# School Health AI v1.3.9 - Installation Guide

## 1. Recommended Environment

* **Operating System**: Windows 10/11 (64-bit)
* **Python Version**: **3.10.x** (Python 3.10.11 is recommended)
* **GPU**: A dedicated GPU (NVIDIA RTX series) is recommended for better performance in offline mode.
* **Minimum RAM**: 16 GB

## 2. Install Python and Virtual Environment

1. Download Python 3.10.x. During installation, make sure to check the box for **"Add Python to PATH"**.
2. Open a Terminal (CMD or PowerShell) and navigate to the project's root folder:
3. cd <path\_to\_project\_folder>
4. Create the virtual environment:
5. python -m venv venv
6. Activate the virtual environment:
7. venv\Scripts\activate.bat

## 3. Install Required Libraries

After activating the virtual environment (you will see (venv) at the beginning of the command line), run the following command:

pip install -r requirements.txt

## 4. Main Libraries

* transformers & torch: Core libraries for running Hugging Face AI models.
* scikit-learn: Crucial for automatic topic discovery (clustering).
* sentence-transformers: Essential for contextual reasoning and semantic topic discovery.
* matplotlib & reportlab: Used for plotting charts and exporting the PDF report.
* pyyaml: Used to read the config.yaml configuration file.
* pandas & openai: Necessary for advanced report generation and optional OpenAI refinement.

## 5. Configuration (Very Important!)

All settings are managed in the config.yaml file. Choose one mode:

**A. Offline Mode (Recommended - Runs locally, downloads models once)**

* Ensure the following settings are in config.yaml:

huggingface:

offline\_mode: false # Set to 'false' initially to allow model download.

# After the first successful run, you can change this to 'true' for purely offline operation.

openai:

enabled: false # Ensure OpenAI refinement is disabled for full offline mode.

**B. Online Mode (Uses OpenAI for best results - Requires API Key)**

* To enable the high-quality OpenAI refinement step:
* Open config.yaml and edit the openai section:

openai:

enabled: true

api\_key: 'sk-...' # Paste your OpenAI API key here

**C. Hugging Face Inference API Mode (Alternative Online Mode - Sends data to HF)**

This mode runs classification on Hugging Face servers instead of locally. It requires specific top-level keys in config.yaml (as checked by modules/group\_classifier.py):

# Add these keys at the TOP LEVEL of config.yaml

mode: hf\_online

hf\_token: "hf\_..." # Paste your Hugging Face API token (read permissions needed)

# You might also need to adjust settings within the 'huggingface:' block

huggingface:

enabled: true # Ensure HF features are generally enabled

# ... other settings might apply depending on specific module usage

openai:

enabled: false # Usually disable OpenAI if using HF API for main classification

## 6. Run the Program

1. Make sure you are in the project's root directory and have activated the virtual environment.
2. Run the main demo script:
3. python run\_demo.py

(Troubleshooting: If you get a *PermissionError* when the script tries to save the PDF, make sure you have closed any previously generated PDF reports.)

## 7. Review the Outputs

After the script finishes, the most important output files will be:

* **PDF Reports**:
  + reports/summary\_report\_v1.3.0.pdf: Based on the initial, raw analysis.
  + reports/summary\_report\_v1.3.3.pdf: Based on the **refined** results after contextual reasoning.
* **Detailed JSON Data**:
  + data/processed\_results\_v1.3.0.json: Raw results from initial analysis.
  + data/final\_results\_v1.3.3.json: Results after the contextual reasoning step.
  + data/final\_results\_v1.3.4.json: (If OpenAI enabled) The **highest-quality results** refined by the LLM.