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 wonderland 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 \( I = \{i_1, i_2, i_3, \ldots, i_n\} \) be the itineraries in a specific town. Each itinerary \( i_0 \in I \) \((1 \leq \vartheta \leq n)\) consists of a set of POIs \( P = \{P_1, P_2, P_3, \ldots, P_k\} \).

Each POI is associated with one or more categories \( C \) 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 \( i_0 = \{P_1, P_2, P_3, \ldots, P_k\} \), where \( k \) is the size of the itinerary \( i_0 \). The total distance covered by the tourist is calculated by adding up the distance from \( P_\gamma \) and \( P_{\gamma+1} \), 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:

\[
D(P) = \frac{\sum_{u=1}^{k} \sum_{j=1}^{\ell} (e^j_u - a^u_j)\delta(P_j = P)}{\sum_{u=1}^{\ell} V_u \delta(P_j = P)} \forall P \in P, \tag{1}
\]

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

3.3 Interest in LUs and GUs Depending on Time

Suppose \( C_\gamma \) is the category of some POI \( P \). The interest of a particular tourist for some category \( C \) is given by Eqn. 2:

\[
\text{Intr}_{u_i} C = \sum_{j=1}^{\ell} \frac{(P^\gamma_j - P^\gamma_{j-1})}{D(P_j)} \delta(C_{P^\gamma_j} = C) \forall C \in C, \tag{2}
\]

where, \( \delta(C_{P^\gamma_j}) = \{1 \text{ if } C_{P^\gamma_j} = C, 0 \text{ otherwise} \} \). Eqn. 2 will later be used to measure tourists’ interest for some POI category \( C \) 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 tourists. For some tourist $u \in \mathcal{U}$, we define a sequence of itineraries $S_u = ((i_1, \tau_{i_1}, t_{i_1}^{\text{dept}}), ..., (i_n, \tau_{i_n}, t_{i_n}^{\text{dept}}))$ where $n$ is the number of itineraries, in a triplet $(i_\vartheta, \tau_{i_\vartheta}, t_{i_\vartheta}^{\text{dept}})$, $i_\vartheta$ is an itinerary, $\tau_{i_\vartheta}$ is the time of entry and $t_{i_\vartheta}^{\text{dept}}$ is the time of departure. The difference between the two time values gives the duration of itinerary $i_\vartheta$. For simplicity $S_u = ((i_1, \tau_{i_1}, t_{i_1}^{\text{dept}}), ..., (i_n, \tau_{i_n}, t_{i_n}^{\text{dept}}))$ 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:

$$i_\vartheta(\text{Intr}) = \frac{\sum_{j=1}^{k} (\tau_{i_\vartheta} - t_{i_\vartheta}^{\text{dept}})}{D(\mathbb{P}_j)}.$$  \hspace{1cm} (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 $C(\text{popl})$:

$$C(\text{popl}) = \sum_{j=1}^{k} \frac{\text{popl}_{\mathbb{P}_j}}{\Phi(\mathbb{P}_j)} \delta(C_{\mathbb{P}_j} = c).$$  \hspace{1cm} (4)

where $\text{popl}_{\mathbb{P}_j}$ 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:

$$i_\vartheta(\text{popl}) = \sum_{c=1}^{\omega} C_c(\text{popl}),$$  \hspace{1cm} (5)

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

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:

$$T_{\text{cost}}(x) = \sum_{\vartheta=1}^{n} \sum_{j=2}^{k} \Pi_{\text{inter}}((i_\vartheta, j, i_{\vartheta+1}^j)) + \sum_{\vartheta=1}^{n} \Pi_{\text{exter}}((i_\vartheta, j)),$$

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 $i_\vartheta$.

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

$$\text{Cos}_\text{sim}(u_x, u_y) = \frac{\|\text{Intr}_{u_x} \cdot \text{Intr}_{u_y}\|}{\|\text{Intr}_{u_x}\| \cdot \|\text{Intr}_{u_y}\|}.$$  \hspace{1cm} (7)
3.9 Users to Tour Group Allocation

Because the tourist group is divided into m tours, let \( G = \{G_1, ..., G_m\} \) is the tourists group’s category, and \( G_k = \{g_1, ..., g_q\} \) indicates a \( k^{th} \) group includes \( q \) tourists. Our goal is to establish the following groups:

\[
\max \frac{\overrightarrow{\text{Int}_{g_1}} \cdot \overrightarrow{\text{Int}_{g_2}}}{||\text{Int}_{g_1}|| \cdot ||\text{Int}_{g_2}||}; g_x, g_y \in G, \forall G; G \in G. \tag{8}
\]

The cosine similarity metric indicates how similar two users’ interests are. \( G_k \), 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:

\[
d_e = \sqrt{\frac{\text{Int}_{\mu_1}(c_1) - \text{Int}_{\mu_2}(c_1)}{||\text{Int}_{\mu_1}||} + \ldots + \frac{\text{Int}_{\mu_1}(c_n) - \text{Int}_{\mu_2}(c_n)}{||\text{Int}_{\mu_2}||}}. \tag{9}
\]

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:

\[
O_y = \frac{(\Theta P_y(\text{int}) + (1 - \Theta) P_y(\text{pop}))) + W(\text{inte})}{\text{Cost}(P_y)}, \tag{10}
\]

\[
X_{1,J} = \sum_{k=1}^{J} S_k, \tag{11}
\]

\[
X_{1,(J+1)} = X_J - S_J + S_{J+1}, \tag{12}
\]

\[
X_{1,(J+2)} = X_{1,(J+1)} - S_{(J+1)} + S_{(J+2)}, \tag{13}
\]

\[
\vdots
\]

\[
X_{1n} = X_{1(n-1)} - S_{(n-1)} + S_n, \tag{14}
\]

\[
X_{2(J+1)} = \sum_{k=2}^{J+1} S_k, \tag{15}
\]

\[
X_{2,(J+1)} = X_{2,(J+2)} - S_{(J+1)} + S_{(J+2)}; \tag{16}
\]

\[
X_{2,(J+3)} = X_{2,(J+2)} - S_{(J+2)} + S_{(J+3)}; \tag{17}
\]

\[
\vdots
\]

\[
X_{2n} = X_{2(n-1)} - S_{(n-1)} + S_n; \tag{18}
\]

\[
\vdots
\]

\[
X_{(n-J)+1} = \sum_{k=1}^{n-J+1} S_k, \tag{19}
\]

\[
\gamma(i) = \frac{\text{Intr}(c_i)}{\sum_{j=1}^{\gamma} \text{Intr}(c_j)}, \tag{20}
\]

\[
\delta(i) = \frac{\text{popl}(c_i)}{\sum_{j=1}^{\gamma} \text{popl}(c_j)}, \tag{21}
\]

\[
L(x) = (X_{1,J}, X_{1,(J+1)}, ..., X_{1n}, X_{2,(J+1)}, ..., X_{2(n-J+1)}).
\]

Eqns. 11-19 are the different formulations of the itinerary. \( X_{i,J} \) represents the total of \( A \) in the list of \( i^{th} \) itineraries of sizes \( J \) 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:

\[
\text{Minimize}(L(x)), \tag{23}
\]

\[
\text{Minimize}(T^{\text{cost}}(x)). \tag{24}
\]
4.1 Dataset

In this analysis, we utilized the data provided in [8].
### 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 $C_{\text{rec}}$ be the list of categories present in the suggested itinerary. $C_{\text{real}}$ be the collection of categories that are visited in a real-life tour by travelers. The TourRec($I$) is presented with Eqn. 25:

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

- **Tour Precision** (TourPre(I)): TourPrecision can be expressed as shown in Eqn. 26:

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

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

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

### 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 $|C_{\text{rec}} \cap C_{\text{real}}|$.

In case of gTravel algorithm, the value of $|C_{\text{rec}} \cap C_{\text{real}}|$ 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 $|C_{\text{rec}}|$ and $|C_{\text{rec}} \cap C_{\text{real}}|$. We found through experiments that $C_{\text{rec}}$ values vary for various benchmark approaches.

The $|C_{\text{rec}} \cap C_{\text{real}}|$ 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.
training the gTravel algorithm.

and traveling costs are calculated effectively for costs. Geo-tagged photos are used by gTravel to

In this research, we have offered a method

attractions or interest of visitors in the location.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>gTravel</th>
<th>MULTITOUR</th>
<th>PERSTOUR</th>
<th>TRIPBUILD</th>
<th>GPOP</th>
<th>GNEAR</th>
<th>RAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delhi-Edinburgh</td>
<td>0.457±0.019</td>
<td>0.374±0.028</td>
<td>0.388±0.039</td>
<td>0.280±0.014</td>
<td>0.340±0.037</td>
<td>0.316±0.038</td>
<td>0.244±0.029</td>
</tr>
<tr>
<td>Osaka-Edinburgh</td>
<td>0.460±0.017</td>
<td>0.442±0.011</td>
<td>0.491±0.024</td>
<td>0.429±0.023</td>
<td>0.333±0.039</td>
<td>0.367±0.022</td>
<td>0.337±0.032</td>
</tr>
<tr>
<td>Vienna-Edinburgh</td>
<td>0.413±0.048</td>
<td>0.300±0.047</td>
<td>0.380±0.046</td>
<td>0.272±0.032</td>
<td>0.337±0.043</td>
<td>0.231±0.042</td>
<td>0.216±0.011</td>
</tr>
<tr>
<td>Delhi-Osaka</td>
<td>0.390±0.019</td>
<td>0.469±0.025</td>
<td>0.329±0.038</td>
<td>0.357±0.027</td>
<td>0.321±0.035</td>
<td>0.260±0.029</td>
<td>0.275±0.017</td>
</tr>
<tr>
<td>Glasgow-Edinburgh</td>
<td>0.395±0.027</td>
<td>0.286±0.035</td>
<td>0.329±0.022</td>
<td>0.254±0.023</td>
<td>0.229±0.034</td>
<td>0.273±0.050</td>
<td>0.256±0.016</td>
</tr>
<tr>
<td>Delhi-Buda</td>
<td>0.531±0.030</td>
<td>0.598±0.038</td>
<td>0.484±0.010</td>
<td>0.414±0.042</td>
<td>0.411±0.041</td>
<td>0.376±0.012</td>
<td>0.382±0.011</td>
</tr>
<tr>
<td>Buda-Edinburgh</td>
<td>0.361±0.032</td>
<td>0.289±0.044</td>
<td>0.304±0.023</td>
<td>0.269±0.048</td>
<td>0.238±0.017</td>
<td>0.217±0.028</td>
<td></td>
</tr>
<tr>
<td>Delhi-Vienna</td>
<td>0.447±0.027</td>
<td>0.366±0.035</td>
<td>0.330±0.022</td>
<td>0.356±0.037</td>
<td>0.311±0.019</td>
<td>0.301±0.029</td>
<td>0.273±0.017</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>gTravel</th>
<th>MULTITOUR</th>
<th>PERSTOUR</th>
<th>TRIPBUILD</th>
<th>GPOP</th>
<th>GNEAR</th>
<th>RAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delhi-Edinburgh</td>
<td>0.375±0.023</td>
<td>0.304±0.050</td>
<td>0.339±0.043</td>
<td>0.232±0.014</td>
<td>0.286±0.019</td>
<td>0.268±0.028</td>
<td>0.196±0.039</td>
</tr>
<tr>
<td>Osaka-Edinburgh</td>
<td>0.418±0.039</td>
<td>0.345±0.022</td>
<td>0.509±0.032</td>
<td>0.382±0.036</td>
<td>0.291±0.017</td>
<td>0.327±0.011</td>
<td>0.273±0.024</td>
</tr>
<tr>
<td>Vienna-Edinburgh</td>
<td>0.365±0.033</td>
<td>0.231±0.024</td>
<td>0.288±0.015</td>
<td>0.212±0.031</td>
<td>0.269±0.043</td>
<td>0.173±0.042</td>
<td>0.154±0.011</td>
</tr>
<tr>
<td>Delhi-Osaka</td>
<td>0.327±0.009</td>
<td>0.388±0.018</td>
<td>0.296±0.042</td>
<td>0.306±0.027</td>
<td>0.265±0.035</td>
<td>0.204±0.029</td>
<td>0.224±0.017</td>
</tr>
<tr>
<td>Glasgow-Edinburgh</td>
<td>0.340±0.031</td>
<td>0.234±0.034</td>
<td>0.277±0.050</td>
<td>0.191±0.016</td>
<td>0.170±0.039</td>
<td>0.255±0.020</td>
<td>0.213±0.014</td>
</tr>
<tr>
<td>Delhi-Buda</td>
<td>0.491±0.024</td>
<td>0.547±0.016</td>
<td>0.434±0.013</td>
<td>0.453±0.014</td>
<td>0.415±0.041</td>
<td>0.358±0.012</td>
<td>0.396±0.011</td>
</tr>
<tr>
<td>Buda-Edinburgh</td>
<td>0.326±0.038</td>
<td>0.283±0.026</td>
<td>0.239±0.050</td>
<td>0.261±0.014</td>
<td>0.152±0.032</td>
<td>0.217±0.044</td>
<td>0.196±0.049</td>
</tr>
<tr>
<td>Delhi-Vienna</td>
<td>0.412±0.037</td>
<td>0.333±0.019</td>
<td>0.294±0.029</td>
<td>0.353±0.017</td>
<td>0.314±0.018</td>
<td>0.275±0.038</td>
<td>0.235±0.028</td>
</tr>
</tbody>
</table>

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.

Computación y Sistemas, Vol. 27, No. 3, 2023, pp. 667–674
doi: 10.13053/CyS-27-3-4550
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


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