Automated Travel Itineraries using Hadoop

Pallavi S. Ghogare[1], Harmeet Khanuja[2]

[1]Department of Computer Engineering, MMCOE, Pune, India

[2]Department of Computer Engineering, MMCOE, Pune, India

Abstract: ***Nowadays travel and tourism activities are becoming increasingly needed to provide travelers with real time information to assist them in itinerary planning. In other words traveler need :i) Customized itinerary plans according to their needs and preferences ,ii)Timely availability and accuracy should be there. This system will provide decision support to user on the basis of: i) Access to real time and dynamically updated information, ii) Access to travel and tourism information on the move., iii)Provision of customized information. The problem of searching customized and cost effective itinerary is a team orienting problem (TOP), well known NP-Complete Problem. Here two stage processing system is designed for lowering cost. In first stage i.e. preprocessing stage which is offline, single day itineraries are precomputed via Map Reduce jobs. In second phase which is online, an approximate search is used to obtain multiple day itineraries by combining multiple single day itineraries. In this way a team orienteering problem is converted into Set-Packing another NP-Complete Problem.***

Keywords: Itinerary planning, team orienteering problem (TOP), Map reduce, set-packing

1. Introduction

a) Traveling market is divided into two parts. For casual customers, they will pick a package from local travel agents. The package, in fact, represents a pregenerated itinerary. The agency will help the customer book the hotels, ar- range the transportation, and preorder the tickets of muse- ums/parks. It prevents the customers from constructing their personalized itineraries, which is very time consuming and inefficient. Although the travel agencies provide efficient and convenient services, for experienced travelers, the itineraries provided by the travel agents lack customization and cannot satisfy individual requirements. Some interested POIs are missing in the itineraries and the packages are too expensive for a backpack traveler. Therefore, they have to plan their trips in every detail, such as selecting the hotels, picking POIs for visiting, and contacting the car rental service. Therefore, to attract more customers, travel agency should allow the users to customize their itineraries and still enjoy the same services as the predefined itineraries [1]

However, it is impossible to list all possible itineraries for users. A practical solution is to provide an automatic itinerary planning service. The user lists a set of interested POIs and specifies the time and money budget. The itinerary planning service returns top-K trip plans satisfying the requirements. In the ideal case, the user selects one of the returned itineraries as his plan and notifies the agent. There are three main objectives that the system must fulfill for its successful operation.

- Customer preferences and Requirement As many traditional systems provide predefined travel plans from that customers have to select any of these plans. If they have to visit any new POI they have to revisit some POI due to static travel plans. So by considering customer value system should be dynamic and input to the system should be customer's preferred POI's so that customer will able to plan their itinerary by himself. By using customer requirement and preferences system should produce alternate travel plans from which customer will select accordingly.

- Multiday Travel Itineraries
Many existing and traditional systems were working for single day itinerary planning but preparing multiday itineraries is very complex which requires to combine many single day itineraries to make multiday as a group of POIs into different days so that one POI can only appear once in the itinerary.

Although it is bit tricky to generate multiday itineraries from multiple single day itinerary. In this system algorithm with different approach from existing systems is used. Here high priority is given to selected POIs and generating trip plans on the fly.[1]

- Optimal and Cost Effective Itinerary
Many multiday itineraries will be generated by combining the single day itineraries. But final itinerary plan should be optimal and cost effective. Many predefined itineraries are costly so that travelers first have to manage money for the itinerary and then select the itinerary plan. In this system traveler can plan their itinerary according to their budget for the trip. Proposed plans are optimal and cost effective.

- Analysis And Recommendation
- Although the travel agencies provide efficient and convenient services, for experienced travelers, the itineraries provided by the travel agents lack customization and cannot satisfy individual requirements. Some interested POIs are missing in the itineraries and the packages are too expensive for a backpack traveler.

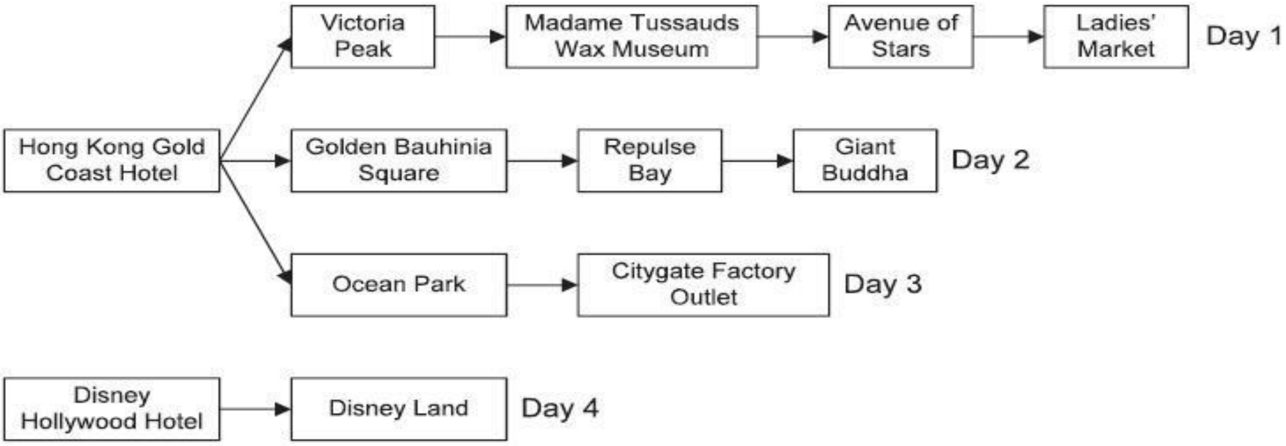

Figure 1: A four-day trip to Hong Kong

Therefore, they have to plan their trips in every detail, such as selecting the hotels, picking POIs for visiting, and contacting the car rental service.[1]

Tourism organizations starve for information that helps them understand: who their tourists are; what expectations and budget they have and what destinations, activities and services encounter they favors. For exploring the customer regarding knowledge agencies should analyze the behavior of the customers and more updates related to new POIs travelers want to visit. In this system by analyzing tour itineraries we can have monthly or weekly reports generating the details such as most visited POIs monthly or weekly, updated ranking of POIs after selected by traveler and many more.

From number of trips generated and selected by the travelers can be stored and if any new customer arrives with same list of POI list the we can recommend tour itineraries already completed successfully with the customer feedback . If the customer is satisfied with that plan then he will choose the same one. So that processing of same itinerary will be reduced and he will get solution in fraction of seconds. Otherwise he can ask for another plan then system will give all alternate plans to the customer to choose.

- Example Fig.1 shows a predefined four day itinerary for Hong Kong provided by Singapore agency. It covers the most popular POIs for the first time traveler and customer just need to follow the schedule of their trips. Although the travel agencies provide efficient and convenient services, for experienced travelers, the itineraries provided by the travel agents lack customization and cannot satisfy individual requirements. Some interested POIs are missing in the itineraries and the packages are too expensive for a backpack traveler. Therefore, they have to plan their trips in every detail, such as selecting the hotels, picking POIs for visiting, and contacting the car rental service.

A. Paper Organization

This paper is organized as follows: Literature Survey is discussed in Section 2, Section 3 introduces system architecture, in this approach MapReduce hadoop is used as parallel processing engine. Section 4 gives conclusions and directions for future work.

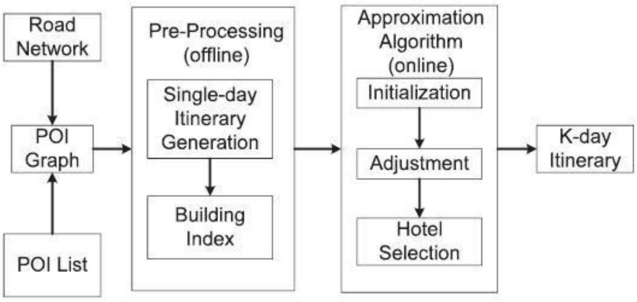

Figure 2: Architecture of a trip planning system

2. Literature Survey

In [2] an interactive and user friendly travel planning system is proposed. It basically works on Point-Of-Interest (POI) feedback model constructed through feedback of the users who completed their itinerary. The system will recommend best itinerary plan on the basis of feedback of the POI. In this paper algorithms used are able to generate single-day itineraries only. And we needed to collect the comments on each new POIs from the users which is very time consuming. In this system some heuristic approximations are used and are based on some assumptions. So that they only provide limited number of itineraries which are not optimal for bag pack traveler.

In [3] photo-streams are used to generate meta-data about the any Point-Of-Interest(POI).All photo-streams by individual user is collected and processed. Then all Photo-streams by all users are combined to form a network graph of POIs. From these graphs automated travel itineraries are generated. In this approach first data mining algorithms are used to retrieve the information and again further processed for itinerary planning. In [4] subgraph analysis is done using Hadoop. Here MapRe- duce framework is used to reduce cost of processing NP- complete problem. But in this approach Team Orinteering problem is used to solve directly which will not ensure about the optimal results. In this paper TOP is converted into weighted Set-Pack problem which further uses initialization-adjustment model to get optimal resulted itineraries.

3. System Overview

To reduce the processing cost here two-stage planning scheme is used. In its preprocessing stage single-day itineraries are precomputed via MapReduce jobs. In its online stage an approximate search algorithm is used to combine single-day itineraries. Fig 2 shows system architecture.

A. Preprocessing

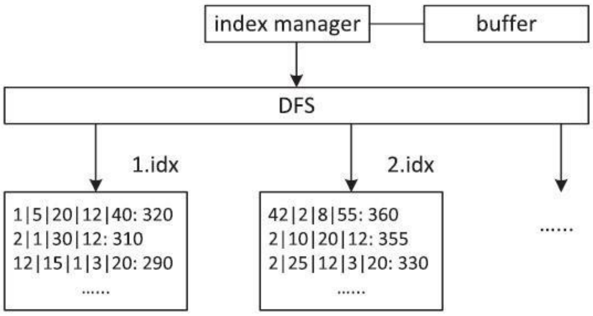

Figure 3: Itinerary Index

In the preprocessing, POIs are organized into an undirected graph, G. The distance of two POIs is evaluated by Google Maps APIs [1].In the preprocessing stage; we iterate all candidate single-day itineraries using a parallel processing framework, Map Reduce [6]. The results are maintained in the distributed file system (DFS) and an inverted index is Built for efficient itinerary retrieval. To construct a multiday itinerary, we need to selectively combine the single itineraries. The preprocessing stage, in fact, transforms the TOP into a set-packing problem [5], which has well-known approximated algorithms.

- Single-Day Itinerary
  Using MapReduce iterations Single-day itineraries are computed. The mappers load the partial paths from the DFS, which are generated in the previous MapReduce jobs. Then try to append new POI to the existing itineraries. For each new path, it tests whether it can be completed within one day. If not, will discard the new path.

- Itinerary Index
  To efficiently locate the single-day itineraries, an inverted index is built. The key is the POI and the values are all itineraries involving the POI. By scanning the index, we can retrieve all the itineraries. Fig. 4 illustrates the index structure. We create an index file for each POI in the DFS. The file includes all single itineraries involving the POI, which are sorted based on their weights. For example, in Fig.3 "1.idx" contains all itineraries for the first POI. The itinerary "1j5j20j12j40"is the most important itinerary in the index file with weight 320. The inverted index is constructed via a MapReduce job. The mappers load the single-day itinerary and generate key-value pairs for each involved POI. The reducers collect all itineraries for a specific POI and sort them based on the weights before creating the index file. In our system, the size of the index file may vary a lot. Some POI may have an extremely large index file, due to its popularity and short visit time. In reducers, those POIs may result in the exception of memory overflow in the sorting process. To address this problem, in the map phase, instead of using the POI as the key, we generate the composite key by combining the POI and the itinerary weight [1].

Advantages of MapReduce

1) Parallel computing effectively reduces the running time of preprocessing. The search space explodes, when the number of POIs and traveling days increases. It is impractical to generate all possible itineraries. But by exploiting the power of Map Reduce, we can share and balance the workload between multiple machines.
2) MapReduce algorithms can remove the duplicated itineraries in a simple way .the framework of Map Reduce, map all the itineraries with the same POIs into the same reducer and only keep one itinerary with the lowest cost.

B. Greedy-Based Approximation Algorithm

After the itinerary indexes are constructed, the user request can be processed by selecting k best itineraries from the indexes. Namely, the problem of generating optimal k-day itinerary is transformed into a weighted set-packing problem. [1] There are three phases in this algorithm.

- Initialization
  The initialization phase applies the greedy-based heuristic approach to generate a k-itinerary as the seed, which is further improved in the adjustment phase by replacing the itineraries with their independent sets [1]. First sort the selected POIs by their weights. Then, in each iteration, we try to form a group, which contains a subset of POIs that can be accessed within one day. Then greedily select the POI with shortest distance and add it into a group. There are maximally k groups generated. All groups are used as our seeds for searching the index. We will use the First itinerary that contains all the POIs in the group as our candidate itinerary. Although after the weight adjustment, itineraries in the index file are no longer sorted by the weights. To improve the weights of the obtained itineraries in the greedy algorithm, we adopt the adjustment phase.

- Adjustment
  In the adjustment phase, new solutions are searched and used to replace the greedy itineraries. The process repeats until no improvement can be obtained. The adjustment phase greatly increases the processing cost. In the adjustment phase, the query engine loads the itinerary index from the DFS, which incurs high I/O cost. One way to reduce the cost is to increase the index buffer size. After an indexed itinerary is loaded from the DFS, we cache it in the buffer. If the buffer is full, we apply the LRU strategy to remove the less used entries.[1]

- Hotel Selection
- In fact, hotels can be considered as a special type of POIs. It must appear as the last POI in the itinerary. We need to calculate the traveling time from other POIs to the hotel POIs. Hotel POIs do not incur access cost and their weights are set as user's rankings for the hotels.

Based on the users preference, we have two processing strategies.[1]

1)Single Hotel
If the user prefers to stay in the same hotel, the itinerary generation problem cannot be easily reduced to the set-packing problem. Instead, we adopt a best-effort solution. In particular, find the candidate k-day itinerary without hotel POIs. After that, we append the hotel POI.

2)Multiple Hotel
If the user does not insist on staying in the same hotel (e.g., he can select k different hotels, one for each day), we can extend the preprocessing algorithm to handle the hotels. In the MapReduce jobs, when a new itinerary is generated, we test every hotel POI and try to append it to the end of single-day itinerary.

4. Conclusion and Future Work

In this paper an automated itineraries are generated as per the traveler’s selected list of Point-Of-Interests (POI).Which gives traveler a customized multiday travel plans. This problem of generating optimal itineraries is NO-complete problem, which has no polynomial time approximate algorithm. For efficient travel itineraries here two-stage processing is used. In first stage Map Reduce framework is used to generate indexed single-day itineraries. Parallel processing engine allows to iterate through whole dataset and index as many as itineraries as possible.

After preprocessing stage Team Orienteering Problem is converted into weighted Set-Pack Problem. In this stage by using Greddy-based Approximation algorithm single-day itineraries are combined to produce multiday itinerary. Here Initialization-Adjustment Model is used.

For the future scope we can have feedback from travelers using which we have the analysis and system. Also dataset used is limited to some geo-locations it can be broader may be global.

References

[1] Gang Chen, Sai Wu, Jingbo Zhou, and Anthony K.H. Tung, "Au- tomatic Itinerary Planning for Traveling Services" TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 3, MARCH 2014.

[2] S.B. Roy, G. Das, S. Amer-Yahia, and C. Yu, "Interactive Itinerary Planning", Proc. IEEE 27th Int’l Conf. Data Eng. (ICDE), pp. 15-26, 2011.

[3] M.D. Choudhury, M. Feldman, S. Amer-Yahia, N. Golbandi, R.Lempel, and C. Yu,"Automatic Construction of Travel Itineraries Using Social Breadcrumbs",Proc. 21st ACM Conf. Hypertext and Hypermedia (HT), pp. 35-44, 2010.

[4] Z. Zhao, G. Wang, A.R. Butt, M. Khan, V.A. Kumar, and M.V.Marathe,"SAHAD: Subgraph Analysis in Massive Networks Using Hadoop", IEEE Int’l Parallel and Distributed Processing Symp.(IPDPS), 2012

[5] V.S.P. de Aragao, H. Viana, and E. Uchoa, "The Team Orienteering Problem: Formulations and Branch-Cut and Price",Proc. Algorithmic Approaches for Transportation Modeling, Optimization,and Systems (ATMOS), vol. 14, pp. 142-155, 2010.

[6] J. Dean and S. Ghemawat, "MapReduce: A Flexible Data Processing Tool",Comm. ACM, vol. 53, pp. 72-77, Jan. 2010.

[7] http://hadoop.apache.org/,2013.

[8] P. Vansteenwegen, W. Soufiriau, and D.V. Oudheusden, "The Orienteer- ing Problem: A Survey", European J. Operational Research, vol. 209, pp. 1-10, Feb. 2010.