A Genetic Algorithm for Travel Itinerary Recommendation with Mandatory Points-of-Interest

Phatpicha Yochum, Liang Chang(✉), Tianlong Gu, Manli Zhu, and Hongliang Chen

Guangxi Key Laboratory of Trusted Software,
Guilin University of Electronic Technology, Guilin 541004, China
changl@guet.edu.cn

Abstract. Traveling as a very popular leisure activity enjoyed by many people all over the world. Typically, people would visit the POIs that are popular or special in a city and also have desired starting POIs (e.g., POIs that are close to their hotels) and destination POIs (e.g., POIs that are near train stations or airports). However, travelers often have limited travel time and are also unfamiliar with the wide range of Points-of-Interest (POIs) in a city, so that the itinerary planning is time-consuming and challenging. In this paper, we view this kind of itinerary planning as MandatoryTour problem, which is tourists have to construct an itinerary comprising a series of POIs of a city and including as many popular or special POIs as possible within their travel time budget. We term the most popular and special POIs as mandatory POIs in our paper. For solving the presented MandatoryTour problem, we propose a genetic algorithm GAM. We compare our approach against several baselines GA, MaxM, and GreedyM by using real-world datasets from the Yahoo Flickr Creative Commons 100 Million Dataset (YFCC100M), which include POI visits of seven touristic cities. The experimental results show that GAM achieves better recommendation performance in terms of the mandatory POIs, POIs visited, time budget (travel time and visit duration), and profit (POI popularity).

Keywords: Travel recommendations · Itinerary recommendations · Recommendation systems · Location recommendations

1 Introduction

Traveling is entirely considered as a pleasure activity and it does not take long for travelers to travel. People have high travel expectations and enjoy their journeys and experiences. Travel itinerary or tour planning is one of the most important tasks for people to travel to unfamiliar cities and places. It mainly focuses on a plan with a sequence of visits of a given number of Points-of-Interest (POIs),

Published by Springer Nature Switzerland AG 2020
Z. Shi et al. (Eds.): IIP 2020, IFIP AICT 581, pp. 133–145, 2020.
https://doi.org/10.1007/978-3-030-46931-3_13

which must be visited within a limited time. Additional information is also considered such as the number of POIs visited, travel time, and POI visit duration of the trip. Especially, in order to maximize the number of POIs and/or the visit duration, the travel time or the distance between POIs should be cut down. It is intuitive to include mandatory POIs, which are everywhere on the user's trip as they are often very popular or special POIs where tourists should visit in a city.

The existing works adapt a simple measure based on user interest and POI popularity for itinerary recommendations. In our work, we view this kind of itinerary planning as MandatoryTour problem, which is tourists have to construct an itinerary comprising a series of POIs of a city and including as many popular or special POIs as possible within their travel time budget. Hence, mandatory POIs are the term of the most popular and special POIs. We propose a travel itinerary recommendation approach, named GAM, to solve the MandatoryTour problem by using a genetic algorithm. Besides, we use real-world datasets which are derived from the Yahoo Flickr Creative Commons 100 Million Dataset (YFCC100M) provided by [1] to evaluate our approach. We make contributions to the field of itinerary recommendation as follows:

1. We introduce and formulate the MandatoryTour problem, which is the term the most popular and special POIs.
2. We propose the GAM algorithm for recommending an itinerary comprising a series of POIs of a city and including as many mandatory POIs as possible within travel time budget.
3. The results show that GAM outperforms better than several baseline methods and achieves good recommendation performance in terms of the mandatory POIs, POIs visited, time budget (travel time and visit duration), and profit (POI popularity).

The remainder of this paper is organized as follows. Section 2 introduces related work. Section 3 describes the problem definition and genetic algorithm model. In Sect. 4, our experiments are presented. The results and discussion are discussing in Sect. 5. The last section summarizes conclusions.

2 Related Work

In this section, we present state-of-the-art methods in related areas of genetic algorithm, itinerary recommendation, POI recommendation, and the differences in our research with existing works.

2.1 Genetic Algorithm in Tourism

Genetic Algorithm (GA) has its origins from the imitations of natural evolution and genetics. It uses multistage processing, such as initialization, selection, crossover, and mutation to optimize the solution. In recent years, researchers have been applied the GA to recommendations in the tourism area [2–5]. The main objectives of these works are to find the optimal travel route comprising a set of POIs, while the GA uses the fitness function to select the best route. More formally, [3] studied the problem of user preferences to recommend a travel route. They estimated popular POIs where user has visited in the past by mining from the GPS trajectories. Then, the GA was used to model the interest of user for an unvisited place and improved the accuracy of the recommendation.

2.2 Itinerary Recommendation

Itinerary recommendation is a well-studied field that typically focuses on suggesting a sequence of POIs to visit. Most existing studies on itinerary recommendations focus on user interests within the given trip constraints towards the POIs [6,7]. Several research works apply itinerary recommendations in the field of Operations Research [8–10]. Most of these works are formulated as an Integer Linear Program based on the Orienteering problem and the traveling salesman problem variants using social media datasets. [9] studied the travel recommendation problem based on the Orienteering problem by proposing the PersTour model. First, user travel histories based on geo-tagged photos were extracted. Next, they used the first and last photos taken at each POI to sort POI visiting time and construct user travel sequence. At last, the PersTour algorithm with the characteristic of POIs, users' interest preferences and trip constraints were used to recommend personalized trip itinerary to users. There are also several real-life constraints like POI popularity [11], visit duration [12], travel time [13], queuing time [8], and photo frequency [9] to recommend itinerary recommendation systems.

2.3 POI Recommendation

In POI recommendation, the problem is to provide users with the suggestion of a set of popular and interesting places to users. The common recommendation techniques have been extensively studied such as content-based [14,15], collaborative [16,17], and hybrid approach [18,19]. One example in [18], the hybrid recommendation was proposed by merging content-based, collaborative, and knowledge-based techniques into a recommendation process for travel destinations to individuals and groups. The algorithm was based on the users' ratings, personal interests, and specific demands for the next destination.

2.4 Differences with Existing Works

We focus on interesting insights into the itinerary recommendation problems. Our proposed approach differs from the above existing works in several aspects. The current state-of-the-art itinerary recommendation approaches consider POIs with various trip constraints. These approaches do not consider mandatory POIs which are the actual place covering attractions, buildings, shopping malls, universities, transports, etc. In contrast, we propose an enhance itinerary recommendation system that considers mandatory POIs through these POIs with a specific starting and ending POI and additional constraints. So, the itinerary planning can be composed of a series of POIs including mandatory POIs within a certain time. We improve an itinerary recommendation by considering several aspects with the GA method to achieve better performance.

3 Problem Definition and Genetic Algorithm Model

In this section, we give the definitions used in our work and formulate the MandatoryTour problem, and a genetic algorithm is presented for dealing with this problem.

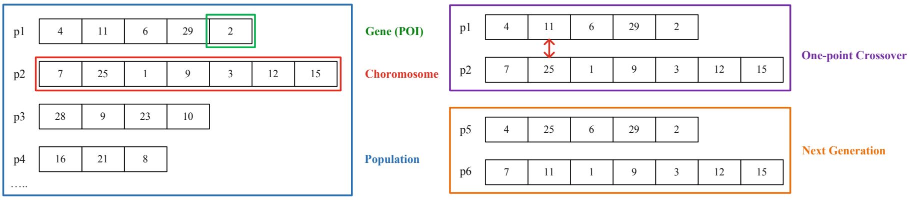

Fig. 1. An example of POI code and one-point crossover

3.1 Problem Definition

Our proposed MandatoryTour problem which recommend an itinerary with mandatory POIs is NP-hard. A shortcoming of traditional methods that use a brute-force approach is that the complexity of MandatoryTour is exponential, which is caused by the increasing of the number of POIs. The objective of this problem is to maximize the number of mandatory POIs, while keeping travel time between POIs and visit duration under a fixed time budget. The recommended itinerary includes a specified starting POI and ending POI.

This problem can be viewed as a directed graph $G = <N, E>$, where N is the set of nodes (or POIs) and E represents the set of edges. Each edge connecting node i to j has a profit, a travel time, and a visit duration, and can be represented as $f_{i,j}$, $t_{i,j}$, and $v_{i,j}$ respectively. The total time cost that includes travel time and visit duration between visited POIs for a tour is no more than the time budget T_{MAX} which limits how many POIs can be visited on the tour.

In this paper, an itinerary is defined as a path between specified starting POI and ending POI, and at least one other POI is contained. Note that all POIs in the itinerary can be visited only once, so sub-tours are excluded. Let $C = \{c_1, ..., c_L\}$ be the set of POIs, and $M = \{m_1, ..., m_K\}$ where $K < L$ be the set of mandatory POIs, ideally an itinerary with mandatory POIs can be described as $I = \{c_s, ..., m_1, ..., m_K, ...c_d\}$, where c_s is the starting POI and c_d is the destination POI and $c_s, c_d \notin M$.

Algorithm 1. GAM algorithm

```
Input: time budget $\sigma$,
       population size $\alpha$,
       crossover rate $\beta$,
       mutation rate $\gamma$,
       iteration number $\delta$
Output: best tour $t$
generate $\alpha$ individuals randomly as initially population set $P$
    TotalTimeCost($p$) $\leq \sigma$ for each $p \in P$
for $i = 1$ to $\delta$ do
    for $j = 1$ to $\alpha$ do
        randomly select two individuals $p_a$ and $p_b$ from $P$
        generate $p_c$ and $p_d$ by one-point crossover to $p_a$ and $p_b$ under the crossover rate $\beta$
        save $p_c$ and $p_d$ to $P_1$
    end for
    for $j = 1$ to $\alpha$ do
        randomly select an individuals $p_a$ from $P_1$
        randomly select a gene position from $p_a$ and mutate it to generate a new individual $p_a^{'}$ under the mutation rate $\gamma$
        if TotalTimeCost($p_a^{'}$)> $\sigma$ then
            update $p_a^{'}$ with a feasible tour by repairing $p_a^{'}$
        end if
        update $p_a$ with $p_a^{'}$ in $P_1$
    end for
    update $P = P_1$
end for
return the best tour $t$ in $P$
```

3.2 Genetic Algorithm Model

In our genetic algorithm, P is the set of population, the set of genes of each individual p_i is represented using POIs directly, in terms of the IDs of POIs, and we encode them as shown in the left part of Fig. 1. An example of one-point crossover process is simply stated in the right part of Fig. 1.

In fact, MandatoryTour problem is a multi-objective optimize problem, and as it is difficult to design a fixed fitness score for every tour. Therefore, we optimize the objective function directly instead. By giving different priorities to the metrics used in Sect. 4, the objective function is defined as follows:

if the number of mandatory POIs in p_i has not been maximized

$$MaxF(p_x) = Max \sum\nolimits_{i=1}^{|p_x|} \chi_{M(p_{x_i})}$$

where

$$\chi_{M(p_{x_i})} = \begin{cases} 1 & \text{if } p_{x_i} \in p_x \\ 0 & \text{otherwise} \end{cases}$$

if the above one has been maximized and the visit duration of p_i has not been maximized

$$MaxF(p_x) = Max \sum_{i=1}^{|p_x|-1} \sum_{j=2}^{|p_x|} v_{p_{x_i}, p_{x_j}}$$

if the above two have been maximized and the total profit of p_i has not been maximized

$$MaxF(p_x) = Max \sum_{i=1}^{|p_x|-1} \sum_{j=2}^{|p_x|} f_{p_{x_i}, p_{x_j}}$$

We use the one-point crossover in our proposed approach because of the relatively small data sets, and the probability $\beta = 0.8$. For other parameters, mutation rate $\gamma = 0.2$, time budget $\sigma \in \{300, 250, 600, 450, 350\}$ as it is different for each city, the population size $\alpha = 60$, the iteration number $\delta = 100$, and finally the best tour t is returned. The detailed optimization procedure is described in Algorithm 1.

4 Experiments

In this section, we describe our experiments, which include our datasets, baseline algorithms, evaluation metrics, and results and discussion.

4.1 Datasets

For our experiment and analysis, we use datasets from the Yahoo! Flickr Creative Commons 100M [1], which contains 100 million photos and videos. POIs with other details were collected from [10]. These geo-tagged photos were then mapped to a list of POIs based on their respective entries on cities in which the details refer to [9]. There are seven cities: Budapest, Edinburgh, Toronto, Vienna, Glasgow, Perth, and Osaka.

4.2 Baseline Algorithms

We compare our proposed GAM with several baseline algorithm to evaluate its recommendation performance.

1. **GA.** Generates an itinerary without mandatory POIs. The generated itinerary comprises a path starting at a specified POI and ending at another specified one where the total profit and visit duration are maximized, the cost is minimized, and the total travel time is limited by a given time budget. Note that mandatory POIs may be included in the itinerary, and we will show its result in the next section.

2. **GAM (our proposed model).** Generates an itinerary with mandatory POIs. This model is built upon the GA model using a similar objective function but adds mandatory POIs and the objective of maximizing the inclusion of the mandatory POIs. This model considers a general tour which generally includes popular or special POIs where tourists often want to visit.
3. **MaxM.** Generates an itinerary with a relatively large profit with mandatory POIs. Mandatory POIs are added first then the other POIs by allocating a large profit value to each POI using the greedy strategy. This approach provides a profit baseline for the Mandatory problem.
4. **GreedyM.** Generates an itinerary by adding the mandatory POIs first, then the remaining POIs. This is the simplest practical method to generate itineraries based on visiting mandatory POIs. As the tour focus on the mandatory POIs, a tour that has the most mandatory POIs within the time budget is preferred.

The algorithms used for this work were implemented using the C++ programming language.

4.3 Evaluation Metrics

We evaluate the performance of our algorithm and the baselines, which involves evaluating a specific starting and ending POI and additional constraints. The recommended itinerary contains the set of mandatory POIs or at least one of the mandatory POIs within a certain time based on travel cost budget and profit. Our algorithm utilizes evaluation metrics for the itinerary recommendation as follows:

1. **Mandatory POIs.** The set of mandatory POIs that are popular or special POIs in the recommended itinerary.
2. **POIs Visited.** The number of unique POIs that can be visited in the recommended itinerary.
3. **Time Budget.** The total time budgets both travel time and visit duration in the recommended itinerary. Hence, travel time is the time traveled from one POI to another POI while visit duration is the time visited in each POI.
4. **Profit.** The total profits of all POIs in the recommended itinerary.

4.4 Results and Discussion

In this section, we present and discuss the experimental results in term of mandatory POIs, POIs visited, time budget (travel time and visit duration), and profit (POI popularity). In addition, we considered four mandatory POI sets including one POI, two POIs, three POIs, and four POIs respectively, and they are randomly selected from the whole POI set.

Number of Mandatory POIs of Recommended Tours. The GA algorithm without mandatory POIs is the basis for comparisons with other algorithms, and

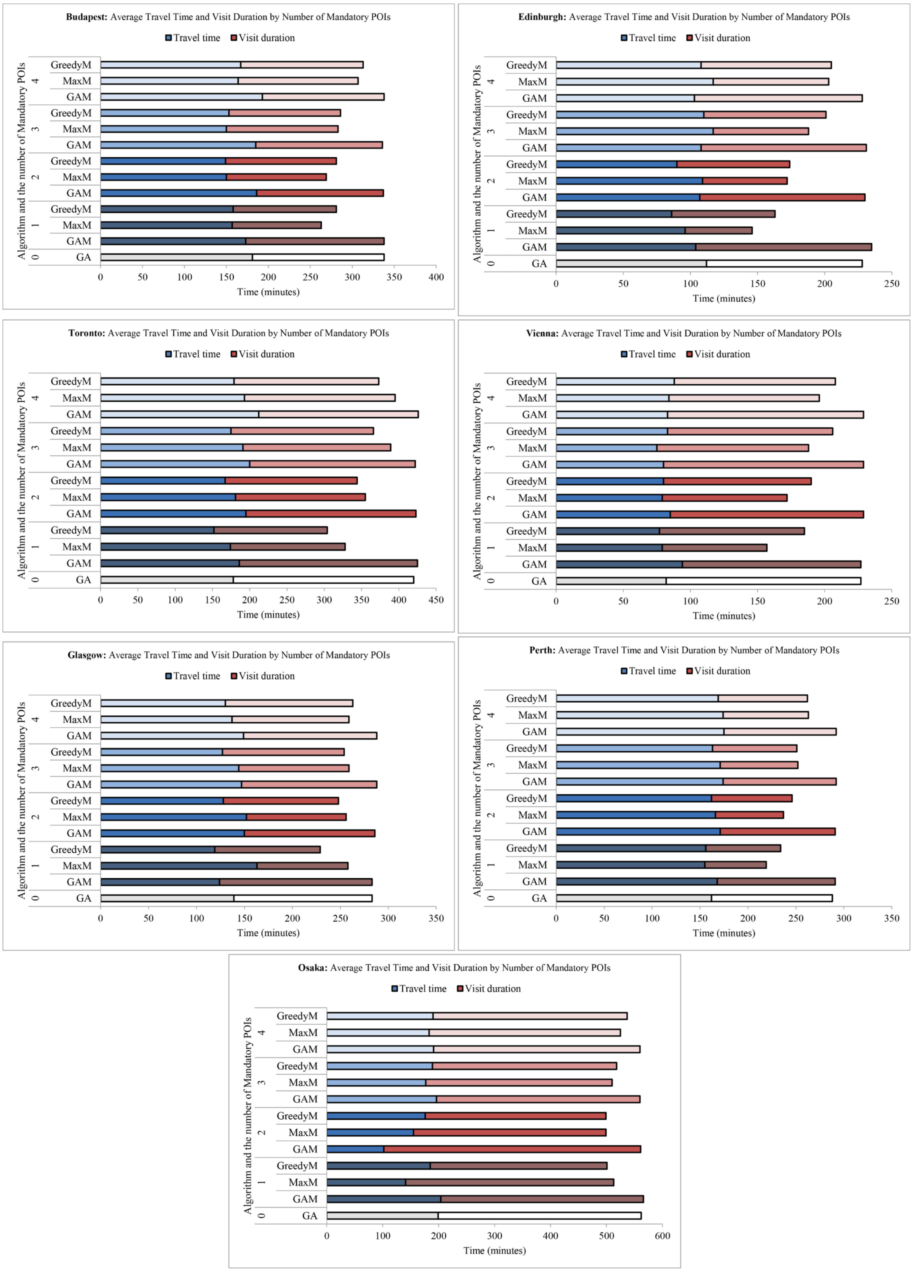

Fig. 2. Average travel time and visit duration by number of mandatory POIs for each city.

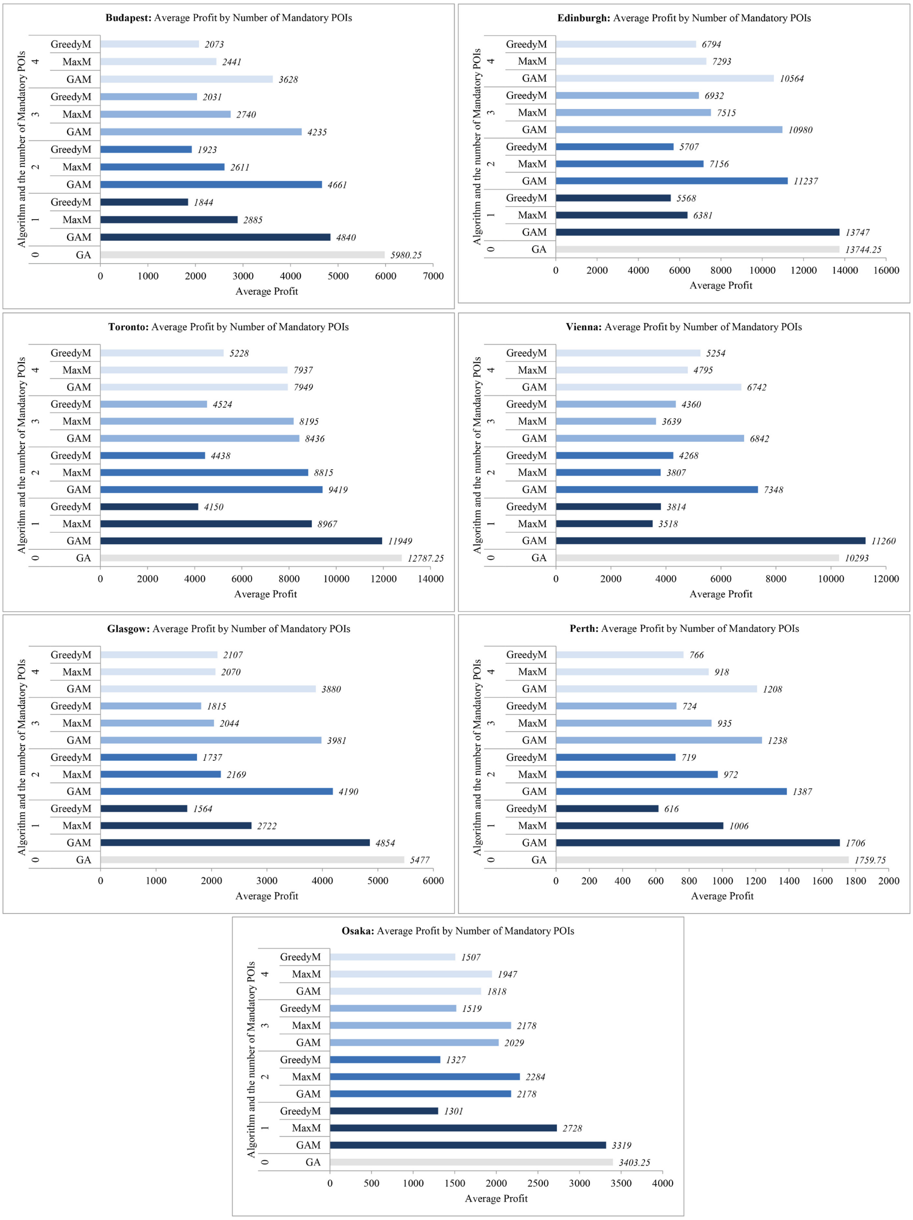

Fig. 3. Average profit by number of mandatory POIs for each city.

Table 1. Number of GA itineraries (out of 100) which visited all or at least one mandatory POIs. Higher values are better.

City	Visited all mandatory POIs				Visited at least one mandatory POIs			
	1	2	3	4	1	2	3	4
Budapest	28	13	1	1	28	65	77	82
Edinburgh	53	22	8	3	53	69	78	86
Toronto	31	18	7	1	31	67	73	88
Vienna	34	5	5	1	34	53	68	77
Glasgow	43	22	12	5	43	68	82	89
Perth	50	20	8	5	50	67	81	88
Osaka	48	15	5	1	48	65	78	88

Table 2. Number of successful itineraries (out of 100) which included mandatory POIs. Higher values are better and the best performance among GAM, MaxM, GreedyM is in bold.

City	GAM mandatory POIs				MaxM mandatory POIs				GreedyM mandatory POIs			
	1	2	3	4	1	2	3	4	1	2	3	4
Budapest	**89**	**81**	**49**	**34**	**89**	78	45	27	**89**	78	45	27
Edinburgh	**87**	50	31	15	**87**	**75**	**58**	**36**	**87**	**75**	**58**	**36**
Toronto	**77**	**54**	**36**	**14**	**77**	50	29	7	**77**	50	29	7
Vienna	**87**	56	23	10	**87**	**65**	**39**	**18**	**87**	**65**	**39**	**18**
Glasgow	**81**	**66**	**42**	**22**	**81**	60	37	16	**81**	60	37	16
Perth	**86**	63	46	21	**86**	**67**	**47**	**23**	**86**	**67**	**47**	**23**
Osaka	**99**	86	64	31	**99**	**88**	**69**	**49**	**99**	**88**	**69**	**49**

visiting any of the mandatory POIs in a generated tour is not guaranteed. The result of the inclusion of mandatory POIs of generated tours by the GA algorithm is shown in Table 1. It can be seen that with the increase of mandatory set size, the number of tours successfully visited all mandatory POIs in the mandatory sets is decreased rapidly, however, the number of tours visiting at least one mandatory POI move up as more options are available.

Table 2 presents the number of successful tours found by GAM, MaxM, and GreedyM, along with the different mandatory POI sets. The GAM algorithm achieves moderate performance in all cities.

Number of Total POIs Visited of Recommended Tours. Table 3 presents the average number of total POIs visited every algorithm's generated tours. Overall, GAM has the best performance among all algorithms over all the cities. Specifically, for the GAM algorithm, the mandatory POI set of four including

Table 3. Average number of POIs visited including failed itineraries by algorithm, mandatory POIs set size and city. Higher values are better and the best performance among GAM, MaxM, GreedyM is in bold.

City	GA	GAM mandatory POIs				MaxM mandatory POIs				GreedyM mandatory POIs			
		1	2	3	4	1	2	3	4	1	2	3	4
Budapest	12.82	**10.3**	**10.1**	**9.4**	**8.4**	4.3	4.6	5.1	5.3	4.6	4.8	5.2	5.4
Edinburgh	12.57	**12.7**	**10.4**	**10.2**	**9.8**	3.8	4.5	5.1	5.5	5.3	5.6	5.9	6.2
Toronto	12.02	**9.8**	**10.1**	**9.2**	**8.4**	4.5	4.9	5.2	5.3	4.1	4.3	4.6	4.9
Vienna	10.62	**10.4**	**7.9**	**7.6**	**7.2**	3.1	3.8	4.3	4.7	4.2	4.7	5.0	5.2
Glasgow	12.42	**10.4**	**10.0**	**9.5**	**9.1**	4.3	4.3	4.6	4.8	4.3	4.6	4.8	5.1
Perth	10.82	**10.2**	**8.9**	**8.3**	**8.2**	4.5	4.8	5.1	5.4	4.6	4.9	5.1	5.5
Osaka	10.75	**11.2**	**7.8**	**7.3**	**7.0**	5.2	5.3	5.6	5.9	5.7	5.5	5.8	6.0

more POIs than smaller mandatory sets, but for MaxM and GreedyM, they see a decline along with the increase of the mandatory POI set size. This is reasonable because of the mandatory POIs may limit the MaxM and Greedy algorithms' performance in the metric of average number POIs visited.

Travel Time and Visit Duration of Recommended Tours. From Fig. 2, we can see that the total travel time for the itineraries generated by all methods was never exceeded, and the GAM uses the time budget efficiently in all seven cities, in the meanwhile, the visit duration also utilized better, and GA has the similar results with GAM. In contrast, the GreedyM often comes the last about allocating the time budget, and MaxM's performance just above GreedyM.

Total Profit of Recommended Tours. The results of the average profit of recommended tours are shown in Fig. 3. It is clear that the GA algorithm gets the highest profit in almost every city as there are no extra constraints except the time budget, and the GAM comes the second. In addition, the reason why the MaxM algorithm does not achieve the best performance is that the greedy strategy limits it to generate a tour with the optimal solution.

5 Conclusions

In this paper, we formulated the kind of itinerary planning as MandatoryTour problem, which comprised a series of POIs of a city and including mandatory POIs within travel time budget. The mandatory POIs can be termed as the most popular and special POIs. We then solved this MandatoryTour problem by a generic algorithm and we proposed this approach as the GAM algorithm. We also used real-world datasets which are derived from the Yahoo Flickr Creative Commons 100 Million Dataset (YFCC100M), which include POI visits of seven touristic cities. Compared with several baselines GA, MaxM, and GreedyM, GAM achieved better recommendation performance in terms of the mandatory POIs, POIs visited, time budget (travel time and visit duration), and profit (POI popularity).

Acknowledgements. This work was supported in part by the Natural Science Foundation of China (Nos. U1711263, U1811264, 61966009), and in part by the Natural Science Foundation of Guangxi Province (No. 2018GXNSFDA281045).

References

1. Thomee, B., et al.: YFCC100M: the new data in multimedia research. Commun. ACM **59**(2), 64–73 (2016)
2. Hang, L., Kang, S., Jin, W., Kim, D.: Design and implementation of an optimal travel route recommender system on big data for tourists in Jeju. Processes **6**(8), 133 (2018)
3. Ke, C., Wu, M., Ho, W., Lai, S., Huang, L.: Intelligent point-of-interest recommendation for tourism planning via density based clustering and genetic algorithm. In: PACIS (2018)
4. Sun, X., Wang, J., Wu, W., Liu, W.: Genetic algorithm for optimizing routing design and fleet allocation of freeway service overlapping patrol. Sustainability **10**(11), 4120 (2018)
5. Tiwari, S., Kaushik, S.: Modeling personalized recommendations of unvisited tourist places using genetic algorithms. In: Chu, W., Kikuchi, S., Bhalla, S. (eds.) DNIS 2015. LNCS, vol. 8999, pp. 264–276. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16313-0_20
6. Corsar, D., Edwards, P., Nelson, J., Baillie, C., Papangelis, K., Velaga, N.: Linking open data and the crowd for real-time passenger information. J. Web Semant. **43**, 18–24 (2017)
7. Sylejmani, K., Dorn, J., Musliu, N.: Planning the trip itinerary for tourist groups. Inf. Technol. Tourism **17**(3), 275–314 (2017). https://doi.org/10.1007/s40558-017-0080-9
8. Lim, K., Chan, J., Karunasekera, S., Leckie, C.: Personalized itinerary recommendation with queuing time awareness. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. Association for Computing Machinery, pp. 325–334 (2017)
9. Lim, K.H., Chan, J., Leckie, C., Karunasekera, S.: Personalized trip recommendation for tourists based on user interests, points of interest visit durations and visit recency. Knowl. Inf. Syst. **54**(2), 375–406 (2017). https://doi.org/10.1007/s10115-017-1056-y
10. Taylor, K., Lim, K., Chan, J.: Travel itinerary recommendations with must-see points-of-interest. In: Companion Proceedings of the The Web Conference 2018. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, pp. 1198–1205 (2019)
11. Wang, X., Leckie, C., Chan, J., Lim, K., Vaithianathan, T.: Improving personalized trip recommendation by avoiding crowds. In: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. Association for Computing Machinery, pp. 25–34 (2016)
12. Cai, G., Lee, K., Lee, I.: Itinerary recommender system with semantic trajectory pattern mining from geo-tagged photos. Expert Syst. Appl. **94**, 32–40 (2018)

13. Zhang, Y., Tang, J.: Itinerary planning with time budget for risk-averse travelers. Eur. J. Oper. Res. **267**(1), 288–303 (2018)
14. Binucci, C., Luca, F., Giacomo, E., Liotta, G., Montecchiani, F.: Designing the content analyzer of a travel recommender system. Expert Syst. Appl. **87**, 199–208 (2017)
15. Laß, C., Herzog, D., Wörndl, W.: Context-aware tourist trip recommendations. In: RecTour@RecSys (2017)
16. Wang, H., Wang, N., Yeung, D.: Collaborative deep learning for recommender systems. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1235–1244 (2015)
17. Yang, C., Bai, L., Zhang, C., Yuan, Q., Han, J.: Bridging collaborative filtering and semi-supervised learning: a neural approach for POI recommendation. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1245–1254 (2017)
18. Al-hassan, M., Lu, H., Lu, J.: A semantic enhanced hybrid recommendation approach: a case study of e-government tourism service recommendation system. Decis. Support Syst. **72**, 97–109 (2015)
19. Kefalas, P., Symeonidis, P., Manolopoulos, Y.: Recommendations based on a heterogeneous spatio-temporal social network. World Wide Web **21**(2), 345–371 (2017). https://doi.org/10.1007/s11280-017-0454-0