用户的各类创作需求提供灵感、内容框架以及高质量的文案等，提升写作效率和质量。
- 代码生成：能够使用多种编程语言进行开发和调试，可以帮助解释代码、解答编程问题、或提供编程建议。

目前智谱清言使用时可在 ChatGLM3 和 ChatGLM4 两个基座模型中进行选择，ChatGLM3 相比智谱清言刚上线时的 ChatGLM2，做了以下改进：

1. 更强大的性能

采用独创的多阶段增强预训练方法，更丰富的训练数据和更优的训练方案，使训练更为充分。评测显示，与 ChatGLM 二代模型相比，在 44 个中英文公开数据集测试中，ChatGLM3 在国内同尺寸模型中排名首位。其中，MMLU 提升 36%、CEval 提升 33%、GSM8K 提升 179%、BBH 提升 126%。

2. 瞄向 GPT-4V 的技术升级

瞄向 GPT-4V，ChatGLM3 实现了若干全新功能的迭代升级，包括：

- 多模态理解能力的 CogVLM，看图识语义，在 10 余个国际标准图文评测数据集上取得 SOTA；
- 代码增强模块 Code Interpreter 根据用户需求生成代码并执行，自动完成数据分析、文件处理等复杂任务；
- 网络搜索增强 WebGLM，接入搜索增强，能自动根据问题在互联网上查找相关资料并在回答时提供参考相关文献或文章链接。

使得 ChatGLM3 的语义能力与逻辑能力大大增强。

3. 全新的 Agent 智能体能力

ChatGLM3 集成了自研的 AgentTuning 技术，激活了模型智能体能力，尤其在智能规划和执行方面，相比于 ChatGLM 二代提升 1000%；开启国产大模型原生支持工具调用、代码执行、游

戏、数据库操作、知识图谱搜索与推理、操作系统等复杂场景。

4. Edge 端侧模型

ChatGLM3 推出可手机部署的端侧模型 ChatGLM3-1.5B 和 ChatGLM3-3B，支持多种手机以及车载平台，甚至支持移动平台上 CPU 芯片的推理，速度可达 20 tokens/s。精度方面 ChatGLM3-1.5B 和 ChatGLM3-3B 在公开 Benchmark 上与 ChatGLM2-6B 模型性能接近。

5. 更高效推理

基于最新的高效动态推理和显存优化技术，当前的推理框架在相同硬件、模型条件下，相较于目前最佳的开源实现，包括伯克利大学推出的 vLLM 以及 Hugging Face TGI 的最新版本，推理速度提升了 2-3 倍，推理成本降低一倍，每千 tokens 仅 0.5 分，成本最低。

全新升级的 ChatGLM3 赋能下，智谱清言可支持图像处理、数学计算、数据分析等使用场景。随着 WebGLM 大模型能力的加入，智谱清言现具有搜索增强能力。智谱清言可以帮助用户整理出相关问题的网上文献或文章链接，并整理出答案。CogVLM 模型则提高了智谱清言的中文图文理解能力，取得了接近 GPT-4V 的图片理解能力。它可以回答各种类型的视觉问题，并且可以完成复杂的目标检测，并打上标签，完成自动数据标注。

新一代基座大模型 GLM-4，整体性能相比 GLM3 全面提升 60%，逼近 GPT-4，支持更长上下文，更强的多模态，支持更快推理速度，更多并发，大大降低推理成本，同时 GLM-4 增强了智能体能力。

GLM-4 实现了自主根据用户意图，自动理解、规划复杂指令，自由调用网页浏览器、Code Interpreter 代码解释器和多模态文生图大模型，以完成复杂任务。简单来讲，即只需一个指令，GLM-4 会自动分析指令，结合上下文选择决定调用合适的工具。GLM-4 能够通过自动调用 python 解释器，进行复杂计算（例如复杂方程、微积分等），在 GSM8K、MATH、Math23K 等多个评测集上都取得了接近或同等 GPT-4 All Tools 的水平，同样 GLM-4 也可以完成文件处理、数据分析、图表绘制等复杂任务，支持处理 Excel、PDF、PPT 等格式文件；GLM-4 能够自行规划检索任务、自行选择信息源、自行与信息源交互，在准确率上能够达到 78.08%；GLM-4 能够根据用户提供的 Function 描述，自动选择所需 Function 并生成参数，以及根据 Function 的返回值生成回复，同时也支持一次输入进行多次 Function 调用，支持包含中文及特殊符号的 Function 名字。除了以上单项工具自动调用外，GLM-4 同样能够实现多工具自动调用，例如结合网页浏览、CogView3、代码解释器等调用方式。

智谱清言的智能体可由用户自己创建，满足人们个性化需求，更加丰富和自定义化，其多智能体协作系统允许用户在同一个对话框中调用多种不同的智能体，实现高效的协同工作，加

上其强大的多模态处理能力能够帮助解决生活和工作中的各种复杂问题。

1.2.2 国外大模型

ChatGPT

ChatGPT 是由 OpenAI 开发的一种大型语言模型，能够基于在预训练阶段所见的模式和统计规律，来生成回答，还能根据聊天的上下文进行互动，真正像人类一样来聊天交流，甚至能完成撰写论文、邮件、脚本、文案、翻译、代码等任务。其官方介绍为“Get answers. Find inspiration. Be more productive.”

从 2022 年 11 月 30 日发布开始，至今已有 GPT-1、GPT-2、GPT-3、GPT-3.5、GPT-3.5 Turbo、GPT-4、GPT-4 Turbo 等版本，目前性能较好且被广泛使用的为 GPT-3.5、GPT-3.5 Turbo、GPT-4、GPT-4 Turbo 四个版本，功能分别如下：

• GPT-3.5：

- (1) 文本生成：能够生成自然语言文本，如文章、故事、新闻、博客等。
- (2) 对话系统：进行对话交流，提供实时的互动对话体验。
- (3) 问题回答：回答各种问题，包括知识问答、技术问题等。
- (4) 语言翻译：将文本从一种语言翻译成另一种语言。
- (5) 文本总结：总结长文本的主要内容和关键点。
- (6) 内容创作：辅助进行创意写作、广告文案、社交媒体内容生成等。
- (7) 信息提取：从文本中提取特定信息，如日期、事件、人物等。

总体来说，GPT-3.5 能够进行复杂的文本生成和理解，但在某些情况下可能会产生不准确或不相关的回答，适用于需要较好文本生成的任务，如写作、内容创作、代码生成等。

• GPT-3.5 Turbo：

- (1) 快速文本生成：在短时间内生成高质量的自然语言文本。
- (2) 实时对话：提供快速响应的对话系统，适用于在线客服和聊天机器人。
- (3) 即时问题回答：快速回答用户提出的各种问题。
- (4) 高效语言翻译：在保持翻译质量的同时，提高翻译速度。
- (5) 快速文本总结：迅速总结长文本的主要内容。
- (6) 高效内容创作：快速生成创意写作、广告文案和社交媒体内容。
- (7) 即时信息提取：快速从文本中提取所需信息。

GPT-3.5 Turbo 在保持与 GPT-3.5 相同参数规模的基础上，通过优化算法和架构，提升了响应速度和成本效率，适用于需要快速响应和成本控制的应用场景，如实时聊天机器人、交互式应用等。

• GPT-4 :

- (1) 高级文本生成：生成高质量、复杂的自然语言文本，适用于专业文档、技术文章等。
- (2) 复杂对话系统：处理复杂的对话，提供连贯和深入的互动交流。
- (3) 精确问题回答：高准确度地回答专业和技术性问题。
- (4) 高级语言翻译：提供高质量的多语言翻译，适用于专业翻译需求。
- (5) 详细文本总结：生成详细和准确的文本摘要。
- (6) 高级内容创作：适用于创作复杂的内容，如小说、剧本和学术论文。
- (7) 深度信息提取：从复杂文本中提取多层次的信息，如详细事件分析。

GPT-4 大幅提升了语言理解和生成的能力，能够处理更复杂和更细致的任务。在一些基准测试中表现出超越人类的能力。适用于高精度、高复杂度的任务，如高级对话系统、高精度翻译、复杂文本分析等。

• GPT-4 Turbo :

- (1) 快速高级文本生成：在更短时间内生成高质量、复杂的自然语言文本。
- (2) 高效复杂对话系统：提供快速响应的复杂对话处理，适用于高级在线客服和对话机器人。
- (3) 即时精确问题回答：快速且准确地回答各种专业和技术问题。
- (4) 高效高级语言翻译：在保持高质量的前提下，提高多语言翻译速度。
- (5) 快速详细文本总结：迅速生成详细和准确的文本摘要。
- (6) 快速高级内容创作：高效生成复杂的创作内容，如小说、剧本、学术论文等。
- (7) 即时深度信息提取：迅速从复杂文本中提取多层次信息，适用于实时数据分析。

GPT-4 Turbo 在保持 GPT-4 高性能的同时，进一步提升了响应速度和成本效率，适合大规模部署。适用于需要高性能和高效能的应用场景，如大规模的企业解决方案、高并发的实时系统等。

• 主要性能差别：

- (1) 性能和能力：

GPT-4 Turbo > GPT-4 > GPT-3.5 Turbo > GPT-3.5

每个新版本在语言理解和生成能力上都有显著提升，尤其是 GPT-4 Turbo 在处理复杂任务和细节方面表现出色。

- (2) 响应速度和成本效率：

GPT-3.5 Turbo 和 GPT-4 Turbo 在各自版本的基础上进行了优化，显著提升了响应速度和成本效率。

(3) 应用场景：

GPT-3.5 和 GPT-4 适用于高质量文本生成和理解任务。

Turbo 版本适用于需要快速响应和高效能的场景，如实时应用和大规模部署。

总的来说，选择哪个版本取决于具体的应用需求、性能要求和成本考虑。

DALL·E

DALL·E 是一个基于 GPT 架构的生成式预训练模型，专门用于处理图像生成任务。

• 介绍：

Images API 提供了三种与图像交互的方法：

- (1) 基于文本提示创建全新图像 (适用于 DALL·E 3 和 DALL·E 2)
- (2) 基于新的文本提示编辑现有图像的某些区域 (仅适用于 DALL·E 2)
- (3) 创建现有图像的变体 (仅适用于 DALL·E 2)

• 用法：

- (1) 基于文本提示创建全新图像：

使用 DALL·E3 时，生成的图像可以有三种尺寸：1024×1024 像素、1024×1792 像素或 1792×1024 像素。

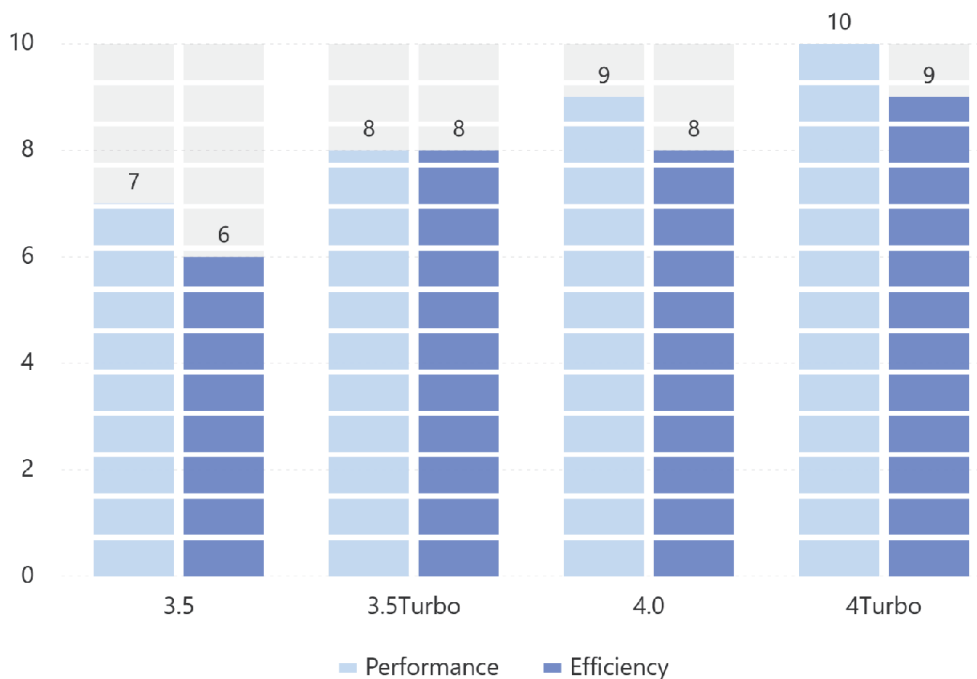


图 1.16: GPT 各版本性能对比

默认情况下，DALL·E 3 会生成标准质量的图像。但是，如果你需要更高的细节度，可以通过设置 `quality` 参数为“hd”来请求增强的图像质量。需要注意的是，高质量的图像可能需要更长的时间来生成。

使用 DALL·E 3 时，每次可以请求生成 1 张图像（可以通过并行请求来请求更多图像）。

而使用 DALL·E 2 时，可以通过指定参数 `n` 来一次性请求最多 10 张图像。

下面是一个 Python 脚本的示例，展示了如何从 OpenAI API 请求图像：

```
from openai import OpenAI
client = OpenAI()

response = client.images.generate(
    model = "dall-e-3",
    prompt = "a white siamese cat",
    size = "1024x1024",
    quality = "standard",
    n = 1
)

image_url = response.data[0].url
```

DALL·E 3 发布后，模型会自动接收并根据安全和细节丰富度的要求，对提供的默认提示进行重写（通常更详尽的提示能够产生更高质量的图像）。目前无法关闭这项自动重写功能。然而，可以通过在提示中加入特定的语句，来获得更符合需求的图像输出，例如：`I NEED to test how the tool works with extremely simple prompts. DO NOT add any detail, just use it AS-IS`。更新后的提示，可以在数据响应对象的 `revised_prompt` 字段中查看。

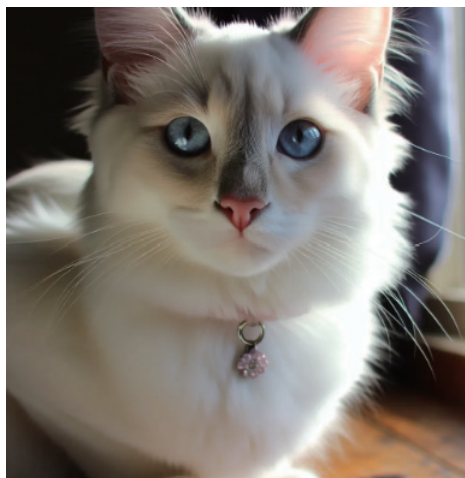


图 1.17: a white Siamese cat

(2) 基于新的文本提示编辑现有图像的某些区域：

DALL·E2 的图像编辑端点使用户能够通过指定要编辑的区域（使用 mask）和提供完整的描述性提示，来指导 AI 模型生成或修改图像。

```
from openai import OpenAI

response = client.images.edit(
    model = "dall-e-2",
    image = open("sunlit_lounge.png", "rb"),
    mask = open("mask.png", "rb"),
    prompt = "A sunlit indoor lounge area with a pool containing a flamingo",
    n = 1,
    size = "1024x1024"
)
image_url = response.data[0].url
```

上传的图像和遮罩都必须是边长相等的正方形 PNG 格式，且文件大小不超过 4MB。遮罩中不透明区域在生成图像输出时不会被使用，因此它们并不需要与原始图像的相应部分完全对应。

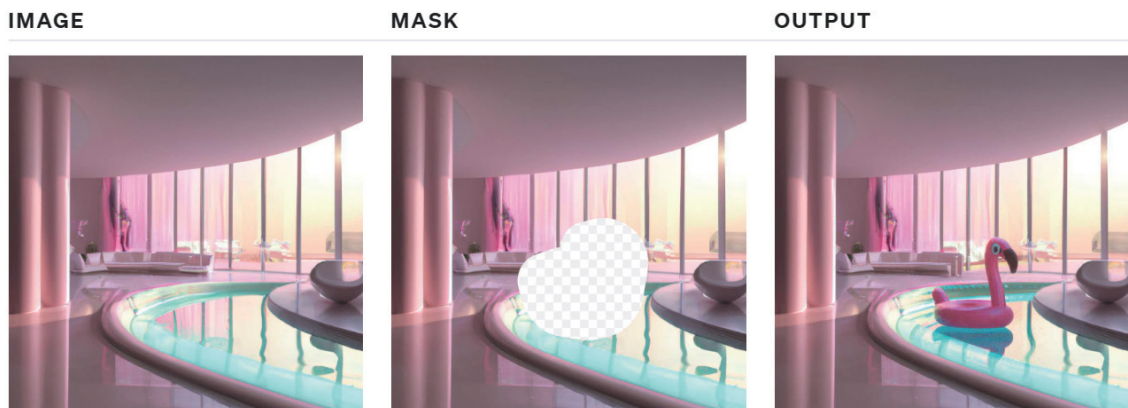


图 1.18: a sunlit indoor lounge area with a pool containing a flamingo

(3) 创建现有图像的变体：

图像变体端点允许生成给定图像的一个变体，以下是生成相似的新图片的 Python 代码方法。

```
from openai import OpenAI

response = client.images.create_variation(
    model = "dall-e-2",
    image = open("corgi_and_cat_paw.png", "rb"),
    n = 1,
    size = "1024x1024"
)
image_url = response.data[0].url
```

输入图像必须是小于 4MB 大小的正方形 PNG 图像。

IMAGE

OUTPUT



图 1.19: Variation

- 相关语言技巧：

- (1) 利用内存图像数据：

在某些情况下，图像数据可能已经加载到内存中，而不是从磁盘读取。前面 Python 示例中使用 `open` 函数从磁盘读取图像数据，但当图像数据在内存中时，可以使用 BytesIO 对象来处理这些数据。

```
from io import BytesIO
from openai import OpenAI
client = OpenAI()
# This is the BytesIO object that contains your image data
byte_stream: BytesIO = [your image data]
byte_array = byte_stream.getvalue()
response = client.images.create_variation(
    image = byte_array,
    n = 1,
    model = "dall-e-2",
    size = "1024x1024"
)
```

- (2) 图像数据处理：

在将图像传递给 API 之前，可能需要对它们执行一些操作。以下是一个使用 PIL 库调整图像大小的示例：

```

from io import BytesIO
from PIL import Image
from openai import OpenAI
client = OpenAI()
# Read the image file from disk and resize it
image = Image.open("image.png")
width, height = 256, 256
image = image.resize((width, height))

# Convert the image to a BytesIO object
byte_stream = BytesIO()
image.save(byte_stream, format = 'PNG')
byte_array = byte_stream.getvalue()

response = client.images.create_variation(
    image = byte_array,
    n = 1,
    model = "dall-e-2",
    size = "1024x1024"
)

```

(3) 错误处理：

API 请求可能会由于无效输入、频率限制或其他问题而返回错误。这些错误可以通过 `try...except` 语句来处理，并且错误详情可以在 `e.error` 中找到：

```

from openai import OpenAI
client = OpenAI()
try:
    response = client.images.create_variation(
        image = open("image_edit_mask.png", "rb"),
        n = 1,
        model = "dall-e-2",
        size = "1024x1024"
    )
except openai.OpenAIError as e:
    print(e.http_status)
    print(e.error)

```

TTS

TTS (Text-to-Speech) 是指一种将文本转为音频的技术。大模型中的 TTS 技术通过深度学习和神经网络实现，通常包含以下几个主要组件：

1. 文本处理

- 文本规范化：将输入的文本标准化，例如将数字、符号转换为其读法。
- 语言模型：分析文本的语法和语义，预测每个词的发音和语调。

2. 声学模型

- 语音特征提取：将文本转换为声学特征序列，这些特征描述了音频的基本元素。
- 神经网络模型：使用深度神经网络（如 LSTM、Transformer）来预测音素和其时长等声学特征。典型的模型有 Tacotron 和 Tacotron 2。

3. 声码器

- 从声学特征到语音：将声学特征转换为实际的音频波形。常见的声码器包括 WaveNet、WaveGlow、MelGAN 等，它们使用生成模型来生成高质量的语音。

4. 后处理

- 音频处理：对生成的音频进行后处理，以提高音质，包括降噪、去除伪影等。

应用领域

目前大模型的 TTS 技术已经广泛应用在了多个领域，主要有以下几个方面：

1. 语音助手

TTS 技术在语音助手中的应用极为广泛，这些助手通过语音与用户互动，提供信息和服务。

- Apple Siri：利用 TTS 技术回答问题、设置提醒、发送消息等。
- Google Assistant：提供导航、播放音乐、控制智能家居设备等功能。
- Amazon Alexa：通过语音合成技术，实现家庭自动化、购物、天气预报等服务。

2. 辅助技术

TTS 技术帮助特殊群体更好地获取信息和进行沟通。

- 视力障碍者：TTS 技术可以将书籍、网站内容和其他文本信息转化为语音，帮助视力障碍者获取信息。例如，JAWS（Job Access With Speech）屏幕阅读器利用 TTS 技术读取屏幕内容。
- 阅读障碍者：利用 TTS 技术，阅读障碍者可以将文本转化为语音，从而更容易理解和学习。例如，Kurzweil 3000 软件为有阅读障碍的学生提供文本朗读功能。

- 语言障碍者：TTS 设备和应用帮助语言障碍者表达自己，如使用 AAC（辅助与替代通信）设备进行沟通。

3. 内容创作

TTS 技术在内容创作中的应用日益广泛，特别是在媒体和娱乐行业。

- 视频配音：许多视频创作者利用 TTS 技术为视频添加配音，节省时间和成本，同时提供多语言支持。例如，YouTube 上的自动生成字幕功能。
- 播客制作：播客制作人可以使用 TTS 技术生成高质量的语音内容，减少录音和编辑的时间。

- 有声书：TTS 技术使得将书籍快速转化为有声书成为可能，拓宽了出版物的受众范围。例如，Audible 等平台提供的有声书。

4. 客户服务

TTS 技术在教育领域的应用有助于提升教学质量和学生的学习体验。

- 在线学习平台：在线课程和教育应用利用 TTS 技术为学生提供文本朗读服务，帮助他们更好地理解和记忆学习材料。例如，Khan Academy 和 Coursera 等平台。

- 语言学习：TTS 技术帮助学生练习听力和口语，通过合成语音进行发音对比和练习。例如，Duolingo 等语言学习应用。

5. 智能家居

智能家居设备利用 TTS 技术提升用户体验，提供便利和舒适的生活环境。

- 智能音箱：如 Amazon Echo 和 Google Home，通过 TTS 技术提供天气预报、新闻播报、日程提醒等服务。

- 家电控制：智能家电通过语音合成与用户互动，提供操作指引和状态反馈。例如，智能冰箱可以通过 TTS 技术提醒用户食物的保质期。

大模型实例

目前已经有大量基于 TTS 技术生成的各种大语言模型，他们涵盖了各个领域，本书将简单介绍其中的两种模型。

1. ChatMusician

ChatMusician 是一款专门可以用来进行音乐创作的 AI 工具。它能够帮助音乐家和音乐爱好者们在创作过程中显著地提高效率。ChatMusician 是基于 LLaMA2 模型进行持续预训练和微调的它将音乐视为一种全新的语言。该模型在生成过程中主要用到的技术包括：

- 自然语言处理（NLP）

歌词生成：利用 NLP 模型，如 GPT-3 或更高级的模型，ChatMusician 能够理解和生成符合特定主题、情感和韵律的歌词。这些模型通过大量的歌词和文学作品训练，能够产生富有创意且连贯的文本。

- 生成对抗网络（GANs）

旋律生成：GANs 被用于生成新的旋律。生成器（Generator）和判别器（Discriminator）相互对抗，使得生成的旋律逐渐逼近真实音乐片段的质量。这种方法特别适用于需要高创意和多样化输出的任务。

- 音乐信息检索（MIR）

音乐分析：MIR 技术用于从音频数据中提取音乐特征，如音高、节奏、和声等。通过这些特征，ChatMusician 能够对音乐作品进行深入分析和理解。

2. SongComposer

SongComposer 是一款专为歌曲创作而设计的一个大语言模型。该模型设计了一种新颖的元组设计来格式化旋律中的歌词和三个音符属性（音高，持续时间和休息持续时间），这种设计保证了大模型对音乐符号的正确理解，并实现了歌词和旋律之间的精确对齐。Song-Composer 在一个大规模的歌曲预训练数据集 SongCompose-PT 上经过充分的预训练后使用 10000 个精心制作的 QA 对来增强 LLMs 的指令遵循能力，从而解决各种任务。它主要使用的技术有：

- 变分自编码器（VAE）：

旋律创作：VAE 被用于生成新的旋律。通过学习音乐数据的潜在空间，VAE 可以生成具有一定风格和特征的旋律。这种方法在处理连续数据生成方面表现出色。

- 长短期记忆网络（LSTM）：

和弦进程生成：LSTM 网络是处理时间序列数据的常用方法，特别适用于音乐生成。

LSTM 能够记住长时间跨度的信息，因此在生成和弦进程时能够保持音乐的连贯性和一致性。

- 注意力机制（Attention Mechanism）：

编曲辅助：Attention 机制允许模型在生成音乐时关注输入的不同部分，从而生成更符合整体风格和结构的编曲方案。这种机制在增强模型生成质量方面起到了重要作用。

- 卷积神经网络（CNN）：

音乐分析和风格识别：CNN 用于分析音频数据的频谱图，通过识别特定的音频特征，CNN 能够准确地识别音乐风格和结构，提供相应的创作建议。

操作简介

OpenAI API 提供了一个可以直接通过调用生成音频的简单例子：

```
from pathlib import Path
from openai import OpenAI
client = OpenAI()

speech_file_path = Path(__file__).parent / "speech.mp3"
response = client.audio.speech.create(
    model = "tts-1",
    voice = "alloy",
    input = "Today is a wonderful day to build something people love!"
)
response.stream_to_file(speech_file_path)
```

通过这个 API 生成的文件默认格式主要为“MP3”，但其他格式如“opus”、“aac”、“flac”和“pcm”也可用。

Whisper

Whisper (speech-to-text) 是一种将语音转换为文本的先进的自然语言处理技术。它基于深度学习和神经网络技术，通过训练模型来识别和理解语音信号中的语言内容，并将其转换为可读的文本形式。本节将详细介绍大语言模型 Whisper 方法的原理、应用领域、优势以及挑战。

Whisper 方法原理

Whisper 是一个自动语音识别 (ASR) 系统，它基于从网络收集的 680,000 小时多语言和多任务监督数据进行训练。研究表明，使用如此庞大且多样化的数据集可以提高对口音、背景噪音和技术语言的鲁棒性。此外，它还可以转录多种语言，并将这些语言翻译成英语。

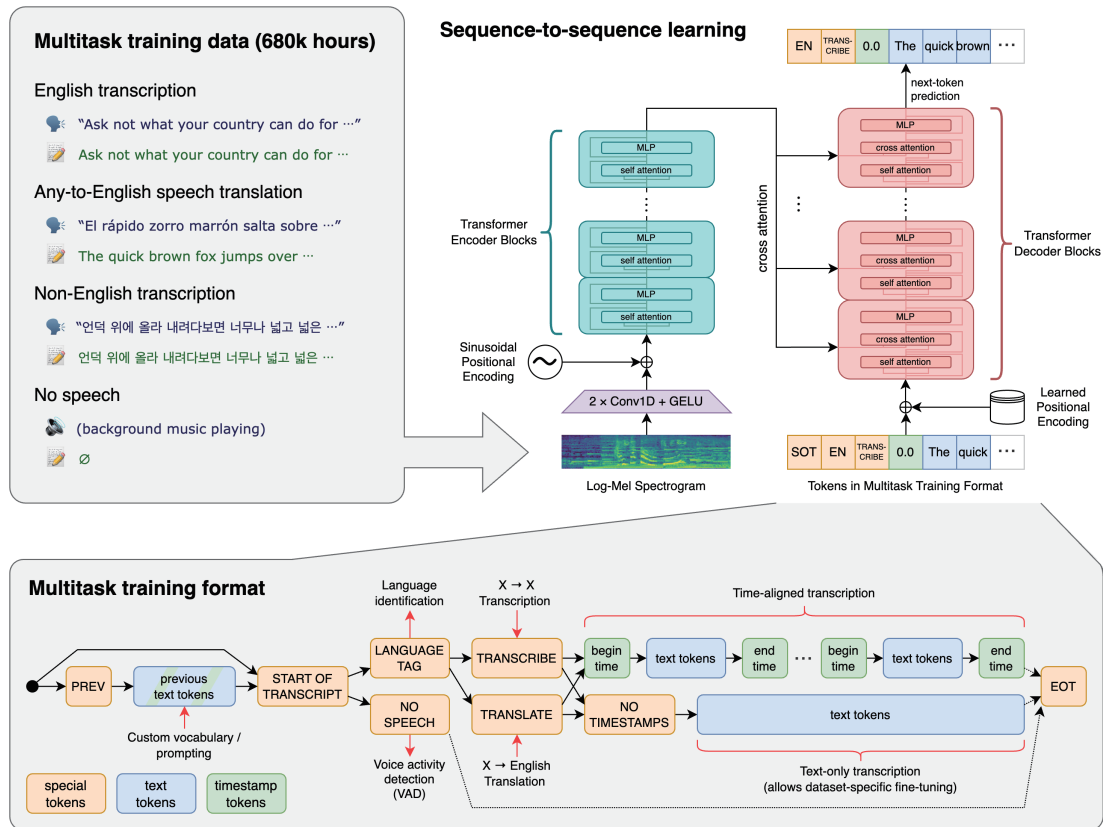


图 1.20: Whisper 方法流程图

1. 语音信号的特征提取：从语音输入中提取相关的声学特征，如声谱图 (spectrogram)、梅尔频谱特征等。这些特征提供了语音信号的频谱和能量分布信息。将输入音频分成 30 秒的片段，转换成对数梅尔频谱图，然后传入编码器。

2. 深度学习模型的应用：解码器经过训练可以预测相应的文本字幕，并与特殊标记混合，

指导单个模型执行语言识别、短语级时间戳、多语言语音转录和英语语音翻译等任务。Whisper 架构是一种简单的端到端方法，以编码器 - 解码器 Transformer 的形式实现。采用变换器架构的深度学习模型，如 Transformer 模型。这些模型通过多层自注意力机制（self-attention）和前馈神经网络层（feedforward neural networks），能够处理输入序列并生成对应的输出序列，即将语音信号序列映射到文本序列。

3. 模型训练和微调：在大规模语料库上对模型进行预训练，然后通过特定任务的微调（finetuning），使其适应特定的语音识别任务。微调过程通常涉及使用标记的语音 - 文本配对数据进行模型参数调整，以提高准确性和适应性。

其他现有方法经常使用较小、更紧密配对的音频文本训练数据集或使用广泛但无监督的音频预训练。由于 Whisper 是在大型多样化数据集上进行训练的，并且未针对任何特定数据集进行微调，因此它无法击败专门针对 LibriSpeech 性能的模型，LibriSpeech 是语音识别领域著名的竞争基准。然而，当我们在许多不同的数据集上测量 Whisper 的零样本性能时，我们发现它比那些模型更加稳健，错误率降低了 50%。

4. 后处理和优化：生成的文本序列可能需要进一步的后处理和优化步骤，例如语言模型的纠错和文本流畅性的增强。

语音识别合成系统

语音合成系统如下图所示：

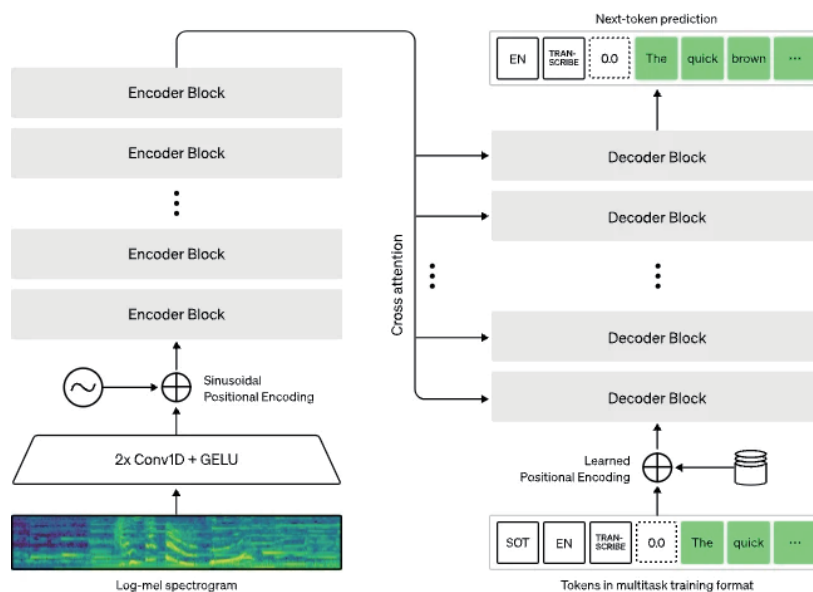


图 1.21: ASR 系统架构

输入部分：底部是一个对数梅尔频谱图 (Log-mel spectrogram)，这是语音信号的一种频率表

示。频谱图通过 2 层卷积神经网络 (2x Conv1D + GELU) 进行处理。处理后的特征与正弦位置编码 (Sinusoidal Positional Encoding) 相加。

编码器部分：包含多个编码器块 (Encoder Block)，这些块可能包含自注意力机制和前馈神经网络。

解码器部分：有多个解码器块 (Decoder Block)。解码器通过交叉注意力 (Cross-attention) 机制与编码器输出相连。

输出部分：解码器的输出与学习到的位置编码 (Learned Positional Encoding) 相加。最终输出是下一个标记的预测，例如 “The quick brown...”。

多任务训练：图片右侧底部显示了输入标记的格式，包括 SOT（序列开始标记）、EN（可能是语言代码）、TRANSCRIPT（转录文本）等。

ASR 系统语音识别和翻译过程

下图展示了一个语音识别或转录系统的界面，大致分为四种类型：

- 英语转录：例如输入英文输出英文
- 任意语言到英语的语音翻译：例如输出西班牙语输出时翻译成英语
- 非英语转录：例如输入韩语输出韩语
- 无语音：例如输入背景音乐播放中，输出一个空白圆圈，表示可能没有检测到语音

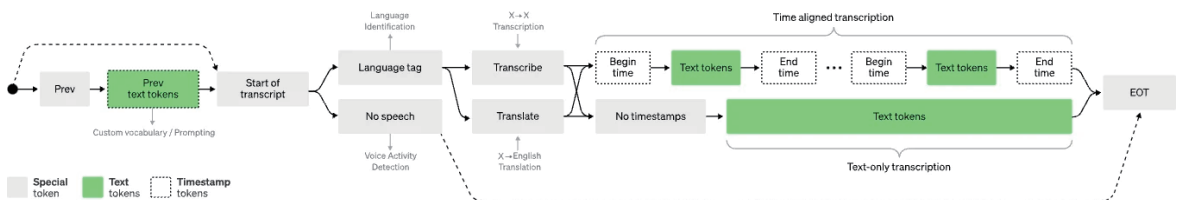


图 1.22: ASR 系统语音识别和翻译过程

ASR 训练数据输入和输出

Whisper 的音频数据集中约有三分之一是非英语的，它交替被赋予转录为原始语言或翻译成英语的任务。我们发现这种方法在学习语音到文本的翻译方面特别有效，并且在 CoVoST2 到英语翻译零样本上优于监督 SOTA。

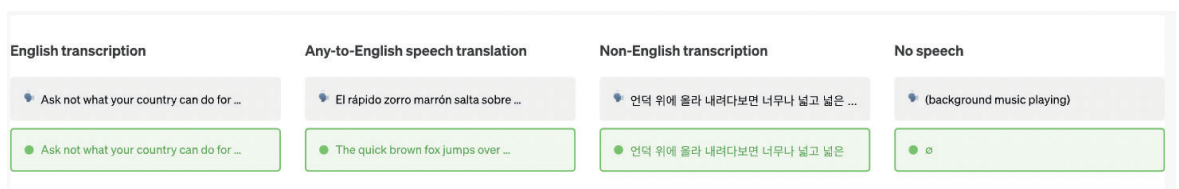


图 1.23: ASR 训练数据输入和输出

Whisper 模型性能

Whisper 的表现因语言而异。下图展示了部分样本显示了按语言划分的 large-v3 和 large-v2 模型的性能分解，使用 WER（单词错误率）或 CER（字符错误率）在 Common Voice 15 和 Fleurs 数据集上评估。

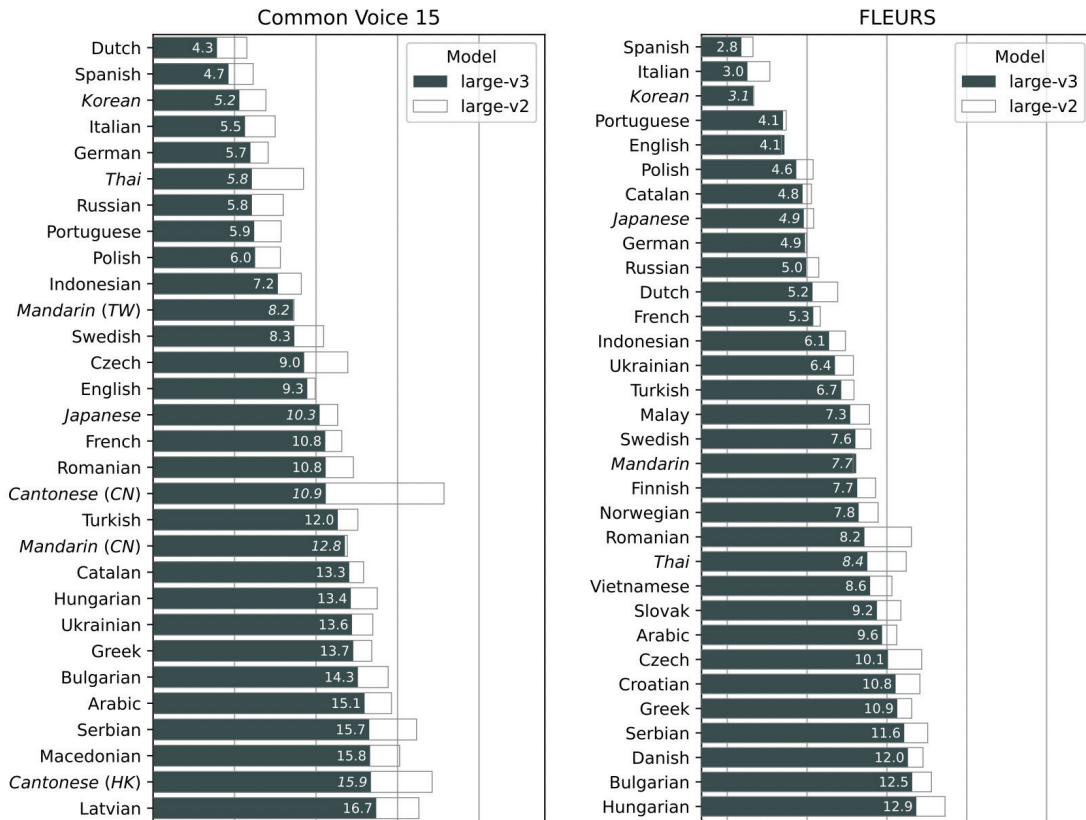


图 1.24: Whisper 在不同数据集上模型性能

Whisper 方法的应用领域

大语言模型 Whisper 方法广泛应用于以下领域：

- 实时语音识别和转录：支持语音到文本的实时转换，如智能手机的语音助手和语音搜索功能。
- 语音命令和控制：用于智能家居、车载系统等领域，用户可以通过语音命令控制设备或系统。
- 语音翻译和跨语言交流：结合机器翻译技术，实现语音到另一语言文本的快速转换，促进国际交流和跨文化沟通。
- 辅助听力技术：为听力受损者提供实时的语音转文本服务，帮助他们理解和参与日常对话和活动。

- 大规模语音数据处理：如电话录音、会议记录等大量语音数据的自动转录和分析，加速信息管理和数据挖掘的过程。

Whisper 方法的优势和挑战

优势

- 高度准确：基于大规模数据的深度学习模型能够显著提升语音识别和转文本的准确性。
- 语境理解：模型能够捕捉语音信号中的语义和上下文信息，提高了文本输出的质量和准确性。
- 多语言支持：支持多种语言和口音的转换，具备一定的通用性和适应性。

挑战

- 环境噪声干扰：嘈杂环境下的语音信号识别准确度较低，需要进一步的信号处理和模型优化。
- 口音和语速变化：不同口音和语速对语音识别结果有较大影响，需要增强模型的鲁棒性和泛化能力。
- 隐私和安全问题：处理和存储大量语音数据涉及用户隐私和数据安全问题，需要严格的数据保护措施和法律规定的遵守。

Whisper 方法的操作简介

OpenAI 提供了一个简单例子，将转录的音频文件和音频转录所需的输出文件格式作为输入。Whisper 模型一方面可以将音频转录成该音频所采用的任何语言，另一方面，通过 `translation.text` 函数可将音频翻译成英文输出。文件上传目前限制为 25MB，支持以下输入文件类型：`mp3`、`mp4`、`mpeg`、`mpga`、`m4a`、`wav`、`webm`。

语音转文本转录实例：

```
from openai import OpenAI
client = OpenAI()
audio_file = open("/path/to/file/audio.mp3", "rb")
transcription = client.audio.transcriptions.create(
    model = "whisper-1",
    file = audio_file
)
print(transcription.text)
```

语音转文本翻译实例：

翻译 API 将任何支持语言的音频文件作为输入，并在必要时将音频转录为英语。例如输入的是德语音频，输出的将是英文文本。

```

from openai import OpenAI
client = OpenAI()
audio_file = open("/path/to/file/german.mp3", "rb")
translation = client.audio.translations.create(
    model = "whisper-1",
    file = audio_file
)
print(translation.text)

```

大文件实例：

默认情况下，Whisper API 仅支持小于 25 MB 的文件。如果您的音频文件大于 25MB，则需要将其拆分为 25MB 以及更小的块，或使用压缩音频格式。为了获得最佳性能，建议避免在句子中间拆分音频，因为这可能会导致某些上下文丢失。处理此问题的一种方法是使用 PyDub 开源 Python 包来分割音频：

```

from pydub import AudioSegment
song = AudioSegment.from_mp3("good_morning.mp3")

#PyDub handles time in milliseconds
ten_minutes = 10 * 60 * 1000

first_10_minutes = song[:ten_minutes]
first_10_minutes.export("good_morning_10.mp3", format = "mp3")

```

提高模型稳定性实例：

使用 Whisper 时面临的最常见挑战之一是模型通常无法识别不常见的单词或首字母缩略词。为了解决这个问题，通过以下技术可以在这些情况下提高 Whisper 的可靠性：

使用 prompt 参数：使用可选的提示参数来传递正确拼写的字典。由于没有使用指令跟踪技术进行训练，Whisper 的运行方式更像一个基础 GPT 模型。注意，Whisper 仅考虑提示的前 244 个标记。

```

from openai import OpenAI
client = OpenAI()
audio_file = open("/path/to/file/speech.mp3", "rb")
transcription = client.audio.transcriptions.create(
    model = "whisper-1",
    file = audio_file,
    response_format = "text",
    prompt = "ZyntriQix, Digique Plus, CynapseFive, VortiQore V8, EchoNix Array, OrbitallLink
              Seven"
)
print(transcription.text)

```

使用 GPT-4 进行后处理：我们首先通过 system prompt 变量为 GPT-4 提供指令。与之前对 prompt 参数所做的类似，我们可以定义公司和产品名称。GPT-4 成功纠正了转录中的许多拼写

错误。此方法可能比使用 Whisper 的提示参数更具可扩展性，并且更可靠。

```

system_prompt = "You are a helpful assistant for the company ZyntriQix. Your task is to
                  correct any mistake."

def generate_corrected_transcript(temperature, system_prompt, audio_file):
    response = cclient.chat.completions.create(
        model = "gpt-4o",
        temperature = temperature,
        messages = [
            {
                "role": "system",
                "content": system_prompt
            },
            {
                "role": "user",
                "content": transcribe(audio_file, "")
            }
        ]
    )
    return completion.choices[0].message.content

corrected_text = generate_corrected_transcript(0, system_prompt, fake_company_filepath)

```

Embeddings

嵌入的定义和应用

嵌入是一种将高维数据（如单词、句子、图像等）转换为低维向量的方法。这些低维向量保留了原始数据的重要特征，同时使得计算更为高效。嵌入的目标是将相似的对象映射到相近的向量，从而在低维空间中保留原始数据的结构和关系。

OpenAI 的文本嵌入用于衡量文本字符串之间的相关性。嵌入通常用于以下几种场景：

- 搜索（根据查询字符串的相关性对结果进行排序）

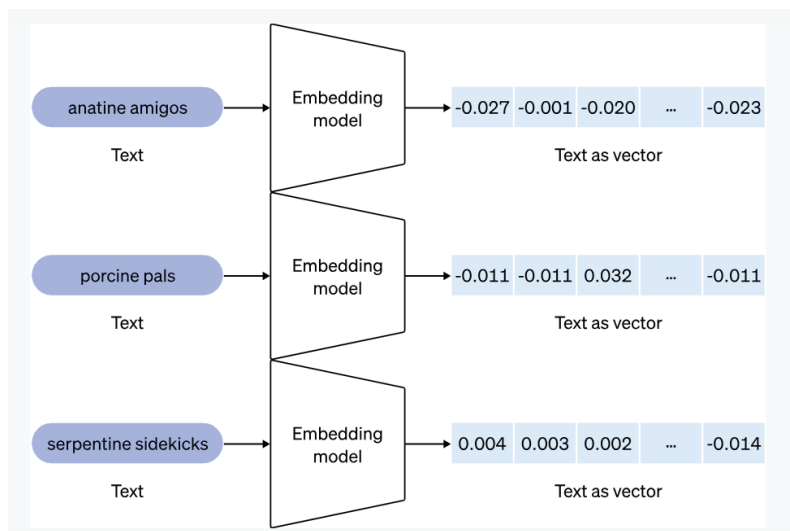


图 1.25: 文本嵌入

- 聚类（按相似性对文本字符串进行分组）
- 推荐（推荐与相关文本字符串相符的项目）
- 异常检测（识别相关性较低的异常值）
- 多样性测量（分析相似性分布）
- 分类（根据最相似的标签对文本字符串进行分类）

嵌入是一个浮点数向量（列表）。两个向量之间的距离衡量它们的相关性。距离小表示高度相关，距离大表示低相关。

截至目前，OpenAI 推出了两个强大的第三代嵌入模型：一种是更小、更高效的 `text-embedding-3-small` 模型，另一种是更大、更强大的 `text-embedding-3-large model` 模型。除此之外，早期模型如 `text-embedding-ada-002` 仍然可以继续使用。

新的文本嵌入模型

维度和大小：

- `text-embedding-3-small`：这个模型生成的嵌入默认大小为 1,536 维。设计上更加高效，同时在性能上相较于早期模型（如 `text-embedding-ada-002`）有所提升。

- `text-embedding-3-large`：这个更强大的模型生成的嵌入最多可达 3,072 维。在多语言任务和大规模数据应用中表现出色。

性能：

- 基准测试性能：在多语言任务的 MIRACL 基准测试中，`text-embedding-3-large` 模型的平均得分为 54.9%，而 `small` 模型为 44.0%，相比之下，旧的 `ada` 模型得分为 31.4%。在英文任务的 MTEB 基准测试中，`large` 模型得分为 64.6%，而 `small` 模型为 62.3%。

- 灵活性：这两个模型都支持将嵌入缩短到更低维度，同时保留性能，使其适应不同的计算和存储需求。

支持减少嵌入的维度

基准测试	ada v2	text-embedding-3-small	text-embedding-3-large
平均 MIRACL 分数	31.4	44.0	54.9
平均 MTEB 分数	61.0	62.3	64.6

表 1.1: 模型性能比较

	ada v2		text-embedding-3-small		text-embedding-3-large	
嵌入维度	1536	512	1536	256	1024	3072
平均 MTEB 分数	61.0	61.6	62.3	62.0	64.1	64.6

表 1.2: 不同维度的模型性能比较

与使用较小的嵌入相比，使用较大的嵌入（例如将它们存储在向量存储中以便检索）通常成本更高，并且消耗更多的计算、内存和存储资源。

OpenAI 的两个新嵌入模型都使用一种技术进行训练，该技术允许开发人员权衡使用嵌入的性能和成本。具体来说，开发人员可以通过传递 `dimensions` API 参数来缩短嵌入（即从序列末尾删除一些数字），而不会失去嵌入的概念表示属性。例如，在 MTEB 基准测试中，一个 `textembedding-3-large` 嵌入可以缩短到 256 维，同时仍然优于一个未缩短的 1536 维 `text-embeddingada-002` 嵌入。

这样可以根据需求非常灵活地生成不同维度的嵌入。例如，当使用仅支持最多 1024 个维度的嵌入的向量数据存储时，开发人员现在仍然可以使用 OpenAI 的最佳嵌入模型 `text-embedding-3-large`，并为 `dimensions` API 参数指定值 1024，这将缩短从 3072 个维度向下的嵌入时间，从而牺牲一些准确性以换取较小的向量大小。

利用 OpenAI 获取嵌入

要获取嵌入，需要将文本字符串与嵌入模型名称（例如 `text-embedding-3-small`）一起发送到嵌入 API 端点。响应将包含一个嵌入向量（浮点数列表）以及一些额外的元数据。用户可以将其提取、保存在矢量数据库中，并用于许多不同的用例。

下面用一个例子具体展示如何获取嵌入。Amazon fine-food reviews 数据集截止 2012 年 10 月共包含亚马逊用户留下的 568,454 条食品评论，本文将使用 1,000 条最新评论的子集进行说明。评论是英文的，往往是正面或负面的。每条评论都有一个产品 Id、用户 Id、分数、评论标题（摘要）和评论正文（文本）。本文将摘要和文本合并为一个组合文本后，对这个组合文本进行编码，并输出单个向量嵌入，如下图所示。

```
from openai import OpenAI
client = OpenAI()

def get_embedding(text, model="text-embedding-3-small"):
    text = text.replace("\n", " ")
    return client.embeddings.create(input = [text], model=model).data[0].embedding

df['ada_embedding'] = df.combined.apply(lambda x: get_embedding(x, model='text-embedding-3-small'))

df.to_csv('output/embedded_1k_reviews.csv', index=False)
```

使用嵌入进行文本搜索

为了检索最相关的文档，我们使用查询的嵌入向量与每个文档之间的余弦相似度，并返回得分最高的文档。

```

from openai import OpenAI
client = OpenAI()

def get_embedding(text, model="text-embedding-3-small"):
    text = text.replace("\n", " ")
    return client.embeddings.create(input=[text], model=model).data[0].embedding

df['ada_embedding'] = df.combined.apply(lambda x: get_embedding(x, model='text-embedding-3-small'))
df.to_csv('output/embedded_1k_reviews.csv', index=False)

```

使用嵌入进行代码搜索

代码搜索的工作方式与基于嵌入的文本搜索类似。我们提供了一种从给定存储库中的所有 Python 文件中提取 Python 函数的方法。然后，text-embedding-3-small 模型对每个函数进行索引。为了执行代码搜索，我们使用相同的模型将查询嵌入到自然语言中。然后，我们计算生成的查询嵌入和每个函数嵌入之间的余弦相似度。最高的余弦相似度结果最相关。

```

from openai.embeddings_utils import get_embedding, cosine_similarity

df['code_embedding'] = df['code'].apply(lambda x: get_embedding(x, model='text-embedding-3-small'))

def search_functions(df, code_query, n=3, pprint=True, n_lines=7):
    embedding = get_embedding(code_query, model='text-embedding-3-small')
    df['similarities'] = df.code_embedding.apply(lambda x: cosine_similarity(x, embedding))
    res = df.sort_values('similarities', ascending=False).head(n)
    return res

res = search_functions(df, 'Completions API tests', n=3)

```

使用嵌入进行推荐

由于嵌入向量之间的距离越短，表示相似性越大，因此嵌入对于推荐类似的项目很有用。下面，我们举例说明一个基本的推荐器。它接受一个字符串列表和一个“源”字符串，计算它们的嵌入，然后返回字符串的排名，从最相似到最不相似的顺序排列。作为一个具体的例子，下面的链接笔记本将此函数的一个版本应用于 AG 新闻数据集（采样到 2,000 篇新闻文章描述），以将前 5 篇最相似的文章返回到任何给定的源文章。

```

def recommendations_from_strings(
    strings: List[str],
    index_of_source_string: int,
    model = "text-embedding-3-small",
) -> List[int]:
    """Return nearest neighbors of a given string."""

    # get embeddings for all strings
    embeddings = [embedding_from_string(string, model=model) for string in strings]

    # get the embedding of the source string

```



```

query_embedding = embeddings[index_of_source_string]

# get distances between the source embedding and other embeddings
distances = distances_from_embeddings(query_embedding, embeddings, distance_metric="cosine")

# get indices of nearest neighbors
indices_of_nearest_neighbors = indices_of_nearest_neighbors_from_distances(distances)
return indices_of_nearest_neighbors

```

2D 数据可视化

嵌入的大小随基础模型的复杂性而变化。为了可视化这些高维数据，我们使用 t-SNE 算法将嵌入的维数从 1536 降低到 2。一旦嵌入被简化为二维，我们就可以将它们绘制在二维散点图中。我们根据 Amazon fine-food reviews 数据集中评论者给出的星级评定对各个评论进行着色，从红色到绿色不等。即使在缩小的 2 维中，我们也可以观察到不错的数据分离。可视化似乎产生了大约 3 个集群，其中一个集群大多是负面评论。

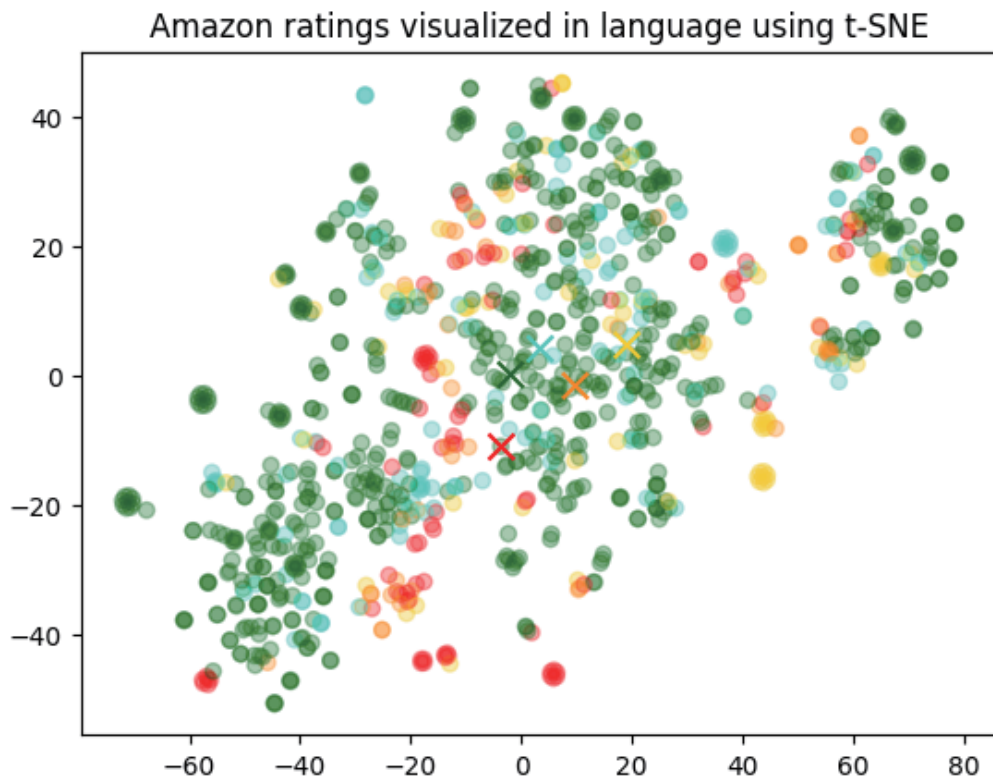


图 1.26: 可视化结果

嵌入为 ML 算法的文本特征编码器

嵌入可用作机器学习模型中的常规自由文本特征编码器。如果某些相关输入是自由文本，则合并嵌入将提高任何机器学习模型的性能。嵌入还可以用作 ML 模型中的分类特征编码器。

如果分类变量的名称有意义且数量众多，例如职位，则这会增加最大价值。对于此任务，相似性嵌入通常比搜索嵌入性能更好。通常嵌入表示非常丰富且信息密集。例如，使用 SVD 或 PCA 降低输入的维数，即使降低百分之十，通常也会导致特定任务的下游性能变差。

此代码将数据拆分为训练集和测试集，以下两个用例将使用它们，即回归和分类。

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(
    list(df.ada_embedding.values),
    df.Score,
    test_size = 0.2,
    random_state = 42
)
```

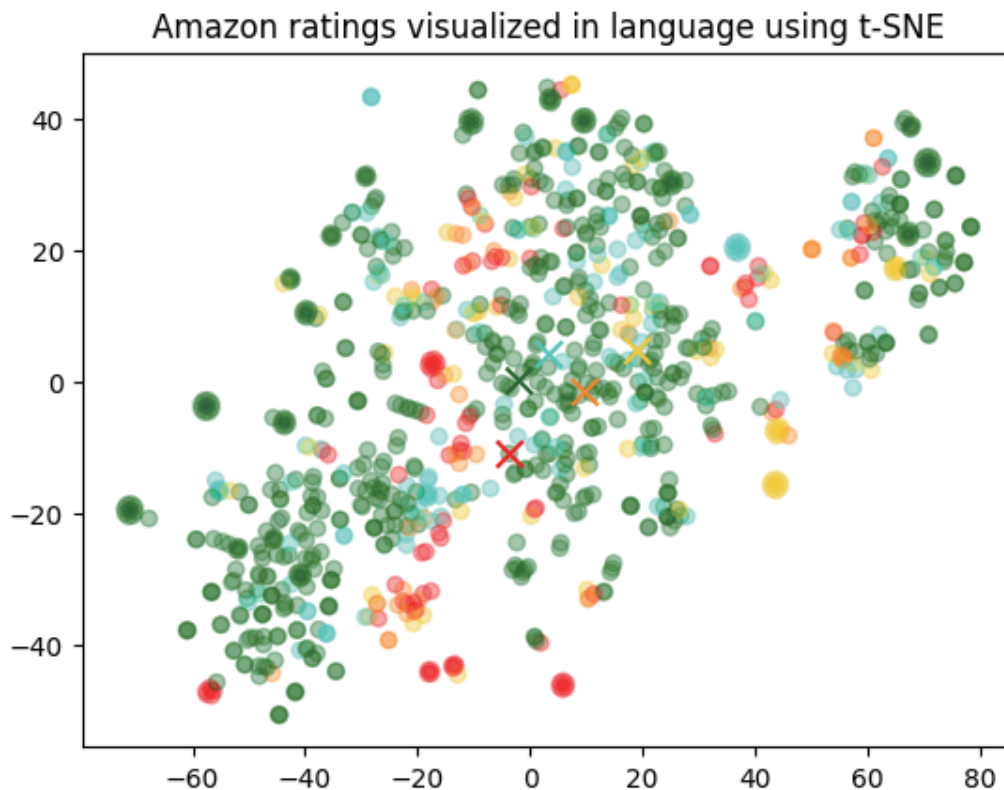


图 1.27: 分类结果

使用嵌入特征进行回归

回归意味着预测一个数字，而不是其中一个类别，嵌入提供了一种预测数值的友好方法。在此示例中，本文根据 Amazon fine-food reviews 数据集中评论的文本预测评论者的星级。由于嵌入中包含的语义信息很高，因此即使评论很少，预测也是不错的。假设分数是介于 1 和 5 之间的连续变量，并允许算法预测任何浮点值。ML 算法将预测值与真实分数的距离降至最低，并

实现 0.39 的平均绝对误差，这意味着平均预测偏差不到半颗星。

```
from sklearn.ensemble import RandomForestRegressor

rfr = RandomForestRegressor(n_estimators=100)
rfr.fit(X_train, y_train)
preds = rfr.predict(X_test)
```

使用嵌入特征进行分类

在文本分类任务中，本文根据 Amazon fine-food reviews 数据集中评论文本的嵌入来预测食品评论的分数（1 到 5），而不是让算法预测 1 到 5 之间的值。训练后，模型学会了预测 1 星和 5 星评论比更细微的评论（2-4 星）要好得多，这可能是由于更极端的情绪表达。

```
import numpy as np
from sklearn.cluster import KMeans

matrix = np.vstack(df.ada_embedding.values)
n_clusters = 4

kmeans = KMeans(n_clusters = n_clusters, init='k-means++', random_state=42)
kmeans.fit(matrix)
df['Cluster'] = kmeans.labels_
```

聚类

聚类是理解大量文本数据的一种方法。嵌入对于此任务很有用，因为它们为每个文本提供了语义上有意义的向量表示。因此，聚类将以一种无监督的方式发现数据集中的隐藏分组。

在这个例子中，在 Amazon fine-food reviews 数据集中发现了四个不同的集群：一个专注于狗粮，一个专注于负面评论，两个专注于正面评论。

```
import numpy as np
from sklearn.cluster import KMeans

matrix = np.vstack(df.ada_embedding.values)
n_clusters = 4

kmeans = KMeans(n_clusters = n_clusters, init='k-means++', random_state=42)
kmeans.fit(matrix)
df['Cluster'] = kmeans.labels_
```

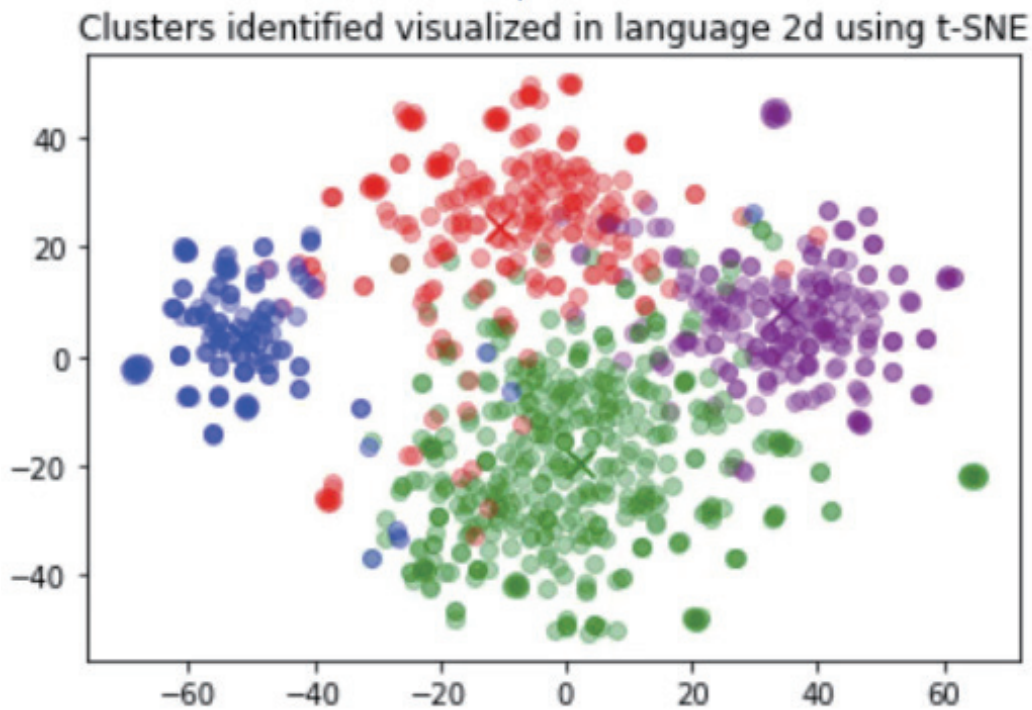


图 1.28: 聚类结果

Moderation

在现代互联网和社交媒体的迅速发展，信息的传播速度和广度前所未有。这种信息自由流动虽然带来了交流的便捷，但也带来了诸如谣言传播、不良信息泛滥等问题。因此，为了维护网络空间的健康和安全，Moderation（安全监测和敏感词检测）模型应运而生。这些模型可以有效地监测、识别并过滤不适当或有害的内容，从而确保信息环境的健康和秩序。

Moderation 是一种先进的大模型产品，专为保障在线平台和社交媒体的安全性而设计。其主要功能包括：

- 检测并过滤敏感词汇：通过识别特定的敏感词汇，避免这些词汇在公共平台上出现。
- 识别不良信息：包括暴力、色情、仇恨言论等内容的检测和屏蔽。
- 用户行为监控：分析用户的发言和行为，防范可能的违规行为。
- 内容审核：协助人工审核，提升审核效率和准确性。

Moderation 模型的技术原理

1. 自然语言处理（NLP）

自然语言处理是 Moderation 模型的核心技术之一。通过对文本数据的处理和理解，模型能够识别出不适当的内容。常用的 NLP 技术包括：

- 文本分类：将文本内容分为不同的类别，如正常、暴力、色情等。

- 情感分析：分析文本的情感倾向，识别仇恨言论等负面情绪。
- 关键词提取：提取文本中的关键敏感词汇，用于内容过滤。

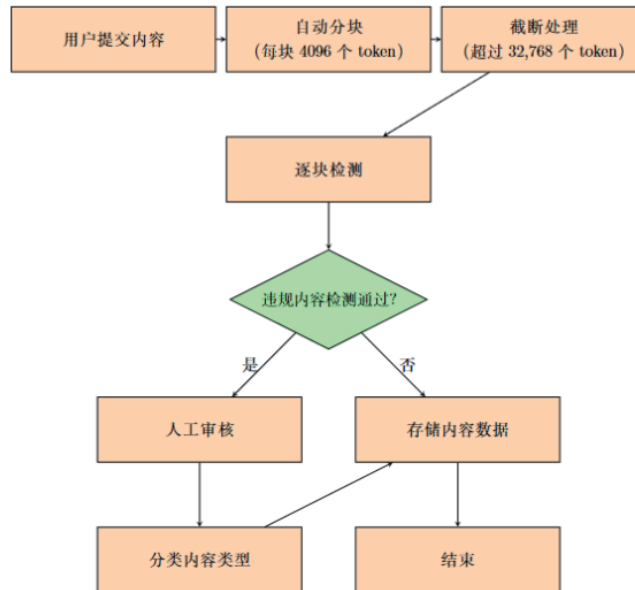


图 1.29: Moderation 模型流程图

2. 机器学习和深度学习

机器学习和深度学习为 Moderation 模型提供了强大的数据处理和分析能力。常用的技术包括：

- 监督学习：通过标注的训练数据，模型学习如何识别不良信息。
- 无监督学习：利用无标注数据进行聚类分析，发现潜在的违规内容。
- 神经网络：尤其是卷积神经网络（CNN）和循环神经网络（RNN），在处理文本分类和情感分析中表现出色。

3. 数据预处理

数据预处理是 Moderation 模型的重要环节，包括：

- 文本清洗：去除无关字符、停用词等。
- 文本分词：将文本分解为词语或短语。
- 特征提取：将文本转换为模型可处理的特征向量，如 TF-IDF、词向量等。

```

curl https://api.openai.com/v1/moderations \
  -X POST \
  -H "Content-Type: application/json" \
  -H "Authorization: Bearer $OPENAI_API_KEY" \
  -d '{"input": "Sample text goes here"}'

```

Moderation 模型的应用场景

如图所示，OpenAI 提供的 Moderation API 可以帮助开发者构建审核功能，用于检查文本内容是否具有潜在的危害性。除了 OpenAI 之外，许多其他数字平台和服务提供商都有自己的 moderation 系统。特别是在社交媒体、在线社区、游戏平台和电子商务等领域，moderation 被用于监控和管理用户生成的内容。

1. 社交媒体

社交媒体是 Moderation 模型的重要应用场景之一。通过实时监测用户发布的内容，可以有效防止不良信息的传播。例如，Facebook 应用了先进的 Moderation 技术来维护社区规范，Facebook 的 Moderation 技术主要体现在以下几个方面：

内容审核系统

- 机器学习算法：Facebook 利用机器学习算法，特别是深度学习模型，对用户生成的文本、图像和视频进行分析和审核。这些算法能够识别各种类型的不良内容，如暴力内容、色情、仇恨言论等。

- 图像和视频识别：Facebook 使用计算机视觉技术，识别和过滤图像和视频中的不适宜内容，包括敏感或暴力内容，以及未经授权的内容。

个性化设置和用户控制

用户可以根据自己的偏好和需求，调整 Facebook 的内容过滤和审核设置。这种个性化设置可以提升用户体验，同时保护用户免受不良内容的影响。

2. 在线论坛和社区

在论坛和社区中，Moderation 模型帮助管理员识别并处理违规内容，保证讨论环境的健康和积极。例如，Reddit 利用机器学习模型来自动标记和删除违规帖子。Reddit 中的 Moderation 技术：

机器学习模型的应用

- 自然语言处理（NLP）Reddit 利用自然语言处理技术来分析帖子内容。NLP 模型能够理解和处理大量的文本数据，从中识别出潜在的违规内容，如仇恨言论、骚扰、虚假信息等。通过对帖子标题、正文和评论的分析，NLP 模型能够准确判断帖子是否违反社区准则。

- 分类器模型 Reddit 使用各种分类器模型，将帖子分为不同的类别，包括正常、潜在违规和明显违规。常用的分类器包括逻辑回归、支持向量机（SVM）和深度神经网络（DNN）。这些模型通过训练数据集学习不同类型帖子的特征，从而能够在新帖子发布时，快速进行分类和处理。

自动化标记和删除

- 自动标记当用户发布新帖子时，机器学习模型会立即对其内容进行分析。如果检测到潜在的违规行为，系统会自动标记该帖子，提醒社区管理员进行进一步审查。自动标记功能不仅提高了审核效率，还能帮助管理员快速识别和处理问题帖子，减少违规内容对社区的影响。

- 自动删除对于明显违规的帖子，机器学习模型会直接进行删除处理。这类帖子通常包含严重的仇恨言论、暴力威胁或其他违反社区准则的内容。自动删除机制确保了严重违规内容能够被及时移除，保护了社区成员的安全和体验。

3. 视频和直播平台

对于 YouTube、Twitch 等视频和直播平台，Moderation 模型可以对视频内容进行分析，识别出可能的违规片段，并提示审核人员进一步处理。YouTube 的 Moderation 技术：

视频内容分析

- 机器学习和人工智能 YouTube 使用机器学习和人工智能技术，对上传的视频进行自动化内容分析。通过训练深度学习模型，YouTube 能够识别出视频中的不适宜内容，如暴力、仇恨言论、色情、滥用药物等。这些模型不仅仅依赖于文本描述，还能够分析视频帧、音频内容以及视频元数据，以全面评估视频内容的合规性。

- 计算机视觉计算机视觉技术用于分析视频帧中的图像内容。通过检测和识别图像中的元素，YouTube 可以识别出暴力行为、武器、成人内容等不适宜的图像片段。计算机视觉技术还能够检测视频中的字幕和文字内容，结合自然语言处理技术进行全面的内容审核。

- 音频分析 YouTube 使用语音识别技术将视频中的语音转换为文本，并结合自然语言处理技术进行分析。这使得平台可以检测出视频中的仇恨言论、暴力威胁和其他不当语言。此外，音频分析还能够识别出背景音乐中的版权侵权问题，帮助平台自动化地进行版权管理。

自动化过滤和实时响应

- 自动化过滤系统上传的视频会首先通过自动化过滤系统进行初步筛查。这个系统能够快速识别出明显违规的内容，并采取相应措施，如自动删除、限制观看或标记为审核待定。自动化过滤系统还会根据内容的严重性，调整审核优先级，确保严重违规内容能够得到及时处理。

- 实时响应和警报当检测到潜在的违规内容时，系统会立即向审核团队发出警报。实时响应机制确保了内容审核员可以尽快处理这些高风险内容。这不仅有助于保护用户免受不良内容的影响，还能及时遏制不适宜内容的传播。

4. 游戏社区

在在线游戏社区中，Moderation 模型可以监控玩家的聊天内容，防止辱骂、欺凌等行为。例如，

暴雪娱乐公司在《守望先锋》中应用了自动检测系统来维护游戏环境的和谐。《守望先锋》中的 Moderation 技术：

自动检测系统

- 自然语言处理（NLP）暴雪娱乐公司利用自然语言处理技术分析游戏内的聊天内容。通过训练 NLP 模型，可以识别出各种形式的不良言论，包括辱骂、欺凌、种族歧视和其他有害言论。NLP 模型会扫描玩家的文本聊天记录，识别关键词和短语，并根据上下文判断其是否属于不良行为。

- 机器学习机器学习算法通过大量的聊天数据进行训练，不断提高对不良行为的检测精度。算法不仅能够识别常见的侮辱性语言，还能够检测出变种词汇和隐晦表达。这些算法可以学习玩家的语言模式，检测出新的不良行为趋势，并动态更新识别规则。

实时监控和响应

- 实时监控游戏内的聊天系统会进行实时监控，自动检测和标记潜在的违规内容。实时监控确保了违规行为能够在发生的第一时间被识别和记录。系统会为每个检测到的不良言论生成报告，记录详细的上下文和相关信息，供后续处理。

- 自动化响应当检测到明显的不良行为时，系统可以自动采取措施，如警告玩家、禁言或临时禁止其游戏权限。这些自动化响应措施有助于快速遏制不良行为的蔓延。对于严重或重复违规的玩家，系统会自动上报给人工审核团队进行进一步处理。

Moderation 模型的挑战

- 数据偏见 Moderation 模型依赖大量的训练数据，而这些数据往往带有偏见。如何保证模型的公平性和公正性，是一个重要的挑战。

- 语义理解敏感词汇和不良信息的判断往往需要深入的语义理解。例如，“炸弹”在不同上下文中可能具有完全不同的含义，这对模型的准确性提出了很高的要求。

- 实时处理互联网内容的产生和传播速度非常快，Moderation 模型需要具备强大的实时处理能力，以便及时发现并处理违规内容。

- 用户隐私在进行内容监测时，如何平衡安全监测与用户隐私之间的关系，是一个复杂的问题。模型需要在不侵犯用户隐私的前提下，完成对内容的有效监测。

未来发展方向

Moderation 模型将随着技术的发展不断进步。未来的发展方向包括：

- 多模态内容分析：不仅限于文本，还包括图片、视频等多种媒体形式的内容分析。

- 更强的上下文理解：通过更先进的 NLP 技术，提升模型的上下文理解能力，提高检测的准确性。

- 自动化与人工审核结合：在自动化检测的基础上，进一步提升人工审核的效率和精准度。
- 用户参与：通过用户的反馈和参与，持续优化 Moderation 模型的表现。

1.2.3 大模型前沿问题

大模型 +RAG

RAG 框架

(1) naive RAG

naive RAG 方案如图 1.2.1 是最简单、最原始的 RAG 方案，也被称为“Retrieve-Read”框架，其包含几个关键阶段：

• Indexing（离线处理）：

从各种格式（PAF、HTML、Markdown、Word 等）的额外语料库中提取纯文本内容。由于 LLM 上下文窗口的限制，比如常见的 2K、4K，需要将提取的文本内容切分为不同的 chunks。

使用文本 embedding 模型，针对每个 chunk 提取相应的文本 embedding。

将文本 embedding 和对应的 chunk 存储为索引，能一一对应。

• Retrieval（在线处理）：

使用文本 embedding 模型，针对用户 query 提取 query embedding。

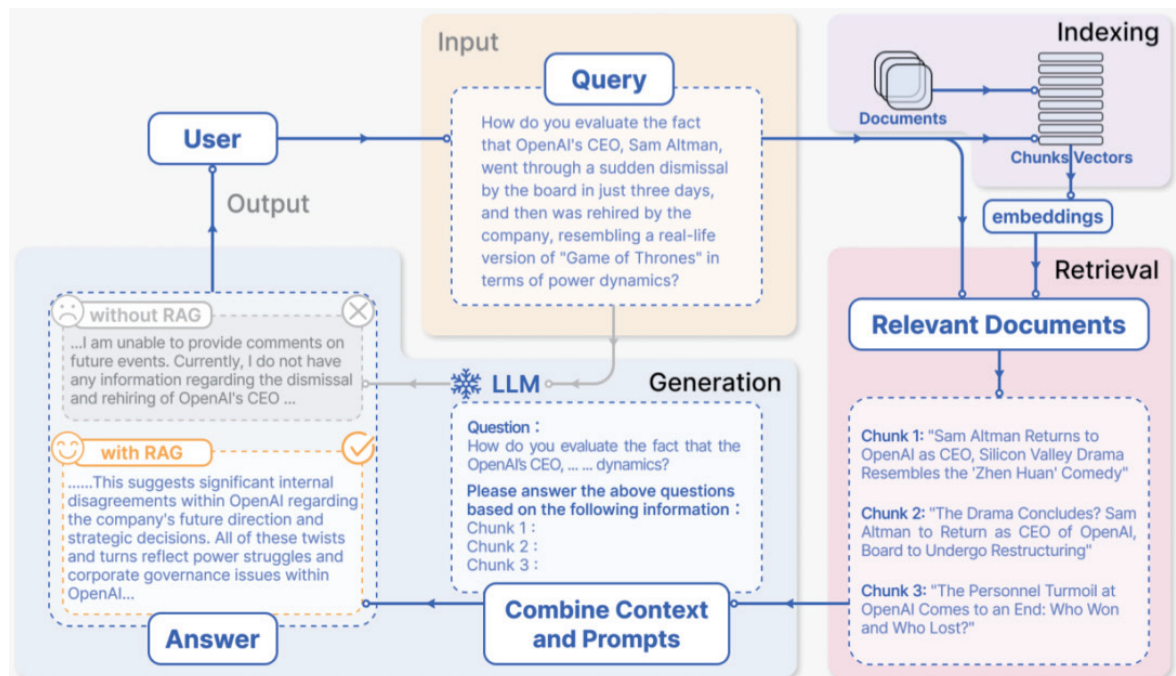


图 1.30: naive RAG

使用 query embedding 与索引中的 chunk embedding 进行比对，找到最相似的 k 个 embedding。

提取 k 个最相似 embedding 对应的 chunk。

• Generation（在线处理）：

将 query 与检索到的 chunks 进行合并。

将合并后的 query 输入 LLM 用于生成，并返回生成结果。

naive RAG 有以下的不足之处：

• 检索质量：低精度的检索系统可能引导 LLM 产生幻觉，生成错误的输出；低召回的检索系统可能导致遗漏部分信息，导致 LLM 生成不准确或不全面的结果。

• 指令遵循：LLM 可能不基于 prompt 中检索到的上下文来生成答案，或者使用了不相关的上下文，从而导致生成结果出现偏差。

• 增强过程：可能无法将检索到的内容很好的与 query 相结合，或者检索到的内容中包含冗余和重复，导致生成结果中出现重复内容。

• 内容相关性：需要平衡检索到的多个内容的价值，辨别其对生成任务的重要性，此外也需尽可能保持风格一致。

• 过度依赖：LLM 可能会过度依赖检索到的内容，而忽略本身已经学习到的知识。

(2) Advanced RAG

如下图所示，Advanced RAG 相比 Naive RAG 进一步增加了 Pre-Retrieval 和 Post-Retrieval 模块：

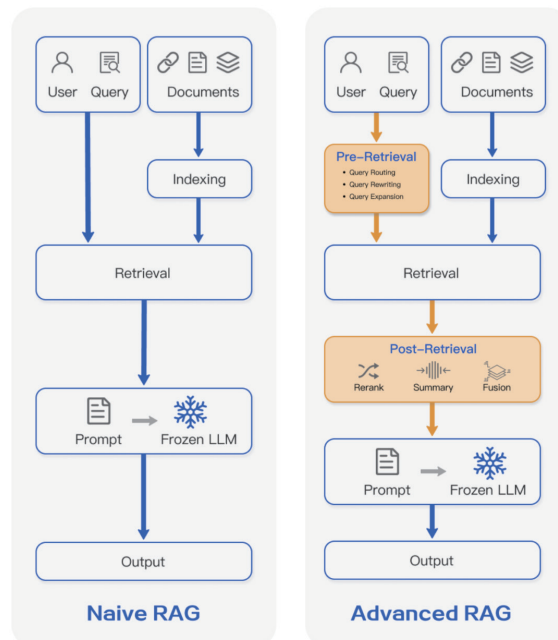


图 1.31: Advanced RAG

Pre-Retrieval Process

检索前的阶段可以分为离线优化、在线优化和混合检索：

- 增强数据粒度（离线优化）：旨在提升文本标准化、一致性、事实准确性和丰富的上下文，以提高 RAG 系统的性能。比如删除不相关的信息、消除实体和数据中的歧义、更新过时文档等。
- 优化索引结构（离线优化）：调整块的大小以捕获相关上下文、跨多个索引路径进行查询。
- 添加元数据（离线优化）：将引用的元数据（如日期和用途）集成到块中以进行过滤，并合并元数据以提高检索效率。
- 对齐优化（离线优化）：通过在文档中引入“假设问题”来解决文档之间的对齐问题和差异，以纠正对齐问题和差异。
- 在线优化：主要是指在检索阶段对用户 query 进行改写、扩展、纠错等，来提升检索质量。
- 混合检索：主要是指充分利用关键词检索、语义检索、向量检索等其他检索技术来提升检索丰富度，同时也可以保证一致性。

Post-Retrieval Process

检索到的内容比较多，重要性也各不相同，如果一味地和 query 合并可能会超过 LLM 上下文限制，同时增加计算开销，也可能引入噪声，导致生成质量不佳。此时，通常需要对检索到的内容进一步处理：

- 重排序（Re-Ranking）：这是搜索领域非常常见的手段，不过在传统搜索中通常是按相关性、质量等进行排序输出；而在 LLM 生成领域要考虑检索到文档的多样性，以及 LLM 对 Prompt 中内容所在位置的敏感度等，比如 `LostInTheMiddleRanker` 将最佳文档交替地放在上下文窗口的开头和结尾。
- 提示压缩（Prompt Compression）：有研究表明，检索到的文档中的噪声会对 RAG 性能产生不利影响。在后处理中，可以重点压缩不相干的上下文，突出关键段落，减少整体上下文长度。也可以用专门的模型对文档进行压缩、总结、过滤等。

(3) Modular RAG

Modular RAG 方案

模块化 RAG 源于传统的 RAG 框架，其集成不同功能的子模块，提供了很大的灵活性。其主要是增加了一些新的模块，并构建了一些新的范式。

新的模块

- Search Module：这里的 Search 模块专指针对特定应用场景的模块。这种能力可以通过 LLM 的代码能力来获得，比如直接让 LLM 生成对应的 SQL 代码。

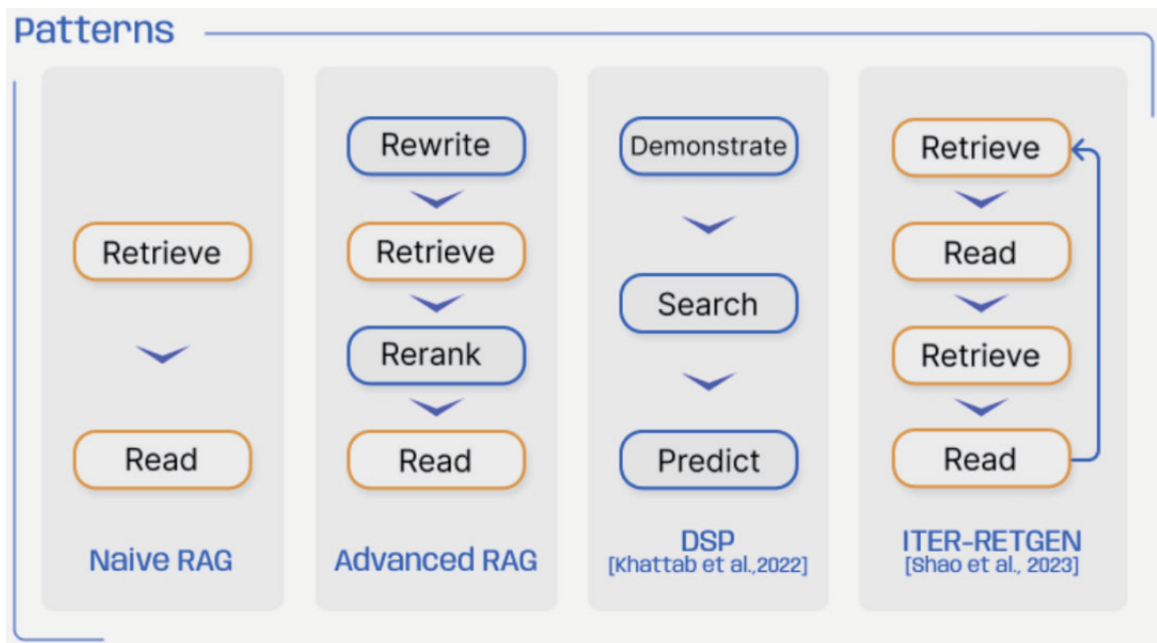


图 1.33: 新的范式

RAG 优化

(1) 索引优化

Chunk 优化

在构建离线索引时通常会将文档分为不同的 Chunk，然后提取 embedding。而 Chunk 的大小有多方面影响，比如，如果 Chunk 过大，则可能超过后续 LLM 的上下文窗口大小；如果 Chunk 过小，则可能导致相关信息被分割开来，影响检索质量。常见的 Chunk 优化手段有：

- **Chunk 大小：**不是按句子划分 Chunk，而是结合 LLM 窗口大小，此时需要考虑 prompt 占据的长度，为 prompt 留下足够空间。
- **滑动窗口：**直接按照某个固定大小切割文档可能导致关键内容被分到两个不同 Chunk 中，从而影响检索质量，此时可以采用有重叠的切割方案。
- **总结：**有些时候文档内容会比较冗余，此时可以考虑对文档进行总结，减少冗余信息。

Metadata

为索引添加时间、类型、文档标题以及页面等信息，可以用于后续的过滤，增强检索效率，也可以帮助增强生成质量。

结构化语料

多级检索也是检索系统中非常常用的手段，比如百亿级的图像检索，如果全量比对其代价非常高，此时可以考虑对索引进行聚类，比如分为 1K 个聚类中心，检索时先与聚类中心比对，

取 Top5 后在每个聚类中比对，计算量可以降低为 1/200。同时也可以按照垂类进行划分，比如划分为服饰、箱包、植物、动物等，然后检索时先识别意图，然后在对应的垂类检索。

类似地，可以先对整个文档进行总结归纳，基于总结构建一级索引，文档的 Chunk 构建二级索引，检索时先从总结索引中检索；此外也可以将文档中的不同元素分别建立对应类别的索引，比如对图表索引。

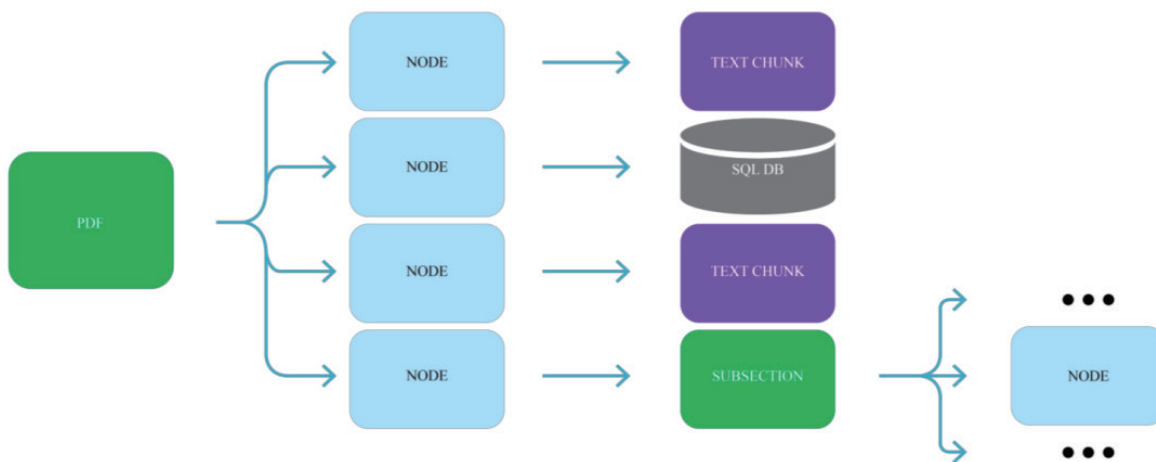


图 1.34: 结构化语料

(2) 检索源优化

RAG 的关键是通过额外的语料库来增强 LLM 的生成质量，数据源可以分为非结构化数据、结构化数据和 LLM 生成的内容，它们也往往需要不同的处理手段。如下图所示为不同阶段采用的不同数据来源，其中叶子的颜色表示数据类型，可以看出，大部分为非结构化数据，但使用结构化数据和 LLM 生成内容也越来越多：

- 结构化数据：越来越多的工作开始把知识图谱（Knowledge Graph, KG）引入到 RAG 中，比如 KnowledGPT 用于推理阶段，SUGRE 用于微调阶段。
- LLM 生成的内容：除了外部知识外，LLM 本身的知识也至关重要，有些工作尝试对用户 query 进行分类，然后有选择的使用外部知识或内部知识。此外也可以用 LLM 直接生成代码等。

(3) Query 优化

Query 优化是检索系统中提升检索结果准确性非常有必要的阶段，通过这些手段，系统可以更好的理解用户意图并提供更准确的结果。通常包含如下手段：

- Query 改写：对用户输入 query 进行修改或重写，通常只调整 query 结构、添加或删除关键词，或者通过近义词替换来扩展检索范围。
- Query 纠错：通常是指尝试修正用户 query 中可能的拼写或语法错误。
- Query Clarification：通常是指修正或分解用户 query 中不明确或模糊的内容，使系统更好

地理解并执行相应的检索，有助于避免歧义。

(4) Embedding 优化

- 挑选合适的 embedding 模型：检索任务通常是判断两段内容是否相关，相似任务是判断两段内容是否相似，相关的内容不一定相似。因此随意挑选一个文本模型用于提取 embedding 不一定合适，学习的目标也不一定相同。

- 针对任务进行微调：embedding 模型的训练预料和实际检索的语料分布可能并不相同，此时在对应语料上进行 finetuning 可以显著提升检索到的内容的相关性。比如 OpenAI 也提供了代码相关的 code embedding 模型。

- Adapter：如下图所示，也有研究者引入 Adapter，在提取 query embedding 后进一步让 embedding 经过 Adapter，以便实现与索引更好的对齐。

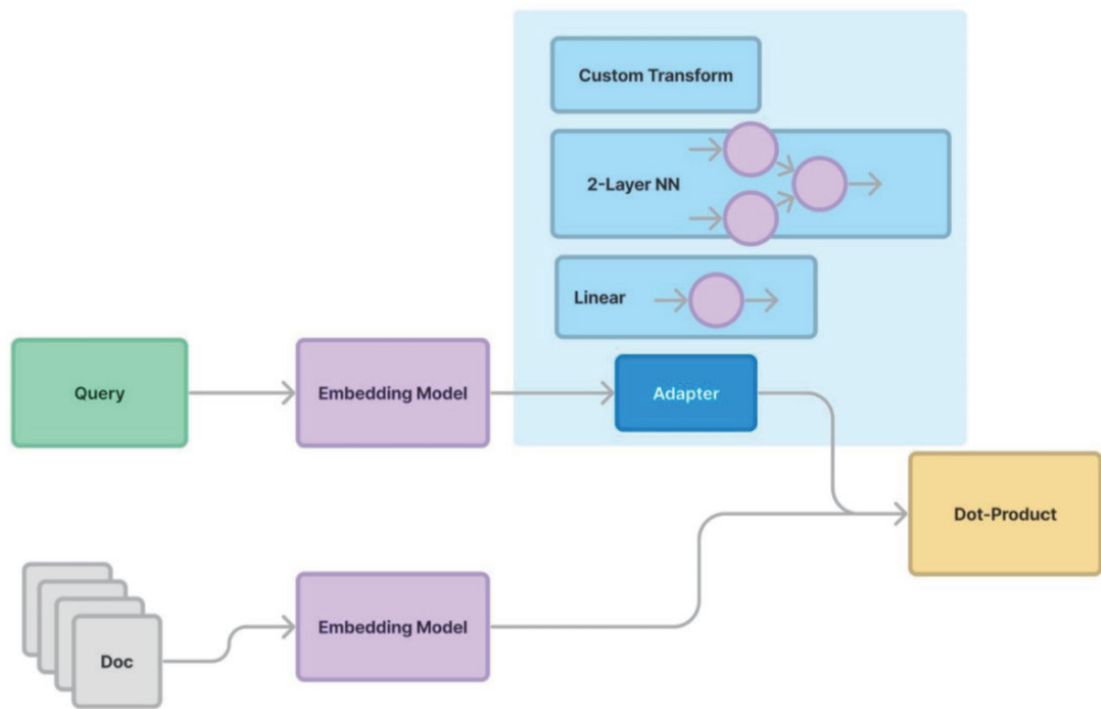


图 1.35: Adapter

(5) 检索过程优化

RAG 中的标准做法是通过一次检索、生成获得最终结果，然而这种方式在复杂任务上可能表现不佳，比如需要多步推理的场景。研究者也提出了相应的改进方案：迭代检索、递归检索和自适应检索。

迭代检索

如下图所示，[2305.15294] Enhancing Retrieval-Augmented Large Language Models with Itera-

tive Retrieval-Generation Synergy 中迭代检索的示例，其结合了“检索增强生成”和“生成增强检索”：

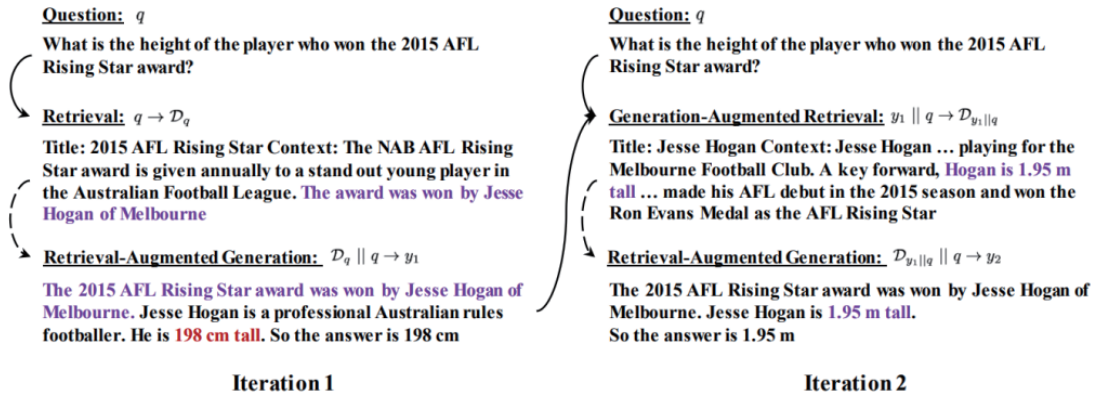


Figure 1: ITER-RETGEN iterates retrieval and generation. In each iteration, ITER-RETGEN leverages the model output from the previous iteration as a specific context to help retrieve more relevant knowledge, which may help improve model generation (e.g., correcting the height of Hesse Hogan in this figure). We only show two iterations in this figure for brevity. Solid arrows connect queries to the retrieved knowledge, and dashed arrows denote retrieval-augmented generation.

图 1.36: 迭代检索

递归检索

递归检索是指可以将之前的检索输出作为后续的检索输入。递归检索通常用于需要渐进式收敛到最终答案的场景，比如学术研究、法律案例分析等。针对多级索引（比如摘要索引，Chunk 索引），可以使用递归检索；此外，针对具有图结构的数据源，使用递归检索也可以检索更全面的信息。

如下图所示为 [2212.10509] Interleaving Retrieval with Chain-of-Thought Reasoning for Knowledge-Intensive Multi-Step Questions 中递归检索的示例，其结合思维链（COT）来指导检索过程，并根据获得的检索结果对 CoT 进行细化：

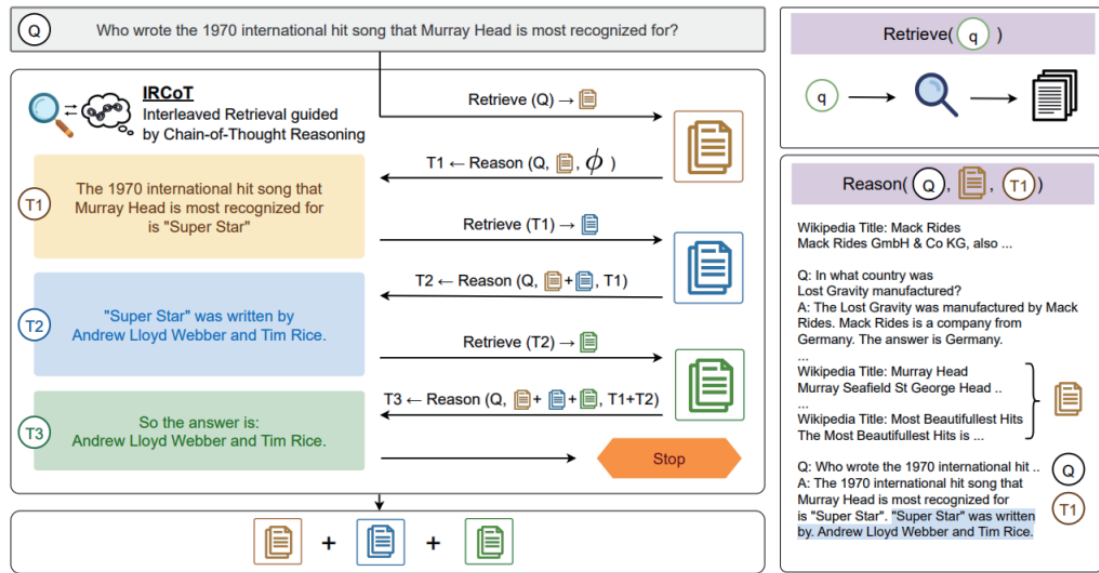


Figure 2: IRCot interleaves chain-of-thought (CoT) generation and retrieval steps to guide the retrieval by CoT and vice-versa. We start by retrieving K documents using the question as they query and repeat two steps alternately until termination. (i) reason-step generates next CoT sentence based on the question, so far retrieved paragraphs, and CoT sentences. (ii) retrieve-step retrieves K more paragraphs based on the last CoT sentence. The process terminates when the generated CoT has “answer is” or the number of steps exceeds a threshold. The collection of all paragraphs is returned as the retrieval result on the termination.

图 1.37: 递归检索

自适应检索

自适应检索提供了一种动态调整机制，根据不同任务和上下文的特定需求定制检索过程。

如下图所示为 [2310.11511] Self-RAG : Learning to Retrieve, Generate, and Critique through Self-Reflection 中自适应检索的过程，其引入了“Reflection” Token，允许模型自省其输出，该模型可自主决定何时激活检索，提高了模型在生成准确结果方面的自主判断能力：

RAG 评估

(1) 评估目标

不管 RAG 系统怎么变，其核心离不开两个关键部分：检索和生成。因此也就可以针对这两个部分进行评估。

- 检索质量：检索系统已经发展多年，在搜索引擎、推荐系统等领域都广泛使用，也相应有很多衡量检索质量的指标，比如命中率、MRR 和 NDCG。

- 生成质量：生成质量主要用于评估模型基于检索到的内容生成连贯、准确答案的能力。其评估通常也包含有标注评估和五标注评估（也都可以采用人工评估和自动评估的方式）

(2) 评估维度

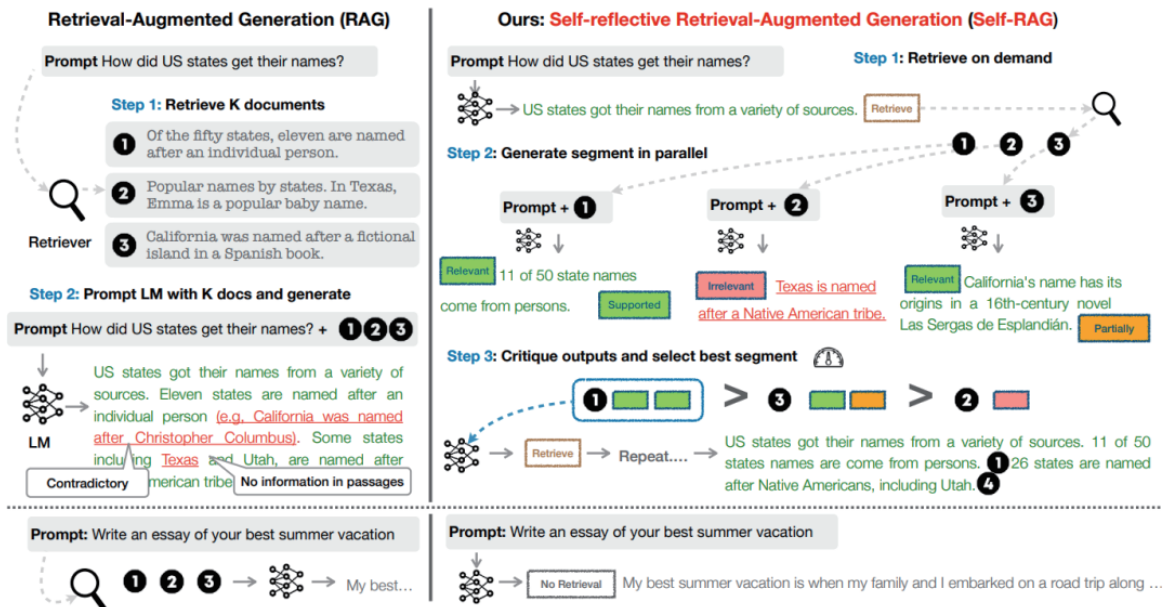


Figure 1: Overview of SELF-RAG. SELF-RAG learns to retrieve, critique, and generate text passages to enhance overall generation quality, factuality, and verifiability.

图 1.38: 自适应检索

- Context Relevance (上下文相关性): 评估检索到的上下文的精确性和特异性, 确保相关性并最大限度降低与无关内容相关的处理成本。

- Answer Faithfulness (答案真实性): 确保生成的答案与检索到的上下文保持一致并避免矛盾。

- Noise Robustness (噪声鲁棒性): 评估模型处理与问题相关但缺乏实质性信息的噪声文档的能力。

- Negative Rejection (否定拒绝): 评估模型对于检索到的文档不包含回答问题的必要知识时不做出响应的辨别力。

- Information Integration (信息融合): 评估模型综合来自多个文档的信息以解决复杂问题的能力。

- Counterfactual Robustness (反事实鲁棒性): 评估模型识别和忽略文档中已知不准确信息的能力。

(3) 评估基准和评估工具

Table 2: Summary of metrics applicable for evaluation aspects of RAG

	Context Relevance	Faithfulness	Answer Relevance	Noise Robustness	Negative Rejection	Information Integration	Counterfactual Robustness
Accuracy	✓	✓	✓	✓	✓	✓	✓
EM					✓		
Recall	✓						
Precision	✓			✓			
R-Rate							✓
Cosine Similarity			✓				
Hit Rate	✓						
MRR	✓						
NDCG	✓						

图 1.39: 评估基准

Table 3: Summary of evaluation frameworks

Evaluation Framework	Evaluation Targets	Evaluation Aspects	Quantitative Metrics
RGB [†]	Retrieval Quality	Noise Robustness	Accuracy
	Generation Quality	Negative Rejection	EM
RECALL [†]	Generation Quality	Information Integration	Accuracy
		Counterfactual Robustness	Accuracy
RAGAS [‡]	Retrieval Quality	Context Relevance	*
	Generation Quality	Faithfulness	*
ARES [‡]	Retrieval Quality	Answer Relevance	Cosine Similarity
	Generation Quality	Context Relevance	Accuracy
TruLens [‡]	Retrieval Quality	Faithfulness	Accuracy
	Generation Quality	Answer Relevance	Accuracy
	Retrieval Quality	Context Relevance	*
	Generation Quality	Faithfulness	*
		Answer Relevance	*

图 1.40: 评估工具

大模型 + 智能体

引言

大模型（Large Language Models, LLMs）在自然语言处理（NLP）领域取得了显著的进展，能够生成高度流畅且上下文相关的文本。智能体（Agents）是能够在动态环境中自主执行任务和决策的系统。将大模型与智能体结合，可以显著提升智能系统的能力，使其在更广泛的应用场景中发挥作用。本文将从多个角度探讨大模型与智能体结合的优势、应用场景、技术挑战和未来展望，并介绍 Formal-LLM 框架在这一领域的应用。

大模型与智能体结合的优势

(1) 自然语言处理能力

大模型在自然语言处理方面表现优异，其强大的语言理解和生成能力可以为智能体提供显著的优势。

(2) 对话系统

大模型可以增强智能体的对话能力，使其能够理解和生成自然语言对话，提供更自然、更流畅的交互体验。通过引入大模型，智能体可以处理更复杂的用户请求，提供详细和精确的回答。

例如，OpenAI 的 GPT-4 模型被用于构建聊天机器人，能够进行深度对话、回答问题、甚至撰写文章。

(3) 多语言支持

大模型可以支持多语言处理，使智能体能够在全球范围内应用。智能体可以根据用户的语言自动切换，提供多语言支持，满足不同语言背景用户的需求。例如，Google 的翻译服务利用大模型提供高质量的多语言翻译，使得跨语言交流更加便捷。

(4) 信息检索

大模型可以帮助智能体从大量的文本数据中快速提取信息，回答用户的问题。智能体可以利用大模型在海量数据中进行搜索和分析，找到最相关的信息。例如，IBM 的 Watson 系统使用大模型从医疗文献中提取信息，帮助医生做出诊断和治疗决策。

(5) 推理与决策

通过大模型的推理能力，智能体可以在复杂的情境中做出更准确的决策。大模型可以模拟人类的思维过程，帮助智能体在不确定环境中进行推理和决策。例如，自动驾驶汽车可以利用大模型分析道路状况、交通信号和其他车辆的行为，做出安全的驾驶决策。

(6) 个性化定制

大模型可以帮助智能体根据用户的偏好和行为进行个性化调整，提高用户体验。智能体可以分析用户的历史行为和偏好，提供定制化的服务。例如，Netflix 利用大模型推荐系统，根据用户的观影历史推荐个性化的电影和电视剧。

(7) 持续学习

智能体可以利用大模型不断学习和更新知识，保持最新的信息和技能。通过持续学习，智能体可以适应环境的变化，不断提高其性能。例如，金融交易系统利用大模型分析市场趋势和历史数据，不断调整交易策略，提升投资回报率。

应用场景

(1) 智能客服

结合大模型和智能体，可以构建更智能、更高效的客服系统，处理用户查询、提供技术支持等。

例如，智能客服可以理解用户的自然语言问题，并利用大模型提供准确且详细的回答，减少客户等待时间，提高客户满意度。例如，阿里巴巴的智能客服系统可以处理海量客户咨询，自动回复常见问题，并在复杂问题上转接人工客服。

(2) 智能助手

在智能家居、办公自动化等领域，结合大模型的智能体可以提供更智能的助手服务，如日程安排、信息查询等。智能助手可以理解用户的复杂指令，提供个性化建议，提升用户的生活和工作效率。例如，亚马逊的 Alexa 智能助手可以控制家电、播放音乐、设置提醒，并通过与大模型集成提供详细的问答服务。

(3) 自动驾驶

在自动驾驶系统中，结合大模型的智能体可以更好地理解交通状况、识别路标和决策路径。大模型可以处理和分析大量的实时数据，帮助自动驾驶车辆在复杂的交通环境中做出安全决策。例如，特斯拉的自动驾驶系统利用大模型分析传感器数据、预测行人和车辆行为，从而实现自动驾驶。

(4) 医疗健康

智能体可以利用大模型提供医疗咨询、健康管理等服务，提升医疗服务质量。例如，智能体可以根据患者的病历和症状，利用大模型提供初步诊断和治疗建议，辅助医生决策，提升医疗效率和准确性。例如，Babylon Health 的 AI 医生利用大模型进行症状分析，提供个性化的健康建议和诊疗方案。

技术挑战

(1) 计算资源

大模型需要大量的计算资源和存储空间，如何在智能体中高效部署和运行大模型是一个挑战。当前，训练和运行大模型通常需要高性能计算集群，这对于资源有限的组织来说是一个显著障碍。例如，GPT-3 模型的训练需要数千个 GPU 的计算能力，这使得其应用成本高昂。

(2) 实时性

智能体通常需要实时响应，而大模型的计算复杂度较高，如何保证响应速度是一个关键问题。在实时性要求高的应用场景中，如自动驾驶和智能助手，延迟可能导致严重后果，因此需要优化模型的运行效率。例如，自动驾驶系统必须在毫秒级别做出反应，任何延迟都可能导致危险情况。

(3) 安全与隐私

在使用大模型的过程中，如何保护用户的隐私数据并防止安全漏洞是必须解决的问题。智

能体需要处理大量的用户数据，这些数据可能包含敏感信息，因此需要采取有效的措施来防止数据泄露和滥用。例如，医疗应用中的智能体必须严格遵守数据隐私法规，确保患者信息的安全。

(4) 模型解释性

大模型往往是黑箱模型，如何提高其解释性以便智能体能够在关键决策过程中提供透明和可信的解释是一个重要的研究方向。提升模型的解释性，有助于用户理解智能体的决策过程，增加信任度。例如，在金融领域，投资决策系统需要提供明确的理由和依据，以便投资者理解和信任其建议。

(5) Formal-LLM 框架

Formal-LLM 框架是一个专门设计用于将大模型与智能体结合的框架，旨在解决上述技术挑战，并提供一个高效、灵活和安全的集成解决方案。以下是 Formal-LLM 框架的关键特性和优势：

- **高效的模型架构**：Formal-LLM 框架采用了模型压缩、量化和蒸馏等技术，显著降低了大模型的计算资源需求。通过这些技术，Formal-LLM 能够在计算资源有限的设备上高效运行，为智能体提供强大的语言理解和生成能力。

- **实时性优化**：为了满足智能体的实时性需求，Formal-LLM 框架对模型的运行效率进行了深入优化。通过引入异步计算和分布式处理技术，Formal-LLM 能够显著降低响应延迟，确保智能体在关键任务中能够实时响应。

- **安全与隐私保护**：Formal-LLM 框架在设计时高度重视安全与隐私保护。通过引入联邦学习和差分隐私技术，Formal-LLM 确保用户数据在处理过程中的安全性和隐私性。此外，框架还提供了完善的访问控制和数据加密机制，防止数据泄露和滥用。

- **模型解释性增强**：为了提高模型的解释性，Formal-LLM 框架引入了可解释 AI (XAI) 技术。通过生成可解释的模型输出，智能体可以向用户提供透明的决策过程和依据。这种解释性增强有助于提升用户对智能体的信任和接受度。

- **多模态支持**：Formal-LLM 框架支持多模态数据的集成处理，包括文本、图像、音频和视频等。通过多模态学习，智能体可以更全面地理解和响应用户需求，提供更加丰富和多样的交互体验。例如，智能家居系统可以结合语音指令和图像识别，提供更智能的家电控制和安防服务。

应用实例

(1) 智能客服结合大模型和智能体，可以构建更智能、更高效的客服系统，处理用户查询、提供技术支持等。例如，智能客服可以理解用户的自然语言问题，并利用大模型提供准确且详细的回答，减少客户等待时间，提高客户满意度。例如，阿里巴巴的智能客服系统可以处理海量客户咨询，自动回复常见问题，并在复杂问题上转接人工客服。

(2) 智能助手在智能家居、办公自动化等领域，结合大模型的智能体可以提供更智能的助手服务，如日程安排、信息查询等。智能助手可以理解用户的复杂指令，提供个性化建议，提升用户的生活和工作效率。例如，亚马逊的 Alexa 智能助手可以控制家电、播放音乐、设置提醒，并通过与大模型集成提供详细的问答服务。

(3) 自动驾驶在自动驾驶系统中，结合大模型的智能体可以更好地理解交通状况、识别路标和决策路径。大模型可以处理和分析大量的实时数据，帮助自动驾驶车辆在复杂的交通环境中做出安全决策。例如，特斯拉的自动驾驶系统利用大模型分析传感器数据、预测行人和车辆行为，从而实现自动驾驶。

(4) 医疗健康智能体可以利用大模型提供医疗咨询、健康管理等服务，提升医疗服务质量。例如，智能体可以根据患者的病历和症状，利用大模型提供初步诊断和治疗建议，辅助医生决策，提升医疗效率和准确性。例如，Babylon Health 的 AI 医生利用大模型进行症状分析，提供个性化的健康建议和诊疗方案。

技术挑战

(1) 计算资源

大模型需要大量的计算资源和存储空间，如何在智能体中高效部署和运行大模型是一个挑战。当前，训练和运行大模型通常需要高性能计算集群，这对于资源有限的组织来说是一个显著障碍。例如，GPT-3 模型的训练需要数千个 GPU 的计算能力，这使得其应用成本高昂。

(2) 实时性

智能体通常需要实时响应，而大模型的计算复杂度较高，如何保证响应速度是一个关键问题。在实时性要求高的应用场景中，如自动驾驶和智能助手，延迟可能导致严重后果，因此需要优化模型的运行效率。例如，自动驾驶系统必须在毫秒级别做出反应，任何延迟都可能导致危险情况。

(3) 安全与隐私

在使用大模型的过程中，如何保护用户的隐私数据并防止安全漏洞是必须解决的问题。智能体需要处理大量的用户数据，这些数据可能包含敏感信息，因此需要采取有效的措施来防止数据泄露和滥用。例如，医疗应用中的智能体必须严格遵守数据隐私法规，确保患者信息的安全。

(4) 模型解释性

大模型往往是黑箱模型，如何提高其解释性以便智能体能够在关键决策过程中提供透明和可信的解释是一个重要的研究方向。提升模型的解释性，有助于用户理解智能体的决策过程，增加信任度。例如，在金融领域，投资决策系统需要提供明确的理由和依据，以便投资者理解

和信任其建议。

未来展望

(1) 更高效的模型架构

未来，探索更高效的模型架构，以降低计算资源需求并提高响应速度，是一个重要方向。研究人员可以通过模型压缩、量化和蒸馏等技术，优化大模型的性能，使其在智能体中更易部署和运行。例如，模型蒸馏技术可以将大型模型的知识压缩到较小的模型中，从而减少计算需求。

(2) 自适应与迁移学习

提高智能体的自适应学习能力，使其能够在不同任务和环境中灵活应用，是未来的重要研究方向。通过迁移学习和终身学习技术，智能体可以不断积累和利用经验，提升其通用性和适应性。例如，自适应学习技术可以帮助智能体在新领域中快速学习和应用，从而扩展其应用范围。

(3) 多模态智能体

将文本、图像、音频等多种模态结合，提升智能体的综合理解和交互能力，是未来的发展趋势。多模态学习可以帮助智能体更全面地理解环境和用户需求，提高其交互体验和应用效果。例如，多模态智能体可以结合语音识别、图像识别和自然语言理解，实现更复杂的智能任务。

(4) Formal-LLM 框架的发展

Formal-LLM 框架在提高智能体可靠性和确定性方面具有重要的潜力。未来，研究人员可以进一步探索 Formal-LLM 框架的理论基础和实际应用，推动其在智能体设计和开发中的广泛应用。例如，结合 Formal-LLM 框架和深度强化学习技术，可以设计出更安全和可靠的自主智能体，应用于复杂的实时决策和控制任务。

大模型与智能体的结合为智能系统的发展带来了巨大的机遇和挑战。通过结合先进的自然语言处理技术和智能体的自主决策能力，可以实现更智能、更高效的应用。然而，要充分发挥大模型与智能体结合潜力，需要解决诸多技术挑战，如计算资源、实时性和安全性等问题。未来，随着技术的不断进步和理论的深入研究，大模型与智能体的结合将在更多领域展现出其独特的价值和前景。



CHAPTER 2

第 2 章 大模型对话交流

2.1 大模型和搜索引擎的区别

大语言模型（LLM）的理论基础和架构

大语言模型（LLM），如 GPT 和 BERT 系列，是建立在深度学习技术上的自然语言处理工具，主要通过深层神经网络来理解和生成人类语言。这些模型通常基于 Transformer 架构，利用自注意力机制处理大量的文本数据，学习语言的细微结构和语义关系。通过在大规模数据集上进行预训练，这些模型能够捕捉复杂的语言规律，并通过微调过程适应特定的任务或领域。

搜索引擎的基本工作机制

相对于 LLM，搜索引擎如 Google 或 Bing，主要依靠索引和检索机制来处理用户查询。搜索引擎通过爬虫程序定期扫描和索引互联网上的内容，将这些信息组织在一个巨大的数据库中。用户的查询通过与这些索引进行匹配来找到相关的结果，然后根据算法确定的相关性和权威性进行排名。

技术差异及其对实际应用的影响

- 生成 vs 检索：大语言模型（LLM），如 GPT 系列，通过自注意力机制和深度学习的方法，不仅能够理解语言模式，还能够基于这些模式生成新的文本。这种模型的训练涉及大规模的文本数据，通过这些数据，模型学习如何预测下一个词或短语，从而能够生成连贯的句子和段落。

例如，当提出一个问题或主题时，LLM 可以创造性地构造答案，而不是简单地从预存的数据库中检索答案。相比之下，搜索引擎依赖于创建和维护广泛的索引数据库，这些数据库包含了互联网上的页面和文档链接。用户的查询通过算法与这些索引匹配，返回最相关的已存在网页。这种技术的核心是信息检索和相关性评分算法，如 PageRank，它根据链接之间的关系来评估网页的重要性。

- 处理动态信息的能力：搜索引擎的一个显著优势是其能够快速响应新信息。通过不断地爬网和索引新内容，搜索引擎能够提供几乎实时的数据访问。这对于需要最新信息的用户来说是极为重要的，如新闻更新或市场变动。而大语言模型则通常依赖于定期的、批量的训练周期。这意味着如果模型自上次训练以来网上出现了新的发展或数据，这些信息不会立即反映在模型的响应中。虽然有技术如增量学习被探索以缩短信息的延迟，但这些方法通常没有搜索引擎那样的即时性。

- 信息的权威性和准确性：搜索引擎通过显示链接的来源和内容提供者，帮助用户判断信息的权威性。这一点对于评估内容的可信度至关重要，特别是在处理健康、法律或其他需要精确数据的查询时。相对地，大语言模型生成的内容虽流畅且符合语境，但它们的输出并没有直接的来源链接，这可能导致验证信息真实性的困难。因此，使用 LLM 输出时，通常需要用户或额外的系统来进行事实核验，尤其是在输出可能影响重大决策或观点形成的情况下。

- 交互性：由于 LLM 的设计初衷是模拟人类的交流方式，它们特别适合于需要持续对话和用户交互的应用。这使得 LLM 在如虚拟助手、自动化客服和交互式娱乐等场景中表现出色。用户可以以自然语言与这些系统交流，系统则能根据上下文生成响应，提供一种更加动态和个性化的交互体验。而搜索引擎则优化了对特定查询的快速和精确回应，特别是当用户需要快速定位信息或答案时。这种交互虽然有效，但通常缺乏个性化和对话连续性。

大模型与搜索引擎的应用场景区分

我们下面列举几个典型的应用场景，以帮助区分大模型和搜索引擎的使用情况：

1. 搜索引擎的应用场景：

- (a) 实时新闻更新与监控：搜索引擎技术非常适合于新闻门户和媒体监控系统，这些系统需不断更新最新的全球新闻和信息。搜索引擎可以快速索引新发布的内容，提供即时的搜索结果，确保用户能够获得最新的新闻报道和信息。

- (b) 市场研究和竞争分析：在进行市场研究或竞争分析时，搜索引擎能够快速聚合和检索

大量的行业报告、市场分析文档和相关数据。这有助于企业迅速获得关键市场动态，从而做出基于数据的战略决策。

(c) 健康和医疗信息查询：在需要快速访问大量健康和医疗资料的应用中，如在线医疗咨询平台，搜索引擎技术可以提供精确的医学信息和研究结果，帮助医疗专业人员和患者获取权威的健康信息。

2. 大模型的应用场景：

(a) 虚拟客服代理：在需要高度个性化服务的客服系统中，大语言模型可以根据用户的历史交互数据生成针对性的回答，提供定制化的客户服务。这种模型可以通过持续的对话理解和回应用户的具体需求，从而增强客户满意度和忠诚度。

(b) 教育和培训模拟：大语言模型适合被应用于教育和培训领域，如模拟不同的交流情景进行语言学习或专业技能培训。通过与模型的互动，学习者可以获得即时反馈和指导，这有助于提升学习效率和质量。

(c) 互动娱乐和游戏：在视频游戏和互动娱乐中，大语言模型能够根据玩家的选择和游戏进程生成动态对话和故事情节，提供更为沉浸和个性化的游戏体验。这使得游戏环境更加丰富多彩，能够适应玩家的不同偏好和行为。

2.2 大模型的提示词使用技巧

在自然语言处理（NLP）领域，大型语言模型（Large Language Models, LLMs）如 GPT-4 的出现，极大地提升了文本生成、理解和处理的能力。这些模型通过大量数据和复杂算法，能够生成高质量的自然语言文本，模拟人类的语言行为，并在各种应用中展示出卓越的性能。近年来，LLMs 在自动化内容创作、机器翻译、对话系统、情感分析等方面取得了显著的进展，其应用前景广阔。

然而，要最大化这些模型的潜力，仅仅依靠其强大的计算能力是不够的。用户需要掌握一系列基础提示词使用技巧，以充分发挥模型的功能。这些技巧不仅能提高模型生成文本的准确性和相关性，还能显著提升用户与模型互动的体验质量。具体来说，这些技巧涉及明确查询、提供上下文信息、分步查询、使用关键词和短语、利用专业术语、检验信息的准确性、结合工具和资源、多次迭代查询、避免歧义和双关、以及优化提示词长度和结构。

2.2.1 基础技巧

本章节旨在详细介绍这些大模型提示词使用的基础技巧，并结合具体例子进行解释。通过对这些技巧的系统性阐述，帮助用户更高效地与大型语言模型互动，从而在各类复杂任务中获得更优质的结果输出。这不仅对于初学者有指导意义，对于专业研究人员和行业从业者同样具有重要的参考价值。通过掌握和应用这些技巧，用户可以更好地利用大型语言模型的潜力，推动自然语言处理技术的发展和應用。

1. 明确的查询

技巧说明

明确的查询能够帮助模型更好地理解用户的意图，从而生成更准确的响应。模糊或不完整

的查询往往会导致不精确或不相关的回答。因此，提出具体和详细的问题是使用提示词的首要技巧。

示例

- 模糊查询：“告诉我一些关于机器学习的东西。”
- 明确查询：“请解释一下监督学习和无监督学习的区别，并举例说明每种方法的应用场景。”

在明确查询的例子中，用户指定了需要解释的内容（监督学习和无监督学习的区别）和需要的细节（应用场景），这使得模型能够提供更有针对性的回答。

2. 提供上下文信息

技巧说明

提供上下文信息有助于模型理解问题的背景，从而生成更相关的回答。上下文信息可以包括之前的对话内容、特定领域的背景知识或问题的具体背景。

示例

- 无上下文信息：“如何优化代码？”
- 有上下文信息：“在 Python 编程中，如何优化处理大数据集的代码以提高执行效率？”

在有上下文信息的例子中，用户明确了需要优化的是 Python 代码，并且目标是处理大数据集，这使得模型能够提供更具体的优化建议。

3. 分步查询

技巧说明

对于复杂的问题，可以将其分解成多个简单的小问题逐个询问。这种方法不仅能提高每个步骤的回答质量，还能帮助用户更好地理解每个环节的细节。

示例

- 复杂查询：“解释一下深度学习的工作原理，并告诉我如何在图像识别中应用它。”
- 分步查询：
 1. “什么是深度学习？”
 2. “深度学习的基本工作原理是什么？”
 3. “如何在图像识别中应用深度学习？”

通过分步查询，用户可以逐步获取所需的信息，每个回答都更加详细和易于理解。

4. 使用关键词和短语

技巧说明

在提示词中明确使用关键词和短语，可以帮助模型抓住核心信息，生成更精确的回答。特

别是在进行数据搜索或特定信息检索时，关键词和短语的使用尤为重要。

示例

- 一般查询：“关于数据库的信息。”
- 关键词查询：“请介绍一下关系型数据库和非关系型数据库的主要区别。”

在关键词查询的例子中，“关系型数据库”和“非关系型数据库”是关键词，使得模型能够针对性地提供相关信息。

5. 使用专业术语

技巧说明

在特定领域的查询中，使用准确的专业术语可以提高模型的响应质量。专业术语能帮助模型识别问题的专业性，从而提供更高质量的回答。

示例

- 非专业术语：“告诉我一些关于网站安全的事情。”
- 专业术语：“请解释一下跨站脚本攻击（XSS）和 SQL 注入攻击的防御方法。”

在使用专业术语的例子中，“跨站脚本攻击（XSS）”和“SQL 注入攻击”是网络安全领域的专业术语，使得模型能够提供更深入和专业的解答。

6. 检验信息的准确性

技巧说明

对于从模型获取的信息，尽可能从多个不同的信源进行验证，确保其准确性和可靠性。这对于涉及快速变化的领域（如科技、医疗等）的查询尤为重要。

示例

- 模型回答：“最新的研究表明，量子计算机将在未来五年内取代经典计算机。”
- 验证信息：用户可以使用多个学术数据库、新闻来源和专业期刊来验证这一说法的真实性和准确性。

通过多角度验证，用户可以确保从模型获取的信息是可信和准确的。

7. 利用工具和资源

技巧说明

结合其他工具和资源（如数据库访问、专业网站等）来验证或深化从模型获取的答案，可以提高信息的质量和准确性。

示例

- 模型回答：“在 Python 中，可以使用 Pandas 库来处理大数据集。”

- 验证和扩展：用户可以参考 Pandas 官方文档或访问 Stack Overflow 等编程社区，获取更多关于使用 Pandas 库的详细信息和实际案例。

这种结合使用多种资源的方法，可以帮助用户更全面地了解问题，并获得更可靠的解决方案。

8. 多次迭代查询

技巧说明

在复杂或多层次的问题上，通过多次迭代查询可以逐步获得更完整和详细的信息。每次查询都基于之前的回答进行扩展和深入。

示例

- 初始查询：“什么是机器学习？”
- 第一次迭代：“机器学习有哪些主要类型？”
- 第二次迭代：“监督学习的工作原理是什么？”
- 第三次迭代：“请详细解释一下线性回归在监督学习中的应用。”

通过多次迭代查询，用户可以逐步深入了解一个复杂的主题，每次迭代都基于之前的回答进行扩展和细化。

9. 避免歧义和双关

技巧说明

在提示词中应避免使用可能引起歧义或双关的词语，以确保模型能够准确理解问题的含义。

示例

- 歧义查询：“银行的安全措施有哪些？”
- 明确查询：“金融机构中的银行为了保护客户数据和资金，采用了哪些安全措施？”

在明确查询的例子中，用户明确了讨论的是金融机构的安全措施，而不是河流的堤坝或其他含义。

10. 提示词长度和结构

技巧说明

提示词的长度和结构会影响模型的理解和响应质量。一般来说，简洁且结构清晰的提示词更易于模型处理和生成高质量的回答。

示例

- 冗长查询：“我想知道在 Python 编程中，有哪些方法可以用来处理大数据集，因为我听说 Pandas 库非常流行，所以想了解一下。”
- 简洁查询：“在 Python 中，如何使用 Pandas 库处理大数据集？”

简洁查询不仅减少了冗余信息，还使得模型能够快速抓住关键词，生成高效回答。

总结

总而言之，掌握大模型提示词使用的基础技巧，可以极大地提高用户与模型交互的效率和效果。通过明确查询、提供上下文信息、分步查询、使用关键词和专业术语、检验信息准确性。

2.2.2 LLM 的 prompt 提示词基础使用技巧

随着大型语言模型（LLM）如 ChatGPT、GPT 系列等技术的飞速发展，如何高效利用这些模型成为了许多人关注的焦点。本章将深入探讨大模型提示词基础使用技巧，包括角色扮演、语气与角度调控、具体任务设定以及上下文关联等，旨在帮助读者更好地掌握与大模型的交互方式，提升工作效率与输出质量。

1. 角色扮演：专业视角的塑造

在大模型的使用中，角色扮演是一种极具创造性的方法，能够显著提升回答的专业性和针对性。通过明确指定大模型所扮演的角色，如专业厨师、金融分析师或论文评审专家，我们可以引导其以该角色的专业知识和视角来回答问题。这种技巧不仅增强了回答的权威性，还使得信息更加贴近实际需求，提高了信息的实用性和价值。

2. 语气与角度的灵活调控

大模型能够根据指令调整回复的语气和角度，这一特性为多样化的信息表达提供了可能。通过要求大模型使用严肃正式或友好轻松的语气，我们可以获得风格迥异的回答内容。更进一步地，结合角色扮演，如“请扮演一名科普博主，使用友好轻松的语气介绍相对论”，可以使得回答更加生动有趣，易于理解。此外，通过调整角度，如“向 5 岁的孩子解释相对论”，大模型能够用简单易懂的方式传达复杂概念，满足不同受众的需求。

3. 具体任务的明确设定

为了从大模型中获取高质量的输出，设定具体明确的任务至关重要，这包括指定字数、时间范围、格式要求等细节信息。通过详细指令，如“帮我写一篇关于 AI 发展的报告，字数在 1000 字以内”，我们可以确保大模型明确理解任务要求，并尽可能按照预期生成内容。值得注意的是，虽然大模型可能无法精确控制字数到每一个字，但它们会努力在给定范围内提供内容。

在设定具体任务时，了解大模型对输出的限制同样重要。目前，大多数大模型（包括 GPT 系列）的输出长度限制在约 4000token（或更精确的 4096token）左右，这相当于大约 2000 字的中文内容。然而，不同模型之间可能存在差异，如 GPT-4 通过优化 token 计算，能够在相同数

量的 token 下输出更多内容，大约可达 2800 字。这一知识点有助于我们在设定任务时更加合理地规划内容长度，避免超出模型的处理能力。

4. 上下文关联的深度对话

大模型的上下文关联能力是其强大功能之一，它允许模型记住之前的对话内容，并在后续对话中加以利用。这一特性在连续写作、深入讨论等场景中尤为重要。通过简单的指令，如“继续写”或“就刚刚提到的 XX 点进一步说明”，我们可以引导大模型在已有内容的基础上继续创作或深入讨论。此外，大模型的回复结果具有一定的随机性，通过要求“再给我更多一些其他的答案”，我们可以获得多样化的回答选项，满足不同的需求。

在优化回答时，明确具体的修改建议同样重要。例如，当大模型的解释难以理解时，我们可以要求“换一种更容易理解的方式再说一次”，并尽可能提供具体的优化方向，以便大模型能够更准确地调整回复内容。

总结

通过掌握角色扮演、语气与角度调控、具体任务设定以及上下文关联等技巧，我们可以更加高效地利用大模型进行工作和学习。这些技巧不仅提升了我们的工作效率和输出质量，还为我们提供了更多元化的思考和表达方式。在未来的日子里，随着大模型技术的不断发展和完善，我们有理由相信这些技巧将发挥更加重要的作用，助力我们在各个领域取得更加卓越的成就。

2.3 大模型的提示词高级技巧

思维链定义

近些年来大语言模型（LLM）的效果肉眼可见的变好，同时随着模型规模的不断增大，模型也变得更好“提示”，尤其是之前一些没有办法做的很好的任务不断取得突破。但是大模型在做算术推理、常识推理和符号推理时的表现还不够好。怎么结合上下文和中间步骤来改善算术推理、常识推理和符号推理等能力是一个问题。思维链的一系列工作就是在这样的大环境下诞生的。尽管增加大语言模型的规模可以提高性能，但在涉及算术、常识和符号推理等具有挑战性的任务上，仅仅增加模型规模是不够的。为了解决这个问题，通过使用思维链提示给模型提供一系列中间的自然语言推理步骤，从而引导模型进行推理。

思维链 (Chain-of-thought, CoT) 的概念是在 Google 的论文“Chain-of-Thought Prompting Elicits Reasoning in Large Language Models”中被首次提出。思维链 (CoT) 是一种改进的提示策略，用于提高 LLM 在复杂推理任务中的性能，如算术推理、常识推理和符号推理。通过思维链提示我们迫使 LLM 输出一系列中间步骤，从而得出所需的答案。它是有益的，因为它允许模型一次专注于解决一个步骤，而不是必须一次考虑整个问题，对于一步很难或不可能解决的复杂问题特别有帮助。它提供了一个了解模型行为的可解释窗口，我们可以通过遵循模型所采取的步骤序列来了解模型如何得出答案。简单来说，思维链是一种离散式提示学习，更具体地，大模型下的上下文学习（即不进行训练，将例子添加到当前样本输入的前面，让模型一次输入这些文本进行输出完成任务），相比于之前传统的上下文学习（即通过 $x_1, y_1, x_2, y_2, \dots, x_{test}$ 作为输入来让大模型补全输出 y_{test} ），思维链多了中间的中间的推导提示，以图 1.1 为例。

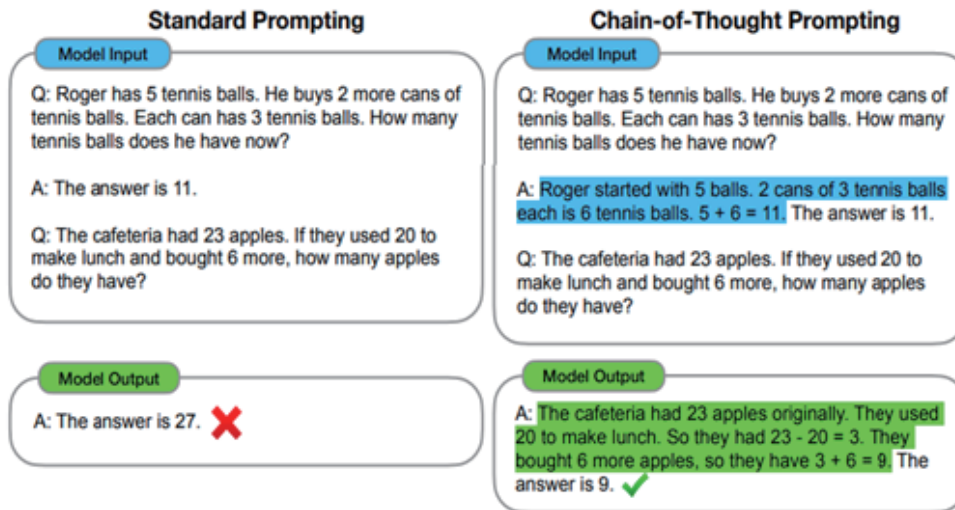


图 2.1: 思维链提示

可以看到，类似的算术题，思维链提示会在给出答案之前，还会自动给出推理步骤：“罗杰先有 5 个球，2 罐 3 个网球等于 6 个， $5+6=11$ ”“食堂原来有 23 个苹果，用 20 个做午餐， $23-20=3$ ；又买了 6 个苹果， $3+6=9$ ”。思维链提示给出了正确答案，而直接给出答案的传统提示学习，结果是错的，连很基本的数学计算都做不好。简单来说，语言模型很难将所有的语义直接转化为一个方程，因为这是一个更加复杂的思考过程，但可以通过中间步骤，来更好地推理问题的每个部分。

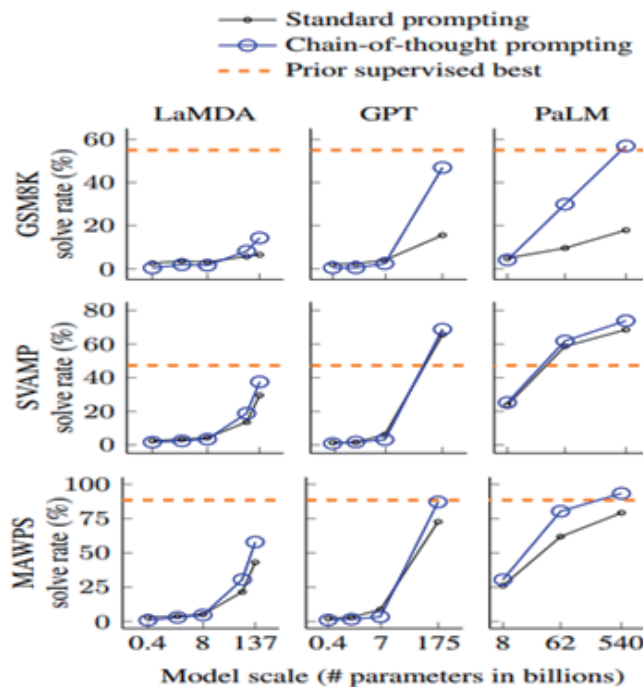


图 2.2: Chain-of-thought prompting enables large language models to solve challenging math problems

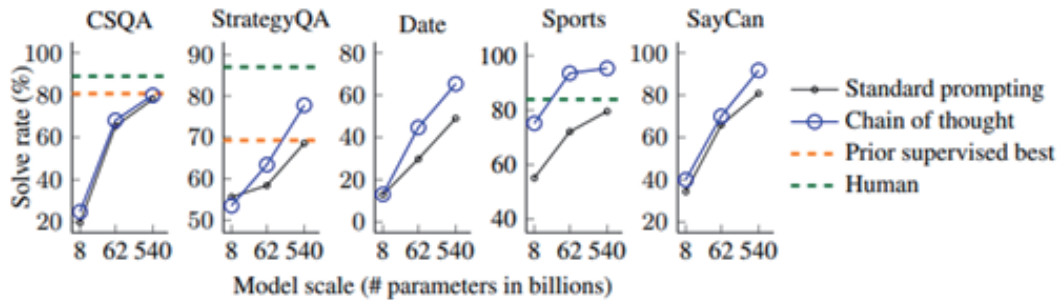


图 2.3: Chain-of-thought prompting also improves the commonsense reasoning abilities of language models

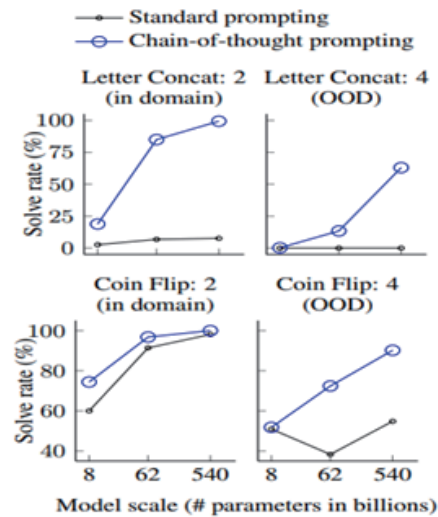


图 2.4: Using chain-of-thought prompting facilitates generalization to longer sequences in two symbolic reasoning tasks

从图 1.2-1.4 可以直观的看出，思维链提示在运算推理、常识推理、符号推理等方面显著的性能提升能力。

需要注意的是，CoT 是一种涌现能力。也就是说，CoT 对小模型作用不大，模型参数至少达到 10B 才有效果，达到 100B 效果才明显。并且，从小模型的输出可以看出，它们大部分是输出了流畅但不合逻辑的 CoT，因此得到错误的结果。CoT 对复杂的问题的性能增益更大，例如 GSM8K（更难，因为基线最低）上 GPT-3 和 PaLM 的性能增加了一倍多。而对于 MAWPSSingleOp（更简单的任务），性能改进非常小甚至是负面的。加上 CoT 的 PaLM 540B 超过了任务特定的用监督学习训练的模型的最优结果。不加 CoT 的话，GSM8K 和 MAWPS 任务上 LLM 的结果比不过最优的监督学习模型。

一个有效的思维链应该具有以下特点：

- 逻辑性：思维链中的每个思考步骤都应该是有逻辑关系的，它们应该相互连接，从而形成一个完整的思考过程。

- 全面性：思维链应该尽可能地全面和细致地考虑问题，以确保不会忽略任何可能的因素和影响。
- 可行性：思维链中的每个思考步骤都应该是可行的，也就是说，它们应该可以被实际操作和实施。
- 可验证性：思维链中的每个思考步骤都应该是可以验证的，也就是说，它们应该可以通过实际的数据和事实来验证其正确性和有效性。

思维链的好处和局限

思维链是解决推理任务时人类思维过程遵循的一系列典型步骤。它可以帮助我们将一个问题分解成一系列的子问题，然后逐个解决这些子问题，从而得出最终的答案。在大型语言模型中，思维链可以用来引出推理。思路链方法带来以下好处：

- CoT 允许模型将多步推理问题分解为中间步骤，这意味着额外的计算可以分配到需要推理的复杂问题上。
- CoT 使大语言模型更具可解释性，更加可信，并提供了调试推理路径错误的机会。
- CoT 推理能够被用于数学应用题、常识推理和符号操作等任务，并且可能适用任何人类需要通过语言解决的问题。
- CoT 可以通过将其加入到 few-shot prompting 示例中，从而在足够大的语言模型中引导出推理能力。

当前的思维链也存在着许多局限性：

- 首先，尽管设计的思维链是在模拟人类的推理过程，但模型是否真正的学会了推理仍需进一步进行验证。
- 人工设计思维链仍然是代价过大，大规模的人工标注思维链是不可行的。
- 思维链只在大规模模型上有效（10B 以上）。

在自然语言处理和语言模型领域，有两种主要的思维链提示类型：零样本思维链提示（Zero-shot-CoT）和多样本思维链提示（Few-shot-CoT）。这些提示帮助语言模型在不同任务或语境中生成连贯和合理的语言。

零样本思维链提示

零样本思维链提示（Zero-shot-CoT）：只需要在问题的结尾里放一句“请逐步思考”或者“让

我们一步一步地思考”就可以提高回答问题的准确度，激发大语言模型的推理能力。当我们向模型提一个逻辑推理问题时，模型返回了一个错误的答案，但如果我们在问题最后加入“让我们一步一步地思考”这句话之后，模型就生成了正确的答案。

具体例子如下：

零样本提示：

【模型输入】

【问题：一名杂技演员可以同时抛接 16 个球。其中一半的球是高尔夫球，而这些高尔夫球中又有一半是蓝色的。那么有多少个蓝色的高尔夫球呢？】

【模型回复】

【答案是 8。】

很显然，这个回答是不正确的。

零样本思维链提示：

【模型输入】

【问题：一名杂技演员可以同时抛接 16 个球。其中一半的球是高尔夫球，而这些高尔夫球中又有一半是蓝色的。那么有多少个蓝色的高尔夫球呢？让我们一步一步地思考。】

【模型回复】

【答案：总共有 16 个球。其中一半是高尔夫球，也就是说有 8 个高尔夫球。而这些高尔夫球中又有一半是蓝色的，也就是说有 4 个蓝色的高尔夫球。

模型回复的这个结果才是正确的。

ChatGPT 这类统计语言模型基于过去的数据进行预测和输出。它通过统计学方法分析数据中的逻辑关系，并将这些逻辑呈现为回答，给人以有逻辑性的感觉。在计算过程中，模型进行多个假设运算，例如解决问题从 A 到 B 再到 C，中间包含多个假设。如果模型在第一次计算时出现错误，通常是因为它跳过了中间的某些步骤（比如 B），导致答案不准确。通过逐步思考，模型能够按照完整的逻辑链（A > B > C）进行计算，避免跳过重要假设，从而得出更准确的答案。这个技巧，除了用于解决复杂问题外，还适合生成一些连贯主题的内容，比如写长篇文章、电影剧本等。

零样本思维链的另一种用法为不仅仅只是让 AI 按步骤行事，还会告知 AI 每一步要做什么。通过将任务拆解，能让 AI 生成的结果更加符合要求，提升 AI 的回答准确性。比如这个案例：

【模型输入】

【判断学生的解答是否正确。

问题：我正在建设一个太阳能发电设施，需要帮助计算财务数据。土地价格为每平方英尺 100 美元，太阳能电池板价格为每平方英尺 250 美元。我谈判了一份维护合同，第一年的费用为 10 万美元，此后每平方英尺 10 美元。请问第一年运营的总成本与设施的平方英尺数之间的关系是什么？

学生的解答：

设 x 为发电设施的面积（单位：平方英尺）。

成本：

1. 土地成本： $100x$

2. 太阳能电池板成本： $250x$

3. 维护成本： $100,000 + 100x$

总成本： $100x + 250x + 100,000 + 100x = 450x + 100,000$ 】

【模型回复】

【学生的解答基本正确。他们正确地计算了各项成本：土地成本： $100x$ （每平方英尺 100 美元），太阳能电池板成本： $250x$ （每平方英尺 250 美元）；

维护成本： $100,000 + 10x$ （第一年的固定费用为 10 万美元，之后每平方英尺 10 美元）。然后，他们将这些成本相加得到了总成本： $450x + 100,000$ 。

因此，学生的解答是正确的，总成本与设施的平方英尺数 x 之间的关系为 $450x + 100,000$ 美元。】

大模型的回答是「学生的解答正确」，但其实学生的答案是错误的，正确答案应该是 $360x + 100,000$ ，我们将提示语调整成这样：

【模型输入】

【你的任务是判断学生的解答是否正确。

要解决这个问题，请执行以下操作：

首先，自己解决这个问题。然后将您的解答与学生的解答进行比较，评估学生的解答是否正确。在您自己解决问题之前，请不要判断学生的解答是否正确。

请使用以下格式：

问题：

##

问题内容

##

学生的解答：

##

学生的解答内容

##

实际解答：

##

解答步骤和您的解答内容

##

学生的解答是否与刚刚计算出的实际解答相同：

##

是或否

##

学生评分：

##

正确或错误

##

问题：

##

我正在建立一个太阳能发电设施，我需要帮助计算财务数据。

土地成本为每平方英尺 100 美元，我可以购买每平方英尺 250 美元的太阳能电池板。我谈判了一份维护合同，将花费我每年固定 10 万美元，以及每平方英尺 10 美元的额外费用。请问第一年运营的总成本作为平方英尺数量的函数是多少？

##

学生的解答：

##

设 x 为发电设施的面积（单位：平方英尺）。

成本：

1. 土地成本： $100x$
2. 太阳能电池板成本： $250x$
3. 维护成本： $100,000 + 100x$

总成本： $100x + 250x + 100,000 + 100x = 450x + 100,000$

##

实际解答：】

【模型回复】

【实际解答：

设 x 为发电设施的面积（单位：平方英尺）。

成本：

土地成本：每平方英尺 100 美元，总计为 $100x$ 美元。

太阳能电池板成本：每平方英尺 250 美元，总计为 $250x$ 美元。

维护成本：第一年固定费用为 10 万美元，之后每平方英尺 10 美元，总计为 $100,000 + 10x$ 美元。

因此，第一年运营的总成本作为平方英尺数量 x 的函数是：

总成本 = $100x + 250x + 100,000 + 10x = 360x + 100,000$ 美元。

学生的解答是否与刚刚计算出的实际解答相同：

否

学生评分：

错误】

这个解答内容是正确的。

多样本思维链提示

多样本思维链提示（Few-shot-CoT）：指的是模型通过少量示例或样本来学习适应特定任务或语境，进而生成更加精确和专业的语言输出。多样本思维链提示会提供一些如何解决类似问题的例子，这些例子的呈现方式鼓励大模型思考问题并提出生成答案的一系列思想。在提问问题时，添加几个回答范例，可以将问题答案固定为某个格式。具体案例如下：

【模型输入】

【这组数据中的中位数是否小于等于 10:17、9、10、12、13、4、2】

仅仅输入这个问题的话，模型的回答有时候是错误的有时候是正确的，并且回答的格式各不相同。这是由大模型回答问题的格式的随机性决定的，比如答案的第一个字是“要”的概率为 50%，是“为”的概率为 30%，是“首”的概率为 3% 等等。

采用多样本思维链提示后：

【模型输入】

【按照下面的格式回答问题,不要添加额外的内容,接着 A 后面的内容输出。

【这组数据中的中位数是否小于等于 20:35、18、22、20、25、30、15。

A: 计算中位数,先对数字进行排序(15、18、20、22、25、30、35),所以中位数为 22。答案为 False。

这组数据中的中位数是否小于等于 5:3、5、1、7、9。

A: 计算中位数,先对数字进行排序(1、3、5、7、9),所以中位数为 5。答案为 True。

这组数据中的中位数是否小于等于 12:25、10、15、8、20、18。

A: 计算中位数,先对数字进行排序(8、10、15、18、20、25),所以中位数为 16.5。答案为 False。

这组数据中的中位数是否小于等于 10:17、9、10、12、13、4、2。

A: 】

【模型回复】

计算中位数,先对数字进行排序(2、4、9、10、12、13、17),所以中位数为 10。答案为 True。

采用多样本思维链提示后,大模型不仅可以按照给定格式回答问题,而且回答问题的准确率获得了显著提升。

自洽性 (Self-consistency)

自洽性是对思维链 (CoT, Chain of Thought) 的一个补充,取代了思维链提示中使用的贪婪解码策略,其想法是首先用思维链提示来提示语言模型,然后提出了一个“采样和边缘化”的解码过程,而不是贪婪地解码最优推理路径:

先从语言模型的解码器中采样,以生成一组多样化的推理路径;由于语言模型不是完美的推理器,模型也可能产生错误的推理路径或者在某一个推理步骤中出错,因此每个推理路径都可能导致不同的最终答案,但错误的方案不太可能得到相同的答案,也就是说,假设推理过程正确时,即使它们是多样化的,在最终答案中往往比不正确的推理过程具有更高的一致性,所以通过边缘化采样推理路径来确定最佳答案,以在最终答案集中找到最一致的答案。

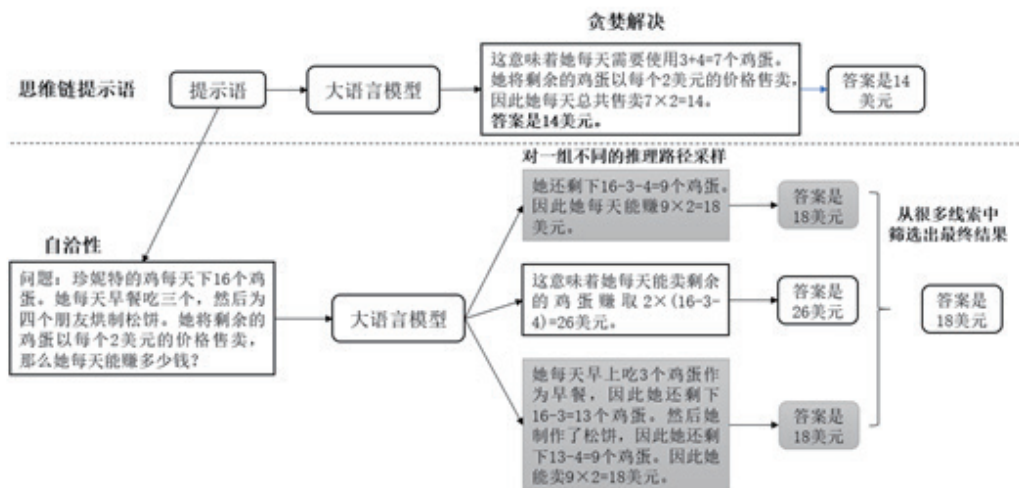


图 2.5: 自洽性图解

这种方法类似于人类的经验，如果多种不同的思维方式导致相同的答案，那么人们就会有信心最终的答案是正确的。与其他解码方法相比，自洽性避免了困扰贪婪解码的重复性和局部最优性问题，同时减轻了单个采样生成的随机性，这有助于提高 CoT 提示在涉及算术和常识推理的任务中的性能。

例如，我们考虑以下多步推理问题：

问题：如果一家商店有 10 个苹果和 8 个橙子，此时店里卖出了 6 个苹果和 4 个橙子，那么店里还剩下多少水果？

思维链提示（chain-of-thought prompting）不是直接回答问题，而是要求语言模型生成一系列模仿人类推理过程的短句：

商店有 10 个苹果。

店里有 8 个橙子。

卖了 6 个苹果。

卖了 4 个橙子。

还剩下 $10 - 6 = 4$ 个苹果。

剩下 $8 - 4 = 4$ 个橙子。

商店现在有 4 个苹果 + 4 个橙子 = 8 个水果。

使用自洽性，语言模型生成多个推理路径：

推理路径 A：

商店有 10 个苹果。

店里有 8 个橙子。

卖了 6 个苹果。

卖了 4 个橙子。

还剩下 $10 - 6 = 4$ 个苹果。

剩下 $8 - 4 = 4$ 个橙子。

商店现在有 4 个苹果 + 4 个橙子 = 8 个水果。

推理路径 B：

商店最初有 10 个苹果和 8 个橙子。

商店卖了 6 个苹果，所以还剩下 $10 - 6 = 4$ 个苹果。

商店卖了 4 个橙子，所以还剩下 $8 - 4 = 4$ 个橙子。

商店现在有 4 个苹果 + 4 个橙子 = 8 个水果。

自洽性从生成的多个推理路径中选择最一致的答案。在这个例子中，推理路径 A 和 B 都得出相同的答案（8 个水果）。可以看到这种做法增加了模型对最终答案正确的信心。

除了准确性的提高之外，自洽性还有助于在使用语言模型执行推理任务时收集基本原理，以及提供不确定性估计和改进语言模型输出的校准。

生成知识提示 (Generated Knowledge Prompting)

生成知识提示法 (GKP) 依托于大型语言模型 (LLM) 的强大功能，旨在创造出能够解决特定问题的知识。这个方法的核心思想是，通过 LLM 生成有价值的知识信息，然后把这些信息作为提示内容，和一个相关问题结合起来，用于解决具体的任务。

知识通常指的是可以被算法或模型利用来执行任务或做出决策的信息。它可以采取多种形式，包括数据，事实，规则，关系，模式，概念和经验：

事实知识：例如，“巴黎是法国的首都。”这是一个明确的事实，可以被用来回答关于地理或政治的问题。

程序性知识：如编程语言的语法规则或者骑自行车的步骤。这些知识说明了如何执行特定的任务。

启发式知识：在解决问题时，经验法则或简单规则可以指导决策过程。例如，在棋类游戏中，“控制棋盘中心”通常是一个有用的策略。

关系知识：诸如“所有的猫都是哺乳动物”这样的陈述，表明了猫和哺乳动物之间的分类关系。

模式知识：通过数据分析发现的趋势，例如股市历史数据中的交易模式。

概念知识：对一类事物的一般理解，例如“民主”或“可持续发展”的定义。

经验知识：个人或集体的经验，如医生对疾病症状的识别。

这些知识形式可以通过不同方式获取和应用，例如通过训练数据集学习，通过知识库直接编码，或者通过模型的推理能力来生成和应用。知识是智能系统的核心组成部分，它使得这些系统能够理解复杂的概念，解决问题，并做出有根据的决策。

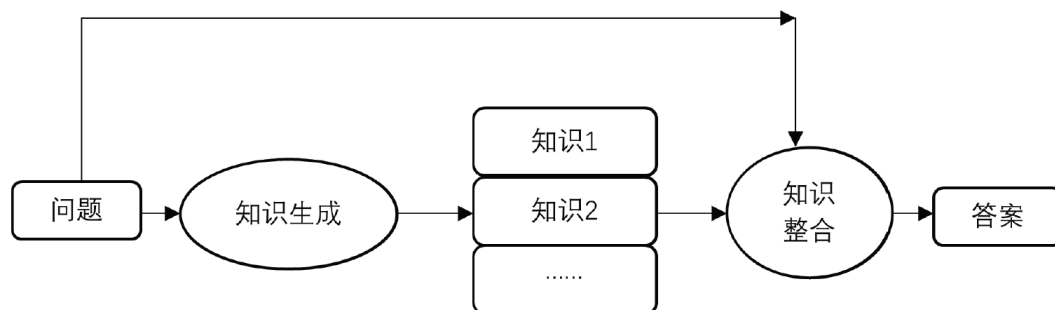


图 2.6: 生成知识提示图解

生成知识提示技术本质上是提高模型的推理能力及准确度，模型本身做的是预测工作，所以在预测之前添加一些提示可以直接影响预测结果，我们可以利用这一特点，引导模型产生稳定以及具有一致性的答案。

我们尝试一个简单的提示来看看对于常识推理等任务，生成知识提示的方法有多大帮助：

先给出一个零样本提示的问题：

【模型输入】

【“高尔夫球的一部分是试图获得比其他人更高的得分”这句话对不对，你只需回答对或不对。】

【模型回复】

【对】

这个回答是错误的。这种错误，说明在知识的推理上，模型存在局限性或者说不确定性，问题如果存在误导，那么其回答也会出现错误的情况，甚至出现事实性错误。

接下来，生成一些知识：

【模型输入】

【产生一些关于“高尔夫球的一部分是试图获得比其他人更高的得分”的知识。】

【模型回复】

【这句话其实有些误导，因为在高尔夫球中，实际上是希望得分越低越好，而不是越高。高

尔夫球比赛的胜者是那位得分最低的球员或球队。因此，与其他竞技项目不同，高尔夫球的目标是尽量用尽可能少的击球次数完成一轮比赛，从而获得低得分。】

然后在此基础上重新回答问题：

【模型输入】

【高尔夫球的一部分是试图获得比其他人更高的得分这句话对不对，你只需回答对或不对。】

【模型回复】

【不对】

该回答是正确的。总之，生成知识提示通过生成作为提问部分的知识，引用知识或规律而非实例来提问，它不仅确保了模型在面对不同输入时能够保持内部状态或行为的一致性，且在多种场合下都有应用。

这些技巧可以提升大模型的思维能力，当然最好的方式还是使用更好的模型，算术和推理能力与大模型本身的关系最大，对于同一个模型，技巧的使用可以提升模型算术和推理的准确性。

提示词小技巧

- 如果有某个要求特别重要，可以反复强调。比如，在对话过程中对大模型提出了若干要求，有些要求被满足，有些未被满足，可以将未被满足的要求多次重复，使得大模型更加重视这个要求。

- 如果有某个要求特别重要，可以把它写在提示词的最后一句。提示词的最后一句是整个提示词中对大模型影响最大的，因此可以将重要的要求写在提示词的最后一句话。

- 如果你写的好 / 回复得好 / 代码运行正确，我会给你 100 美元的小费。对大模型进行语言激励可以提高模型的效果。

2.4 大模型对话应用示例

翻译

大模型翻译任务是指利用大规模语言模型（如 OpenAI 的 GPT 系列、阿里云的通义千问等）进行跨语言文本转换的过程。这类任务利用了模型在大量数据上训练获得的跨语言理解和生成能力，旨在提供更准确、流畅的翻译结果。相较于传统的基于规则或统计的机器翻译系统，大模型翻译具有以下几个显著特点：

- 上下文理解能力：大模型通过深度学习海量文本数据，获得了理解复杂语境的能力。这意味着在翻译时，它们不仅仅关注单个词汇或短语的直接对应，而是能够捕捉到整个句子乃至段落的语义，确保翻译的准确性和自然流畅度。例如，当遇到多义词时，大模型能够根据上下文确定最恰当的含义，避免歧义，使得翻译更加贴合原文意图。

- 多领域适应性：得益于其庞大的训练数据集涵盖了众多领域，大模型能够应对从日常对话到专业文献的各种翻译需求。这种跨领域的灵活性使得它们在面对专业性强或行业特有的术语和表达时，依然能够提供较为准确的翻译。用户无需为每一个特定领域单独训练模型，降低了翻译的门槛和成本。

- 风格保留与创新：大模型不仅能传递原文的内容，还能在一定程度上模仿原作者的语言风格和情感色彩，使得翻译作品不仅忠于原意，还富有感染力。此外，它们在处理翻译任务时，能够根据目标语言的习惯进行适当的语言创新，使译文更加地道。例如，在翻译文学作品时，大模型能够尝试保留原文的艺术美感和修辞手法。

- 自适应学习与微调：虽然大模型本身已经具备广泛的知识 and 技能，但通过微调，即在特定领域的的数据上对模型进行再训练，可以进一步提升其在该领域的翻译精度。这种自适应学习能力使得大模型能够针对政府公文、法律文档、医疗报告等特定类型的文本进行优化，提高翻译的专业性和准确性。

• 交互式和个性化：大模型支持的交互式翻译允许用户与模型进行实时沟通，提出修改意见或询问特定翻译的合理性，模型可以即时调整翻译策略以满足用户的个性化需求。这种方式提升了翻译服务的互动性和用户体验，使得翻译不再是单一的输出过程，而是双向的交流与合作。

翻译任务实例演示

我们下面演示如何通过大模型完成翻译任务：图 1.1 展示了利用大模型进行简单翻译任务，通过利用【】等分隔符将用户问题标注，能够让大模型更好的完成任务。

[Strong as a man may be, he can hardly achieve anything without others' help. Human beings are to cooperation what a fish is to water. Nobody can live without others. Now I have gotten to understand, to approve of, and finally to be fond of working in groups, and I do learn a lot from the other group members. I strongly ask you to treasure people around you and learn to work with your group. It really matters.]请翻译一下这段话

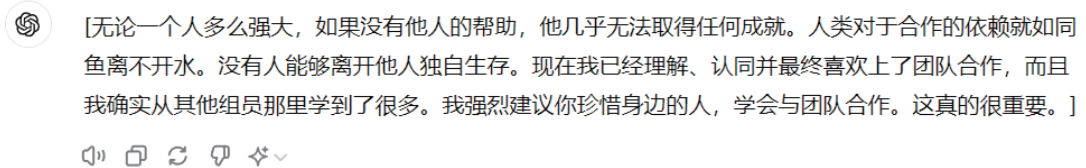


图 2.7: An example of translation

此外，再翻译完成后可再翻译内容基础上继续完成后续任务（图 1.2）。

[Strong as a man may be, he can hardly achieve anything without others' help. Human beings are to cooperation what a fish is to water. Nobody can live without others. Now I have gotten to understand, to approve of, and finally to be fond of working in groups, and I do learn a lot from the other group members. I strongly ask you to treasure people around you and learn to work with your group. It really matters.]请翻译一下这段话，并提炼一下内容

翻译：

尽管一个人可能非常强大，但他几乎不可能在没有他人帮助的情况下完成任何事情。人类对合作的依赖就像鱼离不开水一样。没有人能够离开他人而生存。现在，我已经理解、认同并最终喜欢上了团队合作，而且我确实从其他团队成员那里学到了很多。我强烈建议你珍惜身边的人，学会与团队合作。这真的很重要。

内容提炼：

- 个人难以在没有他人帮助的情况下取得成功。

- 人类需要合作，就像鱼需要水一样。
- 没有人能够独立生存。
- 作者已经理解并喜爱团队合作，并从中学到了很多。
- 作者建议珍惜周围的人，学会与团队合作，因为这非常重要。



图 2.8: An example of translation

大模型翻译与翻译软件的不同

利用大模型完成翻译任务和翻译软件存在显著不同。大模型能够深度理解语言，准确把握复杂的语义和多义表达，更好地适应多样的语境。它能实时学习更新知识，迅速掌握新词汇和术语。还可提供个性化翻译，满足用户独特需求。相比之下，翻译软件较为模式化，语境适应性差，知识更新滞后，难以实现个性化，在处理复杂、专业的翻译任务时往往力不从心。

角色扮演

大模型，特别是指那些基于深度学习的大型语言模型，如 GPT-4、PaLM、Llama 2 等，它们在处理自然语言任务时，能够展示出一定的角色扮演能力。这种能力主要体现在模型能够根据不同的场景和指令，模拟出特定的人物性格、语气或者专业知识背景，进而生成符合该角色身份的文本内容。下面是大模型角色扮演的一些内容和特点：

- **多领域对话模拟**：大模型在模拟不同领域的专家时，不仅能提供基础的理论知识和概念解释，还能深入探讨该领域的前沿研究、争议话题及实践应用。例如，当模拟历史学家时，它能详述历史事件的背景、因果关系及其对现代社会的影响；作为科学家，模型能够解析复杂科研原理，讨论科技进展对社会生活带来的变革；在医疗健康领域，它可以提供疾病预防、诊断流程及最新治疗方法的信息，甚至模拟医生与患者的咨询对话，为用户提供初步的健康咨询建议。

- **文化与时代背景适应**：大模型不仅能适应不同文化背景下的语言习惯和礼仪，还能融入特定时代的语言风格，比如模拟古汉语的文言文交流，或是再现 20 世纪初的西方文学对话。在构建历史场景时，它能精确捕捉到时代特有的词汇、表达方式和思维方式，使对话内容更加真实可信。对于未来设想的构建，大模型则能基于现有科技趋势、社会发展规律进行合理推测，创造出既符合逻辑又富有想象力的未来话语体系。

- **情感表达与个性化**：大模型通过分析海量的人类交流数据，学习到了如何根据不同的对话情境和角色特性，调整语言的语调、情绪色彩和表达方式。例如，在模拟一位乐观的教练角色时，它会使用鼓励性和正面肯定的语言；若扮演一位幽默的朋友，则会在交流中穿插轻松诙谐的言语。此外，模型还能模拟特定角色的口头禅、习惯用语等个性化特征，进一步提升角色的真实

感和独特性。

- **故事创作与角色演绎**：在智能技术的驱动下，大模型不仅能够捕捉创意的火花，将零散的情节线索编织成引人入胜的故事画卷，还能赋予角色鲜活的生命力。通过深度学习人类情感与社交互动的复杂性，大模型能够细腻地刻画角色性格，让他们的对话、决策和情感变化如同真实人物般自然流淌。在虚拟的叙事空间里，这些角色仿佛跃然纸上，他们的相遇、冲突与和解都被精心编排，展现出错综复杂的人物关系网和深刻的情感层次，带领读者沉浸在一个由 AI 智慧构筑的多彩世界，体验一段段既超越想象又触动心灵的旅程。

值得注意的是，大模型的表现仍然属于“角色扮演”，意味着它们的行为是基于算法对数据的模式识别和模仿，而非拥有独立的自我意识或真正理解文本背后的深层含义。因此，虽然它们能够提供引人入胜的角色互动体验，但在深层次理解和创新性上仍有其局限性。

- **灵活性与多样性**：大模型的灵活性体现在它能够无缝切换于多种角色和情境之间，无论是扮演一位历史上的哲学家，还是现代的科技創新者，甚至科幻故事中的外星生命体，都能迅速适应并提供相应角色相符的回应。多样性不仅表现在内容的宽泛，还体现在对不同用户需求的个性化满足上，从学术探讨到日常闲聊，大模型均能应对自如。

- **学习与模仿能力**：通过深度学习算法，大模型能够从互联网上的文本、对话记录、书籍、电影剧本等多种来源中汲取知识，不仅学习到语言的直接表达，更能理解背后的含义、隐喻和文化内涵。这种学习能力使其模仿人类语言时，能够复制复杂的句法结构、地道的俚语表达以及细腻的情感传达，使交流更加自然流畅。

- **创造性与生成性**：尽管大模型的输出基于数据，但它在生成文本时展现出的创造性不容忽视。通过对现有信息的重组和创新性联结，大模型能够构思出新颖的观点、故事情节或解决方案，有时甚至超越原始数据的范畴，提出未曾有人明确表述过的见解。这种生成性不仅丰富了对话内容，也为人工智能的创新应用提供了无限可能。

角色扮演实例演示

我们下面演示如何通过大模型角色扮演更好的完成任务：

- **question**：我希望您能在旅游行业担任专家。【现在是 7 月份，我有 3 天的假期，想带着女朋友去旅游，我们 3 人从济南坐飞机出发，预算在 1 万左右。我们喜欢海边或者是大自然的景色。】
【请给我推荐 3 个中国国内的旅游目的地，并规划每一天的行程，最后计算我们 2 人来回机票门票等一系列的总支出】

- **answer**：旅游目的地推荐

根据您的偏好和时间，以下是两个推荐的国内旅游目的地，并附上一天的行程规划及费用

估算。

目的地 1：青岛

青岛以其美丽的海岸线和丰富的历史文化闻名，适合一日游。

行程安排：

1. 早晨

- 交通：从济南乘坐早班航班前往青岛流亭国际机场（预计飞行时间约 1 小时）。

- 费用：单程机票约 300 元 / 人（共 600 元）。

2. 上午：

- 景点：八大关风景区，漫步在有“万国建筑博览会”之称的历史街区。

- 费用：免费。

3. 中午：

- 午餐：品尝地道的青岛海鲜，可以选择沿海的餐厅，预计费用约 200 元。

4. 下午：

- 景点：栈桥和小青岛景区，欣赏海景和岛上美景。

- 费用：栈桥免费，小青岛门票 10 元 / 人（共 20 元）。

5. 傍晚：

- 景点：五四广场和奥帆中心，欣赏日落和现代城市景观。

- 费用：免费。

6. 晚上：

- 晚餐：在五四广场附近的餐厅用餐，预计费用约 200 元。

- 交通：乘晚班航班返回济南。

- 费用：单程机票约 300 元 / 人（共 600 元）。

总费用估算：

- 机票：1200 元

- 餐饮：400 元

- 景点门票：20 元

- 总计：1620 元

目的地 2：张家界

张家界以其壮丽的自然风光和独特的地貌闻名。

行程安排：

1. 早晨：

- 交通：从济南乘坐早班航班前往张家界荷花国际机场（预计飞行时间约 2 小时）。

- 费用：单程机票约 600 元 / 人（共 1200 元）。

2. 上午：

- 景点：张家界国家森林公园，乘坐天门山索道欣赏全景。

- 费用：门票 248 元 / 人（共 496 元）。

3. 中午：

- 午餐：在景区内的餐厅用餐，预计费用约 150 元。

4. 下午：

- 景点：游览天门洞和玻璃栈道，体验惊险刺激的美景。

- 费用：包含在公园门票内。

5. 傍晚：

- 交通：乘晚班航班返回济南。

- 费用：单程机票约 600 元 / 人（共 1200 元）。

6. 晚上：

- 晚餐：在济南机场或返回市区后用餐，预计费用约 150 元。

总费用估算：

- 机票：2400 元

- 餐饮：300 元

- 景点门票：496 元

- 总计：3196 元

总结：

两个推荐目的地都适合一日游，并且费用均在您的预算范围内。青岛以其悠闲的海滨景色和城市文化景观为特色，而张家界则提供了更多自然景观和冒险体验。根据您的具体喜好和对自然或城市景观的偏好，选择最适合您和女朋友的目的地吧！

2.4.1 提出建议

“提出建议”这一模块通常指的是在对话、文本生成、问题解答等场景下，大模型根据用户的需求或问题，能够提供深思熟虑的建议或解决方案。以下是大模型在【提出建议】方面的内

容和特点概述：

- **广泛的知识覆盖**：大模型具备广泛的知识覆盖范围和深度，能够从历史、文化、科学到技术等多个维度提供见解。这不仅意味着它能回答具体的专业问题，如解释量子力学的基本原理，还能在跨学科领域提供独到的视角，如分析人工智能对社会伦理的影响。这种能力让大模型成为了一个无所不知的百科全书，无论面对何种知识需求，都能给出精准而深刻的解答。

- **情境理解与个性化建议**：大模型通过深度学习用户的行为模式和偏好，能够提供高度个性化的建议。比如，如果你正在策划一场户外探险活动，大模型不仅能提供基础的安全指南，还能基于你的兴趣爱好和体能状况，推荐最适合你的路线和装备。更重要的是，它能在你情绪低落时，识别出你的情感状态，并给出恰当的心理慰藉，成为你心灵的知音。

- **创新思维与未来预测**：大模型具有强大的创新思维能力，它能从海量数据中挖掘出潜在的趋势和模式，预测未来的走向。无论是科技趋势、市场变化还是艺术潮流，大模型都能给出前瞻性的洞察。比如，在商业决策中，它能预测新兴市场的潜力，帮助企业制定战略；在艺术创作上，它能启发新的表现手法，推动艺术创新。

- **实践操作与解决方案**：大模型不仅擅长理论分析，还具备解决实际问题的能力。它能根据具体情况，设计出切实可行的解决方案。比如，在城市规划中，大模型可以分析交通流量、人口分布等因素，提出优化方案；在医疗领域，它能分析病历数据，辅助医生制定个性化治疗计划。这种实践导向的能力，使大模型成为各行各业解决复杂问题的得力助手。

提出建议实例演示

我们下面演示如何通过大模型如何根据用户问题提出建议：

- **question**：我是做医学方面的研究工作，请问我的专业怎么样可以跟人工智能比较好的结合，有哪些方向的应用可以做？

- **answer**：将医学研究与人工智能结合可以带来许多创新和高效的应用。以下是一些主要方向及其应用

方向 1：医学影像分析

疾病检测与诊断：利用 AI 技术分析医学影像（如 X 光、CT、MRI 等），提高疾病的早期检测和诊断准确率。

肿瘤识别：自动识别和分类肿瘤，帮助放射科医生更快速地做出诊断。

图像分割：对组织、器官或病灶区域进行精确分割，辅助医生进行治疗计划。

方向 2：个性化医疗

精准医疗：根据患者的基因组数据和病史，使用 AI 算法定制个性化治疗方案。

药物发现：利用 AI 分析生物数据，加速新药研发过程，提高成功率。

病人管理：根据患者的数据进行风险评估和预测，制定个性化的健康管理计划。

方向 3：自然语言处理（NLP）

电子病历分析：利用 NLP 技术处理和分析大量电子病历，提高信息提取效率。

医学文献挖掘：从海量医学文献中提取有用信息，辅助科研工作。

临床决策支持：根据患者的病史和症状，提供智能化的临床决策支持。

方向 4：预测和预防医学

疾病预测：利用机器学习模型分析患者数据，预测疾病发生的风险，提前采取预防措施。

流行病监测：分析公共健康数据，监测和预测流行病的爆发和传播。

健康监测：通过可穿戴设备收集健康数据，利用 AI 实时监测和预警健康问题。

方向 5：机器人辅助手术

手术规划：利用 AI 分析病灶部位，帮助医生制定更精准的手术计划。

手术导航：在手术过程中提供实时的图像和数据支持，提高手术的准确性和安全性。

术后恢复：利用 AI 监测患者术后的恢复情况，提供个性化的康复建议。

方向 6：临床试验优化

患者招募：利用 AI 筛选适合临床试验的患者，提高招募效率。

数据分析：分析临床试验数据，发现潜在问题和趋势，提高试验的可靠性和效率。

试验设计：利用 AI 优化试验设计，减少试验周期和成本。

方向 7：患者管理和护理

远程医疗：利用 AI 进行远程诊断和咨询，提高医疗资源的利用效率。

健康助手：开发智能健康助手，为患者提供日常健康管理和咨询服务。

心理健康：利用 AI 分析心理健康数据，提供个性化的心理健康支持和干预。

将人工智能应用于医学研究和临床实践，不仅可以提高效率和精确度，还可以为患者提供更优质的医疗服务。这些方向都具有广阔的发展前景，可以根据您的具体研究领域选择适合的方向进行探索。

2.4.2 生成和显示数学工具

生成和显示数学工具的功能通常指的是在对话系统中使用大型语言模型来理解和生成数学表达式、公式以及相关的数学内容，并能在用户界面中清晰地显示这些数学内容的能力。这一

功能在教育、科研、工程和技术交流等场景中尤其有用。以下是该功能的主要特点：

- **理解与生成数学表达式**：大模型能够理解并生成复杂的数学公式，包括代数表达式、微积分公式、线性代数方程、概率论公式、统计模型等。这涉及到对数学符号、运算符、希腊字母、上下标、积分、极限、矩阵等元素的理解和生成能力。

- **支持 LaTeX 格式**：为了准确地显示数学内容，大模型通常会使用 LaTeX 这种广泛接受的数学排版语言。LaTeX 提供了丰富的命令和语法，能够精确控制公式的布局和样式，确保数学内容的专业性和可读性。

- **实时渲染与交互**：在用户界面上，数学工具能够实时渲染 LaTeX 代码为可视化的数学公式，使用户能够直观地看到公式的外观。此外，还可能支持用户与数学内容的交互，比如编辑、修改、保存或导出公式。

显示数学公式实例演示

我们下面演示通过大模型生成数学公式：

- **question**：请用 latex 格式输出傅里叶公式

- **answer**：

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos(nx) + b_n \sin(nx))$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(nx) dx, \quad n = 0, 1, 2, \dots$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(nx) dx, \quad n = 1, 2, 3, \dots$$

2.4.3 对文章中数据进行整理

利用大模型对文章中的数据进行整理通常涉及几个关键步骤：数据抽取、结构化处理、分析和总结。大模型，特别是预训练的深度学习模型，能够理解和处理大量的非结构化文本数据，从中抽取有用的信息，并将其转换为结构化的数据格式。

数据抽取

大模型可以通过自然语言处理（NLP）技术，如命名实体识别（NER）、关系抽取（RE）和事件检测（ED），从文本中抽取关键数据点。

结构化处理

抽取的数据被转换成结构化的格式，如表格或 JSON 对象，便于进一步处理和分析。

数据分析

使用统计方法或机器学习算法对结构化数据进行分析，提取模式、趋势和异常值。

总结与报告生成

根据分析结果，大模型可以生成总结性报告，提供洞察和建议。

实例演示

我们下面演示利用大模型对用户输入进行数据整理：

• **question**：【在最近的一次业务报告演示中，Zana 公司的 CEO 强调了他们在过去财年中显著的增长。她分享了公司营收增长了 0.15，达到了 5000 万美元，利润率为 0.12(净利润为 600 万美元)。报告还展示了客户基础增长了 0.2，现在总共有 10 万名客户。此外，公司的运营费用增加了 0.1，达到了 1000 万美元，员工数量增加了 0.25 导致当前的员工人数为 500 人。】让我们逐步思考，生成包含所有信息的表格。

• **answer**：好的，我们来生成包含所有信息的表格，并将其转换为 LaTeX 格式。

首先，我们列出报告中提到的所有信息：

公司营收增长了 0.15，达到了 5000 万美元。

利润率为 0.12，净利润为 600 万美元。

客户基础增长了 0.2，现在总共有 10 万名客户。

运营费用增加了 0.11，达到了 1000 万美元。

员工数量增加了 0.25，现在总共有 500 人。

我们将这些信息放入一个表格中：

指标	数值	增长率
公司营收	5000 万美元	0.15 (15%)
利润率	12% (净利润 600 万美元)	-
客户基础	10 万名客户	0.2 (20%)
运营费用	1000 万美元	0.1 (10%)
员工数量	500 人	0.25 (25%)

表 2.1: Zana 公司财年报告摘要

2.4.4 利用大模型进行用户评论分类

我们下面演示如何利用大模型进行用户评论分类：

- **question1**：你将扮演一个情感分类机器人，帮我判断用户的评论是“好评”，“无关评论”，“差评”。接下来我会把用户评论告诉你。你只能回复“好评”，“无关评论”，“差评”。

Q：我今天早上吃了一碗面。

A：

- **answer1**：无关评论

- **question2**：你将扮演一个情感分类机器人，帮我判断用户的评论是“好评”，“无关评论”，“差评”。接下来我会把用户评论告诉你。你只能回复“好评”，“无关评论”，“差评”。

Q：我今天早上吃了一碗面，非常美味。

A：

- **answer2**：好评

2.4.5 利用大模型写报告

我们下面演示如何利用大模型写报告：

- **question**：请写一份数据开发员的述职报告，大约 2000 字左右

- **answer**：数据开发员述职报告尊敬的领导、同事们：

大家好！

时间如梭，转眼间在公司的工作又将告一段落。在这段时间里，我作为一名数据开发员，努力工作，不断学习，积极配合团队，完成了多项任务。现对这一阶段的工作进行总结，并对未来工作进行规划。

一、工作回顾项目开发在这一阶段中，我参与了多个数据开发项目，包括数据清洗、数据处理、数据分析等工作。在项目中，我主要负责数据的提取、转换和加载（ETL）流程的开发和优化工作，确保数据处理的高效性和准确性。

数据清洗：通过编写脚本，清洗原始数据，去除无效和重复数据，确保数据质量。

数据处理：使用 SQL 和 Python 等工具对数据进行处理和分析，生成各类数据报表，为决策提供支持。

数据加载：将处理后的数据加载到数据仓库中，确保数据的及时更新和可用性。技术提升为了更好地完成工作任务，我不断学习新的技术和工具，提高自己的技术水平。在这一阶段中，

我重点学习了大数据处理技术，如 Hadoop、Spark 等，并将其应用到实际工作中，提高了数据处理效率。

团队协作在团队协作方面，我积极与团队成员沟通交流，分享经验和心得，共同解决问题。通过团队的努力，我们顺利完成了多个项目，得到了领导和同事们的认可。

问题解决在工作中难免会遇到一些问题和挑战。在面对这些问题时，我保持冷静，积极寻找解决方案。例如，在数据处理过程中遇到性能瓶颈时，通过优化 SQL 查询和改进数据处理流程，有效解决了问题，提升了系统的性能。

二、工作业绩项目成果在这一阶段中，我参与了多个重要项目的开发和实施，按时完成了项目任务，达到了预期效果。例如，在某大型数据分析项目中，我负责的数据处理部分顺利完成，为项目的成功实施奠定了基础。

技术突破通过不断学习和实践，我在大数据处理技术方面取得了一些突破。在实际项目中，成功应用了 Hadoop 和 Spark 等技术，大幅提高了数据处理的效率和性能。

团队贡献作为团队的一员，我积极参与团队建设，与同事们密切合作，共同完成了多个项目。通过团队的努力，我们取得了良好的成绩，得到了领导的肯定。

三、工作体会不断学习在数据开发领域，技术更新换代非常快，只有不断学习，才能跟上时代的步伐。在这一阶段中，我深刻体会到学习的重要性，通过自学和参加培训，不断提升自己的技术水平。

团队合作团队合作是项目成功的关键。在工作中，我深刻体会到团队合作的重要性。通过与团队成员的密切合作，我们能够更好地解决问题，完成任务。

细致严谨数据开发工作需要非常细致和严谨，任何一个小的错误都可能导致严重的后果。在工作中，我始终保持细致严谨的态度，认真对待每一个环节，确保数据的准确性和可靠性。

四、未来计划提升技术在未来的工作中，我将继续学习和掌握新的技术，不断提升自己的技术水平。重点学习和掌握大数据处理和人工智能技术，将其应用到实际工作中，提高数据处理的效率和智能化水平。

优化流程在工作中，我将继续优化数据处理流程，提高数据处理的效率和准确性。通过引入自动化工具和技术，减少手工操作，降低出错率，提高工作效率。

加强合作继续加强与团队成员的合作，共同解决问题，完成任务。通过团队的力量，我们能够更好地应对工作中的挑战，取得更好的成绩。

创新思维在工作中，我将继续保持创新思维，积极探索新的方法和技术，提升工作的创新性和创造力。通过不断创新，我们能够为公司创造更多的价值。

五、结语总的来说，在这一阶段的工作中，我取得了一些成绩，但也存在一些不足。在未来的工作中，我将继续努力，不断提升自己的技术水平和工作能力，为公司的发展贡献自己的力量。

感谢领导和同事们的支持和帮助，我将在未来的工作中，继续保持积极进取的态度，不断学习和进步，为公司的发展做出更大的贡献。

谢谢大家！

这份述职报告详细回顾了工作情况，突出了工作成绩，并提出了未来的工作计划，希望你有所帮助。如果需要进一步调整或补充，请随时告知。

2.4.6 利用大模型生成测试题

我们下面演示如何利用大模型生成测试题：

• **question**：市场营销课堂笔记讨论了市场营销的 4P 理论：产品、价格、渠道、推广

- B2C 和 B2B 的区别：B2C 是直接面向消费者市场，B2B 则是企业间的市场

- 不同类型的市场研究：定量和定性研究，如调查和焦点小组讨论

- 定价策略：撇除、渗透、竞争、价值等

- 分割市场：按照年龄、收入等因素对客户进行细分

- 4 种市场细分：人口统计、地理、心理学、行为学

- 讨论了不同的促销工具：广告、个人销售、公共关系、直销和销售促销

- 基于我的课堂笔记请生成 5 道市场营销的测验题目

• **answer**：根据你的课堂笔记，这里是五道市场营销的测验题目：

1、4P 理论：

问题：请解释市场营销的 4P 理论，并举例说明每个组成部分。

答案：4P 理论包括产品、价格、渠道、推广。

产品：指公司出售的商品或服务。例如，苹果的 iPhone。

价格：指商品或服务的定价策略。例如，撇除定价策略。

渠道：指产品如何到达消费者手中。例如，通过零售商店销售。

推广：指公司如何宣传其产品。例如，通过电视广告进行推广。

2、B2C 与 B2B 的区别：

问题：请描述 B2C 和 B2B 的区别，并给出一个例子来说明每种类型的业务。

答案：B2C（Business to Consumer）是指企业直接面向消费者市场。例如，亚马逊的在线零售业务。B2B（Business to Business）是指企业间的市场。例如，英特尔向戴尔公司销售处理器。

3、市场研究类型：

问题：定量研究和定性研究的主要区别是什么？请分别举例说明。

答案：定量研究通常涉及数值数据和统计分析，例如，通过问卷调查收集的数据。定性研究涉及非数值数据和解释性分析，例如，通过焦点小组讨论收集的观点和感受。

4、定价策略：

问题：请简要解释以下定价策略：撇除定价、渗透定价、竞争定价和价值定价。

答案：

撇除定价：初期价格设定较高，之后逐步降低。例如，电子产品新发布时价格较高。

渗透定价：初期价格设定较低，以迅速占领市场份额。例如，Netflix 初期的低订阅价格。

竞争定价：根据竞争对手的价格进行定价。例如，超市根据其他超市的价格调整商品价格。

价值定价：根据产品的感知价值进行定价。例如，高端品牌服装。

5、市场细分：

问题：市场细分可以按哪些因素进行？请详细说明人口统计、地理、心理学和行为学细分。

答案：

人口统计细分：根据年龄、性别、收入等人口特征。例如，化妆品公司按性别和年龄细分市场。

地理细分：根据地理位置进行细分。例如，按国家、城市划分市场。

心理学细分：根据消费者的生活方式、个性、价值观进行细分。例如，户外品牌按冒险精神进行细分。

行为学细分：根据消费者的购买行为、使用习惯进行细分。例如，按购买频率细分市场。