Models and Practices in Urban Data Science at Scale

Marco Balduini[1], Marco Brambilla[1], Emanuele Della Valle[1],
Christian Marazzi[1], Tahereh Arabghalizi[1], Behnam Rahdari[1],
Michele Vescovi[2]

[1] *Politecnico di Milano. Dipartimento di Elettronica, Informazione e Bioingegneria. Via Ponzio, 34/5. I-20133 Milano, Italy.*
{firstname.lastname}@polimi.it
[2] *SKIL Joint Open Lab – Telecom Italia. Via Sommarive 9, Povo, Trento, Italy*
michele.vescovi@telecomitalia.it

Abstract

Cities can be observed through a broad set of sensing technologies, spanning from physical sensors in the streets, to socio-economic reports, to other kinds of sources that are able to represent the behaviour of the citizens and visitors, such as mobile phone records, social media posts, and other digital traces.

In this paper, we propose a conceptual framework for putting at use this variety of Big Data sources, with a unified approach that applies spatial and temporal analysis over heterogeneous streams of data. We define spatial analysis based on conceptual grids (made of cells) over the city space, and then we study: the time series of signals both at grid and cell level; the correlation across signals and across cells; the prediction of city dynamics based on multiple signals; and the identifications of anomalies based on the difference between the observed dynamics and their prediction.

To implement this model we propose a general architectural framework that uses Big Data technologies (such as HDFS, YARN, HIVE, PIG, Cascalog, Spark, Spark SQL, Spark Streaming and SparkR) and can be deployed in different configurations based on different needs. By taking an inherent data science approach to the problem we are able to address at scale: technical problems such as heterogeneous time and space granularity of the data, as well as appropriate interpretation of the results through tools that enable intuitive and immediate visual perception of emerging patterns and dynamics.

We demonstrate feasibility, generality and effectiveness of our Urban Data Science at scale approach through multiple use cases and examples taken from real-world requirements collected in various cities and accounting for diverse business and city needs.

Keywords: Smart City, Big Data, Social Media, Urban Computing, Urban Sensing, Behavioural Analytics.

1. Introduction

In the last years, cities became a sources of a variety of Big Data. Since the mid 2000's it was possible to observe the digital traces left by citizens and visitors using mobile phone records and social media posts [1], but only recently cities have been equipped with a broad set of sensing technologies that allow observing the physical behavior of citizens and visitors.

Thanks to Big Data technologies, we are now able to capture a sharper and sharper picture of our cities and to track changes in their dynamics with seconds of delay. And thanks to Data Science methods, we are now able to analyze at scale both the digital reflections of our cities and their physical everyday businesses.

1.1. Context

Exploring cities of the 21^{st} century offers a great opportunity to understand the ever evolving modern society. In fact, cities are increasingly attracting new people, with half of the world's inhabitants living in urban areas [2]. Furthermore, cities are not only physical centers but also virtual hubs, where individuals and communities interact and exchange messages through social media[3]. As a consequence of this dense network of interactions, a great amount of data, the so called Big Data [4], can be tracked.

On the one hand, data created by the single identities of the city, i.e. inhabitants, are available; on the other hand, we need to capture the *big picture* by aggregating them. This situation allows us to observe both the activities at a micro-societal level and to draw the main features that characterize the city at a macro-societal level. Therefore, Big Data can be regarded as a lens to understand cities, or, using the words by De Rosnay [5], Big Data is a *macroscope* – i.e., a tool for capturing complex systems – applied to the urban environment.

In this context, data visualization is a recognized method to directly interact with data that allows to absorb more information easily, discover patterns between business and operational activities, and identify emerging trends faster. As such, it allows increasing the value detection and intake based on data insights, and therefore it must be considered since the early phases of the design, thus improving the understanding of city events and phenomena.

1.2. Existing works

Urban computing [6] has clearly shown the huge opportunity for Big Data research to exploit mobile phones data to get insights into urban dynamics and human activities. This type of data was used to estimate the density of crowd and vehicles in urban regions [7, 8, 9, 10, 11] and to predict returning frequencies to points of interest[12, 13]. When merged with other kind of information, mobile phones data can reveal interesting insights for city dynamics and urban monitoring [14, 15, 16, 17]

Although mobile phones data is a priceless source to gather underlying patterns of cities and their citizens, they hide some limitations since they can not

reveal any information about people interests and thoughts. A parallel investigation of social media steams has recently carried out by the research community [18, 19, 20, 21, 22, 23, 24].

However, only few research groups tried to tame the variety present in those Big Data sources [25, 26]. This is exactly were our research began in 2013. We have already published few specific papers on this topic [27, 28], but this is the time we tell the full story in a comprehensive way and providing extensive real-world evidence on the validity of the approach we propose.

1.3. Objectives

In this paper, we aim at defining **a high-level model, a method, and a set of practices** that allow us to represent the urban ecosystem in terms of analysis and aggregations that improve decision making processes and deliver added value to city stakeholders, spanning citizens, tourists and visitors, public officials, and businesses.

In order to obtain this: (1) we define a high-level semantic model of the domain and a logical architecture that fits it; (2) we implement both of them in a set of technical embodiments using Big Data technologies; and, then, (3) we put them at work on concrete cases of urban data monitoring and computing using Data Science method at scale.

Thanks to this three–step–process, we obtain the following benefits:

- We have available a common set of conceptual assets (models) and technical assets (implementations) that can be reused across different scenarios.
- We are able to integrate diverse city-wide data sources, with different types of content, format, and time / space granularity.
- We enable different types of analysis and processing, spanning description, prediction and anomaly detection.
- We allow accurate and intuitive visualization and navigation of the results, which are considered since the early phases of the design, thus improving the understanding of city events and phenomena.

1.4. Research Problems

This paper will, therefore, address the **research problems** of:

RP1. defining a conceptual model as well as a logical architecture;

RP2. defining appropriate technical instantiations of the above models;

RP3. assessing feasibility and effectiveness of those models and practices by deploying them in real world scenarios.

The above research questions will be addressed taking into account the following **requirements**:

Req1. Enable aggregation, analysis and prediction over city-wide data streams along *space* and *time* axes;

Req2. Enable integration of heterogeneous data sources, considering diverse content types and (temporal and spatial) granularity;

Req3. Support intuitive and explanatory visualization and exploration of results.

1.5. Structure of the paper

The remainder of the paper is organized as follows. Section 2 presents the conceptual model at the core of our approach. Section 3 covers the different methods we employ in our experiences. Section 4 describes the architectural infrastructure that implements the proposed model. Section 5 extensively presents our experiences in applying the conceptual model presented in Section 2, the methods illustrated in Section 3 an the implementations described in Section 5 to perform Urban Data Science at scale. Section 6 presents the related work. Finally, Section 7 concludes.

2. Conceptual Framework

In this Section, in order to address the Research Problem **RP1**, we present a conceptual model able to tame the variety of Big Data sources present in smart cities.

One of the main assumptions of any smartcity approach is to work upon a layer of data collected from the city itself, describing its dynamics. The city evolution can span multiple layers, from architecture to urban design, from population composition and migrations, to citizen behaviour and interests. Each of these layers have a different dynamics and speed of change; therefore, it should be monitored collecting different data sources and using multiple analysis techniques. For instance, people moving through a city to attend fashion shows spread in different districts[1] may leave little physical traces, but they may leave large amounts of digital footprints that can be analysed to understand the dynamics of such a fashion-addicted crowd.

2.1. Intuition

To this purpose, in our previous works [29], we proposed a conceptual framework named FraPPE out of its four main concepts: Frame, Pixel, Place and Event. FraPPE enables a high level view of the detection, understanding and interpretation of city data. It uses a digital image processing metaphor (see Figure 2) to track three main dimensions of analysis: space, time, and content.

FraPPE assumes that the real world can be described as a bi-dimensional space, where *events* happen in *places* over time. For instance, a user making a

[1] See `http://fashionweekonline.com/`

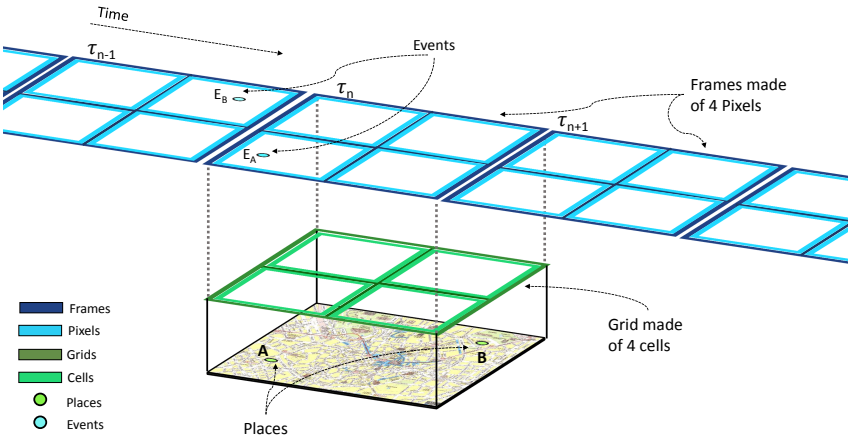


Figure 1: A high-level view of FraPPE including 3 Frames made of 4 Pixels containing the Places where the Events happen.

check-in on geo-located social network generates an event in a place. A taxi ride generates a sequence of two events (a pick-up and a drop-off) in two distinct places. A garbage collector truck generates a sequence of events around the city in different points in time, once for each trash bin it cleans up.

FraPPE proposes to capture the digital footprints of what happens in the real-world as a sequence of *frames*. A *grid* sits between the physical world and the frames of the film. It decomposes the physical world in *cells*, four in the example. Each frame is, therefore, decomposed accordingly in *pixels*. The cells contains the *place*s where *event*s happens on different time. The events are captured in the pixels of the frame at the time they happen. For instance, if A is the place where a taxi picks up a person at time τ_{n-1}, an event can be captured in the upper-left pixel of the frame sampled at τ_{n-1}.

As a *frame* is the the time-varying representation of a *grid*, a sequence of frames composes the film of the evolution of a physical portion of the world over time. In Figure 2, the past frames, which refer to the time interval τ_{n-1}, τ_{n-2},..., τ_{n-m} are on the right, the current frame (for the time interval τ_n) is in the center, and the next empty frames (ready to capture the digital footprints of what happens in the real-world at time τ_{n+1}, τ_{n+2}, etc.) are on the left. Going back to the previous example on a taxi-ride, the current frame captures in the lower-right pixel the drop-off event that occurs in B at τ_n.

Notably, multiple frames can be simultaneously captured at different rate in order to track the changes of the real-world at different time granularity. For instance, let us assume we want to know if there is an extraordinary usage of the parking lots in a given part of a city. A frame captured every hour can be used to calculate the average number μ and the standard deviation σ of occupied parking lots in such a given part of a city, while a frame captured every 5

minutes can be used to measure the current number of parking lots. Assuming that the distribution of the number of parching lots follows a Gaussian process, we can determine if there is an extraordinary usage of the parking lots by *i*) computing the z-score of x – i.e., if x is the current number of occupied parking lots, the z-score of x is $(x - \mu)/\sigma$ – and *ii*) checking if it is greater than 2.

2.2. Formalising the Model

We now proceed with the extension of the original FraPPE and with its formalization as a UML model. In particular, Figure 2 depicts a UML representation of the extended version of the FraPPE model. We elaborated this extended version specifically for this paper. As highlighted by the different colors, this extended version of FraPPE is organised in four different interconnected parts: the green one is related to space, the yellow one to time, the red one to content and the blue one to provenance.

The space-related part includes the classes Grid, Cell and Place. A Grid can cover a portion of the world and it contains a variable number of Cells. A Cell represents a restricted portion of the space and it contains Places. A Place is a point of interest with unique geographical coordinates. The relationship contains between the three spatial objects is transitive. If a grid contains a cell that contains a place, then the place is in the grid. The within relationship is the inverse of contains and allows the backward navigation of the chain.

The classes Frame, Pixel and Event describe the temporal dimension of the framework. The time-varying object are connected by the same properties as the spatial ones. The transitive contains property allows walking the connection chain from the frame to the event, contrariwise the within property guides the application from an event to a frame through a pixel.

Notably, the time-varying element has an exclusive relationship with its spatial counterpart, but a geographical object can be connected to multiple time-varying object. For instance, being a pixel the time-varying representation of a cell, while Pixel refers to a single cell, the cell can be referred by multiple pixels over time.

In the proposed conceptual model, the content can be associated to the time-varying classes and carries information that represents a measure of intensity of the tracked phenomena. At event level the content can be Original or Augmented. The original content represents a simple measure or description of a phenomenon, while any enrichment of an original content produce an augmented content. The content related to Pixel or Frame is Synthetic and it is derived by processing event-related contents.

The last part of FraPPE conceptual framework is related to provenance. FraPPE distinguishes between two types of frames: the CapturedFrames and the SyntheticFrames. The former one contains a pixel for every considered cell and represents a non-filtered collection of all the digital footprints found in the real-world. Different CapturedFrames can refer to different images of the observed phenomena at the same samplingTime. The latter one is generated

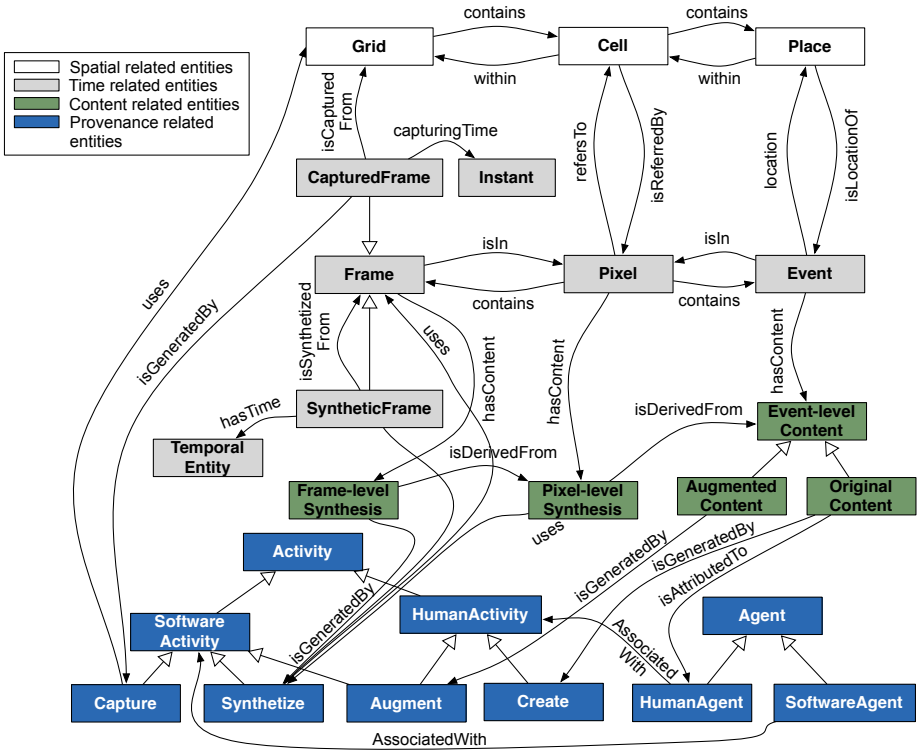


Figure 2: The extended version of the FraPPE model, represented as a colored UML diagram highlighting space-, time-, content-, and provenance- related concepts.

by a SYNTHETIZE activity on two or more frames. The SYNTHETIZE activity can be a filter, or an aggregation of values of pixels across different frames. Other kinds of ACTIVITIES cover creation, capturing, and augmentation of content. Each activity is performed by a HUMANAGENT or by a SOFTWAREAGENT. For a complete algebra of further possible actions, you can refer to [30].

FraPPE is also formalized using the OWL 2 Web Ontology Language[2] and reusing GeoSparql [31] as geographical data model, the Time [32] and Event ontologies [33]) as time/event vocabularies, and PROV-O [34] as provenance ontology. The resulting vocabulary is published in Linked Open Vocabularies[3] [35].

[2] https://www.w3.org/TR/owl2-overview/
[3] http://lov.okfn.org/dataset/lov/vocabs/frappe

3. High Level Methods

In our *Urban Data Science at Scale* approach, we consider three dimensions of analysis over city-wide Big Data: **space**, **time**, and **content** analysis. In this section we summarize the methods we use for defining each analysis dimension.

3.1. Space Analysis

The first dimension of interest in city analysis is space. Therefore, we focus on analyzing events, people presence and flow, content and opinion sharing, or any other type of phenomena (like electrical consumption, traffic, economical value) with respect to the spatial distribution and spreading, also considering its dynamics in time. The spatial dimension is the most complex to deal with, in terms of coping with heterogeneity of the measured variables. Indeed, in smartcity context, the data sources may vary a lot: some information may refer to specific geographical points (geo-coordinates), some others may refer to venues or locations (restaurants or other public or private spaces), while others can provide information referring to broad areas, possibly with different size and shape. Any analysis considering two or more different data sources need to keep this into account. This is one of the main reasons why in Section 2 we defined the concepts of Grid and Cell. The second reason for this is the contribution this gives to the understandability and navigability of geographical-based content.

In practice, it's important to clarify that these concepts can be instantiated in multiple ways: we may define different types of grids and cells, based on the specific data sets and on the analysis needs. We identify three main categories of grids:

- **Regular squared grid**: a regular grid dividing the physical space in cells that are uniform for shape, size, and positioning. For instance, in many of our experience around the city of Milan, we defined a grid of 100 x 100 cells, each cell having a size of 250 x 250 meters.
- **Irregular grid with official business-driven meaning**: a grid of cells that are different in shape, size and orientation based on some official definition (e.g., the boroughs or zones of a city) or based on some business specification (e.g., the commercial areas of the city). An example of this can be the official city districts defined by the municipality or the areas where a large event is located.
- **Irregular grid with data-driven definition**: a grid of cells defined bottom-up based on the domain data available or on partial analysis and aggregations already performed on them. Some examples include the areas served by different electricity substations, the mobile phone cell coverage[4],

[4]In cellular networks, the telecommunication service is provided by a altitude of base stations distributed across the served area. Each base station services a limited portion of space, called âĂIJcell coverage area, which is irregular and based on technical infrastructure and geo-physical features of the terrain and buildingsâĂİ.

or the areas where mobile phone presence can be clustered with sufficient precision with respect to the location of the antennas.

Another important feature of the grid is the coverage of the area of interests. We can define grids with **total coverage** or **partial coverage** of the area. Typically, regular grids tend to feature total coverage, while irregular ones, especially when defined starting from business requirements, may offer only a partial coverage of the area.

Over the above defined concepts of grid and cells, we identify the following types of relevant analysis categories:

1. **Dispersion**: studying the spatial distribution of locations of events or concepts, in particular with respect to the deviation from purely random configuration. This can be achieved with measures such as the Gini coefficient, or the weighted cell coverage.
2. **Distance and relation to places**: studying the spatial relation of events with respect to a set of given locations (e.g., stores or venues for specific happenings such as fashion shows). This is covered by simple measures such as the average Euclidean distance between event and location, or the average Manhattan distance over the grid cells.
3. **Correlation**: studying the relevant correlations between different signals along the space dimension, i.e., within and across cell.
4. **Prediction**: defining predictive analytics within or across cell.

3.2. Time Analysis

Temporal analysis focuses on the study of the evolution and spreading of signals captured by pixels, which refers to cells, over *time* in different frames, e.g. measuring how fast information about an event propagates on geo-located social media. The goals of temporal analysis can be diverse. We identify the following types of relevant analysis categories:

1. **Description**: consisting in defining the signal captured by pixel-level contents as time series.
2. **Correlation**: studying the temporal correlation between different time series and infer common behaviours and dynamics of cities.
3. **Prediction**: allowing generating temporal prediction over observed or correlated phenomena.
4. **Anomaly detection**: identifying discrepancies between expected temporal behaviours and actual happenings.
5. **Causality**: determining possible causality relations between different events.

3.3. Combined Time and Space Analysis

Given the basic space and time analysis aspects described below, the subsequent level of interest is the combined analysis along both directions together. The definition of the concepts of Frame and Pixel in our conceptual model specifically aim at this type of analysis. We combine techniques described in the previous sections for running analysis across time and space.

For instance, in case of anomaly detection we can extend the method discussed by defining the standard behaviour and anomaly index by time slot and by city pixel instead of time slot only.

Furthermore, one can define time series of values that are aggregated or calculated on geographical basis. For instance, we can define the time series of the values of the Gini Index or of the average distance of events from a set of given venues, and then analyze them along the temporal axis.

3.4. Content Analysis

As mentioned in the conceptual model section, content can be associated to an event and thus indirectly to the time and space of the happening, and carries information that represent a measure of intensity of a tracked event. Thus, we aim to analyze synthetic content to extract contextual and behaviour knowledge about what and how users share about an event. Therefore, according to the FraPPE framework, at time τ_n an event E is held in the place P, in the cell C contained in the grid G; C is related to the pixel P and G is captured by the frame F. For instance, in case of social media sources, the content consists of the social message M that is related to the event E (the posting of the content), and is described by a set of properties, including text, photos, and metadata.

We introduce two different approaches for content analysis. In the first approach, the ORIGINAL content is made of text and hashtags. This content can be analyzed and used for profiling social media users who are engaged in the event. In the second approach, an AUGMENTED content can be created by using concept and feature extraction techniques from the shared photos for the purpose of more complex analysis about the event and its attendees.

The analyzed content could consist of different media types including text, image, video, etc. that also contain low-level information about the event like location, time, related social users and so forth. From such content we can extract **low-level features** such as color schema for images or n-gram distribution for text. For instance, we can use the main color schema in photos related to an event to verify the correctness of the estimated location of that event. Furthermore, we can also extract **high-level features** like number of people and their demographics in a photo, list of existing concepts that are represented in a photo or a video (using deep learning techniques) or semantic entities from text using ontology-based matching.

3.5. Real-time vs. periodic vs. a-posteriori

One important aspect in big data analysis problems is *the time at which the analysis is run and the results are produced.* This is a crucial point that combines business requirements and technical constraints associated to the amount of information to process and the computational power availability.

In the experiments presented in Section 5 as well as in the architecture presented in Section 4, we distinguish among three different methods for different purposes and stakeholders: (i) **real-time**, i.e., continuous collection, augmentation and synthesis of data followed by immediate analysis and result display; (ii) **periodic**, i.e. continuous collection, augmentation and synthesis of

data followed by periodic analysis, with arbitrary, but known, temporal lag before result rendering; and (iii) **a-posteriori**, i.e., ad-hoc analyses performed after the completion of the phenomenon.

In general, real-time analysis was preferred in public space installations with public, animated visualization of results, where it is necessary to engage the audience in front of the installation itself (e.g., letting them see the photos they posted on Instagram). Periodic analysis is useful to compare a recent past against a historical collection of data. We used it to allow a user to visually detect patterns, e.g., a security officer that needs to track the movements of large crowds entering or exiting a large venue. The a-posteriori is useful for all those stakeholders that needs to perform analysis without knowing in advance the frequency. This can be useful to plan services for the urban environment or scheduling events, campaigns and commercial offers that suit the need of citizens.

4. Principles, Logical Architecture and Implementations

In this section, we present three guiding principles that we elicit from out implementation experience. We elaborate on how to model abstract operations that manipulate information in accordance with those principles. We illustrate our architecture at a logical level and we shortly discuss how the requirements on vertical and horizontal scalability condition the implementation choices.

4.1. Principles

In our case studies, we noticed that we always deal with information that changes (*Velocity*). Data can come from different sources that vary in format (*Variety*) and size (*Volume*), but it always flows continuously. Even what we normally call *static data*, e.g., a city street grid, is not immutable over time, it slowly evolves.

Based on this observation, we can state our first principle: **(P1)** *everything is a data stream.* According to this principle, a system built on our architecture must indifferently ingest data with different velocities from any sources and of any size. All the incoming information is modeled as a generic data stream S, i.e., a potentially unbounded sequence of timestamped data items (d_i, t_i):

$$S = (d_1, t_1), (d_2, t_2), \ldots, (d_n, t_n), \ldots,$$

where d_i can be any data type in any format, $t_i \in \mathbb{N}$ is the associated time instant and, for each i, it holds $t_i \leq t_{i+1}$ [36]. For instance, the movements of a car is a *fast* data stream where d_i records the identity, the positions and the speed of the car and the distance between two subsequent observations can be seconds. On the other side, the evolution of a city road is a *slowly evolving* data stream, where for instance d_i may record the addition of a bike lane to a road; in this case, the distance between two subsequent observations can be days or months.

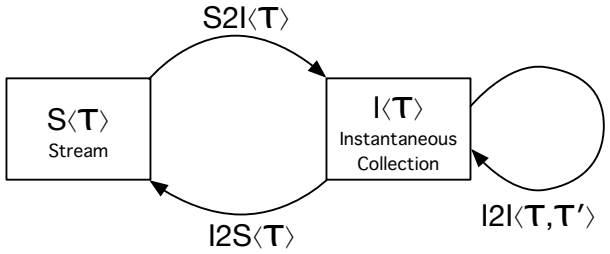


Figure 3: Overview of CQL class of operators

The continuous nature of data streams requires a system built on our architecture to implement our second principle: **(P2)** *Continuous Ingestion*. The data input is continuously captured by the system and, once arrived, it is marked with an increasing timestamp. Notably, some data sources may natively include their own timestamping too (which we call application timestamp).

Our last principle is motivated by the variety of the input data, and we define it as **(P3)** *Lazy transformation*: a system inspired by our architecture operates on the data in its original format as long as it can, and it transforms it only if really needed. Indeed, operations like projections, filters or aggregations can operate on generic data without requiring to cast all data in a single format (e.g., the relational one). Therefore, for those operations we can delay transformations. Contrariwise, a join operation on data of different data format (e.g., a CVS table and a JSON tree), normally, first requires to cast data in a common format (e.g., the relational one) and then perform the join. Even if most of the operations do not require a preventive data transformation, the results are often in a different format if compared to the input. This implies that inside the operators the data are transformed and typed, e.g. grouping and counting any data type will always generate a relational table with two columns: the identifier of the group and the result of the count.

Generic Functions [37] represents the natural abstraction to model at high level the operations that manipulate information in accordance with our three principles. The data items d_i flowing on a data stream S can be modeled in terms of *types to-be-specified-later*; it can be, indifferently, an instance of a tree in a JSON document or in an XML document, a set of tuples in CSV or in parquet format, or a graph in RDF.

Figure 3 shows the three classes of operators that we propose. They are inspired by the CQL stream processing model [38]. Let us denote with $\mathcal{T}$ a generic type to-be-specified-later, with $\mathrm{S}\langle\mathcal{T}\rangle$ a generic data stream and with $\mathrm{I}\langle\mathcal{T}\rangle$ a collection of instantaneous generic data items (e.g., an a-temporal table, a document, or a graph that is normally manipulated by relational, document or graph databases). Three classes of operators allow to move from generic data streams to instantaneous generic collection an vice versa. The operators in the class stream-to-instantaneous $\mathrm{S2I}\langle\mathcal{T}\rangle$ transform a portion of a potentially

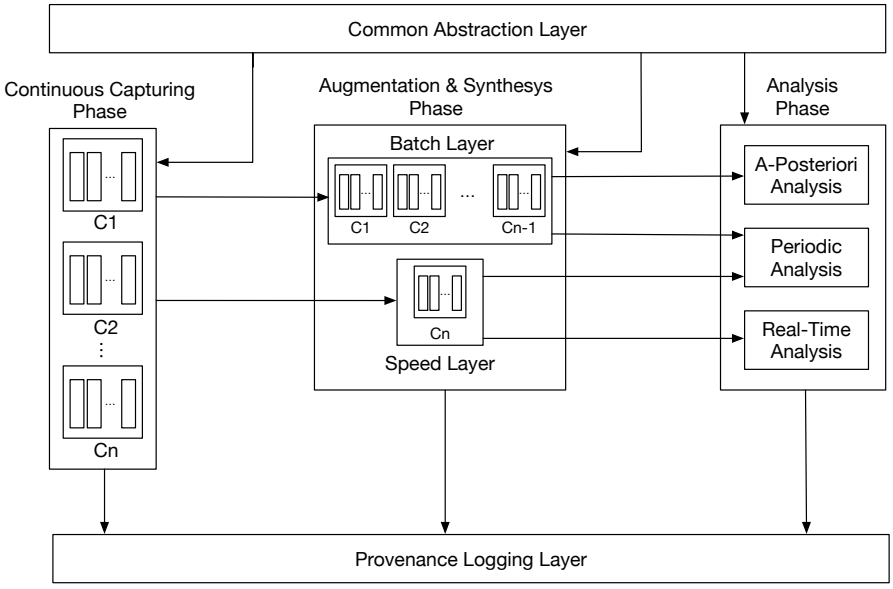


Figure 4: Overview of the general architecture of the system

infinite generic data stream $\mathrm{S}\langle\mathcal{T}\rangle$ into a finite collection of instantaneous generic data items $\mathrm{I}\langle\mathcal{T}\rangle$, e.g. Window Operator [39] can extract a part of a stream based on time or number of items.

The operators in the class instantaneous-to-instantaneous $\mathrm{I2I}\langle\mathcal{T},\mathcal{T}'\rangle$ transform $\mathrm{I}\langle\mathcal{T}\rangle$ into another $\mathrm{I}'\langle\mathcal{T}'\rangle$. Physical operators of this class deal with the different data format, e.g., xQuery operators process XML, SQL operators process relational table, SPARQL operators process RDF graph, etc. Notably, $\mathcal{T}$ and $\mathcal{T}'$can be different types, but they can also be of the same type. For instance, a filter on a table, on a tree or on a graph extract tuples, sub-trees or sub-graphs maintaining the original data type. Contrariwise, as we noticed above, a count aggregation transform the original data type into a table.

The operators in the class instantaneous-to-stream $\mathrm{I2S}\langle\mathcal{T}\rangle$ act on $\mathrm{I}\langle\mathcal{T}\rangle$ to create a new $\mathrm{S}'\langle\mathcal{T}\rangle$. Those operators are used to emit as a new flow of data the results over time of $\mathrm{I2I}\langle\mathcal{T},\mathcal{T}'\rangle$ operators. The new stream $\mathrm{S}'\langle\mathcal{T}\rangle$ can contain all the items produced by the $\mathrm{I2I}\langle\mathcal{T},\mathcal{T}'\rangle$ (namely, R-stream), only the new items in $\mathrm{I}\langle\mathcal{T}\rangle$ that were not in the previous $\mathrm{I}\langle\mathcal{T}\rangle$ (namely, I-stream) or only the items that were in the previous $\mathrm{I}\langle\mathcal{T}\rangle$, but not in the new $\mathrm{I}\langle\mathcal{T}\rangle$ (namely, D-stream).

4.2. Logical Architecture

Figure 4 shows our logical architecture. Information enters from the left and exits to the right. A system, built in accordance with our logical architecture, operates on the data in three phases. It continuously captures data over time (phase 1). It enriches, manipulates and transforms captured data (phase 2) in order to synthesize the data that it analyses (phase 3) to offer results to its

users. This architecture is *variety proof*, i.e., it can accept data in any format, and *velocity first*, i.e., it can handle input data streams regardless the incoming rate.

During the Continuous Capturing Phase the data, which continuously flows in, is just marked with a timestamp, i.e., following the *Lazy Transformation* principles, it is captured in its original form independently from its complexity. We recommend to treat Volume as orthogonal to Variety and Velocity by requiring a system that implements this phase to be partition tolerant by choosing the best partition strategy (see Section 4.3).

For the Augmentation and Synthesis Phase, we recommend to use a lambda architecture[5] with a Batch Layer and a Speed Layer. The former operates over all the data captured in the previous phase, i.e. C_1, C_2, C_{n-1}, while the latter takes in account only the most recently captured information, i.e. C_n.

The final Analysis phase can exploit, based on the information need of the user, indifferently various part of the upstream architecture. The Batch Layer can be used alone for periodic and post-hoc analysis, or in support of the Speed Layer for analysis that needs to compare the most recent data with the historical one. Nevertheless, the speed layer, can be independently used to perform instantaneous analysis. Let us make this more clear with an example. A taxi company can exploit the Batch Layer, to synthesize statistics about the cost and the duration of all the rides captured so far in a city. An a-posteriori analysis of those statistics can determine a complete origin-destination matrix for the taxi rides, i.e., a distribution of durations and prices of all possible routes from any point to any other point in the city. At the same time, the taxi company can exploit the Speed Layer to determine the current most profitable routes using with the latest incoming data. The comparison between the latest price of the rides (computed in the Speed Layer) with the information in the origin-destination matrix (computed in the Batch Layer), can be useful to foil a fraud. Two more layers compose the proposed architecture: The Common Abstraction layer and the Provenance Logging layer. The former logically contains the abstraction used to model or manipulate data. For example, the user should be able declare how to augment captured data with identifiers that refer to FraPPE abstractions like Cell or Frame. Moreover, the user should be able to model its computation flows using the class of operators discuss above. The Provenance Logging layer contains all the artifact useful to document data lineage and to log the system actions in accordance with concepts in the Provenance fragment of FraPPE . As explained in Section 5, this information is useful during the interpretation an analysis.

4.3. Implementation

In our experience, the most important characteristic of an implementation of our architecture is its *cost effectiveness*. In the following, we discuss three

[5] `https://www.manning.com/books/big-data`

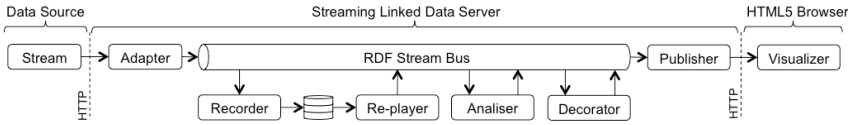


Figure 5: Overview of Natron

alternative implementations (i, ii and iii) where the nature of the data, in particular the *Volume*, and the scalability requirement of the system shape the specific implementation of the presented logical architecture.

An ad-hoc implementation (i) results suitable in all the cases where the amount of data is small and the cost of a complex infrastructure is unaffordable. In this situation, there is no need for scalability and the final artifact can be developed in any language or using any framework, e.g. Python or Java. If the amount of data grows, a scalability requirement arises. An ad-hoc solution results hard to be cost effective, if compared to a more generic and reusable implementation. Conscious that distribution and parallelization does not pay at all scales [40], we developed a vertically scalable (ii) single thread system, namely Natron , that results cost effective for a medium amount of data: it is designed to have very low entry cost for low volumes (around 0,03 € per MB per minute[6]). Moreover, it is pluggable and extensible, in order to be used in different situations. Once the data amount grow above tens of MB per minute, in order to remaining cost effective, it is better to switch to implementations based on big data technologies (which grow linearly in the amount of data) and meet the need of (iii) horizontal scalability.

In the next three sections, we present Natron and two big data implementations of our architecture based on Hive[7] and Spark[8].

4.3.1. Vertically Scalable Implementation - Natron

A single thread implementation, such as Natron , represents the best way to deal with continuously flowing data characterized by medium Volume, high Variety and very high Velocity. Natron is implemented following the principles discussed in previous sections, i.e., it continuously ingests streaming data represented as a time-stamped data items that are typed, only when needed. The typing is declared as an annotation to the captured information.

Figure 5 depicts on overview of the Natron internals. The Receivers (I2S$\langle\mathcal{T}\rangle$ operators based on the data source) allow ingesting external data streams, and push the data on the Generic Stream Bus. Data items remain in their original form, only the ingestion time is added, as recommended by Principle P3; we

[6]To determine this price we run experiments on Azure using F4S_v2. For more information visit: `https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes-compute`

[7]`https://hive.apache.org`

[8]`https://spark.apache.org`

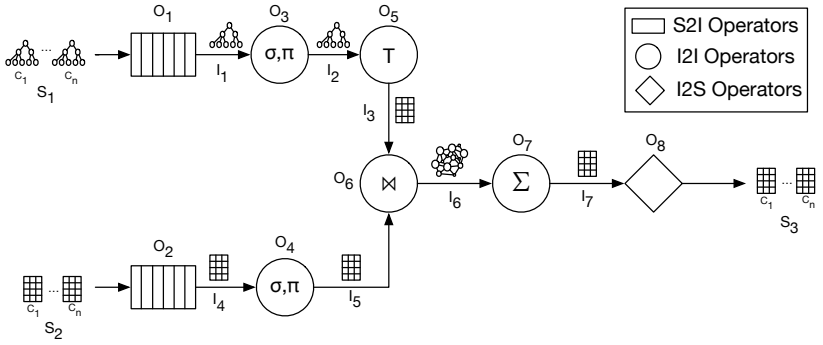


Figure 6: Example of Natron pipeline

postpone the transformation as long as possible in the process. The Processors, e.g. an Information Flow Processor as Esper[9], listen to one or more streams $S\langle\mathcal{T}\rangle$, compute different operations and produce a new stream $S'\langle\mathcal{T}'\rangle$. Translators allow Natron producing in output the streams in multiple formats and represent the implementation of $S2I\langle\mathcal{T}\rangle$ operators. In Natron , the window operator can be implemented in two different ways: either using the ingestion timestamp added during the Continuous Capturing Phase, or using an application timestamp, e.g. a time mark added during the Augmentation & Synthesis Phase referring to the notion of Frame in the Common Abstraction Layer. Figure 6 presents a simple pipeline implemented in Natron to demonstrate the capability of the framework to deal with generic data coming from multiple sources and to perform transformations at various level. The data flowing into the system in the form of two streams of data, a stream of trees, i.e. S_1, a stream of relational observations, i.e. S_2.

```
select * from Event.win:time_batch(40 sec)
output every 20 sec
```

Listing 1: EPL query to window generic stream

Both of them are windowed through a simple EPL[10] query presented in Listing 1. The window operator, i.e. a S2I operator implement in O_1 and O_2, exploits using Esper[11] and transform the stream of generic data items into a collection of data items maintaining the original data format. The downstream operators, namely O_3 and O_4 I2I operators, perform projection and filter on generic data and finally typify the data items to prepare the Join operation.

[9] http://www.espertech.com/

[10] https://docs.oracle.com/cd/E13157_01/wlevs/docs30/epl_guide/overview.html

[11] http://www.espertech.com/esper/

```
{
  "data": [{
     "id": "1",
      "lat": 45.806171,
      "lon": 9.086754,
      "count": 20
    }
  ]
}
```

Listing 2: Wxample of I_2 data

Table 1: O_5 operator template and query for the mapping

Triple Template	**Source Query**
[S] :count [O]	for $data in collection("data") let $S := $data.id let $O := $data.count return $S : $O

Listing 2 presents an example of data contained in the I_2 collection. Table 1 depicts the mapping (*Triple Template* column) and the query (*Source Query* column) exploited by the operator O_5 to transform the trees into relational table. The value of S and O are extracted from the query on the trees. The query in our implementation is written in JSONiq[12], a query and processing language designed for JSON. The results of the mapping in the example is a triple *1 :count 20* The operator O_5 transform the data in order to ease the join operation performed by the O_6 operator.

```
SELECT ?s ?o
WHERE {?s :count ?o}
```

Listing 3: SPARQL query to query the graph data from the join operation

The join produces data items in the form of complex graphs that can be queried by the operator O_7 using the Sparql[13] query presented in Listing 3. The final operator O_8 transform the Instantaneous data items into a new stream of relational table, i.e. S_3.

4.3.2. Horizontally Scalable Implementations

When scaling to large volume is required, a single thread implementation is at risk of loosing cost effectiveness because, even if the entry cost is much lower than a Big Data implementation, its costs grows exponentially in the size of the data. Therefore, an horizontally scalable solution, using big data technology, represents a good choice. In our work, we employed two different solutions respectively based on Spark [41] and Hive [42].

[12] http://www.jsoniq.org

[13] https://www.w3.org/TR/sparql11-overview/

Spark. We propose a system developed using Spark Streaming[14], an extension of Spark that enables the development of stream ready application. Spark Streaming implements I2S$\langle\mathcal{T}\rangle$ operator, i.e. it allows continuous data ingestion from many different sources, and classic S2I$\langle\mathcal{T}\rangle$ operators, e.g. window. Moreover, its Dataframe APIs allow implementing the complete stack of layers of the proposed architecture.

```
val itemsCounts = inputStream.groupBy(
    window($"ts", "40 seconds", "20 seconds"),
    $"Agg(Count)"
  ).count()
```

Listing 4: Example of Window operator in Spark

The Spark Streaming based system is able to implement the Window operator by exploiting the ingestion time, added to the data once entering the system, i.e. during the Continuous Capturing phase. Listing 4 presents the code to compute the aggregation *Agg(Count)*, representing the amount of the data items that entered the system in the last 40 seconds, using a window that slides every 20 seconds.

Hive. Hive is a Big Data warehouse solution and is not originally ready for managing streaming data. The ingestion phase is implemented exploiting big data components to save the time varying data into a compatible format for Hive, e.g. Parquet. During this first phase, the system adds the ingestion timestamp, i.e. *ts*, to each incoming data. The components involved in this phase are multiple and represents I2S$\langle\mathcal{T}\rangle$ operators, i.e. the real time ingestion phase of the data from external sources, and S2I$\langle\mathcal{T}\rangle$ operators, i.e. the save operation of the new stream to static files on HDFS support. The Augmentation and Synthesis Phase is implemented exploiting Hive query capabilities in order to enrich incoming data and transform it using FraPPE Commons Abstractions, e.g. adding a reference to FraPPE Frame that groups the data items by time and space. The components involved in this phase represent I2I$\langle\mathcal{T}\rangle$ operators. The Analysis phase exploits Hive query to offer results to the final user.

Let us now focus on the window operator in Hive. Differently from Spark, the window operator is not natively supported due to the batch nature of the framework. However, if we augment the data items with a frame ID during the Augmentation and Synthesis Phase, tumbling windows can be implemented grouping by Frame ID, while more complex windows, e.g., sliding windows involving multiple Frames, can be implemented using the Hive Window[15].

Figure 7 shows a simple example of a chain of operations that ingest a data stream, augment it with a Frame ID and simulates a tumbling window that counts the number of data items per frame. I_1 represents the data saved on

[14] https://spark.apache.org/docs/latest/streaming-programming-guide.html

[15] https://cwiki.apache.org/confluence/display/Hive/LanguageManual+WindowingAndAnalytics

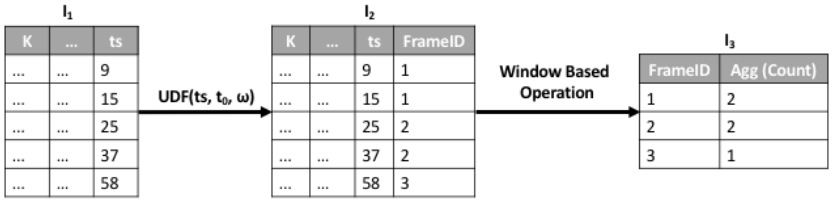


Figure 7: Example of data operation using hive

HDFS during the ingestion phase. I_1 is in form of a table containing various attributes and the ingestion timestamp *ts*. The data is augmented using a query that uses a User Defined Function (UDF) to attach a Frame ID based on the *ts*. Such an UDF takes is configured passing the opening time t_0 of the first window, and the length of a Frame ω. In the example $t_0 = 0$ and $\omega = 20$. The Frame groups the data items in windows and enables operation on time-varying data in a batch oriented system such as Hive. I_2 represents the augmented data. The Window Based Operation exploits the Frame ID to perform a simple count aggregation by applying a Group by on the Frame ID. I_3 represents the aggregated data.

5. Experiences

In this section, we report a subset of the analyses we run on different real–world scenarios. We selected those experiences because they are diverse and cover different business needs, cities, stakeholders, and adopted techniques. We classify them in two categories: 1) large–scale events, like the *Milano Design Week*, the *Milano Fashion Week*, the universal exposition of Milan 2015 (namely, *EXPO 2015*) and; and 2) longitudinal studies, respectively of a touristic city (namely, *Como*), of a business metropolis (namely, *Milano*) and of an open-space art event (namely, *The Floating Piers* by *Christo and Jeanne-Claude*).

In all these cases, we apply our conceptual model and describe the feasibility and advantages of the approach. Along the discussion, we will also present some representative visualizations that let the reader perceive the intuitiveness of the communication permitted by our framework. For each case, we report one or more of the analyzed dimensions.

Table 2 provides an overview of the experiences that we discussed in the remainder of the section. It highlights the complementarity of the difference experiences w.r.t. the different methods we discussed in Section 2 as well as its comprehensiveness in terms of the requirements presented in Section 1. **Req1** – aggregation, analysis and prediction along space and time – and **Req2** – integration of heterogeneous data sources, considering diverse content type and (temporal and spatial) granularity – are directly depicted as annotations to the various columns. The claim that all our experiences satisfy **Req3** – intuitive visualisation and exploration of results – is, instead, directly reported in the examples and illustrated in the figures included in the remainder of this section.

Table 2: Summary of experiences and their characteristics

	Experience	Content – Req1			Space – Req2			Time – Req1		Processing – Req2
		Type	Original	Augmented	Type of grid	Coverage	Analysis	Type of frame	Analysis	
Large-scale events	Milano Design Week	Call Data Records:	# of call in/out, # of sms in/out, # of Internet sessions		Regular 100x100 cells, 250x250 meters & irregular business driven	total for the regular; partial for the irregular	Pearson correlation, Jaccard distance	every 15 min	Gaussian process, anomaly detection	Real-time Natron , Spark, Hive
		Social Media: Twitter, Instagram and Facebook	hashtags, mentions, geo-coordinates	semantic entities					Holt-Winter	Real-time Natron , Spark, Hive
		Official App logs	geo-coordinates and time of access	events in the agenda, nationality of visitors, sentiment						a-posteriori Natron , Spark, Hive
		Official calendar and database	geo-coordinates and time-interval	venues, events						a-posteriori Natron , Spark, Hive
	Milano Fashion Week	Social Media: Twitter and Instagram	hashtags, mentions	semantic entities	Regular 20x20 cells 400x400 meters	total	Pearson correlation, Jaccard distance	every 15 min	social media response to events, k-means, Granger causality test	a-posteriori ad-hoc
	Floating Piers by Christo (art event)	Social Media: Twitter, Instagram	users' bio, tweets, hashtags, photos	photos of the piers, user profiling, user clustering	Irregular business driven	partial	deep learning on images, k-means	one-time data	NA	a-posteriori ad-hoc
Longitudinal studies	Milan (business city)	Google places	# of check-ins		Irregular business driven	total	Spearman correlation	every hour	Pearson correlation	a-posteriori Natron , Hive
		Foursquare	# of check-ins				Spearman correlation	every hour	Pearson correlation	a-posteriori, periodic Natron , Hive
		Social Media: Twitter and Instagram	hashtags, mentions, geo-coordinates	language			anomaly detection	quarterly	anomaly detection, Box-and-Whisker Plot	periodic Natron , Hive
		Census	demographics				Pearson Correlation	one-time data	NA	a-posteriori Natron , Hive
	Como (tourist city)	Call Data Records:	# of call in/out, # of sms in/out, # of internet sessions	gender, age range, nationality	Irregular data driven	total	Visual Analytics	1 hour	Visual Analytics	a-posteriori Spark, Hive
		Social Media: Twitter, Instagram and Facebook	hashtags, mentions	semantic entities						
		People Counting Sensors	# of people flowing in/out	people presence						
		WiFi hotspot logs	# of accesses					1 month		

5.1. Milan Design Week

This section reports our experience in monitoring Milan Design Week in Milano, Italy, across three editions, in three different years (2013, 2014 and 2016), with particular attention to the so-called FuoriSalone, a set of more than 1,200 events spread all over the city. The aim of the project was to feel the pulse of the city during the event. At this purpose we based our work on a set of integrated systems developed by Politecnico di Milano and TIM – Telecom Italia allowing fusion and visualization of social media streams and privacy-preserving aggregates of Call Data Records from the mobile phone network. In 2016 edition we have been able to collect further information coming from the official mobile app of the event, used by around 25,000 visitors.

We divided Milan in 10,000 `cell`s using a grid of 100 x 100 cells. Each cell has a size of 250 x 250 meters. We considered three sources of `event`s at `place`s: mobile phone calls/sms/internet-accesses, geo-referenced micro-posts related to the Milan Design Week and the 1,200 long-lasting events that are organised in 600 places spread around Milan during the Design Week. We captured a `frame` every 15 minutes. In each frame and for each pixel, we count the amount of mobile phone calls, text messages, Internet accesses (namely, mobile phone volume), the amount of the micro-posts on social networks related to the Milan Design Week, the number of Milan Design Week events, and the top hashtags used in each pixels.

The analysis of mobile phone data is based on CDR (Call Data Record) analysis. CDRs are generated by telecommunication networks to log the activity of the users, associated to a cell. Every mobile phone cell has a unique identifier, the Cell Global Identity (CGI). The CGI is characterized by the country, the Mobile Network Operator (MNO), the Location Area of the cell, the latitude and longitude of the barycenter of the cell and of the antenna, the distance between barycenter and antenna, and other properties.

Telco Big Data can provide a very relevant and dynamic overview of presence of people in the context of a specific city or territory, aggregated at the level of the single cell tower. However, since building and morphology of the territory impact on the effectiveness of the cells, the CDR information can describe the city's dynamics at the macro level but it cannot be used to precisely represent such dynamics at the micro level, e.g. input/output flows into one specific street or square.

The CDR analysis based on FraPPE requires to map each CGI to the cells of the grid, by assigning a coverage percentage to each cell. Then, anonymized CDRs are aggregated by CGIs and on time–slots of 15 minutes using privacy–preserving methods and are mapped to the pixel of the corresponding frame using the percentages.

Once this mapping to FraPPE was completed, we analysed 2 months of CRDs from Milan to build two Gaussian models for each cell: one grouping the frames by working days, and one grouping them by week-end days. 1.92

millions Gaussian models were built[16]. During the Milan Design Week 2013 and 2014, we analysed in real-time approximately 172 Mln calls/sms/internet-accesses by aggregating them for each pixel and for each frame and by computing how anomalous they are comparing each of them against the predictions of the Gaussian models built at set-up time. The anomaly index is obtained by computing how far the number of calls/sms/internet-accesses (which we refer as n) is from the average behaviour (which we refer as avg), keeping into account the computed standard deviation (which we refer as std). The formula to obtain the anomaly index can be compactly written as:

$$2\Phi_{avg,std^2}(n) - 1$$

where $2\Phi_{avg,std^2}$ is the cumulative distribution function of a Gaussian random variable with mean avg and variance std^2. Anomalies are identified by filtering all the records with an anomaly index greater than a given threshold.

As detailed in [28], the anomalous pixels correspond with high precision to pixels in which events of the Milan Design Week are happening. This allows us to provide experimental evidence for validating the hypothesis that the extra 400,000 people that come to Milano for the Design Week generate extra calls/sms/internet-accesses from the cells that contain the 600 locations of the 1,200 Milan Design Week events.

To process social streams, we used an evolution of Streaming Linked Data (SLD) framework [24] to tame velocity and variety simultaneously, namely Natron . The original data stream are injected in Natron in Activity Stream 2.0 format[17]. Natron semantically augments them using a custom Named Entity recognition and linking solution tailored on Milan Design Week [43]. A continuous query captures a frame every 15 minutes counting the number of distinct hashtags and semantic entities present in the geo-referenced microposts for each pixel. The results of this continuous query is a stream modelled in FraPPE .

As illustrated in Figure 8.(a) a (partial) *semantic* explanation of the mobile anomalies, can be attempted aggregating the top-10 hashtags used in those pixels. For instance, in Brera district the Italian hashtag of Milan Design Week (i.e., Fuorisalone) emerges. However, this technique is not dependable. For instance, in Tortona district also the hashtags of a popular TV shows (i.e., Amici, a popular Italian TV show) and its protagonists (e.g., Emma) appear. Once again, the solution is in the ability to compare the current top hashtags against the those predicted by a statistical model. This allows highlighting only the emergent hashtags of this frame for the selected pixels (see Figure 8.(b)).

As one can expect, the simple Gaussian model used for the mobile activity is not appropriate to predict hashtag usage. We found, instead, that an Holt-Winter method can be used [44] to predict the usage over time of a specific hashtag (e.g., #milan). In order to use Holt-Winter, we built synthetic frames

[16] 4 frames per hour X 24 hours X 2 day types (working and weed-end days) X 10.000 pixels.
[17] `http://www.w3.org/TR/activitystreams-core/`

(a) (b)

Figure 8: Social media used to explain the reason of anomalous peaks of presence in some pixels: (a) shows the most popular hashtags used in the anomalous pixels during Milan Design Week, whereas (b) highlights the emergent hashtags, i.e., the non predicted ones. While the generic most popular tags contains also hashtag about a popular TV show (i.e., Amici or Emma), the emergent hashtags are those of Milan Design Week.

that aggregate the captured frames in five parts of a day (i.e., 2am-7am, 7am-11am, 11am-2pm, 2pm-7pm and 7pm-2am). Moreover, as for the CDRs, we distinguished between working days and week-ends. This approach allowed to build effective predictive models for hashtags about the point of interest of Milan and about popular TV shows. Figure 9 illustrates how this method detects the anomalous usage of #milan during the Milan Design Week, which is highly correlated to the usage of #mdw – the official hashtag of Milan Design Week.

During the Milan Design Week, using Natron to compare those models with the observed usage of an hashtag, we were able to detect in real-time emerging hashtags. Figure 9 illustrates how the extra usage of #milan is correlated to the appearance of the official hashtag of Milan Design Week (i.e., #mdw).

Using the analyses described above, CitySensing identifies pixels where people is talking about Milan Design Week. As detailed in [28], those pixels are not as numerous as those identified as anomalous using the CDRs. However, they match with almost absolute precision the pixels in which Milan Design Week events happen. The most interesting finding is that almost all those pixels are contained in mobile anomalous ones. This provides further experimental evidence to validate the hypothesis that the anomalies observed in the CDRs are caused by the people coming to Milan for the Design Week.

The particular scheduling and geographical organization of the events of FuoriSalone, the informal happenings of the Milano Design Week, with most of the events concentrated in some specific areas of the city, enable also to perform analysis with irregular `grid`, based on the official areas of Fuorisalone. An examples of analysis performed on this grid is shown in Figure 10, that represents the results of a semantical analysis on the text of tweets related to each area of Fuorisalone. We collect for each (irregular) `pixel` the tweets geolocated in some `place` of the `cell` and the tweets that speaks about the `event`s contained in the `pixel`, and we perform on them a semantical text analysis in order to

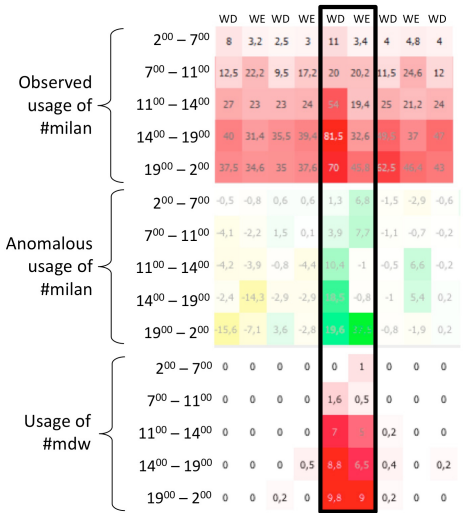


Figure 9: Highlighting of anomalies in hashtag usage: The hashtag #milan is used more often during the Milan Design Week. Forecasting the #milan time-series using Holt-Winter method, we were able to identify the anomalous usage, which is highly correlated to the usage of #mdw – the official hashtag of Milan Design Week.

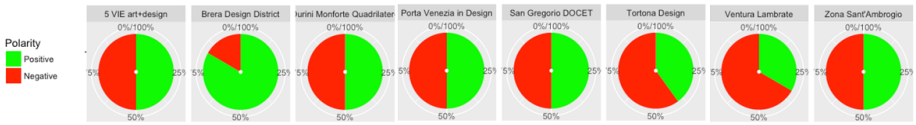


Figure 10: Share of positive and negative sentiment calculated for posts associated to the different MDW cells.

extract the sentiment polarity of the text. Then we filtered neutral tweets and we compare the number of positive and negative tweets in each `pixel` in order to show, for each daily `frame`, which `pixel`s (that corresponds to an official area of Fuorisalone) are most successful according to the opinions of social network users.

During the 2016 edition of the event we had the opportunity to collect data coming from an additional relevant source: the official App of Fuorisalone. In particular, we had access to the GPS position of `place`s where the users open the App and the `event`s inserted in the agenda on the App. In this context, we apply the method of squared `grid` tessellation of the city, in order to analyze the correlation between couple of different signals.

We showed the correlation between the use of the App and the number of Fuorisalone events. To do so, we consider as `event`s the use of the App in a `place`, that generates a GPS record, and the scheduled `event` of MDW with

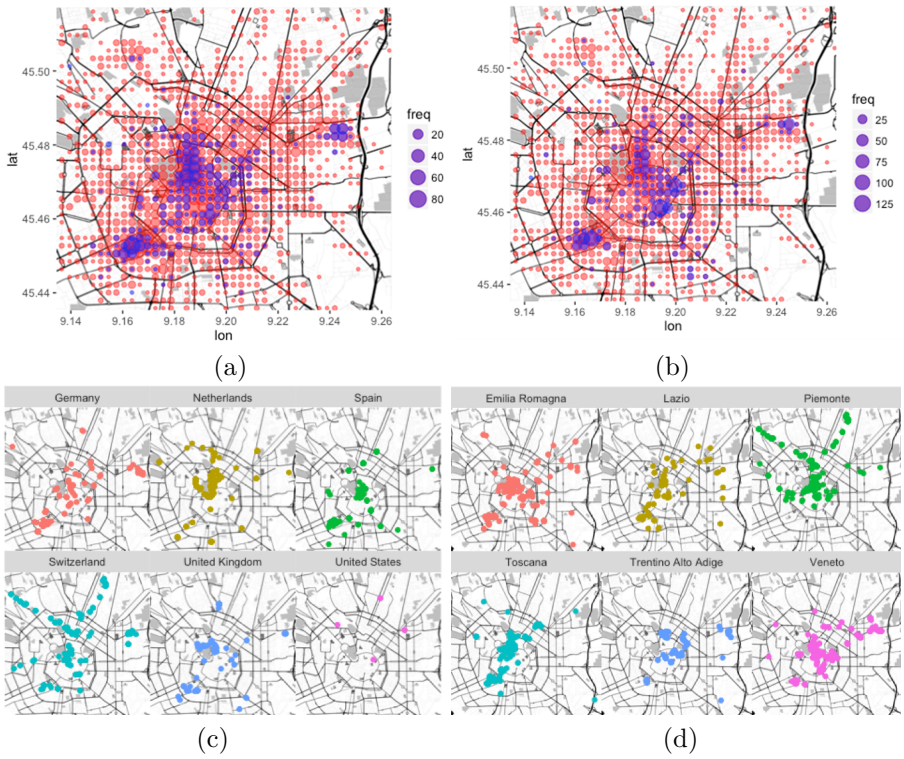
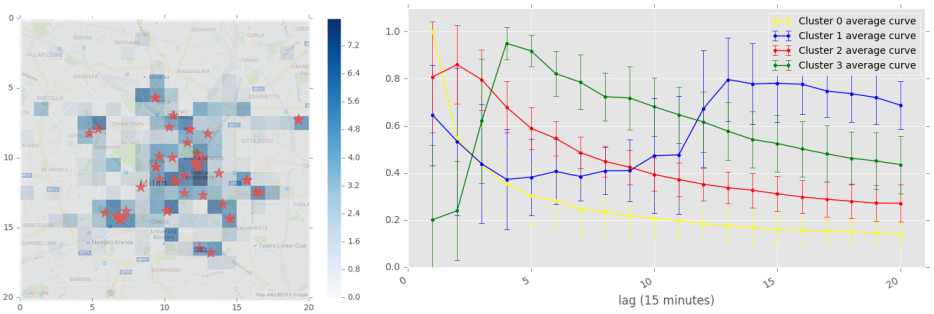


Figure 11: (a) Number of Mobile app GPS observations collected in one day inside each square of the grid (red dots) correlated with the number of events of Milano Design Week scheduled for the same day in the same square. (b) Correlation between GPS observations (red dots) and geolocated posts on social networks (blue dots) inside each square. The localization of the five largest groups of European and US visitors (c) and Italian visitors (d).

their `place` that users put in their agenda. We aggregate on `pixel`s and we captured daily `frame`s as shown in in Figure 11(a).

Another available source of `event`s is the geolocated activity on the public social networks: we collect the Twitter and Instagram posts geolocated in the `place`s contained in each `cell` and we aggregate them in the same `pixel`s of GPS observation `event`s, as shown in Figure 11(b). Both `frame`s show the increasing of the `event`s in the areas of the Fuorisalone.

Another interesting use case is represented by the analysis of the provenance of the visitors during the Milano Design Week. In order to estimate the provenance of the visitors we used GPS information collected by the Official App, extracting the GPS position of the first observation logged for each user before the days of the MDW, assuming that the users download the App before they arrives in Milano (so, probably, at home). Using appropriate shape files it is possible to map each GPS location to a Country in the world, obtaining the provenance of the user.



(a) Density of posts (b) Granger Causality tests result

Figure 12: (a) Geographical dispersion over the cells of physical events (red stars) and density of social media activity (blue); (b) Granger causality test curves between physical events and social media response of each brand during the MFW, clustered by similarity of behaviour.

Mapping the GPS observation `events` in the `grid` of `pixels` it is possible to visualize for each daily `frame` and for each people group what are the most popular areas of the city. Figure 11 (c) shows the localization of the five largest groups of European foreign visitors and United States visitors. Figure 11 (d) shows the region of provenance of Italian visitors and their distribution in the events.

5.2. Milan Fashion Week

This experience deals with the problem of understanding the social media response and the associated physical presence to a scheduled and popular real world event, the Milano Fashion Week (MFW) occurred from the 24^{th} to the 29^{th} February of 2016, analysing the behaviour of users who re-acted (or pro-acted) in relationship with each specific fashion show during the week. MFW represents the most important meeting between market operators in the Italian fashion industry. Out of the 170 shows, we are interested only in the catwalk shows, which are the core of the fashion week. The whole set of catwalks includes a total of 73 brands; among them, 68 brands organise one single event, 4 brands organise 2 events, and 1 brand organises 3 events.

We initially extracted posts by invoking the social network APIs of Twitter and Instagram; for identifying the social reactions to MFW, we used a set of 21 hashtags and keywords provided by domain experts in the fashion sector, i.e., researchers of the *Fashion in Process* group (FIP) of Politecnico di Milano.[18] We focused on 3 weeks: the one before, the one after and the one of the event. In this way, we collected 106K tweets (out of which only 6.5% geolocated) and 556K Instagram posts (out of which 28% geolocated); eventually, we opted for considering only Instagram posts, as they represent a much richer source for

[18] `http://www.fashioninprocess.com/`

the particular domain of Fashion with respect to Twitter [45, 46]. Figure 12 (a) shows the map representing the geographical distribution of events (represented by red stars) and post density using a synthesized frame where the pixel are darker where the density is higher.

Our first goal is to perform a **temporal analysis** aiming at characterizing the time at which social media respond to the events which appear in the official calendar and are linked to specific brands. Informally, we observe either peaks of reactions which then quickly disappear, or instead slower reactions that tend to remain observable for a longer time. Estimating the time latency of social responses to events is important for the brands, which could plan reinforcement actions more accurately, essentially by adding well-planned social actions so as to sustain their social presence over time. We run Granger causality for each brand to compare the physical events and the social media reaction, and then we clustered the brand by similarity of the Granger curves. Figure 12(b) shows the clusters of Granger causality curves of the brands.

Our second goal was to **analyse the geographical dispersion** of social media response. We have two different spatial signals: (1) the calendar events; and (2) the volume of social media posts on the Web with geographical information attached, i.e., latitude and longitude. Given these two signals, several features can be computed in order to describe the spatial dispersion of posts following an event.

Following the FraPPE approach, we built a grid of cells above the area of Milano city, and assigned each post to the appropriate cell. The grid has a square shape, with sides of $10km$, divided into 20 rows and 20 columns, for a total of 400 cells of $500m \times 500m$.

According to the FraPPE model, an event E may be organized by a brand B at time τ_n, hosted in the place P, located the cell C of the grid G. C is related to the pixel P and G is captured by the frame F. A user U may contribute with some original content M (e.g., Instagram post), related to the event E, which in turn is going to be augmented by an automated enriching and analysis process that may add entities, as well as extract visual properties (color, pattern, ...) and concepts (objects, people, ...) from posted images.

We computed pixel level synthesis of the collected contents. We named *Alive pixels* those where the percentage of posts shared in the considered time-window is more than 1% of the total number of posts in the frame in the same time-window. We named *Active pixels* those where the percentage of posts shared in the considered time-window is more than 10% of the total number of posts in the frame. We named *Strongly Active pixels* those where the percentage of posts shared in the considered time-window more than 20% of the total number of posts in the frame.

We computed the number of alive, active and strongly active cells for all brands; we also computed the differences between subsequent durations (e.g. 3h - 6h) by counting how many cells changed their state.

We then computed different measures that reflect the dispersion of the social media signal over time, using: *Gini coefficient*, *Average distance* of the social media signals from the event location, and the number of *alive*, *active* and

strongly active cells.

By observing the behaviour of pixels, we noted that:

- As we increase the width of the time-window, the number of *alive pixels* also increases. On the other way, the number of *active* and *strongly active pixels* is floating in the range from 1 to 3, with very few brands reaching 4 *active pixels*.

- At the start of the event, posts are shared near the event location, but as we look at the bigger picture, including 24 hours or even the entire period of 24 days, the *average distance* is increasing, showing the growing dispersion of the social signal.

- The *Gini coefficient* proves how the concentration of the social signal remains always high, due also to the fact that the low percentage of users that allows Instagram to geo-tag their own photo is reducing the number of authors implied in this study, and so the few authors with high volumes of posts generated are biasing the results. However, looking at the *Gini alive coefficient*, that refers to the Gini coefficient computed only over the pixels that result alive for at least one brand in the specific time-window, we can see a weak smoothing of the concentration strength with the increasing of the time-scope.

5.3. Como SC^2

Como Smart City for Smart Citizens (Como SC^2) is a big-data integration project started in 2016 involving, along with other partners, Municipality of Como, Politecnico di Milano and TIM-Telecom Italia, that represents a pilot in the Smart City context. The purpose of the project is to create a system for integration, analysis and interpretation of the large amount and heterogeneous data coming from different sources, in order to understand the urban dynamics and support the decision making process of the Public Administration.

We analyzed the dynamics of **mobile phone traffic** (preventively anonymized and aggregated, according to privacy-preserving policies) in different areas of the city.

We have identified seven city cells, according to the distribution of the phone antennas and the characteristics of the area. We named those *irregular, data-driven cells*: historical city center, lakeside promenade, touristic areas outside from the historical center, lake area, mountain area around the city, business and universities area, industrial outskirts. The map in Figure 13 represents the distribution of the seven cells.

Inside each area, we analyze the trend of mobile phone traffic capturing `frame`s by day and by hour. Anonymized mobile phone data contains also information about the SIM (like international dial-code) and demographics information about the owner of the SIM (like gender or age-range): these allows us to perform analysis not only about the `event`s of people presence but also about the characteristics of people (`event content`).

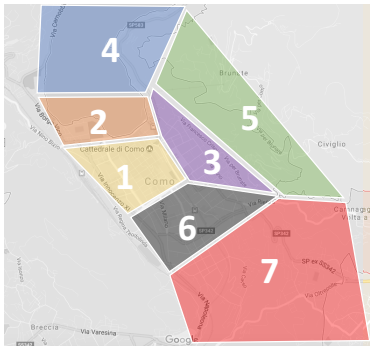
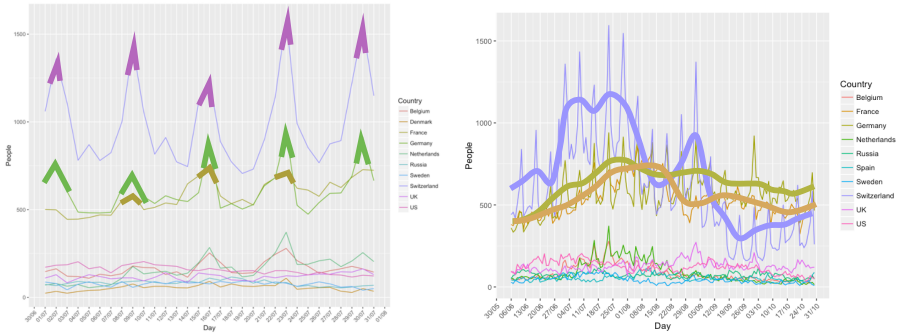


Figure 13: The seven irregular data-driven cells based on mobile phone data that cover the Como territory.



(a) (b)

Figure 14: (a)Number of foreign visitors per country per day in July in Como: Swiss, German, and French are the most present in the weekends. (b) Number of foreign visitors per country per day in Como from June to October. One can notice that Swiss visitors decrease sensibly in September.

One example is the comparison between the number of visitors from neighbouring countries of Italy. As one can see in Figure 14, Swiss people usually come to Como for shopping on Saturdays in July while this trend dramatically decreases in September.

Besides mobile phone data analysis, we instrumented in the Cathedral Square of Como (*Piazza Duomo*) with a set of **IoT (Internet of Things) sensors for counting people** passing in the square. The installation of the IoT sensors covers all the access to the square, and each sensor count how many people pass from the access every minute and push the results in real-time to the collect and visualization system.

Starting from the collected data, and working at a sensible temporal aggregation (for example one `frame` per hour), it is possible to construct the average trend of passages *from* and *to* the Duomo Square according to the day of the week and the hour of the day. This average trend represents the starting point

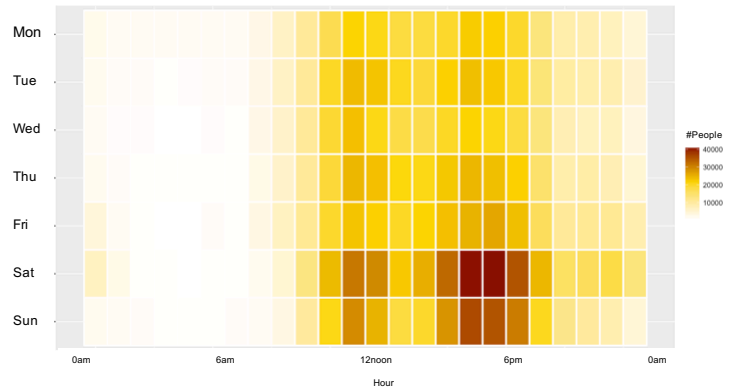


Figure 15: Patterns of people presence in Duomo Square on working days and weekends: Tuesday, Thursday and Saturday are more crowded due to open market in the streets; Week ends are extremely crowded (including Saturday night).

to analyze trending pattern and anomalies.

Analyzing the data is possible to observe, at a high level, two different patterns of people presence in Duomo Square: one for working days and one for weekends, with a significant increase of people during Saturdays and Sundays, with respect to working days. Figure 15 represents graphically these analyses. More in details, it is possible to find some differences inside the two patterns. For example in week-ends clearly emerge a difference during the evenings: Saturday evening shows a sort of persistence of people presence, while Sunday evening appears more similar to working-day evenings. Another significant difference is the increase of people presence on Tuesday and Thursday mornings, with respect to the other working days. They are significant because the local market activities are organized in the square during such mornings.

5.4. Urbanscope

In this experiment [3], we build an analysis to understand multilingualism in the city. We focused on Milano again, and we used Twitter to analyse the language mix of the city and to capture language communities within the city neighbourhoods. We refer to language communities as those groups of individuals sharing either the language used on Twitter or the language of their country of origin. We then compare these "digital" communities, discovered through Twitter, with the real communities identified by the traditional census data. Milan, a city which is increasingly becoming an international melting pot, is chosen as a case study for this work. We use quantitative tools to analyse the micro-level data collected from individuals, but we also develop visual solutions to show and navigate the aggregated results.

Two main data sources are identified as relevant for the study. The first

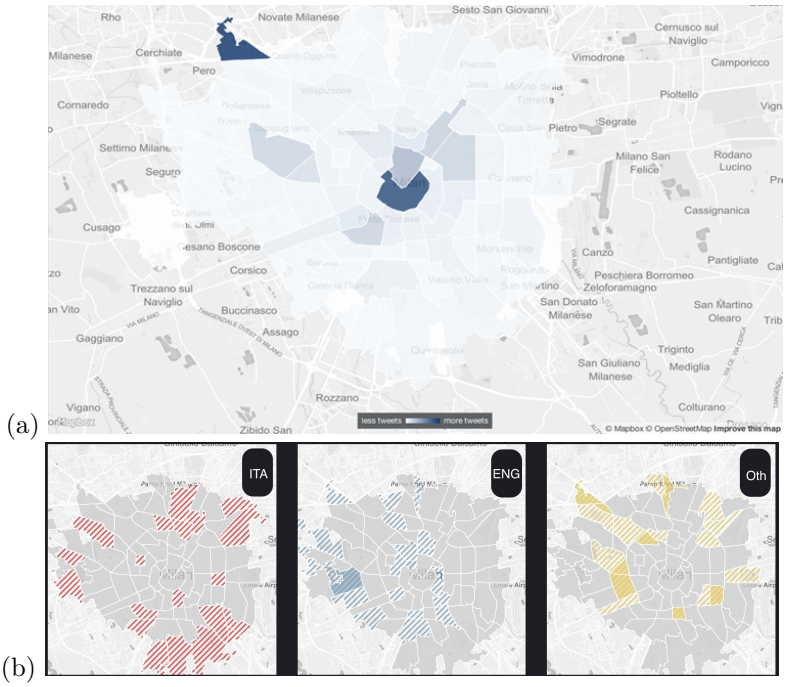


Figure 16: (a) Density of multi-language tweets (excluding English and Italian) in Milan urban area, on a grid where the cells are an irregular, business-driven total coverage of the area, as specified by the official city districts (NILs); (a) anomaly detection on Twitter languages for Italian, English and all the other languages.

dataset, provided by the Municipality of Milan[19], describes the geographical zones (districts, officially named NILs[20]) in which the city is divided. The NILs, in FraPPE terminology, are the cells of an irregular partial business-driven grid in which the city is divided. This dataset provides demographic and economic data for each NIL. The second dataset derives from Twitter. Twitter is considered the most suitable source for the purpose of our analysis, since it is largely based on written text and features the option of geo-locating the posts (although only a very limited share of users actually opt for using this feature).

For this study, we consider only the tweets geo-tagged within the boundaries of the municipality of Milan for a period of 18 months (August 2014–December 2015). The language used in the tweets is part of the metadata of each tweet, as provided by Twitter API. Tweets containing words written in different languages or whose language could not be detected are classified as "undefined language" and are excluded from our dataset. The entire dataset contains 1,109,693 tweets,

[19] `http://dati.comune.milano.it/`

[20] NIL stays in Italian for Nuclei di Identità Locale. Translated in English, a NIL is an area with a Local Identity.

with 1,007,314 being associated to a defined language. There are 793,838 tweets whose metadata position them precisely into one of the 88 NILs within the Milan municipality area, i.e., they are events captured by pixels corresponding to the cells. We run three types of analyses: density analysis, anomaly detection, and correlation with official census data. Figure (a) reports the density of languages by NIL, while Figure (b) reports the anomalous presence of languages in the different NILs in a given month.

For the correlation with the census data, we considered the information for the municipality of Milan, recorded at December 31st 2014. The official census reported 253,334 foreign people residing in Milan, about 18.9% of the total population. Among them, the largest community is that of Filipinos (41,237 people), followed by Egyptians (35,597 people), Chinese (25,928) and Peruvians (20,462). By running correlation analysis between census and Twitter language data, we realized that only in some cases the language communities detected through Twitter correspond to residents density in any given NIL. This is the case for some of the Arabic and Spanish communities, while languages like Portuguese, Dutch, Norwegian and Albanian are underexposed on Twitter, and others like Tagalog, Ukrainian and Romanian are overexposed. The most noticeable overexposed languages in many areas are Arabic and Spanish. These language communities might consist of those generations descending from North African and South American immigrants in the 1980s and in the 1990s. In fact, while the new generations have acquired Italian citizenship, they might have maintained a double language identity and might use their original language to communicate within their community.

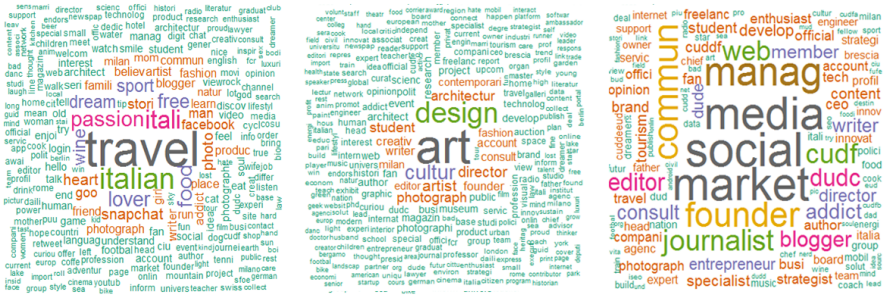
5.5. The Floating Piers

Our most recent studies [47, 48] exploit Instagram and Twitter datasets from a famous art work called "The Floating Piers" that was created by the world-renowned artists Christo and Jeanne-Claude [21] and exposed to the public view at the Lake Iseo in Italy, from June 18 through July 3, 2016. We extracted the social media content relevant to the event, during a time period from June 10^{th} to July 30^{th} 2016, that contains 30,256 Instagram posts and 14,062 tweets, using Twitter and Instagram APIs.

With the aim of building prototypical profile of the visitors of the event, we analysed the original content associated with each user in the dataset, including the biography of the user who posted the content, the text of the post and hashtags in order to explore the user behaviour and profile in various settings. Furthermore, we automatically extract concepts and features from the photos and we use this augmented content to understand the users behaviour and the event attendees' demographics.

We run clustering of users based on their interests. Figure 17 reports the tag clouds of the users in the three resulting clusters. As one can easily see, people in first cluster mostly talk about Travel introducing themselves in their Twitter

[21] `http://christojeanneclaude.net/projects/the-floating-piers`



(a) Travel Lovers (b) Art Lovers (c) Tech Lovers

Figure 17: Word cloud representations for the three clusters of users engaged with the Floating Piers case.

biography. People in second cluster are Art lovers and people in third cluster state their positions as Technology fans and social media marketing addicted.

6. Related Work

The relevance of urban computing, or urban informatics, has been recognized since long, as testified by the rich literature on the topic. A recent survey on urban sensing [6] clearly shows the huge opportunity for research to exploit mobile phones data to get insights into urban dynamics and human activities.

One of the very common applications is to use CDRs to estimate the density of crowd and vehicles in the different urban regions covered by the dataset [7, 8]. Another example involves the detection of people habits. Ratti et al [49] present Mobile Landscape project, one of the first urban analysis based on the geographical mapping of cell phone usage at different times of the day in the metropolitan area of Milan. Becker et al. [9] capture key mobility patterns within Morristown, NJ, by identifying users' home and work locations from CDRs. This information is particularly useful for urban managers and authorities that are responsible for efficient public transportation systems. Also in [10] the authors have tested the four Jacobs conditions that promote life in cities by using CDRs and [11] quantifies the effects of ownership bias on mobility estimates by coupling two data sources from the same country. In other example [11] combine CDR's with other cellphone-related logs (e.g., tower pings, cellular handovers) in order to compare human mobility patterns derived from CDRs vs. from the complete dataset.

Other studies focus on using CDR's to track individuals motion patterns like Gonzalez et al [12] characterized each individual by a time-independent characteristic travel distance and a significant probability to return to a few highly frequented locations and Candia et al [13] investigate patterns of calling activity at the individual level and show that the interevent time of consecutive calls is heavy-tailed.

Especially when merged with other kind of information, CDRs can reveal interesting insights for city dynamics and urban monitoring. The platform built by Calabrese et al. [14] combines the users' cellular data with the real-time location of buses and taxis to model the car traffic in Rome. Krings et al [15] instead express formal laws regulating the communication intensity between pairs of cities in Belgium by exploiting the zip code information of customers. Also Wesolowski et al [50] propose integration architecture to collect data and consolidate in the central DW from many different operational databases of different telecom operators.

Quercia and his colleagues use CDRs to study human mobility related to special planned events in Boston. In [16] the authors show that there is a high correlation between the kind of event, e.g., sport, theater, music, family events, and the home location area of its attendees. In [17] they build a recommendation system for social events and find out that the most effective algorithm recommends those events that are popular among local residents.

In this work, we also investigate special events but, differently from [16, 51], we do not require the events to be isolated in time and space.

Although CDRs are a priceless source to gather underlying patterns of cities and their citizens, they hide some limitations since they can not reveal any information about people interests and thoughts. A parallel investigation of social media steams has recently carried out by the research community as a powerful mean to explore people opinions and preferences with regard to specific venues in the city. Singh et al [18] introduce the concept of social pixel that aggregates social interests of users about any particular theme and from any particular location.

Sentiment analysis covers a wide range of applications in cities. Authors in [19] propose a city sensing architecture from Twitter data to monitor user opinions about events and topics. There are other works like [20, 21, 22] that examine temporal, spatial, and microeconomic patterns of human activities using publicly available digital traces, such as social media data streams. Other efforts [52, 53] analyze geo-located Twitter messages and geographical preferences in order to predict global patterns of human mobility. Furthermore, [23, 24] monitor, analyze, and assess city-scale events using the Streaming Linked Data framework to process social data streams.

The exploration fields of the urban macroscope are infinite. Among all, one of the feature of interest for policy makers and cities managers [54] is the extremely diversified composition of the language mix, or multilingualism. This interest is motivated by the increasing immigration flow towards cities [55], which results in rapidly changing population density [56]. Multilingualism has also a broad scope in academia. In particular, different papers approach the issue of multilingualism from a historical perspective. [57], for example, analyses the city of Singapore, [58] the city of New York, [59] develop a cross-linguistic perspective on Gothenburg, Hamburg, The Hague, Brussels, Lyon and Madrid. Moreover, [60, 61, 62] characterize cities and their neighborhoods from different aspects namely safety, culture and demographics through social media networks specifically Twitter. Authors in [63] present a tool called City Murmur with the

aim to show how different media differently describe the urban space through the attention that is payed on each street of a city. It wants to build a time-based narration, an historical archive of media coverage of the urban space which is able to reveal some hidden dynamics useful for city policy support, critical media analysis, and sociocultural research.

In a nutshell, there has been a growing interest in exploiting CDRs and social media streams to reveal emerging patterns within a city. However, the joined use of both types of sources has not yet been carried out. In [27, 28] we addressed this gap by presenting CitySensing, a system that fuses digital footprints from both CDRs and social media.

In this regard, [25, 26] closely relate to our work by considering both kinds of data. However, the authors analyze the two datasets in parallel and show that the time series of phone communications and social activities related to the same areas reveal a strong similarity. We instead investigate the importance of using the two streams of data jointly together and use social streams to validate key insights obtained with cellular data.

As a final remark, our additional exploitation of customer demographic data, e.g., age and gender, already present in CDRs has never been used for urban sensing and event management before.

7. Conclusion

In this paper, we proposed our approach for Urban Data Science at scale. It is a multi-disciplinary approach that combines computer science, statistics, and visual design disciplines together with domain expertise of stakeholders (city administrators, event organizers, urban planners, resource managers) to find a unifying framework for dealing with large scale data for city analytics.

In particular, we proposed a conceptual model (**RP1**) that was build by extending the FrAPPE framework, which proved valid for describing real world scenarios in the smart city context.

Table 3 presents in which experience the various FraPPE concepts were introduced (the cells of the table whose background is filled in gray) and proposes an analysis of how important they were in the various experiments. The basic FraPPE concepts [29] were introduced during the Milan Design Week experiences. The accent was on *Place* and *Event*. The *Cells* and the *Pixel* – its time-variant counterpart – were important to bridge the gap between the analyzed data and the visual analytics we intend to enable [28]. The *Grid* and *Pixel* – its time-variant counterpart – were useful as abstractions, but they did not play a key role. The provenance of all steps of the analysis were documented using the generic *Action* concept; captured and synthesized were only possible value of an attribute of the action.

The Milano Fashion Week experience was key to extend the original FraPPE with the concepts that allow describing the content as well as to extend FraPPE with some provenance concepts. Specifically for this experience, we introduced the distinction between *original* and *augmented* content at *event-level*. We also

Table 3: A comparison of how the FraPPE concepts are used in the two large-scale events of Milan Design Week (MDW) and Milan Fashion Week (MFW) as well as in the three longitudinal analysis we performed in Milan, Como and for The Floating Pier (TFP). The 'x' symbols have the following meaning: xxx – key concept; xx – important concept; and x – useful concept. The lack of stars means that the concept was not used. The gray cells highlight when the concept was first introduced in FraPPE .

		MDW	MFW	Milan	Como	TFP
Spatial	Place	xxx	xxx	xxx	x	x
	Cell	xx	x	xxx	xx	xx
	Grid	x	x	x	xx	xxx
Temporal	Event	xxx	xxx	x	x	xx
	Pixel	xx	x	xxx	xx	
	Frame	x	x	x	xx	xx
	- CapturedFrame	xx			xxx	
	- SynthetizedFrame	xx			xxx	xx
Content	Content	xxx	x			
	- Event-level content		xxx	x	x	xxx
	- Original content		xxx	x	x	xxx
	- Augmented content		xx	xx	x	xx
	- Pixel-level synthesis		xxx	xxx	xxx	x
	- Frame-level synthesis		xx		xxx	xx
Provenance	Action	x				
	- Capture		x	xx	xx	x
	- Synthetize		xx	xx	xx	x
	- Augment		xx	xx	xx	xx

perceived the need to model the content we connected to each pixels, namely the *pixel-level synthesis*. We reflected this extension also in the provenance part of FraPPE introducing the *capture*, the *augment* and the *synthesize* actions.

The longitudinal analysis, which we performed on Milan, Como and for The Floating Pier open air art exhibition, served as validation for the extended version of FraPPE . All the concepts were used in all deployments, although their use and benefit depends on the different types of analysis performed. In the Milan experience large emphasis was given to the *cells* (i.e., the NILs), the *pixels* – their time-varying counterpart – and the *pixel-level synthesis*. In the Como experience, more emphasis was posed on the *grid*, the *frames* – their time-varying counterpart – and the *frame-level synthesis*, which was introduced in FraPPE during this experience. Last but not least, the experience on the Floating Piers demonstrated that crowd monitoring can be applied also to suburban areas and that content shared by people varies based on the kind of medium used (text or images) and is not necessarily coherent, therefore any analysis needs to keep into account these discrepancies.

We also devised a logical architecture and a set of implementations (**RP2**) for different scenarios and we deployed them in various settings (**RP3**). The experiences reported demonstrate that we are able to addresses diverse data formats,

as well as heterogeneous time and space granularity of the sources (**req2**). We used our method for integrating those sources in a unified framework along time, space (**req2**). Table 2 offers an at–a–glance view of the comprehensive set of experiences that we conducted. Since all our solutions were aimed at city stakeholders, we put a lot of effort in the valorization of the results through intuitive and immediate visualization tools (**req3**). We showed the approach at work on diverse city-scale problems, demonstrating its feasibility, generality and effectiveness, and we discussed the value that could be obtained.

We are aware that our work has some limitations. First of all, it is extremely hard to define the time and space granularity for the analysis. Different phenomena are visible at different granularities and artifacts can be generated by analyzing data at the wrong level. Moreover, we have only started exploring the possible relationships between frames over time (e.g, see our experience on causality conducted during Milan Fashion Week). A richer time model can unveil the possibility to capture relationships between events (e.g., simultaneity, asynchronism, etc.). Last but not least, the strength of FraPPE – i.e., its high level of abstraction – is also its weakness. FraPPE works well as unified modeling framework, when the analysis pivots around *contents* generated at *places* during *events* that need to be aggregated and compared other time and space. While, in this paper, we supported with numerous experience our claim that FraPPE supports those type of analysis, we recognize that more types of analysis exist.

Future and ongoing work consists in defining further scenarios and challenge the approach with the additional requirements that arise from the limitations we discussed in the previous paragraph.

Acknowledgements

We wish to acknowledge the contribution of all the stakeholders of the use cases discussed in the paper, in particular: Como municipality, EXPO 2015, Musei di Brescia, RSE - Ricerca Sistema Elettrico, Fuorisalone-Studiolabo, Fashion in Process research group, and Fluxedo for the interesting and valuable discussions and contributions. The agreement between Politecnico di Milano and Telecom Italia within the Urbanscope project allowed using privacy preserving aggregates of the mobile phone activity of Telecom Italia in Lombardy region. We acknowledge also the contribution to the work of Silvana Bernaola and Irene Pappalardo (Telecom Italia).

References

[1] F. Calabrese, K. Kloeckl, C. Ratti, M. Bilandzic, M. Foth, A. Button, H. Klaebe, L. Forlano, S. White, P. Morozov, S. Feiner, F. Girardin, J. Blat, N. Nova, M. P. Pieniazek, R. Tieben, K. van Boerdonk, S. Klooster, E. van den Hoven, J. M. Serrano, J. Serrat, D. Michelis, E. Kabisch, Urban computing and mobile devices, IEEE Pervasive Computing 6 (3) (2007) 52–57.

[2] McKinsey, How to make a city great. (2013).
URL `http://www.mckinsey.com/insights/urbanization/how_to_make_a_city_great`

[3] M. Arnaboldi, M. Brambilla, B. Cassottana, P. Ciuccarelli, S. Vantini, Urbanscope: A lens to observe language mix in cities, American Behavioral Scientist 61 (7) (2017) 774–793.

[4] M. Batty, Big data, smart cities and city planning, Dialogues in Human Geography 3 (3) (2013) 274–279.

[5] J. de Rosnay, Le macroscope: vers une version globale, Editions du Seuil, 1975.

[6] F. Calabrese, L. Ferrari, V. D. Blondel, Urban sensing using mobile phone network data: A survey of research, ACM Comput. Surv. 47 (2) (2014) 25:1–25:20. doi:10.1145/2655691.

[7] G. McArdle, G. Di Lorenzo, F. Pinelli, F. Calabrese, E. Van Lierde, Analyzing social events in real-time using big mobile data, IEEE COMSOC MMTC E-Letter.

[8] R. Cáceres, J. Rowl, C. Small, S. Urbanek, Exploring the use of urban greenspace through cellular network activity (2012).

[9] R. A. Becker, R. Caceres, K. Hanson, J. M. Loh, S. Urbanek, A. Varshavsky, C. Volinsky, A tale of one city: Using cellular network data for urban planning, IEEE Pervasive Computing 10 (4) (2011) 18–26.

[10] M. De Nadai, J. Staiano, R. Larcher, N. Sebe, D. Quercia, B. Lepri, The death and life of great italian cities: a mobile phone data perspective, in: Proceedings of the 25th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee, 2016, pp. 413–423.

[11] A. Wesolowski, N. Eagle, A. M. Noor, R. W. Snow, C. O. Buckee, The impact of biases in mobile phone ownership on estimates of human mobility, Journal of the Royal Society Interface 10 (81) (2013) 20120986.

[12] M. C. Gonzalez, C. A. Hidalgo, A.-L. Barabasi, Understanding individual human mobility patterns, arXiv preprint arXiv:0806.1256.

[13] J. Candia, M. C. González, P. Wang, T. Schoenharl, G. Madey, A.-L. Barabási, Uncovering individual and collective human dynamics from mobile phone records, Journal of physics A: mathematical and theoretical 41 (22) (2008) 224015.

[14] F. Calabrese, M. Colonna, P. Lovisolo, D. Parata, C. Ratti, Real-time urban monitoring using cell phones: A case study in rome, IEEE Transactions on Intelligent Transportation Systems 12 (1) (2011) 141–151.

[15] G. Krings, F. Calabrese, C. Ratti, V. D. Blondel, Urban gravity: a model for inter-city telecommunication flows, Journal of Statistical Mechanics: Theory and Experiment 2009 (07) (2009) L07003.

[16] F. Calabrese, F. C. Pereira, G. Di Lorenzo, L. Liang, C. Ratti, The geography of taste: Analyzing cell-phone mobility and social events., in: Pervasive, Vol. 10, Springer, 2010, pp. 22–37.

[17] D. Quercia, N. Lathia, F. Calabrese, G. Di Lorenzo, J. Crowcroft, Recommending social events from mobile phone location data, in: Data Mining (ICDM), 2010 IEEE 10th International Conference on, IEEE, 2010, pp. 971–976.

[18] V. K. Singh, M. Gao, R. Jain, Social pixels: genesis and evaluation, in: Proceedings of the 18th ACM international conference on Multimedia, ACM, 2010, pp. 481–490.

[19] K. B. Ahmed, M. Bouhorma, M. B. Ahmed, Smart citizen sensing: A proposed computational system with visual sentiment analysis and big data architecture, International Journal of Computer Applications 152 (6).

[20] K. Cheliotis, Capturing real-time public space activity using publicly available digital traces., in: CitiLab@ ICWSM, 2016.

[21] D. Hristova, D. Liben-Nowell, A. Noulas, C. Mascolo, If you've got the money, i've got the time: Spatio-temporal footprints of spending at sports events on foursquare., in: CitiLab@ ICWSM, 2016.

[22] M. Lee, R. Farzan, B. S. Butler, This is not just a café: Toward capturing the dynamics of urban places., in: CitiLab@ ICWSM, 2016.

[23] A. Psyllidis, A. Bozzon, S. Bocconi, C. T. Bolivar, A platform for urban analytics and semantic data integration in city planning, in: International Conference on Computer-Aided Architectural Design Futures, Springer, 2015, pp. 21–36.

[24] M. Balduini, E. Della Valle, D. DellâĂŹAglio, M. Tsytsarau, T. Palpanas, C. Confalonieri, Social listening of city scale events using the streaming linked data framework, in: International Semantic Web Conference, Springer, 2013, pp. 1–16.

[25] F. Botta, H. S. Moat, T. Preis, Quantifying crowd size with mobile phone and twitter data, Royal Society open science 2 (5) (2015) 150162.

[26] E. Cho, S. A. Myers, J. Leskovec, Friendship and mobility: User movement in location-based social networks, in: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '11, ACM, New York, NY, USA, 2011, pp. 1082–1090.

[27] E. Della Valle, M. Balduini, Listening to and visualising the pulse of our cities using social media and call data records, in: International Conference on Business Information Systems, Springer, 2015, pp. 3–14.

[28] M. Balduini, E. Della Valle, M. Azzi, R. Larcher, F. Antonelli, P. Ciuccarelli, Citysensing: Fusing city data for visual storytelling, IEEE MultiMedia 22 (3) (2015) 44–53. doi:10.1109/MMUL.2015.54.

[29] M. Balduini, E. Della Valle, FraPPE: A Vocabulary to Represent Heterogeneous Spatio-Temporal Data to Support Visual Analytics, Springer, 2015, pp. 321–328.

[30] V. K. Singh, M. Gao, R. Jain, Social pixels: genesis and evaluation, in: ICM 2010, 2010, pp. 481–490. doi:10.1145/1873951.1874030.

[31] R. Battle, D. Kolas, Geosparql: enabling a geospatial semantic web, Semantic Web Journal 3 (4) (2011) 355–370.

[32] J. R. Hobbs, F. Pan, Time Ontology in OWL (September 2006).

[33] Y. Raimond, S. Abdallah, The event ontology, http://motools.sf.net/event (2007).

[34] K. Belhajjame, J. Cheney, D. Corsar, D. Garijo, S. Soiland-Reyes, S. Zednik, J. Zhao, PROV-O: The PROV Ontology, Tech. rep., W3C (2012).

[35] M. Balduini, E. Della Valle, Frappe: A vocabulary to represent heterogeneous spatio-temporal data to support visual analytics, in: The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, Proceedings, Part II, Vol. 9367 of LNCS, Springer, 2015, pp. 321–328.

[36] U. Srivastava, J. Widom, Flexible time management in data stream systems, in: Proceedings of the Twenty-third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS '04, ACM, New York, NY, USA, 2004, pp. 263–274. doi:10.1145/1055558.1055596.
URL http://doi.acm.org/10.1145/1055558.1055596

[37] M. Jazayeri, R. Loos, D. R. Musser (Eds.), Generic Programming, International Seminar on Generic Programming, Dagstuhl Castle, Germany, April 27 - May 1, 1998, Selected Papers, Vol. 1766 of Lecture Notes in Computer Science, Springer, 2000.

[38] A. Arasu, S. Babu, J. Widom, The cql continuous query language: semantic foundations and query execution, VLDB J. 15 (2) (2006) 121–142.

[39] M. Stonebraker, U. Çetintemel, S. Zdonik, The 8 requirements of real-time stream processing, SIGMOD Rec. 34 (4) (2005) 42–47. doi:10.1145/1107499.1107504.
URL http://doi.acm.org/10.1145/1107499.1107504

[40] C. Boden, T. Rabl, V. Markl, Distributed machine learning-but at what cost?

[41] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, I. Stoica, Spark: Cluster computing with working sets, in: 2nd USENIX Workshop on Hot Topics in Cloud Computing, HotCloud'10, Boston, MA, USA, June 22, 2010, 2010.
URL https://www.usenix.org/conference/hotcloud-10/spark-cluster-computing-working-sets

[42] A. Thusoo, J. S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu, P. Wyckoff, R. Murthy, Hive: A warehousing solution over a map-reduce framework, Proc. VLDB Endow. 2 (2) (2009) 1626–1629. doi:10.14778/1687553.1687609.
URL https://doi.org/10.14778/1687553.1687609

[43] M. Balduini, E. Della Valle, A restful interface for RDF stream processors, in: E. Blomqvist, T. Groza (Eds.), Proceedings of the ISWC 2013 Posters & Demonstrations Track, Sydney, Australia, October 23, 2013, Vol. 1035 of CEUR Workshop Proceedings, CEUR-WS.org, 2013, pp. 209–212.

[44] P. S. Kalekar, Time series forecasting using holt-winters exponential smoothing, Kanwal Rekhi School of Information Technology 4329008 (2004) 1–13.

[45] M. Brambilla, S. Ceri, F. Daniel, G. Donetti, Temporal analysis of social media response to live events: The milano fashion week, in: J. Cabot, R. De Virgilio, R. Torlone (Eds.), Web Engineering: 17th International Conference, ICWE 2017, Rome, Italy, June 5-8, 2017, Proceedings, Springer International Publishing, Cham, 2017, pp. 134–150.

[46] M. Brambilla, S. Ceri, F. Daniel, G. Donetti, Spatial analysis of social media response to live events: The case of the milano fashion week, in: Proceedings of the 26th International Conference on World Wide Web Companion, WWW '17 Companion, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 2017, pp. 1457–1462.

[47] T. Arabghalizi, M. Brambilla, B. Rahdari, Analysis and knowledge extraction from event-related visual content on instagram, in: 3rd International Workshop on Knowlege Discovery on the Web, 2017, p. In print.

[48] B. Rahdari, T. Arabghalizi, M. Brambilla, Analysis of online user behaviour for art and culture events, in: Cross Domain Conference for Machine Learning and Knowledge Extraction, 2017, p. In print.

[49] C. Ratti, D. Frenchman, R. M. Pulselli, S. Williams, Mobile landscapes: using location data from cell phones for urban analysis, Environment and Planning B: Planning and Design 33 (5) (2006) 727–748.

[50] S. Mandal, G. Maji, Integrating telecom cdr and customer data from different operational databases and data warehouses into a central data warehouse for business analysis.

[51] D. Quercia, G. Di Lorenzo, F. Calabrese, C. Ratti, Mobile phones and outdoor advertising: measurable advertising, Institute of Electrical and Electronics Engineers, 2011.

[52] B. Hawelka, I. Sitko, E. Beinat, S. Sobolevsky, P. Kazakopoulos, C. Ratti, Geo-located twitter as proxy for global mobility patterns, Cartography and Geographic Information Science 41 (3) (2014) 260–271.

[53] F. Calabrese, G. Di Lorenzo, C. Ratti, Human mobility prediction based on individual and collective geographical preferences, in: Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on, IEEE, 2010, pp. 312–317.

[54] U. Habitat, Urbanization and development: emerging futures; world cities report 2016, Nairobi, UN Habitat.

[55] M. R. Sanderson, B. Derudder, M. Timberlake, F. Witlox, Are world cities also world immigrant cities? an international, cross-city analysis of global centrality and immigration, International Journal of Comparative Sociology 56 (3-4) (2015) 173–197.

[56] P. Deville, C. Linard, S. Martin, M. Gilbert, F. R. Stevens, A. E. Gaughan, V. D. Blondel, A. J. Tatem, Dynamic population mapping using mobile phone data, Proceedings of the National Academy of Sciences 111 (45) (2014) 15888–15893.

[57] J. R. Leimgruber, The management of multilingualism in a city-state, Multilingualism and language diversity in urban areas: Acquisition, identities, space, education 1 (2013) 227.

[58] O. García, J. A. Fishman, The multilingual apple: languages in New York City, Walter de Gruyter, 2001.

[59] G. Extra, K. YaÇğmur, Urban multilingualism in Europe: Immigrant minority languages at home and school, Vol. 130, Multilingual matters, 2004.

[60] D. Tasse, J. T. Chou, J. I. Hong, Generating neighborhood guides from social media., in: CitiLab@ ICWSM, 2016.

[61] M. Arnaboldi, M. Brambilla, B. Cassottana, P. Ciuccarelli, D. Ripamonti, S. Vantini, R. Volonterio, Studying multicultural diversity of cities and neighborhoods through social media language detection., in: CitiLab@ ICWSM, 2016.

[62] E. Bokányi, D. Kondor, L. Dobos, T. Sebők, J. Stéger, I. Csabai, G. Vattay, Race, religion and the city: twitter word frequency patterns reveal dominant demographic dimensions in the united states.

[63] M. Quaggiotto, D. Ricci, G. Scagnetti, G. Caviglia, D. Guido, M. Graffieti, S. Granados Lopez, New maps from the media-city. citymurmur as a tool for the visualization of urban space, in: Nouvelles cartographies, nouvelles villes. HyperUrbain.2, Europia Production, 2010.