edureka! a Veranda ENTERPRISE

Generative AI and ML

Capstone Project

Overview

The goal of this capstone project is to develop a web application using Streamlit that enables users to translate text into various languages, convert the translated text into speech, and download the resulting audio file. The project leverages **Google's Gemini API** for translation and **Google Text-to-Speech (gTTS)** for speech synthesis.

Project Description

This project aims to develop a system that translates text into various languages and converts the translated text into speech. The system is designed to be user-friendly and interactive, allowing users to either enter text directly or upload a file in various formats (PDF, TXT, Excel, CSV). The text is then translated into a language selected by the user from a dropdown menu, utilizing the Gemini API for accurate and context-aware translations.

Tasks for Learners

1. **Research Streamlit and Basics:** Learn the fundamentals of Streamlit by referring to the official documentation and introductory tutorials.
2. **Understand Google Gemini API for Translation:** Research Google's Gemini API for language translation. Understand how to authenticate using an API key, craft translation prompts, and handle API responses effectively.
3. **Explore gTTS for Text-to-Speech:** Investigate the gTTS library for text-to-speech conversion. Learn about available language codes, voice options, and output formats.
4. **Design User Interface:** Plan and design the user interface, considering input text areas, file upload options, dropdown menus for language selection, and buttons for translation and audio download.
5. **Implement Language Translation:** Write code to take user input or file content, select a target language, and use the **Gemini API** for translation.
6. **Add Text-to-Speech:** Implement text-to-speech functionality using gTTS and save the generated audio file in MP3 format for playback and download.
7. **Incorporate File Upload:** Allow users to upload text, PDF, CSV, or Excel files for translation. Implement file reading and text extraction logic.
8. **Enhance User Experience:** Implement error handling for invalid inputs or API issues, provide user-friendly messages, and add clear instructions for interacting with the app.

9. **Test and Debug:** Thoroughly test the application across different languages and file types, identifying and fixing any bugs or translation inaccuracies.
10. **Deploy Application:** Explore deployment options such as Heroku or Streamlit Sharing and deploy the fully functional application for user access.

Submission Guidelines:

- Submit the completed code and a documentation file.
- Documentation should explain how to set up and use the app with the Gemini API key, along with considerations, limitations, and challenges faced during development.