

Article

Towards a Persuasive Recommender for Bike Sharing Systems: A Defeasible Argumentation Approach

Carlos Diez ^{1,*}, Javier Palanca ¹, Victor Sanchez-Anguix ^{2,3}, Stella Heras ¹,
Adriana Giret ¹ and Vicente Julián ¹

¹ Departamento de Sistemas Informáticos y Computación, Universitat Politècnica de València, Camí de Vera s/n, 46022 València, Spain; jpalanca@dsic.upv.es (J.P.); sheras@dsic.upv.es (S.H.); agiret@dsic.upv.es (A.G.); vinglada@dsic.upv.es (V.J.)

² Unidad de Tecnologías de la Comunicación e Información, Florida Universitaria, Carrer del Rei en Jaume I, 2, 46470 Catarroja, Spain; vsanchez@florida-uni.es or victor.sanchez.anguix@ui1.es

³ Facultad de Ciencia y Tecnología, Universidad Isabel I, Calle de Fernan Gonzalez, 76, 09003 Burgos, Spain

* Correspondence: cardial@dsic.upv.es; Tel.: +34-963877000

Received: 21 December 2018; Accepted: 14 February 2019; Published: 19 February 2019



Abstract: This work proposes a persuasion model based on argumentation theory and users' characteristics for improving the use of resources in bike sharing systems, fostering the use of the bicycles and thus contributing to greater energy sustainability by reducing the use of carbon-based fuels. More specifically, it aims to achieve a balanced network of pick-up and drop-off stations in urban areas with the help of the users, thus reducing the dedicated management trucks that redistribute bikes among stations. The proposal aims to persuade users to choose different routes from the shortest route between a start and an end location. This persuasion is carried out when it is not possible to park the bike in the desired station due to the lack of parking slots, or when the user is highly influenceable. Differently to other works, instead of employing a single criteria to recommend alternative stations, the proposed system can incorporate a variety of criteria. This result is achieved by providing a defeasible logic-based persuasion engine that is capable of aggregating the results from multiple recommendation rules. The proposed framework is showcased with an example scenario of a bike sharing system.

Keywords: cyber-physical systems; smart systems; artificial intelligence; bike sharing system

1. Introduction

Nowadays, urban mobility has become a crucial problem as population grows in cities, and so does pollution, traffic congestion, and the consumption of energy in the form of carbon-based fuels. In order to tackle this problem, many cities have implemented alternative transportation methods such as bike or car sharing systems. Despite the increasing adoption of these systems, there are still open problems that need to be solved to further improve the adoption of these services. The digital transformation of cities into smart cities has provided us with a range of different information services that may be employed to implement a large mirage of applications devoted to improve urban mobility. This has opened the door for the employment of new management strategies and policies based on information and cyber-physical systems.

Our work lies between two of the main societal challenges identified by the European Commission (EC Societal Challenges: <http://ec.europa.eu/programmes/horizon2020/en/h2020-section/societal-challenges>): 'secure, clean and efficient energy' and 'smart, green and integrated transport', favouring savings in energy consumption from carbon-based fuels and costs, decreasing the levels of pollution, and promoting a transport system that is resource-efficient, and climate and environmentally friendly.

In this paper we focus on one of the open problems associated with the bicycle sharing service; balancing the distribution of bikes in pick-up stations. The problem of unbalanced resources in these systems has, actually, a double harmful effect. First of all, the lack or excess of bikes at stations may deteriorate the service satisfaction of users, as they may find that their desired trips are not plausible or they need to be rescheduled on the fly. Dissatisfaction levels may impact the economic viability for system operators as more users may decide to leave the system. This, in turn, prevents the social, economical, and energy benefits arising from the use of bike sharing systems. Secondly, traditionally, system operators have employed fleets of trucks to balance and redistribute bikes across the different bike stations. The extensive use of these trucks also impacts the economic viability of these systems, and, therefore it may again preclude benefiting from the social, economical, and energy related advantages of adopting bike sharing systems.

With the purpose of balancing bikes across bike stations, we propose a flexible argumentation-based persuasive approach whose final goal is that of slightly deviating users from their initial trip to both cater for the balancing needs of the network and the preferences of the user. The persuasive system recommends alternative stations to the original one when it predicts that the occupation of the destination station will be complete, or when the system detects that it is better for the general balance of the system to avoid trips to the station chosen by the user. This new station recommendation is associated with a new trip route, and the system provides arguments to the user to select those routes instead of the original one attending to a variety of criteria. This way, the users can be used as balancing agents that help to provide a high quality service. The scope of this persuasion system is for those bike sharing systems where it is not allowed to park the bicycle outside one of the stations (whether they are full or not). Current systems usually suggest the closest station to the desired destination, but the system presented in this work aims to improve user satisfaction by anticipating these situations at the beginning of the journey and taking advantage of the situation to motivate the user with new routes. In addition to this, the general objective of improving the balance of the system and reducing the need for management trucks (which generate CO₂) is met. The proposed system benefits from the transformation of cities into smart cities, becoming a cyber-physical system together with bike sharing systems and human interaction. This proposal constitutes our first step towards solving the problem of persuading users to contribute to the general balance of the bike network.

The paper is organized as follows. Section 2 describes the problem we are targeting and the related works from the specialized literature that solve similar problems. Section 3 presents the general architecture of the system. Section 4 introduces how the occupation forecast is done for bike sharing stations. Section 5 explains the proposed approach to improve the efficiency and quality of service of a bicycle sharing system. Section 6 showcases the proposal with a sample scenario in which the rules are tested. Section 7 presents experimentation and results to show how the persuasive recommender impacts on a bike sharing system balance. Section 8 summarizes the conclusions and future works.

2. Related Work

In the last few years, bike sharing systems have received increasing scholarly attention due to their widespread use in cities. These systems allow users to travel between two points in the city by using a healthy, clean and cost effective transportation method. Despite this, there are several problems that are detrimental to the usability and customer satisfaction of these systems. Many of these problems are a consequence of limited resources (e.g., bikes and bike slots) and people's mobility in cities. For instance, users may not find available bikes when attempting to pick one from a station, or they may not find available parking spots when reaching their destinations. This creates dissatisfaction in the customers of these systems, leading to lack of engagement and, thus, higher use of carbon-based fuels. In general, the travel patterns of users at different times of the day may make the system unbalanced in the number of available bikes/parking slots throughout stations in the system.

As the reader may have guessed, some researchers have attempted to alleviate problems in bike sharing systems by means of information and communication technologies (ICT) and artificial

intelligence (AI). For instance, one of the main areas of contribution of ICT and AI to bike sharing systems is that of predicting the availability of bikes or parking slots in the close future [1–4]. Predicting the availability of bikes or parking slots allows system managers to devise plans of action with the aim of redistributing resources where needed. Generally, these plans can be created by means of automated methods based on optimization algorithms like the ones presented in [5–12]. These optimization methods aim to minimize the energy footprint, fuel, and cost involved in moving bikes between stations to balance the resource distribution. The work carried out for this article is aligned to the problem of balancing resources across bike stations in a city.

As mentioned, reallocating bikes across bike stations is a crucial task for avoiding customer dissatisfaction. This means that bikes should be available at stations where customers desire to start their trips from, and free parking slots should be present at destinations. In general, there are two different approaches to face the aforementioned problem; static and dynamic. On the one hand, static approaches act upon the distribution of bikes at stations considering the system as closed (i.e., ignoring user action) and usually relocate bikes on fixed intervals. The work presented in [8] provides a static and reactive reallocation mechanism for bike sharing system. The algorithms presented take into consideration that each station should have a lower and upper bound on the number of bikes available attending to user behaviour in the system, and take the current state of the network of stations for devising a cost-minimal algorithm that results in bike allocation that are in line with the lower and upper bound constraints of each station. Similarly, the work of Ho et al. [6] provides a static and reactive reallocation strategy based on Tabu search that employs the current state of the stations, operational constraints, and the cost induced in the transportation of the bikes. The authors of [11] provide a static reallocation solution based on a predictive module that uses historical data to determine optimal service levels at stations, and routing algorithms that employ the previous predictive models and heuristic search algorithms to decide on the routes taken by the logistic operator to redistribute resources. From the scholarly point of view, the use of dynamic approaches has been studied with more interest due to the fact that it considers users' actions. The work of Caggiani et al. [5] falls within the latter lines. The authors employ artificial neural networks to predict the bike and parking demand at different stations and variable time gaps. Then, the system employs fuzzy logic controls to determine whether or not the resources need to be redistributed among available stations. Another proactive approach is proposed in [7]. The authors provide mixed integer programming (MIP) formulation of the reallocation problem based on the expected demand of bikes and parking slots at fixed time frames for every station, and constraints in the form of the minimum service level for the system. Due to the large computational complexity of finding an exact solution by means of MIP solvers, the problem is first simplified by means of a neighborhood search method and then the resulting problem is solved by the exact MIP solver. Another relevant approach is that of Reiss et al. [10]. Differently to other approaches, the bike sharing system in Munich is not based on fixed bike stations. Instead, users can leave bikes at any point within operational areas. The authors study use patterns of costumers in the system to determine which variables affect demand. Then, the authors propose a relocation system inspired by ideas employed in car sharing systems. The system is formed by both an offline and an online optimization module that take into consideration the expected demand based on historical data and the current state of the system. It provides recommendations on how the bikes should be redistributed in operational areas. Lastly, a dynamic balancing solution is proposed in [12], where Markov processes and heuristic approaches based on the decomposition of the balancing problem into single vehicle routing problems are applied to real data from several large-sized American cities. All of these approaches rely on the bike sharing system's operator and optimization methods to balance the system.

Differently to the previous approaches, which take a centralized approach where the service provider balances the system by using their own logistic resources, guided by optimization methods, our proposal aims to help the balancing task by persuading users in the choice of a start and end location for their journey. Dealing with the balancing problem by involving the users in the process is

not novel, although it has not been as widely studied as the centralized approach. For instance, the authors in [13] propose the use of monetary incentives to slightly change users' trips with the goal of maximizing user satisfaction in the system and reallocating bikes across stations. The model proposed by the authors employs online learning and regret minimization to dynamically calculate incentives for users. Aeschbach et al. [14] study the problem from a rather different perspective. The authors focus on studying how users would react to different control strategies based on their degree of cooperation. Their simulations based on historical data show that even in the case of low cooperation degrees from customers, bike sharing systems can attain reasonable service levels by involving users into the balancing tasks and employing simple communication mechanisms with users. In [15], a bidirectional reward system is proposed based on three possible scenarios. This means that users whose trips may unbalance the system are penalized, while users whose trips help to balance the system are rewarded. The optimal strategies for those scenarios are calculated based on game theory.

Recently, persuasive systems for promoting sustainable urban mobility have been proposed in the literature [16]. Our proposal falls into the aforementioned category of systems, where we intend to persuade users into changing their trips in order to balance the system and reduce the use of the truck fleet. The persuasion is supported by the recommendation of a specific route between the start and end location according to the characteristics of the user. With that purpose, we propose a system that is capable of recommending different alternative routes to that of the shortest path between two points. Rather than focusing on monetary incentives, we attempt to appeal to users by considering their characteristics and travel preferences. Recently, scholars have discovered that distance is not the only factor that humans take into consideration when preferring a route over another [17–19]. In fact, some studies have suggested that some individuals may even deviate up to approximately a 30% from the shortest path in exchange for a pleasant journey [20]. The factors that may influence the users' decisions are varied. For instance, Quercia et al. [17] propose a set of models that are able to provide more beautiful, quieter, or happier routes while deviating little from the shortest distance between the start and end location. For that, they provide two models built from crowd sourced data and from Flickr tags. Users suggested that proposed routes were more apt for cheering up (i.e., happy route), a stressful day (i.e., quiet route), and for sightseeing (i.e., beautiful route). Later on, Galbrun et al. [18] proposed an algorithm that calculates safer paths as an alternative to the shortest path. For that, the authors employ open crime data from the city of Chicago to map those city segments with the highest crime rates. Their proposed algorithm computes a set of Pareto optimal paths attending to the trade-off between distance and safety. In [19], the authors propose a path recommendation system that takes into consideration the mood of the user, represented by a RGB color, and proposes a path whose distribution of historical buildings, green spaces, and open spaces respectively resembles the distribution of the red, green, and blue channel of the color representing the mood of the user.

Another attractive method for convincing users to change their behaviour is through persuasive games [21]. For instance, in [22] authors present gamification mechanisms to promote voluntary behavioural changes towards sustainable mobility solutions. A similar approach is followed by the app presented in [23], which uses challenges with rewards, feedback, and message functions to promote cycling. However, the successful use of persuasive games focuses on two key points; the user is interested or can invest time into playing the game, and the user receives some reward for participating (incentives, social interaction, etc.). Our approach is different in that the user does not have to spend more time interacting with the system than that of specifying his/her preferences and receiving the suggested recommendations. In addition, an explicit reward is not provided, but an attempt is made to explain to the user the reason behind the system's decisions and why the recommended route is more appropriate for him/her.

The work presented in this article aims to persuade users to choose different routes from the shortest route between a start and an end location. The rationale behind the recommendations provided by the persuasion system may reside in the balance of the bike sharing system, or the preferences of the user. Differently to [17–19], instead of employing a single criteria to recommend routes, the proposed

system can incorporate a variety of criteria such as the ones mentioned above, the balancing of the system, and other criteria. This result is achieved by providing a defeasible logic-based persuasion engine that is capable of aggregating the results from multiple recommendation rules.

3. Bike Sharing Systems as Cyber-Physical Systems

With the advance of technology (connectivity, devices that support human interaction, etc.) cities have become smart cities. This new generation of cities may provide multiple resources in diverse forms: information about the state of the city, new ways to perform physical actions in the environment, city-citizen interaction through technology, and so forth. This digital transformation is not alien to bike sharing systems, which have rapidly evolved from traditional and managed bike rent services to the almost fully automated systems that we currently have in most of the cities.

Traditionally, the following elements can be considered as core parts in bike sharing systems: bicycles, stations and users. The intertwined relationship between the three, as well as the digitization of cities, including bike sharing systems, makes it possible to consider the new generation of bike sharing systems as cyber-physical systems. These systems integrate computing, engineered elements and human factors, and there are reciprocal feedback loops between the three of them, changing the behavior of all of the actors in the system [24]. On the one hand, bikes and stations have integrated and engineered technologies that record information from physical events (e.g., state of the parking stations, location of the bikes, etc). This information flows from the physical devices to the information systems that are employed by system operators. This information helps to make decisions about the redistribution and use of resources by the system operator. In addition to this, many system operators also provide free information services to users about the state of the network, thus influencing their behavior. As mentioned, decisions made by system administrators on the distribution of physical resources (i.e., bikes) across stations directly impacts customer satisfaction and system adoption. In many occasions, given the size of the system, many of these decisions are supported by software and information technology. Obviously, users also influence the physical system, as they move and distribute bikes across the different stations of the city. Indirectly, this also affects the decisions made by system operators. As the reader may observe, there is a close relationship between the physical resources of the system, the computing involved, and the human actuators, making this a clear cyber-physical system.

It is expected that, with the inclusion of the large volumes of data generated from this cyber-physical system and external data sources, the management of these kinds of systems may dramatically improve. For instance, the system can now consider data such as weather (e.g., temperature, precipitation, wind speed), traffic, pollution, and city infrastructure data to better improve the decisions made at the management of the resources in these systems.

In turn, these decisions can help to reduce monetary costs and energy consumption in the form of carbon-based fuels. With these new resources, decision and management processes can be improved in order to make better decisions that can help to reduce the system cost in energy and monetary costs. A representation of the bike sharing system as a cyber-physical system can be found in Figure 1.

Our proposal for bike sharing cyber-physical systems improvement is based on a multi-agent system which makes use of available data to improve the decision making process, thus enhancing resource management. The specific problem that we aim to solve is that of persuading users to pick specific stations as their destinations when the original destination station to which a user wishes to go is predicted to be full, or when we are not interested in that station being full to better balance the overall system. In these cases, the system will suggest an alternative and nearby destination station using persuasion techniques so that the user accepts one of the proposed stations. One of the most frequent arguments that we aim to use in the persuasive process is the possibility of choosing a type of route to the new station that fits the profile or tastes of the user. When a new destination station is proposed, the multi-agent system aims to persuade the users to make small deviations on their routes, thereby modifying the stations where resources are obtained or released. By modifying the users

behaviour this way, the system is guiding users so that the decision takes into account both the users' and service provider's needs. This may improve the bike sharing system balance since these small deviations can be beneficial during pre/after rush hour balancing, and acting as real time balance for unplanned demands.

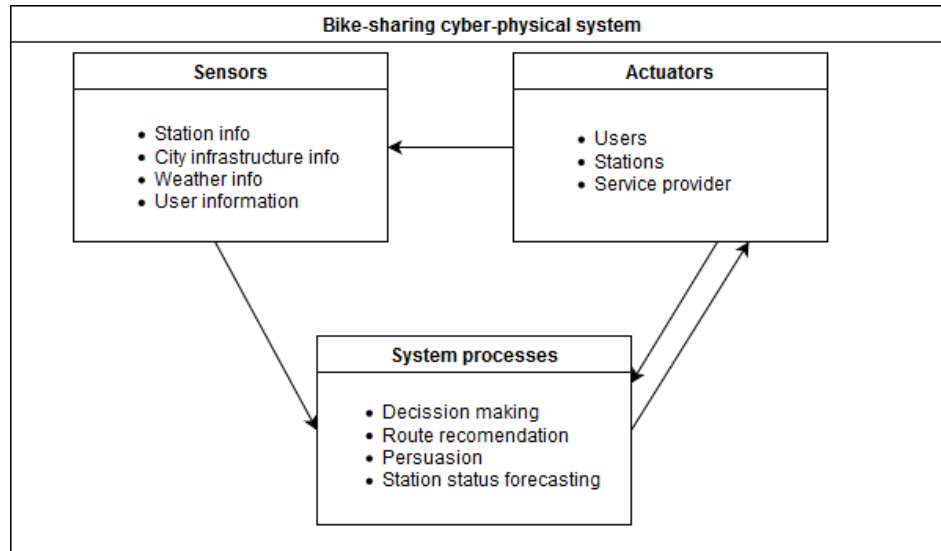


Figure 1. Bike sharing systems as cyber-physical systems.

The multi-agent system presented, encapsulated as the efficient bike trip module, is responsible for calculating the most efficient system balancing by providing alternative stations and routes to the station chosen by the user, given the current system status. This module is composed of an intelligent transportation ontology and three different sub-modules:

- Transportation network analysis module: obtains events and data from the transportation network (stations location, traffic status, etc.) and processes it in order to generate usable information to be used in subsequent phases.
- Bikes an parking availability prediction module: this module is responsible of forecasting the system demand. It is briefly presented in next section.
- Argumentation module: generates arguments aiming to persuade users to accept the proposed routes. This module is the main contribution of this work and it is further discussed in this article.
- Intelligent transportation ontology: The efficient bike trip module merges data from different sources (events from the smart city, queries from the user, etc.). For this reason it is necessary to share a common ontology that allows the module to work with the data sources in an appropriate manner. To do this, we use the intelligent transportation ontology.

As an overview, the system operates as follows: first the user determines his preferred station to start the trip and his destination, making the system aware of the user's initial intention. After this, when the system wants to prevent the user from going to the selected destination station, it chooses an alternative station close enough, and the system uses the information provided by the transportation network analysis module and the bikes an parking availability prediction module to determine possible routes that may be interesting to the user and, ultimately, help to balance the bike sharing system. Lastly, the argumentation module computes arguments in favor of the generated routes (taking into account the user preferences), and the system presents them together to the user with the aim of changing his/her behaviour. Note that the recommender system should only act in the event that it is predicted that the destination station will not be available, or allowing the user to park the bike in the station will significantly impair the system for the purpose of its overall balancing. This general view of the proposed architecture is shown in Figure 2 and it is presented and further discussed in [25].

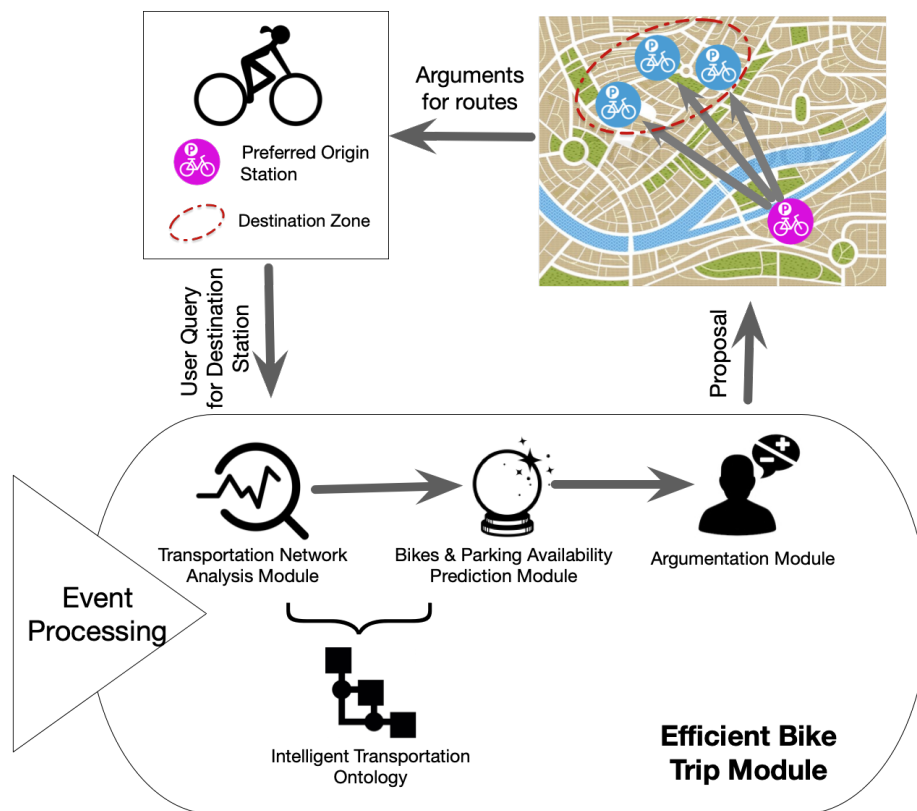


Figure 2. The efficient bike trip module.

4. Station Status Forecasting

As noted in Section 3, with the inclusion of the data that smart cities can provide to the system, the information that supports decision making may be improved. A particular case of this scenario is the use of data to forecast bike stations' state.

If we know the future state of stations, and we are able to predict demand, we can provide an accurate prevision of the use of the system. Therefore, avoiding unwanted situations that may deteriorate user satisfaction such as the lack of bikes or parking slots at stations. As we mentioned, avoiding these situations can lead to an improvement in the system's efficiency, increasing customer satisfaction, a wider adoption of bike sharing systems, and reductions in energy coming from the use of carbon-based fuels.

In order to obtain a high quality prediction, we have designed a station status forecasting module. For this, we have leveraged on the predictive power of machine learning models and the use of real historic data from a bike sharing system. The module employs artificial neural networks, which have been extensively employed in other prediction tasks [26–28]. Particularly, in our system, the predictions of this module are employed to feed the persuasive system, which will recommend trips to users based on the predicted state for the network of stations and the users' preferences.

The station state forecasting design and the experimentation process are further described and discussed in [4]. A brief description of this module is included in this article to better understand the tools on which the persuasion module is supported to make recommendations and suggestions to users.

We designed an experiment to conclude if forecasting was possible within an acceptable accuracy. The experiment was carried out using the case study of Valenbisi (Valencia's bike sharing system) and open data from Valencia, Spain. The first part of the experiment was an exploratory data analysis. With the analysis we determined that the most influential information for the system usage prediction is based on the following characteristics:

- Date: provides information for stationary events as the season or time of the year, fixed events in the calendar (e.g., holidays), and so forth.
- Weekday: the day of the week is one of the features that influences the system state the most. This is due to the fact that users typically behave similarly for the same day of the week and differently to other days of the week. For example, we observed differences in bike and station usage between week and weekend days.
- Temperature: as in other aspects of human life, temperature affects user behaviour. Uncomfortable temperatures (both very low and high) reduce the system usage.
- Rain and wind speed: rain and wind may be uncomfortable and even dangerous when using a bike. These weather conditions also affect the user behavior and the system usage.
- Current station state: a look at the current station state offers helpful information in order to make a prediction for the close future.

The second part of the experiment consisted of hyper-parameter optimization for the neural networks employed to predict the station state at each station. Then, we carried out a set of experiments where we compared the artificial neural network models, support vector regression models and two benchmark models (average number of available bikes throughout history and naive prediction with the current state of the station as a prediction). While both of the machine learning models offered better performance than the benchmark models, experiment results suggested that artificial neural networks outperformed all of them, including support vector regression models. The artificial neural network model improved the predictions of the benchmark that predicted the current status by 16.51%, the benchmark that predicted the average bike availability by 63%, and the SVR model by 10.05%, as can be observed in Table 1. An extended experiment description and results analysis can be consulted in [4].

Table 1. Relative improvement of model (rows) versus benchmark (columns).

VS.	Current	Average	SVR
ANN	16.51%	63.05%	10.05
SVR	6.09%	59.47%	N/A

5. A Framework for Computational Persuasion

The introduction of this work envisions an application where users express their goals (i.e., travel from a start point to an end point in the city) and the application suggests a set of multi-modal routes (i.e., including biking and walking) based on the balancing goals of the system, the probability of carrying out the shortest path route, and the preferences of the user. Since some of these recommendations may not result in the shortest path between the start and end point, a persuasion stage that helps to convince the user to choose one of our recommendations is needed, therefore indirectly persuading the user to balance the system. This persuasion stage is done by means of our argumentation framework presented in this section.

Our persuasion system was based on a defeasible logic programming (DeLP, see [29]) argumentation framework. It used several sources of information to generate persuasive arguments. These knowledge sources were fed from the information collected from the users and from the domain-based information about bike routes in the city. A defeasible logic program can encode information about the facts of the system (e.g., metadata and profiles data) and a set of rules to derive knowledge from these facts.

Definition 1 (DeLP). *In our system, a defeasible logic program $P = (\Pi, \Delta)$ included strict inference rules (Π) to represent facts (i.e., rules without body), and defeasible inference rules (Δ) of the form $P \leftarrow Q_1, \dots, Q_k$ that provided a mechanism to derive arguments to justify P .*

Let us assume that $user_type(john, ecologist)$ represents the fact that a user named 'John' prefers green routes. Thus, this fact may suggest that John would be more easily persuaded by routes that go through green zones and minimize the pollution indexes, avoiding areas whose pollution exceeds a threshold according to open data published by the local government. Contradictions cannot be inferred (e.g., $user_type(john, ecologist)$ and $\sim user_type(john, ecologist)$ cannot be inferred).

Table 2 shows a compendium of the proposed rules. Rules were divided into groups to better understand their purpose. The first group of rules made a content-based filtering based on the features of routes being recommended (C1.1.x) and based on the user profile preferences and constraints (C1.2.x). The next group computed a collaborative filtering by analyzing previous votes of other users and the similarity between those users and the user being recommended. This similarity function may be taken from classical collaborative filtering algorithms. The next group used a knowledge based approach to select routes that were similar to other routes that were previously voted positively by the user. This allowed the recommender to select routes that are most likely to suit the user's preferences. Finally, in this proposal, we have included a system rule that took care of the system balancing. Since we are interested in keeping a balance between all the stations in the system, we have generated a rule that penalized (and therefore allowed us to generate counter-arguments to attack other arguments) the selection of a route that would finish in an unbalanced system (where we had too many stations with no free spots and stations without bikes). The goal of the system was to keep the system balanced, to allow users to always find a free spot or an available bike wherever they go. It should be highlighted that the DeLP framework is flexible and it can incorporate other rules and facts that represent both the preferences/concerns of the user (i.e., safety, mood, contextual information) and the system such as the ones described in the literature review.

Table 2. Defeasible rules.

CONTENT BASED RULES
$C1: recommend(U, R) \leftarrow generally_appropriate(R) \wedge user_appropriate(U, R)$
$C1.1: generally_appropriate(R) \leftarrow available_bike_lane(R) \wedge no_transfer_route(R) \wedge fast_route(R) \wedge distance_in_range(R)$
$C1.1.1: fast_route(R) \leftarrow estimated_route_time(R) < 10$
$C1.1.2: distance_in_range(R) \leftarrow distance_in_close_range(R) \wedge distance_in_mid_range(R) \wedge distance_in_far_range(R)$
$C1.1.2.1: distance_in_close_range(R) \leftarrow distance_to_target(R) < 100$
$C1.1.2.2: distance_in_mid_range(R) \leftarrow distance_to_target(R) < 250$
$C1.1.2.2: distance_in_far_range(R) \leftarrow distance_to_target(R) < 500$
$C1.2: user_appropriate(U, R) \leftarrow competition_appropriate(U, R) \wedge allergen_appropriate(U, R) \wedge healthy_appropriate(U, R) \wedge eco_appropriate(U, R)$
$C1.2.1: competition_appropriate(U, R) \leftarrow user_type(U, competitive) \wedge awarded_route(R)$
$C1.2.2: allergen_appropriate(U, R) \leftarrow user_allergic(U, allergen) \wedge allergen_free_route(R, allergen)$
$C1.2.3: healthy_appropriate(U, R) \leftarrow user_type(U, athletic) \wedge healthy_route(R)$
$C1.2.4: eco_appropriate(U, R) \leftarrow user_type(U, ecologist) \wedge green_route(R)$
COLLABORATIVE RULES
$O1: recommend(U1, R) \leftarrow similarity(U1, U2) > \alpha \wedge vote(U2, R) \geq 4$
KNOWLEDGE BASED RULES
$K1: recommend(U, R1) \leftarrow similarity(R1, R2) > \beta \wedge vote(U, R2) \geq 4$
SYSTEM RULES
$S1: \sim recommend(U, R) \leftarrow system_balance_after_route(R) < \gamma$

The rules Δ of the DeLP program were used to derive literals and generate arguments to support them. Hence, in our system we can recommend a bike route to a specific user by deriving any of the $recommend(U, R)$ defeasible rules, where U is a user and R a route.

Definition 2 (Argument). *An argument \mathcal{A} for c (represented as $\langle \mathcal{A}, c \rangle$) is a minimal set of facts and defeasible rules that can be combined to derive the literal (or conclusion) c without contradictions.*

Therefore, arguments for recommending a route were generated by backward chaining of facts and defeasible rules. With this recursive method, the system was able to create a dialectical tree whose root was the target argument of discussion (i.e., whether to recommend or not a bike route for a specific user), and whose children nodes were arguments that defeat their predecessors [29].

In a similar way, arguments for not recommending a bike route can be generated as attacks to arguments for recommending that route. The system supported two types of classical attacks in computational argumentation: rebut (i.e., propose the opposite conclusion), and undercut (i.e., attack clauses of argument's body).

Definition 3 (Attack). *An argument $\langle \mathcal{B}, x \rangle$ attacks another argument $\langle \mathcal{A}, c \rangle$ if $\sim c$ can be derived from \mathcal{B} or if x overrides one of the clauses of \mathcal{A} (there exists a sub-argument $\langle \mathcal{A}_1, c_1 \rangle$ from $\langle \mathcal{A}, c \rangle$ such that $\Pi \cup \{c_1, x\}$ is contradictory).*

For instance, let us assume that a user $U2$ has previously on voted a bike route R with a score equal to or greater than four. Then, the system can use the rule $O1$ to recommend the bike route R for a similar user $U1$. Furthermore, we can derive an argument for not recommending the bike route R if there is a similar user and he/she does not like the bike route R (undercut).

When the system can derive several arguments in favour (support) or against (attack) of a route recommendation, the argument that prevails (wins the debate) is computed by computing a probability measure. Thus, we used an argumentation framework that assigned probability values to arguments and aggregated them to compute a value to rank and recommend bike routes. Each rule has associated a probability measure that aggregated the probability of the clauses that form the body of the rule to estimate the probability that an argument generated by using the rule succeeds.

Definition 4 (Argumentation Framework). *A argumentation framework for computational persuasion in our system was a tuple (Arg, P_{Arg}, Def) where Arg is a set of arguments, $Def \subseteq Arg \times Arg$ is a defeat relation, and $P_{Arg} : \rightarrow [0 : 1]$ is the probability that an argument $Arg = \langle \mathcal{A}, c \rangle$ holds.*

This probability was computed by means of the following formula:

$$P_{Arg} = \begin{cases} 1, & \text{if } \mathcal{A} \subseteq \Pi \\ \frac{\sum_{i=1}^k P_{Q_i}}{k}, & \text{if } \mathcal{A} \subseteq \Delta \mid c \leftarrow Q_1, \dots, Q_k \end{cases} \quad (1)$$

Facts have probability of one, and defeasible rules have a probability that results from the average of the probabilities their literals Q_1, \dots, Q_k (i.e., one for facts, zero for unresolved literals, or P_{Q_i} for those derived from other rules).

Definition 5 (Defeat). *An argument $\langle \mathcal{B}, x \rangle$ defeats another argument $\langle \mathcal{A}, c \rangle$ if \mathcal{B} attacks \mathcal{A} and $P_B > P_A$.*

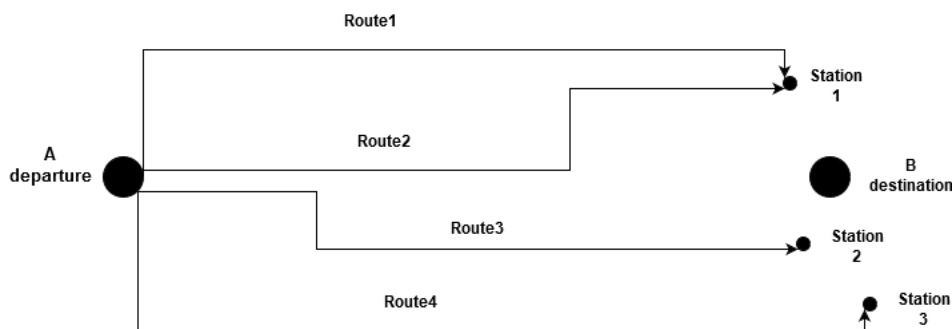
Furthermore, the system included a module for generating persuasive arguments based on rules, which was flexible to accommodate a variety of explanations. The small and finite number of current rules made it possible to associate each one with a scheme that can be used to generate a textual explanation for the underlying argument associated to the rule (see Table 3 for an example).

Table 3. Explanation schemes.

Rule	Explanation	Description
C1	E1	The route R it is suitable for you because it is generally appropriated, and moreover, it is suitable for your lifestyle.
O1	E2	The system has found a user that whose profile is similar to yours who liked route R.
K1	E3	The system has found that you liked route R2, which is similar to route R1.

6. Example Scenario

In this section we present an example scenario that allows us to illustrate how the persuasive recommender system works. In Figure 3 we present the scenario where the example is going to be framed. In this scenario there is a user that wants to transfer, using a bike sharing system inside a city from point A to point B. In order to simplify the trace, the starting point is a station, so we are not dealing with recommending a starting station in this example. The example is focused on the recommendation of a target station close to the user's desired destination. The destination is the place in the city where the user desires to arrive. It is not a station, as it usually is shown in other tools that service providers offer. This is because our persuasive system suggests the most interesting stations for both the user and the system; the station that strikes a balance between being close to the user's destination and balancing the whole system in terms of free spots and available bikes. Moreover, the system can consider the full route features instead of just the destination stations, adding more information from those alternative routes and guiding the user through the whole transfer process.

**Figure 3.** Example scenario.

In Figure 3 destination point B is surrounded by three dock stations. The system limited the stations that were considered by the recommender based on distance. It never considered stations that were too far away from the destination point, with the objective of not making the user walk uncomfortable distances. It also excluded as candidates any stations that were predicted to be full (no free spots) when the user was expected to arrive. This was done using the predictor component developed in [4]. In this scenario we considered three stations in order to show some illustrative cases. Note that there may have been more than one route from a point of origin to a destination, the difference in the route characteristics being the key point on which the recommendation process can obtain the knowledge required to persuade the user when offering him/her the alternatives. As was mentioned in Section 5, some of these characteristics were directly related to the route (availability of a bike lane along the route, the total distance or time it takes to travel the route, etc.). Other characteristics were related to the sharing system itself (availability of free spots, the balanced achieved after recommending a particular station, etc.), and finally, some characteristics were related to the habits or preferences of the user (allergies, awareness with the environment, etc.). These characteristics and their influence over the recommendation system are better described in Table 4.

Table 4. Fact description.

Fact	Description
<i>estimated_route_time</i> (R)	Estimated time needed to complete route R.
<i>distance_to_target</i> (R)	Distance between the departure point and the arrival point.
<i>user_type</i> (U)	Determines what type of user the system is dealing with. This fact can be instantiated more than once for the same user with different types (ecologist, athletic, competitive...)
<i>awarded_route</i> (R)	The route has an associated reward. This type of routes are more suitable for competitive users.
<i>healthy_route</i> (R)	The route is considered physically challenging. This type of routes are more suitable for competitive users.
<i>green_route</i> (R)	Most of the route goes through green areas, generally with less traffic and pollution or more natural elements. This type of routes are more suitable for ecologist users.
<i>user_allergic</i> (U, allergen)	The user U is allergic to that allergen.
<i>allergen_free_route</i> (R, allergen)	The route R is free from that allergen. These routes are advisable for users with allergies to allergen.
<i>similarity</i> (U1, U2)	U1 and U2 are similar users.
<i>similarity</i> (R1, R2)	R1 and R2 are similar routes.
<i>vote</i> (U, R)	U has voted the route R.
<i>system_balance_after_route</i> (R)	Measure for system balance after the proposed trip is completed.

In this example scenario we consider one user, *user1*, with the following characteristics:

- *user1*:
 - *user_allergic*(*user1*, *allergen1*)
 - *user_type*(*user1*, *healthy*).

These facts mean that *user1* is allergic to a specific allergen (e.g., *allergen1*). The user is also classified as *healthy*, which means that he/she has a preference for routes that are more physically challenging. Using this information the system can conclude that, when there are similar routes to recommend to the user, the ones that better fit the user's profile (taking into account these characteristics) are going to be recommended first.

In this scenario we also consider four different routes with the following characteristics:

- *route1*:
 - *fast_route*
 - *no_transfer_route*
 - *distance_in_close_range*
 - *allergen_free_route*
 - *healthy_route*
 - *green_route*
- *route2*:
 - *no_transfer_route*
 - *free_spots*
 - *distance_in_close_range*
 - *allergen_free_route*

- *healthy_route*
- *green_route*
- *route3*:
 - *no_transfer_route*
 - *free_spots*
 - *distance_in_mid_range*
- *route4*:
 - *bike_lane*
 - *no_transfer_route*
 - *fast_route*
 - *distance_in_close_range*
 - *awarded_route*
 - *allergen_free_route*
 - *healthy_route*
 - *green_route*
 - *system_balance_after_route* $< \gamma$

Once the example scenario is presented, we proceed to describe the process that the persuasive system proposed in this work follows. When the user location and the user desired destination were established, the system starts retrieving information from its knowledge base. First, it obtained the user's preferences and characteristics such as the type of user, allergies, previous decisions, etc. Then, the system retrieved possible routes and related information in order to determine their characteristics. When the recommender has collected the user information and the possible routes with their information, it starts to score every route using the DeLP rules. The semantic of each fact has been already explained in Table 4. Figure 4 shows an example of how some of the content-based rules are evaluated for *user1* (*U*) and *route2* (*R*).

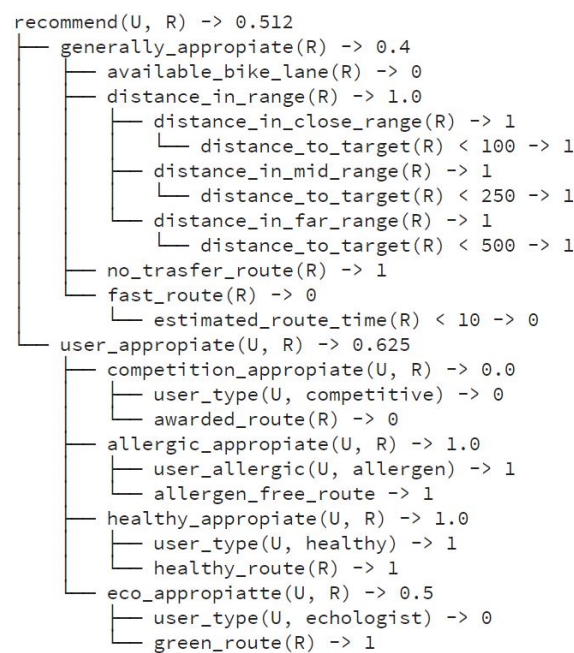


Figure 4. Example of content-based rules execution for *user1* and *route2*.

For the sake of simplicity we are not going to illustrate all the rules in this example. It is important to remember that each route can be supported (or rebutted) by several *recommend* (or \sim *recommend*)

rules with different priorities, since we have defined four types of rules (system rules, content-based rules, collaborative rules and knowledge rules), and any of those recommending rules can be positively evaluated. We also have to take into account that an attack may be generated if the negation of a rule is triggered at the same time of the corresponding opposite rule. As mentioned before, the recommender solved the attacks between rules using the scored priority. The higher the priority was, the more informed and specific the triggered rule was. These priorities are used to sort and select the recommended route, which will be the one with the higher priority that was not revoked by an attack. Table 5 shows the priority of each *recommend* rule for each of the four groups (content-based, collaborative, knowledge-based and system rules) and for the four selected routes.

If we focus on the content based rules, we can observe how the routes and user characteristics affect the given priority. Comparing *route1* and *route2* they have the same user related features, but they differ on route-only related characteristics, the first one being a *fast_route*. This fact provided a higher priority to *route1* in the *generally_appropriate* rule, obtaining a better overall score since the *user_appropriate* rule has the same priority between the two routes.

Table 5. Scores for user1 on each route.

Route	Rule Group	Priority
<i>route1</i>	C1	0.613
	O1	0
	K1	0
	S1	0
<i>route2</i>	C1	0.512
	O1	0
	K1	0
	S1	0
<i>route3</i>	C1	0.417
	O1	0
	K1	0
	S1	0
<i>route4</i>	C1	0.775
	O1	0
	K1	0
	S1	1

Following this rationale, we observe the low priority given to *route3*, since it lacks for some desirable characteristics compared to *route1* and *route2*. Some of them belonged to the *generally_appropriate* rule, such as *distance_in_close_range*, and others belonged to the *user_appropriate* rule, such as *healthy_route*. The *route4* was the one with the highest priority due to the amount of positive characteristics with respect to either the route itself and the relation between the user and the route. Because of this, looking at content-based rules, *route4* should be the one to be recommended since it suits better the user preferences than others in this specific scenario.

Despite this, when accounting the rest of rules, the system arrives to an attack conflict between arguments that must be solved. Fact *system_balance_after_route(route4)* determines that there is a problem regarding to the bike sharing system balance with *route4*. When the system balance for a transfer to a station falls below a determined value it is considered counterproductive and the route should not be recommended. As shown in Table 5, the recommendation system is facing this situation and the $\sim \text{recommend}(\text{route4})$ can be derived from an argument. Since *recommend(route4)* can also be derived, the system is facing an attack between arguments. As stipulated in Section 5 each rule has an associated probability measure with the purpose of estimating the success of the arguments. In this context, *recommend(route4)* with a probability measure of 0.775 value and $\sim \text{recommend}(\text{route4})$ with a probability measure of 1 are attacking each other. Using the procedure explained above,

$\sim recommend(route4)$ with a higher value in its probability measure rebuts $recommend(route4)$, which has a lower probability measure. Thus, $\sim recommend(route4)$ wins the conflict, and $route4$ is not recommended. Once these conflicts are resolved, the system should recommend the first route with the highest priority value, which is $route1$.

We have kept collaborative and knowledge-based rules out of this scenario for readability and space purposes, but they work following the same pattern as the rules mentioned in the example. For instance, if $route3$ had had very positive votes from a user similar to $user1$ (for collaborative routes) or similar routes that $user1$ had previously scored high (for knowledge-based rules), then more $recommend$ rules would have been triggered with a priority higher than zero.

Finally, when the system has all the possible routes to all the possible alternative stations scored, we can propose to the user an alternative station which can be reached following a route that suits the user's preferences and that results in a good system balance.

7. Persuasion Impact on System Balance

The recommender proposed in this work relied on persuading the bike sharing system users to maintain the system as balanced as possible, therefore it is necessary to study how persuasion of users influences in the system balancing. To test this influence we conducted an experiment that consisted of a system simulation. The goal of this experiment was to validate that persuading users to go to alternative stations may help to improve the balance of the system. The experiment was designed as follows: First, a scenario has been designed with characteristics similar to those of a bike sharing system in a real city such as Valencia (Spain). This scenario is made up of 300 stations with different capacities (adding a total of 7000 bike parking slots) and 4500 bikes. Next, we applied our recommender to the scenario and simulated a set of trips. This set of trips has been repeated varying the persuasive capacity of the recommender, from scenarios where the persuader was not able to convince any user (users park their bikes at their preferred destination stations, if they are not full, or at the closest one), to scenarios where it convinces 100% of users (users always park their bikes in the recommended station). Thus, the impact of the system in the balance can be studied based on the percentage of users that are being persuaded. The experiment was repeated two thousand times to verify its reliability in obtaining consistent results between each repetition.

Figure 5 shows the average result of the set of experiments performed confronting the system balance evolution regarding the percentage of users persuaded. This figure shows how positive it is to persuade users to move to alternative stations selected by the system (instead of letting them decide). The more users persuaded (the better the persuader) the better balanced the objective system will be. This, as it can be seen in Figure 5, improves the balancing measure by up to 21.01%.

Figure 6 shows how the system improves over time when a persuasion module such as the one proposed in this paper is included. In this figure the number of trips over time is shown. This is compared to the system balancing for four different configurations of persuasion (0%, 25%, 50% and 100%) acceptance. As it can be observed, when the recommendation strategy is not used, the system balance does not improve along time. However, this changes when we increase the percentage of users persuaded to accept the system recommendations, which enhance the balance accordingly with the acceptance ratio. In particular we can see how, when we convince 25% of users the improvement is 15.25%, when we go to convince 50% of users we improve up to 20.89% and if we convince 100% of users we would have an improvement in the system balance of 32.02%. This experiment shows us that the longer the persuasion is running, the better the balancing results of the system will be achieved. Analyzing these results, we can conclude that our recommendation strategy results in a positive impact for the overall system balance.

These experiments constituted a proof of concept for our persuasive system, which demonstrates the potential of persuasion algorithms to solve the balancing problem. As future work, we plan to carry out an online evaluation test that demonstrates the real effectiveness of our persuasive system in

production (how much it helps the user to make good decisions) [30], when enough registered users are reached.

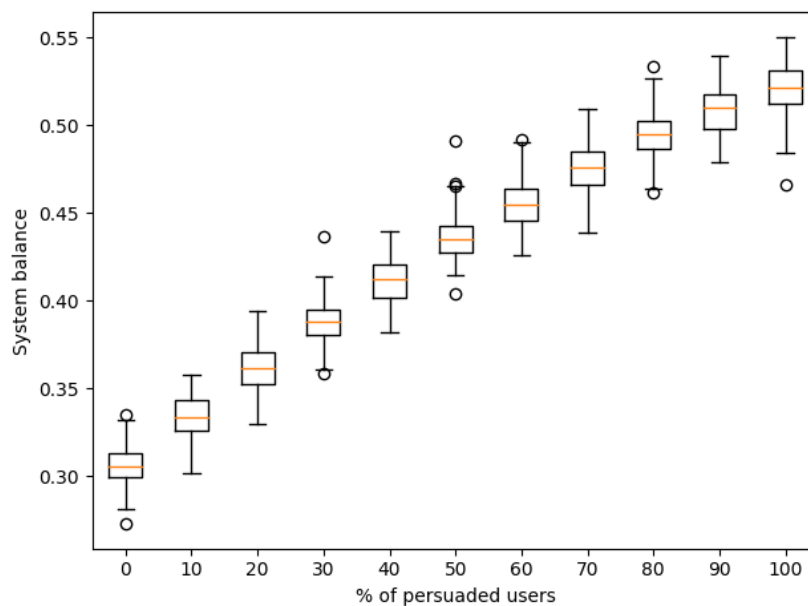


Figure 5. Impact of % of persuaded users in overall system balance.

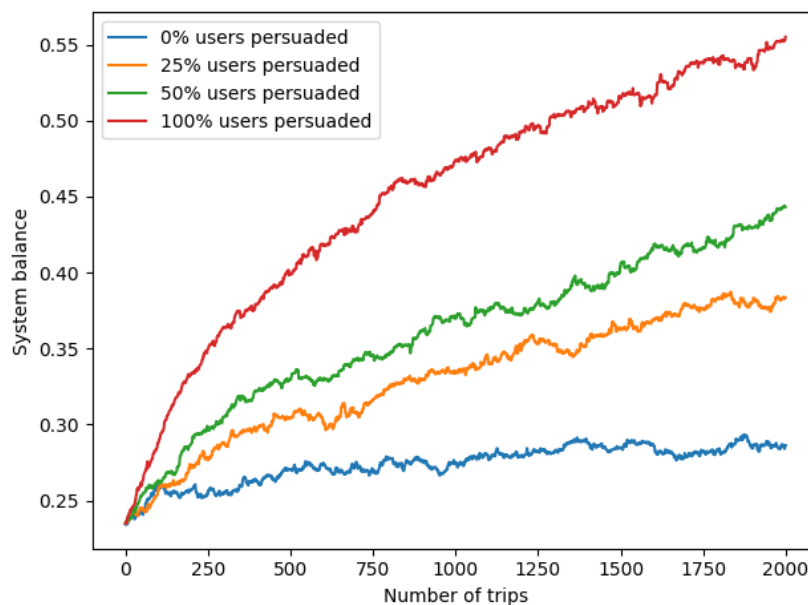


Figure 6. Impact of % of persuaded users along number of trips.

8. Conclusions

In this paper an argumentation-based persuasive approach for recommending alternative stations for bike sharing systems has been proposed. This recommender is useful in bike sharing systems where users cannot always park their bikes in the desired station because the station is full at the time of the user's arrival. In such cases, instead of suggesting a free station after the user's trip (with the consequent annoyance on the part of the user), we propose to predict this situation to be able to suggest an alternative station to the user before his trip, accompanying this change of a proposed route to that station that is adapted to the preferences of the user. This proposal constitutes our first step

towards persuading users to contribute to the general balance of the bike network. The proposed framework is intended to influence the stations chosen by the users in a bicycle sharing system, and, in doing so, to achieve a balanced bicycle station network without dedicated management trucks that move the bicycles among the stations. As far as we know, there is no similar system for trying to improve the overall performance of a bike sharing system using persuasion mechanisms and without using economic incentives.

The proposed system is not designed exclusively for leisure travels. Actually, the system tries to avoid unwanted situations in which a user has to look for a nearby station when trying to leave a bicycle at a station that is full. Whether the trip is for leisure, work or, studies, the user wants to leave the bike as close as possible to its destination and without wasting time looking for alternatives or going to stations that are finally full. This is why this system helps the end user to optimise their trip in advance, with the addition that it tries to improve their satisfaction by means of the persuasion techniques presented in this work. At a global level, the system tries to balance the network of stations in order to avoid this type of situations in the future and reduce bicycle removals via truck.

The proposed framework uses a defeasible argumentation formalism based on logic programming to encode the information about the facts and rules. Its utility is easily expandable to adapt to different use cases by simply adding new rules. The framework uses knowledge about the routes, the bike sharing system and the user preferences and characteristics to intend to arrive to a trade-off between the user desires and the system balance needs, creating arguments in the process. These arguments are used to persuade the user in his route election. To resolve attacks between arguments, each rule has an associated probability measure that estimates the probability that an argument (generated by using the rule) succeeds based on the aggregated probability of the clauses that form the body of the rule. The framework was showcased with an example scenario. As future work, we plan to include more criteria to define the users' preferences and context, a comprehensive evaluation tests with humans, and we will study how to enhance the explanation module with an advanced human-computing interaction module.

Author Contributions: conceptualization, J.P. and V.S.-A.; formal analysis, S.H.; funding acquisition, A.G. and V.J.; investigation, C.D., J.P. and V.S.-A.; methodology, S.H.; project administration, A.G. and V.J.; software, C.D.; supervision, A.G. and V.J.; writing—original draft, C.D.; writing—review and editing, J.P., V.S.-A. and S.H.

Funding: This work was supported by the projects TIN2015-65515-C4-1-R and TIN2017-89156-R of the Spanish government, and by the grant program for the recruitment of doctors for the Spanish system of science and technology (PAID-10-14) of the Universitat Politècnica de València.

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

References

1. Kull, M.; Ferri, C.; Martínez-Usó, A. Bike Rental and Weather Data across Dozens of Cities. In Proceedings of the ICML 2015 Workshop on Demand Forecasting, Lille, France, 11 July 2015.
2. Li, Y.; Zheng, Y.; Zhang, H.; Chen, L. Traffic prediction in a bike-sharing system. In Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, Washington, DC, USA, 3–6 November 2015; p. 33.
3. Yoon, J.W.; Pinelli, F.; Calabrese, F. Cityride: A predictive bike sharing journey advisor. In Proceedings of the 2012 IEEE 13th International Conference on Mobile Data Management (MDM), Bengaluru, India, 23–26 July 2012; pp. 306–311.
4. Diez, C.; Sanchez-Anguix, V.; Palanca, J.; Julian, V.; Giret, A. Station status forecasting module for a multi-agent proposal to improve efficiency on bike-sharing usage. In Proceedings of the 5th International Conference on Agreement Technologies, Évry, France, 14–15 December 2017.
5. Caggiani, L.; Ottomanelli, M. A modular soft computing based method for vehicles repositioning in bike-sharing systems. *Procedia Soc. Behav. Sci.* **2012**, *54*, 675–684. [[CrossRef](#)]
6. Ho, S.C.; Szeto, W. Solving a static repositioning problem in bike-sharing systems using iterated tabu search. *Transp. Res. Part E Logist. Transp. Rev.* **2014**, *69*, 180–198. [[CrossRef](#)]

7. Vogel, P.; Saavedra, B.A.N.; Mattfeld, D.C. A hybrid metaheuristic to solve the resource allocation problem in bike sharing systems. In *Hybrid Metaheuristics, Proceedings of the 9th International Workshop, HM 2014, Hamburg, Germany, 11–13 June 2014*; Springer: New York, NY, USA, 2014; pp. 16–29.
8. Erdoğan, G.; Laporte, G.; Calvo, R.W. The static bicycle relocation problem with demand intervals. *Eur. J. Oper. Res.* **2014**, *238*, 451–457. [[CrossRef](#)]
9. O’Mahony, E.; Shmoys, D.B. Data Analysis and Optimization for (Citi)Bike Sharing. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015; pp. 687–694.
10. Reiss, S.; Bogenberger, K. GPS-Data Analysis of Munich’s Free-Floating Bike Sharing System and Application of an Operator-based Relocation Strategy. In Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC), Las Palmas, Spain, 15–18 September 2015; pp. 584–589.
11. Alvarez-Valdes, R.; Belenguer, J.M.; Benavent, E.; Bermudez, J.D.; Muñoz, F.; Vercher, E.; Verdejo, F. Optimizing the level of service quality of a bike-sharing system. *Omega* **2016**, *62*, 163–175. [[CrossRef](#)]
12. Schuijbroek, J.; Hampshire, R.; van Hoes, W.J. Inventory rebalancing and vehicle routing in bike sharing systems. *Eur. J. Oper. Res.* **2017**, *257*, 992–1004. [[CrossRef](#)]
13. Singla, A.; Santoni, M.; Bartók, G.; Mukerji, P.; Meenen, M.; Krause, A. Incentivizing Users for Balancing Bike Sharing Systems. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015; pp. 723–729.
14. Aeschbach, P.; Zhang, X.; Georghiou, A.; Lygeros, J. Balancing bike sharing systems through customer cooperation—a case study on London’s Barclays Cycle Hire. In Proceedings of the 2015 IEEE 54th Annual Conference on Decision and Control (CDC), Osaka, Japan, 15–18 December 2015; pp. 4722–4727.
15. Li, L.; Shan, M. Bidirectional incentive model for bicycle redistribution of a bicycle sharing system during rush hour. *Sustainability* **2016**, *8*, 1299. [[CrossRef](#)]
16. Anagnostopoulou, E.; Bothos, E.; Magoutas, B.; Schrammel, J.; Mentzas, G. Persuasive technologies for sustainable mobility: State of the art and emerging trends. *Sustainability* **2018**, *10*, 2128. [[CrossRef](#)]
17. Quercia, D.; Schifanella, R.; Aiello, L.M. The shortest path to happiness: Recommending beautiful, quiet, and happy routes in the city. In Proceedings of the 25th ACM Conference on Hypertext and Social Media, Santiago, Chile, 1–4 September 2014; pp. 116–125.
18. Galbrun, E.; Pelechris, K.; Terzi, E. Urban navigation beyond shortest route: The case of safe paths. *Inf. Syst.* **2016**, *57*, 160–171. [[CrossRef](#)]
19. Mirri, S.; Prandi, C.; Rocchetti, M.; Salomoni, P. Walking under a Different Sky: Urban Colored Routes for Creative Engagement and Pleasure. *Int. J. Hum. Comput. Interact.* **2017**, *33*, 1010–1021. [[CrossRef](#)]
20. Mirri, S.; Prandi, C.; Salomoni, P. Personalizing Pedestrian Accessible way-finding with mPASS. In Proceedings of the 2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 9–12 January 2016; pp. 1119–1124.
21. Ferrara, J. Games for persuasion: Argumentation, procedurality, and the lie of gamification. *Games Cult.* **2013**, *8*, 289–304. [[CrossRef](#)]
22. Kazhamiakin, R.; Marconi, A.; Perillo, M.; Pistore, M.; Valetto, G.; Piras, L.; Avesani, F.; Perri, N. Using gamification to incentivize sustainable urban mobility. In Proceedings of the 2015 IEEE First International Smart Cities Conference (ISC2), Guadalajara, Mexico, 25–28 October 2015; pp. 1–6.
23. Huang, B.; Thomas, T.; Groenewolt, B.; Fioreze, T.; van Berkum, E. The Effect of Incentives to Promote Cycling: A Mobility Living Lab. In Proceedings of the Transportation Research Board 97th Annual Meeting, Washington, DC, USA, 7–11 January 2018.
24. Fei, X.; Shah, N.; Verba, N.; Chao, K.M.; Sanchez-Anguix, V.; Lewandowski, J.; James, A.; Usman, Z. CPS data streams analytics based on machine learning for Cloud and Fog Computing: A survey. *Future Gener. Comput. Syst.* **2019**, *90*, 435–450. [[CrossRef](#)]
25. Diez, C.; Sanchez-Anguix, V.; Palanca, J.; Julian, V.; Giret, A. A Multi-agent Proposal for Efficient Bike-Sharing Usage. In *PRIMA 2017: Principles and Practice of Multi-Agent Systems*; An, B., Bazzan, A., Leite, J., Villata, S., van der Torre, L., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 468–476.
26. Faed, A.; Hussain, O.K.; Chang, E. A methodology to map customer complaints and measure customer satisfaction and loyalty. *Serv. Oriented Comput. Appl.* **2014**, *8*, 33–53. [[CrossRef](#)]
27. Xu, W.; Li, Z.; Cheng, C.; Zheng, T. Data mining for unemployment rate prediction using search engine query data. *Serv. Oriented Comput. Appl.* **2013**, *7*, 33–42. [[CrossRef](#)]

28. Hernández, E.; Sanchez-Anguix, V.; Julian, V.; Palanca, J.; Duque, N. Rainfall prediction: A deep learning approach. In *Hybrid Artificial Intelligent Systems, Proceedings of the 11th International Conference, HAIS 2016, Seville, Spain, 18–20 April 2016*; Springer: New York, NY, USA, 2016; pp. 151–162.
29. García, A.; Simari, G. Defeasible logic programming: An argumentative approach. *Theory Pract. Log. Program.* **2004**, *4*, 95–138. [[CrossRef](#)]
30. Tintarev, N.; Masthoff, J. Designing and evaluating explanations for recommender systems. In *Recommender Systems Handbook*; Springer: New York, NY, USA, 2011; pp. 479–510.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).