A Socially Motivating and Environmentally Friendly Tour Recommendation Framework for Tourist Groups

Mehdi Kargar[1], Zhibin Lin[2,#]

[1] *Ted Rogers School of Management, Ryerson University, Toronto, Ontario, Canada*

[2] *Durham University Business School, Durham University, Durham, United Kingdom*

kargar@ryerson.ca, zhibin.lin@durham.ac.uk

[#] *corresponding author*

Abstract

Traveling in a group brings various social and environmental benefits, yet members might have different (and sometime conflicting) preferences. In this study, a tour recommendation framework is proposed that receives a set of must-visit and preferred points of interest from each tourist and forms multi-day tours that cover all must-visit points. Furthermore, the framework attempts to maximize fairness among group members. This ensures all members are motivated to participate in the group tour. While generating the itinerary, the framework also guarantees that a threshold on the commuted distance, time, and monetary budget is met on each day. The benefits of this approach are the maximization of social wellbeing and minimization of energy consumption. The advantages of the proposed framework are confirmed via a test using a Foursquare dataset of two major cities of New York and Tokyo and a user study.

Keywords: Analytical Recommendation, Tour Recommendation, Multi-day Tour, Sustainable Tourism, Social Tourism

1. Introduction

Traveling is an important activity for over one billion tourists that travel each year (UNWTO, 2016). In recent years, there is an increase in the number of tourists that travel in small groups of family members or close friends, rather than travelling with an agency (Tsaur et al., 2010). This is particularly the case because of the Covid-19 pandemic. These groups of tourists usually do not follow a standard tour package, and rely on making their own itineraries. With the advent of the world wide web, these groups of tourists have easy access to many resources (e.g., TripAdvisor) to find a variety of points of interest (POIs) (Yuan et al., 2016). After finding the list of potential POIs, the next step is to form a trip itinerary that comprises the POIs over a number of days.

The task of creating an itinerary is complex as it should satisfy a number of constraints besides choosing the POIs and deciding the sequence of visits (Kotiloglu et al., 2017). These constraints include each daily tour to be finished in a timely manner (considering the travel and visiting time of POIs), and not exceeding a predetermined monetary budget. This problem is computationally intractable and is similar to the traditional orienteering problem in operation research (Souffriau et al., 2008). Despite recent advances in tour planning systems, most of them still do not offer personalised tours and instead offer tourists a set of pre-determined and fixed tours (Kotiloglu et al., 2017). Furthermore, most of them do not take into considerations basic issues such as the category of the POIs. Tourists prefer to visit a variety of different places each day, and not only places from one category (e.g., very few tourists want to visit only museums in one day).

Most existing work on designing personalized itineraries (and orienteering problem) is based on the assumption of a single tourist, or only one tourist within the group decides for all members. In reality, many tourists travel in a group. However, each tourist has her own preference and most of the time, the places that one tourist wants to visit might be different from the places that another tourist prefers to visit (Zheng & Liao, 2019). If each tourist within a group designs and takes her own tour, she commutes in the new city by herself, takes a separate taxi, and does meal or food tasting all by herself. This clearly leads to no sharing of the ride nor the food, which results in greater energy use and less social well-being than travelling in a group.

Since the demand of multiple tourists must be met, this problem is more challenging than designing tours for a single tourist. In this work, a new framework is presented that generates a multi-day tour for a group of tourists. The framework is designed based on realistic assumptions. A framework is proposed that receives simple inputs from the tourists. However, the proposed complex algorithm guarantees all requirements are met. The generated multi-day tours meet intuitive constraints and are fair for all group members.

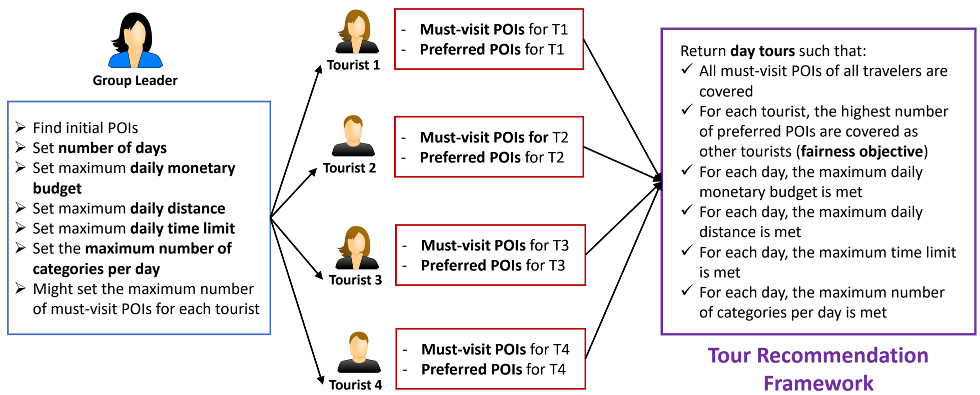

Figure 1: The framework of the tour recommendation system for groups.

The framework of this study is presented in Figure 1. It is assumed that a group of tourists have a designated group leader, which usually is one of the group members. The group leader makes initial decisions (although decisions by the group leader can be made in a collaborative way with other members). The group leader finds initial POIs (through personal experience and/or from the Web), set maximum daily monetary budget (for each group member), maximum daily travel distance, and daily time limit (this includes the time to travel within the city from one POI to another, and the time to visit each POI). The group leader also specifies the number of days a tour lasts, and the maximum number of categories per day (e.g., each day the group visits at most two parks). Each individual tourist then receives the list of potential POIs from the group leader and selects must-visit POIs and preferred POIs. Clearly, must-visit POIs are the points that each tourist absolutely wants to visit, while the preferred POIs are the points that the tourist would like to visit. In order to make realistic multi-day tours, the group leader can specify the maximum number of must-visit POIs each tourist can choose (e.g., five must-visit POIs from each tourist). The proposed framework then generates multi-day tours that satisfy all the monetary, distance, and time constraints, as well as covering all the must-visit POIs.

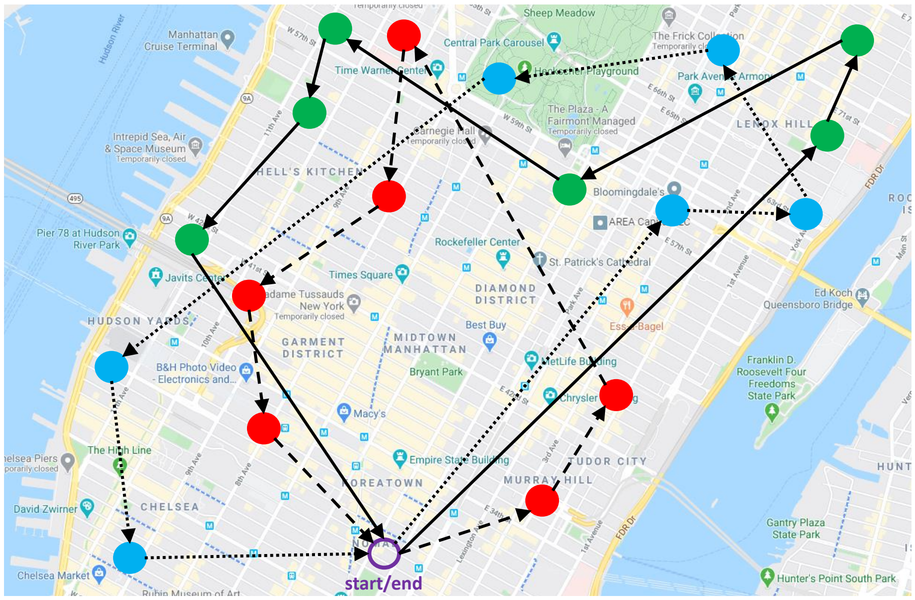

Figure 2: A portion of Manhattan (from Google Map) with preferred points by three tourists (in red, green, and blue points). If each tourist designs and takes her own tour, multiple routs should be taken.

The proposed framework is realistic and practical for designing a personalized itinerary for a group of tourists. While satisfying all the constraints, the main objective is to cover equal (and maximum) number of preferred POIs for each tourist (i.e., fairness objective). The secondary objective is to minimize either the distance, time, or money. In order to motivate the need to optimize the fairness, an example is presented in Figures 2 and 3. Here, a portion of Manhattan is shown with preferred points from three tourists. The preferred points for tourist one, two, and three are shown in red, green and blue circles, respectively. The start and end point is the same and is shown as a purple circle. Figure 2 shows the three possible tours that might be taken by each tourist, if each of them want to take her own tour. Clearly this leads to a number of routes, which means taking many taxis and no sharing. In Figure 3, two possible tours for sharing are shown (assuming monetary and time budget is available for only six points). The first tour, in dotted lines, does not take

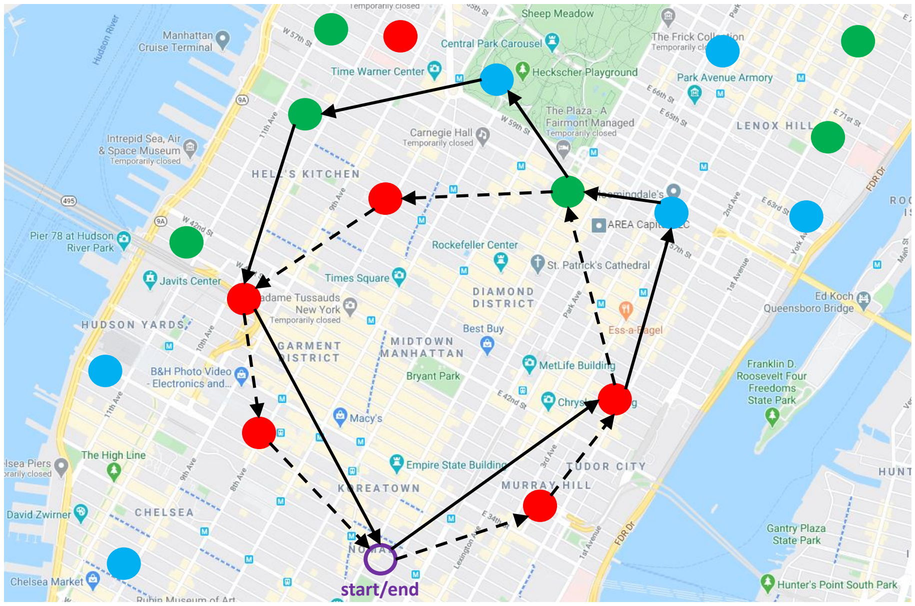

Figure 3: A portion of Manhattan (from Google Map) with preferred points by three tourists (in red, green, and blue points). The tour with dotted lines is designed without taking the fairness of individual tourists into account. The tour with solid lines takes into account the fairness of all tourists.

into account the fairness objective, and simply cover points that minimizes the distance. Therefore, this tour covers five points from the first tourist (i.e., red points) and one point from the second tourist (i.e., green point). No point from the third tourist (i.e., no blue point) is covered. Clearly, this tour is not fair, as majority of the covered points belong to one of the three tourists. On the other hand, the other tour, in solid line, covers two points for each of the three tourists and is fair for all tourists. Although the distance to visit points in this tour is longer than the other one, this one is preferred as it optimizes the main fairness objective.

Note that most prior work in designing personalized tours (and similar works in the orienteering problem) maximize the profit of points to be visited. Although the proposed framework can adapt to optimize the profit of points (as will be explained later), in the context of tourism, assigning scores and points to a POI is not realistic. When a tourist visits a new place, she either wants to see a POI (i.e., must-visit point), prefer to see a POI (i.e., preferred point), or does not want to see a POI (i.e., neither must-visit nor preferred point). However, assigning a numeric score (e.g., a score between 0 to 100) to a POI does not seem realistic. Note that previous work that does optimize the points of the POIs can be adapted easily to work with the proposed setting: the POIs that are preferred to be visited get the score of 1, while the ones that are not preferred get the score of 0. Another point worth mentioning is that in some situations, tourists within a group might have different level of fairness (Zheng & Liao, 2019). For example, in a family, parents might be OK to let their children see more preferred POIs than themselves. Later in this work, it will be shown how the general proposed framework is able to adapt to this situation too.

It should be noted that the parameters, requirements, and constraints that are used in this work are derived based on tourists' ideas. These parameters have been shown to be useful in previous work (through interviews with tourists). See for example Liao & Zheng (2018). Furthermore, these requirements are confirmed in a user study with different types of tourists (details are presented in the experiments).

The performance of the system is extensively evaluated using real data collected from Foursquare for two large cities, New York City and Tokyo. A user study is performed to show the effectiveness of the proposed framework from tourists point of view. The performance of diffident secondary objective functions in designing an itinerary is evaluated. The way different values and types of the input data change itineraries is discussed. The remainder of this paper is organized as follows. Section 2 discusses the related work and how this work differs from them. Section 3 presents the problem, and discuss in more details the requirements, constrains, and the objectives. The proposed multi-day tour framework is presented in Section 4. Experiments are presented in Section 5. Real world applications are discussed in Section 6 and Section 7 concludes the paper.

2. Related works

2.1. Tourism, social well-being and sustainability

Travelling together induces social interaction, such as intimate communication of various issues or topics among members of the group and brings various social benefits such as family capital and social capital including self-esteem and proactive behavior (Minnaert et al., 2009). Enjoying tourism and leisure activities together as a family creates shared memory, enhances familial bonding and cohesiveness, and the family's quality of life. The work in Jepson et al. (2019) suggest that family group tourism improves relationship, strengthens family bonds, increases the subjective well-being in both parents and children, and rekindles marriages and reduces the likelihood of divorce between a couple. However, one of the biggest challenges for group travel is to accommodate the heterogeneous needs and preferences of the different members in a group, which may sometimes result in conflicts and stress for the group members (Gram, 2005).

Moreover, taking a vacation and going on a tour is not an environmentally friendly activity (Dolincar, 2020). Tourists might harm the environment in different ways. This includes a wide range of activities, from taking flights, to staying in a hotel, and using fresh towels every day. Even when it comes to food, tourists usually waste food in taste-test and buffet. Tourists do not necessarily act responsibly toward the environment when they are away from home (Dolnicar & Grun, 2009). A recent study suggests that tourism is responsible for 8% of global greenhouse emissions (Lenzen et al., 2018). Therefore, it is critical to design more environmentally friendly tours, and encourage tourists to be more responsible toward the environment.

There are some recent works in the area of sustainable tourism. The work of Dolincar (2020) designs sustainable tourism services by encouraging tourist behaviour changes (e.g., by reducing the buffet plate size). Similarly, authors of Tiefenbeck et al. (2019) show that real-time feedback to tourists promote energy conservation even without monetary incentives. The work of Dolnicar et al. (2020) introduces a game-based intervention which encourage tourists to be more environmentally friendly while enjoying their holidays (in this case reducing plate waste). The work in Kim et al. (2016) reveals that the environmentally friendly activities of hotels indirectly increase customer satisfaction through perceived quality. While all these recent works contribute to the studies of sustainable tourism, this is the first work to propose a new recommendation system to encourage a group of tourists such as families and friends to explore a city destination together, which helps to reduce the carbon foot print, decreases the food waste through taste-test (since usually the group can share many foods together), and eventually makes their tour more enjoyable by increasing the social aspect of it.

2.2. Tour recommendation

Recommendation systems have been widely used in e-commerce platforms to automatically provide personalized information that closely match consumer needs and preferences of an individual customer (Cai et al., 2020). The information they provide about relevant product and services may attract consumer attention and generate interests in purchase and consumption, which helps to improve firms' performance in promotion and sales (Brynjolfsson et al., 2011). Thanks to recommender systems, consumers can simplify their decision process, by reducing information search costs and improving decision quality and effectiveness (Zhang et al., 2017), consequently more positive consumption experience, satisfaction and loyalty to the e-commerce platform or the service provider (Hostler et al., 2011). However, not all recommendation systems are effective, the outcomes of recommendations may be very different depending on the system's design. Designing a tour recommendation is a challenging task. It is essential that the design be "user-centered", focusing on user needs and preferences.

The problem of designing and planning personalized tours for travelers and tourists have gained significant attention in recent years. This is based on the motivation that over one billion tourists travel around the globe and tourism is an increasingly popular activity for many people (UNWTO, 2016). The tourist trip design problem has been defined for the first time in (Vansteenwegen & Oudheusden, 2007). Solution to the orienteering problem (OP) has been used to design single day tours. The same as traditional orienteering problem, a score (i.e., profit) is assigned to each POI. Authors of (Vansteenwegen & Oudheusden, 2007) assume the time to commute between each pair of points is given and the designed tour should not last more than a given time budget. The objective is to design a tour that maximizes the total point. In the case of having unknown scores for points, the objective is to maximize the total number of visited points.

The original work of (Vansteenwegen & Oudheusden, 2007) has been extended in recent years to address a variety of tourists requirements. This includes designing tours that last for more than one day (multi-day tours), asking tourists to label POIs based on their preference, considering the aesthetic fatigue and variable sightseeing value, taking into account the cost of visiting POIs (e.g., the ticket to visit a museum), and taking into account different categories of POIs (Kotiloglu et al., 2017). The work in (Zhu et al., 2012) divide available visiting points in two categories: accommodation and touristic points. The authors then propose an algorithm to maximize the profit for both categories, while satisfying the budget and time constraints. Visiting a city and commuting with public transportation has been addressed in (Gavalas et al., 2015). Navigating through the complex transport system has been recently addressed in (Zheng et al., 2020b). Recommending tours using user-generated contents in photo sharing social networks is proposed in (Sun & Lee, 2017).

Orienteering problem with hotel selection has been studied in recent years (Zheng et al., 2020a). In this problem, a set of POIs and a set of hotels are given. The goal is to maximize the score of the POIs, and start and end the tour in one of the hotels, while meeting some time, distance, and budget constraints. The work of (Tsai & Chung, 2012) constructs a route for theme parks using real-time information and the behaviour of tourists. Recently, authors of (Chen et al., 2020) propose a contextual collaborative learning model for personalized itinerary recommendation using the POI textual contents.

Although all these recent works make advancement in the field of personalized trip planning, majority of them assume the tour is designed for one tourist, or one tourist makes a decision for everyone. Most tourists travel with friends or families, i.e. in a group, and each group member has her own preference (Zheng & Liao, 2019). Therefore, it is of paramount importance to design personalized tours that meets the preference of all tour members.

The work in Zheng & Liao (2019) propose an algorithm that finds a set of Pareto optimal tours for a group of tourists. One drawback of returning a set of tours (as Pareto set) is that it is not clear how the group of tourists choose the best tour among all the tours returned by the algorithm. This is especially challenging if too many tours are returned as Pareto tours. Also, the tour designed in Zheng & Liao (2019) are single day tours, and the category of POIs are not taken into account. Another recent work that designs tours for a group of tourists is Anagnostopoulos et al. (2017). This work only design single day tours, and does not consider the category of tours. Furthermore, tourists are not able to specify the POIs they would like to visit.

There exist several recent related work to work infrastructure. Authors of Dui & Zhang (2019) analyze the simulations for an urban taxi sharing system and evaluate the relationships between the congestion and cost. The output of Dui & Zhang (2019) could be used to extend the proposed framework when congestion in routes is taken into account for recommending tours. Recently, authors of Tarantino et al. (2019) presents an interactive tool to recommend personalized tours to users through a web application. To increase user satisfaction, the application has been designed with attention to the usability and intuitive graphic interface. The problem of integrated self-driving travel scheme planning is studied in Du et al. (2021). The proposed solution optimizes routing, hotel selection, and time scheduling. Authors of Ceder & Jiang (2020) propose a flexible system for optimal paths for public transportation based on users' preferences at the time of the trip. A new lexicographical comparison methodology with a new consideration is proposed to capture users' perception. Authors of Huang et al. (2020) propose a multi-task deep travel rout planning system that integrates rich auxiliary information to produce high quality output. The system builds a heterogeneous network through the relations between users and points of interest, and use a heterogeneous graph embedding technique to learn the features of both users and points. Authors of Subramaniyaswamy et al. (2019) propose an ontology-based food recommendation system. In the proposed system, the effectiveness of recommendations is improved using a hybrid model of blended filtering. The idea of using an ontology for route recommendation could be an interesting extension to this work.

In this work, a novel framework is designed to recommend multi-day tours to a group of tourists. The proposed framework is practical, and takes the interest of all tourists into account, by receiving the must-visit and preferred POIs from each participant. Multi-day tours that are designed by the proposed algorithm meets various requirements, regrading distance, time, and monetary budget. Although the main objective is to cover equal (and maximum) number of preferred POIs for all tourists, the proposed framework is able to handle cases when some tourists in the group (like parents) might have weaker preferences versus other tourists in the group (like children).

Table 1: List of Notations.

Notation	**Description**
U	entire set of tourists participating in the tour
u_i	an individual tourist
D	total number of days the tour last
d_i	a single day
VP	entire set of POIs
v_i	a single POI
st_i	start point of day d_i
en_i	end point of day d_i
$e(v_i, v_j)$	the distance between v_i and v_j
$c(v_i)$	cost of visiting v_i
$t(v_i)$	time of visiting v_i
$f(v_i)$	profit of visiting v_i
M_i	set of must-visit POIs by tourist u_i
P_i	set of preferred POIs by tourist u_i
MO	set of all must-visit POIs by all tourists
PRF	set of all preferred POIs by all tourists
T	set of all points' categories
t_i	an individual category
$cat(v_i, t_j)$	returns 1 if point v_i is in category t_j, and 0 otherwise
TL_i	maximum number of times POIs in category t_i can be visited
V_i	list of selected POIs in day d_i
q_i	number of selected POIs in day d_i
VT	set of selected POIs for the entire tour
BGT	maximum spending budget of each tourist in each day
$DIST$	maximum travelling distance in each day
TIM	maximum time spent in each day
SP_{mile}	average time to walk per mile
TP_i	total time spent in all POIs in day d_i

3. Problem statement

The high level framework to solve the group tour recommendation problem is presented in Figure 1. The list of notations used in this work is summarized in Table

1. It is assumed that one of the group members takes the role of the group leader, and choose initial parameters. The set of tourists are shown as $U = \{u_1, \dots, u_n\}$ and it is assumed there are n tourists in the group that participate in choosing POIs and forming the itineraries. The number of days that the tour lasts is denoted as D and a single day is denoted as d_i $(1 \leq i \leq D)$. The POIs of the city to be visited for D days is modeled as a complete undirected graph. It is assumed to have m POIs in total and the entire set of POIs is: $VP = \{v_1, \dots, v_m\}$. The distance between v_i and v_j $(v_i, v_j \in V)$ is shown as $e(v_i, v_j)$. The Euclidean distance is used that satisfy the triangle inequality. The cost of visiting a POI is shown as $c(v_i)$ $(c(v_i) \geq 0)$. The time to visit the point v_i is shown as $t(v_i)$ $(c(v_i) > 0)$. Optionally, each POI might be associated with a profit (or score), which is shown as $f(v_i)$ $(f(v_i) \geq 0)$.

Table 2: List of Constraints and Equivalent Equation.

Equation	**Description of the Constraint**
Equation 1	No POI is visited more than once in one day
Equation 2	No POI is visited more than once in the entire duration of the tour
Equation 3	All must-visit points are visited
Equation 4	The quota for each category and in each day is met
Equation 5	Daily monetary budget is met
Equation 6	Daily distance threshold is met
Equation 7	Daily time limit is met

The set of must-visit and preferred POIs by tourist u_i is shown as M_i and P_i respectively ($M_i \subset VP$ and $P_i \subset VP$). The entire set of all must-visit POIs is the union of all M_i points and is shown as: $MO = \cup_{1 \leq i \leq n} M_i$ $(MO \subset VP)$. Each day starts from a start point (i.e., source) and ends in an end point (i.e., destination). The start and end points are usually the places that the group stays over night (e.g., hotels). The start and end points of day d_i are shown as st_i and en_i respectively ($st_i \in VP$ and $en_i \in VP$). Note that the proposed framework allows for the start and end points to be the same or different. Each point belong to at least one category (e.g., museum). The set of all categories is shown as $T = \{t_1, \dots, t_c\}$. $cat(v_i, t_j)$ returns 1 if point v_i is in category t_j, and 0 otherwise. The limit on the number of times points from category t_i can be visited in each day is shown as TL_i.

The following constraints are defined to address the problem's requirements (the list of all constraints are summarized in Table 2). Assume the selected POIs to be visited in day d_i is shown as list $V_i = [v_1, \ldots, v_{q_i}]$. The number of selected POIs in day d_i is q_i. Note that this is a list, which implies unlike a set, the order is important. The start and end points of day d_i (st_i and en_i) is not part of V_i. The entire set of selected points (i.e., the entire tour in D days) is shown as VT:

$$VT = \cup_{i=1}^{D} V_i$$

The following constraint guarantees no POI is visited more than once in one day:

$$\forall v_j \in V_i \quad 1 \leq i \leq D: \quad v_j \notin V_i^{-j} \tag{1}$$

in which V_i^{-j} means the list V_i without the element in the jth position. Similarly, the next constraint confirms that no POI is visited more than once in the entire duration of the tour:

$$\forall V_i, V_j \quad 1 \leq i, j \leq D \;\;\&\;\; i \neq j: \;\; V_i \cap V_j = \emptyset \tag{2}$$

The next constraint ensures all must-visit points are visited:

$$MO \subset VT \tag{3}$$

This constraint guarantees the quota for each category and in each day is met:

$$\forall V_i \;\; 1 \leq i \leq D \;\text{ and }\; \forall t_j \in T : \; \sum_{i=1}^{q_i} cat(v_i, t_j) \leq TL_j \tag{4}$$

The group leader sets the total spending budget for each day and it is shown as BGT (e.g., in dollar value). The sum of the cost of the POIs in each day should not exceed BGT. Note that $BGT \geq 0$. When the value of BGT is set to zero, this implies that tourists are only interested to visit POIs that are free of charge (e.g., visiting a public park). Thus, the following constraint must hold:

$$\forall V_i \;\; 1 \leq i \leq D: \; \sum_{i=1}^{q_i} c(v_i) \leq BGT \tag{5}$$

Also, the group leader sets the total distance the group is willing to commute each day. This distance is shown as $DIST$. The next constraint ensures the daily commute distance does not exceed this threshold:

$$\forall V_i \;\; 1 \le i \le D: \;\; e(st_i, v_1) \; + \sum_{i=1}^{q_i-1} e(v_i, v_{i+1}) \; + \; e(v_{q_i}, en_i) \; \le \; DIST \tag{6}$$

Note that in the above constraint, each day the group leaves from the start point st_i and visits the first point in V_i (which is v_1). Then, the group visits the remaining points in V_i in the order of the list, and at the end of the day, goes to the end point of day en_i.

The group leader sets the total time of the tour in each day (e.g., 8 hours). This is shown as TIM. The time spent in each day is composed of two activities: 1) commuting among POIs, and 2) the time spent in each POI. Therefore, the sum of these two times should not be over TIM. In order to find the first one, a unified average speed is taken by the tourists[1] (e.g., 7 miles per hour using a car or taxi). The average time to commute per mile is shown as SP_{mile}. Therefore, the total commuting time is the distance taken in each day multiplied by SP_{mile}. The total distance taken in day d_i is calculated as follows:

$$Dist_i = e(st_i, v_1) \; + \sum_{i=1}^{q_i-1} e(v_i, v_{i+1}) \; + \; e(v_{q_i}, en_i)$$

After finding $Dist_i$, the travelling time in each day will be: $Dist_i \times SP_{miles}$. It is required to calculate the total time spent in all POIs in day d_i, which is shown as TP_i. This will be calculated as follows:

$$TP_i = \sum_{i=1}^{q_i} t(v_i)$$

The following equation then ensures the time limit is met:

$$\forall V_i \;\; 1 \le i \le D: \;\; (Dist_i \times SP_{miles}) + TP_i \le TIM \tag{7}$$

[1] Note that other ways of computing the time to move from one point to another one can be used too. For example, using Google Map API to estimate the commute time.

The objective of the tour is to maximize the fairness of the selected preferred points for all tourists. In other words, the minimum number of selected preferred points for all tourists is maximized. This is modeled using the following equation:

$$\max \ \min_{u_i \in U} |P_i \cap VT| \tag{8}$$

The above equation calculates the selected number of preferred points by each tourist u_i by taking the intersection of her preferred points (P_i) with the selected points for the tour (VT). In case the POIs are associated with a score (or profit), the minimum of the sum of the profits of the preferred points by each tourist is maximized. This is modeled in the following equation:

$$\max \ \min_{u_i \in U} \sum_{v \in (P_i \cap VT)} f(v) \tag{9}$$

In addition to optimizing the fairness among all tourists, a second objective may be chosen too. The second objective to optimize could be the distance traveled in each day or the money spent in each day. More discussion about the second objective is provided in the next section.

4. Framework

In this section, the details of the framework to form itineraries for a group of tourists that covers all must-visit POIs and as many as preferred POIs for each tourist (the fairness objective) are presented. The problem that is tackled here is an extension of the orienteering problem, and is NP-hard (Kotiloglu et al., 2017). Therefore, a greedy algorithm is proposed to solve it. Since labeled data is not available as part of this problem, supervised machine learning methods are not applicable. On the other hand, greedy algorithms are shown to be effective to solve a range of orienteering problems (Souffriau et al., 2013). As per NFL theorem, new analytical systems are suitable for some set of problems, and may not be suitable for other situation. The proposed method is suitable for situations in which tourists know the set of must-visit and preferred points. Also, tourists should be aware of their re-

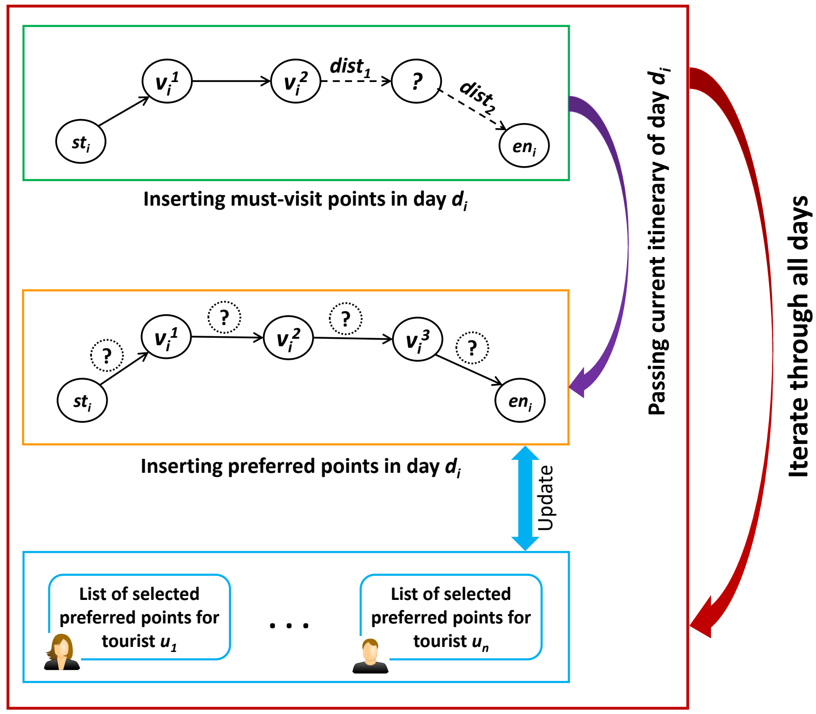

Figure 4: High level overview of the algorithm for covering must-visit and preferred points.

strictions and budgets (in terms of money, distance and time). If tourists want to explore a new city, without knowing the set of POIs, then the proposed solution is not efficient. The proposed greedy algorithm contains two phases. For each day d_i, the first phase covers as many remaining must-visit POI as possible. Then, the second phase inserts as many preferred POIs as possible, while optimizing the fairness objective (i.e., the minimum number of preferred covered points for each tourist is maximized, see Equation 8).

The high level overview of forming the itineraries is presented in Figure 4. The green box shows the first phase, in which must-visit points are added to each day d_i. The idea is as follows (see Algorithm 1 for detail). Starting from day one, the remaining uncovered must-visit points (i.e., MO) are received, which is the union of all must-visit points of all tourists. For each day d_i, new points are added after the current point. At the beginning of each day, the current point is simply the start point

Algorithm 1 Algorithm to Form a Multi-day Tour for Tourist Groups

Input: list of users $U = \{u_1, \ldots, u_n\}$; number of days D; set of all must-visit points as MO; set of preferred points by user u_j as P_j; start point of each day st_i; end point of each day en_i; category visit threshold for category t_k as TL_k; maximum daily budget BGT; maximum daily distance $DIST$; maximum daily time TIM

Output: a list of selected POIs for each day V_i $(1 \le i \le D)$

$covp_j \leftarrow 0$ $(1 \le j \le n)$; number of covered preferred points for each tourist u_j
for each day d_i $(1 \le i \le D)$ **do**
    $V_i \leftarrow []$;   a list of selected points for d_i
    $remBGT \leftarrow BGT$;   initiate remaining budget
    $remDIST \leftarrow DIST$;   initiate remaining distance
    $remTIM \leftarrow TIM$;   initiate remaining time
    $catV_k \leftarrow 0$ $(1 \le k \le c)$;   initiate number of visits to each category to 0
    $curP \leftarrow st_i$;   initiate current point to start point st_i
    while $remDIST > 0$ **do**
        $leastDist \leftarrow \infty$
        $selP \leftarrow \emptyset$
        for each must-visit point $mp \in MO$ **do**
            $dist_1 \leftarrow$ distance from mp to $curP$
            $dist_2 \leftarrow$ distance from mp to en_i
            $newDist \leftarrow dist_1 + dist_2$
            $ctime_{mp} \leftarrow$ time of commuting to mp based on $newDist$
            $price_{mp} \leftarrow$ price of visiting mp
            $ctg \leftarrow$ category of mp
            $vtime_{mp} \leftarrow$ time of visiting mp
            $totTime_{mp} \leftarrow ctime_{mp} + vtime_{mp}$
            if $price_{mp} \le remBGT$ **and** $newDist \le remDIST$ **and** $totTime_{mp} \le remTIM$ **and** $catV_{ctg} \le T_{ctg}$ **then**
                if $newDist < leastDist$ **then**
                    $leastDist \leftarrow newDist$
                    $selP \leftarrow mp$
        if $selP = \emptyset$ **then**
            BREAK
        V_i.**append**$(selP)$
        MO.**remove**$(selP)$
        UPDATE $remBGT$, $remDIST$, $remTIM$, $catV_i$
    InsertPref$(V_i, covp_j, P_j, remBGT, remDIST, remTIM, catV_k)$
return V_i $(1 \le i \le D)$

(i.e., st_i). In order to find the best next point for day d_i, for each remaining point mp in MO, the distance of adding mp to the list is calculated. As shown in Figure 4, this distance is composed of two parts: $dist_1$ which is the distance from the current point to the new point, and $dist_2$ which is the distance from the new point to the end point (i.e., en_i). In order for a point to be qualified to be added to the itinerary of day d_i, the following conditions are checked. The framework checks to makes sure by adding the new point, the monetary budget (BGT), the time limit (TIM), the distance threshold ($DIST$), and the category quota (TL_i) are not violated. Among the new points that do not violate these conditions, the one with the smallest distance is selected. Note that using the smallest distance to select the point means the secondary objective is to minimize the distance. However, it is possible to select a point with the minimum time or budget, which implies the time or price of visiting POIs is optimized. Note that the secondary objective is not critical for adding must-visit points to itineraries, since the ultimate goal is to cover all must-visit points. However, this is more important in the second phase, when the preferred points are covered. When all must-visit points are covered, or when no more must-visit point is eligible for day d_i, the second phase is called. Then, the remaining budgets and thresholds for day d_i along with the current number of covered preferred points for each tourist is passed to a procedure for adding the preferred points. If the last part of Algorithm 1 that inserts preferred points (which will bed discussed after presenting Algorithm 2) is excluded, the run time of Algorithm 1 is $O(D \times |MO|^2)$. In this statement, D is the number of days, and $|MO|$ is the number of must-visit points provided by all tourists. The reason for this is that there is a for loop in line 2 that goes through all days, and the maximum number of times the while loop of line 9 can be executed is equal to the number of must-visit points. Lastly, the for loop of line 12 goes through all must-visit points to choose the best one in each iteration while a distance budget is left.

Next, the second phase of the proposed framework is described, which covers the preferred points for the tourists. The high level overview of this phase is presented in the orange box of Figure 4 (see Algorithm 2 for detail). For each tourist u_j, a counter is kept, that stores the number of preferred points by u_j that is cur-

Algorithm 2 Algorithm to Insert Preferred POIs in Day d_i

Input: list of current selected must POIs V_i; number of covered preferred points for each tourist u_j as $covp_j$; set of preferred points by user u_j as P_j; remaining budget $remBGT$; remaining distance $remDIST$; remaining time $remTIM$; number of visits to each category $catV_k$;
Output: updated list of current selected POIs V_i

while $remDIST > 0$ **do**
    $leastDistL_j \leftarrow \infty$ $(1 \leq j \leq n)$
    $selPL_j \leftarrow \emptyset$ $(1 \leq j \leq n)$
    $posL_j \leftarrow \emptyset$ $(1 \leq j \leq n)$; selected position within V_i for each user
    $VFull_i \leftarrow$ **insert** st_i at the beginning of V_i and en_i at the end of V_i
    for each point $p_x \in VFull_i$ **do**
        if $p_x \neq en_i$ **then**
            for each preferred point $pp \in P_j$ **do**
                $dist_1 \leftarrow$ distance from pp to p_x
                $dist_2 \leftarrow$ distance from pp to p_{x+1}
                $newDist \leftarrow dist_1 + dist_2$
                $ctime_{pp} \leftarrow$ time of commuting to pp based on $newDist$
                $price_{pp} \leftarrow$ price of visiting pp
                $ctg \leftarrow$ category of pp
                $vtime_{mp} \leftarrow$ time of visiting pp
                $totTime_{pp} \leftarrow ctime_{pp} + vtime_{pp}$
                if $price_{pp} \leq remBGT$ **and** $newDist \leq remDIST$ **and** $totTime_{pp} \leq remTIM$ **and** $catV_{ctg} \leq T_{ctg}$ **then**
                    if $newDist < leastDistL_j$ **then**
                        $leastDistL_j \leftarrow newDist$
                        $selPL_j \leftarrow pp$
                        $posL_j \leftarrow x$
    if $\forall j : selPL_j = \emptyset$ $(1 \leq j \leq n)$ **then**
        BREAK
    $SU \leftarrow \emptyset$; selected user with the minimum number of preferred coverage
    $minCov \leftarrow \infty$
    for each user $u_j \in U$ **do**
        if $selPL_j \neq \emptyset$ **then**
            if $covp_j < minCov$ **then**
                $minCov \leftarrow covp_j$
                $SU \leftarrow j$
    insert $selPL_{SU}$ in position $posL_{SU}$ of V_i
    P_{SU}.**remove**$(selPL_{SU})$
    UPDATE $remBGT, remDIST, remTIM, catV_i$
return V_i

rently covered. These counters are depicted as blue boxes in Figure 4. After as many must-visit points as possible are added to day d_i, the itinerary of d_i is passed to the second phase. Note that, if all must-visit points are already covered before current day d_i, then this itinerary is empty before phase 2 is started. The idea in this phase is to try to add a point in between current points in the itinerary. For each user u_j, the point that adding it minimizes the distance while not violating other constraints is selected. Note that as mentioned before, this means the second objective is to minimize the distance. The same as must-visit points, for each tourist, only points that adding them do not violate any constrains (e.g., money or time) are allowed. After selecting a preferred point for all tourists, the counters that keep the number of covered preferred point for each tourist is checked. The tourist with the minimum number of covered preferred points that has a feasible candidate point is selected as the winner and her point is added to the itinerary of day d_i. Any tie is broken randomly. This process continues until no feasible point is found, or the remaining daily distance is met. The run time of Algorithm 2 is $O(|PRF| \times ((|MO| \times |PRF|) + |U|))$ in which $|MO|$ is the number of all must-visit points by all tourists, $|PRF|$ is the number of all preferred points by all tourists, and $|U|$ is the number of tourists. The first $|PRF|$ covers the while loop of line 1, which is executed until all $|PRF|$ points are covered. The $|MO|$ term represents the for loop of line 6, and the second $|PRF|$ terms represents the for loop of line 8. After the for loop of line 6 is terminated, the for loop of line 25 is executed $|U|$ times. Since in practice, $|U| \ll |MO| \times |PRF|$, the run time of Algorithm 2 is $O(|MO| \times |PRF|^2)$. Since Algorithm 2 is called from the for loop of line 2 in Algorithm 1 (which is executed for each day), the final run time of Algorithm 1 is $|D| \times (|MO|^2 + (|MO| \times |PRF|^2))$. In most practical cases, $|MO| < |PRF|^2$. Therefore, the run time of the proposed framework is $O(|D| \times |MO| \times |PRF|^2)$, which is polynomial in terms of all variables.

4.1. Optimizing the profit (score) of points of interest

The main objective is to maximize the minimum number of covered points for each tourist (the fairness objective). Unlike the traditional orienteering problem, it is not realistic to assign scores (or profit) to each point. In other words, a tourist either *wants to visit* a POI or not, and this is a binary selection. However, to have a complete framework, it is shown here how the proposed solution can be adapted to work in situations in which points are assigned with scores. In this case, the objective function is to maximize the minimum sum of the profit of each tourist (see Equation 9). In order to do this, instead of storing the number of covered preferred points for each tourist (i.e., the blue boxes in Figure 4), the cumulative sum of the profit of the preferred points that are covered for each tourist are stored. Also, when choosing the next tourist to add her points to the current itinerary, the one with the smallest sum of cumulative profits is selected.

4.2. Multi-objective optimization (chaotic system)

In a group of tourists, each tourist might have a different expectation, and hence the behaviour of tourists could be considered as a chaotic system. In order to address the conflicting behaviour of tourists, and motivate all to participate in the same tours, first the fairness objective is optimized, in which the same number of preferred points are covered by all tourists. In addition to optimizing the fairness among all tourists, there are other objectives to be optimized. This includes optimizing the traveled distance, the time or money. And this turns the problem into a multi-objective optimization problem. The algorithm that was discussed previously in this section optimizes the distance as the secondary objective. In order to change this to other objectives, in the first phase, among all feasible points, instead of choosing the one with the minimum distance, the one that minimizes the time or money is picked. With the same fashion, among all feasible preferred points of tourists, the one with the minimum time or money (instead of minimum distance) can be picked.

4.3. Different weight coverage for different group members

Until now, it is assumed the preferred points of all tourists should be covered equally. In order words, it is assumed points for all tourists must be covered as fairly as possible. This is usually the case when a group of friends travel together, and each tourist would like to see her preferred covered points are equal to others. However, there are situations within a group, in which tourists might require different level of fairness (Zheng & Liao, 2019). For example, if the group of tourists are composed of a family, parents would be fine to have fewer of their preferred points to be covered, while more preferred points of children are covered (e.g., children's points are covered 50% more than parents' points).

In order to address this situation, first a value between 0.0 and 1.0 is assigned to each tourist. The smaller the value, the more preferred points will be covered for that tourist. The number of covered preferred points depends on the relative difference of these values. Let's call these values $\{pv_1, \ldots, pv_n\}$. This implies that the assigned value for tourist u_i is pv_i. Now, when deciding the preferred point of which tourist should be selected, first the number of current preferred covered points of each tourist u_i is multiplied by the value pv_i. Then, the tourist with the smallest value is selected as the winner. For example, assume there are only two tourists: u_1 and u_2. Also, assume $pv_1 = 0.5$ and $pv_2 = 1.0$. This implies that the number of preferred covered points of tourist u_1 should be twice the one from u_2 (recall that smaller values means more covered points). Now, assume that at one stage, u_1 has 7 preferred covered points, and u_2 has 4. The multiplication of the number of covered points by u_i is $0.5 \times 7 = 3.5$ for u_1 and $1.0 \times 4 = 4.0$ for u_2. Therefore, u_1 will be selected as the next tourist to cover her points. If there is a tie among tourists, the one with the smallest value of pv_i will be selected. The values of pv_i will be chosen among tourists with the supervision of the tour leader.

4.4. Cyber security analysis

Cyber security is an important aspect of any online system today (Lu & Xu, 2019). This is crucial as many cyber attacks are reported around the world on a daily basis. Therefore, when the proposed system is implemented in a real scenario, multiple aspects must be taken into account from the security perspective. Here, potential threats that might threaten the system by a malicious actor is discussed. First, if an unauthorized person access the POI's database, she can delete or add POIs, and mislead the tourists. This can be done to gain personal benefits by the malicious actor. Second, the must-visit and preferred points of each user might be accessed, and the personal opinion of tourists might get exposed. Third, the recommended tours by the system might get exposed, and a malicious actor might be able to track a group of tourists and see where they travel. Therefore, it is of paramount importance that this system is built with highest security standards to protect the tourists from potential cyber security attacks.

4.5. Interactive tour recommendation

It is common that tourists change their mind after one (few) day(s) of the start of the tour. For example, a group of tourists might be interested in exploring museums in a new city, but after visiting a couple of museums, they might change their mind and prefer to visit more outdoor spots. The proposed system is able to adopt to this situation, and update the tours according to the new requirements of the tourists. After any given number of days since the start of the tour, each tourist can update her must-visit and preferred points. The system then receives these points, and keep only the points that have not been covered before. Then, a new tour will be formed using the new points, and priority will be given to the must-visit points. The same as before, for covering preferred points, the priority is fairness (i.e., each tourist get the same number of preferred points). Note that this process might be repeated a few times during the tour. Each time, new must-visit and preferred points are received from the tourists, and new tours will be formed.

5. Experiments

In this section, experiments to evaluate the proposed framework to form multi-day itineraries for a group of tourists are presented. The proposed framework is implemented in Python. First, the dataset that was used in the experiments is described. Second, the results of a user study is presented to confirm the benefits of the proposed model in motivating a group of tourists to take the tours together. Then, a series of experiments are run to show the effectiveness of the proposed framework.

5.1. Dataset and settings

A collection of Foursquare check-ins over the course of ten months (from April 2012 to February 2013) in two big cities of New York City and Tokyo is used. This dataset is originally published in Yang et al. (2015). Foursquare categorizes venues into different types. At the time of collecting the dataset, Foursquare classified venues into 9 main categories, and 417 sub-categories. For this application (i.e., forming routes for tourists), not all Foursquare venues are necessarily interesting. For example, hospitals or government and public service locations are not considered touristic attractions. Thus, these types of venues are excluded from the dataset. The same as Kotiloglu et al. (2017), 66 types are picked and put under 6 major categories. See Table 3 for details of venues' categories and Kotiloglu et al. (2017) for the name of the 66 Foursquare categories. The coordinates (latitude and longitude) of each POI are also obtained. The distance between a pair of POI is calculated using Python's GeoPy package[2]. A unified driving speed of 7 miles per hour is used for calculating the commute time[3].

Table 3: List of categories and relevant information.

Category Name	**Visit Time**	**Cost**
Parks	30 min.	$5
Bars and Night Clubs	60 min.	$10
Museums and Galleries	120 min.	$15
Great Outdoors	30 min.	$10
Scenic Attractions	30 min.	$15
Movie and Theaters	120 min.	$10

In order to create the input parameters for the proposed framework, first the number of tourists (i.e., n) is set. By default, the group consists of 5 tourists (including the group leader). The total number of must-visit points are 5, 10, 20, and 40 (which represents the MO set). This implies each of 5 tourists choose 1, 2, 4, and 8 must-visit points on average. Furthermore, each tourist can select 10 preferred points which gives us 50 preferred points in total. Note that for simplicity, selected

[2] *https://geopy.readthedocs.io/*
[3] *http://www.nyc.gov/html/dot/downloads/pdf/mobility-report-2018-screen-optimized.pdf*

points by tourists do not have an overlap. Both must-visit and preferred points are selected from the entire set of points for each city using random sampling. To be consistent, most of the settings is the same as the work of Kotiloglu et al. (2017). Default parameters of the proposed framework is listed below:

- The trip lasts for 8 days (i.e., $D = 8$)
- POIs are classified under 6 categories. List of categories along with the visiting time and visiting cost of each category is presented in Table 3.
- The limit to visit each category in each day is 5 (i.e., $\forall i \ TL_i = 5$)
- Each day is 12 hours, starting at 8:00am and ending at 8:00pm.
- The starting and ending points of all days are the same ($\forall i, j \ \ st_i = en_j$).
- The default daily budget is \$250.
- The default daily commute distance threshold is 25 miles.

The input parameters, and in particular the three parameters on maximum spending budget (BGT), maximum travelling distance ($DIST$), and maximum time (TIM) affect the final tour that is produced by the proposed algorithm. These parameters are determined by the tourists, and their values depend on the tourists' situation (e.g., how much money they want to spend each day). In order to observe the sensitivity of these parameters, the following experiment is performed. While keeping the values of all parameters fixed, only the value of one parameter (e.g., monetary budget) is changed. The output changes slowly as the value of the selected parameter changes. Specifically, by changing the value of the selected parameter by 10% (e.g., changing monetary budget from 100 to 110), the final tours do not change significantly. More importantly, while changing parameters might produce slightly different tours, the fairness objective (i.e., covering equal number of preferred points for all tourists) is always met.

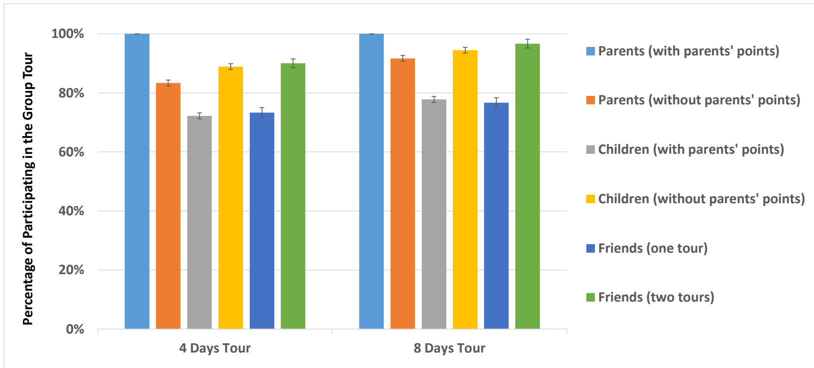

Figure 5: The result of the user study on whether tourists take the group tour or not.

5.2. User study

In order to understand if the proposed framework motivates tourists to participate in the group tour, instead of multiple individual tours, a user study is performed. Twelve group of tourists (each group is composed of five tourists) participate in the user study (in total sixty tourists). Within each group, the members knew each other. Six groups are composed of a family (i.e., two parents and three children) and six other groups are composed of friends (i.e., a group of five friends). Users are asked to select must-visit and preferred points to visit in NYC. Two settings are offered. In the first setting, a 4-day tour is designed. Each tourist could choose 2 must-visit points and 5 preferred points. In the second setting, an 8-day tour is designed. Each tourist could choose 4 must-visit points and 10 preferred points. Within the group of families, one tour is designed that include parents' points, and another tour that does not include parents' points and is only based on children's points. Within the groups of friends, one tour is designed based on all five members, and two tours. The two tours are designed based on two and three members points, which have points close to each other. The group tour generated by the proposed framework is presented to the users, and ask them whether they would participate in the group tour or not. The results are presented in Figure 5. As the results suggest, majority of tourists (from both family and friends groups) are willing to take the group tours. Furthermore, parents generally want to participate in group tours, even if their points are not covered. By removing parents' points, the chance of motivating children to participate in the group tour increases. This means in real applications, there could be more emphasize on the preferred points that are requested by children. Most friends are also willing to participate in group tours. However, offering two tours significantly increases the participation rate (by about 20%). Lastly, users are generally more motivated to participate in group tours that last longer. This could be because longer tours allow more points to be covered from all tourists, and also people are motivated more to spend time with the group when the tour is longer. Therefore, such a framework would be more useful in real applications when tours will last for more than a few days (e.g., one week or more).

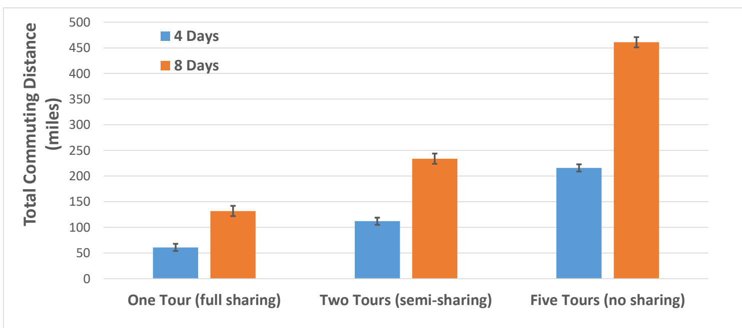

Figure 6: Total commuting distance (on average) for one, two, and five tours.

In order to show the effect of the proposed algorithm on the environment, Figure 6 shows total daily commuting distance of these 4- and 8-day tours (on average across all tours and for all groups). The results of having one tour (when all members share the same tour), two tours (when two members take one tour and three members takes another tour), and five tours (when each tourist takes her own tour) are shown. Based on the result of running t-test, total commuting distance of one tour is significantly different than two tours and four tours (and the same for two tours vs. four tours) with p-value < 0.05. The results suggest that when tourists share the tour, the daily commute decreases, and as a result, the traveling distance by taxis or cars decreases too. This significantly decreases the carbon emission by tourists. Also, in real applications, this implies that tourist will have a less negative impact on the environment, and are more cautious toward a greener environment.

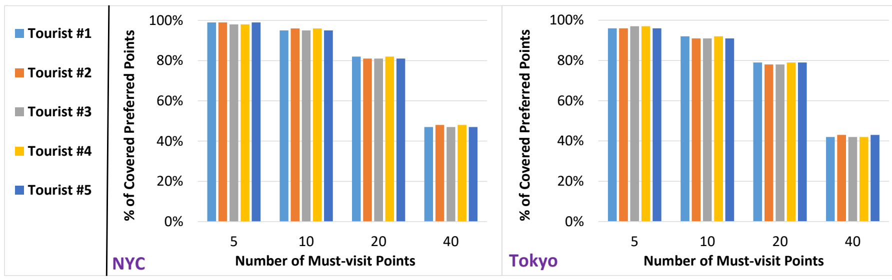

Figure 7: Percentage of covered preferred points for each tourist when the total number of must-visit points changes for an 8 days tour.

To evaluate the proposed framework from the usability point of view, users are asked to complete a usability questionnaire. In particular, users were asked to provide a 10-point scale judgement to the statement of the questionnaire, in which 10 indicates "strongly agree" and 1 means "strongly disagree". Users were asked to judge the following three statements: 1) The framework has a simple design, 2) It is easy to provide the required information to the system, 3) It is easy to understand the information and tours that are returned by the system. The average answers to statements 1, 2, and 3 are 8.9, 9.2, and 8.6 respectively. This result indicates that the proposed framework is easy to be used by end users. Please note that a more comprehensive usability testing is required when this system is implemented for public (either as a mobile app, or a website accessible to the public).

5.3. Effectiveness

The proposed algorithm is run to form itineraries over multi-days and for a group of five tourists. For each city, and each setting, 100 instances of the input parameters are randomly generated and the average results are reported. Figure 7 shows the percentage of covered preferred points for each of the five tourists when the total number of must-visit point varies. The results for both New York City and Tokyo are shown. The results suggest that the proposed algorithm successfully covers equal number of preferred points for each tourist for different number of must-visit points. Furthermore, a t-test shows that the number of preferred points for each pair of

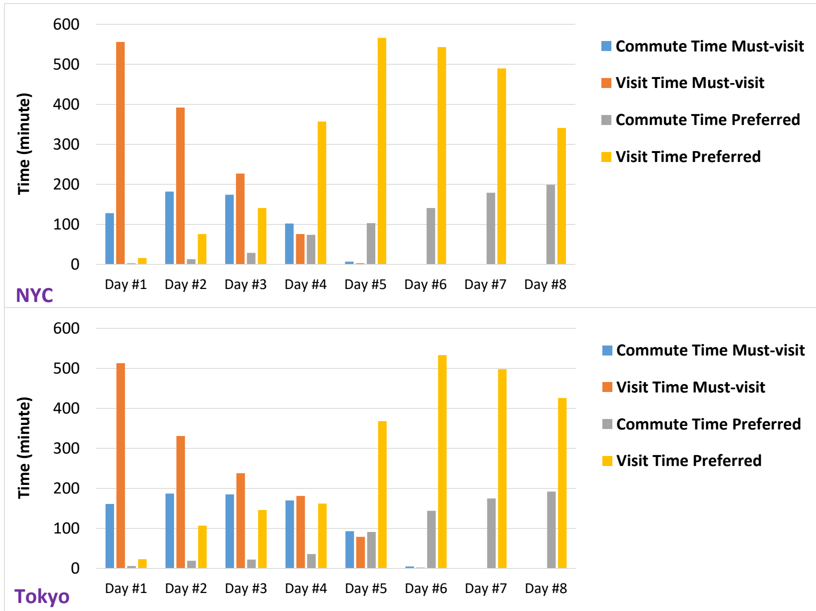

Figure 8: The average time tourists spend on commuting and visiting must-visit and preferred points when the number of must-visit points are 20.

tourists are not significantly different (p-value > 0.05). As expected, by increasing the number of must-visit points, the number of covered preferred points decreases for all tourists. By covering more must-visit points, while keeping the rest of the constraints as is (e.g., distance threshold and monetary budget), there is room for fewer preferred points to be covered during the entire tour. Furthermore, in the proposed algorithm, first the must-visit points are covered, and therefore, if the budget is limited, there will be less room to cover preferred points.

Figures 8 and 9 show the average time tourists spend on commuting and visiting must-visit and preferred points when the number of must-visit points are 20 and 40, respectively. The results are presented for each day. Based on the proposed algorithm, at the beginning of the tour, more time is spent on must-visit points. Closer to the end of the tour, more time is spent on preferred points. This is because one of the constraints is to cover must-visit points. Therefore, the proposed greedy strategy is to cover must-visit points as soon as possible while resources are still available (in terms of time, monetary budget, and distance). Furthermore, at the beginning of the

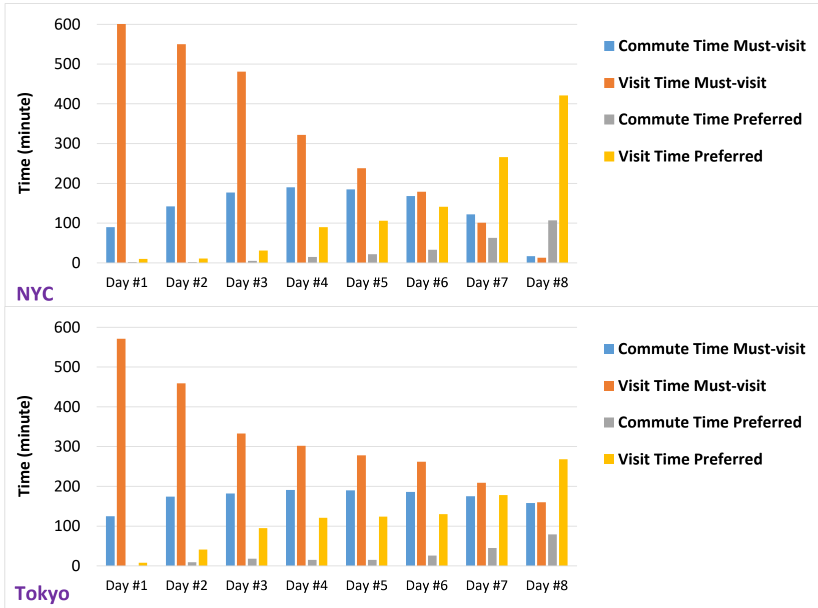

Figure 9: The average time tourists spend on commuting and visiting must-visit and preferred points when the number of must-visit points are 40.

tour, more time is spent on visiting must-visit points, than commuting to the must-visit points. This is because at the beginning of the tour (the beginning of day 1), the proposed greedy strategy picks must-visit points that are closer to each other (and are also close to the start and end points). Therefore, there is less time for commuting and more time for visiting. However, in the second day (and days after), more time will be spend on commuting since points are more far from each other and from the start and end points. Closer to the end of the tour, when most must-visit points are covered, there are more resources to cover preferred points. And this is reflected in the results.

Figure 10 shows the average number of covered preferred points with different second objectives when the number of must-visit points are 40. As mentioned before, the default second objective is to minimize the distance. That is, among all feasible points, the one that minimizes the distance (for both must-visit and preferred points) is selected. However, the time or money can be alternatively chosen to be optimized. The results suggest that optimizing distance generates slightly bet-

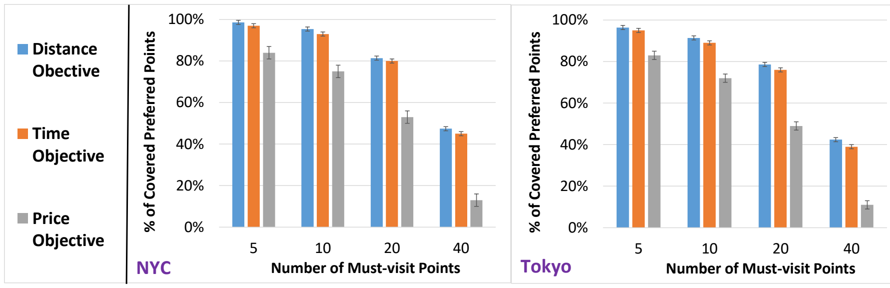

Figure 10: Average number of covered preferred points with different second objectives when the number of must-visit points are 40.

ter results than optimizing time (however it is not statistically significant as p-value > 0.05). This is expected, when the time is calculated, the time that is spent on commuting is taken into account. Therefore, lower distances results in shorter times. However, optimizing the price is not directly related to time nor distance. Thus, by optimizing the budget, points that are closer to each other are not necessarily selected. Also, the values of optimizing distance and time are significantly higher than the values of optimizing money (with p-values < 0.05). This result suggests that in real applications, optimizing distance or time should get priority over optimizing the cost, because the generated tour covers more preferred points, and as a direct result, provides higher satisfaction to tourists. The only exception to this rule is when tourists travel on a tight budget. In this case, visiting more preferred points can be sacrificed by visiting less expensive venues.

5.4. *Efficiency*

In this section, the efficiency of the system in terms of run time is evaluated. Note that based on the run time analysis of the proposed algorithm provided in Section 4, in addition to the number of days, the main parameters that affect the run time are total number of must-visit points ($|MO$) and total number of preferred points ($|PRF|$). Therefore, the run time of the proposed algorithm does not depend on the size of the input dataset, but the number of must-visit and preferred points. Thus, in Figure 11, the run time of the system is provided when the total number of preferred points varies for two fixed size total number of must-visit points. The remaining parameters are set to default (see Section 5.1 for details of default parameter values). This result confirms that the proposed algorithm scales well for larger values of the input parameters, and has polynomial run time.

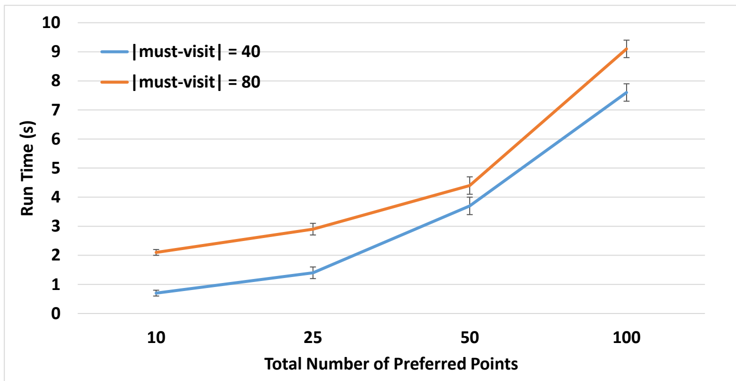

Figure 11: Run time of the proposed algorithm when the total number of preferred points varies.

5.5. *Comparison with other methods*

As mentioned before, there is no previous work that takes into account all the constraints and requirements that are covered in this work. Furthermore, most previous work form a tour for a single tourist, or assume the set of must-visit and preferred points are the same for all tourists, and as a result, ignore the fairness objective. However, in order to show the effectiveness of the proposed method, results of this work are compared with the results of two recent works: (Kotiloglu et al., 2017) and (Zheng et al., 2017). The work of (Kotiloglu et al., 2017) is referred to as Tabu Search, since it is based on the Tabu search approach. The work of (Zheng et al., 2017) is referred to as Heuristic Algorithm since it is a heuristic method. To be fair, only a subset of the requirements and constraints that are covered by all methods are kept. Furthermore, it is assumed there is only one tourist that provides the input parameters (since the other two methods are not designed for group tours). The results of forming 100 randomly generated instances are shown in Figure 12. The results suggest that the proposed method covers more preferred points than the other two (and this is significant with p-values < 0.05). This means the proposed method provides higher satisfaction to tourists, and encourage them more to participate in the generated tour in real life.

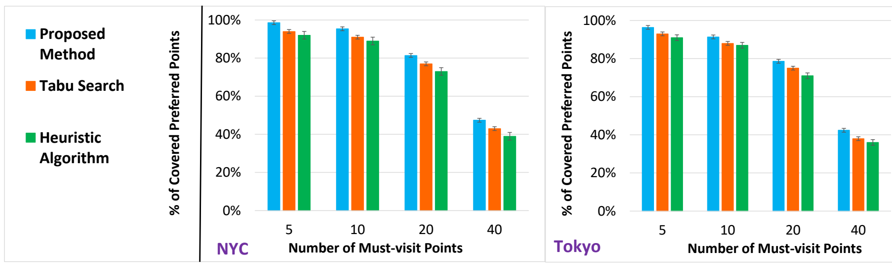

Figure 12: Comparison with other methods.

6. Discussion

Tourists nowadays have access to a variety of resources on the Web to design their itineraries. This includes Google Trips[4], TripHobo[5], and Tripadvisor[6]. These services usually contains map-based applications and offer a variety of services. This includes trip management and POI selection. However, these applications only show a ranked list of POIs based on the points' popularity (e.g., number of check-ins or reviews) and offer standard itineraries. They do not consider individuals' interest, nor some basic limitations (e.g., monetary budget, time, and traveled distance). More importantly, they fail to offer an itinerary that attracts a group of tourists.

Many people travel to a destination together, either as a family, or a group of friends. However, within a group, members have different preferences. Therefore, even though the same group of people might travel to a city together, it does not necessarily mean they visit POIs together. The set of POIs that one tourist might be interested to visit (e.g., a museum), might not attract other tourists. This situation

[4] *https://www.google.com/travel/*
[5] *https://www.triphobo.com/*
[6] *https://www.tripadvisor.com/*

could lead to tourists taking separate routes. However, taking separate routes means the use of more transportation resources (e.g., taxis). It also means less time socializing with other group members and a loss of various social benefits. And it could even results in wasting food, since the chance of sharing a meal decreases when each tourist explores the new city by herself.

6.1. Research contributions

This research contributes to the information management literature in several ways. First, a tour recommendation framework is developed that considers the social and environmental benefits of visiting a new city by a family or a group of friends (and not individually) (Jepson et al., 2019). By taking into account the preference of all tourists, it motivates all members to participate in one tour. Taking the tour by all members means sharing the taxi ride, more socializing, and even saving on food consumption (Dolincar, 2020).

Second, the proposed framework is realistic in achieving the objective of minimizing time, money, and the distance travel. It is assumed that one of the tourists in the group acts as the group leader, and selects the initial POIs, while considering other parameters (e.g., commuting distance, budget, and time). After this, each individual tourist selects a set of must-visit and preferred points. Then, the proposed framework returns a tour, that covers all must-visit points, ensures all requirements and constraints are met, and as many as preferred points as possible are covered for each tourist. In contrast, most prior work in designing personalized tours (and similar works in the orienteering problem) maximize the profit of points to be visited, which is often unrealistic.

Third, this study helps to resolve one of the biggest issues for group travel, i.e. accommodating the heterogeneous needs and preferences of the different members in a group (Gram, 2005). The ultimate objective of the proposed framework is to be *fair* towards all members, which means each tourist gets the same coverage of the preferred points. Moreover, the proposed framework is able to adapt to a tourist group that has different level of fairness (Zheng & Liao, 2019). For example, parents are happy for their children to have more preferred POIs than themselves.

6.2. Practical implications and potential applications

This framework is expected to be used by group of tourists that travel together (either as a family, or a group of friends). This framework is also expected to be used by destination tourism offices, to increase engagement with potential tourists. Destination tourism offices can integrate this framework by adding the POIs of their respective cities, and let groups of tourists design their own personalized tours. In order to make an effective system, destination tourism offices should provide updated information about available POIs, the exact location, and the cost of visiting the place. Destination tourism offices can use this framework for advertising new events during the year by adding the events as part of the set of POIs. The proposed framework can also be used for targeted marketing to replace classic hot-lines and brochures by offering personalised tours that are tailored for a specific group (Kotiloglu et al., 2017).

The proposed framework in this work has a number of applications in real life. First, this can be integrated into smart tourism applications used in mobile devices and be employed by a group of tourists that decide to travel together. This could be a family, a group of friends, or a mix of both. This framework will motivate the group to explore a new city together, and increase their social well-being. In addition to a group of tourists, tourism agencies are other potential applicants of this system. These agencies can apply the proposed framework to their list of POIs and create tours for different group of tourists with different preferences. Since the proposed framework forms tours based on tourists' input, it will increase the satisfaction of agencies' tourists and customers.

6.3. Sustainability perspective

Sustainability is composed of three fundamental dimensions: social equity, economic viability, and environmental protection (Hansmann et al., 2012). In this section, the proposed system from each dimension is analyzed. In terms of the social equity, this system provides a mean for all tourists within a group to participate in designing and forming the tour. Traveling together creates opportunities for social interaction, entertainment, information sharing, co-creating shared experience and memories and relieving stress, contributing to psychological and social well-being. In terms of the economic viability, in addition to sharing the ride to visit different POIs, tourists save on food by sharing among the group members. For the tourism industry, group tourists are an important market segment that brings substantial income to the destination. Providing an enjoyable and memorable experience for group tourists is essential to attract and retain this market segment. In terms of the environmental protection, motivating the members to have the same tour involves taking the same ride (e.g., taxi), and this helps to fully utilize resources, reducing waste and carbon emission.

6.4. Limitations and future research

The current research can be further improved by adding new directions into consideration. First, live traffic and congestion monitoring can be added to this framework. In this regard, routes with traffic, or congested POIs can be avoided to make the daily tour more enjoyable. Furthermore, this consideration makes the tour even more environmentally friendly. Second, the selection of hotels (or a place to stay overnight) can be added to this framework. By doing this, the location of the hotel, as well as its price and services can be taken into consideration. This framework can become more interactive, by receiving daily inputs from the tourists within a group, and design the tours for the remaining days accordingly (e.g., some tourists might remove museums from their must-visit POIs and replace them with parks). Lastly, the problem of tour recommendation when tourists use sustainable personal transporters to explore a new city, rather than taking public transportation, is a promising direction for future studies.

7. Conclusions

In this work, a new approach to recommending tours to a group of tourists who visit a new city is introduced. The goal of this recommendation system is to motivate tourists to explore the city together by covering the interests of all tourists equally. A comprehensive framework is designed that takes into account realistic assumptions and constraints about the tours. First, each tourist is asked to provide a set of must-visit and preferred points. The tour covers all must-visit points, and as many as preferred points as possible. The number of covered preferred points for all tourists are the same (i.e., fairness objective). The returned tour is guaranteed not to exceed the daily monetary budget, commuting distance thresholds, and time. The category of points are also taken into account, and the framework makes sure only a few points from each category is covered in each day. The proposed system has a positive effect on all three pillars of sustainability. In terms of social equity, it motivates tourists to explore a new city together, and hence it increases their social well-being. From the economic point of view, it helps tourist to save on transportation (by sharing the ride), and to save on food (by sharing food). In terms of environmental protection, it decreases the carbon emission of tourists since it motivates tourists to use the same transportation mode (i.e., taxi). Extensive experiments on a real dataset of POIs in New York city and Tokyo, and a user study, confirms the effectiveness of this framework in offering motivating and fair tours to a group of tourists. Using the proposed framework, tourists can explore a new city together, and enjoy visiting different areas, while being more sustainable toward the environment and society.

Acknowledgements

The author would like to thank the anonymous reviewers for their comments and suggestions that greatly improved the quality and presentation of this manuscript.

References

Anagnostopoulos, A., Atassi, R., Becchetti, L., Fazzone, A., & Silvestri, F. (2017). Tour recommendation for groups. *Data Min Knowl Disc*, *31*, 1157–1188.

Brynjolfsson, E., Hu, Y., & Simester, D. (2011). Goodbye Pareto Principle, Hello Long Tail: The Effect of Search Costs on the Concentration of Product Sales. *Management Science*, *57*, 1373–1386.

Cai, X., Hu, Z., Zhao, P., Zhang, W., & Chen, J. (2020). A hybrid recommendation system with many-objective evolutionary algorithm. *Expert Systems with Applications*, *159*, 113648.

Ceder, A., & Jiang, Y. (2020). Route guidance ranking procedures with human perception consideration for personalized public transport service. *Transportation Research Part C: Emerging Technologies*, *118*, 102667.

Chen, L., Zhang, L., Cao, S., Wu, Z., & Cao, J. (2020). Personalized itinerary recommendation: Deep and collaborative learning with textual information. *Expert Systems with Applications*, *144*, 113070.

Dolincar, S. (2020). Designing for more environmentally friendly tourism. *Annals of Tourism Research*, *84*, 102933.

Dolnicar, S., , Juvan, E., & Grun, B. (2020). Reducing the plate waste of families at hotel buffets. *Tourism Management*, *80*, 104103.

Dolnicar, S., & Grun, B. (2009). Environmentally friendly behaviour: Can heterogeneity among individuals and contexts/environments be harvested for improved sustainable management? *Environment and Behavior*, *41*, 693–714.

Du, J., Zhou, J., Li, X., Li, L., & Guo, A. (2021). Integrated self-driving travel scheme planning. *International Journal of Production Economics*, *232*, 107963.

Dui, H., & Zhang, C. (2019). Simulations for urban taxi sharing system on routes and passengers with numerical experiments. *Journal of Simulation*, .

Gavalas, D., Konstantopoulos, C., Mastakas, K., Pantziou, G., & Vathis, N. (2015). Heuristics for the time dependent team orienteering problem: Application to tourist route planning. *Computers & Operations Research*, *62*, 36–50.

Gram, M. (2005). Family holidays. A qualitative analysis of family holiday experiences. *Scandinavian Journal of Hospitality and Tourism*, *5*, 2–22.

Hansmann, R., Mieg, H., & Frischknecht, P. (2012). Principal sustainability components: empirical analysis of synergies between the three pillars of sustainability. *International Journal of Sustainable Development and World Ecology*, *19*, 451–459.

Hostler, R., Yoon, V., Guo, Z., Guimaraes, T., & Forgionne, G. (2011). Assessing the impact of recommender agents on on-line consumer unplanned purchase behavior. *Information and Management*, *48*, 336–343.

Huang, F., Xu, J., & Weng, J. (2020). Multi-Task Travel Route Planning With a Flexible Deep Learning Framework . *IEEE Transactions on Intelligent Transportation Systems*, .

Jepson, A., Stadler, R., & Spencer, N. (2019). Making positive family memories together and improving quality-of-life through thick sociality and bonding at local community festivals and events. *Tourism Management*, *73*, 34–50.

Kim, H., Hlee, S., & Joun, Y. (2016). Green practices of the hotel industry: Analysis through the windows of smart tourism system. *International Journal of Information Management*, *36*, 1340–1349.

Kotiloglu, S., Lappas, T., Pelechrinis, K., & Repoussis, P. (2017). Personalized multi-period tour recommendations. *Tourism Management*, *62*, 76–88.

Lenzen, M., Sun, Y., Faturay, F., Ting, Y., Geschke, A., & Malik, A. (2018). The carbon footprint of global tourism. *Nature Climate Change*, *8*, 522–528.

Liao, Z., & Zheng, W. (2018). Using a heuristic algorithm to design a personalized day tour route in a time-dependent stochastic environment. *Tourism Management*, *38*, 284–300.

Lu, Y., & Xu, L. (2019). Internet of Things (IoT) Cybersecurity Research: A Review of Current Research Topics. *IEEE Internet of Things Journal*, *6*, 2103–2115.

Minnaert, L., Maitland, R., & Miller, G. (2009). Tourism and social policy: The value of social tourism. *Annals of Tourism Research*, *36*, 316–334.

Souffriau, W., Vansteenwegen, P., Berghe, G. V., & Oudheusden, D. V. (2013). The multiconstraint team orienteering problem with multiple time windows. *Transportation Sciencet*, *47*, 53–63.

Souffriau, W., Vansteenwegen, P., Vertommen, J., Berghe, G., & Oudheusden, D. (2008). A personalized tourist trip design algorithm for mobile tourist guidess. *Applied Artificial Intelligence*, *22*, 964–985.

Subramaniyaswamy, V., Manogaran, G., Logesh, R., Vijayakumar, V., Chilamkurti, N., Malathi, D., & Senthilselvan, N. (2019). An ontology-driven personalized food recommendation in IoT-based healthcare system. *The Journal of Supercomputing*, *75*, 3184–3216.

Sun, C., & Lee, A. (2017). Tour recommendations by mining photo sharing social media. *Decision Support Systems*, *101*, 28–39.

Tarantino, E., Falco, I. D., & Scafuri, U. (2019). A mobile personalized tourist guide and its user evaluation. *Information Technology & Tourism*, .

Tiefenbeck, V., Schob, A., Fleisch, E., & Staake, T. (2019). Real-time feedback promotes energy conservation in the absence of volunteer selection bias and monetary incentives. *Nature Energy*, *4*, 35–41.

Tsai, C., & Chung, S. (2012). A personalized route recommendation service for theme parks using RFID information and tourist behavior. *Decision Support Systems*, *52*, 514–527.

Tsaur, S., Yen, C., & Chen, C. (2010). Independent tourist knowledge and skills. *Annals of Tourism Research*, *37*, 1035–1054.

UNWTO (2016). United nations world tourism organization (unwto) annual report 2015. URL: `https://www.unwto.org/archive/global/publication/unwto-annual-report-2015`.

Vansteenwegen, P., & Oudheusden, D. V. (2007). The mobile tourist guide: An OR opportunity. *OR Insight*, *20*, 21–27.

Yang, D., Zhang, D., Zheng, V. W., & Yu, Z. (2015). Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *45*, 129–142.

Yuan, H., Xu, H., Qian, Y., & Li, Y. (2016). Make your travel smarter: Summarizing urban tourism information from massive blog data. *International Journal of Information Management*, *36*, 1306–1319.

Zhang, Q., Wu, D., Lu, J., Liu, F., & Zhang, Z. (2017). A cross-domain recommender system with consistent information transfer. *Decision Support Systems*, *104*, 49–63.

Zheng, W., Ji, H., Lin, C., Wang, W., & Yu, B. (2020a). Using a heuristic approach to design personalized urban tourism itineraries with hotel selection. *Tourism Management*, *76*, 103956.

Zheng, W., & Liao, Z. (2019). Using a heuristic approach to design personalized tour routes for heterogeneous tourist groups. *Tourism Management*, *72*, 313–325.

Zheng, W., Liao, Z., & Lin, Z. (2020b). Navigating through the complex transport system: A heuristic approach for city tourism recommendation. *Tourism Management*, *81*, 104162.

Zheng, W., Liao, Z., & Qin, J. (2017). Using a four-step heuristic algorithm to design personalized day tour route within a tourist attraction. *Tourism Management*, *62*, 335–349.

Zhu, C., Hu, J., Wang, F., Xu, Y., & Cao, R. (2012). On the tour planning problem. *Annals of Operations Research*, *192*, 67–86.