Building City-Scale Walking Itineraries Using Large Geospatial Datasets

Ksenia D. Mukhina, Alexander A. Visheratin, Denis Nasonov
ITMO University
Saint Petersburg, Russia
{mukhinaks, alexvish91, denis.nasonov}@gmail.com

Abstract—**Nowadays, social networks play an important role in many aspects of people's life and in traveling in particular. People share their experience and opinions not only on specialized sites, like TripAdvisor, but also in social networks, e.g. Instagram. Combining information from different sources we can get a manifold dataset, which covers main sights, famous buildings as well as places popular with city residents. In this paper, we propose method for generation of walking tours based on large multi-source dataset. In order to create this dataset, we developed data crawling framework, which is able to collect data from Instagram at high speed. We provide several use cases for the developed itinerary generation method and demonstrate that it can significantly enrich standard touristic paths provided by official site.**

I. Introduction

Tourism became an inherent part of the vacations: a lot of people tend to spend their free time in a different country, and in recent years there is a rise of BRIC countries as an outbound traveling destinations [1]. Moreover, not only destinations change, but travel patterns transform as well. Mobile technologies and applications play an increasingly significant role in the tourism: from travel planning to recommendations of places-to-visit [2]. Usage of mobile or web services allows tourists to freely change their plans, unlike tour services [3]. Since it is impossible to see everything during the travel, people try to decide what to see with the help of supporting information from mobile applications to specific websites like official government guides. However, information available from official sources is mostly just a short list of interesting places and things to do. Official guides in some cases also provide information about touristic routes, but such maps are based on some set of the most famous locations in a city. For instance, on the Saint Petersburg Official City Guide 'Visit Petersburg' only 164 different locations are included in the official walking routes. In contrast, TripAdvisor contains ten times more interesting places for Saint Petersburg. And if we compare it with the number of points of interest in Instagram (6400 points), we can say that official site dramatically lacks information about the city.

The primary purpose of this work is to introduce a method for touristic walking routes construction, which satisfies time limitations and user preferences. To fill routes with interesting places, we use multi-source data that combines data from official city guide, TripAdvisor and Instagram. Instagram is a fast-growing social network dedicated to photo-sharing and a promising source for scientific researches [4]. Almost 20% of its content contains information about location, which is in 30 times more than in Twitter [5]. Combination of information sources allows to cover up all possible types of locations: from citizens′ the most popular Instagram spots to an establishment approved by tourists and the most valuable historical and cultural places. At the same time, this approach can shed the light of the actual popularity of places based on a combination of TripAdvisor rating and Instagram check-ins.

To satisfy time restrictions, we proposed a balancing strategy for daily routes, where the pathway begins in some user-defined location and goes through all selected sites concerning the time, which the person is ready to spend on sightseeing. The resulting path ends at the closest subway station if the individual has the route for next day, and at the last location otherwise. Ant colony optimization algorithm (ACO) [6] was used to construct such paths for each day separately. Interactive maps of routes obtained during experiments are available at GitHub (https://mukhinaks.github.io/walking-route-generation).

II. Related works

The issue of automatic recommendations for tourists has been actively investigated since the beginning of the 21st century [7]. With the spreading of the Internet all across the world, social media started to play a major role in the tourism area from marketing aspects to information search and decision making [8]. Geospatial data obtained from social networks (photos and posts) have been actively used to search for so-called areas or points of interest [9], the usage of big data technologies and clustering methods allows for the automatic detection of key attraction points [10]. As a source for geotagged data the photo sharing sites, like Panoramio [11], [12] and Flickr [13], [14], are widely used, but nowadays scientists are moving to social networks, i.e., Twitter [15], Foursquare, and Instagram [16], [17], due to their high growing rate and availability. Usage of tourists data helps to reveal the main visual tourist attraction areas in a city, which can be used for the future urban development or comparison between cities [18]. A simple method to divide tourist from locals is temporary windows, where all posts of the user must fit into N days. This concept was proposed by a group of researchers in 2008 [19]. In further studies, the size of the window varied, depending on the particular city [20]. In our previous work we successfully applied this concept to distinguish city residents from tourists [21].

Touristic route recommendations. Moreover, social network data is used for touristic recommendation systems, and some systems even suggest paths through city landmarks.

Photo2Trip [22] is a web-based service for trip route planning. The system allows to create paths concerning time spent in some area, e.g. park, the resulting path is constructed from places extracted from geotagged photos. Another approach was presented in [23], the recommended travel trajectory was obtained by a combination of existing popular routes, which were restored from Flickr geotagged photos. The improvements for path construction algorithm were proposed in [24], where authors presented several techniques to reduce the execution time. The time-respected travel recommendation system was described in [25]. The proposed approach consists of two phases: on the first stage route searching is performed to select the appropriate candidates; on the second stage heuristic algorithm is used to enhance the route. The algorithm for trip construction considering travel budget and time required for visiting points of interest was presented in [26]. However, in all works mentioned above potential attraction points were extracted from one source only, whereas recent studies showed that the more accurate results can be obtained using several sources [27]. Combination of multiple sources, Flickr and Wikipedia, was used in TripBuilder [28]. However, Wikipedia provides information only about a list of relevant places and limited by the historical and well-known locations. In [29], authors used touristic guides such as Yahoo Travel and Lonely Planet to extract information about places and combined it with Flickr data. In contrast, in our work we use different sources to enhance final dataset, not to narrow it down. The idea behind using multiple sources is to reveal interesting places, which stay unnoticed by specialized web services and government. It is the generally accepted that recommended route should consist of the list of established places. However, the results showed that people tend to visit both places from the official source and unmentioned areas.

III. Walking tours building method

Places of attraction in the city can be represented as a complex network [30] using the graph $G =< V, E >$, where V is a set of locations, and E is a list of paths between them. Thus, travel routes form a subgraph of the graph G. A search of the optimal path, which combines shortness and best tourists' valuation, is a NP-hard problem, as classical optimization problems, such as traveling salesman problem and knapsack problem [31]. The construction of route with predefined set of locations based on solving of Orienteering Problem with Compulsory Vertices [32], where resulting path must contain all nodes from the set of mandatory places. Metaheuristic algorithms have proven to be very efficient for solving these tasks [33], [34], [35]. The main idea of the method proposed in this paper is to apply ant colony optimization algorithm (ACO) to enrich the basic walking route generated using set of nearest points of interest gathered from social networks and official guide. Authors of [36] showed that ACO is one the most promising algorithm for solving orienteering problem in terms of solution quality and computation time.

Basic route. Before the start of the algorithm execution, the user specifies the number of datum points – locations, which he or she wants to visit. The number of these points is denoted as N_d. The user also specifies two parameters – number of days allocated for sightseeing D and maximum time of walking per day t_{max} The order of points' traversal can be changed using Google Directions API to obtain an optimal path. Thus, the initial route is represented as $G^{rt} =< V^{rt}, E^{rt} >$, where V^{rt} is an ordered datum points and $E^{rt} = \{e_i^{rt} =< v_i^{rt}, v_{i+1}^{rt}, t_i^{rt} >| \, i \in [0, N_d - 2], v_i^{rt}, v_{i+1}^{rt} \in V^{rt}\}$ is an ordered subset of transitions between location v_i^{rt} and v_{i+1}^{rt} where t_i indicates the time required to walk from v_i^{rt} to v_{i+1}^{rt} and visit v_{i+1}^{rt}. t_0 also includes the time required to visit v_0^{rt}. After that, the average time per day is calculated as an arithmetic mean of all times required for visiting places and times for walking between them:

$$t_{day\_average} = \frac{\sum\limits_{i=1}^{|E^{rt}|} t_i^{rt}}{D}, \tag{1}$$

where $\mid E^{rt} \mid$ is the cardinality of edges set. All locations are divided into groups for each day in order to balance daily routes. The group formed as a set of consecutive locations from the raw route represented by intervals $\left\{E_j^{rt}\right\}_{j=1}^{D}$, where $E_j^{rt} = \{e_i^{rt}...e_{i+n}^{rt}\}$, which total time is less than $t_{day\_average}$ and total time for set $E_j^{rt} = \{e_i^{rt}...e_{i+n}^{rt}, e_{i+n+1}^{rt}\}$ is greater than average time. In case when $e_{i+n+1}^{rt} =< (v_{i+n+1}^{rt}, v_{i+n+2}^{rt}), t_{i+n+1}^{rt} >$ cannot be added to E_j^{rt}, locations $v_{i+n+2}^{rt}...v_{N_d}^{rt}$ will be moved to the next day, and the last interval of E_j^{rt} will be the route from v_{i+n+1}^{rt} to the closet subway station. The route for next day will start from the closest subway station to v_{i+n+2}^{rt}. Finally, for each day every time t_i^{rt} for $e_i^{rt} \in E_j^{rt}$ is scaled in order to equalize the total time of daily route and t_{max}.

Path enrichment. For each day we get a basic route with a set of locations, which must be visited and required time for the transition between them. The enriched path for each interval e_i^{rt} is constructed by using ACO. Every ant in a colony tries to reach v_{i+1}^{rt} from v_i^{rt} by going through a set of candidates $V_j^{rt} \subset V$. Places that can be considered as candidates are selected from a circle area between two datum points $\mathcal{V}_j = \{v \in V| \, d({v_i}^{rt}, v) \leq d({v_i}^{rt}, {v_i}^{rt}{}_{+1})\} \backslash V^{rt}$ (Fig. 1a).

The probability of choosing location $\mathfrak{v}$ with previous location v_i^{rt} is defined by the following equation:

$$p_{v_i^{rt},\mathfrak{v}} = \frac{\tau_{v_i^{rt},\mathfrak{v}}^{\alpha} \cdot \eta_{\mathfrak{v}}^{\beta}}{\sum\limits_{v \in \mathcal{V}_j} \tau_{v_i^{rt},v}^{\alpha} \cdot \eta_{v_i^{rt},v}^{\beta}}, \tag{2}$$

where α, β are control parameters of the algorithm; $\tau_{v_i^{rt},\mathfrak{v}} = (1-\rho) \cdot \tau_{v_i^{rt},\mathfrak{v}} + \sum\limits_{ants} \Delta\tau_{v_i^{rt},\mathfrak{v}}$ is the amount of pheromones deposited by single ant, ρ is pheromones evaporation coefficient.

$$\Delta\tau_{v_i^{rt},\mathfrak{v}} = \begin{cases} \frac{\eta}{5} \cdot \left(\frac{t_j^{ant}}{t_{max}}\right)^2, & \text{if the constructed path satisfies time conditions,} \\ 0, & \text{otherwise;} \end{cases} \tag{3}$$

The attractiveness of a certain point is computes by this formula:

$$\eta_{\mathfrak{v}} = a \cdot (w_{spb}^{\mathfrak{v}} + w_{inst}^{\mathfrak{v}} + w_{trAdv}^{\mathfrak{v}}) + b \cdot r_{\mathfrak{v}} + c \cdot \kappa_{\mathfrak{v}}, \tag{4}$$

where a, b, c – weight coefficients,

$$w_{spb}^{\mathfrak{v}} = \begin{cases} 1, & \text{if } \mathfrak{v} \text{ is listed in the official guide,} \\ 0, & \text{otherwise;} \end{cases}$$

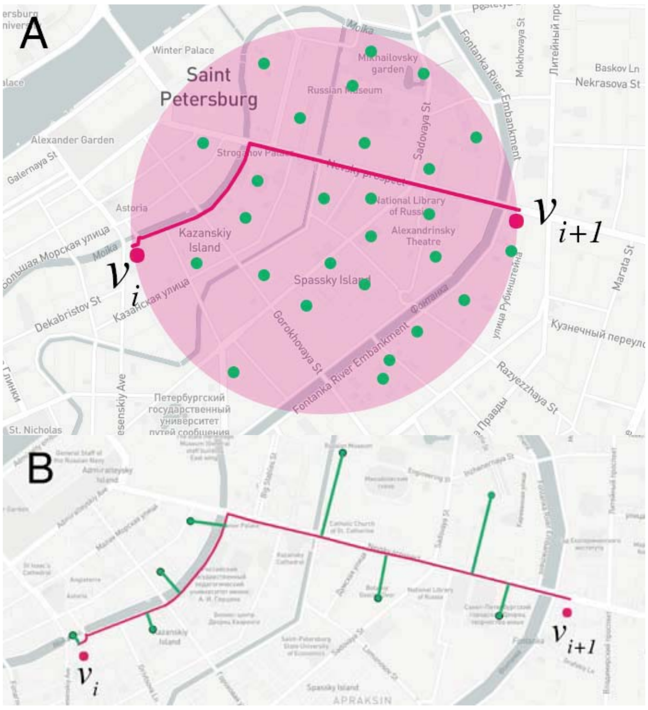

Fig. 1. a – Area of candidates to the resulting path, green dots are candidates; b – distances from candidates to original route (green lines)

$w^{\mathfrak{v}}_{inst} = \frac{k^{\mathfrak{v}}}{k_{max}} \in [0;1]$, where $k^{\mathfrak{v}}$ is the amount of Instagram visitors in point $\mathfrak{v}$, k_{max} is the maximum amount of visitors in selected area $\mathcal{V}_j$;

$w^{\mathfrak{v}}_{trAdv} = \frac{1}{5} \cdot \frac{\zeta^{\mathfrak{v}} \cdot \mu^{\mathfrak{v}}}{\mu_{max}}$, where $\zeta^{\mathfrak{v}} \in [0;5]$ is TripAdvisor's rating, $\mu^{\mathfrak{v}}$ – number of reviews for location $\mathfrak{v}$, and μ_{max} is maximum amount of reviews in area $\mathcal{V}_j$;

$r_{\mathfrak{v}} = \frac{\ell(v^{rt}_i, v^{rt}_{i+1})}{\ell(v^{rt}_i, v^{rt}_{i+1}) + d(\overline{v^{rt}_i, v^{rt}_{i+1}}, \mathfrak{v})}$, where $\ell(v^{rt}_i, v^{rt}_{i+1})$ is path length between points, $d(\overline{v^{rt}_i, v^{rt}_{i+1}}, \mathfrak{v})$ is the distance from point $\mathfrak{v}$ to path calculated as Euclidean the distance to polygonal chain (Fig. 1b);

$\kappa_{\mathfrak{v}} = \frac{1}{M} \sum_{i=1}^{M} \frac{u_i}{N_d}$, where M - amount of Instagram users, who visited current location $\mathfrak{v}$ and at least one of datum points V^{rt}, $u_i \in [0; N_d]$ is the amount of locations from list V^{rt}, which user visited.

Ant repeats the process of a new place selection until the location v^{rt}_{i+1} is reached or the constructed path exceeds the time limit. When all ants finish, the next iteration begins. The final route score is defined as total sum of scores $\eta_{\mathfrak{v}}$ of all locations except first one v^{rt}_0. The number of ants in a colony is dynamic and depends on the cardinality of $\mid \mathcal{V}_j \mid$ and equaled to $N\cdot \mid \mathcal{V}_j \mid$ ants, where $N \in \mathbb{N}$. The appropriate value of N as well as the number of iterations will be discussed in details in section V-A. Meantime, it should be noted that η is varied in the interval $(0; 5]$ since r always greater than 0 and at least one from the sum of $w_{spb}, w_{inst}, w_{trAdv}$ is distinct from zero, otherwise the place won't be on the list. That is why it is normalized by 5 for $\Delta\tau$ calculation. Control influence parameters are usually varied between 1 and 5 [37], [38], in this work $\alpha = 2$ and $\beta = 3$, evaporation coefficient is set to $\rho = 0.1$ since it is a common parameter for solving problems of similar types [39], [40]. In addition to that, due to equipollent importance of all factors $a = b = c = 1$.

Due to the stochastic nature of ACO, the path with the best score from all iterations is taken for each interval. After that, once again Google Directions API is used to construct a final path for each day in vacation. As a result, walking routes are built, and all locations are ordered in a convenient sequence to visit. The path for the first day starts at the first datum point v^{rt}_0 and ends at the subway station, the last day starts at the closest subway station and ends at the last location $v^{rt}_{N_d}$, all days in the middle start and end at subway stations.

IV. Datasets

In order to perform experimental evaluation of the developed method we collected three datasets from the following sources – official city guide, TripAdvisor, and Instagram. Since the site 'Visit Petersburg' provides information from server to web-client in JSON form, we easily gathered 164 unique places along with itineraries for them. TripAdvisor does not provide API for working with data, but all its pages have strict format and structure, we developed data crawler that extracts general information about places, their ratings and reviews. As a result, TripAdvisor's 'things-to-do', excluding touristic agencies and other things, which are not particular places, contain 1724 locations of different types. Collecting data from Instagram is much harder than from other sites because it does not provide public API and takes measures to prevent data crawling, like IP bans and requests signing with client-side tokens. Finally we came to the solution that operates in a distributed fashion using manager and worker nodes. Manager distributes entities, which are required to collect (locations or user profiles) among workers and workers perform requests through Instagram GraphQL API imitating behavior of users scrolling down web pages. As a result, raw Instagram data for the period from 1 January 2016 to 1 July 2017 contains 6436 geo points, 1,163,920 unique users and 11,667,119 posts.

A. Data preparation

TABLE I. Top 10 places by Instagram check-ins

Title	St. Petersburg Official City Guide "Visit Petersburg"	TripAdvisor's reviews	Instagram check-ins
Nevsky Prospect	yes	3445.0	207421.0
Palace square	yes	3904.0	132792.0
Saint Isaac's Cathedral	yes	8143.0	111739.0
Vasilyevsky Island	–	–	88114.0
Church of the Savior on Spilled Blood	yes	15742.0	86246.0
Pulkovo Airport St. Petersburg	–	–	83817.0
Petrogradsky District	–	–	68432.0
Park of the 300th anniversary of St. Petersburg	–	–	67395.0
Grand Palace	–	8281.0	64320.0
Krestovsky Island	–	–	61944.0

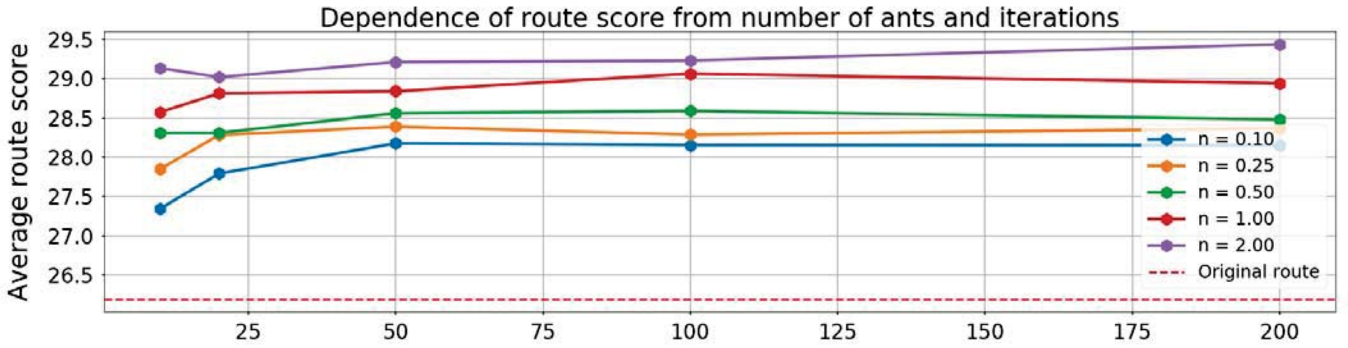

Fig. 2. Dependence of result path score from number of ants and amount of iterations: $n = 0.1$ – 0.1 ant per location candidate; $n = 0.25$ – 0.25 ants per location candidate; $n = 0.5$ – 0.5 ants per location candidate; $n = 1$ – 1 ants per location candidate; $n = 2$ – 2 ants per location candidate

The main problem of using datasets from different sources is to unite them correctly. Since the Internet resources set the geographical coordinates by themselves, there is a common situation when different data points correspond to the same place. In addition to this, the Instagram points related to one place can be at different distances and have different spelling of the name and address, which in the general case may not coincide, so it cannot be simply combined according to coordinates or title. Thus, the final list of locations was obtained by several steps. First, for datasets from TripAdvisor and Official City Guide, it is assumed that they consist of unique lists only. For Instagram places, information about addresses, names, and coordinates are extracted from Google using API Google Places. Points with the exact match of address, coordinates and name are considered as the same place. Finally, places combining by similar address and name considering this priority: Official City Guide, TripAdvisor, and the most popular Instagram place.

As a result, it was obtained 4,434 unique places for the city area. Herewith, from the 100 most popular Instagram places visited by more than 10,000 people, only 25 is listed in the official list and 48 present in TripAdvisor's list. For example, the first 10 places from list is presented in Table I. Thus, the use of data from Instagram opens up opportunities for detection of popular places that have not been taken into account in official sources.

To propose a reasonable duration of visit in some area, it was decided to divide all places into four categories: Nature & Parks, Sights & Landmarks, Concerts & Shows, and Museums & Libraries. Each category has its own estimated time of visit [28]: 15 minutes for Sights & Landmarks, 1 hour for Nature & Parks, and 2 hours for others.

V. Experiments

A. Algorithm adjustment

For the parameters adjustment of ACO, we took a route with 12 locations from the official city guide (http://www.visit-petersburg.ru/en/route/1/?category=1). This set of locations covers city center where the majority of PoIs located and it was also used for experiments. For all further experiments, the final path was computed inside the area where all locations are close to start and finish points (Euclidian distances between location and start and end points are less or equal than the Euclidian distance between start and end points). The ants' number coefficient n varied from 0.1 to 2, it was shown that for a large number of points in the dataset ACO performs better for the number of ants starting from $n = 0.25$. A number of iterations varied in range [10, 20, 50, 100, 200]. For each point the framework was launched 100 times. As it can be seen from the plot (Fig. 4), the average route score is growing with the increase of iterations number, but after 50 iterations the rise is slowing down. A similar tendency is observed for incrementing the ants' number: there is a dramatic rise in the average score when the number of ants doubled, however, in case of $n = 2$ the result is quite similar. Thus, the optimal parameters regarding performance and execution speed are $n = 1$ and 100 iterations.

B. Official routes enrichment

The first example is an improvement of one-day routes from Official City Guide. All points from walking paths were taken as datum points with maximum time for the whole route equaled to 10 hours. On the Fig. 3a, the route 'Alongside the Moika river' (http://www.visit-petersburg.ru/en/route/20/) is shown, it is clear that improved path 3b is quite similar to the original path. This result is caused by several time-consuming datum points. However, the initial number of locations almost doubled (6 and 11 for original and improved paths, respectively). All additional locations belong to 'Sights & Landmarks' category and require not more than 15 minutes to visit.

In Fig. 4a the original route 'Grand Ducal Petersburg' (http://www.visit-petersburg.ru/en/route/13/) is presented. It is clear that this route has small number of places which has a small number of places, and the obtained result differs signif-icantly. The enrichment path contains 15 locations including four datum points from the raw route and 11 extra.

Hence, in case of lack of time and several locations with a long-duration visit, the extra places will be mostly from 'Sights & Landmarks' group. With increased time limits or smaller the set of datum points, number of extra locations and their diversity in categories expands.

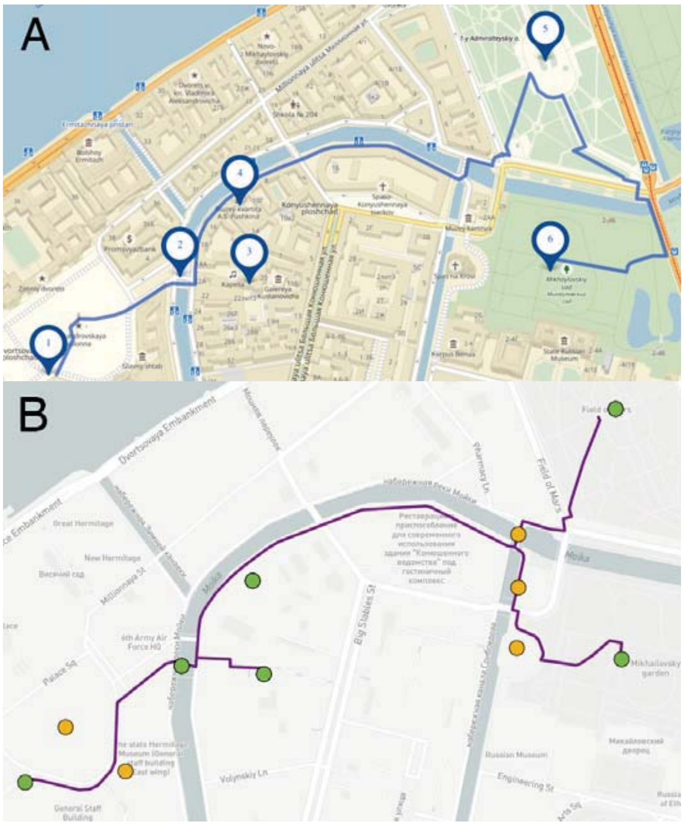

Fig. 3. a – 'Alongside the Moika river'; b – improved route: green indicates places from the original route, yellow – extra locations

C. Vacation length

To illustrate differences for 1-, 2-, and 3-days walking paths, we selected 5 places as the initial locations list: Park Pobedy, Sennaya ploschad' (square), Loft Project Floors Exhibition Hall, The State Hermitage Museum and Kazan Cathedral. This set of locations covers both residential district and city center. Also, these places have all possible types of estimated times of visit and different combinations of characteristics, for example, Loft Project Floors Exhibition Hall is famous on Instagram, but is not listed in the official guide, and The State Hermitage Museum has high ranking in all three sources. As can be seen from the Fig. 5 (top), due to the daily time limit new locations cannot be added to the path, so the route contains only datum points. In contrast, the walking path could be easily managed in 2 days (Fig. 5 center), locations were balanced between two days, that is why the first-day route (purple line) contains fewer datum points than in a single-day route. The path for the second day (green line) also starts at the subway station and finishes at the last mandatory location. 3-days path (Fig. 5 bottom) contains more interesting places for intervals in second and third days due to the balancing. The resulting scores are 13.52, 32.38 and 41.23 for 1, 2 and 3 days, respectively. The total path for the single-day route contains only 5 datum points; 2-days route includes 10 places for the first day and 7 places for the second day excluding subway stations; 3-days route contains 22 locations in total where 10 places are planned to visit on the first day, 5 places for the second day, and 7 places in the last day.

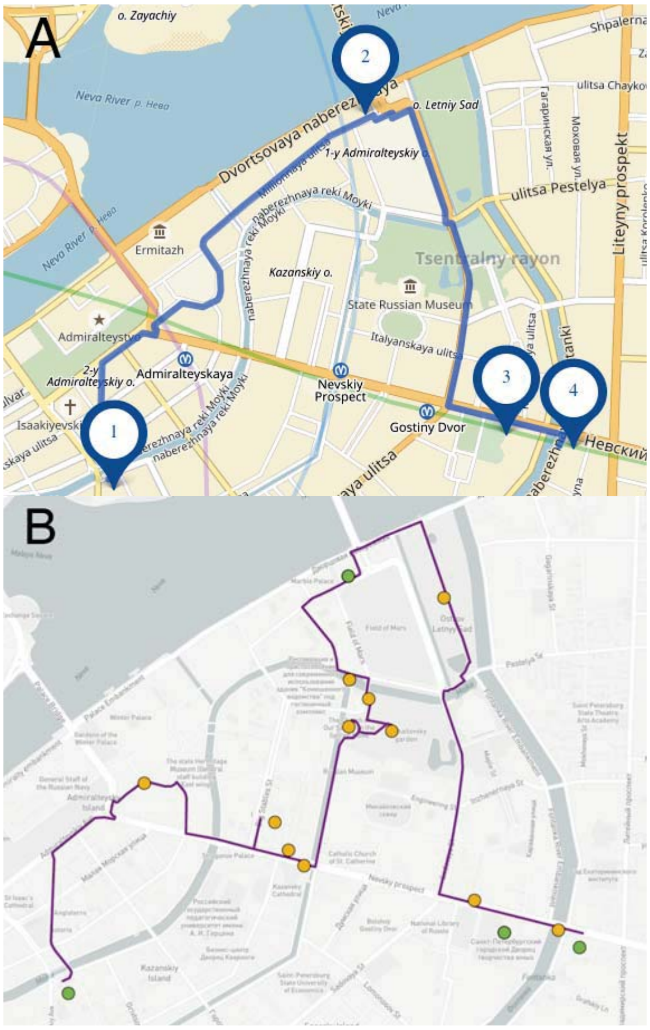

Fig. 4. a – 'Grand Ducal Petersburg'; b – improved route: green indicates places from the original route, yellow – extra locations

VI. Conclusion & future works

In this work, the method for touristic walking pathways generation was presented, resulting paths satisfy the time limitations for vacation length and daily walk and consists of official recommendations, high-ranking TripAdvisor places, and most popular Instagram locations. There was shown a comparison of the actual walking tour from Official City Guide and enrichment paths; results showed that even in worst conditions (large list of time-consuming locations and short duration of vacation) the resulting route contains a comparable number of new locations to the size of the initial list. In case then all locations can not be seen in a single day, the algorithm will produce balanced routes for each day of vacation starting and ending at subway closest to desirable locations.

However, there are several directions to enhance these results. The first direction is based on the idea to expand considered places to predict person's needs: cafe, restaurants, and WI-FI points could be included in the set of places. In addition to that, walking path can start and end in someone's

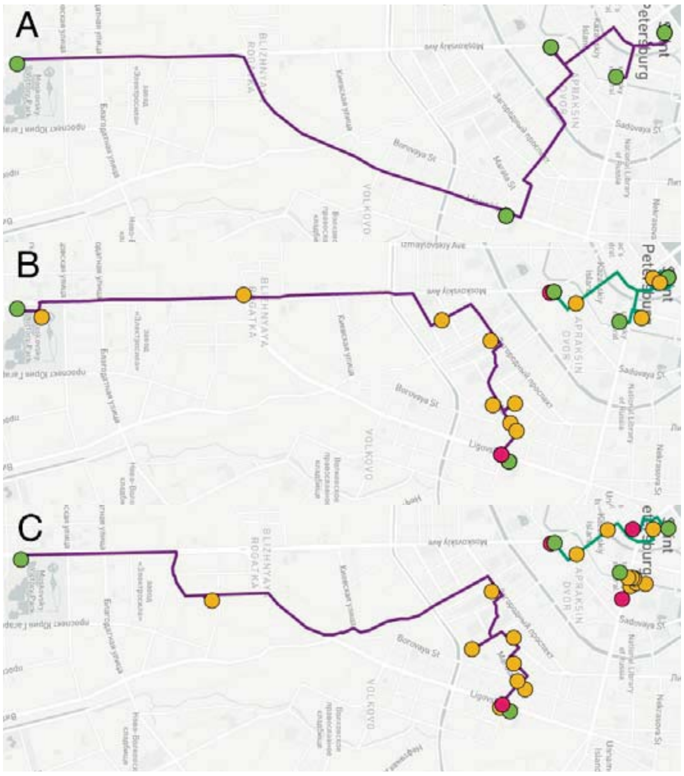

Fig. 5. Obtained routes: green circle indicates datum points, red circles – subway stations, yellow circle added locations, top – single day route; center – 2-days route; bottom – 3-days route

hotel and other types of public transport such as buses, trams could be added for convenience of the end user.

Extra aspect is corresponding to locations merge. If someone takes ancient palace, adjacent park, and fountain inside this park as datum points in our method they will be considered as three different places, but in fact, it is a single one. Nevertheless, this idea should be applied carefully, since if something is mentioned separately it is probably an important and exciting place, and user attention should be drawn to that location.

Another way of route improvement is to add information about work hours into the dataset. The plays in theaters usually take place at night, thus during the day theater should be considered only as Sights & Landmarks, museums commonly have one day off during a week, so usage of work hours combining actual travel dates will lead to increase of quality for resulting paths.

ACKNOWLEDGMENT

This research is financially supported by The Russian Science Foundation, Agreement #14-11-00823, and by RFBR according to the research project #18-37-00076/18.

REFERENCES

[1] M. M. Mariani, D. Buhalis, C. Longhi, and O. Vitouladiti, "Managing change in tourism destinations: Key issues and current trends," *Journal of Destination Marketing & Management*, vol. 2, no. 4, pp. 269–272, 2014.

[2] J. E. Dickinson, K. Ghali, T. Cherrett, C. Speed, N. Davies, and S. Norgate, "Tourism and the smartphone app: capabilities, emerging practice and scope in the travel domain," *Current Issues in Tourism*, vol. 17, no. 1, pp. 84–101, 2014.

[3] R. Kramer, M. Modsching, K. ten Hagen, and U. Gretzel, "Behavioural Impacts of Mobile Tour Guides," *Information and Communication Technologies in Tourism 2007*, pp. 109–118, 2007.

[4] L. Laestadius, "Instagram," in *The SAGE Handbook of Social Media Research Methods*, 2017, pp. 573–592.

[5] L. Manikonda, Y. Hu, and S. Kambhampati, "Analyzing User Activities, Demographics, Social Network Structure and User-Generated Content on Instagram," 2014.

[6] T. S. Marco Dorigo, Mauro Birattari, "Ant Colony Optimization . A Computational Intelligence Technique," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28–39, 2006. [Online]. Available: http://iridia.ulb.ac.be/~mbiro/Paperi/IridiaTr2006-023r001.pdf

[7] D. Dredge, "Destination place planning and design," *Annals of Tourism Research*, vol. 26, no. 4, pp. 772–791, 1999.

[8] B. Zeng and R. Gerritsen, "What do we know about social media in tourism? A review," *Tourism Management Perspectives*, vol. 10, pp. 27–36, 2014.

[9] Y. Hu, S. Gao, K. Janowicz, B. Yu, W. Li, and S. Prasad, "Extracting and understanding urban areas of interest using geotagged photos," *Computers, Environment and Urban Systems*, vol. 54, pp. 240–254, 2015.

[10] X. Zhou, C. Xu, and B. Kimmons, "Detecting tourism destinations using scalable geospatial analysis based on cloud computing platform," *Computers, Environment and Urban Systems*, vol. 54, pp. 144–153, 2015.

[11] S. Kisilevich, M. Krstajic, D. Keim, N. Andrienko, and G. Andrienko, "Event-based analysis of people's activities and behavior using Flickr and Panoramio geotagged photo collections," *Proceedings of the International Conference on Information Visualisation*, pp. 289–296, 2010.

[12] H. Yin, C. Wang, N. Yu, and L. Zhang, "Trip Mining and Recommendation from Geo-tagged Photos," *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2012*, pp. 540–545, 2012.

[13] J. A. Donaire, R. Camprub, and N. Gal, "Tourist clusters from Flickr travel photography," *Tourism Management Perspectives*, vol. 11, pp. 26–33, 2014.

[14] S. Su, C. Wan, Y. Hu, and Z. Cai, "Characterizing geographical preferences of international tourists and the local influential factors in China using geo-tagged photos on social media," *Applied Geography*, vol. 73, pp. 26–37, 2016.

[15] A. Ritter, O. Etzioni, and S. Clark, "Open Domain Event Extraction from Twitter," *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining.*, pp. 1104–1112, 2012.

[16] T. H. Silva, P. O. S. Vaz de Melo, J. M. Almeida, J. Salles, and A. A. Loureiro, "A comparison of Foursquare and Instagram to the study of city dynamics and urban social behaviour," *Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing - UrbComp '13*, pp. 1–8, 2013.

[17] Y. Rykov, O. Nagornyy, and O. Koltsova, "Semantic and geospatial mapping of instagram images in saint-petersburg," *power*, vol. 2607, p. 75, 2016.

[18] J. C. García-Palomares, J. Gutiérrez, and C. Mínguez, "Identification of tourist hot spots based on social networks: A comparative analysis of European metropolises using photo-sharing services and GIS," *Applied Geography*, vol. 63, pp. 408–417, 2015.

[19] F. Girardin, F. D. Fiore, C. Ratti, and J. Blat, "Leveraging explicitly disclosed location information to understand tourist dynamics: a case study," *Journal of Location Based Services*, vol. 2, no. 1, pp. 41–56, 2008.

[20] B. Kádár and M. Gede, "Where Do Tourists Go? Visualizing and Analyzing the Spatial Distribution of Geotagged Photography," *Cartographica*, vol. 48, no. 2, pp. 78–88, 2013.

[21] K. D. Mukhina, S. V. Rakitin, and A. A. Visheratin, "Detection of tourists attraction points using Instagram profiles," in *Procedia Computer Science*, vol. 108, no. June, 2017, pp. 2378–2382.

[22] X. Lu, C. Wang, J.-M. Yang, Y. Pang, and L. Zhang, "Photo2Trip: generating travel routes from geo-tagged photos for trip planning," *Mm*, pp. 143–152, 2010. [Online]. Available: http://dl.acm.org/citation.cfm?id=1873951.1873972

[23] K. Okuyama and K. Yanai, "A travel planning system based on travel trajectories extracted from a large number of geotagged photos on the Web," in *The Era of Interactive Media*, 2013, pp. 657–670.

[24] E. H. C. Lu, C. Y. Lin, and V. S. Tseng, "Trip-mine: An efficient trip planning approach with travel time constraints," in *Proceedings - IEEE International Conference on Mobile Data Management*, vol. 1, 2011, pp. 152–161.

[25] C. Chen, D. Zhang, B. Guo, X. Ma, G. Pan, and Z. Wu, "TripPlanner: Personalized trip planning leveraging heterogeneous crowdsourced digital footprints," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1259–1273, 2015.

[26] S. H. Fang, E. H. C. Lu, and V. S. Tseng, "Trip Recommendation with Multiple User Constraints by Integrating Point-of-Interests and Travel Packages," in *2014 IEEE 15th International Conference on Mobile Data Management*. IEEE, 7 2014, pp. 33–42.

[27] A. Farseev, I. Samborskii, A. Filchenkov, and T.-s. Chua, "Cross-Network Recommendation Via Clustering On Multi-Layer Graphs," *Proceedings of the 40th International ACM SIGIR conference*, vol. 10, pp. 195–204, 2017.

[28] I. Brilhante, J. A. Macedo, F. M. Nardini, R. Perego, and C. Renso, "Where Shall We Go Today ? Planning Touristic Tours with Trip-Builder," in *CIKM*, 2013, pp. 757–762.

[29] M. De Choudhury, M. Feldman, S. Amer-Yahia, N. Golbandi, R. Lempel, and C. Yu, "Automatic construction of travel itineraries using social breadcrumbs," in *Proceedings of the 21st ACM conference on Hypertext and hypermedia - HT '10*, 2010, p. 35.

[30] L. Zhou, Y. Zhang, J. Pang, and C. T. Li, "Modeling city locations as complex networks: An initial study," in *Studies in Computational Intelligence*. Springer, Cham, 11 2017, vol. 693, pp. 735–747.

[31] K. Kulakov and A. Shabaev, "An approach to creation of smart space-based trip planning service," *Conference of Open Innovation Association, FRUCT*, vol. 2014-Decem, pp. 38–44, 2014.

[32] M. Gendreau, G. Laporte, and F. Semet, "A branch-and-cut algorithm for the undirected selective traveling salesman problem," *Networks*, vol. 32, no. 4, pp. 263–273, 1998.

[33] N. Ulder, E. Aarts, H.-J. Bandelt, P. van Laarhoven, and E. Pesch, "Genetic local search algorithms for the traveling salesman problem," in *Parallel Problem Solving from Nature SE - 17*. Springer, Berlin, Heidelberg, 1991, vol. 496, pp. 109–116.

[34] T. Stützle and M. Dorigo, "ACO Algorithms for the Traveling Salesman Problem," *Evolutionary Algorithms in Engineering and Computer Science*, pp. 163–183, 1999.

[35] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, "A survey on algorithmic approaches for solving tourist trip design problems," *Journal of Heuristics*, vol. 20, no. 3, pp. 291–328, 2014.

[36] A. Anagnostopoulos, R. Atassi, L. Becchetti, A. Fazzone, and F. Silvestri, "Tour recommendation for groups," *Data Mining and Knowledge Discovery*, vol. 31, no. 5, pp. 1157–1188, 2017.

[37] M. Reimann, K. Doerner, and R. F. Hartl, "D-Ants: Savings Based Ants divide and conquer the vehicle routing problem," *Computers & Operations Research*, vol. 31, no. 4, pp. 563–591, 4 2004.

[38] T. Stützle and H. H. Hoos, "MAX MIN Ant System," *Future Generation Computer Systems*, vol. 16, pp. 889–914, 2000.

[39] L. M. Gambardella and M. Dorigo, "An Ant Colony System with a New Local Search for the Sequential Ordering Problem," *INFORMS Journal on Computing*, vol. 12, no. 3, pp. 237–255, 2000.

[40] L. M. Gambardella, . D. Taillard, and G. Agazzi, "MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows," *New Ideas in Optimization*, no. IDSIA-06-99, pp. 1-17, 1999.