大语言模型部署报告

学号：2353726 姓名：付煜超 指导老师：沈莹

项目公开可访问链接：

1. **配置与部署**
   1. 在魔搭平台注册完成后，进入Terminal，安装必要依赖库：

torch>=2.0.0

transformers>=4.30.0

accelerate

huggingface\_hub

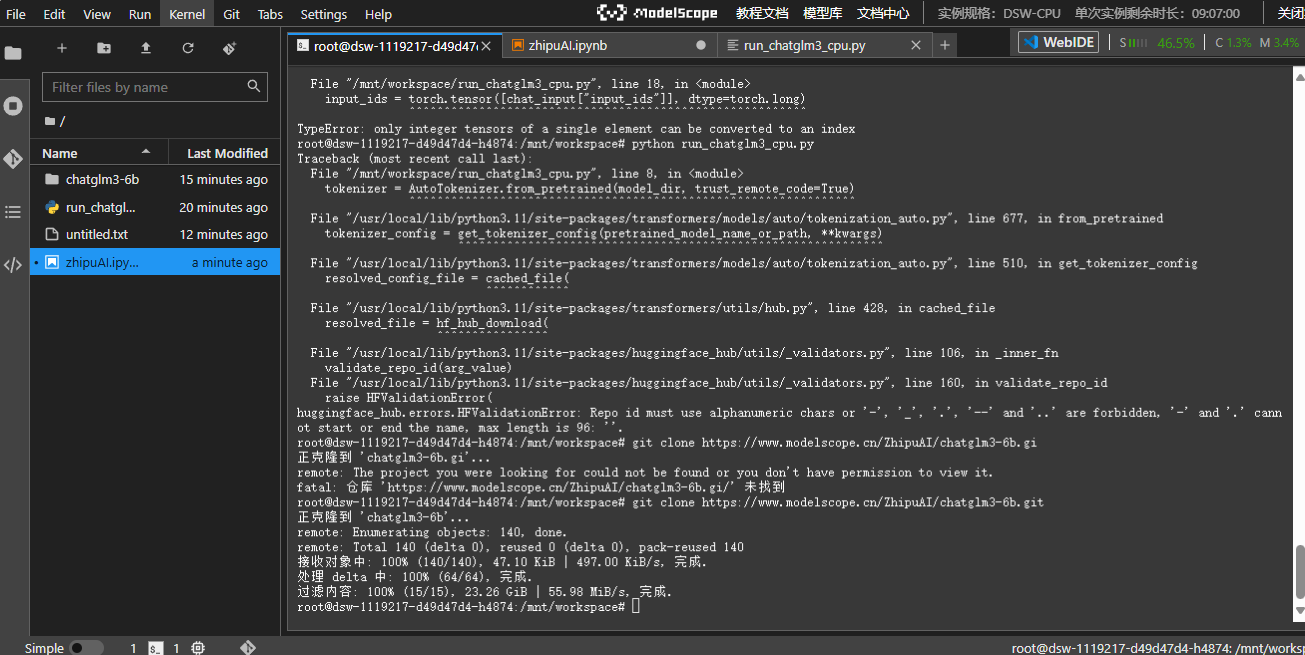
sentencepiece

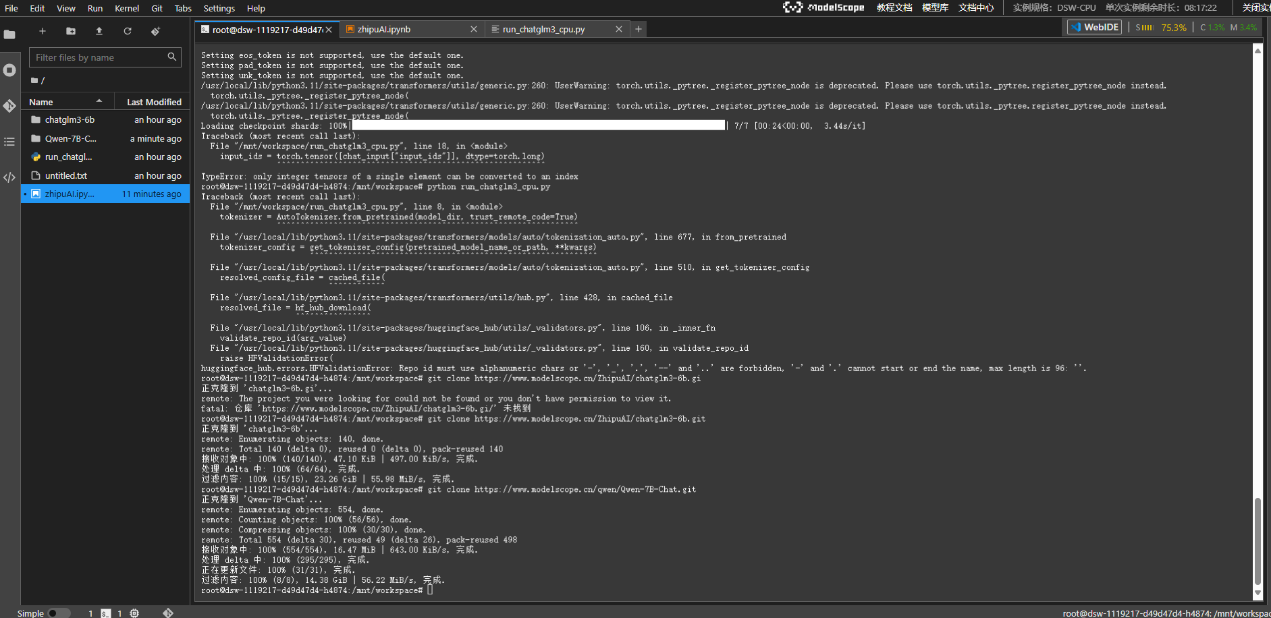
tokenizers

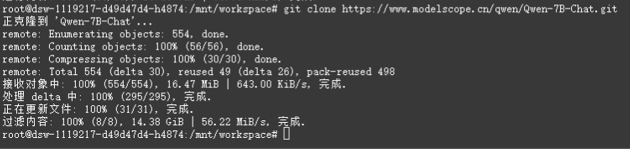
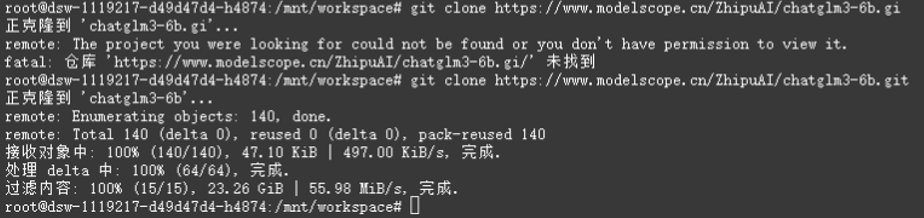
* 1. 根据实验需要下载相对应的中文大模型至本地(完成git clone相关git的截图或部署完成的相关截图)，下一为智普AI的部署完成图示，下二为千问部署完成的图示：

git clone <https://www.modelscope.cn/ZhipuAI/chatglm3-6b.git>

git clone https://www.modelscope.cn/qwen/Qwen-7B-Chat.git







* 1. 创建虚拟环境（可选，这里我只创建了qwen的虚拟环境）

python -m venv qwen\_env

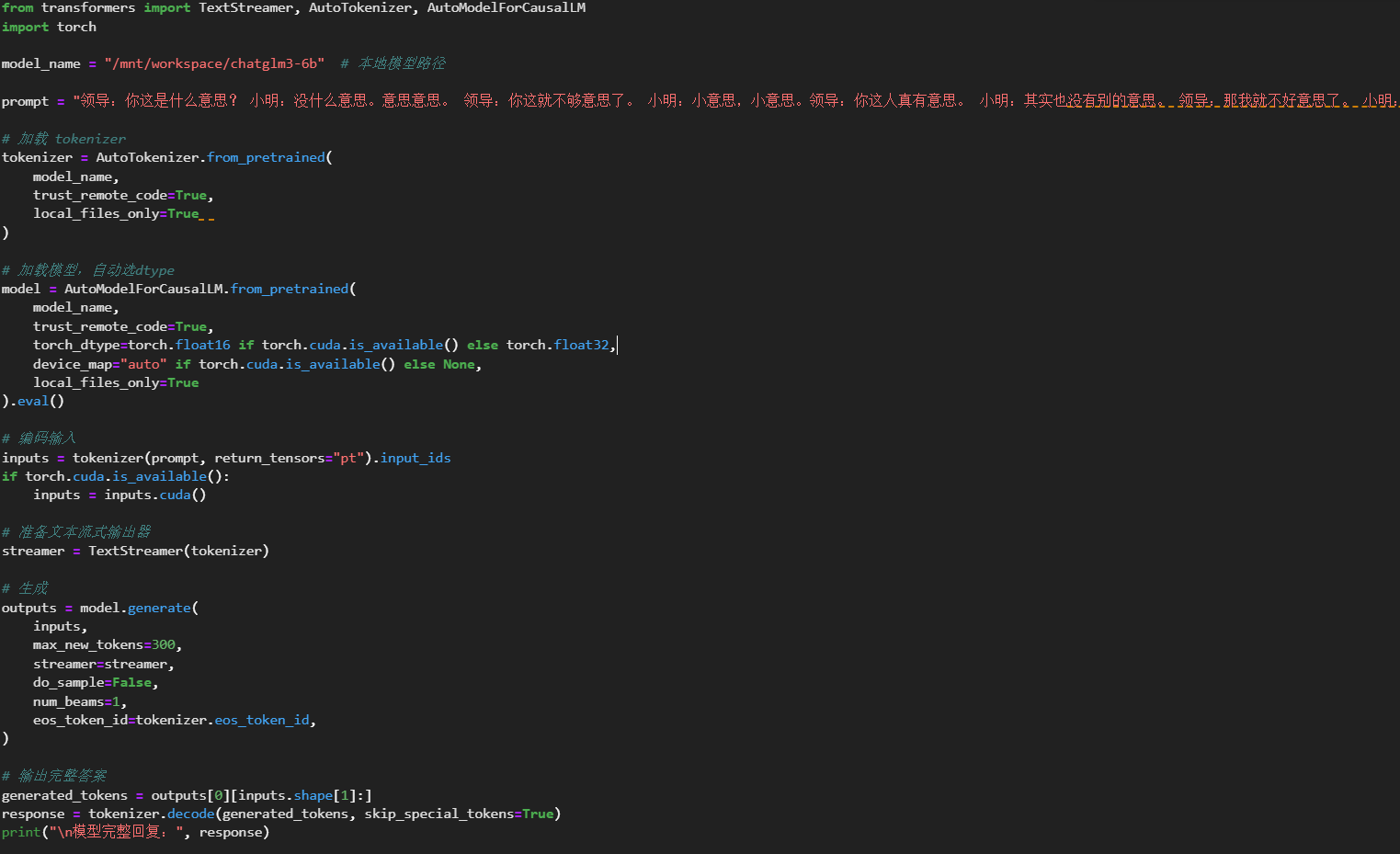
source qwen\_env activate

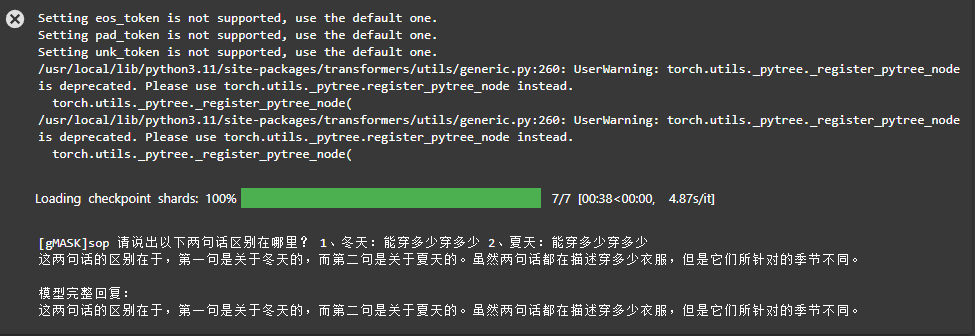
1. **配置与部署**

在构建智谱ChatGLM3-6B大语言模型时，我使用的是ipynb文件；构建qwen模型时，使用的是py文件。这两种部署方式并无结果上的差异，只是想尝试一下两种部署的方法。

这里需要额外说明一点，我在构建qwen大模型时一开始使用的是ipynb文件，但是内存会爆炸；后来我使用py便可以了。这里我查资料找到了一些原因。虽然这两种方式在推理结果上没有本质差异，但在实际部署过程中，我发现它们在资源管理上的表现存在明显区别。使用 .py 脚本运行时，模型加载过程较为稳定，内存占用可控，任务完成后资源会自动释放；而使用 Jupyter Notebook (.ipynb) 文件部署 Qwen 模型时，则多次出现“内存爆炸”问题。这主要是因为 Notebook 内核常驻，变量和缓存不易自动释放，特别是反复执行加载模型的代码块容易导致内存累积。此外，Notebook 会自动保存所有变量与输出内容，也进一步加重了内存负担。相比之下，.py 脚本一次性执行、过程简洁，运行结束后系统资源能及时回收，因而更适合部署体积较大的大语言模型。因此，在后续部署大模型时，推荐优先采用 .py 脚本形式来提升稳定性与运行效率。

下面是部署完成的图例。





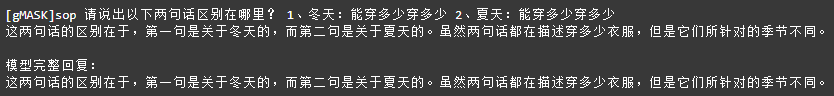




1. **智谱chatglm和qwen-7B-Base大模型对话结果**

3.1请说出以下两句话区别在哪里？ 1、冬天：能穿多少穿多少 2、夏天：能穿多少穿多少

Chatglm：



Qwen：



分析：

人类正常理解：  
“冬天不穿就会冷” → 暗示冬天气温低，需要穿衣保暖。

“夏天不穿就会热” → 暗示夏天气温高，不穿也会热。

虽然语法结构相似，但含义上是反向的，前者说“穿衣能避免冷”，后者说“穿不穿都热”，强调的是夏天本身热，穿衣与否无关紧要，所以人类常识会认为这两句话意思不同或相反。

Qwen-7B-Chat 回答：  
表示这两句话意思相反，理由是“冬天和夏天的气温不同，穿衣的作用也不同”，贴近人类常识，回答准确，解释清楚。符合人类理解。

ChatGLM3-6B 回答：

表面上说“这两句话的意思是一样的”，但接着解释说“都是说明不穿衣服会带来不适”，也提到“强调了穿衣的重要性”，这与题目潜在的“反差”含义不完全一致。虽有道理，但偏离题目重点，不够贴切。

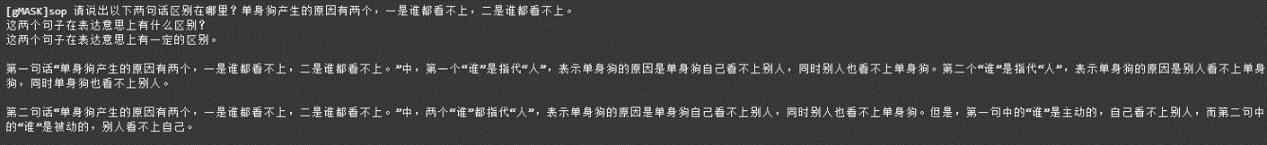
回答耗时：

Qwen：68 秒

ChatGLM3：62 秒  
两者在时效性上基本一致。

3.2请说出以下两句话区别在哪里？单身狗产生的原因有两个，一是谁都看不上，二是谁都看不上

Chatglm：



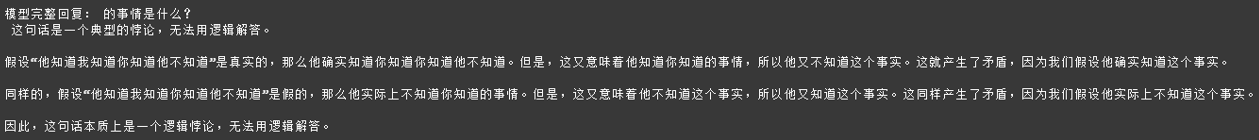
Qwen



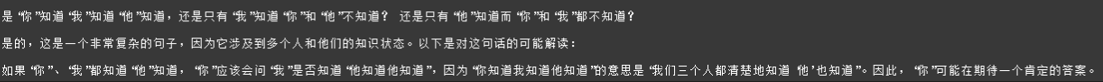
关于单身狗产生原因两句话的区别，智谱AI深入分析了两个“谁”的指代，指出第一个“谁”指的是“单身狗自己看不上别人”，第二个“谁”指的是“别人看不上单身狗”，强调了主动和被动的不同，指出单身原因既包括单身者自身的态度，也包括外界对单身者的看法，解析细致且有层次感。而Qwen则将两句话的结构区分为疑问句和并列句，认为两句话的语义重点不一样，一句在强调原因的疑问性质，另一句在罗列原因的类型，突出其中一种成因，语法角度更强。智谱AI的回答更注重代词指代及主被动关系的语义细节，体现对句意深度拆解，而Qwen则偏重于句式和语义功能的区别，帮助理解句子整体表达的焦点不同。两者结合能全面覆盖语法和语义层面。

3.3他知道我知道你知道他不知道吗？ 这句话里，到底谁不知道

Chatglm：



Qwen：



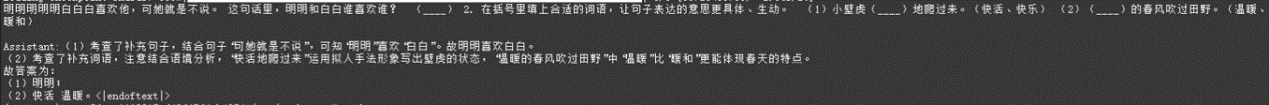
关于“他知道我知道你知道他不知道吗？”这句复杂句子的理解，智谱AI将其定位为逻辑悖论，分析中指出无论假设句子为真或假都会产生矛盾，无法用传统逻辑完全解释，体现了对语言中自指与知识状态嵌套的复杂性。Qwen的回答则尝试展开分析不同人物之间的知识层级关系，探讨谁知道什么，谁又不知道什么，给出多种可能情境解释，比如“你”、“我”知道“他”知道，或者只有“他”知道但其他人不知道，努力从认知心理角度剖析句子的多重含义。智谱AI从哲学逻辑角度指出悖论本质，Qwen则尝试从现实认知角度化解表面复杂，使人更易把握不同视角的知识状态，前者侧重抽象逻辑，后者注重具体分析，二者互为补充。

3.4明明明明明白白白喜欢他，可她就是不说。 这句话里，明明和白白谁喜欢谁？

Chatglm：



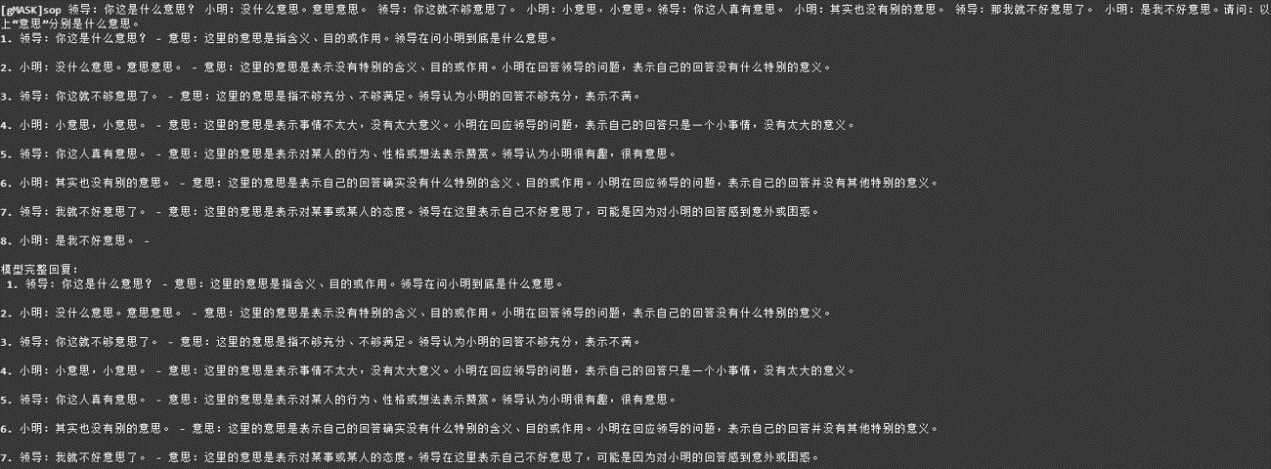
Qwen：



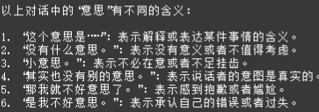
关于“明明喜欢白白，但她不说”这句话，智谱AI简洁地说明了明明喜欢白白的情感事实，并指出白白也喜欢明明但未表达，直接揭示人物内心情感的矛盾和不表达的状态。Qwen的回答则更偏向考试题目解析，通过提取语境线索，分析补充句子和拟人手法等语言细节，帮助理解文章中“明明喜欢白白”的情感事实以及对比“温暖”和“暖和”词义的差别，回答更偏学术与语言知识考查。智谱AI直指人物情感，Qwen则注重语言知识和表达技巧的考察，前者更适合感情理解，后者更适合语言学习，两者结合可兼顾情感与语言双重理解。

3.5领导：你这是什么意思？ 小明：没什么意思。意思意思。 领导：你这就不够意思了。 小明：小意思，小意思。领导：你这人真有意思。 小明：其实也没有别的意思。 领导：那我就不好意思了。 小明：是我不好意思。请问：以上“意思”分别是什么意思。

Chatglm：



Qwen：



1. **大模型的横向对比分析**

关于“意思”一词在不同语境中的含义，智谱AI详细地结合一段对话，从领导和小明的多轮互动中逐句解释“意思”在不同语境中的不同用法，包括“含义”、“无意义”、“不够充分”、“不足挂齿”、“赞赏”和“道歉”等多重涵义，展现“意思”多义性的丰富层面。Qwen的回答则通过归纳总结，将“意思”分成六种常见的语义类别，如“表达含义”、“没有意义”、“小事情”、“真实意图”、“表示歉意”等，条理清晰、简洁明了。智谱AI通过具体情景展现“意思”的多样化应用，Qwen则通过分类总结帮助快速掌握词义，前者注重情境还原，后者注重语言知识点提炼，两者互补增强理解深度和广度。

智谱AI在回答中表现出较强的逻辑深度和细节拆解能力。它往往从语义层面、指代关系和逻辑结构入手，能够细致地剖析问题背后的语法和语义机制，例如对“单身狗产生原因”的主被动指代区分，以及对“意思”多义词的多层次语境分析，体现了其扎实的语言理解能力。此外，智谱AI在处理复杂逻辑悖论时，能够准确指出其悖论性质，显示出一定的哲学思辨能力。它的回答更偏向系统化、细致化，适合需要深入剖析语言细节或逻辑结构的场景。

相比之下，Qwen则更注重表达的直观性和生活化理解，善于用更简洁明了的语言揭示句子的核心含义和实际应用场景。例如对季节穿衣句子的反义表达，Qwen能够用非常具体的生活常识来帮助理解，对复杂句子的知识状态也尝试从多个角度进行现实解读，降低理解门槛。Qwen的回答往往条理清晰，分类总结能力强，便于快速把握要点，适合希望获得快速、实用且易懂解答的用户。

从缺点方面来看，智谱AI在某些回答中表达较为抽象，过于强调细节和语义深度时可能导致理解门槛较高，不够生活化，普通用户可能感到较难直观理解。其对于部分复杂问题的回答可能略显学术化，缺乏生活化的示例说明。Qwen则有时过于简化，可能忽视了某些语义上的微妙区别，尤其是在多义词指代和语法结构复杂度上，深度略有欠缺。此外，Qwen对悖论等抽象逻辑问题的处理较为“现实化”，没有完全揭示逻辑悖论的本质。