AI生成文本检测任务

任务分配

- **检测部分（建议分配1~2人）：**需要根据现有代码，使用BERT类的模型训练一个二分类检测器，实现对deepseek文本的检测任务。
- **服务器部分（建议分配2~3人）：**需要在127.0.0.1:80上构建web服务，使用Nginx将用户请求转发到后端检测器监听的端口中。

最终实现效果：

1. 用户在浏览器中输入127.0.0.1访问Web页面。
2. 用户点击检测按钮，请求发往127.0.0.1:80/api/text子路由。
3. Nginx识别到/api后缀，将请求转发给127.0.0.1:10001端口。
4. 后端检测器使用flask的路由修饰器，监听/text路由。识别到用户的检测请求和文本，进行检测。
5. 检测成功后，返回检测结果。用户在Web页面上看到该文本的AI生成概率。

注意事项：

- 请构建一个交流群，各位同学在交流群中交流进度和问题。不懂的部分优先询问AI（建议询问gemini 2.5 pro）。
- 所有使用windows的同学，请启用wsl安装ubuntu22.04作为Linux环境。所有开发工作**必须在Linux环境下进行**。建议使用vscode或cursor作为开发工具。
- 所有同学**必须使用git**作为代码管理工具，建议使用GitHub作为远程仓库，负责人可以在GitHub上开一个私有仓库。
- 所有同学使用python时，不允许将包装在默认环境中。**强烈建议使用conda**（建议miniconda）作为python版本管理工具。请在conda中启用虚拟环境再安装依赖。

检测任务

已有内容

- 我们提供deepseek的训练数据（在HC3数据集人类问答数据集的基础上，使用deepseek的API构建的AI回答）。

注意事项：

- 负责该部分的同学电脑必须有**NVIDIA独立显卡**，例如RTX 4060。使用CPU去训练模型速度会非常慢！

任务目标

- 对数据集进行合理的清洗和划分（划分训练集、验证集和测试集）
- 安装相关依赖，成功运行训练代码，在cuda平台上使用BERT类的模型训练二分类检测器，并在验证集和测试集上验证检测准确率。

进阶内容

AI检测器目前大致有两种做法，一种是使用监督学习的方法（Encoder做微调），需要使用BERT等预训练模型使用带标签数据训练。另一种是零样本检测方案（Decoder推理每个Token的生成概率，再基于概率寻找人机的特征）

- 复现学术界（例如fast-detectGPT）的零样本检测方法（注意该方法需要计算资源较高，建议寻找导师借用实验室的RTX 4090显卡）
- 优化监督学习的检测效果，可以寻找开源数据集训练，对比性能提升（请确保人机文本尽可能平衡，数据领域尽可能广泛全面）。

服务器任务

任务目标

- 使用python的flask工具构建后端代码（后端代码是检测服务所在的代码，前端代码是Web页面和用户交互的代码）。
- 使用AI工具或者从网上找相关前端模板，实现前端代码。如果使用AI工具，建议让AI使用vue框架搭建项目，且建议给AI提供你们希望实现的网站的风格，让AI照着风格实现代码。
- 配置Nginx服务（在Linux中部署）

进阶内容

- 构建数据库和用户系统。
- 考虑使用docker部署整套环境。
- 部署阿里云服务器（A10显卡按量付费≈10元/h），将服务从本地（127.0.0.1）部署到公网。
- 学习计算机网络。

其他说明

- 建议各个同学之间相互了解彼此的工作，这些内容是计算机专业的基础。
- 后续想继续推进AI生成文本检测项目的同学，非常欢迎能够参与我们当前的工作。
- 实现过程中遇到困难无法解决的，请在交流群中交流或与我沟通。