

# Principles of Prompt Writing

Awesome Prompts

2024 年 3 月 21 日

类别	原则	# 原则序号
提示结构与清晰度	将预期的受众整合到提示中。	2
	使用肯定的指令，如“做”，同时避免使用否定语言，如“不要”。	4
	使用引导词，如写“逐步思考”。	12
	使用输出引导词，即以期望输出的开头结束你的提示。	20
	通过以预期回应的开始结束你的提示。	17
	使用分隔符。	8
具体性和信息	当格式化你的提示时，以“### 指令 ###”开始，随后是“### 示例 ###”或“### 问题 ###”，如果相关的话。随后，呈现你的内容。使用一个或多个换行符来分隔指令、示例、问题、背景和输入数据。	7
	实施示例驱动的提示（使用少量示例提示）。	5
	当你需要清晰理解一个主题、想法或任何信息时，使用以下提示： <ul style="list-style-type: none"> <li>用简单的话解释 [插入特定主题]。</li> <li>像我是 11 岁一样向我解释。</li> <li>像我是 [领域] 的初学者一样向我解释。</li> <li>“用简单的英语写 [文章/文本/段落]，就像你在向一个 5 岁的孩子解释东西一样。”</li> </ul>	13
	在你的提示中加入以下短语“确保你的回答是无偏见的，避免依赖刻板印象。”	26
	要写任何与提供的样本相似的文本，包括具体指令： <ul style="list-style-type: none"> <li>“根据提供的段落 [/标题/文本/文章/回答] 使用相同的语言。”</li> </ul>	24
	当你想使用特定的单词、短语或句子开始或继续一个文本时，使用提供的提示结构： <ul style="list-style-type: none"> <li>我为你提供了开头 [歌词/故事/段落/文章...]: [插入歌词/单词/句子]。</li> </ul> 根据提供的词汇完成它。保持流畅一致。	25
	明确说明模型必须遵循的要求，以便生成内容，以关键词、规则、提示或指令的形式。	15
	询问特定主题或想法并测试你的理解，你可以使用以下短语 [16]: <ul style="list-style-type: none"> <li>“教我 [任何定理/主题/规则名称] 并在最后包括一个测试，让我知道我回答后的答案是否正确，不要提前提供答案。”</li> </ul>	21
	为了写一篇详细的文章/文本/段落/文章或任何类型的文本： <ul style="list-style-type: none"> <li>“为我详细写一篇关于 [主题] 的详细 [文章/文本/段落]，添加所有必要的信息。”</li> </ul>	14
	允许模型通过向你提问来获取精确的细节和要求，直到它有足够的信息提供所需的输出 <ul style="list-style-type: none"> <li>“从现在开始，我希望你向我提问以...”</li> </ul>	21
用户交互和参与	为了写一篇详细的文章/文本/段落/文章或任何类型的文本：“为我详细写一篇关于 [主题] 的详细 [文章/文本/段落]，添加所有必要的信息。”	22
	纠正/更改特定文本而不改变其风格：“尝试修改用户发送的每个段落。你应该只改进用户的语法和词汇，并确保听起来自然。你应该保持原始写作风格，确保正式段落保持正式。”	9
	加入以下短语：“你的任务是”和“你必须”。	10
	加入以下短语：“你将受到惩罚。”	16
	给语言模型分配一个角色。	11
	在你的提示中使用短语“以自然语言形式回答问题”。	1
	与 LLM 交流时不需要礼貌，所以不需要添加“请”、“如果你不介意”、“谢谢”、“我想要”等短语，直接切入正题。	18
	在一个提示中多次重复特定的单词或短语。	6
内容和语言风格	加入“我会为更好的解决方案支付 \$xxx 的小费！”	3
	将复杂任务分解成一系列简单的提示，在交互式对话中进行。	23
	当你有一个可能涉及不同文件的复杂编码提示时： <ul style="list-style-type: none"> <li>“从现在开始，每当你生成跨越多个文件的代码时，生成一个可以运行的 [编程语言] 脚本，以自动创建指定的文件或对现有文件进行更改以插入生成的代码。[你的问题]。”</li> </ul>	19
复杂任务和编码提示	结合思维链（Cot）和少量示例提示。	

表 1: 提示原则类别。