# 📰 AI News Article Extractor Agent

This project is an intelligent news article extractor powered by \*\*Gemini 1.5 Pro (Google Generative AI)\*\* and built using \*\*Streamlit\*\*. It scrapes a news article from a given URL, extracts structured metadata using LangChain and Gemini, stores the data in \*\*Airtable\*\*, and \*\*automatically sends an email via Zapier\*\* if the article is related to \*\*Trade\*\*.

---

## 🔧 Features

- ✅ Scrapes full article content from any public URL

- ✅ Uses Gemini Pro to extract key metadata:

- `headline` (max 7 words)

- `date` (YYYY-MM-DD format)

- `country` (ISO 3166-1 alpha-2)

- `category` (Trade, Economy, Politics, etc.)

- ✅ Stores extracted data in \*\*Airtable\*\*

- ✅ Auto email alert via \*\*Zapier\*\* for Trade articles

- ✅ Simple Streamlit interface

---

## 🛠 Tech Stack

- Python

- Streamlit

- LangChain

- Google Gemini 1.5 Pro (Generative AI)

- Airtable API

- BeautifulSoup (HTML parsing)

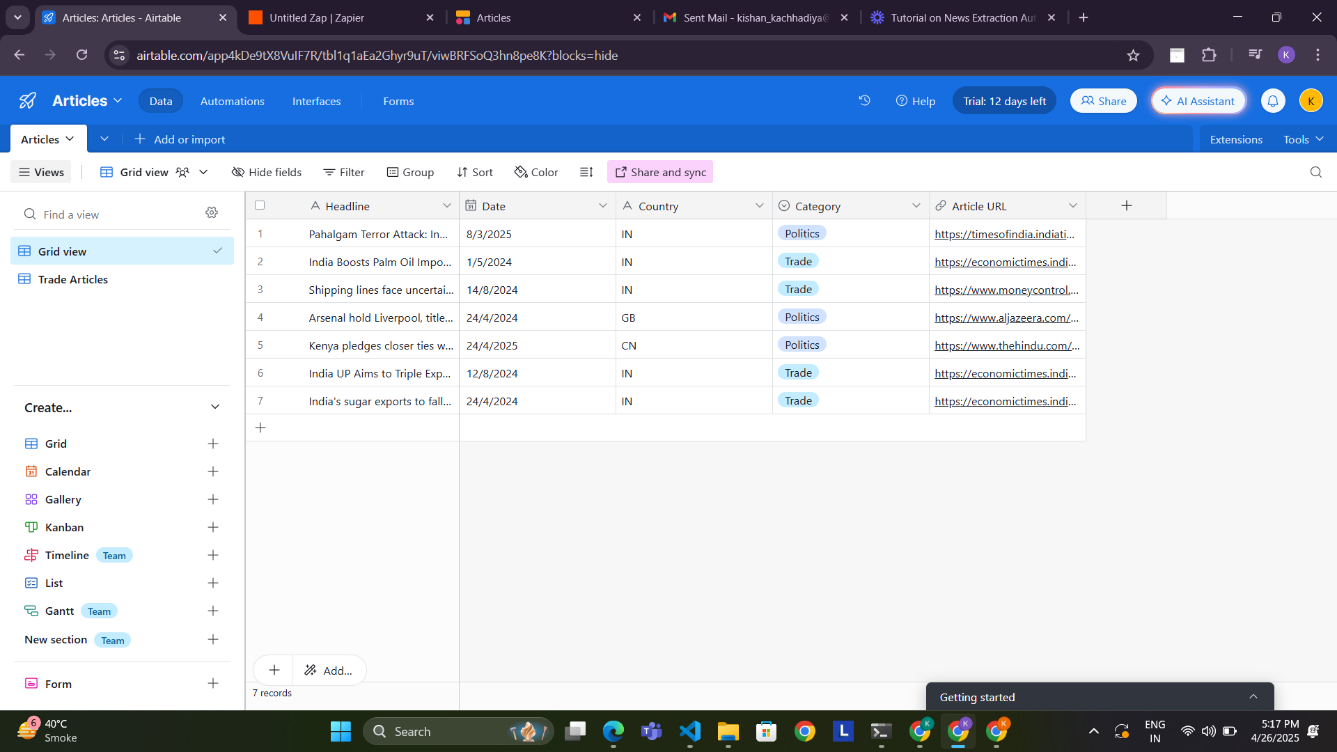
- Zapier (email automation)

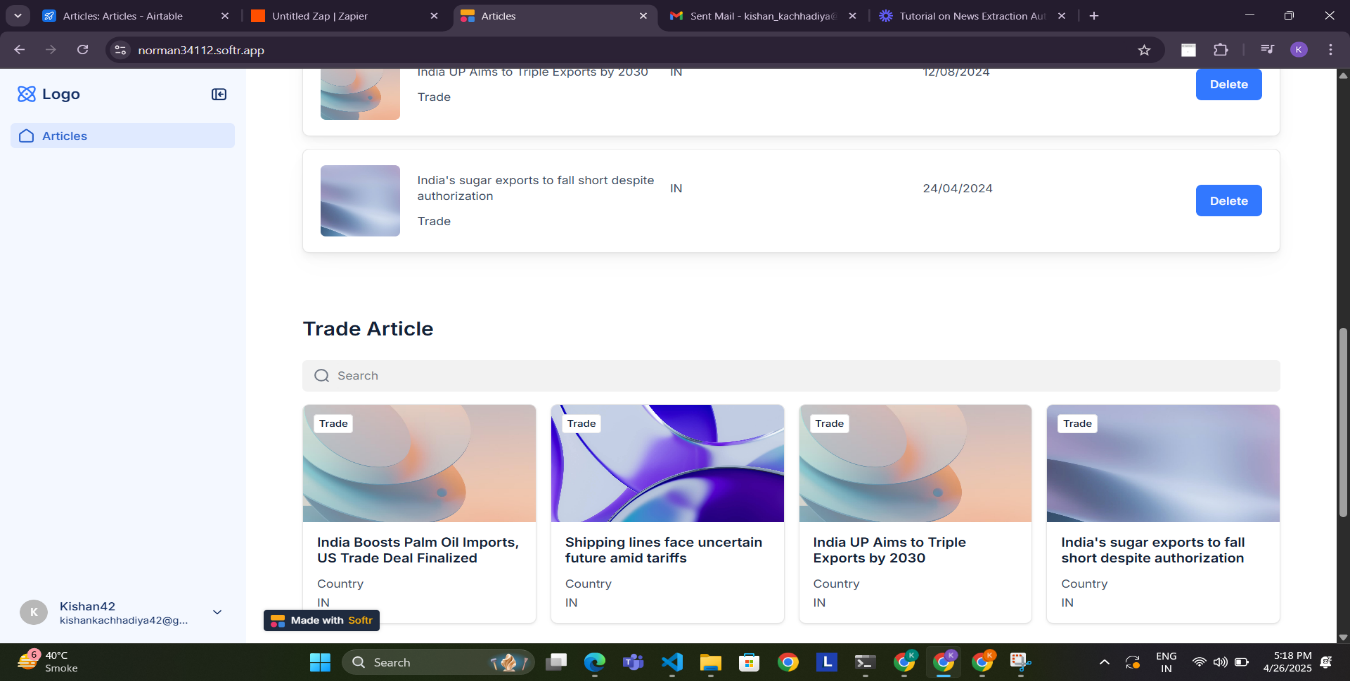
- Regex + JSON parsing

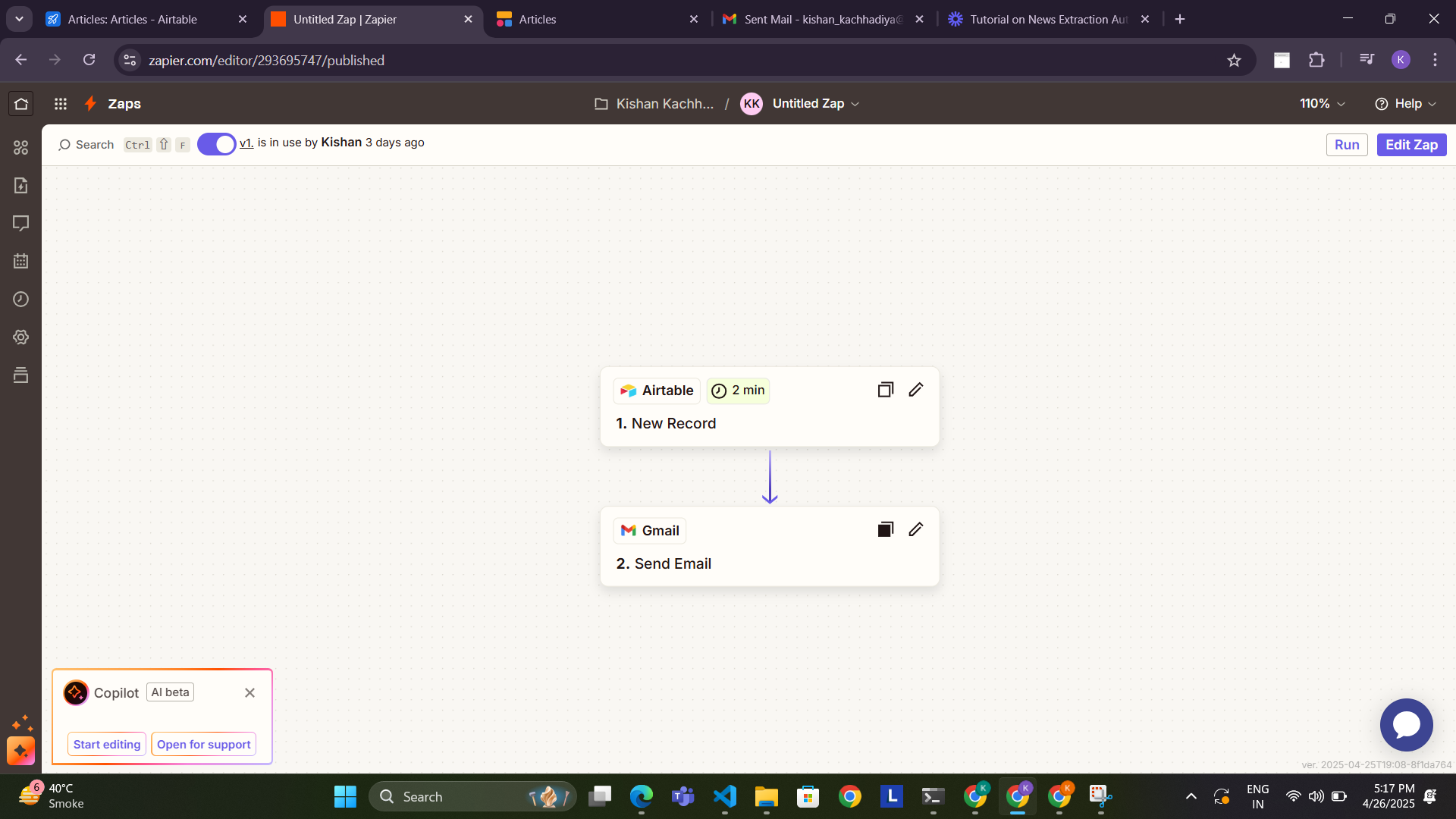
- Requests (HTTP client)

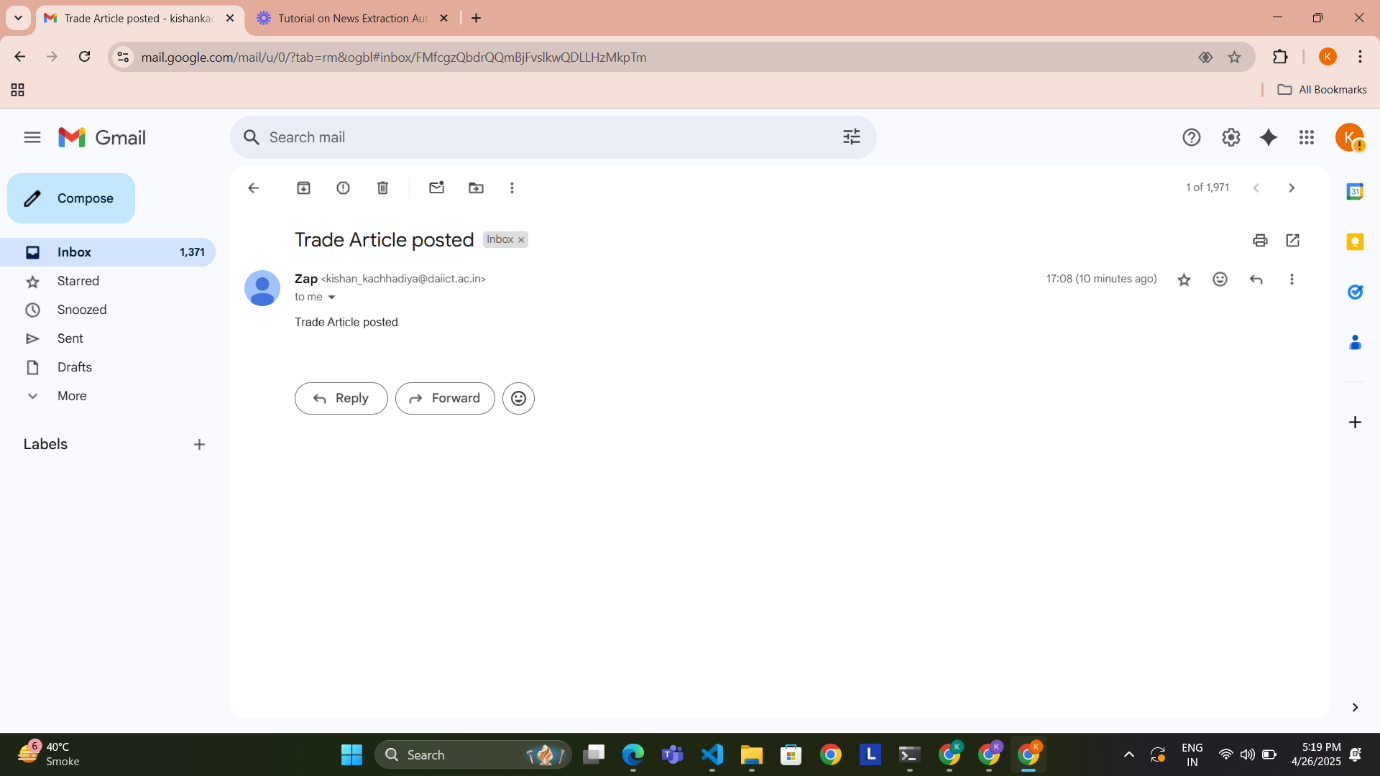
---

## 🚀 Demo









---

## 📦 Installation

```bash

git clone https://github.com/yourusername/news-extractor-agent.git

cd news-extractor-agent

pip install -r requirements.txt

**🧪 Environment Setup**

Create a .env file in your root directory:

GEMINI\_API\_KEY=your\_google\_gemini\_api\_key

AIRTABLE\_API\_KEY=your\_airtable\_api\_key

AIRTABLE\_BASE\_ID=your\_airtable\_base\_id

AIRTABLE\_TABLE\_NAME=Articles

**▶️ Usage**

Run the app using Streamlit:

streamlit run app.py

1. Paste a valid article URL into the input box.
2. Click **"Submit URL"**.
3. The app will:
   * Scrape the article content.
   * Extract structured metadata via Gemini.
   * Display the results.
   * Store the info in Airtable.
   * (If category = "Trade") trigger a Zapier email alert.

**📄 Example Output**

{

"headline": "India Expands Global Trade",

"date": "2025-06-08",

"country": "IN",

"category": "Trade"

}

**🧠 Gemini Prompt Overview**

The Gemini model is prompted with a detailed system message and the first 5000 characters of the article:

* Output format is strictly JSON.
* Categories must be from a defined list.
* Prompts exclude markdown (```json) to ease parsing.

**📤 Airtable Integration**

Create a base in Airtable with a table named Articles and the following fields:

| **Field** | **Type** |
| --- | --- |
| Headline | Single line |
| Date | Date |
| Country | Single line |
| Category | Single select (Trade, Politics, etc.) |
| Article URL | URL |

The app posts extracted data using Airtable’s REST API.

**📧 Email Alerts via Zapier**

This project uses **Zapier** to send automated email alerts when a new Airtable record is added with category = Trade.

**🔄 Zapier Workflow**

* **Trigger App**: Airtable
  + Trigger: New Record in View
  + Filter: Category = Trade
* **Action App**: Gmail / Email by Zapier
  + Action: Send Email
  + Email Template:
  + 📢 New Trade Article Alert!
  + Headline: {{Headline}}
  + Date: {{Date}}
  + Country: {{Country}}
  + Read more: {{Article URL}}

✨ You can customize the Zap to send notifications to Slack, Telegram, Discord, or even SMS.

**📁 File Structure**

📦news-extractor-agent

┣ 📄app.py # Main Streamlit app

┣ 📄.env # Environment config

┣ 📄requirements.txt # Python dependencies

┗ 📄README.md # Project documentation

**🗺️ Future Improvements**

* 🔍 Multilingual article support
* 🧠 Smarter category classification using fine-tuned model
* 🧾 Richer metadata extraction (author, location, summary)
* 🔗 Social media sharing of extracted articles
* 📊 Airtable dashboard for analytics