**Technical Setup Document: Political/Financial News Sentiment Analysis Pipeline**

**Overview**

This document outlines the steps required to set up a Python environment and access the Newsdata.io API for performing sentiment analysis on news articles using HuggingFace Transformers. The use case focuses on extracting and analyzing political and financial news for North American regions.

**1. Environment Setup**

**1.1. Create Project Directory**

mkdir political-sentiment-index && cd political-sentiment-index

**1.2. Create and Activate a Conda Environment**

conda create -n senti python=3.10 -y

conda activate senti

**Note**: Avoid Python 3.12 due to compatibility issues with PyTorch.

**1.3. Install Required Packages**

pip install transformers torch requests python-dotenv pandas

**Optional: Jupyter Kernel Support**

pip install notebook ipykernel

python -m ipykernel install --user --name=senti

**2. Newsdata.io API Access**

**2.1. Register and Retrieve API Key**

* Go to: https://newsdata.io/
* Sign up and verify your account
* Navigate to your **Dashboard > API Key**
* Copy your API key

**2.2. Create .env File**

touch .env

Add the following content:

NEWSDATA\_API\_KEY=your\_api\_key\_here

NEWS\_CATEGORY=business

**3. Newsdata.io API Integration**

**3.1. Endpoint**

https://newsdata.io/api/1/news

**3.2. Allowed Parameters**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Required** | **Description** |
| apikey | Yes | Your personal API key |
| country | Yes | Country code (e.g., us, ca) |
| language | Optional | Language code (en recommended) |
| category | Optional | News category (e.g., business) |

**Important**: from\_date and to\_date are not allowed in the free tier for the /news endpoint.

**3.3. Sample API Call in Python**

import os

import requests

from dotenv import load\_dotenv

load\_dotenv()

api\_key = os.getenv("NEWSDATA\_API\_KEY")

url = "https://newsdata.io/api/1/news"

params = {

"apikey": api\_key,

"country": "us",

"language": "en",

"category": "business"

}

response = requests.get(url, params=params)

data = response.json()

print(data)

**4. Running the Full Pipeline**

Ensure you have the complete script named news\_sentiment\_pipeline.py. Then run:

python news\_sentiment\_pipeline.py

**5. Supported News Categories**

The following categories are supported:

* business
* crime
* domestic
* education
* entertainment
* environment
* food
* health
* lifestyle
* other
* politics
* science
* sports
* technology
* top
* tourism
* world

**Next Steps**

* Add automatic retries and logging to the pipeline
* Export results to CSV for dashboard integration
* Add scheduling (e.g., via cron) for automated daily fetches