```
# Jeffery Dirden / Module 4 / ITAI 2372

# Finance Insights App

This is a simple app that performs sentiment analysis
finance-related text using machine learning and natur
(NLP) techniques.

## Features

- **Sentiment Analysis:** Analyzes the sentiment of f
(positive or negative).
- **Text Summarization:** Generates a short summary o

## Prerequisites

Before you run the application, make sure you have th

- Python 3.7 or higher
- pip (Python package installer)

## Installation

1. **Clone the repository:**

    ```bash
    git clone https://github.com/your-username/financ
    cd finance-insights-app
    ```

2. **Create and activate a virtual environment (optio

    ```bash
    python -m venv venv
    source venv/bin/activate  # On Windows, use `venv
    ```

3. **Install the required dependencies:**

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

    The `requirements.txt` file should contain all th

    ```
    pandas
    joblib
    nltk
    gradio
    scikit-learn
    transformers
    ```

4. **Download NLTK data:**

    The app uses NLTK stopwords for cleaning the text
    be downloaded when running the app for the first
    download it by running:

    ```python
```

Jeffery Dirden / Module 4 / ITAI 2372

Finance Insights App

This is a simple app that performs sentiment analysis and text summarization on finance-related text using machine learning and natural language processing (NLP) techniques.

Features

- **Sentiment Analysis:** Analyzes the sentiment of finance-related text (positive or negative).
- **Text Summarization:** Generates a short summary of the finance-related text.

Prerequisites

Before you run the application, make sure you have the following installed:

- Python 3.7 or higher
- pip (Python package installer)

Installation

1. **Clone the repository:**

   ```
   git clone https://github.com/your-username/f
   cd finance-insights-app
   ```

2. **Create and activate a virtual environment (optional but recommended):**

   ```
   python -m venv venv
   source venv/bin/activate  # On Windows, use
   ```

3. **Install the required dependencies:**

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

   The `requirements.txt` file should contain all the necessary libraries:

   ```
   pandas
   joblib
   nltk
   ```
   ```

```
    import nltk
    nltk.download('stopwords')
    nltk.download('punkt')
    ```

Running the Application

1. **Train the sentiment model and launch the Gradio

    ```bash
    python app.py
    ```

    This will train the sentiment model and open a Gr
    browser. If it's the first time running the app,
    vectorizer will be saved as `.pkl` files for late

2. **Using the application:**

    - Paste finance-related text into the provided te
    - The app will analyze the sentiment (positive or
    summary of the text.

Testing the Application

To test the application:

1. Run the app as mentioned above.
2. Enter some finance-related text in the Gradio UI,
sentiment analysis result and a summary.

Example Inputs:

- "Stock prices for Company A soared today, boosting
- "Analysts warn that a recession is imminent, leadin
sentiment."

Example Outputs:

- **Sentiment:** Positive
- **Summary:** "Stock prices for Company A soared tod
confidence."

Dependencies

- `pandas`: For data manipulation.
- `joblib`: For saving and loading the trained model.
- `nltk`: For text preprocessing.
- `gradio`: For building the user interface.
- `scikit-learn`: For machine learning model (Naive B
- `transformers`: For text summarization using the BA

License

This project is licensed under the MIT License - see
file for details.

```
    gradio
    scikit-learn
    transformers
```

4. **Download NLTK data:**

    The app uses NLTK stopwords for cleaning the text. The necessary data will be downloaded when running the app for the first time, or you can manually download it by running:

```
    import nltk
    nltk.download('stopwords')
    nltk.download('punkt')
```

Running the Application

1. **Train the sentiment model and launch the Gradio interface:**

```
    python app.py
```

    This will train the sentiment model and open a Gradio interface in your browser. If it's the first time running the app, the sentiment model and vectorizer will be saved as `.pkl` files for later use.

2. **Using the application:**
    - Paste finance-related text into the provided textbox.
    - The app will analyze the sentiment (positive or negative) and provide a summary of the text.

Testing the Application

To test the application:

1. Run the app as mentioned above.
2. Enter some finance-related text in the Gradio UI, and you will see the sentiment analysis result and a summary.

Example Inputs:

- "Stock prices for Company A soared today, boosting investor confidence."
- "Analysts warn that a recession is imminent, leading to a bearish market sentiment."

Example Outputs:

- **Sentiment:** Positive

- **Summary:** "Stock prices for Company A soared today, boosting investor confidence."

Dependencies

- `pandas` : For data manipulation.
- `joblib` : For saving and loading the trained model.
- `nltk` : For text preprocessing.
- `gradio` : For building the user interface.
- `scikit-learn` : For machine learning model (Naive Bayes classifier).
- `transformers` : For text summarization using the BART model.

License