UNIVERSITÀ
DEGLI STUDI
FIRENZE

Design and implementation of a Smart Bike-Sharing system

Gabriele Bruni

Firenze, 27/02/2020

Introduction

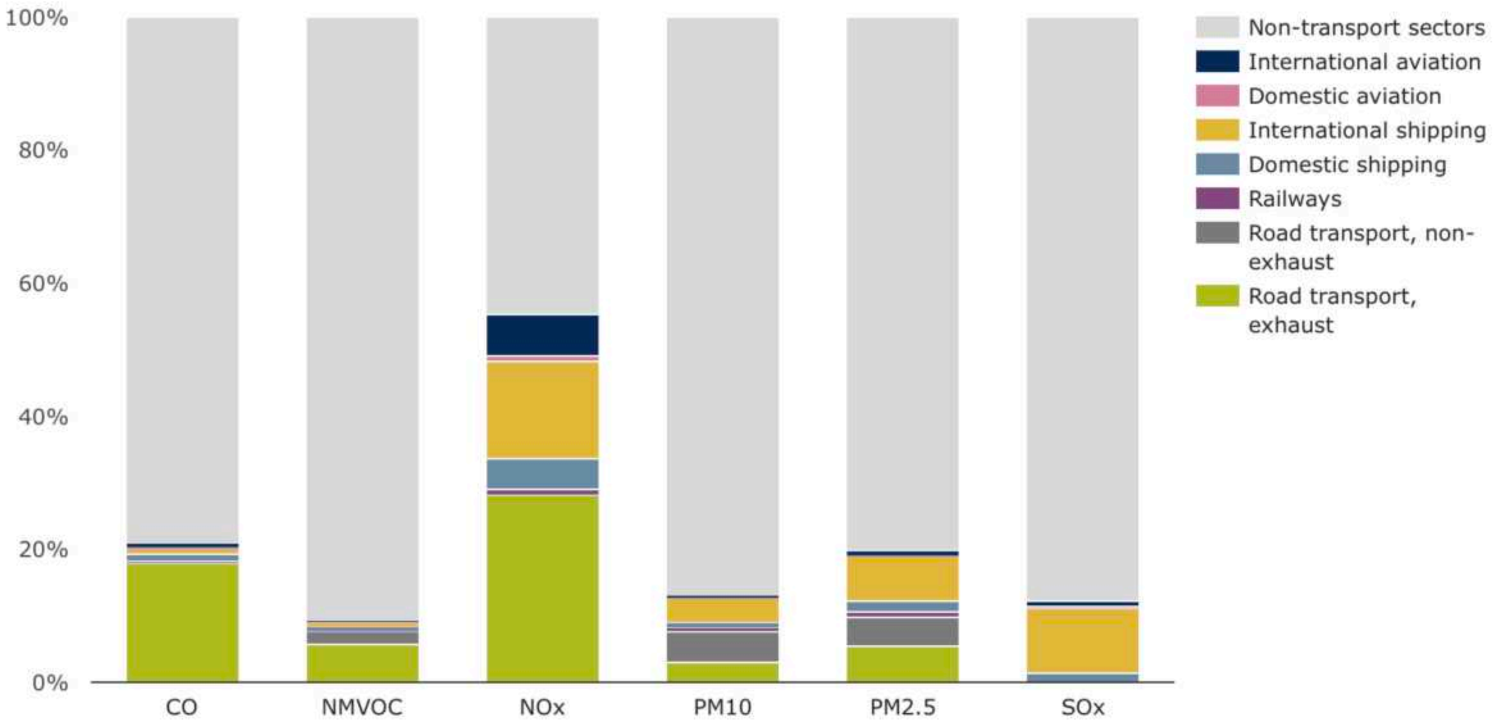
The Smart City vision

Smart Governance	Smart Economy	Smart Mobility
Smart Environment	Smart People	Smart Living

Introduction

The need for a sustainable mobility

Chart — Contribution of the transport sector to total emissions of the main air pollutants



Transport accounts for over two thirds of NOx emissions and contributes over a quarter to the emission of polluting gases.

Introduction

Bike-Sharing as a solution

Alternative to traditional public transport

Environmental friendly

Prevents traffic congestion

Available 24 hours a day

Health friendly

Introduction

Bike-Sharing problem

Load balancing problem: users do not know if bicycles will be available at a certain time of the day.

Unpredictability problem: service operators need to redistribuites bikes during the night.

Introduction

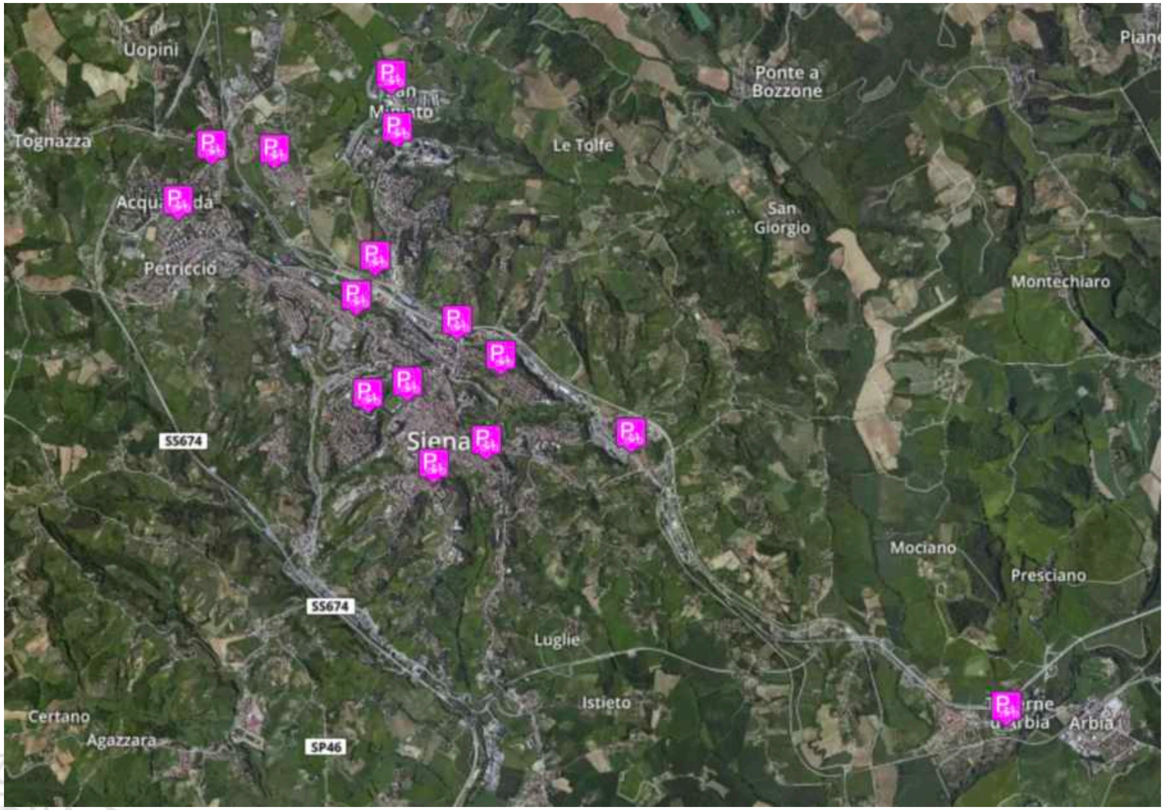
Objective of the thesis work

Implementation of a Smart Bike-Sharing system for the city of Siena:

- Research and experimenting new solution to **predict the availability of bikes for each station**.
- Development of a forecast model to **predict the number of the available bikes with a 15 minutes resolution in the next 24 hours**.
- **Real-time implementation** of the model within the KM4City infrastructure and the “Toscana, dove, cosa” mobile app.

Datasets

Siena's Bike-Sharing service



15 bicycle stations:

- Due ponti
- Taverne D’Arbia
- Ravacciano
- San Girolamo
- Casa dell’ambiente
- San Agostino
- Curtatone
- Fortezza
- Terminal Bus
- Antiporto
- Ospedale
- San Miniato
- Vico Alto
- Napoli
- Acquacalda

Datasets

Data from bicycle stations

Data were recorded in the period between **25 June 2019** and **15 January 2020**.

Each station records the number of bicycles available **every 15 minutes**, for a total of:

- 96 observation a day,
- 672 observation a week,
- about **15.000** in the considered period.

Features recorded:

- number of bicycles available,
- number of free slots,
- number of broken bikes,
- time of detection.

Due Ponti			
availablesBikes	freeSlots	brokenBikes	Time
12	4	0	
8	8	1	

15.000 obs.

Datasets

Data from weather station

Data were recorded in the period between **25 June 2019** and **15 January 2020**.

The station record weather data every 1 hour, for a total of :

- about **4.000** observation in the considered period

Features recorded:

- air temperature,
- air pressure,
- wind speed,
- visibility,
- cloud cover precentage,
- minimum temperature of the day,
- maximum temperature of the day,
- weather,
- sunrise time,
- sunset time

Datasets

Data from air quality station

Data were recorded in the period between **25 June 2019** and **15 January 2020**.

The station record weather data every 1 day, for a total of :

- about **160** observation in the considered period

Features recorded:

- CO
- NO2

Datasets

Other variables

Other features have been created starting from those just listed, with the aim of providing new relationships between the input vector and the output response:

- seasons, month, day
- time slot within the same day,
- number of bicycles available one week and one day before,
- difference between the number of bicycles at time t and the number of bicycles at time t - 1 calculated on the observations of the previous week and the previous day,
- difference between the number of bicycles at time t + 1 and the number of bicycles at time t calculated on the observations of the previous week and the previous day,

Datasets

The final sets of data

That result in:

- **15** datasets
- about **15.000** observations in each dataset
- **30** features

Due Ponti								
Base features			Weather features			Air quality features		
availableBikes				…				NO2
12								0.2
				…				
8								0.6

15.000 obs.

30 features

Descriptive analysis

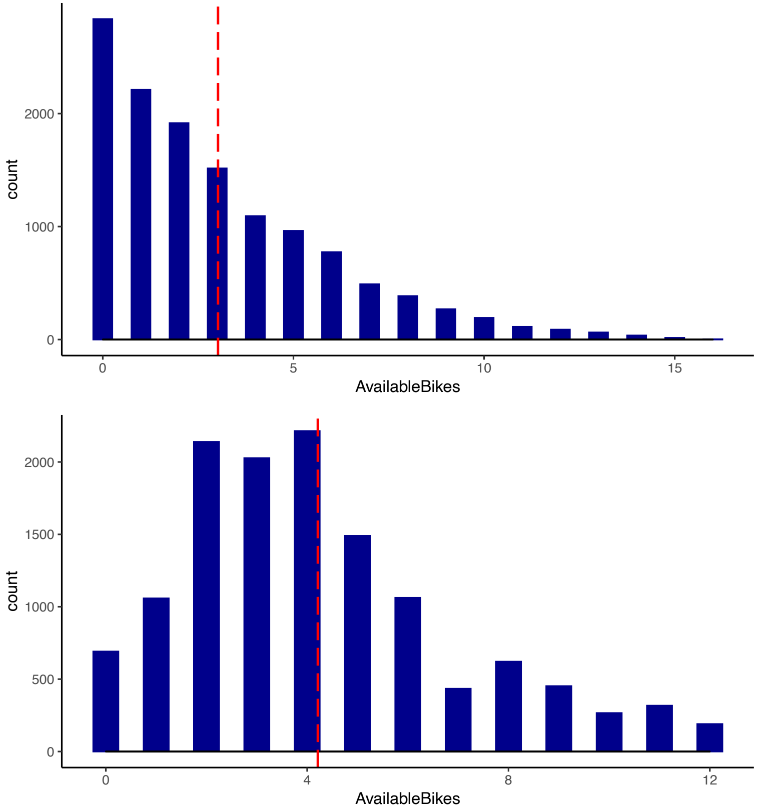
Data distributions and stats

Some descriptive analyzes were carried out to understand the sample:

- **distributions** of available bikes through all the parks,
- **trends** comparison between **working** days and **weekend**,
- **trends** comparison between **winter** and **summer** days,
- **clustering**.

Descriptive analysis

Data distributions and stats



Distribution of the number of bicycles available
in Curtatone and Due Ponti

Result:

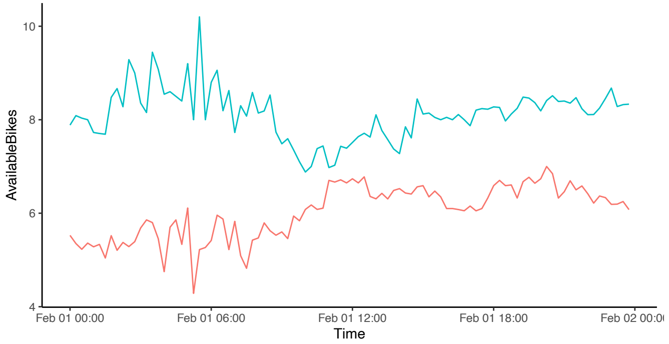
- Heterogeneous distributions
- **Unbalanced data**
- 5 to 16 slot available for storing bicycles

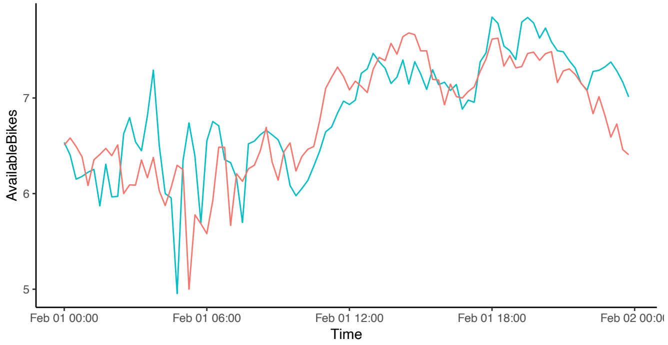
Descriptive analysis

Trends

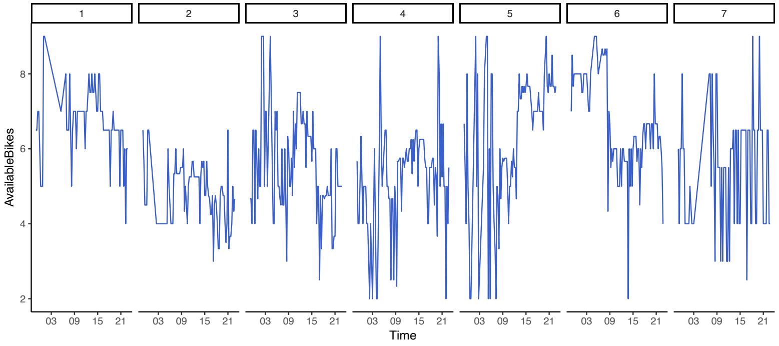
Result:

- Similar working/weekly day behavior
- Different seasonal behavior
- Noisy data





Comparison between the number of bicycles available in summer and winter (up) and in working days and weekend (down) in Terminal Bus.



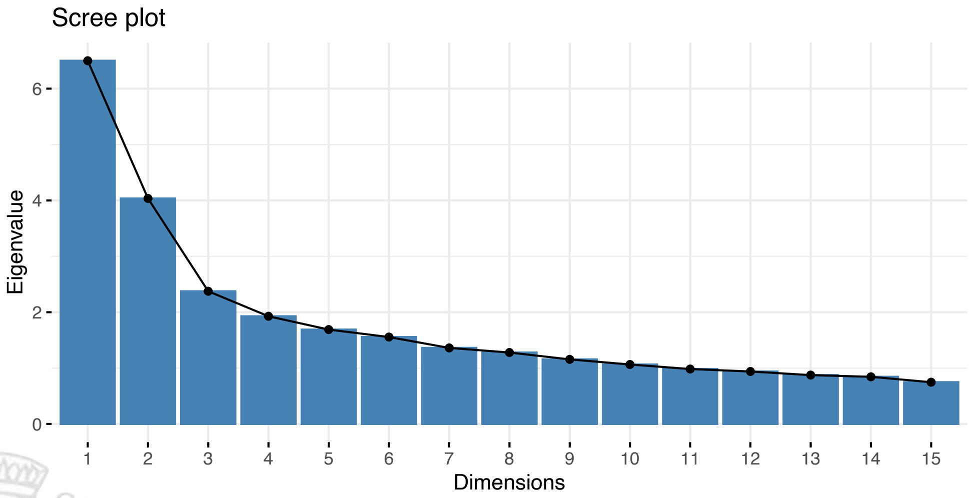
Weekly trend of Due Ponti park.

Descriptive analysis

Principal Component Analysis

Idea: use of the correlation matrix to find sub-sets of highly correlated features that explain the variance of the data.

Purpose: eventually eliminate some of those variables to reduce the complexity of the model.



Result:

- first 10 component account for 70% of variance
- first 10 eigenvalue > 1

Descriptive analysis

Principal Component Analysis

The first 10 components have been analyzed.

Possibly redundant variables:

- Min-Temperature, Max-Temperature, AirTemperature
- CloudCoverPercentage, CO, NO2
- Weekend
- Sunrise, Sunset

Descriptive analysis

Clustering

Hopkins statistic > 0.8 → data is clusterable.

Use of **K-means algorithm** (Hartigan–Wong version).

Objective function: minimization of total intra-cluster variance.

Number of cluster:

- Elbow method: 5 cluster
- Silhoutte method: 3 cluster
- Based on geographic position: 5 cluster

→ selected number of cluster: 5

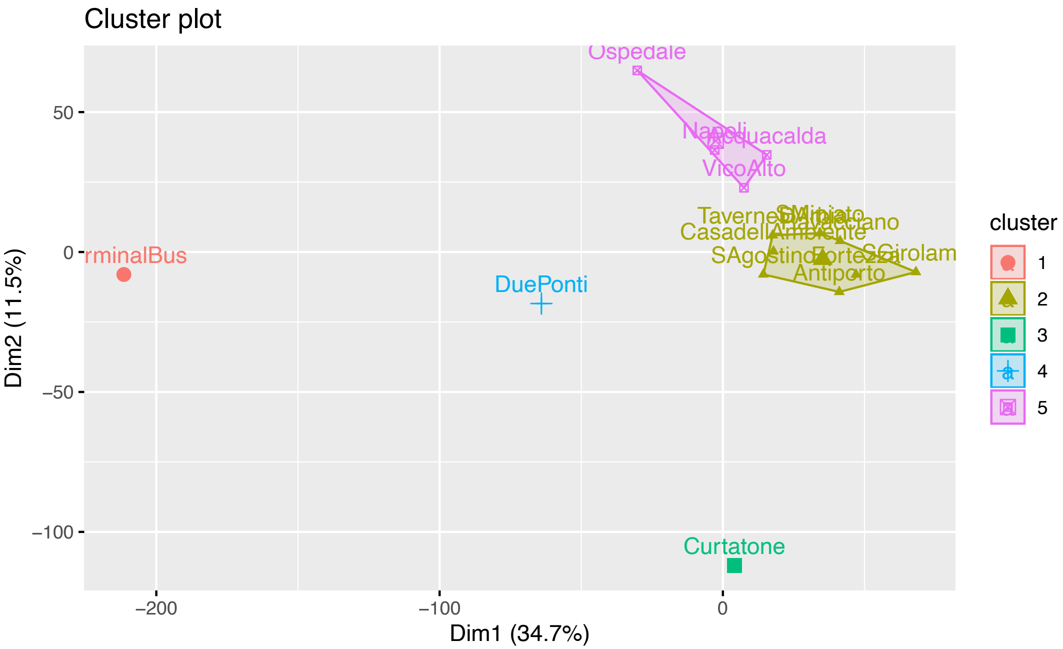
Number of iterations: 10

Clustering:

1. considering the whole year
2. considering summer/winter dynamics
3. considering working day/weekends dynamics

Descriptive analysis

Clustering



Clustering of availablesBikes time series considering the whole year.

Result:

- dynamics of some bikes parks is different between summer and winter,
- no significant differences between working days and weekends,
- 3 car parks are identified with dynamics totally different from the others: TerminalBus, DuePonti, Curtatone.

Modelling

Introduction

Purpose: provide the number of bicycles available in each bicycles park with 15-minute intervals over a 24-hour period.

Focus on:

- computational **efficiency** (processing time),
- **robustness**: behavior of the model in critical cases (when the number of bicycles available in the parking lot is close to 0).

Aim: implementation in a real-time services that run on the KM4City platform.

Two main algorithmic approach used:

- classic **univariate** statistical method
- **multivariate** supervised learning method

Modelling

Introduction

Training set: data between 25 June 2019 and 8 January 2020, for a total of about 12,000 observations.

Test set: each test set consists of 96 subsequent observations, sampled in a period between 8 January 2020 and 15 January 2020, for a total of 7 test sets (672 observations). The final error is calculated on all test sets.

Used data: 5 parks representing the 5 clusters identified with the K-means algorithm, selected also taking into account the imbalance and the noise of the time: Curtatone, Napoli, Terminal Bus, Due Ponti, Ospedale.

Modelling

Metrics

Test Score:

- **MASE:** independent of the data scale. Well defined even when the real observed value is 0.

$$MASE = mean(q_t)$$

$$q_t = \frac{|\hat{y}_t - y_t|}{\frac{1}{n-1}\sum_{i=1}^{n}|y_i - y_{i-1}|}$$

Train Score:

- **R^2:** ratio of the explained variance to the total variance. Good for measuring how well model fit the data.

$$R^2 = \frac{ESS}{TSS}$$

$$ESS = \sum_{i=1}^{n}(\hat{y}_i - \overline{y}_i)^2$$

$$TSS = \sum_{i=1}^{n}(y_i - \overline{y}_i)^2$$

Other score:

- **RMSE:** the classic root means squared error. Available for both training and test score.
- **MAPE:** mean absolute percentage error on test score. Undefined when the real observation is 0.

Modelling

Used models

Univariate method:

- **Auto Regressive Integrated Moving Avarage model (ARIMA)**

Machine learning methods:

- **Random Forest (RF)**
- **Gradient Boosted Machine (GBM)**
- **Bayesian Regularized Neural Network (BRNN)**

Modelling

Models configuration

Random Forest (RF)
- number of trees: 500
- features bagging ratio: 1/3
- minimum number of observation in each leaf: 5

Gradient Boosted Machine (GBM)
- base learner: regression tree
- max depth: 9
- number of trees: 500
- features bagging ratio: 1/3
- minimum number of observation in each leaf: 5
- learning rate: 0.1

Bayesian Regularized Neural Network (BRNN)
- number of hidden layer: 1
- number of neurons: 5
- activation function: hyperbolic tangent
- weight initialization: Nguyen-Widrow method

Modelling

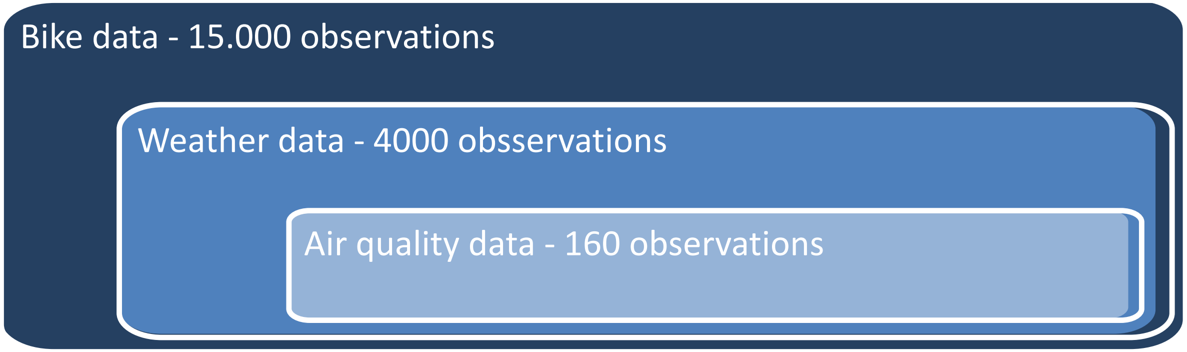
Imputation of missing data

Data was collected through the KM4City platform.

Sensor malfunction or momentary interruptions of the platform itself **could prevent correct detection of information**.

In addition, **the update time used for data collection turned out to be different** between bicycle, atmospheric and air quality stations.

Resulting missing data need to be imputed.



Bike data - 15.000 observations

Weather data - 4000 obsservations

Air quality data - 160 observations

Modelling

Imputation of missing data

Univariate model:

- **Multiple Imputaton by chained equation (MICE)**
  - regressor: random forest
  - n. of iterations: 5

Multivariate model:

- **Manual imputation**
  - deletion of observation where target variables where missing
  - adaptive window moving average
  - sobstitution with previus non-NA value

Resulting set of data:

- 13.000 observation

Modelling

Model overview

	ARIMA	RF	GBM	BRNN
Average training time	30.9 s	410.3 s	21.8 s	13.5 s
Training frequency	1 time per hour	1 time per day	1 time per day	1 time per day
Training period	1 month	7 month	7 month	7 month
Forecast window	1 hour	1 day	1 day	1 day

Note how forecast window for ARIMA model is only one hour.

Modelling

ARIMA results

	Test Score	
	MASE	**RMSE**
Curtatone	4.39	1.58
Napoli	6.95	1.1
Terminal Bus	4.43	1.32
Due Ponti	12.69	1.15
Ospedale	4.43	1.39

Forecasting technique: **multi-step forward with updated iteration**.

The forecast is made for one hour, the real observations recorded in that hour are inserted into the training set, and the forecast is made for the next hour.

Solution discarded as not suitable for implementation in a real-time application.

Modelling

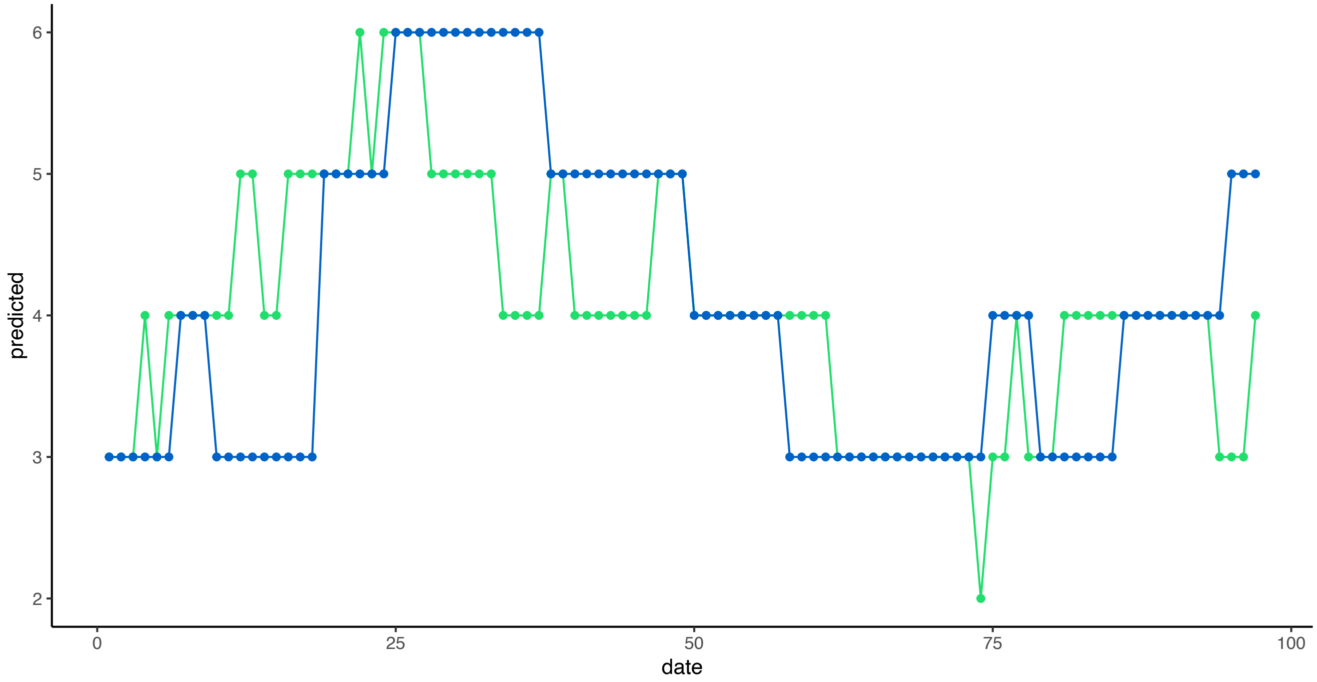
Machine learning models results

	Train Score				Test Score		
Model	**RF**	**GBM**	**BRNN**		**RF**	**GBM**	**BRNN**
	Napoli						
R^2	0.922	0.8	0.39		-	-	-
MASE	-	-	-		9	7.4	7.9
RMSE	0.5	0.81	1.27		1.5	1.4	1.58
MAPE	-	-	-		-	0.37	0.35
	Terminal Bus						
R^2	0.911	0.83	0.49		-	-	-
MASE	-	-	-		7.0	5.8	12.0
RMSE	0.81	1.1	1.92		2.2	2.1	3.2
MAPE	-	-	-		0.24	0.22	0.42

The **best model** was found to be the **GBM**.

Modelling

GBM results



Legend:

🟩 **Predicted** values

🟦 **Real** values

Forecasted values using the selected GBM model on Casa dell’Ambiente.

Implementation

KM4City platform

The prediction algorithm was developed using the **KM4City** (Knowledge Model for City) platform.

This provides an infrastructure that collects heterogeneous data.

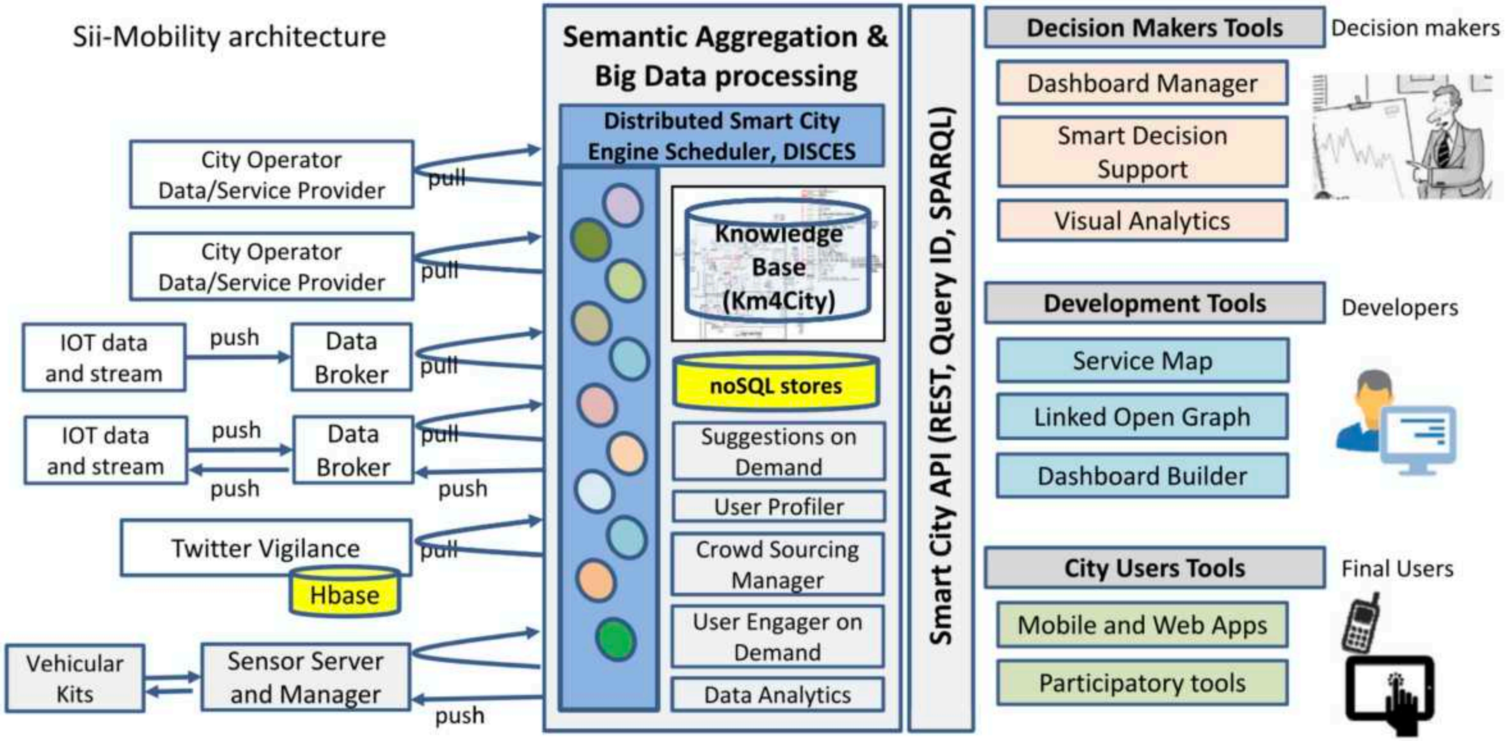
Data is aggregated and reconciliated on the basis of the model defined through the Km4City ontology.

The large amount of data collected allows to perform analyzes and process on them, allowing the development of **services** for smart cities.

Service are made available through the **Smart City API** and the **Service Map.**

Implementation

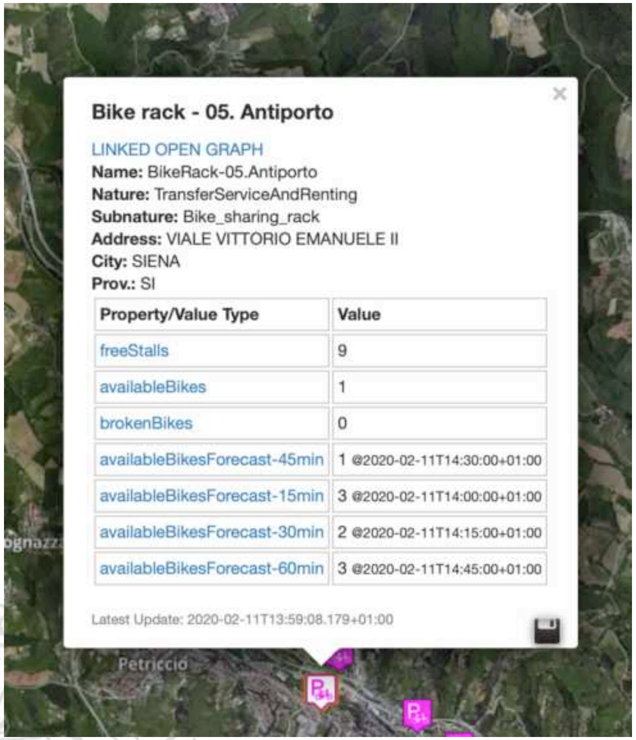
KM4City platform



Implementation

Service

A service that allows users and operators to take advantage of the forecasts made by the model has been released on kM4City platform. Available through Smart City API and Service Map.

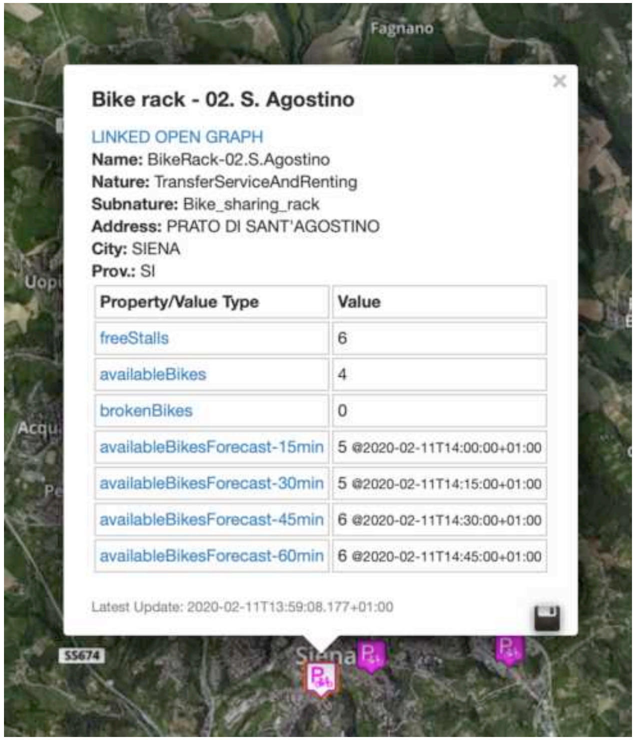


Bike rack - 05. Antiporto

LINKED OPEN GRAPH
Name: BikeRack-05.Antiporto
Nature: TransferServiceAndRenting
Subnature: Bike_sharing_rack
Address: VIALE VITTORIO EMANUELE II
City: SIENA
Prov.: SI

Property/Value Type	Value
freeStalls	9
availableBikes	1
brokenBikes	0
availableBikesForecast-45min	1 @2020-02-11T14:30:00+01:00
availableBikesForecast-15min	3 @2020-02-11T14:00:00+01:00
availableBikesForecast-30min	2 @2020-02-11T14:15:00+01:00
availableBikesForecast-60min	3 @2020-02-11T14:45:00+01:00

Latest Update: 2020-02-11T13:59:08.179+01:00



Bike rack - 02. S. Agostino

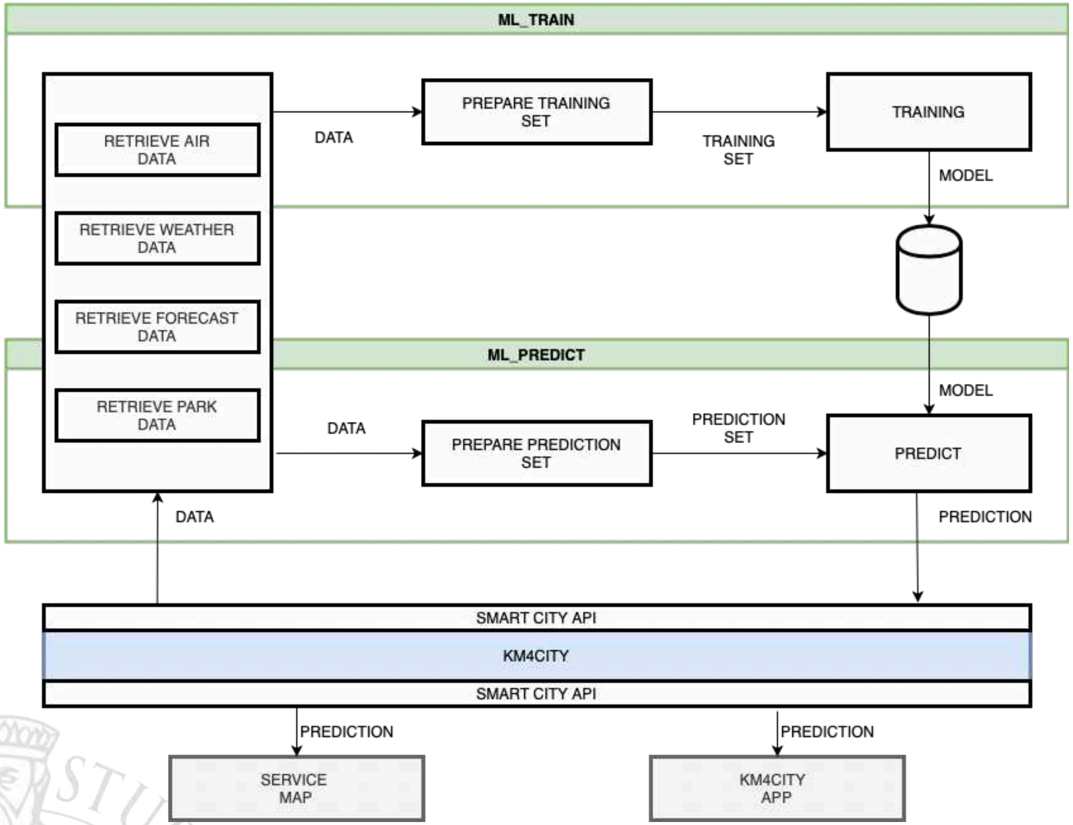
LINKED OPEN GRAPH
Name: BikeRack-02.S.Agostino
Nature: TransferServiceAndRenting
Subnature: Bike_sharing_rack
Address: PRATO DI SANT'AGOSTINO
City: SIENA
Prov.: SI

Property/Value Type	Value
freeStalls	6
availableBikes	4
brokenBikes	0
availableBikesForecast-15min	5 @2020-02-11T14:00:00+01:00
availableBikesForecast-30min	5 @2020-02-11T14:15:00+01:00
availableBikesForecast-45min	6 @2020-02-11T14:30:00+01:00
availableBikesForecast-60min	6 @2020-02-11T14:45:00+01:00

Latest Update: 2020-02-11T13:59:08.177+01:00

Implementation

Service architecture



- **ML_TRAIN:** responsible for the training of the model for each parking lot.
- The frequency with which each model is trained is daily.
- **ML_PREDICT:** responsible for the forecast for each parking lot.
- Forecasts are made every 15 minutes.

Implementation

Service architecture

ML_TRAIN:

- download the data necessary from the KM4City platform using the Smart City API,
- build the training set
- train the Gradient Boosted Machine previously selected on each park.
- save the 15 obtained models (one for each parking lot) inside the server on which it is running.

ML_PREDICT:

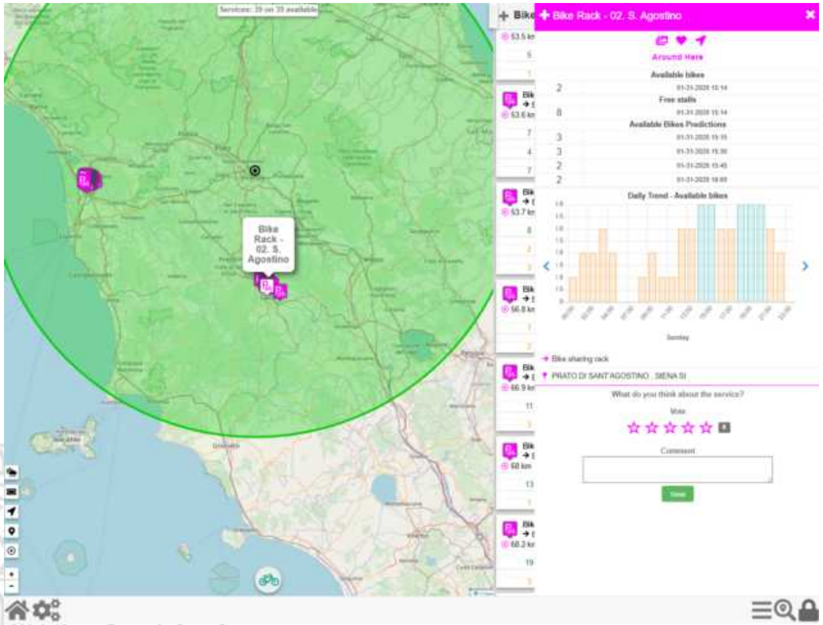
- download the data necessary from the KM4City platform using the Smart City API,
- build the prediction set,
- retrieve the right model for each park,
- perform the forecasts for each park,
- upload the forecast on KM4City platform and made it available on Service Map and for use in third-part application through the Smart City API.

Implementation

Mobile and Web application

The functionality has also been released as an additional module within the **Toscana, dove, cosa** mobile application.

Uses the **Smart City API** to access the entire knowledge base of the platform and allow users to access some of the services offered.



Bike Rack - 02. S. Agostino

Around Here

Available bikes	
2	01-31-2020 15:14
Free stalls	
8	01-31-2020 15:14
Available Bikes Predictions	
3	01-31-2020 15:15
3	01-31-2020 15:30
2	01-31-2020 15:45
2	01-31-2020 16:00

Bike sharing rack

PRATO DI SANT'AGOSTINO . SIENA SI

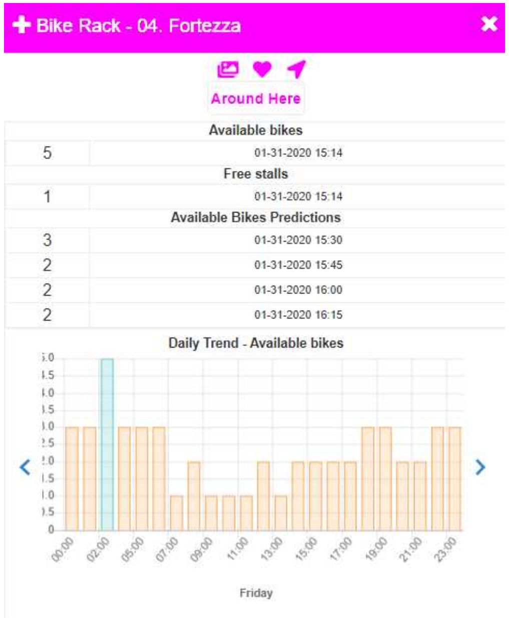
What do you think about the service?

Vote

☆ ☆ ☆ ☆ ☆

Comment

Services: 39 on 39 available
Bike Rack - 02. S. Agostino
Daily Trend - Available bikes
Sunday
Bike
63.5 km
63.6 km
53.7 km
56.8 km
66.9 km
68 km
68.2 km



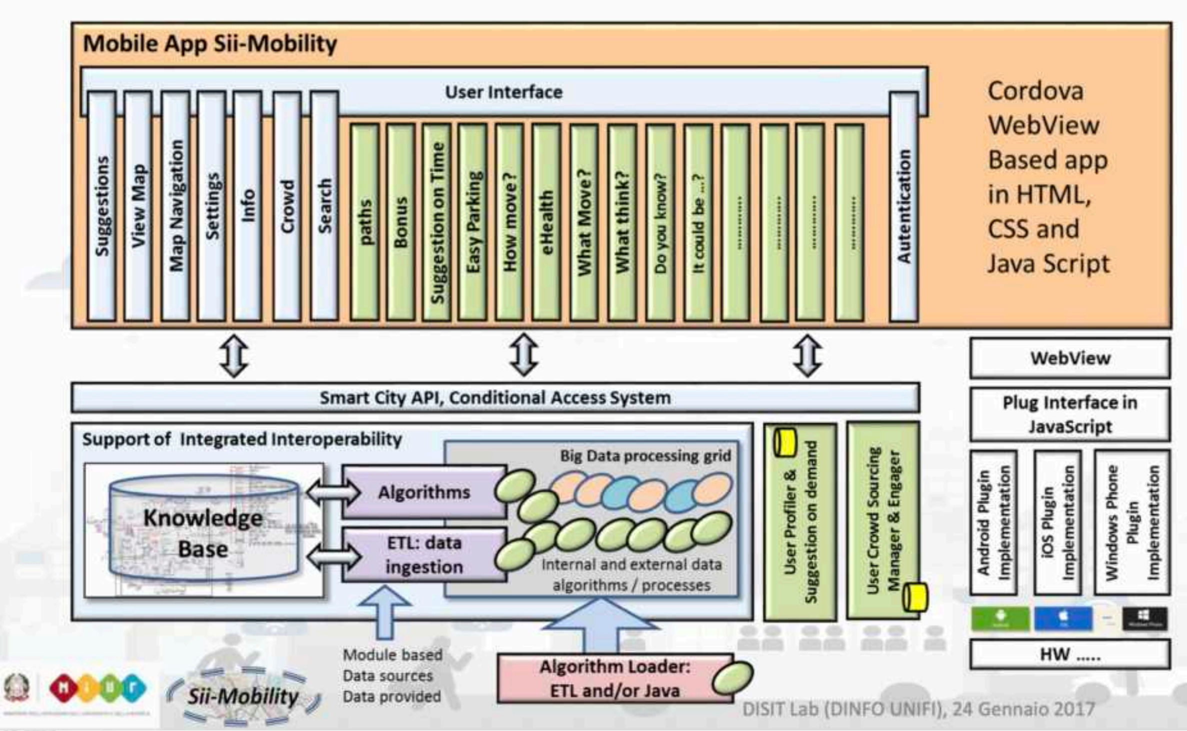
Bike Rack - 04. Fortezza

Around Here

Available bikes	
5	01-31-2020 15:14
Free stalls	
1	01-31-2020 15:14
Available Bikes Predictions	
3	01-31-2020 15:30
2	01-31-2020 15:45
2	01-31-2020 16:00
2	01-31-2020 16:15

Implementation

Mobile application architecture


Mobile App Sii-Mobility
User Interface
Suggestions
View Map
Map Navigation
Settings
Info
Crowd
Search
paths
Bonus
Suggestion on Time
Easy Parking
How move?
eHealth
What Move?
What think?
Do you know?
It could be ...?
....
....
....
....
Autentication
Cordova WebView Based app in HTML, CSS and Java Script
WebView
Smart City API, Conditional Access System
Plug Interface in JavaScript
Support of Integrated Interoperability
Big Data processing grid
Knowledge Base
Algorithms
ETL: data ingestion
Internal and external data algorithms / processes
User Profiler & Suggestion on demand
User Crowd Sourcing Manager & Engager
Android Plugin Implementation
iOS Plugin Implementation
Windows Phone Plugin Implementation
HW .....
Module based Data sources Data provided
Algorithm Loader: ETL and/or Java
Sii-Mobility
DISIT Lab (DINFO UNIFI), 24 Gennaio 2017

- Realized using traditional web technologies and exploiting the **Apache Cordova** set of API.
- Logic – presentation separation using **Mustache JS.**
- Use of open-source javascript libraries such as **OpenLayers** and **OpenLayers Cesium.**

Implementation

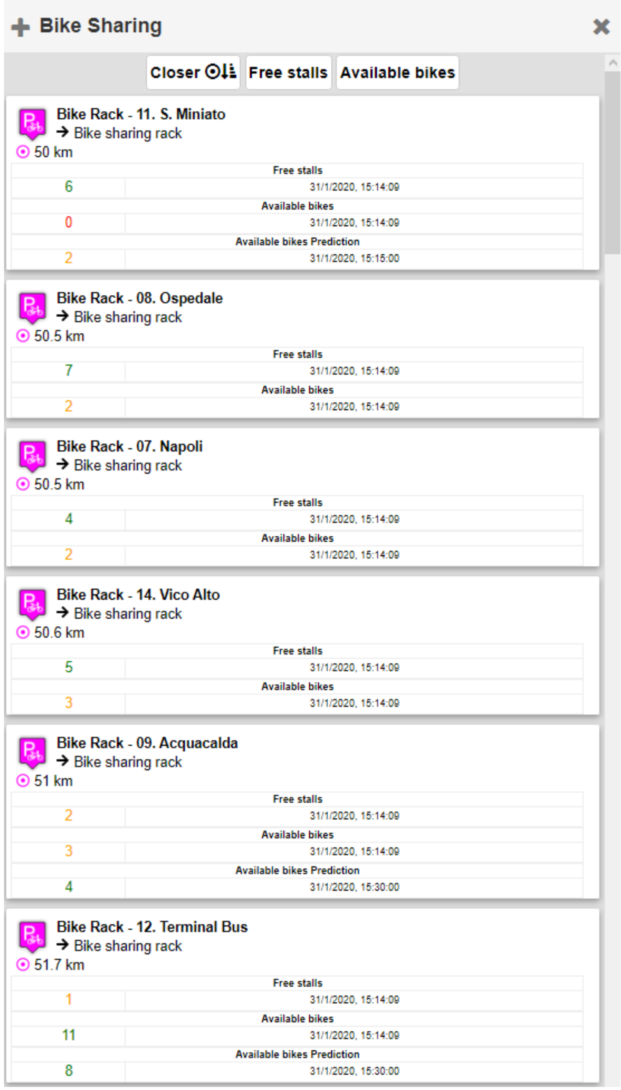
Mobile application architecture

The module created allow to:

- display the **list of parking lots** with a few basic information and
- shows a table where it is possible to view **detailed information** about the status of each bike park

Information displayed are the follows:

- the number of bicycles available,
- the forecasts made,
- the daily average for the current week, calculated over the past 30 days, of the number of bicycles available,
- the number of free slots,
- the number of broken bicycles.



Bike Rack - 11. S. Miniato
→ Bike sharing rack
50 km

Free stalls	
6	31/1/2020, 15:14:09
Available bikes	
0	31/1/2020, 15:14:09
Available bikes Prediction	
2	31/1/2020, 15:15:00

Bike Rack - 08. Ospedale
→ Bike sharing rack
50.5 km

Free stalls	
7	31/1/2020, 15:14:09
Available bikes	
2	31/1/2020, 15:14:09

Bike Rack - 07. Napoli
→ Bike sharing rack
50.5 km

Free stalls	
4	31/1/2020, 15:14:09
Available bikes	
2	31/1/2020, 15:14:09

Bike Rack - 14. Vico Alto
→ Bike sharing rack
50.6 km

Free stalls	
5	31/1/2020, 15:14:09
Available bikes	
3	31/1/2020, 15:14:09

Bike Rack - 09. Acquacalda
→ Bike sharing rack
51 km

Free stalls	
2	31/1/2020, 15:14:09
Available bikes	
3	31/1/2020, 15:14:09
Available bikes Prediction	
4	31/1/2020, 15:30:00

Bike Rack - 12. Terminal Bus
→ Bike sharing rack
51.7 km

Free stalls	
1	31/1/2020, 15:14:09
Available bikes	
11	31/1/2020, 15:14:09
Available bikes Prediction	
8	31/1/2020, 15:30:00

Conclusion

Objective:

- development of a forecast model to predict the number of the available bikes and implementation in a real-time application.

Accomplished results:

- development of a real-time application capable of forecasting over a 24-hour prediction window.
- the service has been working and operating since one month.
- **challenging the state of the art**: only short-term prediction is performed in literature.

THANKS FOR YOUR ATTENTION