2026年过程化多模态智能技术与应用赛项样题

一、初始化环境

1.账号及默认密码如表1所示。

表1 账号及密码规划表

|  |  |
| --- | --- |
| 账 号 | 密 码 |
| 20251001（教师端） | a123456 |
| 20252004（学生端） | a123456 |

表2 服务器端的账号及密码

|  |  |  |
| --- | --- | --- |
| 账 号 | 密 码 | 端口号 |
| user1 | 1234 | 8001 |
| user2 | 1234 | 8002 |
| user3 | 1234 | 8003 |
| user4 | 1234 | 8004 |
| user5 | 1234 | 8005 |
|  |  |  |

2.使用conda安装程序执行所需软件包，程序所需软件包及其对应版本包括如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **软件包1** | **版本1** | **软件包2** | **版本2** |
| absl-py | 2.3.1 | astunparse | 1.6.3 |
| atomicwrites | 1.4.1 | attrs | 25.4.0 |
| beautifulsoup4 | 4.14.2 | bidict | 0.23.1 |
| blinker | 1.9.0 | certifi | 2025.11.12 |
| charset-normalizer | 3.4.4 | click | 8.3.1 |
| colorama | 0.4.6 | contourpy | 1.3.2 |
| cycler | 0.12.1 | deepface | 0.0.96 |
| dnspython | 2.8.0 | eventlet | 0.40.4 |
| filelock | 3.20.0 | fire | 0.7.1 |
| Flask | 3.1.2 | flask-cors | 6.0.1 |
| Flask-SocketIO | 5.5.1 | flatbuffers | 25.9.23 |
| fonttools | 4.60.1 | fsspec | 2025.10.0 |
| gast | 0.6.0 | gdown | 5.2.0 |
| google-pasta | 0.2.0 | greenlet | 3.2.4 |
| grpcio | 1.76.0 | gunicorn | 23.0.0 |
| h11 | 0.16.0 | h5py | 3.15.1 |
| idna | 3.11 | iniconfig | 2.3.0 |
| intel-openmp | 2021.4.0 | itsdangerous | 2.2.0 |
| Jinja2 | 3.1.6 | joblib | 1.5.2 |
| keras | 3.12.0 | kiwisolver | 1.4.9 |
| absl-py | 2.3.1 | astunparse | 1.6.3 |
| atomicwrites | 1.4.1 | attrs | 25.4.0 |
| beautifulsoup4 | 4.14.2 | bidict | 0.23.1 |
| blinker | 1.9.0 | certifi | 2025.11.12 |
| charset-normalizer | 3.4.4 | click | 8.3.1 |
| colorama | 0.4.6 | contourpy | 1.3.2 |
| cycler | 0.12.1 | deepface | 0.0.96 |

|  |  |  |  |
| --- | --- | --- | --- |
| dnspython | 2.8.0 | eventlet | 0.40.4 |
| filelock | 3.20.0 | fire | 0.7.1 |
| Flask | 3.1.2 | flask-cors | 6.0.1 |
| Flask-SocketIO | 5.5.1 | flatbuffers | 25.9.23 |
| fonttools | 4.60.1 | fsspec | 2025.10.0 |
| gast | 0.6.0 | gdown | 5.2.0 |
| google-pasta | 0.2.0 | greenlet | 3.2.4 |
| grpcio | 1.76.0 | gunicorn | 23.0.0 |
| h11 | 0.16.0 | h5py | 3.15.1 |
| idna | 3.11 | iniconfig | 2.3.0 |
| intel-openmp | 2021.4.0 | itsdangerous | 2.2.0 |
| Jinja2 | 3.1.6 | joblib | 1.5.2 |
| keras | 3.12.0 | kiwisolver | 1.4.9 |
| libclang | 18.1.1 | lightecc | 0.0.3 |
| lightphe | 0.0.15 | lz4 | 4.4.5 |
| Markdown | 3.10 | markdown-it-py | 4.0.0 |
| MarkupSafe | 3.0.3 | matplotlib | 3.10.7 |
| mdurl | 0.1.2 | mkl | 2021.4.0 |
| ml\_dtypes | 0.5.4 | mpmath | 1.3.0 |
| mtcnn | 1.0.0 | mysql | 0.0.3 |
| mysql-connector-python | 9.3.0 | mysqlclient | 2.2.7 |
| namex | 0.1.0 | neo4j | 6.0.3 |
| networkx | 3.4.2 | numpy | 1.26.0 |
| opencv-python | 4.12.0.88 | opt\_einsum | 3.4.0 |
| optree | 0.18.0 | packaging | 25.0 |
| pandas | 2.3.3 | pillow | 12.0.0 |
| pip | 25.3 | pluggy | 1.6.0 |
| polars | 1.35.2 | polars-runtime-32 | 1.35.2 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| protobuf | 6.33.1 | psutil | 7.1.3 | protobuf |
| py | 1.11.0 | Pygments | 2.19.2 | py |
| pyparsing | 3.2.5 | PySocks | 1.7.1 | pyparsing |
| pytest | 7.1.2 | python-dateutil | 2.9.0.post0 | pytest |
| python-dotenv | 1.2.1 | python-engineio | 4.12.3 | python-dotenv |
| python-socketio | 5.15.0 | pytz | 2025.2 | python-socketio |
| PyYAML | 6.0.3 | requests | 2.32.5 | PyYAML |
| retina-face | 0.0.17 | rich | 14.2.0 | retina-face |
| scipy | 1.15.3 | seaborn | 0.13.2 | scipy |
| setuptools | 80.9.0 | simple-websocket | 1.1.0 | setuptools |
| six | 1.17.0 | soupsieve | 2.8 | six |
| sympy | 1.14.0 | tbb | 2021.11.0 | sympy |
| tensorboard | 2.20.0 | tensorboard-data-server | 0.7.2 | tensorboard |
| tensorflow | 2.20.0 | termcolor | 3.2.0 | tensorflow |
| tf\_keras | 2.20.1 | tomli | 2.3.0 | tf\_keras |

二、任务描述

为了更好地利用人工智能技术提升教学评价效能，同时为学生提供便捷、可视、智能的过程性考核支持，请您以教师身份，依据上述配置环境及以下任务要求，完成过程性考核的配置与测试。

人工智能与工业控制实训平台任务

任务1. 基础软硬件环境配置

（1）启动后端服务：

检查主机nvidia驱动版本，nvidia-smi命令行工具 （学生端）（无此

操作扣2分）

（2）校准摄像头：海康robot，要求完整覆盖四排触点，触点细节清晰可（学生端）（无此操作扣2分）

（3）安装conda，选择对应包。

任务2. 图像数据处理（Python 3.10 + CUDA 11.8+‌）

（1）照片读取和标注（labelling根据给定的图片1张，做数据标注，程序判定标准区域）

（2）数据增强（Alubumentation）（如何验证，打开图片验证（train and val）

（3）利用YOLO模型对标注的数据进行训练（1个epoch）

任务3. 教师端接线规则的设定

（4）运用电气知识，将给出的电气接线题中复杂的电路图转化为对应的接线触点对（详见附件2）。

（5）生成可视化的规则制定图形界面，根据转化的接线触点对，操作图形界面生成规则文件。（代码实现）

（6）导出json文件，并在学生端测试规则是否正确可用

任务4. 大语言模型的部署（服务器端）

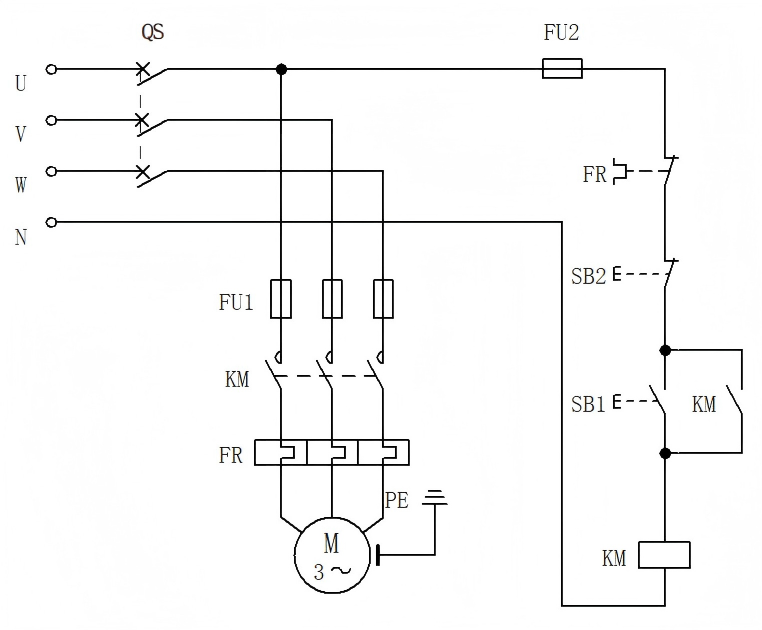
（1）安装cuda驱动（选择配置包），添加环境变量到linux系统（下载安装包，环境变量）（学生端）

（2）完成大模型的本地部署工作（提前下载Ollama/Ollama，deepseek，放到环境变量里）：模型平台加载deepseek（代码实现），启动，测试模型加载正确，往11434端口下发送命令，看是否能通信。 （服务器端）

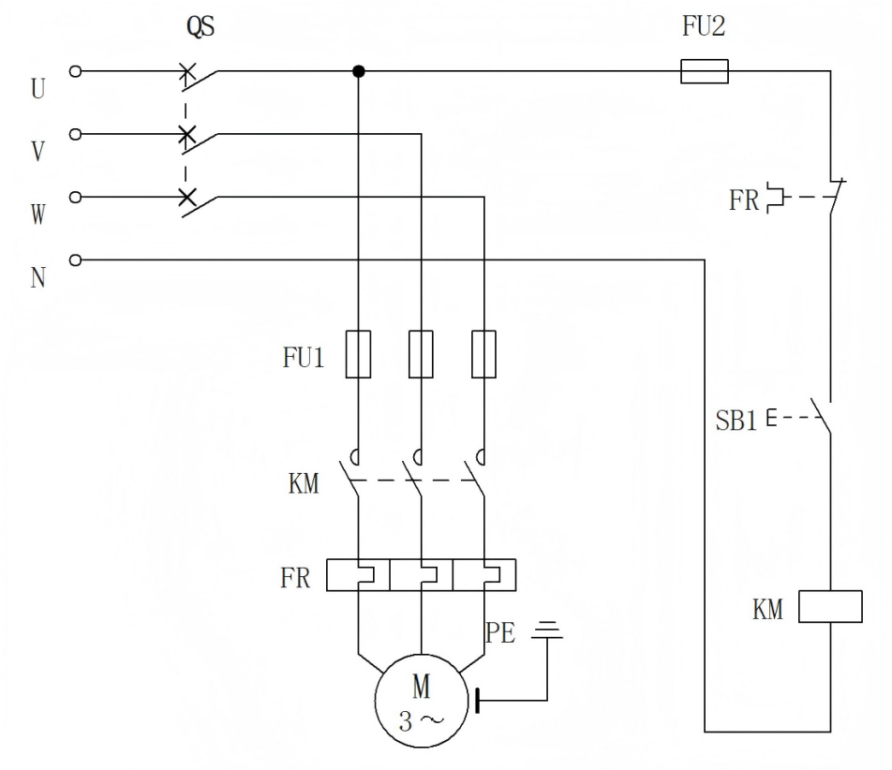
（3）设计带有ui界面的程序，实现和服务器模型端的通信，并且实现与模型的对话，能将模型输出的结果**流式**返回到程序对话框。

（4）能够得到服务器模型指定问题的答案。

附件2：



电气原理图1



电气原理图2