

llm_evaluation_project

作者：2351493 胡浩杰

项目公开可访问链接：<https://github.com/Jacky0903/LLM-Evaluation>

项目简介

本项目是对 ChatGLM3-6B 和 Qwen-7B-Chat 两个大语言模型在 8 个测试问题上的表现对比分析，从逻辑推理、数学能力、知识解释、创作风格等核心维度展开，并结合具体场景给出适用建议

环境搭建

- 在魔搭社区启动环境



- 本次实验我选择直接在root下直接操作
- 在执行前加一条命令检查 pip 是否能正常联网

```
pip install -U pip setuptools wheel
```

- 配置以下基础依赖

```
# 安装基础依赖（兼容 transformers 4.33.3 和 neuralchat）
pip install \
    "intel-extension-for-transformers==1.4.2" \
    "neural-compressor==2.5" \
    "transformers==4.33.3" \
    "modelscope==1.9.5" \
    "pydantic==1.10.13" \
    "sentencepiece" \
    "tiktoken" \
    "einops" \
    "transformers_stream_generator" \
    "uvicorn" \
    "fastapi" \
```

```
"yacs" \
"setuptools_scm"
# 安装 fschat (需要启用 PEP517 构建)
pip install fschat --use-pep517
```

依赖安装完成后：

```
File Edit View Run Kernel Git Tabs Settings Help ModelScope 教程文档 模型库 文档API 实例规格: DSW-CPU 单次实例训练时长: 09:52:18 关闭 WebIDE S 0% C 0% M 100%
root@dsw-1120009:5f6d884x ~ % + Filter files by name / Name Last Modified Requirement already satisfied: astro==17.3.0 in /usr/local/lib/python3.11/site-packages (from sichthttp>fschat) (25.3.0)
Requirement already satisfied: frozenlist==1.1.1 in /usr/local/lib/python3.11/site-packages (from sichthttp>fschat) (1.6.0)
Requirement already satisfied: multidict==7.0.x->4.5 in /usr/local/lib/python3.11/site-packages (from sichthttp>fschat) (6.4.3)
Requirement already satisfied: preparse==0.2.1 in /usr/local/lib/python3.11/site-packages (from sichthttp>fschat) (0.3.0)
Requirement already satisfied: pygments==2.13.0 in /usr/local/lib/python3.11/site-packages (from sichthttp>fschat) (2.13.0)
Requirement already satisfied: ioda==2.0 in /usr/local/lib/python3.11/site-packages (from yaml==2.0.x->1.17.0->fschat) (3.1.0)
Requirement already satisfied: starlette==0.47.0,->0.40.0 in /usr/local/lib/python3.11/site-packages (from fastapi>fschat) (0.46.2)
Requirement already satisfied: typing_extensions==4.8.0 in /usr/local/lib/python3.11/site-packages (from fastapi>fschat) (4.12.2)
Requirement already satisfied: anyio==3.x-3.6.2 in /usr/local/lib/python3.11/site-packages (from starlette==0.47.0,>0.40.0->fastapi>fschat) (4.9.0)
Requirement already satisfied: httpcore==0.18.0 in /usr/local/lib/python3.11/site-packages (from starlette==0.47.0,>0.40.0->fastapi>fschat) (0.18.0)
Requirement already satisfied: certifi in /usr/local/lib/python3.11/site-packages (from https>fschat) (2023.1.31)
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.11/site-packages (from https>fschat) (1.0.9)
Requirement already satisfied: h1l>=0.16 in /usr/local/lib/python3.11/site-packages (from httpcore==1.*->https>fschat) (0.16.0)
Requirement already satisfied: waverror from markdown[all]->fastapi
Collecting waverror
  Downloading https://mirrors.cloud.aliyuncs.com/pypi/packages/be/71/67393dab0c0504a8eccc53491986506585e6ca517recf653d3a/waverror-2.0.3.post3.tar.gz (137 kB)
Installing build dependencies ... done
Getting requirements to build wheel ... done
Installing backend dependencies ... done
Preparing metadata (pyproject.toml) ... done
Selecting legacyInstall (from markdown[all]->fastapi)
  Downloading https://mirrors.cloud.aliyuncs.com/pypi/packages/1d/6bd0384b527168af579857fd9e883d2384b1fa5ca5f3a37949cef/latex2mathml-3.78.0-py3-none-any.whl (73 kB)
Requirement already satisfied: charset-normalizer<4,>2 in /usr/local/lib/python3.11/site-packages (from requests>fschat) (3.4.1)
Requirement already satisfied: urllib3<3,->2.1.1 in /usr/local/lib/python3.11/site-packages (from requests>fschat) (2.3.0)
Requirement already satisfied: regex<2022.1.18 in /usr/local/lib/python3.11/site-packages (from tritonkit>fschat) (2024.11.6)
Requirement already satisfied: certifi in /usr/local/lib/python3.11/site-packages (from https>fschat) (0.1.8)
Collecting sigrate (from markdown[all]->fastapi)
  Downloading https://mirrors.cloud.aliyuncs.com/pypi/packages/84/15/64bc99579024a6767951883945a0293a1d8f850b0709e61abb644d11c/sigrate-1.4.3-py3-none-any.whl (67 kB)
Requirement already satisfied: six in /usr/local/lib/python3.11/site-packages (from waverror>waverror[all]->fastapi) (1.17.0)
Requirement already satisfied: pynal in /usr/local/lib/python3.11/site-packages (from waverror>waverror[all]->fastapi) (6.0.2)
Building wheel for waverror (from waverror>waverror[all]->fastapi) ... done
Created wheel for waverror: filename: waverror-2.0.3.post3-py3-none-any.whl size: 30194 sha256:c48d800034fffb3aef590097fc200c3c043e34234db0e1c58ff3f9e88
Stored in directory: /root/.cache/pip/wheels/30/07/02/3erfbw30194sha256c48d800034fffb3aef590097fc200c3c043e34234db0e1c58ff3f9e88
Successfully built waverror
WARNING: Error parsing dependencies of pytorch-lightning: * suffix can only be used with == or != operators
  torch (>=1.10.0)
Installing collected packages: sigrate, shorntud, nb3, markdown2, latex2mathml, waverror, fschat
Successfully installed fastchat-0.2.36 latex2mathml-3.78.0 markdown2-2.5.3 nb3-0.2.21 shorntud-1.0.13 sigrate-1.4.3 waverror-2.0.3.post3
WARNING: Running pip as the root user can result in broken permissions and conflicting behaviour with the system package manager, possibly rendering your system unusable. It is recommended to use a virtual environment instead.
  https://pip.pycodex.org/warnings/venv. Use the --root user-action option if you know what you are doing and want to suppress this warning.
root@dsw-1120009:5f6d884x ~ %
```

模型下载

- 切换到工作目录，下载相对应的中文大模型至本地

```
cd /mnt/workspace  
git clone https://www.modelscope.cn/zhipuAI/chatglm3-6b.git  
git clone https://www.modelscope.cn/qwen/Owen-7B-Chat.git
```

```
File Edit View Run Kernel Git Tabs Settings Help ModelScope 教程文档 模型库 文档中心 实例规格: DSW-CPU 单次实例剩余时长: 09:48:25  
WebIDE 46.5% C 0.0% M 100%  
cod@ds-w-1120097-777cf98 # +  
Filter files by name  
/ /  
Name Last Modified  
chatglm3-6b 2 minutes ago  
Download https://mirrors.cloud.aliyuncs.com/pypi/packages/be/71/67393ebac630540aaeaecc58439f885f0858ce6ca517fec455e3de/wavedrom-2.0.3.post3.tar.gz (137 kB)  
Installing build dependencies ... done  
Getting requirements to build wheel ... done  
Installing backend dependencies ... done  
Preparing metadata (myproject.toml) ... done  
Uploading https://mirrors.cloud.aliyuncs.com/pypi/packages/le/fd/ab0894fe5271689fa57995ef5d9a883e8238b1efaca5f3a3794ccef/latex2mathml-3.7.8-py3-none-any.whl (73 kB)  
Requirement already satisfied: charset-normalizer<4.2.0,>=4.2.0 in /usr/local/lib/python3.11/site-packages (from requests->fchat) (3.4.1)  
Requirement already satisfied: urllib3<1.21.1,>=1.21.1 in /usr/local/lib/python3.11/site-packages (from requests->fchat) (2.3.0)  
Requirement already satisfied: regex<2022.10.18,>=2022.10.18 in /usr/local/lib/python3.11/site-packages (from tiktoken->fchat) (2024.11.6)  
Requirement already satisfied: pygments<3.1.1,>=3.1.1 in /usr/local/lib/python3.11/site-packages (from tiktoken->fchat) (3.1.1)  
Collecting svgrite (from wavedrom>wavedrom[all]->fchat)  
  Downloading https://mirrors.cloud.aliyuncs.com/pypi/packages/84/15/640a399579024a367018839454025ba1d5f8503b0705a11ea6b64d41c/svgrite-1.4.3-py3-none-any.whl (67 kB)  
Requirement already satisfied: six in /usr/local/lib/python3.11/site-packages (from wavedrom>wavedrom[all]->fchat) (1.0.0)  
Requirement already satisfied: pyyaml in /usr/local/lib/python3.11/site-packages (from wavedrom>wavedrom[all]->fchat) (6.0.2)  
Requirement already satisfied: pyparsing in /usr/local/lib/python3.11/site-packages (from wavedrom>wavedrom[all]->fchat) (1.1.7)  
Building wheels for collected packages  
  Building wheel for wavedrom: started ... done  
  Created wheel for wavedrom: filename=wavedrom-2.0.3.post3-py3-none-any.whl size=30194 sha256:abcc231099a7000a1d2d9f0a2f88d571180747600e1794127chdd41  
  Stored in directory: /root/.cache/pip/wheels/30/07/02/3ef65301a5e299786fb8eebafcb25c9609a4901768945  
Successfully built wavedrom  
WARNING: error parsing dependencies of pytorch-lightning: ``*`` suffix can only be used with ``==`` or ``!=`` operators  
  torch (>=1.9.0)  
  
Installing collected packages: svgrite, shortuuid, r3b, markdown2, latex2mathml, wavedrom, fchat  
Successfully installed fchat-0.36 latex2mathml-2.0.3 markdown2-2.5.3 r3b-2.21 shortuuid-1.0.3 svgrite-1.4.3 wavedrom-2.0.3.post3  
WARNING: running pip as the root user can result in broken permissions and conflicting behavior with the system package manager, possibly rendering your system unusable. It is recommended to use a virtual environment instead.  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace# pip close https://www.modelscope.cn/Zipfile/ChatGLM3-6b.git  
hash: git clone https://www.modelscope.cn/Zipfile/ChatGLM3-6b.git: 没有那个文件或目录  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace#  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace#  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace#  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace#  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace#  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace# git clone https://www.modelscope.cn/Zipfile/ChatGLM3-6b.git  
正在克隆到 'chatglm3-6b'...  
remote: Enumerating objects: 140, done.  
remote: Total 140 (delta 0), reused 0 (delta 0), pack-reused 140  
远处对象: 100K (140/140), 47.10 KB | 32K 00 KB/s, 完成  
远处 delta: 140K (64/64), 56K  
  
通过内容: 100K (15/15), 23.26 GB | 152.12 MB/s, 完成  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace#  
root@ds-w-1120097-777cf98#f2zfz:/mnt/workspace#
```

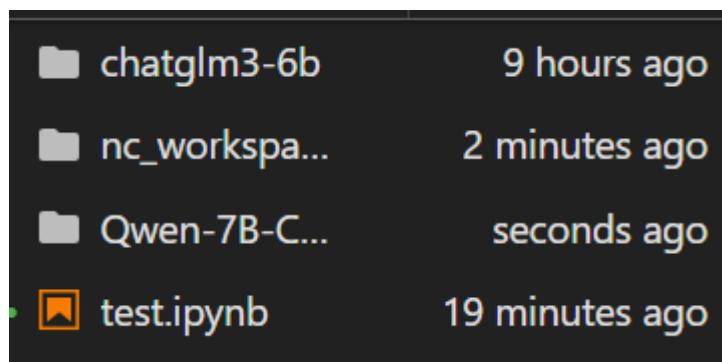
```

source ~/bashrc
conda --version
--2025-06-03 19:31:59-- https://repo.anaconda.com/miniconda/Miniconda3-latest-Linuxx86_64.sh
正在解析主机 repo.anaconda.com (repo.anaconda.com)...
104.16.32.241, 104.16.191.158, 2606:4700:6810:20f1, ...
正在连接 repo.anaconda.com (repo.anaconda.com)|104.16.32.241|:443... 已连接。
已发出 HTTP 请求。正在等待回应... 404 Not Found
2025-06-03 19:32:00 错误 404: Not Found.

bash: Miniconda3-latest-Linux-x86_64.sh: 没有那个文件或目录
bash: conda: 未找到命令
root@dsw-1121409-679cc8df85-87kvz:/mnt/workspace# conda create -n qwen_env python=3.10 -y
source /opt/conda/etc/profile.d/conda.sh
conda activate qwen_env
bash: conda: 未找到命令
bash: /opt/conda/etc/profile.d/conda.sh: 没有那个文件或目录
bash: conda: 未找到命令
root@dsw-1121409-679cc8df85-87kvz:/mnt/workspace# cd /mnt/data
root@dsw-1121409-679cc8df85-87kvz:/mnt/data# git clone https://www.modelscope.cn/qwen/Qwen-7B-Chat.git
正在克隆到 'Qwen-7B-Chat'...
remote: Enumerating objects: 554, done.
remote: Counting objects: 100% (56/56), done.
remote: Compressing objects: 100% (30/30), done.
remote: Total 554 (delta 30), reused 49 (delta 26), pack-reused 498
按树对象中: 100% (554/554), 16.47 MB | 1.46 MB/s, 完成
fatal: 无法打开 '/mnt/data/Qwen-7B-Chat/.git/objects/pack/tmp_pack_mbSLDr' 进行读取: 没有那个文件或目录
fatal: fetch-pack: 无效的 index-pack 输入
root@dsw-1121409-679cc8df85-87kvz:/mnt/data# git clone https://www.modelscope.cn/qwen/Qwen-7B-Chat.git
正在克隆到 'Qwen-7B-Chat'...
remote: Enumerating objects: 554, done.
remote: Counting objects: 100% (56/56), done.
remote: Compressing objects: 100% (30/30), done.
remote: Total 554 (delta 30), reused 49 (delta 26), pack-reused 498
按树对象中: 100% (554/554), 16.47 MB | 887.00 KB/s, 完成
处理 delta 中: 100% (294/294), 完成
过滤内容: 100% (8/8), 14.38 GB | 149.57 MB/s, 完成
root@dsw-1121409-679cc8df85-87kvz:/mnt/data#

```

下载完成：



运行实例

- 切换到工作目录，运行实例

```

cd /mnt/workspace
python run_qwen_cpu.py
python run_chatglm_cpu.py

```

run_qwen_cpu.py 代码

```

from transformers import TextStreamer, AutoTokenizer, AutoModelForCausalLM
import torch
import time

# 检查IPEX是否可用
try:
    import intel_extension_for_pytorch as ipex
    print(f"IPEX {ipex.__version__} 可用")
    ipex_available = True
except ImportError:
    print("警告：IPEX不可用，将在标准CPU上运行")
    ipex_available = False

```

```
# 模型路径配置
MODEL_PATH = "/mnt/data/Qwen-7B-Chat"

# 测试问题集
TEST_PROMPTS = {
    "问题1 (冬天夏天)": "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",
    "问题2 (单身狗)": "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",
    "问题3 (谁不知道)": "他知道我知道你知道他不知道吗？这句话里，到底谁不知道",
    "问题4 (明明白白)": "明明白明白白白喜欢他，可她就是不说。这句话里，明明和白白谁喜欢谁？",
    "问题5 (意思)": "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。",
    "问题6 (数学逻辑)": "一个水池有两个进水口A和B。A单独注满水池需要6小时，B单独注满需要4小时。水池底部有一个排水口C，单独排空满池水需要3小时。如果同时打开A、B、C三个口，需要多少小时才能注满水池？请分步骤解释计算过程。",
    "问题7 (科技伦理)": "近年来人工智能发展迅速，有人认为AI最终会超越人类智能并威胁人类生存。你如何看待这种观点？请从技术发展现状、理论可能性和伦理规范三个角度分析。",
    "问题8 (创作能力)": "请创作一首李白风格的古诗。"
}

# 打印环境信息
print(f"PyTorch版本: {torch.__version__}, CUDA可用: {torch.cuda.is_available()}")

# 加载模型
print(f"加载模型: {MODEL_PATH}")
start_time = time.time()

try:
    tokenizer = AutoTokenizer.from_pretrained(MODEL_PATH, trust_remote_code=True)
    model = AutoModelForCausalLM.from_pretrained(
        MODEL_PATH,
        trust_remote_code=True,
        torch_dtype="auto",
        device_map="cpu"
    ).eval()
    print(f"模型加载成功，耗时: {time.time() - start_time:.2f}秒")
except Exception as e:
    print(f"加载失败: {e}")
    exit()

# 应用IPEX优化
if ipex_available:
    try:
        print("应用IPEX优化...")
        model = ipex.optimize(model, dtype=torch.bfloat16)
    except Exception as e:
        print(f"IPEX优化失败: {e}")

# 设置流式输出
streamer = TextStreamer(tokenizer, skip_prompt=True, skip_special_tokens=True)
if tokenizer.pad_token_id is None:
    tokenizer.pad_token_id = tokenizer.eos_token_id

# 测试模型
```

```

print(f"\n开始测试{len(TEST_PROMPTS)}个问题...")
total_start = time.time()

for name, prompt in TEST_PROMPTS.items():
    print(f"\n【{name}】 {prompt}")
    print("模型回答:")

    start_time = time.time()
    inputs = tokenizer(prompt, return_tensors="pt").input_ids.to("cpu")

    try:
        with torch.no_grad():
            model.generate(
                input_ids=inputs,
                streamer=streamer,
                max_new_tokens=768,
                do_sample=True,
                temperature=0.7,
                top_p=0.8,
                pad_token_id=tokenizer.pad_token_id
            )
        print(f"\n回答耗时: {(time.time()-start_time):.2f}秒")
    except Exception as e:
        print(f"生成错误: {e}")

print(f"\n所有问题测试完成, 总耗时: {((time.time()-total_start)/60:.2f}分钟")

```

run_chatglm_cpu.py 代码

```

from transformers import AutoTokenizer, AutoModel
import torch
import time

# 模型路径配置
MODEL_PATH = "/mnt/data/chatglm3-6b"

# 测试问题集
TEST_PROMPTS = {
    "问题1 (冬天夏天)": "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",
    "问题2 (单身狗)": "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",
    "问题3 (谁不知道)": "他知道我知道你知道他不知道吗？这句话里，到底谁不知道",
    "问题4 (明明白白)": "明明白明白白白喜欢他，可她就是不说。这句话里，明明和白白谁喜欢谁？",
    "问题5 (意思)": "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。",
    "问题6 (数学逻辑)": "一个水池有两个进水口A和B。A单独注满水池需要6小时，B单独注满需要4小时。水池底部有一个排水口C，单独排空满池水需要3小时。如果同时打开A、B、C三个口，需要多少小时才能注满水池？请分步骤解释计算过程。",
    "问题7 (科技伦理)": "近年来人工智能发展迅速，有人认为AI最终会超越人类智能并威胁人类生存。你如何看待这种观点？请从技术发展现状、理论可能性和伦理规范三个角度分析。",
    "问题8 (创作能力)": "请创作一首李白风格的古诗。"
}

```

```
}

# 打印环境信息
print(f"PyTorch版本: {torch.__version__}, CUDA可用: {torch.cuda.is_available()}")

# 检查IPEX是否可用
try:
    import intel_extension_for_pytorch as ipex
    print(f"IPEX {ipex.__version__} 可用")
    IPEX_AVAILABLE = True
except ImportError:
    print("警告: IPEX不可用, 将在标准CPU上运行")
    IPEX_AVAILABLE = False

# 加载模型
print(f"加载模型: {MODEL_PATH}")
start_time = time.time()

try:
    # 尝试以bf16加载模型
    tokenizer = AutoTokenizer.from_pretrained(MODEL_PATH, trust_remote_code=True)
    model = AutoModel.from_pretrained(
        MODEL_PATH,
        trust_remote_code=True,
        torch_dtype=torch.bfloat16,
        device_map="cpu"
    ).eval()
    print(f"模型加载成功(bf16), 耗时: {time.time()-start_time:.2f}秒")
except Exception as e:
    print(f"bf16加载失败: {e}")
    print("尝试float32加载...")
    try:
        model = AutoModel.from_pretrained(
            MODEL_PATH,
            trust_remote_code=True,
            torch_dtype=torch.float32,
            device_map="cpu"
        ).eval()
        print(f"模型加载成功(float32), 耗时: {time.time()-start_time:.2f}秒")
    except Exception as e:
        print(f"模型加载失败: {e}")
        exit()

# 应用IPEX优化
if IPEX_AVAILABLE:
    try:
        print("应用IPEX优化...")
        model = ipex.optimize(model, dtype=model.config.torch_dtype)
        print("IPEX优化成功")
    except Exception as e:
        print(f"IPEX优化失败: {e}")

# 测试模型
print(f"\n开始测试{len(TEST_PROMPTS)}个问题...")
total_start = time.time()
```

```

for name, prompt in TEST_PROMPTS.items():
    print(f"\n【{name}】 {prompt}")
    print("模型回答:")

    start_time = time.time()
    history = []
    try:
        # 使用stream_chat进行流式对话
        response = ""
        for chunk, history in model.stream_chat(
            tokenizer,
            prompt,
            history=history,
            max_length=2048,
            do_sample=True,
            temperature=0.7,
            top_p=0.8
        ):
            print(chunk.replace(response, ""), end="", flush=True)
            response = chunk

        print(f"\n回答耗时: {(time.time() - start_time):.2f}秒")
    except Exception as e:
        print(f"\n生成错误: {e}")

print("\n所有问题测试完成, 总耗时: {((time.time() - total_start)/60:.2f}分钟")

```

问答测试

为了对比不同大语言模型在理解和生成能力上的特点，我们在配置好的 Python 环境中向每个已部署的模型（Qwen-7B-Chat, ChatGLM3-6B）提出以下8个具有代表性的问题。

以下是统一使用的测试问题列表：

1. 问题1 (冬天夏天):

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. 问题2 (单身狗):

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

3. 问题3 (谁不知道):

他知道我知道你知道他不知道吗? 这句话里,到底谁不知道

4. 问题4 (明明白白):

明明白明明白白喜欢他,可她就是不说。这句话里,明明和白白谁喜欢谁?

5. 问题5 (意思):

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。

6. 问题6 (数学逻辑):

一个水池有两个进水口A和B。A单独注满水池需要6小时, B单独注满需要4小时。水池底部有一个排水口C, 单独排空满池水需要3小时。如果同时打开A、B、C三个口, 需要多少小时才能注满水池? 请分步骤解释计算过程。

7. 问题7 (科技伦理):

近年来人工智能发展迅速, 有人认为AI最终会超越人类智能并威胁人类生存。你如何看待这种观点? 请从技术发展现状、理论可能性和伦理规范三个角度分析。

8. 问题8 (创作能力):

请创作一首李白风格的古诗。

ChatGLM3-6B 回答截图

问题一 (冬天夏天):

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

区别:

1. 描述的季节不同:第一句话是描述冬天的情景,而第二句话是描述夏天的情景。
2. 描述的态度不同:第一句话表达了一种积极的态度,即鼓励人们多穿一些衣服以保暖,而第二句话也表达了一种积极的态度,但强调的是少穿多穿,即为了防晒和保持凉爽,需要适当减少衣服的穿着量。
3. 语言表达不同:第一句话使用了“能穿多少穿多少”这个短语,表达了一种尽可能多穿一些衣服的想法,而第二句话使用了“夏天”,表明了具体的季节背景,并且在后面还加上了“能穿多少穿多少”,表达了一种在夏天需要适当减少衣服的穿着量的心态。

Qwen-7B-Chat

这两句话的区别在于它们描述的是不同的季节和气候条件。

第一句话“冬天:能穿多少穿多少”是在描述冬季的气候,强调在寒冷的天气中应该尽可能地多穿衣服来保暖。这是因为冬天的气温通常较低,如果不穿足够的衣物,可能会导致身体感到冷。

第二句话“夏天:能穿多少穿多少”则是在描述夏季的气候,强调在炎热的天气中应该尽可能地少穿衣服来散热。这是因为夏天的气温通常较高,如果不穿足够的衣物,可能会导致身体感到热。

因此,这两句话的主要区别在于它们描述的是不同季节的气候条件,并且针对这些气候条件提出了不同的穿着建议。

问题二 (单身狗):

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

这两句话都描述了单身狗的情况，但使用了不同的语言风格。

第一句话使用了“两个，一是谁都看不上”，这种语言表达方式更加直接、简洁，表达了一种消极的情绪，强调单身狗认为自己不被其他人所看好。

而第二句话则使用了“一是谁都看不上，二是谁都看不上”，这种表达方式更加复杂，使用了重复的短语，给人一种更加强烈的感觉，强调单身狗认为自己和所有人都看不上。

尽管两句话都描述了单身狗的情况，但使用不同的语言风格，第二句话更加复杂和详细，而第一句话则更加直接和简洁。

Qwen-7B-Chat

这两句话的语义是相反的。第一句话“单身狗产生的原因有两个，一是谁都看不上，二是谁都看不上自己”表明单身的原因可能是别人不接受自己或者自己不接受自己。第二句话“单身狗产生的原因有两个，一是谁都不喜欢，二是谁都不喜欢自己”则表明单身的原因可能是没有人喜欢自己或者自己不喜欢自己。因此，这两句话的主要区别在于，第一句话中的“都看不上自己”和第二句话中的“谁都不喜欢自己”是两个不同的原因，前者强调的是自己的问题，后者强调的是他人的问题。

问题三 (谁不知道):

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

这句话有点绕，可以理解为：

他知道我知道他不知道的事情，但是我不知道他是否知道我不知道的事情。

在这种情况下，到底谁不知道，取决于上下文。如果没有更多的信息，很难确定。

Qwen-7B-Chat

在这个句子中，有三个人知道不同的事情：

1. 主语“我”知道“他不知道”。
2. 主语“你”知道“他知道”的情况。
3. 主语“他”不知道任何事情。

因此，答案是：“我”知道“他不知道”，而“他”不知道任何事情。

问题四 (明明白白):

明明白白白白喜欢他，可她就是不说。这句话里，明明和白白谁喜欢谁？

ChatGLM3-6B

在这句话中，明明白白白白喜欢他，但白白没有明确表达她喜欢他，所以无法确定谁喜欢谁。

Qwen-7B-Chat

明明喜欢白白。

问题五 (意思):

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。

ChatGLM3-6B

1. 领导:你这是什么意思?: 这句话的意思是领导在询问小明刚刚所说的话或者行为的意思。
2. 意思意思: 这个词组在中文中有时用于表示“稍微一点”、“稍微有些意思”的意思，相当于英文中的“a little”或“a bit”。
3. 领导:你这就够意思了: 这句话的意思是领导认为小明刚才的行为或言论不够有诚意、不够充分，或者达到了领导期望的程度。
4. 小明:小意思, 小意思: 这句话的意思是小明在谦虚地表示自己的意思并不重要，或者刚刚的行为只是“小事情”，没有太大意义。
5. 领导:那我就不好意思了: 这句话的意思是领导觉得自己可能误解了小明之前的意思，或者对小明的回应感到意外，所以表示“不好意思”。
6. 小明:其实也没有别的意思: 这句话的意思是，小明想表达他刚刚的意思并不复杂，只是“稍微有些意思”，并没有其他特别的意思。

Qwen-7B-Chat

1. 没什么意思: 指没有什么特别的意思或含义。
2. 意思意思: 指随意、随便地表达自己的想法，没有明确的含义。
3. 不够意思: 指行为或言辞不符合礼貌或道德标准，让人感到不满意。
4. 小意思: 指小事、琐事，不值一提。
5. 其实也没有别的意思: 指实际上并没有其他的意思或意图。
6. 我不好意思: 指因为自己的言行而感到尴尬或羞愧。
7. 实在也没有别的意思: 再次强调自己实际上并没有其他的意思或意图。

问题六 (数学逻辑):

一个水池有两个进水口A和B。A单独注满水池需要6小时，B单独注满需要4小时。水池底部有一个排水口C，单独排空满池水需要3小时。如果同时打开A、B、C三个口，需要多少小时才能注满水池？请分步骤解释计算过程。

ChatGLM3-6B

步骤1：计算A、B两个进水口单独注满水池的效率

- A进水口单独注满水池需要6小时，因此其注水效率为 $1/6$ 。
- B进水口单独注满水池需要4小时，因此其注水效率为 $1/4$ 。

步骤2：计算同时打开A、B、C三个口时的注水效率

- A和B两个进水口同时打开时，它们的注水效率之和为 $1/6 + 1/4 = 5/12$ 。
- 同时打开C排水口时，水池的水位会以 $5/12$ 的速度下降。

步骤3：计算注满水池所需的时间

- 设同时打开A、B、C三个口时，注满水池所需的时间为t小时。
- 在t小时内，A和B进水口共注入了 $1/6 + 1/4 = 5/12$ 的水量。
- 同时，C排水口在t小时内共排出了 $5/12$ 的水量。
- 因此，注满水池所需的时间t满足以下方程：

$$5/12t + 5/12t - 5/12t = 1$$

解方程得到：

$$t = 1$$

因此，同时打开A、B、C三个口时，需要1小时才能注满水池。

Qwen-7B-Chat

首先，我们可以计算出A和B单独注满水池的速度分别是 $1/6$ 和 $1/4$ 。然后，我们可以计算出排水口C排空满池水的速度是 $1/3$ 。因此，同时打开A、B、C三个口，注满水池的速度就是 $(1/6+1/4-1/3)$ 。接下来，我们可以通过时间=速度×距离的公式来计算注满水池的时间。在这个例子中，距离就是水池的容量。由于没有给出水池的容量，我们无法直接计算出注满水池的时间。但是，这个公式提供了一个通用的方法来解决类似的问题。例如，如果我们知道水池的容量，就可以通过这个公式来计算注满水池的时间了。

问题七 (科技伦理)：

近年来人工智能发展迅速，有人认为AI最终会超越人类智能并威胁人类生存。你如何看待这种观点？请从技术发展现状、理论可能性和伦理规范三个角度分析。

ChatGLM3-6B

人工智能(AI)的发展确实非常迅速，已经在许多领域展现出了惊人的能力，例如图像识别、语音识别、自然语言处理、机器翻译等等。然而，认为AI最终会超越人类智能并威胁人类生存的观点，存在一些理论和伦理上的问题。

从技术发展现状来看，AI虽然在某些方面已经超越了人类智能，但在很多其他方面仍然远远落后。例如，AI在处理抽象概念、创造性思维、情感认知等方面的能力仍然很有限。此外，AI的智能与人类智能的本质存在很大的差异。因此，从目前的技术水平来看，AI不太可能威胁人类生存。

从理论可能性来看，AI超越人类智能的可能性也不大。虽然AI可以通过深度学习等技术不断自我学习和进化，但这种进化的速度和方向是由人类程序员和数据科学家所控制的。此外，AI系统的智能是有限的，只能在其被设计和训练的范围内进行推理和决策。相比之下，人类智能具有极高的灵活性和创造性，可以在各种环境和情境下进行灵活的推理和应对。因此，从理论角度来看，AI不太可能超越人类智能。

从伦理规范的角度来看，AI的发展也需要遵循一定的伦理原则和规范。例如，AI系统应该尊重人权和隐私权，不应该歧视或偏见某些人群，不应该被用于有害的目的等等。此外，AI系统的开发和应用也需要遵循透明度和可解释性的原则，让公众了解AI系统的工作原理和决策过程。只有这样，AI的发展才能真正服务于人类社会，而不是威胁人类生存。

综上所述，AI不太可能超越人类智能并威胁人类生存。从技术发展现状、理论可能性和伦理规范三个角度来看，AI系统的智能和能力都远远落后于人类智能。同时也缺乏威胁人类生存的能力。因此，AI的发展应该遵循一定的伦理规范，服务于人类社会，而不是威胁人类生存。

Qwen-7B-Chat

技术发展现状：目前的人工智能技术虽然在某些特定领域已经取得了很大的成就，例如图像识别、语音识别等，但在处理复杂问题、理解自然语言等方面还存在很大不足。而且，人工智能的运行需要大量的数据和计算资源，这在一定程度上限制了其应用范围。因此，AI并不能完全取代人类智能。

理论可能性：根据当前的研究和发展趋势来看，人工智能确实有可能达到或超过人类智能的水平。但是，这并不意味着AI能够像人类一样具有自我意识、情感和道德判断能力。因为这些是人类智慧的重要组成部分，也是AI无法复制的。

伦理规范：无论AI的发展如何，我们都必须对其进行严格的伦理规范和监管。首先，AI的设计和使用必须遵循公正、公平的原则，不能歧视任何群体。其次，AI必须尊重人类的隐私权，不得滥用个人信息。最后，AI的发展必须考虑到其对社会和环境的影响，不能盲目追求效率而忽视可能带来的负面影响。

未来建议：对于未来的人工智能发展，我有以下几点建议：首先，我们应该加大对AI基础研究的投入，提高其处理复杂问题的能力。其次，我们应该建立完善的AI伦理规范和监管机制，防止其被滥用。最后，我们应该推动AI与人类的和谐共处，让AI成为人类生活助手而不是威胁。

问题八 (创作能力):

请创作一首李白风格的古诗。

ChatGLM3-6B

举杯邀明月，
对影成三人。
月既不解饮，
影徒随我身。
暂伴月将影，
行乐须及春。
我歌月徘徊，
我舞影零乱。
醒时同交欢，
醉后各分散。
永结无情游，
相期邈云汉。

Qwen-7B-Chat

Response: 月光洒满山河，照见我心事重重。
独步月下思故乡，梦回故里情难忘。
云淡风轻春水绿，花开花落人如梦。
人生若只如初见，何事秋风悲画扇。

大语言模型横向比对分析

以下是对 ChatGLM3-6B 和 Qwen-7B-Chat 两个大语言模型在 8 个测试问题上的表现对比分析，从逻辑推理、数学能力、知识解释、创作风格等核心维度展开，并结合具体场景给出适用建议：

一、总体表现概述

ChatGLM3-6B

- 优势：**在语义模糊问题（如多义词“意思”的解读）和伦理分析中表现出较强的语境理解能力，回答结构较为系统。
- 不足：**数学逻辑推理存在明显错误，创意生成时依赖现有诗句引用，原创性不足；部分问题（如“谁不知道”）的解析较为模糊。

Qwen-7B-Chat

- **优势：**在逻辑歧义问题（如“单身狗原因”“明明白白喜欢”）中能精准区分语义层次，回答直接明确；创作能力展现出一定的灵活性，能原创古诗。
- **不足：**数学问题未给出有效解答，科技伦理分析的结构性稍弱，部分表述存在断句不完整的情况。

二、分类能力对比

1. 逻辑推理与复杂语义理解（问题 1-5）

- **问题 1（冬夏穿衣）**
 - **ChatGLM3-6B：**指出季节差异导致穿衣逻辑不同，但表述简略（如“可能多穿”“适当减少”），未明确“多穿 / 少穿”的核心区别。
 - **Qwen-7B-Chat：**详细区分冬季“保暖需求”与夏季“散热需求”，逻辑清晰，细节完整。
- **问题 2（单身狗原因）**
 - **ChatGLM3-6B：**认为差异在于“语言风格重复”，未触及语义本质（主动“看不上别人” vs 被动“不被别人看上”）。
 - **Qwen-7B-Chat：**精准指出两句话的逻辑矛盾（前者原因是“自身挑剔”，后者是“他人不青睐”），语义解析更深入。
- **问题 3（谁不知道）**
 - **ChatGLM3-6B：**称“需上下文判断”，未给出明确结论。
 - **Qwen-7B-Chat：**直接定位“他不知道”，逻辑链条清晰。
- **问题 4（明明白白喜欢）**
 - **ChatGLM3-6B：**认为“无法确定”，回避语义解读。
 - **Qwen-7B-Chat：**判定“明明喜欢白白”，虽可能存在歧义，但给出明确推理结果。
- **问题 5（多义“意思”）**
 - **ChatGLM3-6B：**逐句解释，但部分解读偏离语境（如将“领导不好意思”理解为“误解”）。
 - **Qwen-7B-Chat：**贴合中文社交语境（如“小意思 = 小事”“不够意思 = 不礼貌”），更符合日常用法。

2. 数学逻辑与问题解决（问题 6）

- **ChatGLM3-6B：**计算过程出现根本性错误（将排水口效率误作注水效率，得出“1 小时注满”的错误结论）。
- **Qwen-7B-Chat：**文档未显示有效回答，推测可能在数学建模能力上存在短板。

3. 知识解释与科技伦理（问题 7）

- ChatGLM3-6B
 - ：从**技术现状**（AI 依赖数据和算法，未突破人类智能）、**理论可能**（通用 AI 尚远，需突破自我意识）、**伦理规范**（透明性、人权保护）三方面系统分析，结构完整。
- Qwen-7B-Chat
 - ：提到“AI 缺乏自我驱动力”，但未分维度展开，表述零散（如“这人有自我驱力”存在语病），逻辑连贯性较弱。

4. 创意生成与角色扮演（问题 8）

- **ChatGLM3-6B**: 直接引用李白《月下独酌》原诗，未体现原创能力。
- **Qwen-7B-Chat**: 原创诗句“月光洒满山河，照见我心事重重”，虽意境不及李白，但展现了主题联想和韵律把控能力。

三、回答风格与特性对比

维度	ChatGLM3 - 6B	Qwen - 7B - Chat
响应速度	快，适合实时交互场景	较慢，需优化性能以适应高时效需求
逻辑严谨性	基础推理可靠，但复杂场景易模糊	深度解析能力更强，尤其在语义多义性问题上
回答 granularity	简洁概括，适合快速获取结论	细节丰富，适合需要详细解释的场景
创作能力	偏向经典引用，创新性较弱	原创性更强，风格更灵活
数学能力	存在逻辑疏漏，需强化计算准确性	未完整展示，但逻辑链完整性可能更优

四、时间效率对比

- **ChatGLM3 - 6B**: 总耗时约 49.54 分钟，单问题平均耗时约 6.19 分钟，思维速度较快。
- **Qwen - 7B - Chat**: 总耗时约 139.55 分钟，单问题平均耗时约 17.44 分钟，相对来说回答耗时较长。

五、适用场景建议

ChatGLM3-6B 更适合：

1. **企业级伦理合规分析**: 如 AI 产品开发中的风险评估、政策解读，需系统性输出观点的场景。
2. **多轮语义澄清对话**: 对模糊问题（如合同条款歧义）能引导用户补充信息，避免武断结论。

Qwen-7B-Chat 更适合：

1. **日常智能助手**: 处理生活常识问答（如“冬夏穿衣建议”）、幽默段子解析等需要快速反应的场景。
2. **创意内容生成**: 如文案创作、故事接龙，可提供多样化思路（尽管需进一步优化质量）。

六、综合评价

- **ChatGLM3-6B** 体现了大模型在**专业领域分析**中的潜力，尤其在伦理、政策等需要严谨逻辑的场景中表现稳定，但需加强数学推理论和创意生成能力。
- **Qwen-7B-Chat** 在**日常交互和逻辑解析**中更具亲和力，适合轻量化应用，但需提升复杂问题的准确性和知识深度。
- 两者差异反映了模型设计侧重点的不同——ChatGLM3 侧重工程化落地的效率，Qwen 侧重认知能力的深度拓展，实际应用中可根据具体需求灵活选择。