Setup Guide: Text-Based Misinformation Detector

Before You Start

System Requirements: This prototype can be set up on Windows, Mac, or Linux systems.

Required Software Installations: You will need Python (version 3.6 or higher) and pip (Python package installer).

Estimated Setup Time: Approximately 30-60 minutes, depending on your familiarity with software installation.

Step 1: Install Python

Download Python:

Go to the official Python website.

Choose the version suitable for your operating system (Windows, Mac, or Linux) and click on the download link.

Install Python:

Windows: Run the downloaded installer and ensure you check the box that says "Add Python to PATH" before clicking "Install Now".

Mac: Open the downloaded `.pkg` file and follow the installation instructions.

Linux: Open a terminal and run the following command:

```
sudo apt-get install python3
```

Verify Installation:

Open a terminal (Command Prompt on Windows, Terminal on Mac/Linux).

Type the following command and press Enter:

```
python --version
```

You should see the installed Python version. If not, check your installation steps.

Common Troubleshooting:

If you receive an error that Python is not recognized, ensure that it is added to your system PATH during installation.

Step 2: Download Project Files

Download the Project Files:

You can download the project files from the provided source (e.g., a GitHub repository or a shared link).

Extract the Project Files:

If the files are in a compressed format (like .zip), right-click on the file and select "Extract All" (Windows) or double-click to unzip (Mac).

Understanding the File Structure:

The main folder will contain the following files:

- `misinformation_detector.py` : The main script to run the detector.
- `text_processor.py` : Handles text input and preprocessing.
- `model_trainer.py` : Trains the AI model on data.
- `credibility_scorer.py` : Scores the credibility of the text.
- `gui_interface.py` : Provides a graphical user interface.
- `config.yaml` : Configuration settings for the application.

Important Files Overview:

Familiarize yourself with these files as they will be essential for running and configuring the application.

Step 3: Install Dependencies

Open a Terminal:

Navigate to the folder where you extracted the project files.

Run the Requirements Installation:

Execute the following command to install the necessary dependencies:

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

If there is no `requirements.txt`, you may need to install dependencies manually based on the imports in the Python files.

What Each Major Dependency Does:

Libraries like `nltk`, `scikit-learn`, and `pandas` are commonly used for natural language processing and data handling.

Troubleshooting Installation Issues:

If you encounter errors during installation, ensure you have an active internet connection and that pip is updated:

```
pip install --upgrade pip
```

Step 4: First Run

How to Start the Application:

In the terminal, run the following command:

```
python misinformation_detector.py
```

What to Expect on First Launch:

The application should open a GUI window. You may see a welcome message and options to input text for analysis.

Initial Configuration if Needed:

If prompted, you may need to configure settings in the `config.yaml` file. Open it in a text editor and adjust parameters as necessary.

Step 5: Test with Sample Data

Step 3: Test with Sample Data

Using the Included Test Examples:

The application may come with sample text files. Load these to see how the detector works.

Loading Your Own Data (if applicable):

You can input your own text directly into the GUI or load from a file, depending on the interface options.

Verifying Everything Works Correctly:

Check the output for credibility scores and ensure the application responds as expected.

Troubleshooting

Common Error Messages and Solutions:

If you see "Module not found", ensure all dependencies are installed correctly.

Performance Issues and Fixes:

If the application runs slowly, consider closing other applications or checking your system resources.

When to Seek Technical Help:

If you encounter persistent issues, consider reaching out for help.

Getting Help

Log File Locations:

Check the project folder for any log files that may provide error details.

Information to Include When Asking for Help:

Provide details about your operating system, Python version, and any error messages you received.

Next Steps if Setup Fails:

Review the troubleshooting section, and if issues persist, consider seeking help from a

technical colleague or community forum.

By following these steps, you should be able to successfully set up and test the Text-Based Misinformation Detector prototype. Good luck!