TruthLens

How to Run the Project

Installation Guide

To set up and run the TruthLens project, follow the steps below:

1. Clone the Repository

First, clone the TruthLens repository from GitHub. This will download the project files to your local machine.

```
git clone https://github.com/soodaryan/TruthLens.git
```

2. Navigate to the Project Directory

Change to the directory where the project has been downloaded. You can do this by using the following command:

```
cd TruthLens
```

3. Set Up a Virtual Environment

It is recommended to create a virtual environment to manage dependencies. Run the following commands to set it up:
For Linux/macOS:

```
python3 -m venv venv
source venv/bin/activate
```

For Windows:

```
python3 -m venv venv
venv/bin/activate
```

4. Install Dependencies

Install the required dependencies using the pip package manager. This will ensure all necessary libraries and packages are available for the project to run.

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

5. Run the Backend

The backend of the application can be started by running the main.py file. To do so, use the following command:

```
python backend/main.py
```

6. **Start the Frontend**

   To start the frontend, navigate to the frontend directory, install the required frontend dependencies using npm, and then start the frontend application.

```
cd frontend
npm install
npm start
```

Run Specific Functionalities

To test specific functionalities, you can run individual files for each type of detection. Below are the commands to run the different functionalities:

1. **Text-based Functionalities**

   For running text-based functionalities, such as Fake News Detection, execute the following:

```
python text/fake_news_detection.py
```

2. **Audio-based Functionalities**

   To run audio-related functionalities like Speech to Text or Sentiment Analysis, use the following commands:

   a. For **Speech-to-text** functionality:

```
python audio/SpeechToText/AudioProcessor.py
```

   b. For **Audio Sentiment Analysis** functionality:

```
python audio/AudioSentimentAnalysis.py
```

3. **Image-based Functionalities**

   For image-based functionalities, such as Deepfake Detection or Optical Character Recognition (OCR), use the following commands:

   a. For **Deepfake Detection**:

```
python images/deepfake_detection.py
```

   b. For **Optical Character Recognition (OCR)**

```
python images/OCR.py
```