PESU I/O Slot 20: Final Presentations

Travel Planer and Itinerary Builder

Roshan, PES1UG24CS387

Ayaan Ahmed, PES1UG24CS102

Tarun gowda p, PES1UG24AM303

Vishal K Konnur, PES1UG25CS596

Idea

A Flask-powered REST API that intelligently plans trips, recommends destinations using RAG, selects optimal travel styles with AI agents, and generates daily itineraries with budget estimates in Indian Rupees.

Core Functionality

This project is a Flask-powered REST API that intelligently recommends destinations using Retrieval-Augmented (RAG), selects optimal travel styles with AI agents, and itineraries with realistic budget estimates in Indian Rupees produces personalized, dynamic trips that combine RAG-local experiences, AI-determined travel modes, and mid-calculations for India.

Implementation

Flask Framework

RESTful API built with Flask and Flask-CORS for cross-origin requests, enabling seamless integration with frontend applications and testing tools.

RAG Pipeline

Retrieval-Augmented Generation using Wikipedia APIs to search, fetch, and summarize top attractions with in-memory caching for performance.

Hugging Face Models

BART-large-CNN for text summarization and BART-large-MNLI for zero-shot classification to intelligently determine travel preferences and modes.

Itinerary Engine

Advanced activity pooling with interest and mode weighting ensures unique, non-repetitive activities across all travel days.

Core Functionality

Personalized Itinerary Creation: The system analyzes user preferences to create personalized trip experiences.

Intelligent Mode Selection: It intelligently selects optimal travel modes ranging from budget-friendly to luxury options.

Destination & Attraction Discovery (RAG): It discovers attractions by analyzing user preferences and retrieving information through **Wikipedia's knowledge base**.

Varied Daily Plans: The system generates diverse daily itineraries that avoid repetition while maintaining thematic consistency.

Realistic Budget Estimation: Every recommendation is grounded in real data, and all budget estimates reflect **realistic mid-range travel costs specific to India** (in Indian Rupees).

API Endpoints Overview

POST /plan_trip
Creates a complete itinerary by running the Agent for mode selection, RAG for attractions, and generating varied daily plans with unique activities.

GET /get_itinerary/:user_id
Retrieves the most recently generated itinerary for a specific user, including all days and activities.

POST /update_day_plan
Replaces all activities for a specific day, allowing users to completely customize their schedule.

POST /add_activity
Appends a custom activity to any day in the itinerary without affecting existing plans.

DELETE /remove_activity/:id
Removes a specific activity by ID from any day, with proper cleanup from the activities lookup table.

GET /recommended_places
Returns RAG-powered destination suggestions based on location and interests, perfect for discovery.

POST /share_itinerary
Generates a human-readable summary in markdown-like format, returned as a JSON string for easy rendering.

POST /budget_estimate
Calculates realistic travel costs in Indian Rupees based on Agent-selected mode or user-provided preferences.

Key Takeaways & Learnings

Integrated AI System: Successfully developed a cohesive system by integrating multiple advanced AI techniques, including Retrieval-Augmented Generation (RAG) for discovery and Zero-Shot Classification for travel mode selection.

Localized Budgeting Logic: Created a critical feature by implementing realistic budget estimation using India-specific costs and returning values in Indian Rupees. This shows an understanding of real-world localization needs.

Content Variety & Uniqueness: Engineered an Itinerary Engine with activity pooling and set-based tracking to guarantee that activities are unique and non-repetitive across the entire trip.

Performance Optimization: Implemented TTL-based caching within the RAG pipeline to store search results and page summaries, drastically improving system speed for repeated destination queries

Challenges Faced

- **Ensuring Non-Repetitive Itineraries:** Developing the **Itinerary Engine** to guarantee activity diversity and prevent repetition across multiple days while maintaining thematic consistency.

- **Model Confidence Balancing:** Implementing a **confidence threshold** for mode selection and developing a robust fallback heuristic to handle ambiguous user input[6].

Going Ahead

Planned next steps include migrating to a persistent database (SQLite/PostgreSQL), expanding to realistic budgets beyond India with multi-currency support, building a front-end UI to showcase the CORS-enabled REST API, and enhancing activity pools and weighting logic to increase variety and personalization based on evolving travel trends.

References

https://en.wikipedia.org/w/api.php
https://huggingface.co/docs/transformers/en/model_doc/bart
https://apitester-ptuy.vercel.app/

Thank You!