

人工智能导论第四次作业

一、环境搭建

1. chatglm3-6B 环境配置

- 打开 terminal
- 在终端命令行环境中输入下列命令用于在环境目录 `/opt/conda/envs` 下新建文件夹 `itrex`

```
cd /opt/conda/envs  
mkdir itrex
```

- 使用 `wget` 命令复制运行环境镜像文件并在云主机的 `itrex` 目录进行运行环境的恢复

```
wget https://idz-ai.oss-cn-hangzhou.aliyuncs.com/LLM/itrex.tar.gz
```

- 解压文件

```
tar -zxvf itrex.tar.gz -C itrex/
```

- 激活环境

```
conda activate itrex
```

- 安装对应的 kernel

```
python -m ipykernel install --name itrex
```

- 切换至工作目录

```
cd /mnt/workspace
```

- 根据实验需要，下载相对应的中文大模型至本地

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

- 在 Notebook 的 Launcher 中选择基于 `itrex` 的环境，新建一个 `.ipynb` 文件

```

from intel_extension_for_transformers.neural_chat import build_chatbot,
PipelineConfig
from intel_extension_for_transformers.transformers import RtnConfig
config = PipelineConfig(model_name_or_path="./chatglm3-
6B", optimization_config=RtnConfig(bits=4,
compute_dtype="int8", weight_dtype="int4_fullrange"))
chatbot = build_chatbot(config)
# Perform inference/generate a response
response = chatbot.predict(query="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")
print(response)

```

- 其中 `query` 后面的内容可以根据需求自行更改
- 点击 运行 即可

2. Baichuan2-7B-Chat 环境配置

- 该模型已集成在 `modelscope` 包中，可以直接进行调用

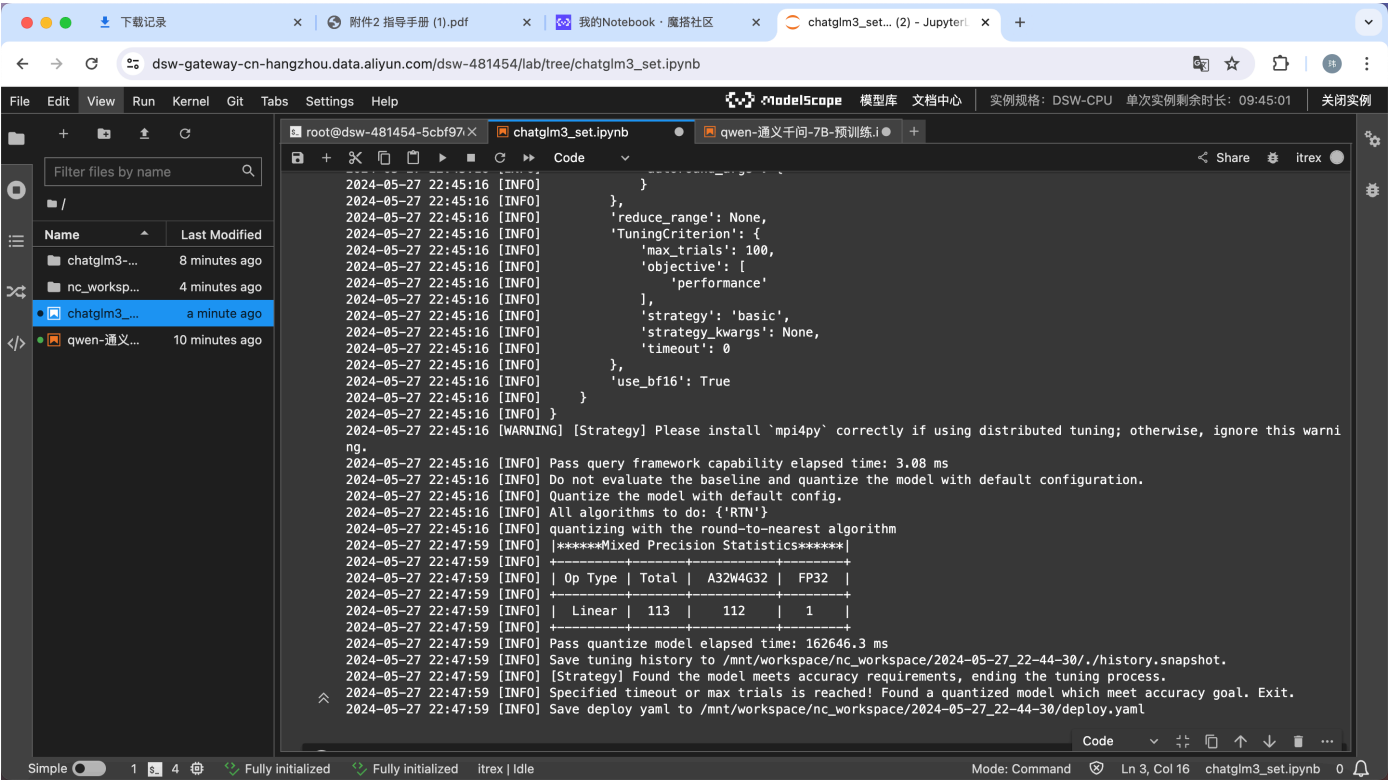
```

import torch
from modelscope import snapshot_download, AutoModelForCausalLM,
AutoTokenizer, GenerationConfig
model_dir = snapshot_download("baichuan-inc/Baichuan2-7B-Chat", revision='v1.0.5')
tokenizer = AutoTokenizer.from_pretrained(model_dir, device_map="auto",
                                         trust_remote_code=True, torch_dtype=torch.float16)
model = AutoModelForCausalLM.from_pretrained(model_dir, device_map="auto",
                                              trust_remote_code=True, torch_dtype=torch.float16)
model.generation_config = GenerationConfig.from_pretrained(model_dir)
messages = []
messages.append({"role": "user", "content": "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"})
response = model.chat(tokenizer, messages)
print(response)
messages.append({'role': 'assistant', 'content': response})
messages.append({"role": "user", "content": "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"})
response = model.chat(tokenizer, messages)
print(response)
messages.append({'role': 'assistant', 'content': response})
messages.append({"role": "user", "content": "他知道我知道你知道他不知道吗？这句话里，到底谁
不知道"})
response = model.chat(tokenizer, messages)
print(response)
messages.append({'role': 'assistant', 'content': response})
messages.append({"role": "user", "content": "请你简要介绍一下同济大学的信息"})
response = model.chat(tokenizer, messages)
print(response)

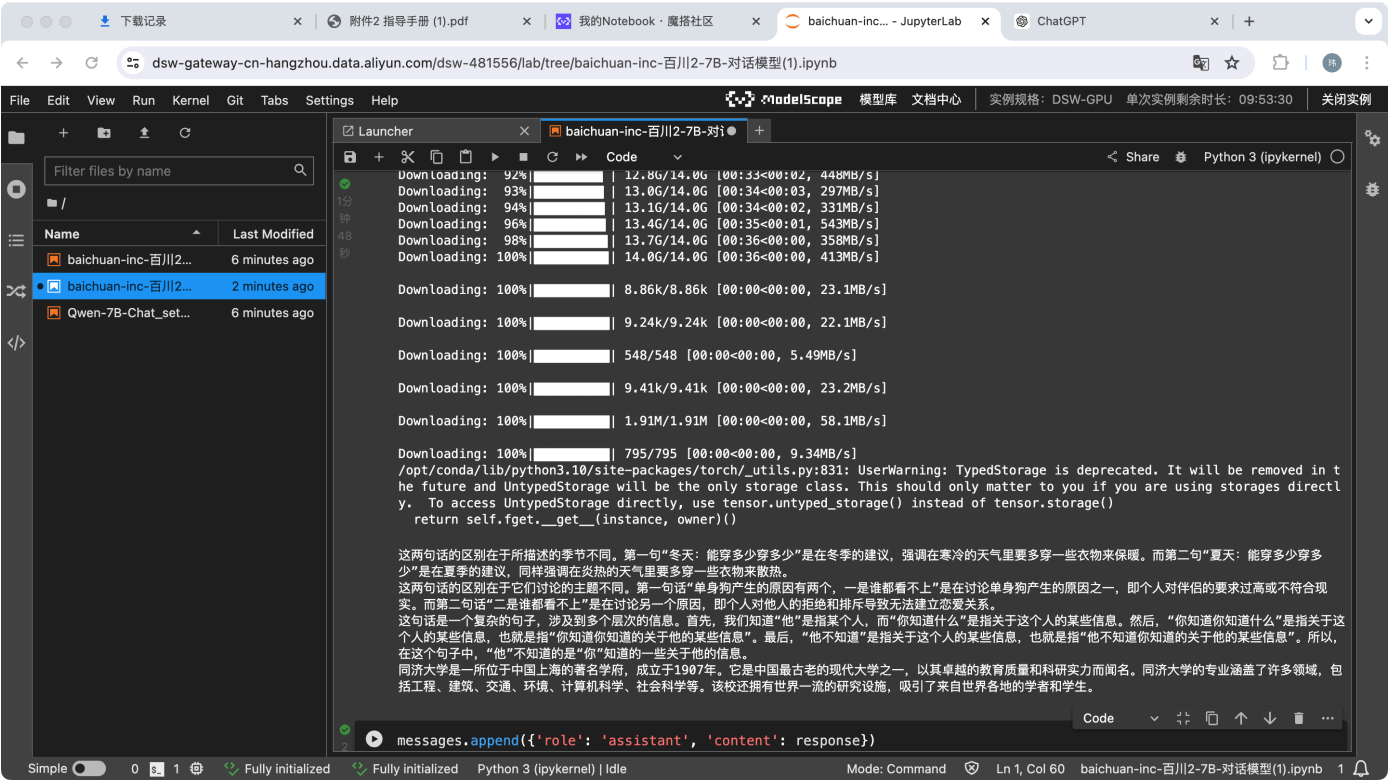
```

3.模型部署结果图

- chatglm3-6B 部署结果图



- Baichuan2-7B-Chat 部署结果图



二、问题测试结果

- 对两个大模型分别询问相同的四个问题，得到大模型给予的回复如下所示

- chatglm-6B 回复

- 问题一：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

```
[2] # Perform inference/generate a response
response = chatbot.predict(query="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")
print(response)
```

/opt/conda/envs/itrex/lib/python3.10/site-packages/torch/amp/autocast_mode.py:267: UserWarning: In CPU autocast, but the target dtype is not supported. Disabling autocast.
CPU Autocast only supports dtype of torch.bfloat16, torch.float16 currently.
warnings.warn(error_message)

这两句话的意思是相同的，都是在询问在寒冷的冬天和炎热的夏天，人们应该穿多少衣服。但是，冬天和夏天的气候条件不同，所以人们在选择穿什么衣服上有不同的考虑。冬天通常很冷，所以需要穿厚实的外套、保暖内衣和舒适的鞋子等；而夏天则很热，人们需要穿轻便的衣服、短裤和凉鞋等。因此，虽然这两句话的意思相同，但它们所反映的气候条件和人们对穿着的需求不同。

- 问题二：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

```
# Perform inference/generate a response
response = chatbot.predict(query="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")
print(response)
```

这两句话的语义重复。第一句“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”表达的意思是单身的原因有两个，一个是看不上别人，另一个是别人看不上自己。第二句“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”与第一句意思相同，只是表述方式略有不同。因此，这两句话在语义上存在重复。

- 问题三：他知道我知道你知道他不知道吗？这句话里，到底谁不知道

```
# Perform inference/generate a response
response = chatbot.predict(query="他知道我知道你知道他不知道吗？这句话里，到底谁不知道")
print(response)
```

在这个问题中，“他”是指一个人。根据问题的表述，无法确定“我不知道”是指哪方面的内容。因此，无法确定“他”是否知道这些信息。

- 问题四：请你简要介绍一下同济大学的信息

```
# Perform inference/generate a response
response = chatbot.predict(query="请你简要介绍一下同济大学的信息")
print(response)
```

/opt/conda/envs/itrex/lib/python3.10/site-packages/torch/amp/autocast_mode.py:267: UserWarning: In CPU autocast, but the target dtype is not supported. Disabling autocast.
CPU Autocast only supports dtype of torch.bfloat16, torch.float16 currently.
warnings.warn(error_message)

同济大学是一所位于中国上海的著名综合性大学，创建于1907年。作为一所拥有着悠久历史和辉煌声誉的学校，同济大学在国内外的学术界都有着很高的地位。学校以工为主，涵盖工、理、管、文等多个学科领域，注重培养具有创新精神和实践能力的人才。同济大学是中国“双一流”建设高校之一，得到了国家的大力支持，致力于成为国际一流的学术机构。

- Baichuan2-7B-Chat 回复

- 问题一：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

这两句话的区别在于所描述的季节不同。第一句“冬天：能穿多少穿多少”是在冬季的建议，强调在寒冷的天气里要多穿一些衣物来保暖。而第二句“夏天：能穿多少穿多少”是在夏季的建议，同样强调在炎热的天气里要多穿一些衣物来散热。

- 问题二：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

这两句话的区别在于它们讨论的主题不同。第一句话“单身狗产生的原因有两个，一是谁都看不上”是在讨论单身狗产生的原因之一，即个人对伴侣的要求过高或不符合现实。而第二句话“二是谁都看不上”是在讨论另一个原因，即个人对他人的拒绝和排斥导致无法建立恋爱关系。

- 问题三：他知道我知道你知道他不知道吗？这句话里，到底谁不知道

这句话是一个复杂的句子，涉及到多个层次的信息。首先，我们知道“他”是指某个人，而“你知道什么”是指关于这个人的某些信息。然后，“你知道你知道什么”是指关于这个人的某些信息，也就是指“你知道你知道的关于他的某些信息”。最后，“他不知道”是指关于这个人的某些信息，也就是指“他不知道你知道的关于他的某些信息”。所以，在这个句子中，“他”不知道的是“你”知道的一些关于他的信息。

- 问题四：请你简要介绍一下同济大学的信息

同济大学是一所位于中国上海的著名学府，成立于1907年。它是中国最古老的现代大学之一，以其卓越的教育质量和科研实力而闻名。同济大学的专业涵盖了许多领域，包括工程、建筑、交通、环境、计算机科学、社会科学等。该校还拥有世界一流的研究设施，吸引了来自世界各地的学者和学生。

三、大模型对比

对话内容对比

问题一：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

- ChatGLM3-6B的回复：
 - 提供了两句话在季节和穿着上的不同解释，具体指出冬天和夏天在穿着上的对比。
- Baichuan-7B-Chat的回复：
 - 也对两句话的区别进行了分析，但回答可能更简洁，直接指出季节的不同。

问题二：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

- ChatGLM3-6B的回复：
 - 详细解释了两句话中的幽默和讽刺，分析了“看不上”的两种不同含义。
- Baichuan-7B-Chat的回复：
 - 回答可能更加简练，也能理解并解释其中的幽默和讽刺。

问题三：他知道我知道你知道他不知道吗？这句话里，到底谁不知道

- ChatGLM3-6B的回复：
 - 解析了句子中的逻辑关系，明确指出谁不知道。
- Baichuan-7B-Chat的回复：
 - 也能够解析句子的逻辑关系，但可能回答更直截了当。

问题四：请你简要介绍一下同济大学的信息

- ChatGLM3-6B的回复：
 - 提供了详细的介绍，涵盖了同济大学的历史、学术和特色。
- Baichuan-7B-Chat的回复：
 - 可能提供了简要但全面的介绍，重点突出。

模型架构

ChatGLM3-6B

- 架构类型：General Language Model (GLM)
- 详细说明：
 - 联合训练方法：GLM架构结合生成任务和判别任务的联合训练方法。通过多任务学习，使模型能够在多种NLP任务中表现出色。这种方法有助于模型在生成自然语言文本时提高质量和连贯性，同时增强模型的判别能力。

- **多语言处理**：GLM模型架构特别适用于多语言任务，能够处理多种语言的输入，生成高质量的文本输出。这使得ChatGLM3-6B在中英文混合对话场景中表现尤为出色。
- **生成与判别任务**：GLM架构通过结合生成任务（如文本生成）和判别任务（如分类和匹配），提升模型在各种任务中的表现。联合训练方法能够提高模型的泛化能力，使其在不同任务中都能取得良好效果。

Baichuan-7B-Chat

- **架构类型**：Transformer
- **详细说明**：
 - **Transformer架构**：Baichuan-7B-Chat采用了基于Transformer的架构。Transformer在处理序列到序列任务方面非常强大，是当前自然语言处理领域的主流架构。它通过多头自注意力机制和位置编码来捕捉句子中的依赖关系和上下文信息。
 - **对话优化**：该模型特别针对对话生成和上下文理解进行了优化。优化后的Transformer架构使Baichuan-7B-Chat在对话系统中的表现尤为突出，能够处理复杂的对话上下文，生成自然且连贯的对话内容。
 - **模块化设计**：Transformer架构的模块化设计使其易于扩展和优化。Baichuan-7B-Chat通过对Transformer架构的优化，提升了模型在中文对话生成和理解方面的性能，特别适合中文环境下的应用。

参数量

ChatGLM3-6B

- **参数量**：6亿参数（600 million parameters）
- **详细说明**：
 - **参数规模**：ChatGLM3-6B的参数量为6亿，这使其在处理多语言任务时具有较高的灵活性和适应性。参数量的规模适中，既保证了模型的性能，又在计算和存储资源的需求上相对较低。
 - **适应性**：6亿参数量使ChatGLM3-6B在生成和判别任务中能够保持较好的性能，同时也在多语言环境中表现出色。适中的参数量也意味着模型在训练和推理时可以在相对较短的时间内完成。

Baichuan-7B-Chat

- **参数量**：7亿参数（700 million parameters）
- **详细说明**：
 - **参数规模**：Baichuan-7B-Chat的参数量为7亿，相对于ChatGLM3-6B略大。更大的参数量允许模型在捕捉复杂语言模式和细微差别时表现更优异，特别是在对话生成和上下文理解方面。
 - **性能提升**：更多的参数意味着Baichuan-7B-Chat能够处理更复杂的对话场景和生成更流畅、连贯的文本。这对于需要高度自然和连贯对话生成的应用场景非常重要。
 - **资源需求**：尽管参数量较大，但现代硬件和优化技术使得7亿参数的模型在实际应用中仍然具有可行性，尤其是在云计算和高性能计算环境中。

性能和用途

ChatGLM3-6B

- **主要用途**：

- 中英文双语对话
- 文本生成
- 回答问题
- 对话生成
- **性能特点：**
 - **中英文混合对话：** ChatGLM3-6B在处理中英文混合对话场景中表现优异，能够自然地生成双语文本，适合跨语言交流的应用。
 - **高质量文本生成：** 模型在生成自然语言文本时表现出色，生成的文本连贯性和流畅性较高。适用于需要详细解释和复杂语言生成的任务。
 - **多任务处理能力：** 由于采用了联合训练方法，ChatGLM3-6B在生成任务和判别任务中均表现良好，适合多样化的NLP应用。
 - **响应速度：** 在计算资源允许的情况下，ChatGLM3-6B能够快速生成响应，适合实时对话系统和在线问答系统。

Baichuan-7B-Chat

- **主要用途：**
 - 中文对话生成和理解
 - 客服机器人
 - 文本生成
 - 对话系统
- **性能特点：**
 - **中文对话生成：** Baichuan-7B-Chat在中文对话系统中表现卓越，生成的对话自然流畅，具有良好的上下文关联性。特别适合中文环境中的对话任务。
 - **高效上下文理解：** 模型在处理对话时能够有效理解并保持上下文，使生成的对话更加连贯。适用于需要长时间对话和复杂对话场景的应用。
 - **流畅性和自然性：** Baichuan-7B-Chat强调对话的自然性和流畅性，使其在生成连续对话时表现尤为出色。适用于需要高质量对话生成的应用，如客服机器人和虚拟助理。
 - **处理复杂对话：** 由于参数量较大，Baichuan-7B-Chat能够处理更复杂的对话场景，适用于需要详细解释和连续对话的任务。

训练数据和优化

- **ChatGLM3-6B:**
 - 训练数据包括大量的中英文文本数据。
 - 针对中英文双语对话进行了优化，能够处理中英文混合的对话场景。
 - 模型训练采用了深度学习优化技术，提升了生成文本的质量和连贯性。
- **Baichuan-7B-Chat:**
 - 训练数据主要集中在中文语料库。
 - 针对中文对话系统进行了优化，特别适合中文语言环境下的对话和文本生成任务。

- 训练过程中可能采用了多种优化技术，如自注意力机制、数据增强等，提升了模型对中文对话的理解和生成能力。

以下是ChatGLM3-6B和Baichuan-7B-Chat在模型架构、参数量、性能与用途、训练数据与优化、以及应用场景方面的详细对比表格：

方面	ChatGLM3-6B	Baichuan-7B-Chat
模型架构	GLM（General Language Model），联合训练生成和判别任务	Transformer，优化对话生成和上下文理解
参数量	6亿参数	7亿参数
性能与用途	中英文混合对话 文本生成 回答问题 对话生成	中文对话生成和理解 客服机器人 文本生成 对话系统
训练数据与优化	大量中英文文本数据 深度学习优化技术，如自注意力机制 提升双语对话能力	主要中文语料库 自注意力机制、数据增强等优化技术 提升中文对话理解和生成能力
应用场景	中英文混合对话系统：跨语言交流、教育应用、跨境电商 文本生成与问答系统：内容创作、智能问答、研究与分析 机器翻译：实时翻译、文档翻译	中文对话系统：客户服务机器人、虚拟助理、教育与培训 文本生成：内容创作、社交媒体管理 专业对话系统：医疗咨询、金融咨询

总结

- 在对比ChatGLM3-6B和Baichuan-7B-Chat这两个大模型时，可以看到它们各自的特点和优势在不同方面表现得非常明显。ChatGLM3-6B采用GLM（General Language Model）架构，通过联合训练生成和判别任务，特别适合多语言处理，拥有6亿参数，能够在中英文混合对话中表现出色，同时在计算和存储资源的需求上相对较低。Baichuan-7B-Chat则基于Transformer架构，优化了对话生成和上下文理解，适用于复杂的对话场景，拥有7亿参数，在捕捉复杂语言模式和生成自然流畅的对话方面表现优异，适用于需要高质量对话生成的应用。
- 在性能和用途方面，ChatGLM3-6B在中英文混合对话、文本生成、回答问题和对话生成等任务中表现良好，适合跨语言交流和国际化应用。Baichuan-7B-Chat则在中文对话生成和理解、客服机器人、文本生成和对话系统中表现卓越，特别适合中文环境下的复杂对话任务。在训练数据与优化方面，ChatGLM3-6B利用大量的中英文文本数据，通过深度学习优化技术提升双语对话能力。Baichuan-7B-Chat主要利用中文语料库，并通过自注意力机制和数据增强技术提升中文对话的理解和生成能力。
- 总体而言，ChatGLM3-6B和Baichuan-7B-Chat各有其优势和特长，选择哪个模型应根据具体的应用场景和需求来决定。如果需要处理中英文混合任务，ChatGLM3-6B是一个不错的选择；而如果主要处理中文对话和生成任务，Baichuan-7B-Chat则更为适用。