山东大学计算机科学与技术学院

可视化技术课程实验报告

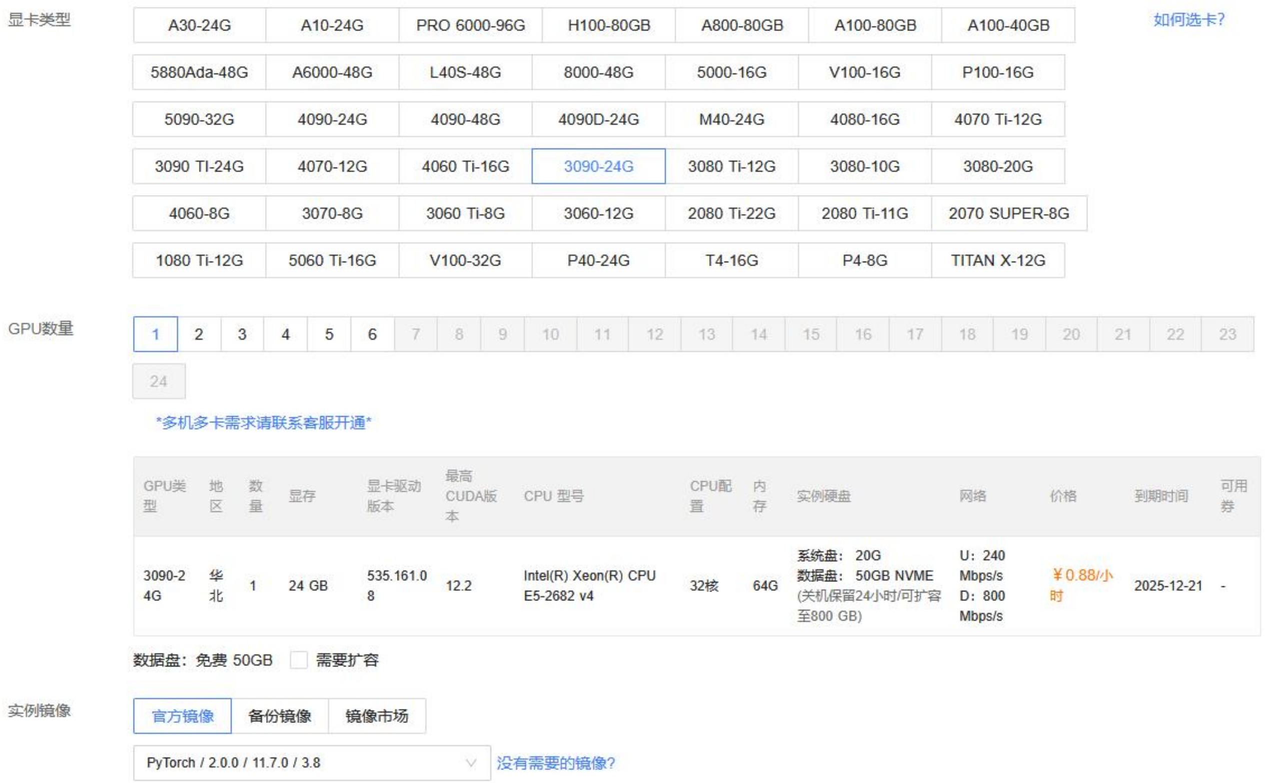
学号：202300130220	姓名：刘傲宇	班级：数据科学与大数据技术班
实验题目：BERT 环境配置		
实验学时：2	实验日期：2025/11/5	

实验目标：

对动手实践利用机器学习方法分析大规模数据有进一步了解，并学习如何利用远程环境进行工程代码的调试

实验步骤与内容：

云服务器配置：



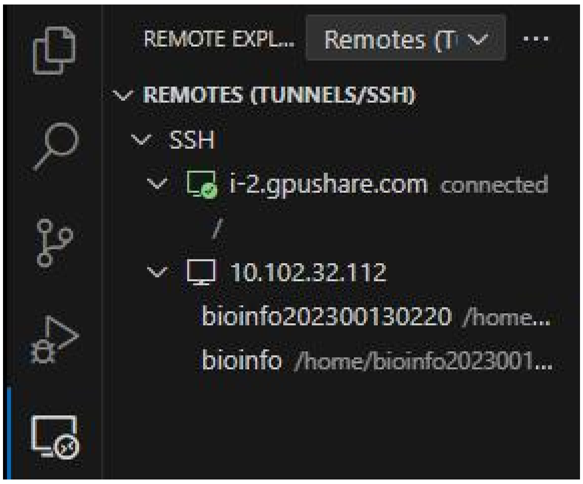
GPU类型	地区	数量	显存	显卡驱动版本	最高CUDA版本	CPU 型号	CPU配置	内存	实例硬盘	网络	价格	到期时间	可用券
3090-24G	华北	1	24 GB	535.161.08	12.2	Intel(R) Xeon(R) CPU E5-2682 v4	32核	64G	系统盘：20G 数据盘：50GB NVME (关机保留24小时/可扩容至800 GB)	U：240 Mbps/s D：800 Mbps/s	¥0.88/小时	2025-12-21	-

数据盘：免费 50GB □ 需要扩容

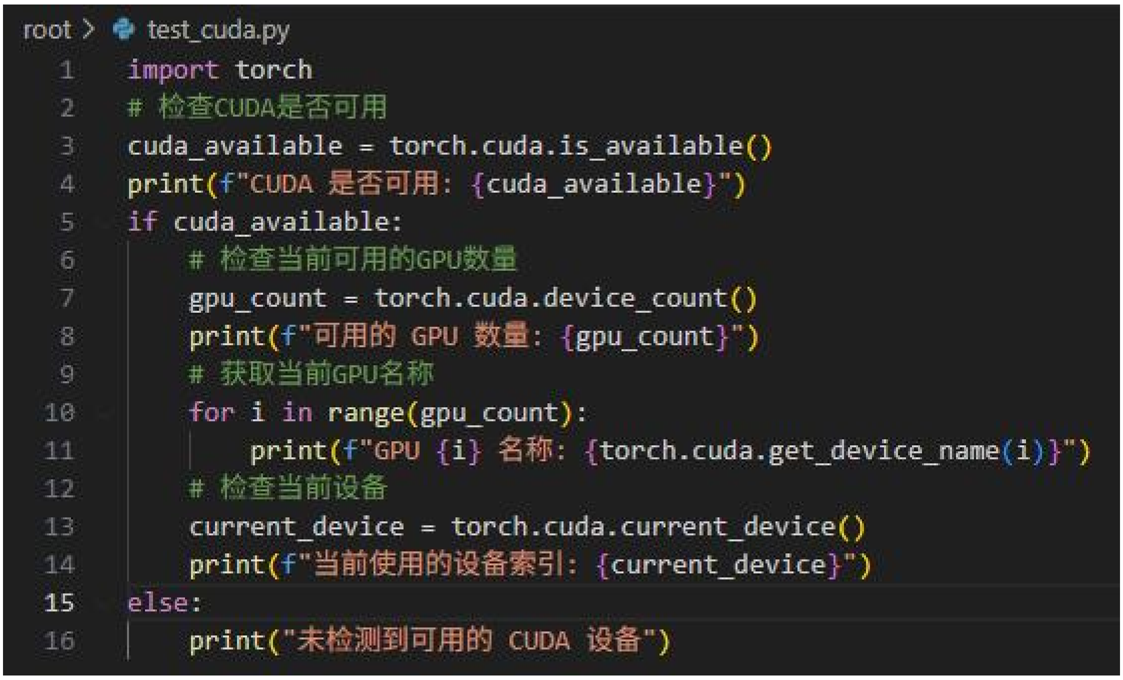
PyTorch / 2.0.0 / 11.7.0 / 3.8

PyTorch==2.0.0，Cuda==11.7.0，Python==3.8，比较 Stable。

在 vscode 中连接服务器：

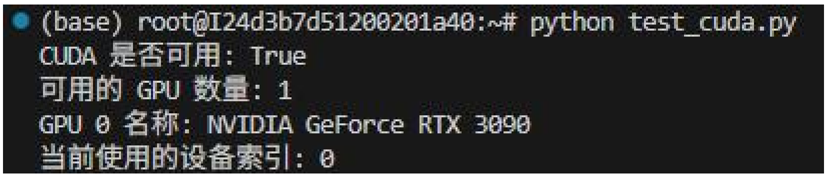

REMOTE EXPL...
Remotes (T
REMOTES (TUNNELS/SSH)
SSH
i-2.gpushare.com connected
/
10.102.32.112
bioinfo202300130220 /home...
bioinfo /home/bioinfo2023001...

验证 CUDA 和 GPU 可用性：



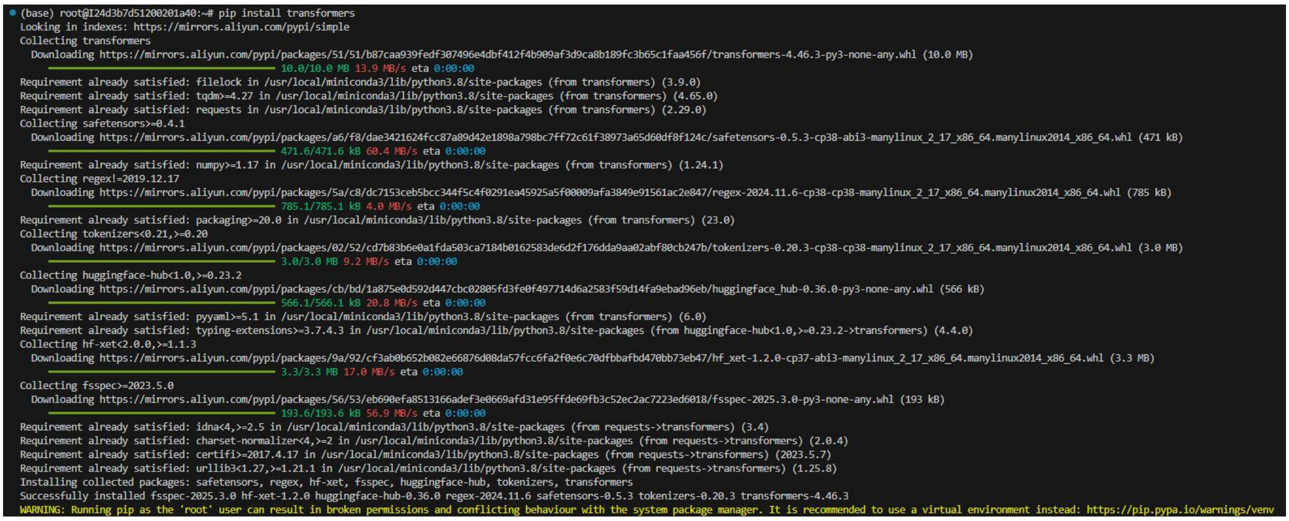
```python
import torch
# 检查CUDA是否可用
cuda_available = torch.cuda.is_available()
print(f"CUDA 是否可用: {cuda_available}")
if cuda_available:
    # 检查当前可用的GPU数量
    gpu_count = torch.cuda.device_count()
    print(f"可用的 GPU 数量: {gpu_count}")
    # 获取当前GPU名称
    for i in range(gpu_count):
        print(f"GPU {i} 名称: {torch.cuda.get_device_name(i)}")
    # 检查当前设备
    current_device = torch.cuda.current_device()
    print(f"当前使用的设备索引: {current_device}")
else:
    print("未检测到可用的 CUDA 设备")
```

可用：



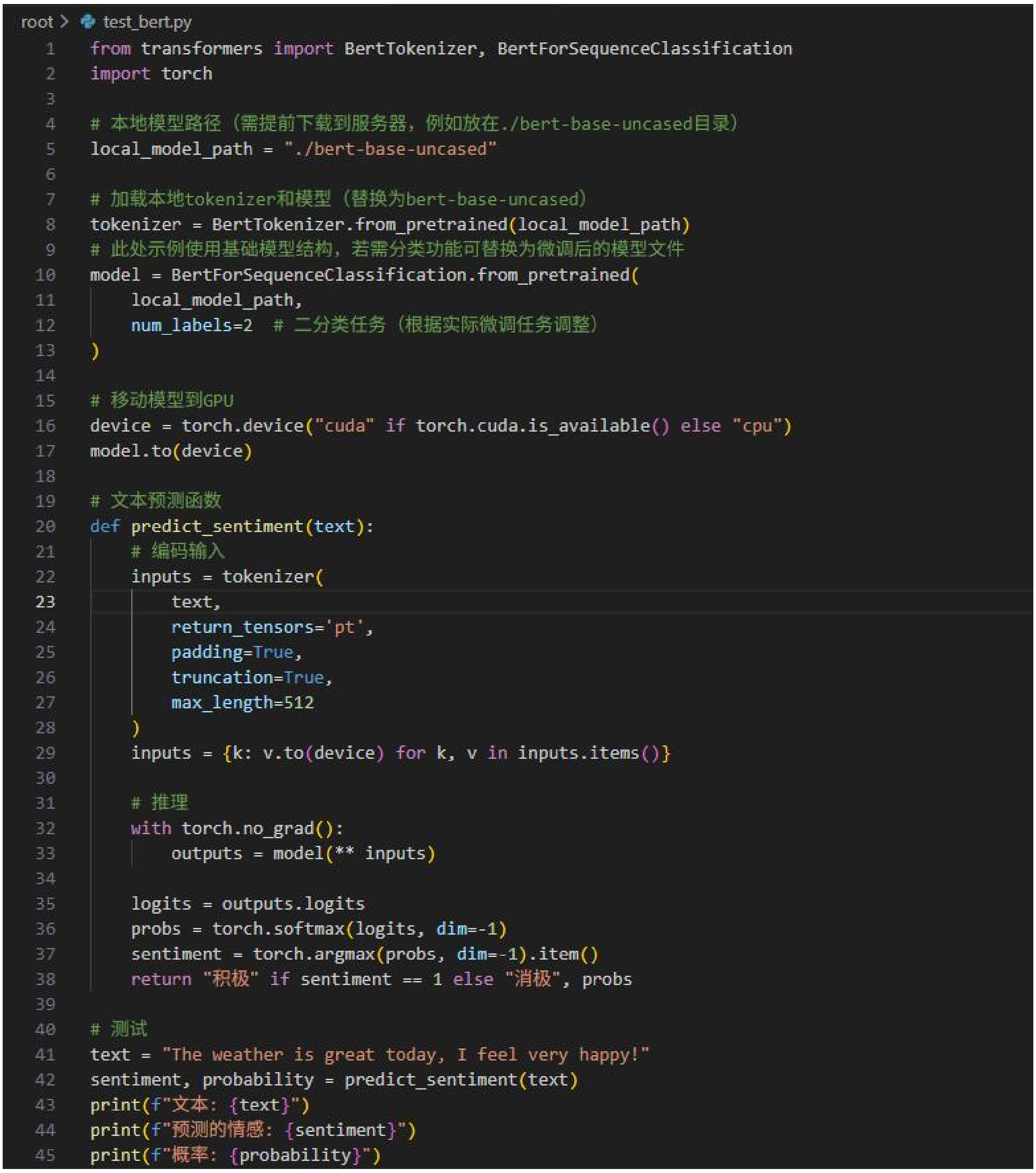
```
(base) root@I24d3b7d51200201a40:~# python test_cuda.py
CUDA 是否可用: True
可用的 GPU 数量: 1
GPU 0 名称: NVIDIA GeForce RTX 3090
当前使用的设备索引: 0
```

安装 transformers：



```
(base) root@I24d3b7d51200201a40:~# pip install transformers
Looking in indexes: https://mirrors.aliyun.com/pypi/simple
Collecting transformers
  Downloading https://mirrors.aliyun.com/pypi/packages/51/51/b87caa939fedf307496e4dbf412f4b909af3d9ca8b189fc3b65c1faa456f/transformers-4.46.3-py3-none-any.whl (10.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10.0/10.0 MB 13.9 MB/s eta 0:00:00
Requirement already satisfied: filelock in /usr/local/miniconda3/lib/python3.8/site-packages (from transformers) (3.9.0)
Requirement already satisfied: tqdm>=4.27 in /usr/local/miniconda3/lib/python3.8/site-packages (from transformers) (4.65.0)
Requirement already satisfied: requests in /usr/local/miniconda3/lib/python3.8/site-packages (from transformers) (2.29.0)
Collecting safetensors>=0.4.1
  Downloading https://mirrors.aliyun.com/pypi/packages/a6/f8/dae3421624fcc87a89d42e1898a798bc7ff72c61f38973a65d60df8f124c/safetensors-0.5.3-cp38-abi3-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (471 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 471.6/471.6 kB 60.4 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.17 in /usr/local/miniconda3/lib/python3.8/site-packages (from transformers) (1.24.1)
Collecting regex!=2019.12.17
  Downloading https://mirrors.aliyun.com/pypi/packages/5a/c8/dc7153ceb5bcc344f5c4f0291ea45925a5f00009afa3849e91561ac2e847/regex-2024.11.6-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (785 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 785.1/785.1 kB 4.0 MB/s eta 0:00:00
Requirement already satisfied: packaging>=20.0 in /usr/local/miniconda3/lib/python3.8/site-packages (from transformers) (23.0)
Collecting tokenizers<0.21,>=0.20
  Downloading https://mirrors.aliyun.com/pypi/packages/02/52/cd7b83b6e0a1fda503ca7184b0162583de6d2f176dda9aa02abf80cb247b/tokenizers-0.20.3-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.0/3.0 MB 9.2 MB/s eta 0:00:00
Collecting huggingface-hub<1.0,>=0.23.2
  Downloading https://mirrors.aliyun.com/pypi/packages/cb/bd/1a875e0d592d447cbc02805fd3fe0f497714d6a2583f59d14fa9ebad96eb/huggingface_hub-0.36.0-py3-none-any.whl (566 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 566.1/566.1 kB 20.8 MB/s eta 0:00:00
Requirement already satisfied: pyyaml>=5.1 in /usr/local/miniconda3/lib/python3.8/site-packages (from transformers) (6.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/miniconda3/lib/python3.8/site-packages (from huggingface-hub<1.0,>=0.23.2->transformers) (4.4.0)
Collecting hf-xet<2.0.0,>=1.1.3
  Downloading https://mirrors.aliyun.com/pypi/packages/9a/92/cf3ab0b652b082e66876d08da57fcc6fa2f0e6c70dfbbafbd470bb73eb47/hf_xet-1.2.0-cp37-abi3-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.3 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.3/3.3 MB 17.0 MB/s eta 0:00:00
Collecting fsspec>=2023.5.0
  Downloading https://mirrors.aliyun.com/pypi/packages/56/53/eb690efa8513166adef3e0669afd31e95ffde69fb3c52ec2ac7223ed6018/fsspec-2025.3.0-py3-none-any.whl (193 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 193.6/193.6 kB 56.9 MB/s eta 0:00:00
Requirement already satisfied: idna<4,>=2.5 in /usr/local/miniconda3/lib/python3.8/site-packages (from requests->transformers) (3.4)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/miniconda3/lib/python3.8/site-packages (from requests->transformers) (2.0.4)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/miniconda3/lib/python3.8/site-packages (from requests->transformers) (2023.5.7)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/miniconda3/lib/python3.8/site-packages (from requests->transformers) (1.25.8)
Installing collected packages: safetensors, regex, hf-xet, fsspec, huggingface-hub, tokenizers, transformers
Successfully installed fsspec-2025.3.0 hf-xet-1.2.0 huggingface-hub-0.36.0 regex-2024.11.6 safetensors-0.5.3 tokenizers-0.20.3 transformers-4.46.3
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
```

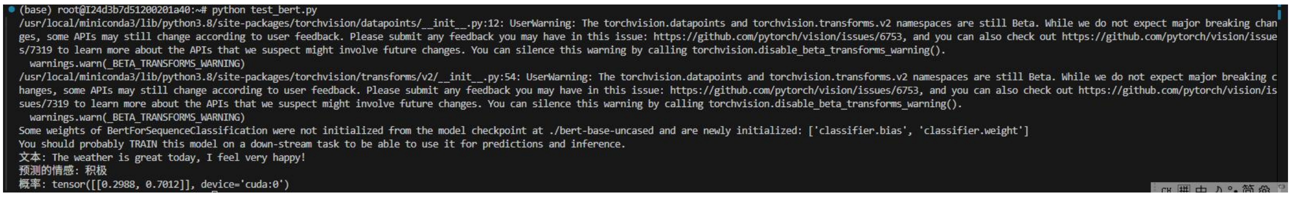
测试 BERT 环境：



root > test_bert.py

```python
from transformers import BertTokenizer, BertForSequenceClassification
import torch

# 本地模型路径（需提前下载到服务器，例如放在./bert-base-uncased目录）
local_model_path = "./bert-base-uncased"

# 加载本地tokenizer和模型（替换为bert-base-uncased）
tokenizer = BertTokenizer.from_pretrained(local_model_path)
# 此处示例使用基础模型结构，若需分类功能可替换为微调后的模型文件
model = BertForSequenceClassification.from_pretrained(
    local_model_path,
    num_labels=2  # 二分类任务（根据实际微调任务调整）
)

# 移动模型到GPU
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
model.to(device)

# 文本预测函数
def predict_sentiment(text):
    # 编码输入
    inputs = tokenizer(
        text,
        return_tensors='pt',
        padding=True,
        truncation=True,
        max_length=512
    )
    inputs = {k: v.to(device) for k, v in inputs.items()}

    # 推理
    with torch.no_grad():
        outputs = model(** inputs)

    logits = outputs.logits
    probs = torch.softmax(logits, dim=-1)
    sentiment = torch.argmax(probs, dim=-1).item()
    return "积极" if sentiment == 1 else "消极", probs

# 测试
text = "The weather is great today, I feel very happy!"
sentiment, probability = predict_sentiment(text)
print(f"文本: {text}")
print(f"预测的情感: {sentiment}")
print(f"概率: {probability}")
```

测试结果如下，BERT 环境正常：



```
(base) root@I24d3b7d51200201a40:~# python test_bert.py
/usr/local/miniconda3/lib/python3.8/site-packages/torchvision/datapoints/__init__.py:12: UserWarning: The torchvision.datapoints and torchvision.transforms.v2 namespaces are still Beta. While we do not expect major breaking chan
ges, some APIs may still change according to user feedback. Please submit any feedback you may have in this issue: https://github.com/pytorch/vision/issues/6753, and you can also check out https://github.com/pytorch/vision/issue
s/7319 to learn more about the APIs that we suspect might involve future changes. You can silence this warning by calling torchvision.disable_beta_transforms_warning().
  warnings.warn(_BETA_TRANSFORMS_WARNING)
/usr/local/miniconda3/lib/python3.8/site-packages/torchvision/transforms/v2/__init__.py:54: UserWarning: The torchvision.datapoints and torchvision.transforms.v2 namespaces are still Beta. While we do not expect major breaking c
hanges, some APIs may still change according to user feedback. Please submit any feedback you may have in this issue: https://github.com/pytorch/vision/issues/6753, and you can also check out https://github.com/pytorch/vision/is
sues/7319 to learn more about the APIs that we suspect might involve future changes. You can silence this warning by calling torchvision.disable_beta_transforms_warning().
  warnings.warn(_BETA_TRANSFORMS_WARNING)
Some weights of BertForSequenceClassification were not initialized from the model checkpoint at ./bert-base-uncased and are newly initialized: ['classifier.bias', 'classifier.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
文本: The weather is great today, I feel very happy!
预测的情感: 积极
概率: tensor([[0.2988, 0.7012]], device='cuda:0')
```

结论分析与体会：

本次实验成功搭建了兼容 BERT 模型的远程环境，PyTorch、CUDA、transformers 等组件版本匹配，GPU 调用正常，通过本地加载 bert-base-uncased 模型完成测试，验证了全流程可用性。体会到版本兼容性和国内镜像源是环境配置的关键，远程服务器与 VSCode 的结合既利用了高性能 GPU，又方便调试，本地模型加载则提升了效率、降低了网络依赖，这些技巧为后续机器学习任务提供了实用参考。