

1 高级提示技巧

前面章节介绍了提示设计的基本策略，而掌握更多高级提示技巧对于成为提示工程专家来说是必不可少的。本章主要介绍一些高级提示技巧，通过将这些技术整合到您的工具包中，您将能够制作充分利用 GPT 和 ChatGPT 等 AI 语言模型功能的提示，使其为我们提供更有针对性、复杂和动态的回答，即使在最具复杂的应用场景中，也能提供很出色的结果。

1.1 零样本提示

经过大量语料库的训练和调优，大多数大型语言模型（LLM）已具备执行零样本（zero-shot）任务的能力。例如：

提示：

将如下文字分类成中性、积极的和消极的。

文本：我觉得这次假期还行。

情绪：

输出：

文本：我觉得这次假期还行。

情绪： 中性

请注意，上面的提示中我们没有向模型提供任何提示范例——这就是零射能力（zero-shot capability）在发挥作用。当零射能力无法生效时，建议在提示中提供演示或示例，以进行少样本提示。

谷歌研究团队证实，通过指令微调可以提升零样本学习的表现。指令微调实质上是对模型在描述指令的数据集上进行微调。此外，基于人类反馈的强化学习（RLHF）已经被用于辅助指令微调，使模型更好地适应人类喜好。这种方法推动了像 ChatGPT 这样的模型的发展，我们将在接下来的章节中对此进行详细探讨。

1.2 少样本提示

虽然大型语言模型展示了卓越的零样本能力，但在使用零样本设置时它们仍然无法完成更复杂的任务。少样本提示（Few-shot prompting）可以用作启用上下文

学习的技术，我们在提示中提供演示以引导模型获得更好的性能。这些演示用作后续示例的条件，我们希望模型在这些示例中生成回应。

让我们透过一个例子来试 Brown 等人提出的 **few-shot** 提示。在这个例子中，任务是在句子中正确使用一个新单词。

提示：

“whatpu”是一种生长在坦桑尼亚的小型毛茸茸的动物。使用 whatpu 这个词的一个例子是：

我们在非洲旅行，看到了这些非常可爱的 whatpu。

“farduddle”的意思是快速跳跃。使用 farduddle 这个词的一个例子是：

输出：

我看到那只兔子正在草地上快速地 farduddle。

我们可以观察到，该模型仅通过提供一个示例（即 **1-shot**）就以某种方式学会了如何执行任务。对于更困难的任务，我们可以尝试增加演示（例如，**3-shot**、**5-shot**、**10-shot** 等）。

根据 Sewon Min 等人的研究结果，以下是在进行少量本学习时关于示例的一些额外提示：

- 标签和示例中指定的输入文字的分配都很重要（标签对个别输入是否正确不重要）
- 即使您只使用随机标签，使用的格式也很重要，远胜于不使用任何标签
- 额外的结果显示，从真实标签分布（非均匀分布）中选择随机标签也有一定作用

让我们试几个例子。首先，让我们试一个带有随机标签的例子（意思是将标签“负面”和“正面”随机分配给输入）。

提示：

这太棒了！ //Negative

这很糟糕！ //Positive

哇，那部电影很酷！ //Positive

这是一个多么可怕的节目！//

输出：

Negative

可以看出，即使标签已经被随机化，我们仍然得到了正确的答案。请注意，我们也保持了标签格式。实际上，新版本 GPT 模型在应对随机格式时更加稳健。例如：

提示：

Positive 这太棒了！

这很糟糕！Negative

哇，那部电影真的很棒！

Positive

真是一场糟糕的表演！--

输出：

Negative

上面描述的格式不一致，但模型仍然预测出正确的标签。我们需要进行更全面的分析，以确认这是否适用于其他更复杂的任务。

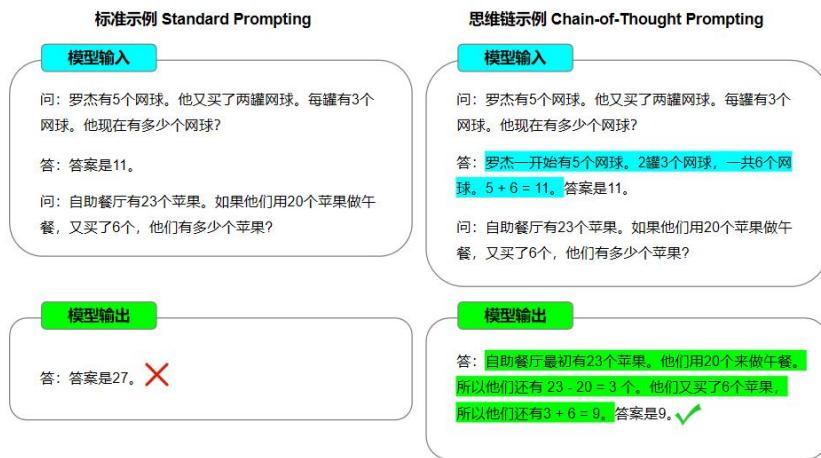
尽管 Few-shot Prompting 表现出色，但在某些情况下，如果任务过于复杂或涉及的领域知识较广泛，GPT 模型可能无法在仅有少量示例的情况下表现良好。例如，处理专业领域的问题或高度技术性的话题时，模型可能无法充分理解并给出精确答案。在这种情况下，需要使用更复杂的模型或者更多的训练数据来进行推理。研究表明，如果我们将问题分析成步骤并向模型演示，可能会有所帮助，这就是 Jason Wei 等人提出的思维链接(CoT)提示，这种提示方法可以解决更复杂的数学、常识和推理任务。

整体而言，提供范例用于解决某些任务是有用的。当零样本提示和少量样本提示还不够时，这可能意味着基于模型所学的内容并不充分，建议开始考虑微调模型或者尝试更深入的提示技术。接下来，我们将讨论思维链提示技术。

1.3 思维链

思维链(Chain-of-thought prompting, CoT)提示过程是一种最近开发的提示方法，它鼓励大语言模型解释其推理过程。在三个大型语言模型上的实验表明，思维链提示在一系列算术、常识和符号推理任务上提高了性能。实证收益可能是显著的。例如，仅用八个思维链示范提示一个 PaLM 540B，就能在数学词问题的 GSM8K 基准上实现最先进的准确性，甚至超过了经过微调的带验证器的 GPT-3。

下图 1 显示了少样本提示（左）与链式思维提示过程（右）的比较。



图片来源: <https://arxiv.org/abs/2201.11903>.

思维链的主要思想是通过向大语言模型展示一些少量的例子，在样例中解释推理过程，大语言模型在回答提示时也会显示推理过程。这种推理的解释往往会导致出更准确的结果。

只需将思维链序列的示例包含到少量提示的范例中，就可以在足够大的现成语言模型中轻松地引发思维链推理。您可以将其与少量样本提示结合使用，要以得到应更好的回答之前进行推理的更复杂任务。

研究发现，思维链不会对小模型的性能产生积极影响，只有在使用大约 100B 参数的模型时才会产生性能提升；且在更复杂的问题上具有更大的性能提升。例如，在 GSM8K(基线性能最低的数据集)上，对于最大的 GPT 和 PaLM 模型，性能提高了一倍以上；通过 GPT-3 175B 和 PaLM 540B 进行 Chain-of-thought prompting 与之前的最先进水平相比具有较大优势，之前的最先进水平通常是在标记的训练数据集上微调特定任务的模型。

但是，该方法并非适用于所有任务，其表现也受模型规模和任务复杂度等因素的影响。通过进一步研究链式思维提示的工作原理，我们可以更好地了解大型语言

模型的推理过程，并优化其性能。

1.4 零样本思维链

尽管在少样本设置下手动增加示例的链式思考成本很小，但这种注释成本对于微调来说可能是不合适的。零样本思维链（Zero Shot Chain of Thought, Zero-shot-CoT）提示过程是对思维链的后续研究，引入了一种非常简单的零样本提示。他们发现，通过在问题的结尾附加“让我们一步步思考。”这几个词，大语言模型能够生成一个回答问题的思维链。从这个思维链中，他们能够提取更准确的答案。



图片来源: <https://arxiv.org/pdf/2109.01652.pdf>

从技术上讲，完整的零样本思维链过程涉及两个单独的提示/补全结果。在下面的图像中，左侧的顶部气泡生成一个思维链，而右侧的顶部气泡接收来自第一个提示（包括第一个提示本身）的输出，并从思维链中提取答案。第二个提示是一个自我增强的提示。

以下是一些演示（仅执行推理提取）。这个演示展示了 GPT-3 模型无法解决一个简单的数学问题，而第二个演示使用了零样本思维链提示，并成功地解决了这个问题。可以看出，与思维链提示相比，零样本思维链提示要简单得多。

正确提示：

输入：

如果 John 有 5 个梨，吃了 2 个，又买了 5 个，然后把 3 个给了他的朋友，他还剩下多少个梨？

让我们进一步思考。

输出：

John 起初有 5 个梨，他吃了 2 个梨，还剩下 3 个梨。他又买了 5 个梨，一共有 8 个梨。他把 3 个梨子给了他的朋友，他现在只剩下 5 个梨。

差的提示：

输入：

如果 John 有 5 个梨，吃了 2 个，又买了 5 个，然后把 3 个给了他的朋友，他还剩下多少个梨？

输出：

John 有 8 个梨。

Kojima 等人尝试了许多不同的零样本思维链提示（例如“让我们按步骤解决这个问题。”或“让我们逻辑思考一下。”），但他们发现“让我们一步一步地思考”对于他们选择的任务最有效。

零样本思维链也有效地改善了算术、常识和符号推理任务的结果。然而它通常不如思维链提示过程有效，不过在获取思维链提示的少量示例有困难的时候，零样本思维链可以派上用场。

作为提示工程师，我们也可以进一步发现类似的多任务提示，以引出大语言模型广泛的认知能力，而不局限在某种的特定的技能。

1.5 多模态思维链

如前所述，大语言模型（LLM）可以通过利用链式思维（CoT）提示来生成中间推理链作为推断答案的理由，具备良好的复杂推理性能。然而，现有的 CoT 研究都集中在语言模态上，忽略了视觉模态的重要性，视觉模态可以提供丰富的信息和知识，有助于解决一些需要多模态推理的问题，如科学问答。例如，如果我们想让一个大语言模型回答一个关于光合作用的问题，我们可以给它一个这样的提示：

Q：光合作用是什么？

A：光合作用是一种 [MASK] 过程。

这个问题需要模型具有一定的科学知识，但是如果我们只给出语言提示，模型可能无法准确地回答。因此，我们可以给出一个多模态提示，即在语言提示的基础上，添加一个相关的图像作为输入。例如，我们可以给出一个这样的提示：

Q: 光合作用是什么？

A: 光合作用是一种 [MASK] 过程。

I: [一张表示光合作用的图] >

这个图像可以帮助模型理解光合作用的原理和步骤，从而更好地回答问题。但是，如何将语言和视觉模态有效地结合起来，并利用它们生成更好的推理链呢？

研究人员提出了多模态思维链（**Multimodal-CoT**），一个将语言和视觉模态结合到一个两阶段的框架中，将理由生成和答案推断分开的方法。这种方法可以利用多模态信息来生成更好的理由，从而提高答案推断的性能。**Multimodal-CoT** 方法的核心思想是：利用 **LLM** 的强大的语言能力来生成基于多模态信息的推理链，而不是直接训练一个多模态推理模型。因此，我们不需要在预训练阶段就将视觉特征融合到 **LLM** 中，而是在微调阶段通过提示来实现。

Multimodal-CoT 包含两个阶段：理由生成和答案推断。在第一阶段，**Multimodal-CoT** 方法使用一个大语言模型（如 **GPT-3.5**）来生成基于多模态信息的推理链作为答案的理由。在第二阶段，使用另一个大语言模型（如 **UnifiedQA**）来根据生成的推理链来推断最终的答案。这样就可以利用不同大语言模型的优势，并避免它们之间的冲突或干扰。

例如，对于上面提到的光合作用问题，我们可以使用 **Multimodal-CoT** 方法来得到以下结果：

Q: 光合作用是什么？

A: 光合作用是一种 [MASK] 过程。

I: [一张表示光合作用的图]

R: 光合作用是一种生物化学过程，它发生在植物、藻类和某些细菌中。它利用太阳能将水和二氧化碳转化为有机物质和氧气。它分为两个主要阶段：光反应和暗反应。光反应在叶绿体的类囊体中进行，它将光能转化为化学能，产生 ATP 和 NADPH。暗反应在叶绿体的基质中进行，它利用 ATP 和 NADPH 将二氧化碳还原为葡萄糖。

A: 光合作用是一种生物化学过程。

可以看到，**Multimodal-CoT** 方法可以生成一个详细的推理链作为答案的理由，并

且可以准确地推断出答案。这说明了此方法可以有效地利用多模态信息来提高大语言模型的推理能力。研究者在 ScienceQA（标注详细解释的多模态科学问答数据集）基准测试上进行了实验，结果表明不到 1B 参数模型 UnifiedQA 的效果超过了 GPT3.5，甚至超过了人类的表现。具体来说，UnifiedQA 在 ScienceQA 数据集上的准确率为 91.68%，而 GPT-3.5 为 75.17%，人类为 90.00%。

Multimodal-CoT 方法不需要模型本身具备多模态的推理能力，提示工程师可以在直接通过提示来引导模型利用多模态信息进行推理。也就是说，Multimodal-CoT 方法并不改变 LLM 的预训练过程，也不需要对 LLM 进行额外的多模态训练。它只是在微调阶段，通过给 LLM 提供一个包含语言和视觉模态的输入，以及一个指定输出格式的提示，来激发 LLM 的多模态推理能力。例如，对于一个需要多模态推理的问题，我们可以给 LLM 一个这样的输入：

Q: 什么是牛顿第一定律？

A: 牛顿第一定律是 [MASK]。

I: [一张表示牛顿第一定律原理的图]

其中 Q 表示问题，A 表示答案，I 表示图像。我们还可以给 LLM 一个这样的提示：

P: 牛顿第一定律是关于 [物体] 的 [性质] 的 [规律]。

其中 P 表示提示，[物体]、[性质]和[规律]表示输出格式中需要填充的槽位。通过这样的输入和提示，我们可以引导 LLM 利用图像中的信息来生成一个合适的答案。例如，LLM 可能会生成这样一个答案：

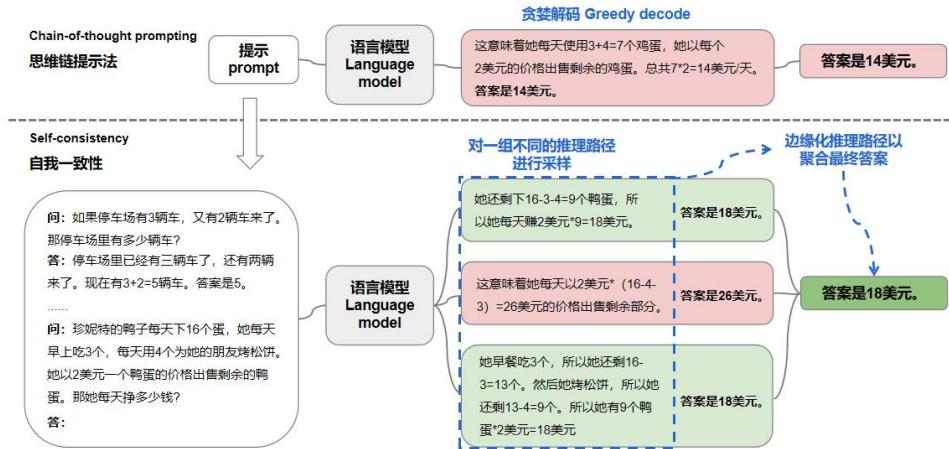
A: 牛顿第一定律是关于 [惯性系中的物体] 的 [运动状态] 的 [保持不变] 的规律。

可以看到，这个答案不仅正确地回答了问题，而且还符合输出格式，并且与图像中的内容一致。这说明了 Multimodal-CoT 方法可以通过提示来激发 LLM 的多模态推理能力。

1.6 自洽性

自洽性（Self-consistency）是对 CoT 的延展，它不仅仅是生成一个思维链，而是生成多个思维链，然后取多数答案作为最终答案。

在下面的图中，左侧的提示是使用少样本思路链范例编写的。使用这个提示，独立生成多个思路链，从每个思路链中提取答案，通过“边缘化推理路径”来计算最终答案。实际上，这意味着取多数答案。



图片来源:<https://arxiv.org/pdf/2203.11171.pdf>

Ye, X.等人的研究表明，自治性在一系列算术和常识推理任务中显著提高了精度，即使普通的思路链提示被发现无效，自治性仍然能够改善结果。除了提高准确率之外，“自治性”在使用语言模型进行推理任务时还有助于收集推理依据，并提供不确定性估计和改进语言模型输出的校准。可以利用“自治性”生成更好的监督数据来微调模型，从而在微调后的单次推理运行中，模型可以给出更准确的预测。

1.7 知识生成

知识生成（Knowledge Generation）是指使用大语言模型生成能够帮助解决特定任务的知识语句。它可以通过提示大语言模型生成与任务相关的语句，然后将这些语句集成到推理模型中，以提高模型的性能。通过这种方式，大语言模型可以作为一种灵活的、高质量的知识来源，帮助解决基于常识推理的任务。

如果要让模型回答一个常识推理问题，例如：高尔夫的一部分是试图获得比其他人更高的总分。是或否？可以先让模型生成一些关于高尔夫的知识，例如：高尔夫的目标是用最少的击球次数完成一组洞。每次击球都算作一分，总分用来决定比赛的胜者。

具体来说，知识生成提示（Generated Knowledge Prompting）要求 LLM 在生成回应之前生成与问题相关的可能有用的信息。该方法由两个中间步骤组成，即知识生成和知识集成。



图片来源:<https://arxiv.org/pdf/2110.08387.pdf>

在知识生成步骤中，要求 LLM 生成有关问题的一组事实。大语言模型将以 **few-shot** 方式进行提示，如下所示。使用相同提示生成 M 个不同的知识。

接下来，我们生成“知识增强”问题，并用它们提示 LLM 获得最终答案。假设我们正在尝试回答问题“大多数袋鼠有 <mask> 肢体”。假设在知识生成步骤中，我们生成了 2 个知识 ($M=2$)：

知识 1：“袋鼠是生活在澳大利亚的有袋动物。”

知识 2：“袋鼠是有 5 条肢体的有袋动物。”

现在，我们将每个知识与问题连接起来，生成知识增强的问题：

知识增强问题 1：“大多数袋鼠有<mask>肢体。袋鼠是生活在澳大利亚的有袋动物。”

知识增强问题 2：“大多数袋鼠有<mask>肢体。袋鼠是有 5 条肢体的有袋动物。”

然后，我们用这些知识增强的问题提示 LLM，并获得最终答案的提案：

答案 1：“4”

答案 2：“5”

我们可以选择概率最高的答案作为最终答案。

研究表明，知识可以在推理时通过简单地插入进行整合，无需为知识整合微调模型。该方法的成功凸显了语言模型可以作为灵活、高质量的常识推理知识的来源。

利用知识生成生成提示的好处是可以利用 LLM 中包含的隐式知识，为模型注入更多的显式知识，从而提高模型的推理能力、准确性和可解释性。利用知识生成生成提示的难点是需要设计合适的生成提示、知识选择和集成方法，以及处理生

成知识中可能存在的错误或不一致性。

1.8 主动提示

主动提示（Active-Prompt）是一种新的提示方法，它可以根据不同的任务需求，自动地生成和优化适合 LLMs 的示例提示。示例提示是一种用人类设计的 CoT 推理来注释的文本，它可以引导 LLMs 进行特定的推断和输出。Active-Prompt 的基本思想是利用 LLMs 的不确定性来选择最需要人类注释的问题，从而提高提示的效率和质量。

例如，如果我们想用 LLMs 来做情感识别任务，我们可以用 Active-Prompt 来生成和优化示例提示。首先，我们可以给出一个或几个简单的示例提示，比如：

-这部电影很好看，我很喜欢。[积极]

-这个餐厅的服务太差了，我很生气。[消极]

然后，我们可以用这些示例提示来查询 LLMs，并让它们对一些训练问题给出 k 个可能的答案。比如：

-你今天过得怎么样？[积极/中立/消极]

-这本书很无聊，我都不想看下去了。[积极/中立/消极]

接着，我们可以让 LLMs 对每个问题给出不止一个答案，然后看看这些答案有没有不同的。如果有不同的，说明 LLMs 对这个问题不太确定，需要人类来帮忙。我们就挑选那些 LLMs 最不确定的问题，让人类给出正确的答案，然后把这些答案告诉 LLMs，让它们学习。比如：

-你今天过得怎么样？[中立]

最后，我们可以把新注释的问题加入示例提示中，并重复上述过程，直到达到满意的效果。

再例如，如果你正在写一篇文章，需要引用一些相关的数据和研究结果，但你并不确定应该怎么表达，这时候就可以使用 Active-Prompt。你可以输入一些关键字或简要的描述。

例如：“最近关于气候变化的研究发现……”

Active-Prompt 会自动生成一些相关的段落或句子，你可以从中选择最合适的进行

引用或修改。这样可以节省你的时间和精力，并且提高文章的质量和可读性。

作为提示工程师，我们可以用好 Active-Prompt 这个提示技术，来提高我们的写作效率和质量。我们可以利用 Active-Prompt 来自动生成和优化适合不同任务和领域的示例提示，从而减少人工注释的工作量和成本。同时，我们也可以利用 Active-Prompt 来评估和改进我们已有的示例提示，从而提高它们的有效性和鲁棒性。

1.9 自动提示工程

受经典的程序合成和人类的提示工程方法的启发，研究者提出通过自动提示工程（APE）用于自动生成和选择指令。APE 是指一种自动生成和选择提示的框架，由 Zhou 等人研究提出。APE 的目标是根据任务的输出示例，利用 LLM 作为生成器和搜索器，生成一些可能的提示候选，并根据评分函数选择最合适的提示。

举例说明：

如果要让模型进行常识推理，可以给模型一些问题和答案的示例来引导它，例如：“高尔夫运动的目标是比其他人获得更高的总分吗？是或者否。答案：否。”通过这种方式，您可以引导模型理解问题并产生正确的答案。

APE 会利用 LLM 生成一些可能的提示。

例如：让我们一步一步地思考这个问题，以确保我们有正确的答案。

APE 会利用另一个 LLM 执行这些提示，并根据正确率、可解释性等指标选择最优的提示。

APE 的好处是可以自动优化提示，提高 LLM 的推理能力、准确性和可解释性，同时减少人工编写提示的成本和时间。APE 的难点是需要设计合适的生成方法、搜索方法和评分函数，以及处理生成提示中可能存在的错误或不一致性。

研究表明，APE 方法在多数 NLP 任务中实现了比人工注释器生成的指令更好或相当的性能，大型语言模型是人类级别的提示工程师，同时在零样本性能、少样本上下文学习性能和真实性方面都具备非常好的效果。作为提示工程师，可以用好这些自动化工具，往往具有事半功倍的效果。

1.10 定向刺激提示

大语言模型可以生成流畅和多样的文本，但是它们也经常出现一些不良的行为，比如偏离主题、重复或幻觉。要解决这些问题，需要一些方法来对 LLMs 进行细

粒度的指导，让它们在特定的下游任务上生成更符合期望的文本。提示工程师的工作是设计和优化这些方法，以提高 LLMs 的生成质量和可控性。

Wang 等研究人员在 2021 年提出了定向刺激提示（Directional Stimulus Prompting, DSP），这是一种利用一个小型的语言模型（称为策略 LM）来指导一个大型的语言模型（称为冻结 LLM）的方法。它的原理是用一个小的语言模型（LM）来给 LLM 一些提示或者方向，让 LLM 知道要做什么，怎么做。这些提示或者方向是一些词或者符号，可以和原来的输入一起输入到 LLM 中，让 LLM 按照这些提示或者方向来生成输出。这个小的语言模型（LM）可以通过学习一些数据或者反馈，来自动地生成和优化这些提示或者方向。

举个例子，如果我们想让 LLM 写一篇关于熊猫的文章，我们可以用 DSP 来给 LLM 一些提示或者方向。首先，我们可以用一个小的语言模型（LM）来生成一些词或者符号，比如：

- 熊猫
- [标题]
- [段落]
- [结尾]

然后，我们可以把这些词或者符号和原来的输入（比如“写一篇关于熊猫的文章”）一起输入到 LLM 中，让 LLM 按照这些提示或者方向来生成输出。比如：

- 熊猫
- [标题] 熊猫：中国的国宝
- [段落] 熊猫是一种可爱的动物，它们主要生活在中国的四川、甘肃和陕西等地。它们有着黑白相间的毛皮，圆圆的脸和大大的眼睛。它们喜欢吃竹子，每天要吃 12 公斤到 38 公斤的竹子。
- [段落] 熊猫是一种濒危物种，它们的数量非常稀少。据估计，目前野生的熊猫只有约 1800 只。人类的活动，如砍伐森林、建设道路、开发旅游等，都对熊猫的生存造成了威胁。为了保护熊猫，中国政府和国际组织都采取了很多措施，如建立自然保护区、进行人工繁殖、开展科学的研究等。
- [结尾] 熊猫是中国的国宝，也是世界的财富。它们不仅代表了中国的文化和历史，也展示了自然界的多样性和美丽。我们应该尊重和爱护熊猫，让它们能够安全和快乐地生活在这个星球上。

作为提示工程师，我们可以用好 DSP 这个技术，来让 LLM 更好地完成各种任务。我们可以利用 DSP 来自动生成和优化适合不同任务和领域的提示或者方向，从而减少人工设计的工作量和成本。同时，我们也可以利用 DSP 来评估和改进我们已

有的提示或者方向，从而提高它们的有效性和鲁棒性。

1.11 图提示

图形是一种可以表示很多事物之间的关系的方式，比如网页之间的链接，或者社交网络中的好友关系。图神经网络（GNN）是一种专门用于图数据的深度学习模型，它可以从图结构和属性中学习有效的表示。但是，不同的图形有不同的特点，计算机要想在不同的图形上做不同的任务，就需要有很多的标注数据来告诉它该怎么做，这样工作量就很大，难度很高，而且很浪费时间和资源。

近年来，预训练 GNN 模型可以一定程度上提高模型的泛化能力和迁移能力，降低对标注数据的依赖。然而，预训练模型和下游任务之间仍然存在一定的差距，导致预训练模型不能充分发挥其潜力。为了缩小这种差距，常见的方法是微调（Fine-tuning），即在下游任务上继续训练预训练模型的所有参数。微调虽然可以提高模型性能，但也有一些缺点，例如需要较多的计算资源、容易过拟合、不利于多任务学习等。

基于此，研究人员提出了图提示（GraphPrompt），一个统一图神经网络预训练和下游任务的方法。简单来说，GraphPrompt 框架就象是一个老师，它可以根据不同的学生（下游任务）和课本（预训练模型）来设计不同的问题（提示），引导学生从课本中学习到最有用的知识，并用这些知识来解决实际问题（下游目标）。

GraphPrompt 框架由两个主要部分组成：一个通用的任务模板和一个可设计的图提示。通用任务模板的作用是将预训练和下游任务的目标对齐，使得预训练模型能够理解下游任务的语义，并为下游任务提供合适的输出空间。图提示是指一个可设计的图结构，它可以根据输入图和问题生成一个新的图作为预训练模型的输入。图提示的作用是将输入图和问题转换为一种更适合预训练模型理解和处理的形式，从而帮助预训练模型从中提取出最相关的知识，并用于下游任务。

例如：我们考虑一个图分类任务，其中给定一个图，要求预测它的标签。例如，给定一个分子图，要求预测它是否是有毒的。为了使用图提示，我们首先定义一个通用的任务模板，如下所示：

输入：一个图 G

输出：一个标签 y

提示：一个矢量 p

然后，我们根据具体的任务生成一个任务特定的提示矢量 p ，例如：

```
p=[toxic,non-toxic]
```

这个提示矢量可以指导模型在预训练阶段学习有关图分类任务的知识，例如如何区分有毒和无毒的分子。在下游阶段，我们将输入图 G 和提示矢量 p 一起喂给模型，让它输出一个标签 y 。例如：

```
G=[C, H, O, N, ...]  
p=[toxic, non-toxic]  
y=toxic
```

这样，我们就可以利用 GraphPrompt 来统一预训练和下游任务，并提高模型在图分类任务上的性能。

总之，作为提示工程师，我们可以利用 GraphPrompt 框架来提高 GNNs 在不同图数据上的表现，从而完成不同类型和领域的下游任务。