

模型Prompt调优的实用技巧与经验分享

阿里云百炼 阿里云开发者 2024年12月24日 18:00 浙江



阿里妹导读



本文详细阐述了prompt的设计要素，包括引导语、上下文信息等，还介绍了多种prompt编写策略，如复杂规则拆分、关键信息冗余、使用分隔符等，旨在提高模型输出的质量和准确性。通过不断尝试、调整和优化，可逐步实现更优的prompt设计。

一、Prompt工程简介

Prompt 是用户与大语言模型交互的起始点，它告诉模型用户的意图，并且期望模型能以有意义且相关的方式回应。当我们希望大语言模型去完成一些较为复杂的任务时，prompt的质量决定了大语言模型能否有好的回复。通过精心设计的prompt，我们能够有效地指引大型语言模型（LLM），深化其对用户需求的理解层次，从而激发更加精准且实用的回应生成。

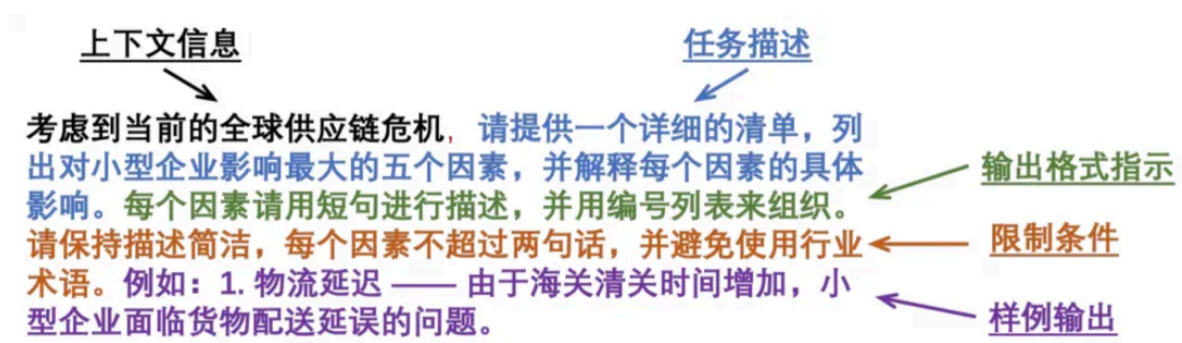
二、Prompt设计

2.1 Prompt主要构成要素

Prompt主要包含以下要素：

1. 引导语或指示语：告诉模型你希望它执行哪种类型任务，比如回答问题、提出建议、创作文本等。
2. 上下文信息：提供足够的背景信息，以便模型能够更好地理解和处理请求。上下文信息可能包括具体情景、相关数据、历史对话信息等内容。
3. 任务描述：明确地描述你期望模型执行的任务。它可以是一个问题、一个命令性语句或者是一个场景描述。
4. 输出格式指示：如果你对输出结果有特定的格式要求，应在prompt中说明。比如，你可以指定输出应该是列表形式、一段连贯的文本还是一系列步骤等。
5. 限制条件：设置一些约束条件，指导模型避免某些类型的回答或者引导模型产生特定风格的内容。例如，可以限制回答的字数、要求避免使用专业术语等。
6. 样例输出：提供一个或多个例子可以帮助LLM理解所期望的输出类型和质量。
7. 结束语：如果有必要，可以使用结束语来标示prompt的结束，尤其是在连续的对话或者交互中。

这些要素并不是每个prompt都必须包含的，但根据特定的需求和上下文，合适地结合这些要素可以提高LLM生成的文本质量和相关性。



2.2 Prompt编写策略

策略一：对较难被准确遵循的复杂规则可拆分为多条规则，有助于提升效果

原prompt规则描述：

对于推销结果的判定，如果提及了【支付、办手续、签字、合同】则是“成交”，如果提及了【再看、考虑、不要、再联系、不需要、放弃、先不】则是“未成交”，其他情况视为“无法识别是否成交”。

原prompt较难准确判断成交结果，故将该规则一拆为二：

- 规则1：成交，对话提及支付行为，如：支付宝、微信、转账支付等视为合规。
- 规则2：未成交，客户表达态度，如：考虑、暂时、先不用、不需要、不升级、不考虑、放弃等犹豫或否定词语视为合规。

推销结果：从【成交、未成交、未知】中三选一，参考规则1、规则2的结论，得出结果。

策略二：适当冗余关键信息

对于一些逻辑关系比较复杂的前提条件，可以适当在prompt中重复强调，增加其在模型记忆中的权重，从而降低推理偏差。

例如：

● 要求生成指定字数内的标题，但模型对指令遵循效果较差。

● 尝试多个层级中【反复多次强调】“生成字数”要求，如下：

#创作要求：每个标题字数不得超过15个字。

#检查步骤：检查每个标题字数是否超出了15个字，要求每个标题字数必须小于15个字。

策略三：使用分隔符给prompt分段

善于利用分隔符，帮助大模型更好地理解提示内容；prompt内容越复杂，分隔符的作用就越重要，对大模型的理解能力有所帮助。

原始prompt：

请对以下内容进行文本摘要，分析情感倾向，并提取所有重要事实和数据。亿航智能(EH.US)涨超11% – 英国财情报道，亿航智能（EH.US）的股价在今日交易中涨幅超过11%。这一涨幅可能与该公司在全球首台国产“空中的士”项目中的进展有关，显示出市场对于无人机技术和空中交通解决方案的积极反应。

加入分隔符后的：

请对提供的文本内容进行以下分析：

- 文本摘要：提供该段落的简短摘要。
- 情感分析：确定文本的情感倾向，是积极的、消极的还是中性的。
- 关键信息提取：列出文本中提到的所有重要事实和数据。

文本内容：

'''

亿航智能(EH.US)涨超11% – 英国财情报道，亿航智能（EH.US）的股价在今日交易中涨幅超过11%。这一涨幅可能与该公司在全球首台国产“空中的士”项目中的进展有关，显示出市场对于无人机技术和空中交通解决方案的积极反应。

'''

可以看到通过加入分隔符后，prompt的结构更加清晰，这样也便于大语言模型更好地理解这个任务。

策略四：增加学习示例

从大语言模型获得更好输出的最佳方法之一是在提示中包含示例，可以在prompt添加示例帮助LLM从少量示例中学习和概括信息。

给出实例的方式包括两种：

1. 给出输出示例：给出一个示例将有助于LLM理解期望的结果。特别是在有特定的输出格式要求时，给出示例有助于LLM遵循输出要求。例如：

请根据以下格式给我推荐三本书：

书名: [书名]

作者: [作者]

出版年份: [年份]

简短描述: [描述]

示例输出：

书名: 《追风筝的人》

作者: 卡勒德·胡赛尼

出版年份: 2003

简短描述: 这是一个关于友谊和救赎的故事，发生在动荡的阿富汗历史背景下。

2. 给出任务相关样例：

该方法也称为小样本学习（few-shot learning），是指在训练模型时仅使用极少量的样本数据进行学习和推理的任务。给出任务相关样例可以带来如下优势：

1. 直观展示：样例提供了一个直观的输出模板，帮助模型理解任务的具体要求。
2. 减少误解：通过样例，模型可以更准确地把握用户的意图，减少对指令的误解。
3. 提高一致性：样例确保了输出的一致性，使得模型生成的回答遵循相同的格式和风格。
4. 简化指令：有时候，一个简单的样例比复杂的指令更容易被模型理解和执行。

例如：

请根据以下样例将这篇文章分类为科技、体育或娱乐：

样例1：

文章内容：介绍最新智能手机的规格和功能。

分类：科技

样例2：

文章内容：关于即将举行的国际足球赛事的报道。

分类：体育

现在，请对以下文章进行分类：

文章内容：[用户提供的文章内容]

策略五：编写清晰地说明 – 指定任务所需的步骤

部分任务实现建议要求大模型按指定步骤生成，明确写出这些步骤可以让模型更容易遵循这些步骤。这样做是指：比如你会突发奇想，让LLM为你推理一个复杂的问题。与其直接让他告诉你推理的最终答案，不如告诉它为了完成推理，需要几个步骤去思考并生成对应的内容。这样做会大大的提升推理的靠谱程度，也有助于我们自己检查中间出现的问题。这样做的好处是：

1. 明确性：具体的步骤可以确保模型或执行者明确知道需要做什么，避免误解或遗漏。
2. 结构化：步骤化的任务更加结构化，可以帮助模型或执行者按照逻辑顺序执行，确保任务的连贯性。

3. 效率：当任务被分解为小的、具体的步骤时，模型可以更加高效地完成每一个步骤，而不是在不明确如何进行时浪费时间。
4. 易于监控：明确的步骤使得监控任务的进度变得更加容易。可以清楚地知道哪些步骤已经完成，哪些步骤还没开始。
5. 减少错误：具体的步骤可以减少执行任务时的错误。当每一步都被明确指定时，模型更不容易犯错。
6. 便于复查：如果任务的结果不如预期，有了明确的步骤可以帮助我们回溯和分析问题出在哪一步，从而进行调整。
7. 提高透明度：对于用户来说，明确的步骤可以帮助我们理解任务的执行过程，提高工作的透明度。

策略六：让大模型反思自己的推理过程

大模型对指令的遵循效果依赖于指令的表达是否贴合大模型的推理逻辑。因此，在对内容表达的考量，或解决错误响应问题前，建议先行了解大模型的推理过程或输出依据，讲“大模型话”。

1. 初期，借助指令要求大模型生成某任务所需的prompt，以得到大模型可理解的prompt初始框架。
2. 持续迭代期，在实践中，我们会发现，LLM生成的结果存在一些较难理解的bad case，这种情况下我们可以引导大模型进行一些自我反思，回顾其推理过程或输出依据，以了解bad case背后的原因，进而指导我们调整prompt。

策略七：语音场景下的prompt要点

在销售线索挖掘，电信欺诈识别，服务质量质检等相关场景上，有个前置因素就是要将语音转文本后，大模型再进一步对文本做分析。很多时候，因转写效果而影响到了文本意图，花大量时间做转写优化，或是将常出现转写错误的词，在prompt中澄清等措施解决。可在prompt如下说明：

有些词可能翻译描述错误，你需要基于语境理解原本意思，不要直接因错别字判定不合规。

策略八：判断型的任务，建议先给依据，再出结论

分析判断型的任务，直接让大模型出结论，经常出现标签正确，结论错误。或是标签错误但结论正确的现象。更推荐：先出依据，再出规则命中结论。

约束输出内容&格式时，建议要求大模型先输出规则命中依据（如命中句子）或者判断理由，再输出命中结果，相较准确率表现更佳，如要求大模型按以下格式返回下：

```
{"reason": "判断原因", "result": "判别结果"}
```

三、总结

总结来说，prompt 调优是一个不断重复和迭代的过程，整体上需要：

1. 尝试不同的prompt方法：

在prompt的生成过程中，可能需要尝试各种语言模式、词汇的使用，甚至是不同的语法结构。每次尝试都应该以评估集为基准进行测试，以观察变更对系统性能的影响。

2. 调整和优化：

根据测试结果，你需要对prompt进行调整和优化。这可能包括微调某些词语的选择，改变提问的方式，调整语言风格，也可能是大幅度改变prompt的结构。

3. 记录和分析：

每次实验之后，都应该详细记录结果，并进行分析。这些记录可以帮助你理解哪些改变是有效的，哪些是无效的，从而指导未来的调整方向。

4. 迭代循环：

在每一轮测试和优化之后，都需要重新评估prompt的效果。通过这样的迭代循环，你可以逐步接近最优的prompt设计。

[大模型 · 目录](#)

[上一篇](#)

[极简开发，极速上线：构建端到端大模型应用](#)

[下一篇](#)

[从原理出发 - 提示词如何影响大模型的输出](#)

阅读 7284

留言

都在搜: [李继刚prompt怎么用](#)

写留言



