fit@hcmus

Group 7 - Sigma Alpha

Computational Thinking

SMART TOURISM:

Viet Nam Sightseeing Destinations Recommender System

exSighting - Vietnam Sightseeing

Instructor:

PhD Lê Trung Hoàng

Mr Trần Hoàng Quân

Mr Đỗ Đức Hào

Members:

Nguyễn Tuấn An (24127011)

Nguyễn Hải Nam (24127088)

Phạm Đinh Gia Khánh (24127420)

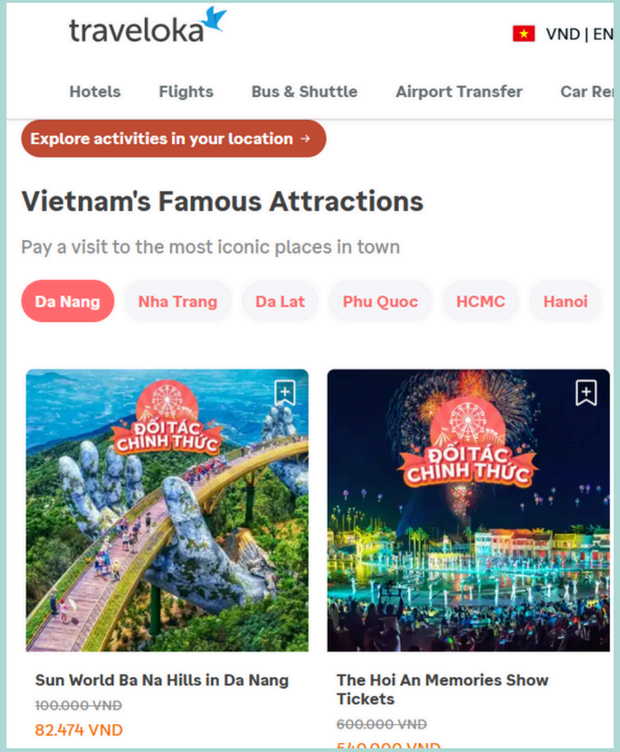
Võ An Phú (24127494)

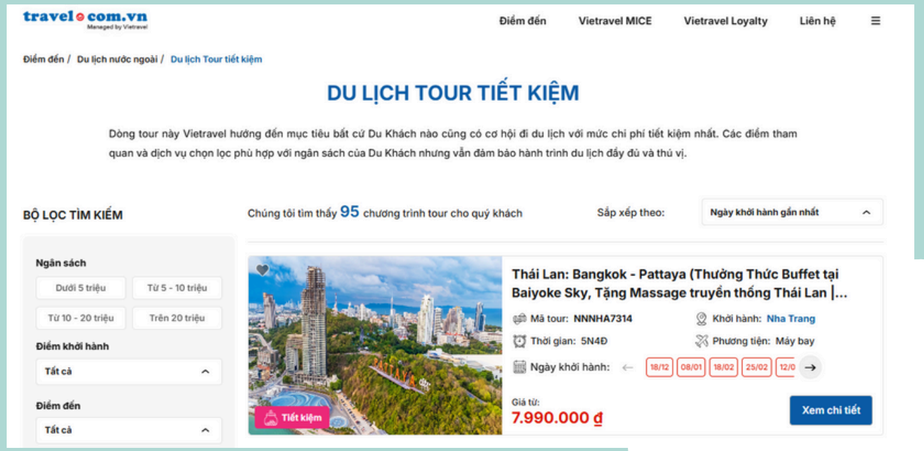
Contents

Problem Analysis

Motivation

- **Oversaturated yet Generic:** Numerous platforms offer convenient tour but lack deep personalization.
- **The "Cookie-Cutter" Experience:** Users often face repetitive suggestions, ignoring individual preferences.
- **Neglected Destinations:** Lesser-known "hidden gems" remain overshadowed by mass tourism sites.





Motivation

Demand for Personalization:

- 71% of consumers expect personalized interactions [McKinsey & Company].

The Rise of "Hidden Gems":

- 69% of travelers actively seek "off-the-beaten-path" destinations to avoid overcrowding [Booking.com Travel Predictions].
- 89% want to discover authentic local culture [Amex Global Trends].

Motivation

- **Our Vision:** Personalized Discovery & Cultural Promotion
- **The Solution:** A website that delivers optimal, customized experiences aligned with specific user desires.
- **The Strategy:** Leveraging AI to connect travelers with overlooked, culturally rich locations.
- **The Impact:** Promoting the sustainable development of Vietnam's culture and tourism ecosystem.
- **Scope:** Vietnam
- **Input:** User interactions (like/dislike, view time, comment)
- **Output:** Destinations matching user's preferences

Decomposition

Decomposition

- The Project is broken into small tasks:
  - **Determine input and output scope:** clear input and output help simplifying and determining the main target, as well as facillitate developing process.
  - **Forming Database:** this is the “root” of the project, because places’ information are all getting from the database, that’s why Database is a crucial and priority section.
  - **Handling cold start recommender:** solve the initial lack of data to ensure a smooth experience for new users.
  - **Handling main recommender system:** Once user data is available, generate recommendations tailored to personal preferences.

Decomposition

- The Project is broken into small tasks:
  - **UI / UX design:** build up from simple appearance while still have base function, then upgrade the UI during the process.
  - **User’s Managing:** administrator should be able to manage customers’ account.

Pattern recognition

Pattern recognition

- These patterns have to be clearly determined for better recommendation:
  - **Pattern of location:** include places' specific characteristics (sea, mountain, river,...)
    - **Goal**: Optimize recommendations to match specific user preferences (tags).
  - **Location Interaction Patterns:** Track metrics like Views, Likes, Dislikes, Comments.
    - **Goal**: Solve the "Cold Start" problem by suggesting popular locations based on high visibility/engagement.
  - **User Behavior Patterns:** Identify preferences through interactions (Likes, Dislikes, Time on Page).
    - **Goal**: Detect positive signals to recommend similar locations ("Content-based Filtering").

Abstraction

Abstraction

- Those mentioned patterns can clearly define project flows and eliminate unnecessary information from the database:
  - **Defining Inputs and Outputs:**
    - **Input:** User interaction behaviors (derived from behavior patterns).
    - **Output:** Accurate recommendation results.
    - **Benefit:** Provides a clear specification for implementation.
  - **Eliminate irrelevant details:** Height, weight, Entrance gate colors, specific nearby food vendors.
  - **Frontend:** Provides the main interface for user interaction.
  - **Backend:** Receives input data from the Frontend, Processes data using the Database and Recommendation Algorithms to return optimized suggestions to the user.

Algorithmic design

Algorithmic design

1. Logical Architecture

A Hybrid System that processes both explicit and implicit data.

- The system integrates User Behavioral Analytics (what they do) with Content Semantics (what the place is).
- Data Flow:
  - **Input**: User Interactions (View Time, Likes, ...) + Natural Language Search.
  - **Processing**:
    - Module A: Dynamic Rating Scorer (converts behavior to scores).
    - Module B: Intent Detection Engine (LLM-based).
    - Module C: Similarity Engine (TF-IDF + Cosine Similarity).
  - **Output**: A personalized, ranked list of sightseeing locations.

Algorithmic design

2. The AI Components & Efficiency

Module A.

- **Implicit Rating Algorithm (The "Smart" Data Collection)**
  - Instead of relying solely on manual star ratings, calculate interest based on View Time.
    - **Noise Filtering**: Interactions under 5 seconds are discarded as accidental clicks.
    - **Linear Interpolation**: View time (5s - 90s) is mapped linearly to a score of 2.5 - 4.0. (score = 2.5 + (time - 5) / (90 - 5) * (4.0 - 2.5))
    - **Efficiency**: The system only updates the database if the new calculated score is higher than the existing one, minimizing database write operations.

Algorithmic design

2. The AI Components & Efficiency

Module A.

- **Explicit Rating Algorithm**
  - **Like**: Adds +4.0 to the score.
  - **Dislike**: Subtracts -5.0 (or sets the score to the minimum of 1.0).

Algorithmic design

2. The AI Components & Efficiency

Module B. Intent Detection Engine

- **Input**: Receives raw text (e.g., "I want a cheap beach trip in Da Nang").
- **Processing**: Sends a prompt to the Groq LLM to extract structured JSON.
- **Output**: Returns standardized tags:
    - Location: "Da Nang"
    - Themes: "Beach"
    - Weather: “Cool”

Algorithmic design

2. The AI Components & Efficiency

Module C. Similarity Engine

- **Vectorization (TF-IDF)**: Converts place names, descriptions, and tags into mathematical vectors.
- **Content-Based Scoring:** Uses Cosine Similarity to measure the angle between the "User Requirement Vector" (derived from Module B + History from Module A) and "Place Vectors".
    - Smaller Angle → Higher Similarity → User is MORE likely to like it.
    - Bigger Angle → Lower Similarity → User is LESS likely to like it.

Algorithmic design

2. The AI Components & Efficiency

Module C. Similarity Engine

- **Hybrid Strategy**: Merges the Content Score with:
  - Collaborative Filtering (CF): "Users like you also liked this" (Item-Item Similarity).
  - Popularity: "This place is trending" (Based on total likes/ratings).
- **Final Formula**: Score = 0.4*Content + 0.4*CF + 0.2*Popularity (for users with history).

Algorithmic design

3. **Post-Processing**

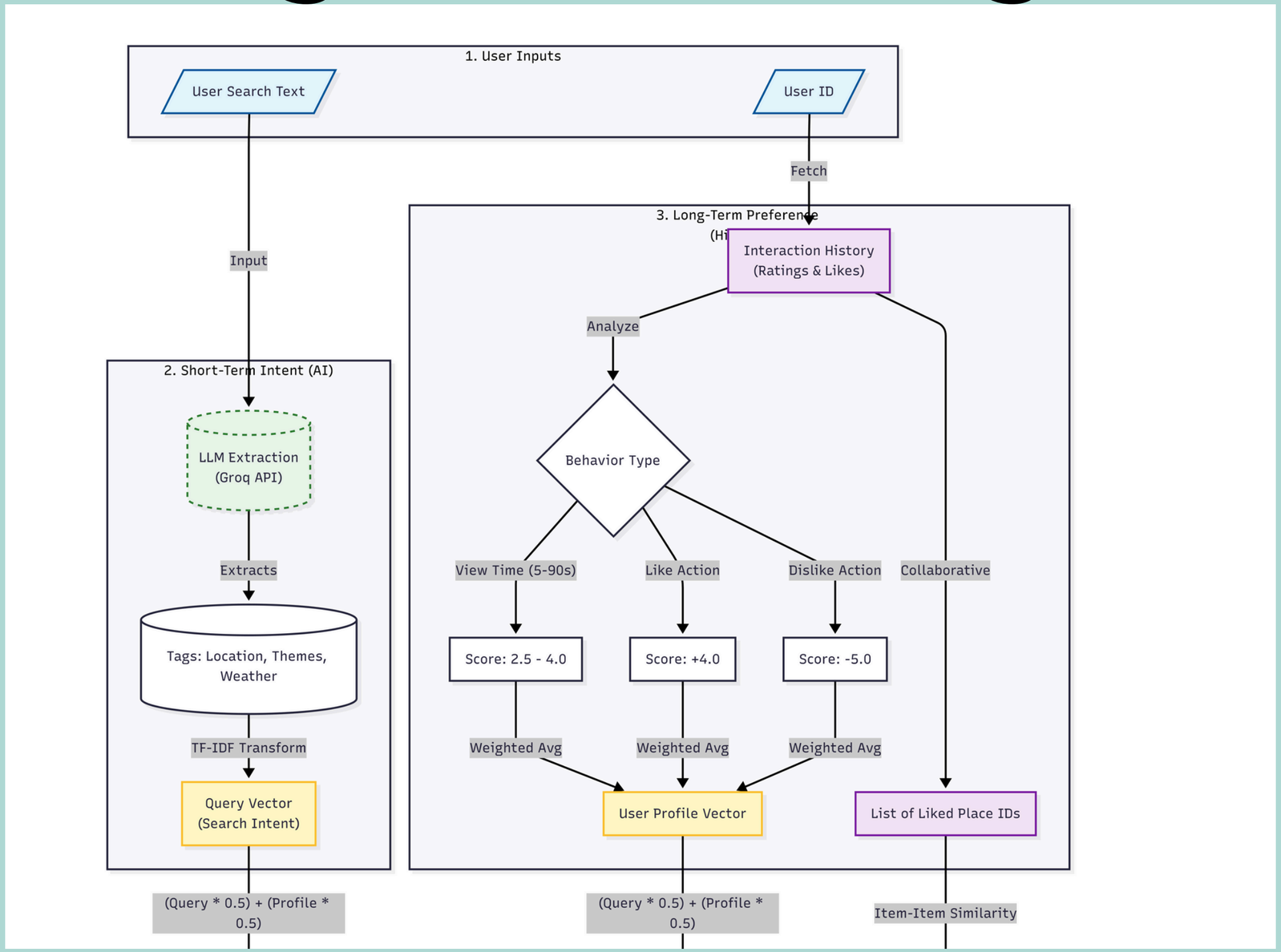
- **Soft Penalty:** If a user previously Disliked a place, we multiply its score by 0.1. It pushes it to the bottom but doesn't strictly delete it (in case they change their mind).
- **Diversity:** The algorithm ensures the top 10 results aren't all identical (e.g., 10 museums). It forces a mix of tags

Algorithmic design

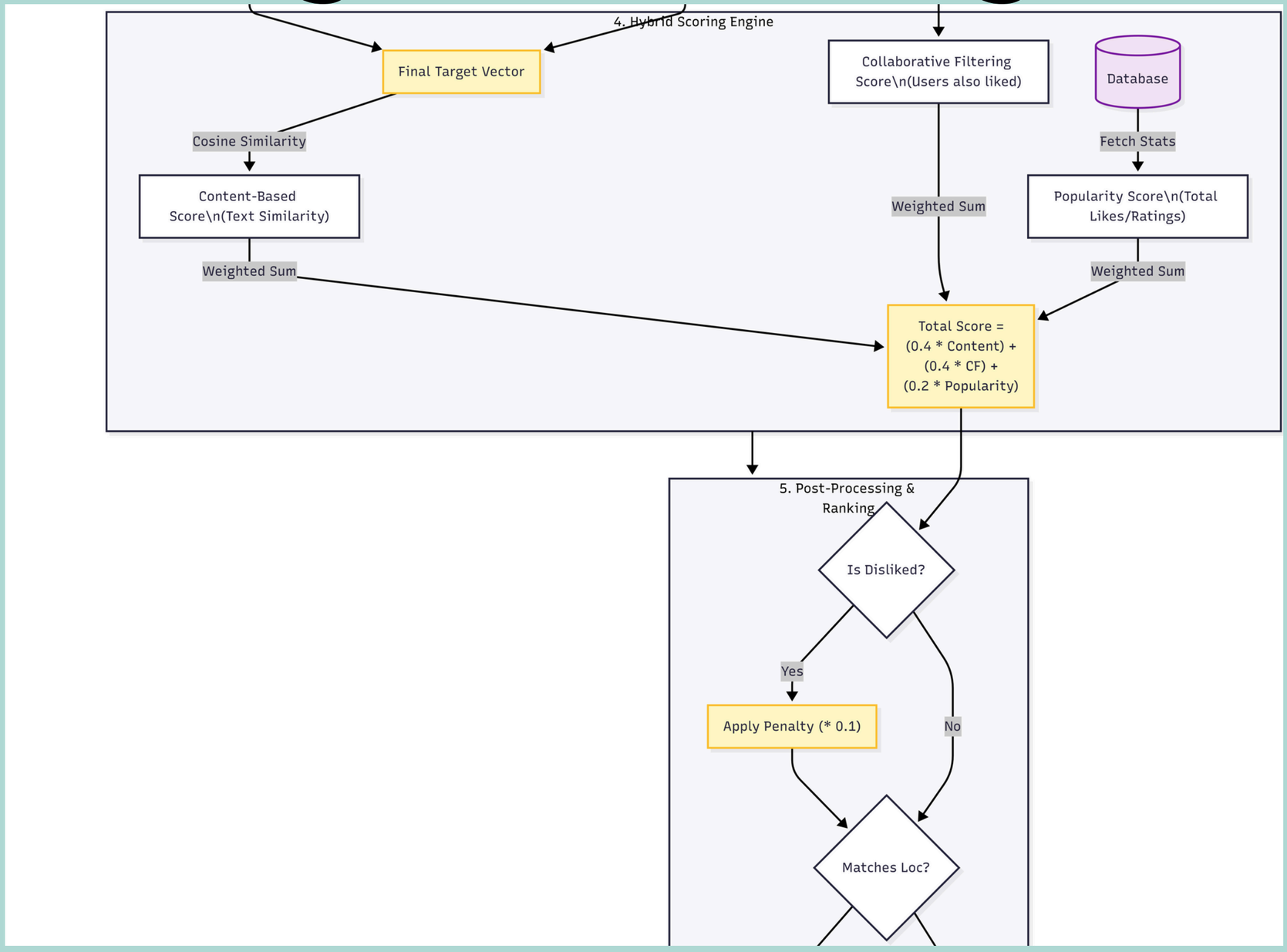
3. **Post-Processing**

- Diverse processing:
  - **Place pool**: top_k * 3 (highest scoring places)
  - **From pool add if satisfied 2 limits**:
    - Maximum ⅓ results from the same province
    - Maximum ½ results from the same category
  - **After reaching 70% of pool:** add if satisfied either limit
  - **Fallback**: If still not enough items, fill remaining slots by score

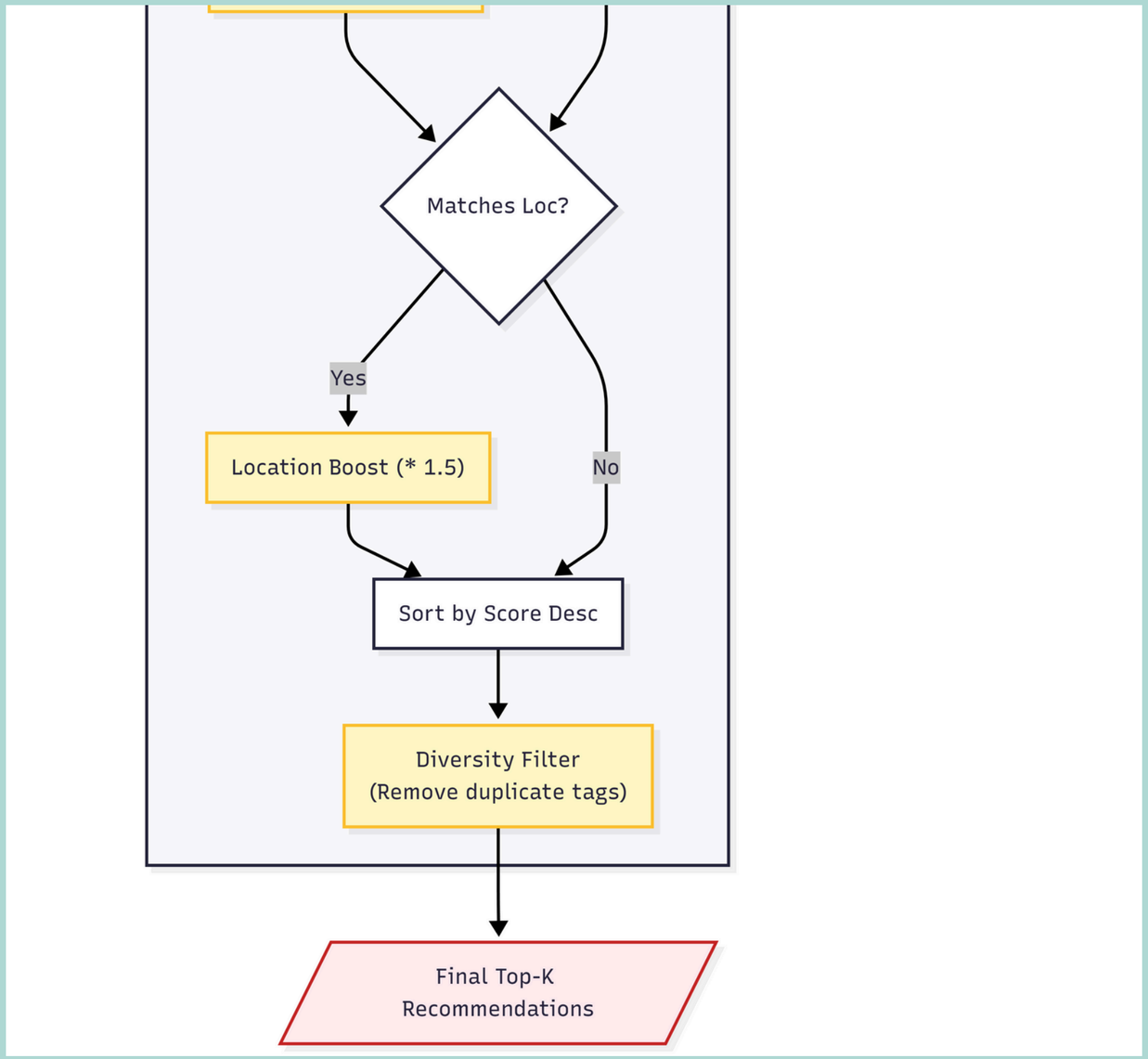
Algorithmic design



Algorithmic design



Algorithmic design



Simulation (Demo)

Demo

- Link Demo: https://drive.google.com/drive/folders/1I6IDUyKd7wKmMFcvwOI2BZNCAhas4Mlr?usp=drive_link

Testing & Improvement

Algorithm Evaluation

1. Experimental Setup

- Total Users: 102 with various behaviors
- Methodology: Train/Test Split (80/20)

2. Overall Performance

Metric	@5	@10	@20
Precision	23.73%	17.84%	12.06%
Recall	23.94%	35.33%	47.11%
F1 Score	23.20%	23.21%	18.93%
NDCG	26.10%	30.43%	35.65%

- MAP: 19.43%
- Coverage: 65.95%
- Diversity: 30.00%

Algorithm Evaluation

3. Analysis: STRENGTHS

- **Model Validity**:
  - The model has successfully captured underlying patterns from the data.
  - Recommendations are non-random and personalized.
- **Relatively Good Recall**: 35.33% @ Top-10, 47.11% @ Top-20
  - The model successfully retrieves nearly 50% of relevant items within the top-20 results.
  - Cold-start potential: Capable of generating suggestions for new users.
- **High Coverage:** 65.95%
  - The system recommends a wide variety of items across the catalog.
  - Avoids popularity bias (does not get stuck recommending only famous locations).
- **Decent NDCG**: 26-36%
  - The ranking logic is reasonable.
  - Relevant items tend to be prioritized and appear in higher positions.

Algorithm Evaluation

3. Analysis: LIMITATIONS

- **Low Precision:** 23.73% @ Top-5
  - On average, only ~1.2 items out of the top 5 are relevant.
  - User Experience (UX) impact: Users encounter "noise" and must scroll past irrelevant items.
- **Low Diversity:** 30%
  - Recommendations lack variety.
  - Risk: The system may suggest items that are too similar to each other (e.g., suggesting 5 beaches in a row).
- **Low MAP** (Mean Average Precision): 19.43%
  - Users are forced to scroll down to find the best match.

Algorithm Evaluation

3. Analysis: Causes

- **Data Challenges:** High Tag Overlap
  - Generic tags like "sightseeing," "historical," and "nature" appear across almost all locations.
  - Consequence: The model lacks the granularity to distinguish between unique destination types.
- **Algorithmic Limitations:** Content-Based Constraints:
  - Relies strictly on static attributes (tags).
  - No Collaborative Signals: Ignores peer influence (what similar users liked).
  - Explicit only: Fails to capture implicit feedback or subtle behavioral patterns.

Algorithm Evaluation

4. **Proposed improvements**
   - **Enrich Feature Granularity:** Introduce more specific and distinctive tags to reduce overlap.
   - **Integrate Collaborative Filtering:** Incorporate user interaction patterns to enhance recommendation accuracy.
   - **Enhanced preference capture:** allow users to specify price ranges or spending limits


ARRIVALS
Thank you!
AIRPORT