

News Summarization and Text to speech Application

Submitted by: Karthik BR

Github link: https://github.com/Robbin808/News_Summarizer

Hugging face deployment link: https://huggingface.co/spaces/karthik808/News_Summarizer

📌 Project Overview

- ◆ **Project Title:** *News Summarization & Sentiment Analysis App*

- ◆ **Objective:**

This project aims to analyse news articles related to a specific company, determine their sentiment (Positive, Neutral, or Negative), and generate a comparative sentiment report. Additionally, it provides Hindi text-to-speech (TTS) conversion for better accessibility.

- ◆ **Problem Statement & Motivation:**

In today's fast-paced world, businesses and investors need real-time insights into how companies are being perceived in the news. However:

- Manually analysing multiple news articles is time-consuming.
- Understanding sentiment trends can be difficult without data-driven insights.
- Language barriers exist for non-English users.

- ◆ **This project solves these problems by:**

- ❖ Automating news collection from multiple sources using News API.
 - ❖ Performing sentiment analysis to classify news as Positive, Neutral, or Negative.
 - ❖ Summarizing insights into a structured comparative sentiment report.
 - ❖ Generating Hindi audio output for accessibility.
-

Summary of the Project:

The News Summarization & Sentiment Analysis App is a full-stack AI-powered web application built using Streamlit (Frontend UI), Flask (Backend API), and NLP-based sentiment analysis tools.

How It Works:

- ❖ User inputs a company name (e.g., Tesla, Apple, Microsoft).
- ❖ App fetches the latest news related to that company from News API.
- ❖ Each news article is analysed for sentiment (Positive, Neutral, or Negative).
- ❖ A sentiment report is generated, showing:
 - Sentiment distribution (e.g., 60% Positive, 20% Neutral, 20% Negative).
 - Key insights & trends extracted from the articles.

- Comparative analysis of different news topics.
5 Hindi TTS Audio is generated to summarize the insights for easy listening.
-

◆ **Key Benefits of This Project:**

- Real-time News Analysis* → Instant sentiment classification from the latest articles.
 - Data-Driven Insights* → Provides a comparative analysis of sentiment trends.
 - Hindi Audio Output* → Ensures accessibility for non-English speakers.
 - User-Friendly Interface* → Streamlit-powered UI for a seamless experience.
-

◆ **Target Users & Use Cases:**

- Business Analysts & Investors* → Understand public perception of companies.
 - Marketing & PR Teams* → Track brand reputation in the media.
 - Journalists & Researchers* → Analyse news trends quickly.
 - General Users & Language Learners* → Get summarized news in Hindi.
-

Features & Functionalities

The News Summarization & Sentiment Analysis App has several key features designed to provide efficient, real-time news analysis with sentiment insights. Below is a detailed breakdown of the features:

◆ **1. Fetch Latest News from News API**

- 1) The app fetches real-time news articles related to a given company using the News API.
- 2) Users enter a company name, and the app retrieves the latest 10 articles from reliable sources
- 3) Each news article includes:
 - Title – The headline of the news article.
 - Summary – A short description or excerpt from the article.
 - Source URL – A direct link to read the full article.

• **Example Output (News Fetching)**

- Company Name: Tesla
1 Tesla Unveils New AI Battery
 -  Read More: <https://news.tesla.com/ai-battery>
 - 2 Tesla Faces Legal Issues Over Self-Driving Tech
 -  Read More: <https://news.tesla.com/lawsuit>
-

◆ **2. Sentiment Analysis for Each News Article**

The app analyzes each news article's summary to determine its sentiment:

-  Positive → Good news, stock growth, company expansion, etc.
-  Neutral → Informational news, no strong bias.
-  Negative → Lawsuits, financial losses, controversies, etc.

Example Sentiment Analysis Output

1. Tesla Unveils New AI Battery

Read More: <https://news.tesla.com/ai-battery>

 Sentiment: Positive

2. Tesla Faces Legal Issues Over Self-Driving Tech

Read More: <https://news.tesla.com/lawsuit>

 Sentiment: Negative

3. Comparative Sentiment Analysis Report

- * The app **compares all news articles** and generates **key insights** on sentiment trends.
- * **Sentiment Distribution Report** → Displays the percentage of **Positive, Neutral, and Negative** articles.
- * **Key Insights** → Highlights common trends & major themes from the news.
- * **Comparative Insights** → Identifies **contrasting news** (e.g., company success vs. legal issues).

Comparative Sentiment Analysis Report:

 Positive Articles: 6

 Neutral Articles: 2

 Negative Articles: 2

 Dominant Sentiment: Positive

◆ 4. Hindi Text-to-Speech (TTS) Generation

- * Converts **key insights** into **Hindi audio output** using **Google Text-to-Speech (gTTS)**.
- * Ensures **accessibility** for users who prefer audio summaries.
- * Uses **Google Translator API** to translate insights into Hindi before generating speech.
- * The generated **MP3 file is playable directly in the app**.

Example Audio Output Flow:

Step 1: Generates insights in English

Step 2: Translates insights to Hindi

Step 3: Converts Hindi text into speech

Step 4: User can play the audio directly in the app

Example Hindi Audio Output:

 "टेस्ला की हाल की खबरें ज्यादातर सकारात्मक हैं, जो बाजार में मजबूत विश्वास को दर्शाती हैं।"

◆ 5. User-Friendly Interface with Streamlit

- 1 The app is built using **Streamlit**, making it **interactive, visually appealing, and easy to use**.
 - 2 Features a **simple text box** where users enter a company name.
 - 3 Automatically displays **news, sentiment analysis, and insights** in a well-organized format.
 - 3 **Audio player included** to listen to the summary in Hindi.
-

◆ 6. API Integration (Backend - Flask)

- 1 Uses **Flask API** to fetch data and process sentiment analysis.
 - 2 Backend provides the following **API endpoints**:
 - /fetch-news?company=Tesla → Retrieves news articles.
 - /analyze-sentiment?company=Tesla → Performs sentiment analysis.
 - /compare-news?company=Tesla → Generates sentiment insights.
 - /generate-tts?text=some_text → Converts text to Hindi speech.
-

📌 System Architecture

This section explains how the **News Summarization & Sentiment Analysis App** works internally, covering:

- ✓ **Overall system workflow**
 - ✓ **Backend & Frontend integration**
 - ✓ **Data flow between components**
-

◆ 1. High-Level Architecture Overview

The system follows a **client-server architecture** where:

- **Frontend (Streamlit UI)** allows users to enter a company name and view results.
 - **Backend (Flask API)** fetches news, analyzes sentiment, and generates insights.
 - **APIs** act as the bridge between the frontend and backend.
 - **External APIs** (NewsAPI, Google Translator, gTTS) provide external services.
-

◆ 2. System Workflow Diagram

Architecture Components:

User → Streamlit UI (Frontend) → Flask API (Backend) → NewsAPI (Fetch news)

TextBlob (Sentiment Analysis) → Google Translator (Hindi Translation) → gTTS (Text-to-Speech)

Streamlit UI (Displays results & Hindi audio output)

◆ 3. Component Breakdown

Frontend (Streamlit UI)

Role: Provides an **interactive user interface** to input a company name and view results.

Technologies Used: Streamlit, HTML, CSS

Key Functionalities:

- ✓ Accepts **company name input** from the user.
 - ✓ Calls **Flask APIs** to fetch news & sentiment analysis.
 - ✓ Displays **news articles, sentiment analysis, and insights**.
 - ✓ Provides an **audio player** to listen to Hindi TTS output.
-

Backend (Flask API)

1 Role: Handles all **data processing, API calls, and computations**.

2 Technologies Used: Flask, Python, Requests, TextBlob, gTTS

3 Key Functionalities:

- ✓ Fetches news using **NewsAPI**.
 - ✓ Performs sentiment analysis using **TextBlob**.
 - ✓ Generates a comparative sentiment report.
 - ✓ Converts insights into Hindi audio using **Google Translator + gTTS**.
 - ✓ Serves data to the Streamlit UI via **API endpoints**.
-

External APIs Used

1 NewsAPI → Fetches real-time news articles about the given company.

2 TextBlob → Performs sentiment analysis on each article summary.

3 Google Translator API → Translates sentiment insights into Hindi.

4 gTTS (Google Text-to-Speech) → Converts Hindi text into speech.

◆ 4. Data Flow in the System

Step-by-Step Data Flow

1 User enters a company name → Streamlit UI sends a request to the Flask API.

2 Flask fetches news articles from NewsAPI.

3 Each article is analyzed for sentiment using TextBlob.

4 A sentiment report is generated (Positive, Neutral, Negative percentages + key insights).

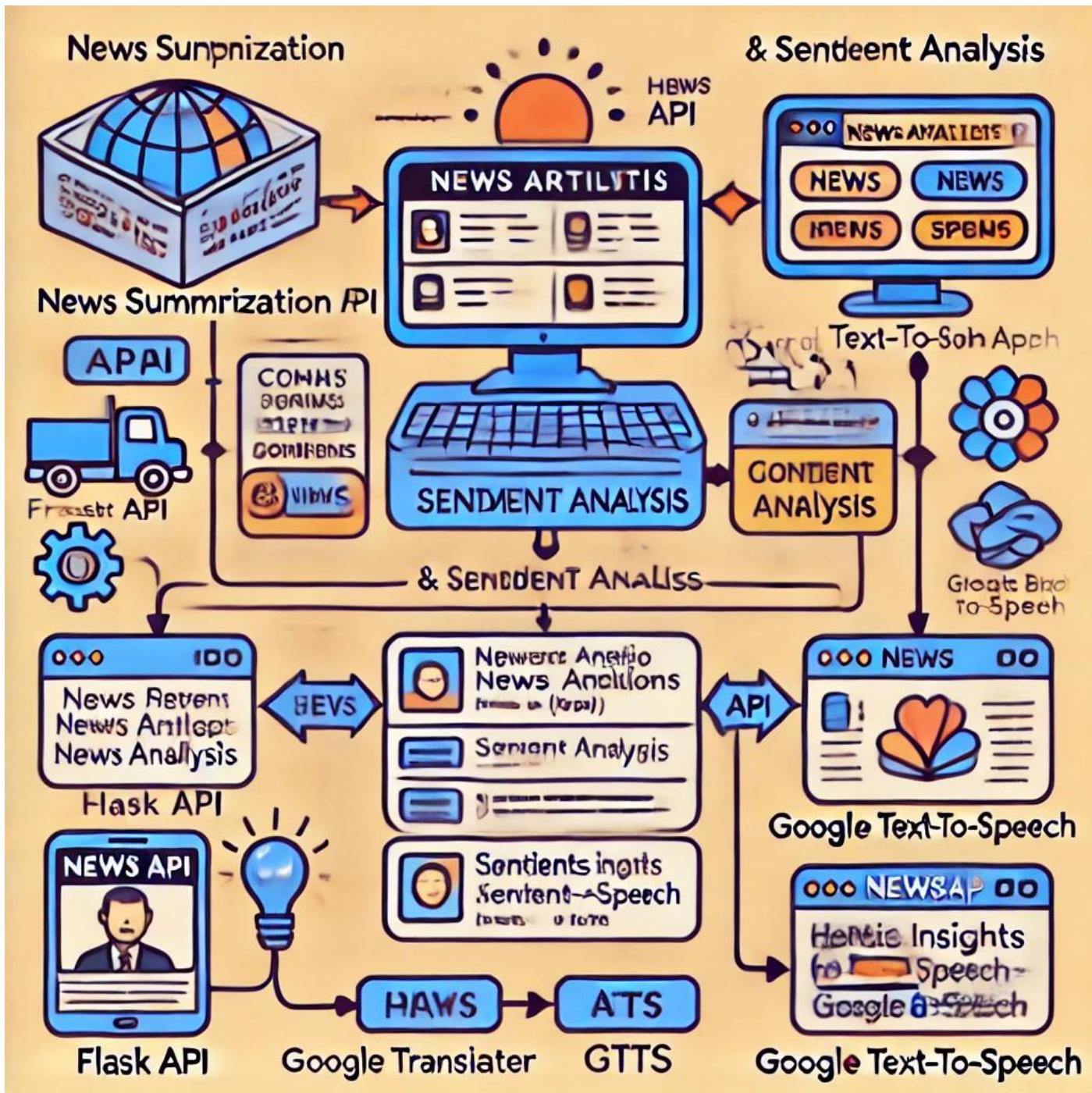
5 The insights are translated into Hindi using Google Translator API.

6 Hindi text is converted into speech using gTTS (Google Text-to-Speech).

7 Streamlit UI displays results and provides an option to listen to the Hindi audio summary.

◆ 5. System Design Diagram

This diagram visually represents how the frontend (Streamlit UI), backend (Flask API), and external APIs interact to fetch news, analyze sentiment, and generate Hindi text-to-speech.



📌 Step 4: Installation & Setup Guide

◆ 1. Prerequisites

Ensure you have the following installed:

- ✓ Python 3.10+
- ✓ pip (Python package manager)
- ✓ Git (for cloning the repository)

◆ 2. Clone the Repository

```
git clone https://github.com/your-username/your-repo-name.git
```

```
cd your-repo-name
```

◆ 3. Install Dependencies

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

This installs all necessary Python libraries, including **Flask**, **Streamlit**, **TextBlob**, and **gTTS**.

◆ 4. Set Up API Keys

- Get a **NewsAPI key** from [NewsAPI.org](https://newsapi.org)
- Replace "your_api_key_here" inside **utils.py**

```
API_KEY = "your_api_key_here"
```

◆ 5. Run the Application

1 Start the Flask API (Backend)

```
python api.py
```

2 Start the Streamlit UI (Frontend)

```
streamlit run app.py
```

 The app will open in your browser at <http://localhost:8501>



Step 5: Deployment Guide (Hugging Face & GitHub)

◆ 1. Deploying on GitHub

Steps to Upload the Project to GitHub:

1 Initialize a Git Repository (If Not Already Done)

```
git init
```

2 Add All Files

```
git add .
```

3 Commit Changes

```
git commit -m "Initial commit"
```

4 Create a New Repository on GitHub

Go to **GitHub** → Click **New Repository** → Name it → Click **Create**.

5 Push Code to GitHub

```
git branch -M main
```

```
git remote add origin https://github.com/your-username/your-repo-name.git
```

```
git push -u origin main
```

Now, your project is live on GitHub!

◆ **2. Deploying on Hugging Face Spaces**

Steps to Deploy on Hugging Face:

- 1 Go to Hugging Face → [Hugging Face Spaces](#)
- 2 Click "Create New Space"
- 3 Enter Space Name & Choose "Streamlit" as SDK
- 4 Clone Your Hugging Face Repository Locally

```
git clone https://huggingface.co/spaces/your-username/your-space-name
```

```
cd your-space-name
```

5 Copy Your Project Files into This Directory

```
cp -r ../your-local-project-folder/* .
```

6 Add requirements.txt (If Not Already Added)

```
streamlit
```

```
flask
```

```
flask-cors
```

```
requests
```

```
textblob
```

```
deep-translator
```

```
gtts
```

7 Commit & Push to Hugging Face

```
git add .
```

```
git commit -m "Deploying to Hugging Face"
```

```
git push origin main
```

8 Go to Hugging Face & Click "Restart Space"

Now, your app is live on Hugging Face!

 **Step 6: Screenshots**

Include UI screenshots showing:

- ✓ Company Name Input Field
- ✓ News Fetching & Sentiment Analysis Output
- ✓ Comparative Sentiment Report
- ✓ Hindi Audio Player

Spaces karthik808/News_Summarizer like 0 Running Logs

News Summarization and Sentiment Analysis

Enter a company name to get the latest news and sentiment analysis.

Enter a company name:

News Articles about tesla:

1. Is Tesla cooked?

Sentiment: Negative

Tesla stock plunged 15 percent on Monday, its steepest drop in five years. The price is down over 50 percent since its December highs. Tesla owners, disgusted with Elon Musk's slash-and-burn tactics for the Trump administration, are selling their vehicles a...

[Read Full Article](#)

2. How the Tesla brand turned so toxic

Sentiment: Neutral

At a surprise all-hands-on meeting Thursday night in Texas, Tesla CEO Elon Musk rallied up his company's employees as the brand's public reputation continues to plummet, Business Insider reports. At the livestreamed meeting, Musk told employees to "hang on to..."

[Read Full Article](#)

Comparative Sentiment Analysis for tesla

Sentiment Distribution: {'Positive': 5, 'Negative': 2, 'Neutral': 3}

Dominant Sentiment: Positive

Key Insights: ↗

- tesla's news coverage is mostly positive, reflecting strong market confidence.
- Example contrast: 'How the Tesla brand turned so toxic' focuses on company success, whereas 'Is Tesla cooked?' highlights a challenge.

[Play audio for Insights](#)

▶ 0:00 / 0:23

Hindi audio generated successfully!

📌 Step 8: Challenges Faced & Future Improvements

◆ 1. Challenges Faced

API Limitations

- NewsAPI has request limits**, which sometimes restricts the number of news articles fetched.
- Solution:** Users need to provide their own API key to avoid exceeding request limits.

Sentiment Analysis Accuracy

- TextBlob sentiment analysis** may not always be 100% accurate.

- **Solution:** Future versions could integrate **more advanced NLP models** like **VADER** or **BERT** for better accuracy.

Hindi TTS Quality

- **Google Translator API sometimes misinterprets complex insights.**
- **Solution:** Implement **manual preprocessing of insights** before translation to improve accuracy.

Deployment Issues on Hugging Face

- Initial deployment faced **missing dependency issues** (e.g., deep-translator not found).
- **Solution:** Fixed by adding all dependencies to requirements.txt

◆ 2. Future Improvements

- ✓ **Advanced NLP Models** → Use **BERT** or **OpenAI GPT** for better sentiment analysis.
- ✓ **More Languages** → Add support for **regional languages** apart from Hindi.
- ✓ **Data Visualization** → Use **interactive charts** to display sentiment trends.
- ✓ **Historical Analysis** → Compare sentiment trends over time for a company.
- ✓ **Mobile App Integration** → Extend the app for **Android/iOS** using Flutter or React Native.

📌 Step 9: Conclusion & References

This final section summarizes the project and provides references for further reading.

◆ 1. Conclusion

The **News Summarization & Sentiment Analysis App** successfully automates the process of **fetching, analyzing, and summarizing news** related to any company. The app provides:

- ✓ **Real-time sentiment analysis** of news articles.
- ✓ **Comparative insights** to understand public perception.
- ✓ **Hindi text-to-speech (TTS) output** for accessibility.
- ✓ **A user-friendly interface** built with Streamlit.

This project demonstrates how **Natural Language Processing (NLP)**, and **APIs** can be combined to deliver **data-driven insights** efficiently. Future improvements, such as **advanced sentiment models** and **multilingual support**, can further enhance the application's effectiveness.

◆ 2. References

- **NewsAPI** (Used for fetching real-time news) → <https://newsapi.org/>
- **TextBlob** (Sentiment Analysis) → <https://textblob.readthedocs.io/>
- **Google Translator API** (Translation to Hindi) → <https://pypi.org/project/deep-translator/>
- **gTTS (Google Text-to-Speech)** → <https://pypi.org/project/gTTS/>
- **Streamlit** (Frontend UI) → <https://streamlit.io/>
- **Flask** (Backend API) → <https://flask.palletsprojects.com/>

Contact:

- Email: karthikbrinfopro@gmail.com
- GitHub: Robbin808