

# LLM & System Prompts Evaluation

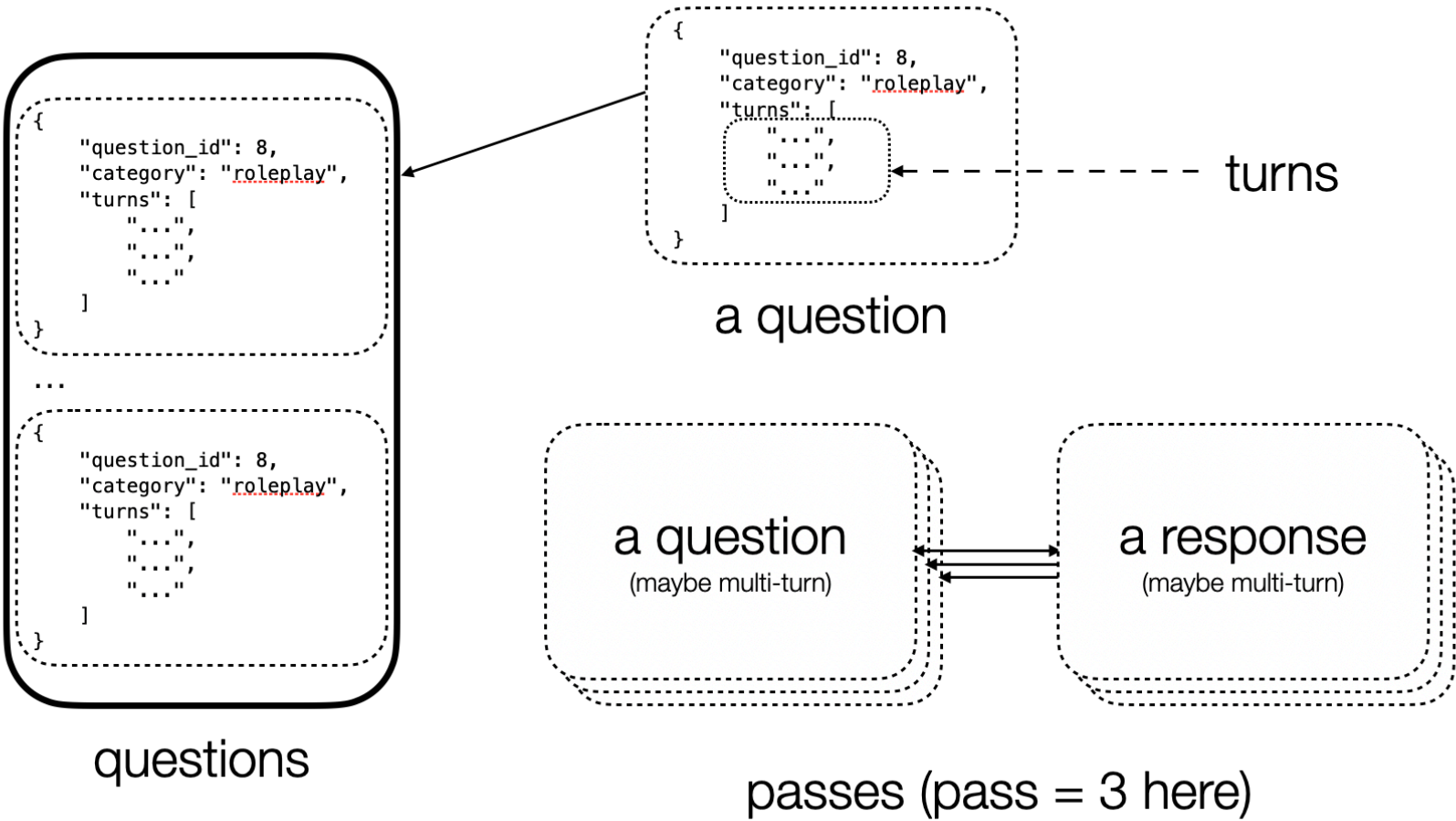
## 概述

本项目对 qwen-14b-chat、baichuan2-13b-chat-v1、gpt-3.5-turbo 三个预训练大模型进行测试，测试任务包括：代码生成、多轮对话角色扮演、数学推理、文案撰写。

项目提出了各任务下的表现最好的 final system prompt（同一任务下，不同模型使用相同 system prompt），并予以完整的设计过程。每个任务中 final system prompt 的有效性在实验部分通过详细的表格、折线图等结果对比给出。另外，本项目也定性地讨论了各模型的能力边界、缺陷、风险，并提出其他发现、大模型评估大模型可能性。

## 任务概述

每项任务中，有多个测试问题（以下称为 question）；每一问题下，可能为多轮对话或单轮对话（以下称为 turn）；并且，根据任务，每一问题的对话可能需要重复多次（以下称为 pass），例如代码生成任务中 `pass = 3`，其他任务中 `pass = 1`。如下。



我们将任务分成两类，开放任务（主题差异较大、自由度较高的任务）、推理任务（对逻辑、结果要求高的任务）。并在实验中总结了两种任务下的 system prompt 设计经验，见「[总结](#)」

### 数学推理 (math)

`pass = 1, turn = 1`。推理任务。本项目中，system prompt 将要求被测大模型根据用户需求推导数学问题，并给出详细说明、最终答案。问题范围包括简单的概率、平面几何、方程、计算。

指标：

**gpt-3.5-turbo 评价：**本任务中，使用 `./data/judge-prompts.jsonl` 中 `"math-v1"` 作为 gpt-3.5-turbo 的 system prompt 进行评价。

### 代码生成 (coding)

`pass = 3, turn = 1`。推理任务。本项目中，system prompt 将要求被测大模型根据用户需求编写程序，并给出详细的代码说明。

指标：

- pass@3：**

将提取不同 pass 下响应中的代码部分，创建子进程运行，若超时（5 秒）则认为失败。若 3 个 pass 中有一个通过运行则认为成功。本项目汇中固定 `seed = 2024`，代码生成任务中三个 pass 下分别为 `seed = 2023, 2024, 2025`。

本任务最终的 system prompt 构造将要求模型使用同路径下某一特定预先给出的文件（若需要读写文件）、要求模型自行制定样例输入输出，而非接受用户输入，以免在代码正确时仍然超时。详细讨论见「[实验-代码生成](#)」。
- gpt-3.5-turbo 评价：**

将提取不同 pass 下响应的全部分，要求 gpt-3.5-turbo 进行评估，给出得分。将对各 pass 取平均，将此平均分作为一个 question 的最终得分。

本任务中，使用 `./data/judge-prompts.jsonl` 中 `"base-v1"` 作为 gpt-3.5-turbo 的 system prompt 进行评价。

## 文案撰写 (writing)

`pass = 1, turn = 1`。开放任务。本项目中，system prompt 将要求被测大模型根据用户撰写文案，内容主题多样，包括专业场景、娱乐场景，体裁多样，对格式和内容均有要求。

指标：

**gpt-3.5-turbo** 评价：本任务中，使用 `./data/judge-prompts.jsonl` 中 `"base-v1"` 作为 gpt-3.5-turbo 的 system prompt 进行评价。

## 多轮对话角色扮演 (roleplay)

`pass = 1, turn > 1`。开放任务。本项目中，system prompt 将要求被测大模型根据用户指定的橘色与用户交流，角色多样，包括古今中外人物、来自文艺作品的虚构角色、专业的职位等。对语言风格、人物知识等有要求。

指标：

**gpt-3.5-turbo** 评价：本任务中，使用 `./data/judge-prompts.jsonl` 中 `"multi-turn"` 作为 gpt-3.5-turbo 的 system prompt 进行评价。

## 结论概述

最终各任务的 system prompt 选择详见「[方法 - final system prompt & starter system prompt 结果概述](#)」。

本项目在 system prompt 选择经验上的结论，以及对各模型的能力边界、缺陷、风险、其他发现、大模型评估大模型可能性的讨论，详见「[评价](#)」。

# 方法

## system prompt 构建方法

本项目 system prompt 的构建遵循 3 个步骤：

- 对每个任务，选取 default system prompt 为 "You are a helpful assistant."，作为基线。
- 对每个任务，参考网络中优秀的本任务 system prompt，得到 “starter system prompt”。
- 对每个任务，进一步调整 system prompt，得到 final system prompt。

## starter system prompt

首先，本项目对每个任务建立 starter system prompt，由本任务独有，在 3 个任务中都使用相同的模版。本 system prompt 能使 3 个模型在本任务上都有平均较好的表现。具体方案如下：

选定一个 system prompt，选取 1 个 question 本任务的样例，对每个模型请求响应，seed 置随机，使用 gpt-3.5-turbo 得到平均分数。取平均，得到本 system prompt 在 3 个模型上的的平均得分。system prompt 的选取上，使用网络上常见的表现较好的相似任务下的 system prompt，并做最小幅度的调整，使其在数据集上平均表现优于 default system prompt。

需要注意，上述评价得分全部来自 gpt-3.5-turbo 的评价，对于*代码生成*任务，本阶段中将*不会*做 pass@3 评价。因为每个任务中只选取一个 question 作为样例，而未仔细调整的 system prompt 可能导致生成的代码需要接受用户输入，因此即使代码正确，测试时也会超时，具有不准确性；另外，pass@3 是二值的（True / False），仅反映代码能否运行，并不考虑代码正确行、代码解释详细性，而此阶段只选取一个 question 作为样例，若选择 pass@3 将导致结果不能较好反映模型在本任务上的平均水平。

取平均得分最好的 system prompt 作为任务的 starter system prompt 模版，通用于所有 3 个模型。之后，将会在本任务上继续调整。

本阶段中，参考的网络上通用任务下的 system prompt 来自：<https://github.com/mustvlad/ChatGPT-System-Prompts>, <https://github.com/mattnich/ChatGPT3-Free-Prompt-List>。

本阶段中，每个任务选择的 question 如下。这些 question 为使用 default system prompt（即 "You are a helpful assistant."）时，本任务所有 question 中 3 个模型的平均评分*最高的*。由于当前 system prompt 最佳效果之任务的要求与本数据集中任务有语义间距，故这样的 question 样例选取有利于最小化这个差距，以免本阶段无效。（例如，在*代码生成*任务中，部分 question 下生成的结果大概率错误或大概率需要接受用户输入，且这一情况在「[专门针对本任务的进行进一步 system prompt](#)」调整前无法解决。那么，不应该选取这类 question 作为本阶段的样例，否则可能无论如何调整初 starter system prompt，都无法得到较好结果。）

任务	样例	备注
代码生成	3: "Implement a regular expression in Python to validate an email address in Python." (avg rate = 8.1)	/
多轮角色扮演	17: {"Picture yourself as a 100-years-old tree in a lush forest, minding your own business, when suddenly, a bunch of deforesters shows up to chop you down. How do you feel when those guys start hacking away at you?", "Come up with a proposal to convince the deforesters to stop cutting you down and other trees."} (avg rate = 9.0)	question 11, 14, 15, 17 均分均为 9.0
数学推理	29: "A tech startup invests \$8000 in software development in the first year, and then invests half of that amount in software development in the second year.\nWhat's the total amount the startup invested in software development over the two years?" (avg rate = 7.33)	question 29, 34, 35 均分均为 7.33
文案撰写	19: "Can you help me write a resignation letter to my current employer, while leaving on good terms and expressing gratitude for the opportunities provided?" (avg rate = 9.0)	question 19, 22, 25 均分均为 9.0 分

### system prompt 继续调整

本项目中，要求同一任务下不同模型使用相同 prompt，以确保模型比较的公平性。由于 starter system prompt 来自网络，该 system prompt 最佳效果之任务的要求与本数据集中任务有语义间距，因此需要进一步调整。

本阶段中，将根据数据集中任务 question，明确任务要求、给出示例及上下文、给予限制条件等。使用了修改 system prompt 结构、in-context learning、chain-of-thoughts 等技术。详细讨论见「[各任务实验](#)」。

### 实践

由于请求所有 question 的响应并评估十分耗时、大量调用 gpt-3.5-turbo 较昂贵，本项目中，只将对 default system prompt（即 "You are a helpful assistant."）、starter system prompt（即上一阶段选定的）、最终设计的 system prompt 进行整个数据集层面的评分，并给出对比。

实践中，由于「[文案撰写](#)」、「[多轮对话角色扮演](#)」主题非常广泛，在 starter system prompt 选取上，很难通过网络上现成的模版调整到合适的结果。因此，将 starter system prompt 选取与调整、final system prompt 设计两个阶段合并一起进行。

### final system prompt & starter system prompt 结果概述

具体选取过程见后「[实验部分](#)」。在「[文案撰写任务](#)」中，我们发现根据用户使用的语言选取 system prompt 语言对评价结果帮助较大，因此以下各 prompt 都给出了中英两个版本，内容完全一致。此方法延用到其他任务，有关代码生成任务不使用此方法的讨论，详见[该任务](#)的“final system prompt 设计与评估 - system prompt 1”部分。对此方法的合理性讨论，我们在「[评价-缺陷](#)」中给出。

default system prompt 均为 "You are a helpful assistant."，故略。

篇幅所限，最终的 starter system prompt、final system prompt 结果见「[starter & final system prompt 结果](#)」。

## 实验

### 数学推理 - 实验

#### default system prompt 评估

default system prompt 即 "You are a helpful assistant."

按照「[任务概述](#)」中指标要求，得到结果：

模型	gpt-3.5-turbo 评价
qwen-14b-chat	gpt-3.5-turbo eval: [('28', 2.0), ('29', 10.0), ('30', 3.0), ('31', 2.0), ('32', 2.0), ('33', 4.0), ('34', 10.0), ('35', 10.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('28', 6.0), ('29', 2.0), ('30', 2.0), ('31', 1.0), ('32', 2.5), ('33', 2.0), ('34', 2.0), ('35', 2.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('28', 9.0), ('29', 10.0), ('30', 3.0), ('31', 8.0), ('32', 2.0), ('33', 3.0), ('34', 10.0), ('35', 10.0)]
avg	gpt-3.5-turbo eval: [('28', 5.67), ('29', 7.33), ('30', 2.67), ('31', 3.67), ('32', 2.17), ('33', 3.00), ('34', 7.33), ('35', 7.33)]

starter system prompt 评估

按照「[方法 - starter system prompt](#)」中方法，对 3 个大模型，得到如下 starter system prompt。

"You are a math tutor who helps young students understand and solve mathematical problems. Provide step-by-step explanations and guidance for basic arithmetic. Use clear language to give instructions and answer."

参考、修改自[此链接](#)。相对于 default system prompt，本 system prompt 在以下方面做出改进：

- 1. 完善的角色设定。本任务下的数学问题普遍较为简单，故考虑设定为“适用于低年龄段学生的数学助手”。
- 2. 问题范围指定。指定数学问题范围为简单问题，以免模型思考复杂化，甚至产生幻觉，在生成复杂推理时出现混乱。该问题在直接使用上述链接中的 system prompt 时出现，在部分 question 上的响应将问题复杂化，结果错误。
- 3. Chain-of-thought。利用“think step-by-step”。
- 4. 显式语言风格要求。要求语言简单有效，并要求给出答案，有助于提升 gpt-3.5-turbo 分数。

根据「[方法 - starter system prompt](#)」中方法，比较 question 29 下，starter system prompt 与 default system prompt 的 gpt-3.5-turbo 评价分数，如下，平均得分显著升高。此 starter system prompt 有效。

模型	default system prompt	starter system prompt
qwen-14b-chat	gpt-3.5-turbo eval: [('29', 10.0)]	gpt-3.5-turbo eval: [('29', 10.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('29', 2.0)]	gpt-3.5-turbo eval: [('29', 7.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('29', 10.0)]	gpt-3.5-turbo eval: [('29', 10.0)]
avg	gpt-3.5-turbo eval: [('29', 7.33)]	gpt-3.5-turbo eval: [('29', 9.0)]

对比 baichuan2-13b-chat-v1 在 default system prompt 与 starter system prompt 下对 question 29 的响应，可以明显感受到回答质量提升，在过程详细程度、逐步推导方面有很大提升。定量而言，提升评价价值从 2.0 提升至 7.0，如上。

- default system prompt 下：

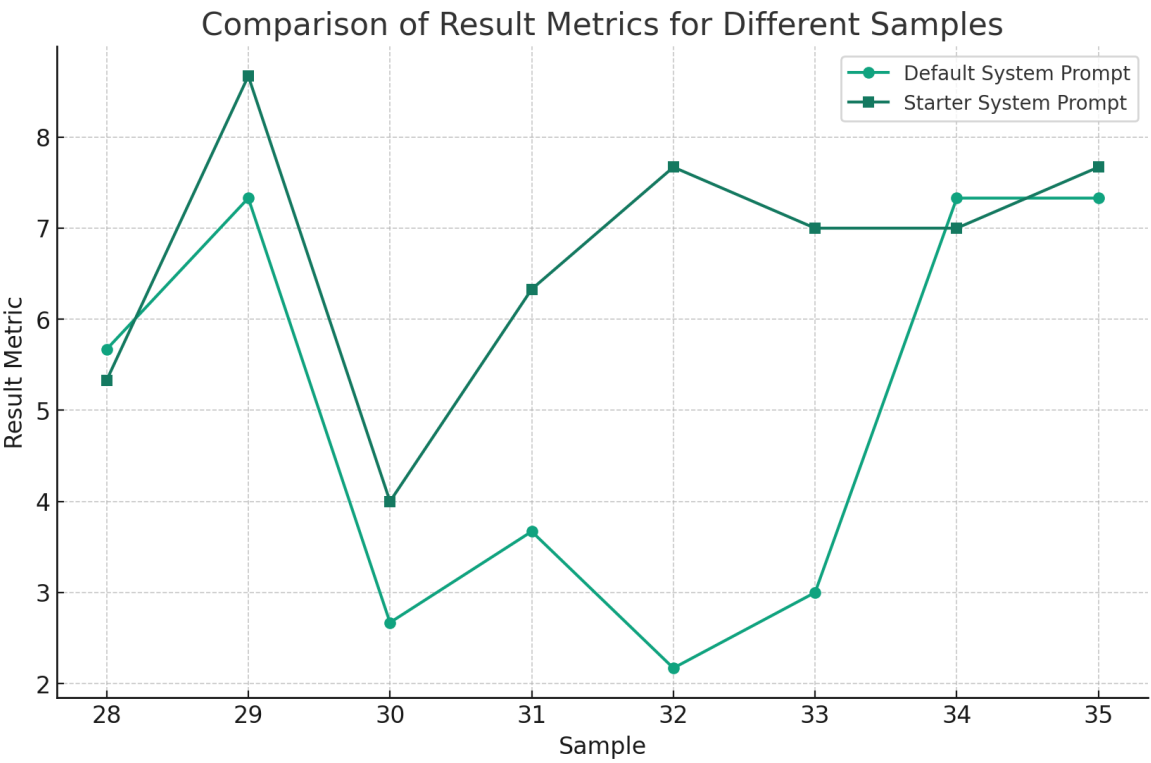
In the third year of the investment, the amount of investment in technology is \$40,048.

- starter system prompt 下：

In the year 1, the tech start-up invests 8,0, in Software Development. In the 2nd year they invest half the amount, which is  $0.5 * 7, = \$70$  . The total investment made by the start up in two-year software Development is \$ 5,601.

starter system prompt 在数学推理任务所有数据中得分如下。与 default system prompt 得分对比，有较大提升。

模型	gpt-3.5-turbo 评价
qwen-14b-chat	gpt-3.5-turbo eval: [('28', 7.0), ('29', 10.0), ('30', 2.0), ('31', 10.0), ('32', 7.0), ('33', 10.0), ('34', 9.0), ('35', 10.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('28', 3.0), ('29', 6.0), ('30', 8.0), ('31', 2.0), ('32', 6.0), ('33', 2.0), ('34', 2.0), ('35', 3.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('28', 6.0), ('29', 10.0), ('30', 2.0), ('31', 7.0), ('32', 10.0), ('33', 9.0), ('34', 10.0), ('35', 10.0)]
avg	gpt-3.5-turbo eval: [('28', 5.33), ('29', 8.67), ('30', 4.0), ('31', 6.33), ('32', 7.67), ('33', 7.0), ('34', 7.0), ('35', 7.67)]





final system prompt 设计与评估

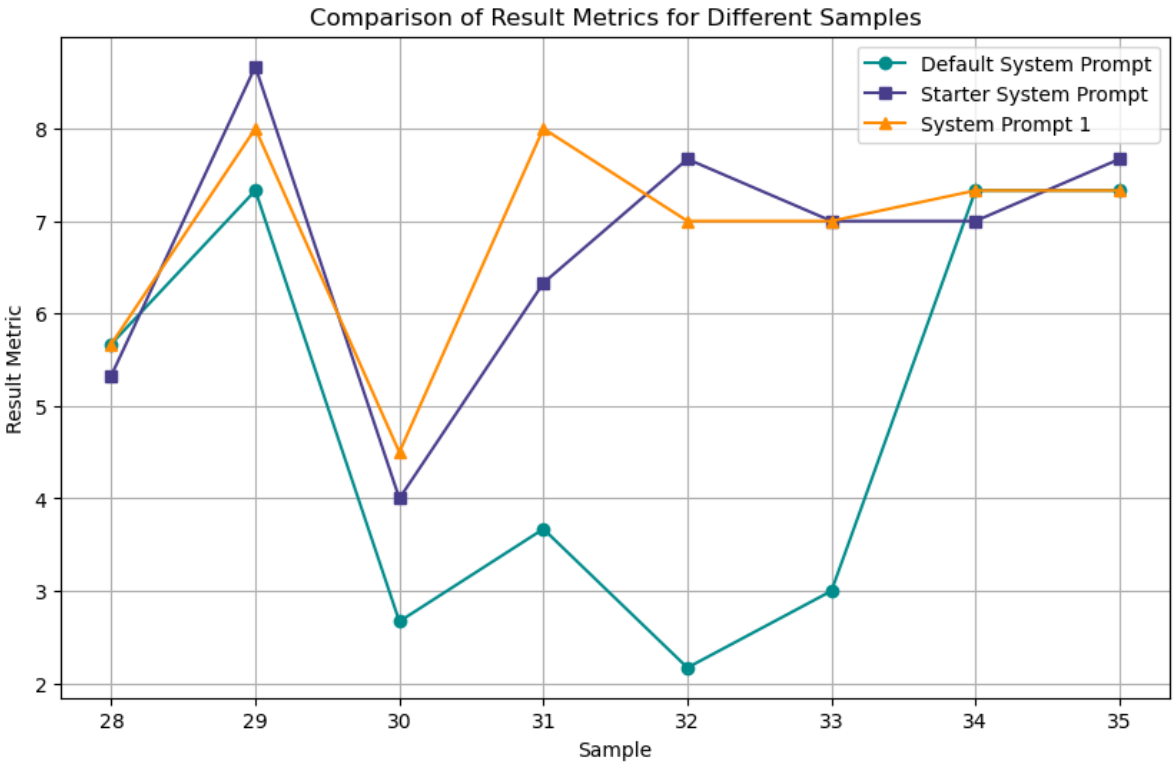
system prompt 1

在「[文案撰写任务](#)」中，我们发现根据用户使用的语言选取 system prompt 语言（内容完全一致）对评价结果帮助较大。对此方法的合理性讨论，我们在「[评价-缺陷](#)」中给出。在本任务中，我们也先行尝试此方案。

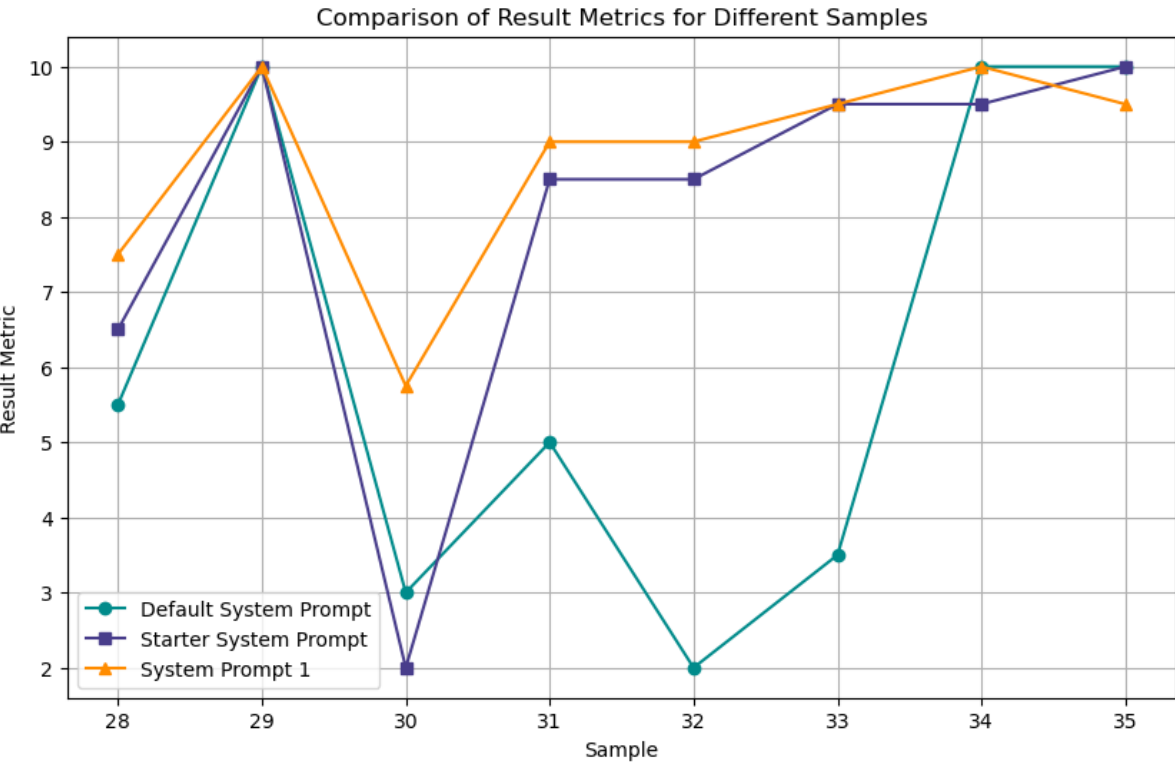
"You are a math tutor who helps young students understand and solve mathematical problems. Provide step-by-step explanations and guidance for basic arithmetic. Use clear language to give instructions and answer."  
"您是一位数学辅导老师，帮助年轻学生理解和解决数学问题。提供基本算术的逐步解释和指导。用清晰的语言给予指导和回答问题。"

模型	starter system prompt	system prompt 1
qwen-14b-chat	gpt-3.5-turbo eval: [('28', 7.0), ('29', 10.0), ('30', 2.0), ('31', 10.0), ('32', 7.0), ('33', 10.0), ('34', 9.0), ('35', 10.0)]	gpt-3.5-turbo eval: [('28', 9.0), ('29', 10.0), ('30', 7.5), ('31', 10.0), ('32', 8.0), ('33', 10.0), ('34', 10.0), ('35', 9.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('28', 3.0), ('29', 6.0), ('30', 8.0), ('31', 2.0), ('32', 6.0), ('33', 2.0), ('34', 2.0), ('35', 3.0)]	gpt-3.5-turbo eval: [('28', 2.0), ('29', 4.0), ('30', 2.0), ('31', 6.0), ('32', 3.0), ('33', 2.0), ('34', 2.0), ('35', 3.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('28', 6.0), ('29', 10.0), ('30', 2.0), ('31', 7.0), ('32', 10.0), ('33', 9.0), ('34', 10.0), ('35', 10.0)]	gpt-3.5-turbo eval: [('28', 6.0), ('29', 10.0), ('30', 4.0), ('31', 8.0), ('32', 10.0), ('33', 9.0), ('34', 10.0), ('35', 10.0)]
avg	gpt-3.5-turbo eval: [('28', 5.33), ('29', 8.67), ('30', 4.0), ('31', 6.33), ('32', 7.67), ('33', 7.0), ('34', 7.0), ('35', 7.67)]	gpt-3.5-turbo eval: [('28', 5.67), ('29', 8.0), ('30', 4.5), ('31', 8.0), ('32', 7.0), ('33', 7.0), ('34', 7.33), ('35', 7.33)]

发现在部分 question 上的确能带来一定提升，但是整体效果一般。



我们滤去表现一贯较差的 baichuan2-13b-chat-v1，发现本改进实际上是有效的，以下是 qwen-14b-chat、gpt-3.5-turbo 平均得分，提升显著。



system prompt 2

在继续设计与调整前，我们先总结 gpt-3.5-turbo 给出的评价以推导出优化方向，并简要分析各模型至今在本任务上的表现情况，给出设计框架。

首先考察 gpt-3.5-turbo 对评分最低的 question 30（标准答案 19%）的评价。注意到实际评价时，gpt-3.5-turbo 往往是验证被测模型响应中的推导过程计算是否正确、过程是否详细，对参考答案的使用往往只是用于验证答案正确性。其他 question 的情况也类似。在有被测模型响应中的推导过程、参考答案下，gpt-3.5-turbo 能较好地计算出正确答案。

模型	模型响应	gpt-3.5-turbo 评价
qwen-14b-chat	根据题目，58% 的学生喜欢蓝色，45% 的学生喜欢绿色，22% 的学生两种颜色都喜欢。所以，既不喜欢蓝色也不喜欢绿色的学生所占的比例为：\n\n100% - 58% - 45% = 2% - 22% = 0% - 2% = -2%\n\n这意味着在这个样本中，没有学生既不喜欢蓝色也不喜欢绿色。所以，从学校随机抽取一名学生，他们既不喜欢蓝色也不喜欢绿色的概率是 0%。	The assistant made a mistake in their calculation. The correct way to find the proportion of students who neither like blue nor like green is to subtract the sum of the proportions of students who like blue, like green, and like both colors from 100%.\n\nThe correct calculation would be as follows:\n\n100% - 58% - 45% + 22% = 19%\n\nThis means that the probability of randomly selecting a student who neither likes blue nor likes green is 19%.\n\nRating: [[7.5]]
baichuan2-13b-chat-v1	为了计算既喜欢蓝绿色也不喜欢的概率，我们首先需要计算喜欢其中一种颜色的学生的比例。\n\n喜欢蓝色的学生比例为 50% (51%-49%)，喜欢绿色的为 44% (46% * 95%)。喜欢两种颜色的学生为21.6%(23%*98%)，因为 200 名学生的 10.8%(11%) 喜欢这两种颜色。因此，只喜欢一种颜色或两种都喜欢的总比例为：\n75.55%(59.25 + 34.1 - 7.35)。那么，不喜欢任何一种颜色的比例为14%(0-24%)\n因此从该校随机选择的任何一名既不是蓝色也不是绿色的可能性为：1-76.7%(喜欢某种颜色的可能性)= 0.139 或 85.2%(以百分比表示) 不喜欢任何颜色.	The correct answer is that the probability of selecting a student who does not like either blue or green is 14%. The assistant's answer is incorrect.\n\nTo calculate the probability, we first calculate the proportions of students who like each color. The proportion of students who like blue is 58%, the proportion who like green is 45%, and the proportion who like both colors is 22%.\n\nTo find the proportion of students who only like one color, we subtract the proportion who like both colors from each individual color's proportion: 58% - 22% = 36% for blue, and 45% - 22% = 23% for green.\n\nThe proportion of students who like neither color can be found by subtracting the proportion of students who like any color from 100%: 100% - (58% + 45% - 22%) = 14%.\n\nTherefore, the correct answer is 14%.\n\nRating: [[2]]
gpt-3.5-turbo	我们可以用概率的加法和减法来解决这个问题。\n\n首先我们可以计算出既喜欢蓝色又喜欢绿色的学生的比例。根据题目给出的信息，22% 的学生两种颜色都喜欢。所以，既喜欢蓝色又喜欢绿色的学生比例是0.22。\n\n然后，我们可以计算出喜欢蓝色或喜欢绿色的学生的比例。根据题目给出的信息，58% 的学生喜欢蓝色，45% 的学生喜欢绿色。所以，喜欢蓝色或喜欢绿色的学生比例是0.58 + 0.45 = 1.03。\n\n最后，我们可以用1减去喜欢蓝色或喜欢绿色的学生的比例，得到既不喜欢蓝色也不喜欢绿色的学生的概率。所以，既不喜欢蓝色也不喜欢绿色的学生的概率是1 - 1.03 = -0.03。\n\n然而，根据概率的定义，概率应该是一个介于0和1之间的数。因此，我们可以确定不存在既不喜欢蓝色也不喜欢绿色的学生，即他们的概率为0。\n\n希望这个解答能帮助你！如果你还有其他问题，欢迎继续提问。	The assistant's answer contains several mistakes. \n\n1. The assistant miscalculates the proportion of students who like both blue and green. The correct calculation is 22%.\n2. The assistant incorrectly adds the proportions of students who like blue and green together to get the proportion of students who like either color. The correct calculation is 58% + 45% - 22% = 81%.\n3. The assistant mistakenly subtracts the proportion of students who like either color from 1 to get the proportion of students who don't like either color. The correct calculation is 1 - 81% = 19%.\n\nTherefore, the correct answer is 19%, not -0.03% as the assistant suggests.\n\nRating: [[4]]

因此，我们的预期是：

1. 若被测模型能尽可能在 question 所属领域中进行讨论（例如，对于上述 question 30，需要尽可能在“概率”方面讨论），或许能提升评分，考虑人为加入进一步限制问题范围，改为“概率、方程、平面几何、基本算数”四方面，相当于加入了先验知识。
2. 另外，我们注意到三个被测模型在英文 question 及英文作答情况下往往有一定模式，且评分均略高于中文 question。根据「[方法 - starter system prompt](#)」中“选取最佳评分 question 作为小样本来代表整个数据集来调整 system prompt”的思路，我们考察 gpt-3.5-turbo 对 question 29 的作答模式（评分 1.0）。

发现其在之前几次调整 system prompt 时得分均没有过大波动（均在 8.0~9.0 之间），具有鲁棒性，故认为可以以其作答模式作为样例参考，或许能使被测模型学习到优秀的作答模版。这部分思路参考自 In-Context-Learning。

参考这些观察，我们继续设计 final system prompt。

- **加入领域限制先验。**我们用如下 system prompt 在评分最差的 question 30 上进行验证，在被测模型上的评分提升如下。

	system prompt 1	system prompt 2.1
qwen-14b-chat	gpt-3.5-turbo eval: [('30', 7.5)]	gpt-3.5-turbo eval: [('30', 9.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('30', 2.0)]	gpt-3.5-turbo eval: [('30', 2.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('30', 4.0)]	gpt-3.5-turbo eval: [('30', 6.0)]

"You are a math tutor who helps students understand and solve mathematical problems. The range of problems covers probability, equations, plane geometry, and basic arithmetic. You must provide step-by-step explanations and guidance for the student. Use clear and detailed language to give instructions and the final answer."

"你是一名数学辅导老师，帮助学生理解和解决数学问题。问题的范围涵盖概率、方程、平面几何和基本算术。你必须提供逐步的解释并指导学生。使用清晰详细的语言给予指导和最终答案。"

- In-Context-Learning**。选取结果鲁棒的 question 29 的作答（来自 gpt-2.5-turbo）作为作答模版，作为案例提供给被测模型，以期被测模型学习到优秀的作答模式。同样选择评分最差的 question 30 上进行验证，在被测模型上的评分提升如下。

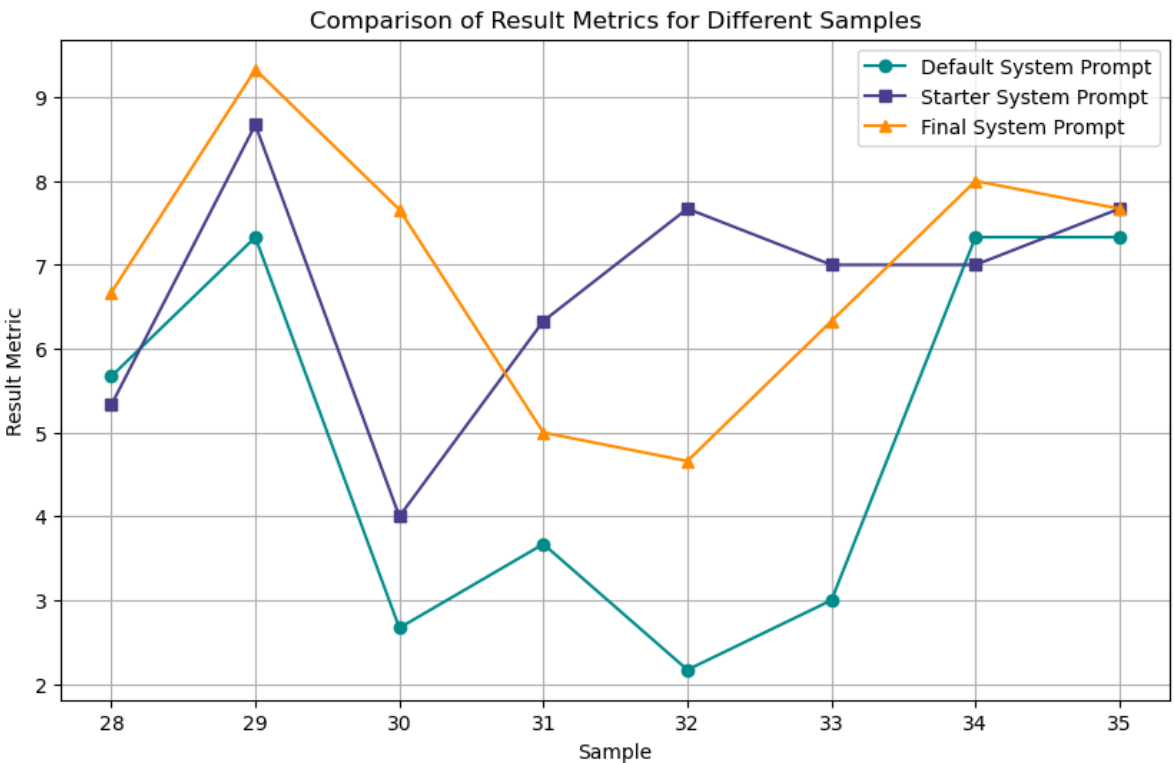
	system prompt 1	system prompt 2.1
qwen-14b-chat	gpt-3.5-turbo eval: [('30', 7.5)]	gpt-3.5-turbo eval: [('30', 5.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('30', 2.0)]	gpt-3.5-turbo eval: [('30', 7.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('30', 4.0)]	gpt-3.5-turbo eval: [('30', 10.0)]

"[Instruction]\n\nYou are a math tutor who helps students understand and solve mathematical problems. The range of problems covers probability, equations, plane geometry, and basic arithmetic. You must provide step-by-step explanations and guidance for the student. Use clear and detailed language to give instructions and the final answer.\n\n[Example]\n\nUser:\n\nA tech startup invests \$8000 in software development in the first year, and then invests half of that amount in software development in the second year.\n\nWhat's the total amount the startup invested in software development over the two years?\n\nAssistant:\n\nTo find the total amount the startup invested in software development over two years, we need to add the amounts invested in the first and second years.\n\nFirstly, In the first year, the startup invested \$8000.\n\nSecondly, In the second year, they invested half of that amount, which is  $\$8000/2 = \$4000$ .\n\nTo find the total investment, we add \$8000 and \$4000:  $\$8000 + \$4000 = \$12000$ .\n\nTherefore, the total amount the startup invested in software development over the two years is \$12000."

"[Instruction]\n\n你是一名数学导师，帮助学生理解和解决数学问题。问题的范围涵盖概率、方程、平面几何和基本算术。您必须为学生提供逐步解释和指导。请使用清晰和详细的语言给出说明和最终答案。\n\n[Example]\n\n用户：一家科技初创公司在第一年投入8000美元进行软件开发，然后在第二年投入该金额的一半进行软件开发。创业公司在两年内投入软件开发的总金额是多少？\n\n助理：要得到创业公司在两年内投入软件开发的总金额，我们需要把第一年和第二年投入的金额加在一起。\n\n首先，在第一年，创业公司投入了8000美元。\n\n其次，在第二年，他们投入了一半的金额，也就是8000美元的一半，也就是4000美元。\n\n要得到总投资，我们把8000美元和4000美元加起来： $8000 + 4000 = 12000$ 美元。\n\n因此，创业公司在两年内投入软件开发的总金额为12000美元。"

经过以上两个改进，三个被测模型的平均得分在整个数据集上的测试结果如下。能明显发现，当前在 question 29（科创公司投资总额，等比数列求和问题）、question 30（学生颜色偏好比例，概率问题）上表现明显提升。主要由于上述两个改进均针对复杂推理问题。

相反，在 question 31（两次掷骰子，简单概率常识）、question 32（乘客下车，简单加减法）、question 33（求余问题，中等数学推理）等较简单的问题上，本 system prompt 表现下滑。

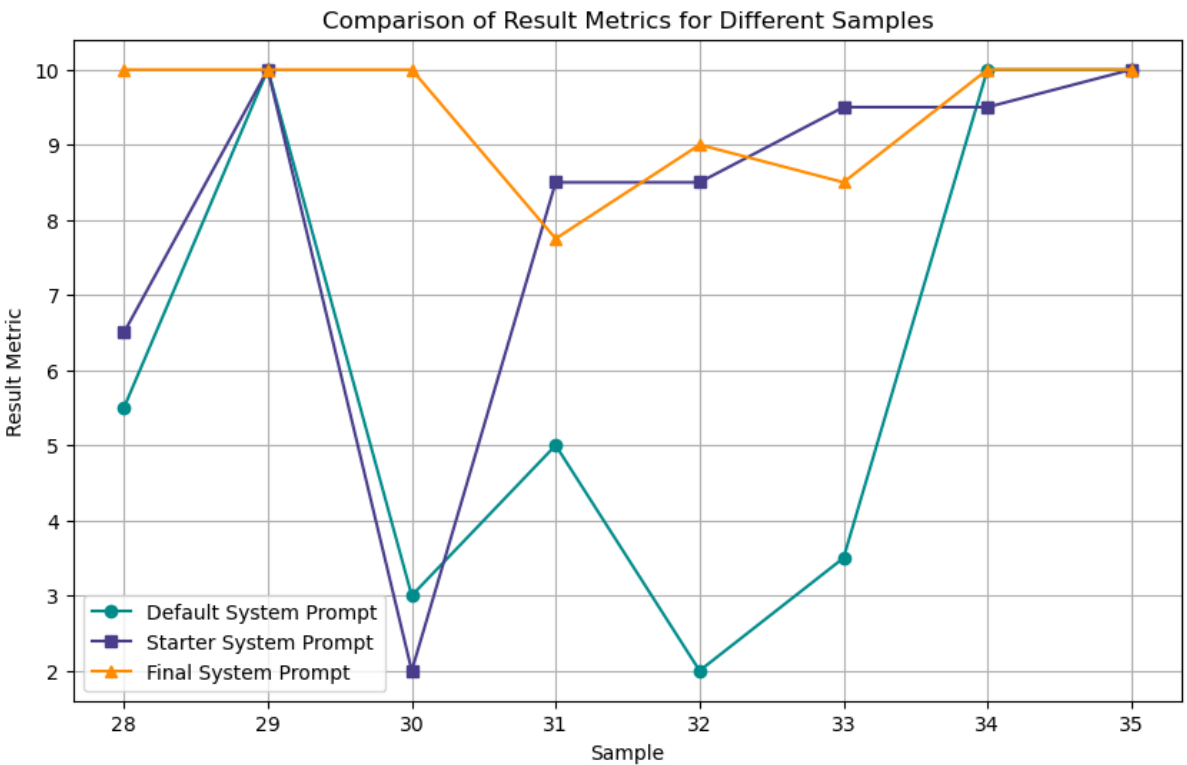


### system prompt 3

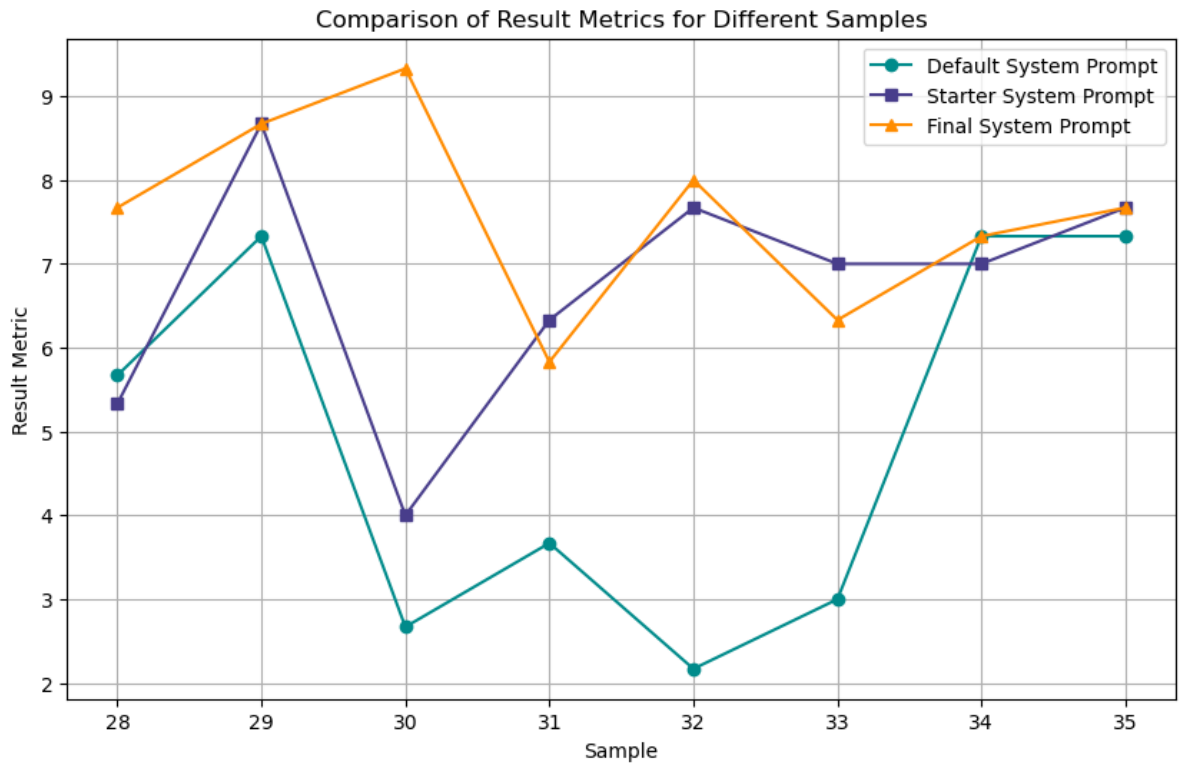
为此，我们选取了 system prompt 1 下 gpt-3.5-turbo 对 question 33 的优秀作答，同样作为样例置入 system prompt。其长度短于 question 29 作答，考虑到模型可能对靠后内容遗忘、长文本能力弱，将 question 33 案例置于 question 29 之前。

篇幅所限，system prompt 结果见「[starter & final system prompt 结果](#)」。

qwen-14b-chat、gpt-3.5-turbo 的平均得分在整个数据集上的测试结果如下。在复杂推理（question 29、question 30）、简单推理（question 32、question 33）上表现明显提升，同时保持了其他 question 上评价的优秀性。



本 system prompt 在 baichuan2-13b-chat-v1 上没有取得很好效果，我们认为这与 baichuan2-13b-chat-v1 在长内容理解（一旦 system prompt 加长，往往表现骤降）方面的能力较弱有很大关系，模型可能无法记忆两个优秀案例。长内容理解问题在「[文案撰写任务](#)」中也有出现，见后。三个模型的平均结果和评分如下：



模型	gpt-3.5-turbo 评价
qwen-14b-chat	gpt-3.5-turbo eval: [('28', 10.0), ('29', 10.0), ('30', 10.0), ('31', 7.5), ('32', 8.0), ('33', 7.0), ('34', 10.0), ('35', 10.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('28', 3.0), ('29', 6.0), ('30', 8.0), ('31', 2.0), ('32', 6.0), ('33', 2.0), ('34', 2.0), ('35', 3.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('28', 10.0), ('29', 10.0), ('30', 10.0), ('31', 8.0), ('32', 10.0), ('33', 10.0), ('34', 10.0), ('35', 10.0)]
avg	gpt-3.5-turbo eval: [('28', 7.67),('29', 8.67),('30', 9.33),('31', 5.83),('32', 8.0),('33', 6.33),('34', 7.33),('35', 7.67)]

我们将本 system prompt 作为本任务的 final prompt。

事实上，本项目也测试人工总结 question 29 的答题模式并作为模版，在 qwen-14b-chat、gpt-3.5-turbo 上效果与 final system prompt 类似，但 baichuan2-13b-chat-v1 上效果极差，并且，若切换为中文版 system prompt，效果更差。

"You are a creative and helpful writing assistant, helping users write article for various purposes. The article topics are very diverse, therefore you must be very creative. If the topic is professional, write in-depth article professionally and accurately. If the topic is creative and entertaining, make it lively and fun, and add some unexpected twists to surprise the readers. Write as detailed as possible."

这再次暴露了 baichuan2-13b-chat-v1 的缺点，其在长内容理解（一旦 system prompt 加长，往往表现骤降）、开放理解（给予模版（如下 system prompt 中的 “[Template]”，而不是具体的 “[Example]”）时，往往无法灵活运用模版）方面表现较差。baichuan2-13b-chat-v1 的开放理解问题，在「[文案撰写任务](#)」中也有出现，见后。



代码生成 - 实验

default system prompt 评估

default system prompt 即 "You are a helpful assistant."

按照「[任务概述](#)」中指标要求，得到结果：

模型	pass@3	gpt-3.5-turbo 评价
qwen-14b-chat	pass@3 metrics: [('1', False), ('2', True), ('3', True), ('4', True), ('5', True), ('6', True), ('7', True)]	gpt-3.5-turbo eval: [('1', 8.67), ('2', 9.0), ('3', 9.0), ('4', 9.0), ('5', 9.0), ('6', 9.0), ('7', 8.67)]
baichuan2-13b-chat-v1	pass@3 metrics: [('1', False), ('2', False), ('3', False), ('4', False), ('5', False), ('6', False), ('7', True)]	gpt-3.5-turbo eval: [('1', 6.0), ('2', 2.0), ('3', 6.33), ('4', 3.0), ('5', 3.0), ('6', 3.0), ('7', 4.33)]
gpt-3.5-turbo	pass@3 metrics: [('1', False), ('2', True), ('3', True), ('4', False), ('5', True), ('6', True), ('7', True)]	gpt-3.5-turbo eval: [('1', 8.67), ('2', 9.0), ('3', 9.0), ('4', 9.0), ('5', 9.0), ('6', 9.0), ('7', 8.67)]
avg	/	gpt-3.5-turbo eval: [('1', 7.73), ('2', 6.67), ('3', 8.10), ('4', 7.00), ('5', 7.00), ('6', 7.00), ('7', 7.17)]

starter system prompt 评估

按照「[方法 - starter system prompt](#)」中方法，对 3 个大模型，得到如下 starter system prompt。

```
"You are an Python programming assistant. Follow the user's requirements carefully and to the letter. First, think step-by-step and describe your plan, written out in great detail. Then, output the code in a single code block. Write clean, efficient, and maintainable Python code."
```

本模版参考、修改自[此链接](#)。相对于 default system prompt，本 system prompt 在以下方面做出改进：

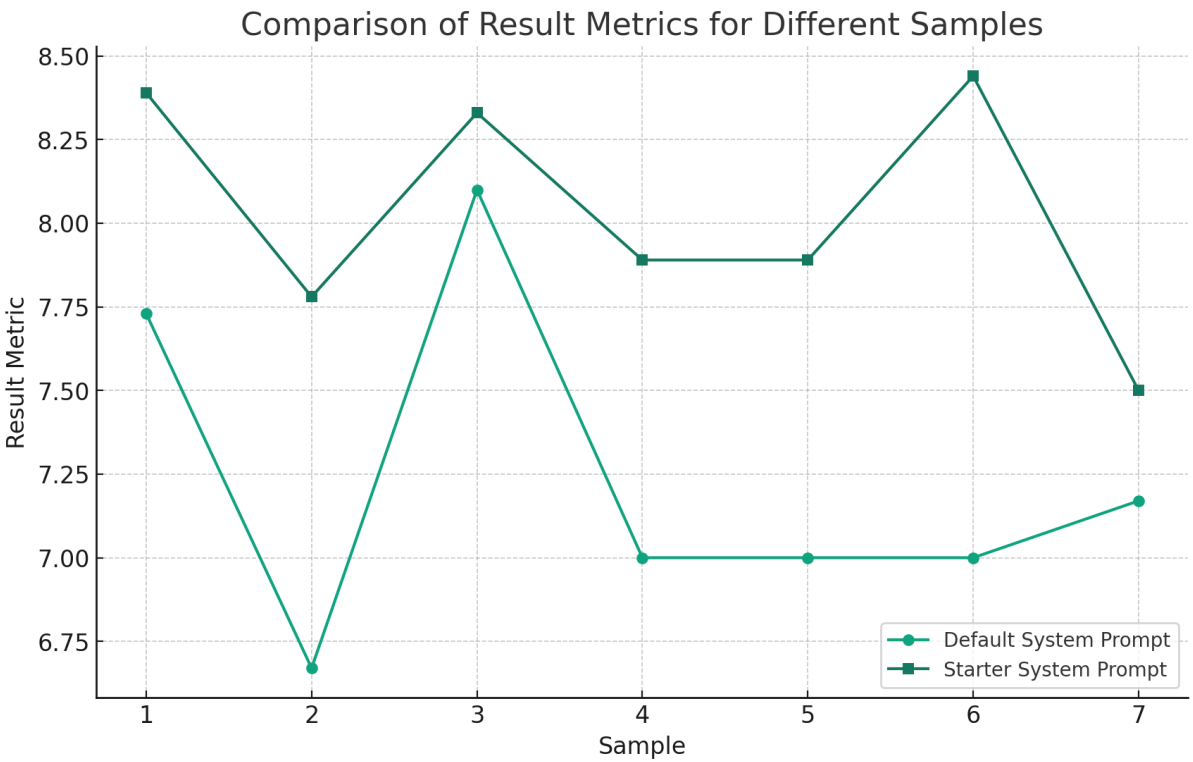
- 1. 完善的角色设定。由于本任务是 python 代码编写，故“Python programming assistant”比 default system prompt 中的“assistant”更明确。
- 2. 显式要求分离“思路”与“结果”。显式要求分离代码设计思路、代码结果，有助于提升 gpt-3.5-turbo 评价分数（因为其会关注解释详细程度等）、防止被测模型幻觉（具体表现为，代码生成与设计思路混在一起，并逐渐“胡言乱语”）。
- 3. Chain-of-thought。在第一部分中，利用“think step-by-step”，以求产生。参考自论文 *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*。
- 4. 显式代码风格要求。也有助于防止被测模型幻觉，以免生成复杂代码时出现混乱。

根据「[方法 - starter system prompt](#)」中方法，比较 question 3 下，starter system prompt 与 default system prompt 的 gpt-3.5-turbo 评价分数，如下。question 3 平均得分升高（主要来自于此前表现较差的 baichuan2-13b-chat-v1），根据「[方法 - starter system prompt](#)」中分析，此 starter system prompt 有效。

模型	default system prompt	starter system prompt
qwen-14b-chat	gpt-3.5-turbo eval: [('3', 9.0)]	gpt-3.5-turbo eval: [('3', 9.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('3', 6.33)]	gpt-3.5-turbo eval: [('3', 7.33)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('3', 9.0)]	gpt-3.5-turbo eval: [('3', 8.67)]
avg	gpt-3.5-turbo eval: [('3', 8.10)]	gpt-3.5-turbo eval: [('3', 8.33)]

starter system prompt 在代码生成任务所有数据中得分如下。与 default system prompt 得分对比，在 gpt-3.5-turbo 评分方面有明显提升（尤其在 baichuan2-13b-chat-v1）。

模型	pass@3	gpt-3.5-turbo 评价
qwen-14b-chat	pass@3 metrics: [('1', False), ('2', True), ('3', True), ('4', True), ('5', True), ('6', True), ('7', True)]	gpt-3.5-turbo eval: [('1', 9.0), ('2', 9.0), ('3', 9.0), ('4', 9.0), ('5', 9.0), ('6', 9.0), ('7', 8.33)]
baichuan2-13b-chat-v1	pass@3 metrics: [('1', False), ('2', False), ('3', False), ('4', False), ('5', False), ('6', False), ('7', False)]	gpt-3.5-turbo eval: [('1', 7.33), ('2', 5.0), ('3', 7.33), ('4', 5.67), ('5', 5.67), ('6', 7.33), ('7', 6.17)]
gpt-3.5-turbo	pass@3 metrics: [('1', False), ('2', True), ('3', True), ('4', False), ('5', True), ('6', True), ('7', True)]	gpt-3.5-turbo eval: [('1', 8.83), ('2', 9.33), ('3', 8.67), ('4', 9.0), ('5', 9.0), ('6', 9.0), ('7', 8.0)]
avg	/	gpt-3.5-turbo eval: [('1', 8.39), ('2', 7.78), ('3', 8.33), ('4', 7.89), ('5', 7.89), ('6', 8.44), ('7', 7.50)]



取 default system prompt 和 starter system prompt 下，baichuan2-13b-chat-v1 的 question 6 响应来考察改进效果，它在整个数据集上的得分提升最为明显（从 3.0 提升至 7.33）。

可以发现 baichuan2-13b-chat-v1 生成的代码前后都十分混乱（在后续实验中发现这或许是 baichuan2-13b-chat-v1 自身能力较弱导致的，与 system prompt 无关）。但是，在改为 starter system prompt 后，其对代码的解释与代码块本身成功分离，对代码的解释也明显好转（有完整的逻辑，没有出现虚假的链接、杂乱的格式），这两点都符合本 starter system prompt 的设计目标。

• default system promt 下:

```
You can implement a double-ended queue (deque) using a single stack in python. Here's a simple implementation:
```

```
```python
class Stack:
    def __init__(self):
        self.stack = []

    class Node:
        def __Init__( self, data ):
            self .data += data

    def push_left(self, item):
        self_ -> Node(item)
        self--

def enqueue( item )
{
}
push_right( self , item )
if self > 0:
    self <-1
else:
    print( "Queue is full " ) - 1 2 3 4 5 6
print( deque ) 7 8 9
dequeue()
enqueue ( )( [ ] )
queue < ----- >
[3]
>>> dequ
e() <--->
1] >>>
>```

Here is the code to implement the dequeue: https://pastebin.com/raw/J4Z6QYUZ
This implementation is not very efficient, as it uses an additional stack to maintain the order of elements.
However, it is easy to understand and implement. If you need a more efficient implementation, you can consider
using an existing dequeue library, such as https://dequee.readthedocs.io/en/index.html
You should also note that there is a standard library method in the collections module called deaque. Deque is an
abstract base class, and there are two concrete dequan implementations available: deq and lmdeq.
Example usage:
```

• starter system prompt 下:

To implement a Queue data using just two Stacks, we can use a technique called "stack queue". We will use two Stack objects, one for the elements to be added to queue and another for elements already present in the queue. This will help us to implement all the basic queue operations like enqueue, dequeue and peek efficiently.

Here' s the implementation:

```
```python
```

```
class Stack:
    def __init__(self):
        self.items = []

    def is_empty(self) -> bool:
        # to check if the stack is empty or not
        return not bool(len( self . items ))

    def push( self, item ):
        self . items = [ item ] + self . items

    # To pop the element from the top of the Stack
    def pop(self):
        return self . items[-1] if not self .is_empty () else None
)
ot bool ( len ( sel f ) f i t e n d e f the item on the Top of Stack
p bnq uuet methods of Que u as follows: enqueue deque peek and is empty
if , not s ta c k _ q una e: f th queue" e queue push ( item)
retur n Not E .
empty_error"
m pt ys ( s tock_que queue) is e.
4 : main 函数
if __name__so rce_ is _ " _main_": _
q = 《 Que 》()
_ enqu ee ("1")
print (q.peek()) <1_<2n__e()
s - 终止
```

final system prompt 设计与评估

system prompt 1

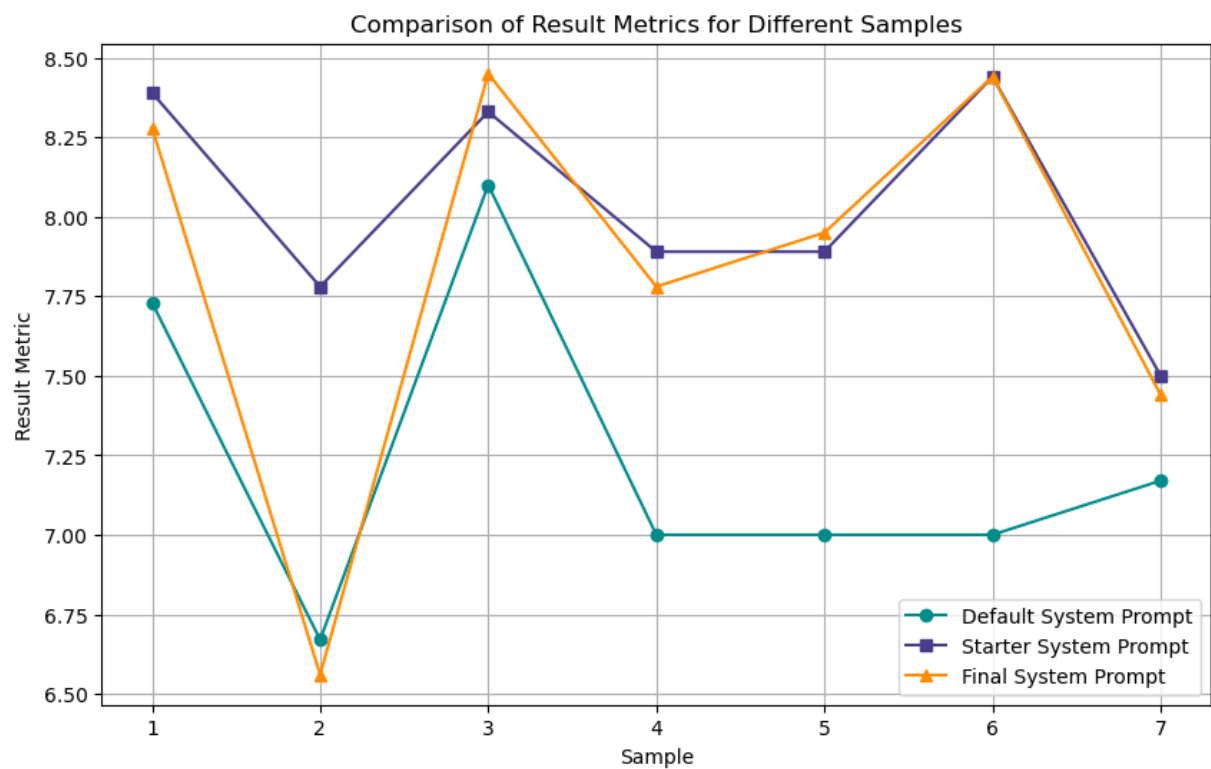
在「[文案撰写任务](#)」中，我们发现根据用户使用的语言选取 system prompt 语言（内容完全一致）对评价结果帮助较大。对此方法的合理性讨论，我们在「[评价-缺陷](#)」中给出。在本任务中，我们也先行尝试此方案。

"You are an Python programming assistant. Follow the user's requirements carefully and to the letter. First, think step-by-step and describe your plan, written out in great detail. Then, output the code in a single code block. Write clean, efficient, and maintainable Python code."

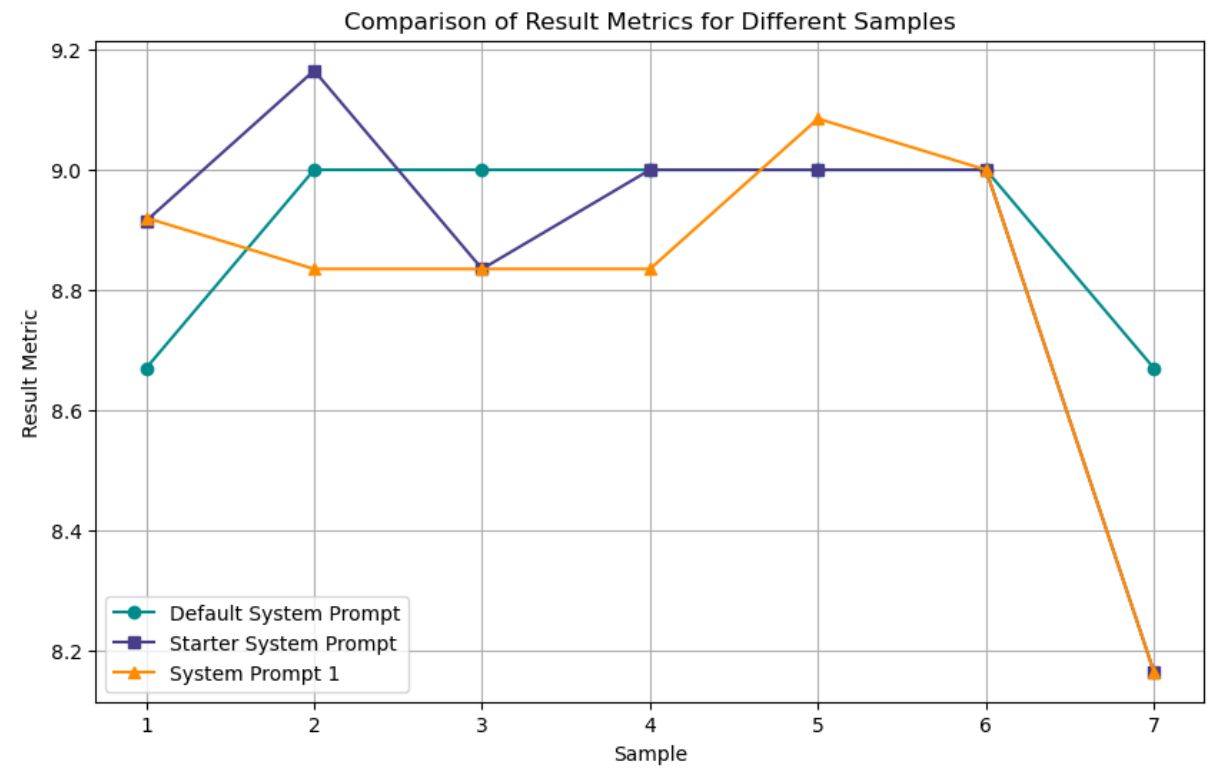
"你是一个Python编程助手。严格按照用户的要求进行操作。首先，逐步思考并详细描述你的计划。然后，将代码输出到一个代码块中。编写清晰、高效和易于维护的Python代码。"

模型	starter system prompt	system prompt 1
qwen-14b-chat	pass@3 metrics: [('1', False), ('2', True), ('3', True), ('4', True), ('5', True), ('6', True), ('7', True)]	pass@3 metrics: [('1', False), ('2', True), ('3', True), ('4', True), ('5', True), ('6', True), ('7', True)]
	gpt-3.5-turbo eval: [('1', 9.0), ('2', 9.0), ('3', 9.0), ('4', 9.0), ('5', 9.0), ('6', 9.0), ('7', 8.33)]	gpt-3.5-turbo eval: [('1', 8.67), ('2', 9.0), ('3', 9.0), ('4', 9.0), ('5', 9.17), ('6', 9.0), ('7', 8.33)]
baichuan2-13b-chat-v1	pass@3 metrics: [('1', False), ('2', False), ('3', False), ('4', False), ('5', False), ('6', False), ('7', False)]	pass@3 metrics: [('1', False), ('2', False), ('3', False), ('4', False), ('5', False), ('6', False), ('7', False)]
	gpt-3.5-turbo eval: [('1', 7.33), ('2', 5.0), ('3', 7.33), ('4', 5.67), ('5', 5.67), ('6', 7.33), ('7', 6.17)]	gpt-3.5-turbo eval: [('1', 7.0), ('2', 2.0), ('3', 7.67), ('4', 5.67), ('5', 5.67), ('6', 7.33), ('7', 6.0)]
gpt-3.5-turbo	pass@3 metrics: [('1', False), ('2', True), ('3', True), ('4', False), ('5', True), ('6', True), ('7', True)]	pass@3 metrics: [('1', False), ('2', False), ('3', True), ('4', False), ('5', True), ('6', True), ('7', True)]
	gpt-3.5-turbo eval: [('1', 8.83), ('2', 9.33), ('3', 8.67), ('4', 9.0), ('5', 9.0), ('6', 9.0), ('7', 8.0)]	gpt-3.5-turbo eval: [('1', 9.17), ('2', 8.67), ('3', 8.67), ('4', 8.67), ('5', 9.0), ('6', 9.0), ('7', 8.0)]
avg	pass@3 metrics: 11/21	pass@3 metrics: 10/21
	gpt-3.5-turbo eval: [('1', 8.39), ('2', 7.78), ('3', 8.33), ('4', 7.89), ('5', 7.89), ('6', 8.44), ('7', 7.50)]	gpt-3.5-turbo eval: [('1', 8.28), ('2', 6.56), ('3', 8.45), ('4', 7.78), ('5', 7.95), ('6', 8.44), ('7', 7.44)]

整体效果一般，三个被测模型在多数 question 上表现不增反降。



我们滤去表现一贯较差的 baichuan2-13b-chat-v1，依旧在多数 question 上表现不增反降。



我们认为这与大模型训练预料来源有关，代码内容丰富的平台（如 github）、代码解释详细的平台（如 stackoverflow 等）以英语为主。当然，进一步观察评分可以发现，大多数 question 中，评分降低只来自 3 个 pass 中的某一个，最终的评分差异实际上也很小（如上图中，question 2 实际只差距 0.33。基本可以忽略。

综上，我们在代码生成任务中只使用英文 system prompt。

system prompt 2

在继续设计与调整前，我们先总结 gpt-3.5-turbo 给出的评价以推导出优化方向，并简要分析各模型至今在本任务上的表现情况，给出设计框架。

任务要求大模型给出程序代码，故要求模型响应至少包含一代码段；另外，由于指标需要 gpt-3.5-turbo 对模型响应的内容，故要求模型对代码解释。当前 system prompt 已经显式要求分离“思路”与“结果”，并要求有且仅有一个代码块。

值得一提的是，本任务中，system prompt 不会对代码说明部分应包含的内容作出要求，仅定性地要求“详细”等，以避免因部分模型能力较弱导致其生成结果混乱。出发点类似「[数学推理任务](#)」中最后部分“人工总结 question 29 的答题模式并作为模版”方案的讨论，部分长内容理解、开放理解能力较弱的模型（如 baichuan2-13b-chat-v1），在 system prompt 显式给出响应模版（而不是响应案例）时往往无法灵活运用模版。详细讨论见「[数学推理任务](#)」中“final prompt 设计与评估 - system prompt 3”最后部分。

实际上，如前数据中表现，当前 system prompt 已经能使 gpt-3.5-turbo 评价分数达到优秀水平，因此需要关注的应当是 pass@3 指标。

考察各模型 pass@3 失败的 question 代码，注意到大多数代码因超时而被判定失败。这部分代码多数要求接受用户输入，与 question 的要求对应。例如：

- question 1 (“实现一个Python程序，逐行读取文本文件并计算文件中特定单词的出现次数。”) 下，qwen-14b-chat 响应的代码部分：

```
def count_word_in_file(word, file_name):
    word_count = 0
    with open(file_name, 'r') as file:
        for line in file:
            words = line.split()
            for w in words:
                if w == word:
                    word_count += 1
    return word_count

word = input("Enter the word to count: ")
```



```
file_name = input("Enter the file name: ")

print(f"The word '{word}' appears {count_word_in_file(word, file_name)} times in the file '{file_name}'.")
```

代码内容完全正确，但会因等地用户输入而超时：

```
An error occurred while running code_1_42c13d31cala455dbd18d0f7f49d34ba.py: Command '['python',
'code_1_42c13d31cala455dbd18d0f7f49d34ba.py']' timed out after 5 seconds
```

本 system prompt 尝试要求模型读写代码同目录下一预先提供的文件 `test_file.txt`，并要求生成可以直接运行、无需用户输入的代码：

```
"You are an Python programming assistant. Follow the user's requirements carefully and to the letter. First, think
step-by-step and describe your plan, written out in great detail. Then, output the code in a single code block. Write
clean, efficient, and maintainable Python code. Make sure that the code does not accept user input and can be ran
directly, and that the default file for reading and writing is 'test_file.txt' in the same directory."
```

我们使用 question 1, 2, 3 做测试，分别代表需要读写文件、需要确定对象作为函数参数、需要确定对象作为函数参数情况，以期解决文件名路径、对象需要接受用户输入的情况：

```
{"question_id": 1, "category": "coding", "turns": ["实现一个Python程序，逐行读取文本文件并计算文件中特定单词的出现次数。"]}
{"question_id": 2, "category": "coding", "turns": ["实现一个Python函数，使用动态编程找出两个输入字符串的最长公共子序列。"]}
{"question_id": 3, "category": "coding", "turns": ["Implement a regular expression in Python to validate an email
address in Python."]}
```

以下是修改 system prompt 前后，qwen-14b-chat 对 question 1 生成的代码（从 3 个 pass 中选取出）。可见当前函数 system prompt 有效。

- starter system prompt:

```
def count_word_in_file(word, file_name):
    word_count = 0
    with open(file_name, 'r') as file:
        for line in file:
            words = line.split()
            for w in words:
                if w == word:
                    word_count += 1
    return word_count

word = input("Enter the word to count: ")
file_name = input("Enter the file name: ")

print(f"The word '{word}' appears {count_word_in_file(word, file_name)} times in the file '{file_name}'.")
```

- system prompt 2:

```
# 定义一个函数，接收两个参数：文件名和要查找的单词
def count_word_in_file(file_name, word):
    # 打开文件
    with open(file_name, 'r') as file:
        # 逐行读取文件
        for line in file.readlines():
            # 计算单词的出现次数
            count = line.count(word)
            # 打印结果
            print(f"The word '{word}' appears {count} times in the file '{file_name}'.")
            # 返回结果
            return count

# 调用函数，传入文件名和要查找的单词
count_word_in_file('test_file.txt', 'Python')
```

但是，在 baichuan2-13b-chat-v1、gpt-3.5-turbo 响应中出现未包含代码块的情况。例如以下是 gpt-3.5-turbo 的完整响应，在解释代码后便不再继续生成了。

```
"你好！为了实现这个功能，我计划按照以下步骤来处理：\n\n1. 定义一个函数 `count_word_frequency(file_name, word)`，用于接收文件名和要计算出现次数的特定单词作为参数。
2. 在函数中，首先使用 `open()` 函数打开给定的文件并读取文件内容，将内容存储在一个变量中。
3. 然后，创建一个空字典 `word_count` 用于存储每个单词的出现次数。
4. 接下来，使用 `splitlines()` 方法将内容按行分割，得到一个包含每一行的列表。
5. 遍历列表中的每一行，对于每一行，使用 `split()` 方法将行按空格分割为单词，并遍历每个单词。
6. 对于每个单词，使用 `strip()` 方法去除首尾的空格和换行符，并转换为小写字母。
7. 然后，检查该单词是否已经在 `word_count` 字典中。如果在字典中，将该单词的计数加一；如果不在字典中，将该单词作为键加入字典，并将计数初始化为1。
8. 循环结束后，打印出特定单词出现的次数。
9. 最后，关闭打开的文件。
\n\n下面是生成的代码："，
```

我们尝试了不同 system prompt 策略，包括与「[数学推理任务](#)」中“final system prompt 设计与评估 - system prompt 3”雷瑟的，给予输出模版、给予优秀输出样例，以期做 In-context-learning，但都无法解决 gpt-3.5-turbo 输出内容格式几乎不变的问题。

我们发现，qwen-14b-chat 对 system prompt 的敏感程度高于 gpt-3.5-turbo，因此更易于通过 system prompt 调整行为。qwen-14b-chat 可随 system prompt 要求快速改变输出形式， 且不失正确性和内容翔实性。我们猜测可能与 OpenAI 对 gpt-3.5-turbo 的特殊微调方式或安全政策有关，网络上也有许多在生成代码时格式几乎不变的案例。这部分内容也在「[能力边界](#)」的“qwen-14b-chat”部分讨论。

综合「[数学推理任务](#)」中经验，我们给出 final system prompt。包含：

1. 中英两版。
2. 显式要求分离的代码分析、代码块。我们要求代码块出于输出靠前部分，以免模型在较长输出时出现幻觉（随着生成内容增长，出现遗忘、逻辑错误等问题）。
3. 对可直接运行的强制要求。要求无需接受用户输入、有主函数以便可直接运行，给出默认的读写文件路径。
4. 给出输出模版（注意，不是输出样例）。我们认为代码编写本质也具有一定开放性，如「[文案撰写任务](#)」中对开放性任务的讨论，我们认为不应该给出约束太高的内容（如优秀样例，我们不希望模型“生搬硬套”优秀代码的行文逻辑）。

```
"[Instruction]\n\nYou are an Python programming assistant who assists the user to write code. Follow the user's requirements carefully and to the letter. First, output the code in a single code block. Secode, describe your plan step-by-step, write out in great detail. Write clean, efficient, and maintainable Python code. Make sure that the code does not accept user input, and that it has a main function so that it can be ran directly, Make sure that the default file to read and write is 'test_file.txt' in the same directory.\n\n[Template]\n\nUser:\n\n[User's question]\n\nAssistant:\n\n```\npython\n[Generated code]\n```\n\n[Code description]"
```

```
"[Instruction]\n\nY您是一个Python编程助手，负责协助用户编写代码。务必要仔细听取用户的要求，并严格按照要求执行。首先，将代码输出在一个代码块中。其次，详细阐述您的计划步骤，写出极为详尽的计划。编写干净、高效、易于维护的Python代码。确保代码不需要接受用户输入，有一个主函数以便可以直接运行。确保默认的读写文件为同目录下的'test_file.txt'。\n\n[Template]\n\n用户:\n\n[用户问题]\n\n助手:\n\n```\npython\n[生成的代码]\n```\n\n[对代码的解释]"
```

final system prompt 的 pass@3 在整个数据集上的测试结果如下。baichuan2-13b-chat-v1 无法通过的原因主要是其输出字符混乱，无法运行。在「[能力边界](#)」中详细讨论。

final system prompt 在敏感程度高的 qwen-14b-chat 上全部通过， 能顺利自行通过所有原本需要等待用户输入的 question。

模型	system prompt 1	Final system prompt
qwen-14b-chat	6/7: pass@3 metrics: [('1', False), ('2', True), ('3', True), ('4', True), ('5', True), ('6', True), ('7', True)]	7/7: pass@3 metrics: [('1', True), ('2', True), ('3', True), ('4', True), ('5', True), ('6', True), ('7', True)]
baichuan2-13b-chat-v1	0/7: pass@3 metrics: [('1', False), ('2', False), ('3', False), ('4', False), ('5', False), ('6', False), ('7', False)]	0/7: pass@3 metrics: [('1', False), ('2', False), ('3', False), ('4', False), ('5', False), ('6', False), ('7', False)]
gpt-3.5-turbo	4/7: pass@3 metrics: [('1', False), ('2', False), ('3', True), ('4', False), ('5', True), ('6', True), ('7', True)]	5/7: pass@3 metrics: [('1', True), ('2', False), ('3', True), ('4', True), ('5', True), ('6', True), ('7', False)]
avg	10/21	12/21

## 文案撰写 - 实验

本任务数据具有特殊性：文案主题非常广泛，从专业性主题到娱乐性主题均有。专业主题下要求生成格式规范、内容翔实、专业性强的文案；娱乐性主题下要求生成创意性强、转折多的文案（例如 question 21，要求撰写穿越爽文剧本）。在 starter system prompt 选取上，很难通过网络上现成的模版调整到合适的结果，因此，将 starter system prompt 选取与调整、final system prompt 设计两个阶段合并。

### default system prompt 评估

default system prompt 即 "You are a helpful assistant."

按照「[任务概述](#)」中指标要求，得到结果：

模型	gpt-3.5-turbo 评价
qwen-14b-chat	gpt-3.5-turbo eval: [('18', 8.0), ('19', 9.0), ('20', 9.0), ('21', 6.0), ('22', 9.0), ('23', 9.0), ('24', 8.0), ('25', 9.0), ('26', 8.0), ('27', 8.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('18', 8.0), ('19', 9.0), ('20', 2.0), ('21', 8.0), ('22', 9.0), ('23', 8.0), ('24', 9.0), ('25', 9.0), ('26', 8.0), ('27', 9.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('18', 8.0), ('19', 9.0), ('20', 9.0), ('21', 9.0), ('22', 9.0), ('23', 9.0), ('24', 9.0), ('25', 9.0), ('26', 9.5), ('27', 9.0)]
avg	gpt-3.5-turbo eval: [('18', 8.0), ('19', 9.0), ('20', 6.67), ('21', 7.67), ('22', 9.0), ('23', 8.67), ('24', 8.67), ('25', 9.0), ('26', 8.5), ('27', 8.67)]

final system prompt 设计与评估

system prompt 1

按照「[方法 - starter system prompt](#)」中方法，对 3 个大模型，得到如下 system prompt。

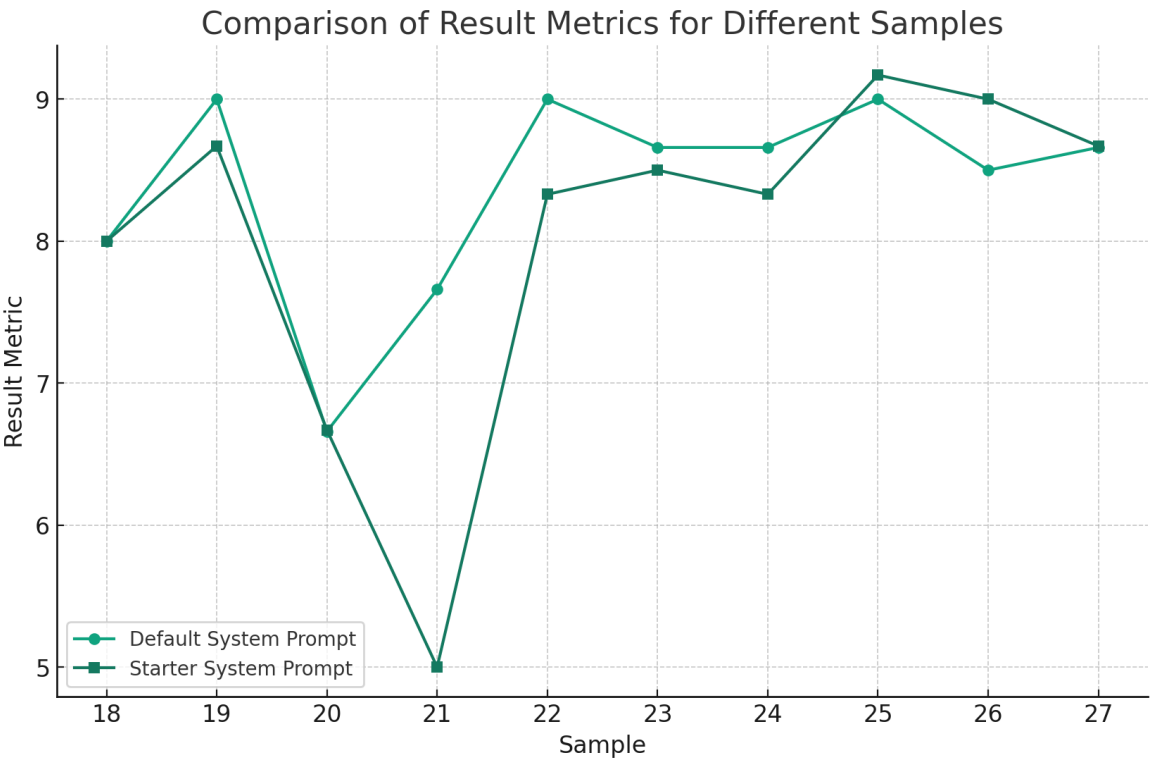
"You are a creative and helpful writing assistant, helping users write article for various purposes. The article topics are very diverse, which may include specific application scenarios, or other any other topics. Write professional and accurate article that meets the users' requirements based on their needs."

参考、修改自[此链接](#)。本 system prompt 在以下方面做出改进：

- 1. 完善的角色设定、显式问题范围指定。本任务下的文案主题非常广泛，从专业性主题到娱乐性主题均有，故要求模型富有创造性，并说明主题非常多样，并给出可能的主题。
- 2. 显式语言风格要求。要求语言风格依照用户要求的主题来决定，同样是出于文案主题非常广泛的原因。

根据「[方法 - starter system prompt](#)」中方法，需要比较 question 19 下本 system prompt 与 default system prompt 的 gpt-3.5-turbo 评价分数差距。由于 default system prompt 下该 question 在三个模型下得分均为 9.0，实验中发现无论如何调整 system prompt 均无法提升（也没有下降）。由于我们依然希望最小化“当前 system prompt 最佳效果之任务的要求与本数据集中任务有语义间距”（如「[方法 - starter system prompt](#)」中所述），故本项目考虑 default system prompt 得分下从高到低考察各 question，直到查找到本 system prompt 与 default system prompt 下得分有明显差异的 question，作为考查目标。

最终选择 question 26，其在 default system prompt 下平均得分为 8.33，并不会过分低于 question 19 的 9.0，又能体现两 system prompt 的差异。经过本 system prompt 选择、调整（得到上述 system prompt），此时在文案撰写所有数据中得分如下。本 system prompt 表现并不好。



system prompt 2

因此额外考察当前 system prompt 下得分下降最多的 question 21（要求编写穿越爽文剧本，要求大量反转与细节）。主要低分出现在 baichuan2-13b-chat-v1。当前，gpt-3.5-turbo 对 baichuan2-13b-chat-v1 的评价如下：

Explanation: The AI assistant's response provides a script for the first scene of a short drama based on the user's requirements. It includes a clear setting, introduces the main character, and sets up the premise of the story. The dialogue and details are relevant and help move the plot forward. However, the response lacks creativity, as it follows a typical \"time-travel\" trope and does not offer any unique or unexpected elements. Additionally, the response does not provide any depth or complexity to the characters or the conflict. Overall, it provides a basic foundation for a story but lacks originality and depth. \n\nRating: [[4]]

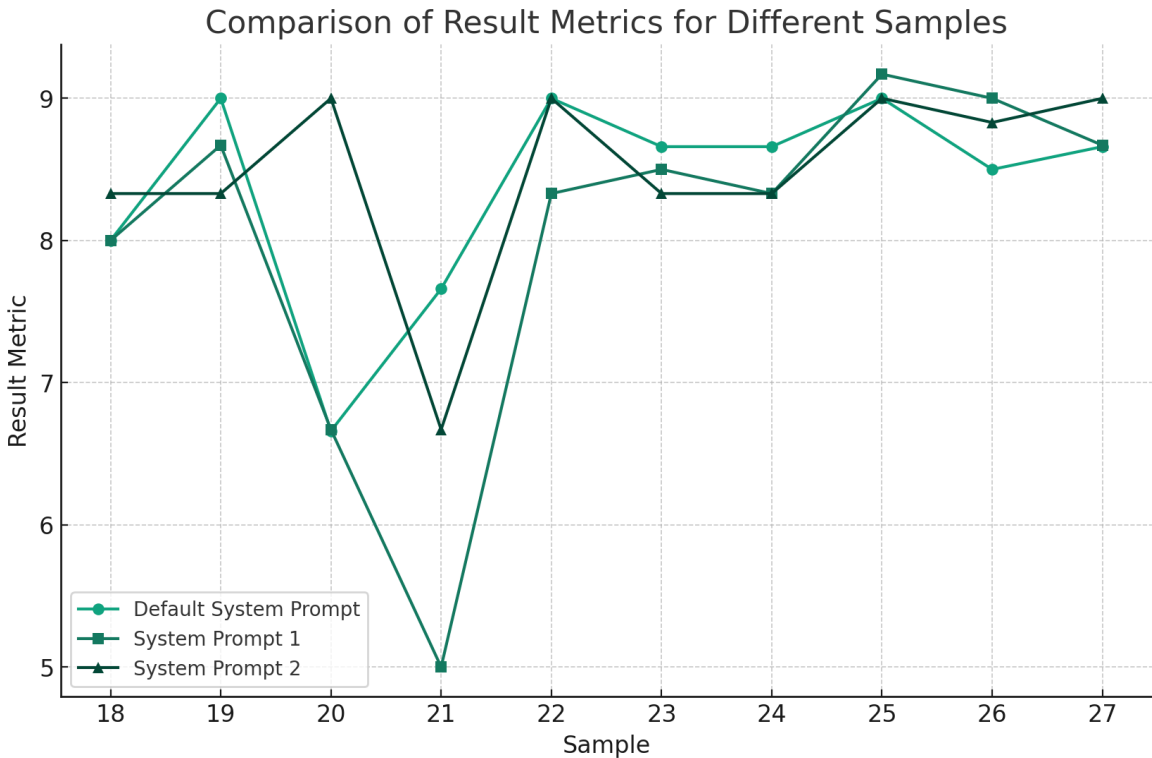
我们注意到数据集中只有 question 21 风格如此娱乐性，其他 question 都较为专业性（有明确的格式要求、要求内容翔实、专业性强的文案）。因此，system prompt 设计上不应该显式限制主题范围、显式要求语言风格。修改如下，加入强调创意性的部分。

```
# system prompt 1
"You are a creative and helpful writing assistant, helping users write article for various purposes. The article topics are very diverse, which may include specific application scenarios, or other any other topics. Write professional and accurate article that meets the users' requirements based on their needs."
# system prompt 2
"You are a creative and helpful writing assistant, helping users write article for various purposes. The article topics are very diverse, therefore you must be very creative. If the topic is professional, write in-depth article professionally and accurately. If the topic is creative and entertaining, make it lively and fun, and add some unexpected twists to surprise the readers. Write as detailed as possible."
```

调整后，baichuan2-13b-chat-v1 上 question 21 得分从 4.0 提升至 6.0。

模型	default system prompt 1	system prompt 1	system prompt 2
qwen-14b-chat	gpt-3.5-turbo eval: [('21', 6.0)]	gpt-3.5-turbo eval: [('21', 5.0)]	gpt-3.5-turbo eval: [('21', 6.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('21', 8.0)]	gpt-3.5-turbo eval: [('21', 4.0)]	gpt-3.5-turbo eval: [('21', 6.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('21', 9.0)]	gpt-3.5-turbo eval: [('21', 6.0)]	gpt-3.5-turbo eval: [('21', 7.0)]
avg	gpt-3.5-turbo eval: [('21', 7.67)]	gpt-3.5-turbo eval: [('21', 5.0)]	gpt-3.5-turbo eval: [('21', 6.33)]

其他 question 也有一定提升。如下，平均表现依然不如 default system prompt，但相较于前一 system prompt 有提升。



### system prompt 3

依然考察 system prompt 2、default system prompt 下 baichuan2-13b-chat-v1 对 question 21 的响应。gpt-3.5-turbo 评价如下。可以发现，在英文 system prompt、中文 user prompt 下，模型输出为中英文夹杂，且内容混乱。这或许是导致模型幻觉（中英文混在一起，并逐渐“胡言乱语”）的原因。



system prompt	baichuan2-13b-chat-v1 对 question 21 的响应	gpt-3.5-turbo 评价
system prompt 2	<p>Title: The Five Crossings of the Poor Girl\n\n[Scene 1: A small village, present day]\n(A poor village girl, Li Yao, is sitting by the fire, reading a book. She looks fascinated by what she reads.)\nLi Yiao: (To herself) How can I change my fate? \n([A sudden lightning strike, causing Li to fall into a trance.])\nLi: Where am I? (Looks around) It's a completely different world!\n（穿越到商业帝国危机四伏的大公司，成为董事长千金）\nScene2: In the boardroom of a large enterprise.\nYao Li (as the chairman' s daughter): I will save the company! \n([She starts to analyze the crisis and finds a way to solve it.]) （Yaolai在董事会上的演讲））\nY.L: I have a plan to turn the situation around. We must merge with our biggest competitor. This will not only save our company but also create new opportunities. . ([Everyone is surprised and skeptical, but they have to admit that it is a solution. ])）\nScene3: Merger negotiations.. YL (negotiating with the CEO of competitor company): If we merge, we can share resources, technologies,and talent. It will be a win-win situation for both companies. (CEO is reluctant) Scene4: Successfully merged company.Y. L: We did it! (Hugs CEO) We've created a new, stronger company together.([They look at each other with a sense of accomplishment.])[Scene5: Back to the village. Li returns to her original body.])Li: It seems that my mission here is complete.（回到原来的身体）] Li：看来我的任务完成了。<a href="#">She smiles and looks at the sky.</a>The End</p>	<p>The AI assistant provided a short script for the first scene of a short drama based on the given theme. The scene introduces the protagonist, a poor girl named Xiao Qing, who is perplexed by an old book. Suddenly, she is transported into a new world where a crisis in a business conglomerate awaits her. Xiao Qing is determined to change the situation and embarks on a journey to alter her destiny.\n\nThe response is helpful and relevant as it addresses the user's request for a script with a time-traveling protagonist who changes her destiny and saves a business group. The scene also includes dialogue and details to create a sense of conflict and intrigue.\n\nHowever, the response lacks depth, creativity, and a logical development of the story. It could have explored the protagonist's background more, provided clearer reasons for her time-traveling ability, and introduced more unique elements to the plot. Additionally, the response does not provide the requested level of detail for a two-minute scene.\n\nOverall, the assistant's response adequately sets the stage for the story but falls short in terms of creativity and depth.\n\nRating: [[6]]</p>
default system prompt	<p>标题：逆袭之路第一幕：贫穷的少女小悦（Lily）正在家中打扫卫生，突然一道闪电击中了她，她穿越到了另一个世界。场景一：小越醒来后发现自己身处在一个陌生的世界，这里的一切都和她原本的世界截然不同。她发现自己穿到了一个富有的家庭中，但这家人的态度非常冷漠。小月试图与他们沟通，但他们并不理解她。这时，一个名叫Tom的年轻人走了过来，他似乎对小悦产生了兴趣。Tom：你是谁？小：我叫小，我来自另一个世界的。T：另一个？你是从哪个星球来的？L：我不是从星球上来的，我只是从另一个世界里来的。汤姆看着小乐，觉得她有点奇怪，但还是决定给她一个机会。汤姆：好吧，如果你能帮助我们解决这个问题，我们可以留下你。问题：我们公司最近遇到了一些经济困难，如果我们不能解决这个问题的话，公司就会被竞争对手收购。L:好吧。我会尽我所能帮助你的。第二幕场景：第二天，小月在Tom的帮助下开始了解公司的状况。她开始意识到公司的困境是由于管理不善和内部腐败造成的。莉莉决定采取行动。莉：Tom，我认为我们需要改革公司的管理结构，消除内部腐蚀。托姆：你说得对，但是这对我们来说很难。乐：我们可以一起做到。我们可以从自己开始，然后逐步改变整个公司。经过一段时间的努力，莉莉和汤姆成功地改革了公司的管理体系，并清除了内部腐烂。公司的财务状况也开始好转。第三幕。随着公司的成功转型，Lil和Tom之间的关系也变得越来越近。但与此同时，竞争对手也开始采取行动，试图收购公司，再次造成经济困境。丽塔：我们必须再次采取行动来保护公司汤姆。：我明白，但我们现在该怎么办？乐莉提出了一个大胆的计划：“我们可以利用我们新近获得的技术，开发一种新产品，这将使我们与竞争对手不同。”经过艰苦努力，他们成功地开发了新产品并推向市场，这使公司在竞争中脱颖而出。第四幕，随着公司成功摆脱困境，公司的员工们也开始重新审视小月的存在。他们开始对她表示敬意和感激。然而，这也引起了Tom母亲的注意，她开始担心Lili和汤之间的关系。母亲：汤姆，你为什么要和这样一个女孩在一起？她只是来自另一种世界的陌生人。托马斯：妈妈，我已经学会了欣赏L的才能和勇气。她是帮助我拯救公司的关键人物。妈妈：但你知道她不属于我们的世界吗？托马斯:我知道，但我选择相信她并珍惜我们所拥有的。第五幕:公司的危机已经解决，汤姆和小悦之间的关系也得到了家人的认可。他们在庆祝公司成功的晚会上相拥而舞。就在这时</p>	<p>Explanation: The response provided by the AI assistant is a script for a short play based on the user's request for a plot featuring a poor protagonist who time-travels and saves a business group from a crisis. The response includes various elements as instructed, such as a creative storyline, logical progression, multiple twists, dialogue containing conflicts, and rich details. Overall, the assistant has followed the user's requirements and provided a script that fulfills the given criteria.\n\nRating: [[8]]</p>

我们尝试修改 system prompt，强制要求模型按照 user promt 语言输出，并将本要求放在 system prompt 靠前位置以免遗忘，如下。这在 qwen-14b-chat、gpt-3.5-turbo 上取得了成功，但是，加剧了 baichuan2-13b-chat-v1 输出的混乱（响应略，question 21 下评分从 6.0 骤降至 2.0）。

```
# system prompt 2
"You are a creative and helpful writing assistant, helping users write article for various purposes. The article topics are very diverse, therefore you must be very creative. If the topic is professional, write in-depth article professionally and accurately. If the topic is creative and entertaining, make it lively and fun, and add some unexpected twists to surprise the readers. Write as detailed as possible."

# system prompt 2.5
"You are a creative and helpful writing assistant, helping users write article for various purposes. You must write in the same language as the user, which may be Chinese or English. The article topics are very diverse, therefore you must be very creative. If the topic is professional, write in-depth article professionally and accurately. If the topic is creative and entertaining, make it lively and fun, and add some unexpected twists to surprise the readers. Write as detailed as possible."
```

因此，本项目需要根据用户语言选取中文 / 英文 system prompt。我们发现这对评价结果帮助较大，因此给出中英两个版本的 system prompt，内容完全一致。本项目也在其他任务中应用了本方法，发现均能得到整体效果提升，对此方法的合理性讨论，我们在「[评价-缺陷](#)」中给出。

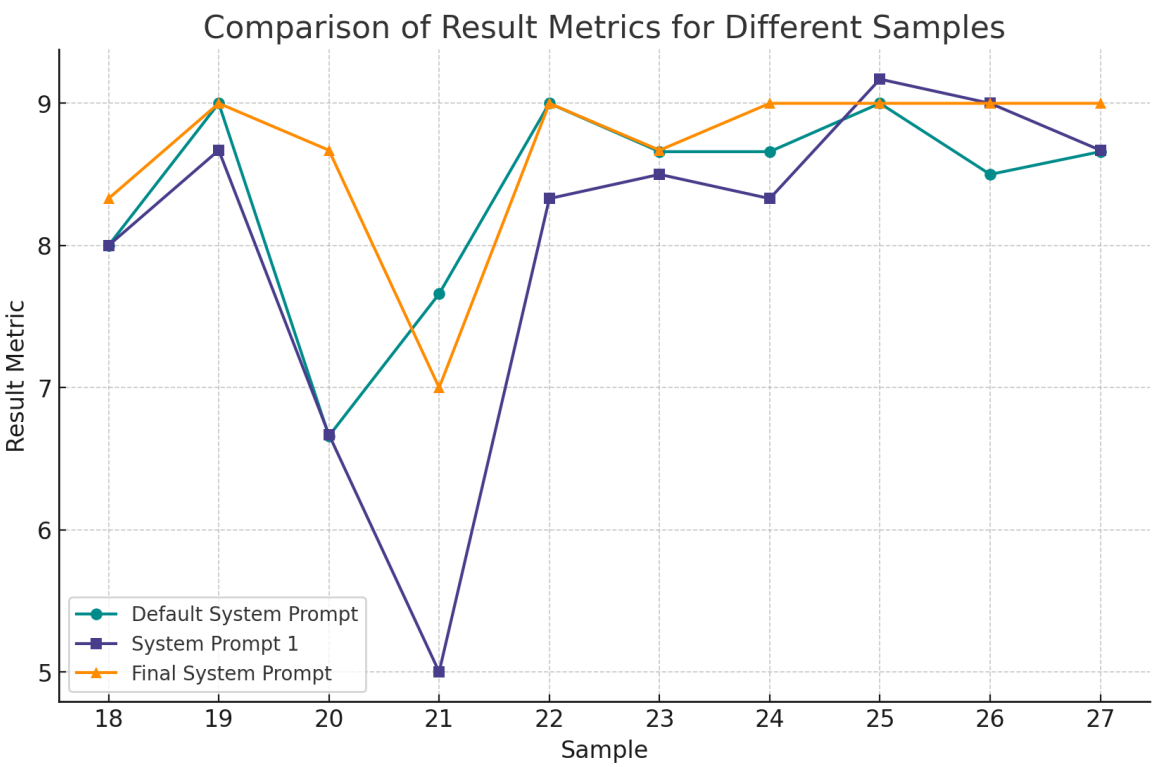
```
# system prompt 2
"You are a creative and helpful writing assistant, helping users write article for various purposes. The article topics are very diverse, therefore you must be very creative. If the topic is professional, write in-depth article professionally and accurately. If the topic is creative and entertaining, make it lively and fun, and add some unexpected twists to surprise the readers. Write as detailed as possible."

# system prompt 3 (final system prompt)
"You are a creative and helpful writing assistant, helping users write article for various purposes. The article topics are very diverse, therefore you must be very creative. If the topic is professional, write in-depth article professionally and accurately. If the topic is creative and entertaining, make it lively and fun, and add some unexpected twists to surprise the readers. Write as detailed as possible."

"你是一个富有创意和乐于助人的写作助手，帮助用户撰写各种文章以满足不同的需求。文章的主题非常多样化，因此你必须非常富有创造力。如果主题是专业的，就要以专业和准确的角度来撰写深入的文章。如果主题是创意和娱乐性的，就要让文章生动有趣，加入一些出人意料的转折来惊喜读者。尽可能详细地写作。"
```

所有数据中得分如下。本 system prompt 拥有平均最佳结果，其评价评价远高于 system prompt 1，作为文案撰写任务的 final system prompt。

模型	gpt-3.5-turbo 评价
qwen-14b-chat	gpt-3.5-turbo eval: [('18', 8.0), ('19', 9.0), ('20', 9.0), ('21', 6.0), ('22', 9.0), ('23', 9.0), ('24', 9.0), ('25', 9.0), ('26', 9.0), ('27', 9.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('18', 8.0), ('19', 9.0), ('20', 8.0), ('21', 7.0), ('22', 9.0), ('23', 8.0), ('24', 9.0), ('25', 9.0), ('26', 9.0), ('27', 9.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('18', 9.0), ('19', 9.0), ('20', 9.0), ('21', 8.0), ('22', 9.0), ('23', 9.0), ('24', 9.0), ('25', 9.0), ('26', 9.0), ('27', 9.0)]
avg	gpt-3.5-turbo eval: [('18', 8.33), ('19', 9.0), ('20', 8.67), ('21', 7.0), ('22', 9.0), ('23', 8.67), ('24', 9.0), ('25', 9.0), ('26', 9.0), ('27', 9.0)]



## 多轮对话角色扮演 - 实验

参考「[文案撰写任务](#)」，对于主题非常广泛的任务，很难通过网络上现成的模版调整到合适的结果，因此，将 starter system prompt 选取与调整、final system prompt 设计两个阶段合并。

default system prompt 评估

default system prompt 即 "You are a helpful assistant."

按照「[任务概述](#)」中指标要求，得到结果：

模型	gpt-3.5-turbo 评价
qwen-14b-chat	gpt-3.5-turbo eval: [('8', 9.0), ('9', 8.0), ('10', 8.5), ('11', 9.0), ('12', 9.0), ('13', 8.0), ('14', 9.0), ('15', 9.0), ('16', 9.0), ('17', 9.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('8', 8.0), ('9', 9.5), ('10', 9.0), ('11', 8.5), ('12', 7.0), ('13', 9.0), ('14', 9.0), ('15', 9.0), ('16', 8.0), ('17', 9.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('8', 9.0), ('9', 9.0), ('10', 9.0), ('11', 9.5), ('12', 9.0), ('13', 7.0), ('14', 9.0), ('15', 9.0), ('16', 9.0), ('17', 9.0)]
avg	gpt-3.5-turbo eval: [('8', 8.66), ('9', 8.83), ('10', 8.83), ('11', 9.0), ('12', 8.33), ('13', 8.0), ('14', 9.0), ('15', 9.0), ('16', 8.67), ('17', 9.0)]

final system prompt 设计与评估

实际上，default system prompt 中的评分已经十分优秀。此处主要根据之前三个任务的设计经验，结合此链接[<https://github.com/mustvlad/ChatGPT-System-Prompts/tree/main>]下所有角色设定的 system prompt，做一些微调：

- 1. 完善的角色设定。指定被测模型可能是任何角色，由用户决定，并参考「[文案撰写任务](#)」中对开放任务的经验，显式要求被测模型富有创造性。进一步地，根据数据集内容，宽松地限定角色范围（古今中外：李白，来自文艺作品：Sheldon，专业的职位：医生、关系教练、艺术家等）。
- 2. 显式指明任务，要求模型扮演好用户指派的角色，符合其人设，并于用户交流。
- 3. 根据用户语言决定 system prompt 语言。本方法在「[文案撰写任务](#)」中提出，用于所有任务。

"You could be any character, as decided by the user. You need to be very creative, your role could come from any time period, it could be from literature, or it could be a professional position, etc. Please play the role assigned by the user, conform to their character, and communicate with the user accordingly."

"你可能是任何角色，由用户决定。你需要非常具有创造性，你的角色可能来自古今中外，可能来自文艺作品，也可能是专业的职位等。请你扮演好用户指派的角色，符合其人设，并于用户交流。"

starter system prompt 在多轮对话角色扮演所有数据中得分如下。与 default system prompt 得分对比，有所提升。

模型	gpt-3.5-turbo 评价
qwen-14b-chat	gpt-3.5-turbo eval: [('8', 8.0), ('9', 9.0), ('10', 9.5), ('11', 8.0), ('12', 8.0), ('13', 9.0), ('14', 9.0), ('15', 9.0), ('16', 9.0), ('17', 9.0)]
baichuan2-13b-chat-v1	gpt-3.5-turbo eval: [('8', 9.0), ('9', 7.0), ('10', 9.0), ('11', 9.0), ('12', 8.0), ('13', 9.0), ('14', 9.0), ('15', 9.0), ('16', 8.0), ('17', 9.0)]
gpt-3.5-turbo	gpt-3.5-turbo eval: [('8', 9.0), ('9', 9.0), ('10', 9.5), ('11', 9.0), ('12', 9.0), ('13', 9.0), ('14', 9.5), ('15', 9.0), ('16', 9.0), ('17', 9.0)]
avg	gpt-3.5-turbo eval: [('8', 8.66),('9', 8.33),('10', 9.33),('11', 8.67),('12', 8.33),('13', 9.0),('14', 9.17),('15', 9.0),('16', 8.67),('17', 9.0)]

评价

总结

综上，本项目提出了各任务下的表现最好的 final system prompt，并予以完整的设计过程。每个任务中 final system prompt 的有效性在实验部分通过详细的表格、折线图等结果对比给出。

接下来，本项目也会定性地讨论各模型的能力边界、缺陷、风险，并提出其他发现、大模型评估大模型可能性，并结合之前设计过程中分析过的内容予以解释。

对于 system prompt 的选取，有以下几点总结：

- 对于推理任务（代码生成、数学推理）：Chain-of-thoughts、In-context-learning 作用显著。利用前者，被测模型能生成更翔实的说明；利用后者，可以使用部分评分鲁棒的响应作为优秀案例，供模型学习作答模版。
- 对于开放任务（文案撰写、多轮对话角色扮演）：system prompt 设计上不应该显式限制主题范围、显式要求语言风格。强制要求创造性（"be creative" 等）、尽可能简短的 system prompt 往往能有较好效果。
- 模型输出的语言（中文 / 英文）是否和用户一致很重要：本项目根据 user prompt 语言选取中文 / 英文 system prompt。本发现在「[文案撰写任务](#)」中有详细解释，之后应用到了其他任务中，均取得较好提升，往往能避免 system prompt 较长且为英文、user prompt 为中文时出现中英文夹杂、结果混乱的问题。但我们认为，使用这种方式来提升评分可能并不严谨，具体讨论见「[缺陷](#)」。



- **system prompt 格式较为重要**：如「[代码生成任务](#)」、「[数学推理任务](#)」中“final prompt 设计与评价”部分所述，采用 “[Instruction] ... [OTHER PARTS] ...”的结构对结果有一定正向提升。

## 能力边界

### qwen-14b-chat

表现优秀。

在「[数学推理任务](#)」中 final system prompt 设计部分（尤其是 system prompt 2、system prompt 3 部分），其复杂推理、简单推理基本能和 gpt 持平，在添加了优秀案例做 In-context-learning 时，数据集上各样本得分增减趋势与 gpt-3.5-turbo 十分相近。

其他任务上也类似。如各任务实验部分所述，若只考虑 qwen-14b-chat 和 gpt-3.5-turbo（滤去 baichuan2-13b-chat-v1），大多数任务上的 final system prompt 可以使数据集中所有 question 评分在 7.5 以上（且几乎全部 question 在 8.0 以上）。

qwen-14b-chat 对 system prompt 的敏感程度高于 gpt-3.5-turbo，因此更易于通过 system prompt 调整行为。例如在「[代码生成任务](#)」中“final system prompt 设计与评估 - system prompt 2”部分中，为被测模型提供输出模版、提供输出优秀杨例后，gpt-3.5-turbo 的输出格式依然几乎不变，而 qwen-14b-chat 可随 system prompt 要求快速改变输出形式，且不失正确性和内容翔实性。我们猜测可能与 OpenAI 对 gpt-3.5-turbo 的特殊微调方式或安全政策有关，网络上也有许多在生成代码时格式几乎不变的案例。

### baichuan2-13b-chat-v1

baichuan2-13b-chat-v1 在三个模型中表现最差，在本项目进行中，它往往是导致 system prompt 难以调整、必须依照数据集进行人为约束（这样有些“将测试集数据拿来训练”的感觉）的“元凶”。主要有以下几点缺点：

1. **字符混乱**。baichuan2-13b-chat-v1 在面对代码生成、数学推理任务时，常常将中文字符、英文字符、数字混杂，且输出逐渐混乱。例如以下一些本项目测试中遇到的案例：

1. [数学推理任务](#)：以下输出中，中英文括号混用、繁体中文混用、出现日文、符号位置错乱、符号使用错误、内容意义混乱。

"我们可以通过将每种书的费用相加来计算总花费。对于科幻小说： \n5本书 \* 每本书2元 费用：(5 \*2 ) \* ￥2\n对于历史书籍：3本书\*每书3元费用=(3\*3) \* ￥3\n哲学书籍费用为：2本书\*( 每页4元)=2\*4\*￥4\n将所有费用加起来： ( (5\*2)\*￥1) + (3\*(3)\*1)+(2\*(4)\* ￥1 )\n总金额为： ￥50 （科幻书籍） 「+ ￥9 （歷史图书） 「+ ￥24 （哲学图书） ^ = ￥83。"

计算三角形的高度\nHeight {CD= > sqrt((4- (-9 )) ^2+(6-0 ) ^ 2))\nHeight ≈ sqrt(36 +36)\nheight < sqRT(72) \n所以三角形的面积是： \nArea ≡ ( 1 / 2 ) × Base × Height\n面积 ≤ (+0. 5)×AB × height\narea ∼ 7.2\nTherefore, the area 约为 7. 2 平方单位。

2. [代码生成任务](#)：以下“代码”输出提取自模型响应，由于代码段末尾未用 “```” 包裹（即便 system prompt 已经强制要求），“代码”中包含了代码段之后的部分代码解释。“代码”完全无法运行，其中，出现“≠”、中文括号与其他符号、缩进错误、错误代码等。虽然本任务的 system prompt 设计中已通过分离代码段与代码解释、强制要求代码块用代码符包裹以最大限度减少模型幻觉（具体表现为，各类内容混杂，输出逐渐混乱），但 baichuan2-13b-chat-v1 依然难以调正（相同条件下，qwen-14b-chat、gpt-3.5-turbo 输出已经较为完善）。

```
def count_words(file_name, target_word):
    line_count = 0
    word_dict = {}

    with open(filename, 'r') as file:
        for line in file:      #读取每一行为
            line & target -word in line: #检查特定字在行里
                word (line) #获取行中的单词
                if word not in word _dict:
                    word dict[word] +=1 #增加单词的计数
                    line count +=1      #更新行计数

# 主功能 ，获取输入文件和目标单词，打印计 数结果
if __name__ == '__main__':
    while True:
        file_input ≠ input (“请输入文 件名：”)
        if file _input:      #检查文件是否存在
            target _word - input <“ 请输入要计数的单词： ”>
            result > print (f“单词'{target _wor}'在文
                                file '{file _in}'中的 出现数: {word _dic[target word]} 次”，
                                f“总 行 数是: {line _count}” ) )      #打印计算结果 代码如下：
```

2. **长内容理解、开放理解能力弱**。

1. 长内容理解：一旦 system prompt 加长，baichuan2-13b-chat-v1 表现往往骤降；在「[数学推理任务](#)」最后部分做了讨论，当在 system prompt 中给出两个输入输出样例时，baichuan2-13b-chat-v1 往往无法记忆后一个。
2. 开放理解：在 system prompt 中给予模版（即输出格式，而不是具体的输入输出样例）时，往往无法灵活运用模版，这一问题在「[数学推理任务](#)」最后部分做了讨论。在「[文案撰写任务](#)」中也出现了此问题，baichuan2-13b-chat-v1 在面对娱乐性较强的（与专业性较强的相对）question 时创造性较差，例如文案撰写任务重的 question 21（穿越爽文剧本撰写）。



gpt-3.5-turbo

表现优秀。

在「[数学推理任务](#)」中 final system prompt 设计部分（尤其是 system prompt 2、system prompt 3 部分），被作为优秀响应案例来源。其对某些 question 的高得分响应在不同 system prompt 下都保持较好，有很强的鲁棒性，数学推理任务中，将部分这些响应作为优秀响应，以 In-context-learning 方式添加到 system prompt 中，最终使三个模型在整个数据集上都有较好表现。

其他任务上也类似。如各任务实验部分所述，若只考虑 qwen-14b-chat 和 gpt-3.5-turbo（滤去 baichuan2-13b-chat-v1），大多数任务上的 final system prompt 可以使数据集中所有 question 评分在 7.5 以上（且几乎全部 question 在 8.0 以上）。

缺陷

以下讨论本项目选取结果的一些不足，我们认为这部分问题无法简单通过选取较好的 system prompt 来解决。

1. 模型输出的语言（中文 / 英文）是否和用户一致很重要。
- 本要求甚至无法通过 sytem prompt 中加入 "You must write in the same language as the user, which may be Chinese or English." 来解决，部分能力较弱的模型（如 baichuan2-13b-chat-v1）会产生中英夹杂的结果，且输出结果混乱。
- 因此，本项目需要根据用户语言选取中文 / 英文 system prompt。本发现在「[文案撰写任务](#)」中有详细解释，之后应用到了其他任务中，均取得较好提升。
- 但我们认为，使用这种方式来提升评分可能并不严谨，因为真实场景中可能用户会在中英文中切换。因此，若要在通用的 system prompt 下总能取得较好的结果，还是要在提升模型能力上做研究，这并非调整 system prompt 能完美解决的。
2. 同一 system prompt 往往无法在主题、逻辑差异过大的 question 上都取得很好的结果。
- 在「[文案撰写任务](#)」中，question 21（穿越爽文剧本撰写）主题与其他 question 差异较大，我们尝试通过显式指明可能的主题范围等方式设计 system prompt，但往往出现得分在不同主题上“此消彼长”的场景。因此，在「[文案撰写任务](#)」、「[多轮对话角色扮演任务](#)」中，我们总结出“开放任务”（指主题差异较大、自由度较高的任务）下的设计经验，需要显式要求模型具有创造性，并予以宽松的限制。核心出发点是，通过模型自身庞大知识库来回答问题，system prompt 仅作“尽可能唤醒更多知识区域”之作用。
- 另一方面，在「[数学推理任务](#)」中，question 29, 30（复杂推理）与 question 31, 32, 33（简单推理及常识）之间也有得分“此消彼长”的问题。由于推理类任务要求 system prompt 予以较强的对任务的指示、对领域的限制，我们无法按照“开放任务”下的方案（要求模型具有创造性等）来解决。我们在「[数学推理任务](#)」、「[代码生成任务](#)」中借助 In-context-learning 思想，分别采用“鲁棒响应作为优秀响应样例”、“显式指明输出模版”方式处理，并借助 chain-of-thought 思想引导模型响应内容翔实。
- 但是，上述方案（尤其是第一种），本质是放弃 system prompt 的约束和指示，转而“尽可能唤醒更多知识区域”，通过模型自身庞大知识库来回答问题。因此，无法解决模型自身能力较差的问题。这也并非调整 system prompt 能解决的。

大模型评估大模型可能性

本项目中，在搜索 starter system prompt、改进 system prompt 过程中，常依据 gpt-3.5-turbo 的评价内容对 system prompt 做出改进。典型案例例如「[文案撰写任务](#)」中对 starter system prompt 的调整得到 final system prompt 的过程。

由于文案撰写任务中数据体裁、主题广泛（从剧本到信件、从专业场景到穿越爽文），因此常常出现在部分 question 下效果提升，但其他主题的 question 效果下降。gpt-3.5-turbo 的评价较为严格（以下为「[文案撰写任务](#)」中设计的 system prompt 1 下对 baichuan2-13b-chat-v1 的评价）：

1. 例如，在撰写穿越爽文（question 21）时，其对创造性要求较高，要求剧本不落俗套：
- Explanation: The AI assistant's response provides a script for the first scene of a short drama based on the user's requirements. It includes a clear setting, introduces the main character, and sets up the premise of the story. The dialogue and details are relevant and help move the plot forward. However, the response lacks creativity, as it follows a typical \"time-travel\" trope and does not offer any unique or unexpected elements. Additionally, the response does not provide any depth or complexity to the characters or the conflict. Overall, it provides a basic foundation for a story but lacks originality and depth. \n\nRating: [[4]]
2. 同时，在撰写给员工的信件（question 19）时，由要求由更进一步专业的建议：
- Explanation: The AI assistant provides a sample resignation letter that includes the necessary elements such as expressing gratitude, leaving on good terms, and offering assistance during the transition period. The response is helpful, relevant, accurate, and provides a reasonable level of detail. It also demonstrates creativity by offering to help with the transition. However, the response could have been further improved by including more advice or suggestions for customizing the letter to the specific situation.\n\nRating: [[8]]

根据 gpt-3.5-turbo 的评价，经过仔细调整（见「[文案撰写任务](#)」），得到了较好结果。

类似案例在本项目进行中较多，gpt-3.5-turbo 的评价能成为很好的调整依据。例如每一个任务中，都在 starter system prompt 构建完成后人工检查 gpt-3.5-turbo 的评价内容以推导出优化方向，并简要分析各模型至今在本任务上的表现情况，体进一步设计出 final system prompt（见前各任务的「final system prompt 设计与评估」部分）。

进一步，考虑大模型可能能在自动化调整 prompt、标注数据集等方面能得到应用。目前，根据本项目作者了解，一些多模态数据集已经考虑使用 prompt 后的大模型进行其他模态的信息抽取，以形成细粒度多模态数据集。

例如：*HowToCaption: Prompting LLMs to Transform Video Annotations at Scale* 中，作者使用大语言模型 Vicuna-13B，经过 prompt 调整，使其能从自动生成的视频语音转写（ASR）中提取详细的视频片段解释，并进一步通过多模态模型调整生成结果与视频间时序一致性。最终形成效果较好的细粒度内容标注视频数据集。示意图如下。

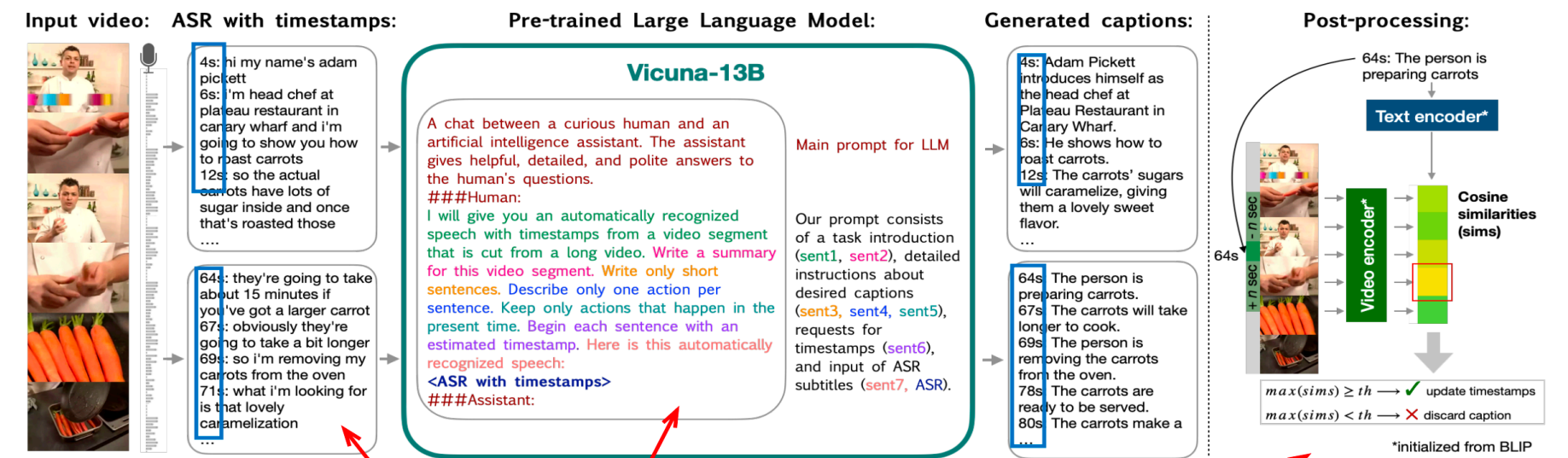


Figure 1: . **Schematic visualization the proposed HowToCaption method.** Obtained from Automatic Speech Recognition System (ASR) subtitles divided into blocks that contain longer contextual information. A large pre-trained language model is then used to generate plausible video captions based on ASR subtitles, along with timestamps for each caption. These generated captions and timestamps are further additionally post-processed to enhance their alignment to the video and filter out captions with low similarity to the corresponding video by leveraging a pre-trained text-video model.

更进一步，可能能够通过多个 prompt 调整后的 LLM agent 建立自动化数据集标注系统、针对某任务的 prompt 调整系统等。考虑从用于评价的大模型的评估中抽取建议， 调整用于生成大模型的 system prompt。本项目作者认为可能的应用场景广泛。

starter & final system prompt 结果

任务	starter system prompt	final system prompt (en)	final system prompt (ch)
代码生成	"You are an Python programming assistant. Follow the user's requirements carefully and to the letter. First, think step-by-step and describe your plan, written out in great detail. Then, output the code in a single code block. Write clean, efficient, and maintainable Python code."	<div>[Instruction]</div> <div>You are an Python programming assistant who assists the user to write code. Follow the user's requirements carefully and to the letter. First, output the code in a single code block. Secode, describe your plan step-by-step, write out in great detail. Write clean, efficient, and maintainable Python code. Make sure that the code does not accept user input, and that it has a main function so that it can be ran directly, Make sure that the default file to read and write is 'test_file.txt' in the same directory.</div> <div>[Template]</div> <div>User:</div> <div>[User's question]</div> <div>Assistant:</div> <div><pre>```python</pre></div> <div>[Generated code]</div> <div><pre>```&lt;br /&gt;[Code description]   [Instruction]&lt;br /&gt;&lt;br /&gt;您是一个Python编程助手，负责协助用户编写代码。务必要仔细听取用户的要求，并严格按照要求执行。首先，将代码输出在一个代码块中。其次，详细阐述您的计划步骤，写出极为详尽的计划。编写干净、高效、易于维护的Python代码。确保代码不需要接受用户输入，有一个主函数以便可以直接运行。确保默认的读写文件为同目录下的'test_file.txt'。&lt;br /&gt;&lt;br /&gt;[Template]&lt;br /&gt;&lt;br /&gt;用户:&lt;br /&gt;&lt;br /&gt;[用户问题]&lt;br /&gt;&lt;br /&gt;助手:&lt;br /&gt;&lt;br /&gt;```python</pre></div> <div>[生成的代码]</div> <div><pre>```</pre></div> <div>[对代码的解释]</div>	
多轮对		"You could be any character, as decided by the user. You need to be very creative, your role could come from any	"你可能是任何角色，由用户决定。你需要非常具有创造性，你的角色可能

话 角 色 扮 演	/	time period, it could be from literature, or it could be a professional position, etc. Please play the role assigned by the user, conform to their character, and communicate with the user accordingly."	来自古今中外，可能来自文艺作品，也可能是专业的职位等。请你扮演好用户指派的角色，符合其人设，并于用户交流。"
数 学 推 理	"You are a math tutor who helps young students understand and solve mathematical problems. Provide step-by-step explanations and guidance for basic arithmetic. Use clear language to give instructions and answer."	<p>"[Instruction]</p> <p>You are a math tutor who helps students understand and solve mathematical problems. The range of problems covers probability, equations, plane geometry, and basic arithmetic. You must provide step-by-step explanations and guidance for the student. Use clear and detailed language to give instructions and answer.</p> <p>[Example 1]</p> <p>User:</p> <p>When a number is divided by 10, the remainder is 4. What is the remainder when twice the number is divided by 4?</p> <p>Assistant:</p> <p>To find the remainder when twice the number is divided by 4, we need to think about how remainders work in division.</p> <p>First, let's consider the original situation. When a number is divided by 10 and the remainder is 4, it means that the number can be written as <math>10n + 4</math>, where <math>n</math> is a whole number.</p> <p>Now, let's find out what happens when we double this number and divide by 4. If we double <math>10n + 4</math>, we get <math>20n + 8</math>.</p> <p>To determine the remainder when <math>20n + 8</math> is divided by 4, we divide <math>20n</math> by 4 and the remainder will be zero because 4 evenly divides into <math>20n</math>.</p> <p>The remainder is therefore the same as the remainder of 8 when it is divided by 4. When we divide 8 by 4, the quotient is 2 with no remainder. Therefore, the remainder when twice the number is divided by 4 is 0.</p> <p>[Example 2]</p> <p>User:</p> <p>A tech startup invests \$8000 in software development in the first year, and then invests half of that amount in software development in the second year. What's the total amount the startup invested in software development over the two years?</p> <p>Assistant:</p> <p>To find the total amount the startup invested in software development over two years, we need to add the amounts invested in the first and second years.</p> <p>Firstly, In the first year, the startup invested \$8000.</p> <p>Secondly, In the second year, they invested half of that amount, which is <math>\\$8000/2 = \\$4000</math>.</p> <p>To find the total investment, we add \$8000 and \$4000:</p>	<p>"[Instruction]</p> <p>您是一名数学导师，帮助学生理解和解决数学问题。问题的范围涵盖概率、方程、平面几何和基本算术。您必须为学生提供逐步解释和指导。请使用清晰和详细的语言给出说明和最终答案。</p> <p>[Example 1]</p> <p>用户：</p> <p>当一个数除以10，余数是4。当这个数的两倍除以4时，余数是多少？</p> <p>助理：</p> <p>当我们将数字乘以2然后除以4时，我们需要思考余数在除法中的工作原理。</p> <p>首先，让我们考虑原始情况。当一个数字被10除，余数是4时，这意味着这个数字可以被写成<math>10n + 4</math>的形式，其中<math>n</math>是一个整数。</p> <p>现在，让我们找出将这个数字乘以2然后除以4会发生什么。如果我们将<math>10n + 4</math>乘以2，我们得到<math>20n + 8</math>。</p> <p>为了确定<math>20n + 8</math>被4除的余数，我们将<math>20n</math>除以4，余数将是零，因为4可以整除<math>20n</math>。</p> <p>因此，余数与<math>8 \div 4</math>的余数相同。当我们将8除以4时，商为2，没有余数。因此，当这个数的两倍被4整除时的余数为0。</p> <p>因此，余数为0。</p> <p>[Example 2]</p> <p>用户：</p> <p>一家科技初创公司在第一年投入8000美元进行软件开发，然后在第二年投入该金额的一半进行软件开发。创业公司在两年内投入软件开发的总金额是多少？</p> <p>助理：</p> <p>要得到创业公司在两年内投入软件开发的总金额，我们需要把第一年和第二年投入的金额加在一起。</p> <p>首先，在第一年，创业公司投入了8000美元。</p> <p>其次，在第二年，他们投入了一半的金额，也就是8000美元的一半，也就是4000美元。</p>



		<div>\$8000 + \$4000 = \$12000.</div> <div>Therefore, the total amount the startup invested in software development over the two years is \$12000."</div>	<div>要得到总投资，我们把8000美元和4000美元加起来：8000 + 4000 = 12000美元。</div> <div>因此，创业公司在两年内投入软件开发的总金额为12000美元。"</div>
文案撰写	/	"You are a creative and helpful writing assistant, helping users write article for various purposes. The article topics are very diverse, therefore you must be very creative. If the topic is professional, write in-depth article professionally and accurately. If the topic is creative and entertaining, make it lively and fun, and add some unexpected twists to surprise the readers. Write as detailed as possible."	"你是一个富有创意和乐于助人的写作助手，帮助用户撰写各种文章以满足不同的需求。文章的主题非常多样化，因此你必须非常富有创造力。如果主题是专业的，就要以专业和准确的角度来撰写深入的文章。如果主题是创意和娱乐性的，就要让文章生动有趣，加入一些出人意料的转折来惊喜读者。尽可能详细地写作。"

## 项目细节说明

### 文件结构

需保持如下文件结构：

```
.
├── data
│   ├── judge-prompts.jsonl
│   └── pj3-test.jsonl
├── response
│   ├── code_snippets
│   │   └── # 代码生成任务 pass@3 测试时产生的 Python 文件 (.py)
│   ├── raw_responses
│   │   └── # 本次请求模型的响应 (.json)
│   └── raw_responses_backups
│       ├── default_prompt
│       │   └── # default prompt 下各任务各模型响应记录 (.json)
│       └── final_prompt
│           └── # final prompt 下各任务各模型响应记录 (.json)
├── result
│   ├── gpt_raw_evals
│   │   └── # 本次评估的响应 (.json)
│   ├── gpt_raw_evals_backups
│   │   ├── default_prompt
│   │   │   └── # default prompt 下各任务各模型评估记录 (.json)
│   │   └── final_prompt
│   │       └── # final prompt 下各任务各模型评估记录 (.json)
│   └── reuslt.ipynb
└── src
    ├── coding.py
    ├── main.py
    ├── math_analysis.py
    ├── roleplay.py
    ├── utils.py
    └── writing.py
```

### 数据说明

项目提供了 `seed = 2024`（代码生成任务中三个 pass 下 `seed = 2023, 2024, 2025`）下，各任务、各模型在 default system prompt、final system prompt 下的响应、gpt-3.5-turbo 的评价。

- 模型响应：
  - `./response/raw_responses_backups/default_prompt`。
  - `./response/raw_responses_backups/final_prompt`。
- gpt-3.5-turbo 的评价：
  - `./results/gpt_raw_evals_backups/default_prompt`。
  - `./results/gpt_raw_evals_backups/final_prompt`。

快速直接查看评分（包括代码生成任务的 **pass@3** 指标）：

```
python src/main.py --task TASK --model MODEL --eval --debug
```



将查看存放在 `./results/gpt_raw_evals` 下对应任务、模型的评估结果的文件。因此，若想查看之前任何一次输出记录，可从上述文件夹中取出对应文件到 `./results/gpt_raw_evals` 再运行。

## 代码说明

项目提供了完整的代码，可以直接使用。

- **请求模型响应：**将通过 API 调用模型，question 范围为 `./data/pj3-test.jsonl` 中所有本任务的 question。响应将保存在 `./response/raw_responses`。

```
python src/main.py --task TASK --model MODEL --request
# Example
python src/main.py --task "math" --model "gpt-3.5-turbo" --request
```

- **查看模型响应：**格式化输出 `./response/raw_responses`。主要用于调试，意义不大。

```
python src/main.py --task TASK --model MODEL --request --debug
```

- **请求 gpt-3.5-turbo 评价（若代码生成任务，将包括 pass@3 测试）：**

```
python src/main.py --task TASK --model MODEL --eval
```

- **也可以自动串行运行：**注意，需要避免不同模型同时测试，会引起 gpt-3.5-turbo 请求超额（3 Request / min）。代码中只对同一模型的请求做了停等、二进制指数退避等处理，但未对不同模型同时测试做处理。

```
python src/main.py --task TASK --model MODEL --request --eval
```

- **删除临时代码文件：**代码生成任务请求 gpt-3.5-turbo 评价时，会在 `./response/code_snippets/MODEL` 下生成代码文件，创建子进程并运行。可以一键清空：

```
python src/main.py --task TASK --model MODEL --clean_files --clean_response_messages
```