Computer Science and Information Systems 16(1):131–154 https://doi.org/10.2298/CSIS171212011S

Estimating point-of-interest rating based on visitors geospatial behaviour

Matej Senožetnik[1,2], Luka Bradeško[1,2], Tine Šubic[1], Zala Herga[1,2],
Jasna Urbančič[1], Primož Škraba[1], Dunja Mladenić[1,2]

[1] Institut Jožef Stefan,
Jamova 39, 1000 Ljubljana
E-mail: name.surname@ijs.si
[2] Jožef Stefan International Postgraduate School,
Jamova cesta 39, 1000 Ljubljana

Abstract. Rating of different services, products and experiences plays an important role in our digitally assisted day-to-day life. It helps us make decisions when we are indecisive, uninformed or inexperienced. Traditionally, ratings depend on the willingness of existing customers to provide them. This often leads to biased (due to the insufficient number of votes) or nonexistent ratings. This was the motivation for our research, which aims to provide automatic star rating. The paper presents an approach to extracting points-of-interest from various sources and a novel approach to estimating point-of-interest ratings, based on geospatial data of their visitors. Our research is applied to campsite dataset where the community is still developing and more than thirty percent of camps are unrated. Our study use case addresses a real-word problem of motorhome users visiting campsites in European countries. The dataset includes GPS traces from 10 motorhomes that were collected over a period of 2 years. To estimate star ratings of points-of-interest we applied machine learning methods including support vector machine, linear regression, random forest and decision trees. Our experimental results show that the duration of visit, which is a crucial part of the proposed approach, is an indicative feature for predicting camp ratings.

Keywords: machine learning, geospatial analysis.

1. Introduction

Crowd-sourcing services and product ratings are a great way to allow the community to comment on their experiences with various services or products. As such, it is not surprising that integrated rating and review systems have become extremely popular, leading to the rise of sites (like GoodGuide and Yelp) dedicated to specific products and services. Recommending hotels and restaurants based on text reviews is a commonly addressed research topic [11,15,16,19]. In this research we focus on addressing a problem of estimating campsites rating based on their available facilities and data about visitors geospatial behavior. In particular, we focus on *motorhomes* which are visiting campsites and other point-of-interests. The data have been collected over more than 2 years from 10 motorhomes traveling across Europe.

Motorhome community is still in the process of development. Motorhome companies have been seeking ways for the motorhomes to be able to dynamically assist the user in discovering new destinations and advise them by taking into account different aspects, such as availability of electricity, fresh water and other resources. With the popularity of motorhomes on the rise (registration of motorhomes has increased by 17.9% in 2016 in comparison to 2015 [9]). We applied our research to their users' geospatial patterns and analyzed their common accommodations - camps, service areas, etc.

The main contribution of this article is a new approach to automatically estimate (predict) point-of-interest ratings based on user behavior and camp characteristics. The next contribution is evaluation of the proposed approach in real-world data. For predicting camp rating we compared several machine learning algorithms including linear regression, k-NN, random forest and SVM. To be able to perform the analysis, we have collected two years of location data for motorhomes that traveled around Europe. The third contribution of our research is building the definitive database of points-of-interest for campers and along with the analysis providing it as a service to the motorhome community.

The remainder of the paper is organized as follows. Section 2 gives an overview of the related work. In Section 3 we provide a detailed description of architecture and interaction between components. The data description is presented in Section 4. Section 5 presents the approach used to check the hypothesis. Finally, we conclude the paper and describe some of the future work directions in Section 6.

2. Related work

To the best of our knowledge, the problem of predicting camp rating based on location history was not addressed before. However, similar works on different datasets and using different methods exist and had been tried before. This section provides a brief overview of that related work. Understanding and predicting what users really want and what suits their needs has become an important and very popular research topic. A lot of work has already been done on predicting ratings for hotels, restaurants, etc. The most popular datasets for research and building personalization applications are provided by Yelp and HRS.com (major European platform for hotel reservations). If users rate places highly, then algorithm would recommend places similar to those that the user gave high ratings. This can go beyond overall ratings, by taking into account advantages of multi-criteria ratings and improving recommendation accuracy compared to single-rating recommendation approaches [10,20]. We could not perform such kind of analysis on our data, because we do not have information on specific users rating of the location. However, our system is designed to work with the data like this, once available.

Other useful information could be extracted from different reviews of the same place. The most common approach is to use sentiment analysis to evaluate hotel ratings [12,20]. There have also been studies dealing with the correlation between hotel star-rating and review rating [18].

Recommendation systems collect information based on the preference of users on different items, such as movies, jokes, books, hotels, restaurants, camps. Some recommendation systems went further and additionally take into account the location as an important component of the user context [22,25,26]. As opposed to our approach, the usual recommendation systems rely on the availability of large amounts of data.

Tiwari and Kaushik [27] propose a rating method for points-of-interest based on fuzzy logic which can take a location stay time, user categorization (native, regional and tourist) and weighted frequency (number of times when the visitor enters a place and stays there for more than a defined time threshold) as inputs. They also propose to categorize users into five categories (global, national, regional, local and native) [28] reporting that user categorization has the impact on stay time. This method returns a rating between 0 and 5. The main issue, as explained by the authors, is that only 0.5% of places from their dataset of 26,807 places have any ratings. Secondly, it is hard to compare ratings between places. If places are internationally known, they have the significantly higher volume of ratings than small places which are more regionally/locally known. Another potential problem is that ratings could be biased by the deliberate promotion of a location as a marketing strategy. However, to measure any impact of a rating method on the users, we would have to include the users in the evaluation.

3. Proposed Approach and Framework Description

Figure 1 shows an overview of the proposed architecture that consists of four main components: data capture, Mobility patterns, analysis and API. The *Data capture* component primarily captures GPS locations from motorhomes, but it also takes care of building point-of-interests (POI) data (*Building POI*). *Building POI* prepares OpenStreetMaps (OSM) files for camp data extraction and collects information about camps from publicly available online resources. Main two sources of the data for our approach are real-time GPS coordinates stream in WGS84 format coming from connected motorhomes, and a POI database which is matched to the stream, to find visited POIs for each user (motorhome). This process consists of cleaning the data and transforming it into manageable form for further analysis. The *Mobility patterns* is a service for processing raw GPS data and performing clustering, global location detection and next place prediction. *Analysis* component aims to determine the correlation of ratings to time spent on location, predicts star rating and analyze number visits to camps.

The main function of the *API* service is to enable access to analyzed data to external parties; in case of camp data to share it with the motorhome community and Optimum mobile application. Optimum is an EU project which aims to improve transit, freight transportation and traffic connectivity throughout Europe [5]. Within the project one of case studies is based on motorhome and their users [6].

3.1. *Mobility patterns* component

Mobility patterns service is capable of accepting and parsing continuous GPS data (Definitions 1 and 2) from various sources, like mobile phones and other devices. It also collects vehicle GPS data from motorhomes. In addition to collecting the data it is an analytics service which classifies raw GPS coordinates into the appropriate paths and staypoints [29]. Staypoints (Definition 3) are classified into global locations (Definition 4) and paths into global routes. Based on the historical data, the service can also predict next user location and detect anomalies in paths. Here we only use the information about visits (staypoints) and resulting global locations classified as camps. We provide definitions related to staypoints in a similar way as in [24,30] .

Definition 1 (GPS point) *GPS point p is a pair of longitude (lng) and latitude (lat) values which can be formally defined as $p = (lat, lng)$.*

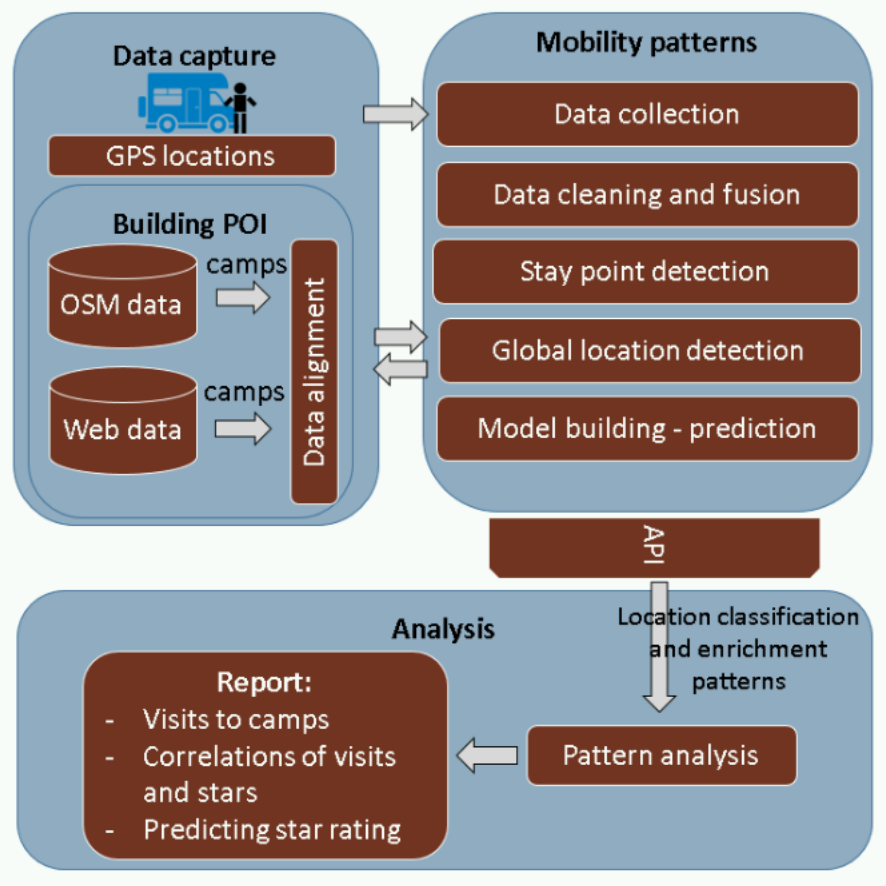


Fig. 1. The architecture of our approach to estimating (predict) points-of-interest rating based on visitors behavior.

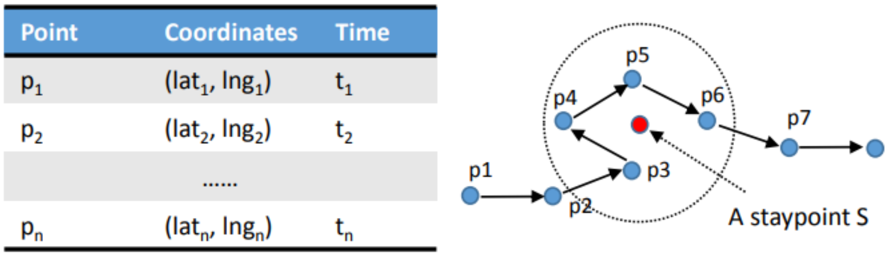
Definition 2 (GPS trajectory) *GPS trajectory is a spatio-temporal sequence which can be formally defined as*

$$Traj = \{(p_1, t_1), (p_2, t_2), \ldots, (p_n, t_n), \}$$

where $p_i, i = 1, \ldots, n$ *are GPS points and* t_i*,* $i = 1, \ldots, n$ *monotonically increasing time values.*

Definition 3 (staypoint, path, activity) ***Staypoint*** *S is a limited geographic area which is shown in Figure 2 at which the trajectory stayed for a certain time. More formally, a staypoint S is a set of pairs* $S = \{(p_i, t_i)\}_{n \leq i \leq m}$*, where*

1. $t_m - t_n > T$*, for a threshold duration* T
2. $Distance(p_i, p_j) < D$*, for each* $i = n, \ldots, m$*,* $j = n, \ldots m$*,* $i \neq j$*, for a threshold distance* D*.*



Point	Coordinates	Time
p_1	(lat_1, lng_1)	t_1
p_2	(lat_2, lng_2)	t_2
	......	
p_n	(lat_n, lng_n)	t_n

Fig. 2. GPS trajectory and staypoint

3. *$S_i \neq S_j$ for $p_i = p_j$ and $t_i \neq t_j$ (staypoint is time dependent).*

*Staypoint is a state of a trajectory. A section between two consecutive staypoints defines a **path**. Path is a sequence of pairs $P = \{(p_i, t_i)\}_{n \leq i \leq m}$, such that $(p_{n-1}, t_{n-1}) \in S_j$ and $(p_{m+1}, t_{m+1}) \in S_{j+1}$. A common name for paths and staypoints is **activity**.*

Several algorithms have been used for staypoint detection [31,32,23]. We use our own modification which was originally described in [14] and is uses a two-stage approach, first clustering GPS coordinates by employing a staypoint detection (SPD) algorithm grouping together GPS coordinates which are located inside a parameter given by threshold distance D and visited within the time period T. In our implementation we utilized thresholds of 50 meters for radius (distance) and 5 minutes for the duration. To handle potential errors in the GPS coordinate reading after performing the clustering we eliminate locations that are detected as outliers (e.g., a big jump in the distance back and forth over a short time) and merging staypoints if appropriate. The output of our 2-pass SPD algorithm is a list of cleaned activities either staypoints or paths.

Our users (motorhomes) $\{u_1, \ldots u_n\}$ collect GPS points and produce unique trajectories which are clustered into staypoints. Some users' staypoints overlap in space. We are interested in grouping those staypoints that correspond to the same geographical location together into a common *global location*.

Definition 4 ***Global location*** *is a circular geographic area defined by location center (point) and a radius. It is not time-dependent and it holds a semantic meaning.*

Each staypoint is assigned to exactly one global location. For each global location we store the center of the location (GPS coordinate pair) and based on included staypoints we calculate of a visit statistics, time and date of the first visit, the last visit and the average duration of visit.

3.2. Data capture component

OpenStreetmaps OpenStreetMap (OSM)[13] provides a free geographic database of the world. While this started with mapping streets, it has already gone far beyond that, and now includes footpaths, buildings, camps, parkings, and many other geographic properties. For downloading OSM data we are using a service called Geofabrik [3]. Geofabrik provides pre-built shape files, maps and map tiles, allowing us to bypass some of the manual extraction and preprocessing work. The data is organized by regions that are regularly updated and saved in PBF format.

We focus on the camps in the European Union, however the approach can use any geographical region worldwide. The data is stored as a collection of SQL tables for points, polygons, roads and lines which enables easy querying and data processing (using Osm2pgsql [7] to convert OSM data to PostGIS-enabled PostgreSQL databases). We were mainly interested in point and polygon tables which contain camp data that can be retrieved by the following query:

```
SELECT * FROM table_name WHERE ((tags->'tourism')='camp_site'
OR (tags->'tourism') = 'caravan_site');
```

In OSM, camps are defined by the *camp_site* tag, while *caravan_site* is generally reserved for short-term parking of motorhomes. Besides spatial point or polygon data, OSM provides other semantic information such as location name, fee, opening hours, phone, website, address and others [4]. However,the presence of this data is very inconsistent and varies from record to record.

Additional data We used data provided by different online forums [1,2] and we composed two datasets. One consists of regional datasets (consisting of Slovenia and its neighbor countries) and the other is a general European dataset.

European dataset consists of over 20,000 records of motorhome stops, separated into following categories: motorhome parking, service areas and motorhome-friendly campsites. Service areas are places, where motorhomes can discharge wastewater and chemical toilet and refill their fresh water supply. Motorhome friendly campsites need to have the possibility to discharge wastewater and chemical toilet and to supply fresh water, as well as allowing overnight stays. Motorhome parkings are areas appropriate for motorhomes and are further divided into multiple subcategories such as mixed parking, private parking near a restaurant, near a marina, motorhomes only, etc...

The Regional dataset is focused on Slovenia, Croatia, Bosnia and Herzegovina, Montenegro and Serbia. Every camp record contains three categories of detailed information:

- Camp is suitable for: families with children, dogs, surfing, rowing, bicycling, etc.
- General info about camp such as open hours, good shaded pitches, location near the sea, outdoor swimming pool, camp at the spa, tents for rent, caravans for rent, mobile homes for rent, bungalows, children playground, animation for children, sanitary facilities for disabled people, baby showers, service station for campers, shop, restaurant, paid wireless internet access, washing machine, mooring for boats, etc.
- Sports and other activities available, such as tennis, volleyball, rent a bike, rent a boat, diving center, table tennis, basketball, football, mini golf, rowing, surfing, climbing wall, etc.

The resulting datasets are merged with OSM data which we consider the main data source. We tried to match coordinates with corresponding position or polygon in OSM. Merging function, shown in Algorithm 1 iterates through all of the additional camps and attempts to find the appropriate camp within a certain radius (delta) inside OSM. If it finds more than one suitable OSM camp, we update the records for all polygons in the vicinity.

Algorithm 1: Algorithm for merging OSM and web data.

```
function Merge ($webCamps$,$\Delta$);
Input  : All camps which are obtained through from web forums. $\Delta$ is distance in meters
for camp in webCamps do
    osmCamps ← findCampInTheDatabase
      ($camp.latitude$,$camp.longitude$,$\Delta$);
    if len (osmCamps) $\geq 1$ then
        updateAll (osmCamps);
    else
        insertNewCamps (osmCamps);
    end
end
```

If, however, no camp is found in a range we conclude that OSM data does not contain it yet and we add a new camp record.

Cleaning data In this step we remove possible duplicates which occur due to combining OSM with other data sources. Some of the camps cover a very large area and may even cross national borders, which causes them to exist in more than one country file and are then present as multiple polygons when combining country data. Because of this, we need to merge these duplicates into single record using PostGIS. Due to that, we need to merge these polygons into a single record. Additionally, we detected that some camp records were not completely consistent, appearing as either points or polygons. Such conflicts must be resolved manually. For this we generate a rule file by hand, where we define which polygons or points will be deleted and which polygons or points are to be merged into one record.

Data alignment This step is necessary in order to improve our analysis. In some cases multiple *Mobility Patterns* global points were identified on one OSM camp polygon. We solved this issue by merging global points into one location. Hence, our aim is to find appropriate point matching between camps and *Mobility patterns* global location. Our assumption is that if a motorhome user is at (or close to) a global location then we treat this as a camping place. Our method goes through all global locations and in each iteration tries to find the containing polygon or point within 100 or 120 meters of distance.

3.3. Data analysis methods

In this section we describe methods used in our analysis. In data processing stage using *Mobility patterns*, we calculated a number of visits to each camp and histograms about arrival and departure times of motorhomes. With *regional* and *European* datasets, we applied Pearson correlation analysis in order to understand correlations between the features (including target value). Then we used machine learning methods such as linear regression, random forest, decision tree and SVM on *European* dataset in order to predict the star rating. We also fitted a model with *the regional* dataset, but the outcome was not promising. Due to mismatching attributes in the datasets, we were unable to perform the merging into a single unified dataset, since that would cause either a certain amount of data loss or empty values in some records.

Visits to the campsite is a simple analysis which counts the number of visits to a specific campsite, provides time and date of first and last visit, and its duration. The practical example is shown in Section 5.2. This analysis is provided by *Mobility patterns*, where each campsite is represented by one global location.

Pearson correlation analysis In order to understand the relationship between attributes we used Pearson correlation coefficient, which is calculated as follows:

$$r = \frac{n \sum xy - (\sum x \sum y)}{\sqrt{[n \sum x^2 - (\sum x)^2][n \sum y^2 - (\sum y)^2]}} \tag{1}$$

where n represents the number of samples. Pearson correlation depicts a linear relationship between two variables. We checked the correlation of all possible feature pairs (variables x and y), including target values. From these correlations, we built a heatmap which is shown in Figure 7. In general heatmap for correlations has values between -1 and 1. In our case the more intense is color the higher is correlation among attributes. The pros are if we have a small number of attributes it can be easily interpretable and we gain more insight into the data. When we have many attributes it is hard to show everything and we may have an issue of interpreting the heatmap.

Machine learning methods for prediction To predict star rating we used the following machine learning methods implemented in *Scikit learn* [8]:

- **Decision tree** is an easy to understand model used for classification and regression, with multiple available decisions and outcomes included in the tree. To avoid data over-fitting we tune the depth parameter. We used the algorithm for classification and regression.
- **k-Nearest Neighbors**(k-NN) algorithm for classification and regression is a simple method of making predictions by searching through the entire training set for the *k* most similar instances and summarizing the output variable for those *k* instances. We used the algorithm for classification and regression purposes.
- **SVM** using linear, polynomial or radial basis kernel. In our experiments we used radial basis kernel. In general it performs well with non-linear boundary depending on the kernel, and handles high dimensional data. It is used for classification and regression.
- **Random forest** is an alternative to decision trees and reduces variance by generating a whole collection of trees. However, the results are not easily interpretable. It is used for classification and regression.
- **Linear regression** is a standard method commonly used when dealing with numeric features. It is used for regression.
- **Logistic regression** is simple model for classification rather than regression.
- **Baseline** We calculated a mean of target values for regression and take majority class for classification.

Feature selection We used the method called stepwise forward selection, which finds the subset of features that produce the best results. The algorithm first processes every feature and evaluates the result to find the feature with the best performance. On every iteration it adds to the subset a feature that produces best results in combination with previously added features. Each step minimizes RMSE error. This process is repeated until we either run out of feature or the addition of new features does not improve the result anymore.

Performance evaluation For the evaluation we used 10-fold cross-validation. To asses the errors and performance, we used the following measures:

- **Accuracy** is the fraction of correct predictions over n calculated as

$$accuracy(y, \widehat{y}) = \frac{1}{n}\sum_{i=0}^{n-1} \mathbb{1}(\hat{y_i} = y_i) \tag{2}$$

  where $\hat{y_i}$ is the predicted value of the i^{th} sample, y_i is the corresponding true value and n is the number of samples.
- **F1** is a weighted average of the precision and recall which is calculated as

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}. \tag{3}$$

  Precision and recall are defined as $precision = tp/(tp + fp)$ and $recall = tp/(tp + fn)$, respectively, where tp are true positives, fp represents false positives, and fn false negatives. F1 score, precision and recall all take values between 0 and 1, where 0 is the worst and 1 is the best value.
- **Root mean square error** is calculated as

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=0}^{n-1}(y_i - \hat{y_i})^2} \tag{4}$$

  which is the square root of the average of squared differences between prediction $\hat{y_i}$ and actual observation y_i. Values can range from 0 to infinity and it is a negatively-oriented score, which means lower values are better. Also, this metrics penalizes large errors more. This is an appreciated property for us since we want to avoid large error, whereas small errors are acceptable because the target value could be biased.
- **Coefficient determination** is calculated by equation

$$R^2 = 1 - \frac{\sum_{i=0}^{n-1}(y_i - \widehat{y_i})^2}{\sum_{i=0}^{n-1}(y_i - \overline{y_i})^2} \tag{5}$$

  and reflects the proportion of the variance in the target variable that is predictable from the explanatory variables. Its values range between -1 and 1. The higher the value the better the result.

3.4. API

To provide the access to collected campsite data to outside users and easier utilization in our internal applications and algorithms, we use a REST API backed by PostgreSQL database, which contains records of campsites and allows querying and filtering by different properties - campsites near a certain point, campsites in a specific country, campsites with specific facilities, etc. Some of this data is already being used by an application in development, intended for multi-modal route-planning application (developed in the scope of Optimum project) that features smart notifications and proactive recommendations to encourage sustainable transport behavior.

4. Data description

This section presents the dataset, which we used to train the models and predict star ratings of the campsites.

4.1. Feature description

Campsite features We represented each campsite as a feature vector where features describe static properties of the campsite. Some features are the boolean type, where the position in the vector represents the property and its value indicates whether the place has this property or not. Category of location is a categorical attribute and continuous attributes are the location near water, distance from the city, camp categorization, star rating and number of votes. Features described below represent the union of features from both datasets.

The specific properties (features) are:

- **Terrain** (hardened, grass). Basic information about camp; whether it is covered with grass or hard material (e.g. concrete) or both.
- **Illumination.** Information whether camp has electric lighting or not.
- **Security.** Information whether camp has taken security measures, like camera coverage or security guards.
- **Facilities** (restroom, service station, dustbin, shower, playground, dogs not allowed, washing/dryer available, wheelchair accessible, camping-like behavior, internet services, wastewater discharge, chemical toilet discharge, fresh water, electricity). The facilities are important for every camp and more facilities camp has the better it is.
- **Possible nearby activities** (swimming, good fishing possibilities, bicycle tours, hiking tours, winter-sports). The best camps provide not just facilities but also activities which are important for longer stay.
- **Category of location.** Motorhome POIs are divided into three groups: service station, parking, campsite.
- **Location near water.** Distance in meters from the POI to the nearest water.
- **Distance from the city.** Distance in meters from campsite to the center of the nearest town.
- **Camp categorization.** Camp categorization is rating between 1 and 5 provided by organization.

- **Star rating, number of votes.** Ratings are provided as a number between 1 and 10, where higher number means better rating. *Regional* dataset additionally provided separate star ratings for campsite position, tidiness, tidiness of toilets, sport and other activities and value for price.

Temporal (*Mobility patterns*) features The following data was retrieved from *Mobility patterns* service. In the last two years the service detected more than $14,000$ staypoints, where the staypoints consisted of more than $7,361$ unique global locations. Global locations are not necessarily POIs as they include stops like border control, gas stations and other general service areas. This was general information about stays for motorhome, but we are showing in Table 3 just stays in campsites, parkings and service areas. Exact numbers of those places are described in Section 4.2 in Table 3. We collected the duration of each place visit; all other attributes are derived. The specific feature items include:

- **Average duration**: average time of visit at specific location (POI).
- **Min time**: minimum time spend on specific POI.
- **Max time**: maximum time spend on specific POI.
- **Time distribution**: histogram of POI visits durations. Each duration interval represents one feature vector. Duration intervals: (0h,1h], (1h,3h], (3h,6h], (6h,9h], (9h,12h], (12h,15h], (15h,18h], (18h,21h], (21h,24h], (24h,248h]).
- **Time distribution percent:** similar like above, but features represent percentage of specific visits, not real number.
- **Seasons** (winter, spring, summer, fall): Number of visits of specific POI in specific season.
- **Proportion of seasons** (winter, spring, summer, fall): Similar to above - proportion of visits in specific season.
- **Average number of days:** we calculate average number of days motorhomes stayed at the specific location. Our assumption is that the more interesting activities camp has to offer the longer the average stay.
- **Average number of nights:** number of nights spent at camping place.
- **Number of visits:** number of all campsite, parking or service area visits by whole motorhome group.

Complex features Time spent at the location and number of its visits could be dependent on the district that it's located in. For example, some places are much more touristic and near really interesting locations, while other places could be of really good quality, but away from the tourists points of interests. Derived features were intended to help us better understand differences between areas. After performing several tests, we decided on a radius of 10 kilometers. Complex features are derived from Mobility patterns and basic camp features [17,21]. We calculated:

- **Competitiveness** as the proportion of POIs in 10km radius whose categories are the same as the category of the target location.
- **Density** as number of POIs in the neighborhood.
- **Neighbor entropy** measures the heterogeneity of POI categories in the neighborhood.
- **The Popular area** as the total number of check-ins for POIs in the neighborhood.

4.2. Type of resting place distribution analysis

European dataset provides multiple different categories of resting places such as:

- **Campsite**: these are places where only motorhomes can park. Campsites are in general destinations for motorhomes users. Usually they are required to facilitate wastewater discharge, supply with water and electricity.
- **Motorhome parking**: reserved spots at a general parking lot, where motorhomes often park among cars.
- **Service area**: places in the parking lot of a restaurant, hotel or cafe. The facilities are not necessarily there and overnight stay is not allowed at these types of resting places.

Cleaned *European* dataset consists of $23,971$ service areas, motorhome parkings and campsites. Figure 3 shows that the most common type is motorhome parking with $15,859$ parkings. There are $7,056$ campsites and the dataset contains $1,056$ service areas. *Mobility patterns* detected 528 visits in different service areas, parkings and campsites. Out of these visits 249 were to parkings, 57 to service areas and 222 to campsites. The analysis (Table 3) presents the average time motorhomes spent at each type of the place. At the service areas campers in general spent 0.77 hours, at campsites 13.9 hours and at parkings 8.32 hours. This shows that there is a difference in the time spend for different types of POI. We can notice that the average time spent at campsites is 13.9 hours, which is lower than one would expect. However we understand that campsites could be used as the transit points to an end location or the user may leave the campsite to visit nearby locations during the day, which in our case is considered as a new visit to the campsite. This is not ideal in the context of our research as it decreases the time spent at a campsite and increases the number of visits. We plan to address such daily exits in the future work. Notice that the proposed approach does not distinguish between regular costumers and not regular costumers when estimating star ratings.

Type	**Average time (h)**	**# of visits**	**# of visits unique locations**
Service area	0.77	126	57
Motorhome parking	8.32	337	249
Campsite	13.9	570	222

Table 1. Type of point of interest and average time spent on location

4.3. Star rating distribution analysis

Place rating is a score between 1 and 10 which is shown in Figure 4. From the data about places, more than $6,000$ locations have no information about the rating. The most common score is 7 which is attributed to $5,166$ places, followed by score 8 attributed to $4,840$ places. Motorhomes that we tracked visited places with the ratings shown in Table 2. The distribution remains the same and the most common visited camps are still the ones with rating 7 and 8.

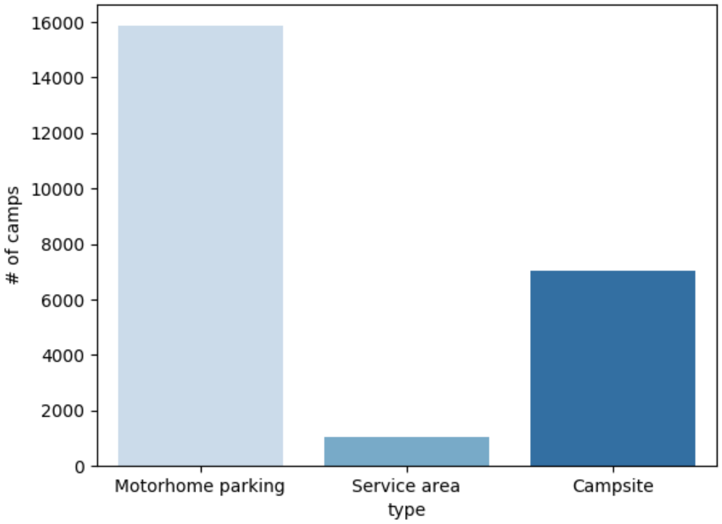


Fig. 3. Category distribution

Star rating	1	2	3	4	5	6	7	8	9	10	Unknown
# of visited camps, parking and service areas	4	7	8	18	44	92	137	110	26	5	77

Table 2. Number of visited camps, parkings and service areas by motorhomes.

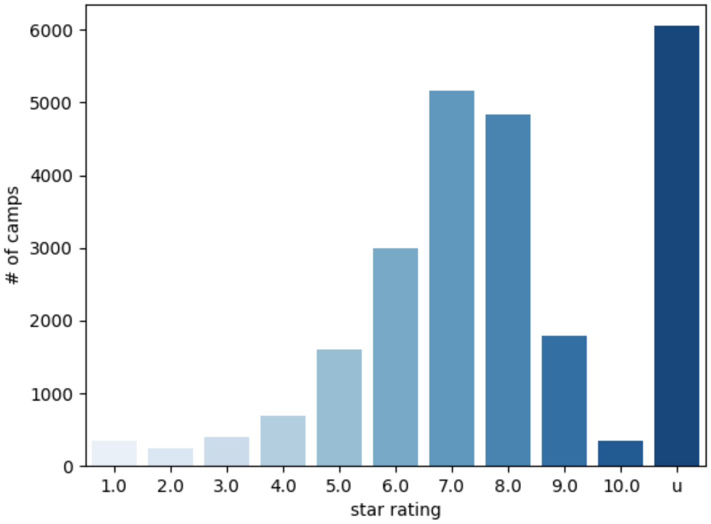


Fig. 4. Star rating distribution

The highest number of camps has France (5,958 camps) and then the number of camps is followed by Germany (4,841 camps), Italy (4,182), Netherlands (1,344 camps).

5. Evaluation

This section presents the methodology of experiments, analysis of the experimental results, and the key findings.

5.1. Experimental setting

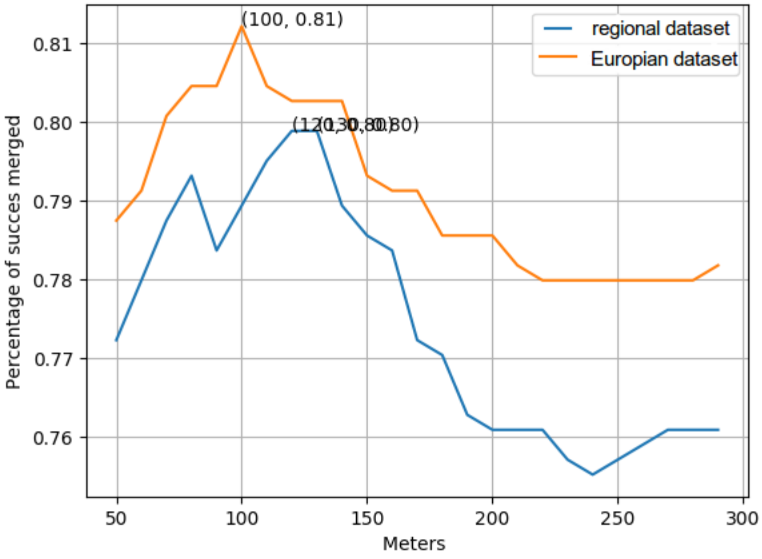


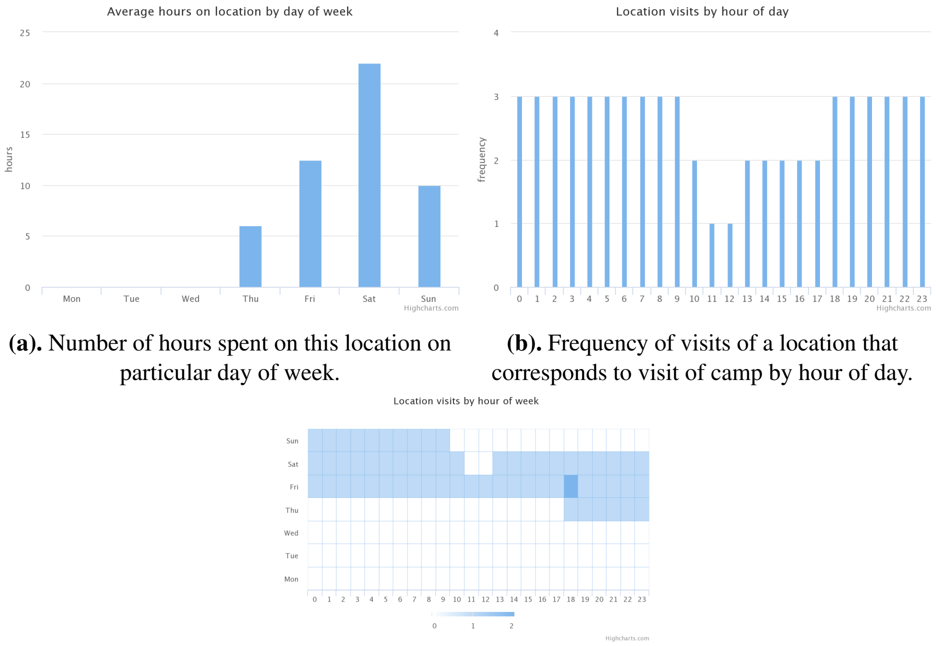
Fig. 5. Tuning the Δ parameter for matching campsites on regional in European dataset.

Parameter selection Multiple data sources can have slightly different coordinates for the same place. In order to overcome this issue we used Algorithm 1, mentioned in Section 3.2. After the algorithm utilization, we perform the evaluation of the merging results. Evaluation dataset was built manually by checking 630 camps from OSM, trying to find a suitable match in other data sources. Some examples were impossible to match even manually. We evaluate the algorithm by comparing its output values to manually annotated data. With this, it calculates the accuracy using following formula:

$$accuracy_{matching} = \frac{true_{matched}}{all_{matched}} \tag{6}$$

The result of accuracy is highly dependent on the appropriateness of the chosen Δ parameter. We wrote evaluation script which checks the accuracy of matching for Δ between 50 and 300 meters within a 10-meter interval (as shown in Figure 5). The higher the number the better the result and the best possible score is 1.

For camps from *European* dataset, the best results have been achieved with Δ of 100 meter (81% correctly merged locations). For camps from *regional* dataset, the best results are with Δ $120 - 130$ m with 80% of correct merges. Both very high and very low Δ produced worse results.



(a). Number of hours spent on this location on particular day of week.

(b). Frequency of visits of a location that corresponds to visit of camp by hour of day.

(c). Frequency of visits of a location by multiple users that corresponds to visit of camp by hour of week.

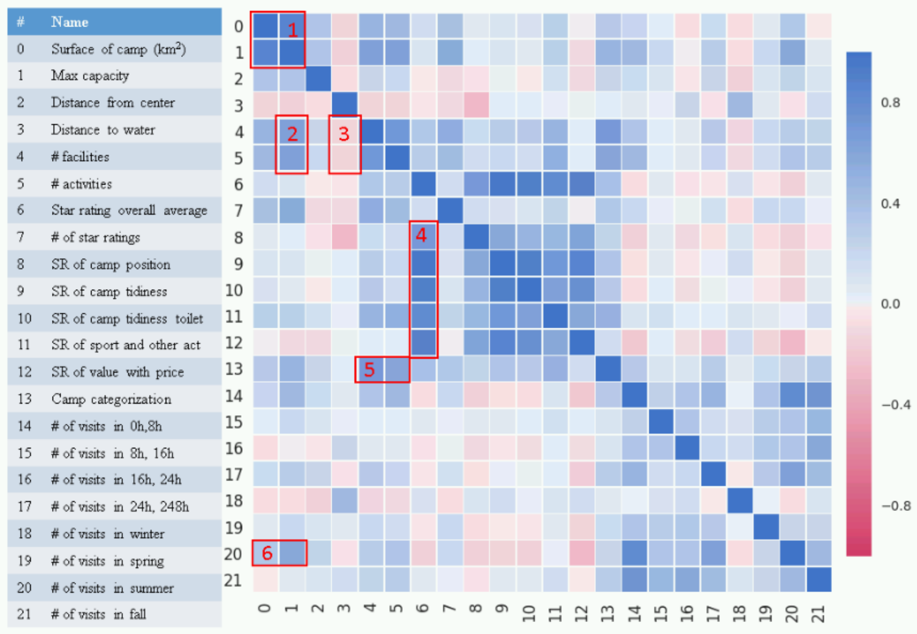
Fig. 6. Mobility patterns.

5.2. Visits to camps

For each global location Mobility patterns service stores the center of the location (latitude, longitude). Based on the underlying visits of this location (staypoints) it calculates statistics like number of visits, time and date of the first visit, last visit and its duration, etc. This type of statistics can also be filtered by time window (take into consideration only the visits that occurred during a specific time range), user (taking into account only visits of a specific user) or group of users, etc. To understand better the semantics of a location it also creates histograms and heatmaps of location visits. A typical example from the collected campsite data is a daily histogram of location visits by motorhomes. Figure 6a shows number of hours spent on this location on the particular day of week. We can observe that the most time in this campsite was spent through weekdays. Figure 6b shows frequency of visits of a location that corresponds to visits for camp an hour of day. We can observe a gap around midday, meaning that motorhomes users more frequently stay in the camp at night. Figure 6c shows heatmap of visits of a location by an hour of week for every week in our dataset. We can observe that the camp is most visited at the end of the week.

5.3. Correlation analysis

Correlation analysis was performed on *regional* and *European* dataset. *European* dataset results are described in the text, but the heatmap matrix is not shown.



#	Name
0	Surface of camp (km^2)
1	Max capacity
2	Distance from center
3	Distance to water
4	# facilities
5	# activities
6	Star rating overall average
7	# of star ratings
8	SR of camp position
9	SR of camp tidiness
10	SR of camp tidiness toilet
11	SR of sport and other act
12	SR of value with price
13	Camp categorization
14	# of visits in 0h,8h
15	# of visits in 8h, 16h
16	# of visits in 16h, 24h
17	# of visits in 24h, 248h
18	# of visits in winter
19	# of visits in spring
20	# of visits in summer
21	# of visits in fall

Fig. 7. Performing correlation analysis on *regional* dataset. The result in red boxes is explained in text from 1 to 6 in text.

Correlation analysis on *regional* dataset On the *regional* dataset we checked which attributes intersect and Figure 7 shows some of the more interesting correlations we found. The red squares in Figure 7 are numbered accordingly with their explanations below:

1. Surface of camp (km^2) is proportional to the maximal capacity of people and campers. This is easy to understand, since higher number of motorhomes and people takes up more space.

2. Number of activities and facilities is also correlated with maximum camp capacity. It is hard for smaller camps to provide so many diverse activities because they can take up a lot of space and can be expensive to maintain. We counted 12 distinct activities: inside swimming pool, tennis, volleyball, bicycle tours, diving, table tennis, basketball, boats, mini golf, rafting and rock climbing. We designated following items as facilities: shop, restaurant, restroom, restroom for wheelchair, chemical toilet discharge, bathroom, service station, dogs allowed, washing machine, yacht quay, a barrier for boats, playground, children bathroom, animation for children, dryer. While smaller camps again cannot provide all the facilities available in larger camping places, some common facilities do exist, such as bathrooms, chemical toilets, service station, dogs allowed, washing machines and dryers, etc.
3. Additionally, number of facilities is negatively correlated with location near water (Distance from water is measured in meters in our analysis). Our explanation is that for many facilities, the presence of lake, river or sea is important.
4. The average star rating of camp is strongly correlated with star rating of camp position, star rating of the tidiness of camp, star rating of the toilet, the star rating of sports activities in camp and star rating of price/value ratio. The most correlated star rating is for tidiness of camp and toilet, because these both express overall cleanliness of camp, making them strongly connected.
5. Camp categorization is a common approach to assessing camps and whether they provide service, activities, facilities from 1 to 5 and also number of activities facilities affect correlation.
6. In the summer people in general visit bigger camps.

Correlation analysis on *European* dataset We also performed correlation analysis on *European* dataset, but this dataset is less interpretable than *the regional* dataset. We can distinguish rules which are that time spent on location is correlated with the presence of facilities.

If people spent more than 24 hours in general area of a camp, they chose a camp with grass, restroom, shower, playground, washing/dryer, internet services and electricity. This is an important discovery, because this shows what is important for users when they are staying at a camp for multiple days. On the other hand, people who stay just up to 3 hours, do not care for the availability of such facilities and we found a strong negative correlation to confirm this.

5.4. Prediction of star rating

Regression Prediction of star rating was performed on *regional* and *European* dataset. In the *regional* dataset the result was close to baseline or even worse, mainly due to a lack of learning examples. We investigated and figured out that the number of examples is too low to predict point-of-interest rating well. Another possible issue could be that the features are not informative enough and maybe additional feature engineering could help. Additionally, there seemed to be a lack of meaningful correlation between camp features and temporal features. Due to mismatching attributes in the datasets, we were unable to perform the merging into a single unified dataset, since that would cause either a certain amount of data loss or empty values in some records.

For every camp we predicted star rating using machine learning methods (described in Section 3.3). In order to make better models the best feature set was chosen for every algorithm. All regression tests were performed on/using training set due to lack of available data. This was done by utilizing stepwise feature selection (described in Section 3.3).

We performed analysis on *European dataset*, and then also separately on camps which were visited in spring and summer, and again separately camps which were visited in fall and winter. Dataset for visited camps in spring and summer will be called *camps visited in warmer months*. The other datasets that include camps which were visited on fall and winter will be called *camps visited in cooler months*. Our assumption is that visitors have different requirements if they visit camp in fall and winter than spring and summer.

Each dataset has a distinct number of visited camps due to different frequency of visits in different seasons. We treat each camp as learning example when it has at least one visit from the motorhomes we are tracking. For evaluation we used 10-fold cross-validation in all three datasets. We tune the parameters for each algorithm separately on every dataset as follows:

- **k-NN**: We varied parameter k between values 1 and 10. Parameter k is number of neighbors.
- **Decision tree:** We varied parameter *max depth* from 1 to 10.
- **Random forest**: We varied parameter *n_estimators* from 1 to 10.
- **SVR**: We varied penalty parameter C of the error term, between 1 and 15. Also we varied ϵ which specifies the epsilon-tube within which no penalty is associated with the training loss function, from 1 to 15. When the SVM is used for regression in this paper is referred as SVR.

This dataset contains 528 camps, which were visited at all seasons. The best k-NN performance was observed with parameter setting k=7. For the random forests algorithm we set *n_estimators*=5. For the decision trees algorithm, we tuned *max depth* to 2. For the SVR algorithm we tuned C=7 and ϵ=1.

Algorithm	**Features**									
	Camps		**Temporal**		**Camps + Temporal**		**Camp+ Complex**		**All**	
	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**
k-NN	0.02	1.53	0	1.5	**0.06**	**1.5**	0.02	1.53	**0.06**	**1.5**
Random for.	0.06	1.5	0.01	1.46	0.11	1.46	0.06	1.5	**0.12**	**1.45**
Decision tree	0.06	1.49	0.02	1.53	0.08	**1.48**	0.06	1.49	**0.09**	**1.48**
Linear reg.	0.05	1.5	0.03	1.52	**0.08**	1.48	0.06	1.5	**0.08**	**1.47**
SVR	0.07	1.49	0.03	1.52	0.01	1.46	0.11	1.45	**0.19**	**1.39**
Baseline	-0.04	1.57	-0.04	1.57	-0.04	1.57	-0.04	1.57	-0.04	1.57

Table 3. Prediction of star rating with different features on dataset where all camps were visited. To evaluate the performance of our model we used measures to evaluate Root mean square error (RMSE) and coefficient of determination (R^2). The best results for every algorithm is in bold.

Table 3 shows results of predicting star rating. SVR outperforms other algorithms with observed RMSE of 1.39. Results indicate that temporal features from *Mobility patterns* improve predicting star rating. Also results improve if we add complex features. For every algorithm we bold the best results in Tables. This means that time spent on location is correlated with star rating. The best features for SVR are

- camp features (internet services, electricity, grass, dog are not allowed, shower)
- temporal features ((24h,248h],(3h,6h],(12h,15h],winter)
- complex features (competitiveness, entropy)

Every algorithm has its own set of features where it performs the best.

Algorithm	**Features**									
	Camps		**Temporal**		**Camps + Temporal**		**Camp+ Complex**		**All**	
	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**
k-NN	0.05	1.51	-0.01	1.49	0.06	1.49	0.05	1.51	0.06	1.49
Random for.	0.06	1.5	0.03	**1.44**	**0.14**	**1.44**	0.06	1.05	**0.14**	**1.44**
Decision tree	0.06	1.5	0.03	1.49	0.09	1.49	0.06	1.5	0.1	1.49
Linear reg.	0.05	1.51	0.02	1.49	0.06	1.49	0.06	1.5	0.08	1.49
SVR	0.06	1.5	0.02	1.46	0.11	1.46	**0.11**	1.46	**0.15**	**1.43**
Baseline	-0.03	1.57	-0.03	1.57	-0.03	1.57	-0.03	1.57	-0.03	1.57

Table 4. Predicting star rating with newly calculated temporal attributes taking to account only spring and summer. To evaluate the performance of our model we used measures to evaluate Root mean square error (RMSE) and coefficient of determination (R^2). The best result for every algorithm is in bold.

Due to different service demand in cooler and warmer months we analyzed camps separately for those seasons. For instance in cooler months we take into account only camps which were visited in fall and winter. We ignore visits in other months. Thus, the temporal feature values change. In this two datasets we again used 10-fold cross-validation method to evaluate results. The dataset for warmer months consists of 486 camps, and dataset for cooler months contains 75 camps.

For dataset in warmer months we set parameters of the algorithms as follows. For k-NN, the number of *k* is 7, for random forest *n_estimators* is 5. For the decision tree we tuned *max depth* to 4. For SVR we set *C*=3 and ϵ=0.9. The parameters for fall and winter are as follows: for k-NN we set *k*=5, for random forest the parameter *n_estimators* was set to 8. For the decision tree we set *maximum depth* to 2. For SVR we set $C = 5$ and $\epsilon = 0.4$. SVR features for spring and summer are the following:

- camp features (internet services, dogs are not allowed, electricity)
- temporal features (time intervals (3h,6h], (9h,12h], (12h,15h], (24h,248h], percent of visits)
- complex features (competitiveness, entropy)

The best features for fall and winter are

- camp features (presence of Internet services, camping-like behavior),
- temporal features((0h,1h], (6h,9h], (21h,24h], number of visits, average duration, spring, min and max stay)

We can see in Table 4 and Table 5 that SVR outperforms the other methods with All features. The best result was obtained on the dataset for cooler months and are presented in Table 5. The best score for R^2 is 0.56 achieved by SVR with Camps + Temporal features and All features. The best RMSE is 0.87 achieved by Linear regression with All the features. We started modeling with POI features only and then added temporal and complex features. Results show that adding temporal and complex features helps with star rating prediction.

Algorithm	**Features**									
	Camps		**Temporal**		**Camps + Temporal**		**Camp+ Complex**		**All**	
	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**	R^2	**RMSE**
k-NN	0.09	1.17	0.16	0.91	0.49	1.22	0.09	1.2	0.49	0.91
Random for.	0.16	1.17	0.15	0.96	0.49	1.17	0.14	1.17	0.49	0.96
Decision tree	0.1	1.17	0.16	1.11	0.24	1.21	0.1	1.21	0.24	1.11
Linear reg.	0.04	1.2	0.12	0.87	0.53	1.25	0.04	1.25	0.53	**0.87**
SVR	0.29	1.06	**0.23**	**0.9**	**0.56**	1.1	**0.29**	1.1	**0.56**	0.9
Baseline	0	1.2	0	1.2	0	1.2	0	1.2	0	1.2

Table 5. Predicting star rating with newly calculated temporal attributes taking to account only fall and winter. To evaluate the performance of our model we used measures to evaluate Root mean square error (RMSE) and coefficient of determination (R^2). The best result for every algorithm is in bold.

The only feature which was selected to every feature subset is internet services also important features are winter sports, when this visit occurs (summer, spring, fall, winter).

Classification Prediction of star rating was performed only for regional dataset. We split star ratings provided by users into three categories:

- (0,4] is range between 0 and 4 and we can mark this as bad camps.
- (4,7] is range between 4 and 7 and it is considered as decent camps.
- (7,10] is range between 7 and 10 and it is considered as very good camps.

We decided to discretize available data in this way because every user has different idea of what they imagine as rating 7 or 8. Due to different personalities and lifestyles, their opinion on what makes a good camp can be biased. We split dataset into test and training set in ratio 80:20. Using cross-validation on the training set we have tuned the parameters and choose the best features subset for each of the algorithms using the algorithm presented in Section 3.3:

- **k-NN:** We varied parameter *k* between values 1 and 10. Parameter *k* is number of neighbors.

- **Decision tree:** We varied parameter *max depth* from 1 to 10.
- **Random forest:** We varied parameter *n_estimators* from 1 to 10.
- **SVC** We varied parameter C of the error term between 1 and 15. When the SVM is used for classification in this paper is referred as SVC.

The best parameter values for which we reported the result are the following. For classification task k-NN, the number of nearest neighbors is 7, for random forest *n_estimator* is 3. For SVC we set *C*=1 and for decision tree will be expanded until all leaves are pure or until all leaves contain less than 2 samples. Table 6 shows results of classifying camps in three categories: bad, decent and very good. Among the tested algorithms, looking at F1 measure, decision tree performed the best with no difference in the performance results when using camps or temporal features. Combining all the features did not increase F1 score, though it did slightly improve accuracy for random forest (from 0.55 to 0.58). Accuracy of the majority class is 0.49, which means that all classifiers perform better or equal as the majority class.

Algorithm	**Features**									
	Camps		**Temporal**		**Camps+Temporal**		**Camps+Complex**		**All**	
	CA	**F1**	**CA**	**F1**	**CA**	**F1**	**CA**	**F1**	**CA**	**F1**
k-NN	0.54	0.56	0.49	0.51	0.52	0.69	0.4	0.39	0.46	0.48
Random for.	**0.55**	0.46	0.52	0.69	0.52	0.69	0.57	0.54	**0.58**	0.57
Decision tree	0.52	**0.69**	0.52	0.69	0.52	0.69	0.52	0.69	0.52	**0.69**
SVC	0.52	0.69	0.52	0.69	0.57	0.5	0.52	0.69	0.53	0.41
Logistic reg.	0.52	0.53	0.52	0.69	0.54	0.49	0.55	0.57	0.49	0.48
Majority class	0.49	0.0	0.49	0	0.49	0	0.49	0	0.49	0

Table 6. Predicting star rating with calculated temporal attributes. To evaluate performance of our model we used measures to evaluate Accuracy (CA), F-measure ($F1$). The best result for every algorithm are in bold.

6. Conclusions and Future work

In this paper, we proposed a framework to merge multiple different sources of point-of-interest data, including the real-time GPS coordinates from multiple vehicles, which can bring valuable additional information. We tested our system on motorhome camps, but it is easy to adopt the process for extracting other points with time component by simply configuring the properties file. We also conducted extensive experiments to investigate if temporal features help with predicting star rating and consequently assessing the place quality. We can say that it is possible to predict star rating via camp characteristic and temporal data of users. Our analysis showed that the temporal features obtained from geospatial information improve the regression prediction by 4-13 %. For classification we observed that classification accuracy improves for 3 % when using all the features set. From the experiments we can see that both classification and regression are better than the baselines. Another finding is that the temporal features improve accuracy of the models.

The regression model which outperforms other models most of the times is SVR. The best performing classification models are SVC and random forest.

In the future, this work could be extended in multiple dimensions. Firstly, by adding additional data about places like gas stations. Another option is to include more data in addition to the GPS coordinates. The motorhomes that we are tracking are already equipped with onboard sensors and are measuring other values, such as clean water level, wastewater, air quality within living space and car batteries state of charge. With this data we can build a recommendation system which suggests users where and when to discharge wastewater, sewage or refill the water.

Acknowledgments. This work was supported by the Slovenian Research Agency and the ICT program of the EC under project OPTIMUM (H2020-MG-636160). The authors would like to thank Anja Polajnar for proofreading of the manuscript.

References

1. Avtokampi description of data. http://www.avtokampi.si/, Accessed: 2017-05-06
2. Campercontact description of data. https://www.campercontact.com/en/, Accessed: 2017-05-06
3. Geofabrik. http://download.geofabrik.de/europe.html, Accessed: 2017-06-06
4. OpenStreetMap: Tag:tourism_camp site. http://wiki.openstreetmap.org/wiki/Tag:tourism%3Dcamp_site, Accessed: 2017-06-06
5. Optimum project. http://www.optimumproject.eu/, accessed: 2018-06-17
6. Optimum project. http://www.optimumproject.eu/about/pilot-cases/pilot-case-3/innovations-3.html, accessed: 2018-06-17
7. Osm2psql. http://wiki.openstreetmap.org/wiki/Osm2pgsql, Accessed: 2017-06-06
8. Scikit-learn regression models. http://scikitlearn.org/stable/supervised_learning.html#supervised-learning, Accessed: 2017-10-06
9. Europe: Registrations of new Motor Caravans per month 2016. Europian Caravan Federation (2017), http://www.e-c-f.com/fileadmin/templates/4825/images/statistics/europazul-9.pdf
10. Adomavicius, G., Kwon, Y.: New recommendation techniques for multicriteria rating systems. IEEE Intelligent Systems 22(3) (2007)
11. Asghar, N.: Yelp dataset challenge: Review rating prediction. CoRR abs/1605.05362 (2016)
12. Barbosa, R.R.L., SÃąnchez-Alonso, S., Sicilia-Urban, M.A.: Evaluating hotels rating prediction based on sentiment analysis services. Aslib Journal of Information Management 67(4), 392–407 (2015), https://doi.org/10.1108/AJIM-01-2015-0004
13. Bennett, J.: OpenStreetMap Be your own Cartographer. Packt Publishing Ltd. (2010)
14. Bradesko, L.: Knowledge acquisition through natural language conversation and crowdsourcing 150 (2018)
15. Carbon, K., Fujii, K., Veerina, P.: Applications of machine learning to predict Yelp ratings (2014)
16. Fan, M., Khademi, M.: Predicting a business star in Yelp from its reviews text alone. CoRR abs/1401.0864 (2014)
17. Hsieh, H.P., Li, C.T., Lin, S.D.: Estimating potential customers anywhere and anytime based on location-based social networks. In: Proceedings of the 2015th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part II. pp. 576–592. ECMLPKDD'15, Springer, Switzerland (2015), https://doi.org/10.1007/978-3-319-23525-7_35

18. Hu, Y.H., Chen, K.: Predicting hotel review helpfulness: The impact of review visibility, and interaction between hotel stars and review ratings. International Journal of Information Management 36(6, Part A), 929 – 944 (2016), http://www.sciencedirect.com/science/article/pii/S0268401215301845
19. Huang, J., Rogers, S., Joo, E.: Improving restaurants by extracting subtopics from Yelp reviews (2014)
20. Jannach, D., Gedikli, F., Karakaya, Z., Juwig, O.: Recommending hotels based on multidimensional customer ratings. In: Information and communication technologies in tourism 2012, pp. 320–331. Springer, Vienna (2012)
21. Karamshuk, D., Noulas, A., Scellato, S., Nicosia, V., Mascolo, C.: Geo-spotting: Mining online location-based services for optimal retail store placement. CoRR abs/1306.1704 (2013), http://arxiv.org/abs/1306.1704
22. Levandoski, J.J., Sarwat, M., Eldawy, A., Mokbel, M.F.: Lars: A location-aware recommender system. In: Proceedings of the 2012 IEEE 28th International Conference on Data Engineering. pp. 450–461. ICDE '12, IEEE Computer Society, Washington, DC, USA (2012), http://dx.doi.org/10.1109/ICDE.2012.54
23. Li, Q., Zheng, Y., Xie, X., Chen, Y., Liu, W., Ma, W.Y.: Mining user similarity based on location history. In: Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. pp. 34:1–34:10. GIS '08, ACM, New York, NY, USA (2008), http://doi.acm.org/10.1145/1463434.1463477
24. Lv, M., Chen, L., Xu, Z., Li, Y., Chen, G.: The discovery of personally semantic places based on trajectory data mining. Neurocomputing 173(Part 3), 1142 – 1153 (2016), http://www.sciencedirect.com/science/article/pii/S0925231215012916
25. Narayanan, M., Cherukuri, A.K.: A study and analysis of recommendation systems for location-based social network (LBSN) with big data. IIMB Management Review 28(1), 25 – 30 (2016), http://www.sciencedirect.com/science/article/pii/S0970389616000021
26. Ravi, L., Vairavasundaram, S.: A collaborative location-based travel recommendation system through enhanced rating prediction for the group of users. Intell. Neuroscience 2016, 7– (Mar 2016), http://dx.doi.org/10.1155/2016/1291358
27. Tiwari, S., Kaushik, S.: User category based estimation of location popularity using the road GPS trajectory databases. Geoinformatica (2014)
28. Tiwari, S., Kaushik, S.: Popularity estimation of interesting locations from visitor's trajectories using fuzzy inference system. Open Computer Science 6(1) (2016), http://www.degruyter.com/view/j/comp.2016.6.issue-1/comp-2016-0002/comp-2016-0002.xml
29. Vitorino, R., Herga, Z., Bradesko, L.: Integrated mobility decision support (318452). EU project Mobis (2015), http://www.europa.eu/whatever
30. Yan, Z.: Towards semantic trajectory data analysis: A conceptual and computational approach (2009)
31. Zheng, Y., Xie, X.: Learning location correlation from GPS trajectories. In: 2010 Eleventh International Conference on Mobile Data Management. pp. 27–32 (May 2010)
32. Zheng, Y.: Trajectory Data Mining: An Overview. ACM Trans. On Intelligent Systems and Technology 6(3), 1–41 (2015)

Matej Senožetnik received his B.Sc. in Computer science and engineering (2015) from University of Ljubljana. He is now pursuing M.Sc. in Information and Communication technologies at Jožef Stefan International Postgraduate School. He works on European and national projects as a student researcher at the Jožef Stefan Institute.

Luka Bradesko is part of the NLP Enrichment team at Bloomberg L.P. He obtained his PhD at Jožef Stefan International Postgraduate School, Ljubljana Slovenia with the topic

Knowledge Acquisition through Natural Language Conversation and Crowdsourcing. His research interests are in Natural Language Processing, Logical Inference and Knowledge Extraction. From 2008 to 2013 he worked as a principal software engineer for Cycorp Europe, which was at the time an EU branch of the American AI company Cyc Inc. During these years, he worked on an EU project developing distributed large scale inference engine (LarKC), and also on an natural language based AI assistant startup built on top of Cyc (Curious Cat). Some of the recent projects include a concept of an intelligent motorhome (reasoning engine with predictive analytics, interacting with sensors and actuators), geo-spatial based predictive analytics, and Named Entity Disambiguation algorithm which is already part of his work at Bloomberg L.P.

Tine Šubic has received B.Sc. in Computer Science and Informatics (2017) and is now pursuing M.Sc. in Electrical Engineering at University of Ljubljana. He works on European and national projects as a student researcher at the Jožef Stefan Institute.

M.Sc. Zala Herga is a researcher and a PhD student at Jožef Stefan Institute, Ljubljana. She finished her master's degree in Financial Mathematics at Faculty of Mathematics and Physics, University of Ljubljana. She has experience in AI, data mining, analytics and predictive statistics. She mainly applied her research to two areas: risk management and traffic domain.

Jasna Urbančič received her B.Sc. in Physics (2015) and is now pursuing M.Sc. in Computer and Information Science at University of Ljubljana. She works on European and national projects as a student researcher at the Jožef Stefan Institute.

Primoz Skraba is currently a senior lecturer of Mathematics at Queen Mary, University of London as well as a researcher in the Artificial Intelligence Laboratory at the Jozef Stefan Institute, Slovenia. He has also held positions assistant professor of computer science at the University of Koper and adjunct professor of mathematics at the University of Nova Gorica. He received his Ph.D. in Electrical Engineering from Stanford University in 2009. His main research interests are applications of topology to computer science including data analysis, machine learning, sensor networks, and visualization.

Prof. Dr. Dunja Mladenić `http://ailab.ijs.si/dunja_mladenic/` works as a researcher and a project leader at Jožef Stefan Institute, Slovenia, leading Artificial Intelligence Laboratory and teaching at Jožef Stefan International Postgraduate School, University of Ljubljana and University of Zagreb. She has extensive research experience in study and development of Machine Learning, Big Data/Text Mining, Sensor Data Analysis, Internet of Things, Data Science, Semantic Technology techniques and their application on real-world problems. She has published papers in refereed journals and conferences, co-edited several books, served on program committees of international conferences and organized international events. She serves as a project evaluator of project proposals for European Commission and USA National Science Foundation. From 2013-2017 she served on the Institute's Scientific Council `http://www.ijs.si/ijsw/Scientific_Council/`, as a vice president (2015-2017). She serves on Executive board of Slovenian Artificial Intelligence Society SLAIS (as a president of SLAIS (2010-2014)) and on Advisory board of ACM Slovenija.

Received: December 12, 2017; Accepted: August 7, 2018.