ISSN 2007-9737

gTravel: Weather-Aware POI Recommendation Engine for a Group of Tourists

Rajani Trivedi[1], Bibudhendu Pati[1], Subhendu Kumar Rath[2]

[1] Rama Devi Women's University, Bhubaneswar, India

[2] Biju Patnaik University of Technology, Rourkela, Odisha, India

{rtrivedi2576, patibibudhendu, rath.subhendu}@gmail.com

Abstract. Weather is a big factor in tourist decisions, and certain places or events aren't even recommended during dangerously bad weather. It is difficult to provide a better recommendation to a group of tourists in these circumstances. We propose gTravel, a weather assistant framework that predicts weather in specified points of interest for a group of tourists. We demonstrate how prior knowledge of climatic patterns at a POI, as well as prior insights into how visitors rank their destinations in a variety of weather conditions, can help improve choice reliability. According to our findings, the recommendations are significantly more valid, and the recommended remedy is more comfortable.

Keywords. POI, tourist, weather, recommendation, interest.

1 Introduction

Location recommender frameworks can make predictions and suggests items to locations dependent on data accumulated from different sources [11]. They gather data about the different users and different locations [10], and the connections between them [7]. At that point the frameworks dissect the examples and inclinations of the users towards locations and make suggestions likewise [6].

Various methodologies are used to built a recommendation system, generally six type of approaches are used: Collaborative filter-based approach: in CF approach, the similarity between users or similarity between locations are measured for making the recommendation system [5]. Probabilistic approach, the mobility pattern of users and checks-in pattern of POIs is considered, based on the patterns probabilistic assumption are made for the recommendation. Tensor-based approach, in the approach multidimensional matrix are built from the user features, POIs features data, or different features. From further formulating and optimizing the latent features of user, location and other data a score is obtained for POIs and recommended to user [1]. Graph-based approach, in this model the location visiting pattern of user are represented on graphs, and greedy algorithms are applied. HITS-based approach, in these techniques the hub and authority values of user and POIs are utilized to make recommendation. Integrated framework-based approaches, the techniques which deploy more than one approach.

These procedures deploy various information for generating the recommendation system and the types of information is divided in five types temporal influences, spatial influenced, categorical information, contextual influence, social influence or multi-influential, i.e., more than two influenced are combined [12].

2 Related Work

Linus et al. [4] insisted on the involvement of the simple travel recommendation model for the forthcoming changes utilizing wondary as a foundation, visitors can save, maintain and share their travel details. The suggested framework specifics were merged and presented in a public way, to enhance user experience with TTRS and get around restrictions on mobile devices like small screens, but this approach is rarely used. When recommending hotels to tourists, Garcia-Crespo et al. [4] talked about Sem-Fit, a semantic TRS.

By finding accommodations, their location, and other amenities according to their preferences, it helps tourists to avoid spending as much money. In [1], Zeng et al., collaborative filtering-based approach is used for constructing the recommendation system. It leverages temporal information and geographical information. A day is divided into 24 equal slots, to capture the frequencies of check-in in each time-slot. These frequencies of check-in of these time slot is used to create the location feature vector.

To normalize the check-in time of a location, the checks-in at a time-slot is divided by the total number of check-ins in 24 hours. Considering the check-in counts in the time slots as allocation feature vector, these vectors are used in the cosine similarity function for calculating the similarity between two location. To get the weight of a location which reflects the user preference, the counts of check-in at a location is divided by the total number of check-ins at all locations visited by the user.

3 Background and Problem Definition

Let $\mathcal{I} = \{i_1, i_2, i_3, ...i_n\}$ be the itineraries in a specific town. Each itinerary $i_\vartheta \in \mathcal{I}$ $(1 \leq \vartheta \leq n)$ consists of a set of POIs $\mathbb{P} = \{\mathbb{P}_1, \mathbb{P}_2, \mathbb{P}_3, ...\mathbb{P}_k\}$.

Each POI is associated with one or more categories $\mathcal{C}$ such as entertainment, shopping, dining, etc. In this work each recommended itinerary consists of a number of POIs based on the travel preferences of the tourist, the popularity of the locations and the travel expenses.

We thus have $i_\vartheta = \{\mathbb{P}_1, \mathbb{P}_2, \mathbb{P}_3, ...\mathbb{P}_k\}$, where k is the size of the itinerary i_ϑ. The total distance covered by the tourist is calculated by adding up the distance from $\mathbb{P}_\gamma$ and $\mathbb{P}_{\gamma+1}$, where $1 \leq \gamma < k$.

The distance between two itineraries i_1 and i_2 is obtained by calculating the distance between the last POI visited in itinerary i_1 and the first POI visited in itinerary i_2. A travel speed of 4 kilometers per hour [9] was assumed.

3.1 Characteristics of Local and Global Users

The Local User (LU) is a user who lives in some city A, where he/she has visited some number of POIs. Suppose the LU from city A wants to visit a new city B. Then the LU of city B is referred to as the Global User (GU) with respect to the LU of city A. [7] contains a detailed discussion about local and global users.

3.2 Average POI Visit Duration

Considering the travel behavior of some user u, the average visit duration for a POI can be calculated using Eqn.1:

$$\mathcal{D}(\mathbb{P}) = \frac{\sum\limits_{u=1}^{k} \sum\limits_{j=1}^{\ell} (t_j^{\text{dept}} - t_j^{\text{ari}}) \delta(\mathbb{P}_j = \mathbb{P})}{\sum_{u=1}^{\kappa'} \mathcal{V}_u \, \delta(\mathbb{P}_j = \mathbb{P})} \forall \mathbb{P} \in \mathcal{P}, \quad (1)$$

where, $j = \{1, 2, ..., \ell\}$, $u = \{1, 2, ..., k\}$ and $\mathcal{V}$ provides the number of trips a tourist makes to a certain POI. $\delta(\mathbb{P}_j = \mathbb{P}) = \{^{1 \text{ if } (\mathbb{P}_j = \mathbb{P})}_{0 \text{ otherwise}}$. $\mathcal{D}(\mathbb{P})$ is the average visit duration for a specific POI. The mean visiting time for a certain POI $\mathbb{P}$ is referred to as $\mathbb{P}$ [2, 3].

3.3 Interest in LUs and GUs Depending on Time

Suppose $\mathcal{C}_\mathbb{P}$ is the category of some POI $\mathbb{P}$. The interest of a particular tourist for some category $\mathbb{C}$ is given by Eqn. 2:

$$\text{Intr}_{u_i} \mathbb{C} = \sum_{j=1}^{\ell} \frac{(t_{\mathbb{P}_j}^{\text{dept}} - t_{\mathbb{P}_j}^{\text{ari}})}{\mathcal{D}(\mathbb{P}_j)} \delta(\mathcal{C}_{\mathbb{P}_j} = \mathbb{C}) \, \forall \mathbb{C} \in \mathcal{C}, \quad (2)$$

where, $\delta(\mathcal{C}_{\mathbb{P}_j}) = \{^{1 \text{ if } \mathcal{C}_{\mathbb{P}_j} = \mathbb{C}}_{0, \text{ otherwise}}$. Eqn. 2 will later be used to measure tourists' interest for some POI category $\mathbb{C}$ with respect to the visiting times of all tourists for that POI category.

It is obvious that a visitor spends more time at a certain POI if he/she is highly interested in that POI.

3.4 History of Travel

Let $\mathcal{U}$ be a set of tourist. For some tourist $u \in \mathcal{U}$, we define a sequence of itineraries $S_u = ((i_1, t^{\text{ari}}_{i_1}, t^{\text{dept}}_{i_1}), ..., (i_n, t^{\text{ari}}_{i_n}, t^{\text{dept}}_{i_n}))$ where n is the number of itineraries, in a triplet $(i_\vartheta, t^{\text{ari}}_{i_\vartheta}, t^{\text{dept}}_{i_\vartheta})$, i_ϑ is an itinerary, $t^{\text{ari}}_{i_\vartheta}$ is the time of entry and $t^{\text{dept}}_{i_\vartheta}$ is the time of departure. The difference between the two time values gives the duration of itinerary i_ϑ. For simplicity $S_u = ((i_1, t^{\text{ari}}_{i_1}, t^{\text{dept}}_{i_1}), ..., (i_n, t^{\text{ari}}_{i_n}, t^{\text{dept}}_{i_n}))$ can be written as $S_u = (i_1, ..., i_n)$.

3.5 Itinerary Interest

POIs $\mathbb{P} = (\mathbb{P}_1, \mathbb{P}_2, \mathbb{P}_3, .., \mathbb{P}_k)$ can be used to form the itinerary of the tourist. An interest value is associated with each itinerary $i_\vartheta \in S_u$. This value can be obtained from Eqn. 3:

$$i_\vartheta(\text{Intr}) = \sum_{j=1}^{k} \frac{(t^{\text{ari}}_{\mathbb{P}_j} - t^{\text{dept}}_{\mathbb{P}_j})}{\mathcal{D}(\mathbb{P}_j)}. \tag{3}$$

3.6 Itinerary Popularity

Each itinerary also has a popularity associated with it, which can be obtained using equations 4 and 5. We first obtain the popularity of a user for a POI category c, denoted by $\mathcal{C}(popl)$:

$$\mathcal{C}(popl) = \sum_{j=1}^{k} \frac{popl_{\mathbb{P}_j}}{\Phi(\mathbb{P}_j)} \; \delta(\mathcal{C}_{\mathbb{P}_j} = c), \tag{4}$$

where $popl_{\mathbb{P}_j}$ is the user's visit frequency to POI $\mathbb{P}_j$. The visit frequency of all the users at POI $\mathbb{P}_j$ is given by $\Phi(\mathbb{P}_j)$. The popularity of an itinerary is then given by the following equation:

$$i_\vartheta(popl) = \sum_{\varsigma=1}^{\omega} \mathcal{C}_\varsigma(popl), \tag{5}$$

where $\varsigma = \{1, 2, 3, ..., \omega\}$ is the total number of categories present in i_ϑ.

3.7 Travel Costs

Travel costs are calculated by the physical distance that has been traveled along the journey. Many earlier works take the entire travel into account. But time is dependent on means of travel like taxis, trains, airlines, walks, etc.

The distance is a significant factor if the visitors want to travel multiple POIs using a broad transportation system. We reduce journey times by using quick way of transportation.

If two POIs are a long way from each other, a fast transportation method is required and the costs of transport gets increased. We consequently seek to maintain a minimal level of the entire physical distance of the trip. Travel expenses are determined by Eqn. 6:

$$\begin{aligned} \mathcal{T}^{\text{cost}}(x) = \sum_{\vartheta=1}^{n} \sum_{j=2}^{k} \Pi^{\text{inter}}(i_\vartheta^{\mathbb{P}_{j-1,j}}) + \\ \sum_{\vartheta=1}^{n} \Pi^{\text{exter}}(i_\vartheta^{\mathbb{P}_n}, i_{\vartheta+1}^{\mathbb{P}_1}), \end{aligned} \tag{6}$$

where, $(\vartheta + 1) < n$. The double summation in Eqn. 6 gives the total distance between the POI attractions of all the itineraries present in the travel package. The overall physical radius among all POI attractions is calculated based on internal distance i_ϑ.

The second summation in Eqn. 6 is the external distance between i_ϑ and $i_\vartheta + 1$ and it is calculated using the physical distance between the last POI of itinerary i_ϑ and the first POI of itinerary $i_\vartheta + 1$.

3.8 LU and GU's Similarity

The degree of similarity between local and global tourists can be determined based on their interests for a given destination. For two distinct tourists u_x and u_y, we can compute their similarity using the cosine similarity measure as shown in equation 7:

$$\text{Cos\_sim}(u_x,\ u_y) = \frac{\vec{\text{Intr}}_{u_x} \cdot \vec{\text{Intr}}_{u_y}}{||\vec{\text{Intr}}_{u_x}|| \cdot ||\vec{\text{Intr}}_{u_y}||}. \tag{7}$$

3.9 Users to Tour Group Allocation

Because the tourist group is divided into m tours, let $\mathcal{G} = \{\mathcal{G}_1, ..., \mathcal{G}_m\}$ is the tourists group's category, and $\mathcal{G}_k = \{g_1, ..., g_q\}$ indicates a k^{th} group includes q tourists. Our goal is to establish the following groups:

$$\text{Max } \frac{\overrightarrow{\text{Int}}_{g_x} \cdot \overrightarrow{\text{Int}}_{g_y}}{||\overrightarrow{\text{Int}}_{g_x}|| \cdot ||\overrightarrow{\text{Int}}_{g_y}||}; g_x,\ g_y \in \mathcal{G}, \forall \mathcal{G}; \mathcal{G} \in \mathcal{G}. \quad (8)$$

The cosine similarity metric indicates how similar two users' interests are. $\mathcal{G}k$, and for all tour groups, in Eqn. 8.

This clustering issue has been demonstrated to have optimum solutions that are NP-hard. As a result, we employ the following approach to provide approximations to the solution to this issue.

3.10 Density-based spatial clustering of applications with noise (DBSCAN)

Based on a collection of points and the computation of euclidean distance and total number of points, DBSCAN grouped points that were similar to one another.

Outliers are frequently categorised as the points in low-density zones. This is what Euclidean distance deals with:

$$d_e = \sqrt{\begin{aligned}&(\text{Int}_{\mu_1}(c_1) - \text{Int}_{\mu_2}(c_1)) + \dots \\ &+(\text{Int}_{\mu_1}(c_n) - \text{Int}_{\mu_2}(c_n)).\end{aligned}} \quad (9)$$

3.11 Problem definition

In this part, we will address the recommendation for multiple itineraries by considering POIs for a specific tourist. The major objective is to enhance visitor and POI's popularity and to decrease traveling expenses.

The optimization issue is known as the gTravelREC problem classified as the [9] version of the Orienteering Problem. This portion deals with the problem of different POIs for one person.

Our main objective is to maximize the interest and popularity of visitors and to reduce expenditures.

A type of orienteering issue [8] could be used to resolve this issue:

$$\mathbb{O}_{\dot{y}} = \frac{\left(\Theta\mathcal{P}_{\dot{y}}(\overline{\text{int}}) + (1-\Theta)\mathcal{P}_{\dot{y}}(\overline{\text{pop}})\right) + W(\text{inte})}{\text{Cost}(\mathcal{P}_{\dot{y}})}, \quad (10)$$

$$\mathcal{X}_{1\mathcal{J}} = \sum_{k=1}^{\mathcal{J}} \mathcal{S}_k, \quad (11)$$

$$\mathcal{X}_{1(\mathcal{J}+1)} = \mathcal{X}_{\mathcal{J}} - \mathcal{S}_{\mathcal{J}} + \mathcal{S}_{\mathcal{J}+1}, \quad (12)$$

$$\mathcal{X}_{1(\mathcal{J}+2)} = \mathcal{X}_{(\mathcal{J}+1)} - \mathcal{S}_{(\mathcal{J}+1)} + \mathcal{S}_{(\mathcal{J}+2)}, \quad (13)$$

$$\vdots$$

$$\mathcal{X}_{1n} = \mathcal{X}_{(n-1)} - \mathcal{S}_{(n-1)} + \mathcal{S}_n, \quad (14)$$

$$\mathcal{X}_{2(\mathcal{J}+1)} = \sum_{k=2}^{\mathcal{J}+1} \mathcal{S}_k, \quad (15)$$

$$\mathcal{X}_{2(\mathcal{J}+2)} = \mathcal{X}_{2(\mathcal{J}+1)} - \mathcal{S}_{(\mathcal{J}+1)} + \mathcal{S}_{(\mathcal{J}+2)}, \quad (16)$$

$$\mathcal{X}_{2(\mathcal{J}+3)} = \mathcal{X}_{2(\mathcal{J}+2)} - \mathcal{S}_{(\mathcal{J}+2)} + \mathcal{S}_{(\mathcal{J}+3)}, \quad (17)$$

$$\vdots$$

$$\mathcal{X}_{2n} = \mathcal{X}_{2(n-1)} - \mathcal{S}_{(n-1)} + \mathcal{S}_n, \quad (18)$$

$$\vdots$$

$$\mathcal{X}_{(n-\mathcal{J}+1)n} = \sum_{k=(n-\mathcal{J}+1)}^{n} \mathcal{S}_k, \quad (19)$$

$$\gamma(i) = \frac{\text{Intr}(c_i)}{\sum_{j=1}^{\vartheta} \text{Intr}(c_j)}, \quad (20)$$

$$\delta(i) = \frac{\text{popl}(c_i)}{\sum_{j=1}^{\vartheta} \text{popl}(c_j)}, \quad (21)$$

$$\begin{aligned}\mathcal{L}(x) = (&\mathcal{X}_{1\mathcal{J}}, \mathcal{X}_{1(\mathcal{J}+1)}, .., \mathcal{X}_{1n}, \mathcal{X}_{2(\mathcal{J}+1)}, \\ &\mathcal{X}_{2(\mathcal{J}+2)}, ..., \mathcal{X}_{(n-\mathcal{J}+1)n}).\end{aligned} \quad (22)$$

Eqns. 11-19 are the different formulations of the itinerary. $\mathcal{X}_{i,\mathcal{J}}$ represents the total of A in the list of i^{th} itineraries of sizes $\mathcal{J}$ for all itineraries accessible.

This work aims mainly to propose several itineraries $i_1, i_2, .., i_n$ for maximizing visitor interest and for reducing travel costs. So, the goal can be written as:

$$\text{Minimize}(\mathcal{L}(x)), \quad (23)$$

$$\text{Minimize}(\mathcal{T}^{\text{cost}}(x)). \quad (24)$$

Algorithm 1: Algorithm for NSGA-II based gTravel approach.

$\mathcal{P}^{\text{size}} \leftarrow len(\mathcal{S}^{\text{sol}})$ ▷ *Population size*
$\mathcal{G}^{\text{no}} \leftarrow 0$
$\mathcal{I}_l \leftarrow 0$
for $\mathcal{U}$ *in* matching_user_list **do**
  $\mathcal{I}_l \leftarrow$ *($\mathcal{U}$, itinerary)* ▷ *itineraries of similar global users*
$\mathcal{S}^{\text{sol}} \leftarrow \mathcal{I}_l$
while *($\mathcal{G}^{no} < \text{max\_gen}$)* **do**
  $\mathcal{F}_1 \leftarrow$ *Calculate* fitness_1, $\mathcal{S}^{\text{sol}}$
  $\mathcal{F}_2 \leftarrow$ *Calculate* fitness_2, $\mathcal{S}^{\text{sol}}$
  $\mathcal{N}_d \leftarrow \text{non\_dominated\_sort}(\mathcal{F}_1, \mathcal{F}_2)$ ▷ *NSGA-II's fast non dominated sort*
  $\mathcal{C}_{\text{crowd}} = [\,]$
  for *each* $\mathcal{N}_d$ **do**
    $\mathcal{C}_{\text{crowd}}.\text{append}(\text{crowding\_distance}(\mathcal{F}_1, \mathcal{F}_2, \mathcal{N}_d))$ ▷ *Calculate crowding distance*
  $\mathcal{S}_1^{\text{sol}} = \mathcal{S}^{\text{sol}}$
  for *each* $\mathcal{S}_1^{\text{sol}}$ **do**
    $\mathcal{S}_2^{\text{sol}} = \text{Calculate-Crossover-Mutation}(\mathcal{S}_1^{\text{sol}})$
  $\mathcal{F}'_1 \leftarrow$ *Calculate* fitness_1, $\mathcal{S}_2$
  $\mathcal{F}'_2 \leftarrow$ *Calculate* fitness_2, $\mathcal{S}_2$
  $\mathcal{N}'_d \leftarrow \text{non\_dominated\_sort}(\mathcal{F}'_1, \mathcal{F}'_2)$
  $\mathcal{C}'_{\text{crowd}} = [\,]$
  for *each* $\mathcal{N}'_d$ **do**
    $\mathcal{C}'_{\text{crowd}}.\text{append}(\text{crowding\_distance}(\mathcal{F}'_1, \mathcal{F}'_2, \mathcal{N}'_d))$ ▷ *Calculate crowding distance*
  $\mathcal{S}_{\text{new}}^{\text{sol}} = [\,]$
  for *each* $\mathcal{N}'_d$ **do**
    $\text{front} = \text{Sort}(\mathcal{N}'_d, \mathcal{C}'_{\text{crowd}})$ ▷ *Function to sort by values*
    for *value in front* **do**
      $\mathcal{S}_{\text{new}}^{\text{sol}}.\text{append}(\text{value})$
      if $(\text{len}(\mathcal{S}_{\text{new}}^{\text{sol}}) == \mathcal{P}^{\text{size}})$ **then**
        break
  $\mathcal{S}^{\text{sol}} = \mathcal{S}_{\text{new}}^{\text{sol}}$ ▷ *Update the solution*
  $\mathcal{G} = \mathcal{G} + 1$

4 Experimental Methodology

4.1 Dataset

In this analysis, we utilized the data provided in [8].

The dataset contains images and videos by Yahoo! Flickr Creative Commons 100M (YFCC100M) [13]. Furthermore, the YFCC100M data set provided in Table 1 was used and geo-tagged images from different areas of the globe have been obtained.

The data set comprises the photo's meta-data. It includes visiting dates and times.

The dataset also contains data from the Geo-coordinate to identify the length among POIs. The data sets utilized in this research could be accessed from this url[1] for free.

4.2 Baseline Algorithms

As per [8], we have taken into account in the experimentation that all benchmark approaches begin at one POI and then choose the following POIs till the target is met.

We utilize the series of POIs to suggest different itineraries. After creating a single itinerary, we used the remaining POIs.

We assumed four hours of travel time in a given route, which might easily aid in the creation of a variety of itineraries.

- **Greedy Nearest (G-NEAR)**: We utilize this method for selecting the next unexplored POI by choosing the neighboring destinations [9].
- **Greedy Most Popular (G-POP)**: By selecting the top three attractions, we choose an unvisited POI [9].
- **Multiple Itinerary Tourist Recommendation (MULTITOUR)**: It provides many itineraries, taking into account the interest and popularity of the attractions and the expense of the trip [7].
- **Personalized Tour Recommendation (PERSTOUR-P)**: In terms of its interest and popularity, this provides single itinerary [9].
- **Trip Builder (TRIPBUILD)**: This generates a personalized tourist itinerary based on the interest and popularity [7].

[1] sites.google.com/site/limkwanhui/datacode?authuser=0

ISSN 2007-9737

Table 1. Datasets description

Town	# of Users	POI Visits	Tourist's Travel Sequences
Osaka	450	7891	145
Edinburgh	1454	34200	5102
Delhi	279	4001	512
Vienna	1155	34603	3234
Glasgow	601	11532	2298

4.3 Real-life Evaluation

Only tourists who have visited at least two sequences or more would be evaluated. The system is used locally and globally, and the related users were defined.

In this analysis, we can assess the corresponding visitors by selecting the top ten linked visitors of the GU's list. From the local data set the various attributes of the relevant visitors are collected.

We pick the following matrices to compare our solution with different benchmarks. For our experiments, the series of real-life sequences are chosen depending on the prior histories of the visitors in an area a tourist desires to explore.

– **Tour Recall (TourRec(I))**: Let C_{rec} be the list of categories present in the suggested itinerary. C_{real} be the collection of categories that are visited in a real-life tour by travelers. The $\mathrm{TourRec}(I)$ is presented with Eqn. 25:

$$\mathrm{TourRec}(I) = \frac{\|C_{\mathrm{rec}} \ \cap \ C_{\mathrm{real}}\|}{\|C_{\mathrm{real}}\|}. \qquad (25)$$

– **Tour Precision (TourPre(I))**: $\mathrm{TourPrecision}$ can be expressed as shown in Eqn. 26:

$$\mathrm{TourPre}(I) = \frac{\|C_{\mathrm{rec}} \ \cap \ C_{\mathrm{real}}\|}{\|C_{\mathrm{rec}}\|}. \qquad (26)$$

– **Tour F1-Score (TourF1-score(I))**: Tour F1-Score can be calculated using Eqn. 27:

$$\mathrm{TourF1-score}(I) = \frac{2 \cdot \mathrm{TourPre}(I) \cdot \mathrm{TourRec}(I)}{\mathrm{TourPre}(I) + \mathrm{TourRec}(I)}. \qquad (27)$$

4.4 Comparison of Precision, Recall and F1-Score

With respect to the benchmark approaches like GPop TOURINT and GNear, the efficiency of the gTravel algorithm is maximum. Tables 2, 3 and 4 present the values of Precision, Recall and F1-Score of the gTravel algorithm and other benchmark approaches.

The results show that in comparison with benchmark approaches, the proposed gTravel algorithm performs better. The Recall scores of the gTravel algorithm is $6.86\% - 26.73\%$ higher compared with other benchmark approaches (see 3). The Recall value depends on values $\|C_v\|$ and $\|C_{\mathrm{rec}} \cap C_{\mathrm{real}}\|$.

In case of gTravel algorithm, the value of $\|C_{\mathrm{rec}} \cap C_{\mathrm{real}}\|$ is higher than the various benchmark approaches. The gTravel algorithm typically runs on two datasets, local and global, and suggests various itineraries. This results in increased Recall values. The popularity of the tour and the interest of visitors is compared with different benchmark approaches such as GPop, TOURINT and GNear.

The Precision value of the gTravel algorithm is higher compared to other benchmark approaches from $4.71\% - 20.49\%$. The precision values depend on $\|C_{\mathrm{rec}}\|$ and $\|C_{\mathrm{rec}} \cap C_{\mathrm{real}}\|$. We found through experiments that C_{rec} values vary for various benchmark approaches.

The $\|C_{\mathrm{rec}} \cap C_{\mathrm{real}}\|$ values are higher for the gTravel algorithm. The F1-Score value of gTravel algorithm is $4.07\% - 22.13\%$, higher compared to other benchmark approaches, as the F1-Score values depend on Precision and Recall.

The performance of MULTITOUR is higher than the various benchmark approaches because it considers the popularity of itineraries, the interest of tourists and traveling expenses.

Table 2. Precision values for gTravel and multiple benchmark approaches

Algorithms	gTravel	MULTITOUR	PERSTOUR	TRIPBUILD	GPOP	GNEAR	RAND
Delhi-Edinburgh	0.457 ± 0.019 1	0.374 ± 0.028 3	0.388 ± 0.039 2	0.280 ± 0.014 6	0.340 ± 0.037 4	0.316 ± 0.038 5	0.244 ± 0.029 7
Osaka-Edinburgh	0.460 ± 0.017 2	0.442 ± 0.011 3	0.491 ± 0.024 1	0.429 ± 0.023 4	0.333 ± 0.039 7	0.367 ± 0.022 5	0.337 ± 0.032 6
Vienna-Edinburgh	0.413 ± 0.048 1	0.300 ± 0.047 4	0.380 ± 0.046 2	0.272 ± 0.032 5	0.337 ± 0.043 3	0.231 ± 0.042 6	0.216 ± 0.011 7
Delhi-Osaka	0.390 ± 0.019 2	0.469 ± 0.025 1	0.329 ± 0.038 4	0.357 ± 0.027 3	0.321 ± 0.035 5	0.260 ± 0.029 7	0.275 ± 0.017 6
Glasgow-Edinburgh	0.395 ± 0.027 1	0.286 ± 0.035 3	0.329 ± 0.022 2	0.254 ± 0.023 6	0.229 ± 0.034 7	0.273 ± 0.050 4	0.256 ± 0.016 5
Delhi-Buda	0.531 ± 0.030 2	0.598 ± 0.038 1	0.484 ± 0.010 3	0.414 ± 0.042 6	0.411 ± 0.041 4	0.376 ± 0.012 5	0.382 ± 0.011 7
Buda- Edinburgh	0.366 ± 0.032 1	0.289 ± 0.044 3	0.272 ± 0.049 4	0.304 ± 0.023 2	0.209 ± 0.048 7	0.238 ± 0.017 5	0.217 ± 0.028 6
Delhi-Vienna	0.447 ± 0.027 1	0.366 ± 0.035 2	0.330 ± 0.022 4	0.356 ± 0.037 3	0.311 ± 0.019 5	0.301 ± 0.029 6	0.273 ± 0.017 7

Table 3. Recall values for gTravel and multiple benchmark approaches

Algorithms	gTravel	MULTITOUR	PERSTOUR	TRIPBUILD	GPOP	GNEAR	RAND
Delhi-Edinburgh	0.375 ± 0.023 1	0.304 ± 0.050 3	0.339 ± 0.043 2	0.232 ± 0.014 6	0.286 ± 0.019 4	0.268 ± 0.028 5	0.196 ± 0.039 7
Osaka-Edinburgh	0.418 ± 0.039 2	0.345 ± 0.022 4	0.509 ± 0.032 1	0.382 ± 0.036 3	0.291 ± 0.017 6	0.327 ± 0.011 5	0.273 ± 0.024 7
Vienna-Edinburgh	0.365 ± 0.033 1	0.231 ± 0.024 4	0.288 ± 0.015 2	0.212 ± 0.031 5	0.269 ± 0.043 3	0.173 ± 0.042 6	0.154 ± 0.011 7
Delhi-Osaka	0.327 ± 0.009 2	0.388 ± 0.018 1	0.286 ± 0.042 4	0.306 ± 0.027 3	0.265 ± 0.035 5	0.204 ± 0.029 7	0.224 ± 0.017 6
Glasgow-Edinburgh	0.340 ± 0.023 1	0.234 ± 0.034 4	0.277 ± 0.050 2	0.191 ± 0.016 6	0.170 ± 0.039 7	0.255 ± 0.020 3	0.213 ± 0.014 5
Delhi-Buda	0.491 ± 0.024 2	0.547 ± 0.016 1	0.434 ± 0.013 4	0.453 ± 0.014 3	0.415 ± 0.041 5	0.358 ± 0.012 7	0.396 ± 0.011 6
Buda-Edinburgh	0.326 ± 0.038 1	0.283 ± 0.026 2	0.239 ± 0.050 4	0.261 ± 0.014 3	0.152 ± 0.032 7	0.217 ± 0.044 5	0.196 ± 0.049 6
Delhi-Vienna	0.412 ± 0.037 1	0.333 ± 0.019 3	0.294 ± 0.029 5	0.353 ± 0.017 2	0.314 ± 0.018 4	0.275 ± 0.038 6	0.235 ± 0.028 7

Table 4. F1-Score values for gTravel and multiple benchmark approaches

Algorithms	gTravel	MULTITOUR	PERSTOUR	TRIPBUILD	GPOP	GNEAR	RAND
Delhi-Edinburgh	0.457 ± 0.019 1	0.374 ± 0.028 3	0.388 ± 0.039 2	0.280 ± 0.014 6	0.340 ± 0.037 4	0.316 ± 0.038 5	0.244 ± 0.029 7
Osaka-Edinburgh	0.460 ± 0.017 2	0.442 ± 0.011 3	0.491 ± 0.024 1	0.429 ± 0.023 4	0.333 ± 0.039 7	0.367 ± 0.022 5	0.337 ± 0.032 6
Vienna-Edinburgh	0.413 ± 0.048 1	0.300 ± 0.047 4	0.380 ± 0.046 2	0.272 ± 0.032 5	0.337 ± 0.043 3	0.231 ± 0.042 6	0.216 ± 0.011 7
Delhi-Osaka	0.390 ± 0.019 2	0.469 ± 0.025 1	0.329 ± 0.038 4	0.357 ± 0.027 3	0.321 ± 0.035 5	0.260 ± 0.029 7	0.275 ± 0.017 6
Glasgow-Edinburgh	0.395 ± 0.027 1	0.286 ± 0.035 3	0.329 ± 0.022 2	0.254 ± 0.023 6	0.229 ± 0.034 7	0.273 ± 0.050 4	0.256 ± 0.016 5
Delhi-Buda	0.531 ± 0.030 2	0.598 ± 0.038 1	0.484 ± 0.010 3	0.414 ± 0.042 6	0.411 ± 0.041 4	0.376 ± 0.012 5	0.382 ± 0.011 7
Buda-Edinburgh	0.366 ± 0.032 1	0.289 ± 0.044 3	0.272 ± 0.049 4	0.304 ± 0.023 2	0.209 ± 0.048 7	0.238 ± 0.017 5	0.217 ± 0.028 6
Delhi-Vienna	0.447 ± 0.027 1	0.366 ± 0.035 2	0.330 ± 0.022 4	0.356 ± 0.037 3	0.311 ± 0.019 5	0.301 ± 0.029 6	0.273 ± 0.017 7

However, the other benchmarks do not support many itineraries and evaluate the popularity of attractions or interest of visitors in the location.

5 Conclusion and Future Work

In this research, we have offered a method gTravel that contributes to maximize the tourist interest, popularity,weather interest and reduced costs. Geo-tagged photos are used by gTravel to show the tourists' actual travel patterns.

Tourist interest, tour popularity, weather interest and traveling costs are calculated effectively for training the gTravel algorithm.

The suggested method is dependent on the selection of many POIs by taking into account the POI time visiting factor.

gTravel will not depend on the traveling history of a certain individual in new locations. The case in which a visitor wants to visit new places is therefore taken into consideration.

b) Tourist has many POIs (c) the weather interest is calculated. Given the Flickr data in several cities, we matched gTravel with various baselines that take multiple criteria such as Precision, Recall, and F1-Score.

The findings of the study demonstrate that the suggested gTravel algorithm in most situations surpasses baseline approaches.

We wish to enhance this research in the future to several travelers who intend to be staying in a new location over many days.

References

1. **Anagnostopoulos, A., Atassi, R., Becchetti, L., Fazzone, A., Silvestri, F. (2016).** Tour recommendation for groups. Data Mining and Knowledge Discovery, Vol. 31, No. 5, pp. 1157–1188. DOI: 10.1007/s10618-016-0477-7.
2. **Brilhante, I., Macedo, J. A., Nardini, F. M., Perego, R., Renso, C. (2014).** TripBuilder: A tool for recommending sightseeing tours. Lecture Notes in Computer Science, Springer International Publishing, pp. 771–774. DOI: 10.1007/978-3-319-06028-6_93.
3. **Chen, C., Zhang, D., Guo, B., Ma, X., Pan, G., Wu, Z. (2015).** Tripplanner: Personalized trip planning leveraging heterogeneous crowdsourced digital footprints. IEEE Transactions on Intelligent Transportation Systems, Vol. 16, No. 3, pp. 1259–1273. DOI: 10.1109/TITS.2014.2357835.
4. **De-Choudhury, M., Feldman, M., Amer-Yahia, S., Golbandi, N., Lempel, R., Yu, C. (2010).** Automatic construction of travel itineraries using social breadcrumbs. Proceedings of the 21st ACM Conference on Hypertext and Hypermedia, ACM, pp. 35–44. DOI: 10.1145/1810617.1810626.
5. **Dumitrescu, L., Fuciu, M. (2015).** Consumer behaviour in the tourist segmentation process–a marketing research. Studies in Business and Economics, Vol. 10, No. 1, pp. 66–76. DOI: 10.1515/sbe-2015-0005.
6. **Lal-Sarkar, J., Majumder, A. (2023).** MultiCity: A personalized multi-itinerary city recommendation engine. Computational Intelligence and Data Analytics, Springer Nature Singapore, pp. 509–520.
7. **Lal-Sarkar, J., Majumder, A., Rani-Panigrahi, C., Roy, S. (2020).** MULTITOUR: A multiple itinerary tourists recommendation engine. Electronic Commerce Research and Applications, Vol. 40, pp. 100943. DOI: 10.1016/j.elerap.2020.100943.
8. **Lim, K. H., Chan, J., Karunasekera, S., Leckie, C. (2017).** Personalized itinerary recommendation with queuing time awareness. Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, pp. 325–334. DOI: 10.1145/3077136.3080778.
9. **Lim, K. H., Wang, X., Chan, J., Karunasekera, S., Leckie, C., Chen, Y., Loong-Tan, C., Quan-Gao, F., Ken-Wee, T. (2016).** Perstour: A personalized tour recommendation and planning system. Extended Proceedings of the 27th ACM Conference on Hypertext and Social Media.
10. **Sarkar, J. L., Majumder, A. (2021).** A new point-of-interest approach based on multi-itinerary recommendation engine. Expert Systems with Applications, Vol. 181, pp. 115026. DOI: 10.1016/j.eswa.2021.115026.
11. **Sarkar, J. L., Majumder, A. (2022).** gTour: Multiple itinerary recommendation engine for group of tourists. Expert Systems with Applications, Vol. 191, pp. 116190. DOI: 10.1016/j.eswa.2021.116190.
12. **Sarkar, J. L., Majumder, A., Panigrahi, C. R., Roy, S., Pati, B. (2022).** Tourism recommendation system: A survey and future research directions. Multimedia Tools and Applications, Vol. 82, No. 6, pp. 8983–9027. DOI: 10.1007/s11042-022-12167-w.
13. **Thomee, B., Shamma, D. A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., Li, L. J. (2016).** YFCC100M: The new data in multimedia research. Communications of the ACM, Vol. 59, No. 2, pp. 64–73. DOI: 10.1145/2812802.

Article received on 06/03/2023; accepted on 23/05/2023.
Corresponding author is Bibudhendu Pati.