2 Methodology

This section will elaborate on collecting travel products, user-profiling methods, the itinerary generator and the implementation used to build this application. Figure 4 outlines the overall process of our personalised itinerary generation framework.

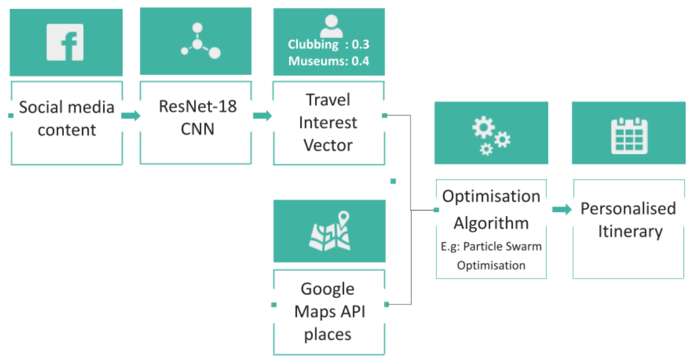

Figure 4: Personalised itinerary generator

2.1 Retrieving travel products

We implemented the Google Maps API as the data source for our application because of its real-time accuracy and massive dataset compared with the other approaches and other APIs that we discussed in the literature review [22, 23]. In addition, the nearby search endpoint allows the app to search for places of a given category within a specified area. In order to retrieve the places for the application, eight requests are made, each requesting places of different categories. To solve the issues with time windows, we split the endpoints into two categories. Five of the requests represent places shown as part of the itinerary during the day:

beaches, natural sights, museums, shopping malls and cafeterias.

and the rest represent places shown during the night:

nightclubs, bars and restaurants.

Figure 5 shows the query parameters used to gather cafeteria related places in Malta. These categories are based on the ones used by Wörndl et al. [21] for their application.

Query Params

	KEY	VALUE
☑	location	35.9375,14.3754
☑	radius	50000
☑	type	cafe
☑	keyword	must visit tourist

Figure 5: Sample Query being made to the google maps nearby search endpoint

In return, the API returns a list of places of the specified area and category and attributes about each place. The attributes used by our application include the place's name, rating, the number of reviews and the coordinates. All of these attributes help the application further optimise the algorithm to find the perfect itinerary.

2.2 Generating the User Profile

Social media's effect on the world is something significant [56]. That is why this application builds a user profile from the user's social media.

The application built by Lim et al. [8] allowed the user to connect the application with their Flickr profile to scan their past trips. However, Facebook provides an API that would allow users to connect both their Facebook and Instagram accounts and request content from the user with their permission. A significant advantage is that the API allows the application not to limit the results to mimic only past user's trips like the application by Lim et al. [8] and gather preferences from his complete profile. The app requests two things from the potential tourist's social media, the photos and the liked pages and tries to classify these into six categories that make up the user's travel interest vector;

[1 Beach, 2 Nature, 3 Shopping, 4 Museums, 5 Clubbing, 6 Bars]

These categories are the same categories that we requested from the google maps API except 'cafeterias' and 'restaurants'. These two categories were not included because the application tries to suggest the best places to eat as part of the timetable, irrelevant to the user's profile. At the start of the application, the app initialises all vector values to zero and increments a value whenever the user's content matches a category. We will describe how the app classifies both the user's liked pages and the user's photos separately in the upcoming subsections.

2.2.1 Transforming the liked pages into the travel interest vector

The Facebook API allows the application to request each category of the user's liked Facebook pages. The API's documentation contains a whole list of possible categories.

The app iterates through all of these user's likes categories and increments a value in the user's vector whenever the Facebook result matches one of the six travel interest vector values. For example, if a user likes a page with class 'DJ', the user's clubbing vector value is incremented, and if a page is labelled as a 'Mountain', the app increments the user's nature vector value.

2.2.2 Transforming the user's photos into the travel interest vector

Convolutional Neural Networks have become a standard for classifying an image because of their high accuracy [57]. Therefore, we decided to test out two approaches for classifying the photos into the app's six categories.

Zhou et al. [57].trained several CNNs on the places365-standard dataset of about 1.8 million images to classify an image into 365 different scene categories. However, the places365 model is not specifically trained on the six categories of our application. Therefore, we need to carefully map the 365 categories with our six application's categories. We also introduced a Tensorflow Keras sequential model, explicitly trained on the six application's categories to compare.

We used the Resnet places365 models, Resnet-18 and Resnet-50 since they achieved the highest top-5 validation accuracy on the places365 dataset. The Resnet 18 comprises 18, and the Resnet 50 comprises 50 convolutional layers. They both converge an output layer representing the 365 output categories. Figure 7 shows a summary of the whole Resnet 18 model.

The Tensorflow Keras Sequential model comprises three convolutional layers with a rectified linear unit (ReLu) activation function. A pooling layer follows each to lower the input volume's spatial dimension for the upcoming layers. The final layer represents a flattening layer and two dense layers to reduce the outputs to the six application categories, and another layer representing the 'None' classification. Figure 6 shows a summary of the whole model. The dataset comprises X public internet images representing the seven classes: Beach, Nature, Museums, Shopping, Clubbing, Bars and None.

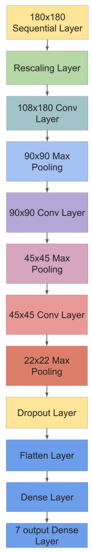

Figure 6: Keras Sequential Architecture Summary

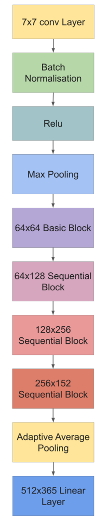

Figure 7: Resnet 18 Architecture Summary

2.3 Producing the activity plan

After the app generates the dataset of POIs and the user's travel interest vector, we formulate an efficient activity plan using these two inputs. This itinerary generator is based on the existing state of the art activity planners [19, 33] with some adjustments: We wanted the trip's output to take the form of an itinerary. The problem takes the form of the 'day' and 'night' category split discussed in the literature review. The scoring of itineraries is adjusted with the travel interest vector.

The problem definition of our novel itinerary planner is mathematically formulated as follows. A tourist trip is made up of some pre-defined user constants alongside the travel interest vector. The predefined constants are:

M:	The number of travelling days
C:	The moderation of activities (ie. the greater the C value, the more activities are generated in a day)

The objective function of our itinerary planner is:

$$\text{MAX}\sum_{m=0}^{M}(S_{D_m}+S_{E_m})$$

where:

m	Travelling day (m=1,2,..., textitM)
D_m	Morning section of day number m
E_m	Evening section of day number m
S_{D_m}	Score of the morning section D_m
S_{E_m}	Score of the evening section E_m

A day is made up of the morning D_m section and the evening E_m section. The morning section is made up of $C+2$ tourist attractions whilst the evening section is just made up of 2.

$$D_m = Y_i + Y_f + C(Y_i)$$

$$E_m = Y_f + Y_j$$

i	Morning Tourist attraction (i = 1, 2, 3,..., n_1)
j	Evening Tourist attraction (j = 1,2, 3,..., n_2)
f	Food Place (f = 1,2, 3,..., n_3)
$Y_{i\|f\|j}$:	1 if a tourist visit attraction i, j or f and 0 if otherwise

Constraints

$\sum_{m=0}^{M}\sum_{i=0}^{n_1} Y_i \leq 1$	Ensures that all morning tourist attractions are not visited more than once throughout the whole itinerary
$\sum_{m=0}^{M}\sum_{j=0}^{n_1} Y_j \leq 1$	Ensures that all evening tourist attractions are not visited only once throughout the whole itinerary

2.3.1 Calculation of Score

The score S_{D_m} or S_{E_m} is calculated using

$$S_{D_m|E_m} = \frac{1}{T} + R + V$$

where:

T	Total distance between each tourist attractions in the morning/evening of day m
R	Average rating of the tourist attractions in the morning/evening of day m
V	how much the tourist attractions of the morning/evening of day m match with the user's travel interest vector

2.3.2 Particle Swarm optimisation algorithm

Kennedy et al. [58] proposed the original PSO algorithm in 1995 designed to solve optimisation problems. The algorithm is a population-based technique that uses n elements called particles. Each particle has a d-dimensional *position* vector representing a solution and a d-dimensional *velocity* vector expressing the direction of the particle during its search period.

When a PSO program initialises all of the particles, they are usually set to a random or predetermined value. In our algorithm, we introduce a method of randomisation bias. Although the initial particles are generated randomly, the randomness is weighted and affected by three things:

1. The user's travel interest vector
2. the place's rating
3. the place's number of ratings.

We implemented the randomness bias to give a head start to the algorithm rather than just starting optimising from purely random itineraries. Figure X shows an example of a sample place with its probability of being chosen as part of the initial particles alongside a sample tourist interest vector. At each iteration in the algorithm, the velocity of each particle is calculated based on the inertia constant and how well it is doing compared with its personal best score and the global best score. The inertia constant helps the particle explore new solutions and escape the local minima. After a few iterations have passed, particles use this velocity and move towards the optimum position. We demonstrate the framework of our PSO algorithm in Figure 8.

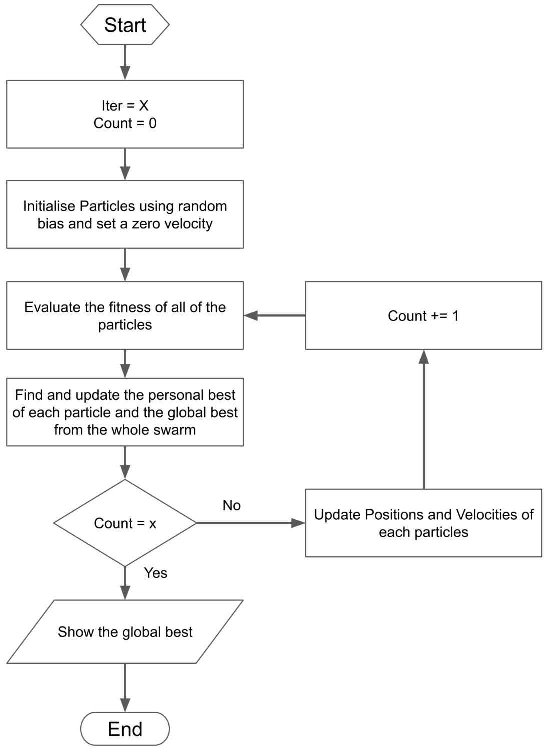

Figure 8: Framework of PSO algorithm

2.4 Web application implementation and user interface

We built the application using several technologies where each communicates with each other to provide a user-friendly website for the potential tourist. Figure 9 shows the tech stack diagram of the website. The website is accessible through the URL `https://www.touristplanner.xyz`. We built the front end of the website using HTML, CSS and javascript and hosted it on a cloud Vultr server. The website is responsive to be accessible from both a mobile phone and a laptop. The website communicates with the back end of the application using REST endpoints, hosted on a separate dedicated server provided by Hetzner using the Java Spring Boot framework. Another Python Gunicorn server is used to generate the itinerary and calculate a travel interest vector which sends the information directly to the Spring boot server. Finally, a local instance of an Open Source Routing Machine server calculates the distances from one tourist attraction to another used by the Gunicorn server to optimise the itinerary.

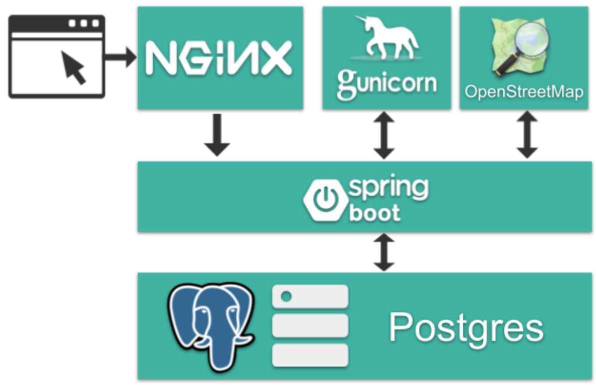

Figure 9: Tech Stack implementation of the application

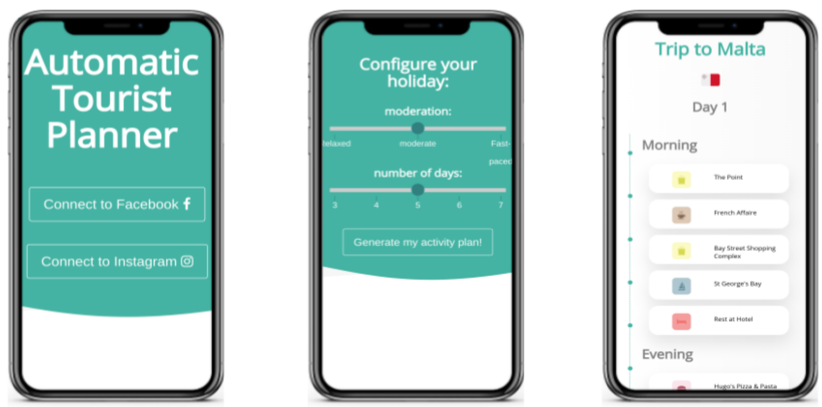

Figure 10: User Experience Timeline

Figure 10 shows screenshots of the website portraying the user's timeline. The user navigates to the homepage, accepts terms and conditions and connect his social media profiles. The user selects the number of days M and the activity moderation C. The website navigates to the final page of the application exhibiting their personalised itinerary.

References

[1] D. A. Herzog, "A User-Centered Approach to Solving the Tourist Trip Design Problem for Individuals and Groups," Tech. Rep., 2020.

[2] L. Santamaria-Granados, J. F. Mendoza-Moreno, and G. Ramirez-Gonzalez, "Tourist Recommender Systems Based on Emotion Recognition—A Scientometric

Review," *Future Internet*, vol. 13, no. 1, p. 2, dec 2020. [Online]. Available: https://www.mdpi.com/1999-5903/13/1/2

[3] P. Di Bitonto, F. Di Tria, M. Laterza, T. Roselli, V. Rossano, and F. Tangorra, "Automated generation of itineraries in recommender systems for tourism," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6385 LNCS. Springer, Berlin, Heidelberg, 2010, pp. 498–508. [Online]. Available: www.expedia.com

[4] K. H. Lim, J. Chan, S. Karunasekera, and C. Leckie, "Tour recommendation and trip planning using location-based social media: a survey," *Knowledge and Information Systems*, vol. 60, no. 3, pp. 1247–1275, dec 2018. [Online]. Available: https://link.springer.com/article/10.1007/s10115-018-1297-4

[5] M. De Choudhury, M. Feldman, S. Amer-Yahia, N. Golbandi, R. Lempel, and C. Yu, "Automatic construction of travel itineraries using social breadcrumbs," in *HT'10 - Proceedings of the 21st ACM Conference on Hypertext and Hypermedia*, 2010, pp. 35–44. [Online]. Available: http://www.munmund.net/pubs/ht{_}10{_}long.pdf

[6] I. Memon, L. Chen, A. Majid, M. Lv, I. Hussain, and G. Chen, "Travel Recommendation Using Geo-tagged Photos in Social Media for Tourist," vol. 80, pp. 1347–1362, 2015.

[7] C. Lucchese, R. Perego, F. Silvestri, H. Vahabi, and R. Venturini, "How Random Walks can Help Tourism," 2012. [Online]. Available: http://www.flickr.com

[8] K. H. Lim, J. Chan, C. Leckie, and S. Karunasekera, "Personalized trip recommendation for tourists based on user interests, points of interest visit durations and visit recency," *Knowledge and Information Systems*, vol. 54, no. 2, pp. 375–406, feb 2018. [Online]. Available: https://doi.org/10.1007/s10115-017-1056-y

[9] K. Hui Lim, J. Chan, C. Leckie, and S. Karunasekera, "Personalized Tour Recommendation based on User Interests and Points of Interest Visit Durations," Tech. Rep., 2015.

[10] K. Hui Lim, S. Karunasekera, C. Leckie, and J. Chan, "Recommending Tours and Places-of-Interest based on User Interests from Geo-tagged Photos." [Online]. Available: http://dx.doi.org/10.1145/2744680.2744693.

[11] T. Kurashima, T. Iwata, G. Irie, and K. Fujimura, “Travel route recommendation using geotagged photos,” *Knowledge and Information Systems*, vol. 37, no. 1, pp. 37–60, oct 2013. [Online]. Available: https://link.springer.com/article/10.1007/s10115-012-0580-z

[12] T. Kurashima and K. Fujimura, *Travel Route Recommendation Using Geotags in Photo Sharing Sites*, 2010.

[13] I. Brilhante, J. Antonio Macedo, F. Maria Nardini, R. Perego, and C. Renso, “Where Shall We Go Today? Planning Touristic Tours with TripBuilder,” 2013. [Online]. Available: http://dx.doi.org/10.1145/2505515.2505643.

[14] I. R. Brilhante, J. A. Macedo, F. M. Nardini, R. Perego, and C. Renso, “On planning sightseeing tours with TripBuilder,” *Information Processing and Management*, vol. 51, no. 2, pp. 1–15, mar 2015. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0306457314000922

[15] Y. Zheng and X. Xie, “Learning Travel Recommendations from User-Generated GPS Traces,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 1, 2011. [Online]. Available: https://doi.org/10.1145/1889681.1889683

[16] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, “Mining Interesting Locations and Travel Sequences from GPS Trajectories,” in *Proceedings of the 18th International Conference on World Wide Web*, ser. WWW ’09. New York, NY, USA: Association for Computing Machinery, 2009, pp. 791–800. [Online]. Available: https://doi.org/10.1145/1526709.1526816

[17] Z. Chen, H. T. Shen, and X. Zhou, “Discovering popular routes from trajectories,” in *Proceedings - International Conference on Data Engineering*, 2011, pp. 900–911.

[18] X. Chou, L. M. Gambardella, and R. Montemanni, “A Tabu Search algorithm for the Probabilistic Orienteering Problem,” *Computers and Operations Research*, vol. 126, p. 105107, feb 2021.

[19] W. Wisittipanich and C. Boonya, “Multi-objective Tourist Trip Design Problem in Chiang Mai City,” in *IOP Conference Series: Materials Science and Engineering*, vol. 895, no. 1. Institute of Physics Publishing, jul 2020, p. 012014. [Online]. Available: https://iopscience.iop.org/article/10.1088/1757-899X/895/1/012014https://iopscience.iop.org/article/10.1088/1757-899X/895/1/012014/meta

[20] E. Erbil and W. Wörndl, "Generating Multi-Day Round Trip Itineraries for Tourists," Tech. Rep., 2021.

[21] W. Wörndl, A. Hefele, and D. Herzog, "Recommending a sequence of interesting places for tourist trips," *Information Technology and Tourism*, vol. 17, no. 1, pp. 31–54, mar 2017. [Online]. Available: http://link.springer.com/10.1007/s40558-017-0076-5

[22] H. Iltifat, "Generation of paths through discovered places based on a recommender system," Ph.D. dissertation, Master's Thesis, Department of Computer Science, Technical University of˜. . . , 2014.

[23] "Geo-location APIs — Google Maps Platform — Google Cloud." [Online]. Available: https://cloud.google.com/maps-platform

[24] T. N. Nguyen and F. Ricci, "A chat-based group recommender system for tourism," *Information Technology and Tourism*, vol. 18, no. 1-4, pp. 5–28, apr 2018. [Online]. Available: https://doi.org/10.1007/s40558-017-0099-y

[25] K. Ikeda, G. Hattori, C. Ono, H. Asoh, and T. Higashino, "Twitter user profiling based on text and community mining for market analysis," *Knowledge-Based Systems*, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.knosys.2013.06.020

[26] C.-C. Hung, Y.-C. Huang, J. Yung-jen Hsu, and D. Kuan-Chun Wu, "Tag-Based User Profiling for Social Media Recommendation," Tech. Rep., 2008. [Online]. Available: http://www.flickr.com/

[27] A. Terttunen, "The influence of Instagram on consumers' travel plan-ning and destination choice," Tech. Rep., 2017. [Online]. Available: http://www.theseus.fi/handle/10024/129932

[28] M. Chen, A. Zheng, and K. Q. Weinberger, "Fast Image Tagging," Tech. Rep., 2013. [Online]. Available: http://tinyurl.com/9jfs7ut

[29] K. Balaji and K. Lavanya, "Medical Image Analysis With Deep Neural Networks," in *Deep Learning and Parallel Computing Environment for Bioengineering Systems*. Elsevier, jan 2019, pp. 75–97.

[30] A. Cufoglu, "User Profiling-A Short Review," Tech. Rep. 3.

[31] T. Tsiligirides, "Heuristic methods applied to orienteering," *Journal of the Operational Research Society*, vol. 35, no. 9, pp. 797–809, sep 1984. [Online]. Available: https://www.tandfonline.com/doi/full/10.1057/jors.1984.162

[32] P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden, "The orienteering problem: A survey," pp. 1–10, feb 2011. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0377221710002973

[33] K. Sylejmani, J. Dorn, and N. Musliu, "Planning the trip itinerary for tourist groups," *Information Technology and Tourism*, vol. 17, no. 3, pp. 275–314, sep 2017. [Online]. Available: https://link.springer.com/article/10.1007/s40558-017-0080-9

[34] P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden, "The orienteering problem: A survey," pp. 1–10, feb 2011.

[35] A. Rezaee Jordehi and J. Jasni, "Parameter selection in particle swarm optimisation: A survey," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 25, no. 4, pp. 527–542, dec 2013. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/0952813X.2013.782348

[36] A. Z. Şevkli and F. E. Sevilgen, "StPSO: Strengthened particle swarm optimization," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 18, no. 6, pp. 1095–1114, nov 2010.

[37] Z. Sevkli and F. E. Sevilgen, "Discrete particle swarm optimization for the orienteering problem," in *2010 IEEE World Congress on Computational Intelligence, WCCI 2010 - 2010 IEEE Congress on Evolutionary Computation, CEC 2010*. IEEE, jul 2010, pp. 1–8. [Online]. Available: http://ieeexplore.ieee.org/document/5586532/

[38] G. Kobeaga, M. Merino, and J. A. Lozano, "An efficient evolutionary algorithm for the orienteering problem," *Computers and Operations Research*, vol. 90, pp. 42–59, feb 2018. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0305054817302241

[39] X. Wang, B. L. Golden, and E. A. Wasil, "Using a genetic algorithm to solve the generalized orienteering problem," *Operations Research/ Computer Science Interfaces Series*, vol. 43, pp. 263–273, 2008.

[40] A. Gunawan, H. C. Lau, and P. Vansteenwegen, "Orienteering Problem: A survey of recent variants, solution approaches and applications," pp. 315–332, dec 2016.

[41] A. Santini, "An adaptive large neighbourhood search algorithm for the orienteering problem," *Expert Systems with Applications*, vol. 123, pp. 154–167, jun 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0957417418308182

[42] 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, jun 2014. [Online]. Available: http://link.springer.com/10.1007/s10732-014-9242-5

[43] M. G. Kantor and M. B. Rosenwein, "The orienteering problem with time windows," *Journal of the Operational Research Society*, vol. 43, no. 6, pp. 629–635, 1992.

[44] F. V. Fomin and A. Lingas, "Approximation algorithms for time-dependent orienteering," *Information Processing Letters*, vol. 83, no. 2, pp. 57–62, jul 2002.

[45] R. A. Abbaspour and F. Samadzadegan, "Time-dependent personal tour planning and scheduling in metropolises," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12 439–12 452, sep 2011. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0957417411005410

[46] F. Glover and M. Laguna, "Tabu Search," in *Handbook of Combinatorial Optimization*. Boston, MA: Springer US, 1998, pp. 2093–2229. [Online]. Available: http://link.springer.com/10.1007/978-1-4613-0303-9{_}33

[47] H. Tang and E. Miller-Hooks, "A TABU search heuristic for the team orienteering problem," *Computers and Operations Research*, vol. 32, no. 6, pp. 1379–1407, jun 2005. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0305054803003265

[48] K. Sylejmani, J. Dorn, and N. Musliu, "A Tabu Search approach for Multi Constrained Team Orienteering Problem and its application in touristic trip planning," in *Proceedings of the 2012 12th International Conference on Hybrid Intelligent Systems, HIS 2012*, 2012, pp. 300–305.

[49] X. Chou, L. M. Gambardella, and R. Montemanni, "A Tabu Search algorithm for the Probabilistic Orienteering Problem," *Computers and Operations Research*, vol. 126, p. 105107, feb 2021.

[50] E. Angelelli, C. Archetti, C. Filippi, and M. Vindigni, "The probabilistic orienteering problem," *Computers and Operations Research*, vol. 81, pp. 269–281, may 2017. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0305054816303331

[51] I. M. Chao, B. L. Golden, and E. A. Wasil, "The team orienteering problem," *European Journal of Operational Research*, vol. 88, no. 3, pp. 464–474, feb 1996.

[52] S. Muthuswamy and S. S. Lam, "Discrete particle swarm optimization for the team orienteering problem," *Memetic Computing*, vol. 3, no. 4, pp. 287–303, dec 2011. [Online]. Available: http://link.springer.com/10.1007/s12293-011-0071-x

[53] V. F. Yu, P. A. A. N. Redi, P. Jewpanya, A. Gunawan, V. F. . Yu, P. A. A. N. . Redi, P. . Jewpanya, V. F. Yua, A. A. N. Perwira Redia, and P. Jewpanyaa, "Selective discrete particle swarm optimization for the team Selective discrete particle swarm optimization for the team orienteering problem with time windows and partial scores orienteering problem with time windows and partial scores Citation Citation S," Tech. Rep., 2019. [Online]. Available: https://ink.library.smu.edu.sg/sis{_}research/4469

[54] R. Gama and H. L. Fernandes, "A REINFORCEMENT LEARNING APPROACH TO THE ORIENTEERING PROBLEM WITH TIME WINDOWS A PREPRINT," Tech. Rep., 2020.

[55] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, and D. Van Oudheusden, "Iterated local search for the team orienteering problem with time windows," *Computers and Operations Research*, vol. 36, no. 12, pp. 3281–3290, dec 2009.

[56] D. Miller, J. Sinanan, X. Wang, T. McDonald, N. Haynes, E. Costa, J. Spyer, S. Venkatraman, and R. Nicolescu, *How the World Changed Social Media*. UCL Press, feb 2016. [Online]. Available: www.ucl.ac.uk/ucl-press

[57] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 Million Image Database for Scene Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1452–1464, jun 2018.

[58] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4. IEEE, pp. 1942–1948. [Online]. Available: http://ieeexplore.ieee.org/document/488968/