Course End Project

NewsGenie – An AI-Powered Information and News Assistant

Overview

In this project, you will build NewsGenie, an AI-powered information and news assistant designed to help users navigate today's fast-paced digital landscape. You will develop a system that efficiently filters misinformation, curates reliable and up-to-date news, and provides quick answers to general queries—all within a single, unified platform.

Instructions

Submission: Submit a detailed report or presentation via the LMS, including:

1. **AI chatbot design**: Guidelines for conversation management and query differentiation
2. **Real-time news integration**: Sample outputs showing news updates for technology, finance, and sports categories
3. **Workflow and error handling**: A proposed process detailing API integration, fallback mechanisms, and overall query processing management

Situation

People struggle to keep up with real-time, reliable news in today's fast-paced world. Users face challenges such as filtering misinformation, finding trustworthy sources, and securing timely answers to general queries—all in one place.

Modern news consumption is fragmented and often overwhelming, leaving users flooded with too much information, which makes it difficult to:

- Stay updated with the most relevant and accurate news
- Filter out unreliable or misleading content
- Access personalized news feeds alongside general information quickly

This environment demands a unified solution that can intelligently manage both conversational queries and real-time news updates.

NewsGenie is designed to address these issues by serving as an intelligent assistant that provides both instant query responses and curated up-to-date news.

Task

The core task is to build NewsGenie as a unified platform that:

1. **Handles conversations:** Develop an AI chatbot that can interpret and answer general queries while distinguishing them from news requests
2. **Integrates APIs:** Combine a real-time news API with a web search tool to fetch topic-specific news and additional external resources
3. **Manages workflow:** Utilize a LangGraph-based workflow to process user requests efficiently and maintain conversation context
4. **Delivers an intuitive UI:** Provide a robust, interactive interface via Streamlit that allows users to select news categories and input queries effortlessly

Actions

To achieve these objectives, the following actions will be implemented:

1. **Chatbot development**:
   - Build and train an AI chatbot using natural language processing techniques to manage and distinguish between different query types
2. **API and web search integration**:
   - Integrate a real-time news API to retrieve the latest news based on user-selected categories
   - Implement a web search tool to dynamically fetch external information that complements the chatbot's responses
3. **Workflow optimization**:
   - Employ a LangGraph-based workflow to streamline query processing, ensuring efficient handling of both news and general queries
   - Develop fallback mechanisms to manage API failures or instances when no relevant news is found
4. **User interface deployment**:
   - Design and deploy a Streamlit-based frontend that is user-friendly, supports session management, and optimizes response times
   - Ensure the interface allows users to easily choose news categories and interact with the assistant
5. **Error handling and performance optimization**:
   - Incorporate strategies to manage missing API keys, failed API calls, and complex query scenarios, ensuring a consistent user experience

Result

The final submission will include:

1. An interactive AI-powered assistant that delivers instant responses to general queries while providing real-time, curated news updates
2. A fully integrated system showcasing the use of a real-time news API, a dynamic web search tool, and a LangGraph-based workflow for efficient query processing
3. A demonstration of the user-friendly interface built with Streamlit, highlighting session management, error handling, and responsive design
4. A detailed explanation of fallback mechanisms and optimization strategies ensuring reliable performance even during API failures

This project will demonstrate your ability to integrate multiple AI components into a cohesive platform that simplifies information access and enhances the overall user experience in a fast-paced digital environment.