

AGI 提示工程指北

Prompt Engineering in Action

Press Space for next page →



教程大纲

- 基础 LLM vs. 指令微调 LLM
- 提示工程关键原则
- 提示工程需要迭代
- 模型能力：概括 (Summerize)
- 模型能力：推断 (Infer) · 情绪
- 模型能力：推断 (Infer) · 主题
- 模型能力：推断 (Infer) · 分类
- 模型能力：转换 (Transform)
- 模型能力：扩展 (Expand)

基础 LLM

Base LLM

- 模型行为：预测下一个（系列）单词
- 训练内容：互联网和其它来源的大量数据

Once upon a time, there was a unicorn
that lived in a magical forest with all her friends

基础 LLM 的默认行为是“续写”文本

What is the capital of France?
What is France's largest city?
What is France's population?
What is the currency of France?

训练数据（比如互联网上的文章）很可能是关于法国的问答列表

指令微调 LLM

Instruction Tuned LLM

- 模型行为：接收指令，输出预测结果
- 训练过程：
 - 从大量文本数据中训练出一个基础 LLM
 - 使用指令和良好尝试的输入输出进行微调和优化
 - 使用“人类反馈强化学习”的技术进行进一步细化

What is the capital of France?
The capital of France is Paris.

💡 现在互联网上大家使用到的 LLM 基本都是这类模型

提示工程关键原则

Principles of Prompting

原则一：编写清晰而具体的指令

- 策略①：使用分隔符清楚地限定输入的不同部分
- 策略②：要求模型结构化输出
- 策略③：要求模型检查是否满足条件
- 策略④：提供^w少样本提示 (Few Shot)

原则二：给模型充足的思考时间

- 策略⑤：指定完成任务所需要的步骤
- 策略⑥：指导模型制定自己的解决方法

模型的局限性

Model Limitations

可能会产生^w幻觉 (Hallucination)

- 不清楚自己的知识边界
- 虚构听起来很有道理但实际伤感不正确的东西

减少幻觉的策略

- 要求模型首先从文本中找到任何相关的引文
- 然后要它使用那些引文来回答问题
- 并将答案追溯回源文件

演示环境准备

基于 Val Town 平台，调用 ^{npm} OpenAI 来执行人机对话



@webup.chat v16

```
1  export const chat = async (prompt = "Hello world", model = "gpt-3.5-turbo") => {
2    // Initialize OpenAI API stub
3    const { Configuration, OpenAIApi } = await import("https://esm.sh/openai");
4    const configuration = new Configuration({
5      apiKey: '@me.secrets.OPENAI',
6      // Bypass llm.report for request logging
7      basePath: "https://api.openai.withlogging.com/v1",
8      baseOptions: {
9        headers: {
10          "X-API-Key": `Bearer ${@me.secrets.LLM_REPORT}`,
11        },
12      },
13    });
14    const openai = new OpenAIApi(configuration);
15    // Request chat completion
16
```

策略一：使用分隔符清楚地限定输入的不同部分

常见的分隔符可以是：` `` `，` "" `，` <> `，` <tag></tag> ` 等

- 输入里面可能包含其他指令，会覆盖掉你的指令
- 可以使用任何明显的标点符号将特定的文本部分与提示的其余部分分开

 @webup.chatSampleDelimiter v2

```
1 export const chatSampleDelimiter = (async () => {
2   const text = `你应该提供尽可能清晰、具体的指示，以表达你希望模型执行的任务。
3   这将引导模型朝向所需的输出，并降低收到无关或不正确响应的可能性。
4   不要将写清晰的提示与写简短的提示混淆。
5   在许多情况下，更长的提示可以为模型提供更多的清晰度和上下文信息，从而导致更详细和相关的输出。
6   `;
7   const prompt = `把用三个双引号括起来的文本总结成一句话。""${text}""`;
8   return await @webup.chat(prompt);
9 }
10 })();
```

给模型提供清晰、具体的提示，以指导其执行任务，避免无关或不正确响应，不要将清晰和简短混淆，长的提示可以提供更多清晰度和上下文信息，产生更详细和相关的输出。

策略二：要求一个结构化的输出

常见的结构化输出可以是 JSON、HTML 等格式

在以下示例中，我们要求 GPT 生成三本书的标题、作者和类别，并要求以 JSON 的格式返回给我们。



@webup.chatSampleFormatOutput v2

```
1 export const chatSampleFormatOutput = (async () => {
2   const prompt = `
3     请生成包括书名、作者和类别的三本虚构书籍清单,
4     并以 JSON 格式提供, 其中包含以下键: book_id、title、author、genre,
5     不要输出 JSON 内容以外的其它文本
6   `;
7   return await @webup.chat(prompt);
8 })();
```

```
{
  "books": [
    {
      "book_id": 1,
      "title": "The Shadow Queen",
      "author": "Sarah J. Maas",
      "genre": "Fantasy"
    },
    {
      "book_id": 2,
      "title": "The Shadow Throne",
      "author": "Sarah J. Maas",
      "genre": "Fantasy"
    }
  ]
}
```

策略三：要求模型检查是否满足条件

如果任务做出的假设不一定满足，我们可以告诉模型先检查这些假设，如果不满足，指示并停止执行。在如下示例中，我们将给模型一段没有明确步骤的文本。模型在判断后应回答未提供步骤。



@webup.chatSampleSatisfication v2

```
1 export const chatSampleSatisfication = (async () => {
2   const text = ` 
3     今天阳光明媚，鸟儿在歌唱。这是一个去公园散步的美好日子。
4     鲜花盛开，树枝在微风中轻轻摇曳。人们外出享受着这美好的天气，有些人在野餐，有些人在草地上放松。
5     这是一个完美的日子，可以在户外度过并欣赏大自然的美景。
6   `;
7   const prompt = ` 
8     您将获得由三个引号括起来的文本。
9     如果它包含一系列的指令，则需要按照以下格式重新编写这些指令：
10
11     第一步 - ...
12     第二步 - ...
13     ...
14     第N步 - ...
15`
```

策略三：要求模型检查是否满足条件（提供出路）

如果模型无法完成分配的任务，有时为模型提供备用路径可能会有所帮助。



@webup.chatSampleHallucination v0

```
1  export const chatSampleHallucination = (async () => {
2    const prompt = "请介绍一下 LangChain";
3    return await @webup.chat(prompt);
4 })();
```

LangChain是一种区块链平台，旨在为不同语言的开发人员创造更容易的方式来共享和使用代码和智能合约。它提供了各种工具和服务，以帮助开发者更方便地在



@webup.chatSampleHallucinationExit v0

```
1  export const chatSampleHallucinationExit = (async () => {
2    const prompt = "请介绍一下 LangChain, 如果你不清楚请直接回答“我不知道”";
3    return await @webup.chat(prompt);
4 })();
```

我不知道。

策略四：提供少量示例

即在要求模型执行实际任务之前，提供给它少量成功执行任务的示例

在如下示例中，我们要求模型以一致的风格作答；由于已经有了少量示例，它将以类似的语气回答。



@webup.chatSampleFewShot v1

```
1 export const chatSampleFewShot = (async () => {
2   const prompt = `
3     你的任务是以一致的风格回答问题。
4     <孩子>：教我耐心。
5     <祖父母>：挖出最深峡谷的河流源于一处不起眼的泉眼；最宏伟的交响乐从单一的音符开始；最复杂的挂毯以一根孤独的线开始编织。
6     <孩子>：教我韧性。
7     `;
8   return await @webup.chat(prompt);
9 })();
```

<祖父母>：韧性是持久不懈地坚持做一件事情，坚定不移地追求目标。就像冬天里的梅花，它在寒风中依然能开出美丽的花朵，因为它有不屈的韧性。我们可以进

2d 0 @ 0

策略五：指定完成任务所需的步骤

接下来我们将通过给定一个复杂任务，给出完成该任务的一系列步骤，来展示这一策略的效果

首先我们描述了杰克和吉尔的故事，并给出一个指令。由于提示不充分，输出的内容带了不需要的序号。



@webup.chatSampleStepByStep v1

```
1 export const chatSampleStepByStep = (async () => {
2   const text = ` 
3     在一个迷人的村庄里，兄妹杰克和吉尔出发去一个山顶井里打水。
4     他们一边唱着欢乐的歌，一边往上爬，然而不幸降临 — 杰克绊了一块石头，从山上滚了下来，吉尔紧随其后。
5     虽然略有些摔伤，但他们还是回到了温馨的家中。
6     尽管出了这样的意外，他们的冒险精神依然没有减弱，继续充满愉悦地探索。
7   `;
8   const prompt = ` 
9     执行以下操作：
10    1-用一句话概括下面用三个引号括起来的文本。
11    2-将摘要翻译成法语。
12    3-在法语摘要中列出每个人名。
13    4-输出一个 JSON 对象，其中包含以下键：French_summary, num_names。
14
15    请用换行符分隔您的答案。``` ${text} ````
16`;
```

策略五：指定完成任务所需的步骤（续）

我们给出一个更好的 Prompt，该 Prompt 指定了输出的格式

 @webup.chatSampleStepByStepFormat v3

```
1 export const chatSampleStepByStepFormat = (async () => {
2   const text = `
3     在一个迷人的村庄里，兄妹杰克和吉尔出发去一个山顶上的井里打水。
4     他们一边唱着欢乐的歌，一边往上爬，然而不幸降临 — 杰克绊了一块石头，从山上滚了下来，吉尔紧随其后。
5     虽然略有些摔伤，但他们还是回到了温馨的家中。
6     尽管出了这样的意外，他们的冒险精神依然没有减弱，继续充满愉悦地探索。
7   `;
8   const prompt = `
9     执行以下操作：
10    1–用一句话概括下面用三个引号括起来的文本。
11    2–将摘要翻译成英语。
12    3–在英语摘要中列出每个名称。
13    4–输出一个JSON对象，其中包含以下键：English_summary, num_names。
14
15    请使用以下格式输出：
16`
```

以下描述了兄妹杰克和吉尔在山顶上的经历。杰克不慎滚下山去，但他们还是回到家中继续探险。

策略六：指导模型在下结论前找出自己的解法

有时候，明确地指导模型在做决策之前要思考解决方案，我们会得到更好的结果

在如下示例中，我们给出一个问题和一个学生的错误解答，要求模型判断解答是否正确。

 @webup.chatSampleCheckAnswer v3

```
1 export const chatSampleCheckAnswer = (async () => {
2   const prompt = `
3     判断学生的解决方案是否正确。
4
5     问题：
6     我正在建造一个太阳能发电站，需要帮助计算财务。
7
8       土地费用为100美元/平方英尺
9       我可以以250美元/平方英尺的价格购买太阳能电池板
10      我已经谈判好了维护合同，每年需要支付固定的10万美元，并额外支付每平方英尺10美元
11      作为平方英尺数的函数，首年运营的总费用是多少。
12
13      学生的解决方案：
14      设x为发电站的大小，单位为平方英尺。
15      费用：
```

学生的解决方案：
设x为发电站的大小，单位为平方英尺。
费用：

策略六：指导模型在下结论前找出自己的解法（续）

我们可以通过指导模型先自行找出一个解法来解决这个问题。通过明确步骤，让模型有更多时间思考



@webup.chatSampleCheckAnswerWithStep v10

```
1 export const chatSampleCheckAnswerWithStep = (async () => {
2   const prompt = `
3     请作为一名初中数学老师判断学生的解决方案是否正确，请通过如下步骤解决这个问题：
4
5     步骤：
6     - 首先，自己解决问题，注意要进行实际的计算并验算得到的结果。
7     - 然后将你的解决方案和答案与学生的解决方案和答案进行比较，并评估学生的答案是否正确。
8     - 在自己完成问题之前，请勿直接决定学生的答案是否正确。
9
10    使用以下格式：
11
12    问题: <问题文本>
13    学生的解决方案: <学生的解决方案文本>
14    实际解决方案和步骤: <实际解决方案和步骤文本>
15    学生的解决方案和实际解决方案是否相同: <是或否>
```

提示工程需要迭代

当使用 LLM 构建应用程序时，很少在第一次尝试中就成功使用最终应用程序中所需的提示词

我们将以从产品说明书中生成营销文案这一示例，展示一些框架，以提示你思考如何迭代地分析和完善提示词。

```
!@webup.chatSamplePolishNothing v2

1 export const chatSamplePolishNothing = (async () => {
2   const spec = `

3     概述
4       美丽的中世纪风格办公家具系列的一部分，包括文件柜、办公桌、书柜、会议桌等。
5       多种外壳颜色和底座涂层可选。可选塑料前后靠背装饰（SWC-100）或10种面料和6种皮革的全面装饰（SWC-110）。
6       底座涂层选项为：不锈钢、哑光黑色、光泽白色或铬。椅子可带或不带扶手。
7       适用于家庭或商业场所。符合合同使用资格。
8
9     结构
10      五个轮子的塑料涂层铝底座。气动椅子调节，方便升降。
11
12     尺寸
13      宽度53厘米|20.87英寸
14      深度51厘米|20.08英寸
15      高度80厘米|31.50英寸
16`
```

这款办公椅是美丽的中世纪风格家具系列的一部分，包括文件柜、办公桌、书柜、会议桌等。我们提供多种外壳颜色和底座涂层可选。你可以选择塑料前后靠背装饰（SWC-100）或10种面料和6种皮革的全面装饰（SWC-110）。底座涂层选项为：不锈钢、哑光黑色、光泽白色或铬。椅子可带或不带扶手。适用于家庭或商业场所。符合合同使用资格。

问题一：生成文本太长

解决方法：要求模型限制生成文本长度

因为它太长了，所以我会澄清我的提示，并说最多使用 50 个字。

```
!@webup.chatSamplePolishTooLong v3

1 export const chatSamplePolishTooLong = (async () => {
2   const spec = `

3   概述
4     美丽的中世纪风格办公家具系列的一部分，包括文件柜、办公桌、书柜、会议桌等。
5     多种外壳颜色和底座涂层可选。可选塑料前后靠背装饰（SWC-100）或10种面料和6种皮革的全面装饰（SWC-110）。
6     底座涂层选项为：不锈钢、哑光黑色、光泽白色或铬。椅子可带或不带扶手。
7     适用于家庭或商业场所。符合合同使用资格。
8
9   结构
10    五个轮子的塑料涂层铝底座。气动椅子调节，方便升降。
11
12   尺寸
13    宽度53厘米|20.87英寸
14    深度51厘米|20.08英寸
15    高度80厘米|31.50英寸
16`
```

问题二：文本关注在错误的细节上

解决方法：要求模型专注于与目标受众相关的方面

修改提示让它更精确地描述椅子的技术细节，并要求在描述的结尾包括对应的 7 个字符产品 ID。

```
 @webup.chatSamplePolishBadFocus v6

1 export const chatSamplePolishBadFocus = (async () => {
2   const spec = `

3     概述
4       美丽的中世纪风格办公家具系列的一部分，包括文件柜、办公桌、书柜、会议桌等。
5       多种外壳颜色和底座涂层可选。可选塑料前后靠背装饰（SWC-100）或10种面料和6种皮革的全面装饰（SWC-110）。
6       底座涂层选项为：不锈钢、哑光黑色、光泽白色或铬。椅子可带或不带扶手。
7       适用于家庭或商业场所。符合合同使用资格。
8
9     结构
10      五个轮子的塑料涂层铝底座。气动椅子调节，方便升降。
11
12     尺寸
13      宽度53厘米|20.87英寸
14      深度51厘米|20.08英寸
15      高度80厘米|31.50英寸
16`
```

问题三：需要一个更好的呈现形式

解决方法：要求模型抽取信息并组织成表格，并指定表格的列、表名和格式（如 HTML）

```
1  export const chatSamplePolishHTMLTable = (async () => {
2    const spec = ` 
3      概述
4        美丽的中世纪风格办公家具系列的一部分，包括文件柜、办公桌、书柜、会议桌等。
5        多种外壳颜色和底座涂层可选。可选塑料前后靠背装饰 (SWC-100) 或10种面料和6种皮革的全面装饰 (SWC-110) 。
6        底座涂层选项为：不锈钢、哑光黑色、光泽白色或铬。椅子可带或不带扶手。
7        适用于家庭或商业场所。符合合同使用资格。
8
9      结构
10     五个轮子的塑料涂层铝底座。气动椅子调节，方便升降。
11
12      尺寸
13      宽度53厘米|20.87英寸
14      深度51厘米|20.08英寸
15      高度80厘米|31.50英寸
16`
```

大语言模型能力概览

概括（Summerize）、推断（Infer）、转换（Transform）、扩展（Expand）

模型能力：概括 (Summarize)

子能力：限制输出长度，侧重关键角度，提取关键信息

如果我们只想要提取某一角度的信息，并过滤掉其他所有信息，可以要求模型进行文本提取 (Extract)。

@webup.chatSampleExtract v1

```
1 export const chatSampleExtract = (async () => {
2   const text = `
3     这个熊猫公仔是我给女儿的生日礼物，她很喜欢，去哪都带着。
4     公仔很软，超级可爱，面部表情也很和善。但是相比于价钱来说，它有点小，我感觉在别的地方用同样的价钱能买到更大的。
5     快递比预期提前了一天到货，所以在送给女儿之前，我自己玩了会。
6   `;
7   const prompt = `
8     你的任务是从电子商务网站上生成一个产品评论的简短摘要。
9     请从以下三个反引号之间的评论文本中提取产品运输相关的信息，最多30个词汇。
10    评论: """${text}"""
11  `;
12  return await @webup.chat(prompt);
13 })();
```

快递提前提一天到货。

模型能力：推断（Infer）· 情绪

大语言模型可以从一段文本中提取正面或负面情感

在如下示例中，我们要求模型对一份客户反馈进行情绪推断，并抽取相关信息后进行格式化输出。

 @webup.chatSampleInferEmotion v0

```
1 export const chatSampleInferEmotion = (async () => {
2   const text = `

3     我需要一盏漂亮的卧室灯，这款灯具有额外的储物功能，价格也不算太高。
4     我很快就收到了它。在运输过程中，我们的灯绳断了，但是公司很乐意寄送了一个新的。几天后就收到了。
5     这款灯很容易组装。我发现少了一个零件，于是联系了他们的客服，他们很快就给我寄来了缺失的零件。
6     在我看来，Lumina 是一家非常关心顾客和产品的优秀公司！

7   `;
8   const prompt = `

9     从评论文本中识别以下项目：
10    - 情绪（正面或负面）
11    - 评论人是否表达了愤怒？（是或否）
12    - 评论者购买的物品
13    - 制造该物品的公司

14

15    评论用三个双引号分隔。将您的响应格式化为 JSON 对象，以“Sentiment”、“Anger”、“Item”和“Brand”作为键。
16`
```

模型能力：推断（Infer）· 主题

大语言模型也可以推断一段文本是关于什么的、有什么话题

在如下示例中，我们要求模型从一片虚构的文章中推断出其中的五个主题。

 @webup.chatSampleInferTopic v0

```
1 export const chatSampleInferTopic = (async () => {
2   const text = `
3     在政府最近进行的一项调查中，要求公共部门的员工对他们所在部门的满意度进行评分。
4     调查结果显示，NASA 是最受欢迎的部门，满意度为 95%。
5     一位 NASA 员工 John Smith 对这一发现发表了评论，他表示：
6     “我对 NASA 排名第一并不感到惊讶。这是一个与了不起的人们和令人难以置信的机会共事的好地方。我为成为这样一个创新组织的一员感到自豪。”
7     NASA 的管理团队也对这一结果表示欢迎，主管 Tom Johnson 表示：
8     “我们很高兴听到我们的员工对 NASA 的工作感到满意。我们拥有一支才华横溢、忠诚敬业的团队，他们为实现我们的目标不懈努力，看到他们的努力得到认可，我们感到非常自豪。”
9     调查还显示，社会保障管理局的满意度最低，只有 45% 的员工表示他们对工作满意。
10    政府承诺解决调查中员工提出的问题，并努力提高所有部门的工作满意度。
11  `;
12  const prompt = `
13    确定以下给定文本中讨论的五个主题。
14    每个主题用 1–2 个单词概括。输出时用逗号分割每个主题。
15    给定文本: """${text}"""
16  `;
```

模型能力：推断 (Infer) · 分类

基于对于主题的推断，大语言模型还可以帮忙把主题和给定的分类进行映射



@webup.chatSampleInferCategory v1

```
1 export const chatSampleInferCategory = (async () => {
2   const text = `
3     主要类别: 计费(Billing)、技术支持(Technical Support)、账户管理(AccountManagement)或一般咨询(General Inquiry)
4
5     计费次要类别:
6       取消订阅或升级(Unsubscribe or upgrade)
7       添加付款方式(Add a payment method)
8       收费解释(Explanation for charge)
9       争议费用(Dispute a charge)
10
11    技术支持次要类别:
12      常规故障排除(General troubleshooting)
13      设备兼容性(Device compatibility)
14      软件更新(Software updates)
15
```

模型能力：转换（Transform）

大语言模型非常擅长将输入转换成不同的格式，如多语种文本翻译、拼写及语法纠正、语气调整、格式转换等。如下的示例展示了一个综合样例，包括了文本翻译、拼写纠正、风格调整和格式转换。



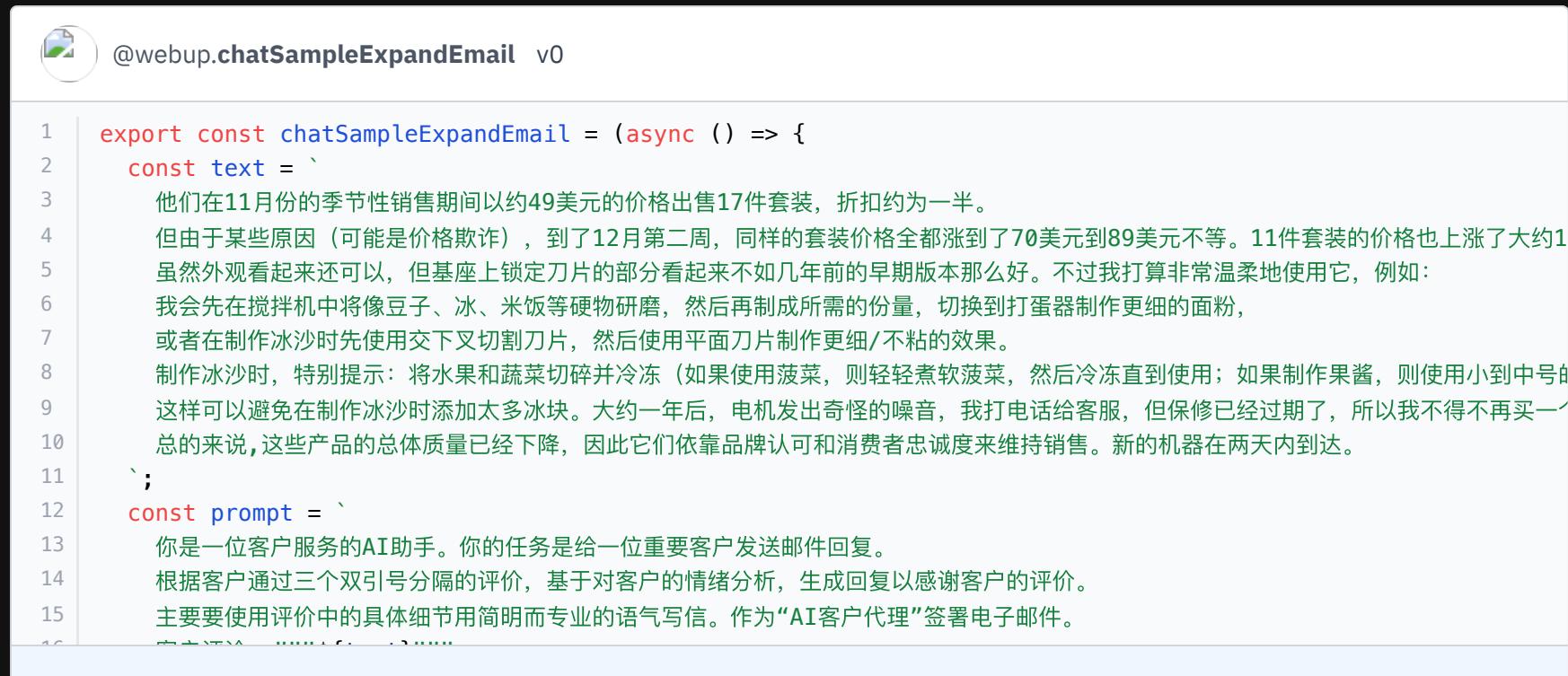
@webup.chatSampleTransform v4

```
1 export const chatSampleTransform = (async () => {
2   const text = `
3     Got this for my daughter for her birthday because she keeps taking
4     mine from my room. Yes, adults also like pandas too. She takes
5     it everywhere with her, and it's super soft and cute. One of the
6     ears is a bit lower than the other, and I don't think that was
7     designed to be asymmetrical. It's a bit small for what I paid for it
8     though. I think there might be other options that are bigger for
9     the same price. It arrived a day earlier than expected, so I got
10    to play with it myself before I gave it to my daughter.
11  `;
12  const prompt = `
13    针对以下三个双引号之间的英文评论文本：
14    首先进行拼写及语法纠错，然后将其转化成中文，
15    再将其转化成优质淘宝评论的风格，从各种角度出发，分别说明产品的就点与缺点，并进行总结。
16  
```

模型能力：扩展（Expand）

扩展是将短文本，例如一组说明或主题列表，输入给大型语言模型，让它生成更长的文本

如下示例中，我们将根据客户评价和对应的情感，要求模型撰写自定义电子邮件响应。



The screenshot shows a code editor window with a file named `@webup.chatSampleExpandEmail v0`. The code is written in JavaScript and defines two variables: `chatSampleExpandEmail` and `prompt`.

```
1 export const chatSampleExpandEmail = (async () => {
2   const text = `
3     他们在11月份的季节性销售期间以约49美元的价格出售17件套装，折扣约为一半。
4     但由于某些原因（可能是价格欺诈），到了12月第二周，同样的套装价格全都涨到了70美元到89美元不等。11件套装的价格也上涨了大约1
5     虽然外观看起来还可以，但基座上锁定刀片的部分看起来不如几年前的早期版本那么好。不过我打算非常温柔地使用它，例如：
6     我会先在搅拌机中将像豆子、冰、米饭等硬物研磨，然后再制成所需的份量，切换到打蛋器制作更细的面粉，
7     或者在制作冰沙时先使用交叉切割刀片，然后使用平面刀片制作更细/不粘的效果。
8     制作冰沙时，特别提示：将水果和蔬菜切碎并冷冻（如果使用菠菜，则轻轻煮软菠菜，然后冷冻直到使用；如果制作果酱，则使用小到中号的
9     这样可以避免在制作冰沙时添加太多冰块。大约一年后，电机发出奇怪的噪音，我打电话给客服，但保修已经过期了，所以我不得不再买一个
10    总的来说，这些产品的总体质量已经下降，因此它们依靠品牌认可和消费者忠诚度来维持销售。新的机器在两天内到达。
11  `;
12  const prompt = `
13    你是一位客户服务的AI助手。你的任务是给一位重要客户发送邮件回复。
14    根据客户通过三个双引号分隔的评价，基于对客户的情绪分析，生成回复以感谢客户的评价。
15    主要要使用评价中的具体细节用简明而专业的语气写信。作为“AI客户代理”签署电子邮件。
16`
```

参考资料

本教程在制作过程中参考和引用了以下资料（排名不分先后）的内容，特此鸣谢！

■ 视频资料

- [Short Courses | Learn Generative AI from DeepLearning.AI](#)

≡ 图文资料

- [入门：Prompts（提示词） | 通往 AGI 之路](#)

</> 代码资料

- [datawhalechina/prompt-engineering-for-developers: 吴恩达大模型系列课程中文版](#)
- [slidevjs/slidev: Presentation Slides for Developers](#)

感谢聆听 ❤

⌚ webup | ⚡ serviceup