A Recommendation Engine for a Smart Parking Ecosystem

Spyros Kontogiannis¹,³, Nikos Zacharatos², and Christos Zaroliagis²,³

¹ Department of Computer Science & Engineering, University of Ioannina, Ioannina, Greece
kontog@uoi.gr
² Department of Computer Engineering & Informatics, University of Patras, Patras, Greece
{zacharato,zaro}@ceid.upatras.gr
³ Computer Technology Institute and Press “Diophantus”, Patras, Greece

Abstract. Due to waste of time and unnecessary pollution when seeking for a parking space in urban areas, the lack of parking spaces has a serious social and environmental impact. To address this problem, we developed the SocialPARK ecosystem which engages a community of interacting citizens, parking vendors and municipalities. SocialPARK revolves around a crowdsourcing scheme that aggregates parking information, for free public spaces reported by commuters, and for commercial parking spaces offered by parking vendors, in a single integrated platform. SocialPARK offers a variety of services, including “Park-and-Ride” options, making the city centers more accessible with less pollution. Apart from commuters’ participation, a central challenge is the involvement of as many parking vendors as possible, to offer a multitude of parking alternatives to the commuters. We present a business-oriented recommendation engine, developed as part of SocialPARK, to motivate parking vendors to participate in the envisioned open market of parking-related services. The engine is composed of two modules: the first module gathers usage information about the utilization of parking houses within a certain area; (ii) the second module exploits this information to exhibit statistical data and create targeted recommendations for parking houses, rendering their businesses more competitive and improving the services they offer.

Keywords: Recommendation systems · Parking

1 Introduction

Cars provide an unrivaled combination of speed, autonomy, and privacy, which is why people prefer to use them on a daily basis. The congregation of cars in big cities that have not been designed to withstand the load, creates many problems and parking is one of the most severe ones (Ibrahim, 2017).

Parking issues have prevailed in cities and metropolitan regions and constitute arguably the most widely debated concern among both the public and professionals. The main challenge is an imbalance between parking offer and demand. Therefore, a parking ecosystem should be an important part of the urban-life system, and its absence is linked to traffic congestion, accidents, and pollution (Shoup, 2006). Although an
effective parking system can improve urban transportation, the city’s environment and inhabitants’ quality of life, parking is a rather overlooked aspect of urban planning. Urban planners should explore more efficient and imaginative solutions to the parking problem, on the level of administration, planning, and design (Mingardo, van Wee, & Rye, 2015) (Lin, Rivano, & Le Mouël, 2017).

SocialPARK tackles the problem by providing technologically innovative solutions to make it easier for commuters to search for free parking spots. Its goal is to maintain a digital platform providing personalized parking-related information and services, i.e., a parking-as-a-service platform. The challenge of SocialPARK is twofold: (i) to improve the commuters’ quality of life by dealing with the loss of time and the environmental repercussions; (ii) to facilitate the monitoring of parking houses of the involved parking vendors and support them in their decision-making process towards improving their business exploitation plans.

This paper focuses on the provision of a parking-business oriented recommendation engine that digests historical information, along with stakeholders’ business profiles, to provide targeted and explainable recommendations which would lead to better exploitation plans for their own parking houses.

The paper is organized as follows. Section 2 presents an overview of the SocialPARK ecosystem’s architecture. Section 3 presents a literature overview for parking-related recommender systems. It also presents our approach for a parking-business oriented recommender system. Section 4 reports some representative experimental results. Finally, Sec. 5 concludes the paper and briefly discusses directions for future work in the area.

2 SocialPARK Architecture

The high-level architecture of the SocialPARK ecosystem is depicted in Fig. 1. It consists of the following conceptual modules:

(a) Parking Information Brokerage Module: This module is in charge of storing and maintaining all parking-related and profile information in an appropriate database, while ensuring privacy and anonymity for all the involved stakeholders. The parking-related data originates from two sources: parking owners, who use automated parking availability notification techniques; and commuters, who spontaneously provide crowdsourced parking-availability information.

Due to the variety of the data to be collected, a NoSQL database was chosen, mainly to cover the needs for scalability and simplicity of managing unstructured data from diverse sources (e.g., fully/partially automated updates, voluntary crowdsourced information, etc.). In particular, MongoDB was our technological choice, which is an open-source, incredibly efficient, and volatile database that also supports the JSON language for data transmission. Because the SocialPARK ecosystem keeps individualized data for commuters, parking houses and parking vendors, data protection and anonymization are particularly critical. An anonymization method is used to obscure commuters’ actual identities when interacting with the system, on a per-session basis, to avoid leakage of IDs and everyday commuting habits. Furthermore, before being granted access to view and edit their personal profile, data or information relevant to parking company regulations
and strategic goals, all the involved stakeholders – i.e., commuters and parking owners – must verify themselves.

**Fig. 1.** Overall architecture of SocialPARK

In addition, the brokerage module includes the required functions for conducting statistical analysis of aggregated parking data, as well as a business-oriented recommendation engine for making targeted suggestions how parking vendors could optimize their business strategies.

(b) Personalized parking & routing: This module takes over the task of monitoring parking house availability and/or providing booking services, as well as allowing closed self-organized groups of commuters (e.g., disabled), to share their own parking spaces.

Specialized rewarding schemes are engaged to motivate voluntary participation of commuters, both in the crowdsourced aggregation of parking related information and in self-organized groups. One rewarding scheme is for the general public, to encourage participation in crowdsourcing, and the other is to ensure the survival of self-organized closed groups of commuters sharing their parking places within the group.

Commuters can also use the module to get point-to-parking-spot routing and real-time guidance.

(c) Commuter frontend API: Commuters interact with the full SocialPARK ecosystem via this module, which is a mobile API. Due to its popularity in the smartphone market, the implementation is for Android OS.
Parking owner frontend API: This module is a web-based API implemented as a unified dashboard that gives parking owners (and parking house managers) access to all the functionalities related to the management of their profiles, the visualization of the results of statistical analyses for parking-related historical data, and the provision of business-improvement recommendations.

2.1 Brokerage Module

This module refers to parking owners of SocialPARK. The front-end is used by the owners to register and log-in to the system, add their parking houses with all their features, and access the module’s back-end system.

The back-end system collects static and dynamic data from the parking houses participating in SocialPARK, which is then analyzed and presented to the parking vendors, while also proposing ideas for better commercial exploitation of their own spaces, towards becoming more competitive. Statistical data concerning information related to parking-usage is collected using two methods.

The first method is automated and receives information via requests to the Data Interconnection Unit, which in turn draws them from SocialPARK’s database, per parking house, at regular intervals (sixty minutes was initially chosen, which can be easily increased/decreased in the future depending on the needs of the owners). More specifically, for each parking house, occupancy data is collected for each parking-space’s subcategory (general, disabled, elderly, pregnant women/parents with prams), as well as for the total occupancy (in absolute values and percentages).

The second method concerns real-time information updates, regarding parking-availability, that is, customers entering or leaving the parking houses. The method is provided via a remote procedure call, provided that the parking house supports an automated mechanism for recording the incoming and outgoing vehicles. Alternatively, in addition to the automated mode, authorized parking employees are provided with the option to manually update parking-availability data through a web interface.

Regardless of how each provider chooses to update their parking-availability information, the data updates occur as follows. When a vehicle enters the parking house, the Brokerage Unit is informed, through an http-post request. The body of the request includes the license plate of the car, the unique ID of the parking house and an alphanumeric ID indicating that this is an entrance. The Brokerage Unit stores the hashed license plate of the vehicle (until the same vehicle leaves the parking house), the parking house ID and the current time (timestamp).

When a vehicle leaves the parking house, the Brokerage Unit is informed with a similar http-post request indicating that this is an exit from the parking house. After matching the hashed license plate and the parking house ID with those of a previous entrance (i.e., ensuring that the same vehicle still resides in the parking house), the stored license plate is deleted and the time difference between these two posts is saved.

From these real-time updates of parking-house availability information, we obtain the parking durations and arrival-times of customers which, apart from providing vital data, will later help us build a representative operational “fingerprint” per parking house.
3 Parking-Related Recommender Systems

Recommender systems (Ricci, Rokach, & Shapira, 2022) predict ratings (or preferences) of users for items which they have not yet considered, in order to provide meaningful suggestions of new items to users. There are different types of recommender systems, each with its own characteristics, pros and cons (Son, Kim, Kim, & Cho, 2015). The major categories are content-based recommender systems, collaborative-filtering recommender systems, demographic-based recommender systems, knowledge-based recommender systems and hybrid recommender systems.

3.1 Review of Related Literature

We present some of the relevant work that has been carried out on parking-related recommender systems. Commuters typically could get recommendations about parking spots as a result of a (commuters, parking-spots) relation. Providers, on the other hand could get recommendations about pricing policies and types of services to invest on, as a result of historical data analysis for their parking-business exploitation. The literature on parking-related recommender systems almost exclusively focuses on commuters.

In (Rizvi, Zehra, & Olariu, 2019), a recommendation system for Smart Cities is proposed, which takes the commuters’ preferences (e.g., type of parking) and constraints (e.g., maximum price to pay, or maximum walking time to destination) into consideration. The Agent-oriented Smart Parking Recommendation System for Smart Cities (ASPIRE) acts like an on-demand parking lookup service, matching the drivers’ preferences with the best matching spots. The availability of parking spots in the various parking houses is updated in real time, using IoT sensors. The data collected by ASPIRE also provides local governments with parking demand and supply statistics for specific geographical areas, aiding them in the decision-making process of building new parking houses.

In (Tsai & Chen, 2021), a parking recommendation system for Smart Cities is presented, which recommends the best parking house to the commuter based on the current traffic flow and parking house information. A “time gap” is taken into account between the commuter’s query-time and the actual arrival-time at the parking lot. E.g., popular parking lots are usually either fully occupied or impose a long waiting line upon arrival, although they appeared empty at query time. A queueing model is used to estimate the probability to obtain a free space upon arrival at a parking lot, by utilizing historical and real-time data. The real-time updates, regarding parking-space availability, are performed using again IoT sensors.

In (Saleem, Rehmani, Crespi, & Minerva, 2021) the anonymity and privacy of commuters’ data is considered. Two solutions are proposed to preserve their privacy in parking recommender systems, while analyzing the parking history using anonymization and differential privacy techniques. An experimental evaluation is carried out with a data set constructed from real parking measurements, to evaluate the trade-off between privacy protection and utilization of the parking spaces.

In (Rahaman, et al., 2021) a multi-criteria parking recommender system is proposed, which takes into consideration criteria like fare, parking rules, walking distance to destination, travel-time and likelihood of a parking spot being unoccupied at arrival time, in
order to create recommendations for the users. The novelty of this recommender system is that these criteria are dynamic, may change over time and sometimes conflict with each other.

In (Montgomery, 2005) a pricing decision support system (PDSS) is proposed, targeting supermarket retailers, with a goal to suggest optimal pricing and promotional strategies based on historical data. The proposed system forecasts movement, revenue and profit in real-time using the current prices and the offers as variables. Moreover, it manipulates prices of (groups of) products. Finally, it creates promotional offers and price strategies about products and even warns users about bad pricing strategies that are already implemented.

To our knowledge, the only recommendation systems targeted for parking vendors are the ones proposed in (Lei & Quyang, 2017) and (Tian, Yang, Wang, & Huang, 2018). These systems dynamically change the prices of parking spots in real time, based on reservations and occupancies per parking house. These two systems look alike our system, but they have two main downsides. First, they rely on bookings/reservations information, which are neither predictable by every customer, nor supported by all parking houses. Second, they ignore historical data of both parking houses and geographical areas, but only increase or decrease the prices based on current vacancies of each parking house. Moreover, no similarity metric is considered for the parking houses that would exploit their business profiles (e.g., whether they target at commuters seeking for short-time parking spots, or for residents within their geographical area seeking for overnight parking-spots), towards providing focused suggestions for business improvement to the parking house.

3.2 Our Approach: Parking-Business Recommender System

Our Parking-Business Recommender System (PBRS), implemented within SocialPARK, targets at parking vendors and suggests ways to increase the utilization of their parking houses. It can be classified as a decision support system (DSS) (Keen, 1980) that uses recommendation systems’ techniques in order to correlate the parking houses and provide targeted suggestions to parking vendors, on a per-parking-house base.

In a nutshell, PBRS is based on the following axes, in order to provide recommendations for a parking house $H$: (i) exploitation of the historical parking-related data of $H$; (ii) consideration of aggregated parking-related historical data from the broader geographical area around $H$; and (iii) comparison of $H$’s utilization with that of a targeted group of competing parking houses within its own geographical area, which have a similar business profile with $H$. The methods implemented within PBRS are a combination of:

- **Similarity metric for parking houses**: Each parking house is described via a “parking fingerprint”, i.e., a vector describing its spatiotemporal usage by diverse types of commuters (e.g., shoppers, workers, residents, etc.). Consequently, the “similarity” for pairs of parking houses is computed, i.e., pairs of houses that focus on the same types of customers and are located in a nearby geographical area, according to the cosine similarity of the corresponding parking fingerprints.
• **Knowledge-based suggestions for pricing policies:** They are proposed to the parking vendors, based on detailed parking data of their own parking houses.

• **Collaborative-filtering:** A parking vendor (or parking house administrator) gets targeted recommendations for improving the utilization of a particular parking house, taking into account successful policies of their direct competitors (similar houses within their geographic area) for common groups of customers.

• **Parking-related aggregated historical information:** A parking vendor gets recommendations for altering the business plan of a particular house (e.g., change targeted groups), based on aggregated historical parking information of the corresponding geographical area.

The “fingerprint” of each parking house acts as its characteristic vector. The parking durations, the customers’ arrival-times, and the occupancy levels do not constitute a representative profile of the house on their own. For example, it would be incorrect to assume that two parking houses are similar because their typical customers park for the same duration, since they could arrive at completely different hours of the day. To avoid this, we create parking-houses’ fingerprints from spatiotemporal data of parking-occupancies. In particular, the fingerprint of a parking house is created as follows. Customers are separated in three categories depending on the duration of their stay: **Shoppers** that park up to four hours; **workers** that park five to eleven hours; and **residents** that park for at least twelve hours.

For each hour of the day, we maintain a triplet of counters, one per customer category, which shows how many new customers of that type arrived at the parking house. The array \( C \) stores 24 triplets of counters (i.e., \( 72 = 24 \cdot 3 \) integers) per day, with the new entries per hour and per category of customers, for the particular parking house. E.g., if the 17th triplet of the array (corresponding to the interval 17:00–17:59) is \([14, 5, 3]\) this indicates that 14 shoppers, 5 workers and 3 residents entered the parking house during this interval. Since we do not know beforehand the parking times of customers, \( C \) is updated upon their departures from the parking house.

Moreover, an array \( D \) of similar structure (three counters per hour) stores the number of customers residing at a parking house during a given time-interval. E.g., if the 17th triplet of the array (corresponding to the time interval 17:00–17:59) is \([34, 12, 6]\), this indicates that, during this interval, there were 34 shoppers, 12 workers, and 6 residents in the parking house. Using the aggregated \( D \) arrays, we can compare the occupancies of different parking houses for a specific category of customers.

PBRS provides a recommendation for a parking house \( P \) as follows: All the data related to parking houses are retrieved from the system database. They are distinguished in two groups, based on their Euclidean distance from \( P \). The ones geographically close to \( P \), and those far away from \( P \) (which are ignored). For each parking house \( Q \) close to \( P \), the aggregated \( C \) array of \( Q \) is computed by averaging the corresponding \( C \) arrays over all available days. This aggregated \( C \) array is perceived as the “fingerprint” of \( Q \) and it is used for comparing it with \( P \), according to the cosine similarity metric. \( Q \) is then perceived as **competitive to** \( P \), if its similarity with \( P \) is greater than a given threshold. PBRS produces recommendations to \( P \) based on three distinct levels of analysis:
(a) **Type-A recommendations:** From $P$’s occupancy data alone, the system can already create recommendations about subcategory spaces (e.g., groups of spots for elderly, disabled, parents with children, etc.) that are underutilized or overutilized. E.g., if the “elderly” subcategory has occupancy over 90% during morning-hours, while another subcategory (e.g., “disabled”) has low occupancy for the same time interval, the system would recommend some rearrangement of parking spaces between these two subcategories for the particular interval.

(b) **Type-B recommendations:** From $P$’s nearby parking houses’ data, the system creates recommendations about the geographical area’s most dominant customer category. E.g., if $P$’s most active category of customers does not coincide with the dominant category of customers within the geographical area of $P$, PBRS would consider lowering a certain price (e.g., long-stay charges), or to promote a targeted offer for specific time slots (e.g., 12:00–14:00), to attract more customers from the dominant category.

(c) **Type-C recommendations:** The performance of $P$ is now compared against the most competing parking houses within $P$’s geographical area, i.e., those targeting similar customer categories with $P$. This is done by aggregating the $D$ arrays of each parking house and comparing the occupancy of their most profitable customer category and their pricing policy. Then PBRS would recommend targeted pricing policy fluctuations, based on aggregated offer and demand among $P$’s competing houses, in the flavor of user-based collaborative filtering.

## 4 Experimental Evaluation

The city of Thessaloniki was chosen by SocialPARK as its main case study. Thessaloniki is the second largest city of Greece with population of 324,766 inhabitants in the municipality 1.12 million inhabitants in the metropolitan area. The densely populated environment of Thessaloniki and the limited public space to serve all urban functions with the consequent impact on the environment and the society, have necessitated an effective utilization of parking space. It is commonplace that one way of devaluing public space in urban environments is by occupying them with parked vehicles, and in particular by abusive/illegal parking, or by vehicles that make more trips while searching for free parking spots. The aforementioned reasons make Thessaloniki an ideal testbed for the SocialPARK ecosystem and our approach for a parking-business oriented recommendation engine.

Five private parking houses participated in SocialPARK and provided real-time occupancy data during the experimental evaluation. All of them reside in the center of Thessaloniki, in close distance to each other. We used the data of those parking houses to create recommendations for a hypothetical new parking house $P$, also supposed to be located in the same geographical area. The data of $P$ were altered so as to try the different scenarios.

The experimental evaluation was executed in three phases. In every phase one additional feature of PBRS was added for testing. In the first phase (type-A recommendations), the system created recommendations using only $P$’s own occupancy data (i.e., a purely content-based approach) and it suggested internal parking-spot rearrangements between underutilized and overutilized subcategories of parking spaces within $P$. 
In the second phase (type-B recommendations), the system considered the parking-occupancy data from the geographical area around \( P \) (i.e., a purely demographic approach). Apart from parking-spot transfers between underutilized and overutilized subcategories of spots, the system also recommended to \( P \) some changes in targeted categories of customers (after calculating the area’s most dominant categories) and price fluctuations (depending on the occupancies and prices of the parking houses in the geographical area).

In the third and final phase (type-C recommendations), the system created recommendations using the data from the area around \( P \), while also computing the relevance of the most competing parking houses of \( P \) (i.e., a hybrid method that combines \( P \)'s content with demographic data and user-based collaborative filtering). This resulted in more targeted price fluctuation recommendations, since PBRS compared similar parking houses within \( P \)'s geographical area.

The following example illustrates the first phase. Assume that \( P \) has 100 conventional spots, 30 disabled spots and 5 elderly spots. Figure 2 shows the mean occupancy percentage of \( P \)'s spot subcategories. PBRS recommended a transfer of spots from the disabled (the lowest occupancy at the peak hours) to the conventional subcategory, since conventional spots have more than 95% occupancy at peak hours. More specifically, PBRS computed the maximum value of the lowest subcategory at the peak hour (the disabled with 65% in this example), and recommended transferring 8 spots \((90\% - 65\% = 25\%, \text{i.e., about 8 out of 30})\) of the disabled subcategory to the conventional subcategory.

Fig. 2. Average occupancies of subcategories in \( P \). The horizontal axis denotes the hours of the day, while vertical axis denotes the occupancy percentages.

The next example illustrates the second phase, where the data of the area around \( P \) is also taken into account. \( P \) was assumed to be in the city-center of Thessaloniki, close to all the other parking houses. Figure 3 shows: (up-left) shows the mean occupancy of \( P \) (red line) and the parking houses in its area (blue line); (up-right) the average parking duration of \( P \) (in hours). PBRS deduced that, obviously, \( P \)'s target group was not the same as that of a typical customer in its area. The dominant category of the area is
Fig. 3. Left: Mean Occupancies of P (red) and the area around it (blue) throughout the day. Right: Average customer parking duration of P.

shoppers. Therefore, PBRS suggested P to target the dominant customer category within this area, by lowering the price of short-parking durations. The recommended price drop was based on the pricing policy of the most successful parking house for this customer category. In our example, the most successful parking house had a short-parking charge of 3€/hour, while P had 4€/hour. Had the short-stay price been at most equal to the most successful parking house, PBRS would suggest creating a targeted offer during the middle of the day (hours 9:00–16:00), where the difference between P’s and the area’s average occupancies is the greatest, towards attracting more customers.

The third example illustrates the final phase of our experimental evaluation. Let us use the same scenario for P as in the previous example. P’s most profitable customer category concerns residents (who park for more than 12 h), who constitute 49% of its clientele. In addition to the recommendations of the previous phase, PBRS would create a recommendation based on the performance of P and its competitors and their pricing policies on that specific customer category. Out of the five parking houses, only one has a similar strategy with P (with their cosine similarity being greater than a given threshold, e.g., 75%). Figure 4 shows their occupancies (upper left chart), their most profitable customer category occupancies (upper right chart) and their pricing policies for long-stay charges (bottom chart). Since P, compared to its competitor, has lower occupancy and higher long-stay charge for residents, PBRS recommended a price decrease for long-stay charge (e.g., to 17 euros in the example of Fig. 4, which matches that of the most successful parking house.

5 Conclusions

In this paper, we presented a parking-business oriented recommendation system, aimed towards parking owners and their parking houses. The spatiotemporal occupancy vectors for each parking house and each geographical area’s corresponding aggregated data are considered as parking-house “fingerprints”, in order to create targeted recommendations based both on historical data and collaborative-filtering techniques, rendering them more competitive and profitable. In the future a more complex way of calculating the competitive parking houses will be considered, instead of the traditional collaborative filtering with cosine similarity.
Fig. 4. Upper Left: General mean occupancies of P and its similar competitor houses throughout the day. Upper Right: Mean Occupancies of most profitable customer category throughout the day. Lower Middle: Price policy of their most profitable category.

Acknowledgment. This work was supported by the Operational Program Competitiveness, Entrepreneurship and Innovation (call Research – Create – Innovate, co-financed by EU and the Greek State) under contract no. T1EDK-03616 (project SocialPARK).

References


Keen, P.G.: Decision support systems: a research perspective. Cambridge: Massachusetts Institute of Technology, Center for Information Systems Research, Alfred P. Sloan School of Management (1980)


