Project Environment Setup Guide

(all setup and testing was done on windows)

This guide walks you through setting up the two Python virtual environments required for this project:

- `S25_34_Demo`: For running notebooks and demonstrations
- `S25_34_Training`: For training-related workflows

All project files, including this guide, are assumed to be located in the same folder.

---

1. Navigate to the Project Directory

Open a terminal or command prompt and change into the project folder:

```
cd path/to/project-folder
```

Replace `path/to/project-folder` with the actual location where this project is stored.

---

2. Create Virtual Environments

You will create two separate virtual environments.

A. Create the Demo Environment (python 3.11.9 was used)

```
python -m venv S25_34_Demo
```

B. Create the Training Environment

```
python -m venv S25_34_Training
```

---

3. Activate the Environment

Use the appropriate command based on your operating system.

A. Activate Demo Environment

Windows:

```
S25_34_Demo\Scripts\activate
```

B. Activate Training Environment

Windows:

```
S25_34_Training\Scripts\activate
```

---

4. Install Dependencies

Install the required packages for each environment.

Warning: If you encounter a build error with `annoy`, ensure you have the Desktop development with C++ workload installed via the Visual Studio Installer

A. For the Demo Environment

install notebook for both environments

```
pip install notebook
```

```
pip install -r requirements-DEMO.txt
```

```
pip install torch torchvision torchaudio
```

B. For the Training Environment

```
pip install notebook
```

```
pip install -r requirements-TRAIN.txt
```

```
pip install torch torchvision torchaudio
```

---

5. Register Jupyter Kernels

Register each environment so they appear in Jupyter Notebook.

A. Register Demo Environment Kernel

```
python -m ipykernel install --user --name=notebook-tui-demo
--display-name "Python (Demo_Env)"
```

B. Register Training Environment Kernel

```
python -m ipykernel install --user --name=notebook-tui-training
--display-name "Python (Training_Env)"
```

---

6. Launch Jupyter Notebook

Start Jupyter using either environment:

```
jupyter notebook
```

Then, open any notebook and go to:

```
Kernel > Change Kernel > Python (Demo_Env)  [for demo notebooks]
```

```
Kernel > Change Kernel > Python (Training_Env) [for training
notebooks]
```

Guardrails Modification

Config file is located within \textconfig or \imageconfig folder named config.yml

Modify engine and model type to correspond to chosen LLM model

Modify the field contained within instructions for LLM instructions on operation

Modify the field contained within self_check_input to instruct the LLM on checking the prompt given for verification

When to use Which Environment

Demo Environment:

- Gui and tui demo
- Nemo tests

Training Environment:

- Training notebook
- Tuning notebook
- Model evaluation script

If either demo gives you an error code 400, this is a DALLE response, for example:
If DALLE is given the prompt: “give me an image” DALLE returns error code 400 “invalid prompt”. We believe this prompt does not give the model enough to work with but we are not certain. This is not an error code in our system