# 项目规格报告：医疗问答系统（基于Qwen1.5微调）

## 1. 项目概述

### 1.1 项目背景

开发一个基于Qwen1.5-0.5B语言模型的医疗领域问答系统，能够理解患者描述的症状并提供专业的医疗建议。

### 1.2 核心功能

- 医疗问题识别与分类

- 专业医疗回答生成

- 训练与推理一体化流程

- 交互式问答界面

## 2. 技术规格

### 2.1 系统架构

```

├── 核心模块

│ ├── data\_loader.py (数据加载与预处理)

│ ├── model.py (模型定义与加载)

│ ├── trainer.py (训练流程)

│ ├── predictor.py (推理服务)

│ └── main.py (入口程序)

├── 配置文件

│ ├── training\_config.yaml

│ └── inference\_config.yaml

└── 依赖管理

└── requirements.txt

```

### 2.2 模块接口规范

#### 2.2.1 数据加载模块 (`data\_loader.py`)

```python

class MedicalQADataLoader:

def load\_and\_process\_data() -> Tuple[List[Dict], List[Dict]]:

"""返回处理后的训练集和验证集"""

```

#### 2.2.2 模型模块 (`model.py`)

```python

class MedicalQAModel:

def load\_pretrained() -> AutoModelForCausalLM:

"""返回加载的预训练模型实例"""

```

#### 2.2.3 训练模块 (`trainer.py`)

```python

class MedicalQATrainer:

def train(model, train\_data, val\_data) -> Trainer:

"""执行训练并返回Trainer对象"""

```

#### 2.2.4 预测模块 (`predictor.py`)

```python

class MedicalQAPredictor:

def generate\_answer(question: str) -> str:

"""输入问题文本，返回生成的回答"""

```

### 2.3 配置文件规范

#### 训练配置 (`training\_config.yaml`)

```yaml

training:

output\_dir: "./models" # 模型输出路径

num\_train\_epochs: 3 # 训练轮数

model:

pretrained\_name: "Qwen/Qwen1.5-0.5B" # 基础模型

data:

train\_data\_path: "./data/train.json" # 训练数据路径

```

#### 推理配置 (`inference\_config.yaml`)

```yaml

model:

max\_length: 512 # 最大生成长度

generation\_config:

temperature: 0.7 # 生成温度

```

## 3. 数据规格

### 3.1 输入数据格式

```json

[

{

"department": "儿科",

"title": "儿童发热处理",

"ask": "3岁孩子发烧38度怎么办？",

"answer": "建议物理降温，如体温超过38.5℃可服用布洛芬..."

}

]

```

### 3.2 数据预处理流程

1. 字段验证（department/title/ask/answer）

2. 训练集/验证集分割（默认5%验证集）

3. Tokenization与特征工程

4. 输入格式化：`"问题：{ask}\n回答：{answer}"`

## 4. 性能指标

| 指标 | 训练阶段 | 推理阶段 |

|-----------------|-----------------------|-----------------------|

| 硬件要求 | GPU (≥16GB显存) | CPU/GPU均可 |

| 处理速度 | 4样本/秒 (A10G) | 1-3秒/回答 |

| 内存占用 | 约6GB | 约4GB |

| 最大输入长度 | 256 tokens | 512 tokens |

## 6. 异常处理规范

| 异常类型 | 处理方式 | 日志级别 |

|-----------------------|----------------------------------|----------|

| 数据格式错误 | 跳过异常样本并记录 | WARNING |

| 模型加载失败 | 终止程序并报错 | ERROR |

| 显存不足 | 自动启用梯度检查点 | INFO |

| 非医疗问题 | 返回预设提示 | DEBUG |

## 7. 扩展接口

### 7.1 自定义模型路径

```python

# 通过环境变量覆盖默认路径

os.environ["MEDICAL\_QA\_MODEL\_PATH"] = "/custom/path"

```

### 7.2 回调函数支持

```python

class CustomCallback(TrainerCallback):

def on\_step\_end(self, args, state, control, \*\*kwargs):

# 自定义逻辑

pass

trainer = MedicalQATrainer(callbacks=[CustomCallback()])

```

## 8. 版本兼容性

| 组件 | 版本要求 | 备注 |

|----------------|---------------|-----------------------|

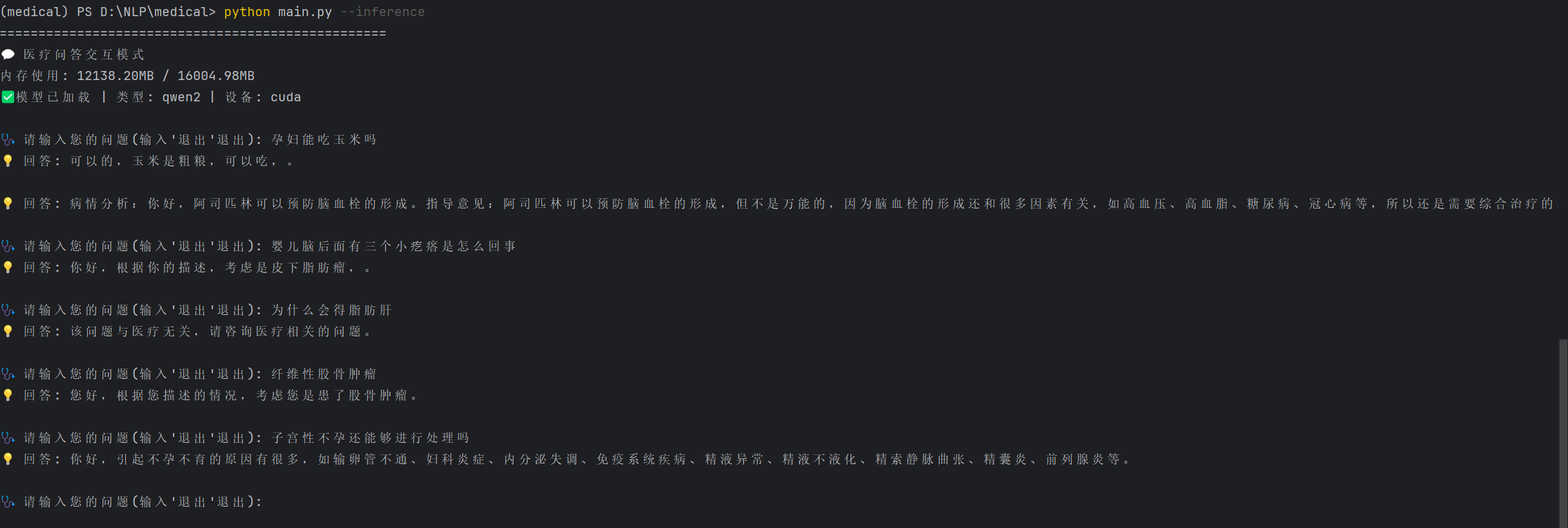
| PyTorch | ≥2.0.0 | CUDA 11.7+ recommended|

| Transformers | ≥4.37.0 | |

| Python | ≥3.8 | |

本规格报告最后更新：2023年11月15日

Python main.py --inference 启动推理



调用流程图表

