

Algorithms for Customized Travel Itinerary Generation

Varsha T., Yamuna C. Shekar,*Sahana P Shankar

Faculty of Engineering and Technology, M.S. Ramaiah University of Applied Sciences, Bengaluru 560 054

*Contact Author E-mail: sahana.cs.et@msruas.ac.in

Abstract

The Algorithms for Customized Travel Itinerary Generation is a Java based web application which help people who love to travel in planning their tour better. This application plans the tour in such a way that the traveler can save more time in visiting maximum number of places instead of wasting time on planning where and what to visit after reaching the destination. This application takes the information of the city that you want to explore, lists the various tourist places in that particular city. The tourist places are arranged and chosen in lights of various preferences preferred by the user. The Application later analyses the preferences of the user and creates an itinerary for the traveler. This application is very useful to all those people who love to travel for planning a time efficient trips where traveler can save time to visit new place rather than getting lost in an unaware place.

KeyWords: Itinerary, Bellman Ford Algorithm, Prim's Algorithm

1. INTRODUCTION

The travel itinerary is important to satisfy the needs of travelers, due to increase in demand for tourism. A tour would only be successful when there is proper planning. Without a proper plan, traveler wouldn't be able to make a successful trip. While travelling to a new place, it is impossible to visit all the places without a proper schedule. Travel Itinerary helps in planning our trip in a more organized way.

The Algorithms for Customized Travel Itinerary Generation is a Java based Web Application that helps the tourists in planning their tour that they want to explore. The application aims at creating an itinerary which includes the places to visit and hotel booking based on the user preferences. This application plans the tour such that traveler can save more time in visiting maximum number of places instead of wasting time on planning where and what to visit after reaching the destination.

The Bellman Ford algorithm calculates shortest route from a single start vertex to the other vertices. The main idea of using this algorithm in this application is that an itinerary can be generated by applying this algorithm where places can be sorted based on distance. When we use this algorithm, a traveler can get a list of places he/she needs to travel one after the other so that the places that are near to the start place can be covered first. This saves so much of time to the traveler and the traveler can cover more places in least time.

2. RELATED WORK

In this paper, we discuss how it is important to generate an itinerary which satisfies the user based on individual needs and expectations. There are several works that has been done associated with planning travel itinerary and decision making. In [1] the authors included idea formation, trying to receive the related information, taking a final call and confirming the booking that has been done. Authors in [2] mainly considered big five personality traits.

In [3] a prototype called "ATIPS (Automatic Travel Itinerary Planning System)" was proposed which computed the score of every tourist spot based on user preference, time, value, popularity and distance. Authors of [4] suggested that many itinerary generation problems are based on orienteering problems where places of interest are suggested based on user preference and popularity. In [5] the authors tried solving routing problem using maximum category constraint and being able to generate an optimal itinerary.

Authors of [6] tried to amplify the total gain from places visited based on interest where planning an itinerary can be completed within a given time-use. In [7] a genetic algorithm was applied based on the interest of the user and trip constraint by reducing jams and queuing time. In [8] a probabilistic algorithm was applied in order to reduce the proportion of input for getting a quick execution time. Authors in [9] made use of an augmentation method of learning to try to fit the data in a probability function.

In [10] an application "Smart Travel Planner" was built which used user fed profiles to suggest places for other users to visit. In [11] authors studied a mathematical model which maximized risk – hedging capacity and minimized time-use. Authors of [12] proposed a mathematical model which tried to minimize the total time in travelling under total cost diminution.

In [13], [14] and [15] authors built models that had user preference as a common factor for optimizing objectives so as to obtain realistic travel trajectories.

The Algorithms for Customized Travel Itinerary Generation provide the user to sort the places based on the user's preference. The preferences include distance which uses Bellman Ford algorithm to calculate shortest route from starting vertex, cost which uses Prim's algorithm to provide minimum cost for entire trip and user ratings that are provided by other users in their feedback.

3. DESIGN OF THE SYSTEM

To design any system, it is important to draw an application outline. This section of paper encloses the application outline shown in fig. 2 with detailed description and pseudo code.

3.1 Application Outline

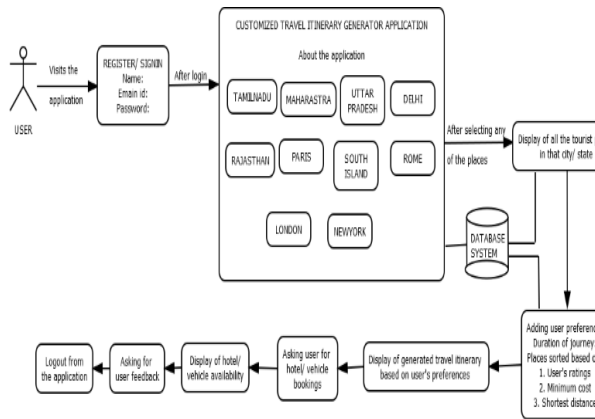


Fig. 1 Application Outline

The fig. 2 above represents the outline of the application. User will open the application and can either login in to the application or register to the application by providing necessary information. After Login in, the user will be taken to the homepage of the application. The homepage contains the most visited places in and around India. The user can select the place which he/she wants to visit in near future. When user selects the particular state the corresponding most visited cities are shown. User can select the city which he/ she wants to visit once the city is chosen, user is required to add the preferences. When the preferences are given, the itinerary is generated. Then

Here $M[v]$ is the length of shortest path, V is the vertex and E is edge of the graph.

4. IMPLEMENTATION

After the designing phase, there comes the implementation part. The following application was implemented in JAVA with MYSQL in the backend. A database was created to store the login/register user and to store the places with user ratings.

4.1 Implementation of GUI

Application with GUI may appear as simple but it is the most important part of any software application. A GUI is a type of interface which provides the available option to the users. The Functionality of an application can be customize perfectly, if GUI is difficult to utilize, at that point the program eventually will be a disappointment and the end client will probably pick something simpler or increasingly advantageous. Making a straightforward, simple to utilize GUI is crucial to the achievement of a software application.

The Fig.4 shows the GUI created for this application. The GUI provides the user to login / register to the application. Once the user registers to the application,

the application asks the user if he/she wants to book vehicle or hotel. The application will further ask the user to give the feedback. After giving feedback, user can logout from the application.

4.2 Algorithm Design

The Bellman Ford algorithm computes shortest route from single vertex. The main aspect of using the Bellman Ford algorithm is algorithm works fine for both weighted and un-weighted graphs. The below block provides the pseudo code for the above stated algorithm.

Bellmanford SP (G, s, t) {

For each node $v \in V$ {

$M[v] \leftarrow \infty$

Successor $[v] \leftarrow \phi$

}

$M[s] = 0$

For $i = 1$ to $n-1$ {

for each node $w \in V$ {

if ($M[w]$ has been updated in previous iteration) {

for each node v such that $(v, w) \in E$ {

if ($M[v] > M[w] + c_{vw}$) {

$M[v] \leftarrow M[w] + c_{vw}$

Successor $[v] \leftarrow w$

}}}

If no $M[w]$ value changed in iteration I , stop.

}}

the page is directed to Homepage where the user can see the popular places.

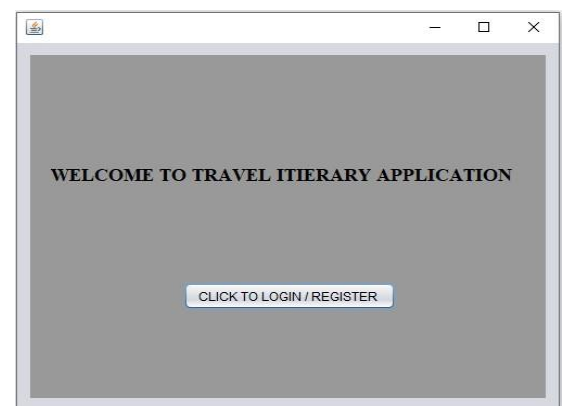


Fig. 2 Welcome Page

The System displays the most visited place within country and outside the country as well. On click of the places, displays the most tourist attraction of that place. In Fig. 5 shows when the Delhi city is selected, the system shows the tourist places of Delhi. The Application reduces the burden and time of the user in searching the attractions manually.

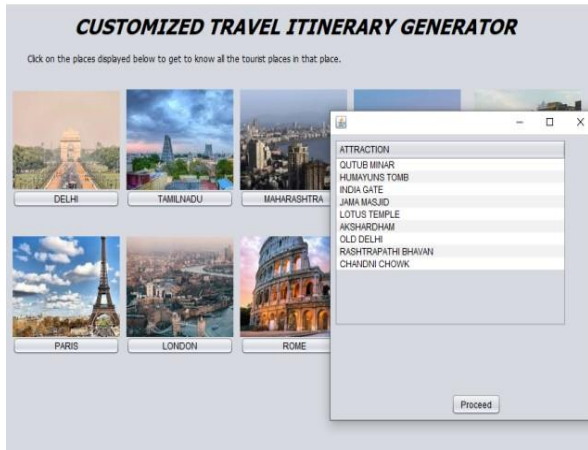


Fig. 3 Tourist Attraction in Delhi

The System provide the user to sort the tourist attraction based on distance, cost and user ratings. The Sorting is done in order to reduce the time and maximize the places visited throughout the tour. For sorting the places based on distance, the Bellman Ford algorithm is used. Using Bellman Ford algorithm we can sort the distance based on shorted path from single source vertex. For sorting the places based on cost, we make use of Prim's algorithm. This algorithm draws cost (graph) as input and tries to find the subset of the edge of that cost graph. It results a tree structure which include every vertex that has least total cost among all the structures formed. The places are sorted by MYSQL query for user rating option. In Fig.6 shows system with option to sort the places based on user's preference.

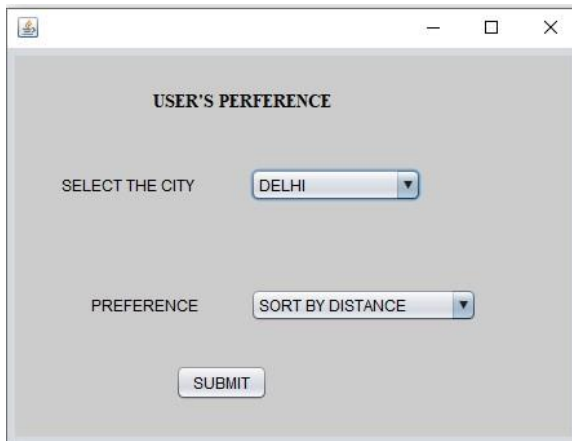


Fig. 4 Preference Page for User

5 RESULTS AND DISCUSSIONS

Implementation of GUI is done, but no application is complete without obtaining the result. In fig 7 the login is successfully for user who has already registered with the application. A pop menu appears to notify whether the user is new to the application or an existing user.



Fig. 5 Login Page for User

Once the place is selected and preference is given, the system should sort the attractions based on user's preference and should display the itinerary generated. In fig 8 the place Delhi is selected and sorted based on distance. The itinerary is generated and displayed to the user.

```

Distance of place Red Fort from source Red Fort is 0. It's path is [ 0 ]
Distance of place Qutub Minar from source Red Fort is 17. It's path is [ 0 1 ]
Distance of place Humayuns Tomb from source Red Fort is 8. It's path is [ 0 2 ]
Distance of place India Gate from source Red Fort is 6. It's path is [ 0 3 ]
Distance of place Jama Masjid from source Red Fort is 1. It's path is [ 0 4 ]
Distance of place Lotus Temple from source Red Fort is 15. It's path is [ 0 5 ]
Distance of place Akshardham from source Red Fort is 8. It's path is [ 0 6 ]
Distance of place Old Delhi from source Red Fort is 1. It's path is [ 0 7 ]
Distance of place Rashtrapathi Bhavan from source Red Fort is 8. It's path is [ 0 8 ]
Distance of place Chandni Chowks from source Red Fort is 1. It's path is [ 0 9 ]
path 0 = Red Fort
path 1 = Qutub Minar
path 2 = Humayuns Tomb
path 3= India Gate
path 4 = Jama Masjid
path 5 = Lotus Temple
path 6 = Akshardham
path 7 = Old Delhi
path 8 = Rashtrapathi Bhavan
path 9 = Chandni Chowk

```

Fig. 6 Sorted Based on Distance

6 CONCLUSION

The design and implementation of this application using the known algorithms like Bellman Ford and Prim's allow us to apply the theoretical concept to solve the real world practical problems like travel itinerary generation. With this users can see routes on a scheduled basis and tourism guidance information they need anytime and anywhere. By using this application, users can get details about major tourist attractions. The user can search the nearby tourist attractions along with their distance, cost and user ratings. System will also collect feedback from different tourists which will be

helpful for further development of application and other users to plan their trip more efficiently.

Nowadays, travelling is an important aspect. It is necessary for proper planning to be done before in terms of time management. Most of the people who don't make use of the latest technology waste a lot of time in planning trips as well as in a dilemma of which places to visit. So, an application like this can help tourists in utilizing their precious time and also at the same time enjoy their trip.

7 ACKNOWLEDGMENT

The authors would like to thank Dr. Sivaguru S Srithran, Vice Chancellor, RUAS, Dr. Govind. R. Kadambi, Pro. Vice Chancellor, RUAS, Dr. H.M RajashekaraSwamy Dean(I/C) Faculty of Engineering and Technology, RUAS and Dr. P.V.R.Murthy HOD(CS), Faculty of Engineering and Technology, RUAS, whose emphasis for quality work helped us keep focus on proposed work.

REFERENCES

1. P. Yochum, L. Chang, T. Gu and M. Zhu, "An Adaptive Genetic Algorithm for Personalized Itinerary Planning," in *IEEE Access*, vol. 8, pp. 88147-88157, 2020
doi: 10.1109/ACCESS.2020.2990916
2. Chiang, H. S., & Huang, T. C. (2015). User-adapted travel planning system for personalized schedule recommendation. *Information Fusion*, 21, 3-17.
3. Kadakia, A., Desai, D., Mistry, U., & D'silva, M. (2018). Voyageur. 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA). doi:10.1109/iccubea.2018.8697358
4. K. H. Lim, J. Chan, C. Leckie, and S. Karunasekera, "Personalized trip recommendation for tourists based on user interests, points of interest visit durations and visit recency," *Knowl. Inf. Syst.*, vol. 54, no. 2, pp. 375_406, Feb. 2018.
5. P. Bolzoni, S. Helmer, K. Wellenzohn, J. Gamperand P. Andritsos, "Efficient itinerary planning with category constraints," in *Proc. 22nd Int. Conf. Adv. Geographic Inf. Syst. (SIGSPATIAL)*, 2014, pp. 203_212.
6. P. Padia, B. Singhal, and K. H. Lim, "User-relative personalized tour recommendation," in *Proc. IUI Workshops*, 2019, pp. 1_6.
7. L. Liu, J. Xu, S. S. Liao, and H. Chen, "A real-time personalized route recommendation system for self-drive tourists based on vehicle to vehicle communication," *Expert Syst. Appl.*, vol. 41, no. 7, pp. 3409_3417, Jun. 2014.
8. P. Bolzoni, F. Persia, and S. Helmer, "Itinerary planning with category constraints using a probabilistic approach," in *Proc. DEXA*, vol. 2, 2017, pp. 363_377.
9. R. Gaonkar, M. Tavakol, and U. Brefeld, "MDP-based itinerary recommendation using Geo-tagged social media," in *Proc. IDA*, 2018, pp. 111_123.
10. R. Jafri, "Smart Travel Planner: A mashup of travel-related web services", *Current Trends in Information Technology (CTIT)*, pp. 181-185, 2013. [Accessed: 15- Aug- 2017]
11. Y. Zhang and J. Tang, "Itinerary planning with time budget for risk-aversetravelers," *Eur. J. Oper. Res.*, vol. 267, no. 1, pp. 288_303, May 2018.
12. D. Jiaoman, L. Lei, and L. Xiang, "Travel planning problem considering site selection and itinerary making," in *Proc. Conf. Res. Adapt. ConvergentSyst. (RACS)*, 2018, pp. 29_36.
13. L. Hang, S.-H. Kang, W. Jin, and D.-H. Kim, Design and implementation of an optimal travel route recommender system on big data for tourists in Jeju," *Processes*, vol. 6, no. 8, p. 133, 2018.
14. J. Han and H. Lee, "Adaptive landmark recommendations for travel planning: Personalizing and clustering landmarks using geo-tagged social media," *Pervas. Mobile Comput.*, vol. 18, pp. 4_17, Apr. 2015.
15. S. Jiang, X. Qian, T. Mei, and Y. Fu, "Personalized travel sequence recommendation on multi-source big social media," *IEEE Trans. Big Data*, vol. 2, no. 1, pp. 43_56, Mar. 2016.