2 Background Research and Literature Review

This section aims to position our study by describing the technologies we will use in this dissertation. In the first part, we provide an overview of the TTDP research area. We then describe methods of retrieving POIs from a location, existing user-profiling techniques and existing research regarding automatic tourist planning.

2.1 Recommender Systems

We introduce the term Recommender Systems (RS) as a solution for the TTDP and present background knowledge and relevant work that forms this thesis's basis. RSs hold use cases in diverse fields, such as e-commerce, media, and tourism [5]. However, in tourism, RSs offer tourists information in a unified and centralised manner, providing them with a plan for their trip [6–8]. Two domains develop current RSs for the TTDP solutions, which are; methods for obtaining tourist products (such as events and Point of Interests (POI)) and tour recommendation algorithms that create tourist trips [8] as shown in Figure 2. The following sections discuss related work in each field and discuss adding personalisation to RSs.

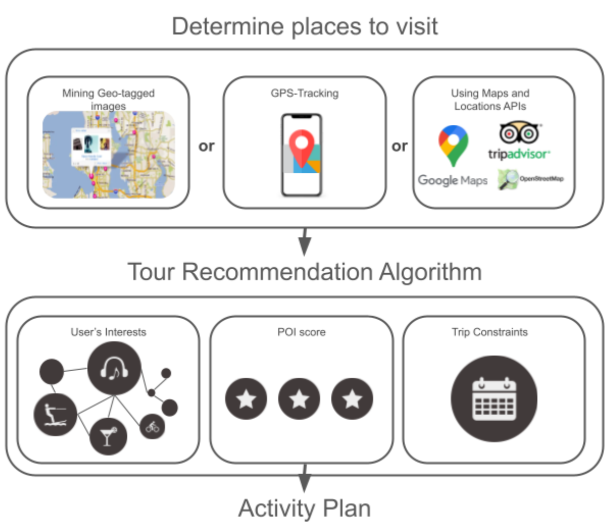

Figure 2: Recommender System Process in Tourism

2.2 Methods of retrieving travel products

Before producing an itinerary, RSs have to formulate a dataset of POIs from some data source. The proposed tour recommendation algorithm will then evaluate a guided path, route or itinerary from this dataset after understanding the users' implicit preferences such as the travel date and activity moderation. There are several ways to identify an appropriate data source representing real-life tourist trajectories.

Geotag mining: One approach is made by gathering tourist products by mining them from geotagged images of Location-Based Social Networks (LSBN) such as Flickr, Facebook or Twitter [2, 9–17]. Lim et al. [8] denote this process into three steps;

1. First, the application assembles an organised series of relevant photographs of the user's destination from the LSBN.
2. The application then maps these pictures with a list of popular places extracted from sites like Wikipedia.
3. Since the photos contain metadata, like the location and the timestamps; the application can calculate an approximate visit duration for each specific POI.

GPS-based data sources: The ubiquitous presence of smartphones and GPS-enabled devices has facilitated another approach to collecting trajectories [18–20]. A system can automatically gather the best POIs to visit based on other users' historical paths providing additional information such as the average time people spend at a specific POI and how many people go there. However, privacy issues are the main caveat towards this approach since it requires people to share their location constantly and publically [8].

Prebuilt dataset: The most straightforward method is done by self-defining the POIs or gathering them from a dataset such as the TSPLIB95 [1]. Manually collecting travel products provides precision and a better understanding of the itinerary that the algorithm will generate. However, the algorithm would be dataset-specific testified and personalised towards what the authors of the dataset think are the best POIs to visit in a location [21–23].

[1]Sample instances for travelling salesman Problem: http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/

Maps APIs: A prompt and accurate strategy towards gathering essential places in the vicinity is using Mapping & Location APIs such as Foursquare, Google or TripAdvisor. Wörndl et al. [24] use this approach and build a dataset of prominent POIs by querying their API with the user's desired location. In return, they receive a sequence of places and information about each site, including its category, other user's ratings, opening hours, coordinates and helpful additional information to use as criteria for the itineraries. However, the API does not return the average amount of time people spend at a specific POI.Wörndl et al. [24] solve this issue by adding a fixed time constant for each category with a variable dependant on the POI's score. For example, suppose a restaurant's time constant is 45 minutes, and the chosen restaurant has a high score (based on its rating and user's characteristics). In that case, the time spent at the restaurant will increase by an additional 15 minutes. A significant advantage of using this approach is that the vast number of POIs that these endpoints return. According to Google's website, the API contains up to 200 million places with 25 million updates daily which an application can achieve with a few REST requests [25, 26].

2.3 User Profiling for Travel Preferences

In 2018, Lim et al. [11] demonstrated how implementing personalisation in their algorithm, which they called PersTours, helped their algorithm to portray real-life scenarios more accurately. The authors built a system where the tourist's level of interest in a specific category is dependant on their time spent at such POIs, relative to the average user. They gathered information from the user's past trips from the social media platform Flickr.

Nguyen et al. [27] produced an android chat application called STSGroup that gathers user's preferences and resolves conflicts between tourists by understanding the messages sent in a group chat. They provided an example of students travelling to South Tyrol (Italy), which gathered information such as the users' mood and recommended POIs from their conversations. Other users in the group chat rate their suggestions through a voting system as the system uses raking lists and logistics to calculate the ideal group preferences in the background.

The average internet user has gone from being a passive content absorber to a content producer through the rise in social media [28]. TTDP RSs can use this as an advantage and provide a fully automated activity plan based on the user's characteristics. This section describes several methods for user profiling and information gathering from the user's social media.

2.3.1 User Profiling based on natural language processing

In 2013, A sentiment analysis by Ikeda et al. [28] based on a hundred thousand Japanese user profiles managed to perform a demographic estimation. This study shows how social media posts can be helpful to gather information about a user, and in fact, Hung et al. [29] demonstrated a user profiling technique based on tag correlation.

2.3.2 User Profiling based on images

Instagram has a significant influence on the tourism industry. Sharing photos of amazing sights and landscapes have impacted the way people choose their POIs [30]. Therefore, a system that uses tourist's social media photos could impact automatic user preference gathering.

Chen et al. [31] produced a system for automatically retrieving tags from images and incomplete tags called *FastTag*. The algorithm can be trained in $O(n)$ time and uses two simple linear mappings. Figure 3 shows an example of an input image used alongside the incomplete input tags *snow, lake, feet*. Given these two inputs, the algorithm produced the following additional tags, mountain, water, legs, boat, trees.

Figure 3: Example of an image input for the FastTag algorithm

Images classification techniques could provide a preference gathering system. Deep neural networks play an essential role in this field of Image Classification [32, 33]. We will describe how this technology could gather information for a tourist's user preferences in the upcoming sections.

2.4 Travel Planners for both individual and grouped travellers

This section will focus on breaking down existing meta-heuristic approaches, notably swarm-based, trajectory-based and evolutionary algorithms, towards the vast number of variants of the TTDP. Gavalas et al. (Gavalas 2014) classify these variants into two; Systems that produce a single route and systems that can handle multiple days.

2.4.1 Single Route Problems

The Orienteering Problem (OP), introduced by Tsiligirdes [34], in observance of the sport, orienteering, is the foundation of single route planners. Figure 4 shows the variants of the OP that will be discussed in this section and how they relate with the original problem. TTDPs [5]. Vansteenwegen et al. [35] describe OP as a travelling salesman problem with profits. In OP, several nodes V representing POIs where $\{V = v_1, \ldots, v_n\}$, are designated in a space G and edge set E with a starting and an ending point. Each node holds a score s calculated from the tourist's constraints and the distance from node v_i to v_j. The objective is to visit a subset of these locations, maximising the s by obeying the tourist's constraints and minimising the travel time [36]. Vansteenwegen et al. [3] mathematically formulate the OP and the methodology contains a mathematical formulation of this instance of the TTDP.

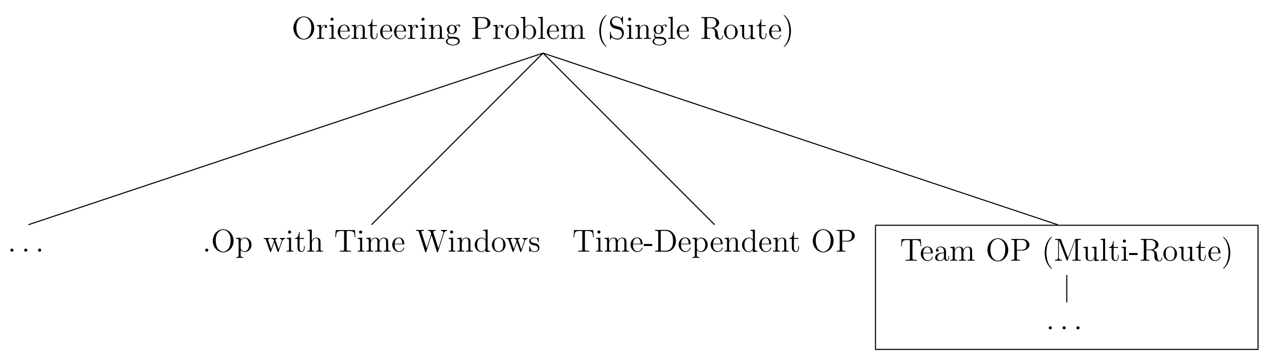

Figure 4: Graph showing the variants that will be discussed in the section

Particle Swarm Optimisation-based (PSO) systems provide prevalent OP solutions with fast computing time. These are bio-inspired meta-heuristic approaches in which, in the TTDP, a particle represents a travel path. The particles aim to optimise themselves by communicating with each other and using their velocity property to move to the most optimal solution [37]. In 2010, Sevkli et al. [38, 39] tested out two PSO variants: Strengthened Particle Swarm Optimization (StPSO) and Discrete Strengthened Particle Swarm Optimization (DStPSO). These two algorithms introduce pioneering particles, which first perform a local search-based technique called Reduce Variable Neighborhood Search (RVNS) between all the particles and then assign a random velocity. These PSO algorithms obtains either the best or competitive solutions compared with other algorithms such as ant colony and genetic algorithms when tested on the Tsiligirides [20, 34] dataset.

There are numerous Evolutionary Algorithms (EA) proposed to solve OP. [40, 41].

EAs are algorithms based on natural evolution which use a fitness score to get to the best solution of a problem, in this case, the TTDP [42]. A novel approach in 2018 by Kobeaga et al. [40] was able to find ambitious solutions for over 400 POI nodes using the steady-state genetic algorithm. The algorithm also implements a local search, which aims to reduce travel time.

In 2019, Santini et al. [43] introduced a heuristic algorithm based on adaptive extensive neighbourhood search. They evaluated their system by comparing it with Kobeage et al.'s EA. The results showed that both algorithms find suitable solutions in a reasonable amount of time. However, the EA finds slightly more suitable solutions, while the extensive neighbourhood search has a lower average gap.

In real-life scenarios, POIs have time constraints that allow them to be visited only during specific hours, such as opening and closing hours or public holiday constraints. Traditional OP is not able to cater for such problems. A single route variant of the OP which solves these issues is the Orienteering Problem with Time Windows (OPTW) [44].

Kantor et al. [45] provided the first attempt towards the OPTW [3]. They developed two heuristics; Insertion and depth-first search. The former algorithm solves the path by selecting a POI with the highest score over-insertion cost incrementally. On the other hand, the depth-first search algorithm gathers parallel tree-based solutions simultaneously and iteratively adds new POIs as long as they follow a set of constraints. Their evaluation showed significant improvements of the second algorithm over the insertion. Most of the novel solutions of OPTW are for the multiple route problems discussed in the upcoming sections.

When travelling between two POIs, the travel time may depend on certain variable time constraints such as the traffic levels and waiting time [5]. The Time-Dependent Orienteering Problem (TDOP) introduced by Fomin et al. [46] is the single route variant of OP, which considers these scenarios since traditional OP does not [42]. In 2011, Abbaspour et al. [47]provide a solution for the Time-Dependent Orienteering Problem with Time Windows, which combines the two previously mentioned OP variants (TDOPTW). They propose two adaptive genetic algorithms and multi-modal shortest pathfinding evaluated in the city of Tehran.

In 1998, Glover et al. [48] introduced a meta-heuristic approach called the Tabu Search, and several RSs used this algorithm [49–51]. This optimisation technique is advantageous when trying to escape from a local optimum [51]. A novel approach by Chou et al. [51] aims at tackling the Probabilistic Orienteering Problem (POP) [52], which is another variant in which every path contains a cost, and the system can access every node within a specific probability. Moreover, each node will be available for a visit only with a certain probability. When evaluated, a simple tabu search could compete with complex meta-heuristics showing its potential in this field.

2.4.2 Multiple Route Problems.

The RSs available from what we discussed in the previous sections can only generate a single efficient path for a tourist's holiday. The Team Orienteering Problem (TOP) [53] is a variant of the OP, which allows for solving the TTDP with multiple days [36]. The system generates a full itinerary for the tourist, with a maximum total score of all routes [5].

Several Recommender Systems use PSO-based solutions to solve the TOP [22, 54, 55] Muthuswamy et al. [54] developed a discrete version of the PSO (DPSO) which can generate n routes where n can be between two to four. The algorithm consists of two procedures; Random initialisation of n-1 routes with a calculated initialisation of the nth route based on partial randomness and the current score divided by the current distance of the particle. Updating the current velocity of each particle. The particles use RVNS and 2-opt techniques to communicate with each other as local search techniques. The authors evaluated their work by comparing the algorithm to seven TOP heuristics in which DPSO performed competitively across all applied benchmark data sets [44].

A few years later, Dang et al. wrote another PSO inspired algorithm (PSOiA) for the TOP. They evaluated their work using an interval graph model, which showed how to examine a more extensive search space faster [42].

Besides swarm-based algorithms, A RS by Sylejmani et al. [50] used the trajectory-based tabu search to solve a Multi Constrained Team OPTW. Their system followed three steps in order to generate an activity plan: a new activity is added as a node to the trip using *Insert*, a node is exchanged with a new activity using *Replace* and two nodes swap with each other using *Swap*.

Several RSs also use PSO-based solutions in novel approaches. For example, in 2019, Yu et al. [55] developed a system for the Team OPTW variant based on selective DPSO. In 2020, Wisittipanich [22] presented an application of a metaheuristic called Global Local and Near-Neighbour Particle Swarm Optimization (GLNPSO). Wisittipanich evaluated their results using LINGO, an optimisation program and showed excellent results.

Recently, Gama [56] et al. compared their reinforcement learning with top-performing heuristics of the TOP, such as Vansteenwegen's Iterated Local Searchl [57]. The authors use a Pointer network as this has been previously to solve TSP-related problems. This study opened a different way of tackling the TTDP and achieved production-level performance and inference times. An advantage of this approach is that the results are probabilistic. So it is possible to retrieve the top-n solutions and use them in a more generalised route recommendation system.