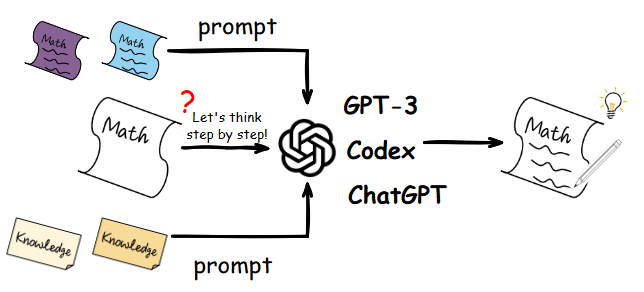
基于提示学习的大型语言模型推理综述

# 1. 引言

推理能力是人类智能的核心之一。随着预训练技术的不断发展，借助提示学习（例如Chain-of-Thought Prompting[1]），大型语言模型展现出了令人惊讶的推理能力，引起了学术界和工业界学者的广泛关注。本文介绍一篇发表于ACL2023的关于"语言模型提示推理"的综述，从提示学习的角度系统地划分、梳理和对比了各种前沿推理工作

本文对「语言模型提示推理」的最新进展进行了梳理，包括预备知识、提示推理方法的分类、深入的比较和讨论、开放的资源和基准、以及未来的潜在方向。



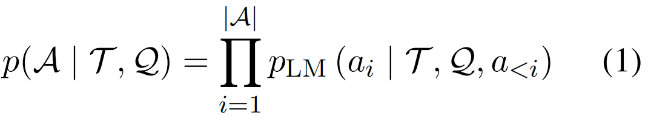
论文链接：https://arxiv.org/abs/2212.09597

资源列表：

https://github.com/zjunlp/Prompt4ReasoningPapers

# 2. 预备知识

对于标准的提示（Prompt）学习，给定推理问题Q、提示T和参数化的概率模型PLM，推理任务的目标是最大化答案A的概率，即：



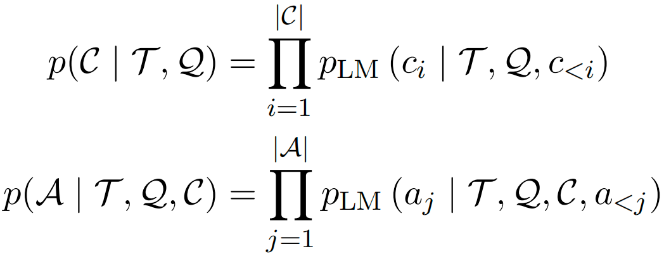
其中ai表示答案A的第i个token，|A|表示答案A的长度。对于少样本提示，T由(Q,A)对形式的K个样例组成。

为了提高预训练模型的提示推理能力，近期的工作有两个主要的研究分支。第一个分支是**增强提示中的推理策略**，包括提示工程、推理过程优化和外部推理引擎。

对于提示工程，许多方法尝试直接提高提示T的质量，这些方法可以称为**单阶段方**法。其他一些方法在每个推理阶段，将ci作为上下文（Context）附加到(T,Q)中，或者为每个ci设计特定的提示Tci，这些方法可以称为**多阶段方法**。例如，可以将一个复杂的问题分解成若干更简单的子问题，逐个推理推理步C加入提示中构成  ，因此公式（1）可以变换为：



其中p(C|T,Q)和p(A|T,Q,C)  定义为:

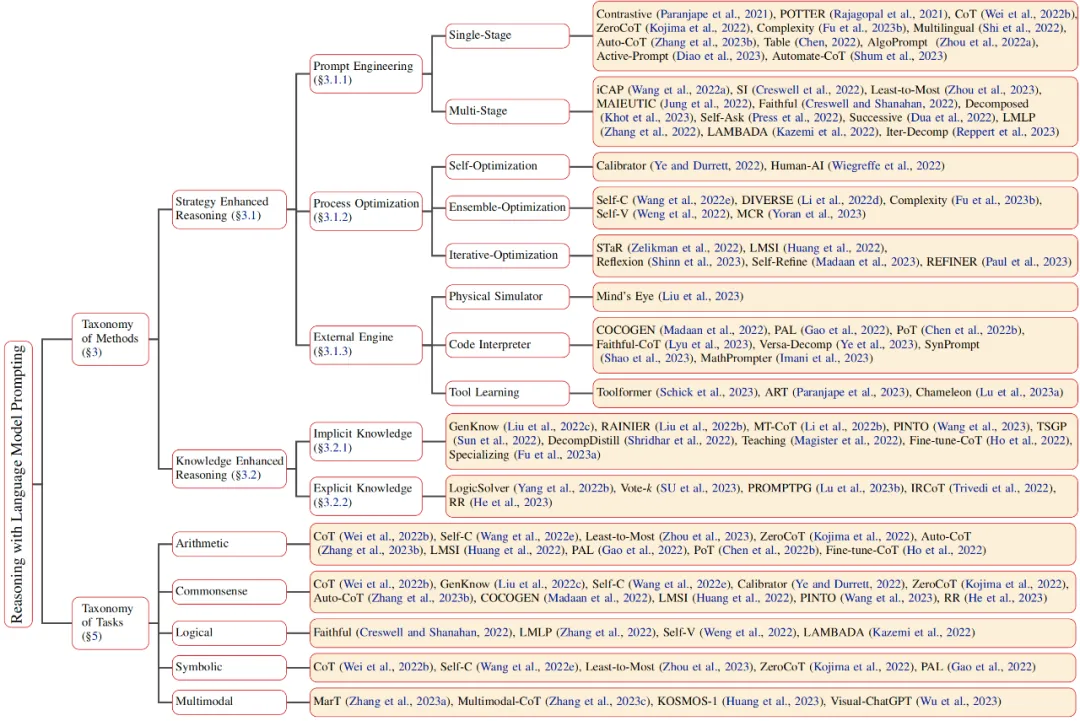


ci是其中一个推理步，总计|C|个推理步。

对于推理过程优化，最简单的方法是引入一个参数化的优化器在生成答案A时校准推理步C，这类工作可以称为**自优化方法**。而**集成优化方法**尝试从多个推理过程中联合得到最终结果。除此之外，整体的优化过程还能以迭代的方式与语言模型微调（生成目标三元组(Q,C,A)）相结合，这类方法可以称为**迭代优化方法**。另外还有一些工作利用**外部引擎生**成提示T，直接执行推理步C，或在C中植入外部工具来进行推理。

另一个研究分支是**增强提示中的知识**。大模型中富含的**隐式知识**可以帮助模型生成知识或推理依据作为知识提示。同时，外部资源中的**显式知识**也可以被利用并通过检索作为知识提示来增强推理。

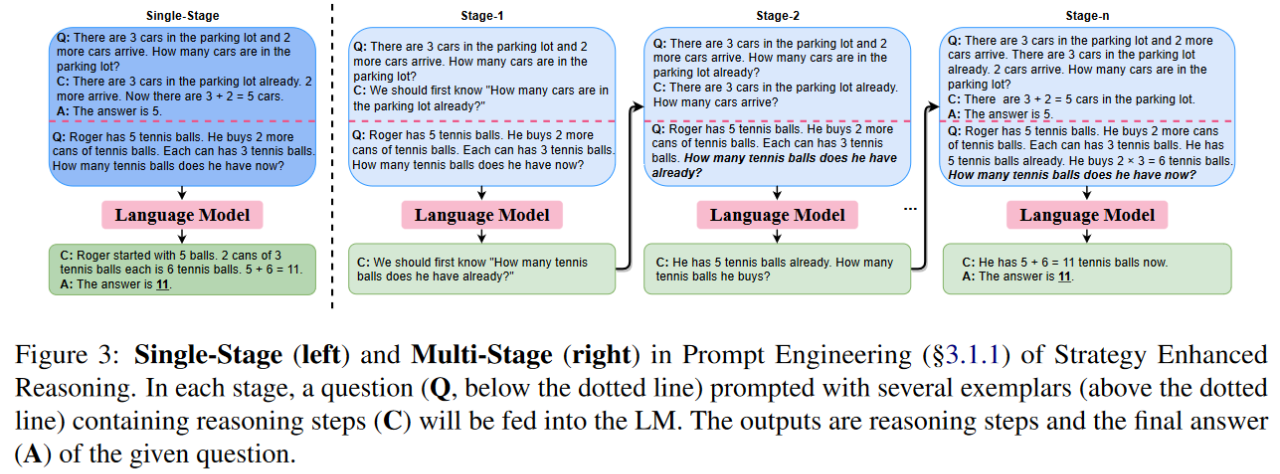
# 3. 方法分类



3.1 策略增强的推理

策略增强系列的工作主要目的是设计更好的推理策略来增强大模型的推理表现，具体分为提示工程、推理过程优化和外部推理引擎。

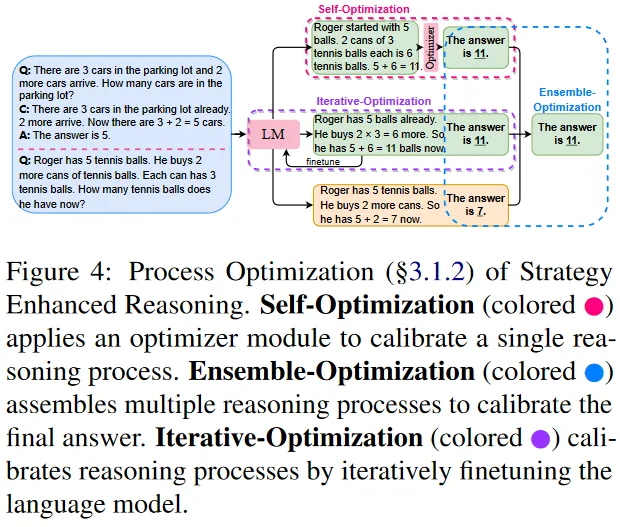
3.1.1 提示工程



**单阶段方法：**在单阶段方法中，预训练模型结合各种提示一次性生成问题推理的结果。早期的工作[4]通常使用基于模板的提示进行推理。然而，考虑到大规模语言模型具有强大的上下文学习（In-context Learning）能力，思维链提示方法[1]引入了称为思维链的中间推理步骤到少样本提示的示例中，以促使预训练模型更好地完成推理。除了少样本推理，预训练模型还能进行零样本推理，只需在问题后附加“Let's think step by step”[5]作为提示，即可引导预训练模型生成推理步骤并完成推理问答任务。

**多阶段方法：**在推理过程中，人类通常难以一次性想出完整的推理路径。一种更直观的解决方法是将复杂问题分解为更简单的子问题，并逐个进行推理解决。类似地，多阶段方法旨在将之前的单阶段提示转变为多阶段提示。Maieutic prompt方法[6]将每个阶段的输出视为独立的新问题，Least-to-most prompt[7]和Iteratively prompt[8]方法则将每个阶段的输出添加到上下文中，以提示预训练模型。而Decomposed prompt[9]方法将任务分解为多个独立的子任务，并为解决每个子任务设计了特定的提示。

3.1.2 推理过程优化

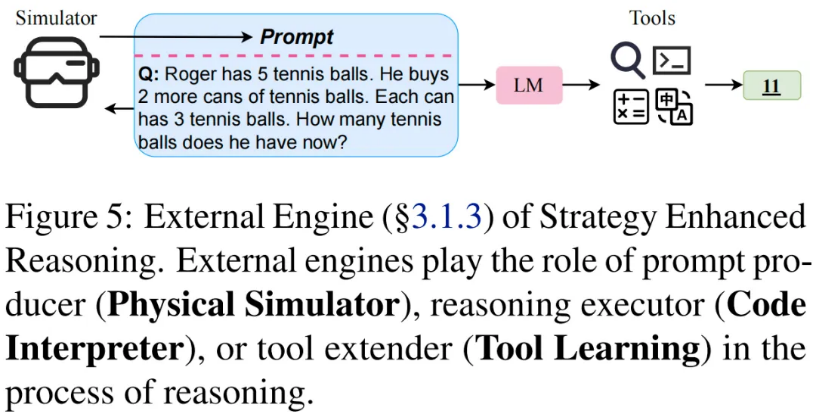


**自优化：**自优化方法通过引入额外的模块来纠正推理过程。Calibrator方法[10]利用一个校准器来调整预测概率，该校准器的分数反映了推理依据的真实性。在生成文本推理依据时，Human-AI方法[11]通过微调一个Seq2seq模型作为过滤器，以预测生成的推理依据是否可接受。

**集成优化：**为了克服单一推理路径的限制，集成优化方法通过集成校准在多个推理路径之间进行操作。Self-consistency方法[12]引入了常见的自然语言生成采样策略，以获得多个推理路径，并通过多数投票产生最一致的答案。然而，考虑到某个推理路径可能会得出错误答案，而不是所有的推理步骤都导致最终的错误，DIVERSE方法[13]提出了一个步骤感知的投票检验器来对每个推理路径进行评分。当错误的推理路径数量较多而正确的推理路径数量较少时，步骤感知的投票检验器可以缓解简单多数投票的限制。

**迭代优化：**迭代优化方法利用预训练模型进行迭代微调，以校准推理过程。具体而言，基于迭代优化的方法可以反复提示预训练模型生成推理路径，并使用包含生成的推理过程的实例来微调自身。STaR方法[14]从一组较小的样本开始，促使预训练模型生成推理步骤并自行回答问题，正确答案和推理步骤将直接添加到用于微调的数据集中。

3.1.3 外部推理引擎



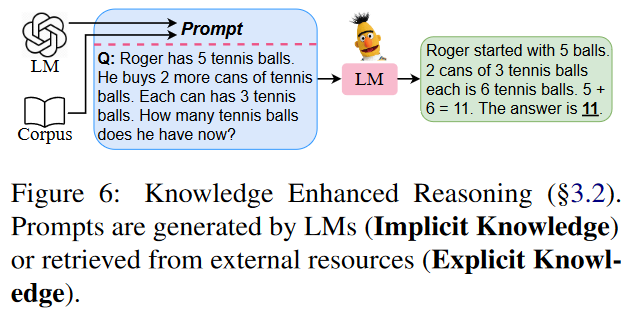
在使用预训练模型提示进行推理时，模型应同时具有语义理解和复杂推理（例如通过生成推理过程）的能力，但往往鱼和熊掌不可兼得。为了解决这种障碍，预训练语言模型还可以借助外部引擎进行推理。

**物理模拟器：**给定一个物理推理问题，Mind’s eye 方法[15]利用一个物理计算引擎来模拟物理过程。模拟结果可以作为帮助预训练模型推理的提示，弥补了预训练模型物理知识的不足。

**代码解释器：**程序和代码在鲁棒性和可解释性方面具有天然的优势，并且能更好地描述复杂的结构和推导复杂的计算。PAL方法[16]通过将求解步骤从预训练模型中分离出来，交由程序来执行。在该方法中，预训练模型主要负责学习任务，而推理过程则由自然语言和编程语言混合组成，在少样本提示和预训练模型输出中，自然语言用作注释来帮助生成程序。这种混合的方式使得推理过程更加清晰和可解释。

**工具学习：**尽管现在的大模型拥有强大的生成和决策能力，它们仍然不擅长一些基础的能力，比如数值计算、知识更新等等，而这些恰恰是一些小而简单的工具所擅长的。基于此，Toolformer[17]将计算器、搜索引擎、翻译器等工具融入模型的训练过程中，通过将工具调用的结构植入到文本的生成过程中，模型的能力得到了极大的扩展。Chameleon[18]利用了大模型强大的决策能力，使他们能够结合各种外部工具来处理组合推理任务。

3.2 知识增强的推理



3.2.1 隐式知识

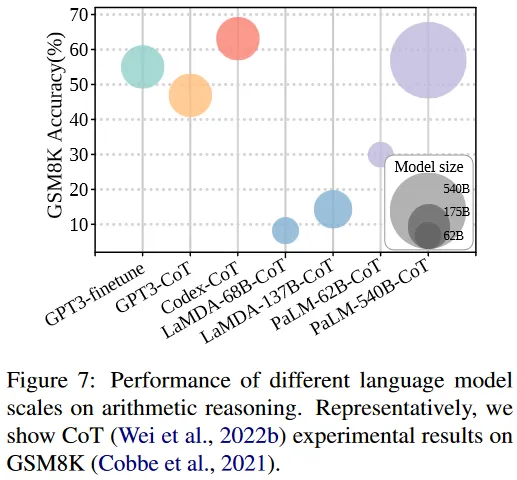
许多研究表明，预训练模型中蕴含了相当数量的隐式知识，这些知识可以通过条件生成来引出，作为知识提示来增强推理能力。Generated Knowledge Prompting方法[19]利用GPT-3生成少样本提示中的知识，并用这些提示来引导下游的预训练模型。在此基础上，Rainier方法[20]使用增强学习进一步校准生成的知识。与之不同的是，TSGP方法[21]提出了一个两阶段的生成提示方法，其中包括答案生成的提示。此外，还有一些工作[22]探索了使用大规模教师模型上的思维链输出来微调小规模学生模型，以实现推理能力的迁移，这是基于知识蒸馏的思想。

3.2.2 显式知识

尽管预训练模型展现出强大的生成能力，但它们仍然存在产生幻觉事实和不一致知识的倾向。最近的研究表明，在外部语料库中检索用于上下文学习的提示，可以向模型注入显式知识，从而取得良好的性能。PROMPTPG方法[23]提出了一种基于梯度策略的动态提示检索方法，无需进行暴力搜索。Vote-k方法[24]提出了一种选择性注释框架，以避免需要大量标记的检索语料库。该方法还开发了一种基于图的技术，通过从大型未标记语料库中构建多样化且具有代表性的小型标记数据库，从而大大降低了注释和检索的成本。然后，可以从这个小型数据库中检索带有上下文标注的示例。这些方法的引入使得模型能够更可靠地获取外部知识，并避免潜在的错误和不一致。

# 4. 比较和讨论

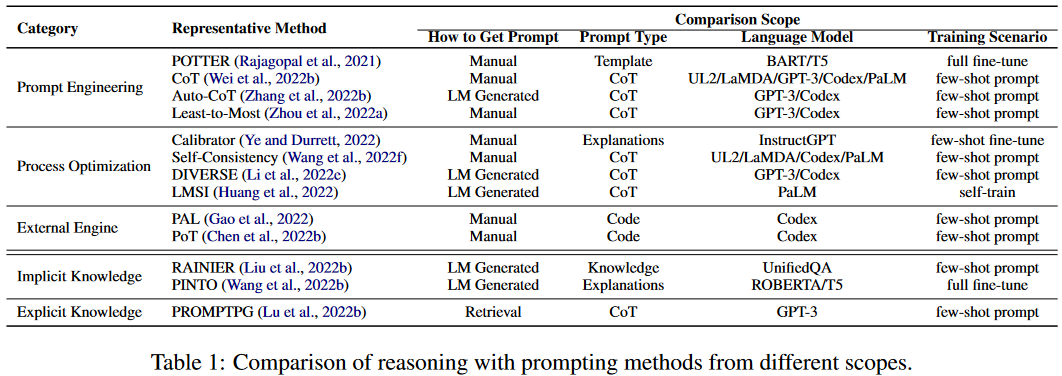
4.1 预训练模型比较



先前工作已经系统地证明，随着模型规模的增加，少样本提示几乎在各项任务中都有更好的表现，这可以解释为**更大规模的模型蕴含更多用于推理的隐式知识**。思维链作为提示能够进一步提高性能，在 540B PaLM 模型上表现出最大增益，但当模型规模小于 100B 时，思维链并不能产生性能增益，甚至可能会导致下降。因此，**思维链可能引发了模型规模上的“涌现”能力，即较小规模模型中不存在而仅存在于大规模模型中的能力。**

另一项有趣的发现是，在相同参数规模下 Codex 的性能显著优于 GPT-3，尽管它们之间的主要区别在于训练语料库（Codex 是代码语料上训练的 GPT-3 变体）。这种现象已经在许多工作中验证，表明**模型在代码语料上进行预训练不仅可以实现代码生成、理解的能力，还可能诱发了思维链推理能力。**

4.2 提示比较



现有方法的提示的构建方法主要有以下三种：

**1. 人工构建提示**适用于模板提示和不太复杂的小样本提示。

**2. 预训练模型生成提示**弥补了人工构建提示费时费力且表现不稳定的缺点。它可以为每个问题定制特定的推理依据，并提供足够的知识和提示进行微调或自我训练。

**3. 基于检索的提示**通常依赖于注释良好的外部资源（例如维基百科）并消耗昂贵的信息检索，但它可以缓解生成不稳定的问题。

无论是哪种方法生成的提示，实验结果表明思维链推理能力仅表现在少样本提示下的大模型。这些结果表明，**输入上下文中显式包含的高质量推理依据是大模型提示推理的关键（近期工作[25]表明思维链和问题的相关性及推理过程更加重要）。**尽管一些工作尝试探索大模型的上下文学习能力，对NLP社区而言，思维链提示能够在大模型取得成功仍然是未解之谜。一种可能的原因是思维链是代码预训练的副产品，并可以通过提示解锁。在少样本提示中包含思维链的样例可以被视为一种激发隐藏在大模型中的推理能力的指令。

# 5. 基准和任务分类体系

**数学推理：**数学推理（或算术推理）能力是在数学单词问题上进行推理的能力。数学推理技能是人类智能的重要能力，也是通用人工智能系统必不可少的能力。早期的工作专注于需要单步或多步推理的小学水平数学问题，且数据量相对较小。后来的工作增加了数据集的复杂性和规模，并引入了更难的问题。

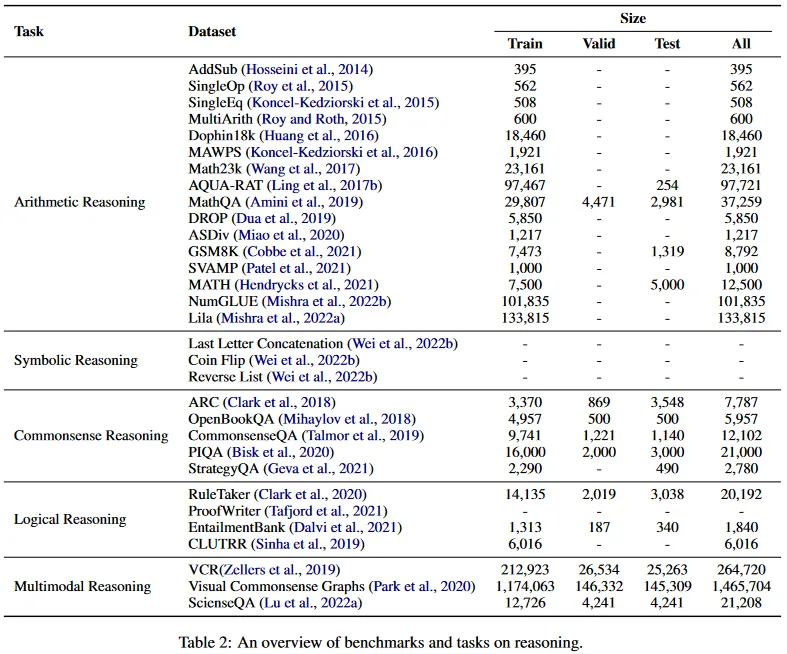
**常识推理：**常识知识和常识推理是机器智能的核心问题。在回答问题时，人类通常会利用上他们丰富的世界知识。而对于预训练模型，执行常识推理的主要挑战在于如何在一般背景知识的假设下涉及物理和人类交互。

**逻辑推理：**逻辑推理的常见形式包括演绎推理和归纳推理。演绎推理是通过从一般信息到特定结论来进行的，该任务的典型数据集由合成规则库和派生结论组成。与演绎推理相反，归纳推理旨在通过从特定到一般来得出结论。

**符号推理：**这里的符号推理仅表示测试一组简单符号操作函数的任务，而不是符号 AI，符号 AI 是由规则引擎或专家系统或知识图谱实现的更通用的概念。这类任务的构建通常由人工明确定义，因此很容易将测试集划分为域内测试集和域外（OOD）测试集。

**多模态推理：**大多数现有的推理基准仅局限于单一文本模态并且域多样性有限。而人类在进行推理时会利用来自不同模态的可用信息。为此，多模态推理基准被提出以缩小这一差距。

各类推理任务的常用数据集如下：



# 6. 未来方向

**预训练模型推理理论。**大模型已经被证明具有“涌现”的零样本学习和推理等能力。为了探究这样的成功中的原因，许多研究人员从经验上探讨了上下文学习（如[26]探讨了In-context Learning可以近似为一种前向梯度下降）和推理依据的作用，但仍需要对大模型提示学习推理的潜在理论原理有更深入的理解。

**高效推理。**现有的方法主要依赖于大模型，这会消耗大量的计算资源。考虑到实用性，研究开发高效大模型训练[27]，比如模型编辑[28]、高效参数微调[29]等，或通过大模型赋能小模型推理是有必要的，在模型训练和推理过程中有利于降低碳排放实现绿色AI。

**鲁棒、可信、可解释推理。**大多数深度学习模型缺乏鲁棒性和可解释性，尤其是在推理等需要强逻辑的任务中。近期有工作[30]发现大模型提示学习推理存在很强的偏见和毒性，因而研究鲁棒可信可解释的推理具有非常重要的意义。

**多模态（交互式）推理。**文本推理仅局限于可以通过自然语言表达的内容。考虑到人类在现实世界中推理时信息的多样性，一个更有前途的方向是多模态推理。此外，多模态（交互式）推理方法也可以受其他领域（例如认知科学等）的启发。随着语言模型能力的增强，大模型与环境、人类(RLHF)、工具(Tool Learning)和其他模型(Multi-Agent)交互工作成为趋势，这一趋势与大模型具身智能连接紧密，是未来NLP社区重要的研究方向。

**泛化（真正的）推理。**泛化是模型获得真正推理能力的最重要标志之一。给定一个推理任务，我们希望 预训练模型不仅可以处理问题本身，还可以解决一组类似的推理任务（在训练阶段未见过）。现阶段，大模型基于提示学习涌现出一定的分布外泛化能力[31]，这种能力是否可以适用于推理并实现新领域、新任务的自适应仍值得探索。

[1] Chain of Thought Prompting Elicits Reasoning in Large Language Models NeurIPS 2022

[2] Towards Reasoning in Large Language Models: A Survey 2022

[3] A Survey of Deep Learning for Mathematical Reasoning 2022

[4] Template Filling for Controllable Commonsense Reasoning 2021

[5] Large Language Models are Zero-Shot Reasoners NeurIPS 2022

[6] Maieutic Prompting: Logically Consistent Reasoning with Recursive Explanations EMNLP2022

[7] Least-to-Most Prompting Enables Complex Reasoning in Large Language Models 2022

[8] Iteratively Prompt Pre-trained Language Models for Chain of Thought EMNLP2022

[9] Decomposed Prompting: A Modular Approach for Solving Complex Tasks 2022

[10] The Unreliability of Explanations in Few-shot Prompting for Textual Reasoning NeurIPS 2022

[11] Reframing Human-AI Collaboration for Generating Free-Text Explanations NAACL2022

[12] Self-Consistency Improves Chain of Thought Reasoning in Language Models 2022

[13] On the Advance of Making Language Models Better Reasoners 2022

[14] STaR: Self-Taught Reasoner Bootstrapping Reasoning With Reasoning 2022

[15] Mind's Eye: Grounded Language Model Reasoning through Simulation 2022

[16] PAL: Program-aided Language Models 2022

[17] Toolformer: Language Models Can Teach Themselves to Use Tools 2022

[18] Chameleon: Plug-and-play Compositional Reasoning with Large Language Models 2023

[19] Generated Knowledge Prompting for Commonsense Reasoning ACL2022

[20] Rainier: Reinforced Knowledge Introspector for Commonsense Question Answering EMNLP2022

[21] TSGP: Two-Stage Generative Prompting for Unsupervised Commonsense Question Answering EMNLP2022 Findings

[22] Explanations from Large Language Models Make Small Reasoners Better 2022

[23] Dynamic Prompt Learning via Policy Gradient for Semi-structured Mathematical Reasoning 2022

[24] Selective Annotation Makes Language Models Better Few-Shot Learners 2022

[25] Towards Understanding Chain-of-Thought Prompting: An Empirical Study of What Matters 2022

[26] Why Can GPT Learn In-Context? Language Models Secretly Perform Gradient Descent as Meta-Optimizers 2022

[27] Black-Box Tuning for Language-Model-as-a-Service ICML2022

[28] Editing Large Language Models: Problems, Methods, and Opportunities 2023

[29] Parameter-efficient fine-tuning of large-scale pre-trained language models Nature Machine Intelligence 2023

[30] On Second Thought, Let’s Not Think Step by Step! Bias and Toxicity in Zero-Shot Reasoning 2022

[31] https://yaofu.notion.site/A-Closer-Look-at-Large-Language-Models-Emergent-Abilities-493876b55df5479d80686f68a1abd72f