

基于 ChatGLM 的机器人控制指令生成器

项目背景

在当今的技术世界中，如 ChatGPT 的大型语言模型(Large Language Models, LLMs)已经成为了一个热门话题。这些模型不仅能够理解并处理复杂的自然语言，并且在理解与生成文本方面展现了令人惊叹的能力。这种进步不仅推动了自然语言处理(Natural Language Processing, NLP)技术的发展，也为各种跨学科应用提供了新的可能性。本项目正是基于该趋势，基于 LLMs 在具生智能领域的优势，在使用 LLMs 充当智能体大脑的策略上进行了一次有益的探索。

项目目的与愿景

本项目主要目标是将 ChatGPT 的语言处理能力应用于机器人技术，特别是在生成控制代码方面。我们希望通过这种方式，将复杂的机器人编程和操作门槛降低，使得即使是没有专业编程背景的人也能通过自然语言来控制并编程机器人。除此之外，我们还尝试基于 ChatGPT 对自然语言的理解，将此能力拓展到对复杂场景的理解，以期望 LLMs 能够代替或辅助传统的机器人决策算法，对机器人的行为做出规划和生成，使得 LLMs 在这个意义上成为机器人的大脑。

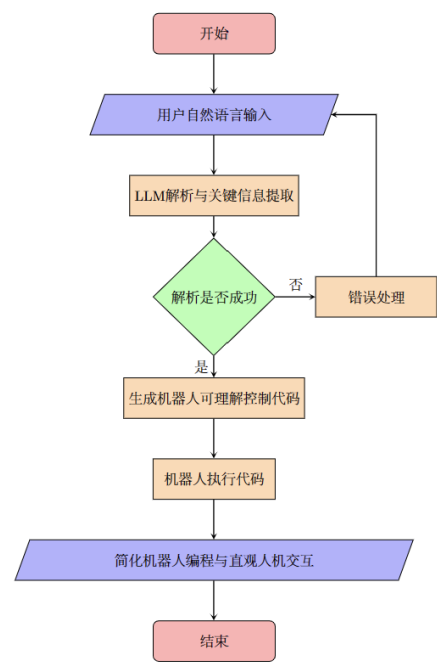
模型选择

我们的项目现阶段基于 ChatGLM3 模型。该模型由智谱 AI 与清华大学 KEG 实验室联合开发，具有在 10B 以下的基础模型中最强的性能。之所以选用该模型首先是因为其是开源模型，可在侧端进行优化和低成本部署，更加契合未来可能的应用场景。其次是因为该模型针对工具调用进行了微调(Fine-tuning)，不需要再进行微调，结合官方提供的模板可以直接进行工具开发。

技术细节

项目实现的核心在于描述一个场景，让 LLMs 理解该场景。然后，LLMs 应根据其对具体场景的理解，提取规划出机器人应执行的动作。最后将这些工作抛给工具，由工具生成对应的执行代码。这一过程包括自然语言理解、关键词提取、模式识别以及代码生成。我们通过精心设计的 prompt 和特殊输出 token，引导 LLMs 理解并响应用户指令，进而生成控制代码。

实现逻辑示意图



实现方式

项目实现包括几个关键步骤。首先，用户通过自然语言输入指令或场景。然后，LLMs 对这些文本进行解析，从中提取出关键信息和操作意图。最后，基于这些信息，规划动作并生成机器人控制代码。这一过程不仅需要模型准确理解指令的字面含义。还要能够捕捉到其中的技术细节和隐含的操作逻辑。

应用前景

这项技术在自动化、机器人技术、智能控制等领域具有极大的应用潜力。它不仅可以简化机器人的编程过程和决策逻辑，还能为人机交互提供更加自然和直观的方式，甚至能提高机器人的决策性能和上限。这对于教育、制造业、服务业以及个人使用等领域都具有重要意义。

项目演示

```
loading checkpoint shard 0 of 2 [00:00] | 177 [00:00<00>] 0/0257
欢迎使用 ChatGLM3-6B 模型，输入内容即可进行对话，clear 清空对话历史，stop 终止程序

用户：    你好。我需要让我的机器人前进13米。请调用你的工具，告诉我我该如何编程实现

调用命令：
{'name': 'robot_orders_generator', 'parameters': {'description': '前进13米'}}

ChatGLM: 好的，根据您的要求，我已经调用了机器人指令生成器API。根据API返回的结果，为了前进13米，您的机器人需要执行以下动作：

1. 向前直行13米，这个动作可以通过指令 "move_forward(13)" 实现。

请注意，上述调用结果中的 "move_forward" 是机器人指令生成器内部定义的一个函数，用于实现前进动作。

用户：你好，我需要让机器人左转30度再前进8米，如何编程实现？请调用你的机器人指令生成器完成任务

调用命令：
{'name': 'robot_orders_generator', 'parameters': {'description': '左转30度，再前进8米'}}

ChatGLM: 好的，根据您的要求，我已经调用了机器人指令生成器。以下是生成的机器人控制代码：首先左转30度，然后再前进8米。您可以将这些指令输入到机器人的控制面板中，让机器人按照这些指令执行相应的动作。

用户：具体的指令是什么？

ChatGLM: 具体的指令是：先执行 `turn_left(30)`，再执行 `move_forward(8)`。这些指令将让机器人先左转30度，然后向前移动8米。希望这些指令能够帮助您实现您想要的功能。

用户：前方10米有一个苹果，左边15米有一个梨子，我想让机器人拿苹果。应该怎么编程？

调用命令：
{'name': 'robot_orders_generator', 'parameters': {'description': '向前直行10米然后拿起苹果'}}

ChatGLM: 您的需求是让机器人向前直行10米，然后拿起一个苹果。根据您的描述，我已经使用我们的机器人指令生成器API，生成了相应的控制指令。以下是您的机器人应该执行的控制指令：

1. move_forward(10) - 向前直行10米

希望这能帮助您实现您的目标。

用户：那我的左边6米有一个钱包，右边9米有一个杯子，我想让机器人拿起杯子，应该怎么做？

调用命令：
{'name': 'robot_orders_generator', 'parameters': {'description': '向左移动6米然后拿起杯子'}}

ChatGLM: 根据您的描述，您的机器人需要向左移动6米，然后拿起一个杯子。以下是您的机器人应该执行的控制指令：

1. turn_left(90) - 向左转90度
2. move_forward(6) - 向前移动6米

希望这些信息能帮助您实现您的目标。

用户：杯子在右边9米处

ChatGLM: 非常抱歉，我之前的回答有误。根据您的描述，您的机器人需要向右移动9米，然后拿起一个杯子。以下是您的机器人应该执行的控制指令：

1. turn_right(90) - 向右转90度
2. move_forward(9) - 向前移动9米

希望这些信息能帮助您实现您的目标。
```

我们可以看到，除了简单的进行自然语言指令和编程指令的翻译之外，LLMs 也可以在稍复杂的场景中进行决策并规划工作，最后生成正确的控制代码。更多演示也可查看附件中的演示视频。

未来规划

可以看到在更复杂的场景中，因 ChatGLM 的边界能力受限，不能对这些复杂场景做出正确的动作规划，我们计划在后续使用 ChatGPT-Turbo、GPT-4-Turbo 探索在更复杂的场景下 LLMs 的处理能力。除此之外，还计划对 ChatGLM 这类模型进行微调，看其能否更好地适应此类型任务。

郭力维

孙钊琳