Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

Article

Characterizing Situations of Dock Overload in Bicycle Sharing Stations

Luca Cagliero*[1,†]0000-0002-7185-5247, Tania Cerquitelli[1,†], Silvia Chiusano[1,†], Paolo Garza[1,†], Giuseppe Ricupero[1,†], and Elena Baralis[1,†]

[1] Dipartimento di Automatica e Informatica, Politecnico di Torino, Corso Duca degli Abruzzi, 24 - 10129 Torino, Italy

* Correspondence: luca.cagliero@polito.it; Tel.: +39-011-090-7179

Abstract: Bike sharing systems are a key element of a smart city as they have the potential for reducing pollutant emissions and traffic congestion thus substantially improving citizens' quality of life. In these systems, bicycles are made available for shared use to individuals on a very short-term basis. They are rented in a station and returned in any other station with free docks. However, to achieve a satisfactory user experience, all the stations in the system must be neither overloaded nor empty. The occupancy level of the stations can be constantly monitored through IoT-based services. The goal of this work is to analyze occupancy level data acquired from real systems to discover situations of dock overload in multiple stations which could lead to service disruption. The proposed methodology relies on a pattern mining approach. A new pattern type, called Occupancy Monitoring Pattern (OMP), is proposed to characterize situations of dock overload in multiple stations. Since stations are geo-referenced and their occupancy levels are periodically monitored, OMPs can be filtered and evaluated by considering also the spatial and temporal correlation of the acquired measurements. The results achieved on real Smart City data highlight the potential of these techniques in supporting domain experts in maintenance activities, such as periodic re-balancing of the occupancy levels of the stations, as well as in improving the user experience, such as suggesting alternative stations in the neighborhood.

Keywords: Smart Cities; Internet Of Things; Bicycle Sharing Systems; Machine Learning; Association Rule Mining

1. Introduction

With the advent of Internet of Things (IoT) monitoring networks, Smart City environments have enabled a vital connection between mobility service providers and customers. IoT-based services in urban environments allow, for instance, providers to monitor customer movements and keep track of user subscriptions to various services as well as monitor the user exploitation of the provided services.

Analyzing mobility data is particularly useful for enhancing the quality of the offered services and for improving the citizens' perception of the high quality standards. (e.g., planning travels [1], monitoring pollutant levels [2]). However, handling data in heterogeneous IoT-based networks poses significant challenges related to data management and analysis [3]. The key challenges in urban computing from the perspective of computer scientists are summarized in [4]. According to the classification given in [4], the urban computing application presented in this work addresses the *human mobility scenario*.

In recent years municipalities have fostered alternative ways of public transportation in order to reduce pollution and traffic congestion. Bicycle sharing systems [5] are a notable example of eco-friendly transportation systems, where citizens can rent bicycles on a short-term basis. Bicycles are retrieved from stations spread throughout the city. Each station has a maximum capacity as it is equipped with a fixed number of docks. Citizens can rent a bicycle parked at any station and return it to any other station with free docks. However, to achieve a satisfactory user experience, system

© 2018 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

managers should carefully monitor the level of occupancy of the stations. For example, if a station is frequently overloaded at business hours then re-balancing actions should be scheduled in order to move some of the parked bicycles to any station located in the neighborhood. In case the problem is more severe, managers may decide to expand the station to fit the increasing demand.

To constantly monitor their level of occupancy, stations are geo-referenced and equipped with sensors interconnected through IoT devices. Each station tracks the occupancy levels of its docks and shares geo-referenced time series data through the IoT network. The occupancy level data acquired from the stations can be collected and stored in a unique repository and analyzed by means of machine learning and data analytics techniques. Automating the process of analysis of the acquired occupancy level data is particularly appealing to computerize the planning of maintenance activities as well to give targeted recommendations to the system users [4].

The analysis of urban data related to bicycle sharing systems has already been addressed in previous studies. Specifically, in this field the main branches of research can be categorized as follows. (i) Grouping stations based on their usage profile [6–8]. (ii) Predicting future station occupancy levels [9–13]. (iii) Repositioning bicycles among the stations [14–18].

Branch (i) focuses on discovering groups of stations with different usage profiles by applying unsupervised machine learning techniques (e.g., clustering [6]). To characterize station usage, temporal features [6], spatial features [7], or a mix of the above [8] are considered. Instead of partitioning the set of stations into disjointed groups according to their common usage pattern, the methodology proposed in this study focuses on discovering sets of nearby stations showing a critical or alternate usage profile (e.g., a station is overloaded whereas the nearby station is almost empty). To the best of our knowledge, these patterns cannot be detected by existing approaches.

Branch (ii) aims at forecasting the occupancy level of a station in the near future by applying supervised machine learning techniques (e.g., regression [9–11,19], classification [12,13]). Based on these predictions, a recommender system can be integrated into the mobile application of the provider to suggest the stations close to the user-specified point of interest with a sufficient number of free docks/available bicycles. Predictions are based not only on past occupancy levels but also on contextual information (e.g., meteorological data [19]). Predicting occupancy levels based on historical data is not contemplated in this study.

Branch (iii) focuses on planning the re-balance of the bicycles in the stations according to the actual user needs (e.g., more bicycles close to parking areas and business centers or more free docks close to restaurants at lunchtime). The aim is to support providers in improving user experience. The problem is complementary to the one addressed in this paper, because detecting dock overload situations could trigger re-balance actions driven by optimization-based strategies such as [14,18].

This work presents a novel exploratory data-driven methodology, named *B*ike Station Ov*E*rLoad Ana*L*yzer (BELL), to analyze the occupancy levels of the stations of a bicycle sharing system. The aim is to characterize situations of dock overload in multiple stations which could lead to either service disruption or low customer satisfaction. For example, when all the docks in a station are occupied, users have to move to a nearby station to park their bike. Gathering insightful information about the occupancy levels of multiple stations can effectively support domain experts in applying targeted actions to avoid and/or limit the unpleasant situations described above. For example, the mobile application of the system may recommend alternative nearby stations with free docks. Furthermore, the maintenance service may re-balance the number of bikes in each station thus avoiding overloaded conditions.

In the BELL methodology occupancy level data acquired from the geo-referenced stations are analyzed to discover a new type of patterns, called *Occupancy Monitoring Patterns* (OMPs). OMPs synthetically represent situations of imbalance in the occupancy levels of spatially correlated stations. Specifically, OMPs model two complementary dock overload situations: (i) Situations in which a set of stations are overloaded in an alternate fashion (hereafter denoted as *intermittent situations*), and (ii) Situations in which the docks of a set of stations are frequently overloaded at the same time

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521

(hereafter denoted as *critical situations*). To consider the spatial correlation between the occupancy level of different stations, spatial constraints can be enforced to represent in OMPs only groups of nearby stations (i.e., stations with a limited geographical distance).

Intermittent and critical situations are treated separately because they cause disservices with varying degree of severity for end users. Specifically, intermittent situations indicate an imbalance in station usage which could be addressed by proposing an alternative nearby stations to end users or by periodically repositioning the bicycles in the neighborhood. Conversely, critical situations indicate that a given area is temporarily inaccessible for parking bikes because all the stations in the area are in a dock overload situation. The latter (more severe) situation can be addressed, for example, by increasing the number of available docks in the stations, or by moving bikes to the not fully occupied stations located in other city areas.

The generated OMPs are explored to discover significant intermittent and critical situations. The exploration is driven by two ad hoc quality indices introduced in this study, namely the *intermittence* and the *criticality*. The proposed indices allow domain experts to focus on the most severe warnings. Furthermore, OMPs allows a spatio-temporal exploration of critical and intermittent situations. On the one hand, since stations are geo-referenced objects, OMPs represent the city areas where disservices are likely to occur. On the other hand, since OMPs can be related to specific time periods, they allow experts to identify the periods in which disservices are likely to be more severe.

The proposed BELL methodology generates OMPs by means of a two-step itemset-based process, which is driven by two proposed quality indices. In this study, BELL was thoroughly evaluated using a real dataset acquired from the bicycle sharing systems of two important Smart Cities, i.e., Barcelona (Spain) and New York (USA). The experimental results demonstrate the effectiveness of BELL in identifying interesting knowledge regarding the spatio-temporal distribution of possible service disruptions for end users of bicycle sharing systems. We envisioned possible scenarios of usage of the extracted patterns aimed at supporting maintenance activities and improving user experience.

This paper is organized as follows. Section 2 presents and thoroughly describes the proposed approach. Section 3 experimentally evaluates the performance of our implementation of the BELL methodology on data acquired in real urban environments and Section 4 draws conclusions and presents future developments of this work.

2. Methodology

*B*ike Station Ov*E*r*L*oad Ana*L*yzer (BELL) is a new data mining methodology aimed at monitoring the occupancy levels of the stations in a bicycle sharing system. The main architecture blocks, depicted in Figure 1, are (i) *Data acquisition, enrichment, and preparation*, (ii) *Pattern extraction*, which entails discovering OMP patterns from the prepared data, and (iii) *Knowledge exploration and exploitation*, which entails exploring the extracted OMPs to discover actionable knowledge. A more thorough description of each step is given in the following sections.

2.1. Data collection, modeling and enrichment

To monitor the usage of the bicycle sharing system, the occupancy levels of all the stations are acquired at different points of time and stored into a *Occupancy Level Dataset* ($\mathcal{D}$). Collected data are then enriched with additional spatial and temporal information needed to support the subsequent data analysis phase.

Data collection and modeling. Given a time window TW and a set TS=$\{t_1, \ldots, t_n\}$ of points of time in TW, for each station s_i in the system the number of free parkings at each time $t_i \in TW$ is acquired and collected in a unique repository named *Occupancy Level Dataset* ($\mathcal{D}$).

Dataset $\mathcal{D}$ has been modeled as a relational dataset. A relational dataset consists of a set of records [20]. Each record is a set of pairs *(attribute_name, value)*. While *attribute_name* is the description of a data feature, *value* is the corresponding value. In our context, attribute names represent stations,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521

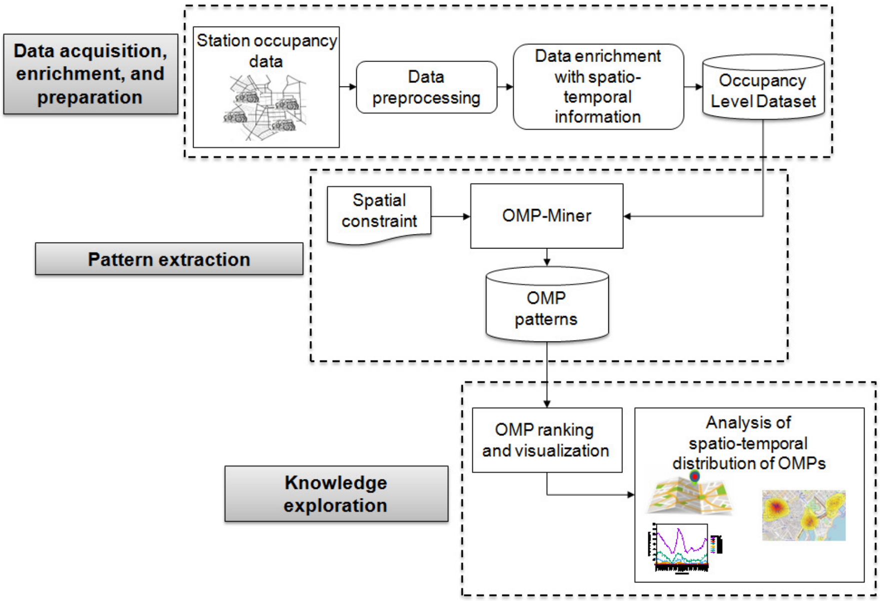

Data acquisition, enrichment, and preparation
Station occupancy data
Data preprocessing
Data enrichment with spatio-temporal information
Occupancy Level Dataset
Pattern extraction
Spatial constraint
OMP-Miner
OMP patterns
Knowledge exploration
OMP ranking and visualization
Analysis of spatio-temporal distribution of OMPs

Figure 1. The *B*ike Station Ov*E*r*L*oad Ana*L*yzer architecture.

while attribute values are the occupancy levels of the corresponding stations. Hence, in the *Occupancy Level Dataset* ($\mathcal{D}$) each record consists of the occupancy levels of all the stations acquired at a different point of time $t_i \in TS$.

Station occupancy values are then categorized into two different classes to indicate the occupancy level of the station. Specifically, the measurements indicating the number of free parkings at a station are labeled as follows: (i) *Overloaded*, if the number of freely available parkings is below a given occupancy threshold *full-th*, or (ii) *Normal*, if the number of freely available parkings is equal to or above *full-th*. The occupancy level threshold *full-th* is an absolute value specified by the domain expert. Label *Overloaded* is used to denote stations with a critical occupancy level, such that end users may not find free docks for parking. Instead, label *Normal* is used to denote station conditions that should not cause a disservice to end users.

Table 1 shows a toy example of an occupancy level dataset. The dataset stores the occupancy levels of three arbitrary stations (s_1, s_2, s_3) at seven points of time (t_1-t_7). For example, record with identifier RID=3 is {(s_1, *Overloaded*), (s_2, *Overloaded*), (s_3, *Normal*)}; it stores the occupancy levels of the three stations at point of time t_3. Hence, at time t_3 stations s_1 and s_2 are overloaded, i.e., their number of free parkings is less than *full-th*. Conversely, the occupancy level of the station s_3 is normal, i.e., the number of free parkings is above or equal to *full-th*.

Notice that this study will not address the complementary problem of detecting sets of underutilized stations. However, since our proposed methodology is general, it can be straightforwardly adapted to deal with this complementary problem.

Data enrichment with temporal information. The analysis of station occupancy levels at different time granularities allows system managers to investigate how overload conditions evolves over time, and identify overload conditions that frequently happen in specific time periods. To support this analysis, the occupancy level data have been enriched with a temporal information with a coarser granularity.

In dataset $\mathcal{D}$ each record includes the occupancy levels of all the stations acquired at a different point of time $t_i \in TS$. Each record is enriched with an additional attribute specifying the corresponding *time period TP* for the point of time t_i. In the example dataset in Table 1, records are associated with three different time periods denoted as TP_1, TP_2, and TP_3. The granularity of the time period can

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

Table 1. Example of *Occupancy Level Dataset* ($\mathcal{D}$).

		Stations			
RID	**Timestamp**	s_1	s_2	s_3	**Time period**
1	t_1	Overloaded	Overloaded	Overloaded	TP_1
2	t_2	Overloaded	Normal	Overloaded	TP_1
3	t_3	Overloaded	Overloaded	Normal	TP_1
4	t_4	Overloaded	Normal	Normal	TP_1
5	t_5	Normal	Overloaded	Normal	TP_2
6	t_6	Normal	Overloaded	Normal	TP_2
7	t_7	Normal	Normal	Normal	TP_3

be defined based on the target analysis. For example, hourly or daily time slots can be selected as reference time periods to monitor dock overload situations during the day.

Data enrichment with spatial information. To detect dock overload situations restricted to a given area, we enrich occupancy level data with spatial information. Since all the stations in the system are geo-referenced, the geographical coordinates of all the stations in the system is collected. This information is used in our approach to compute the pairwise distances between pairs of stations.

2.2. Pattern characterization

To automatically detect recurrent situations of dock overload condition in multiple stations, we propose a new type of pattern, called the *Occupancy Monitoring Pattern* (OMP). OMPs represent sets of stations showing a dock overload condition which may cause a disservice on the usage of the bicycle sharing system for the end users. More specifically, OMPs represent the following situations.

- *Critical situation.* The occupancy levels of a group of stations are frequently overloaded at the same time. In this case, simultaneously, *all* the stations in the group are fully occupied.
- *Intermittent situation.* The occupancy levels of a group of stations are frequently overloaded in an alternate fashion. At a given point of time, *some* stations are fully occupied whereas the other ones are almost empty. At another point of time, the occupancy level of the same stations could be opposite.

To consider only sets of nearby stations, given by stations with a limited geographical distance in the city area, a *spatial constraint* can be enforced. Enforcing such a constraint implies that the OMPs consisting of stations with maximal geographical distance below a given (analyst-provided) threshold.

Critical situations are potentially harmful because when all the stations in the group are overloaded users cannot return the rented bicycles. While considering sets of nearby stations, in particular, the discovery of a group of overloaded stations implies that a specific city area is temporarily unaccessible. To quantitatively evaluate the severity of the issue we introduced a measure denoted as *criticality*. This measure counts the number of recorded timestamps (i.e., the number of dataset records) at which *all* the stations of the considered monitoring pattern have a critical level of occupancy.

Intermittent situations are potentially harmful as well, because the stations in the group are overloaded in an alternate fashion. While considering nearby stations, some free docks are available in the corresponding area, but a potential service disruption may occur when a user arrives at an overloaded station. Still, the user could reach any of the close stations, among which some of them are underutilized. To quantitatively estimate the severity of an intermittent situation, we introduced the *intermittence* measure. Intermittence counts the number of points of time at which *at least one* station (but not all of them) of the considered monitoring pattern has an occupancy level above a given threshold. The higher the intermittence, the more severe the imbalance situation.

A more formal definition of the Occupancy Monitoring Pattern (OMP) and its quality measures follows.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521

Definition 2.1 (Occupancy Monitoring Pattern (OMP))**.** Let $\mathcal{D}$ be an occupancy level dataset and let $\mathcal{S}$ be the corresponding set of dataset attributes s_i, where each attribute s_i represents a station of the bicycle sharing system. A Occupancy Monitoring Pattern P in dataset $\mathcal{D}$ is a set of k distinct stations in $\mathcal{S}$, i.e., P=$\{s_1, \ldots, s_k\}$, $s_i \in \mathcal{S}$.

Definition 2.2 (Criticality and Intermittence measures)**.** Let $\mathcal{D}$ be an occupancy level dataset and let P=$\{s_1, \ldots, s_k\}$, $s_i \in \mathcal{S}$, be an arbitrary occupancy monitoring pattern in $\mathcal{D}$. Pattern P is characterized by criticality and intermittence qualitative measures, defined as follows:

- The *criticality* of pattern P is the number of records $r \in \mathcal{D}$ for which all the stations $s_i \in P$ take value *Overloaded*.
- The *intermittence* of pattern P is the number of records $r \in \mathcal{D}$ for which at least one station $s_i \in P$ (but not all of them at the same time) takes value *Overloaded*.

The *relative criticality* (respectively *relative intermittence*) value of pattern P is the ratio between the (absolute) criticality (respectively intermittence) value of P and the number of records in dataset $\mathcal{D}$.

Example. P=$\{s_2, s_3\}$ is an Occupancy Monitoring Pattern consisting of stations s_2 and s_3. Since these stations both take value *Overloaded* at the same time in one dataset record over seven (i.e., record with RID 1), the criticality of pattern P is 1, while the relative criticality of P is $\frac{1}{7}$ (14.28%). Instead, the two stations take value *Overloaded* in an alternate fashion in four records (with RIDs 2, 3, 5, and 6). Hence, the intermittence of pattern P is equal to 4 and the relative intermittence is $\frac{4}{7}$ (57.14%).

From Definition 2.2, it follows that the criticality measure satisfies the anti-monotone property, namely, the criticality for a pattern P never exceeds the criticality for its subsets. For instance, pattern P=$\{s_2, s_3\}$ has criticality 1 because both stations are overloaded in record RID=1. The subset $P' \subset P$, P'=$\{s_2\}$, has criticality equal to 4 since station s_2 has value *Overloaded* in record RID 1 but also in records with RIDs 3, 5, and 6.

The criticality and intermittence measures of OMP patterns can be computed by considering different time frames with the aim of analyzing from various time perspectives the usage of a sets of stations. Specifically, the two measures can be calculated by considering data acquired in the whole monitoring window to catch an overall view of the occupancy levels of groups of stations. Alternatively, they can be computed by considering data acquired for time periods with a finer time granularity. This second option allows to analyze how the occupancy of stations evolves over time as well as detect dock overload situations happening within limited time ranges. Based on the target application, the time period with a suitable time granularity can be selected for monitoring the usage of stations.

Given an occupancy monitoring pattern P, its criticality and intermittence value in a time period TP_k are computed considering only the subset of records with time period equal to TP_K. A more formal definition follows.

Definition 2.3 (Criticality and Intermittence measures in time period TP_k)**.** Let $\mathcal{D}$ be an occupancy level dataset and let $\mathcal{TP}$ be the corresponding set of dataset attributes TP_i, where each attribute TP_i represents a time period in the monitoring time window TW. Let P be an occupancy monitoring pattern in dataset $\mathcal{D}$. The *criticality* of P in a time period $TP_k \in \mathcal{TP}$ is the number of records $r \in \mathcal{D}$ having (i) time period equal to TP_k and (ii) for which all the stations $s_i \in P$ take value *Overloaded*. The *intermittence* of pattern P in TP_k is the number of records $r \in \mathcal{D}$ having (i) time period equal to TP_k and (ii) for which at least one station $s_i \in P$ (but not all of them at the same time) takes value *Overloaded*.

Example. Consider pattern P=$\{s_2, s_3\}$ and time period TP_1 in the example dataset. Focusing on the four records with time period TP_1, the two stations are both in the overload condition in record with RID 1, while they are overloaded in alternative fashion in records with RIDs 2 and 3. Therefore, the absolute and relative criticality of P are equal to 1 and $\frac{1}{4}$ (25%), respectively. The absolute and relative intermittence of P in TP_1 are equal to 2 and $\frac{2}{4}$ (50%), respectively.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521

Occupancy Monitoring Patterns (OMPs) can be filtered based on the spatial distance between the corresponding stations. For this purpose, we introduce a spatial constraint $C(maxdist)$ on OMPs. This constraint specifies the maximum geographical distance (denoted *maxdist*) between stations in each OMP. OMPs satisfying the spatial constraint represent sets of nearby stations showing an overload situation. The higher *maxdist* the larger area including stations with critical/intermittent levels of dock occupancy.

Definition 2.4 (Spatial constraint)**.** Let $\mathcal{D}$ be an occupancy level dataset, *maxdist* be a positive number, and P={s_1, ..., s_k}, $s_i \in \mathcal{S}$, an arbitrary occupancy monitoring pattern in $\mathcal{D}$. Pattern P satisfies the spatial constraint $C(maxdist)$ if for every pair of stations $s_j, s_k \in P$, $j \neq k$, their geographical distance $\mathrm{d}(s_j, s_k)$ is below *maxdist*.

Given an OMP P={s_1, ..., s_k} that satisfies the spatial constraint, every subset $P' \subset P$ satisfies it as well. In fact, if for all pairs of stations $s_t, s_k \in P$ the condition $\mathrm{d}(s_t, s_k) < maxdist$ is verified, it easily follows that the condition is also verified forall pairs of stations in $P' \subset P$. Such a property, called anti-monotonicity property, will be particularly useful for efficiently generating all the OMPs of interest.

In our implementation of the proposed methodology, geographical distances between stations are computed using the well-known Euclidean distance [20].

2.3. Pattern extraction

Given an occupancy level dataset $\mathcal{D}$ and a maximum distance *maxdist*, an algorithm is proposed in this study to efficiently extract all the Occupancy Monitoring Patterns (OMP) and compute their criticality and intermittence values.

The problem of generating OMPs has been addressed as an itemset mining problem. Itemset mining is an exploratory data mining technique which consists of discovering interesting and useful patterns in transactional databases [21]. More specifically, it entails discovering the groups of attribute values that frequently co-occur in the analyzed database. Itemset mining has been applied in various applications in domains such market basket analysis, bio-informatics, text mining, product recommendation, and Web click stream analysis.

To enable the itemset mining process in our target context, the records contained in the occupancy monitor dataset $\mathcal{D}$ are tailored to a transactional data format. To this purpose, we first introduce the concept of *occupancy item* (o-item, in short); next, a record $r \in \mathcal{D}$ is represented in a transactional data format $\mathcal{T}$ as a set of o-items.

An *o-item* represents a dock occupancy measurement acquired within a given time period and associated with a given station. More formally, an o-item is modeled as a triple $\langle s_j, o_i^j, TP_i \rangle$, where s_j is an arbitrary station, o_i^j is the occupancy level of station s_j at any time stamp $t_i \in TP_i$, and TP_i is a time period. Note that the exact time stamp at which the measurement was acquired is not explicitly reported in the o-item, because the goal is to identify the stations that have acquired critical dock occupancy levels within each time period.

Example. Figure 2 reports the transactional representation of the occupancy level monitoring dataset in Figure 1. Each record in the relational dataset is represented as a transaction characterized by the same identification value. For instance, record with RID=2 is represented in transaction with TID=2 as a set of three o-items $\{\langle s_1, Overloaded, TP_1 \rangle, \langle s_2, Normal, TP_1 \rangle, \langle s_3, Overloaded, TP_1 \rangle\}$. For example, o-item $\langle s_1, Overloaded, TP_1 \rangle$ indicates that station s_1 has occupancy level *overloaded* in time period TP_1.

An *occupancy itemsets* (o-itemsets, in short) is a set of o-items (of arbitrary size) such that all the contained o-items correspond to the same time period. The *frequency of an o-itemset* is the number of transactions including it.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

Table 2. Example of *Occupancy Level Dataset* in transactional format ($\mathcal{T}$).

TID	Transactions
1	$\langle s_1, Overloaded, TP_1\rangle$, $\langle s_2, Overloaded, TP_1\rangle$, $\langle s_3, Overloaded, TP_1\rangle$
2	$\langle s_1, Overloaded, TP_1\rangle$, $\langle s_2, Normal, TP_1\rangle$, $\langle s_3, Overloaded, TP_1\rangle$
3	$\langle s_1, Overloaded, TP_1\rangle$, $\langle s_2, Overloaded, TP_1\rangle$, $\langle s_3, Normal, TP_1\rangle$
4	$\langle s_1, Overloaded, TP_1\rangle$, $\langle s_2, Normal, TP_1\rangle$, $\langle s_3, Normal, TP_1\rangle$
5	$\langle s_1, Normal, TP_2\rangle$, $\langle s_2, Overloaded, TP_2\rangle$, $\langle s_3, Normal, TP_2\rangle$
6	$\langle s_1, Normal, TP_2\rangle$, $\langle s_2, Overloaded, TP_2\rangle$, $\langle s_3, Normal, TP_2\rangle$
7	$\langle s_1, Normal, TP_3\rangle$, $\langle s_2, Normal, TP_3\rangle$, $\langle s_3, Normal, TP_3\rangle$

Example. $\{\langle s_1, Overloaded, TP_1\rangle$, $\langle s_3, Overloaded, TP_1\rangle\}$ is an o-itemset with frequency equal to 2, because it occurs in transactions with TIDs 1 and 2. This o-itemset indicates that stations s_1 and s_3 were temporarily overloaded in two different measurements acquired in period TP_1.

Occupancy monitoring patterns and their criticality and intermittence values can be derived from the mined o-itemsets. Therefore, our proposed methodology for OMP mining is based on the following two steps. First, o-itemsets are mined. Then, OMPs are generated on top of the mined o-itemsets and their criticality and intermittence values are computed. In the following the two steps are separately described.

Step 1: O-itemset mining. A set of o-itemsets is extracted from the transactional representation of the occupancy level dataset. Each of the mined o-itemsets satisfies the following conditions. (i) All the contained o-items have the same occupancy level (i.e., all *normal* or all *overloaded*). (ii) All the stations contained in the o-itemset satisfy the spatial constraint $C(maxdist)$. Thus, for every pair of stations appearing in the o-itemset, their geographical distance is below *maxdist*.

Condition (i) allows us to extract two different types of o-itemsets: the *critical o-itemsets*, which include only the o-items with occupancy level *overloaded*, and the *normal o-itemsets*, which include only the o-items with occupancy level *normal*. These o-itemsets combine the stations having all the same occupancy level in a given time period. As discussed below, these two o-itemset types will be useful at the next step to compute the OMP intermittence value. Condition (ii) allows us to filter out the combinations of o-items related to faraway stations. This will allow us to generate only OMPs including nearby stations in Step 2.

Step 2. OMPs generation. The output of Step 1 is processed at Step 2 to generate the set of OMPs. An occupancy monitoring pattern P is generated from a pair of critical and normal o-itemsets that include (i) the same stations and (ii) the same time period. The criticality and intermittence values of P are computed based on the frequency values of these two o-itemsets.

The OMP generation process is here detailed using an example case. Let consider a pair of critical (denoted $o - I_C$) and normal (denoted $o - I_N$) o-itemsets, having both the same stations and the same time period. Consider for instance the o-itemsets $o - I_C$= $\{\langle s_i, Overloaded, TP_k\rangle$, $\langle s_j, Overloaded, TP_k\rangle\}$ and the normal o-itemset $o - I_N$=$\{\langle s_i, Normal, TP_k\rangle$, $\langle s_j, Normal, TP_k\rangle\}$. Let denote as $freq\_value(critical)$ and $freq\_value(normal)$ their respective frequency in time period T_k in the analyze dataset. Let P be the occupancy monitoring pattern generated from these two o-itemsets. The following statements hold.

(i) Pattern P contains all the stations appearing in the critical o-itemset $o - I_C$ (or equivalently in the normal o-itemset $o - I_N$), i.e., P=$\{s_i, s_j\}$.

(ii) According with Definition 2.3, the criticality of pattern P in time period TP_k is the number of times all the stations in P are overloaded in TP_K. It follows that that criticality of P in period TP_k is equal to the number of transactions in TP_k including the o-itemset $o - I_c$. Thus,

$$criticality = freq\_value(critical) \tag{1}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521

(iii) According with Definition 2.3, the intermittence of pattern P in a time period TP_k is the number of times at least one station in P (but not all stations at the same time) is overloaded in TP_k. If follows that the intermittence of P in period TP_k is equal to the total frequency of all o-itemsets with the same stations as P, such that at least one station (but not all them at the same time) is overloaded in TP_k. For the sake of efficiency, our approach avoids generating all these o-itemsets, but instead it proceeds as follows. Let denote as *card_value* the total number of transactions in period TP_k in the analyzed dataset. It easily follows that *card_value* is equal to the sum of the following three terms: the frequency of the critical o-itemset $o - I_C$ ($freq\_value(critical)$), the frequency of the normal o-itemset $o - I_N$ ($freq\_value(normal)$) and the total frequency of all o-itemsets with the same stations as P, such as at least one station (but not all them at the same time) is overloaded at time TP_k. Therefore, we compute the intermittence of P in period TP_k as

$$intermittence = card\_value - (freq\_value(critical) + freq\_value(normal)) \tag{2}$$

Example. Pattern P={s_2, s_3} has criticality equal to 1 and intermittence equal to 2 in time period TP_1. These measures are computed based on the frequencies of the critical o-itemset $\{\langle s_2, Overloaded, TP_1\rangle, \langle s_3, Overloaded, TP_1\rangle\}$ and of the normal o-itemset $\{\langle s_2, Normal, TP_1\rangle, \langle s_3, Normal, TP_1\rangle\}$. The critical o-itemset has frequency equal to 1 being contained in transaction TID=1. Thus, the criticality of P is equal to freq_value(critical) =1. The normal o-itemset has frequency equal to 1 since it is included in transaction TID=4 (i.e., freq_value(normal) =1). card_(value) is equal to 4 because four transactions refer to time period TP_1. Based on Equation 2, it follows that the intermittence of P is computed as intermittence = 4 - (1+1) = 2. This intermittence value corresponds to the total frequency of the o-itemsets $\{\langle s_2, Normal, TP1\rangle, \langle s_3, Overloaded, TP1\rangle\}$ and $\{\langle s_2, Overloaded, TP1\rangle, \langle s_3, Normal, TP1\rangle\}$, respectively contained in transactions TID=2 and TID=3.

In next Section 2.3.1, we describe the algorithm used in the BELL framework to mine the occupancy patterns including nearby stations according to the spatial constraint *maxdist* as well their criticality and intermittence values.

2.3.1. Algorithm

Algorithms 1 and 2 report the pseudo-code of the algorithm we devised to extract OMPs. It consists of the following three main phases:

- Phase 1: Creation of a compact in-memory representation of the occupancy level transactional dataset (Algorithm 1, line 1).
- Phase 2: Mining of all the critical and normal o-itemsets including nearby stations according to the spatial constraint *maxdist* (Algorithm 1, line 2).
- Phase 3: Generation of the OMPs on top of the mined o-itemsets and computation of their criticality and intermittence levels (Algorithm 1, lines 3-7).

To implement the o-itemset mining phase of the proposed methodology, we exploited an itemset-based approach relying on the established FP-growth algorithm [22]. FP-Growth is a popular algorithm to mine frequent itemsets from transactional data, which has been proposed to scale the traditional Apriori-based approach [23] towards large datasets. The key idea is to compactly represent item co-occurrences in the dataset transactions in a tree-based structure and to recursively visit the tree instead of the original dataset. The main advantage of the FP-growth approach is the selective generation of the candidate itemsets, which prevents the time- and memory-consuming candidate generation phase adopted by the Apriori strategy [23].

Phase 1 entails storing the measurements reported in the transactional representation $\mathcal{T}$ of the original dataset into a compact tree-based structure. To accomplish this task, we exploit the tree-based data structure adopted by FP-Growth, namely the FP-Tree, to store the transactional dataset $\mathcal{T}$ (see Line 1 at Algorithm 1). In our context, each node of the tree is an o-item. A transaction in $\mathcal{T}$ is stored in the FP-tree as a path connecting o-items corresponding to the same time period. A path in the FP-tree

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521

Algorithm 1 OMP-Miner($\mathcal{T}$, $maxdist$, $\mathcal{P}$)

```
Require: $\mathcal{T}$: occupancy level dataset in transactional format
Require: $maxdist$: maximum distance between two stations in the same OMP
Require: $\mathcal{TP}$: set of time periods $TP_1, \ldots, TP_q$
Ensure: $\mathcal{P}$: the set of OMPs for each time period in $\mathcal{TP}$
1: $Tree \leftarrow$ FP-tree($T$) { Create the initial FP-tree from $\mathcal{T}$ }
2: $\mathcal{F} \leftarrow$ O-ITEMSETMining($Tree$, $maxdist$, $\emptyset$) { Recursive projection-based o-itemset mining function}
   { Generate OMPs on top of the mined o-itemsets in $\mathcal{F}$}
3: $\mathcal{F}_{normal}$: normal o-itemsets in $\mathcal{F}$
4: $\mathcal{F}_{critical}$: critical o-itemsets in $\mathcal{F}$
5: $\mathcal{H}$: Hash map with keys $\langle o\text{-}I, TP_k \rangle$ storing the criticality values of each normal o-itemset $o\text{-}I$ in
   $\mathcal{F}_{normal}$ for each period $TP_k$
6: $card\_value$[]: vector storing in the $k$-th element the number of transactions in $\mathcal{T}$ associated with
   period $TP_k$
7: $\mathcal{P}$ = ComputeOMPintermittence($\mathcal{F}_{critical}$,$\mathcal{H}$,$card\_value$)
8: return $\mathcal{P}$
```

from a leaf node to the root node represents all the occupancy level measurements associated with different stations at a given time period. The key advantage of scanning the FP-tree index instead of the original dataset in the o-itemset mining process is that in the FP-tree it is possible to store multiple dataset transactions containing the same o-items in the same path.

Phase 2 entails generating all the critical and normal o-itemsets including only nearby stations (see Line 2 at Algorithm 1). The o-itemset mining phase relies on the recursive FP-tree visit adopted by FP-Growth. However, instead of generating all the possible o-itemsets, in the proposed approach the anti-monotonicity property of the spatial constraint (see Section 2.2) is exploited to reduce the number of generated combinations. Specifically, each o-itemset refers to a given combination of stations, where for each pair of stations their geographical distance is known. To consider only the sets of nearby stations, we enforce the maximal distance constraint *maxdist* during the o-itemset generation. This prevents the generation of a (potentially very large) set of uninteresting o-itemsets and OMPs. To push the maximal distance constraint into the FP-Growth itemset mining process, during the projection phase we do not include in the pattern base conditioned to a given o-item all the o-items including any stations with distance above *maxdist*.

Phase 3 aims at generating OMPs by properly combining the critical and normal o-itemsets mined at Phase 2 and stored in sets $\mathcal{F}_{critical}$ and $\mathcal{F}_{normal}$, respectively. For each critical o-itemset $o\text{-}I_C \in \mathcal{F}_{critical}$, an occupancy pattern P is generated with criticality equal to the frequency (freq_value(critical)) of $o - I_C$ (according to Equation 1). The intermittence value of P is computed according to Equation 2 by considering the number of transactions with time period TP_K ($card_value$), the frequency value in TP_k of the critical o-itemset $o\text{-}I_C$ (freq_value(critical)), and the frequency value in TP_K of normal o-itemset $o\text{-}I_N \in \mathcal{F}_{normal}$ including the same stations as $o\text{-}I_C$.

To efficiently compute the pattern intermittence value, the normal o-itemsets and their corresponding frequency values are stored in a hash map data structure. Given a critical o-itemset, the corresponding normal o-itemset including the same stations is returned by the hash map given the key $\langle oi, x \rangle$ (Algorithm 1, line 7).

2.4. Knowledge exploration and exploitation

The patterns extracted with the OPM-Miner algorithm can be explored by system managers to gain insight into system usage. This explorative analysis allows domain experts to focus their attention on a limited number of stations on given areas and in specific time periods. Based on the mined

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

Algorithm 2 O-ITEMSETMining(*Tree*, *maxdist*, *pj*)

```
Require: $Tree$, an FP-tree
Require: $maxdist$: maximum distance between two stations in the same o-itemset
Require: $pj$, the set of o-items with respect to which $Tree$ has been generated
Ensure: $\mathcal{F}$, the set of o-itemsets extending $pj$
$\mathcal{F} \leftarrow \emptyset$
for all o-item $i^*=\langle s_j, o_i^j, TP_i \rangle$ in the header table of $Tree$ such that for each pair of stations $s_j, s_x \in pj$ distance($s_j$, $s_x$)<$maxdist$ do
    $o\text{-}I \leftarrow pj \cup \{o\text{-}i^*\}$ { Generate a new potential o-itemset $I$ by joining $pj$ and $o\text{-}i^*$ }
    $\mathcal{F} \leftarrow \mathcal{F} \cup \{o\text{-}I\}$
    $Tree_{o-I} \leftarrow$ createFP-tree($Tree$, $o\text{-}I$) { Build $o\text{-}I$'s conditional FP-tree }
    if $Tree_{o-I} \neq \emptyset$ then
        $\mathcal{F} \leftarrow \mathcal{F} \cup$ O-ITEMSETMining($Tree$, $maxdist$, $px$) { Recursive mining}
    end if
end for
return $\mathcal{F}$
```

knowledge, domain experts may recommend targeted maintenance actions with the aims of reducing disruption to end users. To effectively explore the mining result a list of advice is given below.

Exploration of intermittent situations. To detect significant intermittent situations, patterns should be ranked by decreasing intermittence value. To ease pattern exploration, the OMPs with very low intermittence value can be discarded. Patterns with maximal intermittence value indicate groups of stations that are frequently fully occupied in an alternate fashion. These patterns represent station occupancy level conditions that could result in a limited disservice to the end user. If the stations in the pattern are located in the same area, then an alternative arrival station can be recommended to users who reach an occupied station. The severity of the possible disservices for end users can vary based on the criticality value of the pattern. When the pattern criticality level increases, the stations indicated by the pattern are more frequently fully occupied at the same time; thus, end users are unlikely to find a free dock at nearby stations.

To avoid disservices, system managers can suggest an alternative nearby station with free docks for parking; in case of patterns with high intermittence but low criticality values, bicycles may be repositioned in nearby stations because they are rarely fully occupied at the same time.

Exploration of critical situations. In order to detect significant critical situations which could lead to serious disservice for end users, patterns should be ranked by decreasing criticality value. To ease pattern exploration, the OMPs with very low criticality value can be discarded. Patterns with maximal criticality value indicate groups of nearby stations that are frequently fully occupied at the same time. Thus, end users are unlikely to find free docks for their bikes in this area.

Since nearby stations are all fully occupied, maintenance actions such as bicycle repositioning should be carried out considering stations that are further away or located in other areas of the city. Therefore to address these issues, maintenance actions could be much more expensive or even inapplicable. Alternative actions could be considered such as planning station resizing or system enlargement.

Exploration of the spatio-temporal distribution of intermittent and critical situations. To support management of the bicycle sharing system, the mined patterns can be visualized on a map of the city area. Since each station in the pattern is characterized by a geographical position, patterns can be represented as restricted city areas including the corresponding stations. This representation is intuitive and effective for highlighting the areas which could lead to disservices for end users. Pattern representations can be differentiated based on the type of imbalance in station occupancy (i.e., critical,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

intermittent) and the degree of severity of the discovered pattern. Domain experts can also analyse intermittent and critical situations for different values of time periods to identify the time frames associated with more serious disruptions. For example, they can consider 1-hour time slot as time period to analyse the number and significance of intermittent and critical situations for each hour in a day. Alternatively, they can adopt a courser time granularity, as a larger time slot size (e.g., morning, afternoon, evening, night), to gather a more high-level view of the dock overload conditions in the bicycle sharing system.

Domain experts are recommended to adhere to the following guideline in order to properly set up the OMP-Miner algorithm. The spatial constraints *maxdist* should be set according to the geographical distribution of the stations in the city area. For example, stations located at walking distance can be considered as *near* while stations located in different districts can be classified as *distant*. To ensure that the extracted patterns include only close stations, the user should set *maxdist* as the largest distance between a pair of *nearby* stations.

Some examples patterns representing significant intermittent and critical situations in real data collections, and the analysis of their spatio-temporal distribution, are reported in Section 3.

3. Experimental results

The efficiency and usability of the BELL system on real data acquired from bicycle sharing systems were validated in two important Smart Cities: Barcelona, the capital city of the autonomous community of Catalonia and Spain's second most populated city and New York, the most populated city in the United Stated of America.

The experimental evaluation addresses the following aspects. Some examples of interesting occupancy patterns representing significant intermittent and critical situations, extracted from the analyzed data collections, are presented in Section 3.2. Section 3.3 evaluates the impact of the system configuration parameters on the number of mined patterns and on their corresponding intermittence and criticality values, while Section 3.4 reports performance evaluation in terms of execution time for the OMP-Miner algorithm. The main characteristics of the analyzed datasets are summarized in Section 3.1.

The OMP-Miner algorithm was implemented by using the C language. The experiments were performed on a 2.67 GHz six-core Intel(R) Xeon(R) X5650 machine with 32 Gbyte of main memory running Ubuntu 12.04 server with the 3.5.0-23-generic kernel.

3.1. Reference use case datasets

This section briefly presents the main characteristics of the two bike sharing systems considered as reference use case in this study and describes data that we have considered on the system usage.

The *Bicing* system in Barcelona. *Bicing* is the bicycle sharing system in Barcelona which consists of 377 stations distributed all over the city area. Stations have a fixed number of parkings, which vary from 15 to 39. A description of the service is given in [9]. Data from the *Bicing* website[1] can be crawled through the Google Maps APIs. To perform our analyses, the collection of measurements described in [9] have been taken into account. The acquired data include 30 million records from the *Bicing* stations over a period of approximately a semester of service (i.e., between May 15th and November 30th, 2008). Occupancy values were acquired every 5 minutes.

The *Citi Bike* system in New York. *Citi Bike* is the bicycle sharing system in New York which features thousands of bikes at 528 stations across New York and Jersey City. Bicycles are available 24/7, 365

[1] www.bicing.com/localizaciones/localizaciones.php

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521

days a year. More information about the system are available at https://member.citibikenyc.com/. The *Citi Bike* system provides open data in the JSON format through the *Citi Bike* station feed service[2]. To perform our analyses, an ad hoc Web crawler was developed which downloaded and parsed the JSON data from the *Citi Bike* system feed to retrieve the historical occupancy data. Occupancy values were acquired every 5 minutes over a time period of approximately 13 months (i.e., between October 23th 2014 and November 17th, 2015).

Characteristics of the collected data on the system usage. In both bicycle sharing systems, each station is characterized by the information on its *name* and *geographic coordinates* (latitude and longitude). *Historical data* on station occupancy can be collected by submitting periodical requests to the stations in the system and storing the corresponding responses. Specifically, for each station we acquired the information on the *number of free* and *occupied slots in different time instants* within a given time window.

3.2. Examples of monitoring occupancy patterns

Following, some occupancy patterns are discussed as representative examples of the insights mined through our framework. Specifically, some top ranked patterns with maximal intermittence and criticality values are discussed as reference cases. These patterns represent dock overload conditions that could yield to disservices for end users in the usage of the bicycle sharing system.

Occupancy monitoring patterns were extracted from the *Bicing* and *Citi Bike* datasets using a standard system configuration with *maxdist* = 0.5 km, *full-th*=3, and *time period* equal to *time slot size* of 1 hour. This configuration pinpoints a time-space granularity suitable to provide useful information to end users and system managers. For example, we set *maxdist* = 0.5 km because bikers are (usually) more willing to move to physically closed stations if the expected destination is fully occupied. We set the time period equal to *time slot size*=1h to determine more precisely sets of nearby stations that could lead to service disruption. Parameter *full-th* has been set to 3 to represent situations when the station is (almost) full. The impact of the system parameters on the characteristics of the extracted patterns is discussed in Section 3.3.

Example patterns with maximal intermittence. Tables 3 and 4 report some examples of top ranked patterns with maximal intermittence value extracted from the *Bicing* and *Citi Bike* datasets, respectively. In both tables patterns are sorted by decreasing intermittence value.

The example patterns from the *Bicing* dataset (Tables 3) are characterized as follows.

Patterns with IDs 5-7 represent dock overload conditions that could yield a *limited disservice* for end users. Each of these patterns represents a group of stations that the end user is likely to find fully occupied in alternate fashion (in about 62-63% of the recorded timestamps according to the intermittence value). However, the low criticality values of these patterns point out that the stations in each pattern are rarely fully occupied at the same time (in about 0.13%-1.56% of the cases). It follows that, in case the user is unable to park in one station she/he can move to one of the other nearby stations where free parking docks will, with a high probability, be available. For example, pattern with ID 5 indicates that the usage levels of stations *Carrer de Bonavista* and *Pl. del Poble Romaní* are critical in an alternate fashion from 7am to 8am in 63% of the cases, but they are fully occupied at the same time only in 1.56% of the cases.

On the other hand, patterns with IDs 1-2 represent dock overload conditions that could result in *a more serious disservice* for end users. Each of these patterns models a group of stations having both intermittence and criticality values higher than patterns with IDs 5-7. For each pattern, at least one station has a high probability of being occupied (intermittence value higher than 71%), and all stations have a not negligible probability of being fully occupied at the same time (criticality about 8%).

[2] http://www.citibikenyc.com/system-data

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

Therefore, in case the user cannot park in one station, she/he might not find a free dock at a nearby station approximately 8% of the time. As an example, pattern with ID 1 shows that, from 4am to 5am , stations *Vilamara davant*, *Mallorca* and *Calabria* have a critical usage level in an alternate fashion in 73.84% of the recorded timestamps, and they are simultaneously fully occupied in 8.29% of the cases.

Patterns with IDs 3-4 represent an intermediate condition between the two above. These patterns have intermittence and criticality values higher than patterns with IDs 5-7 (intermittence 70%-71% instead of 63% and criticality 1.86-4% instead of 0.13%-1.56%), but lower than patterns with IDs 1-2 (intermittence 70%-71% instead of 73% and criticality 1.86%-4% instead of 8%).

Based on the mined knowledge, domain experts may recommend an alternative nearby station for parking and/or targeted maintenance actions. For instance they may decide to relocate bicycles at the beginning of the time slot, moving them from stations with critical levels to non-critical stations.

Regarding the *Citi Bike* dataset (Table 4), the top ranked patterns mined from it have very high intermittence values (between 90% and 100%) and criticality equal to 0%. For example, pattern with ID 2 consists of four nearby stations (*{W 33 St & 8 Ave, W 29 St & 9 Ave, W 31 St & 8 Ave, Penn Station Valet}*) with 100% intermittence and 0% criticality from 8pm to 9pm. These stations are close to Madison Square Garden Stadium and Pennsylvania Station, which are big subway and train hubs. These type of patterns indicate conditions which could yield to a *limited disservice* for end user. On the one hand, since the pattern intermittence value is very high, at least one of the stations in the pattern is likely to be fully occupied. While on the other hand, since the criticality value is 0%, at least one station has a free dock in all the recorded timestamps. Consequently, the user will probably find a free dock among nearby stations.

Example patterns with maximal criticality. Table 5 reports the top ranked patterns with maximal criticality value. These patterns are mined from the *Bicing* dataset considered here as the representative case. Due to the lack of space, similar results achieved on *Citi Bike* dataset have been omitted.

Patterns in Table 5 represent potentially *severe disservices* for end users, because they identify groups of nearby stations whose levels of usage are *all* critical at the same time. For example, pattern with ID 1 indicates that from 10am to 11am stations *Marquas de l'Argentera* and *Avinguda del Marques Argentera* (approximated distance 300m) both have critical usage levels in approximately 38% of the recorded timestamps. Thus, one third of the time parking is unavailable in the mentioned areas and time slot. If the problem persists, users working or living in the neighborhood are strongly discouraged from using the service. Since nearby stations are all fully occupied, maintenance actions such as bicycle repositioning should be carried out considering stations that are further away or located in other areas of the city. Therefore, in order to address these issues, maintenance actions could be much more expensive or even not feasible.

Hourly distribution of intermittent/critical patterns. Figures 2 and 3 show the hourly distribution of the number of monitoring patterns and their corresponding levels of intermittence and criticality. The two figures report, for each hourly time slot, the *total number* of mined patterns characterized by different ranges of intermittence and criticality values. In order to identify patterns that could lead to a disservice for end users, patterns with an intermittence/criticality value greater than or equal to 20% have been taken into consideration .

In the *Bicing* dataset (Figure 2) a significant number of patterns with intermittence/criticality values greater than or equal to 20% occur in all hourly time slots. However, patterns with higher values of intermittence/criticality mainly occur between 1am-2am, 7am-1pm and 4pm-11pm.

Patterns mined from the *City Bike* dataset (Figure 3) show a similar hourly distribution to patterns from the *Bicing* dataset. However, a lower number of patterns with high intermittence/criticality values appear in New York, probably because the stations of the *Citi Bike* system in New York are more

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

Table 3. *Bicing* (Barcelona). Groups of stations with maximal intermittence in different hourly time slots.

Id	Monitoring pattern	Time slot	Crit. %	Interm. %
1	{Vilamara davant, Mallorca, Calabria}	[4am,5am]	8.29	73.84
2	{Vilamara davant, Mallorca, Calabria}	[2am,3am]	8.58	73.53
3	{Sant Pere Mas Alt, Pl. Carles Sunyer, Pl. Catalunya, Pl. Urquinaona}	[10am,11am]	1.86	71.28
4	{Pl. Catalunya A, Pl. Catalunya B, Pl. Catalunya C, Pl. Urquinaona}	[11am,12am]	4.31	70.72
5	{Carrer de Bonavista, Pl. del Poble Romaní}	[7am,8am]	1.56	63.05
6	{Carrer del Cana, Pl. del Poble Romaní}	[5am,6am]	0.13	62.69
7	{Pl. del Poble Romaní, Montmany}	[6am,7am]	0.13	62.41

Table 4. *Citi Bike* (New York). Groups of stations with maximal intermittence in different hourly time slots.

Id	Monitoring pattern	Time slot	Crit. %	Interm. %

1	{W 42 St & 8 Ave, PABT Valet}
	PABT Valet}
		[7pm,8pm]	0	100

2	{W 33 St & 8 Ave, W 29 St & 9 Ave, W 31 St & 8 Ave, Penn Station Valet}	[8pm,9pm]	0	100
3	{W 41 St & 8 Ave, W 45 St & 9 Ave, W 42 St & 8 Ave, PABT Valet}	[7pm,8pm]	0	100
4	{W 42 St & 8 Ave, PABT Valet}	[6pm,7pm]	0	93.7
5	{E 22 St & Broadway, E 24 St & Park Ave}	[11am,12am]	0	90

Table 5. *Bicing* (Barcelona). Groups of stations with maximal criticality in different hourly time slots.

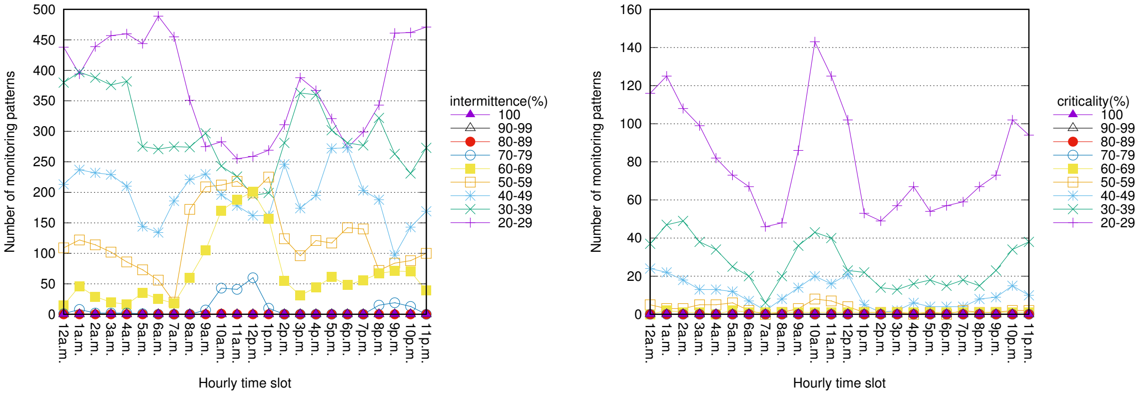
Id	Monitoring pattern	Time slot	Crit. %	Interm. %
1	{Marquas de l'Argentera, Avinguda del Marques Argentera}	[10am,11am]	37.96	19.23
2	{Gran Via, Rocafort}	[11am,12am]	35.94	19.91
3	{Gran Via, Rocafort}	[10am,11am]	34.48	19.84
4	{Marquas de l'Argentera Avinguda del Marques Argentera}	[11am,12am]	33.52	21.15
5	{Paralà lel, Pl. Jean Genet}	[1am,2am]	32.64	25.42
6	{Paralà lel, Sant Oleguer, Pl. Jean Genet}	[1am,2am]	23.41	41.91
7	{Marquas de l'Argentera, Avinguda del Marques Argentera, Pl. Comercial}	[10pm,11pm]	22.99	37.16
8	{Marquas de l'Argentera Avinguda del Marques Argentera, Pl. Comercial}	[12pm,1am]	22.48	32.55

widespread that those in Barcelona. Some of these patterns show maximal criticality (100%), which may indicate a severe warning.

Domain experts can thus gather useful insights on the usage of the bycicle sharing system. On the one hand, they can identify daily time periods in which service disruptions may occur, and on the other hand, they can also identify the set of nearby stations which are involved in these disservices to end users.

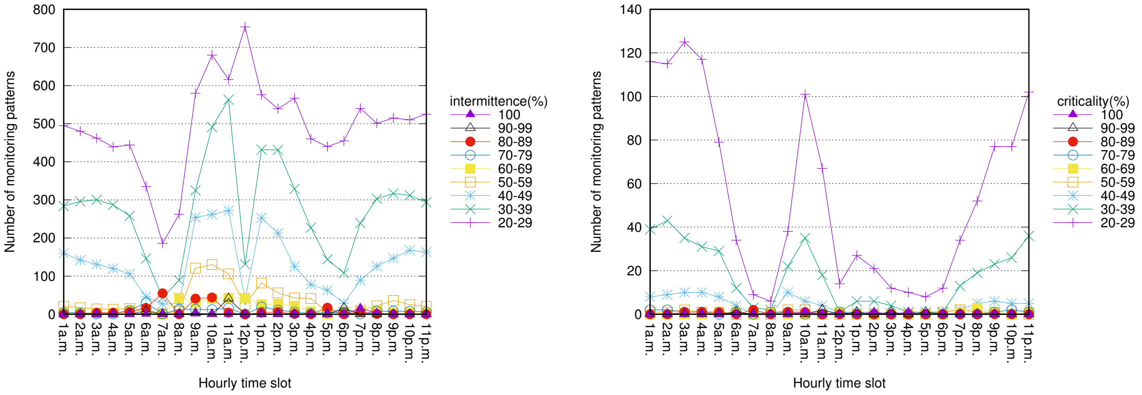
Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521



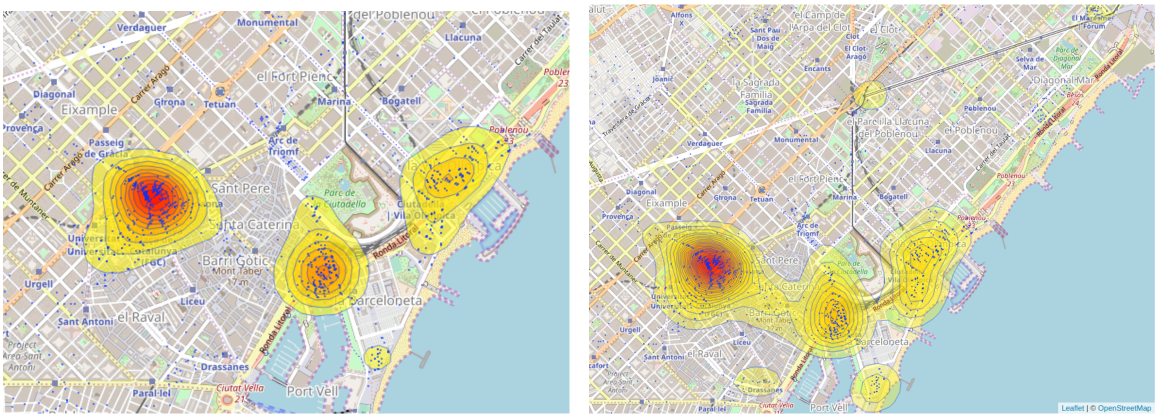
(a) Intermittence (b) Criticality

Figure 2. *Bicing* (Barcelona). Hourly distribution of the number of patterns and their corresponding levels of criticality/intermittence. *maxdist*=0.5 km. *time slot size*=1h. *full-th*=3.



(a) Intermittence (b) Criticality

Figure 3. New York. *Citi Bike* (New York). Hourly distribution of the number of patterns and their corresponding levels of criticality and intermittence. *maxdist*=0.5 km. *time slot size*=1h. *full-th*=3.



(a) Intermittence (b) Criticality

Figure 4. Heat maps representing intermittence and criticality values in Barcelona at the hourly time slot [11am-12am). *maxdist*=0.5 km, and *time slot size*=1h. *full-th*=3.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521

Geographical distribution of significant intermittent and critical patterns. Each occupancy monitoring pattern represents a group of geo-referenced stations. To support the management of the bicycle sharing system, maps can be used to highlight the city areas associated with patterns (i.e., groups stations) with high intermittence and criticality values.

For example, Figures 4(a) and 4(b) show two heat maps[3] of the areas of Barcelona identified by the patterns in hourly time slot [11am-12am). Patterns in this time slot represent significant intermittent and critical situations according to the results in Figure 2. In Figures 4(a) and 4(b) the color intensity of areas increases with the density of occurrence of patterns and their intermittence and criticality values, respectively. The higher the color intensity, the most severe the disservice to end users.

Figure 4(a) shows that intermittent situations are mainly localized in the city center in four distinct areas. The areas with the highest intensity is centered in *Placa Catalunya*, while the other two large areas are centered in *History Museum of Catalonia* and *La Vila Olimpica del Poblenou* and a small area is in *Pla de Miquel Tarradell*.

Instead, based on Figure 4(b), critical situations are more spread over the geographical areas. The larger area in Figure 4(b) cover all the three main areas in Figure 4(a). Moreover, three additional areas show up, two of them located on the top of the map (in the *Torre Glories* and *El Maresme Forum* areas) and one on the bottom (*Drassanes* area).

3.3. Analysis of the impact of the system parameters

We analyzed the impact of the *full-th*, *maxdist*, and *time slot size* parameters on (i) the cardinality of the mined patterns (i.e., the number of monitoring patterns per time slot), (ii) the distribution of the intermittence values of the mined patterns, and (iii) the distribution of the criticality values of the mined patterns.

In the experimental evaluation we varied one parameter at a time, and we set the standard configuration for the remaining parameters. The standard configuration was introduced in Section 3.2 as *maxdist* = 0.5 km, *full-th*=3, *time slot size*=1h.

For the sake of brevity, we will hereafter report the results achieved on the *Bicing* dataset (Barcelona). Similar results have been obtained from the other dataset *Citi Bike*.

Occupancy threshold (*full-th*). Figures 5(a)-5(b) show the impact of the *full-th* parameter on the mined patters. The two figures report the total number of mined patterns for each range of intermittence and criticality value when increasing *full-th*.

A station is in overloaded condition when less than *full-th* free docks are available. Therefore, the higher occupancy threshold value we set, the more patterns with high intermittence/criticality value could be extracted.

The results reported in Figures 5(a)-5(b) show this trend. The number of monitoring patterns for each intermittence and criticality range increases almost linearly when increasing the *full-th* value. This increase is higher for the intermittence index.

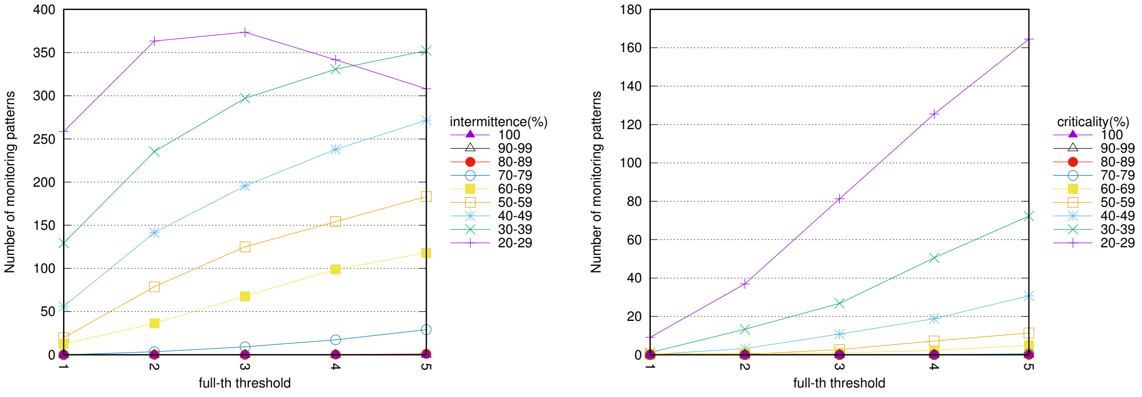
Maximum distance threshold (*maxdist*). Figures 6(a)-6(b) show the impact of the *maxdist* parameter on the number of mined patterns. The two figures report the total number of mined patterns for each range of intermittence and criticality value when increasing *maxdist*.

When the *maxdist* value is increased, the number of nearby stations also increases. Consequently, the number of mined patterns increases because larger patterns including more stations are also generated. Results show that when increasing *maxdist* the number of patterns increases almost exponentially for each intermittence range and almost linearly for each criticality range.

[3] The heat maps have been generated by using the service provided by Babicki et al. [24].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

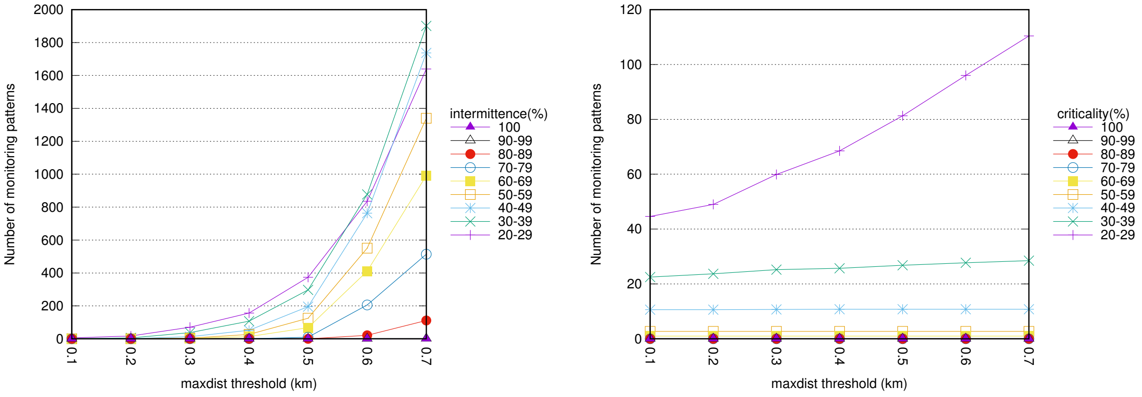
Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521



(a) Intermittence

(b) Criticality

Figure 5. Barcelona. Impact of the occupancy threshold on the characteristics of the mined patterns. *maxdist*=0.5 km. *time slot size*=1h.



(a) Intermittence

(b) Criticality

Figure 6. Barcelona. Impact of the maximum distance threshold on the characteristics of the mined patterns. *full-th*=3. *time slot size*=1h.

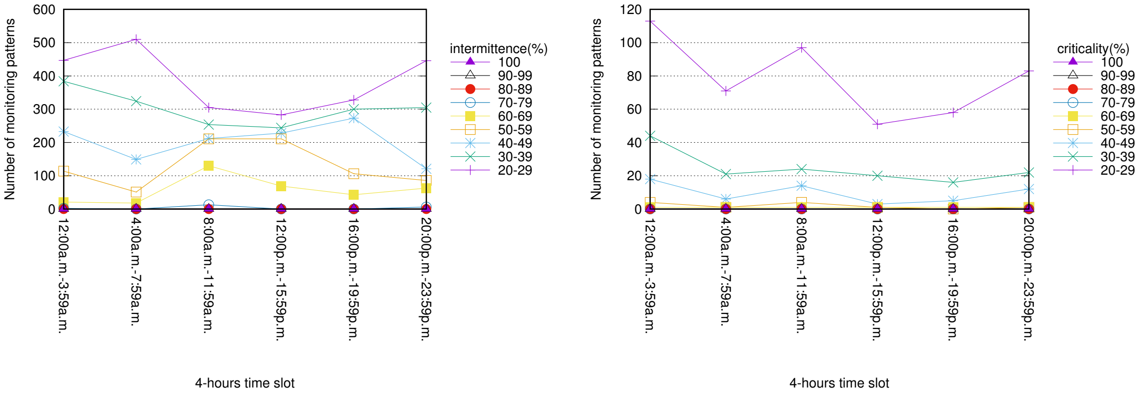
However, the number of patterns that are worth considering for manual inspection (i.e., those with high intermittence/criticality values) remains roughly stable even while enforcing *maxdist* values higher than 0.5 km. Setting *maxdist* values higher or equal to 0.6 km is less interesting in our context of analysis because the end users are willing to move to physically closer stations if the expected destination is fully occupied.

Time slot size. The distribution of the number of extracted patterns for each intermittence and criticality range when varying the time slot size were also analyzed. Experiments were performed for time slots ranging from 2 to 8 hours; as a representative example, Figure 7 reports the results achieved on the *Bicing* dataset with the 4-hours time slot.

Considering a courser time granularity to analyse collected data, as, for example, a larger time slot size, can provide a high-level view of the station overload conditions in the bicycle sharing system. This view can be useful for end users but expecially for system managers to help them identify the time frames when usage conditions are critical. For instance, results in Figure 7(a) point out that the number of monitoring patterns with high intermittence value (between 50%-59%) is significantly higher between 8.00am-12:00pm.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at Appl. Sci. 2018, 8, 2521; doi:10.3390/app8122521



(a) Intermittence

(b) Criticality

Figure 7. *Bicing* (Barcelona). Distribution of the number of patterns and their corresponding levels of criticality/intermittence with a time slot granularity of 4 hours. *maxdist*=0.5 km. *time slot size*=4h.

Domain-experts can then focus on each selected time frame to locally analyze collected data with a finer time granularity (i.e.,a time slot with lower size). This latter analysis can provide more detailed information on dock overload conditions on each selected time frame.

In some cases, using time slots with a larger size could smooth local intermittence and criticality peaks of potential interest. For instance, few patterns with intermittence in the range 70%-79% are mined with a 4-hour time slot (see Figure 7(a)). Instead, when considering 1-hour time slots, around 50 patterns with intermittence between 70%-79% are generated in the 10am, 11am, 12pm time slots (see Figure 2(a)).

3.4. Execution time

The execution time for the OMP-Miner algorithm includes the time for (critical and normal) o-itemset extraction and the generation of the OMPs from them. The o-itemsets extraction is the most computationally expensive step. With the default parameter setting, the o-itemsets' extraction time is approximately 454s for *Bicing* (Barcelona) and 825s for *Citi Bike* (New York), while the time for OMP generation is a few milliseconds in both cases.

We also analysed how the system parameters impact on the execution time. Specifically we focused our analysis on the maximum distance threshold *maxdist* which can impact significantly on the number of mined patterns, and thus on the execution time. Experiments were run by varying the *maxdist* value while the standard configuration was adopted for the other parameters. The execution time, similarly to the number of mined patterns, increases almost non-linearly with respect to the maximum distance threshold value. The time ranges from 3 minutes when *maxdist*=0.1 km up to 42 minutes when *maxdist*=0.6 km. The execution time increases to more than one hour when values of *maxdist* greater than 0.6 km are used, i.e., when *maxdist* is set to values that are considered not interesting in our application domain. Most of the execution time is for o-itemset generation while even in the worse case the OMP generation requires a few seconds.

4. Discussion

This paper presents an itemset-based approach to characterizing the occupancy levels of the stations of bicycle sharing systems. It discovers patterns representing (i) groups of nearby stations whose slots are almost all occupied at most points of time, and (ii) groups of nearby stations among which *at least one of them* (but not all of them) has a high level of occupancy at most points of time (possibly in an alternate fashion).

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

The results achieved using real mobility data highlighted potentially critical situations that could lead to service disruption for end users. The extracted patterns represent situations of imbalance in station occupancy levels contextualized in specific time periods and city areas (e.g., in the central district of Barcelona from 9am to 11am). They represent both critical and intermittent dock overload conditions, which respectively lead to partial and complete service disruption. The type and degree of severity of the imbalance situation are described by the main pattern quality measures (i.e., the criticality and intermittence indices), which have been exploited to drive the result exploration. Therefore, the discovered patterns allow domain experts to monitor critical situations by providing useful insights into system usage.

As future work, we plan to integrate other data sources to enrich the quality of the generated model. We consider, amongst other, variables such as the presence of environmental pollution, the vehicular traffic, and the presence of cycling lines as indicators of favorable/unfavorable conditions for bicycle sharing system usage.

In parallel, we will investigate the portability of the proposed methodology to different mobility services offered in the urban context. For example, we plan to apply the proposed approach to charging stations of electric cars and to indoor car parks.

Author Contributions: Conceptualization, L. Cagliero, Silvia Chiusano and Paolo Garza; Funding acquisition, Silvia Chiusano; Investigation, Paolo Garza and Giuseppe Ricupero; Methodology, Luca Cagliero, Tania Cerquitelli, Silvia Chiusano and Paolo Garza; Software, Giuseppe Ricupero; Supervision, Silvia Chiusano and Elena Baralis; Writing – original draft, Luca Cagliero and Silvia Chiusano; Writing – review & editing, Luca Cagliero, Tania Cerquitelli, Silvia Chiusano and Paolo Garza.

Funding: The research leading to these results was partially funded by the Italian Ministry of Research (MIUR) under the Smart Cities and Communities Grant Agreement n. SCN_00325 (Project s[m2]art).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhu, X.; Fan, Y.; Zhang, F.; Ye, X.; Chen, C.; Yue, H. Multiple-Factor Based Sparse Urban Travel Time Prediction. *Applied Sciences* **2018**, *8*. doi:10.3390/app8020279.
2. Cagliero, L.; Cerquitelli, T.; Chiusano, S.; Garza, P.; Ricupero, G. Discovering air quality patterns in urban environments. Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp Adjunct 2016, Heidelberg, Germany, September 12-16, 2016, 2016, pp. 25–28. doi:10.1145/2968219.2971458.
3. Zheng, Y. Methodologies for Cross-Domain Data Fusion: An Overview. *Big Data, IEEE Transactions on* **2015**, *1*, 16–34.
4. Zheng, Y.; Capra, L.; Wolfson, O.; Yang, H. Urban Computing: Concepts, Methodologies, and Applications. *ACM Trans. Intell. Syst. Technol.* **2014**, *5*, 38:1–38:55.
5. Wang, S.; Zhang, J.; Liu, L.; yu Duan, Z. Bike-Sharing-A new public transportation mode: State of the practice and prospects. Emergency Management and Management Sciences (ICEMMS), 2010 IEEE International Conference on, 2010, pp. 222–225.
6. Etienne, C.; Latifa, O. Model-Based Count Series Clustering for Bike Sharing System Usage Mining: A Case Study with the Velib System of Paris. *ACM Trans. Intell. Syst. Technol.* **2014**, *5*, 39:1–39:21.
7. Sarkar, A.; Lathia, N.; Mascolo, C. Comparing cities' cycling patterns using online shared bicycle maps. *Transportation* **2015**, *42*, 541–559.
8. Ciancia, V.; Latella, D.; Massink, M.; Pakauskas, R. Exploring Spatio-temporal Properties of Bike-Sharing Systems. Self-Adaptive and Self-Organizing Systems Workshops (SASOW), 2015 IEEE International Conference on, 2015, pp. 74–79.
9. Kaltenbrunner, A.; Meza, R.; Grivolla, J.; Codina, J.; Banchs, R. Urban Cycles and Mobility Patterns: Exploring and Predicting Trends in a Bicycle-based Public Transport System. *Pervasive Mob. Comput.* **2010**, *6*, 455–466.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2018 doi:10.20944/preprints201810.0221.v1

Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2521; doi:10.3390/app8122521

10. Froehlich, J.; Neumann, J.; Oliver, N. Measuring the Pulse of the City through Shared Bicycle Programs. Proceedings of the International Workshop on Urban, Community, and Social Applications of Networked Sensing Systems (UrbanSense08), 2008.
11. Girardin, F.; Calabrese, F.; Fiore, F.D.; Ratti, C.; Blat, J. Digital Footprinting: Uncovering Tourists with User-Generated Content. *IEEE Pervasive Computing* **2008**, *7*, 36–43.
12. Hasan, S.; Zhan, X.; Ukkusuri, S.V. Understanding Urban Human Activity and Mobility Patterns Using Large-scale Location-based Data from Online Social Media. Proceedings of the 2Nd ACM SIGKDD International Workshop on Urban Computing; ACM: New York, NY, USA, 2013; UrbComp '13, pp. 6:1–6:8.
13. ter Hofte, H.; Jensen, K.L.; Nurmi, P.; Froehlich, J. Mobile Living Labs 09: Methods and Tools for Evaluation in the Wild: Http://Mll09.Novay.Nl. Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services; ACM: New York, NY, USA, 2009; MobileHCI '09, pp. 107:1–107:2. doi:10.1145/1613858.1613981.
14. Wang, I.L.; Wang, C.W. Analyzing Bike Repositioning Strategies Based on Simulations for Public Bike Sharing Systems: Simulating Bike Repositioning Strategies for Bike Sharing Systems. Advanced Applied Informatics (IIAIAAI), 2013 IIAI International Conference on, 2013, pp. 306–311.
15. Raviv, T.; Tzur, M.; Forma, I.A. Static repositioning in a bike-sharing system: models and solution approaches. *EURO Journal on Transportation and Logistics* **2013**, *2*, 187–229.
16. Vogel, P.; Greiser, T.; Mattfeld, D.C. Understanding Bike-Sharing Systems using Data Mining: Exploring Activity Patterns. *Procedia - Social and Behavioral Sciences* **2011**, *20*, 514 – 523.
17. Schuijbroek, J.; Hampshire, R.; van Hoeve, W.J. Inventory rebalancing and vehicle routing in bike sharing systems. Technical report, Politecnico di Torino, 2009. Working paper. Available at http://repository.cmu.edu/.
18. Singla, A.; Santoni, M.; Bartók, G.; Mukerji, P.; Meenen, M.; Krause, A. Incentivizing Users for Balancing Bike Sharing Systems. Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. AAAI Press, 2015, AAAI'15, pp. 723–729.
19. Lozano, A.; De Paz, J.F.; Villarrubia Gonzalez, G.; Iglesia, D.H.D.L.; Bajo, J. Multi-Agent System for Demand Prediction and Trip Visualization in Bike Sharing Systems. *Applied Sciences* **2018**, *8*. doi:10.3390/app8010067.
20. Pang-Ning, T.; Michael, S.; Vipin, K. Intoduction to Data Mining, 2005.
21. Agrawal, R.; Imielinski, T.; Swami. Mining association rules between sets of items in large databases. ACM SIGMOD 1993, 1993, pp. 207–216.
22. Han, J.; Pei, J.; Yin, Y. Mining Frequent Patterns without Candidate Generation. *In SIGMOD'00, Dallas, TX* **2000**.
23. Agrawal, R.; Srikant, R. Fast Algorithms for Mining Association Rules in Large Databases. Proceedings of the 20th VLDB conference, 1994, pp. 487–499.
24. Babicki, S.; Arndt, D.; Marcu, A.; Liang, Y.; Grant, J.R.; Maciejewski, A.; Wishart, D.S. Heatmapper: web-enabled heat mapping for all. *Nucleic Acids Research* **2016**, *44*, W147–W153. doi:10.1093/nar/gkw419.